

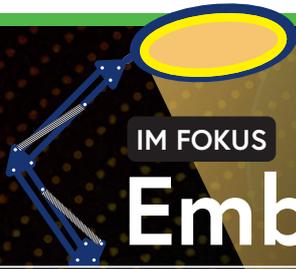


# elektor

design > share > earn

DIGITALE  
**BONUS**  
AUSGABE

608B  
SEIT 1961



IM FOKUS

## Embedded und KI

### Das vernetzte autonome Fahrzeug und seine Umgebung

Eine Einführung in die Welt autonomer Fahrzeuge

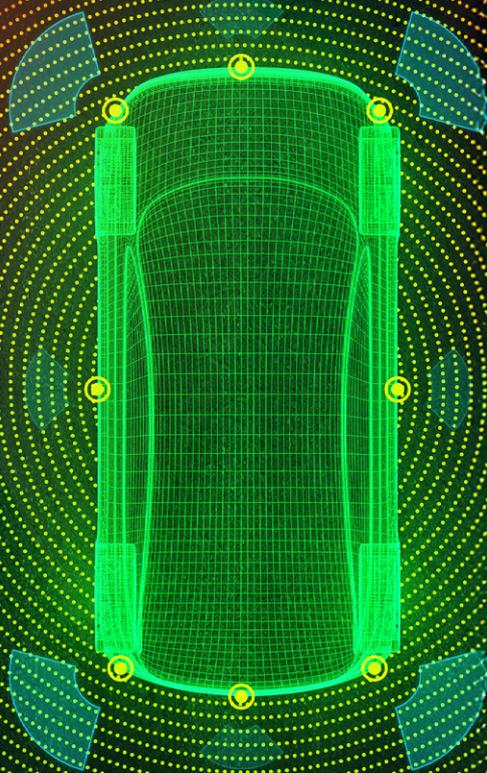
### Virtueller Assistent mit ChatGPT und Raspberry Pi

Entfesseln Sie die Macht der KI auf einfache Weise

### BEMERKENSWERTE

## BAUTEILE

Tastatur/Display-Schnittstellen-IC 8279 von Intel

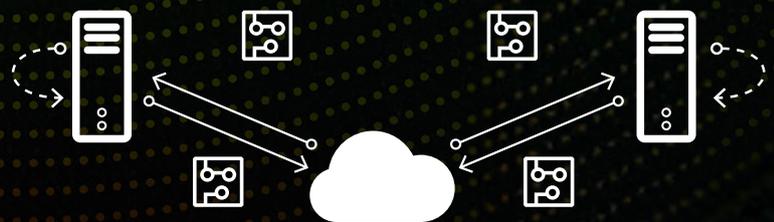


### Makerfabs SenseLoRa

Plug-and-Play-IoT für Gewächshäuser

### Infografik

Embedded und KI 2025



# Treten Sie jetzt der Elektor Community bei!



Jetzt



Mitglied werden!



- ✓ Zugang zum kompletten Online-Archiv (1970-heute)!
- ✓ 8x Elektor Magazin (gedruckt)
- ✓ 8x Elektor Magazin (PDF)
- ✓ 10% Rabatt im Elektor Store und exklusive Angebote
- ✓ Zugriff auf über 5.000 Gerber-Dateien  
u.v.m. aus der Projektplattform Elektor Labs



## Auch erhältlich

Die digitale  
Mitgliedschaft!



- ✓ Zugang zum kompletten Online-Archiv
- ✓ 8x Elektor Magazin (PDF)
- ✓ 10% Rabatt im Elektor Store und exklusive Angebote
- ✓ Zugriff auf über 5.000 Gerber-Dateien  
u.v.m. aus der Projektplattform Elektor Labs



[www.elektormagazine.de/abo](http://www.elektormagazine.de/abo)



# INHALT

## 3 Impressum

## 4 Virtueller Assistent mit ChatGPT und Raspberry Pi

Entfesseln Sie die Macht der KI auf einfache Weise



## 12 Das vernetzte autonome Fahrzeug und seine Umgebung

Eine Einführung in die Welt der autonomen Fahrzeuge

## 20 Infografik: Embedded und KI 2025

## 22 Bemerkenswerte Bauteile

Tastatur/Display-Schnittstellen-IC 8279 von Intel



## 24 Makerfabs SenseLoRa

Plug-and-Play-IoT für Gewächshäuser

Die Elektor-Ausgabe März/April 2025 ist am Kiosk und im Elektor-Store erhältlich.



## C. J. Abate

Content Director, Elektor

## Embedded und KI, heute und morgen

Eingebettete Systeme sind das Herzstück der modernen Technologie. Sie treiben alles an, von vorausschauenden Wartungssystemen in Fabriken bis hin zu lebensrettenden medizinischen Geräten. Mit der fortschreitenden Integration von KI werden eingebettete Lösungen immer intelligenter, effizienter und sind in der Lage, sofortige Entscheidungen zu treffen. In dieser Elektor-Bonusausgabe geben wir hilfreiche Einblicke in diese sich schnell entwickelnden Bereiche. Lesen Sie eine praktische Anleitung zum Bau eines virtuellen Assistenten mit ChatGPT auf Basis eines Raspberry Pi, eine Plug-and-Play-IoT-Lösung für die intelligente Verwaltung von Gewächshäusern und vieles mehr. Diese Projekte und Einblicke zeigen das grenzenlose Potenzial von eingebetteten Systemen in Kombination mit KI und IoT.

Wir freuen uns auch auf die embedded world 2025 in Nürnberg (11.-13. März 2025). Wenn Sie die Veranstaltung besuchen, kommen Sie an den Stand 5-181 und treffen Sie unsere Redakteure und Ingenieure. Gemeinsam können wir erkunden, wie eingebettete KI die Zukunft prägt. Und wenn Sie nicht persönlich dabei sein können, halten Sie Ausschau nach unserer Online-Berichterstattung.

Viel Spaß mit dieser Bonus-Ausgabe. Wir freuen uns darauf, Ihre Projekte auf der Online-Plattform Elektor Labs zu sehen!

## Unser Team

**Chefredakteur:** Jens Nickel (v.i.S.d.P.) | **Redaktion:** Asma Adhimi, Roberto Armani, Eric Bogers, Jan Buiting, Rolf Gerstendorf (RG), Ton Giesberts, Saad Imtiaz, Alina Neacsu, Dr. Thomas Scherer, Jean-Francois Simon, Clemens Valens, Brian Tristam Williams | **Regelmäßige Autoren:** David Ashton, Stuart Cording, Tam Hanna, Ilse Joostens, Prof. Dr. Martin Ossmann, Alfred Rosenkränzer | **Grafik & Layout:** Harmen Heida, Sylvia Sopamena, Patrick Wielders | **Herausgeber:** Erik Jansen | **Technische Fragen:** redaktion@elektor.de

## IMPRESSUM

55. Jahrgang, Nr. 608B, ISSN 0932-5468  
März/April 2025 Digitale Bonus-Ausgabe

Das Elektor Magazin wird 8 Mal im Jahr herausgegeben von  
**Elektor Verlag GmbH**  
Lukasstraße 1, 52070 Aachen (Deutschland)  
Tel. +49 (0)241 95509190  
www.elektor.de | www.elektormagazine.de

**Chefredakteur:** Jens Nickel (v.i.S.d.P.)

**Für alle Ihre Fragen:** service@elektor.de

**Mitglied werden:** www.elektormagazine.de/abo

**Anzeigen:** Büsra Kas  
Tel. +49 (0)241 95509178 - busra.kas@elektor.com  
www.elektormagazine.de/mediadaten

## Urheberrecht

© Elektor International Media b.v. 2025

Die in dieser Zeitschrift veröffentlichten Beiträge, insbesondere alle Aufsätze und Artikel sowie alle Entwürfe, Pläne, Zeichnungen einschließlich Platinen sind urheberrechtlich geschützt. Ihre auch teilweise Vervielfältigung und Verbreitung ist grundsätzlich nur mit vorheriger schriftlicher Zustimmung des Herausgebers gestattet. Die veröffentlichten Schaltungen können unter Patent- oder Gebrauchsmusterschutz stehen. Herstellen, Feilhalten, Inverkehrbringen und gewerblicher Gebrauch der Beiträge sind nur mit Zustimmung des Verlages und ggf. des Schutzrechtsinhabers zulässig. Nur der private Gebrauch ist frei. Bei den benutzten Warenbezeichnungen kann es sich um geschützte Warenzeichen handeln, die nur mit Zustimmung ihrer Inhaber warenzeichengemäß

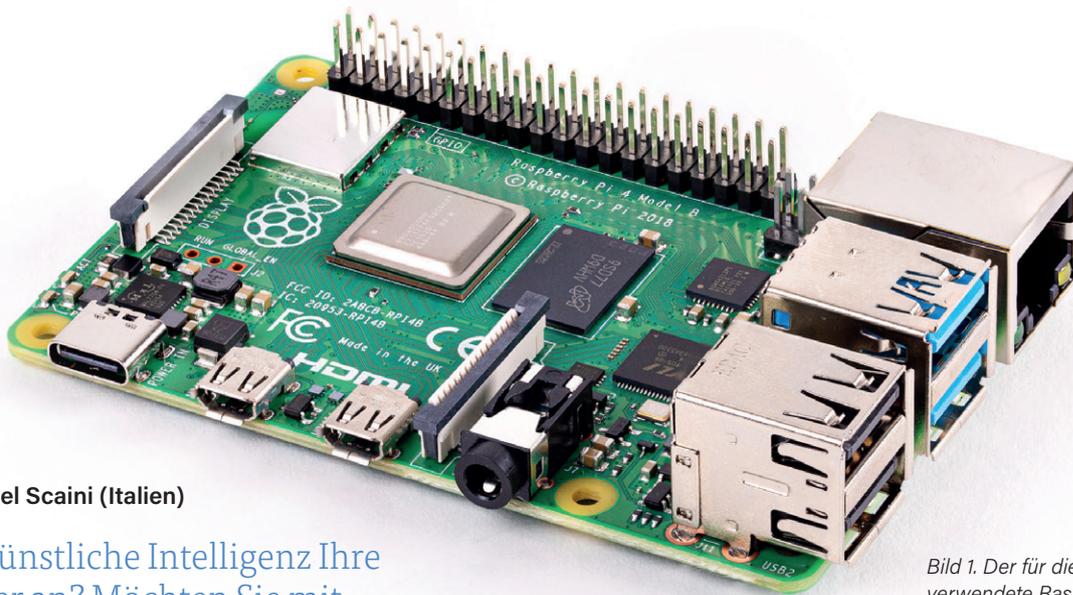
benutzt werden dürfen. Die geltenden gesetzlichen Bestimmungen hinsichtlich Bau, Erwerb und Betrieb von Sende- und Empfangseinrichtungen und der elektrischen Sicherheit sind unbedingt zu beachten. Eine Haftung des Herausgebers für die Richtigkeit und Brauchbarkeit der veröffentlichten Schaltungen und sonstigen Anordnungen sowie für die Richtigkeit des technischen Inhalts der veröffentlichten Aufsätze und sonstigen Beiträge ist ausgeschlossen.

## Distribution

IPS Pressevertrieb GmbH, Carl-Zeiss-Straße 5  
53340 Meckenheim (Deutschland)  
Tel. +49 (0)2225 88010

# Virtueller Assistent mit ChatGPT und Raspberry Pi

Entfesseln Sie die Macht der KI auf einfache Weise



Von Daniel Scaini (Italien)

Regt Künstliche Intelligenz Ihre Neugier an? Möchten Sie mit Experimenten beginnen? In diesem Artikel verwandeln wir einen Raspberry Pi mit Python und ChatGPT in ein leistungsstarkes intelligentes System. Folgen Sie dieser Schritt-für-Schritt-Anleitung, um einen fortschrittlichen virtuellen Assistenten zu erstellen, der bereit ist, auf natürliche und überraschende Weise zu interagieren.

Ein Thema, über das viel gesprochen wird, ist die künstliche Intelligenz (KI) und ihre Entwicklung in unserer Gesellschaft. Seit 1950, als der Mathematiker Alan Turing die Frage stellte, ob Computer denken können, hat sich die Technik der KI stark verändert. Der Begriff Künstliche Intelligenz selbst wurde 1956 eingeführt. Seitdem hat sich die künstliche Intelligenz in mehreren Schritten weiterentwickelt. Zuerst gab es die symbolische KI, die auf logischen Systemen basierte, die von Menschen entwickelt wurden. Danach folgte in den 1970er Jahren eine unproduktive Phase, die als „KI-Winter“ bekannt ist. In den 1990er Jahren trat dann der schachspielende Computer Deep Blue auf.

Bild 1. Der für dieses Projekt verwendete Raspberry Pi 4 (4 GB).

Seit 2011 haben Fortschritte im maschinellen Lernen – einem Teilbereich der KI, der auf statistischen Methoden basiert – die Fähigkeit von Computern verbessert, Vorhersagen aus historischen Daten zu treffen. Die Weiterentwicklung sogenannter „neuronaler Netze“ – einer Technik des maschinellen Lernens – zusammen mit der Speichermöglichkeit für große Datenmengen und eine erhöhte Rechenleistung hat die KI-Entwicklung vorangetrieben.

Heute sehen wir ihre Anwendung in vielen Bereichen. Sprachassistenten großer Hersteller wie IBMs Watson, Amazons Alexa, Google Home und Apples Siri sind bekannt. Diese Software kann Gespräche führen, die den menschlichen Dialog so gut wie möglich nachahmen, dank Techniken der natürlichen Sprachverarbeitung (Neuro-Linguistisches Programmieren, NLP).

Es gibt Tools, die viele Informationen bieten und bei alltäglichen Aufgaben helfen können; in der Medizin, im Automobil- und in vielen anderen Bereichen. Bis heute wurden generative vortrainierte Transformatoren (Generative pre-trained transformers, GPT) entwickelt, wie zum Beispiel ChatGPT, dem wir im Folgenden mehr Aufmerksamkeit schenken werden.

Mit diesen Tools ist eine direkte Interaktion möglich. Sie können Inhalte verstehen, zusammenfassen, generieren und vorhersagen. Die Idee des Projekts in diesem Artikel ist es, von einer Plattform wie dem Raspberry Pi aus mit künstlicher Intelligenz zu interagieren. Der Raspberry Pi,

der nur mit einem Mikrofon, Kopfhörern oder PC-Lautsprechern ausgestattet ist und ständig mit dem Internet verbunden bleibt, wird die Verbindung zur KI herstellen.

## Hardware

Die Hardware für dieses Projekt ist einfach und besteht aus drei Hauptkomponenten. Die erste und wichtigste Komponente ist der Raspberry Pi 4; die Basis, auf der wir das gesamte System aufbauen. Dieses Modell verfügt über 4 GB RAM und eine moderne Audio- und Videoverarbeitung. Er stellt die vierte Produktgeneration der Raspberry Foundation dar und ist ideal für Programmierer, Entwickler und Gamer. Dank zahlreicher GPIO-Pins eignet er sich aber auch besonders für komplexe IoT-Projekte.

Durch seinen ARM-Prozessor liefert das Board eine Leistung, die mit der eines Einsteiger-PCs vergleichbar ist, und es kann sogar 4K-Video-streams verarbeiten. Es ist deshalb ein wichtiges Gerät für alle, die ein solches Projekt starten möchten, und bietet viele Anschlüsse für Peripheriegeräte wie Kameramodule, Sensoren, GPS, LCDs und mehr. Der Raspberry Pi bietet also zahlreiche Pins für die 3,3-V-Spannungsversorgung sowie den Anschluss von Peripheriegeräten, Relais und Sensoren. Diese Anschlüsse können leicht über verschiedene Hersteller-Bibliotheken in Python oder C++ verwaltet werden (**Bild 1**). Als Betriebssystem empfehlen wir *Raspbian OS 64 bit*, um nach der Installation der benötigten Pakete unerwartete Probleme zu vermeiden. Zusätzlich haben wir ein USB-Mikrofon bereitgestellt, das sich ideal für unser System sowie für PCs und Macs eignet. Dieses Mikrofon hat einen Frequenzgang von 100 Hz bis 16 kHz und eine Empfindlichkeit von -67 dBV/μBar (entsprechend -47 dBV/Pascal) (±4 dB).

Zwei der wichtigsten Vorteile der Wahl dieses Mikrofons sind die Reduzierung unerwünschter Hintergrundgeräusche und die einfache Plug-and-Play-Installation, die den Einsatz externer Treiber überflüssig macht.

Als dritte und letzte Komponente haben wir ein Paar PC-Lautsprecher oder einfache Kopfhörer vorgesehen, die wir an die Audiobuchse des Raspberry Pi anschließen werden.

## Programmiersprache

Die Programmiersprache Python ist objektorientiert und eignet sich für verschiedene Arten der Softwareentwicklung. Ihre Benutzerfreundlichkeit und die Möglichkeit der Integration mit anderen Sprachen sind hervorsteckende Merkmale. Python ist klar und leistungsfähig, ähnlich wie Java oder Perl, und bietet eine elegante Syntax, die das Lesen von Programmen erleichtert und die Erstellung komplexer Projekte ohne Beeinträchtigung der Wartbarkeit ermöglicht.

Laut mehreren Umfragen wird Python heute von 84 % der Entwickler als primäre Programmiersprache genutzt. Besonders stark vertreten sind dabei die Bereiche Datenanalyse und Webentwicklung, doch auch im maschinellen Lernen, Computergrafik und Spieleentwicklung gewinnt Python an Bedeutung.

Obwohl Python insbesondere hinsichtlich der RAM- und CPU-Auslastung nicht so optimal ist wie C++, wurde Python gewählt, da es quellen offen ist, viele Bibliotheken ohne zusätzliche Kosten bietet und auf allen gängigen Betriebssystemen (Linux, Mac, Windows) lauffähig ist. In unserem Fall wird es unter Raspbian (oder Raspberry Pi OS, wie es

neuerdings heißt) betrieben, einem Betriebssystem, das auf offiziellen Debian-Versionen basiert und für die ARM-Architektur des Raspberry Pi angepasst wurde. Unsere Software könnte auch leicht unter Windows laufen, ohne dass umfangreiche Änderungen erforderlich wären. Die am meisten verbreitete Version von Python ist V 3.6, die häufig mit IDEs oder Editoren wie der Professional/Community-Edition von PyCharm oder VS Code verwendet wird.

Angesichts der wenigen Zeilen, die zu schreiben sind, sollten Sie einen Editor nur dann installieren, wenn Sie ein Einsteiger in die Materie sind. Andernfalls können Sie problemlos Nano, VI oder den auf dem Raspberry Pi vorinstallierten Texteditor verwenden.

## ChatGPT

Was ist ChatGPT genau? ChatGPT ist ein leistungsfähiger digitaler Assistent, der künstliche Intelligenz nutzt, um ein natürliches Gesprächserlebnis zu schaffen. Es handelt sich technisch gesehen um ein großes Sprachmodell (Large Language Model, LLM), einen generativen Algorithmus der künstlichen Intelligenz, der Deep-Learning-Techniken und große Datensätze verwendet, um Inhalte zu verstehen, zusammenzufassen, zu generieren und vorherzusagen.

Das Produkt von OpenAI, das 2015 in San Francisco entwickelt wurde, sorgt seit einigen Wochen für viel Aufregung in den Medien. Das liegt zum einen an der vorübergehenden Sperrung der Software zur Erstellung von Inhalten in Italien, die später nach der Anpassung an die europäischen Datenschutzbestimmungen wieder aufgehoben wurde. Zum anderen sind die Ereignisse rund um den Ausstieg und die Rückkehr von CEO Sam Altman ein großes Thema. Diese Entwicklungen haben das Interesse und die Diskussion über OpenAI und seine strategischen Entscheidungen weiter angefacht.

Aber was genau ist generative künstliche Intelligenz und was bedeutet das große Sprachmodell? Generative künstliche Intelligenz ist ein Bereich der KI, die Modelle entwickelt, die eigenständig Texte, Bilder und andere Daten erzeugen kann. Ein Beispiel dafür ist GPT, ein Modell, das auf transformatorbasierter Technologie beruht und durch überwachte Lernprozesse mit einer großen Menge an Text trainiert wurde. Es kann Inhalte erstellen, automatische Übersetzungen machen, Informationen bereitstellen und in Echtzeit Antworten geben. Das ist besonders nützlich für Menschen in der Kommunikation und im Marketing, aber auch in anderen Branchen.

ChatGPT hat aber ein viel größeres Potenzial. Anders als eine Suchmaschine oder ein Sprachassistent nutzt dieses Tool künstliche Intelligenz, die ein Verfahren namens Pre-Training anwendet, um den Kontext eines Gesprächs zu erfassen und Antworten zu erzeugen. Die Software wurde mit einer großen Menge an Text aus Büchern, Artikeln, Websites und Gesprächen zwischen Menschen trainiert.

Zu den großen Vorteilen dieses Ansatzes gehören sicherlich ein breites Wissen, eine schnelle Reaktion und ein einfacher Zugang durch eine benutzerfreundliche Oberfläche. ChatGPT kann in vielen Bereichen eingesetzt werden, um die Interaktion mit Nutzern zu verbessern und Aufgaben zu automatisieren. Dazu gehören die folgenden Punkte:

- **Finanzdienstleistungen:** ChatGPT kann virtuelle Unterstützung im Finanzbereich bieten, Fragen zu Konten beantworten, einfache Anlageberatung erteilen oder bei der Finanzplanung helfen

- **Gesundheitsfürsorge:** ChatGPT unterstützt Fachkräfte im Gesundheitswesen, indem es grundlegende Informationen zu Symptomen, Antworten auf häufige Fragen, Tipps zum Umgang mit Krankheiten oder Ratschläge für einen gesunden Lebensstil bereitstellt
- **Ausbildung:** ChatGPT kann als virtueller Tutor eingesetzt werden, um detaillierte Erklärungen zu akademischen oder beruflichen Themen zu geben, Fragen von Schülern zu beantworten, Übungsaufgaben anzubieten oder interaktive Lernszenarien zu erstellen
- **Kundensupportdienste:** ChatGPT kann den Kundensupport automatisieren und verbessern, indem es sofortige Antworten auf häufig gestellte Fragen, Unterstützung bei der Lösung technischer Probleme oder Informationen über Produkte oder Dienstleistungen liefert
- **Reise- und Tourismussektor:** ChatGPT kann bei der Reiseplanung helfen, indem es Informationen über Flüge, Hotels, Sehenswürdigkeiten und Restaurants bereitstellt, individuelle Reiserouten vorschlägt oder bei der Buchung von Tickets und Unterkünften behilflich ist

Einige große Unternehmen wie Mastercard, KLM und Spotify haben bereits Vereinbarungen mit diesem Produkt beziehungsweise seinem Anbieter getroffen. So können sie zum Beispiel einen Kundenservice einrichten oder, im Fall von Mastercard, Unterstützung bei Finanztransaktionen und bei Fragen zur Sicherheit von Kreditkarten erhalten.

## Dienste und APIs

Bei ChatGPT gibt es neben der normalen Web-Oberfläche auch einen Modus, der über eine REST-API funktioniert und häufig von Unternehmen (wie den gerade genannten) verwendet wird. Eine API (Application Programming Interface) besteht aus Regeln, die festlegen, wie Anwendungen oder Geräte miteinander kommunizieren können. Eine REST-API folgt den Prinzipien von REST (Representational State Transfer) und wird deshalb manchmal auch als RESTful API bezeichnet. REST wurde 2000 von Roy Fielding in seiner Dissertation vorgestellt und bietet Entwicklern viel Flexibilität und Freiheit. Diese Flexibilität ist ein Grund, warum REST-APIs die beliebteste Methode ist, um Komponenten und Anwendungen in einer Microservice-Architektur zu verbinden.

Eine API ist im Grunde ein Weg, wie eine Anwendung oder ein Dienst auf eine Ressource einer anderen Anwendung oder einem anderen Dienst zugreifen kann. Die Anwendung oder der Dienst, die den Zugriff anfordert, nennt man *client*, und die Anwendung oder der Dienst, die die Ressource bereitstellt, nennt man *server*.

REST-APIs nutzen HTTP-Anfragen, um grundlegende Datenbankfunktionen wie das Erstellen, Lesen, Aktualisieren und Löschen von Datensätzen durchzuführen. Diese Funktionen werden auch als CRUD bezeichnet, was für *create, read, update and delete* steht. Zum Beispiel verwendet eine REST-API eine GET-Anfrage, um einen Datensatz abzurufen, eine POST-Anfrage, um ihn zu erstellen, eine PUT-Anfrage, um ihn zu aktualisieren, und eine DELETE-Anfrage, um ihn zu löschen. Alle HTTP-Methoden können in API-Aufrufen eingesetzt werden.

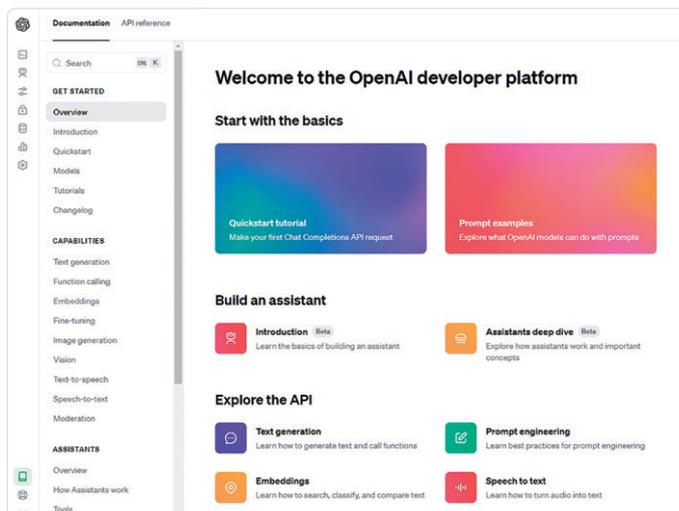


Bild 2. Die OpenAI-Webseite für Entwickler.

Eine gut gestaltete API ähnelt einer Website, die in einem Webbrowser mit eingebauten HTTP-Funktionen läuft. Man kann sie in GET und POST unterteilen. Der Hauptunterschied ist, dass bei GET die Daten sichtbar in der URL stehen, während sie bei POST im Body-Bereich der Webseite stehen und oft nicht auf den ersten Blick zu sehen sind. Für unseren Dienst nutzen wir genau diese POST-Methode. Wenn wir uns auf der Webseite [1] einloggen, können wir über diese Schnittstelle die angebotenen Dienste und Funktionen entdecken (**Bild 2**). Es gibt APIs für Texterzeugung, Prompt-Engineering, Einbettungen, Sprache-zu-Text, Bilderzeugung, Feinabstimmung, Text-zu-Sprache und Vision. Der erste Schritt zur Nutzung der Dienste ist die Erstellung eines neuen Schlüssels. Von der *Overview* [1] aus können Sie oben rechts auf *Dashboard* klicken und dann im Seitenmenü auf *API Keys*. Als Nächstes klicken Sie auf *Create new secret key* (**Bild 3**).

Lassen Sie alle Einstellungen, so wie sie sind, und setzen Sie die Berechtigungen auf *All*. Dann fügen Sie den Namen hinzu, den Sie dem API-Schlüssel geben möchten, und kopieren die Zeichenfolge, die nach dem Druck auf Create erscheint. Diese Zeichenkette ist nur zu diesem Zeitpunkt sichtbar, daher sollten Sie sie sofort auf einem ganz analogen Notizblock oder ähnlichem speichern (**Bild 4**).

Und nun der wichtigste Punkt; es geht ums liebe Geld: Ursprünglich gab es eine kostenlose Premium-Phase, in der OpenAI 5 \$ zur Verfügung stellte. Diese Regelung wurde jedoch kürzlich geändert. Daher ist es nötig, über das Menü auf der linken Seite wieder den Bereich *Usage* aufzurufen. Dort finden Sie auf der rechten Seite die Option *Increase limit*. Folgen Sie den Anweisungen, um Ihre Kreditkarte zu verknüpfen und Guthaben zu erwerben (**Bild 5**). Sogar ein geringer Betrag von nur 5 € reicht für Bildungszwecke aus, da jede Anfrage in wirtschaftlicher Hinsicht nur sehr wenig kostet.

## Konfiguration des Betriebssystems

Sie werden nun das Betriebssystem auf unserem Raspberry Pi installieren. Für diesen Schritt werden zwei Dinge benötigt:

- MicroSD-Karte mit mindestens 16 GB bis 32 GB
- das Tool Raspberry-Pi-Imager

Die Website, von der der Imager heruntergeladen werden kann, finden Sie unter [2]. Oben sehen Sie den Imager, das Tool, das Sie brauchen, um eine bootfähige SD-Karte zu erstellen, mit der der Raspberry Pi gestartet werden kann. Unter *Manually install an operating system*

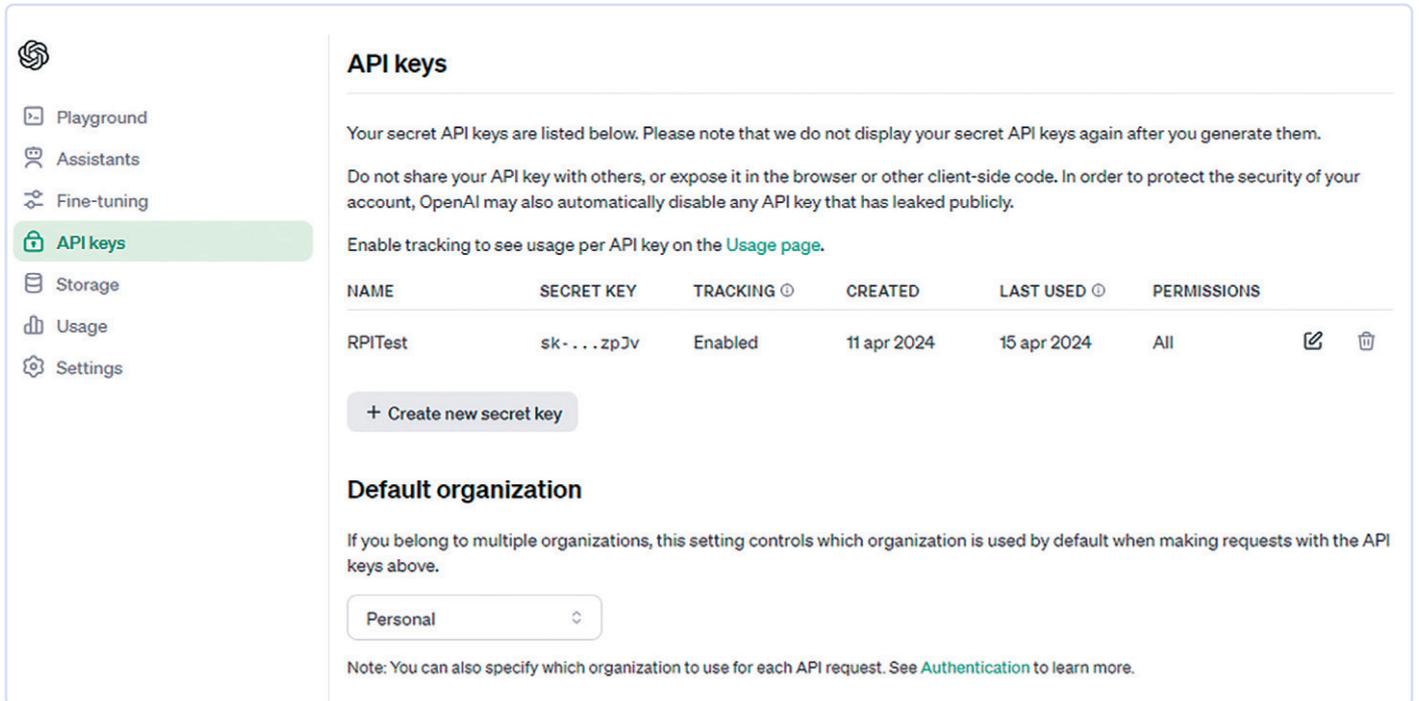


Bild 3. Seite zur Erstellung des Schlüssels, um mit dem System interagieren zu können.

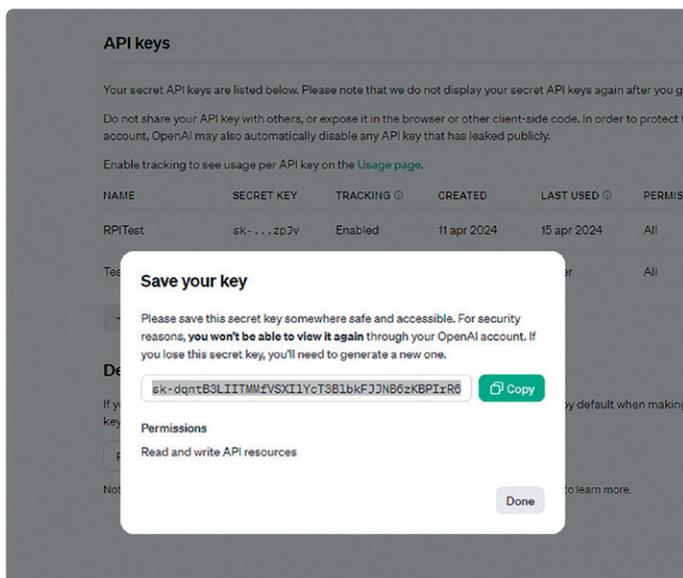


Bild 4. Nicht verlieren: Speichern Sie Ihren Schlüssel zur späteren Verwendung!

*image* finden Sie den Link zu verschiedenen gepackten OS-Dateien für verschiedene Raspberry-Pi-Versionen. Der korrekte und direkte Link zum Herunterladen des Pakets ist [3]. Bitte beachten Sie, dass es sich um eine 1,2 GB große Datei handelt, die bei langsameren Internetverbindungen etwas Zeit in Anspruch nehmen kann. Sobald der Imager installiert und gestartet ist, stecken Sie die SD-Karte ein und klicken auf **OS WÄHLEN (Bild 6)**. Hier wählen Sie den letzten Punkt, also *Use Custom*, und wählen dann die vorher heruntergeladene gezippte Datei aus.

Als Nächstes klicken Sie auf **SD-KARTE WÄHLEN** und wählen das Laufwerk aus, auf dem wir das Betriebssystem erstellen möchten. Schließlich klicken Sie auf **SCHREIBEN**. Sie sollten die Standard-einstellungen ändern, damit Sie sich beim ersten Booten mit den eigenen Anmeldedaten anmelden können.

In unserem Fall haben wir, wie im Screenshot in **Bild 7** zu sehen ist, den Benutzernamen „daniel“ und das Passwort *12345678* gewählt. Nach kurzer Zeit ist die SD-Karte fertig: Stecken Sie die Karte in den Raspberry Pi, schließen Sie ihn an ein Netzwerk mit Internetzugang an und schalten Sie ihn ein. Es ist sehr nützlich, einen Remote-Desktop



Bild 5. Für die Nutzung des Dienstes ist ein Guthaben erforderlich.

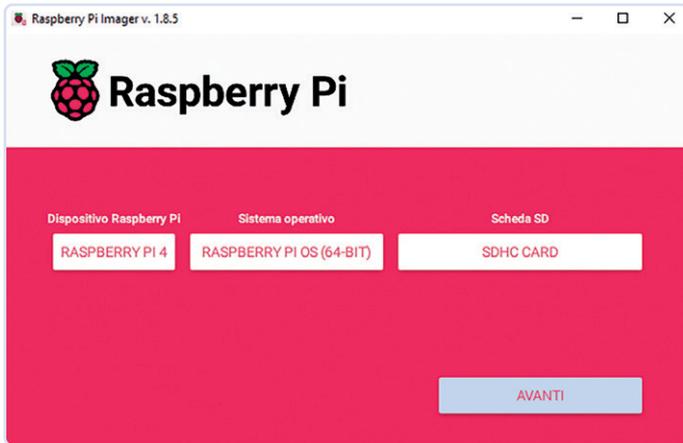


Bild 6. Screenshot des Raspberry-Pi-Imagers.

wie `xrdp` zu installieren, damit der Raspberry Pi ohne zusätzlichen Monitor genutzt werden kann:

```
sudo apt-get install xrdp -y
sudo apt-get update
sudo apt-get upgrade
```

Jetzt haben Sie alles, was Sie brauchen, um mit dem Pi aus der Ferne zu kommunizieren. Sie können den Monitor vergessen und direkt vom PC aus arbeiten, wenn Sie möchten. Installieren Sie die grundlegenden Bibliotheken für die Entwicklung: *Python*, um das Projekt zusammenzustellen, und *libsox*, die hilft, das angeschlossene Mikrofon zu betreiben.

```
sudo apt install python3 python3-pip python3-venv
sudo apt-get install sox libsox-fmt-mp3
```

Erstellen Sie nun einen Arbeitsordner namens *openai*, in dem eine virtuelle Umgebung namens *env* installiert wird, damit die notwendigen Python-Bibliotheken installiert werden können, ohne das gesamte System zu gefährden.

```
mkdir openai
cd openai
python -m venv env
```

Greifen Sie nun auf diese Umgebung zu über den Befehl

```
source env/bin/activate
```

Der Name der Umgebung ist frei wählbar; statt *env* hätte man sie auch *env\_openai* nennen können. Nach der Aktivierung installieren Sie die Python-Bibliotheken:

**Google Speech** kann mit der Google Cloud arbeiten und eine einfache Lösung für Sprache-zu-Text darstellen, die es ermöglicht, Text aus Sprache zu entnehmen:

```
pip install google-speech
```

**SpeechRecognition** greift auf das USB-Mikrofon zu, ohne dass man sich mit Bits, der Kalibrierung des Mikrofons oder anderen Feinheiten ablagen müsste:

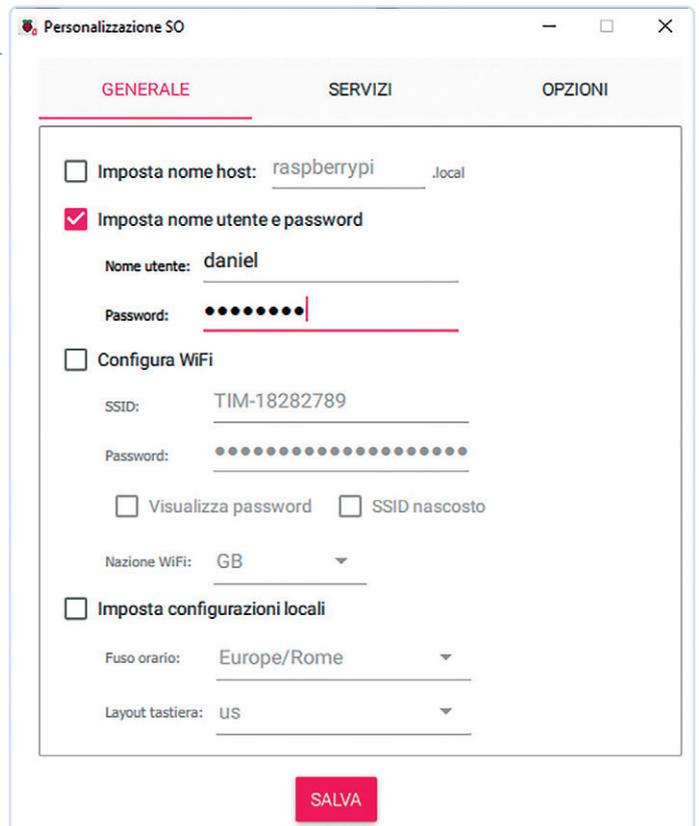


Bild 7. Bei der ersten Anmeldung müssen Sie Ihre Anmeldedaten ändern.

```
pip install SpeechRecognition
pip install PyAudio
```

**OpenAI**, um mit der oben beschriebenen API interagieren zu können und Antworten von der künstlichen Intelligenz zu erhalten:

```
python3 -m pip install openai
```

Die Konsole wird jetzt nicht geschlossen, da sie später zum Ausführen des Codes erforderlich ist. Aber es ist kein Problem, wenn die Konsole doch geschlossen ist, dann gehen Sie einfach in den richtigen Ordner (erster Befehl) und geben ein, um den richtig konfigurierten Bereich wieder zu erreichen:

```
cd openai
python -m venv env
source env/bin/activate
```

## Code

Prozedurale oder objektorientierte Programmierung? Die erste Option ist zwar schneller und einfacher erstellt, aber aus verschiedenen Gründen nicht die beste Wahl. Die objektorientierte Programmierung (OOP) beruht darauf, eine Reihe von Objekten zu schaffen, die miteinander kommunizieren können (auf gut Englisch *message exchange*). Jedes Objekt hat seinen eigenen Zustand und seine eigenen Daten, ähnlich wie in der realen Welt.

Der Begriff *message exchange* beschreibt, wie Objekte die öffentlichen Methoden anderer Objekte aufrufen. Sie können ihnen Daten zur Verarbeitung geben und das Ergebnis zurückerhalten. Im Gegensatz zur prozeduralen Programmierung mit Codeblöcken (Unterprogrammen) erlaubt die objektorientierte Programmierung, mehrere Funktionen in

einer Klasse zu bündeln, die Variablen und Funktionen vereint. Weitere Vorteile der objektorientierten Programmierung sind:

- › natürliche Unterstützung für die Software-Modellierung von realen Objekten oder des abstrakten Modells, das reproduziert werden soll
- › einfachere Verwaltung und Pflege von großen Projekten
- › die Organisation des Codes in Form von Klassen fördert die Modularität und die Wiederverwendung von Code

Nun kommen Sie zum Schreiben des eigentlichen Codes und erstellen die Datei, die für die Interaktion mit der KI genutzt wird. Dazu geben Sie in der Konsole, in der Sie sich befinden, ein:

```
touch gpt.py
```

Als Nächstes wird eine Datei erstellt, die mit dem Standard-Editor des Betriebssystems geöffnet wird. In unserem Fall, wenn Sie über die GUI oder Remote Desktop zugreifen, ist das der Texteditor Geany. Wie in allen Python-Projekten müssen Verweise auf verwendete Bibliotheken eingefügt werden:

```
import speech_recognition as sr
from openai import OpenAI
from google_speech import Speech
```

Zur Demonstration werden hier alle Klassen (in diesem Fall nur eine) zusammen mit dem `main`-Code in einer Datei untergebracht. Dann definieren Sie die `OpenAICustom`-Klasse, die einige Konstanten enthält: das KI-Modell, das Sie nutzen werden, die Sprache, in der Sie kommunizieren und eine letzte Konstante, die es ermöglicht, die Ausgabe zu steuern.

```
class OpenAICustom:
    GPT_MODEL = "gpt-3.5-turbo"
    GPT_LANG = "en"
    isDebug = True
```

Der Konstruktor erhält den Pointer auf das Mikrofon und den Key-String für die API und die ChatGPT-Welt. Hier werden die beteiligten Entitäten und die Nachricht eingerichtet, die Sie dann austauschen und um verschiedene Anfragen erweitern werden.

```
def __init__(self, sr, keys):
    self.recognizer_instance = sr.Recognizer()
    self.key = keys
    self.client = OpenAI(
        api_key = keys
    )
    self.messages = [
        {
            "role": "system",
            "content":
                "You are a helpful assistant"
        }
    ]
```

Jetzt fügen Sie eine Methode hinzu, um eine der außerhalb der Klasse erstellten Instanzen zu exportieren. So sieht die Entität aus, die das Mikrofon steuert:

```
def getReco(self):
    return self.recognizer_instance
```

Eine wichtige Funktion dieser Klasse ist die Kommunikation mit der ChatGPT-API, die bereits besprochen wurde. Sie muss Nachrichteneingaben entgegennehmen und die Objekte nutzen, die im Konstruktor festgelegt sind. Zuerst fügen Sie dem Nachrichten-Array ein neues Element hinzu, das einen `json`-Ausdruck mit der Rolle und dem Inhalt des Benutzers enthält.

Als nächstes erstellen Sie die eigentliche Anfrage an die OpenAI-Server, die als Antwort ein Objekt liefern, das in der `chat`-Variablen gespeichert wird. Dieses sieht wie folgt aus:

```
{
  "choices": [
    {
      "finish_reason": "stop",
      "index": 0,
      "message": {
        "content": "RISPOSTA DI CHATGPT",
        "role": "assistant"
      },
      "logprobs": null
    }
  ],
  "created": 1677664795,
  "id": "chatcmpl-7YqpwdfhqwajicIEznoc6Q47XAyW",
  "model": "gpt-3.5-turbo",
  "object": "chat.completion",
  "usage": {
    "completion_tokens": 17,
    "prompt_tokens": 57,
    "total_tokens": 74
  }
}
```

Die Nachricht befindet sich im ersten Objekt des `choices`-Arrays. Sie können die Nachricht in der Debug-Konsole ausgeben oder auch nicht, und sie dann an die Funktion weitergeben, die den Text in Sprache umwandelt.

```
def sendToAI(self, message):
    self.messages.append(
        {
            "role": "user",
            "content": message
        },
    )
    chat = self.client.chat.completions.create(
        messages=self.messages,
        model=self.GPT_MODEL
```

```

)
reply = chat.choices[0].message
if self.isDebug:
    print("Assistant: ", reply.content)
self.messages.append(reply)
speech = Speech(reply.content, self.GPT_LANG)
speech.play()

```

Nachdem die Klasse, um die es im gesamten Code geht, richtig konstruiert wurde, erstellen Sie die `main`-Klasse, die den Benutzer tatsächlich anruft und mit ihm interagiert. Die Variable `SYS_LANG` legt die Sprache fest, die erkannt werden soll.

Als Nächstes nehmen Sie den OpenAI-Schlüssel als Eingabe und setzen ihn in eine Endlosschleife, in der Sie weiter zuhören, den Benutzer verstehen und entsprechend reagieren.

```

SYS_LANG = "it-IT"
in_key = input("Enter your OpenAI key: ")
myObj = OpenAICustom(sr, in_key)
while True:
    with sr.Microphone() as source:
        myObj.getReco().adjust_for_ambient_noise(source)
        print("I am listening... go ahead and talk!")
        audio = myObj.getReco().listen(source)
        print("Ok! I am now processing the message!")
        try:
            text = myObj.getReco().
                recognize_google(audio, language=SYS_LANG)
            print("Google understood: \n", text)
            myObj.sendToAI(text)
        except Exception as e:
            print(e)

```

Bevor Sie den Code ausführen, sollten Sie sich an die Regel erinnern, die am Ende des Abschnitts über die Konfiguration des Betriebssystems aufgestellt wurde. Das heißt, Sie müssen sich im virtuellen Raum mit allen Bibliotheken befinden. Um das Programm schließlich in der Konsole zu starten, geben Sie den Befehl ein:

```
python3 gpt.py
```

## Künftige Entwicklungen

Eine der einfachsten Weiterentwicklungen für das vorgeschlagene Projekt ist seine Miniaturisierung. So können Kosten und Stromaufnahme gesenkt werden. Die Portabilität eines Geräts wie dem ESP32 könnte einen echten Vorteil bringen und viele weitere Anwendungsmöglichkeiten eröffnen.

In diesem Zusammenhang gibt es zwei Ansätze: Der erste, einfachere Ansatz ist, dass der ESP32 nur als Gateway für die Übermittlung von Informationen dient, während die gesamte Verarbeitung an einen Server oder eine App abgegeben wird. In diesem Fall ist die Rechenlast gering, aber die Entwicklung ist aufwendiger und der Chip arbeitet nicht autonom.

Ein zweiter, viel umfassenderer Ansatz wäre, alle Teile auf einem Chip zu vereinen: einen Teil für den Empfang über ein Mikrofon, einen Teil zur Sprachverarbeitung, um Sprache in Text umzuwandeln, und schließlich eine direkte Verbindung zur ChatGPT-API.

Das größte Problem wird sicher die Umwandlung von Sprache in Text sein. Das könnte den Speicher des Chips stark beanspruchen, da dafür eine Umwandlung in base64 oder Ähnliches nötig ist. Gerüchten zufolge könnte die Interaktion mit der ChatGPT-Engine auch über den Dienstanbieter IFTTT möglich sein, also ganz im Sinne des IoT. Der Code zur Implementierung dieses Sprachassistenten kann auf der Elektor-Labs-Seite für dieses Projekt heruntergeladen werden [4].

Übersetzung: Mahy Arafa — 240619-02

## Sie haben Fragen oder Kommentare?

Haben Sie technische Fragen oder Anmerkungen zu diesem Artikel? Wenden Sie sich bitte an die Elektor-Redaktion unter [redaktion@elektor.de](mailto:redaktion@elektor.de).



## Passendes Produkt

> **Raspberry Pi 4 (4 GB) Offizielles Starterkit**  
[www.elektor.de/20556](http://www.elektor.de/20556)

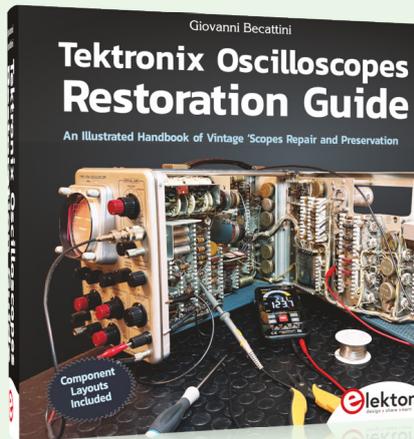


## WEBLINKS

- [1] Entwickler-Seite der OpenAI-Plattform: <https://platform.openai.com/docs/overview>
- [2] Download des Raspberry-Pi-Imagers: <https://www.raspberrypi.com/software>
- [3] Download des Raspberry- Pi-OS: [https://downloads.raspberrypi.org/raspios\\_armhf\\_latest](https://downloads.raspberrypi.org/raspios_armhf_latest)
- [4] Elektor-Labs-Webseite für dieses Projekt: <https://t1p.de/17lzq>

## Tektronix Oscilloscopes Restoration Guide

Tektronix-Oszilloskope sind wahre Meisterwerke der Elektronik und haben der Menschheit in allen Bereichen der Wissenschaft geholfen, überall dort voranzukommen, wo ein physikalisches Phänomen beobachtet und untersucht werden musste. Sie halfen dem Menschen, den Mond zu erreichen, fanden die Ursache von Flugzeugabstürzen und ebneten den Weg für Tausende anderer Entdeckungen.



Preis: 59,95 €

**Mitgliederpreis: 53,96 €**

[www.elektor.de/21051](http://www.elektor.de/21051)

## SunFounder GalaxyRVR Mars Rover Kit für Arduino

Das SunFounder GalaxyRVR Mars Rover Kit wurde entwickelt, um die Funktionalität echter Marsrover nachzuahmen und bietet ein praktisches Erlebnis, das sowohl lehrreich als auch aufregend ist. Der GalaxyRVR ist mit Uno R3 kompatibel und für die problemlose Navigation in unterschiedlichem Gelände geeignet.



Preis: 134,95 €

**Mitgliederpreis: 121,46 €**

[www.elektor.de/21061](http://www.elektor.de/21061)

## Raspberry Pi 5 (16 GB RAM)



**Preis: 139,95 €**

[www.elektor.de/21080](http://www.elektor.de/21080)

## LILYGO T-Panel S3 Development Board



Preis: 89,95 €

**Mitgliederpreis: 80,96 €**

[www.elektor.de/21026](http://www.elektor.de/21026)

# Das vernetzte autonome Fahrzeug und seine Umgebung

Eine Einführung in reale und verkleinerte autonome Fahrzeuge

Quelle: Adobe Stock

Von Jacques Ehrlich (Frankreich)

Mit zunehmender globaler Vernetzung und Netzwerklatenzen, die auf wenige Millisekunden sinken, werden selbstfahrende Fahrzeuge schnell zu Bestandteilen im universellen Ökosystem des Internets der Dinge. In dieser Einleitung zu seinem demnächst erscheinenden Elektor-Buch führt Jacques Ehrlich in die Terminologie des autonomen Fahrens ein und erklärt, wie sich die Technik dieser Fahrzeuge parallel zur Entwicklung der Infrastruktur, die sie beherbergt, weiterentwickelt.

**Anmerkung der Redaktion:** Dieser Artikel ist ein Auszug aus dem Elektor-Buch „*The Connected Autonomous Vehicle and Its Environment*“ das 2025 bei Elektor erscheinen wird. Er wurde so formatiert und leicht bearbeitet, damit er den Konventionen und dem Seitenlayout der Zeitschrift Elektor genügt. Für Rückfragen stehen der Autor und der Herausgeber gerne zur Verfügung. Die Kontaktdaten finden Sie im Textkasten **Fragen oder Kommentare?**

Das autonome Fahrzeug ist eine komplexe und sich ständig weiterentwickelnde Technologie, und das Ziel dieses Buches ist es, ihre vielen Facetten zu beleuchten.

Zu lange wurde das Fahrzeug als isolierte Einheit betrachtet - getrennt von anderen Fahrzeugen und von seiner Umgebung, nämlich der Infrastruktur. Diese isolationistische Sichtweise ist völlig überholt. Moderne Fahrzeuge mit hoch entwickelten Fahrassistenzsystemen (Advanced Driver Assistance Systems, ADAS) und autonome Fahrzeuge werden zu hochgradig kooperativen Systemen. In naher Zukunft werden sie über verschiedene Kommunikationskanäle kontinuierlich Informationen untereinander (Vehicle to Vehicle, V2V) und mit der Infrastruktur (Vehicle to Infrastructure, V2I) austauschen und sich nahtlos in die Verkehrstelematik oder das intelligente Verkehrssystem (Intelligent Transport System, ITS) integrieren.

## Intelligentes Verkehrssystem

ITS kann als ein System definiert werden, das alle Kommunikations-, Steuerungs- und Informationsverarbeitungstechnologien umfasst, um den Nutzern des Verkehrssystems sichere und effiziente Mobilitätsdienste und den Straßendiensten Entscheidungshilfen zur Optimierung des Betriebs des Straßennetzes zu bieten. Der erwartete Nutzen besteht darin, Leben, Zeit, Energie und Geld zu sparen und die Umwelt zu schützen.

**Bild 1** veranschaulicht einen architektonischen Rahmen, der sich an der Architektur der USA orientiert. Viele Länder haben ähnliche Rahmenwerke entwickelt, zum Beispiel KAREN in Europa und ACTIF in Frankreich. Diese Architekturen bestehen in der Regel aus vier Hauptteilsystemen: Verkehrsmanagementzentralen (TMC), Fahrzeuge, Straßeninfrastruktur und Benutzer.

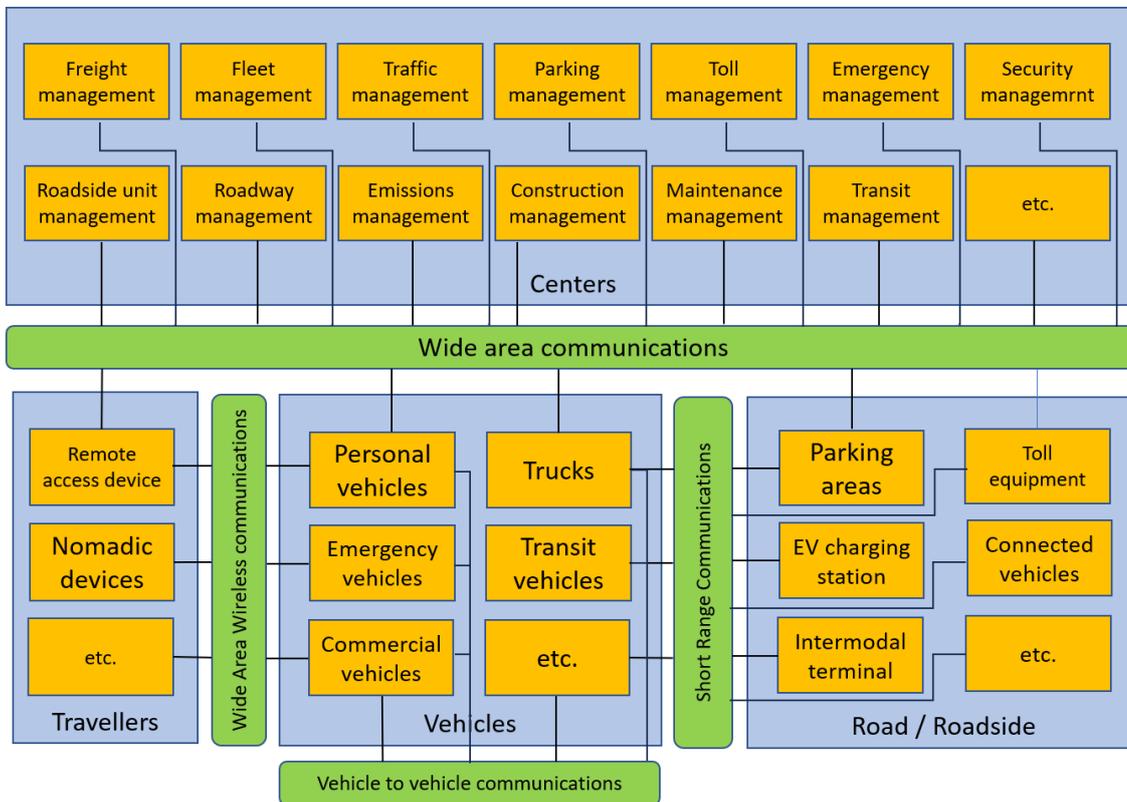


Bild 1. Rahmenarchitektur für intelligente Verkehrssysteme (in Anlehnung an die Arc-IT 9.2-Architektur der USA).

Jedes Teilsystem besteht aus einer breiten Palette von Diensten, die zusammenarbeiten und Daten über verschiedene Kommunikationsmethoden austauschen - sei es über kurze, mittlere oder lange Strecken, drahtlos oder drahtgebunden.

Die übergreifende Architektur ist so konzipiert, dass sie die interoperable Integration von Diensten fördert und die gemeinsame Nutzung von Ressourcen optimiert. Das autonome Fahrzeug ist eine entscheidende Komponente innerhalb dieses Architekturrahmens.

### Fahrerassistenz und autonome Fahrzeuge

Das autonome Fahrzeug ist keine plötzliche Revolution, sondern vielmehr das natürliche Ergebnis einer langen Entwicklung. Sie begann mit Fahrerassistenzsystemen wie ABS und ESP und vor allem mit weiter entwickelten Fahrerassistenzsystemen, von denen viele inzwischen weit verbreitet sind. Diese ADAS dienen als Grundbausteine für vernetzte und autonome Fahrzeuge (Connected Autonomous Vehicles, CAVs), auf denen übergeordnete autonome Funktionen aufbauen. Bevor auf autonome Fahrzeuge eingegangen wird, bietet dieses Buch einen kurzen Überblick über ADAS, deren Entwicklungspfad in **Bild 2** dargestellt ist.

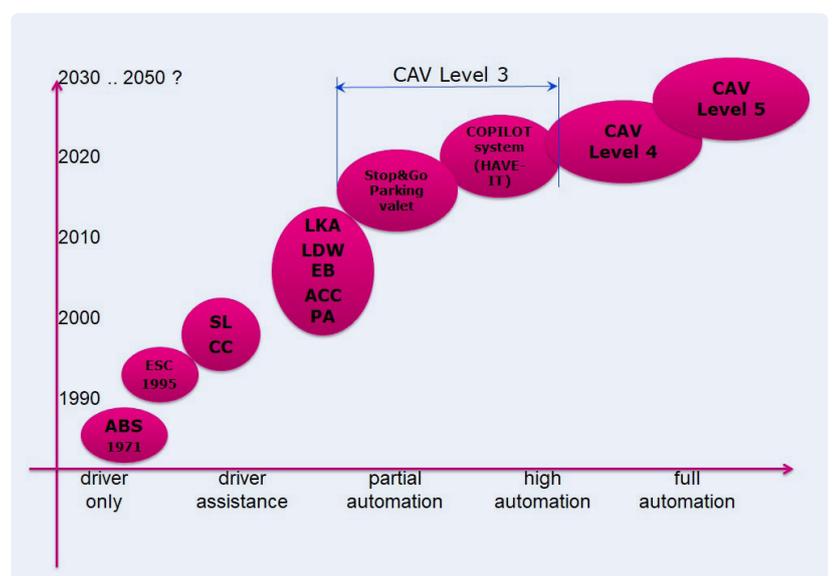
Es wäre jedoch ein Fehler, anzunehmen, dass ein autonomes Fahrzeug lediglich die Summe seiner ADAS-Komponenten ist. Was das autonome Fahrzeug wirklich auszeichnet, ist seine Fähigkeit, eine Route vom Start- zum Zielort zu planen und dabei alle Hindernisse auf dem Weg zu bewältigen, wobei es im Idealfall effizienter arbeitet als ein menschlicher Fahrer. Dies ist eine außerordentlich komplexe

Herausforderung, die noch nicht alle autonomen Fahrzeuge vollständig gemeistert haben.

### Klassifizierung von autonomen Fahrzeugen

Als allgemeine Definition können wir sagen, dass in einem autonomen Fahrzeug die gesamte oder ein Teil der Fahraufgabe an bordseitige Automatisierungssysteme delegiert wird. In bestimmten Situationen kann das Eingreifen des Fahrers weiterhin erforderlich sein, manchmal auch über längere Zeiträume. Diese Definition bleibt jedoch zu vage, was dazu führte, dass 2014 mit der Norm SAE J3016 ein formalisierterer

Bild 2. Fahrerassistenz von den 1990er Jahren bis heute.



Level	Mode Definition	Vehicle Control	Environment Monitoring	Fallback Performance	Contexts
0	<b>No Automation</b>	Human	Human	Human	N/A
1	<b>Driving assistance:</b> actions on steering system, brakes or accelerator in some circumstances	Human and System	Human	Human	Limited
2	<b>Partial automation:</b> actions on steering system, brakes or accelerator in some circumstances	System	Human	Human	Limited
3	<b>Conditional automation:</b> automatic driving, but the driver must be ready to take control at any time	System	System	Human	Limited
4	<b>High automation:</b> as above, even if the driver is not ready to regain control	System	System	System	Limited
5	<b>Full automation:</b> as above, but in all situations	System	System	System	All

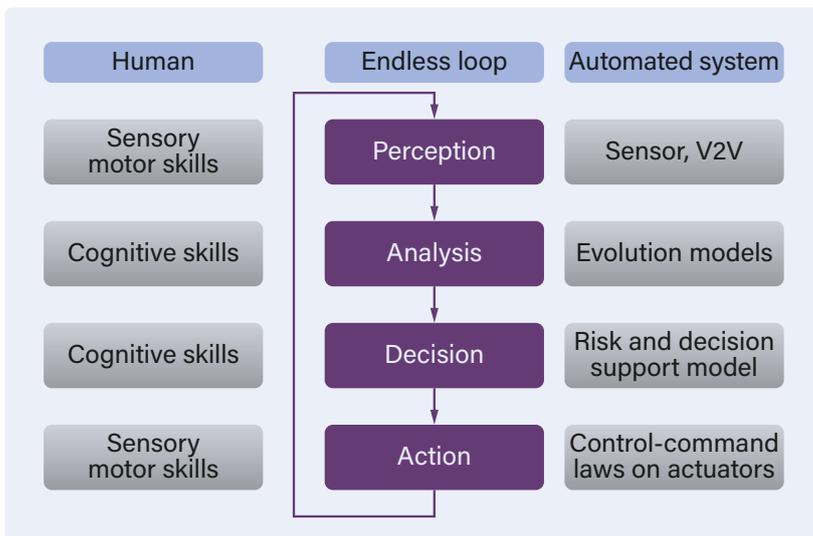
▲  
Bild 3. Die fünf Stufen der Autonomie.

Rahmen vorgeschlagen wurde, der inzwischen breite Akzeptanz gefunden hat.

Die Tabelle in **Bild 3** veranschaulicht die verschiedenen Automatisierungsstufen, wobei die Definitionen und Begriffe aus einem Dokument von SAE International stammen, das kostenlos unter [1] heruntergeladen werden kann.

In dieser Tabelle werden wir fünf Autonomiestufen untersuchen, die jeweils durch die Fähigkeit des Fahrzeugs definiert sind, den menschlichen Fahrer bei Schlüsselaufgaben zu ersetzen: Quer- und Längskontrolle, Umgebungsüberwachung und Fehlermanagement, sowohl im eingeschränkten als auch im uneingeschränkten Zusammenhängen mit der Straße.

▼  
Bild 4. Modellierung der Fahraufgabe.



## Analyse der Fahraufgaben

Um die Grundlagen von CAV zu verstehen, muss man zunächst wissen, wie der Mensch fährt, denn autonome Fahrzeuge sowie die Fahrassistenzsysteme sind vom Menschen inspiriert. Wir beginnen mit einem kleinen Beispiel: ein Fahrzeug auf der Autobahn in einer potenziellen Unfallsituation. Das Buch hebt die Dualität zwischen dem menschlichen Modell und dem Fahrzeugmodell hervor.

Auf der einen Seite nutzt der Fahrer seine kognitiven und sensomotorischen Fähigkeiten, auf der anderen Seite basiert die Autonomie des Fahrzeugs auf Algorithmen, Sensoren und Aktoren. All diese Fähigkeiten beruhen auf einem Vier-Stufen-Modell: Wahrnehmung, Analyse, Entscheidung und Handlung. **Bild 4** zeigt die Ähnlichkeiten zwischen den Handlungen des Fahrers und der Automatisierung des Fahrzeugs. Je nach Automatisierungsgrad können ein Teil oder alle dieser Aktionen von dem einen oder dem anderen unterstützt werden, oder sie können sich beide teilen.

## Die sieben Schlüsselfunktionen des autonomen Fahrzeugs

Das autonome Fahrzeug basiert auf den - wie es in diesem Buch genannt wird - sieben Schlüsselfunktionen des autonomen Fahrzeugs. Dabei handelt es sich nicht um eine offizielle Taxonomie, sondern um einen Vorschlag des Autors.

### 1. Makroskopische Lokalisierung und Kartierung

Die makroskopische Lokalisierung ist von metrischer Präzision. Sie ermöglicht die Positionierung des

Fahrzeugs auf der Straße oder sogar auf einer Fahrspur. Sie stützt sich hauptsächlich auf GNSS-Satellitensysteme (wie GPS, GLONASS, GALILEO und so weiter). Die Satellitenortung kann jedoch in Tunneln oder in Bereichen, in denen die von den Satelliten ausgestrahlten Signale maskiert werden, fehlerhaft sein. Daher ist die makroskopische Lokalisierung das Ergebnis der Fusion von Satellitenlokalisierung und Bewegungsbahnschätzung auf der Grundlage von Daten, die von Trägheits- und Entfernungssensoren geliefert werden (**Bild 5**). Die erzielte Genauigkeit reicht aber nicht aus, um das Fahrzeug auf einer Karte zu lokalisieren. Wir werden die Technik der Kartenanpassung (map-matching) untersuchen, die darin besteht, die geschätzte Position auf eine kartografische Referenz zu projizieren, die aus Segmenten besteht, die Teile der Straße darstellen (**Bild 6**).

## 2. Mikroskopische Lokalisierung

Die mikroskopische Lokalisierung ist sowohl bei ADAS als auch bei autonomen Fahrzeugen für die Kontrolle der seitlichen Flugbahn unerlässlich. Ähnlich wie Straßenbahnen und Züge von Schienen geführt werden, verlassen sich autonome Fahrzeuge auf eigene „Schienen“: die Straßenmarkierungen. Diese Markierungen werden von einer nach vorne gerichteten Kamera erfasst, die es dem Fahrzeug in Verbindung mit Bildverarbeitungstechniken und Straßenkrümmungsmodellierung ermöglicht, sich zentimetergenau auf der Fahrbahn zu positionieren (siehe **Bild 7**).

## 3. Erkennung von Hindernissen

ADAS wie Kollisionsvermeidung oder -minderung, Notbremsung oder Abstandsregelung beruhen auf der Hinderniserkennung. Die Hinderniserkennung stützt sich hauptsächlich auf drei Arten von Sensoren: Radar, Laserscanner (Lidar) und Stereosehen mit zwei Kameras sowie die Datenfusion zwischen diesen drei Quellen. Stereo-Vision und Datenfusionstechniken werden ebenso untersucht wie verschiedene Arten von Radar (Lang-, Mittel- und Kurzstreckenradar).

## 4. Fahrzeugdynamik

Die Kenntnis des dynamischen Zustands des Fahrzeugs gewährleistet, dass es innerhalb des Bereichs seiner Beherrschbarkeit bleibt. Im Falle von ADAS ermöglicht dies die Einleitung von Korrekturmaßnahmen an der Bewegungsbahn, wenn sich das Fahrzeug den Grenzen dieses Bereichs nähert, im Falle von CAV geht es um die Definition von Steuerungs- und Regelungsgesetzen, die innerhalb des Bereichs der Beherrschbarkeit bleiben. Die Untersuchung der Dynamik stützt sich auf Modelle, von denen das einfachste das von den Forschern häufig verwendete „Fahrrad-Modell“ ist (**Bild 8**).

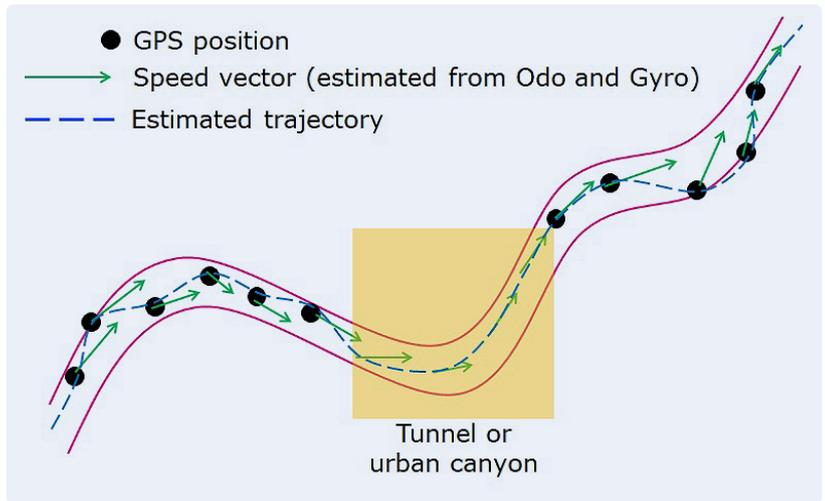


Bild 5. Positionsbestimmung mit Hilfe von GNSS- und Koppelnavigationstechniken.

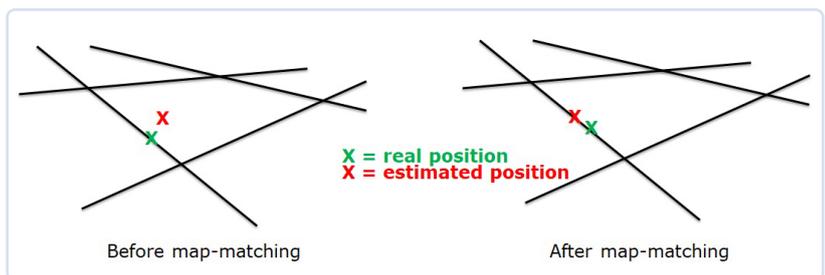


Bild 6. Kartenabgleich.

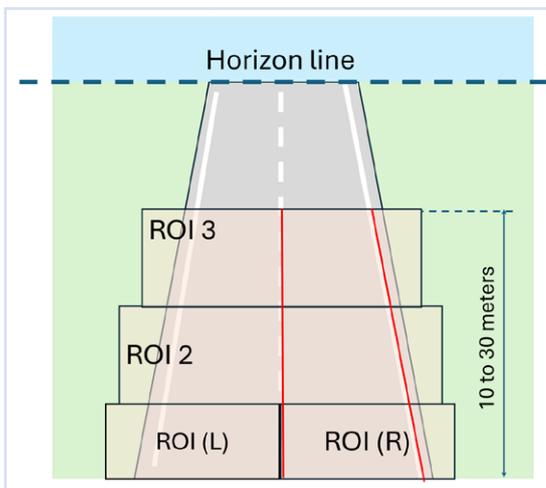
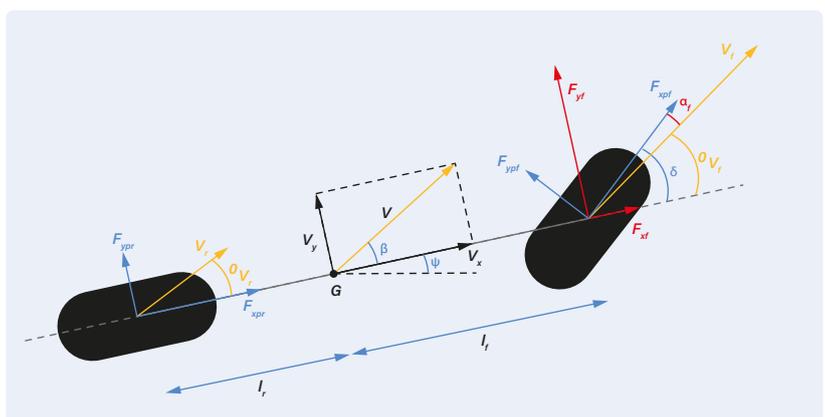


Bild 7. Erkennung von Fahrbahnmarkierungen in den Regionen von Interesse (ROI).

Bild 8. Das Fahrradmodell.



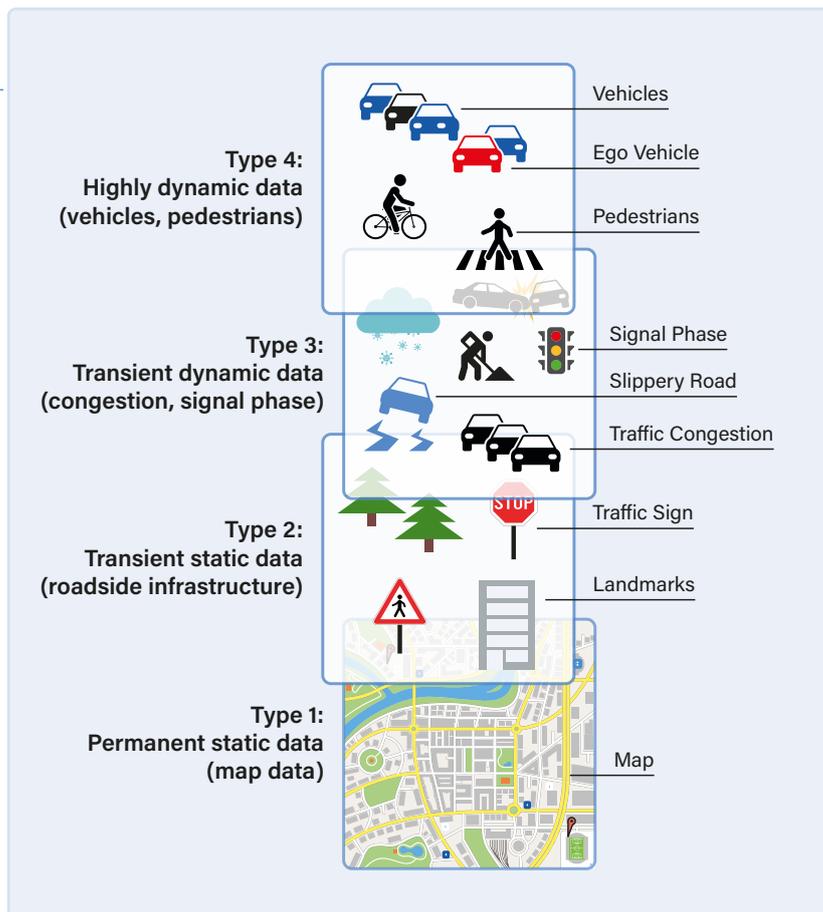


Bild 9. Darstellung der lokalen dynamischen Karte (LDM).

## 5. Situationsbewusstsein

Um sicher zu operieren, muss das autonome Fahrzeug über eine umfassende Kenntnis seiner Umgebung verfügen. Dies geschieht mit Hilfe einer lokalen dynamischen Karte (LDM), ein digitales Karteninformationssystem, in dem alle Straßenattribute und Ereignisse georeferenziert sind, unabhängig davon, ob sie sich langsam oder schnell entwickeln (**Bild 9**). Dieses LDM ermöglicht den Aufbau eines Situationsbewusstseins für das Fahrzeug. Die Konnektivität spielt eine wichtige Rolle, da die Informationen aus verschiedenen Quellen stammen, die sich in der Nähe oder in der Ferne des Fahrzeugs befinden.

## 6. Pfadplanung

Es gibt drei Ebenen der Pfadplanung, die üblicherweise in Betracht gezogen werden: die strategische Ebene, die die gesamte Fahrt vom Ausgangspunkt bis zum Zielort umfasst, die taktische Ebene, deren Zeitspanne zwischen zehn Sekunden und einigen Minuten liegt, und die operative Ebene, die die unmittelbare Pfadsteuerung durch Einwirkung auf die Fahrzeugaktuatoren betrifft.

## 7. Überwachung des Fahrers

Einer der kritischen Punkte des CAV in den Stufen 2...4 ist seine Interaktion mit dem Fahrer. Im Falle eines Systemausfalls oder unerwarteter Ereignisse kann der Fahrer zur manuellen Bedienung des Fahrzeugs herangezogen werden. Daher ist es erforderlich, dass der Fahrer weder schläft noch abgelenkt ist. Die Fahrer-

überwachungsfunktion liefert Echtzeitinformationen über den Status des Fahrers, um zum Beispiel die Übergänge vom automatisierten zum manuellen Fahren sicher zu gestalten. Wir befassen uns auch mit der Frage der Minimalen-Risiko-Manöver (MRM) für den Fall, dass der Fahrer nicht reagiert.

## Konnektivität

Damit das autonome Fahrzeug Schwierigkeiten im Straßenverkehr vorhersehen kann, muss die Wahrnehmung der Umgebung über die Reichweite der eigenen Wahrnehmungssensoren hinaus erweitert werden. Diese erweiterte Wahrnehmung wird durch bidirektionale V2V-, V2I- und I2V-Kommunikation erreicht. Die auf europäischer Ebene entwickelte Kommunikationsarchitektur wird in diesem Buch ebenfalls betrachtet. Die V2X-Kommunikation basiert auf dem Konzept der C-ITS-Stationen (**Bild 10**), die die Knotenpunkte eines von Fahrzeugen, straßenseitigen Einheiten, Verkehrsmanagementzentren und Nutzern (wie Fußgängern oder Radfahrern) getragenen Kommunikationsnetzes sind. Jede C-ITS-Plattform kann über mehrere Kommunikationsvektoren kommunizieren: 3G, 4G, 5G, 802.11p, Ethernet, und so weiter.

## Umsetzung

Die oben erwähnten Schlüsselfunktionen werden durch die Bordelektronik und die Datenverarbeitung unterstützt. In diesem Buch werden wir uns mit der Elektronik- und Computerarchitektur des Fahrzeugs befassen. Die Komponenten dieser Architektur sind Aktoren, Sensoren, Computer (auch elektronische Steuergeräte genannt) und Kommunikationsbusse. Ein Hauptaugenmerk liegt auf den Computern und ihrem Ausführungsrahmen sowie auf den verschiedenen Kommunikationsbussen wie CAN, FlexRay und LIN. Außerdem werden verschiedene Architekturtopologien untersucht, von den einfachsten zentralisierten bis hin zu komplexeren, verteilten Architekturen (**Bild 11**).

Ein wichtiges Kapitel des Buches ist der Methodik des Architekturentwurfs gewidmet. Sie basiert auf einem Top-Down-Ansatz, der von den Bedürfnissen der Benutzer ausgeht und über verschiedene Zwischenstufen zu einer physischen Architektur führt. Die Methode wird anhand des Beispiels eines einfachen Bausteins für autonome Fahrzeuge veranschaulicht: dem Geschwindigkeitsbegrenzer.

## Vorläufige Gefährdungsanalyse

Automobilsysteme sind sicherheitskritisch. Selbst ein kleiner Hardware- oder Softwarefehler kann zu Unfällen mit tödlichen Folgen führen. Wir werden die Norm für funktionale Sicherheit ISO26262 betrachten, an die sich fast alle Automobilhersteller halten. Der erste Schritt ist die Vorläufige Gefährdungsanalyse

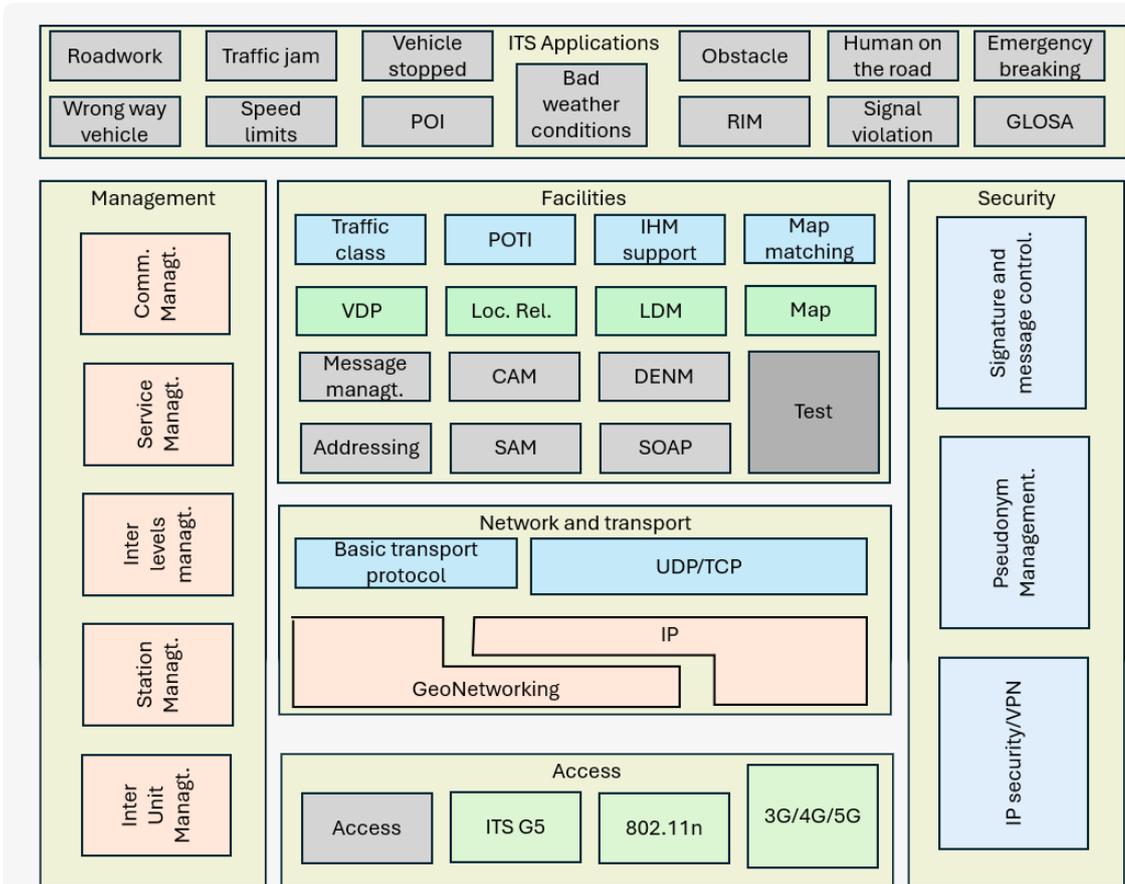


Bild 10.  
Typische Architektur  
einer ITS-Station.

(Preliminary Hazard Analysis, PHA), die aus einer Analyse des Systemverhaltens und der Interaktion mit der Umgebung unter allen möglichen Umständen besteht.

Der zweite Schritt besteht darin, das Risikoniveau auf der Grundlage von drei Merkmalen abzuschätzen: Schweregrad, Risikoexposition und Kontrollierbarkeit, und dann einen ASIL-Index (Automotive Software Integrity Level) zu ermitteln. Schließlich werden wir sehen, wie „Sicherheitsanforderungen“ zur Vermeidung oder Minimierung von Risiken auf ein akzeptables Niveau führen.

### Die Infrastruktur

Jahrzehntlang waren die Welten der Automobilindustrie und der Straßennetzbetreiber voneinander relativ abgeschottet, doch mit der Entwicklung der

Telekommunikation und dem Aufkommen autonomer Fahrzeuge haben die Automobilhersteller und die Straßennetzbetreiber die Notwendigkeit einer engen Zusammenarbeit erkannt. Heutzutage gibt es eine Win-Win-Kooperation, die einerseits die Effizienz und Sicherheit der Fahrzeuge und andererseits den Betrieb des Straßennetzes verbessert. Diese Zusammenarbeit bezieht sich auf das Konzept der „Floating Car Data“ (FCD), auch Sondenfahrzeuge genannt. Wir werden auch das neue Konzept der Operational Design Domain (ODD) und der Infrastrukturunterstützung für autonomes Fahren (ISAD) vorstellen.

### RS-CAV: Das vernetzte und autonome Fahrzeug im kleinen Maßstab

Haben Sie jemals davon geträumt, Ihr eigenes CAV zu bauen? Ein vernetztes und autonomes Fahrzeug in

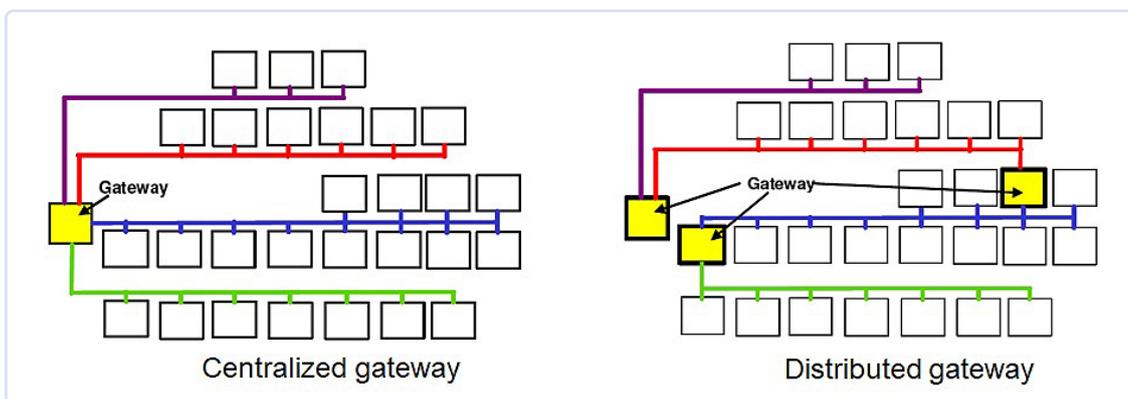


Bild 11. Zentralisierte  
und verteilte  
Architekturtopologie.

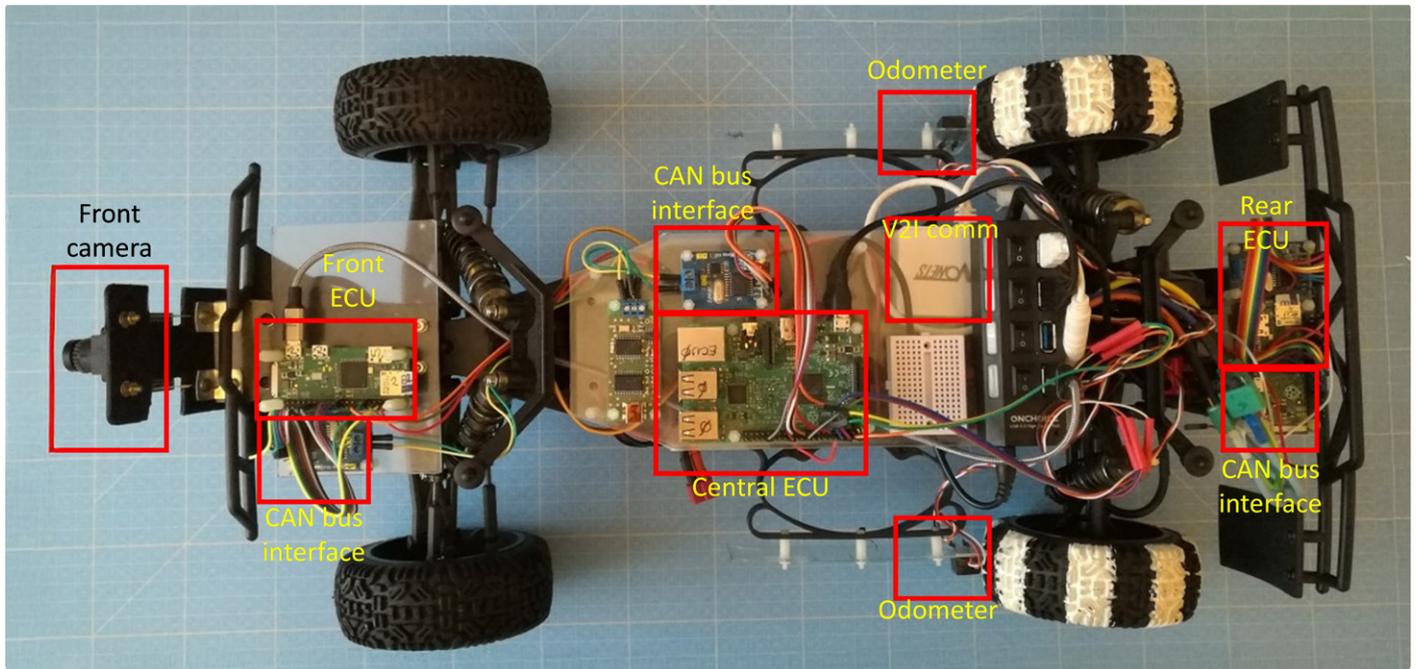


Bild 12.  
RS-CAV, entwickelt vom  
Autor. (Gesponsert von  
der Universität Gustave  
Eiffel, Bild von J. Ehrlich).

verkleinertem Maßstab (RS-CAV) könnte die Antwort auf Ihren Wunsch sein. Im letzten Teil dieses Buches finden Sie Vorschläge zum Bau eines Fahrzeugs im Maßstab 1:10, wie es der Autor dieses Buches geschaffen hat (**Bild 12**).

Eine schlüsselfertige Lösung wird es jedoch nicht geben. Der Bau eines autonomen Fahrzeugs erfordert viele Fähigkeiten in den Bereichen Signal- und Bildverarbeitung, Steuerung und Kontrolle, Telekommunikation und so weiter. Dieses Projekt könnte der Beginn einer Herausforderung für Elektronik- und Computerenthusiasten sein, die als Team zusammenarbeiten könnten, um dieses Ziel zu erreichen: ein voll funktionsfähiges, kleines autonomes Fahrzeug. Es liegt an Ihnen, mitzuspielen... [▶](#)

Übersetzung: Rolf Gerstendorf — 240537-02

### Sie haben Fragen oder Kommentare?

Haben Sie technische Fragen oder Kommentare zu diesem Artikel? Schreiben Sie bitte an Jacques Ehrlich unter [je-cav-book@orange.fr](mailto:je-cav-book@orange.fr) oder an die Elektor-Redaktion unter [redaktion@elektor.de](mailto:redaktion@elektor.de).

### Über den Autor

Jacques Ehrlich hat einen Dokortitel in Elektronik und Telekommunikation von Télécom ParisTech. Am IFSTTAR, der späteren Gustave-Eiffel-Universität, war er bis zu seiner Pensionierung im Jahr 2014 Co-Direktor und später Leiter eines Forschungslabors, das auf hoch entwickelte Fahrassistenzsysteme und autonome Fahrzeuge spezialisiert war. Von 2014 bis 2024 war er emeritierter Forschungsdirektor. Von 2012 bis 2019 war er außerdem Vorsitzender des internationalen technischen Ausschusses „Road Network Operation and ITS“ beim Weltstraßenverband (PIARC). Derzeit koordiniert er einen Kurs über autonome Fahrzeuge im Rahmen des spezialisierten Masterstudiengangs Smart Mobility (MS-SMOB) am Institut Polytechnique de Paris. Als leidenschaftlicher Liebhaber von Elektronik, Computern und Segelflugzeugen ist Jacques auch ein zertifizierter Fluglehrer. Er hat mehrere in Elektor Labs veröffentlichte Projekte verfasst, darunter GONOGO, CHADECHE, COBALT und OUAH!



### Passende Produkte

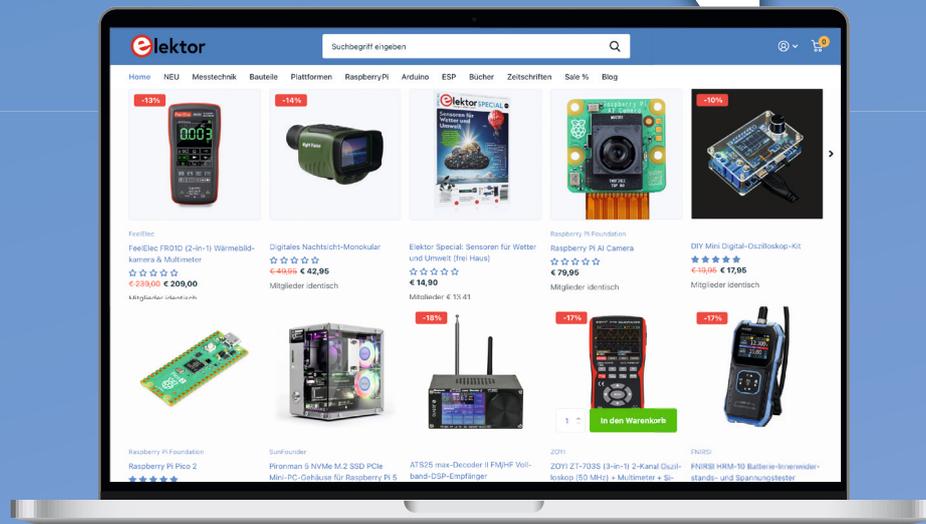
- ▶ **Elektor Ultimate Sensor Kit**  
[www.elektor.de/19104](http://www.elektor.de/19104)
- ▶ **Raspberry Pi 5 (4 GB RAM)**  
[www.elektor.de/20598](http://www.elektor.de/20598)



### WEBLINKS

- [1] SAE International, „Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles“, überarbeitet 2021-04: [https://sae.org/standards/content/j3016\\_202104/preview/](https://sae.org/standards/content/j3016_202104/preview/)
- [2] Yue Wang, Eam Khwang Teoh, Dinggang Shen, „Lane detection and tracking using B-Snake“, Elsevier: <https://sciencedirect.com/science/article/abs/pii/S0262885603002105>
- [3] Intelligent Transport Systems (ITS) Communication Architecture, ETSI EN 302 665, ETSI (PDF): <https://tinyurl.com/itsyspdf>

# Was ist Ihre Meinung?



Bei Elektor bieten wir mehr als nur Elektronik - wir schaffen ein Erlebnis, von Qualitätsprodukten und außergewöhnlicher Kundenbetreuung, unterstützt durch die Leidenschaft unserer Community.

Teilen Sie Ihre Meinung auf  
[www.elektor.de/pages/customer-reviews](http://www.elektor.de/pages/customer-reviews)



**Günther A.**

Das Meßgerät wurde kurzfristig geliefert. Solide Ausführung, gut ablesbares Display, zuverlässige Kontaktierung der 4-Leiter Kelvin Meßleitungen, hohe Genauigkeit der Meßwerte (Spannung, Innenwiderstand)



**Matthias D.**

Ich hatte die Elektor schon in der 80ern und dann etwas aus den Augen verloren. Es ist wie damals: prima Schaltungsideen, anwendbar, sehr gut erklärt, alles bestens. Danke und bitte weiter so.



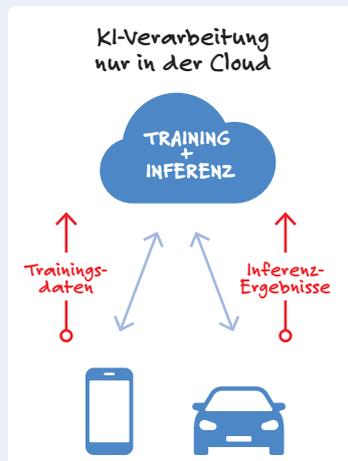
**Hartmut W.**

Sehr guter Service!

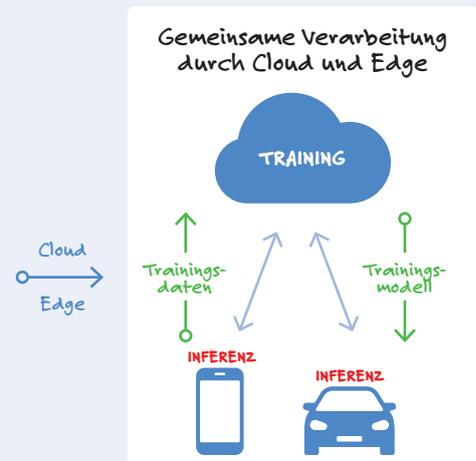
Schnelle Lieferung, gute Bezahlmöglichkeit, Gerät top. Sehr zu Empfehlen! Werde weiter Bestellen.

## Cloud-KI und Edge-KI

Cloud-KI ist ideal für komplexe, ressourcenintensive Aufgaben, hat aber aufgrund der Datenübertragung mit Latenz- und Sicherheitsproblemen zu kämpfen [1]. Edge-KI wird für Echtzeit-Anwendungen bevorzugt, trotz ihrer begrenzten Rechenleistung [1]. Hybride KI-Modelle kombinieren die Stärken von Cloud und Edge, indem sie intensive Aufgaben in der Cloud erledigen und gleichzeitig Echtzeit-Inferenzen an der Edge ermöglichen [2]. Dieser hybride Ansatz erhöht die Effizienz, verringert den Bandbreitenbedarf und verbessert den Datenschutz, da sensible Daten auf dem Gerät verbleiben [2].



- > Erhöhung des Datenübertragungsvolumens
- > Große Auswirkung auf die Datenlatenz
- > Große Auswirkungen auf Verbindungsabbrüche



- > Reduzierung der Datenübertragungsmenge
- > Geringere Auswirkung auf die Datenlatenz
- > Geringere Auswirkung von Übertragungsabbrüchen

Quelle: Lionbridge AI [3]

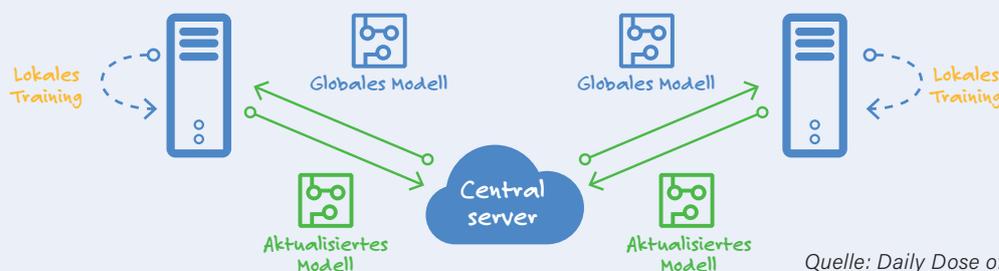
## Edge-KI und Verteilte KI

Verteilte KI (distributed AI) stellt die nächste Stufe der KI-Entwicklung dar und bietet eine Lösung für die Skalierbarkeitsprobleme, die häufig bei der Bereitstellung von Edge-KI an mehreren Standorten auftreten. Durch die Einführung eines Hub-and-Spoke-Modells (Speicherarchitektur) verwaltet ein zentraler Hub die KI-Funktionen, und die Spoke-Standorte senden die zu verarbeitenden Daten. Der Hub

fungiert als Steuerungsebene, stellt Anwendungen bereit und verwaltet den Lebenszyklus der Daten. Dieses Modell verringert die Belastung durch große Datenübertragungen, indem durch die intelligente Datenerfassung nur die notwendigen Daten verarbeitet werden [1]. Herausforderungen wie „Data Gravity“ (die Kosten und die Netzwerkbandbreite für die Übertragung großer

Datenmengen) und „Heterogenität“ (die Notwendigkeit, KI-Modelle für verschiedene Anwendungsfälle an unterschiedlichen Standorten anzupassen) finden sich in der Verteilten KI [4]. Lösungen wie die Automatisierung des Daten- und KI-Lebenszyklusmanagements und die Anpassung von KI-Pipelines für verschiedene Anwendungen sind wichtig, um diese Hindernisse zu überwinden.

## Föderales Lernen



Quelle: Daily Dose of Data Science [6]

Föderales Lernen ist eine Technik des maschinellen Lernens (ML), bei der ein Algorithmus über mehrere dezentrale Edge-Geräte oder Server trainiert wird, die lokale Datenproben tragen, ohne sie zu teilen [5]. Diese Methode unterscheidet sich von herkömmlichen zentralisierten Methoden des maschinellen

Lernens, bei denen alle lokalen Datensätze an einen einzigen Server übermittelt werden müssen. Es wird unter anderem in den Bereichen Datenschutz, Telekommunikation, IoT und Pharmazie eingesetzt.

# \$2,6 bis \$4,4 Billionen

Potenzielle Wertschöpfung durch Generative KI [2]. Die geschätzten globalen Gewinnauswirkungen in Branchen wie Forschung und Entwicklung, Produktentwicklung und Simulation in Biowissenschaften und Fertigung.

## Generative KI: Wirtschaftliches Potenzial

### 0,1% bis 0,6%

Jährliches Produktivitätswachstum [7]. Generative KI könnte bis 2040 ein Wachstum der Arbeitsproduktivität ermöglichen, mit dem Potenzial für ein Produktivitätswachstum von 0,5...3,4%, wenn sie mit allen anderen Technologien kombiniert wird.

## 2030 bis 2060

Der geschätzte Zeitraum, in dem die Hälfte der heutigen Arbeitstätigkeiten automatisiert werden könnte, mit einem Mittelwert im Jahr 2045 [8].

## Generative KI im Jahr 2024: Akzeptanz und Auswirkungen

Mit zunehmender Verbreitung generativer KI berichten die Befragten [8] von klaren Vorteilen und einem geringeren Risiko von Ungenauigkeiten.

### 65%

Unternehmen, die regelmäßig generative KI einsetzen - fast doppelt so viele wie bei einer 10 Monate zuvor durchgeführten Umfrage, was auf eine schnelle Integration in die Geschäftsabläufe hindeutet.

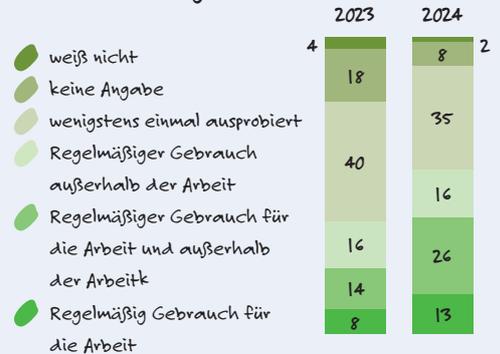
### 75%

Sagen voraus, dass generative KI disruptiv sein wird. Die Befragten glauben, dass generative KI in den kommenden Jahren zu großen Veränderungen in ihrer Branche führen wird.

### 72%

Allgemeine Rate der Einführung von KI. Ein signifikanter Anstieg von 50 % in den Vorjahren, mit weit verbreitetem Interesse in allen Branchen und Regionen.

Deutlicher Anstieg der GenAI-Nutzung im Vergleich zu 2023



Allgemeiner Durchschnitt

(Quelle: McKinsey Global Survey on AI [8])

Übersetzung: Rolf Gerstendorf – 240692-02

### WEBLINKS

- [1] IBM, „What Is Edge AI?“, <https://www.ibm.com/think/topics/edge-ai>
- [2] Edge AI Foundation, „Edge AI Technology Report“, Wevolver 2024: <https://www.wevolver.com/article/the-guide-to-generative-ai-at-the-edge>
- [3] Lionbridge AI, „What Is Edge AI Computing?“, Medium, DataDrivenInvestor: <https://medium.com/datadriveninvestor/what-is-edge-ai-computing-61ece58c76d0>
- [4] M. Keerthi, „Edge AI vs Distributed AI“, Medium 2024: <https://mukul04-sk.medium.com/edge-ai-vs-distributed-ai-154060456f1b>
- [5] Dr. J. Kaur Gill, „Edge AI vs Federated Learning“, Xenonstack, Dezember 2024: <https://tinyurl.com/2brehfn8>
- [6] A. Chawla, „Federated Learning: A Critical Step Towards Privacy-Preserving Machine Learning“, Daily Dose of Data Science 2023: <https://tinyurl.com/dailydoseofds>
- [7] M. Chui et al., „The Economic Potential of Generative AI“, McKinsey Insights 2023: <https://tinyurl.com/ecogenai>
- [8] A. Singla et al., „The State of AI in Early 2024“, QuantumBlack, AI von McKinsey, McKinsey Insights 2024: <https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai#/>

# Tastatur/Display-Schnittstellen-IC 8279 von Intel

Von David Ashton (Australien)

Heutzutage können Sie Ihren Mikrocontroller ganz einfach mit der Welt der Menschen verbinden. Schließen Sie einen HDMI-Monitor und eine USB-Tastatur an und schon kann es losgehen. In der (guten?) alten Zeit der Mikroprozessoren war das nicht so einfach und die Verbindung war eine wahre Kunst.



Bild 1. Ein Second-Source-IC 8279 von Mitsubishi. (Bild von JWBE - Eigenes Werk, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=20019181>)

Die großen Mikroprozessorersteller boten viele Peripheriechips an, um das Leben der Embedded-Entwickler leichter zu machen. Eines davon war die Tastatur/Display-Schnittstelle 8279 von Intel. Sie ermöglichte es einem 8080- oder 8085-Mikroprozessorsystem von Intel, über eine 64-Tasten-Tastatur und zwei 16-stellige numerische 7-Segment-Displays oder einem 16-stelligen alphanumerischen Display mit dem User zu kommunizieren. **Bild 1** zeigt ein „Second-Source-IC“ 8279 von Mitsubishi.

## Multiplexen

In **Bild 2** ist ein einfaches Blockschaltbild des 8279 zu sehen. Entscheidend für den Betrieb waren die vier Scan-Ausgangsleitungen. Sie konnten acht Zeilen einer 8x8-Tastaturmatrix oder 16 Ziffern einer (alpha)numerischen Anzeige ansteuern. Dazu benötigte man zusätzlich einen 3x8-Zeilendecoder (für die Tastatur) oder einen 4x16-Zeilendecoder (für die Anzeige). Bei der Tastaturmatrix zeigten acht Rückleitungen an, welche der bis zu 64 Tasten gedrückt wurde. Es gab zwei zusätzliche Eingänge für die Tasten *Strg* und *Shift*. Der 8279 gab dann einen entsprechenden Tastencode an den Mikrocontroller zurück.

Es gab viele Verfeinerungen, die den Chip sehr vielseitig machten. Normalerweise war die Tastatur eine 8x8-Matrix und der 8279 gab den Code zurück, der der gedrückten Taste entsprach. Aber man konnte auch den so genannten *Sensor-Matrix*-Modus verwenden, bei dem die Tastatur-Abtastzeilen einen Sensor auswählten, der dann 8 Bit Daten an den 8279 zurückgab (zum Beispiel DIP-Schalter-Einstellungen oder ADC-Ausgangsdaten). Tastatur- oder Sensordaten wurden in einem FIFO-Puffer zum späteren Abruf gespeichert.

Die Anzeige konnte auf Linkseingabemodus (Daten werden vom äußersten linken Digit eingegeben und wandern nach rechts) oder auf Rechtseingabemodus eingestellt werden, was bedeutet, dass der Multiplexer die einzelnen Ziffern von L nach R oder in Richtung R nach L anspricht. Die acht Display-Datenleitungen konnten direkt verwendet werden (zum Beispiel für 7-Segmente plus Dezimalpunkt) oder für alphanumerische Anzeigen dekodiert werden. Es bestand auch die Möglichkeit, die Daten in 2x4 Zeilenbusse aufzuteilen und so eine 2x16-stellige Anzeige zu steuern.

## Parallele Schnittstelle

Wie bei den meisten Intel-Peripheriechips wurden viele Funktionen über Steuerregister gesteuert und festgelegt. Neben den acht Datenleitungen zum Mikrocontroller gab es noch sechs oder sieben Steuerleitungen. Wenn man heute 7-Segment-Displays ansteuern will, kann man ICs wie den ICM7218 oder MAX7219 kaufen, die bis zu 16 Displays ansteuern können, aber sie sind nicht annähernd so vielseitig wie ein 8279. ◀

Übersetzung: S. Gerstendorf -- 240691-02

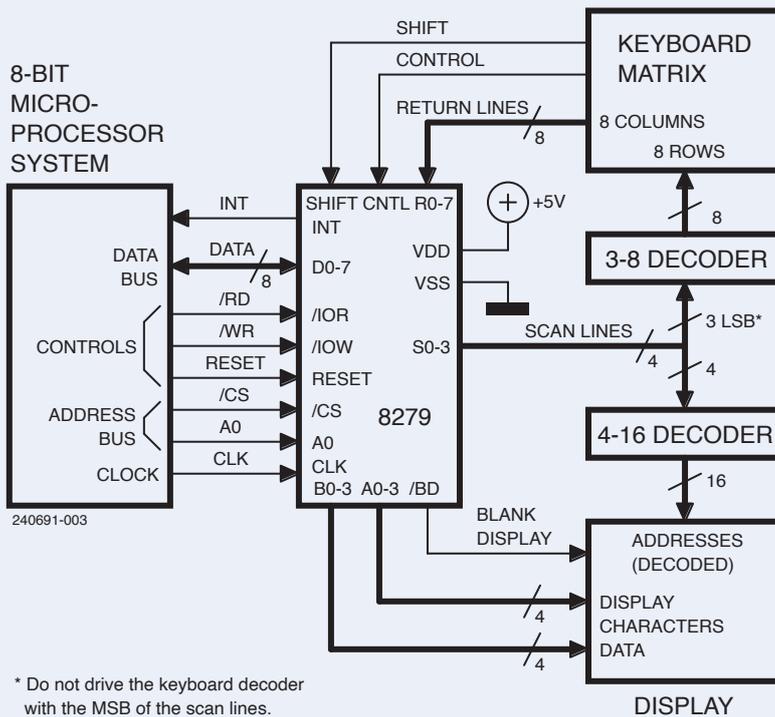


Bild 2. Grundlegendes 8279-Blockdiagramm. (Quelle: Intel-8279-Datenblatt)

# Jede Bewertung spiegelt ein persönliches Erlebnis wider

**Top Service**  
Schnelle Lieferung, guter Preis. Leider war das teure Buch schlecht verpackt (loste im zu großen Karton), so dass es ziemlich angestoßen bei mir ankam. Das Buch wurde anstandslos ersetzt, so geht guter Service!  
Date of experience: June 11, 2024

**Elektor lieferte schnell**  
Elektor lieferte schnell, preiswert und zuverlässig wie immer!  
Date of experience: June 03, 2024



**Elektor, das Elektronik-DIY El Dorado**  
Elektor ist ein El Dorado für Elektronik-DIY-Fans. Es gibt unzählige Projekte und deren Shop bietet kostengünstige Produkte.  
Date of experience: May 31, 2024

**Perfekt**  
Perfekt. Schnelle Lieferung, sehr gut verpackt. Artikel im perfekten Zustand. Danke!  
Date of experience: February 01, 2024



Wir lieben Elektronik und Projekte, und wir setzen alles daran, die Bedürfnisse unserer Kunden zu erfüllen  
Der Elektor-Store: 'Never expensive, always surprising'

**Elektor Store**  
Reviews 365 • Excellent  
★★★★★ 4.3  
VERIFIED COMPANY

Sehen Sie sich weitere Bewertungen auf unserer Trustpilot-Seite an: [www.elektor.com/TP/de](http://www.elektor.com/TP/de)  
Oder bilden Sie sich selbst eine Meinung und besuchen Sie unseren Elektor Store, [www.elektor.de](http://www.elektor.de)



# Makerfabs SenseLoRa

## Plug-and-Play-IoT für Gewächshäuser

Von Clemens Valens (Elektor)

Wenn Sie schon immer ein drahtloses Überwachungssystem in Ihrem Gemüsegarten einrichten wollten, sich aber von der Komplexität einer ferngesteuerten, solarbetriebenen und wasserfesten Anlage abgeschreckt fühlten, könnte das SenseLoRa-System von Makerfabs die ideale Lösung für Sie sein. Es vereint all diese Funktionen und Anforderungen und ist gleichzeitig vollständig Plug-and-Play – das heißt, es erfordert keinen zusätzlichen Konfigurationsaufwand. In diesem Artikel nehmen wir den *Industrial-grade Air Monitor* und den dazugehörigen *LoRa Receiver* genauer unter die Lupe. Dabei prüfen wir unter anderem die Benutzerfreundlichkeit, die Zuverlässigkeit und die tatsächliche Leistung des Systems, um zu sehen, ob es wirklich hält, was es verspricht.

Es war ein warmer, sonniger Tag, als ich mich entschloss, das SenseLoRa-System von Makerfabs auf die Probe zu stellen. Würde das Plug-and-Play-System wirklich so einfach funktionieren, wie es verspricht? Mein Testaufbau bestand aus dem *LoRa-Receiver* und dem *Industrial-grade Air Monitor*. Es gibt auch noch ein drittes Gerät, den *Industrial-grade Soil Remote Monitor*, aber den hatte ich nicht zur Verfügung.

Ein kurzer Blick in die Online-Dokumentation zeigte, dass ich es mit einem Plug-and-Play-Umgebungsüberwachungssystem zu tun hatte, das keine Konfiguration benötigte. Also schloss ich den Empfänger über einen USB-Hub an meinen Laptop an. Die Antenne befestigte ich draußen mit einem Magneten am Metalldach meines Schuppens. Dann öffnete ich das Gehäuse des Air Monitors, um ihn einzuschalten (der Schalter befindet sich im Inneren), schloss es wieder und stellte das Gerät in den Vorgarten, etwa 25 m vom Empfänger entfernt und versteckt hinter dem Haus, sodass keine Sichtlinie bestand. Als ich zum Empfänger zurückkehrte, zeigte dieser bereits die vom Air Monitor gesendeten Daten im JSON-Format an (**Bild 1**):

```
Num:1 | -80dbm
{"ID":"AirM01",
"COUNT":2,
"SLEEP":3600,
"bat":3.90,
"Temp":28.38,
"hum":62.62,
"eco2":400.00,
"lux":189.17}
```

In der Tat, das ist Plug and Play! Beachten Sie die geringe Lichtstärke (*lux*). Dieses Datum wurde wahrscheinlich erfasst, während ich das Gerät vorbeistellte oder es in der Hand hielt.



Bild 1. Der Empfänger verfügt über ein kleines Display, das die empfangenen Daten im JSON-Format anzeigt.

## Luftmonitor

Wie Sie aus den empfangenen Daten (Temperatur, Luftfeuchtigkeit, eCO<sub>2</sub> und Lichtintensität) entnehmen können, ist der Air Monitor für den Einsatz in Gewächshäusern konzipiert. Aber auch in Klassenzimmern und anderen Räumen, in denen Menschen zusammenkommen, könnte er nützlich sein, da er Ihnen sagen kann, wann Sie ein Fenster öffnen oder das Licht ein- oder ausschalten sollten. Falls Ihnen das Konzept, aber nicht die Sensoren gefallen, können Sie diese durch andere Sensoren mit I<sup>2</sup>C-Anschluss ersetzen. Natürlich müssen Sie dann die Open-Source-Software anpassen.

## ESP32-S3 mit RF92

Der Air Monitor besteht aus einem ESP32-S3, der mit einem RF92-LoRa-Modul von HopeRF verbunden ist, und einem I<sup>2</sup>C-Bus, an den drei Sensoren angeschlossen sind:

- SGP30 für Luftqualität (äquivalentes CO<sub>2</sub>, also eCO<sub>2</sub>, 0...1000 ppm)
- BH1750 für Umgebungslicht (1...65535 lx)
- AHT10 für Temperatur (-40°C bis 80°C, ±0,3 C) und Feuchtigkeit (0...100%)

Das Gerät kann über WLAN konfiguriert werden, wenn Sie es in den AP-Modus versetzen. So können Sie seine ID (praktisch, wenn das System aus mehreren Air Monitoren besteht) und den Sendezeitraum ändern.

Der Air Monitor verfügt über einen 1000-mAh-Akku, der durch das Solarpanel des Kits aufgeladen wird und den kontinuierlichen Betrieb während der Nacht gewährleistet.

## Industrielle Qualität?

Der Air Monitor ist als "Industrial Grade" gekennzeichnet (**Bild 2**), was sich wahrscheinlich auf sein IP68-zertifiziertes Gehäuse bezieht. Zum Lieferumfang gehören ein 6-V-, 6-W-Solarpanel mit Halterung, Schrauben und Muttern sowie eine Antenne (1 m Kabel). Die Antenne hat nur einen Magneten im Fuß, mit dem sie an einem Metallobjekt befestigt werden kann. Sie sieht aus wie für den Innenbereich geschaffen, obwohl sie laut Benutzerhandbuch für den Außeneinsatz geeignet sein soll. Die Halterung aus lackiertem (emailliertem?) Eisen wird im Freien wahrscheinlich nicht lange halten. Aber der Air Monitor ist ja auch für den Einsatz in Gewächshäusern gedacht (wo Eisen bekanntlich nie und nimmer rostet).

## Der LoRa-Empfänger

Der LoRa-Receiver ist ein kleines Modul, das aus einer roten Platine besteht, die zwischen zwei transparenten Acrylplatten montiert ist. Er basiert auf



◀ Bild 2. Der Industrial-grade Air Monitor ist im Freien auf einer Holzkonstruktion in fast zwei Metern Höhe montiert.

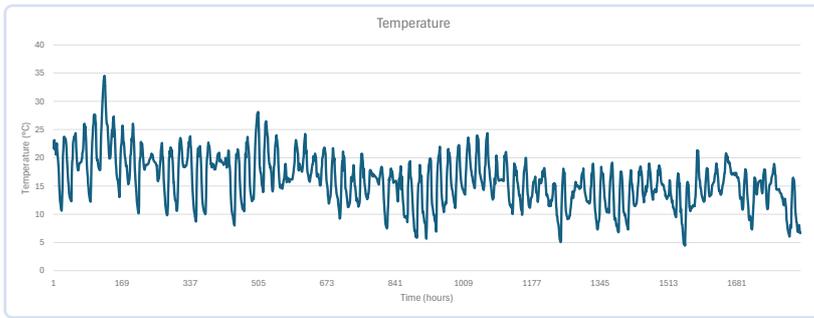


◀ Bild 3. So wurde der SenseLoRa-Empfänger in einer realen Anwendung zu Protokollierungszwecken eingesetzt.

einem RP2040-Mikrocontroller und verfügt über ein OLED-Display, einen microSD-Kartenslot (eine 16 GB microSD-Karte ist im Lieferumfang enthalten), ein RF96-Modul von HopeRF und einen USB-Anschluss. Dies ist der Teil, der mir am wenigsten gefällt. Der USB-Anschluss ist ein männlicher A-Typ (ein Stecker), der das Modul in einen großen Dongle verwandelt, den man in einen Computer stecken soll. Das schränkt die Platzierung stark ein, es sei denn, Sie fügen einen USB-Hub hinzu. Außerdem ist das Modul für einen USB-Stick recht breit (34 mm) und hoch (18 mm), so dass es wahrscheinlich den Zugang zu anderen



◀ Bild 4. Die Empfängerantenne sitzt auf dem Metalldach des Schuppens.



▲ **Bild 5.** Fast elf Wochen kontinuierliche Temperaturaufzeichnung in Grad Celsius, eine Messung pro Stunde. Ich hatte auf einen wärmeren Sommer gehofft :-).

Anschlüssen des Hosts blockieren dürfte. **Bild 3** zeigt, wie ich das Modul letztendlich mit einem USB-Steckernetzteil zur Stromversorgung verwendet habe. Damit das Display in den meisten Einbausituationen gut ablesbar bleibt, gibt es einen Druckknopf namens TFT, um es um 180° zu drehen. Eine Option zum Abschalten wäre ebenfalls wünschenswert gewesen. Die Antenne ist die gleiche wie die des Air Monitors. Ihr langes Kabel (5 m) ist praktisch, ebenso wie ihr Magnet, wenn Sie eine Metallfläche haben, an der Sie sie befestigen können (**Bild 4**). Wenn nicht, müssen Sie die Antenne mit Klebeband oder Kabelbindern oder Ähnlichem fixieren.

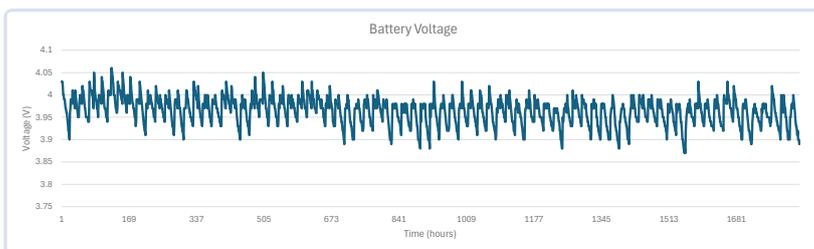
### Endlose Datenaufzeichnung

Der Empfänger zeigt die eingehenden Daten nicht nur an, sondern überträgt sie auch verbatim über seinen seriellen USB-Anschluss und speichert sie auf seiner microSD-Karte (falls eine eingelegt ist). Interessant ist, dass neue Daten an die bestehende Logdatei auf der microSD-Karte angehängt werden, so dass diese auch nach einem Stromausfall oder Neustart erhalten bleibt. Die Datei wird beim Booten nicht überschrieben oder neu erstellt. Daher gehen vorhandene Daten nicht verloren, es sei denn, Sie löschen sie absichtlich (**Bild 5** und **Bild 6**). Beachten Sie, dass Sie beim Aufzeichnen der Daten der seriellen Schnittstelle auch Boot- und Statusmeldungen erfassen können, wenn die TFT-Taste gedrückt wird. Dies könnte ein einfaches Datenkonvertierungsskript unterbrechen. Ein Python-Skript zur Aufzeichnung und Dekodierung von Daten ist unter [1] verfügbar.

▼ **Bild 6.** Die Batterie bleibt geladen, auch wenn die Sonne nicht scheint.

### Sehr hackbar

Sowohl der Luftmonitor als auch der LoRa-Empfänger sind sehr „hackbare“ Geräte. Die Schaltpläne,



Board-Design-Dateien (Eagle) und der Quellcode sind alle auf GitHub zusammen mit dem Benutzerhandbuch veröffentlicht [1][2]. Der LoRa-Empfänger basiert, wie gesagt, auf einem RP2040-Mikrocontroller, während der Air Monitor mit einem ESP32-S3-Modul ausgestattet ist. Beides sind bekannte Mikrocontroller in der Welt der Maker. Die Software ist Arduino-basiert, so dass Sie sie leicht an Ihre Bedürfnisse und Wünsche anpassen können.

### LoRa, einfach zu benutzen

Mit dem SenseLoRa-Konzept hat Makerfabs versucht, LoRa für Punkt-zu-Punkt-Verbindungen einfach nutzbar zu machen. Die Konfiguration von komplizierten LoRaWAN- und Cloud-Diensten wird so vermieden. Das System ist Plug-and-Play und funktioniert sofort nach dem Einschalten. Die Erweiterung des Systems ist mit minimaler Konfiguration möglich (lediglich die Geräte-ID muss eingestellt werden). Durch seine Einfachheit ist SenseLoRa praktisch, um schnell ein System zur Gewächshausüberwachung einzurichten. Was jedoch passiert, wenn Ihr Nachbar beschließt, das Gleiche zu tun, ist unklar.

Auch wenn einige Teile als industrietauglich bezeichnet werden, ist dies mit Vorsicht zu genießen. Der Air Monitor hat zwar ein IP68-Gehäuse, aber wenn man zwei Kabel (Antenne und Strom) durch dieselbe Kabeldurchführung führt, wird aus der „8“ wahrscheinlich eine „3“. Dies erhöht das Risiko des Eindringens von Wasser und reduziert den Schutz effektiv auf etwa IP63. Der Empfänger in Form eines großen USB-Dongles wirkt zudem wenig robust und ist ungeeignet für raue Industrieumgebungen, könnte aber in geschlossenen, weniger anspruchsvollen Umgebungen ausreichend sein. Dennoch überzeugt das SenseLoRa-Konzept mit seiner Einfachheit und Kosteneffizienz. Auch die Tatsache, dass es quelloffen ist und gehackt werden kann, macht es zu einem attraktiven System für Entwickler, Maker und kleine Unternehmen. Insgesamt zeigt SenseLoRa, dass einfache, zugängliche IoT-Lösungen möglich sind – mit einigen Hardware-Kompromissen, aber viel Potenzial für kreative und flexible Anwendungen. ◀

Übersetzung: Shaimaa Elroby -240407-02

### Sie haben Fragen oder Kommentare?

Haben Sie technische Fragen oder Kommentare zu diesem Artikel? Wenden Sie sich bitte per E-Mail an den Autor unter [clemens.valens@elektor.com](mailto:clemens.valens@elektor.com) oder an Elektor unter [redaktion@elektor.de](mailto:redaktion@elektor.de)

### WEBLINKS

[1] SenseLoRa, Benutzerhandbuch für den LoRa Receiver: <https://github.com/Makerfabs/SenseLoRa-LoRa-Receiver>

[2] SenseLoRa, Benutzerhandbuch für den Air Monitor: <https://github.com/Makerfabs/SenseLoRa-Industrial-grade-Air-Monitor>