



1. Termin

- ▶ Einführung in LoRa(WAN) und Micropython
- ▶ Erste Programme auf dem ESP32

2. Termin

- ▶ Name per LoRa versenden und einfache Alarmanlage
- ▶ Das TheThingsNetwork

3. Termin

- ▶ CO₂-Sensor am ESP32

4. Termin

- ▶ CO₂-Sensor kalibrieren
- ▶ Bau einer CO₂-Ampel

5. Termin

- ▶ Projektarbeit: CO₂-Ampel

6. Termin

- ▶ Sendezeit sinnvoll nutzen und Energie sparen
- ▶ Projektideen???

Internet of Things (IoT)

Was ist das?

- ▶ Vernetzung von Gegenständen mit dem Internet
- ▶ Automatisierbare Prozesse
- ▶ Zugriff über Schnittstellen
 - ▶ Maschine-Maschine
 - ▶ Mensch-Maschine

Internet of Things (IoT)

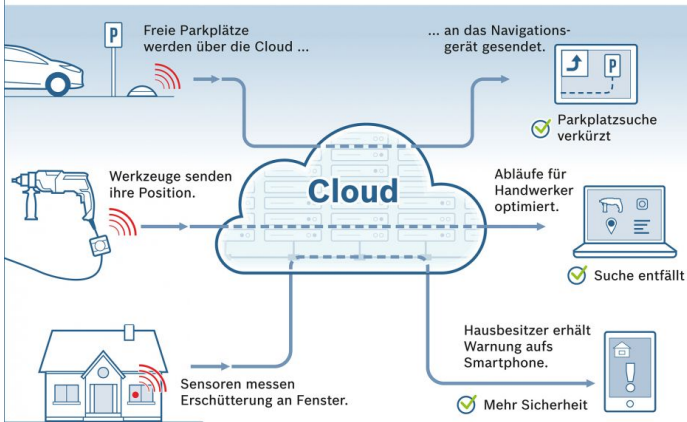
Beispiele

Wenn sich der freie Parkplatz meldet

Cloud Computing zählt zu den technologischen Grundlagen im Internet der Dinge. Drei Anwendungsbeispiele:

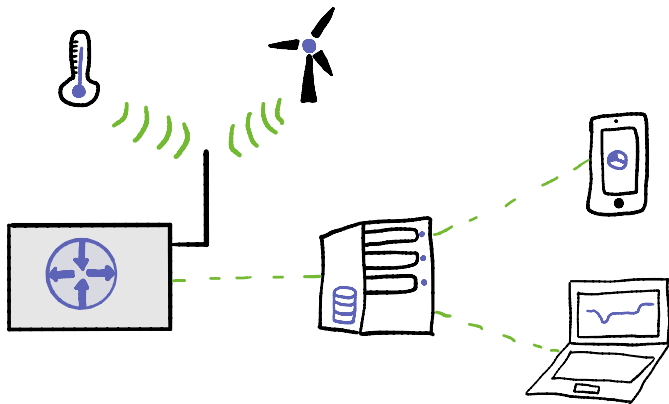


BOSCH
Technik fürs Leben



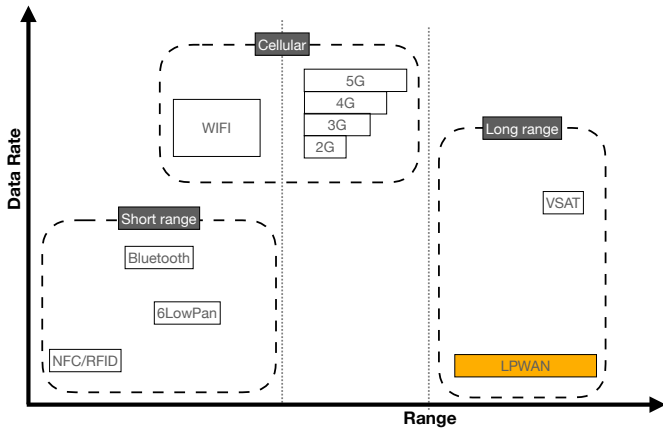
Quelle: Bosch

LoRa(WAN)



LoRa(WAN)

Übersicht der Technologien



LoRa(WAN)

Was ist das überhaupt?

LoRa (**Long Range**):

- ▶ Übertragungsart (physikalische Schicht)
- ▶ *Wie werden Daten übertragen?*

LoRaWAN (**Long Range Wide Area Network**):

- ▶ Netzwerkprotokoll
- ▶ *Wie sind die Daten aufgebaut?*

LoRa(WAN)

Vor- und Nachteile

Vorteile:

- ▶ große Reichweite (2 - 15 km)
- ▶ niedriger Energiebedarf $\Rightarrow$ Batteriebetrieb
- ▶ Verschlüsselung der Daten möglich
- ▶ Bidirektionale Kommunikation möglich

Nachteile:

- ▶ begrenzte Sendezeit (30 Sekunden Uplink/24 Stunden)
- ▶ begrenzte Datenmengen

CO₂ Ampel

Luftqualität und Viren - wie ist der Zusammenhang?



Abbildung: CO₂ Ampel der Firma MB Systemtechnik

- ▶ Viren können über Aerosole verbreitet werden
⇒ Lüften reduziert die Konzentration
- ▶ Über CO₂ Sensoren kann die Luftqualität gemessen werden
- ▶ Luftqualität↓ → Anzahl der Aerosole↑

CO₂ Ampel

Was brauchen wir dafür?

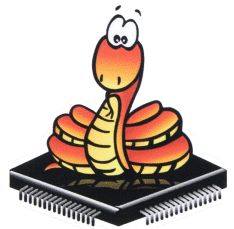
ESP32 mit LoRa-Chip



Gateway



Micropython



Einführung in Micropython

Was ist Micropython?

- ▶ Vereinfachte Version der Programmiersprache *Python*
- ▶ Speziell für Microcontroller wie den ESP
- ▶ Muss zunächst auf dem ESP installiert werden

→ Wie bekommen wir Micropython auf den ESP?

Einführung in Micropython

Voraussetzungen

Benötigte Hardware:

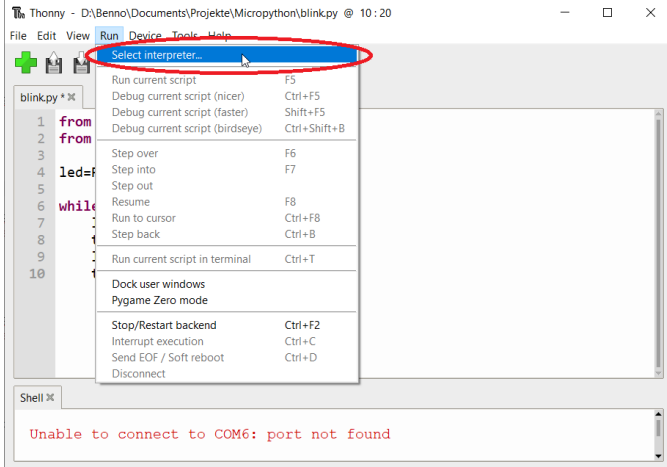
- ▶ ESP32 mit LoRa-Chip
- ▶ Micro-USB-Kabel

Benötigte Software:

- ▶ Texteditor: *Thonny*
→ Download: <https://thonny.org/>
- ▶ Micropython-Firmware für den ESP32
→ Download: <https://micropython.org/download/esp32/>

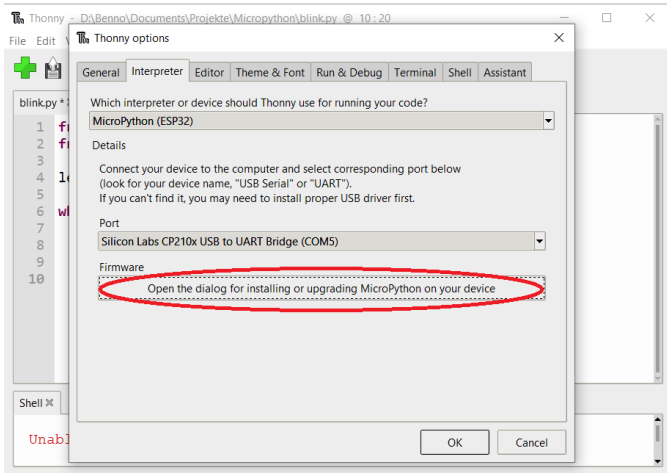
Einführung in Micropython

Installation auf dem ESP



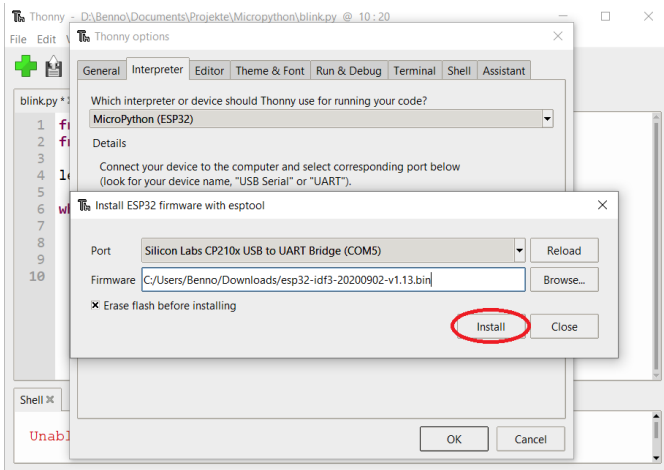
Einführung in Micropython

Installation auf dem ESP



Einführung in Micropython

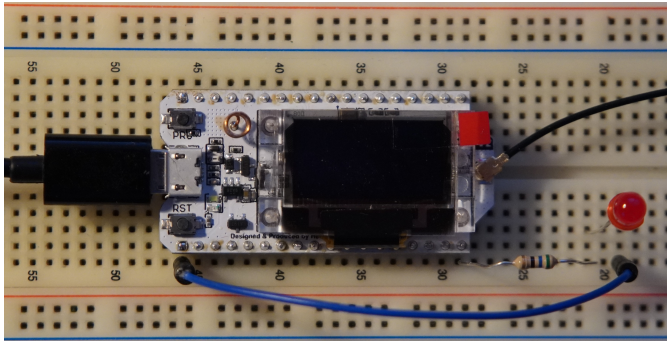
Installation auf dem ESP



Einführung in Micropython

Unser erstes Programm

Wie können wir eine LED blinken lassen?



1x Breadboard, 1x LED, 1x Widerstand (55Ω), 1x Jumper-Kabel

Einführung in Micropython

Unser erstes Programm

Wie können wir eine LED blinken lassen?

ledblink.py

```
1 from machine import Pin
2 from time import sleep
3
4 led = Pin(16,Pin.OUT) # LED-Pin
5
6 while(True):
7     led.value(1) # LED an
8     sleep(0.5)
9     led.value(0) # LED aus
10    sleep(0.5)
```

Erste Schritte mit dem ESP

Unser Name auf dem Display

display.py

```
1 from machine import Pin, I2C
2 import ssd1306
3 from time import sleep_ms
4
5 rst = Pin(16, Pin.OUT) # Display-Reset
6 rst.value(0)
7 sleep_ms(20)
8 rst.value(1)
9
10 i2c = I2C(-1, scl=Pin(15,Pin.OUT,Pin.PULL_UP),sda=Pin(4,
    Pin.OUT,Pin.PULL_UP))
11 oled = ssd1306.SSD1306_I2C(128,64,i2c) # Hoehe,Breite
12
13 oled.text("DEIN_NAME",0,30) # "text",x,y
14 oled.show() # Text anzeigen
```

→ zusätzlich benötigt auf dem ESP: ssd1306.py