|  |
| --- |
| MicroPython  deel 2  *Github, organiseren en sensoren* |

Geschreven door

Arjan Kamberg  
<https://www.linkedin.com/in/arjankamberg/>

Leerjaar: 2023/ 2024

Copyright  
Arjan Kamberg

Van stroomdiagram naar Python code  
© 2023, Arjan Kamberg  
Uitgegeven in eigen beheer  
(Python@AKamberg.nl)

Alle rechten voorbehouden.  
Niets uit deze uitgave mag worden verveelvoudigd, opgeslagen in een geautomatiseerd gegevensbestand en/of openbaar gemaakt in enige vorm of op enige wijze, hetzij elektronisch, mechanisch, door fotokopieën, opnamen of op enige andere manier zonder voorafgaande schriftelijke toestemming van de uitgever. Het gebruik maken van dit boek voor opleidingsdoeleinden mag alleen gedaan worden met uitdrukkelijke toestemming van de uitgever.

Versie Datum: 25-10-2023

Inhoudsopgave

[Voorwoord 3](#_Toc149132083)

[1 Welke kennis heb je nodig. 4](#_Toc149132084)

[2 GitHub 5](#_Toc149132085)

[3 Organiseren van code 6](#_Toc149132086)

[5 MQTT Installeren 7](#_Toc149132087)

# Voorwoord

Alle apparaten om ons heen hebben tegenwoordig wel een computerprogramma in zich dat de functies bepaald. Dit boek is geschreven voor MBO-studenten die programmeren leren aan de hand van de taal Python. Python is een taal die op steeds meer gebieden ingezet wordt door zijn kracht en eenvoud.

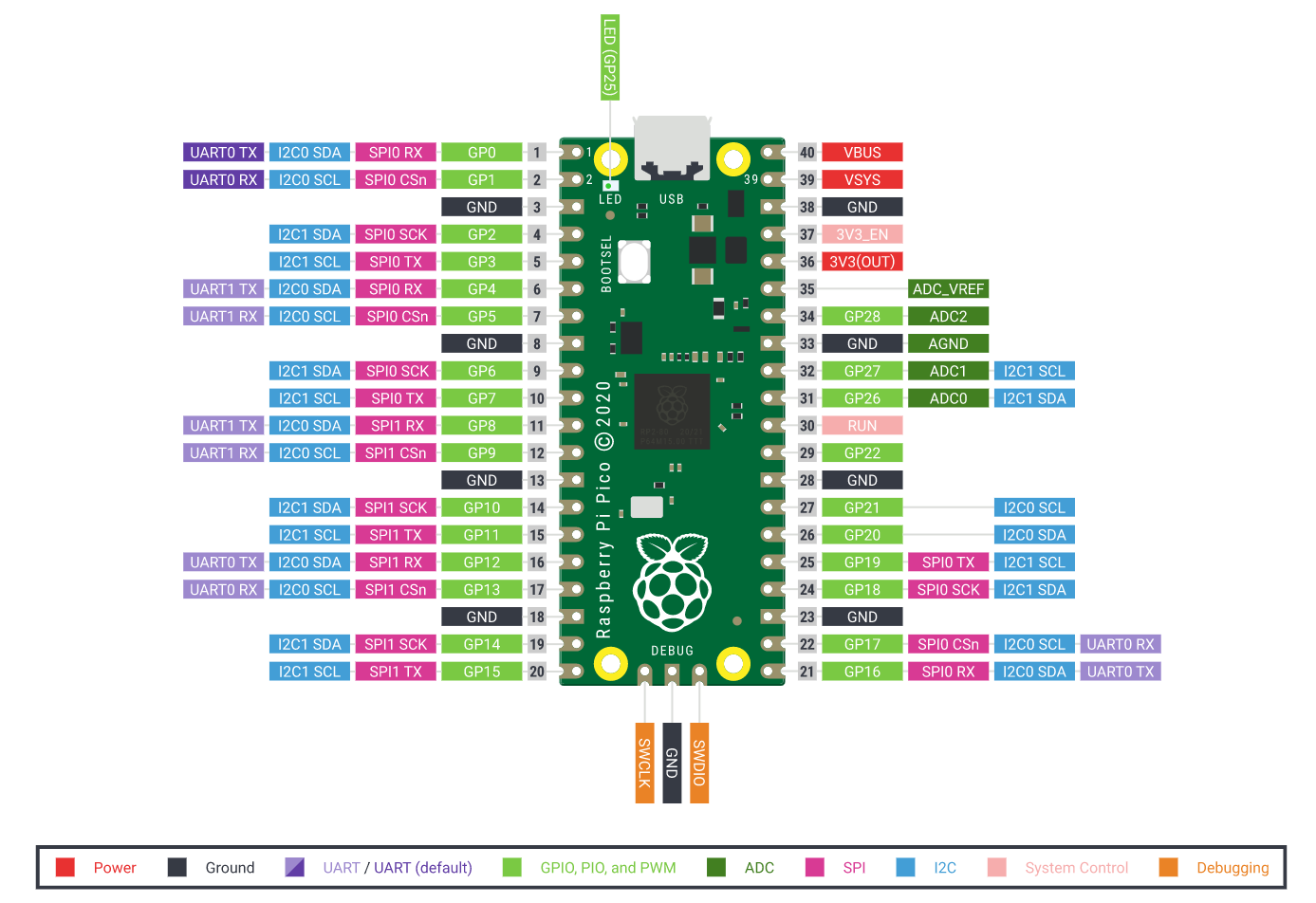
In dit boek worden niet alle dingen behandeld, de focus zit op het begrip van programmeren leren. Zo worden lussen bewust alleen met while-statements gemaakt. De opbouw van het programmeren moet eenduidig zijn en er moet zo min mogelijk keuzestress zijn bij het programmeren. De overwegingen en didactische uitleg wordt beschreven in de docenten versie van dit boek.

Bij een goed begrip van programmeren, en een opbouw die duidelijk en testbaar is wordt bereikt dat het eindproduct ook uitbreidbaar is en dat het zich aan de conventies houdt die bruikbaar zijn bij programmeurs met jaren ervaring.



Figuur 1: Programmeurs die iemand anders code niet snappen. (C) CSEstack.org

# Welke kennis heb je nodig.



# GitHub

# Organiseren van code

# MQTT Installeren

# MQTT Python – Publisher

# MQTT Python Subscriber

# CSV-bestanden

# MQTT-microPython Publisher

# MQTT-microPython Subscriber

# Eindopdracht

Voor de onderstaande opdracht heb je zo’n 6 uur tot je beschikking.

Maak een slim systeem voor het meten van de temperatuur en luchtvochtigheid.

Op een Raspberry Pi Pico met micropython wordt een MQTT publisher opgezet die verschillende topics doorgeeft van de temperatuur en luchtvochtigheid.

De topics zijn: Als er een nieuwe minimale of maximale temperatuur en luchtvochtigheid wordt gemeten.

Als beveiliging wordt iedere minuut de op dat moment bekende minimale en maximale temperatuur en luchtvochtigheid doorgestuurd.

Iedere 5 seconden wordt de luchtvochtigheid en temperatuur gemeten.

Zorg dat alles in verschillende bestanden plaatsvindt om het overzichtelijk te houden.

Dus de MQTT-file is alleen om data te verzenden. Daarin wordt GEEN metingen en berekeningen gedaan.

Een tweede bestand waar de metingen gedaan worden. In eerste instantie is dit een bestand waar random waarden gegenereerd worden. De Temperatuur zal tussen de 10 en 50 graden Celsius gegenereerd worden, en de luchtvochtigheid tussen 0 en 100 %.

Later kunnen we die file wijzigingen met een file die de DHT11 uitleest. De DHT11 is een sensor die zowel temperatuur als luchtvochtigheid meet.