

Das MakePython ESP32 Development Kit Programmierung & Projekte

Ein Schnellkurs zum ESP32-Mikrocontroller



Erweitertes Handbuch von Elektor

● © 2023: Elektor Verlag GmbH, Aachen.

1. Auflage 2023

● Alle Rechte vorbehalten.

Die in diesem Buch veröffentlichten Beiträge, insbesondere alle Aufsätze und Artikel sowie alle Entwürfe, Pläne, Zeichnungen und Illustrationen sind urheberrechtlich geschützt. Ihre auch auszugsweise Vervielfältigung und Verbreitung ist grundsätzlich nur mit vorheriger schriftlicher Zustimmung des Herausgebers gestattet.

Die Informationen im vorliegenden Buch werden ohne Rücksicht auf einen eventuellen Patentschutz veröffentlicht. Die in diesem Buch erwähnten Soft- und Hardwarebezeichnungen können auch dann eingetragene Warenzeichen sein, wenn darauf nicht besonders hingewiesen wird. Sie gehören dem jeweiligen Warenzeicheninhaber und unterliegen gesetzlichen Bestimmungen.

Bei der Zusammenstellung von Texten und Abbildungen wurde mit größter Sorgfalt vorgegangen. Trotzdem können Fehler nicht vollständig ausgeschlossen werden. Verlag, Herausgeber und Autor können für fehlerhafte Angaben und deren Folgen weder eine juristische Verantwortung noch irgendeine Haftung übernehmen.

Für die Mitteilung eventueller Fehler sind Verlag und Autor dankbar

● Erklärung

Autor und Herausgeber haben sich nach besten Kräften bemüht, die Richtigkeit der in diesem Buch enthaltenen Informationen sicherzustellen. Dennoch können sie keine Haftung gegenüber irgendeiner Partei für Verluste oder Schäden übernehmen, die durch Fehler oder Auslassungen in diesem Buch verursacht wurden, unabhängig davon, ob diese Fehler oder Auslassungen auf Fahrlässigkeit, Unfälle oder andere Ursachen zurückzuführen sind.

Umschlaggestaltung: Elektor, Aachen

Übersetzung: Günter Spinner

Satz und Aufmachung: D-Vision, Julian van den Berg | Oss (NL)

Druck: Ipskamp Printing, Enschede, Niederlande

Das Coverbild ist eine Kreation von INKSY: inksy.art

● ISBN 978-3-89576-553-7 Print

ISBN 978-3-89576-554-4 eBook

● Elektor-Verlag GmbH, Aachen

www.elektor.de

Elektor ist ein Teil von EIM, der weltweit wichtigsten Quelle für technische Informationen und Elektronik-Produkte für Ingenieure und Elektronik-Entwickler und für Firmen, die diese Fachleute beschäftigen. Das internationale Team von Elektor entwickelt Tag für Tag hochwertige Inhalte für Entwickler und DIY-Elektroniker, die über verschiedene Medien (Magazine, Videos, digitale Medien sowie Social Media) in zahlreichen Sprachen verbreitet werden. www.elektor.de

Inhalt

Vorwort	11
Kapitel 1 - Der ESP32 Prozessor	13
1.1 Überblick	13
1.2 Die Architektur des ESP32.....	14
1.2.1 Die CPU	15
1.2.2 Interner Speicher.....	16
1.2.3 Externer Speicher	16
1.2.4 Universelle Timer	16
1.2.5 Watchdog Timer	16
1.2.6 Der Systemtakt	16
1.2.7 Die Echtzeituhr (Real Time Clock - RTC).....	16
1.2.8 Allgemeine Ein- und Ausgänge (GPIOs - General Purpose Input-Outputs)	16
1.2.9 Analog-Digital-Wandler (ADCs).....	16
1.2.10 Digital-Analog-Wandler (DACs).....	17
1.2.11 Hallsensor	17
1.2.12 Temperatursensor.....	17
1.2.13 Berührungssensoren	17
1.2.14 UART.....	17
1.2.15 I ² C Schnittstelle	17
1.2.16 I ² S Schnittstelle	17
1.2.17 Infrarot-Schnittstelle.....	17
1.2.18 Pulsweitenmodulation	17
1.2.19 LED PWM	18
1.2.20 Impulszähler	18
1.2.21 SPI Schnittstelle	18
1.2.22 Hardwarebeschleuniger	18
1.2.23 Wi-Fi / WLAN	18
1.2.24 Bluetooth	18
1.2.25 Controller Area Network (CAN)	18
1.2.26 SD-Kartenunterstützung	18

1.3	ESP32 Entwicklungsboards.	18
1.4	Zusammenfassung.	19
Kapitel 2 • Das MakePython ESP32 Development Kit		20
2.1	Überblick	20
2.2	Die MakePython ESP32 Hardware	20
2.3	Das MakePython ESP32-Entwicklungsboard	21
2.4	Inbetriebnahme der Entwicklungsplatine	24
2.5	Zusammenfassung	24
Kapitel 3 • Die MakePython ESP32-Entwicklungsumgebung		25
3.1	Überblick	25
3.2	Die µPyCraft IDE	25
3.2.1	Starten der µPyCraft IDE	28
3.2.2	Ein Beispielprogramm mit der µPyCraft IDE	31
3.2.3	Verwenden der Shell	35
3.3	Die Thonny IDE	36
3.3.1	Ein Beispielprogramm für die Thonny IDE	38
3.3.2	Benutzung der Shell.	40
3.4	µPyCraft oder Thonny?	40
3.5	Reinstallation/Upgrade der ESP32 firmware	40
3.5.1	Neuinstallation/Upgrade der Firmware, wenn noch eine Kommunikation mit dem Entwicklungsboard möglich ist	42
3.5.2	Neuinstallation/Upgrade der Firmware, wenn keine Kommunikation mit dem Entwicklungsboard besteht	43
3.6	Zusammenfassung	44
Kapitel 4 • Einfache Programme – Nur Software		45
4.1	Überblick	45
4.2	Beispiele.	45
4.2.1	Durchschnitt von zwei über die Tastatur eingelesenen Zahlen	45
4.2.2	Durchschnitt von 10 Zahlen, die von der Tastatur gelesen werden.	46
4.2.3	Oberfläche eines Zylinders	46
4.2.4	Konvertierung von °C zu °F	47
4.2.5	Oberfläche und Volumen eines Zylinders – mit Benutzerfunktion	47

4.2.6	Quadratzahlentabelle	48
4.2.7	Tabelle der Sinusfunktion	49
4.2.8	Tabelle der trigonometrischen Funktionen Sinus, Cosinus und Tangens	50
4.2.9	Trigonometrische Funktion eines bestimmten Winkels	50
4.2.10	Wörter in umgekehrter Reihenfolge	51
4.2.11	Taschenrechner	52
4.2.12	Würfel	52
4.2.13	Listen sortieren	53
4.2.14	Dateiverarbeitung – Schreiben	53
4.2.15	Dateiverarbeitung – Lesen	54
4.2.16	Multiplikationstabelle	54
4.2.17	Ungerade oder gerade	55
4.2.18	Binär, oktal und hexadezimal	55
4.2.19	Addieren von zwei Matrizen	56
4.2.20	Flächeninhalte	56
4.2.21	Lösung einer quadratischen Gleichung	58
4.2.22	Matrixmultiplikation	59
4.2.23	Fakultät einer Zahl	60
4.2.24	Zinseszins	60
4.2.25	Erraten Sie eine Zahl	61
4.3	Zusammenfassung	62
Kapitel 5 - LED Projekte mit dem MakePython ESP32 Development Kit		63
5.1	Überblick	63
5.2	Projekt 1: Blinkende LED	63
5.3	Projekt 2: SOS Signal	68
5.4	Projekt 3: Blinkende LED – mit Timer-Steuerung	71
5.5	Projekt 4: Wechselblinker	72
5.6	Projekt 5: Taster-Steuerung	74
5.7	Projekt 6: Ändern der LED-Blinkrate über Taster-gesteuerte Interrupts	76
5.8	Projekt 7: Lauflicht	80
5.9	Projekt 8: Binärzähler mit LEDs	82

5.10	Projekt 9: Weihnachtsbeleuchtung (Acht zufällig blinkende LEDs)	85
5.11	Projekt 10: Elektronischer Würfel	86
5.12	Projekt 11: Glückstag der Woche	90
Kapitel 6 • Projekte zur Pulsweitenmodulation (PWM)		94
6.1	Überblick	94
6.2	Grundlegende Theorie der Pulsweitenmodulation	94
6.3	Die PWM-Kanäle des ESP32	96
6.4	Projekt 1: Erzeugen Sie eine 1000-Hz-PWM-Wellenform mit 50% Einschaltdauer ...	97
6.5	Projekt 2: LED-Helligkeitssteuerung	98
6.6	Projekt 3: Messung der Frequenz und des Tastverhältnisses einer PWM-Wellenform	99
6.7	Projekt 4: Melodiengenerator	101
6.8	Projekt 5: Einfache elektronische Orgel	104
6.9	Projekt 6: Servomotorsteuerung	106
6.10	Projekt 7: Servomotor DS18B20 Thermometer	110
Kapitel 7 • Analog-Digital-Wandler (ADC)-Projekte		114
7.1	Überblick	114
7.2	Projekt 1: Voltmeter	115
7.3	Projekt 2: Aufzeichnen der analogen Eingangsspannung	116
7.4	Projekt 3: Der ESP32-interne Temperatursensor	117
7.5	Projekt 4: Ohmmeter	118
7.6	Projekt 5: Modul mit lichtempfindlichem Widerstand	119
7.7	ADC Linearität	122
Kapitel 8 • Digital-Analog-Wandler (DAC)-Projekte		124
8.1	Überblick	124
8.2	Projekt 1: Erzeugung von Festspannungen	124
8.3	Projekt 2: Erzeugung eines Sägezahnsignals	126
8.4	Projekt 3: Erzeugung eines Dreiecksignals	127
8.5	Projekt 4: Beliebige periodische Wellenform	129
8.6	Projekt 5: Erzeugung eines Sinussignals	131
8.7	Projekt 6: Erzeugen eines genauen Sinussignals mit Timer-Interrupts	134
Kapitel 9 • Verwendung des OLED-Displays		137

9.1	Überblick	137
9.2	Installieren der SSD1306-Treibersoftware	138
9.3	Anzeigen von Text.	139
9.4	Anzeigen geometrischer Figuren.	140
9.5	Weitere nützliche Funktionen Einige andere nützliche OLED-Funktionen sind:	142
9.6	Projekt 1: Sekundenzähler	147
9.7	Projekt 2: Ereigniszähler	149
9.8	Projekt 3: DS18B20 OLED-basiertes Digitalthermometer	150
9.9	Projekt 4: EIN-AUS-Temperaturregler (on/off-Regler)	153
9.10	Projekt 5: Messung von Temperatur und Luftfeuchtigkeit	156
9.11	Projekt 6: Ultraschall-Entfernungsmessung	159
9.12	Projekt 7: Messung der Körpergröße (Stadiometer)	164
9.13	Projekt 8: Messung der Herzfrequenz (Pulsmessung)	166
9.14	Zeichnen von Bitmaps	170
Kapitel 10 - Die anderen Sensoren im Kit		177
10.1	Überblick	177
10.2	Projekt 1: Diebstahlalarm	177
10.3	Projekt 2: Schall-gesteuertes Licht	179
10.4	Projekt 3: Infrarot-Hinderniserkennung mit Summer	181
10.5	Projekt 4: WS2812 RGB-LED-Ring	183
10.6	Verwenden der Echtzeituhr (RTC)	186
10.6.1	Projekt 5: Temperatur- und Feuchtigkeitsmesswerte mit Zeitstempel versehen	187
10.7	Der integrierte Hall-Effekt-Sensor	187
Kapitel 11 - Netzwerkprogrammierung		188
11.1	Überblick	188
11.2	Verbinden mit einem Wi-Fi-Netzwerk	188
11.3	Projekt 1: WLAN-Scanner	189
11.4	Verwenden der Socket-Bibliothek	191
11.4.1	UDP Programme	191
11.4.2	Projekt 2: Fernsteuerung via Internetbrowser (mittels Smartphone oder PC) - Webserver	195
11.4.3	Projekt 3: Speichern von Temperatur- und Feuchte-Daten in der Cloud	200

Kapitel 12 • MakePython ESP32 im Low-Power-Betrieb	206
12.1 Überblick	206
12.2 ESP32-Leistungs-Modi	206
12.3 Projekt 1: Verwenden eines Timers zum Aufwecken des Prozessors	207
ANHANG A	210
Dateiübertragung zwischen MakePython ESP32 und einem PC	210
Index	211