

IoT Projekte mit Mynewt OS

# Band 1: Calliope Mini als Einstieg

Programmieren mit C und CLion

Von Alfred Schilken

# Inhaltsverzeichnis

Vorwort.....	1
Für wen ist dieses Buch?.....	2
Was hier zu finden ist.....	3
Was wird gebraucht?.....	5
Was nicht in diesem Buch steht.....	6
Schreibstil.....	6
Quellcode.....	7
Webseiten zum Buch.....	7
Typographische Vereinbarungen.....	8
<b>Kapitel 1 – Die Maker.....</b>	<b>9</b>
Vom Mikroprozessor zum Mikrocontroller.....	9
Arduino.....	10
ARM mbed.....	11
Raspberry Pi.....	13
Internet of Things (IoT).....	13
<b>Kapitel 2 – BBC micro:bit.....</b>	<b>15</b>
Die technischen Daten.....	15
LED-Matrix und Buttons.....	15
Vergleich mit Arduino und Raspberry Pi.....	16
<b>Kapitel 3 – Der Calliope.....</b>	<b>17</b>
Die Platine.....	19
Zielgruppe: nicht nur für Grundschüler.....	20
BBC-micro:bit kompatibel – und mehr.....	21
Ein Blockschaltbild.....	22
Die RGB-LED.....	23
Die Tonausgabe – „Buzzer“.....	23
Der Motortreiber.....	24
Das Miniatur-Mikrofon.....	25
Die Grove-Buchsen.....	26

Ein externer Speicherchip (optional).....	26
Der Erweiterungsport für Maker.....	27
<b>Kapitel 4 – Laufzeitumgebungen.....</b>	<b>29</b>
Bare Metal.....	29
MikroPython.....	29
RTOS – Real-Time-Operating-System.....	30
<b>Kapitel 5 – Entwicklungsumgebungen.....</b>	<b>31</b>
Calliope mini, PXT Editor und Open Roberta Lab .....	31
MicroPython.....	34
Die Grove-Buchsen.....	26
Yotta – mbed offline .....	37
mbed-cli .....	44
Segger Embedded Studio – SES-IDE .....	44
Keil $\mu$ Vision IDE.....	46
Eclipse + GCC.....	46
Visual Studio Code.....	47
CLion + Yotta.....	47
CLion + newt.....	47
Fazit – Vergleich der Entwicklungstools.....	47
<b>Kapitel 6 – Betriebssysteme für Calliope.....</b>	<b>49</b>
ARM mbed OS.....	49
Lancaster Runtime.....	50
RIOT.....	52
Zephyr.....	53
Apache Mynewt.....	54
Fazit – Vergleich der Betriebssysteme.....	55
<b>Kapitel 7 – Apache Mynewt.....</b>	<b>56</b>
Pures C anstatt C + + .....	56
Projekte und Repositories.....	58
Packages.....	59
Apps.....	60

BSP – Board Support Package.....	60
HAL – Hardware Abstraction Layer.....	65
Architektur.....	67
Target = App + BSP + Profil.....	67
Bootloader.....	69
Aufteilung des Flash-Speichers.....	70
Die Flash-Map.....	71
BSP für den Single-Image-Modus.....	73
Konventionen.....	75
Treiber für Kommunikation und Sensoren.....	75
OS Tasks.....	79
OS Queues.....	80
Timer – "callout".....	81
Statistiken.....	81
Logging.....	84
Systemkonfiguration – syscfg-Parameter.....	84
Config-Variablen.....	87
Konsole.....	87
Shell.....	88
<b>Kapitel 8 – Installation auf macOS.....</b>	<b>90</b>
Installation des newt-Tools.....	90
Installation ARM Toolchain, OpenOCD und CoolTerm.....	91
Installation der „nativen“ Toolchain.....	93
CLion installieren.....	95
<b>Kapitel 9 – Installation auf Windows 10.....</b>	<b>96</b>
MinGW Shell für Windows installieren.....	96
Ausführbare Version des newt-Tools installieren.....	97
Git für Windows installieren.....	99
Installation ARM Toolchain, OpenOCD, CoolTerm.....	100
CLion installieren.....	102
<b>Kapitel 10 – Apps bauen mit dem newt-Tool.....</b>	<b>103</b>

Überblick newt – Subkommandos.....	103
Ein neues Projekt anlegen.....	104
„blinky“ bauen und auf Mac oder Linux ausführen.....	108
Debuggen eines nativen Targets.....	112
„blinky“ Bauen und auf Calliope ausführen.....	113
<b>Kapitel 11 – CLion + newt: ein gutes Gespann.....</b>	<b>118</b>
Mynewt-Projekt für CLion vorbereiten.....	118
Ein Mynewt Projekt in CLion öffnen.....	119
Projekt in CLion aktualisieren.....	122
Debuggen mit gdb und OpenOCD.....	123
<b>Kapitel 12 – Die App mydrivertest.....</b>	<b>129</b>
Ein externes Repository einbinden.....	129
Auf Buttons reagieren.....	135
Der Button-Treiber.....	137
Die Shell aktivieren.....	142
Shell-Commands.....	143
CoolTerm konfigurieren.....	144
Die Shellkommandos der App.....	146
Test des Sound-Treibers.....	149
Die GPIOs des Calliope.....	152
GPIO – den digitalen Status eines Pins einlesen.....	154
GPIO-Scanner.....	157
GPIO Ausgabelevel schalten.....	158
Der I2C Bus.....	159
I2C probe.....	160
I2C Scanner.....	160
Analog-Digital-Wandler.....	163
Zusätzliche Repositories für den ADC-Treiber.....	164
ADC testen.....	168
LED-Matrix testen.....	171
OLED-Display testen.....	173
LED-Balken testen.....	174

LED-Balken testen .....	174
15x7-LED-Matrix von Adafruit testen .....	176
<b>Kapitel 13 – Das Seeedstudio Grove-System.....</b>	<b>178</b>
Verschiedene Steckerbelegungen.....	180
Digitale GPIOs .....	180
Analoger Input.....	181
UART – Serielle Schnittstelle.....	181
I2C – Bus für bis zu 112 Geräte.....	182
Eigene Grove-Module löten.....	183
Grove PROTOSHIELD .....	183
Potenziometer am Grove-Kabel.....	184
Grove LED-Balken.....	184
LEDs direkt am Grove-Kabel .....	185
<b>Kapitel 14 – Interessante Chips und Boards.....</b>	<b>186</b>
I2C OLED-Display.....	187
I2C 15x7 LED-Matrix – Adafruit CharliePlex.....	189
I2C pcf8591-Board A/D und D/A-Wandler mit Sensoren.....	190
I2C MCP3008 ADC 10BIT A/D-Wandler.....	191
I2C RTC Echtzeit-Uhren .....	191
I2C 24FC128 128Kbit EEPROM.....	192
I2C Port-Expander mit MCP23017.....	194
Port-Expander sx1509 I2C-Breakout.....	197
Port Expander pcf8574 I2C-Breakout.....	198
Temperaturmessung mit I2C.....	199
<b>Kapitel 15 – Projekt Lügendetektor – stresstest-App.....</b>	<b>200</b>
Seedstudio Grove-Modul.....	201
ADC konfigurieren.....	201
Der Quelltext der stresstest-App.....	202
Feedback über eine Plotter-Kurve in der Arduino-IDE.....	205
<b>Kapitel 16 – Projekt Herzschlag-Monitor.....</b>	<b>210</b>
Grove Ohrclip Herzfrequenz Sensor – GRV HEART RATE3.....	210

Grove Finger-Clip Herzschlag-Sensor – GRV HEART RATE1.....	215
Pulsesensor Kickstarter Projekt von 2011.....	215
<b>Kapitel 17 – Schaltschwelle der Digital-Eingänge bestimmen</b> .....	<b>216</b>
<b>Kapitel 18 – Konfigurierbarer iBeacon</b> .....	<b>218</b>
iBeacon – ein Beacon von Apple.....	218
Konfigurieren mit Systemkonfiguration in syscfg.yml.....	223
Konfigurationsvariablen in der Shell.....	225
Major und Minor in der Shell konfigurieren.....	226
Statistik-Zähler auswerten.....	233
Eigene Zähler für die Statistik.....	237
Drei Logging-Varianten.....	237
Logging auf die Konsole.....	238
Logging im Flash für die spätere Analyse.....	242
Eddystone Beacon – Googles Beacon-Variante.....	245
Hybrid-Beacon – iBeacon + Eddystone.....	246
<b>Kapitel 19 – Die App bleadc</b> .....	<b>250</b>
BLE-Grundbegriffe: Central, Peripheral, Service, Characteristic.....	250
Service und Characteristics initialisieren.....	251
Auf READ- und WRITE-Zugriffe reagieren.....	253
NOTIFY- und INDICATE-Nachrichten auslösen.....	256
Der GAP-Layer.....	258
Advertising.....	258
Auf GAP-Events reagieren.....	260
<b>Kapitel 20 – Die App ble_uart</b> .....	<b>266</b>
Ein Chat über Bluetooth.....	266
Eine Minimal-Shell über Bluetooth.....	270
Bluefruit App für iOS und Android.....	275
<b>Kapitel 21 – MI-Band fernauslösen – blemib-App</b> .....	<b>278</b>
Bluetooth-Discovery – auf Advertising-Pakete reagieren.....	280
Der GAP-Event-Handler.....	281

Verbindungskandidaten selektieren über Bluetooth-Adresse.....	284
Die Peer-Bibliothek.....	286
Scannen und Speichern der GATT-Details eines Peripherals.....	287
Schreibzugriff auf ein Characteristic des Peripherals.....	289
Logging von Connect, Discovery und write.....	291
<b>Kapitel 22 – Das newtmgr-Tool.....</b>	<b>293</b>
<b>Kapitel 23 – Troubleshooting.....</b>	<b>297</b>
Prüfen, ob das Programm mit einer Reboot-Loop crasht.....	297
Quelltext einer Assert-Adresse mit gdb analysieren.....	298
Absturz-Adresse in Map-Datei suchen.....	298
Keine Reaktion in CoolTerm trotz aktivierter Shell.....	299
Reboot stoppt nicht bei Breakpoint.....	299
Breakpoints funktionieren nicht.....	301
Selten – aber möglich: USB-Kabel defekt.....	301
Laufzeit-Daten des OS-Moduls in der Shell.....	309
<b>Kapitel 24 – Ausblick.....</b>	<b>303</b>
Weitere Projekte mit Grove-Modulen.....	303
Weitere Bluetooth-Projekte.....	303
Es muss nicht immer Bluetooth sein.....	304
Dateisystem für die Speicherung von Messreihen nutzen.....	304
<b>Kapitel 25 – Anhang.....</b>	<b>306</b>
Übergangslösung falls newt target cmake nicht funktioniert.....	306
newt selbst bauen.....	306
Laufzeit-Daten des OS-Moduls in der Shell.....	309
Speicherbedarf reduzieren.....	312
Linux als Entwicklungsrechner.....	313
Alle Tools prüfen.....	315
macOS GDB signieren.....	316
Die 5x5 LED-Matrix.....	317
newt vals – yml-Werte für bsp, app, target auflisten.....	318
Upgrade auf die nächste Mynewt-Version.....	320



<b>Git für die Codeverwaltung benutzen.....</b>	<b>322</b>
<b>Grove LED-Balken.....</b>	<b>184</b>
<b>Bootloader aus dem Repository bauen und installieren.....</b>	<b>324</b>
<b>Target für Bootloader mit newt erzeugen.....</b>	<b>325</b>
<b>Source-Level-Debuggen in CLion.....</b>	<b>327</b>
<b>Funktionen, Symbole, Dateien finden.....</b>	<b>334</b>
<b>Änderungen vorangegangener Versionen wiederfinden .....</b>	<b>334</b>
<b>GDB Kommandos.....</b>	<b>335</b>
<b>Die wichtigsten GDB und OpenOCD Kommandos.....</b>	<b>337</b>
<b>Sublime Text Editor für die Suche.....</b>	<b>338</b>
<b>JLogicanalyzer benutzen.....</b>	<b>338</b>
<b>Meilensteine Hard- und Software.....</b>	<b>341</b>
<b>Bezugsquellen.....</b>	<b>342</b>
<b>Links ins Internet.....</b>	<b>345</b>
<b>Literaturverzeichnis .....</b>	<b>347</b>
<b>Glossar.....</b>	<b>349</b>