

Inhaltsverzeichnis

Einleitung.....	11
Warum STM32F4?	11
Zielgruppe dieses Buchs.....	12
Voraussetzungen.....	13
Aufbau des Buchs	14
Teil I Grundlagen	23
1 Die Hardware des STM32F4xx-Mikrocontrollers	25
1.1 Überblick über die STM32F4xx-Familie	25
1.2 Der STM32F446	27
1.2.1 Varianten des STM32F446	27
1.2.2 Speicherbelegung/Memory-Mapping.....	29
1.2.3 Interner Aufbau des STM32F446.....	33
1.2.4 Bussysteme des STM32F446	36
2 CMSIS und MCAL	39
2.1 CMSIS-Bibliothek beschaffen und erstellen	39
2.2 Aufräumarbeiten	40
2.3 Erstellen der Bibliothek.....	43
2.4 Fertigstellen der CMSIS-Bibliothek.....	45
2.4.1 Build-Variable für das CMSIS-Verzeichnis	45
2.4.2 Einstellen von Suchpfaden	46
2.4.3 Abschlussarbeiten an der CMSIS-Bibliothek.....	47
2.4.4 Verwenden von CMSIS in eigenen Projekten	48
2.4.5 Bezeichnungstechniken in CMSIS von STM.....	51
2.5 Der Bootvorgang	53
2.5.1 Der Reset-Handler	54
2.5.2 SystemInit.....	56
2.5.3 Die Erzeugung des Taktsignals.....	57
2.5.4 Linkerscripts.....	59

2.6	Aufbau der Headerdatei(en)	62
2.6.1	Orientierung in stm32f446xx.h	63
2.7	MCAL	66
3	Embedded C vs. Standard C	71
4	RCC, SYSCFG und SCB	77
4.1	Reset and Clock Control (RCC)	77
4.1.1	Reset	77
4.1.2	Clock Control	78
4.1.3	Ein erstes einfaches Projekt	79
4.1.4	Weitere wichtige Aufgaben des RCC	81
4.2	System Configuration Controller (SYSCFG)	85
4.3	System Control Block (SCB)	86

Teil II	Kernkomponenten der STM32F4xx-Mikrocontroller	87
----------------	--	-----------

5	GPIO: General Purpose Input/Output	91
5.1	Features und Grenzdaten	92
5.2	GPIO-Register	93
5.3	Zwei einfache Beispiele	102
5.3.1	Rechtecksignal mit dem ODR-Register	102
5.3.2	Rechtecksignal mit BSRR	103
6	Polling, Interrupts und Exceptions	107
6.1	Allgemeines zu Polling und Interrupts	108
6.1.1	Polling	108
6.1.2	Interrupts	109
6.2	Interrupts beherrschen	110
6.2.1	Maskierbare und nicht-maskierbare Interrupts	110
6.2.2	Globale Interrupts	112
6.2.3	Der Nested Vector Interrupt Controller NVIC	112
6.3	Externe Interrupts	118
6.3.1	External Interrupt/Event Controller (EXTI)	118
6.3.2	Beispiel zu externen Interrupts	123
6.3.3	Erkennung mehrerer externer Interrupts	128
6.4	Exceptions	133

7	Alternative GPIO-Funktionen	135
7.1	Wiederholung	135
7.2	Aktivieren alternativer Funktionen	137
7.3	Auswahl alternativer Funktionen	137
8	System Tick Timer	141
8.1	Verwendung des SysTick-Timers	143
8.2	Steuerung mehrerer Funktionen	146
8.3	Steuerung mehrerer Funktionen: Ein verbesserter Ansatz	152
8.4	Die MCAL-Version des Projekts Kap08-SysTick-04	162
9	Timer/Counter (SysTick und GP-Timer)	165
9.1	Der SysTick-Timer	168
9.1.1	Das SysTick Control and Status Register (CTRL)	169
9.1.2	Das SysTick Value Register (VAL)	169
9.1.3	Das SysTick Load Register (LOAD)	169
9.1.4	Das SysTick Calibration Register (CALIB)	169
9.2	Allgemeines zu Timern und Countern	171
9.3	Basic Timer TIM6 und TIM7	173
9.3.1	TIM6/TIM7 Control Register 1/2 (TIMx-CR1 und TIMxCR2)	174
9.3.2	TIM6/TIM7 DMA/Interrupt Enable Register (TIMx_DIER)	175
9.3.3	TIM6/TIM7 Status Register (TIMx_SR)	176
9.3.4	TIM6/7 Event Generation Register (TIMx_EGR)	176
9.3.5	TIM6/TIM7 Counter (TIMx_CNT)	176
9.3.6	TIM6/TIM7 Prescaler (TIMx_PSC)	176
9.3.7	TIM6/TIM7 Auto-Reload Register (TIMx_ARR)	177
9.3.8	Beispiel mit Basic Timer TIM6	177
9.4	Prescaler (Vorteiler) der Busse	181
10	General-Purpose Timer (GP-Timer)	185
10.1	GP-Timer, Teil 1: TIM9 bis TIM14	186
10.1.1	Register der GP-Timer TIM9 bis TIM14	187
10.1.2	Beispiel 1 zum Einsatz von TIM12	192
10.1.3	Beispiel 2 zum Einsatz von TIM12	197
10.2	GP-Timer, Teil 2: TIM2 bis TIM5	200
10.2.1	Einschub: Pulsweitenmodulation (PWM)	201
10.2.2	Beispiel: Dimmen einer LED mittels PWM	207

11	Advanced-Control Timer	211
11.1	Neue Register der Advanced-Control Timer	212
11.2	Einschub: Schalten induktiver Lasten	217
11.3	Beispiel zur Totzeitgenerierung mit dem STM32F446	219
11.3.1	Das Projekt Kap11-ACTIM-DeadTime.	219
12	Digital-Analog-Konverter	223
12.1	Technische Verfahren der D/A-Wandlung	223
12.1.1	Die Parallelwandlung	224
12.1.2	Das Zählverfahren	225
12.1.3	Das R-2R-Verfahren	225
12.2	DACs in der STM32F4xx-Familie.	226
12.2.1	Datenhalteregister.	227
12.2.2	Datenformate	228
12.3	Die Register des DAC	229
12.4	Ein einfaches Anwendungsbeispiel	231
12.5	Tipps für eigene Anwendungen	235
13	Analog-Digital-Wandlung	237
13.1	ADCs in der STM-Familie.	238
13.2	Register in den STM-ADCs.	242
13.3	Anwendungsbeispiel	247

Teil III Serielle Schnittstellen **251**

14	Serielle Kommunikation	253
14.1	Grundlegende Begriffe	254
14.1.1	Kommunikationsrichtungen	254
14.1.2	Aufbau der Daten	255
14.1.3	Datenpegel.	257
14.1.4	Übertragungsgeschwindigkeit.	257
14.1.5	Übertragungsprotokolle	258
14.1.6	Asynchrone vs. synchrone Datenübertragung	259
14.2	Ausführungsformen einfacher RS-232-Schnittstellen	259
15	UARTs und USARTs	261
15.1	Was sind UARTs und USARTs?	262
15.1.1	Die USART-/UART-Register	263
15.1.2	Empfangen und Senden von Daten	268

16	Inter-Integrated Circuit I²C	279
16.1	Die ursprüngliche Idee hinter I ² C	279
16.2	Prinzipieller Aufbau einer I ² C-Schaltung	280
16.3	Betriebsarten/Protokoll von I ² C	283
16.3.1	Vier Betriebsarten	283
16.3.2	Das I ² C-Protokoll	284
16.4	I ² C in der STM32F4xx-Familie	287
16.5	Ein Beispiel mit dem PCF8574	294
16.5.1	Der PCF8574	296
16.5.2	Das Programmlisting	297
16.6	Neue MCAL-Funktionen	304
16.7	Weitere Beispiele	305
16.7.1	Daten lesen von I ² C-Komponenten	306
16.7.2	Anmerkungen zu den Beispielprojekten	309
17	Serial Peripheral Interface SPI	313
17.1	Datentransfer in SPI-Interfaces	316
17.1.1	CPHA = 1	317
17.1.2	CPHA = 0	317
17.1.3	Anwendung von SPI	318
17.2	SPI-Register der STM32F4xx-Familie	318
17.2.1	SPI Control Register 1 und 2 (SPI_CR1, SPI_CR2)	319
17.2.2	Das SPI Status Register (SPI_SR)	321
17.2.3	Die Checksummen-Register (CRC)	322
17.2.4	Das SPI Data Register (SPI_DR)	324
17.3	Ein einfaches Beispiel mit dem MAX7219	324
17.3.1	Kurzbeschreibung des MAX7219	332
17.4	Eine kleine Übung	334

Teil IV	Weitere Komponenten	337
----------------	----------------------------	------------

18	Direct Memory Access (DMA)	339
18.1	Funktionsweise	340
18.2	DMAC(s) in der STM32F4xx-Familie	341
18.2.1	Register der DMACs in der STM32F4xx-Familie	344
18.3	Beispiel: Memory → USART2 → PC mit DMA	349
18.3.1	Erläuterung der Funktionsweise	353
18.4	Beispiel: USART2 → PC mit DMA → USART2	355
18.4.1	»Probleme« von DMA	361

19	Watchdog	365
19.1	Independent Watchdog (IWDG).....	365
19.2	Window Watchdog (WWDG)	367
19.2.1	Funktionsweise.....	368
19.3	Debuggen und Watchdogs	370
Teil V	Anhang	371
A	Internetadressen und Literaturnachweise	373
A.1	Literaturnachweise.....	373
A.2	Infos zur STM32F4xx-Familie	374
A.3	Programmierung in C.....	374
A.4	Tastaturkürzel von STM32CubeIDE	374
A.5	Internet-Tutorials.....	374
A.6	Support.....	375
B	Takteinstellung mit STM32CubeMX	377
B.1	Möglichkeit 1	377
B.2	Möglichkeit 2	381
C	Einführung in das Debuggen	385
C.1	Debugger einrichten	386
C.2	Variablen »beobachten«	390
C.3	Anzeige von Prozessorregistern	392
C.4	Anzeige von SRAM-Inhalten	393
C.5	Vorsicht!	393
D	Funktionen der MCAL-Bibliothek	397
D.1	General-Purpose Input/Output GPIO	397
D.2	Externe Interrupts EXTI	398
D.3	SysTick-Timer	398
D.4	Basic-/GP-/Advanced-Control Timer	399
D.5	UART/USART.....	400
D.6	Systemfunktionen	401
D.7	Inter-Integrated Circuit I ² C.....	402
D.8	Serial Peripheral Interface SPI	403
D.9	Direct Memory Access DMA.....	403
	Stichwortverzeichnis	405