

Inhaltsverzeichnis

	Vorwort	11
1.	Einführung in die Mikrocontrollertechnik	13
1.1	Einsatzgebiete für Mikrocontroller	14
1.2	Was ist ein Mikrocontroller?	15
1.2.1	Mikrocontroller-Speicher	16
1.2.2	RISC-Architektur	17
1.3	Der passende Mikrocontroller-Typ	18
1.3.1	Funktionsumfang Befehlssatz	19
1.3.2	Geschwindigkeit	19
1.3.3	Funktionsumfang integrierter Hardware	20
1.3.4	Preis und Beschaffbarkeit	21
1.3.5	Bauform	22
1.3.6	Speichergröße	22
1.3.7	Qualität der Dokumentation	23
1.3.8	Akzeptanz bei anderen Nutzern	24
1.3.9	Verfügbarkeit an Entwicklungstools	25
1.4	Entscheidungshilfe Atmel AVR	26
2.	Die Entwicklungsumgebung	29
2.1	Spannungsversorgung	30
2.1.1	Standard-Spannungsversorgung	30
2.1.2	Spannungsversorgung im Fahrzeug	31
2.1.3	Brownout-Erkennung	33
2.2	Serielle und parallele Schnittstelle	34
2.3	Programmieradapter	35
2.3.1	ISP-Anschluss	36
2.3.2	Low-Cost Programmieradapter	38
2.3.3	Minimalistischer serieller RS232-Adapter	39
2.3.4	STK200-kompatibler Parallelportprogrammieradapter	39
2.3.5	USB-Programmieradapter	41
2.4	C-Compiler WinAVR	42
2.4.1	WinAVR installieren	42
2.4.2	Der Editor Programmers Notepad	44
2.5	Programmiersoftware	46
2.5.1	PonyProg	46
2.5.2	avrdude	47

2.5.3	AVR8 Burn-O-Mat	48
2.6	Brennen und Debugging per JTAG	49
2.6.1	JTAG-Interface	49
2.6.2	AVR Studio	52
2.6.3	JTAG-Interface programmieren	52
2.6.4	JTAG-Testprojekt	54
3.	Die Experimentierplatine	59
3.1	Aufbau des Evaluations-Board	60
3.1.1	Einfach und vielseitig	60
3.1.2	Alles an Bord	63
3.2	Standard Fuse Bits	63
4.	Erste Schritte mit dem μC	65
4.1	μ C Minimalistisch	65
4.1.1	Exkurs: Abblockkondensator	66
4.1.2	Exkurs: Taktgenerator	68
4.2	Das erste Programm compilieren	71
4.2.1	Exkurs: Compiler und makefile	73
4.3	Den Mikrocontroller programmieren	76
4.3.1	ISP-Adapter	77
4.3.2	AVR8 Burn-O-Mat	78
4.3.3	avrdude aus dem Editor heraus aufrufen	78
4.4	Konfiguration des AVR8s mittels Fuse Bits und Security Bits	80
4.4.1	Die wichtigsten Fuse Bits	80
4.4.2	Der Fuse Bit Notfallplan	85
4.4.3	Mit Lock Bits den Prozessorinhalt schützen	86
4.4.4	Fuse und Lock Bits mit avrdude	88
5.	I/O-Grundlagen	91
5.1	Standardbibliotheken kennen lernen und LEDs ansteuern	92
5.1.1	LEDs am Mikrocontroller anschließen	93
5.1.2	Register-Konstanten aus der I/O-Standardbibliothek	94
5.1.3	Lesbarkeit und Anpassung des Codes erhöhen	96
5.1.4	Ein (zweiter) Blick auf die AVR Libc	97
5.1.5	Die Programmausführung verzögern	97
5.1.6	Ganzzahlige (Integer) Datentypen	99
5.1.7	Bitmanipulation und Datenausgabe	100
5.2	7-Segmentanzeigen und alphanumerische LEDs	106
5.2.1	Gemeinsame Anode oder Kathode	107
5.2.2	7-Segmentanzeige ansteuern	107
5.2.3	Alphanumerische LEDs	110
5.3	Größere Lasten betreiben	114
5.3.1	Der Transistor für kleine Lasten	114
5.3.2	Power-MOSFETs für hohe Belastungen	115
5.3.3	Alt aber noch immer praktisch: Relais	116
5.3.4	Galvanische Trennung mit Optokopplern	118

5.4	Signaleingänge	119
5.4.1	Pull-Up oder Pull-Down	119
5.4.2	Eingangstatus im Programm abfragen	120
5.4.3	Tasten entprellen	122
5.5	Ressourcenschonung mit Multiplexbetrieb	123
5.5.1	7-Segmentdisplay multiplexen	123
5.5.2	Zahlenzerlegung	126
5.5.3	Multiplexen mit Timer	127
5.5.4	Exkurs: Schieberegister	129
5.5.5	5x7 Matrix mit Schieberegister	130
5.5.6	Textausgabe auf der Punktmatrix	135
6.	Flüssigkristalldisplays	137
6.1	HD44780-kompatibel	137
6.1.1	LCDs elektrisch anschließen	138
6.1.2	Speicherabbild	139
6.1.3	Befehlsatz HD44780	140
6.2	LC-Displaybetrieb	140
6.2.1	Initialisierung 4-Bit-Modus	141
6.2.2	Displaykonfiguration	143
6.2.3	Ausgabe von Zeichen durch Speicheradressierung	144
6.2.4	Zahlen ausgeben	144
6.3	Eigene Zeichen definieren	148
6.3.1	Zeichendefinition	148
6.3.2	Extra große Zahlen	149
7.	Serielle Datenübertragung	151
7.1	RS232-Schnittstelle	151
7.2	Register zur Konfiguration des USART	153
7.2.1	Software-UART	154
7.3	Daten senden und empfangen	155
7.3.1	Hände schütteln für eine bessere Verständigung	156
8.	Analoge Ein- und Ausgabe	159
8.1	Auflösung und Eingangsbeschaltung des ADC	159
8.1.1	Spannungsteiler	160
8.1.2	Wertberechnung	161
8.1.3	Referenzspannung	161
8.2	A/D-Wandlung durchführen	161
8.2.1	Konfiguration des ADC	162
8.2.2	Tipps für die Praxis	163
8.3	Temperaturmessung	164
8.4	Zufallszahlengenerator	165
8.5	Digital-Analog-Wandler	166
8.5.1	Digitaler Funktionsgenerator	168

9.	Programmablaufsteuerung mit Interrupts	173
9.1	Quellen für Interrupts	174
9.1.1	Interruptbehandlung	175
9.1.2	Interrupts aktivieren	176
9.2	Externe Unterbrechungsanforderungen verarbeiten	177
9.2.1	Exkurs: volatile	179
9.2.2	Atomare Datenzugriffe	180
9.3	Ein Wachhund gegen Programmfehler	181
9.3.1	Den Watchdog nutzen	182
9.4	Batteriebetrieb und Strom sparen im Sleep Modus	185
9.4.1	Batteriebetrieb	186
9.4.2	Pufferbatterie	187
9.4.3	Sleep Modi: vom Nickerchen bis zum Tiefschlaf	189
9.4.4	Sleep Mode Beispiel	192
10.	Timer/Counter	197
10.1	Arbeitsweise eines Timers	198
10.1.1	Bitbreite und Interrupt des Timers	198
10.1.2	Taktquellen und Vorteiler	198
10.2	Die Timer des ATmega16	199
10.2.1	Überlauf mit 8-Bit-Timer 0	199
10.2.2	Timer 0 mit Voreinstellung	201
10.2.3	Timer 0 mit Vergleichswert	202
10.2.4	Der CTC-Modus des Timer 0	204
10.2.5	Timer 0 als Signalgenerator	205
10.2.6	Externe Impulse mit Timer 0 zählen	206
10.2.7	Tonausgabe mit dem 8-Bit-Timer 2	208
10.2.8	Stopuhr mit dem 16-Bit-Timer 1	209
10.3	Pulsweitenmodulation	212
10.3.1	PWM per Software	213
10.3.2	Fast PWM mit Timer 0	216
10.3.3	Timer 1: Fast PWM mit beliebiger Frequenz	216
11.	Speicherzugriffe	219
11.1	Zugriff auf den Programmspeicher (Flash)	219
11.1.1	String-Array im Flash ablegen	221
11.2	Zugriff aufs EEPROM	222
11.2.1	EEPROM-Abbilddatei	225
11.2.2	Für Fließkommazahlen auf Speicherbereiche zugreifen	226
11.3	Bootloader	227
11.3.1	MC für die Massenfertigung programmieren	227
11.3.2	Der Speicher für den Bootloader	229
11.3.3	Softwareupdate per Bootloader	230
11.3.4	Resetvektor und Interruptvektor	231
11.3.5	Das erste Programm im Bootloader	233
11.3.6	Eine Applikation und ein Bootloader	236
11.3.7	Zwischen Applikation und Bootloader wechseln	238

11.3.8	Ein Bootloader für Programmupdates	241
12.	Serieller Datenbus I²C (TWI) und SPI	245
12.1	Two-Wire Interface (TWI) I ² C	245
12.1.1	Funktionsprinzip des I ² C-Bus	246
12.1.2	Adressierung der Slaves	247
12.1.3	I ² C-Busprotokoll	247
12.2	EEPROM per TWI ansteuern	248
12.2.1	Standard-Speichertyp 24Cxx	248
12.2.2	TWI am ATmega	251
12.3	Das Serial Peripheral Interface (SPI)	258
12.3.1	SPI zwischen ATmega16 und ATmega8	259
13.	Beispielprojekte	263
13.1	Funkuhr mit DCF77-Signal	263
13.1.1	Aufbau des Zeitsignals	263
13.1.2	Signalform des Zeitsignals	265
13.1.3	Empfangsmodul	265
13.1.4	Beispielapplikation	266
13.2	Datum und Zeit ganz genau: Real Time Clock	267
13.2.1	Ein IC kontrolliert alle Datums- und Uhrzeitfunktionen	267
13.2.2	Maxim DS1302 RTC	268
13.2.3	Der BCD-Code	271
13.2.4	Ein Blick auf die Anwendung	272
13.3	Globale Navigationssatellitensysteme	274
13.3.1	GPS-Empfänger	274
13.3.2	GPS-Empfänger anschließen	275
13.3.3	NMEA-Daten auswerten	276
13.3.4	GPS bekommt Konkurrenz: GLONASS	277
13.3.5	GLONASS und GPS empfangen und auswerten	279
13.3.6	Software-UART für den Datenempfang	281
13.3.7	VT100-Terminalcodes für formatierte Ausgaben	284
13.4	PC-Tastatur als Eingabegerät	286
13.4.1	Protokoll der Tastatur	287
13.4.2	Tastatur Beispielanwendung	288
13.5	Datenübermittlung und Fernwirken per Handy	288
13.5.1	Kostenlos Schalten	289
13.5.2	Datenverbindung zum Mobiltelefon	290
13.5.3	GSM AT-Kommandos	291
13.5.4	Kurznachrichten per PDU absetzen	292
13.6	Datenaustausch via Bluetooth	295
13.6.1	Der BTM-220/BTM-222	296
13.6.2	Mit Bluetooth in der Ferne steuern	298
13.7	(Fast) Unendlich viele-I/O Ports mit I ² C	299
13.7.1	Der Port-Expander PCF8574	299
13.7.2	Open Collector/Open Drain Ausgänge	300
13.7.3	Ohne Datenrichtungsregister: Ausgänge werden Eingänge	301

13.8	Ein-Draht-Temperaturmessung	301
13.8.1	1-Wire-Protokoll und parasitäre Versorgung	302
13.8.2	Aufbau des LCD Thermometers	303
13.8.3	Assembler für genaues Timing	303
13.8.4	Temperaturwert vom DS18S20 auslesen	306
13.9	Entfernungsmessung mit Ultraschall	307
13.9.1	Kurze Laufzeiten messen	308
13.10	USB und virtuelle COM-Ports	311
13.10.1	USB-Host oder -Slave	311
13.10.2	USB-Controller FT232	313
13.10.3	USB im Handumdrehen nachrüsten	315
13.11	Farbgrafik-LCD	316
13.11.1	Probleme und Lösungen beim Einsatz von GLCDs	316
13.11.2	1,8 Zoll Beispieldisplay mit 160 x 128 Pixel	317
13.11.3	Betrieb des Displays	319
13.11.4	Ein kurzer Blick auf die Software	319
13.12	Richtig viel Speicherplatz auf SD-Karten	321
13.12.1	Einfacher Anschluss dank SPI	322
13.12.2	Das 3,3-V-Problem: Level-Shifter	323
13.12.3	Eine einfache Library für Speicherkarten	325
14.	Anhang: Überblick Register ATmega16	329
	Stichwortverzeichnis	333