

Mikrocontroller-Grundlagen

Die Beschreibung der Speicherorganisation sowie Grundsätzliches zu den Zahlensystemen lesen Sie im vierten Teil dieser Artikelserie.

Teil 4

2.15 Speicherorganisation

Der 8031/51 verfügt über getrennte Adreßbereiche für den Programm- und Datenspeicher. Der Programmspeicher, auch Codespeicher genannt, kann bis zu 64 kByte umfassen. Beim 8051/52 können sich die untersten 4 bzw. 8 kByte, wie in Abbildung 30 gezeigt, auf dem Chip befinden. Hierzu ist dann der \overline{EA} -Anschlußpin auf logisch „1“-Pegel zu legen. Erfolgt ein Zugriff auf den Programmspeicher oberhalb dieser Grenze, wird trotz des Anliegens eines High-Pegels an dem \overline{EA} -Anschlußpin über die \overline{PSEN} -Steuerleitung auf den externen Programmspeicher zugegriffen.

Der 8031/32 besitzt keinen internen Programmspeicher und kann auf bis zu 64 k

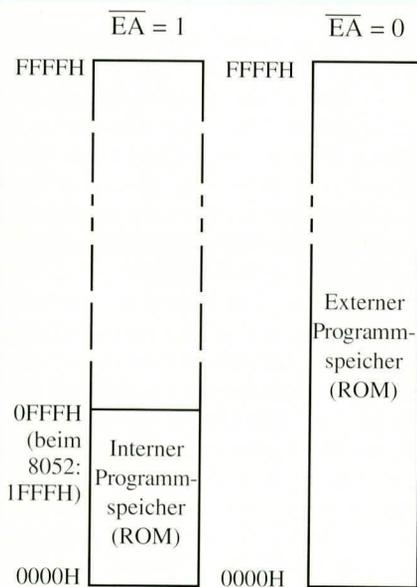
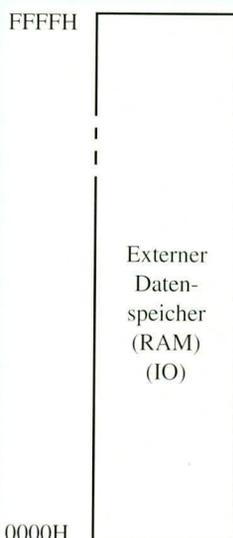


Bild 30 (oben): Der Programmspeicherbereich der MCS-51-Familie mit internem und externem ROM

Bild 31 (rechts): Der max. 64 kByte große externe Datenbereich der MCS-51-Familie



externen ROM-Speicher zugreifen. Voraussetzung ist, daß der \overline{EA} -Steueranschluß auf Low-Pegel liegt.

Der externe Datenspeicher, der durch die Prozessor-Steuerleitungen RD und WR angesprochen wird, hat ebenfalls, wie in Abbildung 31 gezeigt, eine maximale Größe von 64 kByte.

Auf dem Chip ist ein interner RAM-Bereich von 128 Byte integriert. Beim 8032/52 kommen weitere 128 Byte hinzu, so daß hier insgesamt 256 Byte zur Verfügung stehen.

Ferner gibt es eine Reihe spezieller Funktions-Register (Special-Function-Register = SFR).

Das interne RAM und die SFR werden über spezielle Befehle angesprochen und benötigen daher für die Adressierung keine speziellen Steuerleitungen.

2.15.1 Programmspeicher

Nach einem Reset startet der Mikrocontroller mit der Abarbeitung des ersten Befehls, der an der Adresse 0000H im Programmspeicher steht.

Mit der Abarbeitung jedes Befehls wird der Programm-Counter (PC, Befehlszähler) um 1 erhöht und der folgende Befehl dem Programmspeicher entnommen.

Der Mikrocontroller läßt Programmunterbrechungen durch unterschiedliche Ereignisse, wie Überlauf eines der internen Zähler, Aktivierung bestimmter externer Interrupt-Leitungen oder auch das Empfangen oder Senden eines seriellen Datenwortes, zu. Zu diesem Zweck wird bei Aktivierung und entsprechender Freigabe zu einer der fest definierten Adressen gemäß Abbildung 32 verzweigt. Hierzu ist jedem Unterbrechungs (Interrupt)-Ereignis eine feste ROM-Einsprungsadresse zugeordnet. Die Sprungadressen werden auch vielfach Interrupt-Vektoren genannt. In den meisten Fällen ist hier ein Sprung zu der eigentlichen Interrupt-Service-Routine vorgesehen.

Im Programmspeicher können neben dem eigentlichen Programmcode noch Tabellen oder Texte hinterlegt sein, die über spezielle Befehle dem Programm zur Verfügung gestellt werden können.

2.15.2 Externer Datenspeicher

Der zweite, vom Controller adressierbare Speicherbereich ist der externe Datenspeicher, der sowohl aus einem RAM als auch aus einem ROM bestehen kann. Im Unterschied zum Programmspeicher-

ROM, welches über die \overline{PSEN} -Steuerleitung adressiert wird, erfolgt ein Schreib- bzw. Lesezugriff auf den externen Datenspeicher über die WR- bzw. RD-Steuerleitung, die damit gleichzeitig die Richtung des Datenflusses bestimmt.

In diesem In/Out- (I/O) Bereich werden üblicherweise externe Datenspeicher wie RAM, ROM oder EEPROM untergebracht. Weiterhin lassen sich auch hier Standard-I/O-Bausteine wie Timer, Counter, Parallel-In/Out oder andere integrierte Bausteine mit einem 8 Bit-Datenbus anschließen.

Die maximal 64kByte des externen Datenspeichers sind nur indirekt über den Data-Pointer (DPTR) adressierbar.

002BH	Timer 2 (nur 8032/52)
0023H	Serieller Port Interrupt
001BH	Timer 1
0013H	Ext. Interrupt 1
000BH	Timer 0
0003H	Ext. Interrupt 0
0000H	Reset

Bild 32 zeigt die Interrupt-Einsprungsadressen der MCS-51-Familie

2.15.3 Interner Datenspeicher

Der interne Datenspeicher des 8031/51 hat eine Größe von 128 Bytes. Abbildung 33 zeigt die Aufteilung dieses Speicherbereiches. Die untere 128 Bytes stehen für unterschiedlichste Verwendungszwecke zur Verfügung, wobei jede Speicherzelle direkt oder auch indirekt (über Zeiger) erreichbar ist. Der 8032/52 besitzt darüber hinaus noch weitere 128 Byte internen Speicher, der ebenfalls zur freien Verfügung steht. Allerdings ist dieser Speicherbereich nur indirekt erreichbar.

Über die Bits 3 und 4 des Programmstatus-Wortes (PSW) ist eine der 4 Registerbänke selektiert. Auf die Speicherzel-

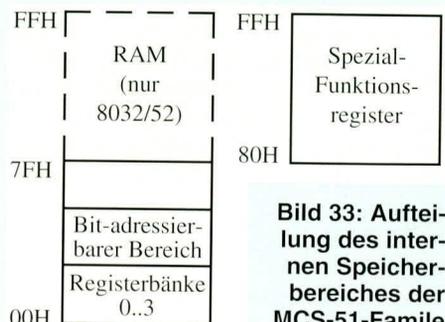


Bild 33: Aufteilung des internen Speicherbereiches der MCS-51-Familie

Byte-Adressen (hex)	Bit-Adressen (hex)	Byte-Adressen (dezimal)
(MSB)		(LSB)
7FH		127
30H		48
2FH	7F 7E 7D 7C 7B 7A 79 78	47
2EH	77 76 75 74 73 72 71 70	46
2DH	6F 6E 6D 6C 6B 6A 69 68	45
2CH	67 66 65 64 63 62 61 60	44
2BH	5F 5E 5D 5C 5B 5A 59 58	43
2AH	57 56 55 54 53 52 51 50	42
29H	4F 4E 4D 4C 4B 4A 49 48	41
28H	47 46 45 44 43 42 41 40	40
27H	3F 3E 3D 3C 3B 3A 39 38	39
26H	37 36 35 34 33 32 31 30	38
25H	2F 2E 2D 2C 2B 2A 29 28	37
24H	27 26 25 24 23 22 21 20	36
23H	1F 1E 1D 1C 1B 1A 19 18	35
22H	17 16 15 14 13 12 11 10	34
21H	0F 0E 0D 0C 0B 0A 09 08	33
20H	07 06 05 04 03 02 01 00	32
1FH	R7 Bank 3	13
18H	R0	24
17H	R7 Bank 2	23
10H	R0	16
0FH	R7 Bank 1	15
08H	R0	8
07H	R7 Bank 0	7
00H	R0	0

Bild 34: Vorbelegung des internen Speichers

len der aktuellen Registerbank sind besondere Adressierungsarten anwendbar, die wir später noch beschreiben. Der Vorteil dieser Registerbanktechnik ist die schnelle Umschaltung auf eine andere Registerbank, beispielsweise in Interrupt-Service-Routinen (Programmunterbrechung). Zu diesem Zweck müssen dann nicht alle Register einzeln gesichert werden.

Die unteren 128 Byte des internen RAMs sind direkt oder indirekt (über R 0 und R 1 der aktuellen Registerbank) adressierbar, während beim 8032/52 die oberen 128 Byte des internen RAMs nur indirekt (über R 0 und R 1 der aktuellen Registerbank) und die SFR nur direkt adressierbar sind.

Abbildung 34 zeigt in detaillierter Form die Vorbelegung der ersten 128 Byte des internen Speichers. Die Speicherzellen 0 bis 7 sind für die Registerbank 0, die Speicherzellen 8 bis 15 für die Bank 1, die Speicherzellen 16 bis 23 für die Bank 2 und der Adreßbereich 24 bis 31 für die Registerbank 3 vorgesehen. Über spezielle As-

sembler-Befehle, die die Direktadressierung der Register R 0 bis R 7 ermöglichen, ist eine übersichtliche Programmstruktur erreichbar. Die Selektierung (Auswahl der Registerbank) kann über 2 Bits des Programm-Statuswortes (PSW) vorgenommen werden, worauf wir im Verlauf dieser Artikelserie noch näher eingehen.

Der Speicherbereich 20H bis 2FH ist neben der direkten Adressierung noch Bit-adressierbar, d. h. jedes Bit in diesem Bereich ist einzeln adressierbar. Hierdurch sind direkte Bitmanipulationen möglich.

2.15.4 Spezielle Funktionsregister

Die Architektur der MCS51-Familie erlaubt es, die speziellen Funktionsregister über eine besondere Adressierungsart anzusprechen. Hierzu werden die Register quasi als prozessorinterne Speicherzellen über eine direkte Adressierung angesprochen. Tabelle 5 zeigt die Zuordnung der einzelnen Register zu deren Adressen. Auffällig ist hier, daß der Adreßbereich sich von 80H bis F0H erstreckt. Die Ursache liegt darin, daß die gleichen Befehle, die für den Zugriff auf diese Spezialregister zuständig sind, auch den internen RAM-Bereich von 0 bis 7FH beim 8031/51 adressieren.

Einige der Spezialregister sind nicht nur komplett über einen 8 Bit-Zugriff adressierbar, sondern zusätzlich noch über die direkte Bit-Adressierung erreichbar. Dies betrifft alle Register die an Adressen liegen, die ohne Rest durch 8 teilbar sind, d. h. alle Adressen, die in hexadezimaler Schreibweise mit 0 oder 8 enden. Daraus ergibt sich die Möglichkeit, jedes einzelne Bit dieser Register, ohne Umwege über die direkte Byte-Adressierung auf direktem Wege anzusprechen, d. h. zu setzen, zu

löschen oder auch auszulesen.

Tabelle 6 zeigt die Zuordnung der Bit-Adressen und ihre Bedeutung, auf die wir im Verlaufe dieser Artikelserie noch detaillierter zu sprechen kommen.

Der obere interne RAM-Bereich und der interne SFR-Bereich haben die gleichen Adressen. Nur die Adressierungsart bestimmt, welche Speicherzellen angesprochen werden.

2.16 Zahlensysteme

Um die einzelnen Befehle und Parameter besser verstehen zu können, ist die Kenntnis der Zahlensysteme hilfreich. Die meistverwendeten sind die Dezimal-, Dual- und Hexadezimal-Systeme.

2.16.1 Dezimalzahlen-System

Das im Alltag gebräuchliche dezimale Zahlensystem hat einen Zahlenvorrat von 10 Ziffern (0 bis 9), woraus sich der Name „Dezimal-System“ ableitet. Eine Zahl größer als 9 wird durch die Kombination von 2 oder mehreren Ziffern dargestellt. Die Zahl 123 setzt sich beispielsweise zusammen aus

$$1 \cdot 100 + 2 \cdot 10 + 3 \cdot 1$$

oder auch

$$1 \cdot 10^2 + 2 \cdot 10^1 + 3 \cdot 10^0.$$

Der Wert einer Ziffer hängt somit von der Stelle ab, an der sie steht. Man spricht von Einer-, Zehner-, Hunderter-Stelle usw. Die Wertigkeit jeder weiteren Ziffer erhöht sich im Dezimal-System um den Faktor 10.

2.16.2 Binärzahlen-System

Ein Mikrocontroller kann nicht mit Zahlen von 1 bis 10 umgehen, sondern lediglich logische Zustände („high“ und „low“) verarbeiten.

Es gibt also 2 definierte Zustände, die auch mit „1“ (high) und „0“ (low) bezeichnet werden. Wie auch beim Dezimal-System können auch im Binär-System durch Aneinanderreihung mehrerer Ziffern beliebige Zahlen dargestellt werden. Lediglich die Wertigkeit der einzelnen Stellen unterscheidet sich zwischen den Zahlensystemen. Die Gewichtungsfaktoren im Binär-System entsprechen jeweils dem Faktor 2. Die Binärzahl 1111011 setzt sich beispielsweise

Tabelle 5: Zuordnung der Adressen der SFR und ihre Bedeutung

Adresse	Bit adressierbar	Bezeichnung	Erläuterung
F0H	Ja	B	Hilfsregister
E0H	Ja	ACC	Akkumulator
D0H	Ja	PSW	Programm Status Wort (Flags)
CDH	Nein	TH2	Timer 2 High-Byte nur 8032/52
CCH	Nein	TL2	Timer 2 Low-Byte nur 8032/52
CBH	Nein	RC AP2H	Capture Register High-Byte nur 8032/52
CAH	Nein	RC AP2L	Capture Register Low-Byte nur 8032/52
C8H	Ja	T2 CON	Timer 2 Control Register
B8H	Ja	IP	Interrupt Priority Register
B0H	Ja	P3	Port 3
A8H	Ja	IE	Interrupt Enable Register
A0H	Ja	P2	Port 2
99H	Nein	SBUF	Serial Buffer
98H	Ja	SCON	Serial Port Control Register
90H	Ja	P1	Port 1
8DH	Nein	TH1	Timer 1 High Byte
8CH	Nein	TH0	Timer 0 High-Byte
8BH	Nein	TL1	Timer 1 Low-Byte
8AH	Nein	TL0	Timer 0 Low-Byte
89H	Nein	TMOD	Timer Modus Register
88H	Ja	TCON	Timer Control Register
87H	Nein	PCON	Power Control Register
83H	Nein	DPH	Data Pointer MSB
82H	Nein	DPL	Data Pointer LSB
81H	Nein	SP	Stack Pointer
80H	Ja	P0	Port 0

Tabelle 6: Zuordnung der Bit-adressierbaren Speicherzellen u. ihre Bedeutung

Bezeichnung der Byte-adresse	Bitadresse (hex)							
	Bezeichnung bzw. Bedeutung der Bitadresse (* nur 8032/52)							
	FFH	FEH	FDH	FCH	FBH	FAH	F9H	F8H
B	F7H B.7	F6H B.6	F5H B.5	F4H B.4	F3H B.3	F2H B.2	F1H B.1	F0H B.0
	EFH	EEH	EDH	ECH	EBH	EAH	E9H	E8H
ACC	E7H ACC.7	E6H ACC.6	E5H ACC.5	E4H ACC.4	E3H ACC.3	E2H ACC.2	E1H ACC.2	E0H ACC.1
	DFH	DEH	DDH	DCH	DBH	DAH	D9H	D8H
PSW	D7H CY	D6H AC	D5H F0	D4H RS1	D3H RS0	D2H OV	D1H -	D0H P
T2CON	CFH TF2*	CEH EXF2*	CDH RCLK*	CCH TCLK*	CBH EXEN2*	CAH TR2*	C9H C/T2*	C8H CP/RL2*
	C7H	C6H	C5H	C4H	C3H	C2H	C1H	C0H
IP	BFH	BEH	BDH PT2*	BCH PS	BBH PT1	BAH PX1	B9H PT0	B8H PX0
P3	B7H P3.7/RD	B6H P3.6/WR	B5H P3.5/T1	B4H P3.4/T0	B3H P3.3/INT1	B2H P3.2/INT0	B1H P3.1/TxD	B0H P3.0/RxD
IE	AFH EA	AEH	ADH ET2*	ACH ES	ABH ET1	AAH EX1	A9H ET0	A8H EX0
P2	A7H P2.7	A6H P2.6	A5H P2.5	A4H P2.4	A3H P2.3	A2H P2.2	A1H P2.1	A0H P2.0
SCON	9FH SMO	9EH SM1	9DH SM2	9CH REN	9BH TB8	9AH RB8	99H TI	98H RI
P1	97H P1.7	96H P1.6	95H P1.5	94H P1.4	93H P1.3	92H P1.2	91H P1.1/T2EX*	90H P1.0/T2*
TCON	8FH TF1	8EH TR1	8DH TF0	8CH TR0	8BH IE1	8AH IT1	89H IE0	88H IT0
P0	87H P0.7	86H P0.6	85H P0.5	84H P0.4	83H P0.3	82H P0.2	81H P0.1	80H P0.0

zusammen aus
 $1 \cdot 64 + 1 \cdot 32 + 1 \cdot 16 + 1 \cdot 8 + 0 \cdot 4 + 1 \cdot 2 + 1 \cdot 1$
 oder
 $1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$

Der Dezimalwert dieser Binärzahl beträgt somit 123. Jede Stelle einer Binärzahl wird als Bit bezeichnet.

Gekennzeichnet werden Binärzahlen durch Anhängen des Buchstabens „B“ oder auch Voranstellen des Prozentzeichens „%“. Gelegentlich wird auch die Basis als tiefgestellte Zahl angefügt z. B. 1111011_2 .

Im Prinzip lassen sich auf diese Weise auch sehr große Zahlen darstellen. Bei Darstellung der Zahl 65.000 würden alleine schon 16 Stellen benötigt, was zur Un-

Tabelle 7: Zahlendarstellung im Dezimal-, Binär- und Hexadezimalsystem

Dezimal	Binär	Hex
0	0000B	0H
1	0001B	1H
2	0010B	2H
3	0011B	3H
4	0100B	4H
5	0101B	5H
6	0110B	6H
7	0111B	7H
8	1000B	8H
9	1001B	9H
10	1010B	AH
11	1011B	BH
12	1100B	CH
13	1101B	DH
14	1110B	EH
15	1111B	FH
16	10000B	10H
:	:	:
:	:	:

zusammen. Daraus ergibt sich ein erforderlicher Zeichenumfang von 16 Ziffern. Für die ersten 10 Ziffern werden die aus der Dezimalschreibweise gewohnten Zahlen 0 bis 9 verwendet. Für die weiteren Ziffern 10 bis 15 wurden die ersten 6 Buchstaben des Alphabets „A“ bis „F“ gewählt. Tabelle 7 zeigt die Zuordnung der 3 wichtigsten Zahlendarstellungsarten.

Beispielsweise entspricht die hexadezimale Ziffernfolge 7BH einer Wertigkeit von

$$7 \cdot 16 + B \cdot 1$$

oder

$$7 \cdot 16^1 + B \cdot 16^0 \quad B_H = 11_D$$

Gekennzeichnet werden Hexadezimalzahlen durch Anhängen des Buchstabens „H“ (Hex) oder auch durch Voranstellen des „\$“-Zeichens oder „0x“.

Neben den vorgestellten Zahlensystemen wird in Ausnahmefällen auch noch das Oktal-System, welches die Basis 8 als Bezugsgröße benutzt, verwendet. Dieses System kommt also mit dem Zahlenumfang von 0 bis 7 aus.

2.16.4 Zweierkomplement-Darstellung

Für viele Anwendungen benötigt man auch negative Zahlen, deren Darstellungsart auf vielfältige Weise möglich ist. Hierfür hat sich die Zweierkomplement-Darstellung durchgesetzt. Bei einer 8 Bit-Zahl muß der Zahlenbereich von 0 bis 255 für die Darstellung der positiven und negativen Zahlen halbiert werden. Hierzu verschiebt sich dann der Bereichsumfang auf -128 bis +127.

Die Darstellung des positiven Zahlenbereiches bleibt wie gewohnt erhalten, d. h.

übersichtlichkeit führt. Um dennoch diese Binärwerte übersichtlich zu machen und dabei prinzipiell in diesem Zahlensystem zu bleiben, wurde das Hexadezimal-System eingeführt.

2.16.3 Hexadezimalzahlen-System

Mikrocontroller arbeiten üblicherweise mit 8- oder auch 16 Bit-Zahlen. Aus diesem und dem obenangeführten Grund wurde das Hexadezimal-System (mit der Basis 16) eingeführt. Es faßt pro Ziffer jeweils 4 Bit

Dezimal	Binär	Erläuterung
123	= 0111 1011	positive Zahl
	1000 0100	Einerkomplement
	+ 1	
-123	= 1000 0101	Zweierkomplement

Bild 35: Umrechnung einer positiven Zahl in deren Zweierkomplement

für den Zahlenbereich 0 bis +127 ist das höchstwertigste Bit immer 0. Sobald nun aber die Zahl negativ ist, ist immer das höchstwertigste Bit gesetzt.

Um aus einer positiven Zahl das Zweierkomplement zu bilden, müssen zunächst die Bits einzeln invertiert und anschließend die Zahl 1, wie Abbildung 35 zeigt, addiert werden. Tabelle 8 zeigt die grundsätzliche Zuordnung zwischen Dezimal-, Binär-Zweierkomplement und Hexadezimal-Darstellung.

Für die Darstellung eines größeren Zahlenbereiches wird entsprechend die Breite (z. B. 16 Bit) erhöht. Das höchstwertigste Bit gibt dann die Polarität der Zahl an.

2.16.5 BCD-Zahlen

Bei der BCD-Zahlendarstellung (Binary-Coded-Decimal) werden die höherwertigen und niederwertigen 4 Bit (die sogenannten Nibbel oder Halbbytes) eines Bytes zur Darstellung jeweils einer binärcodierten Dezimalziffer verwendet. Der Inhalt der Nibbels kann dabei den Wert 0 bis 9 bzw. 0000B bis 1001B betragen.

Tabelle 8		
Dezimalzahl	Binär-Zweierkomplement	Hex-Darstellung
+127	0111 1111	7FH
+126	0111 1110	7EH
:	:	:
:	:	:
+2	0000 0010	02H
+1	0000 0001	01H
0	0000 0000	00H
-1	1111 1111	FFH
-2	1111 1110	FEH
:	:	:
:	:	:
-127	1000 0001	81H
-128	1000 0000	80H

tigen und niederwertigen 4 Bit (die sogenannten Nibbel oder Halbbytes) eines Bytes zur Darstellung jeweils einer binärcodierten Dezimalziffer verwendet. Der Inhalt der Nibbels kann dabei den Wert 0 bis 9 bzw. 0000B bis 1001B betragen.

Der Vorteil dieser Zifferndarstellung liegt darin, daß sie einfach auszugeben ist und dem dezimalen Zahlensystem entspricht. Allerdings hat sie den Nachteil, daß bei Additionen und Subtraktionen jeweils Überlaufkorrekturen notwendig sind. Hierfür stellt der Mikrocontroller einen speziellen Befehl (DA A) zur Verfügung.

Beispielsweise entspricht die dezimale Zahl 77 in BCD-Darstellung 77H bzw. 0111 0111B.

Im fünften Teil dieser Artikelserie folgt die Beschreibung von Testschaltungen.