

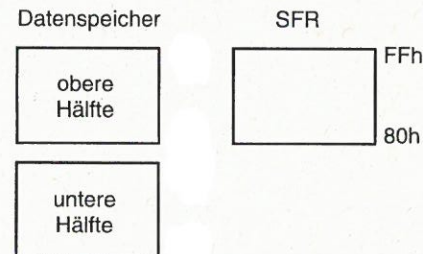
## 5 Organisation des internen Speichers

Der interne Speicher des Controllers besteht aus dem Datenspeicher mit unterer und oberer Hälfte und den Spezial-Funktions-Registern SFR. Es sind also drei Bereiche zu unterscheiden.

Der Speicherbereich für die Spezial-Funktions-Register SFR liegt parallel zur oberen Hälfte des Datenspeichers. Beide haben die gleichen Adressen.

Durch die Art der Adressierung wird dem Controller gesagt, auf welchen Bereich er zugreifen soll. Die obere Hälfte des Datenspeichers wird indirekt adressiert, die SFR werden direkt adressiert.

Interne Speicherbereiche:



### 5.1 Die untere Hälfte des Datenspeichers

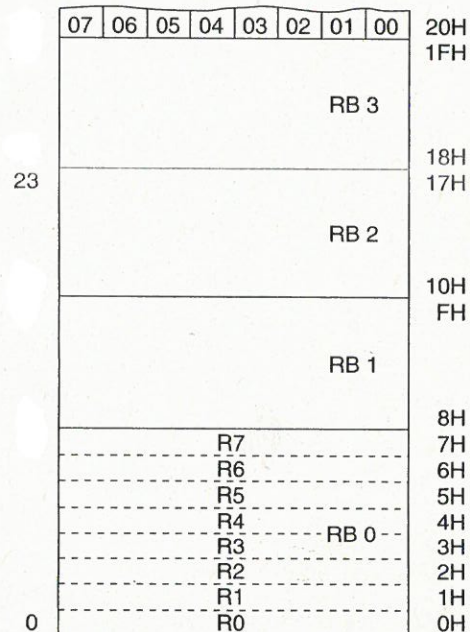
Die untere Hälfte des Datenspeichers ist aufgeteilt in Registerbänke, Bitspeicher und Bytespeicher.

#### Registerbänke RB0 bis RB3

Die vier Registerbänke liegen ab Adresse 0000h an aufwärts. Der Controller arbeitet immer nur mit einer der Registerbänke. Nach einem Reset ist Registerbank null eingeschaltet. Die Register R0 und R1 sind für die indirekte Adressierung des internen Datenspeichers einzusetzen. In die anderen Register lassen sich Zwischenergebnisse ablegen.

Bei einem Sprung in ein Unterprogramm kann die Registerbank gewechselt werden. Beim Rücksprung in das alte Programm lässt sich wieder auf die alte Registerbank umschalten. Damit sind auch die alten Zwischenergebnisse wieder vorhanden. Sie müssen beim Umschalten auf ein Unterprogramm also nicht einzeln auf dem Stack gerettet werden.

Registerbänke:



Das Umschalten der Registerbänke erfolgt über die Bits drei und vier in dem Spezial-Funktions-Register PSW (Programm Status Wort). Die in diese beiden Stellen gesetzte Dualzahl ist die Nummer der benutzten Registerbank.

#### Bit-Speicher

Bei den Bytes 20h bis 2Fh des internen Datenspeichers lässt sich jedes Bit einzeln adressieren. Die 128 Bits sind von 00h bis 7Fh durchnummeriert. In diesem Speicherbereich können sowohl Bits als auch die Bytes mit ihrer Hex-Adresse angesprochen werden. Ob es sich um eine Bit-oder Byte-Adresse handelt, erkennt der Controller an dem jeweiligen Assemblerbefehl.

Der Befehlssatz enthält extra eine Gruppe von Befehlen für die Einzelbit-Verarbeitung. Die Bitverarbeitung ermöglicht ein direktes Programmieren von Verknüpfungssteuerungen.

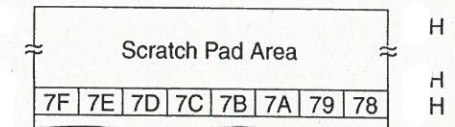
Bit-Speicher:

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

#### Byte-Speicher

Ab Adresse 30h beginnt der Byte-Speicher. Er umfasst 80 Bytes und geht bis Adresse 7Fh.

Byte-Speicher:



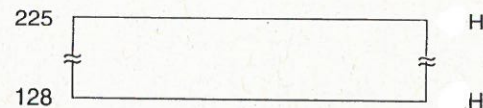


## Untere Hälfte interner Datenspeicher

127	Scratch Pad Area								7FH
48									30H
47	7F	7E	7D	7C	7B	7A	79	78	2FH
46	77	76	75	74	73	72	71	70	2EH
45	6F	6E	6D	6C	6B	6A	69	68	2DH
44	67	66	65	64	63	62	61	60	2CH
43	5F	5E	5D	5C	5B	5A	59	58	2BH
42	57	56	55	54	53	52	51	50	2AH
41	4F	4E	4D	4C	4B	4A	49	48	29H
40	47	46	45	44	43	42	41	40	28H
39	3F	3E	3D	3C	3B	3A	39	38	27H
38	37	36	35	34	33	32	31	30	26H
37	2F	2E	2D	2C	2B	2A	29	28	25H
36	27	26	25	24	23	22	21	20	24H
35	1F	1E	1D	1C	1B	1A	19	18	23H
34	17	16	15	14	13	12	11	10	22H
33	0F	0E	0D	0C	0B	0A	09	08	21H
32	07	06	05	04	03	02	01	00	20H
31	RB 3								1FH
24									18H
23									17H
	RB 2								
16									10H
15									FH
	RB 1								
8									8H
7									7H
				R7					6H
				R6					5H
				R5					4H
				R4					3H
				R3					2H
				R2					1H
				R1					0H
0				R0					

## 5.2 Die obere Hälfte des Datenspeichers

Die obere Hälfte des Datenspeichers umfasst 128 Bytes und geht von Adresse 80h bis FFh. Die Bytes sind mithilfe der Register R0 oder R1 indirekt adressierbar.



## 5.3 Spezial-Funktions-Register

Die Spezial-Funktions-Register befinden sich im Adressbereich 80h bis FFh. Sie sind über direkte Adressierung anzusprechen. Sie lassen sich mit ihrer absoluten Hexadresse benennen oder, bei einem geeigneten Übersetzerprogramm, auch mit ihrer symbolischen Adresse. Einige SFR sind nur als komplettes Byte anzusprechen, andere sind bitadressierbar. Bei geeignetem Übersetzerprogramm sind auch diese einzelnen Bits mit ihrem symbolischen Namen zu benennen.

*Beispiele:*

Setzen des Timer-Modus-Registers auf den Wert 0Fh (auf die Funktion der Register soll hier nicht eingegangen werden):

Symbolische Adresse des SFR: TMOD

Absolute Adresse des SFR: 89h  
nicht bitadressierbar

Freigabe des Timer 0 durch Setzen des Timer Run Flags TR0 auf 1:

Symbolische Adresse des SFR: TCON

Absolute Adresse des SFR: 88h

bitadressierbar

Symbolische Adresse des Bit: TR0

Absolute Adresse des Bit: 8Ch

Die Spezial-Funktions-Register enthalten alle Register, die der Controller für seine Arbeit, für die Ein- und Ausgabe und für interne Einstellungen benötigt.

Auf die Bedeutung der Register oder einzelner Bits wird eingegangen, wenn sie für die Realisierung bestimmter Funktionen benötigt werden.

### Liste der SFR-Register

Block	Symbol	Funktion	Adresse	Bitadr.
CPU	ACC	Akkumulator	E0H	•
	B	B-Register	F0H	•
	DPL	Data Pointer, Low-Byte	82H	
	DPH	Data Pointer, High-Byte	83H	
	PSW	Programmstatuswort	D0H	•
	SP	Stack Pointer	81H	
Ports	P0	Port 0	80H	•
	P1	Port 1	90H	•
	P2	Port 2	A0H	•
	P3	Port 3	B0H	•
	P4	Port 4	E8H	•
	P5	Port 5	F8H	•
	P6	Port 6 (nur Eingang; im 80515 nicht vorhanden)	DBH	



Block	Symbol	Funktion	Adresse	Bitadr.
Serielle Schnittstelle	ADCON (80(C)515)	A/D-Wandler-Steuerregister	D8H	•
	PCON	Stromaufnahme-Steuerregister	87H	
	SCON	Steuerregister der ser. Schnittstelle	98H	•
	SBUF	Buffer der seriellen Schnittstelle	99H	
Timer 0/1	TCON	Steuerregister für Timer 0/1	88H	•
	TMOD	Modusauswahl für Timer 0/1	89H	
	TL0	Timer-0-Register, Low-Byte	8AH	
	TH0	Timer-0-Register, High-Byte	8CH	
	TL1	Timer-1-Register, Low-Byte	8BH	
	TH1	Timer-1-Register, High-Byte	8DH	
Watchdog	IEN0	Interrupt-Freigaberegister 0	A8H	•
	IEN1	Interrupt-Freigaberegister 1	B8H	•
	IP0	Interrupt-Priorität-Steuerung 0	A9H	
	IP1	Interrupt-Priorität-Steuerung 1	B9H	
Interrupt-System	IEN0	Interrupt-Freigaberegister 0	A8H	•
	IEN1	Interrupt-Freigaberegister 1	B8H	•
	IP0	Interrupt-Priorität-Steuerung 0	A9H	
	IP1	Interrupt-Priorität-Steuerung 1	B9H	
	IRCON	Interrupt-Request-Register	C0H	•
	TCON	Steuerregister für Timer 0/1	88H	•
	T2CON	Steuerregister für Timer 2	C8H	•
A/D-Wandler	ADCON (80(C)515)	A/D-Wandler-Steuerregister	D8H	•
	DAPR (80(C)515)	Steuerregister für programmierbare Referenzspannungen	DAH	
	ADDAT (80(C)515)	A/D-Wandler-Ergebnisregister	D9H	
Timer-2-Block	CCEN	Compare/Capture-Freigaberegister	C1H	
	CCL1	Compare/Capture-Register 1, Low-Byte	C2H	
	CCH1	Compare/Capture-Register 1, High-Byte	C3H	
	CCL2	Compare/Capture-Register 2, Low-Byte	C4H	
	CCH2	Compare/Capture-Register 2, High-Byte	C5H	
	CCL3	Compare/Capture-Register 3, Low-Byte	C6H	
	CCH3	Compare/Capture-Register 3, High-Byte	C7H	
	CRCL	Compare/Rel./Capture-Register, Low-Byte	CAH	
	CRCH	Compare/Rel./Capture-Register, High-Byte	CBH	
	TL2	Timer-2-Register, Low-Byte	CCH	
	TH2	Timer-2-Register, High-Byte	CDH	
	T2CON	Timer-2-Steuerregister	C8H	•
Modi für reduzierte Stromaufnahme	PCON	Stromaufnahme-Steuerregister	87H	

### Übung 5.1

1. Wie lässt sich der interne Speicher des Controllers gliedern? Wie lassen sich die einzelnen Speichergruppen ansprechen? Machen Sie eine Skizze.
2. Registerbänke:  
Wie viele sind vorhanden? Wozu werden sie eingesetzt? Wie erfolgt die Umschaltung der Registerbänke?
3. Wie viele direkt adressierbare Bits lassen sich speichern? Unter welchen Adressen?
4. Wie viele direkt adressierbare Bytes lassen sich neben den Registern und dem Bitspeicher im internen RAM ablegen?
5. In welchem Adressbereich liegen die „Spezial-Funktions-Register“? Wie werden sie adressiert?
6. Wie viele Bytes lassen sich in der oberen Hälfte des internen RAMs speichern? Wie werden sie adressiert?