

## **Ausbildungsunterlage für die durchgängige Automatisierungslösung Totally Integrated Automation (T I A)**

***MODUL B4***

**Datenbausteine**

Diese Unterlage wurde von der Siemens AG, für das Projekt Siemens Automation Cooperates with Education (SCE) zu Ausbildungszwecken erstellt.

Die Siemens AG übernimmt bezüglich des Inhalts keine Gewähr.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts ist innerhalb öffentlicher Aus- und Weiterbildungsstätten gestattet. Ausnahmen bedürfen der schriftlichen Genehmigung durch die Siemens AG (Herr Michael Knust [michael.knust@siemens.com](mailto:michael.knust@siemens.com)).

Zuwiderhandlungen verpflichten zu Schadensersatz. Alle Rechte auch der Übersetzung sind vorbehalten, insbesondere für den Fall der Patentierung oder GM-Eintragung.

Wir danken der Fa. Michael Dziallas Engineering und den Lehrkräften von beruflichen Schulen sowie weiteren Personen für die Unterstützung bei der Erstellung der Unterlage

SEITE:

1.	Vorwort .....	4
2.	Hinweise zu Datenbausteinen .....	6
3.	Datenbausteine erstellen .....	7

Die folgenden Symbole führen durch dieses Modul:



Information



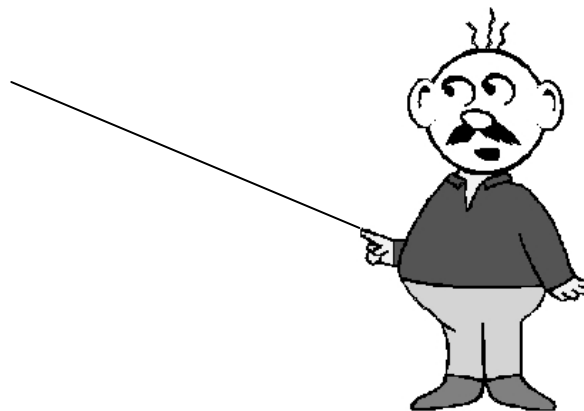
Programmierung



Beispielaufgabe

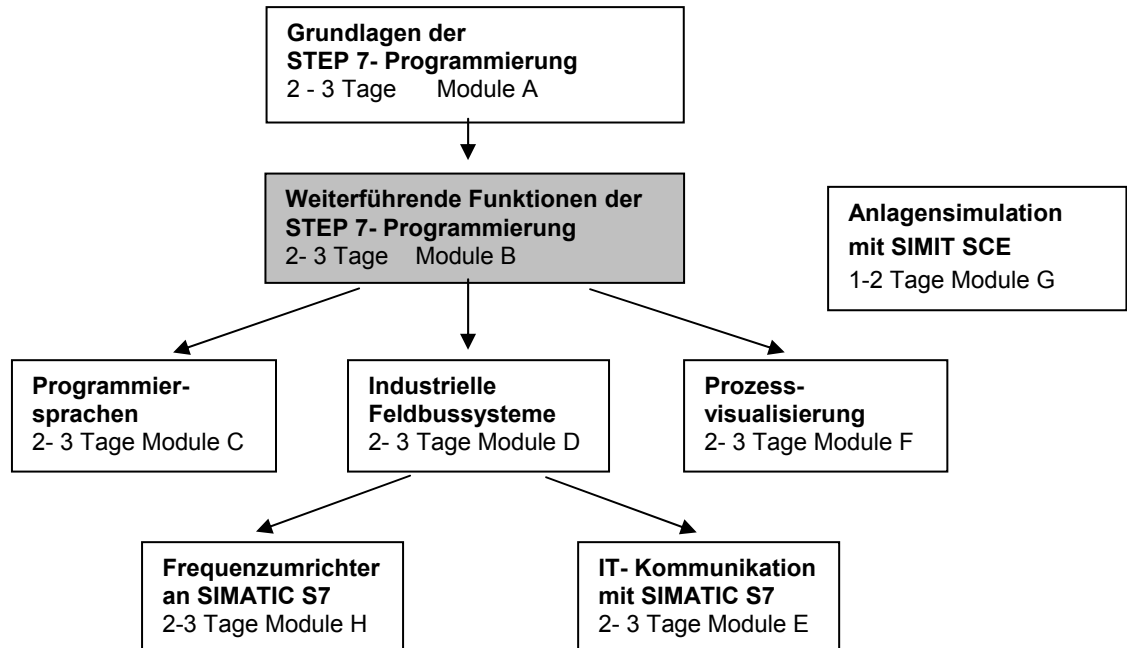


Hinweise



## 1. VORWORT

Das Modul B4 ist inhaltlich der Lehreinheit ‚Weiterführende Funktionen der STEP 7-Programmierung‘ zugeordnet.



### Lernziel:

Der Leser soll in den folgenden Schritten lernen wie ein Datenbaustein zur Datenspeicherung genutzt werden kann.

- Datenbaustein erstellen
- Struktur eines Datenbausteins festlegen
- Zugriff auf Datenelemente im STEP 7- Programm

### Voraussetzungen:

Für die erfolgreiche Bearbeitung dieses Moduls wird folgendes Wissen vorausgesetzt:

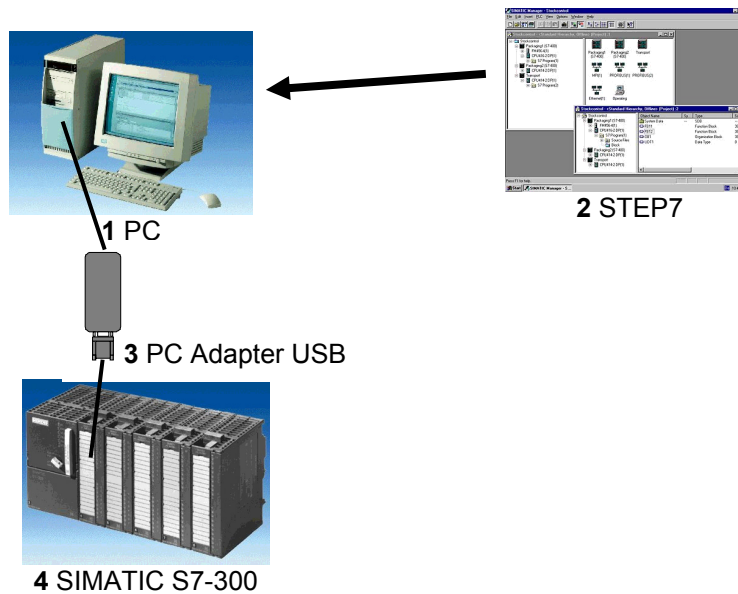
- Kenntnisse in der Handhabung von Windows
- Grundlagen der SPS- Programmierung mit STEP 7 (z.B. Modul A3 - ‚Startup‘ SPS- Programmierung mit STEP 7)
- Grundlagen zur strukturierten Programmierung (z.B. Anhang I - Grundlagen zur SPS – Programmierung mit SIMATIC S7-300)

## Benötigte Hardware und Software

- 1 PC, Betriebssystem Windows XP Professional mit SP2 oder SP3 / Vista 32 Bit Ultimate und Business / Server 2003 SP2 mit 600MHz ( nur XP) / 1 GHz und 512MB ( nur XP) / 1 GB RAM, freier Plattenspeicher ca. 650 - 900 MB, MS-Internet-Explorer 6.0 und Netzwerkkarte
- 2 Software STEP7 V 5.4
- 3 MPI- Schnittstelle für den PC (z.B. PC Adapter USB)
- 4 SPS SIMATIC S7-300 mit mindestens einer digitalen Ein- und Ausgabebaugruppe. Die Eingänge müssen auf ein Schaltfeld herausgeführt sein.

Beispielkonfiguration:

- Netzteil: PS 307 2A
- CPU: CPU 314
- Digitale Eingänge: DI 16x DC24V
- Digitale Ausgänge: DO 16x DC24V / 0,5 A



## 2. HINWEISE ZU DATENBAUSTEINEN



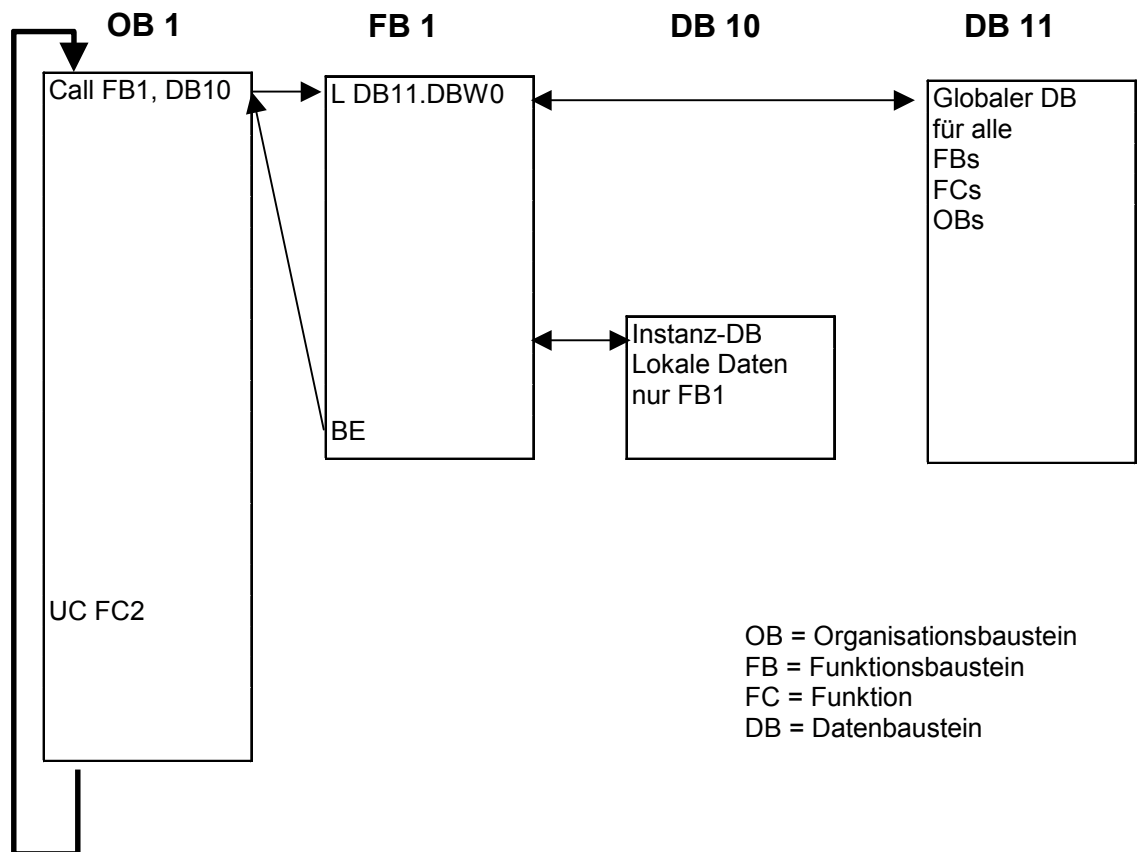
Datenbausteine (DB) können von Ihrem Programm zum Speichern von Daten in der CPU verwendet werden. Ihr Speicherplatz beträgt je nach CPU bis zu 8 KBytes ( 8192 Bytes ) oder mehr.

Es gibt zwei Arten von Datenbausteinen. **Globale** DBs, wo alle OBs, FBs und FCs die gespeicherten Daten lesen oder selbst Daten in den DB schreiben können und **lokale Instanz-** DBs, die einem bestimmten FB zugeordnet sind.

In den DBs können verschiedene Datentypen (z.B. BOOL oder WORD ) in beliebiger Reihenfolge gespeichert werden.

Diese Strukturierung eines DBs erfolgt durch Eingabe in eine Tabelle, mit dem Tool ‚KOP, AWL,FUP - S7 Bausteine Programmieren‘ .

In der Programmstruktur von STEP 7 sind Datenbausteine folgendermaßen zu finden:



### 3. DATENBAUSTEINE ERSTELLEN



Datenbausteine werden wie Programmbausteine im Tool '**KOP,AWL,FUP- S7 Bausteine programmieren**' erstellt und geöffnet. Sie dienen z.B. zur Speicherung von Daten und Anlagenzuständen.

Im Folgenden ist die Erstellung eines einfachen Beispielprogramms unter Verwendung eines globalen Datenbausteins beschrieben:



Dabei sollen mit den Schaltern 'S0' bis 'S7' Werte in einem Datenbaustein angewählt und auf einer Ausgangsbaugruppe 'Anzeige' angezeigt werden. Dabei hat, bei Betätigung mehrerer Schalter, die Darstellung des Wertes zu Schalter S7 die höchste und die Darstellung des Wertes zu Schalter S1 die niedrigste Priorität.

Das Beispiel bezieht sich auf die unten dargestellten Adressen:

#### **Eingänge:**

- Schalter S0 = E 0.0
- Schalter S1 = E 0.1
- Schalter S2 = E 0.2
- Schalter S3 = E 0.3
- Schalter S4 = E 0.4
- Schalter S5 = E 0.5
- Schalter S6 = E 0.6
- Schalter S7 = E 0.7

#### **Ausgänge:**

- Anzeige = AW4



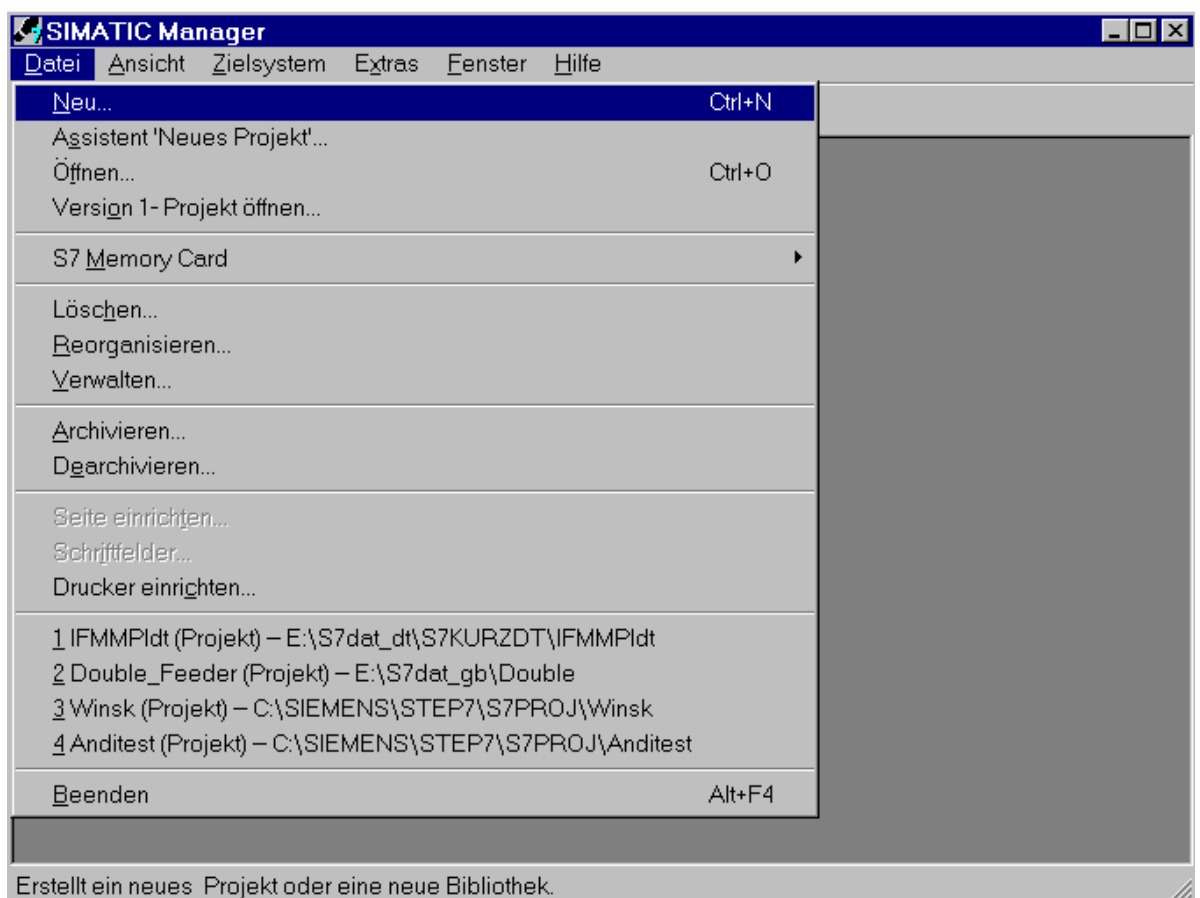
Zum Erstellen dieses Programmbeispiels müssen die folgenden Schritte durchgeführt werden (Dabei wird auf die Erstellung einer Hardwarekonfiguration verzichtet.):

1. Mit einem Doppelklick den ‚**SIMATIC Manager**‘ aufrufen ( → SIMATIC Manager)



SIMATIC Manager

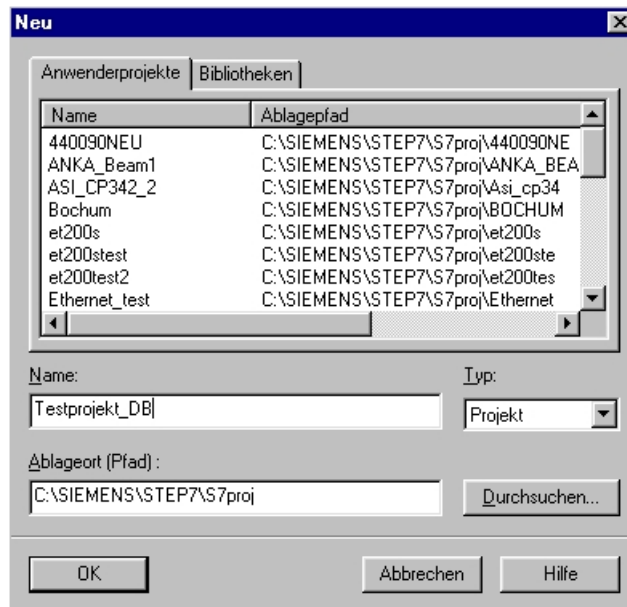
2. Dann ein Neues Projekt anlegen ( → Datei → Neu)



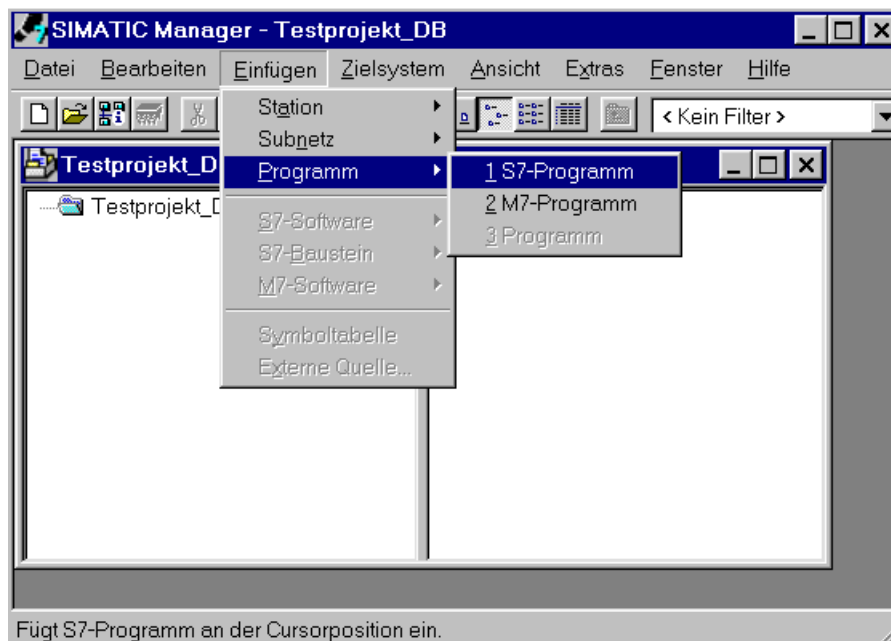




- In dem Standardpfad das Projekt mit dem Namen ‚Testprojekt\_DB‘ erstellen. (→ ‚Testprojekt\_DB‘ → OK)

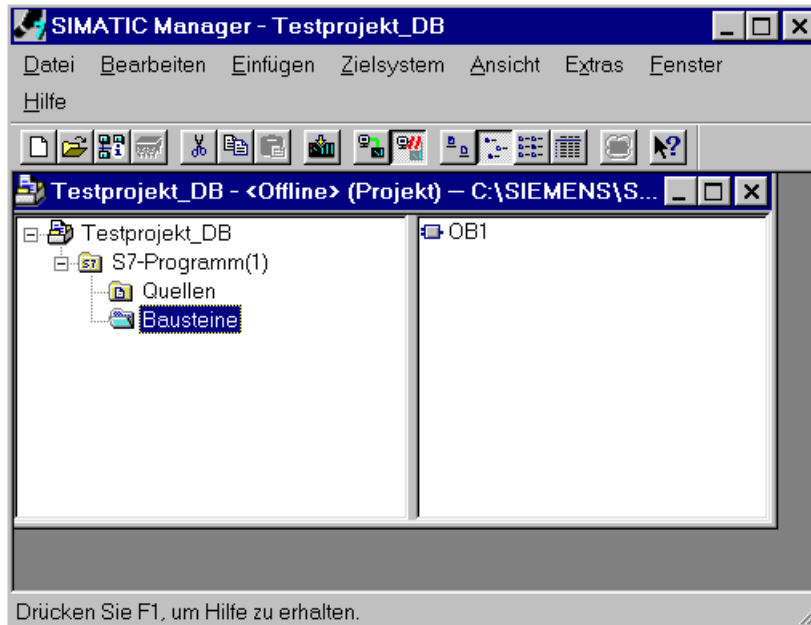


- Neues ‚S7-Programm‘ einfügen ( → Einfügen → Programm → S7-Programm).

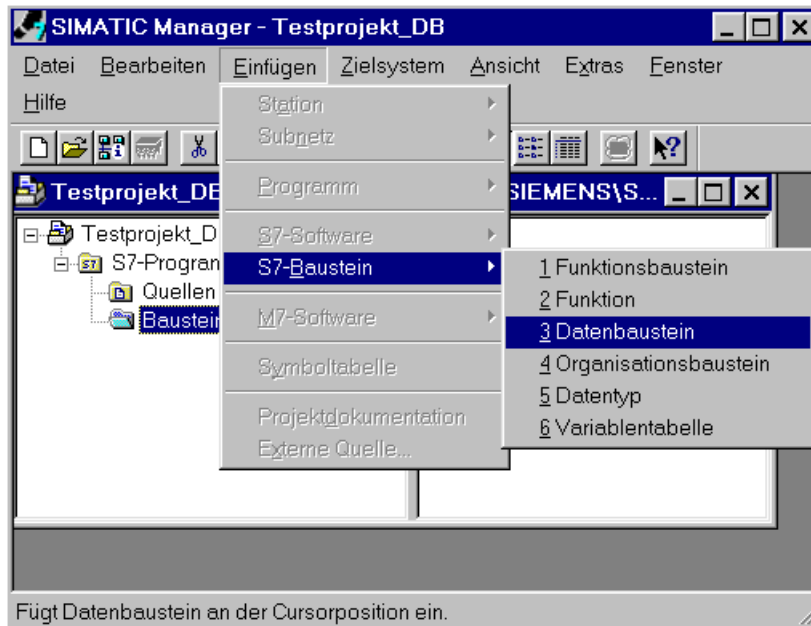




5. Ordner **'Bausteine'** markieren. (→ Bausteine)



6. **'Datenbaustein'** einfügen ( → Einfügen → S7-Baustein → Datenbaustein).





7. Nummer für Datenbaustein ,DB10' eintragen und mit ,OK' bestätigen und als Typ ,Global-DB' wählen. (→ DB10 → Global-DB → OK)

**Eigenschaften - Datenbaustein**

Allgemein - Teil 1 | Allgemein - Teil 2 | Aufrufe | Attribute

Name und Typ: DB10 Global-DB

Symbolischer Name:

Symbolkommentar:

Erstellsprache: DB

Projektpfad:

Speicherort des Projekts: C:\Data an Mde10\0\_S7\_Projekte\TESTPROJ

	Code	Schnittstelle
Erstellt am:	03.03.2003 21:15:24	
Zuletzt geändert am:	03.03.2003 21:15:24	03.03.2003 21:15:24

Kommentar:

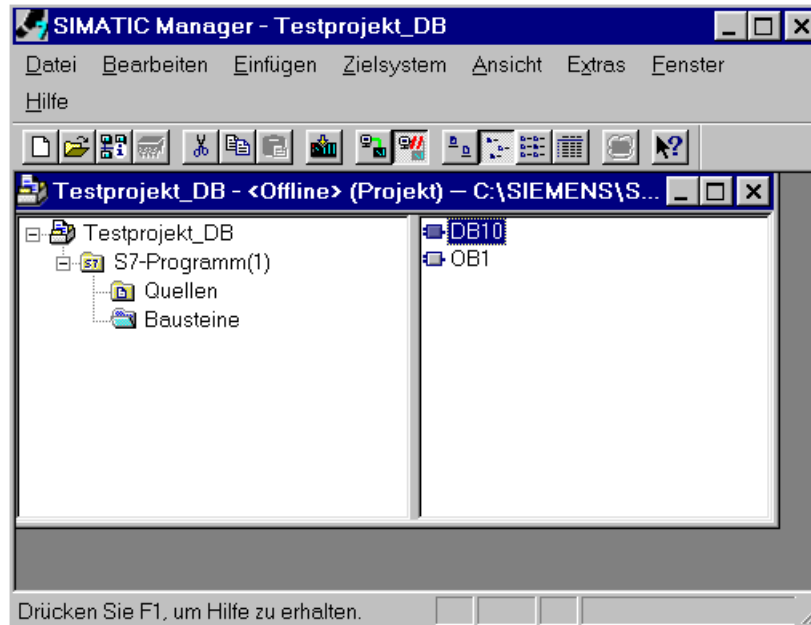
OK Abbrechen Hilfe



- Hinweis:** Bei Typ können ("Global-DB", "Instanz-DB" bzw. "DB vom Typ") ausgewählt werden. "Instanz-DBs" Datenbausteine mit zugeordnetem FB werden beim Aufruf Ihres zugehörigen FBs automatisch erstellt. Es macht also zumeist keinen Sinn diesen hier selbst zu erstellen.
- Existieren im Bausteinordner FBs oder SFBs, so kann in dieser Klappliste der Typ "Instanz-DB" gewählt werden und in der nächsten Klappliste ein FB oder SFB zugeordnet werden.
- Datenbausteine mit zugeordnetem Anwenderdefinierten Datentyp (UDT) sind Datenbausteine, deren Struktur schon vorher in diesem UDT festgelegt wurde. Existieren im Bausteinordner UDTs, so kann in der Klappliste der Typ "DB vom Typ" gewählt werden und in der nächsten Klappliste ein UDT zugeordnet werden.







8. Mit einem Doppelklick den Datenbaustein ‚DB10‘ öffnen (→ DB10).





9. Der Datenbaustein wird erstellt, indem zu jedem Datenelement ein symbolischer **Name**, der **Typ**, ein **Anfangswert** sowie ein **Kommentar** (optional) eingegeben wird. Die Adresse wird automatisch generiert und kann nicht geändert werden.

Dann kann der Datenbaustein gespeichert  und in die SPS geladen  werden. Dabei muss der Schlüsselschalter an der CPU auf STOP stehen! (→ Name → Typ → Anfangswert → Kommentar →  →  )

Adresse	Name	Typ	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	Wert1	WORD	W#16#0	Wert der dem Schalter S0 zugeordnet ist
+2.0	Wert2	WORD	W#16#1	Wert der dem Schalter S1 zugeordnet ist
+4.0	Wert3	WORD	W#16#2	Wert der dem Schalter S2 zugeordnet ist
+6.0	Wert4	WORD	W#16#3	Wert der dem Schalter S3 zugeordnet ist
+8.0	Wert5	INT	16	Wert der dem Schalter S4 zugeordnet ist
+10.0	Wert6	INT	32	Wert der dem Schalter S5 zugeordnet ist
+12.0	Wert7	INT	64	Wert der dem Schalter S6 zugeordnet ist
+14.0	Wert8	INT	256	Wert der dem Schalter S7 zugeordnet ist
+16.0		END_STRUCT		

Symbolischer Name mit der auf die absolute Adresse verwiesen wird.

Gewünschter Datentyp (siehe unten) für Ihr Datenelement.

Anfangswert, der zum Datentyp kompatibel sein muss. (optional)

Kommentar zur Dokumentation. (optional)

Die absolute Adresse wird von STEP 7 automatisch erzeugt, wenn der DB übersetzt oder gespeichert wird. Das Adressformat ist **BYTE.BIT**. Über diese Adresse kann auf das Datenelement zugegriffen werden (z.B. über Lade- und Transferoperationen oder in logischen Verknüpfungen.).



**Hinweis:** Ist der Datenbaustein als lokaler Instanz-DB einem FB zugeordnet, so legt die Deklarationstabelle des FB die Struktur des DB fest.



Daten in einem Datenbaustein müssen durch Datentypen bestimmt werden.  
 Folgende Standard- Datentypen sind unter anderen in der S7 definiert :

Typ und Beschreibung	Größe in Bits	Format-option	Bereich und Zahlendarstellung niedrigster bis höchster Wert	Beispiel
BOOL (Bit)	1	Bool-Text	TRUE/FALSE	TRUE
BYTE (Byte)	8	Hexadezimal	B#16#0 bis B#16#FF	B#16#10
WORD (Wort)	16	Dualzahl	2#0 bis 2#1111_1111_1111_1111	2#0001_0000_0000_0000
		Hexadezimalzahl	W#16#0 bis W#16#FFFF	W#16#1000
		BCD	C#0 bis C#999	C#998
		Dezimalzahl (o.V.)	B#(0,0) bis B#(255,255)	B#(10,20)
DWORD (Doppelwort)	32	Dualzahl	2#0 bis 2#1111_1111_1111_1111_1111_1111_1111_1111	2#1000_0001_0001_1000_10_11_1011_0111_1111
		Hexadezimalzahl	DW#16#0000_0000 bis DW#16#FFFF_FFFF	DW#16#00A2_1234
		Dezimalzahl (o.V.)	B#(0,0,0,0) bis B#(255,255,255,255)	B#(1,14,100,120)
INT (Ganzzahl)	16	Dezimalzahl	-32768 bis 32767	1
DINT (Ganzzahl,32 bit)	32	Dezimalzahl	L#-2147483648 bis L#2147483647	L#1
REAL (Gleitpunktzahl)	32	IEEE Gleitpunktzahl	Oberer Grenze: +/-3.402823e+38 Untere Grenze: +/-1.175495e-38	1.234567e+13
S5TIME (Simatic-Zeit)	16	S7-Zeit in Schritten von 10 ms	S5T#0H_0M_0S_10MS bis S5T#2H_46M_30S_0MS und S5T#0H_0M_0S_0MS	S5T#0H_1M_0S_0MS S5TIME#1H_1M_0S_0MS
TIME (IEC-Zeit)	32	IEC-Zeit in Schritten von 1ms, Ganzzahl mit Vorzeichen	-T#24D_20H_31M_23S_648MS bis T#24D_20H_31M_23S_647MS	T#0D_1H_1M_0S_0MS TIME#0D_1H_1M_0S_0MS
DATE (IEC-Datum)	16	IEC-Datum in Schritten von 1 Tag	D#1990-1-1 bis D#2168-12-31	DATE#1994-3-15
TIME_OF_DAY (Uhrzeit)	32	Uhrzeit in Schritten von 1ms	TOD#0:0:0.0 bis TOD#23:59:59.999	TIME_OF_DAY#1:10:3.3
CHAR (Zeichen)	8	ASCII-Zeichen	'A', 'B' usw.	'B'



10. Sollen Werte in einem Datenbaustein nachträglich geändert werden reicht es nicht aus diese **nur** im Feld **„Anfangswert“** zu ändern. Nach der Änderung des Anfangswertes muss auf die **„Ansicht“** **„Datensicht“** umgeschaltet werden.

The screenshot shows the SIMATIC Manager interface. The main window displays a data block structure with the following table:

Adresse	Name	Typ
0.0		STRUCT
+0.0	Wert1	WORD
+2.0	Wert2	WORD
+4.0	Wert3	WORD
+6.0	Wert4	WORD
=8.0		END_STRUCT

The 'Ansicht' menu is open, showing options like 'Katalog', 'AS-Register', 'KOP', 'AWL', 'FUP', 'Datensicht', 'Deklarationssicht', 'Symbolische Darstellung', 'Symbolinformation', 'Kommentar', and 'Warnungen'. 'Datensicht' is highlighted.

11. Über die Funktion **Datenbaustein initialisieren** im Menü **Bearbeiten** werden dann die Anfangswerte als Aktualwerte übernommen.

The screenshot shows the SIMATIC Manager interface with the 'Bearbeiten' menu open. The 'Datenbaustein initialisieren' option is selected. A dialog box titled 'Datenbaustein initialisieren (30:253)' is displayed, asking: 'Sollen alle Aktualwerte mit den Anfangswerten überschrieben werden?' (Should all actual values be overwritten with the initial values?). The dialog has 'Ja', 'Nein', and 'Hilfe' buttons.

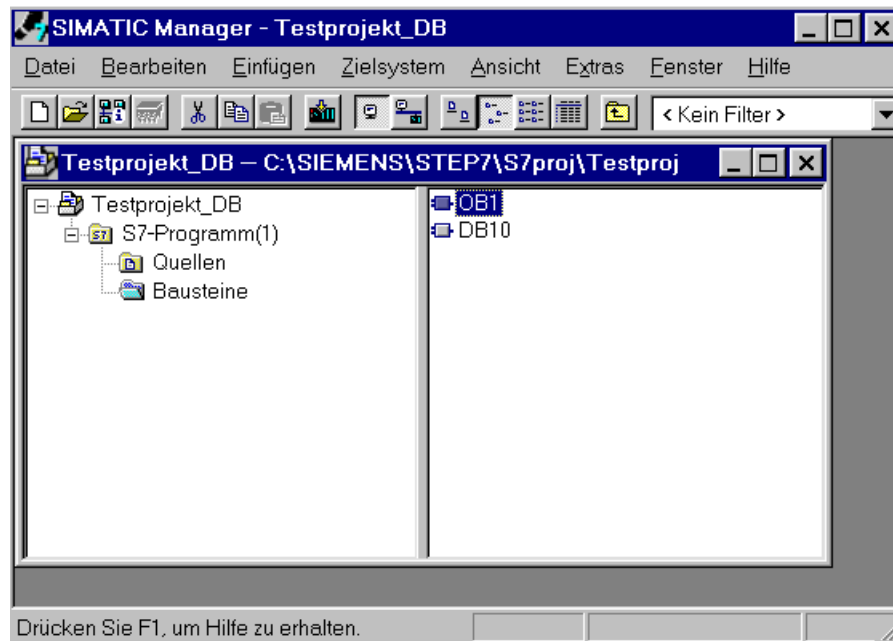
The background table shows the data block structure with columns for 'Typ', 'Anfangswert', and 'Aktualwert':

Typ	Anfangswert	Aktualwert
WORD	W#16#0	W#16#0
WORD	W#16#1	W#16#1
WORD	W#16#2	W#16#2
WORD	W#16#3	W#16#3
INT	16	16
INT	32	32
INT	64	64
INT	256	256

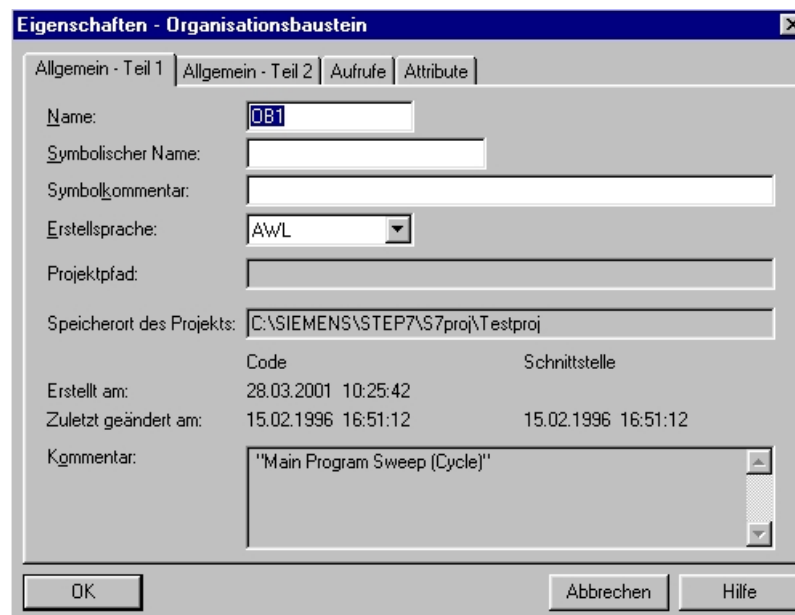
12. Ein neuer Wert kann aber auch direkt im Feld **„Aktualwert“** eingetragen werden. Hier ist jedoch zu beachten, dass bei CPUs mit MMC Card durch Umlöschen die Datenbausteine initialisiert werden und somit wieder alle Anfangswerte als Aktualwerte anstehen.
13. Anschließend wird durch Speichern  auf der Festplatte gesichert sowie durch Laden  der Datenbaustein in die CPU übertragen.



12. Um das Programm mit dem Zugriff auf die Datenelemente zu erstellen wird dann im ‚SIMATIC Manager‘ der ‚OB1‘ durch Doppelklick geöffnet werden ( → SIMATIC Manager → OB1)



13. Dann werden die Einstellungen mit ‚OK‘ übernommen. (→ OK )







**Es gibt drei Möglichkeiten auf Datenelemente zuzugreifen:**

### 1. Direkt adressierter Zugriff:

Auf einzelne Datenelemente in einem Datenbaustein kann mit den folgenden Befehlen direkt zugegriffen werden:

Beispiele:

L	DB 20. DBB2	Lade Datenbyte 2 aus DB20 in AKKU 1
L	DB 22. DBW4	Lade Datenwort 4 aus DB22 in AKKU 1
U	DB 2. DBX5.6	Führe eine UND- Verknüpfung mit Datenbit 5.6 aus DB2 durch.

### 2. Zugriff auf Datenelemente in bereits geöffneten Datenbausteinen:

Um auf einzelne Datenelemente zugreifen zu können kann der DB mit den Befehlen AUF DB oder AUF DI zuerst einmal geöffnet werden. Dann werden einzelne Datenbits (DBX/DIX), Datenbytes (DBB/DIB), Datenwörter (DBW/DIW) oder Datendoppelwörter (DBD/DID) in digitalen oder binären Operationen verarbeitet werden.

Dabei verwendet man AUF DI hauptsächlich für das Öffnen von Instanz- DBs. Es kann jedoch auch für Global - DBs verwendet werden, wenn zwei DBs gleichzeitig geöffnet bleiben sollen.

Beispiel:

AUF	DB 20	Öffnen des DB20
AUF	DI 22	Öffnen des DB22
L	DBW 0	Lade Datenwort 0 aus DB20 in AKKU 1
T	MW 1	Transferiere Inhalt von AKKU 1 nach Merkerwort 1
U	DIX 0.0	UND- Verknüpfung von Datenbit 0.0 aus DB22 mit
U	E 1.0	Eingangsbit 1.0
=	A 4.0	Zuweisung des Ergebnisses zu Ausgangsbit 4.0

### 3. Zugriff auf Daten aus lokalen Instanz- DBs beim Aufruf mit dem Funktionsbaustein:

Schon beim Aufruf eines FB können Daten eines dazugehörigen Instanz - Datenbausteins mit dem Befehl CALL FB1, DB19 übertragen werden. Die Zuordnung der Variablen, die in der Deklarationstabelle des FB definiert wurden und deren Wert im DB steht, zu den absoluten Adressen (z.B. EW0, M 10.0 oder AW4) erfolgt direkt unter dem CALL- Befehl.





Beispiel:

CALL	FB1, DB19	
ZAHL:=	EW 0	Der Variablen ZAHL wird EW 0 als absolute Adresse zugeordnet.
AUS:=	A 4.0	Der Variablen AUS wird A 4.0 als absolute Adresse zugeordnet.



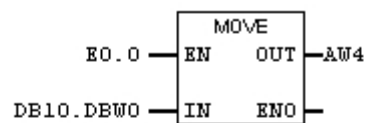
**Hinweis:** Das Format der Variablen und der zugeordneten absoluten Adresse muss gleich sein. Siehe hierzu auch Modul B05 ‚Strukturierte Programmierung mit Funktionsbausteinen‘!



14. Mit 'KOP, AWL, FUP- S7 Bausteine programmieren' haben Sie jetzt einen Editor, der Ihnen die Möglichkeit gibt Ihr STEP7-Programm zu erstellen. Das Schema für die Anwahl der Datenelemente durch die Taster sieht dabei für die ersten vier Eingänge wie unten angegeben aus. Erstellen Sie alle **acht** Netzwerke für die Taster S0 bis S7. Ist der Organisationsbaustein OB1 mit den acht Netzwerken für alle Taster S0 bis S7 richtig erstellt, soll er gespeichert  und in die SPS geladen  werden. Der Schlüsselschalter der CPU steht dabei auf Stop! ( →  →  )

OB1 : Inhalt von DB10 auf Ausgänge schreiben

Netzwerk 1 : Wert zu S0



Netzwerk 2 : Wert zu S1



Netzwerk 3 : Wert zu S2



Netzwerk 4 : Wert zu S3



Netzwerk 5 : Wert zu S4

15. Durch Schalten des Schlüsselschalters auf RUN wird das Programm gestartet. Wird nun einer der Taster S0 bis S7 durch Tippen betätigt so erfolgt die Anzeige des, im Datenbaustein zugeordneten, Wertes auf der Ausgangsbaugruppe mit der Adresse AW4.