

SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

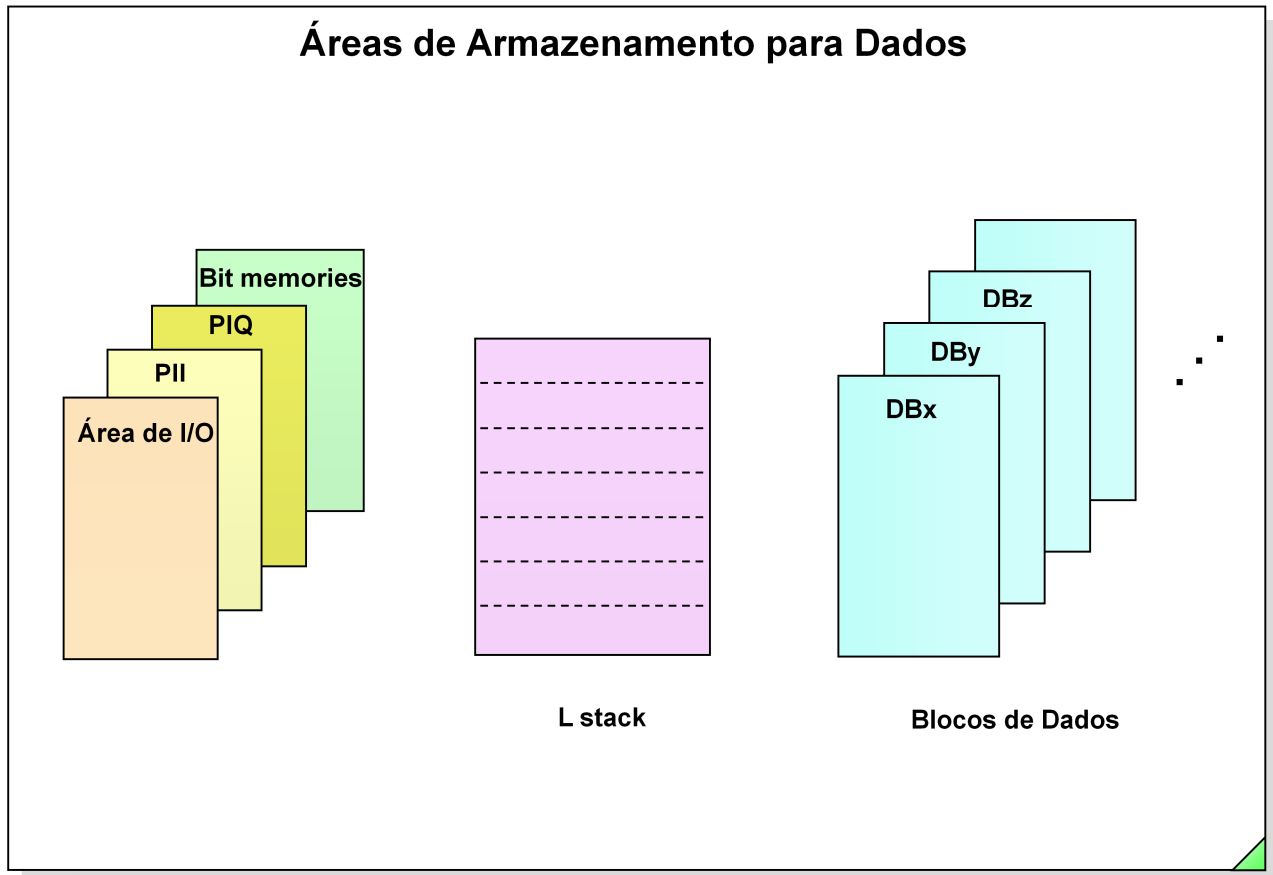
Data: 26/08/2011
Arquivo: S7-Bas-08.1

sitrain

Conteúdo

	Página
Áreas de Armazenamento para Dados	2
Blocos de Dados (DBs)	3
Visão Geral dos Tipos de Dados no STEP 7	4
Tipos de Dados Elementares no STEP 7	5
Criando um Novo Bloco de Dados	6
Editando, Salvando, Transferindo e Monitorando um DB	7
Endereçando os Dados	8
Acessando os Dados	9
Exercício: Contando as Peças Transportadas (FC 18)	10
Tipos de Dados Complexos	11
Exemplo de uma Matriz	12
Exemplo de uma Estrutura	13

Áreas de Armazenamento para Dados



SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

Data: 26/08/2011
Arquivo: S7-Bas-08.2

sitRAIN

Geral

Além dos blocos de programa, o programa de usuário também é constituído por dados que contêm informações sobre os estados do processo, sinais, etc., que são depois processados de acordo com as instruções do programa.

Os dados são armazenados em variáveis do programa do usuário, que são unicamente identificados por:

- Local de armazenamento (endereço: por ex. P, PII, PIQ, bit de memória, L stack, DB)
- Tipo de dado (tipo de dado elementar ou complexo, tipo parâmetro)

Dependendo da acessibilidade, é também feita a distinção entre:

- Variáveis globais, que são declaradas na tabela global de símbolos ou nos blocos de dados globais.
- Variáveis locais, que são declaradas na parte das declarações dos OBs, FBs e FCs.

As variáveis podem ter um local de armazenamento permanente na imagem do processo, área de bits de memória ou num bloco de dados, ou podem ser criadas dinamicamente no L stack quando um bloco está sendo executado.

Local Data Stack

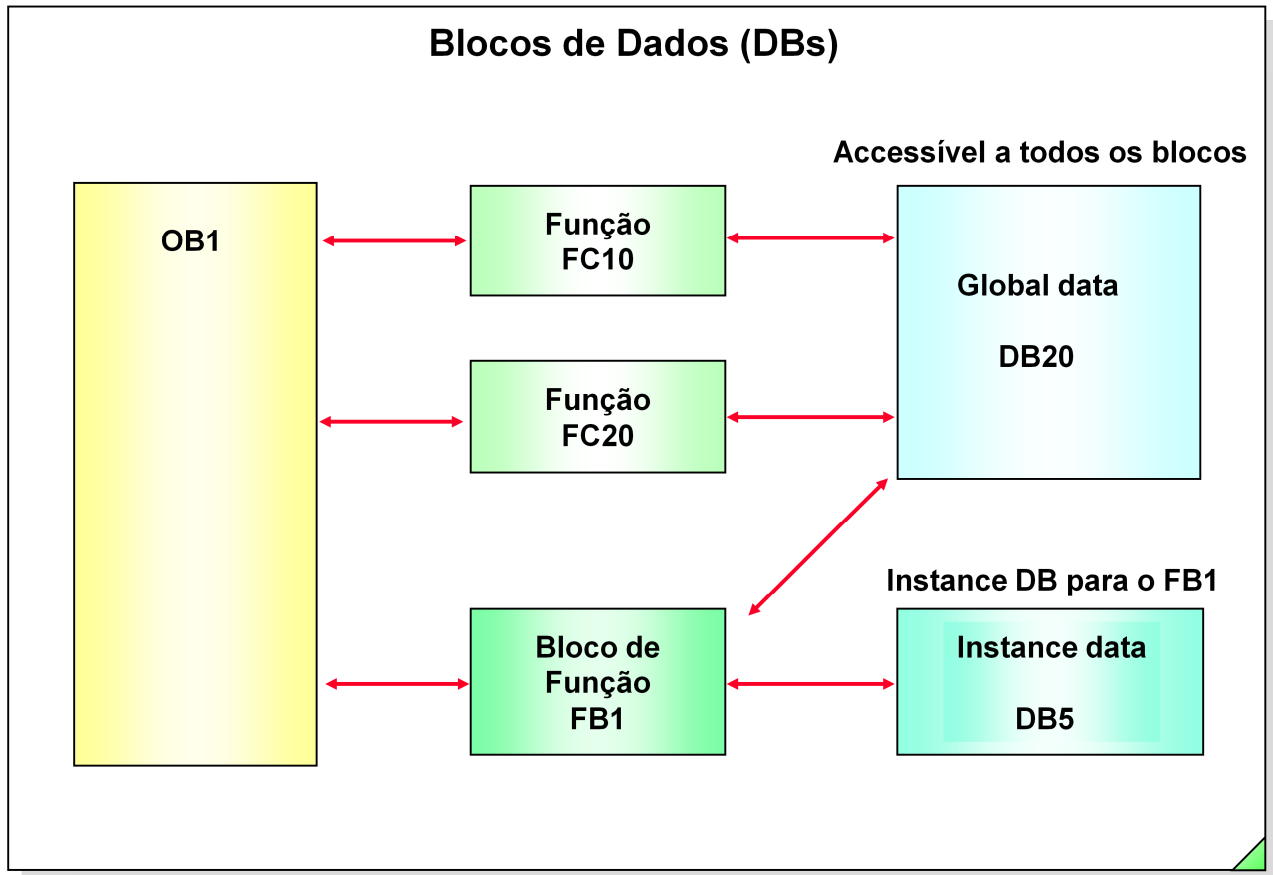
A pilha de dados locais (L stack) é uma área para armazenamento de:

- variáveis temporárias de um bloco lógico, incluindo informações iniciais de Obs;
- parâmetros atuais a serem utilizados nas chamadas das funções;
- resultados lógicos intermediários em programas.

Este tema será tratado no capítulo "Funções e Blocos de Funções".

Blocos de Dados

Os blocos de dados são blocos utilizados pelos blocos lógicos do programa do usuário para o armazenamento de dados. Ao contrário dos dados temporários, os dados nos blocos de dados não são sobrescritos quando a execução do bloco lógico é completada ou quando o DB é fechado.



SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

Data: 26/08/2011
Arquivo: S7-Bas-08.3

sitrain

Geral

Os blocos de dados são utilizados para armazenar dados do usuário. Como acontece com os blocos lógicos, os blocos de dados ocupam espaço na memória de usuário. Os blocos de dados contêm dados variáveis (por ex. valores numéricos) que são utilizadas no programa.

O programa do usuário pode acessar os dados de um bloco de dados através de operações de bit, byte, word (palavra) ou doubleword (dupla palavra). Pode ser utilizado o endereçamento absoluto ou o endereçamento simbólico.

Utilização

Dependendo do seu conteúdo, os blocos de dados podem ser utilizados de diferentes maneiras. Pode-se diferenciar entre:

- **Blocos de dados globais:** estes blocos contêm informação que pode ser acessada por todos os blocos do programa do usuário.
- **Blocos de dados de instance:** estes blocos estão sempre associados a um determinado FB. Os dados em cada DB só devem ser utilizados pelo FB associado. Os blocos de dados instance são tratados em maiores detalhes no capítulo "Funções e Blocos de Funções".

Criando DBs

Os blocos de dados globais podem ser criados tanto com o Editor de Programa como com uma UDT ("user-defined data type") que já tenha sido criado.

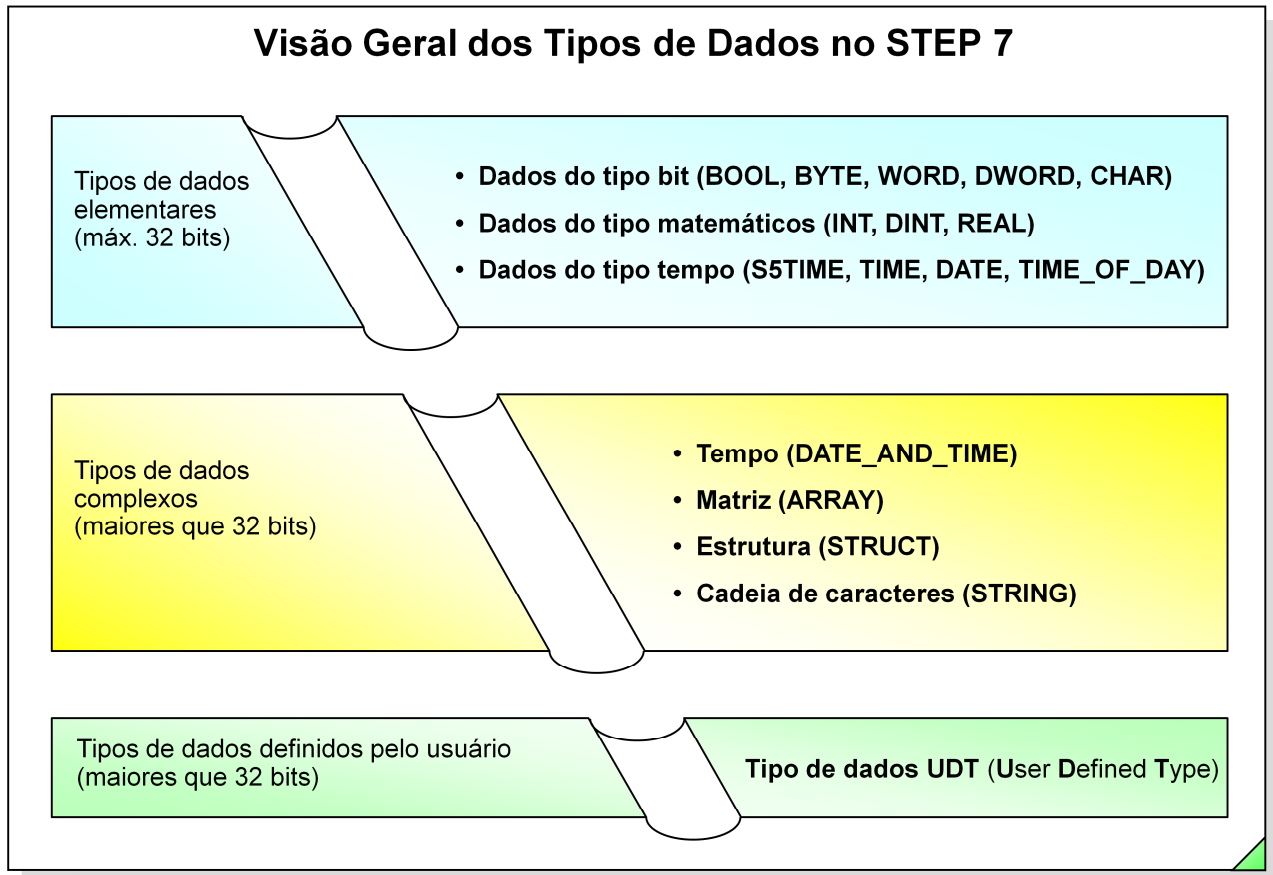
Os blocos de dados instance são criados quando um bloco FB é chamado.

Registradores

A CPU possui dois registradores de bloco de dados, o registrador DB e o registrador DI. Assim sendo, é possível trabalhar com dois blocos de dados abertos ao mesmo tempo.

Maiores informações são encontradas em cursos de programação avançada.

Visão Geral dos Tipos de Dados no STEP 7



SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

Data: 26/08/2011
Arquivo: S7-Bas-08.4

sitRAIN

Geral

Os tipos de dados determinam as propriedades do dado, isto é, a forma como o conteúdo de um ou mais endereços associados deve ser representado e a faixa de valores permitida.

O tipo de dado determina também quais operações podem ser utilizadas.

Dados do Tipo Elementar

Os dados do tipo elementar são pré-definidos de acordo com a norma IEC 1131-3. O tipo de dado determina a quantidade de espaço de memória necessário. Por exemplo, um dado do tipo Word (palavra) ocupa 16 bits na memória de usuário.

Os dados do tipo elementar não têm mais de 32 bits de comprimento. Podem ser carregados com todo o seu conteúdo nos acumuladores de um processador S7 e processados com instruções STEP 7 elementares.

Dados do Tipo Complexo

Os dados do tipo complexo só podem ser utilizados juntamente com variáveis declaradas nos blocos de dados globais. Dados do tipo complexo não podem ser carregados na totalidade do seu conteúdo nos acumuladores através de operações de carregamento. Para estes dados poderem ser processados deve-se utilizar blocos standard da biblioteca ("IEC" S7 Program).

Dados Definidos pelo Usuário

Os dados definidos pelo usuário podem ser utilizados para os blocos de dados ou como um tipo de dado na tabela de declarações de variáveis.

As UDTs são criadas com o editor de blocos de dados.

A estrutura de uma UDT pode conter grupos de dados elementares e/ou dados complexos.

Tipos de Dados Elementares no STEP 7

Palavra-chave	Comprimento (em bits)	Exemplo de uma constante deste tipo
BOOL	1	1 ou 0
BYTE	8	B#16#A9
WORD	16	W#16#12AF
DWORD	32	DW#16#ADAC1EF5
CHAR	8	' w '
S5TIME	16	S5T#5s_200ms
INT	16	123
DINT	32	L#65539
REAL	32	1.2 ou 34.5E-12
TIME	32	T#2D_1H_3M_45S_12MS
DATE	16	D#1993-01-20
TIME_OF_DAY	32	TOD#12:23:45.12

SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

Data: 26/08/2011
Arquivo: S7-Bas-08.5

sitrain

BOOL, BYTE, WORD, DWORD, CHAR

Variáveis do tipo BOOL são compostas por um bit; variáveis do tipo BYTE, WORD, DWORD são seqüências de 8, 16 e 32 bits respectivamente. Os bits não são avaliados individualmente nestes casos. Formas especiais destes tipos de dados são os números BCD e o valor de contagem utilizado juntamente com a função de contagem, bem como os dados do tipo CHAR, que representam um caractere em código ASCII.

S5TIME

Variáveis do tipo S5TIME são necessárias para especificar o valor do tempo em funções que utilizam temporizadores (funções de temporizadores S5). Pode-se especificar o tempo em horas, minutos, segundos e milissegundos. Pode-se introduzir os valores do tempo com "underline" (1h_4m) ou sem "underline" (1h4m). As funções da biblioteca FC 33 e FC 40, convertem o formato S5TIME em formato TIME e formato TIME em formato S5TIME.

INT, DINT, REAL

Variáveis deste tipo representam números que podem ser utilizados em operações matemáticas.

TIME

Uma variável do tipo TIME ocupa uma dupla palavra (doubleword). Esta variável é utilizada, por exemplo, para especificar valores de tempo em funções IEC de temporizadores. Os conteúdos da variável são interpretados como um número DINT (duplo-inteiro) em milissegundos e podem tanto ser positivos como negativos (por ex.: T#1s=L#1 000, T#24d20h31m23s647ms = L#214748647).

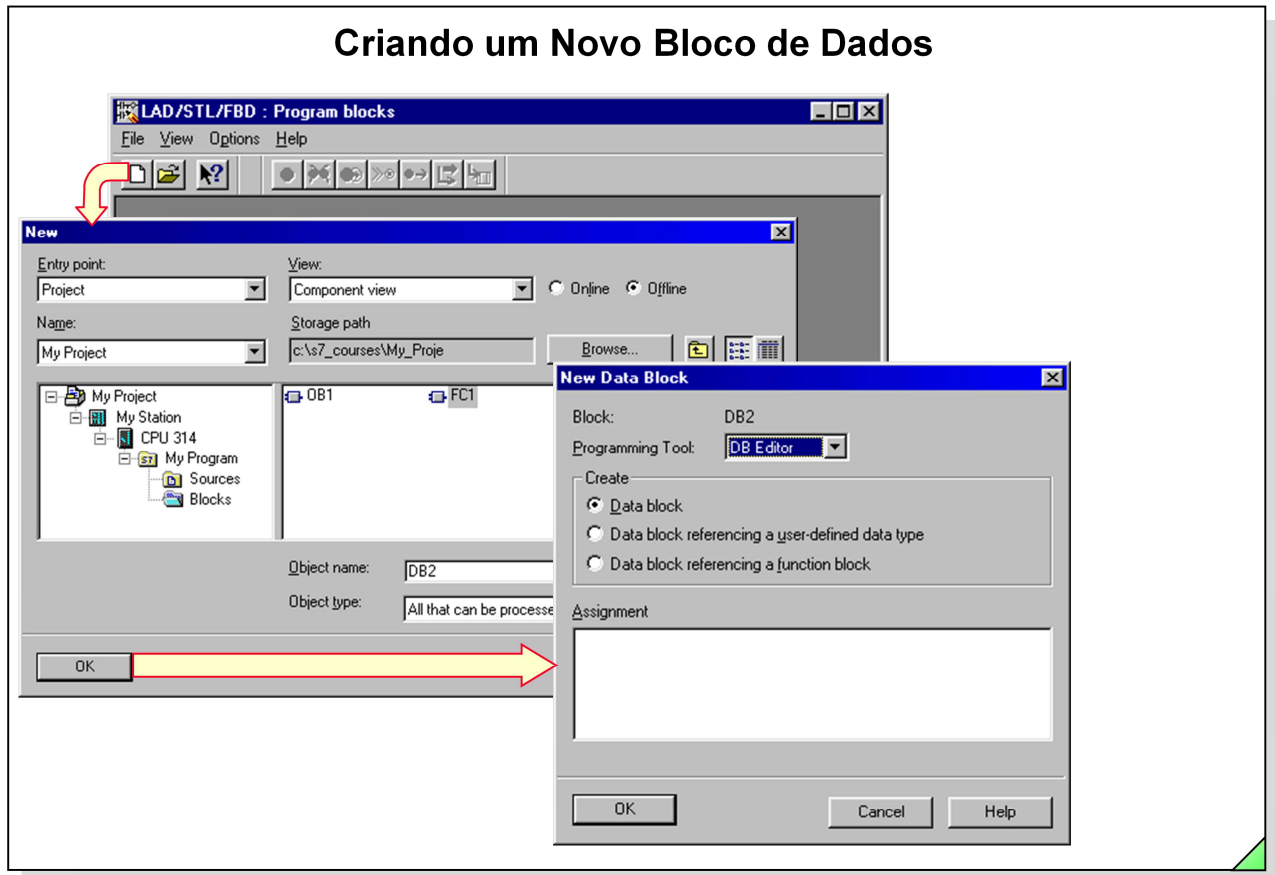
DATE

Uma variável do tipo DATE é armazenada numa palavra (word) no formato de um número inteiro sem sinal. O conteúdo da variável representa o número de dias desde 01.01.1990 (por ex.: D#2168-12-31 = W#16#FF62).

TIME_OF_DAY

Uma variável do tipo TIME_OF_DAY ocupa uma dupla palavra (doubleword). Ela contém o número de milissegundos desde o início do dia (0:00 horas) no formato de um número inteiro sem sinal (por ex.: TOD#23:59:59.999 = DW#16#05265B77).

Criando um Novo Bloco de Dados



SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

Data: 26/08/2011
Arquivo: S7-Bas-08.6

sitrain

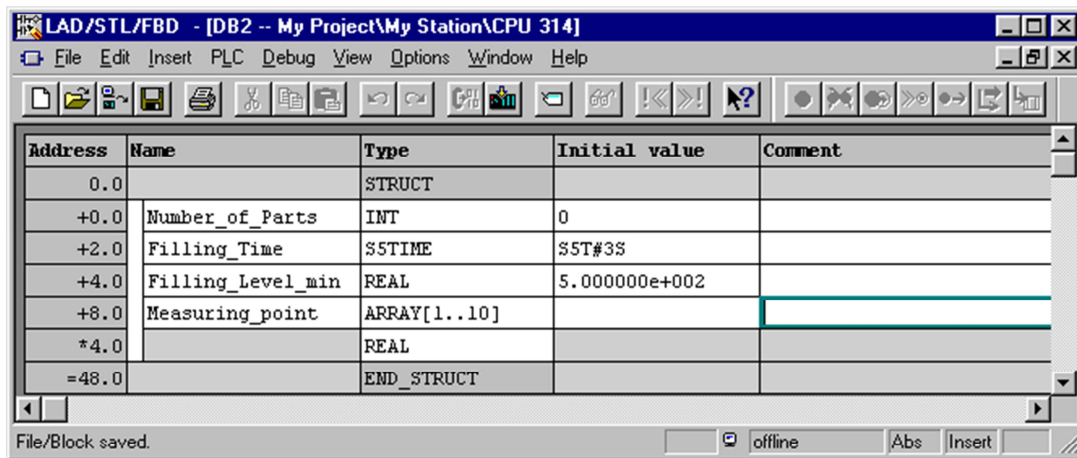
Editor de Programa Com o editor LAD/STL/FBD é possível abrir um bloco de dados já existente ou criar um novo.

Caixa de Diálogo "New" Quando seleciona-se o ícone "New", aparece a caixa de diálogo "New". Após escolher o nome do projeto e do programa de usuário introduza, por exemplo, DB4 no campo "Object name" (como tipo de objeto "Object Type" deverá ser escolhido "Data Block" ou "All Editable"). Após confirmar com o botão "OK", aparece a nova caixa de diálogo chamada "New Data Block".

Caixa de Diálogo "New Data Block" Nesta caixa de diálogo, selecione o tipo de bloco de dados a ser criado:

- Data Block (bloco de dados globais)
- Data Block Referencing a User-Defined Data Type (cria um bloco de dados utilizando a mesma estrutura de um bloco UDT)
- Data Block Referencing a Function Block (cria um bloco de dados instance para um FB). Este ponto será explicado no capítulo "Funções e Blocos de Funções".

Editando, Salvando, Transferindo e Monitorando um DB



SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

Data: 26/08/2011
Arquivo: S7-Bas-08.7

sitrain

Inserindo Dados

Os elementos são introduzidos um a um na tabela. Para isso, selecione a primeira linha vazia na coluna "Name" e introduza a descrição dos elementos. É possível saltar para as outras colunas - "Type" (tipo), "Initial Value" (valor inicial) e "Comment" (comentário) – utilizando-se a tecla Tab.

Colunas

O significado de cada uma das colunas é, respectivamente:

- Address - é introduzido pelo editor de programa ao salvar. Corresponde ao primeiro endereço byte ocupado pela variável no bloco de dados.
- Name - nome simbólico do elemento.
- Type - tipo de dado (selecione com a tecla direita do mouse).
- Initial Value - é utilizado para selecionar um valor default para um elemento. Se não for introduzido um valor neste campo, será utilizado o valor zero como valor inicial.
- Comment - para documentar o elemento da tabela (opcional).

Salvar

Salva o bloco de dados no disco rígido do programador.



Transferir

Transfere o bloco de dados para a CPU da mesma forma que são transferidos os blocos lógicos.

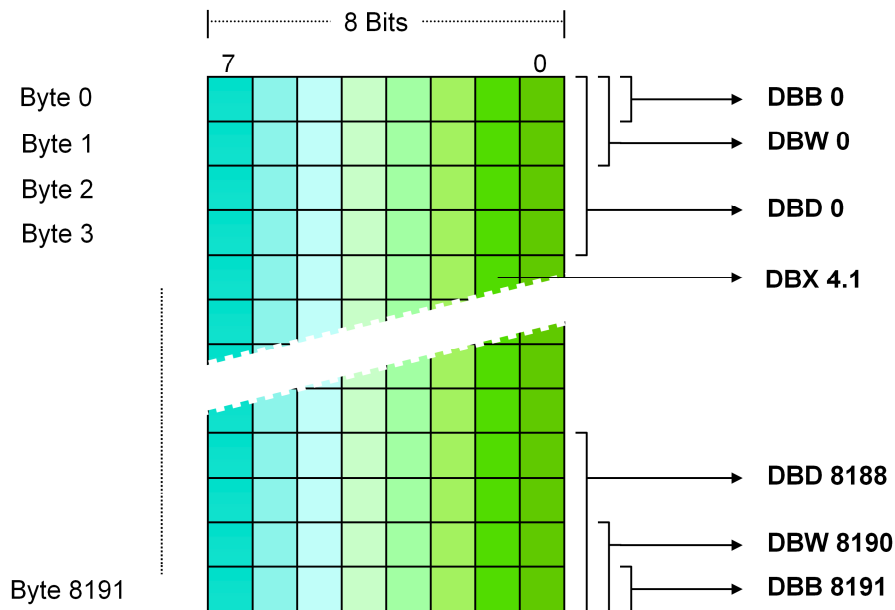


Monitor

Para monitorar os valores atuais de um bloco de dados, deve-se passar para a representação "Data View". Pode-se monitorar um bloco de dados utilizando o ícone dos "Óculos" da barra de ferramentas (visualização permanente dos valores atuais do DB na CPU).



Endereçando os Dados



SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

Data: 26/08/2011
Arquivo: S7-Bas-08.8

sitrain

Geral

Os elementos de um bloco de dados são endereçados byte a byte, ou seja, da mesma maneira que os bits de memória.

Pode-se carregar e transferir bytes de dados, palavras de dados ou duplas palavras de dados. Ao utilizar palavras de dados, deve-se especificar na instrução o endereço do primeiro byte (por ex. L DBW 2). Com esta instrução são carregados dois bytes a partir do endereço referido. Com duplas palavras, são carregados 4 bytes a partir do endereço referenciado.

Número, Comprimento

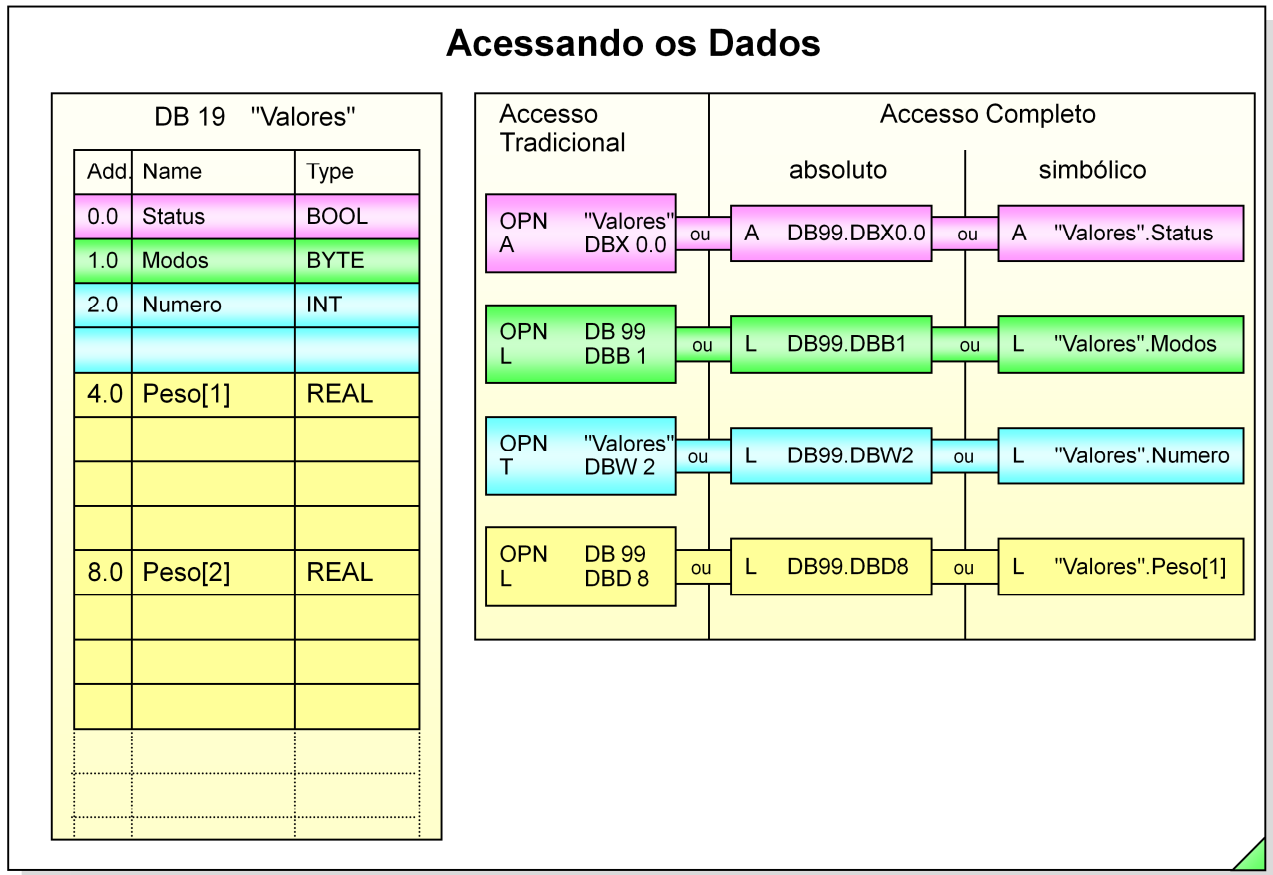
O número de blocos de dados depende da CPU utilizada.

O comprimento máximo de um bloco de dados é de 8KByte para o S7-300 e de 64KByte para o S7-400.

Nota

Ao tentar acessar-se um dado ou um bloco de dados não-existent, a CPU entrará em STOP se não tiver sido programado um OB de erro.

Acessando os Dados



SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

Data: 26/08/2011
Arquivo: S7-Bas-08.9

sitrain

Acesso Tradicional

No acesso tradicional dos blocos de dados (típico na linguagem S5), eles devem ser explicitamente abertos antes do acesso. Isso pode ser feito absoluta ou simbolicamente com a instrução `OPN DB 99` ou `OPN "Valores"` (observe o exemplo). Adicionalmente, o bloco de dados anteriormente aberto é automaticamente fechado. Depois os elementos de dados individuais podem ser acessados bit a bit (DBX...), byte a byte (DBB...), word a word (DBW...) ou doubleword a doubleword (DBD...) sem que o bloco de dados tenha que ser especificado.

Desvantagens:

- Ao acessar os elementos de dados deve-se ter certeza de que o bloco de dados correto está aberto.
- Só pode ser feito acesso absoluto, e assim fica a cargo do programador acessar o endereço correto no bloco de dados. Se a DBW3 no exemplo fosse carregada, nem o valor da variável *Numero* nem o valor da variável *Peso[1]* seria carregado, mas sim um valor inválido.
- Acessos absolutos dificultam correções e tornam o programa difícil de ser lido.

Acesso completo...

A abertura de um bloco de dados e o fechamento do que estava aberto anteriormente estão integrados no acesso completo. O acesso completo pode ser feito absoluto ou simbólico:

...absoluto

A abertura do bloco de dados e o acesso dos dados é feito de forma absoluta em combinação com uma instrução. As desvantagens são similares às do acesso tradicional descritas.

...simbólico

O acesso simbólico a uma variável no bloco de dados é apenas possível no acesso completo, e é completamente simbólico. O Editor permite mesclar os endereçamentos absoluto e simbólico durante a edição; contudo ele altera para a forma completamente simbólica após confirmado o operando.

Exercício: Contando as Peças Transportadas (FC 18)

Address	Name	Type	Initial val.	Comment
0.0		STRUCT		
+0.0	Act Number of Parts	BOOL	FALSE	Actual Value number of Parts
+0.1	Edge_Aux	BOOL	FALSE	Auxiliary Flag for Edge Evaluation
=2.0		END_STRUCT		

SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

Data: 26/08/2011
Arquivo: S7-Bas-08.10

sitrain

Situação Atual: no FC 18

- As peças transportadas no modo AUTO são contadas (por adição no MW 20, com o memory marker auxiliar M 18.0 para detecção de flanco do RLO), assim que atravessam a barreira luminosa.
- O *Número de peças ATUAL* transportadas é exibido no display digital BCD.
- Pode-se escolher o *SETPOINT número de peças* de quantas peças devem ser transportadas usando a chave de pré-seleção BCD. Quando o *SETPOINT número de peças* é atingido, é sinalizado o LED (Q 5.4) na posição de Controle Final.
- Tão logo a mensagem *SETPOINT número de peças atingido* (LED) aparecer, nenhuma outra função de transporte poderá ser iniciada (intertravamento no FC 16).
- O número ATUAL de peças (MW 20) é zerado pela chave não retentiva na posição de Controle Final quando o sistema é desligado ou quando a mensagem *Setpoint número de peças atingido* é reconhecida.

Objetivo:

- A função programada no FC 18 deve permanecer inalterada. Contudo, não continue a utilizar o MW 20 na contagem por adição, use a variável *Número_de_Peças* (INT) a ser declarada no DB 18.
Para a detecção de flanco do RLO necessária a contagem use a variável *Flanco_Aux* (BOOL), no lugar do memory marker auxiliar M 18.0. Essa variável deve também ser declarada no DB 18.

O Que Fazer:

- Modifique o bloco de dados DB 18 (observe a figura) com as variáveis *Número_de_Peças* (INT) e *Flanco_Aux* (BOOL) e transfira-o para a CPU.
- Dê ao DB 18 o nome simbólico "DB_Pecas" na tabela de símbolos globais.
- Faça as modificações no FC 18 como descrito no objetivo. Para isso, trabalhe com acessos simbólicos completos!

Tipos de Dados Complexos

Palavra-chave	Compr. (bits)	Exemplo	
DATE_AND_TIME	64	DT#97-09-24-12:14:55.0	
STRING (max. 254 caracteres)	8 * (número de caracteres) + 16 bits	'Isto é uma string' 'SIEMENS'	
ARRAY (Grupo de elementos do mesmo formato de dados)	definido pelo usuário	Valores Medidos: ARRAY[1..20] INT	
STRUCT (Grupo de elementos de diferentes tipos de dados)	definido pelo usuário	Motor: STRUCT Velocidade: INT Corrente: REAL END_STRUCT	
UDT (User Defined Data Type) "Modelo" composto de dados elementares ou complexos	definido pelo usuário	UDT como bloco	UDT como elemento
		STRUCT Velocidade: INT Corrente: REAL END_STRUCT	Drive: ARRAY[1..4] UDT1

SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

Data: 26/08/2011
Arquivo: S7-Bas-08.11

sitRAIN

Dados do Tipo Complexo

Dados do tipo complexo (matrizes e estruturas) são constituídos por grupos de dados do tipo elementar ou complexo.

Eles permitem criar tipos de dados para resolver determinada tarefa, onde é possível estruturar grandes quantidades de dados e processá-los simbolicamente.

Os dados do tipo complexo não podem ser processados todos ao mesmo tempo (mais de 32 bits) pelas instruções STEP 7, mas apenas um elemento de cada vez. Os dados do tipo complexo são pré-definidos. O dado do tipo DATE_AND_TIME tem um comprimento de 64 bits. O comprimento dos dados do tipo ARRAY, STRUCT e STRING são definidos pelo usuário.

As variáveis de dados do tipo complexo só podem ser declaradas em blocos de dados globais ou parâmetros e variáveis locais de blocos lógicos.

Dados Definidos pelo Usuário

Tipos de dados definidos pelo usuário representam uma estrutura por ele definida. Ela é armazenada em blocos UDT (UDT1 ... UDT65535) e podem ser utilizados como um "modelo" para outro tipo de dado de variáveis. Pode-se poupar tempo de digitação introduzindo um bloco de dados e utilizando a mesma estrutura várias vezes.

Exemplo: É necessária a mesma estrutura 10 vezes num bloco de dados. Primeiro, defina a estrutura e grave-a como UDT1, por exemplo.

No DB, defina a variável "Endereços" como uma matriz ("array") com 10 elementos do tipo UDT1:

```
Endereços  array[1..10]
           UDT 1
```

Assim foram criadas 10 faixas de dados com a estrutura definida na UDT1, sem precisar escrever cada uma individualmente.

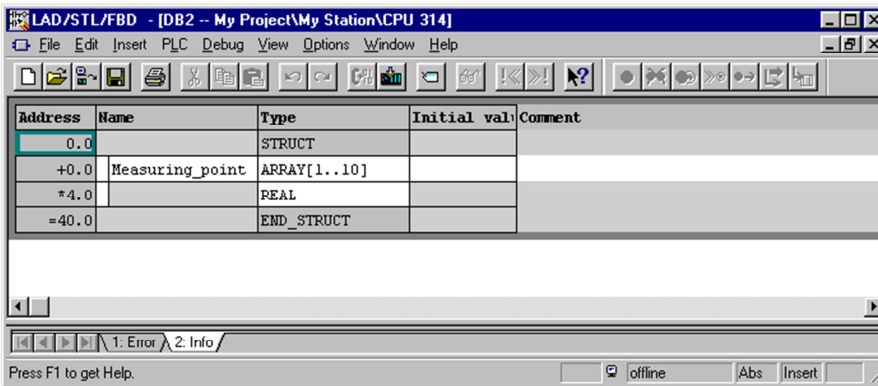
Exemplo de uma Matriz

Measuring_point

- 1. Measuring_point, data type Real
- 2. Measuring_point, data type Real
- 3. Measuring_point, data type Real
- ⋮
- 10. Measuring_point, data type Real

Matriz de nome "Measuring_point"
(vários elementos do mesmo
tipo de dado)

Exibição no Editor de Programa (Bloco de Dados DB 2):



SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

Data: 26/08/2011
Arquivo: S7-Bas-08.12



Array

Uma matriz ("array") é composta por vários elementos do mesmo tipo de dados. Na figura acima ilustrada, pode-se ver a matriz "Measuring_point" com 10 elementos do tipo de dados REAL. Posteriormente, alguns dos valores medidos deverão ser armazenados nesta matriz.

Definir Array num DB

O formato para uma matriz é "ARRAY[n..m]". O primeiro (n) e o último elemento (m) são especificados em colchetes. No exemplo, [1..10] significa 10 elementos, onde o primeiro é endereçado com o index [1] e o último com o index [10]. Em vez de [1..10] poderia, por exemplo, definir [0..9]. Esta representação apenas afeta o acesso aos elementos.

Nota

Para criar um bloco de dados vazio, pode-se definir uma matriz com o tipo de dados pretendidos.

Data View

Para ver que valores estão armazenados nos elementos individualmente, selecione as opções de menu View -> Data View para passar para a outra visualização. Em "Data View", são encontrados na coluna "Actual Value" os valores que estão presentemente armazenados.

Address	Name	Type	Initial value	Actual value	Comment
0.0	Measuring_point[1]	REAL	0.000000e+000	1.000000e+002	
4.0	Measuring_point[2]	REAL	0.000000e+000	1.023000e+002	
8.0	Measuring_point[3]	REAL	0.000000e+000	1.035000e+002	
12.0	Measuring_point[4]	REAL	0.000000e+000	1.067000e+002	
16.0	Measuring_point[5]	REAL	0.000000e+000	1.052000e+002	
20.0	Measuring_point[6]	REAL	0.000000e+000	1.050000e+002	
24.0	Measuring_point[7]	REAL	0.000000e+000	1.055000e+002	
28.0	Measuring_point[8]	REAL	0.000000e+000	1.060000e+002	
32.0	Measuring_point[9]	REAL	0.000000e+000	1.079000e+002	
36.0	Measuring_point[10]	REAL	0.000000e+000	1.079900e+002	

Exemplo de uma Estrutura

Motor_data

Operating Speed, data type Integer
Rated Current, data type Real
Startup Current, data type Real
Turning Direction, data type Bool

Estrutura com o nome "Motor_data"
(vários elementos
com diferentes tipos de dados)

Exibição no Editor de Programa (Bloco de Dados DB 1):

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	Motor_Data	STRUCT		
+0.0	RPM	INT	0	
+2.0	Rated_current	REAL	0.000000e+000	
+6.0	Startup_current	REAL	0.000000e+000	
+10.0	Direction_rotation	BOOL	FALSE	
=12.0		END_STRUCT		
=12.0		END_STRUCT		

SIMATIC S7

Siemens Engenharia e Service 2002. Todos os direitos reservados.

Data: 26/08/2011
Arquivo: S7-Bas-08.13

sitrain

Estrutura

A figura mostra o exemplo de uma estrutura com o nome "Motor_data". A estrutura é constituída por vários elementos de diferentes tipos de dados. Os elementos individuais de uma estrutura podem ser do tipo de dados elementar ou complexo.

O acesso aos elementos individuais de uma estrutura contém o nome da estrutura. O programa fica assim mais fácil de ler.

Para poder acessar aos elementos simbolicamente, deve ser dado ao bloco de dados um nome simbólico, por exemplo, "Drive_1".

Exemplos de como acessar elementos individuais de uma estrutura:

```
L "Drive_1".Motor_data.rated_current
L "Drive_1".Motor_data.operating_speed
```

"Drive_1" é o nome simbólico do bloco de dados, que contém a estrutura. O nome da estrutura é dado (separado por um ponto) depois do nome simbólico. A seguir ao nome da estrutura (separado por um ponto) aparece o nome de um elemento da estrutura.

Definir Estrutura num DB

O tipo para uma estrutura é "STRUCT". O fim de uma estrutura é indicado por "END_STRUCT". Cada estrutura tem que ter um nome (no nosso exemplo: "Motor_data").