## Manual de formación

### para soluciones generales en automatización

## Totally Integrated Automation (TIA)

# MÓDULO B4

## Bloques de datos

Este documento fue suministrado por SIEMENS Siemens A&D SCE (Tecnología en Automatización y Accionamientos, Siemens A&D, coopera con la Educación) para formación. Siemens no hace ningún tipo de garantía con respecto a su contenido.

El préstamo o copia de este documento, incluyendo el uso e informe de su contenido, sólo se permite dentro de los centros de formación.

En caso de excepciones se requiere el permiso por escrito de Siemens A&D SCE (Mr. Knust: E-Mail: michael.knust@hvr.siemens.de). Cualquier incumplimiento de estas normas estará sujeto al pago de los posibles perjuicios causados. Todos los derechos quedan reservados para la traducción y posibilidad de patente.

Agradecemos al Ingeniero Michael Dziallas, a los tutores de las escuelas de formación profesional, así como a todas aquellas personas que nos han prestado su colaboración para la elaboración de este documento.

### PÁGINA:

1.	Introducción	4
2.	Notas sobre los Bloques de Datos	6
3.	Generando Bloques de Datos	7

Los símbolos siguientes acceden a los módulos especificados:



### 1. INTRODUCCIÓN



El módulo B4 pertenece al contenido de las Funciones Adicionales de Programación STEP 7.

### Finalidad del Aprendizaje:

En este módulo, el lector aprenderá sobre como utilizar los Bloques de Datos para almacenar información.

- Generando Bloques de Datos
- Specificando la estructura de un Bloque de Datos
- Accediendo a un elemento de un DB en un programa STEP 7

### **Requirements:**

#### **Requisitos:**

Para el correcto aprovechamiento de este módulo, se requieren los siguientes conocimientos:

- Conocimientos de uso de Windows 95/98/2000/ME/NT4.0
- Programación Básica de PLC con STEP 7 (Módulo A3 'Puesta en Marcha' programando PLC con STEP 7)
- Conocimientos Básicos de programación estructurada (Anexo I Programación Básica de PLC Programando con un SIMATIC S7-300)

Indic	e Notas	Generando Bloques de Dato	S

#### Hardware y software Necesarios

- 1 PC, Sistema Operativo Windows 95/98/2000/ME/NT4.0 con
  - Mínimo: 133MHz y 64MB RAM, aprox. 65 MB de espacio libre en disco duro
  - Óptimo: 500MHz y 128MB RAM, aprox. 65 MB de espacio libre en disco duro
- 2 Software STEP 7 V 5.x
- 3 Interfase MPI para PC (p.e. PC- Adapter)
- 4 PLC SIMATIC S7-300 con al menos un módulo de entradas/salidas

Ejemplo de configuración:

- Fuente de Alimentación: PS 307 2A
- CPU: CPU 314
- Entradas Digitales: DI 16x DC24V
- Salidas Digitales: DO 16x DC24V / 0.5 A



Indice	Notas	Generando Bloques de Datos

i

### 2. NOTAS SOBRE LOS BLOQUES DE DATOS

Los Bloques de Datos (DBs) pueden ser utilizados en el programa para salvar información en la CPU. Tienen una capacidad de memoria de hasta 8 KBytes (8192 Bytes).

Existen dos tipos de Bloques de Datos. DBs **Globales**, en los cuales todos los OBs, FBs y FCs pueden guardar o leer datos y DBs **de Instancia**, los cuales se encuentran asignados a un FB en particular.

En los DBs, se pueden almacenar diferentes tipos de datos (p.e. BOOL o WORD ) de manera aleatoria.

La estructura de un DB se genera a través de la herramienta Editor KOP/ AWL//FUP: Programar Bloques de STEP7 .

En la estructura del programa STEP 7, los bloques de datos se encuentran de la siguiente manera:



Indice	Notas	Generando Bloques de Datos

### 3. GENERANDO BLOQUES DE DATOS

1
---

Los Bloques de Datos son generados y abiertos como cualquier otro tipo de bloque en la herramienta **Editor KOP/ AWL//FUP: Programar Bloques.** Sirven para almacenar datos e información de estados.

En el párrafo siguiente, se describe a través de un ejemplo la utilización de un bloque de datos global:



Por esto, los valores se seleccionarán con los interruptores 'S0' a 'S7' y visualizados en un módulo de 'Display' de salidas. La representación numérica del valor a almacenar se representa con estos interruptores, donde el interruptor S7 tendrá el mayor peso del valor y el interruptor S1 el menor peso.

El ejemplo se corresponde con las direcciones siguientes:

Entradas:

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

#### Salidas:

- Display = AW4

Indice

Notas

Para poder generar este ejemplo del programa, deben de los seguirse los siguientes pasos (Por eso el programa se distribuye con la creación de la configuración del hardware):

1. Llamar al Administrador SIMATIC con un doble click ( -> Administrador SIMATIC)



2. Crear un proyecto nuevo ( $\rightarrow$  Archivo $\rightarrow$  Nuevo)

	🛃 Administrador SIMATIC - [Puesta en Marcha D:\Siemens\Step7\S7proj\Puesta_e]				
Ð	<u>Archivo</u> <u>E</u> dición <u>I</u> nsertar <u>S</u> istema de destino <u>V</u> er <u>H</u> erramient	as Ve <u>n</u> tan	a Ay <u>u</u> da		
	Nuevo	Ctrl+N	< sin filtro >	- 70	8
Ē	Asistente 'Nuevo <u>P</u> royecto'				
<b>+</b>	<u>A</u> brir	Ctrl+O			
	Abrir proyecto de la versión 1				
	<u>C</u> errar				
	Memory Card S7	•			
	Archivo Memory Card	•			
	Gyardar como	Ctrl+S			
	Borrar				
	<u>G</u> estionar				
	Archivar				
	Desarchi <u>v</u> ar				
	Imprimir	+			
	Pr <u>e</u> parar página				
	Encabezado/Pie de página				
	Ins <u>t</u> alar impresora				
	1 Puesta en Marcha (Proyecto) D:\\Step7\S7proj\Puesta_e				
	2 Shopturn (Proyecto) D:\Siemens\Step7\S7proj\Shopturn				
	3 SM060306 (Librería) D:\Siemens\Step7\S7libs\ShopMill_636				
	4 Shopmill (Proyecto) D:\Siemens\Step7\S7proj\Shopmill				
 Cre	Salir	Alt+F4			

Indice

Notas



 En el campo Nombre, asignamos el nombre del proyecto Proyecto\_Prueba\_DB. (→ 'Proyecto\_Prueba\_DB' → OK)

Nu	Nuevo 🛛 🔀				
	Proyectos de usuario Librerías				
	Nombre	Ruta 🔺			
	840Di FP50-80 TECNOMAGNETE (ITALIA) ncu_572_4 Proyecto Correa Puesta en Marcha Puigjaner rapidita serra	D:\Siemens\Step7\S7proj\840 D:\Siemens\Step7\S7proj\892 D:\Siemens\Step7\S7proj\892 D:\Proyecto CORREA\PLC\Pr D:\Siemens\Step7\S7proj\Pue D:\Siemens\Step7\S7proj\Pui D:\Proyecto CORREA\PLC\R/ D:\Siemens\Step7\S7proj\serr			
N	lombre:	<u>T</u> ipo:			
F	Proyecto_Prueba_DB	Proyecto 💌			
	Ubicación (ruta) :				
[	D:\Siemens\Step7\S7proj	E <u>x</u> aminar			
	Aceptar	Cancelar Ayuda			

4. Insertar un nuevo **Programa S7** (  $\rightarrow$  Insertar  $\rightarrow$  Programa  $\rightarrow$  Programa S7).



5.

1	
	100
41	(Same
al	City City

Seleccionar la carpeta **Bloques** ( $\rightarrow$  Bloques)

Administrador SIMATIC - [F	Proyecto_Prueba	a_DB D:\Siemer	ns\Step7\S 💶 🗵 🗙
🖹 🛃 Archivo Edición Insertar 🖇	<u>S</u> istema de destino	<u>V</u> er <u>H</u> erramientas	Ve <u>n</u> tana Ay <u>u</u> da
			_ & ×
	🔬 오 💁	<u>р</u> в+ в+ в+	主 < sin filtro >
Proyecto_Prueba_DB	<b>€</b> OB1		
Pulse F1 para obtener ayuda.			

6. Insertar un **Bloque de Datos** (  $\rightarrow$  Insertar  $\rightarrow$  Bloque S7  $\rightarrow$  Bloque de Datos).



	Indice	Notas	Generando Bloques de Datos
_			





7. Introducir el nombre **DB10**, seleccionar la opción DB Global y aceptar con **OK** ( $\rightarrow$  DB10  $\rightarrow$  DB Global -> OK).

Propiedades - Bloque de datos				
General - 1ª parte Gener	al - 2ª parte 🛛 Llamadas 🗍 Atributos	1		
<u>N</u> ombre y tipo:	DB10 DB global	▼ <b>▼</b>		
N <u>o</u> mbre simbólico:				
<u>C</u> omentario del símbolo:				
Lenguaje:	DB			
Ruta del proyecto:				
Ubicación del proyecto:	D:\Siemens\Step7\S7proj\Proyecto			
Fecha de creación:	Código 10/01/2003 09:15:33	Interface		
Última modificación:	10/01/2003 09:15:33	10/01/2003 09:15:33		
Co <u>m</u> entario:		×		
Aceptar		Cancelar Ayuda		

8. Abrir el bloque de datos **DB10** con un doble click( $\rightarrow$  DB10).

Administrador SIMATI	C - [Proyecto_Prueba	_DB D:\Siemen	s\Step7\S 🗖 🗖 🗙
Archivo Edición Inser	tar <u>S</u> istema de destino	<u>V</u> er <u>H</u> erramientas	Ve <u>n</u> tana Ay <u>u</u> da
		<u>D</u> <u>b-</u> <u>b-b-</u>	<pre>sin filtro &gt;</pre>
Proyecto_Prueba_DE	3 <b>(D</b> 0B1	■ DB10	
, Pulse F1 para obtener ayuda.			
dice	Notas	Generar	ndo Bloques de Datos

Nota: Los bloques de datos asignados a un FB se generan automáticamente en la llamada al FB. No tiene sentido, por tanto, generarlos antes de generar el FB. Por otro lado, los bloques de datos asociados a un tipo de datos (UDT) son DBs cuya estructura es una copia exacta del UDT.

Indice

Notas



Cada registro del DB tiene un Nombre, Tipo, Valor Inicial y Comentario (opcional).
En el momento en que se pasa al campo siguiente, se le genera una dirección en le DB que no puede ser alterada.

El bloque de datos será salvado  $\square$  y cargado el PLC  $\square$ . El selector de modo del PLC deberá estar en STOP! ( $\rightarrow$  Nombre $\rightarrow$  Tipo  $\rightarrow$  Valor Inicial  $\rightarrow$  Comentario  $\rightarrow$   $\square \rightarrow$   $\square \rightarrow$  )

<b>IN KOP</b>	/AWL/	FUP - IDB1	) Provecto_Pri	Jeba DB\Progra	ama SZ	111		
	ivo <u>E</u>	dición <u>I</u> nsertar	<u>S</u> istema de destir	no <u>T</u> est <u>V</u> er <u>H</u>	erramier	itas Ve <u>n</u> tana Ay <u>u</u> da		_ 8
	8~	*		- <u>6</u> % 🔬 🗲	. 661	!«»! <b>№?</b>		
Direct	ción H	lombre	Tipo	Valor inic	: Comes	ntario		
	0.0		STRUCT					
	+0.0	Valor0	INT	0	Valo	r asignado a través	del int	terruptor SO
	+2.0	Valorl	WORD	W#16#0	Valo	r asignado a través	del int	terruptor Sl
	+4.0	Valor2	WORD	W#16#0	Valor	r asignado a través	del int	terruptor S2
	+6.0	Valor3	WORD	W#16#0	Valo	r asignado a través	del int	terruptor S3
	+8.0	Valor4	WORD	W#16#0	Valor	r asignado a través	del int	terruptor S4
+	10.0	Valor5	INT	0	Valor	r asignado a través	del int	terruptor S5
+	.2.0	Valore	INT	0	Valor	r asignado a través	del int	terruptor S6
	14.0	Valor7	BOOL	FALSE	Valor	r asignado través	del int	terruptor 37
Pulse F1	Nor con acc dire	nbre simb el que se ede a la ección abs anada	ólico Tip asi oluta tab reç	po de dato gnado (ver pla abajo) al gistro.		Valor inicial en for compatible con e de dato seleccion opcional ).	rmato I tipo hado (	Abs II Comentario para documentación ( opcional ).
27 cada vez o		s asiyilaud s valida un						
ro con ENTER								
no con Livi Er	<b>.</b> . Σοσιό	n nuada a	or					
	50010	in pueue s	ei					
STIE, WORD, .		., .						
lede acceder	tamb	ien a este						
ento a través o	de su	dirección						
luta (después	de h	naber com	pilado					
raado el DB).								



Si el bloque de datos es declarado como de instancia a un FB, al abrir dicho DB ya aparece la tabla con la estructura del DB y no puede modificarse. Dicha estructura es una copia exacta de la tabla de declaraciones del FB.

Indice	Notas



Los datos en un bloque de datos deben de llevar asignado un tipo de datos.

Los tipos de datos estándar STEP 7 se definen en la tabla mostrada abajo :

Тіро у	Tamaño	Formato-	Rango y notación numérica	Ejemplo
descripción	en Bits	Opciones	(Valores máximo y mínimo)	
BOOL (Bit)	1	Texto Booleano	TRUE/FALSE	TRUE
BYTE (Byte)	8	Número Hexadecimal	B#16#0 a B#16#FF	B#16#10
WORD (Palabra)	16	Número Binario	2#0 a 2#1111_1111_1111_1111	2#0001_0000_0000_0000
		Número Hexadecimal	W#16#0 a W#16#FFFF	W#16#1000
		BCD	C#0 a C#999	C#998
		Número Decimal sin signo	B#(0,0) a B#(255,255)	B#(10,20)
DWORD (Doble Palabra)	32	Número Binario	2#0 a 2#1111_1111_1111_1111_1111_1111 1_1111_1111	2#1000_0001_0001_1000_1 011_1011_0111_1111
		Número Hexadecimal	DW#16#0000_0000 a DW#16#FFFF_FFF	DW#16#00A2_1234
		Número Decimal sin signo	B#(0,0,0,0) a B#(255,255,255,255)	B#(1,14,100,120)
INT (Entero)	16	Número Decimal con signo	-32768 a 32767	1
DINT (Int,32 bit)	32	Número Decimal con signo	L#-2147483648 a L#2147483647	L#1
REAL (Número en coma flotante)	32	Número en coma flotante IEEE	Máximo: +/-3.402823e+38 Mínimo: +/-1.175495e-38	1.234567e+13
S5TIME (Tiempo Simatic)	16	Tiempo S7 en pasos de 10 ms	S5T#0H_0M_0S_10MS a S5T#2H_46M_30S_0MS and S5T#0H_0M_0S_0MS	S5T#0H_1M_0S_0MS S5TIME#1H_1M_0S_0MS
TIME (Tiempo IEC)	32	Tiempo IEC en pasos desde 1ms, entero con signo	-T#24D_20H_31M_23S_648MS a T#24D_20H_31M_23S_647MS	T#0D_1H_1M_0S_0MS TIME#0D_1H_1M_0S_0MS
DATE (Fecha IEC)	16	Fecha IEC en pasos de 1 día	D#1990-1-1 a D#2168-12-31	DATE#1994-3-15
TIME_OF_DAY (Fecha y Hora)	32	Tiempo en pasos de 1ms	TOD#0:0:0.0 a TOD#23:59:59.999	TIME_OF_DAY#1:10:3.3
CHAR (Carácter)	8	Caracteres ASCII	´A´, ´B´ etc.	̈́Β΄

Notas



 El contenido de los valores del bloque de datos se puede modificar. No es recomendable hacerlo a través del campo Valor Inicial. Se debe de hacer a través de la opción de menú Datos ( → Ver → Datos).

ļ	💐 KOP/AW	L/FL	JP - [DB10	Proyecto_Prueb	a_DB	١	Programa S7(1)]						
•	🖬 <u>A</u> rchivo	<u>E</u> dici	ión <u>I</u> nsertar	<u>S</u> istema de destino	<u>T</u> est	7	<u>/er H</u> erramientas	Ve <u>n</u> tana	Ay <u>u</u> da				
	0 🖻 🔓		<u></u>		0%		<u>C</u> atálogo Regi <u>s</u> tros de la CP	U	Ctrl+K.				
I	Dirección	Nor	mbre	Tipo	Valo	~	<u> R</u> esultados						
I	0.0	)		STRUCT		1	• <u>K</u> OP		Ctrl+1				
I	+0.0	) V	alor0	INT	0		A <u>w/</u> L		Ctrl+2	1	el	interruptor	S0
I	+2.0	) V	alorl	WORD	₩#16		FUP		Ctrl+3	1	el	interruptor	S1
I	+4.0	) V	alor2	WORD	₩#16		<u>D</u> atos			1	el	interruptor	S2
I	+6.0	) V	alor3	WORD	₩#16		• Declaración			1	el	interruptor	<b>S</b> 3
I	+8.0	) V	alor4	WORD	₩#16		Mostrar			▶ 1	el	interruptor	S4
I	+10.0	) 🛛	alor5	INT	0	-			Chile Marrier	-1	el	interruptor	S5
I	+12.0	) V	alor6	INT	0		Aleiar		Ctrl+Num+	1	el	interruptor	S6
I	+14.0	) V	alor7	BOOL	FALS		Factor de zoom		Carrinan	1	el	interruptor	S7
E						-	- 4 Parra de herramier			-[			
Ī	1: Error 2: Info					Ĭ	Barra de puntos de	e parada		F	-		
 C	Cambia a la vis	sta Da	atos en el blo	que actua		•	• Barra de <u>e</u> stado				_	At	DS

12. Es ahora cuando se puede introducir un nuevo valor en el campo Valor Actual y salvarlo en el disco duro con  $\square$  así como cargarlo en la CPU con  $\square$  ( $\rightarrow$  Valor Actual  $\rightarrow$   $\square$   $\rightarrow$   $\square$ ).

)irección	Nombre	Tipo	Valor inicial	Valor actual	Comentario	
0.0	Valor0	INT	0	0	Valor asignado a travé	з (
2.0	Valorl	WORD	W#16#0	W#16#0	Valor asignado a travé	з (
4.0	Valor2	WORD	W#16#0	W#16#0	Valor asignado a travé	з (
6.0	Valor3	WORD	W#16#0	W#16#0	Valor asignado a travé	з (
8.0	Valor4	WORD	W#16#0	W#16#0	Valor asignado a travé	3 (
10.0	Valor5	INT	0	0	Valor asignado a travé	з (
12.0	Valor6	INT	0	0	Valor asignado a travé	з (
14.0	Valor7	BOOL	FALSE	FALSE	Valor asignado a travé	3 (
14.0	Valor7	BOOL	FALSE	FALSE	Valor asignado a trav	vé:

Indice

Notas



 Para poder acceder a los registros del DB en el programa STEP 7, el OB1 debe ser abierto con un doble click en el icono asociado del Administrador SIMATIC (→ Administrador SIMATIC → OB1).

Administ	rador SI	MATIC - [	Proyect	to_Prueba	_DB D	:\Siemen	s\Step7\	5 <b>_</b> 🗆 🗙
Archivo	<u>E</u> dición	<u>I</u> nsertar	<u>S</u> istema	de destino	⊻er <u>H</u> e	erramientas	Ve <u>n</u> tana	Ay <u>u</u> da
								<u>_ 8 ×</u>
🗋 🗃 🔡		X 🖻 🖬	L 📩		<u>р</u> в- в-	0-0- 0-0- 0-0-	<b>E</b> <	sin filtro >
Proye	ecto_Prue 'rograma ' J Fuente Bloque	eba_DB S7(1) es es		OB1	1	<b>D</b> B10		
Pulse F1 para	obtener a	ayuda.						

14. Aceptamos la pantalla con  $OK (\rightarrow OK)$ .

Propiedades - Bloque de	organización		×		
General - 1ª parte Genera	al - 2ª parte 🛛 Llamadas 🗍 Atributos 🛉				
Nombre:	OB1				
N <u>o</u> mbre simbólico:					
<u>C</u> omentario del símbolo:					
Lenguaje:	AWL				
Ruta del proyecto:					
Ubicación del proyecto:	D:\Siemens\Step7\S7proj\Proyecto				
	Código	Interface			
Fecha de creación: Última modificación:	10/01/2003 09:13:39 07/02/2001 15:03:43	15/02/1996 16:51:12			
Co <u>m</u> entario:	"Main Program Sweep (Cycle)"		×		
Aceptar		Cancelar A	\yuda		

i

### Existen tres posibilidades de acceso a los datos:

#### 1. Acceso por Direccionamiento Directo:

Se puede acceder a un elemento de un DB con las siguientes instrucciones:

Ejemplo:

L	DB 20. DBB2	Cargamos el byte de datos 2 del DB20 en el ACCU 1
L	DB 22. DBW4	Cargamos el byte de datos 4 del DB22 en el ACCU 1
А	DB 2. DBX5.6	Consultamos el estado del bit de datos 5.6 del DB2.

#### 2. Accesso a un elemento de un DB que ya se encontraba abierto:

Para poder acceder a un elemento de datos, primero se debe abrir su correspondiente DB a través de las instrucciones OPN DB o OPN DI. Una vez hecho esto, varios bits de datos (DBX/DIX),bytes de datos (DBB/DIB), palabras de datos (DBW/DIW) o dobles palabras de datos (DBD/DID) pueden ser procesadas en diferentes operaciones binarias o aritméticas.

Al utilizar la instrucción OPN DI se abre un DB de Instancia. Esta instrucción puede utilizarse también con DBs globales, cuando 2 DBs permanecen abiertos al mismo tiempo.

Ejemplo:		
OPN	DB 20	Abrimos el DB20
OPN	DI 22	Abrimos el DB22
L	DBW 0	Cargamos la palabra de datos 0 del DB20 en el ACCU 1
Т	MW 1	Transferimos el contenido del ACCU 1 a la palabra de marcas 1
А	DIX 0.0	Operación AND sobre el bit de datos 0.0 del DB22 con
А	E 1.0	el bit de entradas 1.0
=	A 4.0	Asignamos el resultado al bit de salidas digitales 4.0

#### 3. Acceso a datos de un DB de Instancia a través de la llamada de un Bloque de Función:

Los datos pueden ser transferidos a un DB de Instancia a través de la llamada a su correspondiente bloque de función con la instrucción CALL FB1, DB19. La asignación de las variables, que deberían de haber sido previamente definidas en la tabla de definiciones del FB, y cuyo contenido se almacena en el DB de instancia, son tomadas de las direcciones asignadas a los parámetros del FB (p.e. EW0, M 10.0 o AW4) con la instrucción CALL.

Ejemplo:		
CALL	FB1, DB19	
CONTADOR:=	EW 0	La Variable CONTADOR corresponde con la dirección EW 0.
SALIDA:=	A 4.0	La variable SALIDA corresponde con la dirección A 4.0.



El formato de las variables del DB de Instancia y la dirección asignada en el parámetro del FB deben de ser iguales.

	Indice	Notas	Generando Bloques de Datos
-			



15. Con el Editor KOP/ AWL//FUP: Programar Bloques, se dispone de una herramienta para generar el programa STEP 7.

La selección del elemento de datos se realiza a través de los diferentes interruptores, como se muestra en la imagen de abajo.

Cuando se hayan contemplado todas las posibilidades de selección con los interruptores (S0 a

S7) en el OB1, se deberá guardar	en el disco duro 🔛 y carga	ado en el PLC 🕮. El interruptor de
modo de la CPU deberá estar en S	STOP! ( $\rightarrow$ $\square$ $\rightarrow$ $\square$ )	

KOP/AWL/FUP - [OB1 Proyecto_Prueba_DB\Programa S7(1)]			_ 🗆 ×			
<u>Archivo</u> <u>Edición</u> <u>Insertar</u> <u>Sistema</u> de destino <u>T</u> est <u>Ver</u> <u>H</u> erramientas	Ve <u>n</u> tana	Ay <u>u</u> da	_ 8 ×			
OB1 : Contenido del DB10 transferido a las salidas			-			
Comentario:						
Segm. 1: Valor seleccionado a través del interruptor 30						
Comentario:						
UN E 0.0						
3PA m00						
L DB10.DBW 0						
T AW 4						
m00: MOP 0						
Segm. 2: Valor seleccionado a través del interruptor 31						
Comentario:						
UN E 0.1						
3PA m01						
L DB10.DBW 2						
T AW 4						
m01: NOP 0						
			<b>_</b>			
Idea NNN 1: Energy 2: Inter (						
Pulse F1 para obtener ayuda.	9	off	ine //			

16. Cuando el selector de modo del PLC se ponga en RUN, el programa comienza su ejecución. Activando sólo uno de los interruptores S0 a S7, el valor almacenado en el correspondiente registro del bloque de datos es visualizado en el módulo de salidas a través de la dirección AW4.

I	ndice	Notas	Generando Bloques de Datos