Siemens S5 (AS511) Driver Help

Table of Contents

Siemens S5 (AS511) Driver Help

Help version 1.015

Overview

The Siemens S5 (AS511) Driver provides an easy and reliable way to connect Siemens S5 (AS511) devices to OPC Client applications, including HMI, SCADA, Historian, MES, ERP and countless custom applications. It is intended for use with Siemens S5 PLCs communicating via the front programming port using AS511 protocol (which is specific for each Siemens device). This driver has been designed to operate with a set range of Siemens equipment: it is not recommended for use on devices that are not supported.

The Siemens S5 PLC family has a unique memory structure. Data within the PLC is not at fixed locations within the PLC's memory space. As the PLC logic is created and modified, this memory space is continuously updated and revised. When these revisions occur, the location of the key data elements (such as flags, timers, counters, I/O, and data blocks) can move around in the PLC's memory. The Siemens S5 (AS511) Driver has been designed to read the location of these memory elements when the driver begins operation or detects a communications error. If the PLC configuration changes, users must restart the Siemens S5 (AS511) Driver or pull and replace the cable connection. Both of these actions will cause the Siemens S5 (AS511) Driver to reacquire the location of all PLC memory elements.

Device Setup

Supported Devices

Siemens S5-90U
Siemens S5-95U
Siemens S5-100U-100
Siemens S5-100U-101
Siemens S5-100U-103
Siemens S5-101U
Siemens S5-115U-941
Siemens S5-115U-942
Siemens S5-115U-943
Siemens S5-115U-944
Siemens S5-115U-945
Siemens S5-135U-921
Siemens S5-135U-922
Siemens S5-135U-928
Siemens S5-155U-946
Siemens S5-155U-947

Communication Protocol

AS511 Current Loop

Supported Communication Parameters

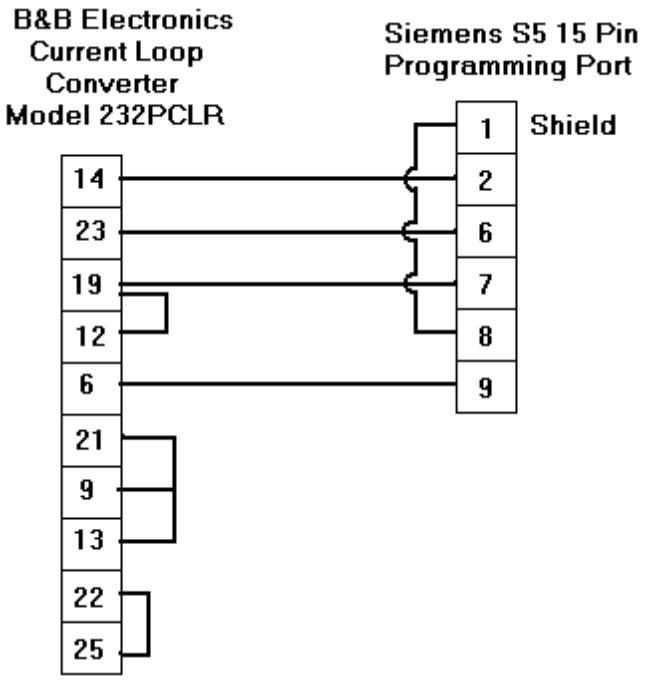
Baud: 9600 (Fixed)
Parity: Even (Fixed)
Data Bits: 8 (Fixed)
Stop Bit: 1 (Fixed)

Ethernet Encapsulation

This driver supports Ethernet Encapsulation, which allows communications with serial devices attached to an Ethernet network using a terminal server or device server. It may be invoked through the COM ID dialog in Channel Properties. When used directly with a serial port, this driver only supports a single connection to a single controller per serial port. When operating in Ethernet Encapsulation Mode, the driver will support up to 30 controllers per channel. In this mode, a single controller can be paired with a terminal server/device server to form a single node. For more information, refer to the server's help documentation.

Note: The Siemens S5 AS511 protocol is sensitive to timing and gaps in the communications stream. If the network experiences heavy packet loss or delay while using Ethernet Encapsulation, the Siemens S5 (AS511) Driver may report a large number of timeout errors or be unable to communicate. In some cases, using a switched network can help reduce these delays; however, it is not a guaranteed solution.

Cable Connections



Modem Setup

This driver supports modem functionality. For more information, please refer to the topic "Modem Support" in the OPC Server Help documentation.

Data Types Description

Data Type	Description
Boolean	Single bit of an 8 bit value*
Byte	Unsigned 8 bit value
Word	Unsigned 16 bit value
Short	Signed 16 bit value
DWord	Unsigned 32 bit value
Long	Signed 32 bit value
Float	32 bit floating point value The driver interprets two consecutive registers as a floating-point value by making the second register the high word and the first register the low word.
String	Null terminated ASCII string

*For more information, refer to **Address Descriptions**.

Address Descriptions

Address specifications vary depending on the model in use. Select a link from the following list to obtain specific address information for the model of interest.

Siemens S5 (AS511) 90U Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB127	**Byte**	Read/Write
	IW0-IW126	**Word**, Short	Read/Write
	ID0-ID124	**DWord**, Long	Read/Write
Discrete Inputs	E0.b-E127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB127	**Byte**	Read/Write
	EW0-EW126	**Word**, Short	Read/Write
Note: I and E access the same memory area.	ED0-ED124	**DWord**, Long	Read/Write
Discrete Outputs	Q0.b-Q127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	QB0-QB127	**Byte**	Read/Write
	QW0-QW126	**Word**, Short	Read/Write
	QD0-QD124	**DWord**, Long	Read/Write
Discrete Outputs	A0.b-A127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	AB0-AB127	**Byte**	Read/Write
	AW0-AW126	**Word**, Short	Read/Write
Note: Q and A access the same memory area	AD0-AD124	**DWord**, Long	Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7	**Boolean**	Read/Write
	FB0-FB255	**Byte**	Read/Write

	FW0-FW254 FD0-FD252	**Word**, Short **DWord**, Long	Read/Write Read/Write
Internal Memory **Note:** F and M access the same memory area.	M0.b-M255.b .b is Bit Number 0-7 MB0-MB255 MW0-MW254 MD0-MD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Data Block Boolean	DB1-N**:**KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N**:**KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N**:**KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N**:**KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N**:**KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N**:**KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N**:**KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write
Data Block Timer	DB1-N**:**KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N**:**KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write
Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 95U Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB127	**Byte**	Read/Write
	IW0-IW126	**Word**, Short	Read/Write
	ID0-ID124	**DWord**, Long	Read/Write
Discrete Inputs	E0.b-E127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB127	**Byte**	Read/Write
	EW0-EW126	**Word**, Short	Read/Write
Note: I and E access the same memory area.	ED0-ED124	**DWord**, Long	Read/Write
Discrete Outputs	Q0.b-Q127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	QB0-QB127	**Byte**	Read/Write
	QW0-QW126	**Word**, Short	Read/Write
	QD0-QD124	**DWord**, Long	Read/Write
Discrete Outputs	A0.b-A127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	AB0-AB127	**Byte**	Read/Write

Note: Q and A access the same memory area.	AW0-AW126 AD0-AD124	**Word**, Short **DWord**, Long	Read/Write Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7 FB0-FB255 FW0-FW254 FD0-FD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory **Note:** F and M access the same memory area.	M0.b-M255.b .b is Bit Number 0-7 MB0-MB255 MW0-MW254 MD0-MD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Data Block Boolean	DB1-N**:**KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N**:**KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N**:**KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N**:**KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N**:**KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N**:**KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N**:**KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write
Data Block Timer	DB1-N**:**KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N**:**KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write
Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 100U-100 Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB127	**Byte**	Read/Write
	IW0-IW126	**Word**, Short	Read/Write
	ID0-ID124	**DWord**, Long	Read/Write
Discrete Inputs	E0.b-E127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB127	**Byte**	Read/Write
	EW0-EW126	**Word**, Short	Read/Write
Note: I and E access the same memory area.	ED0-ED124	**DWord**, Long	Read/Write
Discrete Outputs	Q0.b-Q127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	QB0-QB127	**Byte**	Read/Write

	QW0-QW126 QD0-QD124	**Word**, Short **DWord**, Long	Read/Write Read/Write
Discrete Outputs **Note:** Q and A access the same memory area.	A0.b-A127.b .b is Bit Number 0-7 AB0-AB127 AW0-AW126 AD0-AD124	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7 FB0-FB255 FW0-FW254 FD0-FD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory **Note:** F and M access the same memory area.	M0.b-M255.b .b is Bit Number 0-7 MB0-MB255 MW0-MW254 MD0-MD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Data Block Boolean	DB1-N:KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N:KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N:KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N:KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N:KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N:KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N:KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1–N:KS0.l–KS255.l 1–N is Block Number l is String Length (2–254)	**String**	Read/Write
Data Block Timer	DB1-N:KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N:KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write

Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 100U-101 Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB127	**Byte**	Read/Write
	IW0-IW126	**Word**, Short	Read/Write
	ID0-ID124	**DWord**, Long	Read/Write
Discrete Inputs	E0.b-E127.b .b is Bit Number 0-7	**Boolean**	Read/Write

Note: I and E access the same memory area.	EB0-EB127 EW0-EW126 ED0-ED124	**Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write
Discrete Outputs	Q0.b-Q127.b .b is Bit Number 0-7 QB0-QB127 QW0-QW126 QD0-QD124	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Discrete Outputs **Note:** Q and A access the same memory area.	A0.b-A127.b .b is Bit Number 0-7 AB0-AB127 AW0-AW126 AD0-AD124	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7 FB0-FB255 FW0-FW254 FD0-FD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory **Note:** F and M access the same memory area.	M0.b-M255.b .b is Bit Number 0-7 MB0-MB255 MW0-MW254 MD0-MD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Data Block Boolean	DB1-N:KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N:KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N:KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N:KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N:KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N:KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N:KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number	**String**	Read/Write

	l is String Length (2-254)		
Data Block Timer	DB1-N**:**KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N**:**KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write
Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 100U-103 Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I127.b .b is Bit Number 0-7	**Boolean**	Read/Write

	IB0-IB127	**Byte**	Read/Write
	IW0-IW126	**Word**, Short	Read/Write
	ID0-ID124	**DWord**, Long	Read/Write
Discrete Inputs **Note:** I and E access the same memory area.	E0.b-E127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB127	**Byte**	Read/Write
	EW0-EW126	**Word**, Short	Read/Write
	ED0-ED124	**DWord**, Long	Read/Write
Discrete Outputs	Q0.b-Q127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	QB0-QB127	**Byte**	Read/Write
	QW0-QW126	**Word**, Short	Read/Write
	QD0-QD124	**DWord**, Long	Read/Write
Discrete Outputs **Note:** Q and A access the same memory area.	A0.b-A127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	AB0-AB127	**Byte**	Read/Write
	AW0-AW126	**Word**, Short	Read/Write
	AD0-AD124	**DWord**, Long	Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7	**Boolean**	Read/Write
	FB0-FB255	**Byte**	Read/Write
	FW0-FW254	**Word**, Short	Read/Write
	FD0-FD252	**DWord**, Long	Read/Write
Internal Memory **Note:** F and M access the same memory area.	M0.b-M255.b .b is Bit Number 0-7	**Boolean**	Read/Write
	MB0-MB255	**Byte**	Read/Write
	MW0-MW254	**Word**, Short	Read/Write
	MD0-MD252	**DWord**, Long	Read/Write
Data Block Boolean	DB1-N**:**KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N**:**KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N**:**KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N**:**KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write

Data Block Signed Word	DB1-N:KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N:KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N:KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1–N:KS0.–KS255.l 1–N is Block Number l is String Length (2–254)	**String**	Read/Write
Data Block Timer	DB1-N:KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N:KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write
Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be

used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 101U Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB127	**Byte**	Read/Write
	IW0-IW126	**Word**, Short	Read/Write
	ID0-ID124	**DWord**, Long	Read/Write
Discrete Inputs	E0.b-E127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB127	**Byte**	Read/Write
	EW0-EW126	**Word**, Short	Read/Write
Note: I and E access the same memory area.	ED0-ED124	**DWord**, Long	Read/Write
Discrete Outputs	Q0.b-Q127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	QB0-QB127	**Byte**	Read/Write
	QW0-QW126	**Word**, Short	Read/Write
	QD0-QD124	**DWord**, Long	Read/Write
Discrete Outputs	A0.b-A127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	AB0-AB127	**Byte**	Read/Write
	AW0-AW126	**Word**, Short	Read/Write
Note: Q and A access the same memory area.	AD0-AD124	**DWord**, Long	Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7	**Boolean**	Read/Write
	FB0-FB255	**Byte**	Read/Write
	FW0-FW254	**Word**, Short	Read/Write
	FD0-FD252	**DWord**, Long	Read/Write
Internal Memory	M0.b-M255.b .b is Bit Number 0-7	**Boolean**	Read/Write
	MB0-MB255	**Byte**	Read/Write
	MW0-MW254	**Word**, Short	Read/Write
Note: F and M access the same memory area.	MD0-MD252	**DWord**, Long	Read/Write
Data Block Boolean	DB1-N**:**KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block	DB1-N**:**KL0-KL255	**Byte**	Read/Write

Left Byte	1-N is Block Number		
Data Block Right Byte	DB1-N:KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N:KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N:KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N:KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N:KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write
Data Block Timer	DB1-N:KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N:KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write
Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the

Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters
Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings
String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 115U-941 Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB127	**Byte**	Read/Write
	IW0-IW126	**Word**, Short	Read/Write
	ID0-ID124	**DWord**, Long	Read/Write
Discrete Inputs	E0.b-E127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB127	**Byte**	Read/Write
	EW0-EW126	**Word**, Short	Read/Write
Note: I and E access the same memory area.	ED0-ED124	**DWord**, Long	Read/Write
Discrete Outputs	Q0.b-Q127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	QB0-QB127	**Byte**	Read/Write
	QW0-QW126	**Word**, Short	Read/Write
	QD0-QD124	**DWord**, Long	Read/Write
Discrete Outputs	A0.b-A127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	AB0-AB127	**Byte**	Read/Write
	AW0-AW126	**Word**, Short	Read/Write
Note: Q and A access the same memory area.	AD0-AD124	**DWord**, Long	Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7	**Boolean**	Read/Write
	FB0-FB255	**Byte**	Read/Write
	FW0-FW254	**Word**, Short	Read/Write
	FD0-FD252	**DWord**, Long	Read/Write
Internal Memory	M0.b-M255.b .b is Bit Number 0-7	**Boolean**	Read/Write
	MB0-MB255	**Byte**	Read/Write

Note: F and M access the same memory area.	MW0-MW254 MD0-MD252	**Word**, Short **DWord**, Long	Read/Write Read/Write
Data Block Boolean	DB1-N**:**KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N**:**KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N**:**KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N**:**KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N**:**KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N**:**KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N**:**KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write
Data Block Timer	DB1-N**:**KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N**:**KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write
Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 115U-942 Address Descriptions

Default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB127	**Byte**	Read/Write
	IW0-IW126	**Word**, Short	Read/Write
	ID0-ID124	**DWord**, Long	Read/Write
Discrete Inputs	E0.b-E127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB127	**Byte**	Read/Write
	EW0-EW126	**Word**, Short	Read/Write
Note: I and E access the same memory area.	ED0-ED124	**DWord**, Long	Read/Write
Discrete Outputs	Q0.b-Q127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	QB0-QB127	**Byte**	Read/Write
	QW0-QW126	**Word**, Short	Read/Write
	QD0-QD124	**DWord**, Long	Read/Write
Discrete Outputs	A0.b-A127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	AB0-AB127	**Byte**	Read/Write
	AW0-AW126	**Word**, Short	Read/Write
Note: Q and A access the same memory area	AD0-AD124	**DWord**, Long	Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7	**Boolean**	Read/Write

	FB0-FB255 FW0-FW254 FD0-FD252	**Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write
Internal Memory **Note:** F and M access the same memory area	M0.b-M255.b .b is Bit Number 0-7 MB0-MB255 MW0-MW254 MD0-MD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Data Block Boolean	DB1-N**:**KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N**:**KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N**:**KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N**:**KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N**:**KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N**:**KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N**:**KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write
Data Block Timer	DB1-N**:**KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N**:**KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write
Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers
The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters
Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings
String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 115U-943 Address Descriptions

Default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB127	**Byte**	Read/Write
	IW0-IW126	**Word**, Short	Read/Write
	ID0-ID124	**DWord**, Long	Read/Write
Discrete Inputs **Note:** I and E access the same memory area.	E0.b-E127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB127	**Byte**	Read/Write
	EW0-EW126	**Word**, Short	Read/Write
	ED0-ED124	**DWord**, Long	Read/Write
Discrete Outputs	Q0.b-Q127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	QB0-QB127	**Byte**	Read/Write
	QW0-QW126	**Word**, Short	Read/Write
	QD0-QD124	**DWord**, Long	Read/Write
Discrete Outputs	A0.b-A127.b	**Boolean**	Read/Write

Note: Q and A access the same memory area.	.b is Bit Number 0-7 AB0-AB127 AW0-AW126 AD0-AD124	 **Byte** **Word**, Short **DWord**, Long	 Read/Write Read/Write Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7 FB0-FB255 FW0-FW254 FD0-FD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory **Note:** F and M access the same memory area.	M0.b-M255.b .b is Bit Number 0-7 MB0-MB255 MW0-MW254 MD0-MD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Data Block Boolean	DB1-N:KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N:KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N:KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N:KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N:KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N:KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N:KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write
Data Block Timer	DB1-N:KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N:KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write
Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 115U-944 Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB127	**Byte**	Read/Write
	IW0-IW126	**Word**, Short	Read/Write
	ID0-ID124	**DWord**, Long	Read/Write
Discrete Inputs	E0.b-E127.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB127	**Byte**	Read/Write
	EW0-EW126	**Word**, Short	Read/Write
Note: I and E access the same memory area.	ED0-ED124	**DWord**, Long	Read/Write

Discrete Outputs	Q0.b-Q127.b .b is Bit Number 0-7 QB0-QB127 QW0-QW126 QD0-QD124	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Discrete Outputs **Note:** Q and A access the same memory area.	A0.b-A127.b .b is Bit Number 0-7 AB0-AB127 AW0-AW126 AD0-AD124	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7 FB0-FB255 FW0-FW254 FD0-FD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory **Note:** F and M access the same memory area.	M0.b-M255.b .b is Bit Number 0-7 MB0-MB255 MW0-MW254 MD0-MD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Data Block Boolean	DB1-N:KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N:KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N:KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N:KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N:KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N:KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N:KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write

Data Block Timer	DB1-N:KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N:KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write
Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 115U-945 Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I127.b .b is Bit Number 0-7 IB0-IB127	**Boolean** **Byte**	Read/Write Read/Write

	IW0-IW126 ID0-ID124	**Word**, Short **DWord**, Long	Read/Write Read/Write
Discrete Inputs **Note:** I and E access the same memory area.	E0.b-E127.b .b is Bit Number 0-7 EB0-EB127 EW0-EW126 ED0-ED124	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Discrete Outputs	Q0.b-Q127.b .b is Bit Number 0-7 QB0-QB127 QW0-QW126 QD0-QD124	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Discrete Outputs **Note:** Q and A access the same memory area.	A0.b-A127.b .b is Bit Number 0-7 AB0-AB127 AW0-AW126 AD0-AD124	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7 FB0-FB255 FW0-FW254 FD0-FD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory **Note:** F and M access the same memory area.	M0.b-M255.b .b is Bit Number 0-7 MB0-MB255 MW0-MW254 MD0-MD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Data Block Boolean	DB1-N:KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Right Byte	DB1-N:KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N:KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N:KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N:KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N:KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write

Data Block Float	DB1-N:KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write
Data Block Timer	DB1-N:KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N:KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write
Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 135U-921 Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB511	**Byte**	Read/Write
	IW0-IW510	**Word**, Short	Read/Write
	ID0-ID508	**DWord**, Long	Read/Write
Discrete Inputs	E0.b-E511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB511	**Byte**	Read/Write
	EW0-EW510	**Word**, Short	Read/Write
Note: I and E access the same memory area.	ED0-ED508	**DWord**, Long	Read/Write
Discrete Outputs	Q0.b-Q511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	QB0-QB511	**Byte**	Read/Write
	QW0-QW510	**Word**, Short	Read/Write
	QD0-QD508	**DWord**, Long	Read/Write
Discrete Outputs	A0.b-A511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	AB0-AB511	**Byte**	Read/Write
	AW0-AW510	**Word**, Short	Read/Write
Note: Q and A access the same memory area.	AD0-AD508	**DWord**, Long	Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7	**Boolean**	Read/Write
	FB0-FB255	**Byte**	Read/Write
	FW0-FW254	**Word**, Short	Read/Write
	FD0-FD252	**DWord**, Long	Read/Write
Internal Memory	M0.b-M255.b .b is Bit Number 0-7	**Boolean**	Read/Write
	MB0-MB255	**Byte**	Read/Write
	MW0-MW254	**Word**, Short	Read/Write
Note: F and M access the same memory area.	MD0-MD252	**DWord**, Long	Read/Write
Data Block Boolean	DB1-N:KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N:KL0-KL255 1-N is Block Number	**Byte**	Read/Write

Data Block Right Byte	DB1-N:KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N:KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N:KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N:KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N:KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write
Data Block Timer	DB1-N:KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N:KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write
Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 135U-922 Address Descriptions

Default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB511	**Byte**	Read/Write
	IW0-IW510	**Word**, Short	Read/Write
	ID0-ID508	**DWord**, Long	Read/Write
Discrete Inputs	E0.b-E511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB511	**Byte**	Read/Write
	EW0-EW510	**Word**, Short	Read/Write
Note: I and E access the same memory area.	ED0-ED508	**DWord**, Long	Read/Write
Discrete Outputs	Q0.b-Q511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	QB0-QB511	**Byte**	Read/Write
	QW0-QW510	**Word**, Short	Read/Write
	QD0-QD508	**DWord**, Long	Read/Write
Discrete Outputs	A0.b-A511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	AB0-AB511	**Byte**	Read/Write
	AW0-AW510	**Word**, Short	Read/Write
Note: Q and A access the same memory area.	AD0-AD508	**DWord**, Long	Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7	**Boolean**	Read/Write
	FB0-FB255	**Byte**	Read/Write
	FW0-FW254	**Word**, Short	Read/Write
	FD0-FD252	**DWord**, Long	Read/Write
Internal Memory	M0.b-M255.b .b is Bit Number 0-7	**Boolean**	Read/Write
	MB0-MB255	**Byte**	Read/Write
	MW0-MW254	**Word**, Short	Read/Write
Note: F and M access the same memory area.	MD0-MD252	**DWord**, Long	Read/Write

Data Block Boolean	DB1-N:KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N:KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N:KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N:KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N:KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N:KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N:KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write
Data Block Timer	DB1-N:KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N:KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T127	**Long**	Read/Write
Counter Current Values	C0-C127	**Word**, Short	Read/Write
Counter Current Values	Z0-Z127	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers
The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters
Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings
String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 135U-928 Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB511	**Byte**	Read/Write
	IW0-IW510	**Word**, Short	Read/Write
	ID0-ID508	**DWord**, Long	Read/Write
Discrete Inputs **Note:** I and E access the same memory area.	E0.b-E511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB511	**Byte**	Read/Write
	EW0-EW510	**Word**, Short	Read/Write
	ED0-ED508	**DWord**, Long	Read/Write
Discrete Outputs	Q0.b-Q511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	QB0-QB511	**Byte**	Read/Write
	QW0-QW510	**Word**, Short	Read/Write
	QD0-QD508	**DWord**, Long	Read/Write
Discrete Outputs **Note:** Q and A access the same memory area.	A0.b-A511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	AB0-AB511	**Byte**	Read/Write
	AW0-AW510	**Word**, Short	Read/Write
	AD0-AD508	**DWord**, Long	Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7	**Boolean**	Read/Write
	FB0-FB255	**Byte**	Read/Write
	FW0-FW254	**Word**, Short	Read/Write
	FD0-FD252	**DWord**, Long	Read/Write

Internal Memory **Note:** F and M access the same memory area.	M0.b-M255.b .b is Bit Number 0-7 MB0-MB255 MW0-MW254 MD0-MD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Data Block Boolean	DB1-N:KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N:KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N:KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N:KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N:KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N:KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N:KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write
Data Block Timer	DB1-N:KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N:KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T255	**Long**	Read/Write
Counter Current Values	C0-C255	**Word**, Short	Read/Write
Counter Current Values	Z0-Z255	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 155U-946 Address Descriptions

Default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB511	**Byte**	Read/Write
	IW0-IW510	**Word**, Short	Read/Write
	ID0-ID508	**DWord**, Long	Read/Write
Discrete Inputs **Note:** I and E access the same memory area.	E0.b-E511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	EB0-EB511	**Byte**	Read/Write
	EW0-EW510	**Word**, Short	Read/Write
	ED0-ED508	**DWord**, Long	Read/Write
Discrete Outputs	Q0.b-Q511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	QB0-QB511	**Byte**	Read/Write
	QW0-QW510	**Word**, Short	Read/Write
	QD0-QD508	**DWord**, Long	Read/Write

Discrete Outputs **Note:** Q and A access the same memory area.	A0.b-A511.b .b is Bit Number 0-7 AB0-AB511 AW0-AW510 AD0-AD508	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7 FB0-FB255 FW0-FW254 FD0-FD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory **Note:** F and M access the same memory area.	M0.b-M255.b .b is Bit Number 0-7 MB0-MB255 MW0-MW254 MD0-MD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Data Block Boolean	DB1-N:KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N:KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N:KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N:KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N:KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N:KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write
Data Block Float	DB1-N:KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write
Data Block Timer	DB1-N:KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N:KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T255	**Long**	Read/Write
Counter Current Values	C0-C255	**Word**, Short	Read/Write
Counter Current Values	Z0-Z255	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Siemens S5 (AS511) 155U-947 Address Descriptions

The default data types for dynamically defined tags are shown in **bold**.

Address Type	Range	Type	Access
Discrete Inputs	I0.b-I511.b .b is Bit Number 0-7	**Boolean**	Read/Write
	IB0-IB511	**Byte**	Read/Write
	IW0-IW510	**Word**, Short	Read/Write
	ID0-ID508	**DWord**, Long	Read/Write

Discrete Inputs **Note:** I and E access the same memory area.	E0.b-E511.b .b is Bit Number 0-7 EB0-EB511 EW0-EW510 ED0-ED508	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Discrete Outputs	Q0.b-Q511.b .b is Bit Number 0-7 QB0-QB511 QW0-QW510 QD0-QD508	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Discrete Outputs **Note:** Q and A access the same memory area.	A0.b-A511.b .b is Bit Number 0-7 AB0-AB511 AW0-AW510 AD0-AD508	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory	F0.b-F255.b .b is Bit Number 0-7 FB0-FB255 FW0-FW254 FD0-FD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Internal Memory **Note:** F and M access the same memory area.	M0.b-M255.b .b is Bit Number 0-7 MB0-MB255 MW0-MW254 MD0-MD252	**Boolean** **Byte** **Word**, Short **DWord**, Long	Read/Write Read/Write Read/Write Read/Write
Data Block Boolean	DB1-N:KM0.b-KM255.b 1-N is Block Number .b is Bit Number 0-15	**Boolean**	Read/Write
Data Block Left Byte	DB1-N:KL0-KL255 1-N is Block Number	**Byte**	Read/Write
Data Block Right Byte	DB1-N:KR0-KR255 1-N is Block Number	**Byte**	Read/Write
Data Block Unsigned Word	DB1-N:KH0-KH255 1-N is Block Number	**Word**, Short	Read/Write
Data Block Signed Word	DB1-N:KF0-KF255 1-N is Block Number	**Short**, Word	Read/Write
Data Block Signed Long	DB1-N:KD0-KD254 1-N is Block Number	**Long**, DWord	Read/Write

Data Block Float	DB1-N:KG0-KG254 1-N is Block Number	**Float**	Read/Write
Data Block String	DB1-N:KS0.l-KS255.l 1-N is Block Number l is String Length (2-254)	**String**	Read/Write
Data Block Timer	DB1-N:KT0-KT255 1-N is Block Number	**Long**	Read/Write
Data Block Counter	DB1-N:KC0-KC255 1-N is Block Number	**Word**, Short	Read/Write
Timer Current Values	T0-T255	**Long**	Read/Write
Counter Current Values	C0-C255	**Word**, Short	Read/Write
Counter Current Values	Z0-Z255	**Word**, Short	Read/Write

All offsets for memory types I, Q, and F represent a byte starting location within the specified memory type.

Examples

1. To access bit 3 of Internal Memory F20, declare an address as follows:
F20.3

2. To access Data Block 5 as word memory at element 30, declare an address as follows:
DB5:KH30

3. To access Data Block 2 element 20 and bit 7, declare an address as follows:
DB2:KM20.7

4. To access Data Block 1 as left byte memory at element 10, declare an address as follows:
DB1:KL10

5. To access Internal Memory F20 as a DWORD, declare an address as follows:
FD20

6. To access Input Memory I10 as a Word, declare an address as follows:
IW10

Note: Use caution when modifying Word, Short, DWord, and Long types. For I, Q, and F each address starts at a byte offset within the device. Therefore, Words FW0 and FW1 overlap at byte 1. Writing to FW0 will also modify the value held in FW1. Similarly, DWord, and Long types can also overlap. It is recommended that these memory types be used so that overlapping does not occur. For example, when using DWords, use FD0, FD4, FD8 ... and so on to prevent overlapping bytes.

Timers

The Siemens S5 (AS511) Driver automatically scales T and KT values based on the Siemens S5 time format. The value returned for either a T or KT memory type will already be scaled using the appropriate Siemens time base. As a result, the values are always returned as a count of milliseconds. When writing to T or KT memory types, the Siemens time base will also be applied. To write a value to a timer in the controller, simply write the desired value as a count of milliseconds to the appropriate timer.

Counters

Counters are stored as three BCD digits on the device. The largest value that can be read or written to a counter is 999.

Strings

String data is stored in data block registers, thus the actual number of bytes used to store the data is an even number. For example, if a string of length 5 is specified, say by DB11:KS1.5, then 3 registers (6 bytes) will be used to store the string data. When writing strings shorter than the maximum specified length (5 in this example), a null terminator (0x00) will be added to the end of the string. When strings are read, the full range of registers are read (3 in this example). Use of string tags with overlapping address ranges should be avoided due to the effects of the null terminators.

Error Descriptions

The following error/warning messages may be generated. Click on the link for a description of the message.

Address Validation

Address '<address>' is out of range for the specified device or register
Array support is not available for the specified address: '<address>'
Data Type '<type>' is not valid for device address '<address>'
Device address '<address>' contains a syntax error
Device address '<address>' is not supported by model '<model name>'
Missing address

Serial Communications

Communications error on '<channel name>' [<error mask>]
COMn does not exist
COMn is in use by another application
Error opening COMn
Unable to set comm parameters on COMn

Device Status Messages

Device '<device name>' is not responding
Unable to write to '<address>' on device '<device name>'

Driver Warning Messages

Data Block DB '<block number>' not defined in '<device name>' write operation has failed
Failure reading device '<device name>' configuration
Protocol Error-Number of bytes received = '<num bytes>' Expected = '<num bytes>'
Requested Data Block DB'<block number>' not defined in '<device name>' block has been disabled

Address Validation

The following error/warning messages may be generated. Click on the link for a description of the message.

Address Validation

Address '<address>' is out of range for the specified device or register
Array support is not available for the specified address: '<address>'
Data Type '<type>' is not valid for device address '<address>'
Device address '<address>' contains a syntax error
Device address '<address>' is not supported by model '<model name>'
Missing address

Array support is not available for the specified address: '<address>'

Error Type:
Warning

Possible Cause:
A tag address that has been specified dynamically contains an array reference for an address type that doesn't support arrays.

Solution:
Re-enter the address in the client application to remove the array reference or correct the address type.

Address '<address>' is out of range for the specified device or register

Error Type:
Warning

Possible Cause:
A tag address that has been specified dynamically references a location that is beyond the range of supported locations for the device.

Solution:
Verify the address is correct; if it is not, re-enter it in the client application.

Data Type '<type>' is not valid for device address '<address>'

Error Type:
Warning

Possible Cause:
A tag address that has been specified dynamically has been assigned an invalid data type.

Solution:
Modify the requested data type in the client application.

Device address '<address>' is not supported by model '<model name>'

Error Type:
Warning

Possible Cause:
A tag address that has been specified dynamically references a location that is valid for the communications protocol but not supported by the target device.

Solution:
Verify that the address is correct; if it is not, re-enter it in the client application. Also verify that the selected model name for the device is correct.

Device address '<address>' contains a syntax error

Error Type:
Warning

Possible Cause:
A tag address that has been specified dynamically via DDE contains one or more invalid characters.

Solution:
Re-enter the address in the client application.

Missing address

Error Type:
Warning

Possible Cause:
A tag address that has been specified dynamically has no length.

Solution:
Re-enter the address in the client application.

Serial Communications

The following error/warning messages may be generated. Click on the link for a description of the message.

Serial Communications
Communications error on '<channel name>' [<error mask>]
COMn does not exist
COMn is in use by another application
Error opening COMn
Unable to set comm parameters on COMn

Communications error on '<channel name>' [<error mask>]

Error Type:
Serious

Error Mask Definitions:
B = Hardware break detected.

F = Framing error.
E = I/O error.
O = Character buffer overrun.
R = RX buffer overrun.
P = Received byte parity error.
T = TX buffer full.

Possible Cause:
1. The serial connection between the device and the Host PC is bad.
2. The communications parameters for the serial connection are incorrect.

Solution:
1. Verify the cabling between the PC and the device.
2. Verify that the specified communications parameters match those of the device.

COMn does not exist

Error Type:
Fatal

Possible Cause:
The specified COM port is not present on the target computer.

Solution:
Verify that the proper COM port has been selected.

COMn is in use by another application

Error Type:
Fatal

Possible Cause:
The serial port assigned to a device is being used by another application.

Solution:
Verify that the correct port has been assigned to the channel.

Error opening COMn

Error Type:
Fatal

Possible Cause:
The specified COM port could not be opened due an internal hardware or software problem on the target computer.

Solution:
Verify that the COM port is functional and may be accessed by other Windows applications.

Unable to set comm parameters on COMn

Error Type:
Fatal

Possible Cause:
The serial parameters for the specified COM port are not valid.

Solution:
Verify the serial parameters and make any necessary changes.

Device Status Messages

The following error/warning messages may be generated. Click on the link for a description of the message.

Device Status Messages
Device '<device name>' is not responding

Unable to write to '<address>' on device '<device name>'

Device '<device name>' is not responding

Error Type:
Serious

Possible Cause:
1. The serial connection between the device and the Host PC is broken.
2. The communications parameters for the serial connection are incorrect.
3. The named device may have been assigned an incorrect Network ID.
4. The response from the device took longer to receive than the amount of time specified in the "Request Timeout" device setting.

Solution:
1. Verify the cabling between the PC and the device.
2. Verify the specified communications parameters match those of the device.
3. Verify that the Network ID given to the named device matches that of the actual device.
4. Increase the Request Timeout setting so that the entire response can be handled.

Unable to write to '<address>' on device '<device name>'

Error Type:
Serious

Possible Cause:
1. The serial connection between the device and the Host PC is broken.
2. The communications parameters for the serial connection are incorrect.
3. The named device may have been assigned an incorrect Network ID.

Solution:
1. Verify the cabling between the PC and the device.
2. Verify that the specified communications parameters match those of the device.
3. Verify that the Network ID given to the named device matches that of the actual device.

Driver Warning Messages

The following error/warning messages may be generated. Click on the link for a description of the message.

Driver Warning Messages
Data Block DB' <block number>' not defined in '<device name>' write operation has failed
Failure reading device '<device name>' configuration
Protocol Error-Number of bytes received = '<num bytes>' Expected = '<num bytes>'
Requested Data Block DB'<block number>' not defined in '<device name>' block has been disabled

Data Block DB '<block number>' not defined in '<device name>' write operation has failed

Error Type:
Warning

Possible Cause:
An attempt has been made to write to a nonexistent location in the specified device.

Solution:
Verify the tags assigned to addresses in the specified range on the device and eliminate ones that reference invalid locations.

Failure reading device '<device name>' configuration

Error Type:
Warning

Possible Cause:
A device configuration transaction timed-out. Below are possible causes for this time-out:

1. The serial connection between the device and the Host PC is broken.
2. The communications parameters for the serial port connection are incorrect.

Solution:
1. Verify the cabling between the PC and the device.
2. Verify that the correct baud rate and parity is specified for the named device.

Protocol Error-Number of bytes received = '<num bytes>' Expected = '<num bytes>'

Error Type:
Warning

Possible Cause:
1. Misalignment of packets due to connection/disconnection between PC and device.
2. There is bad cabling connecting the devices causing noise.

Solution:
The driver will recover from this error without intervention. If this error occurs frequently, there may be an issue with the cabling or the device itself.

Requested Data Block DB'<block number>' not defined in '<device name>' block has been disabled

Error Type:
Warning

Possible Cause:
An attempt has been made to reference a nonexistent location in the specified device.

Solution:
Verify that the tags assigned to addresses in the specified range on the device and eliminate ones that reference invalid locations.

Index

E

F

L

M

N

O

P

R

S

U

W