

Guía de Implementación

Envío y recepción de SMS en M221 con la instrucción SEND_RECV_SMS

Producto y Versión:
M221 v1.3.3.3
SoMachine Basic v1.3 SP2
Módem SR2MOD03
Windows 7



Revisión	Fecha	Autor	Modificaciones
1.0	11/2015	Marc Casanova	Primera versión

1. Objetivo

- Realizar un repaso y explicación de los principales parámetros de la nueva función SEND_RECV_SMS para la gestión de SMS en M221.
- Realizar un ejemplo, probado satisfactoriamente en laboratorio, de envío y recepción de un SMS.

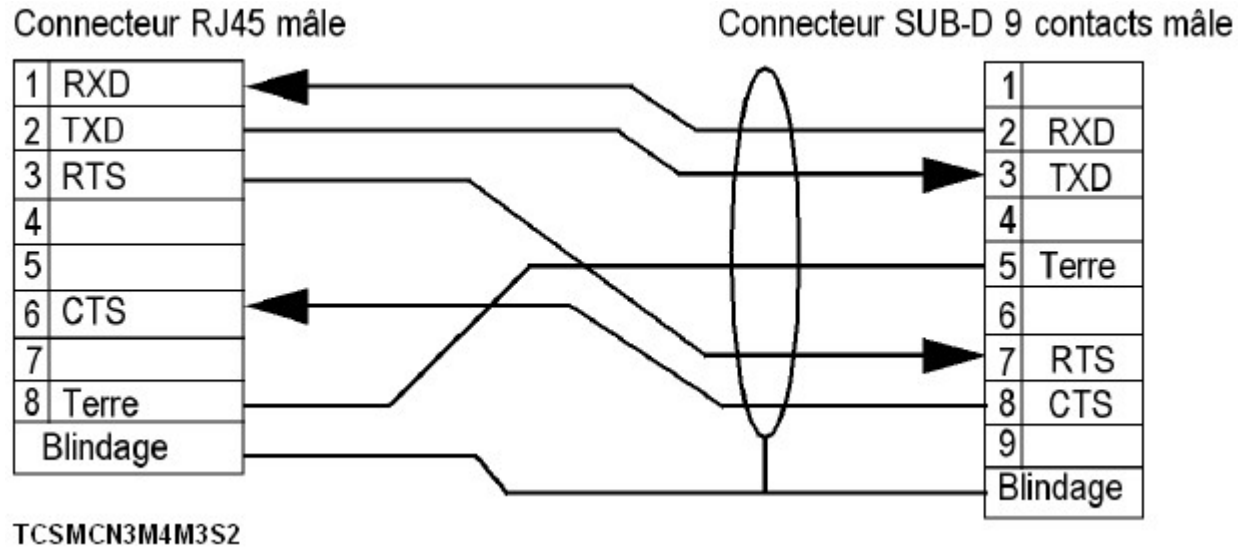
2. Arquitectura y conexionado

- La arquitectura básica para comunicar M221 y módem es:



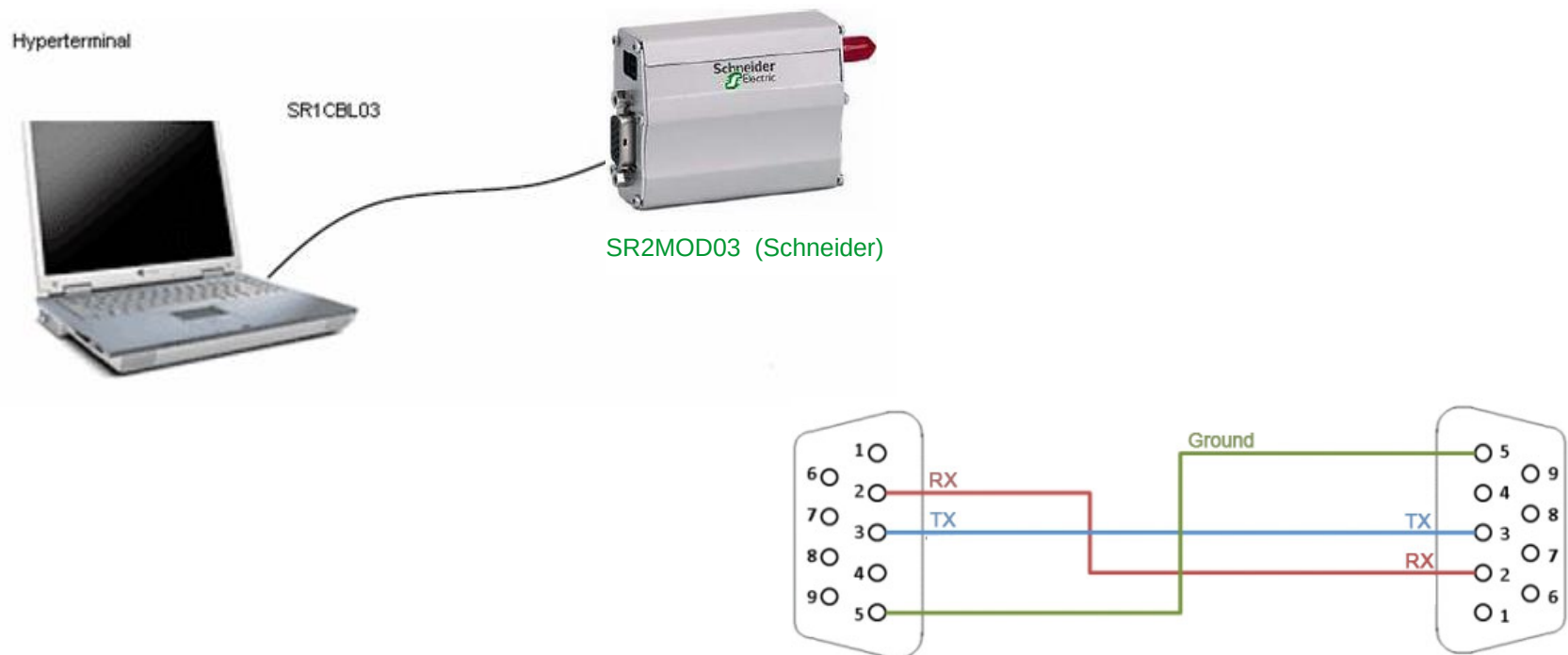
2. Arquitectura y conexionado

- El pinout de este cable es:



2. Arquitectura y conexionado

- La arquitectura básica para comunicar PC y módem (para configurarlo desde el Hyperterminal de Windows) es:



3. Configurar módem

- Este paso sólo es necesario si se desea cambiar la parametrización por defecto del módem usado (SR2MOD03 en nuestro caso).
- Por defecto, el SR2MOD03 viene con la configuración que aparece a continuación. En nuestro caso es la que hemos usado para las pruebas:

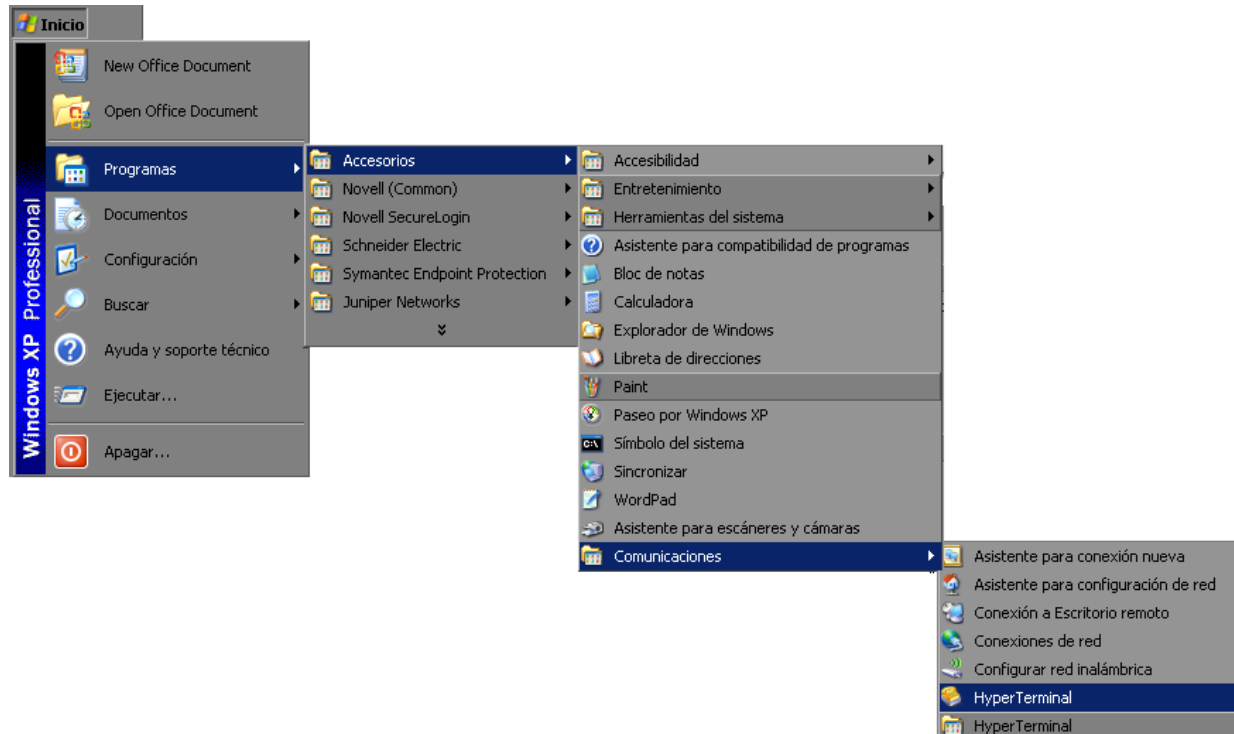
▪ Data rate: 19200 bauds,		
▪ Data size: 8 bits,		
▪ Parity: No Parity,		
▪ Stop bits: 1 Stop bit,		
▪ Flow control: hardware flow control deactivated,		
▪ AT commands Echo: Echo deactivated,		
▪ DSR Signal: DSR OFF,		
▪ Ring Register: S0=2 (answer after 2 rings).		

3. Configurar módem

- Windows 7 no incorpora de serie el Hyperterminal, por lo que para seguir los pasos que se muestran a continuación es necesario descargar una herramienta externa que haga sus funciones.
- Una de las posibles aplicaciones sustitutas es HypeTerminal Private Edition (HTPE), accesible y gratuito buscando en Google.

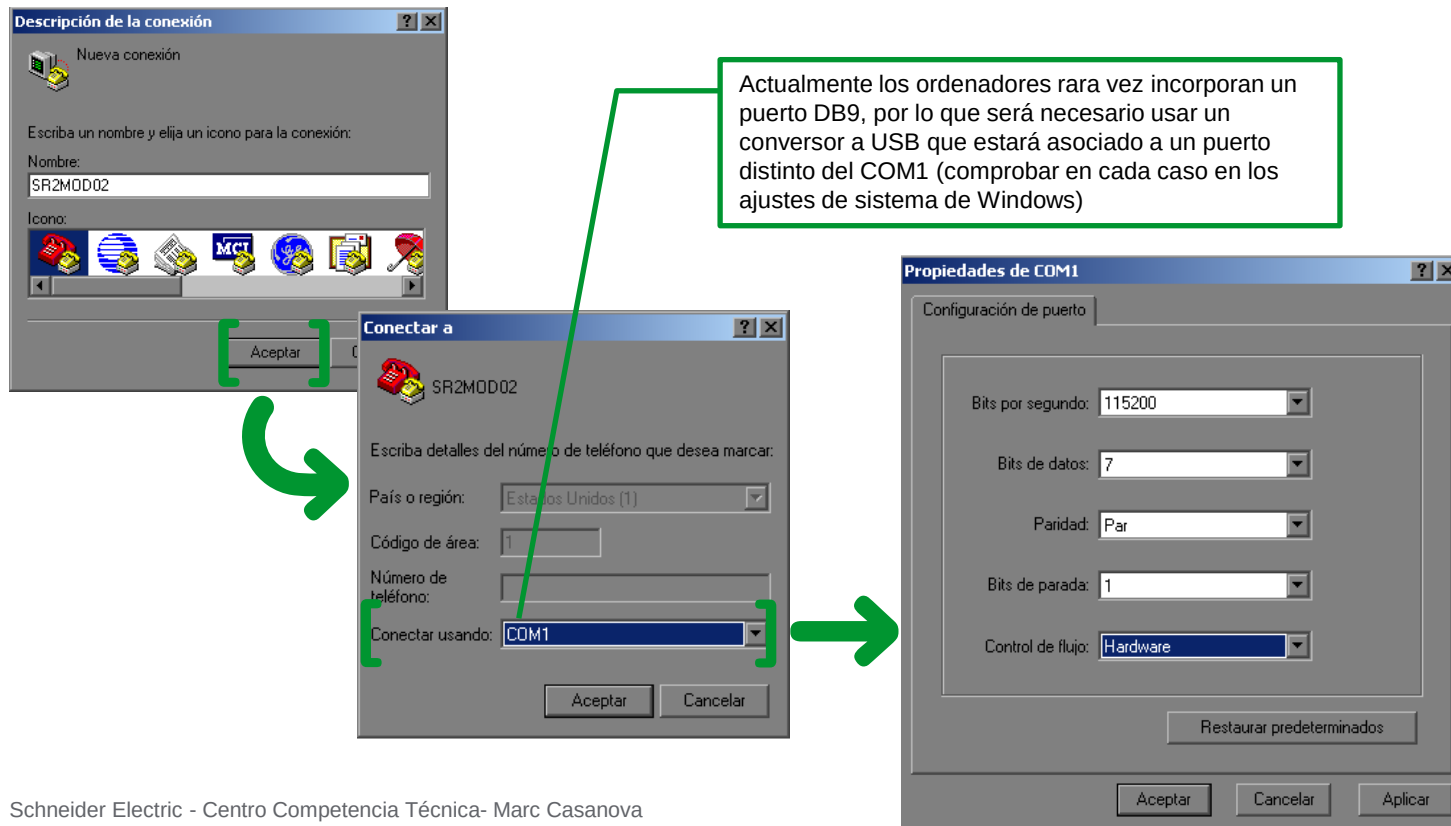
3. Configurar módem

- El primer paso es ejecutar el Hyperterminal de Windows, aplicación desde la cual conectaremos PC con módem para enviar los comandos AT necesarios y cambiar los parámetros de éste:



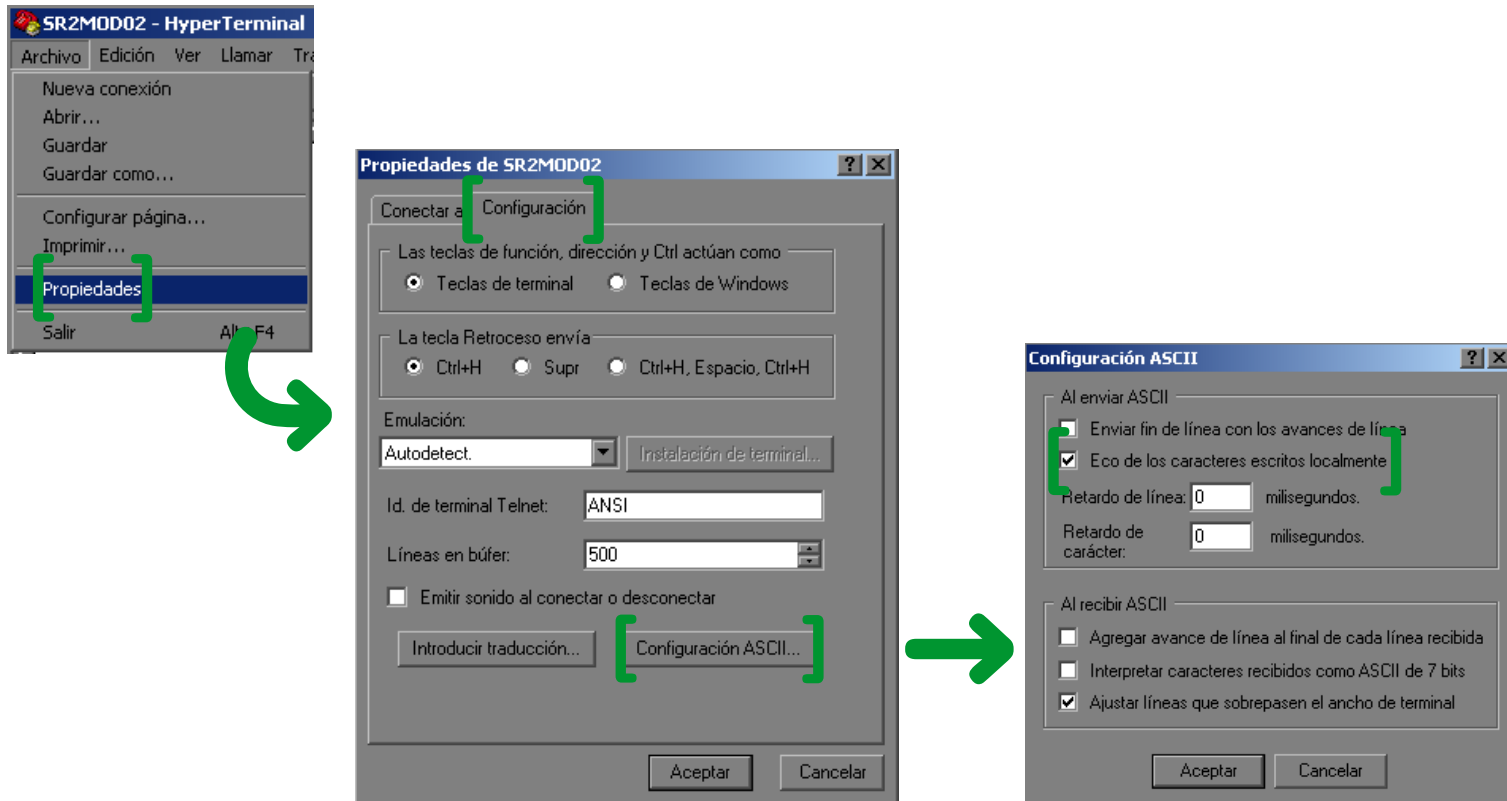
3. Configurar módem

- A continuación crearemos la nueva conexión, seleccionaremos el puerto COM que usaremos para la misma y la configuraremos según los ajustes actuales del módem:



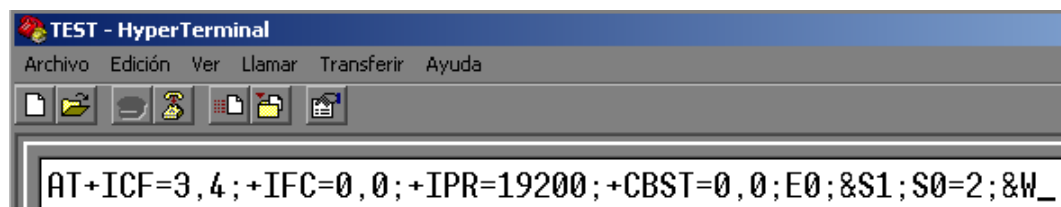
3. Configurar módem

- Para evitar confusiones a la hora de enviar los comandos AT conviene activar la opción de eco local de los caracteres escritos:



3. Configurar módem

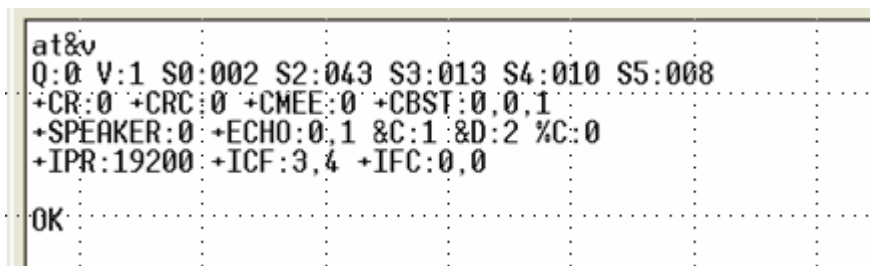
- Desde la pantalla del Hyperterminal debemos enviar los siguientes comandos AT al módem:



```
TEST - HyperTerminal
Archivo Edición Ver Llamar Transferir Ayuda
AT+ICF=3,4;+IFC=0,0;+IPR=19200;+CBST=0,0;E0;&S1;S0=2;&W_
```

El comando &W realizará la escritura de las configuraciones en el módem, por lo que la comunicación desde el PC se interrumpirá y será necesario cambiar las configuraciones de conexión del Hyperterminal para reconectar con el SR2MODxx

- Si al reconectar realizamos un comando AT&V debemos recibir esta respuesta:



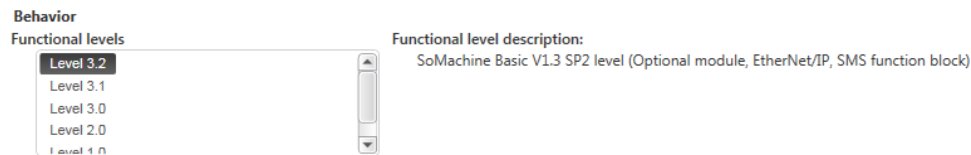
```
at&v
Q:0 V:1 S0:002 S2:043 S3:013 S4:010 S5:008
+CR:0 +CRC:0 +CME:0 +CBST:0,0,1
+SPEAKER:0 +ECHO:0,1 &C:1 &D:2 %C:0
+IPR:19200 +ICF:3,4 +IFC:0,0
OK
```

3. Configurar módem

- A modo informativo, la descripción de los comandos AT enviados en el punto anterior es:
 - » AT+ICF=3,4 → formato de datos 8N1 (8 bits datos, sin paridad, 1 bit de stop)
 - » AT+IFC=0,0 → sin control de flujo
 - » AT+IPR=19200 → velocidad 19200 bps
 - » AT+CBST=0,0 → autovelocidad para llamadas salientes y conexión transparente para salientes y entrantes
 - » ATE0 → caracteres recibidos sin eco
 - » AT&S1 → señal DSR desactivada para modo comando y activada para modo datos
 - » ATS0=2 → respuesta automática tras 2 rings
 - » AT&W → guardar modificaciones en la EEPROM
 - » AT&V → visualizar configuración actual del módem

4. Configurar SoMachine Basic

- En el apartado 'Documentación Asociada de Interés' aparece el proyecto ejemplo de SoMachine Basic usado en esta guía
- El bloque SEND_RECV_SMS es una novedad de SoMachine Basic v1.3 SP1.2. Para asegurar la compatibilidad será necesario tener un nivel lógico en la aplicación >3.2 y un firmware de M221 >v1.3.3.3



4. Configurar SoMachine Basic

Properties **Configuration** **Programming** **Display** **Commissioning**

Messages
MyController (TM21ME16R/G)
Digital inputs
Digital outputs
Analog inputs
High Speed Counters
IO Bus
ETH1
Modbus TCP
Ethernet/IP adapter
SL1 (Serial line)

Serial line configuration

Physical Settings

Device: SR2MOD03
Init command: AT&F;E0;S0=2;Q0;V1;+WIND=0;+CBST=0,0,1;+CNMI=0,2,0,0,0;+CSAS;+CMGF=0;+CMFE=1;&V
Baud rate: 19200
Parity: None
Data bits: 8
Stop bits: 1
Physical medium: ☒ RS-232

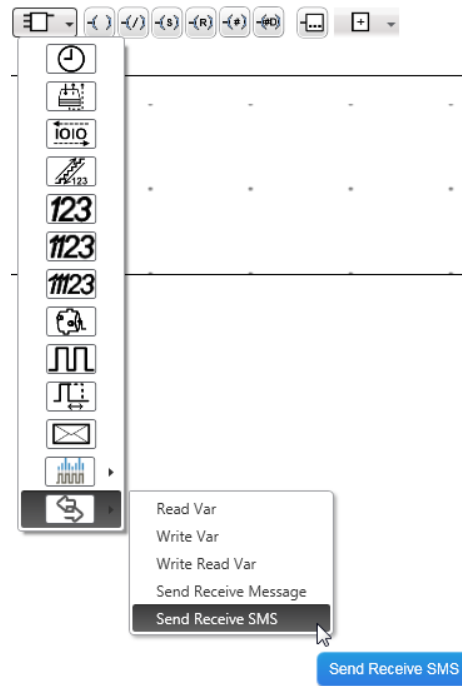
Protocol Settings

Protocol: ASCII
Response timeout (x 100 ms): 0
Stop condition:
☐ Frame length received
☒ Frame received timeout (ms) 255
Frame structure:
☐ Start character 0
☐ First end character 0
☐ Second end character 0
☐ Send frame characters

¡Atención! Los parámetros deben quedar configurados tal como muestra esta slide para que el bloque funcione, incluido el “Init command” (en algunas versiones de SoMB viene con otro valor por defecto).

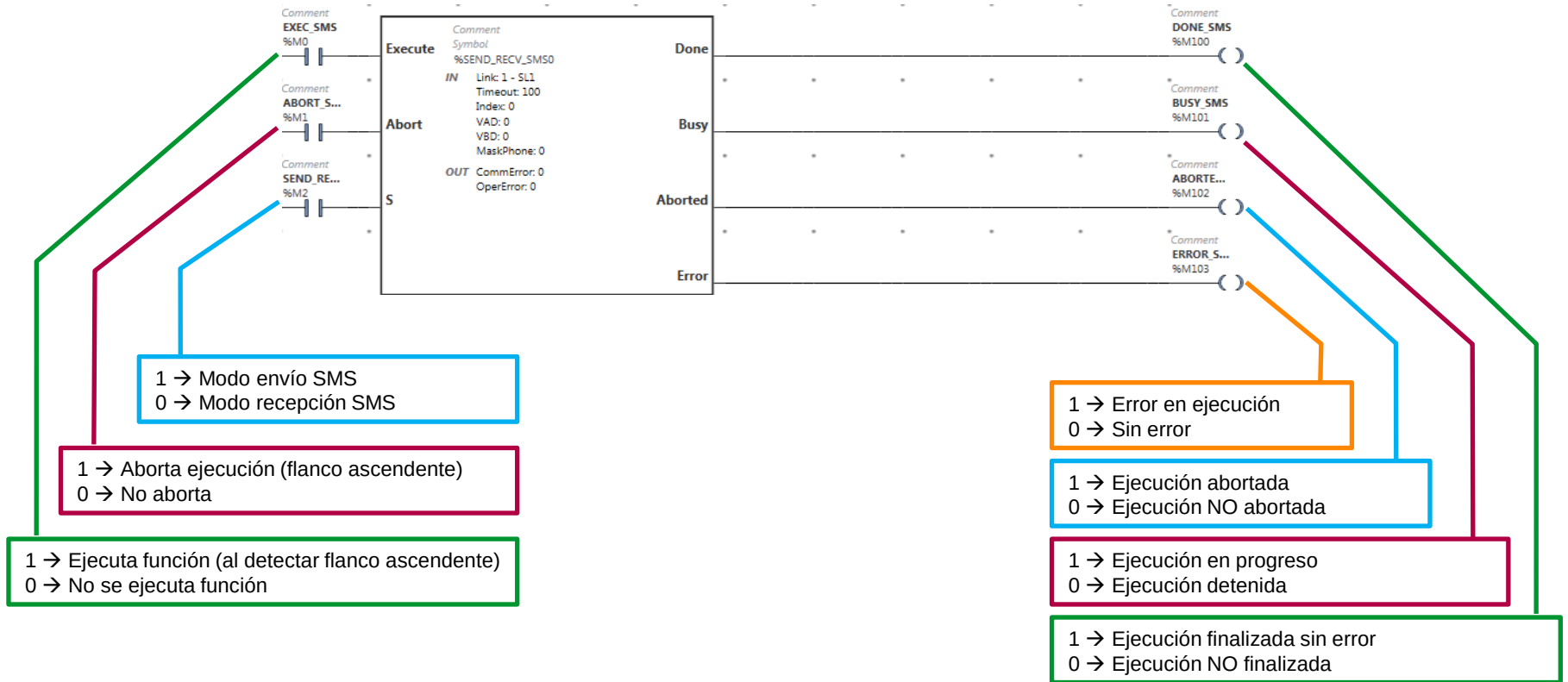
4. Configurar SoMachine Basic

- Introducir el bloque de comunicación SMS:



4. Configurar SoMachine Basic

- Asignar las señales de entrada / salida:



NOTE: When the **Busy** output is set to **TRUE**, the execution continues until one of the **Done**, **Aborted**, or **Error** outputs is set to **TRUE**.

NOTE: While the **Busy** output is set to **TRUE**, changes in the **Execute** input do not affect the execution of the ongoing function block. However, if another **%SEND_RECV_SMS** function block is called, this SMS is refused (CommError = 16 hex and OperError = 0B hex).

4. Configurar SoMachine Basic

- Configurar el bloque:

The screenshot displays the Schneider Electric SoMachine Basic software interface. The top navigation bar includes tabs for Properties, Configuration, Programming, Display, and Commissioning. The left sidebar shows a tree view of various objects, with the 'Tools' tab selected. A green arrow points from the 'Tools' tab to the 'Send Receive SMS' block in the main workspace. The main workspace shows a ladder logic diagram with a 'Send Receive SMS' block. The block is configured with the following parameters:

name	Comment
EXEC SMS	%M0
Abort	%M1
SEND_RE...	%M2

The block is connected to a 'Done' output, which is linked to a 'DONE SMS' output (%M100). The block also has an 'Error' output, which is linked to an 'ERROR S...' output (%M103). The block is also connected to a 'Busy' output, which is linked to a 'BUSY SMS' output (%M101). The block is also connected to an 'Aborted' output, which is linked to an 'ABORTE...' output (%M102).

At the bottom of the interface, the 'Send Receive SMS properties' table is visible:

Used	Address	Symbol	Link	Timeout	Index	VAD	VBD	MASKPHONE	Comment
☒	%SEND_REC...		1 - SL1	100	0	0	0	2#0000000000000000	

A green arrow points from the 'Send Receive SMS properties' table to the 'SMS configuration' tab in the bottom right corner.

4. Configurar SoMachine Basic

- Configurar el bloque. Hay 3 apartados para configurar:

1. Mensajes

Tabla para configurar hasta 16 mensajes a enviar. Mediante la variable %SEND_RECV_SMSi.INDEX es posible seleccionar la posición de la tabla con el mensaje a enviar. Por ejemplo, si el índice vale 0 se enviará el mensaje de la posición 0 de la tabla.

El tamaño máximo del mensaje es:	GSM 7-bit	105 characters max.
	UNICODE	50 characters max.

NOTE: The character format is determined automatically by the characters in the text field.

Se pueden agregar las siguientes variables al mensaje:

Placeholder	Replaced at execution by:	Number of characters in GSM 7-bit formats	Number of characters in UNICODE format
\$DATE ⁽¹⁾	YY/MM/DD (present date)	8 + 1	16 + 2
\$TIME ⁽¹⁾	HH:MM:SS (present time)	8 + 1	16 + 2
\$VAD	The DWORD value of parameter %SEND_RECV_SMSi.VAD converted to text.	12 maximum	24 maximum
\$VBD	The DWORD value of parameter %SEND_RECV_SMSi.VBD converted to text.	12 maximum	24 maximum
\$ \$	The symbol \$	1	2

4. Configurar SoMachine Basic

- Configurar el bloque. Hay 3 apartados para configurar:

2. Comandos

Tabla que especifica hasta 16 mensajes posibles a recibir vía SMS. Mediante la variable %SEND_RECV_SMSi.INDEX se selecciona la posición de la tabla que ocupa el texto del SMS recibido. Por ejemplo, si el índice vale 0 en modo recepción, significa que el SMS recibido contiene el texto declarado en la posición 0 de la tabla. Si el SMS no coincide con ninguna posición, se descarta.

Se pueden agregar las siguientes variables a los comandos:

Placeholder	Replaced at execution by:	Number of characters in GSM 7-bit formats	Number of characters in UNICODE format
\$DATE ⁽¹⁾	YY/MM/DD (present date)	8 + 1	16 + 2
\$TIME ⁽¹⁾	HH:MM:SS (present time)	8 + 1	16 + 2
\$VAD	The DWORD value of parameter %SEND_RECV_SMSi.VAD converted to text.	12 maximum	24 maximum
\$VBD	The DWORD value of parameter %SEND_RECV_SMSi.VBD converted to text.	12 maximum	24 maximum
\$ \$	The symbol \$	1	2

4. Configurar SoMachine Basic

- Configurar el bloque. Hay 3 apartados para configurar:

3. Números de Teléfono

Tabla para configurar hasta 16 números de teléfono para el envío y recepción de SMS. Mediante la variable %SEND_RECV_SMSi.MASKPHONE se selecciona:

I. En caso de envío, el nº de teléfono a enviar el mensaje (sólo un nº por ejecución de la función). Por ejemplo, si en binario la variable vale

2#0000 0000 0001 0000 se enviará al teléfono de la posición 4.

II. En caso de recepción, autoriza la recepción de SMS del nº de teléfono seleccionado. Cualquier SMS recibido por un nº no autorizado no será recepcionado. Por ejemplo, si en binario la variable vale

2#0000 0000 0000 0100 sólo se recibirán los SMS enviados desde el nº configurado en la posición 2 de la tabla.

Ojo, en ambos casos sólo es posible la activación de 1 nº de teléfono mediante la Máscara. No es posible la activación simultanea de varios.

4. Configurar SoMachine Basic

- Configurar el bloque. Hay 3 apartados para configurar:

3. Números de Teléfono

El formato del nº de teléfono debe ser <código internacional><nº teléfono>. Es decir, para trabajar con un nº en España debe declararse como 34612345678. No declarar nunca 0034 ó +34.

4. Configurar SoMachine Basic

● Propiedades del bloque:

Send Receive SMS properties

Used	Address	Symbol	Link	Timeout	Index	VAD	VBD	MASKPHONE	Comment
☒	%SEND_RECV_SMS0		1 - SL1	100	0	0	0	2#0000000000000000	

Property	Value	Description
Used	Activated / deactivated check box.	Indicates whether the address is in use.
Address	%SEND_RECV_SMS <i>i</i> , where <i>i</i> is from 0 to the number of objects available on this logic controller.	<i>i</i> is the instance identifier. For the maximum number of instances, refer to the Programming Guide of your logic controller.
Symbol	User-defined text.	The symbol uniquely identifies this object. For details, refer to Defining and Using Symbols .
Link	1 - SL1	Serial line used to communicate with the modem.
Timeout	0...255 ms A value of 0 means no timeout enforced.	The timeout sets the maximum time to wait to receive a response from the modem. If the timeout expires, the exchange terminates with an error code (%SEND_RECV_SMS <i>i</i> .CommError = 01 hex). If the system receives a response after the timeout expiration, this response is ignored. NOTE: The timeout set on the function block overrides the value configured on the SoMachine Basic configuration screen. For more details, refer to the Programming Guide for your logic controller.
Index	0...15 NOTE: 0 corresponds to the first string of the list.	- While sending, the value of the index is used to select a text to send from the Messages table. - While receiving, the value corresponds to the index in the Commands table that matches the received text.
VAD	-214748364 ...2147483647	- While sending, the value in %SEND_RECV_SMS<i>i</i>.VAD replaces the placeholder \$VAD in the text of the SMS. - While receiving, the value in %SEND_RECV_SMS<i>i</i>.VAD contains the value where the placeholder \$VAD is inserted in the SMS stored in the Commands table.
VBD	-214748364 ...2147483647	- While sending, the value in %SEND_RECV_SMS<i>i</i>.VBD replaces the placeholder \$VBD in the text of the SMS. - While receiving, the value in %SEND_RECV_SMS<i>i</i>.VBD contains the value where the placeholder \$VBD is inserted in the SMS stored in the Commands table.
MASKPHONE	0000000000000000 bin to 1111111111111111 bin	The initial value of the mask. - While sending, this mask is used to select the recipients of the SMS from the Phone numbers table. Example: 0000000000010011 bin = the SMS is sent to the first, second and fifth phone numbers (indexes 0, 1 and 4 respectively) listed in the Phone numbers table. - While receiving, the mask is applied to the Phone numbers table to create a list of valid numbers. A bit of the mask indicates which phone number was used to send the SMS to the logic controller. Example: 000000000000100 bin means the third phone number of the Phone numbers list (index 2) has sent the SMS.
Comment	User-defined text	A comment to associate with this object.

5. Ejemplo (simulación aplicación)

- Supongamos que disponemos de un sensor para contabilizar la cantidad de personas que entran y salen de un recinto.
- El sensor está conectado a un PLC M221 que realiza el conteo en la variable %MD0 (aforo actual) y va actualizando el aforo máximo del día en la variable %MD1.



5. Ejemplo (simulación aplicación)

- El responsable de mantenimiento del edificio desea saber en cualquier momento ambos datos: aforo actual y máximo del día. Para ello quiere enviar un SMS a un módem conectado al PLC para que le sirva los datos.



5. Ejemplo (condiciones funcionamiento)

- Condiciones de funcionamiento:

I. Si el responsable envía el SMS 'Aforo actual' debe recibir como respuesta un SMS con el aforo actual.

II. Si el responsable envía el SMS 'Aforo maximo' debe recibir como respuesta un SMS con el aforo máximo del día.

- En el programa se ha obviado parte de la lógica por ser sólo un ejemplo. Es decir, no se gestionan automáticamente las órdenes de envío / recepción, variable maskphone, etc. Se han de gestionar directamente desde una tabla de animación. Ver lógica de programa usada en las siguientes slides.

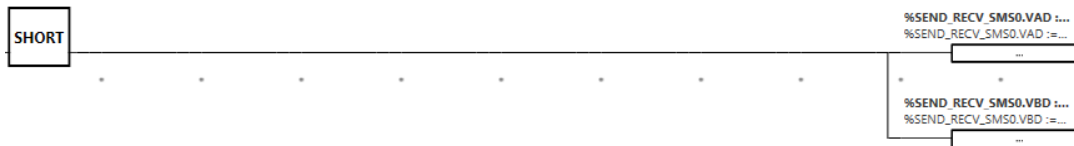
5. Ejemplo (lógica de programa)

Messages	Commands	Phone numbers						
<table><tr><th>Index</th><th>Value</th></tr><tr><td>0</td><td>Aforo a las \$TIME = \$VAD</td></tr><tr><td>1</td><td>Aforo máximo día &DATE = &VBD</td></tr></table>	Index	Value	0	Aforo a las \$TIME = \$VAD	1	Aforo máximo día &DATE = &VBD		
Index	Value							
0	Aforo a las \$TIME = \$VAD							
1	Aforo máximo día &DATE = &VBD							

Messages	Commands	Phone numbers						
<table><tr><th>Index</th><th>Value</th></tr><tr><td>0</td><td>Aforo actual</td></tr><tr><td>1</td><td>Aforo maximo</td></tr></table>	Index	Value	0	Aforo actual	1	Aforo maximo		
Index	Value							
0	Aforo actual							
1	Aforo maximo							

Messages	Commands	Phone numbers				
<table><tr><th>Index</th><th>Value</th></tr><tr><td>0</td><td>34612345678</td></tr></table>	Index	Value	0	34612345678		
Index	Value					
0	34612345678					

name %MD0 = AFORO_ACTUAL Y %MD1 = AFORO_MAX



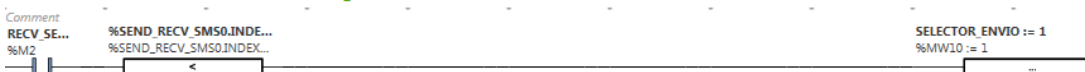
En cada ciclo de scan volcamos sobre las variables VAD y VBD los aforos actual y máximo respectivamente.

name SI MODO RECEPCION Y PETICION AFORO_ACTUAL -> INDICE PROX. SMS=0



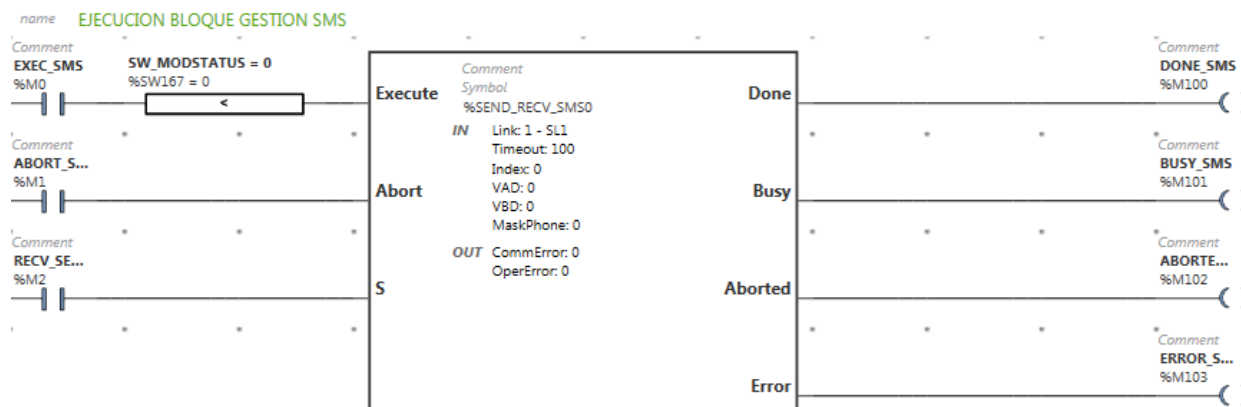
Si el modo de recepción está activo, se detecta recepción de SMS sin error y hemos recibido orden de consultar el aforo actual (índice tabla comandos = 0), configuramos el selector para el envío del próximo SMS a 0 (%MW10 := 0).

name SI MODO RECEPCION Y PETICION AFORO_MAX -> INDICE PROX.SMS=1



Al contrario, si el modo de recepción está activo, se detecta recepción de SMS sin error y hemos recibido orden de consultar el aforo máximo (índice tabla comandos = 1), configuramos el selector para el envío del próximo SMS a 1 (%MW10 := 1).

5. Ejemplo (lógica de programa)



Contactos entrada / salida del bloque. En este ejemplo no se gestiona automáticamente el envío de SMS tras recibir una petición. Se gestionará dicho comportamiento mediante tabla de animación jugando con los bits %M0 (ejecución de la instrucción) y %M2 (selector envío / recepción).

Siempre es bueno comprobar que el estado del módem es OK antes de ejecutar el bloque (%SW167=0).

A continuación se adjunta el proyecto ejemplo usado:



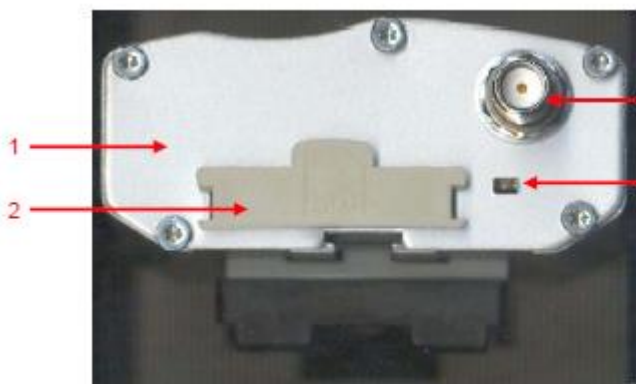
6. Diagnóstico

- En este apartado aparecen los diagnósticos a realizar frente a posibles errores
- Los diagnósticos pueden hacerse desde el módem o a través de los códigos y bits relacionados de la SGT

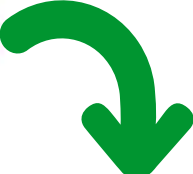


6.1 Diagnóstico módem

- El diagnóstico del módem puede realizarse a través del LED que incorpora:



N°	Description
1	Rear side
2	SIM Card Cover
3	Connector SMA/F (GSM antenna)
4	GSM LED (see § 4.1.4)



GSM LED	LED activity	Modem state
ON	LED on fixed	The modem is powered, it is ready to function but not yet recognized by the network; the PIN code has not yet been entered or the antenna is not connected.
	LED flashing (once every 2 seconds)	The modem is powered, the PIN code is active, the modem is recognized by the network and is ready to make or receive a call (Idle mode).
	LED flashing (Once a second)	The modem is powered and currently in communication (Voice, Data or Fax).
OFF	LED off	The modem is not powered or is in the RESET phase.

A través de comandos AT es posible realizar un diagnóstico más exhaustivo del módem. No se entra en este detalle ya que la SGT se encarga de esta comunicación (ver diagnóstico SGT en apartado 5.2)

6.2 Diagnóstico bloque SEND_RECV_SMS

- El diagnóstico del bloque se realiza consultando las variables *%SEND_RECV_SMSi.CommError* y *%SEND_RECV_SMSi.OperError*

Communication Error Codes

This table describes the error codes written to the %SEND_RECV_SMSi.CommError output object:

Detected error code	Name	Description
00 hex	CommunicationOK	Exchange is correct. NOTE: In this case, the %SEND_RECV_SMSi.OperError output object contains the modem signal level, as opposed to an error code.
01 hex	TimedOut	Exchange stopped because timeout expired.
02 hex	Abort	Exchange stopped on a rising edge on the %SEND_RECV_SMSi.Abort input.
03 hex	BadLink	Link is incorrect.
04 hex	BadCommand	Command is incorrect.
05 hex	BadMgtTable	Management table format is incorrect.
06 hex	BadParameters	Specific parameters are incorrect.
07 hex	ProblemSendingSms	SMS send command unsuccessful.
09 hex	RecvCmdError	Invalid command.
0A hex	SendValueError	Invalid value.
0B hex	SystemResourceMissing	System resource is unavailable.
0E hex	BadLength	Length is incorrect.
FE hex	ProtocolSpecificError	Indicates that a protocol error has been detected. NOTE: In this case, the %SEND_RECV_SMSi.OperError output object contains more details. Refer to Operation Error Codes .
FF hex	Refused	SMS is refused. NOTE: In this case, the %SEND_RECV_SMSi.OperError output object contains more details. Refer to Operation Error Codes .

6.2 Diagnóstico bloque SEND_RECV_SMS

Operation Error Codes

This return code is significant when the communication error code (%SEND_RECV_SMSi.CommError output object) has the value:

- 00 hex (correct protocol)
- FE hex (incorrect protocol)
- FF hex (SMS refused)

When the %SEND_RECV_SMSi.CommError is 00 hex (correct protocol), the %SEND_RECV_SMSi.OperError output object indicates the Received Signal Strength Indication (RSSI):

Decimal value in the %SEND_RECV_SMSi.OperError object	RSSI modem signal level
Lower than 9	Marginal value (the attenuation exceeds the limit to keep the wireless network up)
10 to 14	Ok
15 to 19	Good
Upper than 20	Excellent

When the %SEND_RECV_SMSi.CommError is FE hex (incorrect protocol), the %SEND_RECV_SMSi.OperError output object returns more details:

Hexadecimal value in the %SEND_RECV_SMSi.OperError object	Name	Description
100 hex	ModemConfSLAsciiFailed	The ASCII configuration of the serial line is incorrect.
200 hex	ModemReconfSLFailed	The configuration of the serial line back to the user configuration is incorrect.
300 hex	ModemBusy	The modem answers BUSY to the dial command.
400 hex	ModemNoDialtone	The modem answers NODIALTONE to the dial command.
500 hex	ModemNoCarrier	The modem carrier signal has been lost or disconnected. The modem answers NO CARRIER to the dial command.
600 hex	ModemBadAnswer	The response from the modem is incorrect.
Specific errors for SIM card use		
1000 hex	SimConfigurationFailed	The SIM card configuration is incorrect. For example, a PUK code is requested.
2000 hex	SimPinCodeInvalid	The PIN code is incorrect.
4000 hex	SimSmsCenterInvalid	The SMS center phone number is incorrect.

When the %SEND_RECV_SMSi.CommError is FF hex (SMS refused), the %SEND_RECV_SMSi.OperError output object returns more details:

Hexadecimal value in the %SEND_RECV_SMSi.OperError object	Name	Description
01 hex	TargetResourceMissing	The target system resource is unavailable.
05 hex	BadLength	The length is incorrect.
06 hex	CommChannelErr	An error has been detected on the communication channel.
0B hex	SystemResourceMissing	The system resource is unavailable.
0C hex	TargetCommInactive	The target communication function is not active.
0D hex	TargetMissing	The target is unavailable.
0F hex	ChannelNotConfigured	The communication channel is not configured.

Make the most of your energy

www.schneiderelectric.es

