
Chapter 17: SMS: Short Message Service

The I-8417/8817/8437/8837, I-7188EG, I-7188XG & Wincon-8xx7 controller can integrate with a GSM Modem to support SMS: Short Message Service. This allows user to request information or control something from his own cellular phone to the ISaGRAF controller. Beside, the controller can also send information and alarms to user's cellular phone.

17.1: Hardware Installation

The I-8417/8817/8437/8837 supports SMS since its driver version of 2.24, while version 1.14 for I-7188EG, and version 1.12 for I-7188XG. If your driver is older one, please upgrade the hardware driver to the associate version or a higher version. The driver can be found from the below ICP DAS's FTP site:

I-8417/8817/8437/8837: <ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/isagraf/8000/driver/>
I-7188EG: <ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/isagraf/7188eg/driver/>
I-7188XG: <ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/isagraf/7188xg/driver/>

The I/O library should be re-installed if yours is older one. Please refer to section 1.2. Or you can refer to Appendix A.2 to simply install "C functions" with the below items.

SMS_test, SMS_get, SMS_gets, SMS_send, SMS_sts
and "I/O complex equipment" : SMS.

The **GSM Modem GM29** (900/1800) is recommended for the ISaGRAF controller since its driver version of I-8xx7:2.47, I-7188EG:1.38, I-7188XG:1.35 & Wincon-8xx7:3.09. You may purchase them from ICP DAS or from your local agent. ICP DAS is not sure for other GSM modems working or not.

Note: Please REMOVE the password setting in SIM card , then plug it into GSM modem.

I-8xx7(COM4/5) W-8xx7(COM2)	GSM cable of GM29	7188EG/XG:COM3/4 RS232	GSM cable of GM29
2 RXD	———— 2 TXD	RXD	———— 2 TXD
3 TXD	———— 3 RXD	TXD	———— 3 RXD
5 GND	———— 5 GND	GND	———— 5 GND
4 DTR	----- 4 DSR	DTR (or RTS)	----- 4 DSR
7 RTS	----- 7 CTS	DTR (or RTS)	----- 7 CTS

17.2: A SMS demo example

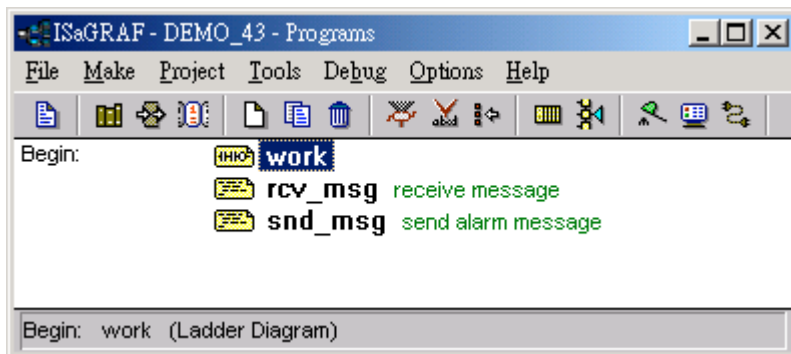
The demo project is located at demo_43, please refer to section 11.1 to install it to your ISaGRAF workbench. Or It can be download at ICP DAS's ftp site.

<ftp://ftp.icpdas.com/pub/cd/8000cd/napdos/isagraf/8000/demo/>

Variables :

Name	Type	Attribute	Description
M1	Boolean	Internal	Trigger to send an alarm message when K1 is pushed
M2	Boolean	Internal	Trigger to send a report message when a message is coming
K1	Boolean	Input	Pushbutton 1, connect to push4key
L1	Boolean	Output	Output 1, connect to show3led
L2	Boolean	Output	Output 2, connect to show3led
L3	Boolean	Output	Output 3, connect to show3led
Q1	Boolean	Internal	Test if message is coming
TMP	Boolean	Internal	Temportary usage
SMS_available	Boolean	Input	is SMS available ? connect to SMS - status
T1	Timer	Internal	Blinking time of L1 to L3, init at T#500ms
data	Message	Internal	The coming Message
phone	Message	Internal	phone No. of sender
Date_time	Message	Internal	Message coming date & time in string format
To_who	Message	Internal	phone No of receiver, please use your own No.
Msg_to_send	Message	Internal	Message to send out
Year1	Integer	Internal	Message coming year
Mon1	Integer	Internal	Message coming month
Day1	Integer	Internal	Message coming date
Wday1	Integer	Internal	Message coming week date
Hour1	Integer	Internal	Message coming hour
Min1	Integer	Internal	Message coming minute
Sec1	Integer	Internal	Message coming second
Q1_cnt	Integer	Internal	Message coming count, declared as retained variable
Msg_status	Integer	Internal	Message sending status
TMP_v	Integer	Internal	temportary usage

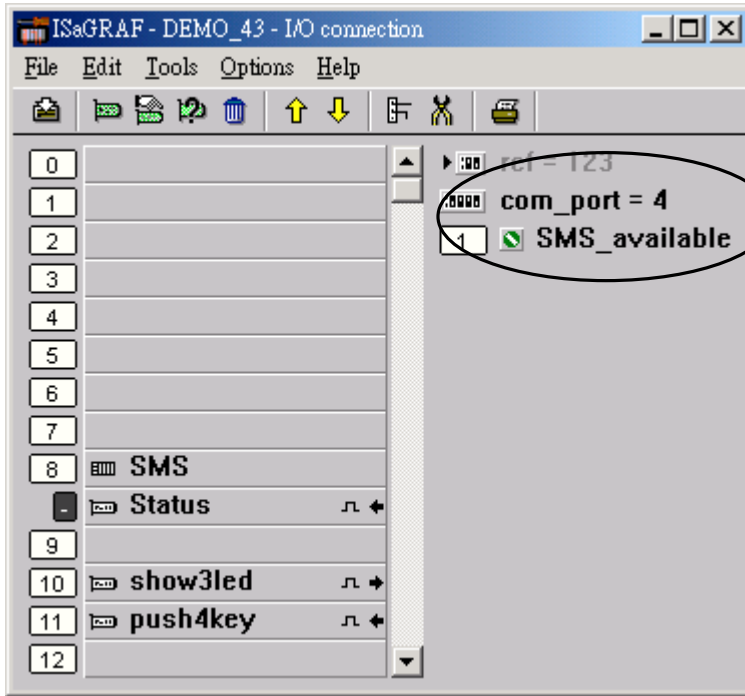
Project architecture :



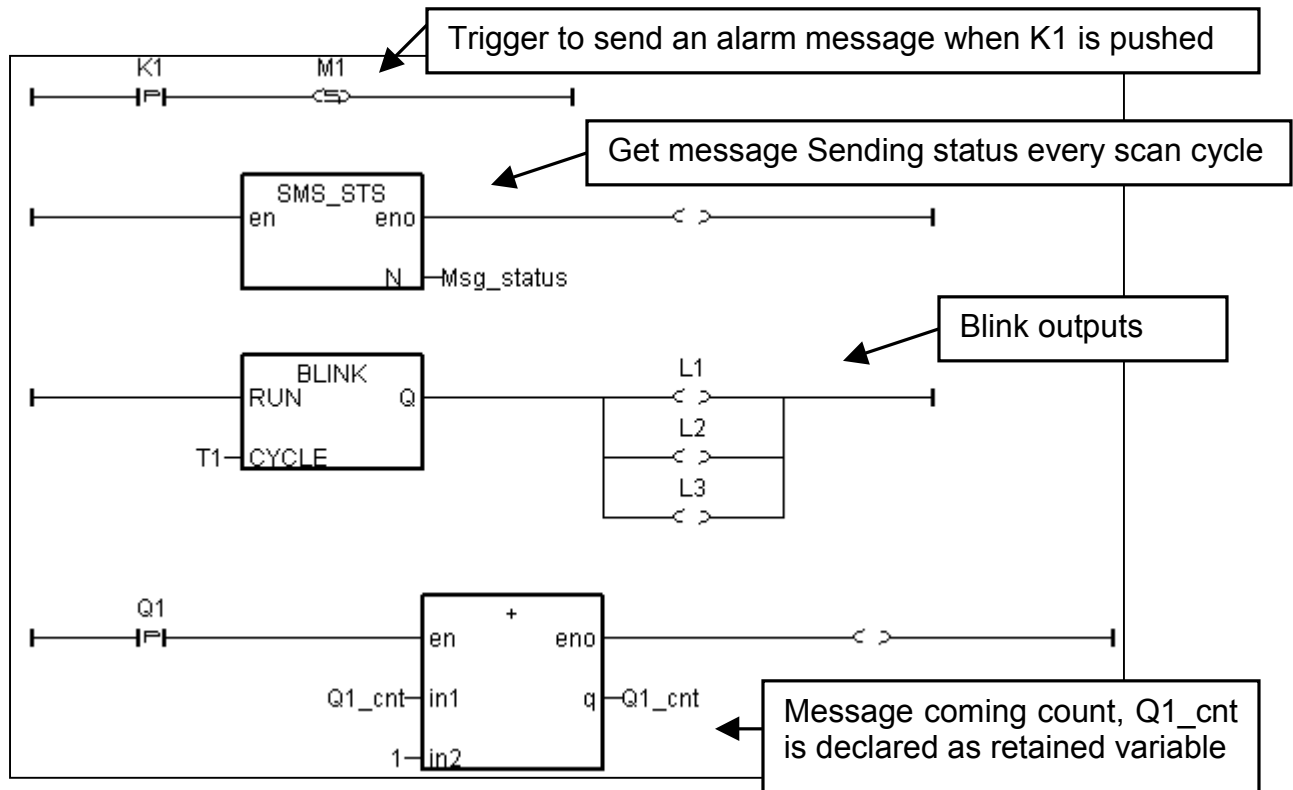
Operation actions:

1. If K1 is pushed, an Alarm message will be sent.
2. If the user send a message in format, for ex. T0200 or T1500 to the controller, the blinking period will change to 200ms and 1500ms. And then the controller will response a report message back to the user.

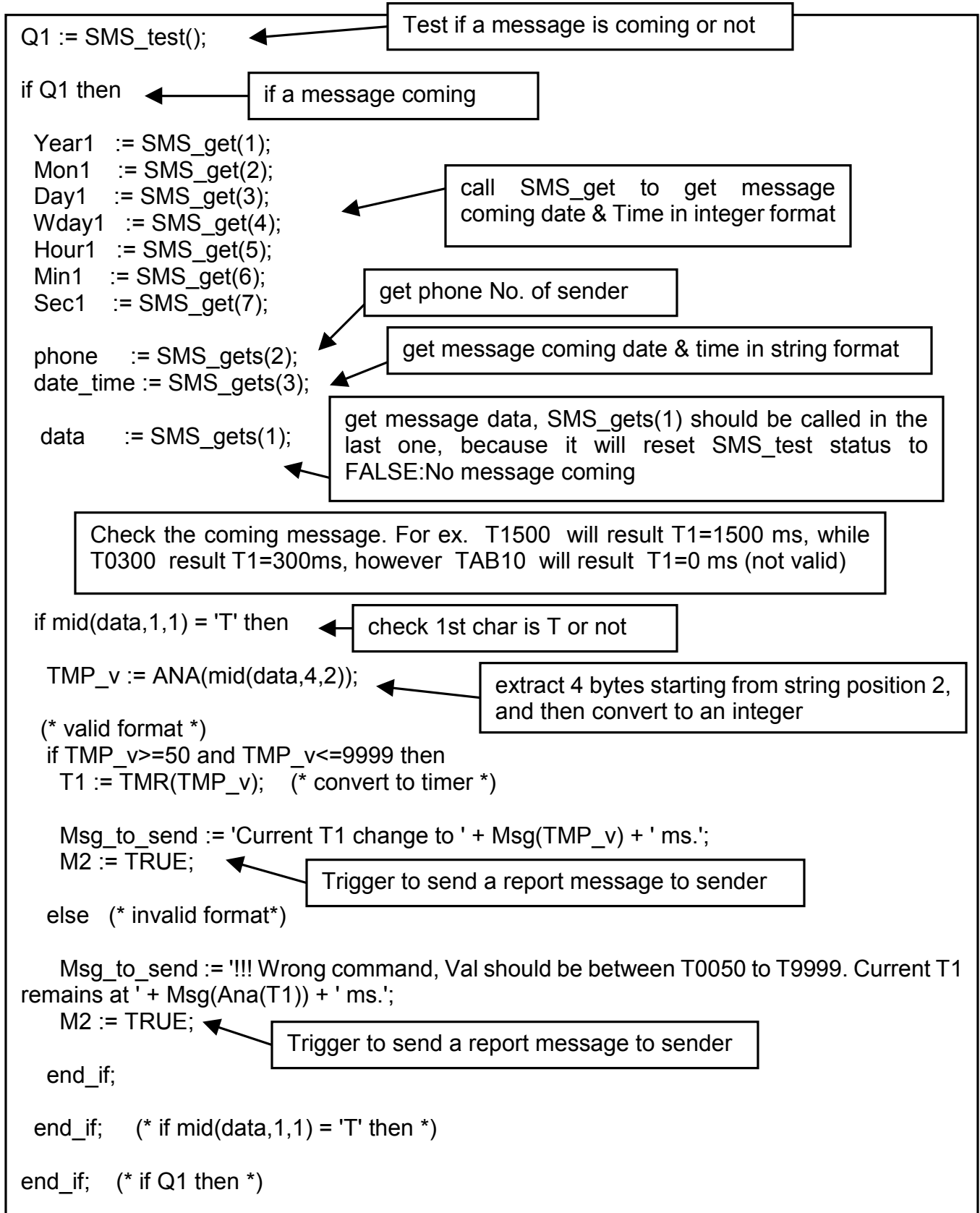
I/O connection:



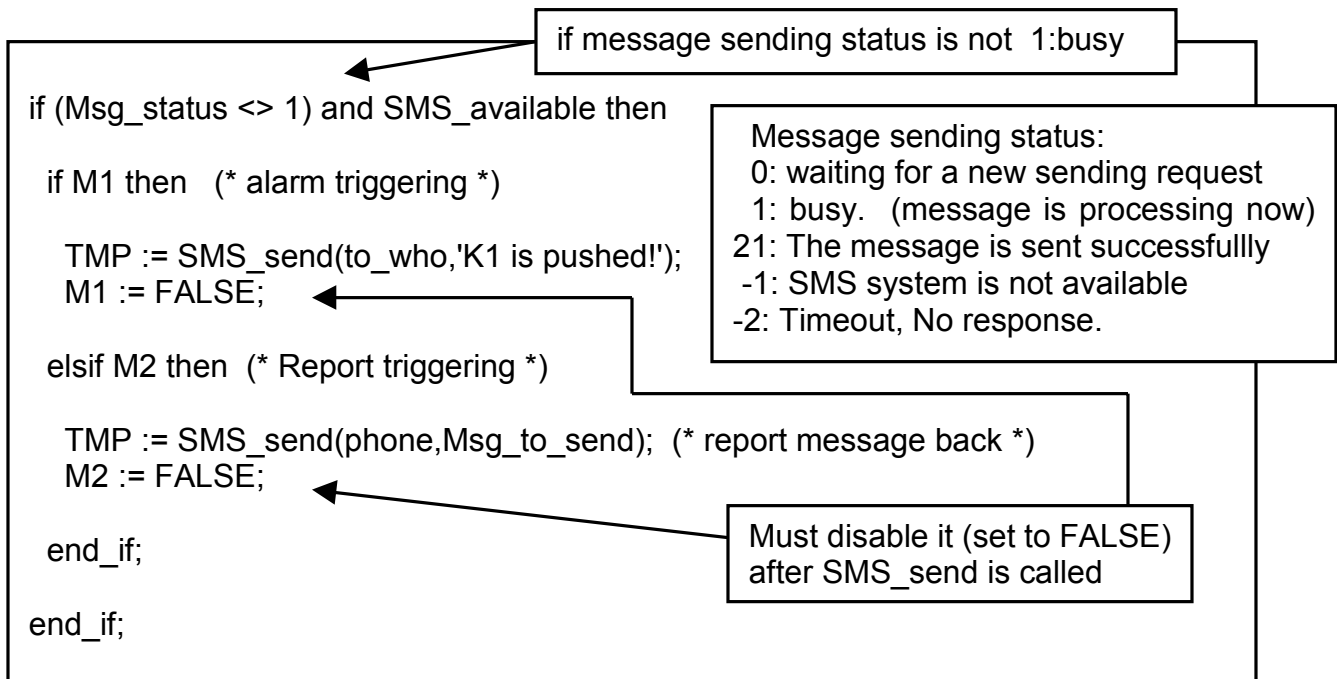
LD program : work



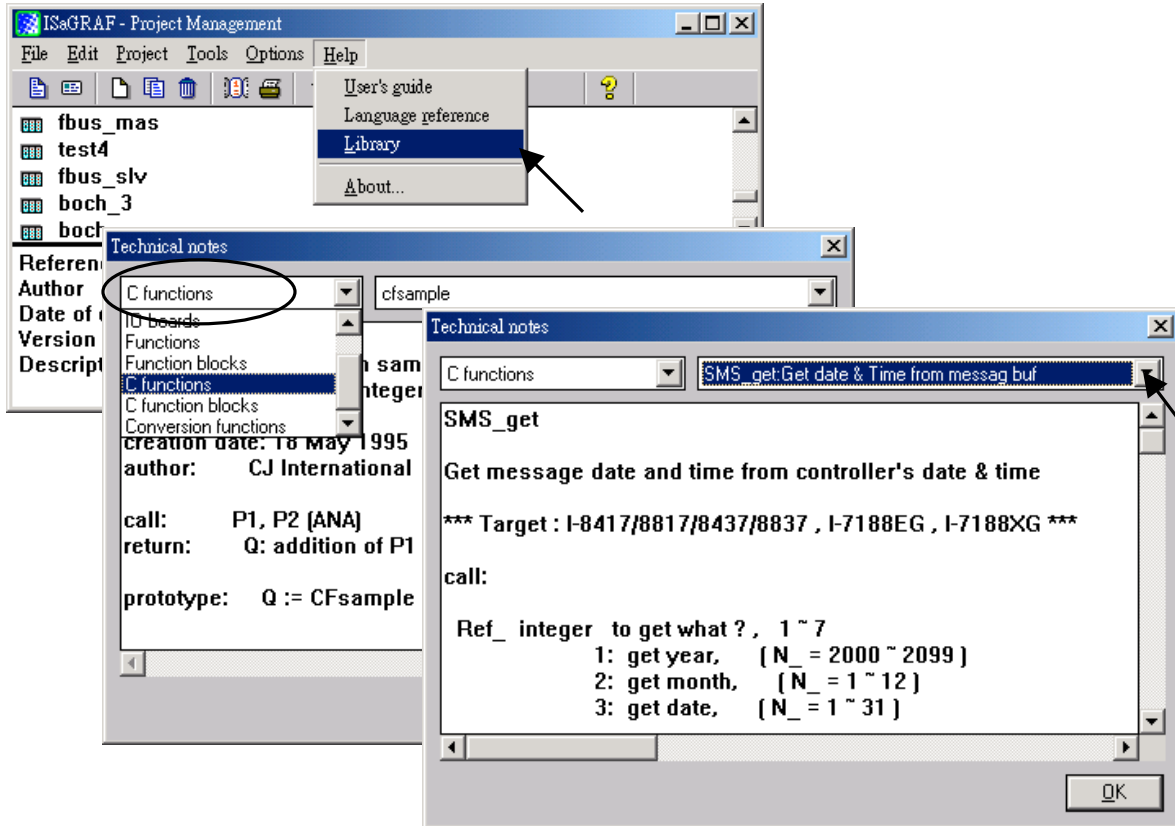
ST program : rcv_msg



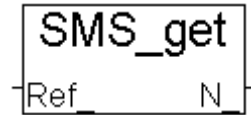
ST program : snd_msg



More description of SMS_sts, SMS_send, SMS_test, SMS_get & SMS_gets, Please refer to ISaGRAF's On-line Help. "Library" – "C functions" – "SMS_xxxx"



SMS_GET



Description:

Function Get message date and time from controller's date & time

Arguments:

REF_ Integer to get what ? , 1 ~ 7

- 1: get year, (N_ = 2000 ~ 2099)
- 2: get month, (N_ = 1 ~ 12)
- 3: get date, (N_ = 1 ~ 31)
- 4: get week date, (N_ = 1 ~ 7, 7 means Sunday)
- 5: get hour, (N_ = 0 ~ 23)
- 6: get minute, (N_ = 0 ~ 59)
- 7: get second, (N_ = 0 ~ 59)

others: return N_=-1 : error

return:

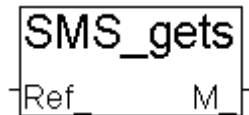
N_ Integer Return associated with Ref_. If return -1, it may be "No message" or Ref_ out of range of 1 ~ 7

Note:

1. SMS_gets & SMS_get can be called to get message
2. After SMS_gets(1) is called (get message data), the message buffer will reset to "No message". So if the other information are need, please call SMS_get(1~7) & SMS_gets(2) & SMS_gets(3) before calling SMS_gets(1)

Example: demo_43

SMS_GETS



Description:

Function Get message data and other information

Arguments:

REF_ Integer to get what ? , 1 ~ 3

- 1: get message data
- 2: get phone No. of sender
- 3: get date & time in string format

others: return M_ = 'error'

return:

M_ Message Return associated with Ref_. If return 'error', it may be "No message" or Ref_ out of range of 1 ~ 3

Note:

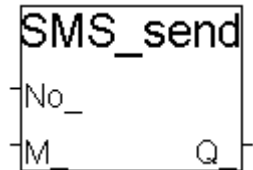
1. SMS_gets & SMS_get can be called to get message
2. After SMS_gets(1) is called (get message data), the message buffer will reset to "No message". So if the other information are need, please call SMS_get(1~7) & SMS_gets(2) & SMS_gets(3) before calling SMS_gets(1)

Example: demo_43

SMS_SEND

Description:

Function Trigger the controller to send a new message



Arguments:

No_ message to which phone No. , fro ex. '+886920119135', max len is 31 digits

M_ message the message to send

return:

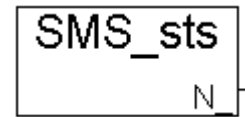
Q_ Boolean True: ok. , False: wrong phone No or "message sending status" is not 0 or 21

Note:

1. Please call SMS_sts to get the "Message Sending status" before calling SMS_send. SMS_send only works when status is not 1:busy
2. A successfully SMS_send request will reset the "Message sending status" to "1:busy", and after that, by the time, it will set to the associate status. For ex. 21:successfully sent

Example: demo_43

SMS_STS



Description:

Function **Get Message Sending status**

return:

N_ Integer

0: waiting for a new sending request
1: busy. (One message is processing now)
21: The message is sent successfully

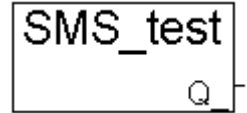
-1: SMS system is not available (Check GSM Modem & SIM card)
-2: Timeout, No response. (It May be no such a phone No.)

Note:

1. Please call SMS_sts to get the "Message Sending status" before calling SMS_send. SMS_send only works when status is not 1:busy
2. A successfully SMS_send request will reset the "Message sending status" to "1:busy", and after that, by the time, it will set to the associate status. For ex. 21:successfully sent

Example: demo_43

SMS_TEST



Description:

Function Test if message coming or not

return:

Q_ **Boolean** TRUE: A message is coming, FALSE: No message

Note:

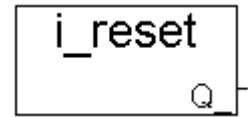
1. SMS_gets & SMS_get can be called to get message
2. After SMS_gets(1) is called (get message data), the message buffer will reset to "No message". So if the orther information are need, please call SMS_get(1~7) & SMS_gets(2) & SMS_gets(3) before calling SMS_gets(1)

Example: demo_43

I_RESET

Description:

Function Reset the controller



***** Target: I-8417/8817/8437/8837**

return:

Q_ boolean The return value has no meaning since the controller will reset

Note:

Please use this function very careful. If the controller is always reset, please refer to section 1.3.7 of the "User's Manual Of The I-8417/8817/8437/8837" to delete the project inside the controller.

Example:

```
(* OK1 is declared as boolean input, TMP as boolean internal *)  
if OK1=TRUE then  
  TMP := i_reset();  
end_if;
```

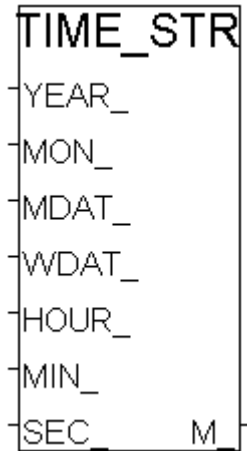
TIME_STR

Description:

Function Convert date & time to string format

Arguments:

YEAR_ integer year, 2000 ~
MON_ integer month, 1 ~ 12 (January ~ December)
MDAY_ integer mday, 1 ~ 31
WDAY_ integer wday, 1 ~ 7 (Monday ~ Sunday)
HOUR_ integer hour, 0 ~ 23
MIN_ integer minute, 0 ~ 59
SEC_ integer second, 0 ~ 59



If given wrong input parameters will return M_ = " (empty string). For. ex. give MON_=14

return:

M_ message length is 24 characters. For ex. 'Feb/18/2003,13:25:45,Tue'

Note: Please use sysdat_r & systim_r to get system date & time