dataTaker

Technical Note

TN-0032

Connecting a *dataTaker* DT80 Range Data Logger to an SMSX GSM/GPRS modem

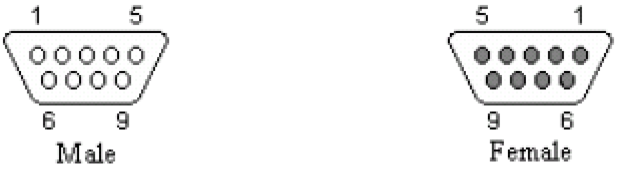
1 Prerequisites

- Nil

2 Required Equipment

- *dataTaker* DT8x data logger
- SMSX GSM/GPRS modem with standard accessories
- Active SIM card
- Serial cable (RS-232 DE-9)

The serial cable is required to have the (RS232 DE-9) pin out:



Pin	Name	Direction	Description
1	CD	→	Carrier Detect
2	RX	→	Receive Data
3	TX	←	Transmit Data
4	DTR	←	Data Terminal Ready
5	GND		System Ground
6	DSR	→	Data Set Ready
7	RTS	←	Request to Send
8	CTS	→	Clear to Send
9	RI	→	Ring Indicator

3 Process

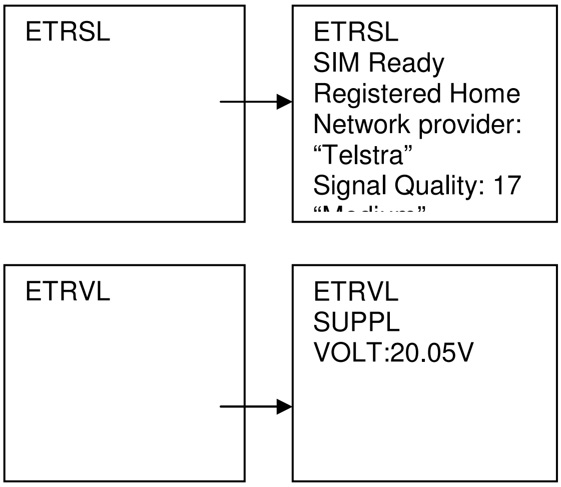
3.1 Basic set up of SMSX modem

- Insert the SIM card into the SMSX modem
- Connect the antenna
- Connect the power supply
- Wait until the green light begins to flash slowly (may take 1-2 minutes)

NOTE: If the green (status) light fails to flash slowly, check your antenna connection and sim-card and ensure you have adequate network coverage.

3.2 Confirm the SMSX unit is operating correctly

Using a mobile phone, send an SMS message to the SMSX phone number. The message should contain only the characters “ETRSL” (without the quote marks). You should receive a reply with the current radio signal level. If the signal is in the “Low” category then you will need to adjust or replace the antenna.
Send an SMS containing “ETRVL” (again without the quote marks) to the SMSX phone number. This will return the current voltage level via SMS.



ETRSL

ETRSL
SIM Ready
Registered Home
Network provider:
“Telstra”
Signal Quality: 17
“Medium”

ETRVL

ETRVL
SUPPL
VOLT:20.05V

If no reply is received for either of the above SMS messages, send “ETPM1” to the unit. If still no reply, insert the SIM in your mobile phone and ensure that you are able to send and receive SMS messages with your phone in the normal manner. If your SIM requests a PIN number you will need to:

- Locate the ETM configuration tool software on the supplied CD
- Connect the SMSX module directly to your PC
- Read the unit configuration by clicking EEPROM>EEPROM read
- On the ‘Main Init’ tab locate the SIM PIN cell and enter your SIM PIN number.
- Click EEPROM>EEPROM write.

3.3 Set up the dataTaker DT8x

The dataTaker needs to be set up to communicate with the SMSX. To do this, you need to send the following profile settings to the dataTaker using the command interface of either the web interface or DeTransfer.

```
PROFILE "HOST_PORT" "BPS"="9600"
PROFILE "HOST_PORT" "DATA_BITS"="8"
PROFILE "HOST_PORT" "STOP_BITS"="1"
PROFILE "HOST_PORT" "PARITY"="NONE"
PROFILE "HOST_PORT" "FLOW"="HARDWARE"
PROFILE "HOST_PORT" "FUNCTION"="COMMAND"

PROFILE "HOST_MODEM" "DIAL"="ETSEND=ATD"
PROFILE "HOST_MODEM" "INIT"=""
PROFILE "HOST_MODEM" "EXT_POWER_SWITCH"="NONE"
PROFILE "HOST_MODEM" "SEND_BANNER_ON_CONNECT"="NO"
```

3.4 Connect the SMSX module to the dataTaker DT8x

Use the RS-232 serial cable to connect the SMSX to the dataTaker serial host port (NOTE: Ethernet connection is not required for this application)

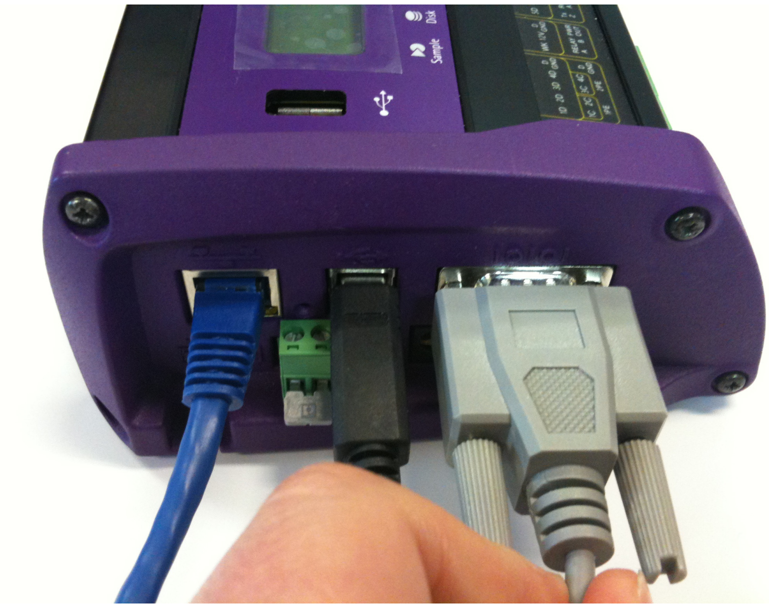


Figure 1 – Serial RS-232 connection to the dataTaker DT80

3.5 Confirm the logger is configured correctly

3.5.1 dataTaker to modem communications test

Using the command interface, send the following command to the dataTaker (Where 0412345678 should be replaced by your mobile phone number):

```
DO'et-sms=0|This is the dataTaker talking |0412345678|60^M'
```

Shortly afterwards, you should receive the message “*This is the dataTaker talking*”, on your mobile phone.

Now type (again replacing the number 0412345678 with your mobile phone number):

```
REFT(=1CV) DO'et-sms=0|Temp = ?1F1 DegC|0412345678|60^M'
```

Shortly afterwards, you will receive the message “*Temp=xx.x DegC*” (where xx.x is the internal temperature of the dataTaker) on your mobile phone.

3.5.2 Modem to dataTaker communications test

Send the message “TEST” (without the quotes) to the SMSX from your mobile phone. You will not receive a response from the dataTaker because you have not written a program to do this yet, however you will hear the dataTaker clicking noise as it performs a self-test.

This indicates that you have correctly configured the dataTaker and the SMSX modem!

3.6 Creating a program to generate simple SMS alarms

NOTE: The SMSX requires a delay of at least 4 seconds between subsequent SMS attempts. If you are utilizing receipts then even more delay time is required. It is possible to use the DELAY command to force a pause between SMS messages, however the DELAY command also pauses activity in all other schedules. If you have schedules that operate faster than 5 seconds, or have several schedules which generate SMS alarms, or believe that the DELAY command will significantly affect your timing, it is highly recommended that you implement the method used in section 3.6.2.

3.6.1 Basic, low-speed SMS alarm application

This program takes a thermocouple measurement every 30 seconds and sends an alarm to the mobile number 0412345678 when the temperature increases to above 28 degrees. The alarm statement chosen is the regular, non-repeating type (ALARM), therefore the SMS will not be re-sent until the condition has been reset and is triggered again.

```
BEGIN"SMSTEST1"
RA1M
    1TK(=1CV)
    ALARM(1CV>28)'ET-SMS=0|Temp > 28 DegC (?1F2)|0412345678|60^M'
END
```

Breaking this code down, we look closely at the ALARM text, which consists of:

- The **ET-SMS=0** AT command (0 = no receipt required)
- The pipe separator |
- The SMS text message **"Temp > 28 DegC (?1F2)"**, followed by a pipe
- The phone number to send , followed by a pipe
- The timeout period (ignored is no receipt required), followed by the carriage return **^M**

The part of the alarm text that contains “?1F2” is a channel variable reference. These can be used to insert data into the text message, in this case it returns channel variable 1 to 2 decimal places. For more information on inserting data into the messages, please consult the DT8x user manual.

3.6.2 Advanced or high-speed SMS alarm application

The advanced SMS alarm application has a different concept altogether. In this case schedule K runs exclusively to service the SMS messaging. Because this schedule is run at 5-second intervals and therefore has the delay built-in.

In the program below there are two schedules that generate alarm flags (A and B), which get serviced by the SMS handler (K) schedule. To prevent SMS requests from colliding, the handler services only one alarm flag at a time, clearing the individual alarm flag when it does, then when the schedule runs next it services the next alarm flag.

```
BEGIN"SMSTEST2"
    101..102CV=0              'reset alarm flags
RA1S
    1-TK("Sensor 1",=1CV)     'take measurement
    ALARM(1CV>30){101CV=1}    'set alarm flag number 1
RB10S
    1+TK("Sensor 2",=2CV)     'take measurement
    ALARM(2CV>30){102CV=1}    'set alarm flag number 2
RK5S
    105CV(W)=0                'SMS sent flag
    ALARM(105CV<1)AND ALARM(101CV>1)'ET-SMS=0|Alarm Message 1
|+61412345678|30^M'{105CV=1 101CV=0}
    ALARM(105CV<1)AND ALARM(102CV>1)'ET-SMS=0|Alarm Message 2
|+61412345678|30^M'{105CV=1 102CV=0}
END
```

For example, all flags were set (101CV=1 and 102CV=1), the handler schedule would first send the first SMS ("Alarm message 1") out, clear the first flag (101CV=0, 102CV=1), run again after 5 seconds, send the second SMS and clear the second flag (101CV=0, 102CV=0).

NOTE: Do not use repeating alarms because it may prevent latter alarm SMS messages from being sent, it may also result in SMS messages being sent every 5 seconds.

3.7 Other SMS commands

3.7.1 Sending an SMS to multiple recipients

It is possible to send an SMS to multiple phone numbers with a single command. This is accomplished by replacing the phone number in the ET-SMS command by a list, separated by a colon ":".

```
'ET-SMS=0|Message goes here|+61412345678:+61412378945|60^M'
```

This can be incorporated into any of the above examples, or into your own program.

3.7.2 Sending an SMS with a receipt request

Delivery receipts are used by many phone networks to identify when a message has successfully been received. This is accomplished by changing the message 'type' specified after the equals sign (ET-SMS=*n*).

The type can be one of the following:

- 0 if no receipt is required (timeout field is ignored)
- 1 if a receipt is required from ANY number in the list
- 2 if a receipt is required from ALL numbers in the list

NOTE: Mobile networks do not guarantee that a delivery receipt will be issued when requested and therefore for sensitive applications this should not be relied upon.