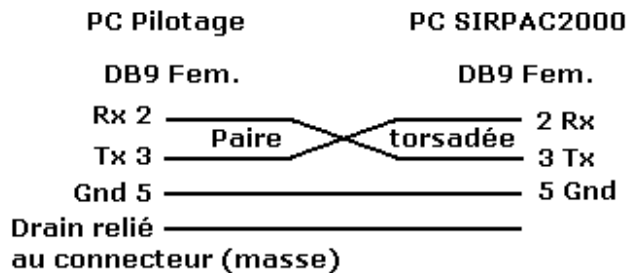


**CONTROL OF SIRPAC2000 OVER  
AN EXTERNAL RS232  
CONNECTION (LE)**

## Control of SIRPAC2000 over an external RS232 connection

### Wiring diagram for connection between Sirpac2000 computer and controlling computer

See below for wiring diagram to be followed for a connection between the two computers via RS232 using 9-pin serial sockets.



### Configuration parameters for Sirpac2000 to allow remote control

- ❖ Go into the Sirpac2000 configuration screen (Screen F6)
- ❖ Click on the computer icon at the top left
- ❖ Click on the Ports button (Uppermost row of buttons)
- ❖ Set the port to which Sirpac2000 is connected to the control software in SLAVE mode (Right hand list)
- ❖ The "RS Mode" value must be RS422
- ❖ The speed of the port must be between 600 and 57600 baud (depending on your software, generally 9600 or 19200 baud)

### **Communications parameters for remote control of Sirpac2000 via an RS232 external connection:**

To test the remote control, you can use the software we provide (running under MS-DOS) which can be found in the following directory: C:\SIRPACW\DIVERS\TEST232 (LE). The program is called Test232.exe. It should be run with a parameter list as described below:

- ❖ **Speed:** Variable, depending on the speed at the COM port in Sirpac2000 (between 600 and 9600 baud for Test232, which cannot handle higher speeds)
- ❖ **Parity:** N
- ❖ **Data bits:** 8
- ❖ **Stop bits:** 1
- ❖ **ASCII Code for end of line (LF):** 10

Example: **Test232 1,9600,N,8,1,10**

### **Message format**

The messages sent to Sirpac2000 must be of the type:

XX[.](LF)

XX (2 characters) represents the command code.

[.] represents optional command parameters

(LF) is the delimiter character which must always be added to the end of the command line sent.

Sirpac2000 may answer:

- XX[responses](LF) *In the case of a request for information*
- XX[.](LF) *Simple echo of the message received in the case of a command.*
- ??( LF) *If the message has not been understood or could not be executed*
- X??( LF) *Message not understood on cabinet X (Multi-cabinet mode)*

For an information query, Sirpac2000 responds extremely fast (a few tens of milliseconds).

For a command, Sirpac2000 responds as soon as it has been executed. This is very fast in the majority of cases but some commands involve disk access, which may take a second or two. We would therefore recommend in general that time out settings for the responses should be around 5 seconds. If Sirpac2000 does not respond within 5 seconds, you can be sure that there is a communication problem.

### **Monitoring communications**

In the Desktop (Screen F2), type the word 'spy' on the keyboard. A dialog box appears offering a number of options for monitoring communications. Click on 'Monitor communications in real time'. You then have access to various options which will enable you inter alia to monitor communications between Sirpac2000 and the external connection

Let's take a look at the list of commands which can be used to control Sirpac2000.

## DATA READINGS

### Read temperature

Function: Priority temperature acquisition.

Code: LT

Response: LT±(value)

eg: LT+200.0 (200°C), LT-5.2 (-5.2°C)

### Read humidity

Function: Priority acquisition of humidity level.

Code: LH

Response: LH(value)

eg: LH91.2 (91.2%), LH8.0 (8%)

### Logical inputs

Function: Acquisition of the status of one or more logical inputs on the CITOR card.

(1) One line:

Code: EL(line)

Response: EL(line)(O/F)

eg: EL5O (input 5 open), EL13F (input 13 closed)

(2) All lines:

Code: EL

Response: EL(line status)

eg:

- ELOFOFOOFF (8 lines: EL1 open, EL2 closed, etc. EL8)

### Logical outputs

Function: Acquisition of status for the logical feed outputs of the CITOR card.

(1) One line:

Code: SL(line)

Response: SL(line)(O/F)

eg: SLO (output 5 open), SL13F (output 13 closed)

(2) All lines:

Code: SL

Response: SL(line status)

eg:

- SLOFOFOFFOFF (11 lines: SL1 open, SL2 closed etc. SL11)

### Events

Function: Acquisition of event status

(1) One line:

Code: EV(line)

Response: EV(line)(O/F)

eg: EV5O (event 5 triggered), EV13F (event 13 deactivated)

(2) All lines:

Code: EV

Response: EV(event status)

Example:

- EVOFOFOFFOFF (11 events: EV1 activated, EV2 deactivated, ...)

### Analogue inputs

Function: Acquisition of measurements from an analogue input (Particularly useful for the management of an acquisition centre).

Code: EA(line)

Response: EA(line)±(value)

Reading is accurate to 4 significant figures maximum (floating point)

eg: EA6+8.76, EA11-123.2, EA8+1421, EA1-0.099

## **OUTPUT ACTIONS**

### **Set a logical output**

Function: Set the status of a logical output of the CITOR card (open/close a contact)

Code: WL(O/F)(line)

Response: ditto (echo)

eg: WLO7 (opens contact SL7), WLF10 (closes contact SL10)

### **Set a logical output with inhibition**

Function: As for the WL command above but prevents actions by anything other than Sirpac2000 on this output. The output status is thus retained.

Code: IL(O/F)(line)

Response: ditto (echo)

eg: ILO7 (opens the contact SL7), ILF10 (closes the contact SL10)

### **Terminate inhibition of a logical output**

Function: Removes the inhibition set up using the function IL above.

Code: AL(line)

Response: ditto (echo)

eg: AL6, AL11

## **ACTIONS ON EVENTS**

### **Set an event**

Function: Control the status of an event (activate/deactivate an event)

Code: WE(O/F)(line)

Response: ditto (echo)

eg: WEO7 (activates event EV7), WEF10 (deactivates event EV10)

## **START/STOP COMMANDS**

### **Launch a cycle**

Function: Starts a cycle en programmed or manual mode and starts up the cabinet.

Code:

MAP(program)

or

MAM(temperature),(humidity),(Duration)

(temperature in °C, humidity in % or absent, duration in seconds)

Response: Ditto (echo)

eg:

Programmed mode: MAPTEST (launch using program 'TEST')

Manual mode: MAM80,90,3600 (launch with settings 80°C and 90% for 1 hour)

MAM-40,,3600 (launch at -40°C without humidity management for 1 hour)

### **Stopping a cycle**

Function: Stops a cycle before it reaches its normal conclusion.

Code: ARS or ARNN

Response: Ditto (echo)

eg:

ARS (stop, saving the cycle carried out)

ARN (stop without saving, not generally used under Sirpac2000, retained for compatibility with Sirpac DOS)

## INFORMATION ON THE CYCLE

### Operating status

Function: Query the current status of the cycle and the cabinet.

Code: EF

Possible cases:

- Programmed mode cycle in progress  
Response: EFP(Program)  
Example: EFPTEST (cycle running using program 'TEST')
- Manual mode cycle in progress  
Response: EFM
- Cycle connecting with program  
Response: EFRP(Program)
- Cycle waiting for start up  
Response: EFI
- No cycle running  
Response: EFN
- Cycle paused  
Response: EFPAUSE
- Cycle running, cabinet stopped due to error  
Response: EFD(Error)  
Error identifies the error detected and can be:
  - EV(no) number of event triggered  
Example: EFDEV3 (event 3)  
**Note: In the preceding version of the LE model, EFDEV3 corresponded to event 4**
  - A software error identified by:  
OVER: overshoot  
FOUR: out of range  
RAMP: ramp monitoring  
SECT: safe temperature  
SNDT: temperature probe  
SNDH: humidity probe  
SECH: safe humidity  
Example: EFDFOUR, EFDSECT

### Total time elapsed

Function: Query the time elapsed since the start of the cycle.

Code: TE

Response: TE(time in minutes)

Example: TE30 (30 minutes elapsed)

### Total time remaining

Function: Queries the time remaining for the cycle.

Code: TR

Response: TR(time in minutes)

Example: TR120 (2 hours left to run)

### Length of the cycle

Function: Query the length of the cycle.

Code: TT

Response: TT(time in minutes)

Example: TT240 (Length of the cycle: 4 hours)

## INFORMATION ON THE OPERATION POINT

### Temperature settings

Function: Query the current temperature settings.

Code: CT

Response: CT±(Value)

eg: CT+200.0 (200°C), CT-7.2 (-7.2°C)

### Humidity settings

Function: Query the current humidity settings.

Code: CH

Response: CH(Value) or CHN (if humidity not managed)

eg: CH75 (75%), CH5 (5%), CHN (not managed)

### Segment type

Function: Query the current segment type.

Code: SE

Response:

SEPAL±(Temperature) for a plateau

or

SERMP±(Gradient),±(Final setting) for a ramp

eg: SEPAL+50.0 (plateau at 50°C)

SERMP-1.20,-30.0 (ramp descending to -30°C with a gradient of 1.2°C/min)

### Segment number

Function: Query the current segment number.

Code: SN

Response: SN(Segment)

Example: SN1 (1st segment), SN12 (12th segment)

### Time elapsed in segment

Function: Query the time elapsed since the start of the current segment.

Code: ES

Response: ES(Time in seconds)

Example: ES60 (1 minutes elapsed), ES3600 (1 hour elapsed)

### Time remaining in segment

Function: Query the time remaining until the end of the current segment.

Code: RS

Response: RS(Time in seconds)

Example: RS60 (1 minute to go), ES3600 (1 hour to go)

### Segment duration

Function: Query the duration of the current segment.

Code: DS

Response: DS(Time in seconds)

Example: DS60 (1 minute), DS3600 (1 hour)

### Connection information

Function: Query whether a connection is in progress.

Code: RA

Response: RAO (connection in progress), RAN (no connection in progress)

## MAKING CHANGES TO THE CURRENT CYCLE

### New segment in manual mode

Function: Terminates the current segment and creates a new segment or changes the length of the cycle.

Code: MC(Temperature),(Gradient),(Humidity),(Time remaining)

- *Temperature*  
New temperature setting in °C. If omitted, the new segment retains the current value. If the gradient is not specified, a new plateau is created. If a gradient is specified, a ramp is created.
- *Gradient*  
Gradient in °C/min to create a ramp. If Gradient = 0 or is not given, a plateau is created. A temperature must also be given after a gradient is indicated.
- *Humidity*  
New humidity setting in %. If omitted, the new segment retains the current value. If "N" is returned, humidity is no longer being managed. A new segment is always created.
- *Time remaining*  
Time remaining until the end of the cycle in seconds. This effectively changes the length of the last segment (a new segment is not created)

All the parameters are optional but the commas must be included.

Response: Ditto (echo)

eg:

MC100 Creation of a plateau at 100°C. Humidity setting unchanged.

MC,,40 Change of humidity setting to 40% for a plateau.

MC,,,3600 Change time remaining: 1 hour (segment length).

MC20,4 Creation of a ramp to change to 20°C at 4°C/min, humidity unchanged.

MC-10,0.5,N Creation of a ramp to change to -10°C at 0.5°C/min, humidity not managed.

### Changes to segment length in programmed mode

Function: Changes the current segment duration if this is of the plateau type.

Code: DR(Time remaining in seconds)

Response: Ditto (echo)

Example: DR0 (Premature termination of the plateau), DR1800 (set to the time remaining for the plateau to 30 minutes)

## MISCELLANEOUS COMMANDS

### Display a message

Function: Causes Sirpac2000 to display a scrolling message on the Desktop

Code: AF(message) The message may not exceed 98 characters

Response: Ditto (echo)

Example: AFHello World via the RS232 (Sirpac2000 displays "Hello World via the RS232")

The basic AF command deletes the preceding scrolling message.

## CONTROLLING MULTI-CABINET CONFIGURATIONS

If your Sirpac2000 software is controlling multiple cabinets from a single PC (multi-cabinet mode), you can send commands to the cabinet of your choice by specifying first the cabinet number, followed immediately by the command.

Example:

- 2LT (Temperature reading for cabinet 2)
- 10RS (Query time remaining in the current segment for cabinet 10)

If whilst operating in multi-cabinet mode you do not specify the cabinet number at the start of the command line, it is the currently selected cabinet which will process the command.