

T+A

DAC 8

RS 232 Control

CONTENTS

- 0 GENERAL**
- 1 HARDWARE**
- 2 PROTOCOL**
 - 2.1 Physical layer
 - 2.2 Data Link Layer
 - 2.3 Application Layer
- 3 COMMANDS**
 - 3.1 Command Overview
 - 3.2 Command Details
- 4 RESPONSES**
 - 4.1 Prompt
 - 4.2 Query Responses
 - 4.3 Query Response Format
- 5 NOTIFICATIONS**
 - 5.1 Error Notifications
 - 5.2 Status Notifications
 - 5.3 Notification Level
- 6 FIRST STEPS**

General

This document defines a simple ASCII-based (but 8 bit) control interface via RS-232C with the following properties:

- Simple enough for to be emulated by any terminal program
- Readable command names
- Numbers in commands transferred as ASCII strings not as binary data
- Notifications about the current settings and state of **DAC 8** transmitted as ASCII strings or binary data
- Asymmetric at the application layer (commands vs. response / notification)
- Asymmetric at the data link layer (one-sided flow control only)
- No error protection

The control interface can be used to control a **DAC 8** from a PC or any other *ControlPoint* equipped with a RS232 interface. USB to RS232 converters can be used if the ControlPoint does not provide a RS232 interface.

1 Hardware

The control interface uses a RS232 type of interface. The socket on the rear panel of the **DAC 8** is a RJ12 socket with the following pinning:

Pinout DAC_8 / RS232 adapter cable		
Pin No. (RJ-12)	Pin No. (Sub-D 9 - female)	Signal
2	5	GND
3	2	RxD (DAC_8 → PC)
5	3	TxD (PC → DAC_8)
1	NC	DAC_8 Trigger Out (+5V)

Notes:

For easy connection an adapter RJ-12 → SUB-D 9 (female) can be obtained from T+A as an accessory.

In case the controlling device (PC) does not feature a RS232 interface, a standard USB/RS232 converter can be used.

Pin 1 of the RJ12 socket carries a **+5V trigger voltage** that is high when DAC_8 is powered ON. This trigger voltage can be used to turn an external device (e.g. amplifier) ON.

2 Protocol

2.1 Physical Layer

Bidirectional communication by RS-232C lines TxD and RxD (no control lines used)

Port settings: Baud rate: 38400, 8 bit, no parity, 1 stop bit

2.2 Data Link Layer

The data link layer uses an asymmetric, connectionless and line-oriented protocol.

The **DAC 8** will echo received characters. The echo can be turned off by sending an „ECHO OFF“ command.

Information from the **DAC 8** is sent as message lines of max. 32 8-bit characters. Every message is terminated by a CR/LF combination.

Flow Control

The **DAC 8** will read in an entire data line of max. 32 8-bit ASCII characters, terminated by a termination character sequence CR/LF (0x0D, 0x0A). As soon as a line termination character sequence has been received the received data will be interpreted and executed by the **DAC 8**.

After processing the received data and when ready for a new command the **DAC 8** will send a prompt character >

After receiving the prompt character, the ControlPoint may send the next command.

2.3 Application Layer

The application layer transfers messages. The message types "*command*", "*response*" and "*notification*" exist. This classification has no effect on the notation of the messages. It describes only the functionality of the messages sent to or received from the device.

Commands are properly formatted ASCII strings sent from a ControlPoint to the **DAC 8**.

Responses are messages sent from **DAC 8** to the ControlPoint as a response to a query sent by the ControlPoint.

Notifications are messages generated by the **DAC 8** to inform the ControlPoint about changed device settings or in case of errors. Notifications can be set to different formats (ASCII or binary) or can be turned OFF completely.

Note:

If notifications are turned ON, the ControlPoint is automatically informed about all changes in device settings (changed inputs, changed volume level etc.), about errors and about or changes in the audio data stream received (sample rate changes etc.).

Instead of automatic notification the ControlPoint can also query device settings and status information from the **DAC 8** and will receive the requested information as response to the query.

3 Commands

A command is a string of 8 bit ASCII characters consisting of a command ID followed by one or more parameters. A command is terminated by a CR/LF (carriage return / line feed) sequence. String length is max. 32 characters. Command ID and parameters are separated by spaces (0x20).

Commands are not case-sensitive (i.e.: the commands „POWER ON“, „Power On“, „power on“ will have the same effect).

For some commands short forms exist (ex. „VOL“ instead of „VOLUME“)

All **DAC 8** commands are described in detail in this chapter.

3.1 Command Overview

Command ID	ID (short form)	Parameter(s)	Example	Comment	RS 232 Version
POWER	PWR	[ON / OFF / ?]	POWER ON	Switches DAC 8 ON	1.0
INPUT	I	[1...8 / ?]	INPUT 7	Selects Input no. 7 (AES/EBU)	1.0
VOLUME	V	{RAMP} [0...99 / ?]	VOL RAMP 70	Ramps VOLUME to 70%	1.0
OVS		[1...4 / ?]	OVS 3	Selects Oversampling no. 3 (Bezier 1)	1.0
INV		[ON / OFF / ?]	INV ON	Turns INVERT function ON	1.0
WIDE		[ON / OFF / ?]	WIDE ?	Returns WIDE Status	1.0
MUTE		[ON / OFF / ?]	MUTE ON	Switches analogue outputs OFF	1.0
ECHO		[ON / OFF / ?]	ECHO OFF	Turns RS232 ECHO OFF	1.0
NOTIFY	NTF	[0, 1, 2, / ?]	NOTIFY 2	Selects „All Notifications ON“ mode	1.0
STATUS	S	[?]	STATUS ?	Returns the current operation status and device settings	1.0
LED	L	[?]		Returns current LED state as ASCII string.	1.0
RATE	R	[?]		Returns current sample rate as ASCII string.	1.0
BRIGHTNESS	BRT	[0...10]	BRT 3	Sets display brightness to 30%	1.0
VER		[?]	VER ?	Returns the current RS 232 Version	1.0

Notes:

- 1.) Required parameters are enclosed in squared brackets ([...]).
- 2.) Parameters in curly braces ({ }) are optional.
- 3.) The column „ID (short form)“ contains alternative short commands (i.e. „POWER ON“ will have the same effect as „PWR ON“).
- 4.) The column “RS232 Version” contains the **DAC 8** firmware version since the command is available.

3.2 Command Details

3.2.1 POWER

This command switches the Device ON or OFF or queries the current power state.

Format: POWER [ON / OFF / ?]

Examples:

Query current power state:	POWER ?	or (short form):	PWR ?
Turn device ON:	POWER ON	or (short form):	PWR ON
Turn device OFF:	POWER OFF	or (short form):	PWR OFF

ID	Parameter(s)	Comment
POWER <i>or</i> PWR	[ON / OFF / ?]	<i>Switch or query the current power state.</i>
	ON	<i>Turn device ON</i>
	OFF	<i>Turn device OFF</i>
	?	<i>Query current power state</i>

3.2.2 INPUT

This command is used to switch the Device ON or OFF or to query the current power state.

Format: INPUT [1...8]

Examples:

Select input 1 (=S/P-DIF 1):	INPUT 1	or (short form):	INP 1
Query current input:	INPUT ?	or (short form):	INP ?

ID	Parameter(s)	Comment
INPUT <i>or</i> INP	[1 ... 8]	<i>Switch or query the current input.</i>
	1	<i>Select input 1 = S/P-DIF 1</i>
	2	<i>Select input 2 = S/P-DIF 2</i>
	3	<i>Select input 3 = S/P-DIF 3</i>
	4	<i>Select input 4 = S/P-DIF 4</i>
	5	<i>Select input 5 = TOS-Link (optical)</i>
	6	<i>Select input 6 = BNC</i>
	7	<i>Select input 7 = AES-EBU</i>
	8	<i>Select input 8 = USB</i>

3.2.3 VOLUME

Set the VOLUME control to a desired value or query the current volume setting.

The volume level can either be set immediately to a desired value or it can be ramped up/down from the current value to the desired value.

Note:

This command only has effect if the **DAC 8** is set to „VARIABLE“ output mode or if Headphone is plugged in.

Format: VOLUME {RAMP} [0 ... 99 / ?]

Examples:

Query current volume setting: VOLUME ?

Set VOLUME to 65: VOLUME 65 or (short form): VOL 65

Ramp-up VOLUME to 99: VOLUME RAMP 99 or (short form): VOL RAMP 99

ID	Parameter(s)	Comment
VOLUME or VOL	{RAMP} [0 ... 99 / ?]	Set <i>VOLUME</i> level. ! only possible in „VARIABLE“ output mode or when Phones plugged in
	0	Set <i>VOLUME</i> level to 0 (= mute output)
	1 ... 99	Set <i>VOLUME</i> level
	RAMP 44	Ramp up/down <i>VOLUME</i> to 44
	?	Query current volume level

3.2.4 OVS

This command selects the oversampling algorithm.

Format: OVS [1 ... 4 / ?]

Examples:

Query currently selected OVS algorithm: OVS ?

Select Bezier 1 (Bez1) oversampling : OVS 1

ID	Parameter(s)	Comment
OVS	[1 ... 4 / ?]	Switch or query the current oversampling.
	1	Select FIR 1 oversampling (FIR 1)
	2	Select FIR 2 oversampling (FIR 2)
	3	Select Bezier + IIR oversampling (Bez1)
	4	Select pure Bezier Spline oversampling (Bez2)
	?	Query current oversampling algorithm

3.2.5 INV

This command switches the absolute phase inversion ON or OFF or queries the current inversion state.

Format: INV [ON / OFF / ?]

Examples:

Query current inversion state: INV ?

Select inverted absolute phase : INV ON

ID	Parameter(s)	Comment
INV	[ON / OFF / ?]	Switch or query the current phase setting.
	ON	Select inverted phase
	OFF	Select non-inverted phase
	?	Query current phase setting

3.2.6 WIDE

This command switches the analogue reconstruction filters of the **DAC 8** to NORMAL or WIDE bandwidth mode or queries the current bandwidth state.

Format: WIDE [ON / OFF / ?]

Examples:

Query current bandwidth state: WIDE ?

Select normal bandwidth: WIDE OFF

ID	Parameter(s)	Comment
WIDE	[ON / OFF / ?]	Switch or query the current analogue bandwidth.
	ON	Select WIDE bandwidth mode
	OFF	Select NORMAL bandwidth mode
	?	Query current bandwidth setting

3.2.7 MUTE

This command switches the analogue outputs ON or OFF or queries the current output state.

Format: MUTE [ON / OFF / ?]

Examples:

Query current mute state: MUTE ?

Switch muting ON (=outputs OFF): MUTE ON

ID	Parameter(s)	Comment
MUTE	[ON / OFF / ?]	Switch or query the muting state.
	ON	Switch muting ON (= outputs OFF)
	OFF	Switch muting OFF (= outputs ON)
	?	Query current muting state

3.2.8 BRIGHTNESS

This command sets or queries the brightness of the display and LEDs of the **DAC 8**.

Format: BRIGHTNESS [1 ... 10 / ?]

Example:

Set the brightness to 80% : BRIGHTNESS 8

ID	Parameter(s)	Comment
BRIGHTNESS or BRT	[0 ... 10 / ?]	Switch or query the current display brightness.
	0	Switch the display OFF
	1 ... 10	Set the brightness to 10 .. 100 % in 10% increments
	?	Query current display brightness

3.2.6 STATUS

This command queries the current operating state of **DAC 8**.

DAC 8 will return Input Number (**\$INP:**), Sample Rate (**\$SRT:**), Volume Setting (**\$VOL:**) and LED state (**\$LED:**)

The STATUS query is a short form of the query sequence **INP ?, RATE ?, VOL ?, LED ?**

Format: STATUS [?]

Examples:

Query current **DAC 8** state: STATUS ?

Short form: S ?

ID	Parameter(s)	Comment
STATUS or S	[?]	Query the current DAC 8 operating state.
	?	Query current DAC 8 state. DAC 8 response: Input_No., Sample_Rate, Volume, LED_State

3.2.9 ECHO

This command switches the echo function for the RS232 interface ON or OFF or queries the current state.

If ECHO is set to „ON“ all characters sent to **DAC 8** via the RS232 interface are echoed. This is useful when sending commands to the **DAC 8** from a PC terminal program (like Hyperterm). The characters sent to the **DAC 8** are then displayed in the terminal window on the PC.

When **DAC 8** is controlled from a home automation system the echo function is normally not needed or the echoed characters might even interfere with the control system. In such cases the echo can be switched OFF.

Switching OFF the echo also minimizes traffic on the RS232 interface which increases performance and throughput.

After power-on the echo function is always set to ON.

It is recommended to switch it OFF after powering on the **DAC 8** by sending a „ECHO OFF“ command.

Format: ECHO [ON / OFF / ?]

Examples:

Query current echo state: ECHO ?
Switch echo function OFF: ECHO OFF

ID	Parameter(s)	Comment
ECHO	[ON / OFF / ?]	Switch or query the muting state.
	ON	Switch echo function ON (= Terminal Mode)
	OFF	Switch echo function OFF
	?	Query current echo state

3.2.10 NOTIFY

This command sets or queries the current notification level.

Format: NOTIFY [0 / 1 / 2 / 4 / ?]

Examples:

Query currently selected OVS algorithm: NOTIFY ?
Select (ASCII) string notification mode: NOTIFY 4

ID	Parameter(s)	Comment
NOTIFY	[0 ... 2 / ?]	Switch or query the current notification level.
	0	Switch all notifications OFF
	1	Only ERROR notifications
	2	All notifications ON (ERROR + STATUS notifications)
	?	Query current notification level

Notes

If notification mode 2 is selected **DAC 8** will send (push) a notification message whenever a setting or the state of the digital audio receiver changes.

By these push-notifications a ControlPoint can correctly display all relevant information without having to regularly request (poll) the current device state of the DAC 8.

Of course additional polling of the **DAC 8** status is possible when notification mode is set to 2.

After each mains power interruption the notification level will be reset to 1.

If push-notifications are wanted it is necessary to send a „NOTIFY 2“ command to the **DAC 8** after a mains interruption. It is a good practice to send the „NOTIFY 2“ command after powering ON the **DAC 8** to ensure push notifications will be sent.

QUERY Commands

Most of the **DAC 8** commands can be used to set or to query a certain setting. To query the current setting just use the appropriate command with a question mark (?) as parameter.

Example: the command **VOL ?** will return the current volume setting.

Besides the normal set / query commands there are some additional query commands which will not alter any settings just return the current status of the queried item.

Query responses

The responses to queries have the same format as *notifications* and are described in chapter „Notification Format“.

The following commands can be used to query the operating state of the **DAC 8**.

3.2.11 STATUS

This command queries the complete operating status of the **DAC 8**. It is a short form for the separate **INP**, **RATE**, **VOL** and **LED** queries.

The response will be:

INPUT, **VOLUME**, **SAMPLE_RATE** and **LED_STATE** as described further down below.

Example:

Query	Response
STATUS ?	\$INP: 01
or	\$SRT: 044
S ?	\$VOL: 00
	\$LED: 00A30260

3.2.12 LED

This command queries the current LED state of **DAC 8**. Response will be a string with the Response ID **\$LED:** followed by four hexadecimal bytes (= 8 ASCII characters) as parameter.

For explanation of the 4 parameter hex-bytes see table 4.3.1.

Example:

Query	Response
LED ?	\$LED: 00A32460
or	
L ?	

3.2.13 RATE

This command queries the sample rate of the currently received audio data stream. Response will be a ASCII string beginning with the Response ID \$SRT: followed by three ASCII characters as parameter.

Example:

Query	Response
RATE ?	\$SRT: 032 : 32 kHz
or SRT ?	044 : 44.1 kHz
	048 : 48 kHz
or R ?	088 : 88.2 kHz
	096 : 96 kHz
	176 : 176.4 kHz
	192 : 192 kHz

3.2.14 VER

This command returns the firmware version.

Example:

Query	Response
VER ?	\$VER: 01.00.00

4 Responses

4.1 Prompt

On receipt of a valid command **DAC 8** will execute the command and then will respond with a prompt character (>) when ready for the next command.

I.e.: After the prompt (>) is received the ControlPoint may send the next command.

4.2 Query Responses

The settings and the state of **DAC 8** can be queried by a number of query commands. Query commands consist of a command ID plus a question mark (?) as parameter.

Examples for query commands are „VOL ?“, „LED ?“ etc.

DAC 8 responds to a query with a response message. The format of response messages is described in chapter 4.3 further down below.

The difference between responses and notifications (see chapter 5) is that responses are sent as a reaction to a query from the ControlPoint whereas notifications are sent automatically by **DAC 8** in case of changes in its operation state.

Note

Responses are not affected by the setting of the notification level.

I.e. a response to a query will always be sent, even if notification level is set to „0“.

Responses can be used additionally to notifications.

Example:

When establishing a fresh connection to a **DAC 8** queries can be used to find out the current state of the **DAC 8**.

4.3 Query Response Format

Responses consist of a response ID followed by a parameter (ASCII string) containing the information.

Table xx: Overview DAC 8 Response Messages		
Response ID	Parameter(s)	Comment
\$LED :	[8 ASCII Char.]	LED Notification (ASCII String mode) <i>Informs about changes in the front panel LED status – for details see chapter 5.4.1.</i> <i>The 8 ASCII characters represent 4 Hexadecimal coded bytes. Each bit within these 4 bytes represents the state of one front panel LED. The status of the DAC 8 (selected input, state of jitter-bug, audio data error etc.) can be determined by evaluating the state of these LEDs.</i>
\$PWR :	[ON / OFF]	
\$INP :	[2 Byte – ASCII coded 2 digit number]	Input number 01...04 : S/P-DIF Coax-input 1 ... 4 05 : TOS-Link (optical) 06 : BNC 07 : AES/EBU 08 : USB
\$VOL :	[2 Byte – ASCII coded 2 digit number / LINE]	Volume Setting <i>This notification informs about changes in volume level (attenuator setting). If VRBL switch set to „LINE“ mode, „LINE“ is returned instead of VOLUME setting.</i> <i>The parameter value (00 ... 99 / LINE) corresponds to the DAC8 front panel volume display.</i>
\$SRT :	[3 Byte – ASCII coded 3 digit number]	Sample Rate information <i>This notification informs about the current sample rate. The value is the integer part of the sample rate (i.e. 000, 032, 044, 048, 088, 096, 176, 192) in kHz.</i>
\$MUT :	[ON / OFF]	Muting information ON : Muting ON (= outputs switched OFF) OFF : Muting OFF (= outputs switched ON)
\$WID :	[ON / OFF]	Setting of the analogue bandwidth ON : WIDE analogue bandwidth OFF : Standard analogue bandwidth
\$INV :	[ON / OFF]	Setting of the phase inversion
\$OVS :	[2 Byte – ASCII coded 2 digit number]	Oversampling information 01 : FIR1 oversampling (long FIR) 02 : FIR2 oversampling (short FIR) 03 : Bezier / IIR oversampling 04 : Bezier oversampling
\$BRT	[2 Byte – ASCII coded 2 dig. number]	Display Brightness 00..10 : Display Brightness 0...10
\$ECH :	[ON / OFF]	ECHO state ON : Received characters echoed (=“Terminal mode“) OFF : Received characters not echoed (=“ControlPoint mode“)
\$NTF :	[2 Byte – ASCII coded 2 digit number]	Notify Level 00 : all notifications OFF 01 : only ERROR notifications 02 : all notifications ON
\$VER :	[ASCII string]	Firmware Version

4.3.1 LED Response / LED Notification

The LED information parameter bytes consist of 4 bytes of data representing the state of the LEDs of the **DAC 8**.

From these 4 data bytes information about the current operating state of the **DAC 8** can be retrieved. The four data bytes are transmitted as a string of 8 ASCII characters representing the HEX notation of the 4 data bytes.

LED information contains information about:

- Power state
- Active input
- Reception state (receive error, jitterbug locking state, de-emphasis etc.)
- Oversampling algorithm
- Phase invert
- bandwidth setting of analogue filters
- Volume control mode + VRBL switch setting
- Phones connection state

Example - LED Response / Notification

Received ASCII data: LED\$: 00A30560

The HEX data string 00 A3 05 60 represents the binary number

0000 0000 1010 0011 0000 0101 0110 0000

Evaluating this bit pattern according to table 4.3.1 leads to the following result:

Byte 1 (0000 0000)	Variable switch: VRBL position
	Phones not plugged in
Byte 2 (1010 0011)	Digital Receiver LOCKED (Jitterbug stage 1)
	ERROR state: No Error
	Local Oscillator LOCKED (Jitterbug stage 2)
	NET control OFF
	Asynchr. mode OFF
	De-emphasis OFF
	Output OFF
	Volume Control ENABLED
Byte 3 (0000 0101)	Oversampling Bezier 1
	WIDE ON
Byte 4 (0110 0000)	Power state ON
	Input S/P DIF 1

TABLE 4.3.1

Response ID	Parameter(s)	Comment	
LED\$:	[8 ASCII char.]	Returns the state of all DAC 8 front panel LED indicators.	
	Byte 1	Bit 0	---
		Bit 1	---
		Bit 2	---
		Bit 3	---
		Bit 4	---
		Bit 5	---
		Bit 6	VRBL switch 1 := LINE 0 := VARIABLE
		Bit 7	Phones 1 := Phones plug inserted
	Byte 2	Bit 0	VOLUME-Control 1 := Volume Control enabled (VRBL or PHONES)
		Bit 1	OUTPUT 1 := output ON 0 := output OFF
		Bit 2	DEEMP 1 := De-emphasis ON
		Bit 3	ASY 1 := asynchronous USB mode
		Bit 4	NET * not yet implemented
		Bit 5	LO 1 := local oscillator ON (jitterbug stage 2)
		Bit 6	ERR 1 := audio data error / not valid
		Bit 7	LCK 1 := digital audio receiver locked (jitterbug stage1)
	Byte 3	Bit 0	WIDE 1 := wide analogue bandwidth mode ON
		Bit 1	Bez 2
		Bit 2	Bez 1
		Bit 3	FIR2
		Bit 4	FIR1
		Bit 5	INV 1 := absolute phase inverted
		Bit 6	Input USB
		Bit 7	Input AES
	Byte 4	Bit 0	Input BNC
		Bit 1	Input OPT
		Bit 2	Input SPD 4
		Bit 3	Input SPD 3
		Bit 4	Input SPD 2
		Bit 5	Input SPD 1
		Bit 6	ON-LED 1 := DAC 8 power ON
		Bit 7	---

5 Notifications

Notifications are messages generated by the **DAC 8** to inform the ControlPoint about changed device settings or in case of errors.

Notifications are terminated with CR/LF after a notification a prompt character (>) is sent to signal that the **DAC 8** is ready for a new command.

Notifications have the same IDs and format as the responses described in chapter 4.

Notifications can be turned OFF completely by setting the notification level to „0“ by sending a „NOTIFY 0“ command.

If notifications are turned ON, the ControlPoint is automatically informed about all changes in device settings (changed inputs, changed volume level etc.), about errors and about or changes in the audio data stream received (sample rate changes etc.). The level of notification information and notification format is depending on the notification level and can be set by sending an appropriate „NOTIFY“ command. For details refer to the description of NOTIFY command in chapter 3.

Note:

The notification level must be set after each interruption of the mains voltage by sending a „NOTIFY“. If notifications shall be received it is a good practice to send a „NOTIFY“ command after sending a „POWER ON“ command.

5.1 Error notifications

On receipt of an invalid command or parameter an ERROR notification message will be sent followed by CR/LF and a prompt character (>) as specified above.

Note:

Error notifications are always sent as ASCII strings.

Note:

When the notification level is set to 0 by the „NOTIFY“ command, ERROR messages will **not** be sent.

5.2 Status Notifications

STATUS notifications will be sent by the device depending on specific events, provided notifications have been enabled by setting the notification level to a value of 2 by sending a n appropriate „NOTIFY“ command.

5.2.1 LED-Status notification

A LED status notification has the notification ID „\$LED: “ (notification level 4 / ASCII string mode).

A LED notification is sent if the state of one or more LEDs on the **DAC 8** front panel changed.

By evaluating the state of the LEDs according to table 4.3.1 the state of the **DAC 8** (selected input, oversampling, reception state etc.) can be determined.

5.2.2 VOLUME notification

A VOLUME notification has the notification ID „\$VOL: “. A volume notification is sent, each time the volume setting changes.

5.2.3 SAMPLE_RATE notification

A SAMPLE RATE notification has the notification ID „\$SRT: “.

A sample rate notification will be sent if a change in the sample rate of the incoming audio signal occurred.

5.2.4 POWER notification

A POWER notification has the notification ID „\$PWR: “.

A POWER notification will be sent if the **DAC 8** is switched ON or OFF.

5.3 Notification level

The notification level determines the amount of notification information sent and its format according to the following scheme:

<u>Notification level</u>	0	all notifications disabled
	1	only error notifications
	2	turn on all notifications (Error + Status notifications)

6 First Steps

To get a „feeling“ for the control procedures and responses of a **DAC 8** we suggest to first connect the **DAC 8** to a PC and use a terminal program such as „Hyperterm“ or similar.

After connecting and choosing the correct RS232 interface parameters (see chapter 2.1, Physical Layer) hit „ENTER“ in your terminal and **DAC 8** should respond with a prompt (>).

Now you can try out some commands.

Example: typical DAC 8 Power-ON sequence		
Command	DAC 8 response	Comment
„Enter“	>	„ENTER“ to test communication and to see if DAC 8 is ready ...
ECHO	ON	Set ECHO to ON to receive an ECHO for each character entered in the terminal program. Keyboard input can be monitored this way ...
	>	PROMPT DAC 8 is ready for next command
NOTIFY 2	>	Set notification mode to output all notifications (ERROR + STATUS)
PWR ON	>DAC 8	„ DAC_8 “ Greeter-Message after POWER_ON
	>\$PWR: ON	Power state: Power is ON
	>\$LED: 00000040	LED-state: Power_LED is ON
	>\$INP: 01	Input No. 1 DAC_8 switched to S/P-DIF 1 input
	>\$OVS: 02	Oversampling OVS 02 (= FIR 2) in operation
	>\$WID: ON	Bandwidth Wide analogue bandwidth mode
	>\$LED: 40402960	LED-state: VRBL-switch in „LINE“ position (=volume control disabled) AudioData not valid absolute phase INVERTED FIR2 oversampling WIDE analogue bandwidth Power-LED is ON INPUT 1 selected
	>\$SRT: 000	Sample Rate: Unknown / not valid
	>\$MUT: OFF	Muting OFF Output switched ON
	>\$SRT: 048	Sample Rate: changed to 48 kHz
	>	PROMPT Ready for next command
WIDE OFF	>\$LED: 40A22860	LED-State: ... as above but WIDE-LED → OFF
	>	PROMPT Ready for next command