

**Control Interface Specification
for Chassis SL1xx
V3.0.0**

Status:

released

1 Revision History

Version	Date	Name	Department	Description
0.1	2005-08-05	HG	SW TV	Initial draft (tentative, not reviewed)
0.2	2006-09-21	AB	SW TV	Results of review on July 26 th 2006 added - baud rate setting via service menu - query values by '?' - add new commands
	2006-09-26	HG	SW TV	Formal changes
0.3	2007-03-16	AB	SW TV	- all supported port settings added - bass/treble changed to bass0,bass1/treble0,treble1 - software version added since a command is available - status values modified - command "frontkeys" exchanged by "keys" - Appendix 1 for IR codes added
0.4	2007-03-22	AB	SW TV	status parameter corrected
0.5	2007-04-26	AB	SW TV	- command "keys" replaced by command "config" - command "data" extended by different audio and video parameter - command "proglis" added - command "help": returns only list of available commands, no additional info for each command - IR codes for local keys added to IR table (Appendix)
0.6	2007-05-02	AB	SW TV	- commands "time" and "timer" added
0.7	2007-05-25	AB	SW TV	- AV channels for prog command: HDR and Photoviewer added - status command: substatus "uninit" added - power command: powermode "init" added - proglis command: parameter avtv and avradio added for reading additional inputs of tv and radio mode - increment and decrement for command data added (via +/-) - notification for command data added.
0.8	2007-05-20	AB	SW TV	some corrections
1.0	2007-06-06	AB	SW TV	- proglis command extended for favourite lists - data command: maxvolume added
	2007-06-11	AB	SW TV	- ir command: value of status bits for single command corrected; ir sequence for complete key press added
	2007-06-19	AB	SW TV	- data command: favourite added to get/set active favourites
1.1	2007-06-25	AB	SW TV	notify command: extended to set the format for notifications
1.1.1	2007-07-09	AB	SW TV	- Software version for command "notify format" corrected
2.0.0	2008-02-14	AB	SW TV	dynamic menus added
	2008-02-22	AB	SW TV	parameter pip and pipProg added for dynamic menu 'message'
	2008-03-30	AB	SW TV	command proglis extended by paging and radio favourites
	2008-04-01	AB	SW TV	command timer extended to program a timer for an arbitrary point of time
	2008-08-04	AB	SW TV	some IR codes added to appendix 1
2.0.1	2009-09-03	AB	SW TV	- ir command: Status bits corrected for simulating a complete key event - prog command: HDMI3 and MusicBox added to AV channels
3.0.0	2010-06-14	AB	SW TV	Modifications for chassis SL1xx added.

2 Open Items

3 Table of Contents

1	Revision History	2
2	Open Items	3
3	Table of Contents	3
4	Related Documents	4
5	General	4
6	Requirements	4
7	Protocol Format	5
7.1	Physical Layer	5
7.2	Data Link Layer	5
7.3	Application Layer	5
7.3.1	Common Message Structure	5
7.3.2	Command	6
7.3.3	Response	6
7.3.4	Notification	6
8	Message Lists	6
8.1	Commands	7
8.2	Responses	8
8.3	Notifications	9
8.4	Message Details	9
8.4.1	data	9
8.4.2	ident	10
8.4.3	ir	10
8.4.4	notify	11
8.4.5	power	12
8.4.6	prog	12
8.4.7	progdata	13
8.4.8	proglis	13
8.4.8.1	TV/Radio program list	14
8.4.8.2	Get tv/radio list page by page	14
8.4.8.3	Favourite lists	15
8.4.8.4	proglis notification	15
8.4.9	range	16
8.4.10	status	16
8.4.11	time	17
8.4.12	timer	17
8.4.13	config	18
8.4.14	help	18
8.4.15	version	18
8.4.16	msg	19
8.4.17	dmenu	19
8.4.17.1.1	Menu type "list"	19
8.4.17.2	Menu type "message"	19
8.4.17.3	Entry points	20
8.4.17.4	Command format	21
8.4.17.4.1	Subcommand define	22
8.4.17.4.2	Subcommand content	23
8.4.17.4.3	Subcommand status	24
8.4.17.4.4	Subcommand event	24
8.4.17.4.5	Subcommand entry	25
8.4.17.4.6	Subcommand undefine	25
8.4.17.5	Handling of Dynamic menus	25
8.4.17.6	Examples	26
8.4.17.6.1	Entry Points	26

8.4.17.6.2	Message.....	26
8.4.17.6.3	List	27
9	Configuration Data and Limit Values.....	28
10	AMX.....	28

4 Related Documents

Document	Description

5 General

This document defines a simple ASCII-based (but 8 bit) control interface via RS-232C.

The protocol has the following properties:

- Simple enough for to be emulated by any terminal program
- Readable command names
- Numbers transferred as ASCII strings, not as 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

Possible applications are:

- Remote control of TV sets in a hotel installation
- Remote control in a home AV installation
- Remote control of public monitors
- Service (with limitations)

This document version (3.0.0 or higher) is valid for chassis SL1xx only. The control interface for chassis L27xx is defined in version V2.x.y.

Differences to chassis L27xx:

- Standby/Wakeup commands:
With chassis SL121 a new standby concept has been introduced to fulfil the new power requirements for TV sets in standby (0.3 Watt). For this a standby processor is used which handles all events relevant for standby meanwhile the main processor is completely switched off. As some commands are required in standby (e.g. to switch on the TV set) a subset of the commands are supported also by the standby controller. The commands which are available in standby are listed in chapter 8.1.
- Status command:
To detect the new standby modes the status command has been adapted (see chapter 8.4.10).
- Dynamic menus – Entry points:
The assist menu has been substituted by the Media Portal. So the access to the entry points via the OSD has been changed. Entry points can't be set in standby anymore. See chapter 8.4.17.3.
- Command prog:
The negative numbers for the different media sources of the Media Player (MediaHome, MediaNet) have been adapted. See chapter 8.4.6).

6 Requirements

The RS232 interface is supported by SL1xx chassis from the beginning.

7 Protocol Format

7.1 Physical Layer

- Bidirectional communication by RS-232C lines TxD and RxD (no control lines used)
- Port settings:
 - modes:
 - 8 bit async
 - 9 bit
 - 8 bit parity
 - Parity: Even, Odd
 - Stopbits: 1 stopbit, 2 stopbit
 - Baud rates: 2400, 4800, 9600, 19200, 28800, 38400, 57600, 76800, 115200 bps
 - Default settings are: 9600 baud, 8 bit async, no parity, 1 stopbit

The port settings can be adjusted in the service menu of the TV set, menu item "RS232", "port settings".

This menu can also be opened by pressing the function key on the front panel (M or C) and simultaneously pressing key '7' of the remote control.

7.2 Data Link Layer

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

The TV set will read in an entire data line of max. RCI_LINE_LENGTH_LIMIT 8 bit characters, and interprets it as soon as a line termination character has been received. Both carriage return (CR, 0x0D) and line feed (LF, 0x0A) character will be accepted as line termination characters, a line feed character following a carriage return character will be ignored.

The TV set will not mirror received characters, and the counterpart must not mirror any received characters.

The TV sends either lines terminated by a CR/LF combination, or the prompt character RCI_PROMPT_CHAR as the first character of a new line (following a LF character). The counterpart should not send (i.e. terminate by CR and/or LF) a new line before the RCI_PROMPT_CHAR has been received, but type-ahead is guaranteed for at least up to another RCI_LINE_LENGTH_LIMIT bytes.

7.3 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 TV set.

7.3.1 Common Message Structure

A message consists of a message id followed by a space character, and optionally followed by a list of parameters.

The message id is a text string conforming to the pattern [A-Za-z][A-Za-z0-9_]*.

Parameters are decimal or hexadecimal numbers or text strings separated by a space. Numbers are decimal by default, hexadecimal numbers are preceded by a single ASCII quote.

Strings must be enclosed in double quotes if they contain spaces. The following control characters are supported using the backslash:

"\n":	line feed
"\r":	carriage return
"\t":	tab
"\"":	backslash
"\"":	double quote

The meaning of a parameter depends on its position in the list.

In the following required parameter are described in squared brackets ([...]). Optional parameters are enclosed by curly brackets ({...}) additionally.

7.3.2 Command

A command is sent to the TV set. It consists of one single line.

7.3.3 Response

A response is sent by the TV set immediately after receipt of a command. It may consist of one or more lines, the last line followed by the prompt character RCI_PROMPT_CHAR.

The content of the response to a valid command will be specific for that command, and has to be specified for each command. If an immediate response is not possible, only a RCI_PROMPT_CHAR will be sent for to allow subsequent commands, but the proper reply will be sent as a single notification later on. This is to be specified in each case.

On receipt of an invalid command, the ASCII "?" will be displayed (followed by CR/LF and RCI_PROMPT_CHAR as specified above).

7.3.4 Notification

Notifications will be sent by the TV set depending on specific events, provided the specific notification has been enabled by a preceding command. Notifications are terminated with CR/LF and the prompt character RCI_PROMPT_CHAR.

Since the RS232 version V1.1 the leading and/or trailing string for notifications can be defined arbitrarily. See the definition of the `notify` command.

8 Message Lists

Notes:

- All numbers in messages are ASCII, there are no binary data.
- Message texts are case-significant.
- The space character is generally used as a word separator. Strings in parameters have to be quoted if they contain spaces.
- Variable 8-bit text strings in message parameters are UTF-8 encoded unless specified otherwise. Any fixed texts are plain 7 bit ASCII.

8.1 Commands

For a detailed description of parameter values, see the additional tables/chapters under 8.4 Message Details. Parameters in curly braces are optional. The column "SL1xx Version" contains the software version of SL1xx chassis since the command is available.

Note: A command is always backward compatible when new features have been implemented with new TV software!

Standby and Wakeup Commands:

Due to the standby concept not all commands are supported in standby. There are two different types of commands which are supported in standby:

- **Standby commands:** Commands which are handled by the standby controller completely. There will be no significant delay for the response of the TV set.
- **Wakeup commands:** Commands which are accepted by the standby controller but which have to be handled by the main controller (e.g. for reading the program list via command `proglis`).
For this the main controller is waked up (without picture and sound). After it has been booted the command is handled. The boot up time is different for the different SL1xx chassis. Example: Chassis SL121 has a boot up time of about 7 seconds!
If there are no further commands for 5 seconds the main controller is switched off again. The different standby power states can be evaluated by the status command (see chapter 8.4.10).

The standby command type is described in the following table in the column "standby type":

standby = Standby command

wakeup = Wakeup command

none = Command is not supported in standby at all.

Id	Parameters	Example	RS232 version	SL1xx version	standby type
config	[ir 0/1/?] [localkeys 0/1/?]	config keys ir 0 localkeys 0	V1.0	PV1.0.0	standby
data	[name] [values] ... [name] [values]	data volume 17	V1.0	PV1.0.0	none
dmenu	[define/content/status/entry/undefined] ... see chapter 8.4.17		V2.0	PV1.0.0	none
help		help	V1.0	PV1.0.0	standby
ident		ident	V1.0	PV1.0.0	standby
ir	[system] [subsystem] [command] [status] [repeat]	ir 0 0 1 '02 1	V1.0	PV1.0.0	standby
msg	[message text]	msg "Hello world!"	V1.0	PV1.0.0	none
notify	[0/1] [id] ... [id] ... [0/1] [id] ... [id]	notify 1 status prog 0 ir	V1.0	PV1.0.0	wakeup
	format [0/1/2/3] {[string]} {[string]}	notify format 1 "\r\n!>"	V1.1	PV1.0.0	
power	[off/tv/audio/radio/photo]	power tv	V1.0	PV1.0.0	wakeup
prog	[program number]	prog 1001	V1.0	PV1.0.0	none
	prog number -22 = MusicBox		V2.0	PV1.0.0	
	prog number -13 = HDMI3		V2.0	PV1.0.0	
proglis	{[tv/radio]} {program number}	proglis tv	V1.0	PV1.0.0	wakeup

Id	Parameters	Example	RS232 version	SL1xx version	standby type
	[avtv/avradio] [fav] [favourite list number]				
	[tv/radio] paging [page size] page [first/last/next/+/prev/-] page [page number] page prog [program number] [favtv/favradio] [favourite list number]	proglis tv paging 10 proglis page + proglis page 10 proglis prog 58 proglis favradio 0	V2.0	PV1.0.0	none
range	[name]	range volume	V1.0	PV1.0.0	none
status		status	V1.0	PV1.0.0	standby
time	[?] [yyyy-mm-dd] [hh:mm:ss] {[local time offset]}	time 2007-05-31 10:30:00 -1	V1.0	PV1.0.0	none
timer	[add] [hdr / av1] [station number] [duration] add [hdr / av1] [station number] [date] [start time] [stop time]	timer add hdr 1 90 timer add hdr 2 2008-03-28 20:15 22:00	V1.0	PV1.0.0	wakeup
version		version	V1.0	PV1.0.0	standby

To request a parameter value a question mark has to be inserted instead of the value. The question mark is not necessary (but valid) if the command has no parameter.

Examples:

data volume ? bass0 ? treble0 ?

data volume ?

status ?

status

Invalid:

data volume bass0 treble0

8.2 Responses

Id	Parameters	Example
data	[name] [value] {[name] [value] ...}	data volume 17
help	[list of all supported commands]	help config data ... version
ident	[type] [version]	ident L2700 V1.0
ir	[system] [subsystem] [command] [status]	ir 0 0 1 '02
keys	ir [0/1] local [0/1]	keys ir 0 local 0
prog	[program number]	prog 1001
proglis	[tv/radio] [program number] [program name] CR/LF [program number] [program name] CR/LF ... [program number] [program name] proglis fav [favourite list number] 1 [program name] CR/LF 2 [program name] CR/LF ... [n] [program name]	proglis 1 "ARD" 2 "ZDF" 3 "SAT.1" 4 "DAS VIERTE" ... Hint: The station name is encoded in UTF-8 format.
	proglis [tv/radio] paging count [number of programs] pages [number of pages] proglis [tv/radio] page [page number] [program number]	proglis tv paging count 1000 pages 100 proglis tv page 1 1 "ARD"

Id	Parameters	Example
	[program name] CR/LF [program number] [program name] CR/LF ... [program number] [program name] proglis [favtv/favradio] [favourite list number] 1 [program name] CR/LF 2 [program name] CR/LF ... [n] [program name]	2 "ZDF" 3 "SAT.1" 4 "DAS VIERTE" 5 "RTL Television" ...
range	[name] [lower limit] [upper limit]	range bass -12 12
status	[main] [sub] [pip] [record]	status tv ttx pipoff recoeff
time	[yyyy-mm-dd] [hh:mm:ss] [local time offset]	time 2007-05-31 10:30:00 -1
version	[protocol version]	version 1.2.0

8.3 Notifications

Id	Parameters	Example
ir	[system] [subsystem] [command] [status]	ir 0 0 1 3
prog	[program number]	prog 123
status	[main] [sub] [pip] [record].	status tv ttx pipoff recoeff
data	[parameter name] [parameter value]	data volume 20
proglis	changed [tv/radio]	proglis changed tv

8.4 Message Details

8.4.1 data

This message may be used to query or to set a selection of internal data, mainly user settings. A parameter value can be modified by a certain value or by incrementing or decrementing.

Format:

data [parameter name] [?/+/-/[value]] {[parameter name] [?/+/-/[value]]...}

Examples:

Query a value: data volume ?
Set a value: data volume 25
Increment a value: data volume +
Decrement a value: data volume -

Parameter name	Contents	Value Range
volume	volume value	0..99 (maximum) Hint: The user setting "maximum volume" is considered!
maxvolume	user defined upper limit for volume	10 .. 99
mute	mute on/off Increment/decrement with +/- is not supported.	0/1
bass0	bass value for custom music sound	-12 .. +12
bass1	bass value for custom movie sound	-12 .. +12

treble0	treble value for custom music sound	-12 .. +12
treble1	treble value for custom movie sound	-12 .. +12
brightness	brightness value for current video mode	0 .. 20
contrast	contrast value for current video mode	0 .. 20
color	color value for current video mode	0 ...20
sharpness	sharpness value for current video mode	1 .. 5
favourite	current active favourite L2700 supports up to 6 favourite lists, referenced by the value 0 to 5. For the standard program list the value -1 is used. Increment/decrement with +/- is not supported.	-1 .. 5

The permitted range of values may be queried by the corresponding `range` message.

Data values can also be activated for notification. Whenever a value of any parameter is modified a data notification for the modified parameter is sent.

8.4.2 ident

The parameter list is a sequence of readable text strings. The first string identifies the TV type, the second one the software version.

8.4.3 ir

This command allows to emulate infrared remote control commands when used as a command, or to check the infrared receiver and decoder by monitoring notifications of received infrared commands.

Via the status byte the ir receiver can be disabled. That means all commands received from the remote control are ignored for TV control, but are forwarded as ir notification (if enabled). Only the ir commands received from the control interface are handled by the TV set. So the external control device connected to the control interface decides which ir commands are relevant. Additionally the local keys are disabled. The events are not forwarded as ir notifications!
Hint: After power off/on via main switch the keys are enabled again. To disable key permanently use the command `config`.

parameter: [system] [subsystem] [command] [status]

For the relevant system, subsystem and command codes see Appendix 1.

status:

bit 0: first command in sequence

bit 1: auto-repeat command

bit 2: delayed command; it defines once that a key has been pressed for more than 600 ms.

bit 3: key released

bit 4: IR receiver disabled

bit 4 – 7: don't care (have to be zero)

To simulate a single IR command bit 0 and bit 1 has to set.

Example:

“TV on”

```
ir 0 0 22 3
```

To disable/enable the IR receiver an ir command has to be sent with bit4 = 1/0. To avoid any action by the ir command use the following ones:

disable the ir receiver: `ir 0 0 0 16`

enable the ir receiver: `ir 0 0 0 0`

Simulating a complete key event

Some OSD elements or device controller evaluates the ir status in detail for different function. Examples:

- When the play key is pressed only short the HDR playback is started, if it pressed longer a bookmark is set.
- When scrolling in the index list (opened via the INFO key) with cursor up/down the corresponding info text at the top of screen is only updated with the “key released” event.

For this the internal ir status is updated every 110 ms as long as IR signals are received from the remote control. As the processing of the RS232 command can't be guaranteed in the same way as the remote control the update of the status has to be handled by the external application.

The status sequence dependent from the time the key is pressed is described below:

Time	Status bits				Status value
	first	auto-repeat	delayed	key released	
0 ms	1	1			3
110 ms					0
220 ms					0
330 ms					0
440 ms					0
550 ms		1			2
660 ms		1	1		6
770 ms		1			2
...					
n * 110 ms		1			2
key is released				1	8

A complete key press exists at least of two `ir` commands, because the release of the key also has to be sent. All other status settings only have to be sent, when the key has been pressed for the stated time.

Hint: If the ir receiver is disabled (“`ir 0 0 0 16`”) and the notification for the `ir` command is activated (“`notify 1 ir`”), the exact sequence is sent via RS232 when a key of the remote control is pressed.

8.4.4 notify

Notifications are not sent automatically, but only on request. This message allows to enable or disable specific notifications. Only those messages listed in chapter 8.3 can be enabled.

Notifications will be sent once after enabling a notification, and whenever the information has changed.

Format:

```
notify [0/1] {[id] ... [0/1] [id]...}
```

To activate/deactivate all notifications `notify 1 / notify 0` can be used.

Since the RS232 version V1.1 the leading and/or trailing strings for notifications can be defined arbitrarily. This might be necessary to distinguish notifications from responses when using command sequences.

```
notify format [format: 0/1/2/3] {[string1] [string2]}
```

format:

- 0: Standard format. Notifications are terminated by CR/LF and the prompt character.
- 1: Notifications are sent with a trailing string defined by string1.
- 2: Notifications are sent with a leading string defined by string1.
- 3: Notifications are sent with a leading string defined by string1 and a trailing string defined by string2.

The strings are limited by a maximum of 5 characters. The `format` parameter can't be used in combination with the parameter to enable/disable notifications.

Examples:

```
notify format 0
```

```
notify format 1 "\r\n!>"
```

```
notify format 2 "!"
```

```
notify format 3 "@" "\r\n@>"
```

8.4.5 power

This command is used to switch the tv set to certain operation states:

parameter	description
init	If the TV is in standby after first power on (via main switch) the program list is not initialized (amongst others), so it is not available with the command <code>proglis</code> . In this case the command "power init" can be used to start the TV without picture and sound. When the initialization is done the TV returns to standby automatically. This can be evaluated with the command <code>status</code> . If the TV is already initialized the command "power init" has no effect. The current state of the TV set is not changed in this case.
off	Switch tv to standby. If a record is active the TV will change to "active" mode only.
tv	Switch tv on with the last active tv channel.
audio	Switch tv to radio mode with inactive screen (= audio mode) reproducing sound of last radio channel.
radio	Switch tv to radio mode with active screen (=radio mode) reproducing sound of last radio channel.

For evaluating the current power mode use command "status".

8.4.6 prog

This message may be used to query the current program number (if used without parameters), or to request a switchover to a specific program number. The message will be confirmed immediately by a single `RCI_PROMPT_CHAR`, and a single `prog` notification containing the current program number will be sent when the program change has either been completed or rejected. For to find out whether the request has been accepted, the returned program number has to be compared with the requested program number.

Negative program numbers are reserved for AV channels:

number	AV channel	additional information
--------	------------	------------------------

number	AV channel	additional information
-1	AV1	
-2	AV2	Not available in SL1xx.
-3	AV3	Not available in SL1xx.
-4	AVS	
-5	VGA (=PC-In)	Not available in radio/audio mode.
-6	HDMI1	
-7	HDMI2	
-8	COMP-IN-1	Not available in SL1xx.
-9	COMP-IN-2	Not available in SL1xx.
-10	Audio-In-1	For radio/audio mode only.
-11	Audio-In-2	Not available in SL1xx.
-12	Audio-Digital-In	For radio/audio mode only.
-13	HDMI3	If available (SL121).
-14 to -19	reserved	
-20	HDR playback	This number is readable only. It can't be used to start a HDR playback by sending "prog -20"!!!
-21	Photoviewer	This number is readable only. It can't be used to start the Photoviewer.
-22	Musicbox	This number is readable only. It can't be used to start the MusicBox.
-23	Videoplayer	This number is readable only. It can't be used to start the VideoPlayer.
-24	Medioplayer	This number is readable only. It can't be used to start the Medioplayer mode.

8.4.7 progdata

T. b. d.

8.4.8 proglst

This message is used to get the current tv program list, radio program list or the favourite lists.

Hint: The programs are stored depending on the source in blocks. So gaps can exist.

Example:

Analogue channels: 1 – 32
 DVB-C channels: 101 – 456
 DVB-T channels : 501 – 511
 DVB-S (Astra) : 600 – 1551

Favourite lists have up to 99 programs without any gaps.

The program name is UTF-8 encoded. DVB channels also can contain some special character ([0xC2 0x86] / [0xC2 0x87]) to encode a short name.

[0xC2 0x86]: Begin of part of short name

[0xC2 0x86]: End of part of short name

Example:

[0xC2 0x86]S[0xC2 0x87]uper[0xC2 0x86] RTL[0xC2 0x87]

Long name: "Super RTL"

Short name: "S RTL"

To get the long name removed these special characters.

8.4.8.1 TV/Radio program list

The program number and the program name are replied. The first parameter of the response always contains the program type information (tv or radio).

The available Audio/Video inputs (e.g. scart AV1) for tv or radio mode are only returned with parameter avtv respectively avradio.

Format for command:

```
proglis [tv/radio/avtv/avradio] {[program number]}
```

Format for response:

```
proglis [tv/radio/avtv/avradio] [[program number] "[program name]"  
CR/LF]]...
```

Command	Response	Description
proglis	proglis [tv/radio] [prog. num.] "[prog. name]" CR/LF [prog. num.] "[prog. name]" CR/LF ... [prog. num.] "[prog. name]" CR/LF	Returns the complete program list of the current mode (tv or radio) is returned. The list is finished with the prompt character RCI_PROMPT_CHAR.
proglis [prog. number]	proglis [tv/radio] [prog. num.] "[prog. name]" proglis [tv/radio] [next valid prog. num] proglis [tv/radio]	Returns the program number and the program name of the given program number dependent on the current mode (tv or radio). If the given program number is invalid the response contains the next valid program number. If there is no next valid program number the response does not contain any data. The av inputs are not supported in this case.
proglis tv proglis radio proglis avtv proglis avradio	proglis [tv/radio] [prog. num.] "[prog. name]" CR/LF [prog. num.] "[prog. name]" CR/LF ... [prog. num.] "[prog. name]" CR/LF >	Returns the complete program list of the given mode independent from the current active mode of the TV set. avtv: returns the available av inputs for tv mode avradio: returns the available av inputs for radio mode
proglis tv [prog. number] proglis radio [prog. number]	proglis [tv/radio] [prog. num.] "[prog. name]" proglis [tv/radio] [next valid prog. num] proglis [tv/radio]	Returns the program number and the program name of the given mode and program number dependent on the current mode (tv or radio). If the given program number is invalid the response contains the next valid program number. If there is no next valid program number the response does not contain any data. The av inputs are not supported in this case.

8.4.8.2 Get tv/radio list page by page

The format of the chapter before is used to get the complete program list or only a single channel.

Since the RS232 version V2.0 (available since TV version V6.0) it is possible to request single pages of the tv or radio list. The page size can be defined arbitrarily (2 – 255 items per page). E.g. this could be necessary for external devices without enough memory to store up to 5000 channels.

For paging the page size has to be defined first:

```
proglis [tv/radio] paging [items per page]
```

The TV will response by sending the total number of programs and the total number of pages:

```
proglis [tv/radio]paging count [number of channels] pages [number of pages]
```

After this it is possible to navigate within the program list:

```
proglis page [first/last/prev(+)/next(-)/<number of page>/
             prog <program number>]
```

Get first page: `proglis page first`

Get last page: `proglis page last`

Get next page: `proglis page next` or `proglis page +`

Get previous page: `proglis page prev` or `proglis page -`

Get certain page by number: `proglis page [page number]`

The page number starts with 1 (first page) and ends with *number of pages* (last page).

Get page containing a certain program number:

```
proglis page prog <program number>
```

The program number needs not to be the first item of the page. It depends on the absolute position of the program number within the complete list.

The TV will send the requested page in the following format:

```
proglis [tv/radio/av/avradiio] page [page number] [[program number]
"[program name]" CR/LF]]...
```

Paging is possible only for one list type (tv or radio) at one time. If the type is changed the paging has to be initialized with defining the page size again.

Paging is not possible for favourites (see next chapter) and the AV inputs (avtv / avradio).

8.4.8.3 Favourite lists

Format for command:

```
proglis fav [favourite list number]
```

Since TV version V6.0:

```
proglis [fav/favtv/favradiio] [favourite list number]
```

In L2700 6 favourite lists are supported. The list number starts with 0.

Format for response:

```
proglis fav [list number] "[list name]" [[program number] "[program name]"
CR/LF]]...
```

The program number of the response is the corresponding program number of the standard program list and not the sequential number within the favourite list. This program number can be used to change to a certain program independent from the active favourite list.

8.4.8.4 proglis notification

If the notification for proglis is activated the modification of the tv or the radio list is notified. This is done whenever a channel or a channel block is moved or deleted.

Format:

```
proglis changed [tv/radio]
```

8.4.9 range

This message may be used to query the permitted range of values for data. For a list of supported data, see the `data` message above.

8.4.10 status

This message may be used to find out some major status information. The information will be transmitted in textual form for to guarantee long-term consistency.

It is not possible to change the status by this command, proper `ir` or `power` messages have to be used instead.

The status can be request by sending `status` or `status ?`.

The response can contain 4 parameters:

```
status <power mode> <sub mode> <pip mode> <record mode>
```

Currently specified is this information:

Position	Contents	Value	Description	
1	power mode	standby	standby	
		active	active standby for EPG nightupdate or recording.	
		audio	audio mode = Radio digital mode with screen switched off	
		radio	Radio digital mode	
		tv	tv mode	
2	sub modes for power mode = "standby"	off	TV is in standby mode. The standby processor handles the RS232 commands, the main processor is switched off completely. Only standby commands can be handled.	
		wakeup	The main processor has been started by the standby processor and boots up to handle a command. This state takes some seconds (e.g. chassis SL121 about 7 seconds). The command will be handled as soon as the main controller has been booted.	
		on	The main processor is running but the TV is still in standby mode (without picture and sound). All commands can be handled now.	
	sub modes for power mode = "active"	off	TV is in active standby (with orange front LED), e.g. for recording (see parameter "record mode").	
		epg	EPG nightupdate is active	
	sub modes for power mode = "tv"	off	no sub mode is active	
		tx	Teletext is active	
		photo	Photo Viewer is active	
		music	Music Box is active	
	sub modes for power mode = "radio" and "audio"	video	Video Player is active	
		off	There are no further submodes.	
	3	pip mode	pipoff	PiP off

Position	Contents	Value	Description
		pipon	PiP on
		pipsplit	split screen
4	record mode	recoff	no record active
		rechdr	record to DR+ active
		recav1	record to AV1 active

8.4.11 time

This message is used to get or set the current system date, time and local time offset (lto). The value of the local time offset is independent from daylight saving time.

get time:

```
time or time ?
```

```
response: time [yyyy-mm-dd] [hh:mm:ss] [lto]
```

set time:

```
time [yyyy-mm-dd] [hh:mm:ss] {[lto]}
```

The lto parameter is optional. If the values are valid the command is confirmed with the prompt. If the number of the parameter or the values are invalid a ? or time "[error message]" is returned.

error messages:

```
time "date invalid"
time "time invalid"
time "lto invalid"
```

8.4.12 timer

With this message it is possible to program the internal TV record timers.

For a direct recording starting at the current time running for a certain time use the following format:

```
timer add [destination] [station number] [duration in minutes]
```

A recording for an arbitrary time can be defined as follows:

```
timer add [destination] [station number] [date: yyyy-mm-dd] [start time:
hh:mm] [stop time: hh:mm]
```

destination:

- hdr: start record to a HDR
- av1: start record to a connected device (VCR or DVD recorder) at AV1

Examples:

Start a recording of station number 2 on HDR for 1.5 hours:

```
timer add hdr 2 90
```

Start a recording of station number 2 on HDR on the 28th of June 2010 from 20:15 to 22:00:

```
timer add hdr 2 2010-06-28 20:15 22:00
```

If the record is possible the command is confirmed with the prompt. Otherwise a ? or timer "[error message]" is returned.

Error messages:

Error message	Description
"invalid parameter"	The parameters are invalid. E.g. not enough parameter.
"timerlist full"	The maximum number of timer are already in use.
"device not present"	The target device is not installed or connected. E.g. HDR is not installed or there is no VCR or DVD-Recorder configured for AV1 socket
"no system time"	Time/Date hasn't been detected or inserted by user.
"conflict"	There is a conflict with another timer record.
"invalid station number"	The station number which has to be recorded doesn't exist.
"invalid duration"	The duration exceed the maximum value (23 hours 59 min = 719 min)

8.4.13 config

This message may be used to set different system configurations.

The following parameters are supported:

Parameter	Value	Description
ir	0: disable IR commands 1: enable IR commands	The modification of this parameter is only possible if the hotel mode of the service menu is activated. "config ir 0" disables the IR commands received from the remote control. The IR commands are forward as ir notification to the RS232 interface independent whether notification of ir is enabled or not. Attention: This setting is stored in the EEPROM, so it is still active after power off/on via mains switch.
localkeys	0: disable local keys at the front panel 1: enable local keys	The modification of this parameter is only possible if the hotel mode of the service menu is activated. "config localkeys 0" disables the local keys at the front panel. The key events are forward as ir notification to the RS232 interface independent whether notification of ir is enabled or not. For the corresponding ir command see Appendix 1, keys for subsystem 27. Attention: This setting is stored in the EEPROM, so it is still active after power off/on via mains switch.

8.4.14 help

This command shows a list of all supported command.

In standby it returns the commands which are supported in standby only.

8.4.15 version

This command returns the version string of the control interface protocol defined in this document.

For chassis SL1xx the version starts with V3.0.0 to point out the difference to the versions of chassis L27xx (2.x.y or lower).

8.4.16 msg

This command is used to display any text in an OSD message window. The message can be closed by sending this command without parameter.

Example:

```
msg "First line.\nSecond line.\tThis is a backslash in double quotes: \" \\ \"\!"
```

8.4.17 dmenu

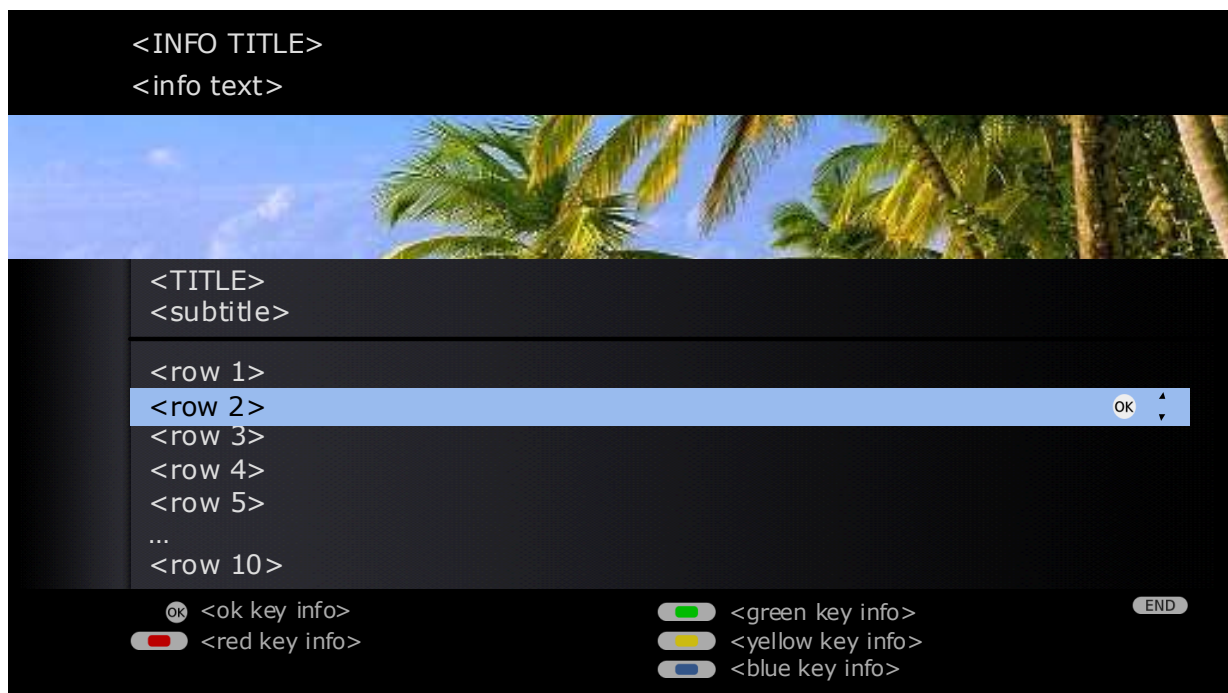
The command is used for an interactive communication via dynamic menus (dmenu) of the TV OSD. So it is possible to ask the user for certain actions to execute, to do configurations for an external system (e.g. home automation) or to display its state or any other information.

Via dmenu different types of menus can be defined and controlled. Additionally it is possible to add menu items to the existing menu of the TV set, so called entry points. Relevant events (mainly key events) necessary for the control of the dynamic menus are reported by notifications.

The menus are subjected to the designing rules of the Loewe OSD. It defines that a dialog is closed by the END key. So the behaviour of this key is fixed. It can't be configured via the dmenu command and will be available whenever a dynamic menu is opened. The displayed texts and the handling of some keys can be defined arbitrarily.

8.4.17.1.1 Menu type "list"

This list contains up to 10 selectable rows. The rows can be divided into three columns (e.g. for a key-value presentation). Optionally an info (title and text) can be added. The four colour keys and the ok key can be logged for event handling. The info text for each key is optional.



8.4.17.2 Menu type "message"

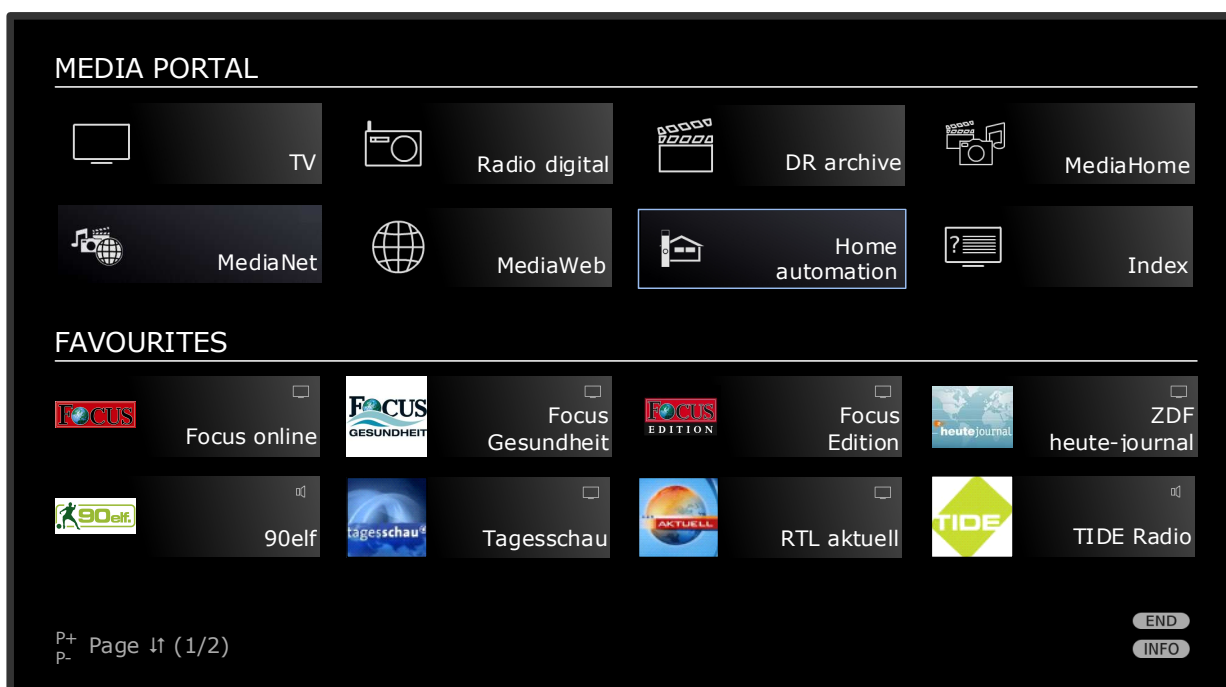
The message is used to display an info text requiring feedback from the user via a defined key event. The four colour keys and the OK key can be logged for event handling. The info text for each key is optional. Additionally the PIP can be opened with a given channel resp. AV input (e.g. for door camera). For a simple message use the msg command (see chapter 8.4.16).



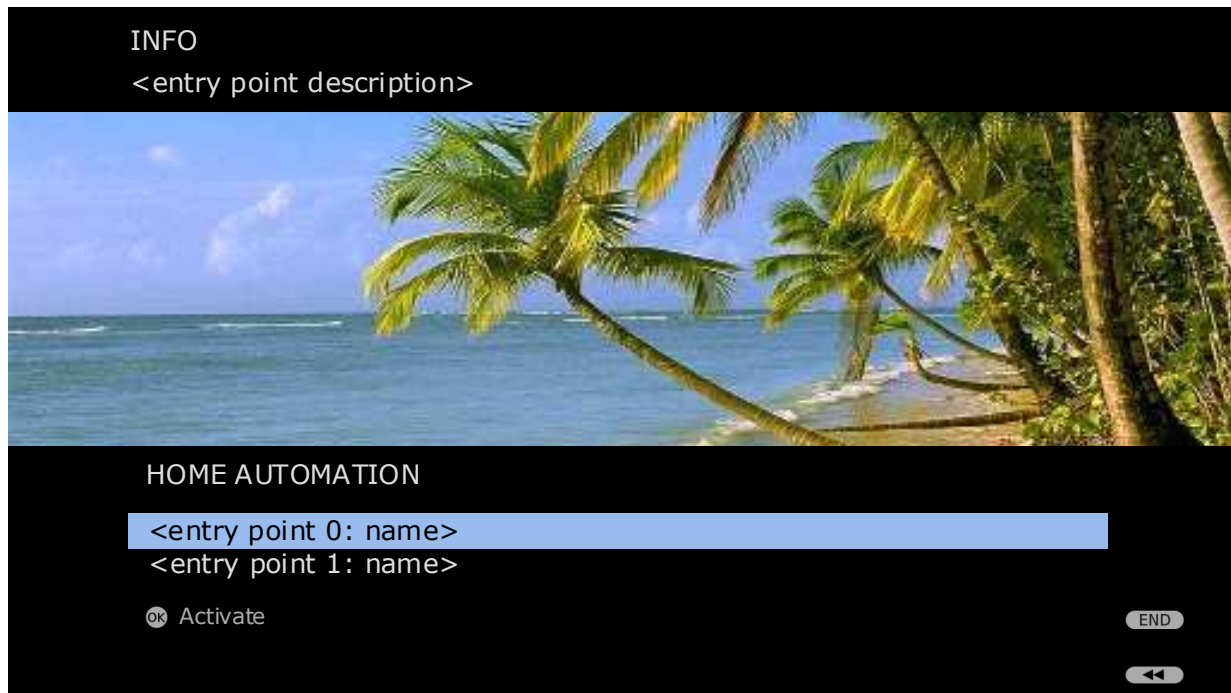
8.4.17.3 Entry points

Up to two menu items can be defined which are accessible via the Media Portal of the TV set. When the user selects an item and ok is pressed the corresponding event is reported. This can be used as the initial event for opening a dynamic menu. These menu items are so called “entry points”. A name and an optional description can be defined.

If at least one entry point has been defined the Media Portal contains the additional item “Home automation”. The Media Portal can be opened via the ASSIST key.



After selecting “Home automation” a selection menu is opened where the entry point(s) can be activated by pressing the OK key.



Hint: The command `dmenu` is not supported in standby at all. So entry points have to be defined when the TV is active. As the information about entry points is deleted when the TV is switched off to standby, the entry points have to be redefined whenever the TV gets active again.

It is recommended to activate notification for the status command to get the information whenever the TV gets active.

8.4.17.4 Command format

Format of `dmenu`;

```
dmenu [subcommand] [parameter list]
```

```
subcommand = [define / undefine / content / status / event / entry]
```

The parameter list is a list of parameter names and its value or values. The list is only terminated by the maximum length of a RS232 command `RCI_LINE_LENGTH_LIMIT`. So it is possible to send several parameters with one `dmenu` command, but only for one subcommand.

The parameter (and its values) depends on the subcommand and the selected menu type.

For some subcommands and parameter shortcuts exists. So it is possible to send more parameter using shortcuts with one RS232 command to get a better performance for opening menus. The corresponding shortcut of a subcommand/parameter is marked by the underlined characters (e.g. d for `define`).

The following chapters give an overview of the subcommands, parameters and parameter values.

The sequence of commands to define and open/close a menu is described in chapter 8.4.17.5.

8.4.17.4.1 Subcommand define

Defines the menu type and the layout parameter.

define parameter	values	description
type [menu type]	menu type: list A list is a menu containing a selectable list of up to 10 rows. A row can be separated into 3 columns for a key/value relationship. message Used to display a message and requesting a confirmation from the user by pressing a key of the remote control. Additionally a PiP can be opened with a given channel or AV input.	It is the very first parameter which has to be set for initialization. Whenever it is set all menu type specific parameter values are set to default and previously defined content strings are deleted.
define parameter for menu type list:		
rows [number of rows]	number of rows: 1 – 10 default: 5	Defines the number of rows displayed for the selection.
cols [number of columns]	number of columns: 1 – 3: default: 1	Defines the number of columns and so the number of strings which have to be defined for an item.
items [number of items]	number if items: 1 – [number of rows] default: 5	Defines the number of items the user can select with the cursor. It can be lower than the number of rows. In this case the remaining rows are empty and can't be selected.
info [0/1]	0: info has to be displayed 1: info has not to be displayed default: 0	Define whether an info additionally has to be displayed at the top of the TV screen.
subtitle [0/1]	0: subtitle has to be displayed 1: subtitle has not to be displayed default: 0	Defines whether a subtitle at the top of the list has to be displayed.
key [key name] [0/1]	key name: ok / red / green / yellow / blue 0: key is not requested for notification 1: key is requested for notification	Defines whether one of the listed keys is requested for notification when it is pressed by the user or not. A key has to be requested to display a corresponding hint in the hint area of the menu. The ok key is exclusively used for activating a selection. So it is displayed additionally as hint in the selected row when it is requested.
selected [item index]	item index: 0 – ([items] – 1)	Defines the selected item of the list. The first item has index 0.
onSelectFast [0/1]	0: fast event handling for event onSelect is off 1: fast event handling for event onSelect is on default: 0	If onSelectFast is off the event onSelect is only notified when the user has released the cursor up/down for selecting an item. So not every selection is reported when the user scrolls within the list by keeping the cursor key pressed. if onSelectFast is on every selections is notified.
define parameter for menu type message:		

define parameter	values	description
info [0/1]	see above	
key [key name] [0/1]	key name: ok	see above.
pip [0/1]	0: no PIP 1: an inactive PIP is opened additionally	For the message menu additionally a PIP can be opened. The program is defined by the content parameter pipProg (see chapter below). If no program is defined the previous PIP program is displayed. Hint: The position of the PIP allows up to three lines in the info text at the top of the screen.

Examples:

```
dmenu define type list rows 6 cols 2 items 4 info 1 subtitle 1
dmenu define onSelectFast 1
```

8.4.17.4.2 Subcommand content

Content parameter contains mainly the text strings displayed in the menus:

content parameter	value	description
content parameter for menu type list		
title [<title string>]		Title displayed at the top of the list.
subtitle [<subtitle string>]		Subtitle displayed at the top of the list below the title. Precondition: The define parameter subtitle has to be set.
info [<title string>] [<text string>]		Defines the info at the top of the TV screen. The info title can be an empty string (""). The text string must not be empty.
item [index] [<string for col0> {[<string for col1>}] {[<string for col2>}]	index: row number starting with 0	Defines the strings displayed for an item in the list. The number of string parameter depends on the value of the define parameter cols. The strings can be empty. Shortcut: i
key_info [key name] [<info string>]	key name: ok / red / green / yellow / blue	Defines the info string displayed besides the key icon in the hint area of the menu. Precondition: the key has to be requested by the define parameter key. Shortcut: ki
content parameter for menu type message		
info [<title string>] [<text string>]	see above	
key_info [key name] [<info string>]	key name: ok	see above.

content parameter	value	description
pipProg [program number]	For AV channels use the negative program ids defined in chapter 8.4.6.	Precondition: To open the PIP the define parameter pip has to be set to 1.

Examples:

```
dmenu content title "Home automation" subtitle "Main menu"
```

```
dmenu content item 0 "Function 1" "Status A" item 1 "Function 2" "Status C"
```

```
dmenu content keyinfo ok "execute" keyinfo blue "F1"
```

using shortcuts:

```
dmenu c i 0 "Function 1" "" i 1 "Function 2" "Status A"
```

```
dmenu c ki ok "execute" ki blue "F1"
```

8.4.17.4.3 Subcommand status

It is used to open, close and update a menu.

Syntax:

```
dmenu status [open / close / update]
```

Example:

```
dmenu status open
```

```
dmenu status close
```

8.4.17.4.4 Subcommand event

This subcommand is sent by the TV set whenever a relevant event occurs.

Syntax:

```
dmenu event [event type] {[event parameter]}
```

Examples:

```
dmenu event onOpen
```

```
dmenu event onKey ok
```

```
dmenu event onSelect 1
```

event type	parameter	Description
onOpen	---	onOpen is sent when a menu has been opened successfully.
onClose	---	onClose is sent when a menu has been closed.
onKey [key name]	key name: ok / red / green / yellow / blue	onKey is sent when a requested key has been pressed.
onSelect [index]	index: index of the selected item in the list menu	The event is only sent when the list menu is active and the user has moved the cursor of the selected list item. See description of the parameter onSelectFast of chapter 8.4.17.4.1.

event type	parameter	Description
onEntry [entry point id]	entry point id: 0: entry point #0 of the assist menu 1: entry point #1 of the assist menu	If an entry point has been defined with subcommand <code>entry</code> the event is sent when the user has selected the menu item by pressing ok.

8.4.17.4.5 Subcommand `entry`

This subcommand is used to add up to 2 menu items to the assist menu of the TV set. When the user select an item and ok is pressed the corresponding `onEntry` event is sent. This can be used as the initial event for opening a dynamic menu.

Syntax:

```
dmenu entry [entry point id] [status] {[<item name>]} {[<item description>]}
```

entry point id: 0 or 1

status: 0: disable entry point
 1: enable entry point

When the entry point is enabled an item name has to be added, an item description is optional.

Examples:

```
dmenu entry 0 1 "Home automation" "Configure your home automation system"
dmenu entry 1 1 "Audio server" "Configure your audio server"
dmenu entry 0 0
```

8.4.17.4.6 Subcommand `undefine`

Use this subcommand to reset all menu definitions and to delete all text strings. After this a dynamic menu has to be defined completely new. This subcommand should be sent whenever a dynamic menu has been closed and will not be used any longer.

8.4.17.5 Handling of Dynamic menus

The "construction" of a dynamic menu is executed in three steps:

1. define the menu type and its layout parameter

Commands: `dmenu define type ...`
 `dmenu define ...`

2. define its content mainly the text strings which have to be displayed

Command: `dmenu content ...`

3. open the menu

Command: `dmenu status open`

If a large menu (containing many information) has to be opened the following procedure possibly makes sense for a better performance to provide a prompt reaction after a user interaction.

1. define the menu type and its layout parameter

Commands: `dmenu define type ...`
 `dmenu define ...`

2. define some basic content strings (e.g. title of the menu)

Command: `dmenu content ...`

3. open the menu

Command: `dmenu status open`

4. define all other content strings

Command: `dmenu content ...`

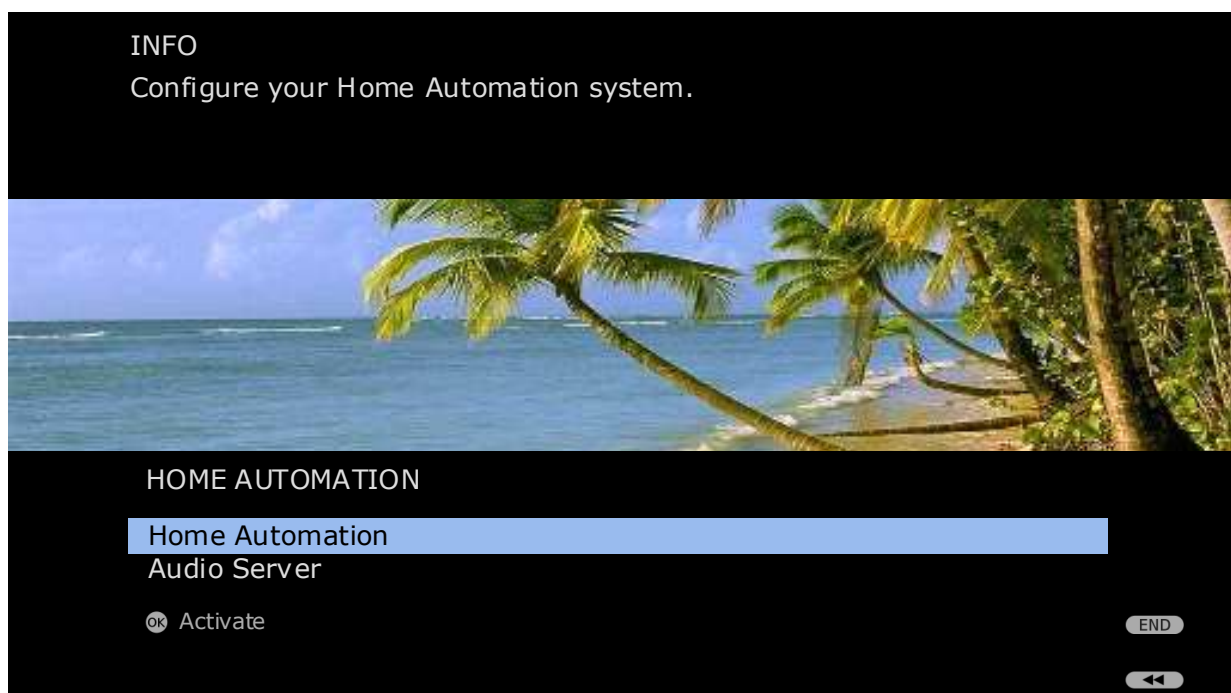
5. update the menu

Command: `dmenu status update`

8.4.17.6 Examples

8.4.17.6.1 Entry Points

```
dmenu entry 0 1 "Home Automation" "Configure your Home Automation system"
dmenu entry 1 1 "Audio Server" "Configure your Audio Server"
```



When the user selects one of these items and presses ok an event notification is sent by the TV:

```
dmenu event onEntry 0
```

or

```
dmenu event onEntry 1
```

8.4.17.6.2 Message

Open a message when a visitor arrives and show the door camera in PiP. The door camera is connected to scart socket AV1:

```

dmenu d type message info 1 key ok 1 pip 1
dmenu c info "Hint" "You have a visitor! Press OK to open the door."
dmenu c keyinfo ok "Open door" pipProg -1
dmenu status open

```



When the user presses OK an onKey event is sent by the TV set:

```
dmenu event onKey ok
```

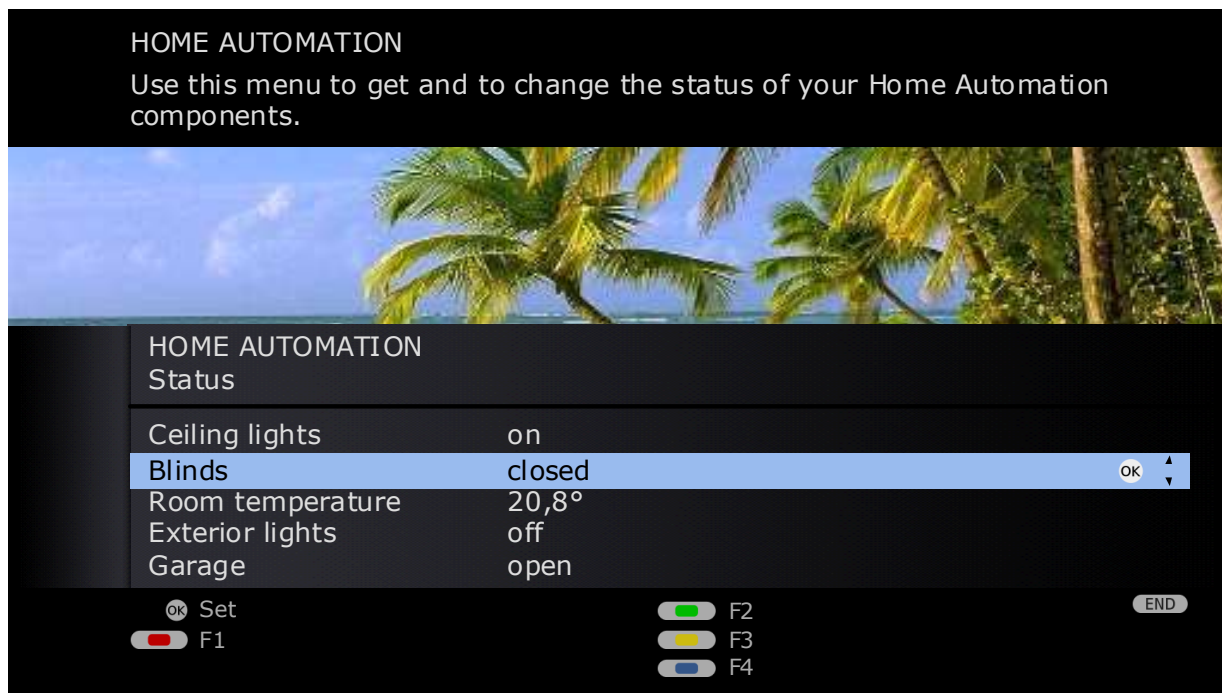
8.4.17.6.3 List

The following example shows a list with 5 rows and 2 columns containing the status of different components controlled by a home automation system. The first column contains the component name the second one the component status. The status can be changed by selecting the corresponding row and pressing OK (possibly by opening another menu). Additionally the colour keys are used to execute the functions called F1 to F4. A corresponding info text is available.

```

dmenu d type list subtitle 1 info 1 rows 5 cols 2 items 5 selected 1
dmenu d key ok 1 key blue 1 key green 1 key red 1 key yellow 1
dmenu c title "Home Automation" subtitle "Status"
dmenu c keyinfo ok "Set" keyinfo blue "F4" keyinfo green "F2" keyinfo red "F1"
dmenu c keyinfo yellow "F3"
dmenu c info "Home Automation" "Use this menu to get and to change the status of your
Home Automation components."
dmenu s open
dmenu c i 0 "Ceiling lights" "on" i 1 "Blinds" "closed" i 2 "Room temperature" "20,8°"
dmenu c i 3 "Exterior lights" "off" i 4 "Garage" "open"
dmenu s update

```



When the menu is opened the TV set sends an onOpen event command:

```
dmenu event onSelect 2
```

Whenever the user selects a row via the cursor up/down key and releases the key of the remote control an onSelect event is sent by the TV set:

```
dmenu event onSelect 2
```

When the user presses OK to change the status an onKey event is reported:

```
dmenu event onKey ok
```

When the menu is closed via the END key an onClose event occurs:

```
dmenu event onClose
```

9 Configuration Data and Limit Values

Symbol	Value
RCI_LINE_LENGTH_LIMIT	128
RCI_PROMPT_CHAR	'>'

10 AMX

For AMX home automation the poll message “AMX” is supported. For this in the service menu of the TV set the following parameter has to be set:

Open the “Service menu” on the TV set. Change to the menu item “RS232”. Set the item “AMX” to “on”. Additionally the port settings required by AMX are adjusted: 9600 baud, 8 bits, no parity, 1 stop bit.

The command is available also in standby as command type “standby” (see chapter 8.1).

Appendix 1: IR codes

The following table contains relevant control codes for control system 0 (=RC5):

subsystem	control code	description
0	0	Key 0
0	1	Key 1
0	2	Key 2
0	3	Key 3
0	4	Key 4
0	5	Key 5
0	6	Key 6
0	7	Key 7
0	8	Key 8
0	9	Key 9
0	10	PIP on/off
0	11	menu
0	12	TV on/off
0	13	mute
0	14	normal (S)
0	15	EPG (H/S)
0	16	cursor right
0	17	cursor left
0	20	Volume -
0	21	Volume +
0	22	TV on
0	23	Program -
0	24	Program +
0	25	TV off
0	26	green
0	27	red
0	28	VCR1
0	30	DVD
0	32	cursor up
0	33	cursor down
0	35	picture
0	36	SAT
0	37	STB
0	38	OK key
0	40	blue key
0	43	yellow key
0	49	Assist+ menu
0	53	Radio mode
0	60	teletext on/off
0	63	end key
0	64	Sound menu
0	72	TV mode on
0	73	Radio mode on
0	74	AV input menu
0	78	EPG (*)
0	79	info key
0	88	PIP (#)
0	90	Aspect ratio

subsystem	control code	description
0	91	timer dialog
0	92	DR+ archive
0	114	AV1 input
0	115	AV2 input
0	116	AV3 input (no function)
0	117	AVS input (front socket)
0	118	VGA input
0	119	DVI/HDMI-1 input
0	120	Component input
0	121	HDMI-2 input
0	122	HDMI-3 input
6	41	HDR: pause
6	50	HDR: rewind
6	52	HDR: fast forward
6	53	HDR: play
6	54	HDR: stop
6	55	HDR: record
27	15	local keys: key +
27	19	local keys: key radio
27	45	local keys: key -
27	46	local key: function key (M key or C)
27	163	RS232 menu