6 Remote Control Commands

In the following, all remote-control commands will be presented in detail with their parameters and the ranges of numerical values.

For an introduction to remote control and the status registers, refer to Chapter 5, "Remote Control Basics", on page 240.

6.1 Conventions used in SCPI command descriptions

The following conventions are used in the remote command descriptions:

- **Command usage**
  If not specified otherwise, commands can be used both for setting and for querying parameters.
  If a command can be used for setting or querying only, or if it initiates an event, the usage is stated explicitly.

- **Parameter usage**
  If not specified otherwise, a parameter can be used to set a value and it is the result of a query.
  Parameters required only for setting are indicated as **Setting parameters**.
  Parameters required only to refine a query are indicated as **Query parameters**.
  Parameters that are only returned as the result of a query are indicated as **Return values**.

- **Conformity**
  Commands that are taken from the SCPI standard are indicated as **SCPI confirmed**. All commands used by the R&S SMB follow the SCPI syntax rules.

- **Asynchronous commands**
  A command which does not automatically finish executing before the next command starts executing (overlapping command) is indicated as an **Asynchronous command**.

- **Reset values (*RST)**
  Default parameter values that are used directly after resetting the instrument (`*RST` command) are indicated as ***RST** values, if available.

- **Factory preset values**
  Default parameter values that are reset only by factory preset.

- **Default unit**
  The default unit is used for numeric values if no other unit is provided with the parameter.

- **Manual operation**
  If the result of a remote command can also be achieved in manual operation, a link to the description is inserted.

6.2 Common commands

Common commands are described in the IEEE 488.2 (IEC 625-2) standard. These commands have the same effect and are employed in the same way on different devices. The headers of these commands consist of "*" followed by three letters. Many common commands are related to the Status Reporting System.

Available common commands:

*CLS

Clear status

Sets the status byte (STB), the standard event register (ESR) and the `EVENt` part of the `QUEStionable` and the `OPERation` registers to zero. The command does not alter the mask and transition parts of the registers. It clears the output buffer.

Usage: Setting only

*ESE <Value>

Event status enable

Sets the event status enable register to the specified value. The query returns the contents of the event status enable register in decimal form.

Parameters:
<Value>   Range: 0 to 255

*ESR?

Event status read

Returns the contents of the event status register in decimal form and then sets the register to zero.

Return values:

<Contents>	Range: 0 to 255

Usage: Query only

---

*IDN?

Identification

Returns the instrument identification.

Return values:

<ID>	"Rohde&Schwarz,<device type>,<part number>/<serial number>,<firmware version>"

Example: `Rohde&Schwarz,SMB,1412.0000K02/000000,03.01.158`

Usage: Query only

Manual operation: See "Hardware Options / Software Options" on page 99

---

*IST?

Individual status query

Returns the contents of the IST flag in decimal form. The IST flag is the status bit which is sent during a parallel poll.

Return values:

<ISTflag>	0	1

Usage: Query only

---

*OPC

Operation complete

Sets bit 0 in the event status register when all preceding commands have been executed. This bit can be used to initiate a service request. The query writes a "1" into the output buffer when all preceding commands have been executed, which is useful for command synchronization.

---

*OPT?

Option identification query

Queries the options included in the instrument. For a list of all available options and their description, refer to the data sheet.

Return values:

<Options>	The query returns a list of options. The options are returned at fixed positions in a comma-separated string. A zero is returned for options that are not installed.

Usage: Query only

Manual operation: See "Hardware Options / Software Options" on page 99

---

*PRE <Value>

Parallel poll register enable

Sets parallel poll enable register to the indicated value. The query returns the contents of the parallel poll enable register in decimal form.

Parameters:

<Value>	Range:	0 to 255

---

*PSC <Action>

Power on status clear

Determines whether the contents of the ENABle registers are preserved or reset when the instrument is switched on. Thus a service request can be triggered when the instrument is switched on, if the status registers ESE and SRE are suitably configured. The query reads out the contents of the "power-on-status-clear" flag.

Parameters:

<Action>	0 | 1
	0
	The contents of the status registers are preserved.
	1
	Resets the status registers.

---

*RCL <Number>

Recall

Loads the instrument settings from an intermediate memory identified by the specified number. The instrument settings can be stored to this memory using the command *SAV with the associated number.

It also activates the instrument settings which are stored in a file and loaded using the MMEMory:LOAD <number>, <file_name.extension> command.

Manual operation: See "Recall Immediate x" on page 131

---

*RST

Reset

Sets the instrument to a defined default status. The default settings are indicated in the description of commands.

The command is equivalent to SYSTem:PRESet.

Usage:	Setting only
Manual operation:	See "Preset" on page 96

---

*SAV <Number>

Save

Stores the current instrument settings under the specified number in an intermediate memory. The settings can be recalled using the command *RCL with the associated number.

To transfer the stored instrument settings in a file, use the command :MMEMory: STORe:STATe.

Manual operation:	See "Save Immediate x" on page 129

---

*SRE <Contents>

Service request enable

Sets the service request enable register to the indicated value. This command determines under which conditions a service request is triggered.

Parameters:

<Contents>	Contents of the service request enable register in decimal form. Bit 6 (MSS mask bit) is always 0.
	Range: 0 to 255

---

*STB?

Status byte query

Reads the contents of the status byte in decimal form.

Usage:	Query only

---

*TRG

Trigger

Triggers all actions waiting for a trigger event. In particular, *TRG generates a manual trigger signal. This common command complements the commands of the TRIGger subsystem.

Usage:	Event
Manual operation:	See "Execute Single Trigger" on page 234

*TST?

Self-test query

Initiates self-tests of the instrument and returns an error code.

Return values:

<ErrorCode>   **integer > 0 (in decimal format)**
                           An error occurred.
                           (For details, see the Service Manual supplied with the instru-
                           ment).

                           0
                           No errors occurred.

Usage: Query only

*WAI

Wait to continue

Prevents servicing of the subsequent commands until all preceding commands have
been executed and all signals have settled (see also command synchronization and
*OPC).

Usage: Event

6.3 Preset Commands

The preset commands are not bundled in one subsystem. Therefore, they are listed
separately in this section. In addition, a specific preset command is provided for each
digital standard and for the fader. These specific commands are described in the asso-
ciated subsystems.

Four presetting actions are available:

● Activating the default state of all internal instrument functions (*RST on page 286).
  Functions that concern the integration of the instrument into a measurement setup
  are not changed, e.g. GPIB address or reference oscillator source settings.

● Activating the preset state of the parameters related to the selected signal path (:
  SOURce<hw>:PRESet on page 289)

● Activating the preset state of all parameters that are not related to the signal path
  (:DEVice:PRESet on page 288)

● Activating the original state of delivery (factory reset, :SYSTem:FPReset
  on page 289). Only functions that are protected by a password remain unchanged
  as well as the passwords themselves.

:DEVice:PRESet

Presets all parameters which are not related to the signal path, including the LF gener-
ator.

Example:	Presets all instruments settings that are not related to the signal path
Usage:	Event

:SOURce<hw>:PRESet

Presets all parameters which are related to the selected signal path.

The following functions are only preset by command `*RST`: Fading, transient recorder.

Example:	`SOUR:PRES`
	Presets all settings that are related to signal path
Usage:	Event

:SYSTem:PRESet

Triggers an instrument reset. It has the same effect as:

- The [Preset] key.
  However, the command does not close open GUI dialogs like the key does.
- The `*RST` command

For an overview of the settings affected by the preset function, see Chapter 4.2.2, "Default Instrument Settings - Preset Key", on page 96.

Example:	`SYST:PRES`
	All instrument settings (also the settings that are not currently active) are reset to their default values.
Usage:	Setting only

:SYSTem:FPReset

Triggers an instrument reset to the original state of delivery.

Note: "Factory Preset" resets the "Remote Channel" and network settings to the default values.

Executing "Factory Preset" via remote control terminates the connection to the instrument, if these settings had been configured to values different to the default ones.

The factory preset function resets nearly all instrument settings. In addition to the regular preset by means of the [PRESET] key, a "Factory Preset" resets also the following values:

- Reference frequency settings ("Ref Oscillator" menu)
- Power on settings ("Level/EMF" menu)
- Network settings including hostname ("Setup" menu)
- Remote channel settings including GPIB address ("Setup" menu)
- Start/Stop display update ("Setup" menu)
- Display and keyboard settings ("Setup" menu).

To maintain security, password settings and all settings protected by these passwords like disabled USB and LAN connections are not changed.

Not affected by the "Factory Preset" are also user data, lists or instrument settings files, created for example by means of the Save/Recall function.

Example:	`SYST:FPR`
	All instrument settings (also the settings that are not currently active) are reset to the factory values.
Usage:	Event
Manual operation:	See "Factory Preset" on page 122

6.4 CALibration Subsystem

The `CALibration` system contains the commands for performing internal adjustment. Adjustment is triggered by the query commands.

Understanding the query reponse

- `0`: error-free execution of teh adjustements

- `1`: indicates that an error ocurred, the process has been canceled.

Suffix <hw>

Suffix	Value range	Description
`CALibration<hw>`	[1]	Optional suffix

:CALibration:ALL[:MEASure]? [<Force>]

Starts all internal adjustments that do not require external measurement equipment.

Query parameters:
<Force>   string

Return values:
<Measure>   0 | 1 | OFF | ON

Example:	`CAL:ALL:MEAS?`
	`// Response "0"`
	`// Adjustment has been performed successfully`
Usage:	Query only

Manual operation: See "Adjust All" on page 495

:CALibration<hw>:FMOFfset[:MEASure]?

Starts all adjustment for the FM/PhiM modulator.

Return values:

<Measure>   0 | 1

Example: CAL:FMOF?

starts the adjustments for the FM/Phim modulator.
Response: "0"
the adjustments have been performed successfully

Usage: Query only

Manual operation: See "Adjust FM Offset" on page 210

:CALibration<hw>:FREQuency[:MEASure]?

Starts all adjustments which affect the frequency.

Return values:

<Measure>   0 | 1

Example: CAL:FREQ:MEAS?

starts the adjustments for maximum frequency accuracy.
Response: "0"
the adjustments have been performed successfully.

Usage: Query only

Manual operation: See "Adjust Synthesis" on page 495

:CALibration<hw>:LEVel[:MEASure]? [<Force>]

Starts all adjustments which affect the level.

The acquired correction values improve the settling time and the signal quality.

Query parameters:

<Force>   string

                     *RST: force

Return values:

<Measure>   0 | 1

Example: CAL:LEV:MEAS?

starts adjustments for maximum level accuracy.
Response: "0"
adjustment has been performed successfully.

Usage: Query only

Manual operation: See "Adjust Level" on page 495

:CALibration<hw>:LEVel:EXTern:DATA <Data>

Queries what data has been used for the level calibration.

By default the instrument uses correction data obtained in the factory before delivery. In addition, customer data can be used for external level correction. The customer data is obtained using a R&S NRP power sensor. External level correction is a protected function (see service manual, chapter 2, "Adjustment").

Parameters:

<Data>	FACTory	CUSTomer
	*RST: FACTory

Example:	`CAL:LEV:EXT:DATA FACT` selects the use of the data aquired at the factory for external level correction.
Manual operation:	See "Adjustment Data" on page 152

:CALibration:ROSCillator[:DATA] <Data>

Sets the calibration value for the custom defined external adjustment.

Parameters:

<Data>	integer
	Range: 0 to INT_MAX *RST: 0

[:SOURce]:CALibration:STEReo:ANALog[:MEAS]?

The command starts all adjustments which affect the analog channels of the stereo coder option.

Return values:

<Meas>	0	1

Example:	`CAL:STER:ANAL?`
	starts the adjustments for analog channels of the stereo coder.
	`Response: 0`
	the adjustments have been performed successfully.

Usage:	Query only
Options:	R&S SMB-B5
Manual operation:	See "Adjust Stereo Coder" on page 496

6.5 DIAGnostic Subsystem

The `DIAGnostic` system contains the commands used for instrument diagnosis and servicing. SCPI does not define any DIAGnostic commands; the commands listed here

are all device-specific. All `DIAGnostic` commands are query commands which are not influenced by `*RST`.

:DIAGnostic<hw>:BGINfo? [<Board>]

Checks the modules available in the instrument using the variant and revision state.

If the command is sent without parameters being specified, a complete list of all modules is returned (the various entries are separated by commas). The length of the list is variable and depends on the instrument equipment configuration.

If the command is sent with parameters, a list of the specified modules is returned (the various entries are separated by commas). A list of modules names can be called up using the command `:DIAGnostic<hw>:BGINfo:CATalog?` on page 293.

Query parameters:

<Board>   string

Return values:

<BgInfo>   < Module name> <Module stock number incl. variant> <Module revision> <Module serial number>

                             Each entry for one module consists of four parts which are separated by space characters.

Example: `DIAG:BGIN`

                             Queries the instrument configuration.

                             Returns the data of all available modules.

                             `DIAG:BGIN? 'MBRD'`

                             Queries the configuration of the motherboard.

                             `Response: MBRD 1141.3501.02 1.5.3 100023`

                             Module motherboard with part number 1141.3501.01 has revision 1.5.3 and serial number 100023.

Usage: Query only

Manual operation: See "Assembly" on page 98

:DIAGnostic<hw>:BGINfo:CATalog?

Queries the names of the assemblies available in the instrument.

Return values:

<Catalog>   string

                             A complete list of all assemblies is returned (the various entries are separated by commas). The length of the list is variable and depends on the instrument equipment configuration.

Example: `DIAG:BGIN:CAT`

                             Queries the names of the assemblies.

Usage: Query only

:DIAGnostic:INFO:OTIMe?

The command queries the number of operation hours.

Return values:

<OTIMe>   float

Example: `DIAG:INFO:OTIM`
queries the operation hours.
`Response: 100023`
The instrument was operated for 100023 hours up to now.

Usage: Query only

Manual operation: See "Operation Time / h" on page 98

:DIAGnostic:INFO:POCount?

The command queries the number of power-on events.

Return values:

<Pocount>   float

Example: `DIAG:INFO:POC`
queries the number of power on events.
`Response: 123`
The instrument was switched on for 123 times up to now.

Usage: Query only

Manual operation: See "Power On Count" on page 98

6.6 DISPlay Subsystem

The DISPlay subsystem contains the commands to set the power-save mode of the instrument.

:DISPlay:ANNotation:AMPLitude <State>

Indicates asterisks instead of the level values in the status bar.

Parameters:

<State>   0 | 1 | OFF | ON

                           *RST: n.a. (factory preset: 1)

Example: DISP:ANN:AMPL ON
                           Suppresses the level display.

Manual operation: See "Annotation Amplitude" on page 119

:DISPlay:ANNotation:FREQuency <State>

Indicates asterisks instead of the frequency values in the status bar.

Parameters:

<State>   0 | 1 | OFF | ON

                           *RST: n.a. (factory preset: 1)

Example: DISP:ANN:FREQ ON
                           Supresses the frequency display.

Manual operation: See "Annotation Frequency" on page 119

:DISPlay:ANNotation[:ALL] <State>

Displays asterisks instead of the level and frequency values in the status bar of the instrument. This setting is useful when you remotely control the instrument.

Parameters:

<State>   0 | 1 | OFF | ON

                           *RST: 1

Example: DISP:ANN:AMPL ON
                           Shows asterisks instead of frequency and level values.

:DISPlay:DIALog:CLOSe <DialogId>

Closes the specified dialog. To determine the dialog identifier, use command :
DISPlay:DIALog:ID?.

Setting parameters:

<DialogId>   string

Example: DISP:DIAL:CLOS "<dialog ID>"
                           Closes the dialog, determined with the "<dialog ID>".

Usage: Setting only

:DISPlay:DIALog:CLOSe:ALL

Closes all open dialogs.

Example: DISP:DIAL:CLOS:ALL

Usage: Event

:DISPlay:DIALog:ID?

Returns the dialog identifiers of the open dialogs in a string separated by blanks.

Return values:
<DialogIdList>   string

Example: DISP:DIAL:ID?
Response: "<dialog ID(1)> <dialog ID(2)> ...
<dialog ID(n)>"
Returns the dialog identifiers of all opened dialogs.

Usage: Query only

:DISPlay:DIALog:OPEN <DialogId>

Opens the specified dialog. To determine the dialog identifier, use command :
DISPlay:DIALog:ID?.

Setting parameters:
<DialogId>   string

Example: DISP:DIAL:OPEN "<dialog ID>"
Opens the dialog, determined with the "<dialog ID>".

Usage: Setting only

:DISPlay:PSAVe:HOLDoff <HoldoffTimeMin>

Sets the waiting time for the screen-save mode of the display.

Parameters:
<HoldoffTimeMin>   integer

        Range: 1 to 60
        *RST: n.a. (factory preset: 10)
        Default unit: minute

Example: DISP:PSAV:HOLD 8
Sets the timeout of the screen saver to 8 minutes.

Manual operation: See "Wait Time" on page 108

:DISPlay:PSAVe[:STATe] <State>

Activates the screen-save mode of the display.

If activated, the display including backlight is switched off after the wait time elapses and if no entries via front panel, external mouse or external keyboard are made. To set the wait time, use command `:DISPlay:PSAVe:HOLDoff`.

This mode is recommended for protecting the display, especially if you operate the instrument via remote control.

Parameters:

<State>	0 | 1 | OFF | ON
	*RST: n.a. (factory preset: 0)

Example: `DISP:PSAV ON`
Activates screen saver mode.

Manual operation: See "Screen Saver Active" on page 108

:DISPlay:UPDate <Update>

Activates the refresh mode of the display.

Parameters:

<Update>	0 | 1 | OFF | ON
	*RST: n.a. (factory preset: 1)

Example: `DISP:UPD ON`
Activates automatic update of the display at defined time intervals.

6.7 FORMat Subsystem

The `FORMat` subsystem contains the commands which determine the format of the data that the R&S SMB returns to the controller. This affects all query commands which return a list of numerical data or block data. Reference is made to this in the descriptions of the commands.

:FORMat:BORDer <Border>

Determines the sequence of bytes within a binary block. This only affects blocks which use the IEEE754 format internally.

Parameters:

<Border>	NORMal | SWAPped
	NORMal
	The instrument expects (with setting commands) and sends (with queries) the least significant byte of each IEEE754 floating-point number first and the most significant byte last.

SWAPped

The instrument expects (with setting commands) and sends (with queries) the most significant byte of each IEEE754 floating-point number first and the least significant byte last.

*RST: NORMal

Example: `FORM:BORD SWAP`

The data is transferred with the most significant bit first.

:FORMat[:DATA] <Data>

Determines the data format which the R&S SMB uses to return data. When data is transferred from the control computer to the instrument, the instrument detects the data format automatically. In this case, the value set here is irrelevant.

Parameters:

<Data>   ASCii | PACKed

ASCii

Numerical data is transferred as plain text separated by commas.

PACKed

Numerical data is transferred as binary block data. The format within the binary data depends on the command. The various binary data formats are explained in the description of the parameter types.

*RST: ASCii

Example: `FORM ASC`

The data is transferred as ASCII data.

:FORMat:SREGister <Format>

Determines the numerical format which is returned when the status registers are queried.

Parameters:

<Format>   ASCii | BINary | HEXadecimal | OCTal

ASCii

The register content is returned as a decimal number.

BINary

The register content is returned as a binary number. #B is placed in front of the number.

HEXadecimal

The register content is returned as a hexadecimal number. #H is placed in front of the number.

OCTal

The register content is returned as an octal number. #Q is placed in front of the number.

*RST: ASCii

Example: FORM:SREG HEX

The register content is returned as a hexadecimal number.

6.8 HCOPy Subsystem

The HCOPy subsystem contains the commands to generate a hardcopy of the display.

Example: Store a hard copy of the display

The following example lists commands to configure and execute a hard copy to an automatic named file.

```
// ********************************************************************
// Hard copy settings
// ********************************************************************
:HCOPy:DEVice:LANGuage PNG
:HCOPy:FILE:NAME:AUTO:STATe 1
// defines the output format
// sets the instrument to automatically create output file names

// ********************************************************************
// Configure hard copy options, set automatic naming rules
// An automatically generated file name consists of:
// <Prefix><YYYY><MM><DD><Number>.<Format>
// ********************************************************************
:HCOPy:DEVice:LANGuage BMP
// defines output format *.bmp
:HCOPy:REGion DIALog
// selects the region to be copied
:HCOPy:FILE:AUTO:DIR "usb/HCopy"
// sets destination directory of automatic named file to "/usb/HCopy"
:HCOPy:FILE:NAME:AUTO:FILE:PREFix:STATe 1
:HCOPy:FILE:NAME:AUTO:FILE:PREFix:"hardcopy"
:HCOPy:FILE:NAME:AUTO:FILE:YEAR:STATe 1
:HCOPy:FILE:NAME:AUTO:FILE:MONTh:STATe 1
// uses automatic naming prefix
// sets automatic naming prefix to "hardcopy"
// uses automatic naming date parameters year and month

// ********************************************************************
// Execute and transfer the hard copy
// ********************************************************************
:HCOPy:EXECute
:HCOPy:DATA
// generates a hard copy
// transfers the hard copy to the remote client
:HCOPy:FILE:AUTO:FILE?
// queries the automatic file name
```

```
// response: "hardcopy1607001.bmp"
:HCOPy:FILE:AUTO:NUMBer?
// queries the number in the automatic file name
// response: "001"
:HCOPy:FILE:AUTO?
// queries the path and file name of the automatically generated file
// response: "/usb/HCopy/hardcopy1607001.bmp"
```

:HCOPy:DATA?

Transfers the hardcopy data directly as an NByte stream to the remote client.

Return values:

<Data>	block data

Example:	See Example "Store a hard copy of the display" on page 299.
Usage:	Query only

:HCOPy:IMAGe:FORMat <Format>
:HCOPy:DEVice:LANGuage <Language>

Selects the graphic format for the hard copy. You can use both commands alternatively.

Parameters:

<Language>	BMP | JPG | XPM | PNG	
	*RST:	PNG

Example:	See Example "Store a hard copy of the display" on page 299.
Manual operation:	See "File Options" on page 125

:HCOPy[:EXECute]

Generates a hard copy of the current display. The output destination is a file.

The data is written into the file selected/created with the `HCOP:FILE` commands.

Example: See Example "Store a hard copy of the display" on page 299.

Usage: Event

Manual operation: See "Save" on page 126

:HCOPy:FILE[:NAME] <Name>

Determines the file name and path to save the hard copy, provided automatic naming is disabled.

Note: If you have enabled automatic naming, the instrument automatically generates the file name and directory.

Parameters:
<Name>   string

Example: See Example "Store a hard copy of the display" on page 299.

Manual operation: See "File Info" on page 124

:HCOPy:FILE[:NAME]:AUTO?

Queries path and file name of the hard copy file, if you have enabled *Automatic Naming*.

Return values:
<Auto>   string

Example: See Example "Store a hard copy of the display" on page 299.

Usage: Query only

Manual operation: See "File Options" on page 125

:HCOPy:FILE[:NAME]:AUTO:DIRectory <Directory>

Determines the path to save the hard copy, if you have enabled *Automatic Naming*.

If the directory does not exist, the instrument automatically generates a new directory, using the instrument name and /var/user/ by default.

Parameters:
<Directory>   string
                          *RST: /var/user/

Example: See Example "Store a hard copy of the display" on page 299.

Manual operation: See "File Options" on page 125

:HCOPy:FILE[:NAME]:AUTO:DIRectory:CLEar

Deletes all files with extensions `*.bmp`, `*.jpg`, `*.png` and `*.xpm` in the directory set for automatic naming.

Example: See Example "Store a hard copy of the display" on page 299.

Usage: Event

Manual operation: See "File Options" on page 125

:HCOPy:FILE[:NAME]:AUTO:FILE?

Queries the name of the autmatically named hard copy file.

An automatically generated name consists of:

<Prefix><YYYY><MM><DD><Number>.<Format>.

You can activate each component separately, to individually design the file name.

Return values:
<File>   string

Example: See Example "Store a hard copy of the display" on page 299.

Usage: Query only

Manual operation: See "File Info" on page 124

:HCOPy:FILE[:NAME]:AUTO[:FILE]:DAY:STATe <State>
:HCOPy:FILE[:NAME]:AUTO[:FILE]:MONTh:STATe <State>
:HCOPy:FILE[:NAME]:AUTO[:FILE]:YEAR:STATe <State>

Uses the date parameters (day, month or year) for the automatic naming. You can activate each parameter separately.

Parameters:
<State>   0 | 1 | OFF | ON
                      *RST: 1

Example: See Example "Store a hard copy of the display" on page 299.

Manual operation: See "File Options" on page 125

:HCOPy:FILE[:NAME]:AUTO[:FILE]:NUMBer?

Queries the number that is used as part of the file name for the next hard copy in automatic mode.

At the beginning, the count starts at 0. The R&S SMB searches the specified output directory for the highest number in the stored files. It increases this number by one to achieve a unique name for the new file.

The resulting auto number is appended to the resulting file name with at least three digits.