



Script Language description

General Syntax

The general style is Pascal:

- x There is no case sensitivity.
- x The program starts with **begin** and stops at **end..**
- x () - brackets are used in mathematical formulas.
- x Strings are enclosed in ' ' (" " is also working).
- x Decimal separator for floating point numbers is a point: ".".
- x Each command is completed with a semicolon: ";".
- x Available mathematical operations: + - * / ^
- x Available mathematical functions: **TAN, SIN, COS, ARCTAN, SGN, ABS, LN, LOG, SQRT, SQR ..**
- x Available relational operations: = > < <>

Variables & Assignments

Variables up to 8 characters long. The data format of all variables is real.

Example for variable usage: **A := A + 1.2;**

Command Overview

Command	Meaning
Beep	Play a sound
ClrScr	Deletes the visible data on the data screen
CCDScan	Takes a spectrum with the optical spectrometer Spec64
DNCPara(n,v)	Sets the n-th parameter of the DNC window to the value v
DNCStart	Record a Frequency Spectrum
Execute('filename')	Executes another script <i>./script/filename.scr</i>
FeedPara('t', v)	Sets the parameter t for the feedback to the value v
FileName('name1')	Defines the name of the data storage file as <i>./data/name1.txt</i>
for .. to .. begin	Loop definition
GetChannel (Ch)	Acquires data from output or input channels of the software
GetFeedPara(v)	Gets the value of the feedback parameter v and stores it in the variable c
GetScanPara(p)	Get the parameters of the scan parameter window
Goto <i>marke</i>	Jumps to the position <i>marke</i> defined as Label
GoXY(x,y)	Moves tip to a position inside the current scan range



Supplier: Anfatec Instruments AG, Melanchthonstr. 28, 08606 Oelsnitz
Device: Scanning Probe Controlling System -- SXM-Software 28.8
Document: **Language Description**

if .. then	Condition
Move(dir,dist)	Uses the current coarse positioning settings to move sample/ tip
MLockin	Controls the Multi-Lockin Window
ScanPara(p,v)	Sets the parameter p in the ScanParameter window to the value v
ScanImage	Starts to acquire an image
ScanLine(Nr)	Scan Nr lines
SetChannel(Ch, v)	Sets an output channel number Ch to a value v
SetHardChannel(a, v)	Writes to Address a the value v
SetLaser()	Switches the laser On or Off
SpectPara(n,v)	Sets parameter n for the spectrum acquisition to the value v
SpectStart	Starts to acquire and save a spectrum with current settings.
TipCond(type, val)	Tip Conditioning
VPulse()	Voltage Pulse
Wait(t)	Waits for a time t in seconds
WriteLn('text',...)	Displays text in the Memo

Commands (alphabetical)

Beep

Example: Beep ;

ClrSrc

deletes all visible data on the memo.

Example: ClrSrc ;

CCDScan

takes a spectrum with the attached spectrometer with the current spectrometer settings

Example:

```
GoXY(0.5,0.5) ; // Moves to middle of the scan range
CCDScan ; // Takes a spectrum
```

DNCPara (n, value) command

Set the n-th parameter of the DNC window to the value

Parameters: n: number of the parameter (1 ... 8)
or "AUTOSAVE", INPUTGAIN

value: real

Assignment: n <--> **Meaning**

1 Start Frequency



Supplier: Anfatec Instruments AG, Melanchthonstr. 28, 08606 Oelsnitz
Device: Scanning Probe Controlling System -- SXM-Software 28.8
Document: **Language Description**

- 2 Stop Frequency
- 3 Use Frequency
- 4 Drive Amplitude
- 5 Bandwidth Button (On = 1 /off = 0)
- 6 & 7 TimeConst., RollOff used if Bandwidth button is ON
- 8 Thermal Noise Button (On = 1 /off = 0)

Example:

DNCPara (3,100) ; → sets the used frequency to 100

DNCPara ('INPUTGAIN', 10);

Attention: The values are overtaken in the currently selected physical unit. If the unit of frequency is set to “kHz”, then **DNCPara (3,10) ;** results in 10.000,00 Hz. If the unit is set to “Hz”, then **DNCPara (3,10) ;** results in 10,00 Hz.

Execute (scriptfilename)

Calls and executes another script file.

Parameters: **scriptfilename:** string

Example:

Execute ('Ref-HOPG') ; → calls the script “ ./scripts/Ref-HOPG.scr ”.

FeedPara ('type', value)

Sets parameters for the feedback system.

Parameters: **type:** string from list
allowed values: Ref, Ki, Kp, Bias, Flyheight ,
enable, preamp, ref2, ki2, kp2, mode, Ratio, BiasDiv
and zOffset, zOffsetSlew, enable from the zControl Window
value: real

Example:

FeedPara ('Ki' , 100) ; → sets Ki in the parameter window to 100

FileName (YourFileName)

defines a file name for the log file, which collects all data during one script run.

If no file name is defined in the script, the system writes its messages into a file *NoName.txt*. With each run of the script, old data in *NoName.txt* are deleted.

If a file name is defined, the file name given by the user is extended automatically by the system with a number representing current date and time (date-time-string), so that the log file is *YourName_DateTime.txt*. Thus, the user does not need to rename its file in the script for every run.

Parameters: **YourFileName :** string → file is *./data/YourFileName_DateTime.txt*

Example: **FileName ('t3') ;** → the data are written to *./data/t3_XXXXXXXX.txt*

for start counter to stop counter do begin command(s) end;

Loop definition based on an integer counter. The commands **do begin** and **end** are always required, even if there is one program line in between, only.

Example (measures 20 points along a line in x-direction):



Supplier: Anfatec Instruments AG, Melanchthonstr. 28, 08606 Oelsnitz
Device: Scanning Probe Controlling System -- SXM-Software 28.8
Document: **Language Description**

```
x:=100;  
for i:=1 to 20 do begin  
  Move('CX',X);  
end;
```

GetFeedPara ('type')

Gets parameters from the feedback system.

Parameters: **type:** string from list of values: Ref, Ki, Kp, Bias, Flyheight
result: real number

Example:

```
c:=GetFeedPara('Ki'); Writeln(c);  
→ writes the value of Ki on the screen
```

GetChannel (Address)

Acquires the data of one A/D input channel or, if negative, the content of the Mirror Memory for the corresponding DAC

Parameters: **Address:** integer

for the A/D-Channels, the Address can found in /Options/Scale/ADC as Port. Here, "32" stands for the 1st ADC-channel (e.g. "T-B" for AFM). To get data from the D/A-channels, one chooses negative channel numbers. E.g, "0" equals the channel "Z" and "-1" the channel "Bias".

Example: **c:=GetChannel(32);** → reads the input data of the channel AD1
c:=GetChannel(-1); → reads the bias output provided at DAC 1

GetScanPara('para')

Returns the currently set scan parameters.

Parameters: **para:** string from list of values: Range, Speed, Pixel, X, Y, Angle, SlopeX, SlopeY, LineNr, 1D, AspectRatio, PixelDensity, DriftX, DriftY, Scan, Autosave
result: real number

Example: **c:=GetScanPara('scan');** → returns status of scan, 0 = scan off, 1 = scan on
Writeln(c); → writes status to screen

Goto marke

Allows to jump to a *marke*. It requires:

- the definition of a label with a **LABEL** statement
- a **Goto** command followed by the labels name
- the label followed by a colon

Labels always start with a character..

Example:

```
LABEL foo, exit;  
begin  
foo:  
  GetChannel(32);
```



Supplier: Anfatec Instruments AG, Melanchthonstr. 28, 08606 Oelsnitz
Device: Scanning Probe Controlling System -- SXM-Software 28.8
Document: **Language Description**

```
if c > 1000 then Goto foo;  
exit: writeln('Done. ');  
end.
```

GoXY (*x_rel*, *y_rel*)

Moves the tip to the relative positions “x_rel” and “y_rel” inside the currently given scan range. The upper left corner of the image equals (0,0), while the lower right corner is (1,1).

Parameters: *x_rel*, *y_rel*: real numbers between 0 and 1

Example: `GoXY (0.5, 0.5) ;` → moves the tip to image center
`GoXY (0.1, 0.1) ;` → moves tip to a point close to the upper left corner

if *condition* then *command*

Evaluation of conditions. ELSE is not supported. Implementation is not fully Pascal compliant.

Example:

```
if c < -160 then writeln('Channel X =', c);
```

Note: Starting from version 28.8, **begin** and **end** are supported in order to apply several commands after the condition.

Move (*'direction'*, *distance*)

Allows to operate the coarse positioning system, to set the center position for whole images and to address the functions behind the approach button and retract button in the parameter window. Latter are addressed by their number in the item list, that appears with the right mouse button.

Parameters: *direction* : string from list:

allowed string values: cx, cy, cz, x, y, retract, approach

distance : real → value in physical units or steps

Examples: `Move ('CX', 10) ;` → moves the stepper 10 steps in x-direction

`Move ('Y', -1.5) ;` → sets the center position to -1.5 μm

`Move ('Approach', 2) ;` → starts automated approach

`Move ('Retract', 0) ;` → retracts the piezo, only

MLockin (*'type'*, *value*) SW.Ver. > 28.7

Control the Multi Channel Lockin Window.

Parameters: *type* : Component Name as string

Examples: `MLockin ('Edit1', 1000) ;` → Lia1 Frequency = 1000

Note: Visualization of components names is given on page 9.

ScanPara (*'type'*, *value*)

allows to set scan parameters in the window “Parameter”. With the type 'Autosave', the knob 'Autosave' for the image acquisition is toggled.

Parameters: *type* : string from list:



Supplier: Anfatec Instruments AG, Melanchthonstr. 28, 08606 Oelsnitz
Device: Scanning Probe Controlling System -- SXM-Software 28.8
Document: **Language Description**

allowed string: range, speed, pixel, x,y, angle, linenr, autosave, aspectratio, pixeldensity, slope, slopex, slopey, 1D, scan, driftx, drifty or the component Name

value : real → value in physical units
→ or index for Combo-Boxes start with counting at 0

Examples:

ScanPara ('Speed', 1.5); → line speed set to 1.5 lines/s
ScanPara ('X', -0.5); → image center X set to -0.5 μm
ScanPara ('Autosave', 1); → switches automated save on
ScanPara ('Pixel', 2); → 3th entry in drop down list => 128
ScanPara ('Edit24', 12); → Ki for Ampl.-Feedback to 12

Note: More parameters can be called by their component names as described on page 99.

ScanImage

Starts the acquisition of an image. When two or more images need to be acquired, this command has to be repeated.

Example: **ScanImage;** → starts to acquire a whole image

ScanLine (Number)

Starts the scan of Number lines with the current settings. If the scan mode is “2D”, it starts to scan a number of lines in an image at the current scan position. If the scan mode is “1D” and “cont. on restart” is off, it scans the 1st line of an image. In “1D” with “cont. on restart” off, it scans the same line (number) times at the current position.

Example: **StartLine (10);** → starts to acquire 10 lines with the current settings

SetLaser(value)

Switches the laser on or off.

Example: **SetLaser (0); Wait (1); SetLaser (1);**
→ switches the laser off for 1 second

SetChannel(channelnumber, value)

Sets the output of an I/O channel to the value given as “v”.

Parameters: **channelnumber:** integer
value: real

SetChannel is possible for D/A-channels only. In order to set a value of a D/A-channel, one chooses a (positive) channel number in contrast to GetChannel. E.g, “0” equals the channel “Z” and “1” the channel “Bias”.

Example: **c:=GetChannel (0);** → reads the position of z
SetChannel (0, c+0.1); → set the position of z 0.1 nm higher
SetChannel (1, -0.6); → set the Bias at DAC1 to -0.6 V

Note: Physical unit and Scaling are set in /Options/Scale/DAC



SpectPara (Parameter, value)

allows to set parameter in the window “spectroscopy”. With *Parameter* as 'Autosave', the knob 'Autosave' in the spectroscopy window can be set. When data are saved automatically, they are saved numbered in the directory ./data/.. In order to find out which data equal which setting, the complete path of the saved file is written into the log file (*NoName.txt* or *YourName_DateTime.txt*).

Parameters: Parameter: number or string
number=0 type of spectrum from the drop down list
number>0 Number of the Edit Box
Strings 'AUTOSAVE', 'POINTS', 'REPEAT'

Examples: `SpectPara (0, 0);` → sets the type to 1st index → “X(z)”
`SpectPara (5, 50);` → sets the distance dz to 50
`SpectPara ('autosave', 0);` → AutoSave off
`SpectPara ('autosave', 1);` → AutoSave on

SpectStart

Starts a spectroscopy data acquisition with the current settings.

Example: `SpectStart;` → starts a spectrum with current settings

TipCond(type, val)

Type: BIAS, TIME, DIVE MODE DO

Example: `TipCond('Bias', 10);`
`TipCond('Do', 0);` → start

Vpulse (PulseTime, Bias);

Change the Bias-Voltage for “Time” to “Bias”

Example: `VPulse(1.0, 2);`

Note: Physical unit is set in /Options/Scale/DAC

Wait (time)

Allows to pause the operation a time given in seconds.

Parameters: `time` : real

Example: `Wait(1.5);` → waits for 1.5 seconds

Writeln ('text1', variable, 'text2', variable2 ...)

Allows to write additional information in the memo and to the log file (*NoName.txt* or *YourName_DateTime.txt*) simultaneously. In the brackets, a series of strings, variables and numbers can be written, which are separated by commas. Strings should be enclosed in ' '.

Parameters: `text1, text2` : string; `variable1, variable2` : real

Example: `a := 5;` → sets the value of the variable *a* to 5
`writeln('a=', a);` → writes “a = 5” in the Memo



Component names

Components are Edit windows, Labels, TrackBars, ComboBoxes or others. In the SXM script software, almost all parameters can be addressed in read (e.g. GetScanPara) or write (e.g. ScanPara) calls by the name of their component. Also, in the ini-file, their parameters are stores by their component name.

Example calls using component names

These two command equal each other

```
ScanPara('Edit1', 0.89); and ScanPara('SetPoint', 0.89);
```

This command addresses Kp for the amplitude feedback in PLL feedback mode:

```
ScanPara('Edit32', 50);
```

TrackBars are addressed with values between 0 (left end) and 100 (right end).

```
ScanPara('TrackBar1', 90); results in
```

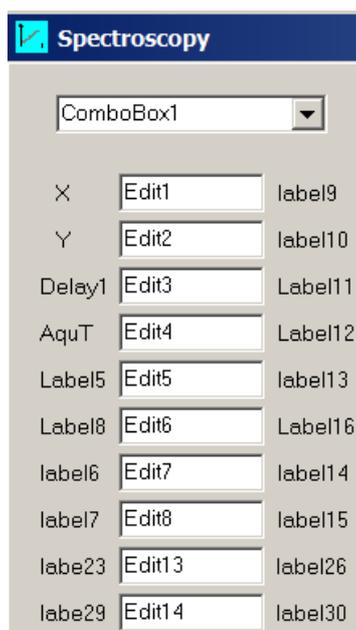


ComboBoxes and SpeedButton are addressed by the count of their list entries. The first visible entry has the count "0".

```
ScanPara('ComboBox1', 1); equals ScanPara('Pixel', 64);
```

It is not possible to enter values which are different from the already provided list entries.

Spectroscopy Windows



Examples:

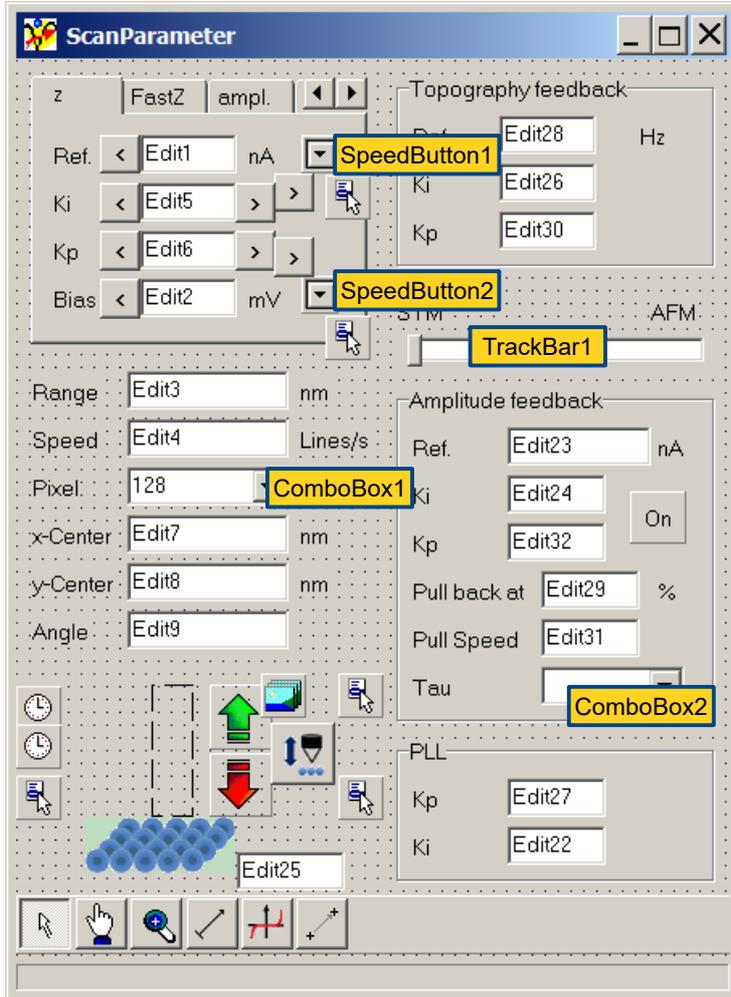
```
SpectPara('ComboBox1', 1);
```

```
SpectPara('Edit5', 500);
```

Note that values in the Edit windows are overtaken in the physical unit which is currently selected in the related labels. As label names can be changed by clicking on them from V → mV → μV → ... and back, this is a common error source.

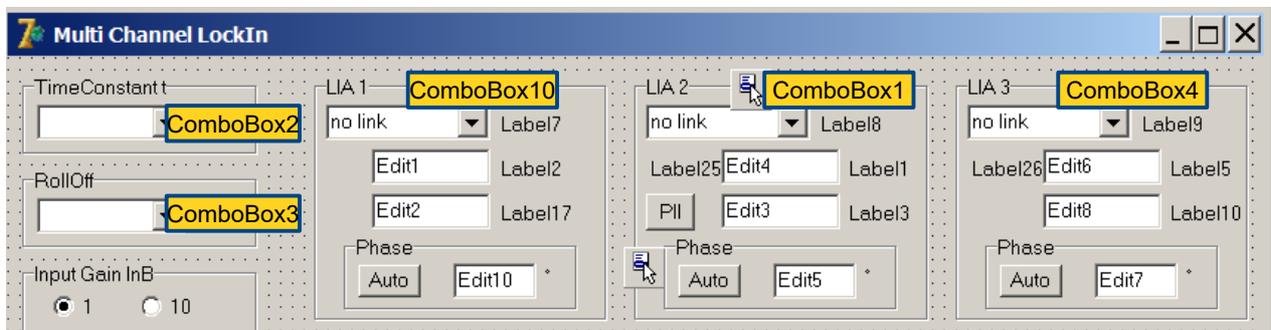


Components of Scan Parameter Window



Note that values in the Edit windows are overtaken in the physical unit which is currently selected in the related labels. As label names can be changed by clicking on them e.g. from μm \rightarrow nm and back, this is a common error source.

Components of MultiLockIn window





Example programs

Scan a set of images with different sizes

Begin	required at the begin of each program
Move ('Approach', 2);	Automated approach
Wait (1);	waits 1 s to stabilize conditions
FeedPara ('Ki', 1000);	sets Ki=1000
ScanPara ('Speed', 2);	sets Speed = 2 lines/s
ScanPara ('X', 0);	sets center position X = 0 μm
ScanPara ('Y', 0);	sets center position X = 0 μm
ScanPara ('AutoSave', 1);	switches the automated image save knob ON
for i:=1 to 3 do begin	loop for I = 1, 2 and 3, so that
R:=i*20-10;	this equation sets R to 10, 30 and 50.
ScanPara ('Range', R);	sets scan range to R = 10 μm , 30 μm , or 50 μm
ScanImage;	starts one image
end;	end of loop.
end.	end of program.

Data acquisition with Stop at a certain condition

this program exits, if the input channel AD1 is > 1000 .

Label back, exit;	defines jump labels “back” and “exit”
begin	begin of program
c:=0;	sets the variable c to zero.
back:	position of jump label “back”
GetChannel (32);	read Channel No. 32 = T-B and put the result into the variable c
writeln(c);	writes result into result file
if c < -1000 then Goto exit;	exit the program, if result < -1000
Goto back;	otherwise jump back to check channel 32
exit:	position of jump label “exit”
writeln('exit');	writes the text “exit!” in the result file.
End.	end of program