The SSA® project file format

28. October 2016

Table of Contents

1 The SSA® project file format

1.1 General description

The SSA® project file format is equivalent to the standard .ini file format. It consists of line-by-line key-value pairs, which are subdivided in sections due to structural reasons.

General rules:

- key and value are separated by an „=“
- blanks before or after the „=“ will be ignored
- section names are delimited by a preceding “[” and a following “]”
- a section name must be unique within a file
- a key name must be unique within a section
- upper-/lowercase will be ignored
- comments don't need a special preceding sign, but have to be in extra lines

A short example of such a project file:

```
[file]
path=c:\programme\sio\filmdoctor
name=test.fdssa
```

[data]
amount=7
sum=12

1.2 Specific rules of the SSA® project file format

There are some specific rules of the project file format of the SSA® module:

- the file extension must be „**.fdssa**“
- the indenter geometry parameters must be provided in the section „[indenter]“
- The decimal separator is the dot (“.”).
- Basis for geometric values is the SI base unit meter (m).
- Basis for time values is the SI base unit second (s).
- Consequently, basis for all pressure and stress values is the unit Pascal (Pa) and basis for all force values is the unit Newton (N).

1.3 Sections of the SSA® project file format

The following subchapters briefly describe the different sections of the SSA® project file format.

1.3.1 Material

Section name: material

Content: Parameters of each constituent of the sample material (Young's modulus, Poisson's ratio, thickness and so forth)

For a list of all keys see the subchapter “material parameter keys” in appendix.

1.3.2 Definition of the contact situation

Section name: indenter

Content: Description of the contact situation (indenter material and geometry, maximum normal load)

For a list of all keys see subchapters “load definition keys” and “indenter geometry keys” in appendix.

1.3.3 Pre-scan topography

Section names: topography, prescan

Content: The topography data before scratch measurement.

The topography before the the measurement can be described in two ways. As an area in x- and y-direction or as a scan along one axis. In the first way the data is stored in the topography section and a list of all keys is shown in subchapter “topography keys” in appendix. The details for the one axis scan are shown in the subchapter “scan keys” in the appendix.

1.3.4 Scratch

Section name: scratch

Content: The parameters measured during the scratch measurement. Either friction or lateral force values must be given.

For a list of all keys see subchapter “scratch keys” in appendix.

1.3.5 Post-scan topography

Section name: postscan, postscan-topography

The behaviour is similar to the prescan but the section names are different.

1.3.6 Animation

Section name: animation

Content: The parameters for the animation of the scratch in SSA®.

For a list of all keys see subchapter “animation” in appendix.

1.3.7 Scratch series

Section name: scratch_series

Content: The measurement information of additional scratch measurements if more than one scratch was performed. The measured values of each scratch are stored in seperate sections.

These sections are named “scratch_NR” , where NR = {1, …, scratch_count}

For a list of all keys see subchapter “scratch series keys” in appendix.

1.4 Special features

1.4.1 Definition of the order of magnitude of units

To enhance the readability the exponent of a value is stored in a different key. The exponent will always be stored as the exponent of the base unit (m, m², N, Pa, s). For example: The key-value pairs

```
normal_force_value=1.5
normal_force_factor=-2
```

define a normal load of 15 mN.

You don't have to provide digits to the left of the decimal point. The following entries are valid and equivalent to each other:

```
normal_force_value=1.5
normal_force_factor=-2

normal_force_value=0.015
normal_force_factor=0

normal_force_value=1.5e-2
normal_force_factor=0
```

2 Appendix

2.1 A material parameter keys

name	type	description	standard value
layer_NR_ny	double	Poisson's ratio of the layer NR	different
layer_NR_E_value	double	Young's Modulus of the layer NR	different
layer_NR_E_factor	integer	order of magnitude of E value	9 (stands for GPa)
layer_NR_height_value	double	height of the layer NR	1, NR={1, ..., number of layers
layer_NR_height_factor	integer	order of magnitude of height	-6 (stands for µm)
layer_NR_inx_value	double	intrinsic stress in x direction	0
layer_NR_inx_factor	integer	order of magnitude of the inx value	9 (stands for GPa)
layer_NR_iny_value	double	intrinsic stress in y direction	0
layer_NR_iny_factor	integer	order of magnitude of the iny value	9 (stands for GPa)
layer_count	integer	number of layers	1

Table 1: description of the material keys

NR = {0, …, number of layers}; NR>=1: coating layers; NR=0: substrate

2.2 B load definition parameters

name	type	description	standard value
indenter_ny	double	Poisson's ratio of the indenter	0,07
indenter_E_value	double	Young's Modulus of the indenter	1141
indenter_E_factor	integer	order of magnitude of Young's modulus	9 (stands for GPa)
indenter_geometry	integer	Defines the geometry of the indenter (see section B1)	0 (stands for a sphere)
contact_load_value	double	Defines the maximum normal load of the scratch measurement	1
contact_load_factor	integer	Order of magnitude of load	0 (stands for N)

Table 2: description of the load definition keys

2.2.1 B1 indenter geometry keys

4 different indenter geometries are possible, each of which needs different parameters.

Sphere: indenter_geometry=0

name	type	description	standard value
indenter_radius_value	double	radius of the indenter	200
indenter_radius_factor	integer	Order o magnitude of the indenter radius	-6 (stands for µm)
indenter_angle_value	double	angle of the sphere	60

Paraboloid: indenter_geometry=1

The shape is calculated by the following formula: $Z(r) = d_0 \cdot r^2 + d_2 \cdot r^4 + d_4 \cdot r^6 + d_6 \cdot r^8$

name	type	description	standard value
indenter_d0_value	double	D0 parameter of the indenter	1
indenter_d2_value	double	D2 parameter of the indenter	0
indenter_d4_value	double	D4 parameter of the indenter	0
indenter_d6_value	double	D6 parameter of the indenter	0

Cone with rounded tip: indenter_geometry=2

name	type	description	standard value
indenter_radius_value	double	radius of the round indenter tip	200
indenter_radius_factor	integer	order of magnitude of the indenter radius	-6 (stands for µm)
indenter_angle_value	double	angle of the cone	60

Flat punch with rounded edges: indenter_geometry=3

name	type	description	standard value
indenter_edge_radius_value	double	radius of the round indenter edge	200
indenter_edge_radius_factor	integer	order of magnitude of the indenter edge radius	-6 (stands for µm)
indenter_radius_value	double	radius of the indenter	5
indenter_radius_factor	integer	order of magnitude of the indenter radius	-6 (stands for µm)

Area function: indenter_geometry=4

area_func_exponent_X	double	part of the area function for h^X, where X can be [5, 4, 3, 2, 3/2, 1, 2/3, 0, 1/2 , 1/4 , 1/8 , 1/16 , 1/32, 1/64, 1/128]	0 for all except for X=2, if X=2 the value is 24,5
area_function_is_square_root	integer	indicates whether the area function is defined as square root (1) or not (0)	0
area_func_start	double	start value of the definition range for the area function	0
area_func_end	double	end value of the definition range for the area function	0
area_func_unit	integer	order of magnitude for the area function	-6 (stands for µm)

2.3 C Topography keys

name	type	description	standard value
x_point_count	integer	number of measured points in x-direction	1
y_point_count	integer	number of measured points in y-direction	1
depth_factor	integer	order of magnitude of depth values	-6 (stands for µm)
x_factor	integer	order of magnitude of values in x-direction	-6 (stands for µm)
y_factor	integer	order of magnitude of values in y-direction	-6 (stands for µm)
x_start	double	startpoint in x-direction	0

x_end	double	endpoint in x-direction	0
y_start	double	startpoint in y-direction	0
y_end	double	endpoint in y-direction	0
x_NR1_y_NR2	double	depth value for a measurement point	0, NR1={1, …, x_point_count}; NR2={1, …, y_point_count}

Table 3: description of the topography keys

2.3.1 C1 Examples

The following example describes a 2D topography scan with 3 measurement points in each direction. The values are given in µm.

```
[topography]
x_point_count=3
y_point_count=3
depth_factor=-6
x_factor=-6
y_factor=-6
x_start=0
x_end=3
y_start=4
y_end=7
x_1_y_1=0,2
x_1_y_2=0,3
x_1_y_3=0,4
x_2_y_1=0,24
x_2_y_2=0,35
x_2_y_3=0,23
x_3_y_1=0,04
x_3_y_2=0,09
x_3_y_3=0,13
```

The next example describes a 1D topography in x-direction, with 10 measured values in µm. Unnecessary keys are left out. SSA® will use standard values instead.

```
[topography]
x_point_count=10
x_start=3
x_end=5
x_1_y_1=0,3
x_2_y_1=0,23
x_3_y_1=0,21
x_4_y_1=0,34
x_5_y_1=0,1
x_6_y_1=0,26
```

x_7_y_1=0,22

x_8_y_1=0,19

x_9_y_1=0,13

x_10_y_1=0,09

2.4 D Scratch keys

name	type	description	standard value
scratch_point_count	integer	number of measured points in scratch-direction	1
scratch_speed	double	scratch velocity in m/s	0
loading_rate	double	scratch loading rate in N/s	0
path_factor	integer	order of magnitude of path values in scratch-direction	-6 (stands for µm)
normal_force_factor	integer	order of magnitude of normal force values	0 (stands for N)
lateral_force_factor	integer	order of magnitude of lateral force values	0 (stands for N)
time_factor	integer	order of magnitude of time values	0 (stands for s)
penetration_depth_factor	integer	order of magnitude of penetration depth values	-6 (stands for µm)
post_scan_factor	integer	order of magnitude of post scan values	-6 (stands for µm)
path_value_NR	double	the path value of point NR	NR / scratch_point_count
normal_force_value_NR	double	the normal force value of point NR	NR / scratch_point_count
lateral_force_value_NR	double	the lateral force value of point NR	0
time_value_NR	double	the time value of point NR	NR / scratch_point_count)
friction_value_NR	double	the friction value of point NR	0
penetration_depth_value_NR	double	the penetration depth value of point NR	0
post_scan_value_NR	double	the post scan value of point NR	0
use_x_and_y_values	integer	determines if the scratch was performed along the x-axis (0 if yes, 1 if not)	0 (means scratch along x axis was performed)
x_factor	integer	order of magnitude of x values during scratch	-6 (stands for µm)
y_factor	integer	order of magnitude of y values during scratch	-6 (stands for µm)
x_value_NR	double	the position in x-direction of point NR	path_value_NR
y_value_NR	double	the position in y-direction of point NR	0

Table 4: description of the scratch measurement keys

NR={1, …, scratch_point_count}

2.5 E Scan keys

name	type	description	standard value
point_count	integer	number of measured points	1
x_factor	integer	order of magnitude of values in x-direction	-6 (stands for µm)
y_factor	integer	order of magnitude of values in y-direction	-6 (stands for µm)
z_factor	integer	order of magnitude of values in z-direction	-6 (stands for µm)
x_value_NR	double	x-axis value for a measurement point	0, NR={1, …, point_count}

y_value_NR	double	y-axis value for a measurement point	0, NR={1, …, point_count}
z_value_NR	double	depth value for a measurement point	0, NR={1, …, point_count}

2.6 E Animation keys

name	Type	description	standard value
step_count	Integer	number of animation steps to be calculated	10
x_start_pos	Double	start position in x-direction	0
x_end_pos	Double	end position in x-direction	0 (stands for N)
x_factor	Integer	order of magnitude of x-values	-6 (stands for µm)
y_factor	Integer	order of magnitude of y-values	-6 (stands for µm)
y_start_pos	Double	start position in y-direction	0
y_end_pos	Double	end position in y-direction	0

Table 5: description of the animation keys

2.7 F Scratch series keys

name	type	description	standard value
scratch_count	integer	number of performed scratches, the measured values of each scratch must be given in sections named “scratch_NR”	0

Table 6: description of the scratch series keys

NR = {1, …, scratch_count}