Altair® FluxMotor® 2024.1

Synchronous Machines with wound field – Inner Salient Pole - Inner rotor

Motor Factory – Export

General user information

Altairhyperworks.com

Contents

1 MOTOR FACTORY – EXPORT AREA – HOME PAGE VIEW

The area “EXPORT” of Motor Factory groups two main families of functions:

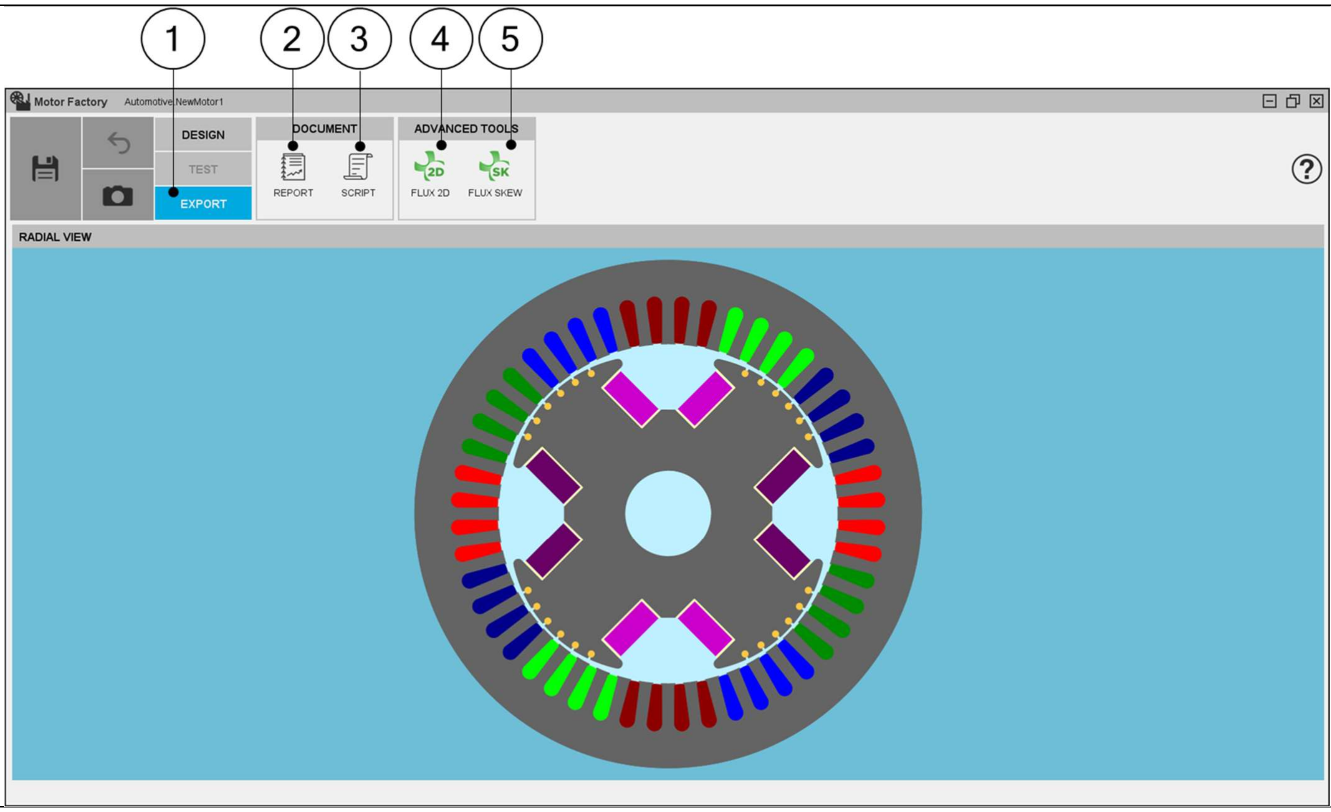
1.1 “DOCUMENT”

In “DOCUMENT”, the function “REPORT” allows building reports automatically to describe all the work achieved for the design as well as for the tests.

Then, the function “SCRIPT” allows to build and export a python script of a current motor in the application Script Factory or in a targeted folder.

1.2 “ADVANCED TOOLS “

In “ADVANCED TOOLS”, the functions “FLUX 2D”, FLUX SKEW allow to build and export a model in Altair® Flux® environment (2D, or Skew) for performing advanced studies either with magneto static or transient applications.



Motor Factory - EXPORT area

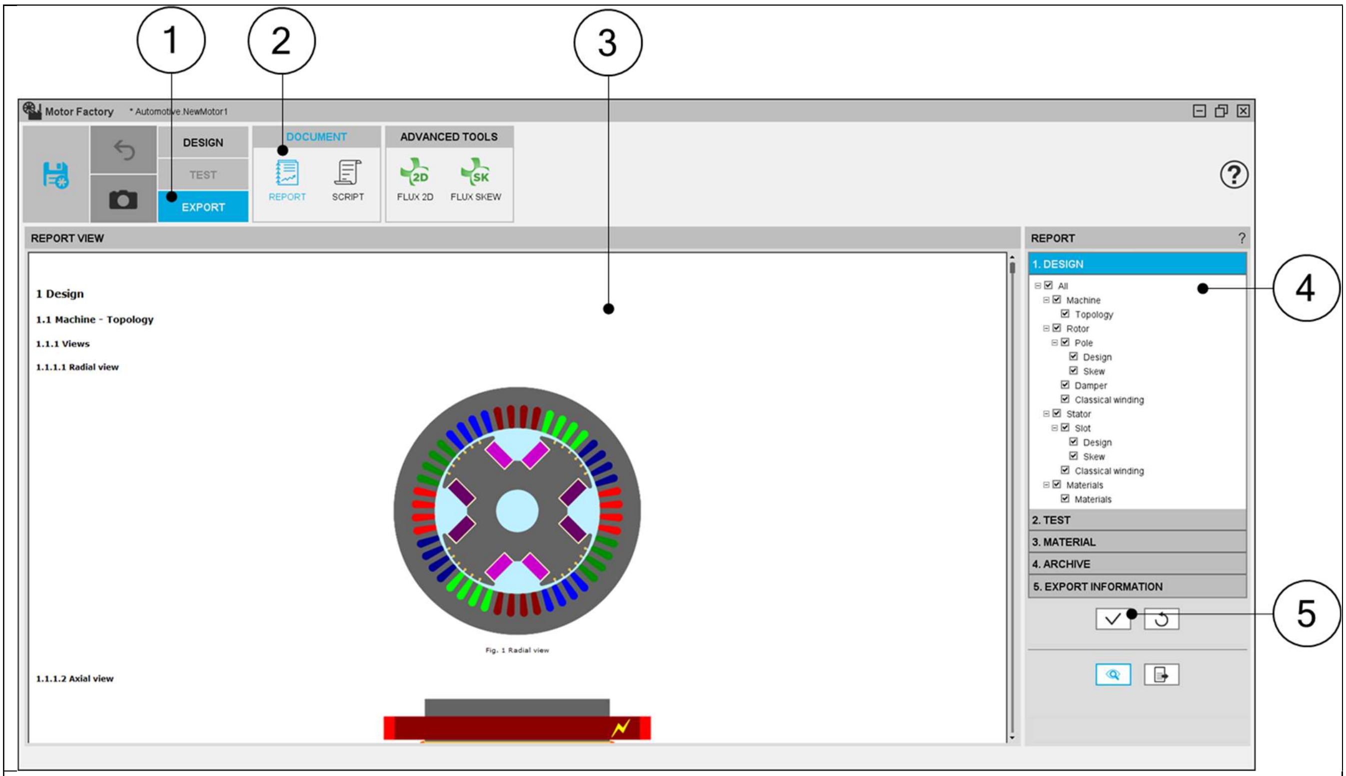
1	Selection of the EXPORT area of Motor Factory.
2	Access the area “REPORT” in which a report can be made
3	Access the area “SCRIPT” for generating a python file in which all the needed command lines are written to rebuild the motor
4	Access the area “FLUX2D” in which a model can be made and sent to Altair® Flux® 2D
5	Access the area “FLUX SKEW” in which a model can be made and sent to Flux® Skew

2 MAKE A REPORT

2.1 Overview

The aim of this export is to build and quickly export a report showing all the work achieved to design and test the machine.
As a result, the report can be exported in a pdf or html file format. It can also be attached to the motor in the "Motor Catalog" or simply displayed in the report area.

2.2 Area to build the report.



Motor Factory – EXPORT AREA – Export a report

1	Selection of the EXPORT area of Motor Factory.
2	Access the area in which a report can be made
3	Zone to visualize the report (= preview)
4	5 steps to build the report which user need
5	Buttons to validate inputs, display a preview and export a report

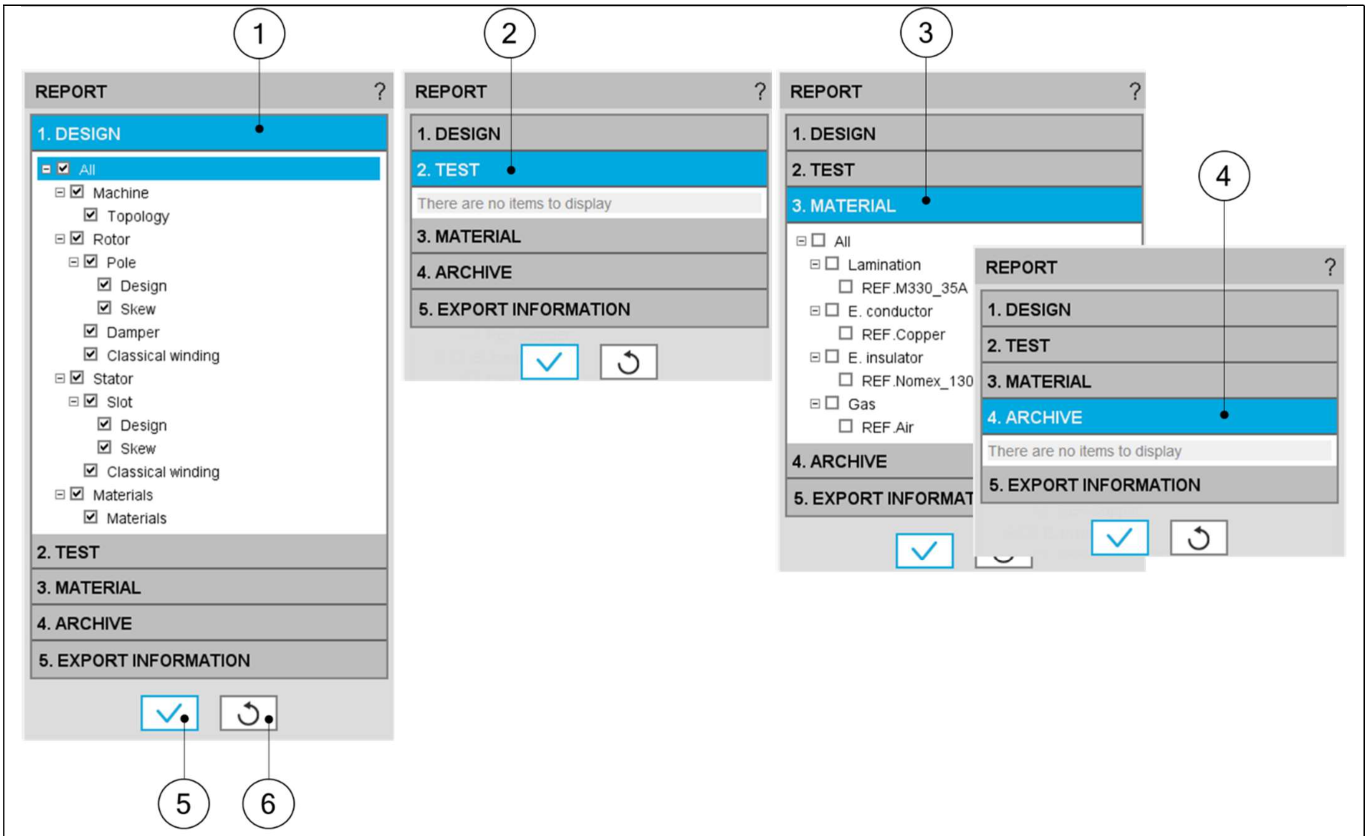
2.3 Steps to build and export a report.

Five steps are needed to build and export a report: In EXPORT / DOCUMENT / REPORT area:

1) Select the sections to write dealing with the design.
2) Select the sections to write dealing with the tests.
3) Select the sections to write dealing with the materials.
4) Select the “saved test results” you want to add as archive in the report.
5) Define the export information.

2.4 Section selection

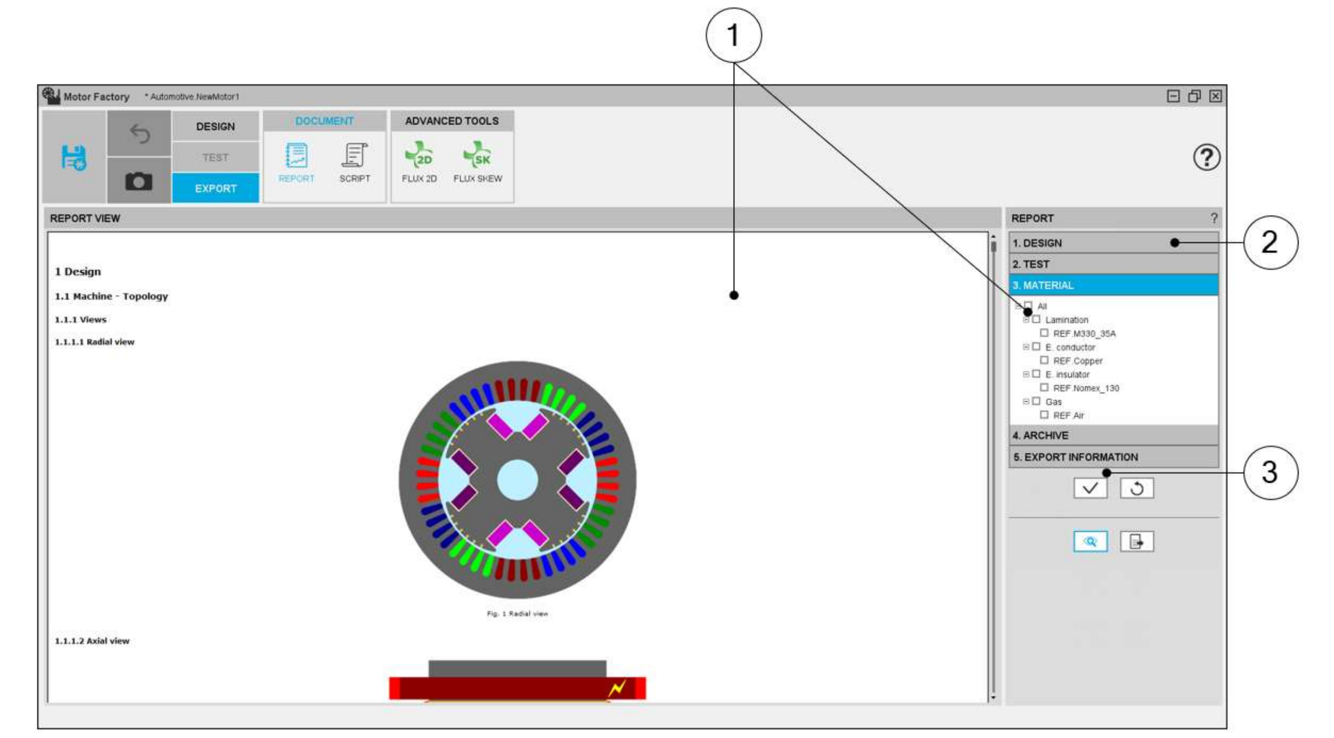
2.4.1 List of sections available to build the report



Motor Factory - EXPORT AREA – Export a report – Chapters to be selected

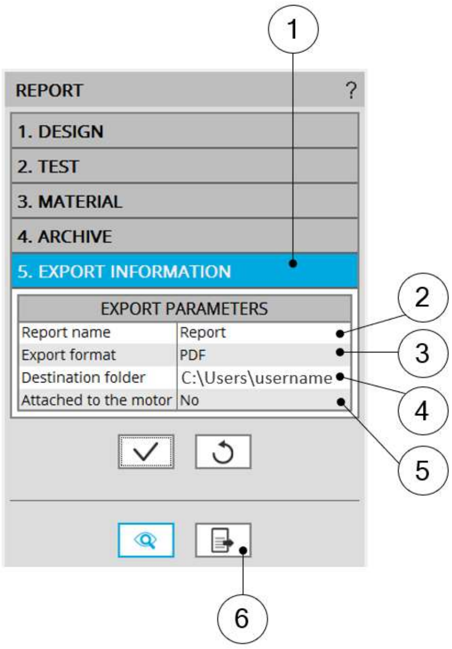
1	Chapters to describe the DESIGN. Machine, Rotor and Stator characteristics.
2	Chapters to describe the TEST results. (TEST will be available in the next versions) All the test results are available as soon as the corresponding computations are performed.
3	List of materials used to build the machine can be added to the report with all their physical properties.
4	Archive groups all the tests which have been saved during the process. These can be added to the report. Note: A maximum of five results per test can be added to the report.
5	Button to apply the selection of the user input selections (selection of chapters)
6	Button to restore default values.

2.4.2 Selection of sections



Motor Factory - EXPORT AREA – Export a report – Chapters to be selected	
1	Section names are shortcuts for displaying the corresponding section of the report: DESIGN, TEST, MATERIAL, ARCHIVE, EXPORT INFORMATION. Note: TEST and ARCHIVE will be available later when one will be able to perform tests dedicated to Wound field synchronous machines in Motor Factory.
2	Check the section to add chapters to the report
3	Button “Preview” considers the selected chapters and displays the report

2.5 Export information



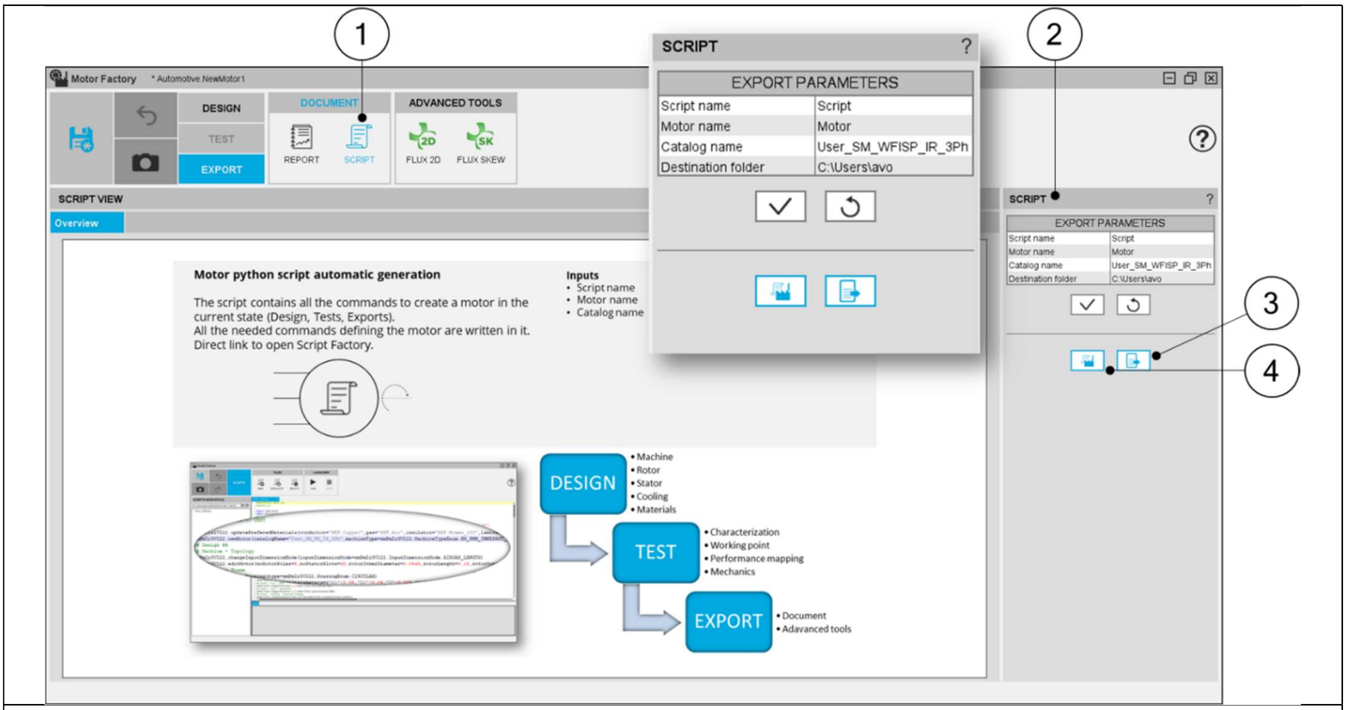
Motor Factory - EXPORT AREA – Export a report – Export information	
1	Access to the area in which export parameters can be defined
2	A file name must be written (Default name = “Report”)
3	The file format must be chosen (pdf or html) to build the report
4	A folder in which storing the report must be selected via the browser
5	It is possible to attach the report (HTML or PDF file) to the motor in the “Motor Catalog”
6	Button to export the report by considering all the previous defined parameters

3 EXPORT A SCRIPT

3.1 Overview

Next to the function “Report”, the function “Script” gives the capability to build and export a python script file, in which all the needed command lines are written to rebuild the considered motor. The script is generated with all the needed sections and sub-sections in Motor Factory, dedicated to the design, the test, and the exports.
Then Script Factory can be used to automate some study such like running serial tests or serial design configurations.

3.2 Area to build the script export.



Motor Factory – EXPORT AREA – Export a python script

1	In Motor Factory select EXPORT / SCRIPT environment.
2	The EXPORT / SCRIPT environment user inputs allow to define the name and the location of the new script file + the name of the motor to be rebuilt from the new python script file, with the catalog name in which it will be stored. Note: Without defining other names for the motor and/or the catalog, the original motor would be overwritten while running the new python script file.
3	Button to build and to export the resulting python file.
4	Button to build and export the resulting python file, and then to open it directly in the Script Factory

4 BUILD AND EXPORT A MODEL IN ALTAIR® FLUX® 2D ENVIRONMENT

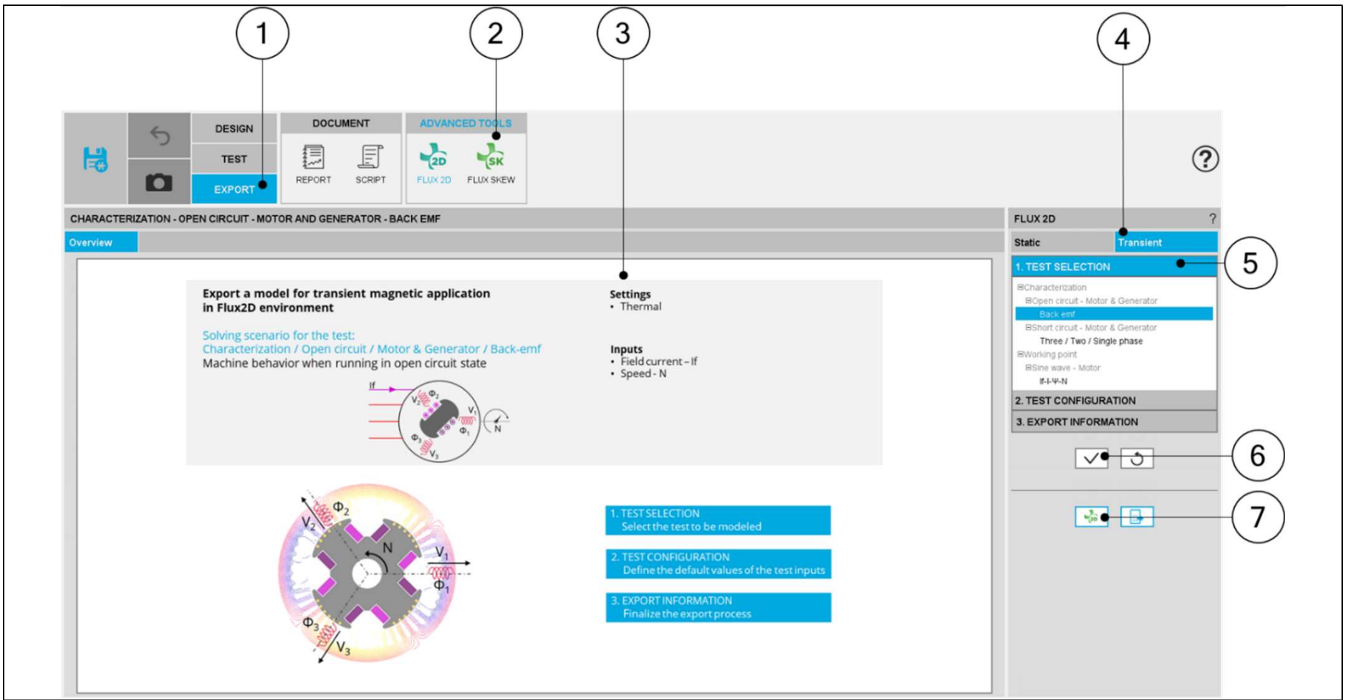
4.1 Overview

The aim of this export is to provide a python file which allows to get a full parametrized model ready to be used in Altair® Flux® 2D environment.
In the current version, models can be exported for static application or transient application in Altair® Flux® 2D environment.

Four models can be exported to Flux® 2D environment:

Application	Model family	Package	Convention	Model / Test
STATIC	Without solving scenario	Current source	Motor & Generator	Basic model
TRANSIENT	Characterization	Open circuit	Motor & Generator	Back emf
	Characterization	Short circuit	Motor & Generator	Three / Two / Single Phase
	Working point	Sine wave	Motor	If-I-Ψ-N

4.2 Area to build and to export a model to Flux® 2D environment.



Motor Factory – EXPORT AREA – Export model for Flux® 2D environment

1	Selection of the EXPORT area of Motor Factory.
2	Access the area in which a model for Flux® 2D environment can be made
3	Zone to visualize the overview of the selected model to be exported
4	Click on the tab to select the application (STATIC or TRANSIENT)
5	3 steps to build the model to be exported for Flux® 2D environment
6	Button to validate inputs before building the model in Flux® 2D environment.
7	Exports the python file for building the model in Flux® 2D environment or launch the project directly in Flux® 2D.

4.3 Steps to build and export a model to Flux® 2D environment.

In EXPORT / ADVANCED TOOLS / FLUX2D area, one must indicate that on which application of Flux® 2D environment, the models must be built: static application or transient application.

Then, the 3 next steps are:

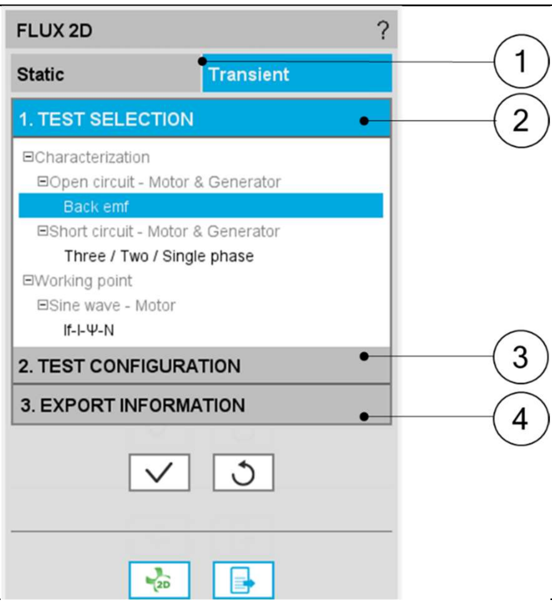
1) Define the type of scenario one wants to get in Flux® 2D environment (Test selection).
   This means the simulation, that one wants to perform in Flux® 2D environment for evaluating the electromagnetic behavior of the considered machine.

2) Define the test configuration. This is to give an initial value for the user inputs, which will be set in the scenario of the simulation available in the Flux® 2D model.

Note: For each Flux® 2D model available in the current version, a short description of the user inputs is done in the following sections.

3) Define the export information

The resulting models are fully parameterized, and these are built in Flux® 2D environment for static or transient applications.

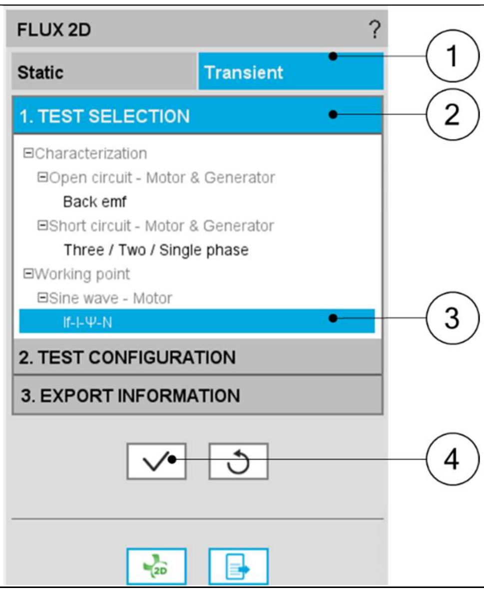


Motor Factory – EXPORT AREA – Export model for Flux® 2D environment

1	Select application (STATIC or TRANSIENT) in which the model must be built in Flux® 2D
2	Choose one scenario (or test) to be provided
3	Define the initial conditions for the simulation process in Flux® 2D environment
4	Define export information

4.4 Test selection

After selecting an application type (STATIC or TRANSIENT), the corresponding test inputs (settings and user inputs) must be defined. This allows to define the initial conditions for testing.



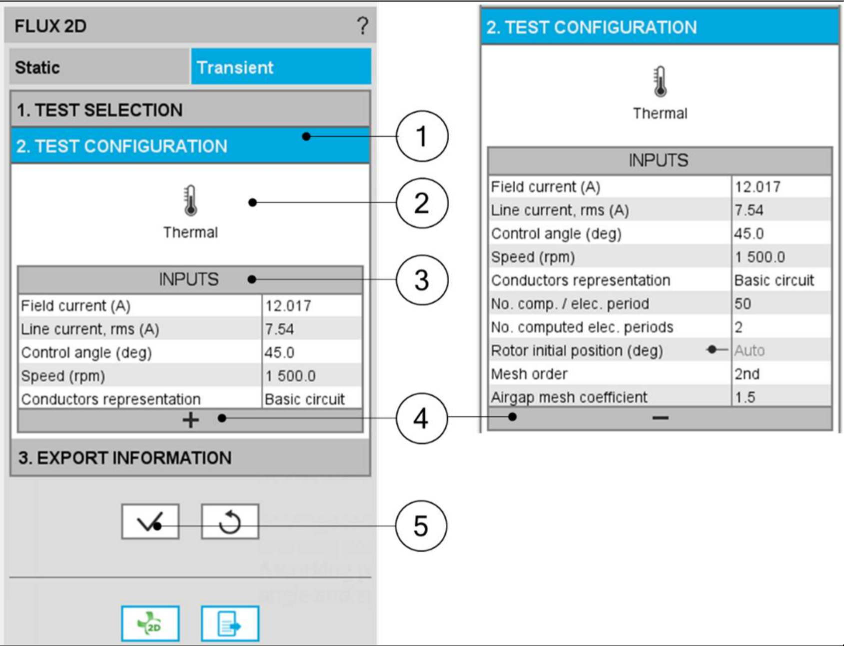
Motor Factory – EXPORT AREA – Export a model for Flux® 2D – Test selection

1	Selection of application (STATIC or TRANSIENT) in which the model must be built for Flux® 2D
2	Tab to choose one scenario (or test) to be provided
3	Selection of the scenario (or test) to be provided
4	Button to validate the previous choices

Note: The user help information about the test parameters is defined in the user help guide of the corresponding test. Please refer to the corresponding section.

4.5 Test configuration

After selecting an application type (STATIC or TRANSIENT), the corresponding scenario (or test) inputs (settings and user inputs) must be defined. This allows to define the initial conditions for the simulation process in Flux® 2D environment.



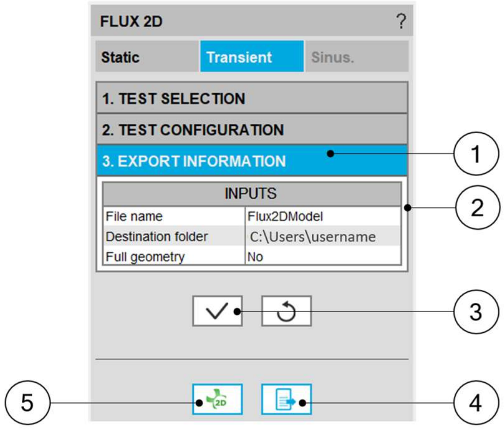
Motor Factory – EXPORT AREA – Export a model for Flux® 2D

1	Tab to define the initial conditions for the simulation process in Flux® 2D environment
2	Settings like thermal conditions can be defined
3	User inputs dealing with the considered test can be defined
4	The tab corresponding to advanced parameters can be expanded. Advanced parameters can also be defined if needed.
5	Button to validate the previous choices

4.6 Export information

The last step for building a model for Flux® 2D is to define the export information.
There are three data to be defined:

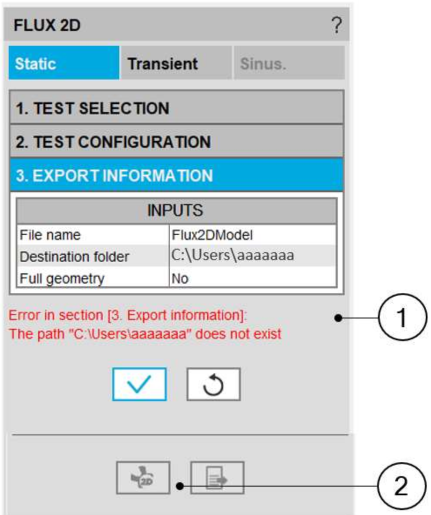
- The name of the python file which will build the model in Flux® 2D environment.
- The folder in which the provided file must be stored.
- The last answer “Full geometry “allows the user to get a full geometry in the provided model, even if it is possible to work with a reduced model considering the number of poles and the number of slots.



Motor Factory – EXPORT AREA – Export model for Flux® 2D – Export information

1	Tab to be expanded to define export information for Flux® 2D
2	Area in which the export parameters to be defined are listed
3	Button to validate the previous choices
4	Button to finalize the export of the model. When one clicks on this button, the folder gets opened where the python file to build the model is stored.
5	Button to finalize the export of the model. This button launches Flux® 2D and builds the model.

Note 1: When data is missing in third table; “Export information” for instance, an error message is displayed in the red colored font which indicates what data is missing. If all the needed information is missing, exporting a model is not allowed.



Motor Factory – EXPORT AREA – Export model for Flux® 2D – Error message

1	Error message displayed in the red colored font
2	The button to export the model is not active if all the needed data are missing

Note: Exporting a model to Flux® 2D (i.e. provide the python file to build the model) can take a few seconds. This is since parameters like initial position of the rotor must be computed first by using internal processes, and then the simulation scenario must be considered.

4.7 Available models to be exported and user inputs.

4.7.1 Overview

All the models to be exported are first classified by considering the type of application for which they are built (STATIC or TRANSIENT). Then, for the tests in Motor Factory Test environment, the models are associated with a convention of operating (Motor or Generator) and grouped into packages itself to get classified into model families.

In the current version of FluxMotor® four models can be exported to Flux® 2D environment:

Application	**Model family**	**Package**	**Convention**	**Model / Test**
STATIC	Without solving scenario	Current source	Motor & Generator	Basic model
TRANSIENT	Characterization	Open circuit	Motor & Generator	Back emf
	Characterization	Short circuit	Motor & Generator	Three / Two / Single Phase
	Working point	Sine wave	Motor	If-I-Ψ-N

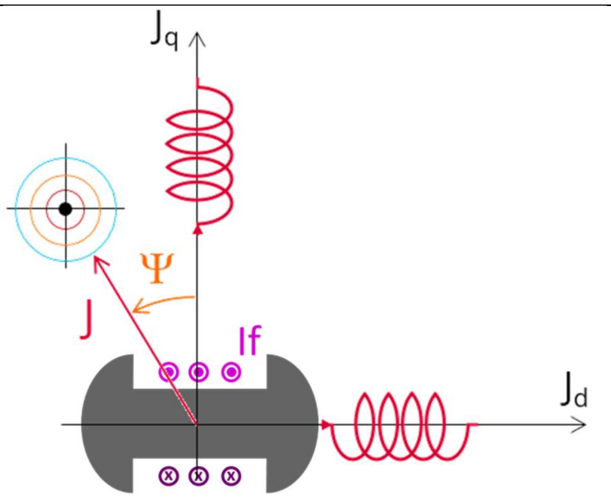
The following section gives a short description of all the models available for exportation to Altair® Flux® 2D environment.

4.7.2 Without scenario – Current source – Motor and generator – Basic model

4.7.2.1 Positioning and objective

This export allows the users to build a model in Flux® 2D, static application to perform magneto-static and multi-static simulations. User inputs like, field current, line current, and control angle are predefined to get quick access into Flux® 2D environment for performing computations.

The resulting model is fully parameterized, and it is built in Flux® 2D environment, static application.



Basic Model - Illustration

The following section describes all the user inputs to initialize the exported model.
All these parameters can be modified in Flux® 2D environment, if needed.

4.7.2.2 Standard inputs

1) Line current, rms

The line current supplied to the machine: “**Line current, rms”** (*Line current, rms value*) must be provided.

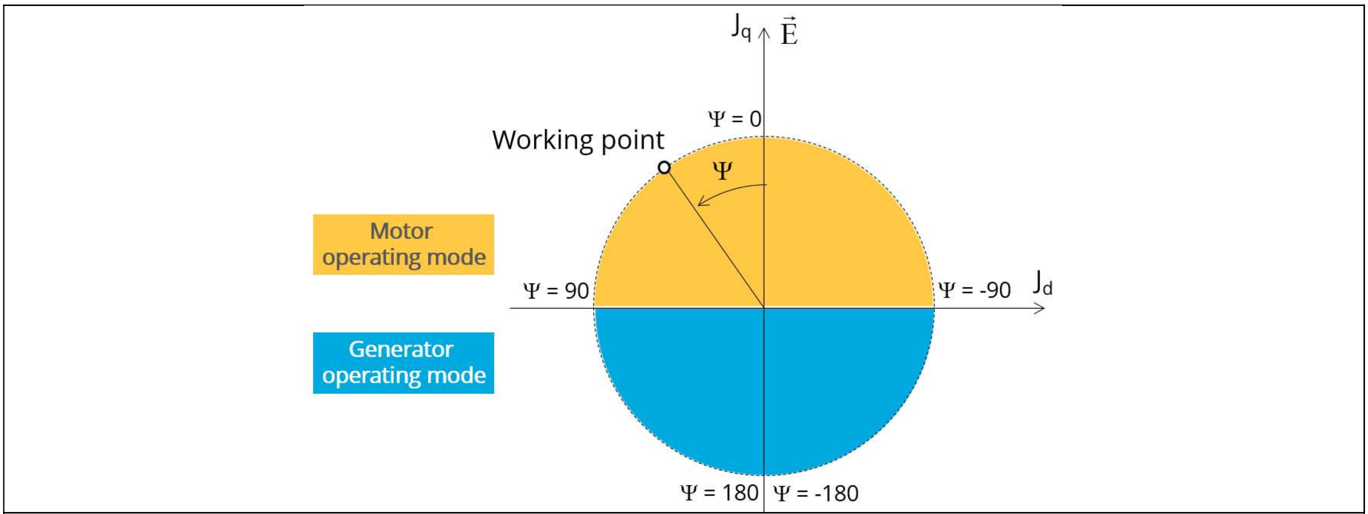
2) Field current

The current supplied to the field winding of the machine: “**Field current”** must be provided.

3) Control angle

Considering the vector diagram shown below, the “**Control angle**” is the angle between the electromotive force E and the electrical current (J) (Y = (E, J)).

It is an electrical angle. The default value is 45 degrees. It must be set in a range of -90 to 90 degrees.
This range of values covers all the possible working point in motor convention.



Definition of the control angle Ψ - Motor convention

4.7.2.3 Advanced inputs

The list of advanced inputs dedicated to this export are listed below.
For more details, please refer to the section 4.7.6 - List of generic advanced inputs.

1) Rotor initial position mode

By default, the “**Rotor initial position mode**” is set to “**Auto**”.

2) Rotor initial position

3) Mesh order

The default level is second order mesh.

4) Airgap mesh coefficient

Airgap mesh coefficient is set to 1.5 by default.

4.7.3 Characterization – Open circuit – Motor & Generator – Back – emf

4.7.3.1 Positioning and objective

The aim of the test “**Characterization - Open circuit – Motor & Generator - Back-EMF**” is to characterize the behavior of the machine, when running in open circuit state.
The resulting model is fully parameterized, and it is built in Flux® 2D environment, transient application.

The following section describes all the user inputs to initialize the exported model.
All these parameters can be modified in Flux® 2D environment, if needed.

4.7.3.2 Settings

One button gives access to the following setting definition:

- Temperature of active components: winding, damper bars, and end rings

For more details, refer to the section dealing with the test settings.

4.7.3.3 Standard inputs

1) Field current

The current supplied to the field winding of the machine: “**Field current”** must be provided.

2) Speed

The operated speed of the machine to be used in the back-EMF test.

4.7.3.4 Advanced inputs

The list of advanced inputs dedicated to this export are listed below.
For more details, please refer to the section 4.7.6 - List of generic advanced inputs.

1) Number of computations per electrical period

The default value is equal to 50. The minimum allowed value is 13.

2) Number of computed electrical periods

The default value is equal to 2. The minimum allowed value is 1 and the maximum value is equal to 10.

3) Rotor initial position

By default, the “**Rotor initial position**” is set to “**Auto**”.

4) Mesh order

The default level is second order mesh.

5) Airgap mesh coefficient

Airgap mesh coefficient is set to 1.5 by default.

4.7.4 Characterization – Short circuit – Motor & Generator – Three / Two / Single Phase

4.7.4.1 Positioning and objective

The aim of the export "**Characterization - Short circuit – Motor & Generator – Three / Two / Single Phase**" is to characterize the behavior of the machine, when running in short circuit state happening between three phases, two phases or a single phase.
The resulting model is fully parameterized, and it is built in Flux® 2D environment, transient application.

The following section describes all the user inputs to initialize the exported model.
All these parameters can be modified in Flux® 2D environment, if needed.

4.7.4.2 Settings

One button gives access to the following setting definition:

- Temperature of active components: winding, damper bars, and end rings

For more details, refer to the section dealing with the test settings.

4.7.4.3 Standard inputs

1) Type of short circuit

The type of short circuit can be chosen among a list of choice:
- Three-phase: the three phases of the stator windings are short circuited.
- Two-phase: the first phase and second phase of the stator windings are short circuited.
- Single-phase: the first phase of the stator windings is short circuited.

2) Field voltage

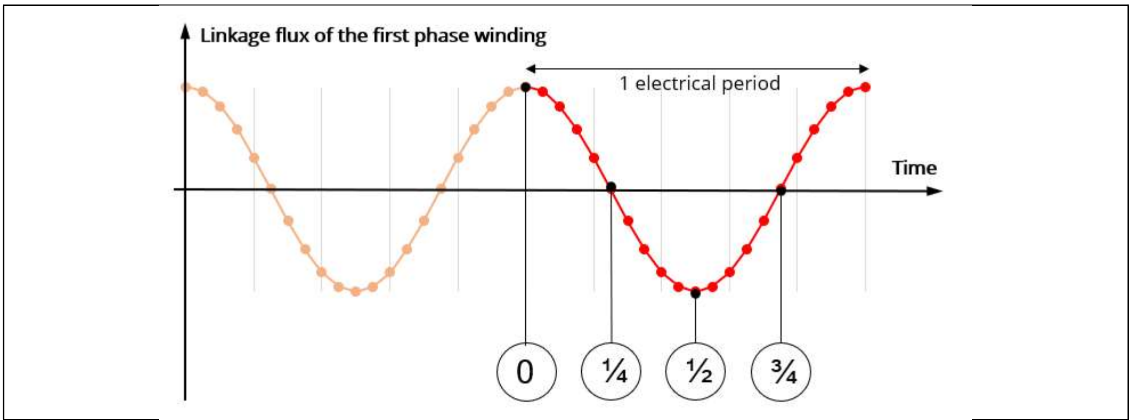
The voltage supplied to the field winding of the machine: "**Field voltage"** must be provided.

3) Speed

The operated speed of the machine to be used in the short circuit test.

4) Short circuit instant

The short circuit instant can cause different behavior of the short circuit transient curve. It can be chosen among a list of choice:
- Instant 0: the linkage flux of the first phase winding is supposed to be maximum.
- Instant 1/4: the linkage flux of the first phase winding is supposed to be zero.
- Instant 1/2: the linkage flux of the first phase winding is supposed to be minus maximum.
- Instant 3/4: the linkage flux of the first phase winding is supposed to be zero.



Short circuit instant illustration

4.7.4.4 Advanced inputs

The list of advanced inputs dedicated to this export are listed below.
For more details, please refer to the section 4.7.6 - List of generic advanced inputs.

1) Number of computations per electrical period

The default value is equal to 50. The minimum allowed value is 13.

2) Number of electric periods in open circuit

The default value is equal to 1. The minimum allowed value is 0.

3) Number of electric periods in short circuit

The default value is equal to 2. The minimum allowed value is 1.

4) Rotor initial position

By default, the "**Rotor initial position**" is set to "**Auto**".

5) Mesh order

The default level is second order mesh.

6) Airgap mesh coefficient

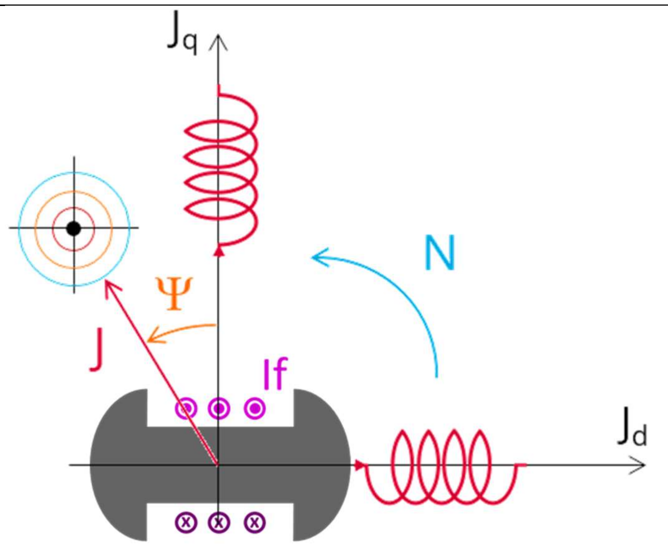
Airgap mesh coefficient is set to 1.5 by default.

4.7.5 Working point – Sine wave – Motor – If, I, Ψ, N

4.7.5.1 Positioning and objective

The aim of the test "**Working point – Sine wave – Motor – If, I, Ψ, N"** is to characterize the behavior of the machine when operating at the targeted input values If, I, Ψ, N (Magnitude of current, Magnitude of field current, Control angle, Speed).
Hence, these three inputs are enough to impose a precise working point.

The resulting model is fully parameterized, and it is built in Flux® 2D environment, transient application.



"Working point – Sine wave – Motor – If, I, Ψ, N" illustration

The results of this test give an overview of the electromagnetic analysis of the machine considering its topology.
It also gives the capability to make comparisons between results obtained from the measurements and those with the FluxMotor®.

The following section describes all the user inputs to initialize the exported model.
All these parameters can be modified in Flux® 2D environment, if needed.

4.7.5.2 Settings

One button gives access to the following setting definition:

- Temperature of active components: winding, damper bars, and end rings

For more details, refer to the section dealing with the test settings.

4.7.5.3 Standard inputs

1) Field current

The current supplied to the field winding of the machine: “**Field current”** must be provided.

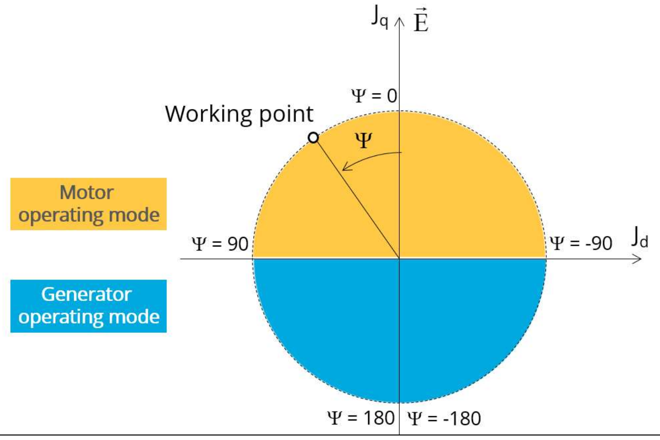
2) Line current, rms

The line current supplied to the machine: “**Line current, rms”** (*Line current, rms value*) must be provided.

7) Control angle

Considering the vector diagram shown below, the “**Control angle**” is the angle between the electromotive force E and the electrical current (J) (Ψ = (E, J)).

It is an electrical angle. The default value is 45 degrees. It must be set in a range of -90 to 90 degrees.
This range of values covers all the possible working point in motor convention.



Definition of the control angle Ψ - Motor convention

8) Speed

The imposed “**Speed**” (*Speed*) of the machine must be set.

9) Represented coil conductors.

In transient application, it is possible to export a project into Flux® environment, where the elementary wires will be modeled with solid conductors. The geometry, the meshing and the corresponding electric circuit will be defined to well represent these.

Three choices are possible:

- “No”: The coils will be represented with face regions. The elementary wires won’t be represented in the Finite Element model (Flux®).
- “One phase”: The elementary wires will be represented for only one phase. This will allow to compute AC losses for conductors into the first phase. This choice allows to get a good ratio between the quality of results and computation time.
- “All phases”: The elementary wires will be represented into all the phases.

4.7.5.4 Advanced inputs

The list of advanced inputs dedicated to this export are listed below.
For more details, please refer to the section 4.7.6 - List of generic advanced inputs.

1) Number of computations per electrical period

The default value is equal to 50. The minimum allowed value is 13.

10) Number of computed electrical periods

The default value is equal to 2. The minimum allowed value is 1 and the maximum value is equal to 10.

11) Rotor initial position

By default, the "**Rotor initial position**" is set to "**Auto**".

12) Mesh order

The default level is second order mesh.

13) Airgap mesh coefficient

Airgap mesh coefficient is set to 1.5 by default.

4.7.6 List of generic advanced inputs

1) Number of computations per electrical period

The number of computations per electrical period "**No. comp. / elec. period**" (*Number of computations per electrical period*) influences the accuracy of results and the computation time.
The default value is 50. The minimum allowed value is 13. This default value provides a good balance between the accuracy of results and computation time.

14) Number of computed electrical periods

The default value is 2. The minimum allowed value is 1 and the maximum value is equal to 10.

15) Rotor initial position

By default, the "**Rotor initial position**" is set to "**Auto**".

When the "**Rotor initial position mode**" is set to "**Auto**", the initial position of the rotor is automatically defined by an internal process. The resulting relative angular position corresponds to the alignment between the axis of the stator phase 1 (reference phase) and the direct axis of the rotor north pole.

When the "**Rotor initial position**" is set to "User input" (i.e. toggle button on the right), the initial position of the rotor considered for computation must be set by the user in the field « **Rotor initial position** ». The default value is equal to 0. The range of possible values is [-360, 360].
For more details, please refer to the document: MotorFactory_SMPM_IOR_3PH_Test_Introduction – section "Rotor and stator relative position".

16) Mesh order

To get the results, the computation is performed using a Finite Element Modeling. The geometry of the machine is meshed.
Two levels of meshing can be considered for this finite element calculation: first order and second order.
This parameter influences the accuracy of results and the computation time.

By default, second order mesh is used.

17) Airgap mesh coefficient

The advanced user input "**Airgap mesh coefficient**" is a coefficient which adjusts the size of mesh elements inside the airgap. When one decreases the value of "**Airgap mesh coefficient**", the size of the mesh elements reduces, thus increasing the mesh density inside the airgap and the accuracy of results.

The imposed Mesh Point (size of mesh elements touching points of the geometry) is described as:

Mesh Point = (airgap) x (airgap mesh coefficient)

Airgap mesh coefficient is set to 1.5 by default.
The variation range of values for this parameter is [0.05; 2].
0.05 gives a very high mesh density, and 2 gives a very coarse mesh density.

Caution:
Be aware, a very high mesh density does not always mean a better result quality. However, this always leads to a huge number of nodes in the corresponding finite element model. So, it means the need of huge numerical memory, which increase the respective computation time considerably.

5 BUILD AND EXPORT A MODEL IN ALTAIR® FLUX® SKEW ENVIRONMENT

5.1 Overview

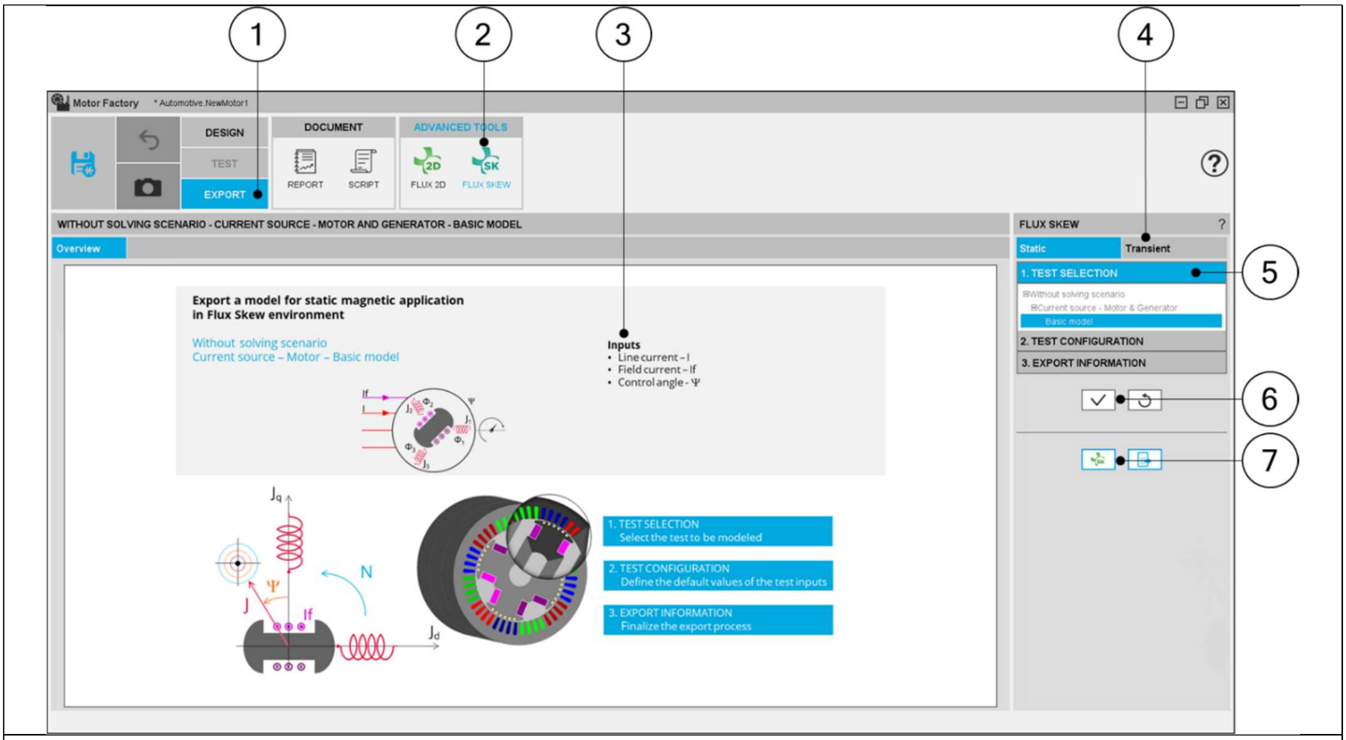
The aim of this export is to provide a python file which allows to get a full parametrized model ready to be used in Flux® SKEW environment.
All the models to be exported are first classified by considering the type of application for which they are built (STATIC or TRANSIENT).
Then, for the tests in Motor Factory Test environment, the models are associated with a convention of operating (Motor or Generator) and grouped into packages itself to get classified into model families.

In the current version of FluxMotor® three models can be exported to Flux® SKEW environment:

Application	Model family	Package	Convention	Model / Test
STATIC	Without solving scenario	Current source	Motor & Generator	Basic model
TRANSIENT	Characterization	Open circuit	Motor & Generator	Back emf
	Characterization	Short circuit	Motor & Generator	Three / Two / Single Phase
	Working point	Sine wave	Motor	If-I-Ψ-N

The following section gives a short description of all the models available for exportation to Flux® 2D environment.

5.2 Area to build and to export a model to Flux® SKEW environment.



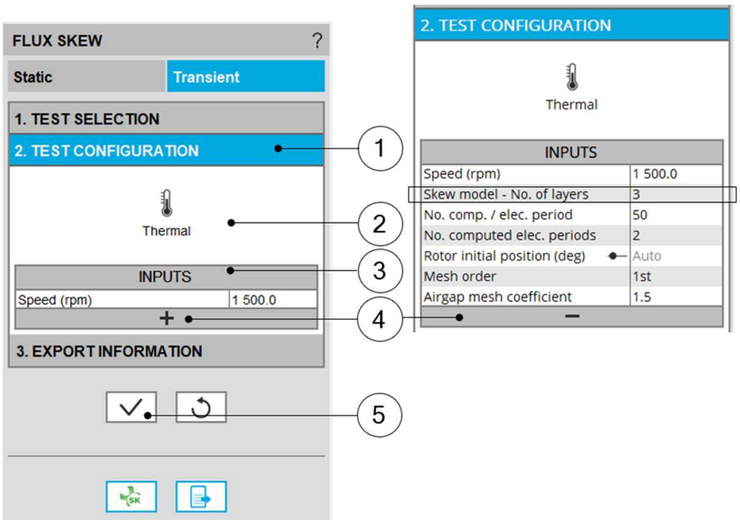
Motor Factory – EXPORT AREA – Export model for Flux® Skew environment

1	Selection of the EXPORT area of Motor Factory.
2	Access to the area in which a model for Flux® Skew environment can be made
3	Zone to visualize the overview of the selected model to be exported
4	Click on the tab to select the application (STATIC or TRANSIENT)
5	3 steps to build the model to be exported for Flux® Skew environment
6	Button to validate inputs and export the python file for building the model in Flux® Skew environment.
7	Buttons to export the python file for building the model in Flux® Skew environment or launch directly Flux® Skew.

5.3 Particularities in building and to exporting a model to Flux® Skew environment.

A user who wants to build and export a model to Flux® SKEW has to follow the same steps and recommendations as with the function "FLUX 2D".

The main particularity of function "FLUX SKEW" is that the "**Skew number of layers**" is an input, that must be defined. Its default value is 3.
Even if the design of the machine is defined with "continuous skew", the "**Skew number of layers**" is necessary for Flux® to define the finite elements model in the FLUX SKEW environment. A high number of layers gives more accurate finite elements computations. However, it needs higher computation time. For that purpose, the value 3 is a good compromise between accuracy and speed.



Motor Factory – EXPORT AREA – Export a model for Flux® SKEW

1	Tab to define the initial conditions for the simulation process in Flux® SKEW environment
2	Settings like thermal and mechanical conditions can be defined
3	User inputs dealing with the considered test can be defined
4	The tab corresponding to advanced parameters can be expanded. Advanced parameters can also be defined, if needed.
5	Button to validate the previous choices

6 EXPORT TO SYSTEM

6.1 Overview

The area SYSTEM, in the EXPORT environment of Motor Factory, allows exporting data like constants, curves and maps in lookup table (LUT) formats, such as FMU and MAT format files.

In the current version, the test Characterization/Model/Maps can be selected for exporting the data.

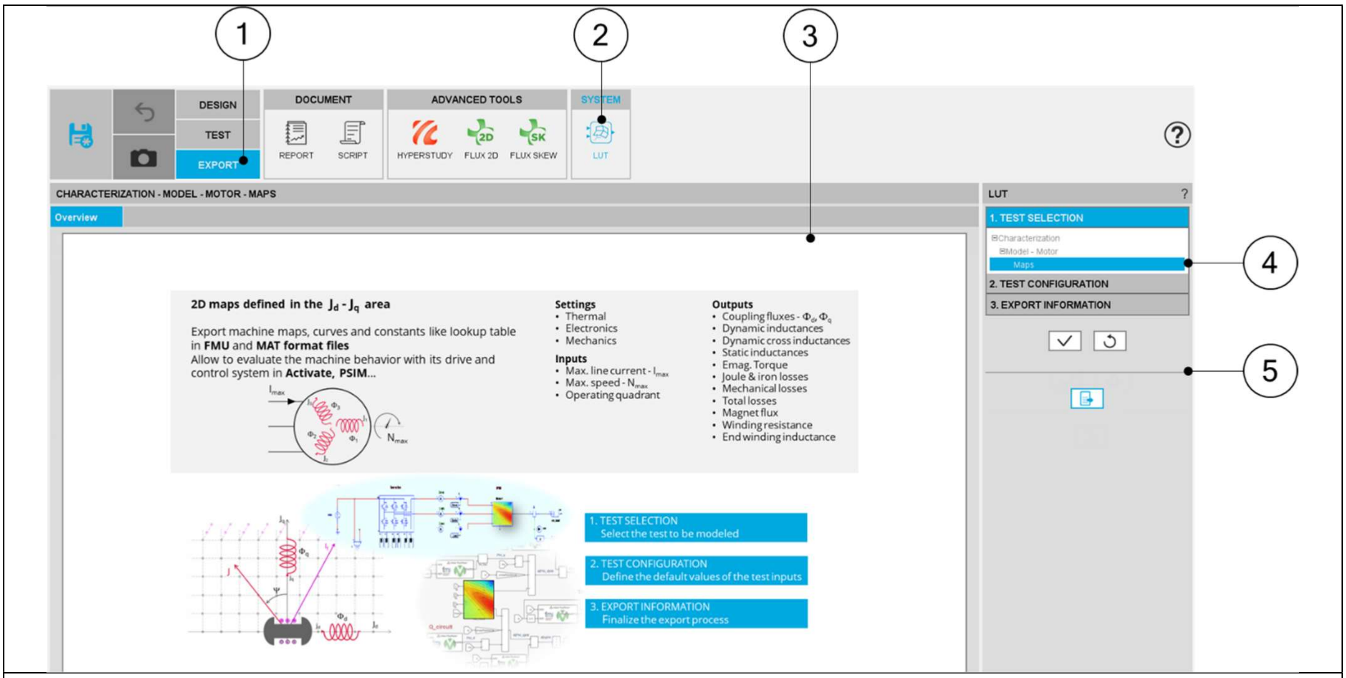
Constants, curves and maps" given in J_d-J_q plane, for characterizing the 3-Phase synchronous machines with permanent magnets are computed and exported.

These files can be imported directly into environments like Altair® Activate®, Altair® Compose® or Altair® PSIM® as binary variables files (.mat) or inside block functions, ready to be integrated into schemes to represent the model of the corresponding rotating electrical machine.

These functionalities are useful to represent the machine at the system level. Therefore, electrical machine and other system components, such as the drive and the control command, can be represented and simulated altogether into the same area.

Note: This functionality is not implemented for polyphase machines. It will be addressed in a future version.

6.2 Area to export LUT



Motor Factory – EXPORT AREA – Export data in LUT formats.

1	Selection of the EXPORT area of Motor Factory.
2	Access the area (SYSTEM) in which data can be exported in lookup table (LUT) formats.
3	Zone to visualize either the overview of the selected test
4	3 steps to compute and to export LUT data
5	Button to validate inputs, display a preview and export the data.

6.3 Steps to build an export LUT

6.3.1 Introduction

In EXPORT / ADVANCED TOOLS / SYSTEM area, 3 steps are needed to build and export data:

1) Select the test which will be performed for building data to be exported.
2) Define the test configuration, that means the user inputs/outputs parameters needed to perform the test (settings and user inputs of the considered test)
3) Define the export type (FMU or MAT formats) and information.

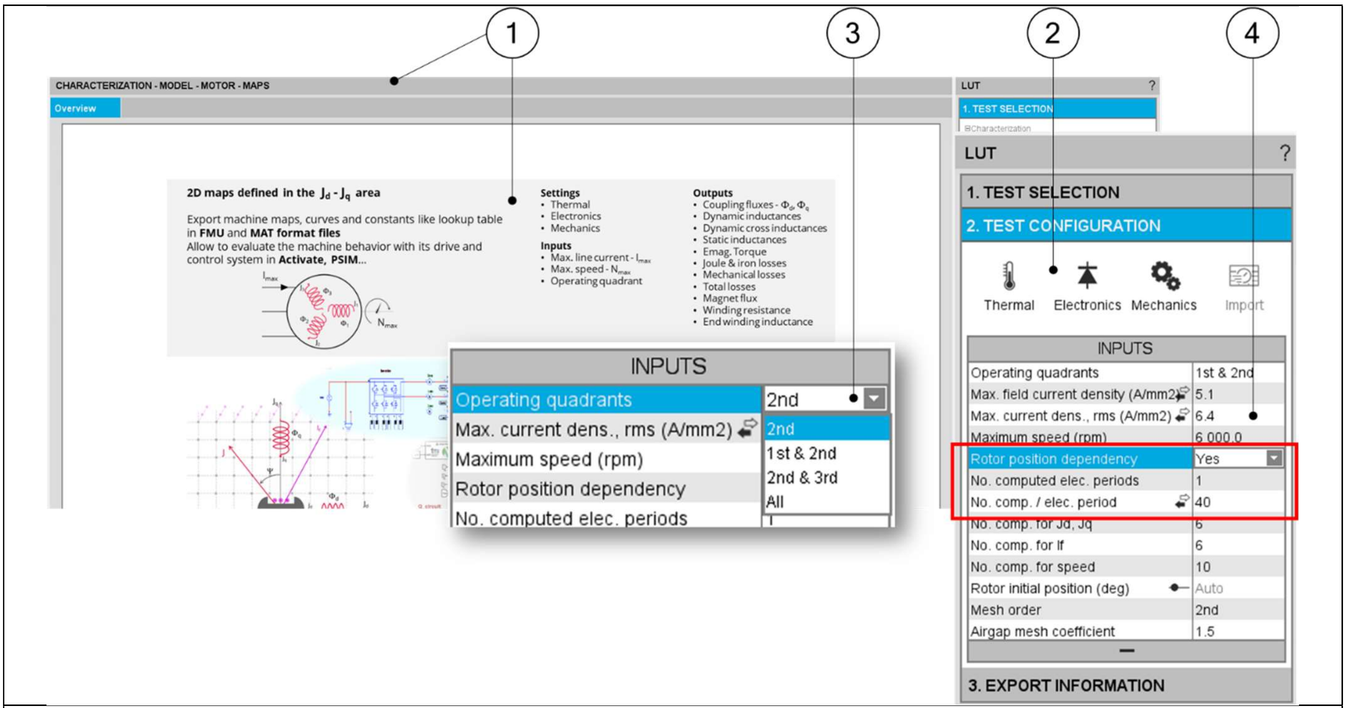
6.3.2 Test selection

In the current version of FluxMotor®, one test can be selected:

- Characterization / Model / Motor / Maps

6.3.3 Test configuration

After selecting a test, the corresponding test inputs (settings and user inputs) must be defined. This allows to define the initial conditions for testing.



Motor Factory – EXPORT AREA – SYSTEM – LUT / Test configuration for Characterization / Model / Motor / Maps

1	Overview of the selected test is displayed
2	User inputs can be defined in the test area.
3	User's inputs to export data based on 1, 2 or 4 quadrants
4	User's inputs to export data with respect to the rotor position dependency.

Note: The user help information about the test parameters is defined in the user help guide of the corresponding test. Please refer to the corresponding section.

Note: Operating quadrants
Export / System LUT (Activate or PSIM) allows exporting data based on 1, 2 or 4 quadrants.
This user's input defines the quadrants in the J_d - J_q plane, where the test will be carried out. By default, the only considered quadrant is the 1st & 2nd one (i.e., the grid is defined for negative and positive values of the current in the d axis and positive ones in the q axis). This corresponds to the motor behavior of the machine.

The other possible values for this input are: “2nd and 3rd “, “1st and 2nd “and “all”. Options allow computing and displaying 1, 2 or 4 quadrants.

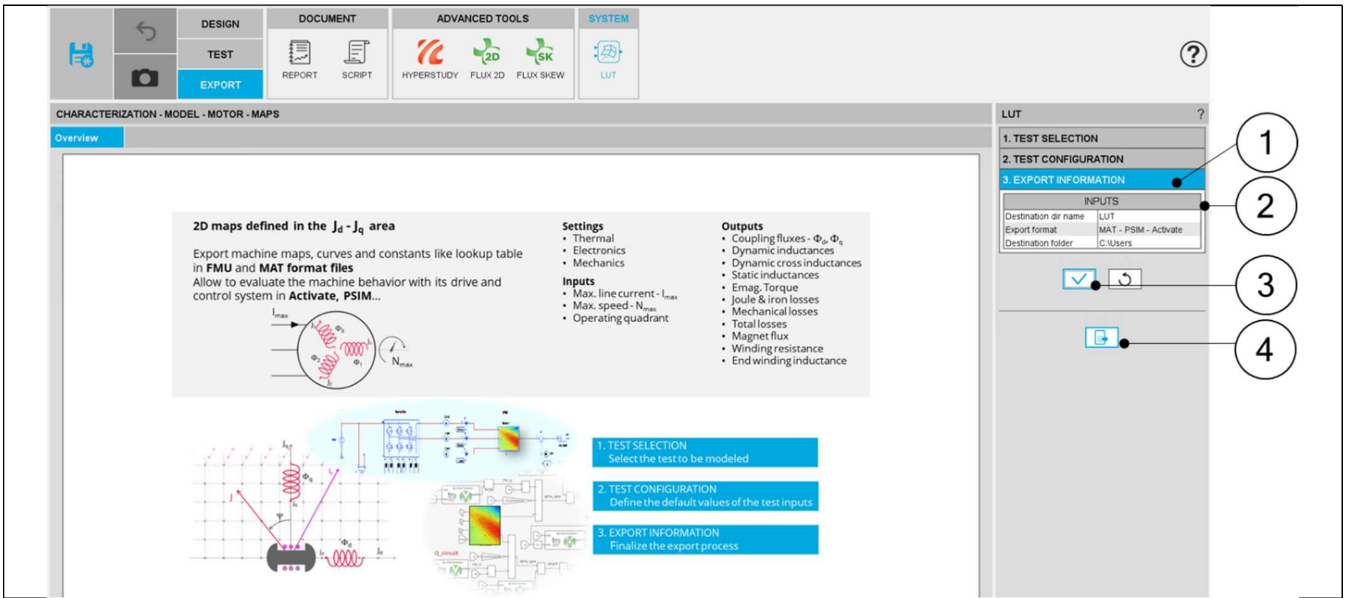
Note: Rotor position dependency
Export / System LUT (Activate or PSIM) allows exporting data with respect to the rotor position dependency.
This user’s input defines the rotor position dependency, where the test will be carried out. By default, the rotor position dependency is set to “No” but it can be set to “Yes”. In this case the computation will be done in the J_d - J_q plane with an additional fourth axis corresponding to the rotor position θ_r.
In case the rotor dependency is set to “Yes”, whatever the operating quadrant choice, the finite element computation is done over all selected quadrants (in case the rotor dependency is set to “No”, symmetries are used).

6.3.4 Export information

The last step for building and exporting data in FMU format files is to define the export information.

Three inputs must be defined:

- The name of the directory in which the created files will be stored
- The format of the file to be exported. Three options are available: FMU for Activate, FMU generic and MAT file - for Activate and PSIM.
- The destination folder in which the previous directory will be located



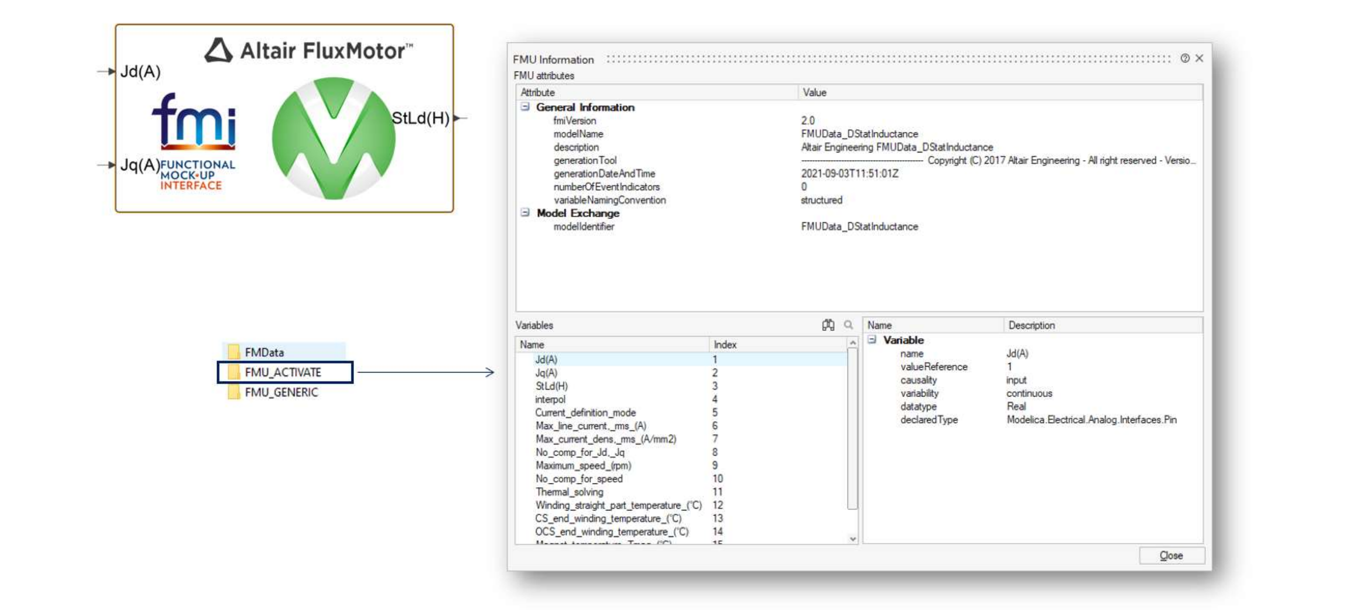
Motor Factory – EXPORT AREA – SYSTEM – LUT / Export information

1	Tab to be expanded to define the export information
2	Area in which the export information of parameters to be defined are listed
3	Button to validate the previous choices
4	Button to finalize the export of the data files. One click opens the folder where the directory is located

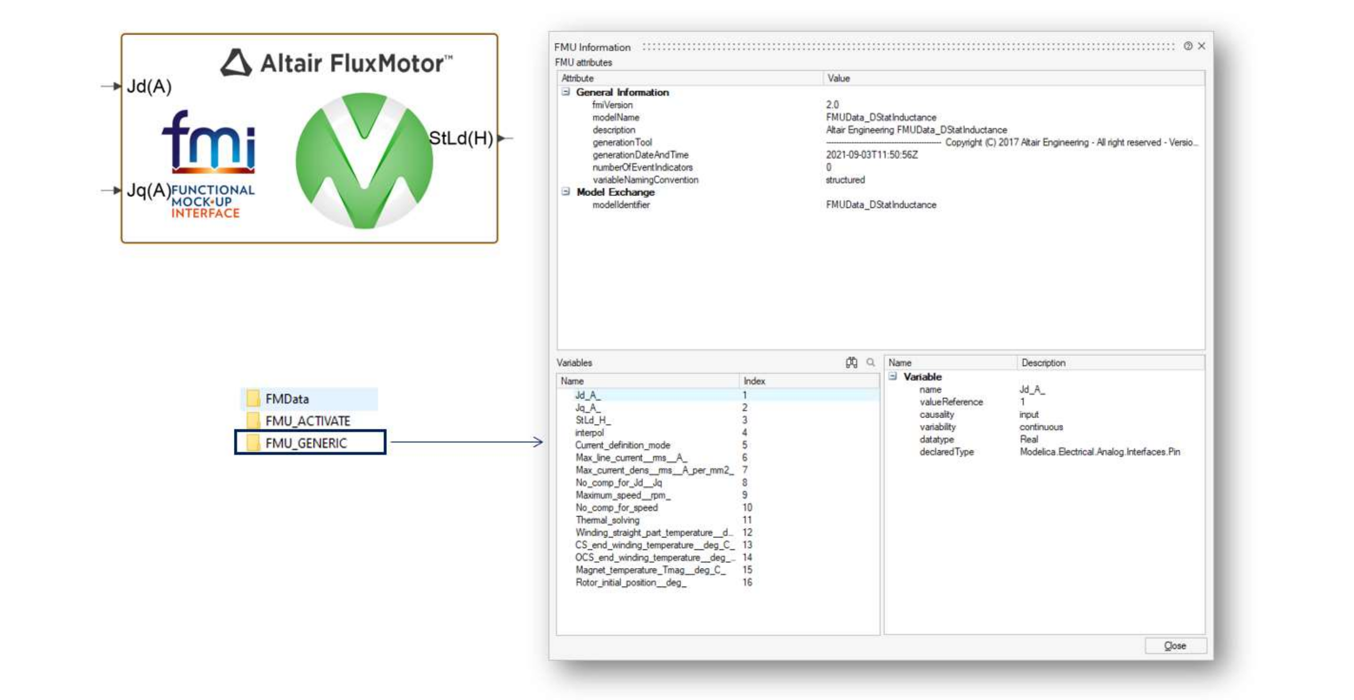
6.4 FMU format files

6.4.1 Compatibility

Two packages of FMU format files are automatically provided, one dedicated to Activate® and another one compatible with other system simulation tools.
Hence, the user will be able to select the required system simulation tool without any problem of compatibility. One of the main differences between the two files is how the units are managed in the name labels (See below illustrations).



Exported FMU format files dedicated to Activate®



Exported FMU format files (Generic) compatible with other system simulation tools

6.4.2 A C/C++ compiler is needed

6.4.2.1 C/C++ compiler / System requirements

FluxMotor® requires a C/C++ compiler to perform some operation for creating FMU blocks.

Here is the list of the Visual Studio compilers supported:

Microsoft® Visual Studio 2019, Community, Professional, Enterprise

Microsoft® Visual Studio 2017, Community, Professional, Enterprise

Microsoft® Visual Studio 2019/2017/2015: Build Tools
Note: the option for **Windows 10 SDK** must be selected

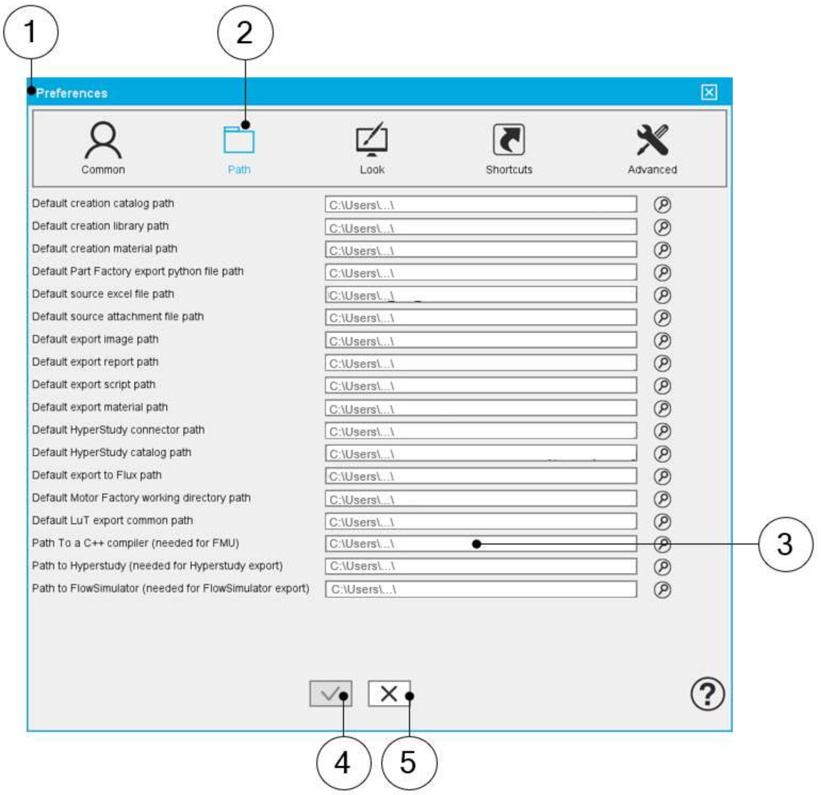
Microsoft® Visual Studio C++ 2015 (VC 14.0 Express, Community and Professional)

Important Remark
The table above proposes different versions of Microsoft Visual Studio. Make sure the version you install is approved by your IT department and you have the right license (e.g. if you decide to use Professional Edition, a license is required)

6.4.2.2 Access path of the C/C++ Compiler

Once the C/C++ Compiler is installed on the computer, its access path must be specified in the user's preferences.

Note: When opening FluxMotor®, if a C/C++ Compiler is already installed on the computer, the corresponding install path is automatically written in the user's preferences.



FluxMotor® / Supervisor / User's preferences / Location of the C/C++ Compiler

1	The Preferences dialog box is opened from the FluxMotor® supervisor.
2	Second tab is Path Preferences.
3	Define the location of the C/C++ Compiler on the computer

Here below is a list of files to select in the installation directory (path) according to the Visual Studio version installed:

Visual Studio 2019, Community	C:\Program Files (x86)\Microsoft Visual Studio\2019\Community\VC\Auxiliary\Build\vcvarsall.bat
Visual Studio 2019, Professional	C:\Program Files (x86)\Microsoft Visual Studio\2019\Professional\VC\Auxiliary\Build\vcvarsall.bat
Visual Studio 2019, Enterprise	C:\Program Files (x86)\Microsoft Visual Studio\2019\Enterprise\VC\Auxiliary\Build\vcvarsall.bat
Visual Studio 2017, Community	C:\Program Files (x86)\Microsoft Visual Studio\2017\Community\VC\Auxiliary\Build\vcvarsall.bat
Visual Studio 2017, Professional	C:\Program Files (x86)\Microsoft Visual Studio\2017\Professional\VC\Auxiliary\Build\vcvarsall.bat
Visual Studio 2017, Enterprise	C:\Program Files (x86)\Microsoft Visual Studio\2017\Enterprise\VC\Auxiliary\Build\vcvarsall.bat
Microsoft® Visual Studio C++ 2015 Express	C:\Program Files (x86)\Microsoft Visual Studio\14.0\VC\Build\vcvarsall.bat
Microsoft® Visual Studio C++ 2015 Community	C:\Program Files (x86)\Microsoft Visual Studio\14.0\VC\Build\vcvarsall.bat
Microsoft® Visual Studio C++ 2015 Professional	C:\Program Files (x86)\Microsoft Visual Studio\14.0\VC\Build\vcvarsall.bat
Microsoft® Visual Studio 2019, Build Tools	C:\Program Files (x86)\Microsoft Visual Studio\2019\BuildTools\VC\Auxiliary\Build\vcvarsall.bat
Microsoft® Visual Studio 2017, Build Tools	C:\Program Files (x86)\Microsoft Visual Studio\2017\BuildTools\VC\Auxiliary\Build\vcvarsall.bat

Note that the executable command is detected if the Visual Studio is already installed before or if the preference is set to an empty value and then reopening the preference.

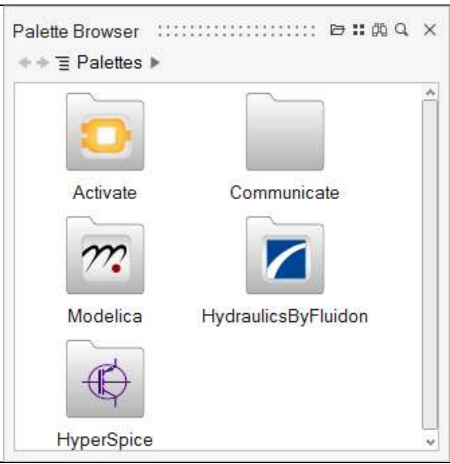
6.4.3 Import FMU data in Altair® Activate®

Once FMU files are generated by FluxMotor®, these can be imported in environments like Activate®.
This section explains how FMU files generated from FluxMotor® are used in Activate®.
The FMU file of the D-axis flux is taken as an example.

First, Activate® is opened.
Either start creating a new project via a new modeling window or open an existing scm file.

To use FMU files from FluxMotor®, locate the FMU block in the palettes of the System library.

1) Select View > Palette Browser



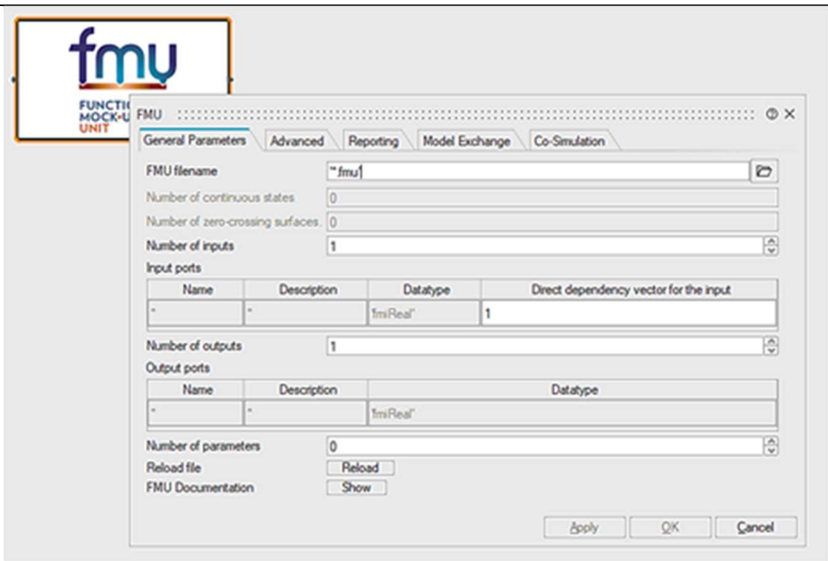
The Palette Browser displays the installed library palettes.

2) Double-click Activate® > CoSimulation.
The Palette Browser displays the blocks available in the CoSimulation palette.

3) Select the FMU block, then drag and drop it into the modeling window.

One can also write down “FMU” in the quick search field.

4) Double-click on the FMU dragged in the modeling window, or right-click, and from the context menu, select Parameters.



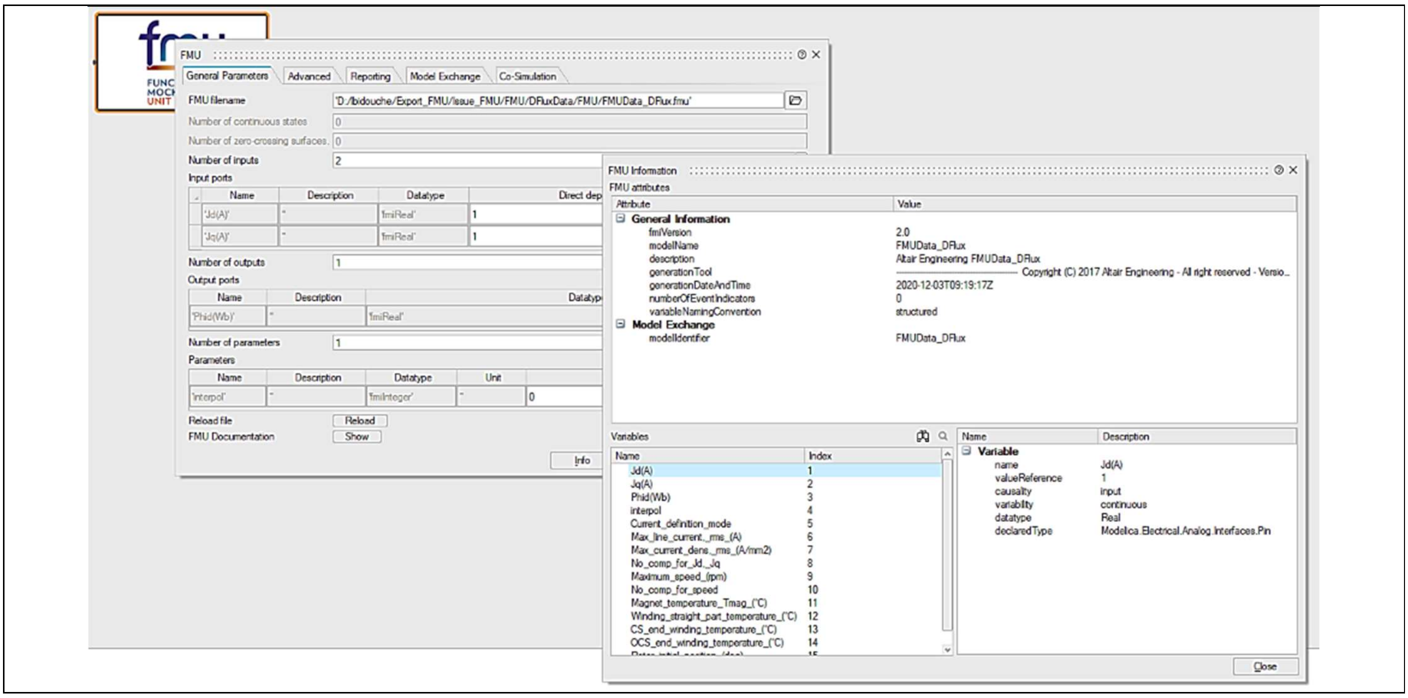
Then:

- In General parameters > FMU filename, indicates the path to the D-axis flux FMU (the directory in which the FMU file is located).
- All the information regarding the D-axis flux FMU appears along with a new content under the name parameters appears.
  In this area, you can set the boundaries of the quadrant by choosing a value from 0 to 3.
  These boundaries reflect the FMU response when the user is outside the quadrant in which the calculations were made.

  The meaning of each value is listed in the table that follows:

Value	Meaning
0	NAN
1	Zero
2	Hold
3	Linear extrapolation

All the information related to the resolution of the test map can be seen by clicking on the info tab.



The FMU generated will have its inputs and outputs. The D-axis flux FMU in Activate® will look like this:

Along with FMU files, an oml file, that contains important constant values of the test map is generated. These values can be loaded and used in the Activate® model by executing the oml file.
This oml file could be read in Activate® diagram home by indicating its path, and using the function execution as follow:

run('D:\UserFolder\Export_FMU\FMU_AD\oml\constants.oml')

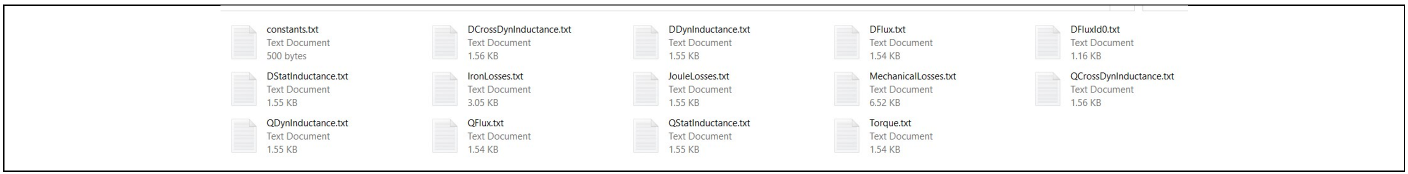
6.5 MAT format files

6.5.1 Introduction

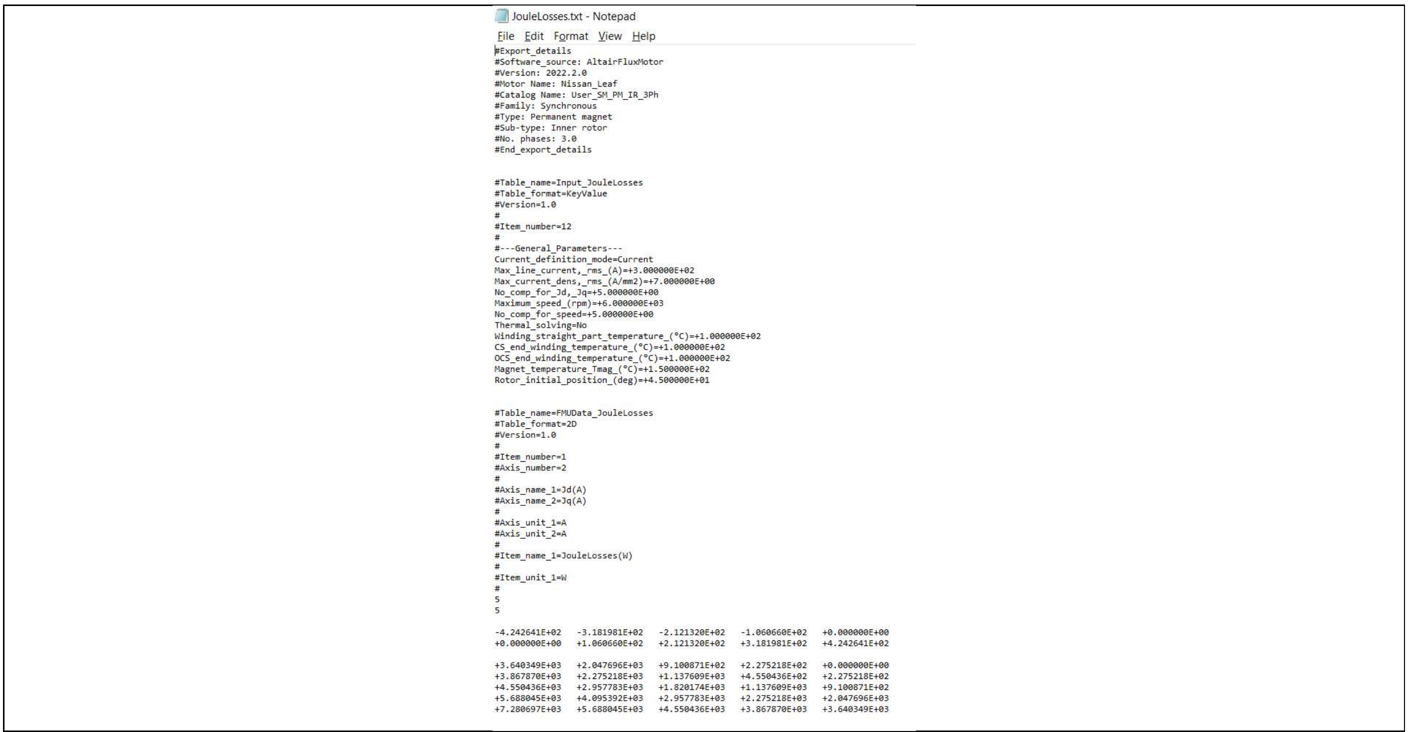
Once MAT files are generated by FluxMotor®, they can be imported in environments like PSIM® Activate® or Compose®.
This section explains the folder structure and the files generated by the export format "MAT – PSIM – Activate".

The export generates a folder named according to the user choice. Inside it, two different folders can be found:

- A folder "FMData", containing several ".txt" files. Each file stocks the information about one exported variable (e.g., fluxes or inductances):



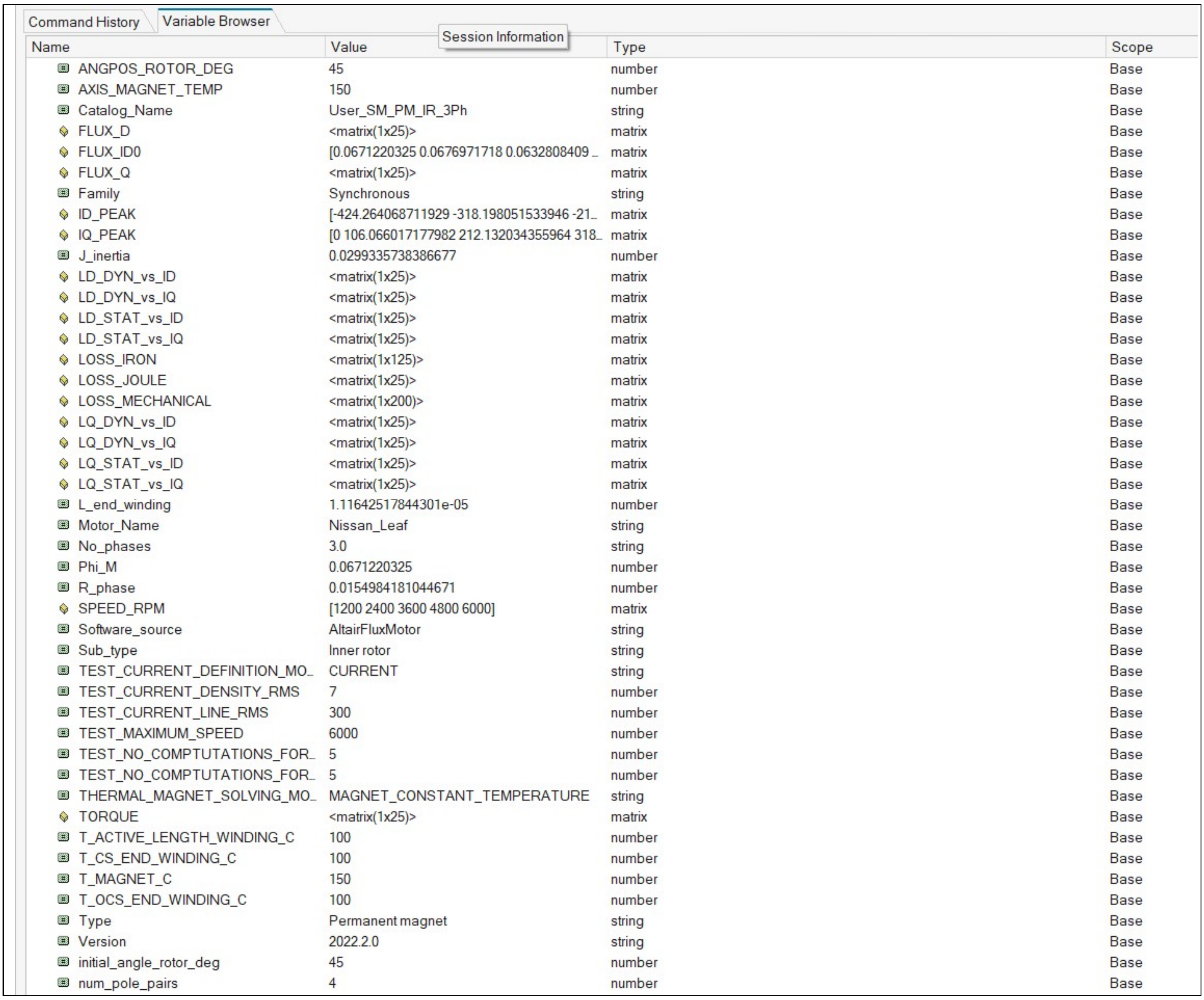
These files have the same format as the results exported from the FluxMotor test Characterization / Model. An example of the file containing the Joule losses is shown below.


JouleLosses.txt - Notepad

File Edit Format View Help

```
#Export_details
#Software_source: AltairFluxMotor
#Version: 2022.2.0
#Motor Name: Nissan_Leaf
#Catalog Name: User_SM_PM_IR_3Ph
#Family: Synchronous
#Type: Permanent magnet
#Sub-type: Inner rotor
#No. phases: 3.0
#End_export_details


#Table_name=Input_JouleLosses
#Table_format=KeyValue
#Version=1.0
#
#Item_number=12
#
#---General_Parameters---
Current_definition_mode=Current
Max_line_current,_rms_(A)=+3.000000E+02
Max_current_dens,_rms_(A/mm2)=+7.000000E+00
No_comp_for_Jd,_Jq=+5.000000E+00
Maximum_speed_(rpm)=+6.000000E+03
No_comp_for_speed=+5.000000E+00
Thermal_solving=No
Winding_straight_part_temperature_(°C)=+1.000000E+02
CS_end_winding_temperature_(°C)=+1.000000E+02
OCS_end_winding_temperature_(°C)=+1.000000E+02
Magnet_temperature_Tmag_(°C)=+1.500000E+02
Rotor_initial_position_(deg)=+4.500000E+01


#Table_name=FMUData_JouleLosses
#Table_format=2D
#Version=1.0
#
#Item_number=1
#Axis_number=2
#
#Axis_name_1=Jd(A)
#Axis_name_2=Jq(A)
#
#Axis_unit_1=A
#Axis_unit_2=A
#
#Item_name_1=JouleLosses(W)
#
#Item_unit_1=W
#
5
5

-4.242641E+02   -3.181981E+02   -2.121320E+02   -1.060660E+02   +0.000000E+00
+0.000000E+00   +1.060660E+02   +2.121320E+02   +3.181981E+02   +4.242641E+02

+3.640349E+03   +2.047696E+03   +9.100871E+02   +2.275218E+02   +0.000000E+00
+3.867870E+03   +2.275218E+03   +1.137609E+03   +4.550436E+02   +2.275218E+02
+4.550436E+03   +2.957783E+03   +1.820174E+03   +1.137609E+03   +9.100871E+02
+5.688045E+03   +4.095392E+03   +2.957783E+03   +2.275218E+03   +2.047696E+03
+7.280697E+03   +5.688045E+03   +4.550436E+03   +3.867870E+03   +3.640349E+03
```

- A folder MAT_PSIM stocking the MAT file. It is written in version 5.0 and can be read by any software supporting this format. Here, it can be seen that the variables contained by this file are read by Compose®.



Command History | Variable Browser

Session Information

Name	Value	Type	Scope
ANGPOS_ROTOR_DEG	45	number	Base
AXIS_MAGNET_TEMP	150	number	Base
Catalog_Name	User_SM_PM_IR_3Ph	string	Base
FLUX_D	<matrix(1x25)>	matrix	Base
FLUX_ID0	[0.0671220325 0.0676971718 0.0632808409 ...	matrix	Base
FLUX_Q	<matrix(1x25)>	matrix	Base
Family	Synchronous	string	Base
ID_PEAK	[-424.264068711929 -318.198051533946 -21...	matrix	Base
IQ_PEAK	[0 106.066017177982 212.132034355964 318...	matrix	Base
J_inertia	0.0299335738386677	number	Base
LD_DYN_vs_ID	<matrix(1x25)>	matrix	Base
LD_DYN_vs_IQ	<matrix(1x25)>	matrix	Base
LD_STAT_vs_ID	<matrix(1x25)>	matrix	Base
LD_STAT_vs_IQ	<matrix(1x25)>	matrix	Base
LOSS_IRON	<matrix(1x125)>	matrix	Base
LOSS_JOULE	<matrix(1x25)>	matrix	Base
LOSS_MECHANICAL	<matrix(1x200)>	matrix	Base
LQ_DYN_vs_ID	<matrix(1x25)>	matrix	Base
LQ_DYN_vs_IQ	<matrix(1x25)>	matrix	Base
LQ_STAT_vs_ID	<matrix(1x25)>	matrix	Base
LQ_STAT_vs_IQ	<matrix(1x25)>	matrix	Base
L_end_winding	1.11642517844301e-05	number	Base
Motor_Name	Nissan_Leaf	string	Base
No_phases	3.0	string	Base
Phi_M	0.0671220325	number	Base
R_phase	0.0154984181044671	number	Base
SPEED_RPM	[1200 2400 3600 4800 6000]	matrix	Base
Software_source	AltairFluxMotor	string	Base
Sub_type	Inner rotor	string	Base
TEST_CURRENT_DEFINITION_MO...	CURRENT	string	Base
TEST_CURRENT_DENSITY_RMS	7	number	Base
TEST_CURRENT_LINE_RMS	300	number	Base
TEST_MAXIMUM_SPEED	6000	number	Base
TEST_NO_COMPTUTATIONS_FOR...	5	number	Base
TEST_NO_COMPTUTATIONS_FOR...	5	number	Base
THERMAL_MAGNET_SOLVING_MO...	MAGNET_CONSTANT_TEMPERATURE	string	Base
TORQUE	<matrix(1x25)>	matrix	Base
T_ACTIVE_LENGTH_WINDING_C	100	number	Base
T_CS_END_WINDING_C	100	number	Base
T_MAGNET_C	150	number	Base
T_OCS_END_WINDING_C	100	number	Base
Type	Permanent magnet	string	Base
Version	2022.2.0	string	Base
initial_angle_rotor_deg	45	number	Base
num_pole_pairs	4	number	Base