MEGARA Control System. User Manual.

Fiber MOS Positioning Tool

GTC Code:

Code: TEC/MEG/171

Issue: 1.D

Date: 15/06/2016

Pages: 25

Authors:	Isaac Morales Durán
Revised by:	África Castillo Morales Ana Pérez Calpena Armando Gil de Paz Jorge Iglesias Páramo
Approved by:	Armando Gil de Paz Maria Luisa García Vargas

Distribution List:

Name	Affiliation	Date
GRANTECAN	GRANTECAN	23/06/2015
MEGARA Team	All Partners	23/06/2015

Acronyms:

CP	Configuration Plan
CB	Configuration Block
DDS	Disjoint Disperse Subsets
FMPT	Fiber MOS Positioning Tool
FMAT	Fiber MOS Assignment Tool
FMOSA	Fiber MOS Assignment file
PPA	Pair of Positions Angles
PP	Positioning Program
DP	Depositioning Program
SP	Sky Point
SPM	Security Perimetral Margin
S/W	Software
TBD	To Be Defined

Change Control

Issue	Date	Section	Page	Change description
1.A	26/01/15	All	All	New Issue
1.B	23/06/15	3	9	Description on how to get the help of the tool changed.
		4	9	Description of the input file for the FMPT rewritten.
		5	11	Description of the output files from the FMPT rewritten.
		6	14	Description of the new functionalities of the FMPT rewritten.
1.C	21/02/16	-	7-8	Index updated.
		All	All	Corrected use of abbreviation S/W.
		1	9	Implemented functionality for generate parking program.
		All	All	Version number updated to 3.0.7.
		2	10	Execute ‘ldconfig’ at the end of installation.
		2	10	Installation tested in Ubuntu 14.04.4
		2	10	Application functioning in Solaris 10.
		3	10	Output command help updated.
		4	11-12	Corrected description of format file FMOSA.
		5	13	Modified example of FMOSA file.
		5.1	13	Modified description of content of log file.
		5.2	13	PP and DP are stored in a single file jointly the content of FMOSA input file.
		5.2	14-15	Modified description of format of the output file.
		5.3	15	Modified description of file ‘outputs-from-<inputfile>’.
		6.1	16	Generating a pair (PP, DP) is generated a file called ‘PPDPandFMOSA-from-<inputfile>’.
		6.2	16-17	Motion program for recovery now is called parking program instead depositioning program. Furthermore has been described the syntax of the function for generate pairs (PP, DP).
		6.3	17	Described the syntax of the function for generate parking programs.

1.D	15/06/16	All	All	Strings changed: “<inputfile>” → “<inputfilename>” “<inputfile>.log” → “fmpt_saa.log” “PPDPandFMOSA” → “outputs” “outputs” →”other_outputs” “FMOSA file” → “file type FMOSA” “3.0.6” → “3.3.0” “3.0.7” → “3.3.0”
		3	10-11	Updated output of command $ fmpt_saa help
		4	11	Added phrase: “These parameters are ignored by the FMPT.”
		5.1	13-14	Updated content of log file ‘fmpt_saa.log’.
		5.3	17	Updated content of file ‘other_outputs...txt’.
		6.1	18	Changed strings: generatePPDP → generatePairPPDP_offline <path> → <path_FMOSA>
				Replaced figure due to error:
		7.1.2	20	Figure 1 (error in labels P0 and P1).
		7.1.3	21	Figure 2 (error in label P3).
				Updated content of files of the FMM instance:
		7.6	23	Instance.txt
		7.7	24	ExclusionArea<Id>/Contour_.txt
		7.8	24	ExclusionArea<Id>/Instance.txt
		7.9	24	RoboticPositioner<Id>/Contour_____.txt
		7.10	24	RoboticPositioner<Id>/Contour_.txt
		7.11	24	RoboticPositioner<Id>/F1.txt
		7.12	25	RoboticPositioner<Id>/F2.txt
		7.13	25	RoboticPositioner<Id>/Instance.txt

Reference Documents

Nº	Document Name	Code

Reference Documents (GTC codes)

Nº	Document Name	Code

INDEX

1. INTRODUCTION

This document contains the User Manual of the MEGARA Fiber MOS Positioning Tool (FMPT). The manual will allow the user to install the FMPT in his computer and to understand the FMPT functionality.

The S/W to configure the Fiber MOS has been split in two tools: Fiber MOS Assignment Tool (FMAT) and Fiber MOS Positioning Tool (FMPT). The FMAT will allow the user to prepare a Configuration Plan (CP) composed by several Configuration Blocks for the MEGARA MOS mode by assisting the user in the assignment of their sources to the different in-use robotic positioners of the MEGARA MOS (up to 92 positioners). Each file containing the information of a particular Configuration Block will be the input for the Fiber MOS Positioning Tool. This tool will compute the sequence of movements that will be commanded before starting the exposure for reaching the source positions (positioning program, PP) as well as the sequence of movements that will be commanded after the exposure is finished for coming back to the parking position (depositioning program, DP). The Fiber MOS Positioning Tool will also allow computing a depositioning program to come back from an anomalous situation (during the observation) receiving, in this case, as input the current positions of the positioners that are not at the parking position.

This manual describes only the installation and use of the FMPT, which is composed by the FMPT library, containing all necessary functionalities, and the application FMPT SAA, which will use the dynamic library 'libfmpt.so'. The functionalities of the FMPT will be the following:

1. The GRANTECAN staff will use the FMPT as a standalone tool to generate pairs (PP, DP) for the Configuration Blocks of a CP generated and previously validated by the FMAT. The FMPT also includes the validation of the source assignments to ensure there is no risk of collisions between the RPs.

2. MEGARA Control System (MCS) will make use of some FMPT functionalities to:

    - Generate a depositioning program to come back from an anomalous situation (during the observation) knowing the disabled RPs and the positions of all RPs.

    - Regenerate the pair (PP, DP) from the information about the positions of the disabled RPs. The rest of the RPs are in parking positions after executing the depositioning program previously computed.

2. HOW TO INSTALL THE FMPT

The process to install this S/W can be found in the file INSTALL. The process is summarized below:

1. Decompress tar file 'megara-fmpt-3.3.0.tar.gz' (or .xz).

2. Change to directory './megara-fmpt-3.3.0'.

3. Execute './configure; make;' to configure and build this package.

4. Grant superuser privileges to have access to '/usr/local/...'.
5. Execute 'make install' (as superuser) to install the library and the application.
6. Execute 'ldconfig' (as superuser) for update the linkage with the libraries of the system.

Both dynamic and static library will be installed in the usual directories of the operative system, according to the user distribution and version.

As an example, in Ubuntu 14.04.4 and Fedora 20:

- Static and dynamic libraries will be installed in '/usr/local/lib';
- The executable file fmpt_saa will be installed in '/usr/local/bin';
- The instance of the Fiber MOS Model will be installed in:

  '/usr/local/share/megara-fmpt/Models/MEGARA_FiberMOSModel_Instance';
- The sample input files will be installed in: 'usr/local/bin/share/megara-fmpt/Samples'.

These paths can vary depending on the operating system and distribution where this package is installed. In this document the paths mentioned assume that the installation has been done in Ubuntu 14.04.4.

The current version of the FMPT allows installation in Linux systems and Solaris 10 systems.

3. GETTING HELP RUNNING THE FMPT SAA

The user can obtain help about the arguments of the FMPT SAA by typing:

```
$ fmpt_saa help
```

And the terminal shows:

```
user@lenovo-W520:~$ fmpt_saa help
FMPT SAA 3.3.0 is running...

Arguments with has called the program:
    argv[0]: fmpt_saa
    argv[1]: help

The FMM instance will be searched in:
    '/usr/local/share/megara-fmpt/Models/MEGARA_FiberMOSModel_Instance'
    '/home/user/../data/Models/MEGARA_FiberMOSModel_Instance'

$ fmpt_saa help
    Print this help.

$ fmpt_saa generatePairPPDP_offline <path_FMOSA>
    <path_FMOSA>: absolute or relative path to file type FMOSA.
    Generate a pair (PP, DP) offline.
    In the file type FMOSA:
        The following parameters could be empty: Name, Mag, Pr, Bid and Comment.
        Parameter Bid indicates if the source is allocated or no.
        Parameter Enabled indicates if the RP is enabled or no.
        When Bid is empty:
            parameters Name, Mag, Pr and Comment will be empty;
            parameter Type will be UNKNOWN.

$ fmpt_saa aboutOf
         Print the legend about of...
```

4. INPUTS OF THE FMPT SAA

In the following sections we describe in detail the input and output files of the FMPT SAA. In the appendix of this document, the files that constitute the instance of the Fiber MOS Model, which is the major defining parameter for the program, are described.

For each Configuration Block the user needs to execute the FMPT to generate the positioning and depositioning programs (PP and DP). The input file for the FMPT is the Fiber MOS Assignment file (FMOSA), which should be loaded by the astronomer during the Phase-2. This input file is comprised by two sections, the configuration block (CB) and the source list:

The file starts with a comment row (ignored by the S/W) with the column description of the CB:

```
# Id| Ra| Dec| Pos
```

These field names correspond to the Identification number (Id) for the Block, Right Ascension and Declination where the center of the MEGARA MOS FOV (which coincides with the FC optical axis) is pointing along the FC rotator Position Angle (between 0º and 360º). All the angles are in degrees with six decimal positions. These parameters are ignored by the FMPT.

A row indicating that the information about the configuration block starts:

```
@@SOB@@
```

A line representing a CB, for example:

```
0| 15.01564| 45.004671| 0.050512
```

A row indicating that the information about the CB ends:

```
@@EOB@@
```

A comment row (ignored by the S/W) with the brief indication of the meaning of the different columns for each source included in this input file:

```
# Name Ra Dec Mag Type Pr Bid Pid X(mm) Y(mm) Enabled Comment
```

These correspond to the Source Name, Right Ascension and Declination of the source, Magnitude, Type of source, Priority, Block Id, Positioner Id, X, Y position of the positioner and whether the positioner is enabled/disabled and a Comment field. All the angles are in degrees with six decimal positions.

A row indicating that the list of sources starts.

```
@@SOS@@
```

A number of lines, each one representing a source, for example:

```
s:116|15.011244|45.022607|15.04|SOURCE|1|0|1|-9.171006|53.226932|1|foo comment
s:596|14.986557|45.032844|18.66|SOURCE|0|0|2|-60.909235|83.658767|1|foo comment
…
```

Note that both sources are assigned to the block 0 to positioners 1 and 2 and with priority 1 and 0 respectively.

A row indicating that the list of sources ends.

```
@@EOS@@
```

For each source, in each row, we can find 12 fields separated by vertical bars corresponding to parameters of a given Sky Point (SP). The fields are the following ones:

1. **Name**: Name of the source (catalog name, e.g. Mk348, NGC4151, etc.). Can be empty for unallocated sources.
2. **Ra:** Right Ascension in J2000.0. The format is degrees with six decimal positions.
3. **Dec:** Declination in J2000.0. The format is degrees with six decimal positions.
4. **Mag:** Magnitude. In this release this field interprets that magnitudes are all in the same band and photometrical system. Can be empty for unallocated sources.
5. **Type:** There are 4 types of SPs in the input preselected-sky-point list for the FMAT:
    - **SOURCE**: target of scientific interest. It could be allocated or not to a RP.
    - **REFERENCE**: target to be used as reference. It could have scientific interest or not. Reference-source SP corresponds to a source (star) that can be used to validate the correct position of the telescope and PA. It is generally recommended that every observation (at least for long exposures) contains, at least, three reference-source SPs.
    - **BLANK**: sky background source. A blank SP is a point in a blank-sky region of the field. It is generally recommended one blank SP at least in every observation with the Fiber MOS. The blank SP observation will be used in the DFP/DRP and quick-look tools for sky subtraction.
    - **UNKNOWN**: unknown type point (type has not been previously identified). This is the type for unallocated sources.
6. **Pr:** level of priority assigned to each source. The highest priority is “0” and the lowest priority is “10”. Can be empty for unallocated sources.
7. **Bid:** Block Id. **Will be empty for unallocated sources.**
8. **Pid:** Positioner Id. Can be empty for unallocated sources.
9. **X (mm):** Positioner x co-ordinate (mm with six decimals) in the Fiber MOS co-ordinate system.
10. **Y (mm):** Positioner y co-ordinate (mm with six decimals) in the Fiber MOS co-ordinate system.
11. **Enabled:** Boolean field to indicate if the RP identified by the Pid is enabled (1 value) or

disabled (0 value).

12 **Comment:** Field for comment.

The FMPT will read from this input file the values of the properties (Pid, X, Y) for each assigned RP and also the Priority and Enabled columns information. These properties represent a Projection Point allocated to the RP whose identifier matches with this particular Id.

Example of an input file for the FMPT ('megara-cb0.txt')

```
# Id| Ra| Dec| Pos
@@SOB@@
0| 15.027879| 45.000422| 0.060858
@@EOB@@
# Name     RA        Dec       Mag   Type        Pr Bid Pid   X(mm)   Y(mm)    Enabled  Comment
@@SOS@@
s:751     |15.027787|45.017925|20.67|SOURCE     |6 |0  |1  |-0.137909 |51.932865 |1|foo comment
s:404     |15.003376|45.028110|21.74|SOURCE     |0 |0  |2  |-51.296480|82.217116 |1|foo comment
…
          |15.066176|45.029778|     |UNKNOWN    |  |   |97 |80.400000 |87.035553 |1|
@@EOS@@
```

5. OUTPUTS OF THE FMPT SAA

After generating a pair (PP, DP) the output files from the FMPT, located at the working directory, would be the following:

5.1 Log file

This file, named fmpt_saa**.log**, contains the same output written in the terminal when the application is executed:

```
user@lenovo-W520:~$ fmpt_saa generatePairPPDP_offline /usr/local/share/megara-
fmpt/Samples/megara-cb0.txt
FMPT SAA 3.3.0 is running...

Arguments with has called the program:
    argv[0]: fmpt_saa
    argv[1]: generatePairPPDP_offline
    argv[2]: /usr/local/share/megara-fmpt/Samples/megara-cb0.txt

The FMM instance will be searched in:
    '/usr/local/share/megara-fmpt/Models/MEGARA_FiberMOSModel_Instance'
    '/home/user/../data/Models/MEGARA_FiberMOSModel_Instance'

Fiber MOS Model instance loaded from '/usr/local/share/megara-
fmpt/Models/MEGARA_FiberMOSModel_Instance'.
FMOSA table loaded from '/usr/local/share/megara-fmpt/Samples/megara-cb0.txt'.
Allocations got from the FMOSA table in MPG.
RPs moved to observing positions.
Observing position list saved in './OPL-from-megara-cb0.txt'.
Observing position list (in cartesian coordinates respect S0) saved in './OPL_S0-from-
megara-cb0.txt'.
Observing position list (in cartesian coordinates respect S1) saved in './OPL_S1-from-
megara-cb0.txt'.
Generating pair (PP, DP)...
Generated pair (PP, DP) is valid.
PP in FMPT format saved in './PP-FMPT-from-megara-cb0.txt'.
DP in FMPT format saved in './DP-FMPT-from-megara-cb0.txt'.
Initial position list saved in './IPL-from-megara-cb0.txt'.
Displacement corners 1 saved in './Disp_Corners1-from-megara-cb0.txt'.
Displacement corners 2 saved in './Disp_Corners2-from-megara-cb0.txt'.
Positiong program translated to the MCS format.
PP in MCS format saved in './PP-from-megara-cb0.txt'.
Depositiong program translated to the MCS format.
DP in MCS format saved in './DP-from-megara-cb0.txt'.
Pair (PP, DP) saved in './outputs-from-megara-cb0.txt'.
PP-Dmin saved in './PP-Dmin-from-megara-cb0.txt'.
DP-Dmin saved in './DP-Dmin-from-megara-cb0.txt'.
There are collided RPs: {34, 44, 46, 55, 57, 67, 75}
Other outputs saved in './other_outputs-from-megara-cb0.txt'.
```

5.2 File containing the pair (PP, DP)

This file, named **outputs-from-<inputfilename>**, contains three blocks of data (the positioning programa, the depositioning program and the same content of the input file type FMOSA).

Each motion program is composed by a list of groups, where each group is a list of sentences representing each an instruction to be sent to the controller of a rotor of a RP, before starting a jointly movement.

The general description of this file is the following:

```
# Positioning program
@@SPP@@
pos_<cb_string_id> {
        group_01 {
        rp01 r1 <p_1>
        rp01 r2 <p___3>
        rp17 r1 <p_1>
        rp17 r2 <p___3>
        ...
        }
        group_02 {
        rp03 r1 <p_1>
        rp03 r2 <p___3>
        rp22 r1 <p_1>
        rp22 r2 <p___3>
        ...
        }
        ...
}
@@EPP@
# Depositioning program
@@SDP@@
depos_<cb_string_id> {
        group_01 {
        rp01 r1 <p_1>
        rp01 r2 <p___3>
        rp17 r1 <p_1>
        rp17 r2 <p___3>
        ...
        }
        group_02 {
        rp03 r1 <p_1>
        rp03 r2 <p___3>
        rp22 r1 <p_1>
        rp22 r2 <p___3>
        ...
        }
        ...
}
@@EPP@
# Id| Ra| Dec| Pos
@@SOB@@
0| 15.027879| 45.000422| 0.060858
@@EOB@@
# Name     RA        Dec       Mag   Type        Pr Bid Pid   X(mm)    Y(mm)    Enabled  Comment
@@SOS@@
s:751     |15.027787|45.017925|20.67|SOURCE     |6 |0  |1  |-0.137909 |51.932865 |1|foo comment
s:404     |15.003376|45.028110|21.74|SOURCE     |0 |0  |2  |-51.296480|82.217116 |1|foo comment
…
          |15.066176|45.029778|     |UNKNOWN    |  |   |97 |80.400000 |87.035553 |1|
@@EOS@@
```

Where the <cb_string_id> is the identifier of the CB for which a motion program (the PP or the DP) has been computed.

Each motion sentence must be written in two lines (corresponding to the two rotors of a RP), and has the following format:

rp<Id> r1 <p_1>
rp<Id> r2 <p___3>

where:

- <Id> represents the Id of the RP, completed with zeros on the left to get the length of two figures.
- <p_1>: is the position where the rotor 1 of the RP is going to be moved in steps.
- <p___3>: is the position where the rotor 2 of the RP is going to be moved in steps.

We note that (<p_1>, <p___3>) are the positions where the rotors shall be moved.
Therefore, in the case of computing a depositioning program, the final instructions for each rotor of each RP will be:

```
rpid r1 0
rpid r2 0
```

In the case of a positioning program, the final instructions for each rotor of each RP are not necessarily (0, 0) since the final positions must be the ones corresponding to the coordinates of the corresponding source.

5.3 File containing other output data

This file, named **`other_outputs-from-<inputfilename>`**, is intended to contain simple data generated by the application.

Example: ‘other_outputs-from-megara-cb0.txt’

```
DPvalid: True
PPvalid: True
Collided: {34, 44, 46, 55, 57, 67, 75}
Obstructed: {}
```

Where:

- “PPvalid”: indicates if the positioning programs have been successfully generated or conversely the (PP, DP) could not be obtained.
- “DPvalid”: indicates if the depositioning programs have been successfully generated or conversely the (PP, DP) could not be obtained.
- “Collided”: Two RPs are considered as "collided" when the minimum distance between them is lower than the sum of the SPMs of the individual RPs. In the case that all RPs are enabled there will be no collided RPs and this field will contain an empty list. But in the case that some RPs are blocked in insecurity positions there could be some collided RPs and this field will contain the list of those RPs.
- “Obstructed”: A RP is obstructed when there is one or more adjacent disabled RPs blocked in insecurity positions that prevent the retraction of this RP to a security position. In the case that all the RPs are enabled there will be no obstructed RPs and this field will contain an empty list. But in the case that some RPs are blocked in insecurity positions there could be some obstructed RPs and this field will contain the list of those RPs.

6. EXECUTION OF THE FMPT

6.1 Generating a pair (PP, DP)

The GRANTECAN staff will make use the FMPT as a stand-alone tool to generate pairs (PP, DP) for the Configuration Blocks of a CP generated, and previously validated by the FMAT. The FMPT also includes a validation to ensure there is no risk of collisions between the RPs.

The syntax for this is as follows:

```
$ fmpt_saa generatePairPPDP_offline <path_FMOSA>
      <path_FMOSA>: absolute or relative path to file type FMOSA.
```

Setting parameters:

A directory containing the Fiber MOS Model Instance in the path: **/usr/local/share/megara-fmpt/Models**.
This parameter is loaded by default from ‘MEGARA_FiberMOSModel_Instance’.

Inputs:

File **<path_FMOSA>**[1] containing a table with the projection points and their allocations to RPs.

The following output files will be saved in the same directory where the program is executed: **fmpt_saa.log**, **outputs-from-<filename>**, **other_outputs-from-<filename>**. Where filename is the name of the input file type FMOSA indicated in <path_FMOSA>.

6.2 Generating a parking program

This functionality will be used by MCS when an anomalous situation has occurred during the execution of the observation with the Fiber MOS and one or more RPs have been disabled, or simply they has been stopped in intermediate positions of their trajectories, while the pair (PP, DP) was being executed.

MCS will call the FMPT to compute a parking program to move the RPs to their original positions (parking positions). The FMPT would need as input the list of the disabled RPs and the list of the current positions of all the RPs to compute the parking program. This information will be provided by MCS as arguments when this function is called. The syntax to execute this functionality in MCS is the following:

```
bool generateParkingProgram_online(TMotionProgram& ParkingProgram,
    TFiberMOSModel& FMM,
    const vector<double>& p_1s, const vector<double>& p___3s,
    const vector<int>& Ids);
```

Where:

- ParkingProgram: is the parking program to be generated.
- FMM: is the Fiber MOS Model.

[1] This file is the output of the FMAT whose name is ‘megara-cb0.txt’ or similar (cb1, cb2, etc.).

- (p_1s, p___3s): are the starting positions (in steps), of all RPs in the FMM.
- Ids: is the list of identifiers of the RPs to be disabled.

This function is internally documented in the file 'MotionProgramGenerator.h' (provided with the release 3.0.7 of the FMPT).

6.3 Regenerate a pair (PP, DP)

When it is the moment of execute the pair (P, DP) corresponding to a CB, a malfunction or anomaly could occur. This could occur before execute the pair, or during the execution of it. For solve the anomalous situation could be required disable the fault RPs, and recovery the original positions of the RPs not anomalous outside the origin, through a parking program. Disabling of any RP in insecure position, invalidate the pair, because their execution could produce a collision.

MCS will call the FMPT for regenerate the pair (PP, DP), calling the function whose syntax is briefly described below:

```
bool generatePairPPDP_online(TMotionProgram& PP, TMotionProgram& DP,
    TFiberMOSModel& FMM,
    const vector<double>& p_1s, const vector<double>& p___3s,
    const vector<int>& Ids);
```

Where:

- (PP, DP): the pair to be generated.
- FMM: the Fiber MOS Model.
- (p_1s, p___3s): are the initial positions in steps, of all RPs in the FMM.
- Ids: is the list of identifiers of the RPs to be disabled.

This function is similar to the function for generate the original pair, but including the parameters (p_1s, p___3s) and Ids.

This function is internally documented in the file 'MotionProgramGenerator.h' (provided with the release 3.0.7 of the FMPT).

7. APPENDIX: INSTANCE OF THE FIBER MOS MODEL

7.1 FMPT coordinate systems

The Fiber MOS Positioning Tool (FMPT) will use internally four different coordinate systems, which are described in the following subsections.

7.1.1 FMPT S0 coordinate system

The FMPT S0 coordinate system coincides with the Fiber MOS focal frame coordinate system.

The Fiber MOS focal frame coordinate system is defined as follows:

- The origin is in the center of the frame as defined in A.2.
- The Z-axis runs in direction of the Z-axis of the GFCCS against the incoming light.
- The Y-axis runs in direction of the Y-axis of the GFCCS, positive sense of Y-axis going away from the base plate.
- The X-axis form a right handed system with the two previous.

The Fiber MOS focal frame shall be used to define the position of the each positioner and, therefore, locate the origin of each individual Fiber MOS positioner coordinate system.

7.1.2 FMPT S1 coordinate system

The FMPT S1 coordinate system is defined as follows:

- This is dextrorotatory.
- The origin of the system is at the centre of rotation 1.
- The x-axis is oriented passes through rotation 1 and rotation 2 centres.

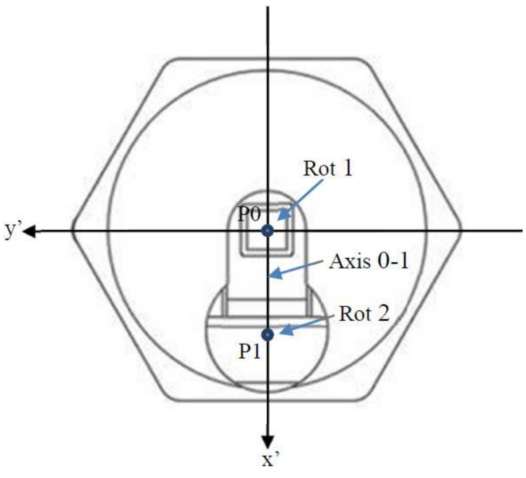

Figure 1: *FMPT S1 coordinate system*

7.1.3 FMPT S2 coordinate system

The FMPT S2 coordinate system is defined as follows:

- This is dextrorotatory.
- The origin of the system is at the centre of rotation 2.
- The x-axis is orientated in such way that it works as symmetrical axis to the arm's contour.

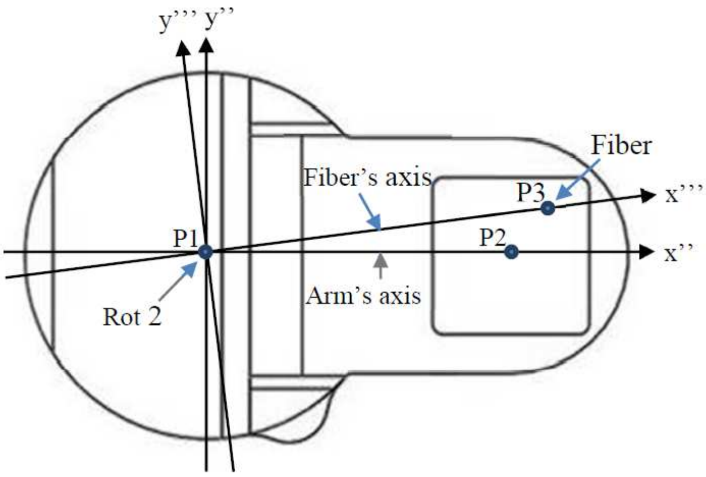

Figure 2: *FMPT S2 and S3 coordinate systems*

7.1.4 FMPT S3 coordinate system

The FMPT S3 coordinate system is defined as follows:

- This is dextrorotatory.
- The origin of the system is at the centre of rotation 2.
- The x-axis is orientated in such way that passes through rotation 1 and the centre of the fiber minibundle.

7.2 Set of files composing the instance of the Fiber MOS Model

An instance of the Fiber MOS Model is composed by a directory containing the following sets of data:

- **RP Origins Table**: containing a table to indicate the position and orientation of each RP in the Fiber MOS Model, containing the following fields: (Id, x0, y0, thetaS1).
- **EA Origins Table**: containing a table to indicate the position and orientation of each exclusion area (EA) in the Fiber MOS Model, containing the following fields: (Id, x0, y0, thetaS1).

- **RP Map**: containing a matrix to transform each point (x, y) given in Cartesian coordinates in S0, in a list of identifiers of RPs {Id} in whose observing domain could be the point (x, y).
- **Fiber MOS Additional Properties**: containing the additional properties of the Fiber MOS Model instance.
- **A subdirectory for each RP containing the following text files**:
  - **RP Arm Contour____**: containing a list of segments and arcs to describe the template of the contour of the RP Arm, in S4 (in mm).
  - **RP Frontier Contour_**: containing a list of segments and arcs to describe the template of the contour of the RP Frontier, in S1 (in mm).
  - **RP F1**: containing the compression function of the quantifier of rot 1 of the RP (in step/rad). Shall be defined in a domain that contains rot 1 domain [theta_1min, theta_1max].
  - **RP F2**: containing the compression function of the quantifier of rot 2 of the RP in (step/rad). Shall be defined in a domain that contains rot 2 domain [theta___3min, theta___3max].
  - **RP Additional Properties**: containing the additional properties of the RP Model instance. **WARNING:** this file must be updated by the GRANTECAN staff each time a RP is disabled in the real Fiber MOS, by changing to "True" the field "Disabled" (at the bottom of this file) corresponding to that RP.
- **A subdirectory for each EA containing the following text files**:
  - **EA Barrier Contour_**: containing a list of segments and arcs to describe the template of the contour of the EA Barrier, in S1 (in mm).
  - **EA Additional Properties**: containing the additional properties of the EA Model instance.

Each set of these sets of data is stored in a file. These files are stored in the directory of the instance being root their proprietary, because the user usually shall not modify the instance.

The names of these files and the default values of the data are shown below, with comments added to facilitate understanding.

7.3 Fiber MOS Model - RP Origins Table (RoboticPositionerOriginsTable.txt)

```
#A table for indicate the position and orientation of each RP (Robotic Positioner):
#    Id: identifier of the RP (a nonnegative integer number)
#    x0: abscissa of the point P0 of the RP in S0 (in mm)
#    y0: ordinate of the point P0 of the RP in S0 (in mm)
#    theta01: orientation of S1 in S0 (in rad)

Id      x0      y0              theta01
2       -60.3   87.035553       0
4       -40.2   87.035553       0
…       …       …               …
91      60.3    -87.035553      0
```

7.4 Fiber MOS Model - EA Origins Table (ExclusionAreaOriginsTable.txt)

```
#A table for indicate the position and orientation of each EA (Exclusion Area):
#    Id: identifier of the EA (a nonnegative integer number)
#    x0: abscissa of the point P0 of the EA in S0 (in mm)
#    y0: ordinate of the point P0 of the EA in S0 (in mm)
#    theta01: orientation of S1 in S0 (in rad)

Id      x0      y0      theta01
1       0       0       0
```

7.5 Fiber MOS Model - RP Map (InstanceMap.txt)

```
#A matrix to transform each point (x, y) given in Cartesian coordinates in S0,
#in a list of identifiers of RPs {Id} in whose observing domain could be the point
(x, y):

xmin = -102.055                # lower limit in x-axis of the matrix of cells (in mm)
xmax = 102.055                 # upper limit in x-axis of the matrix of cells (in mm)
ymin = -98.640553              # lower limit in y-axis of the matrix of cells (in mm)
ymax = 98.640553               # upper limit in y-axis of the matrix of cells (in mm)

I = 49                         # number of rows of the cell matrix
J = 50                         # number of columns of the cell matrix

#Cells[i][j]: list of RPs whose observing domain is in the square cell (i, j)
Cells[0][0] = {}
Cells[0][1] = {}
…
Cells[1][0] = {}
Cells[1][1] = {}
…
Cells[50][49] = {}
Cells[50][50] = {}
```

7.6 Fiber MOS Model - Instance properties (Instance.txt)

```
#Instance properties of the FMM (Fiber MOS Model):

Tolerances:                # Tolerances of the Fiber MOS:
    PAem = 0                   # orientation error margin of the system S0 (in rad)
    Pem = 0                    # pointing error margin of the system S0 (in mm)
```

7.7 EA - Barrier Contour_ (ExclusionArea<Id>/Countour_.txt)

```
#A list of segments (Pa, Pb) and arcs (Pa, Pb, Pc, R) for describe
#the template of EA.Barrier.Contour_, in S1 (in mm):

({-9, -9.95}, {-14.2, -4.75}, {-9, -4.75}, 5.2)
({-14.2, -4.75}, {-14.2, 6.721909})
({-14.2, 6.721909}, {-13.0999999516436, 8.62716491624432}, {-12, 6.721909}, 2.2)
({-13.0999999516436, 8.62716491624432}, {-10.6000001182939, 10.070541204891})
({-9.7, 11.629387}, {-10.6000001182939, 10.070541204891}, {-11.5, 11.629387}, 1.8)
({-9.7, 11.629387}, {-9.7, 21.2})
({-9.7, 21.2}, {-7.5, 23.4}, {-7.5, 21.2}, 2.2)
({-7.5, 23.4}, {7.5, 23.4})
({7.5, 23.4}, {9.7, 21.2}, {7.5, 21.2}, 2.2)
({9.7, 21.2}, {9.7, 11.629387})
({10.6000001182939, 10.070541204891}, {9.7, 11.629387}, {11.5, 11.629387}, 1.8)
({10.6000001182939, 10.070541204891}, {13.0999999516436, 8.62716491624432})
({13.0999999516436, 8.62716491624432}, {14.2, 6.721909}, {12, 6.721909}, 2.2)
({14.2, 6.721909}, {14.2, -4.75})
({14.2, -4.75}, {9, -9.95}, {9, -4.75}, 5.2)
({9, -9.95}, {-9, -9.95})
```

7.8 EA - Instance properties (ExclusionArea<Id>/Instance.txt)

```
#Instance properties of the EA (Exclusion Area):

BarrierInstance:  # Instance properties of the EA.Barrier (position and orientation):
    P0 = {0, 0}       # position of S1 respect S0 (in mm)
    theta01 = 0       # orientation of S1 respect S0 (in rad)

Tolerances:       # Tolerance properties of the EA:
    Eo = 0.000872665  # error margin in theta_ orientation (in rad)
    Ep = 0.017        # error margin in P0 position (in mm)
```

7.9 RP - Arm Contour____ (RoboticPositioner<Id>/Contour____.txt)

```
#A list of segments (Pa, Pb) and arcs (Pa, Pb, Pc) for describe
#the template of RP.Actuator.Arm.Contour____ in S4 (in mm):

({0.807111, 2.35}, {5.807111, 2.35})
({5.807111, 2.35}, {5.807111, -2.35}, {5.807111, 0}, 2.35)
({5.807111, -2.35}, {0.807111, -2.35})
({0.807111, -2.35}, {0.807111, 2.35})
```

7.10 RP - Barrier Contour_ (RoboticPositioner<Id>/Contour_.txt)

```
#A list of segments (Pa, Pb) and arcs (Pa, Pb, Pc, R) for describe
#the template of the RP.Barrier.Contour_, in S1 (in mm):

({13.955, 0}, {-13.955, 0}, {0, 0}, 13.955)
({-13.955, 0}, {13.955, 0}, {0, 0}, 13.955)
```

7.11 RP – Rotor 1 compression function (RoboticPositioner<Id>/F1.txt)

```
#The compression-function of the quantifier of rot 1 of the RP in step/rad.
#Must be defined almost in the rot 1 domain [theta_1min, theta_1max]:
#    theta_1: position of rot 1 (in rad)
#    p_1: position of rot 1 (in step)

-6.28318530717959   -194953.8462
0                   0
6.28318530717959    194953.8462
12.5663706143592    389907.6924
```

7.12 RP – Rotor 2 compression function (RoboticPositioner<Id>/F2.txt)

```
#The compression-function of the quantifier of rot 2 of the RP in step/rad.
#Must be defined in the rot 2 domain [theta___3min, theta___3max]:
#    theta___3: position of rot 2 (in rad)
#    p___3: position of rot 2 (in step)

-6.28318530717959   -30720
0                   0
6.28318530717959    30720
12.5663706143592    61440
```

7.13 RP - Instance properties (RoboticPositioner<Id>/Instance.txt)

```
#Instance properties of the RP (Robotic Positioner):

ActuatorInstance:              # Instance properties of the RP.Actuator (sizing, orientation and others):
    L01 = 5.8025                   # distance between P0 and P1 (in mm)
    theta_1min = -0.043633231      # position angle lower limit of the axis 0-1 respect S1 (in rad)
    theta_1max = 6.326818538       # position angle upper limit of the axis 0-1 respect S1 (in rad)
    theta_1 = 0                    # position angle of axis 0-1 respect S1 (in rad)
    theta_O3o = 3.14159265358979   # orientation of S3 respect S1 when theta_1 = 0 (in rad)
    SB1 = 194953.8462              # number of steps by lap of rotor 1 (in steps)

    ArmInstance:                   # Instance properties of the RP.Actuator.Arm (sizing, orientation…):
        L12 = 5.8025                   # distance between P1 and P2 (in mm)
        L13 = 5.8025                   # distance between P1 and P3 (in mm)
        theta__O3 = 0                  # orientation of S3 respect S2 (in rad)
        R3 = 0.75                      # radio of representative circle of the microlens (in mm)
        thetaO3 = -3.14159265358979    # orientation of S3 respect S2 (in rad)
        theta___3min = -0.043633231    # position angle lower limit of axis 1-3 respect S3 (in rad)
        theta___3max = 3.185225885     # position angle upper limit of axis 1-3 respect S3 (in rad)
        theta___3 = 0                  # position angle axis 1-3 respect S3 (in rad)
        SB2 = 30720                    # number of steps by lap of rotor 2

    SPMmin = 0.005         # SPM minimum: is the SPM due to the minimum jump during generation (in mm)
    SPMsim = 0.052         # SPM of simulation: is the maximum deviation in the radial trajectory (in mm)
    PAkd = Pre             # position angles knowledge degree [Pre | Apr | Unk]

CMFInstance:               # Instance properties of the RP.CMF:
    SF1.vmaxabs = 3            # absolute maximum velocity of rot 1 when MFT = mftSquare (in step/ms)
    SF2.vmaxabs = 0.945        # absolute maximum velocity of rot 2 when MFT = mftSquare (in step/ms)
    RF1.vmaxabs = 3            # absolute maximum velocity of rot 1 when MFT = mftRamp (in step/ms)
    RF2.vmaxabs = 0.945        # absolute maximum velocity of rot 2 when MFT = mftRamp (in step/ms)
    RF1.amaxabs = 4294.967295  # absolute maximum acceleration of rot 1 when MFT = mftRamp (in step/ms^2)
    RF2.amaxabs = 4294.967295  # absolute maximum acceleration of rot 2 when MFT = mftRamp (in step/ms^2)
    MFM = Square               # motion function type [Square | Ramp]
    SSM = Free                 # square synchronous mode [Free, Tmin, MaxTmin]
    RSM = Free                 # ramp synchronous mode [Free | Tmin | MaxTmin | Tv | MaxTv]
    Id1 = 0                    # CAN identifier of the rot 1 controller (a nonnegative integer number)
    Id2 = 0                    # CAN identifier of the rot 2 controller (a nonnegative integer number)

Tolerances:            # Tolerances of the RP:
    Eo = 0.000872665       # error margin in theta_ orientation (in rad)
    Ep = 0.1               # error margin in P0 position (in mm)
    Tstop = 0              # margin time from last position stored in memory, to stopping rotors (in ms)
    Tshiff = 1             # margin time to shift of all rotors of RPs in Fiber MOS (in ms)
    SPMadd = 0.2           # SPM additional: is a component of SPM added once (in mm)

Status:                # Qualitative status of the RP:
    Disabled = False       # disabling status [False | True]
    FaultProbability = 0   # probability of fault status (a real number in [0, 1])
    FaultType = Unk        # type of faul [Unk | Sta | Dyn]
```