CAYLAR
INSTRUMENTATION SCIENTIFIQUE

Magnetic field system for Atomic Force Microscopy

USER MANUAL

Magnetic Field Controller and Module

Caylar SAS
14 avenue du Québec
BP 612
91140 Villebon-sur-Yvette
France
Tél +33 1 69 29 91 62

MFC5001-001 to 015
April 2024
Rev 1.02

CAYLAR
INSTRUMENTATION SCIENTIFIQUE

Summary

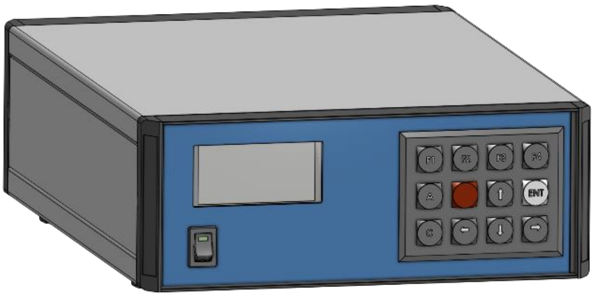
CAYLAR
INSTRUMENTATION SCIENTIFIQUE

Introduction

The aim of this system is to generate a variable bipolar magnetic field in two configurations mode: IN and OUT of plane.

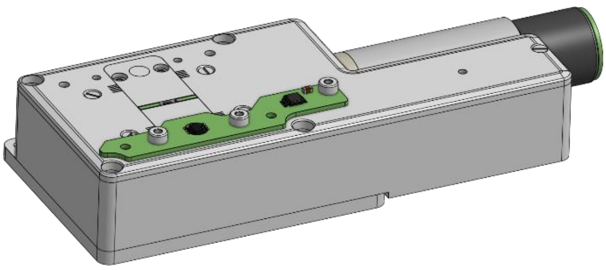
The system consists of two parts:

- A **M**agnetic **F**ield **C**ontroller (**MFC**) which contain all the electronic for supplies, communication and field module control.

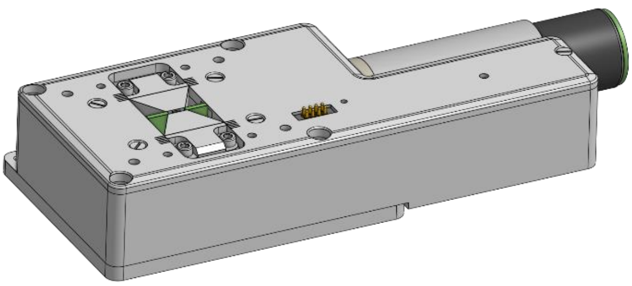


- A **M**agnetic **F**ield **M**odule (**MFM**) which contain a stepper motor and a permanent magnet in order to generate the magnetic field. This module has two configurations manually settable by the user: IN and OUT of plane. The module includes its connection cable to the controller. The cable length is ~ 2.5 meters.

Out of plane version:



In plane version:



The system can be controlled fully manually from the front panel or fully remotely via ethernet, USB or RS232 communication protocols.

The system has a field closed loop control but it can also be used in open loop.

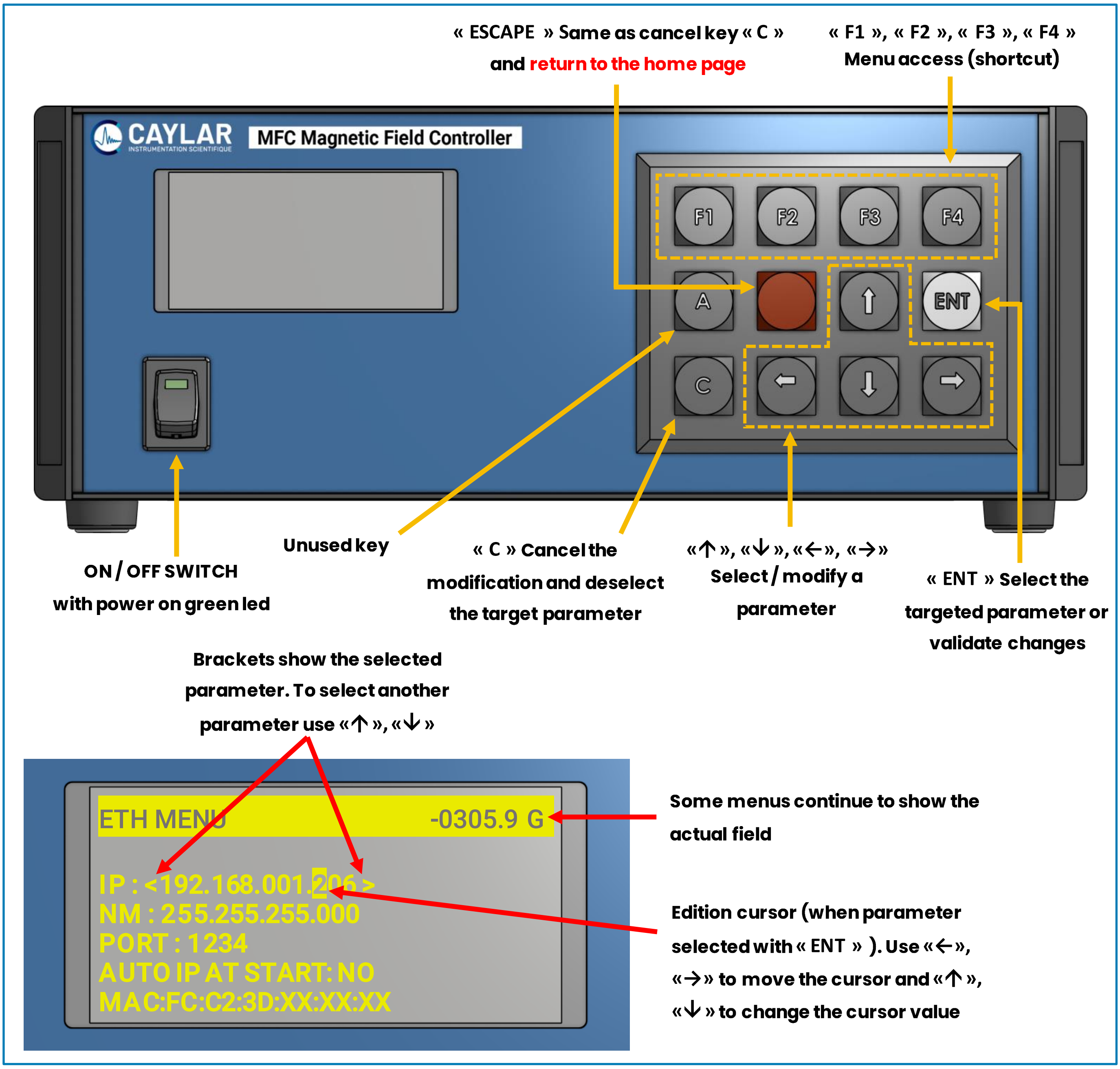
Magnetic Field Controller (MFC) Presentation

The MFC is a 2U rack cabinet which contains all the electronic controls.

From its front panel keyboard and oled screen you can control and monitoring all the system manually. From its rear panel you have access to 3 communication connectors (ethernet, RS232 and serial via USB) for remote control.

Each change made in menus are then permanently saved on the controller SD card. You will therefore find the same parameters during the next MFC power on.

a) Front panel – Keyboard and screen

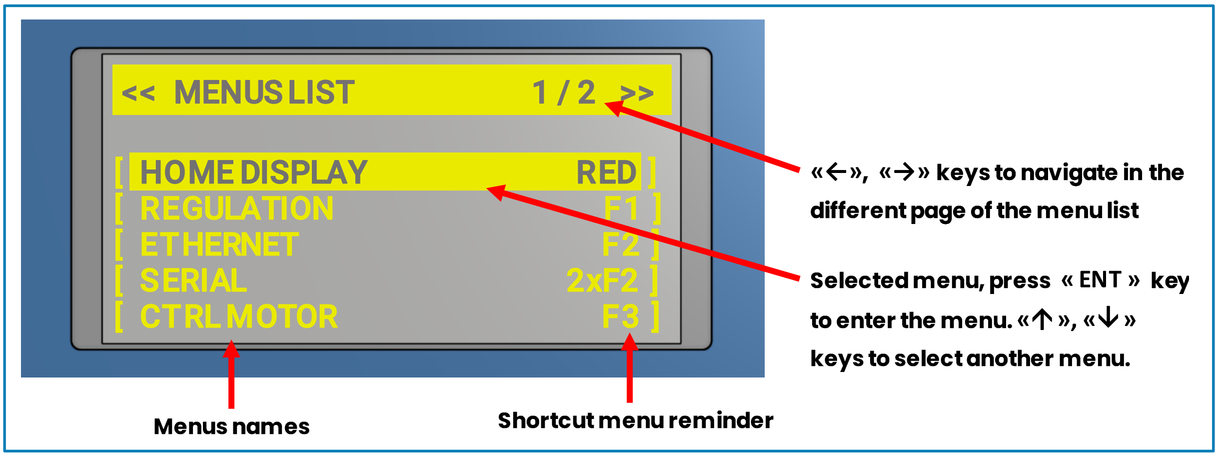


MFC front panel view / example of menu displayed on the oled screen

b) Menus list

The various menus are explained in their associated sections of this documentation. A list of the MFC menus is available from the home page (**« ESCAPE »** key) by pressing the **«→»** key. Then use **«←»** and **«→»** keys to navigate in the different page of the menus list or return to the home page. You can selec a menu with **«↑ »**, **«↓ »** keys and enter the menu with **« ENT »** key.

The menu list also reminds you the shortcuts associated with each menu. The shortcuts are given at the right of the menu names.



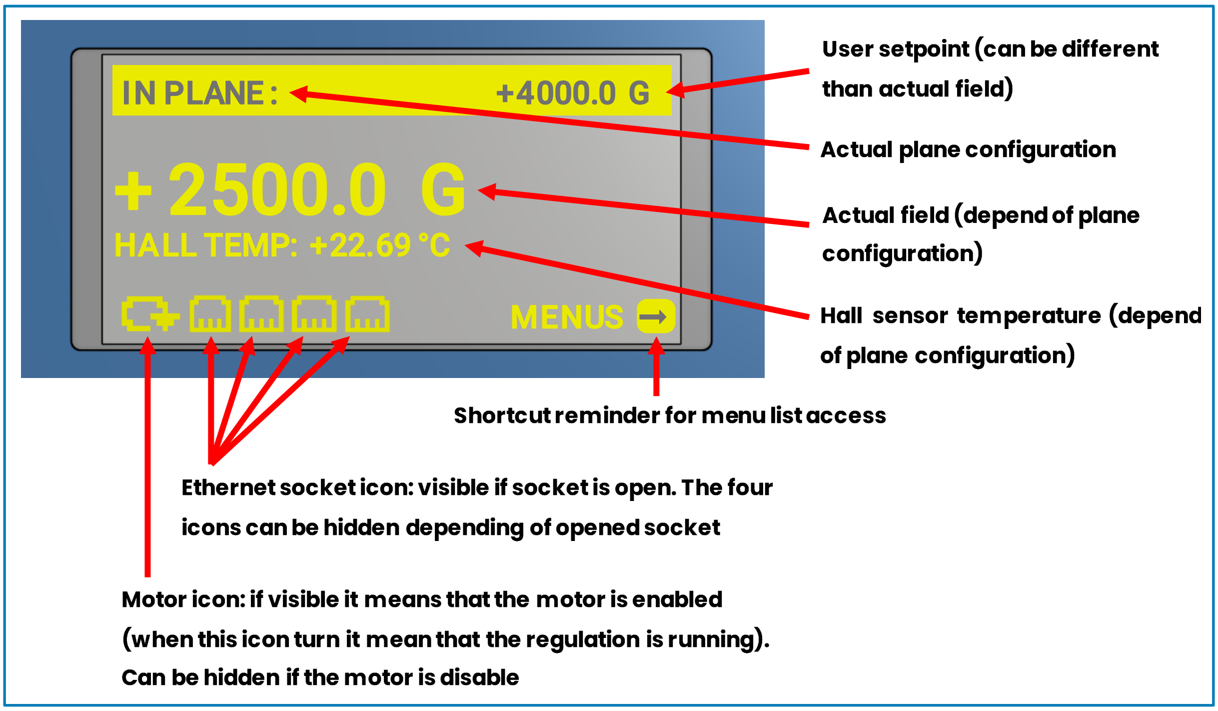
Oled screen view of MENUS LIST page 1 / 2 on MFC front panel

List of MFC menus and pages:

Menu name	Function	Shortcut
Home page	Displays the overall system status (field, setpoint, temperature, ethernet socket connected....)	**ESCAPE**
REGULATION	Control the regulation manually (field setpoint, plane configuration...)	**F1**
OUTP PARAMS	Change the regulation parameters for the out-plane configuration	**F1 x2**
INP PARAMS	Change the regulation parameters for the in-plane configuration	**3x F1 x3**
ETHERNET	Change the ethernet parameters	**F2**
SERIAL	Change the serial parameters	**2x F2**
CTRL MOTOR	Control the motor manually	**F3**
DIVERS CONFIG	Change other parameters (field unit, sound...)	**F4**
MFC INFOS	Display MFC / MFM information's	**2x F4**

c) Home page

The home page displays the overall system status. It's the startup page. You can access it from other menus with the « **ESCAPE** » key (red).

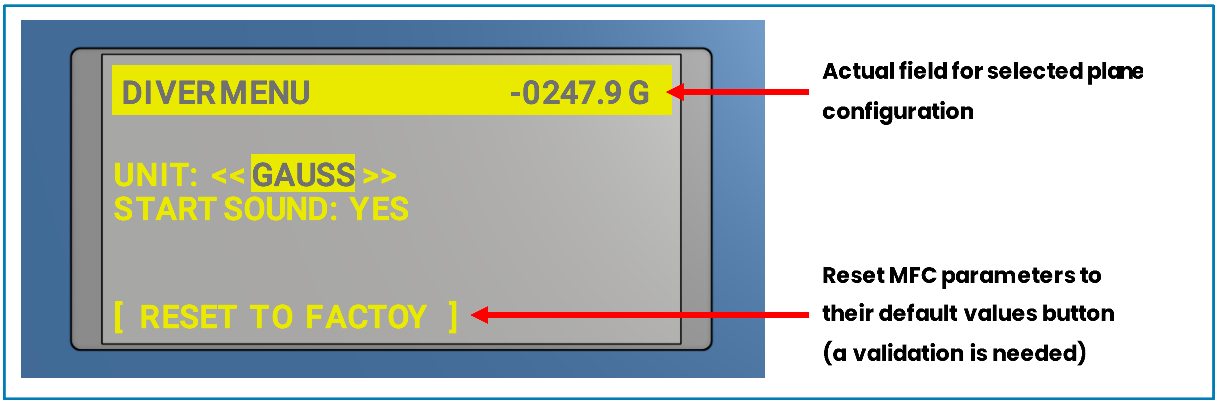


Oled screen view of the home page on MFC front panel

d) Diver menu

To access the "DIVER MENU", press the « **F4** » key. On this menu you can change the MFC display unit (Gauss / mTesla / Tesla), enable or disable the starting sound of the system and reset the various parameter of the MFC to their default value.

NOTE: the user magnetic field unit is only for screen display of the MFC. The remote commands and their returned values are in Gauss.



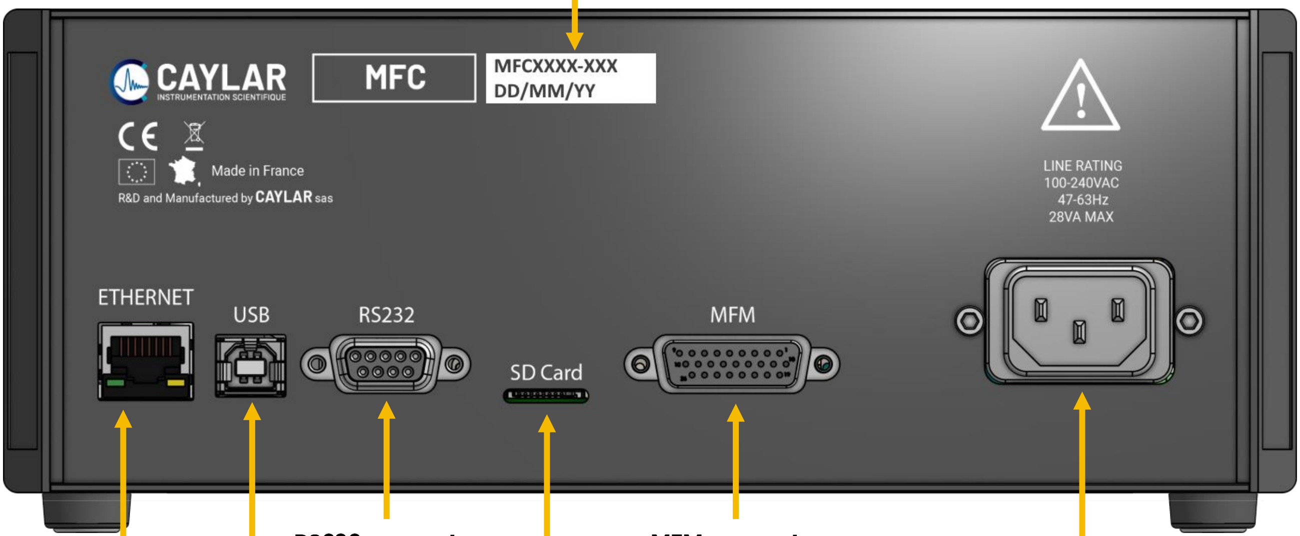
Oled screen view of DIVER MENU on MFC front panel

e) Infos menu

To access the "INFOS MENU", press the **«F4»** key twice. This menu shows you the serial number of the MFM / MFC and the program date and revision.

Oled screen view of INFOS MENU on MFC front panel

f) Rear panel – Communication connectors

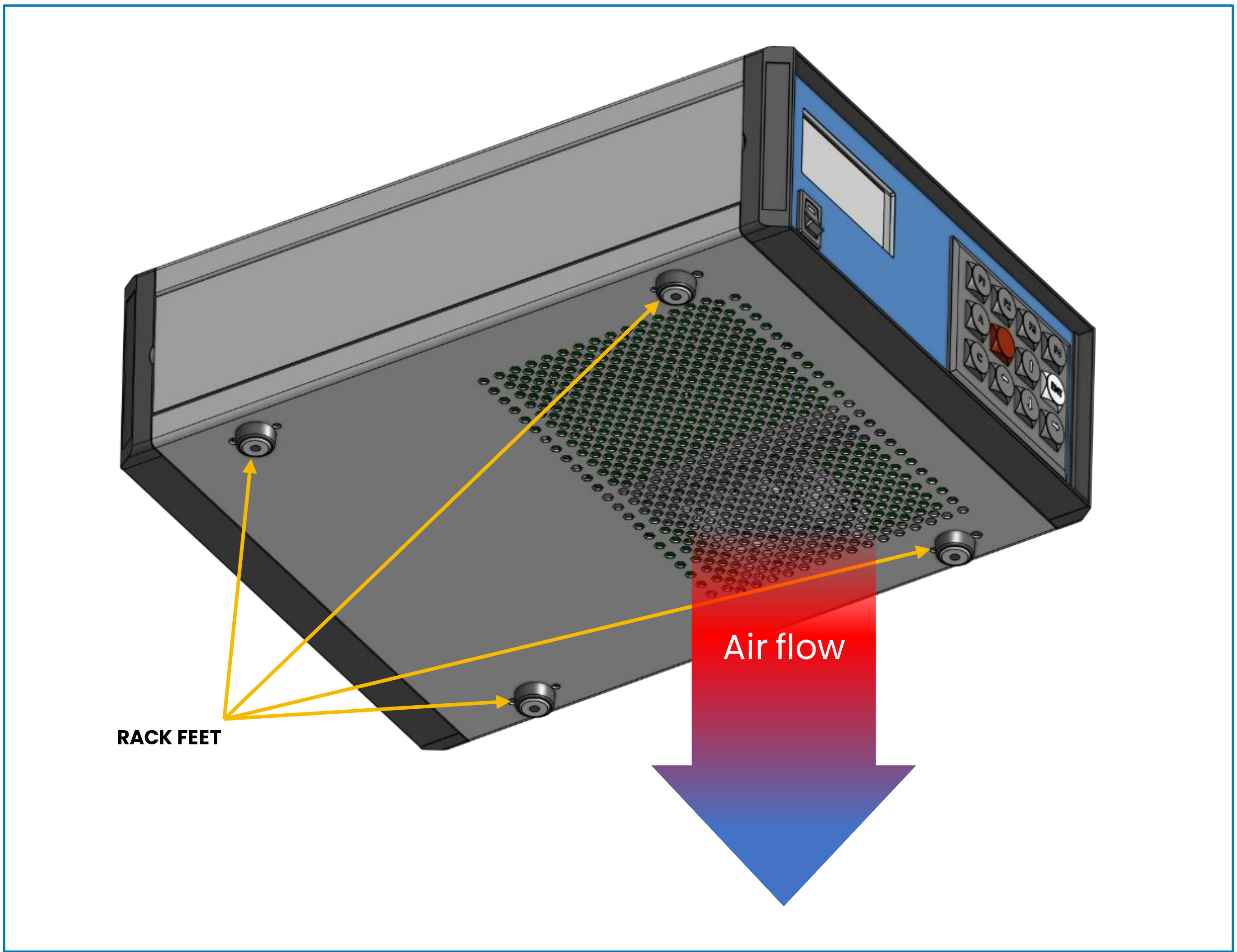


MFC rear panel view

NOTE: the SD card is used to save the user configuration and calibration datas. It is also used for program update. Don't remove or modify without instructions from Caylar.

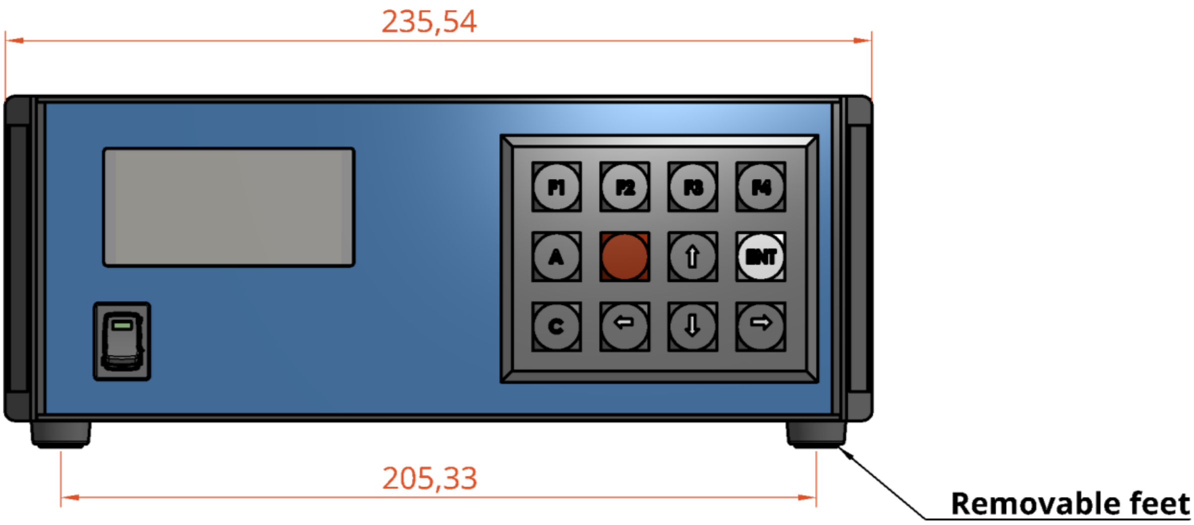
g) Rack Air flow

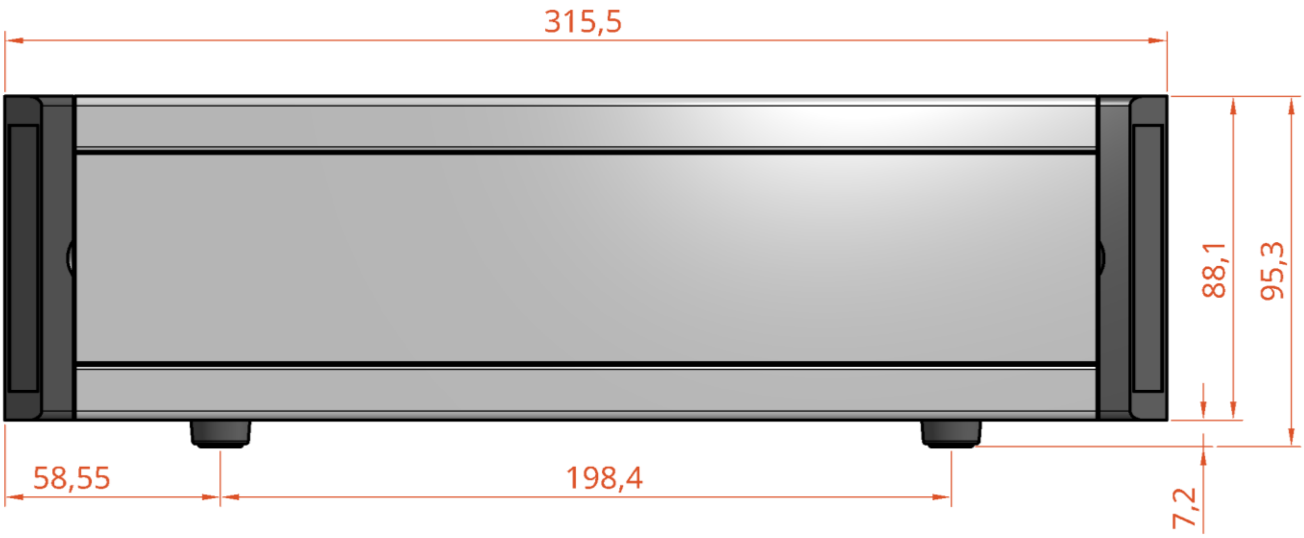
The MFC include a fan which extracts the internal hot air through the bottom grid of the rack. The rack can be placed on a flat surface. Its feet allow it to have a sufficient space with the holding surface and the grid so the air can move freely. If the rack feet are removed, be careful to leave the fan outlet free enough to allow sufficient air flow.



MFC bottom view – Air flow

h) Rack dimensions





i) Calibration process

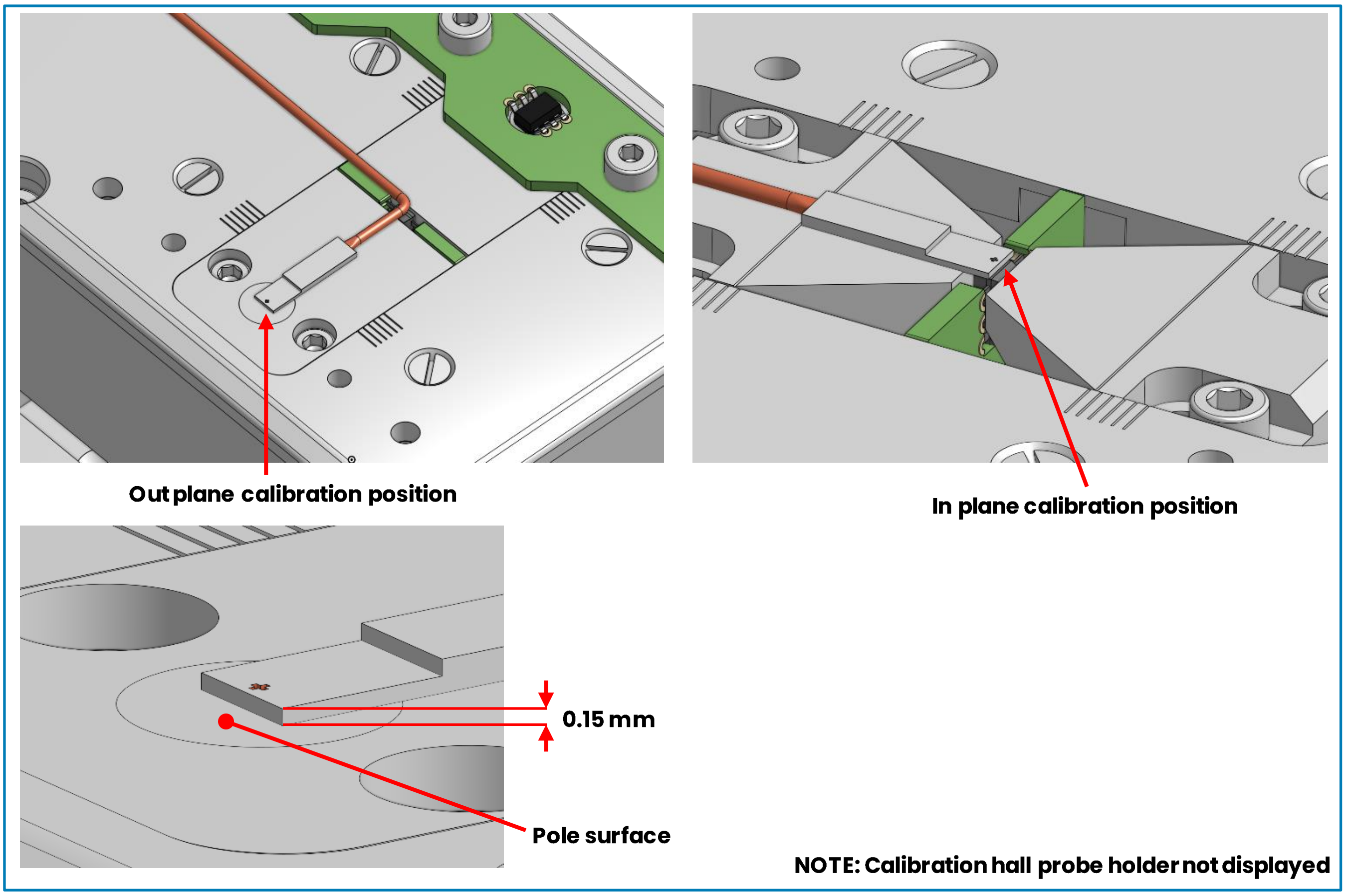
The MFM and MFC are calibrated together. So a MFM can be used only with his associated MFC.

We calibrate the module for both in and out of plane configurations. An external reference hall sensor placed at the position of user sample is used. We choose an ultra-thin hall probe in order to measure the field rely close to the module poles surface where the user sample surface is.

The calibration hall probe sensor is located at **0.15 mm** from the poles surface for in and out of plane configuration.

For the in-plane configuration, the probe is placed at the center of the two poles. For the outplane configuration it is placed at the center of the user poles. Images of the calibration positions are given on the figure bellow.

NOTE: Because the calibration hall probe isn't at the same place as the MFM hall probe (and so the user sample), there are some hysteresis errors that affect the calibration. To minimize these errors, we made two different calibrations. The first one goes from the max negative field to the max positive field and the second one goes from the max positive field to the max negative field. At the end we made an average of these two calibrations which is a compromise between the two calibrations, minimizing hysteresis errors.



Calibration hall probe position for IN and OUT of plane configuration

After the calibration we keep the calibration hall probe in place and we test the system during more than 8 hours for each module. We test random setpoint each 5min to check the calibration performance. Calibration datas and test datas are available for each module.

Magnetic Field Module (MFM) Presentation

The MFM is the field generator placed on the microscope just under the user sample.

It uses a bipolar permanent magnet connected to a stepper motor to generate the variable magnetic field. The magnet is placed at the middle of a ferromagnetic core which have a U-shape. At the extremities of the U-shape are placed two matched poles. There is two different shape of poles depending of the field orientation configuration: IN or OUT of plane. The rest of the module structure is composed of non-magnetic parts.

To vary the magnetic field, the permanent magnet is rotated by the stepper motor to orient more or less its lines of magnetic flux with the ferromagnetic core.

The produced magnetic field is read back by an internal hall sensor placed on one of the MFM poles. It permits to achieve a closed loop control of the magnetic field.

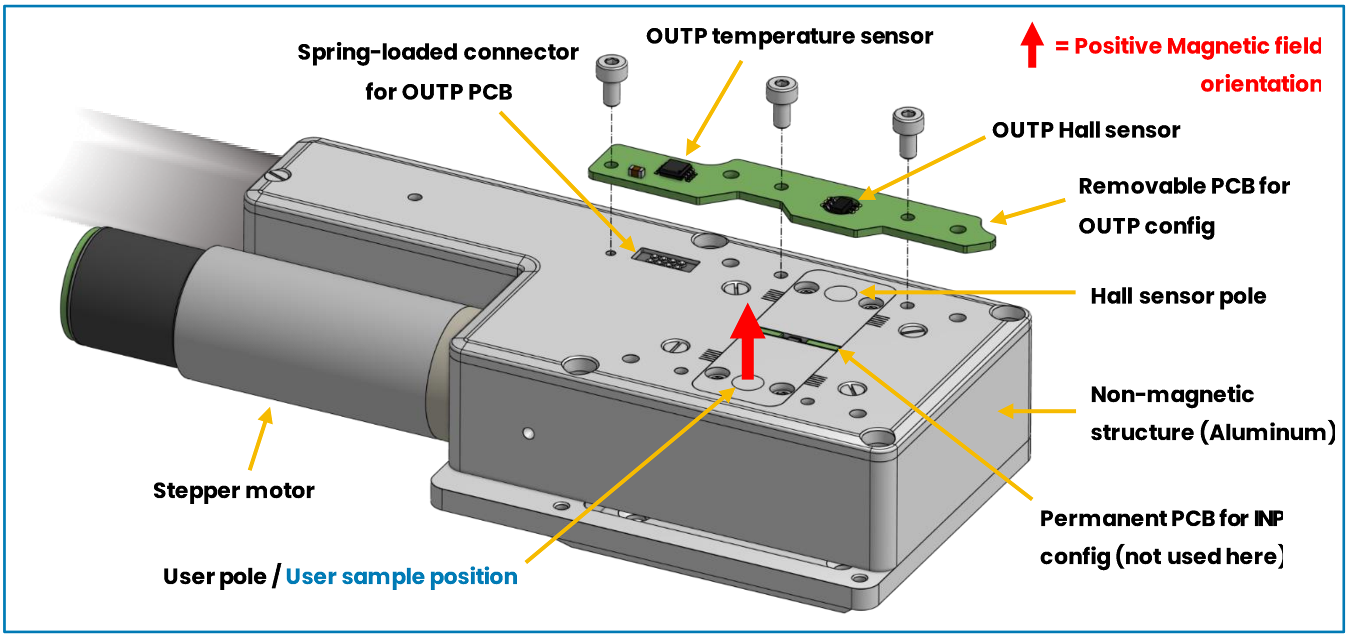
The module also contains two temperature sensors for in-plane and out of plane configurations in order to monitoring the temperature variations that can affect the magnetic field intensity. **NOTE:** No temperature calibration is performed by Caylar. The temperature is only given as an indication to the user.

The module includes its connection cable to the controller. The cable length is ~ 2.5 meters.

a) Out of plane configuration (OUTP)

For the out of plane configuration (OUTP), the generated magnetic field is oriented perpendicular to the user sample surface.

The user sample is placed on the user pole surface and the magnetic field is read back on the over pole which have a hall sensor placed on its surface.



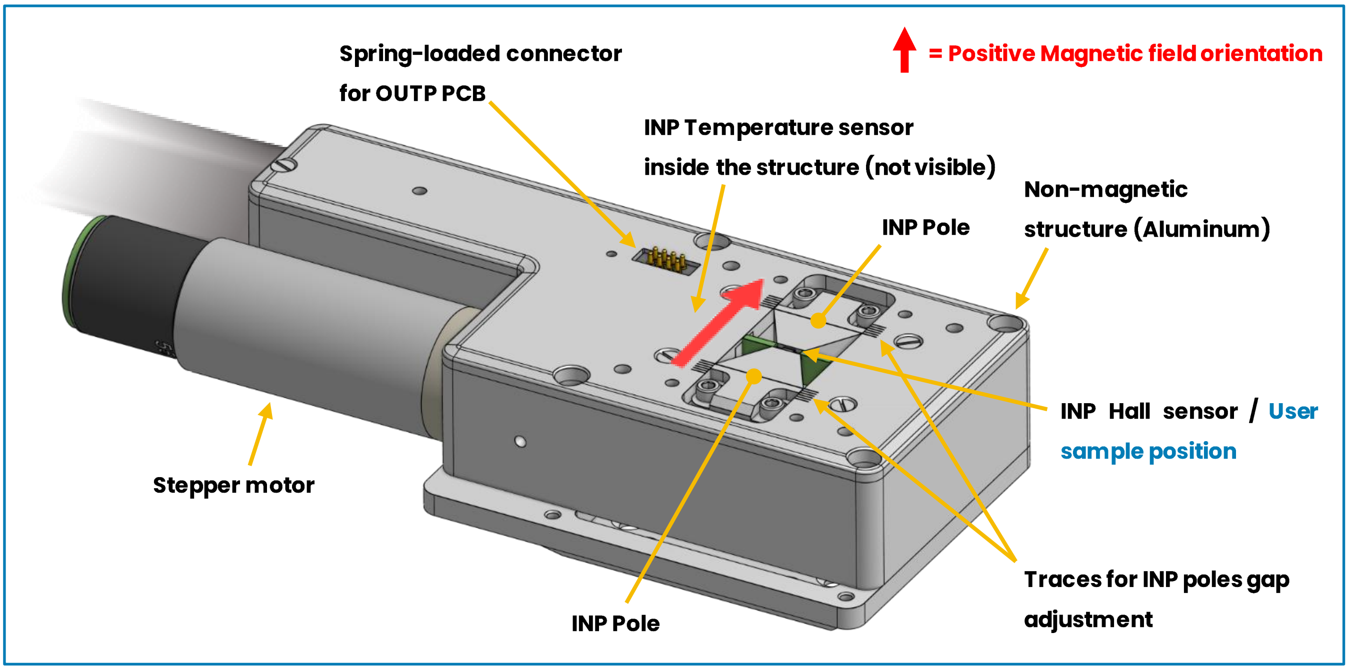
MFM out of plane configuration – Parts presentation

NOTE: For this configuration, the hall sensor and temperature sensor are placed on a removable PCB in order to easily change the poles configuration. The in-plane sensors are permanently present even in out of plane configuration.

b) In-plane configuration (INP)

For the in-plane configuration (INP), the generated magnetic field is oriented within the user sample surface.

The user sample is placed at the center of the two in plane pole, just above the in-plane hall sensor.

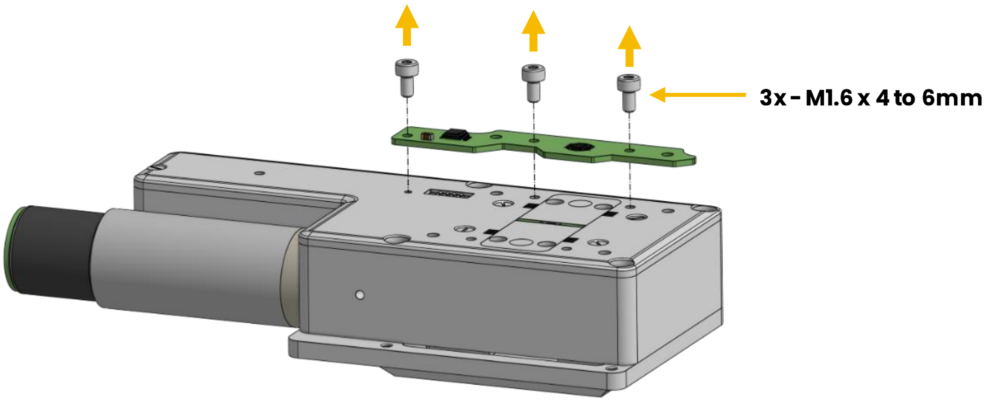


MFM in-plane configuration – Parts presentation

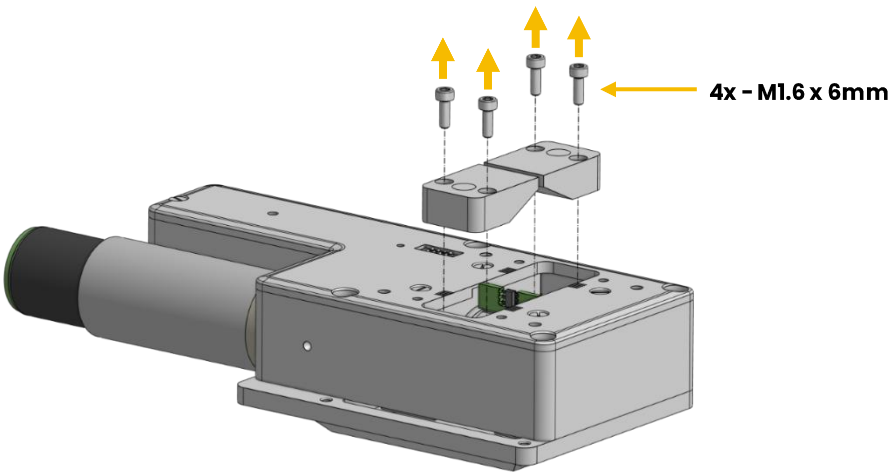
c) Change module configuration out to in-plane.

Step 1: Turn on the MFC with the MFM connected to it and set the out of plane magnetic field to zero in order to minimize the attraction force on polar parts (especially the poles we want to change). Then turn off the MFC.

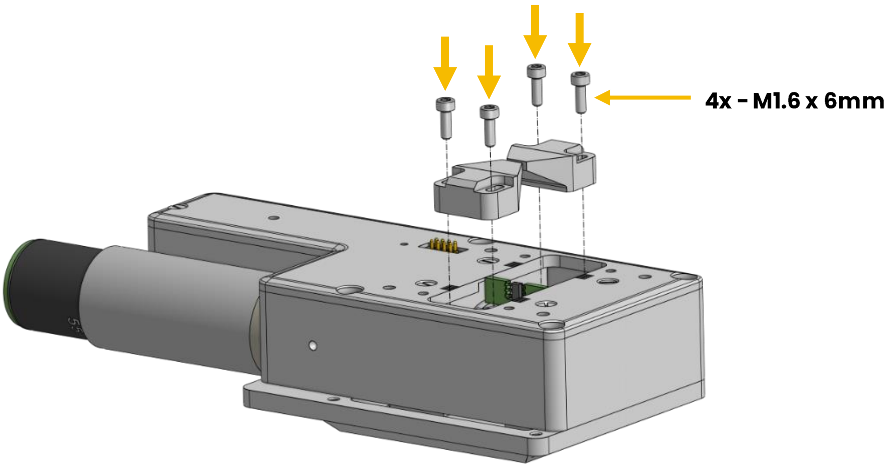
Step 2: Remove the out of plane sensors pcb by unscrew the three pcb screws. Be careful to unscrew the three screws at the same time to minimize the bending of the pcb due to the spring-loaded connector. Do not unscrew the screws one by one.



Step 3: Remove the two out of plane poles by unscrew the two screws for each pole.

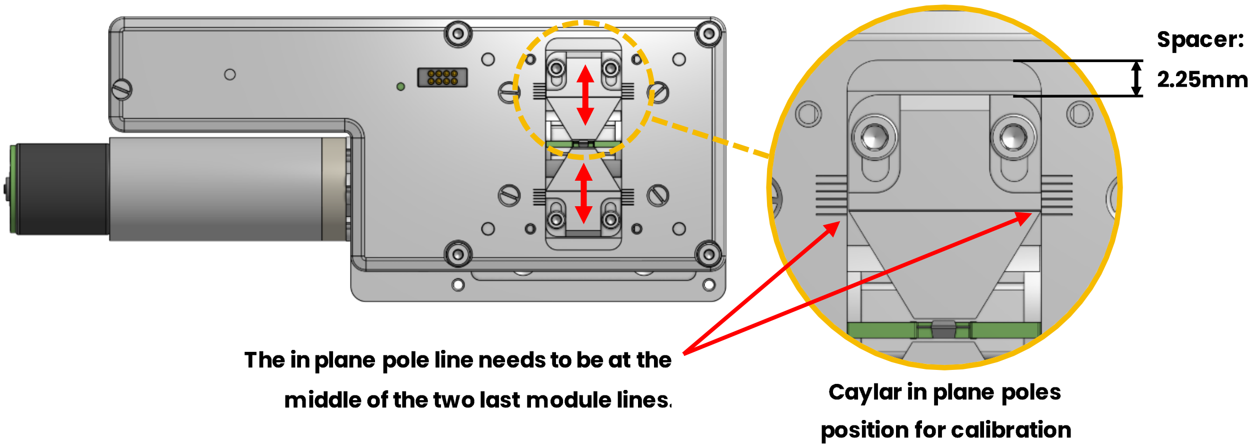


Step 4: Insert the two in plane poles on the module with the four screws. Tighten the screws but don't overtighten them. Be very careful when inserting the poles, do not touch the center pcb with the hall sensor.

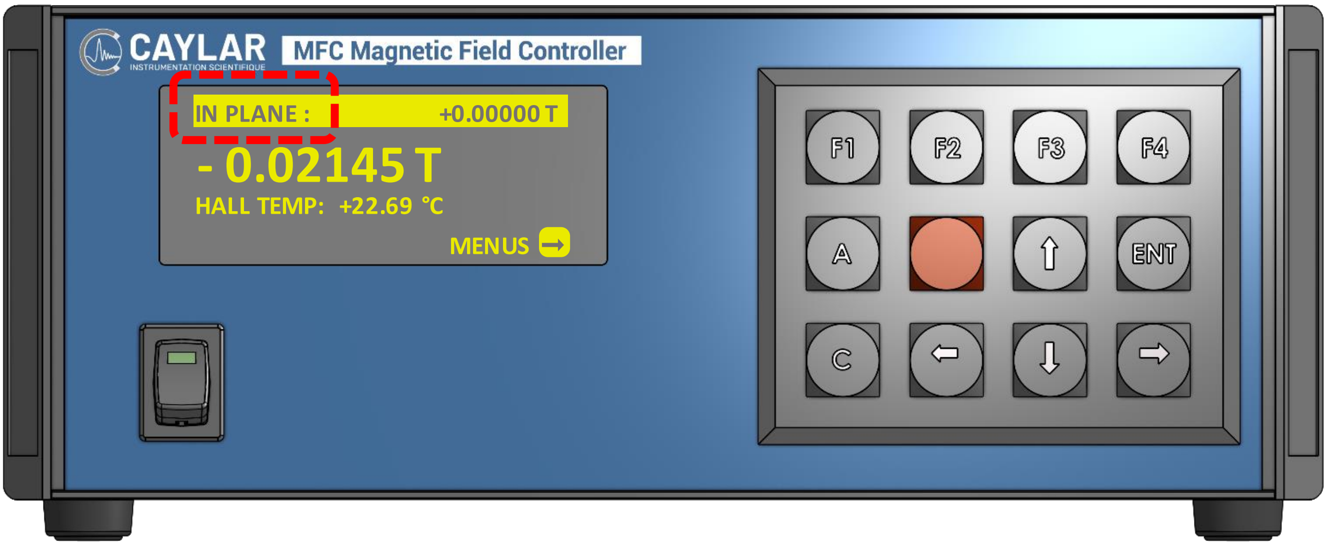


<u>Step 5:</u> The in plane poles gap is adjustable so before tightening the screws adjust the position of the module poles using the marks present on the module structure and poles (see image below). The calibration is made for the position given below which corresponds to a poles gap of 1.5mm.

Tips: you can use shim (spacer) of 2.25 mm between the back of the in plane pole and the structure to adjust precisely the poles position.



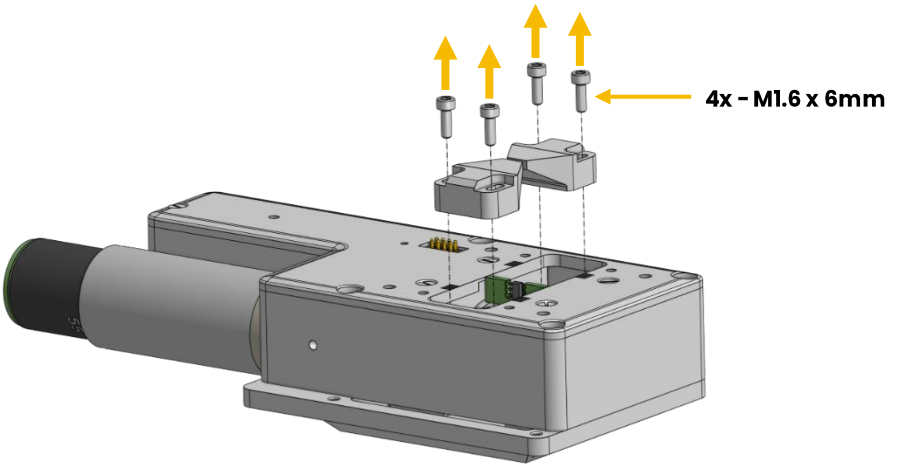
<u>Step 6:</u> Power on the MFC. After the initialization you will receive this message from the controller: "SAVED CONFIG OUTPLANE REGUL ERROR. NO HALL SENSOR DETECT. AUTO CHANGE TO IN PLANE.". You can also switch the configuration manually on the REGUL MENU of the controller (see manual control section).
NOTE: The actual configuration is displayed on the home page of the MFC screen (see image below) or you can check the configuration by remote commands (see remote control section).



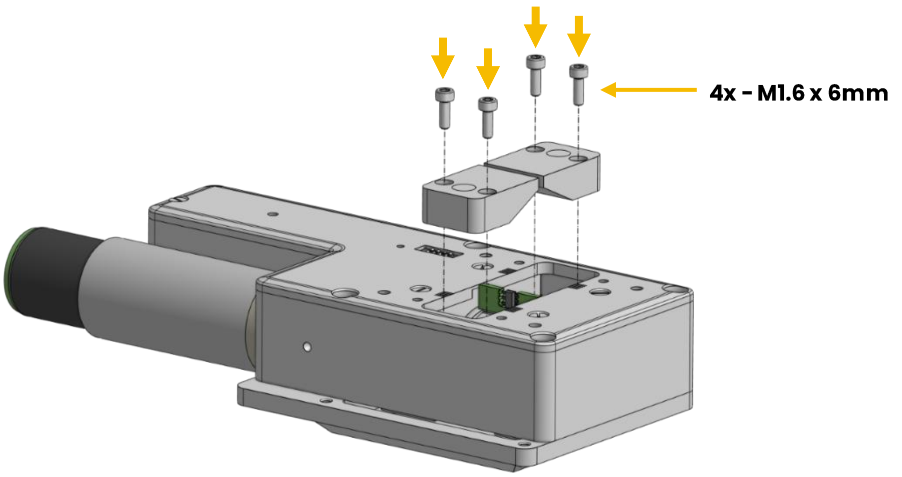
d) Change module configuration in to out-plane.

Step 1: Turn on the MFC with the MFM connected to it and set the out of plane magnetic field to zero in order to minimize the attraction force on polar parts (especially the poles we want to change). Then turn off the MFC.

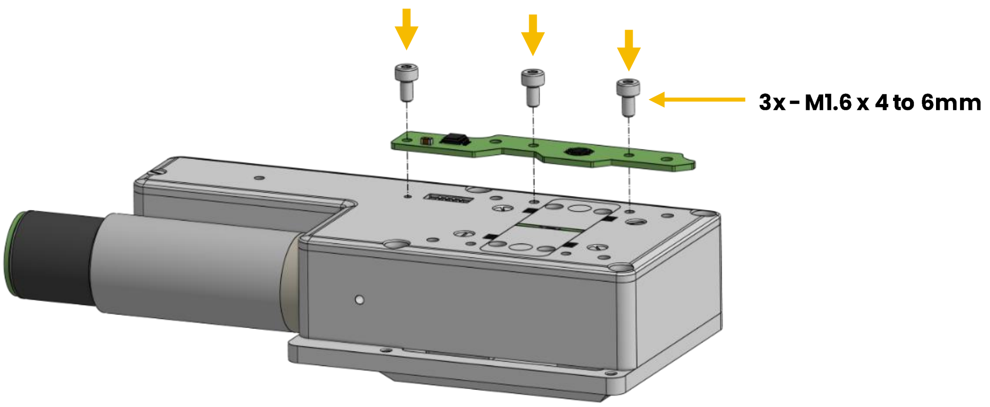
Step 2: Remove the two in plane poles by unscrew the two screws for each pole. Be very careful when inserting the poles, do not touch the center pcb with the hall sensor.



Step 3: Insert the two out of plane poles on the module with the four screws.



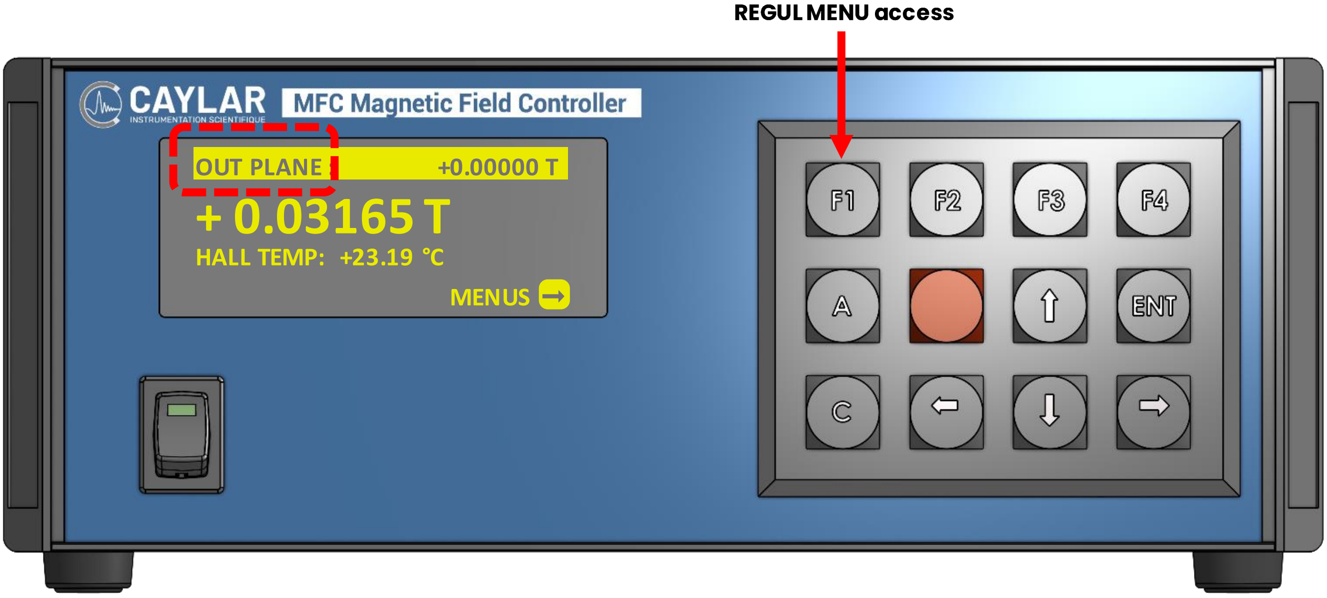
Step 4: Place the out of plane sensors pcb on the module with the three pcb screws. Be careful to screw the three screws at the same time to minimize the bending of the pcb due to the spring-loaded connector. Do not screw the screws one by one.



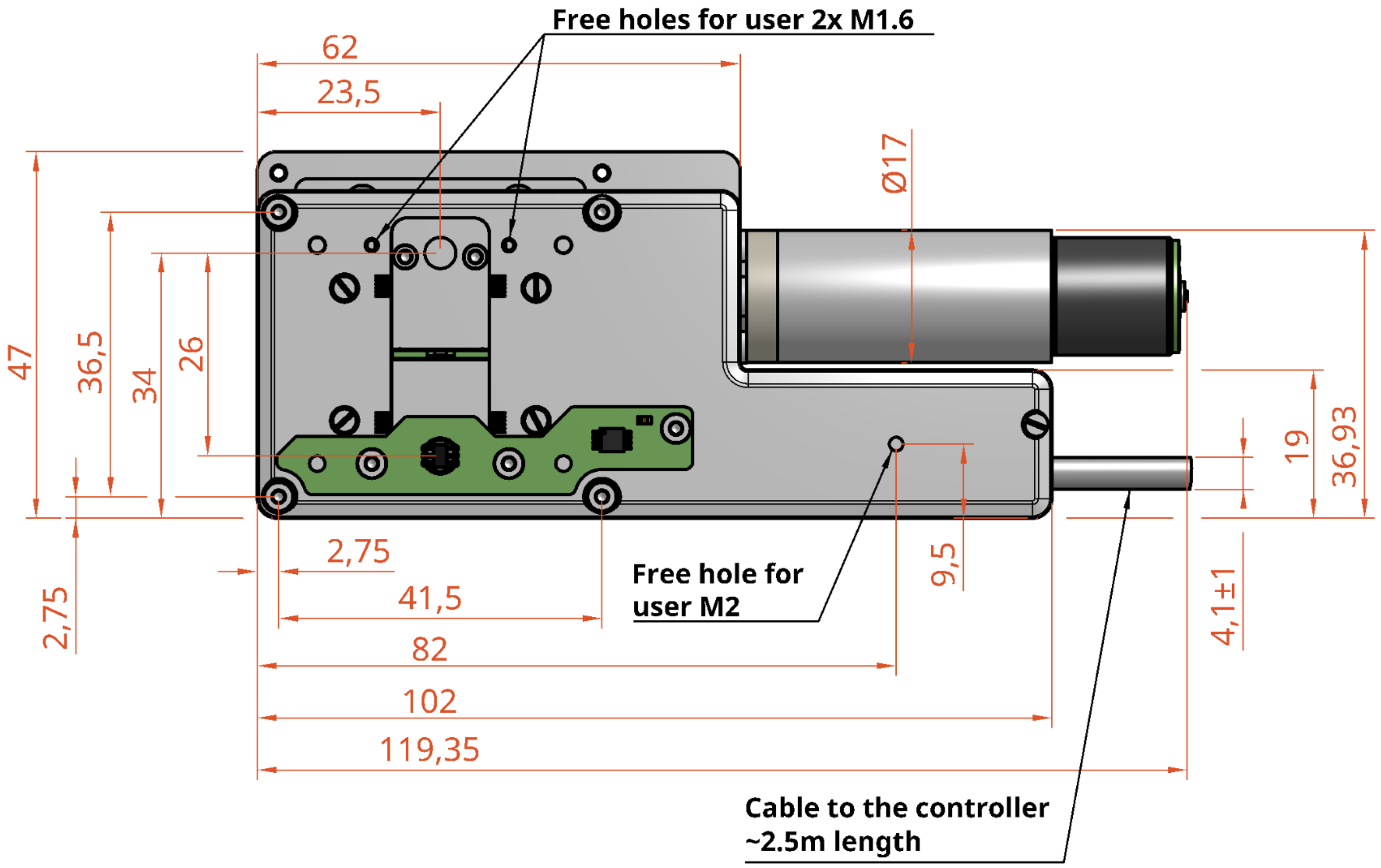
Step 6: Power on the MFC. After the initialization you need to change manually the plane configuration to out of plane. You can do this manually or remotely (see remote control section).

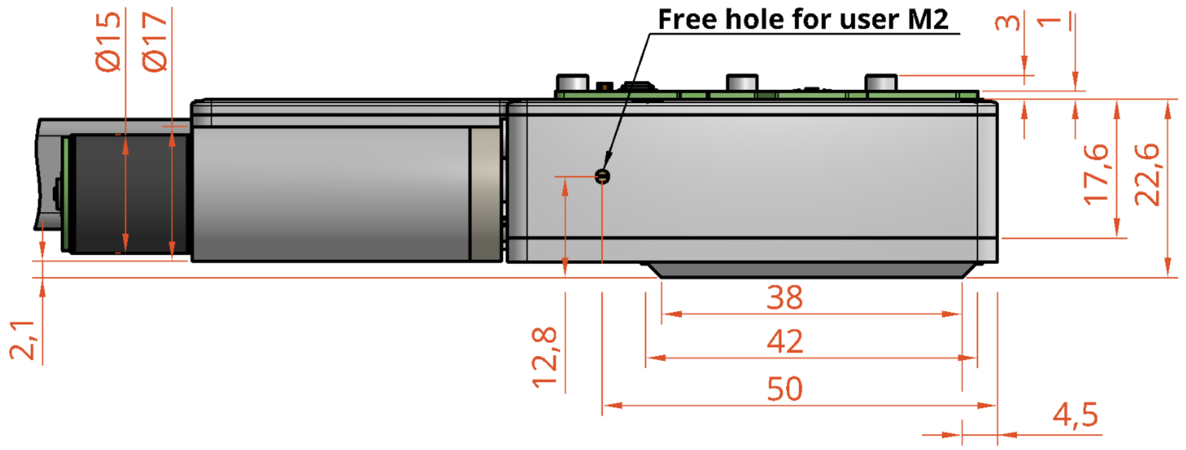
Manual: Enter the REGUL MENU by pressing the **« F1 »** key. Select the regul plane setting by pressing the **«↓»** and **« ENT »** keys. Change the setting to "OUT" by pressing **«→»** and **« ENT »** keys. Then you can exit the menu by pressing the **« ESCAPE »** key (red button). Then you should see on the home page the out-plane mode displayed.

NOTE: The actual configuration is displayed on the home page of the MFC screen (see image below) or you can check the configuration by remote commands (see remote control section).



e) Module dimensions (Out plane configuration)





CAYLAR
INSTRUMENTATION SCIENTIFIQUE

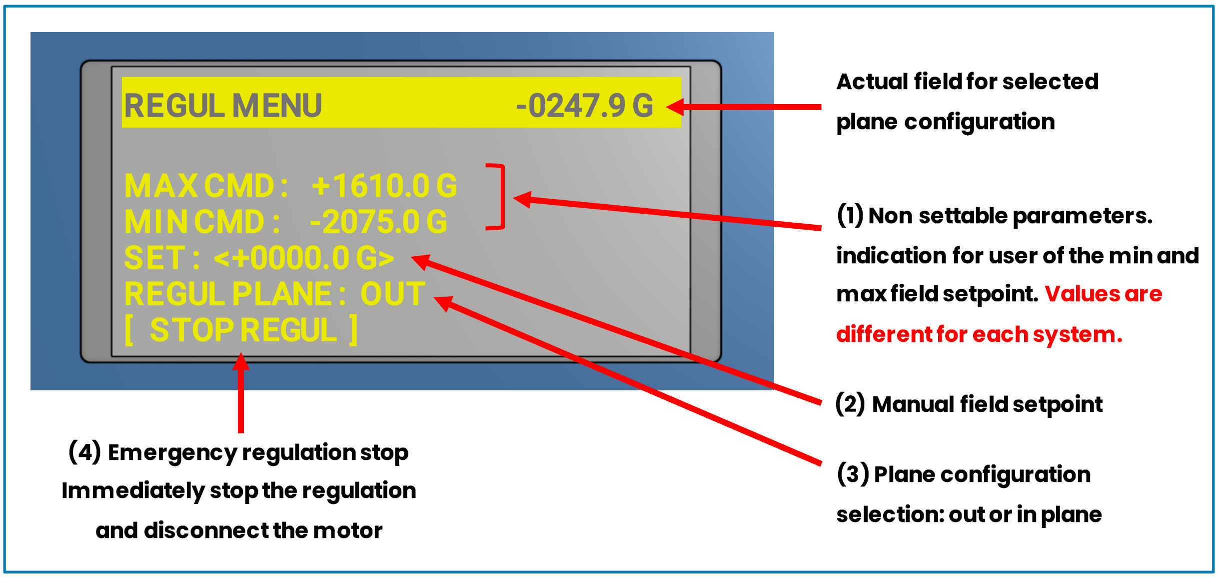
Field regulation operation – Closed loop

a) Main regulation menu – set the field / plane mode

The magnetic field regulation allows the user to set a desired field with an adjustable control of the regulation speed, precision and overshoot. The regulation can be controlled manually or by remote control (see remote control section).

WARNING: Be careful when you often change the field setpoint within short time intervals. When the motor is used it create heat from its driving current. This heat dissipation creates long-term drift of the magnetic field and can affect the user sample too. Temperature sensors are present on the module to monitor the temperature variations. This allows the user to know if the engine is used too often. Our tests show that a setpoint change every 3 minutes does not affect the module significantly.

For manual control press **« F1 »** key to enter the "REGUL MENU":



Oled screen view of REGUL MENU on MFC front panel

For remote control here is a list of remote commands (see remote control section) related to "REGUL MENU" :

(1) GET_INPLANE_MAX_SETPOINT
(1) GET_INPLANE_MIN_SETPOINT
(1) GET_OUTPLANE_MAX_SETPOINT
(1) GET_OUTPLANE_MIN_SETPOINT
(2) SET_FIELD
(2) GET_FIELD
(3) SET_REGUL_MODE_INPLANE
(3) SET_REGUL_MODE_OUTPLANE
(3) GET_REGUL_MODE
(4) SET_REGUL_STOP (Note: to restart the regulation use the SET_FIELD command with the same setpoint)
(4) GET_REGUL_STATE

b) Operating principle – Regulation start / stop conditions:

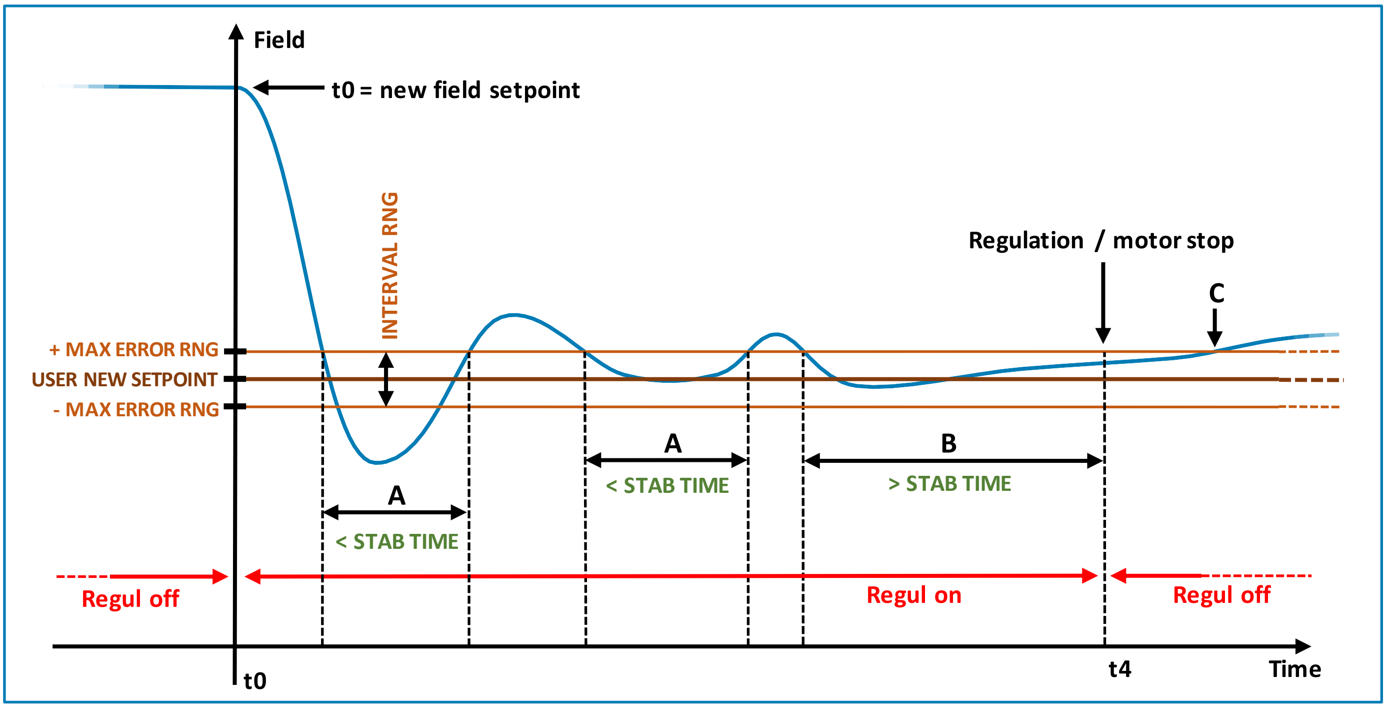
The operating principle of this system field regulation is a little bit different than the conventional PID systems: it doesn't continuously drive the module to keep the field constant. It means that the system doesn't stay in closed loop but it changes from closed loop to open loop for each setpoint change.

When the user requests a new setpoint, the module motor and the regulation loop are activated in order to reach the desired field. When the feedback field enters and stays in a certain interval range around the field setpoint (**INTERVAL RNG**) during a certain period of time (**STAB TIME**), the controller fully disconnects the module motor and stop the regulation loop in order to keep the permanent magnet orientation. It means that the controller assume it has reached and stabilized the user setpoint for enough time.

The purpose of fully disconnect the motor and stop the regulation is to reduce the electrical noise / interferences and heat from the motor that can interfere with user measurements during an atomic force macroscope scan for example and causing magnetic field drift.

The interval range and stabilization time are settable by the user (see remote control section or manual parameter edition present on this section). These parameters are different for the in plane and out of plane configuration.

The figure bellow shows an example of field regulation run :



Example of field regulation error evolution and stop conditions

t0 – The user set a new field setpoint. The module motor and regulation start running and the field start to reach the new value.

Interval A – The regulation didn't manage to maintain the field in the configured interval range around the user setpoint during more than the configured stabilization time. Each time the field exit the interval range, the counter for the stabilization time is resumed to zero and restart at the next interval range re-entry.

Interval B – The regulation manages to maintain the field in the interval range during more than the configured stabilization time so the controller assumes that it has reached and stabilized the user setpoint. The field regulation is stopped and the module motor is fully disconnected from the controller at **t4**.

WARNING: C – The field can continue to drift a little bit for multiple reasons like module mechanical deformations due to magnetic field force applied on mobile part or external/module temperatures variations. So it can exit the interval range after the regulation stop. Since the regulation is stopped, it won't try to restart itself in order to avoid disturbing the user during his measurements. However, the user can monitor the field and restart the magnetic field regulation if desired by applying the same setpoint. The system is designed to minimize this drift and make it lower than its calibration performance for example.

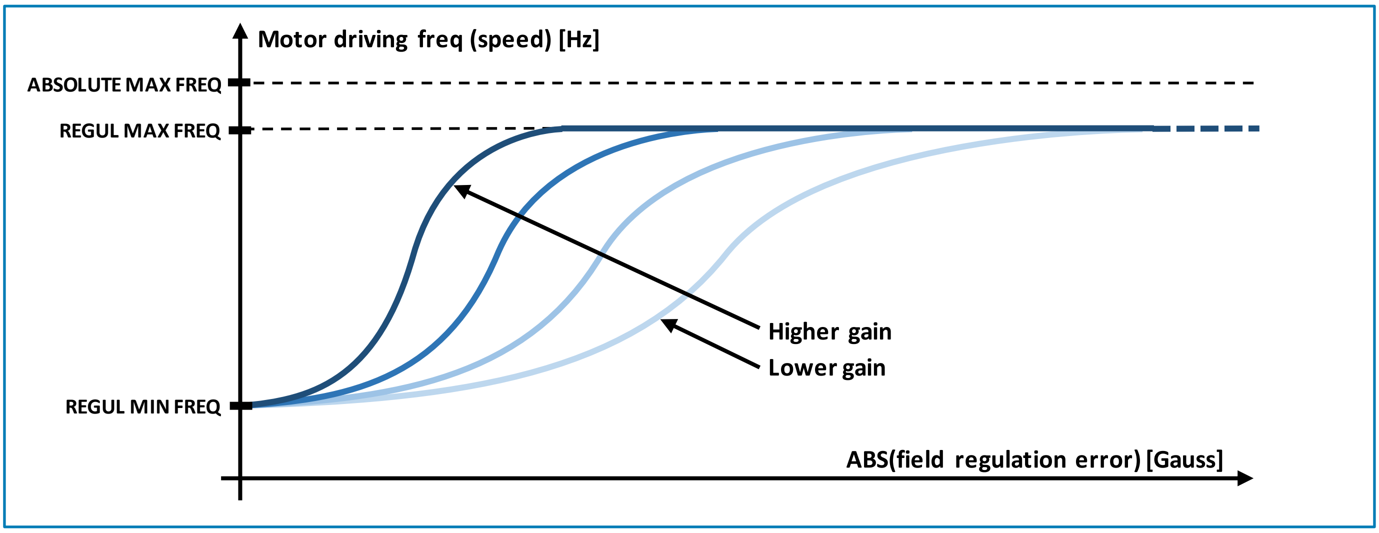
c) Operating principle – Regulation speed:

The regulation characteristics (speed, precision, overshoot) are controlled by three parameters in addition to the stop condition parameters explained in the previous chapter: the regulation min/max motor frequencies and the regulation gain. These parameters are different for in plane and out of plane configuration.

The min and max frequencies are the min and max speed for the motor depending of the regulation error value. The regulation error value is the difference between the actual magnetic field feedback read from the module hall sensor and the user setpoint. Be carful with the min frequency parameter, the module must have enough speed control at low error to be able to fast enough change the field if you want a relatively fast response.

The gain is similar to the proportional coefficient for a PID regulation but with a little bit of dumping around the regulation error extremum. More we increase the gain parameter, faster the regulation will react to a field error by requiring a higher speed to the motor in order to vary the field more quickly. To minimize the overshoot, it is better to lower the value of this parameter.

The figure below explains the effect of these 3 parameters for regulation characteristics:

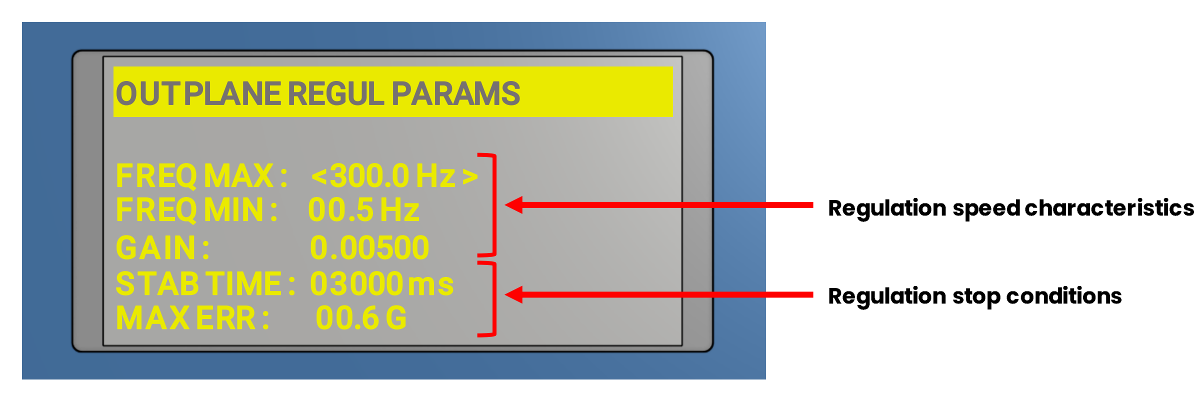


Motor speed vs field error during regulation vs various parameter

d) Regulation Parameters Editing

For manual editing the regulation parameters for the in plane and out of plane configurations are located in two identical menus for each configuration.

Press **« F1 »** key twice for the "OUTPLANE REGUL PARAMS" menu or three times for the "INPLANE REGUL PARAMS" menu:



Oled screen view of OUTPLANE REGUL PARAMS menu on MFC front panel

Caution: you must set the « MAX ERR » parameter correctly to take into account the measurement noise that is not the same over the entire field range. If the amplitude of the measurement noise is greater than the given interval range then the regulation will never be able to stop because the field will always exit the setpoint interval range.

List of regulation parameters and related remote commands:					
Name	**Related remote command (1)**	**min**	**Default**	**max**	**Unit**
Min frequency	GET_REGUL_OUTPLANE_PARAM_MIN_FREQ GET_REGUL_INPLANE_PARAM_MIN_FREQ SET_REGUL_OUTPLANE_PARAM_MIN_FREQ SET_REGUL_INPLANE_PARAM_MIN_FREQ SET_REGUL_PARAM_MIN_FREQ	0	0.5	10	Hz
Max frequency	GET_REGUL_OUTPLANE_PARAM_MAX_FREQ GET_REGUL_INPLANE_PARAM_MAX_FREQ SET_REGUL_OUTPLANE_PARAM_MAX_FREQ SET_REGUL_INPLANE_PARAM_MAX_FREQ SET_REGUL_PARAM_MAX_FREQ	0	300.0	350	Hz
Gain	GET_REGUL_OUTPLANE_PARAM_GAIN GET_REGUL_INPLANE_PARAM_GAIN SET_REGUL_OUTPLANE_PARAM_GAIN SET_REGUL_INPLANE_PARAM_GAIN SET_REGUL_PARAM_GAIN	0.00001	0.005	0.1	No Unit
Stabilization time	GET_REGUL_OUTPLANE_PARAM_STAB_TIME GET_REGUL_INPLANE_PARAM_STAB_TIME SET_REGUL_OUTPLANE_PARAM_STAB_TIME SET_REGUL_INPLANE_PARAM_STAB_TIME SET_REGUL_PARAM_STAB_TIME	0	3000	99999	ms

Max error range	GET_REGUL_OUTPLANE_PARAM_MAX_ERR GET_REGUL_INPLANE_PARAM_MAX_ERR SET_REGUL_OUTPLANE_PARAM_MAX_ERR SET_REGUL_INPLANE_PARAM_MAX_ERR SET_REGUL_PARAM_MAX_ERR	0	0.6	99.9	G

(1) See remote control section

Motor operation – Open loop

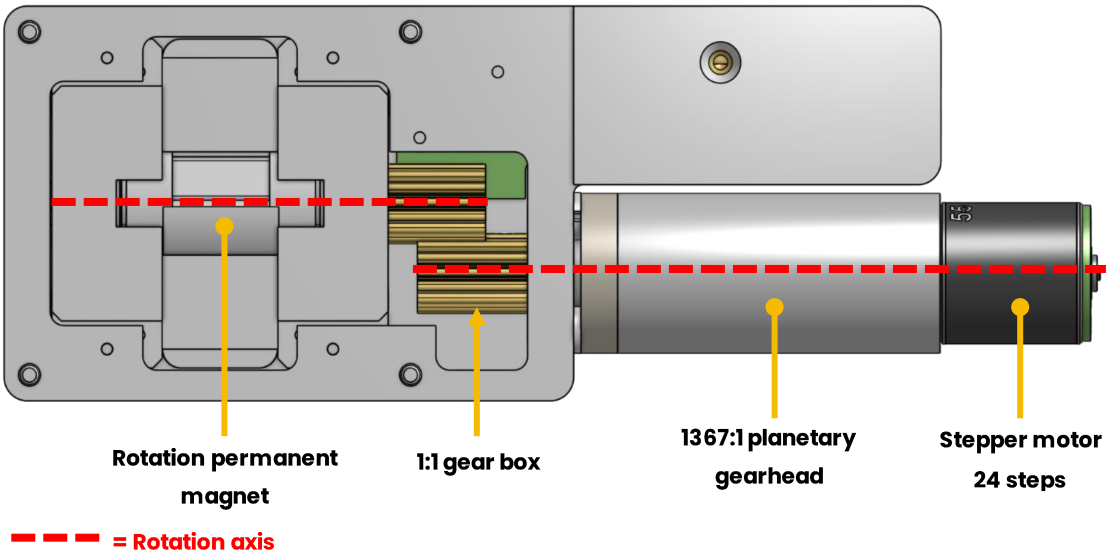
a) Introduction

The system is designed to be scalable and fully open to the user. The user can therefore control the MFM motor by himself to create his own regulation for example.

WARNING: When the motor is used, it generates a heating effect due to its driving current. This heat dissipation creates long-term drift of the magnetic field and can affect the user sample to. Temperature sensors are present on the module to monitor the temperature variations. This allows the user to know if the engine is used too often. Our tests show that using the motor 1 min at full speed every 4min does not affect the module significantly.

The motor is controlled by three parameters:

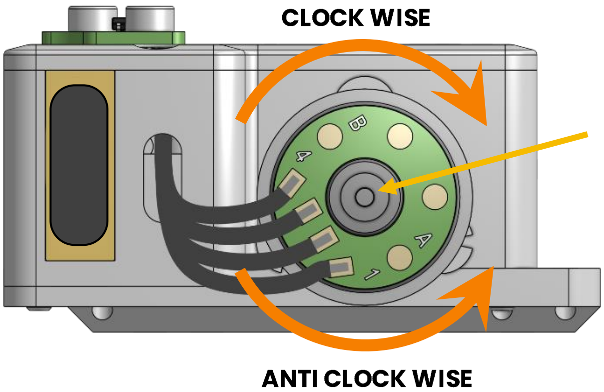
- **Speed:** The speed corresponds to the stepper motor steps frequency in Hz. The motor has 24 steps per turn. It is connected to a 1367:1 planetary gearhead reducer. This reducer is connected to the permanent magnet thanks to 1:1 gear box. So a full turn of the permanent magnet inside the module correspond to 24 * 1367 * 1 = 32808 steps.



The equation bellow gives the conversion between stepper motor steps speed in Hz and permanent magnet speed in turn/minutes:

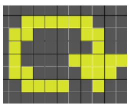
$$magnet\ rotation\ speed\left[\frac{turn}{min}\right] = \frac{32\ 808}{motor\ \ freq\ [Hz] \times 60}$$

- **Rotation side:** define the rotation side of the stepper motor view from the back of the motor.
  WARNING: the rotation side does not give the direction of field variation! During a full turn of the permanent magnet in the same direction, the field increases and decreases (or vice versa) between the minimum and maximum field! If you use the controller field regulation, we use the part of the field curve where the clockwise side decreases the magnetic field and vice versa.



The direction and rotation speed of the motor can be observed from the rear washer of the motor. Tip: draw a line with a marker on the washer to better see its rotation.

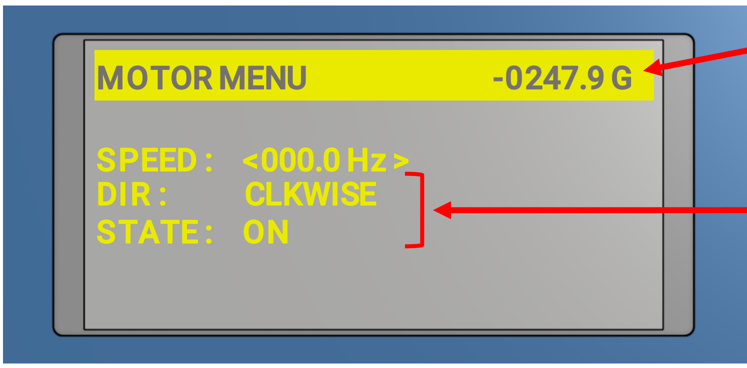
- **Motor state:** The state defines if the module motor is disable (fully disconnected from the MFC) or enable. When enabled, the motor start to turn if the configured speed is superior to 0Hz.
  Note: you can enable the motor and set a speed of zero to stop the motor but the current will continue to flow thought the motor. This creates heat that can affect the user sample and the permanent magnet of the MFM by create long-term magnetic field drift. It is hardly recommended to disable the motor when you don't want to change the module magnetic field.



Note 2: When the motor is enabled, a dedicated icon appears on the Home page (see home page section). It is removed when the motor is disable. The icon is static for manual motor control and dynamic (icon turning) when the field regulation drives the motor (closed loop).

b) Motor parameters editing

For manual motor control press the **« F3 »** key to enter the "MOTOR MENU":



Actual field for selected plane configuration

On the fly configurable parameters: when you the select the parameter, you can directly edit it with «←», «→» keys, you don't need to press the enter button first.

Oled screen view of MOTOR MENU on MFC front panel

List of motor parameters and related remote commands:					
Name	**Related remote command (1)**	**Values**			
Speed	SET_MOTOR_FREQ GET_MOTOR_FREQ	**min**	**Default**	**max**	**Unit**
		000.0	000.0	350.0	Hz
Direction (Rotation side)	SET_MOTOR_DIR GET_MOTOR_DIR	Clock wise / Anti clock wise			
State	SET_MOTOR_STOP SET_MOTOR_START GET_MOTOR_STATE	ON / OFF			

(1) See remote control section

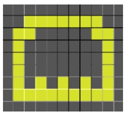
REMOTE CONTROL (Ethernet / Serial)

a) Ethernet introduction / menu

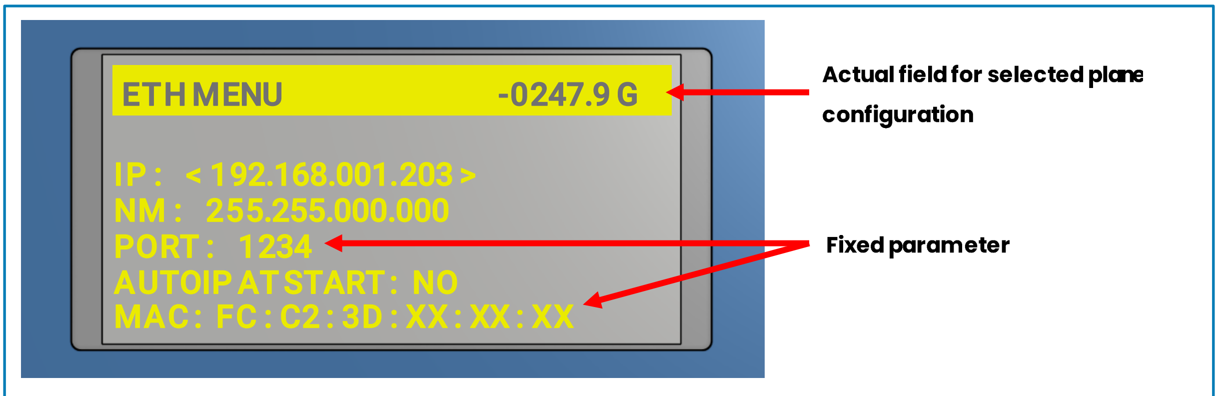
The system can be fully controlled via a sockets TCP/IP communication protocol from the rear RJ45 ethernet connector (see rear panel section of this documentation). The communication port is 1234.

To configure the IP and mask go to the "ETH MENU" by pressing **«F2»** key. You can do it manually or set the "AUTOIP AT START" parameter to "YES" and restart the controller. A DHCP request will be send at the startup to find IP and MASK automatically. Note: Remember to set the parameter to "NO" if the IP and mask should no longer be found automatically at startup and keep the actual settings.

The system manages the **simultaneous connection of 4 sockets maximum**.



Each time an ethernet socket is open, a dedicated icon appears on the Home page (see home page section). It is removed when the connection with the socket ends.



Oled screen view of DIVER MENU on MFC front panel

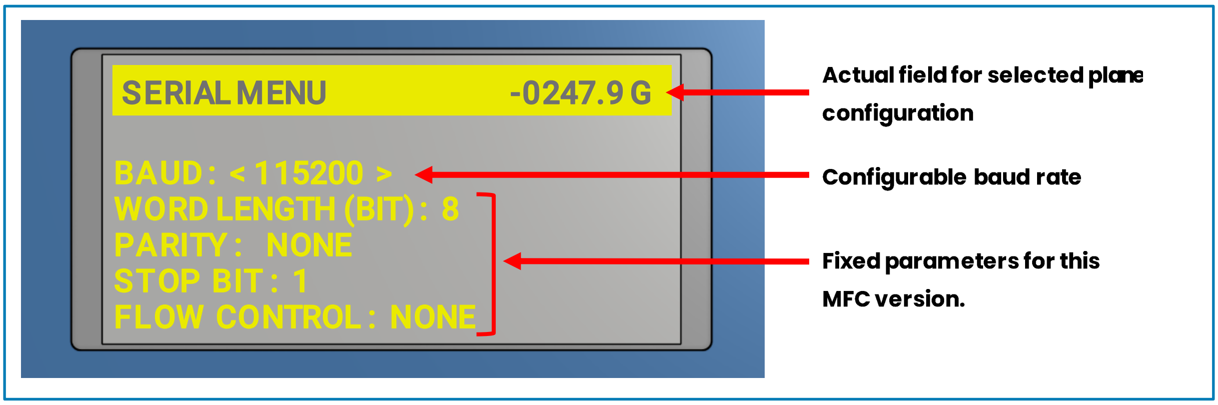
b) Serial configuration / menu

The system can be fully controlled via serial communication protocol from the rear RS232 or USB connectors (see rear panel section of this documentation).

The USB connector is connected to an USB to serial UART interface (FT234XD). The MFC appear as a normal COM port on your PC and you just have to open a serial communication via this COM port.

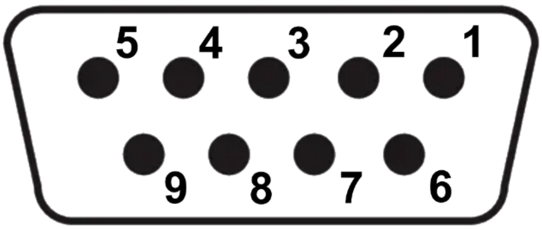
IMPORTANT NOTE: The two RS232 and USB connectors work together and their drivers are internally connected to the same serial uart communication with the MFC microprocessor. It means that when the MFC sends a command response, it is sent at the same time on the two communication channels (both connectors). It also means that these two communication channels cannot be used at the same time as it would cause collisions between received commands.

To configure the serial communication parameters, go to the "SERIAL MENU" by pressing **«F2»** key twice.



Oled screen view of DIVER MENU on MFC front panel

RS232 Sub-D connector pinning:



1 – N.C
2 - RX
3 - TX
4 – N.C
5 – 0V (floating from rack ground)
6 – N.C
7 - RTS
8 - CTS
9 – N.C

c) Sending / receive command

Sending commands:

Ethernet commands must be sent as a string (array of byte) in ASCII format with an « \n », « \r\n » or « \r » character at the end of each order sent. The string must respect upper/lower characters and spaces.

Multiple commands can be sent in one time or a single command can be sent in multiple parts. Receiving one of the end-of-string characters « \n », « \r\n » or « \r » results in processing the received bytes stored in a 1024-byte buffer.

If a command requires additional arguments, these must be separated by a single space between each argument. Example: in the command "SET_FIELD +100.5\n", the argument +100.5 is separated from the command SET_FIELD by a single space.

Response to a command:

The MFC respond for each command received. Feedback responses are returned as strings in ascii format with an end-of-line character (LF or « \n ») at the end of each response.

For « SET » commands that have the format « SET_XXX » (where « XXX » is the command), the start of the response always corresponds to the copy of the command « SET_XXX » sent with an « _OK » added if the command was executed without problem. Example: for the command « SET_REGUL_MODE_INPLANE » the response is « SET_REGUL_MODE_INPLANE_OK ». In case of errors during the command processing, the « _OK » confirmation is replaced by « _ERROR » followed by an argument which explain the error source. For example, if « SET_REGUL_MODE_OUTPLANE » command is sent while the module is in inplane configuration, the controller will returns « SET_REGUL_MODE_OUTPLANE_ERROR BAD_SENSOR_CONFIG».

For « GET » commands type that have the format « GET_XXX » (where « XXX » is the name of the value to get), the start of the response always corresponds to the copy of the name of the value to get « XXX » sent with an « = » added. Example: for the command « GET_FIELD » the response is « FIELD=-309.58 G »

In case of unrecognized commands, the power supply will return the following response: « WRONGCOMMAND\n ».

In the case of a return of multiple arguments, these are separated by a single space between them.

Some « GET » commands return the value with the unit as additional argument after the value like the example of « GET_FIELD » which return in addition to the field value

IMPORTANT NOTE :

- It is important to check the response of the MFC for each command sent to ensure that it has been processed correctly and to avoid MFC microcontroller saturation in the case of commands sent too quickly at the same time.
- Some commands take time to run, so the MFC response may take few seconds to arrive.
- All commands and responses are in Gauss independently of the unit configured for the MFC display.

d) List of Commands

*IDN?

Response : **CAYLAR_MFC<serial_number>**	Ex: CAYLAR_MFCXXXX-XXX

Returns the unique serial number of the MFC.

<serial_number> is a string corresponding to the serial number. This serial number is unique and specific to CAYLAR products.

GET_FIELD

Response : **FIELD= <field> G**	Ex: FIELD= +100.17 G

Returns the last measurement of the field made by the MFC hall sensor which is used for the field regulation. The field is measured approximately every second depending on options and settings. The field is returned in Gauss.

<field> format is « %+.2 lf ».

- The returned field come from the actual used plane sensor for the field regulation (out or in plane), see the command « GET_REGUL_MODE » to know the actual used plan where the measure comes from.
- Related commands: « SET_FIELD », « GET_OUTPLANE_FIELD », « GET_INPLANE_FIELD ».

GET_INPLANE_FIELD

Response : **INPLANE_FIELD= <field> G**	Ex: INPLANE_FIELD= -500.24 G

Returns the last measurement of the inplane field made by the MFC inplane hall sensor. The field is measured approximately every second depending on options and settings. The field is returned in Gauss.

<field> format is « %+.2 lf ».

- Related commands: « SET_FIELD », « GET_OUTPLANE_FIELD », « GET_FIELD ».
- Since the inplane field value isn't updated in outplane mode this command shouldn't be used in outplane mode.

GET_OUTPLANE_FIELD

Response : **OUTPLANE_FIELD= <field> G**	Ex: OUTPLANE_FIELD= +1200.05 G

Returns the last measurement of the outplane field made by the MFC outplane hall sensor. The field is measured approximately every second depending on options and settings. The field is returned in Gauss.

<field> format is « %+.2 lf ».

- Related commands: « SET_FIELD », « GET_INPLANE_FIELD », « GET_FIELD ».
- Since the outplane field value isn't updated in inplane mode this command shouldn't be used in inplane mode.

GET_REGUL_ERROR

Response : **REGUL_ERROR= <error> G**	Ex: REGUL_ERROR= -0.43 G

Returns the actual error of the MFC field regulation. The error formula is : $\mathrm{field}_{measure} - \mathrm{field}_{user\ setpoint}$, where $\mathrm{field}_{measure}$ and $\mathrm{field}_{user\ setpoint}$ depends on the actual regulation mode : inplane or outplane. The error is returned in Gauss.

<error> format is « %+.2 lf ».

- WARNING: The error is calculated from a measurement that is updated only approximately every second.
- Allows to monitor the regulation overshoots.
- Since the regulation no longer try to correct the field once the user's setpoint has been reached, this command allows the user to check the long-term field drift after the regulation has stopped.

GET_REGUL_CMD

Response : **REGUL_CMD= <setpoint> G**	Ex: REGUL_CMD= +1000.00 G

Return the last field setpoint that the MFC received via Ethernet command (see « SET_FIELD » command) or that was configured by the user from the keyboard on the front MFC panel. The field setpoint is returned in Gauss.

<setpoint> format is « %+.2 lf ».

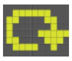
- Related commands : « SET_FIELD » (change the field setpoint).

GET_REGUL_STATE

Response : **REGUL_STATE= <state>**	Ex: REGUL_STATE= 0

Returns whether or not the field regulation is currently active.

<state> is a Boolean corresponding to the field regulation state: 0 = regulation is stopped / 1 = regulation is active (motor is running and field is changing to reach the user-defined setpoint (see « GET_REGUL_CMD » command)).

- When the regulation is active it implies that the motor is switched on. When the regulation has reached the user-defined field setpoint it turns off the motor which become completely disconnected from the MFC rack to avoid electromagnetic interferences around the user sample.
-  This icon appears on OLED screen in front of the MFC rack to indicate that the regulation is active (warning: it also means that the user can manually drive the motor).
- Related commands : « SET_FIELD » (start the regulation), « SET_REGUL_STOP ».

GET_MOTOR_FREQ

Response : **MOTOR_FREQ= <freq> Hz**	Ex: MOTOR_FREQ= +0.0 Hz

Returns the actual motor control frequency which corresponds to the motor/magnet rotation speed. The frequency is returned in Hz. The frequency range is [0.0 to 350.0] Hz.

<freq> format is « %+.1 lf ».

- WARNING : The frequency may be zero (no motor/magnet rotation) but the motor may still be powered. A powered motor continues to cause electromagnetic interferences and heat the field module that can cause long-term field drift.
- Related commands : « SET_MOTOR_FREQ ».

GET_MOTOR_DIR

Response : **MOTOR_DIR= <dir>**	Ex: MOTOR_DIR= 0

Returns the actual motor/magnet rotation direction.

<dir> is a Boolean corresponding to the motor/magnet rotation direction seen from the rear motor axle : 0 = clockwise / 1 = anticlockwise.

- Related commands : « SET_MOTOR_DIR ».

GET_REGUL_MODE

Response : **REGUL_MODE= <mode>**	Ex: REGUL_MODE= 0

Returns the actual plane used by the field regulation.

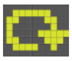
<mode> is a Boolean corresponding to the plane used by the field regulation : 0 = IN_PLANE mode / 1 = OUT_PLANE mode.

- Related commands : « SET_REGUL_MODE_OUTPLANE », « SET_REGUL_MODE_INPLANE ».

GET_MOTOR_STATE	
Response : **MOTOR_STATE= <state>**	Ex: MOTOR_STATE= 0

Returns whether or not the motor is currently enabled (powered on).

<state> is a Boolean corresponding to the motor power state: 0 = motor is disabled (completely disconnected from the MFC rack) / 1 = motor is enabled (powered on).

-  This icon appears on OLED screen in front of the MFC rack to indicate that the motor is enable by the user or the field regulation.
- WARNING : A powered motor cause electromagnetic interferences and heat the field module that can cause long-term field drift. For good field stability performance, do not use it too repeatedly and for too long periods (see « GET_INPLANE_TEMP » and « GET_OUTPLANE_TEMP » command to monitor the field module temperature).
- WARNING 2 : A powered motor doesn't mean that it is rotating if its control frequency is 0 Hz (see « GET_MOTOR_FREQ » command).

GET_INPLANE_TEMP	
Response : **INPLANE_TEMP= <temp> Deg**	Ex: INPLANE_TEMP= +31.20 Deg

Returns the last measurement of the MAX31826 digital temperature sensor placed next to the inplane hall sensor on the same PCB. The temperature is measured approximately every second depending on options and settings. The temperature is returned in degrees Celsius.

<temp> format is « %+.2 lf ».

- Temperature sensor (MAX31826) datasheet link: https://datasheets.maximintegrated.com/en/ds/MAX31826.pdf .
- Since the inplane temperature value isn't updated in outplane mode this command shouldn't be used in outplane mode.

GET_OUTPLANE_TEMP	
Response : **OUTPLANE_TEMP= <temp> Deg**	Ex: OUTPLANE_TEMP= +31.20 Deg

Returns the last measurement of the MAX31826 digital temperature sensor placed next to the outplane hall sensor on the same PCB. The temperature is measured approximately every second depending on options and settings. The temperature is returned in degrees Celsius.

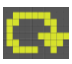
<temp> format is « %+.2 lf ».

- Temperature sensor (MAX31826) datasheet link: https://datasheets.maximintegrated.com/en/ds/MAX31826.pdf .
- Since the outplane temperature value isn't updated in inplane mode this command shouldn't be used in inplane mode.

SET_FIELD <setpoint>	Ex: SET_FIELD 1200.25 / SET_FIELD -120
Response : **SET_FIELD_OK <field> G**	Ex: SET_FIELD_OK +1200.25 G / SET_FIELD_OK -120.00 G
Response in case of unrecognized argument <setpoint> : **SET_FIELD_ERROR BAD_ARG**	
Response in case of overrange in <setpoint> argument : **SET_FIELD_ERROR OVERRANGE**	

Set the field setpoint for the actual field regulation mode (inplane or outplane) and start the regulation to reach the new user setpoint.

<setpoint> format can be double, float or integer / signed or not signed given in gauss.
<field> format is « %+.2 lf ». <field> is the recopy of <setpoint> for the MFC feedback response.

- Related commands : « GET_REGUL_CMD ».
-  While the regulation reaches the new user setpoint this icon appears on the MFC screen main page.

SET_MOTOR_DIR <side>	Ex: SET_MOTOR_DIR 1
Response : **SET_MOTOR_DIR_OK <side>**	Ex: SET_MOTOR_DIR_OK 1
Response in case of unrecognized argument <side> : **SET_MOTOR_DIR_ERROR BAD_ARG**	
Response if field regulation is running : **SET_MOTOR_DIR_ERROR REGUL_RUNNING**	
Set the motor rotation direction. <side> is a Boolean corresponding to the motor/magnet rotation direction seen from the rear motor axis, can be : 0 = clockwise / 1 = anticlockwise. • Related commands : « SET_MOTOR_FREQ », « SET_MOTOR_STOP », « SET_MOTOR_START ».	

SET_UNIT <unit>	Ex: SET_UNIT GAUSS
Response : **SET_UNIT_OK <unit>**	Ex: SET_UNIT_OK GAUSS
Response in case of unrecognized argument <side> : **SET_UNIT _ERROR BAD_ARG or SET_UNIT_ERROR UNKNOWN_UNIT**	
Set the MFC display field unit. <unit> is a string corresponding to MFC front panel display field unit, can be : GAUSS / TESLA / mTESLA. • Only changes the values units on display screen : Does not change the ethernet commands unit and ethernet response.	

SET_REGUL_STOP
Response : **SET_REGUL_STOP_OK**
Turns off the field regulation if it is active. Turn off the motor power and set the motor control frequency to zero.

SET_MOTOR_FREQ <freq>	Ex: SET_MOTOR_FREQ 250.251
Response : **SET_MOTOR_FREQ_OK <freq_feedback>**	Ex: SET_MOTOR_FREQ_OK +250.3
Response in case of unrecognized argument <freq> : **SET_MOTOR_FREQ_ERROR BAD_ARG**	
Response if field regulation is running : **SET_MOTOR_FREQ_ERROR REGUL_RUNNING**	
Response in case of overrange in <freq> argument : **SET_MOTOR_FREQ _ERROR OVERRANGE**	
Set the motor control frequency which corresponds to the motor/magnet rotation speed. <freq> format can be double, float or integer / signed or not signed given in Hertz. Positive number only. The configurable frequency range is [0 - 350.0] Hz. <freq_feedback> format is « %+.1 lf ». <freq_feedback> is the copy of <freq> for the MFC feedback response. • Related commands : « SET_MOTOR_DIR », « SET_MOTOR_STOP », « SET_MOTOR_START ».	

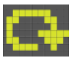
SET_MOTOR_STOP
Response : **SET_MOTOR_STOP_OK**
Response if field regulation is running : **SET_MOTOR_STOP_ERROR REGUL_RUNNING**
Turns off the motor power. • Related commands : « SET_MOTOR_DIR », « SET_MOTOR_FREQ », « SET_MOTOR_START ». • Different from setting the control frequency to zero to stop motor rotation. With this command the motor is powered off while when setting control frequency to zero stop the motor rotation but the motor is still powered. A powered motor continues to cause electromagnetic interferences and heat the field module that can cause long-term field drift.

SET_MOTOR_START

Response : **SET_MOTOR_START_OK**

Response if field regulation is running : **SET_MOTOR_START_ERROR REGUL_RUNNING**

Turns on the motor Power.

- Related commands : « SET_MOTOR_DIR », « SET_MOTOR_FREQ », « SET_MOTOR_STOP ».
- WARNING : Does not necessarily rotate the motor if the control frequency is zero.
-  While the motor is powered on this icon appears on the MFC screen main page.

SET_REGUL_MODE_INPLANE

Response : **SET_REGUL_MODE_INPLANE_OK**

Response if field regulation is running : **SET_REGUL_MODE_INPLANE_ERROR REGUL_RUNNING**

Switch to inplane hall sensor for the field regulation.

- Related commands : « SET_REGUL_MODE_OUTPLANE ».

SET_REGUL_MODE_OUTPLANE

Response : **SET_REGUL_MODE_OUTPLANE_OK**

Response in case of bad module configuration (outplane PCB not mounted on the module) :
SET_REGUL_MODE_OUTPLANE_ERROR BAD_SENSOR_CONFIG

Response if field regulation is running : **SET_REGUL_MODE_OUTPLANE_ERROR REGUL_RUNNING**

Switch to outplane hall sensor for the field regulation.

- Related commands : « SET_REGUL_MODE_INPLANE ».

e) Python ethernet example

This program connects to the MFC module via python socket Ethernet library. It requests the last inplane field measurement made by the module and print the result in the python console.

Screenshot of the program:

```python
#!/usr/bin/env python
# coding: utf-8

"""
Demo code of ethernet communication with the MFC Caylar.
This program connects to the MFC module, ask the actual
inplane field value and print it before disconnecting.
"""

import socket
import traceback

mfc_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

mfc_ip = "192.168.1.232" # remplace this IP by your MFC IP
mfc_port = 1234

try:
    mfc_socket.connect((mfc_ip, mfc_port))
    mfc_socket.send(bytes("GET_INPLANE_FIELD\n", "ascii"))
    inplane_field = float(mfc_socket.recv(50).decode("ascii").split(' ')[1])
    print("Actual MFC inplane field is : " + str(inplane_field) + " Gauss")

except:
    print("Ethernet socket error ...")
    traceback.print_exc()

finally:
    mfc_socket.close()
```

Screenshot of the program output (Console View):

```
IDLE Shell 3.10.1
File  Edit  Shell  Debug  Options  Window  Help
Python 3.10.1 (tags/v3.10.1:2cd268a, Dec  6 2021, 19:10:37) [MSC v.1929 64 bit (
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
================== RESTART: C:/Users/User/Desktop/test_mfc.py ==================
Actual MFC inplane field is : -305.33 Gauss
>>> |
```

Notes