MC-UM-MPM500

Revision number: 11.1.43

Mecademic Robotics

July 17, 2025

Contents

User Manual for the MPM500 Pneumatic Module

For firmware version: 11.1

Document revision: A

Online release date: July 17, 2025

Document ID: MC-UM-MPM500

Original instructions

About this manual

This user manual describes how to install Mecademic's MPM500 pneumatic module onto the Meca500 (R3 & R4) industrial robot arm and how to use it. You must read this manual thoroughly before installing or operating the MPM500.

Symbol definitions

The following table lists the symbols that may be used in Mecademic documents to denote certain conditions. Particular attention must be paid to the warning and danger messages in this manual.

Note

Identifies information that requires special consideration.

Warning

Provides indications that must be respected in order to avoid equipment or work (data) on the system being damaged or lost.

Danger

Provides indications that must be respected in order to avoid a potentially hazardous situation, which could result in injury.

Revision history

The firmware that is installed on Mecademic products has the following numbering convention:

{major}.{minor}.{patch}.{build}

Each Mecademic manual is written for a specific {major}.{minor}.{*}.{*} firmware version. On a regular basis, we revise each manual, adding further information and improving certain explanations. We only provide the latest revision for each {major}.{minor}.{*}.{*} firmware version. Below is a summary of the changes made in each revision.

Revision	Date	Comments
A	March 17, 2025	This version

The document ID for each Mecademic manual in a particular language is the same, regardless of the firmware version and the revision number.

Introduction

The MPM500 is a custom pneumatic module developed by Mecademic, specifically for the Meca500 robot arm. The module essentially consists of two 3/2 NC valves and is easy to install, connect and dismount. The module allows easy integration between the Meca500 and a large variety of single- or double-action pneumatic grippers, suction cups, tool changers, and other pneumatic devices.

Note

Familiarity with the Meca500 robot and its user manual is required prior to installing and using the pneumatic module.

Warning

Improper installation of the pneumatic module could seriously damage the Meca500 and the module itself. This module should therefore be used only by technical personnel who are familiar with the Meca500.

Inside the box

The MPM500 module is supplied either separately or with the Meca500. In both cases, the module, shown in Figure 1, is delivered as a standard kit in an anti-static bag, the content of which is:

- 1 MPM500 module with a MGC-AA25 25 mm communication cable with a 90° connector;
- 3 M5X0.8 pneumatic barb fittings for tubes of 1/16 inch internal diameter (installed);
- 1 M5X0.8 pneumatic silencer (installed).

Figure 1: MPM500 module with barb fittings and silencer installed

> **Note**
>
> Note that we also supply the MPM500 as a kit with a pneumatic gripper from Schunk, either MPG-plus 25 or MPZ 30, the appropriate adapter plates and 4-mm tubing.

Technical specifications

Table 1 lists the main technical specifications for the MPM500 pneumatic module and Figure 2 shows its principal dimensions. The module essentially consists of two 3/2 NC solenoid valves that can be controlled independently.

Table 1: Technical specifications for the MPM500 module

Characteristics	Value
Pneumatic valve type	3/2 NC solenoid valves
Pneumatic valves action lines	ports 1 & 2
Compressed air connector	port IN
Operating pressure	0–7 bar (0–102 psi)
Flow rate	29.74 LPM (1.05 CFM)
Response time	ON/OFF: 8/10 ms
Ports	Four M5 threaded orifices, of which ports 1, 2, and IN are with preinstalled barb fittings for 1/16-in ID tubes and port OUT is with a preinstalled pneumatic muffler.
Housing	Coated aluminum alloy
Total weight	108 g
Operating temperature	5° to 55°C
Operating humidity	10% to 95% RH (non-condensing)

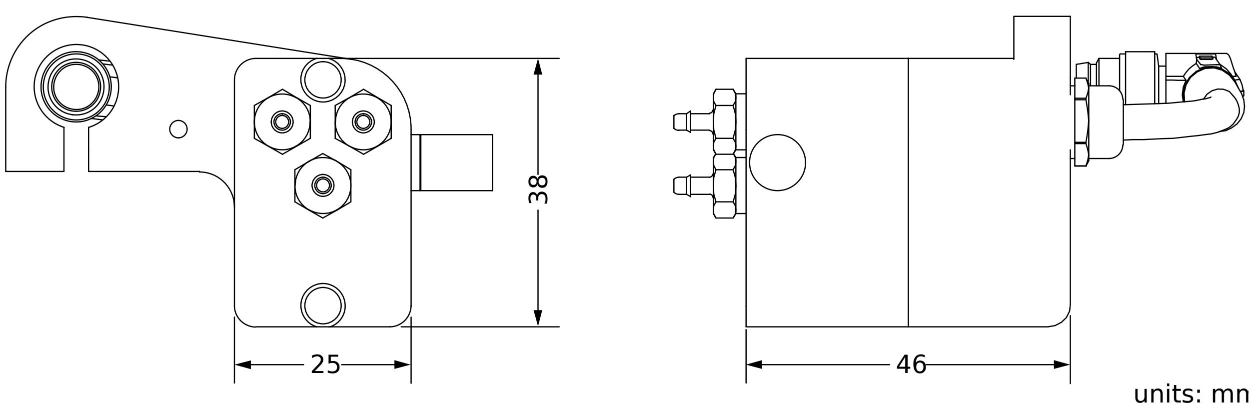


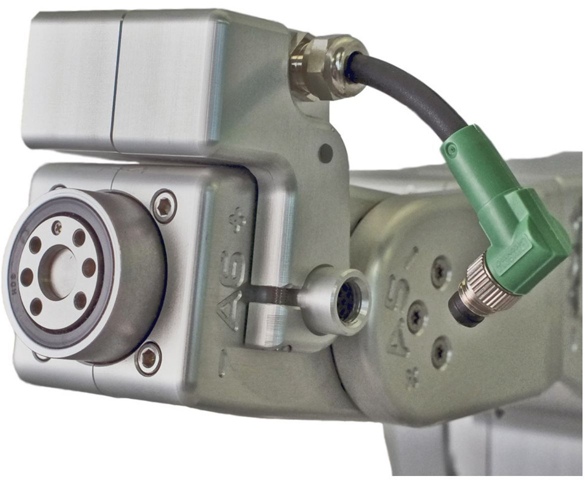
Figure 2: Dimensions of the MPM500 module

> **Note**
>
> You can download the CAD file of the MPM500 pneumatic module from here.

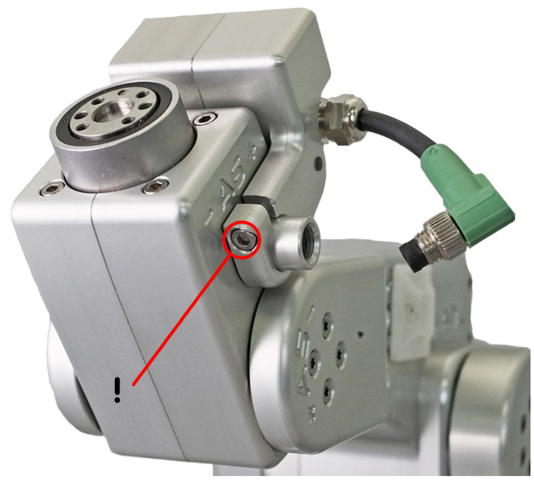
Installing the module

The MPM500 pneumatic module is designed for only one type of installation, as shown in Figure 3. The module can be used with a variety of pneumatic grippers, suction cups, pneumatic tool changers or other pneumatic devices.

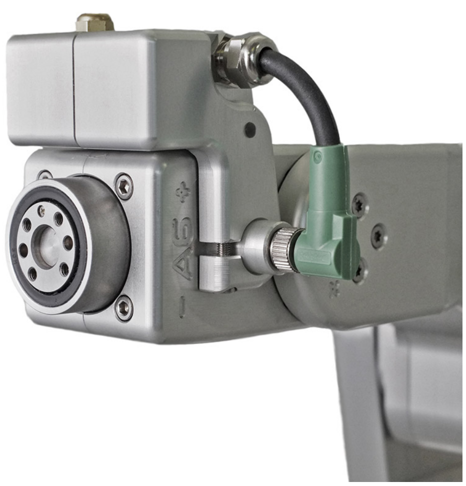
> ⚠ **Warning**
>
> Make sure the Meca500 robot is powered off, before installing the MPM500 module.



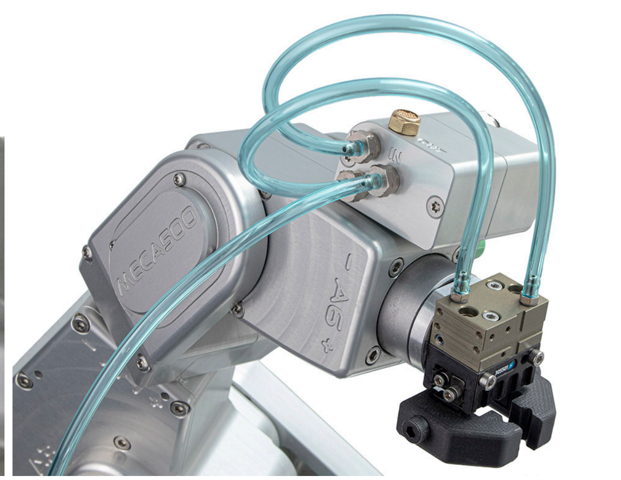
(a) positioning the module



(b) securing the module



(c) connecting the electric cable



(d) connecting the pneumatic tubes

Figure 3: Installing the MPM500 module with Schunk's MPG-plus 25 pneumatic gripper

To install the MPM500 pneumatic module, make sure the robot is switched off, and then follow these steps:

1. Slide the MPM500 module on top of the Meca500's wrist, aligning it with the robot's electrical connector (Figure 3a).
2. Tighten the screw of the module clamp with an Allen key in order to secure it in place as shown in Figure 3b.
3. Connect the 6-pin SDLR cable of the module to the Meca500's electrical connector (Figure 3c).
4. To complete the set-up, install your pneumatic tooling and then the required pneumatic tubing (Figure 3d). Examples of pneumatic circuits are given in the next section. Make sure the tubing is sufficiently long for the required range of motion of joint 6 of the robot.
5. Power the Meca500 (see its User Manual for more information).
6. After activation of the Meca500, the green LED on the module should emit steady light, indicating that the electrical connection is functional, as shown in Figure 4.

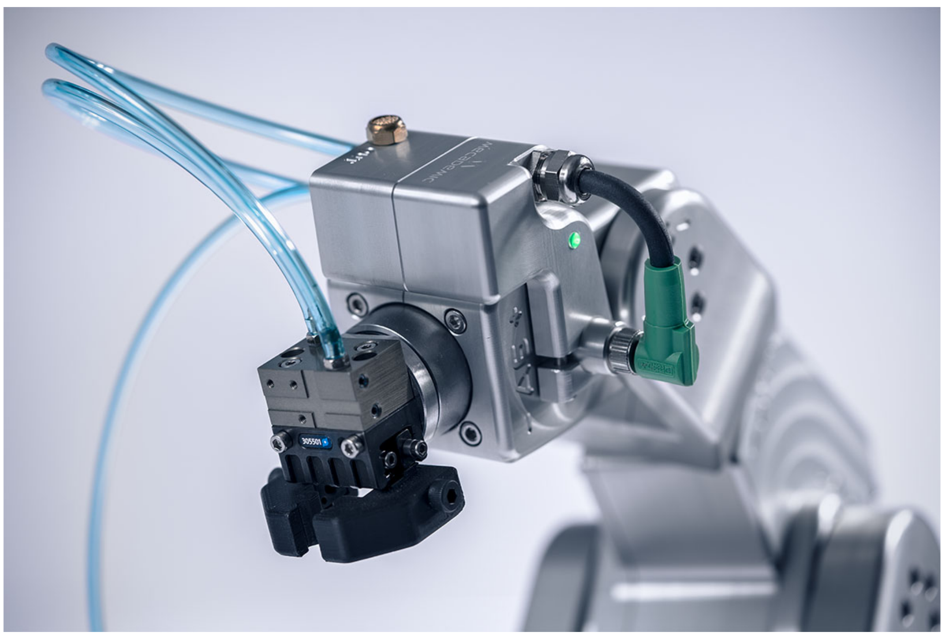


Figure 4: Pneumatic module connected to the Meca500 and to a double-action pneumatic gripper

Pneumatic circuit examples

As already specified, the MPM500 consists of two three-way, two-position normally closed valves that can be controlled independently. The valves share the same inlet port (IN) and exhaust port (OUT). The following two sections present three typical examples of pneumatic circuits.

> **Note**
>
> If you operate the valves in a continuous manner, the module may heat up. This is normal.

Double-action pneumatic gripper

To connect your MPM500 to a double-action pneumatic gripper, as in Figure 4, follow the circuit of Figure 5 The compressed air should be introduced in the MPM500's IN port. It will be distributed to both 3/2 NC valves. The solenoid switch activates one valve which allows for the gripper to open/close its fingers, depending on which valve is activated. The gripper's pneumatic connectors should go in the MPM500's Port 1 and Port 2.

You should use the same pneumatic circuit in the case of a tool changer.

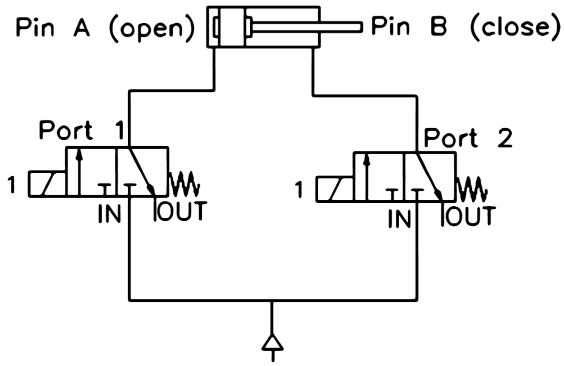


Figure 5: Pneumatic circuit in the case of a double-action pneumatic gripper

Vacuum suction cup

To connect the MPM500 to vacuum suction cups, follow the circuits on Figure 6 or Figure 7. In each of these two cases, only one of the valves is used.

With the first circuit (Figure 6), the compressed air must enter the MPM500's IN port. When the valve is activated the air flows towards a venturi vacuum pump and connects to its positive pressure port. The suction cup, and an additional air filter if necessary, are connected to the venturi vacuum pump's negative pressure port. The vacuum is toggled with the activation (open state) and release (close state) of the valve. If you have a compact venturi vacuum generator, this circuit is the most efficient one.

A more practical way to connect a suction cup, especially if your vacuum generator is bulky, is presented on Figure 7. The compressed air is sent directly to the venturi pump on its positive pressure port.

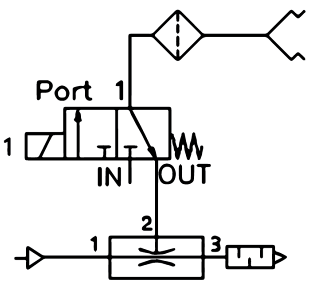


Figure 6: Pneumatic circuit in the case of a suction cup and a small venturi vacuum generator

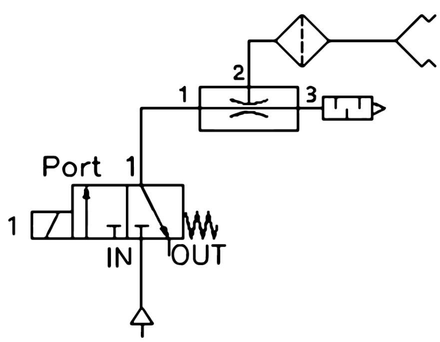


Figure 7: Pneumatic circuit in the case of a suction cup and a bulkier venturi vacuum generator

This circuit should also be used when working with a direct vacuum line. The vacuum line is connected to the OUT port of the MPM500, while the end-of-arm tooling is connected either to Port 1 or Port 2 of the pneumatic module.

Note

If you are using a vacuum line instead of a compressed air line, make sure to connect it to the OUT port of the MPM500, as shown in Figure 7.

Operating the module

The MPM500 pneumatic module is controlled in the same way as the Meca500. Its valves can be opened or closed using the SetValveState(v_1,v_2) command, where v_1 and v_2 are the states of valves 1 and 2, respectively, with 1 for open, 0 for closed, and −1 for keep unchanged. You can also request the state of the valves.

Firmware update

If you upgrade the firmware of your robot (using the procedure described in the MecaPortal operating manual) while the pneumatic module is installed, the firmware of the module will be automatically updated. Otherwise, you can update the firmware of your module separately by following the same procedure, but selecting the file m500_exttools_*.update, instead of the file Meca500_E_LD_*.update.

Integration into the MecaPortal

The MecaPortal web interface automatically recognizes the MPM500 module and displays a CAD model of it, as well as a widget specific to the module (Figure 8). In the widget, you can control the states of the two valves.

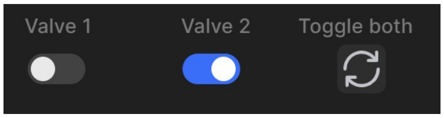


Figure 8: Widget that appears in the MecaPortal jogging panel when the MPM500 is detected

Safety

The MPM500 pneumatic module is designed with safety in mind. However, additional tools connected to the module may lead to risks of injuries. Make sure that all connections are properly made and that you respect the technical specifications of the pneumatic module.

Activation and E-Stop

If an activated robot with an MPM500 pneumatic module becomes deactivated, the valves are no longer controllable and regain their default state, i.e., they close.

If an E-Stop is activated on a Meca500 R3, the whole system is powered off, while on a Meca500 R4, power is removed from the robot motors AND the MPM500 pneumatic module connected to the robot. Thus, in both cases, the valves return to their default state.

Danger

After a deactivation or an E-Stop, the valves regain their default states (i.e., closed). Therefore, if the robot's pneumatic EOAT is holding a part, the part might fall.

Terminology

Below is the list of terms used by us in our technical documentation.

active line: The line in the MecaPortal where the cursor is currently positioned.

BRF: Base Reference Frame.

Cartesian space: The six-dimensional space defined by the position (x, y, z) and orientation (α, β, γ) of the TRF with respect to the WRF.

control port: The TCP port 10000, over which commands to the robot and messages from the robot are sent.

data request commands: Commands used to request some data regarding the robot (e.g., GetTrf, GetBlending, GetJointVel). These commands are executed immediately and generally return values for parameters that have already been configured (sent and executed) with a Set* command (or the default values).

default value: There are different settings in the robot controller that can be configured using Set* commands (e.g., SetCartAcc). Many of these settings have default values. Every time the robot is powered up, these settings are initialized to their default values. In the case of motion commands settings, their values are also initialized to their default values every time the robot is deactivated. In contrast, some settings are persistent and their values are stored on an SD drive.

detailed event log: This file mirrors the content of the event log panel in the MecaPortal when in detailed mode. It can be downloaded from the MecaPortal (see *troubleshoot-prog* of the Programming Manual).

EOAT: End-of-arm tooling.

EOB: End-of-block message, [3012][], sent by default every time the robot has stopped moving AND its motion queue is empty. You can disable this message with the command SetEob.

EOM: End-of-motion message, [3004][], sent by the robot whenever it has stopped moving for at least 1 ms, if this option is activated with SetEom.

error mode: The robot goes into error mode when it encounters an error while executing a command or a hardware problem (see *tab:error-messages*).

Euler angles: A set of three angles, {α, β, γ}, used the define an orientation in space. We use the mobile (intrinsic) XYZ convention. See *Euler-angles* of the Programming manual for more details.

FCP: Flange Center Point. The origin of the FRF.

FRF: Flange Reference Frame.

instantaneous commands: These are commands that are executed immediately, as soon as received by the robot. All data request commands (Get*), all robot control commands, all work zone supervision and collision prevention commands and some optional accessories commands (*_Immediate) are instantaneous.

inverse kinematics: The problem of obtaining the robot joint sets that correspond to a desired end-effector pose. See *inverse-kinematics* of the Programming manual for more details.

joint position: The joint angle associated with a specific joint.

joint set: The set of all joint positions.

joint space: The six-dimensional space defined by the positions of the robot joints.

monitoring port: The TCP port 10001, over which data is sent periodically from the robot.

motion commands: Commands used to construct the robot trajectory (e.g., Delay, MoveJoints, SetTRF, SetBlending). When a Mecademic robot receives a motion command, it places it in a motion queue. The command will be run once all preceding motion commands have been executed.

motion queue: The buffer where motion commands that were sent to the robot are stored and executed on a FIFO basis by the robot.

offline program: A sequence of commands saved in the internal memory of the robot. The term *offline* is often omitted and will eventually be removed altogether.

online mode programming: Programming the robot in online mode involves moving it directly to each desired robot position, typically using jogging controls.

PDO (**Process Data Object**): In EtherCAT, a Process Data Object (PDO) is a data structure used for exchanging real-time cyclic data between an EtherCAT master and its slave devices. PDOs can contain individual bits, bytes, or words.

persistent settings: Some settings in the robot controller have default values (e.g., the robot name set by the command SetRobotName), but when changed, their new values are written on an SD drive and persist even if the robot is powered off.

pose: The position and orientation of one reference frame with respect to another.

position mode: One of the two control modes, in which the robot's motion is generated by requesting a target end-effector pose or joint set (see *pos-vel-modes* of the Programming Manual).

robot posture configuration: The set of two-value (−1 or 1) parameters c_s, c_e, and c_w that normally defines each of the eight possible robot postures for a given pose of the robot's end-effector.

queued commands: Commands that are placed in the motion queue, rather than executed immediately. All motion commands are queued commands, as well as some external-tool commands.

reach: The maximum distance between the axis of joint 1 and the center of the robot's wrist.

real-time data request commands: Commands used to request some real-time data regarding the current status of robot (e.g., GetRtTrf, GetRtCartPos, GetStatusRobot).

robot control commands: Commands used to immediately control the robot, (e.g., ActivateRobot, PauseMotion, SetNetworkOptions). These commands are executed immediately, i.e., are instantaneous.

robot is ready for motion: The robot is considered *ready* to receive motion commands, i.e. when it is activated and homed, or alternatively when recovery-mode is enabled while the robot is activated but not homed.

Note that if the robot is in error or if a safety stop condition is present, it will refuse motion commands, but it will still be considered *ready* since its motion queue remains initialized and retains the latest received settings (e.q., velocity, acceleration, blending, WRF, TRF, etc.).

robot log: This file is a more detailed version of the user log, intended primarily for our support team. It can be downloaded from the MecaPortal (see *troubleshoot-prog* of the Programming Manual).

robot position: A robot position is equivalent to either a joint set or the pose of the TRF relative to the WRF, along with the definitions of both reference frames, and the robot posture and last joint turn configuration parameters.

robot posture: The arrangement of the robot links. Equivalent to a joint set in which all joint angles are normalized, i.e. have been converted to the range $(-180°, 180°]$.

SDO (**Service Data Object**): In EtherCAT, a Service Data Object (SDO) is a data structure used for non-real-time communication between an EtherCAT master and its slave devices. SDOs are typically used to configure device parameters and access diagnostic information through the object dictionary. Unlike PDOs, SDOs exchange structured data rather than individual bits or bytes.

singularities: A robot posture where the robot's end-effector is blocked in some directions even if no joint is at a limit (see *singularities* of the Programming Manual).

TCP: Tool Center Point. The origin of the TRF. Not to be confused with Transmission Control Protocol.

TRF: Tool reference frame.

turn configuration parameter: Since the last joint of the robot can rotate multiple revolutions, the turn configuration parameter defines the revolution number.

user log: This file is a simplified log containing user-friendly traces of major events (e.g., robot activation, movement, E-Stop activation). It can be downloaded from the MecaPortal (see *troubleshoot-prog* of the Programming Manual).

velocity mode: One of the two control modes, in which the robot's motion is generated by requesting a target joint velocity vector or end-effector Cartesian velocity vector (see *pos-vel-modes* of the Programming Manual).

workspace: The Cartesian workspace of a robot is the set of all feasible poses of its TRF with respect to its WRF. Note that many of these poses can be attained with more than one set of configuration parameters.

WRF: World reference frame.

wrist center: the point where the axes of joints 4, 5, and 6 intersect.