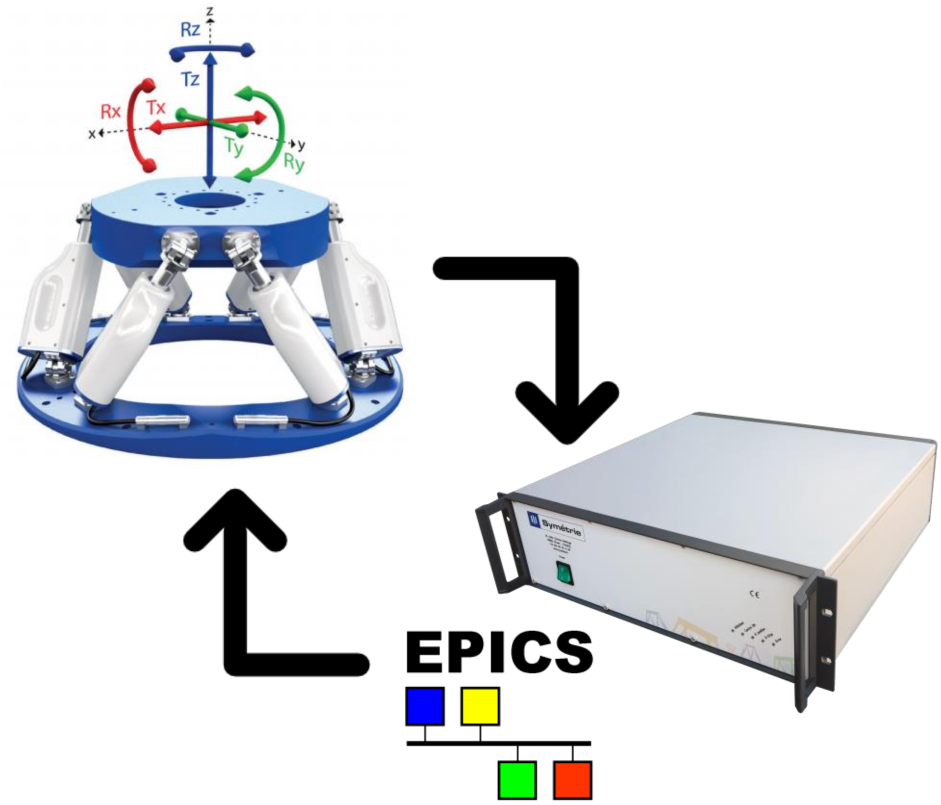
Software user manual

EPICS module

Release 5.0.7.210830



Document:	MAN_SOFT_EPICS
Issue:	A
Date (DD/MM/YYYY):	30/08/2021
Number of pages:	66

INFORMATION AND PROPERTY

SYMETRIE

10, Allee Charles Babbage
30035 Nimes - Cedex 1
FRANCE
Phone: +33 (0)4 66 29 43 88
Technical support: info@symetrie.fr

MODIFICATIONS TABLE

Issue	Date	Pages	Nature of change
A	30/08/2021	All	Document creation

APPROBATION TABLE

A	30/08/2021	A. L'HOSTIS Mechatronic engineer	P. NOIRE Product R&D manager	T. ROUX Technical manager
Issue	**Date**	**Prepared by**	**Checked by**	**Released by**

SUMMARY

Information and property 2

Modifications table 2

Approbation table 2

Summary 3

1 Introduction 5

2 Requirements 6

2.1 Hardware requirements 6

2.2 Software requirements 6

3 Installation under Linux 7

3.1 Core OS package installation 7

3.2 Installation folder 8

3.3 libssh2 Installation 8

3.4 EPICS Base, modules and extensions 9

3.4.1 EPICS Base 9

3.4.2 IPAC module 10

3.4.3 Seq module 10

3.4.4 Async module 11

3.4.5 Calc module 12

3.4.6 Autosave module 13

3.4.7 Busy module 13

3.4.8 Motor module 14

3.4.9 PMAC module 16

3.4.10 PyDM installation 19

3.4.11 EDM installation 20

3.4.12 SYMETRIE hexapod module installation 24

3.4.13 Demonstration application installation 26

4 Running the Test Application 28

4.1 EPICS base path in terminal default paths list 28

4.2 Running the EPICS database 29

4.3 Running the PyDM screens 30

4.3.1 Start the PyDM main screen 30

4.3.2 Main screen presentation 30

4.3.3 Advanced states screen presentation 33

4.3.4 Parameters screen presentation 36
4.4 Running the EDM screens *41*
4.4.1 Start the EDM main screen 41
4.4.2 Main screen presentation 41
4.4.3 Advanced states screen presentation 44
4.4.4 Parameters screen presentation 48
5 EPICS records 53
5.1 Presentation *53*
5.2 Basic example *63*
6 Software release history 66
6.1 Release 5.0.7.210830 .... *66*
6.2 Release 5.0.5.191103 .... *66*

1 INTRODUCTION

This document describes the installation and use of the EPICS support module for the SYMETRIE positioning hexapods. This module is compatible with controller hardware based on the Delta Tau Power PMAC motion controller. This EPICS support module utilises the SYMETRIE Application programming interface (API) and is built on the existing EPICS "pmac" module which communicates with the hardware controller by opening an SSH connection. This document describes the steps required to install the EPICS support module and presents the example application provided with the support module. The example application can be integrated into any existing EPICS control system or used as a base for writing a custom integration.

The SYMETRIE Hexapod EPICS software module contains an asyn driver and a set of GUI screens for PyDM and EDM. This module is built on top of the pmac software module and makes use of the existing low level ssh asyn driver. The module uses standard EPICS records to make integration with existing facilities easy to manage.

The module contains a demonstration application with EPICS database files and start-up scripts to quickly get started with a test EPICS IOC. There are some PyDM and EDM screens that can be used to interact with the hexapod records which are used within the demonstration application.

This document often refers to the Application programming interface (API) manual. The API manual offers detailed description of the command's parameters and of the status, while this EPICS manual concentrates on the EPICS module description. If an information on a command or a status cannot be found in this manual, please refer to the API manual.

The release is number is composed as followed: MajorVersion.MinorVersion.SvnNumber.APIversion.

MajorVersion	Software family major version number. This number must match with the MajorVersion number of the SYM_Positioning version, the API version, and other interface libraries.
MinorVersion	Software family minor version number. This number must match with the MinorVersion number of the SYM_Positioning version, the API version, and other interface libraries.
SvnNumber	Trace the EPICS module version.
APIversion	Give the API compatibility version. It makes the link with the Application programming interface (API) manual.

The development and most of this documentation has been realized by the company Observatory Sciences (http://www.observatorysciences.co.uk/).

2 REQUIREMENTS

This section details the hardware and software requirements for using the SYMETRIE hexapod EPICS software module.

2.1 HARDWARE REQUIREMENTS

The following hardware is required to run the support module code:

- Standard PC with an Ethernet port.
- SYMETRIE Hexapod control enclosure present on the network. For operation with the EPICS device support code the controller should be configured to have a static IP address and this address must be specified in the start-up script.

2.2 SOFTWARE REQUIREMENTS

The following software must be installed to run the driver and device support code:

- Linux operating system. The support module code has been written using general asyn, motor, pmac and EPICS base function calls and methods and as a result will be fully functional on any system that can compile the required versions of EPICS base and the asyn, motor and pmac modules. The device support code was developed on Debian 10 (64 bit).
- EPICS Base (version 3.15.7 or later)
- EPICS module ‘asyn’ (version 4.38 or later)
- EPICS module ‘pmac’ (version 2.4.10 or later).
- EDM to run engineering screens, if they are required.
- PyDM to run the ui screens, if they are required.
- libssh2 library (not an EPICS module, see installation instructions below).

3 INSTALLATION UNDER LINUX

This section covers installation of the support module for the SYMETRIE hexapod on a Linux operating system.

3.1 CORE OS PACKAGE INSTALLATION

To be able to compile and run all software modules there are a set of system development tools and libraries that must first be installed. To install all of the necessary tools and libraries the following commands should be executed from a terminal command line.

The packages below are required for compiling the EPICS base applications and support modules required by the SYMETRIE hexapod module.

```
sudo apt-get install build-essential
sudo apt-get install re2c
sudo apt-get install libssh2-1-dev
sudo apt-get install libboost-all-dev
```

Git can be used to download many of the EPICS modules although it is not necessary.

```
sudo apt-get install git
```

The packages below are required for the setup of the PyDM environment and to be able to execute the application. Note the use of these packages is not necessarily the only way to install and run PyDM, they are required for the installation method described below.

```
sudo apt-get install python3-venv
sudo apt-get install libxcb-xinerama0-dev
```

The packages below are required for the compilation of the EDM application. Note the installation of EDM can be fairly complex and these packages were required for a Debian installation. The exact package requirements may differ for alternative Linux distributions.

```
sudo apt-get install multiarch-support
sudo apt-get install libxt-dev
sudo apt-get install libxmu-dev
sudo apt-get install libmotif-dev
sudo apt-get install libxtst-dev
sudo apt-get install libgif-dev
```

One additional package is required for EDM that could not be found with apt-get. Instead, it can be manually installed.

```
cd ~/Downloads
wget http://ftp.uk.debian.org/debian/pool/main/libx/libxp/libxp6_1.0.2-2_amd64.deb
sudo dpkg -i ~/Downloads/libxp6_1.0.2-2_amd64.deb
sudo ln -s /usr/lib/x86_64-linux-gnu/libXp.so.6 /usr/lib/x86_64-linux-gnu/libXp.so
```

3.2 INSTALLATION FOLDER

The installation root is defined as follows

```
/home/$USER/epics/R3.15.7
```

$USER is an environment variable representing the user session name. R3.15.7 is the EPICS base version used to develop the SYMETRIE hexapod support module.

The environment variable $HOME will be used to define the home path to be independent of the user and to shorter the home path.

```
$HOME/epics/R3.15.7
```

An environment variable can be read with the command "printenv":

```
printenv HOME
```

Please create the following folder:

```
mkdir -p $HOME/epics/R3.15.7/download
```

A more detailed explanation is given on the choice made for this installation folder at the chapter 3.4.

The environment variable cannot be used in the RELEASE files of an EPICS build. When that it is the case, the user "symetrie" will be used as an example, giving the following path:

```
/home/symetrie/epics/R3.15.7
```

3.3 LIBSSH2 INSTALLATION

The lower-level driver class provided by the EPICS module requires that an external library be installed on the system to handle SSH encryption and connections. The library libssh2 (http://www.libssh2.org) may already be available for use within the OS, or else it can be downloaded and installed.

Either download using the link on the libssh2 website, or from a command line download using wget. Unpack the archive to a suitable location, configure, build and install.

```
cd $HOME/epics/R3.15.7/download
wget http://www.libssh2.org/download/libssh2-1.4.3.tar.gz
gunzip -k libssh2-1.4.3.tar.gz
tar -xvf libssh2-1.4.3.tar
cd libssh2-1.4.3
./configure
make
```

The libraries are by default installed in /usr/local/lib. Note that installation in that location requires root access.

```
su
```

```
make install
exit
```

```
cd $HOME/epics/R3.15.7/download
rm -r libssh2-1.4.3.tar libssh2-1.4.3
```

Note that a libssh2 Debian package already exists but the current package version may not be compatible with this module, depending on the Debian OS version.

3.4 EPICS Base, modules and extensions

3.4.1 EPICS Base

The SYMETRIE hexapod support module has been developed against EPICS base version 3.15.7. EPICS base can be downloaded from https://epics-controls.org/download/base/base-3.15.7.tar.gz

The following instructions will assume a common installation directory and all instructions will be valid starting from this point. The particular directory structure presented by this installation guide is not enforced by EPICS, it is simply chosen to offer a clean and consistent way of naming and installing EPICS modules and applications within this guide.

The installation root is defined as follows, with $USER representing the user session name

```
/home/$USER/epics/R3.15.7
```

The environment variable $HOME is used to be independent of the user and to shorter the home path.

```
$HOME/epics/R3.15.7
```

Now download and install EPICS base

```
cd $HOME/epics/R3.15.7/download
wget https://epics.anl.gov/download/base/base-3.15.7.tar.gz
gunzip -k base-3.15.7.tar.gz
tar -xvf base-3.15.7.tar
mv base-3.15.7 $HOME/epics/R3.15.7/base
rm -r base-3.15.7.tar
cd $HOME/epics/R3.15.7/base
make
```

Create a support area in which the modules will be installed

```
mkdir -p $HOME/epics/R3.15.7/support
mkdir -p $HOME/epics/R3.15.7/download/support
```

Now it is necessary to install the modules that the SYMETRIE Hexapod module depends on.

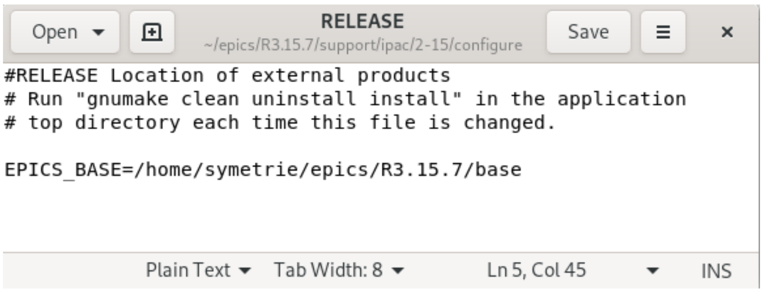
3.4.2 IPAC module

Install the ipac module (required by the asyn and motor modules)

```
cd $HOME/epics/R3.15.7/download/support
wget https://github.com/epics-modules/ipac/releases/download/2.15/ipac-2.15.tar.gz
gunzip -k ipac-2.15.tar.gz
tar -xvf ipac-2.15.tar
mkdir -p $HOME/epics/R3.15.7/support/ipac
mv ipac-2.15/ $HOME/epics/R3.15.7/support/ipac/2-15
rm ipac-2.15.tar
```

Edit the RELEASE file located in the module configure directory, update the EPICS_BASE definition, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/ipac/2-15/configure/RELEASE
```



```
#RELEASE Location of external products
# Run "gnumake clean uninstall install" in the application
# top directory each time this file is changed.

EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

```
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

Note that “symetrie” should be replaced by the user.

Build the module

```
cd $HOME/epics/R3.15.7/support/ipac/2-15
make
```

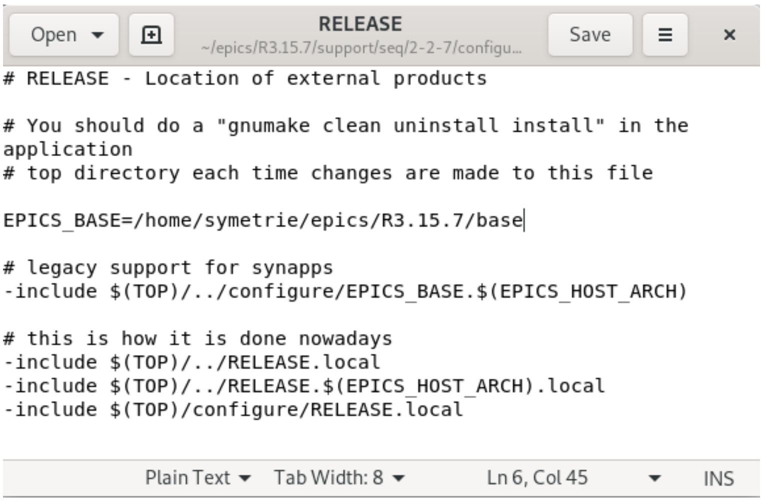
3.4.3 Seq module

Install the seq module (required by the asyn and motor modules)

```
cd $HOME/epics/R3.15.7/download/support
wget https://www-csr.bessy.de/control/SoftDist/sequencer/releases/seq-2.2.7.tar.gz
gunzip -k seq-2.2.7.tar.gz
tar -xvf seq-2.2.7.tar
mkdir -p $HOME/epics/R3.15.7/support/seq
mv seq-2.2.7 $HOME/epics/R3.15.7/support/seq/2-2-7
rm seq-2.2.7.tar
```

Edit the RELEASE file located in the module configure directory, update the EPICS_BASE definition, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/seq/2-2-7/configure/RELEASE
```



```
# RELEASE - Location of external products

# You should do a "gnumake clean uninstall install" in the
application
# top directory each time changes are made to this file

EPICS_BASE=/home/symetrie/epics/R3.15.7/base

# legacy support for synapps
-include $(TOP)/../configure/EPICS_BASE.$(EPICS_HOST_ARCH)

# this is how it is done nowadays
-include $(TOP)/../RELEASE.local
-include $(TOP)/../RELEASE.$(EPICS_HOST_ARCH).local
-include $(TOP)/configure/RELEASE.local
```

```
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

Note that “symetrie” should be replaced by the user.

Build the module

```
cd $HOME/epics/R3.15.7/support/seq/2-2-7
make
```

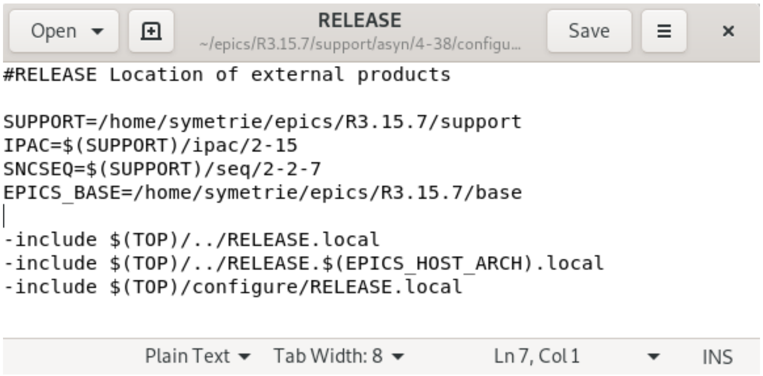
3.4.4 Async module

Install the asyn module (required by the busy, motor and pmac modules)

```
cd $HOME/epics/R3.15.7/download/support
wget https://epics.anl.gov/download/modules/asyn4-38.tar.gz
gunzip -k asyn4-38.tar.gz
tar -xvf asyn4-38.tar
mkdir -p $HOME/epics/R3.15.7/support/asyn
mv asyn4-38 $HOME/epics/R3.15.7/support/asyn/4-38
rm asyn4-38.tar
```

Edit the RELEASE file located in the module configure directory, update the SUPPORT, EPICS_BASE, IPAC and SNCSEQ definitions, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/asyn/4-38/configure/RELEASE
```



```
#RELEASE Location of external products

SUPPORT=/home/symetrie/epics/R3.15.7/support
IPAC=$(SUPPORT)/ipac/2-15
SNCSEQ=$(SUPPORT)/seq/2-2-7
EPICS_BASE=/home/symetrie/epics/R3.15.7/base

-include $(TOP)/../RELEASE.local
-include $(TOP)/../RELEASE.$(EPICS_HOST_ARCH).local
-include $(TOP)/configure/RELEASE.local
```

```
SUPPORT=/home/symetrie/epics/R3.15.7/support
IPAC=$(SUPPORT)/ipac/2-15
SNCSEQ=$(SUPPORT)/seq/2-2-7
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

Build the module

```
cd $HOME/epics/R3.15.7/support/asyn/4-38
make
```

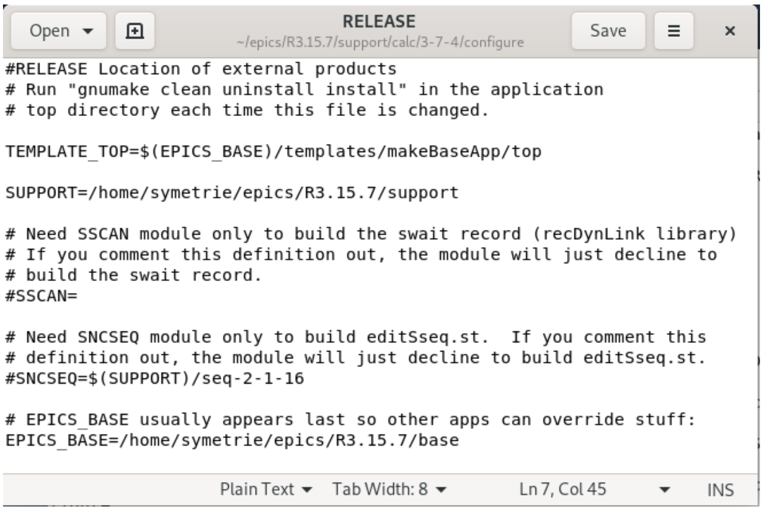
3.4.5 Calc module

Install the calc module (required by the pmac module)

```
cd $HOME/epics/R3.15.7/download/support
wget https://github.com/epics-modules/calc/archive/R3-7-4.tar.gz
gunzip -k R3-7-4.tar.gz
tar -xvf R3-7-4.tar
mkdir -p $HOME/epics/R3.15.7/support/calc
mv calc-R3-7-4 $HOME/epics/R3.15.7/support/calc/3-7-4
rm R3-7-4.tar
mv R3-7-4.tar.gz calc-R3-7-4.tar.gz
```

Edit the RELEASE file located in the module configure directory, comment out the SSCAN definition and update the EPICS_BASE definition, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/calc/3-7-4/configure/RELEASE
```



```
#RELEASE Location of external products
# Run "gnumake clean uninstall install" in the application
# top directory each time this file is changed.

TEMPLATE_TOP=$(EPICS_BASE)/templates/makeBaseApp/top

SUPPORT=/home/symetrie/epics/R3.15.7/support

# Need SSCAN module only to build the swait record (recDynLink library)
# If you comment this definition out, the module will just decline to
# build the swait record.
#SSCAN=

# Need SNCSEQ module only to build editSseq.st.  If you comment this
# definition out, the module will just decline to build editSseq.st.
#SNCSEQ=$(SUPPORT)/seq-2-1-16

# EPICS_BASE usually appears last so other apps can override stuff:
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

```
SUPPORT=/home/symetrie/epics/R3.15.7/support
#SSCAN=
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

Build the module

```
cd $HOME/epics/R3.15.7/support/calc/3-7-4
make
```

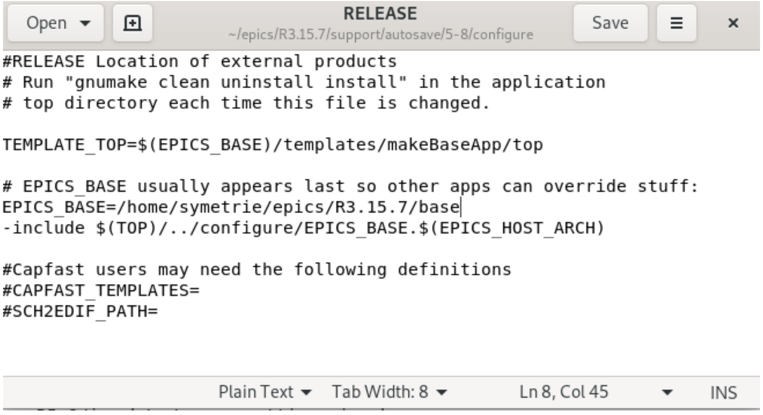
3.4.6 Autosave module

Install the autosave module (required by the busy module)

```
cd $HOME/epics/R3.15.7/download/support
wget https://github.com/epics-modules/autosave/archive/R5-8.tar.gz
gunzip -k R5-8.tar.gz
tar -xvf R5-8.tar
mkdir -p $HOME/epics/R3.15.7/support/autosave
mv autosave-R5-8 $HOME/epics/R3.15.7/support/autosave/5-8
rm R5-8.tar
mv R5-8.tar.gz autosave-R5-8.tar.gz
```

Edit the RELEASE file located in the module configure directory, update the EPICS_BASE definition, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/autosave/5-8/configure/RELEASE
```



```
#RELEASE Location of external products
# Run "gnumake clean uninstall install" in the application
# top directory each time this file is changed.

TEMPLATE_TOP=$(EPICS_BASE)/templates/makeBaseApp/top

# EPICS_BASE usually appears last so other apps can override stuff:
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
-include $(TOP)/../configure/EPICS_BASE.$(EPICS_HOST_ARCH)

#Capfast users may need the following definitions
#CAPFAST_TEMPLATES=
#SCH2EDIF_PATH=
```

```
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

Build the module

```
cd $HOME/epics/R3.15.7/support/autosave/5-8
make
```

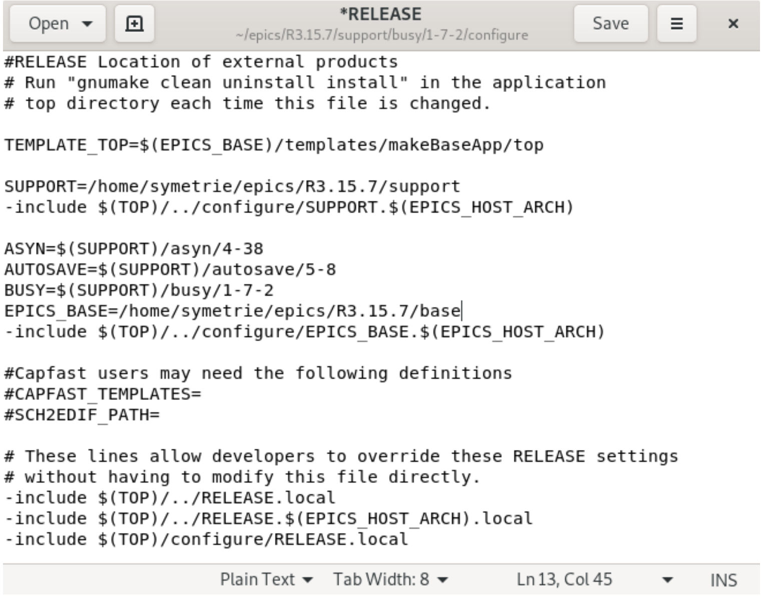
3.4.7 Busy module

Install the busy module (required by the motor and pmac modules)

```
cd $HOME/epics/R3.15.7/download/support
wget https://github.com/epics-modules/busy/archive/R1-7-2.tar.gz
gunzip -k R1-7-2.tar.gz
tar -xvf R1-7-2.tar
mkdir -p $HOME/epics/R3.15.7/support/busy
mv busy-R1-7-2 $HOME/epics/R3.15.7/support/busy/1-7-2
rm R1-7-2.tar
mv R1-7-2.tar.gz busy-R1-7-2.tar.gz
```

Edit the RELEASE file located in the module configure directory, update the SUPPORT, EPICS_BASE, ASYN, AUTOSAVE and BUSY definitions, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/busy/1-7-2/configure/RELEASE
```



```
#RELEASE Location of external products
# Run "gnumake clean uninstall install" in the application
# top directory each time this file is changed.

TEMPLATE_TOP=$(EPICS_BASE)/templates/makeBaseApp/top

SUPPORT=/home/symetrie/epics/R3.15.7/support
-include $(TOP)/../configure/SUPPORT.$(EPICS_HOST_ARCH)

ASYN=$(SUPPORT)/asyn/4-38
AUTOSAVE=$(SUPPORT)/autosave/5-8
BUSY=$(SUPPORT)/busy/1-7-2
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
-include $(TOP)/../configure/EPICS_BASE.$(EPICS_HOST_ARCH)

#Capfast users may need the following definitions
#CAPFAST_TEMPLATES=
#SCH2EDIF_PATH=

# These lines allow developers to override these RELEASE settings
# without having to modify this file directly.
-include $(TOP)/../RELEASE.local
-include $(TOP)/../RELEASE.$(EPICS_HOST_ARCH).local
-include $(TOP)/configure/RELEASE.local
```

```
SUPPORT=/home/symetrie/epics/R3.15.7/support
ASYN=$(SUPPORT)/asyn/4-38
AUTOSAVE=$(SUPPORT)/autosave/5-8
BUSY=$(SUPPORT)/busy/1-7-2
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

Build the module

```
cd $HOME/epics/R3.15.7/support/busy/1-7-2
make
```

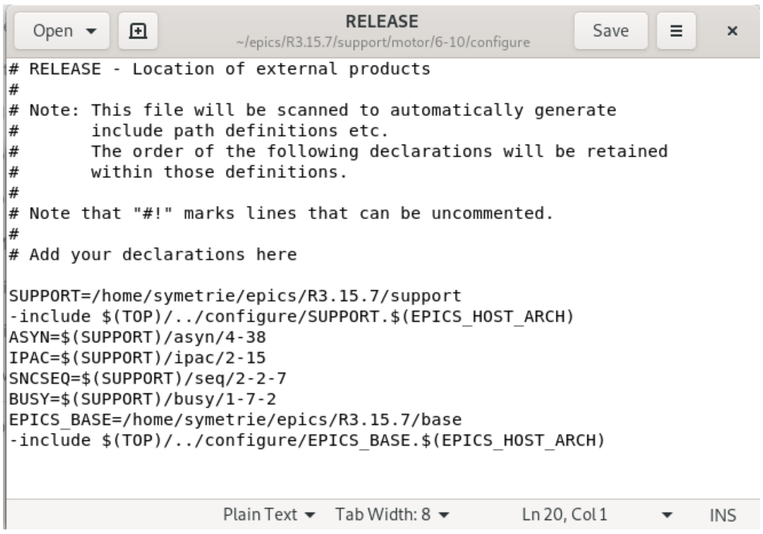
3.4.8 Motor module

Install the motor module (required by the pmac module)

```
cd $HOME/epics/R3.15.7/download/support
wget https://github.com/epics-modules/motor/archive/R6-10.tar.gz
gunzip -k R6-10.tar.gz
tar -xvf R6-10.tar
mkdir -p $HOME/epics/R3.15.7/support/motor
mv motor-R6-10 $HOME/epics/R3.15.7/support/motor/6-10
rm R6-10.tar
mv R6-10.tar.gz motor-R6-10.tar.gz
```

Edit the RELEASE file located in the module configure directory, update the SUPPORT, EPICS_BASE, ASYN, IPAC, SNCSEQ and BUSY definitions, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/motor/6-10/configure/RELEASE
```

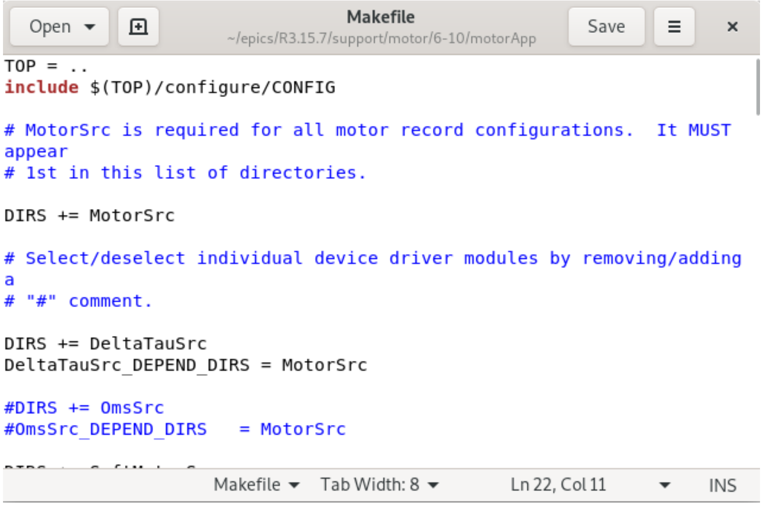


```
# RELEASE - Location of external products
#
# Note: This file will be scanned to automatically generate
#       include path definitions etc.
#       The order of the following declarations will be retained
#       within those definitions.
#
# Note that "#!" marks lines that can be uncommented.
#
# Add your declarations here

SUPPORT=/home/symetrie/epics/R3.15.7/support
-include $(TOP)/../configure/SUPPORT.$(EPICS_HOST_ARCH)
ASYN=$(SUPPORT)/asyn/4-38
IPAC=$(SUPPORT)/ipac/2-15
SNCSEQ=$(SUPPORT)/seq/2-2-7
BUSY=$(SUPPORT)/busy/1-7-2
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
-include $(TOP)/../configure/EPICS_BASE.$(EPICS_HOST_ARCH)
```

```
SUPPORT=/home/symetrie/epics/R3.15.7/support
ASYN=$(SUPPORT)/asyn/4-38
IPAC=$(SUPPORT)/ipac/2-15
SNCSEQ=$(SUPPORT)/seq/2-2-7
BUSY=$(SUPPORT)/busy/1-7-2
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

Edit the Makefile file located in the module motorApp directory, comment the OmsSrc lines, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/motor/6-10/motorApp/Makefile
```



```
TOP = ..
include $(TOP)/configure/CONFIG

# MotorSrc is required for all motor record configurations.  It MUST
appear
# 1st in this list of directories.

DIRS += MotorSrc

# Select/deselect individual device driver modules by removing/adding
a
# "#" comment.

DIRS += DeltaTauSrc
DeltaTauSrc_DEPEND_DIRS = MotorSrc

#DIRS += OmsSrc
#OmsSrc_DEPEND_DIRS    = MotorSrc
```

```
#DIRS += OmsSrc
#OmsSrc_DEPEND_DIRS    = MotorSrc
```

Build the module

```
cd $HOME/epics/R3.15.7/support/motor/6-10
make
```

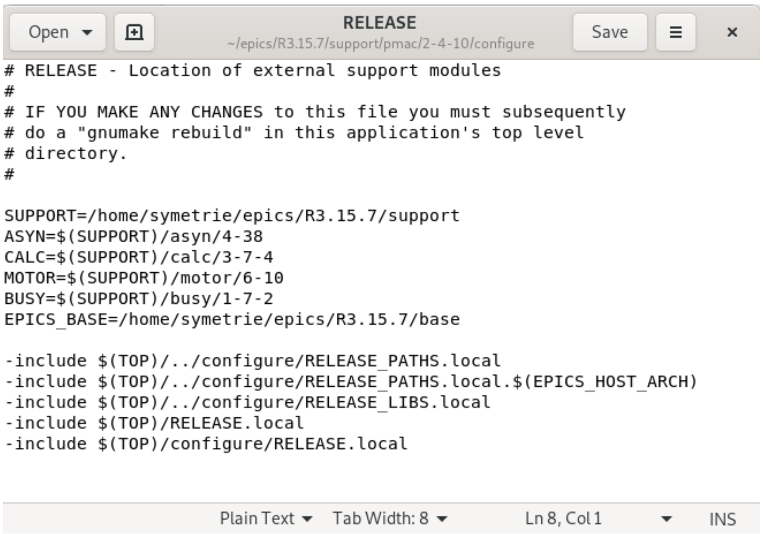
3.4.9 PMAC module

Install the pmac module

```
cd $HOME/epics/R3.15.7/download/support
wget https://github.com/dls-controls/pmac/archive/2-4-10.tar.gz
mv 2-4-10.tar.gz pmac-2-4-10.tar.gz
gunzip -k pmac-2-4-10.tar.gz
tar -xvf pmac-2-4-10.tar
mkdir -p $HOME/epics/R3.15.7/support/pmac
mv pmac-2-4-10 $HOME/epics/R3.15.7/support/pmac/2-4-10
rm pmac-2-4-10.tar
```

Edit the RELEASE file located in the module configure directory, update the SUPPORT, EPICS_BASE, ASYN, CALC, MOTOR and BUSY definitions, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/pmac/2-4-10/configure/RELEASE
```

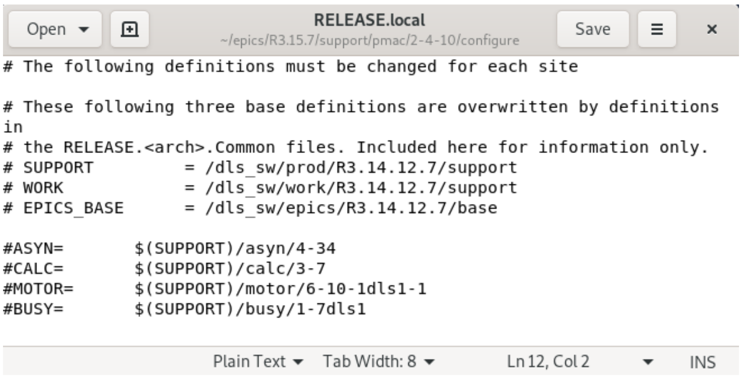


```
# RELEASE - Location of external support modules
#
# IF YOU MAKE ANY CHANGES to this file you must subsequently
# do a "gnumake rebuild" in this application's top level
# directory.
#

SUPPORT=/home/symetrie/epics/R3.15.7/support
ASYN=$(SUPPORT)/asyn/4-38
CALC=$(SUPPORT)/calc/3-7-4
MOTOR=$(SUPPORT)/motor/6-10
BUSY=$(SUPPORT)/busy/1-7-2
EPICS_BASE=/home/symetrie/epics/R3.15.7/base

-include $(TOP)/../configure/RELEASE_PATHS.local
-include $(TOP)/../configure/RELEASE_PATHS.local.$(EPICS_HOST_ARCH)
-include $(TOP)/../configure/RELEASE_LIBS.local
-include $(TOP)/RELEASE.local
-include $(TOP)/configure/RELEASE.local
```

```
SUPPORT=/home/symetrie/epics/R3.15.7/support
ASYN=$(SUPPORT)/asyn/4-38
CALC=$(SUPPORT)/calc/3-7-4
MOTOR=$(SUPPORT)/motor/6-10
BUSY=$(SUPPORT)/busy/1-7-2
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

Edit the RELEASE.local file located in the module configure directory, comment or delete all lines, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/pmac/2-4-10/configure/RELEASE.local
```



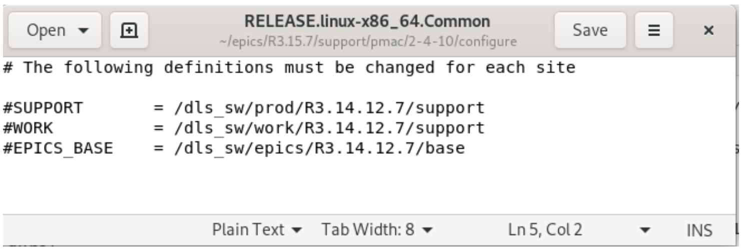
```
# The following definitions must be changed for each site

# These following three base definitions are overwritten by definitions
in
# the RELEASE.<arch>.Common files. Included here for information only.
# SUPPORT         = /dls_sw/prod/R3.14.12.7/support
# WORK            = /dls_sw/work/R3.14.12.7/support
# EPICS_BASE      = /dls_sw/epics/R3.14.12.7/base

#ASYN=        $(SUPPORT)/asyn/4-34
#CALC=        $(SUPPORT)/calc/3-7
#MOTOR=       $(SUPPORT)/motor/6-10-1dls1-1
#BUSY=        $(SUPPORT)/busy/1-7dls1
```

The RELEASE.local file could also simply be deleted.

```
rm $HOME/epics/R3.15.7/support/pmac/2-4-10/configure/RELEASE.local
```

Edit the RELEASE.linux-x86_64.Common file located in the module configure directory, comment or delete all lines, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/pmac/2-4-10/configure/RELEASE.linux-x86_64.Common
```


Open
RELEASE.linux-x86_64.Common
~/epics/R3.15.7/support/pmac/2-4-10/configure
Save
Plain Text
Tab Width: 8
Ln 5, Col 2
INS

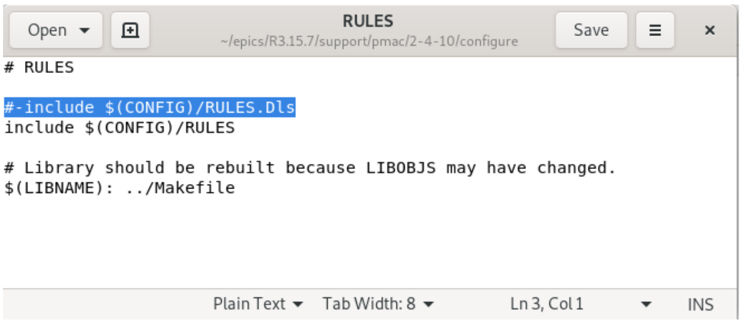
```
# The following definitions must be changed for each site

#SUPPORT       = /dls_sw/prod/R3.14.12.7/support
#WORK          = /dls_sw/work/R3.14.12.7/support
#EPICS_BASE    = /dls_sw/epics/R3.14.12.7/base
```

The RELEASE.linux-x86_64.Common file could also simply be deleted.

```
rm $HOME/epics/R3.15.7/support/pmac/2-4-10/configure/RELEASE.linux-x86_64.Common
```

Edit the RULES files, comment the line with RULES.Dls, save and exit the file.

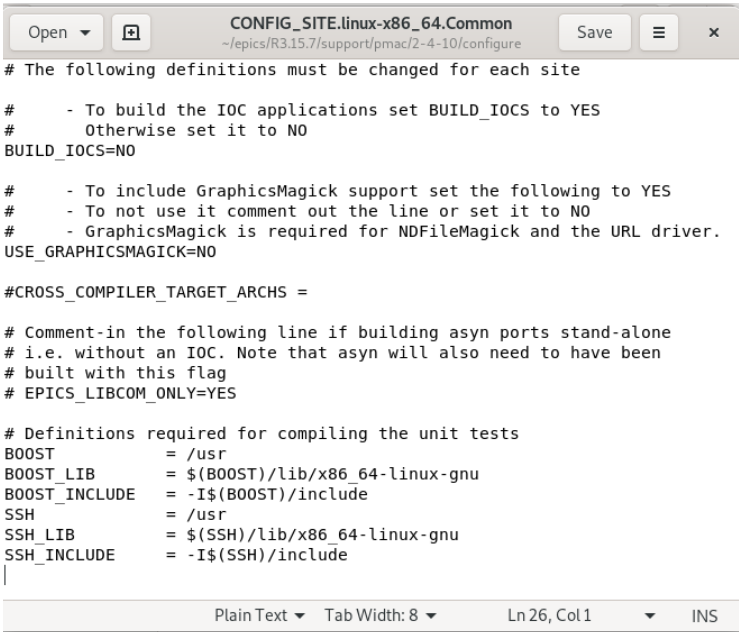
```
gedit $HOME/epics/R3.15.7/support/pmac/2-4-10/configure/RULES
```



```
# RULES

#-include $(CONFIG)/RULES.Dls
include $(CONFIG)/RULES

# Library should be rebuilt because LIBOBJS may have changed.
$(LIBNAME): ../Makefile
```

Edit the CONFIG_SITE.linux-x86_64.Common file located in the module configure directory, update the BOOST and SSH locations to point to the installed versions, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/pmac/2-4-10/configure/CONFIG_SITE.linux-x86_64.Common
```

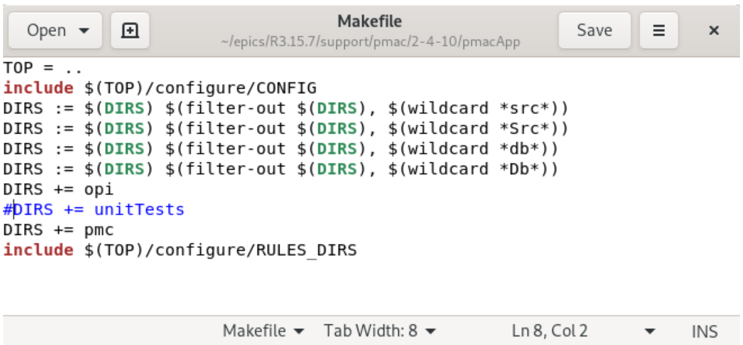

Open
Save
CONFIG_SITE.linux-x86_64.Common
~/epics/R3.15.7/support/pmac/2-4-10/configure
Plain Text
Tab Width: 8
Ln 26, Col 1
INS

```
# The following definitions must be changed for each site

#     - To build the IOC applications set BUILD_IOCS to YES
#       Otherwise set it to NO
BUILD_IOCS=NO

#     - To include GraphicsMagick support set the following to YES
#     - To not use it comment out the line or set it to NO
#     - GraphicsMagick is required for NDFileMagick and the URL driver.
USE_GRAPHICSMAGICK=NO

#CROSS_COMPILER_TARGET_ARCHS =

# Comment-in the following line if building asyn ports stand-alone
# i.e. without an IOC. Note that asyn will also need to have been
# built with this flag
# EPICS_LIBCOM_ONLY=YES

# Definitions required for compiling the unit tests
BOOST          = /usr
BOOST_LIB      = $(BOOST)/lib/x86_64-linux-gnu
BOOST_INCLUDE  = -I$(BOOST)/include
SSH            = /usr
SSH_LIB        = $(SSH)/lib/x86_64-linux-gnu
SSH_INCLUDE    = -I$(SSH)/include
```

```
BOOST         = /usr
BOOST_LIB     = $(BOOST)/lib/x86_64-linux-gnu
BOOST_INCLUDE= -I$(BOOST)/include
SSH           = /usr
SSH_LIB       = $(SSH)/lib/x86_64-linux-gnu
SSH_INCLUDE   = -I$(SSH)/include
```

Edit the Makefile in the pmacApp directory, comment out the unitTests directory, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/pmac/2-4-10/pmacApp/Makefile
```



```
TOP = ..
include $(TOP)/configure/CONFIG
DIRS := $(DIRS) $(filter-out $(DIRS), $(wildcard *src*))
DIRS := $(DIRS) $(filter-out $(DIRS), $(wildcard *Src*))
DIRS := $(DIRS) $(filter-out $(DIRS), $(wildcard *db*))
DIRS := $(DIRS) $(filter-out $(DIRS), $(wildcard *Db*))
DIRS += opi
#DIRS += unitTests
DIRS += pmc
include $(TOP)/configure/RULES_DIRS
```

```
#DIRS += unitTests
```

Build the module

```
cd $HOME/epics/R3.15.7/support/pmac/2-4-10
make
```

3.4.10 PyDM installation

There are many ways in which Python applications can be installed. This guide presents only one way which works for PyDM within the Debian operating system. It is simple to follow and makes use of virtual environments. Ensure python3 is available on the system.

Change directory to where PyDM should be installed

```
cd $HOME/epics
```

Create a python virtual environment for the PyDM installation

```
python3 -m venv pydm
```

Activate the virtual environment

```
source pydm/bin/activate
```

This will alter the bash prompt to signify the user is running the pydm virtual environment, see the “(pydm)” prefix below:

```
(pydm) symetrie@debian:~/epics$
```

Upgrade pip and the virtual environment using pip.

```
pip install --upgrade pip
pip install --upgrade virtualenv
```

Install PyQT5 package

```
pip install PyQt5
```

Install PyDM

```
pip install pydm
```

Verify the installation by running pydm

```
pydm
```

This should open a blank PyDM application instance.

3.4.11 EDM installation

The EDM display manager tool must be installed as an EPICS extension. EPICS extensions are built into an area alongside the base code and are built against the already installed base libraries.

Download and install EPICS extensions:

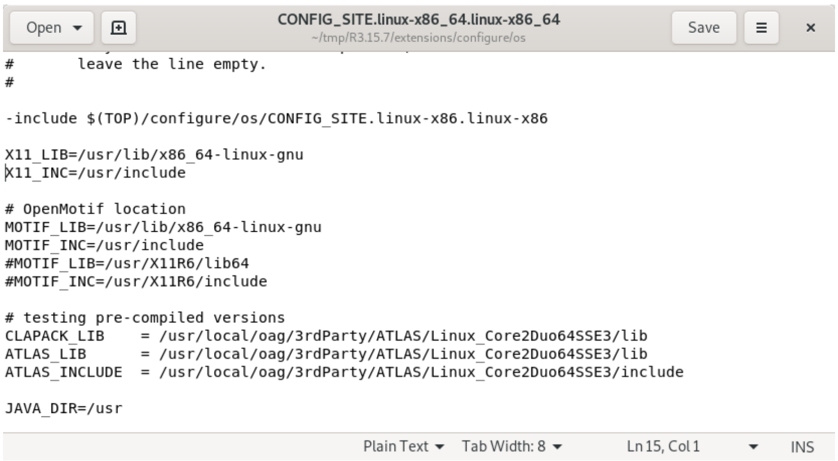
```
mkdir -p $HOME/epics/R3.15.7/download/extensions
cd $HOME/epics/R3.15.7/download/extensions
wget https://epics.anl.gov/download/extensions/extensionsTop_20120904.tar.gz
gunzip -k extensionsTop_20120904.tar.gz
tar -xvf extensionsTop_20120904.tar
mv extensions $HOME/epics/R3.15.7/extensions
rm -r extensionsTop_20120904.tar
```

Download and unpack the EDM application source into the extensions area:

```
cd $HOME/epics/R3.15.7/download/extensions
wget https://github.com/gnartohl/edm/archive/V1-12-105.tar.gz
gunzip -k V1-12-105.tar.gz
tar -xvf V1-12-105.tar
mv edm-1-12-105 $HOME/epics/R3.15.7/extensions/src/edm
rm -r V1-12-105.tar
mv V1-12-105.tar.gz edm-V1-12-105.tar.gz
```

Edit the CONFIG_SITE files to setup the locations of X11 and MOTIF libraries.

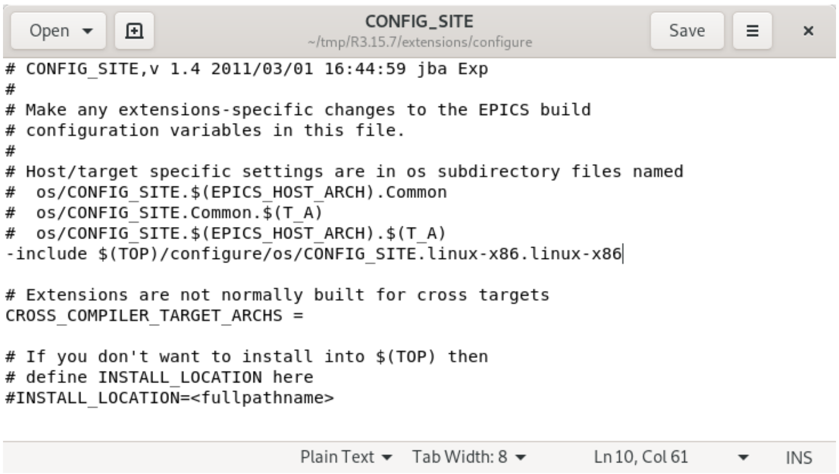
```
gedit $HOME/epics/R3.15.7/extensions/configure/os/CONFIG_SITE.linux-x86_64.linux-x86_64
```


Open
Save
CONFIG_SITE.linux-x86_64.linux-x86_64
~/tmp/R3.15.7/extensions/configure/os
Plain Text
Tab Width: 8
Ln 15, Col 1
INS

```
#       leave the line empty.
#

-include $(TOP)/configure/os/CONFIG_SITE.linux-x86.linux-x86

X11_LIB=/usr/lib/x86_64-linux-gnu
X11_INC=/usr/include

# OpenMotif location
MOTIF_LIB=/usr/lib/x86_64-linux-gnu
MOTIF_INC=/usr/include
#MOTIF_LIB=/usr/X11R6/lib64
#MOTIF_INC=/usr/X11R6/include

# testing pre-compiled versions
CLAPACK_LIB    = /usr/local/oag/3rdParty/ATLAS/Linux_Core2Duo64SSE3/lib
ATLAS_LIB      = /usr/local/oag/3rdParty/ATLAS/Linux_Core2Duo64SSE3/lib
ATLAS_INCLUDE  = /usr/local/oag/3rdParty/ATLAS/Linux_Core2Duo64SSE3/include

JAVA_DIR=/usr
```

```
X11_LIB=/usr/lib/x86_64-linux-gnu
MOTIF_LIB=/usr/lib/x86_64-linux-gnu
```

```
gedit $HOME/epics/R3.15.7/extensions/configure/CONFIG_SITE
```

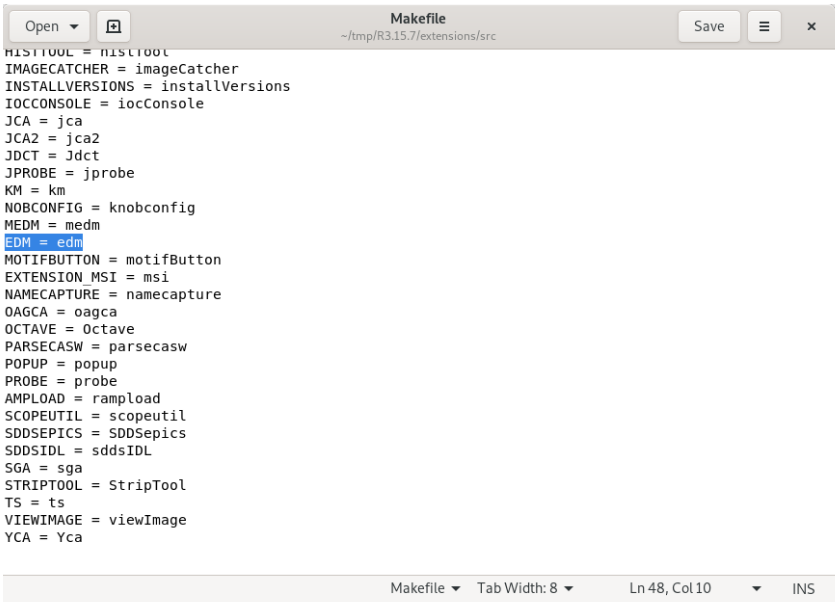


```
# CONFIG_SITE,v 1.4 2011/03/01 16:44:59 jba Exp
#
# Make any extensions-specific changes to the EPICS build
# configuration variables in this file.
#
# Host/target specific settings are in os subdirectory files named
#  os/CONFIG_SITE.$(EPICS_HOST_ARCH).Common
#  os/CONFIG_SITE.Common.$(T_A)
#  os/CONFIG_SITE.$(EPICS_HOST_ARCH).$(T_A)
-include $(TOP)/configure/os/CONFIG_SITE.linux-x86.linux-x86

# Extensions are not normally built for cross targets
CROSS_COMPILER_TARGET_ARCHS =

# If you don't want to install into $(TOP) then
# define INSTALL_LOCATION here
#INSTALL_LOCATION=<fullpathname>
```

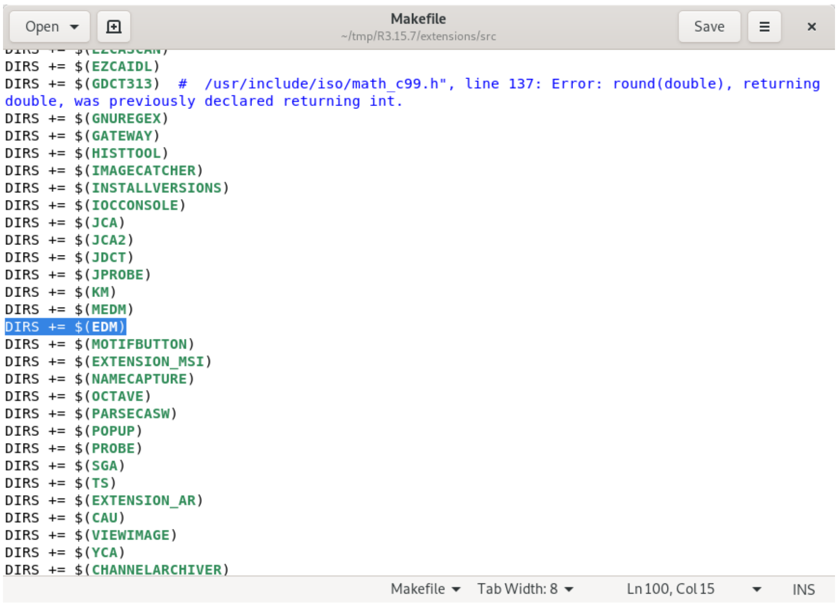
```
-include $(TOP)/configure/os/CONFIG_SITE.linux-x86.linux-x86
```

Edit the source directory Makefile to add the EDM application directory:

```
gedit $HOME/epics/R3.15.7/extensions/src/Makefile
```



```
HISTTOOL = histTool
IMAGECATCHER = imageCatcher
INSTALLVERSIONS = installVersions
IOCCONSOLE = iocConsole
JCA = jca
JCA2 = jca2
JDCT = Jdct
JPROBE = jprobe
KM = km
NOBCONFIG = knobconfig
MEDM = medm
EDM = edm
MOTIFBUTTON = motifButton
EXTENSION_MSI = msi
NAMECAPTURE = namecapture
OAGCA = oagca
OCTAVE = Octave
PARSECASW = parsecasw
POPUP = popup
PROBE = probe
AMPLOAD = rampload
SCOPEUTIL = scopeutil
SDDSEPICS = SDDSepics
SDDSIDL = sddsIDL
SGA = sga
STRIPTOOL = StripTool
TS = ts
VIEWIMAGE = viewImage
YCA = Yca
```



```
DIRS += $(EZCASCAN)
DIRS += $(EZCAIDL)
DIRS += $(GDCT313)  #  /usr/include/iso/math_c99.h", line 137: Error: round(double), returning
double, was previously declared returning int.
DIRS += $(GNUREGEX)
DIRS += $(GATEWAY)
DIRS += $(HISTTOOL)
DIRS += $(IMAGECATCHER)
DIRS += $(INSTALLVERSIONS)
DIRS += $(IOCCONSOLE)
DIRS += $(JCA)
DIRS += $(JCA2)
DIRS += $(JDCT)
DIRS += $(JPROBE)
DIRS += $(KM)
DIRS += $(MEDM)
DIRS += $(EDM)
DIRS += $(MOTIFBUTTON)
DIRS += $(EXTENSION_MSI)
DIRS += $(NAMECAPTURE)
DIRS += $(OCTAVE)
DIRS += $(PARSECASW)
DIRS += $(POPUP)
DIRS += $(PROBE)
DIRS += $(SGA)
DIRS += $(TS)
DIRS += $(EXTENSION_AR)
DIRS += $(CAU)
DIRS += $(VIEWIMAGE)
DIRS += $(YCA)
DIRS += $(CHANNELARCHIVER)
```

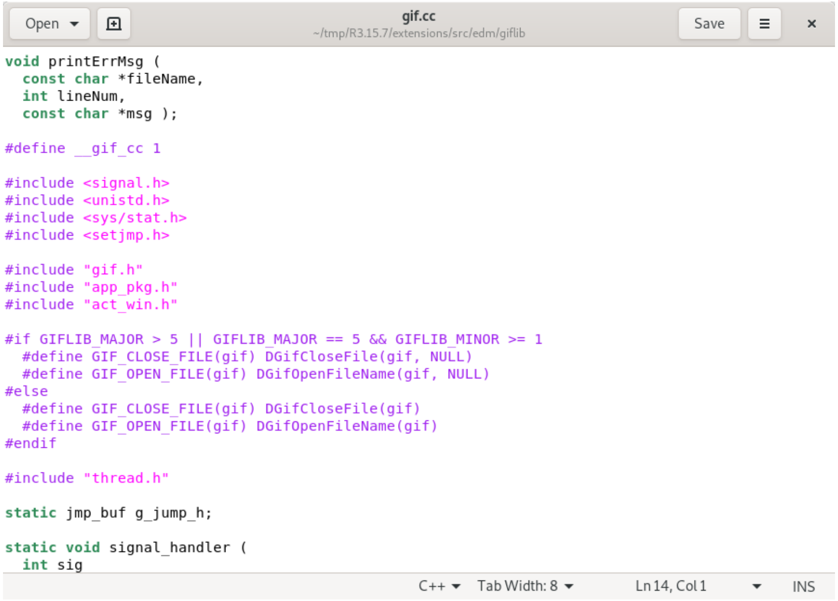
```
EDM = edm
DIRS += $(EDM)
```

Update the giflib Makefile within the EDM source directory:

```
cd $HOME/epics/R3.15.7/extensions/src/edm/giflib
sed -i -e 's| ungif||g' Makefile*
```

Update the source file gif.cc to ensure the OPEN and CLOSE macros are correctly defined:

```
gedit $HOME/epics/R3.15.7/extensions/src/edm/giflib/gif.cc
```



```
void printErrMsg (
  const char *fileName,
  int lineNum,
  const char *msg );

#define __gif_cc 1

#include <signal.h>
#include <unistd.h>
#include <sys/stat.h>
#include <setjmp.h>

#include "gif.h"
#include "app_pkg.h"
#include "act_win.h"

#if GIFLIB_MAJOR > 5 || GIFLIB_MAJOR == 5 && GIFLIB_MINOR >= 1
  #define GIF_CLOSE_FILE(gif) DGifCloseFile(gif, NULL)
  #define GIF_OPEN_FILE(gif) DGifOpenFileName(gif, NULL)
#else
  #define GIF_CLOSE_FILE(gif) DGifCloseFile(gif)
  #define GIF_OPEN_FILE(gif) DGifOpenFileName(gif)
#endif

#include "thread.h"

static jmp_buf g_jump_h;

static void signal_handler (
  int sig
```

```
#include "gif.h"
```

```
#if GIFLIB_MAJOR > 5 || GIFLIB_MAJOR == 5 && GIFLIB_MINOR >= 1
  #define GIF_CLOSE_FILE(gif) DGifCloseFile(gif, NULL)
  #define GIF_OPEN_FILE(gif) DGifOpenFileName(gif, NULL)
#else
  #define GIF_CLOSE_FILE(gif) DGifCloseFile(gif)
  #define GIF_OPEN_FILE(gif) DGifOpenFileName(gif)
#endif
```

Now compile the module:

```
cd $HOME/epics/R3.15.7/extensions
make clean
make
```

The EDM executable application will be compiled and installed into the extensions bin directory.

Finals EDM installation steps, setup the configuration files in the setup directory.

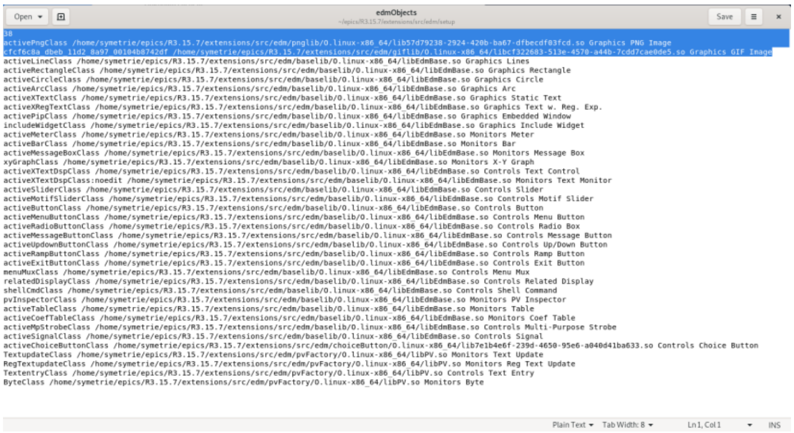
```
cd $HOME/epics/R3.15.7/extensions/src/edm/setup
export HOST_ARCH=linux-x86_64
export EPICS_HOST_ARCH=linux-x86_64
setup.sh
```

Then some images management need to be added. Edit the “edmObjects » file :

```
gedit $HOME/epics/R3.15.7/extensions/src/edm/setup/edmObjects
```

Add +2 to the number (here from 36 to 38) and add the 2 following lines:

```
38
activePngClass /home/symetrie/epics/R3.15.7/extensions/src/edm/pnglib/O.linux-x86_64/lib57d79238-2924-420b-ba67-dfbecdf03fcd.so Graphics PNG Image
cfcf6c8a_dbeb_11d2_8a97_00104b8742df /home/symetrie/epics/R3.15.7/extensions/src/edm/giflib/O.linux-x86_64/libcf322683-513e-4570-a44b-7cdd7cae0de5.so Graphics GIF Image
```



```
38
activePngClass /home/symetrie/epics/R3.15.7/extensions/src/edm/pnglib/O.linux-x86_64/lib57d79238-2924-420b-ba67-dfbecdf03fcd.so Graphics PNG Image
cfcf6c8a_dbeb_11d2_8a97_00104b8742df /home/symetrie/epics/R3.15.7/extensions/src/edm/giflib/O.linux-x86_64/libcf322683-513e-4570-a44b-7cdd7cae0de5.so Graphics GIF Image
activeLineClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Graphics Lines
activeRectangleClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Graphics Rectangle
activeCircleClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Graphics Circle
activeArcClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Graphics Arc
activeXTextClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Graphics Static Text
activeXRegTextClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Graphics Text w. Reg. Exp.
activePipClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Graphics Embedded Window
includeWidgetClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Graphics Include Widget
activeMeterClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Monitors Meter
activeBarClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Monitors Bar
activeMessageBoxClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Monitors Message Box
xyGraphClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Monitors X-Y Graph
activeXTextDspClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Text Control
activeXTextDspClass:noedit /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Monitors Text Monitor
activeSliderClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Slider
activeMotifSliderClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Motif Slider
activeButtonClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Button
activeMenuButtonClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Menu Button
activeRadioButtonClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Radio Box
activeMessageButtonClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Message Button
activeUpdownButtonClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Up/Down Button
activeRampButtonClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Ramp Button
activeExitButtonClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Exit Button
menuMuxClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Menu Mux
relatedDisplayClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Related Display
shellCmdClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Shell Command
pvInspectorClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Monitors PV Inspector
activeTableClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Monitors Table
activeCoefTableClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Monitors Coef Table
activeMpStrobeClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Multi-Purpose Strobe
activeSignalClass /home/symetrie/epics/R3.15.7/extensions/src/edm/baselib/O.linux-x86_64/libEdmBase.so Controls Signal
activeChoiceButtonClass /home/symetrie/epics/R3.15.7/extensions/src/edm/choiceButton/O.linux-x86_64/lib7e1b4e6f-239d-4650-95e6-a040d41ba633.so Controls Choice Button
TextupdateClass /home/symetrie/epics/R3.15.7/extensions/src/edm/pvFactory/O.linux-x86_64/libPV.so Monitors Text Update
RegTextupdateClass /home/symetrie/epics/R3.15.7/extensions/src/edm/pvFactory/O.linux-x86_64/libPV.so Monitors Reg Text Update
TextentryClass /home/symetrie/epics/R3.15.7/extensions/src/edm/pvFactory/O.linux-x86_64/libPV.so Controls Text Entry
ByteClass /home/symetrie/epics/R3.15.7/extensions/src/edm/pvFactory/O.linux-x86_64/libPV.so Monitors Byte
```

Done.

3.4.12 SYMETRIE hexapod module installation

Assuming the required software described above has been installed, unpack the SYMETRIE hexapod EPICS module archive file and cd into the top directory. For this presentation, the “symetrie_hexapod_epics-5.0.7.210830.tar.gz” archive was manually placed in the same download folder as the other downloaded modules.

```
cd $HOME/epics/R3.15.7/download/support
gunzip -k symetrie_hexapod_epics-5.0.7.210830.tar.gz
tar -xvf symetrie_hexapod_epics-5.0.7.210830.tar
mkdir -p $HOME/epics/R3.15.7/support/symetrie_hexapod
mv symetrie_hexapod_epics-5.0.7.210830 $HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830
rm symetrie_hexapod_epics-5.0.7.210830.tar
cd $HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830
```

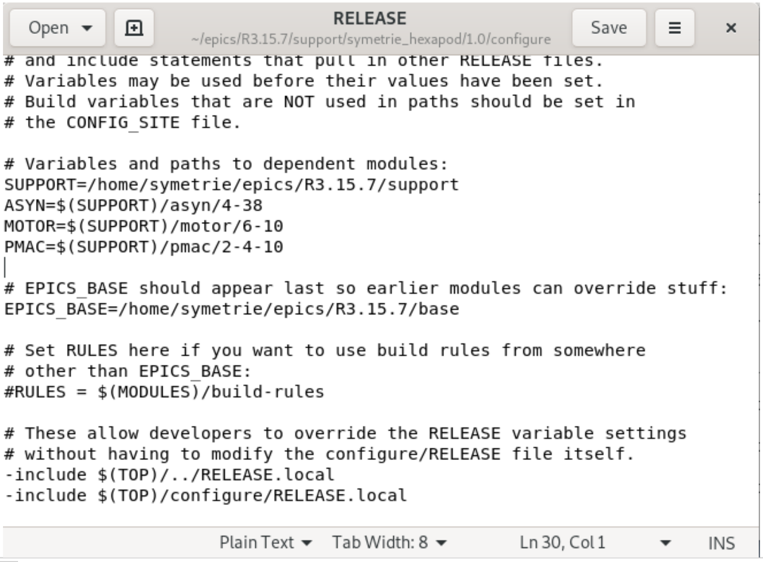
Here, you will see the following files and directories:

Directory	Description
SYMETRIE_EPICSApp	This directory contains the asyn driver code, database templates for control and EDM and PyDM screen files.
configure	The configuration directory.
iocs	The iocs directory contains a demonstration application which controls a single Hexapod instance. This application can be built and executed and demonstrates how to integrate the Hexapod software into an IOC.
Makefile	Top-level Makefile to build all the software.
docs	The directory containing documentation (including this manual).

Before building the support module it is necessary to define the installation locations of the EPICS base installation, asyn, motor and pmac modules.

Edit the RELEASE file located in the module configure directory, update the SUPPORT, EPICS_BASE, ASYN and PMAC definitions, save and exit the file.

```
gedit $HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830/configure/RELEASE
```



```
# and include statements that pull in other RELEASE files.
# Variables may be used before their values have been set.
# Build variables that are NOT used in paths should be set in
# the CONFIG_SITE file.

# Variables and paths to dependent modules:
SUPPORT=/home/symetrie/epics/R3.15.7/support
ASYN=$(SUPPORT)/asyn/4-38
MOTOR=$(SUPPORT)/motor/6-10
PMAC=$(SUPPORT)/pmac/2-4-10

# EPICS_BASE should appear last so earlier modules can override stuff:
EPICS_BASE=/home/symetrie/epics/R3.15.7/base

# Set RULES here if you want to use build rules from somewhere
# other than EPICS_BASE:
#RULES = $(MODULES)/build-rules

# These allow developers to override the RELEASE variable settings
# without having to modify the configure/RELEASE file itself.
-include $(TOP)/../RELEASE.local
-include $(TOP)/configure/RELEASE.local
```

```
SUPPORT=/home/symetrie/epics/R3.15.7/support
ASYN=$(SUPPORT)/asyn/4-38
MOTOR=$(SUPPORT)/motor/6-10
PMAC=$(SUPPORT)/pmac/2-4-10
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

The main record name and device port macros are set during the build procedure. It can be changed by editing the substitutions file. The delivered hexapod-demo application has the following main record prefix name: "SYM:HEX01:".

```
gedit $HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830/iocs/hexapod-
demo/hexapodDemoApp/Db/hexapodDemo_expanded.substitutions
```

Build the module

```
cd $HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830
make clean
make
```

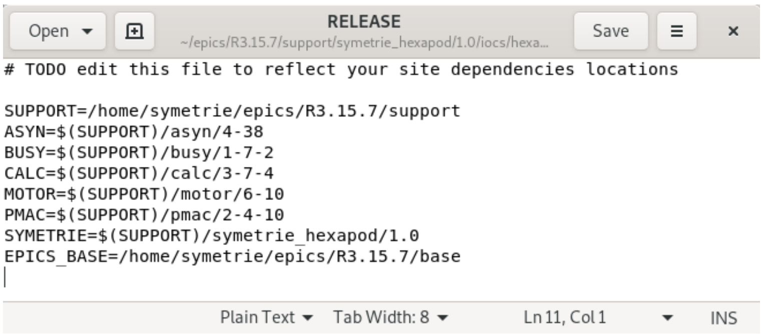
3.4.13 Demonstration application installation

The module is supplied with a demonstration application that will connect to and control a single hexapod. Change directory into the top level of the IOC application.

```
cd $HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830/iocs/hexapod-demo
```

Edit the RELEASE file located in the configure directory, update the SUPPORT, EPICS_BASE, ASYN, BUSY, CALC, MOTOR, PMAC and SYMETRIE definitions, save and exit the file.

```
gedit configure/RELEASE
```

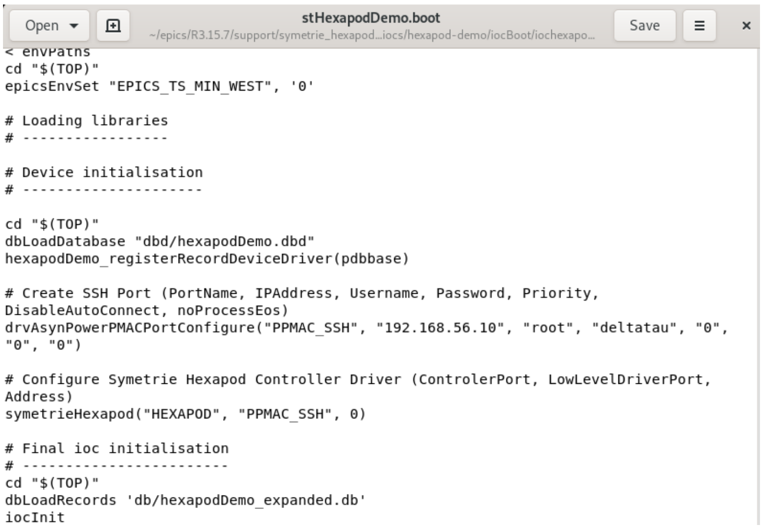


```
# TODO edit this file to reflect your site dependencies locations

SUPPORT=/home/symetrie/epics/R3.15.7/support
ASYN=$(SUPPORT)/asyn/4-38
BUSY=$(SUPPORT)/busy/1-7-2
CALC=$(SUPPORT)/calc/3-7-4
MOTOR=$(SUPPORT)/motor/6-10
PMAC=$(SUPPORT)/pmac/2-4-10
SYMETRIE=$(SUPPORT)/symetrie_hexapod/1.0
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

```
SUPPORT=/home/symetrie/epics/R3.15.7/support
ASYN=$(SUPPORT)/asyn/4-38
BUSY=$(SUPPORT)/busy/1-7-2
CALC=$(SUPPORT)/calc/3-7-4
MOTOR=$(SUPPORT)/motor/6-10
PMAC=$(SUPPORT)/pmac/2-4-10
SYMETRIE=$(SUPPORT)/symetrie_hexapod/5.0.7.210830
EPICS_BASE=/home/symetrie/epics/R3.15.7/base
```

Edit the stHexapodDemo.boot file located in the “iocBoot/iochexapod-demo” folder, update the line that configures the asyn Power PMAC port with the IP address of the SYMETRIE hexapod unit on your network, save and exit the file.

```
gedit iocBoot/iochexapod-demo/stHexapodDemo.boot
```


Open
Save
stHexapodDemo.boot
~/epics/R3.15.7/support/symetrie_hexapod...iocs/hexapod-demo/iocBoot/iochexapo...
×

```
< envPaths
cd "$(TOP)"
epicsEnvSet "EPICS_TS_MIN_WEST", '0'

# Loading libraries
# -----------------

# Device initialisation
# ---------------------

cd "$(TOP)"
dbLoadDatabase "dbd/hexapodDemo.dbd"
hexapodDemo_registerRecordDeviceDriver(pdbbase)

# Create SSH Port (PortName, IPAddress, Username, Password, Priority,
DisableAutoConnect, noProcessEos)
drvAsynPowerPMACPortConfigure("PPMAC_SSH", "192.168.56.10", "root", "deltatau", "0",
"0", "0")

# Configure Symetrie Hexapod Controller Driver (ControlerPort, LowLevelDriverPort,
Address)
symetrieHexapod("HEXAPOD", "PPMAC_SSH", 0)

# Final ioc initialisation
# ------------------------
cd "$(TOP)"
dbLoadRecords 'db/hexapodDemo_expanded.db'
iocInit
```

For example

```
drvAsynPowerPMACPortConfigure("PPMAC_SSH", "192.168.56.10", "root", "deltatau", "0", "0", "0")
```

Now build the example application. The build should complete with no errors.

```
cd $HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830/iocs/hexapod-demo
make clean
make
```

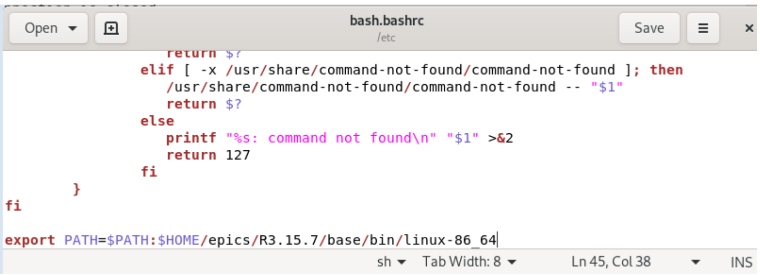
4 RUNNING THE TEST APPLICATION

Once the system has successfully compiled the test application should be executed to ensure good connection to the hexapod.

4.1 EPICS BASE PATH IN TERMINAL DEFAULT PATHS LIST

For user which have access to root:

```
su
gedit /etc/bash.bashrc
```

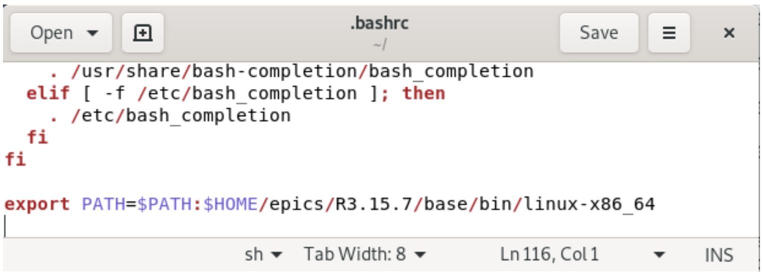


```
                        return $?
                elif [ -x /usr/share/command-not-found/command-not-found ]; then
                   /usr/share/command-not-found/command-not-found -- "$1"
                   return $?
                else
                   printf "%s: command not found\n" "$1" >&2
                   return 127
                fi
        }
fi

export PATH=$PATH:$HOME/epics/R3.15.7/base/bin/linux-86_64
```

```
export PATH=$PATH:$HOME/epics/R3.15.7/base/bin/linux-x86_64
export PATH=$PATH:$HOME/epics/R3.15.7/extensions/bin/linux-x86_64
export EDMPVOBJECTS=$HOME/epics/R3.15.7/extensions/src/edm/setup
export EDMOBJECTS=$HOME/epics/R3.15.7/extensions/src/edm/setup
export EDMFILES=$HOME/epics/R3.15.7/extensions/src/edm/setup
```

For user which do not have access to root:

```
gedit $HOME/.bashrc
```



```
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi

export PATH=$PATH:$HOME/epics/R3.15.7/base/bin/linux-x86_64
```

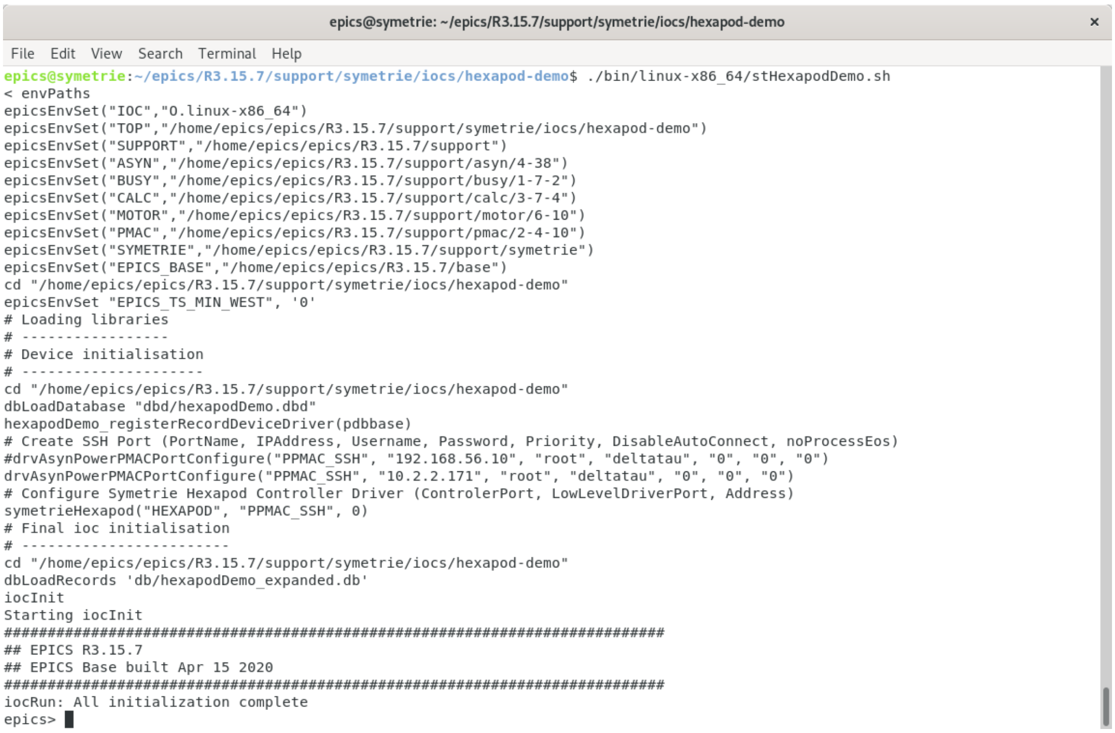
```
export PATH=$PATH:$HOME/epics/R3.15.7/base/bin/linux-x86_64
export PATH=$PATH:$HOME/epics/R3.15.7/extensions/bin/linux-x86_64
export EDMPVOBJECTS=$HOME/epics/R3.15.7/extensions/src/edm/setup
export EDMOBJECTS=$HOME/epics/R3.15.7/extensions/src/edm/setup
export EDMFILES=$HOME/epics/R3.15.7/extensions/src/edm/setup
```

4.2 RUNNING THE EPICS DATABASE

To run the IOC change directory into the demonstration application top level and execute the installed shell script.

```
cd $HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830/iocs/hexapod-demo
./bin/linux-x86_64/stHexapodDemo.sh
```

Below is an example of the output generated when the system is started on Debian 10.


epics@symetrie: ~/epics/R3.15.7/support/symetrie/iocs/hexapod-demo
×
File Edit View Search Terminal Help

```
epics@symetrie:~/epics/R3.15.7/support/symetrie/iocs/hexapod-demo$ ./bin/linux-x86_64/stHexapodDemo.sh
< envPaths
epicsEnvSet("IOC","O.linux-x86_64")
epicsEnvSet("TOP","/home/epics/epics/R3.15.7/support/symetrie/iocs/hexapod-demo")
epicsEnvSet("SUPPORT","/home/epics/epics/R3.15.7/support")
epicsEnvSet("ASYN","/home/epics/epics/R3.15.7/support/asyn/4-38")
epicsEnvSet("BUSY","/home/epics/epics/R3.15.7/support/busy/1-7-2")
epicsEnvSet("CALC","/home/epics/epics/R3.15.7/support/calc/3-7-4")
epicsEnvSet("MOTOR","/home/epics/epics/R3.15.7/support/motor/6-10")
epicsEnvSet("PMAC","/home/epics/epics/R3.15.7/support/pmac/2-4-10")
epicsEnvSet("SYMETRIE","/home/epics/epics/R3.15.7/support/symetrie")
epicsEnvSet("EPICS_BASE","/home/epics/epics/R3.15.7/base")
cd "/home/epics/epics/R3.15.7/support/symetrie/iocs/hexapod-demo"
epicsEnvSet "EPICS_TS_MIN_WEST", '0'
# Loading libraries
# -----------------
# Device initialisation
# ---------------------
cd "/home/epics/epics/R3.15.7/support/symetrie/iocs/hexapod-demo"
dbLoadDatabase "dbd/hexapodDemo.dbd"
hexapodDemo_registerRecordDeviceDriver(pdbbase)
# Create SSH Port (PortName, IPAddress, Username, Password, Priority, DisableAutoConnect, noProcessEos)
#drvAsynPowerPMACPortConfigure("PPMAC_SSH", "192.168.56.10", "root", "deltatau", "0", "0", "0")
drvAsynPowerPMACPortConfigure("PPMAC_SSH", "10.2.2.171", "root", "deltatau", "0", "0", "0")
# Configure Symetrie Hexapod Controller Driver (ControlerPort, LowLevelDriverPort, Address)
symetrieHexapod("HEXAPOD", "PPMAC_SSH", 0)
# Final ioc initialisation
# ------------------------
cd "/home/epics/epics/R3.15.7/support/symetrie/iocs/hexapod-demo"
dbLoadRecords 'db/hexapodDemo_expanded.db'
iocInit
Starting iocInit
############################################################################
## EPICS R3.15.7
## EPICS Base built Apr 15 2020
############################################################################
iocRun: All initialization complete
epics> 
```

Figure 1: IOC output during start-up

Once the start-up has completed and the records have been loaded the SYMETRIE hexapod can be controlled by setting records present in the database.

The available records are presented in chapter 5.1.

The main record name and device port macros are set during the build procedure and can be changed by editing the substitutions file present in the demonstration application database directory. Main record name and device port macros:

```
gedit $HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830/iocs/hexapod-demo/hexapodDemoApp/Db/hexapodDemo_expanded.substitutions
```

The delivered hexapod-demo application has the following main record prefix name: “SYM:HEX01:”.

4.3 Running the PyDM screens

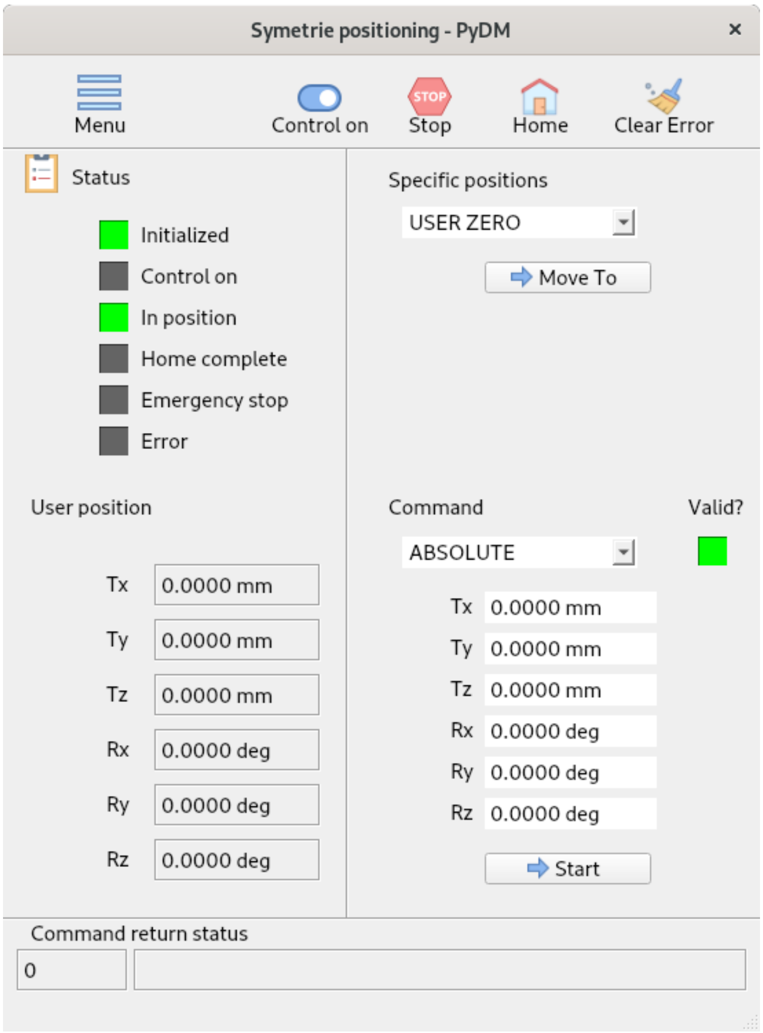
4.3.1 Start the PyDM main screen

To provide a mechanism for testing the example application a set of PyDM screens have been added to the support module. These can be started on Linux by activating the python virtual environment and then executing PyDM with some specific macros set.

```
source $HOME/epics/pydm/bin/activate
cd $HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830/SYMETRIE_EPICSApp/opi/ui
pydm --hide-nav-bar --hide-menu-bar -m "P=SYM,R=:HEX01:"
$HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830/SYMETRIE_EPICSApp/opi/ui/SymetrieHex
apodMain.ui
```

Once started the user is presented with the main screen.

4.3.2 Main screen presentation



Status

🟩 Initialized
■ Control on
🟩 In position
■ Home complete
■ Emergency stop
■ Error

Specific positions: USER ZERO

User position

Tx	0.0000 mm
Ty	0.0000 mm
Tz	0.0000 mm
Rx	0.0000 deg
Ry	0.0000 deg
Rz	0.0000 deg

Command: ABSOLUTE

Valid? 🟩

Tx	0.0000 mm
Ty	0.0000 mm
Tz	0.0000 mm
Rx	0.0000 deg
Ry	0.0000 deg
Rz	0.0000 deg

Command return status: 0

Figure 2: Main PyDM screen

The top of the screen contains buttons to perform the following actions:

- Menu: clicking on this presents a menu where the user can select either "Advanced States" or "Parameters". Clicking on "Advanced States" opens the corresponding PyDM display. Clicking on "Parameters" opens the corresponding PyDM display.
- Control on/Control off: when the status “Control on” is off, the “Control on” button is displayed. Clicking on it executes the C_CONTROLON command. When the status “Control on” is on, the “Control off” button is displayed. Clicking on it executes the C_CONTROLOFF command.
- Stop: clicking on this button executes the C_STOP command.
- Home: clicking on this button executes the C_HOME command.
- Clear Error: clicking on this button executes the C_CLEARERROR command.

The left-hand side of the screen displays the status of the hexapod. The status consists of the following:

- Initialized: controller start-up is terminated.
- Control on: at least one motor is enabled, or in other words at least one motor has its control loop closed.
- In position: no axis is moving.
- Home complete: hexapod home task has been complete successfully.
- Emergency stop: informs that a hardware emergency stop button is pressed.
- Error: an error is present. Errors can be read in the advanced states error screen.

The hexapod user position is displayed below the status indicators.

The right-hand side of the display presents the positioning commands available. It is divided into two parts:

- Specific positions section allows to move to predefined positions.
- Command section allows to start a movement defined in a selected move type.

The specific positions section allows to move to predefined positions. The drop-down list should give access to the following three specific positions:

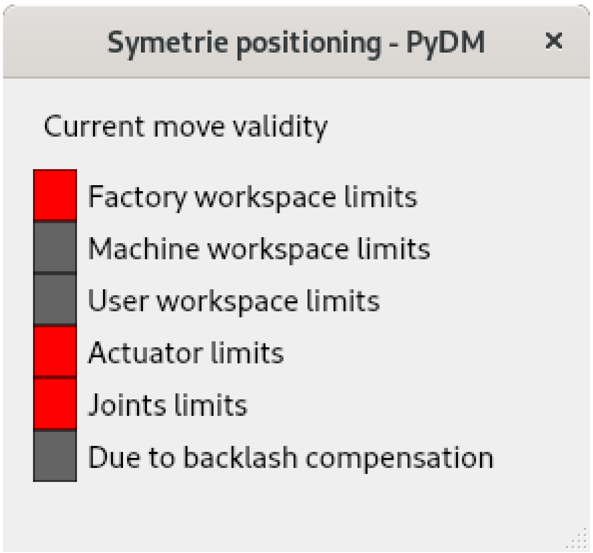
- “USER ZERO”: zero position (Tx=Ty=Tz=Rx=Ry=Rz=0) expressed in the User coordinate system. It is equivalent to an absolute zero command sent from the Command section below.
- “RETRACTED”: refers to an hexapod position were actuators are retracted. In that position the hexapod is close to its smallest size. It is useful for loading, storage, transportation. The position is expressed in the Machine coordinate system.
- “MACHINE ZERO”: zero position (Tx=Ty=Tz=Rx=Ry=Rz=0) expressed in the Machine coordinate system. Most of the time, it refers to the middle travel range of the hexapod.

The Move To button starts the movement.

Below the specific positions section is the Command section which allows the user to start a movement defined in a selected move type. The move type should be selected from the drop-down list:

- Absolute
- Relative object
- Relative user

Tx, Ty, Tz, Rx, Ry and Rz values can be chosen and will be applied according to the selected move type (Absolute, Relative User or Relative Object). The Start button launch the movement if the positioning command is valid according to the hexapod limits. The validity of the proposed move is shown by the LED next to Command drop down menu. If the LED is red then the move is invalid and clicking on the LED opens a new window which displays the reason(s) why the proposed move is invalid. See example below.



Symetrie positioning - PyDM ×

Current move validity

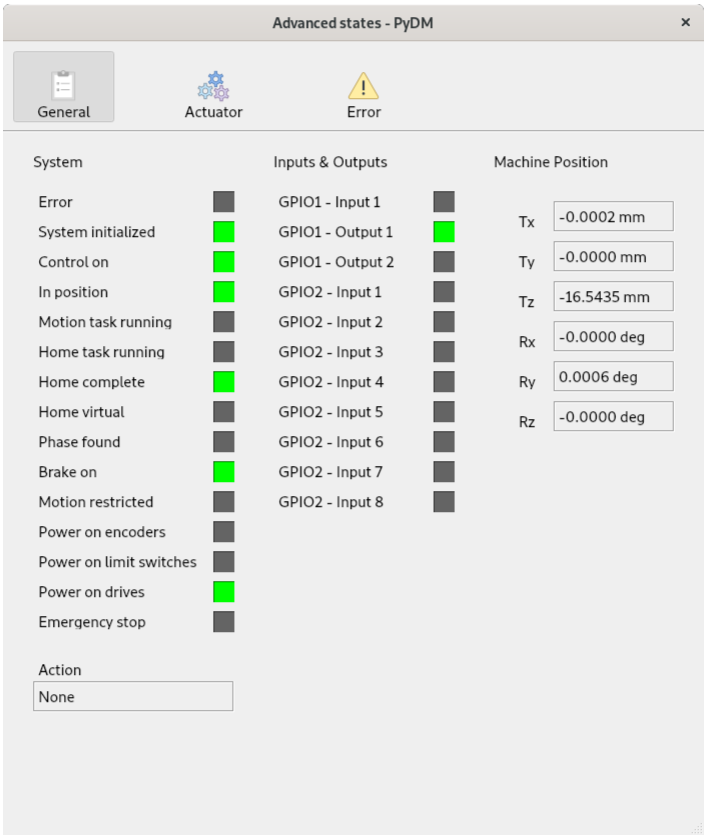
🟥 Factory workspace limits
⬛ Machine workspace limits
⬛ User workspace limits
🟥 Actuator limits
🟥 Joints limits
⬛ Due to backlash compensation

Figure 3: Move validity screen

At the bottom of the main screen the command execution status is displayed. If the value is zero then the latest command has been successfully executed. If any command fails then the return value is reported and the corresponding error message is displayed.

4.3.3 Advanced states screen presentation

Clicking on the "Menu" and selecting the "Advanced states" option opens the advanced states general screen.



System

Error	■
System initialized	🟩
Control on	🟩
In position	🟩
Motion task running	■
Home task running	■
Home complete	🟩
Home virtual	■
Phase found	■
Brake on	🟩
Motion restricted	■
Power on encoders	■
Power on limit switches	■
Power on drives	🟩
Emergency stop	■

Action: None

Inputs & Outputs

GPIO1 - Input 1	■
GPIO1 - Output 1	🟩
GPIO1 - Output 2	■
GPIO2 - Input 1	■
GPIO2 - Input 2	■
GPIO2 - Input 3	■
GPIO2 - Input 4	■
GPIO2 - Input 5	■
GPIO2 - Input 6	■
GPIO2 - Input 7	■
GPIO2 - Input 8	■

Machine Position

Tx	-0.0002 mm
Ty	-0.0000 mm
Tz	-16.5435 mm
Rx	-0.0000 deg
Ry	0.0006 deg
Rz	-0.0000 deg

Figure 4: Advanced states general screen

The General screen is divided into four sections:

- System: retrieves main controller status
- Action: display the current running action
- Inputs & outputs: retrieves states of the inputs and outputs
- Machine position: hexapod position expressed in Machine coordinate system.

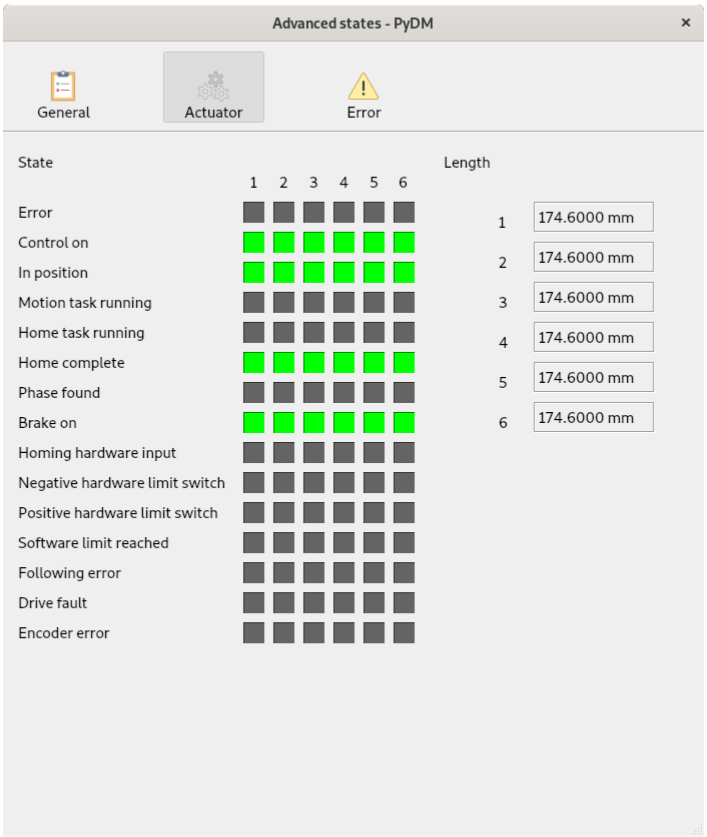
The system status is described fully in the Application programming interface (API) manual. The individual LEDs represent a single bit from the bitmask returned from the s_hexa value. Below the system indicators the widget displays the current action which is executing. The following actions are possible:

- None
- Stop
- ControlOn
- ControlOff
- Home

- HomeVirtual
- MovePTP
- MoveSpecificPos
- MoveSequence
- MoveJog
- Handwheel
- Maintenance

The GPIO displays the current status of the individual GPIO signals. On the right of the screen the current machine position is displayed in the machine coordinate system.

Clicking on the Actuator button at the top of the screen switches to the Actuator screen.



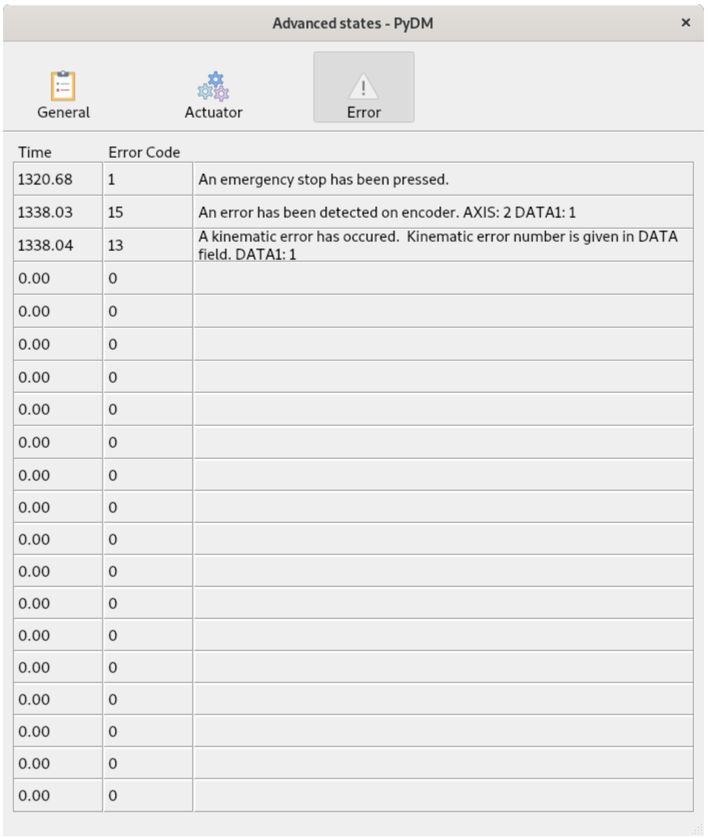
State	1	2	3	4	5	6
Error	■	■	■	■	■	■
Control on	🟩	🟩	🟩	🟩	🟩	🟩
In position	🟩	🟩	🟩	🟩	🟩	🟩
Motion task running	■	■	■	■	■	■
Home task running	■	■	■	■	■	■
Home complete	🟩	🟩	🟩	🟩	🟩	🟩
Phase found	■	■	■	■	■	■
Brake on	🟩	🟩	🟩	🟩	🟩	🟩
Homing hardware input	■	■	■	■	■	■
Negative hardware limit switch	■	■	■	■	■	■
Positive hardware limit switch	■	■	■	■	■	■
Software limit reached	■	■	■	■	■	■
Following error	■	■	■	■	■	■
Drive fault	■	■	■	■	■	■
Encoder error	■	■	■	■	■	■

Length	
1	174.6000 mm
2	174.6000 mm
3	174.6000 mm
4	174.6000 mm
5	174.6000 mm
6	174.6000 mm

Figure 5: Advanced states actuator screen

The actuator screen contains the full status for each individual axis of the hexapod, along with the raw lengths (in mm) of each axis. The actuator status is described fully in Application programming interface (API) manual. The individual LEDs represent a single bit from the bitmask returned from the specific axis status variable (s_ax_1..6).

Clicking on the Error button at the top of the screen switches to the Error screen.



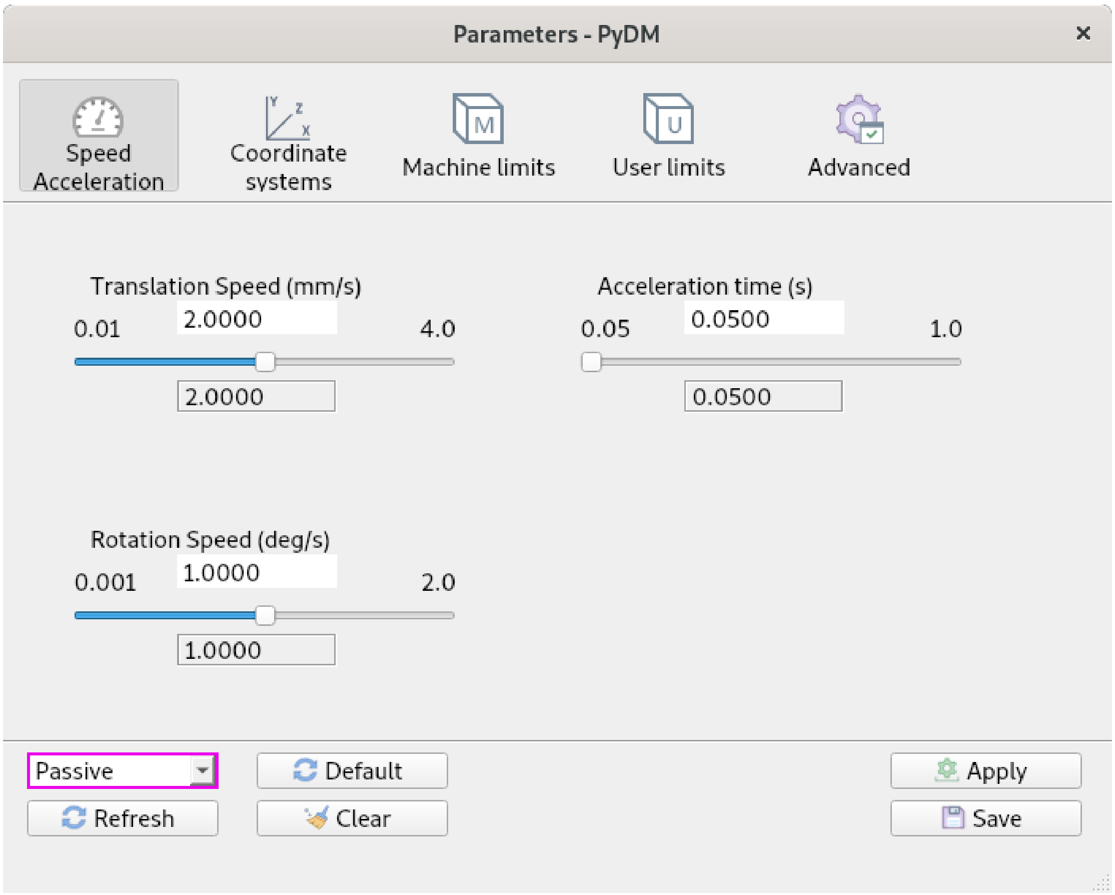
Time	Error Code	
1320.68	1	An emergency stop has been pressed.
1338.03	15	An error has been detected on encoder. AXIS: 2 DATA1: 1
1338.04	13	A kinematic error has occured. Kinematic error number is given in DATA field. DATA1: 1
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	

Figure 6: Advanced states error screen

The error screen displays a table of errors present on the hexapod controller. Each error that occurs is displayed with the time at which the error occurred, a numerical value representing the error code and a description of the error itself. Any additional information required to specify the error conditions is appended to the error message. In the figure above three errors have occurred. Errors can only be cleared by clicking on the "Clear errors" button on the main display.

4.3.4 Parameters screen presentation

Returning to the main screen, clicking on the "Menu" and selecting the "Parameters" option opens the parameters speed/acceleration screen.



Parameters - PyDM

Translation Speed (mm/s): 0.01 – 4.0, demand 2.0000, read back 2.0000

Acceleration time (s): 0.05 – 1.0, demand 0.0500, read back 0.0500

Rotation Speed (deg/s): 0.001 – 2.0, demand 1.0000, read back 1.0000

Figure 7: Parameters speed / acceleration screen

All parameter screens contain the same block of widgets at the bottom. Parameter values by default are only read out of the hardware controller when a new value is applied, or when defaults are loaded. A manual read of parameters is possible by clicking on the "Refresh" button. It is also possible to set an automatic refresh rate by change the "Passive" value in the drop-down menu to one of the other scan rates. Whenever a read of parameters is executed, the corresponding EPICS read back records (those whose names contain _RBV) will be updated.

All parameters have demand records and read back records, represented by the two sets of widgets for each parameter seen above. The demand can be changed and when the "Apply" button is clicked the current demands (of all parameters) are written to the control hardware.

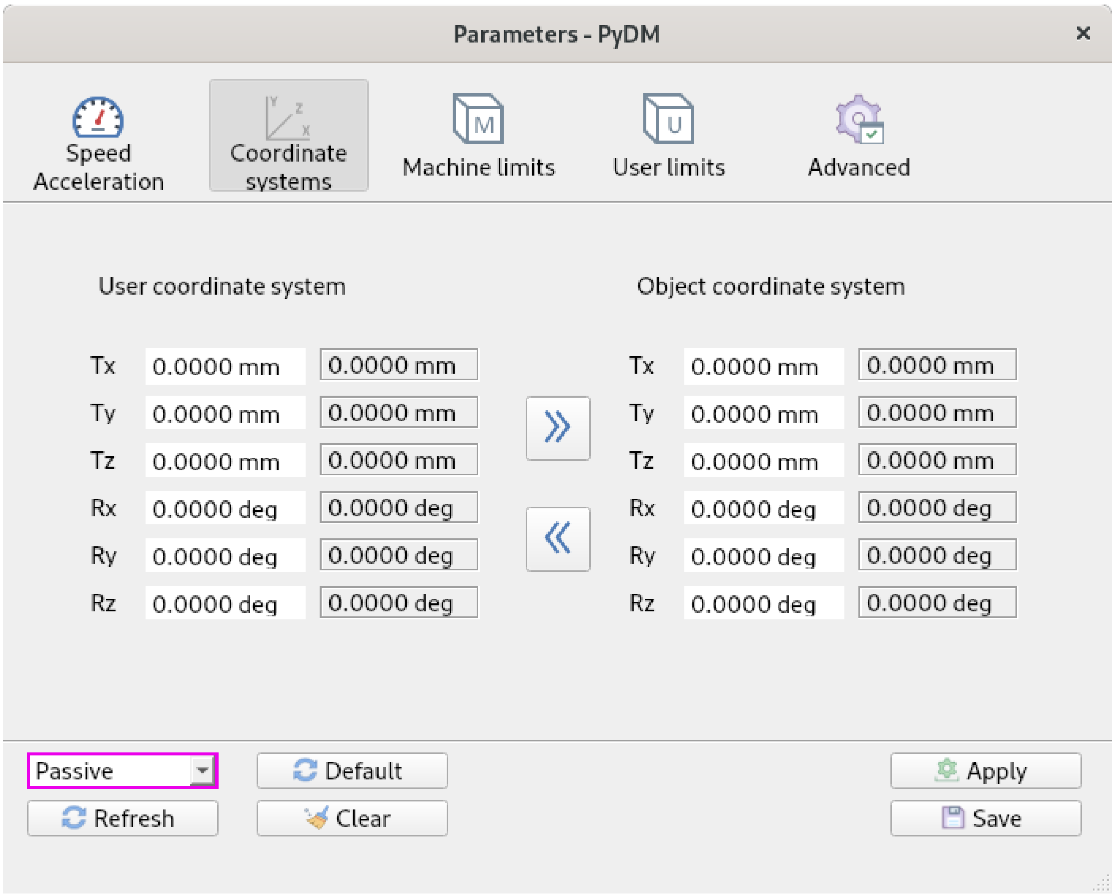
The default values can be re-applied by clicking on the "Default" button. After a default reset, all demand records are updated to match the read back records.

Clicking on the "Clear" button will reset any changed demand values back to the currently read back value.

Clicking on the "Save" button will save the currently applied demands so that they persist after a reboot of the control hardware.

The speed / acceleration screen allows the user to configure the speed and acceleration time used for the positioning of the hexapod. Speeds are specified separately for the translation and rotation. New demand values can either be typed into the demand boxes or the sliders can be used. Minimum and maximum values are shown as the limits of the slider widgets.

Clicking on the "Coordinate Systems" button at the top of the screen opens the parameters coordinate systems screen.



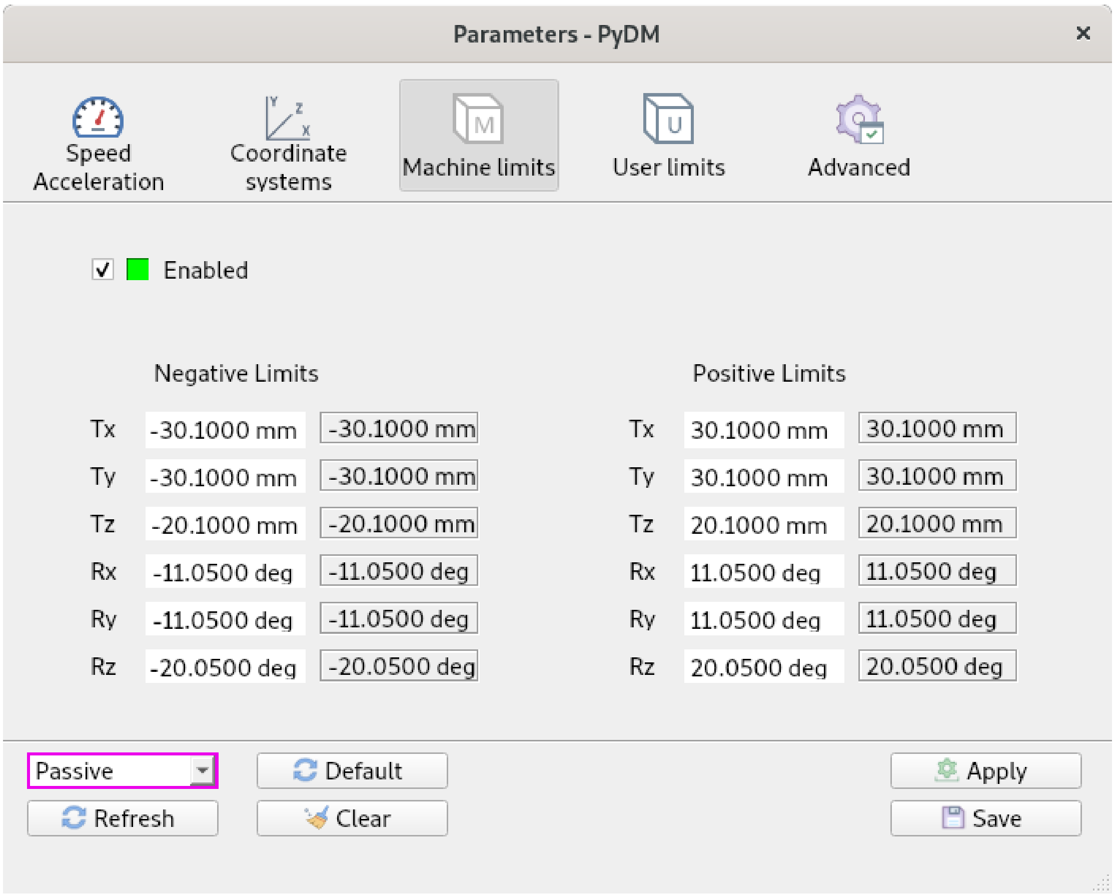
User coordinate system			Object coordinate system		
Tx	0.0000 mm	0.0000 mm	Tx	0.0000 mm	0.0000 mm
Ty	0.0000 mm	0.0000 mm	Ty	0.0000 mm	0.0000 mm
Tz	0.0000 mm	0.0000 mm	Tz	0.0000 mm	0.0000 mm
Rx	0.0000 deg	0.0000 deg	Rx	0.0000 deg	0.0000 deg
Ry	0.0000 deg	0.0000 deg	Ry	0.0000 deg	0.0000 deg
Rz	0.0000 deg	0.0000 deg	Rz	0.0000 deg	0.0000 deg

Figure 8: Parameters coordinate systems screen

The coordinate systems parameters are explained in detail in Application programming interface (API) manual. This screen can be used to define both User and Object coordinate systems. In most applications, the User and Object coordinate systems must be identical. The arrowed buttons can be used to copy the demands of one coordinate system into the demands of the other. Note that clicking on the arrowed buttons will not apply the changes, it will simply setup the demands ready to be applied.

Coordinate system parameters cannot be applied if a motion task is running.

Clicking on the "Machine limits" button at the top of the screen opens the parameters Machine limits screen.



☑ 🟩 Enabled

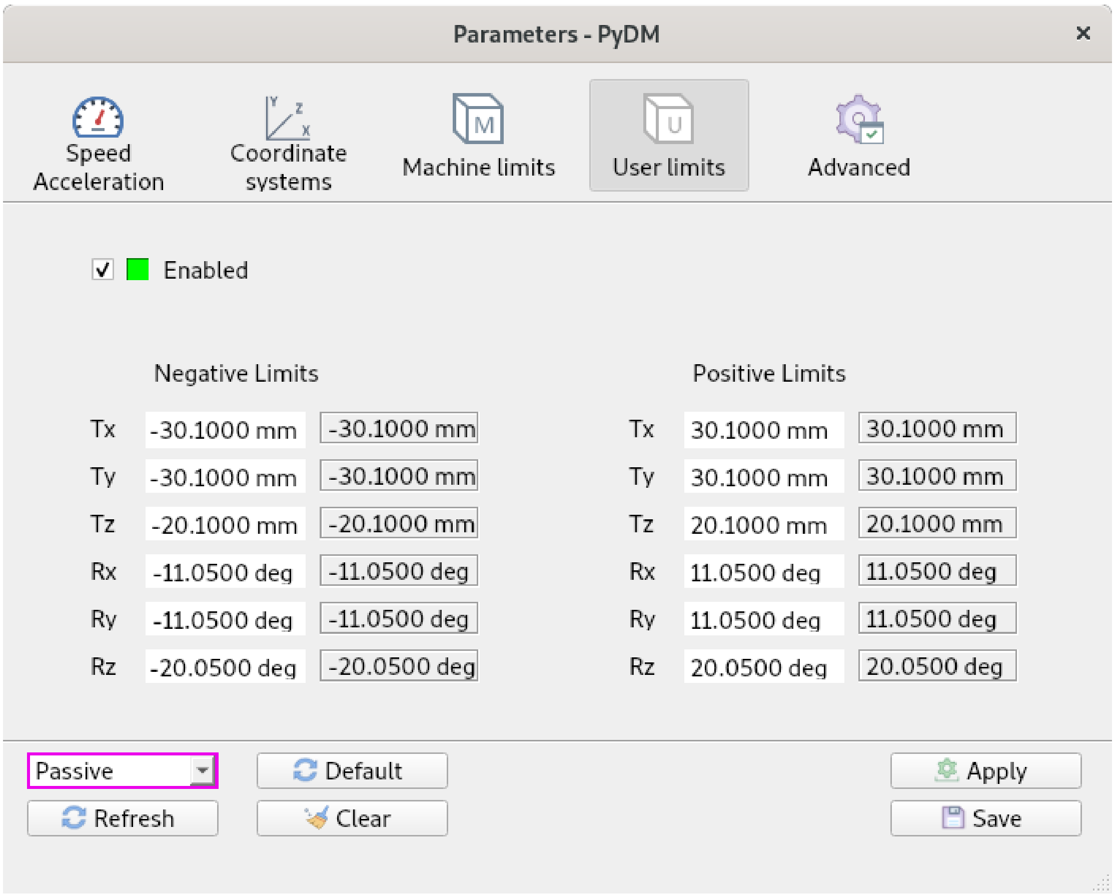
	Negative Limits			Positive Limits	
Tx	-30.1000 mm	-30.1000 mm	Tx	30.1000 mm	30.1000 mm
Ty	-30.1000 mm	-30.1000 mm	Ty	30.1000 mm	30.1000 mm
Tz	-20.1000 mm	-20.1000 mm	Tz	20.1000 mm	20.1000 mm
Rx	-11.0500 deg	-11.0500 deg	Rx	11.0500 deg	11.0500 deg
Ry	-11.0500 deg	-11.0500 deg	Ry	11.0500 deg	11.0500 deg
Rz	-20.0500 deg	-20.0500 deg	Rz	20.0500 deg	20.0500 deg

Passive

Figure 9: Parameters machine limits screen

The machine limits screen provides widgets for setting the negative and positive Machine limits, as well as enabling or disabling the limit enforcement.

Clicking on the "User limits" button at the top of the screen opens the parameters User limits screen.



☑ 🟩 Enabled

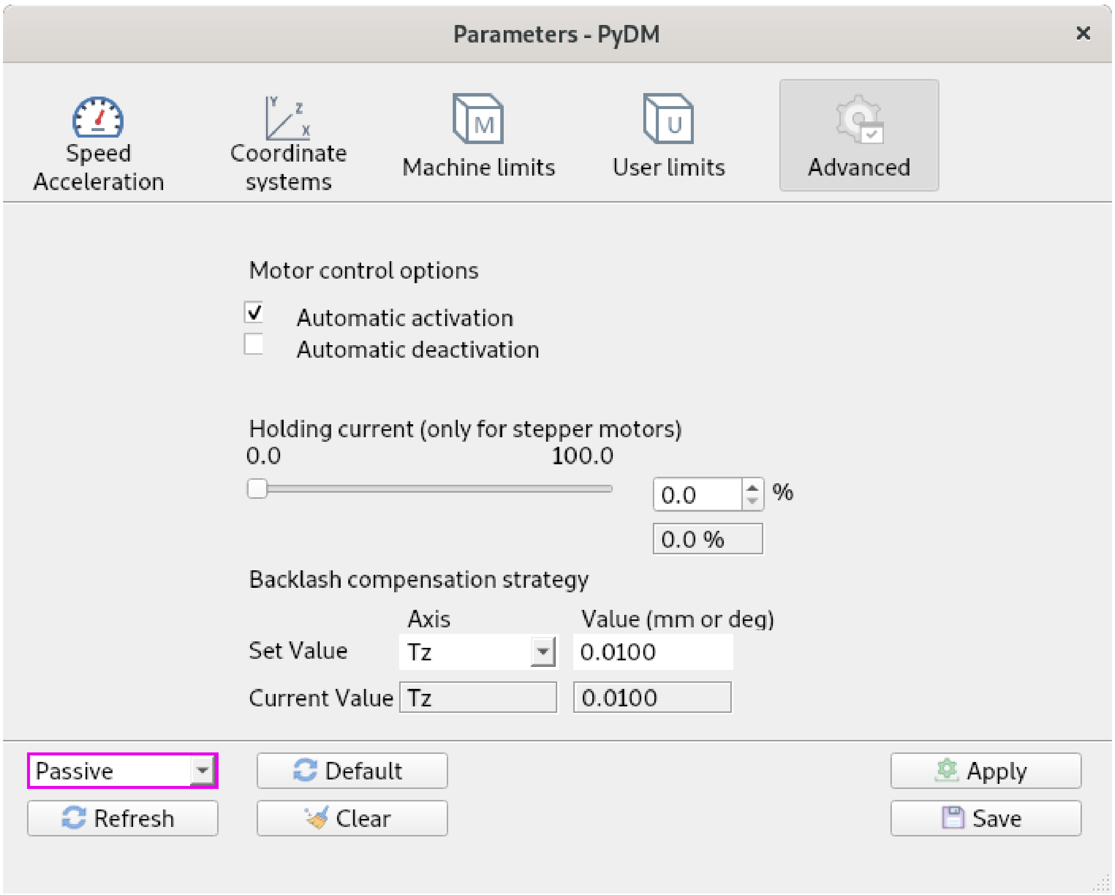
	Negative Limits			Positive Limits	
Tx	-30.1000 mm	-30.1000 mm	Tx	30.1000 mm	30.1000 mm
Ty	-30.1000 mm	-30.1000 mm	Ty	30.1000 mm	30.1000 mm
Tz	-20.1000 mm	-20.1000 mm	Tz	20.1000 mm	20.1000 mm
Rx	-11.0500 deg	-11.0500 deg	Rx	11.0500 deg	11.0500 deg
Ry	-11.0500 deg	-11.0500 deg	Ry	11.0500 deg	11.0500 deg
Rz	-20.0500 deg	-20.0500 deg	Rz	20.0500 deg	20.0500 deg

Passive

Figure 10: Parameters user limits screen

The user limits screen provides widgets for setting the negative and positive User limits, as well as enabling or disabling the limit enforcement.

Clicking on the "Advanced" button at the top of the screen opens the parameters advanced screen.



Motor control options

☑ Automatic activation
□ Automatic deactivation

Holding current (only for stepper motors)
0.0 100.0
0.0 %
0.0 %

Backlash compensation strategy

	Axis	Value (mm or deg)
Set Value	Tz	0.0100
Current Value	Tz	0.0100

Figure 11: Parameter advanced screen

The advanced screen provides widgets to set the motor control parameters. Automatic activation will activate the motor control loop when a motion task is executed. If this is set, then it is not necessary to turn control on from the main screen. Automatic deactivation will automatically disable the motor control loop whenever the hexapod has reached the requested position. The holding current for stepper motors can be set, it is specified as a percentage (between 0 and 100%) of the nominal motor current. The backlash compensation strategy can be selected, with all positioning tasks always finishing with the same approach movement. A detailed explanation of these parameters can be found in Application programming interface (API) manual.

4.4 Running the EDM screens

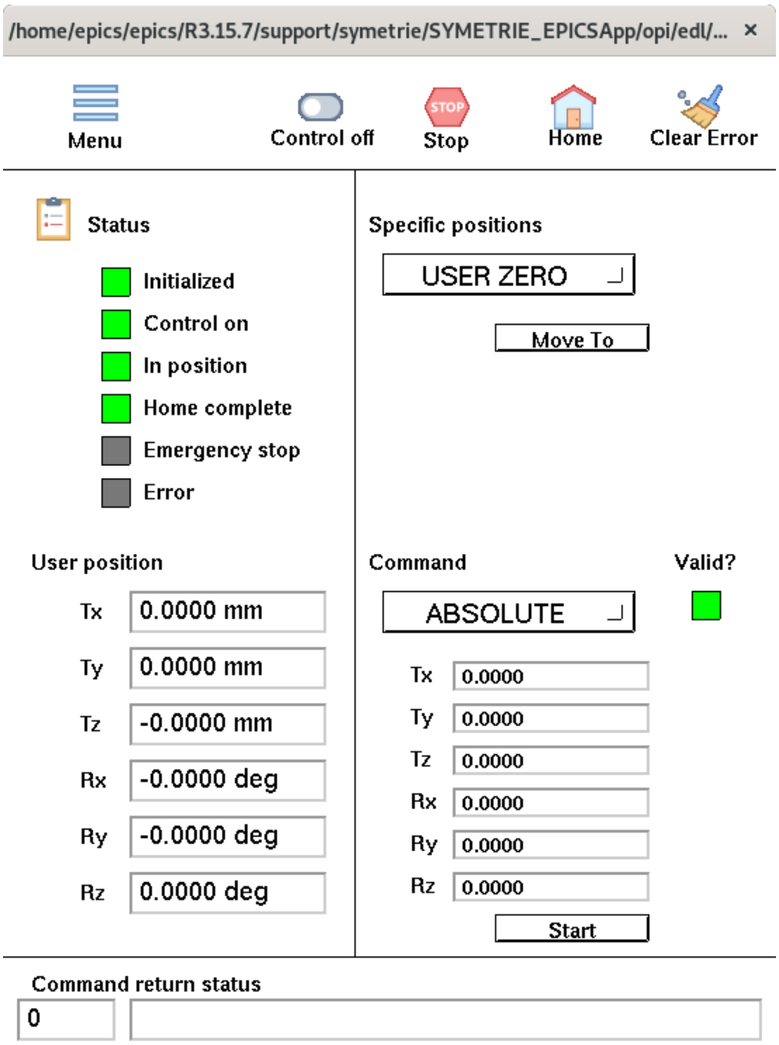
4.4.1 Start the EDM main screen

As well as the PyDM screens a set of EDM engineering screens have been provided. To provide a mechanism for testing the example application a set of EDM engineering screens have been added to the support module; these can be started on Linux by executing EDM with some specific macros set.

```
cd $HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830/SYMETRIE_EPICSApp/opi/edl
edm -m "P=SYM,R=:HEX01:" -x
$HOME/epics/R3.15.7/support/symetrie_hexapod/5.0.7.210830/SYMETRIE_EPICSApp/opi/edl/SymetrieHe
xapodMain.edl
```

Once started the user is presented with the main screen.

4.4.2 Main screen presentation


/home/epics/epics/R3.15.7/support/symetrie/SYMETRIE_EPICSApp/opi/edl/... ×
Menu
Control off
Stop
Home
Clear Error
Status
Specific positions
Move To
Start

🟩 Initialized
🟩 Control on
🟩 In position
🟩 Home complete
■ Emergency stop
■ Error

Specific positions: USER ZERO

User position	
Tx	0.0000 mm
Ty	0.0000 mm
Tz	-0.0000 mm
Rx	-0.0000 deg
Ry	-0.0000 deg
Rz	0.0000 deg

Command: ABSOLUTE — Valid? 🟩

Command	
Tx	0.0000
Ty	0.0000
Tz	0.0000
Rx	0.0000
Ry	0.0000
Rz	0.0000

Command return status: 0

Figure 12: Main EDM screen

The top of the screen contains buttons to perform the following actions:

- Menu: clicking on this presents a menu where the user can select either "Advanced States" or "Parameters". Clicking on "Advanced States" opens the corresponding EDM display. Clicking on "Parameters" opens the corresponding EDM display.
- Control on/Control off: when the status “Control on” is off, the “Control on” button is displayed. Clicking on it executes the C_CONTROLON command. When the status “Control on” is on, the “Control off” button is displayed. Clicking on it executes the C_CONTROLOFF command.
- Stop: clicking on this button executes the C_STOP command.
- Home: clicking on this button executes the C_HOME command.
- Clear Error: clicking on this button executes the C_CLEARERROR command.

The left-hand side of the screen displays the status of the hexapod. The status consists of the following:

- Initialized: controller start-up is terminated.
- Control on: at least one motor is enabled, or in other words at least one motor has its control loop closed.
- In position: no axis is moving.
- Home complete: hexapod home task has been complete successfully.
- Emergency stop: informs that a hardware emergency stop button is pressed.
- Error: an error is present. Errors can be read in the advanced states error screen.

The hexapod user position is displayed below the status indicators.

The right-hand side of the display presents the positioning commands available. It is divided into two parts:

- Specific positions section allows to move to predefined positions.
- Command section allows to start a movement defined in a selected move type.

The Specific positions section allows to move to predefined positions. The drop-down list should give access to the following three specific positions:

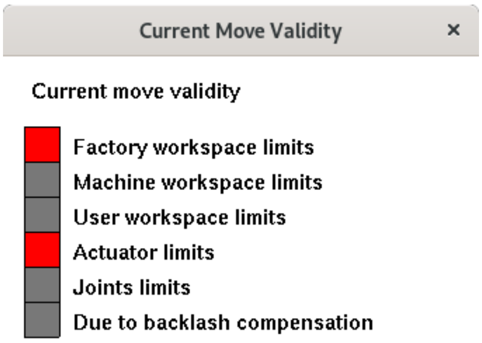
- “USER ZERO”: zero position (Tx=Ty=Tz=Rx=Ry=Rz=0) expressed in the User coordinate system. It is equivalent to an absolute zero command sent from the Command section below.
- “RETRACTED”: refers to a hexapod position were actuators are retracted. In that position the hexapod is close to its smallest size. It is useful for loading, storage, transportation. The position is expressed in the Machine coordinate system.
- “MACHINE ZERO”: zero position (Tx=Ty=Tz=Rx=Ry=Rz=0) expressed in the Machine coordinate system. Most of the time, it refers to the middle travel range of the hexapod.

The Move To button starts the movement.

Below the Specific positions section is the Command section which allows the user to start a movement defined in a selected move type. The move type should be selected from the drop-down list:

- Absolute
- Relative object
- Relative user

Tx, Ty, Tz, Rx, Ry and Rz values can be chosen and will be applied according to the selected move type (Absolute, Relative User or Relative Object). The Start button starts the movement if the positioning command is valid according to the hexapod limits. The validity of the proposed move is shown by the LED next to Command drop down menu. If the LED is red then the move is invalid and clicking on the LED opens a new window which displays the reason(s) why the proposed move is invalid. See example below.



Current move validity

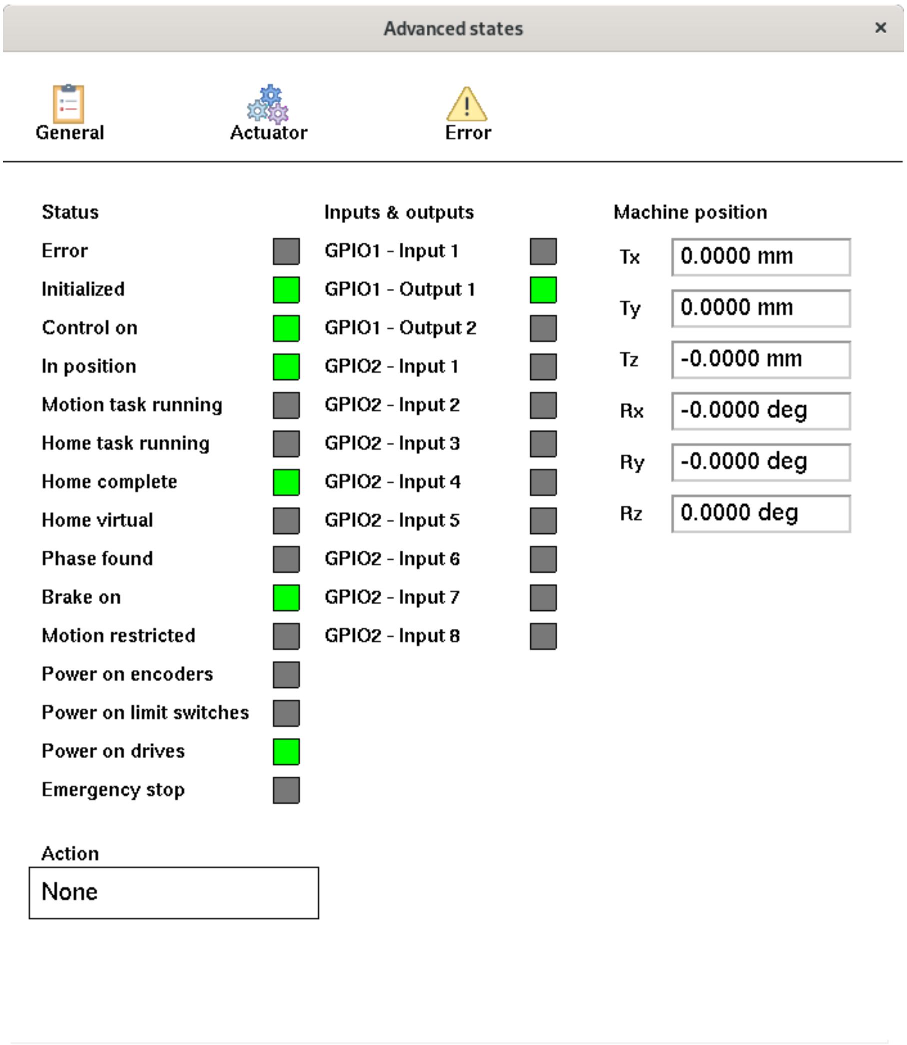
🟥 Factory workspace limits
⬛ Machine workspace limits
⬛ User workspace limits
🟥 Actuator limits
⬛ Joints limits
⬛ Due to backlash compensation

Figure 13: Move validity screen

At the bottom of the main screen the command execution status is displayed. If the value is zero then the latest command has been successfully executed. If any command fails then the return value is reported and the corresponding error message is displayed.

4.4.3 Advanced states screen presentation

Clicking on the "Menu" and selecting the "Advanced states" option opens the advanced states general screen.



Status	
Error	◼
Initialized	🟩
Control on	🟩
In position	🟩
Motion task running	◼
Home task running	◼
Home complete	🟩
Home virtual	◼
Phase found	◼
Brake on	🟩
Motion restricted	◼
Power on encoders	◼
Power on limit switches	◼
Power on drives	🟩
Emergency stop	◼

Action

None

Inputs & outputs	
GPIO1 - Input 1	◼
GPIO1 - Output 1	🟩
GPIO1 - Output 2	◼
GPIO2 - Input 1	◼
GPIO2 - Input 2	◼
GPIO2 - Input 3	◼
GPIO2 - Input 4	◼
GPIO2 - Input 5	◼
GPIO2 - Input 6	◼
GPIO2 - Input 7	◼
GPIO2 - Input 8	◼

Machine position	
Tx	0.0000 mm
Ty	0.0000 mm
Tz	-0.0000 mm
Rx	-0.0000 deg
Ry	-0.0000 deg
Rz	0.0000 deg

Figure 14: Advanced states general screen

The General screen is divided into four sections:

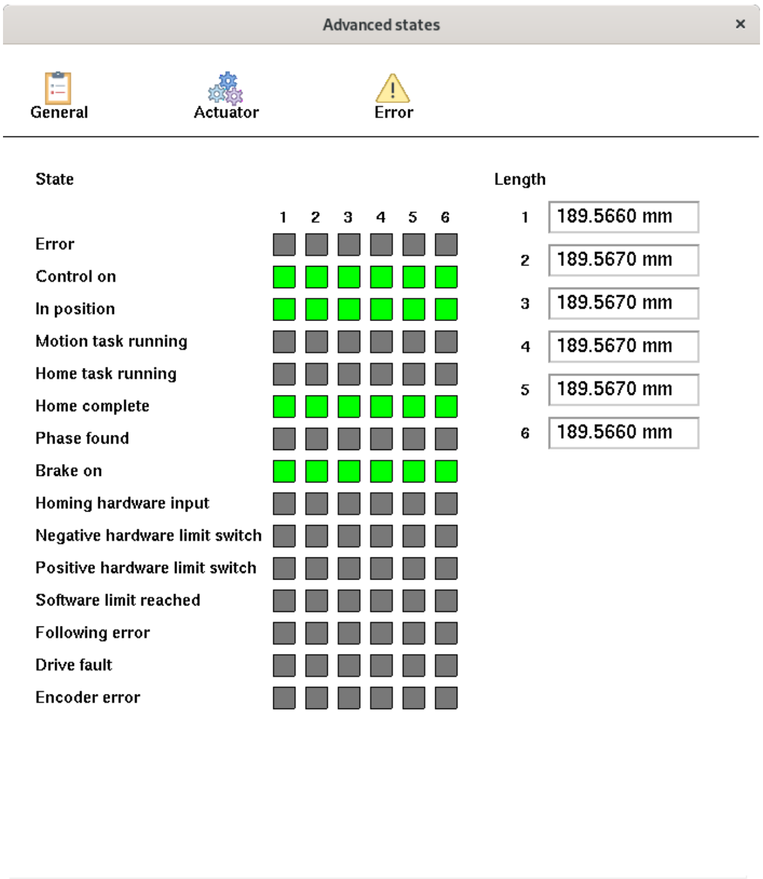
- System - retrieves main controller status
- Action - display the current running action
- Inputs & outputs - retrieves states of the inputs and outputs
- Machine position - hexapod position expressed in Machine coordinate system.

The system status is described fully in the Application programming interface (API) manual. The individual LEDs represent a single bit from the bitmask returned from the s_hexa value. Below the system indicators the widget displays the current action which is executing. The following actions are possible:

- None
- Stop
- ControlOn
- ControlOff
- Home
- HomeVirtual
- MovePTP
- MoveSpecificPos
- MoveSequence
- MoveJog
- Handwheel
- Maintenance

The GPIO displays the current status of the individual GPIO signals. On the right of the screen the current machine position is displayed in the machine coordinate system.

Clicking on the Actuator button at the top of the screen switches to the Actuator screen.



State

	1	2	3	4	5	6
Error	■	■	■	■	■	■
Control on	🟩	🟩	🟩	🟩	🟩	🟩
In position	🟩	🟩	🟩	🟩	🟩	🟩
Motion task running	■	■	■	■	■	■
Home task running	■	■	■	■	■	■
Home complete	🟩	🟩	🟩	🟩	🟩	🟩
Phase found	■	■	■	■	■	■
Brake on	🟩	🟩	🟩	🟩	🟩	🟩
Homing hardware input	■	■	■	■	■	■
Negative hardware limit switch	■	■	■	■	■	■
Positive hardware limit switch	■	■	■	■	■	■
Software limit reached	■	■	■	■	■	■
Following error	■	■	■	■	■	■
Drive fault	■	■	■	■	■	■
Encoder error	■	■	■	■	■	■

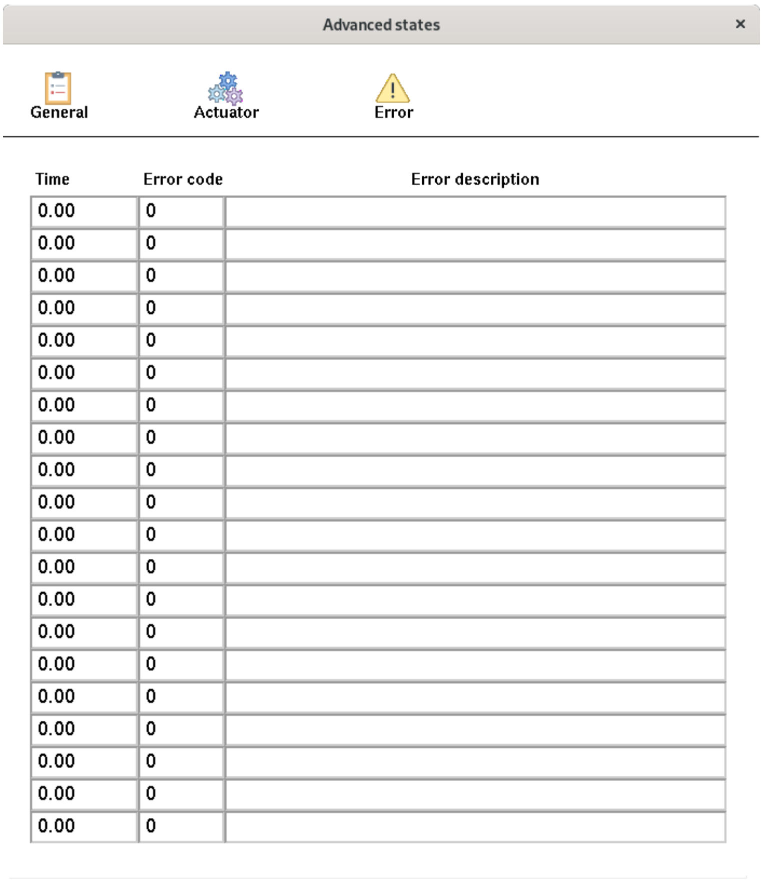
Length

1	189.5660 mm
2	189.5670 mm
3	189.5670 mm
4	189.5670 mm
5	189.5670 mm
6	189.5660 mm

Figure 15: Advanced states actuator screen

The actuator screen contains the full status for each individual axis of the hexapod, along with the raw lengths (in mm) of each axis. The actuator status is described fully in Application programming interface (API) manual. The individual LEDs represent a single bit from the bitmask returned from the specific axis status variable (s_ax_1..6).

Clicking on the Error button at the top of the screen switches to the Error screen.



Time	Error code	Error description
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	
0.00	0	

Figure 16: Advanced states error screen

The error screen displays a table of errors present on the hexapod controller. Each error that occurs is displayed with the time at which the error occurred, a numerical value representing the error code and a description of the error itself. Any additional information required to specify the error conditions is appended to the error message. In the figure above three errors have occurred. Errors can only be cleared by clicking on the "Clear errors" button on the main display.

4.4.4 Parameters screen presentation

Returning to the main screen, clicking on the "Menu" and selecting the "Parameters" option opens the parameters speed/acceleration screen.

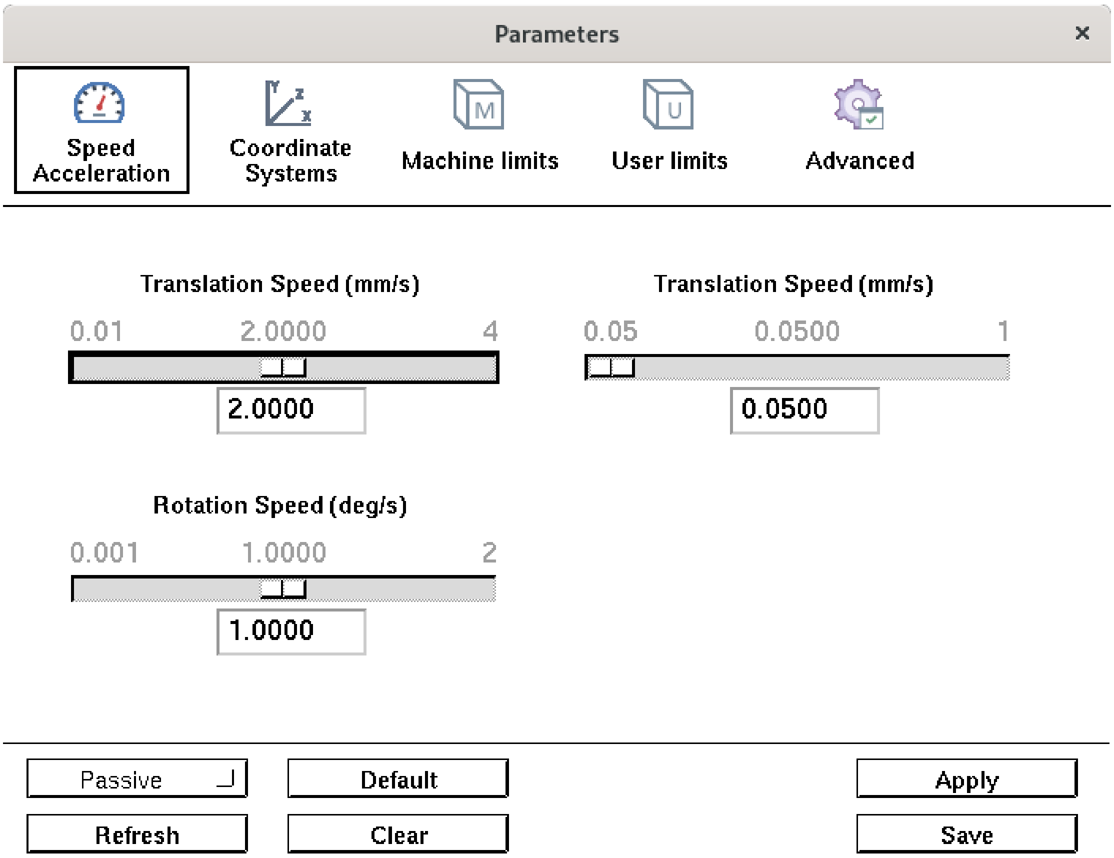


Figure 17: Parameters speed / acceleration screen

All parameter screens contain the same block of widgets at the bottom. Parameter values by default are only read out of the hardware controller when a new value is applied, or when defaults are loaded. A manual read of parameters is possible by clicking on the "Refresh" button. It is also possible to set an automatic refresh rate by change the "Passive" value in the drop-down menu to one of the other scan rates. Whenever a read of parameters is executed, the corresponding EPICS read back records (those whose names contain _RBV) will be updated.

All parameters have demand records and read back records, represented by the two sets of widgets for each parameter seen above. The demand can be changed and when the "Apply" button is clicked the current demands (of all parameters) are written to the control hardware.

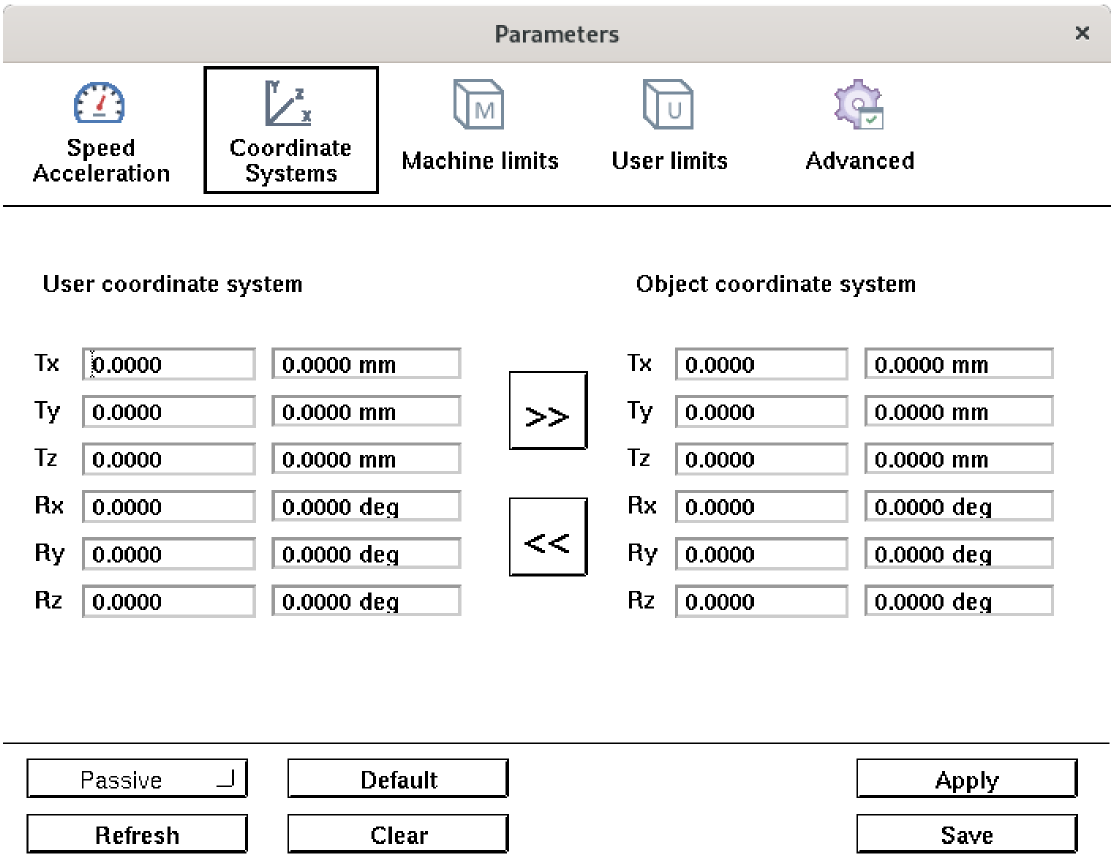
The default values can be re-applied by clicking on the "Default" button. After a default reset, all demand records are updated to match the read back records.

Clicking on the "Clear" button will reset any changed demand values back to the currently read back value.

Clicking on the "Save" button will save the currently applied demands so that they persist after a reboot of the control hardware.

The speed / acceleration screen allows the user to configure the speed and acceleration time used for the positioning of the hexapod. Speeds are specified separately for the translation and rotation virtual axes. New demand values can either be typed into the demand boxes or the sliders can be used. Minimum and maximum values are shown as the limits of the slider widgets.

Clicking on the "Coordinate Systems" button at the top of the screen opens the parameters coordinate systems screen.



User coordinate system

Tx	0.0000	0.0000 mm
Ty	0.0000	0.0000 mm
Tz	0.0000	0.0000 mm
Rx	0.0000	0.0000 deg
Ry	0.0000	0.0000 deg
Rz	0.0000	0.0000 deg

Object coordinate system

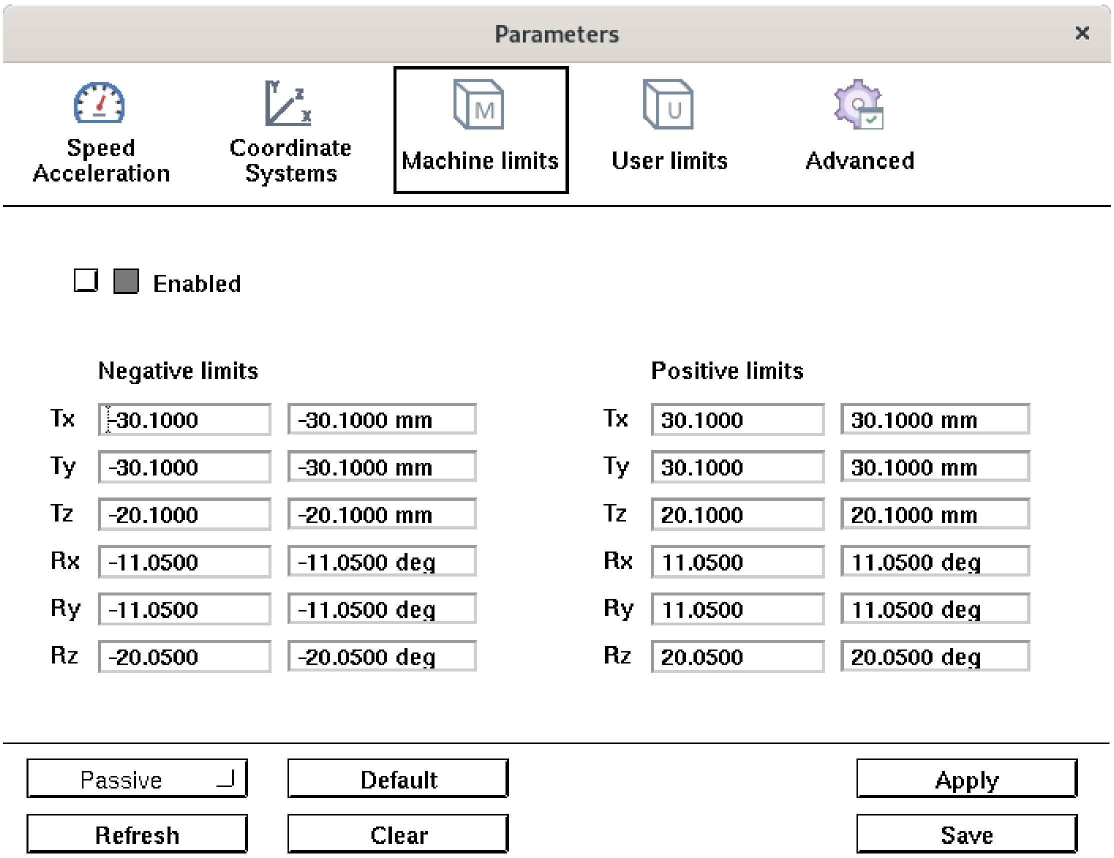
Tx	0.0000	0.0000 mm
Ty	0.0000	0.0000 mm
Tz	0.0000	0.0000 mm
Rx	0.0000	0.0000 deg
Ry	0.0000	0.0000 deg
Rz	0.0000	0.0000 deg

Figure 18: Parameters Coordinate systems screen

The Coordinate Systems parameters are explained in detail in Application programming interface (API) manual. This screen can be used to define both user and object coordinate systems. In most applications the user and object coordinate systems must be identical. The arrowed buttons can be used to copy the demands of one coordinate system into the demands of the other. Note that clicking on the arrowed buttons will not apply the changes, it will simply setup the demands ready to be applied.

Coordinate system parameters cannot be applied if a motion task is running.

Clicking on the "Machine limits" button at the top of the screen opens the parameters Machine limits screen.



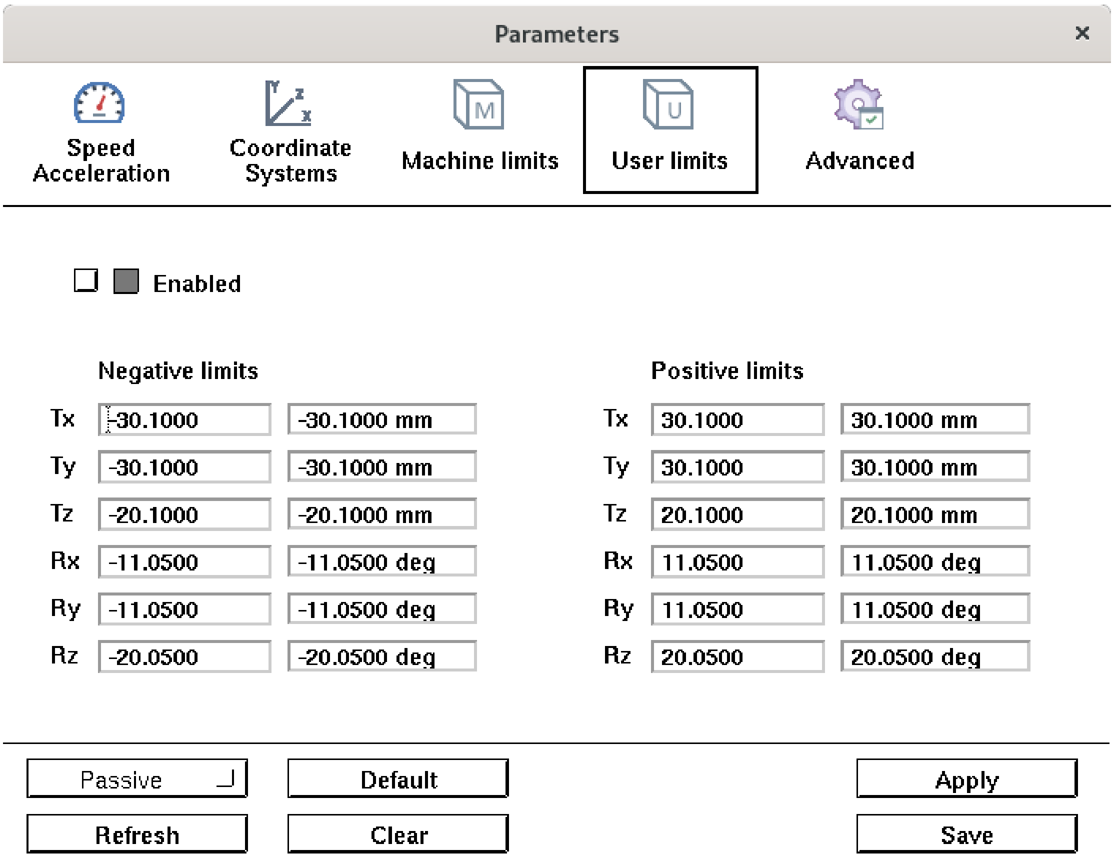
□ ■ Enabled

	Negative limits			Positive limits	
Tx	-30.1000	-30.1000 mm	Tx	30.1000	30.1000 mm
Ty	-30.1000	-30.1000 mm	Ty	30.1000	30.1000 mm
Tz	-20.1000	-20.1000 mm	Tz	20.1000	20.1000 mm
Rx	-11.0500	-11.0500 deg	Rx	11.0500	11.0500 deg
Ry	-11.0500	-11.0500 deg	Ry	11.0500	11.0500 deg
Rz	-20.0500	-20.0500 deg	Rz	20.0500	20.0500 deg

Figure 19: Parameters Machine limits screen

The Machine limits screen provides widgets for setting the negative and positive Machine limits, as well as enabling or disabling the limit enforcement.

Clicking on the "User limits" button at the top of the screen opens the parameters User limits screen.



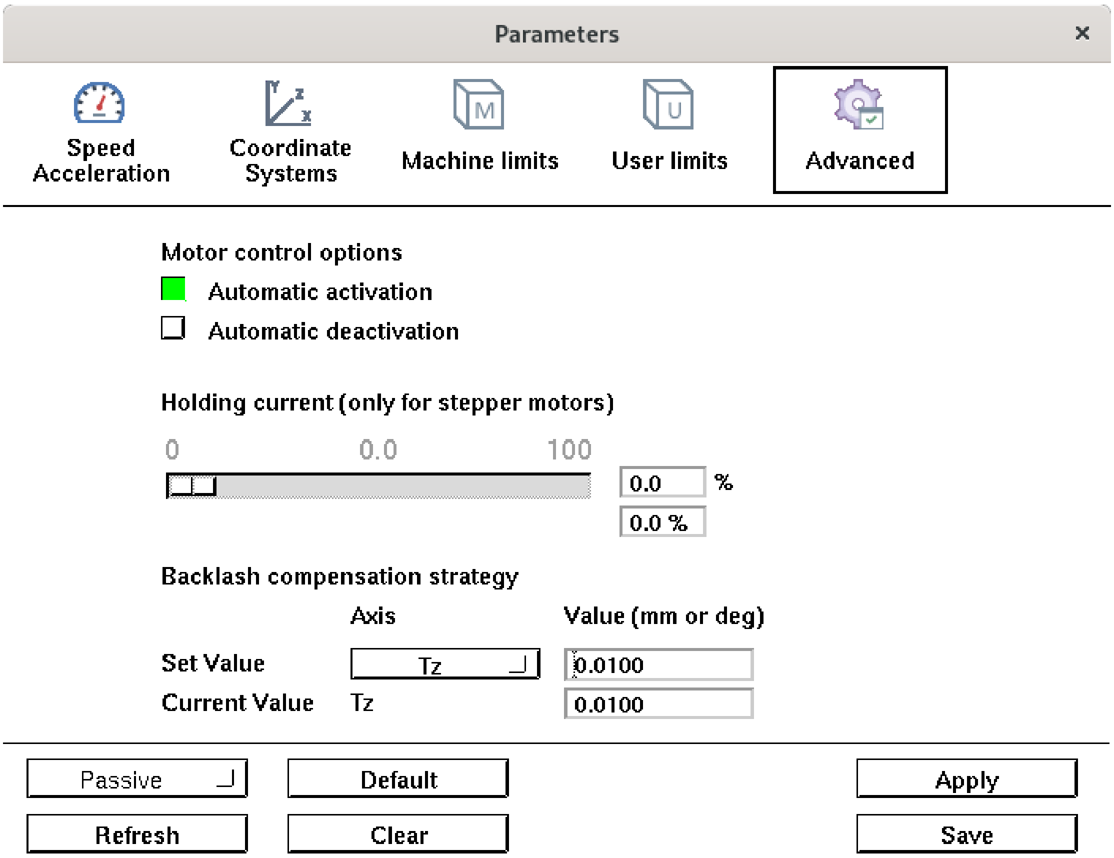
☐ ■ Enabled

	Negative limits			Positive limits	
Tx	-30.1000	-30.1000 mm	Tx	30.1000	30.1000 mm
Ty	-30.1000	-30.1000 mm	Ty	30.1000	30.1000 mm
Tz	-20.1000	-20.1000 mm	Tz	20.1000	20.1000 mm
Rx	-11.0500	-11.0500 deg	Rx	11.0500	11.0500 deg
Ry	-11.0500	-11.0500 deg	Ry	11.0500	11.0500 deg
Rz	-20.0500	-20.0500 deg	Rz	20.0500	20.0500 deg

Figure 20: Parameters User limits screen

The User limits screen provides widgets for setting the negative and positive User limits, as well as enabling or disabling the limit enforcement.

Clicking on the "Advanced" button at the top of the screen opens the parameters advanced screen.



Motor control options

🟩 Automatic activation

□ Automatic deactivation

Holding current (only for stepper motors)

0 0.0 100

0.0 %

0.0 %

Backlash compensation strategy

	Axis	Value (mm or deg)
Set Value	Tz	0.0100
Current Value	Tz	0.0100

Figure 21: Parameter advanced screen

The advanced screen provides widgets to set the motor control parameters. Automatic activation will activate the motor control loop when a motion task is executed. If this is set then it is not necessary to turn control on from the main screen. Automatic deactivation will automatically disable the motor control loop whenever the hexapod has reached the requested position. The holding current for stepper motors can be set, it is specified as a percentage (between 0 and 100%) of the nominal motor current. The backlash compensation strategy can be selected, with all positioning tasks always finishing with the same approach movement. A detailed explanation of these parameters can be found in Application programming interface (API) manual.

5 EPICS RECORDS

5.1 PRESENTATION

The table below presents the records used to control a SYMETRIE hexapod using the support module. The record name, corresponding SYMETRIE command/status item and a supporting description are shown. This table can be used to integrate the support module into existing EPICS IOC applications. Every record in the template files have the prefix (P)(R) to provide macro substitution for specific applications.

Detailed explanations of the SYMETRIE commands can be found in Application programming interface (API) manual and so are not discussed here.

The status records (s_xxx) are updated whenever the STATUS_GET record is processed. That record has a scan rate of 0.5 seconds (2.0 Hz) as a default setting and can be configured using any channel access client by updating the value of the scan field "STATUS_GET.SCAN".

The configuration RBV (Read Back Value) records (CFG_xxx_RBV) are updated by processing the CFG_GET record. That record can also be processed by other template records as a result of performing specific actions (for example applying a new configuration). It has a "Passive" scan rate by default which can be configured using any channel access client by updating the value of the scan field "CFG_GET.SCAN".

Possible values for SCAN are the following strings: "Passive", "Event", "I/O Intr", "10 second", "5 second", "2 second", "1 second", ".5 second", ".2 second", ".1 second".

The configuration set commands can by executed command by command, by processing the CFG:xxx:Set record of the command. It is also possible to execute all configuration set commands at once, using the CFG_SET record.

When the configuration commands have parameters, the parameters records must be set before the execution record ("CFG:xxx:Set" or "CFG_SET").

When the action commands have parameters, the parameters records must be set before the execution record. For the MOVE_PTP command for example, the parameters record "MOVE_PTP:MoveType", "MOVE_PTP:Tx", …, "MOVE_PTP:Rz" must be modified before the use of the "MOVE_PTP" execution record.

The response value of the latest processed command is stored in the records "Drv:CmdReturnCode_RBV" and Drv:CmdReturnDescription_RBV".

Record (all prefixed with (P)(R))	SYMETRIE API	Description
VERSION:ControllerSoftwareID_RBV	C_VERSION	These records hold the current versions of firmware and hardware. They are all read out once from the hexapod controller at driver start-up, directly after the connection to the controller
VERSION:SystemID_RBV	C_VERSION	
VERSION:SystemNum_RBV	C_VERSION	
VERSION:SystemCfgVersion_RBV	C_VERSION	
VERSION:ControllerSoftwareVersion_RBV	C_VERSION	
VERSION:APISoftwareVersion_RBV	C_VERSION	
s_hexa_RBV	s_hexa	General state of the hexapod. The bitmask is stored in the main record, individual bits are represented by each of the other records.
s_hexa:Error_RBV	s_hexa	
s_hexa:Initialised_RBV	s_hexa	
s_hexa:ControlOn_RBV	s_hexa	
s_hexa:InPosition_RBV	s_hexa	
s_hexa:MotionTaskRunning_RBV	s_hexa	
s_hexa:HomeTaskRunning_RBV	s_hexa	
s_hexa:HomeComplete_RBV	s_hexa	
s_hexa:HomeVirtual_RBV	s_hexa	
s_hexa:PhaseFound_RBV	s_hexa	
s_hexa:BrakeOn_RBV	s_hexa	
s_hexa:MotionRestricted_RBV	s_hexa	
s_hexa:PowerOnEncoders_RBV	s_hexa	
s_hexa:PowerOnLimits_RBV	s_hexa	
s_hexa:PowerOnDrives_RBV	s_hexa	
s_hexa:EmergencyStop_RBV	s_hexa	
s_action_RBV	s_action	Currently executing task.
s_uto_tx_RBV	s_uto	Current hexapod position in the user coordinate system.
s_uto_ty_RBV	s_uto	
s_uto_tz_RBV	s_uto	
s_uto_rx_RBV	s_uto	
s_uto_ry_RBV	s_uto	
s_uto_rz_RBV	s_uto	
s_mtp_tx_RBV	s_mtp	Current hexapod position in the machine coordinate system.
s_mtp_ty_RBV	s_mtp	
s_mtp_tz_RBV	s_mtp	
s_mtp_rx_RBV	s_mtp	
s_mtp_ry_RBV	s_mtp	
s_mtp_rz_RBV	s_mtp	
s_ax_<n>_RBV	s_ax	(n = 1,2,3,4,5,6)
s_ax_<n>:Error_RBV	s_ax	

Record (all prefixed with (P)(R))	SYMETRIE API	Description
s_ax_<n>:ControlOn_RBV	s_ax	General state of the actuators. The bitmask is stored in the main record, individual bits are represented by each of the other records.
s_ax_<n>:InPosition_RBV	s_ax	
s_ax_<n>:MotionTaskRunning_RBV	s_ax	
s_ax_<n>:HomeTaskRunning_RBV	s_ax	
s_ax_<n>:HomeComplete_RBV	s_ax	
s_ax_<n>:PhaseFound_RBV	s_ax	
s_ax_<n>:BrakeOn_RBV	s_ax	
s_ax_<n>:HomeHardwareInput_RBV	s_ax	
s_ax_<n>:NegativeLimitSwitch_RBV	s_ax	
s_ax_<n>:PositiveLimitSwitch_RBV	s_ax	
s_ax_<n>:SoftwareLimit_RBV	s_ax	
s_ax_<n>:FollowingError_RBV	s_ax	
s_ax_<n>:DriveFault_RBV	s_ax	
s_ax_<n>:EncoderError_RBV	s_ax	
s_pos_ax_<n>_RBV	s_ax	(n = 1,2,3,4,5,6) Actuator positions (in mm).
s_dio_RBV	s_dio_1	Status of the digital IO. The bitmask is stored in the main record and individual bits are represented by each of the other records.
s_dio:GPIO1:Input1_RBV	s_dio_1	
s_dio:GPIO1:Output1_RBV	s_dio_1	
s_dio:GPIO1:Output2_RBV	s_dio_1	
s_dio:GPIO2:Input1_RBV	s_dio_1	
s_dio:GPIO2:Input2_RBV	s_dio_1	
s_dio:GPIO2:Input3_RBV	s_dio_1	
s_dio:GPIO2:Input4_RBV	s_dio_1	
s_dio:GPIO2:Input5_RBV	s_dio_1	
s_dio:GPIO2:Input6_RBV	s_dio_1	
s_dio:GPIO2:Input7_RBV	s_dio_1	
s_dio:GPIO2:Input8_RBV	s_dio_1	
s_ai_1_RBV	s_ai	Analogue input values.
s_ai_2_RBV	s_ai	
s_ai_3_RBV	s_ai	
s_ai_4_RBV	s_ai	
s_ai_5_RBV	s_ai	
s_ai_6_RBV	s_ai	
s_ai_7_RBV	s_ai	
s_ai_8_RBV	s_ai	
s_cycle_RBV	s_cycle	Cycle of the currently running sequence.

Record (all prefixed with (P)(R))	SYMETRIE API	Description
s_index_RBV	s_index	Index of the currently running sequence.
s_err_nr_RBV	s_err_nr	Number of errors currently present on the hexapod controller.
ERR_INFO:<n>:Index	C_ERR_INFO	(n = 0..20) These records hold the information relating to any errors currently present on the hexapod controller. Each time a new error occurs the next set of records will be populated. Clearing errors will result in the records also being cleared. The update rate is linked to the status scan rate (STATUS_GET.SCAN). The Drv: <n>:ErrorDescription_RBV is a description string. The description string table is part of the driver. The "Code" record is used to index this description string table.
ERR_INFO: <n>:Code	C_ERR_INFO	
ERR_INFO: <n>:Group	C_ERR_INFO	
ERR_INFO: <n>:Axis	C_ERR_INFO	
ERR_INFO: <n>:Return	C_ERR_INFO	
ERR_INFO: <n>:Data	C_ERR_INFO	
ERR_INFO: <n>:Data2	C_ERR_INFO	
ERR_INFO: <n>:Time	C_ERR_INFO	
Drv: <n>:ErrorDescription_RBV		
STOP	C_STOP	Stop the current movement.
CONTROLON	C_CONTROLON	Activate the control loop on motors.
CONTROLOFF	C_CONTROLOFF	Disable the control loop on motors.
HOME	C_HOME	Start the home task on the hexapod controller.
HOMEVIRTUAL	C_HOMEVIRTUAL	Execute a virtual home.
CLEARERROR	C_CLEARERROR	Clear any errors currently present on the hexapod controller.
MOVE_PTP:MoveType	C_MOVE_PTP	Move the hexapod to the specified demand positions according to the specified move type.
MOVE_PTP:Tx	C_MOVE_PTP	
MOVE_PTP:Ty	C_MOVE_PTP	
MOVE_PTP:Tz	C_MOVE_PTP	
MOVE_PTP:Rx	C_MOVE_PTP	
MOVE_PTP:Ry	C_MOVE_PTP	
MOVE_PTP:Rz	C_MOVE_PTP	
MOVE_PTP	C_MOVE_PTP	
MOVE_SPECIFICPOS:Index	C_MOVESPECIFICPOS	Move the hexapod to a predefined specific position.
MOVE_SPECIFICPOS	C_MOVESPECIFICPOS	
MOVE_SEQ	C_MOVE_SEQ	Start executing the last downloaded sequence.
VALID_PTP:ValidationMode	C_VALID_PTP	Validate the positions. The validation result is contained in the Result_RBV record. See also the Drv:ValidateMove records
VALID_PTP:MoveType	C_VALID_PTP	
VALID_PTP:Tx	C_VALID_PTP	
VALID_PTP:Ty	C_VALID_PTP	
VALID_PTP:Tz	C_VALID_PTP	
VALID_PTP:Rx	C_VALID_PTP	
VALID_PTP:Ry	C_VALID_PTP	
VALID_PTP:Rz	C_VALID_PTP	

Record (all prefixed with (P)(R))	SYMETRIE API	Description
VALID_PTP	C_VALID_PTP	
VALID_PTP:Result_RBV	C_VALID_PTP	
POWER:Power	C_POWER	Turn on or off power to the encoders and limit switches.
POWER	C_POWER	
MAINTENANCE:Mode	C_MAINTENANCE	Start execution of a maintenance cycle.
MAINTENANCE:Axis	C_MAINTENANCE	
MAINTENANCE	C_MAINTENANCE	
AXIS_JOG:Increment	C_AXIS_JOG	Start a jog type movement on the selected actuator.
AXIS_JOG:Axis	C_AXIS_JOG	
AXIS_JOG	C_AXIS_JOG	
REBOOT	system reboot	Reboot the controller. NOTE: This will result in the EPICS application losing communication with the hexapod controller.
CFG_DEFAULT	C_CFG_DEFAULT	Restore the default configuration parameters.
CFG_SAVE	C_CFG_SAVE	Save the configuration parameters currently applied to the hexapod controller.
CFG_SAVE:saved_RBV	C_CFG_SAVE	
CFG_SAFETYINPUT:Value	C_CFG_SAFETYINPUT	Enable safety inputs on the hexapod controller.
CFG_SAFETYINPUT:Set	C_CFG_SAFETYINPUT	
CFG_SAFETYINPUT:Value_RBV	C_CFG_SAFETYINPUT	
CFG_CHANNEL:Actuator1	C_CFG_CHANNEL	Configure the channels used to control the hexapod.
CFG_CHANNEL:Actuator2	C_CFG_CHANNEL	
CFG_CHANNEL:Actuator3	C_CFG_CHANNEL	
CFG_CHANNEL:Actuator4	C_CFG_CHANNEL	
CFG_CHANNEL:Actuator5	C_CFG_CHANNEL	
CFG_CHANNEL:Actuator6	C_CFG_CHANNEL	
CFG_CHANNEL:Set	C_CFG_CHANNEL	
CFG_CHANNEL:Actuator1_RBV	C_CFG_CHANNEL	
CFG_CHANNEL:Actuator2_RBV	C_CFG_CHANNEL	
CFG_CHANNEL:Actuator3_RBV	C_CFG_CHANNEL	
CFG_CHANNEL:Actuator4_RBV	C_CFG_CHANNEL	
CFG_CHANNEL:Actuator5_RBV	C_CFG_CHANNEL	
CFG_CHANNEL:Actuator6_RBV	C_CFG_CHANNEL	
CFG_SPEED:Trans	C_CFG_SPEED	Set the translational and rotational speed of the hexapod moves. Minimum and maximum values are read from the controller and these cannot be set.
CFG_SPEED:Rot	C_CFG_SPEED	
CFG_SPEED:Set	C_CFG_SPEED	
CFG_SPEED:Trans_RBV	C_CFG_SPEED	
CFG_SPEED:Trans_Min_RBV	C_CFG_SPEED	
CFG_SPEED:Trans_Max_RBV	C_CFG_SPEED	

Record (all prefixed with (P)(R))	SYMETRIE API	Description
CFG_SPEED:Rot_RBV	C_CFG_SPEED	
CFG_SPEED:Rot_Min_RBV	C_CFG_SPEED	
CFG_SPEED:Rot_Max_RBV	C_CFG_SPEED	
CFG_TA:Value	C_CFG_TA	Set the acceleration time of movements. Minimum and maximum values are read from the controller and these cannot be set.
CFG_TA:Set	C_CFG_TA	
CFG_TA:Value_RBV	C_CFG_TA	
CFG_TA:Min_RBV	C_CFG_TA	
CFG_TA:Max_RBV	C_CFG_TA	
CFG_CS:Txu	C_CFG_CS	Set the definition of the coordinate systems (user and object). See also the Drv:Copy_UserCSToObjectCS and Drv:Copy_ObjectCSToUserCS records.
CFG_CS:Tyu	C_CFG_CS	
CFG_CS:Tzu	C_CFG_CS	
CFG_CS:Rxu	C_CFG_CS	
CFG_CS:Ryu	C_CFG_CS	
CFG_CS:Rzu	C_CFG_CS	
CFG_CS:Txo	C_CFG_CS	
CFG_CS:Tyo	C_CFG_CS	
CFG_CS:Tzo	C_CFG_CS	
CFG_CS:Rxo	C_CFG_CS	
CFG_CS:Ryo	C_CFG_CS	
CFG_CS:Rzo	C_CFG_CS	
CFG_CS:Set	C_CFG_CS	
CFG_CS:Txu_RBV	C_CFG_CS	
CFG_CS:Tyu_RBV	C_CFG_CS	
CFG_CS:Tzu_RBV	C_CFG_CS	
CFG_CS:Rxu_RBV	C_CFG_CS	
CFG_CS:Ryu_RBV	C_CFG_CS	
CFG_CS:Rzu_RBV	C_CFG_CS	
CFG_CS:Txo_RBV	C_CFG_CS	
CFG_CS:Tyo_RBV	C_CFG_CS	
CFG_CS:Tzo_RBV	C_CFG_CS	
CFG_CS:Rxo_RBV	C_CFG_CS	
CFG_CS:Ryo_RBV	C_CFG_CS	
CFG_CS:Rzo_RBV	C_CFG_CS	
CFG_LIMIT:Machine_Tx_Neg	C_CFG_LIMIT	Set the Machine limits of the hexapod.
CFG_LIMIT:Machine_Ty_Neg	C_CFG_LIMIT	
CFG_LIMIT:Machine_Tz_Neg	C_CFG_LIMIT	
CFG_LIMIT:Machine_Rx_Neg	C_CFG_LIMIT	

Record (all prefixed with (P)(R))	SYMETRIE API	Description
CFG_LIMIT:Machine_Ry_Neg	C_CFG_LIMIT	
CFG_LIMIT:Machine_Rz_Neg	C_CFG_LIMIT	
CFG_LIMIT:Machine_Tx_Pos	C_CFG_LIMIT	
CFG_LIMIT:Machine_Ty_Pos	C_CFG_LIMIT	
CFG_LIMIT:Machine_Tz_Pos	C_CFG_LIMIT	
CFG_LIMIT:Machine_Rx_Pos	C_CFG_LIMIT	
CFG_LIMIT:Machine_Ry_Pos	C_CFG_LIMIT	
CFG_LIMIT:Machine_Rz_Pos	C_CFG_LIMIT	
CFG_LIMIT:Machine:Set	C_CFG_LIMIT	
CFG_LIMIT:Machine_Tx_Neg_RBV	C_CFG_LIMIT	
CFG_LIMIT:Machine_Ty_Neg_RBV	C_CFG_LIMIT	
CFG_LIMIT:Machine_Tz_Neg_RBV	C_CFG_LIMIT	
CFG_LIMIT:Machine_Rx_Neg_RBV	C_CFG_LIMIT	
CFG_LIMIT:Machine_Ry_Neg_RBV	C_CFG_LIMIT	
CFG_LIMIT:Machine_Rz_Neg_RBV	C_CFG_LIMIT	
CFG_LIMIT:Machine_Tx_Pos_RBV	C_CFG_LIMIT	
CFG_LIMIT:Machine_Ty_Pos_RBV	C_CFG_LIMIT	
CFG_LIMIT:Machine_Tz_Pos_RBV	C_CFG_LIMIT	
CFG_LIMIT:Machine_Rx_Pos_RBV	C_CFG_LIMIT	
CFG_LIMIT:Machine_Ry_Pos_RBV	C_CFG_LIMIT	
CFG_LIMIT:Machine_Rz_Pos_RBV	C_CFG_LIMIT	
CFG_LIMIT:User_Tx_Neg	C_CFG_LIMIT	Set the User limits of the hexapod.
CFG_LIMIT:User_Ty_Neg	C_CFG_LIMIT	
CFG_LIMIT:User_Tz_Neg	C_CFG_LIMIT	
CFG_LIMIT:User_Rx_Neg	C_CFG_LIMIT	
CFG_LIMIT:User_Ry_Neg	C_CFG_LIMIT	
CFG_LIMIT:User_Rz_Neg	C_CFG_LIMIT	
CFG_LIMIT:User_Tx_Pos	C_CFG_LIMIT	
CFG_LIMIT:User_Ty_Pos	C_CFG_LIMIT	
CFG_LIMIT:User_Tz_Pos	C_CFG_LIMIT	
CFG_LIMIT:User_Rx_Pos	C_CFG_LIMIT	
CFG_LIMIT:User_Ry_Pos	C_CFG_LIMIT	
CFG_LIMIT:User_Rz_Pos	C_CFG_LIMIT	
CFG_LIMIT:User:Set	C_CFG_LIMIT	
CFG_LIMIT:User_Tx_Neg_RBV	C_CFG_LIMIT	
CFG_LIMIT:User_Ty_Neg_RBV	C_CFG_LIMIT	
CFG_LIMIT:User_Tz_Neg_RBV	C_CFG_LIMIT	

Record (all prefixed with (P)(R))	SYMETRIE API	Description
CFG_LIMIT:User_Rx_Neg_RBV	C_CFG_LIMIT	
CFG_LIMIT:User_Ry_Neg_RBV	C_CFG_LIMIT	
CFG_LIMIT:User_Rz_Neg_RBV	C_CFG_LIMIT	
CFG_LIMIT:User_Tx_Pos_RBV	C_CFG_LIMIT	
CFG_LIMIT:User_Ty_Pos_RBV	C_CFG_LIMIT	
CFG_LIMIT:User_Tz_Pos_RBV	C_CFG_LIMIT	
CFG_LIMIT:User_Rx_Pos_RBV	C_CFG_LIMIT	
CFG_LIMIT:User_Ry_Pos_RBV	C_CFG_LIMIT	
CFG_LIMIT:User_Rz_Pos_RBV	C_CFG_LIMIT	
CFG_LIMITENABLE:Machine	C_CFG_LIMITENABLE	Enable or disable the machine and user limits.
CFG_LIMITENABLE:Machine:Set	C_CFG_LIMITENABLE	
CFG_LIMITENABLE:Machine_RBV	C_CFG_LIMITENABLE	
CFG_LIMITENABLE:User	C_CFG_LIMITENABLE	
CFG_LIMITENABLE:User:Set	C_CFG_LIMITENABLE	
CFG_LIMITENABLE:User_RBV	C_CFG_LIMITENABLE	
CFG_CONTROL:AutomatedControlOn	C_CFG_CONTROL	Turn on or off the automatic enabling and disabling of the motor control loop when moving the hexapod.
CFG_CONTROL:AutomatedControlOff	C_CFG_CONTROL	
CFG_CONTROL:Delay	C_CFG_CONTROL	
CFG_CONTROL:Set	C_CFG_CONTROL	
CFG_CONTROL:AutomatedControlOn_RBV	C_CFG_CONTROL	
CFG_CONTROL:AutomatedControlOff_RBV	C_CFG_CONTROL	
CFG_CONTROL:Delay_RBV	C_CFG_CONTROL	
CFG_STALLCURRENT:Value	C_CFG_STALLCURRENT	Set the holding current for stepper motors as a percentage of nominal current.
CFG_STALLCURRENT:Set	C_CFG_STALLCURRENT	
CFG_STALLCURRENT:Value_RBV	C_CFG_STALLCURRENT	
CFG_BACKLASH:Axis	C_CFG_BACKLASH	Set the backlash compensation movement (axis and size).
CFG_BACKLASH:Value	C_CFG_BACKLASH	
CFG_BACKLASH:Set	C_CFG_BACKLASH	
CFG_BACKLASH:Axis_RBV	C_CFG_BACKLASH	
CFG_BACKLASH:Value_RBV	C_CFG_BACKLASH	
CFG_HOME:Auto_RBV	C_CFG_HOME	Read only. Gets the home task options for the hexapod controller. Auto if the controller auto homes on startup. Virtual if the home virtual feature is enabled. Type of home (absolute, relative).
CFG_HOME:Virtual_RBV	C_CFG_HOME	
CFG_HOME:Type_RBV	C_CFG_HOME	
CFG_KIN:Mode	C_CFG_KIN	Set the kinematic mode of the hexapod.
CFG_KIN:Set	C_CFG_KIN	
CFG_KIN:Mode_RBV	C_CFG_KIN	
CFG_KIN:OperationalAxisNum_RBV	C_CFG_KIN	

Record (all prefixed with (P)(R))	SYMETRIE API	Description
CFG_TUNING:Index	C_CFG_TUNING	Set the tuning selection for the hexapod.
CFG_TUNING:Set	C_CFG_TUNING	
CFG_TUNING:Index_RBV	C_CFG_TUNING	
CFG_POWER:Enable	C_CFG_POWER	Set the power dissipation reduction strategy parameters.
CFG_POWER:Auto	C_CFG_POWER	
CFG_POWER:Set	C_CFG_POWER	
CFG_POWER:Enable_RBV	C_CFG_POWER	
CFG_POWER:Auto_RBV	C_CFG_POWER	
STATUS_GET		When this record is processed all status is read from the hexapod controller.
CFG_GET		When this record is processed all configuration parameters are read from the hexapod controller.
CFG_SET		When this record is processed all configuration parameter demands are applied to the hexapod controller.
TERMINAL:String		Writing a string into the String record will result in an immediate write into the hexapod controller. Any response from the controller will be written into the RBV record.
TERMINAL:String_RBV		
Drv:CmdReturnCode_RBV		Whenever a command is executed the response value is written into the ReturnCode record. If the code is negative then the command has failed and the reason is written into the Description record.
Drv:CmdReturnDescription_RBV		
Drv:SetLimit_Speed_Trans_Min		These records monitor the limits read out from the hexapod controller hardware and apply them to the limit fields of the demand records, providing EPICS limits at the record level.
Drv:SetLimit_Speed_Trans_Max		
Drv:SetLimit_Speed_Rot_Min		
Drv:SetLimit_Speed_Rot_Max		
Drv:SetLimit_Speed_TA_Min		
Drv:SetLimit_Speed_TA_Max		
Drv:ValidateMove:CopyAndExecute		These records monitor changes to the move demands and immediately validate them by executing the VALID_PTP command. This provides an immediate feedback to any moves before the move is executed.
Drv:ValidateMove:DetectAndExecute		
Drv:ValidateMove:Valid_RBV		
Drv:Copy_UserCSToObjectCS		These records copy the RBV values from one CS into the demand records of the other CS.
Drv:Copy_ObjectCSToUserCS		
Drv:Copy_CfgStatusToCfgCmd		These records copy all configuration parameters read from the hexapod controller (the _RBV) into the EPICS demand records to provide consistent values. The main Copy_CfgStatusToCfgCmd record call the _Sub01, which call the _Sub02, and so on. The main record is processed at driver start-up, at the end of the CFG_SET record process.
Drv:Copy_CfgStatusToCfgCmd_Sub01		
Drv:Copy_CfgStatusToCfgCmd_Sub02		
Drv:Copy_CfgStatusToCfgCmd_Sub03		
Drv:Copy_CfgStatusToCfgCmd_Sub04		
Drv:Copy_CfgStatusToCfgCmd_Sub05		
Drv:Copy_CfgStatusToCfgCmd_Sub06		
Drv:Copy_CfgStatusToCfgCmd_Sub07		

Record (all prefixed with (P)(R))	SYMETRIE API	Description
Drv:Copy_CfgStatusToCfgCmd_Sub08		On the PyDM parameters screen, it is processed when the "Clear" button is pressed.
Drv:Copy_CfgStatusToCfgCmd_Sub09		
Drv:Copy_CfgStatusToCfgCmd_Sub10		

5.2 BASIC EXAMPLE

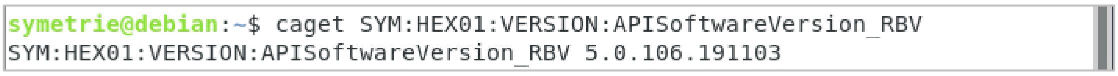
This chapter presents one way of reading and writing records in the terminal. The “caget” terminal command is used get a record value, the “caput” terminal command is used to set a record value.

Read the controller API software version:

Command:

```
caget SYM:HEX01:VERSION:APISoftwareVersion_RBV
```

Command and reply in a terminal:



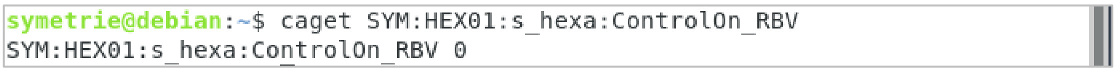
```
symetrie@debian:~$ caget SYM:HEX01:VERSION:APISoftwareVersion_RBV
SYM:HEX01:VERSION:APISoftwareVersion_RBV 5.0.106.191103
```

Read the s_hexa:ControlOn_RBV status:

Command:

```
caget SYM:HEX01:s_hexa:ControlOn_RBV
```

Command and reply in a terminal:



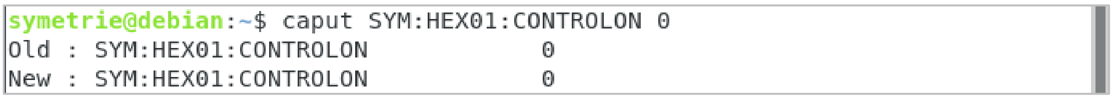
```
symetrie@debian:~$ caget SYM:HEX01:s_hexa:ControlOn_RBV
SYM:HEX01:s_hexa:ControlOn_RBV 0
```

Execute the CONTROLON command:

Command:

```
caput SYM:HEX01:CONTROLON 0
```

Command and reply in a terminal:



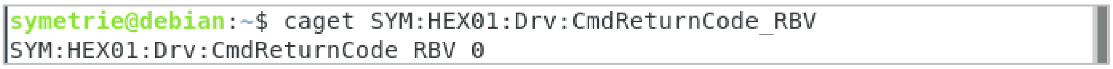
```
symetrie@debian:~$ caput SYM:HEX01:CONTROLON 0
Old : SYM:HEX01:CONTROLON           0
New : SYM:HEX01:CONTROLON           0
```

Check command return after execution:

Command:

```
caget SYM:HEX01:Drv:CmdReturnCode_RBV
```

Command and reply in a terminal:



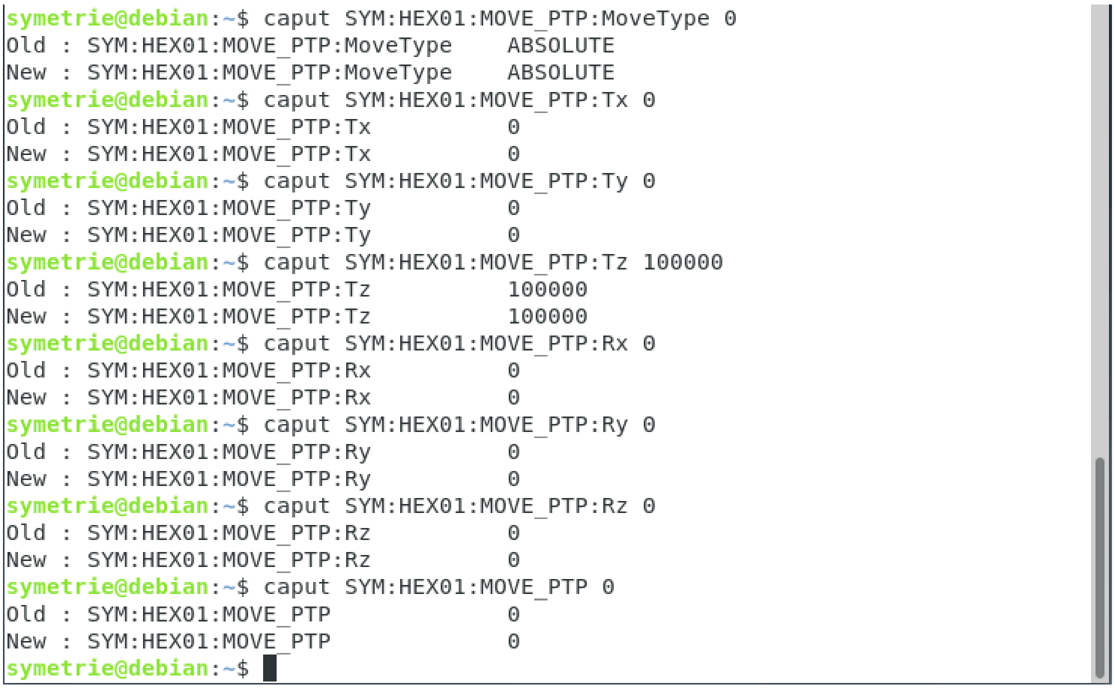
```
symetrie@debian:~$ caget SYM:HEX01:Drv:CmdReturnCode_RBV
SYM:HEX01:Drv:CmdReturnCode_RBV 0
```

Execute the MOVE_PTP command, on purpose a command which is not feasible:

Command:

```
caput SYM:HEX01:MOVE_PTP:MoveType 0
caput SYM:HEX01:MOVE_PTP:Tx 0
caput SYM:HEX01:MOVE_PTP:Ty 0
caput SYM:HEX01:MOVE_PTP:Tz 100000
caput SYM:HEX01:MOVE_PTP:Rx 0
caput SYM:HEX01:MOVE_PTP:Ry 0
caput SYM:HEX01:MOVE_PTP:Rz 0
caput SYM:HEX01:MOVE_PTP 0
```

Command and reply in a terminal:



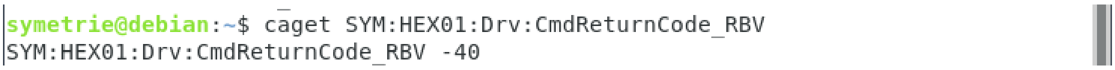
```
symetrie@debian:~$ caput SYM:HEX01:MOVE_PTP:MoveType 0
Old : SYM:HEX01:MOVE_PTP:MoveType    ABSOLUTE
New : SYM:HEX01:MOVE_PTP:MoveType    ABSOLUTE
symetrie@debian:~$ caput SYM:HEX01:MOVE_PTP:Tx 0
Old : SYM:HEX01:MOVE_PTP:Tx          0
New : SYM:HEX01:MOVE_PTP:Tx          0
symetrie@debian:~$ caput SYM:HEX01:MOVE_PTP:Ty 0
Old : SYM:HEX01:MOVE_PTP:Ty          0
New : SYM:HEX01:MOVE_PTP:Ty          0
symetrie@debian:~$ caput SYM:HEX01:MOVE_PTP:Tz 100000
Old : SYM:HEX01:MOVE_PTP:Tz          100000
New : SYM:HEX01:MOVE_PTP:Tz          100000
symetrie@debian:~$ caput SYM:HEX01:MOVE_PTP:Rx 0
Old : SYM:HEX01:MOVE_PTP:Rx          0
New : SYM:HEX01:MOVE_PTP:Rx          0
symetrie@debian:~$ caput SYM:HEX01:MOVE_PTP:Ry 0
Old : SYM:HEX01:MOVE_PTP:Ry          0
New : SYM:HEX01:MOVE_PTP:Ry          0
symetrie@debian:~$ caput SYM:HEX01:MOVE_PTP:Rz 0
Old : SYM:HEX01:MOVE_PTP:Rz          0
New : SYM:HEX01:MOVE_PTP:Rz          0
symetrie@debian:~$ caput SYM:HEX01:MOVE_PTP 0
Old : SYM:HEX01:MOVE_PTP             0
New : SYM:HEX01:MOVE_PTP             0
symetrie@debian:~$ 
```

Check command return after execution:

Command:

```
caget SYM:HEX01:Drv:CmdReturnCode_RBV
```

Command and reply in a terminal:



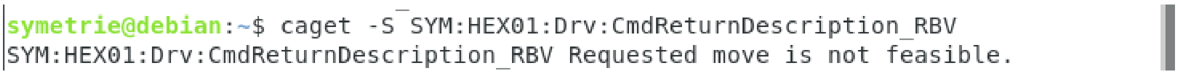
```
symetrie@debian:~$ caget SYM:HEX01:Drv:CmdReturnCode_RBV
SYM:HEX01:Drv:CmdReturnCode_RBV -40
```

Check command return description:

Command:

```
caget -S SYM:HEX01:Drv:CmdReturnDescription_RBV
```

Command and reply in a terminal:



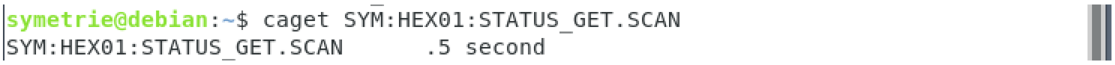
```
symetrie@debian:~$ caget -S SYM:HEX01:Drv:CmdReturnDescription_RBV
SYM:HEX01:Drv:CmdReturnDescription_RBV Requested move is not feasible.
```

Check status scan rate:

Command:

```
caget SYM:HEX01:STATUS_GET.SCAN
```

Command and reply in a terminal:



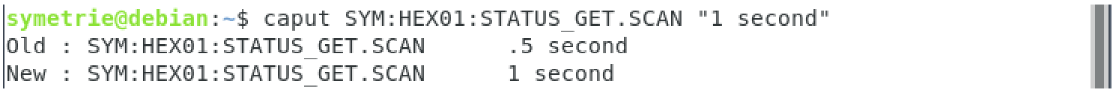
```
symetrie@debian:~$ caget SYM:HEX01:STATUS_GET.SCAN
SYM:HEX01:STATUS_GET.SCAN      .5 second
```

Modification of status scan rate:

Command:

```
caput SYM:HEX01:STATUS_GET.SCAN "1 second"
```

Command and reply in a terminal:



```
symetrie@debian:~$ caput SYM:HEX01:STATUS_GET.SCAN "1 second"
Old : SYM:HEX01:STATUS_GET.SCAN      .5 second
New : SYM:HEX01:STATUS_GET.SCAN      1 second
```

6 SOFTWARE RELEASE HISTORY

6.1 RELEASE 5.0.7.210830

Update to guarantee compatibility with API version: 5.0.210830:

Add:

- API: Error codes: 34 to 36
- API: Command returns: -92; -1017

Update to guarantee compatibility with API version: 5.0.201028:

Add:

- API: CFG_POWER command
- API: Error codes: from 30 to 33.
- API: Command returns: from -83 to -90.

Update:

- API: POWER command: modification of the command arguments.
- API: CFG_CONTROL command: add of delay parameter

6.2 RELEASE 5.0.5.191103

First release