

**Motor Unit for Oblique Viewer
Product OV2SMZ-MOT**

Contents

Introduction	2
Motor Unit for Oblique Viewer	3
Installation	4
Installation of Motor Unit for Oblique Viewer.....	4
Operation	5
Use of the Motor unit for Oblique Viewer	5
Computer Control	6
Communication Specifications	6
General Format Of Commands	6
ASCII Commands	8
Technical Notes	13
RS-232C Hardware Connections	13

Introduction

The Motor Accessory for the Oblique Viewer provides motorized operation of an Oblique Viewer. It allows either push-button controlled motorized rotation of the Oblique Viewer, remote control from an external computer, or foot-switch actuation using an optional Foot-Switch accessory. The Motor Accessory consists of a motor unit which attaches to the Oblique Viewer, a Pendant control unit, and an AC power supply.

FEATURES:

- Motorized continuous rotation through 360 degrees.
- Pushbutton actuation of rotation.
- Remote control of rotation from remote computer via RS-232C communication.
- An optional foot switch is available for use with the motor accessory.

Installation

Installation of Motor Accessory

Remove the Oblique Viewer from its Stereo Microscope.

To install the Motor Unit on an Oblique Viewer, align the "yoke" over the mounting ring of the Oblique Viewer with the three holes in the yoke aligned with the corresponding threaded holes in the mounting ring. Attach the yoke with the three pan-head screws provided.

Stretch the urethane drive-belt over the pulley of the Oblique viewer and the motor pulley.

Motor Unit installed on Oblique Viewer

Operation

Use of the Motor Accessory for the Oblique Viewer

To change the rotational view of the specimen rotate the Oblique Viewer either manually or by the motorized system. The motorized system can be actuated by push-button control, optional foot-switch, or remote control by an external computer via RS-232C communication.

Computer Control

Communication Specifications

The Oblique Viewer Communications interface is an interface between a host computer and the controller. The communication is established through an RS-232C serial connection. The programming protocol is with text (standard ASCII alphanumeric characters), along with some special control characters such as carriage returns, spaces and tabs. The controller responds to a set of built-in commands with unique names. The commands can be executed by simply sending the command name with some parameters (if required). The controller will respond in ASCII and may include the result requested. The Oblique Viewer controller is a stand alone system, but was designed to operate with other components, therefore it is necessary to distinguish it from, say, an xy-stage.

General Format Of Commands

Each line sent to the Controller should have a command and be terminated with a carriage return character. The first item on the line is the command. Each line can contain only one command and the Controller's commands are not case sensitive. The allowed commands are listed below. After the command are the parameters, some commands have no parameters. Finally, each command must be terminated with a carriage return character. The carriage return indicates to the Controller the end of a command. The specific items can be separated with white space characters (such as spaces, tabs). The entire command string cannot exceed 40 characters.

(command) [*data*] <cr>

where:

(command) any valid ASCII command.

[data] ASCII numeric data (if applicable).

For Example:

Command: **Where<cr>**
Response: **:A 1002<cr>**

Response

:A <DATA><cr> Everything is ok <returned data>
:N <ERROR CODE><cr> Error.

Every command returns a response: The response is in the form of a colon followed by a status character (either an A or N). The colon is sent by the Controller as soon as the command is received. The status character is not sent until the function has completed (i.e. after the motor has moved/stopped). Do not send another command until the last function has been completed and returned a response. If for some unknown reason the Controller does not respond with a colon, then the command was not received properly (due to communications problems) and the command must be resent. In this case, the Controller's internal buffer must be emptied by sending an ESC character (ASCII 27). This is necessary since your last command may have been partially received and still reside in the Controller's internal buffer. It is not a bad idea to send an ESC character before every command, but it is not necessary.

Examples:

command:	M 1001<cr>	(move to location 1001)
response:	:A <cr>	(everything is ok)
command:	W <cr>	(where is viewer?)
response:	:A 1001 <cr>	(z-axis position is 1001)
command:	AQRST<cr>	(an illegal command)
response:	:N -1 <cr>	(error code -1)

PRESENTLY ASSIGNED ERROR CODES

-1 unknown command

ASCII Commands

Halt Motor:

format: **HALT<cr>**

Hex Code: 0x7D

This command halts the motor during a move command. It effectively cancels the current movement.

Response:

A positive response is sent back when the command is complete.

:A<CR>

ASCII Commands

Set Current Location:

format: **HERE XXXX <cr>**

or **H XXXX<cr>**

This command will adjust the internal (to the controller) current location of the viewer. This will effectively adjust the location of the origin.

Response:

A positive response is sent back immediately after the command is received.

:A<CR>

Example:

HERE 1000<cr> The current location becomes the 1000 position internally.

ASCII Commands

Move Absolute:

format: **MOVE XXXX<cr>**
 or **M XXXX<cr>**

This command will move the viewer to the location XXXX.

Response:

A positive response is sent back when the command is complete.

:A<CR>

Example:

MOVE 1000 <cr>

ASCII Commands

Move Relative:

format: **RELMOVE XXXX<cr>**
 or **R XXXX<cr>**

This command will move the z-axis a relative amount XXXX from the current location.

Response:

A positive response is sent back when the command is complete.

:A<CR>

Example:

RELMOVE 1000 <cr> +1000 from the current location.

ASCII Commands

Speed:

format: **SPEED<cr>**

This command will tell the operator the current value of the maximum speed of movement for the MOVE commands. The range of speed is 0 to 10, with a larger number representing a faster speed.

Response:

A positive response is sent back immediately after the command is received.

:A<CR>

Example: **SPEED<cr>**

Response: **:A 10<cr>** The speed is set at 10.

ASCII Commands

Get Version:

format: **VERSION<cr>**

Hex Code: 0x7C

This command returns the current version code of the firmware.

Response:

A positive response is sent back when the command is complete with the current version number.

:A 1.0<CR>

Example: **Version<cr>** What is the ROM version Number

Response: **:A 1.0**

ASCII Commands

Who:

format: **WHO<cr>**

This command will return the current microscope accessory being used. In this case it returns, "**Oblique Viewer Attachment**".

Response:

A positive response is sent back immediately after the command is received.

:A<CR>

Example:

WHO<cr>

Response:

:A Oblique Viewer Attachment<cr>

ASCII Commands

Get Current Location:

format: **WHERE <cr>**

or **W <cr>**

This command will query the controller for the current location of the viewer.

Response:

A positive response is sent back immediately after the command is received.

:A XXXX<CR> The current location.

Example:

WHERE <cr>

The current location is sent back from the controller.

Response:

:A 1456<CR> The current location.

ASCII Commands

Set Zero Of Origin:

format: **ZERO <cr>**

This command will set the origin to the current location. This results in the current location being the new ZERO (origin).

Response:

A positive response is sent back immediately after the command is received.

:A<CR>

Example:

ZERO <cr> The current location becomes the ZERO position.

Technical Notes

RS-232C Settings:

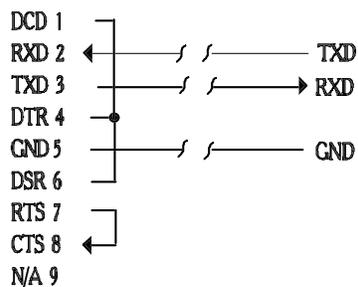
Format: 8-Bits, 1-Stop Bit, No Parity

Baud Rates: 9600

The stepper motor rotates 400 steps per revolution. While the motor is stationary, the stepper motor is automatically powered-down. A current only strong enough to provide the required detent torque is applied in this power-down mode. This reduces the power dissipation in the motor and the controller. Full current is automatically applied during rotation of the motor.

The drive belt is made of polyurethane and thus is very clean and abrasion resistant to ensure clean room compatibility.

RS-232C Hardware Connections



DB9F
(IBM AT-Style Serial
Mating Connector)

RJ-11 MOD4 to
Remote Focus
Accessory

For warranty repair return the product to the warranty department of
Conix Research at:

Conix Research Inc.
857 28th Street
Springfield, OR 97477
(541) 747-8512

You should provide a written description of the problem with the unit.
Consumer must prepay all postage, shipping, insurance, and delivery costs
associated with the return of the product.

For more information refer to the Conix Research Limited Warranty Card
provided with this product.

 **CONIX RESEARCH**
Inc
857 28TH SPRINGFIELD, OR 97477
PHONE (541) 747-8512 FAX (541) 747-8528