Ealing

USB-Motor Controller

Ealing Catalog, a Hyland Optical Technologies Company

Hyland Optical Technologies

340 El Pueblo Road, Suite E

Scotts Valley, CA

United States

Phone 1-831-461-1930

Fax 1-831-461-1136

Internet ealingcatatlog.com

E-mail sales@ealingcatalog.com

Images inside this manual may show slight deviation from the actual delivered system.
This does not affect function or operation.
Responsible for the content published: Hyland Optical Technologies

The manufacturer maintains the right to change information and technical equipment without notice.

Safety Rules

WARNING!

Read the safety precautions in this section before installing, powering, operating or servicing this product. Risk of Danger

Symbols used in this manual

The following symbols are used to identify important safety information

 WARNING! Danger of crushing if connected to motorized devices.

 CAUTION: Special attention to this point is necessary to meet health regulations and to avoid damage.

 ATTENTION: Please be aware of ESD damage when adding or removing electronic modules or control devices for this product.

 An important note to ensure safe or stable operation.

Important safety information

Before operating the USB-Motor Controller system, read the manual carefully to prevent damages to humans or animals, integrated devices and connected devices.

Hazardous Considerations

Notes about the electrical safety:

- The USB-Motor Controller is exclusively supplied by USB 1.0 – 3.0.
- For electrical insulation safety please consider the insulation of the primary power supply of the USB supporting device.
- Always check for loose connectors or wires.
- Do not touch any uninsulated (due to damage) electrical parts, beware of electrical shock.
- Always check for fixed cables and connectors.

> ⚠️ **CAUTION for OEM-usage:**
>
> **The USB-Motor Controller is also available as an OEM-product. In this case, it is only for usage in systems such as customer designs, and the customer has to include the USB-Motor Controller in the safety-concept. The OEM-customer is responsible for a safe operation of the USB-Motor Controller. In any kind of OEM usage, the safety-concept must be evaluated and tested in the end-use application. We will be glad to give detailed assistance to our customer, if there are any questions in the safe system integration.**

Identification of manufacturer explanatory label:

HYLAND Optical Technologies Scotts Valley, CA 95066 Model: USB-Motor Controller® S/N: 123456 Manufactured: MM/YYYY	Size: 46 x 26mm Colour: silver / black Location: on the bottom side

Principles of ESD Control

 ATTENTION:

Please be aware of the following measures against ESD-damages when connecting motors to the USB-Motor Controller.

1. Grounding:
The best prevention against ESD damages is to issue a grounding device to personnel:

- Grounding with help of a special reel
- Conductive footwear
- Heel bands should be worn on both feet, in order to ensure a constant contact to the ground.

2. Prevention of ESD damages of the USB-Motor Controller:
Avoid touching the metallic contacts. Please avoid the transfer of modules from one person to another one. Please keep the parts as long as possible inside the protection bags.
If possible, please build up an ESD protected and staticsafe workplace.
The minimum requirement for the workplace contains:

- Grounding cable for hand reel (grounded)
- Grounding cable for dissipative workplace (grounded)
- Grounding cable for conductive flooring (grounded)
- Hand reel
- Dissipative workplace
- Conductive flooring
- Use of protected insulation bags (comes with the part)

3. Neutralization:

If an isolator is statically loaded, ESD damage can be caused by contact. In order to reduce this risk, an ionizer can be used, which produces billions of charged particles and thus neutralizes the static loading of the isolator.

ATTENTION:

Neutralization replaces neither grounding nor prevention! The consideration of these measures reduces the occurrence of ESD damages, but cannot prevent them completely.

Table of contents

1. Introduction

1.1 USB-Motor Controller general description and key facts

The Ealing USB-Motor Controller is an electronic device to control and drive up to two micro DC servo motors typically used in ultra-precision positioning system with resolutions of less than one µm.

The USB-Motor Controller is very simple to operate, insensitive to transportation and environmental influences, in order to approach "hands-free operation".

Key facts:

- Closed and open loop control for positioning 2 DC servo motors
- Power supply and communication with PC over USB. No additional power supply needed
- Rated voltage range from 3V to 12V
- Maximum output power per motor is 2.5W
- Integrated current measurement to limit power consumption
- Hardware inputs for 2 limit switches and 2 manual buttons per motor channel
- Controller is usable without PC connection just by the hardware inputs
- Communication by ASCII commands list
- USB driver supports virtual COM port
- Use more than one USB-Motor Controller on one PC (multi axis control)
- Windows™ based control software included
- OEM versions available

2. Technical Data

2.1 General Information

It is possible to connect the USB-MotorController directly to the USB-Port of a PC. The controller is able to draw up to 0.5A current from the USB-Port. Be sure that the USB-Port can provide this current to avoid damages on the USB-Port. We recommend using an USB-Hub with an extra power supply.

The user can select the rated voltage for both motors from 3V to 12V. Possible options are 3V, 6V (for 5V motors, please select 6V) and 12V. The controller measures continuously the power consumption of both motors and stops moving when power exceeds 2.5W. After exceeding 2.5W the controller is in an error state which needs to be resetted by a reset command.

Principally the controller has to positioning modes, the jog mode and the PID controlled mode. In jog mode the user defines a speed of the motor in percent and starts the jogging in the desired direction. In PID controlled mode the controller uses feedback of the encoder to control the motor to a target position. The user needs to define the target position in micro meters and the three PID parameters. There is a default parameter set stored in the controller. If needed, please feel free to modify the PID parameters according to the software manual. Also it is necessary to define (enter) the mechanical parameters of the motor like gear ratio, spindle feed per spindle rotation and encoder counts per motor rotation. Please refer to the specification of the linear motor.

2.2 Mechanical and electrical specifications

The USB-Motor Controller is consists of up to 2 independent motor outputs.

The following table shows the typical mechanical and electrical specifications.

Parameter	Motor 1	Motor 2
Rated voltage:	3V-12V	3V-12V
Leadscrew pitch (mm), depends on type of spindle:	0,5	0,5
Motor speed (rpm) no load	12000	12000
Gearhead ratio (configurable by software, others on request)	485:1, 256:1	485:1, 256:1
Encoder counts per revolution	60	60

Table 1: typical mechanical and electrical specifications

2.3 Operating modes and signal specifications for the inputs and outputs

Operating Modes	Mode 1 jog Mode 2 PID controlled
Limit Inputs	> 500kHz 0…5V; Input high voltage >2,4V
Jog Inputs	>500kHz 0…5V; Input high voltage >2,4V

Table 2: Operating modes and signal specifications

2.4   Dimensions and weight

Case style	Tube housing
Dimensions	84 x 69.1mm x 28mm (l x w x h)
Weight	Approx. 0,2kg
Protection Class	IP2x according to IEC (EN) 60529

Table 3: Dimensions and weight
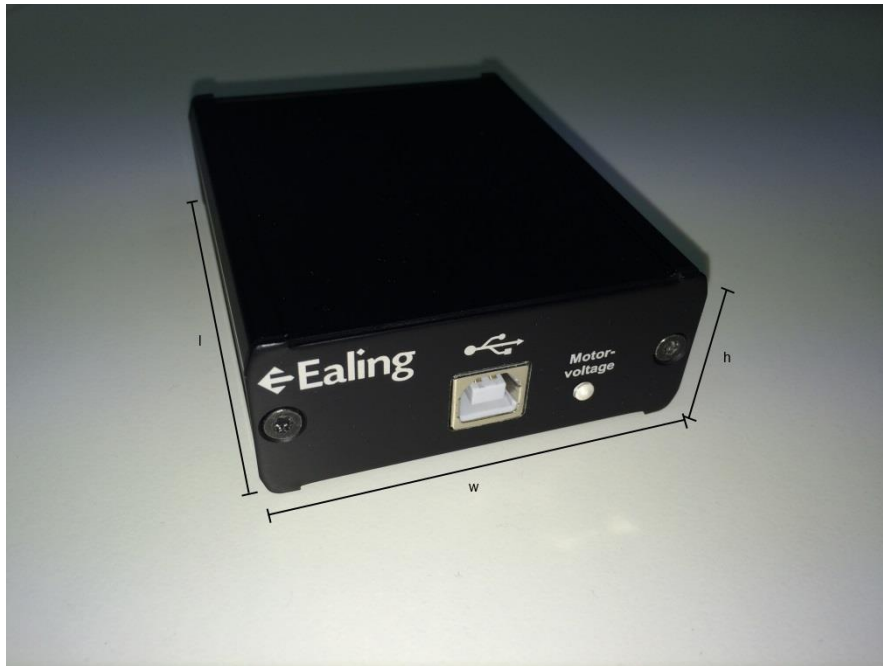
Figure 1: USB-Motor Controller main view

2.5   Electrical supply and interface specifications

Supply voltage	5.00 VDC +/- 0.50 VDC
Supply current	typical 0.4A, max. 1.0A
Control interface	USB 1.0-3.0, recommended USB 2.0

Table 4: Electrical supply and interface specifications

2.6   Thermal management

The specified and recommended operating temperature range is 0…50°C. The USB-Motor-Controller is designed to dissipate the heat through its housing to the air.

Do not cover the housing!

3. System overview

For easy handling, the motor voltage status LED shows if the internal power management is working. Please refer to the *?BMo* command from the command list to choose the motor voltage.

Orientation of the connectors of the USB-Motor Controller
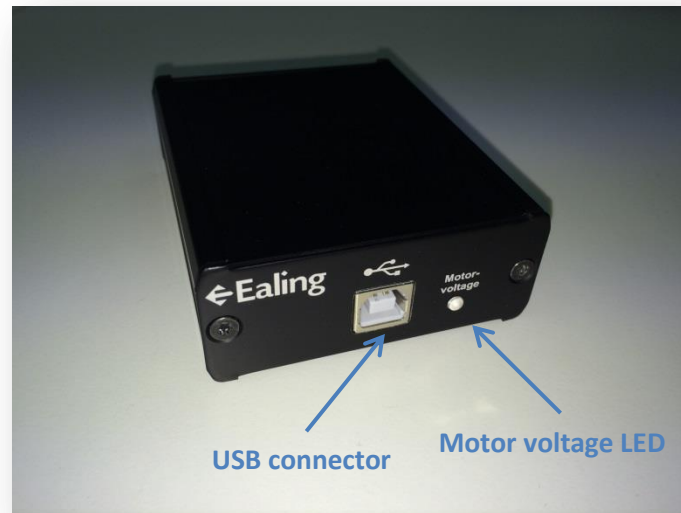
Figure 2: USB-Motor Controller front view

Figure 3: USB- USB-Motor Controller® back view

3.1 System connectors

USB port	
USB type-B 4pin	**Function**
1	+5V bus power (up to 1000mA current consumption)
2	USB D-
3	USB D+
4	GND USB

Table 5: Pin allocation of the USB port
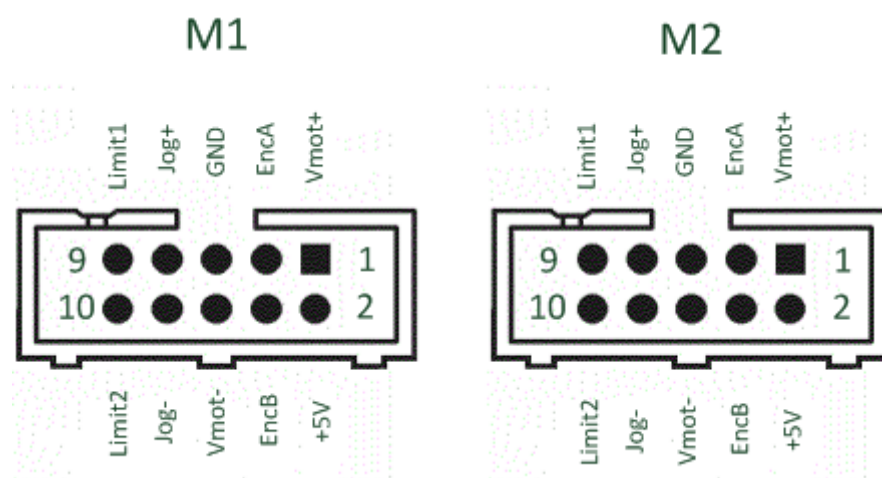
Figure 6: Pinout of the connectors for the motor, Encoder, limit switches and jog switches.

Motor 1 I/O port	
10pol male	**Function**
1	Vmot+ : Output for motor 1
2	+5V: supply voltage for encoder, limit switches and jog switches
3	EncA: Input for Encoder channel A
4	EncB: Input for Encoder channel B
5	GND: Ground for +5V supply
6	Vmot-: Output for motor 1
7	Jog+: Input for jog switch (configurable as TTL active low or high input)
8	Jog-: Input for jog switch (configurable as TTL active low or high input)
9	Limit1: Input for limit switch (configurable as TTL active low or high input)
10	Limit2: Input for limit switch (configurable as TTL active low or high input)

Table 7: Pin allocation of M1 I/O port

Pin allocation for M2 I/O port is similar to M1.

4. System components

4.1   Standard set of delivery

The USB-Motor Controller is shipped with the following parts:

- USB-Motor Controller

- USB 2.0 PC to USB-Motor Controller connection cable (1.5m)

- Ealing Motor Controller Software (content of the storage device)

- USB-Motor Controller user manual (content of the storage device)

For ordering spare parts, please use the above description of the sub-part or contact your local distributor for assistance in getting the item number.

5. Installation and starting into operation

5.1   Unpacking

When you receive your USB-Motor Controller please immediately inspect the shipping container. If there is any damage (holes or crushing, etc.) insist that a representative of your local carrier is present while you unpack the contents.

Carefully inspect your USB-Motor Controller as you unpack it. If any damage is evident, such as dents or scratches on the covers or broken parts etc., immediately notify your carrier and your local sales distributor.

Please keep the original shipping boxes and packaging in case a claim is filed for damage that has occurred during transit.
Anyway, if you find a damage claim, you may need them to demonstrate that the damage occurred as a result of shipping.

The USB-Motor Controller is shipped with the following components:

1. USB-Motor Controller
2. USB cable
3. storage device (USB stick)
4. User manual

Please make sure that all components are present.

5.2   General Installation

The USB-Motor Controller is very easy to set up. Please verify the ambient temperature and the USB-Motor Controller temperature before operating. If the USB-Motor Controller housing temperature is out of the specified operating temperature (0 – 50°C) wait until the USB-Motor Controller has reached the specified temperature range.

After unpacking and inspection the controller, follow these instructions for connecting the unit:

1. Place the USB-Motor Controller on a table.

2. Please install the control software on a local PC with a free USB 2.0 port. Please refer to the USB-Motor Controller – software manual.

3. Connect the optional remote control box to the motor connectors. If you will be using your own connection cable (for example as an interconnection cable for machine interfaces), please refer to chapter 3.2, pin allocation and additional explanation.

4. Select a free USB 2.0 port on a local PC and use the USB cable for connecting the USB-Motor Controller to the local PC with the installed software.

    Be sure that the USB port of the PC is able to handle currents up to 0.5A. Otherwise use an USB 2.0 HUB with external power supply.

5. Please open the software and check the logically connected LEDs.

6. Select the right motor voltage in the software and connect motors to the motor connectors.

6. System description

6.1   Description of the USB-Motor Controller

The USB-Motor Controller Global control and positioning modes

Principally the USB-Motor Controller has two positioning modes, the jog mode and the PID controlled mode. In jog mode the user defines a speed of the motor in percent and starts the jogging in the desired direction. In PID controlled mode the controller uses feedback of the encoder to control the motor to a target position. The user needs to define the target position in nanometers and the three PID parameters. It is necessary to define the mechanical parameters of the motor like gear ratio, spindle feed per spindle rotation and encoder counts per motor rotation.

Since the controller has no hardware control elements it can only be remotely controlled by the integrated USB-interface.

You may use the provided graphical user interface software for Microsoft Windows (GUI) to control the device via the USB interface to get easy access to all main functions of the USB-Motor Controller.

See the additional "Ealing USB-Motor Controller – Software Manual" available on the provided storage device as an Adobe PDF file for details.

See the additional "USB-Motor Controller Command List" available on the provided storage device as an Adobe PDF file for details on the different available commands.

6.2 Controllable functions and operating modes

6.2.1 General controller information and control

It is possible to readout general data of the controller like firmware version, device id and serial number. Please refer to the ?GFw and ?GSN command in the command list. Further the ?RsC command allows the user to reset the motor controller. As soon as the reset cycle is finished the controller sends the $RsC message (adhoc).

The user can update the firmware over the USB port with flasher tool. The bootloader state remains for 3 seconds and waits for the update. If there comes no update the controller starts into the normal operating program.

The controller is able to detect if one or two motors are connected on the motor ports. Please refer to the ?GSI command of the command list.

6.2.2 Setting motor voltage and measure current

The USB motor controller is able to change the supply output voltage for the connected motors. It is possible to activate or deactivate the motors power supply and choose the motor voltage between 3V, 5V, 6V and 11.7V (12V). Further the controller can be configured to power up the output voltage or not. Please refer to the commands ?BMo in the command list.

The controller has an integrated measurement device to measure the motor current of the unit. This device measures the current drawn by the motor power supply. The measurement can be triggered by. Please refer to command ?MCu.

6.2.3 Motor and spindle information

For correct positioning the controller needs information about the connected motors and spindles. The available parameters are the gear-ratio, encoder-counts per encoder revolution and spindle pitch per spindle revolution. Please refer to the command ?MDa in the command list.

6.2.4 Positioning

The controller handles each motor separately so the description of the positioning is similar for both.

The controller recognizes each count of the encoder and will send this change of position and the speed as an adhoc message. Please refer to ?APo command.

The mode of positioning can be set by the ?PMo command. In jog mode it is possible to set the speed and direction. Please refer to the command ?JoS.

In the PID mode the controller tries to move the actual position to a defined target position. The quality of the closed loop control depends on the adjusted PID parameters and the minimum of the control-variable. The minimum control-variable is a current threshold when the motor starts to move. The controller has a routine to find this threshold. The controller increases current till the motor is moving and then stops the motor again. Please refer to the commands ?TPo, ?KPa and ?FMC to adjust the PID control loop.

The devices are delivered with standard settings for the PID control loop.

6.2.5 Set reference and software limits

The user can set the actual position to a defined position to set the internal reference point of the system. Further the user can set a minimum and maximum limit to define an area for save movement inside the allowed travel length.

Please note: The software limits are ignored in jog mode. If the user sets the limits in a way that the actual position is out of bounds the controller will set the target position to the depending limit. This will produce a movement if the PID mode is active. Please refer to the commands ?SZe and ?Lim.

6.2.6 External inputs

Each motor port has two pins to connect buttons to trigger by a hardware switch a jog in each direction. The speed is also set by the ?JoS command. There are also two pins to connect hardware limit switches. The limit switches are assigned to a direction. Limit 1 blocks movement in the negative direction and allows movement in the other direction. For Limit 2 it is the other way around. The user can set the trigger slope of the input pins to rising or falling edge or disable the trigger (reaction). Please refer to the commands ?REL and ?REJ.

If a high signal (TTL) is detected on the jog input pins, the controller sends adhoc the new position mode (?PMo). If a limit switch is triggered the controller sends an adhoc message for the detected event. Please refer the command ?GAS.

6.2.7 Error handling

As soon as the controller detects an error, it will send an adhoc for the detected error event. If the controller was forced to switch into an error state the error needs to be cleared (reset). Please refer to the commands ?GLF and ?RsC.

7. Communication Interfaces

7.1 Logic Interface

7.1.1 Overview

The USB-Motor Controller has a USB control interface.

There is an adhoc mode integrated, which promptly sends status changes via USB. This mode cannot be deactivated.

7.1.2 Protocol

Communication protocol is pre-defined with the following parameters:

- USB: 500000bits per second, 8 data bits, 1 stop bit, no parity, accessible via virtual comport

- No hardware handshake

- All commands consist of printable ASCII characters, naturally "A-Z", "0-9", ">" (ASCII greater than), "!" (ASCII exclamation mark), "?" (ASCII question mark), "$" (ASCII dollar sign) and "[…]" (ASCII square brackets).

    IMPORTANT NOTE:

    If a communication string contains more than one parameter, the parameters will be separated by a "§" sign. The "§" (section character) is not a standard ASCII character. It is an iso-8859-1, also called "Latin1" encoded character. It has a corresponding hex byte value of 0xA7. For further details like keyboard layout also see http://en.wikipedia.org/wiki/Section_sign

- Each command ends with a carriage return (CR).

- Each question command sent from the host computer to the device starts with a "?" question mark. Please note that a command which sets a parameter also starts with a preceding question mark as it is a command sent from the computer to the device.

- Each answer command to a question, sent from the device to the host computer, starts with an "!" exclamation mark.

- Each Adhoc command, which is sent from the device to the host computer without a preceding question starts with a "$" dollar sign. For example: After powering up the system sends a "$RsC>" to confirm the reset.

- To address a single motor, a reference to the "position number" of the motor is added after the command in brackets. E.g. to get the positioning mode of motor 2 send: ?PMo[2]

- The protocol is case sensitive.

- The length of a command is not specified. The maximum length is set to 42 characters including preceding sign and carriage return.

- A successful execution of a question is indicated by a ">" greater than sign.

- An unsuccessful execution is indicated by an "x" sign, for example, if the value you try to set is out of valid range or if it is not possible to execute the command in the actual operating state.

- An unknown command or an incomplete command is indicated by a *"!UK"* answer.

7.1.3 Questions (PC ➜ Device)

A question command from the host computer to the device always starts with a "?" question mark followed by a three character long command indication. Remember that the protocol is case sensitive.

There are two types of question commands. One asks for a value and one sets a new value in the device. The first one is completed by a CR carriage return directly after the command indication. The other one needs the desired value and is then concluded with a CR carriage return. The desired value sometimes has to be given in decimal or hexadecimal representation.

7.1.4 Answers (Device ➜ PC)

Every question causes an answer from the device. The computer must not send any more commands, until he gets the answer or unless a timeout occurs.

If the device is unable to identify the question, it will send back a *"!UK"* (Unknown Command).

7.1.5 Adhoc commands (Device ➜ PC)

Switching on the power supply causes the device to send a *"$RsC>"* if the boot operation was successful.

A reset command can be sent anytime and is also followed by a *"$RsC>"* answer, if booting was successful.

7.2   Command details

As a matter of form all existing commands and answers are listed in the command list. This document only exists in digital form on the enclosed storage device.

The following declaration is needed to understand the table:

abbreviation	character	ASCII HEX
CR	carriage return	0D
>	greater than	3E
$	dollar sign	24
?	question mark	3F
!	exclamation mark	21
§	section sign	A7
[	open square bracket	91
]	close square bracket	93

Table 8: ASCII command abbreviations

As an example, the following image shows one command from the command list to explain the setup of the table.

command type:	GFw														
function:	Get Firmware														
explication:	Ask for the model code and firmware version														
byte number:	1	2	3	4	5										
command:	?	G	F	w	cr										
answer 1:	!	G	F	w	Model code		§	Device-ID		§	Firmware			cr	
interpretation 1:	The answer contains the the Model code of the device. After a "§" a Device-ID is displayed. After another "§" the firmware version is displayed. Max byte length for model code and firmware is 14byte, for Device-ID 7byte.														

Table 9: Example of one command from the command list

The table shows the command type and function in line one and two and gives a short explication in row three. After that, the byte numbers for the complete command are given. If byte number ends up before the command ends, this means that the byte length of the command or answer can vary. The command and answer line show the exact appearance of the data. The command sent from the computer to the device is preceded by the question mark. Afterwards, the command to ask for model code and firmware is sent. The answer 1 line contains the answer, sent from the device back to the computer. The exclamation mark is followed by the repetition of the question. After that, three different parameters are given, the model code, a device-id and the firmware, each separated by a "§" sign. The byte length of the three parameters can be different from device to device, so the byte length is not specified. The string is closed with a CR sign. Finally there is a complete description of the command in the last line.

8. Troubleshooting

Error	Action needed
Over current	Check selected motor voltage. Check if motor is blocked. Reset error by "?RsC" command
Controller fault	Reset error by "?RsC" command

Table 10: Troubleshooting

9. Maintenance

The USB-Motor controller is a maintenance free product. Please operate the controller under a clean and dry environment condition.

10. Warranty

The warranty is limited by the following:

- Unauthorized modification or misuse
- Operation outside the environmental specifications of the product

11.   Waste disposal

WEEE (Waste of Electrical and Electronic Equipment)

Recycling of Electronic Products, disposing of this product.

In 2012 the European Union introduced regulations (WEEE – 2012/19/EG) for the collection and recycling of all waste electrical and electronic equipment. It is no longer allowable to simply throw away electrical and electronic equipment. Instead, these products must enter the recycling process.
This product must not be disposed in normal garbage!

When this Ealing product is no longer used or out of order and should be disposed please refer to the service address indicated in the scope of delivery to send back the device in its original package to the manufacturer to enter the recycling process.

We thank you for your understanding!

12.   Spare parts and service (Manufacturer identification)

More detailed specifications, declarations of conformity for the EU, the USA, and help in your solutions for applications on demand.

Please contact service at:

Hyland Optical Technologies

e_mail: tech@ealingcatalog.com

website: www.ealingcatalog.com

Tel.: 831-461-1930

Fax.: 831-461-1136