

Theia lens motor controller instructions



Lens motor control
specification and instructions

MCR600 assembly
MCR500 assembly
MCR400 board

Theia lens motor controller instructions

Table of Contents

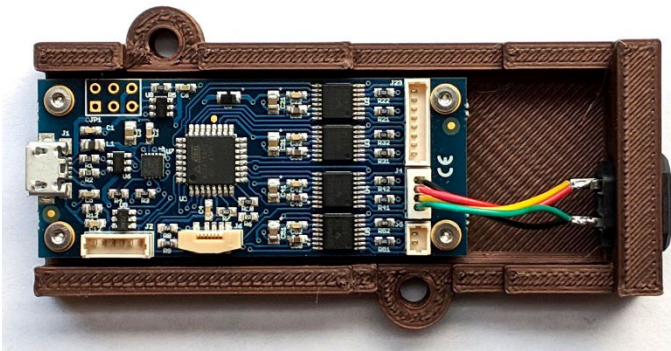
1. Precautions and disclaimers	3
2. Windows Test Software	4
2.1 Installation	4
2.2 Run the demonstration controller software.....	7
2.3 Communication.....	8
2.4 Firmware update.....	8
3. Hardware specification	9
3.1. Features:	9
3.2. Ratings and parameters	9
3.2.1 Absolute maximum ratings:.....	9
3.2.2 Mechanical (board only):	9
3.2.3 Operating parameters:.....	9
3.3. Connectors	10
3.3.1. USB (J1)	10
3.3.2. I2C (J2).....	10
3.3.3. UART (J2).....	11
3.3.4. SPI (JP1).....	11
3.3.5. Reset (JP1).....	12
3.3.6 Factory default reset.....	12
3.3.7. General Purpose I/O	12
3.3.8. Lens Connectors (J23, J4, J5, J6).....	12
3.3.9. Power Supply (J1, JP1, J2)	12
4. Software communication structure	13
4.1. Overview	13
4.2. Different bus behaviors.....	13
4.2.1 USB bus mode	13
4.2.2 I2C server bus mode	14
4.3. Serial Protocol	14
4.3.1. Move Stepper Motor	15
4.3.1.1. Motion Control for DC motors.....	15
4.3.2. Read Firmware Version.....	16
4.3.3. Read Serial Number of the board	16
4.3.4. Read Motor Setup.....	17
4.3.5. Write Motor Setup.....	18
4.3.6: Switch active bus.....	19
4.3.7: Change i2c address	19

Theia lens motor controller instructions

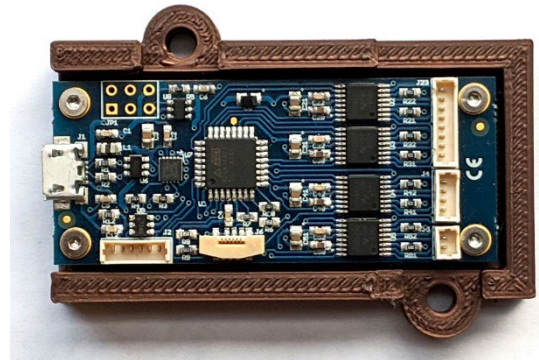
This lens control board can be used with Theia Technologies® motorized lenses that use a P-iris.

Supported lenses	Motor control board
TL936P R3, TL936P R5	MCR400 or MCR500
TL936P R4, TL936P R6	MCR400 or MCR600
TL1250P R3, TL1250P R5, TL1250P N5	MCR400 or MCR500
TL1250P R4, TL1250P R6, TL1205P N4, TL1250P N6	MCR400 or MCR600
TL410P R3, TL410P R5	MCR400 or MCR500
TL410P R4, TL410P R6	MCR400 or MCR600
ML410P R3, ML410P R5	MCR400 or MCR500
ML410P R4, ML410P R6	MCR400 or MCR600

The motor controller is available in 3 versions. There are 2 assembly versions (MCR500 and MCR600) which come with a 3D printed base to allow easier handling of the board and are packaged in individual boxes. The only difference between the two is an adapter cable that allows connection to Theia's -R3/-R5 lenses which use the EYC221 CCTV connector instead of the Molex connector which is built into the board. Theia's -R4/-R6 lenses can be used without this adapter cable.

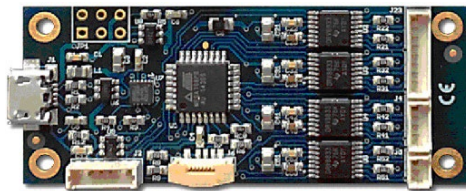


MCR500



MCR600

The third version of the controller is the board-only configuration. This version doesn't include the plastic base part and is packaged for higher volume purchases.



MCR400

1. Precautions and disclaimers

- Do not short circuit any part of the board
- Keep the controller board dry
- The controller board contains fuses but proper ESD protection procedures should be followed
- If the controller board becomes too hot disconnect it

- The controller is not designed for use in life sustaining or lifesaving applications.

Theia lens motor controller instructions

- The controller is not designed for use in applications where their failure may lead to significant loss of property or health of humans or animals.
- The Information given herein is believed to be accurate and reliable. However, users should independently evaluate the suitability of and test each product selected for their own applications.
- The products described herein are intended for integration. Users must get all required certifications and registrations for the resulting assembly including for the products described herein.
- We reserve the right to make changes to the products described herein without notification. A new specification replaces any previous ones.
- See Theia's standard terms and conditions for warranty and other information.

2. Windows Test Software

2.1 Installation

Theia provides a simple control program for interacting with the control board. This application is intended to be used as a test and demonstration program for controlling the lens motors. It can be downloaded from <https://www.theiatech.com/lenscontroller/>



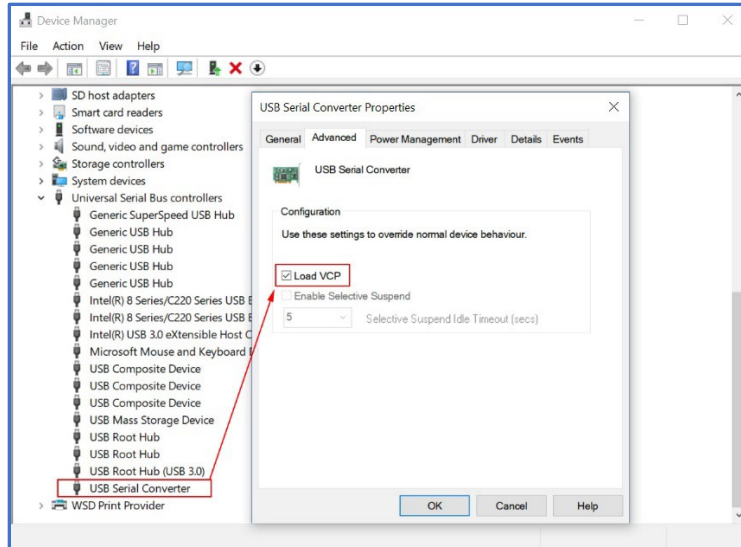
This software (installation and use) and the driver board are intended to be used by a skilled person in a laboratory or industrial setting. This is not intended to be a fool-proof consumer electronics ready controller.

Download and unzip the motor control program to a folder that is not system protected such as C:\Users\yourAccount\AppData\Local\Theia. Be sure to copy a short cut to your desktop for easy access to the test program.

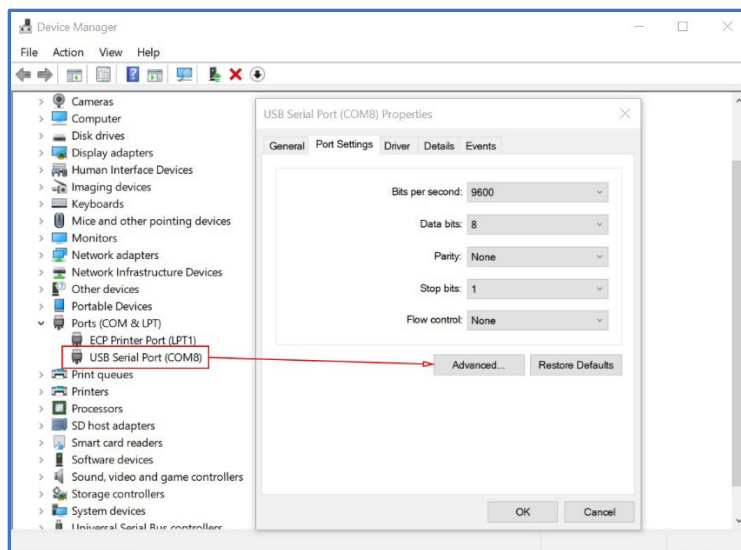
For initial setup the serial port controller may need to be configured with these following steps. For Windows:

1. Connect USB cable between the lens control board and computer with Windows 7/10
2. Open the "Device Manager"
3. Expand "Universal Serial Bus controllers" and open "USB Serial Converter"
 - a. If this option is not available, you may have to install the driver for the FTDI chip. Install the VCP driver from <https://www.ftdichip.com/Products/ICs/FT200XD.html>
4. In "Advanced" tab, check "Load VCP" (Checkbox must be checked)

Theia lens motor controller instructions

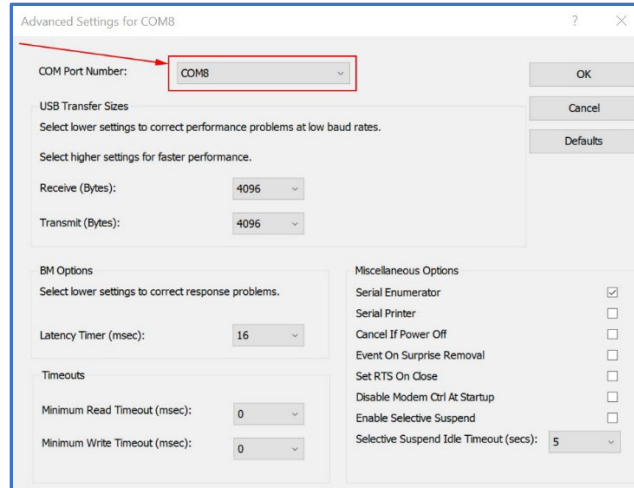


5. Click "OK" to save the settings
6. Disconnect the USB cable then reconnect the USB cable.
7. Expand "Ports (COM & LPT)" and open "USB Serial Port (COMx)"
The COM port for your controller board may have a different number. If the COM port is greater than 9 continue with the following steps.



8. In "Port Settings" tab, chose "Advanced .."
9. Set the COM Port below or equal COM9

Theia lens motor controller instructions



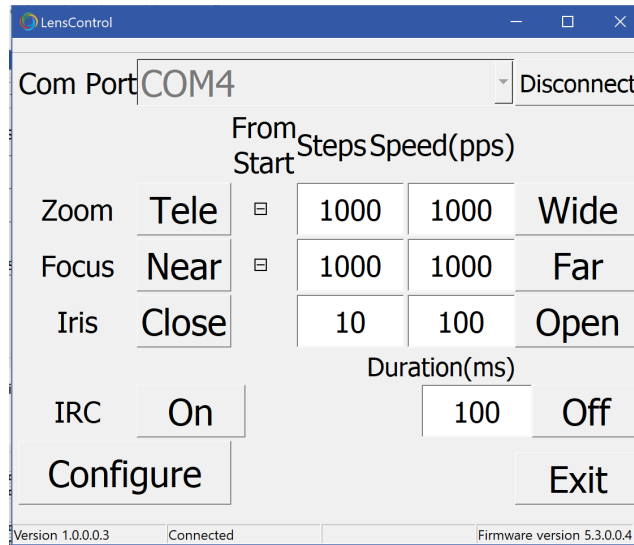
10. Click “OK” and “OK” again. Close the device manager. Your control board should now be setup with the COM port that was selected.

Theia lens motor controller instructions

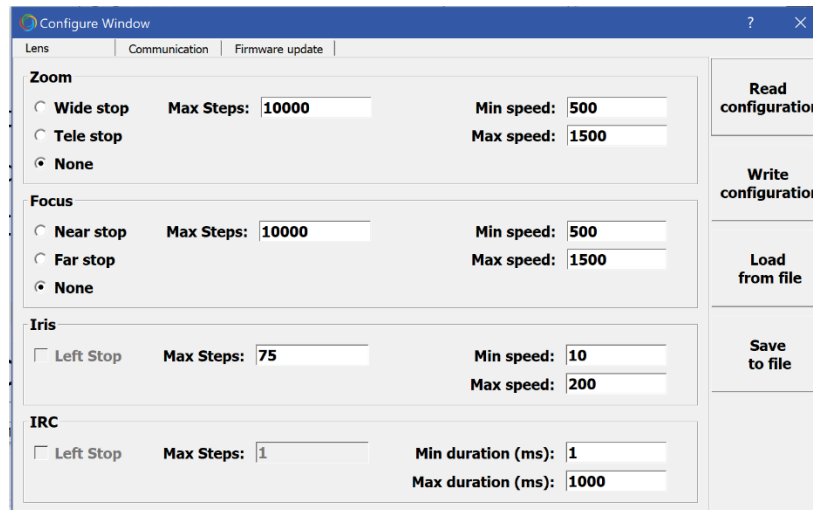
2.2 Run the demonstration controller software

The Windows control program can only communicate with the motor controller over USB. If the communication path was previously set to UART or I2C then change the communication back to USB via the “switch active bus” command over the current communication path. See section 4.3.6 and issue the switch bus command 0x68 0x01 0x0D. Or the controller can be reset to factory defaults by following instructions in section 3.3.6.

Start the LensControl software. Select the correct COM port from the drop down list and “connect” to the motor control board.



The configuration settings may need to be adjusted for your lens. Click on “configure” to bring up the parameter limits.



Check the datasheet of the lens for maximum speeds for zoom and focus. In general, these maximums should be about 1000pps. There is no issue setting the maximum number of steps greater than the steps available. This just limits the user input choices. Be sure to click “write” to store the changed values before closing the configuration window.

Theia lens motor controller instructions

The “stop” photo interrupter (PI) will be triggered when the focus or zoom ring reaches full rotation if the PI is on the correct side. Due to design constraints, the PI could be on either side of the focus and zoom movements.

Lens	Zoom PI	Focus PI
TL410P R6, TL410A R6	Tele stop	Far stop
TL1250P R6, TL1250A R6 TL1250P N6	Wide stop	Far stop
TL936P R6, TL936A R6	Wide stop and/or Tele stop	Near stop and/or Far stop

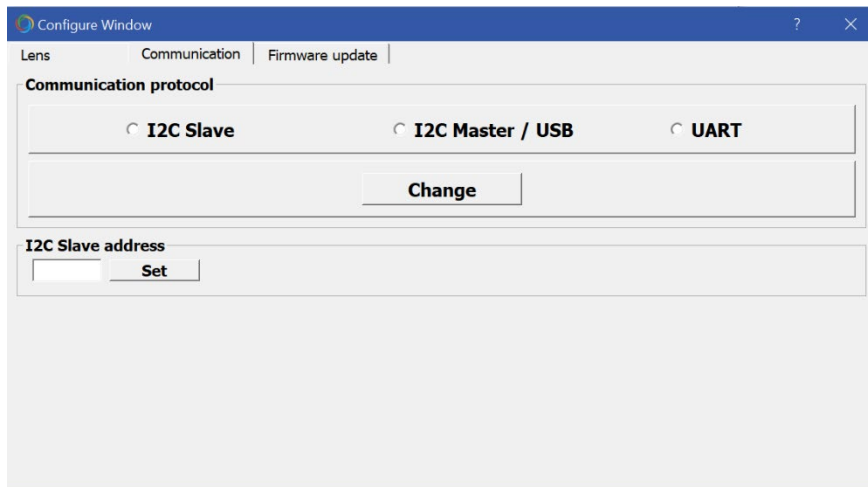
Older versions of the control firmware don't support the 2 photo interrupters and should be updated to version >1.0.0.1.2

Trouble shooting: If the program doesn't seem to work and then quits automatically, check that the com port is com9 or less (software section, step 7).

2.3 Communication

(FW >5.3.0.0 2)

The board communication protocol can be changed via the Windows control program. Follow the steps in section 2.2 to open the configuration window. Select the “communication” tab at the top of the window.



From here, you can change the communication path to I2C or UART. After switching to the new path, the board will automatically reboot and the Windows control program will no longer respond.

You can also set the I2C address in this window. The new I2C address will only become effective on a forced power cycle reboot of the controller.

2.4 Firmware update

Don't update firmware unless required to or requested by Theia. There could be hardware/firmware incompatibilities that can prevent the board from working.

Theia lens motor controller instructions

3. Hardware specification

3.1. Features:

- supports Theia Technologies® lenses
- controls P-iris, focus, zoom and IR cut filter motors
- reads photo interrupter limit switches for zoom and focus
- supports USB, I2C, and LV-TTL-UART
- small size of only about 2.4" x 1" (60x25 mm²) without base
- over-current protection
- over-temperature protection
- single 5V supply (via USB or pin connector)
- optional base has 2x M4 mounting holes (50mm separation at 45deg to board orientation)

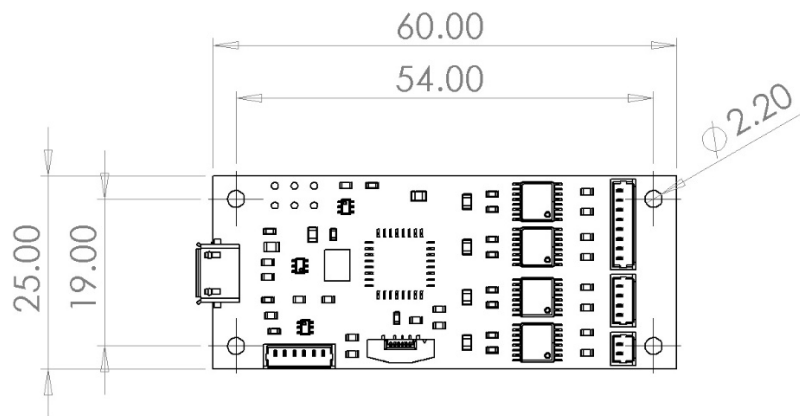
3.2. Ratings and parameters

3.2.1 Absolute maximum ratings:

Maximum Operating Voltage	5.5 V
Maximum current draw	<500mA
Storage Temperature	-40 °C ... 100 °C
Operating Temperature	-35 °C ... 85 °C

3.2.2 Mechanical (board only):

Length	60mm
Width	25mm
Height (without cables)	10mm
Weight	10g
Mounting holes	4x M2



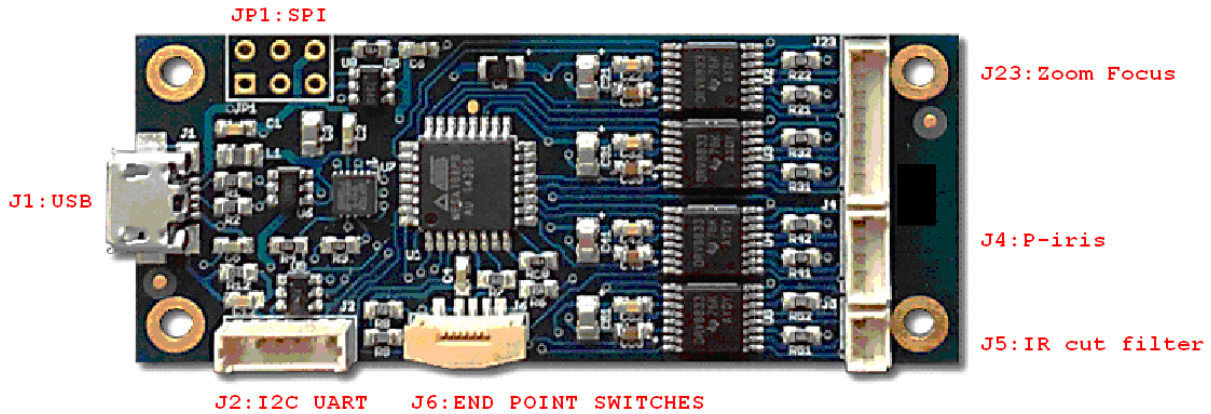
3.2.3 Operating parameters:

Operating Voltage	5.0V (±0.5V)
Operating Current (while stepping one motor at a time)	<250mA
Operating Current (while idle)	<50mA
In-rush current protection	1000mA

Note: Power consumption depends on many parameters of the operating mode.

Theia lens motor controller instructions

3.3. Connectors



3.3.1. USB (J1)

Micro USB connector able to supply and control the board via a single cable. This connector interfaces with a FTDI® FT200 USB to I2C bridge chip and uses drivers available for that chip. That way this interface appears as a serial port under some major operating systems. General compatibility is USB 2.0 but please refer to the datasheet of that chip to get full information about its features and compatibility.

The I2C side of that bridge chip is also connected to the I2C lines accessible at connector J2. Configuration options accessible through software need to be used to decide which interface (USB or I2C) to use. By default, the USB interface is active.

The USB interface is protected by a Texas Instruments® TPD3S014-Q1 which limits the supply current and provides ESD protection. Please refer to the datasheet of that chip for the details. Additional protection and filtering comes from passive elements, the FT200 and the controller chip.

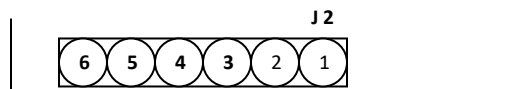
*/ FTDI is a trademark of Future Technology Devices International Ltd, Glasgow, United Kingdom

**/ Texas Instruments is a trademark of Texas Instruments Incorporated, Dallas, Texas, USA

3.3.2. I2C (J2)

J2 is a Molex® 53047-0610 or compatible connector. The fitting cable connector might be the Molex® 51021-0600 or compatible.

J2 has a total of 6 circuits (pins) of which the I2C interface uses two for its signals. To function the board would additionally need a ground reference (GND) and a 5V power supply. Both are available on J2 as well.



3 – I2C SCL

4 – I2C SDA

5 – Power GND

6 – Power +5 V

Theia lens motor controller instructions

The I2C signals have 4.7 kOhm pull-up resistors to 5V on board. The board will only interface with 5V I2C bus. If external logic level converter is used it must be I2C-safe. Connecting any electronics that attempt to drive the I2C pins high could result in damage to the control board. Speeds up to 400 kBaud are supported. ESD protection is provided to these signals by Littelfuse^{®**} SP724AHTG clamping diode arrays. Please see its datasheet for details.

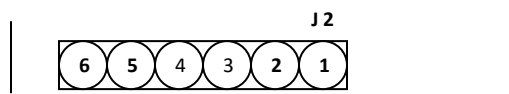
*/ Molex is a trademark of Molex LLC, USA

**/ Littelfuse is a trademark of Littelfuse Inc., USA

3.3.3. UART (J2)

J2 is a Molex^{®*} 53047-0610 or compatible connector. The fitting cable connector might be the Molex^{®*} 51021-0600 or compatible.

J2 has a total of 6 circuits (pins) of which the UART interface uses two for its signals. To function the board would additionally need a ground reference (GND) and a 5V power supply. Both are available on J2 as well.



1 – UART TxD

2 – UART RxD

5 – Power GND

6 – Power +5 V

The UART signals run at (5V CMOS) TTL levels at 19200 Baud and are 8N1 encoded. Please ask if you need that to be changed to something else. Similar boards with RS232 signal levels are available.

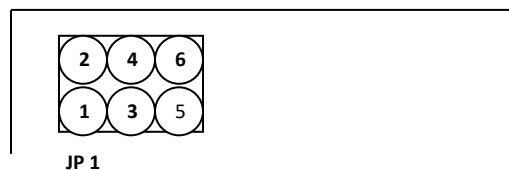
ESD protection is provided to these signals by Littelfuse^{®**} SP724AHTG clamping diode arrays. Please see its datasheet for details.

*/ Molex is a trademark of Molex LLC, USA

**/ Littelfuse is a trademark of Littelfuse Inc., USA

3.3.4. SPI (JP1)

JP1 is 3x2 pin header with 2.54 mm (0.1”) spacing. This header is normally used for the initial firmware upload to the controller but is not available for normal communication with the board. If communication by SPI is required contact us to discuss a custom firmware update.



1 – SPI MISO

2 – Power +5 V

3 – SPI SCK

4 – SPI MOSI

5 – Reset

6 – Power GND

ESD protection is provided to these signals by Littelfuse^{®*} SP724AHTG clamping diode arrays. Please see its datasheet for details.

Theia lens motor controller instructions

3.3.5. Reset (JP1)

Pin 5 of JP1 (see above) is an active-low reset input to the board. It has a pull-up resistor of 4.7 kOhm. ESD protection is provided to it by a Littelfuse®* SP724AHTG clamping diode array. Please see its datasheet for details.

*/ Littelfuse is a trademark of Littelfuse Inc., USA

3.3.6 Factory default reset

The board can be reset to its factory default in case of lost communication. This will reset all EEPROM data (speeds and steps), reset the communication to be USB/I2C client, and reset the I2C address to the 0x10 default.

Steps for doing the factory default:

1. Disconnect USB and serial cables from the board. Only motor cables may be connected.
 2. Set a jumper between Pin 1 and Pin 3 of the SPI connector JP1.
 3. Power the board via USB or 5V on the I2C/UART connector J2
 4. After 1 minute, power cycle the board
 5. EEPROM should now be reset to factory default settings. Reset the board to I2C server mode or UART mode if applicable and restore normal cable connections.
-

3.3.7. General Purpose I/O

Some Pins on some connectors could be reprogrammed into GPIO if certain interfaces are not needed. They could then be used to drive certain periphery like LEDs or read input. Please ask if such special features are needed.

3.3.8. Lens Connectors (J23, J4, J5, J6)

These connectors are intended for cables coming from the lens. More details can be found in the lens datasheet.

J23	Zoom and Focus stepper motor coil connections
J4	P-iris stepper motor coil connections
J5	IR cut filter coil connections
J6	Zoom and Focus end point switches

Those functions work independently from each other and are optional in that leaving any of these connections unpopulated will not affect the communication or behavior with the control board. For instance manual zoom/focus lenses with a P-iris may also be interfaced.

The end point switches are optional. The board needs a software setting to work with them – and would fail to work properly with that setting if the switches were missing. See the section above on checking the board configuration in the demonstration software.

3.3.9. Power Supply (J1, JP1, J2)

As mentioned above in the sections covering the connectors J1, JP1 and J2 there are several ways to supply power to the board. It is important to note that all the power pins of those connectors are directly connected – all three GND pins are connected and all three +5 Volt pins are connected (note that this includes USB).

DO NOT use more than one power supply at the same time because there are always small differences in their voltages and high cross currents would most probably run through the board – leading to energy loss, heat and may even end up in overheating and ignition.

The board is also not intended to be used for power distribution. It may be tempting to receive the power via USB and then serve it to periphery connected to one of the connectors mentioned here. However we discourage to use

Theia lens motor controller instructions

such a topology. The user would be responsible to check the current capability of the connectors and monitor the heat of the PCB traces under worst case conditions in order to verify how much energy can be distributed safely that way. He would also have to make the correct setting to the register telling the USB bus what the maximum power requirement is – adding the stepper motor current of the lens to that going to the additional periphery (see below). Depending on the device supplying energy to the USB bus that request may be denied if such a high current could not be supplied. That would disable the stepper motor operation.

The common +5 Volt supply line goes through a Texas Instruments®* TPD3S014-Q1 which limits the supply current to the board – both its maximum level and its rush-in rate on power-up. Please refer to the datasheet of that chip for the details.

*/ Texas Instruments is a trademark of Texas Instruments Incorporated, Dallas, Texas, USA

4. Software communication structure

4.1. Overview

A simple control software example is provided for testing and demonstration. That software is given “as is” in order to help with getting started and testing.

The user can further integrate communication with the board into your own program. In addition to basic stepping commands there are commands which set device parameters by writing values to a permanent memory on the board.

There is the firmware update mechanism. It may be activated during power-up and in that case a new firmware version can be uploaded, verified and saved to the device memory. Please contact us if there is any questions about firmware upgrades.

4.2. Different bus behaviors

With exception of the SPI bus the boards have already been used with different serial buses, namely USB, UART and I2C. In all these cases the commands to a board come from an external source and the board executes them and answers. But there are limitations depending on the type of bus used. The board may be unable to answer while a stepping command is being executed and/or the board may be unable to actually process any new command arriving while currently executing a stepping command. The later most notably disables the usability of the STOP command which would normally interrupt the execution of a current stepping sequence.

The current firmware supports one serial bus mode at a time. The default is USB which is accessed by the controller acting as a I2C client. The bus can be switched to I2C server mode or UART mode via serial command. SPI is not currently supported.

4.2.1 USB bus mode

The USB bus as it is currently handled on those boards include both limitations listed – unable to answer or process new commands while executing the current one. Commands arriving at such a time will remain in the 512 byte input buffer until they can be processed. That buffer may overflow so we recommend not to send new commands until execution of the previous command has finished. In the case of an overflow a partial command will most probably remain in the buffer. To clean this a byte containing “Carriage Return” (0x0D) needs to be sent.

If commands are being discarded (like a STOP command when no movement is currently executed) the board will not send the related confirmation. Since via USB any arriving command is being delayed until the end of any movement this will always happen so the STOP commands becomes useless and “invisible” via USB. Such behavior may also concern other commands.

Theia lens motor controller instructions

4.2.2 I2C server bus mode

The 7-bit address of the board in this mode is (hex) 0x10. This is the address for use with I2C libraries. It represents the upper 7 bits of the address. The least significant bit is the read/write flag so that address byte should be 0x20 (write) or 0x21 (read).

4.3. Serial Protocol

The current protocol is very simple. That makes it suitable for fast commands coming from low-power hardware. Each command starts with its ID and ends with carriage return CR(0x0D). The answers coming from the board also end with carriage return CR(0x0D).

The response buffer is not cleared after reading which may lead to confusion in some situations. When checking the buffer for completed motor movements, it is possible to send a 0x76 (get firmware version) command between movements and check the first byte of the response buffer. This overwriting with a different command is the best way to clear the buffer.

Theia lens motor controller instructions

4.3.1. Move Stepper Motor

Bytes:

Command ID	Motor ID	Steps (MSB)	Steps (LSB)	Start/Stop	Speed (MSB)	Speed (LSB)	CR
Command ID:	0x66	Rotate far/wide		count steps relative to current position			
	0x62	Rotate near/tele		count steps relative to current position			
	0x73	Rotate from PI stop		count steps absolute from the PI end point switch (zero) (only for motors 0x01 and 0x02 when PI switches enabled) (execution always starts with going back to the switch)			
Motor ID:	0x01	Focus					
	0x02	Zoom					
	0x03	Iris					
	0x04	IR-cut filter		not a stepper motor (see explanations below)			
Steps:	###	Number of steps to do:		(256 * MSB) + LSB For motors of type 0x01 this parameter works differently. Please see the explanations below.			
Start / Stop:	0x01	Start					
	0x00	Stop		set the number of steps to zero for this (Stop will not work on USB – no parsing while moving)			
Speed:	###	Steps per second:		(256 * MSB) + LSB Must be between min./max. speed of the motor setup. For motors of type 0x01 this parameter works differently. Please see the explanations below.			
CR:	0x0D	End of Command marker					

Answer after the command was executed successfully:

Bytes:

Answer ID	Status	CR
Answer ID:	0x74	End of Rotation
Status:	0x00	Success
CR:	0x0D	End of Answer marker

Note: The step count and speed parameters work differently for motors configured as DC motors (motor type 0x01). Please read the details on the next page. Motor number four (IR cut filter) is set as a DC motor by default.

4.3.1.1. Motion Control for DC motors

There is a command to write the setup of each motor (see below). There are stepper and DC motors. DC motors are powered for [Number of steps] times of a pulse width. The pulse width is set by 1 / [Steps per second]. Setting [Steps per second] to 500 gives a measuring unit of 2 ms and then setting [Number of steps] to 300 gives a single drive of 600 ms. [Steps per second] **must** be between min./max. speed of the motor setup.

Theia lens motor controller instructions

4.3.2. Read Firmware Version

Bytes:

Command ID	CR
------------	----

Command ID: 0x76 Read Firmware Version

CR: 0x0D End of Command marker

Answer after the command was executed successfully:

Bytes:

Answer ID	Version 1	Version 2	Version 3	Version 4	Version 5	CR
-----------	-----------	-----------	-----------	-----------	-----------	----

Answer ID: 0x76 Firmware Version

Version [1..5]: ### Values identifying the installed firmware version

CR: 0x0D End of Answer marker

4.3.3. Read Serial Number of the board

Bytes:

Command ID	CR
------------	----

Command ID: 0x79 Read Serial Number

CR: 0x0D End of Command marker

Answer after the command was executed successfully:

Bytes:

Answer ID	SN 1	SN 2	SN 3	SN 4	SN 5	SN 6	CR
-----------	------	------	------	------	------	------	----

Answer ID: 0x79 Serial Number

SN [1..6]: ### Values identifying the serial number of the board

CR: 0x0D End of Answer marker

Theia lens motor controller instructions

4.3.4. Read Motor Setup

Bytes:

Command ID	Motor ID	CR
Command ID:	0x67	Read Motor Setup
Motor ID:	0x01	Focus
	0x02	Zoom
	0x03	Iris
	0x04	IR-cut filter
CR:	0x0D	End of Command marker

Answer after the command was executed successfully:

Bytes:

Answer ID	Motor ID	Motor type	Use left stop	Use right stop	Steps (MSB)	Steps (LSB)	Min speed (MSB)	Min speed (LSB)	Max speed (MSB)	Max speed (LSB)	CR
Answer ID:	0x67	Motor Setup									
Motor ID:	0x01	Focus									
	0x02	Zoom									
	0x03	Iris									
	0x04	IR-cut filter									
Motor Type:	0x00	Stepper motor									
	0x01	DC motor									
Use left stop:	0x00	No									
	0x01	Yes				wide/far stops					
Use right stop:	0x00	No									
	0x01	Yes				tele/near stops					
Steps:	###	Max. number of steps:				(256 * MSB) + LSB					
						(The board does NOT keep track of the position)					
Min. Speed:	###	Min. steps per second:				(256 * MSB) + LSB					
Max. Speed:	###	Max. steps per second:				(256 * MSB) + LSB					
CR:	0x0D	End of Answer marker									

Note: For an invalid motor ID the return will be:

0x67 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0xFF 0x0D

Theia lens motor controller instructions

4.3.5. Write Motor Setup

Bytes:

Command ID	Motor ID	Motor type	Use left stop	Use right stop	Steps (MSB)	Steps (LSB)	Min speed (MSB)	Min speed (LSB)	Max speed (MSB)	Max speed (LSB)	CR
------------	----------	------------	---------------	----------------	-------------	-------------	-----------------	-----------------	-----------------	-----------------	----

Command ID: 0x63 Write Motor Setup

Motor ID: 0x01 Focus
 0x02 Zoom
 0x03 Iris
 0x04 IR-cut filter

Motor Type: 0x00 Stepper motor use this for motors number one to three (Focus, Zoom, Iris)
 0x01 DC motor use this for motor number four (IR cut filter)

Use left stop: 0x00 No
 0x01 Yes wide/far stops

Use right stop: 0x00 No
 0x01 Yes tele/near stops

Steps: ### Max. number of steps: $(256 * \text{MSB}) + \text{LSB}$
 (The board does NOT keep track of the position)

Min. Speed: ### Min. steps per second: $(256 * \text{MSB}) + \text{LSB}$

Max. Speed: ### Max. steps per second: $(256 * \text{MSB}) + \text{LSB}$

CR: 0x0D End of Command marker

Answer after the command was executed successfully:

Bytes:

Answer ID	Status	CR
-----------	--------	----

Answer ID: 0x63 Motor Setup Written

Status: 0x00 Success
 0x01 Wrong Motor ID

CR: 0x0D End of Answer marker

Theia lens motor controller instructions

4.3.6: Switch active bus

Bytes:

Command ID	New bus ID	CR
Command ID:	0x6B	Change active bus
New bus ID:	0x00	I2C server
	0x01	USB (I2C client) factory default
	0x02	UART
CR:	0x0D	End of Command marker

The command will not return any answer but will write the new bus type to EEPROM (semi-permanently) and then execute a controller reboot. Any communication after reboot will need to occur over the new bus type.

4.3.7: Change i2c address

Bytes:

Command ID	I2c address	CR
Command ID:	0x41	Change i2c address
I2c address:	0x##	New I2C address range 0x01~0x7F (1~127)
CR:	0x0D	End of Command marker

The command will not return any answer but will write the new address to EEPROM (semi-permanently). The new address will be active after a reboot of the motor controller. It is possible to issue a 0x68 command (switch active bus) with the currently active bus to reboot the controller.

Note: I2C uses the upper 7 bits of a byte for the address and inserts a Read/Write flag into the lowest bit. This may lead to confusion of the address sometimes. The address being programmed is the 7-bit value. For example, executing a <0x41 0x10 0x0D> command will set the I2C address to 0x10. Behind the scenes, an I2C library will issue a 0x20 command for reading and 0x21 command for reading.

Revision list, see document comment