

miniE shield

mini*Engine* - an open-source motion control
software for timelapse photography

Documentation of the
miniE shield v1.3

Airic Lenz

Content

1 Introduction	1
2 General principle	1
3 Assembly	3
3.1 Camera port (CAM)	3
3.2 EasyDriver port (EASY-DRIVER)	4
3.3 Limit-switch port (LSW)	5
3.4 RTC Module port (RTC MODULE)	6
3.5 Alternative camera port (IR)	6
3.6 Alternative motor driver port (MOTOR-RAW)	6
4 Appendix	7
Software License	8
Hardware License	8

1 | Introduction

The miniE shield is designed as an Arduino shield that holds all required hardware components to sport the *miniEngine motion-control-system for timelapse photography*. The following components can be placed:

1x Easydriver v4.4

1x Sparkfun RTC Module (DS1307)

1x Camera connector (2.5mm headphone jack + Resistors + Optocouplers)

1x generic 2.45mm-pitch port delivering the raw Arduino signals for the Camera

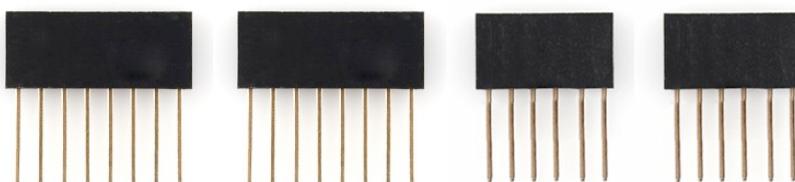
1x Motor connector (generic 2.45mm-pitch port)

1x Motor connector (4P4C jack)

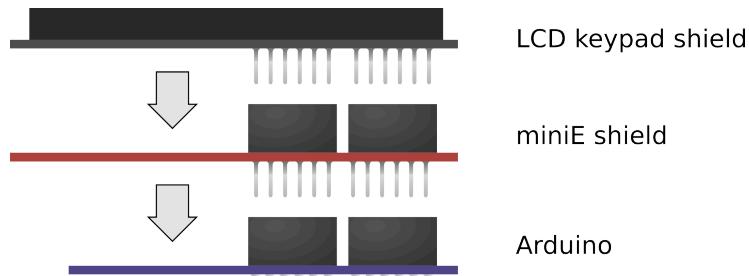
1x Limit switch connector (generic 2.45mm-pitch port + Resistors)

2 | General principle

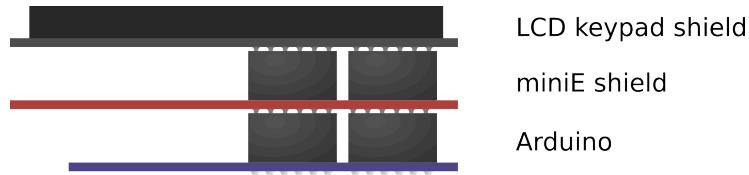
The miniE shield is designed to sit between the Arduino and the LCD keypad shield. To be able to be connected properly, it uses stackable pin headers like shown here:



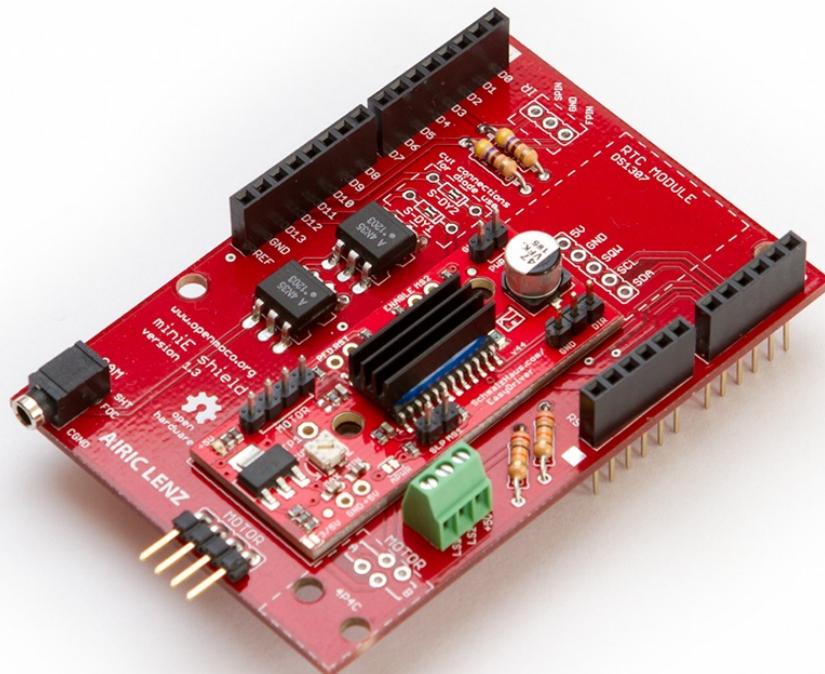
The boards are then stacked on top of each other with the miniE shield in the middle layer.



After putting the 3 boards together, the stack should look like this when seen from the side:



The miniE shield assembled with the most basic parts and with stackable pin headers:



| 3 | Assembly

The following chapter explains which components need to be assembled in which way.

| 3.1 | Camera port (CAM)

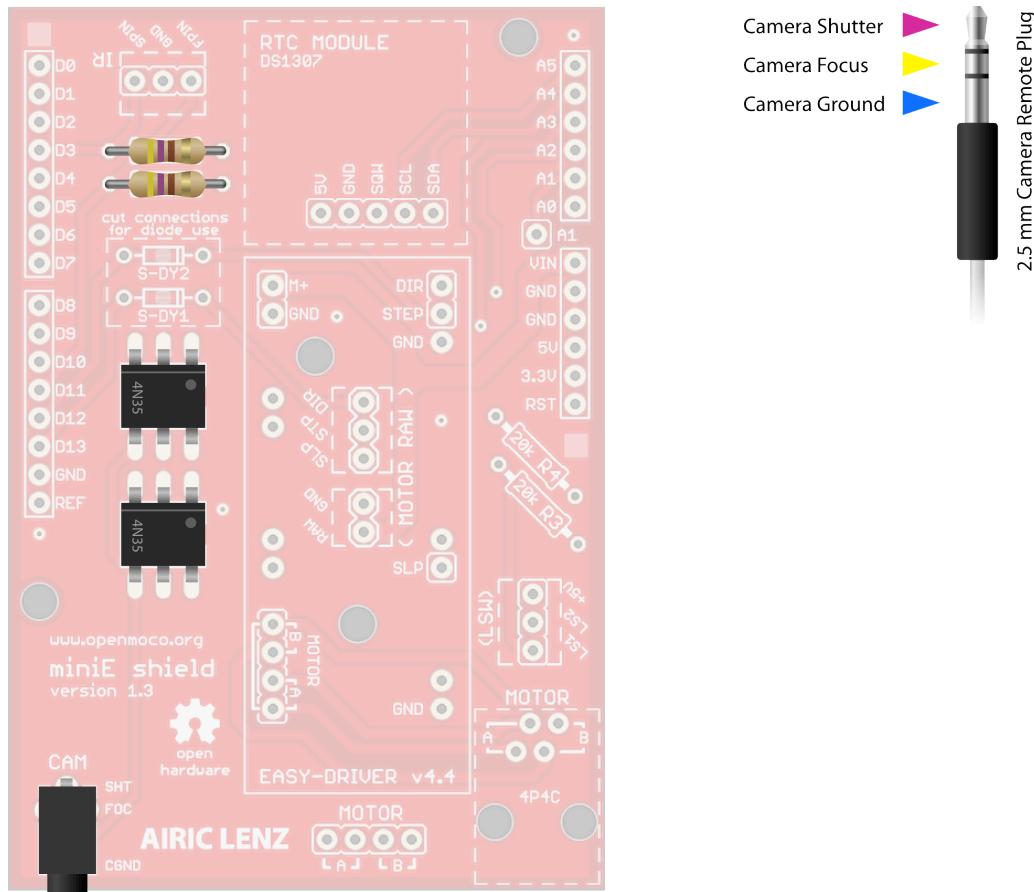
For the camera connector 5 parts have to be assembled.

2x Optocoupler (OK1, OK2)

2x Resistor 470 Ω (R1, R2)

1x Camera connector (2.5mm headphone jack - DigiKey PartNo.: CP1-2503A-ND)

The polarity of the resistors doe not matter. The polarity of the optocouplers does! Place it so that the pin 1 is placed where the white dot marks the position of pin one on the board. Pin 1 usually marked with a white dot on the devices too. The following picture shows the assembled components:



| 3.2 | EasyDriver port (EASY-DRIVER)

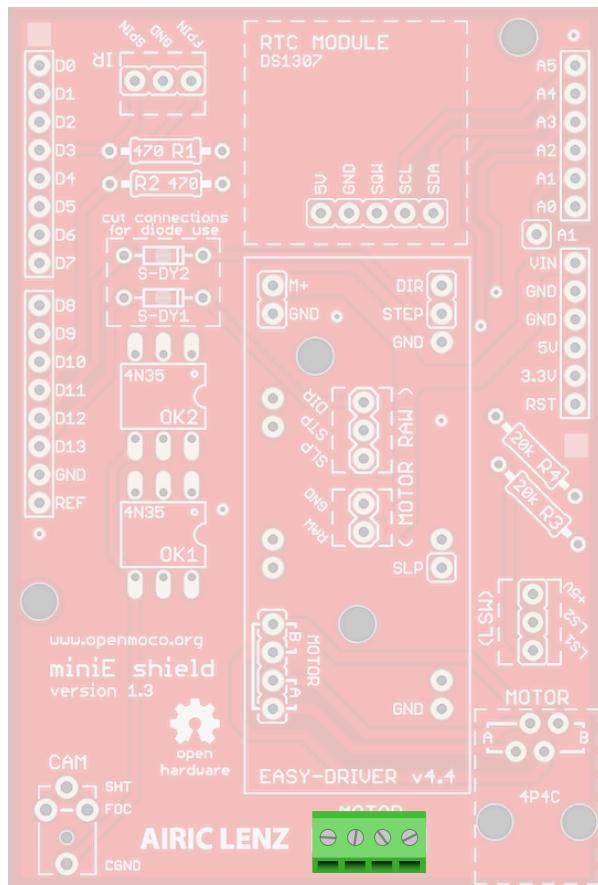
The miniE shield provides space for placing a stepper motor driver. More precisely - it has space for an EasyDriver v4.4. There is only one way the EasyDriver can be assembled. Anyhow - not all pins of the EasyDriver need to be connected. The pins that need to be connected are framed with a white line (M+, GND, MOTOR, DIR, STEP, SLP).

Male pin headers (2.54mm pitch) can be used to create the connection between the miniE shield and the EasyDriver. Solder them first to the one of the both components (place the pins as straight as possible) and then plug both components together - and of course solder the connections.

1x EasyDriver v4.4 (EASY-DRIVER)

1x Motor connector (MOTOR)

There are two possibilities to connect a motor to the board. The first one is to use a modular 4P4C jack (MOTOR port marked with 4P4C; DigiKey PartNo: AE10381-ND). The second variant is to use any connector you want that has a 2.54mm pitch (4x1). To go this way, assemble your connector to the port named MOTOR:



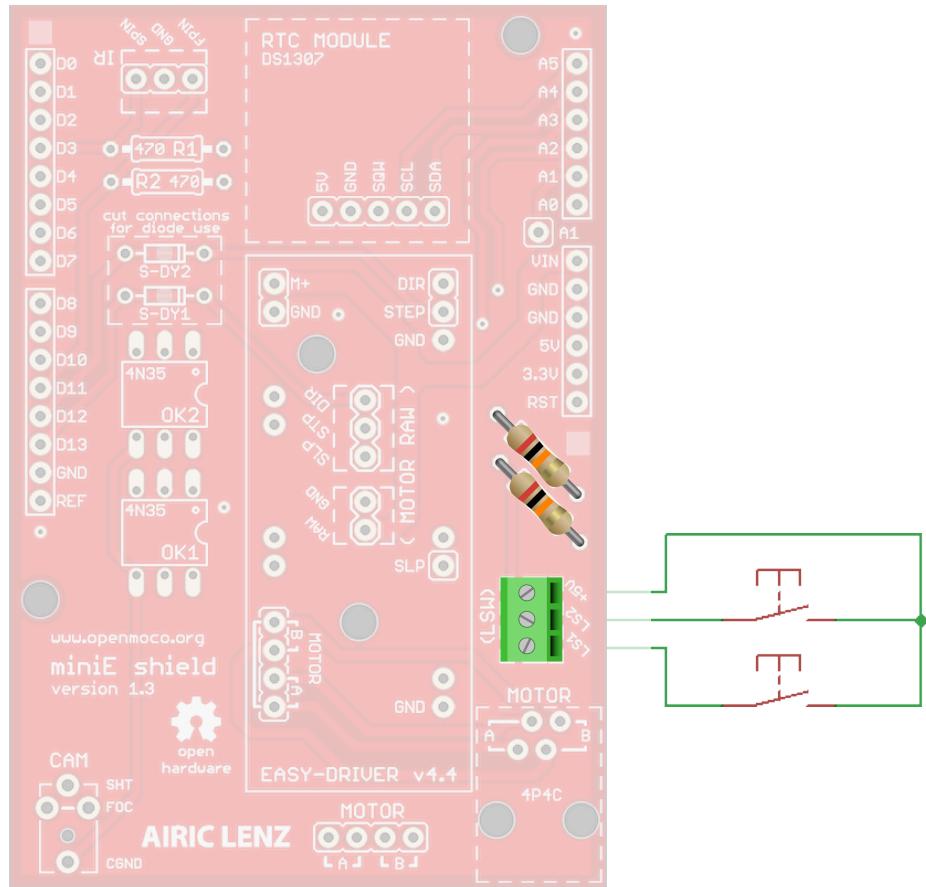
| 3.3 | Limit-switch port (LSW)

To connect limit switches to the board you'll need the following components:

2x Resistor 10~20 kΩ (R3, R4)

1x Limit switch connector (LSW)

For the connector you can use any connector that has a 2.54mm pitch (3x1):



You can then connect 2 switches (default state off) so that each switch is connected on one side either to LS1 or LS2 and on the other side to +5V.

| 3.4 | RTC Module port (RTC MODULE)

The miniE shield provides space for placing a real-time-clock module. This module has an own power source and is so able to store the current time.

1x RTC Module (RTC MODULE - SparkFun RTC module: BOB-00099)

Please place the module so that the battery is on its top side and thus stays reachable. You can use the same technique (male pin headers) for connecting this module as described for the motor driver.

| 3.5 | Alternative camera port (IR)

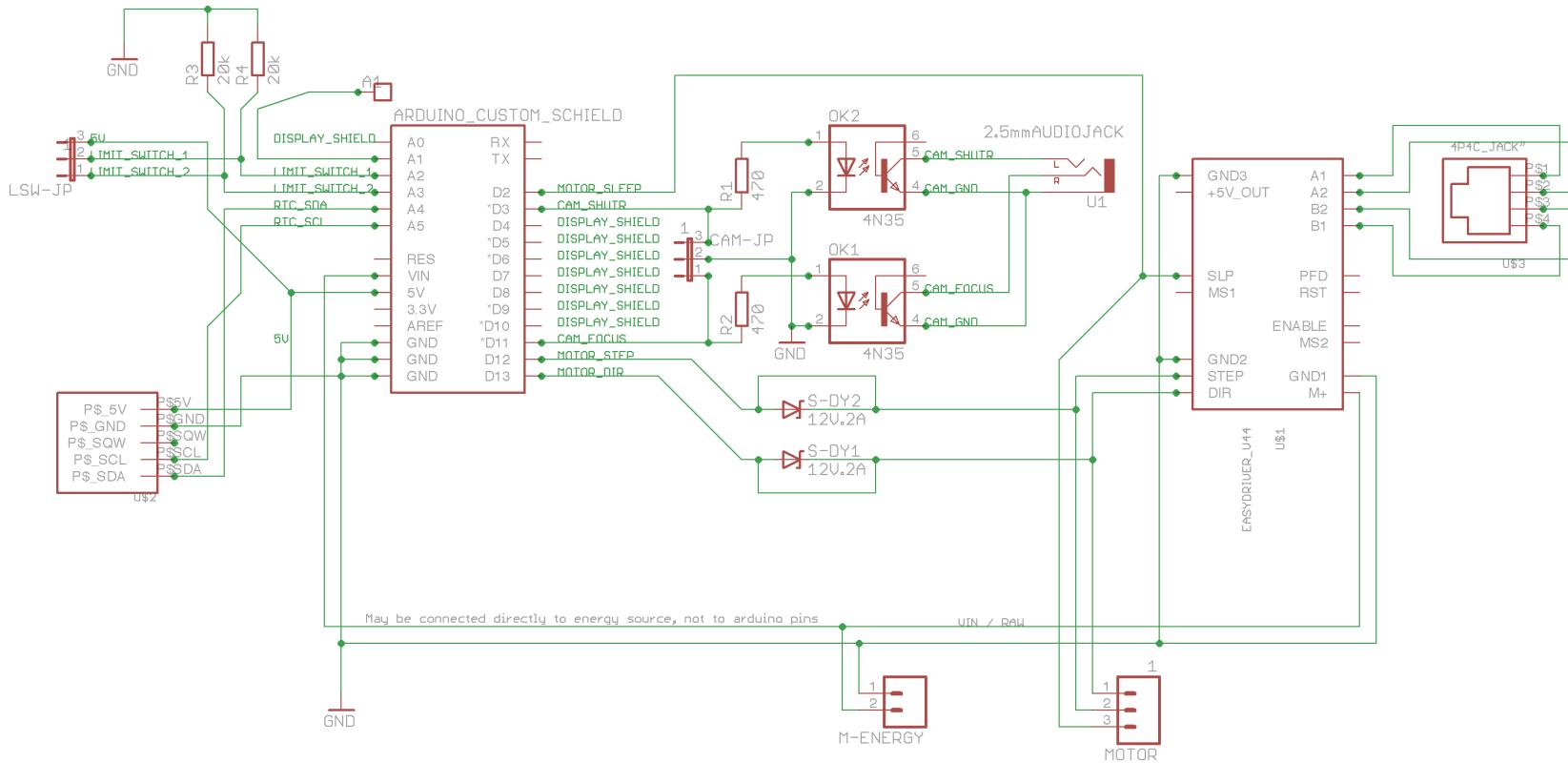
The IR named port delivers the pure Arduino signals used for the Camera focus and for the Camera shutter line. It also provides the general Ground bus. This port can be used to implement alternative trigger methods - other than the remote-cable variant. One example would be triggering the camera via infrared. The port has a generic 2.54mm pitch (3x1).

| 3.6 | Alternative motor driver port (MOTOR-Raw)

The MOTOR-RAW named port delivers all the pure Arduino signals and power busses used for controlling and powering the motor driver. This port can be used to connect to another stepper driver in case you don't want to use the EasyDriver. The port has 2 generic 2.54mm pitched ports (3x1 & 2x1).

APPENDIX 1 - Circuit diagram

7



Software License

The miniEngine software is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

The miniEngine software and all belonging material is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Hardware License

All hardware designs and schematics provided along with the miniE software are free. You can redistribute it and/or modify it under the terms of the CERN Open Hardware Licence 1.1, or (at your option) any later version.

The hardware documentation and hardware files are distributed in the hope that they will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the CERN Open Hardware Licence for more details.

You should have received a copy of the CERN Open Hardware Licence along with these files. If not, see <http://www.ohwr.org/projects/cernohl/wiki/>.