

Interlock system for a high power laser

David Hälg

Supervisor: Christoph Fischer

Professor: Dr. Jonathan Home

Department of Physics, ETH Zürich

November 27, 2016

Abstract

In this semester thesis the setting up of an interlock system for a high power solid-state laser at 1064 nm in the laboratory of Prof. Dr. Jonathan Home is described. This system prevents valuable laboratory equipment from being damaged through failure of the cooling water circuit and also provides safety for people working with or near the laser. Furthermore this semester project gives insight into the different parts of the interlock system and how it can be expanded with further functionalities.

Contents

1	Introduction	3
2	Implementation	4
2.1	Power supply	5
2.2	Temperature interlock	6
2.3	Cooling cycle	6
2.4	Indicator for the laser being switched on	7
2.5	Safety switch	8
2.6	Software	8
2.6.1	Arduino	8
2.6.2	Microprocessor	9
2.7	Known Issues	10
3	Conclusion	11
4	Appendix	12

1 Introduction

The goal of this semester thesis in the group of Prof. Home was to build an interlock system for a high power solid-state laser with 50 W output power at 1064 nm. Like any other interlock system the one built during this thesis is designed to shut down the laser as well as its cooling circuit in the event of malfunction in order to protect valuable equipment from severe damage. Furthermore, given the high output power of the laser, the interlock is meant to prevent misuse by untrained users.

Considering possible worst case scenarios the following situations in which the interlock has to be tripped come up:

- First of all the laser has to be shut down in case of it overheating.
- The laser should also be shut down in the instance of an user who is unfamiliar with the high power beam path, opening the laser's safety enclosure.
- Finally, the laser as well as its cooling water circuit have to be shut down immediately in case of a leakage to prevent water from spilling on the optical table and damaging the equipment.

These functionalities were realized in the interlock. Additionally two key switches were added to bypass the interlock temporarily. One can be used to do maintenance work while the laser is running and the other disables the temperature monitoring. This feature is needed for starting the laser because the temperature will overshoot and would trip the interlock before stabilizing to the set temperature.

In the following chapters the implementation of this interlock system will be described in detail, known issues will be discussed and in the end possible extensions to the system will be presented.

A special thanks goes to Prof. Dr. Jonathan Home for giving me the chance to work in his group, to Christoph Fischer for supervising this semester project as well as helping me all the way with even the smallest problems and also to the whole TIQI group for the great help and atmosphere.

2 Implementation

In this chapter every part of the interlock as well as their interaction will be explained in detail. The interlock was implemented on a home-built breakout board for a complex programmable logic device (CPLD¹) that was designed by Ben Keitch. The board is equipped with several BNC connectors, two relays controlled by the microprocessor and a digital communication interface to an Arduino Uno [1], mounted underneath the board. The CPLD board itself is described on the TIQI-Wiki [2]. A list of all inputs and outputs of the CPLD board is given in table 1 in the appendix. How the temperature relays on the CPLD board have to be connected to the laser can be seen in figure 1.

All the electronics including the power supply were put into a small box in order to minimize the space taken by the interlock. The front panel of the box was made by the ETH machine shop and has through holes for the BNC connectors and LEDs on the CPLD board. As it can be seen in Figure 2, only two of the BNC connectors are used to indicate that the laser is running on a doorsign at the entrance of the laboratory.

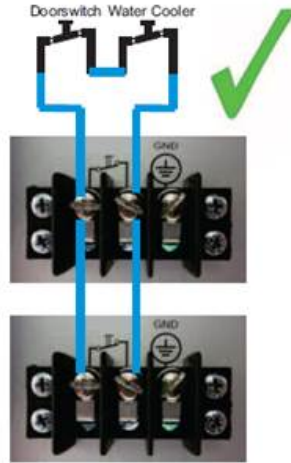


Figure 1: A graphic of how the temperature interlock is connected to the laser. Note that no doorswitch is built in this interlock. [4]

¹Xilinx CoolRunner-II

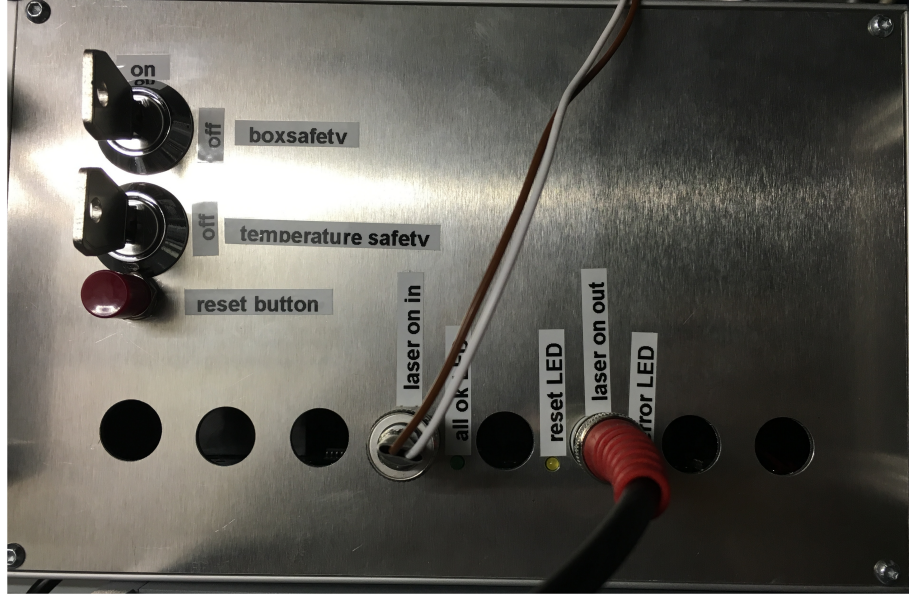


Figure 2: The front panel of the interlock system.

2.1 Power supply

The interlock system is powered by a IHTAA-16W power supply from International Power[3]. It provides a direct current of 5 V with up to 2 A in the first channel, 12 to 15 V with up to 0.4 A in the second channel and -12 to -15 V with up to 0.4 A in the third channel. Using these channels it supplies the solenoid valve with a 24 V , the CPLD with 12 V and the flow meter with a 5 V current. It is connected to the normal 220 V socket.

The power supply was assembled according to the Assembly Instructions on the TIQI server ². All the outputs have labels, stating the output voltage. The two grounds need to be connected in order to ensure a working interlock. Furthermore it is important to notice that even though the power supply has a switch to turn it off it will be turned on all the time for practical reasons. This means that as soon as one connects the power supply to

²[smb://windata.phys.ethz.ch/tiqi/Projects/Electronics/PSUs/PSUcommercial/AssemblyInstructions\(PSU\).pdf](smb://windata.phys.ethz.ch/tiqi/Projects/Electronics/PSUs/PSUcommercial/AssemblyInstructions(PSU).pdf)

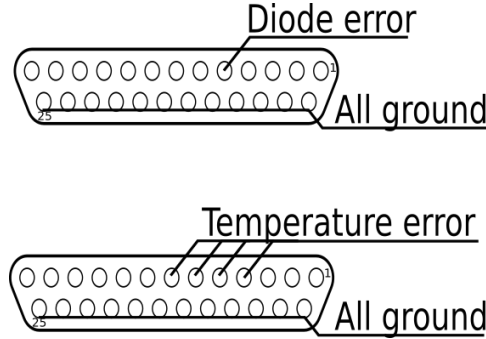


Figure 3: Top: the Master Laser Diagnostics Connector. Bottom: the Amplifier Diagnostics Connector.

the socket all the outputs will be active.

The power supply has been checked and approved by the security responsible.

2.2 Temperature interlock

For not burning through the laser in case of overheating the temperature interlock was built in. In case of overheating, this system has to shut down the laser.

The system reads in the temperature error signal ('temperature error LD') from the amplifier's diagnostics connector and the temperature error signal of the master laser diode diagnostics connector ('TEC error signal'). All of these signals are connected to the Arduino through a D-Sub connector at the back of the interlock box and give a signal of 10 V per degree Celsius that the actual temperature differs from the setpoint. One can see the connections in figure 3. The error signals are routed to the analog inputs of 'A1' to 'A5' of the Arduino. The order in which they are connected is not important as they are treated equally.

2.3 Cooling cycle

The main reason the interlock was built is that any leakage in the cooling cycle could be immediately detected and that anything getting damaged through this could be prevented. For this the cooling cycle interlock was

designed.

The laser is connected to the main cooling water cycle of the HPF building. In case of the pipes leaking, the laser has to be shut down immediately and the flow of the coolant has to be stopped. The water flow also has to be interrupted if there is a power cut.

The hardware required to implement the cooling cycle interlock are a flow meter and a magnetic valve. The flow meter is a custom ordered ultrasonic flow meter of the 100 series from flownetix [5]. It is powered by the internal 5 V power supply of the interlock. The flow meter measures the water flow through the outlet of the cooling cycle and outputs a signal proportional to the flux. It has been tested that this signal changes rapidly with the slightest anomaly in the water cycle, like for example a small leak. The output of the flow meter is connected to the Arduino via the analog pin A0. If the Arduino registers a leak, the laser and the water cycle is shut down through the microprocessor on the CPLD board.

The water flow is stopped by a magnetic valve that operates with 24 V provided by the power supply. It is built so that it only opens if the voltage is applied, so it is normally closed without power. This ensures that also in the event of a power cut no water can leak out. The valve is connected to the second relay on the CPLD board.

2.4 Indicator for the laser being switched on

For laser safety reasons it is important to know which laser is switched on and running before entering the lab. For this reason a sign was installed at the entrance of the laboratory by Frieder Lindenfelser on which LEDs indicate which lasers are currently switched on. This was implemented using a photodiode which is measuring residual transmission through the first mirror of the optical setup, only present when the laser is switched on.

The photodiode was installed so that it outputs 4.65 V to avoid that the CPLD board receives more than 5 V. The signal from the diode then gets processed by the CPLD board and if the laser is running, the door sign LED is switched on over the other BNC connector. The cables of the photodiode are white/brown and run out of the laser enclosure together with the ones of the safety switch (see below). The cable to the door sign LED runs through the cable channel above the experimental table.

In the future a less expensive photodiode could be substituted for this one, since the application of this diode does not need high precision or a fast reaction time.

2.5 Safety switch

A micro switch was installed underneath the lid of the laser enclosure. If the lid of the enclosure is opened accidentally while the laser is running, the interlock immediately switches off the laser.

As one wants to open the box for maintenance work during which the laser has to be running, a key switch to turn this function off has been installed. One can see the key on the front panel displayed in Figure 2.

2.6 Software

Software-wise the interlock system is divided in two parts: the microprocessor which has software written in Verilog HDL and the Arduino which has its own software language.

The code for both, the Arduino ³ and for the microprocessor ⁴, can be found on the group network drive. It has been initially written by Christoph Fischer [6] and has been modified by Vlad Negnevitsky.

2.6.1 Arduino

The Arduino is used for readout of analog sensors (which the microprocessor cannot) while the microprocessor is in charge of the shut-down mechanism. Additionally, the Arduino also processes the analog signals from the flow meter and the temperature sensor. As soon as the Arduino measures a non-zero input from the temperature error connectors, corresponding to a too high or too low temperature of the laser, that does not change for approximately half a second, the Arduino triggers the temperature interlock by sending a signal to the microprocessor using one of the digital communication lines. In the same way the Arduino measures the coolant flow and triggers the flow interlock if necessary. It is important to note that for this

³`smb://windata.phys.ethz.ch/tiqi/Git/electronics/christoph_interlock/Arduino/interlock_segtrap`

⁴`smb://windata.phys.ethz.ch/tiqi/Git/electronics/christoph_interlock/CPLD`

the Arduino program compares the input to reference values that have to be set correctly beforehand.

Another functionality implemented in the Arduino code, as well as in the CPLD code is the watchdog timer [7]. This ensures that if the Arduino stops working, the interlock is triggered and if the CPLD board stops working a buzzer connected to the Arduino should get switched on. But since there is no buzzer connected to the Arduino at the moment, this functionality is not used for now. The buzzer would need to be connected to pin 11 on the Arduino.

After the interlock has been triggered the Arduino and the CPLD remain in the locked state until they are manually reset. The reset is performed by pressing the 'reset' button on the front panel. This resets the microprocessor and if the microprocessor is reset, the Arduino automatically gets reset too through a digital communication line to the microprocessor.

2.6.2 Microprocessor

The microprocessor is controlling if the laser is shut down or can continue working. The decision if it can continue is made based on the input from the Arduino, as well as the state of the switches connected to it. If the Arduino sends the signal that there is a problem with the cooling system, the microprocessor shuts everything down immediately. There is no possibility to bypass this behaviour. If the Arduino sends the signal that there is a problem with the temperature of the laser, the microprocessor will ignore this if the input if the temperature interlock has been disabled using the key switch. Otherwise it will shut down the laser, but the cooling will continue.

The microprocessor will also shut everything down if the Arduino is not responding, i.e. stops sending a pulsed signal. Furthermore there is a micro switch in the box which will, if opened, shut down the laser, unless the key on the front panel saying 'boxsafety' is turned off or the laser is not running.

The microprocessor is also forwarding the information if the laser is on from the photodiode to the LED panel outside of the lab. The diode and the door sign LED are connected via the BNC connectors on the front panel. The push button on the front panel resets the microprocessor if pressed. The microprocessor is programmed so that it forwards the reset signal to the Arduino, which is then reset as well.

The LEDs on the front panel are also controlled by the microprocessor. There is the green LED ('all ok LED') that lights up when everything is working fine, the yellow LED ('reset LED') that lights up when the reset button is being pressed, as well as the red LED ('error LED') that lights up when the interlock has been tripped. Note that the red LED on the board used for this particular interlock seems to have a bad connection and therefore rather look at the green LED for troubleshooting (it is off when something is wrong).

2.7 Known Issues

There are still some issues and things that are good to know related to the interlock system, most of them regarding testing and setting up the interlock

Problems with the setup:

- The biggest problem at the end of the semester project was that the cooling water suddenly was too cold. This led to a heatsink error on the amplifier and the temperature guard of the laser kicked in. The use of an intermediate heat exchanger has solved this problem in the meantime.
- The red LED of the CPLD board is sometimes not working and it seems that the contact to the board is not perfect. Thanks to the green LED which is the inverse of the red LED, this is not really a problem.

Things that are good to know when testing and setting up the interlock:

- When one wants to test the microprocessor program in the Xilinx ISE Design Suite [8], then one should set the 'reset counter' variable (which ensures that the Arduino, which is operating at a lower frequency, also gets reset) in the program to a very low value, in order to not just see the reset.
- Also in the Xilinx ISE Design Suite, one should set the unused pins to ground or pullup, but not to floating. Otherwise one can get unexpected behaviour of the microprocessor.
- When programming the Arduino for use in the interlock, one has to leave out the debug code. Otherwise the Arduinos serial output used

for debugging will interfere with the microprocessor, triggering the interlock.

- One should also note that in the power supply the two grounds are not connected. Therefore there is a short cable connecting the two. If they are not connected, the interlock is triggered because the reference ground of the Arduino will be floating with respect to the ground of the flow meter.

3 Conclusion

In this semester project a working interlock system has been built for the high power solid-state laser at 1064 nm. It has been shown to work if the cooling water is at an appropriate temperature. New functionality was added, but there are also enough BNC connectors as well as Arduino pins free to add new functions. One idea would be to add a key switch to turn the flow interlock off to avoid it triggering when one is testing the cooling cycle. Furthermore one could add a buzzer to the Arduino to indicate that the microprocessor stopped responding, which people working in the laboratory could notice.

Furthermore, one could also exchange the photodiode. The current one is equipped with a low noise high gain transimpedance amplifier, which is not needed in this setup. Therefore the current photodiode could be interchanged for a cheaper and maybe smaller one.

4 Appendix

Inputs Arduino		
	flow meter temperature error diode temperature error T1 temperature error T2 temperature error T3 temperature error T4	pin A0 pin A1 pin A2 pin A3 pin A4 pin A5
Inputs microprocessor		
	reset button box safety key switch over-temperature key switch box open switch photodiode	greenbtn redbtn amberbtn bluebtn BNC CONN500
Outputs microprocessor		
	LED outside lab error LED reset LED ok LED temperature interlock flow interlock	BNC CONN303 redled amberled greenled relay 1 K300 relay 2 K301

Table 1: The inputs and outputs of the CPLD board.

References

- [1] Arduino LLC, *Arduino homepage* [Online], October 2016, <https://www.arduino.cc>.
- [2] C. Fischer, *cpld_board* [Online], December 2015, http://wiki.phys.ethz.ch/TiqiWiki/cpld_board.
- [3] Mouser electronics, *International Power IHTAA-16W power supply* [Online], October 2016, <http://www.mouser.ch/ProductDetail/International-Power/IHTAA-16W/?qs=%2fha2pyFaduj3G7lDHcrJGBA5dtwb4pnn2gY6K8pz1as%3d>.
- [4] Coherent Inc. *Mephisto MOPA Laser Operator's Manual*.
- [5] Flownetix Limited, *Flownetix homepage* [Online], October 2016, <http://www.flownetix.co.uk>.
- [6] C. Fischer, *Optical trapping in a build-up cavity* [Online], 30. April 2015, https://www.ethz.ch/content/dam/ethz/special-interest/phys/quantum-electronics/tiqi-dam/documents/masters_theses/mastersthesis-christoph_fischer.pdf.
- [7] Wikipedia, *Watchdog timer* [online], August 2016, https://en.wikipedia.org/wiki/Watchdog_timer.
- [8] Xilinx ISE Design Suite [Online], September 2016, <https://www.xilinx.com/products/design-tools/ise-design-suite.html>.