Motor Control

CTIO 60 inches CHIRON

CHI60HF-8.1

La Serena, December 2010

Contents

Introduction

The present document is just a brief summary of the work done on the CHIRON motor control hardware. This document does not describe the upper level software (host) that handles the motors, but just the hardware involved, as well as the actual description of the box used to enclosure the drivers

The motors handled by the box described in this document are the ones corresponding to the FOCUS and SLICER mechanisms.

An old SUN hard drive box was modified to be used as the enclosure for both drivers. The advantages of using this box was that it had already a built-in 12 Volts supply, fan and power switch (on/off).

The driving mechanism for both motors is identical. See **Figure 0.1** The driver used was in both cases an AllMotion board, model EZSV23, that implements a servo mechanism using the encoder signal as the input for servo positioning. However, the board requires a non-differential signal for the encoder (A/B/GND), while the controlled motors both have differential encoder outputs (A+/A-, B+/B-). To adapt the encoder outputs we used an already-designed interface encoder driver board by Manuel Martinez. So, we just took that design and build it into our control chain.

The EZSV23 boars uses for commands/control ascii commands over an RS-485 line.
Due to the lack of serial ports, we decided to use a Ethernet-RS485 Terminal Server. The model chosen was the Lantronix EDS4100, that provides 4 serial outputs, 2 of which also support RS-485. So we configured the Terminal server to use both RS-485 ports, assigning one to each motor. In this way, both motors are independent (they do not share the same RS-485 bus)

Each control channel uses 2 DB connectors, a DB9 for the serial communication, and a DB15 for the motor communication. So, the back of the box was modified to install these 4 DB conectors

See **Appendix A** for the actual schematics, and **Appendix B** for some photographs of the box.

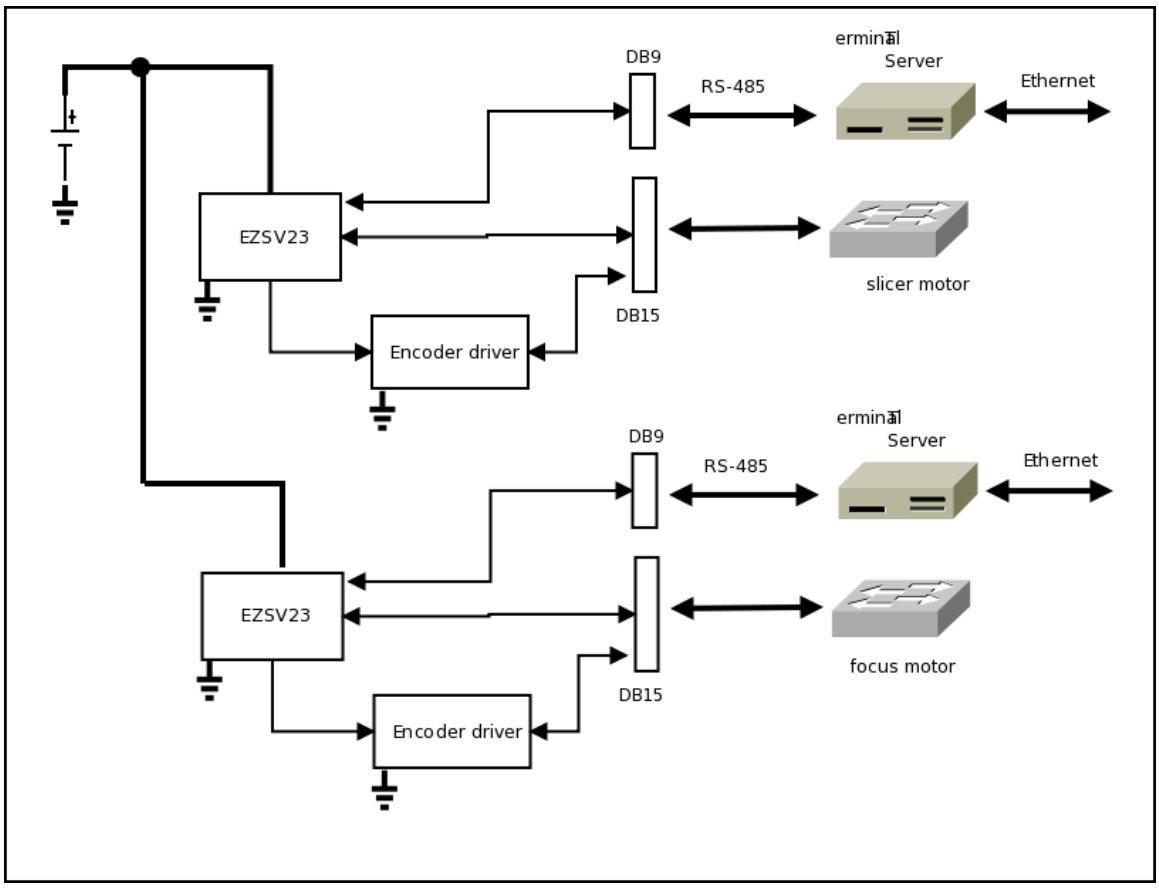


Figure 0.1: Internal Box scheme

1.- Image Slicer motor

The Image Slicer motor is a Physik Instrumente (PI) motor, model PI M-110.1DG. It is a high precision miniature translation stage, with 5 mm range and 0.1 um precision.
It uses an incremental encoder, and PWM to control the position and velocity.

The AllMotion EZSV23 has the direct output to control the movement (amplified PWM output), but the encoder needed to be interfaced from the motor to the board. For this we used Manuel Martinez's adapter board (CH8947.910-C36) that is based on the quadruple differential line receiver 26LS32.

The control board was powered with a 12 volts power supply -the board itself provides both the phase reversion for the motor (which is +-12 V) and the 5 V power that the motor encoder requires.

The pin assignment for the DB15 connector for the motor was chosen to match the connector of the PI M-110 motor (See **Appendix A, b)**)

The connection to the control board is RS-485, but in this case we used a Lantronix Terminal Server that allowed us to talk to it using tcp/ip, assigning one specific port for the slicer.

The actual schematic and pin assignment can be seen in **Appendix A, a)**
The Lantronix settings can be found on **Appendix C**

2. - Focus motor

The Focus motor is a Physik Instrumente (PI) motor, model PI M-605.1DD. It is a high precision translation stage, with 25 mm range and 0.1 um precision.
It uses an incremental encoder, and PWM to control the position and velocity.

The regular AllMotion EZSV23 board has amplified PWM output, but this stage has incorporated a PWM amplifier, that requires as input control SIGN and MAGNITUDE signal inputs. Due to this incompatibility, we asked AllMotion to modify a standard EZSV23 board to have as output SIGN/MAG instead of amplified Motor+/Motor-, but the modified board was unable to servo-control the motor. For this reason, the motor was opened, and the built-in PWM amplifier (a LM182201) was bypassed, allowing us to connect directly to the motor inputs. Fortunately there were two unused pins in the PI DB 15 motor connector, and we used those to pass the M +/- signals from the driver board. Since we skipped/bypassed all the active amplifier elements, the usage of the external 24V power supply for the motor is no longer required. See **Appendix A, c)** for indications of the modified Pinout of the DB motor connector and a summary of the motor modifications.

The encoder needed to be interfaced from the motor to the board in exactly the same fashion we did for the slicer motor (see point 1)

The control board was also powered with the 12 volts power supply available in the box. The driver board also supplies the 5 V required for the motor encoders.

The pin assignment for the DB15 connector for the motor was chosen to match the connector of the PI M-605 motor + the motor modification.

As with the slicer, we used the terminal server, assigning one specific port for the focus.
The actual schematic and pin assignment can be seen in **Appendix A, a)**
The Lantronix settings can be found on **Appendix C**

3.- Quick info on software configuration

3.1 Slicer configuration

The software module that controls the slicer is an SML device. For detailed information see the Software Description, CHI60S-X.

The quick summary is:
configuration file location: $HOME/apps/CHIRON/config
configuration file name: DEV_SLICER.cfg

In the file, there are three relevant section for this box:

[COMMS]
params="type tcp, address 139.229.12.39, port 10001"
motorID=1
s
pecifies the tpc/ip address of the Terminal Server, service port (always 10000 + tport, with tport = [1,4])
The address of the motor in the 485 bus. IN this case, 1 (the only device in this bus)

[CONVERSIONS]
mm2ticks=145635

States the conversion to apply to transform the "mm" into encoder counts ("ticks") and vice-versa. This is important to being able to specify the desired position in mm.
The specified number comes from
*=gear_ratio * [encoder_resolution (ticks/rev) / driver_screw_pitch (mm/rev)]*
=28.44444 * 2048 / 0.4 [ticks/mm]

[MISC]
inispeed=0.3 */*speed, in mm/s to find home*/*
maxspeed=1.37 */*maximum allowed speed, in mm/s*/*
workspeed=0.7 */*speed for regular use, in mm/s*/*
maxpos=5 */*maximum allowed absolute position, in mm*/*
zerooffset=0.0 */*distance, in mm, away from limit to be considered 0*/*
tolerance=0.001 */*maximum distance allowed away from requested pos., in mm*/*

3.1 Focus configuration

The software module that controls the slicer is an SML device. For detailed information see the Software Description, CHI60S-X.

The quick summary is:
configuration file location: $HOME/apps/CHIRON/config
configuration file name: DEV_FOCUS.cfg

In the file, there are three relevant section for this box:

[COMMS]
params="type tcp, address 139.229.12.39, port 10003"
motorID=1
s
pecifies the tpc/ip address of the Terminal Server, service port (always 10000 + tport, with tport = [1,4])
The address of the motor in the 485 bus. IN this case, 1 (the only device in this bus)

[CONVERSIONS]
mm2ticks=10000

States the conversion to apply to transform the "mm" into encoder counts ("ticks") and vice-versa. This is important to being able to specify the desired position in mm.
The specified number comes from
=gear_ratio * [encoder_resolution (ticks/rev) / driver_screw_pitch (mm/rev)]
=1 * 10000 / 1 [ticks/mm]

[MISC]
inispeed=0.3 */*speed, in mm/s to find home*/*
maxspeed=1.37 */*maximum allowed speed, in mm/s*/*
workspeed=0.7 */*speed for regular use, in mm/s*/*
maxpos=25 */*maximum allowed absolute position, in mm*/*
zerooffset=0.0 */*distance, in mm, away from limit to be considered 0*/*
tolerance=0.01 */*maximum distance allowed away from requested pos., in mm*/*

Acknowledgments

Thanks to Rodrigo Alvarez who did the actual job on modifying and mounting the box. Thanks to Manuel Martinez for suppling the encoder driver design and help with general questions.

References

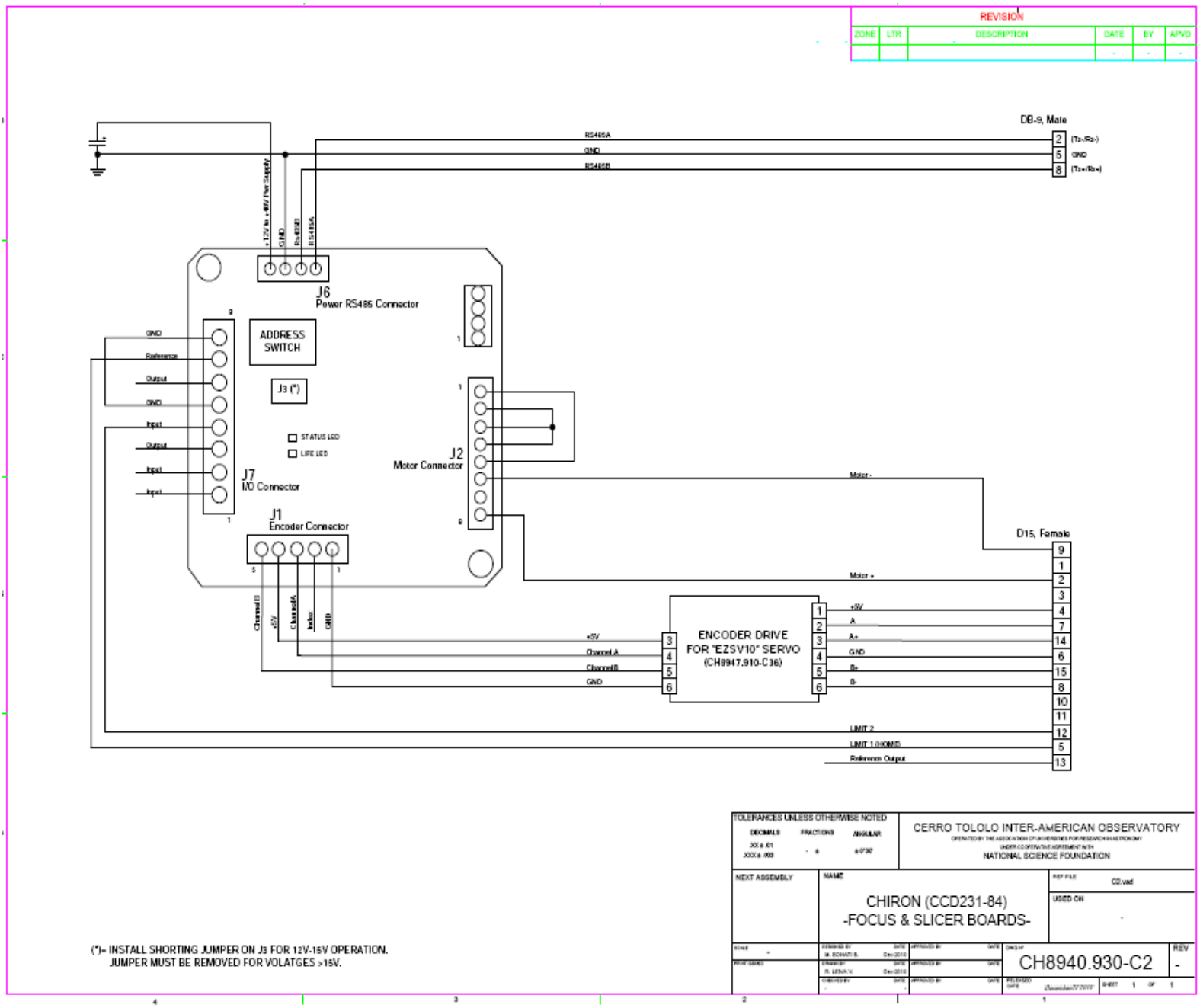
- *EDS4100 User Guide*
  Lantronix, www.lantronix.com

- *CH8940.930-C2 schematic*
  R.Leiva, Dic. 2010

- *EZSV23 board wiring*
  Allmotion, www.allmotion.com

Appendix A: final schematics

a) Slicer/Focus Schematic



b) PI-110 Pins

5.4.1 DC-Motor Versions (M-11x.xDG)

Connector type: Dsub15(m)

Pin #		Function	
1		internal use	n.c.
	9	Motor (-) input	
2		Motor (+) input	
	10	internal use	n.c.
3		internal use	n.c.
	11	internal use	n.c.
4		+5 V input	
	12	Limit Switch (negative side)	
5		Limit Switch (positive side)	
	13	~~Reference signal output~~	n.c.
6		GND (Limit Switch and Logic)	
	14	Encoder A (A(+) when using RS-422-type transmission)	
7		A(-) when using RS-422-type transmission	
	15	Encoder B (B(+) when using RS-422-type transmission)	
8		B(-) when using RS-422-type transmission	

c) PI-605 modified pins

Connector J2 (Signals, Controller connection)

Type: 15-pin sub-D connector
Reference No.: AMP #9-215594-1

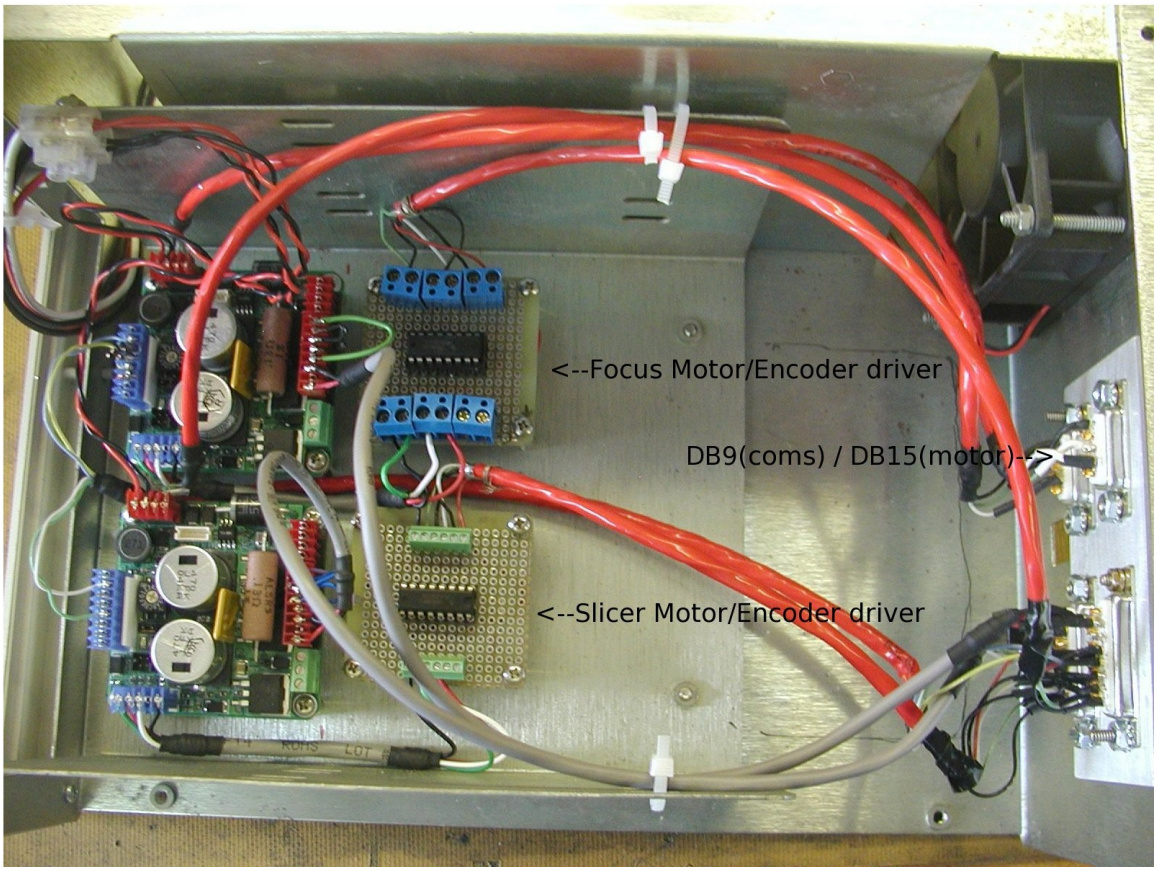
Pin		Signal	
1		Control input for optional ~~motor brake (0 to 24 V)~~	n.c
	9	~~n.c.~~ Motor Input (-)	
2		~~n.c.~~ Motor Input (+)	
	10	~~PGND~~	n.c.
3		~~PWM MAGN input~~	n.c.
	11	~~PWM SIGN input~~	n.c.
4		+5 V input	
	12	NLIMIT output (Limit signal of negative side)	
5		PLIMIT output (Limit signal of positive side)	
	13	~~Reference sensor output~~	n.c.
6		GND (Limit)	
	14	Encoder A(+) output	
7		Encoder A(-) output	
	15	Encoder B(+) output	
8		Encoder B(-) output	

d) Cable Pin out (Slicer and Focus motors)

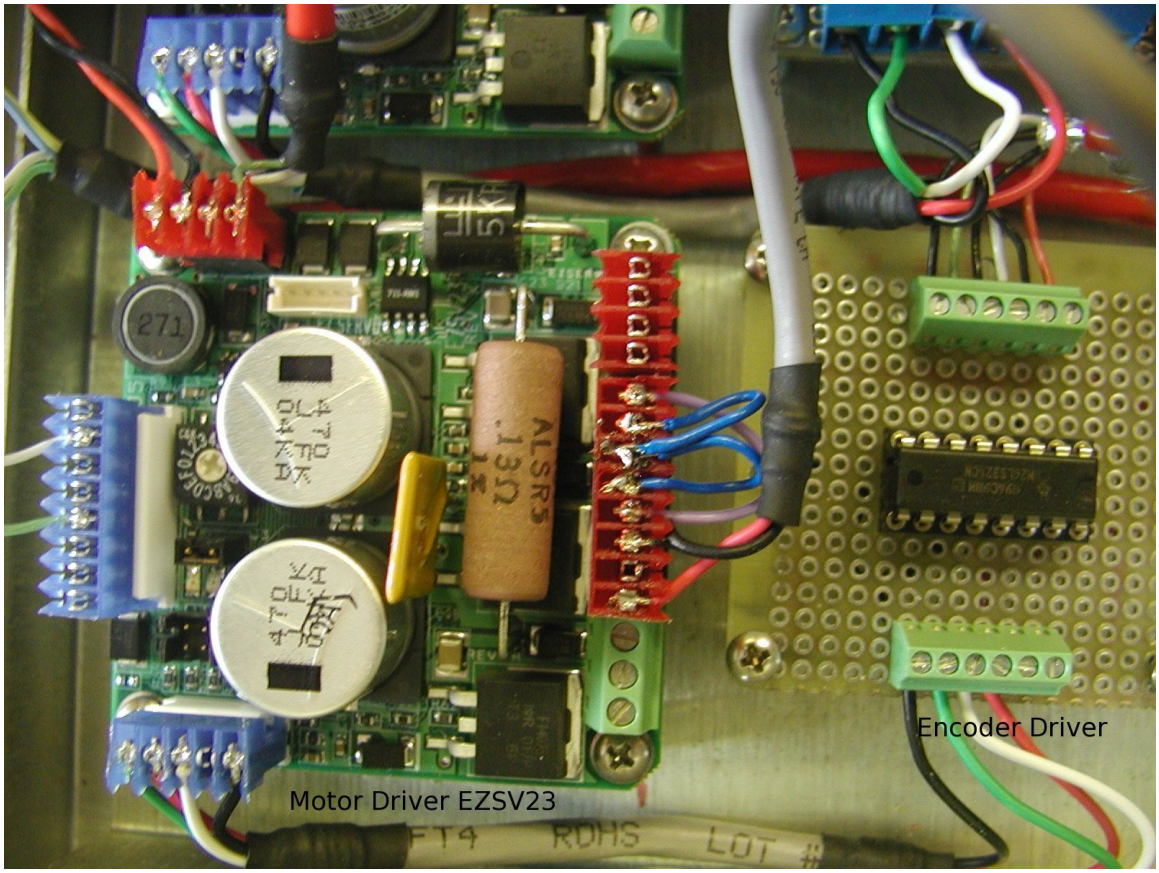
DB-15 Female (box)	DB-15 Male (Motor)	Description
2	2	Motor (+)
4	4	+5V (encoder)
5	5	Positive Limit
6	6	GND (Limit)
7	7	Encoder A(-)
8	8	Encoder B(-)
9	9	Motor (-)
12	12	Negative Limit
14	14	Encoder A(+)
15	15	Encoder B(+)
DB-9 Male (box)	**DB-9 Male (T.Server)**	**Description**
2	5	Tx/Rx-
8	7	Tx/Rx+
5	5	GND

Appendix B: Box photographs

b.1) General view



b.1) Detail



Appendix C: Lantronix EDS4100 Terminal Server Settings

The connection between the Lantronix and both motors is through a Fenmale/Female, DB9/DB9 null modem cable (crossed) cable

Settings using the Terminal Server Web interface:

Login: admin

Password: PASS

Identical for Line 1 and Line 3. We will show here for Line 1 only.

Line -> Line 1 -> Configuration

Interface: RS485 Half-Duplex

State: Enabled

Protocol: Tunnel

Baud Rate: 9600

Parity: None

Data Bits: 8

Stop Bits: 1

Flow Control : None (disabled)

Xon/Xoff Char (disabled)