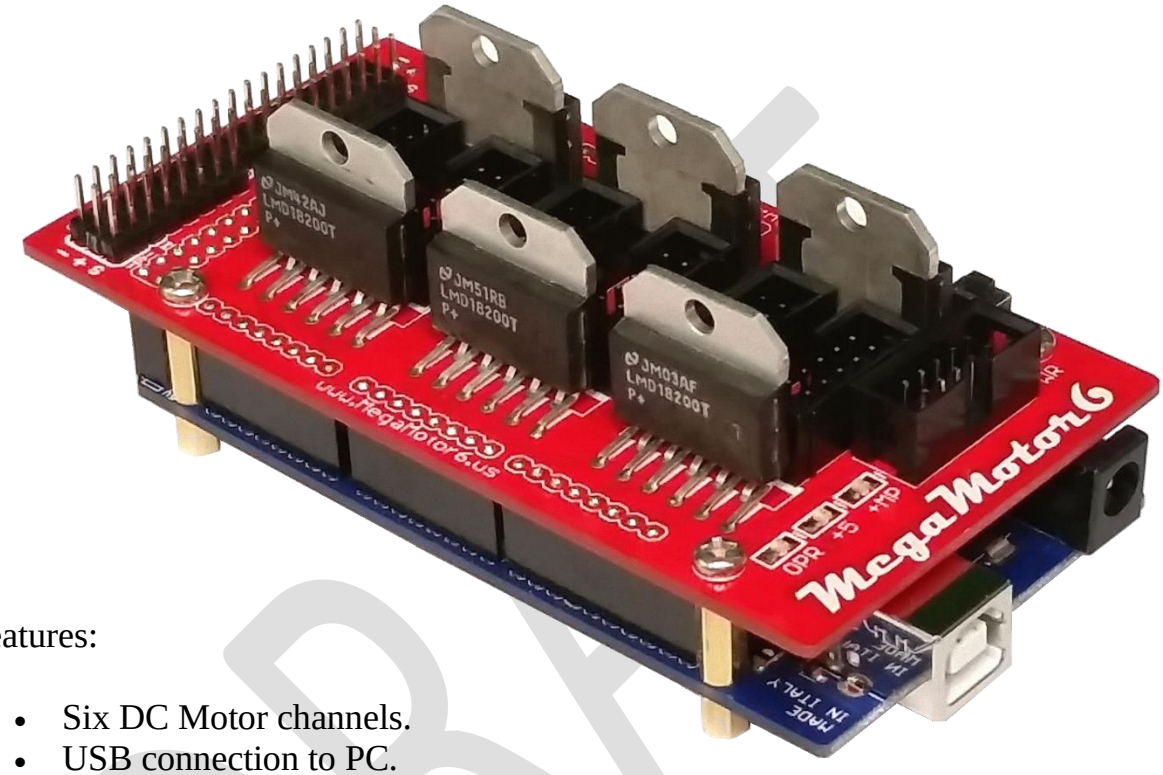
MegaMotor6

Six Channel Motor Controller Shield for the Arduino Mega



Features:

- Six DC Motor channels.
- USB connection to PC.
- Eight user-assignable Digital I/O ports.
- Four user-assignable Analog input ports.
- WIFI port. (Serial with flow control)
- Pixy Camera port. (I2C)

All six DC Motor channels have the following features:

- Large power output: 3-Amp (5-Amps peak) 12-55Volts
- Speed Control (Each channel can move at an independent speed.)
- Direction Control
- Active Braking
- Current Draw measurement.
- Thermal Protection
- Quadrant Encoding for accurate positioning
- Limit Switch input.

Draft Notice

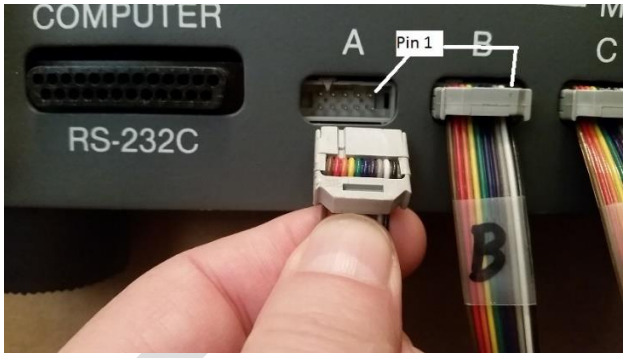
This document is in Draft form as the pertinent information is being gathered and the base software is expanded.

- Nothing is in any particular order.
- All information is assumed correct – but not guaranteed.

Report any errors to the email posted at
www.MegaMotor6.us

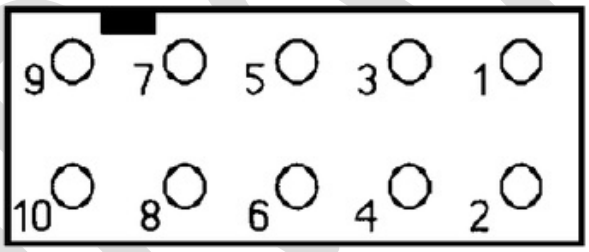
Table of Contents

The original Rhino Robot Controller



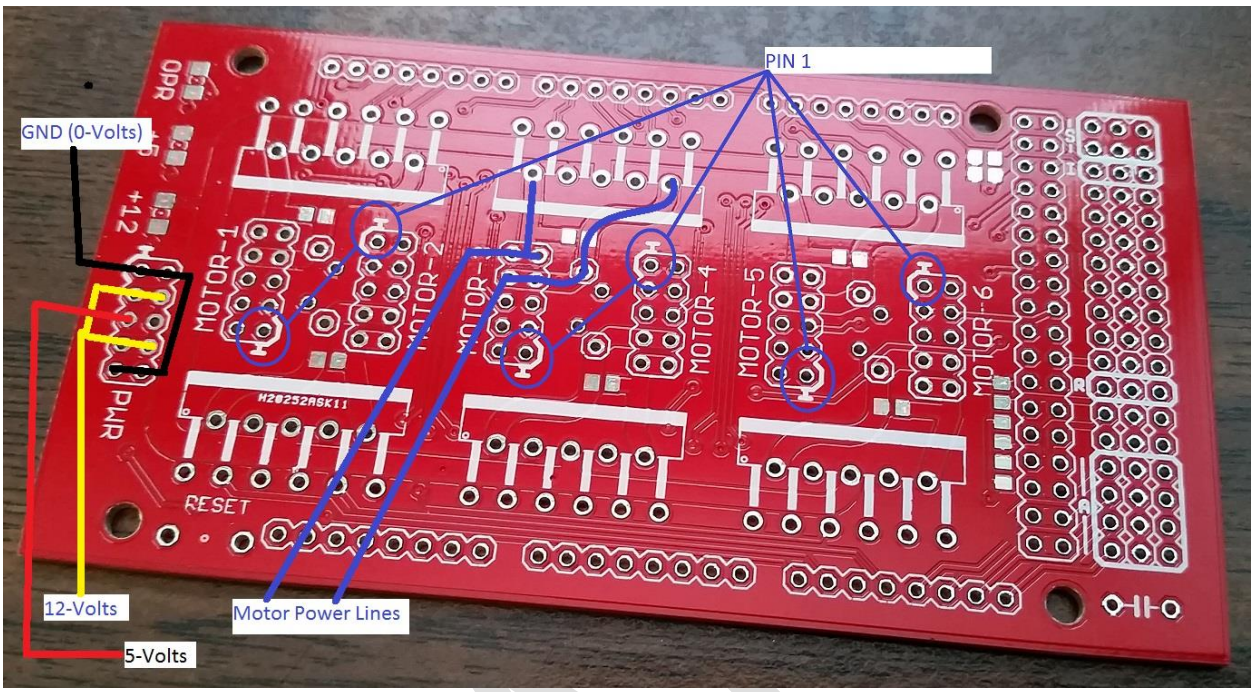
RHINO ROBOTICS LTD.
PINOUTS

The pinouts of the motor connector,
as seen with the plug removed, are as follows:



MARK III CONTROLLER	MARK IVC CONTROLLER
1. Logic Ground	1. Logic Ground
2. Optic "B"	2. Optic "B"
3. +5 Volts	3. +5 Volts
4. Optic "A"	4. Optic "A"
5. N/C	5. Chassis Ground
6. Limit Switch	6. Limit Switch
7. Motor Common	7. Motor Power - Positive
8. Motor Common	8. Motor Power - Positive
9. Motor Power	9. Motor Common - Negative
10. Motor Power	10. Motor Common - Negative

MegaMotor6 Board



The power pins are as follows.

0:0 - Gnd
0:0 - +12
- :0 - +5
0:0 - +12
0:0 = Gnd

When the power is applied, the Yellow and Red power LEDs will light up.
Yellow indicates that the 12 volts is good.
Red indicates that the 5 volts is good.

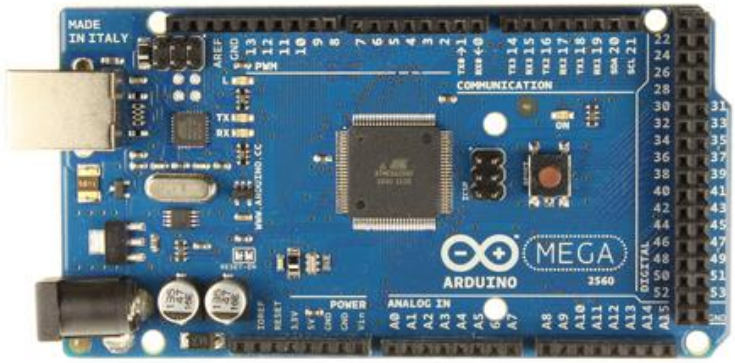
Note that the Red LED and the Rhino Robot electronics are powered via the PWR +5 connector - not the Arduino's +5.

The reset button is tied to the Arduino's reset line.
Hitting the reset button will shut off all motors while retaining the motor's encoder count.

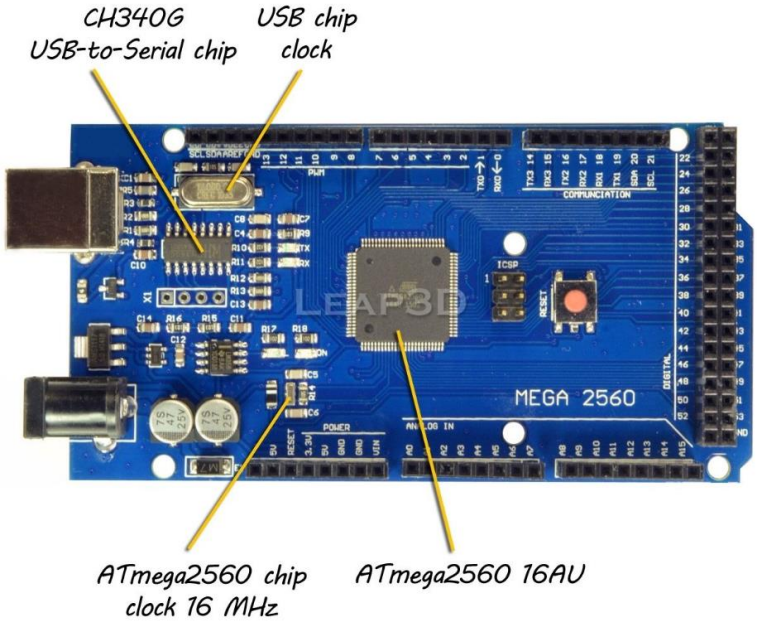
Arduino Mega

Being open source, the Arduino Megas come from a variety of manufactures.

Genuine Arduino Mega

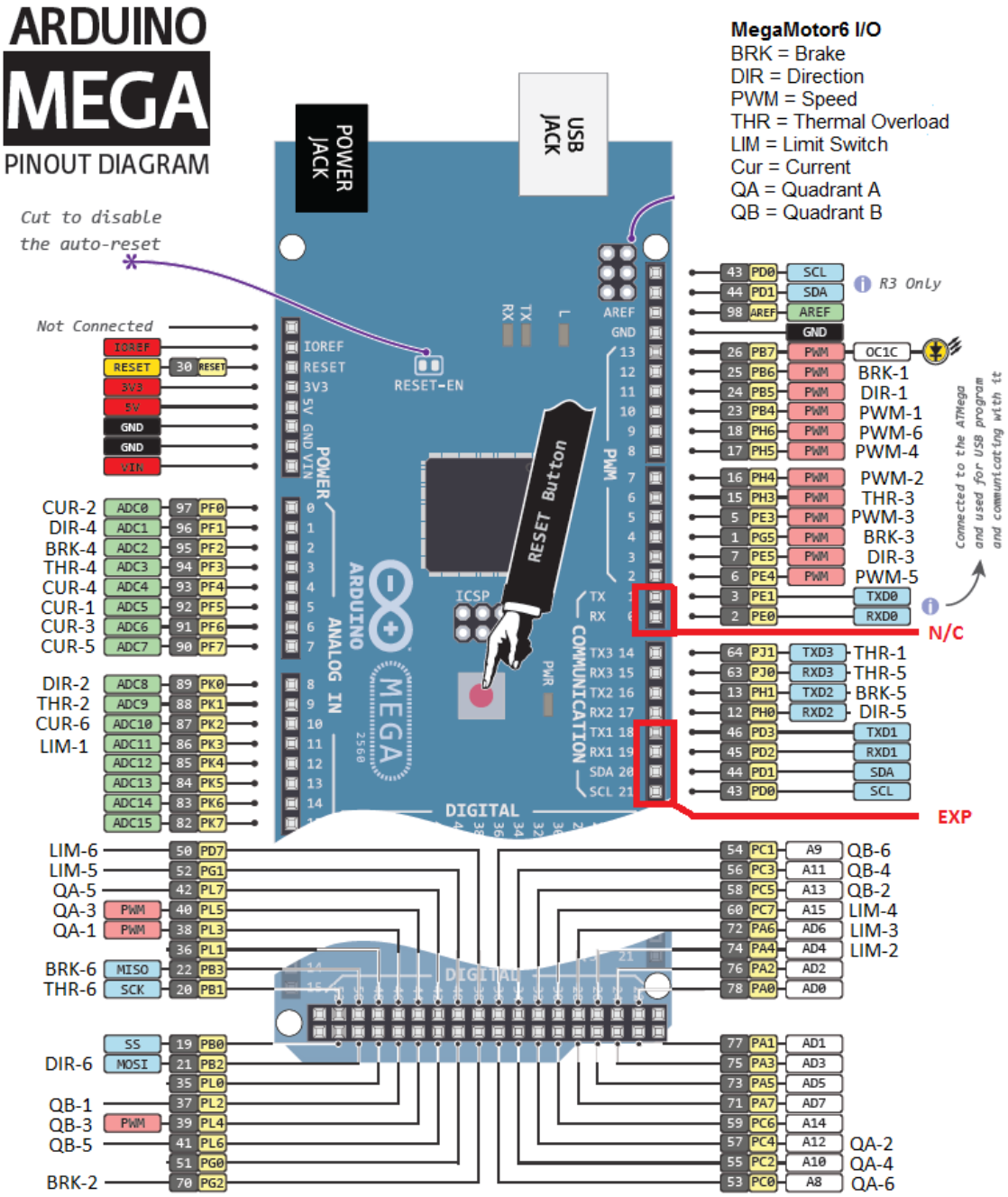


Clone Arduino Mega

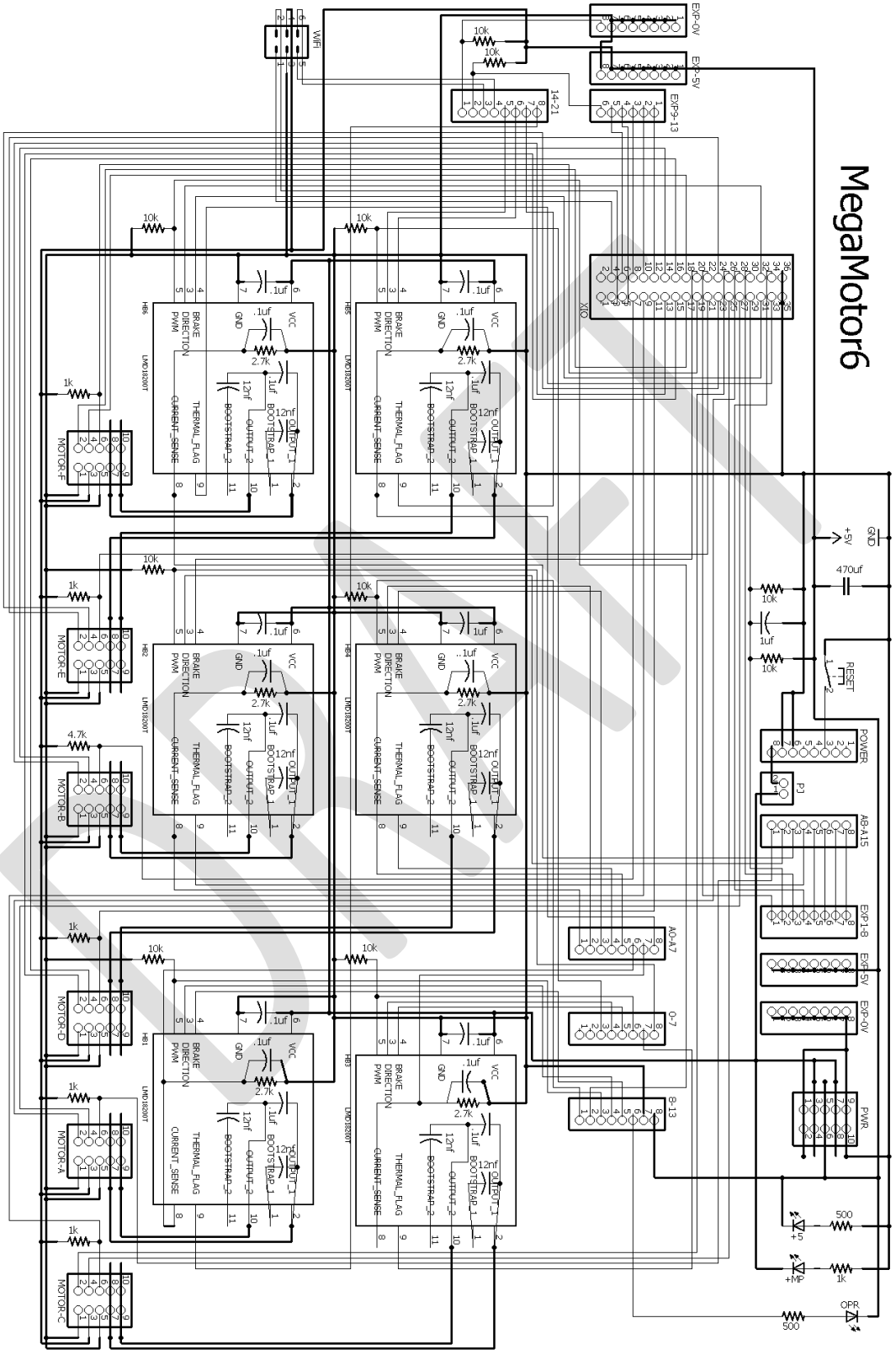


All Arduino Megas from all manufactures should work adequately and several have been verified to be compliant.

Arduino Mega / MegaMotor6 connections



MegaMotor6 Schematic



Rhino Robot Encoders

Figure 1 - Encoders Encased in Aluminum

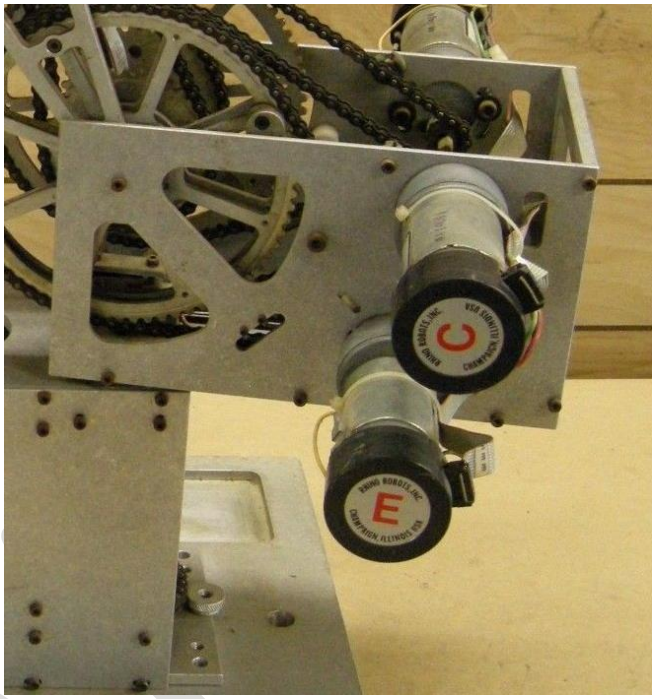


Figure 2 - Encoders Encased In plastic

Each of the Rhino Robot motors have an encoder attach which monitors the rotation of the motor. The MegaMotor6 uses these encoders to track the position of each motor. The tracking occurs even when the motors aren't being powered so that any movement cause by an outside force is also tracked.

Note that the Aluminum Encased Encoders (Figure 1 - Encoders Encased in Aluminum) short the "Motor Common" connection of the motor to the chassis. These shorting connectors **must be** insulated from the Chassis for the motors to work properly with the MegaMotor6.

Refer to the Insulating Aluminum Encased Encoders section for details.

Figure 3 - Open Frame Encoders

Insulating Aluminum Encased Encoders

Some of the Rhino Robots line of robots were shipped with an encoder that shorted one lead of the motors to the chassis - thereby forming a common connection between all the motors.

A common connection between motors worked fine for the Original Rhino Robot Controller because the Original Rhino Robot Controller used a +/- 12 volt system with one lead of the motor at ground and the other lead switching between +12 and -12 volts to drive the motor in one direction or the other.

The MegaMotor6 on the other hand uses a single +12-volt power source and reverses the polarity on the motor leads to switch between forward and backward movement which doesn't work with commonly connected motors.

If your Rhino Robot has the aluminum-encased encoders, then you will need to insulate the encoder's shorting pad from the chassis so that the motors no longer are commonly connected.

Figure 4 - Encoder with cap removed

Figure 5 - Encoder Shorting Screw Removal

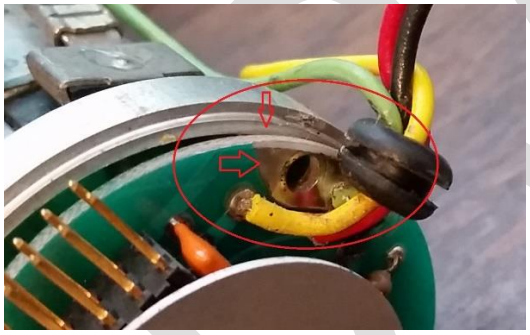


Figure 6 - Close up of the shorting pad

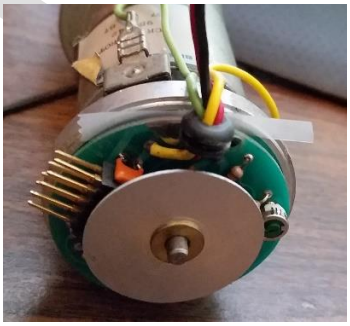


Figure 7 - Encoder with insulation

1. **Disconnect all motors from the MegaMotor6.**
2. Using an Allan Wrench, remove the aluminum cap.
3. Using the same Allan Wrench, remove the shorting screw AND loosen the screw on the other side of the encoder enough that the encoder can be tilted forward.
4. Place an insulator (Transparent, Cellulose or Masking tape works fine) between the chassis and the circuit board making sure that the shorting pad on both the front and back of the encoder can no longer can touch the chassis.
5. Retighten the screw on the other side of the encoder.
6. Replace the shorting screw with a nylon screw. (optionally, the shorting screw can be left out as the combination of the screw on the other side of the encoder and the aluminum frame will hold the encoder firmly in place)
7. Replace aluminum cap.

Diagnosing Motors.

When refurbishing a 30-year-old Rhino Robot, it is often the case that the Motors have been disconnected, reconnected in reverse, and/or damaged. In addition, damage to the encoders may have occurred over the years –or– the encoders may be shorting a common lead of the motors.

The “m” command diagnoses the motors and encoders in just such situations.

Typing the letter "m" and hitting enter will start a sequence that puts all Motor Drive outputs into a High-Z state and then test each motor sequentially by driving the motor forward for 1/20 of a second and then reverse for 1/20 of a second without any PID or any other control systems in effect.

While doing so, it monitors the encoder activity and reports that information to the PC’s screen via the Arduino’s USB connection.

This is what it a typical run will look like: (With the D Motor connected in reverse.)

```
------------------------------------------------------------
        m
        Test Motors
        Motors are now off.
        Current Positions: A=0,B=0,C=0,D=1,E=1,F=0.
        Setting Drive Power for all Motors to High-Z.
        Moving Motor A
           backward On Off
           forward  On Off
        Moving Motor B
           backward On Off
           forward  On Off
        Moving Motor C
           backward On Off
           forward  On Off
        Moving Motor D
           backward On Off
           forward  On Off
        Moving Motor E
           backward On Off
           forward  On Off
        Moving Motor F
           backward On Off
           forward  On Off
        Forward encoder count : A=+25 : B=+26 : C=+37 : D=-33 : E=+50 : F=+40.
        Reverse encoder count : A=-25 : B=-25 : C=-37 : D=+49 : E=-32 : F=-39.
        Current Positions: A=-2,B=1,C=0,D=23,E=21,F=1.
        Done Testing Motors
------------------------------------------------------------
```

What you are looking for is:

- The Forward encoder count to be a positive number
- The Reverse encoder count to be a negative number.

This will tell you if the motors are connected correctly and if your encoders are working.
Note that this will also work if you want to attach (and therefore diagnose) one motor at a time.
(If you do that, then only the motor you attached will show any results.)

If the Positive and Negative numbers are reversed, (As D is in the example above) then the Motor is connected in reverse.

Arduino Sketch

NOTE: Before Doing ANYTHING - There is a style of Rhino Robot motor that needs modification to operate properly with the MegaMotor6. See the section on Insulating Aluminum Encased Encoders**.**

Note: It is best to program your Arduino-Mega ***before*** attaching the MegaMotor6 Shield. This will verify that the control lines are set properly.

To install software in Arduino Sketch Editor, load the sketch named RhinoRobotController and then type Ctrl-U to upload.

Once you have uploaded the software to your Arduino, the Arduino's Status LED will start blinking.

Under Tools, Select the Serial Monitor option.

Set the baud rate to: 38400

You should see:

```
Ready
Current Positions: A=0,B=0,C=0,D=0,E=0,F=0.
```

At this point, it is safe to attach the MegaMotor6 Shield to the Arduino.

NOTE: Before attaching the Rhino Robot to the MegaMotor6, some diagnostics **MUST** be performed to make sure that your Rhino Robot has not been damaged over the years. Failure to verify certain electrical aspects are in working order could result in further damage to your robot and/or to the MegaMotor6
Refer to the Chapter: Diagnosing Motors.

Once your Rhino Robot is attached, Type the letter t and hit enter.

You should see:

```
t
Motor Status
Motors are off
A Home=1: Sta=0 Pos=0 Tar=0 Err=0 Spd=0 PWM=0 Cur=0 LmH=0,0,0,0,0
B Home=1: Sta=0 Pos=0 Tar=0 Err=0 Spd=0 PWM=0 Cur=0 LmH=0,0,0,0,0
C Home=1: Sta=0 Pos=0 Tar=0 Err=0 Spd=0 PWM=0 Cur=0 LmH=0,0,0,0,0
D Home=1: Sta=0 Pos=0 Tar=0 Err=0 Spd=0 PWM=0 Cur=0 LmH=0,0,0,0,0
E Home=1: Sta=0 Pos=0 Tar=0 Err=0 Spd=0 PWM=0 Cur=0 LmH=0,0,0,0,0
F Home=1: Sta=0 Pos=0 Tar=0 Err=0 Spd=0 PWM=0 Cur=0 LmH=0,0,0,0,0
```

The Home positions will reflect the status of the switched.

1 = Not at home

0 = At Home.

To manually drive the motors to home,

Type the motor letter (lower case) and an angle

For example:

f45 will turn the base 45 degrees.

f-45 will turn the base -45 degrees.

f0 will take the base back to where it started.

f = Base

e = Sholder

d = elbow

c = wrist x

b = wrist y

a = gripper

Upper case letters will move the motors in encoder clicks.

For example:

F6 will turn the encoder 360 degrees.

F0 will take the base back to where it started.

Once all motors have been positioned to home,

Type the letter z and hit enter to set the current position as "Home"

Type the letter i and hit enter to initiate the find-home-center routine seen in this video:

https://www.facebook.com/pg/RhinoRobot/videos/?ref=page_internal

Finally, making sure that you have room around the arm.

Type q and hit enter to run the simple move sequence seen in this video:

https://www.youtube.com/watch?v=dqY2jAFxeUc

Common Command Codes and responses.

g

Motors are now on.

s

Motors are now off.

h

Setting Targets to Home Position.

z

Current Positions set to Zero.

p

Current Positions: A=0,B=0,C=0,D=0,E=0,F=0.

t

Motor Status

Motors are off

A Home=1: Sta=0 Pos=0 Tar=0 Err=0 Spd=0 PWM=0 Cur=5 LmH=0,0,0,0,0

B Home=1: Sta=0 Pos=0 Tar=0 Err=0 Spd=0 PWM=0 Cur=4 LmH=0,0,0,0,0

C Home=1: Sta=0 Pos=0 Tar=0 Err=0 Spd=0 PWM=0 Cur=5 LmH=0,0,0,0,0

D Home=1: Sta=0 Pos=0 Tar=0 Err=0 Spd=0 PWM=0 Cur=4 LmH=0,0,0,0,0

E Home=1: Sta=0 Pos=0 Tar=0 Err=0 Spd=0 PWM=0 Cur=5 LmH=0,0,0,0,0

F Home=1: Sta=0 Pos=0 Tar=0 Err=0 Spd=0 PWM=0 Cur=5 LmH=0,0,0,0,0

Home indicates the status of the home position switch. 1 not closed, 0 closed.

Sta indicates how close the motors are to the target position.

Pos indicates the current encoder count

Tar indicates the current target encoder count.

Err indicates the PID error value.

Spd indicates the motor's speed.

PWM indicates the PWM value that drives the motor.

Cur indicates the current draw. (4-5 idle, > 100 possible stall)

LmH indicates home position statuses.

o

Opening Gripper.

v

Closing Gripper.

i - Interrogates home position switches.

q - Starts the test sequence.

w - Starts position tracking which sends any change in any motor's position out the serial port.

u - Stops position tracking.

All of the commands can be modified via the source code.

Installing the Rhino Robot Controller Sketch step-by-step.

- 1. Download the software zip file with the name RhinoRobotController.zip
- 2. Right click on the zip.
- 3. Choose "Extract all"
- 4. Accepted default of:

C:\{your directory}\RhinoRobotController

- 5. Run the Arduino software – (Ver 1.6.12 was used for development)
- 6. Clicked File-Open
- 7. Navigated to:

C:\{your directory}\ RhinoRobotController

- 8: Doubled click on:

RhinoRobotController.ino

- 9: Plugged in the Arduino (Without the MM6) into the USB
- 10: Select the correct port under Tools->Port
- 11: Select Sketch->Upload.

Your Arduino should now be blinking its LED.

- 12: Select Tools->Serial monitor
- 13: Set Baud to 38400 and End-Of-Line to: “Carriage return”

Your monitor should now say: "Ready" - "Current Positions: A=0,B=0,C=0,D=0,E=0,F=0."

- 14: Disconnect the Arduino from the USB
- 15: Reattached the MM6 to the Arduino (with no motors attached)
- 16: Attached the power supply to the MM6

The MM6's should now show this status:

- The Green light blinking.
- The red on.
- The yellow on.

- 17: Connect Motor.
- 18: Diagnose Motors (Refer to chapter: Diagnosing Motors.)