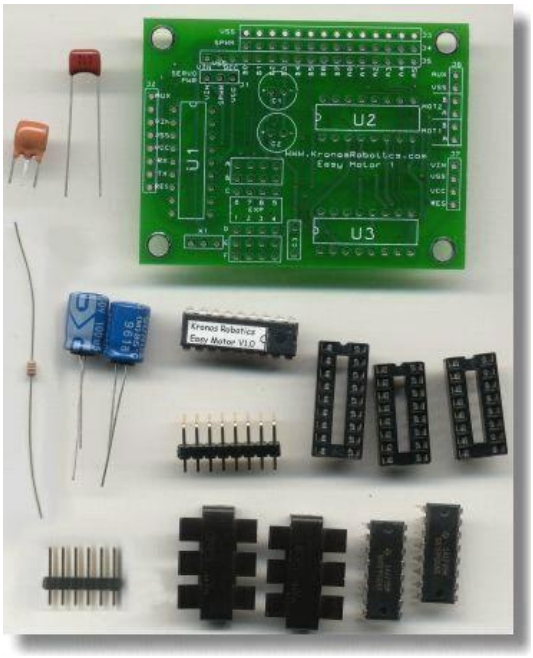
Easy Motor 1
Assembly and Hookup

Firmware version 1.4

Manual Version 1.1

Your Kit will include the following parts

- Printed circuit board
- Documentation
- 18 pin socket
- 2 16 pin sockets
- 6 pin strait header
- 8 pin right angle header
- 20 Mhz resonator
- 2 heat sinks
- 10K resistor
- 2 100uf capacitors
- .1uf capacitor
- Programmed 18 pin chip
- 2 motor driver chips



If you opted not to receive the motor driver chips and heat sinks they will not be included in the Kit. I offer this option for users who have already purchased the motor driver chips.

Also if you purchased the Deluxe headers they were not included in the previous list. They were however included in the kit.

Kronos Robotics and Electronics

Notice:

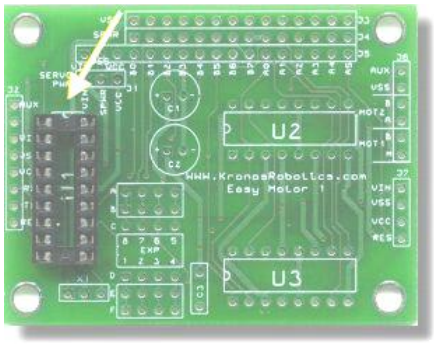
It is extremely important that you read this manual thoroughly before assembling and operation this module. Failure to do so may result in damage to the Easy Motor Module and/or your controller.

www.kronostobotics.com
msimpson@kronosrobotics.com

Table of Contents

Assembly

If you purchased your Serial LCD already assemble disregard this section.

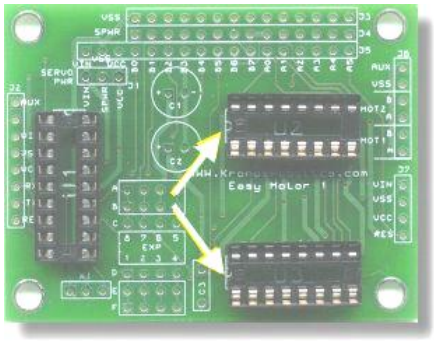


Step 1

Insert the 18 pin socket into the position labeled U1. Solder in place. Make sure the notch is facing the correct direction as indicated. It should be facing the top of the board. Solder in place.

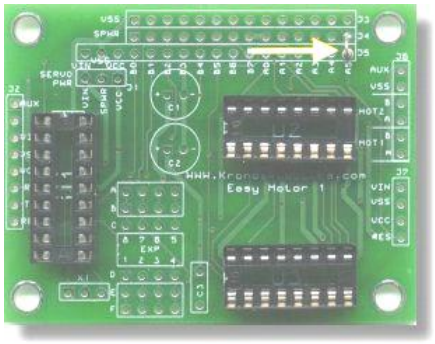
Tip

I placed all my sockets on the board all at once. I then place a small piece of wood on top of the sockets then flip the whole thing. This takes some practice but its well worth the effort once mastered.



Step 2

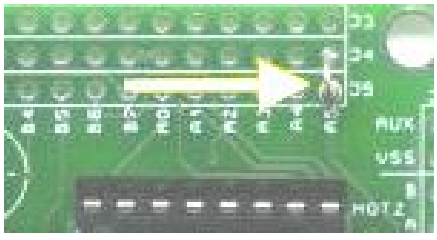
Insert the two 16 pin sockets into the positions labeled U2 and U3 with the notches located as indicated. Solder in place



Step 3

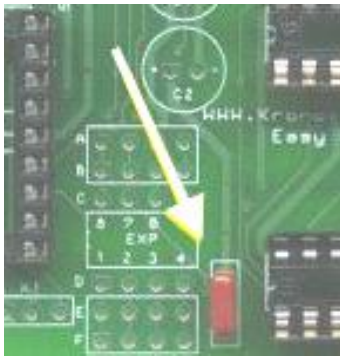
Insert the 10k Resistor into the A5 slot as shown.

Solder in place and clip the excess leads.



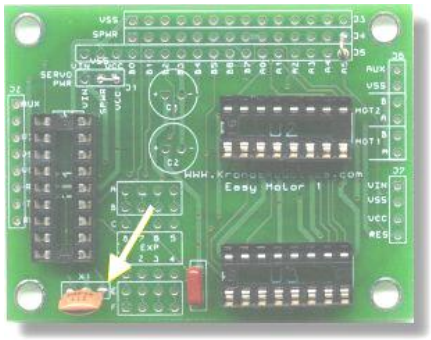
Step 4

Insert one of the clipped leads into the position marked J1 as shown. Solder in place and clip the excess leads.



Step 5

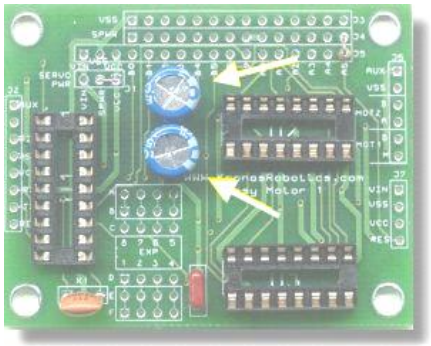
Insert the .1uf capacitor into the position marked C3 as shown. Solder in place and clip the excess leads.



Step 6

Insert the 20Mhz resonator into the position marked X1 as shown. Solder in place and clip excess leads.

X1 is the only 3 pin component in the kit.

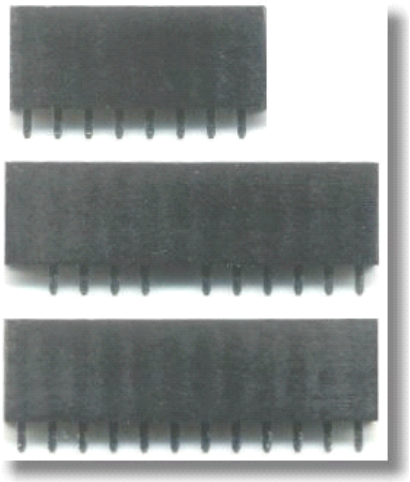


Step 7

Insert the two 100uf capacitors into the positions marked C1 and C2 as shown. The negative side of each capacitor marked with a - should face the U2 Socket (right).

Solder in place and clip the excess leads.

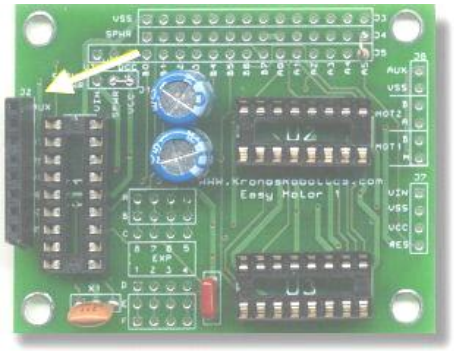
If you did not purchase the *Deluxe Headers* proceed to step 9.



The three female headers present you with more options for connecting the motor wires to your Easy module.

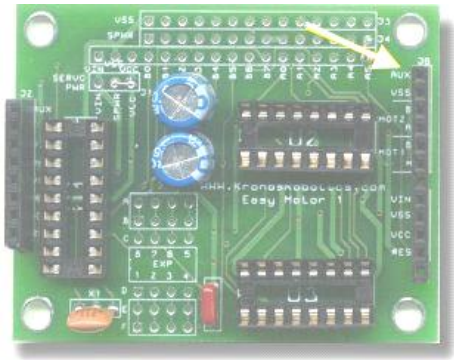
The main advantage of using these headers is that you can just plug your wires into the module just like a breadboard. And while they don't offer as much current flow as soldering your wires in place they work just fine in most circumstances.

Also don't feel you are restricted to the instructions that I provide here. Feel free to mix and match both the male and female connectors to meat your needs.



Step 8a

Insert the 8 pin female header into J2 as shown. Solder in place.

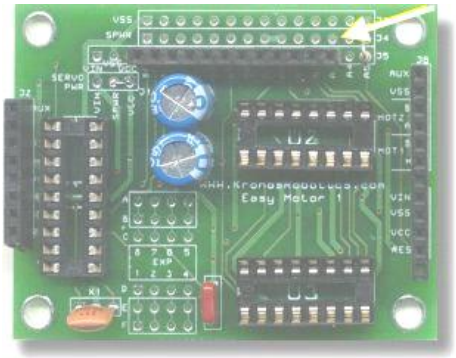


Step 8b

Insert the modified 12 pin connector into J6 and J7 as shown.

This connector is the one with the pins missing. Place the end with the 6 connectors into the J6 slot.

Solder in place.



Step 8c

Insert the 12 pin header into the J5 slot as shown. Note that the A5 and A4 holes are not used.

Solder in place.

Once all the headers have been installed proceed to Step 11.

Step 9

Insert the 8 pin right angle male header into the J2 position as shown.

Solder in place.

Step 10

Insert the 6 pin strait header into the J6 position as shown.

Solder in place.

Step 11

Insert the Easy Motor 1 smart chip into the 18 pin socket as shown. Note the position of pin 1. (marked by the dot)

The notch should face the top of the board.

Quick Test

At this point we need to do a test to verify that the module has been properly assembled.

Connect +5v to Vcc and Vss.

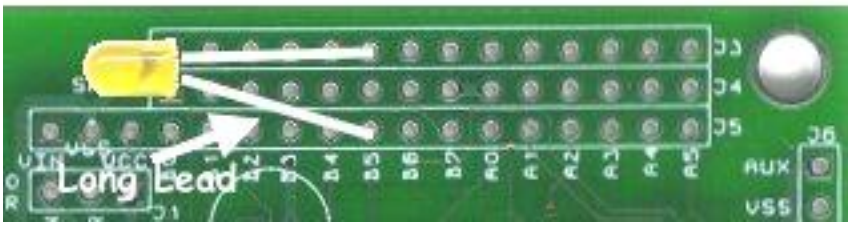


Figure 1

Touch the included LED to the pads as shown in figure 1. The long lead touches the B5 pad on J5 and the short lead (flat side of LED) touches any of the pads on J3.

The light LED should light.

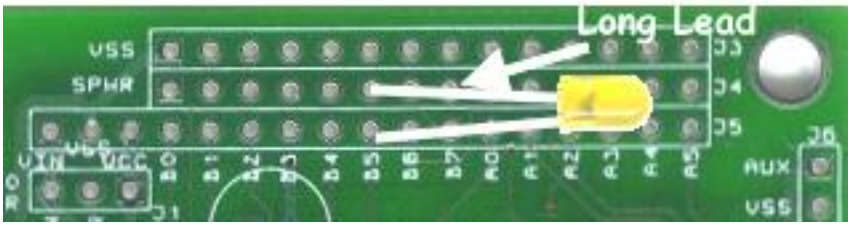


Figure 2

Referring to Figure 2 touch the long lead to any pad on J4. Touch the short lead to the B5 pad on J5.

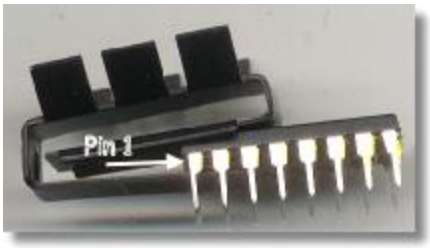
Again the LED should light.

What did we just do?

One of the PWM generators is output on pad B5 of J5. It cycles between low and high states and by reversing the LED we test it in its high state and then again in its low state. If the PWM generator is running then you have connected everything up correctly at this point.

Tip: The other PWM generator is located on pad B6.

If the LED did not light in both directions you need to go back and check all your connections. If it did light proceed to step 12.



Step 12

Insert the motor driver chip into the heat sink as shown. Insert the pin 1 side first. If you insert the pin 1 end first then pin 1 will always be the low end of the heat sink.

Slide the chip in until it stops.

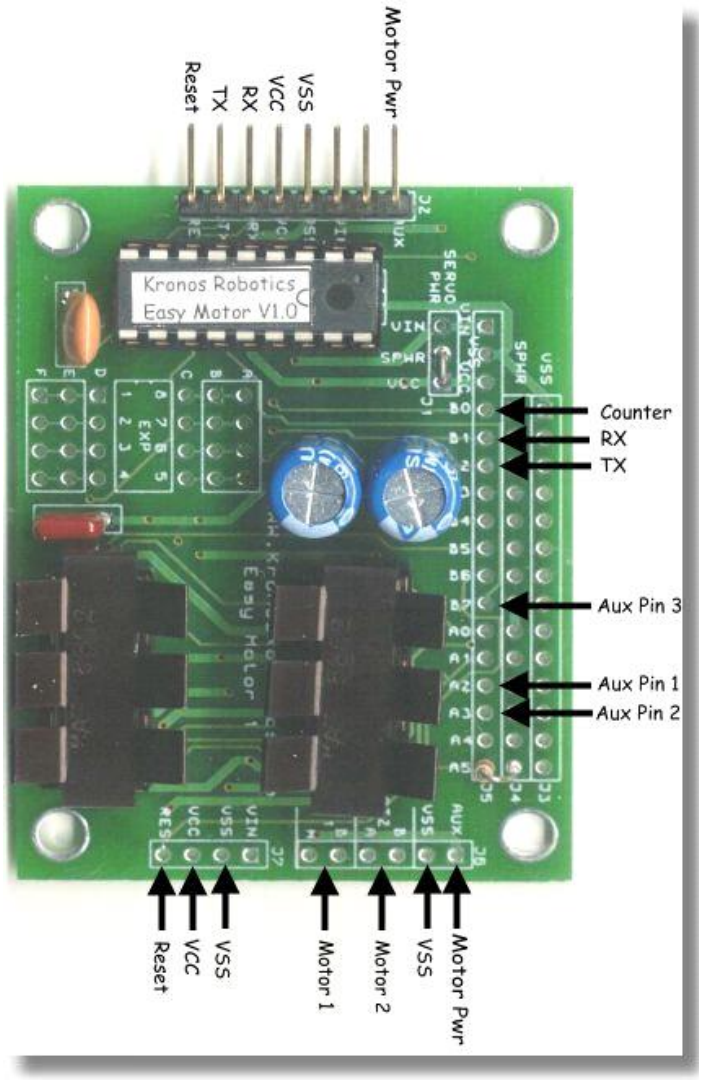
Step 13

Now insert the driver chips into the sockets. Pin 1 of the chips (The low end) should face in the direction of the capacitors.

Please note that you will have to squeeze the pins a bit to make them narrower. For some reason TI chips seem to come with leads that are a bit spread out.

Hint
Once the chip is in place you may find it necessary to use a screwdriver to press the front of the chip in place to make a better connection.

Hookup



For versatility many of the connections appear at multiple locations. You need only connect to one place on the board. For instance all VSS connections are connected together. I do recommend that you use the Motor Power and VSS connections located next to the 2 motor connections for your motor power.

Also I found it best to power the Easy Motor via the VSS and VSS connections located on the lower left of the board. Then power you microcontroller from the VSS and VCC connections located on the top of the board.

VSS

Connect this pin to GND

VCC

Connect this pin to +5v DC

Reset

This pin is held high and you need not connect this pin to anything unless you want to provide a circuit reset button.

RX

This is the motor controller serial receive pin. Connect this pin to the output serial pin of your microcontroller.

TX

If you are using any of the commands that require a transmit path you will need to connect this pin to the microcontroller serial input. Also note that when the motor controller receives a command that tells it to transmit some data there is a small amount of delay so that the microcontroller can setup the serin command.

Motor 1

These two pins connect to your Motor #1

Motor 2

These two pins connect to your Motor #2

Aux Pin 1-3

Auxiliary I/O pins

Motor Pwr

This pin supplies power to the motors. In some cases you can use a single power source for both your motor power and logic. It all depends on your power supply and controller. I recommend the using separate power source for your motor power until you get your configuration operating correctly.

Connect the negative side of your motor power source to the VSS lead closes to the motor connections.

Easy Motor Command Syntax

Lets take a look at some basic syntax for communicating with the Easy motor controller

A command consists of 5 bytes of information in the following form.

ID Command Data1 Data2 Data3

ID

255,1,100,0,0

By default all Easy Modules will respond to an ID of 255. If you connect your controller to to the Rx Pin of 2 Easy Modules and issue a command with the ID of 255 both will respond.

You can as an option program the ID of any Easy Module by sending the fallowing sequence. 255,1,NID,0,0 Where NID = is the new ID of the Module.

Command

This is where we tell the module what to do. For instance the command sequence.

255,1,100,0,0

Will set the unique ID of the connected easy module to 100.

Data1,Data2,Data3

255,1,100,0,0

Each command will may require one or more data elements. In this case only the 100 element is used but he other two elements must be filled with the 0 value as a place holder.

In the next section we will look at the syntax of each command.

Easy Motor 1 Command Syntax

Set Motor Direction

Cmd: 8

Syntax: ID,8,M1 Dir,M2 Dir,0

Description:

This is the most important command for motor direction control.

Motor direction settings are as follows.

- 0 : Stop
- 1 : Forward
- 2 : Reverse

Example:

SmartModule

serout 7,255,8,1,1,0 or hserout 255,8,1,1,0

ATOM

serout 7,i9600,[255,8,1,1,0]

BasicStamp

serout 7,84,[255,8,1,1,0]

This will put both motors in the forward direction. Also not that if you set a motor to forward and it turns the opposite direction you intend just reverse the leads to the motor in question.

Set Motor duty cycles as a percent

Cmd: 9

Syntax: ID,9,M1 Duty,M2 Duty,0

Description:

This command set the duty cycle as a percentage. For instance a setting of 90 sets the duty cycle to 90%.

Example:

SmartModule

serout 7,255,9,90,90,0 or hserout 255,9,90,90,0

ATOM

serout 7,i9600,[255,9,90,90,0]

BasicStamp

serout 7,84,[255,9,90,90,0]

This example sets both motors to a 90% duty cycle.

Easy Motor 1 Command Syntax (Advanced)

Request Module ID

Cmd: 0

Syntax: ID,0,0,0,0

Description:

This command was created just in case you programmed a new ID into the module and forgot it. Just use the Global ID of 255 and the module will return its private ID.
Please note that this command requires the use of an additional pin so that the module can have a transmit path back to your controller. The module will return a single byte in the range of 1 - 255.

Example:

SmartModule

serout 7,255,0,0,0,0 or hserout 255,0,0,0,0

serin 6,IDVARB or hserin IDVARB

ATOM

serout 7,i9600,[255,0,0,0,0]

serin 6,i9600,[IDVARB]

BasicStamp

serout 7,84,[255,0,0,0,0]

serin 6,84,[IDVARB]

Set New Module ID

Cmd: 1

Syntax: ID,1,New ID,0,0

Description:

If you are going to stack multiple modules you will need to program each module with its own private ID. Each module has two ID's. The global ID which is always 255 and a private ID that can be set from 1-255. If you don't know the current private ID of a module just use the global ID of 255 and your new ID (100 in this case) will override the old one.

Example:

SmartModule

serout 7,255,1,100,0,0 or hserout 255,1,100,0,0

ATOM

serout 7,i9600,[255,1,100,0,0]

BasicStamp

serout 7,84,[255,1,100,0,0]

Reset Factory Defaults

Cmd: 2

Syntax: ID,2,0,0,0

Description:

Sets the factory defaults.

Example:

SmartModule

serout 7,255,2,100,0,0 or hserout 255,2,100,0,0

ATOM

serout 7,i9600,[255,2,100,0,0]

BasicStamp

serout 7,84,[255,2,100,0,0]

Read Smart Chip Version

Cmd: 3

Syntax: ID,3,0,0,0

Description:

Sends the version number

Note that this command requires the use of an additional pin so that the module can have a transmit path back to your controller. The module will return a single byte in the range of 1 - 255.

Example:

SmartModule

serout 7,255,3,0,0,0 or hserout 255,3,0,0,0

ATOM

serout 7,i9600,[255,3,0,0,0]

serin 6,i9600,[VERVARB]

BasicStamp

serout 7,84,[255,3,0,0,0]

serin 6,84,[VERVARB]

Deactivate Transmitter

Cmd: 4

Syntax: ID,4,0,0,0

Description:

The transmitter will turn itself off automatically when traffic on the serial bus stops. However for a busy bus it may be necessary for the host to free the bus by turning off the transmitter. Sending 255,4,0,0,0 tells all devices on the bus to shut down there transmitters.

Example:

SmartModule

serout 7,255,4,0,0,0 or hserout 255,4,0,0,0

ATOM

serout 7,i9600,[255,4,0,0,0]

BasicStamp

serout 7,84,[255,4,0,0,0]

Set Transmit Delay

Cmd: 5

Syntax: ID,5,Delay,0,0

Description:

Sets the amount of time in milliseconds that the module will wait before transmitting requested data. The delay is also inserted between multiple data segments as well.

Example:

SmartModule

serout 7,255,5,5,0,0 or hserout 255,5,5,0,0

ATOM

serout 7,i9600,[255,5,5,0,0]

BasicStamp

serout 7,84,[255,5,5,0,0]

Set PWM Frequency and duty cycle

Cmd: 6

Syntax: ID,6,Freq,Duty1,Duty2

Description:

This command sets the frequency and duty cycles of the PWM signal generator. Here is a short sample of the frequencies generated for various settings.

- 255 : 2140Hz
- 200 : 2720Hz
- 150 : 3620Hz
- 100 : 5400Hz
- 50 :10500Hz

If the duty setting exceeds the frequency setting the PWM for that motor will be turned off and a constant active signal will be sent to the motor control chips. For example if 200 is used for the frequency setting and a setting of 100 is used for the duty of motor 1 it will have a 50% duty cycle. Which means it is turned on 50% of the time. If a duty value for motor 1 is set to 201 or greater the duty cycle will be 100%.

Also note that the lowest duty setting is 1. If the duty is set to 0 at any time the motor will be activated constantly (100% Duty).

In general this command is only provided to give the user access to the raw counters used in the PWM signal software. For the most part the Frequency setting will be set to 255 (default) and command #5 will be used to set the duty cycles.

Example:

SmartModule

serout 7,255,6,255,200,200 or hserout 255,6,255,200,200

ATOM

serout 7,i9600,[255,6,255,200,200]

BasicStamp

serout 7,i9600,[255,6,255,200,200]

This example sets the Frequency to 1380Hz and the duty cycle to 78% for both motors.

Set Motor directions and Duty

Cmd: 7

Syntax: ID,7,Motors,Duty1,Duty2

Description:

Bits 0 and 1 of the motors setting are used to set the direction of motor1. Bits 2 and 3 are used for motor 2.

Duty1 and Duty2 work the same as in command 2 above.

Again this command is only provided to give the user low-level access to the motor control leads. Command #4 should be used most of the time.

Example:

SmartModule

serout 7,255,7,12,200,200 or hserout 255,7,12,200,200

ATOM

serout 7,i9600,[255,7,12,200,200]

BasicStamp

serout 7,84,[255,7,12,200,200]

Set Motor Directions

Cmd: 8

Syntax: ID,8,M1 Dir,M2 Dir,0

Description:

This is the most important command for motor direction control.

Motor direction settings are as follows.

- 0 : Stop
- 1 : Forward
- 2 : Reverse

Example:

SmartModule

serout 7,255,8,1,1,0 or hserout 255,8,1,1,0

ATOM

serout 7,i9600,[255,8,1,1,0]

BasicStamp

serout 7,84,[255,8,1,1,0]

This will put both motors in the forward direction. Also not that if you set a motor to forward and it turns the opposite direction you intend just reverse the leads to the motor in question.

Set Motor duty cycles as a percent

Cmd: 9

Syntax: ID,9,M1 Duty,M2 Duty,0

Description:

This command set the duty cycle as a percentage. For instance a setting of 90 sets the duty cycle to 90%.

Example:

SmartModule

serout 7,255,9,90,90,0 or hserout 255,9,90,90,0

ATOM

serout 7,i9600,[255,9,90,90,0]

BasicStamp

serout 7,84,[255,9,90,90,0]

This example sets both motors to a 90% duty cycle.

Set Output state of 3 Aux pins

Cmd: 10

Syntax: ID,10,state1,state2,state3

Description:

Each Easy Motor module has 3 auxiliary ports. See the connections section for the location of these ports.

The ports can be set for input or output as needed. When set for output this command will se the state of each pin to high if 1 and low if 0.

Example:

SmartModule

serout 7,255,10,1,1,1 or hserout 255,10,1,1,1

ATOM

serout 7,i9600,[255,10,1,1,1]

BasicStamp

serout 7,84,[255,10,1,1,1]

This example sets all three auxiliary ports to a high state.

Set Aux pins as Input or Output

Cmd: 11

Syntax: ID,11,Dir1,Dir2,Dir3

Description:

This command is used to actually se the auxiliary pins to input or output.

- 0 : Output
- 1: Input

Example:

SmartModule

serout 7255,11,0,0,1 or hserout 255,11,0,0,1

ATOM

serout 7,i9600,[255,11,0,0,1]

BasicStamp

serout 7,84,[255,11,0,0,1]

This example sets the first two ports as outputs and the third as input.

Read State of 3 Aux Pins

Cmd: 12

Syntax: ID,12,Pin1,Pin2,Pin3

Description:

This command lets us read the state of the three pins. If you specify a 1 in the Pin data field that auxiliary pin will be read and displayed.

Please note that this command requires the use of an additional pin so that the module can have a transmit path back to your controller.

Example:

SmartModule

serout 7,255,12,1,1,1 or hserout 255,12,1,1,1

serin 6,in1,in2,in3 or hserin in1,in2,in3

ATOM

serout 7,i9600,[255,12,1,1,1]

serin 6,i9600,[in1,in2,in3]

BasicStamp

serout 7,84,[255,12,1,1,1]

serin 6,84,[in1,in2,in3]

This example tells the easy motor module to send the state of all three auxiliary pins.

Start Interrupt Counter

Cmd: 13
Syntax: ID,13,0,0,0
Description:
Example:

Stop Interrupt Counter

Cmd: 14
Syntax: ID,14,0,0,0
Description:
Example:

Set rising or falling edge of interrupt counter

Cmd: 15
Syntax: ID,15,Edge,0,0
Description:
Example:

Clear Interrupt Counter

Cmd: 16
Syntax: ID,16,0,0,0
Description:
Example:

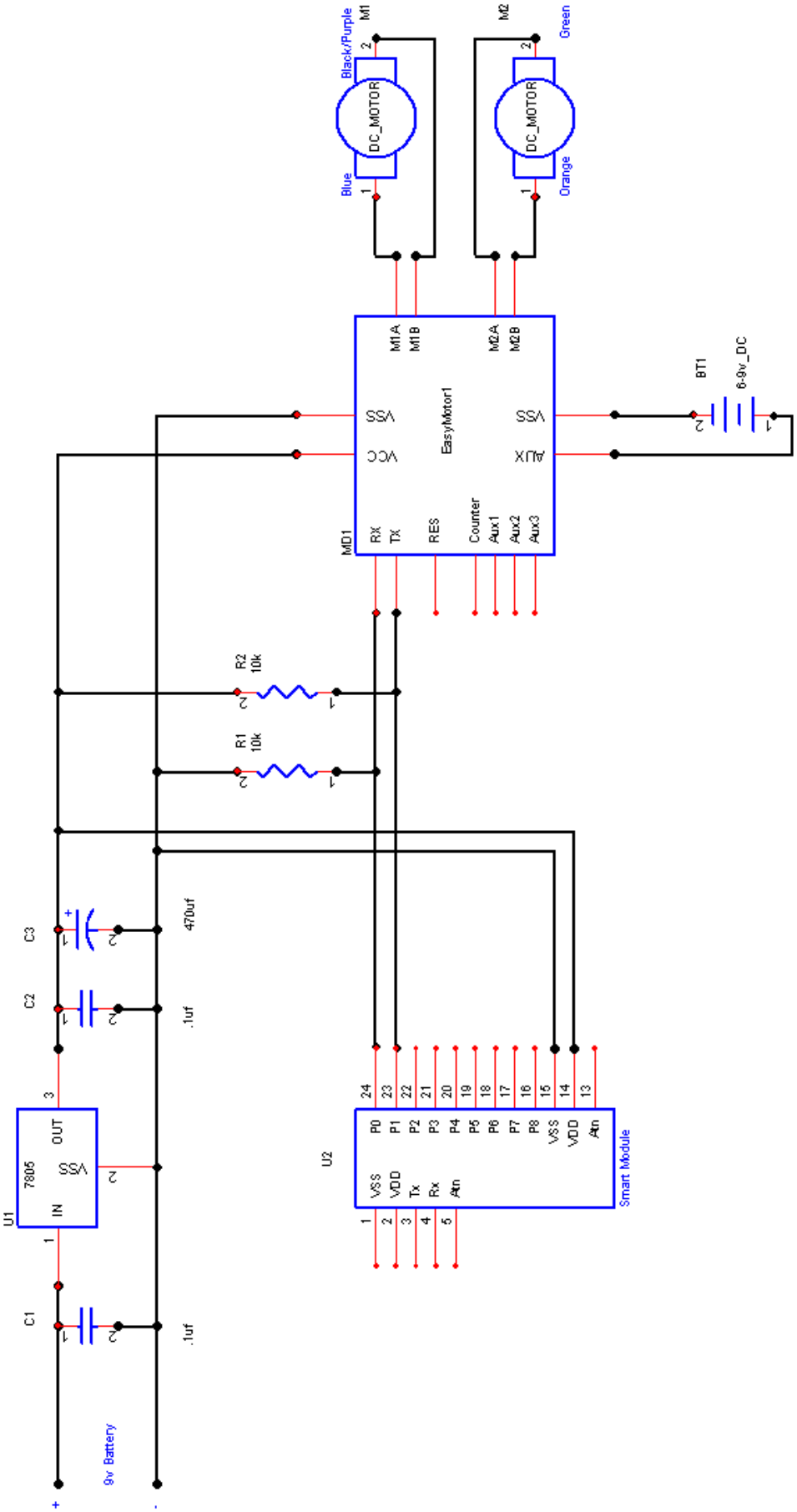
Read Interrupt counter

Cmd: 17
Syntax: ID,17,0,0,0
Description:
Example:

Code Example for Smart Module

See Schematic 1

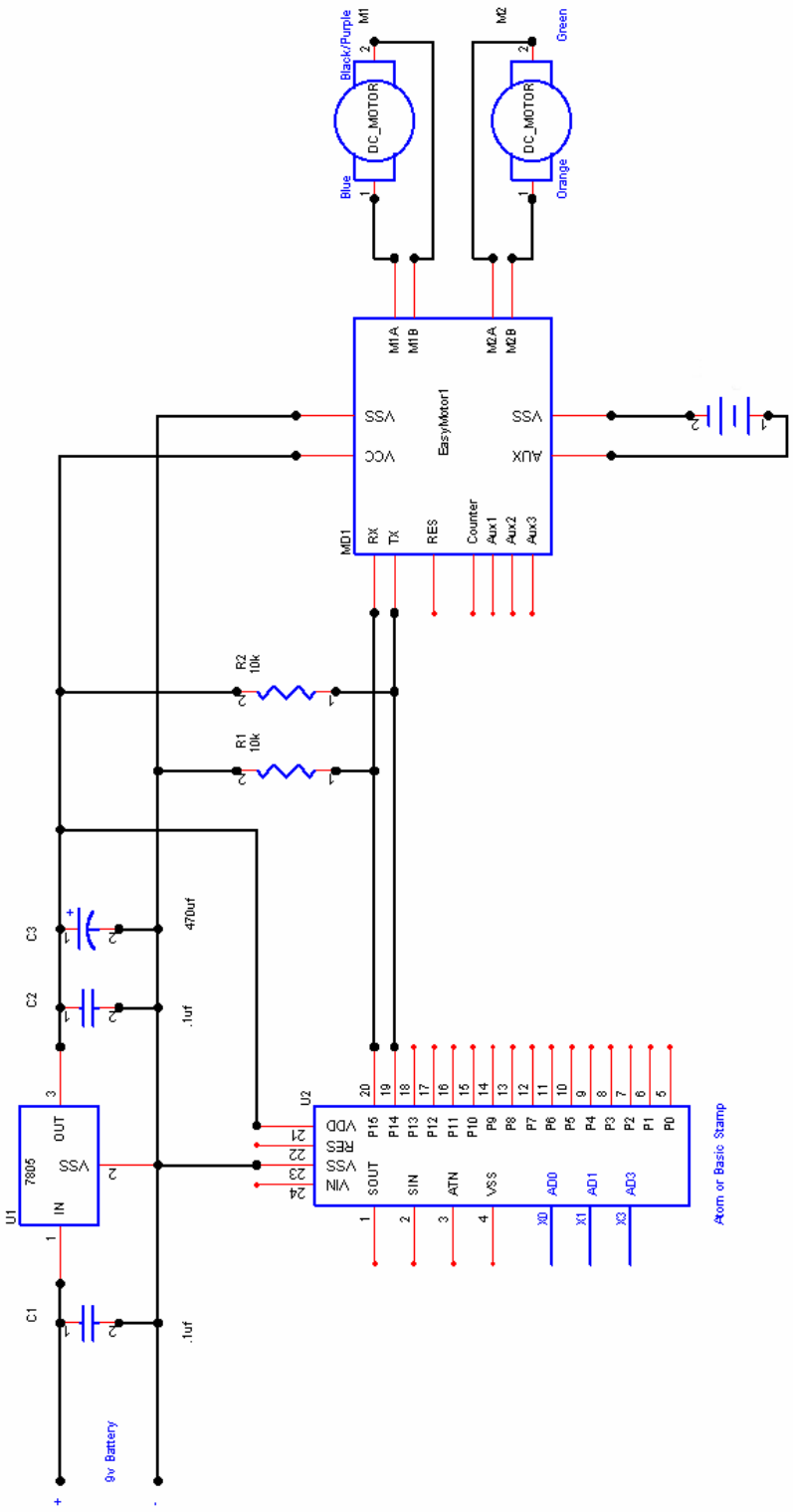
```
'Give it all a chance to init
pause 200

'Stop motors just in case
serout 0,255,8,0,0,0

'Set the speed to 90%
Serout 0,255,9,90,90,0

Loop:

'Move forward
Serout 0,255,8,1,1,0

Pause 2000

'Move backward
Serout 0,255,8,2,2,0

Pause 2000

Goto loop
```



Schematic 1

Code Example for Atom and Stamp

See Schematic 2



Schematic 2

Atom

```
'Give it all a chance to init
Pause 200

'Stop motors just in case
Serout 15,i9600,[255,8,0,0,0]

'Set the speed to 90%
Serout 15,i9600,[255,9,90,90,0]

Loop:

'Move forward
Serout 15,i9600,[255,8,1,1,0]

Pause 2000

'Move backward
Serout 15,i9600,[255,8,2,2,0]

Pause 2000

Goto loop
```

Stamp

```
'Give it all a chance to init
Pause 200

'Stop motors just in case
Serout 15,84,[255,8,0,0,0]

'Set the speed to 90%
Serout 15,84,[255,9,90,90,0]

Loop:

'Move forward
Serout 15,84,[255,8,1,1,0]

Pause 2000

'Move backward
Serout 15,84,[255,8,2,2,0]

Pause 2000

Goto loop
```

Changing the Baud Rate of an Easy Module to 115200

The Easy Module supports two baud rates. 9600 (default) and 115200. Just keep in mind that if you change the baud rate to 115200 it will effect both transmit and receive. Your controller must be able to keep up.

It is recommended that you keep the default baud rate until you have the module wired and working in your application before you change it.

You can change the default baud rate from 9600 to 115200 by following the following steps.

1. Turn off power to the controller.
2. Short out pins A0 and B7 as shown in Fig 1
3. Apply power to controller
4. Wait 5 seconds
5. Turn off power to controller and remove short.

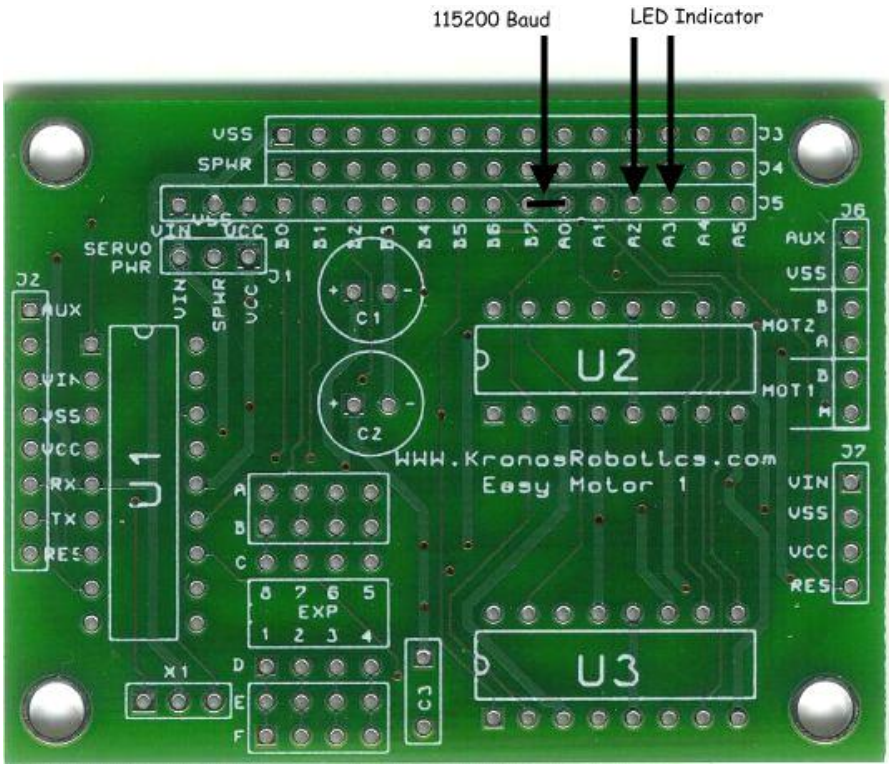


Fig 3

Changing the Baud Rate of an Easy Module to 9600

You can change back to 9600 baud by using the following steps

1. Turn off power to the controller
2. Short out pins A0 and A1 as shown in Fig 2
3. Apply power to controller
4. Wait 5 seconds
5. Turn off power to the controller and remove short.

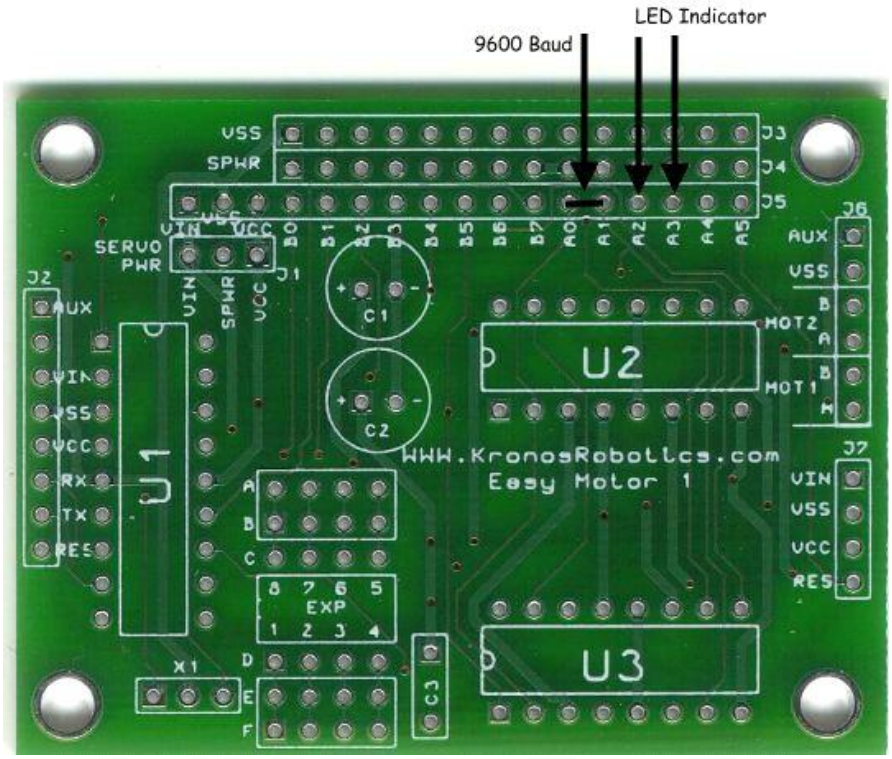


Figure 4

Notes on changing baud rate:

If you are not sure what that you are correctly setting the baud rate just connect a led between pins labeled A2 and A3. Short out one of the indicated pins and power up. The LED will blink slow for 9600 and fast for 115k.

Links

Be sure to visit the Kronos Robotics web site for more information and updates. You can also download example programs for various processors for the serial LCD.

Web Site
http://www.kronosrobotics.com

Full color assembly instructions and other downloads
http://www.kronosrobotics.com/products/pdfs/pdfs.htm