Getting Started with the Supermodified™

May, 2010 – REV. 1.00

The Supermodified™ achieves unsuppressed functionality in a miniscule package. Measuring just 15.5x15.5x12mm offers industrial-grade control capabilities. Astounding performance in such a small form-factor translates to everything being miniaturised and very closely stacked.

This guide provides step-by-step instructions for the installation of the Supermodified™ motion controller inside standard-sized hobby RC-servos. The instructions provided are straightforward and is **highly recommended** that they are accurately followed to avoid undesirable results and ensure operational fidelity of the end-product. Please note that the installation procedure requires minor mechanical modifications on the servo that will definitely void its warranty, as well as, a number of soldering operations on the Supermodified™ combo boards. If the instructions below are not coherently followed there is always a possibility that your servo or the Supermodified™ motion controller might be irreversibly damaged. For those reasons we strongly recommend that you read through and understand this guide before you begin your installation.

Table of Contents

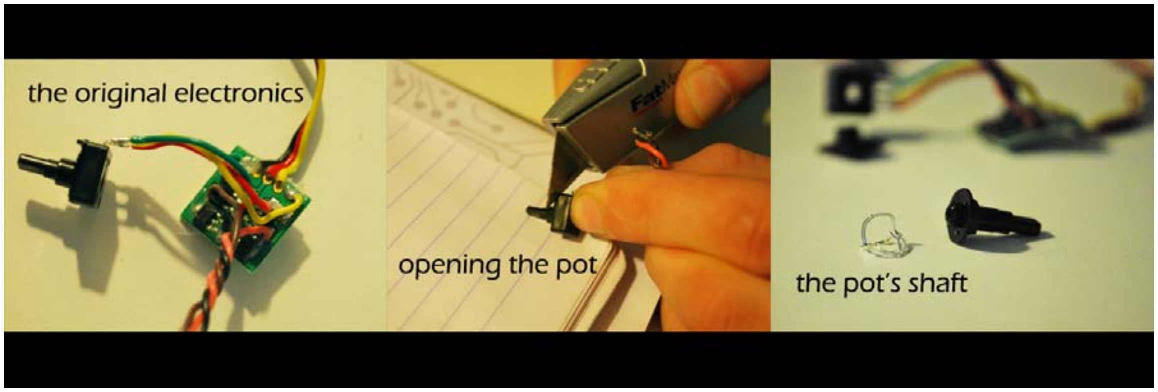
1. Tools & Materials

- Scalpel
- Studding Knife
- Flat file
- Sanding paper or abrasive disk
- Epoxy glue
- Hand or press - drill
- 3.5mm drill bit
- Small piece of wood 10mm thick
- Soldering iron with fine tip
- Electronics vice or Blu-Tack
- Flux
- Solder sucker
- Copper braid
- Flush cutters
- Nose Pliers

2. Preparing the servo

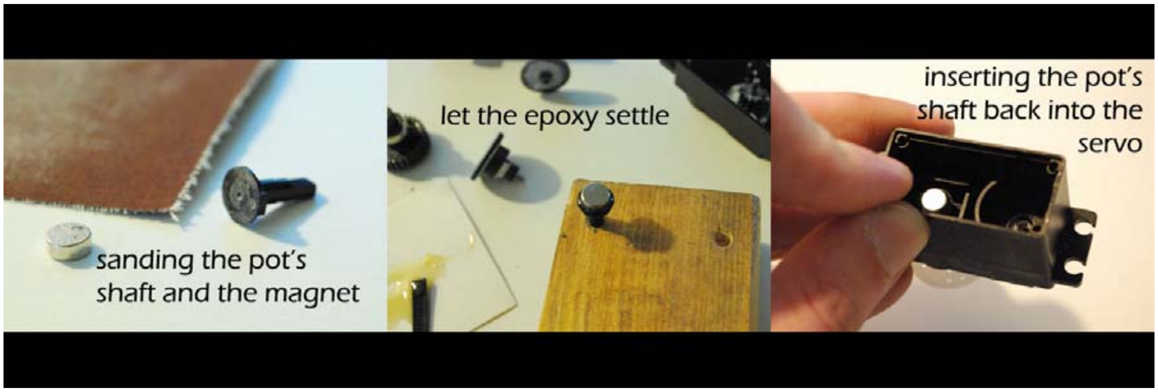
When continuous rotation is required, access the servo's top compartment where the gearbox is located and remove the mechanical stop from the output gear. Take extra care not to damage the gear's teeth as well as, keeping any foreign particles from entering the gearbox assembly. Particles residing on any of the gears' teeth will cause undesirable noise during operation and may also affect the performance of your servo.

Start by accessing the servo's bottom compartment and by removing the control / power electronics from the servo. Unsolder the motor's leads together with the motor chassis grounding (if present) and proceed by removing the feedback potentiometer.



Next disassemble the potentiometer keeping the rotation shaft. Remove the potentiometer slider from the shaft. What remains is going to be used as a support shaft for the encoder's disk magnet. The magnet is going to be glued onto it. Carefully flatten the potentiometer's base using sandpaper or an abrasive disk. Roughen the side of the disk magnet to be glued onto the potentiometer's shaft by using sandpaper. This step is essential and will ensure a strong assembly.

Take a small piece of wood, 10mm thick and drill a blind 3.5mm diameter hole - approximately 5mm deep. This is going to hold the potentiometer's shaft and the disk magnet vertically while the epoxy settles.


sanding the pot's shaft and the magnet
let the epoxy settle
inserting the pot's shaft back into the servo

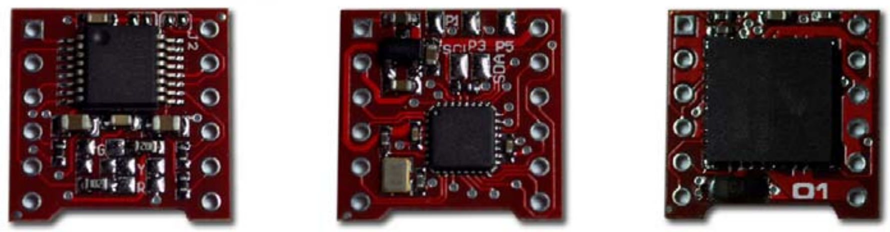
Prepare a small epoxy mix, insert the pot's shaft to the hole and place the disk magnet on the pot's shaft with a drop of epoxy between them. Remove any excess epoxy. Ideally you should aim to leave a small epoxy ring around the circumference of your magnet for lateral support. Remember that your servo has an output speed of about 60 rpm but the system that the servo is going to be installed will most likely be subjected to other sources of vibration. Accurately centre the magnet on the shaft and leave to settle <u>for at least 6 hours</u> (for a 5min epoxy).

We know you cannot wait to start using the Supermodified™ on your project but you must show patience if you do not want to repeat this step in the near future. In any case there is some soldering that needs to be done and now that you warmed-up your fingers and you have to inevitably wait for the epoxy to settle, is a good opportunity to prepare the MagEnc™, PicoMCU™ and MD01™ boards for installation inside the servo.

Remember that when the epoxy settles you have to insert the pot's shaft back into the servo's output gear. Make sure the flatten area of the shaft goes fully into the corresponding 'pocket' of the servo's output gear like it did before the modification.

Notes:

3. Soldering the boards



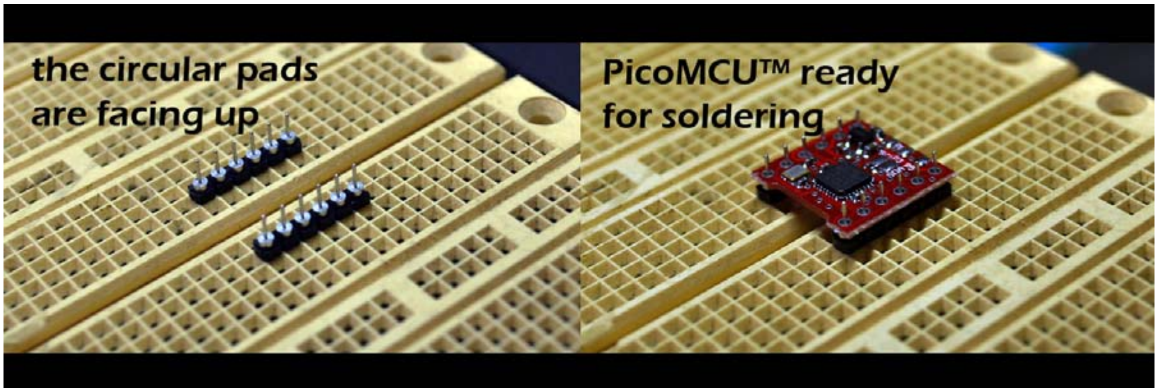
12-bit Magnetic Encoder ATMega 328P @20MHz 5 Amp Motor Driver

Soldering pin headers, jumpers and cables on the Supermodified™ boards, requires concentration and precision. The instructions provided in the guide below will make your life much easier. A fine-tipped soldering-iron should be used since everything is very tightly-spaced on the tiny boards. Flux should also be used to facilitate soldering. In the case of a mistake during soldering, use a solder-sucker and / or copper braid to remove excess solder.

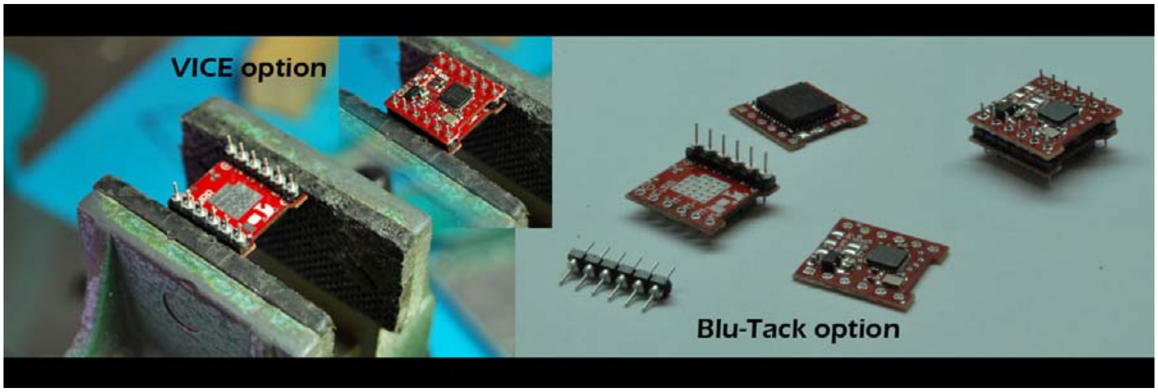
3.1. Soldering board - to - board connectors

We will be soldering 2.54mm pitch, low-profile, board-to-board pin headers and sockets across the three PCBs. Our aim is to stack the PicoMCU™ on top of the MD01™ and the MagEnc™ on top of the PicoMCU™ to form the Supermodified™ combo.

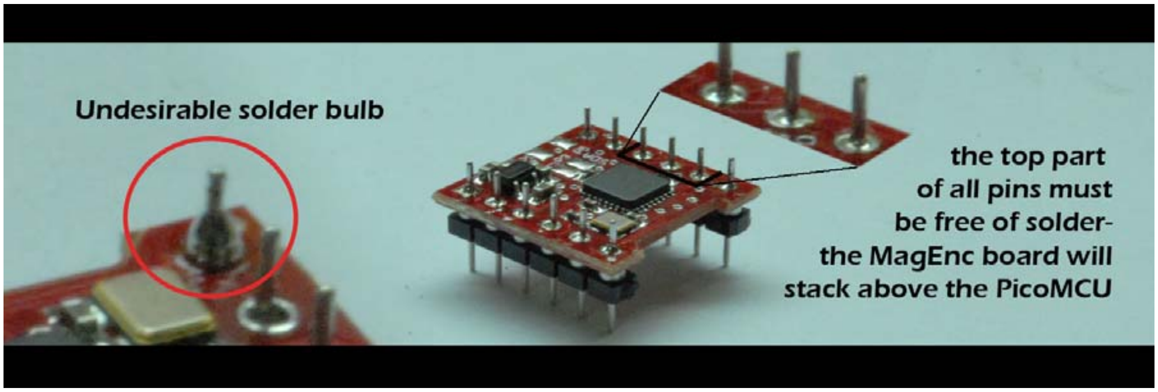
Begin by taking an electronics breadboard, together with two single rows of 6 dual-pin-headers, and locate them in the breadboard facing each other and 4 pins apart, making sure that the circular pads face upwards. We will be using the breadboard to hold the two pin rows in place while we solder them on the PicoMCU™.



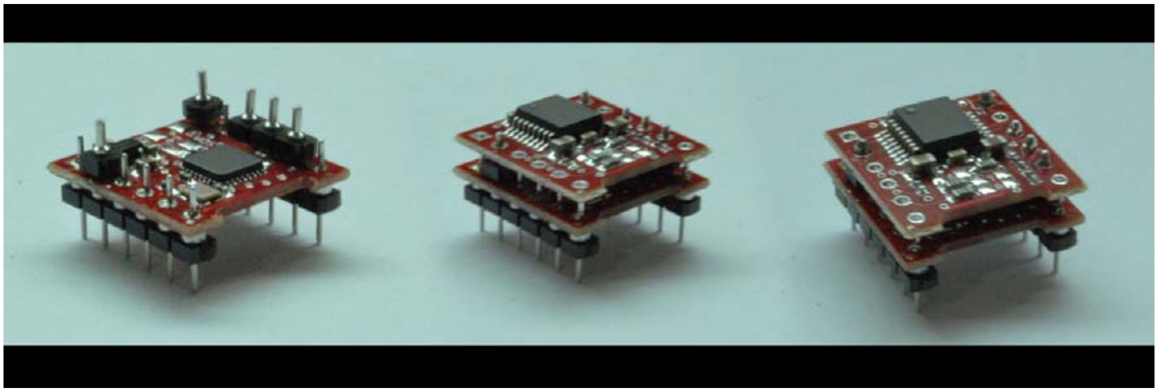
If you do not have a breadboard, use the MD01™ board as a guide. You can mount it carefully on the electronics vice, component face – down, and put the pin-rows into place. Alternatively you can use a piece of Blu-Tack and your bench. Whatever your method the important thing is to have the two pin-rows aligned while they are being soldered on the PicoMCU™



The pins coming out of the PicoMCU™ are utilised by the MagEnc™ board's socket connectors, which stack on top. It is essential to ensure that no solder bulbs occur, especially in pins: **2, 8, 9, 10 and 12**. Therefore when soldering the pin headers on the PicoMCU™ apply flux and use minimum amount of solder for each of the pins.



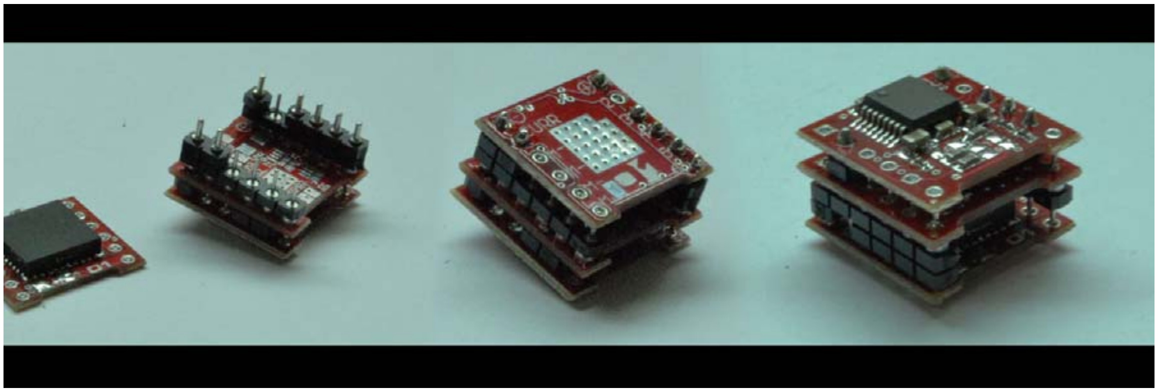
Take the socket connectors (five of: 1 + 1 + 3) and place them in the following top pins of the PicoMCU™: **pin 2, pins 8 to 10 and pin 12**, as shown in the figure below. Place the MagEnc™ board on top, component face-up, in an aligned fashion. Make sure the two boards are parallel from all sides and solder the five socket connector pins onto the MagEnc™ board.



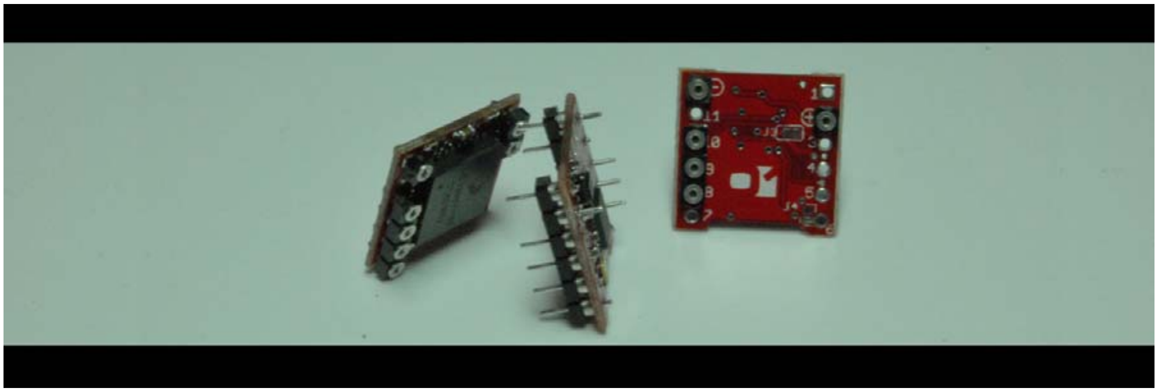
Next we have to solder the MD01™ board's connectors in place. Using the exact same method, this time we will utilise the bottom pins of the PicoMCU™ board to align and solder the MD01™ board's socket connectors.

Take the PicoMCU™ and MagEnc™ boards from the previous step, secure them (components side facing down), on your small electronics vice (or on your bench with a small piece of Blu-Tack) and place socket connectors on the following pins: **pin 1, pins 3 to 6, pins 11 and 12**.

Place the MD01™ board on the socket connectors (also with its components' side facing down), and solder them. Remember to always check that all board edges are kept parallel during soldering in the three board assembly, since this will ensure easy installation inside the servo casing as well as, good alignment of the magnetic encoder board (MagEnc™) with the servo's output gear-shaft.

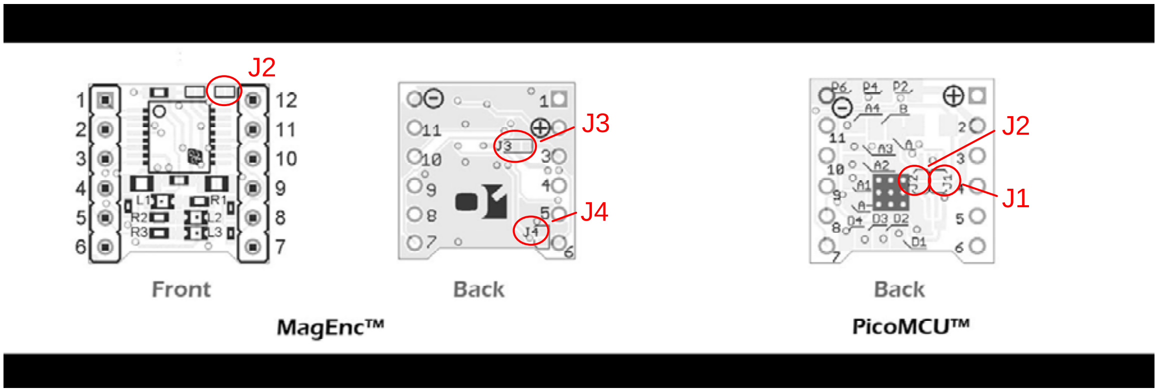


Finally, using the flush cutters cut off all the unconnected pins from PicoMCU™ to the two other boards. Below is an exploited view of the three-board assembly after cutting the unused pins from the PicoMCU™.

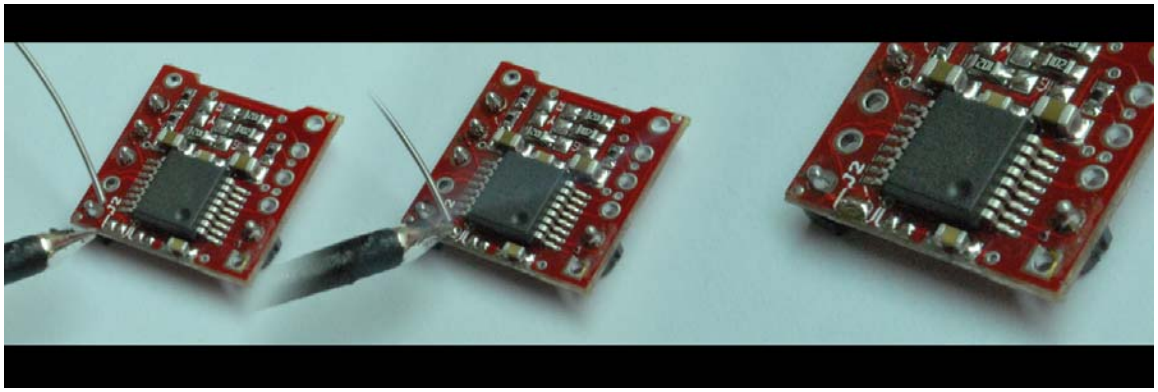


Notes:

3.2. Selecting & setting the appropriate **jumpers**



The MagEnc™ and PicoMCU™ boards both have a number of configuration jumpers. When using these boards in the Supermodified™ combo, some of their jumpers must be closed:



- In the **MagEnc™** jumpers **3** and **4** must be closed. When these jumpers are closed the corresponding pins of the AS5145 are grounded.

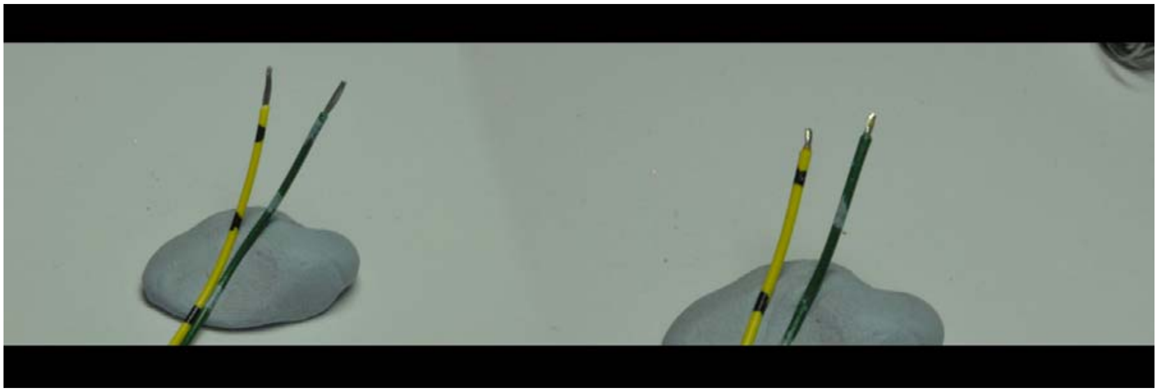
Jumper **3**:	Mode Select	→ *High* Frequency Mode
Jumper **4**:	Zero Position Programming Pin	→ Disabled

- In the **PicoMCU™** there are only two jumpers. When **J1** and **J2** are closed the 5V, UART, TX/RX pins are exposed to pads A, B respectively. When an RS-485 transceiver chip is used, **J1** and **J2** must be left open.

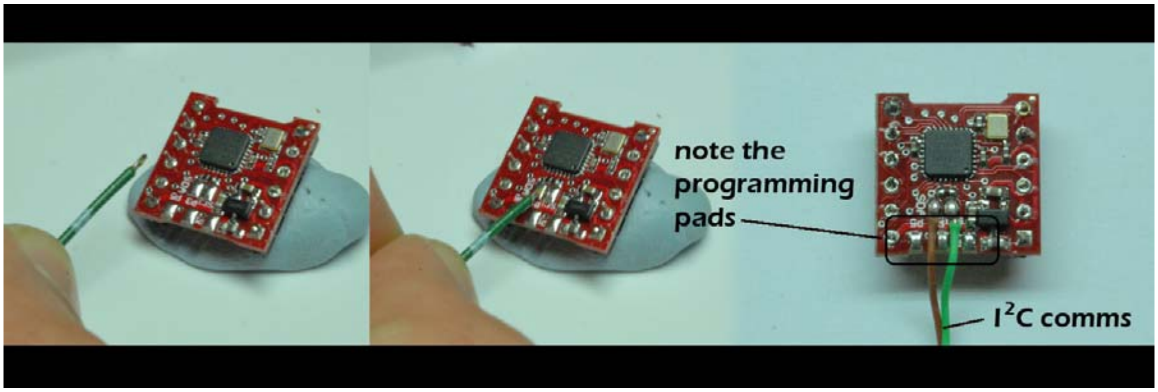
Note: By default the Supermodified™ combo, ships without an RS-485 transceiver fitted.

3.3. Soldering cables and assembling

To successfully solder cables in such confined space one must start by **tinning well** the tip of the cable to be soldered, and then cutting the tinned section of the wire short (3mm max). To hold-down the cables during the process use a small vice or a piece of Blu-Tack.



One of the simplest ways to interface to a Supermodified™ from a PC, must be using an Arduino controller acting as a USB to I^{2}C communications-bridge. Associated files can be found here. The Supermodified™ offers a number of different options for communicating with the outside world. Detailed information on the communication capabilities of the Supermodified™ controller can be found in the *Datasheet*. In this guide we will assume an I^{2}C communications interface.

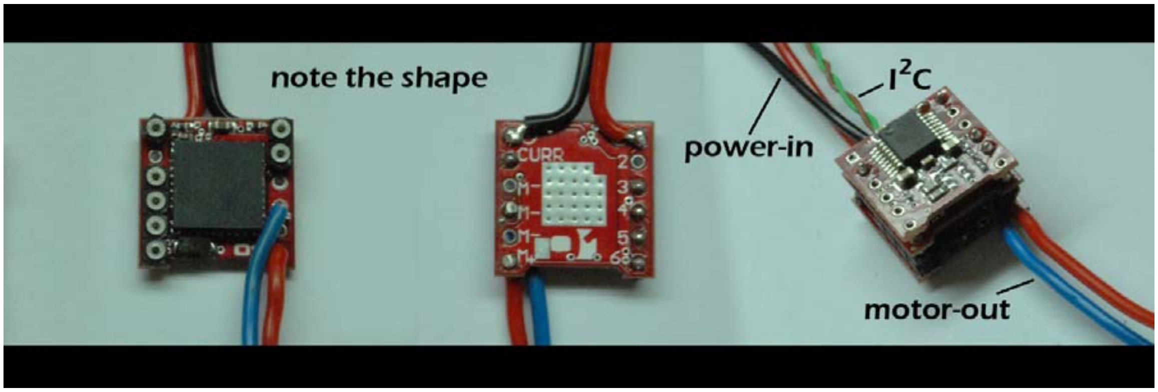


To solder a tinned cable-tip on a PCB pad, first apply flux on the pad and a small quantity of solder. Bring the tinned part of the cable directly in contact with the pad, ensuring comfortable grip. Apply the soldering iron for a short 2 seconds. The example below illustrates soldering cables on the SDA and SCL lines of the I^{2}C bus of the

PicoMCU™. Care should be practised to ensure the two cables (SDA, SCL of the I^2C bus) pass <u>between</u> the programming pads.

The remaining cables solder onto the MD01™ PCB. Signals include the motor connections (M+, M-), the power connections (V_{in}, GND) and the motor shield (GND). More specifically:

Motor	**Power**
Pin **7**: **M+**	Pin **1**: **Vin**
Pin **8**: **M-**	Pin **12**: **GND**
Motor Chassis Pad: **Motor Chassis**	



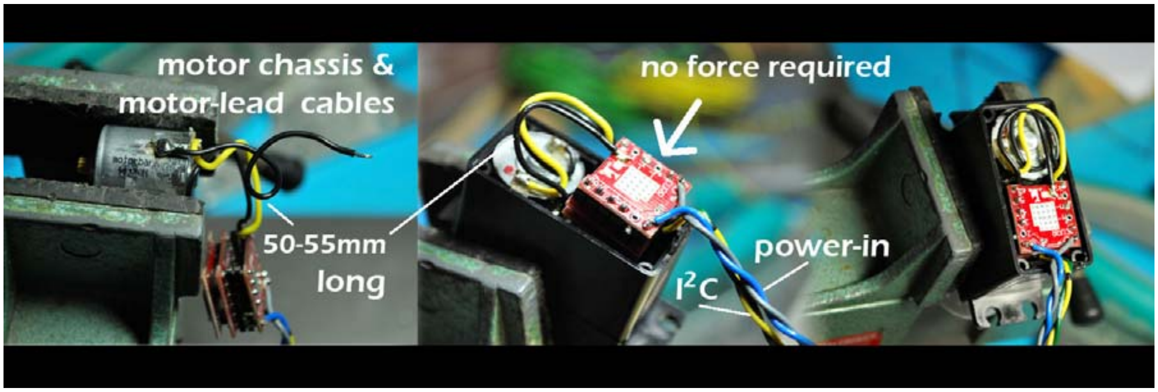
<u>Remember to keep the cables' tips well-tinned and short</u>.

The power-in leads are soldered at the back of the MD01™ board, both facing inwards. This simply makes best use of the available space and facilitates the exit of the cables from the servo.

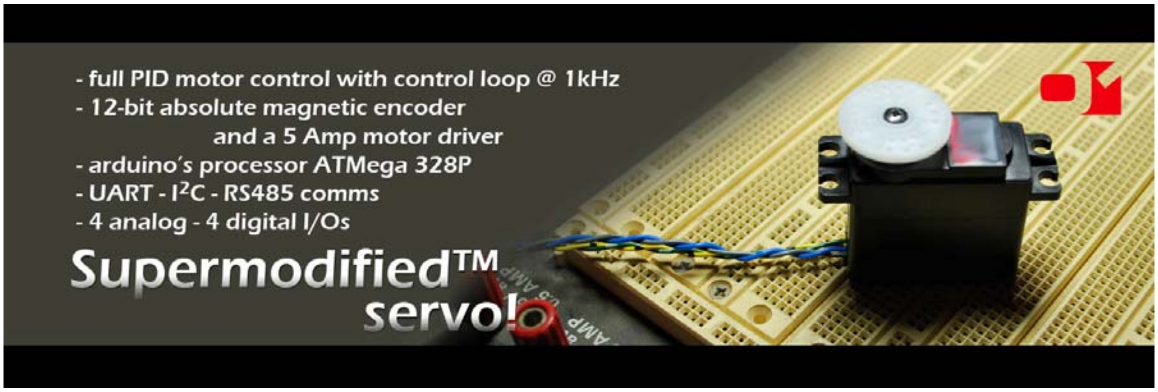
Next, solder the motor chassis grounding cable onto the motor casing. The other end must be soldered on the **motor chassis** pad in bottom side of the MD01™ board. The cable must not exceed 50-55mm in length. The motor-lead cables (M+,M- pins) must also be cut at 50-55mm, tinned and soldered on the corresponding motor lead.

Once all cables in place group the M+, M- and motor chassis cables together and shape them avoiding sharp bends. You will need to make a small cable loop to install the Supermodified™ combo inside the servo.

Note: This guide suggests that you use 50-55mm cables for the motor-to-MD01™ board connections. This is only to offer ease in assembling / disassembling operations. In high-performance coreless servos it might be advantageous to use shorter leads.



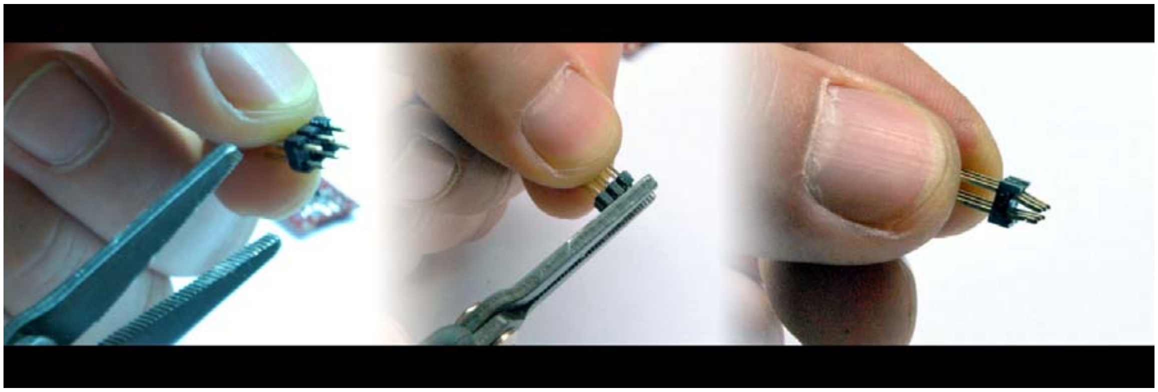
Re-insert the motor in the servo casing by pushing very gently until it engages the gearbox. Exercising care move the Supermodified™ board assembly inside the servo casing with the MagEnc™ face entering first, having the power-in and I^2C communications cables facing the servo's original cable-exit hole. Depending on the size of your cables you might need to extend the size of the servo's cable-exit hole.


- full PID motor control with control loop @ 1kHz
- 12-bit absolute magnetic encoder
and a 5 Amp motor driver
- arduino's processor ATMega 328P
- UART - I²C - RS485 comms
- 4 analog - 4 digital I/Os
Supermodified™
servo!

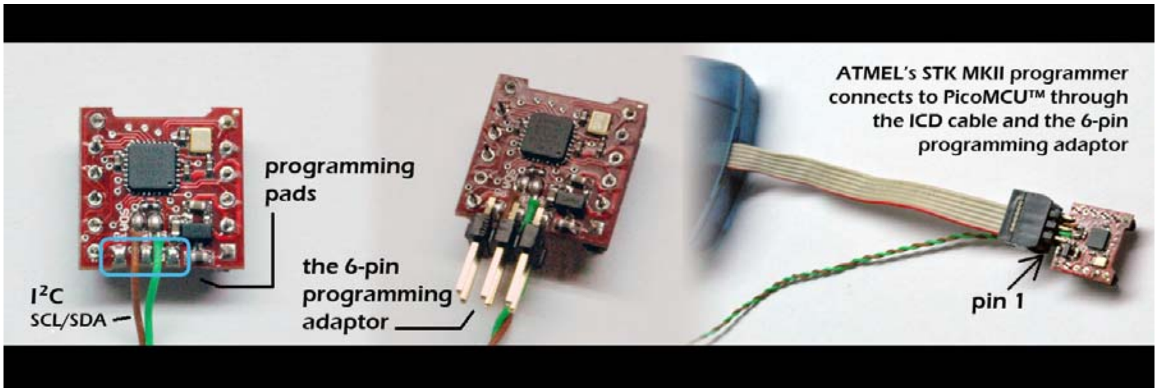
Notes:

4. Programming Connector

The PicoMCU™ board is the 'brains' of the Supermodified™ motor controller. It is built around ATMEL's ATMega 328P one of the most popular MCUs ever. The PicoMCU™ is totally Arduino compatible, meaning you can use the same code and development tools across the two. To flash your code on the PicoMCU™ using a programmer, like ATMEL's ISP MKII, you have to make an adaptor / connector in order to interface to the programmer's ICD cable connector. Simply use a 2.54 mm pitch, dual-row, 6-pin header and a pair of nose pliers.



Use the pliers evenly on the short pin-header end, to bend the pins downwards such that the PicoMCU™ board's programming pads, can clamp in place with the 6-pin header, as illustrated in the picture sequence below. The SDA and SCL cables, if correctly soldered, will align the programming adaptor / connector during insertion operations.


programming pads
I²C SCL/SDA
the 6-pin programming adaptor
ATMEL's STK MKII programmer connects to PicoMCU™ through the ICD cable and the 6-pin programming adaptor
pin 1

Note that the programming adaptor / connector should never be soldered on the PicoMCU™ board. It is to be used exclusively when programming the PicoMCU™ board. The ISP cable female connector from ATMEL's ISP MKII programmer plugs-in directly to the 6–pin header which in turn clamps onto the PicoMCU™ board. Pin 1 of the ICD cable's connector should on the same side and close to the regulator of the PicoMCU™ board. More information on the programming pins of the PicoMCU™ controller board can be found in the *Datasheet*

Notes:

5. Useful Links

- Supermodified™ V2.00 Datasheet.
- Google Code *zo*Supermodified page. Download the latest source and header files for ***01 Mechatronics'*** Supermodified™ motor controller.
- Google Code *zo*avrlib page. C library for Mega AVRs from ***01 Mechatronics***.
- WinAVR 101 - How to use WinAVR with AVR Studio, Visual C++ and Eclipse to program your AVR controller in C. Suitable for Supermodified™ as well as Arduino programmers