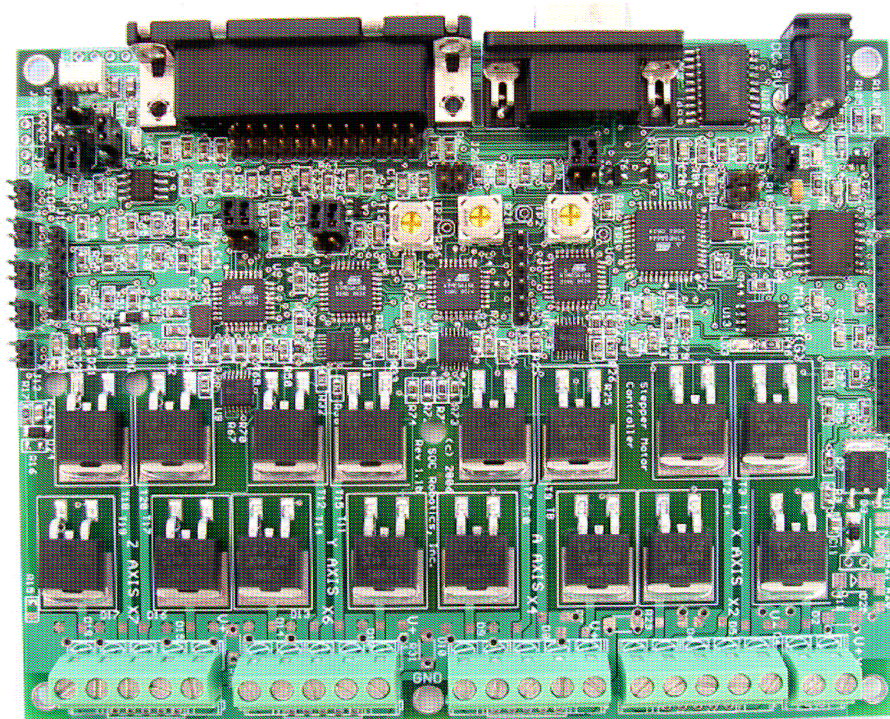


# MC433G Stepper Motor Controller With G Code Option

4 Axis, 10A PWM  
Hardware Reference Guide  
PCB Rev 1.1,1.1b

GStep Software Version 1.64  
Motor Control Software Version 0.95



[www.soc-robotics.com](http://www.soc-robotics.com)

## Warranty Statement

SOC Robotics warrants that the Product delivered hereunder shall conform to the applicable SOC Robotics Data Sheet or mutually agreed upon specifications and shall be free from defects in material and workmanship under normal use and service for a period of 30 days from the applicable date of invoice. Products that are “samples”, “design verification units”, and/or “prototypes” are sold “AS IS,” “WITH ALL FAULTS,” and without a warranty. If, during such warranty period, (i) SOC Robotics is notified promptly in writing upon discovery of any defect in the goods, including a detailed description of such defect; (ii) such goods are returned to SOC Robotics, DDP SOC Robotics facility accompanied by SOC Robotics Returned Material Authorization form; and (iii) SOC Robotics examination of such goods discloses to SOC Robotics satisfaction that such goods are defective and such defects are not caused by accident, abuse, misuse, neglect, alteration, improper installation, repair, improper testing, or use contrary to any instructions issued by SOC Robotics. SOC Robotics shall (at its sole option) either repair, replace, or credit Buyer the purchase price of such goods. No goods may be returned to SOC Robotics without SOC Robotics Returned Material Authorization form. Prior to any return of goods by Buyer pursuant to this Section, Buyer shall afford SOC Robotics the opportunity to inspect such goods at Buyer’s location, and any such goods so inspected shall not be returned to SOC Robotics without its prior written consent. SOC Robotics shall return any goods repaired or replaced under this warranty to Buyer transportation prepaid, and reimburse Buyer for the transportation charges paid by Buyer for such goods. The performance of this warranty does not extend the warranty period for any goods beyond that period applicable to the goods originally delivered.

**THE FOREGOING WARRANTY CONSTITUTES SOC ROBOTICS EXCLUSIVE LIABILITY, AND THE EXCLUSIVE REMEDY OF BUYER, FOR ANY BREACH OF ANY WARRANTY OR OTHER NONCONFORMITY OF THE GOODS COVERED BY THIS AGREEMENT. THIS WARRANTY IS EXCLUSIVE, AND IN LIEU OF ALL OTHER WARRANTIES. SOC ROBOTICS MAKES NO OTHER WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING WITHOUT LIMITATION ANY WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. THE SOLE AND EXCLUSIVE REMEDY FOR ANY BREACH OF THIS WARRANTY SHALL BE AS EXPRESSLY PROVIDED HEREIN.**

### Limitation on Liability

Notwithstanding anything to the contrary contained herein, SOC Robotics shall not, under any circumstances, be liable to Buyer or any third parties for consequential, incidental, indirect, exemplary, special, or other damages. SOC Robotics total liability shall not exceed the total amount paid by Buyer or SOC Robotics hereunder. SOC Robotics shall not under any circumstances be liable for excess costs of re-procurement.

### © Copyright 2008. SOC Robotics, Inc. All rights reserved.

SOC Robotics, Inc. makes no warranty for the use of its products, other than those expressly contained in the Company’s standard warranty which is detailed in SOC Robotics Terms and Conditions located on the Company’s web site. The Company assumes no responsibility for any errors which may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of SOC Robotics are granted by the Company in connection with the sale of SOC Robotics products, expressly or by implication. SOC Robotics products are not authorized for use as critical components in life support devices or systems.

Marks bearing ® and/or ™ are trademarks of SOC Robotics, Inc.  
Terms and product names in this document may be trademarks of others.

1935A-08/00/5M

## Table of Contents

Warranty Statement.....	2
1.0 Introduction.....	5
2.0 Quick Start Guide.....	9
2.1 Introduction.....	9
2.2 Wiring the Stepper Motor Setup.....	9
2.3 Hardware Setup.....	11
2.4 Host Software Setup.....	13
3.0 Detailed Description.....	15
3.1 Introduction.....	15
3.2 Theory of Operation.....	16
4.0 MC433G Hardware Expansion Port Description.....	18
4.1 Introduction.....	18
4.2 Serial Communication Port.....	18
4.3 Parallel Communication Port.....	19
4.4 Joystick and Analog/Digital Ports.....	20
4.5 Main IO Expansion Port.....	21
4.6 Stepper Motor Drive Port.....	22
4.7 Step Sequence Selection .....	23
4.8 DC Motor and Board Logic Ports.....	24
4.9 Limit Switch Input Port.....	25
4.10 Relay Adapter Port.....	25
4.11 Auxiliary Output Port.....	26
4.12 ISP Programming Port.....	26
4.13 I2C TWI Communication Port.....	27
5.0 Programming the MC433G.....	28
5.1 Overview.....	28
5.2 Programming Motor Controller/G Code Processor Flash.....	28
5.2 Updating EEPROM Contents.....	31
5.3 Post Programming Procedure.....	31
6.0 G Code Command Description.....	32
6.1 Overview.....	32
6.2 G Code Commands.....	34
6.3 M Commands.....	35
6.4 Other Commands.....	36
6.5 Host Support Utilities.....	36
6.6 Updating EEPROM Contents.....	39

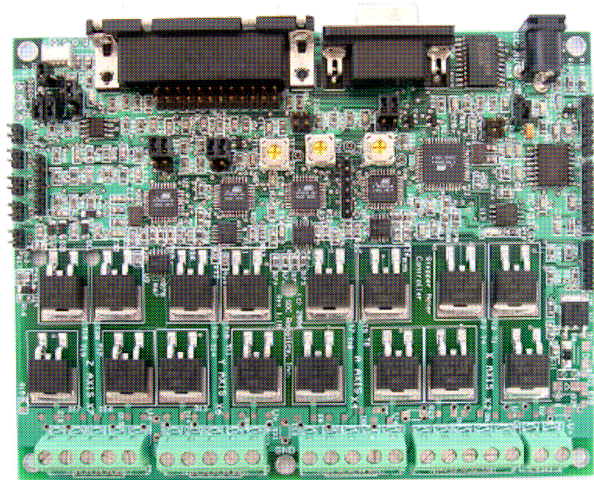
---

6.7 Host Support Utilities.....	40
6.8 Updating EEPROM Contents.....	40
7.0 Electrical and Mechanical Description .....	41
7.1 Component Layout.....	41
7.2 Electrical Specifications.....	42
7.3 Mechanical Dimensions .....	42
8.0 Circuit Schematics.....	43

## 1.0 Introduction

### Features:

- Unipolar Stepper Motor Controller
- 10A continuous, 33A peak
- 4 Axis - X, Y, Z and A
- Wave, Half and Full (High Torque) step
- Variable on time, chop duty cycle and duty cycle slew
- Dedicated 8bit RISC microprocessor for each axis
- Limit switch inputs routed to parallel port
- Stepper motor input voltage from 7-35V
- Noise reduction filter on each input step/dir
- 4 Open NPN outputs
- PC Parallel Port interface
- Logic supply from motor or separate supply
  - o Separate G Code interpreter
  - o RS-232 and RS-485 port
  - o External Serial Flash for program storage
  - o Software controlled step mode selection
  - o Joystick interface
  - o 4 Analog input channels
- Relay daughter card interface
- I2C TWI Communication port
- Onboard programming interface via Parallel Port
- Works with most Windows based G Code programs
- 9-12VDC power input (5VDC board)
- Small form factor (3.85x3.00 in)



### Overview

The MC433G is a 4-axis unipolar stepper motor controller with an onboard G Code processor. The G Code processor accepts G Code and configuration commands via a RS-232 serial communications port (DB9) and converts these commands into individual step and direction signals that are fed to the four stepper motor control processors. The G Code processor also monitors limit switch inputs, joystick inputs and sets auxiliary outputs based on commands received via the serial port. The G Code processor is in control of each motor controller and command sent on the parallel port are ignored. The MC433G can be converted to a standard MC433 by sending a disable command via the serial line. The disable command is the two characters "cm". On power up the MC433G outputs a signon message on the serial channel.

### Hardware

The MC433G is a 4-Axis unipolar stepper motor controller that converts step/direction inputs to jumper selectable wave, full or half step 4-phase power control. Each of the four phase coils of a unipolar stepper motor is connected to an IRF540 Power MOSFET switch. The IRF540 is capable of 10A continuous current load with the ability to peak at 33A. Each phase is pulse width modulated (PWM) to allow operation over a 7 to 35 volt range. PWM adjusts average current delivered to the stepper motor to better match the resistance/inductance of the motor without the need for load resistors. This ensures maximum power is delivered to the motor over a range of load conditions and motor types. PWM parameters are adjusted using three potentiometers - initial on time, chop rate duty cycle and duty cycle slew rate verse step rate. Chop rate is fixed at 12KHz. On time is the amount of time a switch remains on before PWM operation takes over - this allows the current to build to a maximum level as quickly as possible. Duty cycle is the percentage of on-time to off-time at a given chop frequency. Duty cycle is used to set the average current delivered to the motor. Duty cycle slew rate adjusts the duty cycle based on the duration of a step. As the step rate slows down (motor turns slower) the duty cycle is adjusted to deliver less power to the motor thereby maintaining constant current. The PWM potentiometer settings

are described in more detail in section 3.2. Each motor can be individually set to wave, full, half step mode or be disabled by setting two jumpers on a four pin header. An onboard programming interface allows new Motor Controller programs to be loaded into Flash

A dedicated 8bit 20MHz RISC microprocessor (ATmega168) controls each axis. The microprocessor monitors step/direction inputs, limit switch input and PWM settings to generate the appropriate step sequence (wave, full or half) based on jumper settings. Three potentiometers set PWM operation for all four axis. The control processor automatically reduces current to the motor after 4 seconds by reducing the duty cycle and turns the motor off after 8 seconds – these parameters can be adjusted.

The MC433G can be controlled via a PC compatible DB25 parallel port interface connector by disabling the G Code processor using the “cm” command. The Parallel port interface has four Step/Direction inputs, four auxiliary outputs and five limit switch inputs. The four auxiliary outputs are connected to a Relay Adapter connector and an open NPN control connector. Limit switch inputs are connected to four 0.1” pin headers with pull up resistors - pulling the limit switch input to ground deactivates a stepper motor.

An I2C TWI communicate port allows an external TWI controller to communicate with each motor control processor individually to set step mode, step, PWM parameters and turn off conditions.

The MC433G has a 24pin header connector (J18) for attaching an optional G Code control processor. The optional high performance G Code processor supports USB 2.0.

## Host Software Applications

The MC433G is designed to work with a variety of third party G Code software programs. Step input for each axis requires an active high pulse lasting at least 2useconds – step inputs are edge triggered. A high level on the Direction input will turn the stepper clockwise or counterclockwise depending on how the motor is connected to the drive port. The direction signal is sampled after the step trigger so it must remain stable for at least 4useconds after the step trigger.

## Control Processor Reprogramming

New programs can be loaded into each control processors on chip Flash using the onboard programming interface. The programming interface is enabled by installing jumper J30. Once the programming jumper is installed all subsequent signals sent to the MC433G via the parallel port are interpreted as programming commands. Host programming program MC433Prog.exe is used to load new programs into the MC433. See the section on control processor programming for more information.

## Expansion Options

The MC433G has several expansion options: MC5 Relay Adapter, CJ14 Joystick Controller, CJ34 Smart Joystick controller, DRO4 LCD display, DRO84 LED display, MC-USB USB 2.0 Communication Controller and a Higher speed G Code Processing Adapter – MA48/SAM48.

The Relay Adapter provides four isolated contact closures for control of external devices. Power can be optionally supplied by the adapter if isolation is not required using a jumper. The relay board supports low voltage DC switching relays capable of handling 2A at 24volts.

The MA48/SAM48 is an ARM7 processor based daughter card that attaches to the expansion header on the board and converts G Code commands transmitted to it on a USB 2.0 connection

into step/direction commands. The MA48/SAM48 is described by a separate document.

## Differences Between Rev 1.1 and Rev 1.1b

The MC433G was upgraded to PCB Revision 1.1 and then 1.1b.

Summary of Revision 1.1 changes:

1. On board programming interface added – separate ISP10 programming adapter is no longer required. Use MC433Prog.exe to program each processor
2. 74HC08 added to each motor control processor step output to lower chop load in the processor
3. Emergency Stop (EStop) interface added – an external panic stop button can be used to stop all four motor controllers. EStop is routed to the parallel port interface.
4. A few routing errors detected on Rev 1.0 were corrected.
5. Filter circuit and 5V Zener added to each Limit Switch/EStop input to protect these inputs from spurious voltage levels inducing in switch wiring.
6. Circuit allowing each motor controller to measure motor voltage added
7. 20MHz crystal added to allow the motor controllers to run at 20MHz rather than 8MHz

Summary of Revision 1.1b changes (in addition to Rev 1.1):

- External Serial Flash pad added to allow G Code program storage on board.
- 74HC08 QFN replace with TSSOP package for increased manufacturing yield.
- A few routing errors were corrected.

Revision 1.1b PCB is now the current release.

## Software Release Status

### Motor Control Software

Current software release version is V0.95.

Next release V0.97 is scheduled for release December 2008.

V0.97 will support the following new features:

- EEPROM based configuration parameter storage.
- EEPROM configuration image can be uploaded and downloaded using MC433Prog.exe.
- PC based script files converted into hex files by MC433Convert.exe loadable in EEPROM.
- Each motor axis PWM settings can be set individually allowing different motors on each axis
- Automatic motor power reduction and shut off set by the user
- Limit and EStop switch enable/disable
- I2C interface for remote communication with each motor control processor

### GStep Control Software

Current software release is V1.64.

Next release V1.75 is scheduled for release December 2008.

V1.75 will support the following new features:

- TWI communication interface with pass through mode
- Support for TWI peripherals such as the smart joystick, LCD display, LED display, MC-USB 2.0 communication controller
- On the fly stepping mode change with automatic resolution change
- Automatic home find



## 2.0 MC433G Quick Start Guide

### 2.1 Introduction

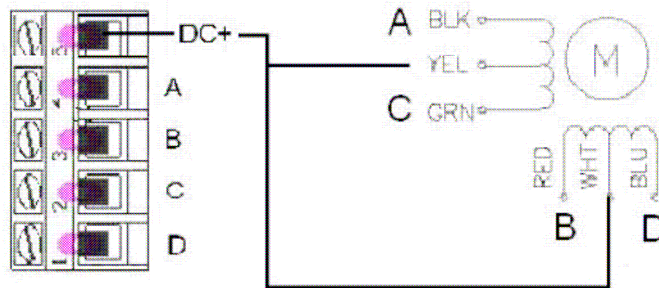
Before using the MC433G please read this quick start guide.

**WARNING: Before connecting Motor DC Power make sure the stepper motors are wired correctly. A Unipolar stepper motor has two sets of windings each with a common lead. The common leads should be connected to DC+ and each motor coil connected to a MOSFET power switch A,B,C or D. Depending on how the windings are connected the motor will turn clockwise or counterclockwise. Ensure the motor leads are securely attached – intermittent or loose connections can generate a several KiloVolt back EMF spike in the motor possibly damaging the motor and/or MC433.**

### 2.2 Wiring the Stepper Motor

The wiring of a Unipolar Stepper motor to the MC433G is shown in the diagram below. The color code used in the diagram is for a SOC Robotics SM2006A 2.5A Unipolar Stepper Motor and may not match the color code of your particular stepper motor – check the phase (coil) configuration diagram for your stepper to determine the correct lead connection to A,B,C and D before attaching the stepper to the MC433G. Do not connect or disconnect stepper motor leads while the MC433G is generating a step pattern – always turn the DC Motor Power Supply Off first. The inductive kickback of the motor coils during disconnection can generate several kilovolts, which may cause an arc failure in the motor or personal injury. Also never touch the open leads of the stepper motor while they are in operation.

Typical 6 Wire Stepper  
Motor Wiring Configuration  
(Color codes may vary)



Never run the steppers without an inline fuse between the DC Motor Power supply and the MC433G – Power MOSFETs can fail and when they do they typically fail in a shorted state. Ensure there is a good ground between the DC Motor Controller, Board Logic Power and the controlling PC – the MC433G is a non-isolated design. All attached equipment must be at a common ground potential. Do not defeat the ground lug of all AC cords.

Unipolar Stepper motors can be attached to the MC433G one of two ways:

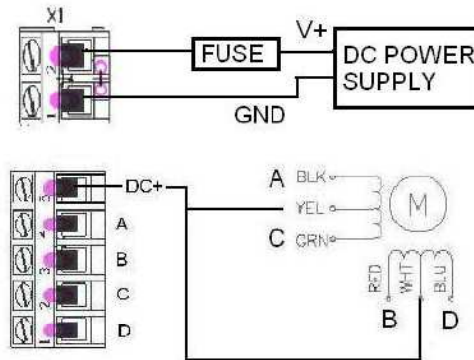
- Common Motor DC Connection – Configuration A.
- Independent Motor DC Connection – Configuration B.

**Common Motor DC Connection**

For low amperage steppers (<2-3amps) the Common Motor DC connection configuration can be used. For this configuration DC Motor Power is connected to screw top terminal X1 where it is then routed to the DC+ terminal of each stepper motor connector (see the diagram below).

Configuration A - Common DC Motor Power with inline fuse for 6 Lead Stepper Motor.

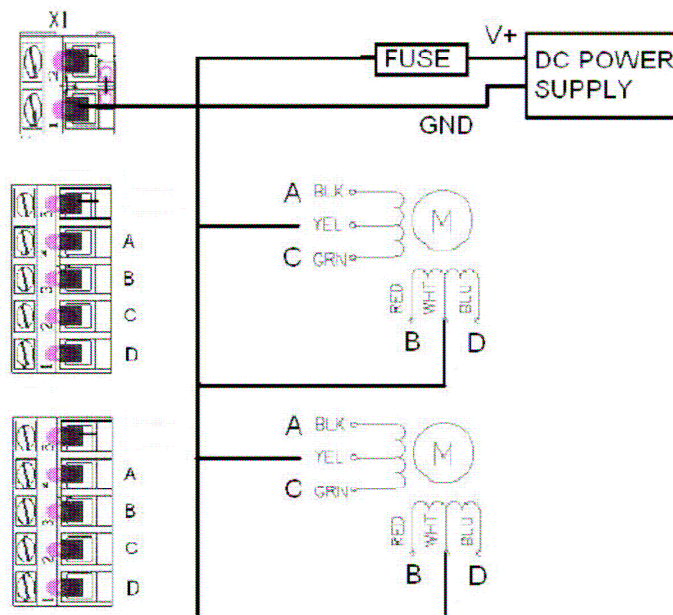
(Color codes may vary)



**Independent Motor DC Connection**

High amperage steppers should be wired using Configuration B shown below. The screw terminal connectors used on the MC433G are rated at 10A and are not designed to handle higher current. The Rev 1.1b PCB has a common ground hole located between the A and Y Axis connectors – a lead should be run from this hole to earth ground. Never run the MC433G without an inline fuse.

Configuration B - Recommended wiring diagram for high amperage Stepper Motor attachment.



## 2.3 Hardware Setup

The MC433G must be configured for correct operation. The stepper motors may not turn reliably if this setup procedure is not completed. There are two steps – first the stepper motor PWM characteristics must be tuned then the G Code processor preference settings must be adjusted.

The following procedure will guide you through the tuning of the stepper motor controllers. If you have a 10A multi-meter connect it between the Motor V+ lead and the power supply. Start with the board unpowered:

1. Set the step sequence for each motor – wave, full or half step by installing jumpers on connectors J24, J25, J26, J27. The unit is shipped with Wave step selected (no jumpers installed). Wave turns one phase on at a time, Full turns two phases on and Half turns one or two phases. It may be best to leave the default setting to ensure the motors are wired correctly.
2. Attach a serial cable to DB9 and your PC.
3. Start a serial communication application such as HyperTerminal and set the baud rate to 115,200, no parity, 8 data bits and 1 stop bit.
4. Connect a DC power adapter to the Board Logic input (9-12VDC – center positive – reverse polarity protected). The initial turn on state of the MC433G is all motors off.
5. The following sign-on message should appear on your terminal screen:

```
MC433 Stepper Controller - Gstep V1.64
Copyright 2007, SOC Robotics, Inc.
Type H for help menu - Control mode active
```

-

The '-' characters is the command prompt – commands are entered after the prompt and executed when the RETURN key is pressed. Make sure Control Mode is active. The sign-on message displays the state of Control mode.

6. Now enter the '!' command to start the stepper motor test function – the following output should appear:

```
-!
Test Mode (Enter 'e' to exit)
Commands:
  <axis>[c|w]<step rate/sec>
  Example: Xc200 Yw1500 Zc700 <CR>
>
```

This function can drive a single axis or all four at the same time. If you want to drive the x axis stepper motor at 50pps (pulse per second) enter the following command:

```
>x200<CR>
>
```

The step and direction signals will be sent to the x-axis stepper motor controller continuously until another command is entered or test mode is exited.

7. Turn potentiometer R27 (on time) fully counterclockwise (zero on time) then turn clockwise half a turn. Turn potentiometer R28 (duty cycle) fully counterclockwise then turn clockwise half a turn (50% duty cycle). Turn potentiometer R36 (duty cycle slew rate) fully counter clockwise. This sets on time at half, duty cycle slew rate to the minimum setting and duty cycle to 50%.
8. Make sure jumper J30 (programming enable) is not installed.

9. Connect the DC Motor Main power to X1. The DC Motor Power connector is located at the top of the board with DC+ near the board top. If 5V is available this can be used for initial testing before connecting 12 or 24V.
10. Connect a unipolar stepper motor to the Motor X Axis connector. This is the first connector next to the DC Motor Power connector on the top right of the board if the DB25 connector is on the left. An easy way to verify correct motor connection is to send a one step per second pulse to the motor and connect motor leads in different configurations until the motor steps reliably in one direction (see SOC Robotics web site for setup software).
11. Now enter a step command in the test function. If the motor is connected correctly it should turn at the set step rate. If it doesn't or is jerky then the wiring is not correct or the power supply is insufficient.
12. Disconnect motor.
13. Repeat steps 6 to 8 for the other three axis.
14. If 5V was used in the steps above connect 10-35 volts to Motor Main power connector.
15. Attach a motor to the X Axis.
16. Send a continuous low speed step command to the controller (<25pps). Adjust initial on time, duty cycle with the slew pot untouched for the most reliable step operation.
17. Now enter a fast step rate of about 1200-1500pps and adjust the slew pot slightly to ensure the motor runs reliably. Slew compensation boosts high speed current while leave low speed current at the same level. This adjusts compensates for increased Back EMF current at high stepping speed. If you have a multimeter make sure the current at high speed is close as possible to the current at low speed. Be careful not to deliver too much current at slow speeds.
18. If the step mode is changed the potentiometers may need to be adjusted again for reliable operation.
19. Once the potentiometers have been adjusted they shouldn't need adjusting again.

See the MC433 Hardware Technical Manual for setup procedures using the parallel port.

Now the G Code processor preference settings must be adjusted. The G Code processor assumes that the resolution of each axis is 0.00025in per step, there is no backlash, maximum rapid move is 20ipm and step direction polarity is P N N N. See the G Code Processor command section to understand what these setting are. The preference setting menu is entered by entering the "se" command:

```
-se
Entering Preference Mode
Type 'le' to leave - dp to display preferences - ss save settings
=
```

For see the current preference settings enter "dp".

```
=dp
Preference Settings:
Baud rate = 115200
Max Feed rate (mf)= 20
Resolution [X,Y,Z,A] (rr)= 0.000125 0.000125 0.000125 0.000125
Backlash [X,Y,Z,A] (bl)= 0 0 0 0
Direction polarity [X,Y,Z,A] (pl)= P N N N
=
```

Now change any preference settings by entering a two character preference selection command followed by the new values. For example to change the resolution from 0.00025 to 0.000125 enter the following command:

```
=rr0.000125 0.000125 0.000125 0.000125
=
```

Changed preferences are not saved in EEPROM until the “ss” command is entered.

The MC433G should now be ready to use. To move the X axis 1 inch enter the following G Code command at the prompt:

```
-G1 X1.00 F10.0 <CR>
-
```

## 2.4 Host Software Setup

Desktop application software sends G Code commands and/or MC433G configuration commands to the MC433G via the serial line. If commands are sent to the controller to fast the serial receive buffer will overflow so commands must be allowed to complete before sending the next command.

The MC433G expects a low to high transition on the Step input to generate a step. The Direction input causes a clockwise step or counterclockwise step depending on how the motor is wired to the drive. The direction input is sampled after the step is activated so it must remain unchanged for at least 5useconds after the step low to high transition. Step and direction inputs can be changed at the same time. Host software must the set the Direction line high or low and cause a low to high transition on a step line to cause a step. The Direction and Step inputs are pulled high with a 10K resistor. Noise reduction logic in the motor processor eliminates false steps due to noise.

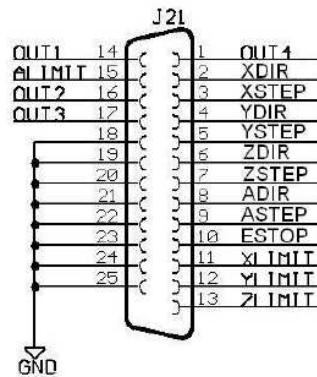


Figure 2-1. Parallel port signal function assignment.

Limit switch inputs are pulled high with 10K resistors – an external contact closure will pull the limit switch input low indicating a limit has been reached. Each motor processor monitors the limit switch input pertaining to its stepper motor and automatically stops the motor without requiring input from the host. Host software should interpret a low level on a limit pin as an indication that a limit has been reached.

Four Auxiliary outputs are available for host software to control. Each Auxiliary output is routed to two different connectors: Relay adapter connector J17 and open NPN transistor output connector J18. A high level on an OUTX pin sends a high level to the relay port and causes the appropriate transistor to conduct pulling the output lead to ground on J18. Host software must be configured accordingly to control external devices.

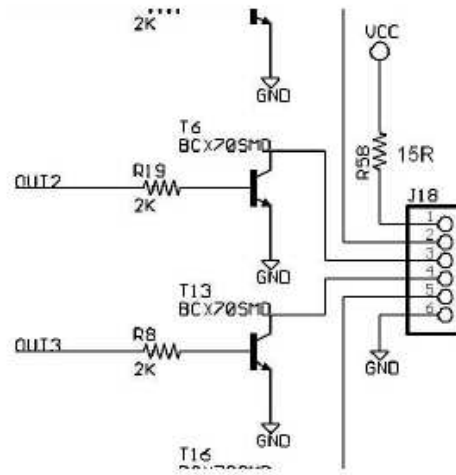


Figure 2-2. Typical open NPN Auxiliary Output circuit on J18.

### 3.0 Detailed Description

#### 3.1 Introduction

The MC433G Stepper Motor Controller is a 4-Axis unipolar stepper motor controller that uses four high performance 8bit 8MHz RISC processors to regulate the power delivered to each stepper using Pulse Width Modulation (PWM). Wave, Full and Half step operation is selected on an individual motor basis using two jumpers. PWM initial on time duration, duty cycle and duty cycle slew rate is controlled by three potentiometers eliminating the need for load resistors. The processors monitor step and direction inputs, step sequence jumpers and limit switch input to determine how and when to control the Power MOSFETs controlling power delivery to each motor phase. After 1.5 seconds of step inactivity the duty cycle is automatically adjusted to reduce unnecessary power consumption and motors are turned off completely after 20 seconds. Although PWM operation can be changed to delivery significant power to the motors it can not be turned off completely at slow step rates in order to protect the motors and drive circuit from overheating. Four auxiliary outputs are available to control external devices - both 5V TTL levels and open NPN transistor outputs are available. Limit switch inputs are routed to the Parallel Port interface and each motor processor.

The MC433G accepts an optional expansion adapters: Relay adapters. The Relay Adapter is controlled by the four auxiliary outputs and provides isolated relay closure for external 12 or 24 volt devices.

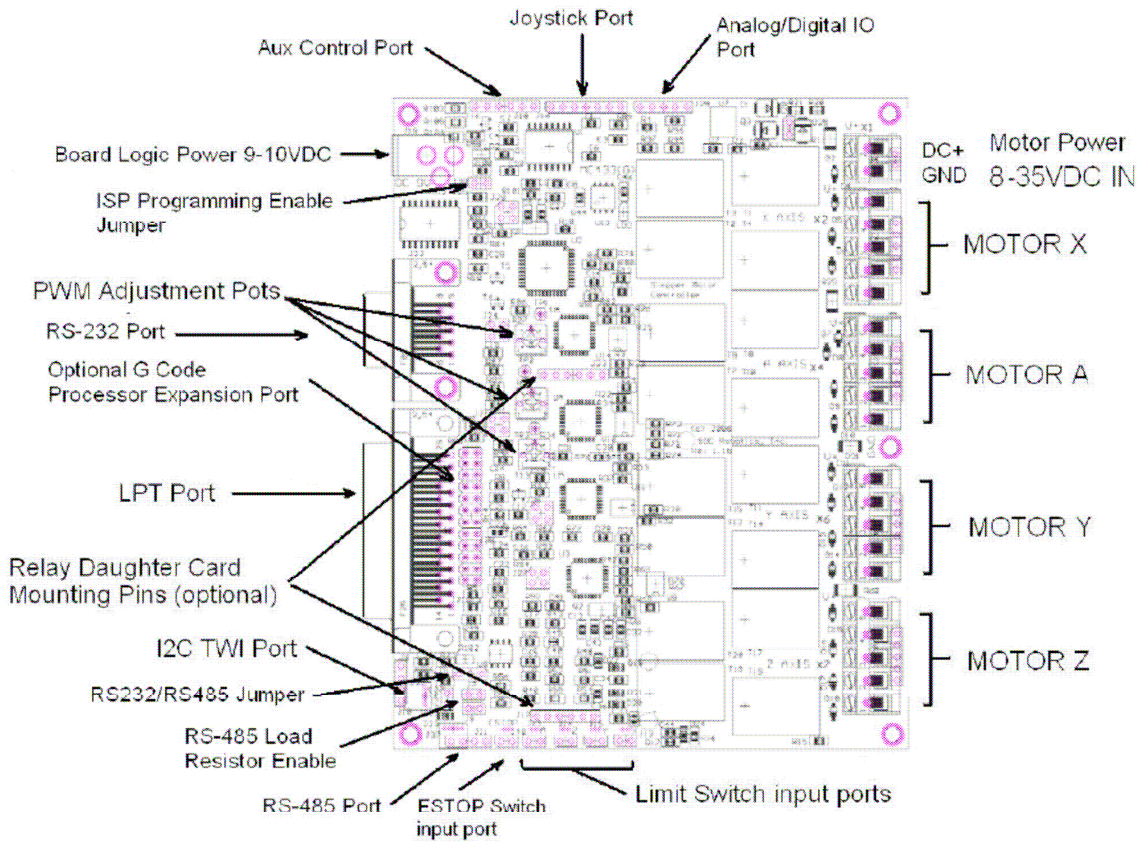


Figure 3.1. MC433G Top Side Connector Layout.

### 3.2 Theory of Operation

The MC433G is designed to control unipolar stepper motors. A unipolar stepper motor is usually supplied with six wires having two center tap leads. A typical wiring configuration is to connect the center leads (YEL and WHT in the picture below) to the DC+ supply and the remaining four leads to four power transistors (MOSFETS in the MC433) that are in turn connected to ground. By turning the power transistors on and off in the correct sequence the motor steps in the clockwise or counterclockwise direction.

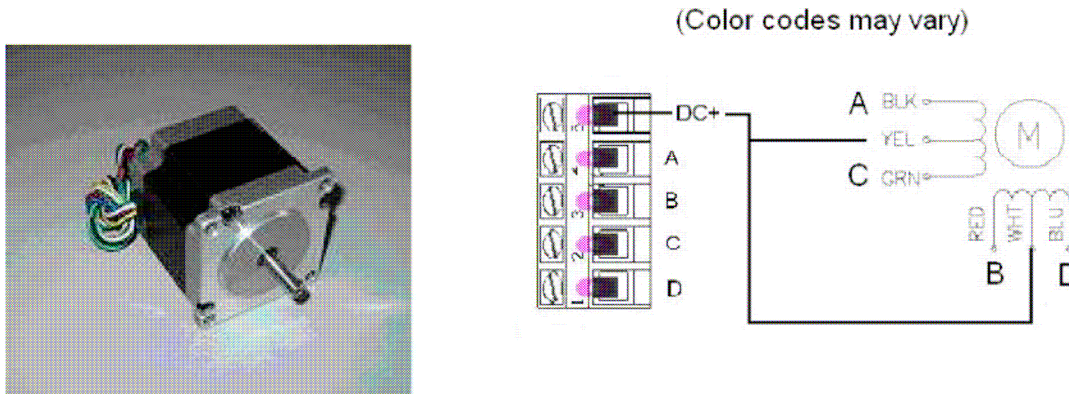


Figure 3-2. Typical Unipolar Stepper Motor and winding diagram.

The MC433G implements three different step sequences: Wave, Full and Half step. In Wave stepping each winding or phase is turned on one at a time in sequence while the other phases are turned off. In Full step two phases are turned on at the same time in the correct sequence. Full step operation develops more torque while consuming more power. Half step operation turns one or two phases on at a time generating a half step with a drop in torque. By varying the current to each phase it is possible to microstep a unipolar stepper motor while trading off even more torque. The MC433G does not support microstepping.

The MC433G generates the following step sequences for Wave, Full and Half stepping.

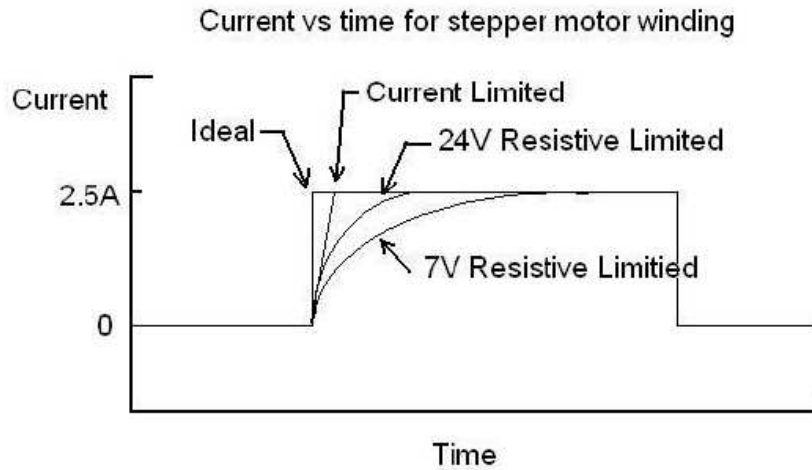
WAVE STEP				FULL STEP				HALF STEP			
A	B	C	D	A	B	C	D	A	B	C	D
1	0	0	0	1	1	0	0	1	0	0	0
0	1	0	0	0	1	1	0	1	1	0	0
0	0	1	0	0	0	1	1	0	1	1	0
0	0	0	1	1	0	0	1	0	0	1	0
								0	0	1	0
								0	0	1	1
								0	0	0	1
								1	0	0	1

1 - Motor Phase Powered - MOSFET ON  
 0 - Motor Phase Not Powered - MOSFET OFF

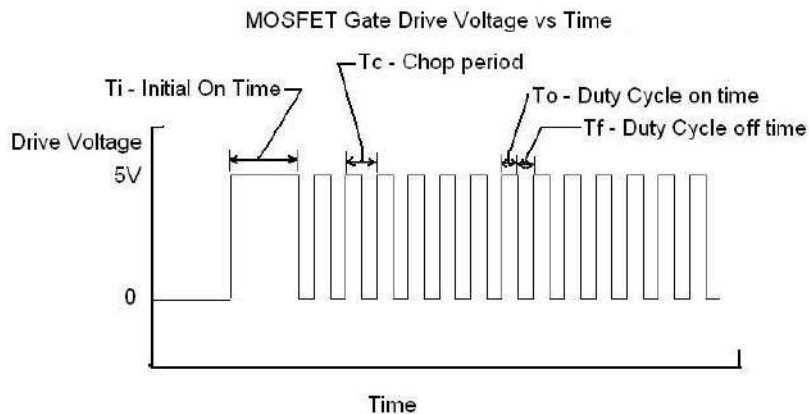
Unipolar stepper motors are available in various step resolutions (with 1.8 or 0.9 degree per step being the most popular) and torque/step speed ratings. High speed steppers usually have lower winding resistance and inductance so they reach maximum current (and hence torque) quickly. Winding inductance determines how quickly the stepper reaches maximum current - the higher the voltage across the windings the faster this current level is reached. Voltage can only be



increased so much, however, before the current rating of the motor is exceeded. One way to limit current is to use series resistors but this also increases the time constant of the motor counteracting to some extent the benefit gained from increasing the voltage. Once the motor reaches maximum rated current if some form of current limiting could be applied an ideal trade off between operating voltage and maximum performance is reached. This is accomplished by turning the voltage on and off rapidly at a rate sufficient to keep the motor at a constant current state - Pulse Width Modulation accomplishes this. The figure below shows the trade off between the different approaches.



The MC433G is designed to operate in a constant current mode by regulated the power delivered to the stepper using PWM. The MC433G uses an open loop PWM approach. The user is responsible for adjusting the PWM parameters for best performance. PWM parameters are changed by adjusting three potentiometers. The potentiometers adjust initial on time, chop duty cycle and duty cycle slew rate. The chop frequency is fixed at 12KHz - duty cycle ranges from 5% to 95% or off at high step rates. Initial on time  $T_i$  ensures each winding reaches maximum current in the least time. Chop Duty cycle sets the current while chop slew rate changes the duty cycle as step rate slows down to reduce current to the motor at low step rates.



## 4.0 MC433G Hardware Expansion Port Summary

### 4.1 Introduction

The MC433G has a PC Parallel Port interface for step/direction input for of the four stepper motor axis, four Limit switch inputs and four Auxiliary control outputs. Expansion connectors for an optional Relay Adapter, G Code IO processor and I2C TWI communication port are also on the MC433.

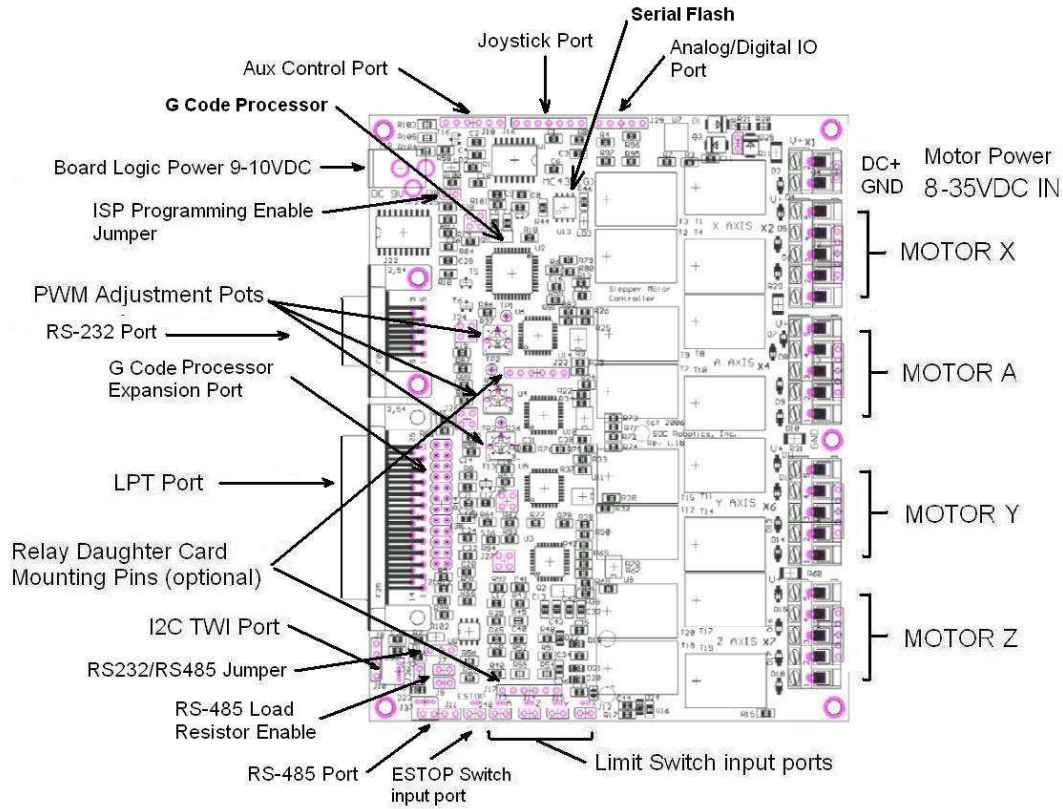
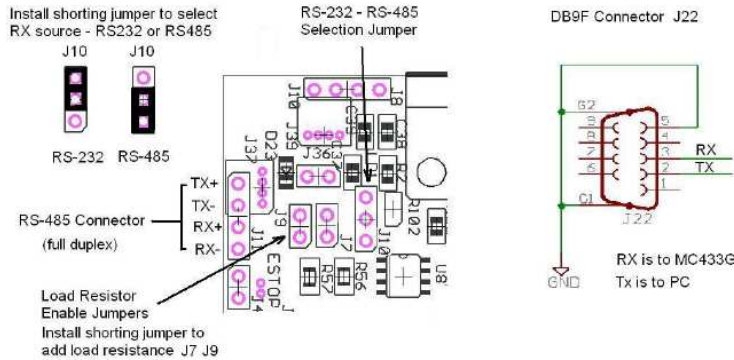


Figure 4-1. MC433G Expansion Connector locations.

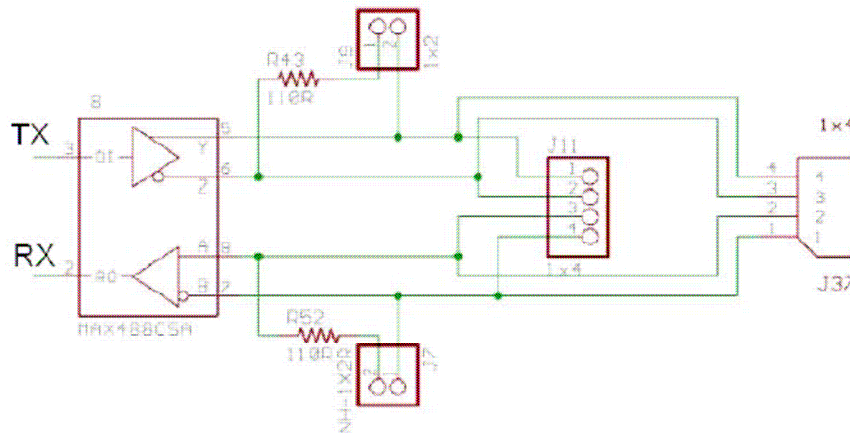
### 4.2 Serial Communication Port – RS-232 and RS-485

The MC433G has two shared serial communication interfaces – RS-232 and RS-485. Default baud rate is 115,200, no parity, 8 data bits and one stop bit. The RS-232 serial communication port is a female DB9 connector with MC433G RX on pin 3, TX on pin 2 and GND on pin 5. The RS-485 connector is a 0.1" pin header with signals shown in the diagram below.



The RS-485 circuit is full duplex with optional load resistors that can be enabled by installing supplied shorting jumpers on J9 and J7.

RS-485 Circuit showing Load Resistors



MC433G TX is connected to both the RS-232 and RS-485 level translator. The source of RX must be selected by installing a shorting jumper on J10. The diagram on the previous page shows the correct installation of the shorting jumper.

### 4.3 Parallel Communication Port DB25

The MC433G Parallel Communication port (J21-DB25) is the main interface for step/direction input, limit switch detection and aux output control. J21 is compatible with the standard PC parallel port.

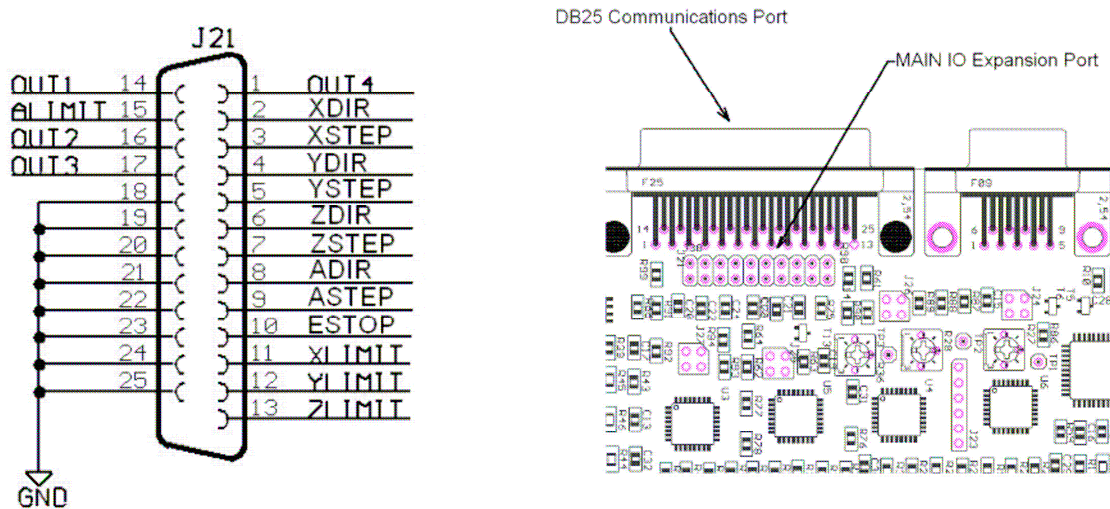


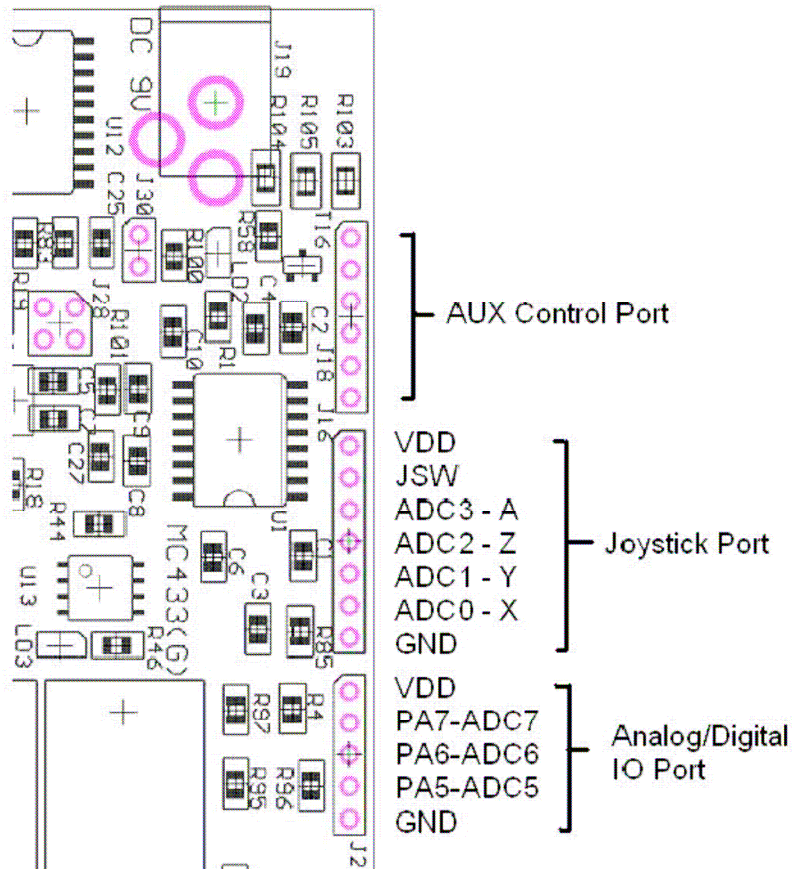
Figure 4-2. Parallel Port DB25 Connector J21

### 4.4 Joystick and Analog/Digital IO Ports

The Joystick port can be used to attach an optional 4-Axis joystick/switch box to allow independent movement of all four axis without host control. The G Code processor monitors these inputs and moves each axis based on the value supplied by the joystick position. The absolute position of each axis is maintained as the steppers move under joystick control so all movements are correct for the next G Code command. Switch JSW is used to enable and disable recognition of the joystick port - this is a normally open momentary action switch to toggle state. A command can also be sent on the serial port to enable or disable joystick recognition. If a serial port command enable joystick recognition then the switch becomes a trigger to generate current absolute axis position for transmission back to the host control software. This allows the joysticks to move all four axis for setup and alignment purposes. By default the joystick interface is disabled.

The Analog/Digital Port provides three additional signal pins for general purpose use. Each signal of the Analog/Digital IO port can be configured in software as either an analog input pin or a digital input or output pin. See the section on Other Commands for a description of the command sequence used to setup these pins. These pins have 10K pull ups on them so they can be used as switch closure inputs. If the pins are used as analog inputs then resistors R95, R96 and R97 must be removed. For early generation MC433G boards resistor R59 must also be removed if ADC3 - A axis joystick is used.

Joystick and Analog/Digital IO Ports



### 4.5 Main IO Expansion Port

The Main IO expansion port (J38 - 20pin 0.1" pin header) provides a connection point for the addition of a high performance G Code processor that can drive the step inputs directly without host support. The optional G Code processor accepts G Code command via RS-232, RS-485 or USB.

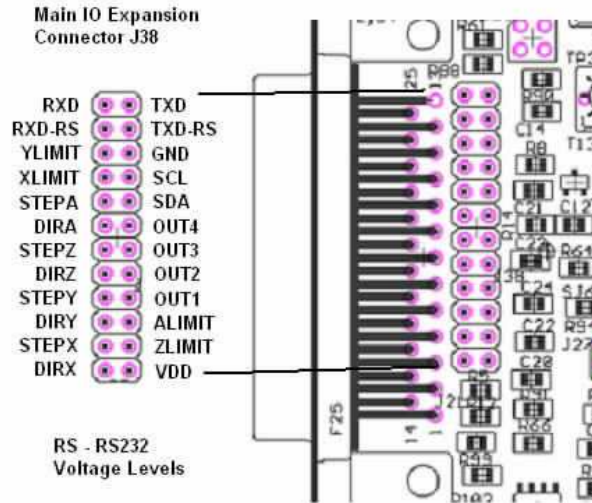


Figure 4-3. Main IO pin assignment.

### 4.6 Stepper Motor Drive Ports

The MC433G has four stepper motor drive ports. Each motor is connected to a five terminal screw top connector. The top connector of each motor port provides DC+ to the motor. The other four connectors connect to each phase of the motor. DC+ is connected to the common center lead of each phase pair (see diagram below). A,B,C,D leads are controlled by four 33A Power MOSFETs. By pulling each of these inputs low a motor phase is energized.

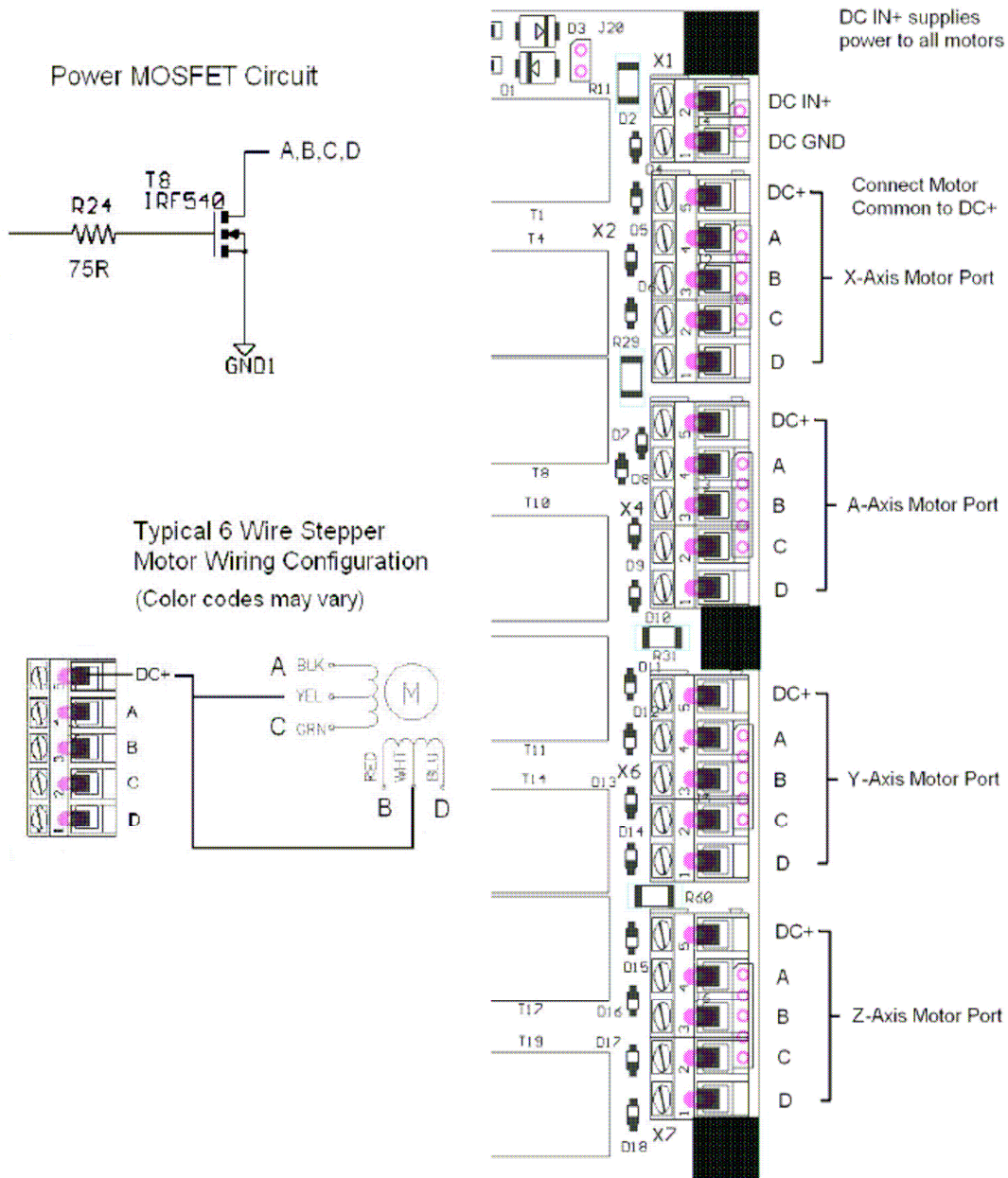


Figure 4-4. Stepper motor output drive ports and Motor Power In.



### 4.8 DC Motor Power Connector and Board Logic Power Connector

The MC433GLogic requires an unregulated DC input source between 9-12V DC. A wall mount DC power adapter is recommended. Power Jack should be center tap positive. A reverse polarity protection diode protects the on board voltage regulator from damage if the polarity is reversed. Although the board has provision (jumpers) to source logic power from the DC Motor Power source this is not recommended as inductive voltage spikes from the stepper motors may damage the on board 5V Logic.

The DC Motor power connector is a screw top connector. The DC Motor does not have reverse polarity protection.

**CAUTION: DO NOT CONNECT THE DC MOTOR POWER INPUT INCORRECTLY OR THE BOARD MAYBE DAMAGED.**

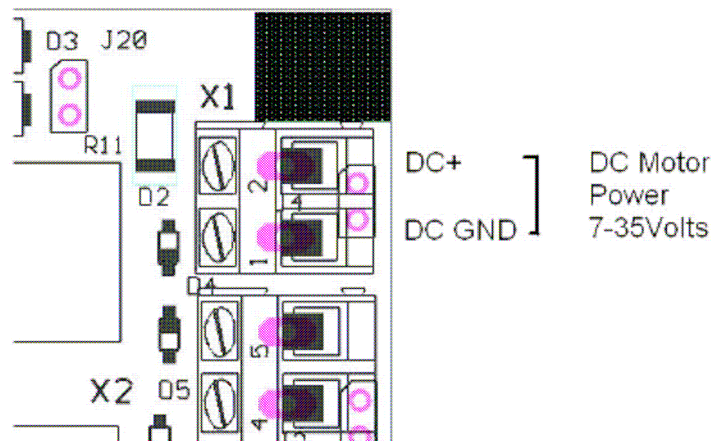


Figure 4-6. DC Motor Power Connector.

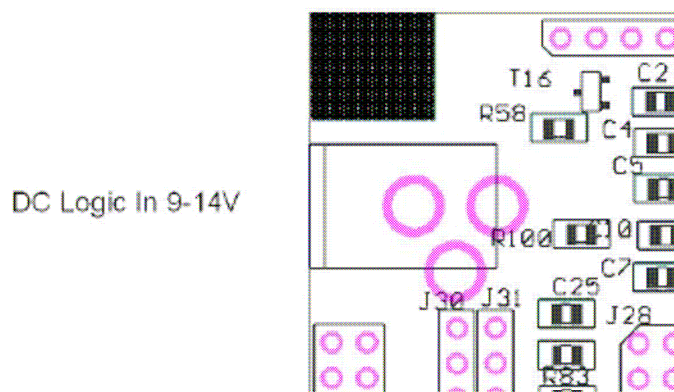


Figure 4-7. Board Logic Power Connector.



### 4.9 Limit Switch Input Ports

Each motor has a limit switch input circuit. The Limit detection logic input line is routed to the DB25 and each motor control processor. A Limit event is activated when an external switch pulls the level to ground through a contact closure. Each Limit Switch input is pulled high with a 10K resistor.

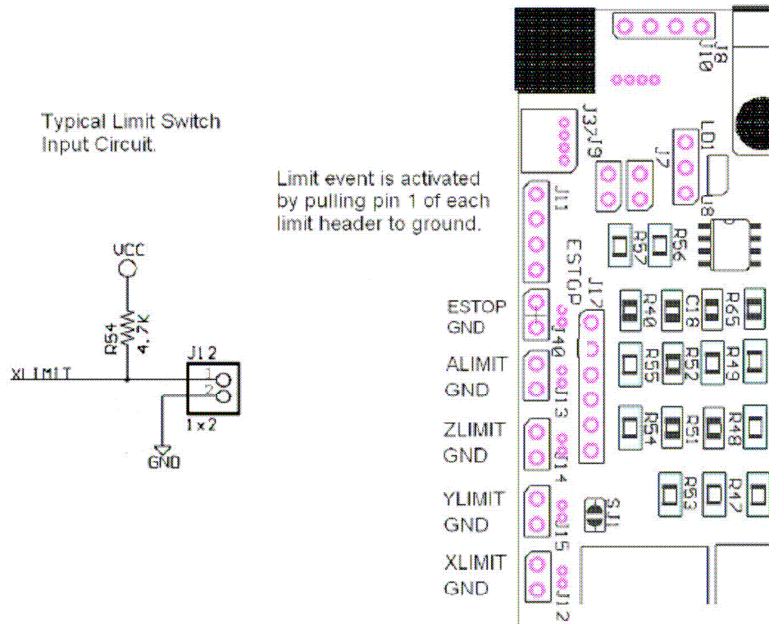


Figure 4-8. Limit Switch input ports.

### 4.10 Relay Adapter Port

The Relay Adapter port connectors J17 and J23 provide an attachment point for an optional Relay Adapter board. Connector J17 and J23 support the Auxiliary Output and Power/IO signals respectively. The Relay Adapter board pulls power from either VDD (Board Logic 5V supply) or DC IN (unregulated power supplying the Board Logic 7-14VDC). Signals SDA/SCL are the I2C communication lines connected to each motor control processor. Signals OUT1-4 are controlled via the Parallel Port connector.

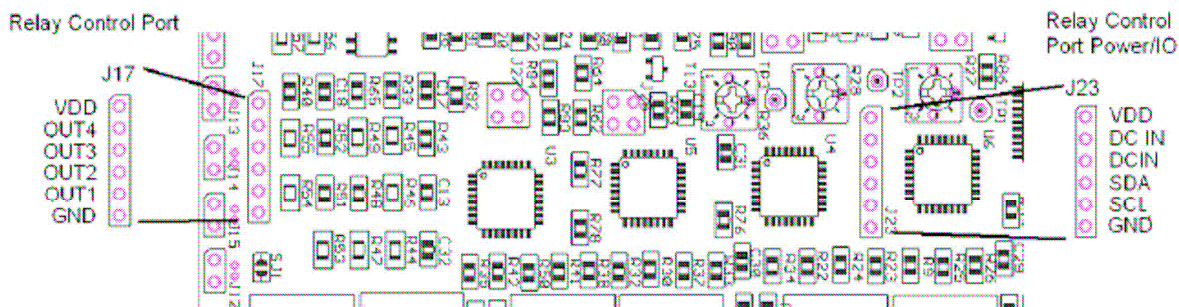


Figure 4-9. Relay Adapter connectors.

### 4.11 Auxiliary Output Port

The Auxiliary output port can be used to control external devices via open NPN transistor circuit. Setting AUX OUT X high causes the respective transistor to conduct pulling the external circuit to ground. A current limited DC output (5V with 10ohm series resistor) is provided on pin1 to power external devices. Maximum pull down current is limited to 100ma.

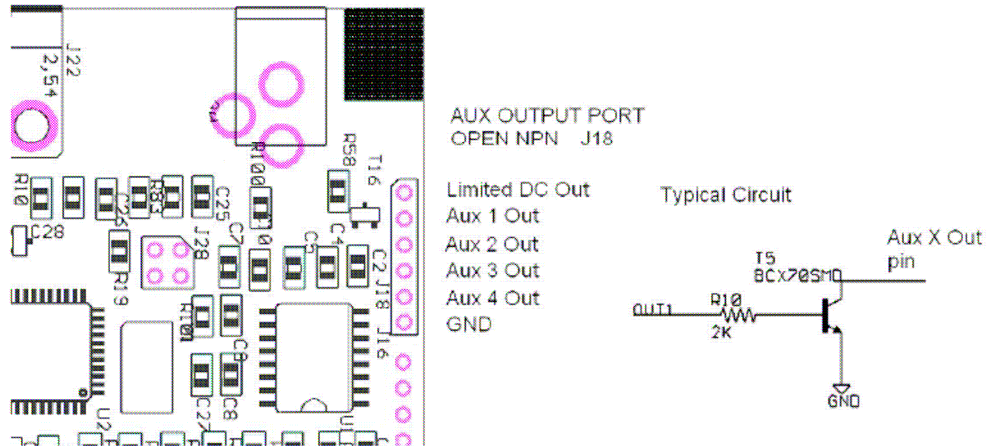
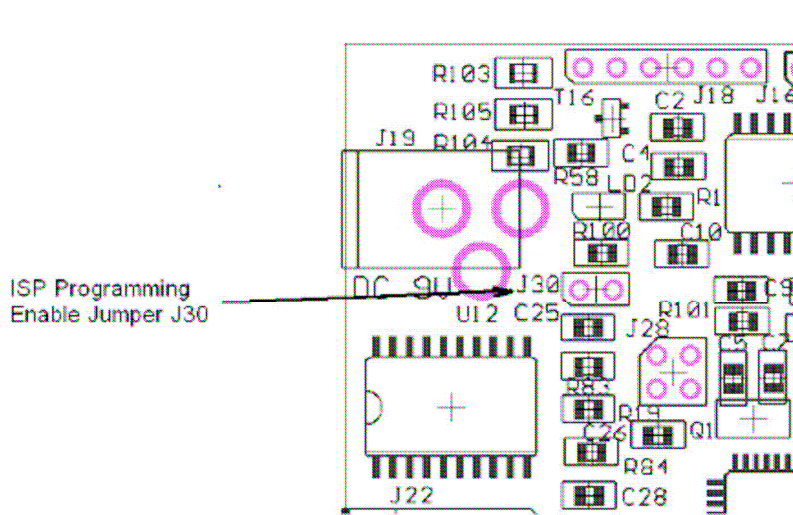


Figure 4-10. Auxiliary Output Port and drive circuit.

### 4.12 ISP Programming

An ISP programming circuit is now built onto the board. Installing a shorting Jumper on connector J30 enables ISP programming. New programs are loaded into Motor Controller Flash using programming utility MC433Prog.exe (available from the SOC Machines web site). **SHORTING JUMPER J30 MUST BE REMOVED BEFORE SENDING STEP/DIRECTION COMMANDS TO THE MC433G– FAILURE TO REMOVE THIS JUMPER WILL RESULT IN THE MOTOR CONTROLLERS BEING REPROGRAMMED TO AN UNKNOWN STATE AND POSSIBLY DAMAGED.**



### 4.13 I2C TWI Communications Port

The MC433GI2C TWI Communication port is connected to the I2C port of each for the four motor control processors. I2C (called Two Wire Interface TWI by Atmel) is a bidirectional communications protocol developed by Philips to allow embedded processors to communicate with each other at a 400KHz rate. Through the TWI port an external processor can communicate with each of the control processors to set PWM parameters, step the motors, change step sequence (wave, full or half step), change motor turn off conditions, change chop frequency and request status. The TWI port allows the characteristics of each motor controller to be tailored to the specific requirements of the stepper motor. The protocol used between the MC433G and an external TWI controller is described in a separate document.

Using the TWI port it is possible to step the motors, change from full step to half step and back on the fly and change PWM characteristics in real time to suite the specific needs of the application. The optional ISP-USB adapter (sold separately) provides direct control of the MC433G via a PC's USB port.

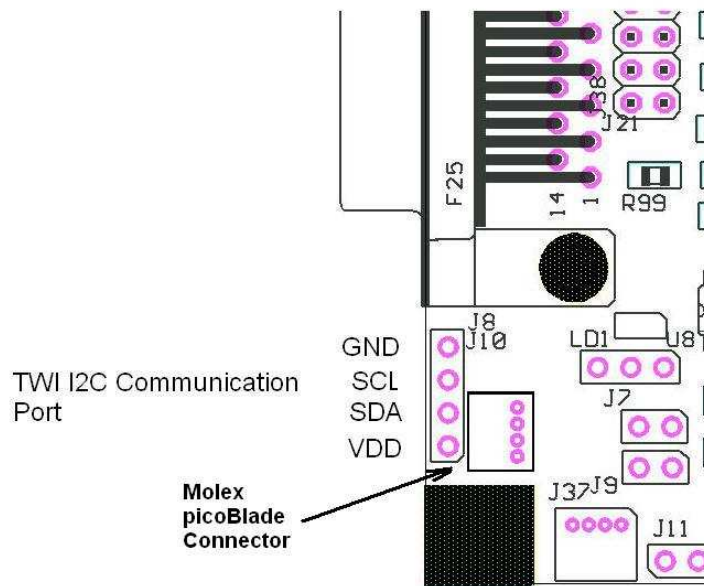


Figure 4-12. TWI I2C Communication Port.

## 5.0 Programming the MC433G

### 5.1 Overview

The MC433G can be programmed one of two ways:

- Re-Program Motor Controller Flash using MC433Prog.exe.
- Re-Program GStep Controller Flash using MC433 Prog.exe.

Programming Motor Controller Flash overwrites the contents of the current Motor Controller Flash with a new program. A new program may add new features, fix bugs or update the main PWM control algorithm. Flash contents can only be changed by enabling Flash Programming Mode by installing a shorting jumper on connector J30 and downloading a new program using utility **MC433Prog.exe**. Changing Motor Controller contents must be done carefully to avoid damaging the AVR processors. Flashing the Motor Controller erases the content of EEPROM.

**CAUTION: Once programming is complete the programming enable jumper must be removed. If the jumper is left in place and step/dir commands are sent to the controller the motor controllers will be reprogrammed with incorrect information and possibly damaged.**

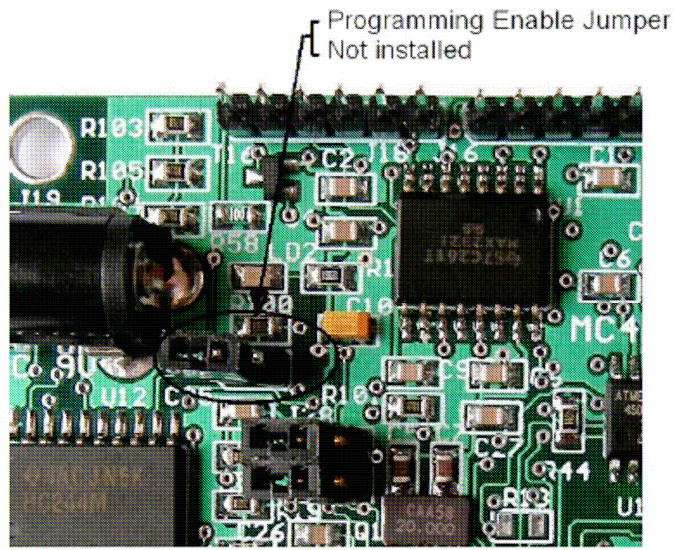
The soon to be released Motor Control Software Version 0.97 loads control parameters from on chip EEPROM. Updating EEPROM parameters such as chop rate, chop frequency, automatic shut off, etc is done by modifying the contents of Motor Controller EEPROM. Changing EEPROM contents requires loading a new EEPROM image using MC433Prog.exe. Flash Programming Mode shorting jumper.

Before attempting to Re-Flash the controllers or change EEPROM content turn motor power off. The programming procedure may turn on all Power MOSFETs - drawing significant power from the motor power supply.

### 5.2 Programming Motor Controller Flash

The MC433G Rev 1.1 and Rev 1.1b PCB's have a programming interface on the board. The Flash contents of the motor control processors and G Code processor can be re-programmed through the parallel port interface. Rev 1.1 and 1.1b are programmed the same way - there is no functional difference between Rev 1.1 and 1.1b PCBs except for the G Version - Rev 1.1b has an additional SPI Serial Flash device for on board G Code storage.

To enable Programming Mode the supplied shorting jumper must be installed on connector J30. J30 is located next to the DC Logic Power connector (see picture below). To download a new control program to the processors use programming utility **MC433Prog.exe** only - do not use any other programming utility. **MC433Prog.exe** is available on the SOC Machines web site on the MC433G product page. As soon programming mode is enabled all commands sent to the parallel port are interpreted as programming commands - sending any other command such as step/dir sequences will probably damage the control processors. Utility **MC433prog.exe** implements the AVR ISP programming protocol. Use of any other program or programming utility may damage or lock the processors from further programming. Remove the program enable jumper immediately after programming is complete. Do not remove or attach the parallel port cable with the programming enable jumper installed.



**MC433Prog.exe** is a command line program running under Windows98, 2000 and XP. The program emulates the operation of **ISProg.exe** (an AVR ISP programming utility) and is used to download new Flash and/or EEPROM files to the Motor Controllers and G Code Processor. **MC433Prog.exe** can be used to program each of the four processors individually or (using a command file) perform a sequence of programming functions on all four motor controllers in one pass.

New motor control programs and G Code programs are provided as Intel Hex files. New code releases will always include a master program configuration text file. **MC433Prog.exe** has been tailored to program the MC433 and MC433G only – do not use this utility to program AVR processors with the ISP10 programming adapter – use **ISProg.exe** instead.

An MC433G has five AVR processors identified as X1,X2, X3, X4 and X5. Each of these processors is either an ATmega168. The correct code must be loaded into each processor. The relationship between X number and motor axis is as follows:

Processor ID	Motor Axis
X1	G Code Processor
X2	Z
X3	A
X4	Y
X5	X

To select a specific processor enter the correct processor identifier at the command prompt – X1,X2, X3, X4 or X5 – subsequent programming commands are then sent to the selected processor. Note these identification numbers do not correspond to the motor drive connector numbers.

If the MC433G is to be re-programmed CONTROL MODE must be disabled first using the “**cm**”

command – this forces the ATmega644 to release control of the Step/Dir parallel port controls lines so **MC433Prog.exe** can control them. MC433G motor controllers can only be programmed correctly if the “**cm**” command is first sent via the RS-232 port – CONTROL MODE is re-enabled by sending the “**cc**” command.

Remove power from the MC433G DC Logic and Motor Power DCIN. Remove the Flash Programming Enable jumper (J30). Attach a parallel port cable to the controller and apply power. Now install the Flash Programming Enable jumper.

Start **MC433Prog.exe** by double clicking or starting it a cmd window – the first command to use is the “**xa**” command – this interrogates each processor to verify it’s type and fuse settings. Type “**h**” to show all commands – note that it is possible to send commands to the MC433G with **MC433Prog.exe** that will lock the board from further programming so please use the commands carefully. Contact the company If in doubt about how to proceed.

```
>MC433prog <CR>
MC433GProgramming Utility V1.21
© Copyright 2008, SOC Robotics, Inc.
Type 'e' to exit or 'h' for help
-xa
Reset 1: Target not responding
Reset 2: Target ATmega168 responding - default fuses set
Reset 3: Target ATmega168 responding - default fuses set
Reset 4: Target ATmega168 responding - default fuses set
Reset 5: Target ATmega168 responding - default fuses set
-
```

Having confirmed all the processors are alive and talking new motor controller software can be downloaded to each motor controller by typing “**f**” followed by the name of the Program Configuration file.

```
-fMC433config_R95.txt<CR>
< contents of MC433config_R95.txt is executed >
```

The program will now program all four processors in turn loading the correct hex file from the MC433config\_R95.txt text file.

Below is an example of the contents of Program Configuration file **mc433config\_R95.txt**:

```
x2
dfmc433_motor_controller_R95z.hex
x3
dfmc433_motor_controller_R95.hex
x4
dfmc433_motor_controller_R95.hex
x5
dfmc433_motor_controller_R95.hex
xa
```

Exit **MC433Prog.exe** by entering the “**e**” command. Note processor X2 – Z Axis is loaded with a different program than the other three processors. X2 is unable to monitor ESTOP so must get this information from the other processors.

All Program Configuration files supplied by the company end with a Version Identification Number. In the example file above the VIN is R95 – Version 0.95.

After programming is complete **REMOVE** the programming enable jumper – the board does not need be powered down to remove the jumper. You are now ready to send new step/dir

commands to the controller.

### 5.3 Updating EEPROM Contents

Update Motor Controller EEPROM contents using **MC433Setup.exe**. In order to use **MC433Setup.exe** Motor Controller Version 0.97 or later must be installed in motor controller Flash. To check the current motor controller Flash version run **MC433Setup.exe** with the EEPROM update jumpers installed - **MC433Setup.exe** interrogates all four motor controllers retrieving all setup information. **MC433Setup.exe** has not been released.

**MC433Convert.exe** is a related application that converts a text file of setup configuration parameters into an Intel Hex file that is subsequently downloaded into each motor controllers' EEPROM using **MC433Prog.exe**. **MC433Convert.exe** will be released with Version 0.97 Motor Control software.

### 5.4 Post Programming Procedure

After completing the programming procedure it's good practice to interrogate the processors one more time with the "**xa**" command to ensure all Motor Controllers are responding. Motor controllers will only respond to **MC433Prog.exe** if the Flash Programming Enable jumper is installed.

One last thing:

**DO NOT FORGET TO REMOVE THE FLASH PROGRAMMING ENABLE JUMPER BEFORE SENDING G CODE COMMANDS TO THE MC433.**

## 6.0 G Code Command Description

### 6.1 Overview

The MC433G has a dedicated processor that interprets and executes G Code commands sent to it over the RS-232 or RS-485 Serial Interface. G Code is a language introduced over twenty years ago to provide a common protocol for the control of CNC equipment. The G Code standard is described by RS274NGC. This manual does not intend to teach G Code or describe in detail how G Code is used but rather to describe the specific implementation in the MC433G. The version of G Code implemented in the MC433G upgraded frequently. For the latest version release and change log go to the MC433G product page under the download section of the SOC Machines web site. The G Code program executing on the MC433G is called GStep. The current version of GStep is output in the sign on message when the MC433G powers up.

The MC433G supports several G, M, S, T and related NS274 commands along with proprietary commands unique to the MC433G. Commands are entered one after the other at the dash prompt. Commands can be separated by space or not and are case insensitive. A command is executed after a Carriage Return <CR> is entered. All commands are echo'd automatically. At the completion of a command the prompt character is output indicating the MC433G is ready for the next command. Host software can use reception of the prompt character to manage the flow of commands. Although the MC433G implements XON/XOFF flow control host software does not always react fast enough to prevent a buffer overflow in the MC433G. Buffer overflow is detected by the presence of the @ character in the returned data stream.

The MC433G supports a number of proprietary commands to allow interaction with attached peripherals such as a joystick or LED display. The host can request information from the MC433G using various status commands such as Get Current Position or Get Joystick Position. In lead screw CNC applications Backlash Compensation can be entered on a per axis basis in integer steps - backlash correction are applied symmetrically for all direction changes.

The MC433G maintains an internal absolute coordinate system using 32bit signed long integers. This allows axis position changes caused by joystick movements to be accurately tracked so that subsequent G Code position commands can move to the new correct position while maintaining correct absolute position. G0,1,2,3, I,J,K and F commands can be entered as integers or as single precision floating point numbers.

The X, Y and Z axis are assumed to be linear. The fourth axis is interpreted as either a linear or rotary axis using the A or B command respectively. Parameters after the B command are interpreted as angles in degrees. The relationship between an axis parameter and the number of steps executed by the stepper is set by a resolution parameter. The default resolution is 0.050" per 360 rotation of the stepper or 0.00025" per step- so a command such as N0001 G1 X0.250 F4.25 causes the X axis stepper to rotate 360 degrees five times or to execute 5\*200 = 1000 steps at a position change rate of 4.25 inch per minute. The resolution parameter is programmable. The current release assumes the resolution parameter once set is the same for all axis except for rotary motion - this parameter is set separately. In the examples <CR> indicates a carriage return and <LF> indicates a line feed.

A new release of GStep (Version 1.75) is planned for mid December 2008. The new version will have significant new features and will include a real time communication protocol with the Motor Controllers as well as support for several peripherals such as joystick and Displays.



The Table 6-1 below summaries the command codes supported by the MC433G.

**Table 6-1 G Code Commands Supported By MC433G**

<b>Command</b>	<b>Function</b>
G0	Rapid Move maximum rate on each axis
G1/G01	Linear Move – constant speed
G2/G02	Circular Interpolation - Clockwise
G3/G03	Circular Interpolation – Counter Clockwise
G4/G04 Psss	Dwell in seconds
G46Cxxx	Back lash Compensation – C is X,Y,Z or A
G90	Absolute Positioning
G91	Relative Positioning
G92	Set home position
G94	Output status
G96mmmmm	Dwell for mmmmm mseconds
G97	Output current axis position
I,J,K,R	Variables for Circular Interpolation
N	G Code number
F	Feed Rate – Fxx.xxxx ipm
X	X Axis
Y	Y Axis
Z	Z Axis
A	A Axis (fourth axis – linear)
B	B Axis (fourth axis – rotary)
M51 2	Aux Output 1 On/Off
M53 4	Aux Output 2 On/Off
M55 6	Aux Output 3 On/Off
M57 8	Aux Output 4 On/Off
H	
J	Enable/disable Joystick recognition
CM	Set Monitor Mode
CC	Set Control Mode
SP	Output current position
SB	Output Backlash Values
SJ	Output Joystick Values
SA	Output Aux Port Values
SX	Output Step Mode Values
SE	Enter preference setup menu
LE	Leave preference setup menu
SS	Output Status
P	Display preference settings
!	Enter Test Mode

## 6.2 G Code Command Description

Supported G Code commands are described in detail in this section.

### **G0/00 Rapid Move - Maximum rate on each axis**

Rapidly move to the new position moving each axis at the maximum specified rate.

Example: N00001 X1.0 Y1.0 Z0.25 A2.25 F8.0 <CR> - Move all four steppers at the maximum step rate until the destination is reached.

### **G1/01 Linear Interpolation – Specified feed rate**

Move the X, Y, Z and A axis at the specified feed rate so all axis arrive at the destination coordinates at the same time. The MC433G is capable of 1000 steps per second when executing a linear interpolation move.

Example: N00001 X1.0 Y1.0 Z0.25 A2.25 F8.0 <CR> - Move all four steppers at the different step rates so they all arrive at their respective destinations at the same time.

### **G2/02 Circular Interpolation - Clockwise**

Move the X and Y axis in an clockwise arc from the current position to the destination position using parameters defined by the I,J or R settings. If a new Z or A position is present then movement on these axis will also occur. The MC433G supports both center and radius circular interpolation.

Example 1: N00001 X1.0 Y1.0 I0.0 J0.0 <CR> - Assuming current position is (1.0, 0.0) move in an arc to (1.0, 1.0) using (0.0,0.0) as the origin.

Example 2: N00001 X1.0 Y1.0 R1.0 <CR> - Assuming current position is (1.0, 0.0) move in an arc to (1.0, 1.0) using a radius of 1.0.

### **G3/03 Circular Interpolation – Counter Clockwise**

Move the X and Y axis in an counter clockwise arc from the current position to the destination position using parameters defined by the I,J or R settings. If a new Z or A position is present then movement on these axis will also occur. The MC433G supports both center and radius circular interpolation.

Example 1: N00001 X1.0 Y1.0 I0.0 J0.0 <CR> - Assuming current position is (1.0, 0.0) move in an arc to (1.0, 1.0) using (0.0,0.0) as the origin.

Example 2: N00001 X1.0 Y1.0 R1.0 <CR> - Assuming current position is (1.0, 0.0) move in an arc to (1.0, 1.0) using a radius of 1.0.

### **G4/G04 Psss Dwell in seconds**

Set dwell time in seconds. The MC433G pauses for the specified delay period then return the dash prompt.

Example: N00001 G4 P10 <CR> - Delay for 10 seconds.

### **G90 Absolute Positioning**

Set absolute positioning mode.

**G91 Relative Positioning**

Set relative positioning mode. Note relative position mode is not implemented but is scheduled for implementation in new version.

**G92 Set home position**

Zero all four axis to 0.0000. Defines a new absolute start position.

**G94 Output status**

Output the system status such as current position, backlash setting, etc.

**G96mmm Dwell for mmm mseconds**

Delay for a specified period of time measured in mseconds. Alternative command to G4 for finer resolution delays.

Example: N00001 G9612 <CR> - Wait for 12mseconds before accepting the next command.

**G97 Output current axis position**

Output the current position of all four axis. The output is in floating point format using the current resolution in effect. The command converts the current step position of each axis in 32 bit Signed Long format to floating point format.

**6.3 M Command Description**

Supported M Code commands are described in detail in this section. The M code commands are used to control the four Auxiliary digital output signals AUX1 to AUX4.

**M51|2 Aux Output 1 On/Off**

M51 turns the Aux 1 output on - high level on Relay Expansion port pins and a logic low for the Aux output port. The Aux output port is an NPN open collector circuit. A high level pulls the line to ground while a low level. M52 turns the Aux 1 output Off.

**M53|4 Aux Output 2 On/Off**

M53 turns the Aux 2 output on - high level on Relay Expansion port pins and a logic low for the Aux output port. The Aux output port is an NPN open collector circuit. A high level pulls the line to ground while a low level. M54 turns the Aux 1 output Off.

**M55|6 Aux Output 3 On/Off**

M55 turns the Aux 3 output on - high level on Relay Expansion port pins and a logic low for the Aux output port. The Aux output port is an NPN open collector circuit. A high level pulls the line to ground while a low level. M56 turns the Aux 3 output Off.

**M57|8 Aux Output 4 On/Off**

M57 turns the Aux 4 output on - high level on Relay Expansion port pins and a logic low for the Aux output port. The Aux output port is an NPN open collector circuit. A high level pulls the line to ground while a low level. M58 turns the Aux 4 output Off.

## 6.4 Other RS274 Commands

Other supported RS274 commands are described in detail in this section.

### **N G Code Line Number**

The Line number is optional and is ignored but helpful to read G Code files.

Example: N0001 G0 X2.025 Y3.125 F4.000 <CR>

### **F Feed Rate – Fxx.xxxx ipm**

The Feed Rate is used by G1, G2 and G3 commands to set stepper step rate. Step rate is related to step resolution that is related to the specific mechanical configuration of an application. The feed rate remains in effect until changed.

Example: N00001 G1 X2.000 Y2.0000 F4.0000 <CR>

### **X X Axis**

Sets a new position for the X axis. All X Axis moves are linear. The number of steps and direction moved depends on the difference between the current X axis position, the end position, direction change (if any), backlash compensation and resolution. For example if the current X position is 1.00000 the target X position is 1.25000, step resolution is 0.00025 then the stepper steps 10000 times clockwise.

Examples: N0001 X0.2500 F4.0 <CR> - Move to X=0.2500 at 4.0ipm  
N0001 X2 F3 <CR> - Move to X=2.0000 at 3.0ipm

### **Y Y Axis**

Sets a new coordinate for the Y axis. All Y Axis moves are linear.

### **Z Z Axis**

Sets a new coordinate for the Z axis. All Z Axis moves are linear.

### **A A Axis (Fourth Axis - Linear)**

Sets a new coordinate for the A axis. All A Axis moves are linear.

### **B B Axis (Fourth Axis - Rotary)**

Sets a new coordinate for the B axis. All B Axis moves are rotary. The B Axis is actually the fourth axis re-interpreted as rotary motion. The B Axis provides a means to control rotary motion such as a turntable. Parameter is degrees.

Example: N00001 G1 X1.0 B25.0 F4.0 <CR> - Move to X Axis location 1.0 and rotate B axis 25.0 degrees.

## 6.5 Proprietary Commands

The MC433G implements proprietary commands not supported by the RS274 command set. Commands to set axis resolution, return current position, set backlash compensation, etc are members of this set of commands.

## H Output online Help

Output a summary of all commands by entering the H command.

If the “h” command is entered then the following help message is output. Note that the actual help may have new features added at each revision upgrade.

```
-h
MC433 Stepper Controller - Gstep V1.63
Copyright 2006, SOC Robotics, Inc.
Type H for help menu - Control mode active

-
Single Step commands:
> = step X CW
< = step X CCW
i = step Y CW
o = step Y CCW
u = step Z CW
d = step Z CCW
z = step A axis CW
w = step A axis CCW

G Code Commands ([]-optional - floats xx.xxxxx):
Fourth axis is linear A or rotary B (angle)
G0 Xx.xx Yx.xx Zz.zz Aa.aa Bb.bbb- Rapid move
G1 Xx.xx Yx.xx Zz.zz Aa.aa Bb.bbb Ff.fff - Linear Interpolation
G2 Xx.xx Yx.xx Zz.zz [Ii.ii Jj.jj]Rr.rrr|Daaa.a] - Circular Interpolation - CW
G3 Xx.xx Yx.xx Zz.zz [Ii.ii Jj.jj]Rr.rrr|Daaa.a] - Circular Interpolation - CCW

G46Cxxx - Set backlash for each axis where C is X,Y,Z or A
G4Ppppp - Dwell for pppp seconds
G92 - Reset origin x,y,z,a
G96mmmm - Dwell mmmm msec
G97 - Output Position

M Code Commands:
M5[1|2] - Aux 1 1-ON 2-OFF
M5[3|4] - Aux 2 1-ON 2-OFF
M5[5|6] - Aux 3 1-ON 2-OFF
M5[7|8] - Aux 4 1-ON 2-OFF

Other Commands:
sp - Get Current position
ss - Get All Status information
sb - Get Backlash values
sj - Get Joystick values
sl - Get Limit switch values
sa - Get Aux IO values
sd - Get Digital IO values
sx - Get Step mode values
se - Enter preference setting mode
le - Leave preference setting mode
cc - Set command mode active
cm - Set monitor mode active
p - Display preference settings
b - Change baud rate

-
```

## Single Step Commands

The MC433G is capable of single stepping the steppers using simple character sequences entered at the command prompt. Single step commands are executed immediately and do not require a <CR> or <LF>. The commands are summarized in the table below and are case insensitive:

Step Command	Action
<	Step X axis CW
>	Step X axis CCW
i	Step Y axis CW
o	Step Y axis CCW
d	Step Z axis CW
u	Step Z axis CCW
z	Step A axis CW
w	Step A axis CCW

### CM Set Monitor Mode

Set Control Monitor mode – the G Code Processor releases all step/direction and Aux IO lines and allows the parallel port to take control. The Control Monitor mode can be used if step/direction control signals are sent to the MC433G via the parallel port. The Control Monitor Mode must be set to allow Flash Programming of the Motor Controllers.

### CC Set Control Mode

Set Control Active Mode – the G Code processor takes control of the Step/Dir signals. The parallel port cable should not be connected in this state or signals sent to the Motor Controllers by the G Code Processor will conflict.

### SP Output Current Position

Output the current position of all four axis. The output is in floating point format converted using the current resolution in effect. The MC433G maintains current position using a 32bit Signed Long.

### SB Output Backlash Setting

Output the current backlash settings for all four axis - X, Y, Z and A. Backlash settings are assumed symmetric and are stated in steps.

### SJ Output Joystick Values

Output the current joystick values for all four axis - X, Y, Z and A and the JSW switch level. The joystick is an 8 bit number converted to plus or minus 128. The switch is either open or closed.

### SA Output Auxiliary Output Port Settings

The state of the four auxiliary output signals is returned.

### SS Output Status Summary

Output all the current status information of the MC433G.

### ! Enter Test Mode

Enter test mode by entering the '!' character. In test mode the MC433G drives one or all Motor Controllers at a user entered step rate. This helps setup the steppers during initial installation. The test mode command is as follows:

```
-!  
Test Mode (Enter 'e' to exit)
```

```

Commands:
  <axis>[c|w]<step rate/sec>
  Example: Xc200 Yw1500 Zc700 <CR>
>

```

The following example starts the x and y axis stepping at 200 pps (pulse per second – steps).

```
>xc200 yw200<CR>
```

## SE Enter Preference Configuration Menu

This command is used to enter the Preference Configuration Menu. The G Code processor assumes that the resolution of each axis is 0.00025in per step, there is no backlash, maximum rapid move is 20ipm and step direction polarity is P N N N. The Preference setting menu is entered by entering the “se” command:

```

-se
Entering Preference Mode
Type 'le' to leave - dp to display preferences - ss save settings
=

```

For see the current preference settings enter “dp”.

```

=dp
Preference Settings:
Baud rate = 115200
Max Feed rate (mf)= 20
Resolution [X,Y,Z,A] (rr)= 0.000125 0.000125 0.000125 0.000125
Backlash [X,Y,Z,A] (bl)= 0 0 0 0
Direction polarity [X,Y,Z,A] (pl)= P N N N
=

```

Now change any preference setting by entering a two character preference selection command followed by the new values. For example to change the resolution from 0.00025 to 0.000125 enter the following command:

```

=rr0.000125 0.000125 0.000125 0.000125
=

```

Changed preferences are not saved in EEPROM until the “ss” command is entered.

Preference setting changes take effect immediately for the time the MC433G is powered but revert to stored settings unless saved by the “ss” command. The “le” command is used to leave Preference Configuration Mode.

## 6.6 Host Support Utilities

To regulate the flow of commands to the MC433G host command line application **ngload.exe** can be used to feed one line at a time. **ngload.exe** accepts as input a file of G Code commands and sends them one line at a time to the MC433G. Unrecognized commands are ignored by the MC433G. **ngload.exe** expects to receive a dash prompt after each command.

## 6.7 Updating EEPROM Contents

The MC433G settings are stored in the G Code processors on chip EEPROM. EEPROM contents can be changed using the Change Preference command.

## 6.8 I2C Pass Through Protocol

The MC433G has an I2C communications port that allows GStep to communicate with and control a number of smart peripherals such as LED displays, Joysticks and smart relay boards. SOC Machines has a rich set of smart I2C peripherals all of which can be controlled by the MC433G using a unique pass through protocol that allows the Host PC to send I2C commands to the MC433G that are “passed through” to the I2C interface - this ensures the MC433G is able to accommodate new I2C peripherals as they become available without waiting for new version of GStep.

The Pass Through protocol will be implemented in the next release of GStep.



## 7.0 Electrical and Mechanical Description

### 7.1 Component Layout

All components are mounted on the top of the board. The Layout shows Rev 1.1b MC433G.

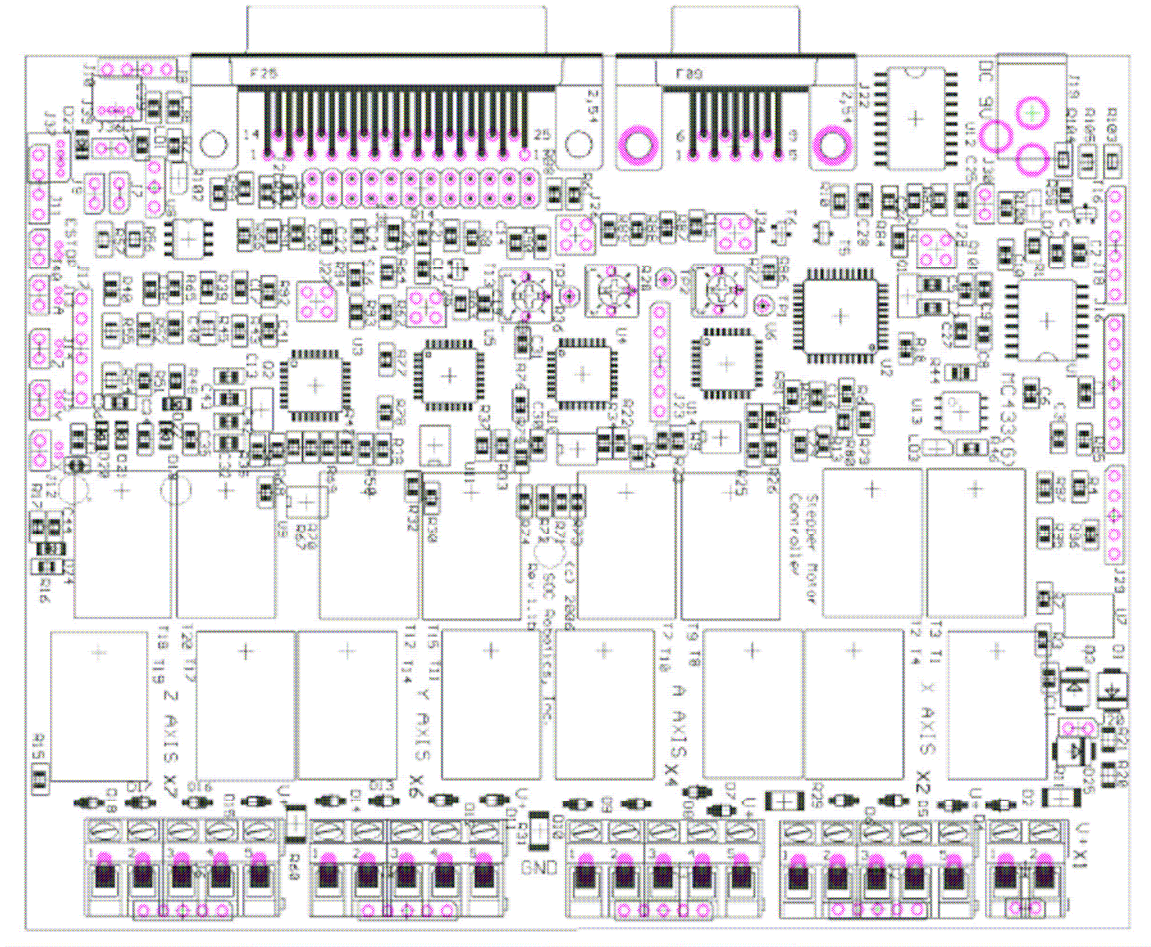


Figure 5-1. MC433G Rev 1.1bComponent Layout.

## 7.2 Electrical Specifications

### Electrical

Input power: 7-12VDC @ 120ma

Board power: 5V DC @ 90ma

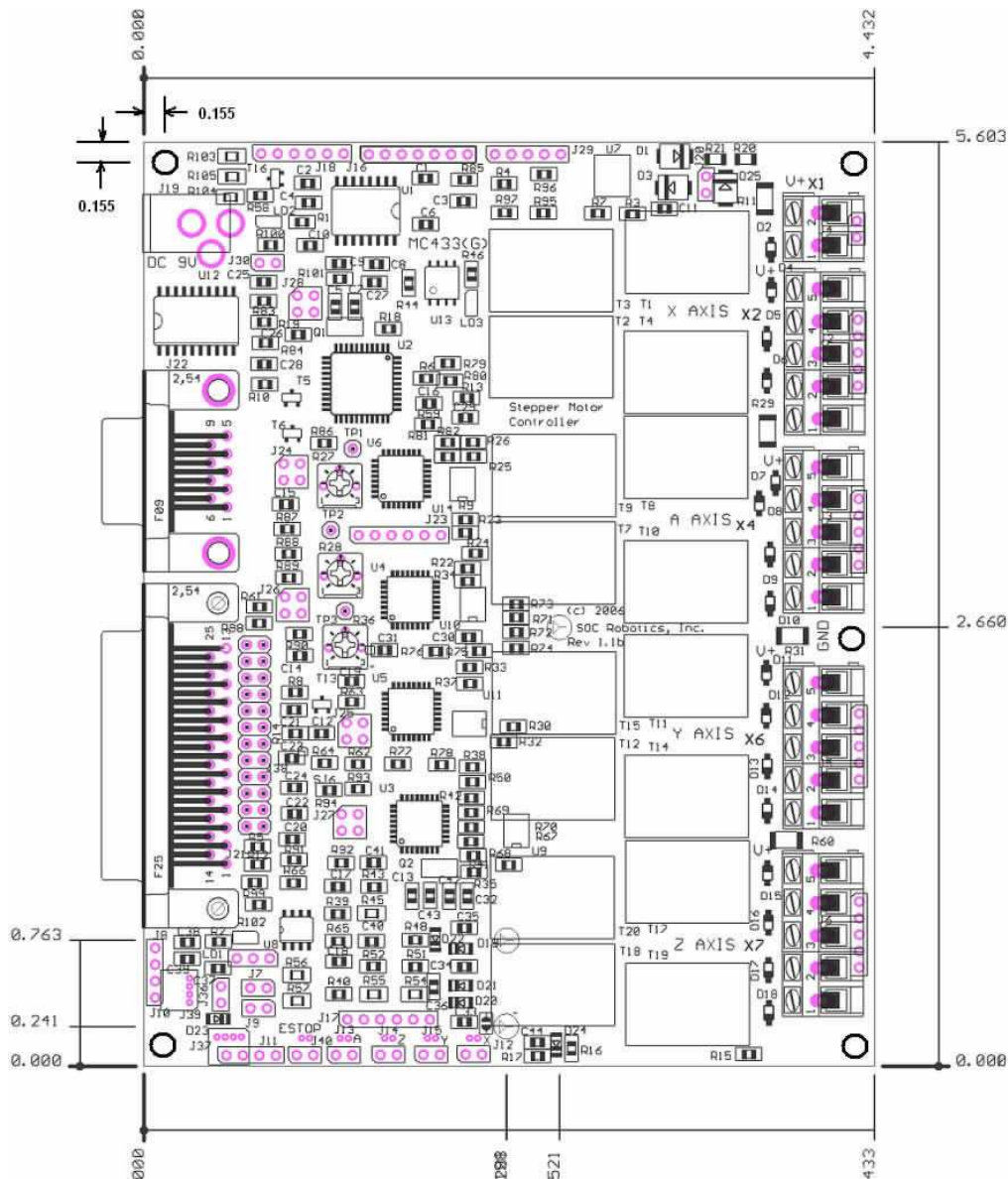
### Mechanical

Dimensions: 4.42x5.61 in (four mounting holes)

Weight: 290grams

## 7.3 Mechanical Dimensions

Board dimensions stated in inches. Connector locations with respect to the lower left corner are annotated in the drawing. A sample schematic with connector library and board layout in Eagle CAD format is available at [www.soc-robotics.com/download/mc433qlayout.htm](http://www.soc-robotics.com/download/mc433qlayout.htm).

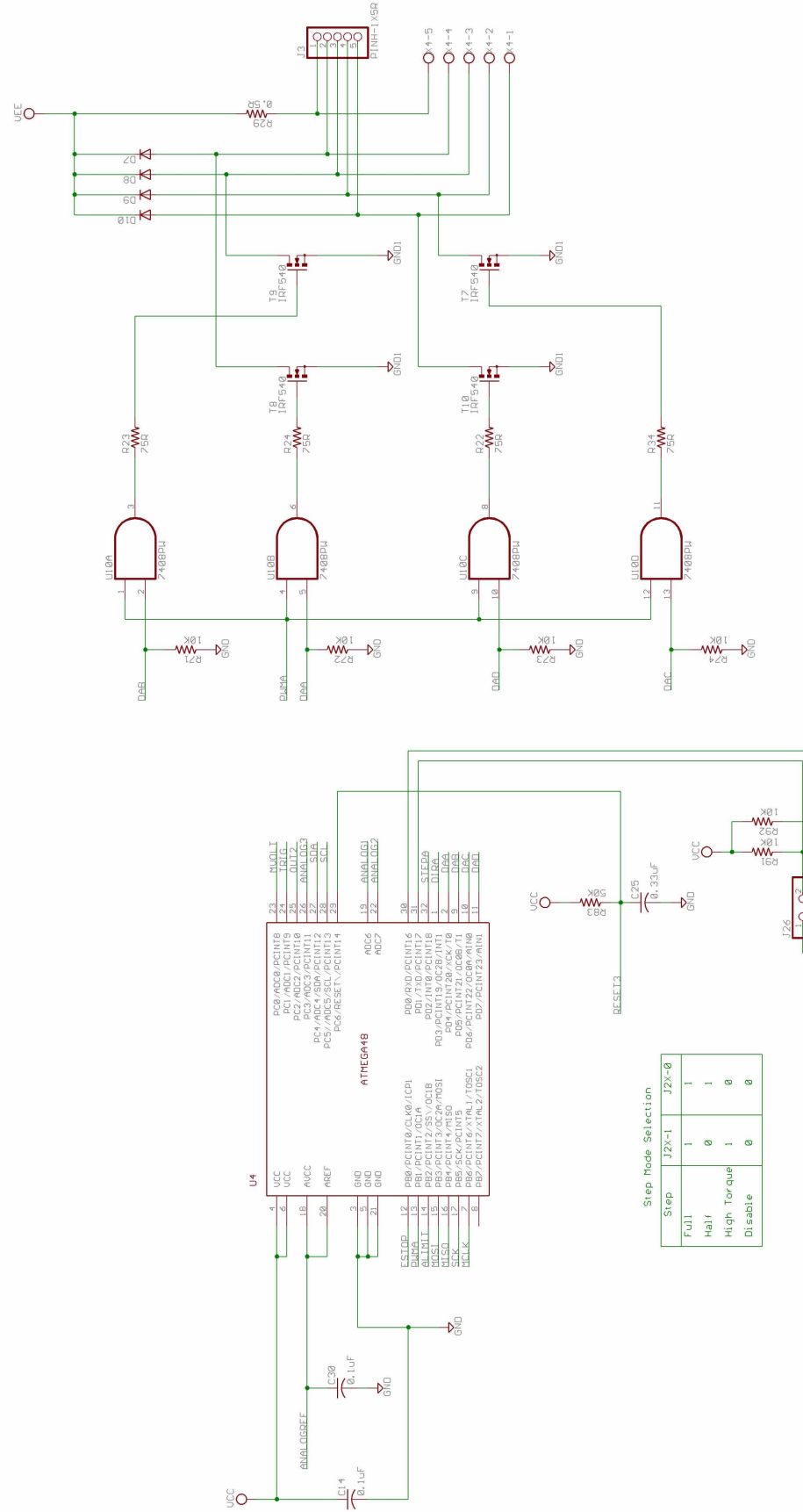


## 7.0 MC433G Rev 1.1b Schematics

<p>(c) Copyright 2006, SOC Robotics, Inc. All rights reserved.</p> <p>Uni-Polar Stepper Motor Controller</p> <p>4-Axis</p> <p>10 Amp/Motor, Full, Half, High Torque</p> <p>PCB Rev 1.1b</p>		<b>SOC Robotics, Inc.</b> Vancouver, BC			
		TITLE: MC433R21			
		Document Number:	20050410	REV:	A
		Date:	9/04/2006 08:57:29a	Sheet:	1/8



(c) Copyright 2006, SOC Robotics, Inc. All rights reserved.



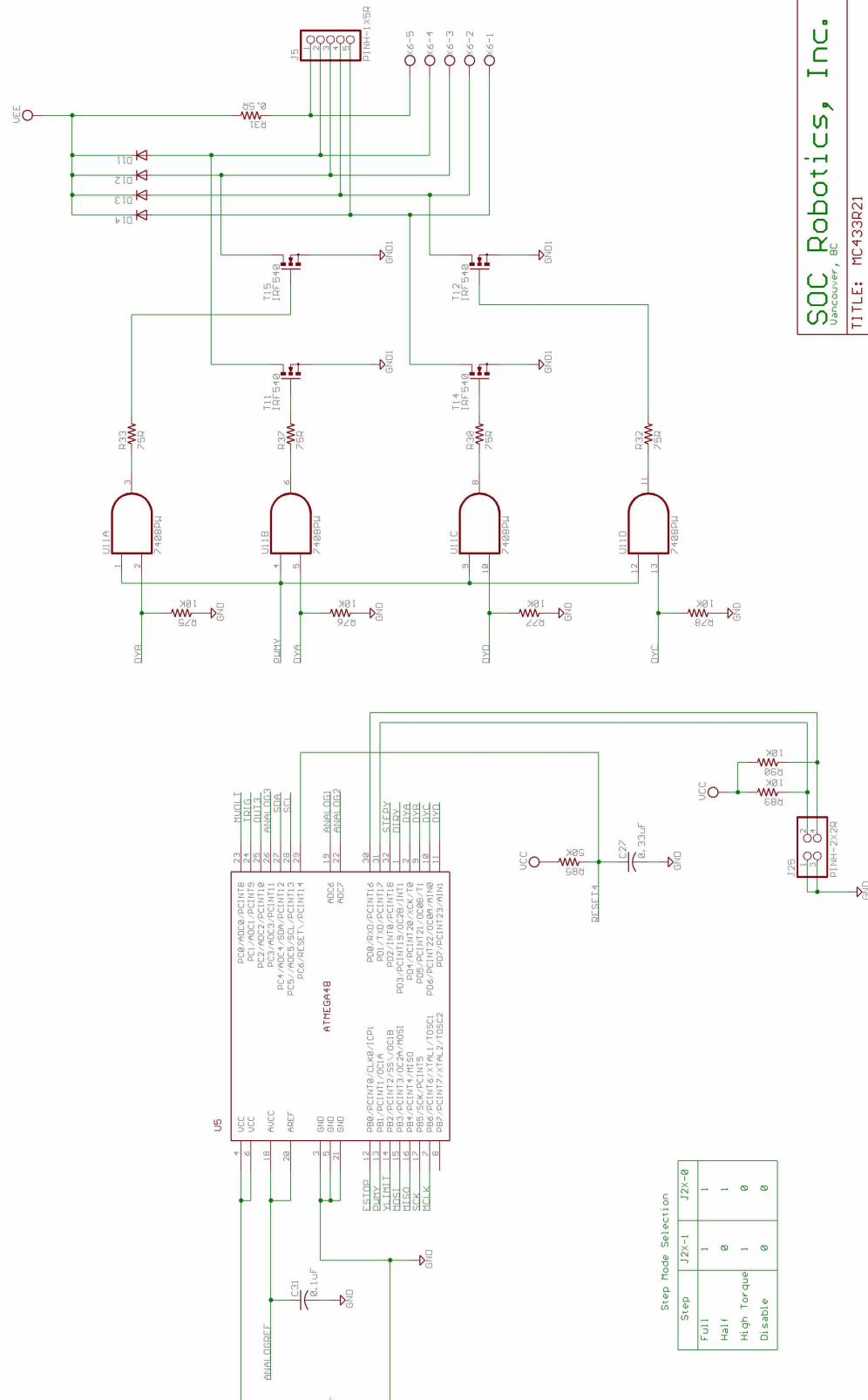
**SOC Robotics, Inc.**  
Warehouser, IL

TITLE: MC433R21	REV: A
Document Number: 20050410	
Date: 9/04/2006 08:57:29a	Sheet: 3/8

Step Mode Selection

Step	J2X-1	J2X-0
Full	1	1
Half	0	1
High Torque	1	0
Disable	0	0

(c) Copyright 2006, SOC Robotics, Inc. All rights reserved.



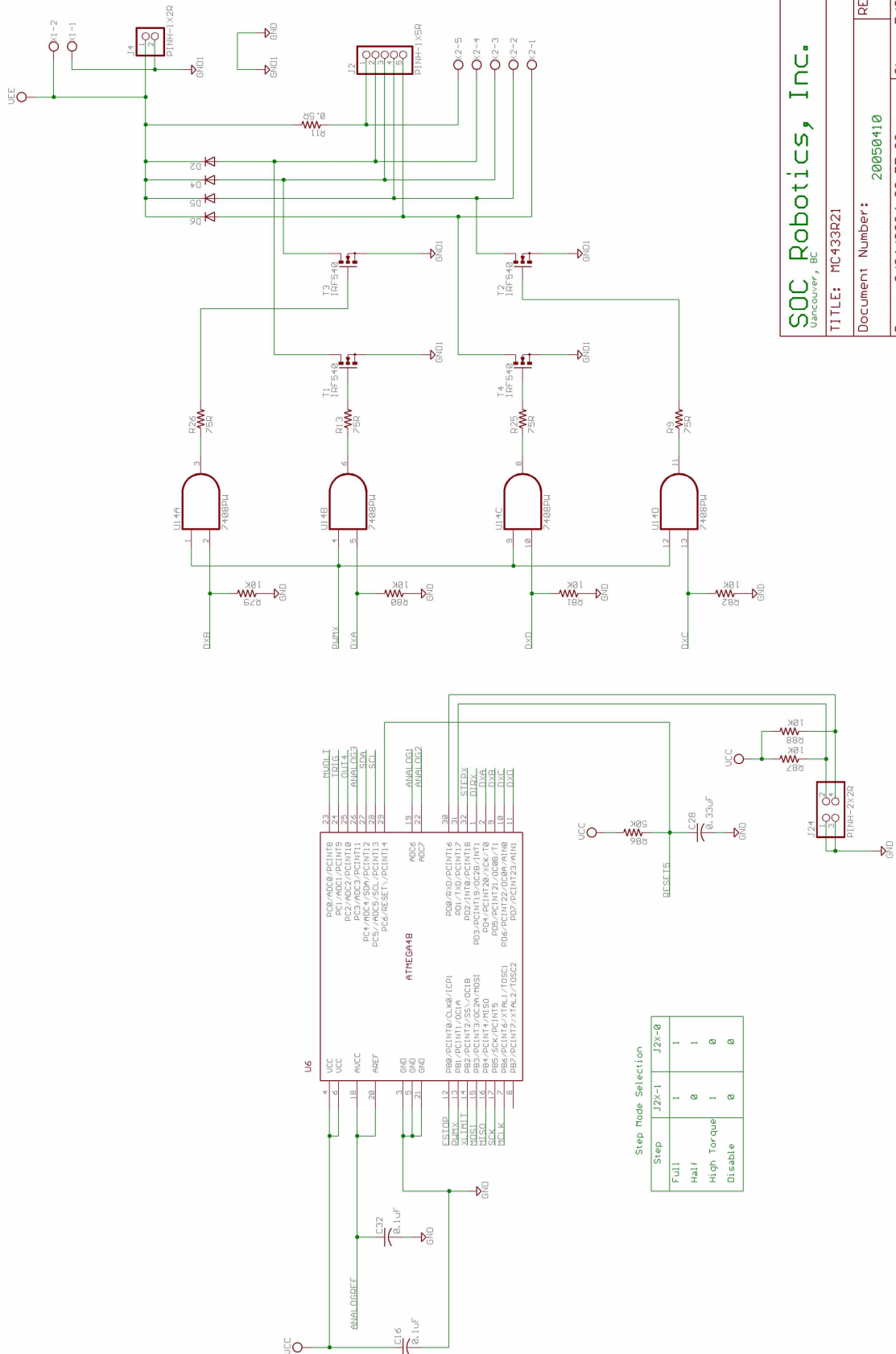
Step Mode Selection

Step	J2X-1	J2X-8
Full	1	1
Half	0	1
High Torque	1	0
Disable	0	0

**SOC Robotics, Inc.**  
 Vancouver, BC

TITLE: MC433R21  
 Document Number: 20050410  
 Date: 9/04/2006 08:57:29a  
 Sheet: 4/8  
 REV: A

(c) Copyright 2006, SOC Robotics, Inc. All rights reserved.

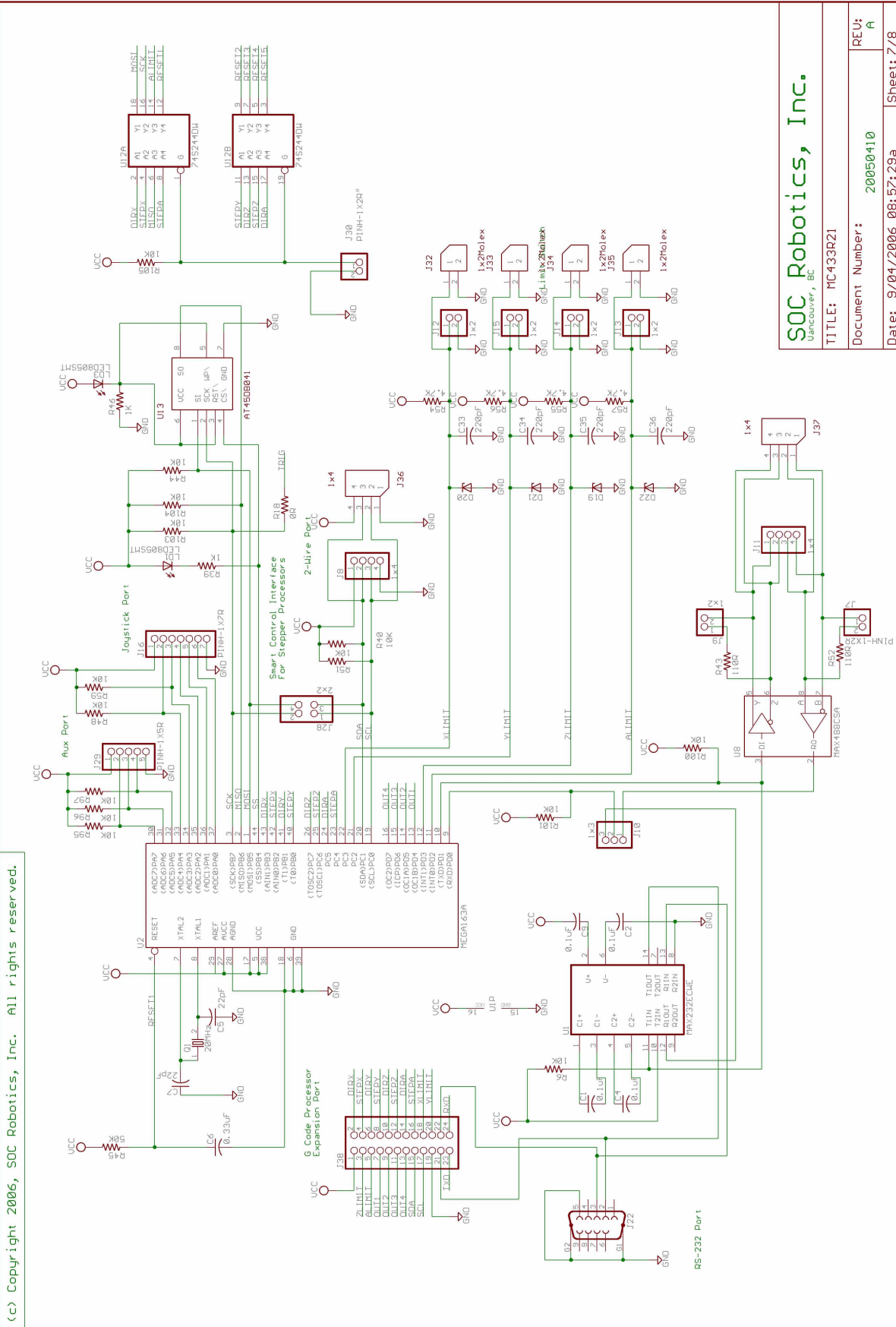


**SOC Robotics, Inc.**  
 Uncover, BC

TITLE: MC433R21  
 Document Number: 20050410  
 Date: 9/04/2006 08:57:29a  
 REV: A  
 Sheet: 5/8







SOC Robotics, Inc.  
Unconover, BC

TITLE: MC433R21

Document Number: 20050410

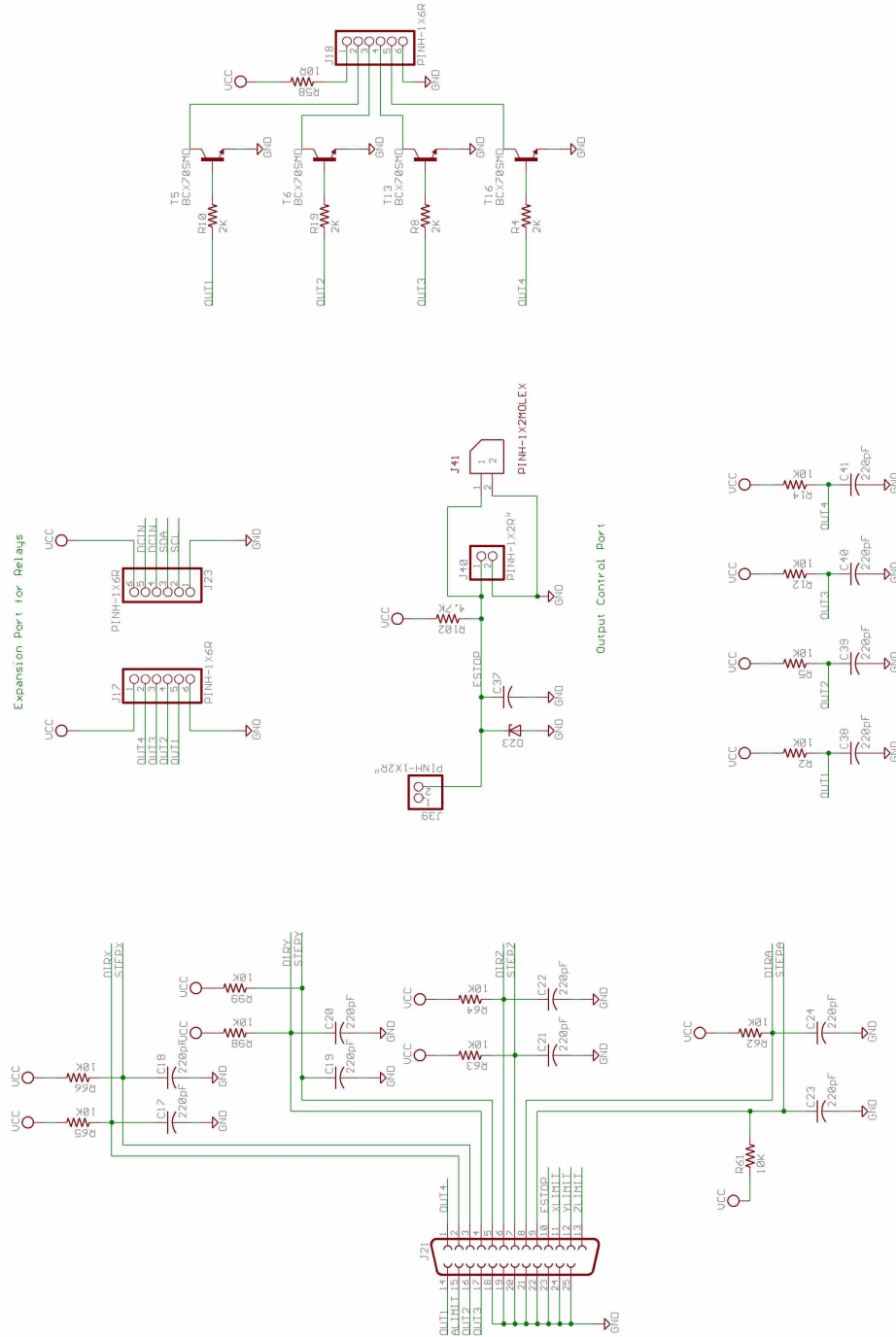
Date: 9/04/2006 08:57:29a

REV: A

Sheet: 7/8

(c) Copyright 2006, SOC Robotics, Inc. All rights reserved.

(c) Copyright 2006, SOC Robotics, Inc. All rights reserved.



<b>SOC Robotics, Inc.</b> <small>Manchester, BC</small>	
TITLE: MC433R21	REV: A
Document Number: 20050410	
Date: 9/04/2006 08:57:29a	Sheet: 8/8

**NOTES:**