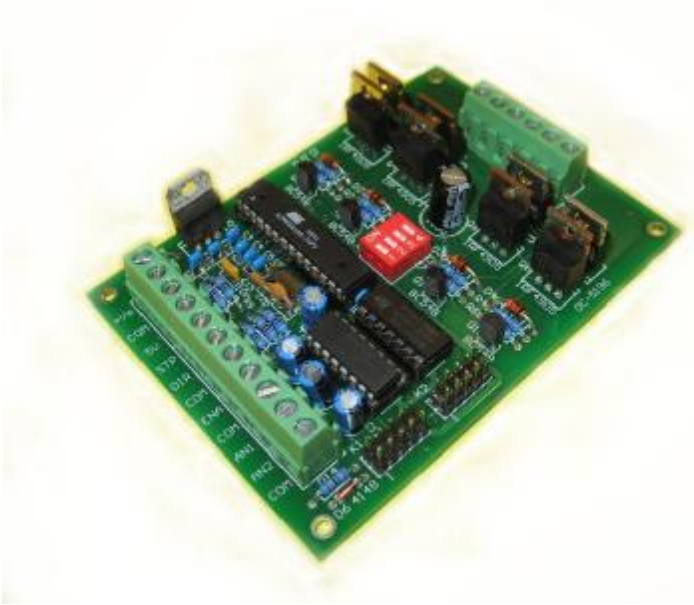


Ocean Controls KT-5197 Serial Bipolar Stepper Motor Driver



- Microcontroller Based
- Controls 4, 6 and 8 wire Stepper Motors
- Full and Half Stepping Modes
- Step and Direction Input Mode
- Serial Command Input Mode
- Analog (0-5V) Speed Control Input Mode
- High Quality Through Plated Dual Layer PCB
- Screw Terminals for all Connections
- Enable Input Removes Current From Coils when Pulled Low
- Controls Motors up to 6A

The KT-5197 is a Serial Stepper Motor Controller and Driver Combination. Combining the easy to use KT-5190 Serial Stepper Motor Controller with the KT-5196 Bipolar Stepper Motor Driver gives a single board solution for easy control of Bipolar Stepper Motors from a PC.

Connections:

Table 1 - Connections

Label	Description
+Vs	Logic Power Supply Positive Input (8-26)VDC
COM	Common (Ground) Connection and Power Supply Negative Input
5V	5VDC Output
STP	Step Input 0-5V
DIR	Direction Input 0-5V Internally Pulled High
COM	Common (Ground) Connection
ENA	Enable/Disable Input Low Level removes current from Motor Coils
COM	Common (Ground) Connection
AN1	Unidirectional Analog Input
AN2	Bidirectional Analog Input
COM	Common (Ground) Connection
M1	Motor Coil 1 Connection
M1	Motor Coil 1 Connection
V+	Motor Power Supply Input 5-50VDC
COM	Common (Ground) Connection
M2	Motor Coil 2 Connection
M2	Motor Coil 2 Connection
K1	Serial Port Connection 1
K2	Serial Port Connection 2

DIP Switches:

The four DIP switches located in the centre of the board control the serial addressing, half and full stepping modes and serial/pulse input modes. The analog input modes are selected by putting the controller in serial mode and sending a command to set the controller to one of the analog modes.

Table 2 – DIP Switches

	1	2	3	4
ON	Address	Address	Full Step	Serial Input
OFF	Address	Address	Half Step	Pulse Input

Table 3 – Address Selection

SW \ ADDR	00	01	02	03
1	OFF	ON	OFF	ON
2	OFF	OFF	ON	ON

Stepper Motor Driver Mode:

With DIP Switch 4 in the OFF position, the KT-5197 behaves in the same way as the KT-5196. In this mode the falling edges of pulses coming in on the STP input are converted to the coil activation sequence required to turn the stepper motor and the DIR input defines which way the motor turns. Connecting the ENA input to COM will remove the current from the motors coils and disable it.

Serial Stepper Motor Controller/Driver Mode:

If DIP Switch 4 is in the ON position the KT-5197 will ignore pulses on the STP input and will now listen for commands on the serial port. Commands are sent over the serial port using a baud rate of 9600, 1 start bit, 1 stop bit and No Parity (9600, 8, N, 1). The command structure is based on that of the KT-5190, in the form:

@AA CMND XXXXCR

Where AA is the 2 digit number of the motor being addressed, between 00 and 03 (Refer to Table 3 for addressing details), CMND is the 4 letter command, refer to Table 4 for available commands, XXXX is a numeric value associated with the command, and CR is the Carriage Return byte (0x0D) .

Table 4 - Commands

Command	Description
POSN	Set the position that the motor is currently at to be XXXX where XXXX is between -99,999,999 and 99,999,999
PSTT	Returns the current position of the motor
AMOV	Move the motor to the absolute position XXXX where XXXX is between -99,999,999 and 99,999,999
RMOV	Move the motor relatively from the current position by XXXX where XXXX is between -99,999,999 and 99,999,999
STOP	Stop the motor immediately
STAT	Get the status of the motor, see "Status Command Detail"
ACCN	Set the maximum stepping rate of the motor to XXXX where XXXX is between 0 and 9999, see "Acceleration". If the value for ACCN is 0 or less than RATE then no acceleration or deceleration occurs
ACCI	Set the Acceleration interval of the motor to XXXX where XXXX is between 1 and 9999, see "Acceleration"
RATE	Set the minimum stepping rate of the motor to XXXX where XXXX is between 1 and 9999, see "Acceleration"
OPTN	Bit 1 of the parameter turns Verbose mode on and off, bit 2 turns Checksum mode on and off, bit 3 turns Analog mode on and off, bit 4 selects Analog unidirectional or bidirectional modes, bit 5 sets the ENA input mode to hold or free. See "Option Command"
ACCF	When set to 1 the alternate Acceleration Curve is used, when set to 0 the standard Acceleration Curve is used. See "Acceleration"
SAVE	Saves RATE, ACCI, ACCN, ACCF and OPTN parameters to EEPROM, which are then automatically loaded on the next power up.

Status Command Detail:

The status command returns the state of the motors movement, direction and whether the ENA input is activated. Bit 1 is 1 when the motor is moving and 0 when it is not, Bit 2 is 1 when the motor is moving in reverse and 0 when it is moving forwards, Bit 3 is 1 when the ENA input is activated (Pulled low).

Acceleration:

Each time the command to move a stepper motor is issued the microcontroller calculates the stepping times to give a gradual acceleration and deceleration. The acceleration and final speed is determined by the ACCN, ACCI and RATE parameters. These default to 50, 2 and 10 respectively and are changeable by issuing the appropriate command.

The motor will start stepping at one step every ACCN interval, decrease this by ACCI intervals every step until the interval is RATE. When the motor is approaching the final position the step interval will increase by ACCI intervals from RATE until the final position is reached, at which point the interval will be back at ACCN.

An alternate acceleration curve can be activated by using the ACCF command with parameter 1.

In this mode the motor starts one step at ACCN, then two steps at ACCN+ACCI, four at ACCN+2xACCI, eight at ACCN+3xACCI etc until RATE is reached. This can be used to reduce the effects of the non-linearity of the normal acceleration. This mode of acceleration will take $2^{(ACCN-RATE)/ACCI}$ steps to reach top speed. Deceleration is done in the same manner.

Option Command:

Table 4 – Option Command

8	7	6	5	4	3	2	lsb
			1=Hold when ENA active 0=Free when ENA active	1=Bidirectional 0=Unidirectional	Analog Mode On	Checksum Mode On	Verbose Mode On

The option command is used to turn the Verbose, Checksum and Analog modes on, as well as being used to select whether the ENA input removes current from the motor coils when activated when in serial mode or whether the motor will stop turning but hold its position. When the option is set to hold position, any movement command sent to the controller will only move the motor 1 step. This can be used to move the motor off a limit switch if the limit switch has been tripped.

Turning the Verbose mode on will cause the SSMC to transmit !AA (where AA is the address) when the motor has reached its target position through an RMOV or AMOV command, or hit the limit switch during a move.

With Checksum mode on the controller will ignore commands unless a XOR checksum byte follows the command immediately after the Carriage Return. The XOR is performed on all the bytes of the command, including the @ and CR.

Eg. The checksum for the command "@01 RMOV 100" will be (64 xor 48 xor 49 xor 32 xor 82 xor 77 xor 79 xor 86 xor 32 xor 49 xor 48 xor 48 xor 13) = 123 which is the "{" character.

This is generally only useful if you have high level language controlling the SSMC, but not if you are typing the commands as you go. Note also, there must not be a Line Feed character between the Carriage Return character and Checksum character.

It may be handy to know that the checksum for the "@01 OPTN 0" command is the "O" character. In case you accidentally turn it on.

Analog Modes:

The Analog Speed control modes are activated using the option command. In unidirectional analog mode 0V is stopped, ACCN sets the slowest speed of the motor (just above 0V), and RATE sets the highest speed of the motor (5V). Unidirectional mode uses the AN1 terminal. The direction of the motor rotation can be changed by pulling the DIR terminal low.

In bidirectional analog mode 2.5V is stopped, ACCN sets the slowest speed of the motor and RATE sets the highest speed of the motor. The motor will increase in speed in one direction for voltages increasing from 2.5V to 5V and increase in speed in the opposite direction for voltages decreasing from 2.5V to 0V.

Note, although the ACCN and RATE parameters are set in exactly the same manner as with serial control, they do not equate to the same speeds, and some experimentation may be required.

How to use the Driver:

Using the Wiring Diagram as a guide, connect 12VDC (or anything between 8VDC and 26VDC) positive to the +Vs (Not the V+) terminal connect the 12VDC negative to the COM terminal closest to the +Vs terminal.

Connect the KT-5197 to a PC using a 9-pin straight through serial cable and the IDC connector cable.

A safety switch can be connected between ENA and COM, when it closes the motor will be disabled.

Connect the first motor coil between the M1 connections.

Connect the second motor coil between the M2 connections.

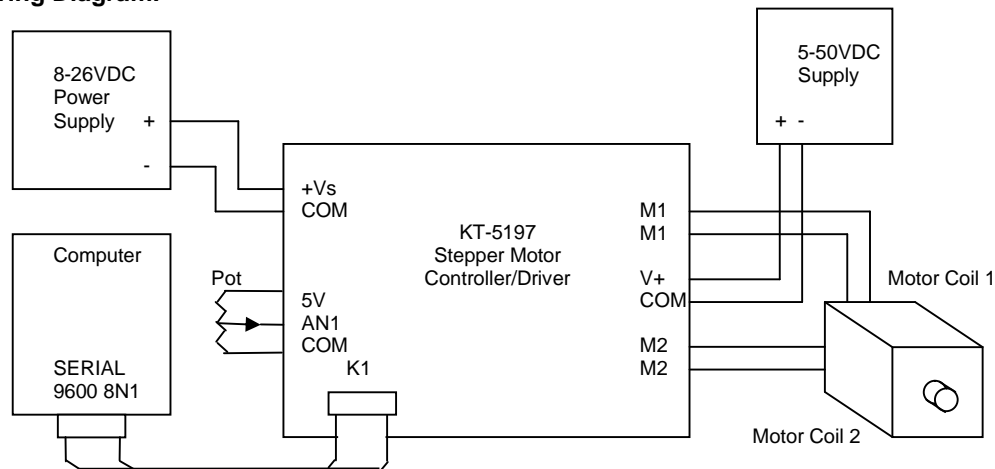
If you are using an 8-wire motor you can wire pairs of the coils in series or parallel.

Connect the motor power supply positive to the V+ terminal and the Negative to the COM terminal closest to the V+ terminal.

By default the driver uses half stepping mode, if you wish to use full stepping turn dip switch number 3 on. Dip switches 1 and 2 are used for addressing.

Put the controller in serial mode by setting dip switch 4 to the on position.

If you wish to use Analog inputs you can input 0-5V in AN1 or AN2, check the Analog Input section for more detail.

Wiring Diagram:**Half and Full Stepping:**

Because the KT-5197 is controlled by a Microcontroller it is possible to use half stepping. In half stepping mode input pulses and commands will advance the motor half of one full step. This is done by removing current from one coil before reversing the current. This allows for smoother operation and in many cases a higher top speed for the motor.

By default the driver will be in half stepping mode. To use full stepping mode turn dip switch 3 on.

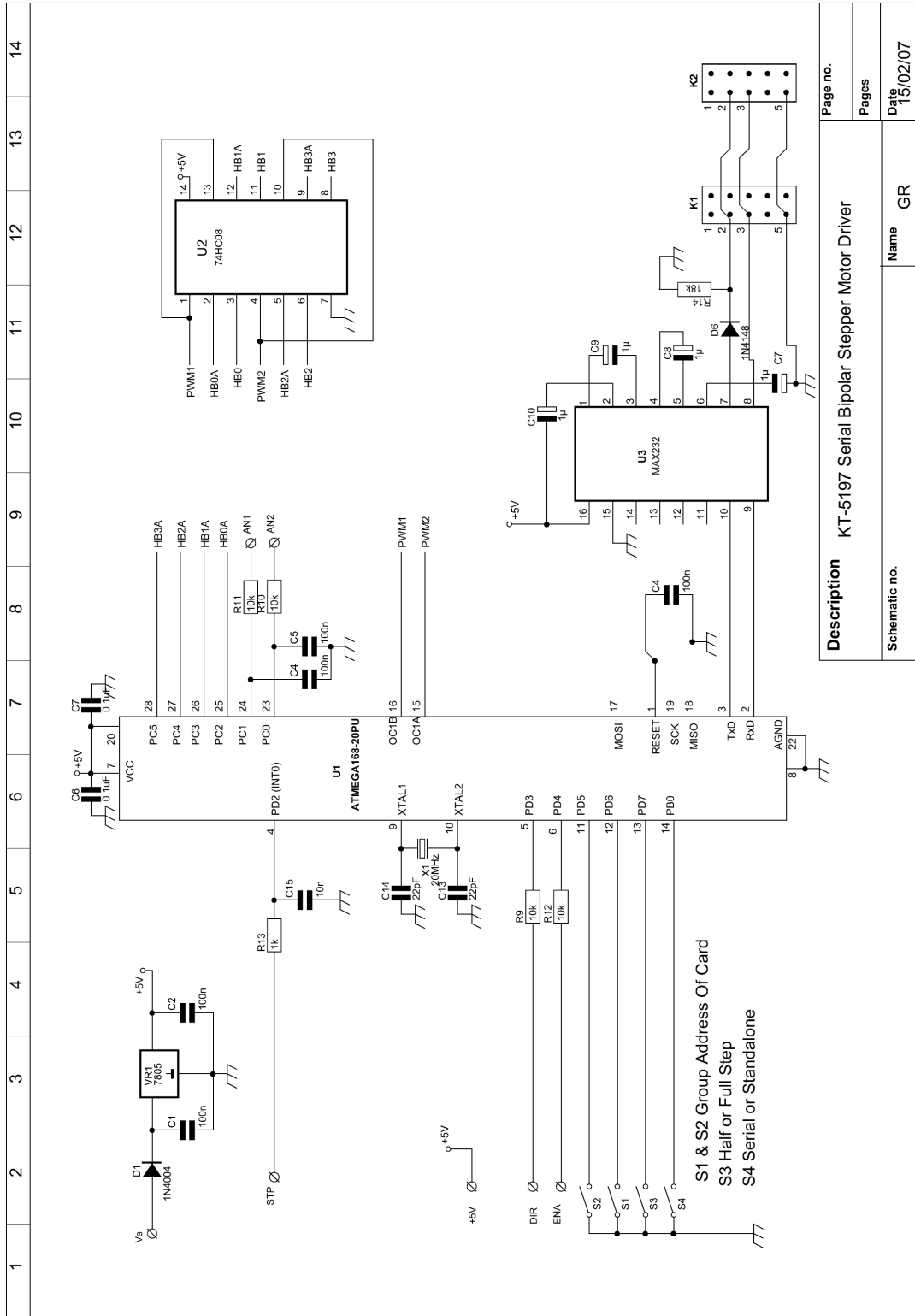
Heat:

It is recommended that you monitor the heat output of the FET's on the KT-5197 for the first few minutes, if you are using a motor which draws more than 2A of current. If you notice the FET's getting too hot you may need to use a heatsink (such as a drilled piece of aluminium) and fan to dissipate the heat. You will need to use insulating bushes and washers to ensure that there is no electrical connection between any of the FET's as they do not sit at the same potential.

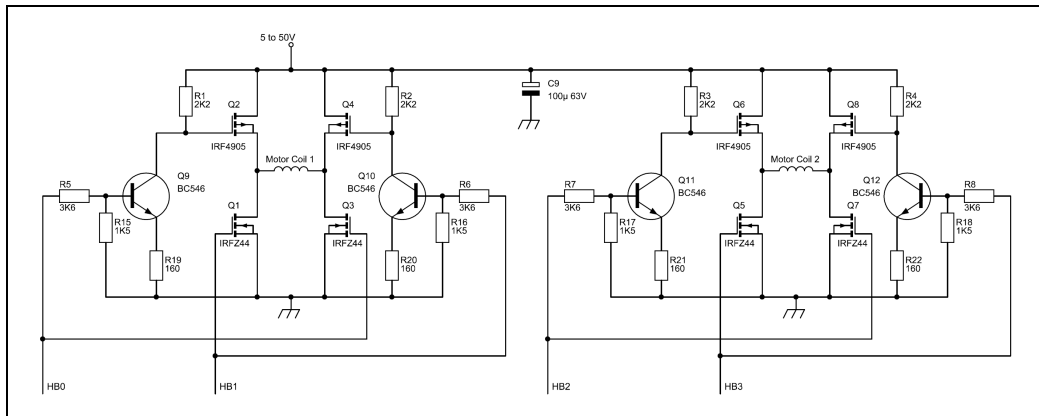
Performance:

It is possible to get better performance (higher speed and better torque response) from your stepper motor if you use current limiting resistors and a higher motor supply voltage. This will help overcome the coil inductance. You can find plenty of information on the internet on how to do this with stepper motor drivers. A good resource to look for is "Jones on Stepper Motors".

Circuit Diagram:



Circuit Diagram Page 2 (H-Bridge):



Parts List:

Part	Designator	Quantity
160 Ohm Resistor	R19-R22	4
1K Resistor	R13	1
1K5 Resistor	R15-R18	4
2K2 Resistor	R1-R4	4
3K6 Resistor	R5-R8	4
10K Resistor	R9-R12	4
18K Resistor	R14	1
1N4004 Diode	D5	1
1N4148 Diode	D6	1
22pF Capacitor Ceramic	C13, C14	2
10nF Capacitor Monolithic	C15	1
0.1uF Capacitor Monolithic	C1-C8	8
1uF Capacitor Electrolytic	C9-C12	4
100uF 63V Capacitor Electrolytic	C16	1
BC546 NPN Transistor	Q9-Q12	4
IRFZ44N N-Channel MOSFET	Q1, Q3, Q5, Q7	4
IRF4905 P-Channel MOSFET	Q2, Q4, Q6, Q8	4
7805 5V Regulator	VR1	1
4-way DIP Switch		1
3-way Terminal Block	T4-T6	3
2-way Terminal Block	T1-T3, T7	4
28-pin Narrow IC Socket	U1	1
14-pin IC Socket	U2	1
16-pin IC Socket	U3	1
20MHz Low Profile Crystal	X1	1
ATMega168 IC Programmed	U1	1
74HC08	U2	1
MAX232	U3	1
10-pin IDC Header (2x5pins)	K1, K2	2
D9 Female IDC socket		1
10-pin IDC Header Connector		1
9-wire IDC Ribbon		120mm