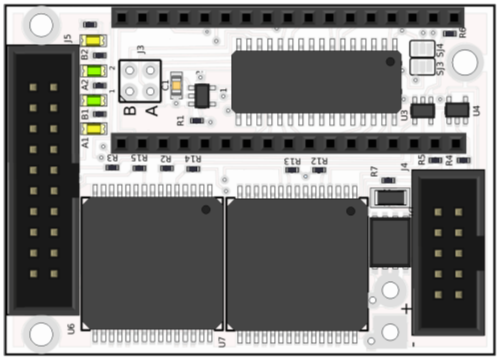
freaKontrol

NEO 1.0

Arduino Micro compatible

Data Sheet



DESCRIPTION

The NEO 1.0 module is an efficient digital closed loop motion control system, designed to control two bipolar inductive loads and perform two digital quadrature decoder function.

NEO 1.0 is Arduino Micro compatible, working with 32u8 Atmel or another generic micro controller compatible board.

The opensource firmware (uploaded on Arduino micro board) for general purpose applications is available on the **download section** of freakontrol web site.

FEATURES

- up to 20kHz PWM operation
- 30A output current per channel
- 33MHz max. quadrature decoder freq.
- TTL/CMOS encoder compatible
- High Noise Immunity
- two 32 bit decoder
- 24Vdc load power supply
- 5Vdc logic operation
- Protection:
  - Undervoltage and Overvoltage shutdown
  - Current limitation
  - Thermal shutdown
  - Reverse polarity
  - against short to GND and Vcc

APPLICATIONS

- CNC and robotic precision motor driving
- *proportional fluid valves controlling
- *automation voice coil
- *magnetic bearing
- *other inductive load

(*waiting for applicative software)

HOW NEO 1.0 WORKS

Neo 1.0 is suitable for the feedback control of two DC brushed motors with incremental optical encoder which provides two signals in quadrature (and possibly an “index” pulse at each round). The routine control is summarized below:

- during the rotation, the encoder generates two digital rectangular waves frequency proportional to the number of revolution of the motor. The quadrature decoder on the Board stores the number of periods (1x) or the number of rising edges (2x) or the rising and falling edge (4x).
- the micro controller interfaced with the board, requires to the decoder, the value that identifies the position of the motor.
- the micro controller requires the target position (SetPoint) supplied from the computer or from an external device.
- the micro controller executes a PID algorithm generating a PWM signal proportional to the motor supply voltage.
- the H-bridge receives and amplifies the PWM signal to drive the motor.

All take place at constant frequency (at the moment up to 4kHz).

OPERATING CHARACTERISTICS

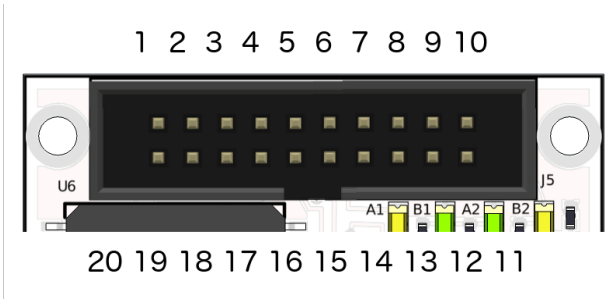
Table 1. Decoding Part Absolute Maximum Ratings

Parameter	Symbol	Limits	Units
Logic Power Supply	V_{CC}	-0.3 to +6.0	V
H-bridge Power Supply	V_{BAT}	-16 to +41	V
H-bridge Current	Imax (continuos)	30	A

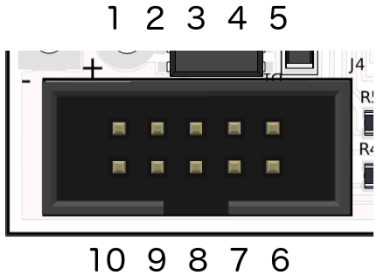
Table 2. Decoding Part Recommended Operating Conditions

Parameter	Symbol	Limits	Units
Logic Power Supply	V_{CC}	4.5 to 5.5	V
Intput voltage	(all data input)	4.5 to 5.5	V
Quad. Decoding Freq.	F_{DEC}	33	MHz
H-bridge Power Supply	V_{BAT}	24	V
External power source	V_{IN}	7 to 12	V

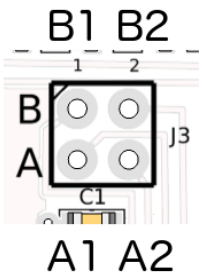
PINOUT



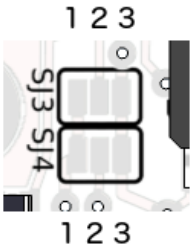
Pins	Pinout j5		Pins	Pinout j5	
1	Mx+	X Axis output + (from H-bridge)	11	By/	Inverting Encoder Y input from chB
2	Mx-	X Axis output - (from H-bridge)	12	Ay/	Inverting Encoder Y input from chA
3	My+	Y Axis output + (from H-bridge)	13	Bx/	Inverting Encoder X input from chB
4	My-	Y Axis output - (from H-bridge)	14	Ax/	Inverting Encoder X input from chA
5	Ix	input from Encoder X Index	15	Vcc	Encoder Power Supply +5V
6	Iy	Index Rotation Encoder Y input	16	GND	Encoder Ground
7	Ax	input from Encoder X chA	17	My-	connect to j5 Pin 4
8	Bx	input from Encoder X chB	18	My+	connect to j5 Pin 3
9	Ay	input from Encoder Y chA	19	Mx-	connect to j5 Pin 2
10	By	input from Encoder Y chB	20	Mx+	connect to j5 Pin 1



Pins	Pinout j4	
1	Vcc	Logic Power Supply +5V
2	RS	MIcro controller reset Pin
3	SK	Clock (SPI port)
4	MO	MOSI (SPI port)
5	TX	Used to receive (RX) and transmit (TX) TTL serial data using ATmega32U4 hardware serial capability
6	RX	
7	SS	Slave Select (SPI port)
8	MI	MISO (SPI port)
9	VIN	external power source
10	GND	Ground



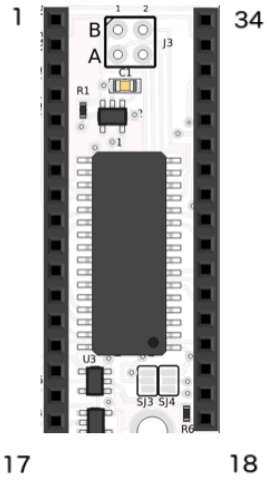
Pins	Pinout j3
A1	Encoder X output chA
A2	Encoder Y output chA
B1	Encoder X output chB
B2	Encoder Y output chB



Pins	Pinout sj3	Pinout sj4
1	GND	GND
2	EN2	EN1
3	Vcc	Vcc

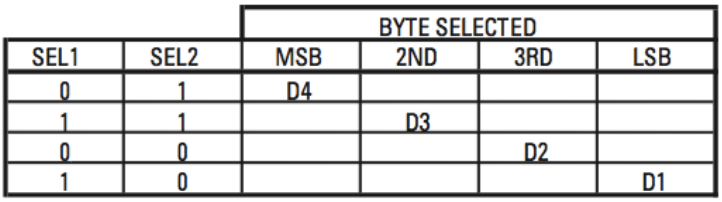
Decoder Count Modes (sj3, sj4 jumper configuration)

		Count Modes		
EN1	EN2	4x	2x	1x
0	0	Illegal Mode		
1	0	On		
0	1		On	
1	1			On



These 34 female pin strips are the input and output connections to the Arduino micro board or another micro controller

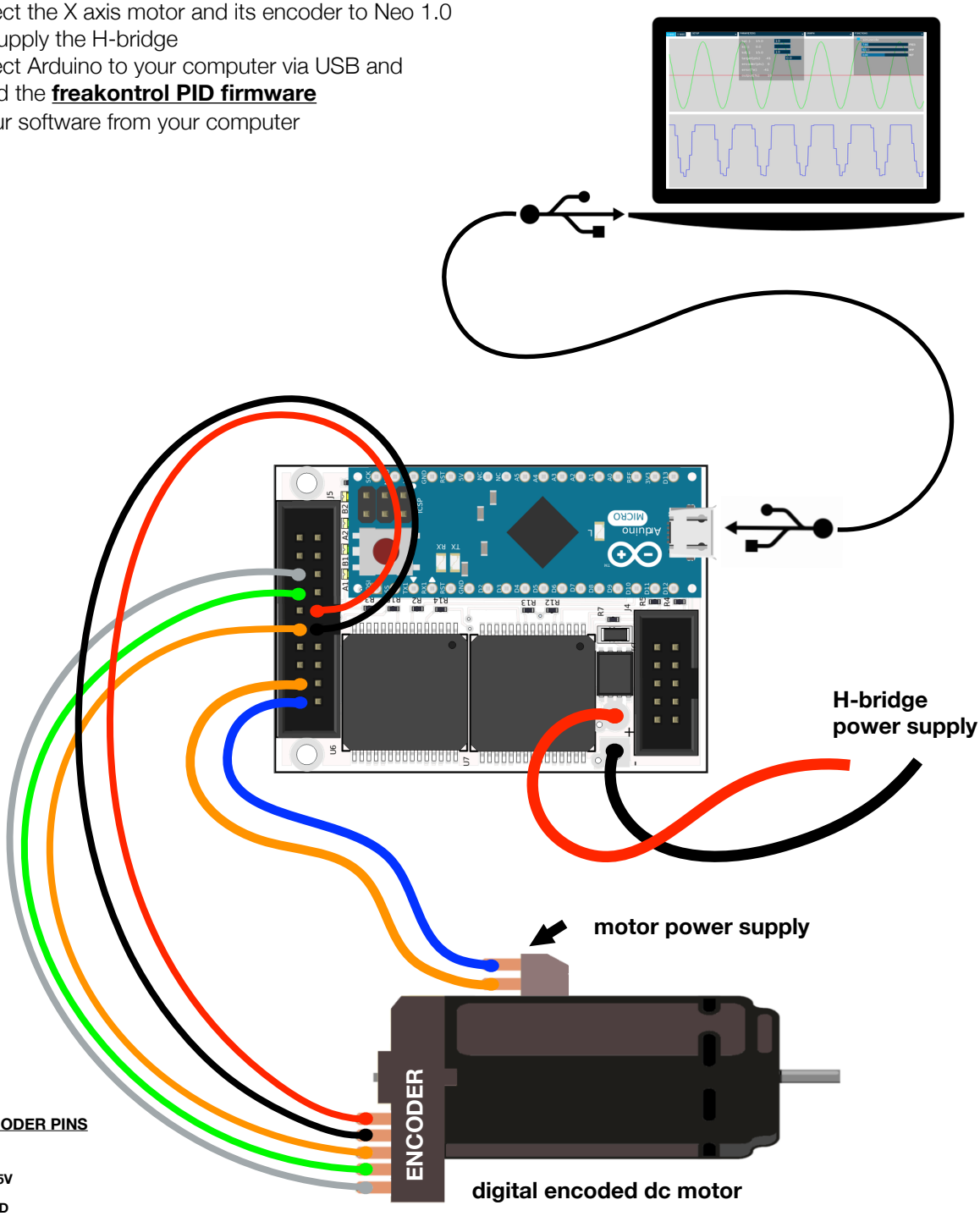
Pins	Pinout µControler connections	
1	MOSI	connected to j4 Pin4
2	SS	connected to j4 Pin7
3	TX1	connected to j4 Pin5
4	RX1	connected to j4 Pin6
5	RST	connected to j4 Pin2
6	GND	ground
7	D7	These LSTTL-compatible tri-state outputs form an 8-bit output ports through which the contents of the 32-bit position latch may be read in 4 sequential bytes. The MSB is read first followed by the rest of the bytes with the LSB is read last.
8	D6	
9	D5	
10	D4	
11	D3	
12	D2	
13	D1	
14	D0	
15	PWM_Y	Y Axis H-Bridge pwm input
16	PWM_X	X Axis H-Bridge pwm input
17	DIR_Y	Y Axis H-Bridge direction

Pins	Pinout µControler connections	
18	DIR_X	X Axis H-Bridge direction
19	NC	no connected
20	NC	no connected
21	X/Y	Select X or Y Axis data to be read
22	SEL1	These CMOS inputs directly controls which data byte from the position latch is en- abled into the 8-bit tri-state output buffer. As in OE/ above, SEL also control the internal inhibit logic.
23	SEL2	
24	OE/	This CMOS active low input enables the tri-state output buffers. The OE/, SEL1, and SEL2 inputs are sampled by the internal inhibit logic on the falling edge of the clock to control the loading of the internal position data latch.
25	RST_Y/	if reset Y axis counter
26	RST_X/	reset X axis counter
27	NC	no connected
28	NC	no connected
29	Vcc	connected to j4 Pin1
30	RST	reset of the micro controller
31	GND	ground
32	VIN	connected to j4 Pin9
33	MISO	connected to j4 Pin8
34	SCK	connected to j4 Pin3

		BYTE SELECTED			
SEL1	SEL2	MSB	2ND	3RD	LSB
0	1	D4			
1	1		D3		
0	0			D2	
1	0				D1

BASIC CONNECTION FOR X AXIS TEST MODE

- connect the Arduino Micro Board to the Neo 1.0
- connect the X axis motor and its encoder to Neo 1.0 and supply the H-bridge
- connect Arduino to your computer via USB and upload the **freakontrol PID firmware**
- run our software from your computer



DIGITAL ENCODER PINS

- +VCC +5V
- GROUND
- INDEX (optional)
- X AXIS CH_A
- X AXIS CH_B