Steer-by-Wire Control System

Adem Kader
Swarthmore College Department of Engineering
Advisor: Erik Cheever

May 2006

Abstract

The automotive industry has already implemented many advanced computer systems in an attempt to increase safety and comfort of drivers. In parallel with these advancements we see a big shift from mechanical systems to electrical systems and steer-by-wire is another implementation that is very promising in terms of safety and functionality. Already, there are some commercial prototypes of such 'by-wire' systems[1] and there is a lot of research, both academic[2] and comercial[3], in the field. For my Engineering Senior Design Projet at Swarthmore College, I chose to work on a steer-by-wire system to gain more insight into control theory and I thought the double-control system that provided the crucial feedback to the driver was an interesting engineering problem.

Contents

1 Introduction

1.1 Definition and Benefits of Steer-By-Wire

A steer-by-wire system aims to eliminate the physical connection between the steering wheel and the wheels of a car by using electrically controlled motors to change the direction of the wheels and to provide feedback to the driver.

Today's automobiles benefit more and more from the many uses of electronic systems. The integration of a steer-by-wire system can enhance these systems in many ways. In particular, the handling and the safety of the cars can be improved significantly. Since a steer-by-wire system is easily modifiable, different drivers will be able to adjust the system to accommodate their styles and this will enhance handling. I addition, disabled people and the elderly will benefit immensely from steer-by-wire because they will be able to situate the steering wheel to meet special needs. Traction control systems are very closely tied with driving safety and they can be enhanced with steer-by-wire vastly. For instance, in a situation where the car starts oversteering (when the rear of the vehicle heads towards the outside of the corner), the natural instinct of many inexperienced drivers is to turn the steering wheel towards the inside, which in turn causes more oversteer. A steer-by-wire system could be modified to take control in a situation like this to steer to the outside.

Since there are virtually no physical connections between the steering wheel and the wheels, a steer-by-wire system can be implemented on different cars easily. The steering wheel could be placed on either side of a car (or anywhere else). Both of these improvements would reduce costs of production and allow a wider range of designs.

The downsides of a steer-by-wire system are maintenance and power cost. Concievably steer-by-wire will use more power than the currently used system, however considering the power consumption of power steering the power cost will be insignificant. There might also be more electrical failures, but presumably steer-by-wire systems will last longer because they have fewer mechanical parts and will improve safety and therefore help the overall maintenance costs.

1.2 Introduction to the Project

I chose to work on a steer-by-wire system control because it is an interesting double-control problem. As seen in Figure 1, my system has two rotary encoders that provide angular information to a computer controller through two microcontrollers. Then the computer controller regulates two external controllers that are necessary for voltage and current adustments. The adjusted output drives the feedback actuator and the steering actuator to their desired positions. Whenever there is a difference in the angles of the two motors, the system tries to bring the difference back to zero, which is the equilibrium and the desired confiuration.

The system has two different motors with different parameters that depend on each other and this necessitates two controllers that have to be merged into one big controller that is capable of reacting to outside disturbances. The outside disturbances are the driver and the road. The driver controls the smaller of the two motors in the system since it is easier to steer a motor with lower torque. The road might cause quick disturbances (for instance when a car hits a curb and the wheels instantly change direction) and clearly these are unwanted and dangerous disturbances so a motor with high torque is required to control the wheels in order to minimize destructive interferences.

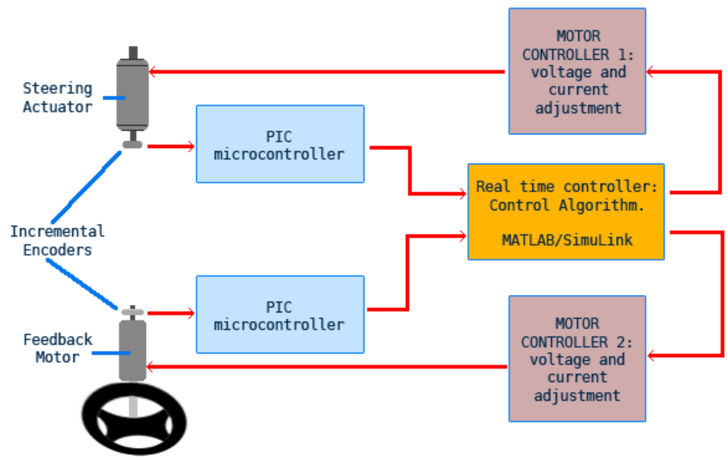


Figure 1: Overview of system

2 Design

2.1 System Overview

The steer-by-wire system consists of two main parts. The steering section consists of the steering wheel, the feedback actuator and the feedback actuator angle sensor. The wheel section contains the wheels, the rack and pinion, a steering actuator and the pinion angle sensor. Figure 2 shows the system components. In my system I only demonstrated the double control mechanism and did not implement it in a rack and pinion configuration.

The feedback angle sensor provides the steering actuator with its primary input signal and the pinion angle sensor provides the feedback motor primary signal. The small size of the feedback motor lets the driver rotate the steering wheel with little difficulty. As soon as the driver starts steering, the control mechanism tries to push the steering wheel back into place (and the wheels into the position dictated by the current position of the steering wheel) and this mimics the resistive force of a real steering wheel. However, changing the proportional constant of the feedback motor can make it harder/easier for the driver to steer and allows for adjustable steering (with some drawbacks such as more vibration).

2.2 Physical Components

2.2.1 Steering Actuator

The steering actuator needs to be very powerful in order to turn the wheels of a car when the car is loaded. Minimizing the effects of unwanted disturbances also requires a powerful motor. In the design, I wanted to use a small wagon as a model and measurements showed that a motor with a torque of about 80lb-in was necessary. Ideally this motor would be a brushless DC motor in order to reduce noise and maximize motor life. However, the high cost and low availability of brushless motors led me to acquire the DC motor shown in Figure 3. This is a Groschopp 50757, a 88.9 lb-in

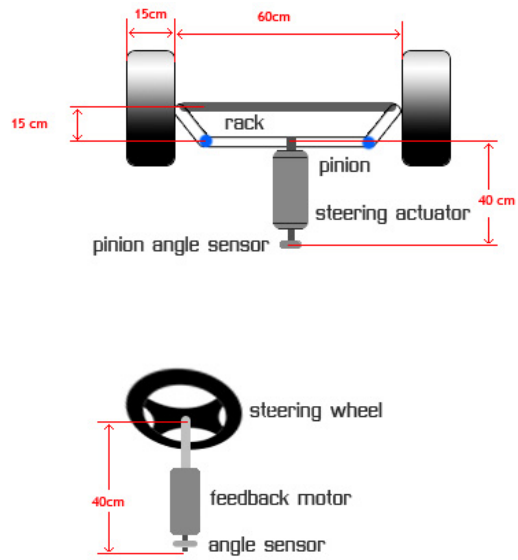


Figure 2: Components of a steer-by-wire system

motor that uses a 12V source and draws approximately 3.4 amps.

Initially, I wanted to use a small wagon to implement my project and show that I could steer the wagon with a person seated on it. In order to find the right size motor, I put a 200lb weight located in the center of the wagon body and used a 39.5 inch crank to turn the rack of the wagon. With the wheels on a painted floor the force required was about 1.9lbs (and it never exceeded 2lbs). Thus the total torque was measured to be $39.5 \cdot 2 \approx 80lbs$.

Although this motor has sufficient torque, the maximum rotation speed is limited to 13.3 RPM. Using only 13.3 RPM, it is impossible to simulate rapid movements experienced in a car, therefore this motor requires gears to increase rotational speed at the expense of reducing force.

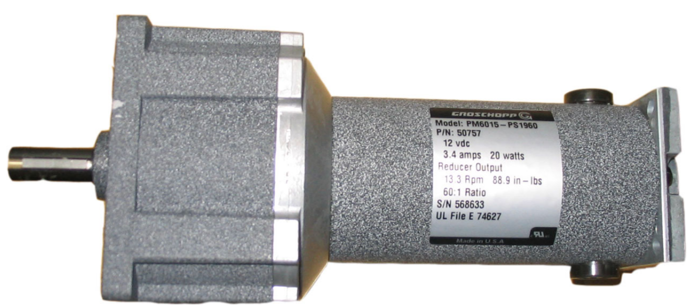


Figure 3: Steering Actuator

The high current characteristics of this motor make it impossible to control using an H-bridge configuration and require a special controller. The specification sheet for this motor can be found in the appendix.

2.2.2 Feedback Motor

Figure 4: Feedback Motor

The feedback motor does not have to be as powerful as the steering actuator. In fact, it has to be much less powerful in order to be turned easily by a driver. I opted to use the motor seen in Figure 4. It was a used motor I acquired from the department. Initially I tested this motor with full power, and the torque it provided was similar to the torque felt in a real car and I concluded that this would be a sufficient motor. The power requirements were also reasonable. It operates at 12V and draws less then 1.5A, so the low current requirements allow for an H-Bridge controller.

2.2.3 Angular Sensors

The angular sensors of the system are very crucial and they need to be very accurate because little perturbations or errors ultimately make the control of the system much harder for a driver. In a real implementation of a steer-by-wire system then would have be very high sensitivity and accuracy in order to minimize risks. In my project I used two optical digital encoders that were used in a previous project. These are BEI Duncan's EX-11 and MX-15 encoders and both of the sensors are seen in Figure 5.

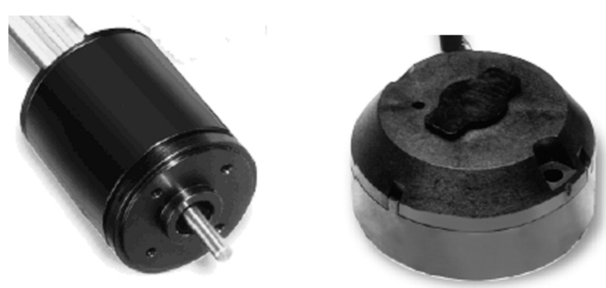


Figure 5: Angular sensors: Incremental encoders

Optical digital encoders' precision and accuracy make them preferable over potentiometers. They output 1024 pulses per revolution in two channels. The channels have a 90-degree offset to indicate direction of rotation. Figure 6 shows the output wave form of the encoders. The specifications of both encoders can be found in the appendix.

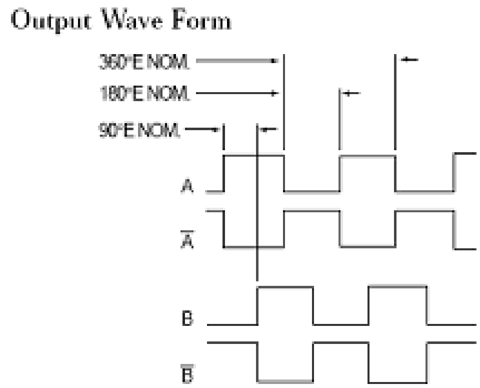


Figure 6: Output wave form of the encoders (taken from the specifications)

2.3 Electronics

Although most of the control mechanism is done in software, I needed some electronics that provided sensible inputs for the controller and regulated the output from the computer to drive the motors. Firstly, since the rotary encoders did not provide absolute angle information, I used PIC microcontrollers to convert line counts to a voltage in the range of 0-5V. Secondly, the output of the digital acquisition board attached to the computer could generate voltages from -5 to +5 volts with a resolution of 1.22 mV and the current output is too low to drive the motors used. Although I never measured the maximum current output, I believe it is in the milliamp range. At least it is essentially zero compared to 3.4A needed by the steering actuator. Figure 1 shows where the electronics are located in the system.

2.3.1 Microcontrollers

The rotary sensors have four possible combinations of outputs and these outputs have to be processed in order to measure the actual angle of rotation. This is done using the PIC microcontrollers. Once the C code for PICs is compiled, the ICD interface lets us install the program onto the PIC and the EEPROM technology allows the program to stay on the PIC even when the power is turned off and on. You can see a picture of the PIC microcontroller in Figure7, and see the schematics of the PIC in Figure 8.

I appropriated C code from a project done by Emery Ku to run the PIC controllers. The code can be found in the appendix.

Since the PIC is a digital device, it only outputs 0 or 5V. In order to get the intermediate values, we have to use the pulse-width-modulated output of the pics. This is done by coupling the output

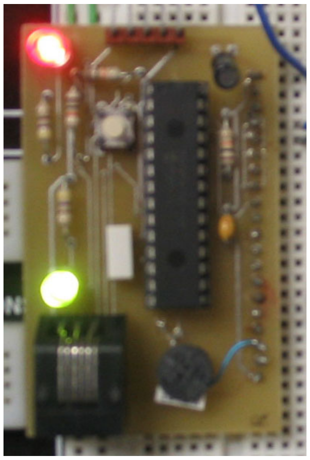


Figure 7: PIC microcontroller.

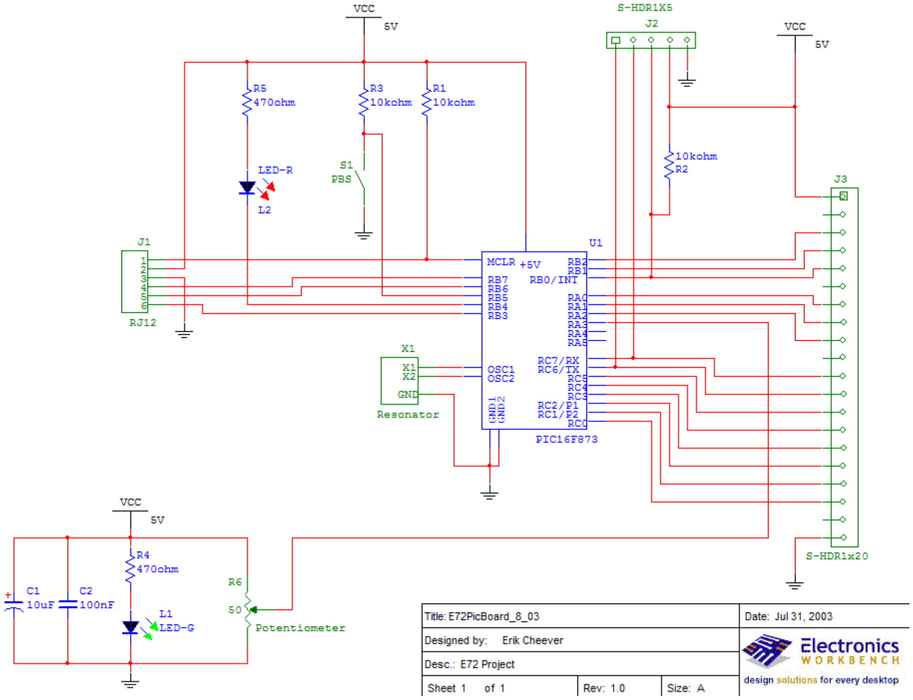


Figure 8: PIC microcontroller schematics (courtesy of Erik Cheever).

with a resistor and a capacitor. Depending on the values of the capacitor and resistor, the noise and speed of the output can be varied. I opted for a 12kΩ resistor and a 33μF capacitor.

2.3.2 Pulse Width Modulation and the H-Bridge

The feedback motor requires low current and this makes the H-bridge configuration coupled with a pulse width modulated input signal (seen in Figure 9) a good choice for a controller. The pulse-width-modulation uses a comparator to compare a triangular wave to a user-specified input. The output is a square wave that has a duty cycle proportional to the specified input. The square wave is very useful because it can be used to switch transistors on and off.

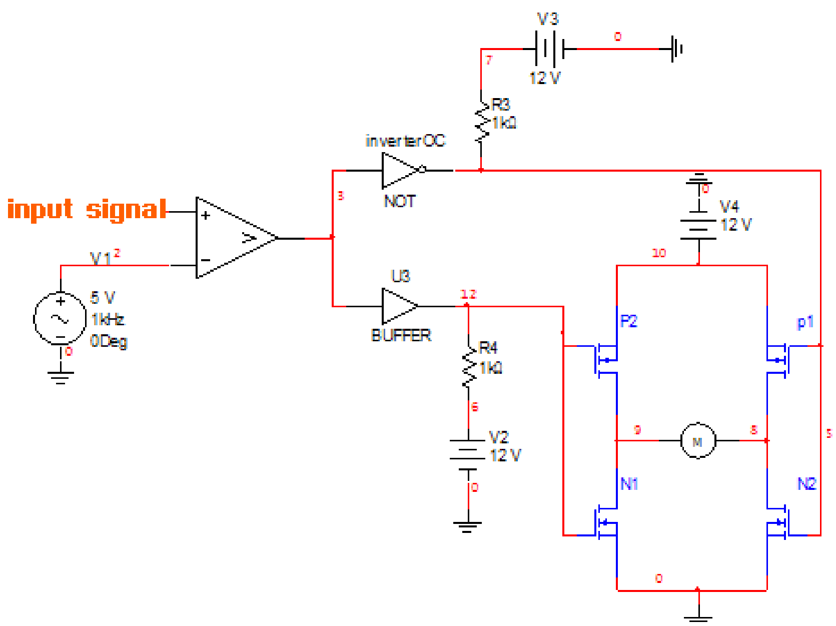


Figure 9: Pulse width modulation and the H-Bridge Configuration

The pulse-width modulated signal goes through an inverter (4017) and a buffer (4016) to adjust the voltage to a range of 0-12V instead of the comparator output range of 0–5V. The outputs of the buffer and inverter are adjusted using a resistor between the outputs and a 12V source. Using the same source for the H-bridge and the output adjustment is a good idea because that way the gate and drain voltages of the mosfets are adjusted accordingly.

The H-Bridge takes its name from the way it looks (see Figure 9). It consists of pairs of P-channel and N-channel mosfets. The right side and the left side of the H-bridge get exactly opposite signals and since the pulse-width-modulated signal runs the gate voltages, we get a very quick switching effect through the motor. Note that at a given time we only have P1 and N1 (which are on opposite sides) or P2 and N2 in operation. The current direction keeps changing and the motor stays idle when the duty cycle is 50%. When the duty cycle increases (or decreases), it allows more current to flow though the motor in one direction than the other and thus the motor starts rotation.

One drawback of the h-bridge is that it dissipates most power when the motor is idle. If we consider the fact that most of the time the steering wheel is idle the h-bridge can be very inefficient in terms of power consumption. However, for our purposes it gets the job done.

2.3.3 Motor Controller for Steering Actuator

As mentioned earlier, the steering actuator is very powerful and draws a lot of current. Therefore a powerful controller is necessary to control it. I chose the KBBC-24M (see Figure 10) because it can be modified easily for use in various motors and can be used in future projects. With the

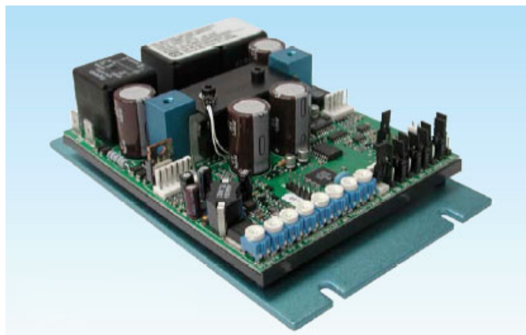


Figure 10: KBBC-24M controller for the steering actuator.

wig-wag option selected, KBBC-24M accepts an input in the 0–5V range where 0 and 5V represent the fastest reverse and forward speeds respectively.

2.4 Computer Modeling

Alhough the control system can be done in hardware, it is much easier to implement the control system in software. I chose to use Matlab because I was familiar with it from previous classes. There were some simulations of steer-by-wire systems done using Matlab[4] so I knew it would be a great choice.

2.4.1 Simulink Modeling

Most of the modeling and control is done using SimuLink and Matlab . The Real-Time Workshop in Simulink allows the user to design the controller in blocks and then compiles the scheme and runs the program in real-time as long as the computer is on. As opposed to past years, when the DAQ Board could only be used for a limited number of data samples (and therefore time), the Real-Time Workshop is a huge improvement.

In the steer-by-wire system, I implement the model seen in Figure 11. Although the controls of both motors seem similar, their inherent differences require different control parameters. The input to both controllers is governed by the difference in the rotary sensor readings. In other words, the error signal is the difference in the angular sensor voltages. Both systems use PID controllers but the parameters vary. Also, since integration causes more noise, the integration constant is very low.

Because the steering wheel angle sensor and the pinion angle sensor will usually be very close to each other, the integral control component does not help the system very much.

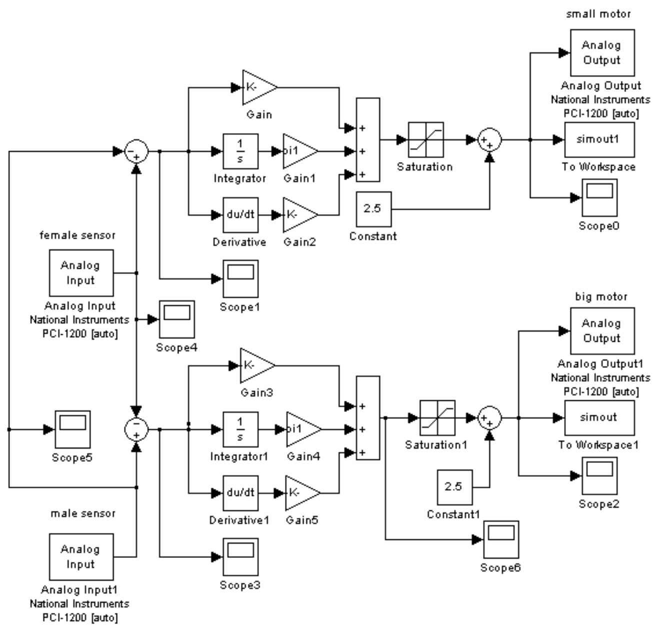


Figure 11: Control model

In Figure 11, you can see that the output of the PID goes through a saturation and a constant of 2.5 is added after the saturation. This is because both motors are at rest when the inputs are 2.5 V and they work in the range of 0–5V. The saturation adjusts the voltage to a range of -2.5–2.5V and the addition of 2.5V puts the final controller output in a range of 0–5V.

3 Testing

3.1 Testing Procedure

Once I implemented my system with all individual parts working, I had to find optimal constants for the PID controller and adjust the saturation levels of the PID output. First I tested each motor individually, and observed their behavior for step inputs. I saw that the steering motor was very slow and that the controller output would be at a maximum (or minimum for reverse) most of the time. On the other hand, the feedback actuator would spin very fast with controller outputs of 5V or 0V, so I adjusted the saturation levels to -0.75–0.75V. With this saturation level, the input to the H-bridge was between 1.75V and 3.25V. This still provided enough feedback and a fast response.

Since the controller output was saturated most of the time, I did not do simulations on the computer, and instead I found the PID constants by trial and error. However this did not mean that I would have to rely on my luck to find the right values. Since I was familiar with the effects of changing control parameters, I could reach good values pretty soon and build on them.

3.2 Determining PID constants

There are a few tuning methods to find the constants (or get in the ballpark of good constants). I first used the Ziegler-Nichols method introduced by John G. Ziegler and Nathaniel B. Nichols. This method starts with finding a critical gain K_c (the gain for which a proportional controller using this gain starts to oscillate). Then at the oscillation frequency it finds the oscillation period, P_c. Then the Ziegler-Nichols method suggests the values seen in Table 1.

parameter	K_p	K_i	K_d
value	$0.6K_c$	$P_c/2$	$P_c/8$

Table 1: PID controller constants using the Ziegler-Nichols method

Initially, this gave me some guidance, and I had some success with controlling each of the motors individually, but I noticed that when I put the two systems together, these parameters did not help me much. The feedback actuator would vibrate a lot especially because the steering motor was very slow in reacting. I noticed that if I held the steering rod in place and waited for the steering motor to approach its desired position, the integral control would increase the output to the feedback actuator too much, and releasing the steering rod would cause it to overshoot a lot. When the system was in equilibrium, small differences in the sensor angles caused vibrations in the feedback actuator so eventually, I opted to eliminate the integral component of the feedback motor.

The steering motor controller was very slow in switching from maximum forward to maximum reverse direction, so the control of the motor was very hard. When I was deciding the PID constants, I noticed that all combinations pretty much gave similar results. In the end, I decided to use values for which the step response would cause the motor controller to switch in time to drive the motor in the opposite direction and stop it. In general I followed the guideline in Table 2 to achieve acceptable controller constants. The final values I used can be found in Table 3

Parameter	Rise Time	Overshoot	Settling Time
K_p	decrease	increase	small change
K_i	decrease	increase	increase
K_d	small change	decrease	decrease

Table 2: Effets of increasing parameters: a guideline for picking PID controller constants

	steering motor	feedback actuator
K_p	0.5	0.65
K_i	0.05	0
K_d	0.8	0.4

Table 3: PID controller values for the final design

The Matlab set-up during an operation of the system can be seen in Figure 12.

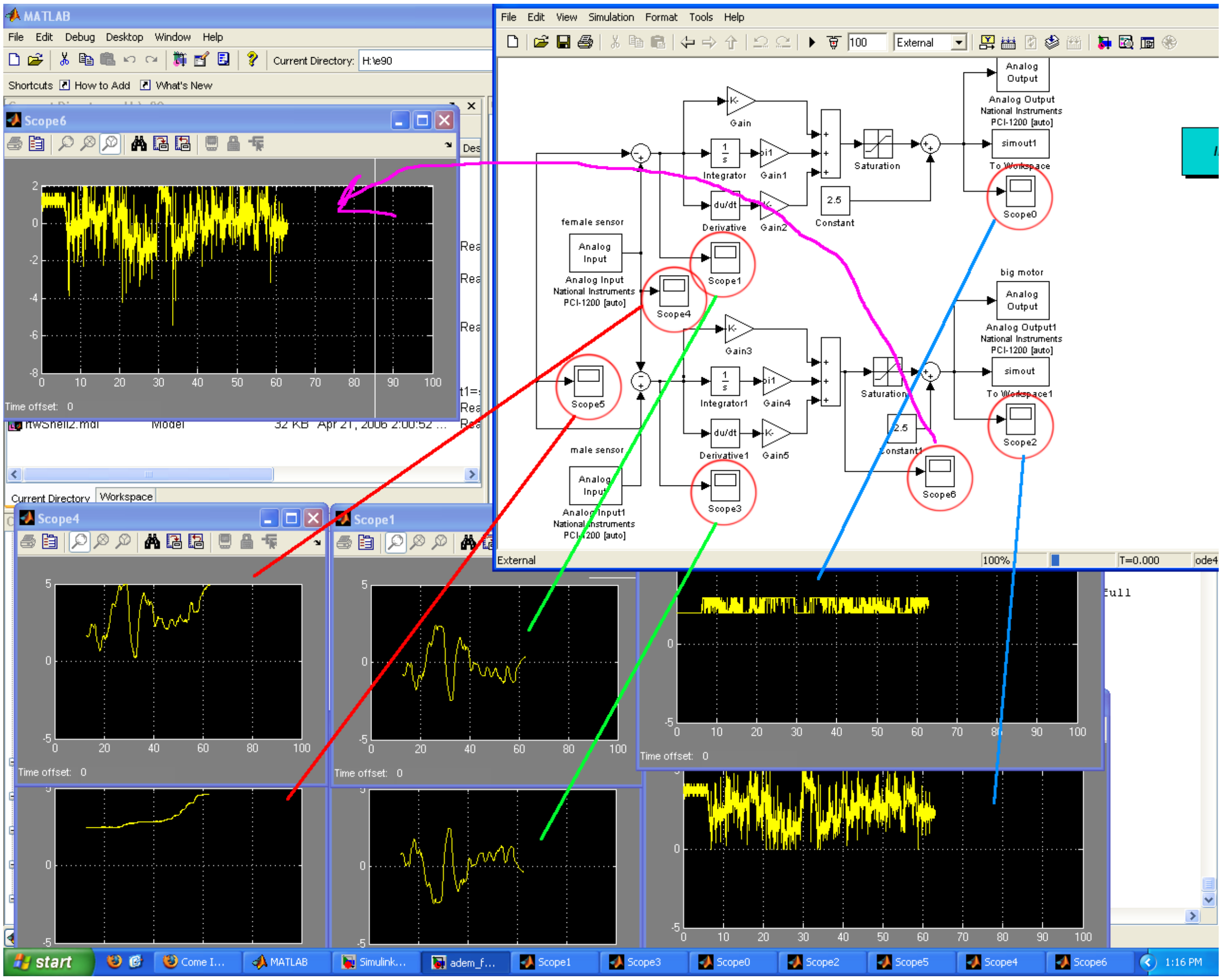


Figure 12: Sample operation Scope 0: Saturated feedback actuator output, Scope 1: error signal for feedback actuator, Scope 2: output for steering motor, Scope3: error signal for steering motor, Scope 4: input from feedback actuator sensor, Scope 5: input from steering angle sensor, Scope 6: Unsaturated output from steering motor PID

4 Discussion

Although there were many problems I encountered with each part of the project, in the end, I was able to show that a steer-by-wire system could work. However, given the slow responses of the system, my setup is far from ideal and needs many improvements. The feedback motor worked very well and to a certain extent it simulated the real driving experience. The steering motor didn't work very well at all, and this can be attributed to the characteristics of the motor and the motor controller. The motor controller had a very slow switching time. It would take about 0.4 seconds to switch from one direction to the other and this really hindered the operation. Also, the steering motor was quite powerful so even though the motor controller were able to switch very fast, the motor would still be slow to react. In a real implementation, it might be a good idea to use two dedicated motors, one for each direction. The steering wheel can also be improved by adding dampers and springs.

I believe that with various improvements, steer-by-wire can achieve some success. Sometimes even very small delays in reaction can be fatal so the delay between steering wheel rotation and actual steering might pose a great threat in the development of SBW systems. However, given other advancements in automotive technology, we see that more and more AI systems are being integrated into cars and combining SBW systems with these can be very promising.

Doinng this project has improved my understanding of control systems greatly and I have gained significant experience in combining electronic and mechanical systems. Although I have not accomplished my initial goals of the project 100%, I believe that there was a lot of progress made and I think that this system can be improved upon.

5 Future Work

The SBW system I build can be improved a lot, but the main problem seems to be with the choice of controllers and motors. For a future project, given better equipment, this system could be implemented in a small model car and can be used for control theory demonstrations. New control systems, such as state-space controls, can be implemented to enhance the performance of the system.

Although not in the near future, given enough resources, this system can be implemented in real roadcars and perhaps be combined with regular steering to take advantage of the safety benefits of a steer-by-wire system.

6 Acknowledgements

I would like to thank my advisor Prof. Erik Cheever for his invaluable help with the project. Without his guidance, patience and encouragement, this project would never be complete. I would also like to thank Prof. Fred Orthlieb, Grant Smith, Ed Jaoudi, Aron Dobos, Danielle Miller and Emily Kan with their guidance and help with mechanical parts of the project. I would like to thank Prof Bruce Maxwell and Prof Cheever for all the engineering classes I took with them (a lot). Finally, I would like to thank all fellow Swarthmore Engineers for being the people they are.

References

[1] Newlaunch.com. Steer-by-wire concept cars exhibited at tokyo motor show.

[2] Kassakian Dominguez-Garcia. Haptic interface for automotive steer-by-wire systems.

[3] Jon Demerly Sanket Amberkar, Farhad Bolourchi and Scott Millsap. A control system methodology for steer by wire systems.

[4] Joachim Langenwalter and Tom Erkkinen. Model-based design with production code generation for steer-by-wire system development.

7 APPENDIX

Miniature Incremental Rotary Optical Encoder

EX11 Series

The EX11 Series was developed to provide a high precision, low cost enclosed shaft encoder for light duty applications. The EX11 offers benefits of the Opto-ASIC design with 1024 line counts in a 1.1 inch diameter size.

Packaged in a glass filled polycarbonate housing with a 1/8" stainless steel shaft and precision bearings, the EX11 provides superior performance at a lower cost.

The EX11 Series is capable of operating over a temperature range of -10°C to +70°C without degradation of signals.

Proven design and Duncan Electronics' experience makes the EX11 perfectly suited for high volume OEM applications, including: robotics, process control and instrumentation.

Standard Features

- Line counts up to 1024 PPR
- RS422 compatible 26LS31 line driver
- 1.1 inch diameter package in servo or flange mount configurations
- 2-year warranty

Figure 1

Servo Mount

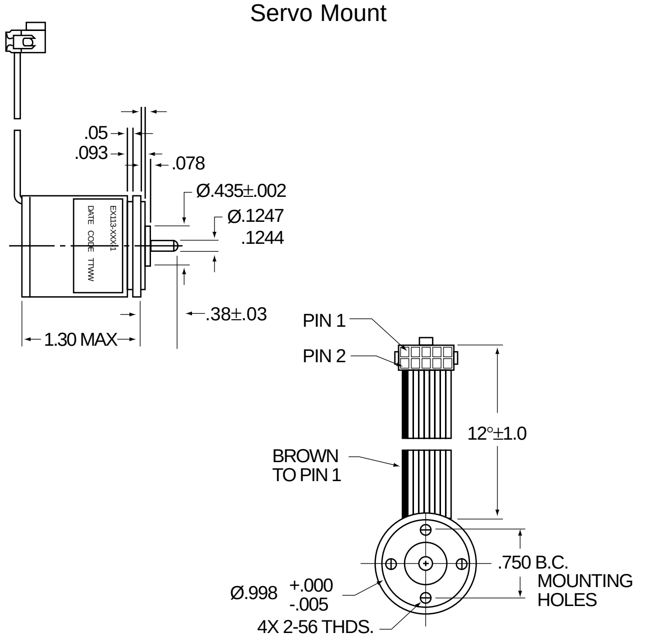
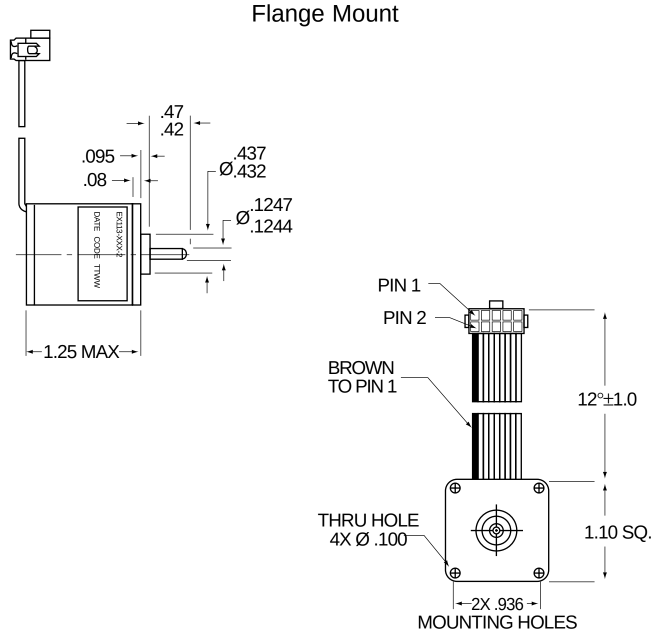


Figure 2

Flange Mount



All dimensions in inches

.xx = ±.02
.xxx = ±.005

BEI DUNCAN ELECTRONICS DIVISION
BEI TECHNOLOGIES, INC.

15771 Red Hill Avenue
Tustin, CA 92780-7303
(714) 258-7500 Fax: (714) 258-4629
email: sales@beiduncan.com
www.beiduncan.com

ISO9001 Certified/QS9000 Compliant

EX11 Series
Miniature Incremental Rotary Optical Encoder

Performance Specifications

Mechanical

Dimensions	see Figure 1
Weight	2.0 oz. (Approx.)
Shaft Diameter	0.1247 +.0000/-.0003
Shaft Load	axial 2 lbs., Radial 1 lb.
Torque, starting	less than 0.4 oz. in.
running	less than 0.2 oz. in.
Inertia	3.0 x 10^{-5} oz. in./sec.

Motor Interface

Servo Mounting Holes	4 places #2-56 @ 90° on 0.75" B.C.
Servo Mount	designed to accommodate motor mount cleat "PIC type" L2-2
Flange Mounting Holes	4 places .100 dia. thru holes
Shaft Coupling	must be flexible (do not hard mount)

Electrical

Code	incremental
Pulses per Revolution	see "Ordering Information"
Supply Voltage	+5 volts ±5% @ 80mA max.
Output Format	dual channel quadrature and index with complements (no index on EX112)
Output Type	TTL differential line driver (26LS31 or equiv.) should be terminated into a line receiver (26LS32, or equivalent circuit)
Rise Time	1.0µsec. max.
Frequency Response	see graph: Fig. 3

Environmental

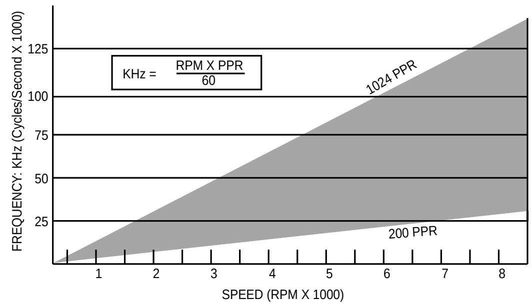
Temperature	operating: -10°C to +70°C storage: -40°C to +125°C

Termination

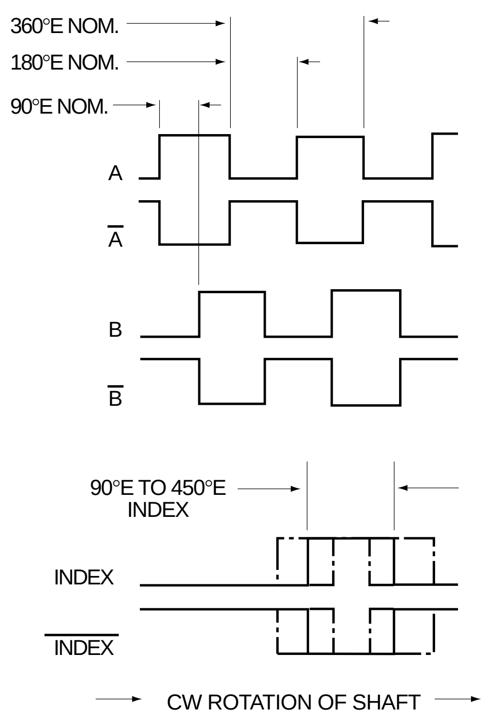
Type	28 AWG flat ribbon cable with 10 position connector Berg P/N 65863-165 or equiv. Mates with Berg P/N65863-165 or equiv. (mating connector not provided)

Pin No.	Color	Signal
1	Brown	N/C
2	Red	+5V
3	Orange	$\overline{B}$
4	Yellow	B
5	Green	$\overline{\text{index}}$
6	Blue	index
7	Violet	$\overline{A}$
8	Gray	A
9	White	N/C
10	Black	Ground

Figure 3



Output Wave Form



- EX112 OUTPUTS A & B ONLY
- EX113 OUTPUTS A, B & INDEX ONLY
- EX116 OUTPUTS AS SHOWN

Ordering Information

EX11 X - XXXX - X

Basic Model No.

Output Format
- 2 = Quadrature
- 3 = Quadrature w/index
- 6 = Quadrature w/index & complements

Pulses Per Revolution (PPR)
- -200, -256, -500,
- -512, -1000, -1024

Mounting
- 1= Servo mount
- 2 = Flange Mount

EXAMPLE: EX113-500-2

BEI DUNCAN ELECTRONICS DIVISION
BEI TECHNOLOGIES, INC.

15771 Red Hill Avenue
Tustin, CA 92780-7303
(714) 258-7500 Fax: (714) 258-4629
email: sales@beiduncan.com
www.beiduncan.com

Modular Incremental Rotary Optical Encoder

MX15 Series

By the time you have read this first sentence, you could have installed BEI's model MX15 INSTA-MOUNT™ modular optical encoder. In addition to its quick and easy installation, the MX15 is designed to operate with jitter-free output signals without tight controls on shaft endplay, runout or perpendicularity. The new INSTA-MOUNT™ encoder is capable of operating within a temperature range of -10˚ to +70˚C, requiring less than 30 milliamps of L.E.D. current, without degradation of output signals and is short circuit protected. The MX15 is perfectly suited for motor manufacturers and other high volume OEMs.

BEI's INSTA-MOUNT™ Series encoder offers 5V TTL compatible quadrature outputs with index and complements as options. Axial shaft movements during operation, of ±0.010", will not adversely affect the output signals. Shaft runouts of 0.005" TIR can also be absorbed by this device without affecting output signal performance.

Standard Features

- Resolutions to 1024 PPR
- Quick and easy installation
- Tolerant of axial shaft movement often associated with less expensive motors
- Jitter-free outputs
- Index options
- Increased MTBF (lower component count)
- 26LS31 line driver output from MX156
- High Frequency response
- 2-year warranty

Figure 1

(MX152/MX153)

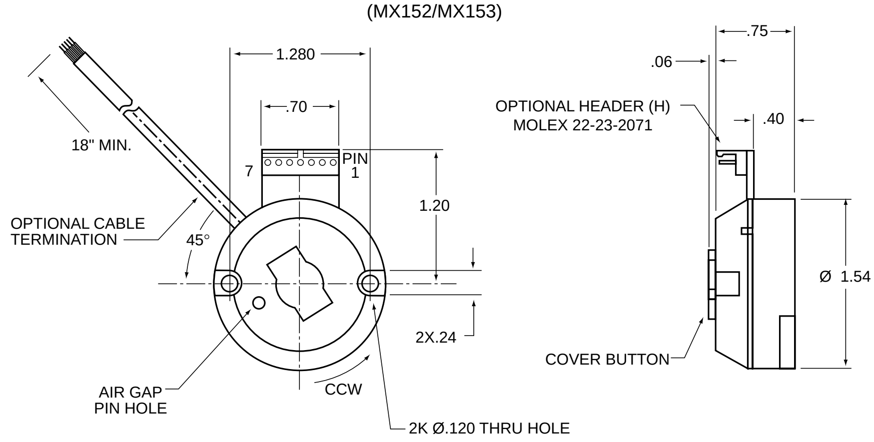
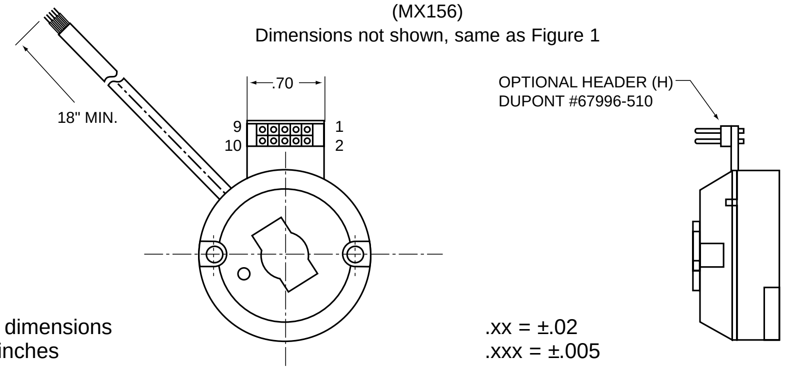


Figure 2

(MX156)

Dimensions not shown, same as Figure 1



All dimensions in inches

.xx = ±.02
.xxx = ±.005

BEI DUNCAN ELECTRONICS DIVISION
B E I T E C H N O L O G I E S, I N C.

15771 Red Hill Avenue
Tustin, CA 92780-7303
(714) 258-7500 Fax: (714) 258-4629
email: sales@beiduncan.com
www.beiduncan.com

ISO9001 Certified/QS9000 Compliant

MX15 Series
Modular Incremental Rotary Optical Encoder

Performance Specifications

Mechanical

Dimensions	see Figure 1
Weight	2.0 oz.
Moment of Inertia	2.6 x 10^{-5}oz in sec^2
Bore Size	see "Ordering Information"

Motor Interface

Mount Holes	#2-56 threads @ 180° on 1.280 dia. B.C.
Mount Hardware	#2-56 x 3/4 in. long (provided)
Perpendicularity Shaft to Mount	±0.002" TIR
Shaft Runout	0.005" max (each 0.0001 degrades accuracy by 0.5 arc minutes)
Shaft Endplay Dynamic or Static	±0.005"
Shaft Finish	16 microinches or better End must be chamfered or rounded
Shaft Tolerance	0.0002"/-0.0007" (e.g. Ø.2493/.2498)
Shaft Length	0.45" minimum (remove cover button for motor through-shafts)

Electrical

Code	incremental
Pulses per Revolution	see "Ordering Information"
Index Pulse Options (no index on MX152)	ungated index (U) gated index (G)
Supply Voltage	5 volts ±5% @ 80mA max.
Output Format (MX152 & MX153)	dual channel quadrature and index (no index on MX152)
Output Format (MX156)	dual channel quadrature and index with complements
Output Type (MX152 & MX153)	square wave TTL. 16mA sink 500µA source. Short circuit protected
Output Type (MX156)	TTL differential line driver (26LS31 or equiv.) should be terminated into a line receiver (26LS32, or equivalent circuit)
Frequency Response	see graph: Fig. 3
Rise Time	1.0µsec. max.

Environmental

Temperature	operating: -10°C to +70°C storage: -40°C to +125°C
Enclosure	unsealed housing unit must be protected from harsh environments

Termination

Terminal Board (Header)

(MX152 & MX153) Pinout

Pin #	Signal	Pin #	Signal
1	N/C	5	data B
2	index (MX153)	6	data A
3	N/C	7	ground
4	+5 volt		

(MX156) Pinout

Pin #	Signal	Pin #	Signal
1	+5 volt	6	data $\overline{A}$
2	+5 volt	7	ground
3	$\overline{\text{index}}$	8	ground
4	index	9	data B
5	data A	10	data $\overline{B}$

Round Shielded Cable

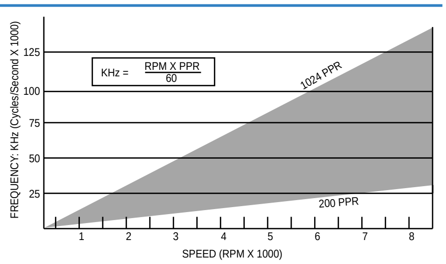
(MX152 & MX153) Color Code

Color	Function	Color	Function
Red	+5 volt	Green	data B
Black	ground	Orange	index (MX153)
White	data A		

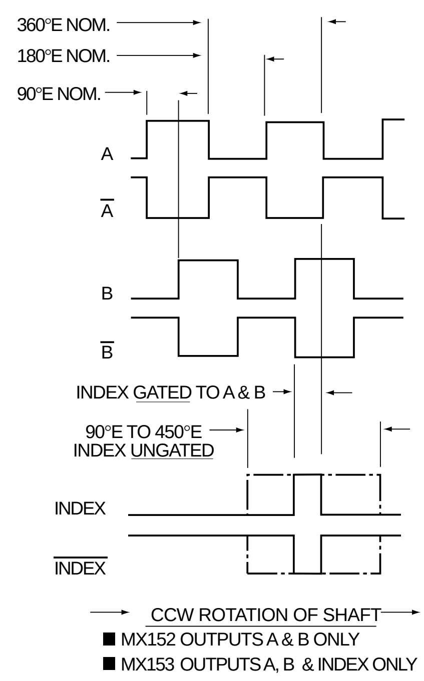
(MX156) Color Code

Color	Function	Color	Function
Red	+5 volt	Green	data B
Black	ground	Wht/Blk	date $\overline{B}$
White	data A	Orange	index
Blue	data $\overline{A}$	Red/Blk	$\overline{\text{index}}$

Figure 3



Output Wave Form



Ordering Information

MX15 X - XX - XXXX - X - X

Basic Model No.

Output Format
- 2 = Quadrature
- 3 = Quadrature w/index
- 6 = Quadrature w/index & complements

Bore Size
- 25-.25", 38-.375"
- 6M-6mm, 8M-8mm

Pulses Per Revolution (PPR)
- 500, 512, 1000, 1024

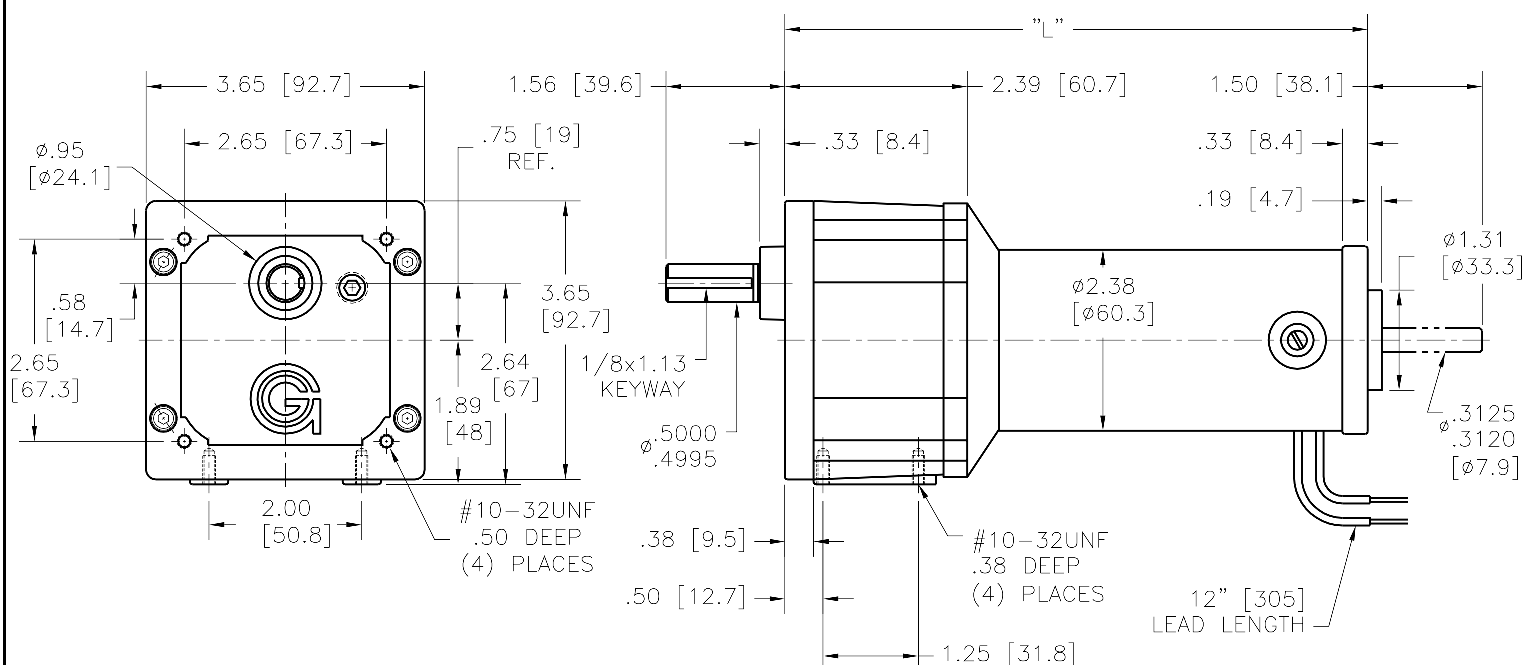
Index Option
- G = gated to data A & B
- U = ungated

Electrical Termination
- T = terminal board
- H = terminal board w/header
- P = round shielded cable

EXAMPLE: MX153-25-500-U-P

BEI DUNCAN ELECTRONICS DIVISION
BEI TECHNOLOGIES, INC.

15771 Red Hill Avenue
Tustin, CA 92780-7303
(714) 258-7500 Fax: (714) 258-4629
email: sales@beiduncan.com
www.beiduncan.com



NOTES:

1. ALL DIMENSIONS ARE NOMINAL.
2. [mm] = MILLIMETERS.

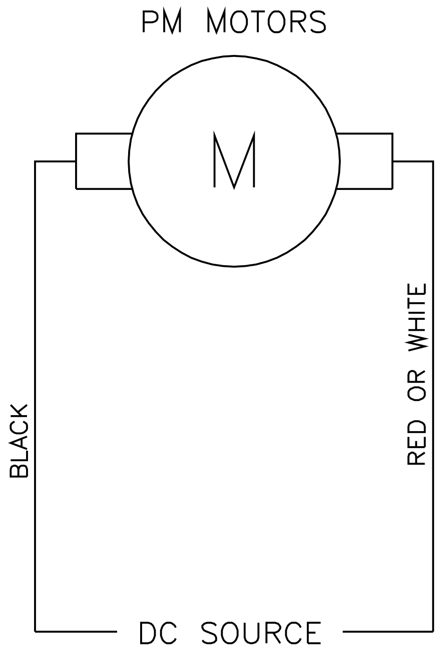
GROSCHOPP
ELECTRIC MOTORS • REDUCERS • GEAR MOTORS • MOTOR SETS

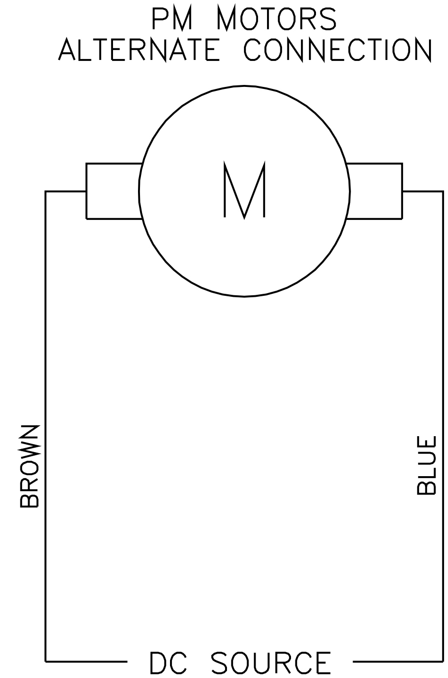
MODEL	"L"
PM6013-PS19__	6.81 [173]
PM6015-PS19__	7.64 [194]

PM601_-PS19__
CATALOG STANDARD

REV: 1
DATE: 7-28-03

AS-172





FOR COUNTER CLOCKWISE ROTATION WHEN FACING DRIVE END; CONNECT BLACK LEAD TO NEGATIVE AND RED OR WHITE LEAD TO POSITIVE. REVERSE LEADS FOR CLOCKWISE ROTATION.

FOR COUNTER CLOCKWISE ROTATION WHEN FACING DRIVE END; CONNECT BROWN LEAD TO NEGATIVE AND BLUE LEAD TO POSITIVE. REVERSE LEADS FOR CLOCKWISE ROTATION.

ISSUE	REVISION	DATE	BY	AUTH.
2	029-05; ADDED WHITE TO PM MOTORS DIAGRAM &	3-10-05	WM	RD
	PM MOTORS ALTERNATE DIAGRAM TO PRINT			
1	ORIGINAL	1-23-91	WK	RD

DRAWN BY	WK	1-23-91	.X ±.015	ANGLES ±3°
CHECKED BY	RD	1-23-91	.XX ±.010	FRACTIONS ±1/64
APPROVED BY	RD	1-23-91	.XXX ±.005	DIMENSIONS IN INCHES
MAT'L.				
CUST.			CONNECTION DIAGRAM	
MODEL	PM MOTORS			

GROSCHOPP
ELECTRIC MOTORS•REDUCERS•GEAR MOTORS•MOTOR SETS

AS-172

GROSCHOPP
ELECTRIC MOTORS • GEARMOTORS • MOTOR SETS

420 15th Street
Sioux Center, IA 51250-2100
Voice: 800-829-4135

info@groschopp.com
www.groschopp.com
Fax: 712-722-1445

MOTORTEC™

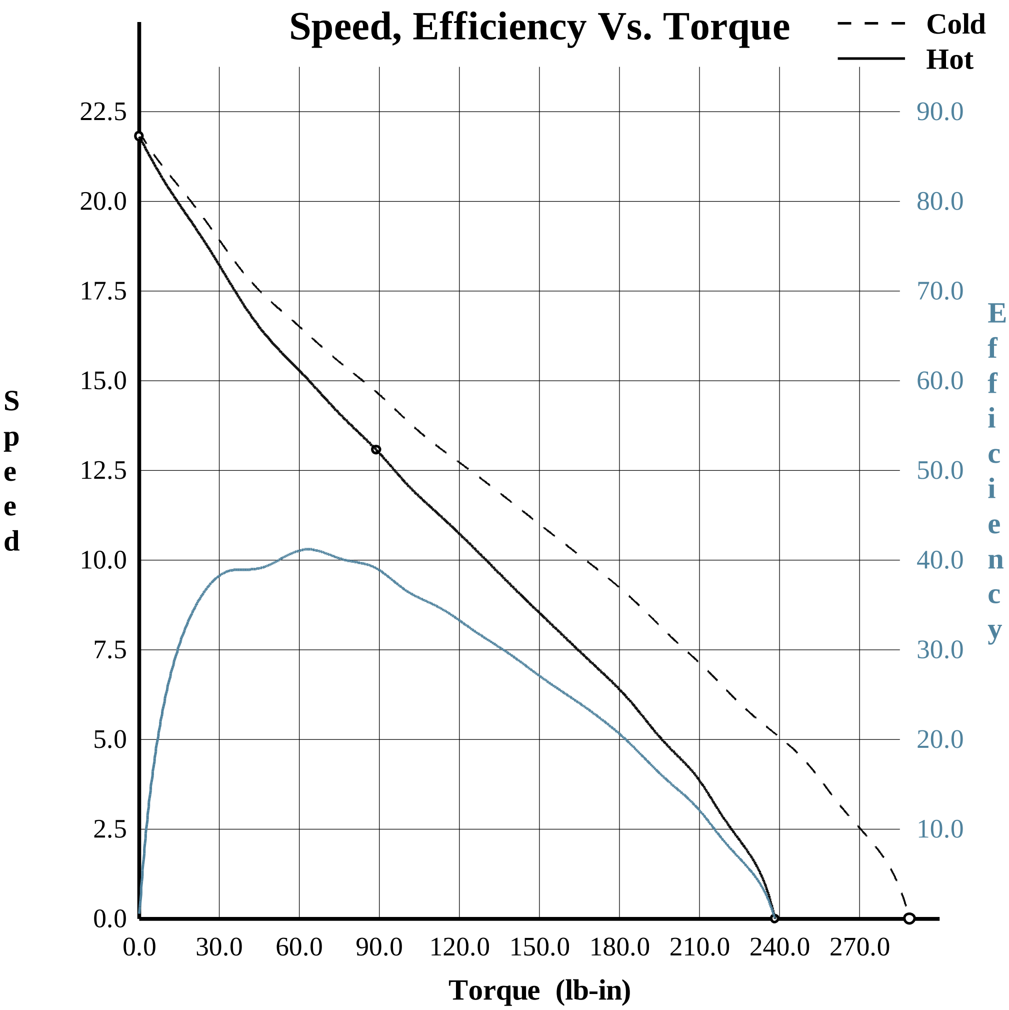
DATA SHEET

Apr 26, 2006

Catalog Part # 50757

Customer: GG-Web
Application:
Quote:

Motor: PM 6015 12 v DC
2 Poles 800 rpm
Gearbox: PS 1900 60:1 Ratio



Rating:
Speed: 13.1 rpm
Torque: 88.9 lb-in
Current: 3.4 amps
Output: 14 watts
Output: 0.0184 HP

Duty Cycle:
On: Continuous
Off:

Efficiency:
Gearbox: 67.7 %
Motor: 57.7 %
System: 39.1 %

Start/Stall Conditions:
Current: 8.15 amps
Torque: 288.90 lb-in

Constants:
Ke: 8.70 v/krpm
Kt: 0.74 lb-in/amp

PM6015-PS1960

Notes:

IMPORTANT:

Please note that there are not implied warranties that the goods shall be merchantable or that they are fit for a particular purpose.

Submitted By: Groschopp.com

800-829-4135
info@groschopp.com

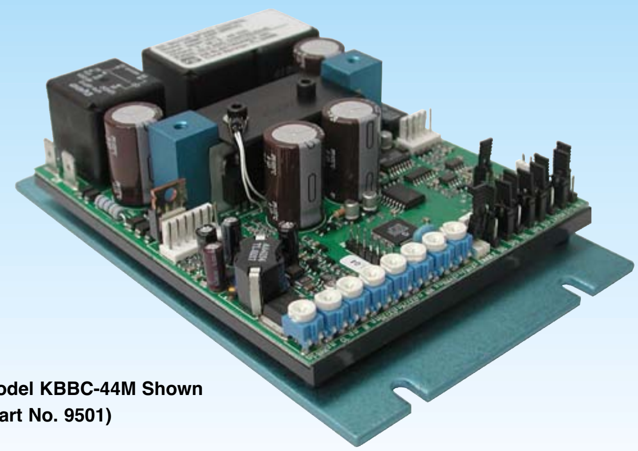
KBBC SERIES

MICROPROCESSOR CONTROLLED
BATTERY POWERED DC/DC
Variable Speed Motor Control

for 12, 24, 36 and 48 Volt
PM and Series Wound DC Motors
thru 2HP Continuous Duty and 4HP Peak Duty

TYPICAL APPLICATIONS

• Scooters • Personnel Carriers • Carts • Electric Boats
• Portable Pumps • Lifts • Floor Polishers



Model KBBC-44M Shown
(Part No. 9501)

STANDARD FEATURES

- **High Frequency PWM Operation:** Reduces motor noise and increases efficiency.
- **Controlled Acceleration and Deceleration:** Provides timed acceleration to set speed and deceleration to zero speed.
- **Diagnostic LEDs:** Provide indication of power on (PWR ON) and control status (STATUS).
- **Built-In Reversing Contactor:** Provides forward/reverse operation with a low power reversing switch or with a center-off throttle potentiometer (wigwag).
- **Run Relay:** Used to turn on or off equipment or signal a warning if a fault has occurred.
- **Brake Driver Circuit:** Powers an optional electromechanical brake (current regulated and short circuit protected).
- **Key Switch Operation with Built-In Battery Power Contactor:** Allows the use of a low power switch to turn control on and off.
- **Inhibit Circuit:** Allows control to be turned off electronically with a separate low power switch.
- **Latching Circuit:** Allows momentary switches to start, stop, and reverse the control.
- **Limit Switch Circuit** (Stop Forward and Stop Reverse)**:** Allows limit switches to be used to immediately stop the control in forward or reverse directions.
- **Single-Ended or Wigwag Potentiometer Control:** Allows the Main Speed Potentiometer to be used as single-ended (zero speed is at 0% rotation) or wigwag (zero speed is at 50% rotation).

PROTECTIVE FEATURES

- **Electronic Current Limit:** Protects the motor and control against overload.
- **Polarity Protected:** Prevents control damage if the battery is wired incorrectly.
- **Short Circuit Protected:** Protects main power transistor from failure due to a short at the motor.
- **Overtemperature Protection:** Reduces control output as the transistors reach maximum operating temperature.
- **Overvoltage Protection:** Will turn off the control if the battery voltage exceeds 125% of nominal.
- **Undervoltage Protection:** Will turn off the control if battery voltage reduces below 65% of nominal.

SAFETY FEATURES

- **Potentiometer Fault Circuit:** Turns the control off if a short, open, or ground occurs at the potentiometer.
- **High Pedal Disable Function:** Prevents control startup until the potentiometer returns to zero.

DESCRIPTION

The KBBC series of battery powered variable speed controls are designed for 12, 24, 36, and 48 Volt PM and Series Wound DC motors. Microcontroller design provides superior performance and ease of tailoring to specific applications. Operating in a regenerative mode, precise and efficient control is obtained using state-of-the-art MOSFET technology. The KBBC operates at a switching frequency of 16 kHz, which provides high motor efficiency and quiet operation.

The KBBC contains many standard features such as current limit, short circuit protection, speed potentiometer fault detector, overtemperature sensing, and undervoltage/overvoltage protection. A variety of trimpots are provided, which can be used to tailor the control to exact specifications. The control also contains LEDs that indicate “power on” and “status.” A DC power contactor allows a low power switch to turn the control on and off. Reversing contactors provide arcless forward, stop, and reverse operation. In addition, a brake driver circuit is used to power an optional electromagnetic brake.

The KBBC can be controlled in several ways, such as single-ended or wigwag speed potentiometer and 0 - 5 Volts DC signal following. The controls contain a built-in heat sink that also serves as a mounting base.

TRIMPOT ADJUSTMENTS

- **Timed Brake Delay (T-BRK):** Sets the delay time before the brake is engaged.
- **Current Limit (CL):** Sets the current limit (overload), which limits the maximum current to the motor.
- **IR Compensation (IR):** Sets the amount of compensating voltage required to keep the motor speed constant under changing loads.
- **Deceleration (DECEL):** Sets the amount of time for the motor to decelerate from the set speed to zero speed.
- **Acceleration (ACCEL):** Sets the amount of time for the motor to accelerate from zero speed to the set speed.
- **Minimum Speed (MIN):** Sets the minimum motor speed.
- **Reverse Maximum Speed (RMAX):** Sets the maximum motor speed in the reverse direction (a % of FMAX setting).
- **Forward Maximum Speed (FMAX):** Sets the maximum motor speed in the forward direction.

PENTA KB POWER™
A Complete Line of Motor Drives

GENERAL PERFORMANCE SPECIFICATIONS

Parameter	Specification	Factory Setting
Input Voltage Range (% Nominal)	75 – 125	100
Intermittent Duty Operation (Minutes)	2	—
Peak Duty Operation (Seconds)	7	—
Overvoltage Shutdown (% Nominal Input Voltage)	125	—
Undervoltage Warning (% Nominal Input Voltage, ± 10%)	85	—
Undervoltage Shutdown (% Nominal Input Voltage)	65	—
Nominal Carrier Frequency (kHz)	16	—
Electromagnetic Brake Delay Trimpot (T-BRK) Range (Seconds)	0.2 – 2.5	1
CL Trimpot (CL) Range (% Range Setting)	0 – 200	150
IR Compensation Trimpot (IR) Range (% Nominal Battery Voltage)	0 – 25	4
Acceleration Trimpot (ACCEL) Range (% Base Speed)	0.1 – 15	2
Deceleration Trimpot (DECEL) Range (% Base Speed)	0.1 – 15	2
Minimum Trimpot (MIN) Range (% Base Speed)	0 – 30	0
Forward Maximum Speed Trimpot (FMAX) Range (% Base Speed)*	60 – 100	100
Reverse Maximum Speed Trimpot (RMAX) Range (% Forward Maximum Speed)	50 – 100	100
Electromagnetic Brake Current Rating (Amps DC)	1	—
Heat Sink Overtemperature Protection Point (°C)	100	—
Deadband in Wigwag Throttle Mode (Volts DC)	± 0.3	—
Wigwag Throttle Signal Input Voltage for Maximum Forward (Volts DC)	2.5 – 5.0	5
Wigwag Throttle Signal Input Voltage for Neutral (Volts DC)	1.2 – 2.5	2.5
Wigwag Throttle Signal Input Voltage for Maximum Reverse (Volts DC)	0	0
Single Ended Throttle Signal Range for Full Speed Forward or Reverse (Volts DC)	0 – 2.5 to 5.0	0 – 5
Timed Current Limit (TCL) Trip Time (Seconds)	7	—
Run Relay Output Contact Rating (Amps at 30 Volts DC, Amps at 125 Volts AC)	1, 0.5	—
Auxiliary Power Connector (P2) Rating (Maximum Amps DC)	10	—
Operating Temperature Range (°C)	0 – 45	—

*FMAX trimpot is also used as an input/output gain potentiometer.

ELECTRICAL RATINGS

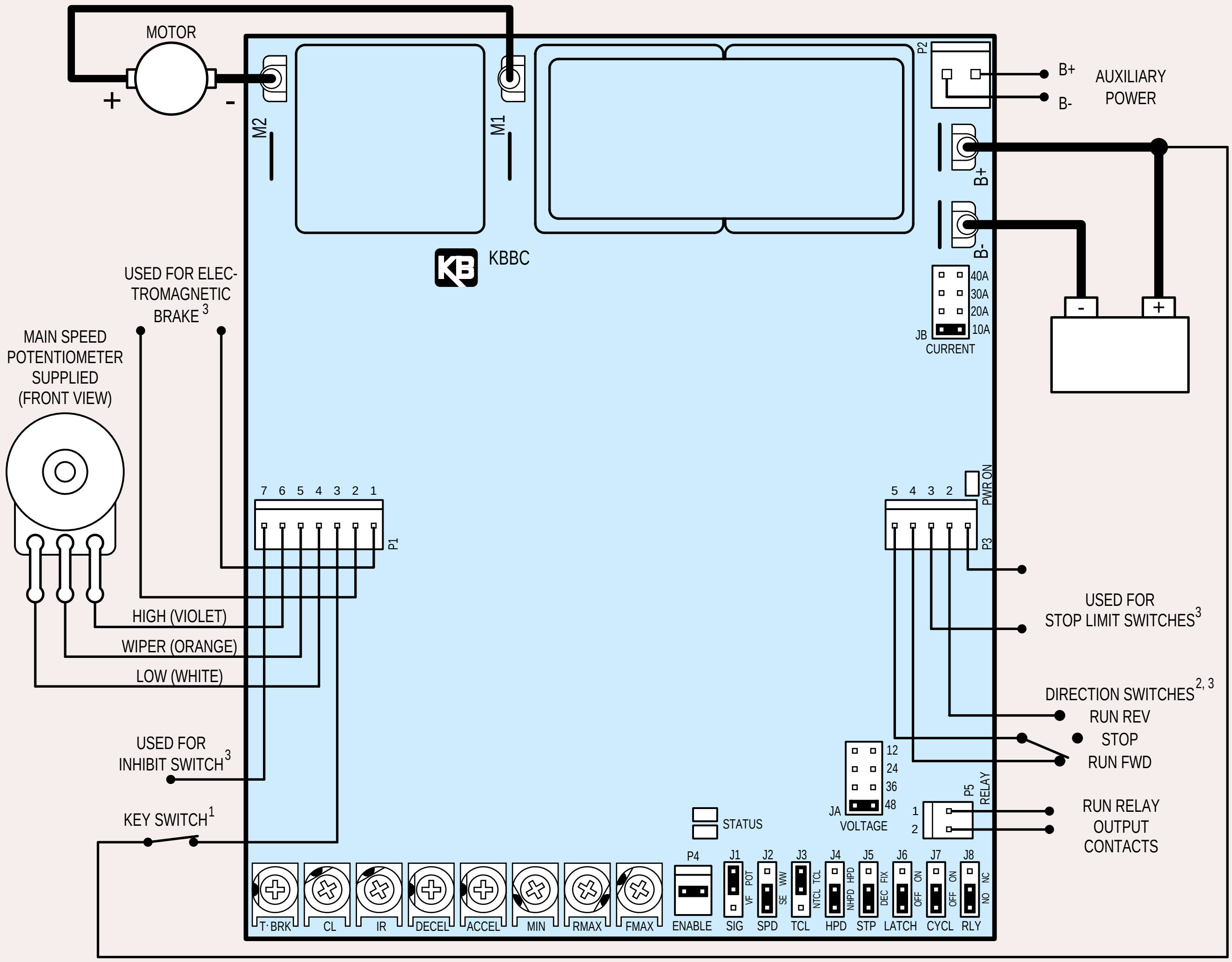
Model No.	Part No.	Nominal Battery Voltage (Volts DC)	Nominal Motor Voltage (Volts DC)	Continuous Duty		Intermittent Duty (2 Minutes)		Peak Duty (7 Seconds)	
				Maximum HP	Amps DC	Maximum HP	Amps DC	Maximum HP	Amps DC
KBBC-24M	9500	12	0 – 12	1/2	40	3/4	60	1	80
		24	0 – 24	1	40	1½	60	2	80
KBBC-44M	9501	12	0 – 12	1/2	40	3/4	60	1	80
		24	0 – 24	1	40	1½	60	2	80
		36	0 – 36	1½	40	2	60	3	80
		48	0 – 48	2	40	3	60	4	80

Note: Custom units are available with various voltages and currents with or without DC Power Contactor or Reversing Contactor.

JUMPER SELECTABLE FEATURES

- **JA - Battery Voltage (VOLTAGE - 12/24/36/48):** Selects nominal battery voltage.
- **JB - Motor Current (CURRENT - 10A/20A/30A/40A):** Selects nominal motor current.
- **J1 - Signal Type (SIG - VF/POT):** Selects voltage following or potentiometer operation.
- **J2 - Speed Potentiometer Mode (SPD - SE/WW):** Selects single-ended or wigwag speed control.
- **J3 - Current Limit Mode (TCL - NTCL/TCL):** Selects non-timed current limit or timed current limit.
- **J4 - High Pedal Mode (HPD - NHPD/HPD):** Selects non-high pedal disable or high pedal disable.
- **J5 - Deceleration Mode (STP - DEC/FIX):** Selects adjustable or fixed (0.1 second) deceleration when a stop command is given.
- **J6 - Direction Switch Type (LATCH - OFF/ON):** Selects maintained or momentary direction commands.
- **J7 - Cycling Mode (CYCL - OFF/ON):** Selects cycling of relay which is used to brake the motor.
- **J8 - Relay Output Contacts (RLY - NO/NC):** Selects normally open or normally closed Run Relay contacts.

CONTROL LAYOUT & CONNECTION DIAGRAM

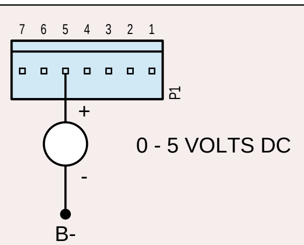


Notes: 1. *If Key Switch is not used, a connection must be made between the red wire of Connector P1 and quick-connect Terminal B+ for the control to operate.*

2. *RUN FWD and RUN REV Direction Switches are not used in wigwag operation.*

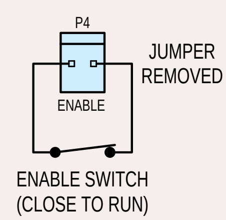
3. *Customer supplied.*

VOLTAGE FOLLOWING CONNECTION

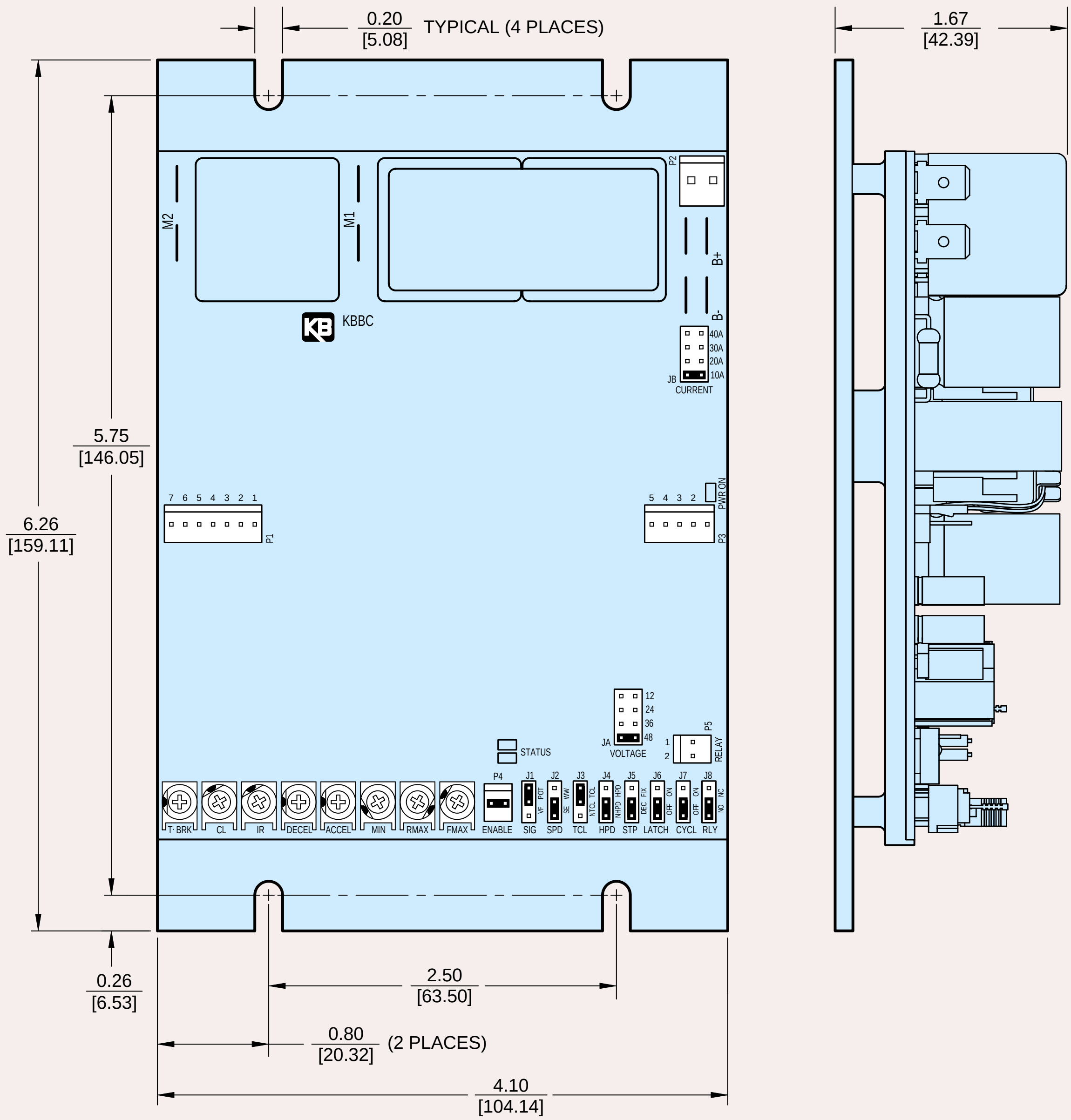


Note: Set Jumper J1 to the "VF" position

ENABLE SWITCH CONNECTION



MECHANICAL SPECIFICATIONS (Inches / [mm])



GREEN AND RED STATUS LEDs

Control Status	Green LED	Red LED	Flash Rate*
Run	On	Off	Slow
Stop	On	Off	Quick
Curent Limit (Warning)	Off	On	Steady
Undervoltage (Warning)	On	On	Slow
Overvoltage/Undervoltage Fault (Shutdown)	On	On	Quick
Overtemperature Fault (Shutdown)	On	On	Slow Alternating
Main Speed Potentiometer Fault (Shutdown)	On	On	Quick Alternating
Motor or Brake Fault (Shutdown)	On	On	Double Quick Alternating
Timed Current Limit (Shudown)	Off	On	Quick

*Flash Rate: Slow = 1 second on / 1 second off. Quick = 0.15 second on / 0.15 second off.

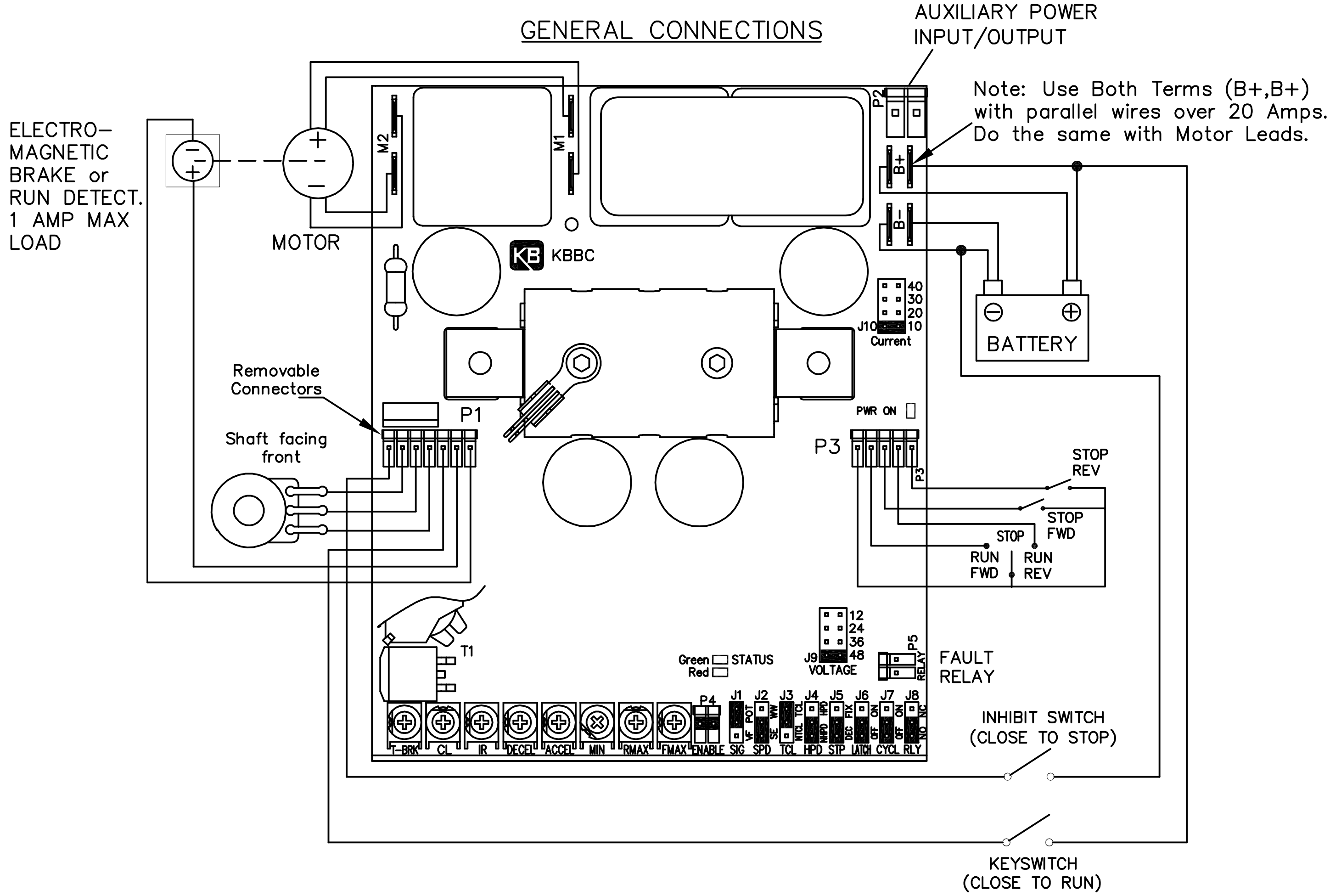
KB ELECTRONICS, INC.
12095 NW 39th Street, Coral Springs, FL 33065-2516 • (954) 346-4900 • Fax (954) 346-3377
Outside Florida Call **TOLL FREE** (800) 221-6570 • **E-mail** – info@kbelectronics.com
www.kbelectronics.com

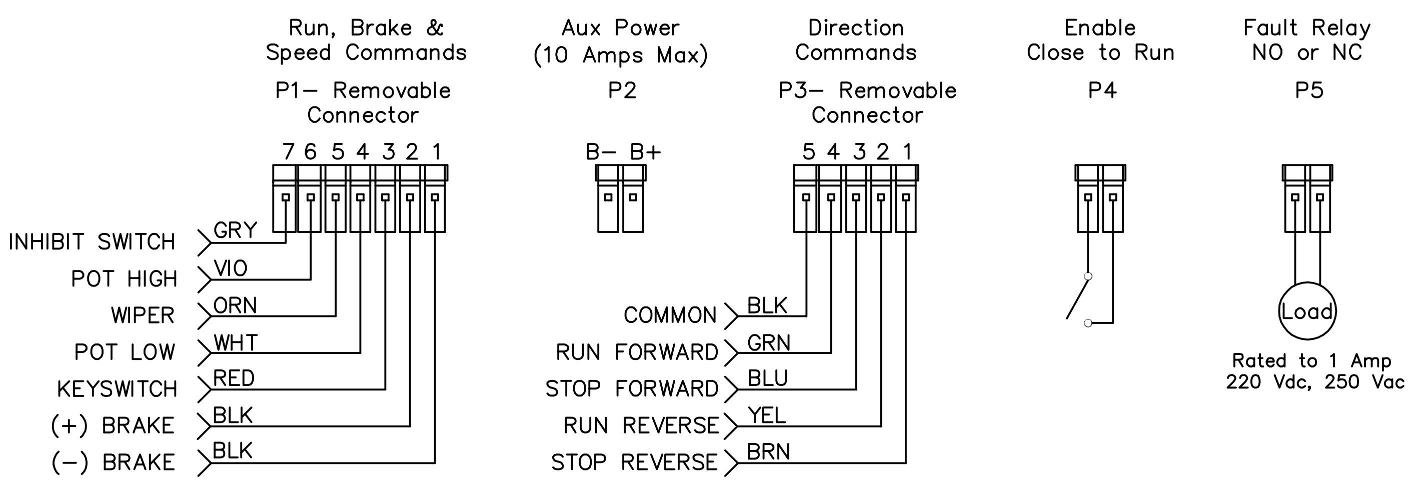
(A42116) – Rev. A – 11/2001

KB ELECTRONICS, INC. • 12095 NW 39th Street, Coral Springs, Florida 33065 USA
(954) 346-4900 • FAX (954) 346-3377

"The Right Control for your Application"

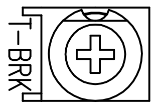
KBBC-MICRO CONNECTION DIAGRAM



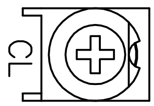


WF–9/25/03

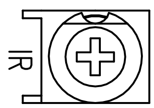
KBBC-MICRO FUNCTIONAL DESCRIPTIONS



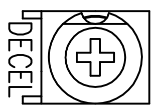
TIMED BRAKE DELAY (T-BRK)- This adjusts time delay before the brake engages after the drive is told to stop. Brake is initiated by Enable, or Keyswitch opening, or inhibit terminals closing.



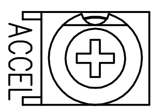
CURRENT LIMIT (CL)- Allows adjustability of current limit setpoint. Typically set at 1.5 X the Motor FLA. When Current Limit engages the status light will indicate by turning red.



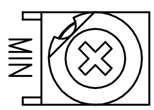
IR COMPENSATION (IR)- Allows adjustment of load compensation for different motors. Smaller motors require more compensation to overcome losses in armature winding. Typically set by checking No Load to Full Load speed changes.



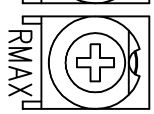
DECELERATION (DECEL)- Allows for controlled deceleration from full speed to zero speed, from 0.1 to 15 seconds as pot is turned clockwise. Decel works with all stop modes except inhibit. When inhibit is used the decel pot has no effect, output will go to zero in 0.1 seconds.



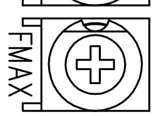
ACCELERATION (ACCEL)- Allows for controlled acceleration from zero to full speed. From 0.1 to 15 seconds as pot is turned clockwise. Accel is active in any turn on condition, including Enable, Keyswitch, or release of inhibit.



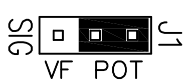
MINIMUM SPEED (MIN)- Sets the minimum speed the motor will run. Factory set to zero speed, but can be adjusted from 0-30% of full speed.



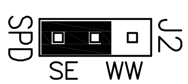
MAXIMUM REVERSE SPEED (RMAX)- Limits the maximum allowable speed in the reverse direction. Range is 50-100% of Forward speed (RMAX is dependant on FMAX setting).



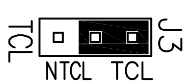
MAXIMUM FORWARD SPEED (FMAX)- Limits the maximum allowable speed in the forward direction. Range is 60-100 % of full speed. Set for Full Travel 5K Pot. For limited travel of Pot (ex. 1/4 rotation=desired full range), FMAX can be turned clockwise to compensate.



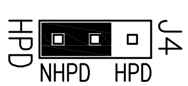
SIGNAL JUMPER (J1)- Voltage Following/ Potentiometer. Selects either potentiometer speed control, or remote 0-5 Vdc signal input. (See page 4).



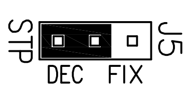
SPEED MODE JUMPER (J2)- Wig Wag/ Single Ended. Allows for choice of Wig Wag speed control, (center of pot is zero speed, clockwise is forward, counter is reverse), or Single Ended, (contact closure chooses direction). (See page 4).



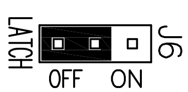
TIMED CURRENT LIMIT JUMPER (J3)- Non Timed Current Limit/ Timed Current Limit. Disables or enables the shutdown of the drive due to motor overcurrent after 7 seconds.



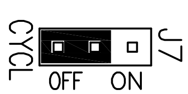
HIGH PEDAL DISABLE JUMPER (J4)- Non High Pedal Disable/ High Pedal Disable. In HPD the main speed pot needs to be reset to zero before motor is allowed to run.



STOP MODE JUMPER (J5)- Decel/ Fixed. Allows stop function (Enable, Keyswitch, Direction Command) to use deceleration trimpot for controlled stop, or fixed stop of 0.1 second.

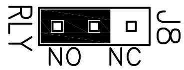


LATCH JUMPER (J6)- Off/ On. Allows choice of how direction commands are activated. If latch is in "OFF" position, direction commands need to be maintained to run. If Latch is in the "ON" position, direction commands are momentary to run or stop. (See page 4).

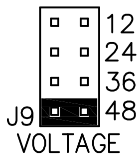


CYCLE (J7)- Off/ On. When the drive is commanded to stop, an output relay closes to short motor leads together. This action will act like a dynamic brake and impede motor travel. If this action is not desired, the cycle jumper can be placed in the "ON" position. (see page 4).

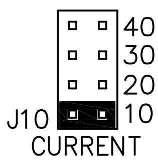
KBBC-MICRO FUNCTIONAL DESCRIPTIONS



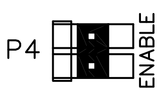
RELAY (J8)- NO/ NC. Used to give option of fault relay output condition.



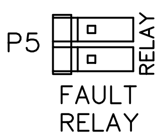
VOLTAGE SELECTION JUMPER (J9)- This Jumper calibrates the Drive to input and output for 12, 24, 36, or 48 Vdc inputs. This Jumper primarily sets up the Overvoltage and Undervoltage set points.



CURRENT SELECTION JUMPER (J10)- This Jumper calibrates the Drive for motors rated 10, 20, 30 or 40 amps. The Current Limit will be set up based on this setting X 1.5. The CL Trimpot can be used to modify this setting.

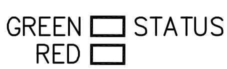


ENABLE (P4)- This connection is an additional method for Run/Stop. Close to Run, Open to stop.



FAULT RELAY CONNECTOR (P5)- Provides a dry contact to indicate fault condition has occured. This output is used in conjuction with Jumper J8 for NO or NC operation. The fault relay will change state when the Keyswitch is applied. The relay will trip on any fault condition: Speed Pot Fault (open lead), Over Temperature Fault, Over/Under Voltage, Motor Brake Fault, Internal Fault (micro failure), and Timed Current Limit. Reset by cycling Keyswitch.

POWER ON LED- This green LED is illumiated when the Keyswitch is engaged.

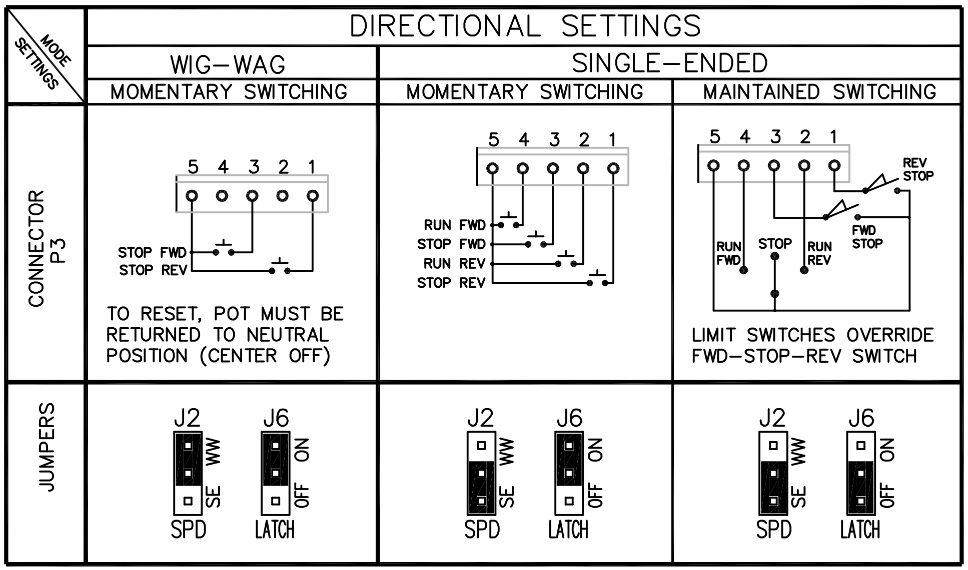


STATUS LEDs- These two LED's are used to indicate drive status. The top LED is green, the one directly below it is red. The Drive condition will be determined by the table shown below:

LED Ref.	Function	Flash Code	LED Color
	Normal Control Operation	Slow	Green
	Stop Mode	Quick	Green
	Speed Pot Fault	Quick	Red/Green (Alternate)
	Temperature Fault	Slow	Red/Green (Alternate)
STATUS	Over/Under voltage	Quick	Red+Green
Green, Red	Undervoltage Warning	Slow	Red+Green
	Motor/Brake Fault	Quick	Red, Red / Green, Green
	Internal Fault)	Slow	Red, Red / Green, Green
	Current Limit	Steady	Red
	TCL (Current Limit Time Out)	Quick	Red
"PWR" (Power)	Normal Control Operation	Steady	Green
Green	Bus & Power Supply Fault	Off	
Slow flash: 1 Sec. on, 1 Sec. off			
Quick flash: 0.15 Sec. on, 0.15 Sec off			

KBBC-MICRO FUNCTIONAL DESCRIPTIONS

DIRECTION AND SPEED COMMAND SETTINGS



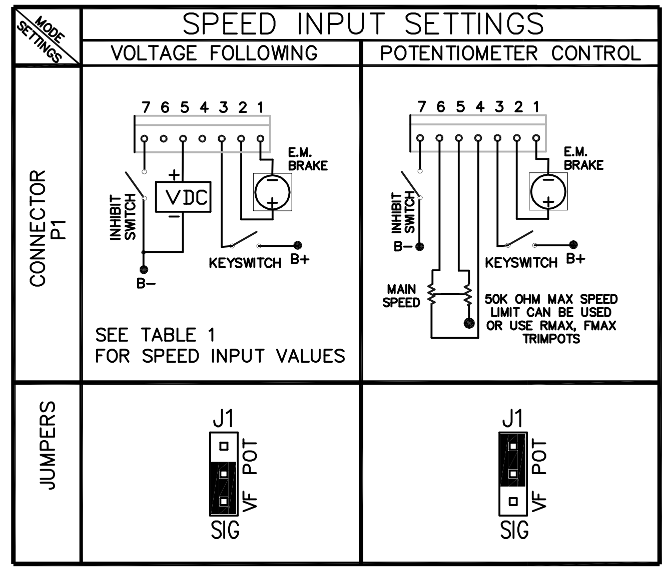


TABLE 1, SPEED INPUTS.

DIRECTION	INPUT SIGNAL (VDC)	
	WIGWAG	SINGLE END
MAXIMUM FORWARD	4.7 + 0.3	4.7 + 0.3 (RUN FWD SELECTED)
NEUTRAL	2.5 ± 0.3	0 + 0.3
MAXIMUM REVERSE	0 + 0.3	4.7 + 0.3 (RUN REV SELECTED)

Notes:

1) Inhibit Function is used for immediate (0.1 sec) deceleration. Close to stop.

2) Keyswitch Function is used to enable power to the drive. "Power On" light will illuminate to indicate the keyswitch is activated.

3) Main speed potentiometer (included) is rated 5 Kohm, 1/3 watt, wirewound.

4) Wig-Wag applications typically use a spring return to center potentiometer (not supplied or available through KB Electronics).

5) Cycle Jumper (J7)- The mechanical life of a relay is 10 million cycles. The Cycle Jumper is useful for repetitive cycling ON/OFF. When in "OFF" position the relay will engage to brake. When in "ON" position it will not. This will limit the use of the relay.

6) Keyswitch Function is used to enable power to the drive. "Power On" light will illuminate to indicate the keyswitch is activated.

7) Momentary Limit Switch bypass protection. If limit switch is engaged in either direction (Stop Fwd, Stop Rev), the same direction run command will not allow continued travel. Ex. Run Fwd, Stop Fwd, will not allow Run Fwd Command until Run Reverse is called. If indexing is required, (Momentary Run Fwd, Momentary Stop Fwd, then Momentary Run Fwd in same direction), the stop reverse (P3-1) must be connected to COM (P3-5), otherwise unit will not go forward the second time.

PIC-C Code

```
//Things we need.
#include <16F873.h>  //ADC set to 10 (open 16F873.h)
#include <STDLIB.H>  //Required by read_adc()
#fuses HS,NOWDT,NOPROTECT,NOLVP
#use delay(clock=10000000)
#use rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7, BRGH1OK) // Jumpers: 8 to 11, 7 to 12

//Define quadrature-based states as grey-code.
#define  state_0 0   //Both encoder outputs A and B are low
#define  state_1 1   //A is high, B is low
#define  state_2 3   //Both encoder outputs A and B are high
#define  state_3 2   //A is low, B is high

//Inputs
#use fast_io(A)      //this requires the set_tris_X command

long int angle;  //Keep track of the angular position (10-bit)
long int control;
int cur_state;
int old_state;

void main()
{
   SET_TRIS_A( 0x0F );
   // A7,A6,A5,A4 are outputs
   // A3,A2,A1,A0 are inputs

   SET_TRIS_B( 0x0F );
   // B7,B6,B5,B4 are outputs
   // B3,B2,B1,B0 are inputs

   setup_ccp1(CCP_PWM);   // Configure CCP1 and CCP2 as a PWM
   setup_ccp2(CCP_PWM);
   setup_timer_2(T2_DIV_BY_1, 127, 1); //  20 KHz signal

   setup_port_a(ALL_ANALOG);
   setup_adc(adc_clock_internal);
   set_adc_channel( 0 );   //A0 is the motor control voltage pin

   angle = 0;

   //Loop to collect/output data
   do {
      control=read_adc();  //Control should be a 10-bit long int, set in #device
       cur_state = input(PIN_B0) + 2*input(PIN_B1);

      if (cur_state != old_state) {
         if (cur_state == state_1 && old_state == state_0 && angle > 200){
            //angle+=64; //Encoder shaft is turning clockwise (theta)
            angle-=256; //Use this for the phi-direction encoder
         }
         if (cur_state == state_3 && old_state == state_0 && angle < 65200) {
            //angle-=64; //Encoder shaft is turning CCW (theta)
            angle+=256;  //The max output comes to 1.04V if left @ +/-64 (phi)
         }
      }

      old_state = cur_state;  //For the next cycle...

/////////////////////////////////  Outputs  //////////////////////////////////

      //Need to convert angle to a voltage to output to DAQ
      set_pwm1_duty(angle/128);

      //Need to output our motor control voltage as PWM:
      set_pwm2_duty(control/2);

    } while(1);
}
```