Application Note AN2278

Automotive BLDC Motor Control for PSoC™

Associated Project: Yes
Associated Part Family: CY8C27443 and CY8C29466
PSoC Designer Version: 4.2

Abstract

This Application Note describes PSoC device implementation of a BLDC (Brushless DC) windscreen wiper motor controller for automotive applications. LIN Bus 2.0 is implemented to provide an interface for controlling wiper operation. This Application Note sufficiently describes the technical details so that the user is familiar with the important aspects of the design and is able to recreate the design specific to their exact requirements. This note also describes LIN Bus 2.0 implementation in detail.

Introduction

Brushless DC motors and LIN Bus 2.0 are gaining popularity in automotive applications. This Application Note describes the implementation of a LIN master and a LIN slave using a PSoC device in a windscreen wiper motor control configuration. The LIN master serves as an interface for the user to key in the different modes and control motor operations and direction, similar to a wiper control arm on the steering column.

Implementation uses three different motor control modes:

- **Continuous Mode**: In this mode, the motor runs continuously in one direction as set by the user.
- **Wiper Mode**: In this mode, the direction of the motor is periodically reversed. This reversal can be implemented using limit switches. These switches are provided on the board for demonstration purposes.
- **Wash Mode**: This mode is similar to wiper mode except that it executes a fixed number of sweeps.

Additional keys provide controls such as starting/stopping the motor and setting the direction of movement. The user can also set the method of driving the BLDC motor using either Back EMF (Electromotive Force) for sensor-less function or Hall-Effect Sensors.

The LIN slave and BLDC motor controller combine the functionalities of receiving commands from the LIN master and providing the BLDC motor control per user settings. The LIN slave implementation provides a good example of the dynamic reconfiguration capabilities within PSoC Designer. With dynamic reconfiguration, a single PSoC block resource may be utilized for multiple purposes during run-time. The LIN slave implementation also provides sensing of the baud rate used by the LIN master as a part of the requirements in LIN Bus 2.0 Specifications.

BLDC in Automotive

BLDC motors are becoming widely used in the field of control motors. These kinds of synchronous motors are used as servo drives in numerous applications.

One of the advantages of BLDC motors is the absence of sparks. These motors are also immune to noise, brush-life, and dust problems associated with brush motors. BLDC motors are potentially cleaner, faster, more efficient, less noisy, and more reliable. However, BLDC motors require a more complex electronic control.

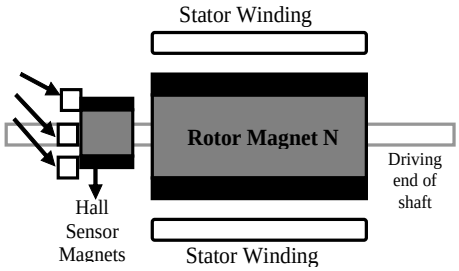

Figure 1. BLDC Motor Transverse Section

A BLDC motor consists of a stator and a rotor. The **stator** consists of stacked steel laminations with windings placed in the slots that are axially cut towards inner periphery. Most BLDC motors have three stator windings connected in a star configuration. Each winding consists of numerous interconnected coils. These windings are distributed over the stator to form an even numbers of poles.

The **rotor** is generally a ferrite permanent magnet and can vary from two to eight pole pairs with alternate north and south poles. The material for the rotor is chosen according to the required magnetic field density.

In BLDC motors, the commutation is controlled electronically. The motor requires that the stator windings be energized in a particular sequence. In order to obey this sequence, it is important to know the rotor position. This can be done either by using a sensor such as a Hall-Effect Sensor or by sensing Back EMF. Hall-Effect Sensors are embedded in the stator. When the rotor magnetic poles pass near the hall sensors, they give a high or low signal, indicating that the north or south pole is passing near. The position of the rotor can be derived with the exact combination of the three hall sensor signals.

BLDC Motor Control Basics

In Back EMF detection, the Back EMF is detected during a windings' non-energized phase. To get maximum efficiency of the motor, commutation should take place when the current in a winding is in phase with the Back EMF in the same winding.

The actual speed of the motor can be measured and the error between set and actual can be used to adjust the PWM (pulse width modulator) duty cycle, thereby dynamically implementing the PI (proportional and integral) control loop.

BLDC motor applications, such as wiper control, require starts, stops, and reversals of rotation under load. The demands for torque under reversal and standstill conditions must be satisfied. Such a requirement is met by keeping the initial duty cycle on the higher side of a pre-determined amount of time. Once the motor attains the speed, the duty cycle will be governed by the PI control loop.

LIN Basics

LIN (Local Interconnect Network) bus is a single wire, low-cost bus used for communication in vehicle electronics. LIN implements a single master and multiple slave configurations, providing communication speed up to 20 kbits/second. The LIN bus is most commonly used in automotive applications using smart sensors, actuators, or illumination. These nodes are often connected to the car network and become accessible to all types of diagnostics and services.

A particular feature of LIN is the synchronization mechanism that allows clock recovery by slave nodes without additional quartz or ceramic resonators.

The LIN communication takes place in frames. Figure 2 shows the structure of a frame.

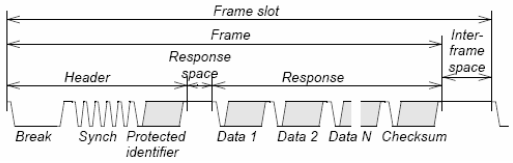

Figure 2. Structure of a LIN Frame

Each frame is made of a break field followed by 4- to 11-byte fields. Each byte field is transmitted as a serial byte as shown in Figure 3.

Figure 3. Structure of a Byte Field

Break Field

The break symbol is used to signal the beginning of a new frame. The break is the only field that does not comply with Figure 3. A break is always generated by the master task (in the master node), and is at least 13 bits of dominant value (low level on bus), including the start bit, followed by a break delimiter, as shown in Figure 4. The break delimiter is at least one nominal bit-time in duration.

A slave node shall use a break detection threshold of 11 nominal bit times.

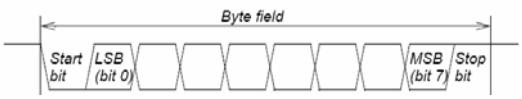

Figure 4. Break Field

Synch Byte

The synch byte helps the slave synchronize to the master's baud rate. The synch byte is nothing but a data field with 0x55 as its data. The synch byte is shown in Figure 5.

Figure 5. Synch Field

The slave measures the time between the start bit and the fourth falling edge of the synch byte. Dividing this by eight (right shift three times), gives the single bit time. Based on this time, the slave sets the clock of its UART (Universal Asynchronous Receiver Transmitter) to send and receive data bytes at the master's bit rate.

Protected Identifier

The byte immediately following the synch byte is the protected identifier. This byte has two parts. Bits 0-5 form the actual identifier. Bits 6 and 7 form the identifier parity. The identifiers can have values between zero and 63 and can be split into four different categories:

- Identifiers 0 - 59 (0x00 - 0x3B) are used for signal carrying frames.
- Identifiers 60 (0x3C) and 61 (0x3D) are used for diagnostic frames.
- Identifier 62 (0x3E) is used for user-defined extensions.
- Identifier 63 (0x3F) is used for future protocol enhancements.

More details on LIN can be found in the LIN 2.0 Design IP and LIN2.0 specification at linsubbus.org. Additional details on LIN Bus applications using PSoC are available at www.cypress.com.

PSoC BLDC Motor Control

BLDC motor control involves the following functions:

- **Motor startup** requires different methods based on whether the Hall-Effect Sensor or Back EMF method is used.
- **Commutation** keeps the motor running at a predetermined speed.
- **Speed control** uses a PWM technique.
- **Current protection** senses the current and stops the motor if it draws more than specified current.

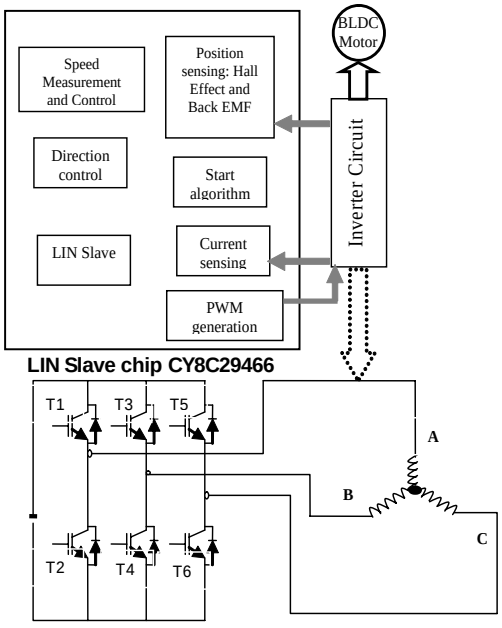

Figure 6. Main Blocks of BLDC Motor Control

The following paragraphs explain these functions in detail.

First the start algorithm is implemented, whereby the rotor is first aligned and then ramped up using the ramp-up parameters. The motor then switches to auto-commutated mode.

The motor speed is directly proportional to the applied voltage. The voltage applied to the motor windings is controlled using a PWM circuit. If the PWM duty cycle is varied linearly, the motor speed will also vary linearly.

The torque provided is proportional to the current in the windings. Every motor has a certain torque-to-speed characteristic. By varying the current in the windings, the torque varies, which will, in turn, vary the speed.

Speed variation is achieved by varying the voltage applied to the windings or the current through the windings. To vary the current, a PWM circuit is employed. The input to the PWM circuit (duty cycle or reference current) is derived from the PI controller. The PWM circuit controls the inverter circuit in the integrated power module.

The integrated power module drives the motor. It contains a three-phase bridge using IGBTs to control the voltage and current in the motor windings. The input to the integrated power module is derived from the motor controller. The motor controller decides the sequence for turning ON/OFF the switches in the power module. Only the high side switches in the power module are supplied with the PWM signal. The low side switches are kept ON for a particular commutation step. Back EMF sensing is done during the OFF time of the PWM signal. When the high side switch is OFF (PWM OFF time), the path for the current in the windings completes through the diode across the IGBT. This results in almost zero potential at the neutral point of the windings. With this, the motor controller can read the Back EMF referenced to ground.

Commutation

The motor controller can be configured in:

- Hall-Effect Sensor (sensor mode).
- Back EMF (sensor-less mode).

The motor controller is capable of sensing the rotor position either by the Back EMF method or the Hall-Effect Sensor method. In the Back EMF method, the Back EMF in the non-energized winding is monitored. The speed of the motor is calculated from the step time (time between two zero crossings of the Back EMF signal in one motor winding). This calculated speed is used in the feedback loop in closed-loop mode.

For optimum usage of the PSoC resources, only one comparator is used along with a multiplexer for sensing the Back EMF signals. The rotor position is derived by the motor controller software and depends on Back EMF signals. This rotor position is then used to decide the commutation sequence.

In the Hall-Effect Sensor method, the position of the rotor is provided by the Hall-Effect Sensors embedded in the motor. The commutation sequence is determined from the rotor position given by the sensors.

Zero-Crossing Detection for Back EMF Sensing

The Back EMF sensing technique is based on the fact that only two phases of a BLDC motor are connected at any one time. The third phase can be used to sense the Back EMF voltage.

The Back EMF of the three motor phases is measured using a resistive divider. The DC bus voltage is also measured using an identical divider circuit. Three comparators are used to provide a zero-crossing signal for each phase.

Figure 7 shows the subsystem to detect the zero crossing of the Back EMF.

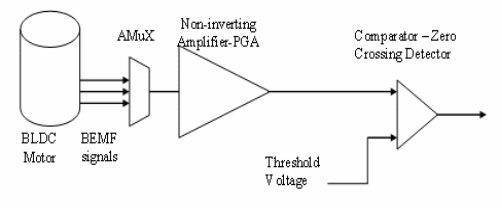

Figure 7. Zero-Crossing Detection (Sensor-Less Mode)

The Back EMF signals route into the PSoC device through an analog mux (multiplexer) to a non-inverting amplifier, which is implemented using a PGA. All of these components internally exist in the PSoC device. There is a diode on the Back EMF sense inputs. This diode clips the input at 0.7 volts, not letting the amplifier go into saturation. The maximum value of the analog output is Vdd - 0.5.

The gain for this PGA should be:

$$Gain = \frac{Vdd - 0.5}{0.7} \quad \textbf{(1)}$$

$$Gain = \frac{5.0 - 0.5}{0.7} = \frac{4.5}{0.7} = 6.428 \quad \textbf{(2)}$$

We chose a gain of 5.333 to avoid the saturation of the opamp. Because the measured voltage is very small, the PGA's reference is chosen to be Vss.

The comparator is utilized to compare the output of the PGA with a fixed reference value. The reference must be sufficiently elevated in order to avoid false triggering, due to noise. The lower limit is chosen to be Vss and the reference value is 0.75* (Vdd - Vss), which corresponds to an input voltage level of:

$$\frac{0.75\times(Vdd-Vss)}{AMP\_Gain}=\frac{0.75\times5}{5.33}=0.7035\ \text{Volt} \quad \textbf{(3)}$$

The output of this comparator is routed to port pin P0[3]. This pin is used as an input by the PSoC digital section. The zero-cross detection (ZCD) edges should be detected during PWM OFF time. To accomplish this, an extra counter is used. The enable signal for the counter is an inverted PWM signal. The count is set to 1 and the ZCD event is applied to the clock. This provides an interrupt on every ZCD event, when the PWM output is OFF.

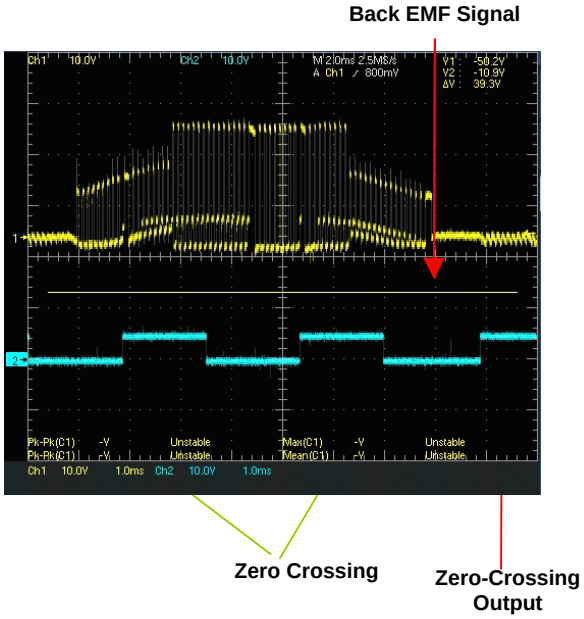

Figure 8. Back EMF Zero-Crossing Detection, Zero-Crossing Signals

Hall-Effect Sensor

There are three Hall-Effect Sensors; one for each phase. They provide the output signals as illustrated in Figure 9.

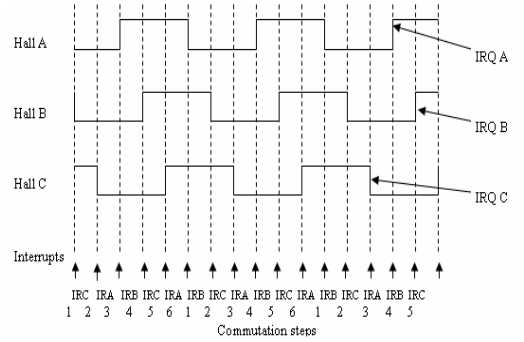

Figure 9. Output of Hall-Effect Sensors

The output of the Hall-Effect Sensors is provided to the GPIO port pin 0 with interrupts enabled in such a manner that generates an interrupt at the rising and falling edge of the Hall-Effect signals.

The Hall A output is routed to the capture of the speed_sensor timer, which measures speed. Commutation signals are generated at each GPIO interrupt.

Start Algorithm

When the motor is given an initial start command, a start algorithm is applied, whereby the motor gets aligned and is ramped up and subsequently switched into the auto-commutated mode. The start algorithm is required mainly for Back EMF feedback and consists of the following phases:

Alignment Phase

Since the initial position of the rotor is not known when using the sensor-less method, an alignment phase is applied. The motor is controlled using a triple half bridge as shown in Figure 10.

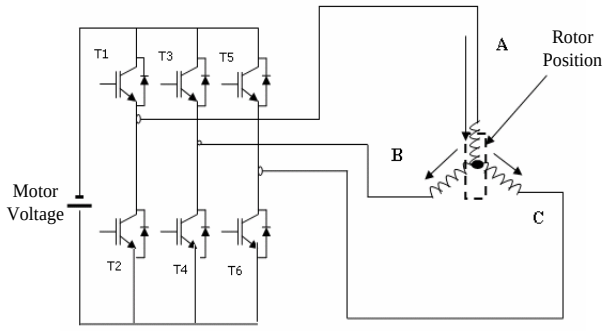

Figure 10. Alignment Phase Triple Half Bridge

Transistors T1, T4 and T6 are turned on at the same time. When the current is applied in this configuration, it will be positive in phase A and negative in phases B and C at half of the phase A value. Thus, the rotor becomes aligned as shown in Figure 10. Depending on the individual rotor inertia, the rotor requires a certain level of current to move. If we apply a strong current level directly to the rotor, the rotor might oscillate around its final position. To avoid this, a progressive current level is applied. This is done using an alignment current ramp. The progressive current ramp consists of a certain number of steps, with each step being the same length. The current level is increased with each step. This is done by increasing the PWM duty cycle. To apply a constant step length, the SleepTimer is used.

Ramp Phase

After the alignment phase, a ramp table is applied to the motor in order to accelerate it and detect the information needed to switch to auto-commutated synchronous mode.

In order to accelerate the motor, the ramp table is made of consecutively decreasing step times with a constant current level for all steps. The different parameters in the start sequence of the motor are software programmable.

In the starting phase of the motor, the Back EMF signal is not strong enough to be detected. The number of steps in the ramp table, without detection of a Back EMF zero-crossing event, is software programmable. This mode, in which the detection of Back EMF/Hall-Effect Sensor signals is disabled, is called the forced synchronous mode. After the forced synchronous mode, Back EMF/ Hall-Effect Sensor-signal detection is enabled.

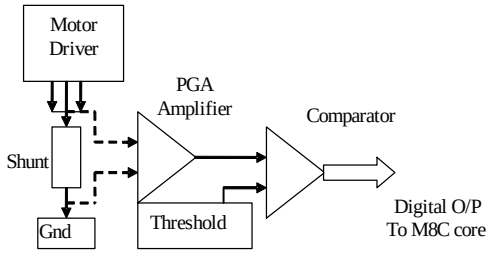

Figure 11. Current Measurement

The current through all three inverter half bridges is combined by connecting the emitters of all three transistors to a common shunt or sense resistor.

The voltage across the shunt is amplified using a PGA. This amplified signal is then fed to a comparator, the threshold of which is set per the particular requirement.

The over-temperature signal in the driver module is provided using the NTC thermistor built in the module. When the temperature increases, the voltage at the shutdown pin increases and finally causes the module to shut down.

PSoC Automotive BLDC

Figure 12 shows the block diagram of the PSoC automotive BLDC design.

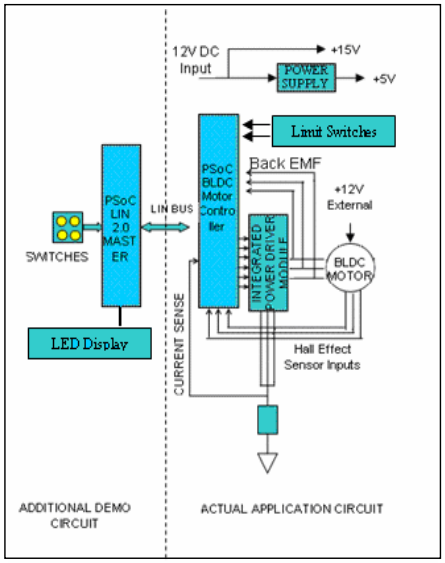

Figure 12. PSoC Automotive BLDC Block Diagram

The major elements of the design are:

PSoC CY8C27443 acts as the LIN master. The LIN master detects any key press and sends the motor setting commands to the LIN slave accordingly. It also requests at equal intervals the current motor status from the LIN slave. The LIN master updates the motor mode and setting on the LED. The details of the PSoC firmware are given in section "PSoC Automotive BLDC Firmware."

PSoC CY8C29466 acts as LIN slave and motor control. The LIN slave receives the motor setting commands from the master and drives the BLDC motor accordingly. It also sends the motor current status to the master.

The **LED** indicates the current mode and setting of the BLDC motor. Whenever there is a key-press event, the LIN master updates the LED display.

The four **keys** on the master side include Start/Stop, Forward/Reverse, Setting, and Mode. Key operations are as follows:

- The **Start/Stop** key is used to start and stop the motor. If the motor is in run state, pressing this key changes it to stop state and vice versa.
- The **Mode** key is used to choose from three modes of motor operation: continuous, wiper, and wash. Pressing this key selects the next mode of operation. The mode obeys this cyclic order: continuous, wiper then wash.
- The **Setting** key is used to select from four preset settings in a cyclic order.
- The **Forward/Reverse** key is used to toggle the clockwise and anti-clockwise direction for continuous mode.

There are two keys on the slave side, which act as limit switches to be used in the wiper position sense module.

BLDC Firmware

This Application Note features two separate PSoC devices, namely, LIN master and LIN slave. The LIN master senses the key press and transmits the appropriate commands to the slave via the LIN bus. It also drives the LED to indicate wiper mode and current setting. The LIN slave accepts the commands from the master and then drives the BLDC motor in the selected mode. The LIN master is implemented using the CY8C27443 and the LIN slave using the CY8C29466.

The PSoC device requires its hardware modules, clock signal and interconnections, to be configured for the application. This entire process is done using PSoC Designer software. This section describes the user module configuration and firmware design of the PSoC Automotive BLDC Motor Control application.

LIN Master: The LIN master uses dynamic reconfiguration. The three dynamic reconfigurations are: Synchro Break, Data Transmission, and Data Reception. The Synchro Break configuration generates the break field. The Data Transmission configuration sends the synchronization byte and any data bytes to be transmitted followed by the checksum byte. The Data Reception configuration receives the slave's response data.

The user modules in the Synchro Break configuration include the Baud Rate Counter, which generates the baud clock. The output frequency of this clock generator is eight times the baud rate. The Bit Time Counter is used to generate an interrupt every bit time. The Synchro Break Counter generates the synchro break field.

The Data Transmission configuration uses the Baud Rate Counter and Bit Time Counter to detect bit errors and the TX8 User Module to transmit data.

The Data Reception configuration uses the Baud Rate Counter and the Bit Time Counter for bit-time synchronization and the RX8 User Module to receive data.

LIN Slave: The LIN slave design includes the BLDC controller and has two configurations:

The Synchro Reception configuration detects the break/synch signal and calculates the master's bit rate. It has one 16-bit timer to generate the time between the rising and falling edges of the break/synch signal.

The Data Reception configuration receives and decodes the protected ID, and then sends data to or receives data from the master. The user modules in this configuration include an 8-bit Baud Rate Counter that generates the baud rate according to the bit rate calculated during the break/synch detection stage. An 8-bit Bit Time Counter generates interrupts at bit times either for detecting timeout while receiving data or for checking bit errors while transmitting data. One RX8 User Module receives data and processes it. Protected ID and checksum are checked and proper flags are set. One TX8 User Module transmits data to the master.

Motor Control Module: The motor control module in the LIN slave includes the following user modules. A 16-bit counter is used to generate the commutation delay when the motor is commutated using the Back EMF. Another 16-bit timer is used to measure the speed of the motor. The Hall-Effect Sensor or the ZCD signal is used as a capture interrupt for the Hall-Effect and the Back EMF, respectively, to measure the speed. A 16-bit PWM generator is used to control the BLDC motor driver module. Using this variable PWM generator, the voltage given to the motor is varied. A Digital Inverter and 8-bit counter are used in Back EMF mode to generate the capture event for speed counter only when the Back EMF zero crossing is detected during the PWM OFF time.

PSoC Designer: Digital Block

Figures 13, 14 and 15 provide a view of the digital user modules and their block placement and interconnection on the PSoC device.

PSoC Designer: Analog Block

Figure 16 gives a view of the analog user modules and their block placement and interconnection on the PSoC device.

The parameters for analog blocks, such as the threshold for a comparator, can be changed through software.

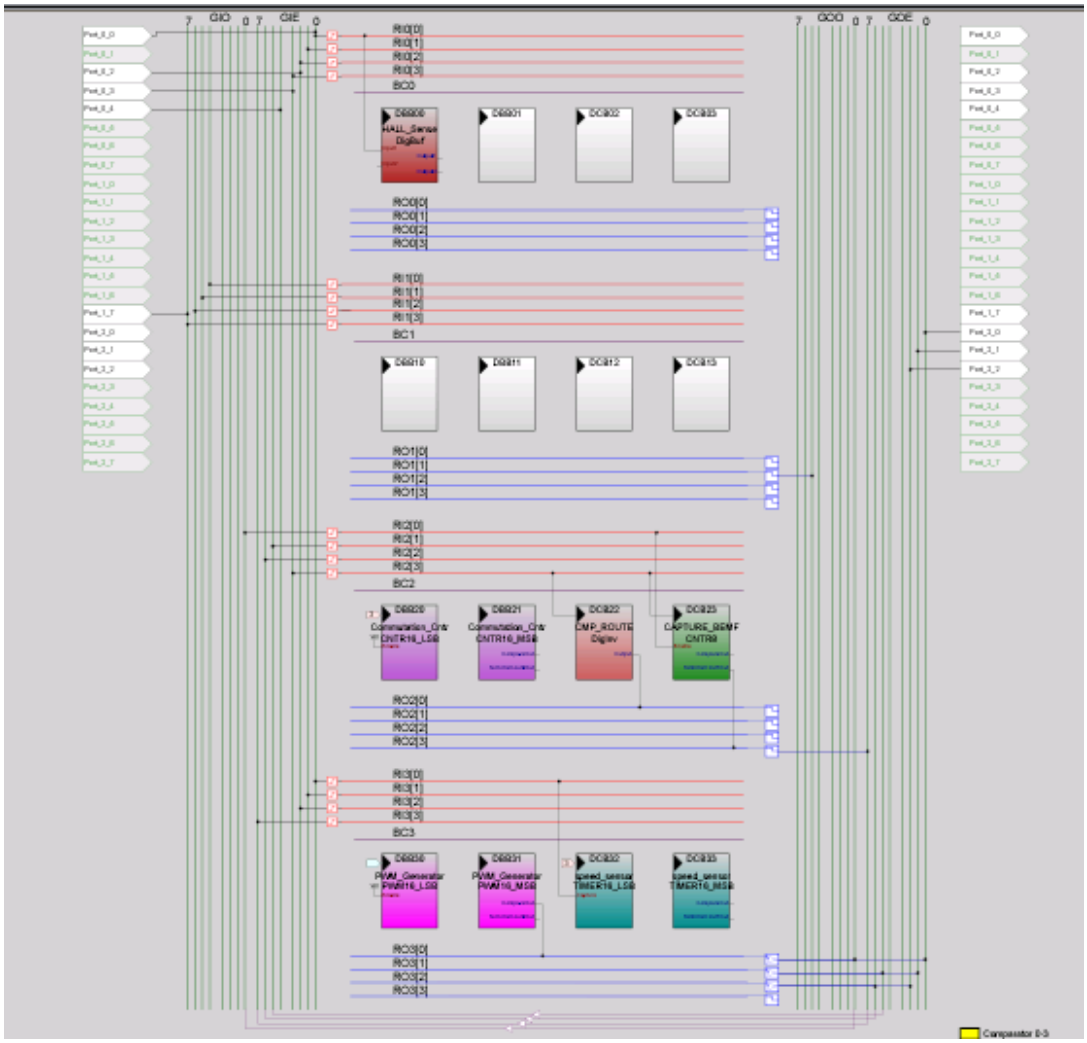

Figure 13. PSoC Designer Digital Blocks used for LIN Slave and Motor Controller

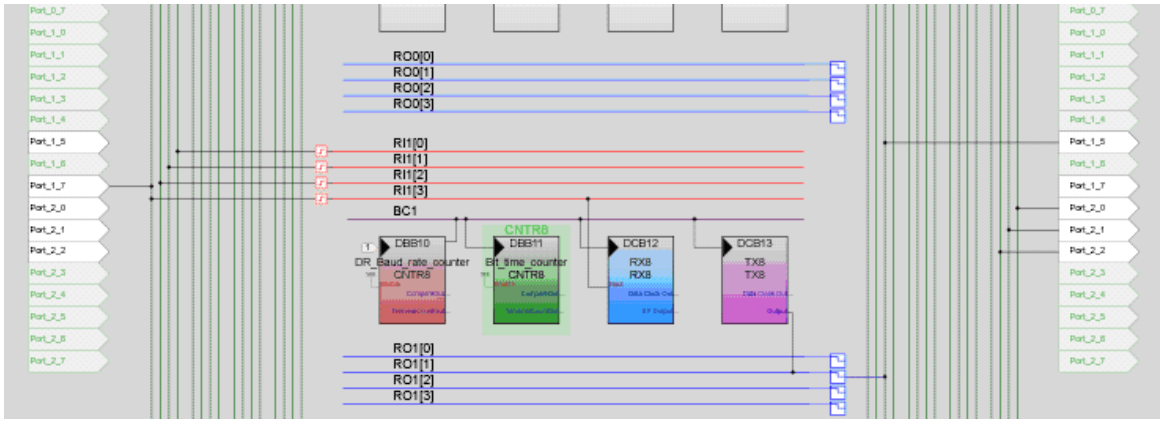

Figure 14. PSoC Designer Digital Blocks used for LIN Slave Data Reception Configuration

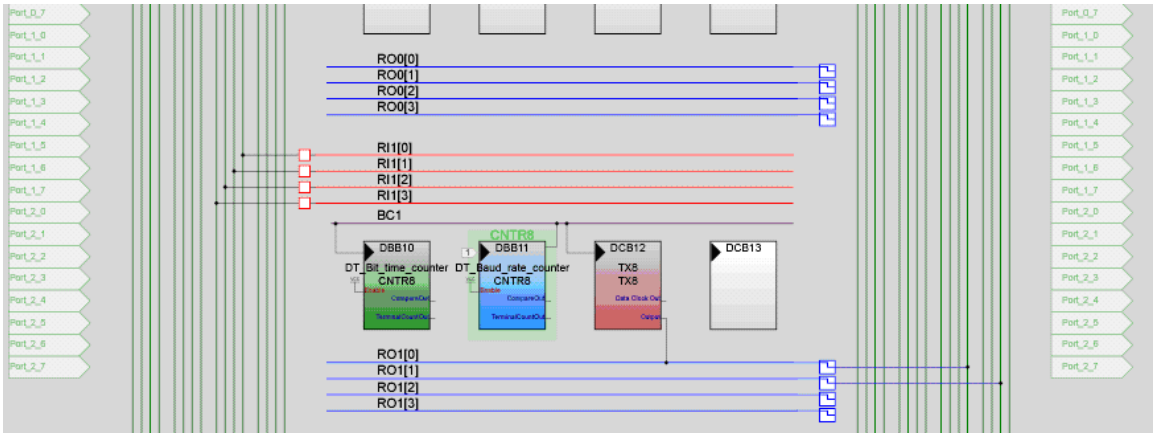

Figure 15. PSoC Designer Digital Blocks used for LIN Master Data Reception Configuration

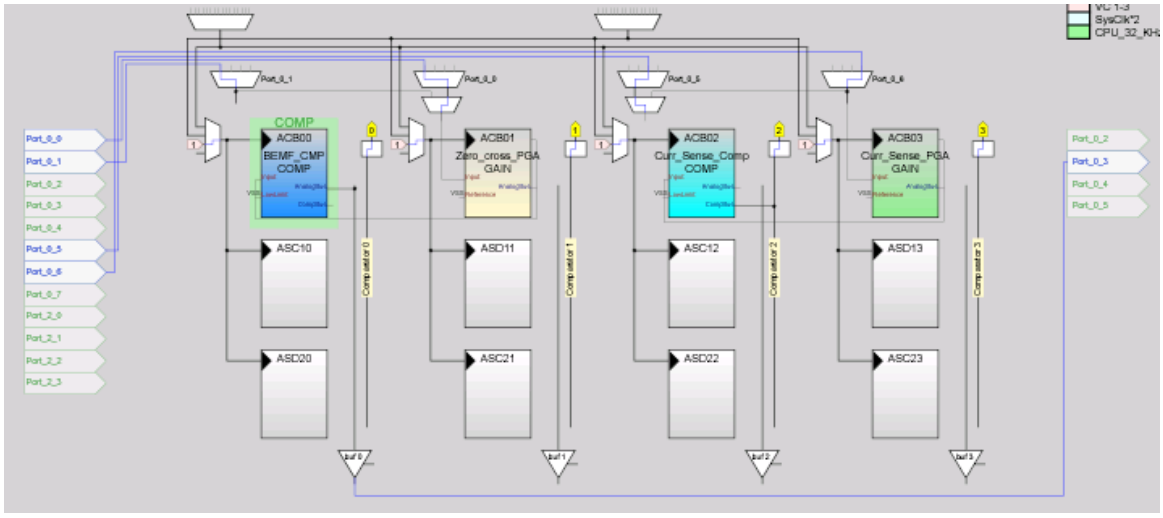

Figure 16. PSoC Designer Analog Blocks used for LIN Slave and BLDC Motor Controller

LIN Master

The subsystems in the LIN master are as follows:

- Boot Code
- Main Program
- Individual Modules
  - Key Detection Module
  - Frame Transfer Module

Boot Code

PSoC Designer provides a standard boot code for each project built. The boot code generated by PSoC Designer, *boot.asm*, is used without any modification.

Main Program

The main program contains the following:

- Initialization
- Main Continuous Loop
  - Node Configuration
  - Key Detection Module
  - Frame Transfer

Initialization

All needed user modules are initialized during initialization. The base configuration is loaded and the RAM variables initialized. Node configuration is also performed in this module.

Key Detection Module

The key detection module is used to detect key-press events. Whenever any key is pressed, frame 2 transfer is initialized, provided that the motor is in a stopped condition. When the motor is running, the key detection is disabled.

Frame Transfer Module

Two frames are used for communication between LIN master and LIN slave. Frame 1 is the slave-to-master frame and consists of the current motor status byte. This frame is sent at some fixed delay. Whenever there is any key-press event, frame 2 will be transferred from master to slave. This frame contains the information for the motor settings. The slave controller updates its local settings only when a start command is issued.

LIN Slave and Motor Controller

The subsystems in the LIN slave are as follows:

- Boot Code
- Main Program
- Initialization

Boot Code

PSoC Designer provides a standard boot code for each project built. The boot code generated by PSoC Designer, *boot.asm*, is used without any modification.

Main Program

After boot code execution, the CPU executes the main program. The main program contains the following:

- Initialization
- Main Continuous Loop
  - Motor State Machine
  - LIN Communication

The main functions are sourced as follows:

```
Initialization ()

While (1)
{
Motor state machine ();
LIN Communication ();
}
```

Initialization

This module initializes all the user modules needed for the BLDC motor and LIN slave. The initial base configuration for the LIN slave is loaded. This module also initializes program RAM variables to their reset values and reads the mode and respective settings from the Flash memory.

Motor State Machine

This module is called from the continuous loop `While (1)` in the main program. It updates the current state of the motor. Commutation and speed sensing for the motor is done in the interrupts.

LIN Communication

This module handles status data received from the LIN master and takes action accordingly. It also maintains the current motor status byte to be sent to the master. Data reception and other checks are done in the interrupt service routine (ISR). For details, refer to the LIN 2.0 Slave Node Design IP.

Motor Controller
The BLDC motor controller includes:

- Wiper control module
- Motor start and ramping algorithm
- Rotor position sensing and commutation handler
- Speed sensing
- Wiper position interrupt
- Limit switch module

Interrupt Service Routines (ISRs)

- Speed sensor ISR for BLDC motor controller
- Sleep ISR
- LIN slave ISR, which includes RX, TX, GPIO, bit timer and syncro ISR, and the timer ISR

The following ISRs are implemented:

- The **Speed Sensor ISR** is used for commutation. Based on this interrupt, the next commutation phase is applied. This interrupt is also used to measure motor speed.
- The **Commutation Counter ISR** is used when the motor is running in Back EMF mode. The timer is used to introduce the delay between zero-cross detection and the next commutation sequence.
- The **Sleep ISR** is used for various wiper modes during the alignment and ramp-up stages of the motor.

PSoC Resources
The application firmware uses the following resources from the PSoC device.

LIN Master:

- 111 Bytes of RAM - Global Variables
- 33% Flash - 5371 Bytes
- 5 of 8 Available Digital Blocks
- 0 Analog Blocks

LIN Slave and BLDC Motor Controller:

- 123 Bytes of RAM - 105 Bytes Global Variables
- 35% Flash - 11379 Bytes
- 13 of 16 Available Digital Blocks
- 4 of 12 Available Analog Blocks (4 Continuous and 0 Switched Capacitor)
- 64 Bytes Flash used as EEPROM

Display

The LED display is used to indicate the mode and setting selected by the user. There are a total of four LEDs, out of which two are used to indicate the mode and two to indicate the setting. LED interpretation definitions are given in tables 1 and 2.

Table 1. Mode LED Definitions

Mode	D5	D3
Mode	D5	D3
Disabled	OFF	OFF
Continuous	OFF	ON
Wiper	ON	OFF
WASH	ON	ON

Table 2. Setting LED Definitions

Setting	D4	D6
Setting 1	OFF	OFF
Setting 2	OFF	ON
Setting 3	ON	OFF
Setting 4	ON	ON

Refer to Appendix D for details on settings and modes.

Applications

The PSoC Automotive BLDC Motor Control design can be connected using one of several serial communication interfaces including I2C, CAN, LIN, SPI or UART with little modifications. I2C, SPI and UART are some of the interfaces that are in-built components of the PSoC device.

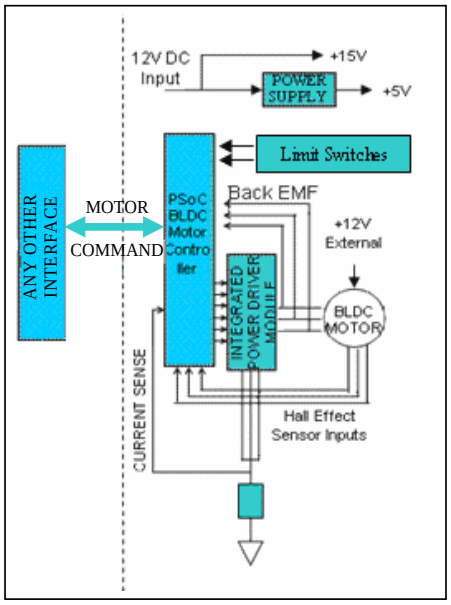

Figure 17. PSoC Automotive BLDC Interfacing to Other Systems

Demo Board User Guide

Power

The demonstration board is powered through a single +12V connection (Item N in Appendix C, Figure 21). 5V is provided to the logic through a regulator from the +12V supply. Additional means of supplying +9V to the +5V regulator logic is provided (Item M in Appendix C, Figure 21). However, this is considered optional depending upon the desired regulator circuit.

Programming

The demonstration board for the PSoC LIN master and slave devices is programmed via two programming headers. The master is programmed via the J2 connector. The slave is programmed via the J4 connector. These ISSP programming headers are to be used for field upgrades and subsequent development tasks.

Motor Connections

The Ametek motor has eight connections of interest. These connections are identified in Appendix I. The Hall-Effect Sensor connections are not required for operation in Back EMF mode.

Operating Instructions

Prior to applying power, it is important to configure the board for sensor or sensor-less operation. In the sensor operation, the Hall-Effect signals from the motor must be connected to J10. J7, J8 and J9 must be configured for Hall mode. After applying power, the setting and mode buttons must be configured for the desired operation. The various operations are identified in Appendix D. Reset buttons for both PSoC devices are provided to restore the device settings to their default conditions (Items G and H in Appendix C, Figure 21).

The start/stop switch (SW3, item A in Appendix C, Figure 21) provides the user with both a stop and start, based on the existing configuration. Once started, the mode and setting switches do not impact present operation.

During the motor stop condition, changes to motor direction, setting, and mode are available. During certain motor mode/setting options, switches SW6 and SW7 are active (Items I in Appendix C, Figure 21). These switches simulate the impact of a limit switch during wiper mode. Pressing these switches immediately initiates a motor reversal.

Summary

This Application Note provides an in-depth introduction to the PSoC Automotive BLDC wiper control design.

The purpose of this design, where the PSoC device is the main controller, is to assist designers in evaluating usage of the PSoC device for automotive applications. It can be used as an “off-the-shelf” solution and customized to specific conditions for BLDC motor control.

Appendix A: Firmware Flowchart

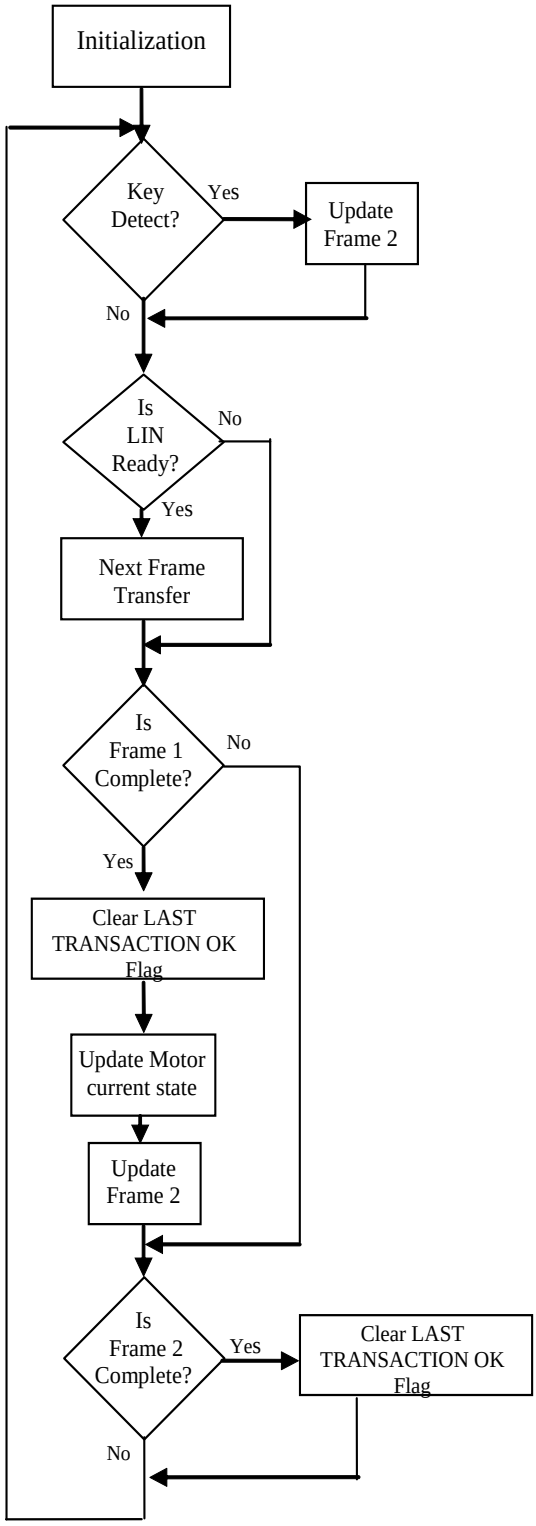

Figure 18. Sequence of LIN Master Main Program

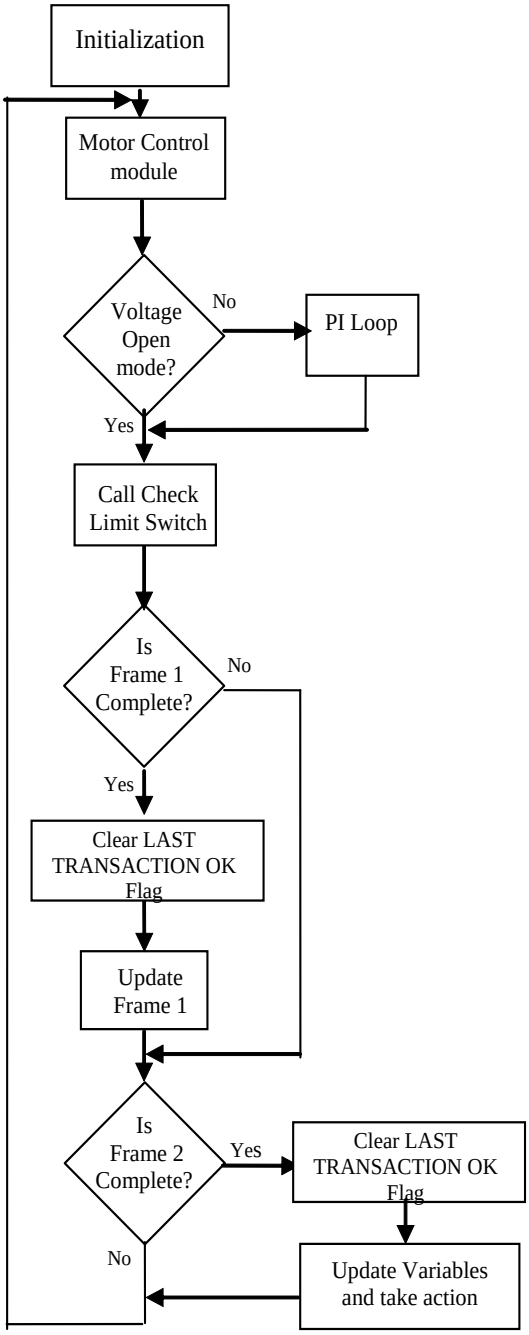

Figure 19. Sequence of LIN Slave Main Program

Appendix B: Motor Control State Machine

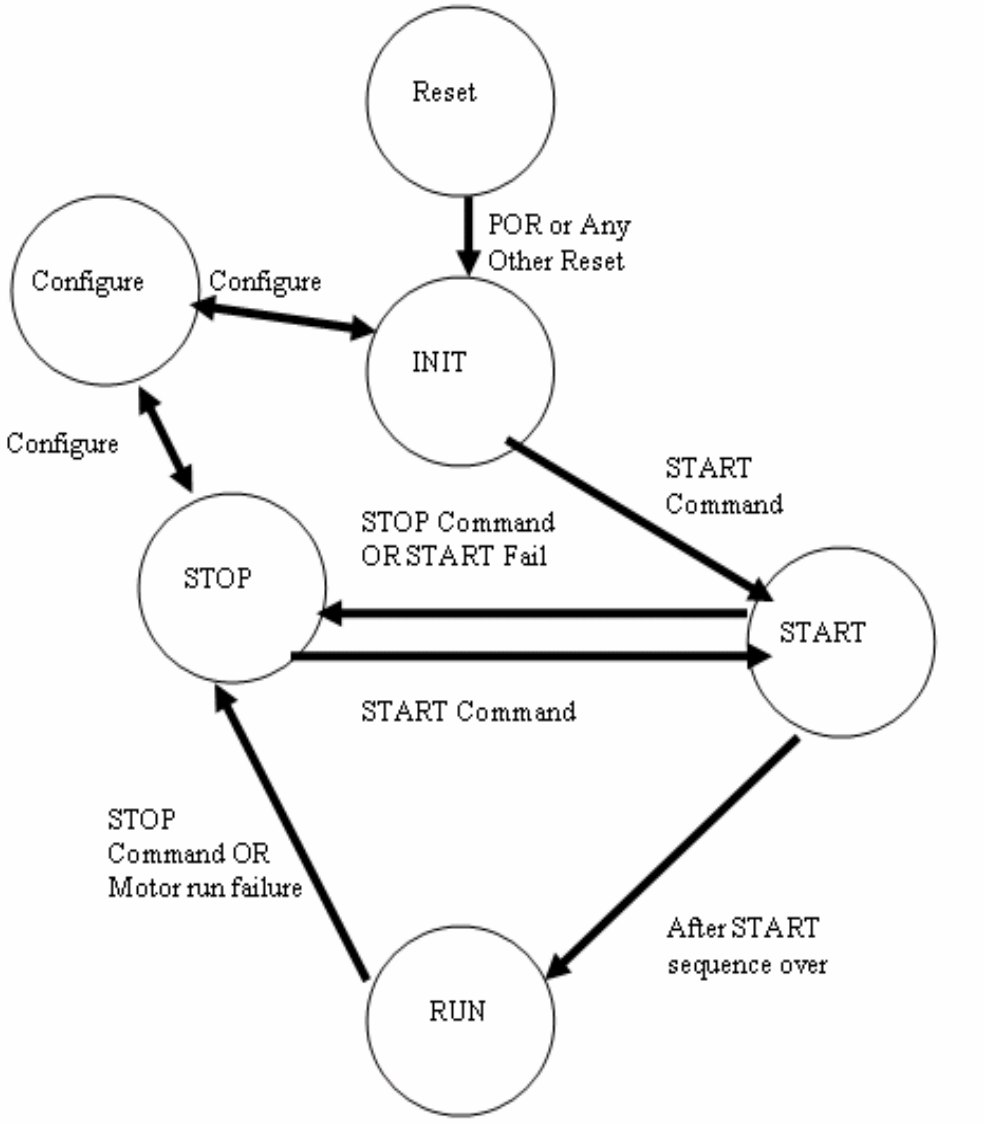

Figure 20. Motor Control State Machine

Appendix C: PSoC Automotive BLDC Motor Control Board

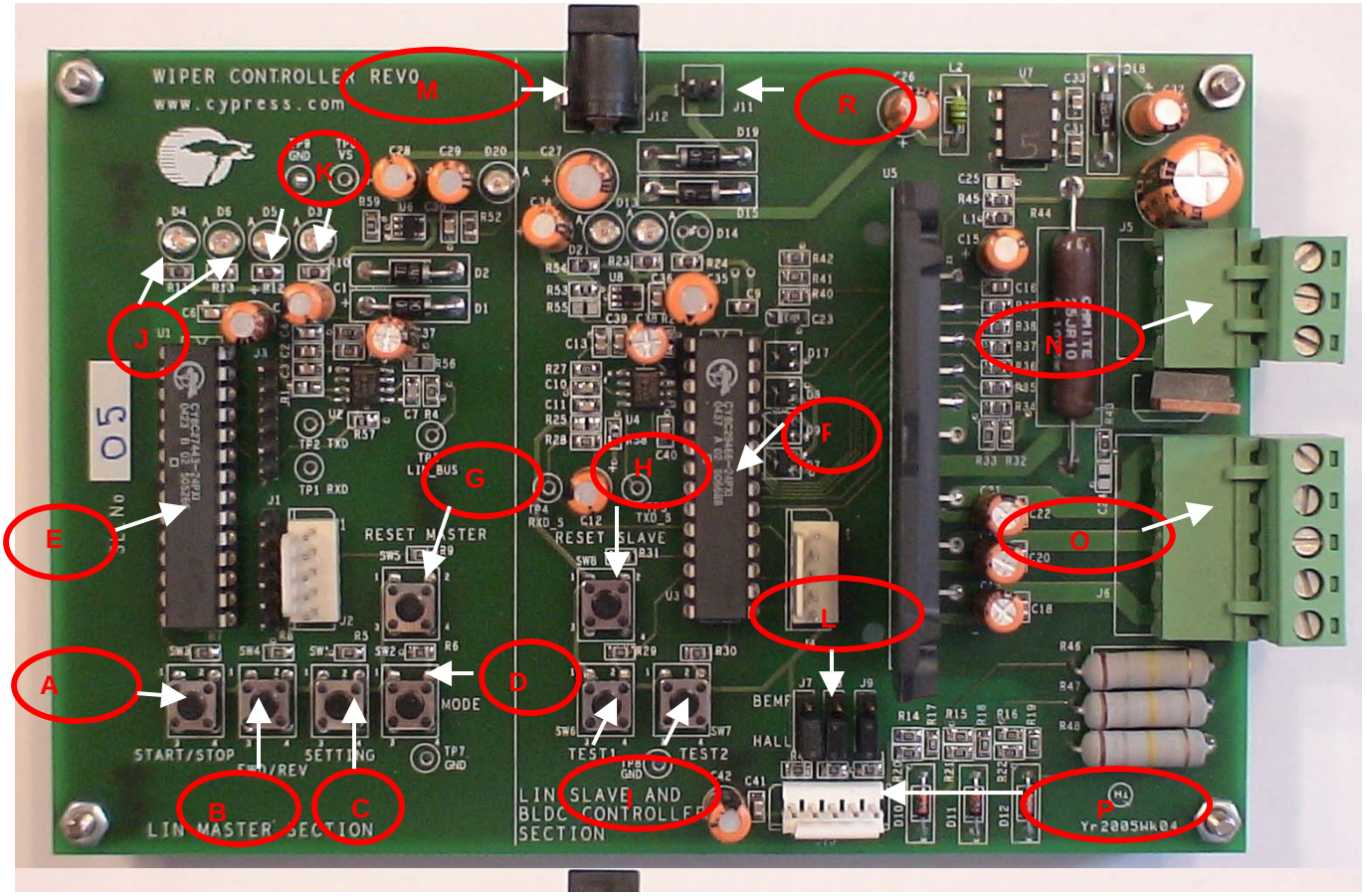

Figure 21. PSoC Automotive Motor Control BLDC Board

A. Start/Stop Key
B. Forward/Reverse Key
C. Setting Key
D. Mode Key
E. LIN Master
F. LIN Slave
G. Reset Master
H. Reset Slave
I. Limit Switches
J. Setting LED s (D4 and D6)
K. Mode LEDs (D5 and D3)
L. Jumpers for Commutation
M. DC Jack for +9V Input DC Supply (Optional: Remove short from Link J11 if +9V supply is used.)
N. Connector for Motor Supply (Do not exceed +12V DC.)
O. Connector for Motor
P. Connector for Hall-Effect Inputs from Motor
R. Link J11: Short this if +9V supply is not used. The logic supply is derived from +12V motor supply.

Appendix D: Settings and Modes and their LED Definitions

Table 3 Settings and Modes and their LED Definitions

Setting		Setting Description	Mode		Mode Description	J7, J8 and J9 Jumper Position	Remarks
D4	D6		D5	D3			
OFF	OFF	1500 RPM with Hall Mode	OFF	ON	Continuous Mode	Hall Position	
OFF	ON	1500 RPM with BEMF Mode	OFF	ON	Continuous Mode	BEMF Position	
ON	OFF	2000 RPM with Hall Mode	OFF	ON	Continuous Mode	Hall Position	
ON	ON	2000 RPM with BEMF Mode	OFF	ON	Continuous Mode	BEMF Position	
OFF	OFF	Rotation period - 4 seconds delay before reversal - 2 seconds	ON	OFF	Wiper Mode	Hall Position	Does not work with BEMF.
OFF	ON	Rotation period - 5 seconds delay before reversal - 2.5 Seconds	ON	OFF	Wiper Mode	Hall Position	Does not work with BEMF.
ON	OFF	Rotation period - 6 seconds delay before reversal - 3 Seconds	ON	OFF	Wiper Mode	Hall Position	Does not work with BEMF.
OFF	OFF	Rotation period - 4 seconds delay before reversal - 2 Seconds	ON	ON	Wash Mode 4 Wash Cycles	Hall Position	Does not work with BEMF.
OFF	ON	Rotation period - 5 seconds delay before reversal - 2.5 Seconds	ON	ON	Wash Mode 4 Wash Cycles	Hall Position	Does not work with BEMF.
ON	OFF	Rotation period - 6 seconds delay before reversal - 3 Seconds	ON	ON	Wash Mode 4 Wash Cycles	Hall Position	Does not work with BEMF.

Appendix E: Diagrams

Automotive BLDC Motor Control Block Diagram

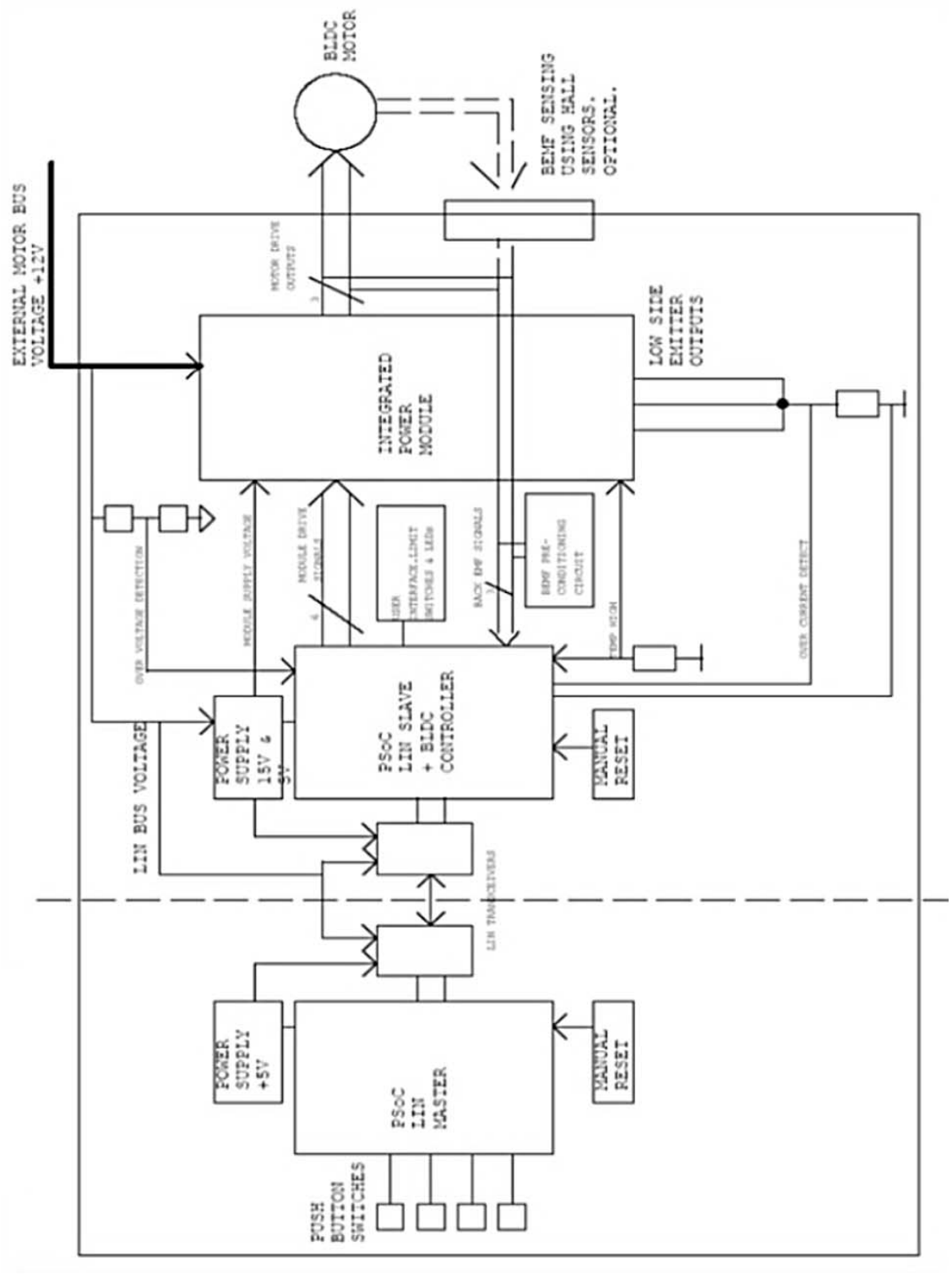

Figure 22. Block Diagram

PSoC Controller as LIN Master Schematic

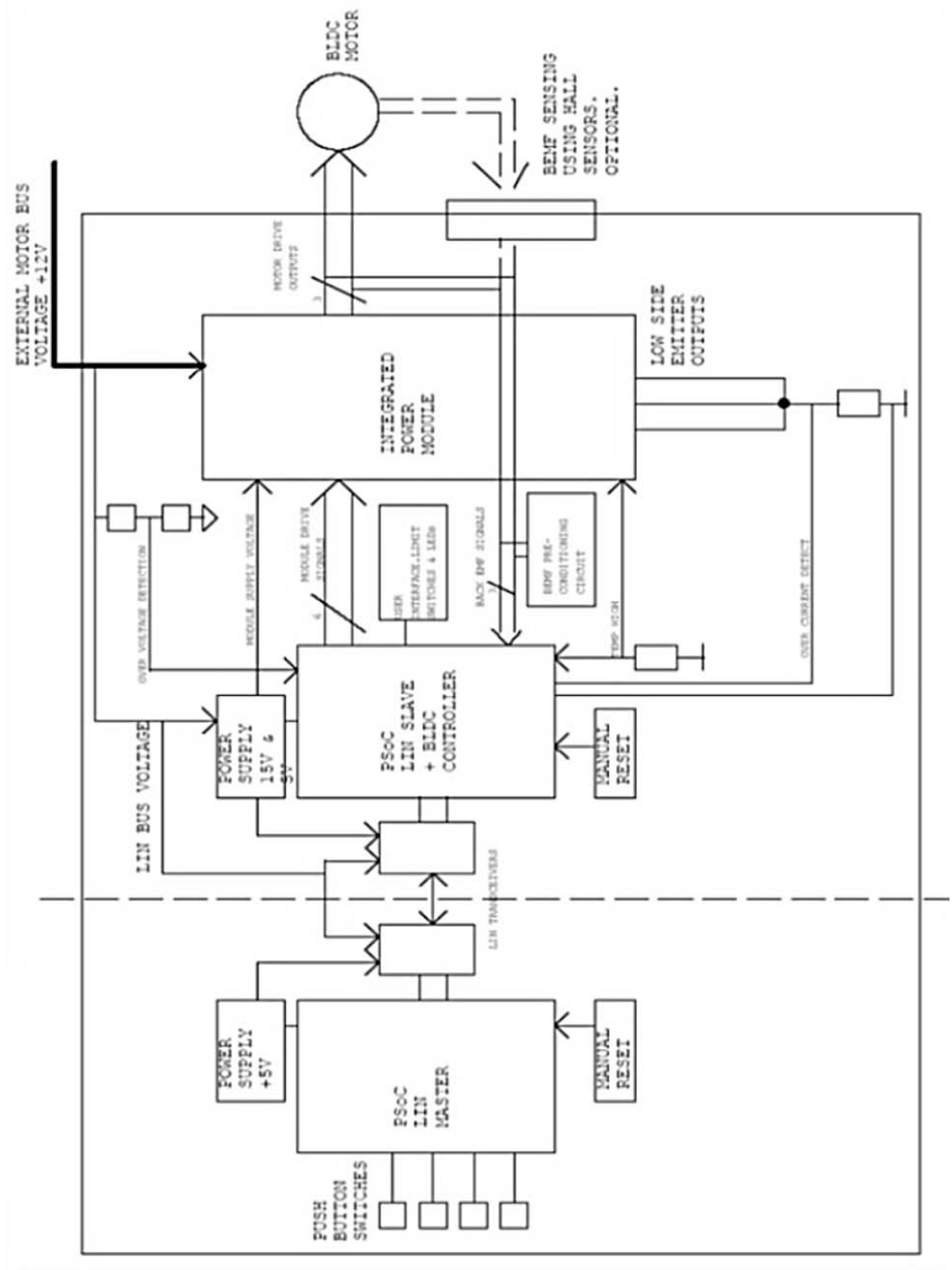

Figure 23. PSoC Controller as LIN Master Schematic

PSoC Controller as LIN Slave and BLDC Motor Controller Schematic

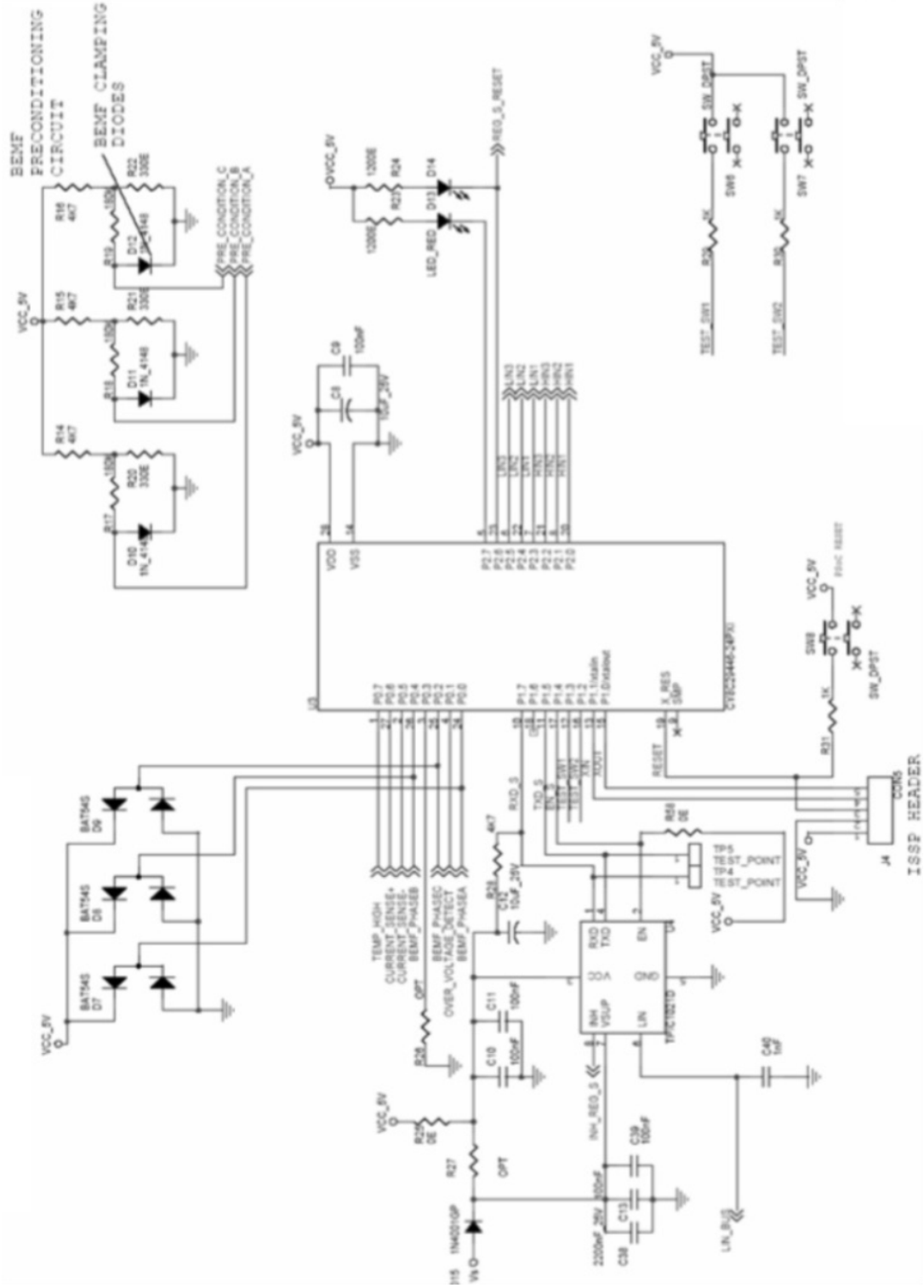

Figure 24. PSoC Controller as LIN Slave and BLDC Motor Controller Schematic

Integrated Power Module, and Hall and BEMF Feedback Schematic

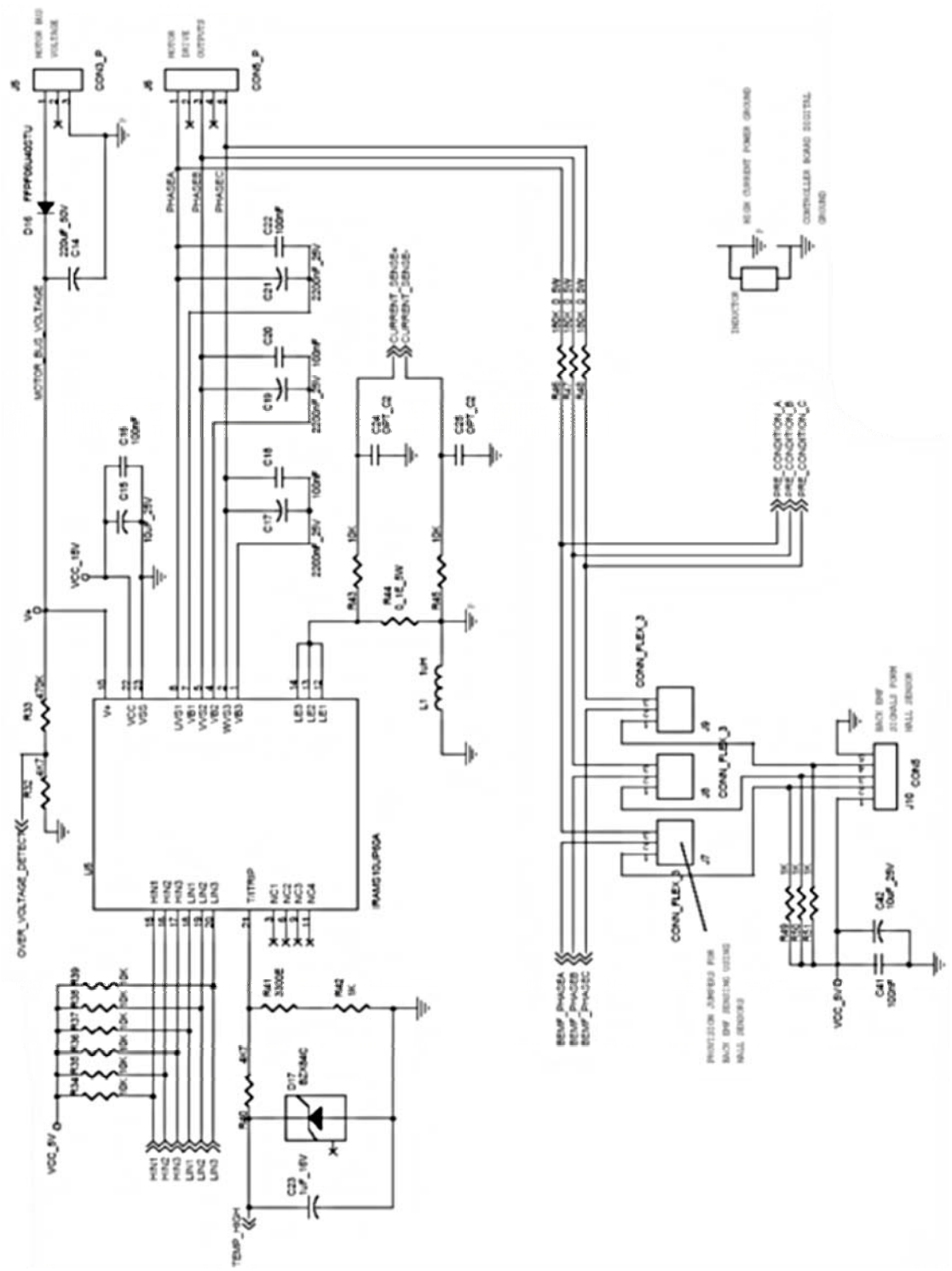

Figure 25. Integrated Power Module, and Hall and BEMF Feedback Schematic

Power Supply Schematic

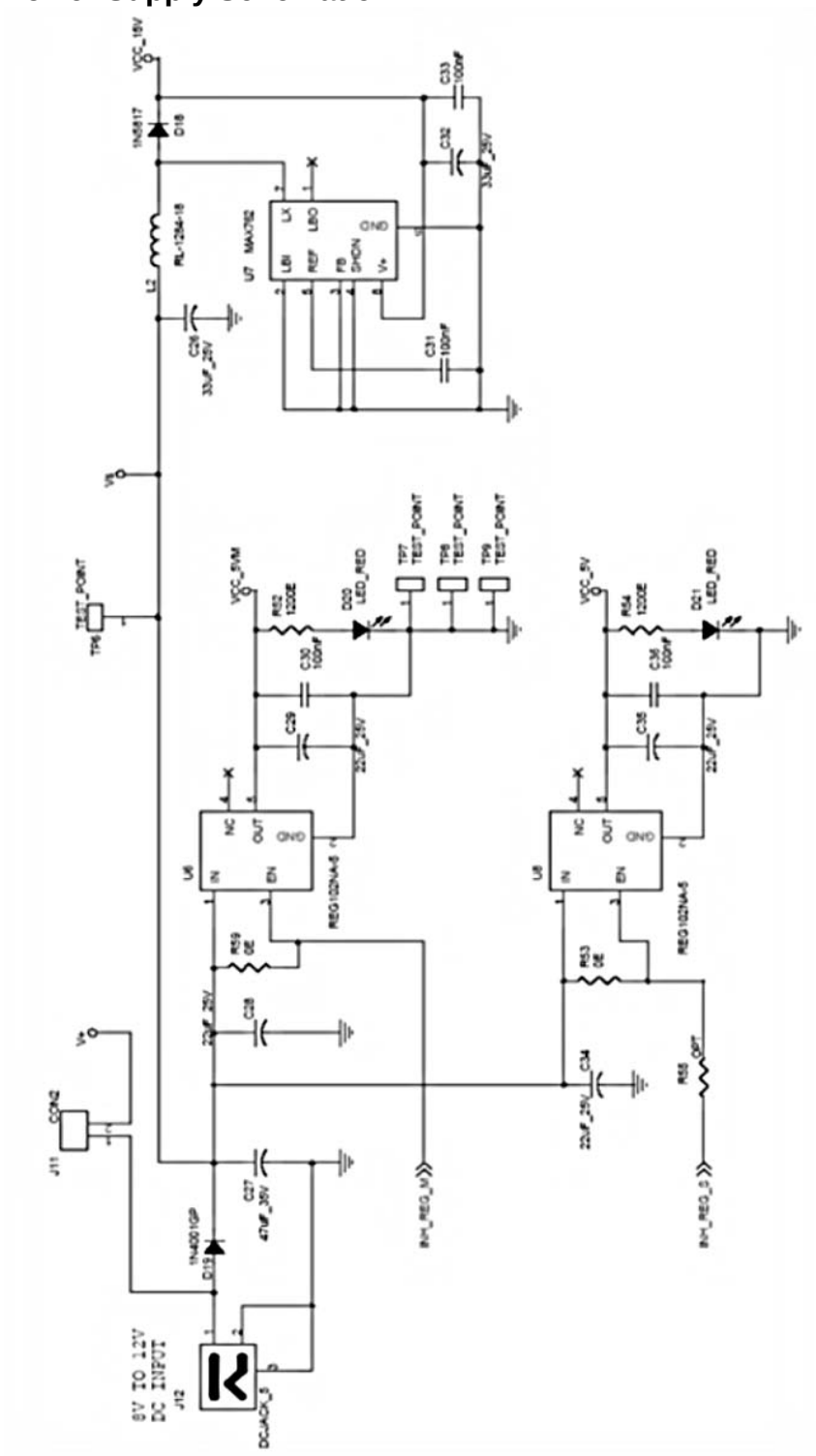

Figure 26. Power Supply Schematic

Advantages of using an integrated power module to drive the motor rather than a discreet device option: Integrated power module IRAMS10UP60A is used to drive the motor. The driving can also be achieved using six discreet switching devices to form a bridge as shown in Figure 10. But using the integrated power module has the following advantages:

- Integrated Gate Drivers and Bootstrap Diodes
- Temperature Monitor
- Temperature and Over-Current Shutdown
- Low VCE (on) Non-Punch through IGBT Technology
- Under Voltage Lockout for all Channels
- Matched Propagation Delay for all Channels
- Low-Side IGBT Emitter Pins for Current Control
- Schmitt-Triggered Input Logic
- Cross-Conduction Prevention Logic
- Lower dI/dT Gate Driver for Better Noise Immunity
- Motor Power Range 0.75~2kW/85~253 VAC
- Isolation 2000VRMS min

Appendix F: Bill of Materials

Item	Quantity	Reference	Part	Description	Manufacturer
1	6	C1, C5, C8, C12, C15, C42	10uF_25V	CAP ELECT 10uF 25V RADIAL	PANASONIC
2	18	C2, C3, C4, C6, C9, C10, C11	100nF	CAP 0.1uF 25V X7R 0805	YAGEO
		C13, C16, C18, C20, C22, C30			
		C31, C33, C36, C39, C41			
3	2	C40, C7	1nF	CAP CER 1000pf 50V X7R 10% 0805	YAGEO
4	1	C14	220uF_50V	CAP ELECT 220uF 50V RADIALT	PANASONIC
5	5	C17, C19, C21, C37, C38	2200nF_25V	CAP ELECT 2.2uF 25V RADIAL	PANASONIC
6	1	C23	1uF_16V	CAP TANTALUM 1.0uF 16V SMD	KEMET
7	2	C24, C25	OPT_C2	CAP 0.1uF 25V X7R 0805	YAGEO
8	2	C32, C26	33uF_25V	CAP ELECT 33uF 25V RADIAL	PANASONIC
9	1	C27	47uF_35V	CAP ELECT 47uF 35V RADIAL	PANASONIC
10	4	C28, C29, C34, C35	22uF_25V	CAP ELECT 22uF 25V RADIAL	PANASONIC
11	4	D1, D2, D15, D19	1N4001GP	GEN PURPOSE RECTIFIER DIODE	FAIRCHILD
12	2	D3, D4	LED	LED 3MM GREEN	LUMEX OPTO
13	1	D5	LED	LED 3MM YELLOW	LUMEX OPTO
14	1	D6	LED	LED 3MM RED	LUMEX OPTO
15	3	D7, D8, D9	BAT54S	DUAL SCHOTTKEY DIODE	FAIRCHILD
16	3	D10, D11, D12	1N_4148	SMALL SIGNAL DIODE	FAIRCHILD
17	3	D13, D20, D21	LED_RED	LED 3MM RED	LUMEX OPTO
18	1	D14	OPT_LED	LED 3MM YELLOW	LUMEX OPTO
19	1	D16	FFPF06U40STU	RECTIFIER DIODE 6A FWD CURRENT	FAIRCHILD
20	1	D17	BZX84C	ZENER DIODE 3.3V SOT23	FAIRCHILD
21	1	D18	1N5817	SCHOTTKEY BARRIER DIODE	FAIRCHILD
22	2	J3, J1	HEADER6	CON HEADER 6 POS VERTICAL	MOLEX
23	3	J2, J4, J10	CON5	CON HEADER VERT 5POS MTA-100	MOLEX
24	1	J5	CON3_P	TEMINAL BLK 3 POS 5.08MM STRAIGHT	MOLEX
25	1	J6	CON5_P	TEMINAL BLK 2+3 POS 5.08MM STRAIGHT	MOLEX
26	3	J7, J8, J9	CONN_FLEX_3	CON HEADER 3 POS VERT	MOLEX
27	1	J11	CON2	CON HEADER 2 POS VERTICAL	MOLEX
28	1	J12	DCJACK_5	PWR JACK 2.1x5mm	CUI INC.
29	1	L1	1uH	1uH SMD INDUCTOR	TDK corp.

Bill of Materials (continued...)

Item	Quantity	Reference	Part	Description	Manufacturer
30	1	L2	RL-1284-18	POWER LINE CHOKE	RENCO Elec.
31	6	R1, R25, R53, R57, R58, R59	0E	RES 0 OHM 1/8W, 5%	YAGEO
32	1	R2	OPT	RES 10K 1/8W 5% 0805	YAGEO
33	7	R3, R14, R15, R16, R28, R32	4K7	RES 4.7K 1/8W 5% 0805	YAGEO
		R40			
34	13	R4, R5, R6, R7, R8, R9, R29	1K	RES 1K 1/8W 5% 0805	YAGEO
		R30, R31, R42, R49, R50, R51			
35	8	R10, R11, R12, R13, R23, R24	1200E	RES 1.2K 1/8W 5% 0805	YAGEO
		R52, R54			
36	3	R17, R18, R19	180K	RES 180K 1/8W 1% 0805	YAGEO
37	3	R20, R21, R22	330E	RES 330E 1/8W 5% 0805	YAGEO
38	2	R26, R27	OPT	RES 0 OHM 1/8W, 5%	YAGEO
39	1	R33	470K	RES 470K 1/8W 5% 0805	YAGEO
40	8	R34, R35, R36, R37, R38, R39	10K	RES 10K 1/8W 5% 0805	YAGEO
		R43, R45			
41	1	R41	3300E	RES 3.3K 1/8W 5% 0805	YAGEO
42	1	R44	0_1E_5W	RES 0.1 OHM WIREWOUND 5W	OHMITE
43	3	R46, R47, R48	180K_0_5W	180K 5% METAL OXIDE change footprint	XICON
44	1	R55	OPT	RES 1.2K 1/8W 5% 0805	YAGEO
45	1	R56	10K		
46	8	SW1, SW2, SW3, SW4, SW5, SW6	SW_DPST	PB SWITCH DPST	ITT IND
		SW7, SW8			
47	9	TP1, TP2, TP3, TP4, TP5, TP6	TEST_POINT	MULTIPURPOSE TEST POINT	KEYSTONE
		TP7, TP8, TP9			
48	1	U1	CY8C27443	PSoC Device 27 SERIES	CYPRESS
49	2	U4, U2	TPIC1021D	LIN PHYSICAL INTERFACE	TI
50	1	U3	CY8C29446-24PXI	PSoC Device 29 SERIES	CYPRESS
51	1	U5	IRAMS10UP60A	MOTOR DRIVER POWER MODULE	IRF
52	2	U6, U8	REG102NA-5	250mA, LDO +5V	TI
53	1	U7	MAX762	12/15V STEP_UP DC TO DC CONVERTER	MAXIM

Appendix G: Board Layout Components (Not to Scale)

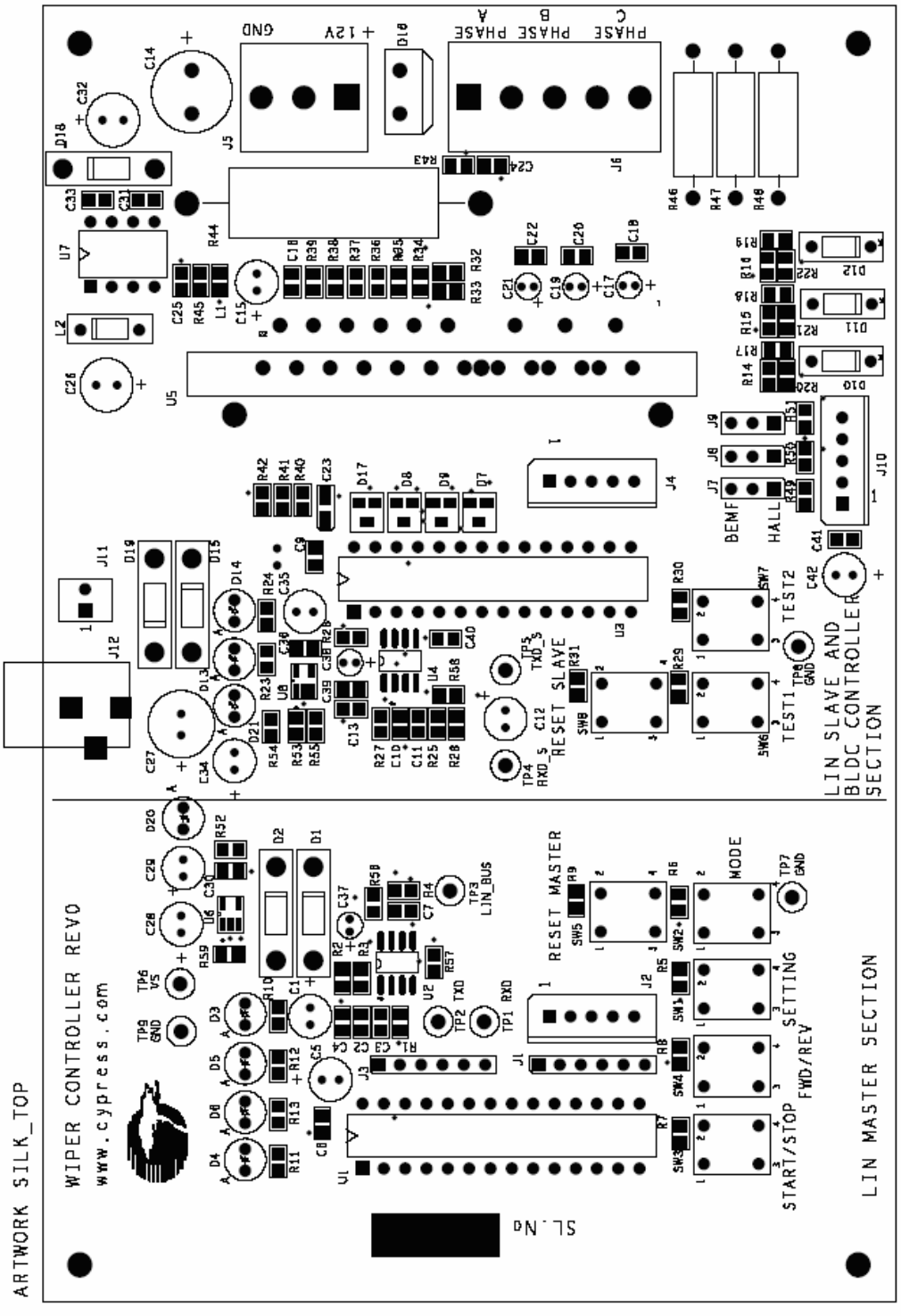

Figure 27. BLDC Motor Control Board Component Layout

Appendix H: Board Layout Components and Track (Not to Scale)

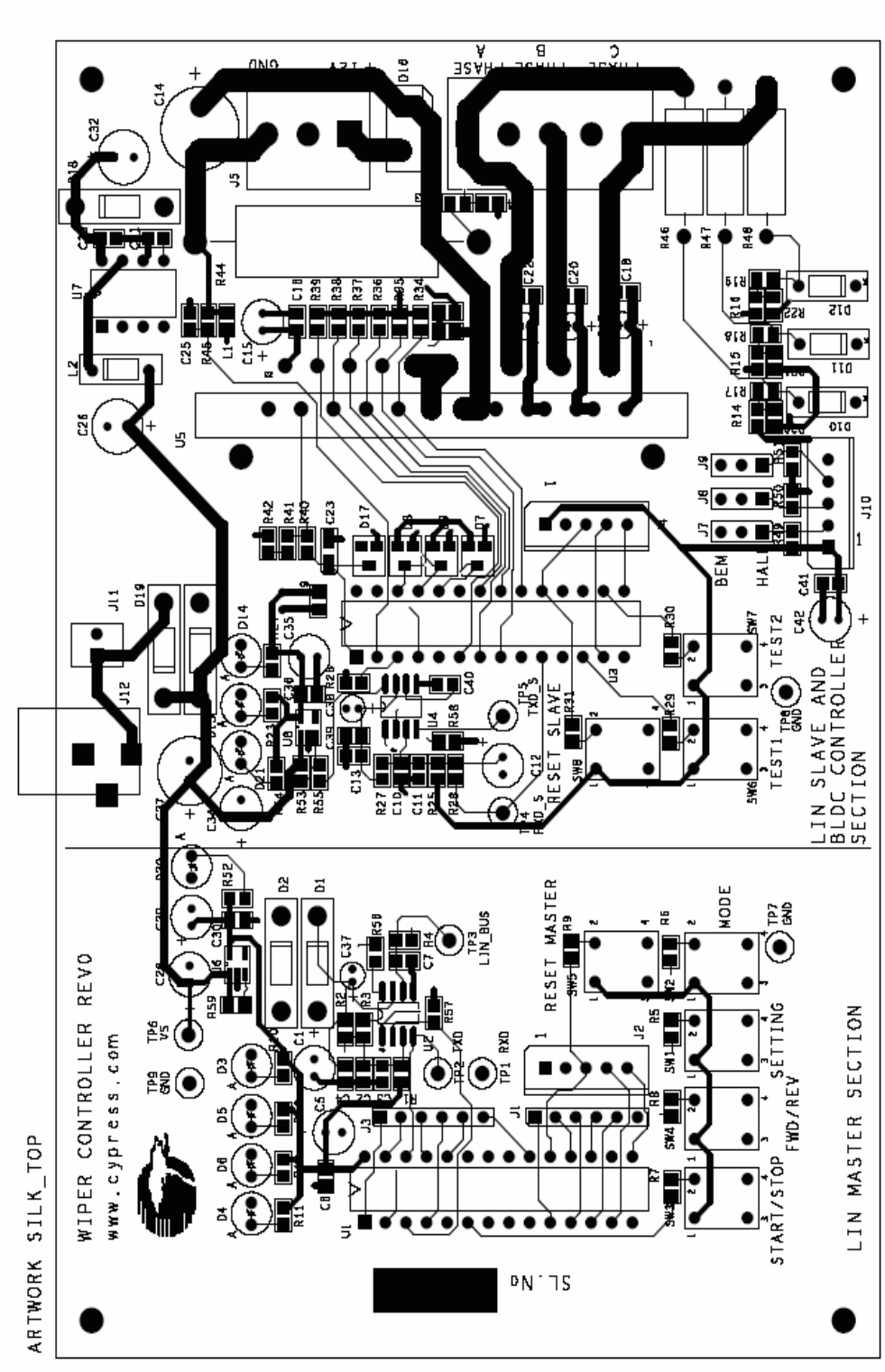

Figure 28. BLDC Motor Control Board Component and Track Layout

Appendix I: Wiper Controller Motor Wiring Guide

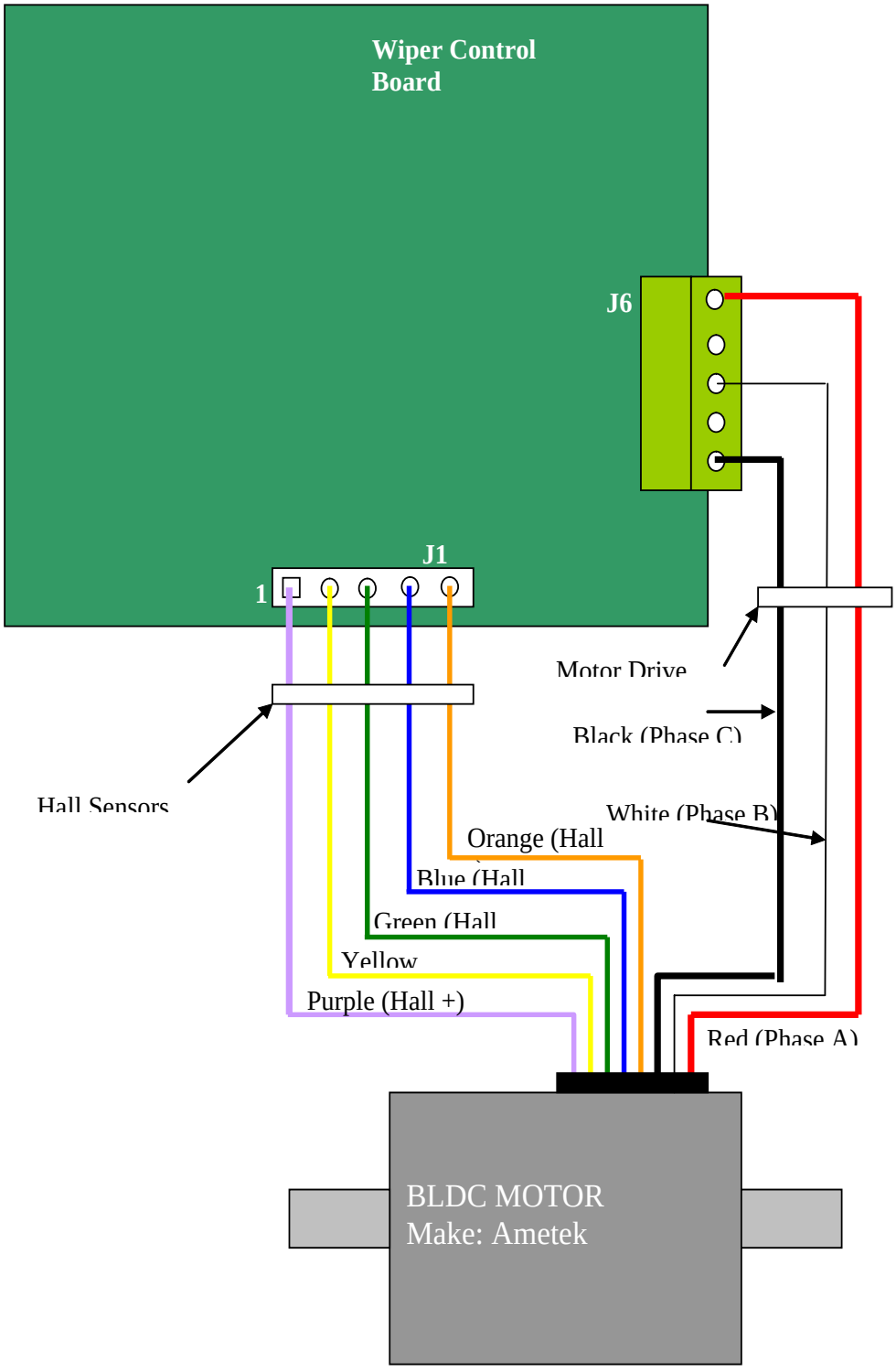

Figure 29. Ametek Motor Wiring Guide

Appendix J: Acronyms and Glossary

ADC: Analog-to-Digital Converter
BEMF: Back Electro Motive Force
BLDC: Brushless Direct Current
DAC: Digital-to-Analog Converter
EEPROM: Electrically Erasable Programmable Read Only Memory
EMF: Electromotive Force
GUI: Graphical User Interface
ISR: Interrupt Service Routine
LED: Light Emitting Diode
LIN: Local Interconnect Network
Mux: Multiplexer
PI: Proportional and Integral
PSoC: Programmable System-on-Chip
PWM: Pulse Width Modulator
UART: Universal Asynchronous Receiver Transmitter

Cypress Semiconductor
2700 162nd Street SW, Building D
Lynnwood, WA 98037
Phone: 800.669.0557
Fax: 425.787.4641
http://www.cypress.com/