Application Note

Electric Bike BLDC Hub Motor Control Using the Z8FMC16100 MCU

AN026004-0413

Abstract

This application note describes a controller for a 200 W, 24 V Brushless DC (BLDC) motor used to power an e-bike (i.e., electric bicycle). The design uses Zilog's Z8FMC16100 MCU and associated circuitry to implement motoring control, regenerative braking, and fault protection.

Note: The source code file associated with this application note, AN0260-SC01.zip, is available for download on zilog.com. This source code has been tested with version 5.0.0 of ZDS II for Z8 Encore! XP MCUs. Subsequent releases of ZDS II may require you to modify the code supplied with this application note.

Features

The main features of this high-torque motor control application include:

- Hall sensor commutation
- Motor speed measurement
- Potentiometer-adjustable motor speed
- Open-loop or closed-loop speed control for precise speed regulation
- Protection logic for over-voltage, over-current, and thermal protection.

Discussion

The Z8FMC16100 MCU features a flexible Pulse Width Modulation (PWM) module with three complementary pairs or six independent PWM outputs supporting dead-band operation and fault protection trip input. These features provide multiphase control capability for a variety of motor types and ensure safe operation of the motor by providing pulse-by-pulse or latched fast shutdown of the PWM pins during fault condition.

The Z8FMC16100 MCU features up to eight single-ended channels of 10-bit analog-to-digital conversion (ADC), with a sample and hold circuit. It also features one operational amplifier for current sampling and one comparator for over-current limiting or shutdown.

A high-speed ADC enables voltage and current sensing, while dual-edge interrupts and a 16-bit timer provide a Hall-effect sensor interface.

A full-duplex 9-bit UART provides serial, asynchronous communication and supports an option for the Local Interconnect Network (LIN) serial communications protocol. The

LIN bus is a cost-efficient single Master, multiple Slave organization that supports speed up to 20 kbps.

The Z8FMC16100 MCU has a rich set of peripherals and other features such as: additional 16-bit timer with capture/compare/PWM capability, SPI, and I^2C Master/ Slave for serial communication, and an internal precision oscillator.

The single-pin debugger and programming interface simplifies code development and allows easy in-circuit programming.

A block diagram of the Z8FMC16100 MCU is shown in Figure 1.

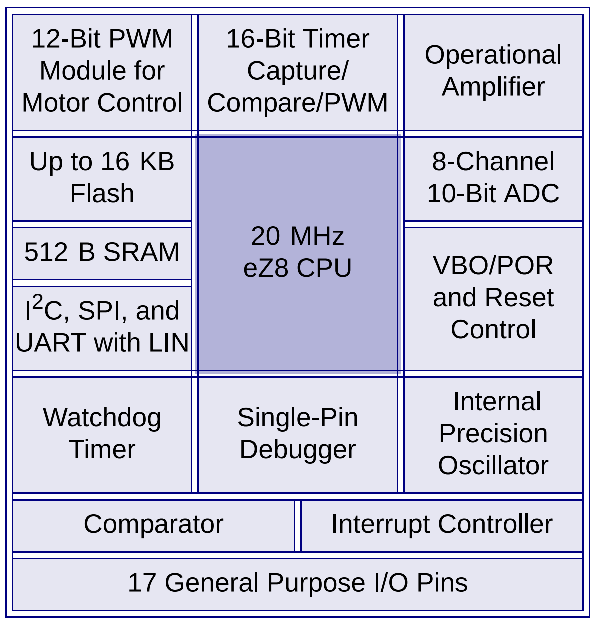

Figure 1. Z8FMC16100 MCU Block Diagram

Braking and Regenerative Charging

In this e-bike application, the Z8FMC16100 MCU's PWM registers are configured to operate in a complementary PWM mode when applied to an inverter bridge that consists of six MOSFETs. This complementary PWM mode allows for greater control of the e-bike's braking and regenerative charging process. The BLDC motor will effectively run in either a *motoring mode* or a *generative mode*.

Generative Mode is achieved when the the applied operating voltage is less than the BEMF voltage produced by the rotating motor. Motoring Mode is achieved when the applied voltage is equal to or greater than the BEMF voltage produced by the rotating motor. Varying the applied operating voltage can be accomplished by changing the duty cycle for each phase in the inverter bridge.

For quicker braking, the lower MOSFET devices of the inverter bridge are turned on while the upper MOSFETs are turned off, thereby quickly producing a negative current (i.e., a negative torque) to stop the motor. To gain a regenerative charge, the upper MOSFETs are turned on to allow current flow back to the battery, while the lower MOSFETs are turned off.

Hardware Architecture

In a Brushed DC motor, commutation is controlled by brush position. In a BLDC motor, however, commutation is controlled by the supporting circuitry. The rotor's position must therefore be fed back to the supporting circuitry to enable proper commutation.

Two different techniques can be used to determine rotor position:

Hall Sensor-Based Commutation. In the Hall sensor technique, three Hall sensors are placed inside the motor, spaced 120 degrees apart. Each Hall sensor provides either a High or Low output based on the polarity of magnetic pole close to it. Rotor position is determined by analyzing the outputs of all three Hall sensors. Based on the output from hall sensors, the voltages to the motor's three phases are switched.

The advantage of Hall sensor-based commutation is that the control algorithm is simple and easy to understand. Hall sensor-based commutation can also be used to run the motor at very low speeds. The disadvantages are that its implementation requires both separate Hall sensors inside the motor housing and additional hardware for sensor interface.

Sensorless Commutation. In the sensorless commutation technique, the back-EMF induced in the idle phase is used to determine the moment of commutation. When the induced idle-phase back-EMF equals one-half of the DC bus voltage, commutation is complete.

The advantage of sensorless commutation is that it makes the hardware design simpler. No sensors or associated interface circuitry are required. The disadvantages are that it requires a relatively complex control algorithm and, when the magnitude of induced back-EMF is low, it does not support low motor speeds.

When a BLDC motor application requires high torque, when the motor is running at low speed, or when the motor is moving from a standstill, the Hall sensor commutation technique is an appropriate choice. A motor used in an electric bicycle application, for example, requires high initial torque and is a perfect application for Hall sensor commutation.

Furthermore, two voltage application techniques can be applied, based on the configuration of the supply-to-motor windings:

Sinusoidal. Sinusoidal voltage is continuously applied to the three phases. Sinusoidal voltage provides a smooth motor rotation and fewer ripples.

Trapezoidal. DC voltage is applied to two phases at a time, and the third phase remains idle. Trapezoidal voltage is less complex to implement. The idle phase is generating the BEMF from the rotating magnet that passes the unenergized idle phase and provides the BEMF zero-crossing data.

How Hall Sensor Commutation Works

To better understand how Hall sensor commutation works, let's look at how it's implemented with a two-pole motor. Six different commutation states are required to turn the rotor one revolution. The motor's commutation states are shown in Figure 2.

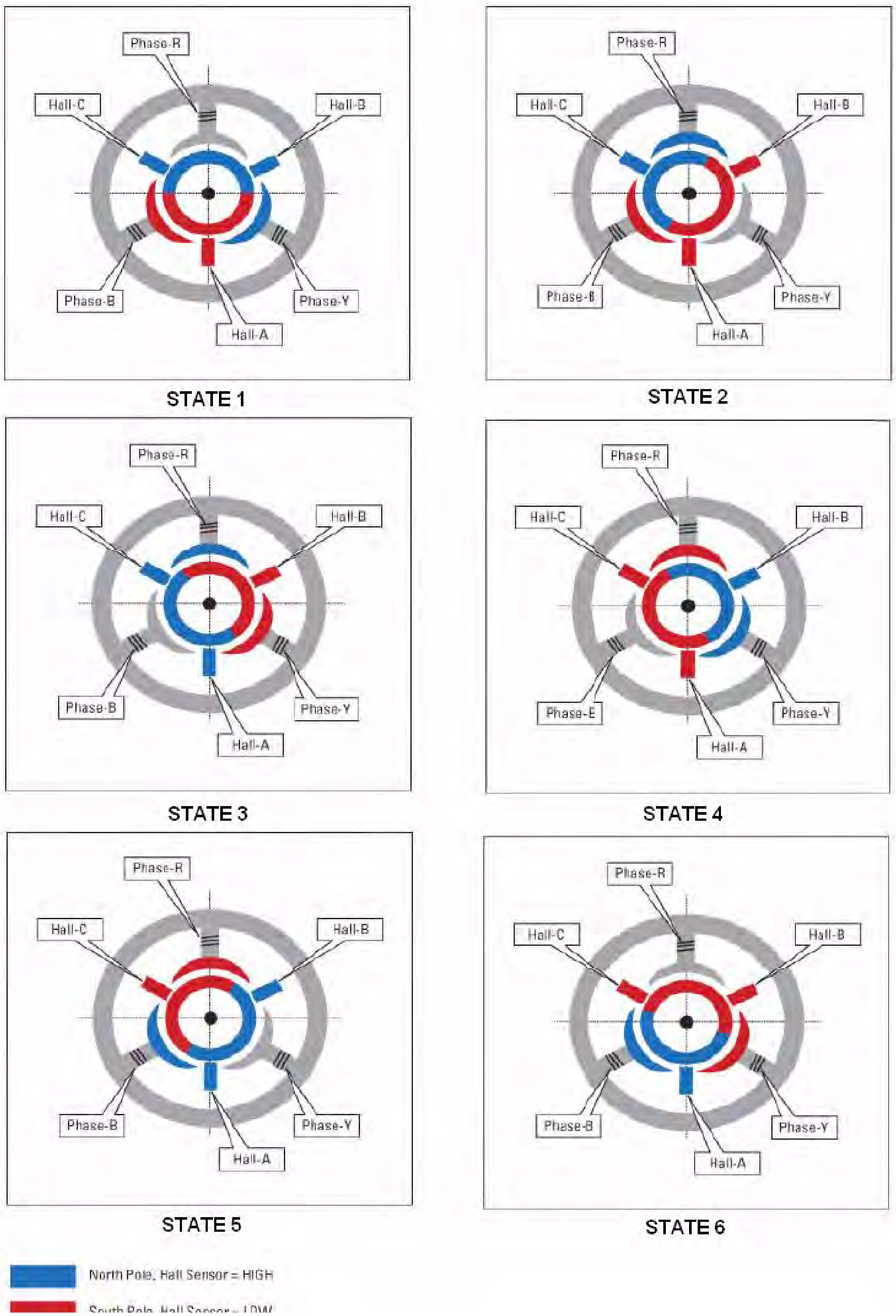

Figure 2. Hall Sensor Commutation States for a 2-Pole Motor

Table 1 indicates the relationship between the Hall sensor output and phase switching operations shown in Figure 2.

Table 1. Relationship Between Hall Sensor Output and Phase Switching

State	Hall A	Hall B	Hall C	Phase B	Phase C	Phase A
1	0	1	1	0	$+V_{DC}$	$–V_{DC}$
2	0	0	1	$+V_{DC}$	0	$–V_{DC}$
3	1	0	1	$+V_{DC}$	$–V_{DC}$	0
4	0	1	0	0	$–V_{DC}$	$+V_{DC}$
5	1	1	0	$–V_{DC}$	0	$+V_{DC}$
6	1	0	0	$–V_{DC}$	$+V_{DC}$	0

Table 2 lists the rating of the motor used to develop this application.

Table 2. Motor Rating for Electric Bike BLDC Motor Control Application

Type of Motor	Linix BLDC
Power Rating	30W
Speed	3200RPM
Number of poles	6
Voltage	24V

Additionally, the application uses a 3-amp High Rupturing Capacity (HRC) fuse.

Using the Z8FMC16100 MCU in an Electric Bike BLDC Motor Controller

Figure 3 offers a visual overview of the electric bike BLDC motor controller. For more details about hardware connections, see Appendix A. Schematics on page 12.

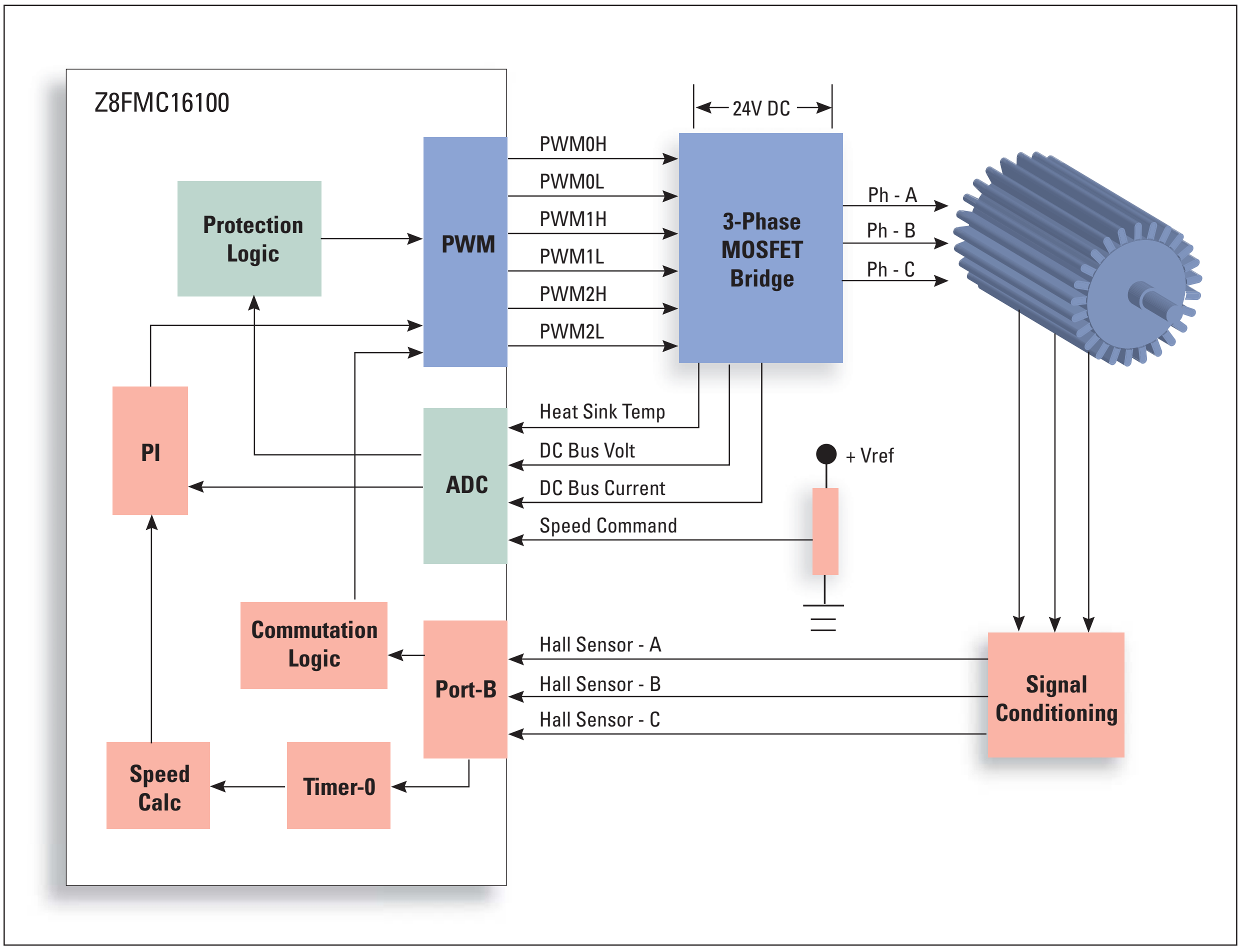

Figure 3. Electric Bike BLDC Motor Controller Block Diagram

Hardware Architecture

The design involves running the BLDC motor in a closed loop or an open loop, with speed as set by a potentiometer. As shown in the architecture diagram, the design generates PWM voltage via the Z8FMC16100 MCU's PWM module to run the BLDC motor.

After the motor is running, the states of the three Hall sensors change based on the rotor position. Voltage to each of the three motor phases is switched based on the state of the sensors (commutation). Hall sensor interrupts capture timer ticks every sixty degrees to measure the rotor speed of the motor. Other peripheral functions can be used to protect the system in case of overload, undervoltage, and overtemperature.

The hardware is described in the following sections.

Three-Phase Bridge MOSFET

The three-phase bridge MOSFET consists of six MOSFETs connected in bridge fashion used to drive the three phases of the BLDC motor. The DC bus is maintained at 24 V, which is same as voltage rating of BLDC motor. A separate Hi-Lo gate driver is used for each high- and low-side MOSFET phase pair, making the hardware design simpler and robust. The high-side MOSFET is driven by charging the bootstrap capacitor.

The DC bus voltage is monitored by reducing it to suitable value using a potential divider. The DC bus current is monitored by putting a shunt in the DC return path. An NTC-type temperature sensor is mounted on MOSFET heat sink, providing analog voltage output proportional to temperature.

PWM Module

The Z8FMC16100 MCU contains a six-channel, 12-bit PWM module configured in this application to run in Complementary Mode. The switching frequency is set to 20KHz. The output on the PWM outputs is controlled according to the inputs from the Hall sensors.

The inputs from the Hall sensors determine the sequence in which the three-phase bridge MOSFET is switched. The Duty cycle of the PWM is directly proportional to the accelerator potentiometer input. The change in the duty cycle controls the current through the motor winding, thereby controlling motor torque.

Commutation Logic

The Hall sensors are connected to ports PB0, PB1 and PB2 on the Z8FMC16100 MCU. An interrupt is generated when the input state on any pin changes. An interrupt service routine checks the state of all three pins and accordingly switches the voltage for the three phases of the motor.

Trapezoidal commutation is used for this application to make implementation simple. In this process of commutation, any two phases are connected across the DC bus by switching the top MOSFET of one phase and bottom MOSFET of another phase ON. The third phase is left un-energized (both top and bottom MOSFET of that phase are switched OFF).

Speed Measurement

The Hall sensor outputs are connected to to ports PB0, PB1 and PB2. One out of three Hall sensors is used to capture the Timer0 ticks, which represent the actual Hall period for closed loop calculations.

Closed Loop Speed Control

Closed-loop speed control is implemented using a PI loop, which works by reducing the error between the speed set by the potentiometer and actual motor speed. The output of this PI loop changes the duty cycle of the PWM module, thereby changing the average voltage to the motor, and ultimately changing the power input. The PI loop adjusts the speed at the same rate as the Hall frequency from one of three Hall sensors. In this application, Open Loop operation is selected in the software by default because any rider of the e-bike will control the speed of the bike.

Protection Logic

The ADC module periodically checks DC bus voltage, DC bus current, and heat sink temperature. If these values go beyond the set limits, the motor is shut down. These checks are timed by Timer0 interrupt.

Over-Current Hardware Protection

The Z8FMC16100 MCU has a built-in comparator that is used to shut down the PWM for over-current protection. When the current exceeds the set threshold, a PWM Comparator Fault is generated to turn OFF the PWM Module.

Software Implementation

During implementation of the software, the following actions are performed:

Initialization. Hardware modules are initialized for the following functions:

- Switch from internal to external oscillator for system operation
- Enable alternate function on respective pins for ADC, Comparator, UART, PA6 as GPIO configured to drive LED
- Configure Timer0 to run in Continuous Mode to capture the Hall period timing
- Configure the comparator to shut down the PWM module when an overcurrent results
- Enable the Op Amp to measure the DC bus current flowing to the motor
- Configure the ADC to read analog values such as DC bus voltage, current, temperature, and acceleration potentiometer (only one channel at a time)
- Configure the PWM module for the individual mode of operation with a 20 kHz switching frequency, control output depending on the values in the PWMOUT Register, and drive the PWMOUT as defaulted to a low off state at Power-On Reset and at any Reset
- Configure the Reset/Fault0 pin functions as a Fault0 function
- Write-protect Flash memory
- Enable Open Loop operation (shown in the main.h file)
- Hardware control of the application, with the UART disabled

Interrupt. The Port B interrupt controls commutation. The Hall sensor output is read on pins PB0:2, the software performs its filtering operation, and the switching sequence of the MOSFET is determined. The PWM timer interrupt is used to time periodically occurring tasks and for the background loop to read analog values from different channels and average these values, update the LED indicator status, and update the read parameters on the UART.

For a visual representation of the application, see Appendix B. Flowcharts on page 17.

Testing/Demonstrating the Application

This section presents a list of the equipment used and procedures observed to test this e-bike application.

Equipment Used

Testing for this application was conducted using the following equipment:

- Z8FMC16100 Series Motor Control Development Kit
- Tektronix digital oscilloscope
- Fluke multimeter
- 30W BLDC motor
- 24V 7Ah battery
- Tektronix power supply

System Configuration

The system requirements on your PC are as follows:

- Windows 7 OS
- ZDS II version 5.0.0 installed
- Optically isolated USB smart cable for program download and debugging

Procedure

Observe the following steps to test the BLDC motor:

1. On the Zilog MC MDS Board, configure the following jumpers:
   a. Shunt the 1–2 positions on J4, J5, and J6
   b. Shunt J2 and J3
2. Set the R7 potentiometer on MC MDS Board (99C0987; contained in the Z8FMC16100 Series Motor Control Development Kit) to the middle position to start the motor.
3. Connect the following wires from the motor to the 3-Phase Motor Control Application Board (contained in the Z8FMC16100 Series Motor Control Development Kit but not shown in the schematic).

   a. Connect the heavy-gauge blue wire to Motor Phase A (P1)
   b. Connect the heavy-gauge green wire to Motor Phase B (P2)
   c. Connect the heavy-gauge white wire to Motor Phase C (P3)

4. Connect the following wires from the motor to the MC MDS Board (contained in the Z8FMC16100 Series Motor Control Development Kit but not shown in the schematic).
   a. Connect the Hall Sensor-C light gauge blue wire to PB2 at J1 pin 6 with additional added 680pF ceramic capacitor to GND.
   b. Connect the Hall Sensor-B light gauge green wire to PB1 at J1 pin 4 with additional added 680pF ceramic capacitor to GND.
   c. Connect the Hall Sensor-A light gauge white wire to PB0 at J1 pin 2 with additional added 680pF ceramic capacitor to GND.
   d. Connect the light gauge black wire (Sensor Power Ground) to GND at TP1.
   e. Connect the light gauge red wire (Sensor Power V_{CC}) to V_{CC} (3.3V) at TP2.

5. Connect the oscilloscope across the motor terminals.
6. Connect the motor control board to the 24V power supply.
7. Build the code on ZDSII v5.0.0 and download the code through USB smart cable.
8. Measure the performance of motor at different loads, for each speed setting of the potentiometer.
9. Record the readings and carry out the process for each step in the test sequence.

Test Results

Laboratory performance test of BLDC motor is as follows:

1. Minimum motor speed: 800 RPM
2. Maximum motor speed: 3200 RPM
3. Power consumption: 6W at 3200 RPM (no load)
4. Regenerative Current at 200RPM: 350 mA

Future Implementation

The application discussed in this document covers the motoring and regenerative braking features for a BLDC hub motor used in an electric bike. Further improvements can be made to the design by adding the following features:

- Controlled charging of SLA batteries by plugging to the AC Mains adaptor
- Implementing ‘Torque-boost’ functionality (through a push-switch), which will give a boost to motor performance

- Utilizing LIN/UART communication to create a dashboard display of measured parameters (speed, Battery voltage, Current, and Fault conditions)

References

The following supporting documents are available free for download from the Zilog website.

- eZ8 CPU User Manual (UM0128)
- Z8FMC16100 Series Product Specification (PS0246)
- PID Motor Control with the Z8PE003 Application Note (AN0030)
- Z8 Encore!-Based SLA Battery Charger Application Note (AN0223)
- Sensorless Brushless DC Motor Control with Z8 Encore! MC Microcontrollers (AN0226)
- Z8 Encore! XP-Based BLDC Fan Control Application Note (AN0228)

Appendix A. Schematics

Figures 4 through 8 show schematic representations of each of the application's key blocks.

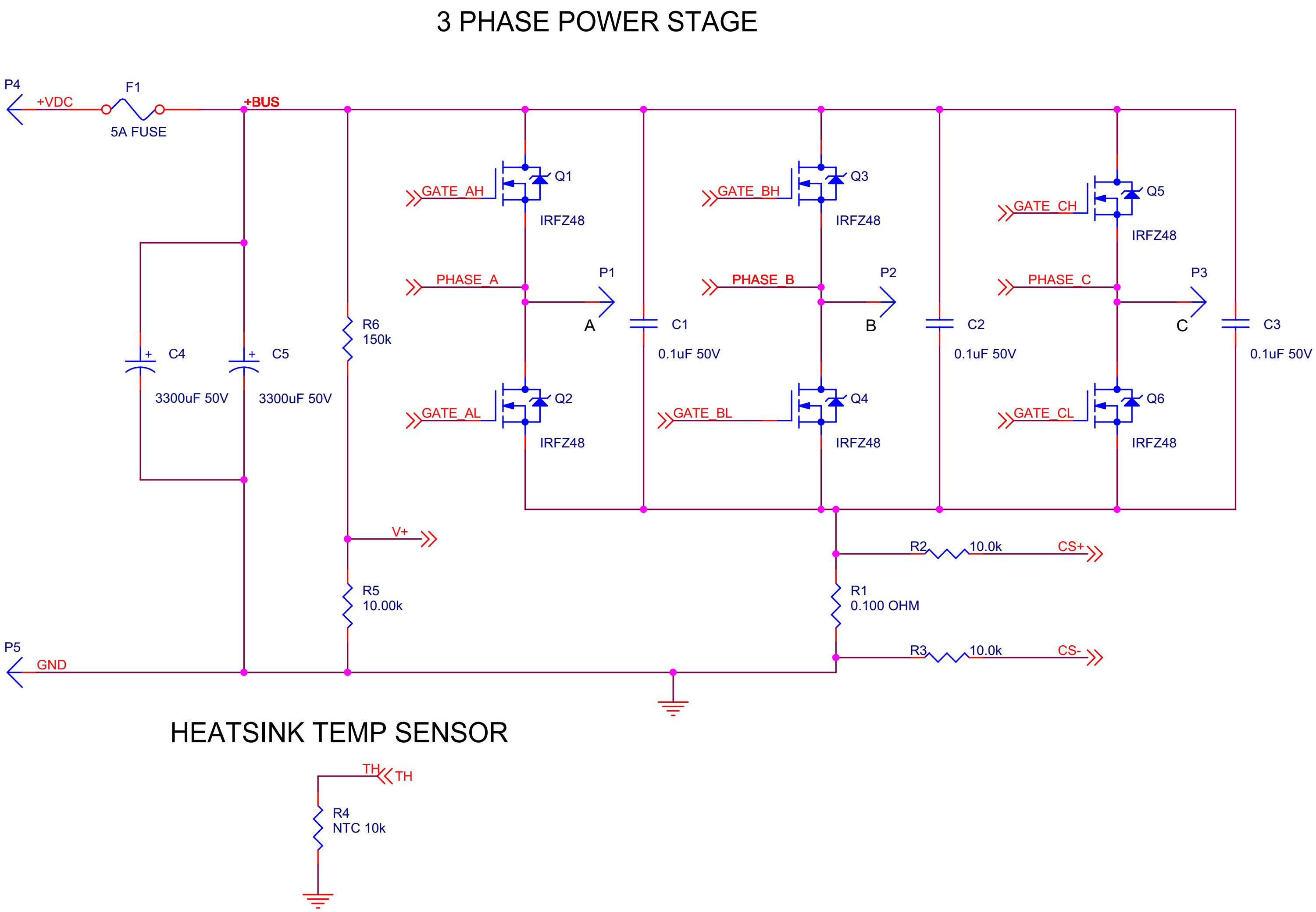

Figure 4. Electric Bike BLDC Motor Controller Application Schematic, #1 of 5

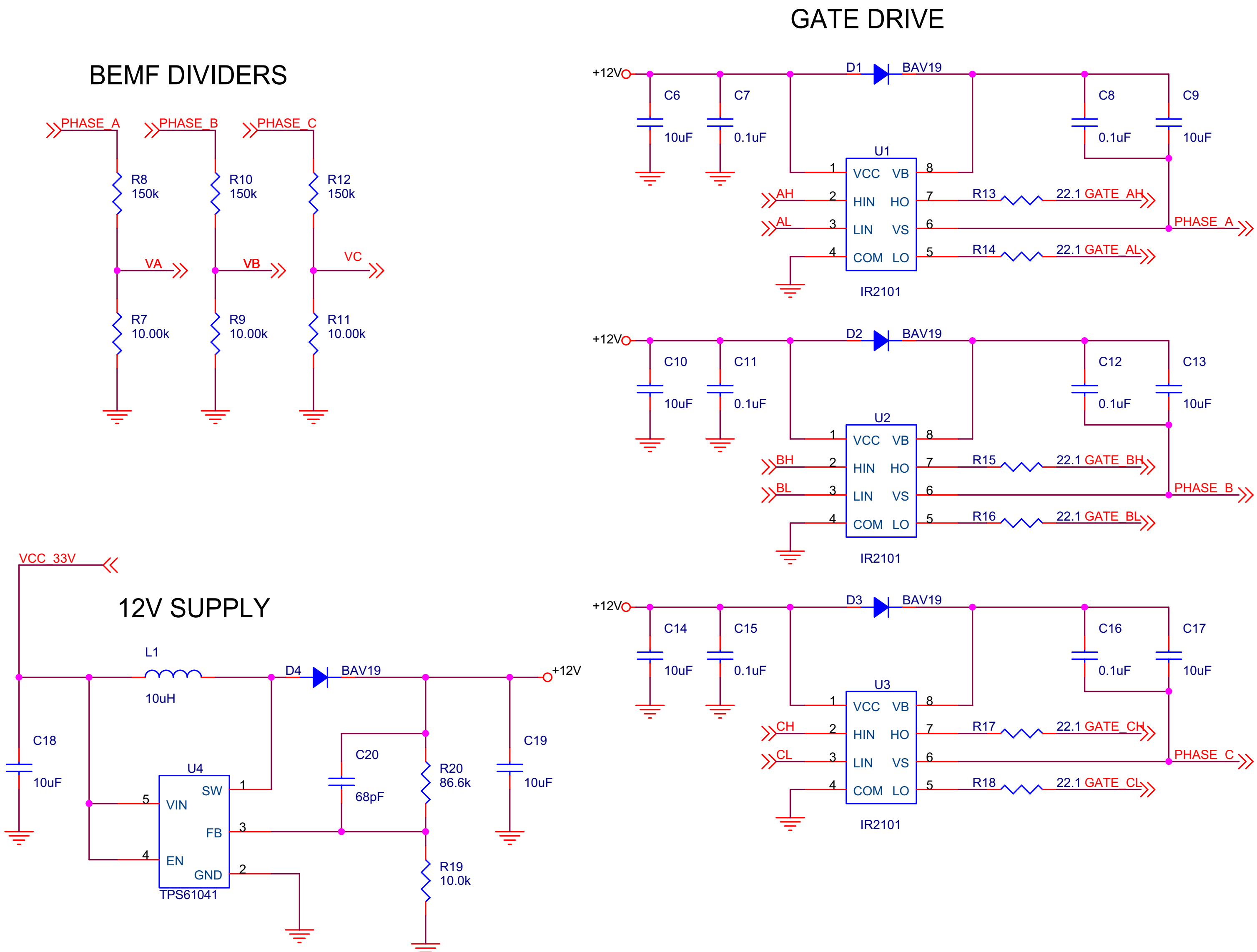

Figure 5. Electric Bike BLDC Motor Controller Application Schematic, #2 of 5

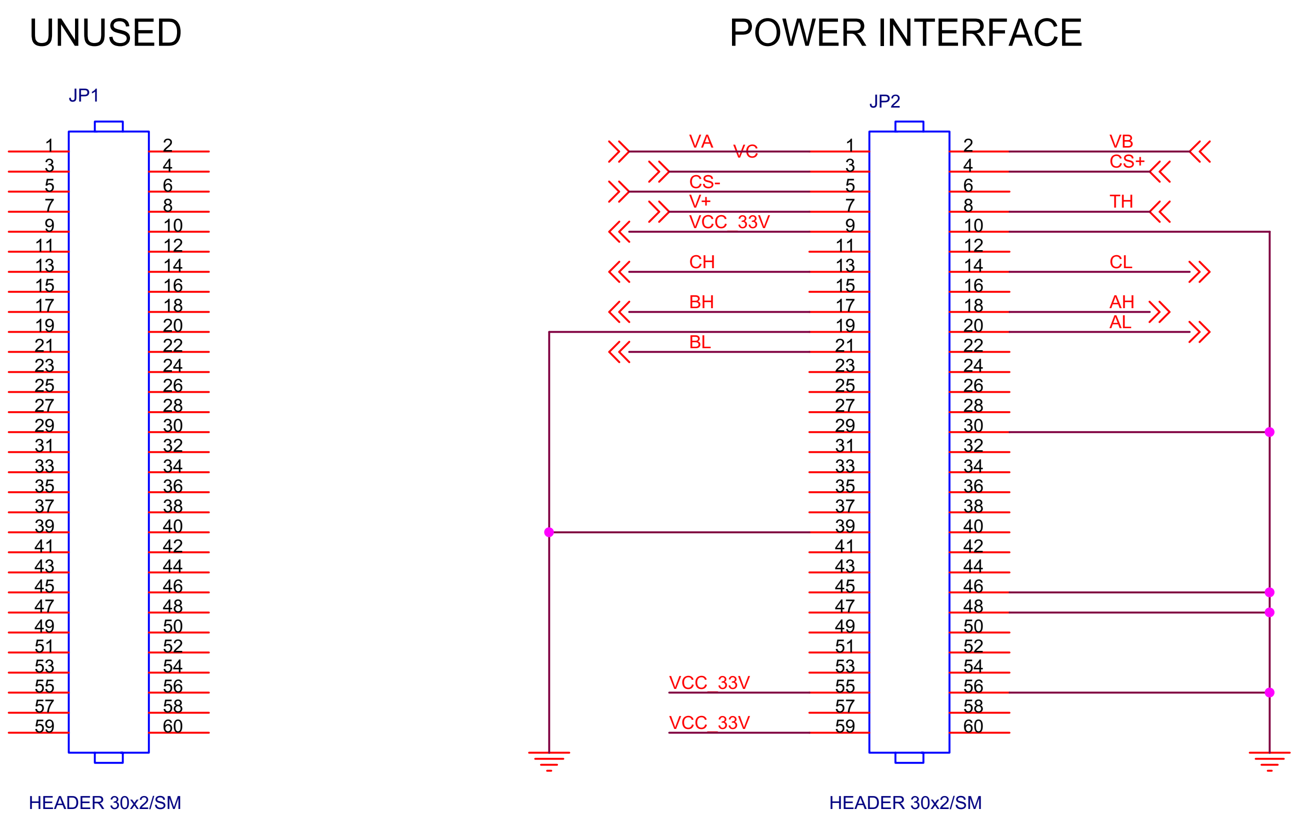

Figure 6. Electric Bike BLDC Motor Controller Application Schematic, #3 of 5

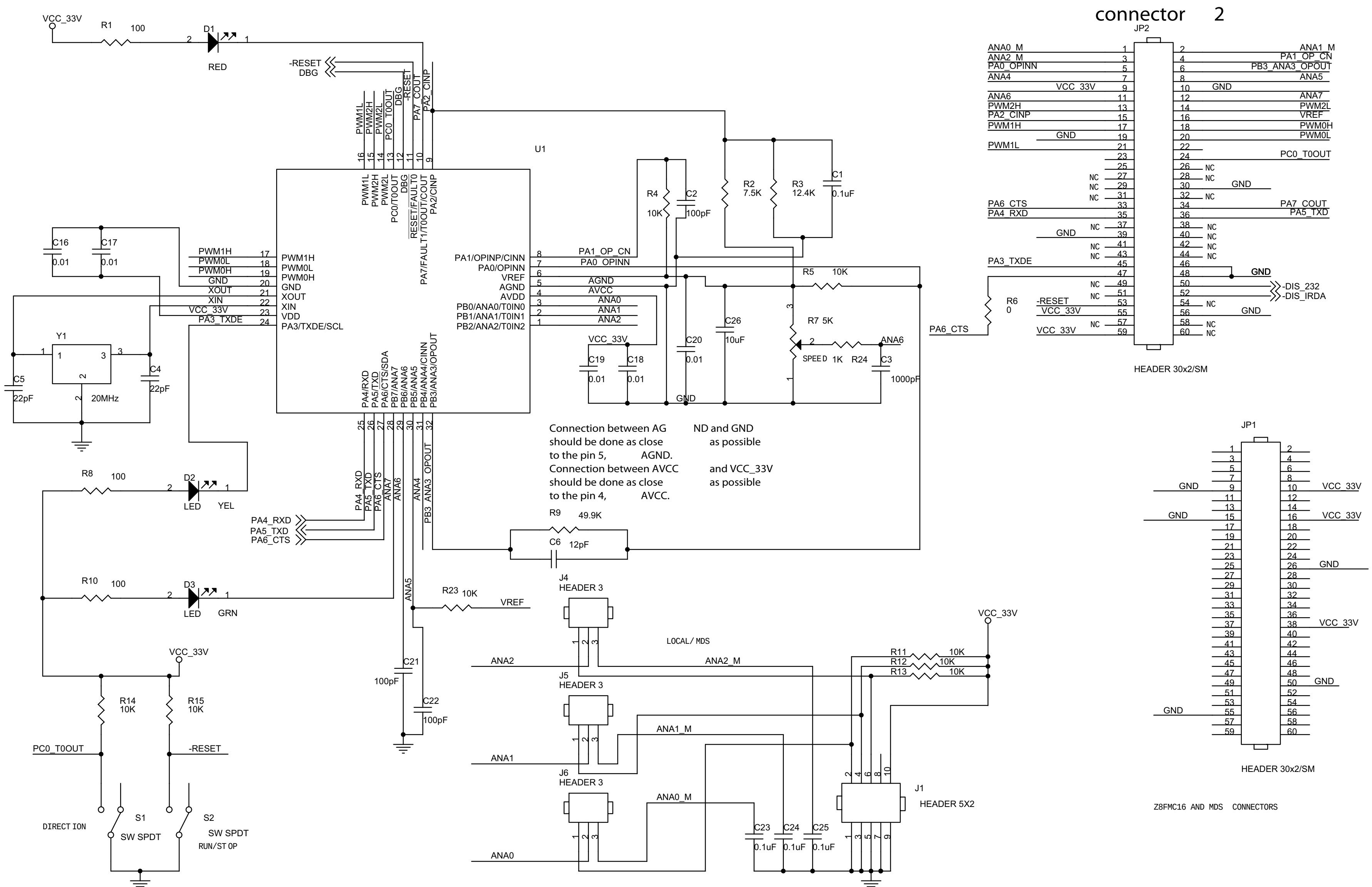

Figure 7. Electric Bike BLDC Motor Controller Application Schematic, #4 of 5

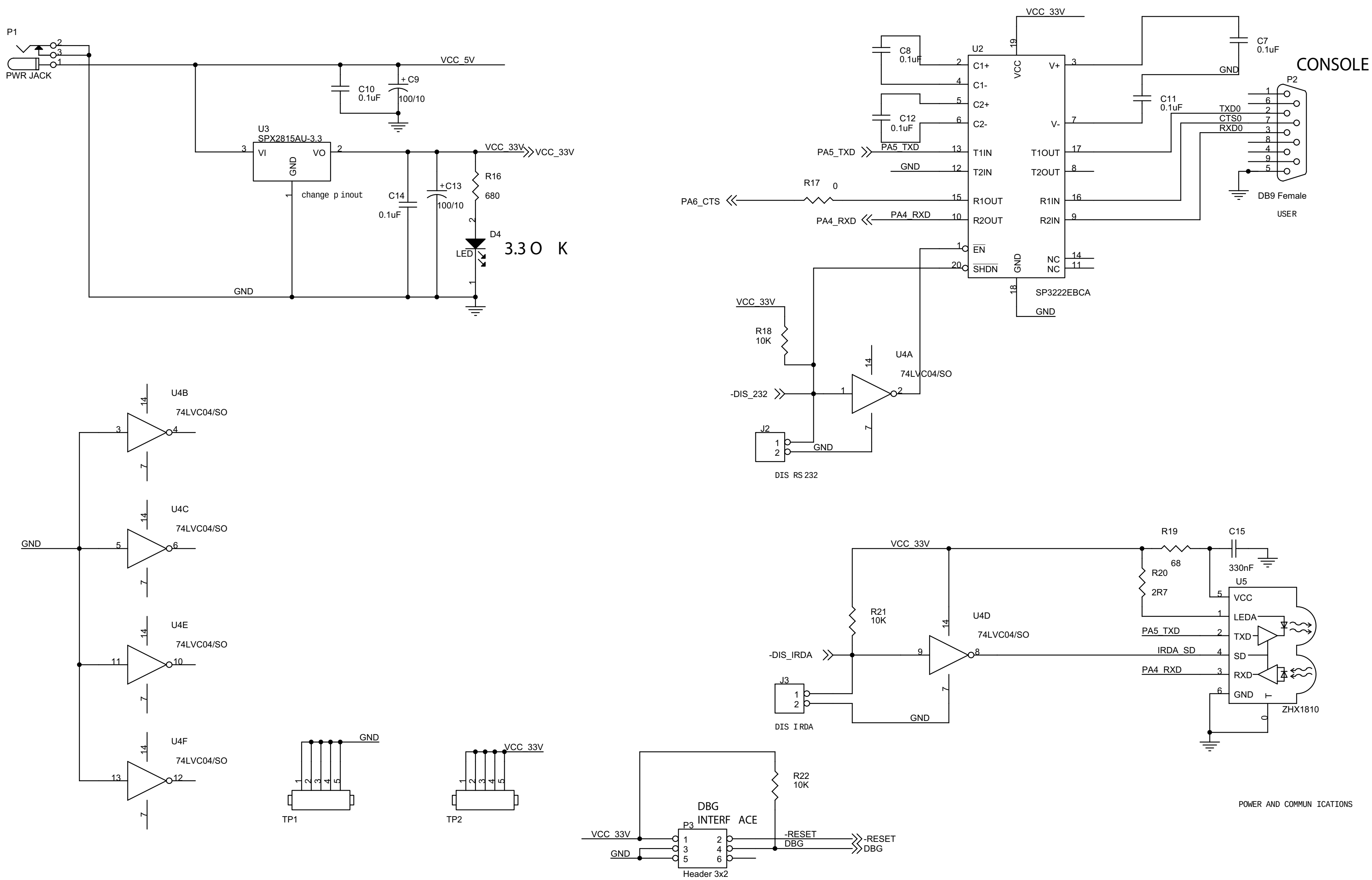

Figure 8. Electric Bike BLDC Motor Controller Application Schematic, #5 of 5

Appendix B. Flowcharts

Figure 9 presents a simple flow chart of the main, timer interrupt and Port B interrupt routines for the electric bike BLDC motor controller application.

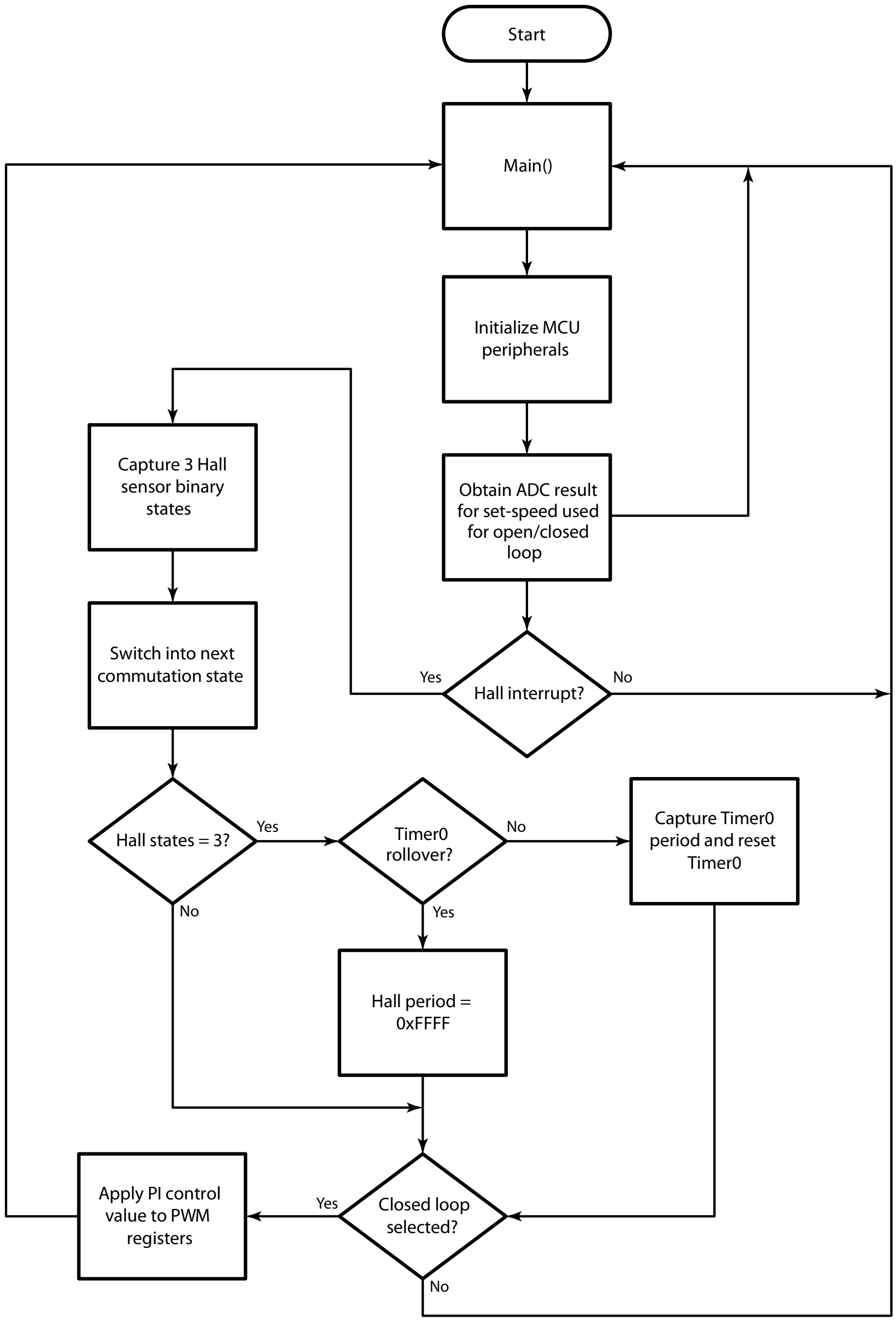

Figure 9. Electric Bike BLDC Motor Controller Application Flowchart

Customer Support

To share comments, get your technical questions answered, or report issues you may be experiencing with our products, please visit Zilog's Technical Support page at http://support.zilog.com.

To learn more about this product, find additional documentation, or to discover other facets about Zilog product offerings, please visit the Zilog Knowledge Base at http://zilog.com/kb or consider participating in the Zilog Forum at http://zilog.com/forum.

This publication is subject to replacement by a later edition. To determine whether a later edition exists, please visit the Zilog website at http://www.zilog.com.

Warning: DO NOT USE THIS PRODUCT IN LIFE SUPPORT SYSTEMS.

LIFE SUPPORT POLICY

ZILOG'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS PRIOR WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF ZILOG CORPORATION.

As used herein

Life support devices or systems are devices which (a) are intended for surgical implant into the body, or (b) support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in a significant injury to the user. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system or to affect its safety or effectiveness.

Document Disclaimer