*User's Guide*

# *MCF8316AEVM Evaluation Module*

## ABSTRACT

This document is provided with the MCF8316A customer evaluation module (EVM) as a supplement to the MCF8316Ax data sheet (MCF8316A Three-Phase Sensorless-FOC BLDC Motor Driver). This User's Guide details the hardware implementation of the EVM and how to setup and power the board.

## Table of Contents

## Trademarks

LaunchPad™ are trademarks of Texas Instruments.
All trademarks are the property of their respective owners.

# 1 Cautions and Warnings

Observe the following cautions and warnings as printed on the EVM board.

**HOT SURFACE:**

**Caution Hot Surface! Contact may cause burns. Do not touch. Please take the proper precautions when operating.**

## 2 Introduction

The MCF8316A is a 4.5-V to 35-V, 8-A peak integrated three-phase gate driver IC with sensorless field-oriented control for motor drive applications. It provides three accurately trimmed and temperature compensated half-bridge MOSFETS, gate drivers, charge pump, current sense amplifier, linear regulator for the external load and adjustable buck regulator. The I2C interface variant (MCF8316A) also provides a standard I2C interface for configuring the various device settings and reading fault diagnostic information through an external controller.

The MCF8316AEVM includes an onboard FTDI chip to convert USB communication from the micro-USB connector into UART and an onboard MSP430FR2355 MCU to interface with the MCF8316A. It can also provide SPI communication for the SPI variant of the MCF8316A device. There are many user-selectable jumpers, resistors, connectors, and test points to evaluate many features of the device and configure device-specific settings.

This document serves as a startup guide to supplement the MCF8316AEVM. It is intended for engineers to design, implement, and validate reference hardware for the MCF8316A device.

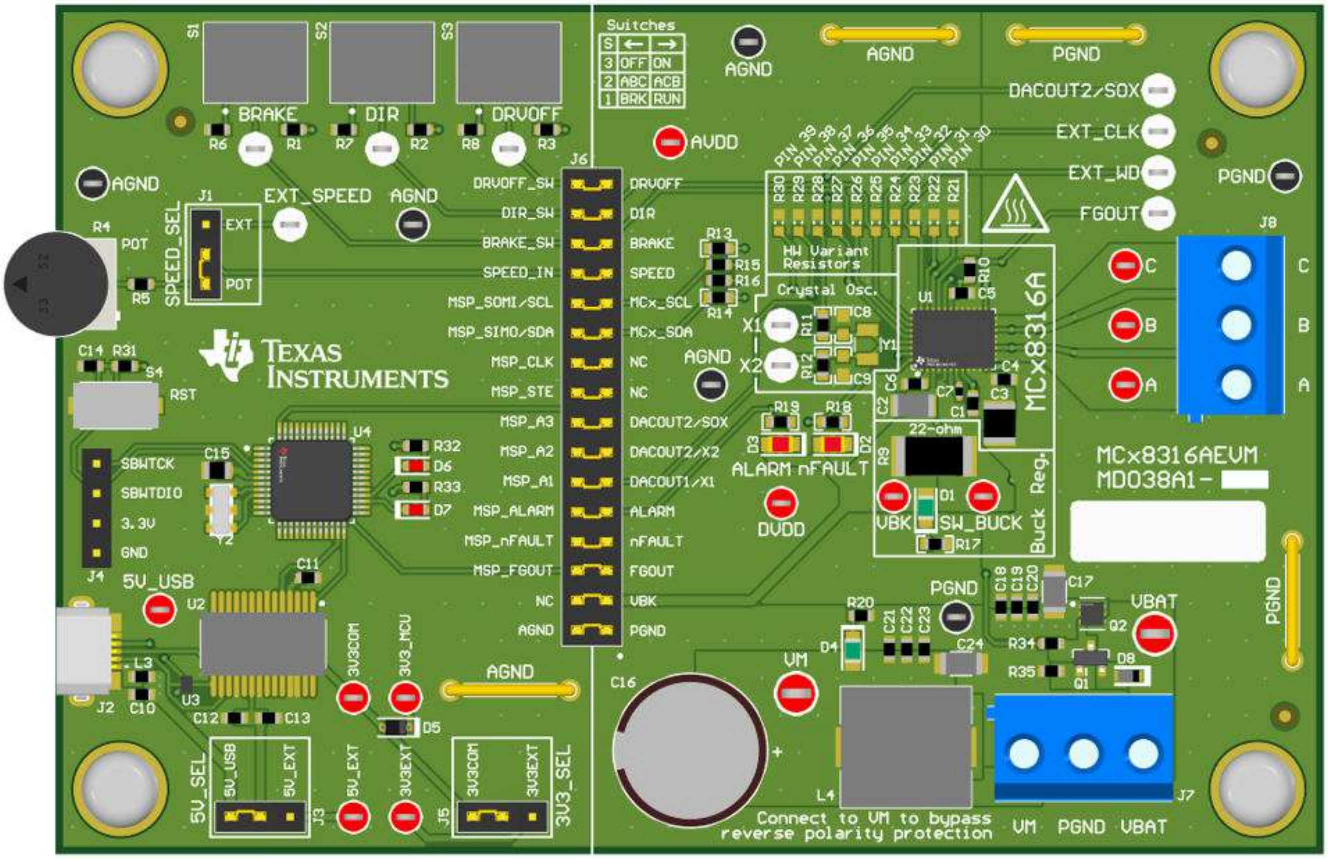


**Figure 2-1. MCF8316AEVM Printed Circuit Board (PCB - Top View)**

## 3 Quick Start Guide

The MCF8316AEVM requires a power supply source, which has a recommended operating range from a 4.5-V to 35-V. To setup and power the EVM, follow the sequence below:

1. Connect motor phases to A, B, C on connector J8.
2. Do not turn on the power supply yet. Connect motor supply to VBAT/VM and PGND on connector J7.
    a. To enable reverse polarity protection and Pi filter, connect to VBAT. Note that when connecting to VBAT, VM will be VM – 0.7 V less, due to a diode drop in the reverse-polarity protection circuit.
    b. To disable reverse-polarity protection and the Pi filter, connect to VM.
3. Select J3 to 5V_USB and J5 to 3V3COM to power MSP430 from USB power supply.
4. Connect the micro-USB cable into the computer.
5. Turn the potentiometer fully clockwise to set the motor to zero speed upon powerup.
6. Flip the switch S1 to the right to configure BRAKE = RUN, switch S2 to the left to configure DIR = ABC, and switch S3 to the right to configure DRVOFF = ON
7. Turn on the motor power supply.
8. Use the potentiometer R4 to control the speed of the motor and the switches to disable the motor driver, change the direction, or apply a brake to the motor. Optionally, use the GUI (as shown in Section 6) to monitor real-time speed of the motor, put the MCF8316A into a low-power sleep mode, and read status of the LEDs.

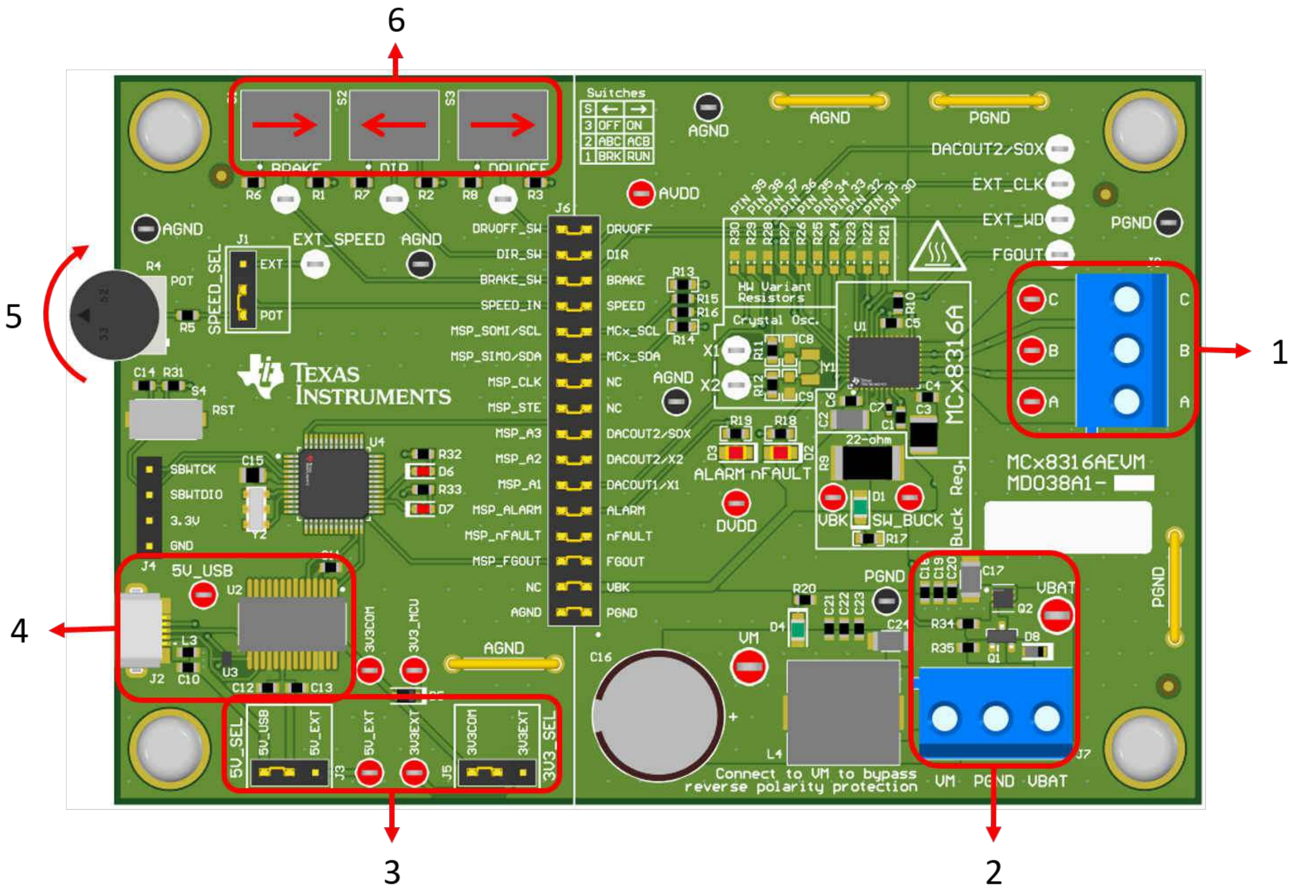


Figure 3-1. Reference for Quick Start Guide

# 4 Hardware and Software Overview

## 4.1 Hardware Connections Overview – MCF8316AEVM

Figure 4-1 shows the major blocks of MCF8316AEVM evaluation module. The MCF8316AEVM is designed for an input supply from 4.5-V to 35-V. The MCF8316A includes three integrated half-bridges and implements a sensorless FOC algorithm to spin a motor with up to 8-A peak current. It also integrates an adjustable buck regulator and can support many types of Hall sensor configurations.

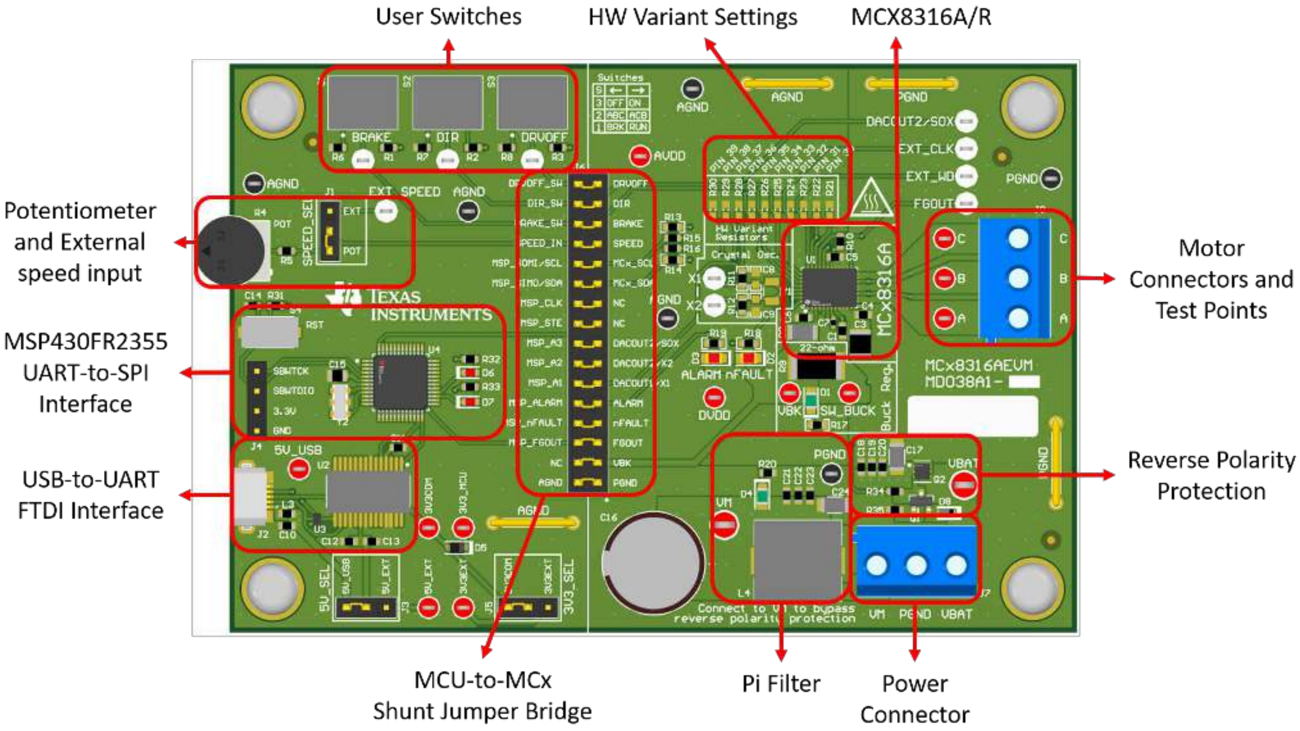

User Switches
HW Variant Settings
MCX8316A/R
Potentiometer and External speed input
MSP430FR2355 UART-to-SPI Interface
USB-to-UART FTDI Interface
Motor Connectors and Test Points
Reverse Polarity Protection
MCU-to-MCx Shunt Jumper Bridge
Pi Filter
Power Connector

**Figure 4-1. MCF8316AEVM Major Hardware Blocks**

## 4.2 Connection Details

Figure 4-2 shows the connections made to the MCF8316AEVM in order to spin a 3-phase sensorless Brushless-DC motor.

An 4.5-V to 35-V power supply or battery is connected to the VBAT or VM and PGND terminals on connector J7. There is a reverse polarity protection and Pi filter implemented on the VBAT and PGND terminals. To bypass the reverse polarity protection and Pi filter, connect the power supply to the VM terminal or VM test point on the board and PGND.

The three phases of the BLDC motor connect directly to the A, B, and C terminals of the screw terminal connector J8 provided on the MCF8316AEVM.

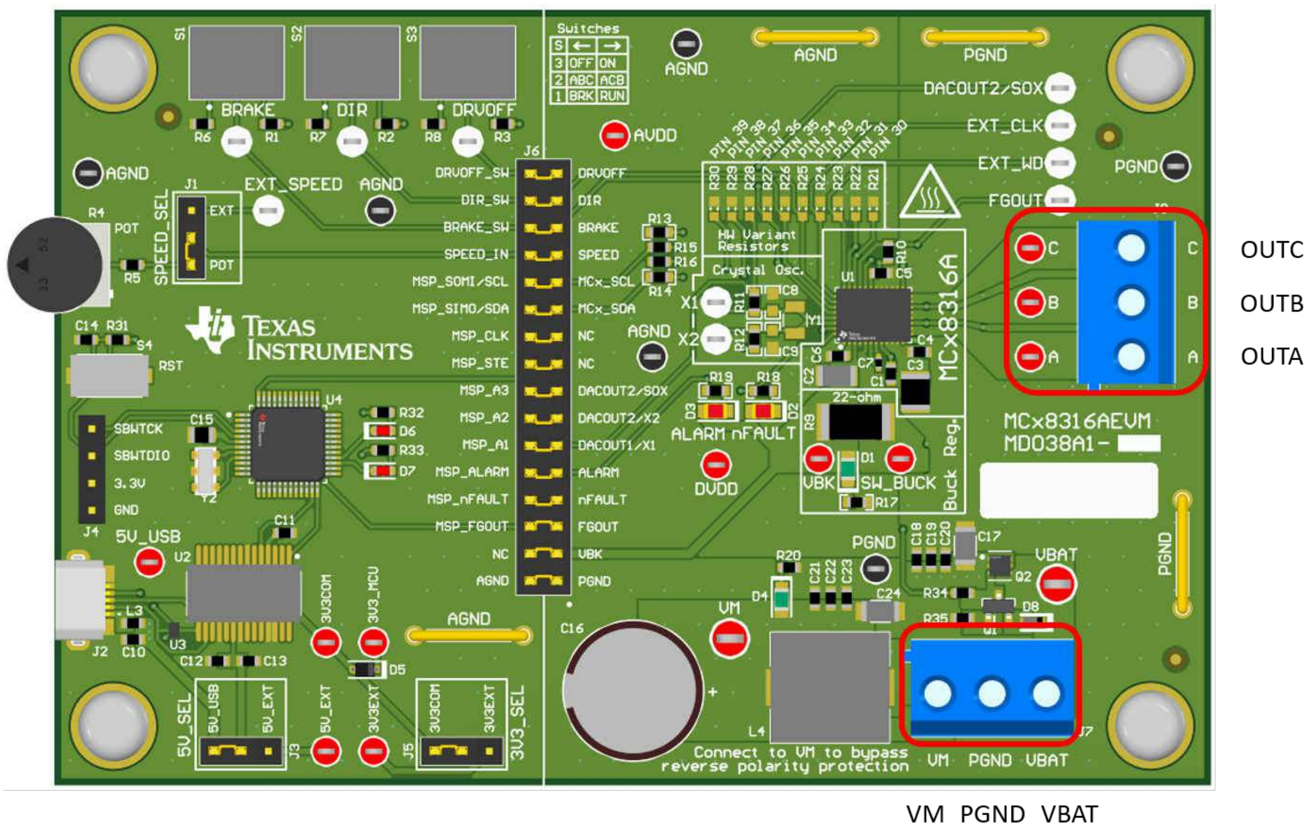


**Figure 4-2. Connections from Motor to MCF8316AEVM**

Figure 4-3 MSP430FR2355 Microcontrollershows where the micro-USB cable is plugged into the MCF8316AEVM to provide communication between evaluation module and GUI. The USB data and 5 V power from the USB is converted into UART data and 3.3 V power to power the MSP430FR2355 microcontroller. The 5 V from the USB power is limited to 500 mA and the 3.3 V from the FTDI chip is limit to 30 mA. If the user wishes to supply more current to these rails, they may use the 5V_SEL jumper J3 and 3V3_SEL jumper J5 to connect external power rails.

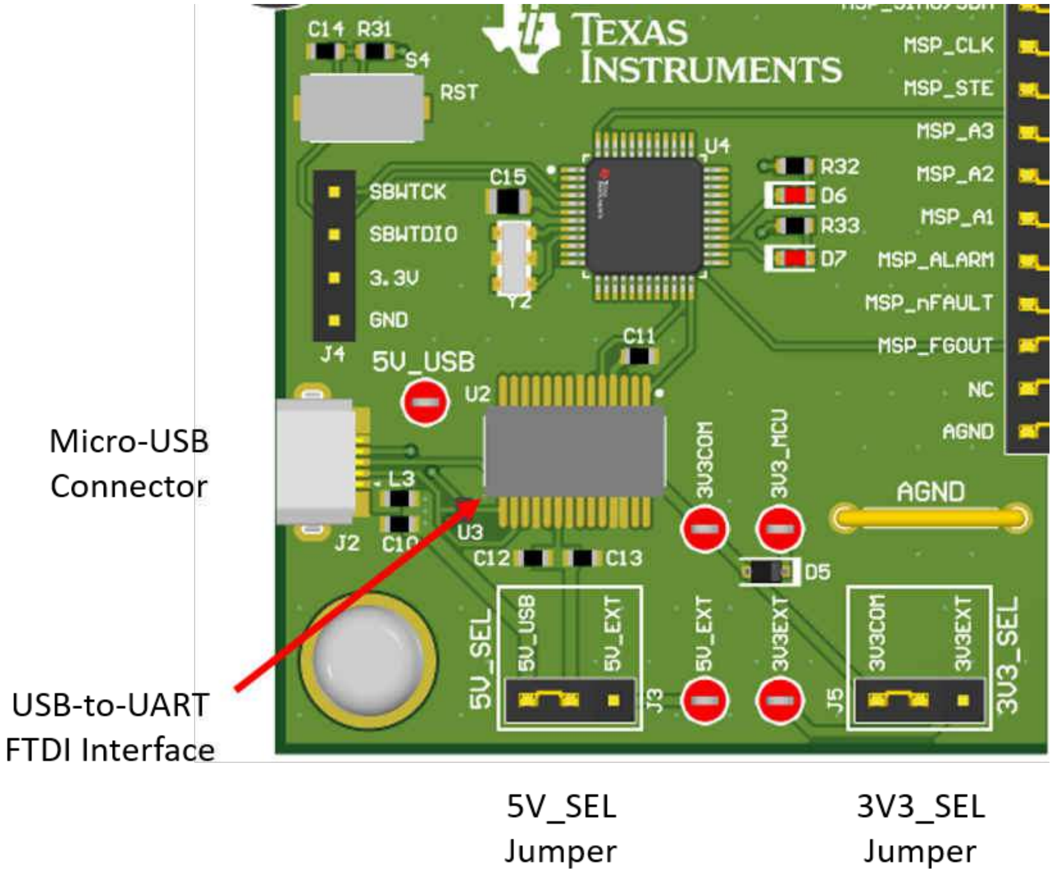


**Figure 4-3. Micro-USB Connector and UART for MCF8316AEVM**

### 4.3 MSP430FR2355 Microcontroller & User Interface

The MCF8316AEVM includes the MSP430FR2355 low-power MCU (as shown in Figure 4-4) to communicates via I2C with the MCF8316A.

In order to program the MSP430FR2355, an external MSP430 FET programmer must be connected to the Spy-Bi-Wire (SBW) interface connector J4. Many MSP430 LaunchPad™ provide an onboard eZ-FET Debug Probe that can be jumper-wired to the MCF8316AEVM to flash the firmware into the MSP430FR2355 microcontroller.

The user can use the Reset (RST) button at any time to reset and restart the MCU program. Two active-low LEDs, D6 and D7, can be used for debug purposes as well.

Finally, a shunt jumper bridge on the 32-pin connector J6 ties all signals between the microcontroller and MCF8316A. These jumpers can be inserted or removed as needed in order to isolate the microcontroller from the gate driver. This allows for microcontroller signal debugging or using the MCF8316AEVM as a standalone gate driver with an external microcontroller.

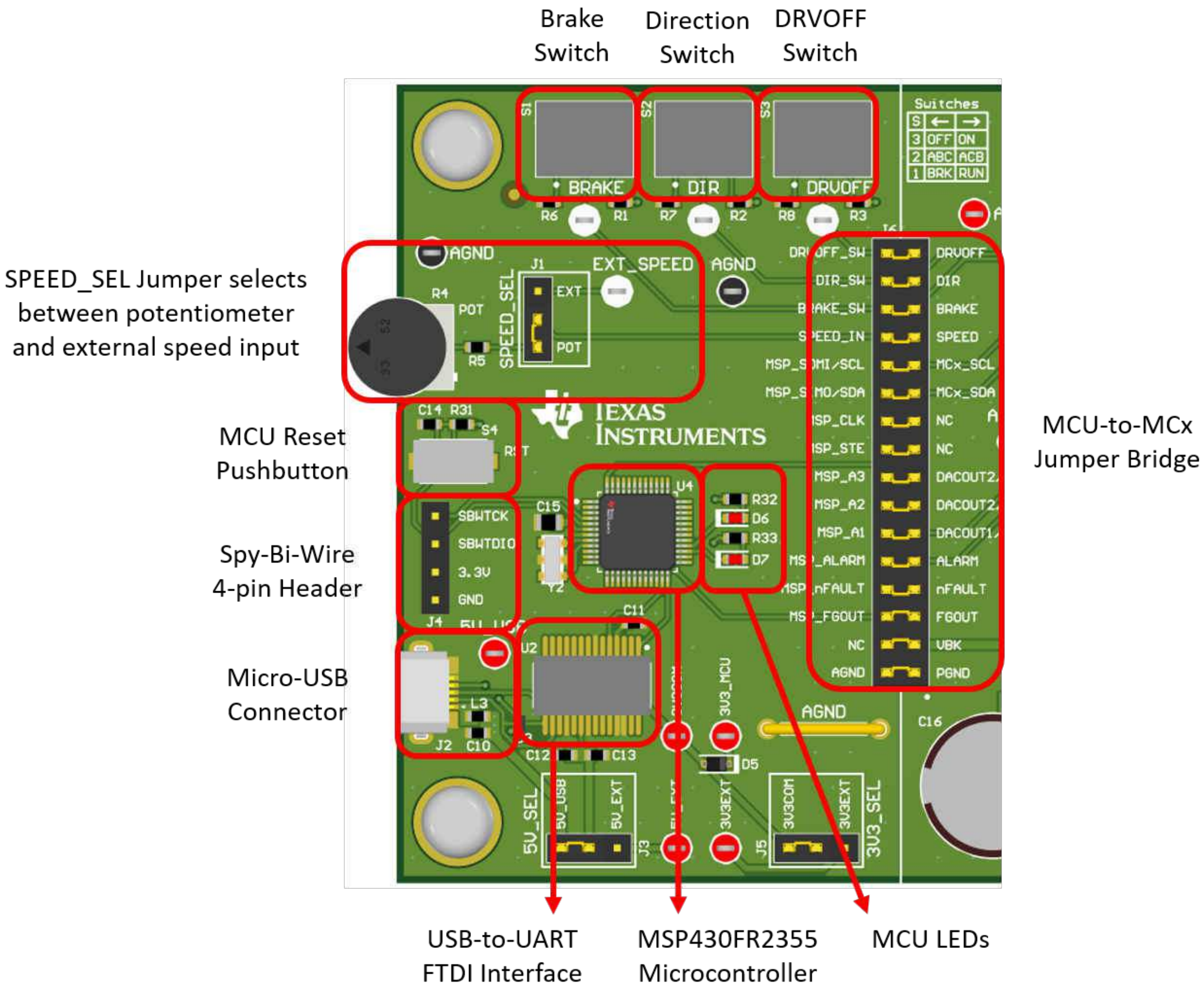


**Figure 4-4. MSP430FR2355 MCU and User Interface on MCF8316AEVM**

## 4.4 LED Lights

The MCF8316AEVM has 5 status LEDs implemented that provide the status of power supplies and functionalities of the evaluation module. By default, the VM LED and 3.3 V Buck LEDs will light up when the board is powered and the program has been flashed onto the microcontroller. Table 4-1 shows LED descriptions including those that are on during power up in bold and Figure 4-5 shows the locations of the LEDs.

**Table 4-1. Description of MCF8316AEVM LEDs (default in bold after powerup)**

| Designator | Name | Color | Description |
|---|---|---|---|
| **D1** | **Buck Regulator** | **Green** | **Internal buck regulator is voltage output** |
| D2 | nFAULT | Red | Lights up when fault condition has occurred on MCF8316A |
| D3 | ALARM | Red | Lights up when alarm condition has occurred on MCF8316A |
| **D4** | **VM** | **Green** | **Motor power is supplied to the board** |
| **D5** | **MSP_LED1** | **Red** | **Used for UART or debugging** |
| D6 | MSP_LED2 | Red | Used for UART or debugging |

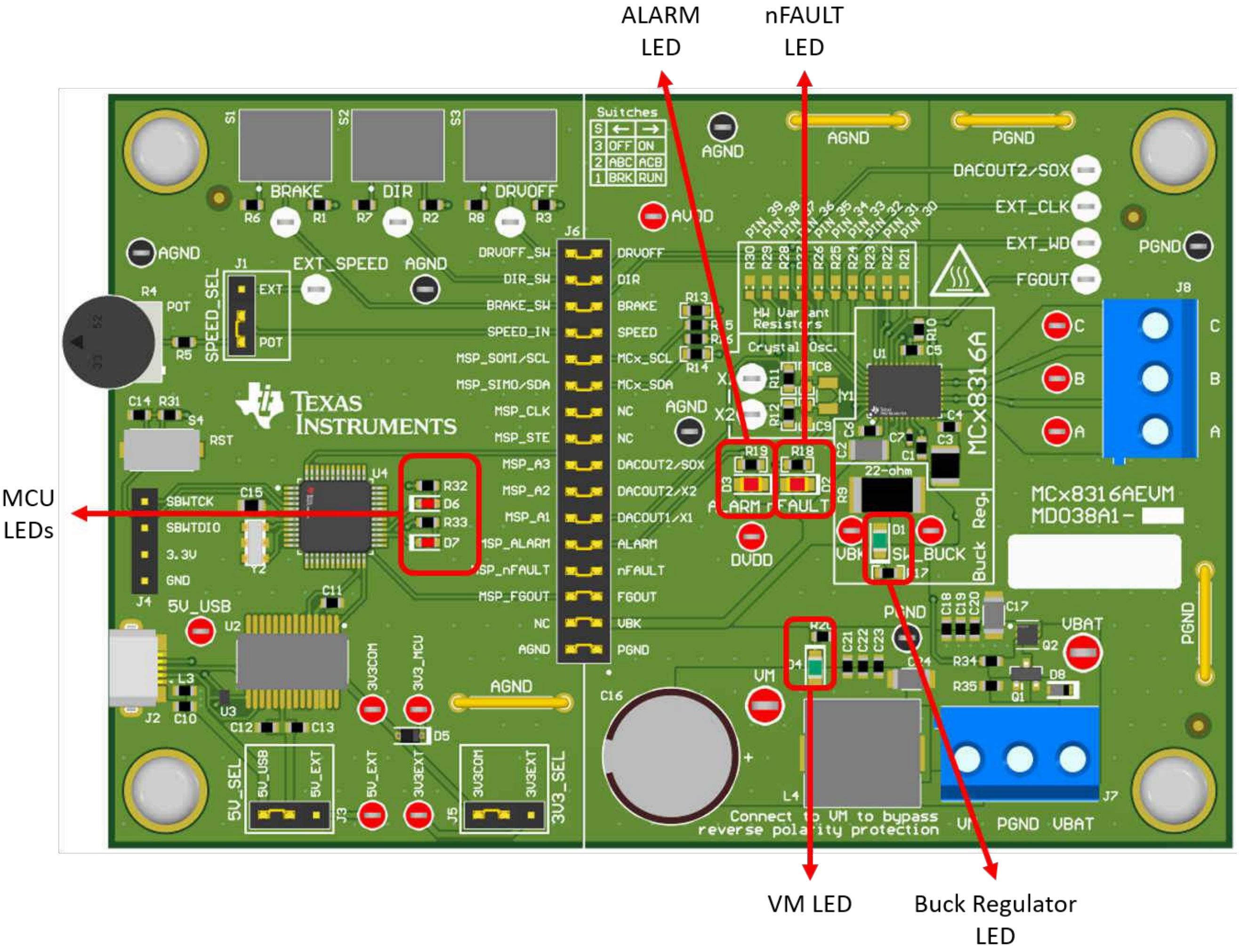


**Figure 4-5. MCF8316AEVM LEDs**

## 4.5 User-Configurable Settings

The MCF8316AEVM includes a variety of user-selectable jumpers, switches, and resistors on the entirety of the evaluation board to configure settings. Table 4-2 summarizes all of these configurable settings.

**Table 4-2. Description of User-Selectable Settings on MCF8316AEVM (Default in Bold)**

| Designator | Setting Name | Description | Layer | Position | Function |
|---|---|---|---|---|---|
| L1/L2/R9 | Buck Regulator mode | User populates L1, L2, or R9 to choose switching component for buck regulator | Top | L1 = 47uH Inductor | Inductor Mode |
| | | | Bottom | L2 = 22 µH | Inductor Mode |
| | | | Bottom | **R1 = 22 Ω** | **Resistor Mode** |
| J5 | 3V3_SEL | Select 3.3 V for MCU power | Top | J5 = 3V3EXT | External |
| | | | | **J5 = 3V3COM** | **From FTDI (30 mA)** |
| J3 | 5V_SEL | Select 5 V for FTDI power | Top | J3 = 5V_EXT | External |
| | | | | **J3 = 5V_USB** | **From USB power (500 mA)** |

### Table 4-2. Description of User-Selectable Settings on MCF8316AEVM (Default in Bold) (continued)

| Designator | Setting Name | Description | Layer | Position | Function |
|---|---|---|---|---|---|
| J1 | SPEED_SEL | Selects SPEED input source | Top | J1 = EXT | External EXT_SPEED test point |
| | | | | **J1 = POT** | **From Potentiometer R4** |
| | | | | J1 is removed | Floating |
| J6 | MSP to MCx Shunt jumper bridge | Connects signals from MCU and user switches to MCx8316A when jumpers are inserted | Top | **DRVOFF_SW** | **DRVOFF** |
| | | | | **DIR_SW** | **DIR** |
| | | | | **BRAKE_SW** | **BRAKE** |
| | | | | **SPEED_IN** | **SPEED** |
| | | | | **MSP_SOMI/SCL** | **MCx_SCL** |
| | | | | **MSP_SIMO/SDA** | **MCx_SDA** |
| | | | | **MSP_CLK** | **NC** |
| | | | | **MSP_STE** | **NC** |
| | | | | **MSP_A3** | **DACOUT2/SOX** |
| | | | | **MSP_A2** | **MCX_DACOUT2/X2** |
| | | | | **MSP_A1** | **MCX_DACOUT1/X1** |
| | | | | **MSP_ALARM** | **ALARM** |
| | | | | **MSP_nFAULT** | **nFAULT** |
| | | | | **MSP_FGOUT** | **FGOUT** |
| | | | | **NC** | **VBK** |
| | | | | **AGND** | **AGND** |
| S1 | BRAKE | Turns on all low-side MOSFETs | Top | **Left** | **Brake enabled** |
| | | | | Right | Brake disabled |
| S2 | DIR | Controls direction of motor | Top | **Left** | **ABC** |
| | | | | Right | ACB |
| S3 | DRVOFF | Disables gate drivers | Top | **Left** | **MCF8316A enabled** |
| | | | | Right | MCF8316A disabled |

# 5 Hardware Setup

The hardware required to run the motor is the MCF8316AEVM, a Micro-USB cable, and a power supply with a DC output from 4.5-V to 35-V. Follow these steps to start up the MCF8316AEVM:

1. Connect the DC power supply to header J7. Connect to VBAT and PGND to apply reverse polarity protection and the pi filter to the EVM. Otherwise, connect to VM and PGND to bypass the reverse polarity protection and pi filter.
2. Apply user configurable jumper settings. See Section 4.5 section for more information.
3. Flash program into the MCU as described in Section 4. Launch the GUI in GUI Composer and disconnect the 4-pin JTAG connections.
4. Connect a Micro-USB cable to the MCF8316AEVM and computer.
5. Turn on the power supply and power up the PCB.

If using the MCF8316AEVM with an external microcontroller, remove all shunt jumpers from jumper bridge J6. Connect with external jumpers to the left side of the jumper bridge from the external MCU.

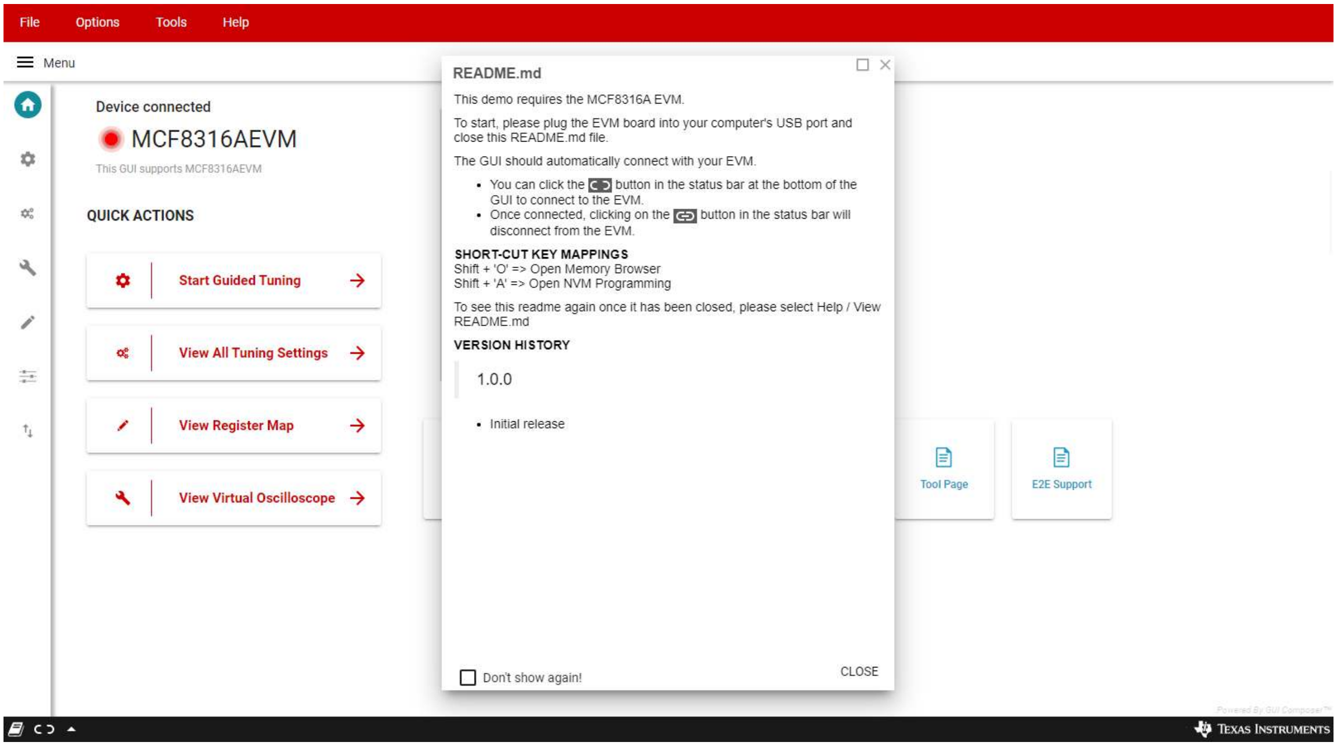
# 6 MCF8316A GUI Application

The MCF8316AEVM includes a USB-UART interface, using a MSP430FR2355 microcontroller, that serves as a communication bridge between a host PC and the MCF8316A device for configuring various device settings and reading fault diagnostic information. A MCF8316A GUI is available to interface with and configure the MCF8316A using this communication interface.

Access the MCF8316A GUI through the TI Cloud Gallery.

## 6.1 Running the GUI

The MCF8316A GUI can be run directly inside a web browser (supported in Google Chrome and Firefox).



README.md

This demo requires the MCF8316A EVM.

To start, please plug the EVM board into your computer's USB port and close this README.md file.

The GUI should automatically connect with your EVM.

- You can click the button in the status bar at the bottom of the GUI to connect to the EVM.
- Once connected, clicking on the button in the status bar will disconnect from the EVM.

**SHORT-CUT KEY MAPPINGS**
Shift + 'O' => Open Memory Browser
Shift + 'A' => Open NVM Programming

To see this readme again once it has been closed, please select Help / View README.md

**VERSION HISTORY**

1.0.0

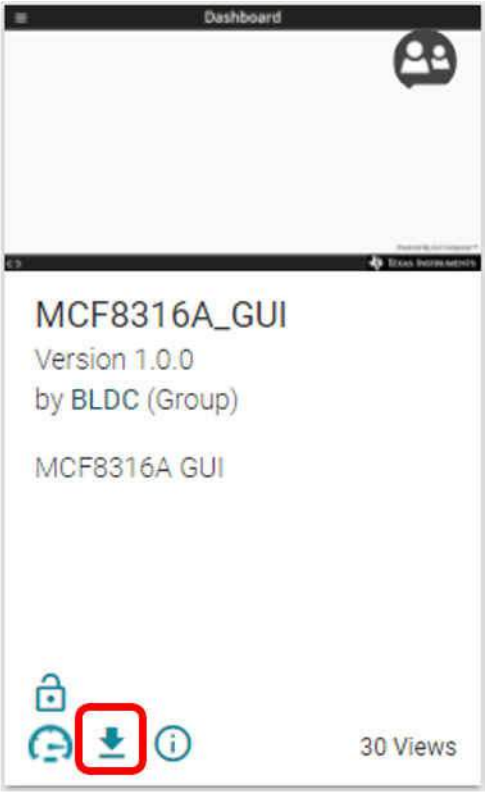
- Initial release

**Figure 6-1. MCF8316A GUI**

Once the GUI is loaded, follow the step-by-step Guided Tuning section of the GUI to configure the device.

## 6.2 Offline Installer

Alternatively, the MCF8316A GUI can be downloaded and installed offline using the download feature in the TI Cloud Gallery.



MCF8316A_GUI

Version 1.0.0

by BLDC (Group)

MCF8316A GUI

30 Views

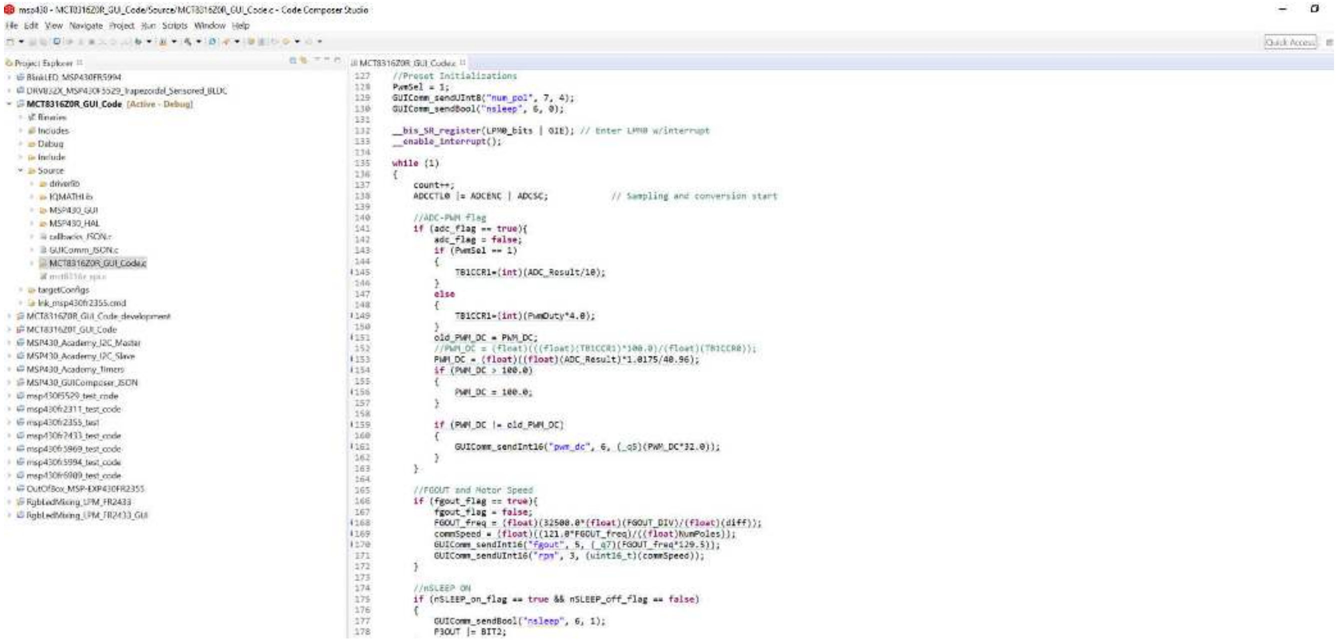
Figure 6-2. MCF8316A GUI Offline Installer

## 7 MSP430FR2355 Interface Firmware

The MSP430FR2355 on the MCF8316AEVM comes pre-programmed with the firmware necessary for communicating with the PC GUI and the MCF8316A. In order to reprogram or flash custom code on the MSP430FR2355, you will need an external MSP430 LaunchPad™that includes the eZ-FET Debug Probe. In this example, we use the MSP-EXP430FR2355 LaunchPad Development Kit to provide the debug probe. Follow the steps below to download the code for the MCF8316AEVM to use with the GUI.

### 7.1 Downloading Code Composer Studio and Importing MSP430FR2355 Interface Firmware Code

1. Extract the “MCF8316AEVM_MSP430FR2355_Firmware_GUI.zip” to a location on the computer.
2. Download the latest version of Code Composer Studio. This will set up a ti folder in the directory C:\ti.
   a. Accept all agreements, default install locations, and hit “Next” to proceed through menus.
   b. In the “Select Components” window, ensure to check “MSP430 Low-Power MCUs” to install the required packages for the MSP430 LaunchPad Evaluation Kits.
3. After installing, run CCS and select a folder or the default to use as the Workspace to store any new projects. The location and naming convention can be changed based on the user's preference. Click the OK button to accept.
4. In CCS, click on the Project tab and select “Import CCS Projects”. Click on Browse.
5. Select the “MCF8316AEVM_MSP430FR2355_Firmware_GUI” folder installed from step 1.
6. Import the project “MCF8316AEVM_MSP430FR2355_Firmware_GUI” into your workspace as shown in Figure 7-1.



```
//Preset Initializations
PwmSel = 1;
GUIComm_sendUInt8("num_pol", 7, 4);
GUIComm_sendBool("nsleep", 6, 0);

__bis_SR_register(LPM0_bits | GIE); // Enter LPM0 w/interrupt
__enable_interrupt();

while (1)
{
    count++;
    ADCCTL0 |= ADCENC | ADCSC;            // Sampling and conversion start

    //ADC-PWM flag
    if (adc_flag == true){
        adc_flag = false;
        if (PwmSel == 1)
        {
            TB1CCR1=(int)(ADC_Result/10);
        }
        else
        {
            TB1CCR1=(int)(PwmDuty*4.0);
        }
        old_PWM_DC = PWM_DC;
        //PWM_DC = (float)(((float)(TB1CCR1)*100.0)/(float)(TB1CCR0));
        PWM_DC = (float)((float)(ADC_Result)*1.0175/40.96);
        if (PWM_DC > 100.0)
        {
            PWM_DC = 100.0;
        }

        if (PWM_DC != old_PWM_DC)
        {
            GUIComm_sendInt16("pwm_dc", 6, (_q5)(PWM_DC*32.0));
        }
    }

    //FGOUT and Motor Speed
    if (fgout_flag == true){
        fgout_flag = false;
        FGOUT_freq = (float)(32500.0*(float)(FGOUT_DIV)/(float)(diff));
        commSpeed = (float)((121.0*FGOUT_freq)/((float)NumPoles));
        GUIComm_sendInt16("fgout", 5, (_q7)(FGOUT_freq*129.5));
        GUIComm_sendUInt16("rpm", 3, (uint16_t)(commSpeed));
    }

    //nSLEEP ON
    if (nSLEEP_on_flag == true && nSLEEP_off_flag == false)
    {
        GUIComm_sendBool("nsleep", 6, 1);
        P3OUT |= BIT2;
```

**Figure 7-1. MSP430FR2355 Interface Firmware Code in Code Composer Studio**

### 7.2 Using the eZ-FET to Program the MSP430FR2355

The eZ-FET Debug Probe on the MSP430FR2355 LaunchPad uses a SPI-by-Wire JTAG interface to program the MSP430FR2355 MCU on the MCF8316AEVM. Consult the MSP430 LaunchPad Development Kits for MSP430 LaunchPads that include an onboard eZ-FET Debug Probe.

1. Remove the GND, 3V3, SBWTDIO, and SBWTCK jumpers from the MSP430 LaunchPad.
2. Connect the top pins on the eZ-FET side of the LaunchPad of the GND, 3V3, SBWTCK, and SBWTDIO signals to their respective pins on J4 of the MCF8316AEVM as shown in Table 7-1 and Figure 7-2.
3. Connect a micro-USB cable to the MSP430 LaunchPad and the PC.
4. Click on the Build Project icon or “Ctrl” + B to ensure the project builds successfully. Accept any updates if needed from the Console.

5. Click on Debug Project to set up a debug session and press the Play button to run the code.
6. Stop the debug session, close Code Composer Studio, disconnect the SPI-by-Wire jumpers, and unplug the micro-USB cable from the MSP430 LaunchPad.

**Table 7-1. SPY-BI-Wire Connections Needed to Program MSP430FR2355**

| MSP430 LaunchPad™ (eZ-FET Debug Probe Side) (J101) | MCF8316AEVM 4-pin SPI-by-Wire Header (J4) |
|---|---|
| GND | GND |
| 3V3 | 3.3V |
| SBWTDIO | SBWTDIO |
| SBWTCK | SBWTCK |

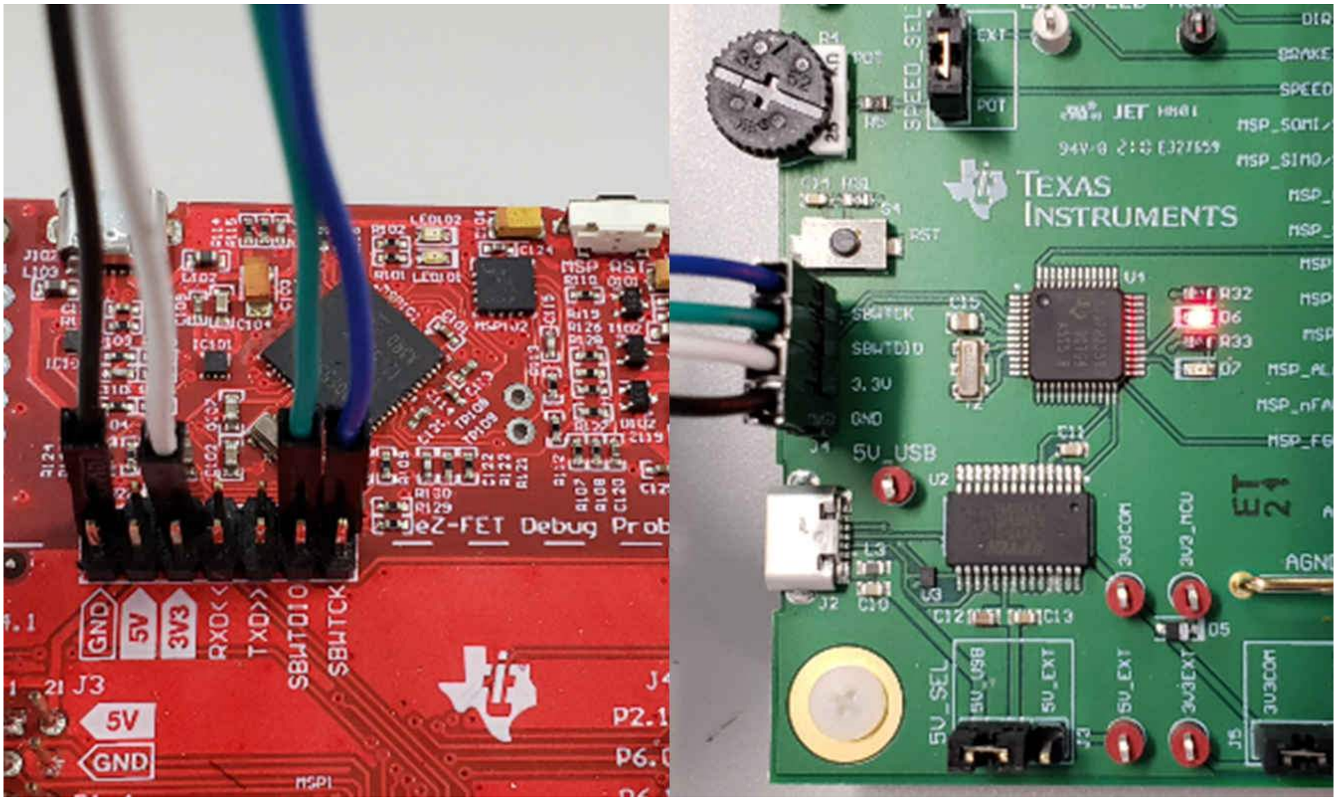


**Figure 7-2. MSP430 LaunchPad™ eZ-FET Probe Connected to MCF8316AEVM**

# 8 Schematics

## 8.1 Main Supply and Pi Filter

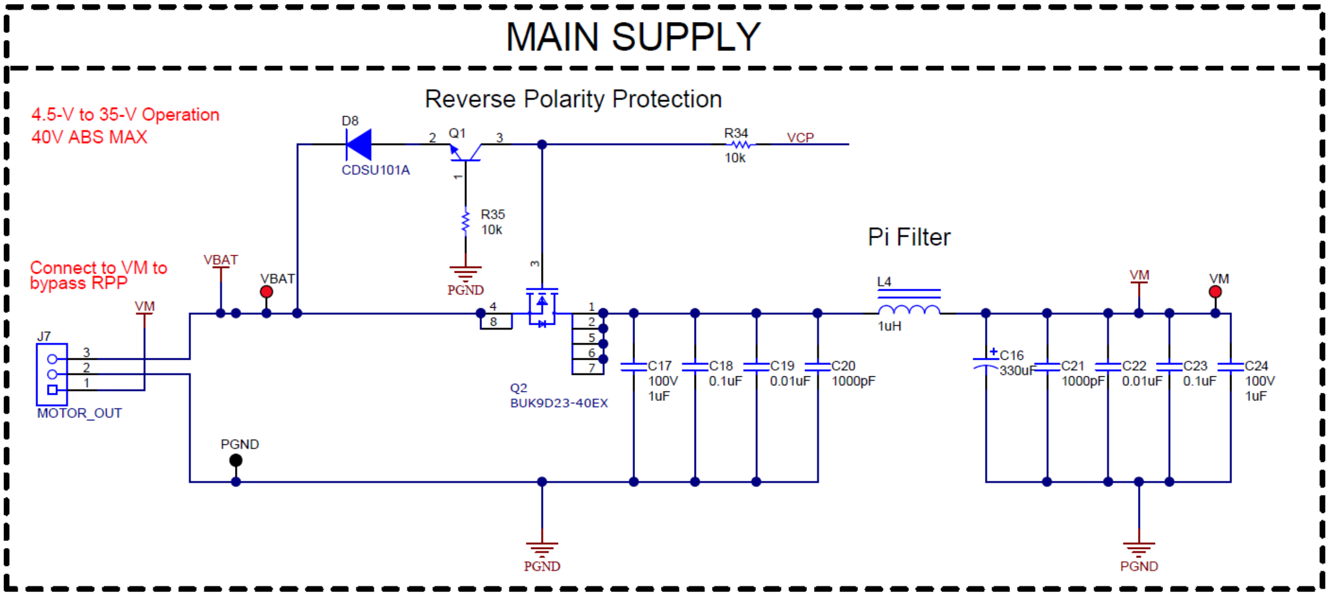


**Figure 8-1. Main Supply and Pi Filter Schematic**

## 8.2 Connectors and Interface

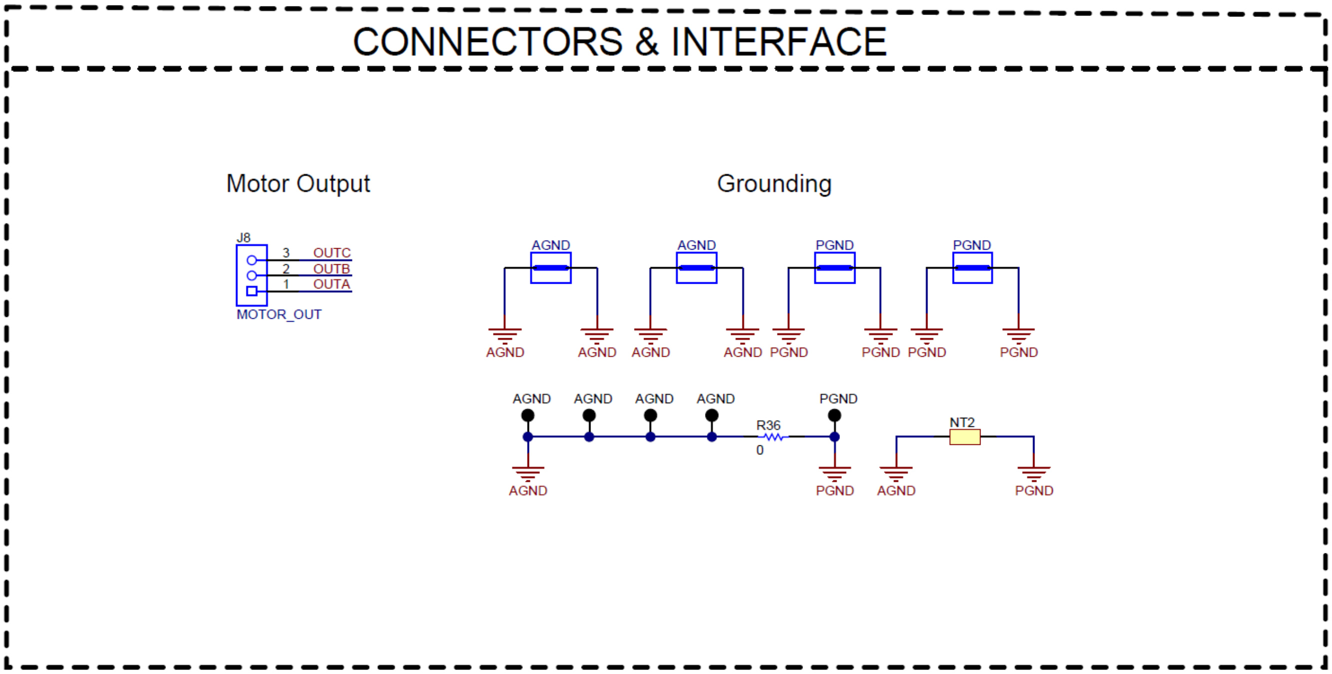


**Figure 8-2. Connectors and Interface Schematic**

## 8.3 USB to UART

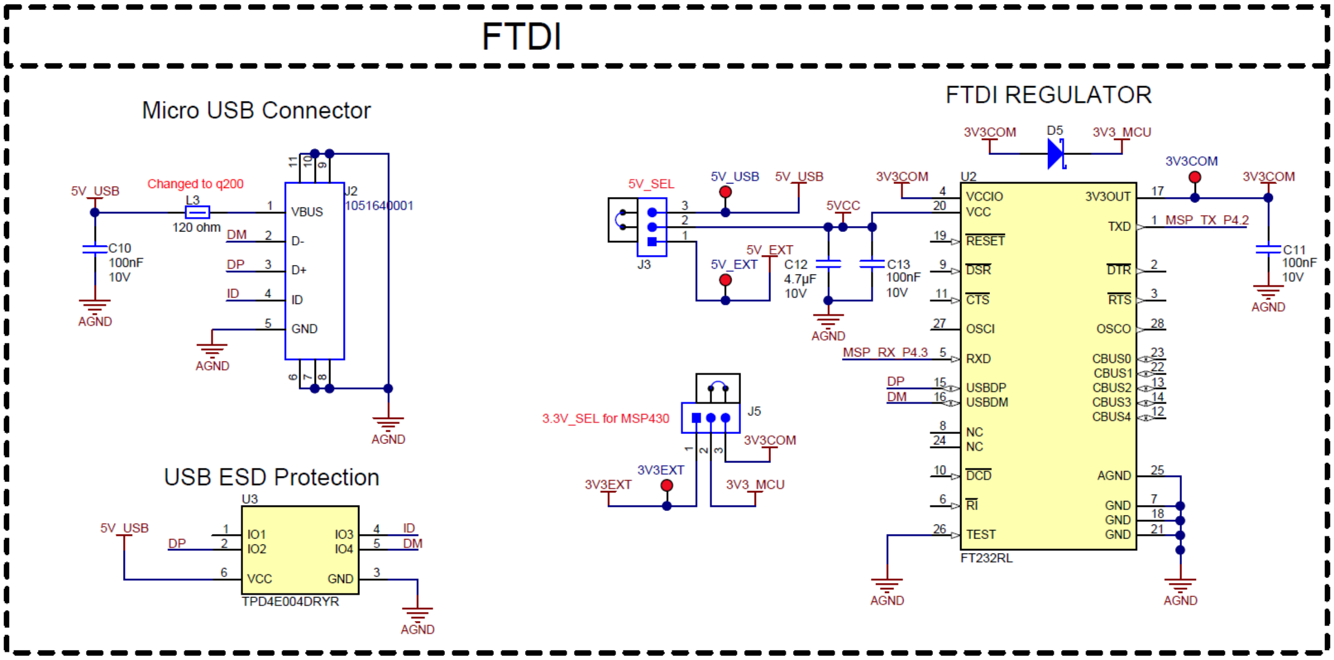


**Figure 8-3. USB to UART Schematic**

## 8.4 MCU Programming and Debug

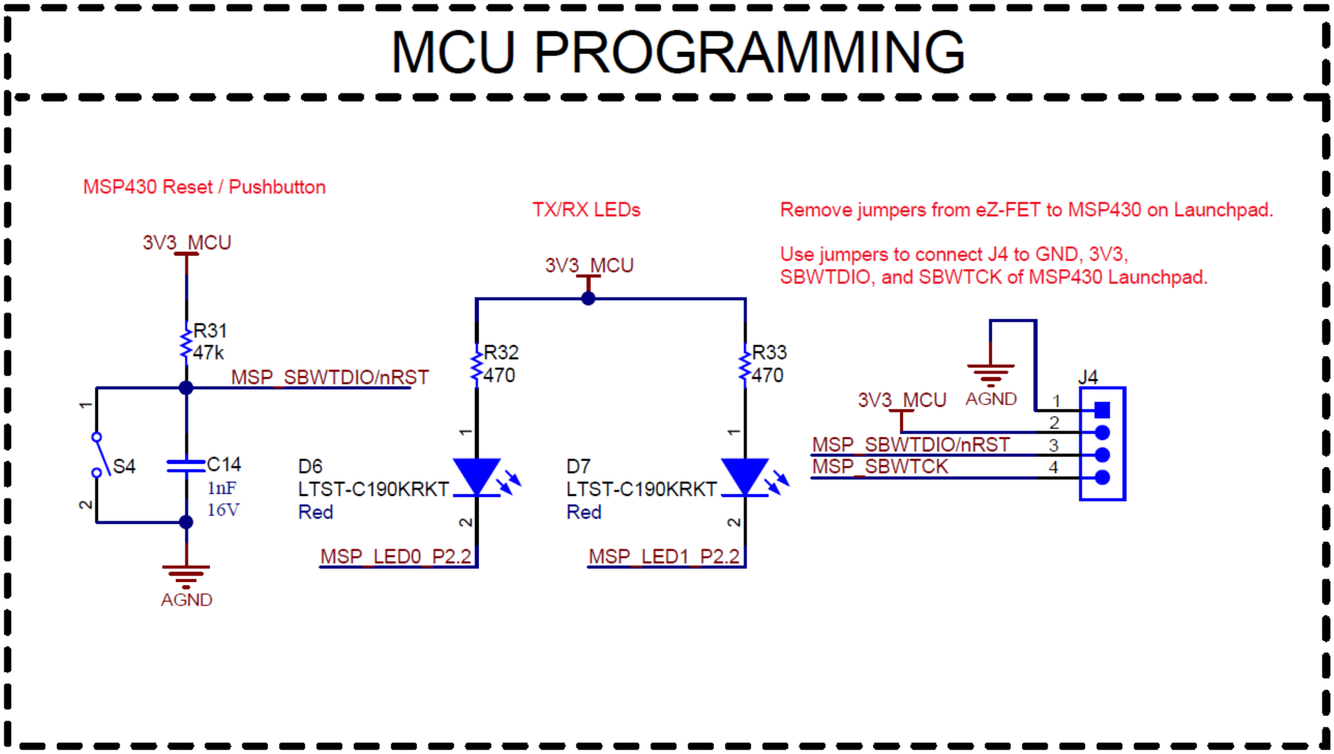


**Figure 8-4. MCU Programming and Debug Schematic**

## 8.5 MSP430FR2355 MCU

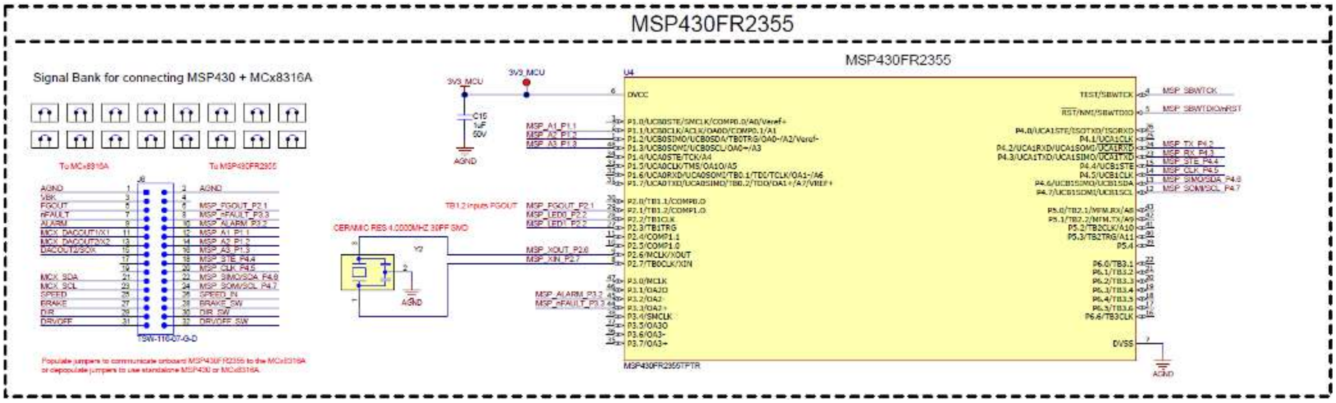


**Figure 8-5. MSP430FR2355 MCU Schematic**

## 8.6 MCF8316A 3-Phase Sensorless FOC Integrated Driver

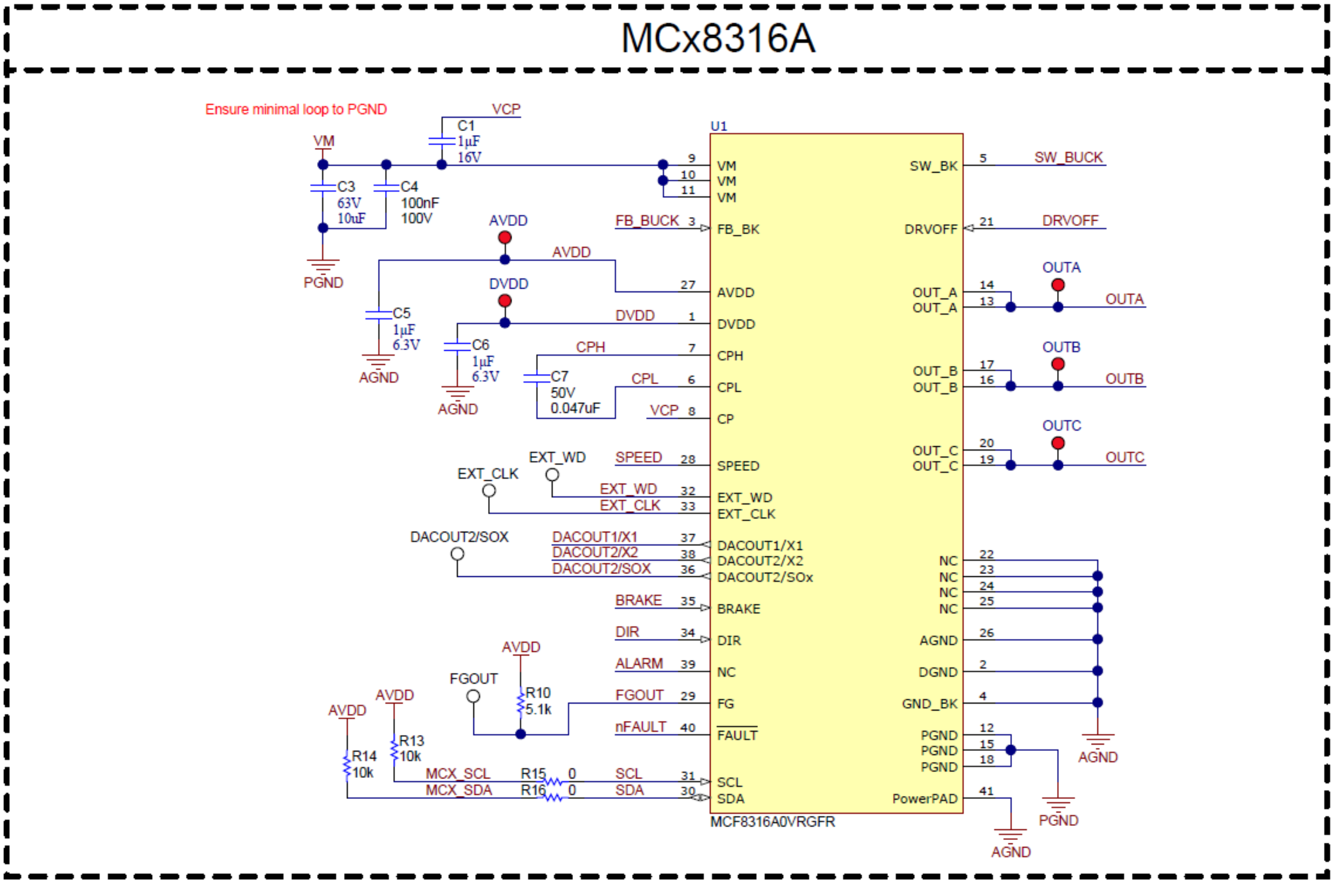


**Figure 8-6. MCF8316A 3-Phase Sensorless FOC Integrated Driver Schematic**

### 8.7 Buck Regulator

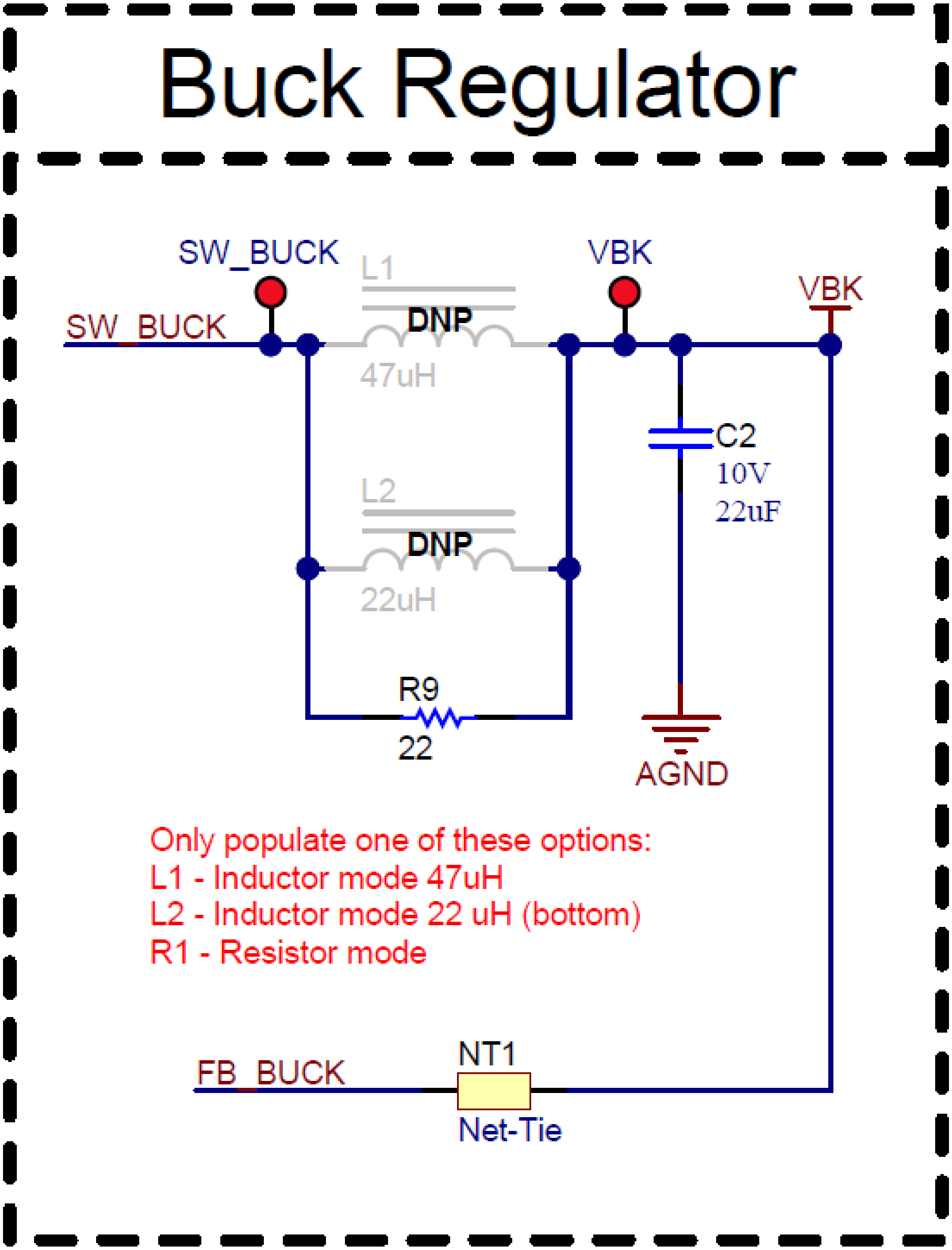


**Figure 8-7. Buck Regulator Schematic**

### 8.8 Status LEDs

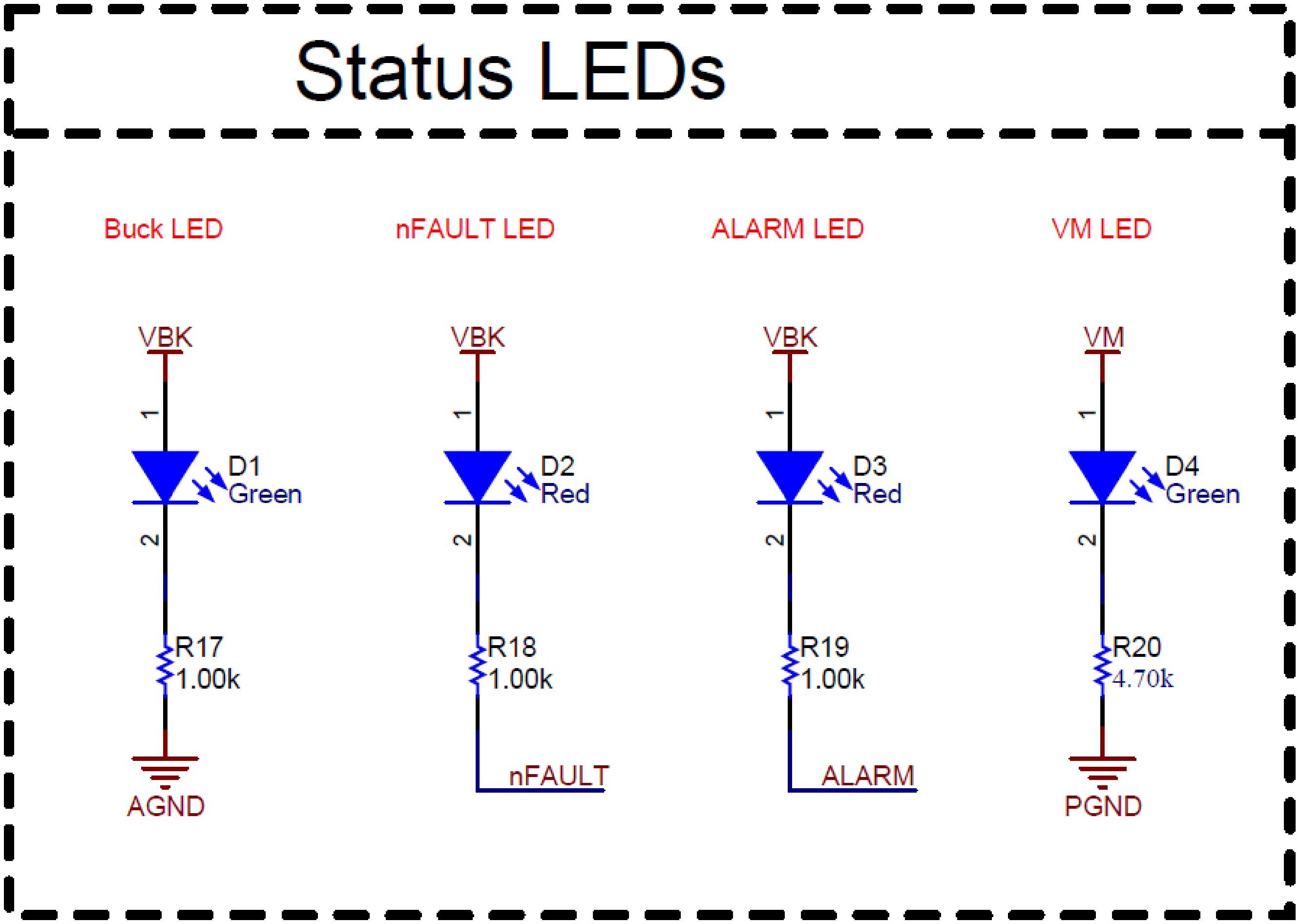


Figure 8-8. Status LEDs Schematic

### 8.9 Switches and Speed Input

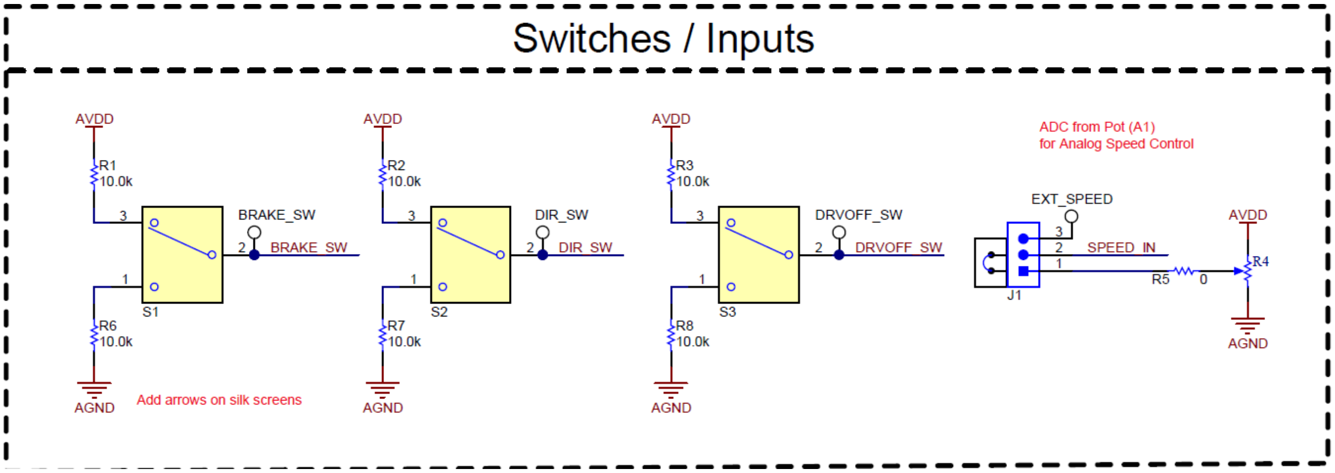


Figure 8-9. Switches and Speed Input

## 9 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

| DATE | REVISION | NOTES |
|---|---|---|
| August 2021 | * | Initial Release |

# IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265