DRV8886AT Evaluation Module User's Guide

This document is provided with the DRV8886AT customer evaluation module (EVM) as a supplement to the DRV8886AT datasheet (*DRV8886AT 2-A Stepper Motor Driver With Integrated Current Sense*). This user's guide details the hardware implementation of the EVM.

Contents

List of Figures

List of Tables

Trademarks

MSP430 is a trademark of Texas Instruments.
Windows is a registered trademark of Microsoft Corporation.
All other trademarks are the property of their respective owners.

1 Board Overview

shows the top view of the printed circuit board (PCB).

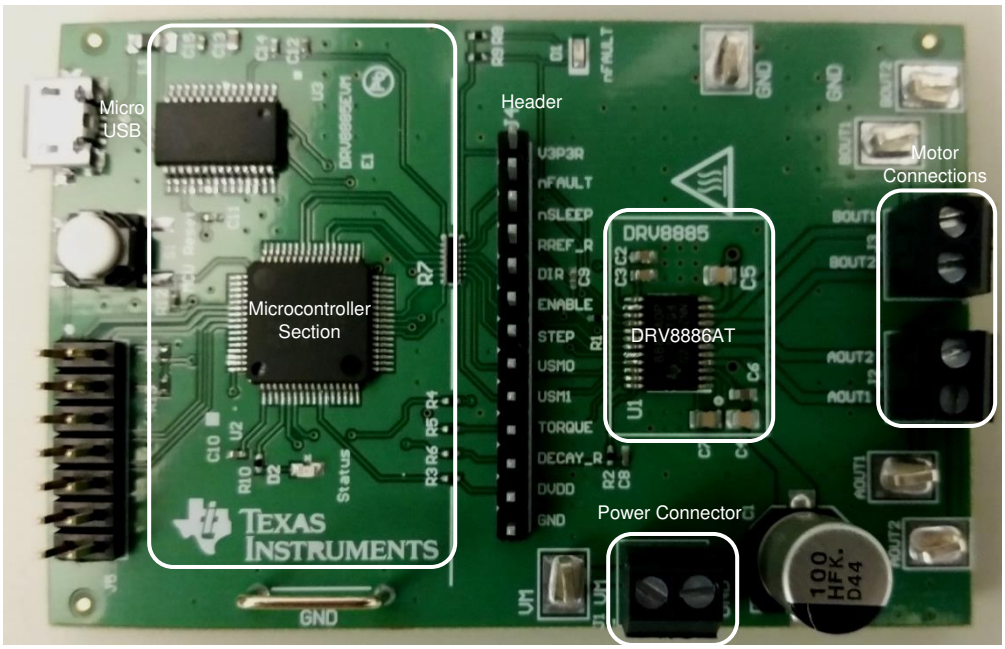


Figure 1. Top View of Typical Board Configuration (EVM Provided May Vary)

WARNING

The DRV8886AT (U1) can operate at temperatures approaching 150ºC. This device should not be touched.

2 Introduction

The DRV8886AT customer EVM is a platform revolving around the DRV8886AT, a medium-voltage dual H-bridge driver and highly-configurable power stage. This device has been optimized to drive a single bipolar stepper with up to 16 degrees of internally generated microstepping.

The EVM houses an MSP430™ microcontroller and an USB interface chip. The USB chip allows for serial communications from a PC computer where a Windows® application is used to schedule serial commands. These commands can be used to control each of the device signals, and drive the stepper motor by issuing the step commands at the desired rate.

The microcontroller firmware operates using an internal index mode.

This user's guide details the operation of the EVM, as well as the hardware configurability of the evaluation module.

2.1 Connectors

The DRV8886AT EVM offers access to the VM (motor voltage) power rail through a terminal block (J1). A set of test clips in parallel with the terminal block allows for the monitoring of the input power rail.

The user must apply the VM voltage according to recommended parameters listed in the data sheet.

NOTE: The VDD voltage for the microcontroller is derived from the microUSB connector.

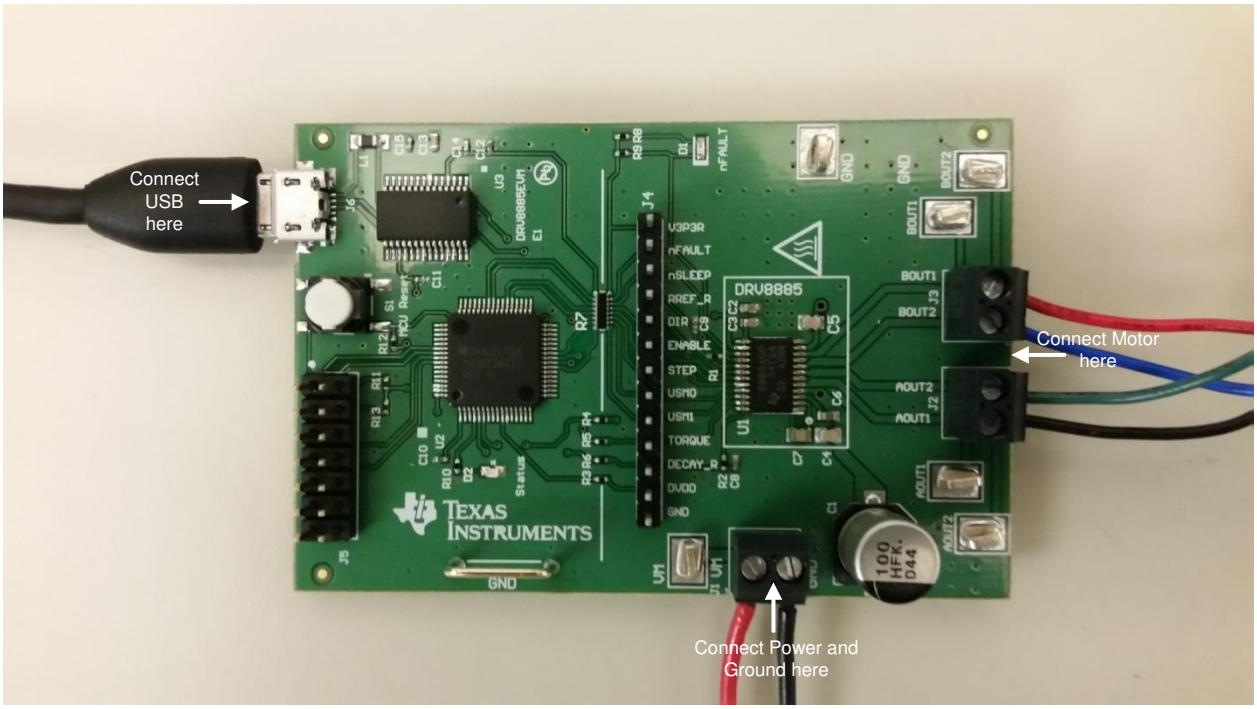


Figure 2. Connections (DRV8885 EVM Shown Has Similar Connections)

2.2 Test Points

A 0.100-inch pitch header connector (J4) provides access to every device signal in the event that a different microcontroller is used. To disconnect the internal MSP430 microcontroller, remove the R3, R4, R5, and R6 resistors, and resistor pack R7. Table 1 describes the connections available on the J4 header. Each header pin is labeled on the evaluation module, and connects to a similarly named pin of the DRV8886AT.

Table 1. Connections to DRV8886AT Using External Microcontroller

Header Label	Description
V3P3R	3.3 V after 0 Ω resistor
nFAULT	Fault output
nSLEEP	Sleep Mode input
RREF_R	Chopping current selection
DIR	DIR, Direction input
ENABLE	ENABLE, Stepper motor enable
STEP	STEP, Step input
USM0	M0, Step mode
USM1	M1, Step mode
TORQUE	TRQ, Output current scale
DECAY_R	Decay Mode select after 0-Ω resistor
DVDD	Internal supply voltage to set DAC voltage
GND	Ground

2.3 Jumpers

The DRV8886ATEVM has no jumpers.

2.4 Motor Outputs

Two motor connectors are provided. shows the avaialable J2 and J3 connectors.

2.5 Operation of the EVM

Use the steps that follow to operate the EVM:

Step 1. Install the drivers and GUI. For instructions see Appendix A.

Step 2. Connect the wires of the stepper motor to the AOUT1, AOUT2, BOUT1, and BOUT2 terminals.

Step 3. Connect the VM power supply but do not apply power at this step.

Step 4. Connect the USB cable between the PC and the EVM. When the USB is connected to the EVM, the status LED will begin to blink.

Step 5. Open the GUI. The GUI can be found in the start menu at Texas Instruments → DRV8886_EVM → DRV8886_EVM X.Y.Z, where X, Y, and Z are the revision numbers. If a shortcut was created, double-click on the shortcut to open the GUI. The GUI can take up to 30 s to establish a connection. If connection is not established, select the COM port under the *Options* menu. The BaudRate is 9600.

NOTE: The DRV8886 EVM GUI is designed to control both the DRV8886AT EVM and the DRV8886 EVM. The GUI automatically identifies the device and changes the pulldown menus as needed.

Step 6. Apply the desired voltage (6.5 to 37 V) to the VM and GND connections.

Step 7. Configure the current settings, step mode, decay mode, and torque to the desired values as shown in Figure 3.

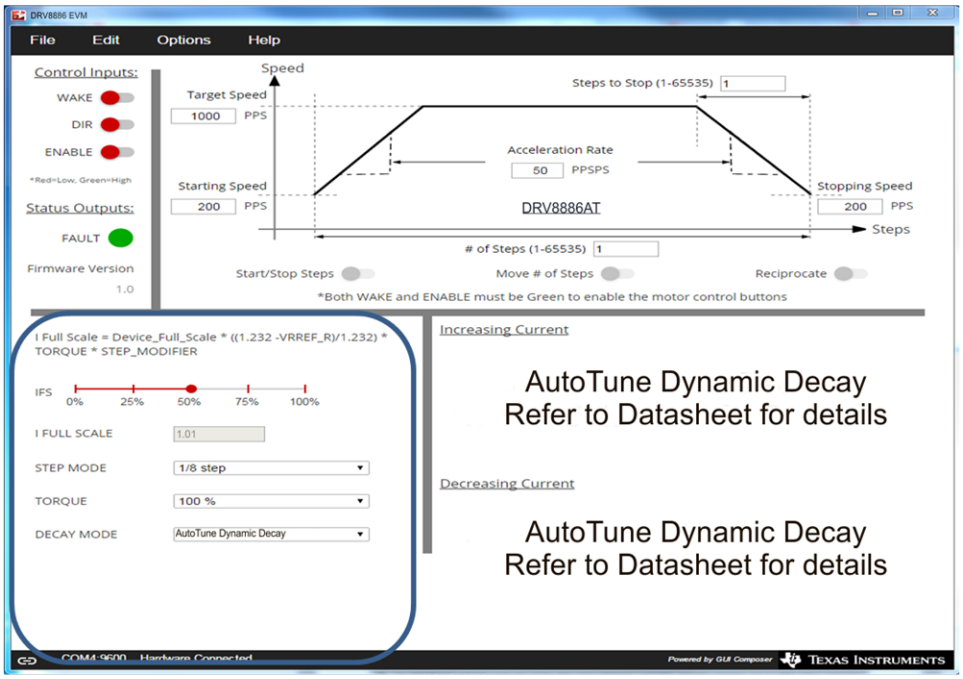


Figure 3. GUI at Startup

The current is calculated using the VREF slider, the sense resistor value, the torque setting, and the step mode setting using Equation 1.

$$I_{FS} = Device\_Full\_Scale \times \left(\frac{1.232 - VRREF\_R}{1.232} \right) \times TORQUE \times StepModifier \quad (1)$$

where

- StepModifier is 0.71 for full step and 1 for other STEP MODE settings

The number 1.232 is based on the maximum current allowed using the configuration. If VREF = 0 V, the maximum current is 1 A. The 12-bit DAC channel 1 is connected to the DRV8886AT analog input RREF through a 15-kΩ series resistor. The DAC voltage begins at 1.232 V (0%) and ends at 0 V (100%).

Step 8. After setting up the control signals for the DRV8886AT, select both the WAKE and ENABLE toggle buttons to enable the DRV8886AT. When toggled, the WAKE or ENABLE status toggles between red and green.

The WAKE toggle button, which controls the nSLEEP pin, is used to wake the DRV8886AT. The ENABLE toggle button, which controls the ENABLE pin, is used to enable the DRV8886AT outputs.

A message stating *Both WAKE and ENABLE must be green to enable the motor control buttons* is visible until both the WAKE and ENABLE toggle buttons are activated. When these two toggle buttons have been activated, the message disappears and the *Start/Stop Steps* and *Move # of Steps* toggle buttons are available.

If the WAKE or ENABLE toggle buttons are selected during motor operation, the motor is immediately stopped and the STEP control signal from the microcontroller is reset.

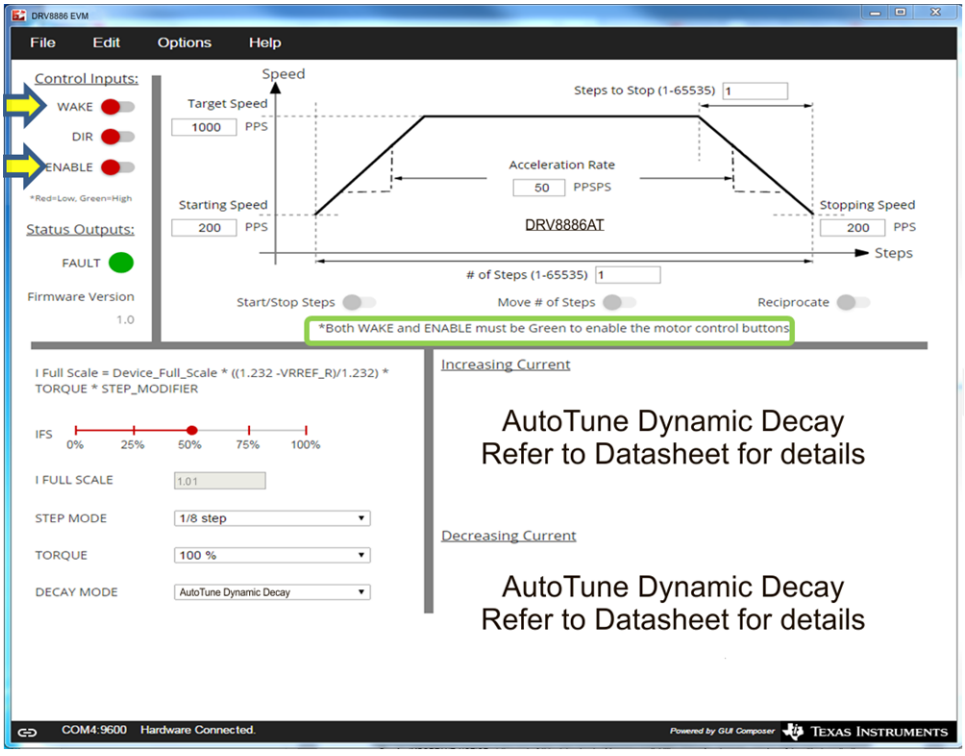


Figure 4. Wake and Enable Toggle Buttons

Step 9. The DRV8886AT EVM is now awake and can be commanded to turn the motor. Turning the motor occurs by either selecting the *Start/Stop Steps*, *Move # of Steps*, or *Reciprocate* toggle button.

The *Start/Stop Steps* toggle button is used to run the motor indefinitely. The motor will accelerate to the target speed and run until the *Start/Stop Steps* toggle button is selected. When the *Start/Stop Steps* toggle button is selected, the red button changes to green, and the *Move Steps* and *Reciprocate* toggle buttons are disabled.

The *Move Steps* toggle button is used to allow movement of an exact number of steps. When the *Move Steps* toggle button is selected, the *Move Steps* toggle button turns green, and the *Start/Stop Steps* and *Reciprocate* toggle buttons are disabled until the number of steps have completed.

The *Reciprocate* toggle button is a special case of the *Move Steps* option. When selected, the motor advances the specified number of steps in the direction initially set by the control inputs. After a short pause, the motor then advances the same number of steps in the opposite direction. This sequence is repeated until the *Reciprocate* toggle button is selected.

When the *Reciprocate* toggle button is selected, the *Reciprocate* toggle button turns green, and the *Start/Stop Steps* and *Move # of Steps* toggle buttons are disabled until the *Reciprocate* toggle button is set to red and the number of steps have completed.

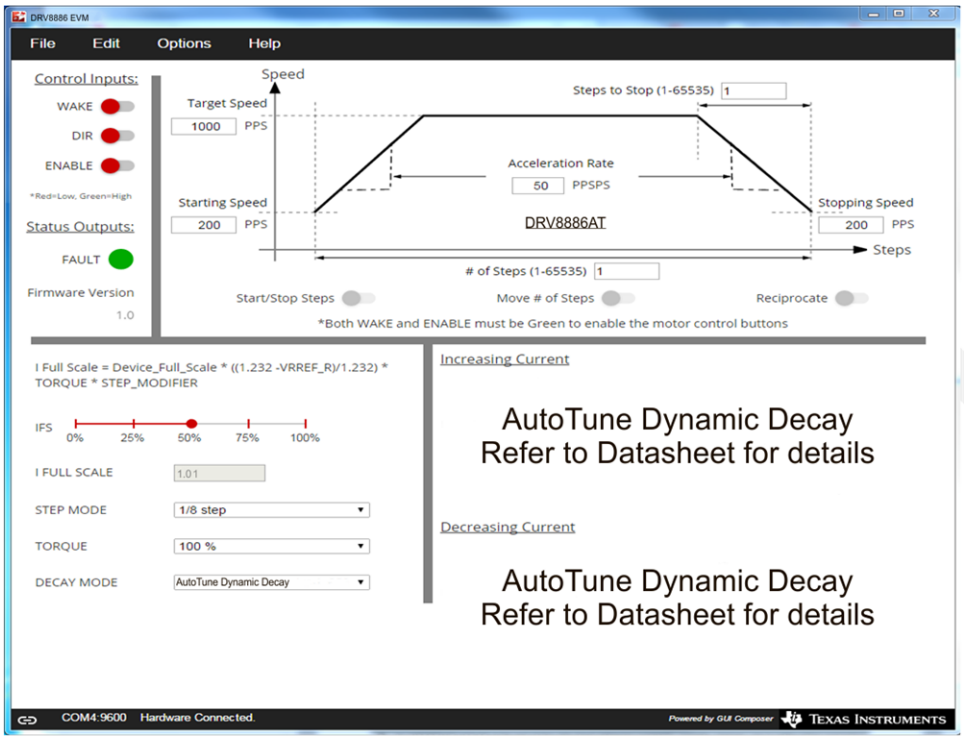
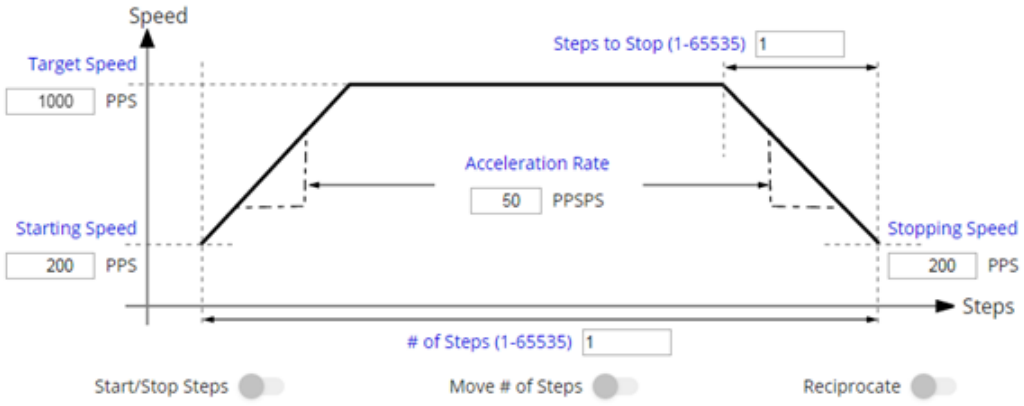


Figure 5. Reciprocate Toggle Button

As an extra precaution, the motor can be stopped by selecting either the WAKE or ENABLE toggle buttons. When selected, the STEP commands are stopped and the motor control buttons are disabled. To re-enable motor control, set the WAKE and ENABLE toggle buttons to green.

2.6 Motion Control Frame (Includes Start/Stop Steps and Move Steps)

The GUI has an area which offers access to a series of very useful stepper-control algorithms. This area allows for determining the best current settings during running at various speeds, and when holding torque is applied.



Note: Both the WAKE and ENABLE toggle buttons must be green to enable the motor control buttons.

Figure 6. Motor Motion Profile

Motor motion can only happen by using an acceleration profile which is described in Section 2.6.1. A detailed explanation of each stepper control section follows (see Section 2.6.1 and Section 2.6.2).

This frame allows the configuration and running of the stepper with the direction as specified by the *DIR* toggle button, with the current decay mode as specified under the *Decay Mode* drop-down menu, and the microstepping resolution as specified under the *STEP MODE* drop-down menu.

The *Motion Control* frame gathers user information regarding stepping rate or motor speed. An acceleration profile is used to start at a programmable speed and increase the stepping rate until reaching the programmable desired speed.

An internal 8-MHz timer is used to measure time and generate the steps on a timely manner. The GUI sends the information to the microcontroller as pulse-per-second (PPS) signal, and the microcontroller transforms it into the respective clock cycles required for the timer to generate accurate STEP pulse timing.

2.6.1 Start/Stop Steps

The acceleration profile is coded inside of the microcontroller to accept both the starting-speed PPS signal and target-speed PPS signal as a clock cycle number. When the START STEPS command is issued (the *Start/Stop Steps* toggle button is selected), the PWM timer generates steps at a rate specified by the start speed PPS parameter.

When accelerating or decelerating, the PPS signal is adjusted every 32 ms based on the integer value of PPSPS / 32 ms. If a nonzero value of PPSPS is entered, a minimum value of 1 is used. The step rate is increased by the calculated value until the target speed is reached.

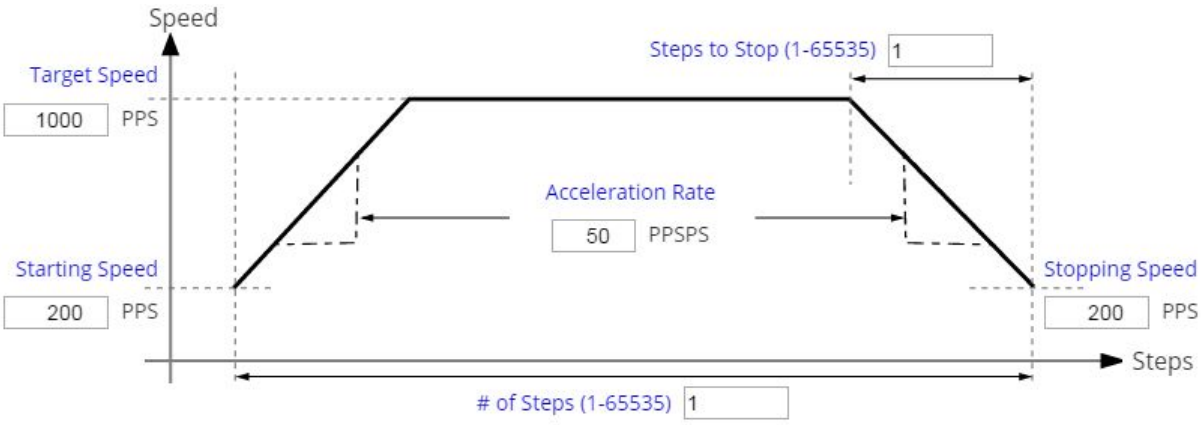
If the starting speed is greater than or equal to the target speed, the acceleration rate is set to 0. This setting will prevent further speed changes until the motor is stopped, and new starting and target speeds are entered. Texas Instruments recommends setting the starting speed to ½ the target speed to avoid this scenario.

The very same START STEPS command computes how frequently automatic speed updates are issued and a second timer is used to change the speed according to the programmed acceleration rate profile.

When the target speed PPS value is reached, the acceleration profile ends and the motor continues running until the STOP STEPPER command is issued (the *Start/Stop Steps* toggle button is selected again). When the stepper is commanded to stop, the controller performs the same actions as it did while accelerating, but in reverse order to decelerate until the stop speed PPS value is reached, in which case the motor fully stops.

A second motor actuation is provided by the *Move # of Steps* and *Reciprocate* commands in which a programmed number of steps are issued and then the motor is stopped. The acceleration and deceleration profiles work similarly as before, except when the deceleration starts and when the motor actually stops are a function of the *Steps to Stop* and deceleration rate parameters.

Figure 7 shows the acceleration profile and the role each parameter plays during speed computation.



Note: Both the WAKE and ENABLE toggle buttons must be green to enable the motor control buttons.

Figure 7. Acceleration Profile

2.6.2 Move Steps

To move the stepper a certain number of steps, use the move steps function to easily perform this action. Parameters from the other frames are reused and their use is as previously explained. Two new parameters have been added to properly control the limited number of steps actuation. These parameters are described as follows:

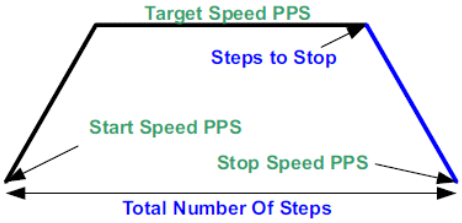
Number of Steps — This parameter is the number of steps that the controller will issue.

Steps to Stop — The controller is continuously monitoring the step being issued and when the current step is equal to the *Steps to Stop* parameter, a deceleration profile is issued. If the value of the *Steps to Stop* parameter is larger than the number of steps, then the motor stops abruptly and without undergoing a deceleration profile.

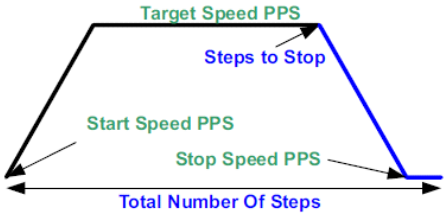
When a deceleration profile is issued, the controller decreases the speed until reaching the value of the *Stopping Speed* parameter. If the *# of Steps* parameter is met before the deceleration profile is complete, then the motor stops at the current speed. If the stop speed is met before all the number of steps is issued, then the motor rotates at the stop speed value until all the steps are executed.

Ideally, the system should be tuned to resemble the case in which the controller executes all the commanded steps at a speed as close as possible to the stop speed. In the event this is not possible because of the particular parameters that were selected, stopping the motor at a speed very close to the *Stopping Speed* parameter is often good enough to ensure good motion quality and application performance.

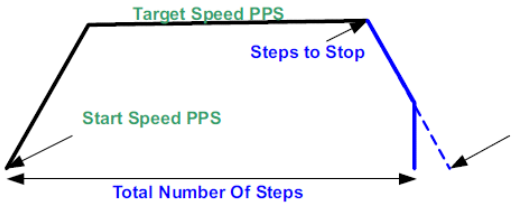
Figure 8 shows the three conditions possible when stopping and the action taken.



1. Motor reaches stop speed at the stop speed.



2. Motor reaches stop speed before the stop speed is reached.



3. Motor runs out of steps before reaching stop speed.

Figure 8. Stop Conditions

2.7 EVM documentation

The EVM schematics, layout, and BOM are provided in the DRV8886ATEVM Hardware Files. The GUI, USB drivers, and MSP430F2617 source code are provided in the DRV8886ATEVM Firmware and GUI software file.

Driver and GUI Installation Instructions

A.1 Installing the FTDI Driver

In many cases, connecting the EVM to the computer will automatically install the FTDI driver. If necessary, download the driver from the DRV8886ATEVM Firmware and GUI. Unzip the file and install the USB driver following these steps based on the operating system:

- If using Windows XP, run \USB driver*CDM v2.10.00 WHQL Certified.exe*.
- If using Windows 7, go to the \USB driver\ folder and right-click the *CDM v2.10.00 WHQL Certified.exe* file. Then select *Properties*, go to the *Compatibility* tab, check the *Run this program in compatibility mode for* option, select the *Windows XP (Service Pack 2)* option, and click the *OK* button. Next run the *CDM v2.10.00 WHQL Certified.exe* file and click the *Yes* button in the pop-up window.

A.2 Installing the GUI

NOTE: The DRV8886 EVM GUI is designed to control both the DRV8886AT EVM and the DRV8886 EVM. The GUI automatically identifies the device and changes the pulldown menus as needed. If the DRV8886 EVM GUI has been previously installed, installing the GUI again is not required.

Locate the *DRV8886_EVM_installer.zip* file in the GUI folder. Unzip the file to any location and then double click the *GUIComposerApp-1.0.0.setup-win_2.0.6.exe* file in the unzipped folder.

The installer will begin and the window shown in Figure 9 will appear.

Click the *Yes* button to continue (see Figure 9).

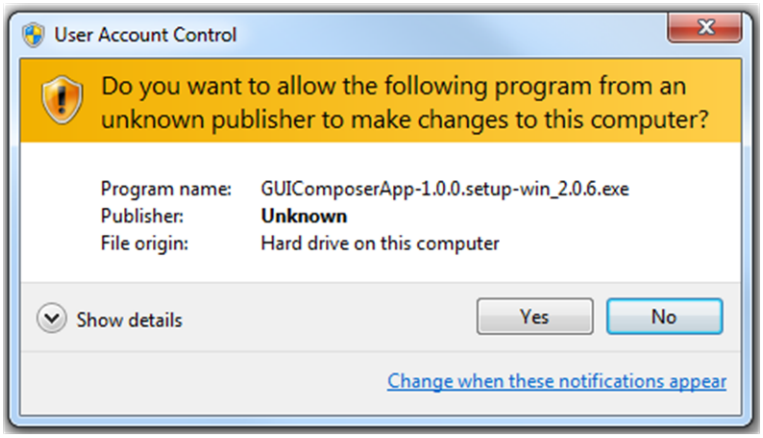


Figure 9. User Account Control Window

Click the *Next* button to continue (see Figure 10).

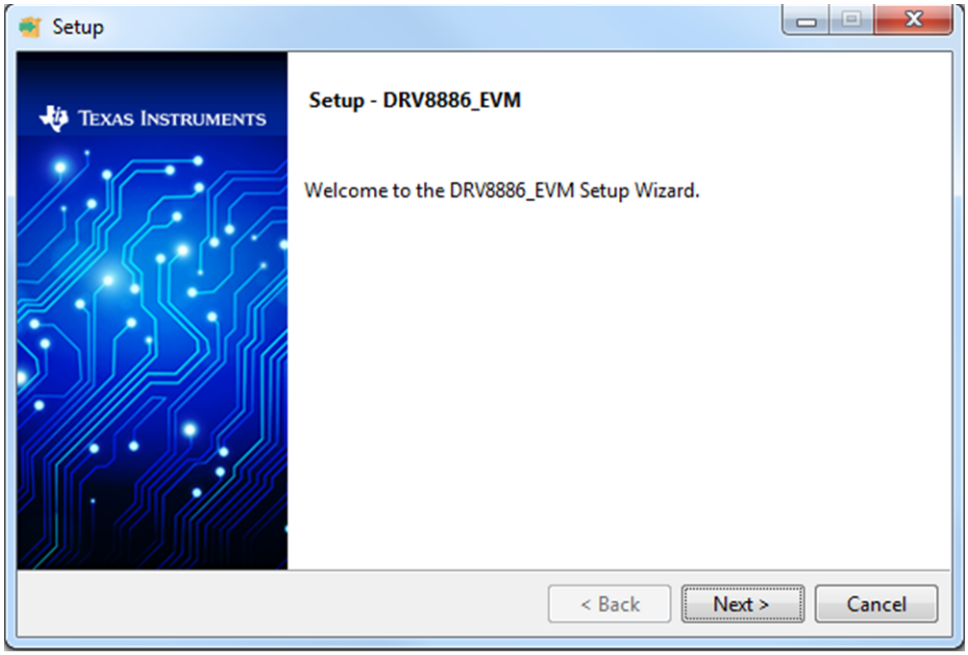


Figure 10. EVM Setup Wizard

Click the *I accept the agreement* radio button and click the *Next* button to continue (see Figure 11).

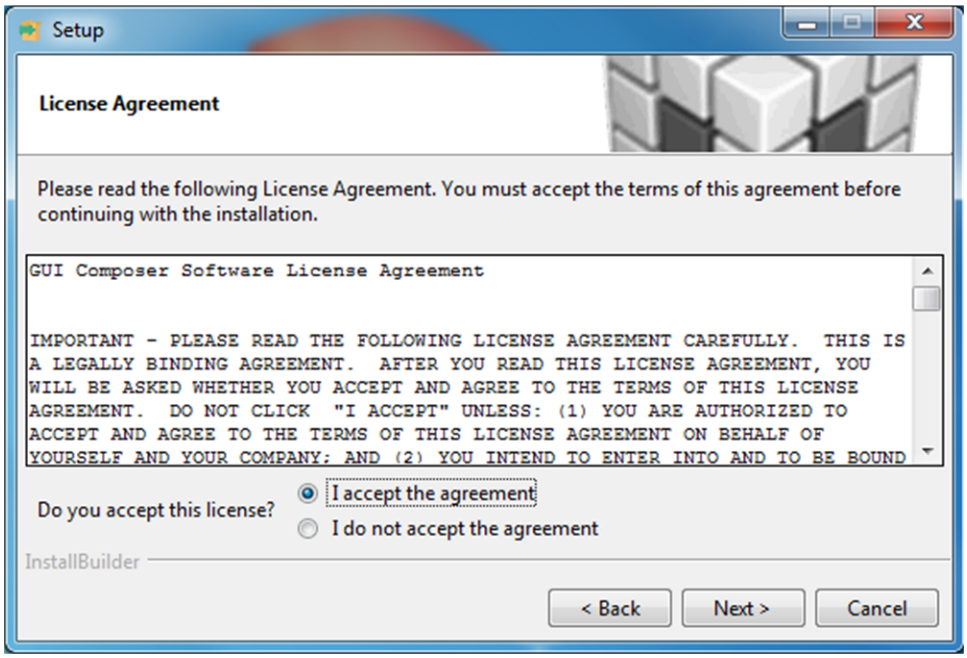


Figure 11. License Agreement

Click the *Next* button to continue (see Figure 12).

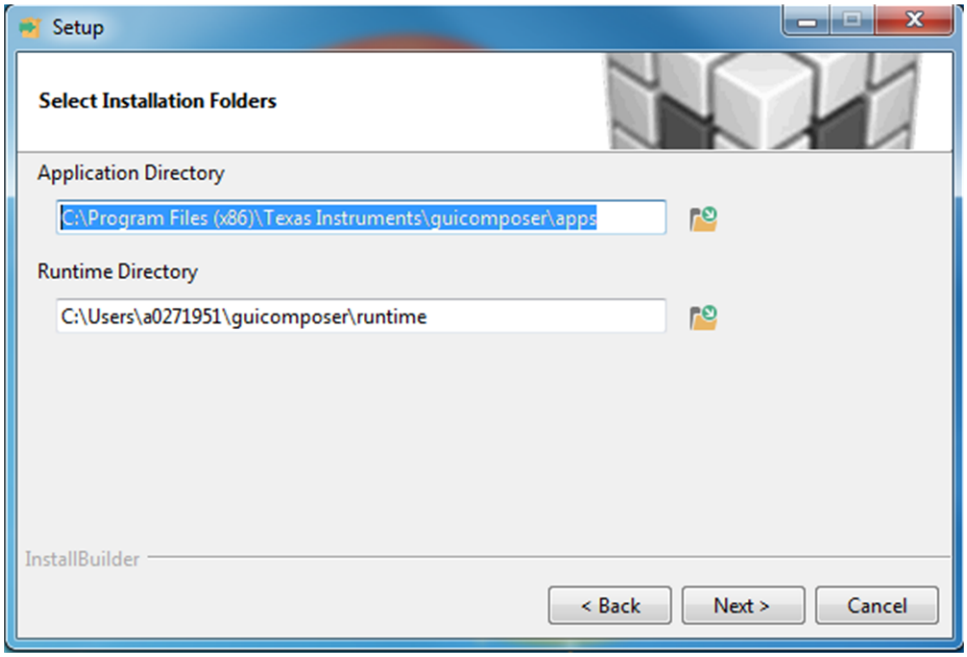


Figure 12. Installation Folders

At this point, a few options may appear. If the GUI Composer Runtime v.2.0.6 has not been previously installed, select *Install from File* radio button and download the runtime from the following link, GC Runtime v2.0.6. Click the Search button to the right of the text box next to *Install from File* and select the downloaded runtime v.2.0.6 file. Click *Next* to continue.

If the GUI has been previously installed a message similar to the one in Figure 13 appears. If this message appears, click the *Yes* button and then click the *Next* button to continue (see Figure 13).

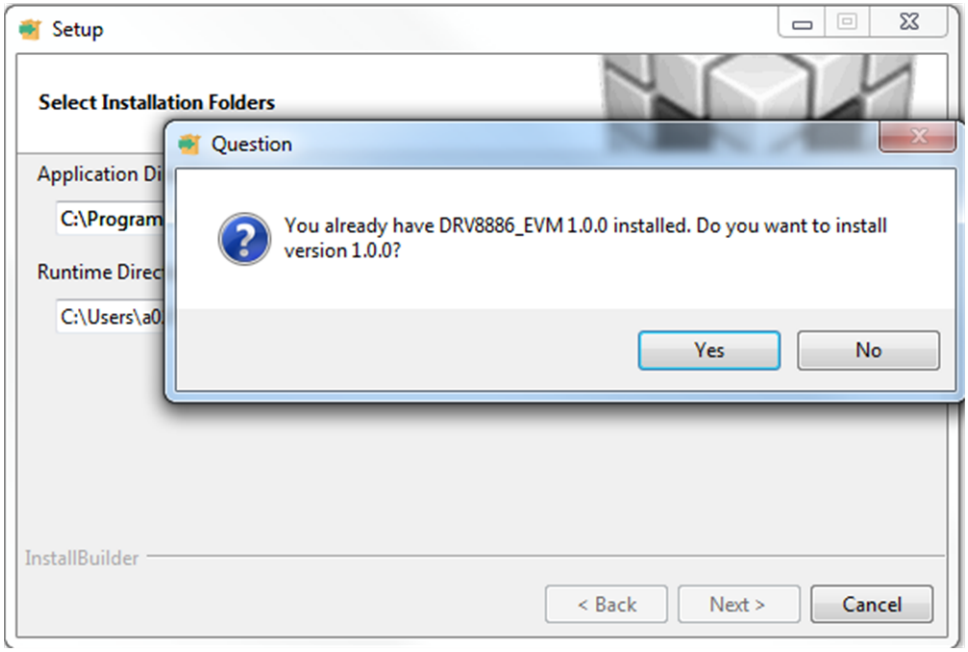


Figure 13. Possible Upgrade Question

Click the *Next* button to continue (see Figure 14).

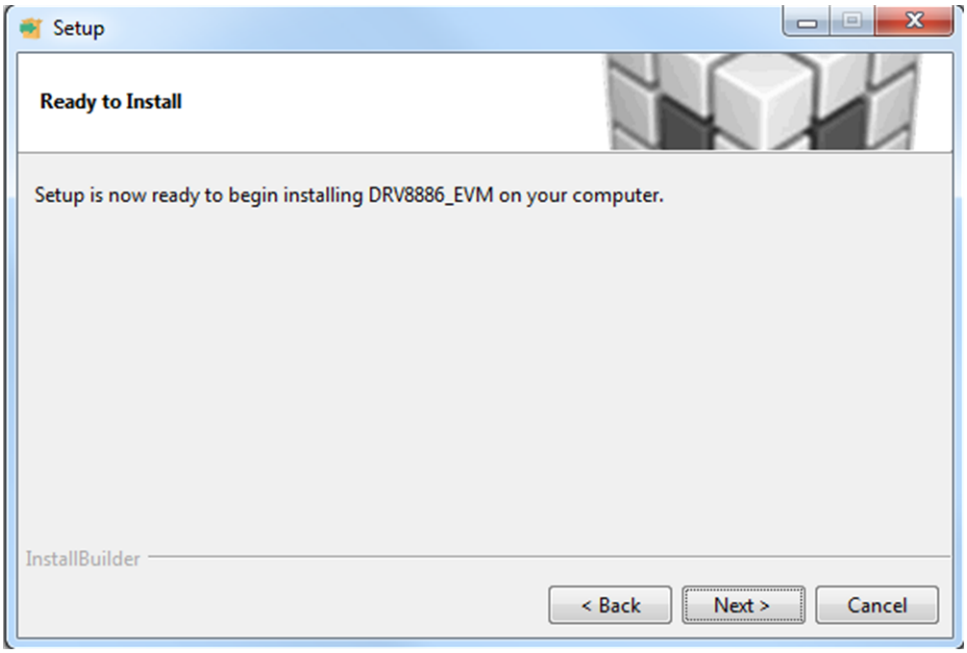


Figure 14. Ready to Install

Click the check box next to the desired results and then click the *Finish* button to complete the setup wizard (see Figure 15).

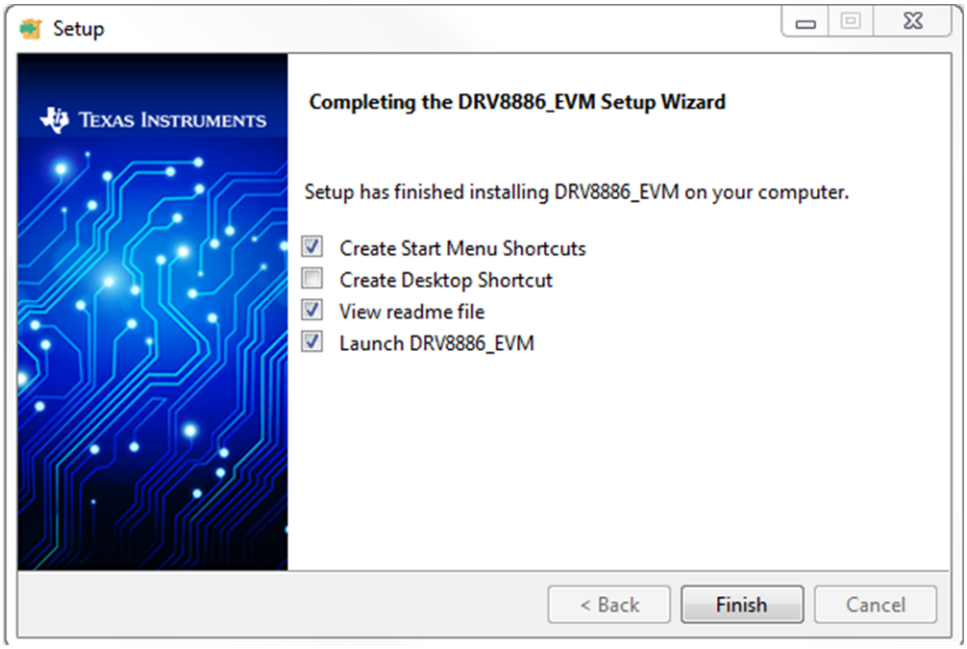


Figure 15. Completed

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), 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, 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 (www.ti.com/legal/termsofsale.html) 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.

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