LVHB_K64F_1751xEVB_Brush

Example project for Low Voltage H-Bridge SW Driver

Date: *25/05/2017*

Revision: *1.0*

Overview

The purpose of this example project is to demonstrate how to control a DC brushed motor using single H-Bridge device and the Low Voltage H-Bridge (LVHB) SW Driver. The project contains several cases to show how to use most of the driver functions related to DC brushed motor control.

Hardware Requirements

Following is required:

- FRDM-K64F (MCU freedom board)
- FRDM-17510EVB or FRDM-17511EVB (H-Bridge freedom board)
- DC Motor (supply voltage according to LVHB device operation voltage range)
- External Power Source (according to DC motor supply voltage)
- USB Micro B cable

Setting up Hardware

Target platform for this example is FRDM-K64F as MCU board and FRDM-17510EVB or FRDM-17511EVB as H-Bridge device board. Note that the driver supports also other LVHB devices and other MCUs. MCUs supported by SDK 2.x can be found in a roadmap on the NXP community. For more information about supported devices refer to LVHB SW driver user guide.

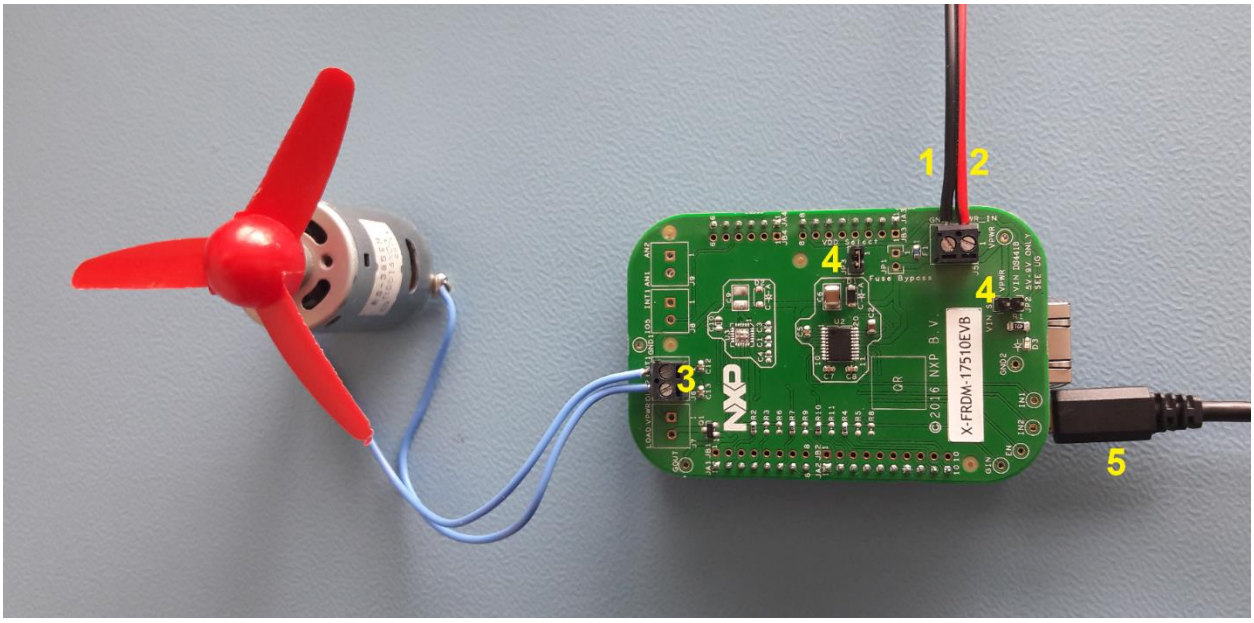

Figure 1. HW connection of FRDM-17510-EJ-EVB

In Figure 1 you can see HW connection and jumper configuration of FRDM-17510EVB freedom board with load. Jumper JP3 must be short and JP2 open. Description of HW connection is in Table 1. HW connection of FRDM-17511EVB is similar.

Table 1. Legend for HW connection

Label	Description
1.	DC Power supply (GND)
2.	DC Power supply (+)
3.	DC brushed motor connection
4.	Jumper configuration
5.	USB Micro

Setting up Software

Make sure that you have installed KDS 3.2.0 or newer.

The application uses debug interface with virtual serial port to print user messages. Check that your debug connection has been set up properly. Type of used debug connection depends on used MCU. FRDM-K64F uses **J-Link**, see Figure 2. Note that number of COM port may differ because of different system resource usage. Baud rate is 115200 Bd.

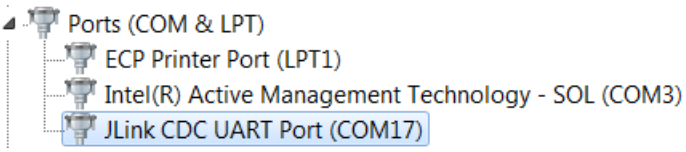

Figure 2. J-Link virtual port

Description

The purpose of this example project is to show how to control a DC brushed motor using single H-Bridge device. The project contains several cases to show how to use different driver functions.

The project uses the following peripherals:

- FTM0 – IN1 and IN2 pins.
- GPIO – GIN and EN pins.
- UART0 – Print info to serial COM console.

Pin selection for all mentioned peripherals follows in Table 2 for selected MCU.

Table 2. Pin selection

Pin Function	FRDM-K64F
IN1	PTA2/FTM0_CH7
IN2	PTC2/FTM0_CH1

EN	PTC17
GIN	PTE25
UART RX	PTB16
UART TX	PTB17

Application uses virtual serial port to print user messages that describe executed test cases. Serial port settings are following:

- Data width: 8 bits
- Baud rate: 115 200 Bd
- Parity: none

In module *main.c* the board hardware is initialized. Then the configuration structure of LVHB driver is filled in. According to the configuration structure utilized timer periphery and GPIO pins are initialized.

IN1 and IN2 pin are configured as timer outputs. It enables speed adjustment of the motor in forward and reverse direction.

The project consists of several test cases:

1. Setting H-Bridge output to tri-state – the motor is running when function *LVHB_SetTriState* is called. You can see that motor slowly stops. Motor brake (H-Bridge outputs to LOW) follows so you can compare immediate and slow stopping.
2. Disabling H-Bridge device – the motor is running when H-Bridge device is disabled using function *LVHB_SetMode*. Motor stops immediately. Then the device mode is set back to normal. The method uses EN pin.
3. Control of GOUT H-Bridge output – value of GOUT output is initialized to LOW by default. In this test case function *LVHB_SetGateDriver* sets the output to HIGH and after a while back to LOW state. The other H-Bridge outputs are not influenced.
4. Control of motor speed – PWM duty is increasing to reach value 100% (the highest speed). Then the duty is decreased to 0%. Motor direction is reversed and the procedure described above is repeated. Function *LVHB_RotateProportional* is used in this test case.

In *main.c* following set of functions is implemented covering LVHB SW driver functionality:

- *GetDefaultConfig* – Fills the driver configuration by default values.
- *ConfigureGpio* – Configures GPIO for usage with the driver.
- *ConfigureTimer* – Configures timer for usage with the driver.
- *Init* – Initializes the device.
- *RotateProportional* – Spins the motor in desired direction at PWM duty speed.
- *SetDirection* – Sets direction of brush motor movement.
- *SetMode* – Sets H-Bridge device mode using enable pin.
- *SetTriState* – Sets output of H-Bridge to tri-state (high impedance).

- *SetGateDriver* – Sets device gate driver output.

Import the Example Project

The following steps show how to import an example project into KDS 3.2.0.

1. In KDS click on the ***File / Import***.
2. Choose ***General / Existing Projects into Workspace***.
3. Click ***Browse to select root directory*** with your downloaded example projects.
4. ***Select project*** named **LVHB_K64F_1751xEVB_Brush** and click ***Finish*** to complete the process.
5. **Find *source* folder** of imported project in Project Explorer window. It contains two main files.
6. Rename either ***main.c_FRDM-17510EVB*** or ***main.c_FRDM-17511EVB*** to ***main.c*** according to used H-Bridge freedom board.
7. Now the example project should be in your workspace and ready to run.

Building and Running the Project

In order to build and run the project you need to ***build*** the project usual way. If the build is successful, ***debug and run*** the project. This can be accomplished in following steps:

1. Click on the **arrow** next to the **debug icon** and select **Debug Configurations**.
2. **Select** one of the existing configurations with **project name** under **SEGGER J-Link** group or **create** one by double clicking on this group.
3. Apply changes and click on **Debug**.

If you have any questions related to how to work with debug configurations, see ***Kinetis Design Studio User's Guide***.