# LV8702VSLDGEVK

# Stepper Motor Driver Module Solution Kit Quick Start Guide

**ON Semiconductor®**

www.onsemi.com

## EVAL BOARD USER'S MANUAL

### Overview

The LV8702VSLDGEVB is an **ON Semiconductor** motor driver module featuring the **LV8702V.** This module is capable of easily driving a **stepper motor.**

Motor driving is made easy with Arduino Micro[1] compatibility. The LV8702VSLDGEVB comes with a Baseboard for facilitated plug−and−play connectivity with an Arduino Micro.

GUI and Open−source API Functions are available for custom, user−specific motor driving programs.

### Features

- $VCC_{max}$ = 36 V, $IO_{max}$ = 2.5 A[2]
  (between OUT_A and B, between OUT_C and D)
- **24 V Applications** Recommended (VCC = 9~32 V Max)
- Rotation of the Motor is **Controlled by a Single Pulse Signal**, which is **Advantageous for Smooth Driving (Micro−step) and Rotating at High Speed Rotation**
- 4 Patterns can be Set for Full, Half (2 Patterns) and Quarter Step
- **High−efficiency Drive that Adjusts the Current to the Optimum Current** according to the Motor Load and Rotational Speed (in Half, Quarter Step Setting)
- **The Reduction of Power Consumption, Heat Generation, Vibration and Noise** is Achieved
- User−friendly **GUI Allows Operation Check and Sample Program Generation**

For further product information, please visit:
http://www.onsemi.com/PowerSolutions/product.do?id=LV8702V

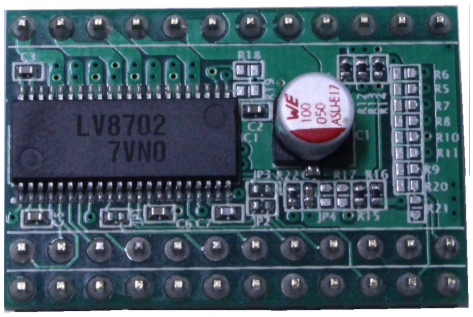



**Figure 1. LV8702VSLDGEVB Board**



**Figure 2. ONBB4AMGEVB Board with Arduino Micro and LV8702VSLDGEVB Board**

1) Arduino/Genuino are trademarks registered by Arduino AG.
2) Stress exceeding this voltage or current may risk damage to the device. Functionality and reliability may be hindered if the device exceeds these values. The maximum current value, IOmax, does not guarantee the module can handle that much current. If the device overheats, thermal shutdown will occur.

© Semiconductor Components Industries, LLC, 2018

## WHAT YOU NEED

- PC
  Recommended OS: Windows 7/Windows 10, 64 bit
  **Connected to the Internet**
  **(Only for "Including the TimerOne Libary" on Page 3)**
  Arduino IDE 1.8.4 [3] Installed
  User with an Administrator rights
- Power Supply
  AC Adapter Output voltage: 9~32 V
  Output current: ~5 A
  Connector specification
  Polarization: Positive Center
  Inner diameter = 2.1 mm, outer diameter = 5.5 mm
  Recommended: WSU120−1000 (Triad Magnetics)
  or stabilized DC power supply, dry cell battery, etc.

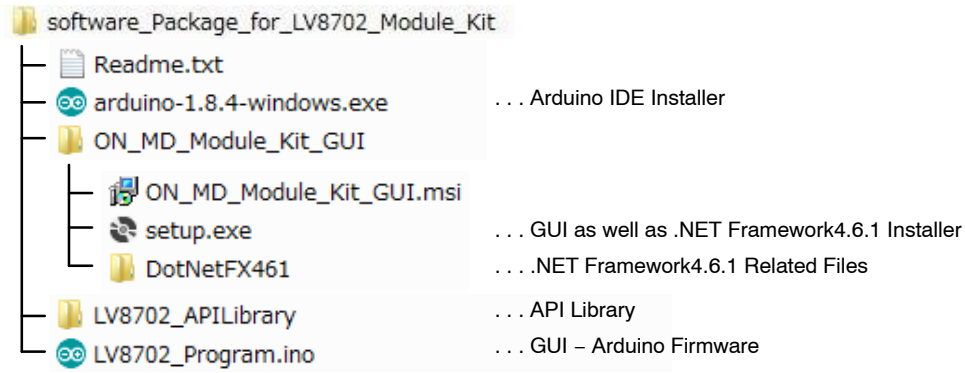
## CONTENTS OF THE KIT

### Hardware

- LV8702VSLDGEVB: Motor driver module
- ONBB4AMGEVB: Baseboard
- Arduino Micro
- USB Cable (Micro B−A)
- Flat−tip Screw Driver
- Stepper motor MDP−35A
  (Nidec Seimitsu, Step angle= 7.5 deg, 12 V/ 300 mA) 1 pc

### Software

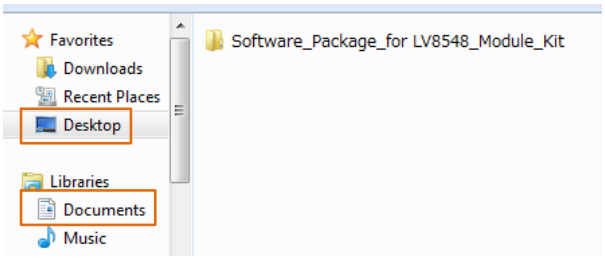
All of the following files are included in LV8702VSLDGEVK Software (English) that is available on the website of the following links.
http://www.onsemi.com/PowerSolutions/evalBoard.do?id=LV8702VSLDGEVK




[3] Contents of kit may not be compatible with different versions. Please see the Appendix attached if specified version is not installed

### Software Setup

Please download LV8702VSLDGEVK_SOFTWARE.ZIP and expand software_Package_for_LV8702_Module_Kit to any directories (E.g. Libraries\Documents or Desktop).

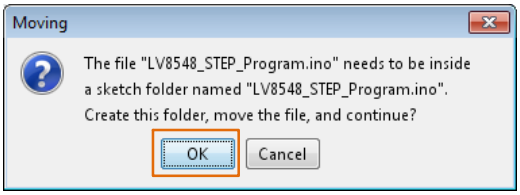



### API Library

The following process should be operated with the Arduino Micro disconnected.

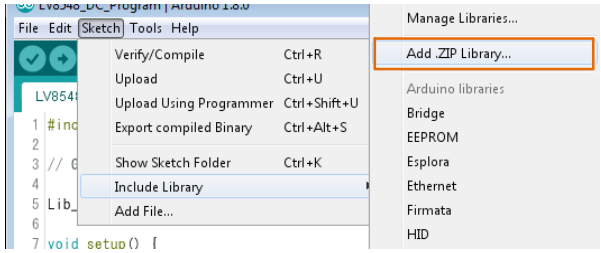
1. Double click on the Arduino sketch for GUI
   LV8702_Program.ino
   This will launch the Arduino IDE.
   (Check the Appendix, "Arduino IDE Installation", if the Arduino IDE is not installed)

   The IDE will display the following message when the LV8702_Program.ino file is opened for the first time. Click "OK" to continue.

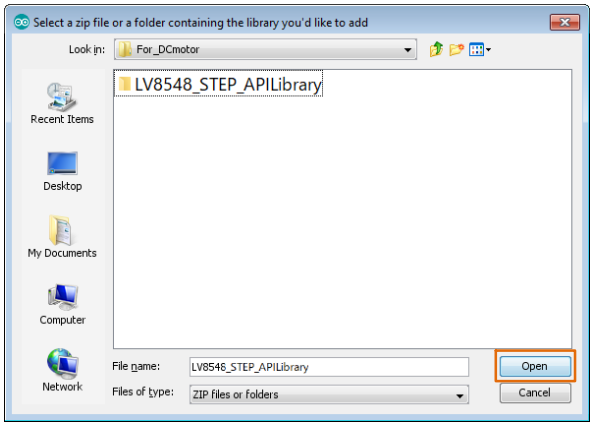



   This message will be displayed if the .ino file is not inside of a folder with the same name.
   (E.g. Sketch12345.ino must be in the Sketch12345 folder).

2. Include the API Library
   As shown below, navigate to "Sketch → Include Library → Add .Zip Library"



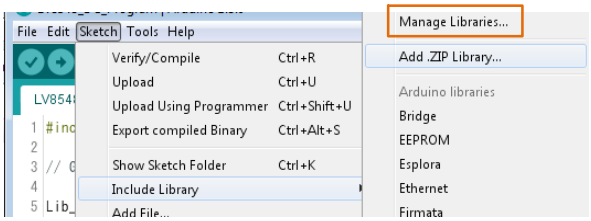

The following window will appear. Select the "LV8702_APILibrary" folder, and open this folder as shown below.
Click the folder only once and push "Open" button.
Do not double click the folder.



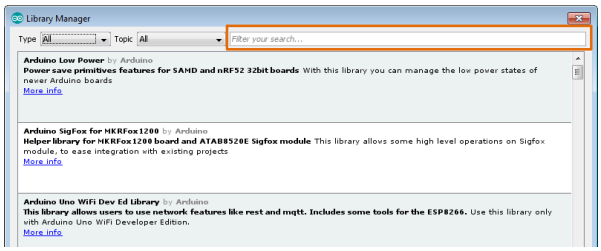

Unless updating the function library, including the API function library only needs to be done once.

### Including the TimerOne Library

As shown below, navigate to "Sketch → Include Library → Manage Libraries"

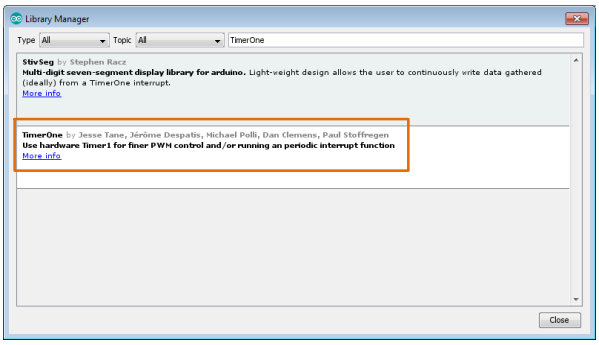



Once the library manager starts up, please type "TimerOne" in the search bar at the top.




*Some screen captures are taken from different motor driver modules

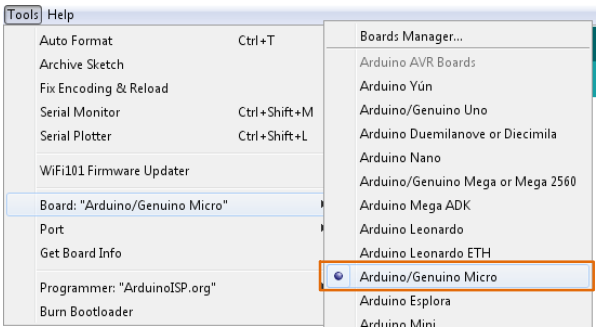
Select and install “TimerOne” in the search results



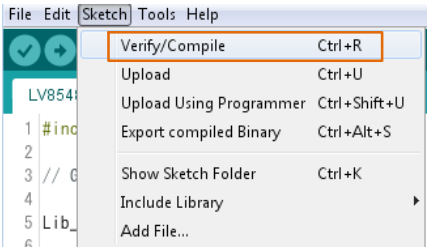

## Compiling the Arduino Program

### ・Write to Arduino

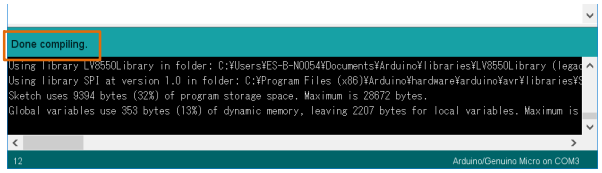
1. Select the Arduino board to upload to by navigating to “Tools → Board → Arduino/Genuino **Micro**”.



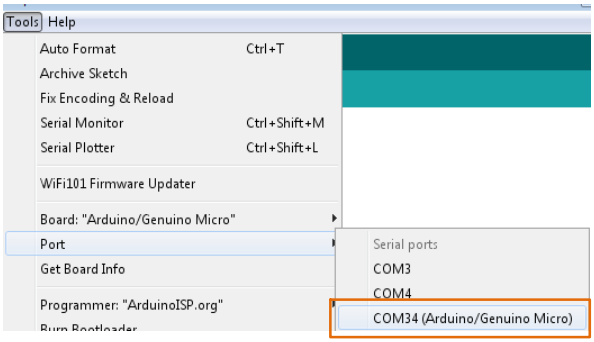

2. Write an Arduino sketch and navigate to “Sketch → Verify/Compile” when finished writing.



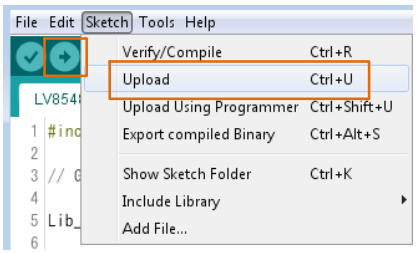

The IDE will display “Done compiling” after a successful compile has been verified.




3. **Connect the PC to the Arduino Micro via USB** and select the corresponding COM port as shown below.



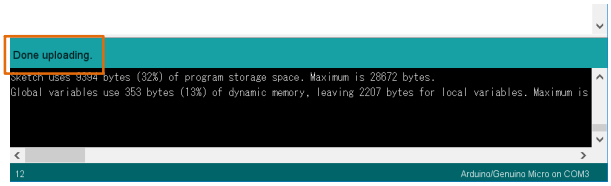

4. Upload the sketch by clicking “Sketch → Upload” or by pressing the ⟶ button.




In the process of uploading, the Arduino Micro bootloader will be installed.




The IDE will display “Done uploading” after a successful upload to the Arduino.




Sketches written to the Arduino will not be erased unless rewritten.
If the uploading of the program fails confirm the details in step 1 check board name and step 3 check serial port connection.

### Rewrite the Arduino Program

In the case of 1. or 2. below, please rewrite the program to the Arduino

1. When updating the API Function library.
   a) Delete the existing API function library by navigating to Documents\Arduino\libraries and deleting the LV8702_APILibrary folder.
   b) Include the latest API function library. Please save the latest API function library in your preferred directory on your PC.
   (See Page 3, "Software Setup")
   Include the latest API function library.
   (See Page 3 "API Library")
   c) Compile and write the program to the Arduino
   (See Page 4, "Compiling the Arduino program • Write to Arduino")
2. When evaluating the LV8548 DC and other motor driver modules.
   (Assuming other motor driver module libraries have already been included)
   a) Compile and write the program to the Arduino
   (See Page 4, "Compiling the Arduino program • Write to Arduino")

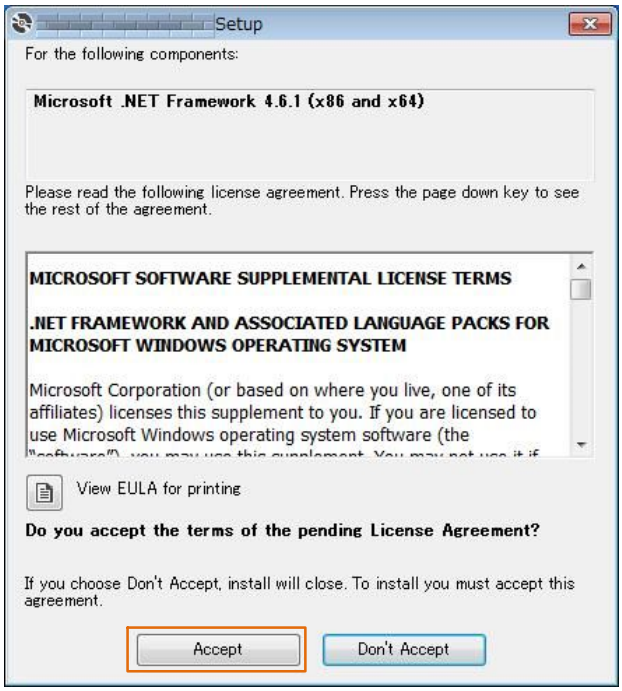
If you are testing the any other motor driver module library for the first time, please operate according to their corresponding manual.

### GUI Installation

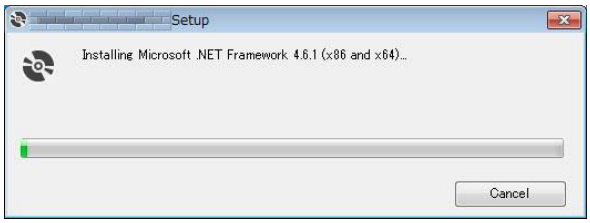
1. Double click on setup.exe in the ON_MD_Module_Kit_GUI folder and proceed with the installation as shown below.

   *If a previous version of this GUI (ON_MD_Module_Kit_GUI) has already been installed and the most recent GUI will be installed, please overwrite and update the existing GUI.
   (No need for uninstallation)

2. The .NET Framework4.6.1 installation window will appear if the PC does not have the necessary version. If .NET Framework4.6.1 is not installed, click Accept to install.

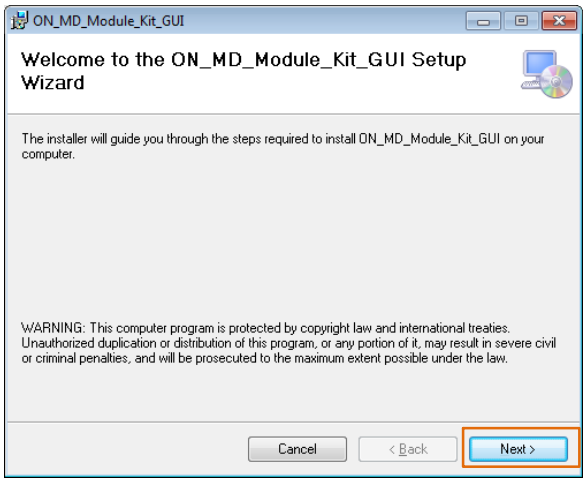



The following window shows the installation in progress. (This may take several minutes)

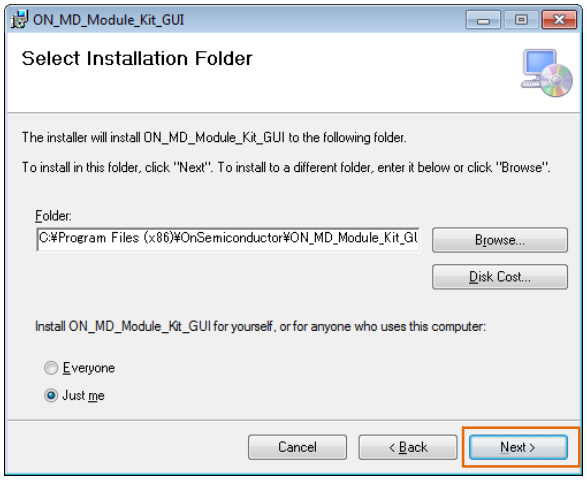



If the installation requires a reboot to complete, a message will appear on the screen. Please press "Yes" to reboot.
Once rebooted, run setup.exe installation file again.

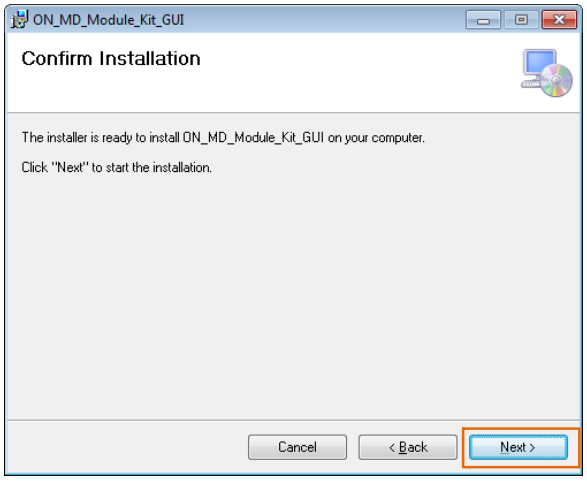
3. Install the GUI
   Press “Next”




4. Specify installation folder, or just click “next” to continue with the installation

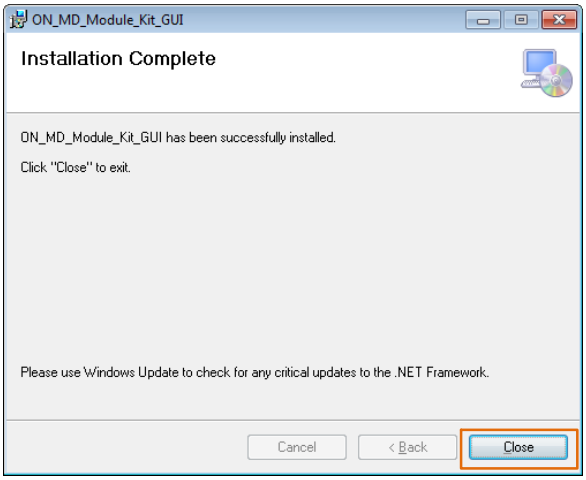



Please press “next”.




User administrative rights may be required for installation. If an alert for user account control is displayed, please select “Yes.”

The following screen will be displayed after a successful installation.

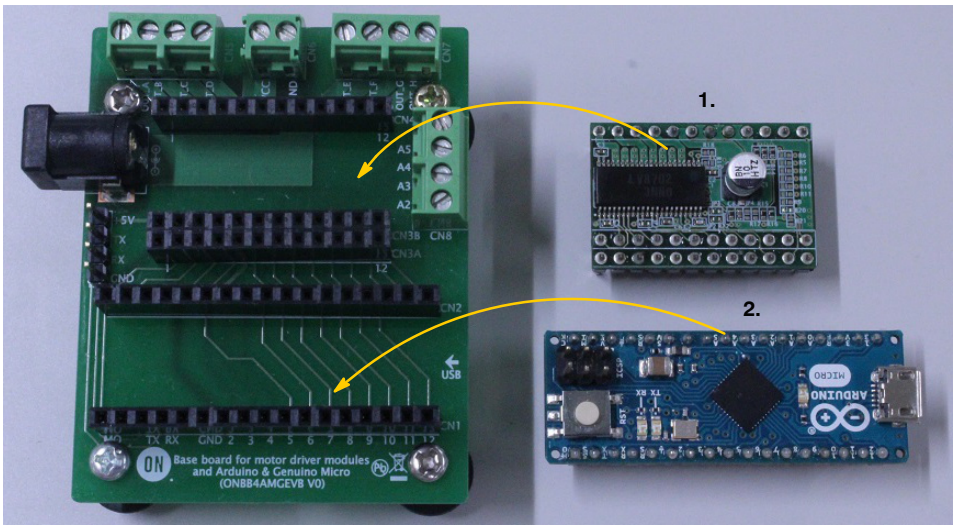



Make sure the GUI shortcut icon is created on the desktop as shown below and that the program has been added to the Windows start menu.



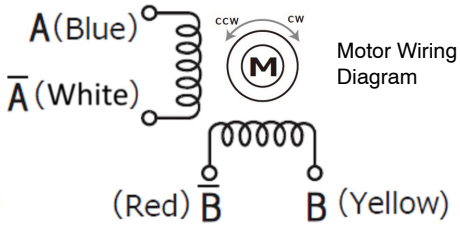
**Hardware Setup**

1. Connect the LV8702VSLDGEVB (Motor Driver Module) to the ONBB4AMGEVB (Motor Driver Baseboard). Insert the module into the baseboard, while being careful not to bend the header pins
2. Connect the Arduino/Genuino Micro to the baseboard, as seen in the figure above. The USB connection port should be labeled on the baseboard "←USB" to dictate the orientation of the Arduino




3. Plug in the motor wires (Insulation between 5 mm – 10 mm) into the output terminals on the baseboard – OUT_A/B/C/D (CN5). Firmly fix the wires by screwing down the terminals with a flat−heat screwdriver. Refer to the table below for motor connectivity:

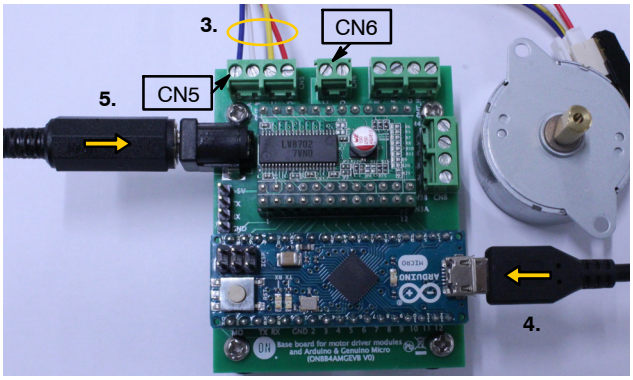
| | |
|---|---|
| OUT_A | A (Blue) |
| OUT_B | $\overline{A}$ (White) |
| OUT_C | B (Yellow) |
| OUT_D | $\overline{B}$ (Red) |



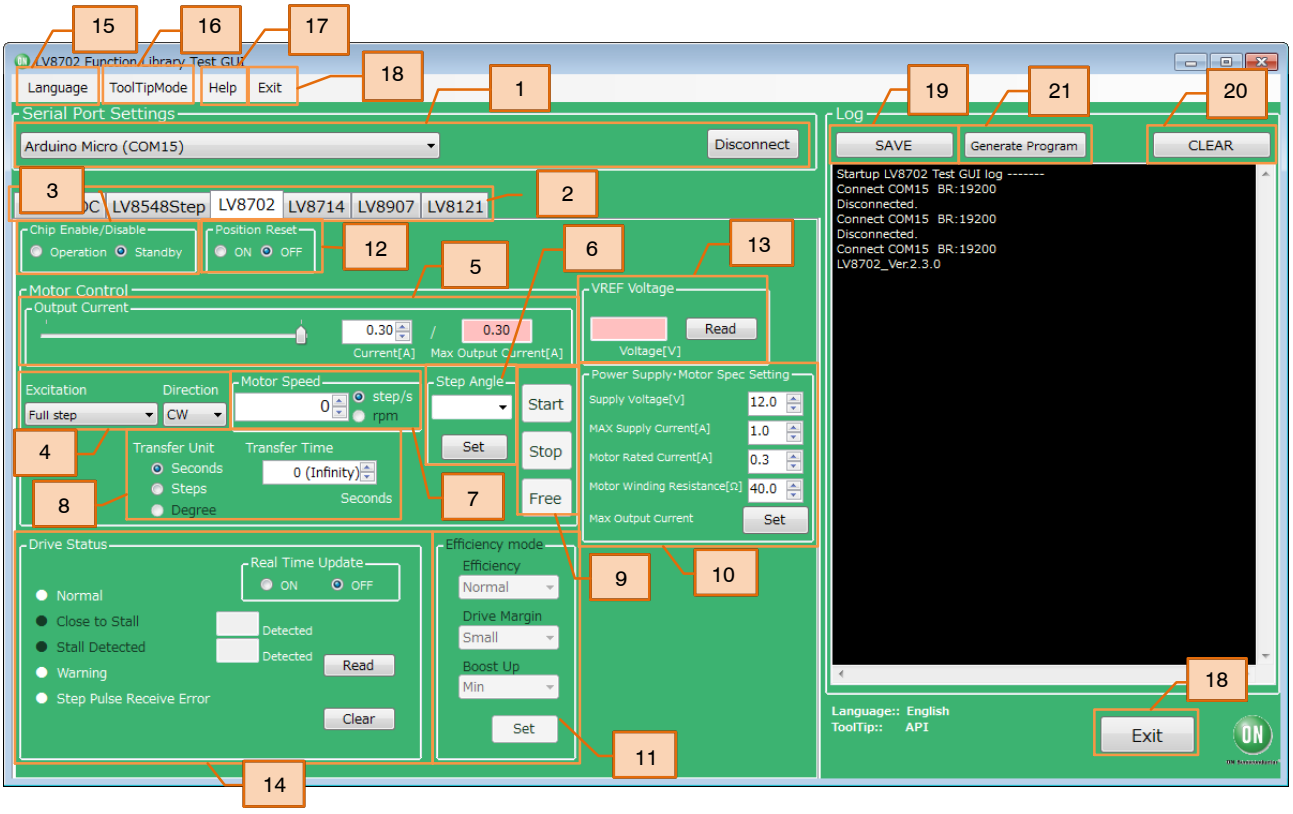

Motor Wiring Diagram

(Do not worry about incorrect wiring. Incorrect wiring will NOT cause any damage to hardware.)

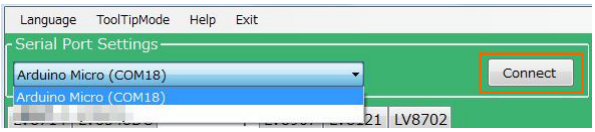
4. Connect the Arduino to a PC using a USB cable
5. Insert the AC Power adapter into the DC Jack on the baseboard. Be sure that a **Center positive type** adapter is being used with an **output voltage 9~32 V**
   If using power supply cables, please connect the positive terminal to VCC and the negative terminal to GND in the CN6 connector. **The power supply terminals +/− at CN6 are polarity sensitive**. (VCC = +, GND = −)



## How to Use the GUI




1. Double−click the GUI shortcut located on the desktop and connect the COM serial port




Available COM ports will appear in the drop−down box.
If the Arduino is connected correctly to the PC, an option for “Arduino Micro (COMx)” will be available (where x is the number associated with the port); select that COM port and click connect.

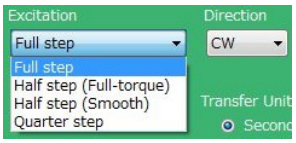
2. After connecting the Arduino, the GUI will automatically navigate to the LV8702 tab
3. Set the chip enable function
   As LV8702V to the chip enable function
   This function switches between Standby and Operating modes.
   [Operation] … Operating mode
   [Standby] … Standby mode
4. Specify the desired motor excitation and direction of rotation

| Excitation (Excitation Method) | One−step Rotation Angle |
|---|---|
| Full step | 1/1 Step angle |
| Half step (Full−torque) | 1/2 Step angle |
| Half step (Smooth) | 1/2 Step angle |
| Quarter step | 1/4 Step angle |

The following table compares LV8702 data sheet notation.

| GUI Notation | Data Sheet Notation |
|---|---|
| Full step | Full step |
| Half step (Full−torque) | Half step full−torque |
| Half step (Smooth) | Half step |
| Quarter step | Quarter step |

Direction:
CW – Clockwise
CCW – Counter clockwise




NOTE: The direction of rotation may vary depending on how the motor wires are connected to the terminals.

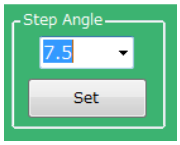
5. Set the output motor current




The output current can be set within the limit value (Box on the right) according to 10 by operating the slider or the value of Box on the left.
The setting values are incremented by 0.01 A, but the accuracy is limited due to the specifications of the Arduino.

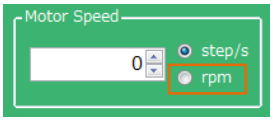
NOTE: The output current is set to 0.3 [A] for the reference motor and the recommended adapter at startup, and it is not necessary to set it again for use in the same environment.

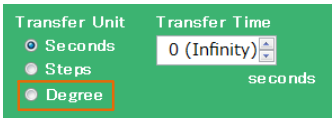
6. Set the step angle for the motor



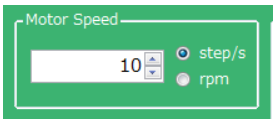

NOTE: The step angle will vary depending on the motor. If using the included reference motor, please set it to 7.5°.

The step angle setting enables the “rpm” setting under Motor Speed and “Degree” setting under Transfer Unit.




7. Set the motor speed in step/s or rpm



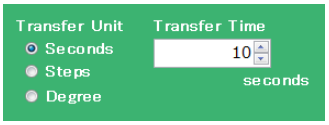

step/s (=pps): Frequency of the number of steps per second
rpm: Rotational speed of the motor per minute

Stepper motors are not intended for sudden acceleration, and will vibrate if the control exceeds the maximum rotation speed. When using the included reference motor with the recommended AC adaptor, please set the rotation speed according to the table below.

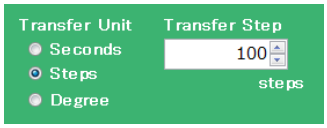
| | | |
|---|---|---|
| Full step | 1~560 step/s | 2~700 rpm |
| Half step (Full−torque) | 1~1120 step/s | 1~700 rpm |
| Half step (Smooth) | | |
| Quarter step | 1~2240 step/s | |

8. Set the units and value of the transfer condition for the control signal. After the specified transfer condition elapses, the motor will stop and hold its torque. **Since current continues to flow to the motor, it is important to pay attention to heat generation.** Please select the Free button to stop powering the motor.
For an unspecified rotation time, select 0 (Infinity)

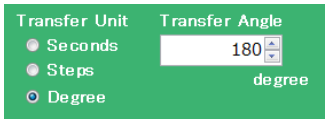
Example 1: Rotating the motor for **10 seconds**
Transfer Unit = *Seconds*
Transfer Time = *10 [seconds]*




Example 2: Rotating the motor for **100 steps**
Transfer Unit = *Steps*
Transfer Step = *100 [steps]*



xample 3: Rotating the motor **180 degrees**
Transfer Unit = *Degree*
Transfer Angle = *180 [degree]*




9. When the Start button is pressed, the motor will rotate. If changes to the Output Current, Excitation or Motor Speed are made, the changes will take effect upon pressing the Start button.
   To change the direction of rotation, it is recommended to stop the motor first with the Stop button, change the value, and then press the Start button to begin the motor rotation.
   Pressing the Stop button causes the motor to stop, and hold the torque.
   When the Free button is pressed, the motor will stop and lose the torque.
   To maintain the position while the motor rotates, or to pause the motor and restart from the same position, select the Stop button.
   At this time, because of **the current still flowing** to the motor, **attention must again be paid to heat generation.**
10. Set the specifications of the power supply and motor to be used.
    Supply Voltage... Power supply voltage
    MAX Supply Current... Maximum power supply current
    Motor Rated Current... Motor Rated Current
    Motor Winding Resistance... Motor coil resistance
    Max Output Current Set buttons ... Calculate the maximum output current from the above four setting values, and 5 limit the set range of the output motor current to enable safer motor driving.
11. Set the high−efficiency drive function.
    (This function does not work in slow speed and fast speed range where the feedback signal can't be stably detected. This function cannot be selected when Full Step is set.)
    The high−efficiency drive function automatically optimizes the motor current according to the motor load and rotational speed. By turning on this function, power consumption, heat, vibration, and noise can be reduced. The high−efficiency drive function consists of the following three parameters.

    **Efficiency**... ON/OFF of high−efficiency drive function
    When the high−efficiency drive function is turned on, the motor current is automatically adjusted within the range of the limit set in 5.

    **Drive Margin**... Margin adjuster Function
    Adjust the margin from the minimum current that can rotate the motor.
    When the margins are set to Small, the current consumption is the smallest, but the number of boost−ups may be increased for small load−changes.
    For this reason, set the margin to be larger for applications with excessive load variation.

| Drive Margin | Current Consumption | Robustness |
|---|---|---|
| Small | Low | Weak |
| Middle | ↕ | ↕ |
| Large | Normal | Strong |

    **Boost Up**... boost−up adjuster function
    This process increases the motor current rapidly and temporarily in order to maintain the motor rotation in response to load variation.
    Increasing the boost increases the rush load resistance, but increasing the boost increases the rotational stability.
    The worsened of rotational stability can be suppressed by increasing the Drive Margin.

| Boost Up | Robustness | Rotational Stability |
|---|---|---|
| Min | Weak | Stable |
| Low | ↕ | ↕ |
| High | | |
| Max | Strong | Unstable |

12. Set the reset function.
    The reset function resets the excitation position and locks the motor.

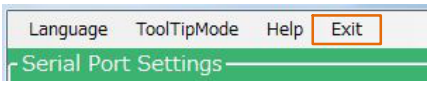
    [ON] ... Reset ON
    [OFF] ... Reset OFF

    **Neither is turned off the power.**

13. Voltage reading function
14. Status check function
15. GUI Language Settings
16. GUI Tool Tip Display
17. Help function
    (See Appendix for more information)

18. Closing the GUI

When finished, exit the GUI by pressing the “Exit” button at the bottom right of the screen or by clicking “Exit” item at the top menu bar.

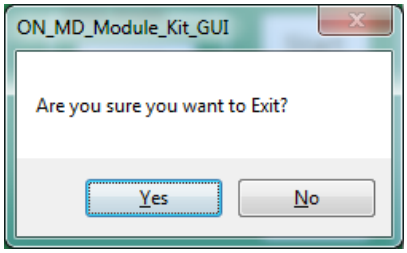








The following popup message will be displayed when exiting the GUI.




Select “Yes” to quit the GUI.

Select “No” to cancel the exit and return to the main screen.

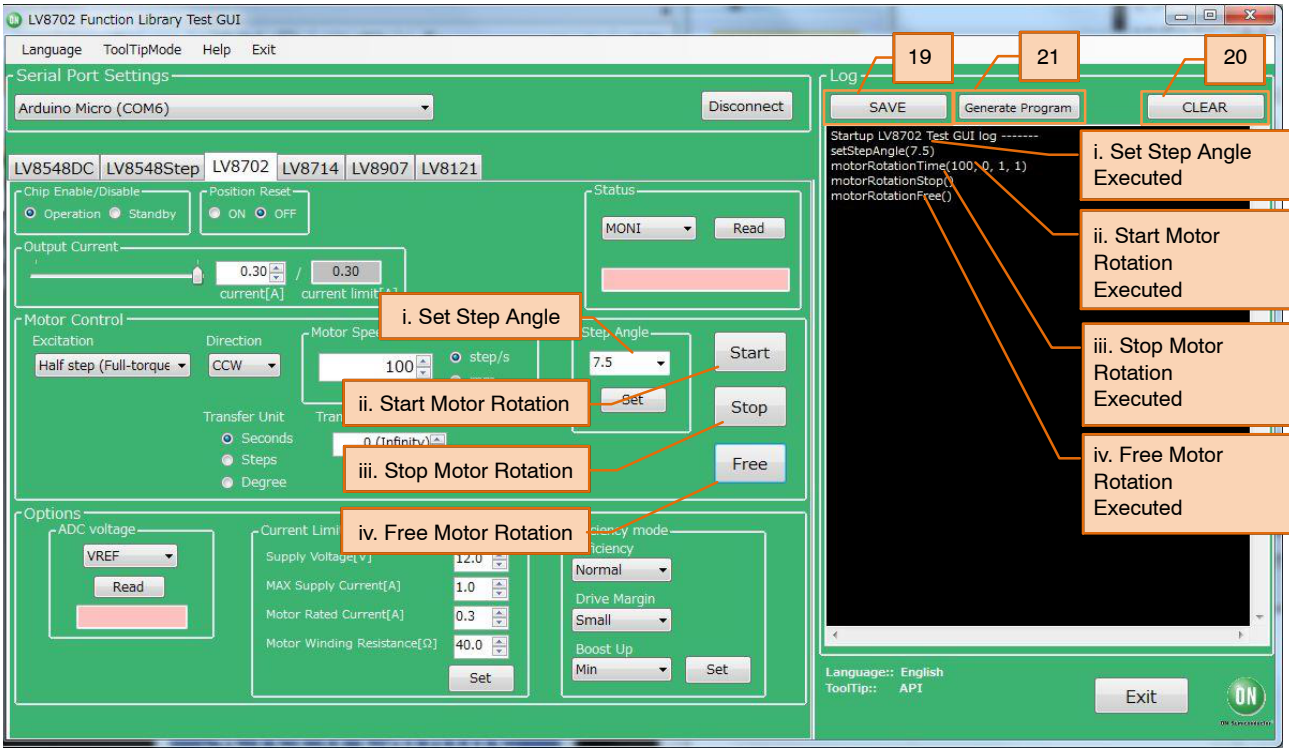
If the GUI is closed while the motor is still running, the motor will be stopped and the window will close.

For more details on the following features, refer to the next section on How to use the GUI Log:

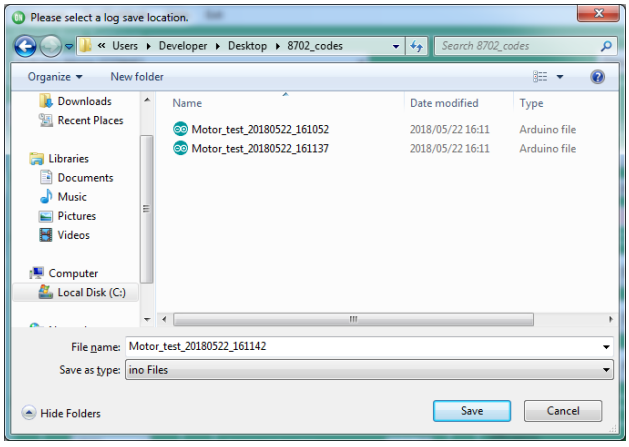
19. Saving the GUI Log
20. Clear the GUI Log
21. Program Generation

### How to Use the GUI Log

The log screen in the GUI shows the serial data sent to Arduino to control its API functions.




19. Saving the GUI Log
    By pressing the “SAVE” button, the content displayed on the work log can be saved as a .txt file or a .csv file.

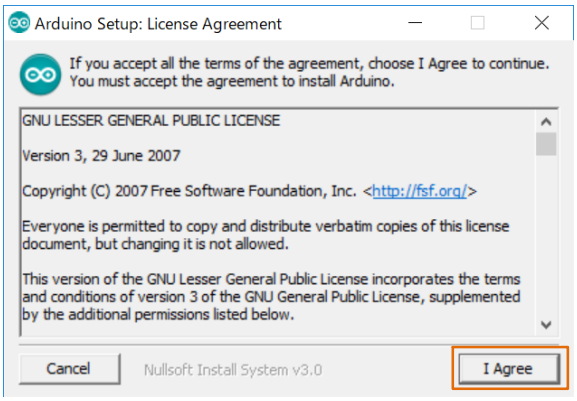



20. Clear GUI Log
    By pressing the “CLEAR” button, the content displayed on the work log will be erased.
21. Program Generation
    Pressing the “Generate Program” will output the executed API functions on the work log into a .ino program file that can be uploaded directly with Arduino. By writing the output of the .ino file to Arduino, motor control can be executed automatically through standalone operation according to the procedure generated through the GUI Log.
    For more details regarding the Arduino program generation function, refer to the attached Appendix under 21 Arduino Program Automatic Generation.

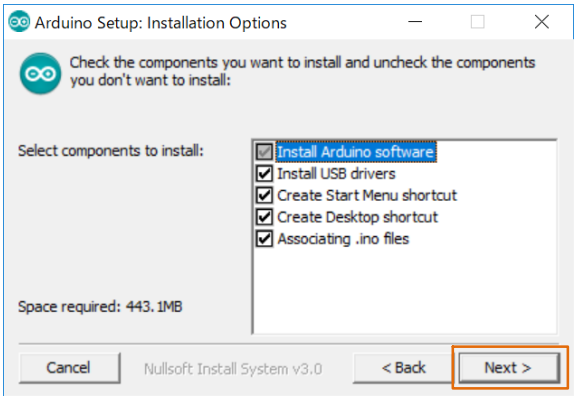
# APPENDIX

## Arduino IDE Installation

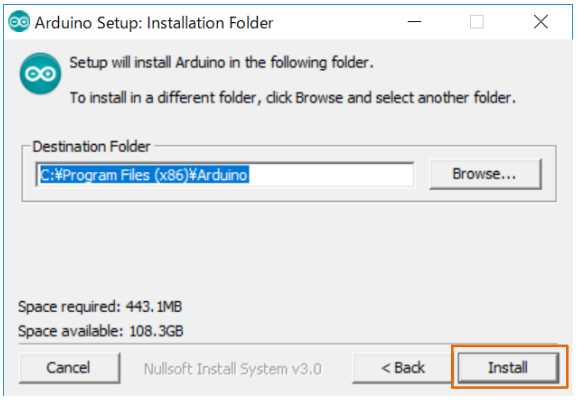
1. Run arduino-1.8.4-windows.exe installer that are included in.ZIP files downloaded from our web site.
   (To avoid software from not operating properly, please use this version and refrain from updating)
2. Read through and agree to the terms and conditions during the setup by clicking the "I Agree" button.



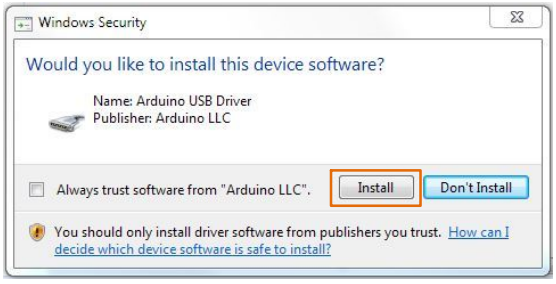

3. Leave the following default components to install as is and press "Next".



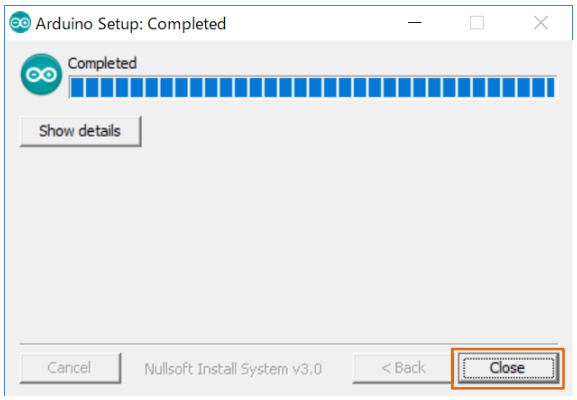

4. Set the install directory and press "Install".




5. If prompted, please install the Arduino USB Drivers
   - Arduino USB Driver
   - Genuino USB Driver
   - libusb−win32
   - Adafruit Industries LLC Port (COM and LPT)
   - Linino Port (COM and LPT)



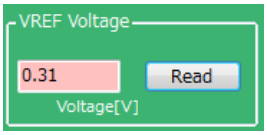

6. Below shows a completed installation for the Arduino IDE. Feel free to click "Close".



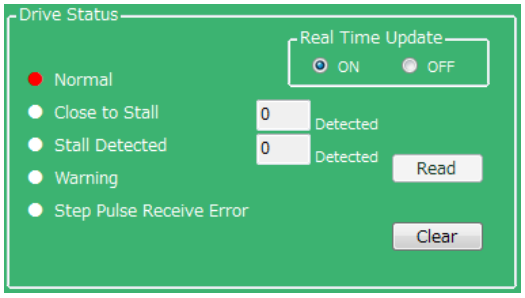
**Supplemental GUI Content**

The following step numbers correspond to the step numbers in the "How to use the GUI" section in the Quick Start Guide.

13. Voltage Reading Function
    READ to display the voltages in the pink column.
    VREF … Output Motor Current Reference Voltage
    Displays the reference voltage for 5 Output motor current setting
    VREF = (Output Motor current) × 5 × 0.22 Ω




14. Status Check Function
    The red lamps and counters indicate the status detected by the LV8702 from the outputs of each terminal (DST1, DST2, MONI) of the LV8702.
    When the Real Time Update is turned on, the outputs from the terminals are periodically Read and displayed.
    Close to Stall and Stall Detected are enabled when the high−efficiency drive function is set to Efficiency = High, and the number of times of occurrence is displayed in boxes.
    This count is reset by pressing Clear.
    (Maximum count = 255)
    When the Real Time Update is turned off, the status when the Read button is pressed is displayed.




Normal:
Normal conditions (DST1:H, DST2:H)

Close to Stall:
Electrical detection of danger of stall
(DST1:H, DST2:L)

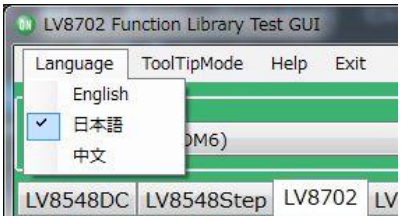
Stall Detected:
Electrical detection of stall state
(DST1:L, DST2:H)

Warning:
Detects the state of short−circuit between the OUT terminal and the power supply and GND of the OUT terminal and the IC overheat state
(DST1: L, DST2: L).

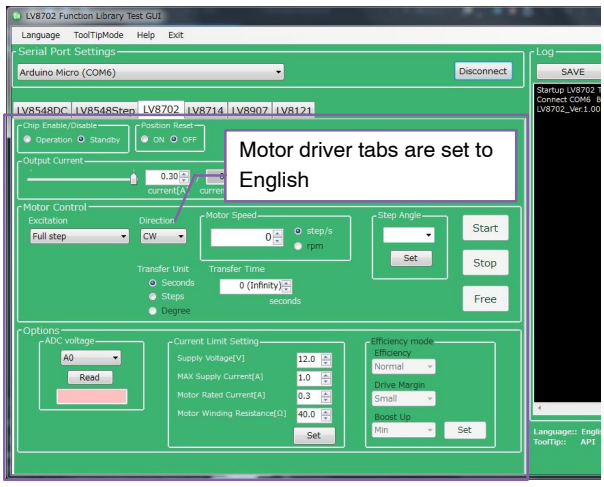
Step Pulse Receive Error:
Detects (MONI) positional deviation of the motor caused by noise−jumping into the STEP input terminal of the LV8702.

About Warning:
When the overheat protection function is activated, all OUTs turn OFF and the motor stops.
When the temperature drops, the Warning state is released and the motor drive is restarted.
When the short−circuit protection function of the OUT terminal is activated, all OUTs are turned OFF and the motor stops.
This condition can be reset by switching Chip Enable/ Disable from Operation to Standby to Operation.

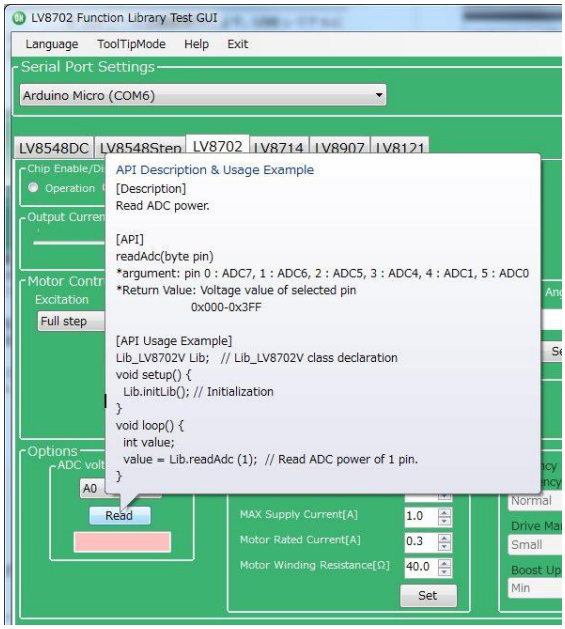
15. GUI Language Settings



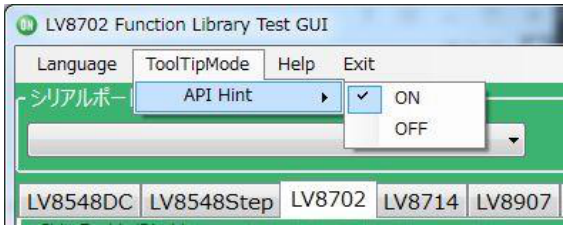

Languages can be changed from the menu on the top left of the window.
Motor driver tab languages are not changed.
(Please see tool tip for translations)



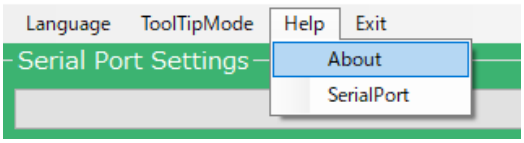
16. GUI Tool Tip Display
The GUI sends serial data to execute API functions in the Arduino through USB.
With the API Hints turned on, hover over different buttons and settings to display descriptions and functions associated with that item.



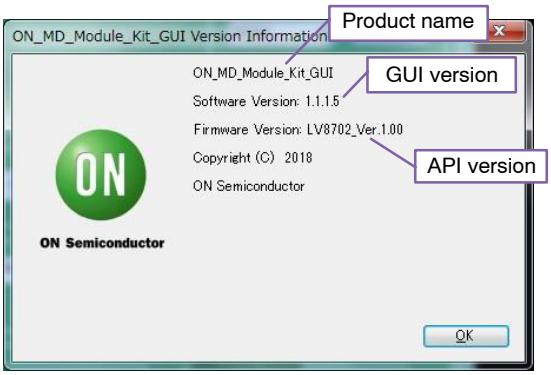

The API hints can be turned off by navigating through the ToolTipMode menu at the top of the window



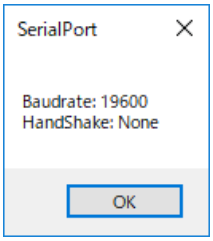

17. Help Function
From the help menu, it is possible to view GUI and API version information, as well as serial communication details.




Selecting About will display the following window.

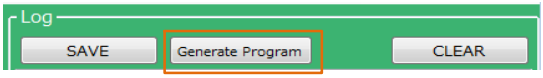



Selecting SerialPort will display the overview of serial communication.



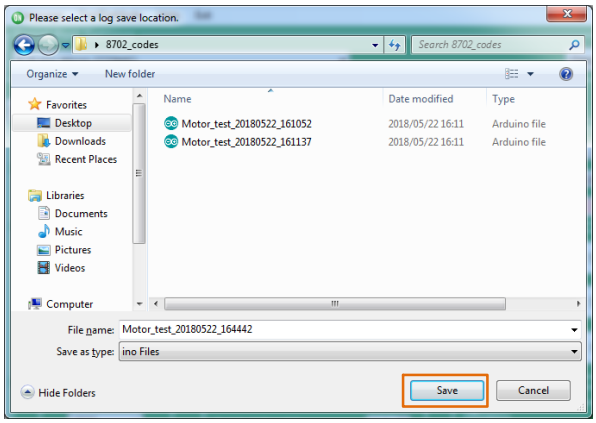

21. Arduino Program Automatic Generation
When using the GUI, automatically generated Arduino code will be output to the log window.
It is possible to export the code to an Arduino sketch (.ino) file that can be used with the Arduino IDE.
Once the automatically generated code is imported into the Arduino IDE, it is possible to mimic the procedures from the GUI.

1) Generating and writing Arduino code
Pressing Generate Program will open a save file dialog.

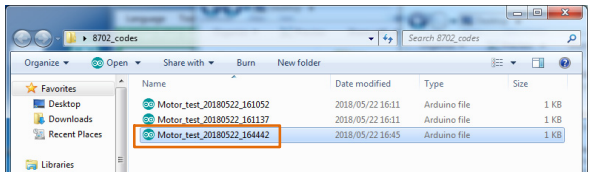


Please select the desired directory for the file to be saved (e.g. Desktop or Documents) and click Save.

When using the GUI, LV8702_Program.ino is the name of the program, so a different name will need to be used to save the exported file.




Double click the newly saved Arduino program to open the Arduino IDE.




The GUI and Arduino IDE cannot be connected to a PC at the same time. To upload the Arduino program, please exit or select Disconnect in the GUI, and proceed working in the Arduino IDE. Follow the instructions in the Quick Start Guide in P3 “Compiling the Arduino Program – Write to Arduino”.

2) Using the generated Arduino program

After each API operation in the generated program, a delay(0) is inserted. By changing the value of the argument (0)* to the delay function, the user is able to freely adjust the rotation time, as well as the interval time between each API call execution. This can be used to achieve the desired stand−alone operation.

*Parameter units are msec (1000$^{th}$ of a second). For a delay of 1 second, use delay(1000).

[Example of changing the interval time]

```
#include <LV8702_Lib.h>
#include <TimerOne.h>
Lib_LV8702V Lib;
void setup()
{
  Serial.begin(19200);
  Lib.initLib();
  Timer1.initialize(50);
  Timer1.attachInterrupt(interrupt);
  delay(5000);  ⟶ NOTE
  Lib.setChipEnable(1);
  delay(0);//0msec
  Lib.setStepAngle(7.5);
  delay(0);//0msec
  Lib.motorRotationDeg(10, 720, 0, 0);
  delay(5000);//0msec
  Lib.motorRotationStop();
  delay(0);//0msec
  Lib.motorRotationFree();
  delay(0);//0msec
}
```

*“motorRotationDeg(10, 720.0, 0, 0)”*
will start the motor with Full step, CW, 10 Step/s, and an angle of 720 deg
*“delay(5000)”*
will drive the motor for 5000[msec] (5s)
*“motorRotationStop()”*
will stop the motor

After the time specified in the delay command has elapsed, the next command will be executed.
If the delay setting is 0 or too short, some motor operations will complete so quickly, that it will be imperceptible.

NOTE: The delay setting at the beginning of the setup method will execute in the case of a new USB connection, Arduino reset, or upon uploading a sketch to the Arduino. In this case, the Arduino will delay for 5 seconds while writing the initial settings.

## BOARD SCHEMATIC (1/2)

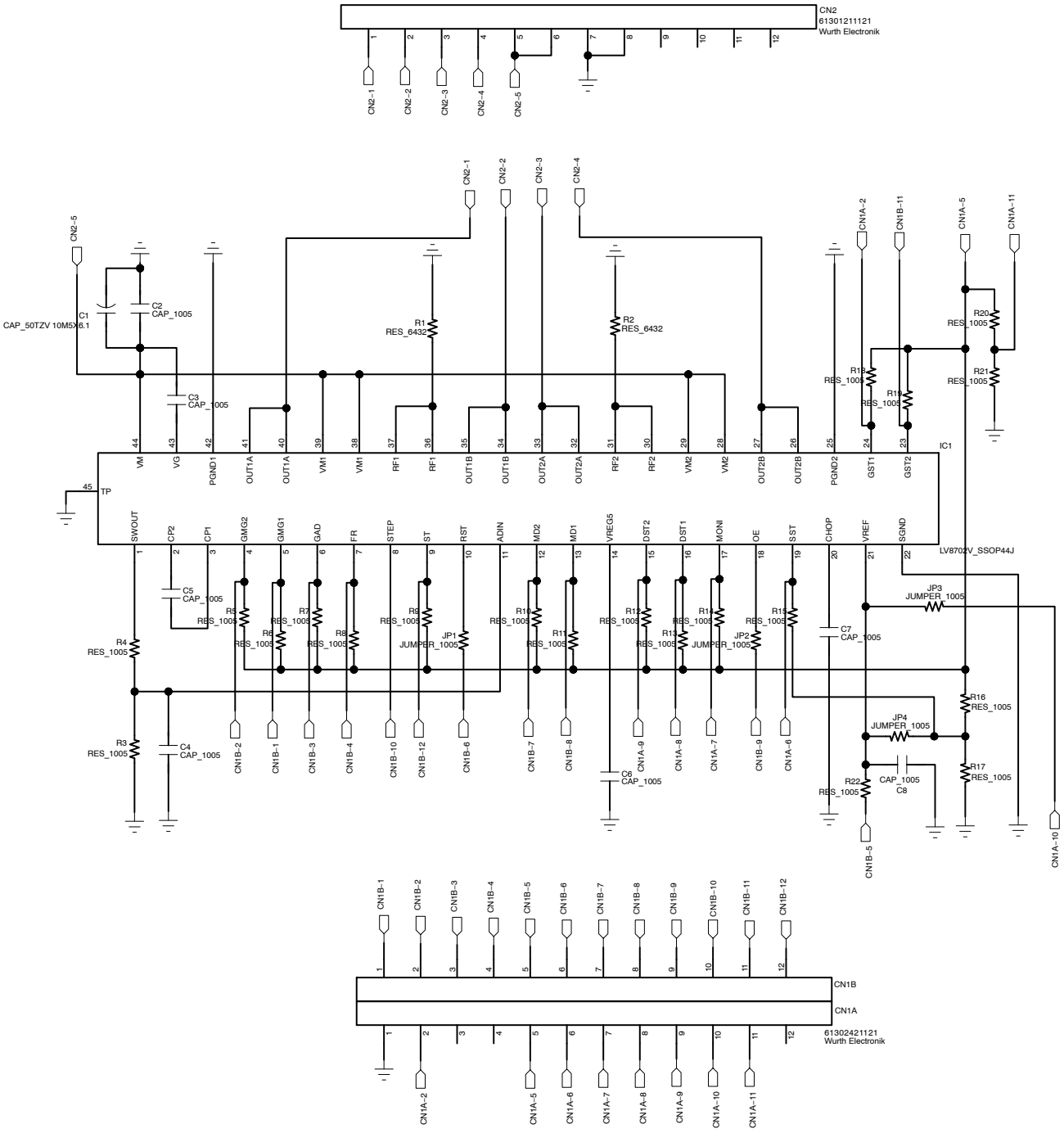



**Figure 3. LV8702VSLDGEVB Schematic**

**Table 1. LV8702VSLDGEVB BILL OF MATERIALS**

| Designator | Qty. | Description | Value | Tolerance | Footprint | Company | Part Number |
|---|---|---|---|---|---|---|---|
| IC1 | 1 | Motor Driver | – | – | SSOP44J | ON semiconductor | LV8702V |
| R1−2 | 2 | Thick Film Resistor | 0.22 Ω,1 W | ±5% | 6432(2512Inch) | KOA | SR73W3AT**R22J |
| R3 | 1 | Thick Film Resistor | 15 kΩ, 0.1 W | ±5% | 1005(0402Inch) | KOA | RK73B1ET**153J |
| R4 | 1 | Thick Film Resistor | 100 kΩ, 0.1 W | ±5% | 1005(0402Inch) | KOA | RK73B1ET**104J |
| R5−11, R18−19 | 9 | Thick Film Resistor | 47 kΩ, 0.1 W | ±5% | 1005(0402Inch) | KOA | RK73B1ET**473J |
| R12−14, R16 | 4 | Thick Film Resistor | 47 kΩ, 0.1 W | ±5% | 1005(0402Inch) | KOA | RK73B1ET**473J |
| R15 | 11 | Thick Film Resistor | 12 kΩ, 0.1 W | ±5% | 1005(0402Inch) | KOA | RK73B1ET**123J |
| R17 | 1 | Thick Film Resistor | 15 kΩ, 0.1 W | ±5% | 1005(0402Inch) | KOA | RK73B1ET**153J |
| R20 | 1 | Thick Film Resistor | 100 kΩ, 0.1 W | ±5% | 1005(0402Inch) | KOA | RK73B1ET**104J |
| R21 | 1 | Thick Film Resistor | TBD, 0.1 W | ±5% | 1005(0402Inch) | | TBD |
| R22 | 1 | Thick Film Resistor | 27 kΩ, 0.1 W | ±5% | 1005(0402Inch) | KOA | RK73B1ET**273J |
| JP1−3 | 3 | Jumper | 0 Ω, 1 W | ±20% | 1005(0402Inch) | KOA | RK73Z1ET** |
| JP4 | 1 | Jumper | 0 Ω, 1 W | ±20% | 1005(0402Inch) | KOA | RK73Z1ET** |
| C1 | 1 | VCC Bypass Capacitor | 10 μF, 50 V | ±20% | 5 x 5.5 | Wurth Electronik | 865080642006 |
| C2, C3, C5, C6, C8 | 5 | Ceramic multilayer Capacitor | 0.1 μF, 100 V | ±10% | 1005(0402Inch) | Murata Manufacturing | GRM155R62A104KE14D |
| C4 | 1 | Ceramic multilayer Capacitor | 1000 pF, 50 V | ±5% | 1005(0402Inch) | Murata Manufacturing | GRM1555C1H102JA01J |
| C7 | 1 | Ceramic multilayer Capacitor | 150 pF, 50 V | ±10% | 1005(0402Inch) | Murata Manufacturing | GRM1555C1H151JA01J |
| CN1A, 1B | 1 | Pin header to baseboard | 12 pins x 2 | – | 30.48 x 5.08 | Wurth Electronik | 61302421121 |
| CN2 | 1 | Pin header to baseboard | 12 pins | – | 30.48 x 2.54 | Wurth Electronik | 61301211121 |
| PCB | 1 | PCB | – | | 30.48 x 20.32 | | |

NOTE: Parts highlighted in yellow are not mounted at the time of product shipment.

## BOARD SCHEMATIC (2/2)

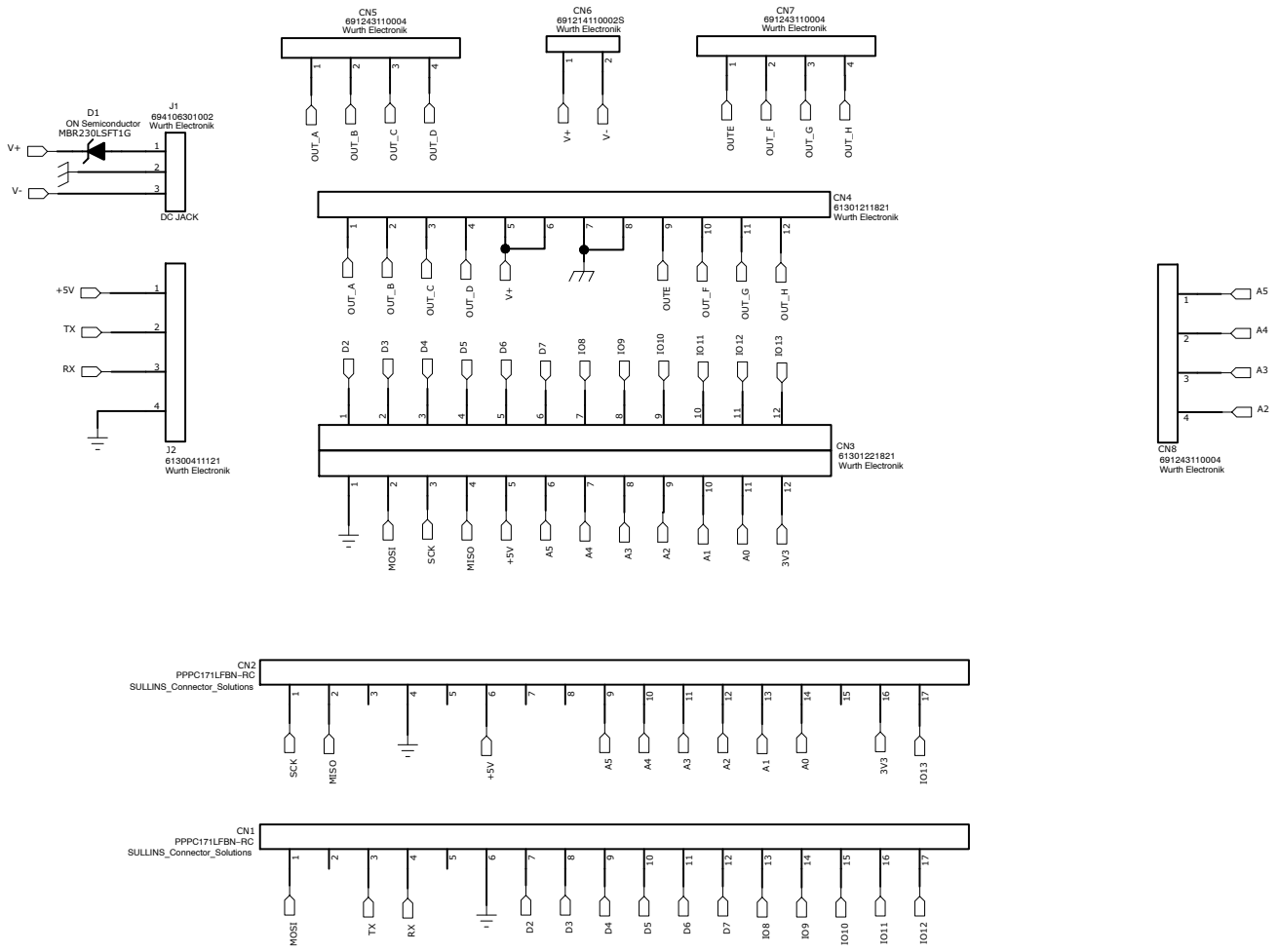



**Figure 4. ONBB4AMGEVB Board Schematic**

### Table 2. BASEBOARD BILL OF MATERIALS

| Designator | Qty. | Description | Value | Tolerance | Footprint | Company | Part Number |
|---|---|---|---|---|---|---|---|
| D1 | 1 | Diode | – | – | SOD123 | ON Semiconductor | MBR230LSFT1G |
| CN1,2 | 2 | Arduino Micro connector | – | – | ∅1.02 x17 −2.54 pitch | Hirosugi−Keiki | FSS−41085−17 |
| CN3 | 1 | Module connector | – | – | ∅1.02 x12 x2lines −2.54 pitch | Wurth Electronik | 61302421821 |
| CN4 | 1 | Module connector | – | – | ∅1.02 x12 −2.54 pitch | Wurth Electronik | 61301211821 |
| CN5,7,8 | 3 | Motor connectors | – | – | ∅1.1 x4 −3.5 pitch | Wurth Electronik | 691243110004 |
| CN6 | 1 | Power connectors | – | – | ∅1.1 x2 −3.5 pitch | Wurth Electronik | 691214110002S |
| J1 | 1 | DC barrel jack | – | – | 9.0 x 14.5 | Wurth Electronik | 694106301002 |
| J2 | 1 | UART pin headers | – | – | ∅1.1 x4 −2.54 pitch | Wurth Electronik | 61300411121 |
| C1 | 1 | Electrolytic capacitor | 100 μF, 50 V | ±10% | – | Wurth Electronik | 860020674015 |
| PCB | 1 | PCB | – | – | 80 x 60 | | |

NOTE: When using a custom−made baseboard, be sure to **mount an electrolytic capacitor equivalent to C1 between VCC and GND**. Neglecting to install this capacitor may lead to damage and malfunction of any connected driver modules.

ON Semiconductor and the ON Semiconductor logo are trademarks of Semiconductor Components Industries, LLC dba ON Semiconductor or its subsidiaries in the United States and/or other countries. ON Semiconductor owns the rights to a number of patents, trademarks, copyrights, trade secrets, and other intellectual property. A listing of ON Semiconductor's product/patent coverage may be accessed at www.onsemi.com/site/pdf/Patent−Marking.pdf. ON Semiconductor is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner.

The evaluation board/kit (research and development board/kit) (hereinafter the "board") is not a finished product and is as such not available for sale to consumers. The board is only intended for research, development, demonstration and evaluation purposes and should as such only be used in laboratory/development areas by persons with an engineering/technical training and familiar with the risks associated with handling electrical/mechanical components, systems and subsystems. This person assumes full responsibility/liability for proper and safe handling. Any other use, resale or redistribution for any other purpose is strictly prohibited.

The board is delivered "AS IS" and without warranty of any kind including, but not limited to, that the board is production−worthy, that the functions contained in the board will meet your requirements, or that the operation of the board will be uninterrupted or error free. ON Semiconductor expressly disclaims all warranties, express, implied or otherwise, including without limitation, warranties of fitness for a particular purpose and non−infringement of intellectual property rights.

ON Semiconductor reserves the right to make changes without further notice to any board.

You are responsible for determining whether the board will be suitable for your intended use or application or will achieve your intended results. Prior to using or distributing any systems that have been evaluated, designed or tested using the board, you agree to test and validate your design to confirm the functionality for your application. Any technical, applications or design information or advice, quality characterization, reliability data or other services provided by ON Semiconductor shall not constitute any representation or warranty by ON Semiconductor, and no additional obligations or liabilities shall arise from ON Semiconductor having provided such information or services.

The boards are not designed, intended, or authorized for use in life support systems, or any FDA Class 3 medical devices or medical devices with a similar or equivalent classification in a foreign jurisdiction, or any devices intended for implantation in the human body. Should you purchase or use the board for any such unintended or unauthorized application, you shall indemnify and hold ON Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that ON Semiconductor was negligent regarding the design or manufacture of the board.

This evaluation board/kit does not fall within the scope of the European Union directives regarding electromagnetic compatibility, restricted substances (RoHS), recycling (WEEE), FCC, CE or UL, and may not meet the technical requirements of these or other related directives.

FCC WARNING − This evaluation board/kit is intended for use for engineering development, demonstration, or evaluation purposes only and is not considered by ON Semiconductor to be a finished end product fit for general consumer use. It may generate, use, or radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to part 15 of FCC rules, which are designed to provide reasonable protection against radio frequency interference. Operation of this equipment may cause interference with radio communications, in which case the user shall be responsible, at its expense, to take whatever measures may be required to correct this interference.

ON Semiconductor does not convey any license under its patent rights nor the rights of others.

LIMITATIONS OF LIABILITY: ON Semiconductor shall not be liable for any special, consequential, incidental, indirect or punitive damages, including, but not limited to the costs of requalification, delay, loss of profits or goodwill, arising out of or in connection with the board, even if ON Semiconductor is advised of the possibility of such damages. In no event shall ON Semiconductor's aggregate liability from any obligation arising out of or in connection with the board, under any theory of liability, exceed the purchase price paid for the board, if any.

For more information and documentation, please visit www.onsemi.com.

## PUBLICATION ORDERING INFORMATION

**LITERATURE FULFILLMENT**:
**Email Requests to:** orderlit@onsemi.com

**ON Semiconductor Website:** www.onsemi.com

**TECHNICAL SUPPORT**
**North American Technical Support:**
Voice Mail: 1 800−282−9855 Toll Free USA/Canada
Phone: 011 421 33 790 2910

**Europe, Middle East and Africa Technical Support:**
Phone: 00421 33 790 2910
For additional information, please contact your local Sales Representative

◊