



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



## Contact us

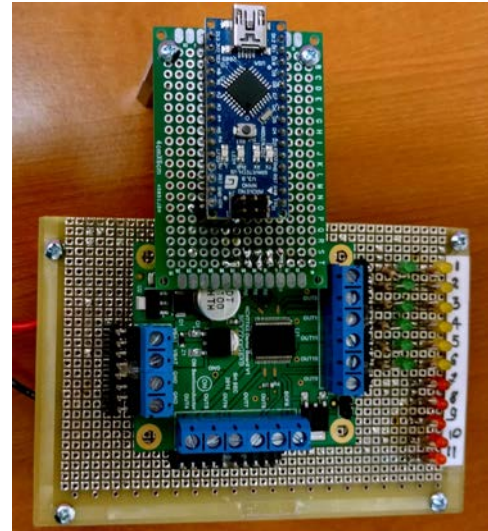
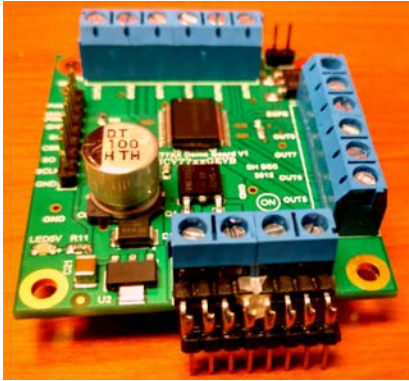
Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: [info@chipsmall.com](mailto:info@chipsmall.com) Web: [www.chipsmall.com](http://www.chipsmall.com)

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



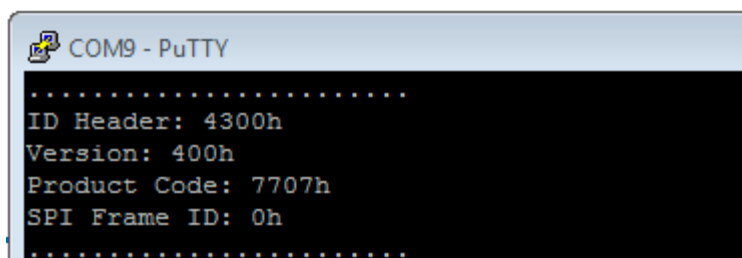
## Test Procedure for the NCV7707GEVB Evaluation Board



1. Attach the connectors on the NCV7707 board output terminals.
2. Plug the module in on the board.
3. Mount the Arduino board on the NCV7707 EVB.
4. Attach the USB Cable to Arduino board and make sure the USB drivers are installed.
5. Apply power 12V.
6. Open a Serial Terminal like PUTTY.exe with baudrate of 38400 and select the device's COM port.

**NOTE: Using Serial Terminal is not a mandatory part for testing the board. Testing can be done by just looking at the output LED patterns.**

7. As the terminal opens the microcontroller will reset itself and loop through the program as follows:
  - a. Read the device information from NCV7707 and display it on the Serial terminal.  
All the outputs are in high impedance mode, thus both LEDs Green and Yellow for each output with half bridge(Outputs 1,2,3,4,5 and 6) will be lit.





- b. Enable outputs 1, 3, and 5 only. In this case high side MOSFET in the half bridge is driving. Remaining all the outputs are in high impedance mode.

```
.....  
High Side Drive  
Outputs: 1, 3, 5  
.....
```

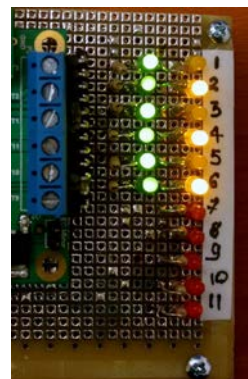


- c. Enable outputs 2, 4, and 6 only. In this case high side MOSFET in the half bridge is driving. Remaining all the outputs are in high impedance mode.



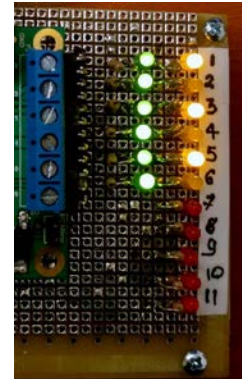
- d. Enable outputs 1, 3, and 5 only. In this case low side MOSFET in the half bridge is driving. Remaining all the outputs are in high impedance mode.

```
.....  
Low Side Drive  
Outputs: 1, 3, 5  
.....  
High Side Drive  
Outputs: 2, 4, 6  
.....
```



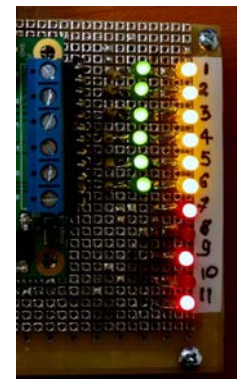
- e. Enable outputs 2, 4, and 6 only. In this case low side MOSFET in the half bridge is driving. Remaining all the outputs are in high impedance mode.

```
.....  
Low Side Drive  
Outputs: 2, 4, 6  
.....
```



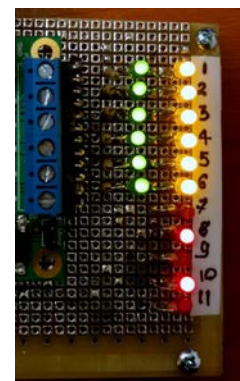
- f. Enable outputs 7, 9, and 11 only. In this case pulled up MOSFET is driving. Remaining all the outputs are in high impedance mode. Red LED with respect to the output will lit.

```
.....  
Pull Ups Drive  
Outputs: 7, 9, 11  
.....
```



- g. Enable outputs 8 and 10 only. In this case pulled up MOSFET is driving. Remaining all the outputs are in high impedance mode.

```
.....  
Pull Ups Drive  
Outputs: 8, 10  
.....
```



And the program loops through all the steps ove