USING NOTEPAD++ AND COLOR SYNTAX HIGHLIGHTING

Application Note **v1.0**

INTRODUCTION

If you intend to use Notepad++ to edit and develop *smart*BASIC source code, then the goal of this document includes the following:

- To explain how to configure Notepad++ so that *smart*BASIC code is displayed with color syntax highlighting.

OVERVIEW

When editing and developing source code for programs most modern text editors can display the code with color syntax highlighting.

The color scheme is based on the underlying syntax of the language for which the code was written. *smart*BASIC has a BASIC flavour and in addition, for each implementation, there are platform specific extension functions.

Notepad++ can be configured directly from the software by using the built in syntax editor which defines the color syntax highlighting for a specifically named programming language. As of September 2013, all BL600 firmware upgrade zip files from Laird contain a file called smartBASIC(Notepad++).xml which can be used to configure Notepad++ to display *smart*BASIC source code.

A typical look and feel of smartBASIC source fragment in Notepad++ (when configured correctly) is shown in Figure 1. For example, UartCloseEx() is an extension function displayed in a different color from the token function which is part of the core language.

```
'//*****************************************************************************
'// Handler definitions
'//*****************************************************************************

'//===========================================================================
'// Uart Inactivity timer handler
'//===========================================================================
function handlerUartTimer() as integer
  dim rc
  '//Close the uart, and set up TX/RX/RTS lines as gpio and for a hi-lo transition
  '//on the RX line to be detected
  if UartCloseEx(1) == 0 then
    rc=GpioSetFunc(21,2,1)    '//TX  - set high on default
    rc=GpioSetFunc(23,2,0)    '//RTS - set low by default
    rc=GpioSetFunc(22,1,2)    '//RX  - Pull high input & irq on hi2lo transition
    rc=GpioAssignEvent(UART_GPIO_ASSIGN_CHANNEL,22,1)
    if rc != 0 then
      print "\nGpioAssignEvent() Failed"
    endif
  endif
endfunc 1

'//===========================================================================
'// Delay before uart is opened
'//===========================================================================
function handlerOpenDelay() as integer
  dim rc
  '// free up the level transition detection
  rc=GpioUnAssignEvent(UART_GPIO_ASSIGN_CHANNEL)
  '//Open the uart
  rc=UartOpen(9600,0,0,"CN81H")
  '//send an ack character
  print "!"
endfunc 1
```

Figure 1: Standard smartBASIC source code with color syntax highlighting

REQUIREMENTS

- Windows PC (XP or newer) with Notepad++ version 6.4.5 or newer installed. You can install it by downloading it from http://notepad-plus-plus.org

*SMART*BASIC NOTEPAD++ SYNTAX SETUP

To configure Notepad++ for color syntax highlighting for the first time perform the following actions:

1. If necessary, install Notepad++ (version 6.4.5 or newer).
2. From the BL600 firmware upgrade zip file copy the file smartBASIC(Notepad++).xml to the desktop (it will be deleted in the final step).
3. Start Notepad++.
4. If Notepad++ has already been configured for *smart*BASIC color syntax highlighting, then *smart*BASIC must be removed. To do this, follow these steps:
   a. Select **Language > Define your language**.
   b. In the User Language dropdown menu, select **smartBASIC**.
   c. Click **Remove**.
   d. Close the User Defined Language dialog box.
5. Close Notepad++.
6. Start Notepad++.
7. Close all current (non-empty) Notepad++ documents.
8. From the top menu, select **Language > Define your language** to open the User Defined Language vW.X.Y.Z window.
9. Click **Import**.
10. From the new popup screen, find and select the *smartBASIC(Notepad ++).xml* file in the folder saved in step 2.
11. Click **Open**.
12. Close the remaining popup screen and then close Notepad++.
13. Reopen Notepad++.
14. From the top menu, select **Language > smartBASIC**.

    You can now delete the "smartBASIC (Notepad++).xml" file from your desktop

Loading a .sb or a .sblib file from the sample apps folder in the firmware zip file should now result in the source code being displayed with color syntax highlighting.

*SMART*BASIC NOTEPAD++ SYNTAX EDITING

1. Start Notepad++.
2. From the taps on the top select **Language > Define your language**.
3. From the new window, select **User Language > smartBASIC**.

   You are now able to edit your syntax as needed.

Further Information

Further information relating to firmware and the use of UWTerminal is available from the Laird website at http://www.lairdtech.com/Products/Embedded-Wireless-Solutions/Bluetooth-Radio-Modules/BL600-Series