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:

- 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 that the code has been written for. *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 it to display *smart*BASIC source code which is easy to develop and edit.

A typical look and feel of smartBASIC source fragment in Notepad++ when configured appropriately 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.

```
57  '//****************************************************************************
58  '// Handler definitions
59  '//****************************************************************************
60
61  '//===========================================================================
62  '// Uart Inactivity timer handler
63  '//===========================================================================
64  function handlerUartTimer() as integer
65    dim rc
66    '//Close the uart, and set up TX/RX/RTS lines as gpio and for a hi-lo transition
67    '//on the RX line to be detected
68    if UartCloseEx(1) == 0 then
69      rc=GpioSetFunc(21,2,1)   '//TX  - set high on default
70      rc=GpioSetFunc(23,2,0)   '//RTS - set low by default
71      rc=GpioSetFunc(22,1,2)   '//RX  - Pull high input & irq on hi2lo transition
72      rc=GpioAssignEvent(UART_GPIO_ASSIGN_CHANNEL,22,1)
73      if rc != 0 then
74        print "\nGpioAssignEvent() Failed"
75      endif
76    endif
77  endfunc 1
78
79  '//===========================================================================
80  '// Delay before uart is opened
81  '//===========================================================================
82  function handlerOpenDelay() as integer
83    dim rc
84    '// free up the level transition detection
85    rc=GpioUnAssignEvent(UART_GPIO_ASSIGN_CHANNEL)
86    '//Open the uart
87    rc=UartOpen(9600,0,0,"CN81H")
88    '//send an ack character
89    print "!"
90  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

SMARTBASIC NOTEPAD++ SYNTAX SETUP

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

1. This assumes that Notepad++ version 6.4.5 or newer is installed.
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 smartBASIC color syntax highlighting then it needs to be removed. This can be done by selecting **Language->Define your language** and in the dialog box that pops up, in the dropdown box labeled 'User Language:' select smartBASIC which will result in a '**Remove**' button being displayed. Click the '**Remove**' button and acknowledge it in the new pop-up dialog box that results. Close the 'User Defined Language' dialog box.
5. Close Notepad++
6. Start Notepad++
7. Close all non-empty documents.
8. From the top menu select **Language->Define your language** and the popup screen titled 'User Defined Language vW.X.Y.Z' should open.
9. Press the button called "**Import**" from the popup screen and a new popup screen will open.
10. From the new popup screen find the "smartBASIC(Notepad ++).xml" file in the folder saved at step (2) and choose it, then click the "Open" button
11. Close the remaining popup screen and then close Notepad++
12. Reopen Notepad++
13. From the top menu select **Language->smartBASIC**
14. Finished. 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.

SMARTBASIC NOTEPAD++ SYNTAX EDITING

1. Start Notepad++
2. From the taps on the top select **Language->Define your language** and a popup screen should open
3. From the new popup screen on the top left where it says "User language:" select "smartBASIC"
4. Now you can edit the syntax to your likings

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