UWTerminal *smartBASIC*

Quick Start Guide v1.1

Introduction

While UWTerminal is primarily a terminal program for communicating with a Laird module, you can run *smart* BASIC programs from within the UWTerminal program itself to automate terminal emulation functionality. This allows users to become familiar with the fundamentals of *smart* BASIC without the need for any additional hardware. However, you will not be able to include any module functionality such as GPIO or Bluetooth commands.

Requirements

- Windows XP, or later PC
- UWTerminal v6.21, or later
- Text Editor of your choice
- *smart* BASIC User Manual – Release v1.0 or later

Note:
If using the BL600 Dev Kit please refer to the DVK-BL600 Series – Hardware User Manual for additional requirements.

Writing a *smart*BASIC Program

From within your preferred text editor, write a simple *smart* BASIC program; for example, a simple for/next loop to print "Hello world" to the screen 5 times.

Save the program as a text file using a ".sb" extension to the same folder where UW terminal is to be located.

```
dim n
for n = 1 to 5
 print "Hello world\n"
next
```

UWTerminal Setup

If you haven't already, extract the UWTerminal to a folder on your PC. It is a standalone executable and does not need installing.

Open UWTerminal and on the opening screen click **Accept**.

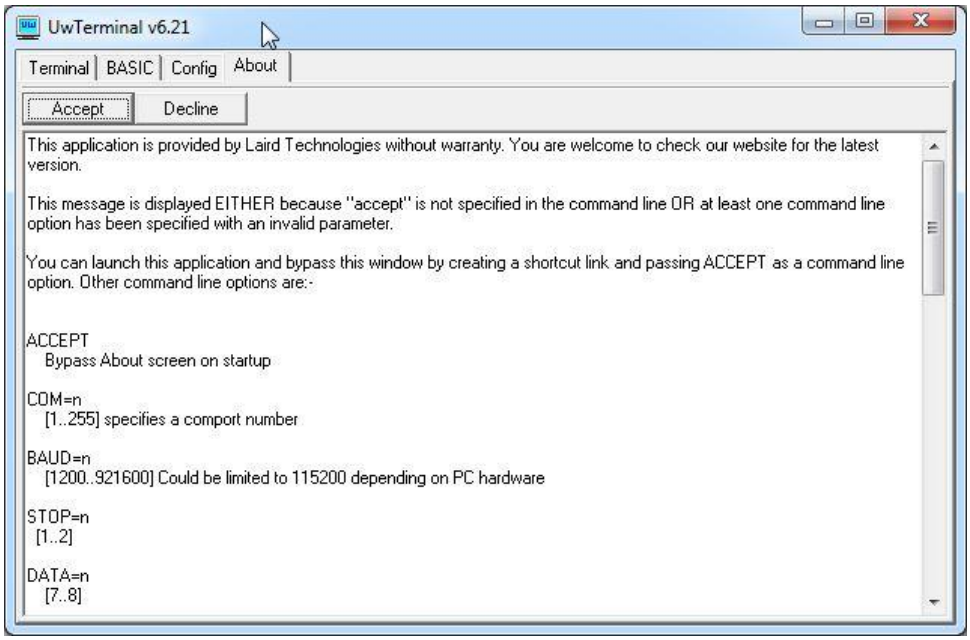

Figure 1: UWTerminal start screen

Before you can access the "BASIC" tab you must first configure a COM port, even if the COM port is not being used. Once the COM port is configured, click **OK**. Note that the COM port defined is not actually used so the values are not important, but be aware that it has to exist on the PC.

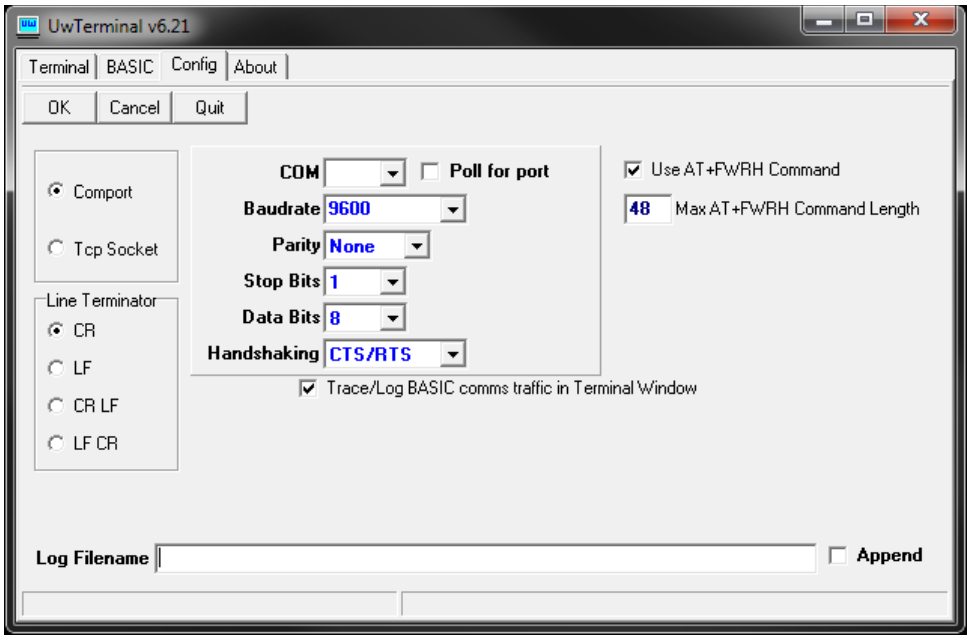

Figure 2: Configuring the COM port

You can now click the "BASIC" tab, which takes you to a blank screen.

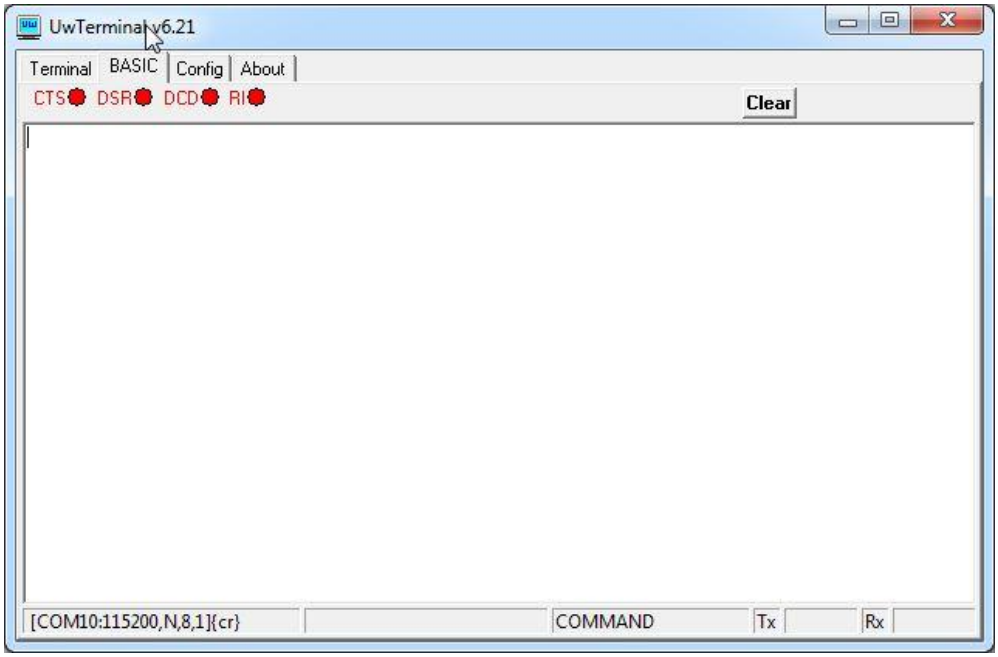

Figure 3: Blank BASIC tab

Compiling your *smart*BASIC program

You can check the spelling and location of your *smart* BASIC program using the command "at+dir" followed by carriage return, which will list all of the files in the folder where UWTerminal resides. In the example below we are looking for the "HelloWorld.sb" program previously saved from the text editor.

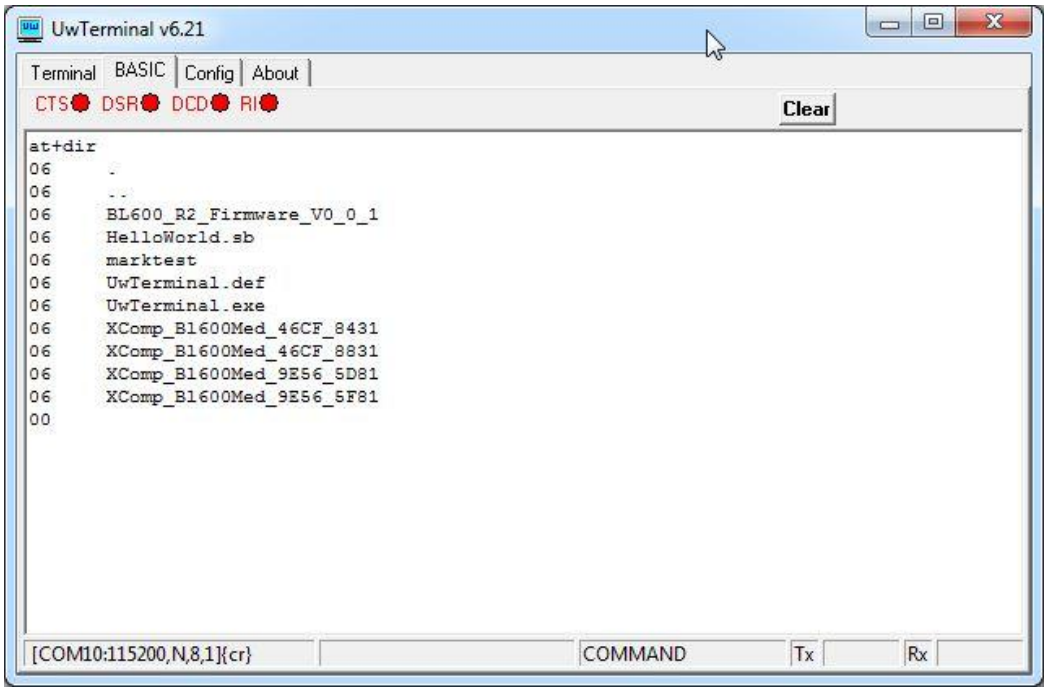

Figure 4: Checking the location of the programs

Type the command "run" followed by the filename of your *smart* BASIC program as in the example below:

run "Helloworld"

The program compiles and runs automatically after you see the compile end message.

"00" indicates a successful command.

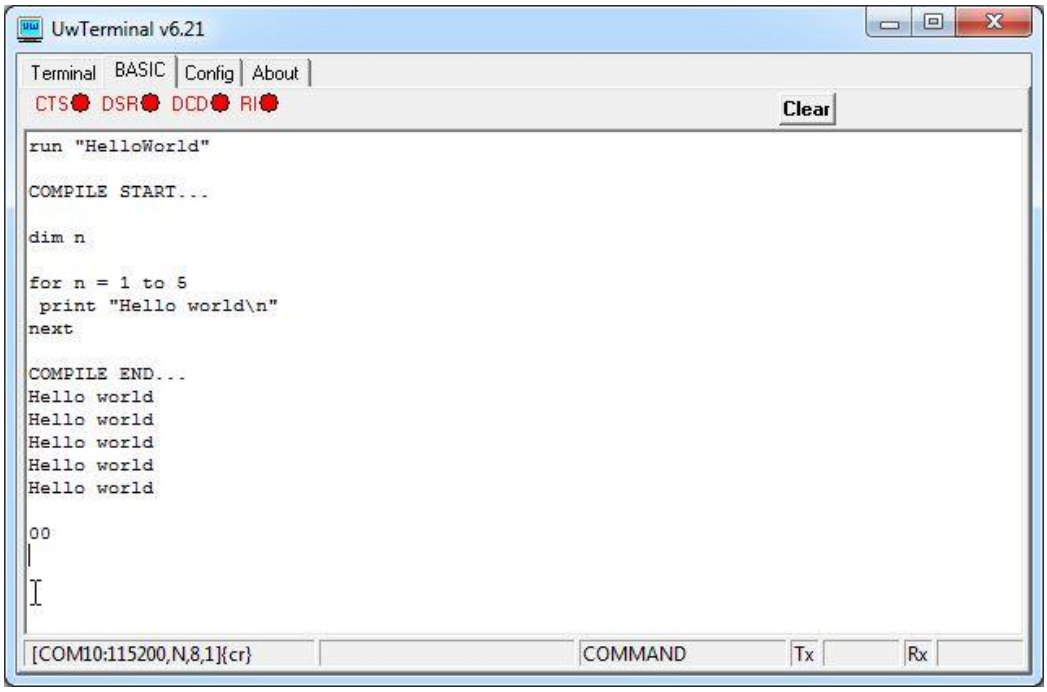

Figure 5: Compiling and successfully running your smart BASIC program

Running your *smart*BASIC program

If a program is already compiled, you can run the program directly using "run". In the example below the same command used to compile the program was used again.

run "Helloworld"

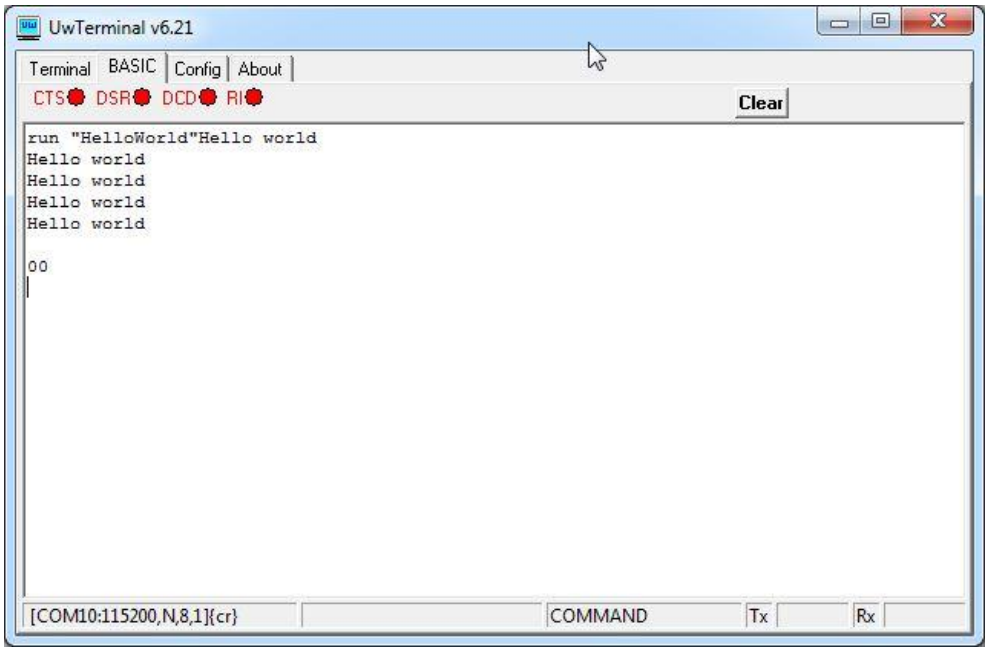

Figure 6: Running a pre-compiled smart BASIC program

"00" printed to screen means the action was successful.

Resources

For more information on programming with *smart* BASIC, please refer to the *smart* BASIC User Manual and other documentation on the BL600 product page.