



Running the UART example from the LPC1700 Code Bundle from NXP (for Keil MDK-ARM)

Product Line Microcontrollers
Business Line Standard ICs

Contents

- ▶ [Prerequisites for running this example](#)
- ▶ [Example description](#)
- ▶ [Running the UART example from the LPC1700 Code Bundle using the ULINK2 or ULINK-ME JTAG debugger](#)
- ▶ [Running the UART example from the LPC1700 Code Bundle using the FlashMagic ISP software](#)

Prerequisites for running this example

- ▶ There are several hardware and software tools necessary in order to run this example:
 - MCB1760 or MCB1750 Evaluation Board from Keil
 - LPC1700 Code Bundle software package
 - PC with 2 USB ports, serial port, and 125MB of available hard disk space
 - Keil MDK-ARM IDE (free eval version is ok)
 - FlashMagic In-System Programming software – OR –
ULINK2 / ULINK–ME JTAG debugger
 - Terminal program (e.g. TeraTerm, Hyperterminal, etc.); serial cable
- ▶ Please refer to the LPC1700 Sample Code Bundle Introduction / Setup file (code.bundle.lpc17xx.keil.intro.setup.pdf) for more details on setting up the hardware and software environment
 - The file can be found here:
<http://www.standardics.nxp.com/support/software/code.bundle.lpc17xx.keil/>

Example Description

- ▶ The UART project is a simple program for the NXP LPC17xx microcontrollers using Keil 'MCB1700' evaluation board
- ▶ When sending some characters from a terminal program on the PC at a speed of 57600 Baud to the LPC17xx UART0 or UART1 the LPC17xx will echo those same characters back to the terminal program

**Running the UART example from
the LPC1700 Sample Code Bundle
using the ULINK2 or ULINK-ME
JTAG debugger**

UART example using JTAG debugger

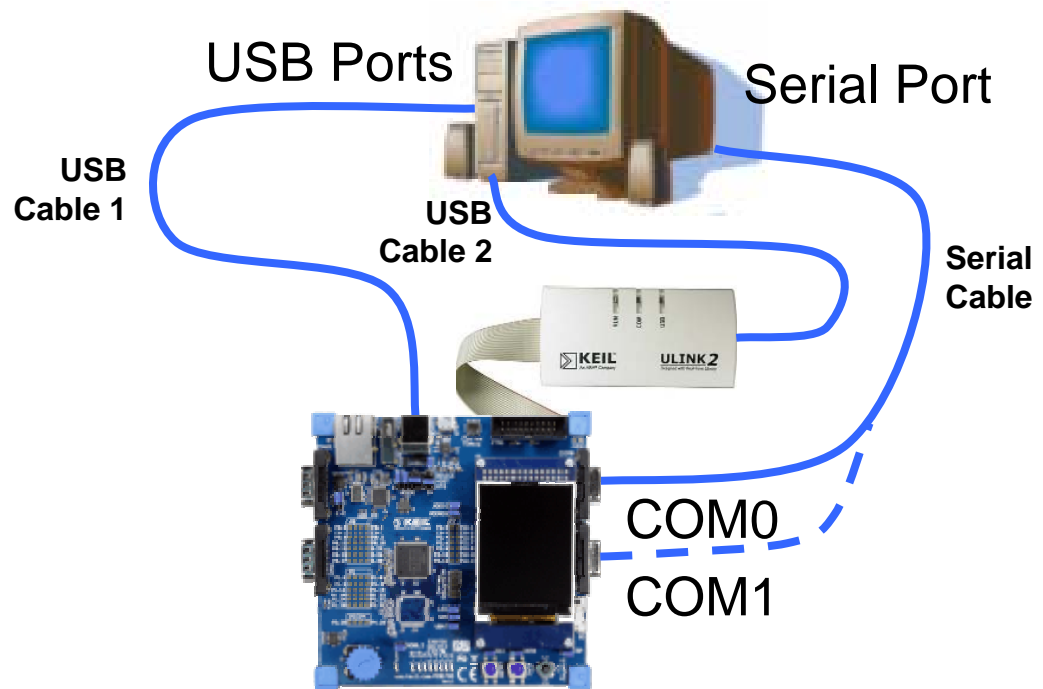
Step 1: Hardware Setup

- ▶ Make sure your computer and your board are connected as shown
- ▶ This powers the board, provides output from the UART0/1 to the PC, and provides JTAG debugging / flash programming

Note: Remove the RST and ISP jumpers on the MCB1700 board:



Note: instead of ULINK2 you can also use ULINK-ME:



UART example using JTAG debugger

Step 2: Opening up the Keil uVision3 project

- ▶ Open the project by double-clicking on the following file here:
`..\LPC17xxSampleSoftware.xxx\Keil\UART\uart.uv2`
 - Note: Ignore the ‘invalid path’ warning if you see one (just click ‘OK’)
- ▶ Familiarize yourself with the following files:
 - uart.c: contains the UART0 / UART1 handlers / driver functions
 - uarttest.c: contains a small test program utilizing the UART driver
 - system_LPC17xx.c: Cortex-M3 Device Peripheral Access Layer Source File (CMSIS-compliant)
 - startup_LPC17xx.s: CMSIS Cortex-M3 Core Device Startup File (CMSIS-compliant)
 - Abstract.txt: Describes what the uarttest.c program does

UART example using JTAG debugger

Step 3: Compiling the example project

- ▶ Make sure the uVision3 Target setting is 'FLASH'



- ▶ Build the project clicking the “Rebuild all target files”-button



- ▶ Example should build without errors or warnings

```
Build target 'FLASH'
assembling startup_LPC17xx.s...
compiling system_LPC17xx.c...
compiling uart.c...
compiling uarttest.c...
linking...
Program Size: Code=1616 RO-data=304 RW-data=24 ZI-data=736
FromELF: creating hex file...
".\FLASH\UART.axf" - 0 Error(s), 0 Warning(s).
```


UART example using JTAG debugger

Step 4: Programming the code to on-chip Flash

- ▶ Select Flash as the target (if not already done)



- ▶ Make sure JTAG debugger is connected to PC and board

- ▶ Click on the LOAD button to start download



- ▶ Verify that the download is ok (in the Build window)

Erase Done.

Programming Done.

Verify OK.

- ▶ Please note that the code is not running yet...
- ▶ Make sure the serial cable is connected to the PC and to COM0 (=UART0) or COM1 (UART1) on the MCB1700 board

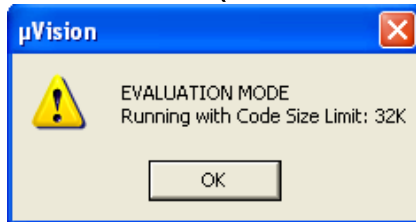
UART example using JTAG debugger

Step 5: Starting a Debugging session

- ▶ Click “Debug > Start/Stop Debug Session”



- ▶ Click OK here (Evaluation Version Message)



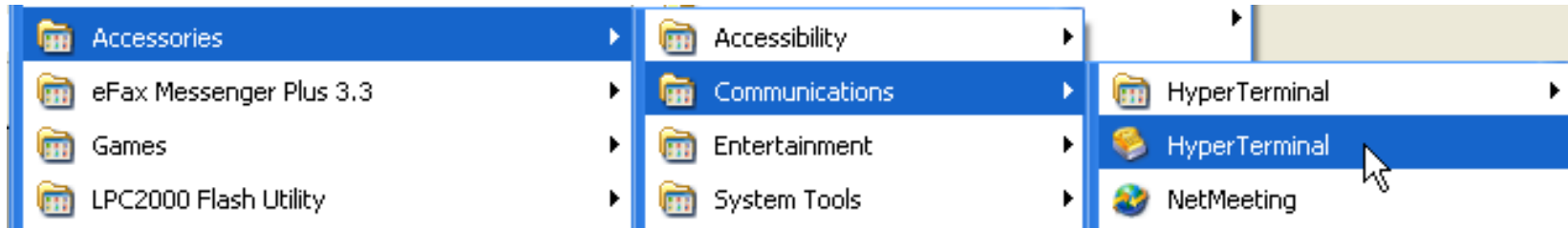
- ▶ The code will stop at main() function in file uarttest.c:

```
21  ** Main Function main()
22  This program has been test on Keil LPC1700 board.
23  ****
24  int main (void)
25  {
26      SystemInit();
27
28      UARTInit(0, 57600); /* baud rate setting */
29      UARTInit(1, 57600); /* baud rate setting */
30
31      ...
```

UART example using JTAG debugger

Step 6: Establishing a Terminal connection


- ▶ Start HyperTerminal by clicking “START > All Programs > Accessories > Communications > HyperTerminal”

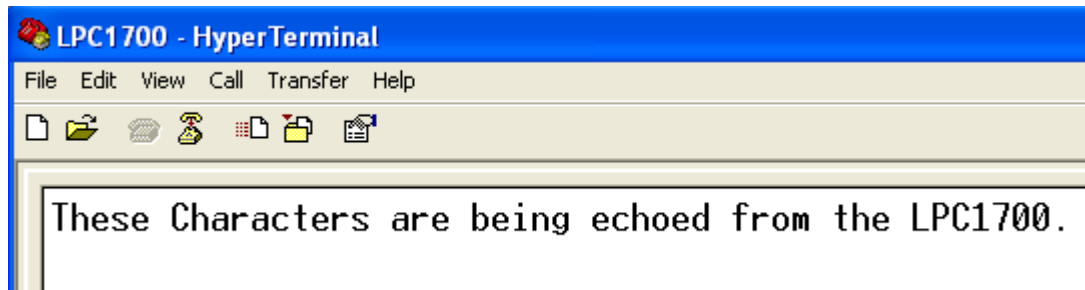


- ▶ Connect HyperTerminal to the serial port (COM1)
- ▶ UART settings: baud rate **57600**, 8N1, no flow control.
- ▶ Note: You can also use a different terminal program (e.g. TeraTerm) for this (@ 57600 Bd).

UART example using JTAG debugger

Step 7: Running the example


- ▶ Click Run Button to start code execution 
- ▶ In the HyperTerminal Window, type some text:




- ▶ Switch the serial cable to the other COM port on the MCB1700 board
 - The same result happens as above

UART example using JTAG debugger

Step 8: Setting a breakpoint

- ▶ Click Stop Button to stop code execution 
- ▶ Set a breakpoint in file uarttest.c on line 42 by double-clicking it



```
40     if ( UART1Count != 0 )
41     {
42         UART1->IER = IER_THRE | IER_RLS;           /* Disable RBR */
43         UARTSend( 1, (uint8_t *)UART1Buffer, UART1Count );
44         UART1Count = 0;
45         UART1->IER = IER_THRE | IER_RLS | IER_RBR; /* Re-enable RBR */
```

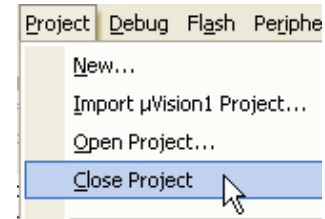
- ▶ Run the program 
- ▶ Now whenever a character is received on UART1, the breakpoint is hit and the program execution will stop

```
40     if ( UART1Count != 0 )
41     {
42         UART1->IER = IER_THRE | IER_RLS;           /* Disable RBR */
43         UARTSend( 1, (uint8_t *)UART1Buffer, UART1Count );
44         UART1Count = 0;
45         UART1->IER = IER_THRE | IER_RLS | IER_RBR; /* Re-enable RBR */
```

UART example using JTAG debugger

Step 9: Closing the debugging session

- ▶ Click Stop Button to stop code execution (if running) 
- ▶ Click  to stop Debug session
- ▶ Go to “Project > Close Project” to close the project
- ▶ Click “File > Exit” to close uVision3 (if desired)



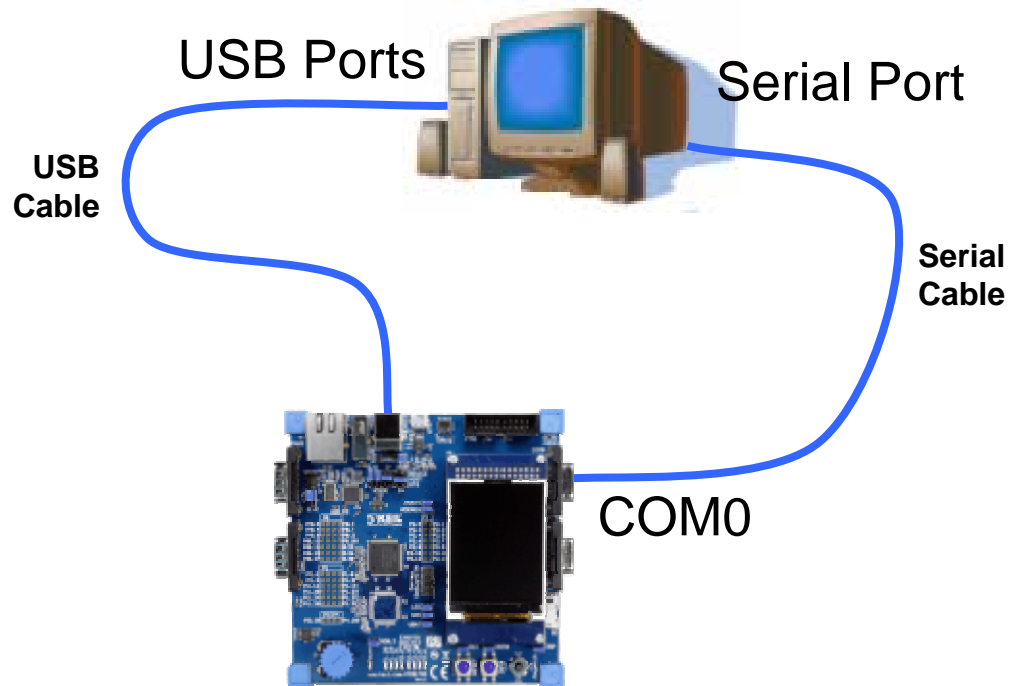
**Running the UART example from
the LPC1700 Sample Code Bundle
using the FlashMagic ISP software**

UART example using FlashMagic

Step 1: Hardware Setup

- ▶ Make sure your computer and your board are connected as shown
- ▶ This powers the board, provides output from the UARTs to the PC, and provides for ISP programming

Note: Make sure the RST and ISP jumpers are **closed** on the MCB1700 board:



UART example using FlashMagic

Step 2: Opening up the Keil uVision3 project

- ▶ Open the project by double-clicking on the following file here:
`..\LPC17xxSampleSoftware.xxx\Keil\UART\uart.uv2`
 - Note: Ignore the ‘invalid path’ warning if you see one (just click ‘OK’)
- ▶ Familiarize yourself with the following files:
 - uart.c: contains the UART0 / UART1 handlers / driver functions
 - uarttest.c: contains a small test program utilizing the UART driver
 - system_LPC17xx.c: Cortex-M3 Device Peripheral Access Layer Source File (CMSIS-compliant)
 - startup_LPC17xx.s: CMSIS Cortex-M3 Core Device Startup File (CMSIS-compliant)
 - Abstract.txt: Describes what the uarttest.c program does

UART example using FlashMagic

Step 3: Compiling the example project

- ▶ Make sure the uVision3 Target setting is 'FLASH'



- ▶ Build the project clicking the “Rebuild all target files”-button



- ▶ Example should build without errors or warnings

```
Build target 'FLASH'
assembling startup_LPC17xx.s...
compiling system_LPC17xx.c...
compiling uart.c...
compiling uarttest.c...
linking...
Program Size: Code=1616 RO-data=304 RW-data=24 ZI-data=736
FromELF: creating hex file...
".\FLASH\UART.axf" - 0 Error(s), 0 Warning(s).
```

- ▶ As you can see, a .hex file is created in the progress...

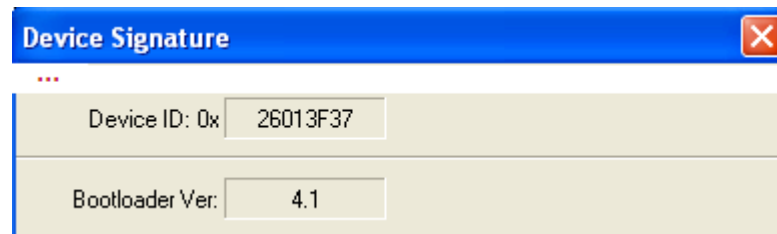
- If not, check the 'Create HEX File' box under
'Project > Options for Target 'FLASH' > Output':



UART example using FlashMagic

Step 4: Setting up FlashMagic

- ▶ Make sure no other program is using the serial port (e.g. TeraTerm, Hyperterminal, etc.)
- ▶ Make sure the serial cable is connected to the PC and to COM0 (=UART0) on the MCB1700 board
- ▶ Launch FlashMagic application
- ▶ To make sure you're connected to the LPC1768, click "ISP > Read Device Signature" and look for the following response:



- If you get an error message instead, check your connections...

UART example using FlashMagic

Step 5: Programming the code to on-chip Flash

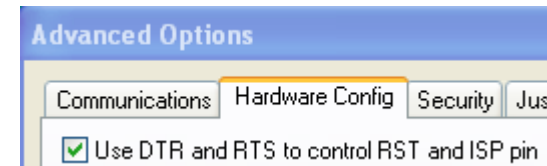
- ▶ Make the following selections:



Step 1 - Communications

Device:	LPC1768
COM Port:	COM 1
Baud Rate:	19200
Interface:	None (ISP)
Oscillator (MHz):	12.000000

- ▶ Go to “Options > Advanced Options > Hardware Config”
 - Make sure the following box is checked:



Advanced Options

Communications Hardware Config Security Jus

☒ Use DTR and RTS to control RST and ISP pin

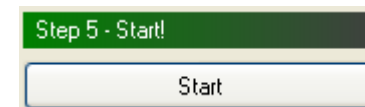
- ▶ Use the “Browse” button to select the UART project hex file:
... \LPC17xxSampleSoftware.xxx\keil\UART\FLASH\UART.hex



Step 3 - Hex File

Hex File:

- ▶ Click Start!
 - FlashMagic programs the selected hex file into the LPC1768

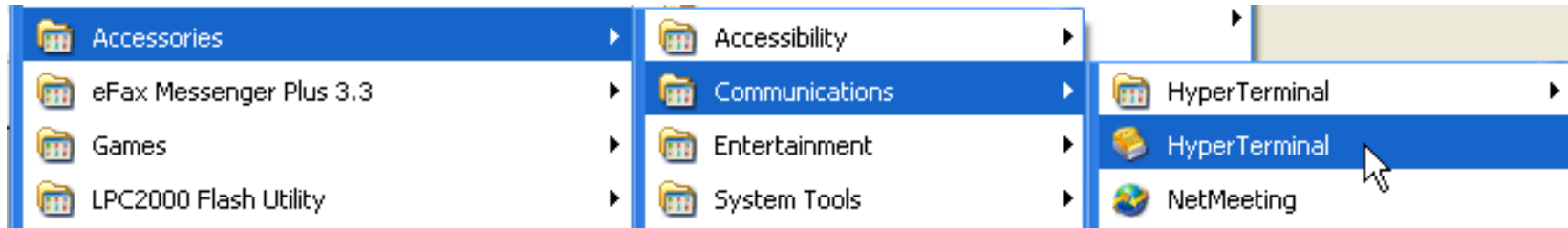


Step 5 - Start!

UART example using FlashMagic

Step 6: Establishing a Terminal connection

- ▶ Start HyperTerminal by clicking “START > All Programs > Accessories > Communications > HyperTerminal”

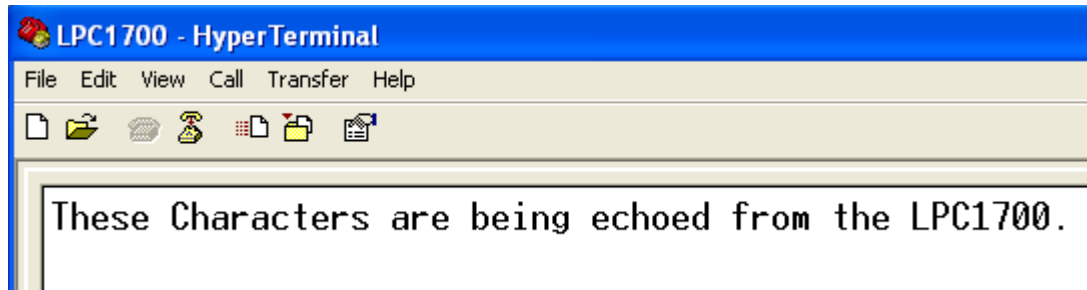


- ▶ Connect HyperTerminal to the serial port (COM1)
- ▶ UART settings: baud rate **57600**, 8N1, no flow control.
- ▶ Note: You can also use a different terminal program (e.g. TeraTerm) for this (@ 57600 Bd).

UART example using FlashMagic

Step 7: Running the example

- ▶ Remove the RST and ISP jumpers on the MCB1700:
- ▶ Press Reset Button to start code execution
- ▶ In the HyperTerminal Window, type some text:



- ▶ Switch the serial cable to the other COM port on the MCB1700 board
 - The same result happens as above

