LPC32X0 Serial Loader Tool

1 Introduction

The LPC32x0 serial loader (LSL) is a Windows tool that connects to the LPC32x0 through a serial port. This tool communicates with the LPC32x0 when the LPC32x0 is reset and provides a method to get code to the chip and board without a JTAG device, or even if the board boot devices have been erased.

1.1 Using the LPC3250 serial loader (LSL) tool

The LSL is meant to be used on various LPC devices and all of its functionality may not apply to the LPC32x0. *As of this time, only the primary boot (IRAM) option is supported with the LSL*

Using the LSL to load and start code requires have a binary executable linked to load at start at address 0x00000000. The total linked image size should not exceed 56K and the executable image's memory usage should not exceed the amount of internal RAM in the LPC32x0 device.

The LPC32x0 and LSL interact as shown in the following table. If the '5' sent from the LPC32x0 is not responded to within a short period of time, the LPC32x0 will try another boot method.

LPC32x0	LPC3250 serial loader
LPC32x0 is reset	
LPC32x0 sends a '5' on UART5.	LSL responds with a 'A'.
LPC32x0 sends a '5' on UART5.	LSL responds with a 'U'.
	LSL sends the address to the board (0x00000000)
	LSL sends the image size to the board in bytes
	LSL sends the code to the board
LPC32x0 boots loaded code at address 0x00000000	

There is a configuration file called LPC3250_loader.ini that allows the default values to be setup for the various LSL settings. The format of the file is shown below:

```
[BINFILES]
PRIMARY=<FILENAME>, 0x00000000
SECONDARY=NOTUSED, 0x00000000
FLASH=NOTUSED, 0x00000000
PG_SIBL=NOTUSED, 0x00000000

[IMGFILES]
UBOOT=NOTUSED, 0x00000000
ROOTFS=NOTUSED, 0x00000000
```

KERNEL=NOTUSED, 0x00000000

Currently, the only usable function in the configuration file is the [BINFILES]/PRIMARY entry, which defines the image to boot and its load address.

1.2 Using the LPC3250 serial loader (LSL) tool

The following steps show how to use the LSL tool.

1.2.1 Build the binary

The binary to boot must be linked to load and execute at address 0x0, be less than 56K in size, and use only up to the amount of memory that the LPC32x0 device provides. The binary should be placed in the same location as the LSL executable.

1.2.2 Edit the configuration file

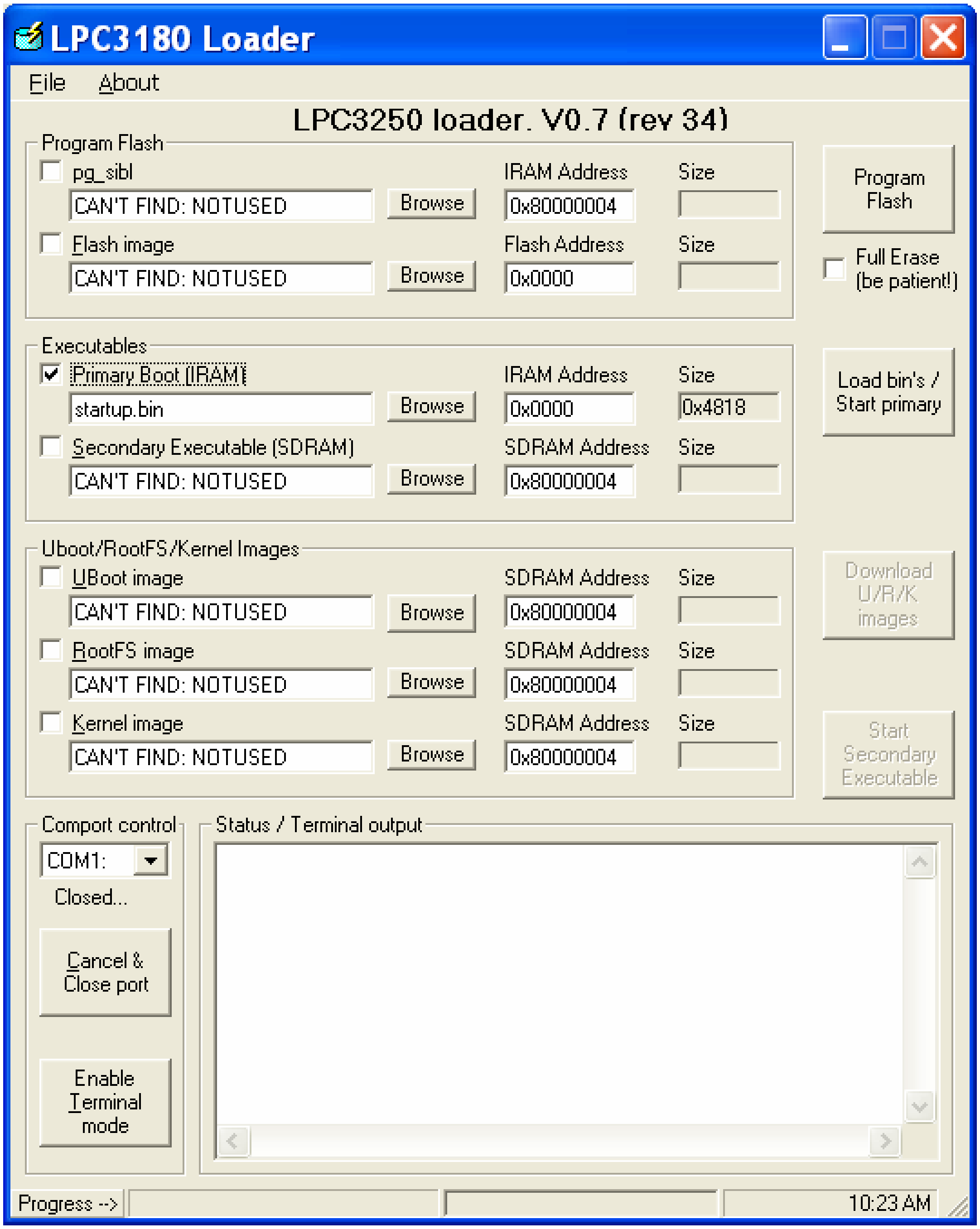
Modify the LPC3250_loader.ini file's PRIMARY entry to match the name of your binary and its load address (0x00000000).

1.2.3 Serial cable connection

Connect a serial cable between the PC running LSL and the board with the LPC32x0 device. The LPC32x0 device should be using UART5.

1.2.4 Start LSL

Start LSL and setup for the correct COM port on the PC. Press the Load bin/Start primary button on the LSL window once everything is setup.



LPC3250 loader. V0.7 (rev 34)

Program Flash

□ pg_sibl: CAN'T FIND: NOTUSED; IRAM Address 0x80000004; Size

□ Flash image: CAN'T FIND: NOTUSED; Flash Address 0x0000; Size

□ Full Erase (be patient!)

Executables

☑ Primary Boot (IRAM): startup.bin; IRAM Address 0x0000; Size 0x4818

□ Secondary Executable (SDRAM): CAN'T FIND: NOTUSED; SDRAM Address 0x80000004; Size

Uboot/RootFS/Kernel Images

□ UBoot image: CAN'T FIND: NOTUSED; SDRAM Address 0x80000004; Size

□ RootFS image: CAN'T FIND: NOTUSED; SDRAM Address 0x80000004; Size

□ Kernel image: CAN'T FIND: NOTUSED; SDRAM Address 0x80000004; Size

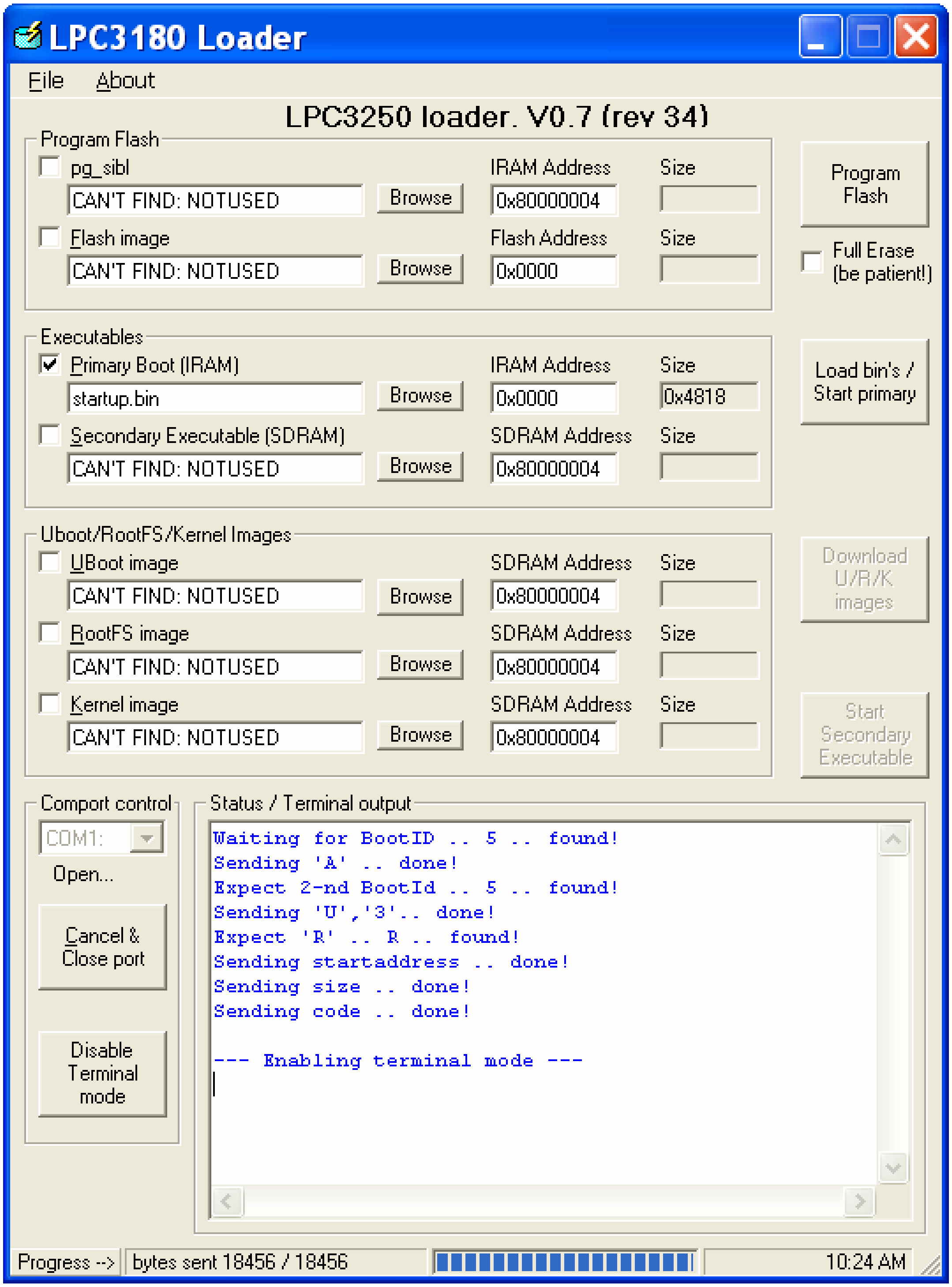
Comport control: COM1:

Closed...

Status / Terminal output

1.2.5 Boot the LPC32x0 based board

Reset the board with the LPC32x0 device and the image will load and run via LSL. After the image has completed loading, LSL will switch to terminal mode.



LPC3250 loader. V0.7 (rev 34)

Program Flash

- □ pg_sibl: CAN'T FIND: NOTUSED — IRAM Address 0x80000004 — Size
- □ Flash image: CAN'T FIND: NOTUSED — Flash Address 0x0000 — Size

Executables

- ☑ Primary Boot (IRAM): startup.bin — IRAM Address 0x0000 — Size 0x4818
- □ Secondary Executable (SDRAM): CAN'T FIND: NOTUSED — SDRAM Address 0x80000004 — Size

Uboot/RootFS/Kernel Images

- □ UBoot image: CAN'T FIND: NOTUSED — SDRAM Address 0x80000004 — Size
- □ RootFS image: CAN'T FIND: NOTUSED — SDRAM Address 0x80000004 — Size
- □ Kernel image: CAN'T FIND: NOTUSED — SDRAM Address 0x80000004 — Size

Comport control: COM1:

```
Waiting for BootID .. 5 .. found!
Sending 'A' .. done!
Expect 2-nd BootId .. 5 .. found!
Sending 'U','3'.. done!
Expect 'R' .. R .. found!
Sending startaddress .. done!
Sending size .. done!
Sending code .. done!

--- Enabling terminal mode ---
```

Progress --> bytes sent 18456 / 18456 10:24 AM