

BlueBoard-LPC11U24/37

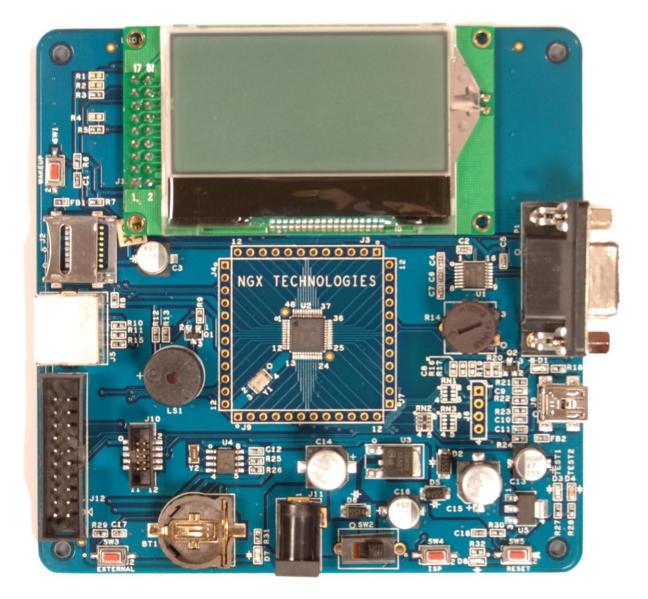


Fig. 1



About NGX Technologies

NGX Technologies is a premier supplier of development tools for the ARM7, ARM Cortex M0, M3 and M4 series of microcontrollers. NGX provides innovative and cost effective design solutions for embedded systems. We specialize in ARM MCU portfolio, which includes ARM7, Cortex-M3, M0 & M4 microcontrollers. Our experience with developing evaluation platforms for NXP controller enables us to provide solutions with shortened development time thereby ensuring reduced time to market and lower development costs for our customers. Our cost effective and feature rich development tool offering, serves as a testimony for our expertise, cost effectiveness and quality.

Contact Information:

NGX Technologies Pvt. Ltd. No.216, 5th main Road, R.P.C. Layout, Vijayanagar 2nd Stage, Bangalore – 560 104 Phone : +91-80-40925507 email:sales@ngxtechnologies.com

CE certification

NGX Technologies BLUEBOARD-LPC11U24/37 board have been tested for radiated emission as per EN55022 class A standard. The device is under the limits of the standard EN55022 class A and hence CE marked. No other test have been conducted other than the radiated emission (EN55022 class A standard). The device was tested with the ports like USB, Serial, and Power excluding the GPIO ports. Any external connection made to the GPIO ports may alter the EMC behaviour. Usage of this device under domestic environment may cause unwanted interference with other electronic equipment's. User is expected to take adequate measures. The device is not intended to be used in and end product or any subsystem unless the user re-evaluates applicable directive/conformance.



Table of Contents

| 1.0 INTRODUCTION | 4 |
|--|----|
| 2.0 BLUEBOARD-LPC11U24/37 Development Tool Setup | |
| 2.1 IDE and debugger | 5 |
| 2.2 Installation & Configuration of KEIL software | 5 |
| 2.3 Setup for ULINK2 and BLUEBOARD-LPC11U24/37 Board | 9 |
| 2.4 Configuration of ULINK2 Debugger | |
| 3.0 BLUEBOARD-LPC11U24/37 Software Development | |
| 3.1 Executing the sample projects | |
| 3.2 Creating New project | 14 |
| 3.3 Creating Bin File | |
| 4.0 BLUEBOARD-LPC11U24/37 Programming | 20 |
| 4.1 Programming options | 20 |
| 4.1.1 On-Chip bootloader (USB or UART) | |
| 4.1.2 Flashing the board using USB | |
| 4.2 Flashing the Hex file through UART | |
| 5.0 Schematic & Board Layout | 23 |
| 5.1 Schematic | 23 |
| 5.2 Board layout | 23 |
| 6.0 Trouble Shooting | 24 |
| 7.0 CHANGE HISTORY | 25 |
| 7.1 Change History | 25 |
| 8.0 REFERENCES | |



1.0 INTRODUCTION

This document is the User Manual for the BLUEBOARD-LPC11U24/37, a low cost ARM Cortex-M0 based board by NGX Technologies. This document reflects its contents which include system setup, debugging, and software components. This document provides detailed information on the overall design and usage of the board from a systems perspective.

Before proceeding further please refer the quick start guide for BLUEBOARD-LPC11U24/37 features and BLUEBOARD-LPC11U24/37 verification.

For BLUEBOARD-LPC11U24/37 Quick Start Guide: Click here.

For the most updated information on the BLUEBOARD-LPC11U24 board please refer to NGX'website.

For the most updated information on the BLUEBOARD-LPC11U37 board please refer to NGX'website.



2.0 BLUEBOARD-LPC11U24/37 Development Tool Setup

2.1 IDE and debugger

As mentioned in the earlier section, NGX's MCU evaluation platforms are not coupled tightly with any one particular combination of IDE and debugger. The following sections will explain the setup for KEIL and ULINK as the IDE and debugger respectively.

2.2 Installation & Configuration of KEIL software

The Installation of KEIL software is explained below:

Note: We have used **Keil uvision version 4.23** while creating the User manual for this evaluation kit. Please ensure that you are using uvision version 4.23 or above.

Step 1: Open the keil setup

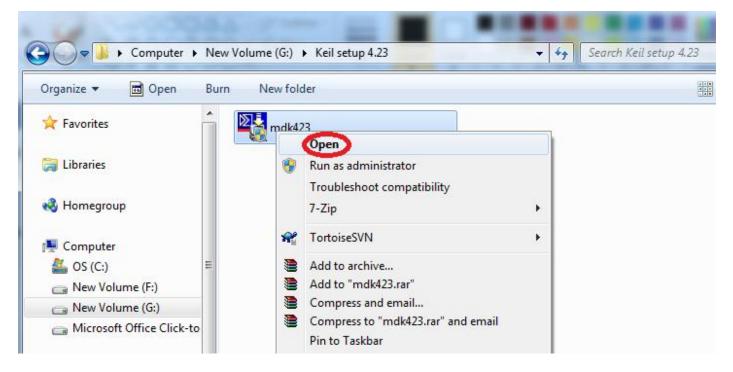


Fig. 2

User Manual: BlueBoard-LPC11U24/37



Step 2: Keil µvision4.23 information Click on Next

| Setup MDK-ARM V4.23 | × |
|--|--------------------------|
| Welcome to Keil μVision Release 12/2011 | |
| This SETUP program installs: MDK-ARM V4.23 This SETUP program may be used to update a previous pr However, you should make a backup copy before proceed It is recommended that you exit all Windows programs befor Follow the instructions to complete the product installation. | ling. |
| — Keil µVision4 Setup ———————————————————————————————————— | << Back Next >> [Cancel] |

Fig. 3

Step 3: Terms & conditions

| Setup MDK-ARM V4.23 |
|--|
| License Agreement Please read the following license agreement carefully. |
| To continue with SETUP, you must accept the terms of the License Agreement. To accept the agreement, click the check box below. |
| END USER LICENCE AGREEMENT FOR MDK-ARM THIS END USER LICENCE AGREEMENT ("LICENCE") IS A LEGAL AGREEMENT BETWEEN YOU (EITHER A SINGLE INDIVIDUAL, OR SINGLE LEGAL ENTITY) AND ARM LIMITED ("ARM") FOR THE USE OF THE SOFTWARE ACCOMPANYING THIS LICENCE. ARM IS ONLY WILLING TO LICENSE THE SOFTWARE TO YOU ON CONDITION THAT YOU ACCEPT ALL OF THE TERMS IN THIS LICENCE. BY CLICKING "I AGREE" OR BY INSTALLING OR OTHERWISE USING OR COPYING + |
| ✓ If agree to all the terms of the preceding License Agreement — Keil μVision4 Setup << Back Next >> Cancel |

Fig. 4

User Manual: BlueBoard-LPC11U24/37



Step 4: Provide the destination path and Click on Next

| Setup MDK-ARM V4.23 | |
|--|--|
| Folder Selection Select the folder where SETUP will install files. | |
| SETUP will install $\mu Vision4$ in the following folder. | |
| To install to this folder, press 'Next'. To install to a diff folder. Destination Fuldor UC:\Keil | erent folder, press 'Browse' and select another BIowse |
| — Keil µVision4 Setup ———————————————————————————————————— | << Back Next >> Cancel |



Step 5: Fill your Personal information and Click on Next

| Setup MDK-ARM V4.2 | 3 | × |
|---|--|-------------------------------|
| Customer Informal Please enter your | | |
| | name, the name of the company for whom you | work and your E-mail address. |
| First Name: | abc | |
| Company Name: | NGX Technologies Pvt Ltd | / |
| E-mail: — Keil μVision4 Setup | abc@ngxtechnologies.com | |
| | < | K Back Next >>> Cancel |

Fig. 6

Step 5: Click on Next



| | | × |
|---|---------|------------------|
| | Tools t | ™ by ARM |
| | | |
| | | |
| | 1 | |
| • | | |
| | | EXECUTION |

Fig. 7

Step 6: Keil $\mu Vision 4.23$ setup is completed. Click on Finish

| Setup MDK-ARM V4.23 | × |
|---|-----------------|
| Keil µVision4 Setup completed MDK-ARM V4.23 | |
| μVision Setup has performed all requested operations successfully. ✓ Launch Driver Installation: "ULINK Pro Driver V1.0" | |
| ☑ Show Release Notes. | |
| | |
| — Keil µVision4 Setup ———————————————————————————————————— | |
| << Bac | k Finish Cancel |

Fig.8



2.3 Setup for ULINK2 and BLUEBOARD-LPC11U24/37 Board

The BlueBoard-LPC11U24/37 board has on board 20 pin SWD box, the ULINK2 is not a part of the BlueBoard-LPC11U24/37 package, user need to buy separately.

To run the BlueBoard-LPC11U24/37 examples you need the following components and the image shows the each components:

- ULINK2
- BlueBoard-LPC11U24/37 Board
- USB type-B cable



Fig. 9

Connections of components are as shows in the following image.



Fig. 10

The above setup is ready to use for development in Keil IDE



2.4 Configuration of ULINK2 Debugger

The configuration flow of ULINK2 Debugger is explained below:

Step 1: Open the Keil Workspace then by clicking on the **target** option, the window opens as shown below. Next click on Debug option and select the ULINK2 debugger as shown in the image.

| 🔣 Options for Target 'Blinky' | |
|---|--|
| Device Target Output Listing User C/C++ Asm | Linker Debug Utilities |
| C Use Simulator Settings | Use ULINK2/ME Cortex Debugger Settings |
| Load Application at Startup Run to main() Initialization File: | Load Application at Startup Initialization File: VFLASH.ini Edit |
| Restore Debug Session Settings Preakpoints Watch Windows & Performance Analyzer Memory Display | Restore Debug Session Settings Breakpoints Watch Windows Memory Display |
| CPU DLL: Parameter: SARMCM3.DLL | Driver DLL: Parameter: SARMCM3.DLL |
| Dialog DLL: Parameter: DARMP1.DLL pLPC11U24 | Dialog DLL: Parameter: TARMP1.DLL pLPC11U24 |
| ОК Са | ncel Defaults Help |

Fig.11

Step 2: Click on the settings option, the Cortex-M Target Driver Setup window opens then select SW port. After selection of the SW port the ULINK2 detected is as shown in the image below

| Use Simulator | Settings 🔍 Use: ULINK2/ME Cortex Debugger 💌 Settings |
|---|--|
| ortex-M Target Driver Setup | |
| Debug Trace Rash Download | |
| ULINK USB - JTAG/SW Adapter- | SW Device |
| Serial No: V0168AVR | IDCODE Device Name |
| ULINK Version: ULINK2 | SWDIC Ox0BB11477 ARM CoreSight SW-DP |
| Device Family: Cortex-M | Dow |
| Firmware Version: V1.42 | Automatic Detection ID CODE: |
| 💌 SWJ Pitt SW 🔍 | C Manual Conïguration Device Name: |
| Max Clock: 1MHz 💌 | Add Delete Update IR len: |
| Debug | Casha Oritina Davialard Oritina |
| Connect & Reset Options Connect: Normal V Re | set: Autodetect Cache Options Download Options set: Autodetect Verify Code Download |
| Reset after Connect | Cache Memory Developed to Each |

Fig.12



Step 3: Click on Utilities and select ULINK2 Cortex Debugger as shown below

| 🛛 Options for Ta | rget 'Blinky' | × |
|------------------|--|---|
| Device Target | Output Listing User C/C++ Asm Linker Debug Utilities | |
| Configure Flash | n Menu Command | |
| Use Targe | t Driver for Flash Programming | |
| • | 1ULINK2/ME Cortex Debugger 💽 Settings 🔽 Update Target before Debugging | |
| Init File: | Edit | |
| O Use Extern | nal Tool for Flash Programming | |
| Command: | | |
| Arguments: | | |
| | 🗖 Run Independent | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | OK Cancel Defaults Help | |

Fig.13

Step 4: By Clicking on Settings the Cortex-M Target Driver Setup window opens, Click on Add to select the flash as shown below

| Options for Target 'Blinky' | | | | |
|---|--|--|--|-----|
| Device Target Output Listing User C/C++ Asm Configure Rash Menu Command © Use Target Driver for Rash Programming ULINK2/ME Cortex Debugger | Linker Debug Utilities Settings IV Update Target t | before Debugging | | |
| Cortex-M Target Driver Setup Debug Trace Flash Download | Add Flash Programming Algorit | hm | | x |
| Download Function ○ Erase Full Chip ♥ Program ● Erase Sectors ♥ Verify ○ Do not Erase ♥ Reset and Run Programming Algorithm □ □ Description □ □ Device Type □ | Description LM4Fxxx 64kB Flash LPC122x IAP 128kB Flash LPC11xx/13xx IAP 16kB Flash LPC11xx/13xx IAP 24kB Flash LPC11xx/13xx IAP 24kB Flash LPC122x IAP 48kB Flash LPC122x IAP 80kB Flash LPC122x IAP 80kB Flash LPC12x IAP 96kB Flash LPC12x IAP 96kB Flash LPC17xx IAP 256kB Flash LPC17xx IAP 256kB Flash LPC17xx IAP 32kB Flash LPC17xx IAP 512kB Flash LPC17x IAP 512kB Flash LPC17x IAP 512kB Flash LPC17x IAP 512kB Flash LPC17x IAP 512kB Flash | Device Type On-chip Flash On-chip Flash | Device Size 64k 128k 16k 24k 48k 64k 8k 80k 96k 128k 256k 32k 512k 64k 256k | * · |
| | Add | Cancel | | |

Fig.14

Click OK to complete the ULINK2 Debugger configuration



3.0 BLUEBOARD-LPC11U24/37 Software Development

3.1 Executing the sample projects

The sample projects are provided with the available kit.

Steps to execute the sample projects:

Step 1: Open the project folder.

Step 2: Then open the file **project_**name.uvproj eg blinky.uvproj.

| 🔓 C:\Work\ngx\Blueboard_I | LPC11U24\BLUE | BOARD_LPC11U24_Documents | \BLUEBOARD_LPC11U24_Documents\B | lue B | |
|--------------------------------------|----------------------------|-------------------------------|---------------------------------------|--|--|
| File Edit View Favorites | <u>T</u> ools <u>H</u> elp | | | | |
| 🕞 Back + 🕑 + 🏄 🎝 | 🔎 Search 🌔 | Folders 🔃 🔞 Folder S | ync | | |
| Address 🔁 C:\Work\ngx\Blueb | oard_LPC11U24\BL | .UEBOARD_LPC11U24_Documents\E | LUEBOARD_LPC11U24_Documents(Blue Boar | d_LPC11U24\Source_programs\BLUEBOARD_L | LPC11U24_KEIL_with_GLCD\Blinky\Keiworkspace |
| File and Folder Tasks | * | Lst | Cbj | blinky.uvgui.IVSX12 NGX12 Fie 146 KB | blinky.uvgui NGX12.bak BAK File 146 KB |
| Other Places | * |) binky.uvopt | binky.uvproj | blinky_Blinky.dep | blinky_uvopt.bak |
| 🛅 Blinky 🔒 My Documents | | UVOPT File 61 KB | William Project | DEP File 2 1/8 | BAK File 61 KB |
| 💡 My Computer 🔍 My Network Places | 444 | BAK File | | | |
| V my network mates | * | 30 KB | | | |
| | | | | | |

Fig.15

Step 3: This launches the IDE

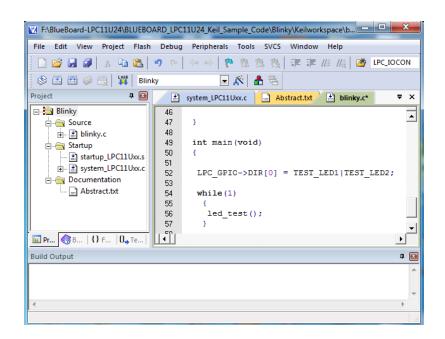


Fig.16



Step 4: Click on Build to build the project as shown in the below image

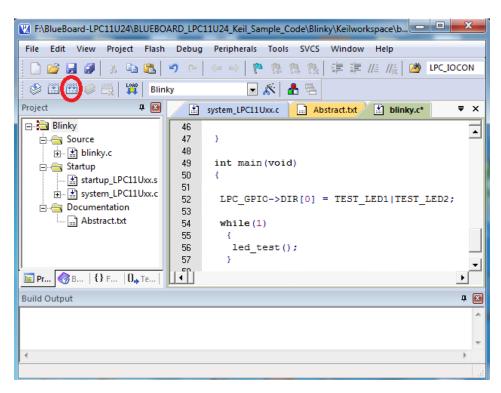


Fig. 17

Step 5: Click on Load to download as shown in the below image

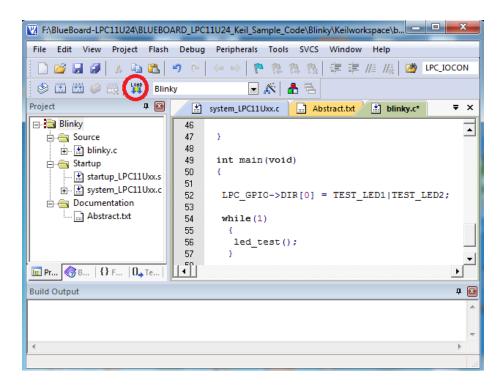


Fig. 18



Step 6: To debug the code click on Debug option then click on Start/Stop Debug session as shown in the below image. Press F5 to free run or press F10 to line by line debug.

| F:\BlueBoard-LPC11U24\BLUEBO | | Peripherals Tools SV | | space\b |
|--------------------------------|------------------|---------------------------|---------------|-------------------|
| | | Start/Stop Debug Session | Ctrl+F5 | E //2 DC_IOCON |
| 🕸 🏥 🕮 🧼 👯 🛙 Blin | k RST | Reset CPU | | |
| Project 🛛 🗜 🔳 | | Run | F5 | 🖹 blinky.c* 🛛 🔻 > |
| 🖃 🔒 Blinky | 0 | Stop | | |
| ia | { ¹ } | Step | F11 | |
| E- Startup | 9. | Step Over | F10 | |
| startup_LPC11Uxx.s | {}- | Step Out | Ctrl+F11 | |
| | -*{} | Run to Cursor Line | Ctrl+F10 | ED1 TEST_LED2; |
| 🖮 🔄 Documentation | \$ | Show Next Statement | | _ |
| Abstraction | | Breakpoints | CtrI+B | |
| | • | Insert/Remove Breakpoint | F9 | |
| | 0 | Enable/Disable Breakpoint | Ctrl+F9 | |
| E Pr | 0 | Disable All Breakpoints | | • |
| Build Output | | Kill All Breakpoints | Ctrl+Shift+F9 | P (|
| | | OS Support | Þ | |
| | | Execution Profiling | + | |
| < | | Memory Map | | Þ |
| Enter or leave a debug session | | Inline Assembly | | |

Fig. 19

3.2 Creating New project

Follow the below steps, for creating new project:

Step 1: Open the keil IDE.

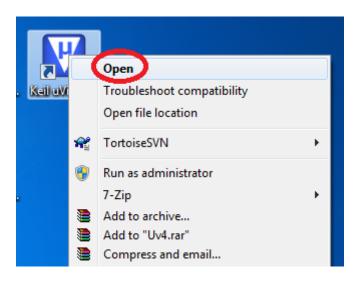


Fig. 20



Step 2: Click on to the Project tab – new uvision project.

| 👿 μVision4 | |
|-------------------|--|
| File Edit View 💽 | oject Flash Debug Peripherals Tools SVCS Window Help |
| i 🗋 😂 🖌 🗿 🗖 | New µVision Project |
| 🖉 🖾 🖉 🥥 | New Multi-Project Workspace |
| Project | Open Project Close Project |
| | Export Manage |
| | Select Device for Target Remove Item |
| S. | Coptions |
| | Clean target |
| ■ Pr 🧑 B {] | |
| | Rebuild all target files |
| Build Output | Batch Build |

Fig. 21

Step 3: Give project name then click Save.

| Create New Project | | - | | | | × |
|--|---|------|------------|------------------|---------|------|
| 🕞 🕞 🗢 📔 🕨 New folder | _ | • | 4 9 | Search New fold | der | ٩ |
| Organize 🔻 New folder | | | | | - | 0 |
| ☆ Favorites | - | Name | | ^ | | Date |
| Nesktop | = | | No | items match your | search. | |
| bownloads | = | | | | | |
| 🔚 Recent Places | | | | | | |
| 🧊 Libraries | | | | | | |
| 🤣 Homegroup | | | | | | |
| 💻 Computer | | | | | | |
| 🚨 OS (C:) | - | • | | | | P. |
| File name: blinky | | | | | | - |
| Save as type: Project Files (*.uvproj) | | | | | | - |
|) Hide Folders | | | | Save | Canc | el |

Fig. 22



Step 4: Select the controller.

| Note: Both LPC11U24 and LPC11U3 | ' are pin and binary co | mpatible; we can select | LPC11U24 device |
|---------------------------------|-------------------------|-------------------------|-----------------|
| for BB-LPC11U37. | | | |

| Select Device for Target 'Target 1' | x |
|--|---|
| CPU Vendor: NXP (founded by Philips) Device: LPC11U24/401 Toolset: ARM | |
| Data base Description: Image: Description in the state of the st | |
| Cancel Help |) |

Fig. 23

Step 5: Go to file – new, & start writing the code.

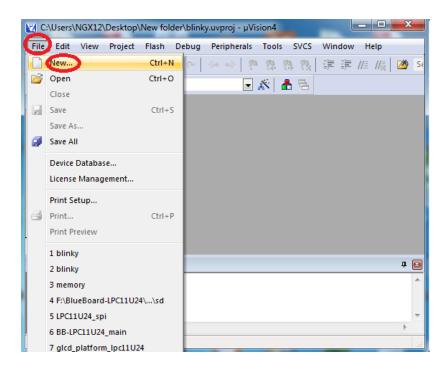


Fig. 24



Step 6: Save the file with some name.

| 🛛 Save As | other state of the | × |
|--|--|----------|
| 🔾 🖯 📕 🕨 New folder | ✓ 4 Search New | folder 🔎 |
| Organize 👻 New folder | | H • 📀 |
| 🔆 Favorites | Name | Dat |
| 🔜 Desktop | blinky.plg | 21- |
| 〕 Downloads | ■ ■ blinky | 21- |
| 📃 Recent Places | blinky_Target 1.dep | 21- |
| 🥽 Libraries | ism startup_LPC11Uxx | 17- |
| 📜 Computer | | |
| 🙇 OS (C)) | |) |
| File names <mark>(blinky.c</mark> Save as type: All Files (*.*) | | • |
| Hide Folders | Save | Cancel |

Fig. 25

Step 7: Add the file to the source group as shown in the below image.

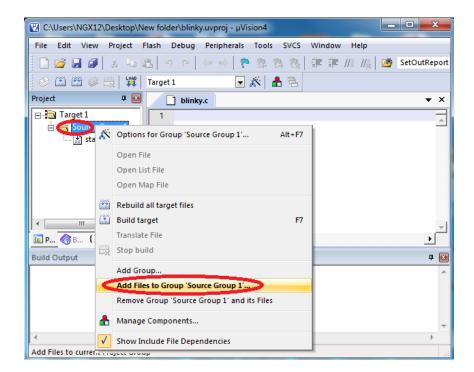




Fig. 26

| Add Files to Group 'Source Group 1' | | x |
|-------------------------------------|------------------|----|
| Look in: 🌗 New folder 💌 | ← 🗈 💣 📰 ◄ | |
| Name | Date modified | Ту |
| blinky | 20-06-2012 11:18 | С |
| | | |
| • |] | P. |
| File name: blinky | Add | |
| Files of type: C Source file (*.c) | ▼ Close | |
| | | |

Step 8: Select the file and click on Add as shown in the below image.

Fig. 27

Step 9: To build, download and debug follow the steps 4, 5 and 6 in section 3.1.

3.3 Creating Bin File

For creating bin file follow the below steps:

Step 1: Open the project & click 'Target Options' and a window will appear.

| 👿 Options | for Targe | et 'Blinky' | | - | - | - | - | | × |
|--|------------|-----------------|------------|---------|---------|-----------|------------------|------------|--------|
| Device | Target 0 | utput Listing | User C/C++ | Asm | Linker | Debug l | Jtilities | | |
| NXP (founded by Philips) LPC11U24/401 Xtal (MHz): Code Generation | | | | | | | | | |
| Operating | system: | None | | • | | | Nodule Optimizat | | |
| System-V | iewer File | (.Sfr): | | | U: | e MicroLl | вГ | Big Endian | |
| SFD\NX | P\LPC11 | Jook/LPC11Uxx. | SFR | | | | | | |
| | Only Memo | • | | | | | iory Areas | | |
| default | off-chip | Start | Size | Startup | default | off-chip | Start | Size | Nolnit |
| | ROM1: | | | 0 | | RAM1: | | | |
| | ROM2: | | | 0 | | RAM2: | | | |
| | ROM3: | | | 0 | | RAM3: | | | |
| | on-chin | | _ | | | on-chip | | | |
| | IROM1: | 0x0 | 0×8000 | 6 | • | IRAM1: | 0x10000000 | 0x2000 | |
| | IROM2: | | | 0 | | IRAM2: | | | |
| | | | | | | | | | |
| | | [| ок | Can | cel | Defa | ults | | Help |



| Fig. | 28 |
|------|----|
| 115. | 40 |

Step 2: Check the device & the start address of IROM1 should be 0x0.

Step 3: Then go to the User tab & insert the below line in the Run #1:

fromelf --bin "./Obj/file.axf" --output "./Obj/file.bin"

| 💟 Options for Target 'Blinky' |
|--|
| Device Target Output Listing User C/C++ Asm Linker Debug Utilities |
| Run User Programs Before Compilation of a C/C++ File |
| Run #1: DOS16 |
| □ Run #2: □ DOS16 |
| ∼ Run User Programs Before Build/Rebuild |
| □ Run #1: □ DOS16 |
| □ Run #2: □ DOS16 |
| |
| Run User Programs After Build/Rebuild |
| Run #1: fromelf -bin ".\Obj\Blinky.axf" -output ".\Obj\Blinky.bin" |
| □ Run #2: □ DOS16 |
| Image: When Complete □ Start Debugging |
| |
| |
| |
| OK Cancel Defaults Help |

Fig. 29

Step 4: Click on the Linker tab & select the 'Use Memory Layout from Target Dialog', then click ok and build the project, finally .bin file will be created.

| Options for Target 'Blinky' | × |
|---|---|
| Device Target Output Listing User | C/C++ Asm Linker Debug Utilities |
| Use Memory Layout from Target Dialog | |
| Make RW Sections Position Indep Make RO Sections Position Indep Don't Search Standard Libraries Report 'might fail' Conditions as Err | endent R/W Base 0x10000000 |
| Scatter File | Edit |
| Misc controls | * |
| | type=microlib -strict -scatter ",\Obj\Blinky.sct" ▲ narysizes -map -xref -callgraph -symbols ▼ |
| | K Cancel Defaults Help |

Fig. 30



Note: For newly built bin file you should update crc. Refer 6.0 Trouble Shooting for more details.

4.0 BLUEBOARD-LPC11U24/37 Programming

4.1 Programming options

BLUEBOARD-LPC11U24/37 can be programmed using the

- On-chip bootloader (USB or UART)
- Debugger (ULINK2)

4.1.1 On-Chip bootloader (USB or UART)

In order to program the board either through USB or UART we need to get the board under programming mode.

Getting the board in programming mode:

Theory: The On-chip bootloader looks for a logic LOW to be present on a pre-defined PIN (ISP pin) during reset. If the ISP pin is held LOW and reset signal is provided to the MCU, the MCU enters into programming mode.

Practical:

On the BLUEBOARD-LPC11U24/37 the RESET and ISP signals are connected to buttons provided on the board. Look for the RESET and ISP marking on the board. Therefore to enter into programming mode:

- A **Press and hold** the ISP button
- \checkmark Press the RESET button and release it
- \checkmark Now release the ISP button
- \checkmark The board is in the programming mode

We know that the on-chip bootloader can be used with USB or UART. Please note that if you have connected a USB cable to the board the USB bootloader is activated else the UART bootloader is activated. Meaning, if you have connected the USB cable as your power source then you cannot use UART bootloader, you need to use an alternate external power source (DC jack) to enable UART bootloader. If a particular MCU supports USB bootloading it is highly recommended to use the same for programming.

Programming through USB is the most convenient way to program the BlueBoard-LPC11U24. The LPC11U24 has an on-chip USB bootloader support which makes programming the board very simple. You don't require any PC application to program using USB bootloader. Once the board enters the programming mode it appears as a drive on your Windows machine and all you need to do is just dragn-drop your binary to this drive.

Note: Not all NXP USB MCUs support USB boot loading. For example the LPC11U14 does not support USB bootloader although it has support for USB on the chip.



4.1.2 Flashing the board using USB

The pre-build binaries can be used to flashing onto to the board for each peripheral by using the USB bootloader as a Mass storage device. Press SW4, then SW5; release SW5, then SW4, the mass storage device will appear on your screen. On the board LED D1 glows. Remove firmware.bin file and then place your bin file and then press reset switch to execute the specific code written on to the flash.

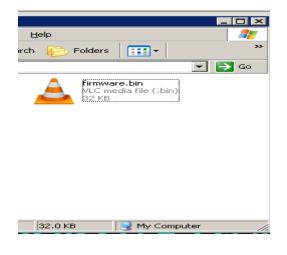


Fig. 31

Flashing the bin file (Drag & drop).

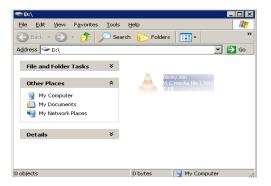


Fig. 32

Note: For newly built bin file you should update crc. Refer 6.0 Trouble Shooting for more details.



4.2 Flashing the Hex file through UART

Step 1: Connect the serial cable to the PC as well as to the board UART0 and open the flash magic tool.

Step 2: Input all the parameters as shown in below Fig.

| 🏶 Flash Magic | - NON PRODUCTION US | E ONLY | | | |
|--|--------------------------------|--------------|--------------|------------------------------|---|
| <u>File ISP Option</u> | s <u>T</u> ools <u>H</u> elp | | | | |
| 🖻 🗟 🔍 🎯 | 🐗 🗸 🎩 🔈 😻 [| 💐 🕜 😂 | | | |
| Step 1 - Communi | cations | Step 2 - Era | se | | |
| Select Device | LPC11U24/401 | | | 00-0x000FFF) | ^ |
| COM Port: | СОМ 1 🛛 🖌 | Erase block | 2 (0x0020 | 00-0x001FFF) 00-0x002FFF) | |
| Baud Rate: | 9600 💌 | Erase block | 4 (0x0040) | 00-0x003FFF) 00-0x004FFF) | |
| Interface: | None (ISP) 🛛 🗸 | | | 00-0x005FFF) | |
| Oscillator (MHz): | 12 | Erase all F | cks used b | | |
| Step 3 - Hex File Hex File: C:\Work\ngx\Blueboard_LPC11U24\BLUEBOARD_LPC11U24_D Browse Modified: Monday, November 28, 2011, 7:38:36 PM more info | | | | | |
| Step 4 - Options | | Ste | p 5 - Start! | | |
| ✓ Verify after programming ✓ Fill unused Flash ✓ Gen block checksums ✓ Execute | | | | | |
| Technical on-line a | articles about 8051 and XA pro | gramming | | | |
| www.esacademy.e | com/fag/docs | | | | • |
| | | | 0 (| | |

Fig. 33

Step 3: Click **Star**t to flash the hex file. Press Reset to run.

NOTE: Make sure that the Board is not powered through USB.



5.0 Schematic & Board Layout

5.1 Schematic

This manual will be periodically updated, but for the latest documentations please check our <u>website</u> for the latest documents. The Board schematic and sample code are available after the product has been registered on our website.

5.2 Board layout

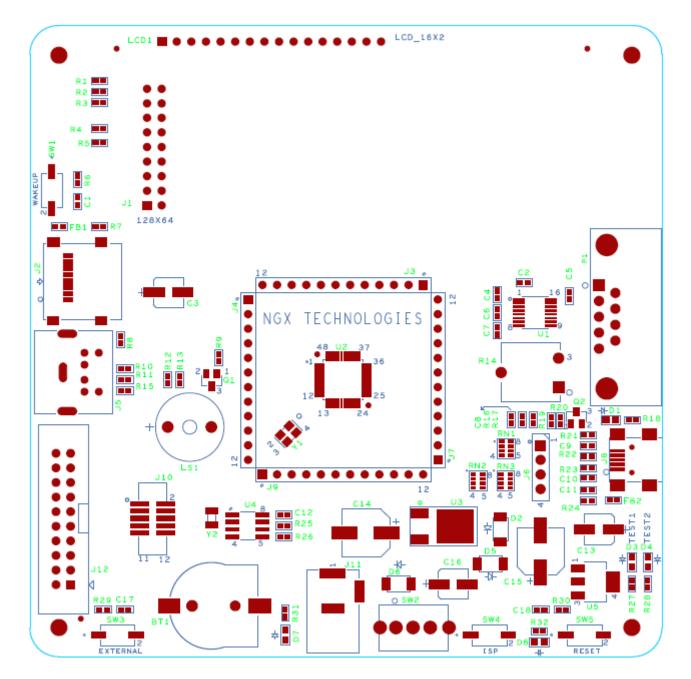


Fig. 34



6.0 Trouble Shooting

For newly created bin file you should update crc.

Unfortunately, the checksum generated is not correct and unless the checksum of the .bin file is modified, the firmware will be rejected by the USB bootloader. Thankfully, this is relatively easy to fix. There is free utility to fix the checksum. A pre-compiled version for Windows (named lpcrc.exe) is also located in the root folder. To fix the checksum, simply go into the command-line and go to the root folder where both the lpcrc.exe tool and your .bin file are located (the file will be named 'firmware.bin' unless you have modified the Makefile), and enter the following command:

Fixing the firmware.bin Checksum

lpcrc firmware.bin

| 🔤 C:\WIND0W5\system32\cmd.exe | | | | |
|--|--|--|--|--|
| Directory of C:\Work\ngx\Blueboard_LPC11U24\BLUEBOARD_LPC11U24_Documents\BLUEBO | | | | |
| ARD_LPC11U24_Documents\Blue Board_LPC11U24\Source_programs\BLUEBOARD_LPC11U24_KE | | | | |
| IL_with_GLCD\Blinky\Keilworkspace\Obj | | | | |
| | | | | |
| 12/09/2011 05:51 PM <dir> .</dir> | | | | |
| 12/09/2011 05:51 PM <dir></dir> | | | | |
| 12/09/2011 05:51 PM 16.112 Blinky.axf | | | | |
| 12/09/2011 05:51 PM 1,300 Blinky.bin | | | | |
| 12/08/2011 12:49 PM 31,238 blinky.crf | | | | |
| 12/08/2011 12:49 PM 366 blinky.d | | | | |
| 12/09/2011 05:51 PM 3,717 Blinky.hex | | | | |
| 12/09/2011 05:51 PM 15,275 Blinky.htm | | | | |
| 12/09/2011 05:51 PM339 Blinky.lnp | | | | |
| 12/08/2011 12:49 PM 33.624 blinky.o | | | | |
| 12/09/2011 05:51 PM 572 Blinky.plg | | | | |
| 10/10/2011 05:08 PM 479 Blinky.sct | | | | |
| 12/09/2011 05:51 PM 1,089 Blinky.tra | | | | |
| 11/09/2011 05:04 PM 19 ExtDll.iex | | | | |
| 03/22/2011 12:49 PM 47,342 lpcrc.exe | | | | |
| 12/08/2011 12:49 PM 4,344 startup_LPC11Uxx.o 12/08/2011 12:49 PM 34.458 system lpc11uxx.crf | | | | |
| 12/08/2011 12:49 PM 34,458 system_lpc11uxx.crf 12/08/2011 12:49 PM 456 system_lpc11uxx.d | | | | |
| 12/08/2011 12:49 PM 37,340 system_lpc11uxx.o | | | | |
| 12/06/2011 12:47 IN 57,540 System_1pt110xx.0 17 File(s) 228.070 bytes | | | | |
| 2 Dir(s) 22.316,539.904 bytes free | | | | |
| 2 DITS/ 22,510,537,701 Bytes 1166 | | | | |
| C:\Work\ngx\Blueboard_LPC11U24\BLUEBOARD_LPC11U24_Documents\BLUEBOARD_LPC11U24_D | | | | |
| ocuments\Blue Board_LPC11U24\Source_programs\BLUEBOARD_LPC11U24_KEIL_with_GLCD\B | | | | |
| linky\Keilworkspace\Obj}lpcrc Blinky.bin | | | | |
| succesfully undated crc to: effff9b3 | | | | |
| | | | | |

Fig. 35



7.0 CHANGE HISTORY

7.1 Change History

| Rev | Changes | Date (dd/mm/yy) | Ву |
|-----|-------------------------------|-----------------|-------------------|
| 1.0 | Initial release of the manual | 18/06/2012 | Veeresh Tumbaragi |

8.0 REFERENCES

In addition to this document, the following references are included on the NGX BLUEBOARD-LPC11U24/37 product and can also be downloaded from <u>www.ngxtechnologies.com</u>:

• NGX BLUEBOARD-LPC11U24/37 schematic for the Development board.

Additional references include:

- Information on development tool being used:
 - Keil uvision 4.23, <u>http://www.keil.com/download/product/</u>
 - Flash magic, http://www.flashmagictool.com/

About this document:

Revision History

Version: V1.0 author: Veeresh Tumbaragi

Company Terms & Conditions

Legal

NGX Technologies Pvt. Ltd. provides the enclosed product(s) under the following conditions:

This evaluation board/kit is intended for use for ENGINEERING DEVELOPMENT, DEMONSTRATION, and EDUCATION OR EVALUATION PURPOSES ONLY and is not considered by NGX Technologies Pvt. Ltd to be a finished end-product fit for general consumer use. Persons handling the product(s) must have electronics training and observe good engineering practice standards. As such, the goods being provided are not intended to be complete in terms of required design-, marketing-, and/or manufacturing-related protective considerations, including product safety and environmental measures typically found in end products that incorporate such semiconductor components or circuit boards. This evaluation board/kit does not fall within the scope of the European Union directives regarding electromagnetic compatibility, restricted substances (RoHS), recycling (WEEE), FCC, CE or UL and therefore may not meet the technical requirements of these directives or other related directives.

The user assumes all responsibility and liability for proper and safe handling of the goods. Further, the user indemnifies NGX Technologies from all claims arising from the handling or use of the goods. Due to the open construction of the product, it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge.



EXCEPT TO THE EXTENT OF THE INDEMNITY SET FORTH ABOVE, NEITHER PARTY SHALL BE LIABLE TO THE OTHER FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.

NGX Technologies currently deals with a variety of customers for products, and therefore our arrangement with the user is not exclusive. NGX Technologies assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein.

Please read the User's Guide and, specifically, the Warnings and Restrictions notice in the User's Guide prior to handling the product. This notice contains important safety information about temperatures and voltages.

No license is granted under any patent right or other intellectual property right of NGX Technologies covering or relating to any machine, process, or combination in which such NGX Technologies products or services might be or are used.

Disclaimers

Information in this document is believed to be reliable and accurate. However, NGX Technologies does not give any representations or warranties, expressed or implied, as to the completeness or accuracy of such information and shall have no liability for the consequences of use of such information.

NGX Technologies reserves the right to make changes to information published in this document, at any time and without notice, including without limitation specifications and product descriptions. This document replaces and supersedes all information supplied prior to the publication hereof.

Trademarks

All referenced trademarks, product names, brands and service names are the property of their respective owners.