STMicroelectronics UM1769 CubeL0 Nucleo Demonstration Firmware User Manual

Contents

life.augmented

STMicroelectronics UM1769 CubeL0 Nucleo Demonstration Firmware

Product Information

- **Product Name:** STM32CubeL0
- **Product Type**: Microcontroller
- **Manufacturer:** STMicroelectronics
- **Release Date:** June 2014
- **Documentation ID:** UM1769
- **Revision:** Rev 1
- **Website**: www.st.com

STM32CubeL0 is a comprehensive package that includes all the necessary embedded software components for developing applications on STM32L0 microcontrollers. It follows the STM32Cube initiative and is highly portable, not only within the STM32L0 series but also to other STM32 series.

Key Features

- Includes all generic embedded software components
- Compatible with STM32CubeMX code generator
- Provides a low-level hardware abstraction layer (HAL)
- Extensive set of examples running on STMicroelectronics boards
- Open-source BSD license for user convenience

Product Usage Instructions

1. Getting started with the demonstration
    - Ensure you have the necessary hardware requirements.
    - Configure the STM32 Nucleo board and assemble the Adafruit shield.
    - Install the demonstration firmware package.
2. Hardware requirements

STM32L053 Nucleo board

Adafruit TFT shield

3. Hardware configuration

   STM32 Nucleo board configuration

   Assembling the Adafruit shield

4. Demonstration firmware package

   Demonstration repository

   Nucleo board BSP (Board Support Package)

   - Joystick configuration
   - LCD configuration
   - MicroSD configuration

5. Programming firmware application

   Follow the instructions provided in section 4.1 of the user manual to program the firmware application.

6. **FAQs:** Refer to section 5 of the user manual for frequently asked questions.
7. **Revision history**: The revision history can be found in section 6 of the user manual.

For more detailed information and instructions, please refer to the complete user manual available at the STMicroelectronics website (**www.st.com**) using the documentation ID UM1769.

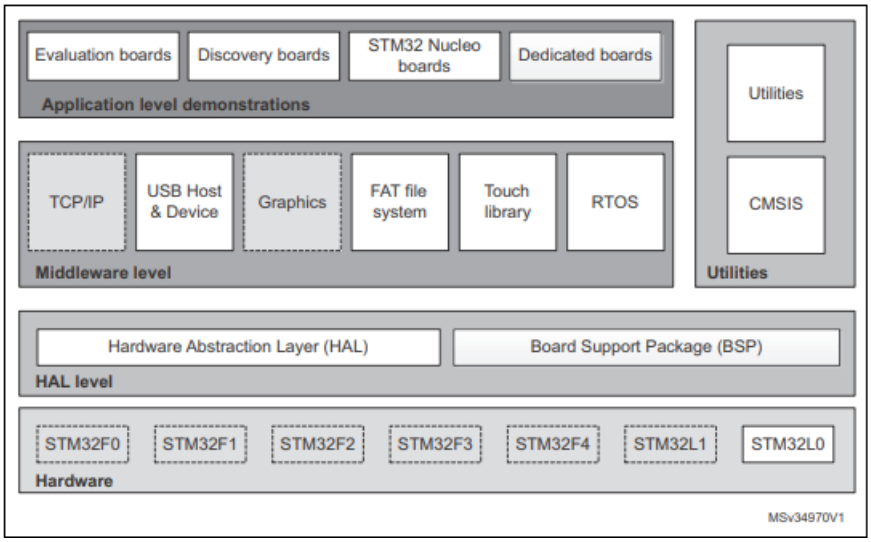
STM32CubeL0 main features

STM32CubeL0 gathers together, in a single package, all the generic embedded software components required to develop an application on STM32L0 microcontrollers. In line with the STM32Cube initiative, this set of components is highly portable, not only within the STM32L0 series but also to other STM32 series.
STM32CubeL0 is fully compatible with STM32CubeMX code generator that allows the user to generate initialization code. The package includes a low level hardware abstraction layer (HAL) that covers the microcontroller hardware, together with an extensive set of examples running on STMicroelectronics boards. The HAL is available in an open-source BSD license for user convenience.
STM32CubeL0 package features a set of middleware components with the corresponding examples. They come with very permissive license terms:

- Full USB Device stack supporting many classes (HID, MSC, CDC, Audio, DFU)
- CMSIS-RTOS implementation with FreeRTOS open source solution
- FAT File system based on open source FatFs solution
- STMTouch touch sensing solution.

A demonstration implementing all these middleware components is also provided in the STM32CubeL0 package.
The block diagram of STM32Cube is shown in Figure 1.

Figure 1. STM32Cube block diagram



Getting started with the demonstration

Hardware requirements

The hardware requirements to start the demonstration application are as follows

- STM32L053 Nucleo board
- Adafruit 1.8″ TFT shield with Joystick and microSD (**reference ID:** 802)
- one ‘USB type A to Mini-B’ cable to power up the STM32 Nucleo board from the USB ST-LINK (USB connector CN1)
- a Standard Capacity SD card (SDSC) with a capacity up to 4GBytes.

STM32L053 Nucleo board

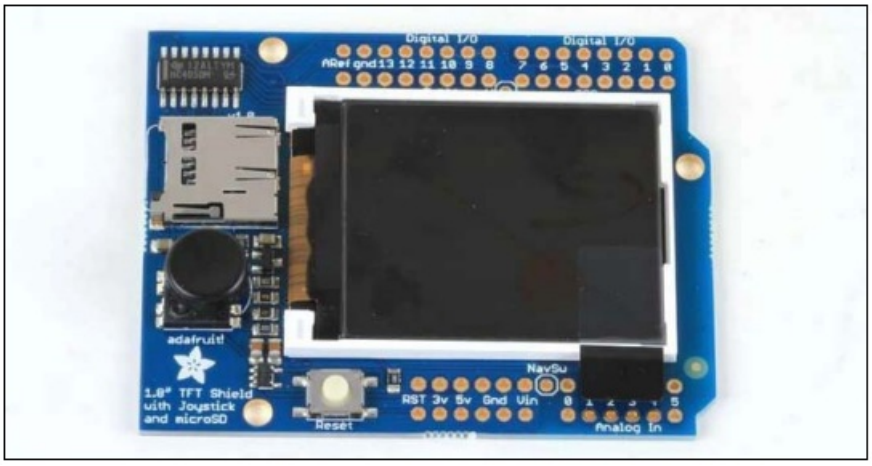
The STM32 Nucleo board is a low-cost and easy-to-use development kit to quickly evaluate and start some development with ARM® 32-bit Cortex®-M microcontrollers of the STM32 series (STM32F103, STM32F030, STM32L152 and STM32L053). Before installing and using the product, please accept the Evaluation Product License Agreement available at **www.st.com/epla**.
For more information on the STM32 Nucleo board visit **www.st.com/stm32nucleo**.

Figure 2. STM32L053 Nucleo board

Adafruit TFT shield
The STM32 Nucleo board supports Arduino connectivity.
This Adafruit 1.8″ TFT shield may be found on the Adafruit website (reference ID 802) with the following features

- one 1.8” TFT display with 128×160 color pixels
- one microSD card interface slot
- one 5-way joystick navigation switch (left, right, up, down, select).

Figure 3. Adafruit 1.8” TFT shield



Note: This shield is just an example of Arduino shield usage; you can get more details on Adafruit website.

Hardware configuration
In addition to gathering the hardware please follow the recommendations below, to start using the Adafruit 1.8” TFT shield with the STM32 Nucleo board.

STM32 Nucleo board configuration
Check jumpers’ positions on the STM32 Nucleo board as follows:

- JP1 OFF

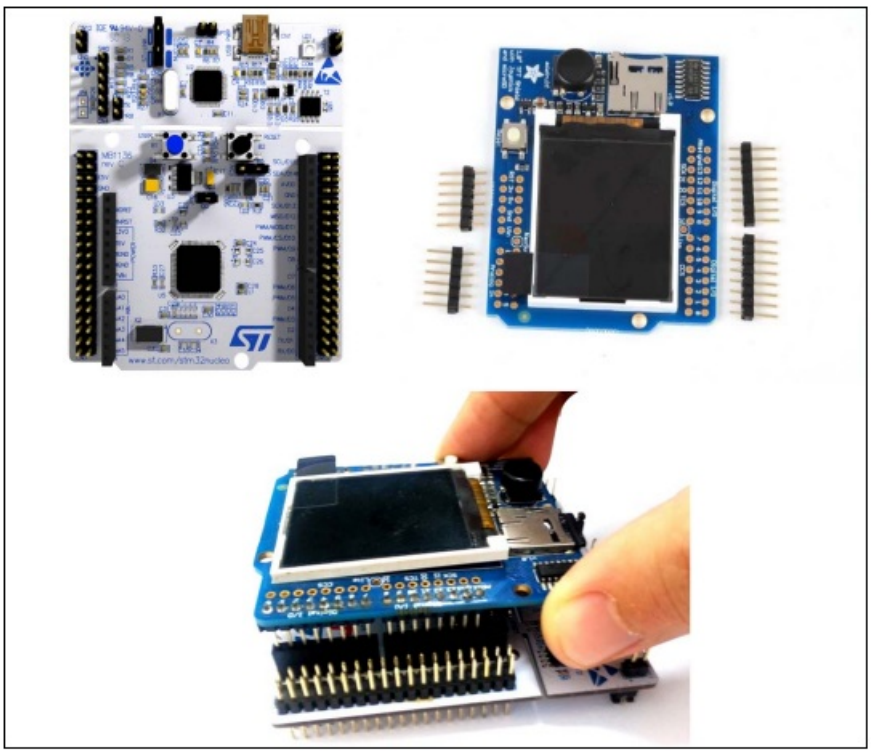
- JP5 (PWR) on U5V side ON
- JP6 (IDD) ON.

Assembling the Adafruit shield
The Adafruit TFT shield comes with all surface mount parts pre-soldered. User can install the headers following the next steps:

- **Cut the breakaway header strip into sections to fit the holes on the edge of the shield:** two sections of 6 pins and two other sections of 8 pins are needed.
- To align the header strips for soldering, insert them (long pins down) into the headers of the STM32 Nucleo board using the connectors CN5, CN6, CN8 and CN9.
- Place the shield over the header strips so that the short pins stick up through the holes.
- Solder on each pin of the header onto the shield PCB to ensure good electrical contact.

The sequence is shown in Figure 4.

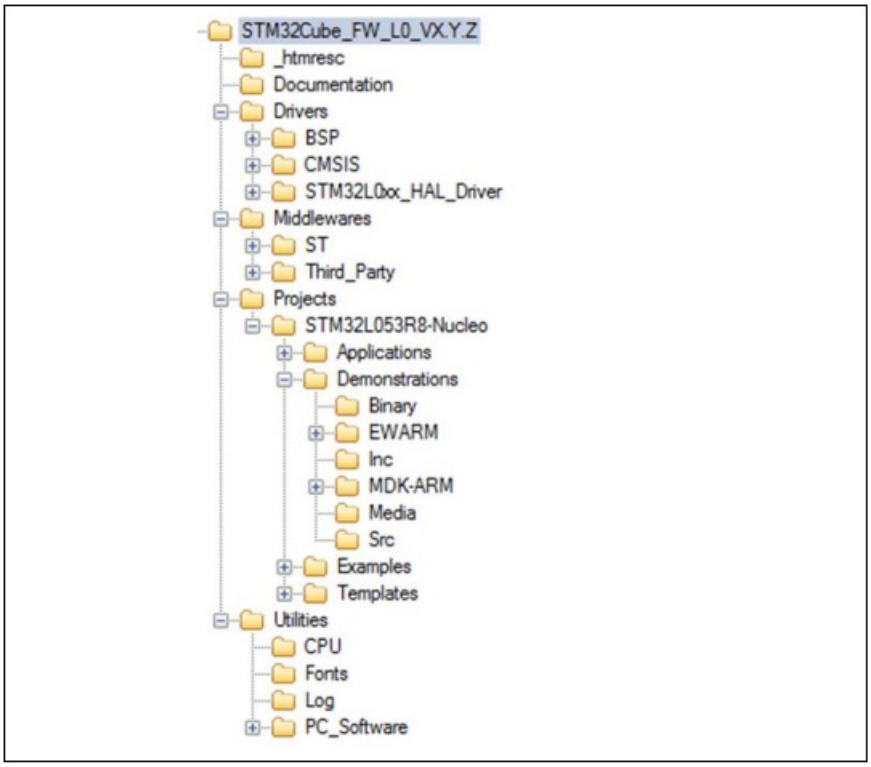
Figure 4. Assembling the Adafruit 1.8" TFT shield



Demonstration firmware package

Demonstration repository
The Nucleo demonstration is provided within the STM32CubeL0 firmware package as shown in Figure 5.

Figure 5. Folder structure



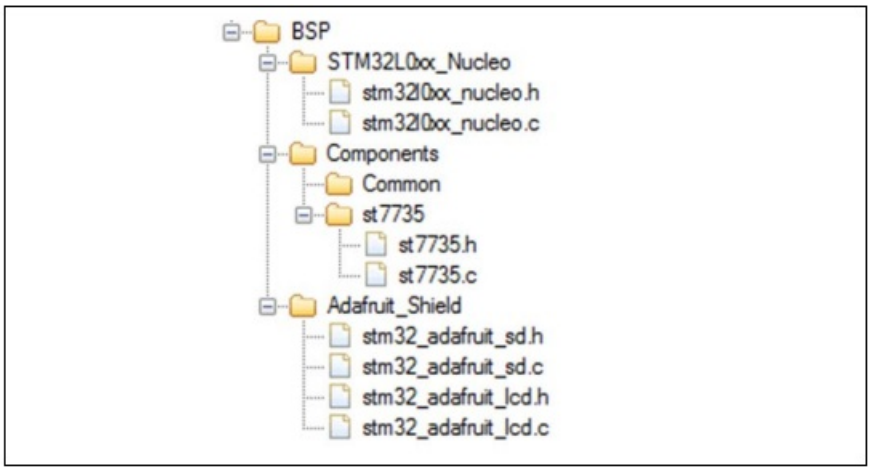
The demonstration sources are located in the projects folder of the STM32Cube package for the STM32L053R8 Nucleo board. The sources are divided into five groups described as follows:

1. **Binary:** demonstration binary file in Hex format
2. **Media**: contains the media files (*.bmp) required to run the demonstration
3. **Inc**: contains the demonstration header files
4. **Src:** contains the demonstration source files
5. **Project settings**: a folder per toolchain containing the project settings and the linker files.

Nucleo board BSP

For each board, a set of Button, LED and Joystick drivers is available within the stm32l0xx_nucleo.c/.h files, implementing the board capabilities and the bus link mechanism.

Figure 6. Nucleo BSP architecture



Joystick

The 5-way joystick on the shield is based on a resistor trick to permit all the switches to share one analog pin.

Each movement of the joystick control connects a different resistor and results in a different voltage reading.
The ADC peripheral is configured within the stm32l0xx_nucleo.c/.h driver in order to get analog voltage values through the analog I/O pin 3.
The BSP_JOY_GetState() function reads the analog pin and compares the result with 5 different ranges to determine which (if any) direction the stick has been moved (left, right, up, down, select).

LCD

The LCD available on the Adafruit 1.8″ TFT shield uses 4-wire SPI to communicate with the STM32L0 chip (Digital I/O pins 13, 11, 10 and 8) and has its own pixel-addressable frame buffer to display text, shapes, lines, pixels, etc.
The SPI peripheral is configured within the stm32l0xx_nucleo.c/.h driver which contains also the SPI bus link mechanism and IO operations.
The LCD is controlled by a dedicated BSP LCD driver stm32_adafruit_lcd.c/.h which uses the st7735 component that exports in a generic way the LCD IO operations needed for its process.

MicroSD

The microSD slot available on the Adafruit 1.8″ TFT shield uses 4-wire SPI to communicate with the STM32L0 chip (Digital I/O pins 13, 12, 11 and 4).
The SPI peripheral is configured within the stm32l0xx_nucleo.c/.h driver which contains also the SPI bus link mechanism and IO operations.
The microSD is controlled by a dedicated BSP SD driver stm32_adafruit_sd.c/.h which exports in a generic way the SD IO operations needed for its process.

Demo functional description

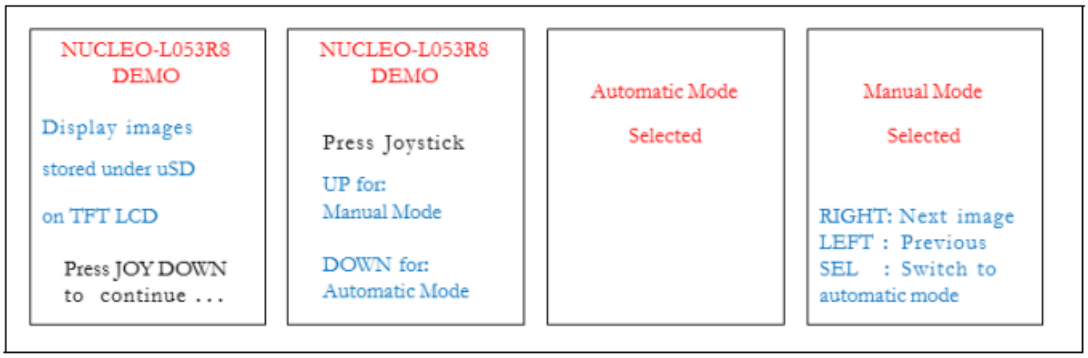
In this demonstration application, we will show how to use the STM32CubeL0 firmware package with the NUCLEO-L053R8 board and the Adafruit 1.8” TFT shield to display a 128×160 pixel full color bitmap from a microSD card using the FatFs file system.
To start with this demonstration application user has to copy the provided 128×160 pixel bitmap pictures available within the FW package under “\Media” folder to the root directory of a FAT formatted microSD card and insert the microSD card into the Adafruit shield microSD holder.
Note that the microSD card can have a storage capacity up to 4GBytes (SDSC) and that the bitmap images must have the properties detailed in Table 1.

Table 1. Bitmap image properties

	Dimensions	128 x 160
	Width	128 pixels
	Height	160 pixels
	Bit depth	16
	Item type	BMP file
	Name	Must not exceed 11 characters (including .bmp extension).

Once started, the application checks the availability of Adafruit 1.8″ TFT shield on top of STM32 Nucleo board. This is done by reading the state of IO PB.00 pin (mapped to Joystick available on the shield). If the state of PB.00 is high then the shield is available.
If the Adafruit 1.8″ TFT shield is not available, the LED2 is toggling with a frequency equal to ~1Hz. A second press on the User button lets LED2 toggling with a second frequency equal to ~5Hz. The third press, changes LED2 toggling frequency to ~10Hz. The described process is done in an infinite loop.
If the Adafruit 1.8″ TFT shield is available, LED2 is turned ON, because it’s sharing the same pin with the SPI CLK signal used to communicate with the LCD and microSD available on the shield.
A menu is displayed on Adafruit 1.8″ TFT describing the demonstration application, as shown in Figure 7.

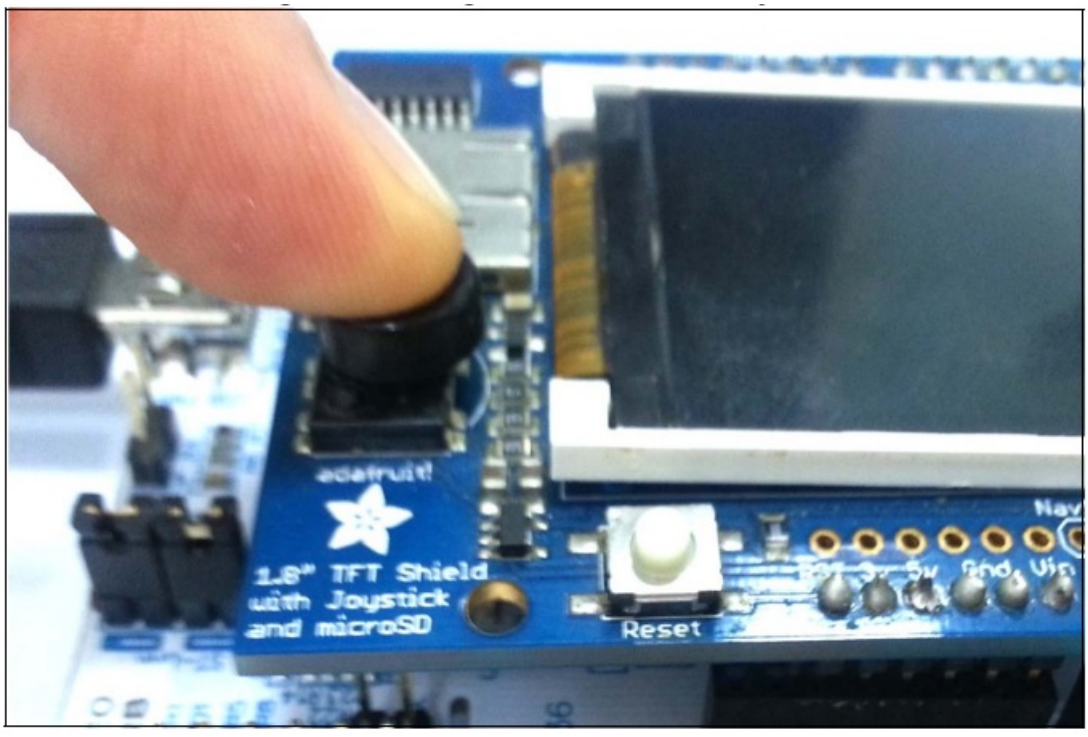
Figure 7. Demonstration application menu



User has to follow the instructions below

- Press the Joystick DOWN to continue menu display (see Figure 8)
- Choose one of the available display modes (manual and automatic) using the Joystick button
  - **Automatic mode**: by pressing Joystick DOWN
    The bitmap images available on the microSD card are displayed sequentially in a forever loop.
  - **Manual mode**: by pressing Joystick UP
    The bitmap images available on the microSD card are displayed by pressing Joystick RIGHT to display next images, or Joystick LEFT to display previous one. Pressing long (~1s) the Joystick SEL, switches the display mode from manual to automatic.

Figure 8. Reading the Adafruit shield Joystick



It is worth noting that the application manages some errors (refer to Figure 9) that can occur during the access to microSD card to load the bitmap images

- If the microSD card is not FAT formatted, a message will be displayed on TFT. In this case, format the microSD card and put into its root directory the bmp files available within the FW package under \Media folder.
- If the content of the microSD card is other than a bitmap file, a message will be displayed on TFT mentioning

that it is not supported. User has to ensure that the files available under the microSD card root directory are respecting the above described bitmap properties.

Figure 9. Demonstration application error messages

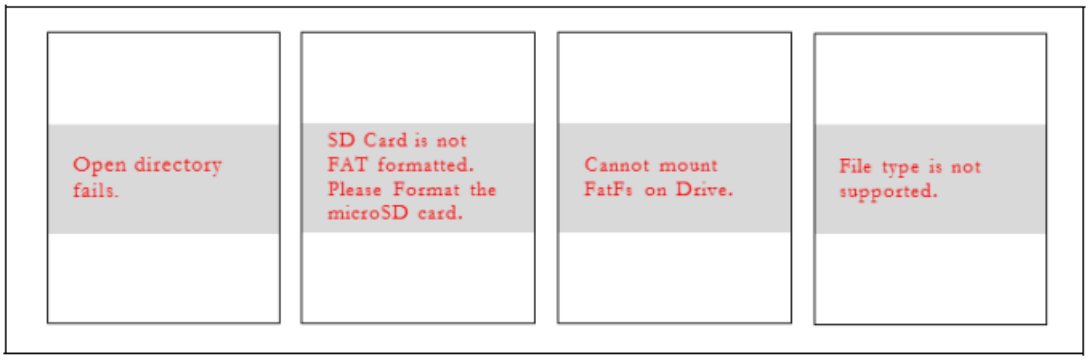
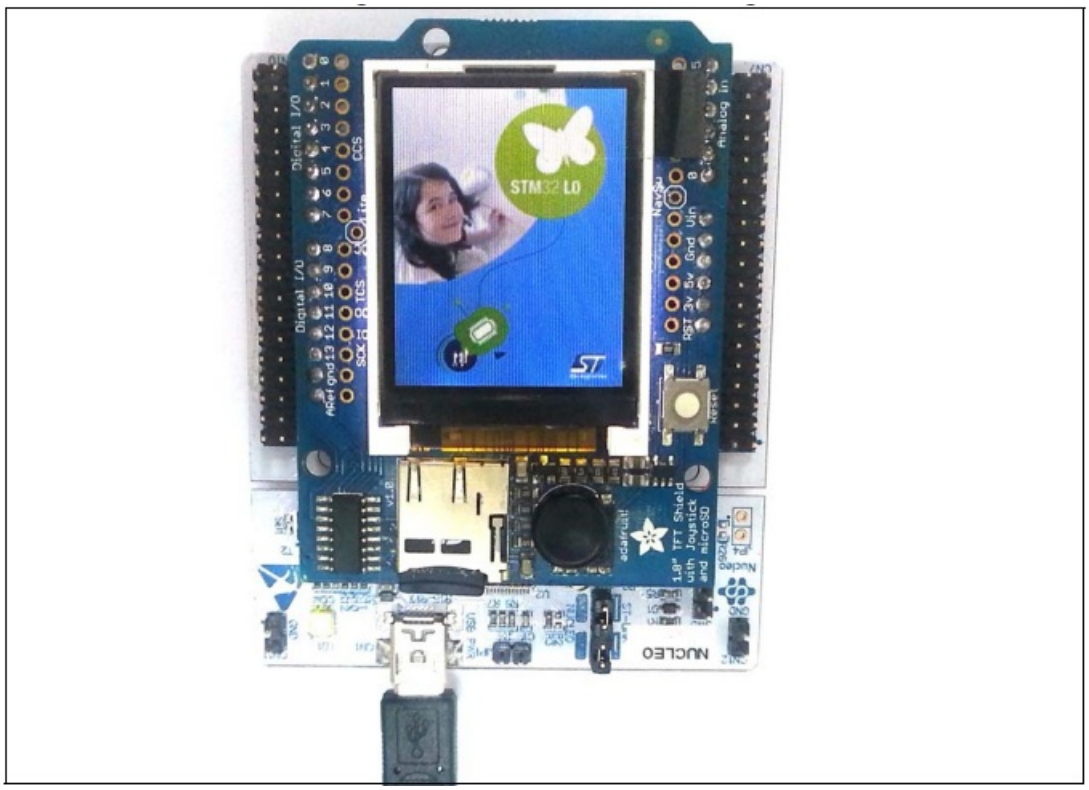


Figure 10. Demonstration running



Programming firmware application
To program the STM32 Nucleo board with the demonstration application please proceed as follows

1. Install the preferred Integrated Development Environment (IDE)
2. Install the ST-LINK/V2.1 driver available on ST website. There are two ways of programming the STM32 Nucleo board

Method1
Upload the STM32CubeL0_Demo_Nucleo.hex from the firmware package available under Projects\STM32L053R8-Nucleo\Demonstrations\Binary using your preferred in-system programming tool.

Method2
Choose one of two supported tool chains (IAR/ Keil) and follow the steps below

- **Open the application folder:** Projects\STM32L053R8-Nucleo\Demonstrations\
- Chose the desired IDE project (EWARM for IAR, MDK-ARM for Keil)
- Double click on the project file (for example Project.eww for EWARM)
- **Rebuild all files**: Go to Project and select Rebuild all
- **Load the project image:** Go to Project and select Debug
- **Run the program:** Go to Debug and select Go.

The demonstration software as well as other software examples that allow you to discover the STM32 microcontroller features are available on ST website at
www.st.com/stm32nucleo.

FAQs

How can I use this application to display my own images?
Use any image editing tool and crop your image to no larger than 160 pixels high and 128 pixels wide. Save it as a 16-bit color BMP format file.

Can I display more bitmap files?
Yes. You can display more pictures, by copying them under the microSD root directory. Just modify the define value of MAX_BMP_FILES constant to the desired number of files. In this case, you must fine tune the _FS_LOCK value, defining the number of files that can be opened simultaneously, under “ffconf.h” the FatFs configuration file.

What about putting the bitmap files not under the root directory of the microSD?
Once put under another folder different from the root directory, the bitmap files cannot the accessed by the demonstration application. The “File type not supported” error message is displayed on the LCD.
To make it work you have to add the new directory path within f_open() and f_opendir() FatFs APIs calls’ under the fatfs_storage.c file.

Revision history

Table 2. Document revision history

Date	Revision	Changes
10-Jun-2014	1	Initial release.

PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

ST PRODUCTS ARE NOT DESIGNED OR AUTHORIZED FOR USE IN

- (A) SAFETY CRITICAL APPLICATIONS SUCH AS LIFE SUPPORTING, ACTIVE IMPLANTED DEVICES OR SYSTEMS WITH PRODUCT FUNCTIONAL SAFETY REQUIREMENTS;
- (B) AERONAUTIC APPLICATIONS;
- (C) AUTOMOTIVE APPLICATIONS OR ENVIRONMENTS, AND/OR
- (D) AEROSPACE APPLICATIONS OR ENVIRONMENTS. WHERE ST PRODUCTS ARE NOT DESIGNED FOR SUCH USE, THE PURCHASER SHALL USE PRODUCTS AT PURCHASER'S SOLE RISK, EVEN IF ST HAS BEEN INFORMED IN WRITING OF SUCH USAGE, UNLESS A PRODUCT IS EXPRESSLY DESIGNATED BY ST AS BEING INTENDED FOR "AUTOMOTIVE, AUTOMOTIVE SAFETY OR MEDICAL" INDUSTRY DOMAINS ACCORDING TO ST PRODUCT DESIGN SPECIFICATIONS. PRODUCTS FORMALLY ESCC, QML OR JAN QUALIFIED ARE DEEMED SUITABLE FOR USE IN AEROSPACE BY THE CORRESPONDING GOVERNMENTAL AGENCY.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.
ST and the ST logo are trademarks or registered trademarks of ST in various countries.
Information in this document supersedes and replaces all information previously supplied.
The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

STMicroelectronics group of companies

- Australia
- Belgium
- Brazil
- Canada
- China
- Czech Republic
- Finland
- France
- Germany
- Hong Kong
- India
- Israel
- Italy
- Japan
- Malaysia
- Malta
- Morocco
- Philippines
- Singapore

- Spain
- Sweden
- Switzerland
- United Kingdom
- United States of America

  www.st.com

Documents / Resources

	STMicroelectronics UM1769 CubeL0 Nucleo Demonstration Firmware [pdf] User Manual UM1769 CubeL0 Nucleo Demonstration Firmware, UM1769, CubeL0 Nucleo Demonstration Firmware, Demonstration Firmware, Firmware

References

- STMicroelectronics: Our technology starts with you