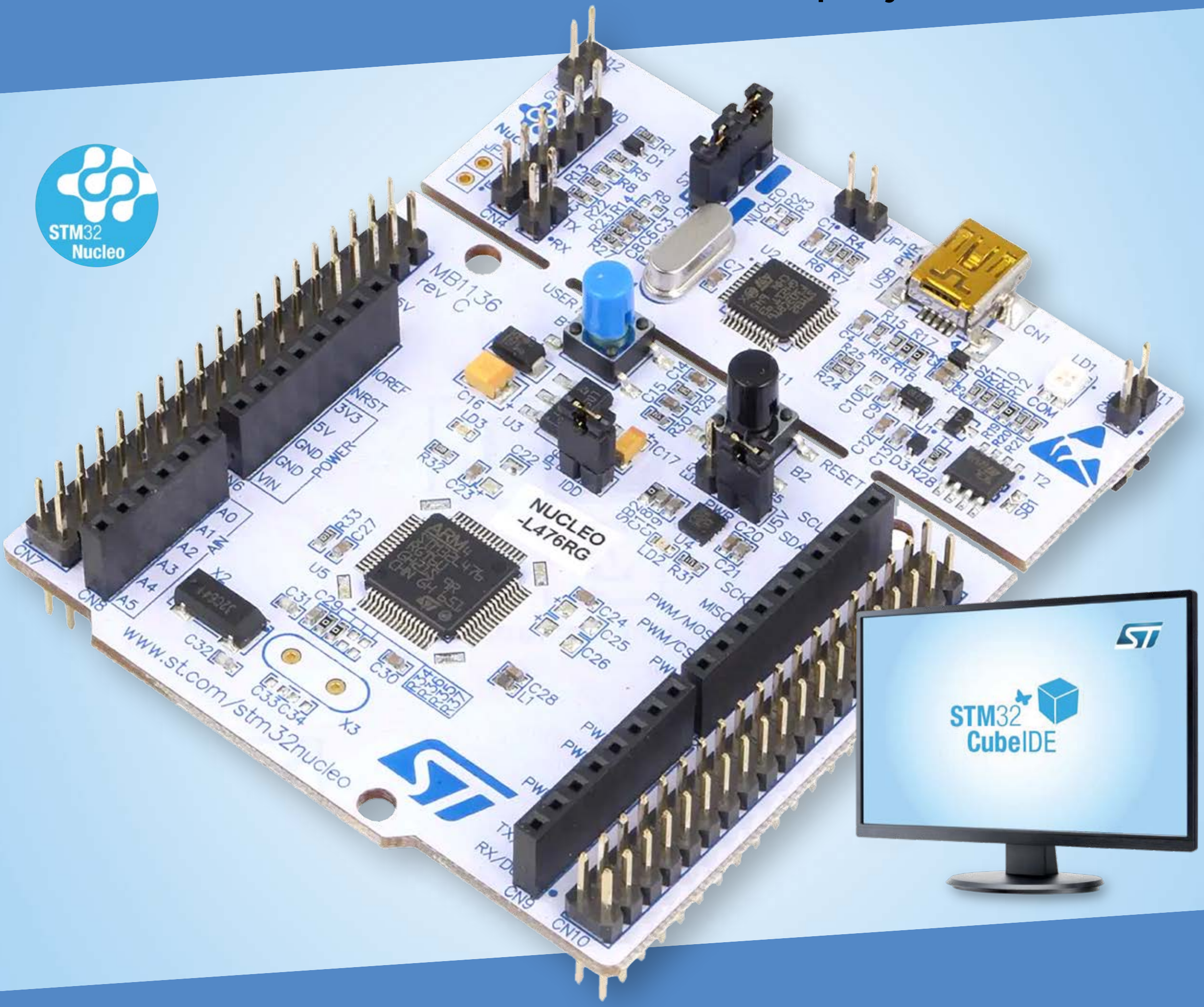
elektor books

Nucleo Boards Programming with the STM32CubeIDE

Hands-on in more than 50 projects



Dogan Ibrahim

elektor
design > share > sell

Nucleo Boards Programming with the STM32CubeIDE

Hands-on in more than 50 projects

Dogan Ibrahim

elektor
design > share > sell

This is an Elektor Publication. Elektor is the media brand of
Elektor International Media B.V.
PO Box 11, NL-6114-ZG Susteren, The Netherlands
Phone: +31 46 4389444

All rights reserved. No part of this book may be reproduced in any material form, including photocopying, or storing in any medium by electronic means and whether or not transiently or incidentally to some other use of this publication, without the written permission of the copyright holder except in accordance with the provisions of the Copyright Designs and Patents Act 1988 or under the terms of a licence issued by the Copyright Licencing Agency Ltd., 90 Tottenham Court Road, London, England W1P 9HE. Applications for the copyright holder's permission to reproduce any part of the publication should be addressed to the publishers.

Acknowledgements

The following figures and pictures in this book, numbered: 1.1, 1.5, 1.6, 1.7, 1.9, 2.2, 2.3, 2.9, 2.10 through 2.20, 2.22 through 2.32, 4.1, 4.2, 6.1, 14.1, 15.1 through 15.10, 16.2, and 16.7 are taken from these STMicroelectronics sources:

- UM1724 User Manual, STM32 Nucleo-64 Boards (DocID025833 Rev 14);
- RM0351 Reference Manual STM32L4x5 and STM32L4x6 advanced ARM®-based 32-bit MCUs (DocID024597 Rev 5).

The pictures of Nucleo Expansion Boards in Chapter 16 of the book are taken from the following STMicroelectronics Internet source:
http://www.st.com/en/evaluation-tools/stm32-nucleo-expansion-boards.html?querycriteria=productId=SC1971.
All the above figures/pictures are used with the written permission of:
© STMicroelectronics. Used with permission.

The author would like to thank to Michael Markowitz of the STMicroelectronics for giving permission to reproduce the above pictures/figures, as well as for providing sample Nucleo expansion boards for use in the projects contained in the book. The author is also grateful to Elektor International Media for editing and publishing the book.

The author and publisher have used their best efforts in ensuring the correctness of the information contained in this book. They do not assume, and hereby disclaim, any liability to any party for any loss or damage caused by errors or omissions in this book, whether such errors or omissions result from negligence, accident or any other caus

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

ISBN 978-3-89576-416-5 Print
ISBN 978-3-89576-416-6 eBook
ISBN 978-3-89576-416-7 ePub

© Copyright 2020: Elektor International Media B.V.
Prepress Production: D-Vision, Julian van den Berg

Elektor is part of EIM, the world's leading source of essential technical information and electronics products for pro engineers, electronics designers, and the companies seeking to engage them. Each day, our international team develops and delivers high-quality content - via a variety of media channels (including magazines, video, digital media, and social media) in several languages - relating to electronics design and DIY electronics. **www.elektormagazine.com**

PREFACE

Arm (previously written as ARM; ARM) is a family of processors based on the RISC architecture that requires fewer transistors than conventional processors. As a result, Arm processors cost less, have reduced power consumptions, and are therefore desirable for use in portable battery-operated devices. Currently, a very large percentage of smartphones, laptops, tablet computers, and portable games devices are all designed using the Arm architecture. Most Arm processors have 32-bit instruction sets with floating point support. It is estimated that over 100 billion Arm processors were produced in the year 2017.

Arm-Cortex is a family of 32-bit RISC processors consisting of Cortex-M0/M0+/M1/M3 and Cortex-M4. Lately, new models such as Cortex-M323 and Cortex-M33 are available. Although 8-bit microcontrollers have been very popular in the past, the devices in the Cortex-M series have become widespread replacements.

The STM32 Nucleo family of processors is manufactured by STMicroelectronics. These are low-cost Arm microcontroller development boards. This book is about developing projects using the Nucleo-L476RG development board with the STM32CubeIDE software. In the early chapters of the book the architecture of the Nucleo family is briefly described.

All the projects in the book have been designed using the STM32CubeIDE software development tool which can be downloaded free of charge from the STMicroelectronics website. The book covers many projects using most features of the Nucleo-L476RG development boards where the full software listing for STM32CubeIDE are given for every project with the description of each software. The projects range from simple flashing LEDs to more complex projects using modules and devices such as GPIO, ADC, DAC, I^{2}C, SPI, LCD, power management, analogue inputs, and others. In addition, several projects are given using the compatible Nucleo Expansion Boards. These Expansion Boards plug on top of the Nucleo development boards and provide features such as industrial input/output, DC motor drive, stepper motor drive, environmental MEMS sensors, accelerometer, gyroscope, Wi-Fi, and many more.

All the projects in the book have been fully tested and are working. The following sub-headings are given for each project where possible:

- Project Title
- Description
- Aim
- Block Diagram
- Circuit Diagram
- Program Listing (based on STM32CubeIDE)

Complete program listings for each project can be obtained from the book support and resources page on the Elektor website, www.elektor.com/xxxxxxx.

I hope you find the projects interesting and the book becomes a useful source of reference for your future STM32 Nucleo projects based on the STM32CubeIDE.

Prof Dr Dogan Ibrahim

London, 2020

CHAPTER 1 • STM32 Nucleo Development Boards

1.1 Overview

This Chapter is about the STM32 Nucleo development boards, and brief specifications of their various guises are described here. In addition, the popular Nucleo-L476RG board used in the projects in this book is described in greater detail.

1.2 STM32 Nucleo development boards

STMicroelectronics is a company with a reputation of supplying quality microcontroller development boards and assorted hardware and software development tools. The Nucleo family of low-cost development boards comprises small but powerful boards based on the 32-bit Arm (also: 'ARM') Cortex architecture. These development boards are targeted for a large audience, including students, professional engineers, and hobbyists at all levels. The boards are compatible with the popular Arduino, mbed, ST-LINK, and ST Morpho, making them accessible to users with different backgrounds.

There are over 30 different boards in the Nucleo family, aimed to satisfy the needs of almost all types of user. The Nucleo boards come in three different sizes: small (**Nucleo-32**), short (**Nucleo-64**), and long (**Nucleo-144**) where the numbers refer to the pin counts of the MCUs. These three groups are further divided into three sub-groups: ultra-low power (**green**), mainstream (**blue**), and high performance (**magenta**).

The ultra-low power boards are based on the STM32 L family and these boards are targeted for low-power applications, such as watches, smart meters etc. Examples of the ultra-low power boards are: Nucleo-L011K4, Nucleo-L031K6 and Nucleo-L432KC. There are three sub-categories in the STM32 L family:

- L0, Arm Cortex-M0+
- L1, Arm Cortex-M3
- L4, Arm Cortex-M4

About half of the STM32 Nucleo boards are in the mainstream category. Examples of the mainstream boards are: Nucleo-F303K8, Nucleo-F042K6, Nucleo-F303RE etc. There are three sub-categories in the mainstream category:

- F0, Arm Cortex-M0+
- F1, Arm Cortex-M3
- F3, Arm Cortex-M4

The high-performance boards have large memories and faster MCUs. Examples of high-performance boards are: Nucleo-F410RB, Nucleo-F401RE, Nucleo-F722ZE, etc. There are three subcategories in the high-performance category:

- F2, Arm Cortex-M3
- F4, Arm Cortex-M4
- F7, Arm Cortex-M7

The Nucleo-32 boards are small (50 mm × 19 mm) and Arduino Nano compatible. The Nucleo-64 and Nucleo-144 are Arduino Uno compatible and they also have ST Morpho extension connectors which carry the MCU pins. There are large number of Arduino Nano/Uno compatible shields available in the market and these shields can easily be used with the

Nucleo boards, making it easy to quickly develop projects based on them.

Depending upon the model, the Nucleo boards have flash program memory sizes from 16 KB up to 2 MB, and RAM memories from 4 KB to 320 KB. The clock frequency varies from 32 MHZ to 216 MHz.

Figure 1.1 shows a comparison of the Nucleo boards.

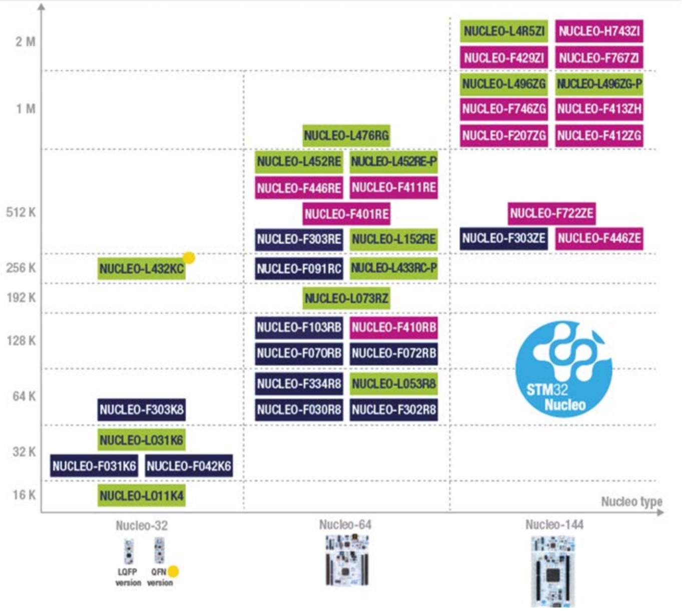


Figure 1.1: Nucleo boards. (©STMicroelecronics. Used with permission)

1.2.1 STM32 processor family numbering

Table 1.1 shows the STM32 processor family codes and their basic specifications.

Code	Processor	Max. speed (MHz)	Max. flash (KB)	Description
F0	Cortex-M0	48	256	Widely used
F1	Cortex-M3	72	1024	Widely used
F2	Cortex-M3	120	1024	High performance
F3	Cortex-M4	72	512	Widely used
F4	Cortex-M4	180	2048	High performance
F7	Cortex-M7	216	2048	High performance
H7	Cortex-M7	400	2048	High performance
L0	Cortex-M0	32	192	Low power
L1	Cortex-M3	32	512	Low power
L4	Cortex-M4	80	1024	Low power
L4+	Cortex-M4	120	2048	Low power

Table 1.1: STM2 family processor numbering.

The STM32 processors are numbered as follows:

STM32*xxyyzpqr*

Where

xx is the processor code as shown in Table 1;
yy is a number;
z is the MCU pin number count as shown in Table 1.2.

Code (z)	Pin count
A	169
B	208
C	48
F	20
G	28
H	40
I	176
J	72
K	32
M	81
N	216
Q	132
R	64
T	36
U	63
V	100
Z	144

Table 1.2: MCU pin counts.

p is the size of the flash memory as shown in Table 1.3;

Code (p)	Flash memory (KB)
4	16
6	32
8	64
B	128
Z	192
C	256
D	384
E	512
F	768

G	1024
H	1536
I	2048

Table 1.3: Flash memory size.

q is the processor's case type as shown in Table 1.4;

Code (q)	Casing
T	LQFP
H	UFBGA
Y	WLCSP

Table 1.4: Processor casing.

r is the operating temperature range as shown in Table 1.5.

Code (r)		Temperature range
6		−40 to 85 ºC
7		−40 to 105 ºC

Table 1.5: Temperature range.

An example of a processor type number is given below.

STM32L031K6T6: **L0** family, number **31**, **32** pin device, **32 KB** flash memory, **LQFP** type packaging, **−40 to 85** ºC operating temperature range.

1.2.2 Nucleo-32 development boards

Figure 1.2 shows an example Nucleo-32 board, the Nucleo-L031K6. This is an ultra-low power low-cost board incorporating the STM32L031K6T6 microcontroller. The board is Arduino Nano compatible allowing many Arduino shields to be used with the board. This board has the following features:

- 32 MHz Cortex M0+ microcontroller in 32-pin package
- 32 KB flash memory
- 8 KB RAM
- 1 KB EEPROM
- Real-time clock
- Serial interfaces (USART, SPI, and I^2C)
- Three LEDs (USB communication, power, user)
- Pushbutton Reset
- Flexible power supply options: ST-LINK USB VBUS or external sources
- Arduino Nano compatible expansion connector
- ST-LINK/V2-1 debugger/programmer with mass storage, virtual COM port, and debug port

- Support for Integrated development Environment software (IAR, Keil, Arm Mbed, GCC-based IDEs).

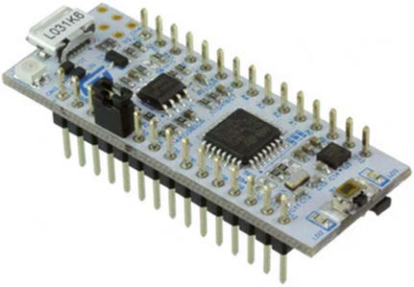


Figure 1.2: Nucleo-32 development board (Nucleo-L031K6).

1.2.3 Nucleo-64 development boards

Figure 1.3 shows an example Nucleo-64 board, the Nucleo-F091RC. This is a mainstream board incorporating a 64-pin MCU. The board is Arduino Uno compatible and as such, a large array of Arduino shields can be used with the board. This board has the following features:

- One user LED
- One user pushbutton switch
- 32.768 kHz crystal oscillator
- ST morpho connector
- Arduino Uno expansion socket
- Flexible power-supply options: ST-LINK USB V_{BUS} or external sources
- ST-LINK/V2-1 debugger/programmer with mass storage, virtual COM port, and debug port
- Comprehensive free software libraries
- Support of a wide choice of Integrated Development Environments (IAR, Keil, Arm Mbed, GCC-based IDEs)

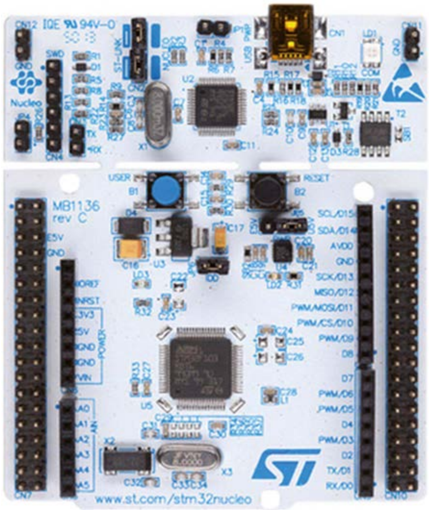


Figure 1.3: Nucleo-64 development board (Nucleo-F091RC).

1.2.4 Nucleo-144 development boards

Figure 1.4 shows an example Nucleo-144 board, the Nucleo-F722ZE. This is a high-performance board incorporating a 144-pin MCU. The board is Arduino Uno compatible. The features of this board are:

- Ethernet compliant with RJ45 connector
- ST morpho connector
- ST-LINK/V2-1 debugger/programmer with mass storage, virtual COM port, and debug port
- ST Zio connector
- 3 user LEDS
- 2 pushbutton switches
- 32.768 kHz crystal oscillator
- Flexible power-supply options: ST-LINK USB V_{BUS} or external sources
- Comprehensive free software libraries
- Support of a wide choice of Integrated Development Environments (IAR, Keil, Arm Mbed, GCC-based IDEs)

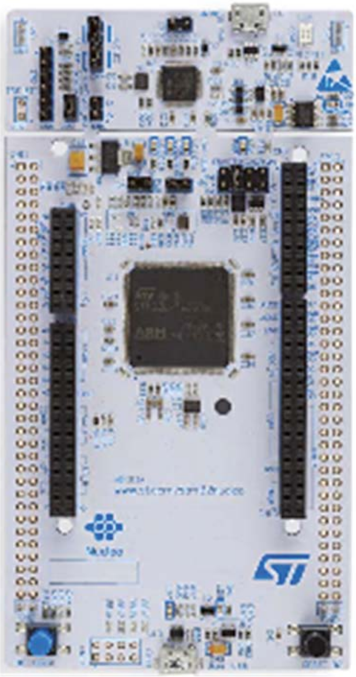


Figure 1.4: Nucleo-144 development board (Nucleo-F7222ZE).

1.3 The Nucleo-L476RG development board

The Nucleo-L476RG board is one of the most popular Nucleo development boards. This is a 64-pin ultra-low power board (see Figure 1.1). This board is used as the example development board in all the projects in this book. Therefore, full details of this board are given in this section so that the readers are familiar with this board. Further details about the Nucleo-64 boards can be obtained from the *STMicroelectronics User Manual UM1724, STM32 Nucleo-64 boards, 2017*.

1.3.1 Two-part board

The Nucleo-L476RG board measures 70 mm × 82.5 mm. As shown in Figure 1.5, the board consists of two parts: the smaller ST-LINK part with the mini USB port, and the MCU part. The ST-LINK part of the PCB can be cut if desired to reduce the overall board size. If this is done the MCU part can only be powered by $V_{IN,}$ E5V and 3.3 V through the VIN on CN7 ST morpho connector, or 3.3 V on CN6 Arduino connector. It is possible to program the MCU after the ST-LINK part is cut by connecting wires between CN4 on the ST-LINK board and SWD signals on connector CN7 (pin 15, SWCLK; and pin 13, SWDIO).

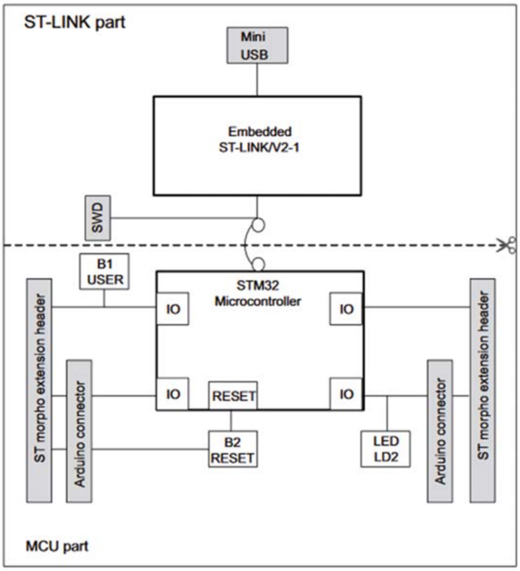


Figure 1.5: Nucleo-L476RG board parts. (©STMicroelecronics. Used with permission).

The components on the top side of the board are shown in Figure 1.6. Some of the important components that you may need to know their locations on the board are:

- CN1: mini USB socket
- CN2: ST-LINK/Nucleo selector
- B1: User pushbutton
- B2: Reset button
- LD1: red/green communications LED
- LD2: green user LED
- LD3: red power LED
- Arduino connectors
- ST morpho connectors

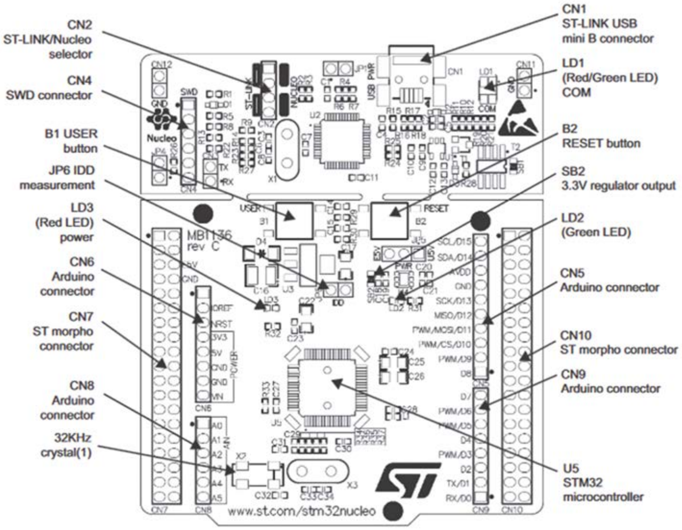


Figure 1.6: Components on the top side of the board.
(©STMicroelecronics. Used with permission).]

The bottom side of board is shown in Figure 1.7.

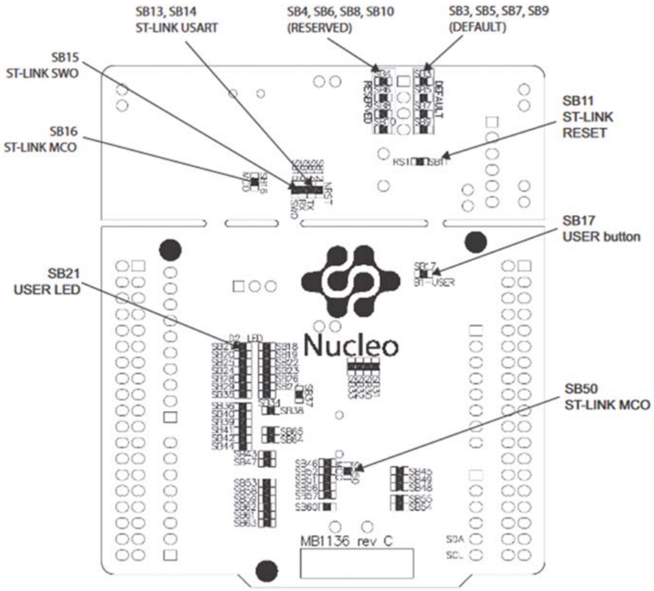


Figure 1.7: Bottom side of the board. (©STMicroelecronics. Used with permission).

1.3.2 The power supply

Power to the board is normally supplied through the mini USB connector CN1 using a USB cable, connected to a PC. It is also possible to use an external power source V_{IN} (7 V to 12 V), E5V (5 V) or +3.3 V.

Powering through the mini USB connector

When powered through the CN1 mini USB port (U5V), a jumper must be connected between pin 1 and pin 2 of jumper JP5 on the MCU board as shown in Figure 1.8. Notice that this is the default state of this jumper.

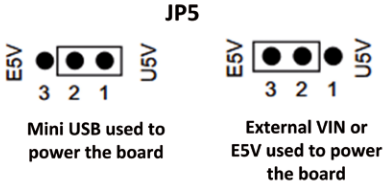

JP5
E5V
U5V
3 2 1
Mini USB used to power the board
E5V
U5V
3 2 1
External VIN or E5V used to power the board

Figure 1.8: Powering the board through the mini USB port.

Jumper JP1 on the ST-LINK board can be configured depending on the maximum current consumption of the MCU board. This jumper should be OFF (default state) for an allowed current of 300 mA, and it should be ON for an allowable current of 100 mA. If the total current consumption of the board exceeds 300 mA then it is important that an external power supply must be used through VIN or E5V.

Powering through external inputs

External power can be applied through VIN or E5V inputs. When power is to be supplied by an external power supply, the following jumpers must be configured:

- jumper JP5 pin 2 and pin 3 connected;
- jumper JP1 OFF (removed).

V_{IN} can be supplied through pin 8 of connector CN6 or pin 24 of connector CN7. The voltage range must be between 7 VDC to 12 VDC. The maximum current capacity should be 800 mA when the input voltage is 7 V, 450 mA when the voltage is between 7 V and 9 V, and 250 mA when the voltage is greater than 9 V and less than 12 V.
E5V can be supplied through pin 6 of connector CN7. The voltage range is 4.75 V to 5.25 V, with maximum current capacity of 500 mA.
When power is supplied through V_{IN} or E5V, it is still possible to use the USB connector for communication, programming or debugging. In this situation, the board must first be powered using VIN or E5V and then the USB cable should be connected to the PC.
The following procedure should be followed if external power will be used and at the same time the USB connector will be used for communication/programming/debugging:

- connect pin 2 and pin 3 of jumper JP5;
- remove jumper JP1;
- connect external power V_{IN} or E5V;
- make sure that LD3 is turned ON;
- connect the USB connector to your PC.

External power input +3.3 V
It is also possible to power the board using external +3.3 V power supply. The range of this supply must be between +3 V to +3.6 V. When powered through the +3.3 V, the ST-LINK board is not powered and therefore the programming and debugging features are not available. This is usually the case when the ST-LINK part of the PCB has been cut. The +3.3 V external power supply must be applied to pin 4 of connector CN6.
Notice that when powered by USB, VIN or E5V, +5 V power is available at pin 5 of CN6 or pin 18 of CN7. These pins for example can be used to provide power to an extension board (e.g. to an Arduino).

1.3.3 The LEDs

LD1 is a tricolour LED (green, orange, red) providing visual indication about the ST-LINK communication. The default colour of this LED is red and it turns green when communication is in progress between the PC and the ST-LINK. The various states of LD1 are:

- slow blinking Red: at power-ON before USB initialization;
- fast blinking Red: after first communication between PC and ST-LINK/V2;
- red ON: initialization between PC and ST-LINK/V2 is complete;
- green ON: after successful target communication initialization;
- blinking Red/Green: during communication with target;
- green ON: communication finished successfully;
- orange ON: communication failure.

LD2 is a green colour user-controlled LED. This LED is connected to pin 11 of connector CN10 (STM32 I/O pin PA_5) on the Nucleo-L476RG board. The Arduino D13 port pin is also connected to LD2. Applying logic 1 turns ON the LED.
LD3 is the red power LED that indicates when +5 V power is available on the MCU board.

1.3.4 Pushbutton switches

There are two pushbutton switches on the MCU board called B1 and B2. B1 is the user button and is connected to STM32 I/O pin PC_13 (pin 23 on connector CN7 on the Nucleo-L476RG board). B2 is the Reset button used to reset the MCU.

1.3.5 Jumper JP6

This jumper is labelled IDD and can be used to measure the current consumption of the MCU by removing the jumper and then connecting an ammeter to the jumper pins. This jumper is ON by default.

1.3.6 The ST-LINK/V2-1

The ST-LINKV2-1 programming and debugging tool is integrated in the Nucleo boards and it makes the boards Mbed Enabled. ST-LINK/V2-1 supports only SWD for STM32 devices. The ST-LINK/V2-1 does not support SWIM interface and the minimum supported application voltage is limited to 3 V. The ST-LINK/V2-1 supports virtual COM port interface on USB, USB software renumeration, mass storage interface on USB, and USB power management request for more than 100 mA power on USB.
There are two different ways to use the embedded ST-LINK/V2-1 depending on the CN2 jumper settings. When both CN2 jumpers are ON (the default state) then the ST-LINK/

V2-1 functions are enabled for on-board programming. When both CN2 jumpers are OFF then the ST-LINK/V2-1 functions are enabled for external CN4 connector (SWD supported). Before connecting the Nucleo-64 board to a Windows PC, a driver for ST-LINK/V2-1 must be installed. This can be downloaded from the following site. You will have to register at the site so that you can download the driver. At the time of writing this book the driver was called **en.stsw-link009.zip** and found here:
http://www.st.com/en/development-tools/stsw-link009.html#getsoftware-scroll

1.3.7 Input-Output connectors

Figure 1.9 shows the input-output connectors for the Nucleo-L476RG board. Notice that different models may have different pin configurations and you should check the appropriate User Manual for the correct configuration of the model you are using. CN5, CN6, CN8 and CN9 are Arduino Uno compatible connectors. CN7 and CN10 are the ST morpho connectors that carry most of the MCU signals.

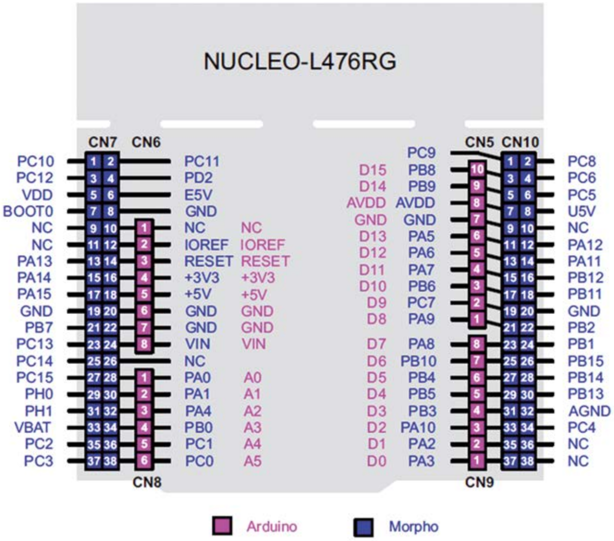


Figure 1.9: Nucleo-L476RG I/O connectors.
(©STMicroelecronics. Used with permission).

The Arduino Uno connector pin configuration on the Nucleo-476RG board is shown in Table 1.6.

Connector	Pin	Pin name	STM32 pin	Function
CN6 POWER	1	NC	-	-
	2	IOREF	-	3.3V Ref
	3	RESET	NRST	RESET
	4	+3.3V	-	3.3 V input/output
	5	+5V	-	5 V output
	6	GND	-	Ground
	7	GND	-	Ground
	8	VIN	-	Power input
CN8 ANALOG	1	A0	PA0	ADC12_IN5
	2	A1	PA1	ADC12_IN6
	3	A2	PA4	ADC12_IN9
	4	A3	PB0	ADC12_IN15
	5	A4	PC1	ADC123_IN2
	6	A5	PC0	ADC123_IN1
RIGHT CONNECTORS				
CN5 DIGITAL	10	D15	PB8	I2C1_SCL
	9	D14	PB9	I2C1_SDA
	8	AREF	-	AVDD
	7	GND	-	GND
	6	D13	PA5	SPI1_SCK
	5	D12	PA6	SPI1_MISO
	4	D11	PA7	TIM17_CH1 or SPI1_MOSI
	3	D10	PB6	TIM4_CH1 or SPI1_CS
	2	D9	PC7	TIM3_CH2
	1	D8	PA9	-
CN9 DIGITAL	8	D7	PA8	-
	7	D6	PB10	TIM2_CH3
	6	D5	PB4	TIM3_CH1
	5	D4	PB5	-
	4	D3	PB3	TIM2_CH2
	3	D2	PA10	-
	2	D1	PA2	USART2_TX
	1	D0	PA3	USART2_RX

Table 1.6: Arduino UNO connector pin configuration.

1.3.8 The demo software

The Nucleo boards are shipped with a pre-loaded demo software. To run the demo software on the Nucleo-L476RG board, follow the sequence below:

- check that jumper JP1 is OFF, JP5 is ON, JP6 is ON, and CN2 are ON;
- connect the Nucleo-L476RG board connector CN1 to a PC with a mini USB cable (Type A to mini-B cable). The red LED LD3 (PWR) and LD1 (COM) should light up. LD1 (COM) and green LED LD2 should blink;
- press button B1 (left button);
- click button B1 and you should see the blinking rate of the green LED LD2 change accordingly.

If the demonstration software is run successfully then the board should be ready for use in projects.

1.4 Summary

In this Chapter we have learned about the types of STM Nucleo development boards. In addition, the specifications and configuration of the popular Nucleo-L476RG board have been described in greater detail, and the steps to run the demonstration software on the Nucleo-L476RG are given. This is the development board used in all the projects described in this book.

In the next Chapter we shall be looking at the STM32 processor architecture.

CHAPTER 2 • STM32 Nucleo Processor Architecture

2.1 Overview

In this Chapter we shall be looking at the internal architecture of the STM32 Nucleo processors. This is important for developing efficient programs and projects. The popular STM32L476RGT6 microcontroller used in all the projects in this book will be considered as an example Arm processor.

2.2 Arm processors

The choice of a microcontroller for a particular application depends on many factors such as the following:

- cost
- speed
- power consumption
- size
- number of digital and analogue input/output ports
- digital input-output port current capacity
- analogue port resolution and accuracy
- program and data memory sizes
- interrupt support
- timer support
- USART support
- special bus support (e.g. USB, CAN, SPI, I^{2}C, and so on)
- ease of system development (e.g. programming)
- working voltage

For example, if you need to develop a battery powered device such as a mobile phone or a games device, then very high clock speed as well as long battery life are main requirements. But if you are developing a traffic lights controller then very high performance is not a requirement. In general, as the clock speed goes up so does the power consumption and as a result a trade-off should be made in choosing a microcontroller for a specific application.

Arm has been designing 32-bit processors for over 20 years and in the last few years they have also started to offer 64-bit designs. In actual fact, Arm is a company specialized in designing the processor architecture and they do not manufacture or sell processor chips. Arm makes money by licensing their designs to chip manufacturers. The manufacturers use the core Arm processors (e.g. the core CPU) and integrate with their own peripherals to end up in a complete microcontroller chip. Arm is then given royalty fees for each chip manufactured by the third-party companies. Companies that use Arm core processors include Apple, Atmel, Broadcom, Cypress Semiconductors, Freescale Semiconductors, Analog Devices, Nvidia, NXP, Samsung Electronics, Texas Instruments, Qualcomm, Renesas, and many others.

Arm was originally known as the Acorn Computers and they have developed the first Acorn RISC Machine (Arm) architecture in the 1980s to use in their personal computers. First Arm processors were co-processor modules used in the BBC Micro series. After failing to find suitable high-performance microprocessor chips in the market, Acorn decided to design their own processors. In 1990, the research section of Acorn formed the ARM Ltd.

Currently Arm is the most widely used processor in terms of quantity manufactured. Over 50 billion Arm processors have been produced as of 2014, where 10 billion were produced in 2013 only. The Arm 32-bit architecture is the most widely used in mobile devices where about 98% of all mobile phones sold in the year 2005 used at least one Arm processor. The Arm architecture is known to offer the best MIPS-to-watts ratio as well as MIPS-to-$ ratio in the industry, and the smallest CPU size.

The small size, low power consumption, and low cost make Arm an ideal processor in embedded applications. Arm processors are based on an instruction set called Thumb. With clever design this instruction set takes 32-bit instructions and compresses them down to 16-bits, thus reduces the hardware size, which also reduces the overall cost. The processor makes use of multistage pipelined architecture that is easier to learn, build, and program.

Arm processors are based on the RISC (Reduced instruction Set Computer) architecture and are available as 32-bit or 64-bit multi-core structures. RISC processors, as opposed to CISC (Complex Instruction Set Computer) processors have smaller number of instructions and fewer transistors (hence smaller die size), as a result they can operate at higher speeds. Unimportant or not frequently used instructions are removed, and the pathways are optimized to result in superior performance.

It is important to realize that Arm's core architecture is only a processor and it does not include graphics, input-output ports, USB, serial communication, wireless connectivity, or any other form of peripheral modules. Chip manufacturers build their systems around the Arm core design and therefore different manufacturers offer different types of Arm based microcontrollers.

Over the last 20 years or so Arm (ARM) had developed many 32-bit processors. Figure 2.1 shows some of the popular members of the Arm processor family. The first successful member was the ARM7TDMI which had high code density and low power consumption. This processor, based on the Von Neumann architecture, was operating at 80 MHz and was used in early mobile phones. ARM9 was developed in 1997 with Harvard architecture and operated with 150 MHz, thus offered higher performance. ARM10 and ARM11 were developed in the years 1999 and 2003, respectively. Both processors were based on the Harvard architecture. ARM10 operated at 260 MHz and ARM11 operated at 335 MHz. Around the year 2003 Arm decided to improve their market share by developing new series of high-performance processors. As a result, the Cortex family or processors were created. The Cortex family consists of three processor families: Cortex-M, Cortex-R, and Cortex-A. We shall now briefly look at these families.

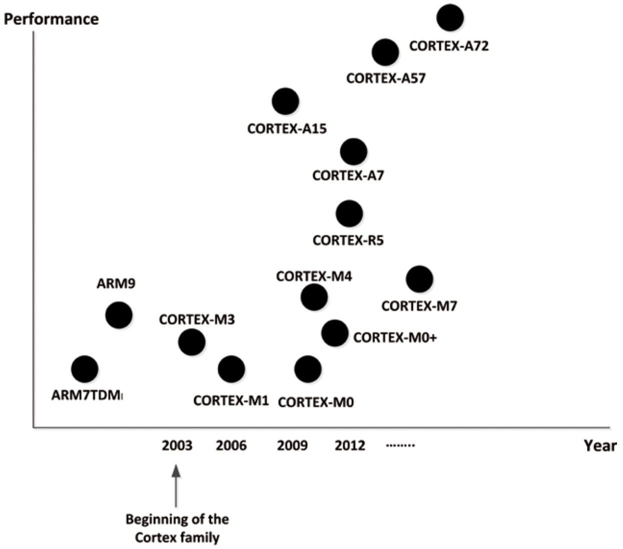


Figure 2.1: Overview of the Arm processor family.

2.2.1 Cortex-M

Cortex-M series are built around the ARMv6-M architecture (Cortex-M0 and Cortex-M0+) and the ARMv7-M architecture (Cortex-M3 and Cortex-M4). These processors are specifically designed for the microcontroller market, offering quick and deterministic interrupt responses, low power consumption, low cost, fairly high performance, and ease of use. The Cortex-M3 and Cortex-M4 are very similar in architecture and have the same instruction set (Thumb 2) with the difference that the Cortex-M4 offers digital signal processing (DSP) capability and has optional floating point unit (FPU). Cortex-M4 with its DSP and floating-point capability is an ideal processor for the IoT and wearable applications. For cost sensitive and lower performance applications the Cortex-M0 or the Cortex-M0+ can be used. The Cortex-M0 processor has small gate count (12K gates) and consumes only 12.5 µW/MHz. The Cortex-M0+ consumes only 9.85 µW/MHz and is based on a subset of the Thumb 2 instruction set and its performance is slightly above that of Cortex-M0 and below that of the Cortex-M3 and Cortex-M4. Cortex-M7 is a high performance processor that can handle fast DSP and single or double precision floating point operations and is mainly used in applications where higher performance than the Cortex-M4 is required.

2.2.2 Cortex-R

Cortex-R series are real-time higher performance processors than the Cortex-M and some members are designed to operate at high clock rates in excess of 1 GHz. These processors are commonly used in hard-disk controllers, network devices, automotive applications, and in specialized high-speed microcontroller applications. Cortex-R4 and Cortex-R5 are the early members and can be used at clock speeds of up to 600 MHz. Cortex-R7 is a newer member that incorporates 11-stage pipeline for high performance, and it can operate in excess of 1 GHz. Although the Cortex-R processors are high performance their architecture is complex and these processors consume high power, making them unsuitable for use in mobile battery powered devices.

2.2.3 Cortex-A

Cortex-A are the highest performance Arm processors designed for use with real-time operating systems in mobile applications such as in mobile phones, tablets, GPS devices and so on. These processors support advanced features for operating systems such as Android, ioS, Linux, Windows etc. In addition, advanced memory management is supported with virtual memory. Early members of the family included processors such as Cortex-A5 to Cortex-A17, based on the ARMv7-A architecture. Latest members of the family are the Cortex-A50 and Cortex-A72 series designed for low power and very high-performance mobile applications. These processors are built using the ARMv8-A architecture which offers 64-bit energy-efficient operation with the capability of more than 4 GB of physical memory.

2.2.4 Cortex-M processor comparison

A comparison of the various Cortex-M series processors is given in Table 2.1. As can be seen from this table, Cortex-M0 and Cortex-M0+ are used at low speed and low power consumption applications. Cortex-M1 is optimized for use in programmable gate array applications. Cortex-M3 and Cortex-M4 are medium power processors used in microcontroller applications with the Cortex-M4 supporting DSP and floating-point arithmetic operations. Cortex-M7 is a high-performance member of the family which is used in applications requiring higher performance than the Cortex-M4.

Processor	Description
Cortex-M7	High performance processor, used in applications where Cortex-M4 is not fast enough, supports DSP and single and double precision arithmetic
Cortex-M4	Similar architecture as the Cortex-M3 but includes DSP and floating point arithmetic, used in high-end microcontroller type applications
Cortex-M3	Very popular, low power consumption, medium performance, debug features, used in microcontroller type applications
Cortex-M1	Designed mainly for programmable gate array applications
Cortex-M0+	Lower power consumption and higher performance than the Cortex-M0
Cortex-M0	Low power consumption, low to medium performance, smallest Arm processor

Table 2.1: Cortex-M processor comparison.

2.2.5 Processor performance measurement

Processor performance is usually measured using benchmark programs. There are many benchmark programs available and one should exercise care when comparing the performance of various processors as the performance depends upon many external factors such as the efficiency of the compiler used and the type of operation performed for the measurement.

Many attempts were made in the past to measure the performance of a processor and quote it as a single number. For example, MOPS, MFLOPS, Dhrystone, DMIPS, BogoMIPS, and so on. Nowadays, CoreMark is one of the most commonly used benchmark programs used to indicate the processor performance. CoreMark is developed by Embedded Microprocessor Benchmark Consortium (EEMBC, www.eembc.org/coremark) and is one of the most reliable performance measurement tools available.

Table 2.2 shows the CoreMark results for some of the commonly used microcontrollers. As can be seen from this table, Cortex-M7 achieves 5.01 CoreMark/MHz, while the PIC18 microcontroller achieves only 0.04 CoreMark/MHz.

Processor	CoreMark/MHz
Intel Zeus	8.01
Cortex-M7	5.01
Cortex-A9	4.15
Espressif ESP32	4.13
Cortex-M4	3.40
Cortex-M3	3.32
Cortex-M0+	2.49
Espressif ESP8266	2.38
Cortex-M0	2.33
dsPIC33	1.89
PIC24	1.88
MSP430	1.11
PIC18	0.04

Table 2.2: CoreMark/MHz for some commonly used microcontrollers.

2.2.6 Cortex-M compatibility

Processors in the Cortex family are upward compatible with each other. Cortex-M0 and Cortex-M0+ processors are based on the ARMv6-M architecture, using the Thumb instruction set. On the other hand, Cortex-M3, Cortex-M4 and Cortex-M7 are based on the ARMv7-M architecture, using the Thumb 2 instruction set which is a superset of the Thumb instruction set. Although the architectures are different, software developed on the Cortex-M0 and Cortex-M0+ processors can run on the Cortex-M3, Cortex-M4, and Cortex-M7 processors without any modifications provided the required memory and input-output ports are available.

2.2.7 Choice of an STM32 processor

Developers may find it difficult to choose the correct STM32 processor for the correct project. STMicroelectronics offers a MCU selection tool which makes it easy to choose the processor with the features you are interested in. This tool is available at the following website: http://www.st.com/web/en/catalog/mmc/FM141/SC1169

2.3 The STM32L476RGT6 microcontroller

In this book we shall be using the highly popular Arm microcontroller STM32L476RGT6 in the STM Nucleo-L476RG development board. In this Chapter we shall be looking at the features of the STM32L476RGT6 microcontroller. This microcontroller type STM32L476 belongs to the family 'STM32L476xxxx' where R is for 64 pins, G is for 1 MB flash memory, T is for LQPF package, and 6 is for industrial temperature range –40º to+85ºC. The internal architecture of this microcontroller is very complex and we shall only look at the important modules used in most projects, such as I/O, timers, ADC converter and DAC converter, interrupts, I²C, USART, and so on. Interested readers can get detailed information from the manufacturer's datasheets available for download on the Internet. Readers should note that the structure of the STM32 family of processors is similar and after becoming familiar with the STM32L476RGT6, it should not be too difficult to learn the structure of the others in the family.

2.3.1 Basic features of the STM32L476RGT6

The STM32L476RGT6 microcontroller is based on the Cortex-M4 architecture and has the following basic features (Further information can be obtained from the ST datasheet STM-32L476xx, DocID025976, 2017):

- Arm Cortex-M4 32-bit CPU with FPU
- 80 MHz max CPU frequency
- Operating voltage 1.71 to 3.6 V
- 1.25 DMIPS/MHZ performance (Drystone 2.1)
- 4 to 48 MHz crystal oscillator
- 32 MHz crystal oscillator for RTC
- Internal 16 MHz RC oscillator
- Internal 32 kHz RC oscillator
- Internal 100 kHz to 48 MHz oscillator
- 3 PLLs for system clock
- 1 MB flash memory
- 128 KB SRAM
- 16× timers
- 3× SPI and 3× I²C interfaces
- 3× USART, 2× UART, 1× LPUART interfaces
- 2× SAI and 1× CAN interfaces
- 4× digital filters for sigma-delta modulators
- 1× Real Time Clock (RTC)
- LCD interface (8 × 28 or 4 × 32)
- 51× General Purpose I/O (GPIOs)
- 12× capacitive sensing channels
- 3× 16 channel 12-bit ADC
- 2× 12-bit DAC
- 14 channel DMA
- Temperature sensor
- Random number generator
- Cyclic Redundancy Check (CRC)
- USB, SDMMC and SWPMI communication interface
- 2× analogue comparator and 2× operational amplifiers
- 64-pin package

2.3.2 Internal block diagram

Figure 2.2 shows the functional block diagram of the STM32L476xx processor. The 80 MHz Arm Cortex-M4 CPU is shown at the middle top of the figure. The digital connectivity modules are at the top left, the GPIOs are at the bottom left, the LCD interface is at the top