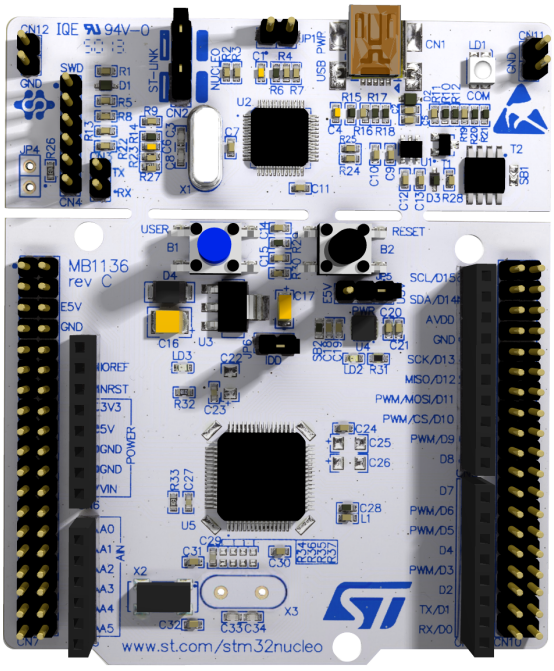
UM1724
User manual

STM32 Nucleo-64 board

Introduction

The STM32 Nucleo-64 board (NUCLEO-F030R8, NUCLEO-F070RB, NUCLEO-F072RB, NUCLEO-F091RC, NUCLEO-F103RB, NUCLEO-F302R8, NUCLEO-F303RE, NUCLEO-F334R8, NUCLEO-F401RE, NUCLEO-F410RB, NUCLEO-F411RE, NUCLEO-F446RE, NUCLEO-L053R8, NUCLEO-L073RZ, NUCLEO-L152RE, NUCLEO-L452RE, NUCLEO-L476RG) provides an affordable and flexible way for users to try out new concepts and build prototypes with the STM32 microcontrollers in LQFP64 package, choosing from the various combinations of performance, power consumption and features. The Arduino™ Uno V3 connectivity support and the ST morpho headers allow to expand easily the functionality of the Nucleo open development platform with a wide choice of specialized shields. The STM32 Nucleo board does not require any separate probe as it integrates the ST-LINK/V2-1 debugger and programmer. The STM32 Nucleo board comes with the STM32 comprehensive software HAL library together with various packaged software examples, as well as direct access to the ARM® mbed™ online resources at http://mbed.org/.

Figure 1. STM32 Nucleo-64 board (1)



1. Picture is not contractual.

Content

List of Tables

List of Figures

1 Features

The STM32 Nucleo board offers the following features:

- STM32 microcontroller in LQFP64 package
- Two types of extension resources
  - Arduino™ Uno V3 connectivity
  - ST morpho extension pin headers for full access to all STM32 I/Os
- ARM® mbed™ (see http://mbed.org)
- On-board ST-LINK/V2-1 debugger and programmer with SWD connector
  - Selection-mode switch to use the kit as a standalone ST-LINK/V2-1
- Flexible board power supply:
  - USB VBUS or external source (3.3V, 5V, 7 - 12V)
  - Power management access point
- Three LEDs:
  - USB communication (LD1), user LED (LD2), power LED (LD3)
- Two push-buttons: USER and RESET
- USB re-enumeration capability. Three different interfaces supported on USB:
  - Virtual COM port
  - Mass storage
  - Debug port
- Comprehensive free software HAL library including a variety of software examples

2 Product marking

Evaluation tools marked as "ES" or "E" are not yet qualified and therefore they are not ready to be used as reference design or in production. Any consequences deriving from such usage will not be at ST charge. In no event, ST will be liable for any customer usage of these engineering sample tools as reference design or in production.

"E" or "ES" marking examples of location:

- On the targeted STM32 that is soldered on the board (for illustration of STM32 marking, refer to the section “Package information” of the STM32 datasheet available at *www.st.com*).
- Next to the evaluation tool ordering part number, that is stuck or silk-screen printed on the board.

3 Ordering information

Table 1 lists the order codes and the respective targeted STM32.

Table 1. Ordering information

Order code	Targeted STM32
NUCLEO-F030R8	STM32F030R8T6
NUCLEO-F070RB	STM32F070RBT6
NUCLEO-F072RB	STM32F072RBT6
NUCLEO-F091RC	STM32F091RCT6
NUCLEO-F103RB	STM32F103RBT6
NUCLEO-F302R8	STM32F302R8T6
NUCLEO-F303RE	STM32F303RET6
NUCLEO-F334R8	STM32F334R8T6
NUCLEO-F401RE	STM32F401RET6
NUCLEO-F410RB	STM32F410RBT6
NUCLEO-F411RE	STM32F411RET6
NUCLEO-F446RE	STM32F446RET6
NUCLEO-L053R8	STM32L053R8T6
NUCLEO-L073RZ	STM32L073RZT6
NUCLEO-L152RE	STM32L152RET6
NUCLEO-L452RE	STM32L452RET6
NUCLEO-L476RG	STM32L476RGT6

The meaning of the NUCLEO-TXXXRY codification is explained in *Table 2* with an example:

Table 2. Codification explanation

NUCLEO-TXXXRY	Description	Example: NUCLEO-L452RE
TXXX	STM32 product line	STM32L452
R	STM32 package pin count	64 pins
Y	STM32 Flash memory size (8 for 64 Kbytes, B for 128 Kbytes, C for 256 Kbytes, E for 512 Kbytes, G for 1 Mbyte, Z for 192 Kbytes)	512 Kbytes

The order code is printed on a sticker placed at the top or bottom side of the board.

4 Conventions

Table 3 provides the conventions used for the ON and OFF settings in the present document.

Table 3. ON/OFF conventions

Convention	Definition
Jumper JP1 ON	Jumper fitted
Jumper JP1 OFF	Jumper not fitted
Solder bridge SBx ON	SBx connections closed by solder or 0 ohm resistor
Solder bridge SBx OFF	SBx connections left open

In this document the references are “STM32 Nucleo board” and “STM32 Nucleo boards” for all information that is common to all sale types.

5 Quick start

The STM32 Nucleo board is a low-cost and easy-to-use development platform used to quickly evaluate and start a development with an STM32 microcontroller in LQFP64 package.

Before installing and using the product, accept the Evaluation Product License Agreement from the www.st.com/epla webpage.

For more information on the STM32 Nucleo board and to access the demonstration software, visit www.st.com/stm32nucleo website.

5.1 Getting started

Follow the sequence below to configure the STM32 Nucleo board and launch the demo software:

1. Check the jumper position on the board, JP1 off, JP5 (PWR) on U5V, JP6 on (IDD), CN2 on (NUCLEO) selected.
2. For correct identification of all device interfaces from the host PC, install the Nucleo USB driver available from the www.st.com/stm32nucleo webpage, prior to connecting the board.
3. Connect the STM32 Nucleo board to a PC with a USB cable 'Type-A to Mini-B' through USB connector CN1 to power the board. The red LED LD3 (PWR) and LD1 (COM) should light up. LD1 (COM) and green LED LD2 should blink.
4. Press button B1 (left button).
5. Observe the blinking frequency of the three LEDs LD1 to LD3, by clicking on the button B1.
6. The demonstration software and several software examples on how to use the STM32 Nucleo board features are available at the www.st.com/stm32nucleo webpage.
7. Develop the application using the available examples.

5.2 System requirements

- Windows® OS (XP, 7, 8) or Linux 64-bit or Mac OS® X
- USB Type-A to Mini-B cable

5.3 Development toolchains

- ARM® Keil®: MDK-ARM(a)
- IAR™ EWARM(a)
- GCC-based IDEs (free AC6: SW4STM32, Atollic® TrueSTUDIO®(a) and others)
- ARM® mbed™ online

a. On Windows® only.

5.4 NUCLEO-L476RG bootloader limitations

Boot from system Flash memory results in executing **bootloader** code stored in the system Flash memory, protected against write and erase. This allows in-system programming (ISP), that is, flashing the STM32 user Flash memory. It also allows writing data into RAM. The data come in via one of the communication interfaces such as USART, SPI, I^2C bus, USB or CAN.

Bootloader version can be identified by reading Bootloader ID at the address 0x1FFF6FFE.

The STM32L476RGT6 part soldered on the NUCLEO-L476RG main board is marked with a date code, corresponding to its date of manufacturing. STM32L476RGT6 parts with the date code prior or equal to week 22 of 2015 are fitted with **bootloader V 9.0,** affected by the limitations to be worked around, as described hereunder. Parts with the date code starting from week 23 of 2015 contain bootloader V 9.2 in which the limitations no longer exist.

To locate the visual date code information on the STM32L476RGT6 package, refer to the section “Package information” of the datasheet (DS10198) available at *www.st.com.* Date code related portion of the package marking, takes Y WW format, where Y is the last digit of the year and WW is the week. For example, a part manufactured in week 23 of 2015 bares the date code 5 23.

Bootloader ID of the bootloader V 9.0 is 0x90.

The following limitations exist in the bootloader V 9.0:

1. RAM data get corrupted when written via USART/SPI/I^2C/USB interface

Description:

Data write operation into RAM space via USART, SPI, I^2C bus or USB results in wrong or no data written.

Workaround:

To correct the issue of wrong write into RAM, download STSW-STM32158 bootloader V 9.0 patch package from the *www.st.com* website and load "Bootloader V9.0 SRAM patch" to the STM32, following the information in readme.txt file available in the package.

2. User Flash memory data get corrupted when written via CAN interface

Description:

Data write operation into user Flash memory space via CAN interface results in wrong or no data written.

Workaround:

To correct the issue of wrong write into Flash memory, download STSW-STM32158 bootloader V 0.9 patch package from the *www.st.com* website and load "Bootloader V9.0 CAN patch" to the STM32, following the information in readme.txt file available in the package.

5.5 Hardware configuration variants

The board can be delivered with different configurations of the oscillator of the target STM32. For all the details concerning high-speed configurations of the oscillator refer to *Section 6.7.1*. For all the details concerning low-speed configurations of the oscillator refer to *Section 6.7.2*.

6 Hardware layout and configuration

The STM32 Nucleo board is designed around the STM32 microcontrollers in a 64-pin LQFP package.

Figure 2 shows the connections between the STM32 and its peripherals (ST-LINK/V2-1, push-button, LED, Arduino connectors and ST morpho connector).

Figure 3 and *Figure 4* show the location of these features on the STM32 Nucleo board. *Figure 5* shows the mechanical dimension of the STM32 Nucleo board.

Figure 2. Hardware block diagram

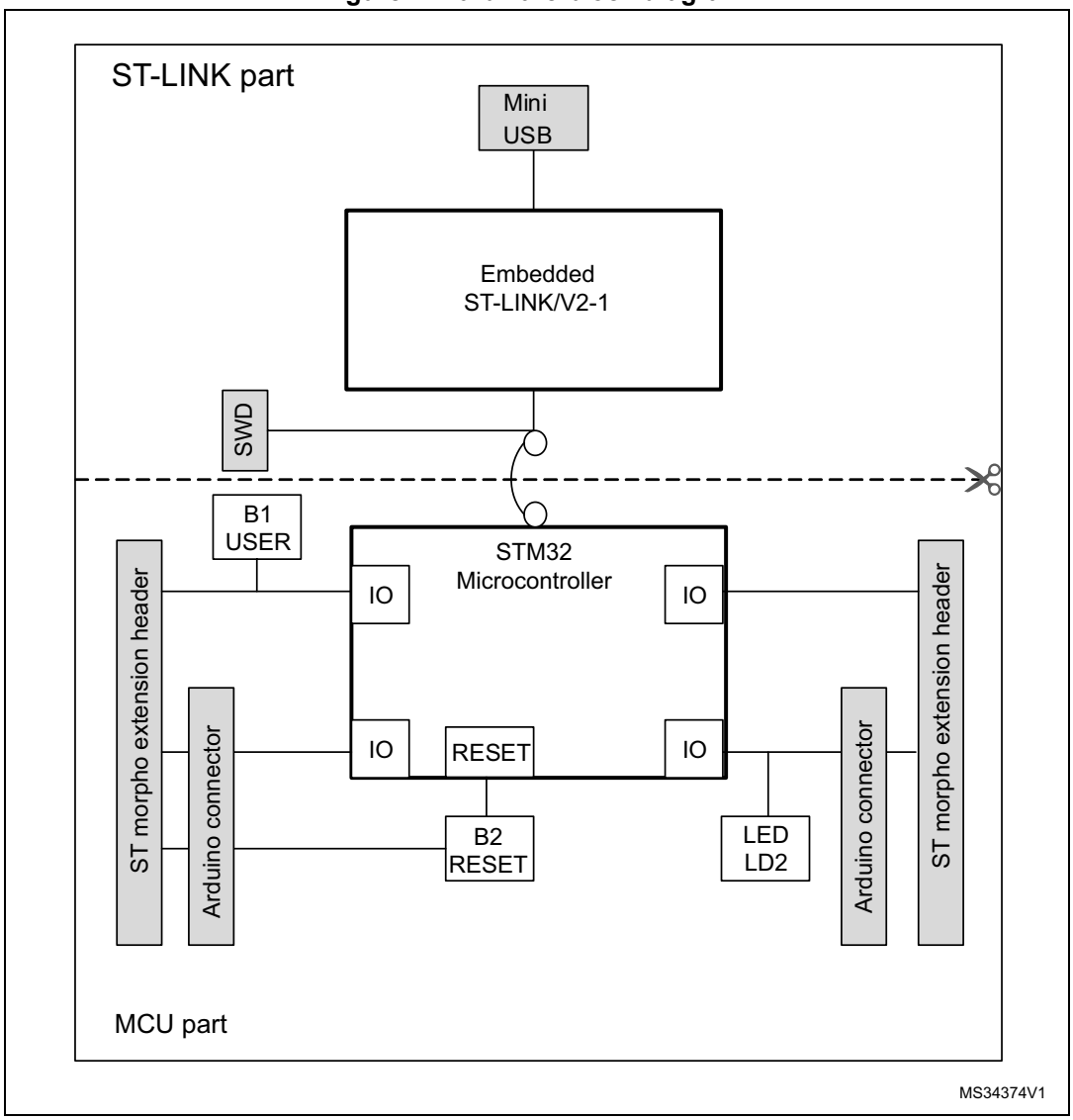
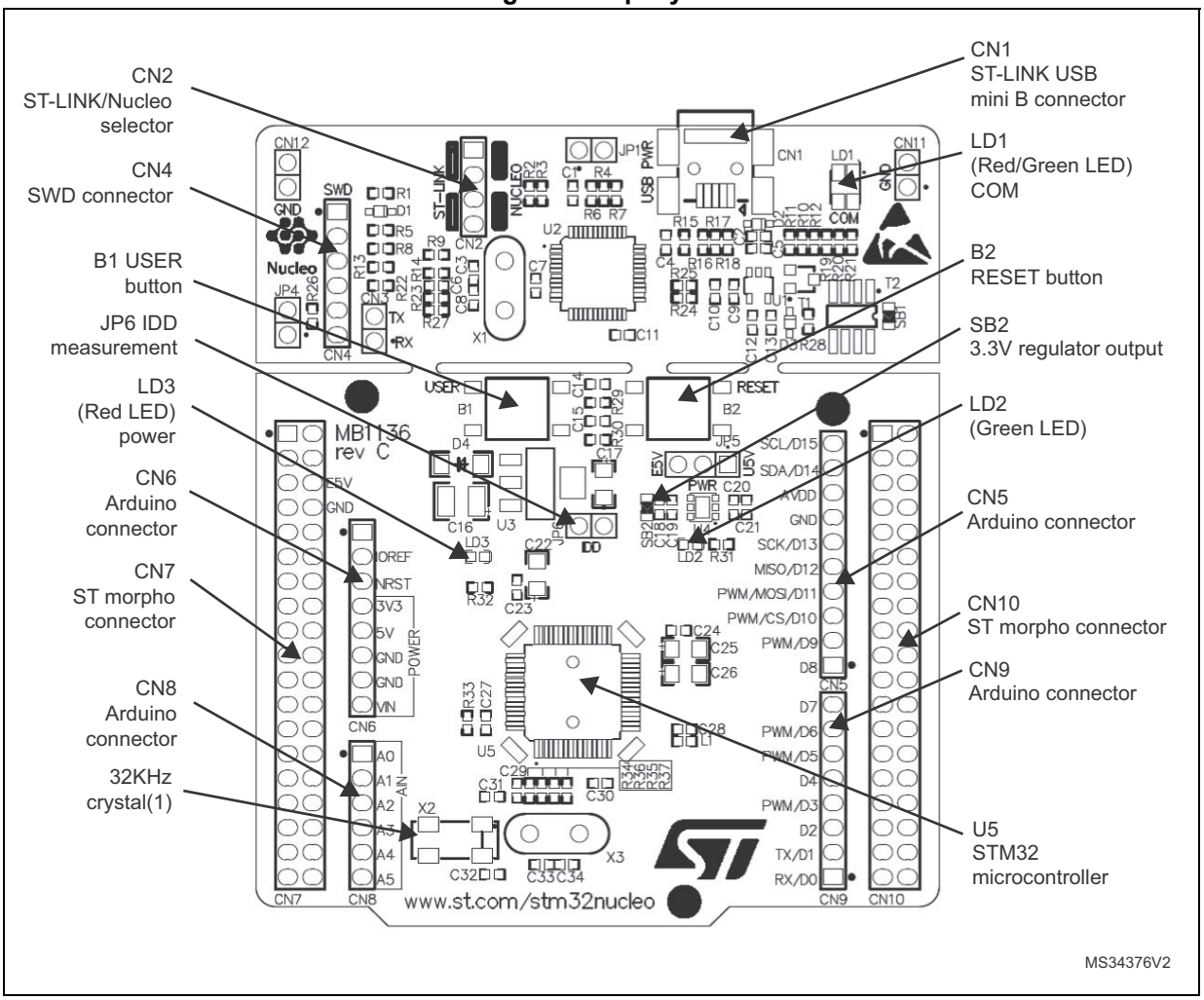


Figure 3. Top layout



1. Crystal may be present or not depending on board version, refer to *Section 6.7.2*.

Figure 4. Bottom layout

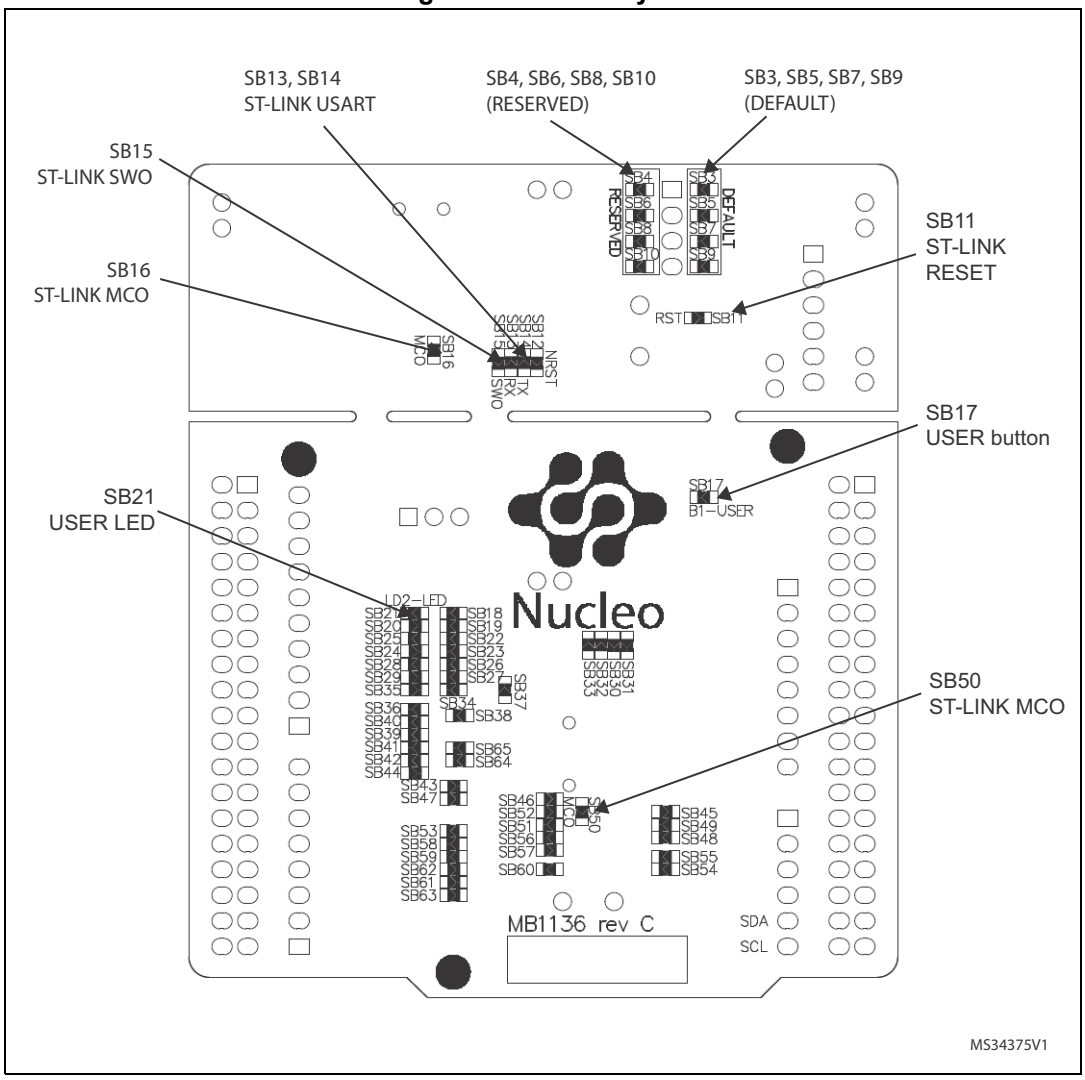
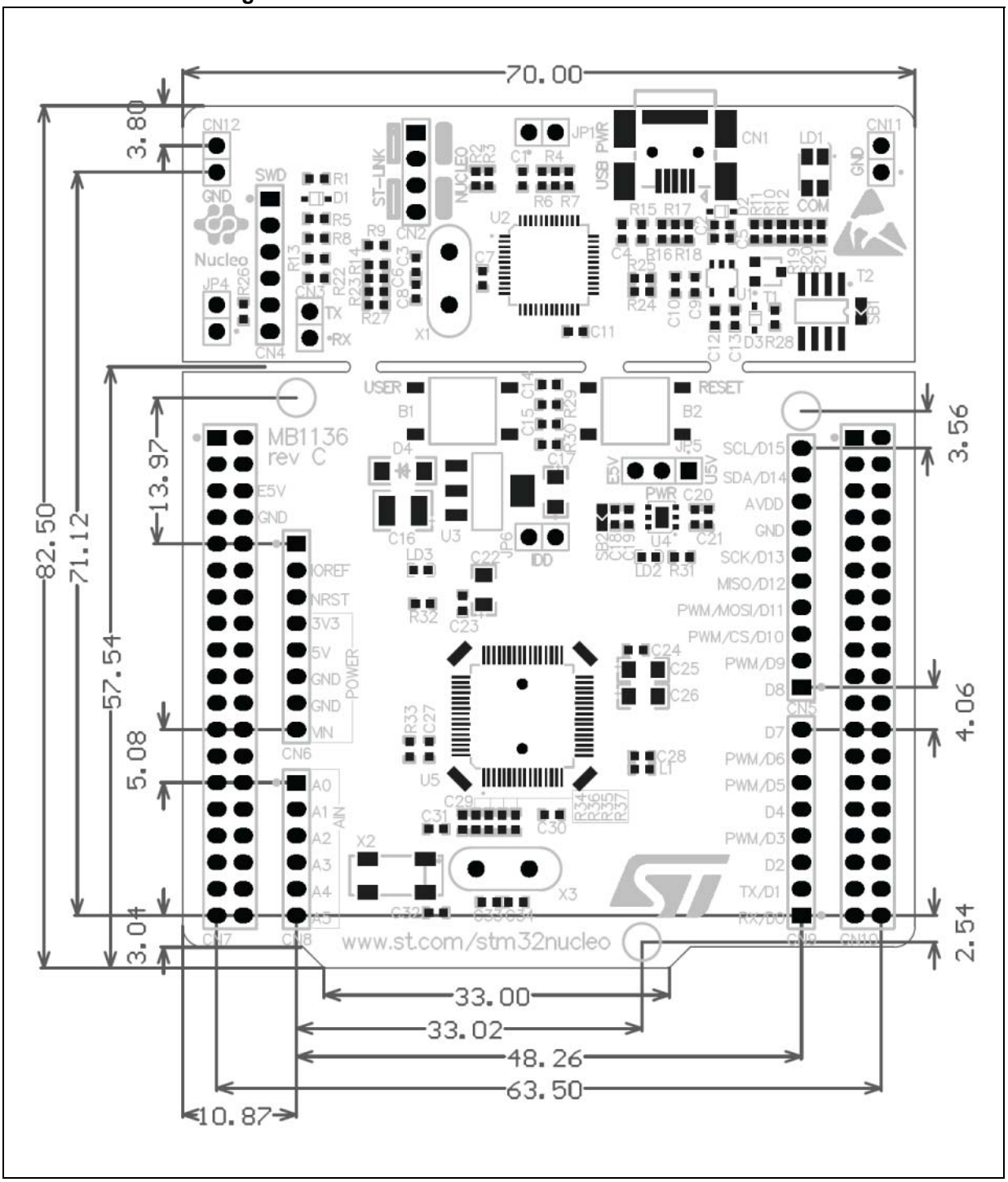


Figure 5. STM32 Nucleo board mechanical dimensions



6.1 Cuttable PCB

The STM32 Nucleo board is divided into two parts: ST-LINK part and target STM32 part. The ST-LINK part of the PCB can be cut out to reduce the board size. In this case the remaining target STM32 part can only be powered by VIN, E5V and 3.3V on ST morpho connector CN7 or VIN and 3.3V on Arduino connector CN6. It is still possible to use the ST-LINK part to program the main STM32 using wires between CN4 and SWD signals available on ST morpho connector (SWCLK CN7 pin 15 and SWDIO CN7 pin 13).

6.2 Embedded ST-LINK/V2-1

The ST-LINK/V2-1 programming and debugging tool is integrated in the STM32 Nucleo board.

The ST-LINK/V2-1 makes the STM32 Nucleo boards mbed enabled.

The embedded ST-LINK/V2-1 supports only SWD for STM32 devices. For information about debugging and programming features refer to *ST-LINK/V2 in-circuit debugger/programmer for STM8 and STM32* User manual (UM1075), which describes in details all the ST-LINK/V2 features.

The changes versus ST-LINK/V2 version are listed below.

- New features supported on ST-LINK/V2-1:
  - USB software re-enumeration
  - Virtual COM port interface on USB
  - Mass storage interface on USB
  - USB power management request for more than 100 mA power on USB
- Features not supported on ST-LINK/V2-1:
  - SWIM interface
  - Minimum supported application voltage limited to 3 V
- Known limitation:
  - Activating the readout protection on ST-LINK/V2-1 target prevents the target application from running afterwards. The target readout protection must be kept disabled on ST-LINK/V2-1 boards.

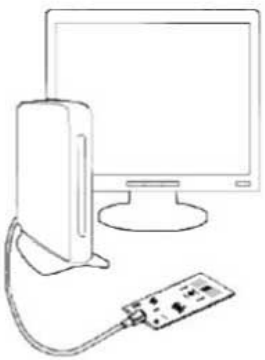
There are two different ways to use the embedded ST-LINK/V2-1 depending on the jumper states (see *Table 4* and *Figure 6*):

- Program/debug the on-board STM32 (*Section 6.2.2*)
- Program/debug an MCU in an external application board using a cable connected to SWD connector CN4 (*Section 6.2.4*).

Table 4. Jumper states

Jumper state	Description
Both CN2 jumpers ON	ST-LINK/V2-1 functions enabled for on board programming (default)
Both CN2 jumpers OFF	ST-LINK/V2-1 functions enabled for external CN4 connector (SWD supported)

Figure 6. Typical configuration



Hardware requirements:
- USB cable type A to mini-B
- computer with Windows XP, 7, 8

Development toolchain:
- IAR EWARM
- Keil MDK-ARM
- GCC-based IDE

MS19052V4

6.2.1 Driver

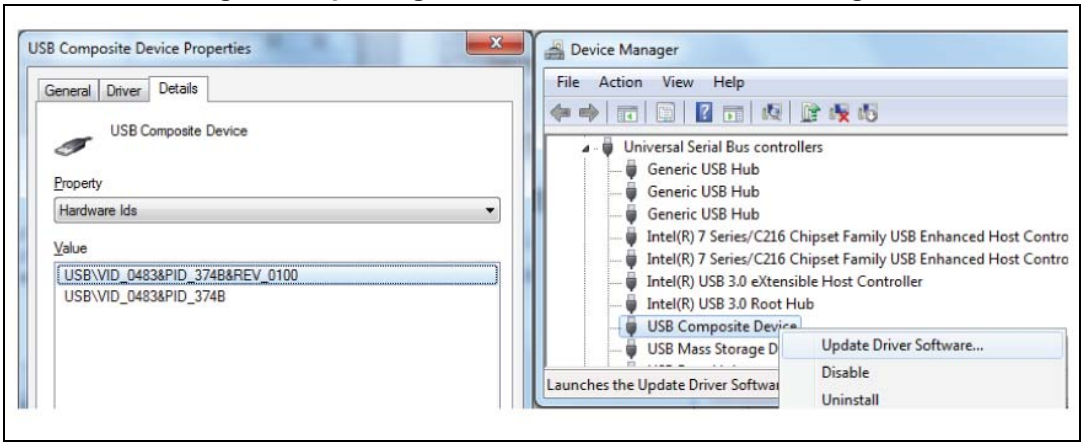
Before connecting the Nucleo-64 board to a Windows 7, Windows 8 or Windows XP PC via USB, a driver for ST-LINK/V2-1 must be installed. It can be downloaded from the *www.st.com* website.

In case the STM32 Nucleo-64 board is connected to the PC before installing the driver, the PC device manager may report some Nucleo interfaces as “Unknown”.

To recover from this situation, after installing the dedicated driver, the association of “Unknown” USB devices found on the STM32 Nucleo-64 board to this dedicated driver, must be updated in the device manager manually.

Note: It is recommended to proceed using USB Composite Device, as shown in Figure 7.

Figure 7. Updating the list of drivers in Device Manager



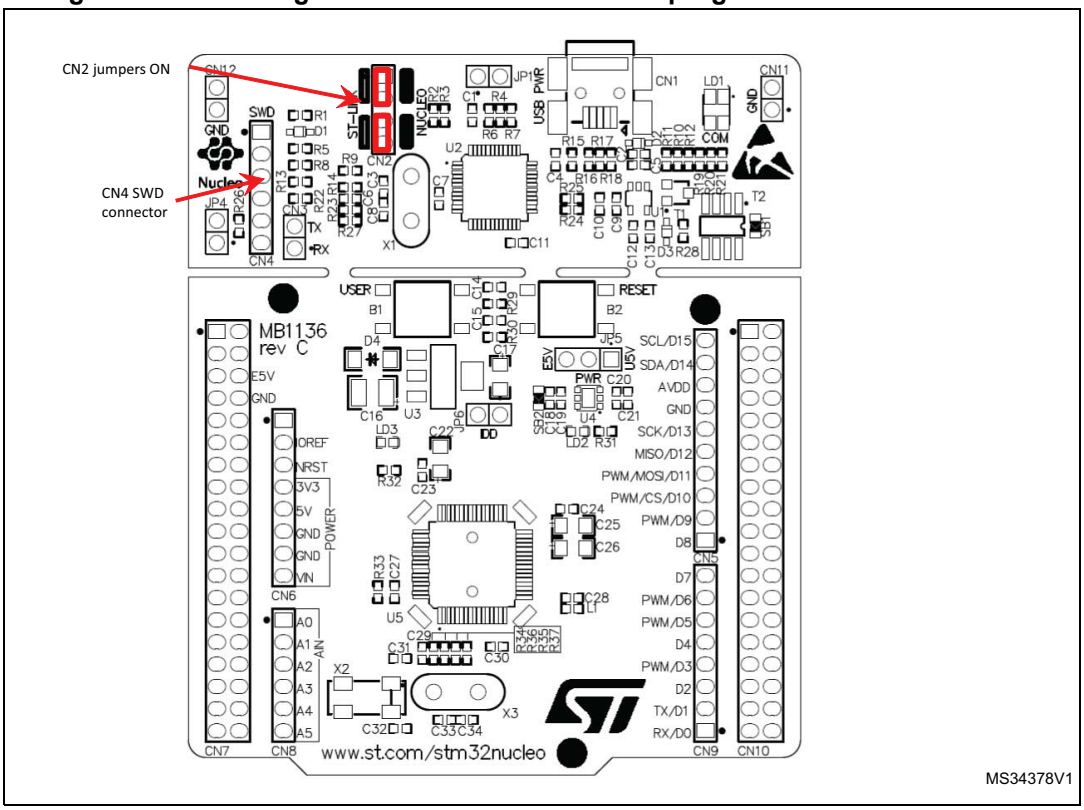
6.2.2 ST-LINK/V2-1 firmware upgrade

The ST-LINK/V2-1 embeds a firmware upgrade mechanism for in-situ upgrade through the USB port. As the firmware may evolve during the life time of the ST-LINK/V2-1 product (for example new functionality, bug fixes, support for new microcontroller families), it is recommended to visit *www.st.com* website before starting to use the STM32 Nucleo board and periodically, in order to stay up-to-date with the latest firmware version.

6.2.3 Using the ST-LINK/V2-1 to program and debug the STM32 on board

To program the STM32 on the board, plug in the two jumpers on CN2, as shown in red in *Figure 8*. Do not use the CN4 connector as this could disturb the communication with the STM32 microcontroller of the STM32 Nucleo board.

Figure 8. Connecting the STM32 Nucleo board to program the on-board STM32



6.2.4 Using ST-LINK/V2-1 to program and debug an external STM32 application

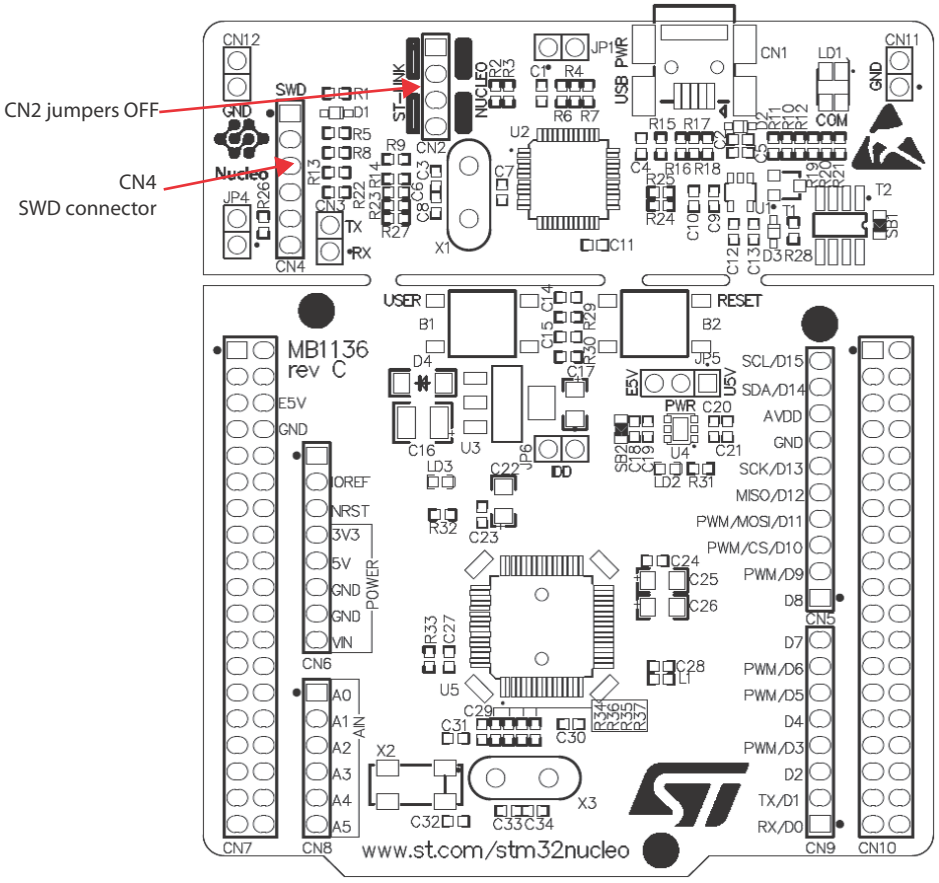
It is very easy to use the ST-LINK/V2-1 to program the STM32 on an external application. Simply remove the two jumpers from CN2 as illustrated in *Figure 9: Using ST-LINK/V2-1 to program the STM32 on an external application*, and connect the application to the CN4 debug connector according to *Table 5*.

Note: SB12 NRST (target STM32 RESET) must be OFF if CN4 pin 5 is used in the external application.

Table 5. Debug connector CN4 (SWD)

Pin	CN4	Designation
1	VDD_TARGET	VDD from application
2	SWCLK	SWD clock
3	GND	ground
4	SWDIO	SWD data input/output
5	NRST	RESET of target STM32
6	SWO	Reserved

Figure 9. Using ST-LINK/V2-1 to program the STM32 on an external application



6.3 Power supply and power selection

The power supply is provided either by the host PC through the USB cable, or by an external source: VIN (7V-12V), E5V (5V) or +3.3V power supply pins on CN6 or CN7. In case VIN, E5V or +3.3V is used to power the STM32 Nucleo board, using an external power supply unit or an auxiliary equipment, this power source must comply with the standard EN-60950-1: 2006+A11/2009, and must be Safety Extra Low Voltage (SELV) with limited power capability.

6.3.1 Power supply input from the USB connector

The ST-LINK/V2-1 supports USB power management allowing to request more than 100 mA current to the host PC.

All parts of the STM32 Nucleo board and shield can be powered from the ST-LINK USB connector CN1 (U5V or VBUS). Note that only the ST-LINK part is power supplied before the USB enumeration as the host PC only provides 100 mA to the board at that time. During the USB enumeration, the STM32 Nucleo board requires 300 mA of current to the host PC. If the host is able to provide the required power, the targeted STM32 microcontroller is powered and the red LED LD3 is turned ON, thus the STM32 Nucleo board and its shield can consume a maximum of 300 mA current, not more. If the host is not able to provide the required current, the targeted STM32 microcontroller and the MCU part including the extension board are not power supplied. As a consequence the red LED LD3 remains turned OFF. In such case it is mandatory to use an external power supply as explained in the next *Section 6.3.2: External power supply inputs: VIN and E5V*.

When the board is power supplied by USB (U5V) a jumper must be connected between pin 1 and pin 2 of JP5 as shown in *Table 8*.

JP1 is configured according to the maximum current consumption of the board when powered by USB (U5V). JP1 jumper can be set in case the board is powered by USB and maximum current consumption on U5V does not exceed 100 mA (including an eventual extension board or Arduino shield). In such condition USB enumeration will always succeed since no more than 100mA is requested to the PC. Possible configurations of JP1 are summarized in *Table 6*.

Table 6. JP1 configuration table

Jumper state	Power supply	Allowed current
JP1 jumper OFF	USB power through CN1	300 mA max
JP1 jumper ON		100 mA max

Warning: If the maximum current consumption of the NUCLEO and its extension boards exceeds 300 mA, it is mandatory to power the NUCLEO using an external power supply connected to E5V or VIN.

Note: In case the board is powered by an USB charger, there is no USB enumeration, so the led LD3 remains set to OFF permanently and the target STM32 is not powered. In this specific case the jumper JP1 needs to be set to ON, to allow target STM32 to be powered anyway.

6.3.2 External power supply inputs: VIN and E5V

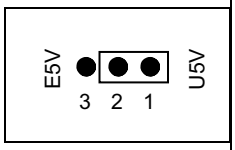
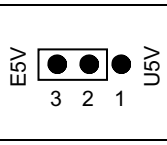
The external power sources VIN and E5V are summarized in the *Table 7.* When the board is power supplied by VIN or E5V, the jumpers configuration must be the following:

- Jumper on JP5 pin 2 and pin 3
- Jumper removed on JP1

Table 7. External power sources

Input power name	Connectors pins	Voltage range	Max current	Limitation
VIN	CN6 pin 8 CN7 pin 24	7 V to 12 V	800 mA	From 7 V to 12 V only and input current capability is linked to input voltage: 800 mA input current when Vin=7 V 450 mA input current when 7 V<Vin (< or =) 9 V 250 mA input current when 9 V<Vin (< or =) 12 V
E5V	CN7 pin 6	4.75 V to 5.25 V	500 mA	-

Table 8. Power-related jumper

Jumper	Description
JP5	U5V (ST-LINK VBUS) is used as power source when JP5 is set as shown below (Default setting) 
	VIN or E5V is used as power source when JP5 is set as shown below. 

Using VIN or E5V as external power supply

VIN or E5V can be used as external power supply in case the current consumption of the STM32 Nucleo and extensions boards exceeds the allowed current on USB. In this condition it is still possible to use the USB for communication, for programming or debugging only, but it is mandatory to power supply the board first using VIN or E5V then connect the USB cable to the PC. Proceeding this way ensures that the enumeration occurs thanks to the external power source.

The following power sequence procedure must be respected:

1. Connect the jumper between pin 2 and pin 3 of JP5
2. Check that JP1 is removed
3. Connect the external power source to VIN or E5V
4. Power on the external power supply 7 V< VIN < 12 V to VIN, or 5 V for E5V
5. Check that LD3 is turned ON
6. Connect the PC to USB connector CN1

If this order is not respected, the board may be supplied by VBUS first then by VIN or E5V, and the following risks may be encountered:

1. If more than 300 mA current is needed by the board, the PC may be damaged or the current supply can be limited by the PC. As a consequence the board is not powered correctly.
2. 300 mA is requested at enumeration (since JP1 must be OFF) so there is risk that the request is rejected and the enumeration does not succeed if the PC cannot provide such current. Consequently the board is not power supplied (LED LD3 remains OFF).

6.3.3 External power supply input: + 3.3V

It can be of interest to use the +3.3V (CN6 pin 4 or CN7 pin 12 and pin 16) directly as power input for instance in case the 3.3V is provided by an extension board. When the STM32 Nucleo board is power supplied by +3.3V, the ST-LINK is not powered thus the programming and debug features are unavailable. The external power sources +3.3V is summarized in the *Table 9.*

Table 9. +3.3V external power source

Input power name	Connectors pins	Voltage range	Limitation
+3.3V	CN6 pin 4 CN7 pin 12 and pin 16	3 V to 3.6 V	Used when ST-LINK part of PCB is cut or SB2 and SB12 OFF

Two different configurations are possible when using +3.3V to power the board:

- ST-LINK is removed (PCB cut) or
- SB2 (3.3V regulator) and SB12 (NRST) are OFF.

6.3.4 External power supply output

When powered by USB, VIN or E5V, the +5V (CN6 pin 5 or CN7 pin 18) can be used as output power supply for an Arduino shield or an extension board. In this case, the maximum current of the power source specified in *Table 7* must be respected.

The +3.3V (CN6 pin 4 or CN7 pin 12 and 16) can be used also as power supply output. The current is limited by the maximum current capability of the regulator U4 (500 mA max).

6.4 LEDs

The tricolor LED (green, orange, red) LD1 (COM) provides information about ST-LINK communication status. LD1 default color is red. LD1 turns to green to indicate that communication is in progress between the PC and the ST-LINK/V2-1, with the following setup:

- Slow blinking Red/Off: at power-on before USB initialization
- Fast blinking Red/Off: after the first correct communication between the PC and ST-LINK/V2-1 (enumeration)
- Red LED On: when the initialization between the PC and ST-LINK/V2-1 is complete
- Green LED On: after a successful target communication initialization
- Blinking Red/Green: during communication with target
- Green On: communication finished and successful
- Orange On: Communication failure

User LD2: the green LED is a user LED connected to Arduino signal D13 corresponding to STM32 I/O PA5 (pin 21) or PB13 (pin 34) depending on the STM32 target. Refer to *Table 11* to *Table 23* when:

- the I/O is HIGH value, the LED is on
- the I/O is LOW, the LED is off

LD3 PWR: the red LED indicates that the STM32 part is powered and +5V power is available.

6.5 Push-buttons

B1 USER: the user button is connected to the I/O PC13 (pin 2) of the STM32 microcontroller.

B2 RESET: this push-button is connected to NRST, and is used to RESET the STM32 microcontroller.

Note: The blue and black plastic hats that are placed on the push buttons can be removed if necessary, for example when a shield or when an application board is plugged on top of the Nucleo board. This will avoid pressure on the buttons and consequently a possible permanent target STM32 RESET.

6.6 JP6 (IDD)

Jumper JP6, labeled IDD, is used to measure the STM32 microcontroller consumption by removing the jumper and by connecting an ammeter:

- Jumper ON: STM32 microcontroller is powered (default).
- Jumper OFF: an ammeter must be connected to measure the STM32 microcontroller current. If there is no ammeter, STM32 microcontroller is not powered.

6.7 OSC clock

6.7.1 OSC clock supply

There are four ways to configure the pins corresponding to external high-speed clock (HSE):

- **MCO from ST-LINK**: MCO output of ST-LINK MCU is used as input clock. This frequency cannot be changed, it is fixed at 8 MHz and connected to PF0/PD0/PH0-OSC_IN of the STM32 microcontroller.

  The following configuration is needed:

  – SB55 OFF and SB54 ON
  – SB16 and SB50 ON
  – R35 and R37 removed
- **HSE oscillator on-board from X3 crystal (not provided):** for typical frequencies and its capacitors and resistors, refer to STM32 microcontroller datasheet. Refer to the AN2867 Application note for oscillator design guide for STM32 microcontrollers.The X3 crystal has the following characteristics: 8 MHz, 16 pF, 20 ppm, and DIP footprint. It is recommended to use 9SL8000016AFXHF0 manufactured by Hong Kong X'tals Limited.

  The following configuration is needed:

  – SB54 and SB55 OFF
  – R35 and R37 soldered
  – C33 and C34 soldered with 20 pF capacitors
  – SB16 and SB50 OFF
- **Oscillator from external PF0/PD0/PH0**: from an external oscillator through pin 29 of the CN7 connector.

  The following configuration is needed:

  – SB55 ON
  – SB50 OFF
  – R35 and R37 removed
- **HSE not used**: PF0/PD0/PH0 and PF1/PD1/PH1 are used as GPIO instead of clock

  The following configuration is needed:

  – SB54 and SB55 ON
  – SB16 and SB50 (MCO) OFF
  – R35 and R37 removed

There are two possible default configurations of the HSE pins, depending on the version of the STM32 Nucleo board hardware.

The board version MB1136 C-01 or MB1136 C-02 is mentioned on the sticker, placed on the bottom side of the PCB.

The board marking MB1136 C-01 corresponds to a board, configured as HSE not used.

The board marking MB1136 C-02 (or higher) corresponds to a board, configured to use ST-LINK MCO as clock input.

Note: For NUCLEO-L476RG and NUCLEO-L452RE the ST-LINK MCO output is not connected to OSCIN to reduce power consumption in low power mode. Consequently NUCLEO-L476RG and NUCLEO-L452RE configuration corresponds to HSE not used.

6.7.2 OSC 32 KHz clock supply

There are three ways to configure the pins corresponding to low-speed clock (LSE):

- **On-board oscillator:** X2 crystal. Refer to the *Oscillator design guide for STM8S, STM8A and STM32 microcontrollers* application note *(*AN2867) for oscillator design guide for STM32 microcontrollers. It is recommended to use ABS25-32.768KHZ-6-T, manufactured by Abracon corporation.
- **Oscillator from external PC14**: from external oscillator through the pin 25 of CN7 connector.

  The following configuration is needed:
  - SB48 and SB49 ON
  - R34 and R36 removed
- **LSE not used**: PC14 and PC15 are used as GPIOs instead of low speed clock.

  The following configuration is needed:
  - SB48 and SB49 ON
  - R34 and R36 removed

There are three possible default configurations of the LSE depending on the version of the STM32 Nucleo board hardware.

The board version MB1136 C-01 or MB1136 C-02 is mentioned on the sticker placed on the bottom side of the PCB.

The board marking MB1136 C-01 corresponds to a board configured as LSE not used.

The board marking MB1136 C-02 (or higher) corresponds to a board configured with on-board 32KHz oscillator.

The board marking MB1136 C-03 (or higher) corresponds to a board using new LSE crystal (ABS25) and C26, C31 and C32 value update.

6.8 USART communication

The USART2 interface available on PA2 and PA3 of the STM32 microcontroller can be connected to ST-LINK MCU, ST morpho connector or to Arduino connector. The choice can be changed by setting the related solder bridges. By default the USART2 communication between the target STM32 and ST-LINK MCU is enabled, in order to support virtual COM port for mbed (SB13 and SB14 ON, SB62 and SB63 OFF). If the communication between the target STM32 PA2 (D1) or PA3 (D0) and shield or extension board is required, SB62 and SB63 should be ON, SB13 and SB14 should be OFF. In such case it is possible to connect another USART to ST-LINK MCU using flying wires between ST morpho connector and CN3. For instance on NUCLEO-F103RB it is possible to use USART3 available on PC10 (TX) and PC11 (RX). Two flying wires need to be connected as follow:

- PC10 (USART3_TX) available on CN7 pin 1 to CN3 pin RX
- PC11 (USART3_RX) available on CN7 pin 2 to CN3 pin TX

6.9 Solder bridges

Table 10. Solder bridges

Bridge	State(1)	Description
SB54, SB55 (X3 crystal)(2)	OFF	X3, C33, C34, R35 and R37 provide a clock as shown in *Section Appendix A: Electrical schematics* PF0/PD0/PH0, PF1/PD1/PH1 are disconnected from CN7.
	ON	PF0/PD0/PH0, PF1/PD1/PH1 are connected to CN12. (R35, R37 and SB50 must not be fitted).
SB3,5,7,9 (DEFAULT)	**ON**	Reserved, do not modify.
SB4,6,8,10 (RESERVED)	OFF	Reserved, do not modify.
SB48,49 (X2 crystal)(3)	OFF	X2, C31, C32, R34 and R36 deliver a 32 kHz clock. PC14, PC15 are not connected to CN7.
	ON	PC14, PC15 are only connected to CN7. Remove only R34, R36.
SB17 (B1-USER)	**ON**	B1 push button is connected to PC13.
	OFF	B1 push button is not connected to PC13.
SB12 (NRST)	ON	The NRST signal of the CN4 connector is connected to the NRST pin of the STM32.
	OFF	The NRST signal of the CN4 connector is not connected to the NRST pin of the STM32.
SB15 (SWO)	ON	The SWO signal of the CN4 connector is connected to PB3.
	OFF	The SWO signal is not connected.
SB11 (STM_RST)	**OFF**	No incidence on STM32F103CBT6 (ST-LINK MCU) NRST signal.
	ON	STM32F103CBT6 (ST-LINK MCU) NRST signal is connected to GND.
SB1 (USB-5V)	OFF	USB power management is functional.
	ON	USB power management is disabled.
SB2 (3.3 V)	ON	Output of voltage regulator LD39050PU33R is connected to 3.3V.
	OFF	Output of voltage regulator LD39050PU33R is not connected.
SB21 (LD2-LED)	ON	Green user LED LD2 is connected to D13 of Arduino signal.
	OFF	Green user LED LD2 is not connected.
SB56,SB51 (A4 and A5)	ON	PC1 and PC0 (ADC in) are connected to A4 and A5 (pin 5 and pin 6) on Arduino connector CN8 and ST morpho connector CN7. Thus SB46 and SB52 should be OFF.
	OFF	PC1 and PC0 (ADC in) are disconnected to A4 and A5 (pin 5 and pin 6) on Arduino connector CN8 and ST morpho connector CN7.
SB46,SB52 (I2C on A4 and A5)	OFF	PB9 and PB8 (I2C) are disconnected to A4 and A5 (pin 5 and pin 6) on Arduino connector CN8 and ST morpho connector CN7.
	ON	PB9 and PB8 (I2C) are connected to A4 and A5 (pin 5 and pin 6) on Arduino connector CN8 and ST morpho connector CN7 as I2C signals. Thus SB56 and SB51 should be OFF.

Table 10. Solder bridges (continued)

Bridge	State[1]	Description
SB45 (VBAT/VLCD)	ON	VBAT or VLCD on STM32 is connected to VDD.
	OFF	VBAT or VLCD on STM32 is not connected to VDD.
SB57 (VDDA/VREF+)	ON	VDDA/VREF+ on STM32 is connected to VDD.
	OFF	VDDA/VREF+ on STM32 is not connected to VDD and can be provided from pin 8 of CN5 (Used for external VREF+ provided by Arduino shield)
SB62, SB63 (USART)	**OFF**	PA2 and PA3 on STM32 are disconnected to D1 and D0 (pin 2 and pin 1) on Arduino connector CN9 and ST morpho connector CN10.
	ON	PA2 and PA3 on STM32 are connected to D1 and D0 (pin 2 and pin 1) on Arduino connector CN9 and ST morpho connector CN10 as USART signals. Thus SB13 and SB14 should be OFF.
SB13, SB14 (ST-LINK-USART)	**ON**	PA2 and PA3 on STM32F103CBT6 (ST-LINK MCU) are connected to PA3 and PA2 on STM32 to have USART communication between them. Thus SB61,SB62 and SB63 should be OFF.
	OFF	PA2 and PA3 on STM32F103CBT6 (ST-LINK MCU) are disconnected to PA3 and PA2 on STM32.
SB16,SB50(MCO)[2]	OFF	MCO on STM32F103CBT6 (ST-LINK MCU) are disconnected to PF0/PD0/PH0 on STM32.
	ON	MCO on STM32F103CBT6 (ST-LINK MCU) are connected to PF0/PD0/PH0 on STM32.

1. The default SBx state is shown in bold.
2. Default configuration depends on board version. Refer to *Section 6.7.1: OSC clock supply* for details.
3. Default configuration depends on board version. Refer to *Section 6.7.2: OSC 32 KHz clock supply* for details.

All the other solder bridges present on the STM32 Nucleo board are used to configure several I/Os and power supply pins for compatibility of features and pinout with STM32 supported.

All STM32 Nucleo boards are delivered with the solder-bridges configured according to the target supported STM32.

6.10 Extension connectors

Figure 10 to *Figure 26* show the signals connected by default to Arduino Uno V3 connectors (CN5, CN6, CN8, CN9) and to ST morpho connector (CN7 and CN10), for each STM32 Nucleo board.

Figure 24. NUCLEO-L476RG

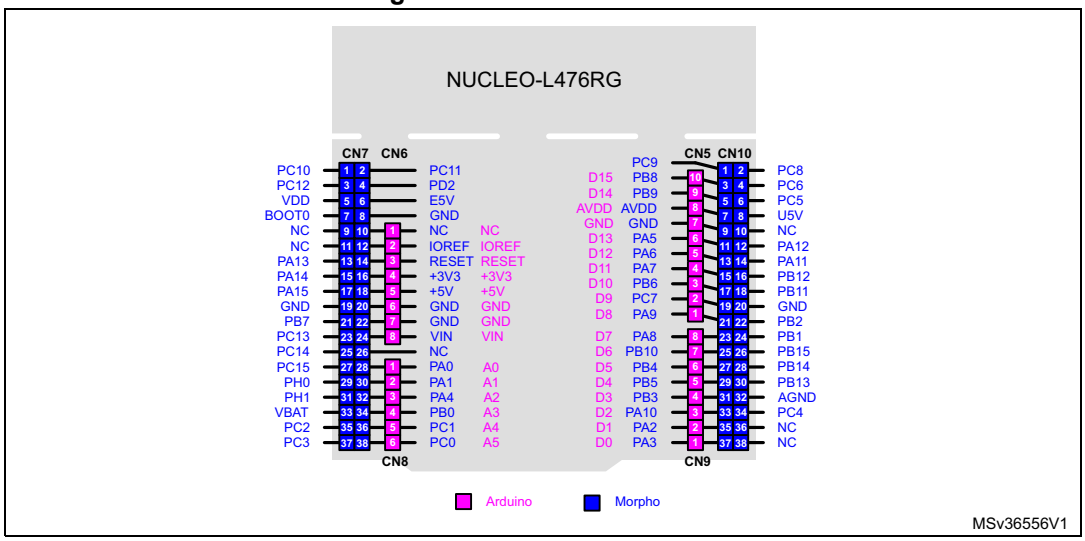
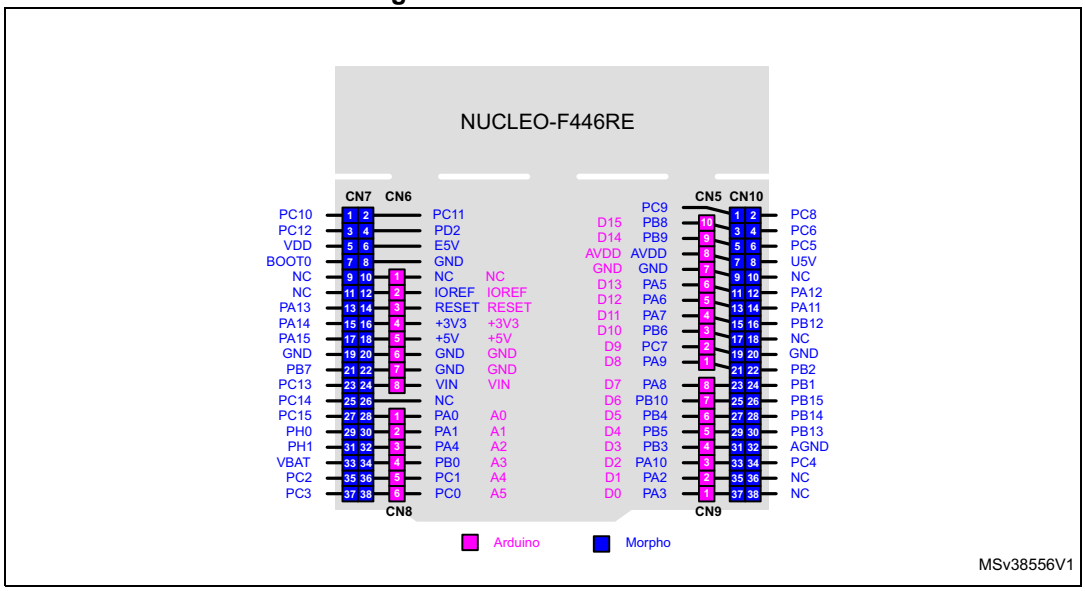


Figure 25. NUCLEO-F446RE



6.11 Arduino connectors

CN5, CN6, CN8 and CN9 are female connectors compatible with Arduino standard. Most shields designed for Arduino can fit to the STM32 Nucleo boards.

The Arduino connectors on STM32 Nucleo board support the Arduino Uno V3.

For compatibility with Arduino Uno V1, apply the following modifications:

- SB46 and SB52 should be ON,
- SB51 and SB56 should be OFF to connect I^2C on A4 (pin 5) and A5 (pin 6 of CN8).

Caution 1: The I/Os of STM32 microcontroller are 3.3 V compatible instead of 5 V for Arduino Uno V3.

Caution 2: SB57 should be removed before implementing Arduino shield with VREF+ power being provided on CN5 pin 8. Refer to *Table 10: Solder bridges* for details on SB57.

Table 11 to *Table 23* show the pin assignment of each main STM32 microcontroller on Arduino connectors.

Table 11. Arduino connectors on NUCLEO-F030R8, NUCLEO-F070RB, NUCLEO-F072RB, NUCLEO-F091RC

Connector	Pin	Pin name	STM32 pin	Function
Left connectors				
CN6 power	1	NC	-	-
	2	IOREF	-	3.3V Ref
	3	RESET	NRST	RESET
	4	+3.3V	-	3.3V input/output
	5	+5V	-	5V output
	6	GND	-	ground
	7	GND	-	ground
	8	VIN	-	Power input
CN8 analog	1	A0	PA0	ADC_IN0
	2	A1	PA1	ADC_IN1
	3	A2	PA4	ADC_IN4
	4	A3	PB0	ADC_IN8
	5	A4	PC1 or PB9(1)	ADC_IN11 (PC1) or I2C1_SDA (PB9)
	6	A5	PC0 or PB8(1)	ADC_IN10 (PC0) or I2C1_SCL (PB8)
Right connectors				
CN5 digital	10	D15	PB8	I2C1_SCL
	10	D15	PB8	I2C1_SCL
	10	D15	PB8	I2C1_SCL
	9	D14	PB9	I2C1_SDA
	8	AREF	-	AVDD
	7	GND	-	ground

Table 22. Arduino connectors on NUCLEO-L452RE (continued)

Connector	Pin	Pin name	STM32 pin	Function
CN9 digital	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

1. Refer to *Table 10: Solder bridges* for details.

Table 23. Arduino connectors on NUCLEO-L476RG

Connector	Pin	Pin name	STM32 pin	Function
Left connectors				
CN6 power	1	NC	-	-
	2	IOREF	-	3.3V Ref
	3	RESET	NRST	RESET
	4	+3.3V	-	3.3V input/output
	5	+5V	-	5V 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 or PB9(1)	ADC123_IN2 (PC1) or I2C1_SDA (PB9)
	6	A5	PC0 or PB8(1)	ADC123_IN1 (PC0) or I2C1_SCL (PB8)
Right connectors				
CN5 digital	10	D15	PB8	I2C1_SCL
	9	D14	PB9	I2C1_SDA
	8	AREF	-	AVDD
	7	GND	-	ground
	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

Table 23. Arduino connectors on NUCLEO-L476RG (continued)

Connector	Pin	Pin name	STM32 pin	Function
CN5 digital	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

1. Refer to *Table 10: Solder bridges* for details.

6.12 ST morpho connector

The ST morpho connector consists in male pin headers (CN7 and CN10) accessible on both sides of the board. They can be used to connect the STM32 Nucleo board to an extension board or a prototype/wrapping board placed on top or on bottom side of the STM32 Nucleo board. All signals and power pins of the STM32 are available on ST morpho connector. This connector can also be probed by an oscilloscope, logical analyzer or voltmeter.

Table 24 to *Table 33* show the pin assignments of each STM32 on ST morpho connector.

Table 24. ST morpho connector on NUCLEO-F030R8

CN7 odd pins		CN7 even pins		CN10 odd pins		CN10 even pins	
Pin	Name	Name	Pin	Pin	Name	Name	Pin
1	PC10	PC11	2	1	PC9	PC8	2
3	PC12	PD2	4	3	PB8	PC6	4
5	VDD	E5V	6	5	PB9	PC5	6
7	BOOT0(1)	GND	8	7	AVDD	U5V(2)	8
9	PF6	-	10	9	GND	-	10
11	PF7	IOREF	12	11	PA5	PA12	12
13	PA13	RESET	14	13	PA6	PA11	14
15	PA14	+3.3V	16	15	PA7	PB12	16
17	PA15	+5V	18	17	PB6	PB11	18
19	GND	GND	20	19	PC7	GND	20
21	PB7	GND	22	21	PA9	PB2	22
23	PC13(3)	VIN	24	23	PA8	PB1	24
25	PC14(3)	-	26	25	PB10	PB15	26

Table 32. ST morpho connector on NUCLEO-L476RG

CN7 odd pins		CN7 even pins		CN10 odd pins		CN10 even pins	
Pin	**Name**	**Name**	**Pin**	**Pin**	**Name**	**Name**	**Pin**
1	PC10	PC11	2	1	PC9	PC8	2
3	PC12	PD2	4	3	PB8	PC6	4
5	VDD	E5V	6	5	PB9	PC5	6
7	BOOT0(1)	GND	8	7	AVDD	U5V(2)	8
9	-	-	10	9	GND	-	10
11	-	IOREF	12	11	PA5	PA12	12
13	PA13(3)	RESET	14	13	PA6	PA11	14
15	PA14(3)	+3.3V	16	15	PA7	PB12	16
17	PA15	+5V	18	17	PB6	PB11	18
19	GND	GND	20	19	PC7	GND	20
21	PB7	GND	22	21	PA9	PB2	22
23	PC13	VIN	24	23	PA8	PB1	24
25	PC14	-	26	25	PB10	PB15	26
27	PC15	PA0	28	27	PB4	PB14	28
29	PH0	PA1	30	29	PB5	PB13	30
31	PH1	PA4	32	31	PB3	AGND	32
33	VBAT	PB0	34	33	PA10	PC4	34
35	PC2	PC1 or PB9(4)	36	35	PA2	-	36
37	PC3	PC0 or PB8(4)	38	37	PA3	-	38

1. Default state of BOOT0 is 0. It can be set to 1 when a jumper is on pin5-7 of CN7.
2. U5V is 5 V power from ST-LINK/V2-1 USB connector and it rises before +5V.
3. PA13 and PA14 share with SWD signals connected to ST-LINK/V2-1, it is not recommend to use them as IO pins if ST-LINK part is not cut.
4. Refer to *Table 10: Solder bridges* for details.

Appendix A Electrical schematics

Figure 27 to *Figure 30* show the electrical schematics of the STM32 Nucleo-64 board.

Figure 27. Top and Power

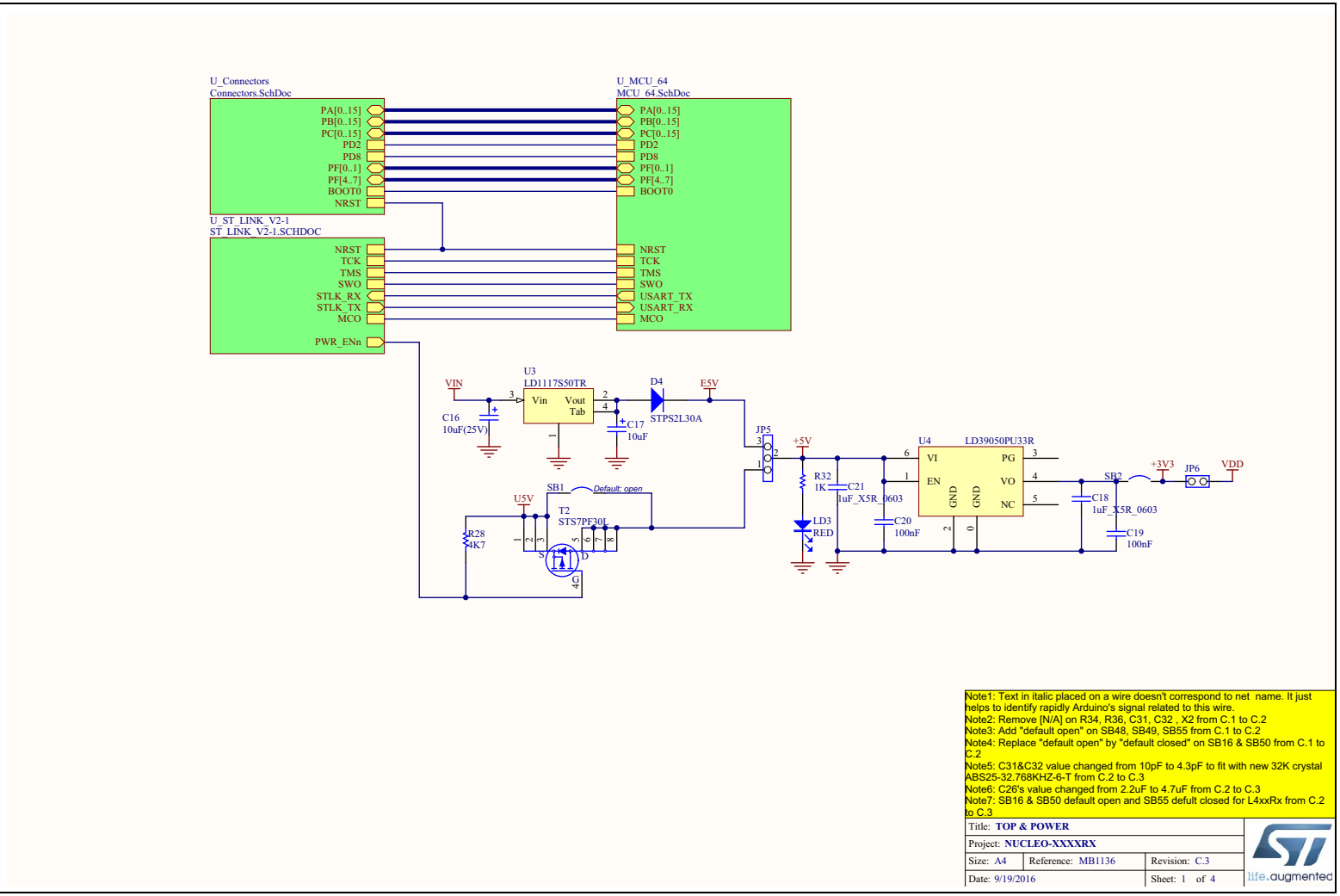


Figure 28. STM32 MCU

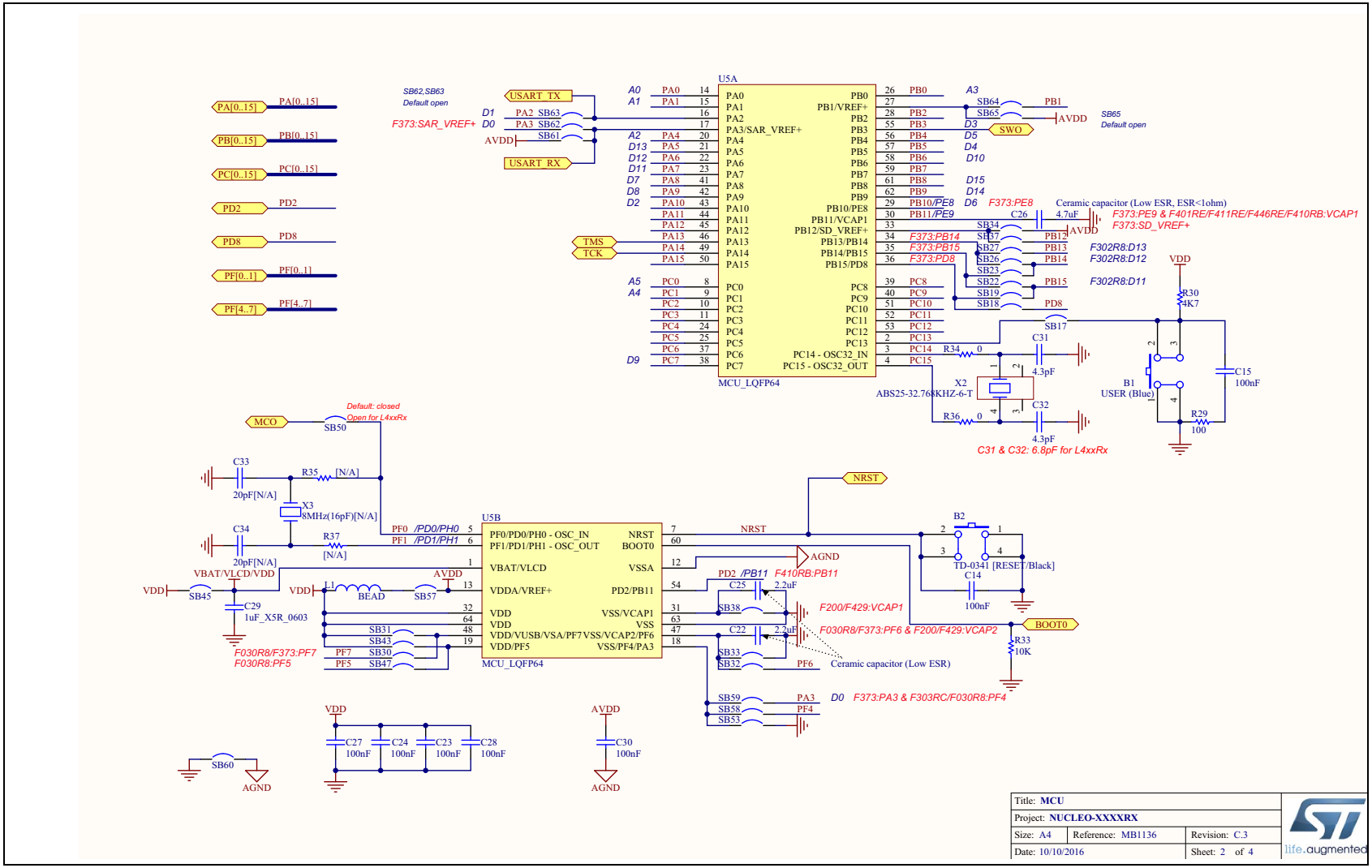


Figure 29. ST-LINK/V2-1

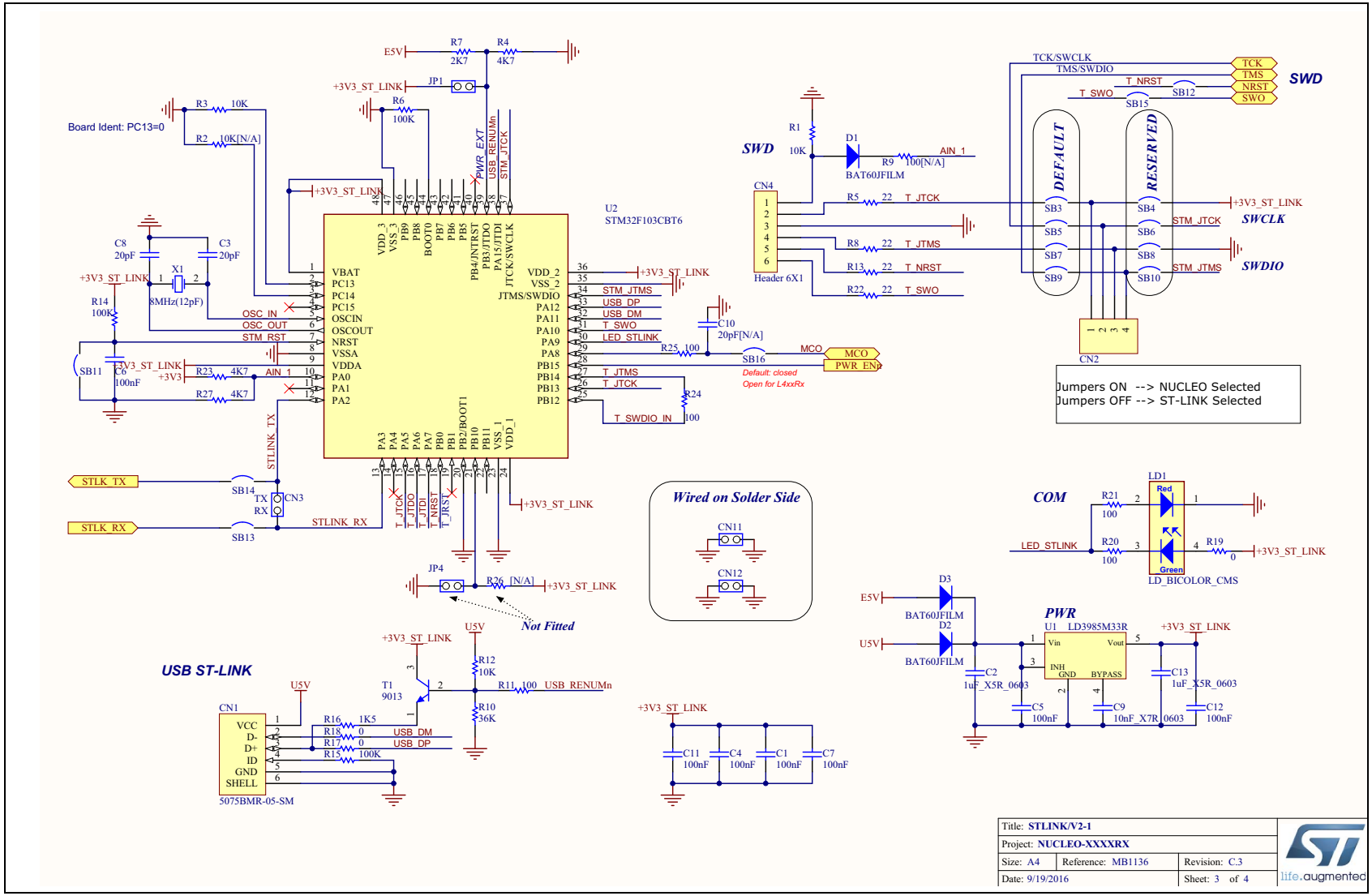
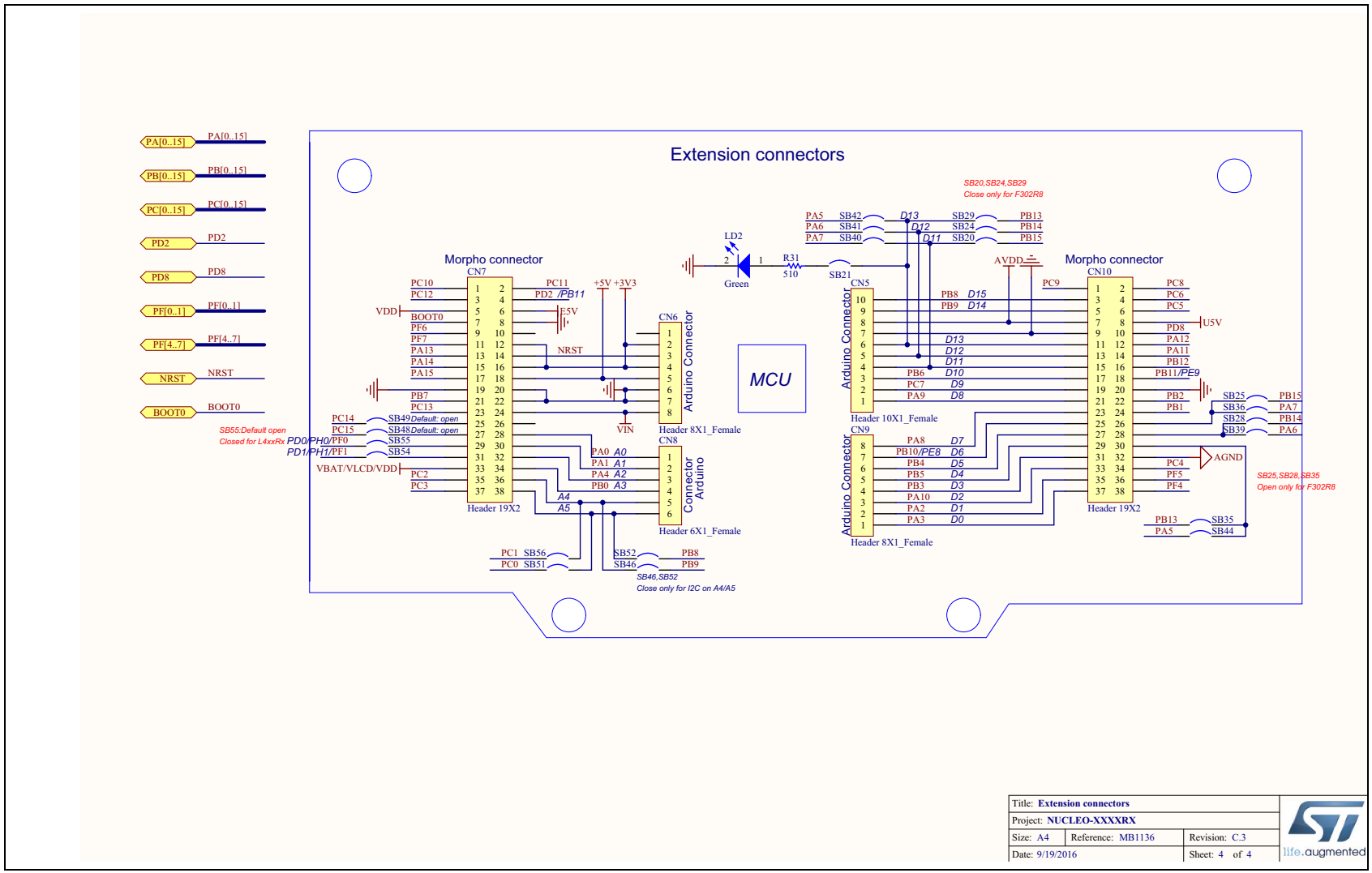


Figure 30. Extension connectors



Revision history

Table 34. Document revision history

Date	Revision	Changes
10-Feb-2014	1	Initial release.
13-Feb-2014	2	Updated *Figure 1*, *Chapter 5.5* and *Table 10*.
11-Apr-2014	3	Extended the applicability to NUCLEO-F302R8. Updated *Table 1: Ordering information*, *Section 6.11: Arduino connectors* and *Section 6.12: ST morpho connector*. Updated *Figure 1*
10-June-2014	4	Updated the board figure: *Figure 1*. Updated HSE and LSE configuration description: *Section 6.7.1*, *Section 5.5* and *Section 6.7.2*. Extended the applicability to NUCLEO-F334R8, NUCLEO-F411RE and NUCLEO-L053R8.
20-June-2014	5	Updated the electrical schematics figures: *Figure 27*, *Figure 28*, *Figure 29* and *Figure 30*. Refer to the AN2867 for oscillator design guide for STM32 microcontrollers in *Section 6.7.1: OSC clock supply* and *Section 6.7.2: OSC 32 KHz clock supply*.
30-Sept-2014	6	Extended the applicability to NUCLEO-F091RC and NUCLEO-F303RE; Updated *Table 1: Ordering information*; Updated *Table 11: Arduino connectors on NUCLEO-F030R8, NUCLEO-F070RB, NUCLEO-F072RB, NUCLEO-F091RC*; Updated *Table 26: ST morpho connector on NUCLEO-F072RB, NUCLEO-F091RC, NUCLEO-F303RE, NUCLEO-F334R8*; Updated *Figure 6: Typical configuration*; Added *Figure 13: NUCLEO-F091RC*; Added *Figure 16: NUCLEO-F303RE*; Updated *Section 6.7.2: OSC 32 KHz clock supply*; Updated *Figure 27: Top and Power(1/4)* ,*Figure 28: STM32 MCU*;

Table 34. Document revision history (continued)

Date	Revision	Changes
19-Jan-2015	7	Extended the applicability to NUCLEO-F070RB, NUCLEO-L073RZ and NUCLEO-L476RG; Updated *Table 1: Ordering information*; Updated *Section 6.2: Embedded ST-LINK/V2-1*; Updated *Section 6.7.1: OSC clock supply*; Added *Figure 11: NUCLEO-F070RB*; Added *Figure 21: NUCLEO-L073RZ*; Added *Figure 24: NUCLEO-L476RG* Updated *Table 11: Arduino connectors on NUCLEO-F030R8, NUCLEO-F070RB, NUCLEO-F072RB, NUCLEO-F091RC* Added *Table 18: Arduino connectors on NUCLEO-L073RZ* Added *Table 23: Arduino connectors on NUCLEO-L476RG* Added *Table 25: ST morpho connector on NUCLEO-F070RB* Updated *Table 30: ST morpho connector on NUCLEO-L053R8, NUCLEO-L073RZ, NUCLEO-L152RE* Added *Table 32: ST morpho connector on NUCLEO-L476RG* Updated schematics from *Figure 27: Top and Power(1/4)* to *Figure 30: Extension connectors*
08-Jul-2015	8	Extended the applicability to Updated *Table 1: Ordering information*; Added *Figure 25: NUCLEO-F446RE* and *Figure 26: NUCLEO-F410RB* Updated *Section 6.11: Arduino connectors on page 37* and *Section 6.12: ST morpho connector on page 53*
04-Aug-2015	9	Added *Section 5.4: NUCLEO-L476RG bootloader limitations*.
17-Nov-2015	10	Updated *Section 6.9: Solder bridges* and *Section 6.7.1: OSC clock supply*.
29-Nov-2015	11	Updated *Introduction*, *Section 3: Ordering information*, *Section 6.10: Extension connectors*, *Section 6.11: Arduino connectors*, *Section 6.12: ST morpho connector* to add NUCLEO-L452RE.