

Introduction

STMCube™ initiative was originated by STMicroelectronics to ease developers' life by reducing development efforts, time and cost. STM32Cube covers STM32 portfolio. STM32Cube Version Y.x includes:

- The STM32CubeMX, a graphical software configuration tool that allows to generate C initialization code using graphical wizards
- A comprehensive embedded software platform, delivered per series (such as STM32CubeL0 for STM32L0 Series):
 - The STM32CubeL0 HAL, an STM32 abstraction layer embedded software, ensuring maximized portability across STM32 portfolio.
 - A consistent set of middleware components such as RTOS, USB, STM Touch and FatFs.
 - All embedded software utilities coming with a full set of examples. The STM32CubeL0 Nucleo-32 demonstration platform is built around the STM32Cube HAL BSP and uses HAL I²C to communicate with a third-party-display shield.

This user manual describes how to get started with the STM32CubeL0 demonstration firmware.

Contents

1	STM32CubeL0 main features	5
2	Getting started with the demonstration	6
2.1	Hardware requirements	6
2.1.1	STM32L031 or STM32L011 Nucleo-32 board	6
2.1.2	Gravitech 7-segment display shield	7
2.2	Hardware configuration	7
2.2.1	STM32 Nucleo-32 board configuration	7
2.2.2	Stacking the Gravitech shield	7
3	Demonstration firmware package	9
3.1	Demonstration repository	9
3.2	Nucleo-32 board BSP	10
4	Demonstration functional description	11
4.1	Scheduled power-mode sequence	12
4.2	Power measurement	12
5	Programming firmware application	13
6	Revision history	14

List of tables

Table 1.	Pin description	11
Table 2.	Document revision history	14

List of figures

Figure 1.	STM32Cube block diagram	5
Figure 2.	STM32 Nucleo-32 board	6
Figure 3.	Gravitech 7-segment shield	7
Figure 4.	Gravitech 7-segment shield plugged in	8
Figure 5.	Folder structure overview	9
Figure 6.	STM32 Nucleo-32 board BSP structure	10
Figure 7.	Demonstration running overview	11
Figure 8.	Low-power-mode sequence	12

1 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 the 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.

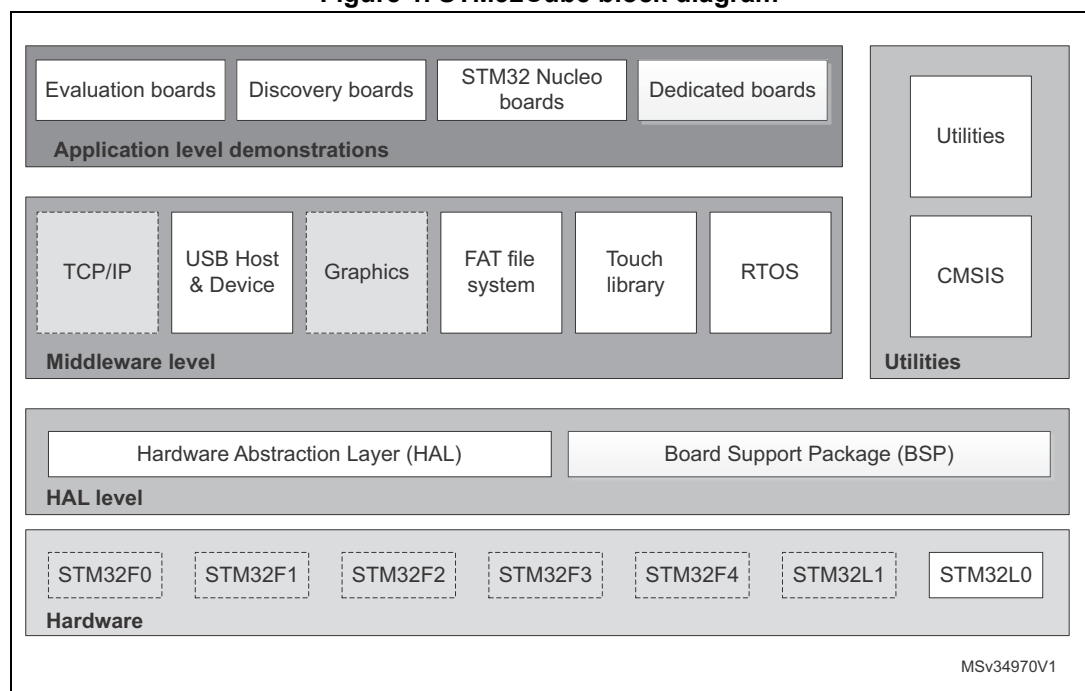
For user convenience the HAL is available in an open-source BSD license.

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, 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



2 Getting started with the demonstration

2.1 Hardware requirements

The hardware requirements to start the demonstration application are as follow:

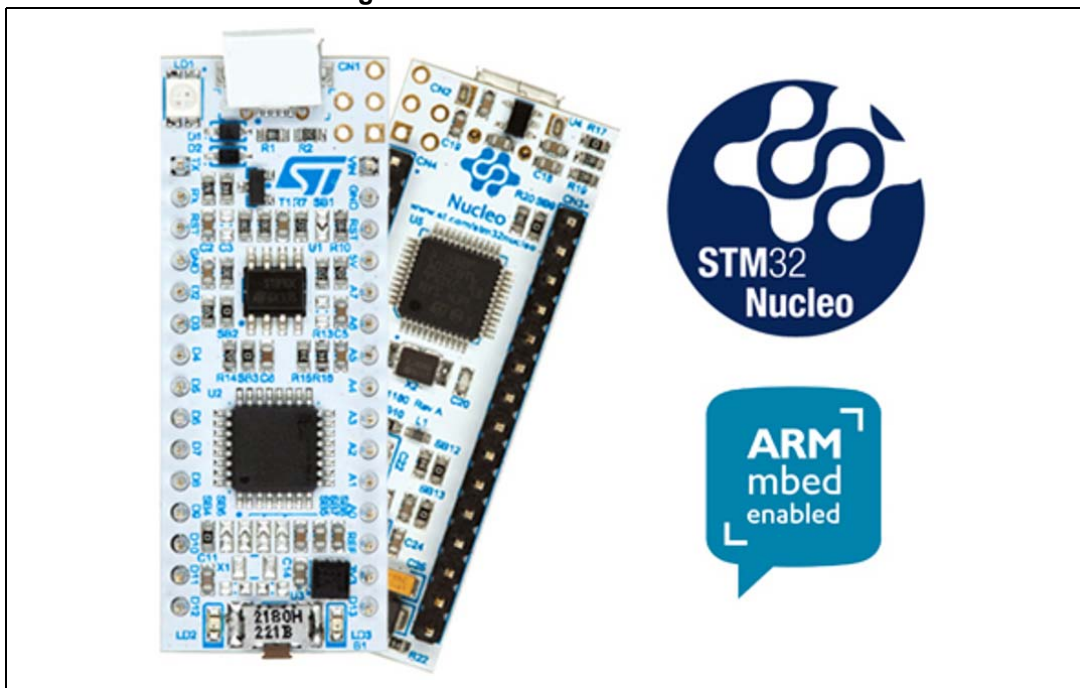
- STM32L031 or STM32L011 Nucleo-32 board
- Gravitech 7-segment display for Arduino™ Nano (from www.gravitech.us)
- One USB Type A to Mini-B cable to power up the Nucleo-32 board from the USB ST-LINK (USB connector CN1)

2.1.1 STM32L031 or STM32L011 Nucleo-32 board

The STM32 Nucleo-32 board (Arduino Nano compatible with 32-pin MCU) is a low-cost and easy-to-use development kit to quickly evaluate and start some development with ARM® 32-bits Cortex®-M microcontrollers of the STM32 Series (STM32L031, STM32L011). Before installing and using the product, accept the Evaluation Product Licence Agreement available at www.st.com/epla.

For more information on the Nucleo-32 boards visit www.st.com/stm32nucleo and select the STM32 Nucleo-32 board inside the product table list.

Figure 2. STM32 Nucleo-32 board



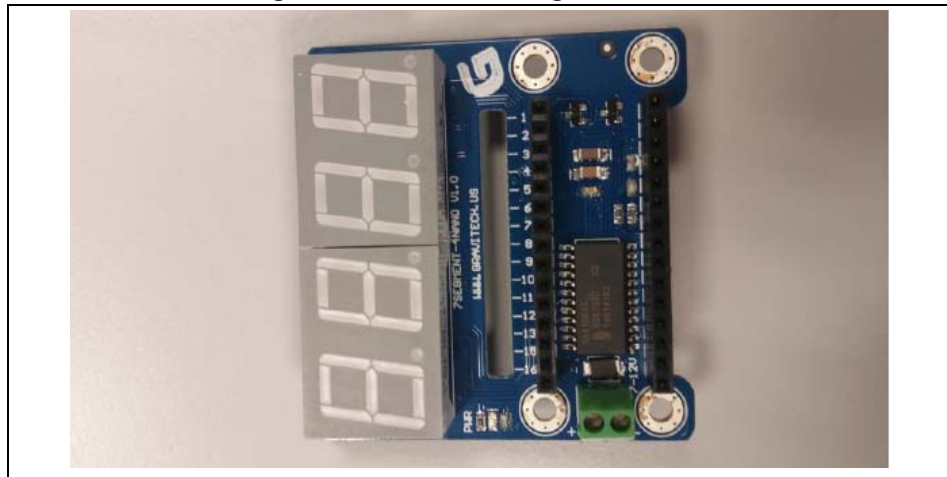
2.1.2 Gravitech 7-segment display shield

The STM32 Nucleo-32 board supports Arduino Nano connectivity. This Gravitech 7-segment display shield for Arduino Nano may be ordered on the www.gravitech.us website.

This shield contains the following features:

- 4-digit, 7-segment display
- Long-header pins for stackability
- Dimensions: 2.25" * 2.18" * 0.48"

Figure 3. Gravitech 7-segment shield



2.2 Hardware configuration

2.2.1 STM32 Nucleo-32 board configuration

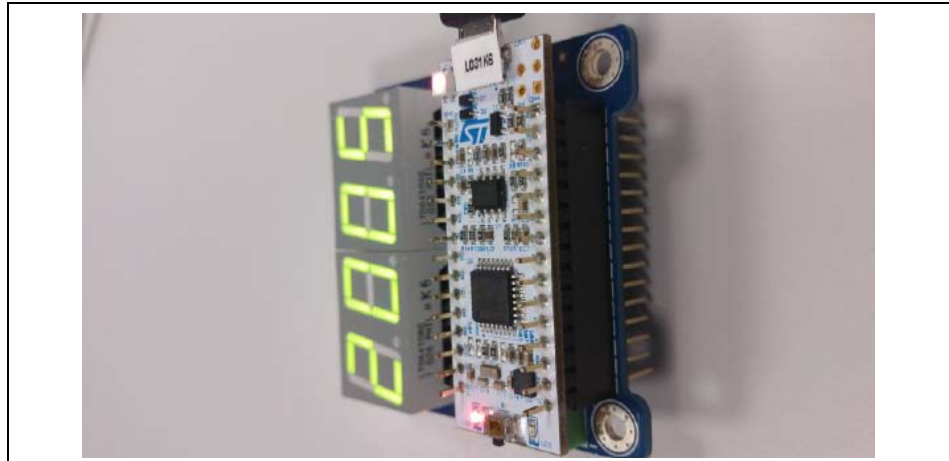
After gathering the hardware, the default hardware solder-bridge configuration of the STM32 Nucleo-32 board meets the requirements.

Afterward the user must plug the STM32 Nucleo-32 board on top of the Gravitech STM32 Nucleo-32 7-segment shield for Arduino Nano.

2.2.2 Stacking the Gravitech shield

The Gravitech 7-segment Arduino Nano shield comes with the surface mounted parts pre-soldered. User has just to plug the STM32 Nucleo-32 board on top of the 7-segment Arduino Nano shield, placing the STM32 Nucleo-32 board over the header strips. The requested alignment is shown in [Figure 4](#).

Figure 4. Gravitech 7-segment shield plugged in

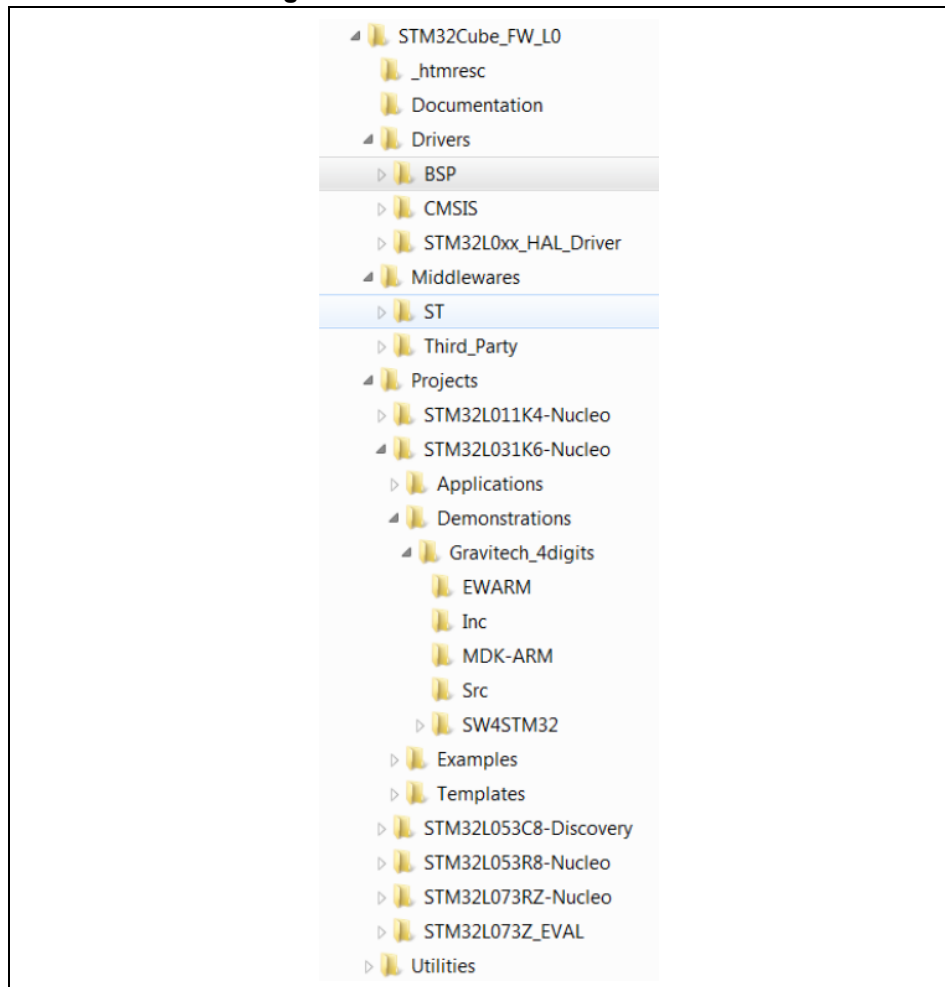


3 Demonstration firmware package

3.1 Demonstration repository

The demonstration with the STM32 Nucleo-32 board on top of the Gravitech 7-segment shield for Arduino Nano is provided with the STM32CubeL0 firmware package. The folder structure for this package is shown in [Figure 5](#).

Figure 5. Folder structure overview



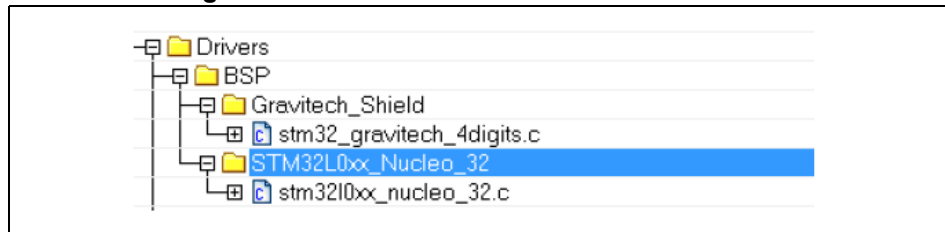
The demonstration source files are located under the STM32Cube firmware package for the STM32L031K6 Nucleo-32 boards. This demonstration package is composed of three groups as described below:

- Src: contains the demonstration source files.
- Inc: contains the demonstration header files.
- Project settings: a folder per toolchain (as EWARM for IAR™, MDK-ARM™ for Keil® and SW4STM32 for AC6) containing the project settings, the startup file and the scatter file for the linker.

3.2 Nucleo-32 board BSP

For the STM32 Nucleo-32 board, a set of buttons, LED, I2C, SPI, SD_IO, LCD_IO, ADC and joystick drivers are available within the stm3210_nucleo_32.c/h files implementing the board capabilities and the bus link mechanism (see [Figure 6](#)).

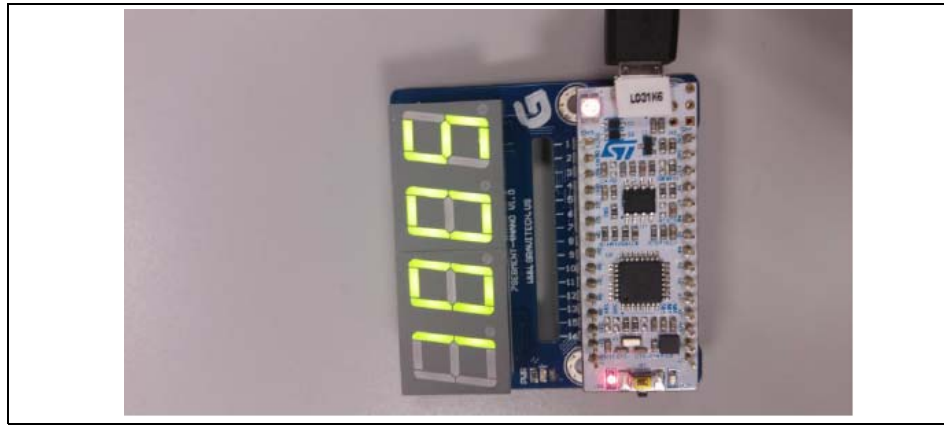
Figure 6. STM32 Nucleo-32 board BSP structure



4 Demonstration functional description

In this demonstration application, it is shown how to use the STM32CubeL0 firmware package with the STM32 Nucleo-32 board and the Gravitech 7-segment Arduino Nano shield to display on four digits the different low-power modes (Run, Sleep, Stop, Standby, LP Run, and LP Sleep) supported by the STM32L03x or STM32L01x. The demonstration changes the STM32 power mode every ten seconds, displaying the time elapsed on fourth digit, counting from 1 to 9 and displaying current low-power mode, thanks to the second counter, on the first digit incremented from 1 to 6 (see [Figure 7](#)). For details about the low-power modes, refer to the STM32L0xx reference manual (RM0367) at the www.st.com website.

Figure 7. Demonstration running overview



This demonstration needs a Gravitech 4-digit, 7-segment Arduino Nano shield that the user can get from the website: <http://www.gravitech.us>. On top of this shield, the user must plug the STM32 Nucleo-32 board connected on the Arduino Nano connectors.

The Gravitech display shield uses the I²C bus to communicate with the STM32 Nucleo-32 board using internal predefined I²C slave address. Only four Arduino Nano pins are needed to run the demonstration with the connected ST-LINK cable (see [Table 1](#)):

Table 1. Pin description

STM32 CN4 pin number	Arduino Nano name	Type	Description
4	5V	PWR	Supply voltage
2	GND	PWR	Supply ground
8	A4 (SDA)	I/O	I ² C serial data
7	A5 (SCL)	OUTPUT	I ² C serial clock

For shield power supply, the user can use either the ST-LINK cable connected between the PC and the STM32 Nucleo-32 board (as already used for the flash loading) which supplies 5V to display through related Arduino Nano pin or he can disconnect the ST-LINK and use an external power supply (7-12V), connected to a dedicated electrical connector on the Gravitech 7-segment Arduino Nano board, which supplies STM32 Nucleo-32 board through the V_{in} pin of the Arduino Nano board.

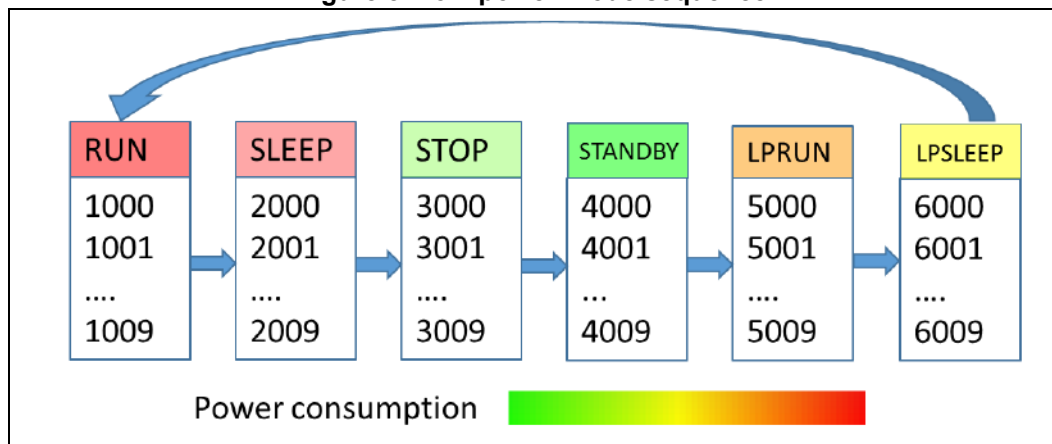
4.1 Scheduled power-mode sequence

The fourth digit corresponds to the time counter and the first digit corresponds to the power mode sequentially changed as described below:

- 100X = RUN Mode
- 200X = SLEEP Mode
- 300X = STOP Mode
- 400X = STANDBY Mode
- 500X = LPRUN Mode
- 600X = LPSLEEP Mode

The user can find the power modes sequence to discover the demonstration and the power consumption classification mapped with color scale as shown in *Figure 8*.

Figure 8. Low-power-mode sequence



4.2 Power measurement

The STM32 Nucleo-32 board has a tagged jumper Idd to perform the current consumption measurement using an ammeter. Note that power measurements done on default demonstration setup would also include the Gravitech 4-digit, 7-segment power consumption, which is not negligible compared to the MCU consumption.

For more details about the current consumption characteristics in different low-power modes, refer to STM32L0xx Datasheet at the www.st.com website. Note that the STM32CubeL0 firmware package also includes some examples for different low-power modes, which can be used as a reference.

Refer to the examples under \Projects\STM32L031K6-Nucleo\Examples\PWR to get specific projects insuring the current measurement performance (one project for each low-power mode).

5 Programming firmware application

To program the STM32 Nucleo-32 board with the demonstration application proceed as follows:

- Install the preferred Integrated Development Environment (IDE)
- Install the ST-LINK driver available at the www.st.com website

There are two methods of programming the STM32 Nucleo-32 board:

Method 1: Upload the STM32CubeL0_Demo_Nucleo32.hex from the firmware package available under:

Projects\STM32L031K6-Nucleo\Demonstrations\Gravitech_4digits\ Binary using the preferred in-system programming tool.

Method 2: Choose one of the three supported toolchains (IAR/ Keil/AC6) and follow the steps below:

- Open the application folder:
Projects\STM32L031K6-Nucleo\Demonstrations\Gravitech_4digits
- Choose the desired IDE project (EWARM for IAR, MDK-ARM for Keil and SW4STM32 for AC6)
- 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 firmware as well as other firmware examples, that allow user to discover the STM32 microcontroller features, are available at the www.st.com/stm32nucleo website.

6 Revision history

Table 2. Document revision history

Date	Revision	Changes
07-Apr-2016	1	Initial version.

IMPORTANT NOTICE – PLEASE READ CAREFULLY

STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST's terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers' products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2016 STMicroelectronics – All rights reserved

