



STM3210B-EVAL demonstration software

Introduction

This user manual describes the demonstration firmware running on the STM3210B-EVAL evaluation board, which you can use to evaluate the capabilities of the microcontroller and the on-board peripherals.

The STM3210B-EVAL board is delivered with the demonstration and the DFU (device firmware upgrade) programmed in the internal Flash memory, and all the files needed by the demonstration are programmed in the SPI Flash. At each reset (board power-up, external reset, etc.) the demonstration is executed. To run the DFU, the Key push-button must be kept pressed at reset.

In case the STM3210B-EVAL board was not factory-programmed, the demonstration application was erased or the SPI Flash contents were modified, the DFU can be used to program those files. For more details, refer to [Section 3](#) and [Section 4](#).

This demonstration firmware, the USB library, and other such firmware are available for download from the STMicroelectronics website: www.st.com.

Contents

1	Functional description	7
1.1	Power control	8
1.2	Clocking	8
1.3	Reset control	8
1.4	Debug JTAG interface	8
1.5	Serial wire debugger interface	8
1.6	Display devices	8
1.6.1	LCD	8
1.6.2	LED	8
1.7	Interfaces	8
1.7.1	RS232	8
1.7.2	CAN	9
1.7.3	USB	9
1.8	Motor control	9
1.9	IrDA	9
1.10	Miscellaneous peripherals	9
1.10.1	Joystick	9
1.10.2	Push-buttons	9
1.10.3	12-bit analog-to-digital converter (ADC)	9
1.10.4	Audio	9
1.10.5	Storage memories	10
1.10.6	Temperature sensor	10
2	Running the demonstration	11
2.1	Menu	11
2.1.1	Demonstration startup	12
2.1.2	Navigation	16
2.2	Clock sources	17
2.2.1	Clock control	17
2.2.2	Clock failure	17
2.3	STM32F10xxx resources	19
2.3.1	Peripherals	19
2.3.2	Interrupts	20

2.3.3	External interrupts	21
2.3.4	Internal memory size	21
2.3.5	External memory size	21
2.4	Demo applications	22
2.4.1	Product presentation	22
2.4.2	Calendar	24
2.4.3	Low-power modes	29
2.4.4	Thermometer	36
2.4.5	USB mass storage	37
2.4.6	Funny submenu	39
2.4.7	Help submenu	40
2.4.8	About submenu	42
3	Firmware package description	43
3.1	Demo folder	43
3.2	FWLib folder	43
3.3	images folder	44
3.4	USBLib folder	44
3.5	Demonstration firmware architecture	44
4	Demonstration programming using the DFU	46
4.1	Installing the DFU PC software	46
4.2	Programming the M25P64 Flash memory	47
4.3	Demonstration programming	50
5	Revision history	54

List of tables

Table 1.	STM32F10xxx evaluation board peripherals	19
Table 2.	STM32F10xxx demo interrupts.	20
Table 3.	STM32F10xxx demo external interrupts.	21
Table 4.	Document revision history	54

List of figures

Figure 1.	Evaluation board overview	7
Figure 2.	Structure of the demonstration menus	12
Figure 3.	Warning message	13
Figure 4.	ST Logo	13
Figure 5.	STM32 family	13
Figure 6.	Time and date configuration	13
Figure 7.	Main menu	14
Figure 8.	Corresponding submenus.	15
Figure 9.	Navigating in the demonstration menus	16
Figure 10.	No HSE clock detected.	17
Figure 11.	Standby mode entered	18
Figure 12.	Internal Flash memory organization	21
Figure 13.	SPI Flash memory organization	21
Figure 14.	Product presentation is ready to start	22
Figure 15.	First presentation slide	22
Figure 16.	Last presentation slide.	23
Figure 17.	No loaded wave file	23
Figure 18.	End of slide show	24
Figure 19.	Setting the time and date	24
Figure 20.	Time Adjust submenu.	25
Figure 21.	Time Show submenu	25
Figure 22.	Setting the Year	26
Figure 23.	Setting the Month	27
Figure 24.	Setting the day of the month.	27
Figure 25.	Exiting the Date Show submenu.	27
Figure 26.	Setting the alarm activation time.	28
Figure 27.	The time and date need setting	28
Figure 28.	Alarm Show submenu.	29
Figure 29.	Exiting the Stop mode.	29
Figure 30.	Stop mode entered	30
Figure 31.	MCU in the Stop mode	30
Figure 32.	RTC Alarm causes the MCU to exit the Stop mode	31
Figure 33.	Setting the Wakeup time.	31
Figure 34.	RTC Alarm wakeup configured.	32
Figure 35.	RTC Alarm wakeup.	32
Figure 36.	Time and Date configuration prompt	32
Figure 37.	Entering the Standby mode	33
Figure 38.	MCU in Standby mode	33
Figure 39.	RTC Alarm causes the MCU to exit the Standby mode	34
Figure 40.	Setting the wakeup time	34
Figure 41.	RTC Alarm wakeup configured.	35
Figure 42.	Time and Date configuration prompt	35
Figure 43.	Thermometer submenu selected	36
Figure 44.	Temperature display	36
Figure 45.	Temperature sensor error.	37
Figure 46.	USB Mass Storage submenu	37
Figure 47.	USB Mass Storage submenu selected.	38
Figure 48.	USB cable connected	38

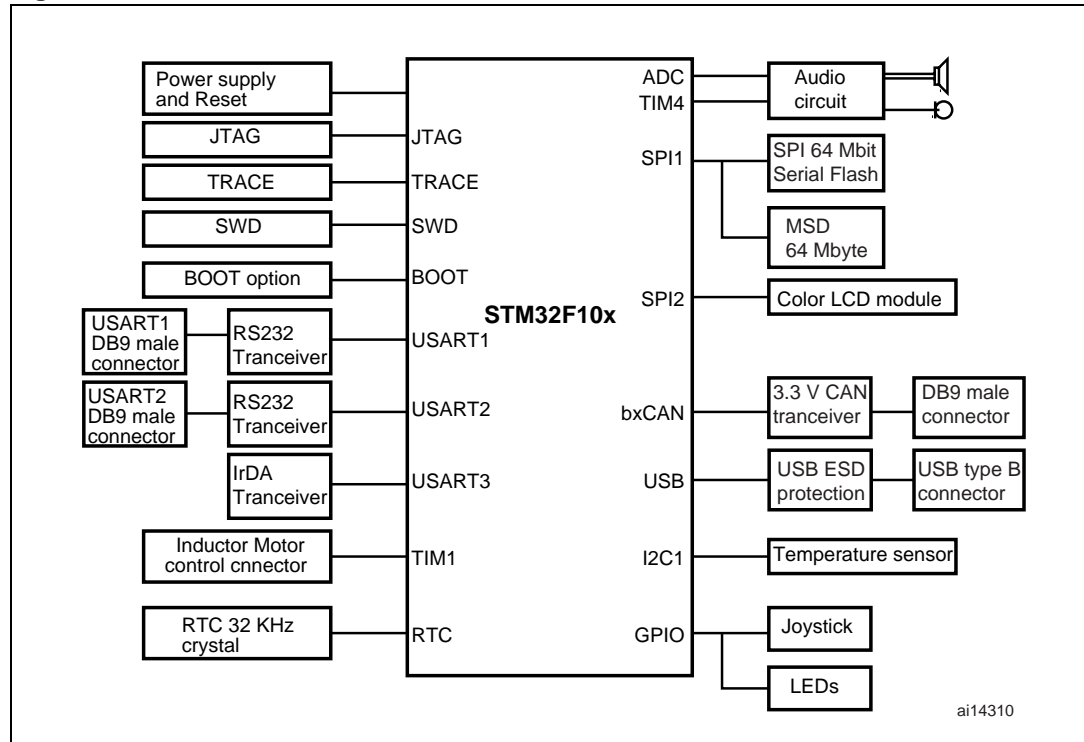
Figure 49.	No MSD connected.	39
Figure 50.	Funny submenu	39
Figure 51.	Animation	40
Figure 52.	Help submenu.	40
Figure 53.	Joystick buttons	41
Figure 54.	Second Help slide.	41
Figure 55.	About submenu.	42
Figure 56.	STM3210B-EVAL information.	42
Figure 57.	Demonstration firmware folder structure.	43
Figure 58.	Demonstration firmware architecture	45
Figure 59.	Programming the Flash memory	47
Figure 60.	Selecting the STM3210B-EVAL_SPIFlash.dfu file	48
Figure 61.	Click Yes to continue	48
Figure 62.	Flash memory programming.	49
Figure 63.	Programming the demonstration	50
Figure 64.	Selecting the STM3210B-EVAL_Demo.dfu file	51
Figure 65.	Click Yes to continue	51
Figure 66.	Internal Flash memory programming	52
Figure 67.	DFU mode left.	53

1 Functional description

The STM32F10xxx microcontroller evaluation board provides a development and demonstration platform for STM32F10xxx-based applications. It is designed to allow the user to try out the major functions of the STM32F10xxx microcontroller.

Figure 1 summarizes the main functional blocks of the evaluation board.

Figure 1. Evaluation board overview



Note: Due to a Joystick pin assignment change between the STM3210B-EVAL board RevA (Joystick DOWN pin connected to PA.02) and the STM3210B-EVAL board RevB (Joystick DOWN pin connected to PD.14), the demonstration may not work correctly on the STM3210B-EVAL board RevA. As a temporary solution, connecting the PA.02 and PD.14 pins with a wire (on the CN12 connector) will allow the demonstration to run correctly even on an STM3210B-EVAL RevA board.

1.1 Power control

The evaluation board can be powered by an external 5 V supply or from the USB connector, all other required voltages are provided by on-board voltage regulators.

1.2 Clocking

Two clock sources are available on the STM3210B-EVAL evaluation board:

- 32 kHz Crystal for embedded RTC.
- 8 MHz Crystal for STM32F10xxx.

1.3 Reset control

The Reset can be generated by hardware or software:

- Reset button: activates the RESET input when pressed
- JTAG reset

1.4 Debug JTAG interface

Software debug is done via the standard ARM® JTAG connection: 20-pin IDC (insulation displacement connector) for connection to the standard ARM host interface.

1.5 Serial wire debugger interface

The Serial Wire Debug Port (SWD-DP) provides a 2-pin (clock + data) interface to the AHP-AP port.

1.6 Display devices

1.6.1 LCD

A color LCD module is mounted on the STM3210B-EVAL board. It is interfaced through the SPI2 peripheral.

1.6.2 LED

Four general-purpose LEDs are available. They are used as a display.

1.7 Interfaces

1.7.1 RS232

The STM32F10xxx evaluation board (STM3210B-EVAL) provides two on-board RS-232 serial ports. Both RS232 ports are terminated by DB9 connectors.

1.7.2 CAN

The STM3210B-EVAL evaluation board supports CAN 2.0 A/CAN 2.0 B-compliant CAN bus communication based on a 3.3 V CAN transceiver. Both High-speed mode and slope control mode are available and can be selected by setting a dedicated jumper.

1.7.3 USB

The STM3210B-EVAL evaluation board supports full-speed USB 2.0 communication thanks to its embedded USB peripheral.

1.8 Motor control

The STM3210B-EVAL evaluation board supports inductor motor control via a 34-pin connector that provides all required control and feedback signals to and from the motor power-driving board.

1.9 IrDA

The STM3210B-EVAL evaluation board supports IrDA communication. The interface is mounted on USART3

1.10 Miscellaneous peripherals

1.10.1 Joystick

Four-direction joystick with selection key.

1.10.2 Push-buttons

The following push-buttons are provided:

- Key: user push-button
- Tamper: user push-button
- Wakeup: push-button used to wake up the processor from low-power modes

1.10.3 12-bit analog-to-digital converter (ADC)

Varistor: ADC channel14 connected to an on-board variable resistor. The variable resistor provides a voltage in the range of 0 V to 3.3 V.

1.10.4 Audio

The STM3210B-EVAL evaluation board supports both audio (*.wav format) recording and playback.

1.10.5 Storage memories

The STM3210B-EVAL evaluation board has two different storage memories: an SPI Flash memory of 64 Mbits and an MSD (mass storage device) of 64 Mbytes connected together on SPI1.

1.10.6 Temperature sensor

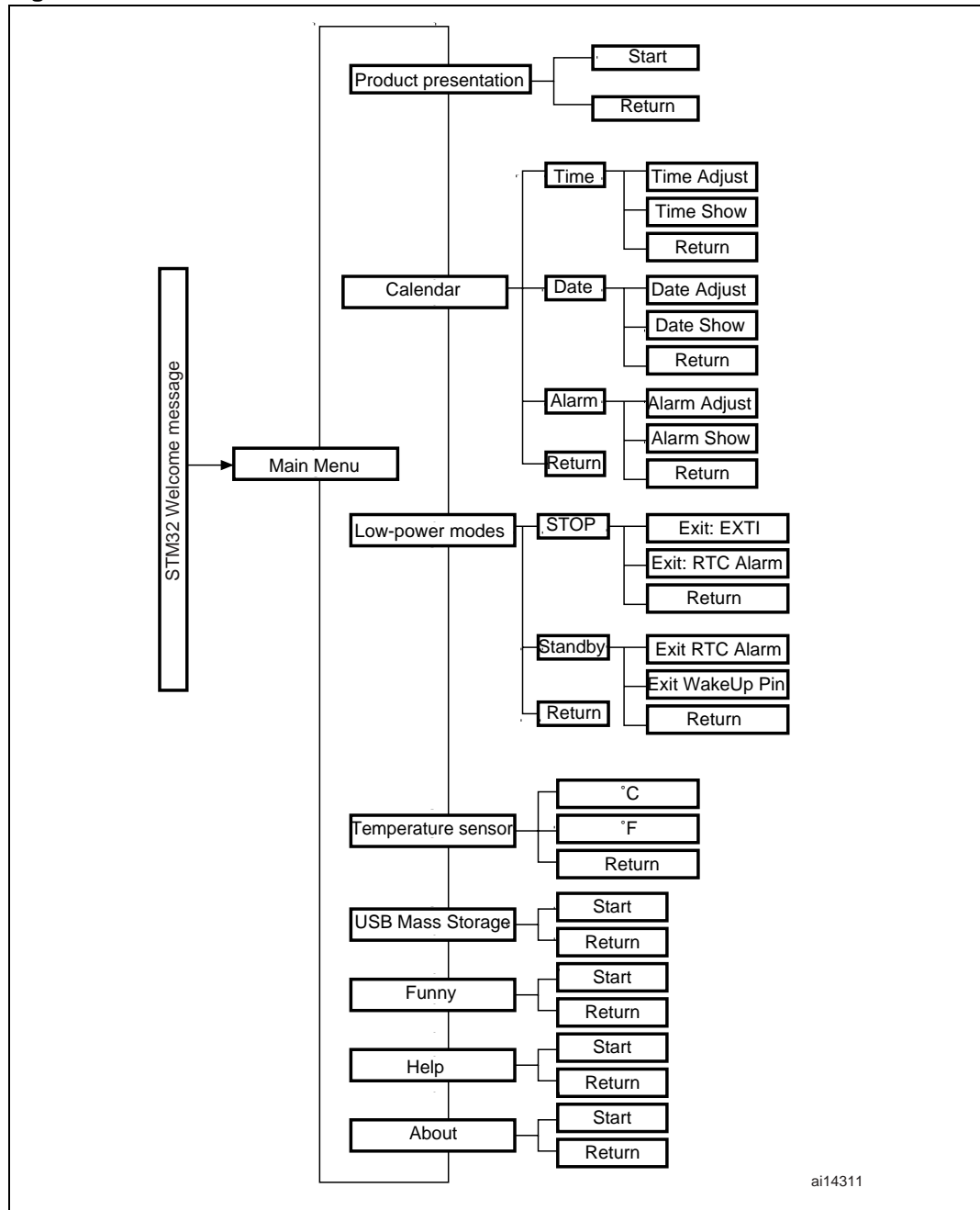
The STM3210B-EVAL evaluation board includes an I²C temperature sensor connected to the I2C1 peripheral.

2 Running the demonstration

2.1 Menu

Figure 2 shows the menu system of the STM32F10xxx demonstration. The main menu is shown on the left hand side. The UP, DOWN, RIGHT and LEFT joystick directions allow the user to navigate between items in the main menu and the submenus. To enter a submenu, press the SEL push-button on the joystick (pressing the SEL push-button causes the switch to close and selects the current menu/submenu). To exit a submenu select the **Return** menu and press SEL.

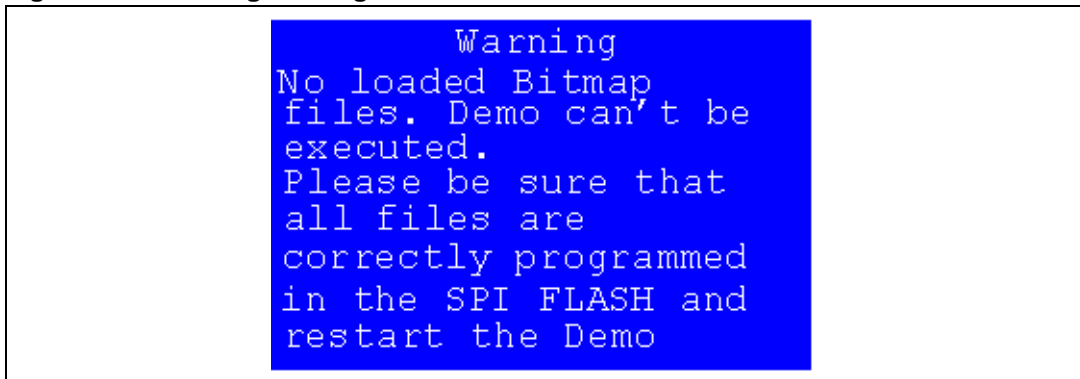
Figure 2. Structure of the demonstration menus



2.1.1 Demonstration startup

After a board reset, at demonstration startup, the icons and bitmap files are checked in the SPI Flash memory. All the icons have to be correctly programmed in the SPI Flash memory for the demonstration to start, so if an icon is missing, the demonstration will not start and the message shown in [Figure 3](#) will be displayed on the LCD screen.

Figure 3. Warning message



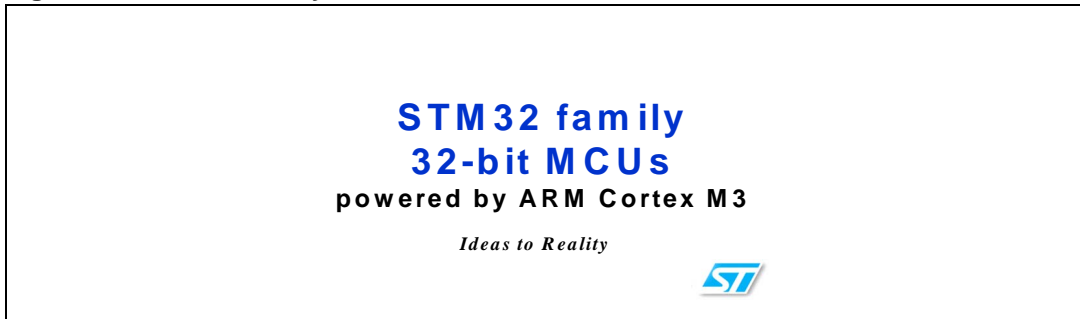
However, if the icons are correctly loaded into the SPI Flash memory, the Welcome screen is displayed and the ST Logo appears on the LCD (see [Figure 4](#)).

Figure 4. ST Logo



Then, after 0.8 second, the next slide is displayed on the LCD screen as shown in [Figure 5](#).

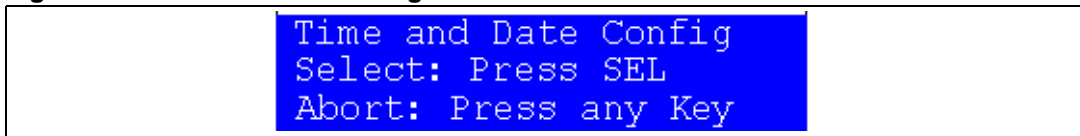
Figure 5. STM32 family



When the board is powered up for the first time and no power supply on V_{BAT} is detected (Battery), the user is prompted to set the time, year, month and day. The user may choose to ignore it by pressing any key except for the SEL push-button to abort the configuration sequence. To set the time and date, the user should press SEL and follow the setting sequence.

The message shown in [Figure 6](#) appears on the LCD screen.

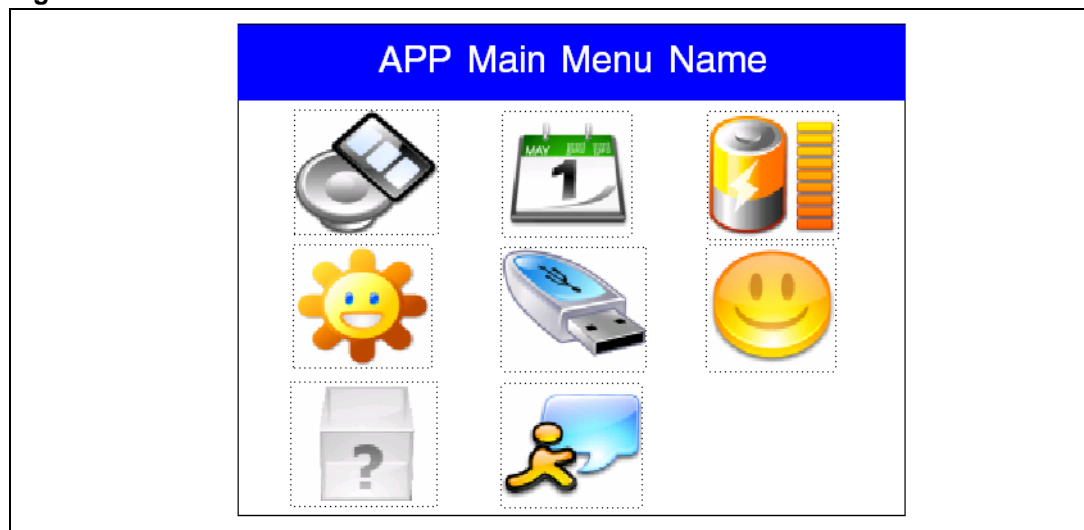
Figure 6. Time and date configuration



- Note:
- 1 If the user chooses to configure the time and date, the **Time Adjust** and **Date Adjust** menu are displayed. Otherwise the main menu is displayed and the user can set the time parameters in the **Calendar** menu. To set the time/date use the joystick UP/DOWN and SEL push-buttons.
 - 2 If the time configuration has already been done, then the number of elapsed days (higher than 1 day) from the last time the evaluation board was powered up appears on the LCD screen. It is soon followed by the current date.

Once the time and date have been set, the main menu appears. It is displayed in the form of a set of icons. It presents all the submenus in the same screen. The user can navigate using the UP, DOWN, RIGHT and LEFT joystick directions to select the desired submenu. To enter the desired submenu, the user has to press the SEL joystick push-button, and the new submenu corresponding to the selected icon is displayed.

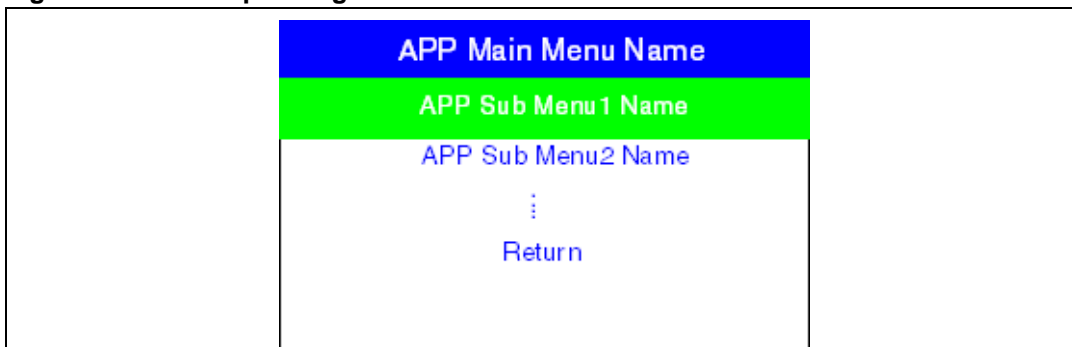
Figure 7. Main menu



1. The icons shown in [Figure 7](http://commons.wikimedia.org/wiki/Crystal_Clear) are taken from http://commons.wikimedia.org/wiki/Crystal_Clear.

Once a submenu has been selected, the name of the application is listed at the top of the display and all the corresponding submenus are listed below as shown in *Figure 8*.

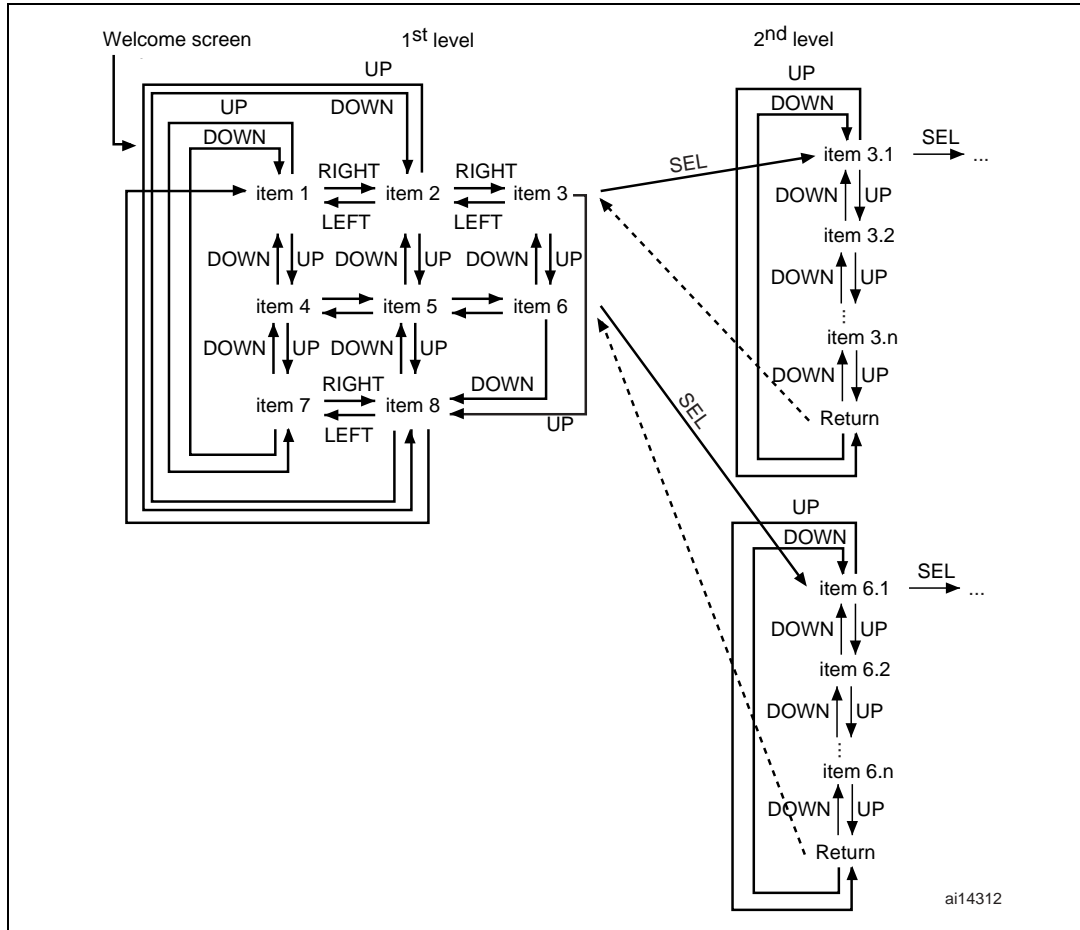
Figure 8. Corresponding submenus



2.1.2 Navigation

The demonstration menu is based on circular navigation, submenu selection, item selection and back capability as described in [Figure 9](#).

Figure 9. Navigating in the demonstration menus



The user navigates using the joystick push-buttons: RIGHT, LEFT, SEL, UP and DOWN located on the evaluation board.

- UP, DOWN, RIGHT and LEFT push-buttons are used to perform circular navigation in the main menu and the current menu items
- the SEL push-button selects the current item
- the UP and DOWN push-buttons are used for vertical navigation in the submenus
- To return to the upper menu, go to the Return line and press SEL

2.2 Clock sources

2.2.1 Clock control

The STM32F10xxx internal clocks are derived from the HSE (clocked by the external 8 MHz crystal).

In this demo application, the different system clocks are configured as follows:

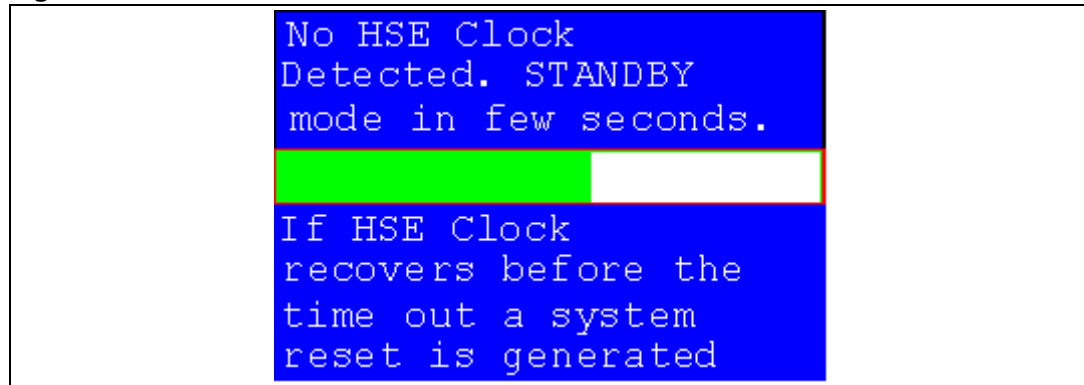
- System clock is set to 72 MHz: The PLL will be used as the system clock source: 72 MHz (two wait states, Flash Prefetch Buffer Enabled).
- HCLK frequency is set to 72 MHz
- Timer clock (TIMCLK) is set to 72 MHz
- PCLK1 is set to 36 MHz
- PCLK2 is set to 72 MHz
- USB clock (USBCLK) is set to 48 MHz (internal clock: PLLCLK / 1.5)

Only the RTC is clocked by a 32 kHz external oscillator.

2.2.2 Clock failure

At any demo level, if no clock is present on OSC_IN (broken or disconnected Crystal), the message shown in [Figure 10](#) is displayed on the LCD screen.

Figure 10. No HSE clock detected

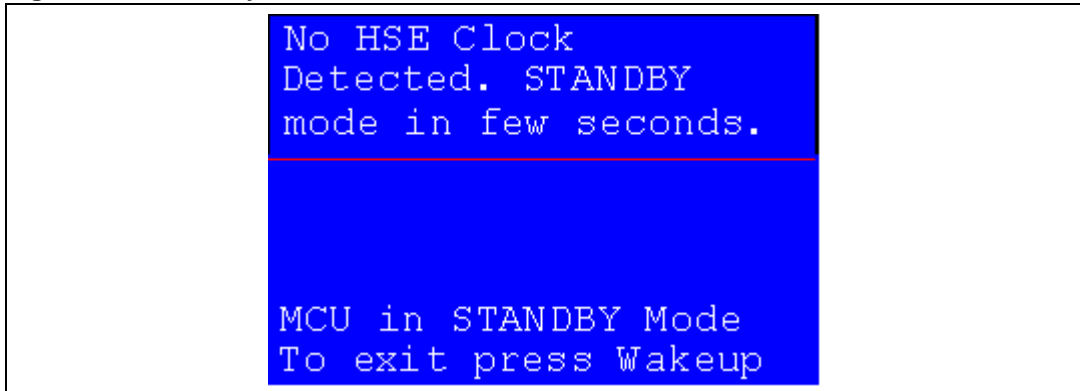


If the 8 MHz Crystal is not reconnected in the next few seconds, the MCU enters the Standby mode. If the 8 MHz crystal is reconnected within a few seconds, a system reset is generated.

Note: The clock security system (CSS) feeds the MCU with the HSI OSC used as an emergency clock if no clock is detected.

When a timeout occurs, the MCU enters the Standby mode and the message shown in [Figure 11](#) is displayed on the LCD screen.

Figure 11. Standby mode entered



1. The demonstration will not restart as long as the 8 MHz crystal is not present.
2. Connecting the 8 MHz crystal after reset may not restart the demonstration correctly. The crystal must be connected before starting the demonstration.

2.3 STM32F10xxx resources

2.3.1 Peripherals

All used peripherals are described in [Table 1](#).

Table 1. STM32F10xxx evaluation board peripherals

Used peripherals	Application
I2C1	Temperature sensor
BKP	Calendar + Demo kernel
EXTI	Menu navigation + joystick + Push Button + low-power modes
GPIO	All applications + LEDs
NVIC	All applications using interrupts
PWR	Low-power modes
RCC	All applications + Demo kernel
RTC	Calendar
SPI1 & SPI2	Color LCD + SPI Flash + MSD
SysTick	Generate 10 ms time base
TIMx	Wave Player
TIM1	LED toggling

2.3.2 Interrupts

Table 2 shows all the enabled interrupts.

Table 2. STM32F10xxx demo interrupts

Interrupts	Priority	Used for
SysTick	Preemption: 0 SubPriority: 0	System timing
RTC	Preemption: 0 SubPriority: 0	Calendar, date update
NMI	Preemption(fixed): -2	CSS interrupt
EXTI9_5	Preemption: 2 SubPriority: 1	Menu navigation
EXTI15_10	Preemption: 2 SubPriority: 0	Menu navigation
TIM2	Preemption: 0 SubPriority: 1	Wave Player
USB_LP_CAN_RX0	Preemption: 1 SubPriority: 0	USB Mass Storage
USB_HP_CAN_TX	Preemption: 0 SubPriority: 1	USB Mass Storage
TIM1_UP	Preemption: 1 SubPriority: 3	LED toggling
RTCAlarm	Preemption: 1 SubPriority: 1	Alarm generation

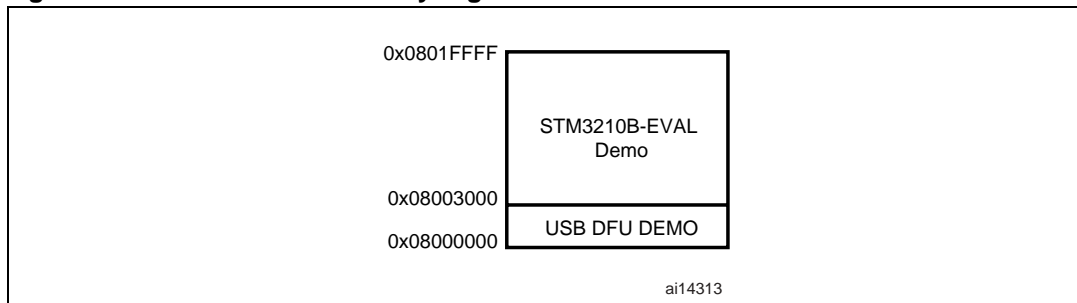
2.3.3 External interrupts

Table 3. STM32F10xxx demo external interrupts

External interrupts	Used for
EXTI line14	Joystick DOWN (interrupt mode, falling edge)
EXTI line8	Joystick UP (interrupt mode, falling edge)
EXTI line12	Joystick SEL (interrupt mode, falling edge)
EXTI line9	User Button (interrupt mode, falling edge)
EXTI line17	RTC Alarm (interrupt mode, rising edge)

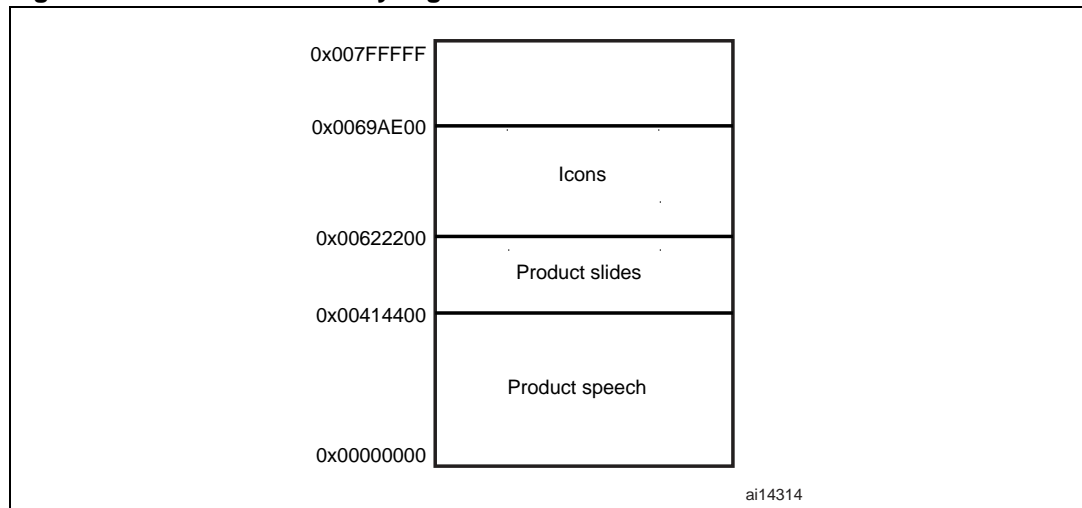
2.3.4 Internal memory size

Figure 12. Internal Flash memory organization



2.3.5 External memory size

Figure 13. SPI Flash memory organization



2.4 Demo applications

The following section provides a detailed description of each part of the demonstration.

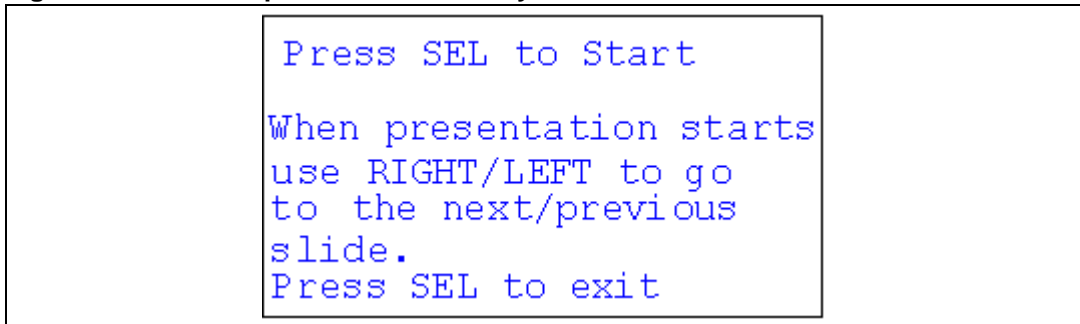
- Note:
- 1 In the demonstration, the core runs at $HCLK = 72\text{ MHz}$.
 - 2 Four LEDs: LD1, LD2, LD3 and LD4 are always flashing at a frequency depending on the core clock.

2.4.1 Product presentation

This part of the demonstration is dedicated to listing all the embedded STM32F10xxx peripherals and features. This presentation of the microcontroller is made with a set of slides accompanied by a speech. Each slide is associated with a dedicated speech. When the user starts the product presentation, the first slide appears and the corresponding speech starts. Once the speech is finished, the second slide is displayed accompanied by its speech and so on until the last slide.

When the **Product presentation** menu is selected, the message shown in [Figure 14](#) is displayed on the LCD screen.

Figure 14. Product presentation is ready to start



Product presentation slides

The set of slides is composed of 14 slides where all features and advantages of the STM32F10xxx are listed. [Figure 15](#) and [Figure 16](#) show the first and last slides, respectively.

Figure 15. First presentation slide

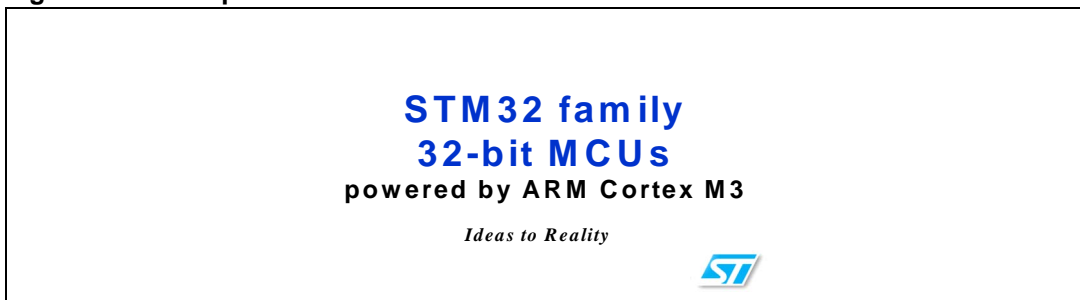
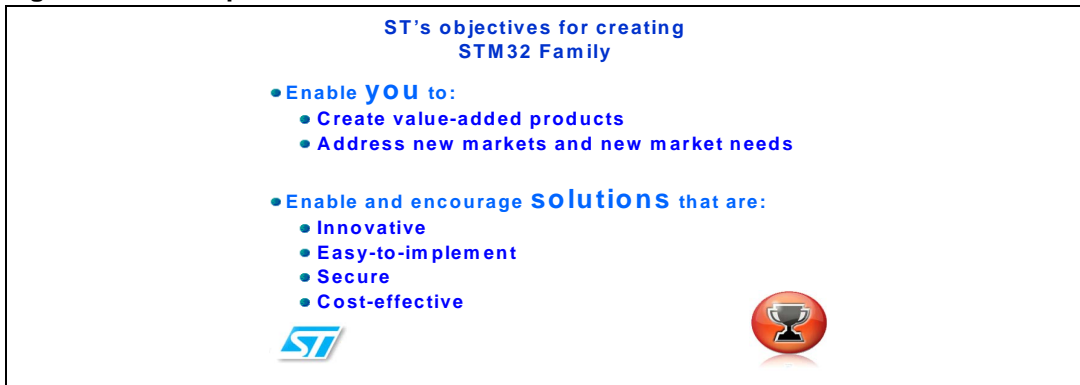


Figure 16. Last presentation slide



Product presentation speech

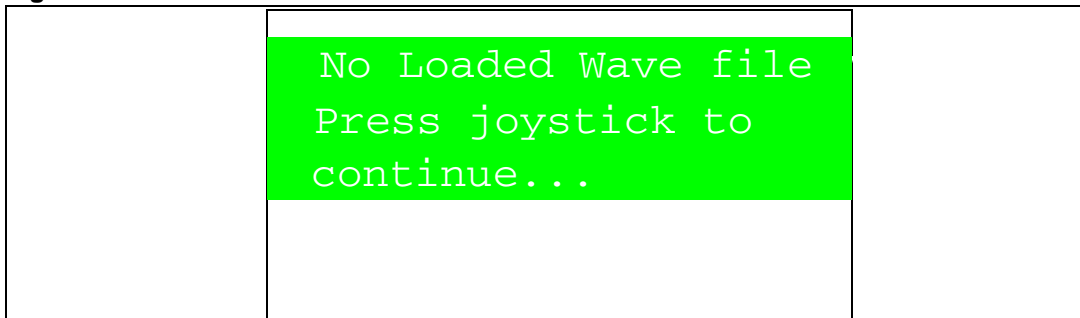
The STM32F10xxx microcontroller has several embedded timers that can be used for timing purposes and for generating the output signals. In the latter case, two timers are used. The first timer (TIM4) is configured to generate a PWM signal with a tunable duty cycle. The second timer (TIM2) is used to generate an Update interrupt (for this speech with a 11.025 KHz frequency) during which the new TIM4 duty cycle is loaded.

The Product presentation speech wave file properties are:

- Playing time: 6 min 27 s
- File size: 4,277,166 bytes
- Format tag: PCM
- Channels: MONO
- Sample Rate: 11025 Hz
- Bits per sample: 8 bits

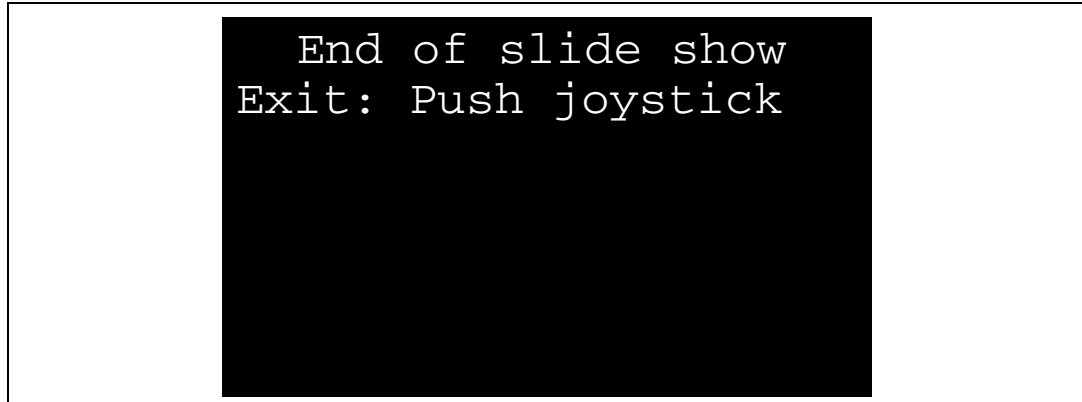
Note: If the promotion presentation speech wave is not loaded into the reserved SPI Flash memory, the message shown in [Figure 17](#) is displayed on the LCD screen.

Figure 17. No loaded wave file



To stop the product presentation slide display and speech, push the SEL push-button. The message shown in [Figure 18](#) is displayed.

Figure 18. End of slide show



At the end of the product presentation or if the presentation was stopped, the user only has to press any key of the joystick to exit and return to the **Product Presentation** submenu.

2.4.2 Calendar

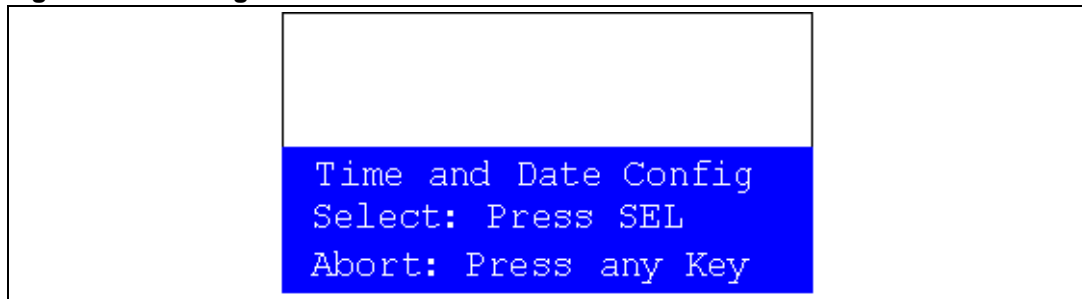
The STM32F10xxx features a real-time clock (RTC) that provides a set of continuously running counters. These can be used, with suitable software, to implement a clock-calendar function. The counter values can be written to set the current time of the system.

This submenu is used to configure the time, date and alarm. The date, time and alarm settings are not lost when the board is powered off owing to the battery connected to the V_{BAT} pin. The V_{BAT} pin supplies power to the RTC unit, allowing the RTC to operate even when the main digital supply (V_{DD}) is turned off.

Note: To be able to use the battery to back up the RTC, the JP11 jumper must be in the position 1-2 in the STM3210B-EVAL board.

In any submenu, if the time and date parameters have not yet been configured, the message shown in [Figure 19](#) is displayed on the LCD screen.

Figure 19. Setting the time and date



The user has the choice to set or not the time, year, month and day. Press any key (except for SEL) to ignore the prompt and abort the configuration sequence. Press on SEL and follow the setting sequence to set the time and date.

Time submenu

This submenu is divided into two items that allow the user to display or set the current time.

- **Time Adjust:** After the evaluation board is powered up, the user has to select this submenu to change the default time (00:00:00) to the current time. Once "Time Adjust" has been selected, the first digit of the hour field can be changed. Pressing the "UP" button will display the current value plus one. Pressing the "DOWN" button will display the previous digit value. After setting the digit value, press "SEL", the cursor automatically jumps to the next digit. When all the time digits have been set, the "Time" submenu appears. Some digit values are limited to a range of values depending on the field (hour, minute or seconds). The following message (with the default time or the current time) is displayed on the LCD when this submenu is selected.

Figure 20. Time Adjust submenu



- **Time Show:** this item displays the current time. If time and date have not been configured before, a message is displayed, that gives the choice to set the time and date or, to exit to the upper submenu. When this submenu is selected, the message shown in [Figure 21](#) appears on the LCD. In the example, the time has not been set yet.

Figure 21. Time Show submenu



To exit the **Time Show** submenu press the SEL push-button. To exit **Time** submenu select the **Return** line and press the SEL push-button.

Date submenu

This submenu is divided into two items that allow the user to display or set the current date.

- **Date Adjust:** this item has to be selected after each power-up in order to set the current date. If the time and date have not been configured before, a message is displayed, that gives the choice to set the time and date or, to exit to the upper submenu. The user is asked to set the current date to be stored in the application memory. The date is displayed as **Year, Month, Week Nbr, Day Nbr** (number of the day in the year) with the selected day shown in the month. There is no default date since the user has to set the date at least once. Once the submenu has been selected, the user starts by setting the **Year**, then the **Month** and the day of the selected month. The **Month** and the **Year** are selected using the UP or DOWN push-button. For the day, the UP, DOWN, RIGHT and LEFT push-buttons can be used. Pressing the UP push-button will display the current value plus one, pressing the DOWN push-button will display the previous value. To confirm the selected month, the user has to press the SEL push-button and an automatic jump to the year configuration takes place. The same procedure is applicable for the year configuration. After configuring the day, pressing the SEL push-button will store the entered value and exit to the **Date** submenu. The current date value is then shown and the user can change the setting if required. The messages shown in *Figure 22*, *Figure 23* and *Figure 24* are successively displayed on the LCD when this submenu is selected.

Figure 22. Setting the Year

Month		Year				
Week Nbr		Day Nbr				
Mo	Tu	We	Th	Fr	Sa	Su
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

UP/DOWN: Set Year

Figure 23. Setting the Month

Month						Year
Week Nbr						Day Nbr
Mo	Tu	We	Th	Fr	Sa	Su
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

UP/DOWN: Set Month

Figure 24. Setting the day of the month

Month						Year
Week Nbr						Day Nbr
Mo	Tu	We	Th	Fr	Sa	Su
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

All Key: Set Day

- **Date Show:** this item displays the current date. If the time and date have not been configured before, a message is displayed, that gives the choice to set the time and date or, to exit to the upper submenu. The message shown in *Figure 25* is displayed on the LCD when the submenu is selected (with the date already configured).

Figure 25. Exiting the Date Show submenu

Month						Year
Week Nbr						Day Nbr
Mo	Tu	We	Th	Fr	Sa	Su
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

To Exit Press SEL

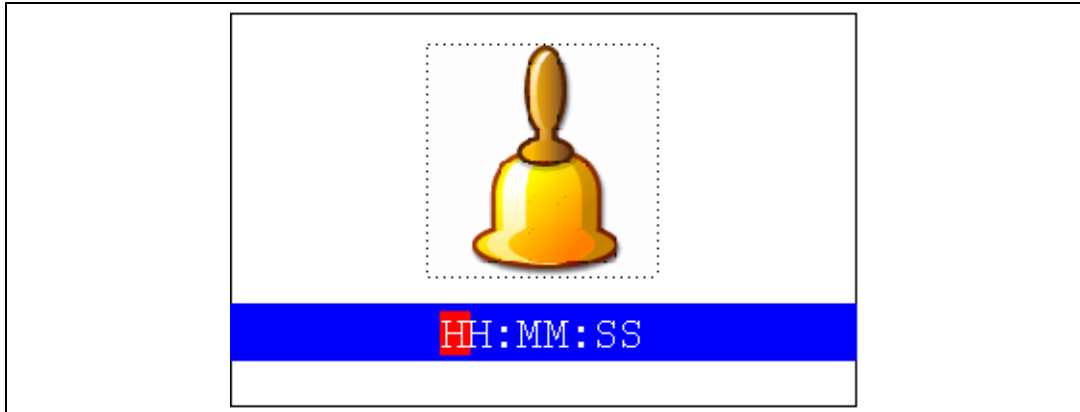
To exit this submenu press the SEL push-button. To exit the **Date** submenu, select the **Return** line and press the SEL push-button.

Alarm submenu

Using this submenu, the user can configure the alarm activation time. When the alarm time value is reached, all the LEDs (LED1 to LED4) start flashing together and so for 30 seconds. This submenu is divided into two items to display or set the current alarm.

- **Alarm Adjust:** the alarm time activation is set in the same way as the time is set in the **Time Adjust** submenu. The following messages are successively displayed on the LCD when this submenu is selected:

Figure 26. Setting the alarm activation time



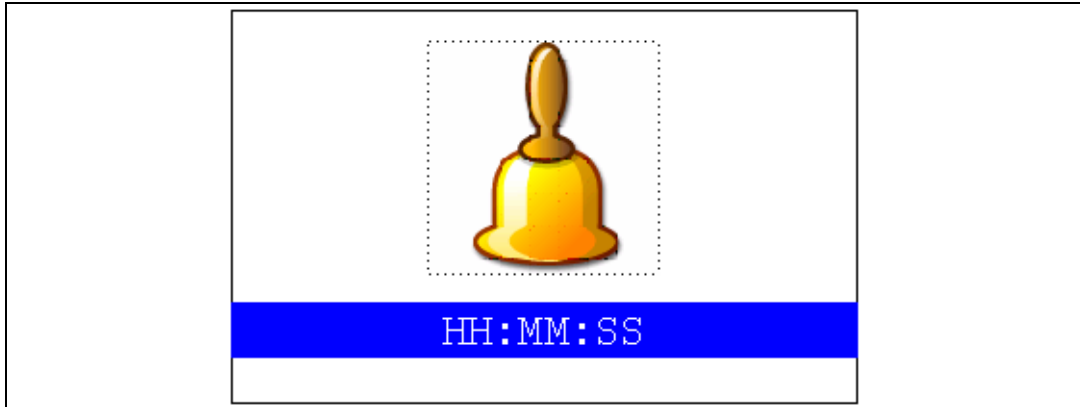
- **Alarm Show:** this item displays the current alarm time. The default Alarm activation time displayed after power-up and before setting in the **Alarm Adjust** submenu is 00:00:00. If the time and date have not yet been configured, the message shown in [Figure 27](#) is displayed.

Figure 27. The time and date need setting



By then choosing to press SEL, the user is redirected to the **Alarm Adjust** submenu. When this submenu is selected, the screen shown in [Figure 28](#) is displayed on the LCD.

Figure 28. Alarm Show submenu



To exit the **Alarm Show** submenu press the SEL push-button. To exit the **Alarm Adjust** submenu, select the **Return** line and press the SEL push-button.

Note: In the **Alarm Adjust** and **Alarm Show** menus, if the time and date have not yet been configured, the message shown in [Figure 27](#) is displayed on the LCD screen.

2.4.3 Low-power modes

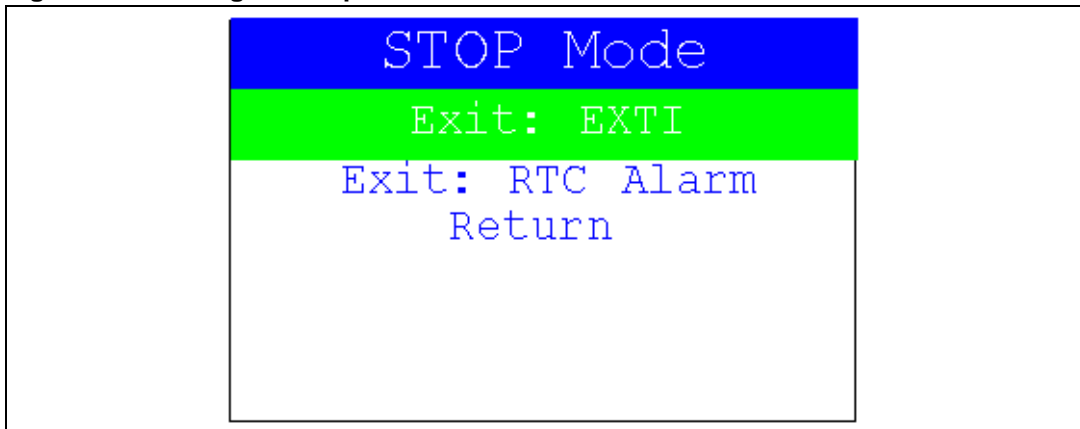
The STM32F10xxx microcontroller provides different operating modes in which the power consumption is reduced. The purpose of this menu is to show the behavior of the microcontroller in different low-power modes. The Stop and Standby modes are taken as examples.

Stop mode menu

This menu allows the user to put the STM32F10xxx in the Stop mode. The software performs the specific instruction sequence needed to enter the Stop mode.

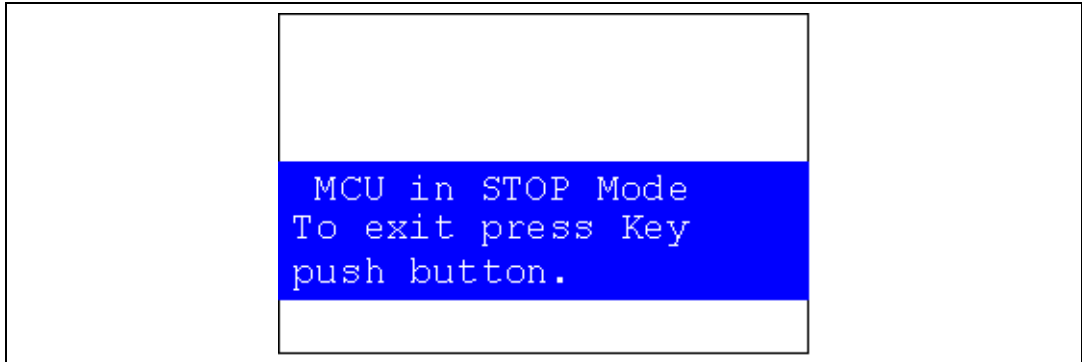
In this application, the STM32F10xxx can be made to exit the Stop mode in two ways as shown in [Figure 29](#).

Figure 29. Exiting the Stop mode



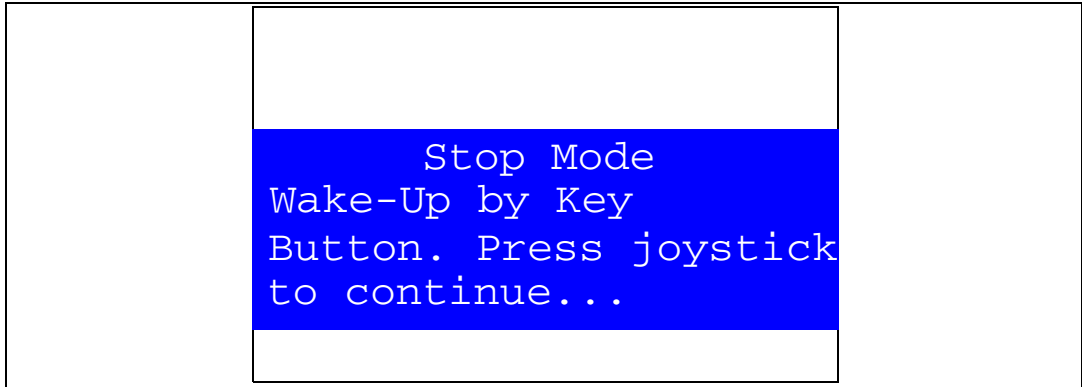
- In the first case, The EXTI Key button is used to make the MCU exit the Stop mode. Once the **Stop mode** submenu has been selected, the red LEDs continue blinking until the "SEL" push-button is pressed, and the system enters the Stop mode. When the MCU is in the Stop mode, the message shown in [Figure 30](#) is displayed on the LCD.

Figure 30. Stop mode entered



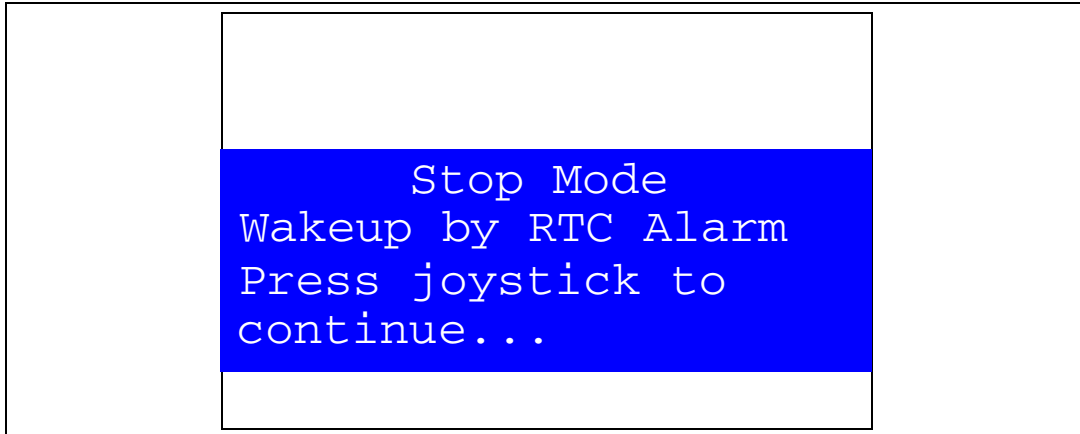
The MCU remains in the Stop mode until the Key push-button is pressed and the message shown in [Figure 31](#) is displayed on the LCD screen. Once the Key push-button has been pressed, the MCU exits the Stop mode. The system clock is then set to 72 MHz and the application resumes execution.

Figure 31. MCU in the Stop mode



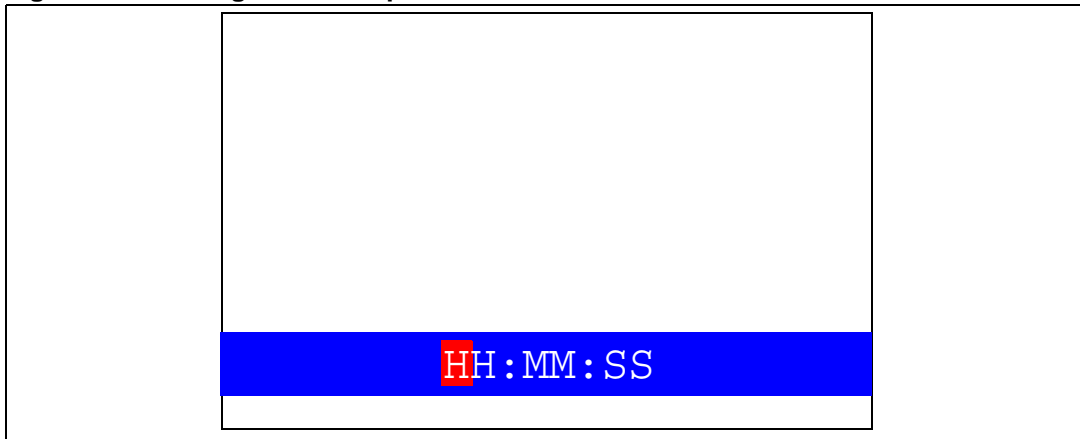
Note: if an RTC Alarm is generated while the MCU is in the Stop mode and the message shown in [Figure 31](#) is displayed (which means that the Key push-button needs to be pressed to exit the Stop mode), the RTC Alarm causes the MCU to exit the Stop mode. The message shown in [Figure 32](#) is then displayed.

Figure 32. RTC Alarm causes the MCU to exit the Stop mode



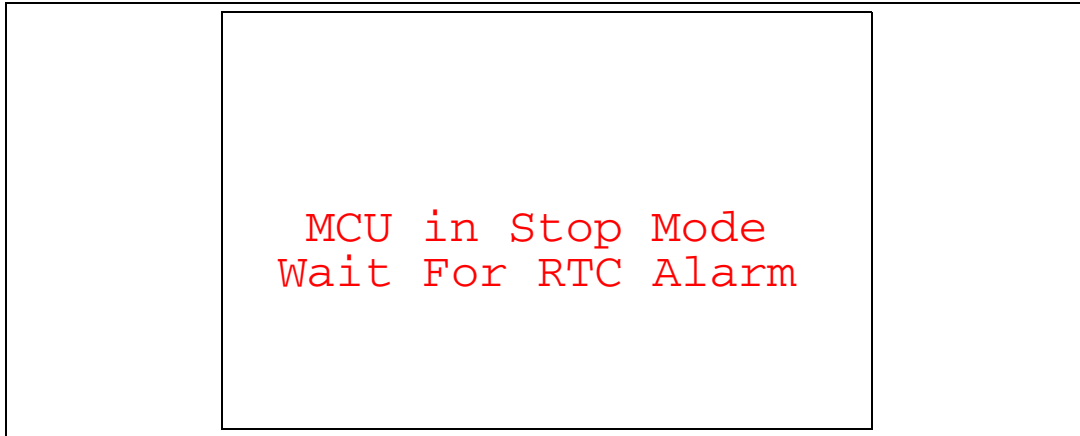
- In the second case, the RTC Alarm wakes up the MCU from the Stop mode after the programmed time has elapsed. When selecting this submenu, the user has to set the alarm to the time when the MCU is to exit the Stop mode. *Figure 33* shows how to set the wakeup time.

Figure 33. Setting the Wakeup time



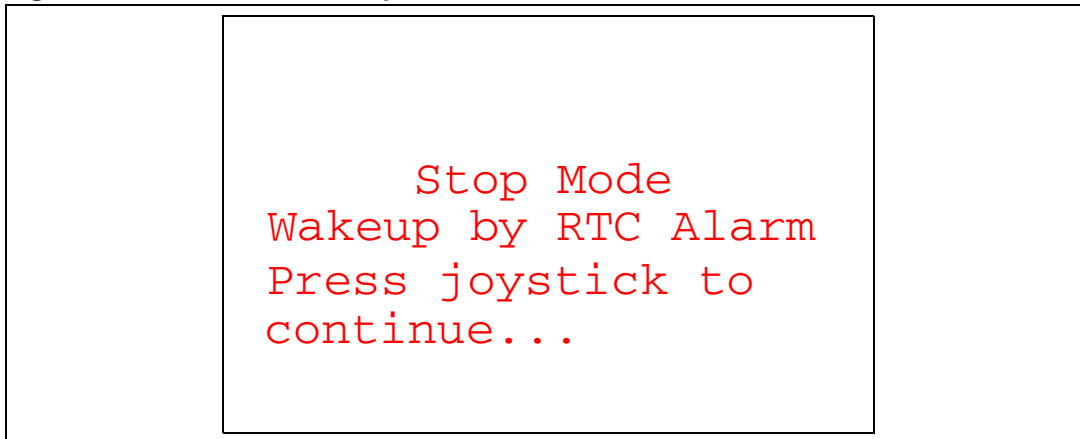
Once the alarm has been configured, the red LEDs stop blinking and the system enters the Stop mode. The message shown in *Figure 34* is displayed on the LCD.

Figure 34. RTC Alarm wakeup configured



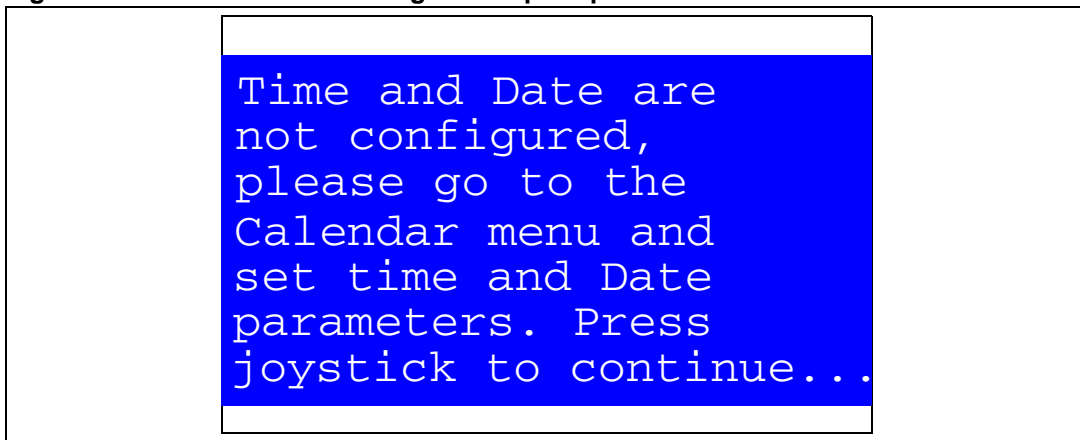
After the programmed time has elapsed, the system exits the Stop mode. The system clock is then set to 72 MHz and the application resumes execution. The message shown in [Figure 35](#) is displayed on the LCD screen.

Figure 35. RTC Alarm wakeup



Note: if the Time and Date have not been set, the message shown in [Figure 36](#) is displayed on the LCD screen.

Figure 36. Time and Date configuration prompt

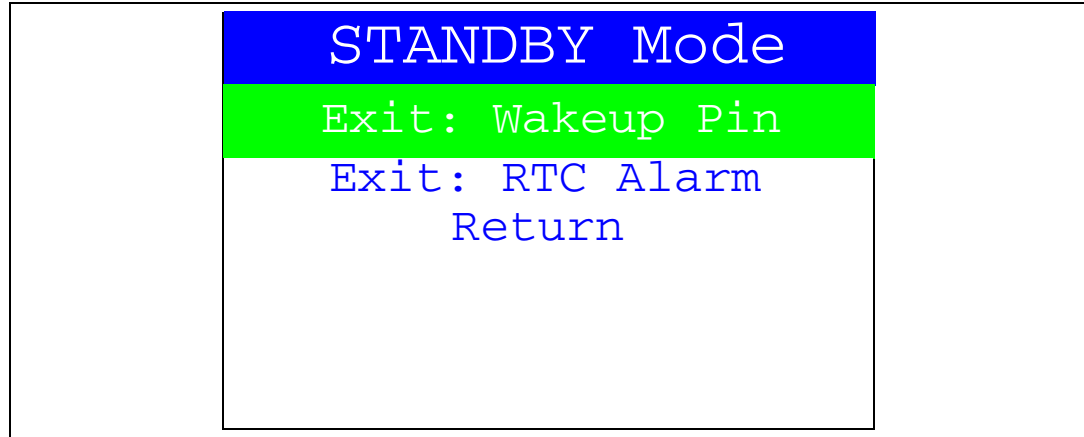


Standby mode menu

This menu allows the user to put the STM32F10xxx in the Standby mode. The software runs the specific instruction sequence needed by the STM32F10xxx to enter the Standby mode.

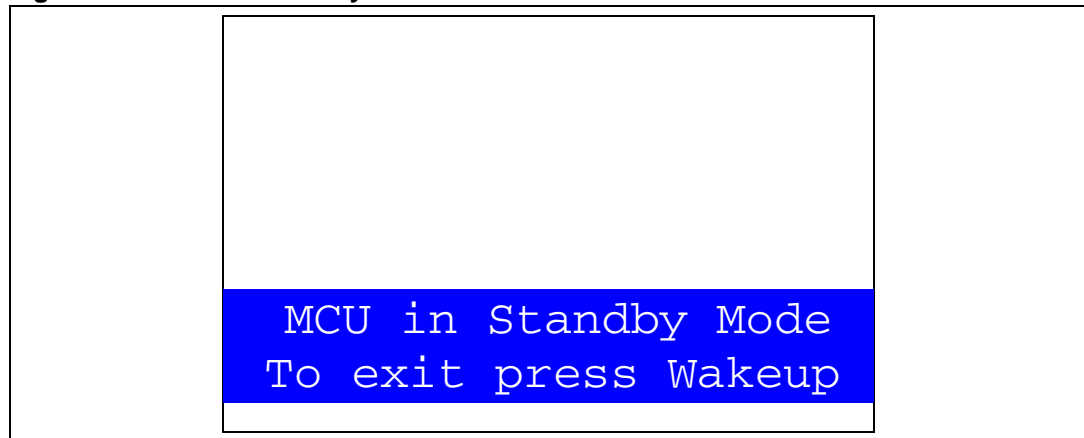
In this application, the STM32F10xxx can be made to exit the Standby mode in two ways as shown in [Figure 37](#).

Figure 37. Entering the Standby mode



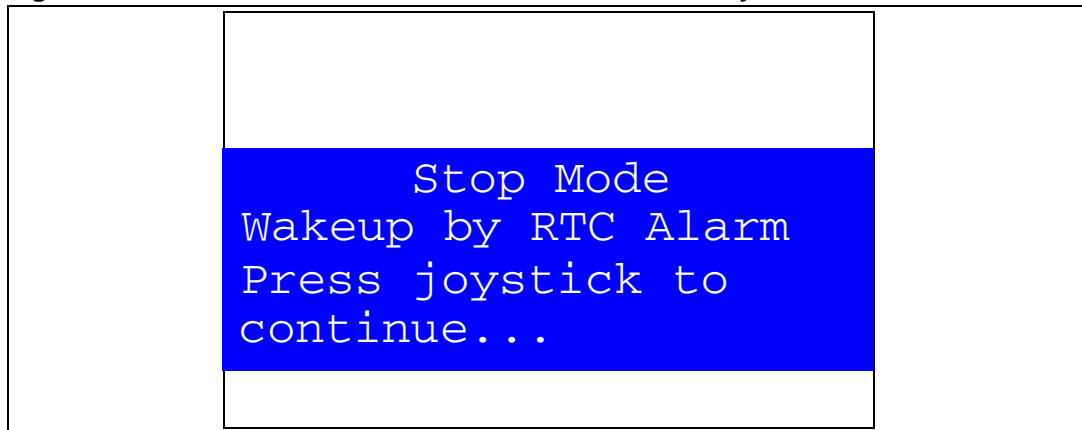
- In the first case, the Wakeup push-button is used to wake up the MCU from Standby mode. Once the **Standby mode** submenu has been selected, the red LEDs continue blinking until the "SEL" push-button is pressed, and the system enters the Standby mode. When the MCU is in the Standby mode, the message shown in [Figure 38](#) is displayed on the LCD

Figure 38. MCU in Standby mode

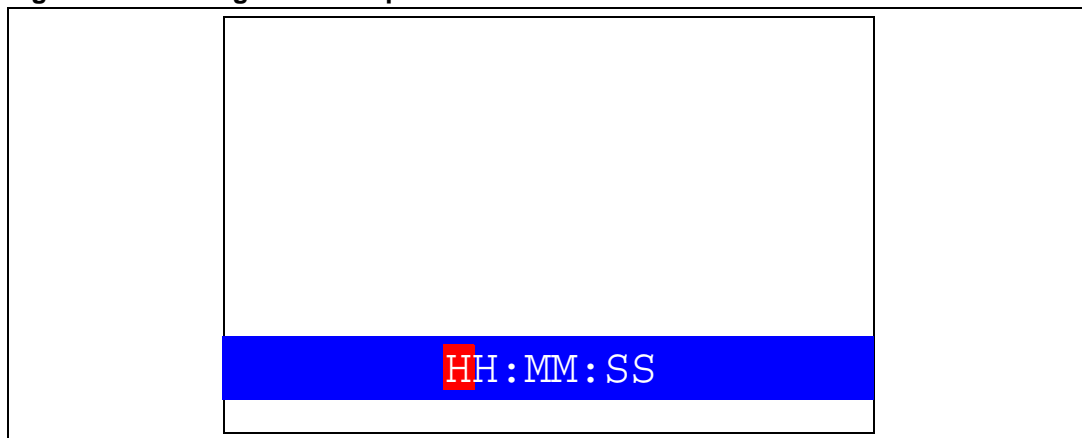


The MCU remains in the Standby mode until the Wake-up push-button is pressed. Once the Wakeup push-button has been pressed, the MCU exits the Standby mode and the system reset signal is generated.

Note: if an RTC Alarm is generated while the MCU is in the Standby mode and the message shown in [Figure 38](#) is displayed (which means that the Wakeup push-button needs to be pressed to exit the Standby mode), the RTC Alarm causes the MCU to exit the Standby mode and a system reset signal is generated.

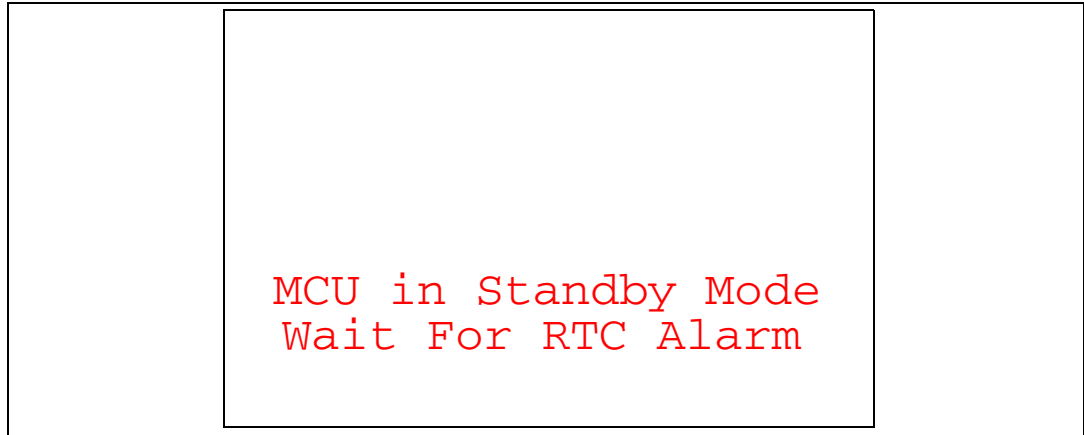
Figure 39. RTC Alarm causes the MCU to exit the Standby mode

- In the second case, the RTC Alarm wakes up the MCU from the Standby mode after the programmed time has elapsed. When selecting this submenu, the user has to set the alarm to the time when the MCU is to exit the Standby mode. [Figure 40](#) shows how to set the wakeup time.

Figure 40. Setting the wakeup time

Once the alarm has been configured, The red LEDs stop blinking and the system enters the Standby mode. The message shown in [Figure 41](#) is then displayed on the LCD.

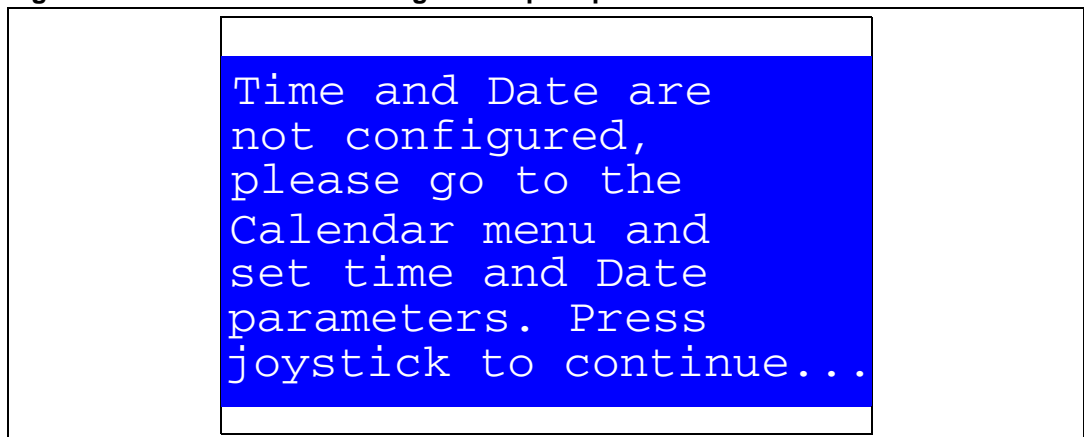
Figure 41. RTC Alarm wakeup configured



After the programmed timing has elapsed, the system exits the Standby mode and a system reset signal is generated.

Note: if the Time and Date have not been set, the message shown in [Figure 42](#) is displayed on the LCD screen.

Figure 42. Time and Date configuration prompt

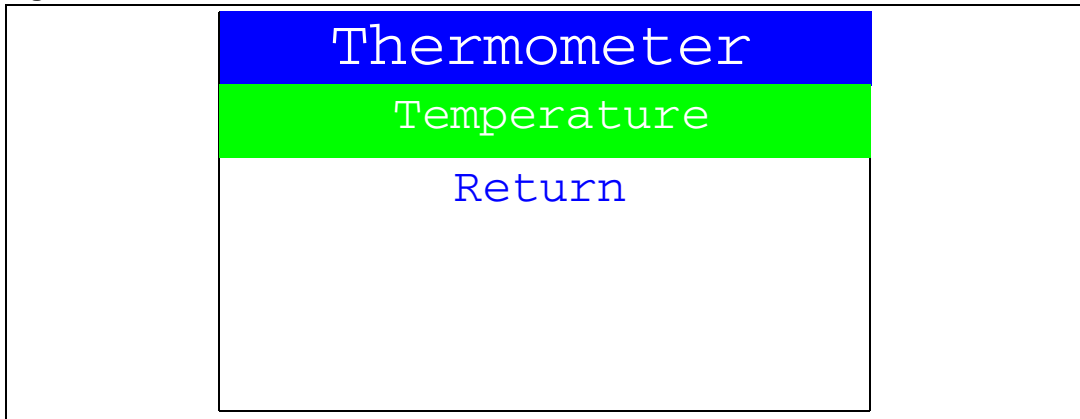


2.4.4 Thermometer

The STM32F10xxx microcontroller has two embedded I²C peripherals that can be connected to any device supporting the I²C protocol including the system management bus (SMBus) mode. An STLM75 (or compatible devices) I²C temperature sensor is mounted on the STM3210B-EVAL board and used to get instantaneous external temperature.

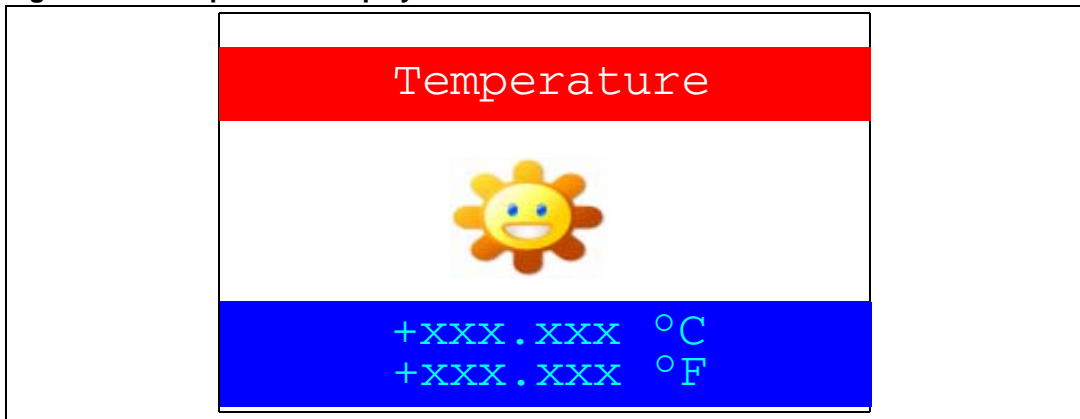
When the **Thermometer** submenu is selected, the message shown in *Figure 43* is displayed on the LCD.

Figure 43. Thermometer submenu selected



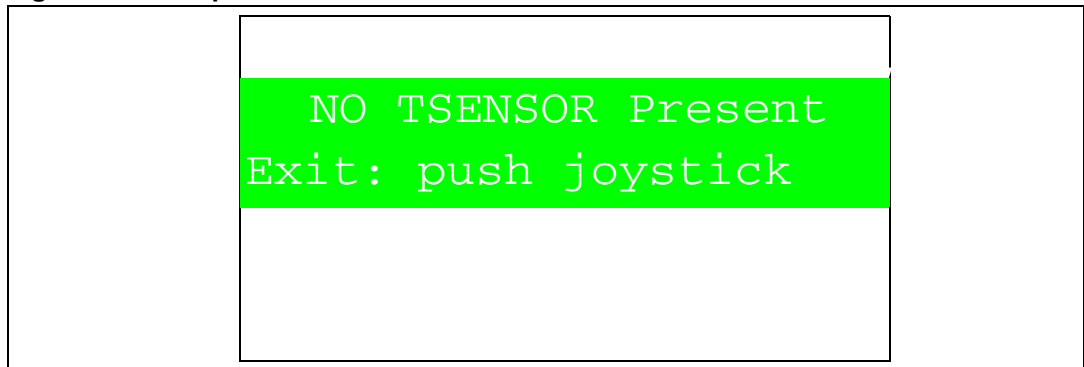
Once the **Temperature** submenu has been selected by pressing the SEL push-button, the temperature value is displayed in Celsius and Fahrenheit as shown in *Figure 44*.

Figure 44. Temperature display



Press any key to return to the **Thermometer** submenu.

Note: Any hardware trouble on the board with the temperature sensor is detected by a test. In this event, the message shown in *Figure 45* is displayed.

Figure 45. Temperature sensor error

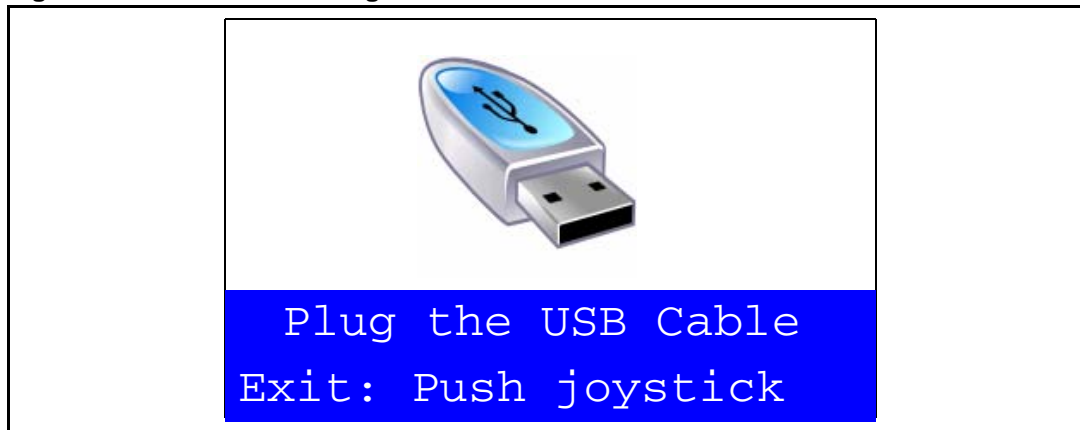
2.4.5 USB mass storage

The STM32F10xxx microcontroller features a USB (universal serial bus) that provides full-speed interface to a USB host PC.

The **USB Mass Storage** submenu is used to configure the USB interface for communication with the PC and run the mass storage demonstration using an MSD card.

Figure 46. USB Mass Storage submenu

If the SEL push-button is pressed at this stage, the message shown in [Figure 47](#) appears on the LCD screen until the cable is plugged in.

Figure 47. USB Mass Storage submenu selected

To return to the previous submenu, the user has to connect a USB cable between the USB connector type B (CN1) and the PC. If the user connects a cable or presses any joystick push-button, the message shown in [Figure 48](#) is displayed on the LCD.

Figure 48. USB cable connected

Once the cable has been connected, the PC recognizes the board as a mass storage device and consequently opens a window to show the contents of the MSD mounted on the STM3210B-EVAL board. The user can transfer files between the MSD and the PC.

If no MSD card is connected, the message shown in [Figure 49](#) is displayed on the LCD.

Figure 49. No MSD connected



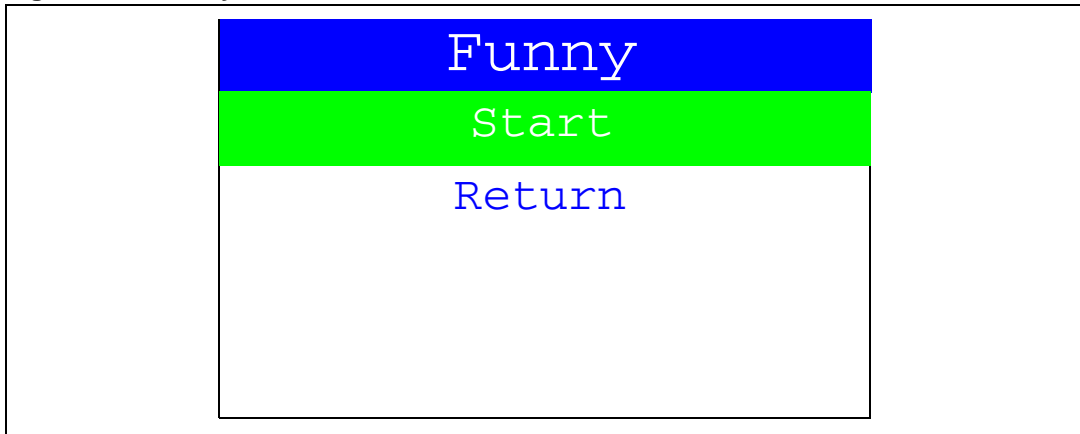
To exit this submenu and return to the **USB Mass Storage** submenu, press SEL.

2.4.6 Funny submenu

The STM32F10xxx microcontroller provides a high-speed SPI interface that is used with a serial color LCD.

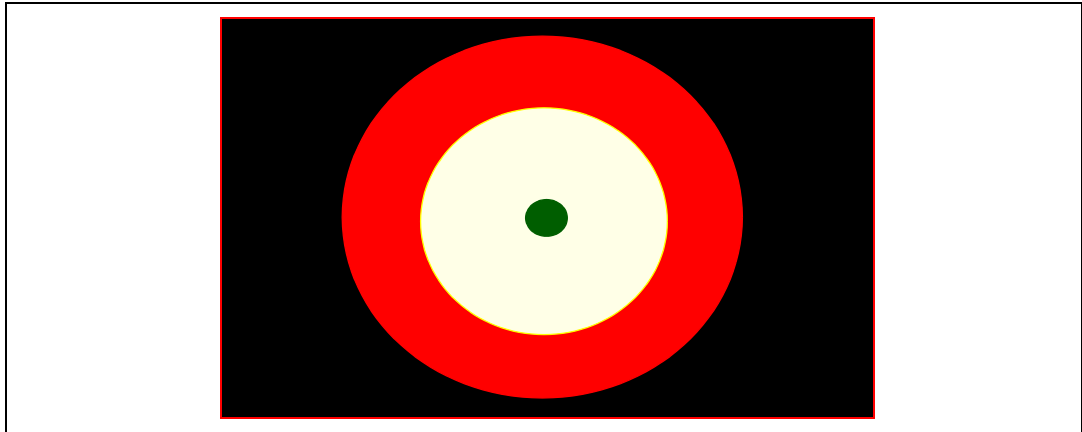
The **Funny** submenu is used to demonstrate the LCD control performance using the embedded SPI interface. The application is a successive display of superposed circles with different colors and radii. The result is a circle animation on LCD.

Figure 50. Funny submenu



If the user presses SEL, the image (from the animation) shown in [Figure 51](#) is displayed on the LCD screen.

Figure 51. Animation

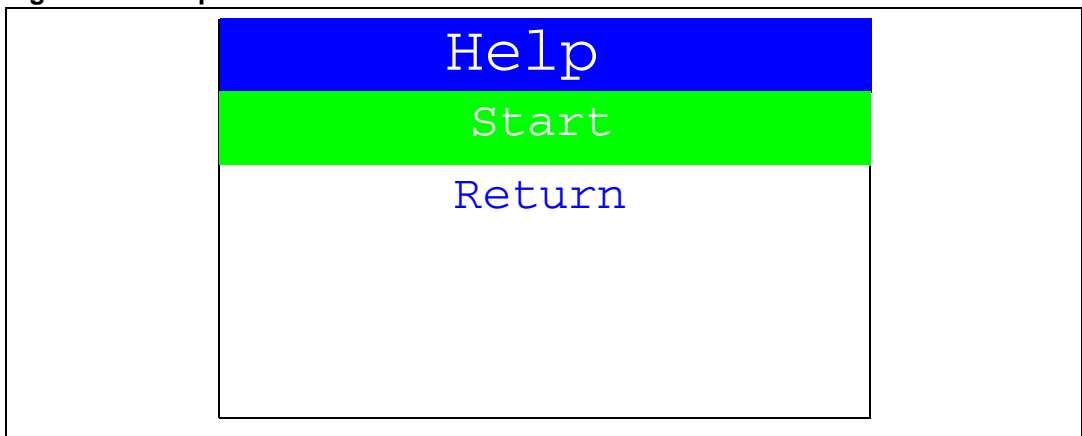


To exit the **Funny** submenu and return to the **Funny** submenu, press SEL.

2.4.7 Help submenu

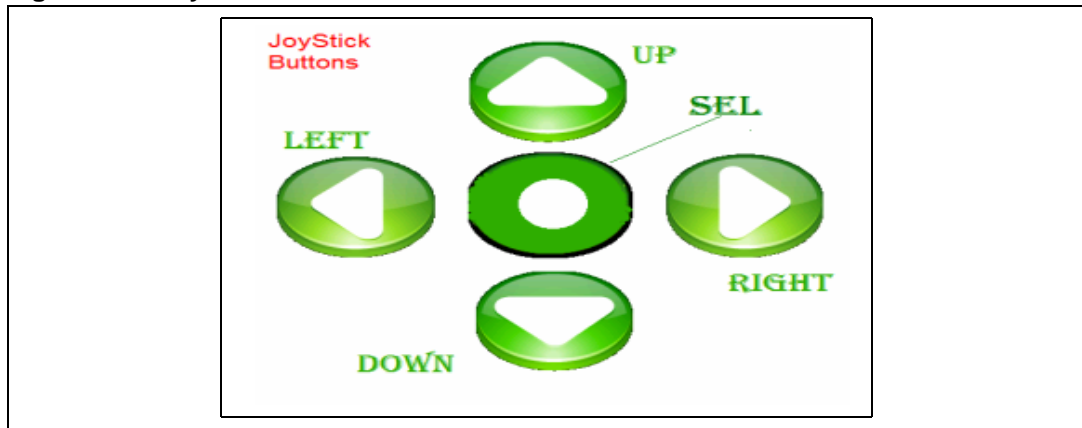
This submenu provides help on the different keys used in the STM32F10xxx demonstration. When this submenu is selected, the message shown in [Figure 52](#) is displayed on the LCD screen.

Figure 52. Help submenu



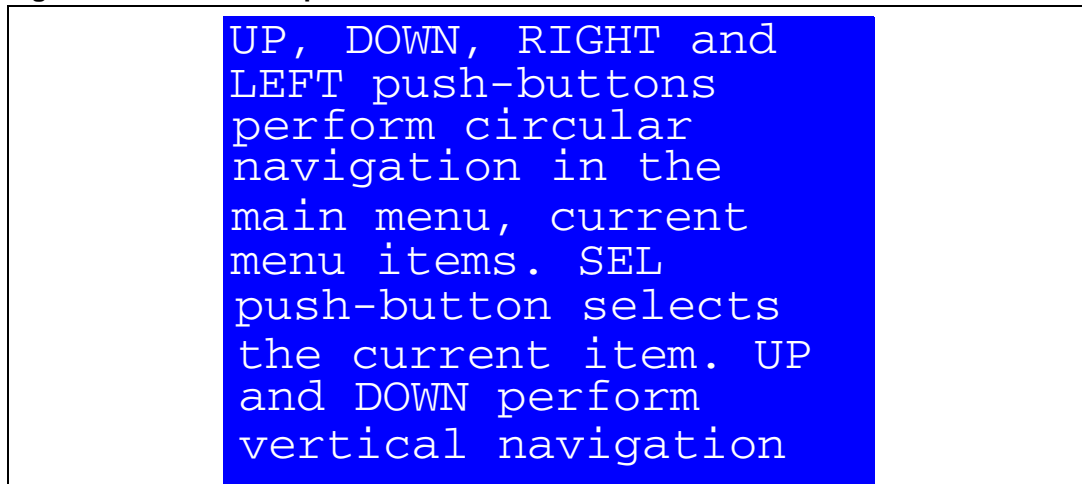
If the user presses on SEL, the image shown in [Figure 53](#) is displayed on the LCD screen.

Figure 53. Joystick buttons



Pressing on any joystick push-button will display the next help slide shown in [Figure 54](#).

Figure 54. Second Help slide

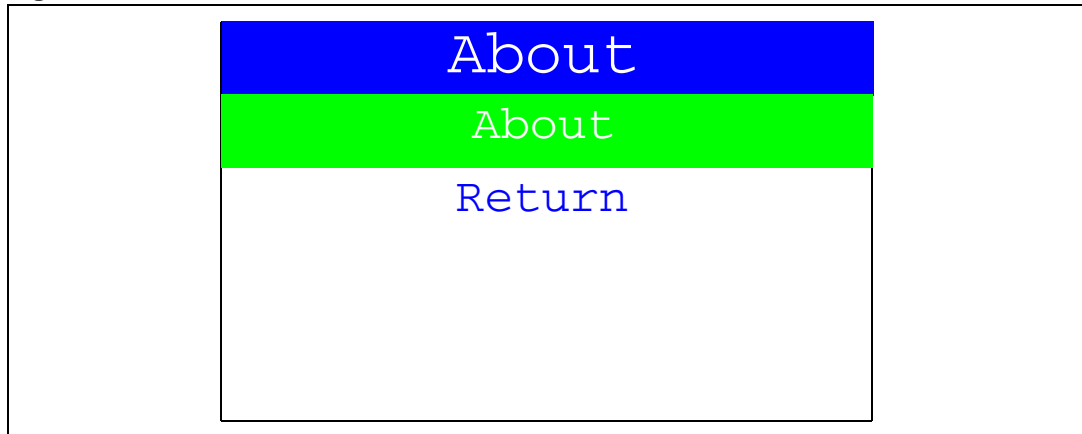


To exit the slide and return to the **Help** submenu, press SEL.

2.4.8 About submenu

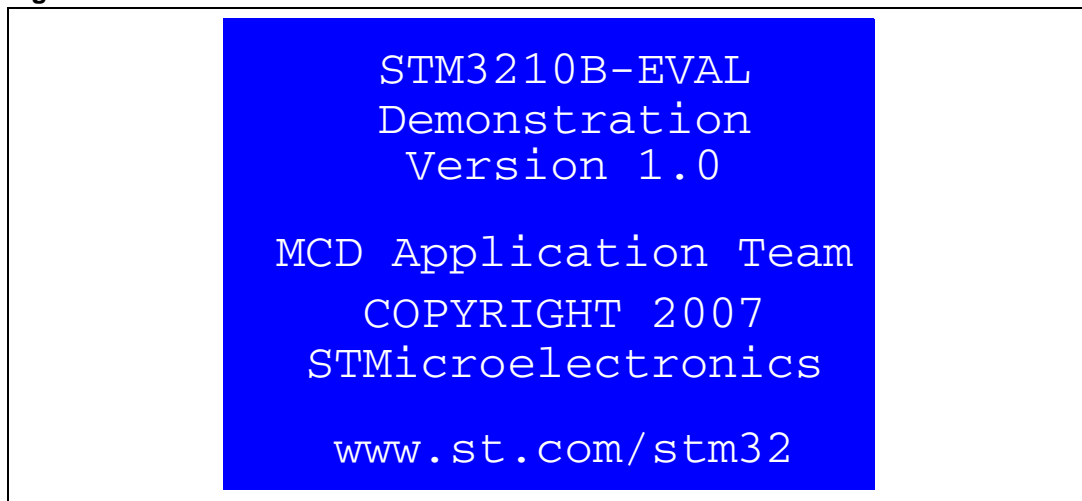
This submenu shows the version of the STM32F10xxx demonstration software. When the **About** submenu is selected, the message shown in *Figure 55* is displayed on the LCD screen.

Figure 55. About submenu



If SEL is pressed, the message shown in *Figure 56* is displayed on the LCD screen.

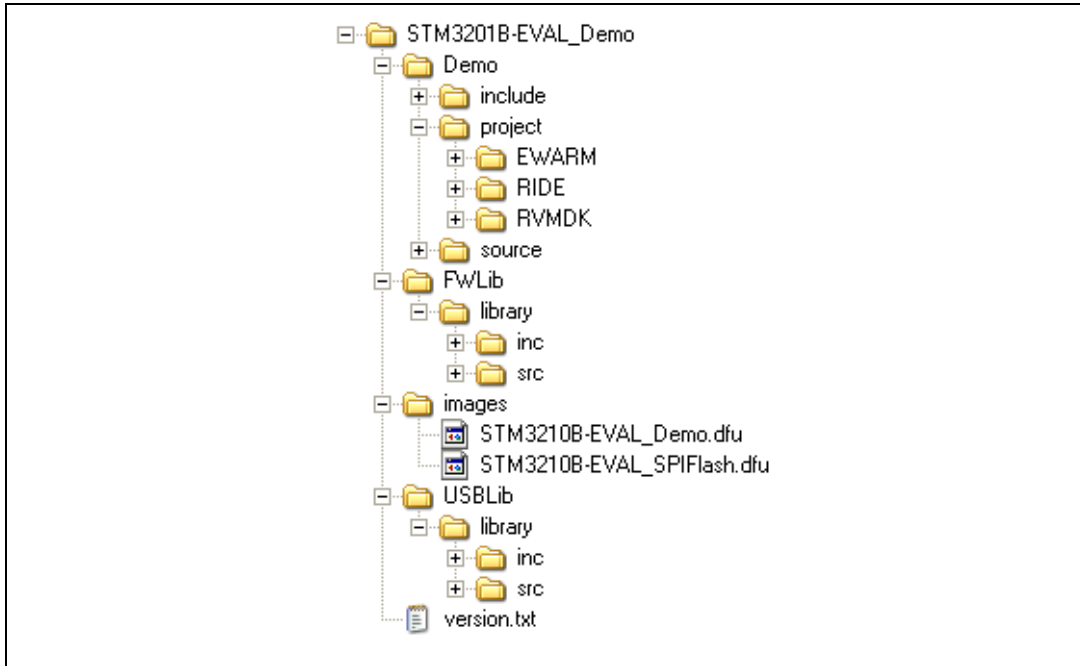
Figure 56. STM3210B-EVAL information



3 Firmware package description

The Demonstration firmware is supplied in a single zip file. The extraction of the zip file generates a folder that contains the subfolders shown in [Figure 57](#).

Figure 57. Demonstration firmware folder structure



3.1 Demo folder

The **Demo** folder contains all the subdirectories and files that make up the core of the demonstration:

- **include** subfolder contains the demonstration firmware header files
- **source** subfolder contains the demonstration firmware source files
- **project** subfolder contains three projects that compile all the demonstration files:
 - **EWARM**: contains the project for the EWARM toolchain (v4.41A and later)
 - **RIDE**: contains the project for the RIDE toolchain (v7 and later)
 - **RVMDK**: contains the project for the RVMDK toolchain (v3.05 and later)

3.2 FWLib folder

The **FWlib** folder contains all the subdirectories and files that make up the core of the STM32F10xxx Firmware library:

- **inc** subfolder contains the firmware library header files
- **src** subfolder contains the firmware library source files

3.3 images folder

The **images** folder contains two binary files that can be used with the DFU application to program the SPI Flash memory and the internal Flash memory, see [Chapter 4: Demonstration programming using the DFU on page 46](#):

- **STM3210B-EVAL_Demo.dfu** dfu file contains a binary image of the demonstration
- **STM3210B-EVAL_SPIFlash.dfu** dfu file contains all the files used by the demonstration (wav file, product presentation, icons) that must be programmed into the SPI Flash memory (mounted on the STM3210B-EVAL board).

3.4 USBLib folder

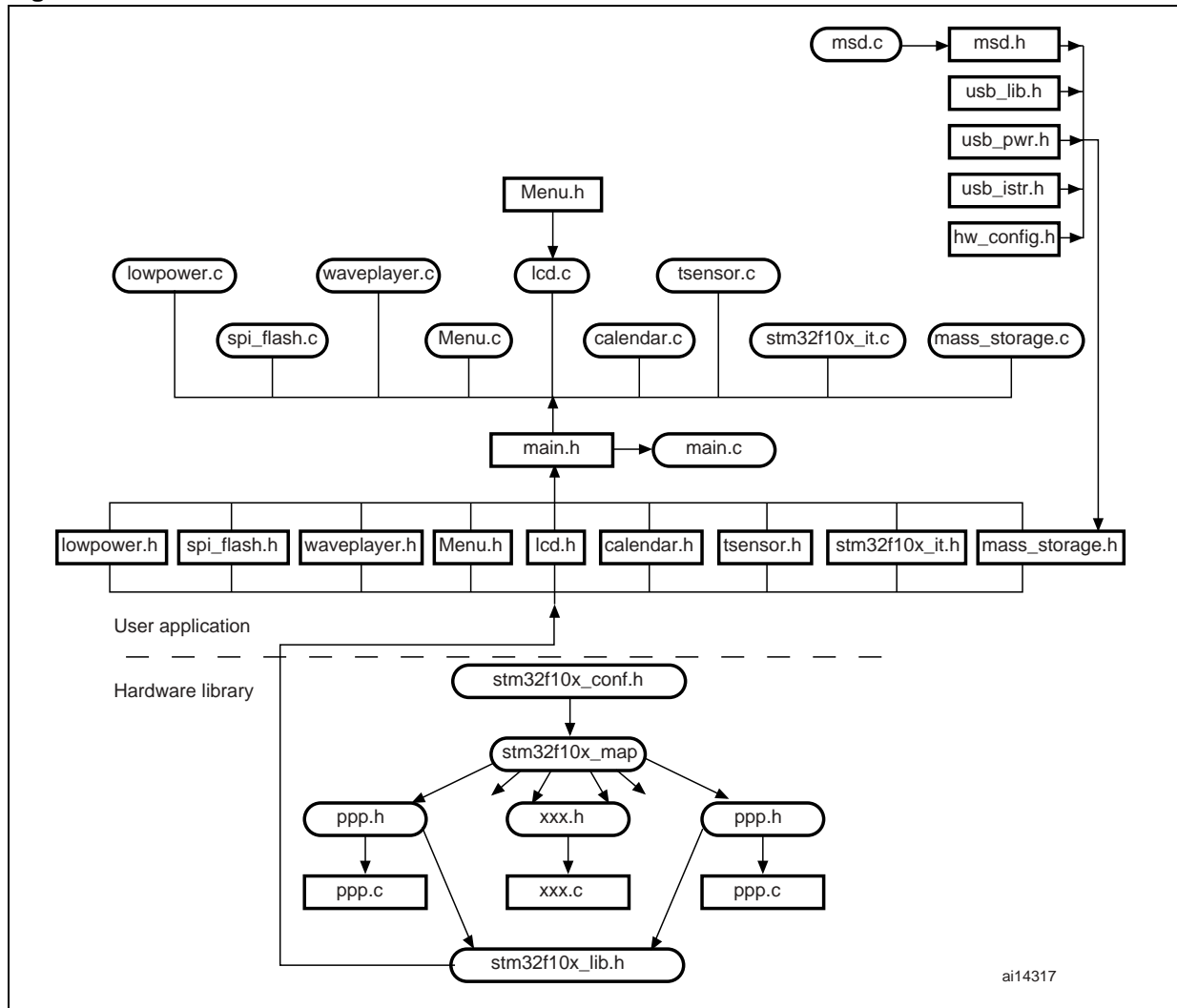
The **USBLib** folder contains all the subdirectories and files that make up the core of the STM32F10xxx USB library:

- **inc** subfolder contains the USB library header files
- **src** subfolder contains the USB library source files

3.5 Demonstration firmware architecture

The demonstration firmware architecture and file inclusion relationships are shown in [Figure 58](#).

Figure 58. Demonstration firmware architecture



4 Demonstration programming using the DFU

This section explains how to use the DFU (device firmware upgrade) application to program the demonstration application. It also describes the files needed for the STM3210B-EVAL board.

Note: When using the DFU application make sure that in the STM3210B-EVAL board, jumper JP1 is in the position 2-3.

4.1 Installing the DFU PC software

The DfuSe demonstration user interface is designed to work with all STMicroelectronics devices that support USB Device Firmware Upgrade. The installation software and the documentation are available for download from the ST website <http://www.st.com>.

Software installation

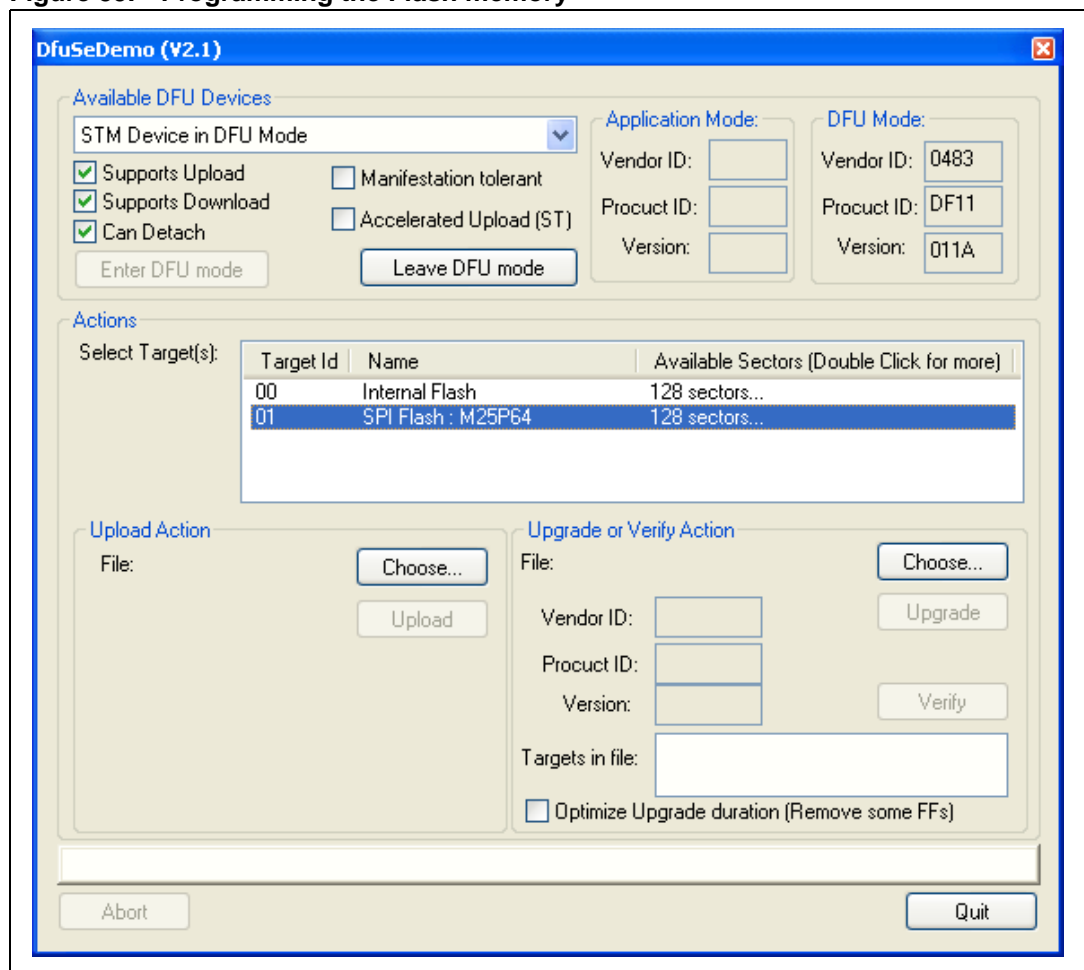
Run the *setup.exe* file: the InstallShield Wizard will guide you through the installation of the *DfuSe* applications and source code on your computer. The driver files are located in the *Driver* folder in the installation path (default: *C:\Program Files\STMicroelectronics\DfuSe*).

4.2 Programming the M25P64 Flash memory

To program the Flash memory, go through the following steps:

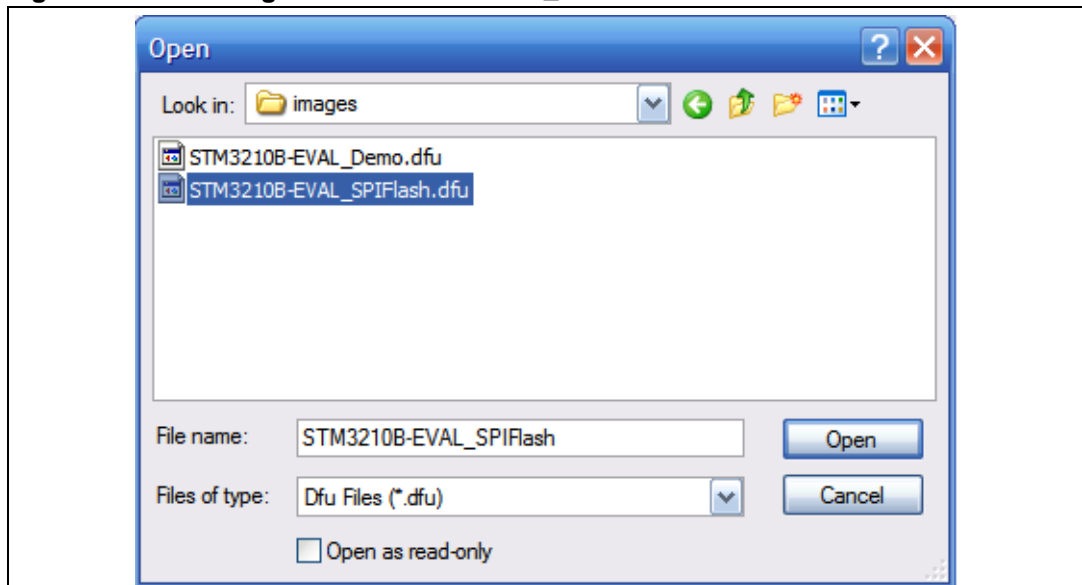
1. Load the DFU firmware project using your preferred Toolchain and load the DFU image using JTAG for the first time.
2. Connect the STM3210B-EVAL board to the PC with a USB cable. The first time this is done, you will be asked to install a driver for the board. The driver is already available in the installation directory.
Hold down the Key push-button on the STM3210B-EVAL board to enter the DFU mode.
3. Run the *DfuSeDemo.exe* program
4. In the **Action (Select Target(s))** field, select **SPI Flash: M25P64** and in the **Upgrade or Verify Action** field, click on the *Choose...* button as shown in [Figure 59](#).

Figure 59. Programming the Flash memory



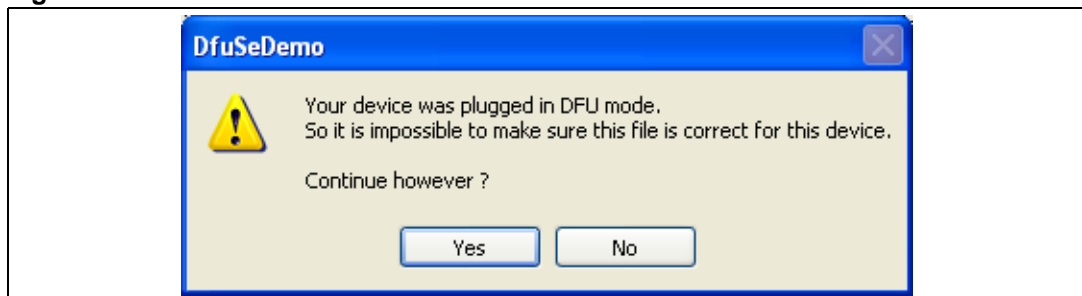
5. Select the *STM3210B-EVAL_SPIFlash.dfu* file, as shown in [Figure 60](#).

Figure 60. Selecting the *STM3210B-EVAL_SPIFlash.dfu* file



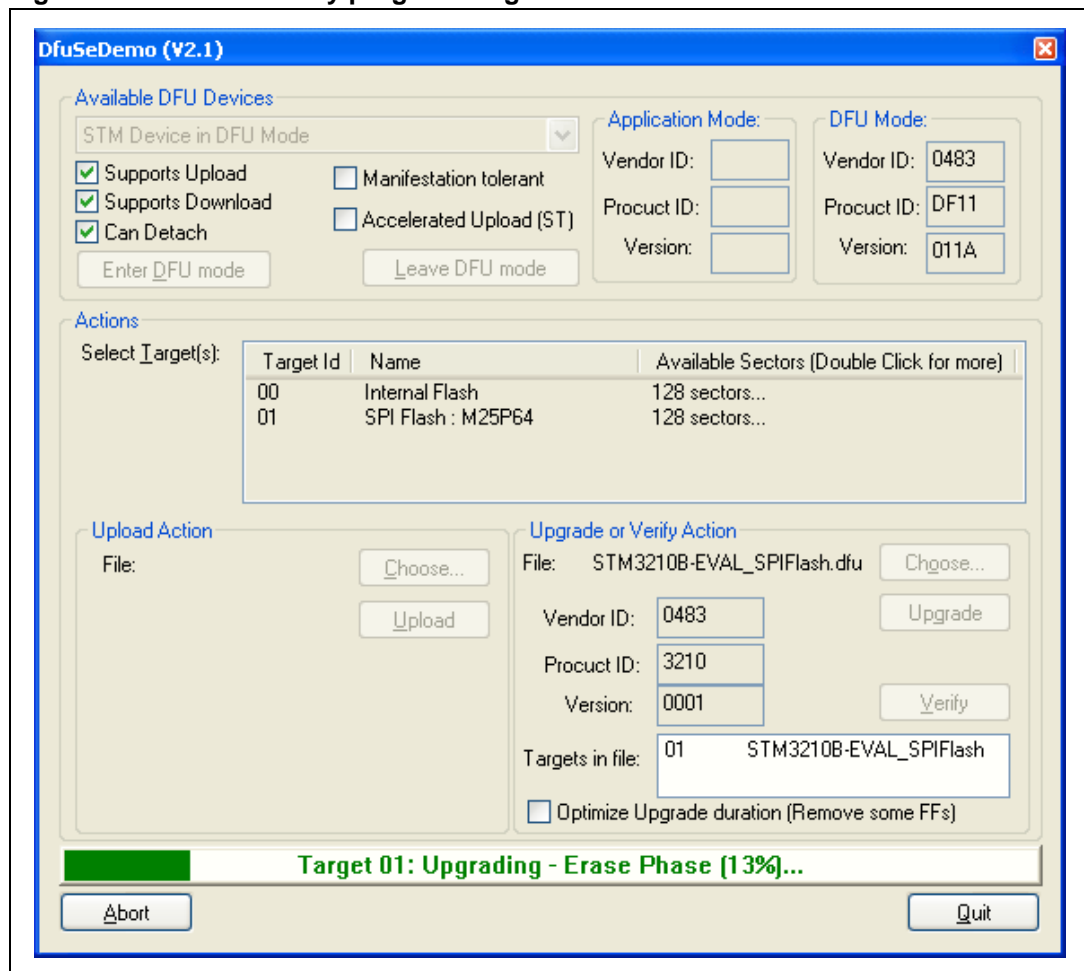
6. Then click on the Upgrade button, a dialog window will appear. Select **Yes** to continue (see [Figure 61](#)).

Figure 61. Click **Yes to continue**



7. At this step the M25P64 Flash programming will start as shown in *Figure 62*.

Figure 62. Flash memory programming

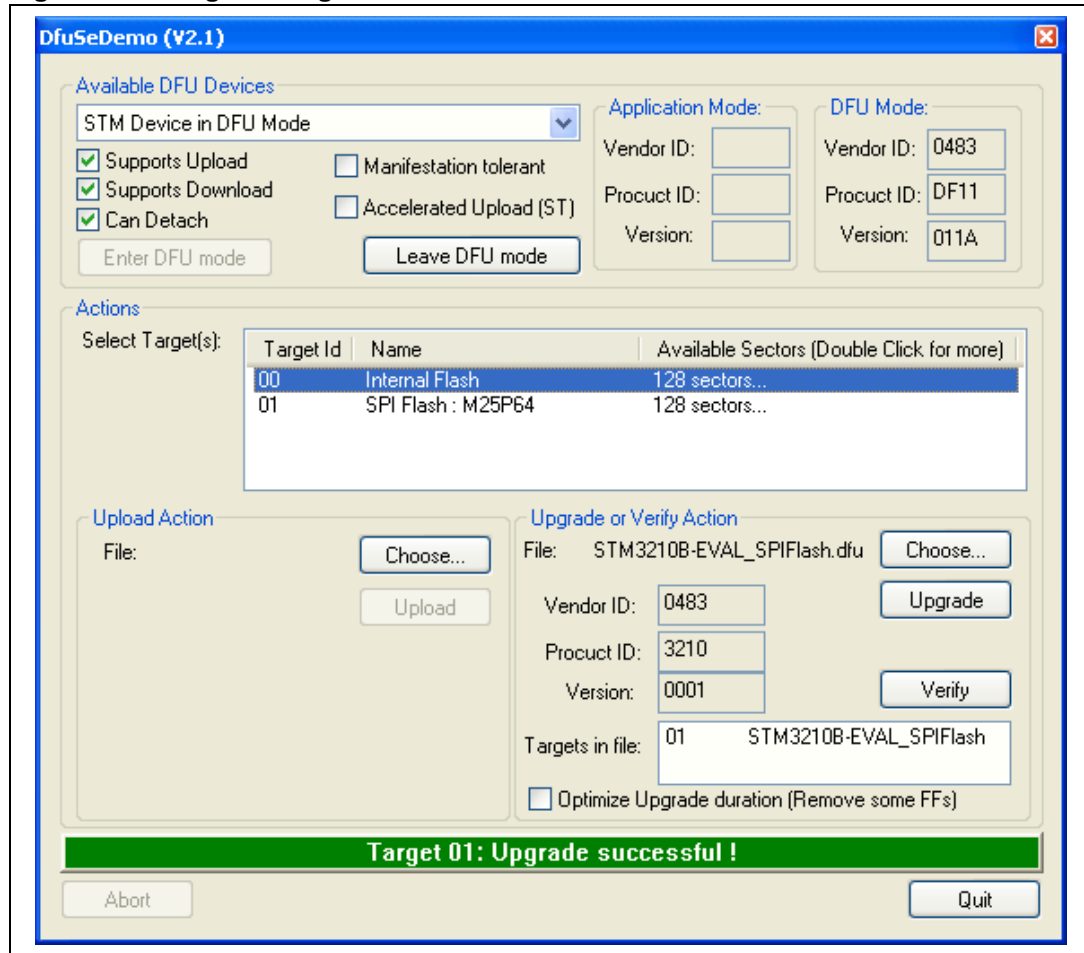


4.3 Demonstration programming

To program the internal Flash memory, go through the following steps:

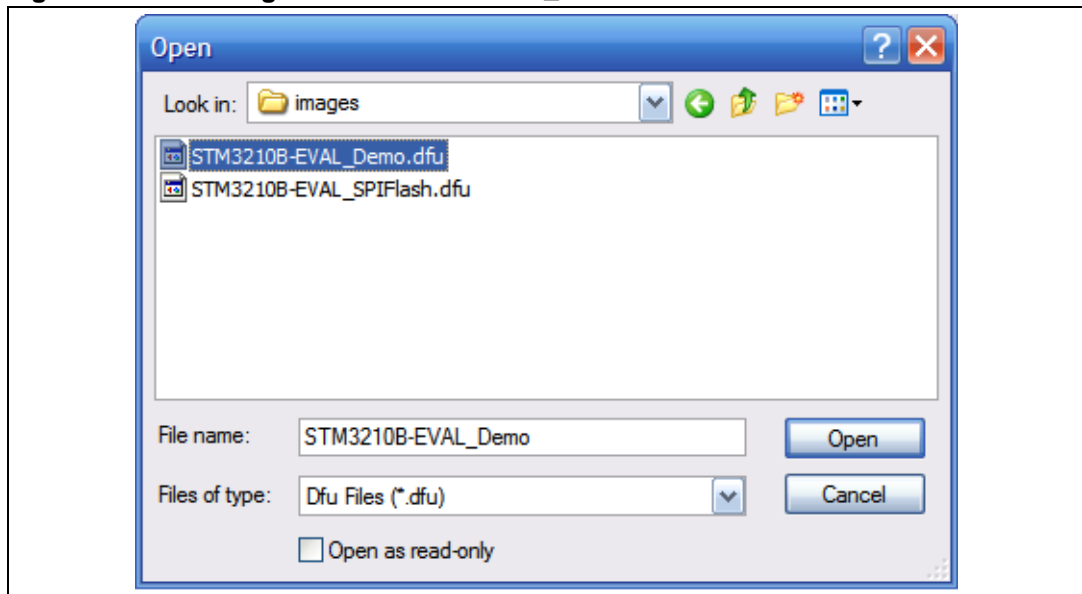
1. When the SPI Flash memory has finished programming, in the **Action (Select Target(s))** field, select *Internal Flash* and in the **Upgrade or Verify Action** field, click on the *Choose...* button.

Figure 63. Programming the demonstration



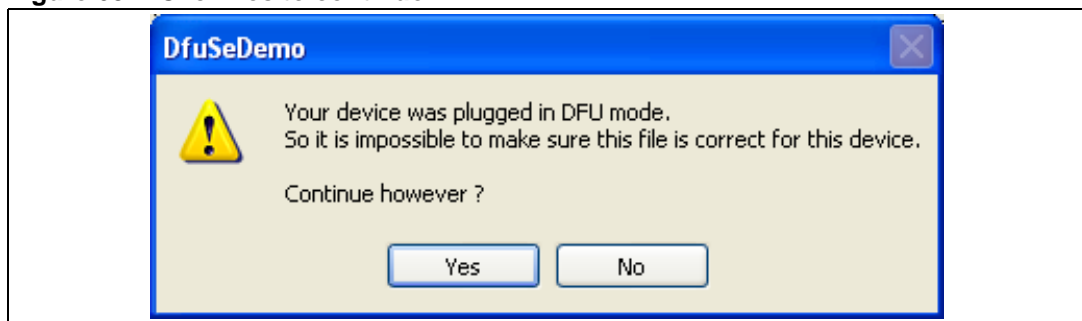
2. Select the *STM3210B-EVAL_Demo.dfu* file, as shown in [Figure 64](#).

Figure 64. Selecting the *STM3210B-EVAL_Demo.dfu* file



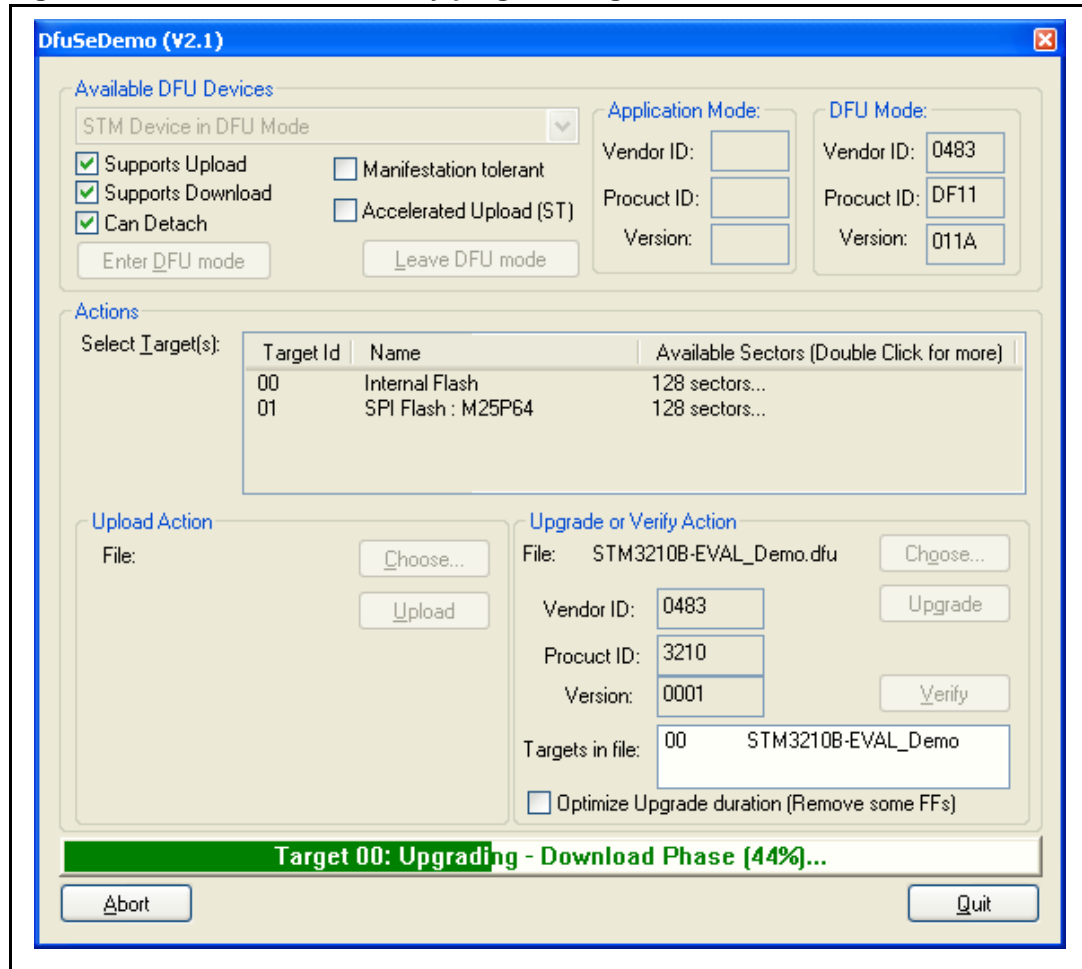
3. Then click on the *Upgrade* button, a dialog window will appear. Select *Yes* to continue.

Figure 65. Click *Yes* to continue



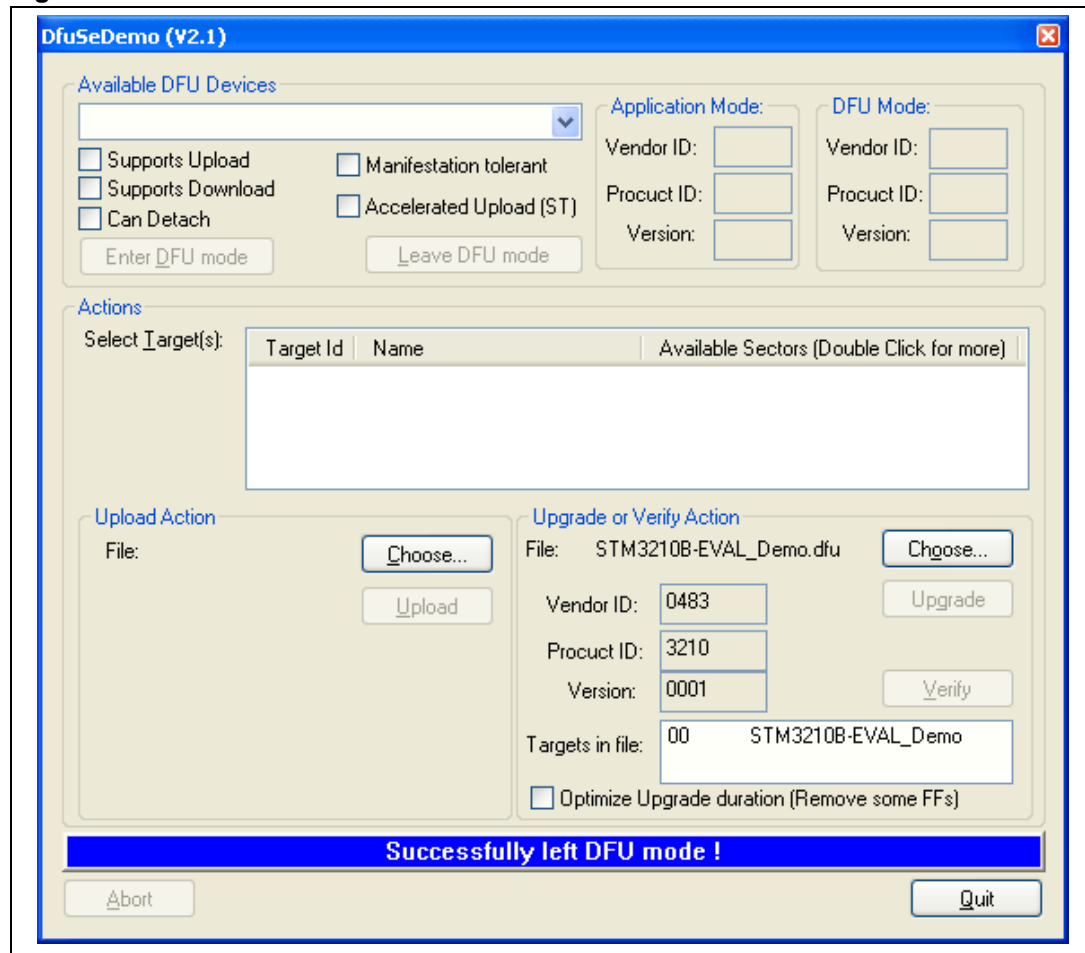
- 4. At this step the STM32F10xxx's internal Flash memory programming starts as shown in *Figure 66*.

Figure 66. Internal Flash memory programming



- Once the internal Flash programming is finished, click on *Leave DFU Mode*, This causes the target to switch from DFU mode to Application mode, as shown in *Figure 67*. And the demonstration starts executing.

Figure 67. DFU mode left



5 Revision history

Table 4. Document revision history

Date	Revision	Changes
01-Jun-2007	1	Initial release.
02-Jul-2007	2	Section 2.4.2: Calendar and Section 4.1: Installing the DFU PC software modified, Figure 59 , Figure 62 , Figure 63 , Figure 66 and Figure 67 updated. Small text changes.
05-Oct-2007	3	SEL push-button specified in Section 2.1: Menu . Section 2.1.1: Demonstration startup clarified. Date Adjust and Alarm Show descriptions updated in Date submenu on page 26 . RIDE project added in Section 3.1: Demo folder . Figure 57: Demonstration firmware folder structure updated. Section 2.4.5: USB mass storage updated. Diode behavior modified in the first cases of the Stop mode menu and Standby mode menu . Figure 56: STM3210B-EVAL information modified. Section 1.10.3: 12-bit analog-to-digital converter (ADC) modified.

Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries ("ST") reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST's terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST'S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS EXPRESSLY APPROVED IN WRITING BY AN AUTHORIZED ST REPRESENTATIVE, ST PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. ST PRODUCTS WHICH ARE NOT SPECIFIED AS "AUTOMOTIVE GRADE" MAY ONLY BE USED IN AUTOMOTIVE APPLICATIONS AT USER'S OWN RISK.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.

Information in this document supersedes and replaces all information previously supplied.

The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

© 2007 STMicroelectronics - All rights reserved

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com