



Clock configuration tool for STM32F0xx microcontrollers

Introduction

This application note presents the clock system configuration tool for the STM32F0xx microcontroller family.

The purpose of this tool is to help the user configure the microcontroller clocks, taking into consideration product parameters such as power supply and Flash access mode.

The configuration tool is implemented in the “STM32F0xx_Clock_Configuration_VX.Y.Z.xls” file which is supplied with the STM32F0xx Standard Peripherals Library and can be downloaded from www.st.com.

This tool supports the following functionalities for the STM32F0xx:

- Configuration of the system clock, HCLK source and output frequency
- Configuration of the Flash latency (number of wait states depending on the HCLK frequency)
- Setting of the PCLK1, PCLK2, TIMCLK (timer clocks) and I2SCLK frequencies
- Generation of a ready-to-use *system_stm32f0xx.c* file with all the above settings (STM32F0xx CMSIS Cortex-M0 Device Peripheral Access Layer System Source File)

The STM32F0xx_Clock_Configuration_VX.Y.Z.xls is referred to as “clock tool” throughout this document.

Before using the clock tool, it is essential to read the STM32F0xx microcontroller reference manual (RM0091). This application note is not a substitute for the reference manual.

This tool supports only the STM32F0xx devices.

For VX.Y.Z, please refer to the tool version, example V1.0.0

Contents

1	Glossary	5
2	Getting started	6
2.1	Software requirements	6
2.2	Hardware requirements	7
2.2.1	Introduction	7
2.2.2	Clock scheme for STM32F0xx microcontrollers	7
2.2.3	I2S clock generator	9
3	Tutorials	10
3.1	Wizard mode	10
3.2	Expert mode	12
4	Known limitations	14
5	Conclusion	15
6	Revision history	16

List of tables

Table 1.	Definition of terms	5
Table 2.	Document revision history	16

List of figures

Figure 1.	Clock scheme	8
Figure 2.	I2S clock generator architecture	9
Figure 3.	Wizard mode user interface	10
Figure 4.	Select the clock source	11
Figure 5.	File generation error	11
Figure 6.	Expert mode user interface	12
Figure 7.	System clock frequency is exceeded	13

1 Glossary

Table 1. Definition of terms

Term	Description
HCLK	AHB clock
PCLK1	APB1 clock
PCLK2	APB2 clock
TIMCLK	Timer clock
F _{CPU}	Cortex-M0 clock
Ext.Clock	External clock
V _{DD}	Power supply
HSI	High-speed internal clock
HSE	High-speed external clock
MCLK	Master clock
I2S	Integrated interchip sound
F _s	Sampling frequency
I2SCLK	I2S clock

2 Getting started

This section describes the requirements and procedures needed to start using the clock tool.

2.1 Software requirements

To use the clock tool with Windows[®] operating system, a recent version of Windows, such as Windows XP, Vista or Windows 7 must be installed on the PC with at least 256 Mbytes of RAM.

Before starting to use the clock tool, make sure that Microsoft[®] Office is installed on your machine and then follow these steps:

- Download the latest version of the **clock tool** for the STM32F0xx product from www.st.com.
- Enable macros and ActiveX[®] controls:

Excel[®] 1997-2003 version

1. Click **Tools** in the menu bar.
2. Click **Macro**.
3. Click **Security**.
4. Click **Low (not recommended)**.

Note: If ActiveX controls are not enabled, a warning message is displayed asking you to enable ActiveX. In this case, you should click "OK" to enable it.

Excel 2007 version

1. Click the **Microsoft Office** button and then click **Excel options**.
2. Click **Trust Center**, click **Trust center settings**, and then click **Macro settings**.
3. Click **Enable all macros (not recommended, potentially dangerous code can run)**.
4. Click **Trust Center**, click **Trust center settings**, and then click **ActiveX settings**.
5. Click **Enable all controls without restrictions and without prompting (not recommended; potentiality dangerous controls can run)**.
6. Click **OK**.

Note: For more information about how to enable macros and ActiveX controls please refer to the Microsoft Office website.

2.2 Hardware requirements

2.2.1 Introduction

The clock tool is designed to configure the system clocks and generate the `system_stm32f0xx.c` file for STM32F0xx microcontrollers.

The `system_stm32f0xx.c` file is provided as a template system clock configuration file which can be easily modified to select the corresponding system clock frequency and to configure the Flash latency.

2.2.2 Clock scheme for STM32F0xx microcontrollers

This section describes the system clock scheme that is dependent on the voltage requirements (V_{DD}) versus the system clock frequency and Flash latency versus the system clock frequency.

Three different clock sources can be used to drive the system clock (SYSCLK):

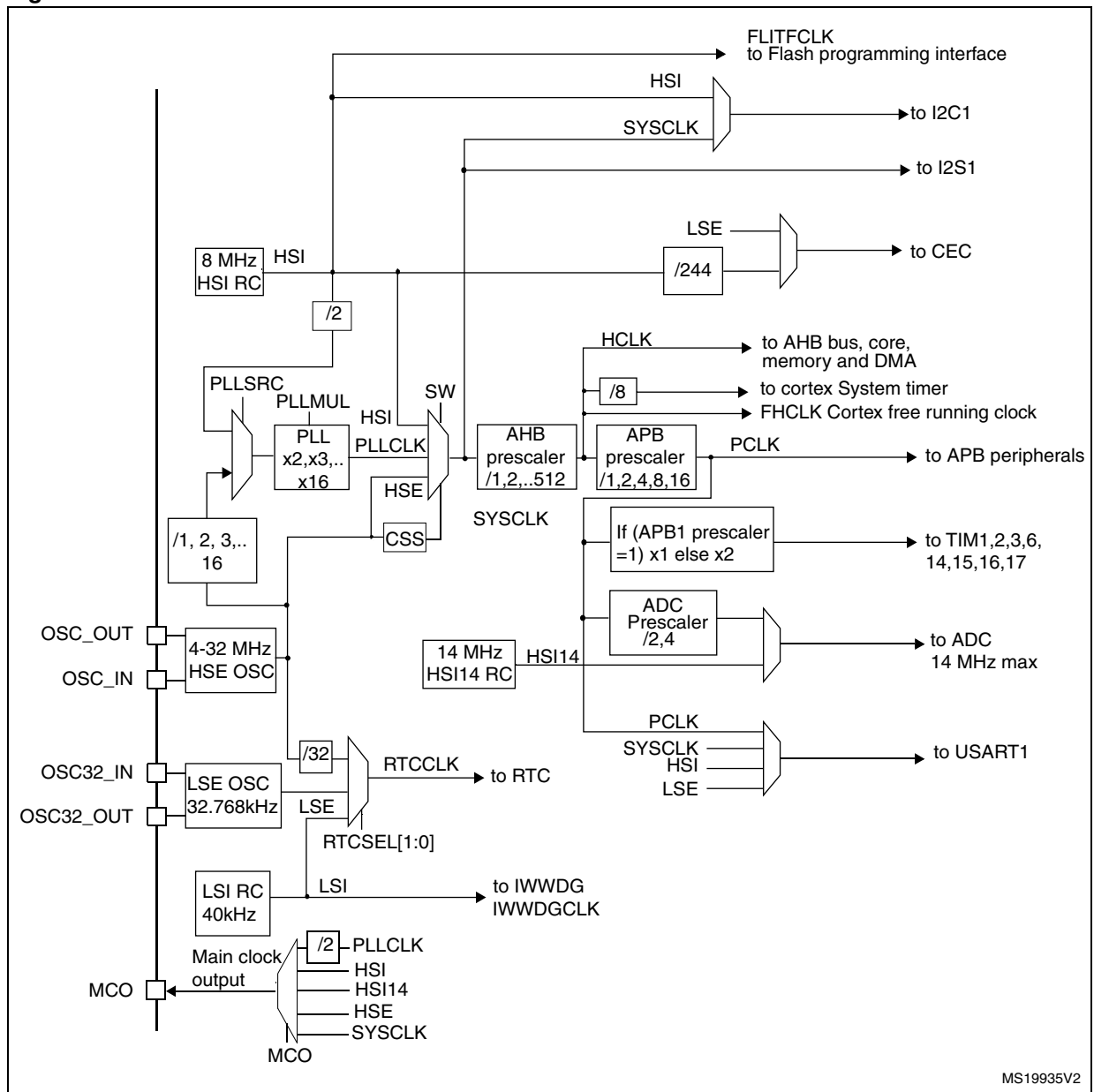
1. **HSI** (8 MHz) oscillator clock
2. **HSE** (4 MHz to 32 MHz) oscillator clock
3. **Main phase-locked loop (PLL)** clock with a PLL voltage-controlled oscillator (PLLVCO) input frequency.

All peripheral clocks are derived from the SYSCLK.

Note: The number of Flash memory wait states (latency) is defined according to the frequency of the CPU (Cortex-M0):

- Zero wait states if $0 < \text{SYSCLK} \leq 24 \text{ MHz}$
- One wait state if $\text{SYSCLK} > 24 \text{ MHz}$

Figure 1. Clock scheme



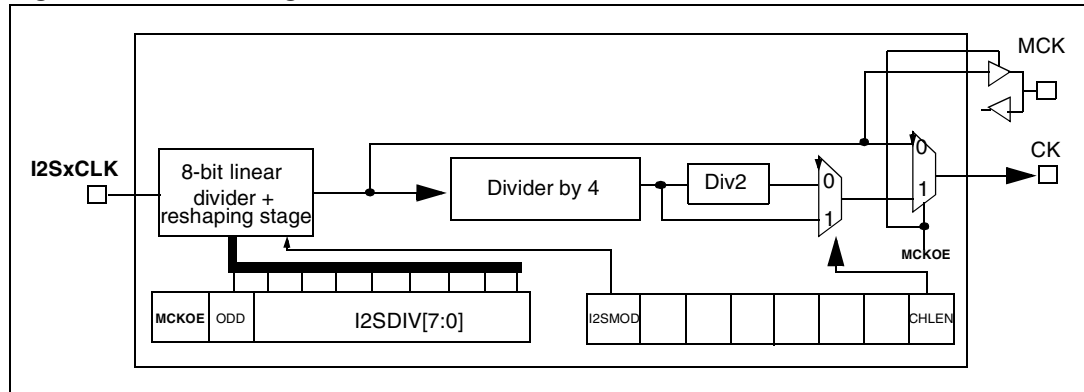
MS19935V2

2.2.3 I2S clock generator

This section describes the I2S clock generator. It is dependent on:

- Master clock MCLK (enable or disable)
- Frame width
- I2S peripheral clock (I2SCLK).

Figure 2. I2S clock generator architecture



The audio sampling frequency may be 192 kHz, 96 kHz, 48 kHz, 44.1 kHz, 32 kHz, 22.05 kHz, 16 kHz, 11.025 kHz or 8 kHz. To reach the desired frequency, the linear divider (DIV) needs to be programmed according to the formulas below:

When the master clock is generated (MCKOE bit in the SPI_I2SPR register is set):

- $FS = I2SxCLK / [(16 \cdot 2) \cdot ((2 \cdot I2SDIV) + ODD) \cdot 8]$ when the channel frame is 16 bits wide
 - $FS = I2SxCLK / [(32 \cdot 2) \cdot ((2 \cdot I2SDIV) + ODD) \cdot 4]$ when the channel frame is 32 bits wide
- Where ODD is an odd factor for the prescaler.

When the master clock is disabled (MCKOE bit cleared):

- $FS = I2SxCLK / [(16 \cdot 2) \cdot ((2 \cdot I2SDIV) + ODD)]$ when the channel frame is 16 bits wide
- $FS = I2SxCLK / [(32 \cdot 2) \cdot ((2 \cdot I2SDIV) + ODD)]$ when the channel frame is 32 bits wide

Note: This tool does not configure the I2S register.

The sampling frequency error is computed as an indicator according to the I2S parameters which are not configured in the output file "system_stm32f0xx.c".

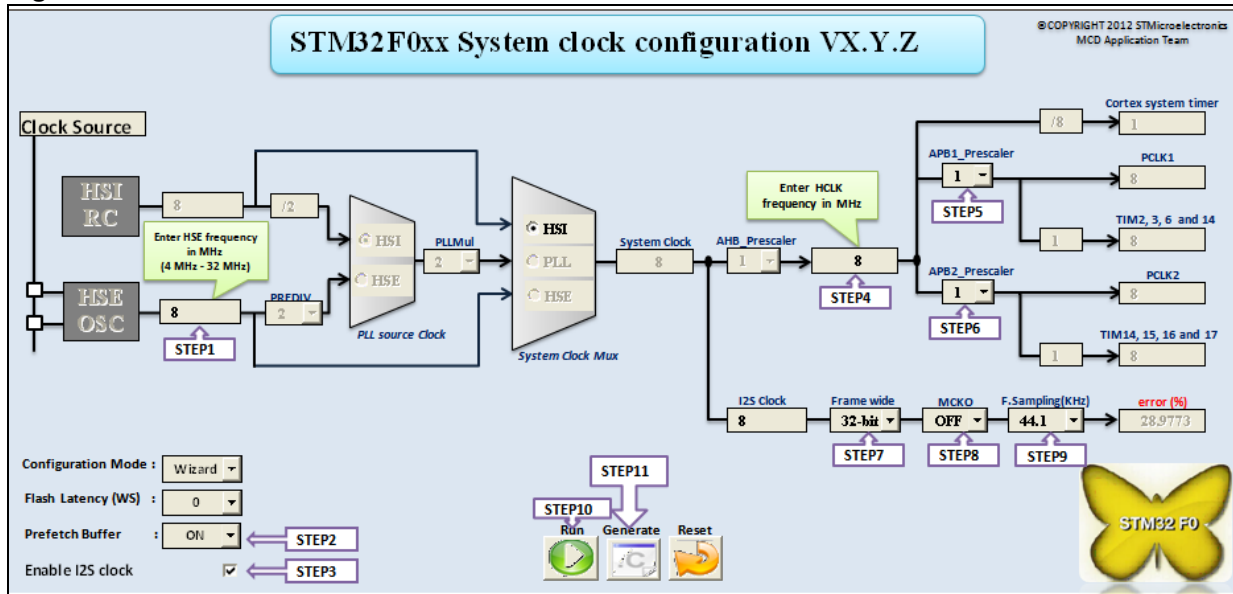
3 Tutorials

This section describes how to use the clock tool to configure all system clocks and generate the *system_stm32f0xx.c* file. Two modes are available: **Wizard** and **Expert**. The selection is made in the **Configuration mode** list box.

3.1 Wizard mode

This mode (default mode) guides you through a series of steps to obtain the desired clock system configuration quickly and easily.

Figure 3. Wizard mode user interface



Note: The **Reset** button permits the system clock for the default configuration to be set.

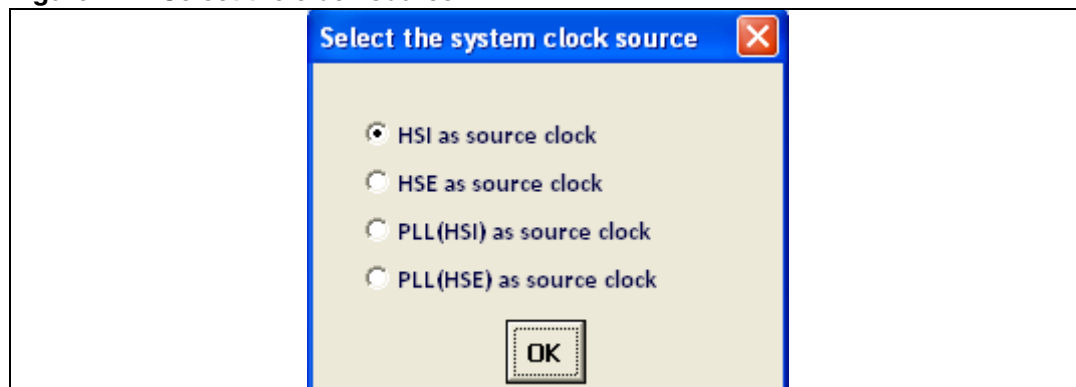
The wizard guides you through the following steps:

1. **Set the HSE frequency** (if is used in your application) between a minimum of 4 MHz, and a maximum of 32 MHz if a crystal oscillator is used for the STM32F0xx. If the frequency entered is out of range, an error message is displayed, and a valid frequency must be entered.
The definition of HSE_VALUE in the *stm32f0xx.h* file must be modified each time the user changes the HSE oscillator value.
2. Configure the **Prefetch buffer** (select ON or OFF from the list box).
3. Specify if the I2S clock is needed. If needed, enable it and follow steps 7, 8 and 9. Otherwise, go to step 4.
4. Set the desired HCLK frequency. If the value entered is higher than the maximum HCLK frequency, an error message is displayed.
5. Select the PCLK1 and PCLK2 prescaler settings from the list box to obtain the desired PCLK1 and PCLK2 frequencies. The TIMCLK frequencies are configured automatically depending on the PCLK1 and PCLK2 prescaler settings.

Note: In this product PCLK1 and PCLK2 share the some clock signal, so APB1 prescaler should always equal APB2 prescaler.

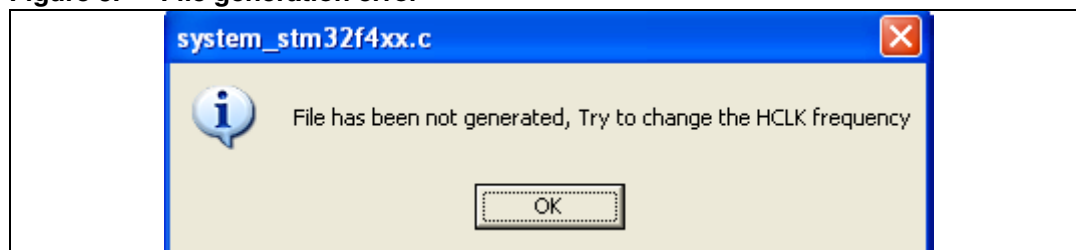
6. If the I2S clock is needed, select the frame width (16 or 32 bits).
7. Specify if the master clock is enabled or disabled (Select ON/OFF from the list box).
8. Select the Frequency from the list box. The Fs value can be 192 kHz, 96 kHz, 48 kHz, 44.1 kHz, 32 kHz, 22.05 kHz, 16 kHz, 11.025 kHz, or 8 kHz.
9. Click the **RUN** button.
If more than one clock source is possible, a message box displays the clock sources that can be selected (see [Figure 4](#)). Choose HSE, HSI or PLL (which are sourced by the HSI or HSE).

Figure 4. Select the clock source



10. Click the **Generate** button to automatically generate *system_stm32f0xx.c* file.
The *system_stm32f0xx.c* file is generated in the same location as the clock tool. Display the file to verify the value of the system clock, *SystemCoreClock*, and the values of HCLK, PCLK1, PCLK2, Flash access mode, and other parameters which are defined in the *SetSysClock* function.
If the file is not generated, an error message is displayed as shown [Figure 5](#).

Figure 5. File generation error

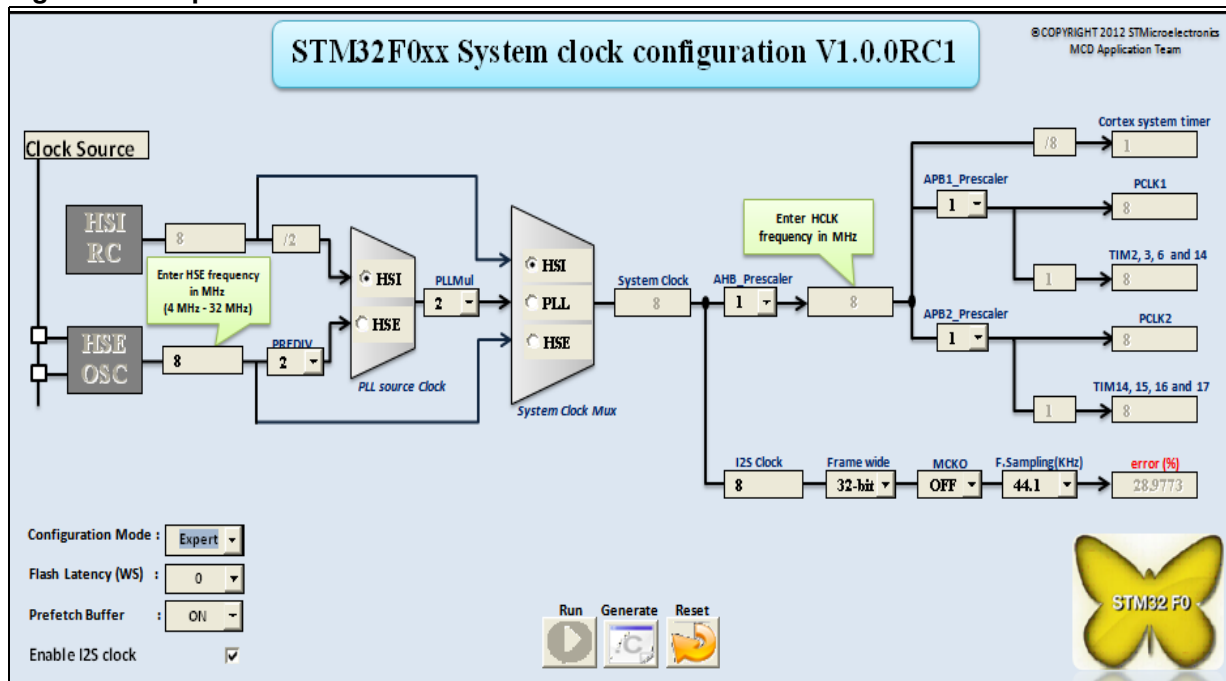


11. The *system_stm32f0xx.c* file must be added to the working project to be built.

3.2 Expert mode

This mode provides more flexibility regarding the configuration setup but the user must ensure that the configuration is correct.

Figure 6. Expert mode user interface



The **'View'** button permits the .xls file to be viewed in full screen, to be activated or deactivated.

The **'Reset'** button permits the system clock to be reset to the default configuration.

These main steps are described in detail in this section:

1. Configure the SYSCLK frequency.
2. If required, enable the I2S clock and configure the I2S clock frequency.
3. If required, configure the Prefetch buffer.
4. Generate the *system_stm32f0xx.c* file.
5. Add the *system_stm32f0xx.c* file to the working project to be built.

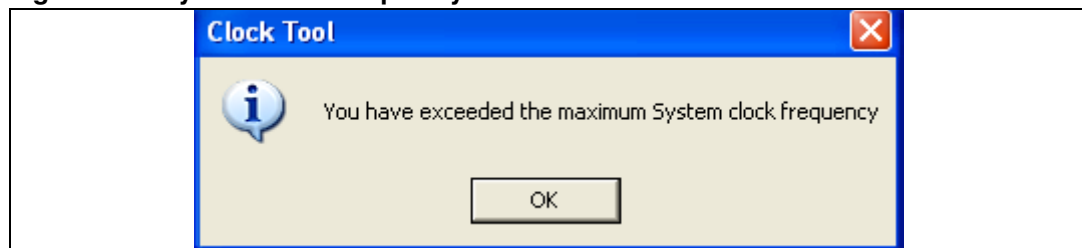
1. Configure the SYSCLK frequency.

- a) If the HSE is used in your application, set its frequency to between 4 MHz and 32 MHz (set it to 32 MHz if a crystal oscillator is used for the STM32F0xx. If the frequency entered is out of range, an error message is displayed. A valid frequency must be entered.

Note: The definition of `HSE_VALUE` in the `stm32f0xx.h` file must be modified each time the user changes the HSE oscillator value.

- b) Configure the SYSCLK source (PLL, HSE or HSI). If the clock source selection is invalid (HCLK frequency is too high) the error message in [Figure 7](#) is displayed.

Figure 7. System clock frequency is exceeded



- c) If PLL is selected as the SYSCLK source, it is necessary to select the source clock for the PLL (HSE or HSI).
- d) If PLL is selected as the SYSCLK source, configure the main PLL (PLLMUL) and the PLL division factor (PREDIV) if the HSE is selected as PLL clock source.
- e) Set HCLK prescaler using the **AHBPrescaler** list box to obtain the desired HCLK frequency.
- f) Select PCLK1 prescaler settings from the list box to obtain the desired PCLK1 frequency. The TIMCLK frequencies are configured automatically depending on the PCLK1 prescaler settings.
- g) Select PCLK2 prescaler settings from the list box to obtain the desired PCLK2 frequency. The TIMCLK frequencies are configured automatically depending on the PCLK2 prescaler settings.

Note: In this product the PCLK1 and PCLK2 share the some clock signal, so APB1 prescaler should always equal APB2 prescaler.

- h) Configure the Flash Latency: after setting the HCLK prescaler, the number of Flash wait states is configured automatically with the best value (*lowest* possible value) which can be modified to any value higher than the best value.
- i) Generate the clock configuration files by clicking on the **Generate** button.

2. If required, enable the I2S clock and configure the I2S clock frequency.

- a) Select frame width (16 or 32 bits) and specify if the master clock is enabled or not.
- b) Select the Fs from the list box. The Fs value can be 192 kHz, 96 kHz, 48 kHz, 44.1 kHz, 32 kHz, 22.05 kHz, 16 kHz, 11.025 kHz and 8 kHz.

3. Optionally configure the Prefetch buffer.

4. Generate the `system_stm32f0xx.c` file.

Click the **Generate** button to automatically generate the `system_stm32f0xx.c` file in the same location as the clock tool. It can be displayed to verify the value of the SYSCLK, *SystemCoreClock*, and the values of HCLK, PCLK1, PCLK2, Flash access mode, and other parameters which are defined in the "SetSysClock" function.

5. The `system_stm32f0xx.c` file must be added to the working project to be built.

4 Known limitations

This section describes the known limitations of the clock configuration tool.

This tool does not support configurations that use the HSE external clock source (HSE bypass).

5 Conclusion

This application note provides a description of how to use the clock tool with the STM32F0xx microcontroller devices.

This tool generates a source code file *system_stm32f0xx.c* to configure the clock system of the STM32F0xx. It can be accessed from either of the two configuration modes:

- Wizard mode: provides a quick and easy way to configure the system clocks.
- Expert mode: offers more flexibility in setting up the system clock configuration while still respecting all the product constraints.

6 Revision history

Table 2. Document revision history

Date	Revision	Changes
03-May-2012	1	Initial release

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 TWO AUTHORIZED ST REPRESENTATIVES, 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.

© 2012 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 - Philippines - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com

