AN4646
Application note

Peripheral interconnections on STM32F401 and STM32F411 lines

Introduction

On top of the highest performance and the lowest power consumption of the STM32F4 family, STM32F401/411 peripherals can communicate autonomously, without any intervention from the CPU, via a network known as peripheral interconnect matrix.

This STM32F401/411 feature enhances CPU real-time performance while substantially reducing its energy consumption.

The present document first describes the peripheral interconnect matrix features and then provides an overview on the peripheral interconnections and on how to configure them depending on your application. This description is completed by a detailed application example.

This application note must be read in conjunction with STM32F401 line reference manual (RM0383) and STM32F411 line reference manual (RM0368). Both documents are available from http://www.st.com/stm32.

Table 1. Applicable products

Type	Product lines
Microcontrollers	STM32F401 STM32F411

Contents

List of tables

List of figures

1 Module overview

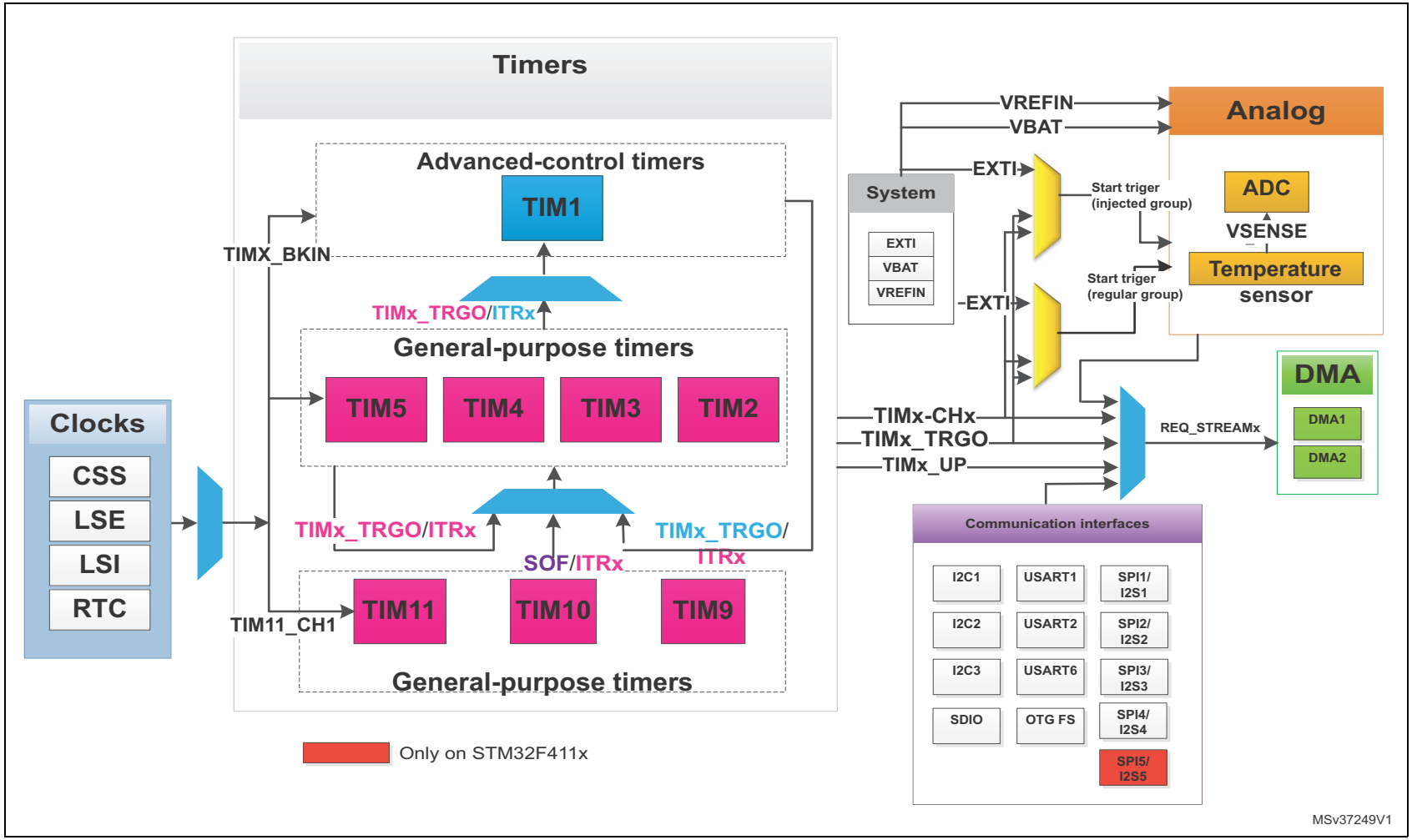
Several peripherals can be directly interconnected and configured to send/respond to event signals that can be internally routed to/from other peripherals in the device.

The STM32F401/411 autonomous peripherals include:

- **Timers** that can either be internally connected to each other or connected to the DMA or to the analog block.
- **Analog block** that can receive events from a timer or send events to the DMA.
- **Clocks block** that can send events to timers.
- **System block** that can send events to the analog block.
- **Communication interfaces** that can send events to the timers or to the DMA.

An overview of STM32F401/411 peripheral interconnections is given in *Figure 1: STM32F401/411 peripheral interconnection overview*.

Figure 1. STM32F401/411 peripheral interconnection overview



2 Peripheral interconnect matrix

STM32F401/411 peripherals are interconnected through the peripheral interconnect matrix that allows to directly connect one peripheral to another without waking up the CPU.

These interconnections can operate in Run, Sleep or Stop modes, depending on peripherals.

Peripherals that respond to events are referred to as **users** while peripherals that send events are referred to as **generators**.

Table 2. STM32F401/411 peripheral interconnect matrix(1)

		Users								
Generators		Timers						DMA		Analog
		TIM 1	TIM 2	TIM 3	TIM 4	TIM 5	TIM 11	DMA1	DMA2	ADC1
Timers	TIM1	-	X	X	X	-	-	-	X	X
	TIM2	X	-	X	X	X	-	X	-	X
	TIM3	X	X	-	X	X	-	X	-	X
	TIM4	X	X	X	-	X	-	X	-	X
	TIM5	X	-	X	-	-	-	X	-	X
Analog	ADC1	-	-	-	-	-	-	-	X	-
	VSENSE	-	-	-	-	-	-	-	-	X
Clocks	LSI	-	-	-	-	X	-	-	-	-
	LSE	-	-	-	-	X	-	-	-	-
	RTC	-	-	-	-	X	X	-	-	-
	CSS	X	-	-	-	-	-	-	-	-

Table 2. STM32F401/411 peripheral interconnect matrix(1) (continued)

Generators		Users								
		Timers						DMA		Analog
		TIM 1	TIM 2	TIM 3	TIM 4	TIM 5	TIM 11	DMA1	DMA2	ADC1
Communication interfaces	OTG FS	-	X	-	-	-	-	-	-	-
	SPI1/I2S1	-	-	-	-	-	-	-	X	-
	SPI2/I2S2	-	-	-	-	-	-	X	-	-
	SPI3/I2S3	-	-	-	-	-	-	X	-	-
	SPI4/I2S4	-	-	-	-	-	-	-	X	-
	SPI5/I2S5	-	-	-	-	-	-	-	X	-
	I2C1	-	-	-	-	-	-	X	-	-
	I2C2	-	-	-	-	-	-	X	-	-
	I2C3	-	-	-	-	-	-	X	-	-
	USART1	-	-	-	-	-	-	-	X	-
	USART2	-	-	-	-	-	-	X	-	-
	USART6	-	-	-	-	-	-	-	X	-
	SDIO	-	-	-	-	-	-	-	X	-
System	VBAT	-	-	-	-	-	-	-	-	X
	VREFINT	-	-	-	-	-	-	-	-	X
	EXTI	-	-	-	-	-	-	-	-	X

1. In gray, the peripherals that are available only on STM32F411 line.

2.1 Timers block

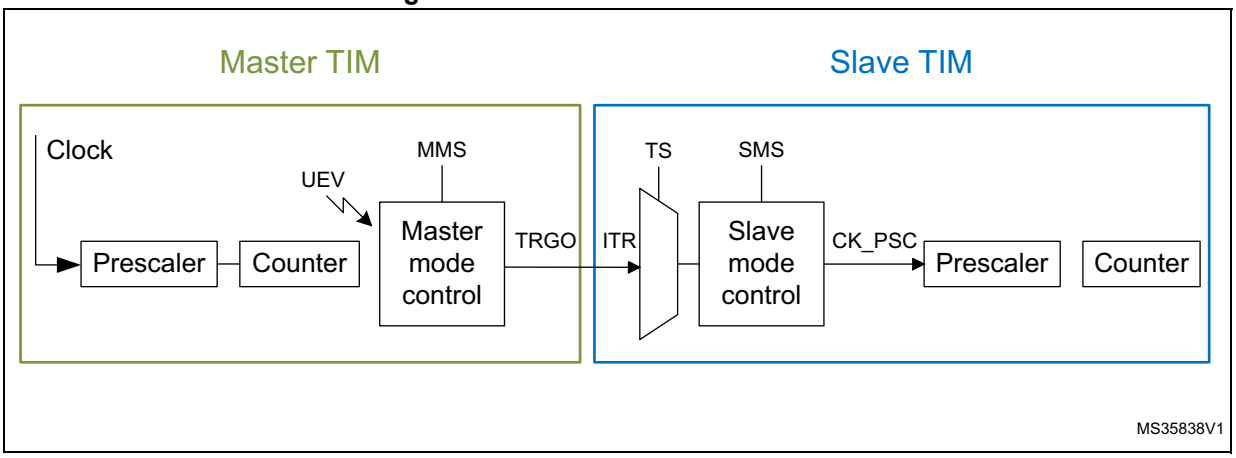
2.1.1 From TIM to TIM

Some timers are linked together internally for timer synchronization or chaining. When one timer (TIMx) is configured in Master mode, it can reset, start, stop or clock the counter of any other timer configured in Slave mode (TIMy).

The master output is TIMx_TRGO signal. This output is configured as a timer event through TIMx_CR2 register. It is sent to TIMy_ITR0/ITR1/ITR2/ITR3 inputs.

Figure 2: Master/Slave timer overview gives an overview of the trigger selection and the master mode selection blocks.

Figure 2. Master/Slave timer overview



A description of this feature is provided in the *Timer synchronization* section on RM0368 and RM0383 reference manuals, while all the possible master/slave connections are described in the *TIMx internal trigger connection* tables.

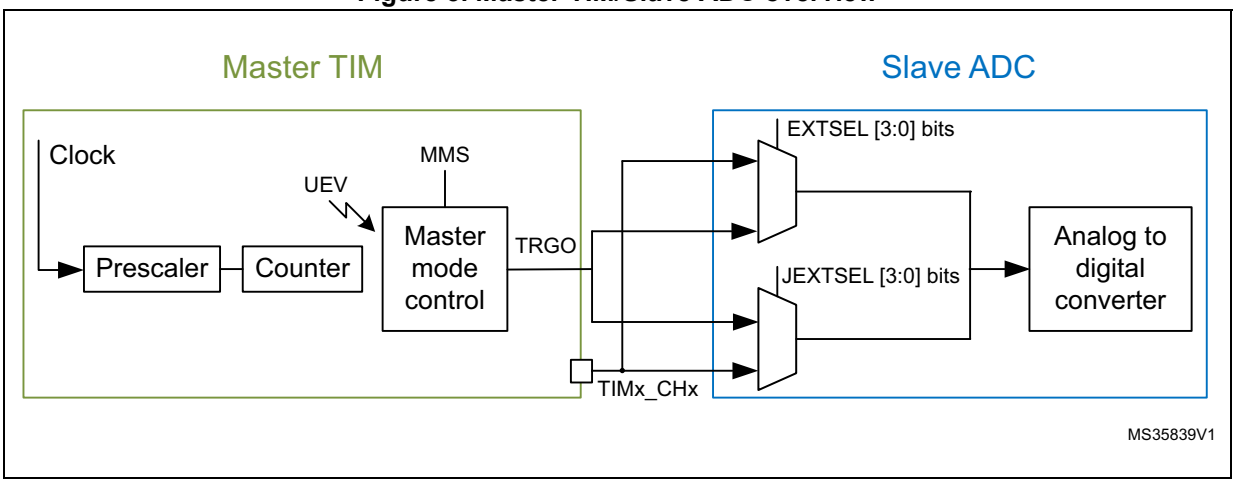
2.1.2 From TIM to ADC

Some timers can be used to generate an ADC trigger event.

The timer output can be either TIMx_TRGO signal or TIMx_CHx event. It is input to ADC EXTSEL[3:0] and JEXTSEL [3:0] signals.

Figure 3: Master TIM/Slave ADC overview gives an overview of the trigger selection and master mode selection blocks.

Figure 3. Master TIM/Slave ADC overview



ADC synchronization is described in the *Conversion on external trigger and trigger polarity* section of RM0368 and RM0383 reference manuals. The connection between timers and ADCs regular and injected channels is described in *External trigger for regular channels* and *External trigger for injected channels* tables.

2.1.3 From TIM to DMA

Refer to *Section 2.6: DMA block*.

2.2 Analog block

The analog block includes:

- the ADC block (ADC1)
- the temperature sensor block.

2.2.1 From temperature sensor to ADC1

On STM32F401/411 devices, the temperature sensor is internally connected to ADC1_IN18 input channel. It is used to convert the sensor output voltage into a digital value.

The *Temperature sensor* reference manual section describes the connection between sensor and ADC as well as the procedure to read the converted value.

2.2.2 From Analog block to DMA

Refer to *Section 2.6: DMA block*.

2.3 Clocks block

The clock block includes:

- the LSE clock
- the LSI clock
- the clock security system (CSS)
- the real-time clock (RTC)

2.3.1 From CSS to TIM

The CSS can generate system errors. In this case a clock failure event is sent to TIM1 break inputs.

The purpose of the break function is to protect the power switches driven by PWM signals generated by the timer.

The list of possible break sources is given in the *Using the break function (TIM1)* reference manual section.

2.3.2 From LSE, LSI, RTC to TIM

External clock (LSE), internal clock (LSI) and RTC wakeup interrupt can be input to general-purpose timer (TIM5 channel 4/TIM11 channel 1).

This feature is described in the following sections of RM0368 and RM0383 reference manuals:

- *Internal/external clock measurement using TIM5/TIM11*
- *TIM5 option register (TIM5_OR)*
- *TIM11 option register 1 (TIM11_OR).*

2.4 System block

The system blocks include:

- Internal reference voltage (V_{REFINT})
- V_{BAT} supply voltage
- External interrupt/event controller (EXTI).

2.4.1 From VBAT, VREFINT to ADC

The V_{BAT} channel is connected to ADC1_IN18 channel. It can be converted either as an injected or as a regular channel.

V_{REFINT} is connected to ADC1_IN17.

Refer to the following reference manual sections for a description of the interconnection between V_{BAT}, V_{REFINT} and ADC:

- *Channel selection*
- *Battery charge monitoring.*

2.4.2 From EXTI to analog block

An external interrupt/event (EXTI) can be used to generate an ADC trigger event or to start DAC conversion.

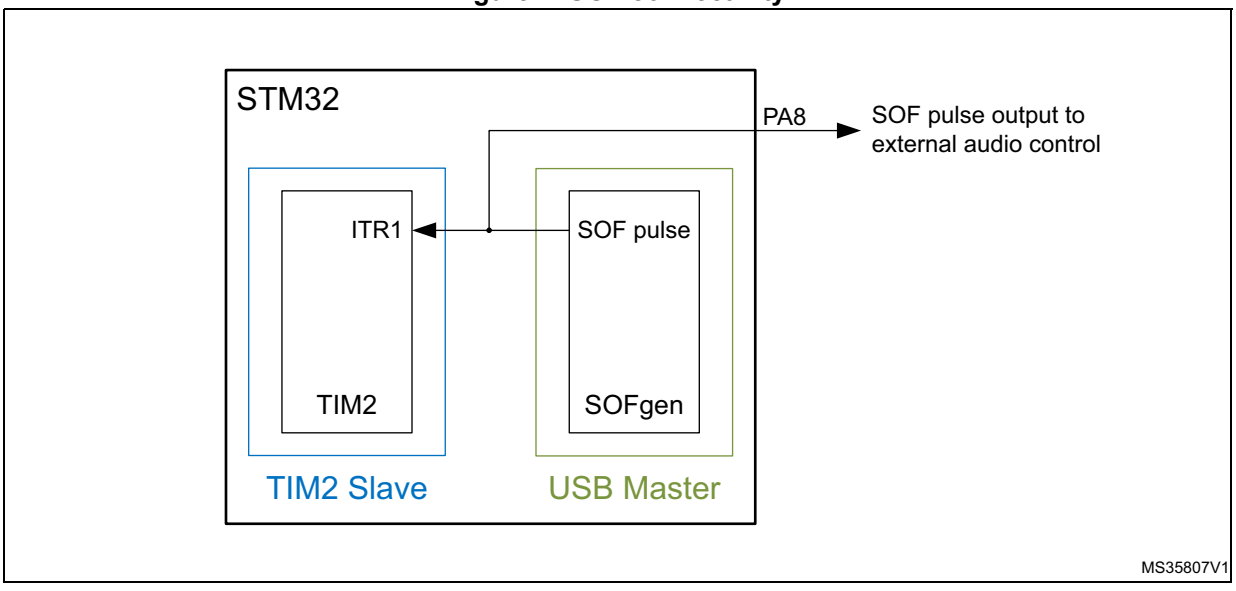
ADC synchronization is described in the *Conversion on external trigger and trigger polarity* section of the reference manuals.

2.5 Communication interfaces

2.5.1 From USB block to TIM

The USB OTG_FS SOF pulse signal can trigger TIM2 general-purpose timer.

Figure 4. SOF connectivity



The interconnection between USB and TIM2 is described in the *SOF trigger* section of RM0368 and RM0383 reference manuals.

2.5.2 From Communication interfaces to DMA

Refer to *Section 2.6: DMA block*.

2.6 DMA block

Each stream is associated with a DMA request that can be selected out of 8 possible channel requests. This selection is controlled by the CHSEL [2:0] bits of the DMA_SxCR register. The 8 requests from the peripherals (TIM, ADC, SPI, I2C, etc.) are independently connected to each channel and their connection depends on the product implementation.

This interconnection is explained in the following tables of RM0368 and RM0383 reference manuals:

- *DMA1 request mapping*
- *DMA2 request mapping.*

3 Application example

This application example demonstrate how to use the peripheral interconnect matrix on STM32F401/411 lines and shows how to synchronize TIM peripherals in parallel mode.

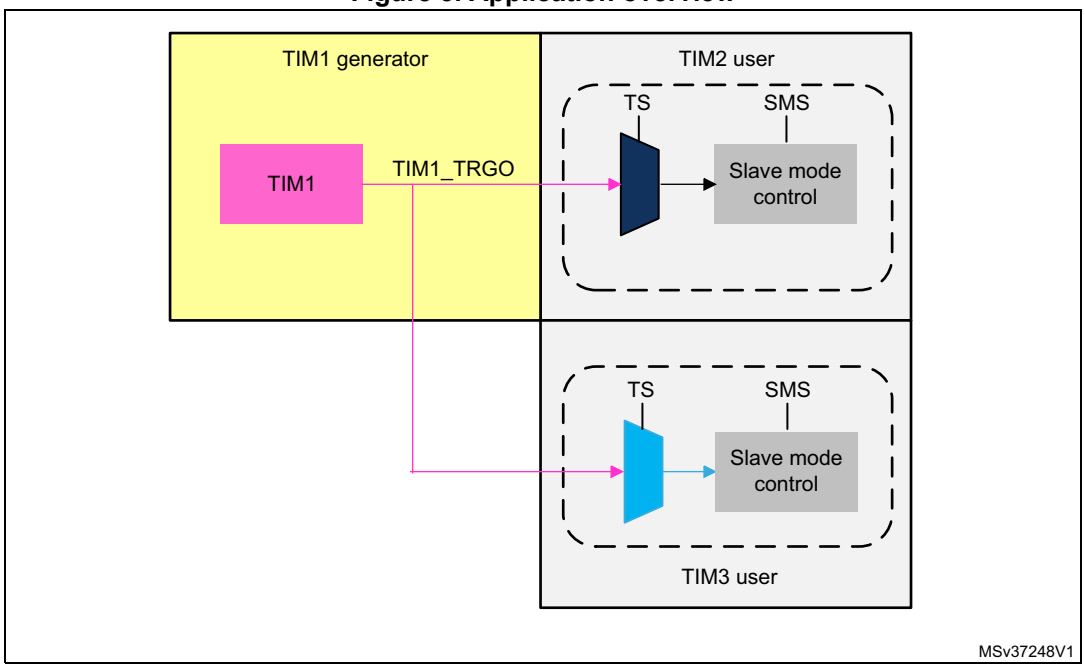
This firmware is based on STM32F4xx HAL drivers which is part of STM32CubeF4.

3.1 Hardware description

Three timers are used in this example:

- TIM1 is configured as master timer:
  - PWM mode is enabled
  - TIM2 update event is used as trigger output.
- TIM2 and TIM3 are slaves for TIM1
  - PWM mode is enabled
  - ITR0(TIM1) is used as trigger input for both slave timers.

Figure 5. Application overview



1. The generator (TIM1) is shown in yellow, while the user blocks (TIM2 and TIM3) are highlighted in gray.

3.2 Software description

TIM1 counter clock is 84 MHz.

TIM1 master timer is running at TIM1 frequency:

- TIM1 frequency = TIM1 counter clock/ (TIM1 period + 1) = 328.125 KHz
- The duty cycle is equal to TIM2_CCR1/ (TIM2_ARR + 1) = 25%.

TIM2 slave timer is running at:

- (TIM1 frequency)/ (TIM2 period + 1) = 32.815 KHz
- The duty cycle is equal to TIM2_CCR1/ (TIM2_ARR + 1) = 30%

TIM3 slave timer is running at:

- (TIM1 frequency)/ (TIM3 period + 1) = 65.630 KHz
- The duty cycle is equal to: TIM3_CCR1/ (TIM3_ARR + 1) = 60%

Table 3: Peripheral interconnect configuration detail gives code examples to configure TIM1, TIM2 and TIM3 as described above.

Table 3. Peripheral interconnect configuration detail

Interconnect	Example	Comments
TIM1 master timer	**sMasterConfig.MasterOutputTrigger = TIM_TRGO_UPDATE** **sMasterConfig.MasterSlaveMode = TIM_MASTERSLAVEMODE_ENABLE;** HAL_TIMEx_MasterConfigSynchronization(&htim1, &sMasterConfig);	Configures TIM2 Update event as Trigger Output

TIM2 slave timer	**sSlaveConfig.SlaveMode = TIM_SLAVEMODE_GATED;**
	sSlaveConfig.InputTrigger = TIM_TS_ITR0;
	HAL_TIM_SlaveConfigSynchronization(&htim2, &sSlaveConfig);
	sMasterConfig.MasterOutputTrigger = TIM_TRGO_RESET;
	sMasterConfig.MasterSlaveMode=TIM_MASTERSLAVEMODE_DISABLE;
	HAL_TIMEx_MasterConfigSynchronization(&htim2, &sMasterConfig);
		Configures ITR0 as trigger input for both slaves.
		Enables the Gated mode so that the start/stop of slave counters is controlled by the master trigger output signal (update event).

TIM3 slave timer	**sSlaveConfig.SlaveMode = TIM_SLAVEMODE_GATED;**
	sSlaveConfig.InputTrigger = TIM_TS_ITR0;
	HAL_TIM_SlaveConfigSynchronization(&htim3, &sSlaveConfig);
	sMasterConfig.MasterOutputTrigger = TIM_TRGO_RESET;
	sMasterConfig.MasterSlaveMode=TIM_MASTERSLAVEMODE_DISABLE;
	HAL_TIMEx_MasterConfigSynchronization(&htim2, &sMasterConfig);	

4 Conclusion

This application note complements STM32F401/411 datasheets and reference manuals by introducing the Peripheral interconnect matrix.

It also gives a detailed description of a basic example that can be a used as a starting point to develop your own application.

5 Revision history

Table 4. Document revision history

Date	Revision	Changes
19-Mar-2015	1	Initial release.