Peripherals interconnections on STM32F405/7xx, STM32F415/7xx, STM32F42xxx, STM32F43xxx, STM32F446xx and STM32F469/479xx

Introduction

This application note describes how peripherals of the microcontrollers listed in *Table 1* can communicate autonomously without any intervention from the CPU via a network known as Peripherals interconnect matrix.

This feature enhances the CPU real-time performance, while at the same time substantially reducing its power consumption.

The document begins with the description of the Peripherals interconnect matrix features, then it provides an overview of the peripherals configuration and their interconnections. An example is presented and described in detail.

This application note has to be read in conjunction with reference manuals RM0090, RM0386 and RM0390, all available at www.st.com/stm32.

Table 1. Applicable devices

Type	Product lines
Microcontrollers	STM32F405/415
	STM32F407/417
	STM32F427/437
	STM32F429/439
	STM32F446
	STM32F469/479

Contents

List of tables

List of figures

1 Module overview

Several peripherals can have direct connections between them, as a matter of fact they can be configured to send or respond to event signals that can be internally routed, directly with other peripherals in the device.

The STM32F4 autonomous peripherals include:

- Timers: can be internally connected to each other, or can be connected to DMA and to the Analog block.
- Analog block: can receive event from timers or can generate event to DMA.
- Clocks: can produce event to Timers.
- System block: can send event to Analog block.
- Communication interfaces block: can generate event to timers or to DMA.

An overview of peripheral interconnections for STM32F4 is shown in *Figure 1*.

Figure 1. Interconnection of STM32F405/7xx, STM32F415/7xx, STM32F42xxx, STM32F43xxx, STM32F446xxand STM32F469/479xx peripherals
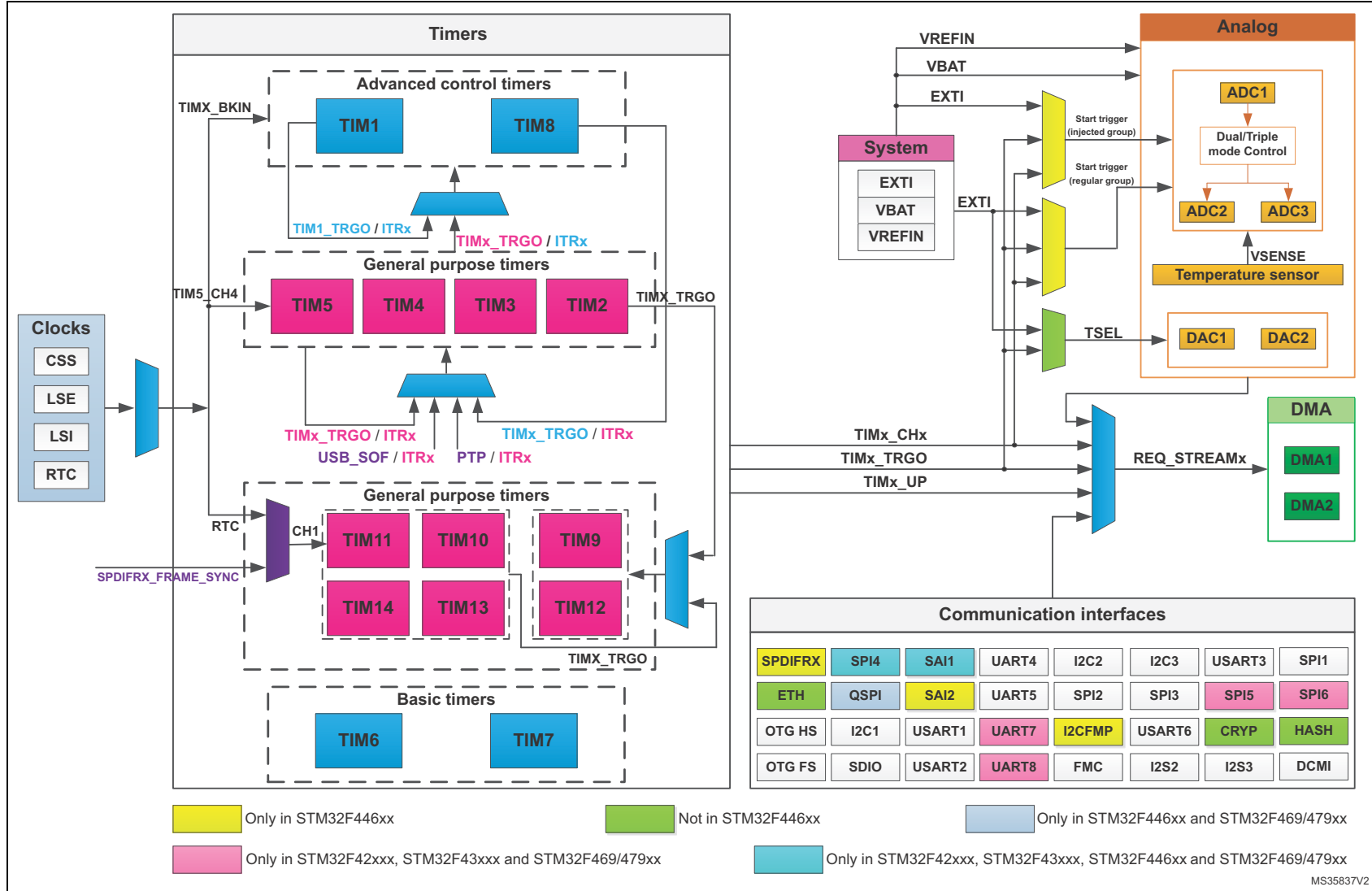
2 Peripherals interconnect matrix

Peripherals in STM32F4 are interconnected by a network named Peripherals interconnect matrix, that makes it possible to directly connect one peripheral to another without waking up the CPU. Depending on peripherals, these interconnections can operate in Run, Sleep and Stop modes.

Peripherals that respond to events are called Users, those that send events are called Generators. Both types are listed in *Table 2*.

Table 2. Peripherals interconnect matrix for products listed in *Table 1*

Generators			TIM1	TIM2	TIM3	TIM4	TIM5	TIM8	TIM9	TIM11	TIM12	DMA1	DMA2	ADC 1	ADC 2	ADC 3	DAC 1	DAC 2
Timers	TIM1		-	X	X	X	-	X	-	-	-	-	X	X	X	X	-	-
	TIM2		X	-	X	X	X	X	X	-	-	X	-	X	X	X	X	X
	TIM3		X	X	-	X	X	-	X	-	-	X	-	X	X	X	-	-
	TIM4		X	X	X	-	X	X	-	-	X	X	-	X	X	X	X	X
	TIM5		X	-	X	-	-	X	-	-	X	X	-	X	X	X	X	X
	TIM6		-	-	-	-	-	-	-	-	-	X	-	-	-	-	X	X
	TIM7		-	-	-	-	-	-	-	-	-	X	-	-	-	-	X	X
	TIM8		-	X	-	X	X	-	-	-	-	-	X	X	X	X	X	X
	TIM10		-	-	-	-	-	-	X	-	-	-	-	-	-	-	-	-
	TIM11		-	-	-	-	-	-	X	-	-	-	-	-	-	-	-	-
	TIM13		-	-	-	-	-	-	-	-	X	-	-	-	-	-	-	-
	TIM14		-	-	-	-	-	-	-	-	X	-	-	-	-	-	-	-
Analog	ADC	1	-	-	-	-	-	X	-	-	-	-	X	-	X	X	-	-
		2	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
		3	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	DAC	1	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
		2	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	VSENSE		-	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-
Clocks	LSI		-	-	-	-	X	-	-	-	-	-	-	-	-	-	-	-
	LSE		-	-	-	-	X	-	-	-	-	-	-	-	-	-	-	-
	RTC		-	-	-	-	X	-	-	X	-	-	-	-	-	-	-	-
	CSS		X	-	-	-	-	X	-	-	-	-	-	-	-	-	-	-

Table 2. Peripherals interconnect matrix for products listed in *Table 1* (continued)

		Users																
		Timers									DMA		Analog					
													ADC			DAC		
Generators		TIM1	TIM2	TIM3	TIM4	TIM5	TIM8	TIM9	TIM11	TIM12	DMA1	DMA2	1	2	3	1	2
Communication interfaces	**OTG FS**	-	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	OTG HS	-	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	SPI1	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	SPI2	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	SPI3	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	SPI4[1]	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	SPI5[2]	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	SPI6[2]	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	CRYP[3]	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	HASH[3]	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	ETH[3]	-	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	SPDIFRX[4]	-	-	-	-	-	-	-	X	-	X	-	-	-	-	-	-
	QSPI[5]	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	I2C1	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	I2C2	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	I2C3	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	I2CFMP[4]	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	I2S2[3]	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	I2S3[3]	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	SAI1[1]	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	SAI2[4]	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	USART1	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	USART2	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	USART3	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	USART6	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	UART4	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	UART5	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	UART7[2]	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	UART8[2]	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-	-
	DCMI	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-
	SDIO	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-	-

Table 2. Peripherals interconnect matrix for products listed in *Table 1* (continued)

		Users															
		Timers									DMA		Analog				
													ADC			DAC	
Generators		TIM1	TIM2	TIM3	TIM4	TIM5	TIM8	TIM9	TIM11	TIM12	DMA1	DMA2	1	2	3	1	2
System	VBAT	-	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-
	VREFIN	-	-	-	-	-	-	-	-	-	-	-	X	-	-	-	-
	EXTI	-	-	-	-	-	-	-	-	-	-	-	X	X	X	X	X

1. Only in STM32F42xxx, STM32F43xxx, STM32F446xx and STM32F469/479xx.

2. Only in STM32F42xxx. STM32F43xxx and STM32F469/479xx.

3. Not in STM32F446xx.

4. Only in STM32F446xx.

5. Only in STM32F446xx and STM32F469/479xx.

2.1 Timers block

2.1.1 From TIM to TIM

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

A description of this feature is provided in the "Timer synchronization" section of RM0090, RM0386 and RM0390 reference manuals, while all the possible master/slave connections are described in the "TIMx internal trigger connection" tables of the same documents.

The output (from Master) is on signal TIMx_TRGO following a configurable timer event. The input (to slave) is on signals TIMx_ITR0/ITR1/ITR2/ITR3.

*Figure 2* is an overview of the trigger selection and the Master mode selection blocks.

Figure 2. Master / Slave timer overview
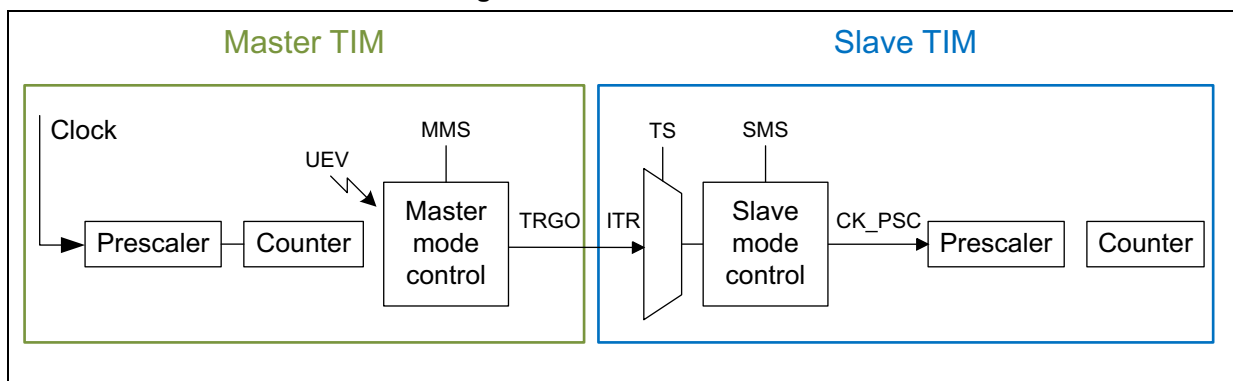
2.1.2 From TIM to ADC

As shown in *Figure 3*, some timers can be used to generate an ADC triggering event.

The output (from timer) is on signal TIMx_TRGO or TIMx_CHx event.

The input (to ADC) is on signal EXTSEL [3:0], JEXTSEL [3:0].

**Figure 3. Master TIM / Slave ADC overview**
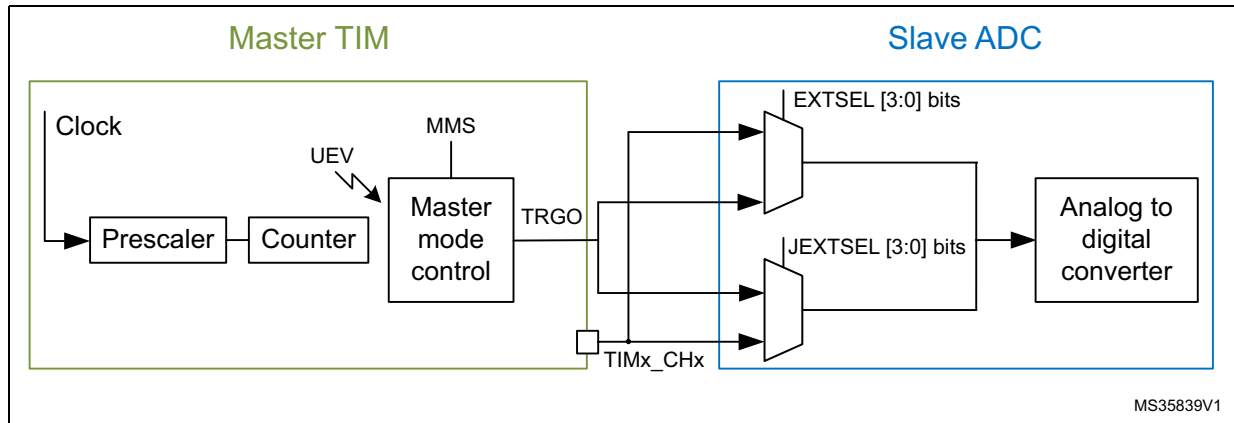
ADC synchronization is described in the "Conversion on external trigger and trigger polarity" section of RM0090, RM0386 and RM0390 reference manuals. The connection between timers and ADCs regular and injected channels is described in the "External trigger for regular channels" and "External trigger for injected channels tables" of the same documents.

2.1.3 From TIM to DAC

Some timers can be used as triggering event to start a DAC conversion (see *Figure 4*).

The output (from timer) is on signal TIMx_TRGO directly connected to corresponding DAC Inputs.

The selection of the input on DAC is provided in the "DAC trigger selection" section of the RM0090, RM0386 and RM0390 reference manuals.

Figure 4. Master TIM / Slave DAC overview
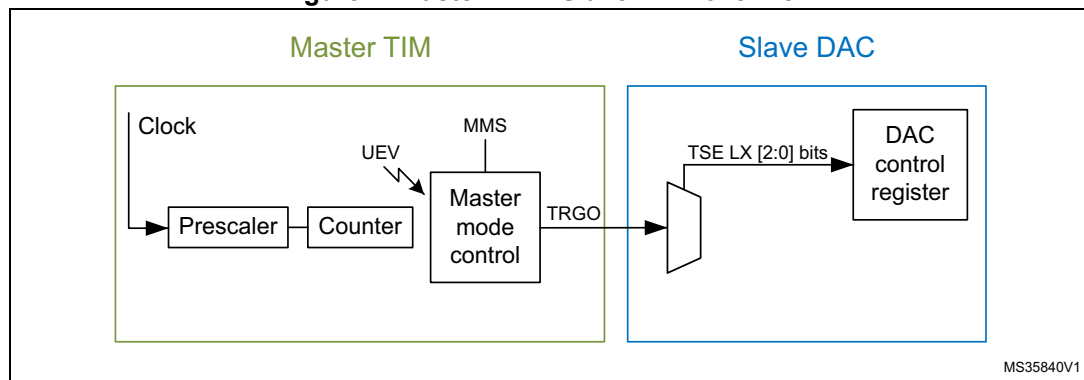
2.1.4 From TIM to DMA

Refer to *Section 2.6: DMA block*.

2.2 Analog block

Analog block includes:

- ADC block: three ADCs;
- DAC block: two DAC converters;
- Temperature sensor Block.

2.2.1 From ADC to ADC

In multi ADC mode, the start of conversion is triggered alternately or simultaneously by the ADC1 master to the ADC2 and ADC3 slaves, depending on the mode selected by the MULTI[4:0] bits in the ADC_CCR register.

This feature is explained in the "Multi ADC mode" section of RM0090, RM0386 and RM0390 reference manuals.

2.2.2 From Temperature sensor to ADC1

On STM32F4xx devices, the temperature sensor is internally connected to the input channel, either ADC1_IN16 or ADC1_IN18 can be used to convert the sensor output voltage into a digital value.

The section "Temperature sensor" in the RM0090, RM0386 and RM0390 reference manuals describes the connection between the sensor and the ADC and the procedure to apply in order to read the converted value.

2.2.3 From Analog block to DMA

Refer to *Section 2.6: DMA block*.

2.3 Clocks block

System block includes:

- LSE clock;
- LSI clock;
- Clock security system (CSS);
- Real-time clock (RTC).

2.3.1 From CSS to TIM

CSS can generate system errors in the form of timer break toward timers.

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

The list of possible break sources is given in the "Using the break function (TIM1/TIM8)" section of the RM0090, RM0386 and RM0390 reference manuals.

2.3.2 From LSE, LSI, RTC to TIM

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

This feature is described in the following sections of the RM0090, RM0386 and RM0390 reference manuals:

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

2.4 System block

Power supplies block includes:

- The internal reference voltage VREFINT;
- VBAT;
- External interrupt/event controller (EXTI).

2.4.1 From VBAT, VREFIN to ADC

The VBAT channel is connected to channel ADC1_IN18. It can be converted as an injected or regular channel.

The VREFINT is connected to ADC1_IN17.

Refer to the following sections of the RM0090, RM0386 and RM0390 reference manuals for a description of the interconnection between VBAT, VREFINT and ADC:

- "Channel selection";
- "Battery charge monitoring".

2.4.2 From EXTI to Analog block

EXTI can be used to generate an ADC triggering event or to start a DAC conversion.

ADC synchronization is described in the "Conversion on external trigger and trigger polarity" section of the RM0090, RM0386 and RM0390 reference manuals, while selection of input triggers on DAC is provided in the "DAC trigger selection" section of the same documents.

2.5 Communication interfaces block

2.5.1 From SPDIFRX to TIM

SPDIFRX (SPDIFRX_FRAME_SYNC) is connected to TIM11_CH1 to measure the clock drift of received SPDIFRX frames.

This interconnection is explained in the section "TIM11 option register 1 (TIM11_OR)" of the RM0390 reference manual.
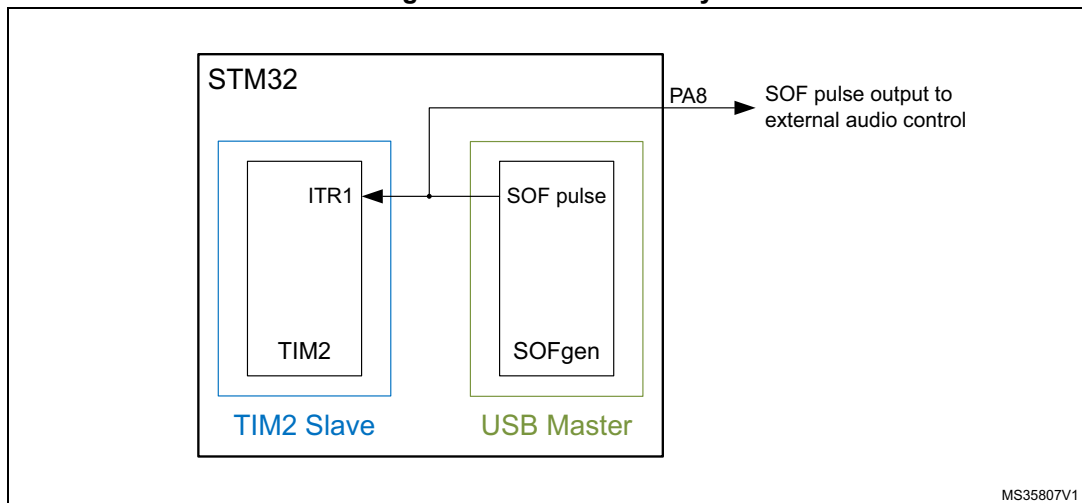
2.5.2 From USB block to TIM

USB block includes:

- USB on-the-go full-speed (OTG_FS);
- USB on-the-go high-speed (OTG_HS).

USB (OTG_FS SOF) and USB (OTG HS SOF) can generate a trigger to general-purpose timer (TIM2), as shown in *Figure 5*.

Figure 5. SOF connectivity

The interconnection between USB and TIM2 is described in the "SOF trigger" section of RM0090, RM0386 and RM0390 reference manuals.

2.5.3 From ETH block to TIM

The MAC can generate a trigger to general-purpose timer (TIM2).

This PTP trigger signal is connected to the TIM2 ITR1 input selectable by software. The connection is enabled through bits 11 and 10 in the TIM2 option register (TIM2_OR).

Figure 6. PTP trigger output to TIM2 ITR1 connection
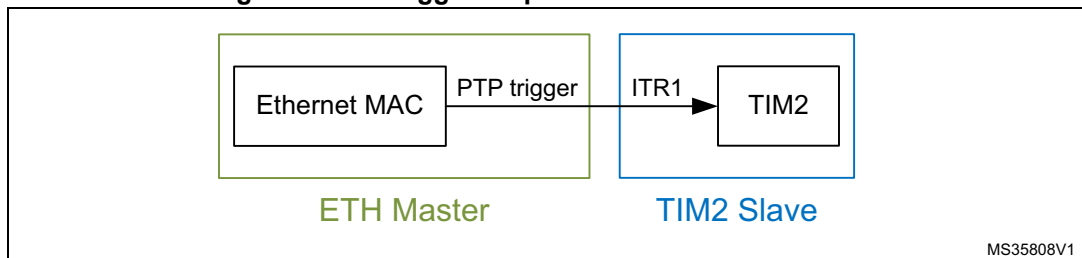
Connection of ETH to TIM2 is described in the section "Precision time protocol (IEEE1588 PTP)" of the RM0090 and RM0386 reference manuals.

2.5.4 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. The selection is controlled by the CHSEL[2:0] bits in 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 RM0090, RM0386 and RM0390 reference manuals:

- "DMA1 request mapping";
- "DMA2 request mapping".

3 Application example

The example described in this section will demonstrate how to use the peripherals interconnect matrix on STM32F4 microcontrollers, namely how to set up ADC1 to start single conversions every time TIM 8 overflows. Each time an end of conversion occurs the DMA transfers the converted data from the ADC to memory.

This application uses the STM32F4xx HAL API.

3.1 Hardware description

Please refer to *Figure 7*, where the same color scheme of *Figure 1* has been used:

- TIM 8 peripheral: used to generate an ADC triggering event;
- ADC1 peripheral: used in Slave mode;
- DMA peripheral: used to transmit data from the slave ADC to the memory.

Figure 7. Application overview
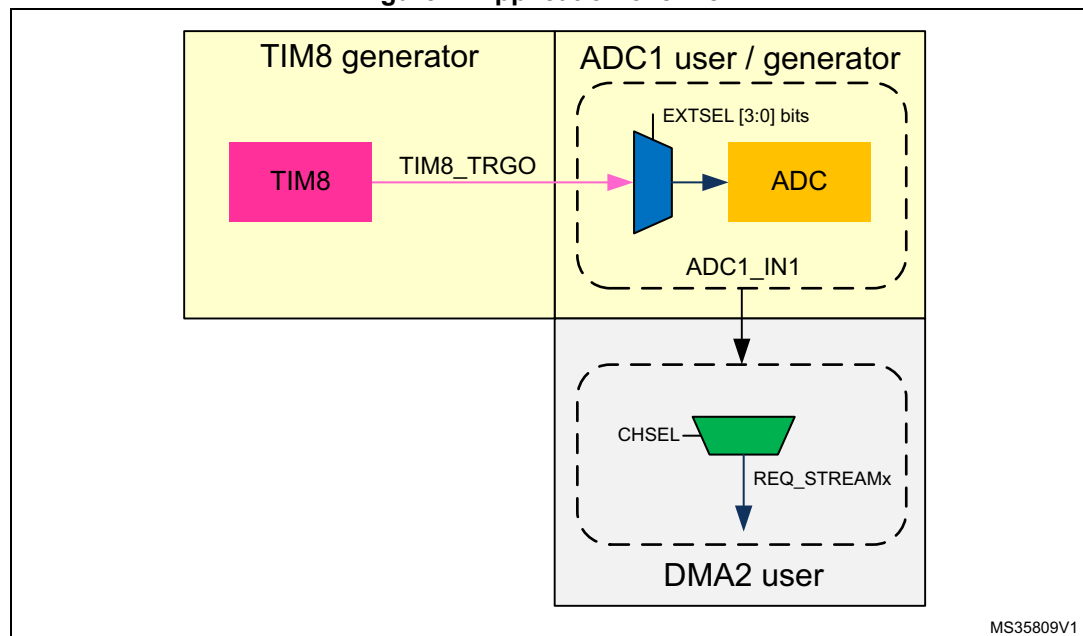
3.2 Software description

The ADC1 is configured to convert TIM8 capture compare event ADC channel1. Each time an end of conversion occurs, the DMA transfers, in normal mode, the converted data from ADC1 DR register to the *aDST_Buffer[ ]* table.

```
/* Enables ADC DMA request after last transfer and enables ADC peripheral*/
HAL_ADC_Start_DMA (&hadc1,(uint32_t*)&aDST_Buffer[0], BUFFER_SIZE);
```

In this example, the system clock is 180 MHz, APB2 = 90 MHz and ADC clock = APB2/2. Since ADC1 clock is 45 MHz and sampling time is set to 3 cycles, the conversion time to 12 bit data is 12 cycles so the total conversion time is $(12 + 3) / 45 = 0.33$ µs.

The detail for related code is provided in *Table 3*.

Table 3. Peripherals interconnect configuration detail

Interconnect	Code example	Comments

TIM8_TRGO selection	`sMasterConfig.MasterOutputTrigger=TIM_TRGO_UPDATE;`
	`sMasterConfig.MasterSlaveMode=TIM_MASTERSLAVEMODE_DISABLE;`
	`sMasterConfig.MasterOutputTrigger = TIM_TRGO_RESET;`
	`HAL_TIMEx_MasterConfigSynchronization(&htim8, &sMasterConfig);`
		Configures the Master TIM to generate a triggering event (TIM_TRGO_UPDATE).

ADC1 external trigger source	`hadc1.Init.ExternalTrigConvEdge=ADC_EXTERNALTRIGCONVEDGE_RISING;`
	`hadc1.Init.ExternalTrigConv = ADC_EXTERNALTRIGCONV_T8_TRGO;`
	`hadc1.Init.DMAContinuousRequests = ENABLE;`
	`hadc1.Init.EOCSelection = ENABLE;`
	`HAL_ADC_Init(&hadc1);`
		The TIM8_TRGO event triggers conversion for the regular group with rising edge.
		Since converted regular channel values are stored into a unique data register, the DMA mode is enabled.

DMA handle	`__HAL_LINKDMA(hadc, DMA_Handle, hdma_adc);`	Associates the initialized DMA handle to the ADC handle.

4 Conclusion

In this application note a useful complement to datasheets and reference manuals has been described by introducing the Peripherals interconnect matrix.

A basic use case has been presented and described in detail.

Users can start from it when developing their own solutions based on microcontrollers of the STM32F405/415, STM32F407/417, STM32F427/437, STM32F429/439, STM32F446 and STM32F469/479 lines.

5 Revision history

Table 4. Document revision history

Date	Revision	Changes
03-Feb-2015	1	Initial release.
19-Mar-2015	2	Changed document classification from ST Restricted to Public. Updated title of *Table 2* and caption of *Figure 1*. Updated references to RM0090 and RM0390 reference manuals in: – *Section 2.1.1: From TIM to TIM*; – *Section 2.1.2: From TIM to ADC*; – *Section 2.1.3: From TIM to DAC*; – *Section 2.2.1: From ADC to ADC*; – *Section 2.2.2: From Temperature sensor to ADC1*; – *Section 2.3.1: From CSS to TIM*; – *Section 2.3.2: From LSE, LSI, RTC to TIM*; – *Section 2.4.1: From VBAT, VREFIN to ADC*; – *Section 2.4.2: From EXTI to Analog block*; – *Section 2.5.1: From SPDIFRX to TIM*; – *Section 2.5.2: From USB block to TIM*; – *Section 2.5.3: From ETH block to TIM*; – *Section 2.6: DMA block*.
17-Aug-2015	3	Added product lines STM32F469/479, hence updated: – *Introduction* and *Table 1* on cover page; – *Figure 1*, *Table 2* and its footnotes. Added reference to RM0386 reference manual in Sections *2.1.1*, *2.1.2*, *2.1.3*, *2.2.1*, *2.2.2*, *2.3.1*, *2.3.2 2.4.1*, *2.4.2*, *2.5.2*, *2.5.3* and *2.6*. Updated *Section 2.5.1: From SPDIFRX to TIM*.

IMPORTANT NOTICE – PLEASE READ CAREFULLY