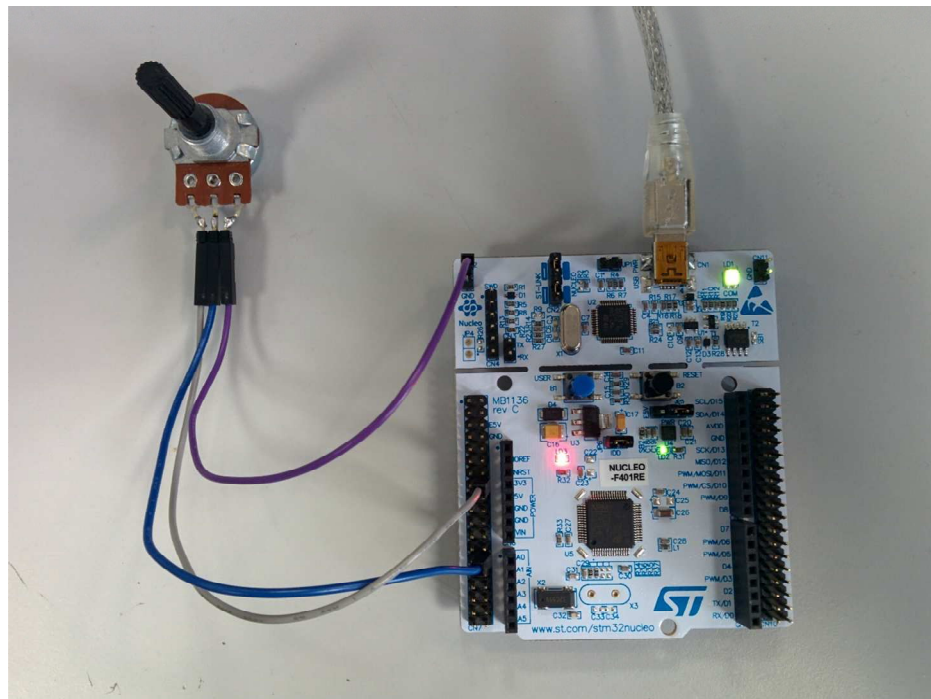


Esercitazione Talentis

Marco Branciforte



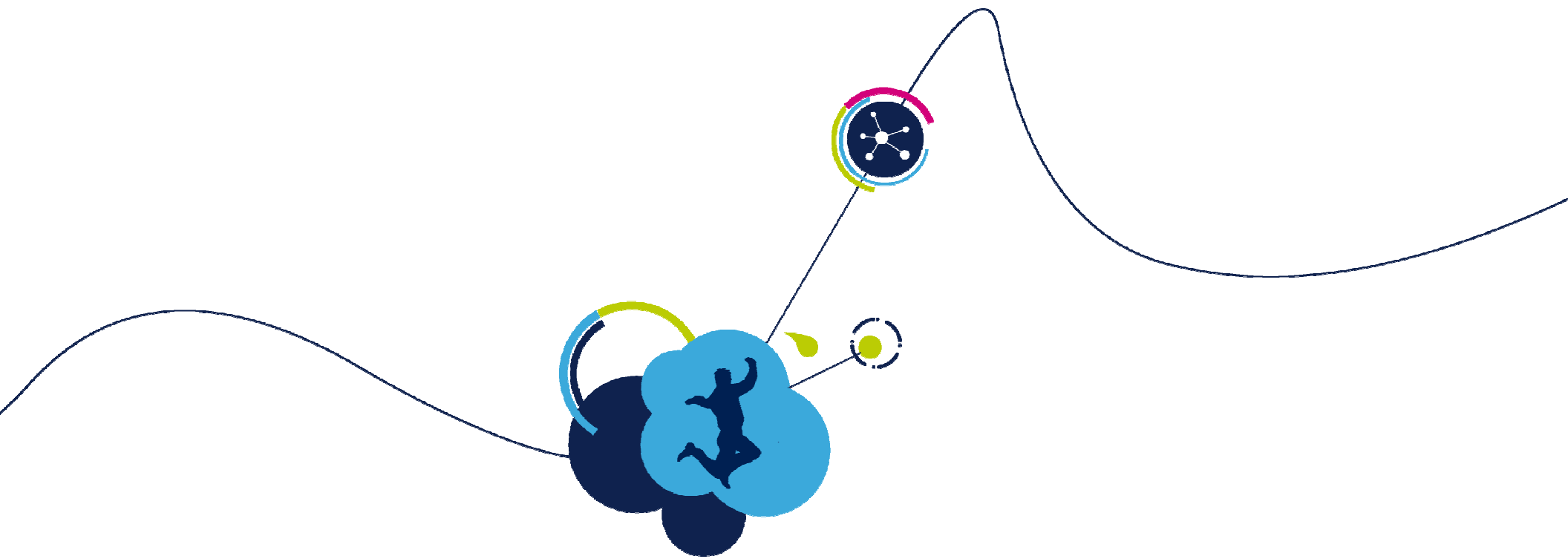
Accendere e variare la luminosita` di un LED



Periferiche utilizzate

3

- PWM => per pilotare il LED
 - Timer 2 Channel 1 - PWM mode
 - 100Hz
- ADC => per variare la luminosita` del LED variando proporzionalmente al valore analogico acquisito il Duty Cycle del PWM
 - ADC1 IN0 – Single and regular conversion
- TIMER INTERRUPT => per sincronizzare le acquisizioni dell'ADC
 - Timer 3 - Base functionality
 - 1KHz (1ms)
- PULSANTE BLU => per accendere e spegnere il LED
 - External Interrupt

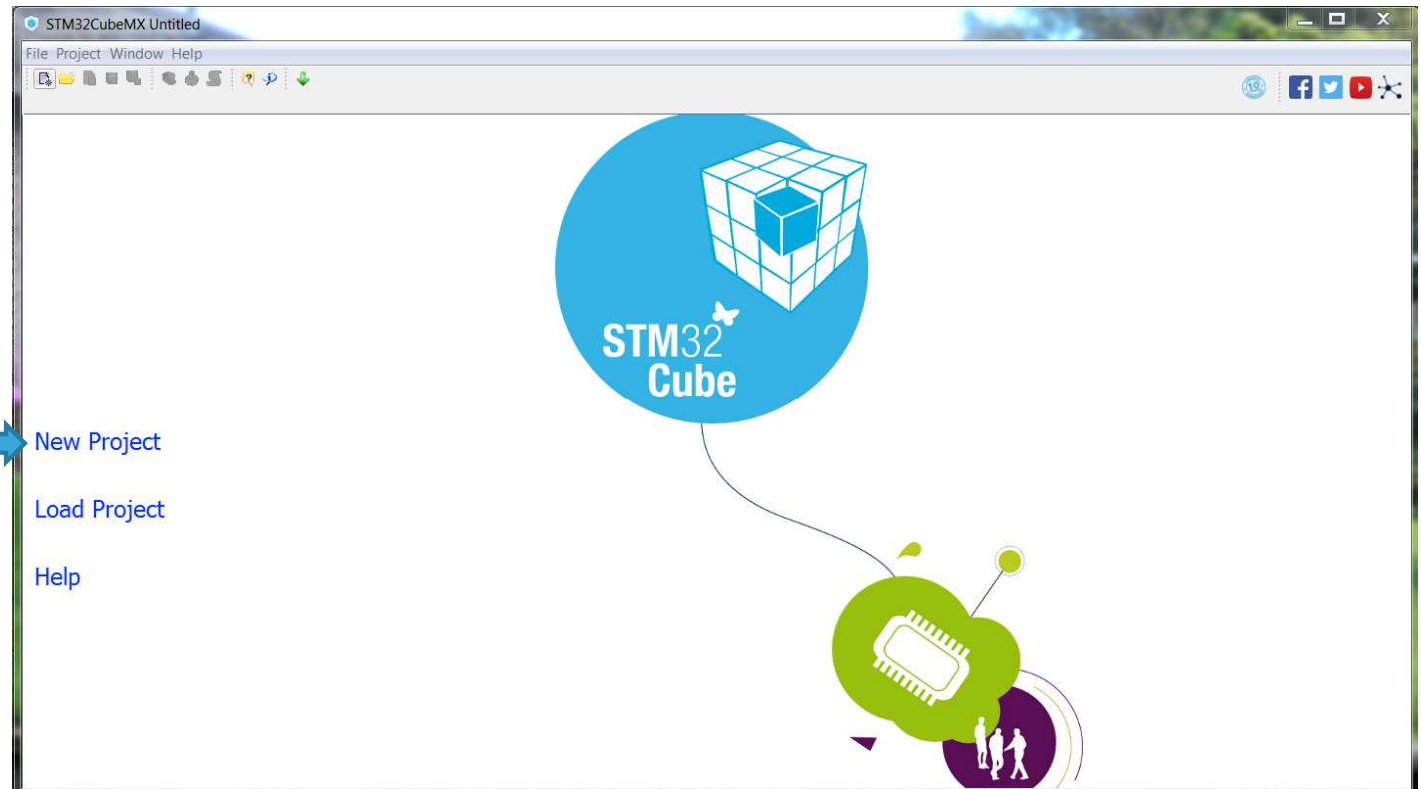


Cube MX

Cube MX Project: New Project

5

Selezionare
New Project



Cube MX Project: New Project

6

New Project

MCU Selector Board Selector

MCU Filters

Part Number Search

Core

Series

Check/Uncheck All

- ☐ STM32F0
- ☐ STM32F1
- ☐ STM32F2
- ☐ STM32F3
- ☐ STM32F4
- ☐ STM32F7
- ☐ STM32H7
- ☐ STM32L0
- ☐ STM32L1
- ☐ STM32L4
- ☐ STM32L4+

Line

Package

Check/Uncheck All

- ☐ EWLCP49
- ☐ EWLCP66
- ☐ LFBGA100
- ☐ LFBGA144
- ☐ LQFP32
- ☐ LQFP48

Cliccare sul tab: Board Selector

Features Block Diagram Datasheet Docs & Resources Buy Start Project

MCUs List: 1129 items

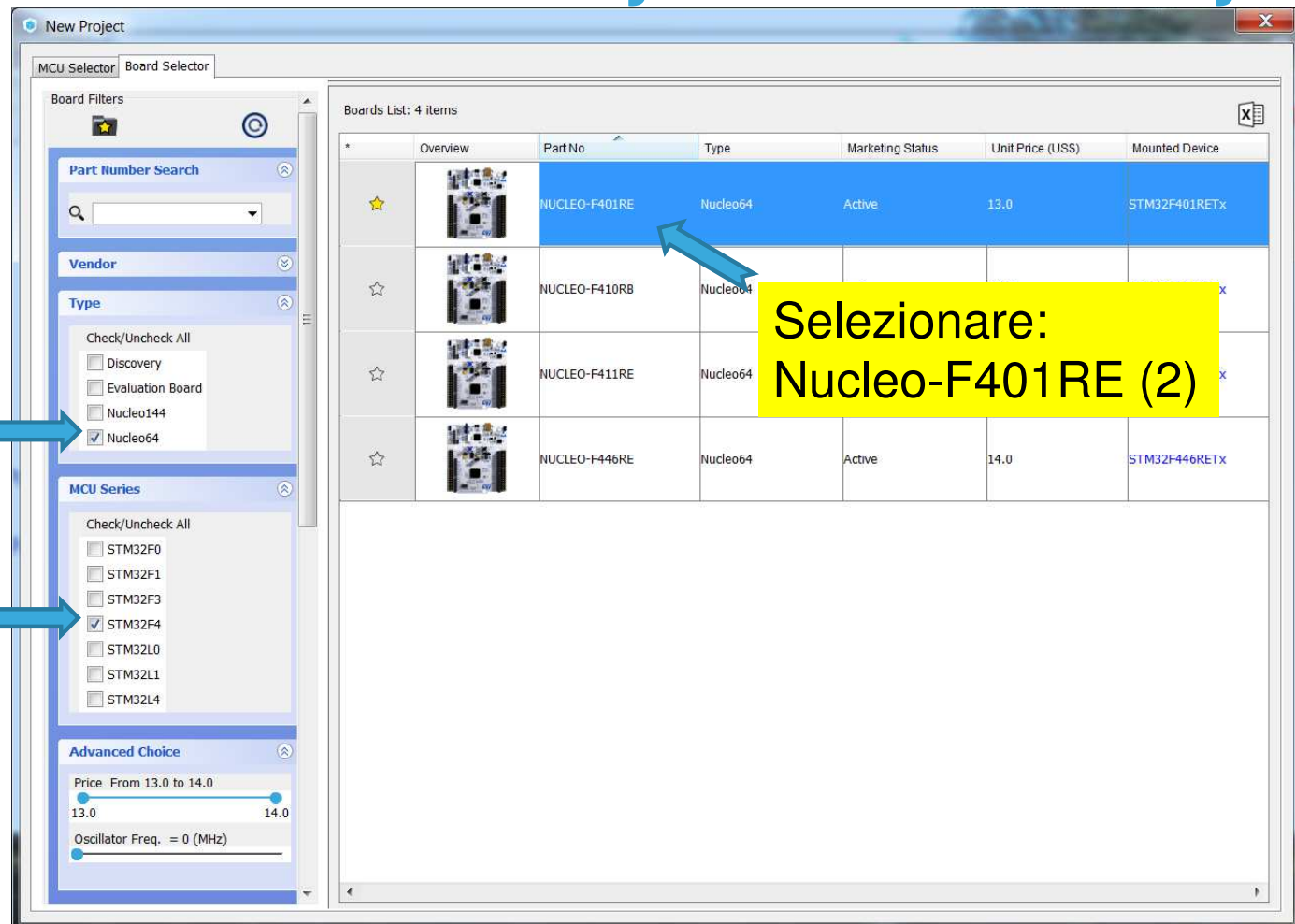
Display similar it...

* Part No	Reference	Marketing Stat...	Unit Price for 10KU (...)	Board	Package	Flash	RAM	IO	Freq.	GFX Score
☆ STM32F030C6	STM32F030C6Tx	Active	0.597		LQFP48	32 kBytes	4 kBytes	39	48 MHz	0.0
☆ STM32F030C8	STM32F030C8Tx	Active	0.723		LQFP48	64 kBytes	8 kBytes	39	48 MHz	0.0
☆ STM32F030CC	STM32F030CCTx	Active	1.1		LQFP48	256 kBytes	32 kBytes	37	48 MHz	0.0
☆ STM32F030F4	STM32F030F4Px	Active	0.424		TSSOP20	16 kBytes	4 kBytes	15	48 MHz	0.0
☆ STM32F030K6	STM32F030K6Tx	Active	0.519		LQFP32	32 kBytes	4 kBytes	25	48 MHz	0.0
☆ STM32F030R8	STM32F030R8Tx	Active	0.754	...	LQFP64	64 kBytes	8 kBytes	55	48 MHz	0.0
☆ STM32F030RC	STM32F030RCTx	Active	1.21	...	LQFP64	256 kBytes	32 kBytes	51	48 MHz	0.0
☆ STM32F031C4	STM32F031C4Tx	Active	0.971		LQFP48	16 kBytes	4 kBytes	39	48 MHz	0.0
☆ STM32F031C6	STM32F031C6Tx	Active	1.014		LQFP48	32 kBytes	4 kBytes	39	48 MHz	0.0
☆ STM32F031E6	STM32F031E6Yx	Active	0.777		WLCSP25	32 kBytes	4 kBytes	20	48 MHz	0.0
☆ STM32F031F4	STM32F031F4Px	Active	0.711		TSSOP20	16 kBytes	4 kBytes	15	48 MHz	0.0
☆ STM32F031F6	STM32F031F6Px	Active	0.756		TSSOP20	32 kBytes	4 kBytes	15	48 MHz	0.0
☆ STM32F031G4	STM32F031G4Ux	Active	0.733		UFQFPN28	16 kBytes	4 kBytes	23	48 MHz	0.0
☆ STM32F031G6	STM32F031G6Ux	Active	0.777		UFQFPN28	32 kBytes	4 kBytes	23	48 MHz	0.0
☆ STM32F031K4	STM32F031K4Ux	Active	0.863		UFQFPN32	16 kBytes	4 kBytes	27	48 MHz	0.0
☆ STM32F031K6	STM32F031K6Tx	Active	0.906	NUCLEO-F103RB	LQFP32	32 kBytes	4 kBytes	25	48 MHz	0.0
☆ STM32F031K6	STM32F031K6Ux	Active	0.906		UFQFPN32	32 kBytes	4 kBytes	27	48 MHz	0.0
☆ STM32F038C6	STM32F038C6Tx	Active	1.014		LQFP48	32 kBytes	4 kBytes	38	48 MHz	0.0
☆ STM32F038E6	STM32F038E6Yx	Active	0.777		WLCSP25	32 kBytes	4 kBytes	20	48 MHz	0.0
☆ STM32F038F6	STM32F038F6Px	Active	0.756		TSSOP20	32 kBytes	4 kBytes	14	48 MHz	0.0
☆ STM32F038G6	STM32F038G6Ux	Active	0.777		UFQFPN28	32 kBytes	4 kBytes	22	48 MHz	0.0
☆ STM32F038K6	STM32F038K6Ux	Active	0.906		UFQFPN32	32 kBytes	4 kBytes	26	48 MHz	0.0
☆ STM32F042C4Tx	STM32F042C4Tx	Active	1.057		LOFP48	16 kBytes	6 kBytes	38	48 MHz	0.0

Cube MX Project: New Project

7

Selezionare:
Nucleo64 e
STM32F4 (1)

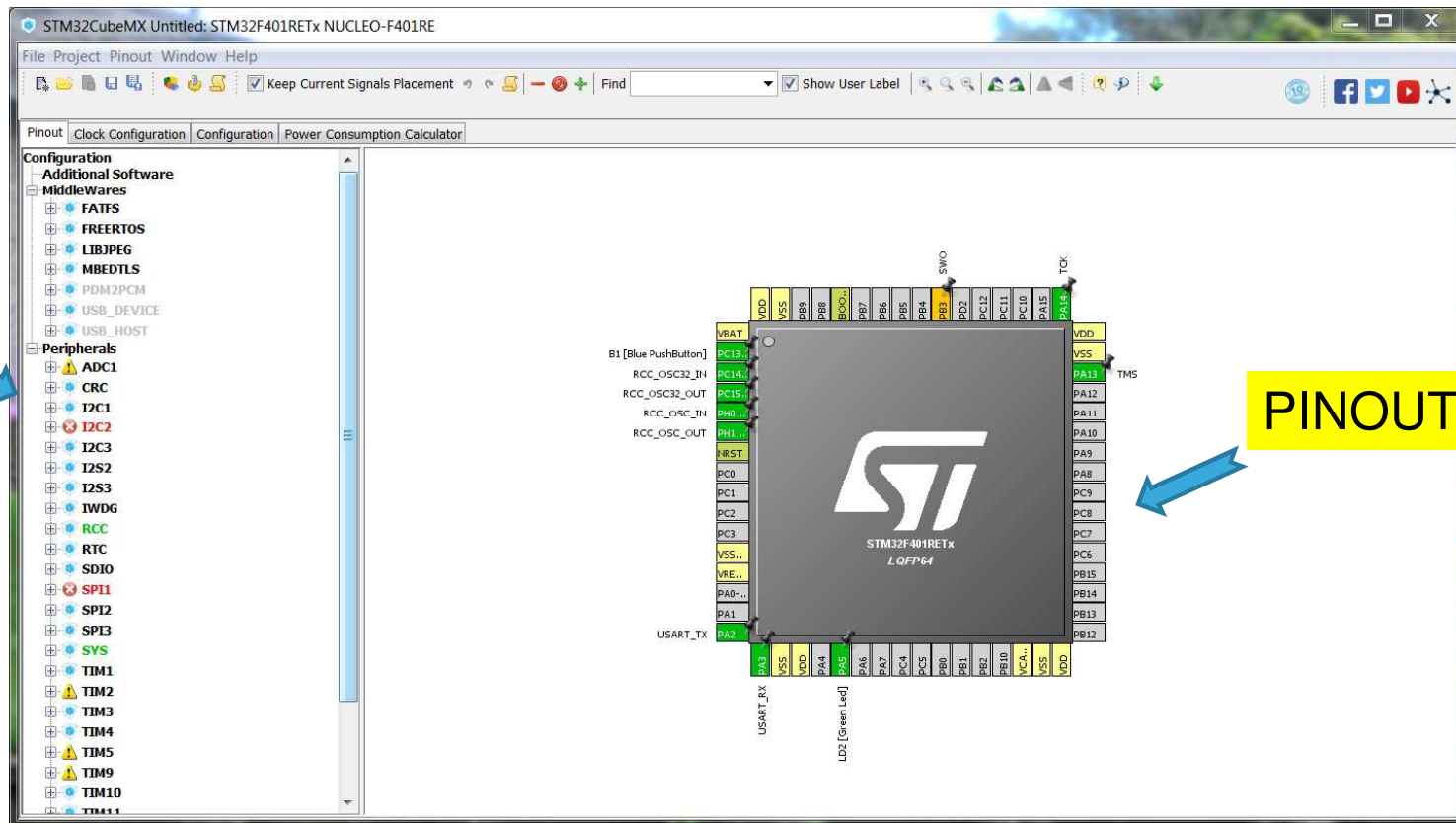


Cube MX Project: New Project

8

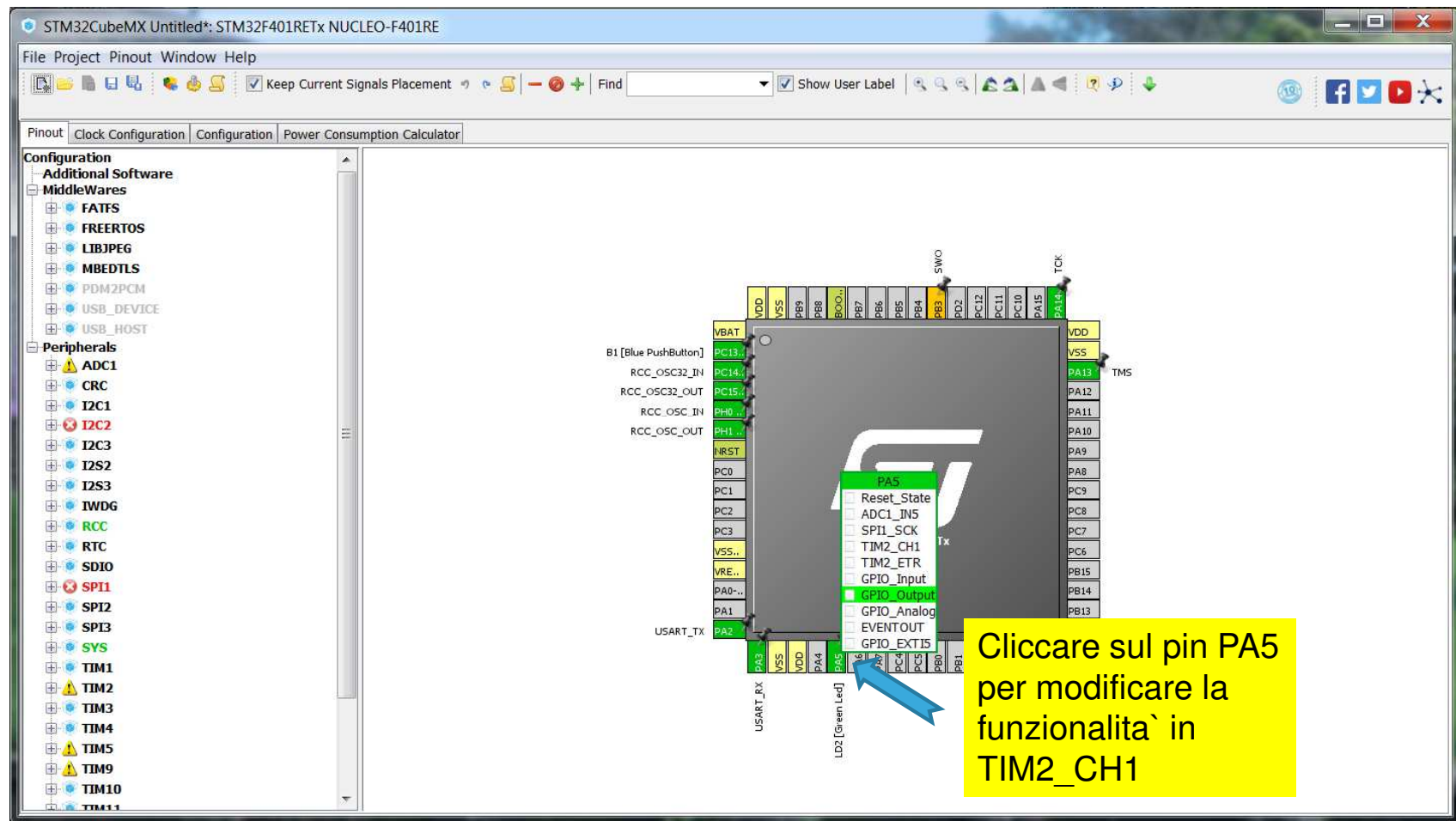
Periferiche

PINOUT



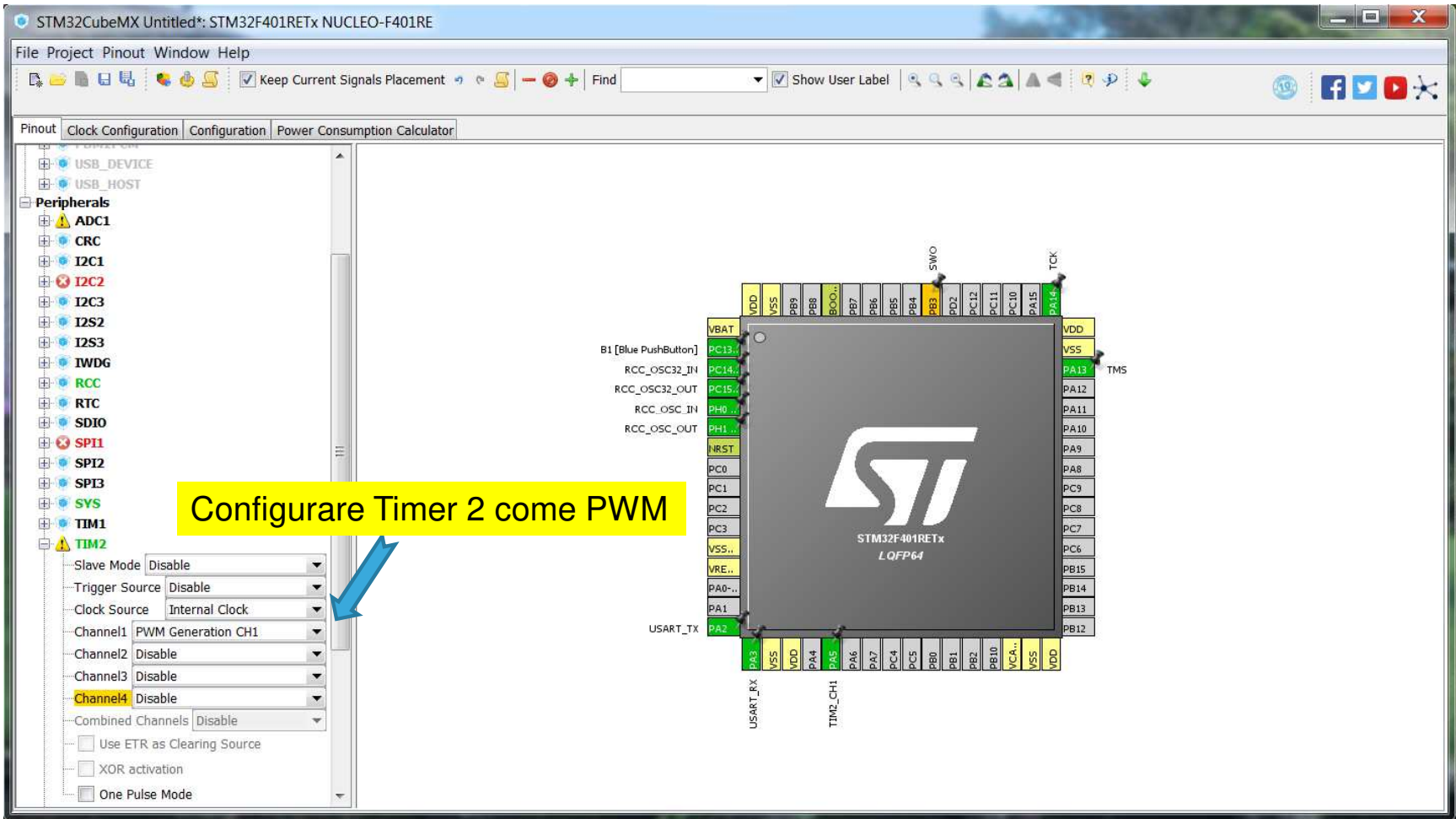
Cube MX: attivazione PIN e periferiche

9



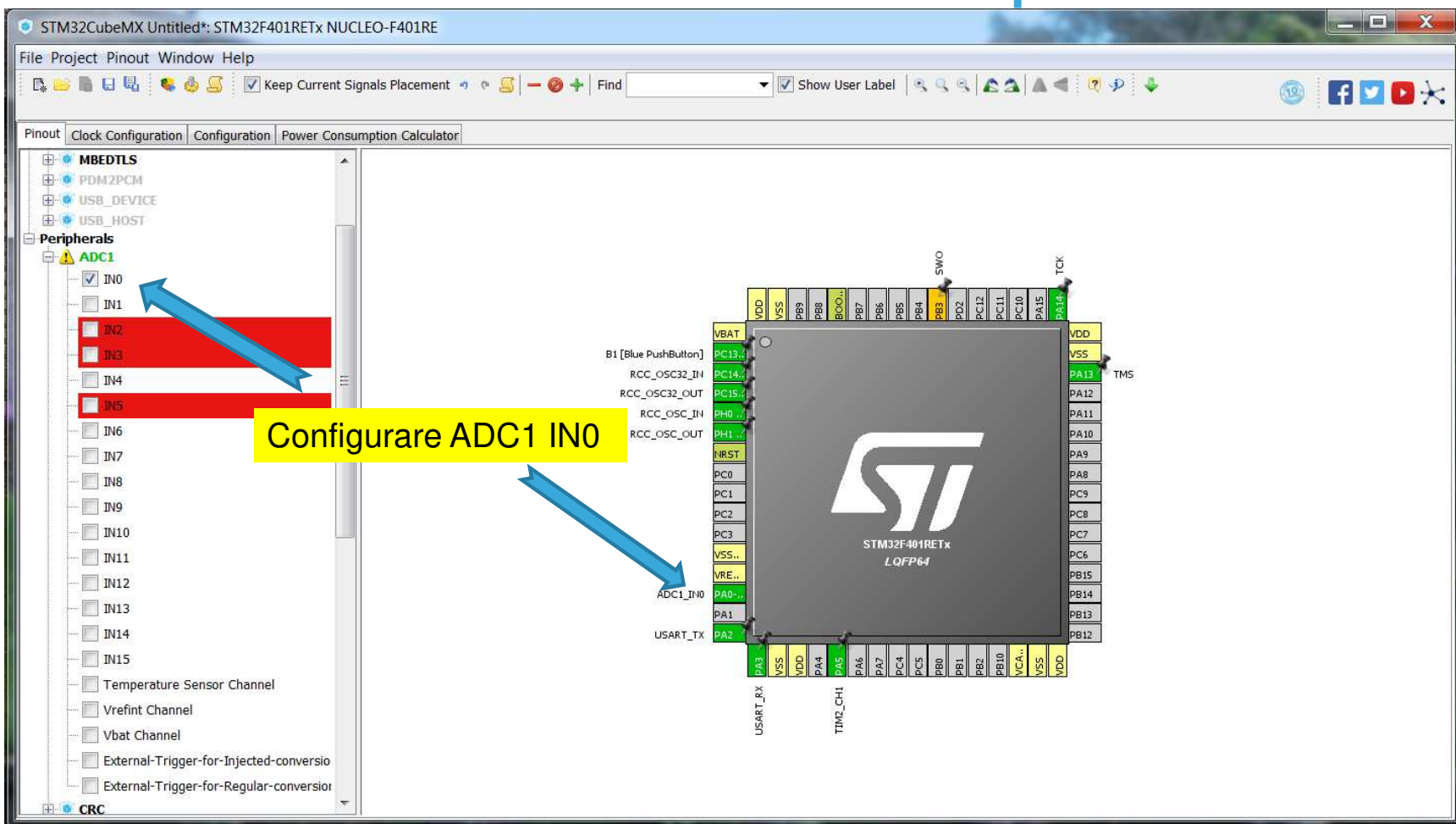
Cube MX: attivazione PIN e periferiche

10



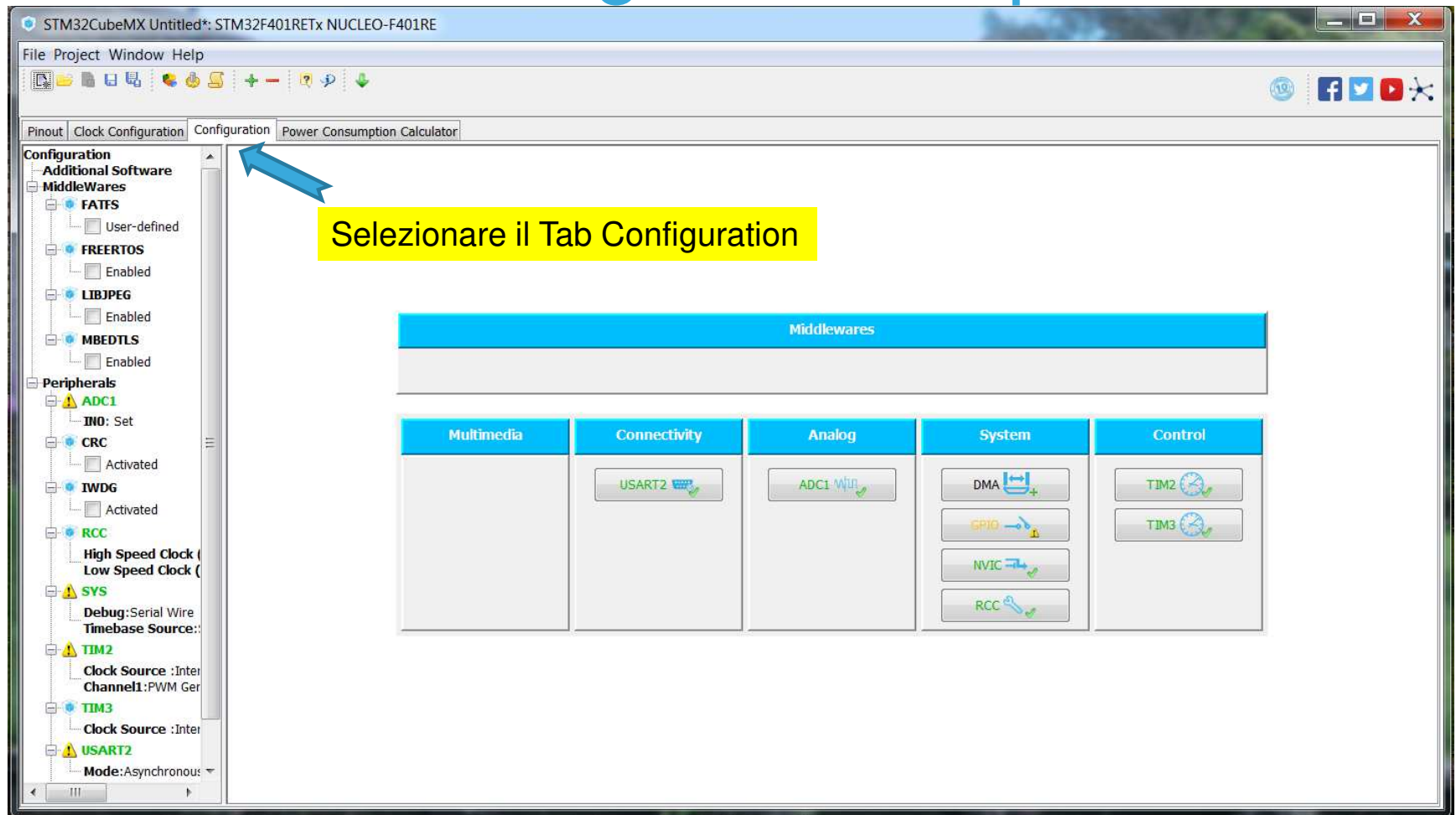
Cube MX: attivazione PIN e periferiche

12



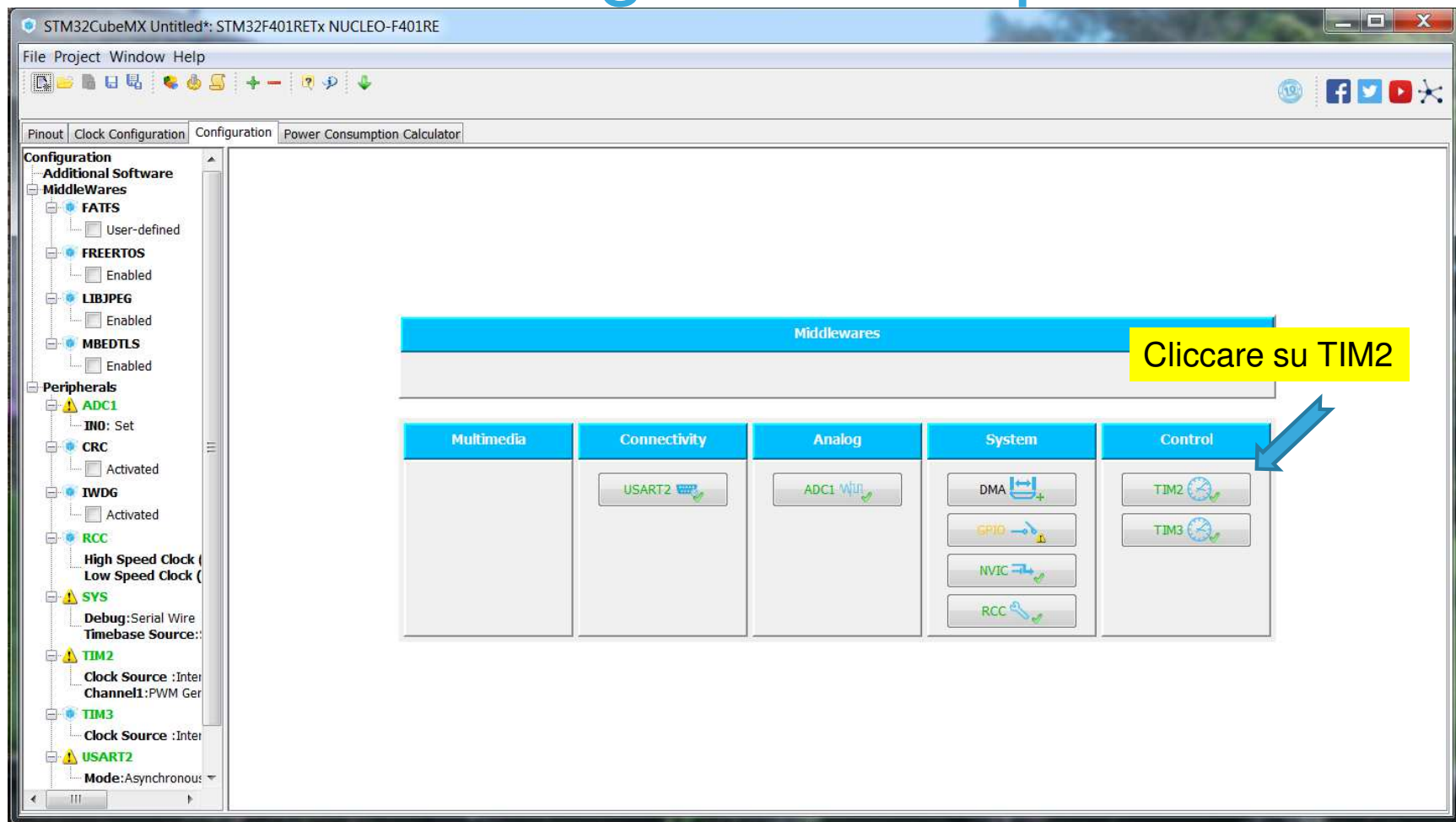
Cube MX: configurazione periferiche

13



Cube MX: configurazione periferiche

14



Cube MX: configurazione periferiche

15

TIM2 Configuration

✓ Parameter Settings ✓ User Constants ✓ NVIC Settings ✓ DMA Settings ✓ GPIO Settings

Configure the below parameters :

Search :

☐ Counter Settings

Prescaler (PSC - 16 bits value)	83
Counter Mode	Up
Counter Period (AutoReload Register - 32 bit...)	9999
Internal Clock Division (CKD)	No Division

☐ Trigger Output (TRGO) Parameters

Master/Slave Mode (MSM bit)	Disable (Trigger input effect n)
Trigger Event Selection	Reset (UG bit from TIMx_EGR)

☐ PWM Generation Channel 1

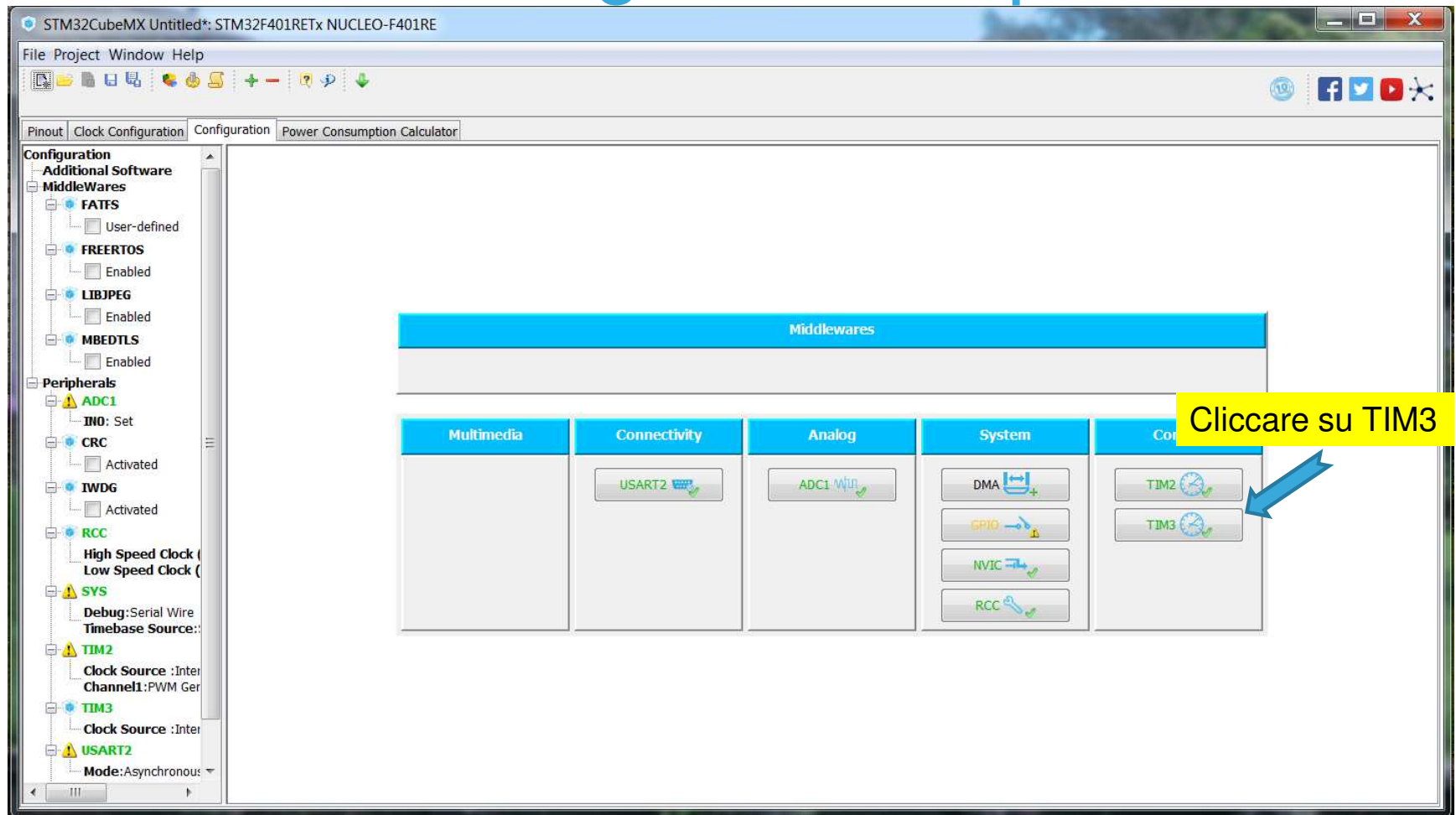
Mode	PWM mode 1
Pulse (32 bits value)	5000
Fast Mode	Disable
CH Polarity	High

Restore Default Apply Ok Cancel

Inserire i parametri per avere un PWM a 100Hz e Duty Cycle al 50%

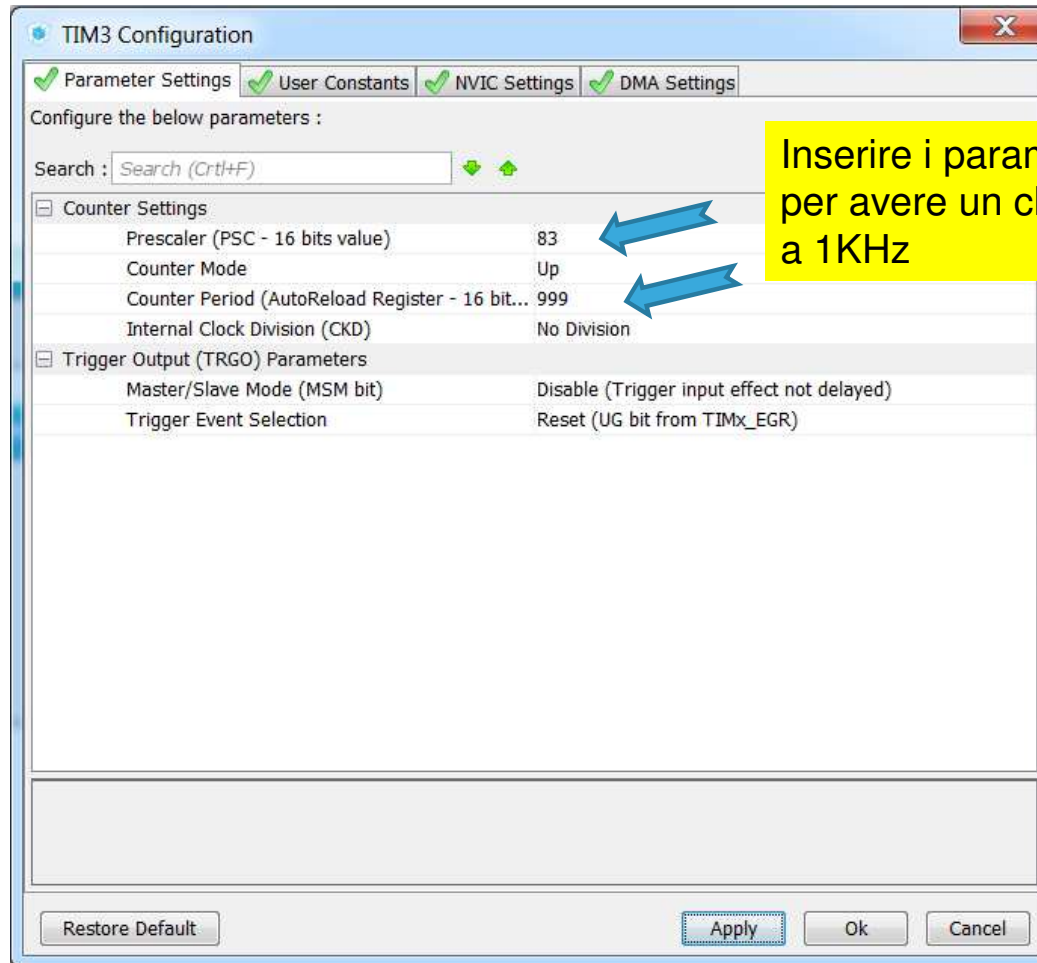
Cube MX: configurazione periferiche

16



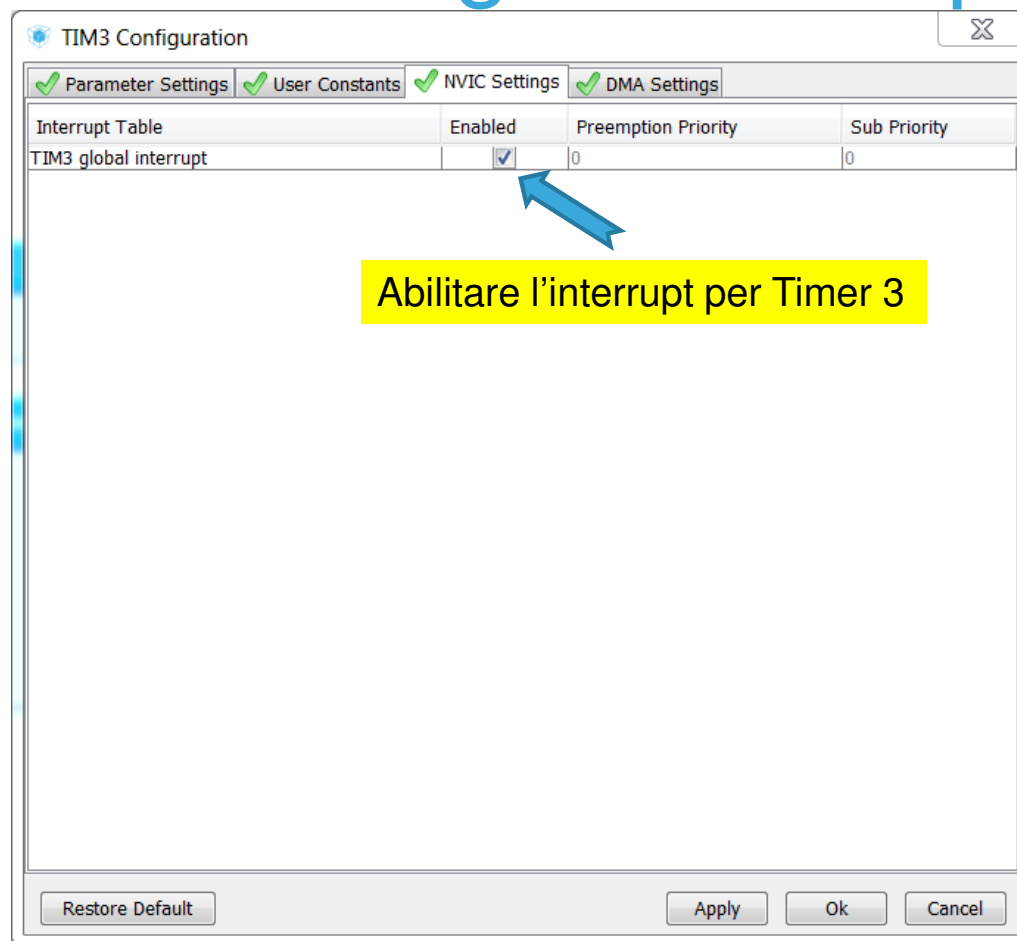
Cube MX: configurazione periferiche

17



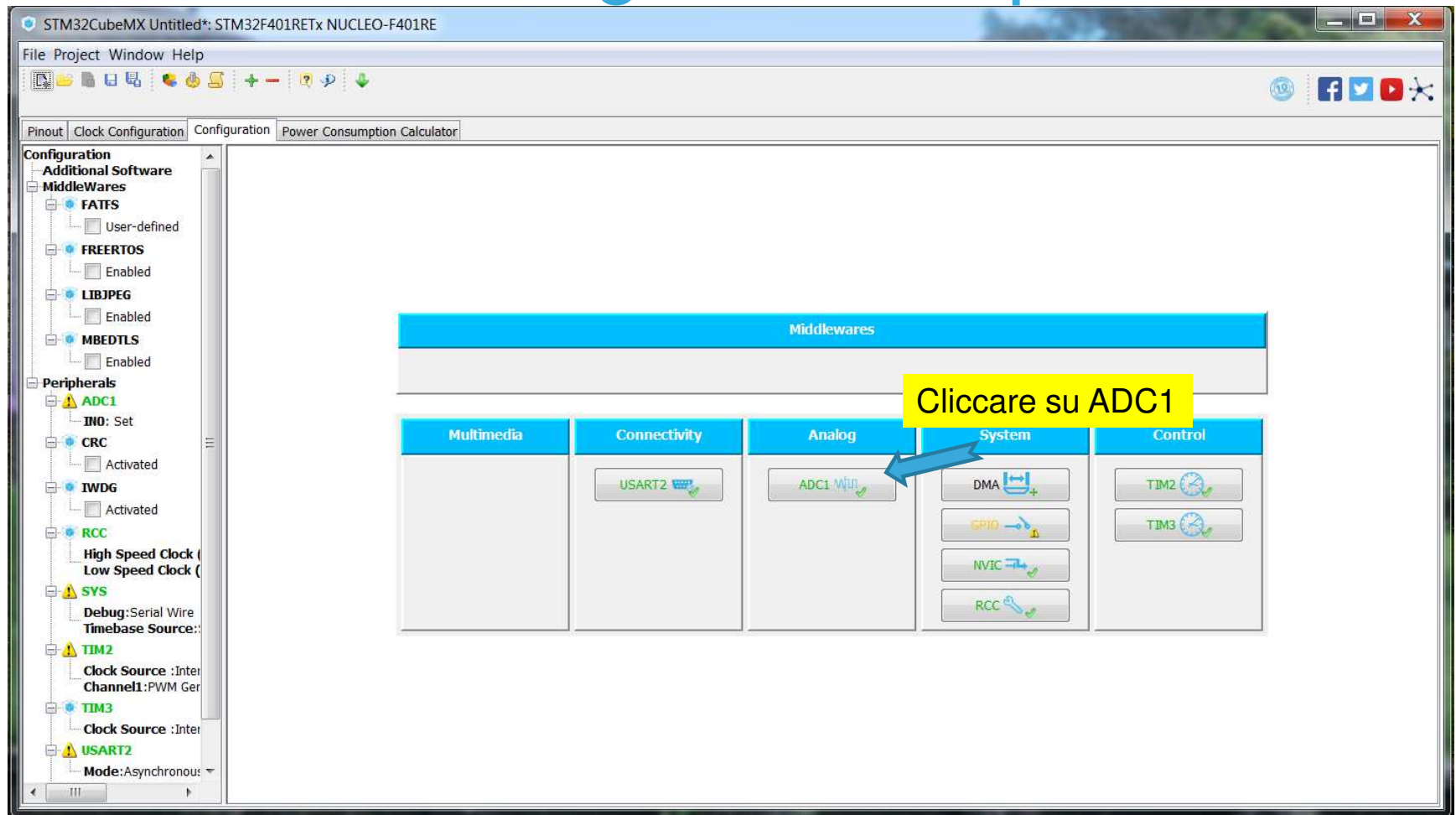
Cube MX: configurazione periferiche

18



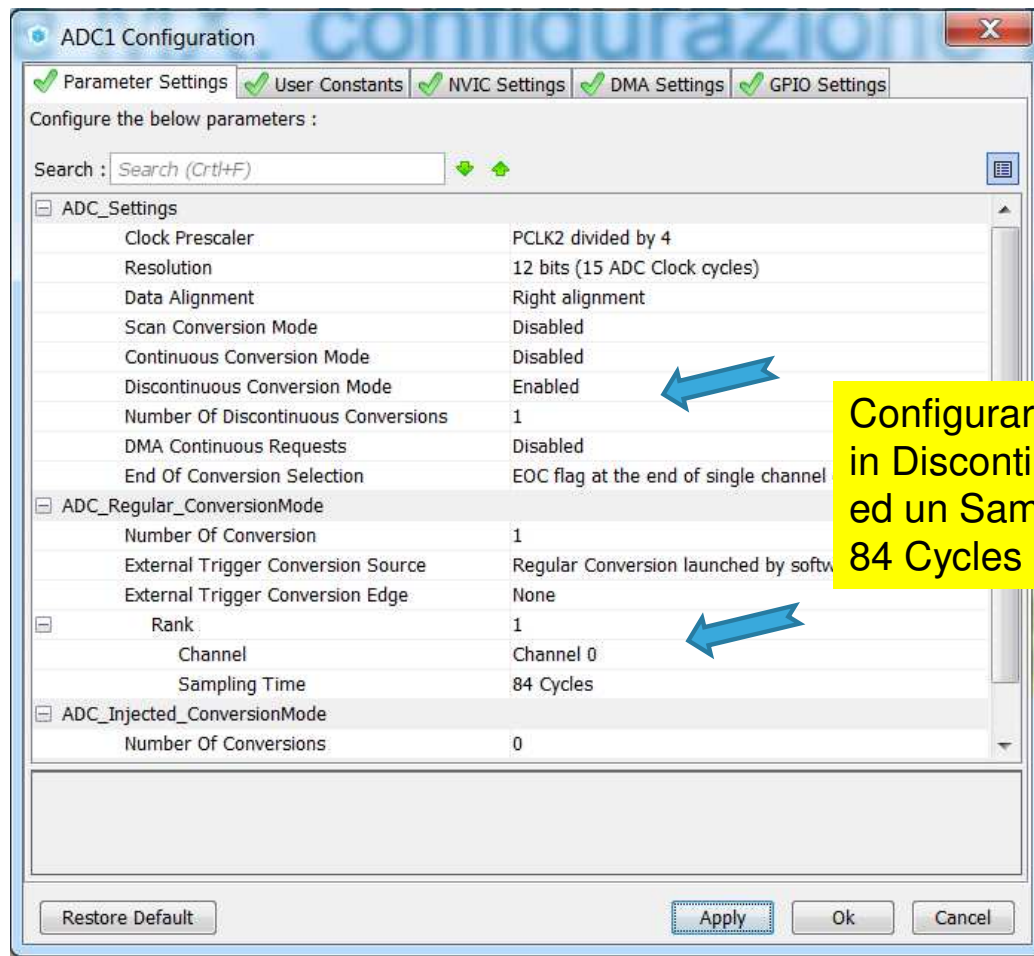
Cube MX: configurazione periferiche

19



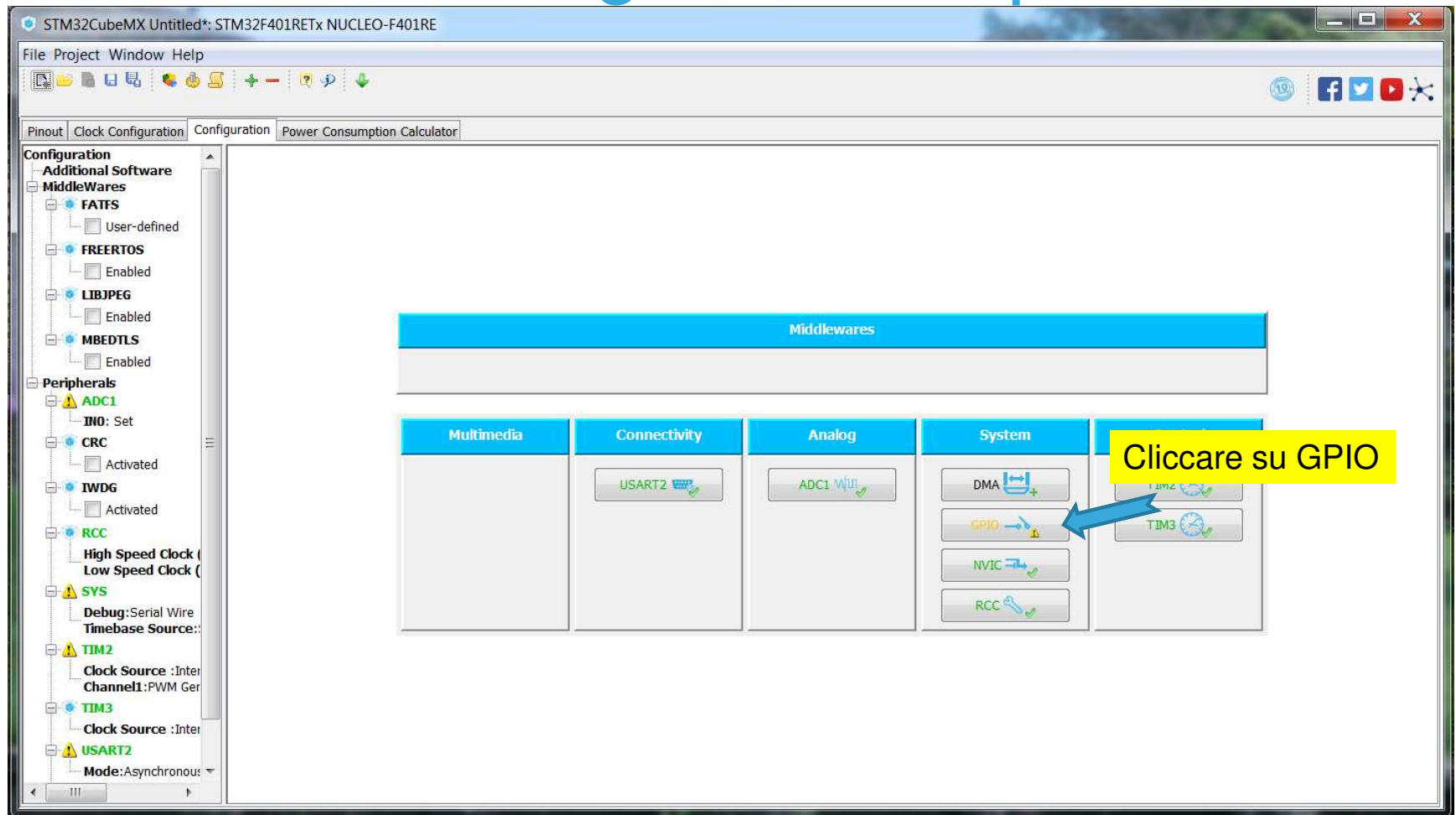
Cube MX: configurazione periferiche

20



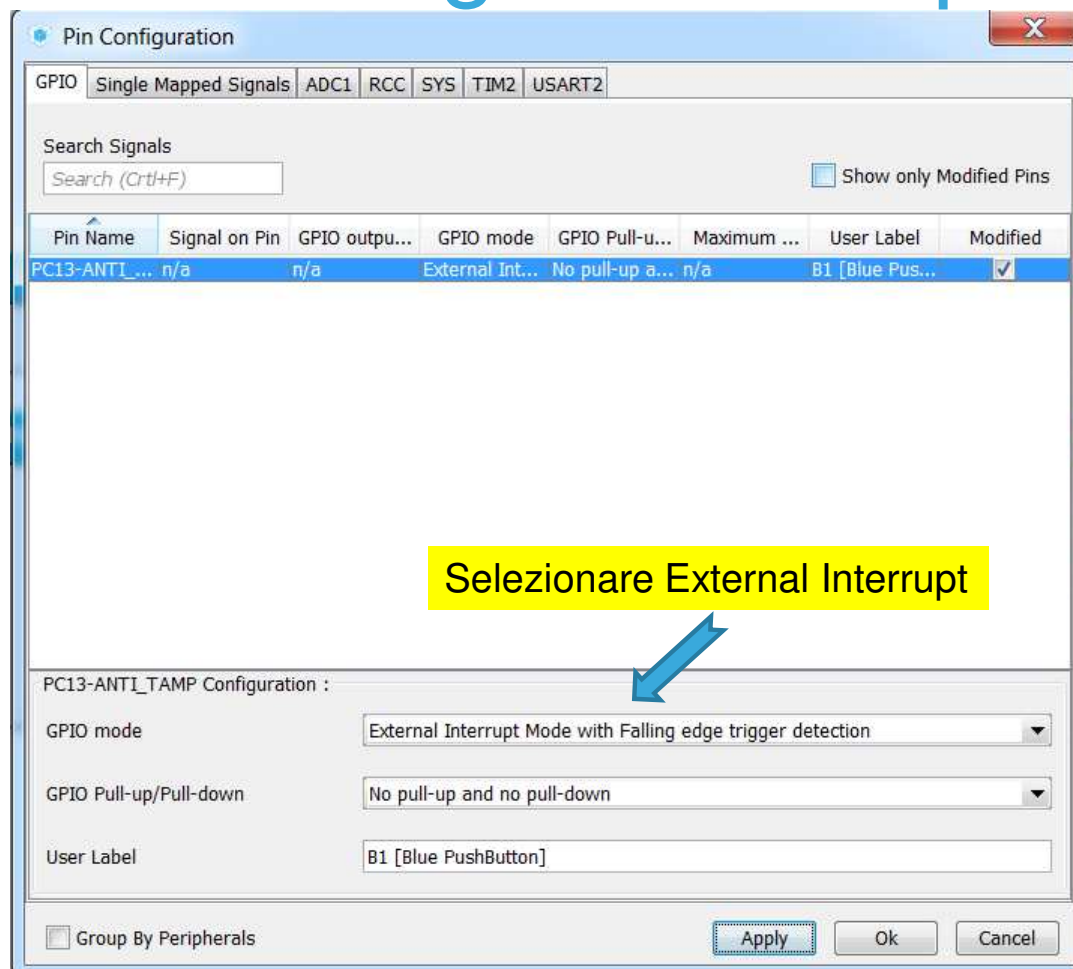
Cube MX: configurazione periferiche

21



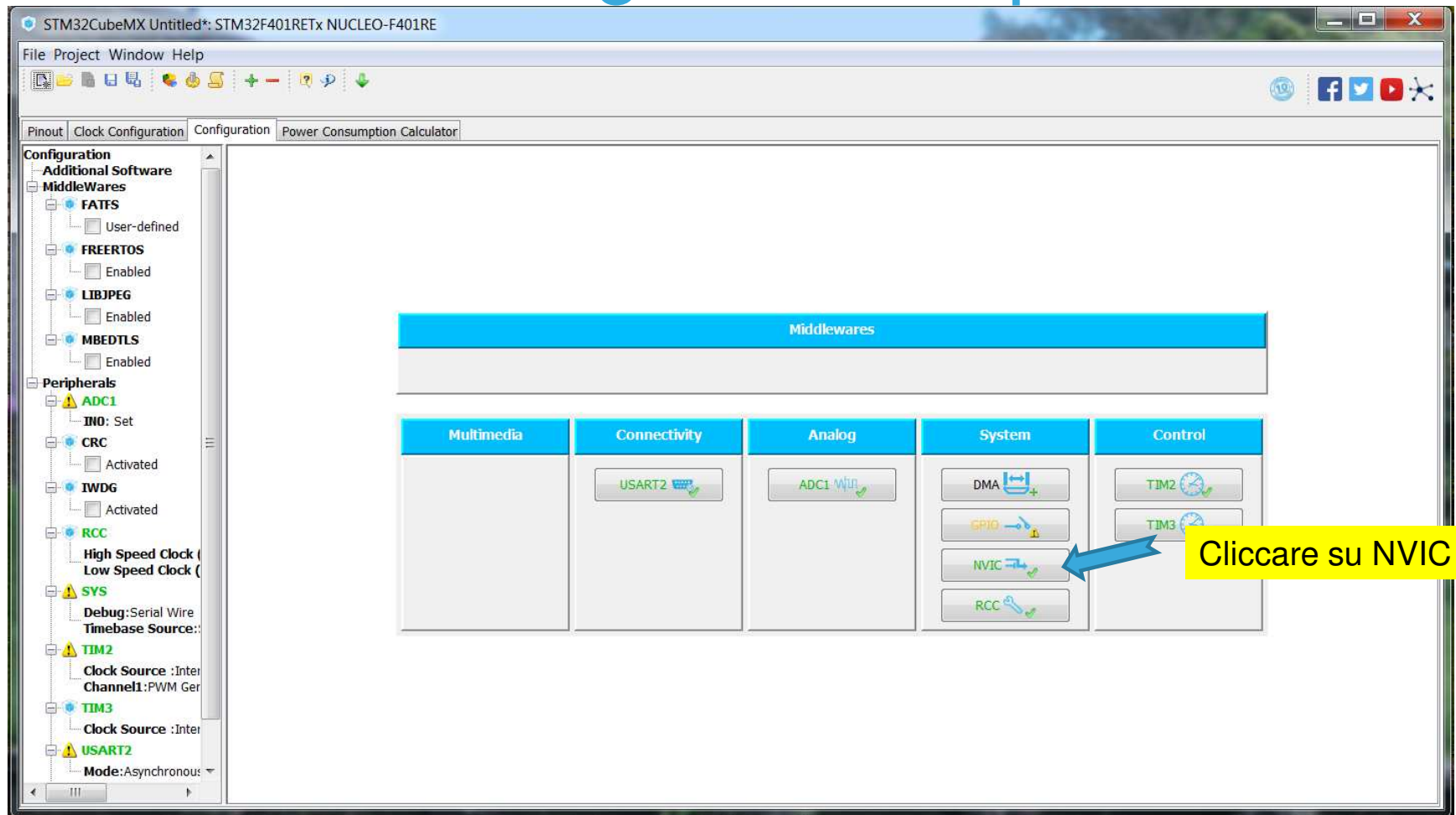
Cube MX: configurazione periferiche

22



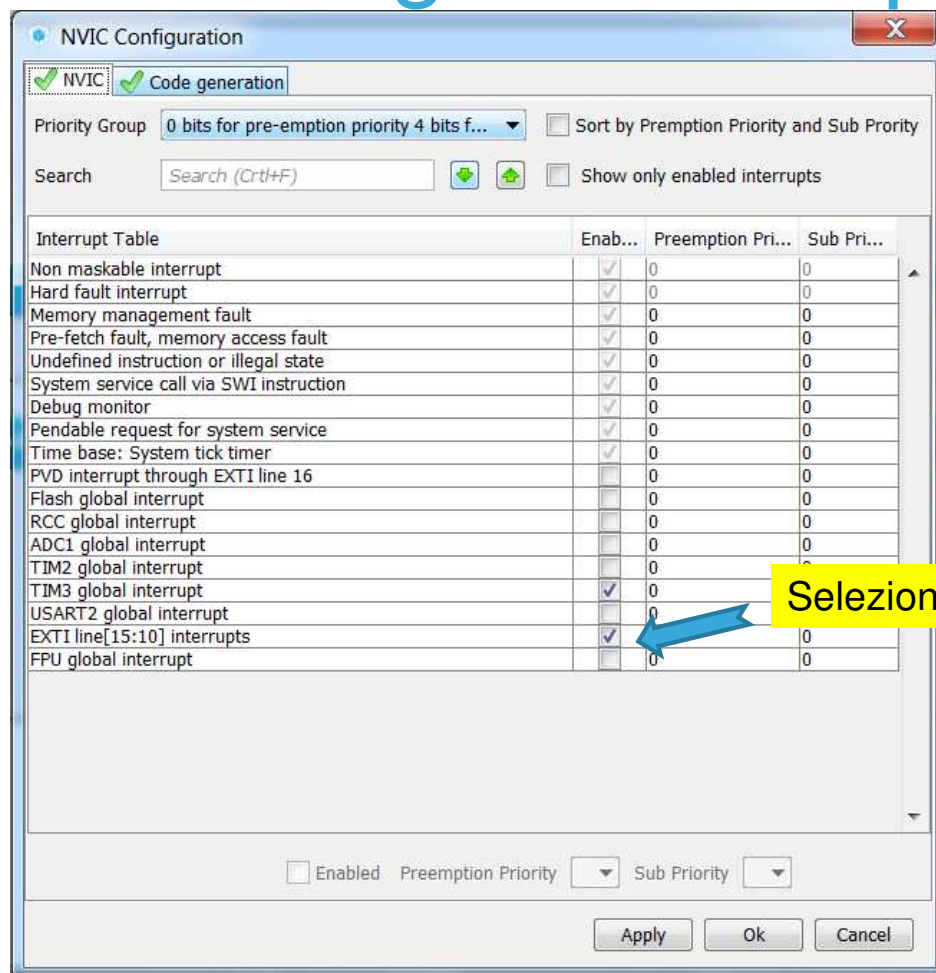
Cube MX: configurazione periferiche

23



Cube MX: configurazione periferiche

24



Selezionare EXTI line

Cube MX: generare il progetto

25



Cube MX: generare il progetto

26

Project Settings

Project Name: Test

Project Location: C:\Work\Talentis\CubeMX [Browse]

Application Main Location: Src

Toolchain Folder Location: C:\Work\Talentis\CubeMX\Test

Toolchain / IDE: SW4STM32 ☒ Generate Under Root

Linker Settings

Minimum Heap Size: 0x200

Minimum Stack Size: 0x400

Mcu and Firmware Package

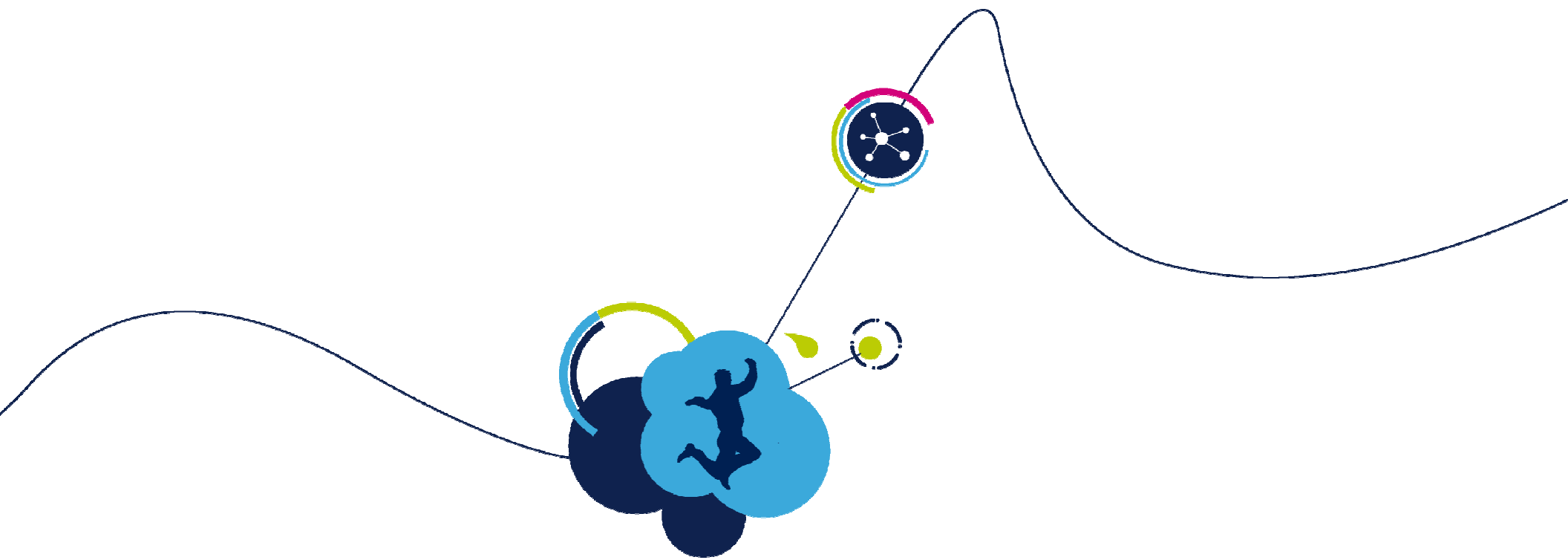
Mcu Reference: STM32F401RETx

Firmware Package Name and Version: STM32Cube FW_F4 V1.21.0

Ok Cancel

Selezione IDE
SW4STM32

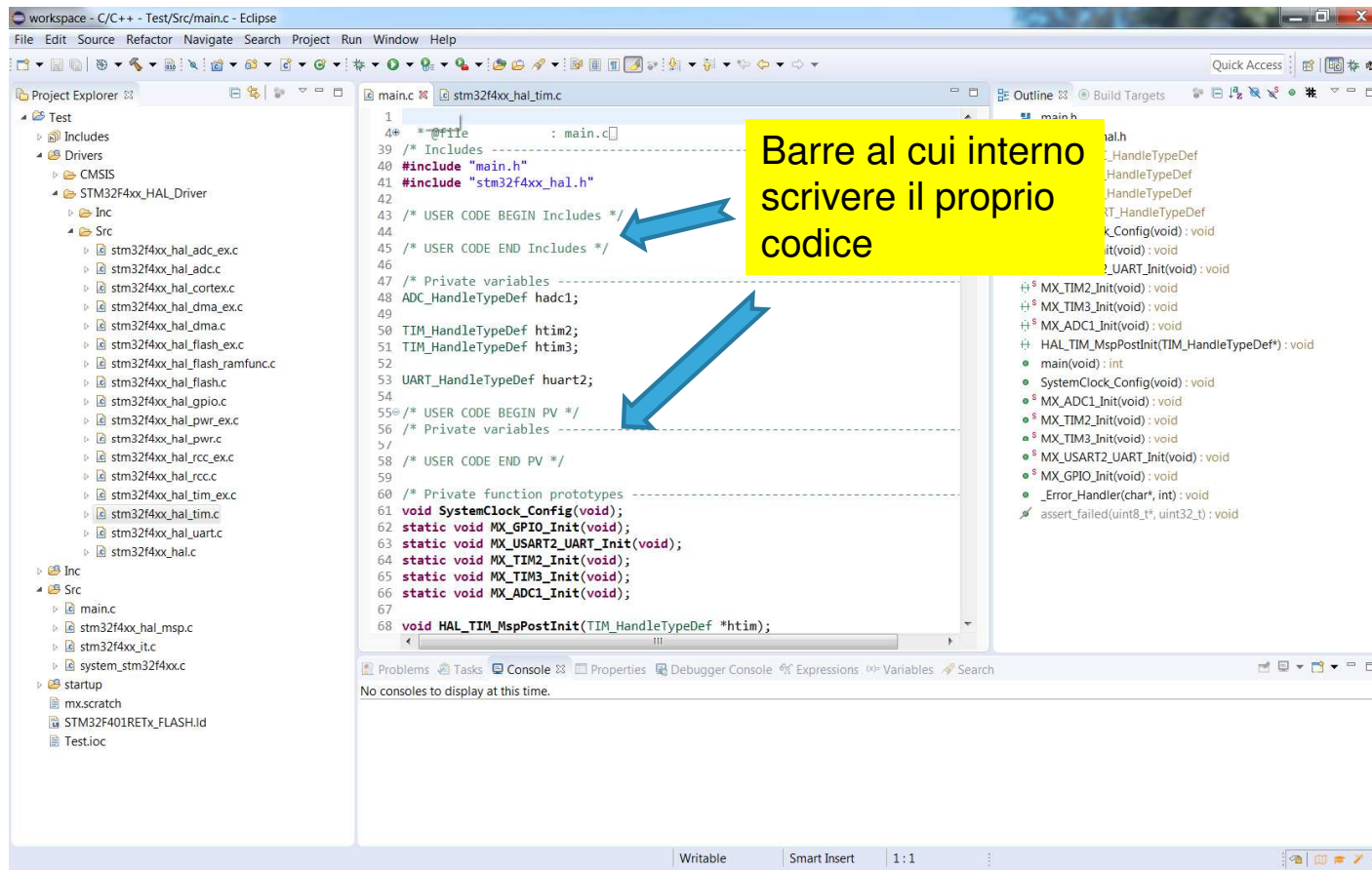
Selezione il path



SW4STM32 (AC6)

SW4STM32 IDE

28



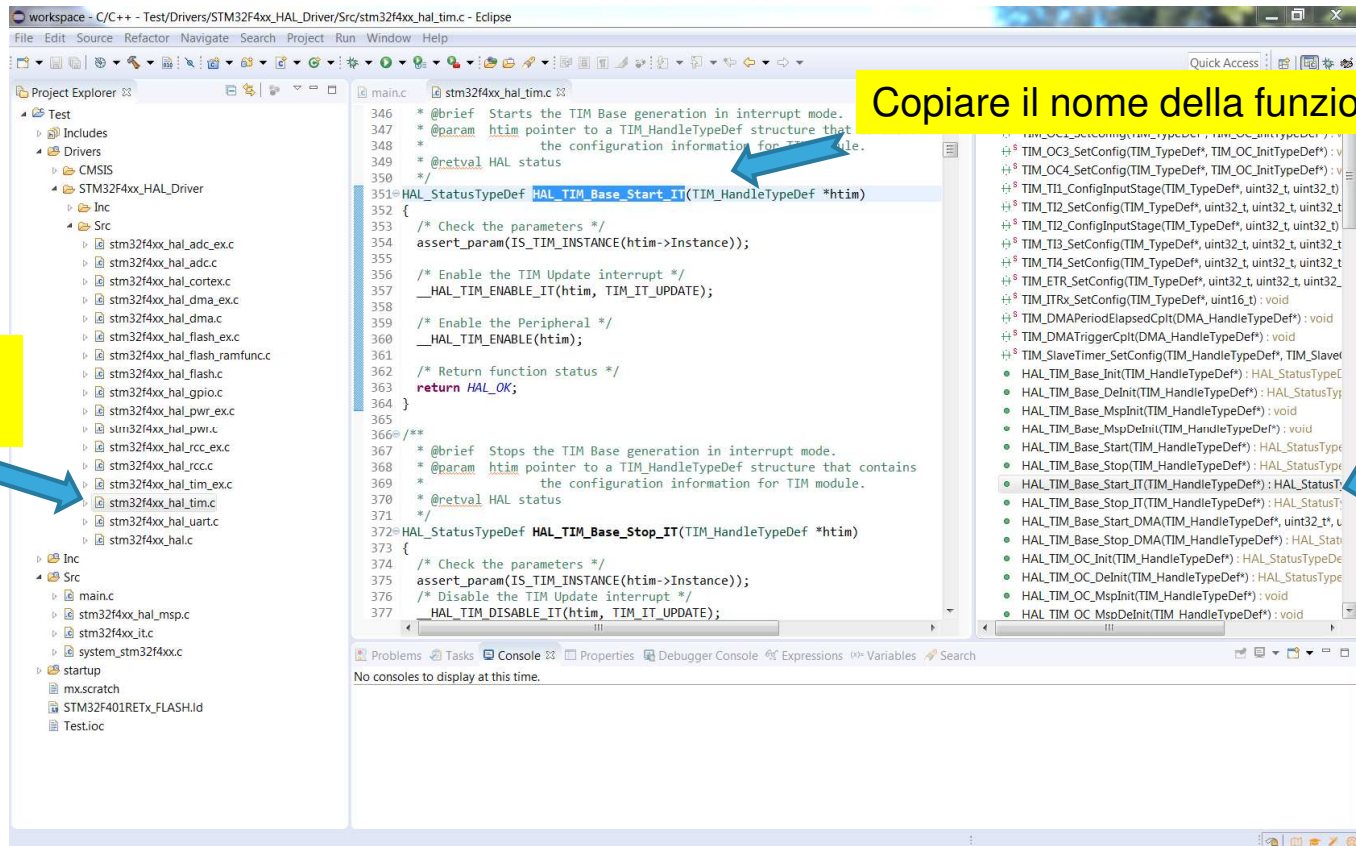
SW4STM32: come utilizzare funzioni di libreria

29

Aprire la libreria
dei Timer

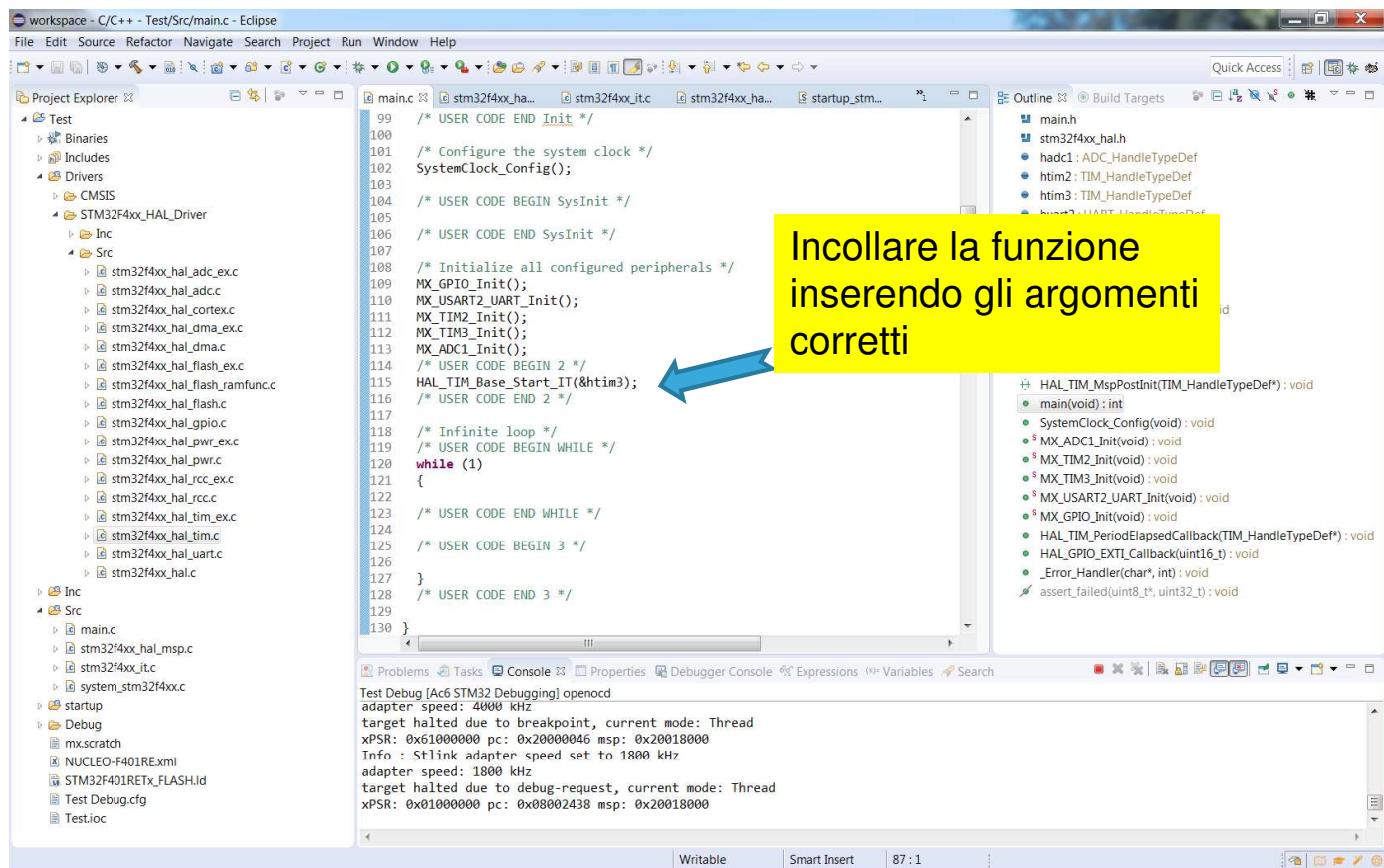
Copiare il nome della funzione

Selezionare la
funzione voluta



SW4STM32: come utilizzare funzioni di libreria

30



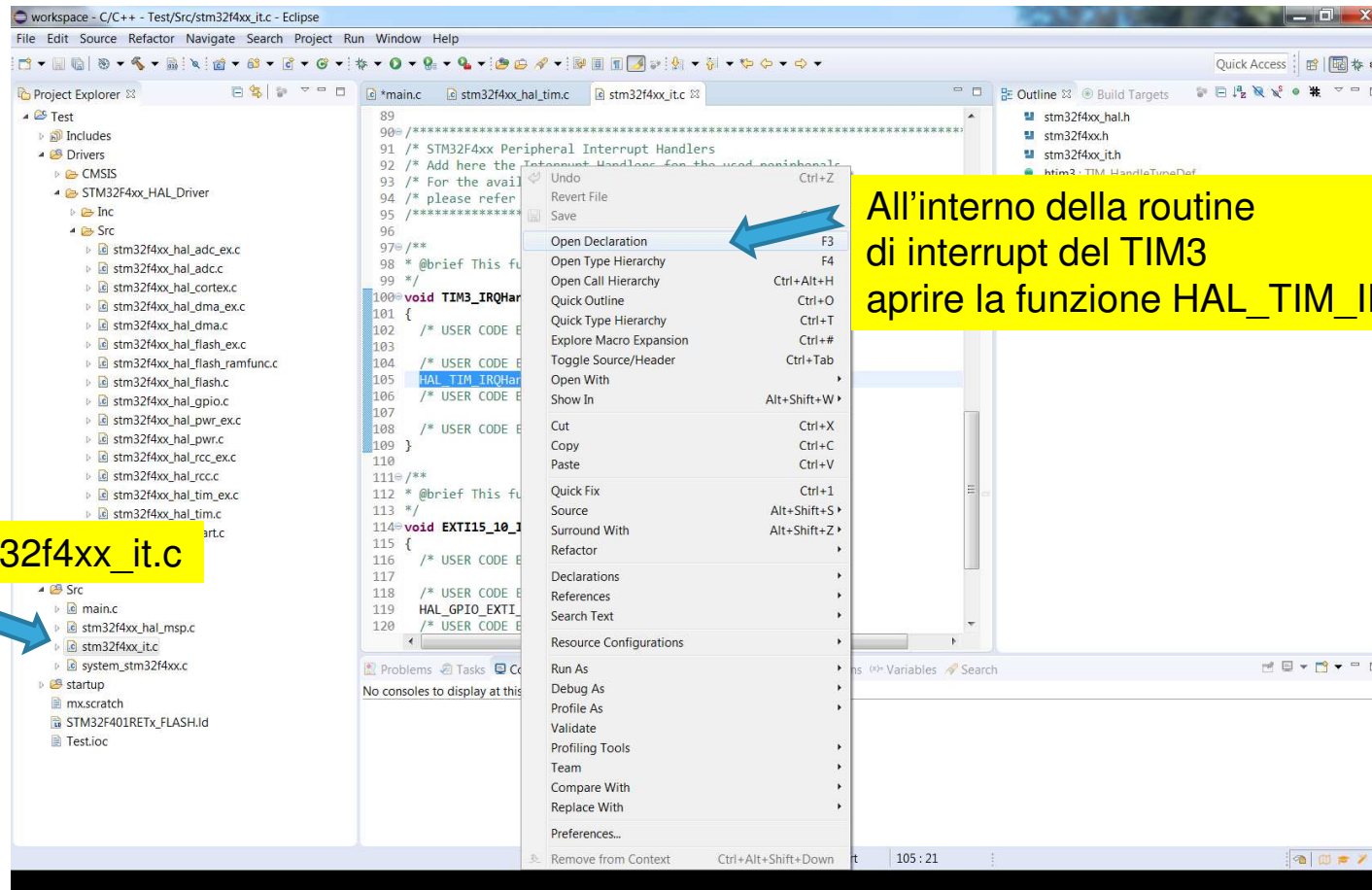
```
99  /* USER CODE END Init */
100
101  /* Configure the system clock */
102  SystemClock_Config();
103
104  /* USER CODE BEGIN SysInit */
105
106  /* USER CODE END SysInit */
107
108  /* Initialize all configured peripherals */
109  MX_GPIO_Init();
110  MX_USART2_UART_Init();
111  MX_TIM2_Init();
112  MX_TIM3_Init();
113  MX_ADC1_Init();
114  /* USER CODE BEGIN 2 */
115  HAL_TIM_Base_Start_IT(&htim3);
116  /* USER CODE END 2 */
117
118  /* Infinite loop */
119  /* USER CODE BEGIN WHILE */
120  while (1)
121  {
122
123  /* USER CODE END WHILE */
124
125  /* USER CODE BEGIN 3 */
126
127  }
128  /* USER CODE END 3 */
129
130 }
```

Incollare la funzione inserendo gli argomenti corretti

Test Debug [Ac6 STM32 Debugging] opened
adapter speed: 4000 kHz
target halted due to breakpoint, current mode: Thread
xPSR: 0x61000000 pc: 0x20000046 msp: 0x20018000
Info : Stlink adapter speed set to 1800 kHz
adapter speed: 1800 kHz
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x08002438 msp: 0x20018000

SW4STM32: Interrupt Routine

31



All'interno della routine di interrupt del TIM3 aprire la funzione HAL_TIM_IRQHandler

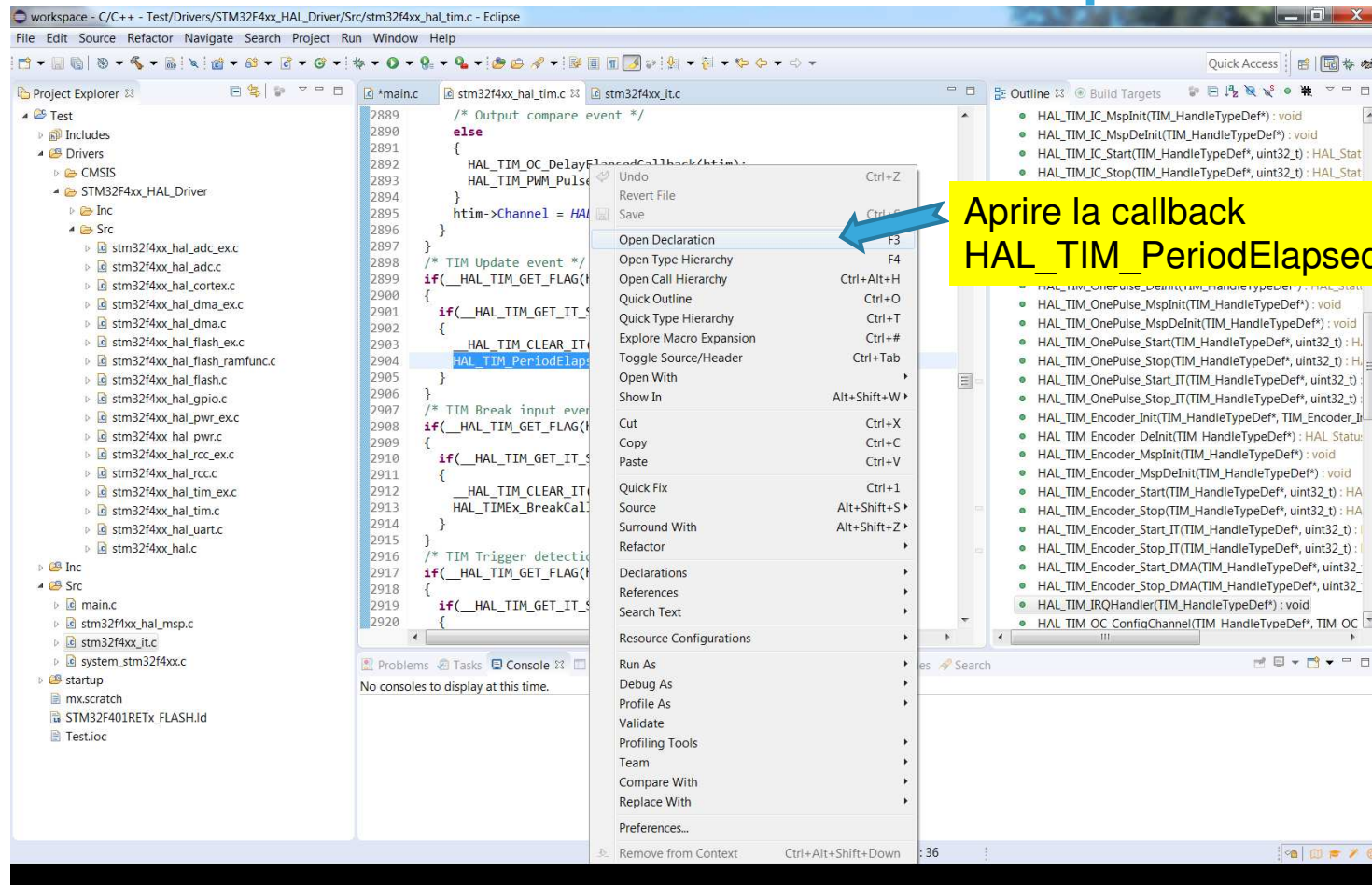
Aprire il file stm32f4xx_it.c

32



SW4STM32: Interrupt Routine

33



SW4STM32: Interrupt Routine

34

workspace - C/C++ - Test/Drivers/STM32F4xx_HAL_Driver/Src/stm32f4xx_hal_tim.c - Eclipse

File Edit Source Refactor Navigate Search Project Run Window Help

Project Explorer

- Test
 - Includes
 - Drivers
 - CMSIS
 - STM32F4xx_HAL_Driver
 - Inc
 - Src
 - stm32f4xx_hal_adc_exc.c
 - stm32f4xx_hal_adc.c
 - stm32f4xx_hal_cortex.c
 - stm32f4xx_hal_dma_exc.c
 - stm32f4xx_hal_dma.c
 - stm32f4xx_hal_flash_exc.c
 - stm32f4xx_hal_flash_ramfunc.c
 - stm32f4xx_hal_flash.c
 - stm32f4xx_hal_gpio.c
 - stm32f4xx_hal_pwr_exc.c
 - stm32f4xx_hal_pwr.c
 - stm32f4xx_hal_rcc_exc.c
 - stm32f4xx_hal_rcc.c
 - stm32f4xx_hal_tim_exc.c
 - stm32f4xx_hal_tim.c
 - stm32f4xx_hal_uart.c
 - stm32f4xx_hal.c
 - Inc
 - Src
 - main.c
 - stm32f4xx_hal_msp.c
 - stm32f4xx_it.c
 - system_stm32f4xx.c
 - startup
 - mx.scratch
 - STM32F401RETx_FLASH.ld
 - TestIoc

main.c stm32f4xx_hal_tim.c stm32f4xx_it.c

```
4294 @endverbatim
4295 * @{
4296 */
4297
4298 /**
4299 * @brief Period elapsed callback in non blocking mode
4300 * @param htim pointer to a TIM_HandleTypeDef structure that contains
4301 *         the configuration information for TIM module.
4302 * @retval None
4303 */
4304 __weak void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
4305 {
4306     /* Prevent unused argument(s) compilation warning */
4307     UNUSED(htim);
4308     /* NOTE : This function Should not be modified, when the callback is needed,
4309     the __HAL_TIM_PeriodElapsedCallback could be implemented in the user file.
4310     */
4311 }
4312
4313 /**
4314 * @brief Output Compare callback in non blocking mode
4315 * @param htim pointer to a TIM_HandleTypeDef structure that contains
4316 *         the configuration information for TIM module.
4317 * @retval None
4318 */
4319 __weak void HAL_TIM_OC_DelayElapsedCallback(TIM_HandleTypeDef *htim)
4320 {
4321     /* Prevent unused argument(s) compilation warning */
4322     UNUSED(htim);
4323     /* NOTE : This function Should not be modified, when the callback is needed,
4324     the __HAL_TIM_OC_DelayElapsedCallback could be implemented in the user file.
4325     */
4326 }
```

Outline

- HAL_TIM_Encoder_Init(TIM_HandleTypeDef*, TIM_Encoder_InitTypeDef*)
- HAL_TIM_Encoder_DeInit(TIM_HandleTypeDef*)
- HAL_TIM_Encoder_MspInit(TIM_HandleTypeDef*)
- HAL_TIM_Encoder_Start(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_Encoder_Stop_DMA(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_OC_Init(TIM_HandleTypeDef*, TIM_OC_InitTypeDef*)
- HAL_TIM_OC_DeInit(TIM_HandleTypeDef*)
- HAL_TIM_OC_MspInit(TIM_HandleTypeDef*)
- HAL_TIM_OC_Start(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_OC_Stop(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_PWM_Init(TIM_HandleTypeDef*, TIM_PWM_InitTypeDef*)
- HAL_TIM_PWM_DeInit(TIM_HandleTypeDef*)
- HAL_TIM_PWM_MspInit(TIM_HandleTypeDef*)
- HAL_TIM_PWM_Start(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_PWM_Stop(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_OnePulse_Init(TIM_HandleTypeDef*, TIM_OnePulse_InitTypeDef*)
- HAL_TIM_OnePulse_Start(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_OnePulse_Stop(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_DMA_Init(TIM_HandleTypeDef*, TIM_DMA_InitTypeDef*)
- HAL_TIM_DMA_DeInit(TIM_HandleTypeDef*)
- HAL_TIM_DMA_MspInit(TIM_HandleTypeDef*)
- HAL_TIM_DMA_Start(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_DMA_Stop(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_DMA_Start_IT(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_DMA_Stop_IT(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_GenerateEvent(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_ConfigOCrefClear(TIM_HandleTypeDef*, TIM_ClearTypeDef*)
- HAL_TIM_ConfigClockSource(TIM_HandleTypeDef*, TIM_ClockConfigTypeDef*)
- HAL_TIM_ConfigTI1Input(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_SlaveConfigSynchronization(TIM_HandleTypeDef*, TIM_SlaveConfigTypeDef*)
- HAL_TIM_SlaveConfigSynchronization_IT(TIM_HandleTypeDef*, TIM_SlaveConfigTypeDef*)
- HAL_TIM_ReadCapturedValue(TIM_HandleTypeDef*, uint32_t)
- HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef*)
- HAL_TIM_OC_DelayElapsedCallback(TIM_HandleTypeDef*)

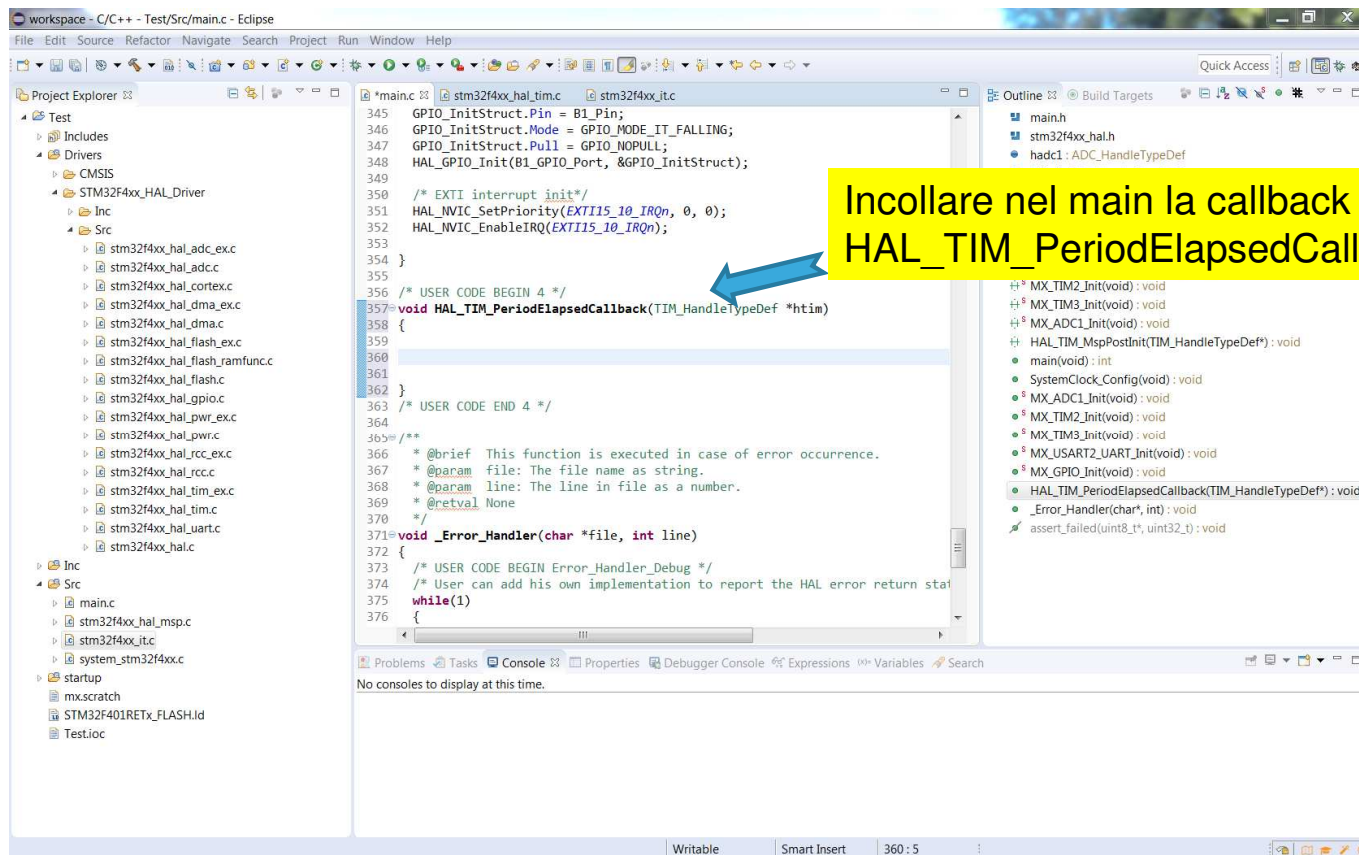
Problems Tasks Console Properties Debugger Console Expressions Variables Search

No consoles to display at this time.

Writable Smart Insert 4304 : 42

SW4STM32: Interrupt Routine

35



```
workspace - C/C++ - Test/Src/main.c - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help

Project Explorer
Test
Includes
Drivers
CMSIS
STM32F4xx_HAL_Driver
Inc
Src
stm32f4xx_hal_adc_exc
stm32f4xx_hal_adc.c
stm32f4xx_hal_cortex.c
stm32f4xx_hal_dma_exc
stm32f4xx_hal_dma.c
stm32f4xx_hal_flash_exc
stm32f4xx_hal_flash.c
stm32f4xx_hal_gpio.c
stm32f4xx_hal_pwr_exc
stm32f4xx_hal_pwr.c
stm32f4xx_hal_rcc_exc
stm32f4xx_hal_rcc.c
stm32f4xx_hal_tim_exc
stm32f4xx_hal_tim.c
stm32f4xx_hal_uart.c
stm32f4xx_hal.c
Inc
Src
main.c
stm32f4xx_hal_msp.c
stm32f4xx_it.c
system_stm32f4xx.c
startup
mxscratch
STM32F401RETx_FLASH.ld
TestIoc

main.c
345 GPIO_InitStruct.Pin = B1_Pin;
346 GPIO_InitStruct.Mode = GPIO_MODE_IT_FALLING;
347 GPIO_InitStruct.Pull = GPIO_NOPULL;
348 HAL_GPIO_Init(B1_GPIO_Port, &GPIO_InitStruct);
349
350 /* EXTI interrupt init*/
351 HAL_NVIC_SetPriority(EXTI15_10_IRQn, 0, 0);
352 HAL_NVIC_EnableIRQ(EXTI15_10_IRQn);
353
354 }
355
356 /* USER CODE BEGIN 4 */
357 void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
358 {
359
360 }
361
362 }
363 /* USER CODE END 4 */
364
365 /**
366  * @brief This function is executed in case of error occurrence.
367  * @param file: The file name as string.
368  * @param line: The line in file as a number.
369  * @retval None
370  */
371 void _Error_Handler(char *file, int line)
372 {
373 /* USER CODE BEGIN Error_Handler_Debug */
374 /* User can add his own implementation to report the HAL error return state */
375 while(1)
376 {
377
378 }
379 }

Outline
main.h
stm32f4xx_hal.h
hdc1 : ADC_HandleTypeDef

MX_TIM2_Init(void) : void
MX_TIM3_Init(void) : void
MX_ADC1_Init(void) : void
HAL_TIM_MspPostInit(TIM_HandleTypeDef*) : void
main(void) : int
SystemClock_Config(void) : void
MX_ADC1_Init(void) : void
MX_TIM2_Init(void) : void
MX_TIM3_Init(void) : void
MX_USART2_UART_Init(void) : void
MX_GPIO_Init(void) : void
HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef*) : void
_Error_Handler(char*, int) : void
assert_failed(uint8_t*, uint32_t) : void

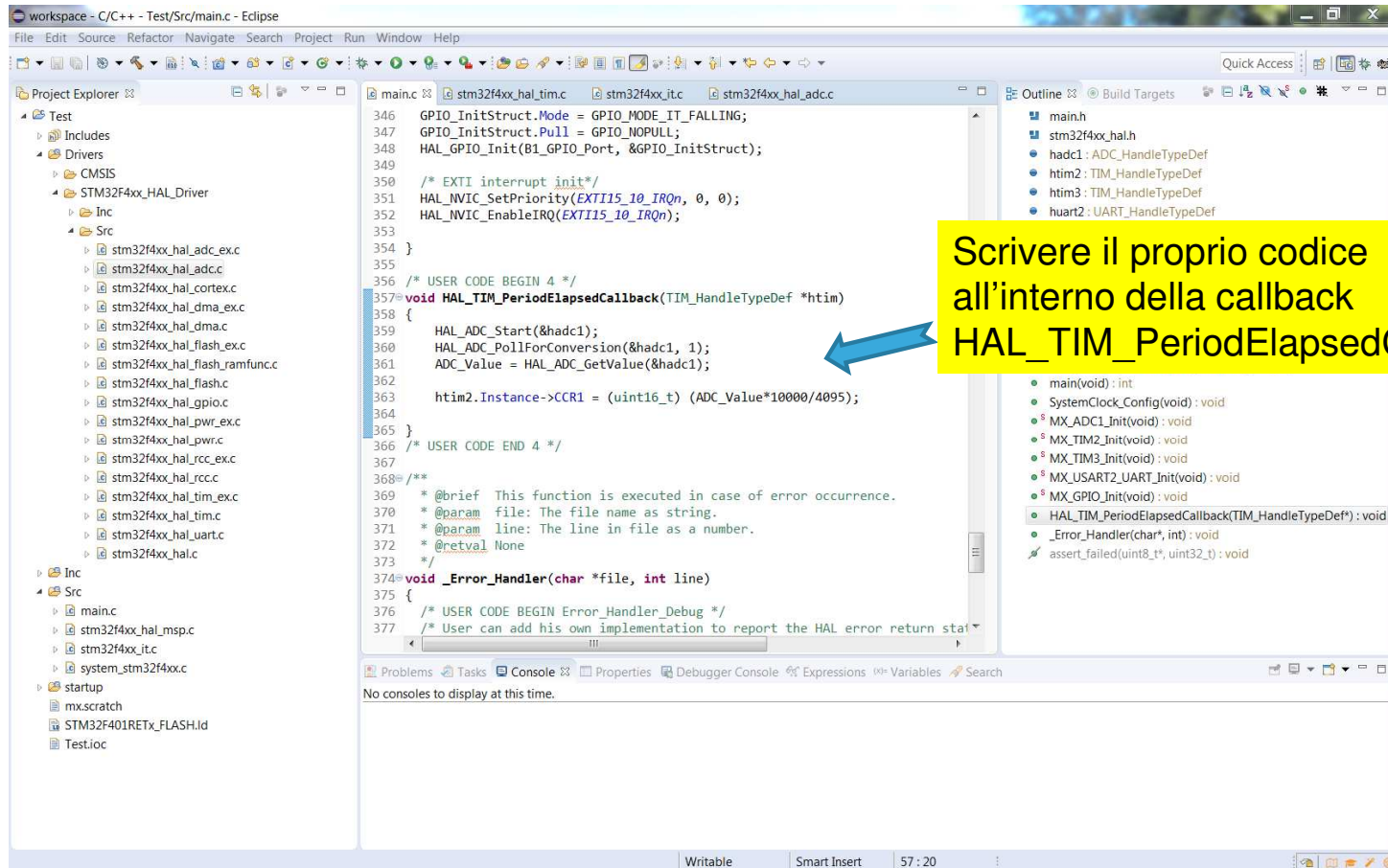
Problems Tasks Console Properties Debugger Console Expressions Variables Search
No consoles to display at this time.

Writable Smart Insert 360 : 5
```

Incollare nel main la callback
HAL_TIM_PeriodElapsedCallback

SW4STM32: Interrupt Routine

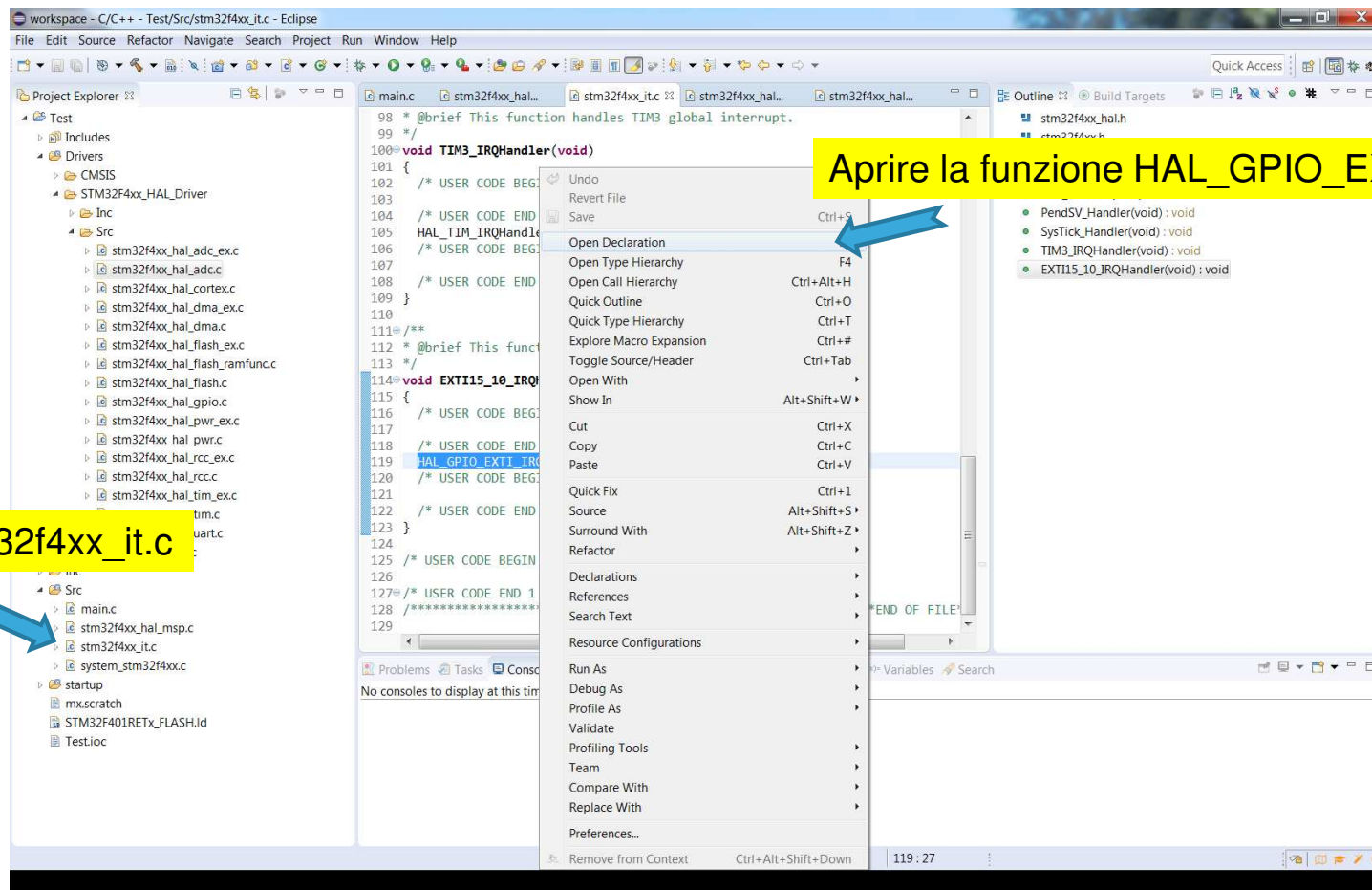
36



```
346 GPIO_InitStruct.Mode = GPIO_MODE_IT_FALLING;
347 GPIO_InitStruct.Pull = GPIO_NOPULL;
348 HAL_GPIO_Init(B1_GPIO_Port, &GPIO_InitStruct);
349
350 /* EXTI interrupt init*/
351 HAL_NVIC_SetPriority(EXTI15_10_IRQn, 0, 0);
352 HAL_NVIC_EnableIRQ(EXTI15_10_IRQn);
353
354 }
355
356 /* USER CODE BEGIN 4 */
357 void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
358 {
359     HAL_ADC_Start(&hadc1);
360     HAL_ADC_PollForConversion(&hadc1, 1);
361     ADC_Value = HAL_ADC_GetValue(&hadc1);
362
363     htim2.Instance->CCR1 = (uint16_t) (ADC_Value*10000/4095);
364 }
365 /* USER CODE END 4 */
366
367 /**
368  * @brief This function is executed in case of error occurrence.
369  * @param file: The file name as string.
370  * @param line: The line in file as a number.
371  * @retval None
372  */
373 void _Error_Handler(char *file, int line)
374 {
375     /* USER CODE BEGIN Error_Handler_Debug */
376     /* User can add his own implementation to report the HAL error return state */
377 }
```

SW4STM32: Interrupt Routine

37



SW4STM32: Interrupt Routine

38

```
workspace - C/C++ - Test/Drivers/STM32F4xx_HAL_Driver/Src/stm32f4xx_hal_gpio.c - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help

Project Explorer
Test
  Includes
  Drivers
    CMSIS
    STM32F4xx_HAL_Driver
      Inc
      Src
        stm32f4xx_hal_adc_exc.c
        stm32f4xx_hal_adc.c
        stm32f4xx_hal_cortex.c
        stm32f4xx_hal_dma_exc.c
        stm32f4xx_hal_dma.c
        stm32f4xx_hal_flash_exc.c
        stm32f4xx_hal_flash_ramfunc.c
        stm32f4xx_hal_flash.c
        stm32f4xx_hal_gpio.c
        stm32f4xx_hal_pwr_exc.c
        stm32f4xx_hal_pwr.c
        stm32f4xx_hal_rcc_exc.c
        stm32f4xx_hal_rcc.c
        stm32f4xx_hal_tim_exc.c
        stm32f4xx_hal_tim.c
        stm32f4xx_hal_uart.c
        stm32f4xx_hal.c
      Inc
      Src
        main.c
        stm32f4xx_hal_msp.c
        stm32f4xx_it.c
        system_stm32f4xx.c
      startup
        mx_scratch
        STM32F401RETx_FLASH.ld
        TestIoc

main.c stm32f4xx_hal... stm32f4xx_it.c stm32f4xx_hal... stm32f4xx_hal...
499  /* This function handles EXTI interrupt request.
500  * @param GPIO_Pin Specifies the pins connected EXTI line
501  * @retval None
502  */
503  void HAL_GPIO_EXTI_IRQHandler(uint16_t GPIO_Pin)
504  {
505  /* EXTI line interrupt detected */
506  if(__HAL_GPIO_EXTI_GET_IT(GPIO_Pin) != RESET)
507  {
508  __HAL_GPIO_EXTI_CLEAR_IT(GPIO_Pin);
509  HAL_GPIO_EXTI_Callback(GPIO_Pin);
510  }
511 }
512
513 /**
514  * @brief EXTI line detection callbacks.
515  * @param GPIO_Pin Specifies the pins connected EXTI line
516  * @retval None
517  */
518  __weak void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
519  {
520  /* Prevent unused argument(s) compilation warning */
521  UNUSED(GPIO_Pin);
522  /* NOTE: This function Should not be modified, when the callback is needed
523  the HAL_GPIO_EXTI_Callback could be implemented in the user file
524  */
525 }
526
527 /**
528  * @}
529  */
530
```

Aprire la callback
HAL_GPIO_EXTI_IRQHandler

Copiare la callback
HAL_GPIO_EXTI_IRQHandler

SW4STM32: Interrupt Routine

39

workspace - C/C++ - Test/Src/main.c - Eclipse

Project Explorer

- Test
 - Includes
 - Drivers
 - CMSIS
 - STM32F4xx_HAL_Driver
 - Inc
 - Src
 - stm32f4xx_hal_adc_exc.c
 - stm32f4xx_hal_adc.c
 - stm32f4xx_hal_cortex.c
 - stm32f4xx_hal_dma_exc.c
 - stm32f4xx_hal_dma.c
 - stm32f4xx_hal_flash_exc.c
 - stm32f4xx_hal_flash_ramfunc.c
 - stm32f4xx_hal_flash.c
 - stm32f4xx_hal_gpio.c
 - stm32f4xx_hal_pwr_exc.c
 - stm32f4xx_hal_pwr.c
 - stm32f4xx_hal_rcc_exc.c
 - stm32f4xx_hal_rcc.c
 - stm32f4xx_hal_tim_exc.c
 - stm32f4xx_hal_tim.c
 - stm32f4xx_hal_uart.c
 - stm32f4xx_hal.c
- Inc
- Src
 - main.c
 - stm32f4xx_hal_msp.c
 - stm32f4xx_it.c
 - system_stm32f4xx.c
- startup
 - mx.scratch
 - STM32F401RETx_FLASH.ld
 - Test.ioc

main.c

```
358 void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
359 {
360     HAL_ADC_Start(&hadc1);
361     HAL_ADC_PollForConversion(&hadc1, 1);
362     ADC_Value = HAL_ADC_GetValue(&hadc1);
363
364     htim2.Instance->CCR1 = (uint16_t) (ADC_Value*10000/4095);
365 }
366
367 void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
368 {
369     if (flag == 0)
370     {
371         flag = 1;
372         HAL_TIM_PWM_Start(&htim2, TIM_CHANNEL_1);
373     }
374     else
375     {
376         flag = 0;
377         HAL_TIM_PWM_Stop(&htim2, TIM_CHANNEL_1);
378     }
379 }
380
381 /* USER CODE END 4 */
382
383 /**
384  * @brief This function is executed in case of error occurrence.
385  * @param file: The file name as string.
386  * @param line: The line in file as a number.
387  */
388 void Error_Handler(void)
389 {
390     /* USER CODE BEGIN Error_Handler_Debug */
391     /* User can add long list of diagnostic message, examples:
392     * printf("Error in %s line %d", file, line);
393     * HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin, GPIO_PIN_RESET);
394     * ...
395     */
396 }
```

Outline

- main.h
- stm32f4xx_hal.h
- MX_USART2_UART_Init(void): void
- MX_TIM2_Init(void): void
- MX_TIM3_Init(void): void
- MX_ADC1_Init(void): void
- HAL_TIM_MspPostInit(TIM_HandleTypeDef*): void
- main(void): int
- SystemClock_Config(void): void
- MX_ADC1_Init(void): void
- MX_TIM2_Init(void): void
- MX_TIM3_Init(void): void
- MX_USART2_UART_Init(void): void
- MX_GPIO_Init(void): void
- HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef*): void
- HAL_GPIO_EXTI_Callback(uint16_t): void
- _Error_Handler(char*, int): void
- assert_failed(uint8_t*, uint32_t): void

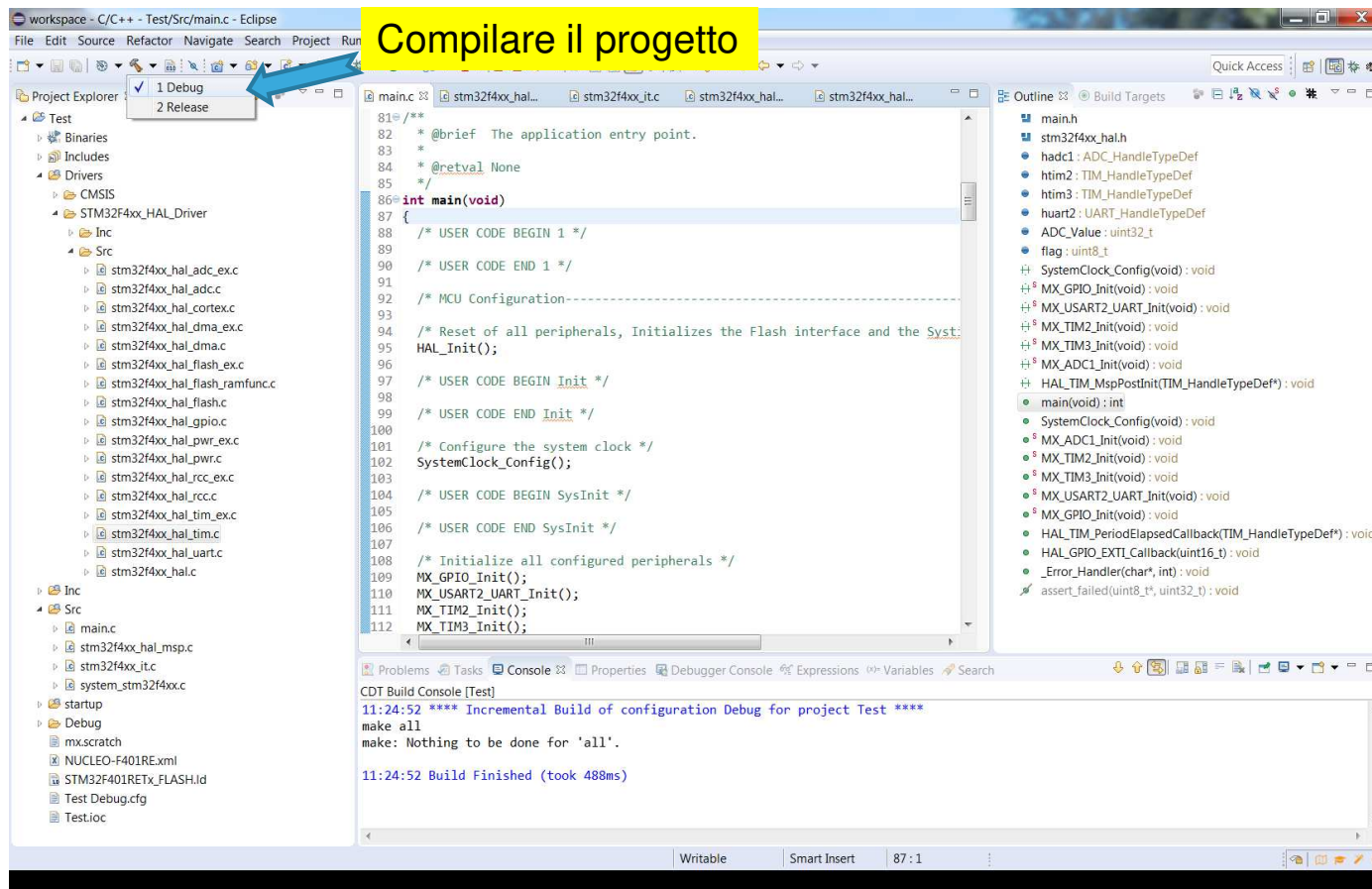
Problems Tasks Console Properties Debugger Console Expressions Variables Search

No consoles to display at this time.

Writable Smart Insert 379 : 17

SW4STM32: Compilazione

40



SW4STM32: Debug Mode

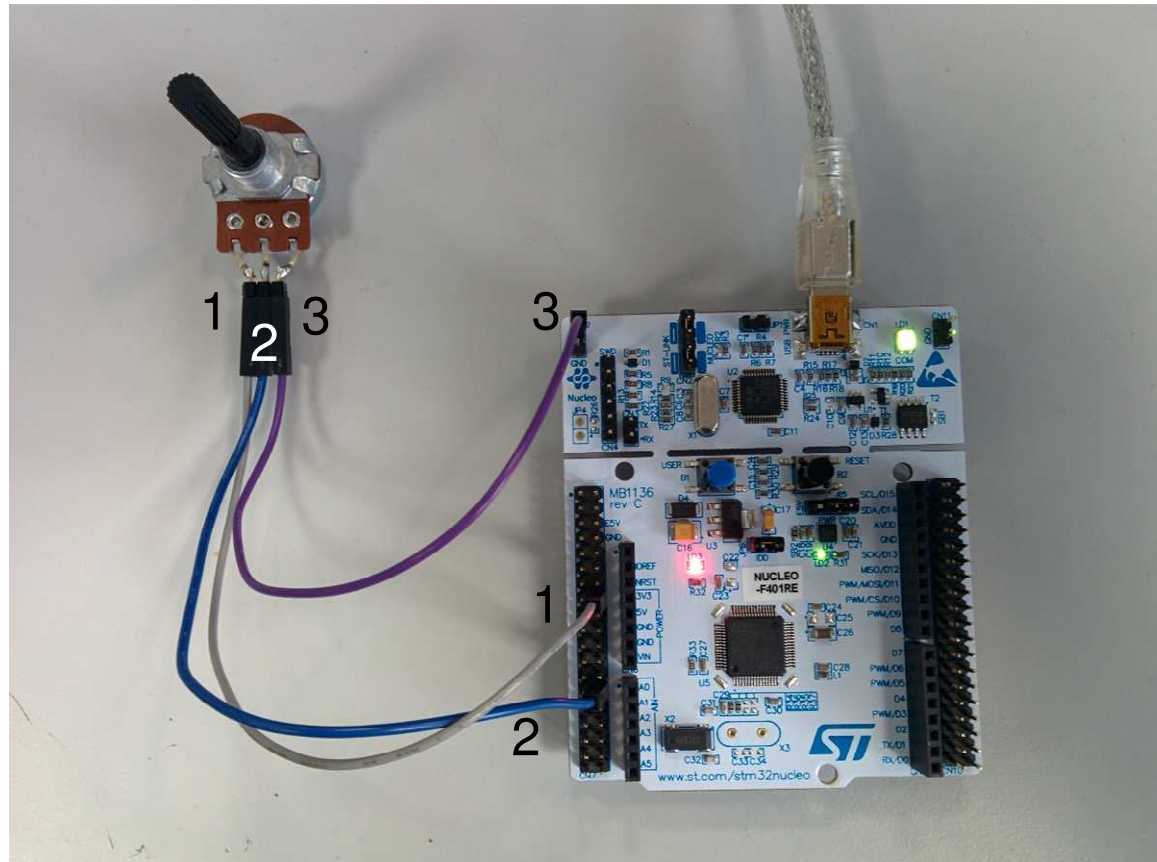
41

The screenshot shows the Eclipse IDE interface. The 'Run' menu is open, and 'Debug As' is selected, which has opened a sub-menu where '1 Ac6 STM32 C/C++ Application' is highlighted. A yellow callout box with a blue arrow pointing to this option contains the text 'Avviare il progetto In Debug Mode'. The Project Explorer on the left shows a project named 'Test' with a source folder 'Src' containing various STM32 HAL driver files. The main editor displays the 'main.c' file with C code for initializing the system and peripherals. The Console window at the bottom shows the build output for 'Test', indicating a successful incremental build.

```
CDT Build Console [Test]
11:24:52 **** Incremental Build of configuration Debug for project Test ****
make all
make: Nothing to be done for 'all'.
11:24:52 Build Finished (took 488ms)
```

SW4STM32: Connection

42



1. 3.3V => Pin 16 CN7
2. AIN0 => Pin 28 CN7
3. Ground