

Project_5.3.1_TIM_InputCapture_Stopwatch

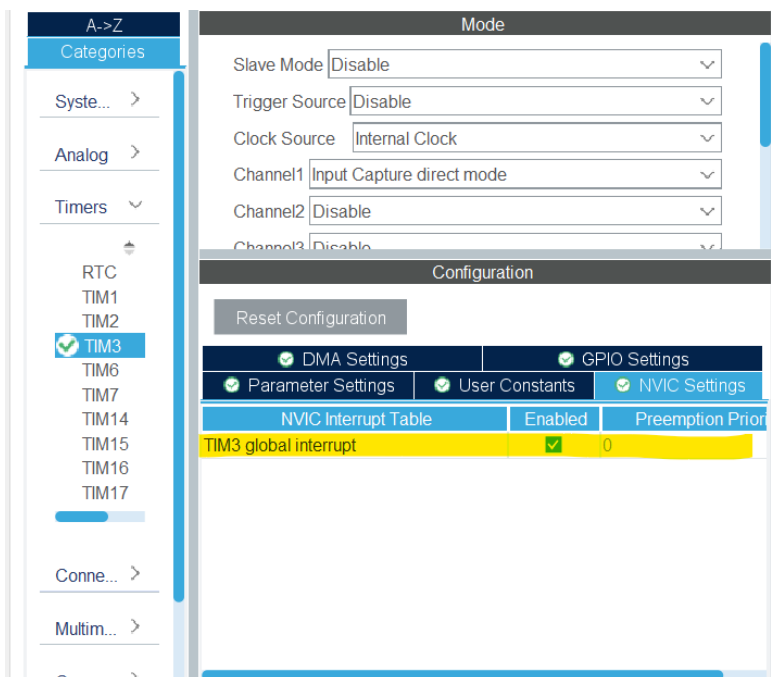
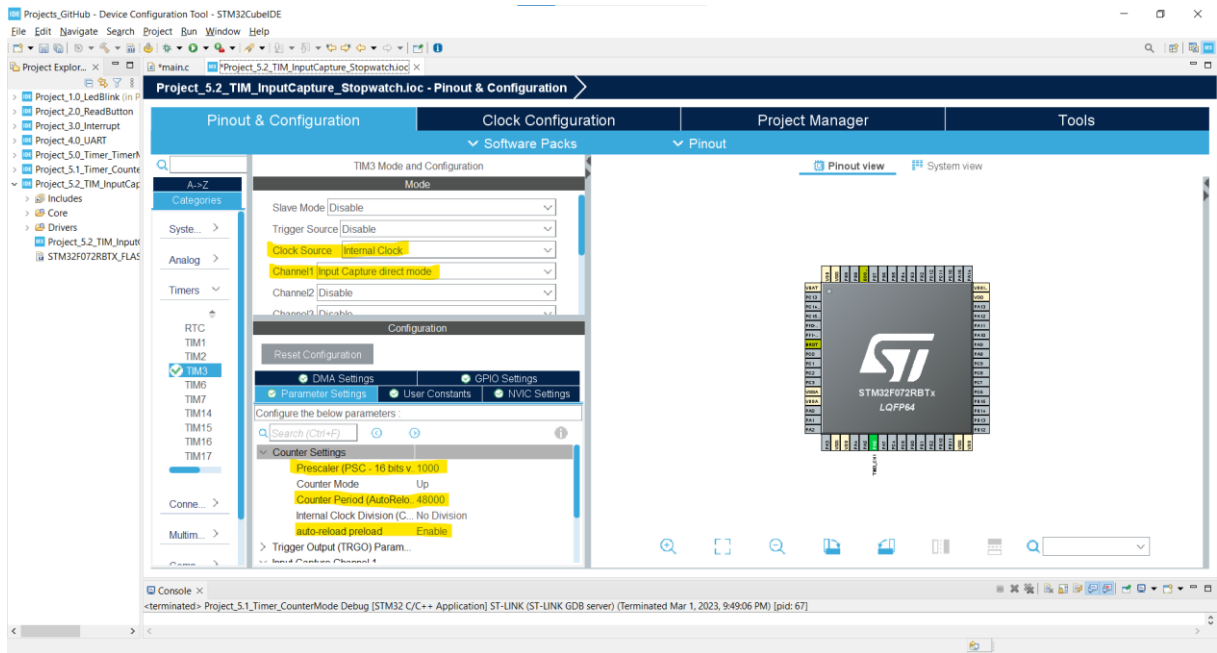
Bu projede timer ı input capture modunda kullanarak bir kronametre yapacağız.

STEP 1 : Open STM32CubeIDE and Create New Project.

STEP 2 : Select Target microcontroller and double Click. My MCU is STM32F072RBT6.

STEP3 : Enter the project name and finish .

Step 5 : Configure Timer 3:



STEP 6: We configure the uart module.

STEP 7 : Set the RCC External Clock Source and then CTRL + S to generate the project code. And we open our main.c file in the project files.

STEP 8: Code :

```
32 /* Private define ----- */
33 /* USER CODE BEGIN PD */
34 #define IDLE 0
35 #define DONE 1
36
37 volatile uint8_t State = IDLE;
38 volatile uint8_t MSG[100] = {'\0'};
39 volatile uint8_t i_MSG[50] = {'\0'};
40 volatile uint8_t msg[100] = {"Stopwatch started\n\n\n"};
41 volatile uint32_t first_cptr_vrbl = 0;
42 volatile uint32_t last_cptr_vrbl = 0;
43 volatile uint32_t total_second = 0;
44 volatile uint16_t overflow = 0;
45 volatile uint8_t ticks = IDLE;
46 volatile uint8_t flag = IDLE;
47 volatile uint8_t second = 0;
48 volatile uint8_t minute = 0;
49 volatile uint8_t hours = 0;
50 /* USER CODE END PD */

77 void instantClock(void){
78     last_cptr_vrbl = TIM3->CCR1;
79     total_second = ((last_cptr_vrbl + (overflow * 48000)) - first_cptr_vrbl) / 48000;
80
81     minute = (total_second/60);
82     hours = minute / 60;
83
84     second = total_second - minute*60;
85     minute = minute - hours*60;
86
87 }
88
90 void HAL_TIM_IC_CaptureCallback(TIM_HandleTypeDef *htim) {
91     HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_8);
92
93     switch (ticks){
94     case IDLE:
95         HAL_UART_Transmit(&huart2, msg, sizeof(msg), 100);
96         first_cptr_vrbl = TIM3->CCR1;
97         overflow = 0;
98         ticks = DONE;
99         flag=DONE;
100         break;
101     case DONE:
102         flag = IDLE;
103         instantClock();
104         if (total_second != 0) {
105             sprintf(MSG, "Stopwatch ended\nPassing time = %lu:%lu:%lu\n\n\n", hours,minute,second);
106             HAL_UART_Transmit(&huart2, MSG, sizeof(MSG), 100);
107         }
108         first_cptr_vrbl = 0;
109         last_cptr_vrbl = 0;
110         ticks = IDLE;
111         break;
112     }
113 }
114
119 void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim) {
120
121     instantClock();
122     if(flag == DONE){
123         printf(i_MSG, "**** %lu:%lu:%lu ****\n\n", hours,minute,second);
124         HAL_UART_Transmit(&huart2, i_MSG, sizeof(i_MSG), 100);
125     }
126     overflow++;
127 }
128
```

```
162 HAL_TIM_Base_Start_IT(&htim3);
163 HAL_TIM_IC_Start_IT(&htim3, TIM_CHANNEL_1);
164 HAL_GPIO_WritePin(GPIOA, GPIO_PIN_8, GPIO_PIN_SET);
```

That's it.

