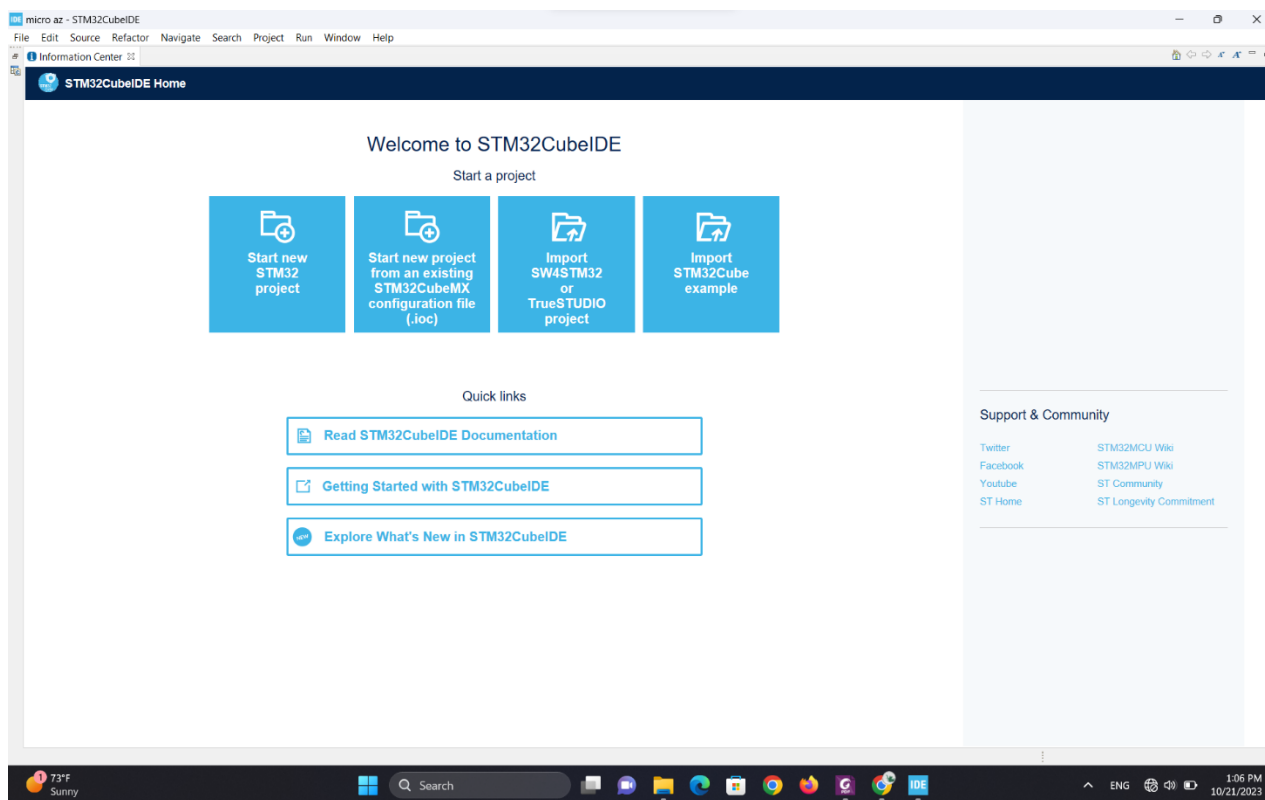


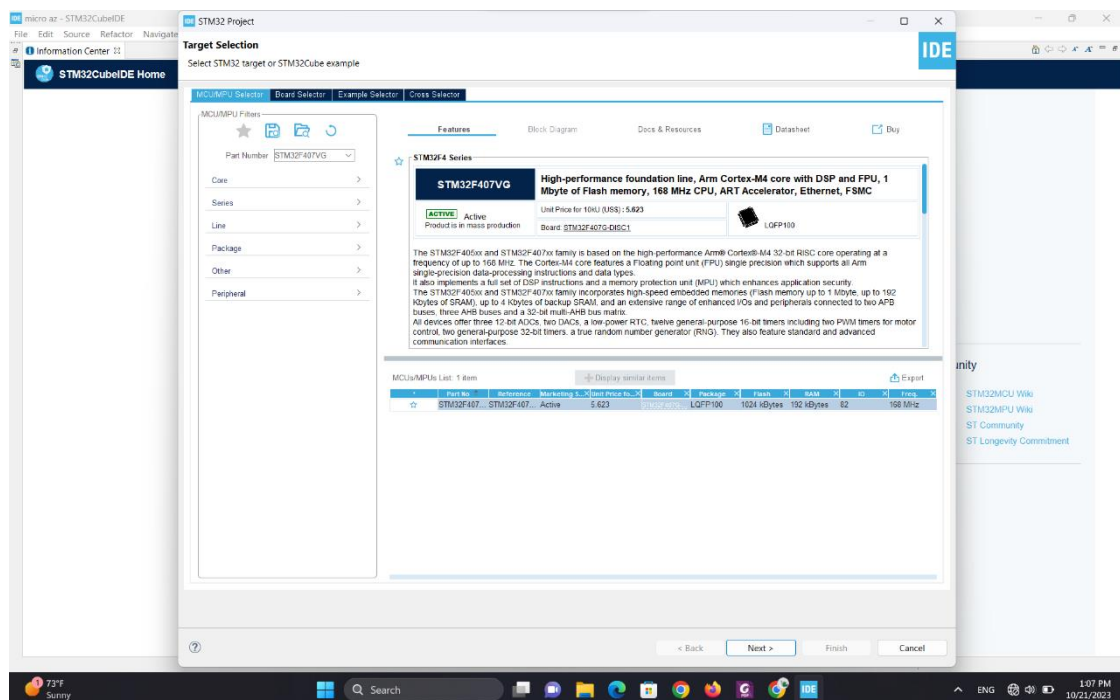
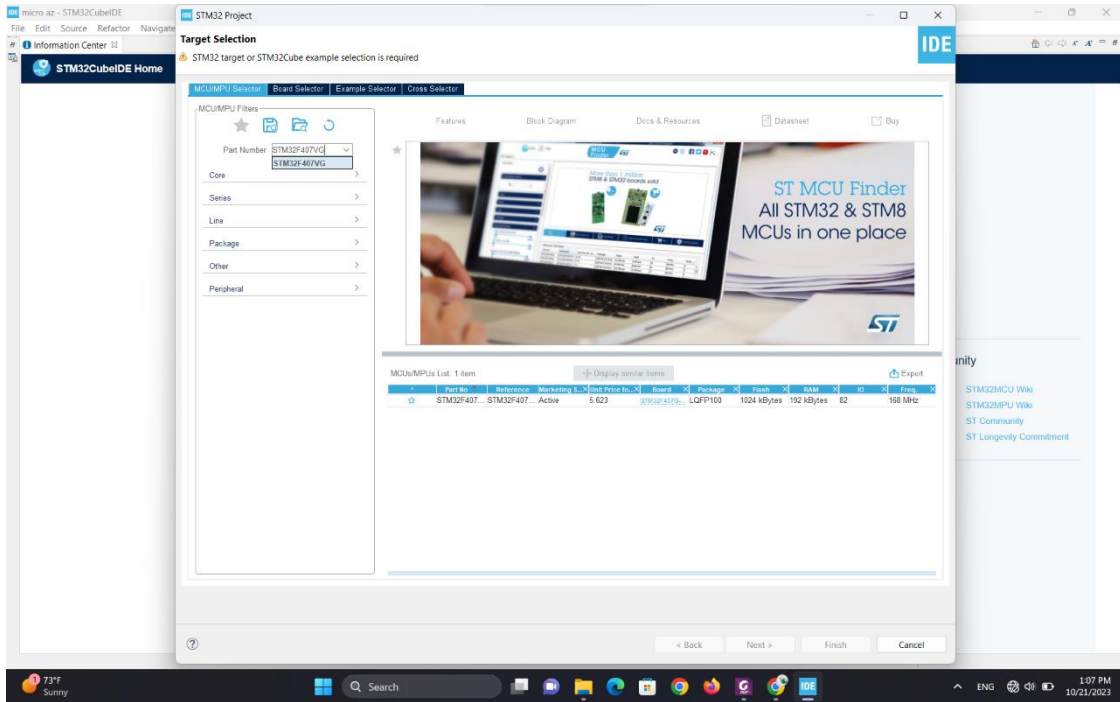
مهشاد اکبری سریزدی 9923093

آنوشا شریعتی 9923041

1- در ابتدا همان طور که در گزارش کار خواسته شده است یک work space جدید مانند شکل زیر ایجاد می کنیم.

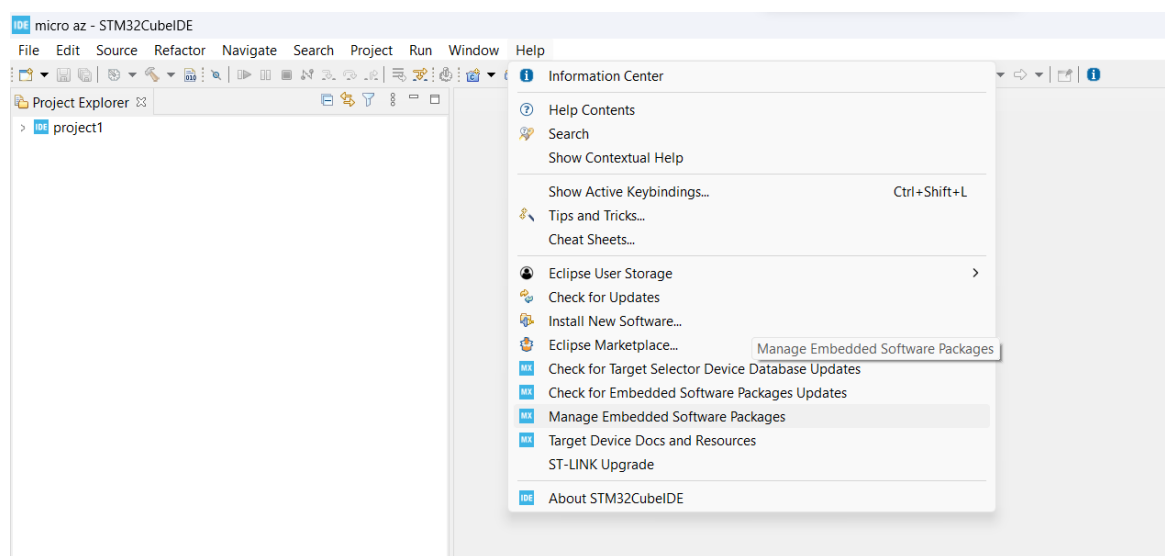


2- حال به انتخاب IC خود مطابق خواسته دستور کار می پردازیم در اینجا IC، STM32F407VG می باشد .

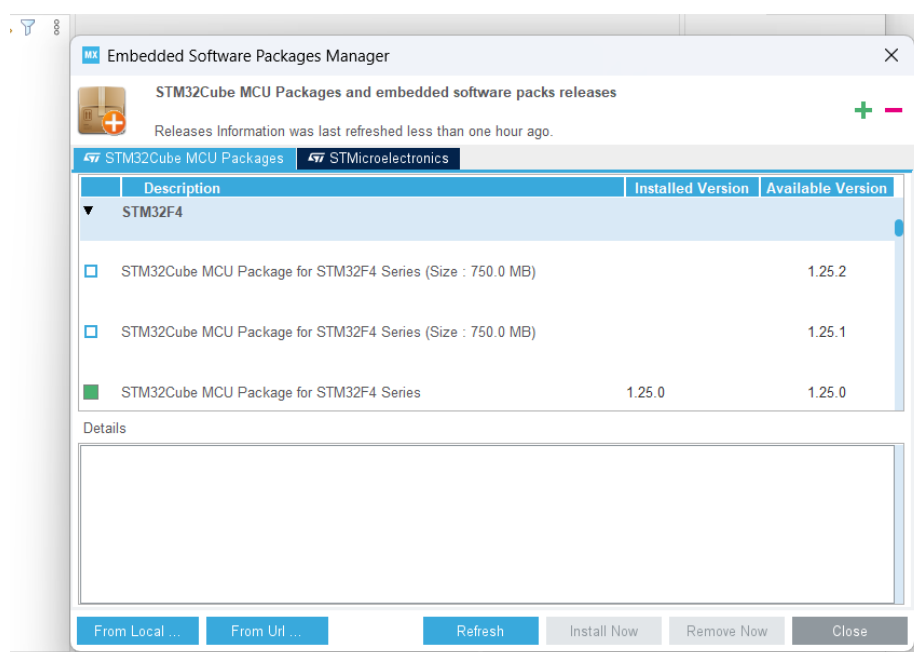


3- اضافه کردن کتابخانه :

وارد تب help شده و گزینه manage embedded software packages را انتخاب می کنیم .

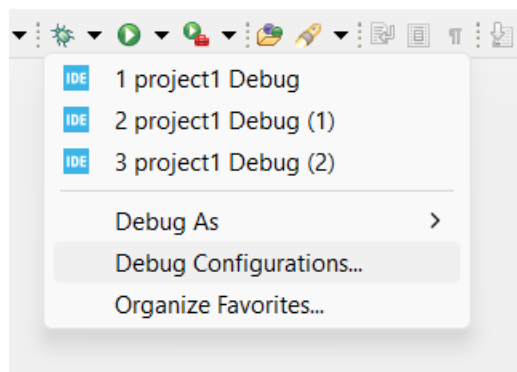


از این بخش مطابق شکل پایین کتابخانه مدنظر را انتخاب می کنیم .

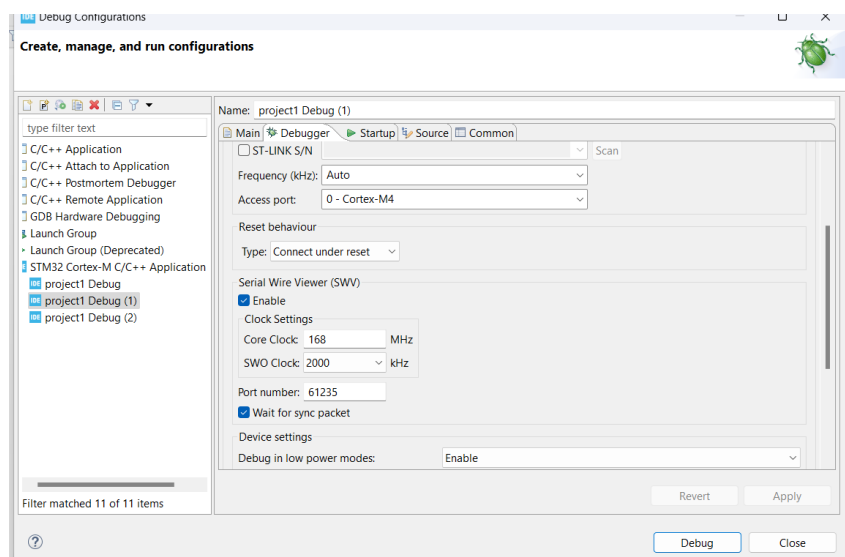


4- آغاز دیباگ :

برروی دکمه دیباگ کلیک کرده debug configuration را برای تنظیمات لازم باز می کنیم .

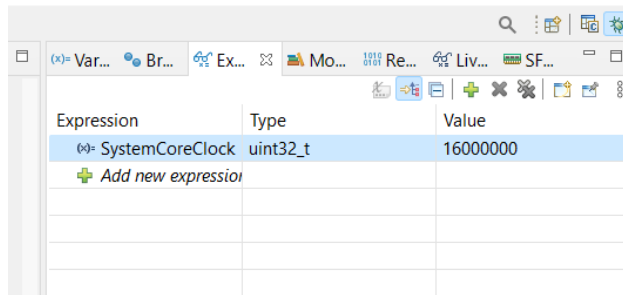


در صفحه باز شده بخش SWD را فعال می کنیم و مقدار CORE CLOCK را وارد می کنیم .



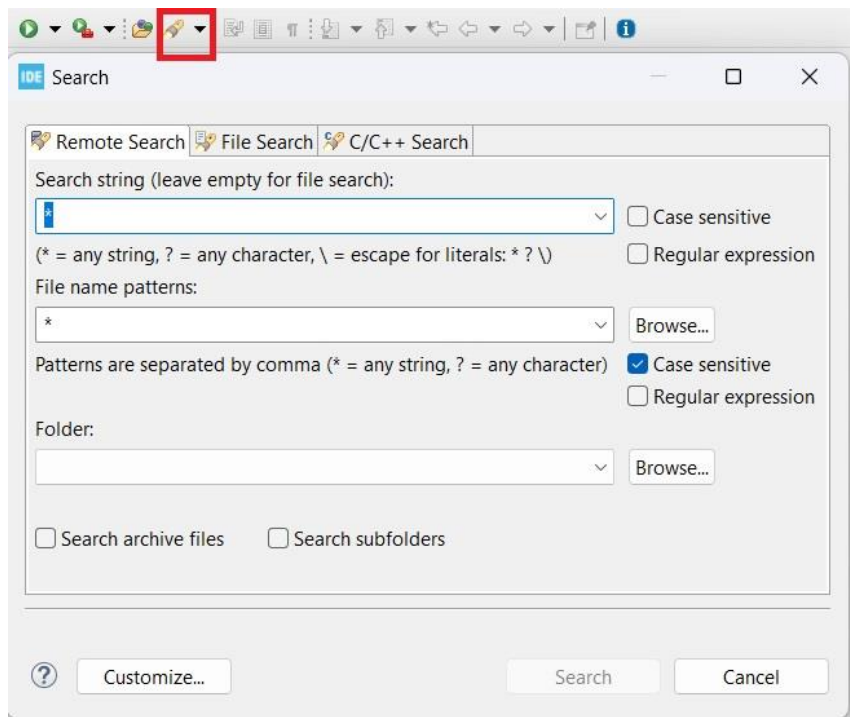
5- افزودن متغیر System core clock :

متغیر SystemCoreClock را به Expressions اضافه می کنیم.

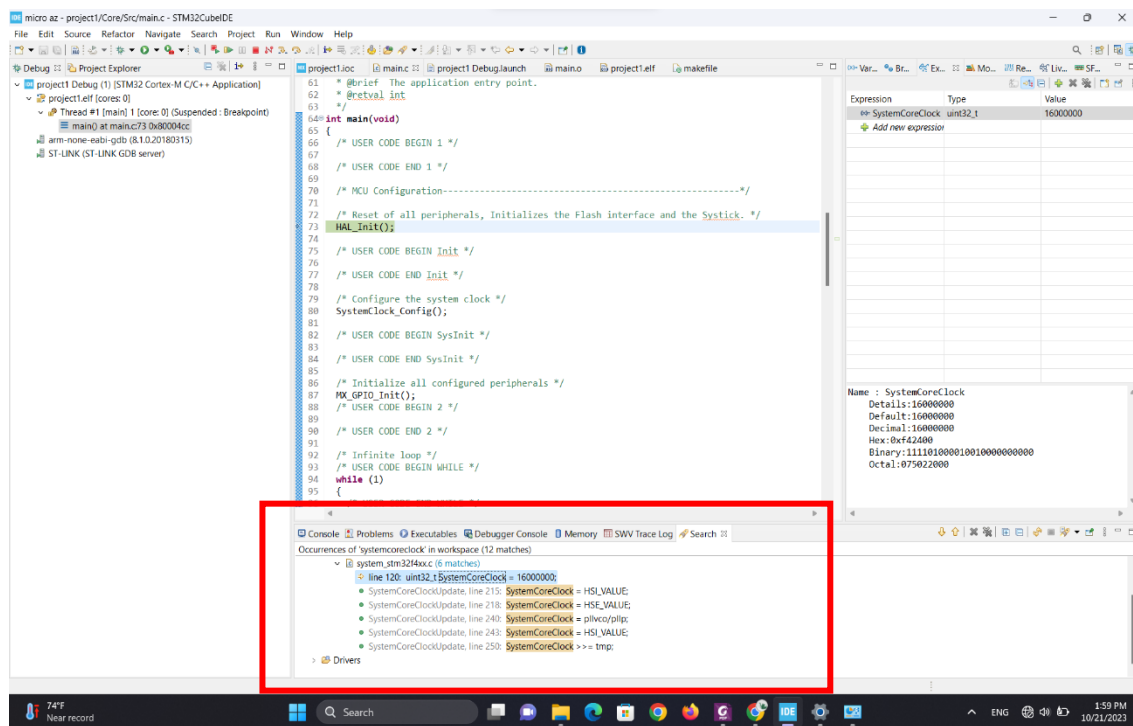


6- یافتن مکان SystemCoreClock با استفاده از torch tool :

آیکون torch tool را انتخاب کرده و از آنجا با انتخاب systemcoreclock مکان آن مانند عکس دوم مشخص می شود.



نتیجه سرچ نشان می دهد این عبارت در فایل system_stm32f4xx.c در خط 120 تعریف شده است.



-7

ابتدا تابع write را به صورت زیر اضافه می کنیم .

```
29 /* USER CODE BEGIN PTD */
30
31 int _write(int file, char *ptr, int len)
32 {
33     int DataIdx;
34
35     for (DataIdx = 0; DataIdx < len; DataIdx++)
36     {
37         ITM_SendChar(*ptr++);
38     }
39     return len;
40 }
41
42
```

کارکتر i را به صورت گلوبال تعریف می کنیم .

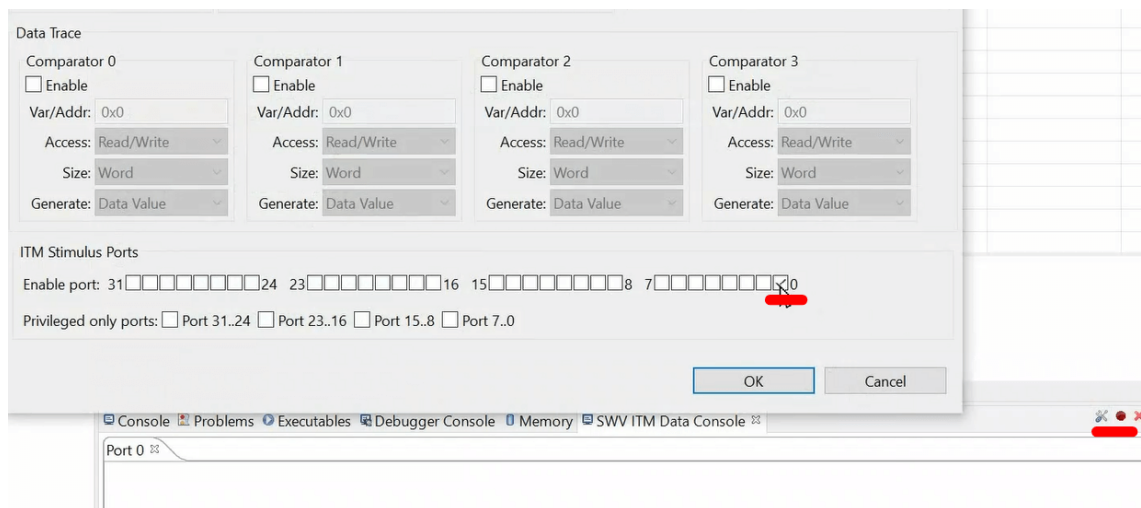
```
40 }
41
42
43 /* USER CODE END PTD */
44
45 /* Private define -----
46 /* USER CODE BEGIN PD */
47 uint32_t i=0 ;
48 char ch[100];
49 /* USER CODE END PD */
50
51 /* Private macro -----
52 /* USER CODE BEGIN PM */
53
54 /* USER CODE END PM */
55
56 /* Private variables -----
57
58 /* USER CODE BEGIN PV */
59
60 /* USER CODE END PV */
61
62 /* Private function prototypes -----
63 void SystemClock_Config(void);
64 static void MX_GPIO_Init(void).
```

کد مد نظر را داخل حلقه while(1) به صورت زیر نوشتیم . در اینجا عبارت hello و بعد مقدار i چاپ می شود و در هر بار چاپ مقدار i یک واحد افزوده می شود .

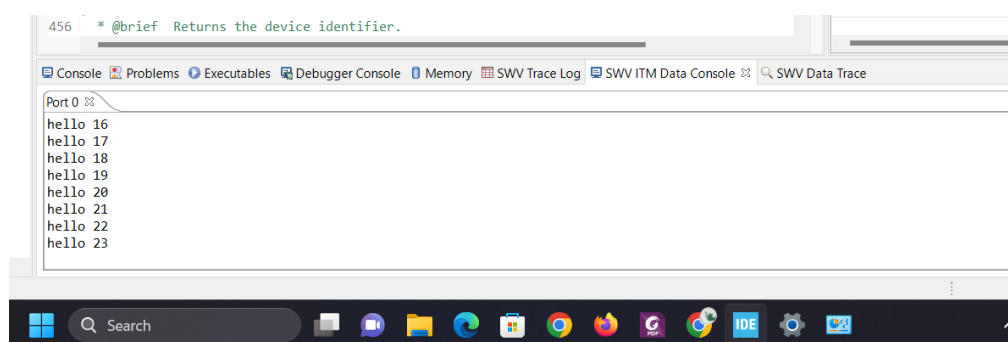
```
106 /* Infinite loop */
107 /* USER CODE BEGIN WHILE */
108 while (1)
109 {
110     /* USER CODE END WHILE */
111
112     /* USER CODE BEGIN 3 */
113     sprintf(ch, "hello %ld \n", i);
114     printf(ch) ;
115     i++;
116
117     HAL_Delay(1000);
118 }
119 /* USER CODE END 3 */
120 }
121
122 /**
123  * @brief System Clock Configuration
124  * @retval None
125  */
126 void SystemClock_Config(void)
127 {
```

از منوی show view-> trsce ، SWV ITM Data Console را اضافه می کنیم و نتیجه کدمان را مشاهده می کنیم .

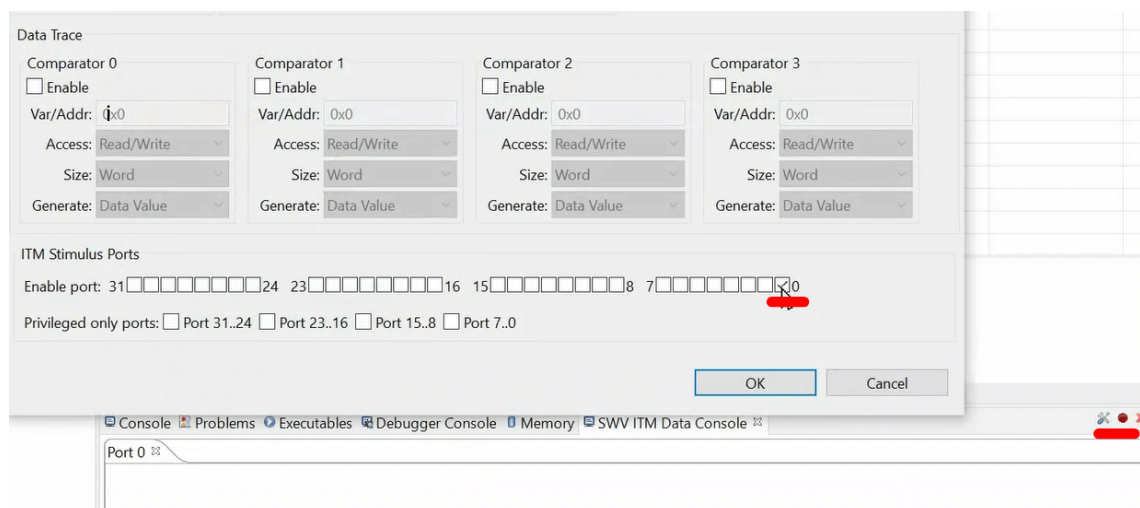
قبل از آن طبق شکل اول از کنسول گزینه قرمز را انتخاب میکنیم و پین 0 را فعال می کنیم . دلیل اینکه پین 0 را فعال می کنیم این است که در ابتدا هنگام تعریف تابع write داده را داخل پین 0 تعریف کردیم .



کنسول SWV ITM Data به صورت زیر کد را چاپ کرد :



در اینجا در data trace یکی از comparator ها را فعال کرده و نام آن را نوشته (در این جا i)، و با اضافه کردن SWC Data trace تغییرات i را مشاهده می کنیم (شکل دوم)



Console Problems Executables Debugger Console Memory SWV ITM Data Console

Port 0

Comp	Name	Value
0	i	5

Access	Value	PC	Cycles	Time
READ	0		71897	427.958333 μs
WRITE	1		71901	427.982143 μs
READ	1		168239507	1.001426 s
WRITE	2		168239511	1.001426 s
READ	2		336407406	2.002425 s
WRITE	3		336407410	2.002425 s
READ	3		504575323	3.003425 s
WRITE	4		504575327	3.003425 s
READ	4		672743230	4.004424 s
WRITE	5		672743234	4.004424 s