YTMicro MCAL应用 - Icu模块配置及实例

版本

Config Tool Version：3.0.0

HA01 MCAL version：2.3.0

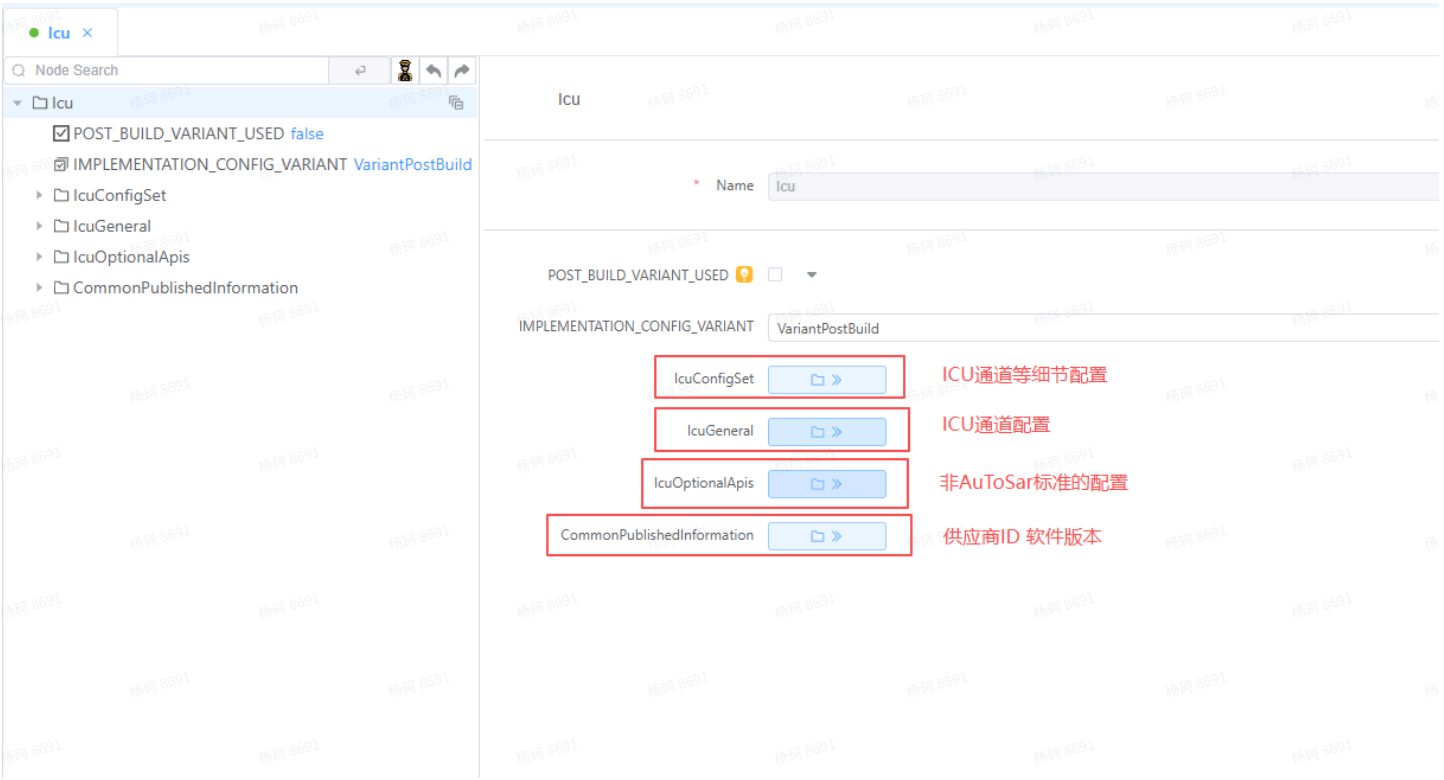
前言

1. Icu一般依赖Mcu、Platform和Port模块，Mcu模块里面需要配置外设时钟，Platform里面可以配置外设中断，Port配置使用到的端口资源。
2. Port、Mcu和Platform的默认配置参考Port篇、Setup篇和Mcu篇介绍。
3. 配置中包含AutoSar规范标准选项和云途自定义选项，左边树选框中粗体字为AutoSar标准选项，细体字为云途自定义选项。
4. MCAL应用_Icu模块配置及实例主要说明Icu模块的基本配置，本文章主要介绍描述Etmr、MPWM作为输入捕获的实例。

1. Icu配置介绍

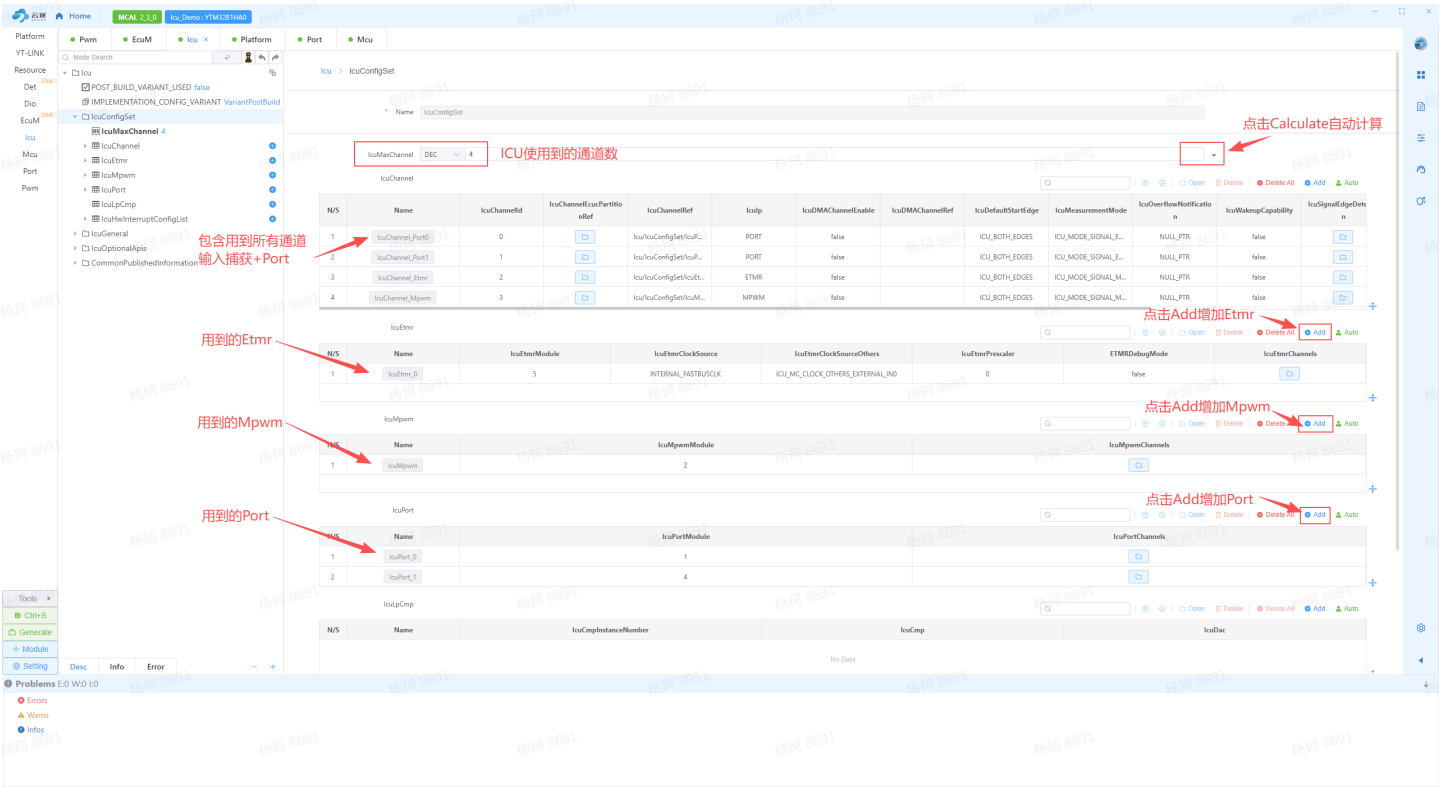
1.1 框架说明

此处介绍了整个ICU模块的配置框架，主要需要注意的是IcuConfigSet和IcuGeneral两部分，具体细节下面章节介绍。



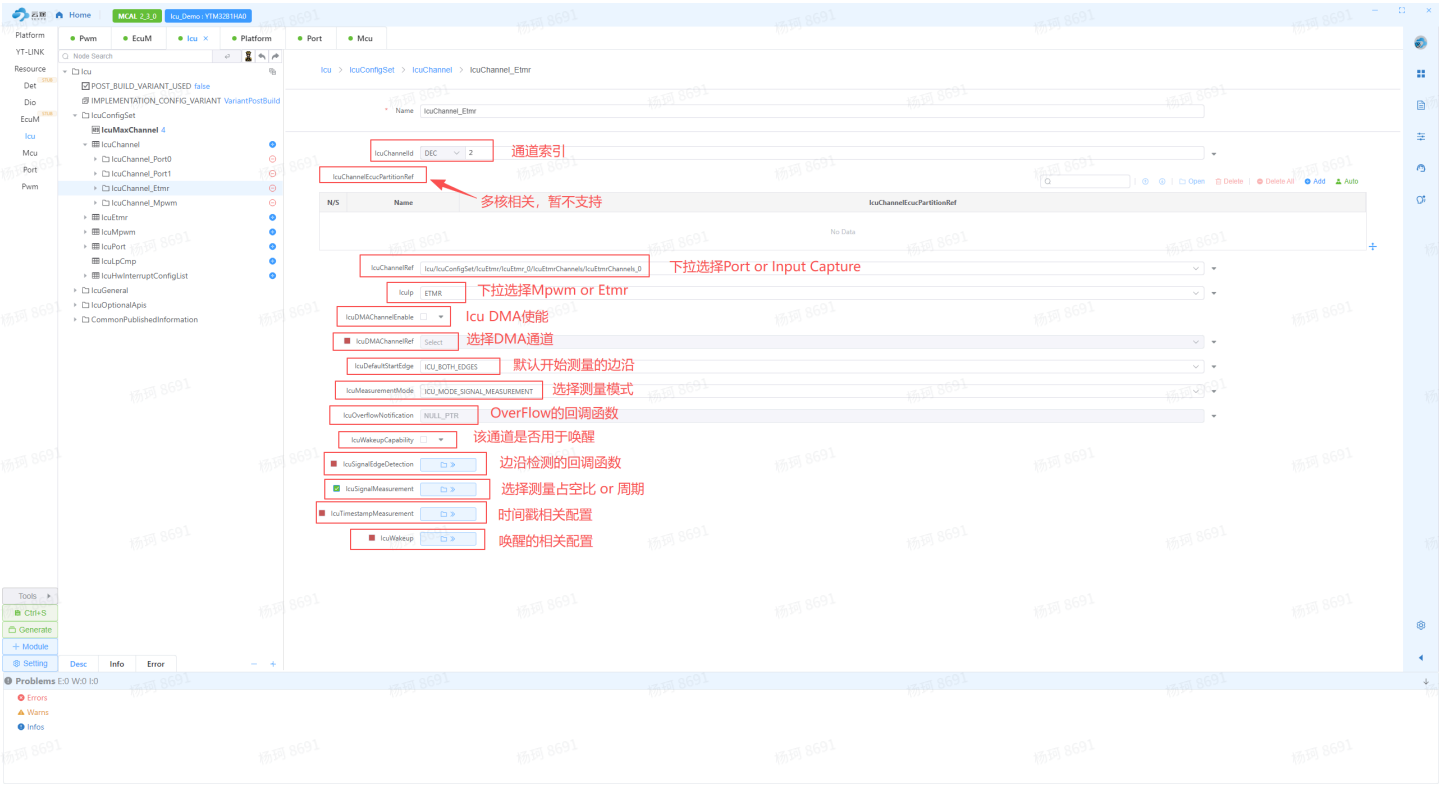
1.2 IcuConfigSet

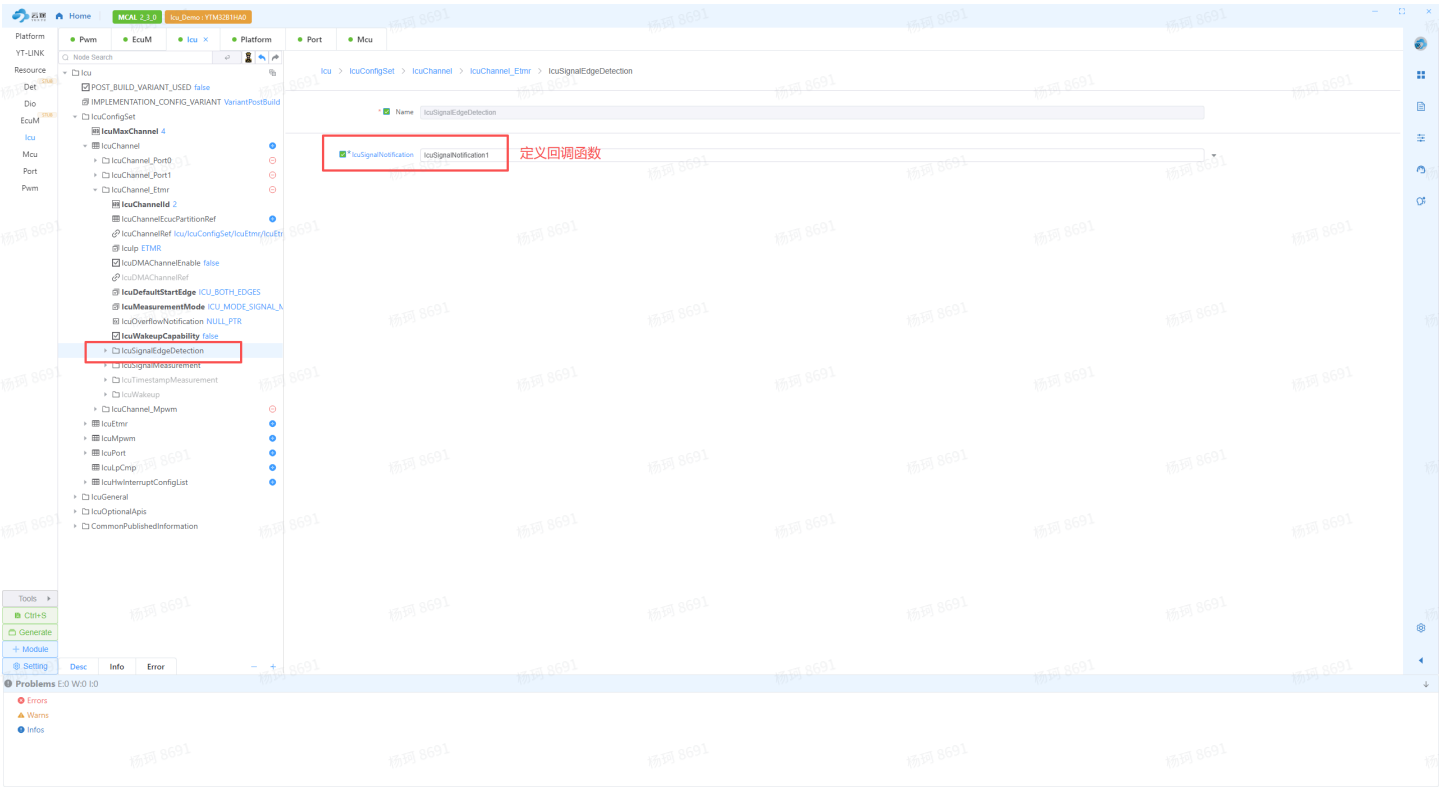
此处介绍了ICU通道的设置，ICU可作为ETMR（输入捕获）、MPWM（输入捕获）、PORT、LpCmp功能。

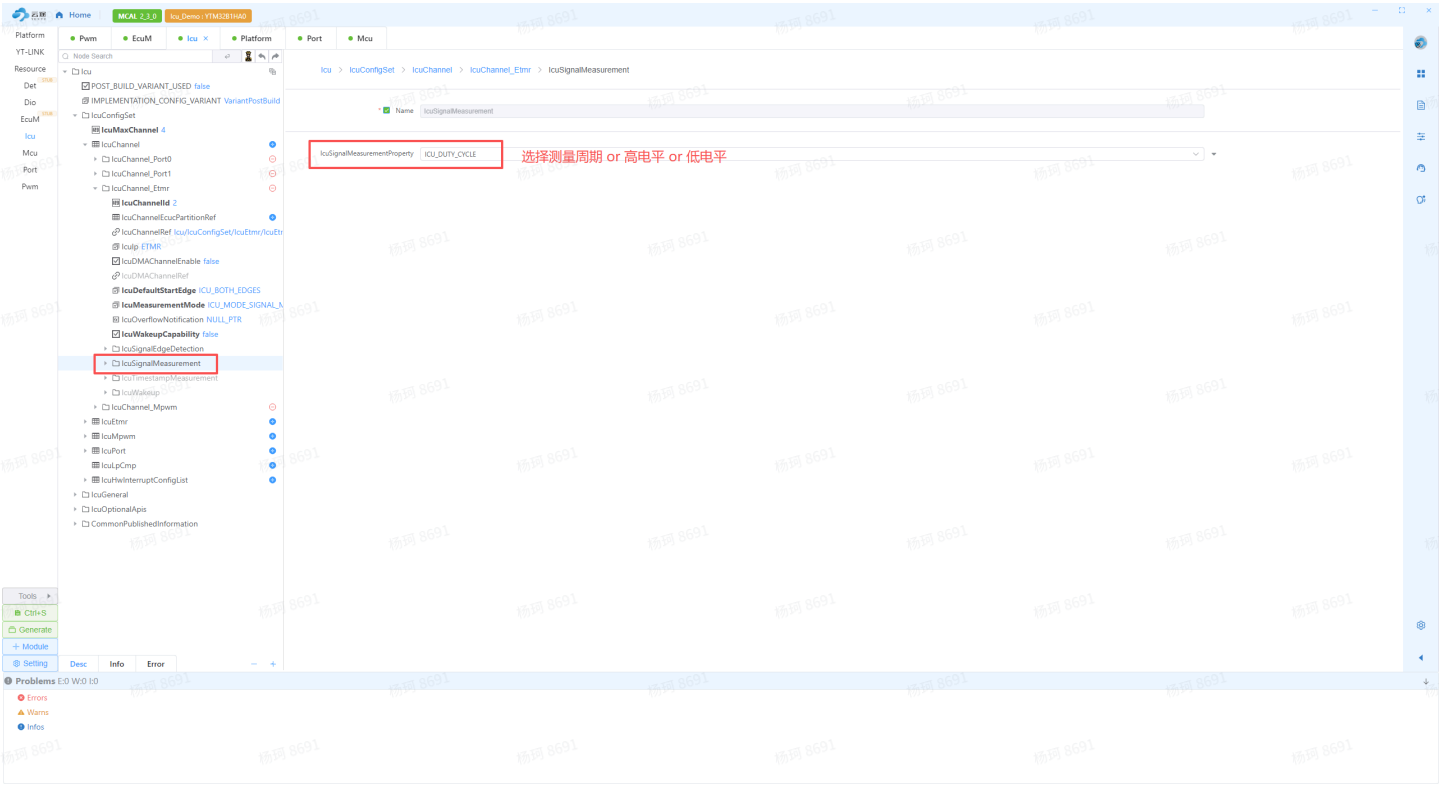


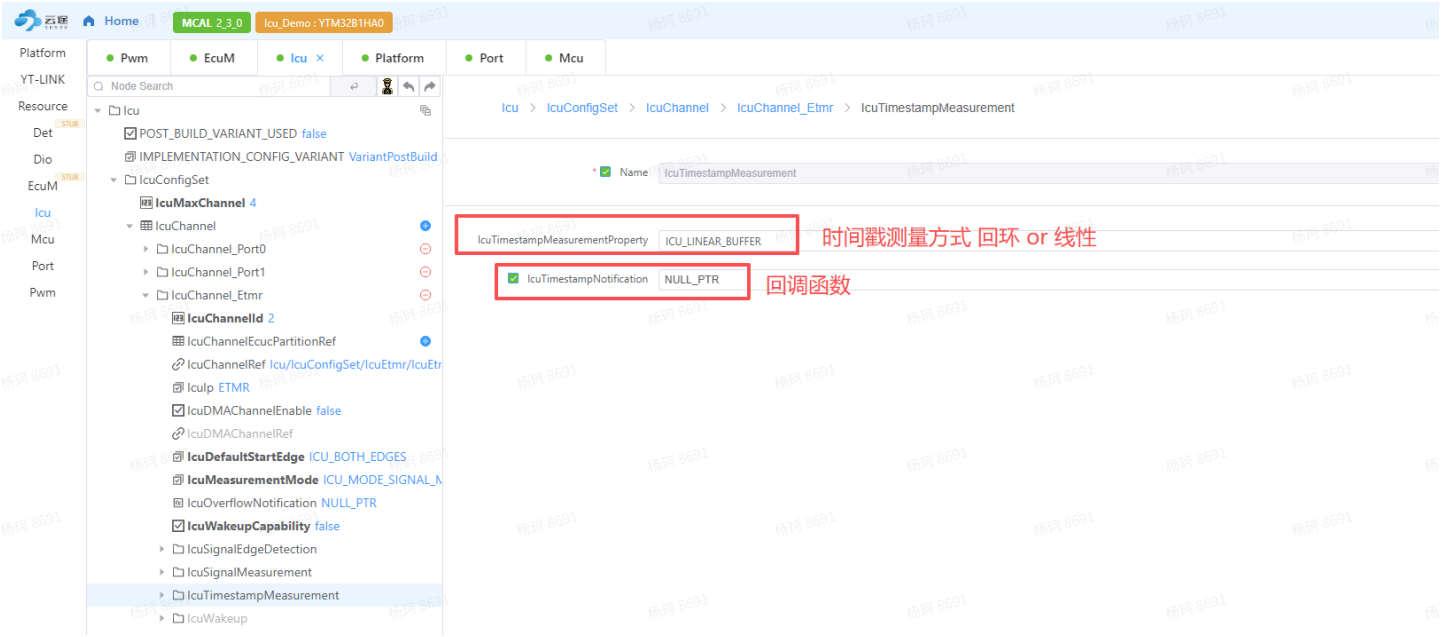
1.2.1 IcuChannel

此处介绍了ICU作输入捕获时的具体配置细节。


通道索引
多核相关，暂不支持
下拉选择Port or Input Capture
下拉选择Mpwm or Etmr
Icu DMA使能
选择DMA通道
默认开始测量的边沿
选择测量模式
OverFlow的回调函数
该通道是否用于唤醒
边沿检测的回调函数
选择测量占空比 or 周期
时间戳相关配置
唤醒的相关配置


定义回调函数





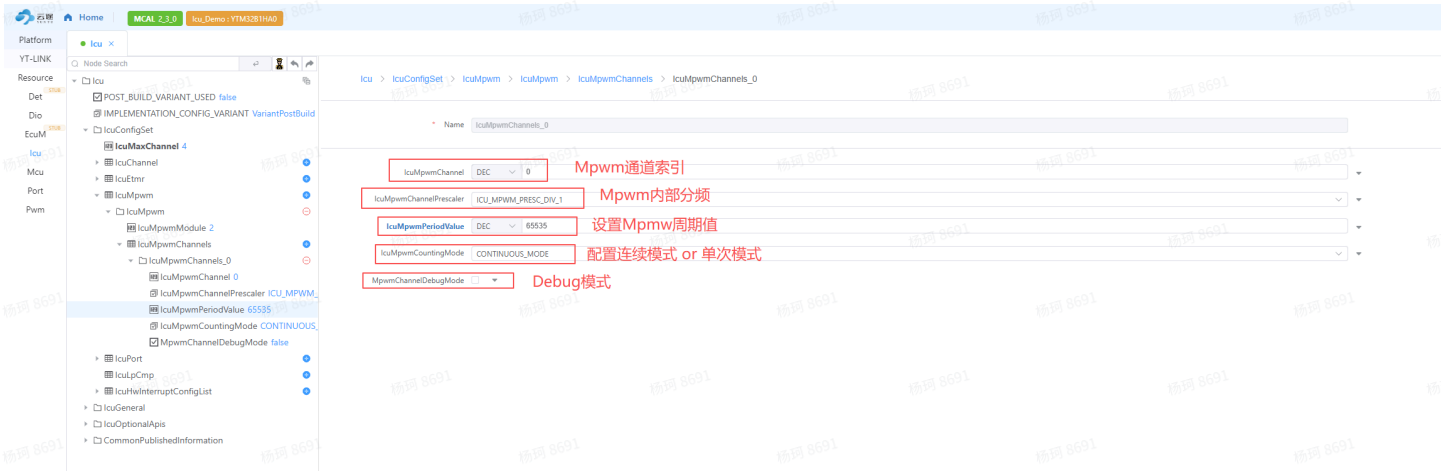
1.2.2 IcuEtmr

ICU作为输入捕获可以依赖于芯片的ETMR模块，此处需要配置ETMR模块参数及使用的Channel。



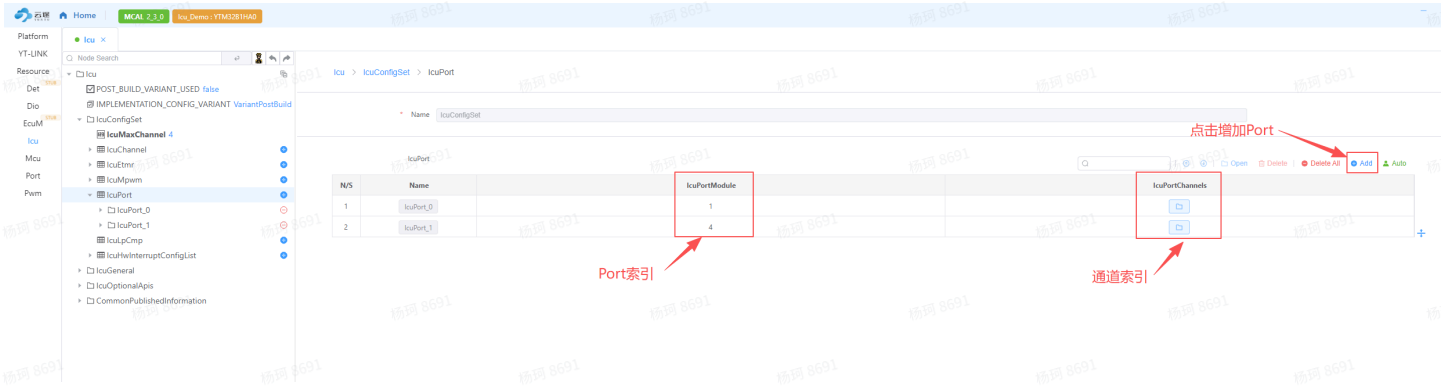
1.2.3 IcuMpwm

ICU作为输入捕获可以依赖于芯片的Mpwm模块，此处需要配置Mpwm模块参数及使用的Channel。



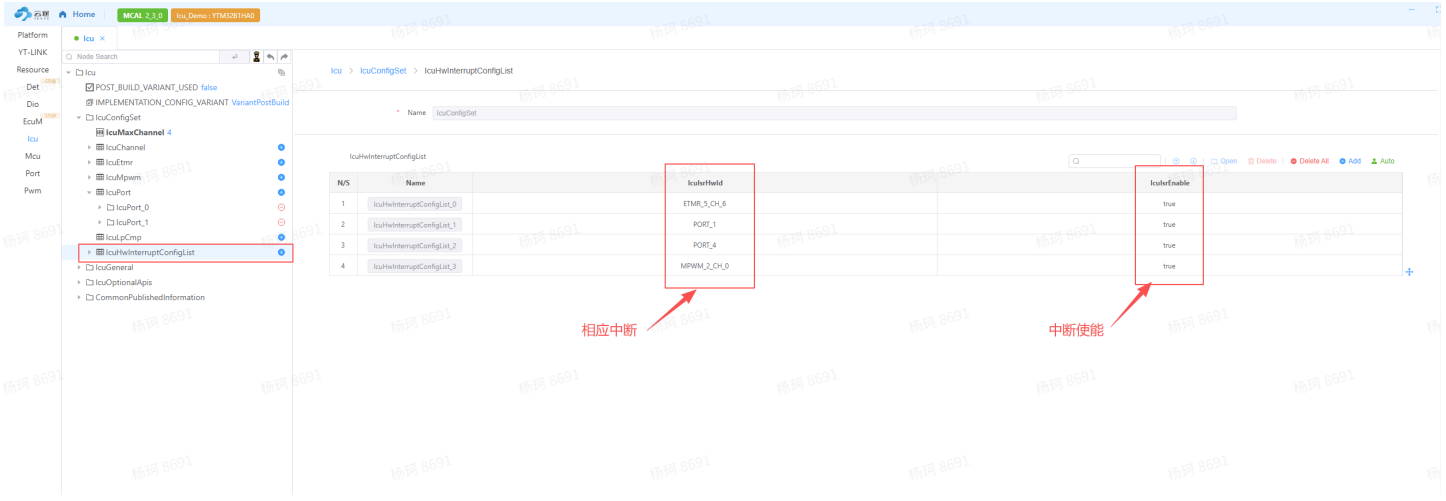
1.2.4 IcuPort

此处是设置PORT，当需要使用GPIO中断的时候，需要在此部分设置相应的通道。



1.2.5 IcuHwInterruptConfigList

此处使能相应模块的中断



1.2.6 IcuLpcmp

后续章节介绍

1.3 IcuGeneral

此处设置一些通用配置，一般需要打开DET，其它配置按照实际需求配置即可。



1.4 IcuOptionalApis

此处是AutoSar可选的配置，根据需求配置即可。



2. ICU模块应用实例

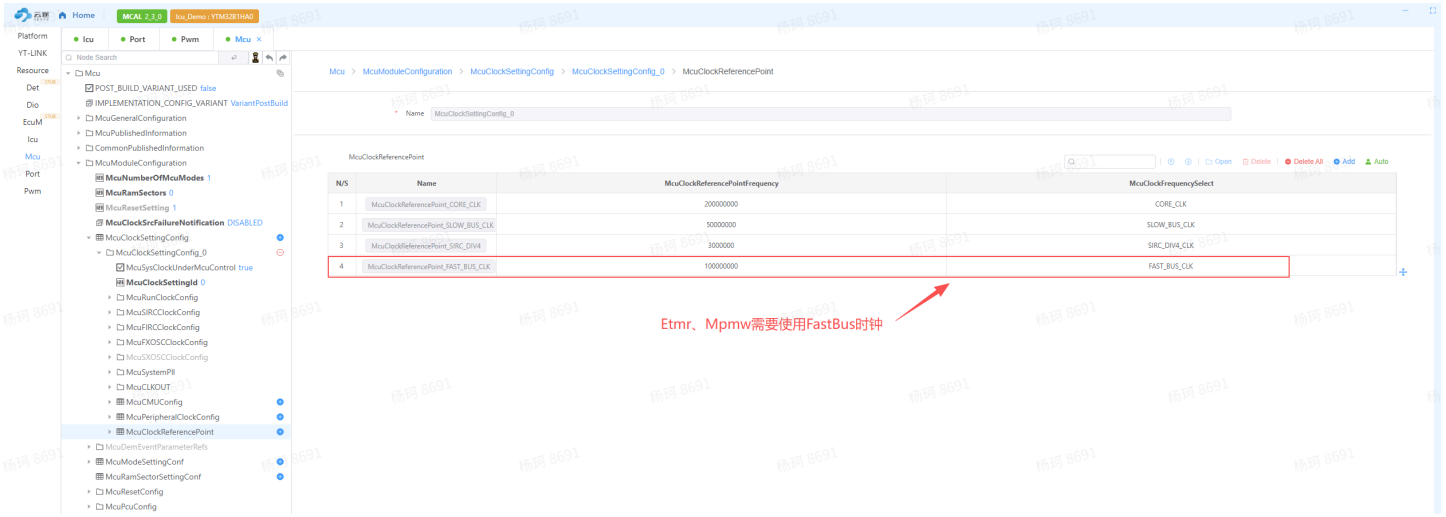
2.1 应用场景

- 基于HA01 Demo板，使用ETMR5、MPWM2进行输入捕获。
- PORT选择PTE3、PTE26引脚用作按键输入。
- PWM0_CH0输出10Khz的PWM用于ICU测试。

2.2 ETMR做输入捕获的配置步骤

2.2.1 时钟配置

在MCU模块中配置时钟，ETMR需要选择快速时钟。

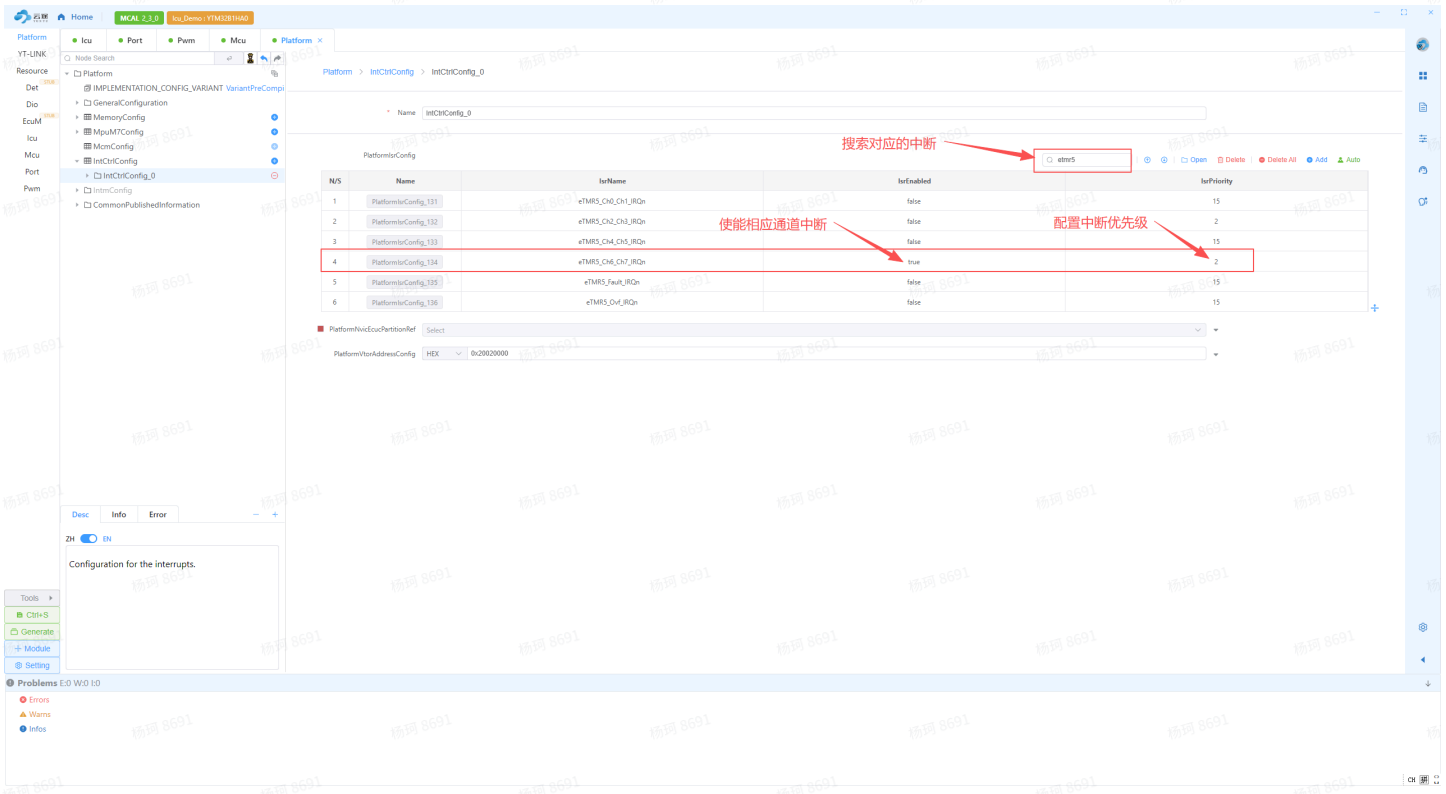


2.2.2 中断配置

配置使能相应的中断。

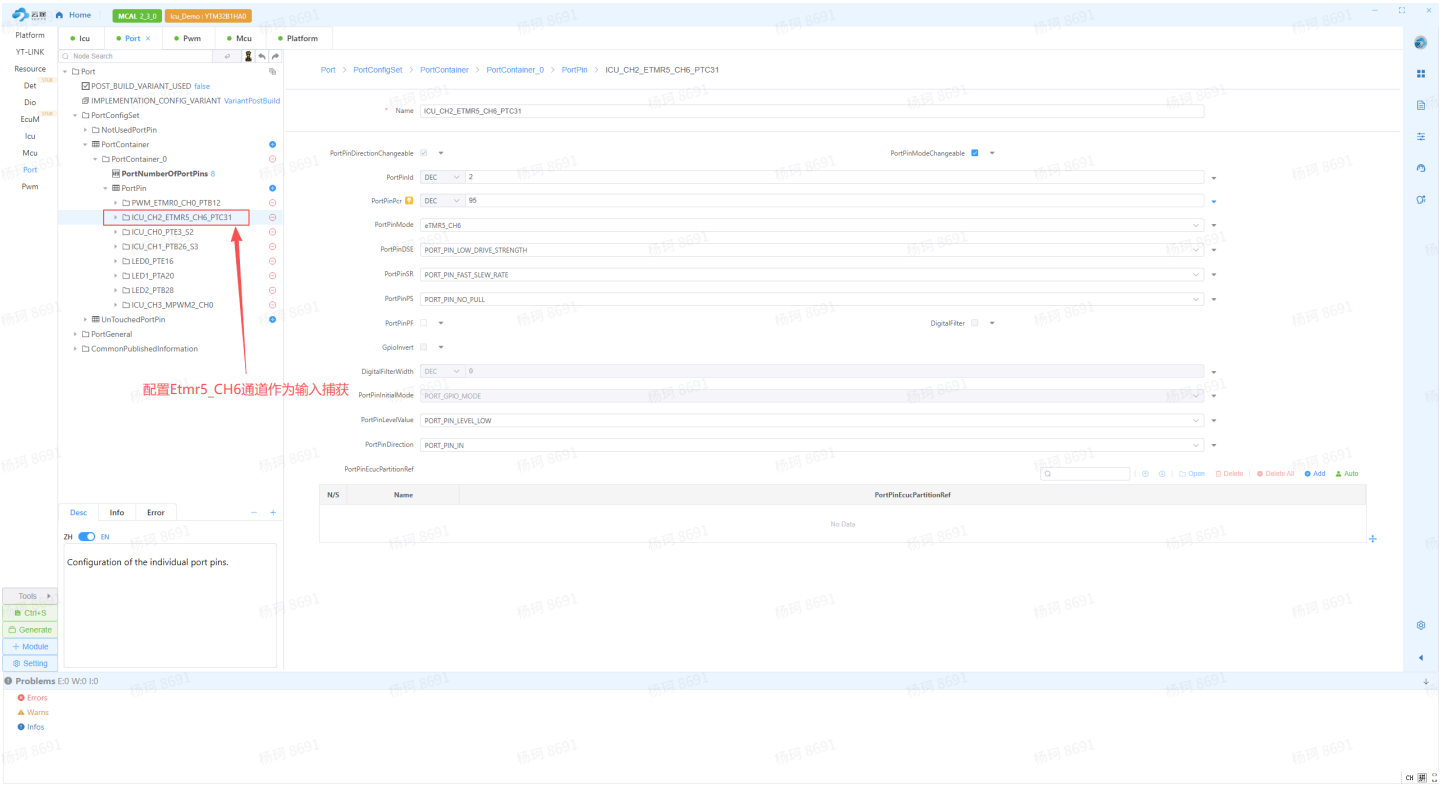
Tips:

这里需要说明的是，由于ETMR的中断是两个通道共用一个中断号，因此需要根据实际通道配置。



2.2.3 PORT配置

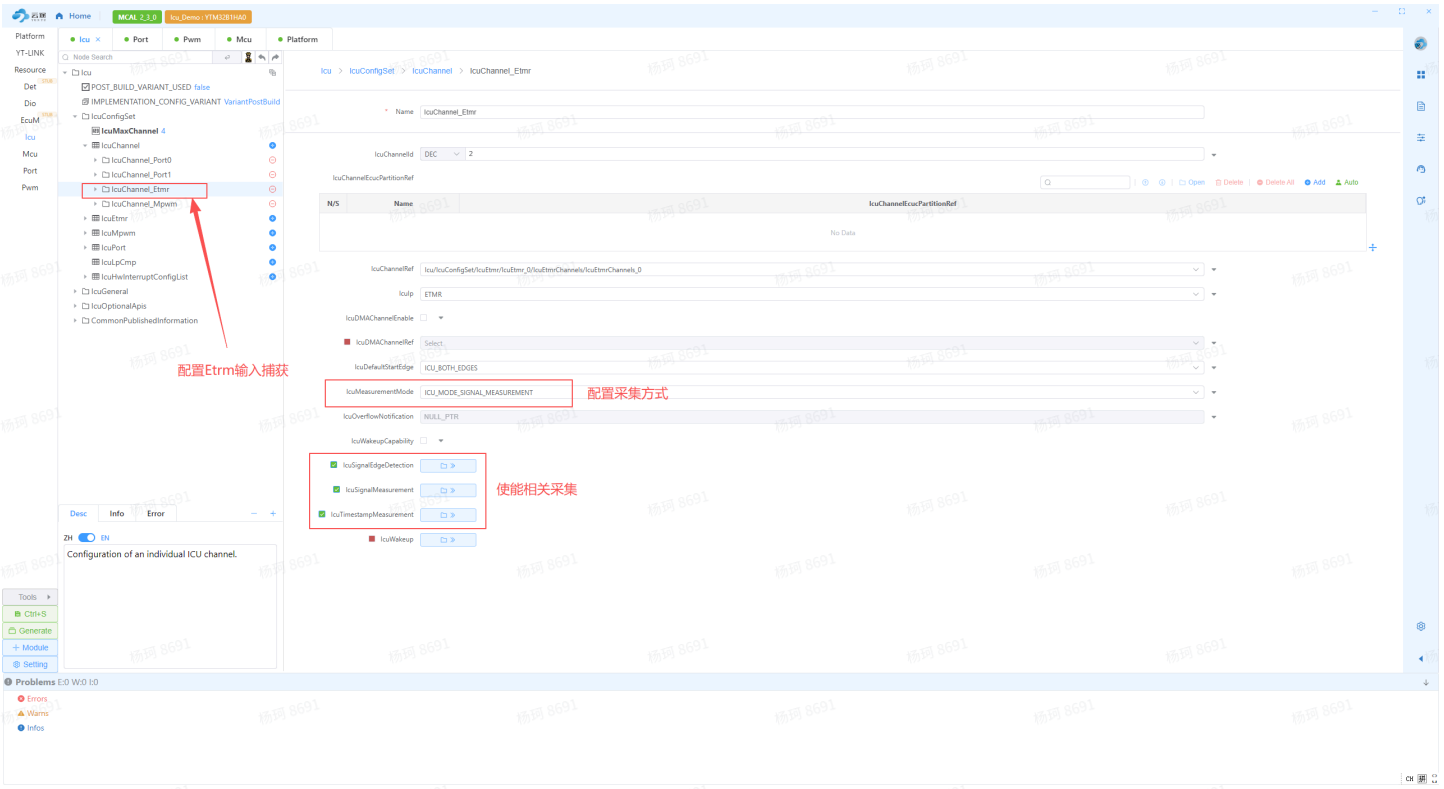
这里选择ETMR5的CH6作为输入捕获，对应芯片PTC31引脚。



2.2.4 ICU配置

2.2.4.1 配置ICU通道

这里配置Etmr输入捕获通道，并配置捕获方式。

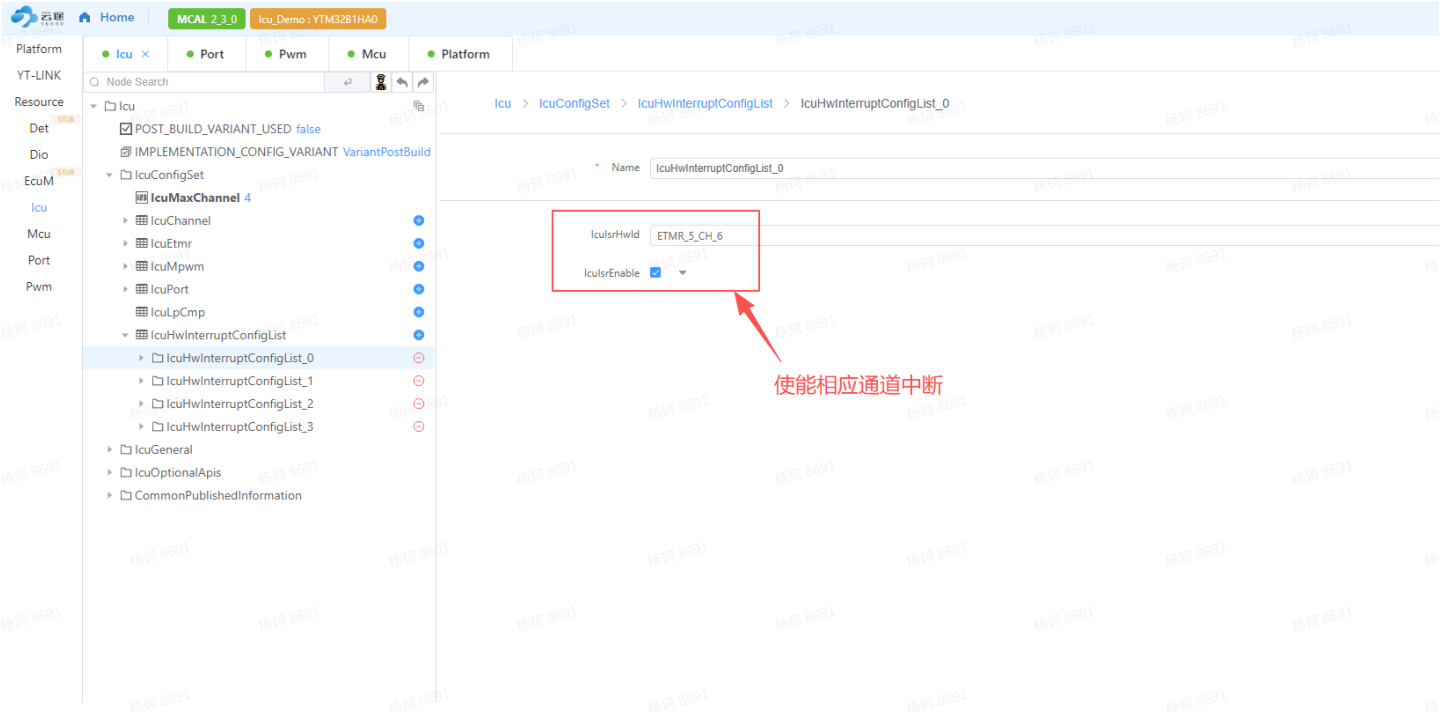


2.2.4.2 配置ETMR

配置用到的ETMR5_CH6通道。



2.2.4.3 配置中断



2.2.5 代码说明

main.c

Icu_Demo > app > main.c > main

```
{
  /*Start Measure*/
  Icu_StartSignalMeasurement(Channel: IcuConf_IcuChannel_IcuChannel_Etmr);  //Icu pwm signal measurement
  Icu_StartSignalMeasurement(Channel: IcuConf_IcuChannel_IcuChannel_Mpwm);  //Icu pwm signal measurement
  /*****************************************************************************/
  /*******************************Icu port edge detection*************************************/
  Icu_EnableNotification(Channel: IcuConf_IcuChannel_IcuChannel_Port0);
  Icu_EnableNotification(Channel: IcuConf_IcuChannel_IcuChannel_Port1);
  Icu_EnableEdgeDetection(Channel: IcuConf_IcuChannel_IcuChannel_Port0);
  Icu_EnableEdgeDetection(Channel: IcuConf_IcuChannel_IcuChannel_Port1);
  Icu_SetActivationCondition(Channel: IcuConf_IcuChannel_IcuChannel_Port0, Activation: ICU_FALLING_EDGE);
  Icu_SetActivationCondition(Channel: IcuConf_IcuChannel_IcuChannel_Port1, Activation: ICU_FALLING_EDGE);
  /*****************************************************************************/
  /* Wait Switch(sw2 and sw3) trigger the Port edge detection* */
  // while ((IcuInterruptCount0 == 0) || (IcuInterruptCount1 == 0));
  /* USER CODE END 2 */

  /* Infinite loop */
  /* USER CODE BEGIN WHILE */
  while (1)
  {
    if(IcuInterruptCount0 != 0)
    {
      if (ICU_ACTIVE == Icu_GetInputState(Channel: IcuConf_IcuChannel_IcuChannel_Etmr))
      {
        /* Wait until Icu is active */
        Icu_GetDutyCycleValues(Channel: IcuConf_IcuChannel_IcuChannel_Etmr, DutyCycleValues: &Dbg_MeasureDutyCycleValue_Ch0);
        if (Dbg_MeasureDutyCycleValue_Ch0.PeriodTime <= TEST_10KHZ + 100 && Dbg_MeasureDutyCycleValue_Ch0.PeriodTime >= TEST_10KHZ - 100)
        {
          Icu_StopSignalMeasurement(Channel: IcuConf_IcuChannel_IcuChannel_Etmr);
          IcuInterruptCount0 = 0;
          // while (1);  //test pass
        }
      }
    }
    if(IcuInterruptCount1 != 0)
    {
      if (ICU_ACTIVE == Icu_GetInputState(Channel: IcuConf_IcuChannel_IcuChannel_Mpwm))
      {
        /* Wait until Icu is active */
        Icu_GetDutyCycleValues(Channel: IcuConf_IcuChannel_IcuChannel_Mpwm, DutyCycleValues: &Dbg_MeasureDutyCycleValue_Ch1);
        if (Dbg_MeasureDutyCycleValue_Ch1.PeriodTime <= TEST_10KHZ + 100 && Dbg_MeasureDutyCycleValue_Ch1.PeriodTime >= TEST_10KHZ - 100)
        {
          Icu_StopSignalMeasurement(Channel: IcuConf_IcuChannel_IcuChannel_Mpwm);
          IcuInterruptCount1 = 0;
          // while (1);  //test pass
        }
      }
    }

    /* USER CODE END WHILE */
    /* USER CODE BEGIN 3 */
```

开始测量

按下SW2

Etmr做为输入捕获，判断是否读取到Active

获取DutyCycle

按下SW3

Mpwm作为输入捕获，判断是否读取到Active

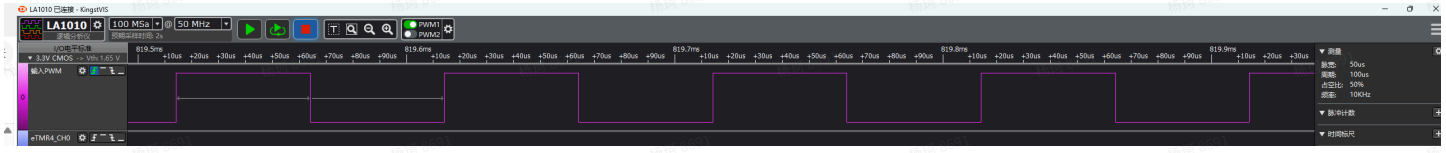
获取DutyCycle

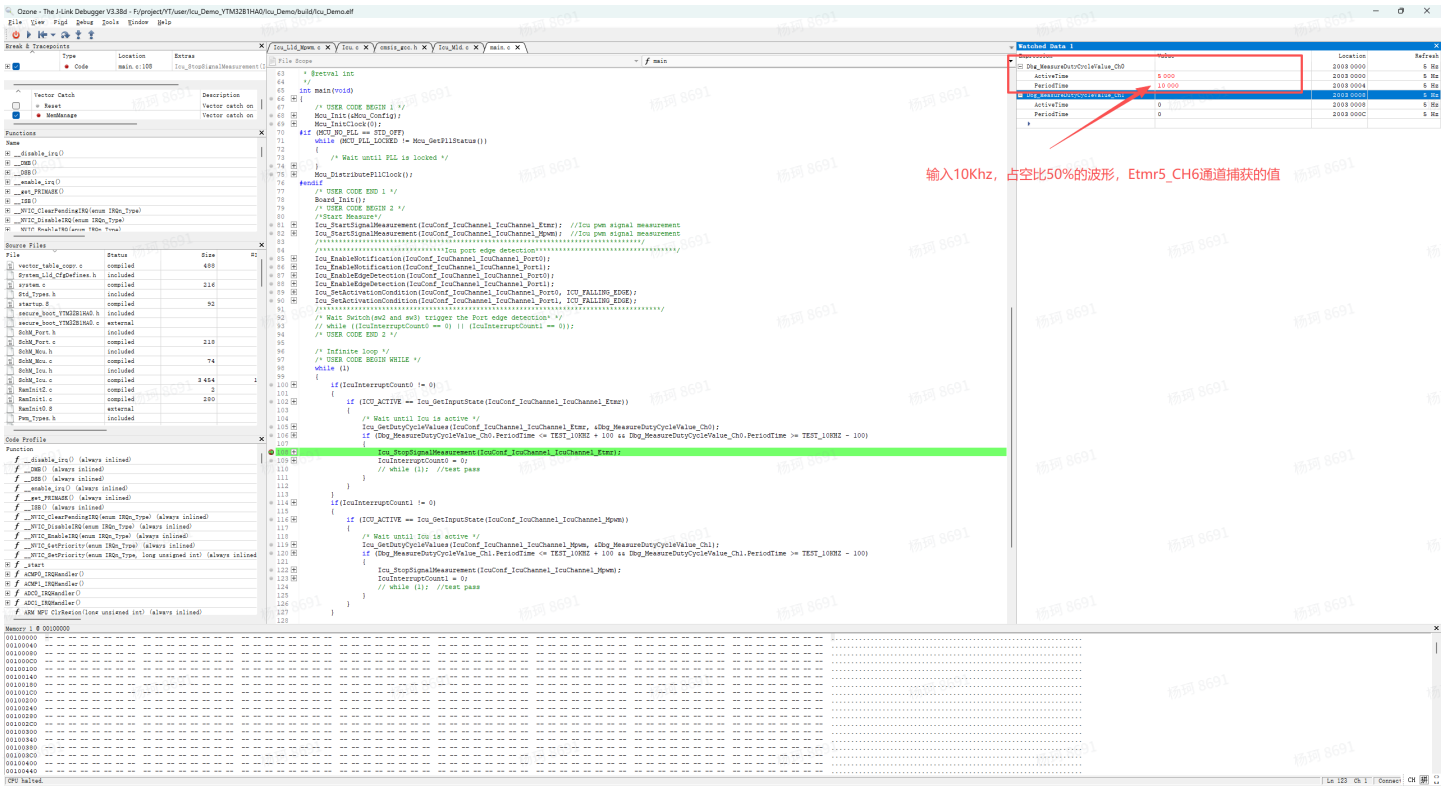
2.2.6 现象

根据Port配置的通道，在板子上选中相应的通道引脚。

周期：1/(100M) * 10000 = 100us

占空比：1/(100M) * 5000 = 50us

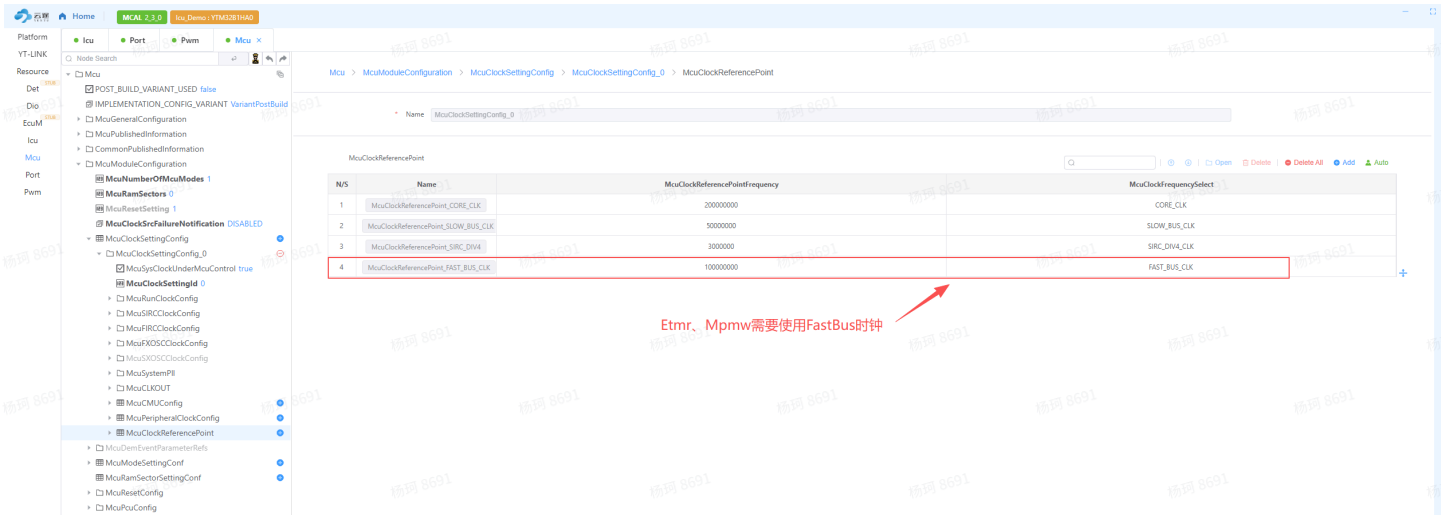




2.3 MPWM做输入捕获的配置步骤

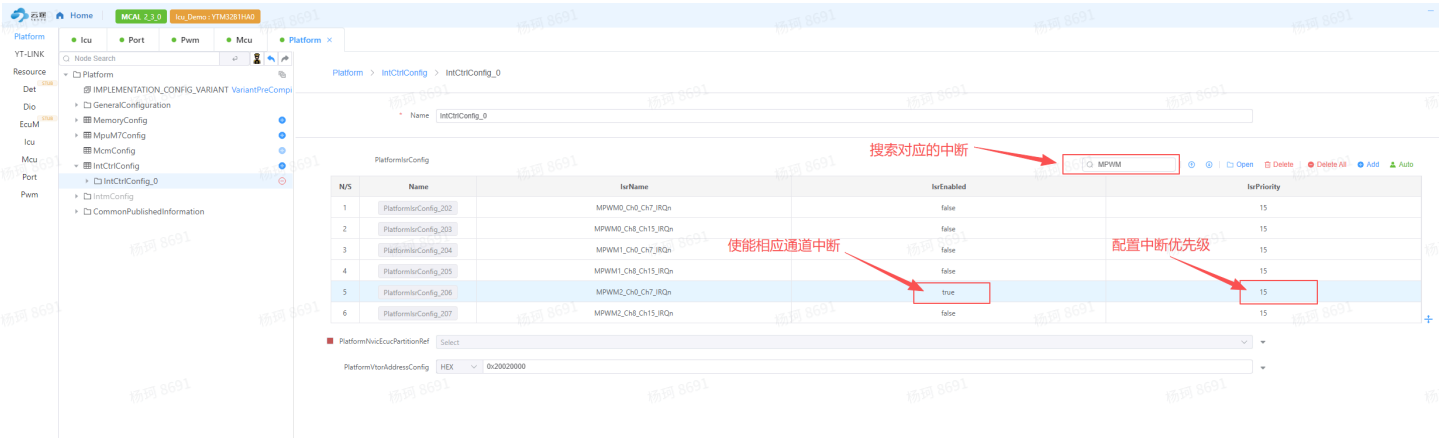
2.3.1 时钟配置

在MCU模块中配置时钟，Mpwm需要选择快速时钟。

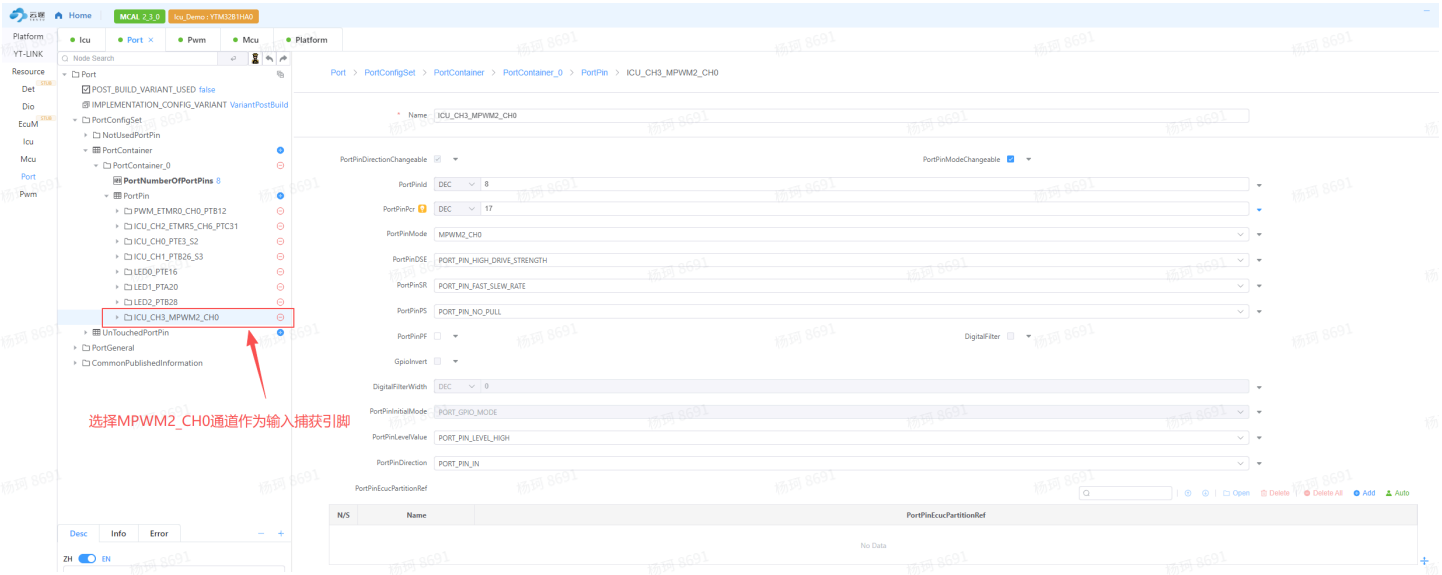


2.3.2 中断配置

配置使能相应的中断。

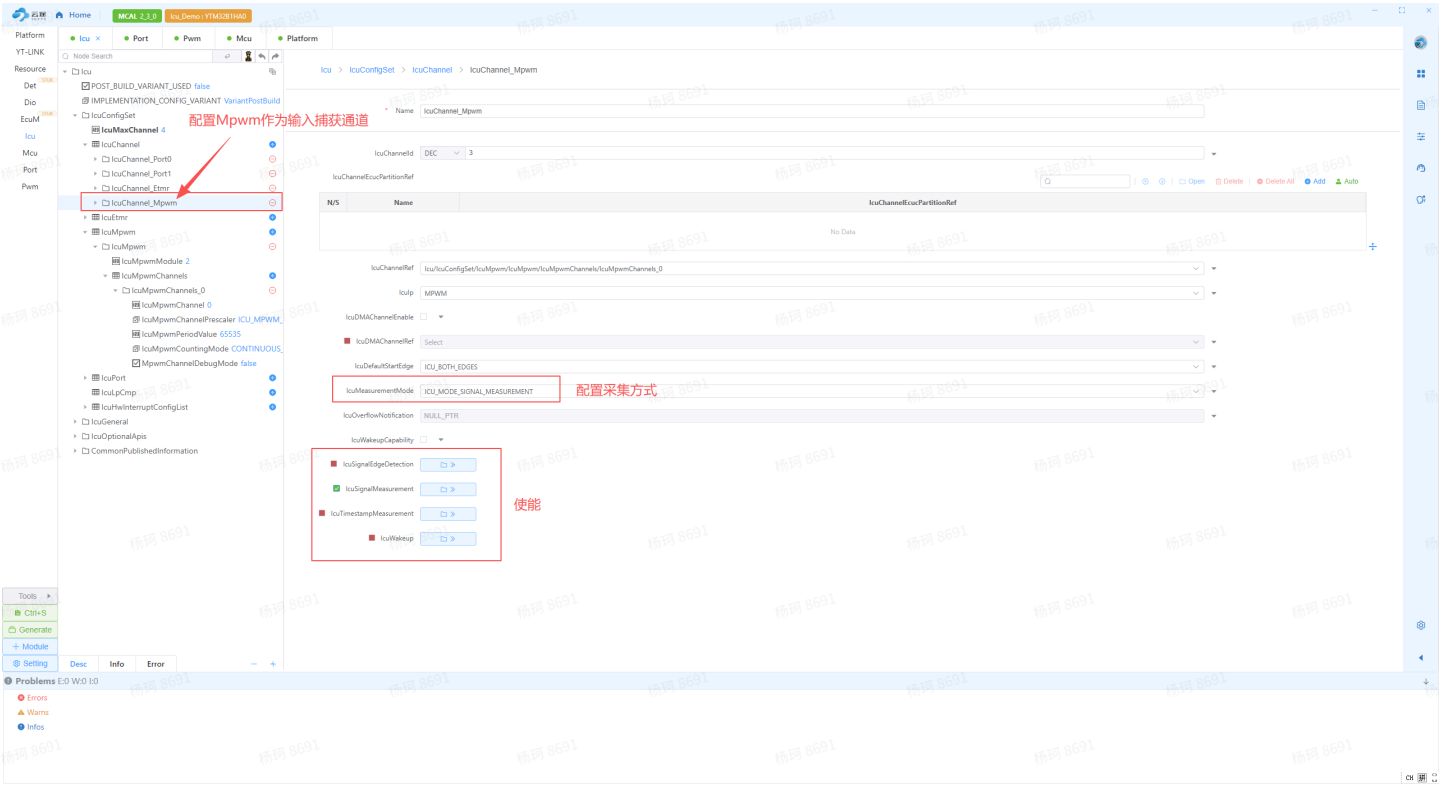


2.3.3 PORT配置

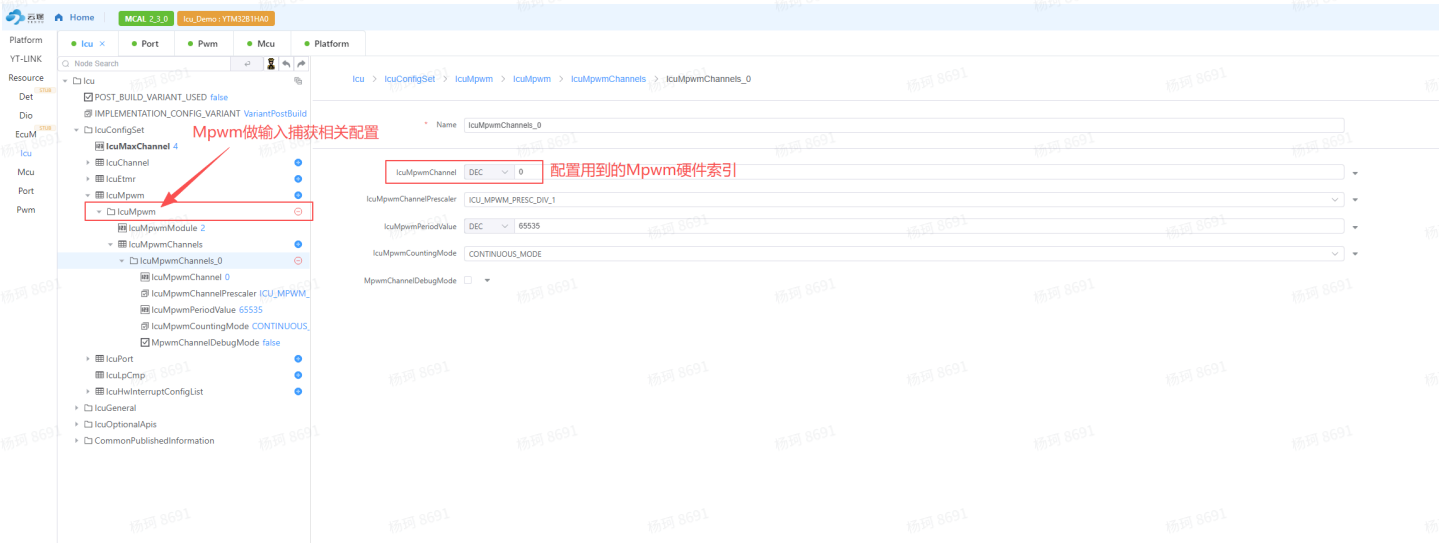


2.3.4 ICU配置

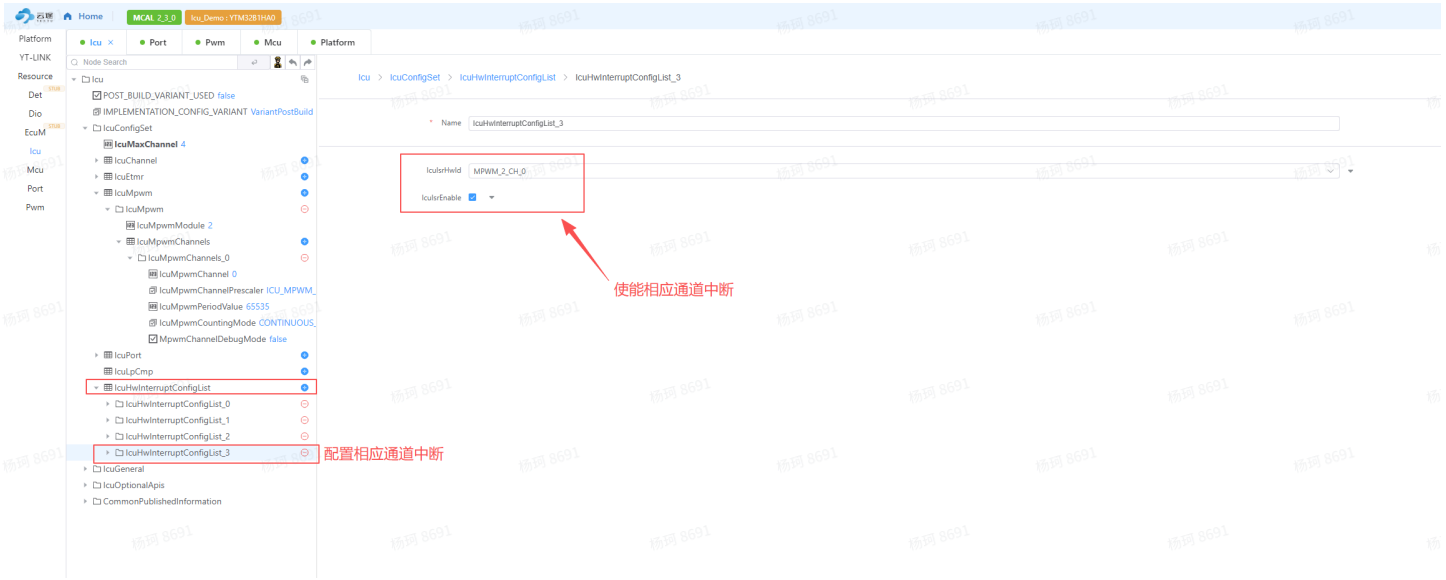
2.3.4.1 配置ICU通道



2.3.4.2 配置MPWM



2.3.4.3 配置中断



2.3.5 代码说明

```c
/*Start Measure*/
Icu_StartSignalMeasurement(Channel: IcuConf_IcuChannel_IcuChannel_Etmr);  //Icu pwm signal measurement
Icu_StartSignalMeasurement(Channel: IcuConf_IcuChannel_IcuChannel_Mpwm);  //Icu pwm signal measurement
/*********************************************************************************/
/*********************************Icu port edge detection*********************************/
Icu_EnableNotification(Channel: IcuConf_IcuChannel_IcuChannel_Port0);
Icu_EnableNotification(Channel: IcuConf_IcuChannel_IcuChannel_Port1);
Icu_EnableEdgeDetection(Channel: IcuConf_IcuChannel_IcuChannel_Port0);
Icu_EnableEdgeDetection(Channel: IcuConf_IcuChannel_IcuChannel_Port1);
Icu_SetActivationCondition(Channel: IcuConf_IcuChannel_IcuChannel_Port0, Activation: ICU_FALLING_EDGE);
Icu_SetActivationCondition(Channel: IcuConf_IcuChannel_IcuChannel_Port1, Activation: ICU_FALLING_EDGE);
/*********************************************************************************/
/* Wait Switch(sw2 and sw3) trigger the Port edge detection* */
// while ((IcuInterruptCount0 == 0) || (IcuInterruptCount1 == 0));
/* USER CODE END 2 */

/* Infinite loop */
/* USER CODE BEGIN WHILE */
while (1)
{
    if(IcuInterruptCount0 != 0)
    {
        if (ICU_ACTIVE == Icu_GetInputState(Channel: IcuConf_IcuChannel_IcuChannel_Etmr))
        {
            /* Wait until Icu is active */
            Icu_GetDutyCycleValues(Channel: IcuConf_IcuChannel_IcuChannel_Etmr, DutyCycleValues: &Dbg_MeasureDutyCycleValue_Ch0);
            if (Dbg_MeasureDutyCycleValue_Ch0.PeriodTime <= TEST_10KHZ + 100 && Dbg_MeasureDutyCycleValue_Ch0.PeriodTime >= TEST_10KHZ - 100)
            {
                Icu_StopSignalMeasurement(Channel: IcuConf_IcuChannel_IcuChannel_Etmr);
                IcuInterruptCount0 = 0;
                // while (1);  //test pass
            }
        }
    }
    if(IcuInterruptCount1 != 0)
    {
        if (ICU_ACTIVE == Icu_GetInputState(Channel: IcuConf_IcuChannel_IcuChannel_Mpwm))
        {
            /* Wait until Icu is active */
            Icu_GetDutyCycleValues(Channel: IcuConf_IcuChannel_IcuChannel_Mpwm, DutyCycleValues: &Dbg_MeasureDutyCycleValue_Ch1);
            if (Dbg_MeasureDutyCycleValue_Ch1.PeriodTime <= TEST_10KHZ + 100 && Dbg_MeasureDutyCycleValue_Ch1.PeriodTime >= TEST_10KHZ - 100)
            {
                Icu_StopSignalMeasurement(Channel: IcuConf_IcuChannel_IcuChannel_Mpwm);
                IcuInterruptCount1 = 0;
                // while (1);  //test pass
            }
        }
    }

    /* USER CODE END WHILE */
    /* USER CODE BEGIN 3 */
```

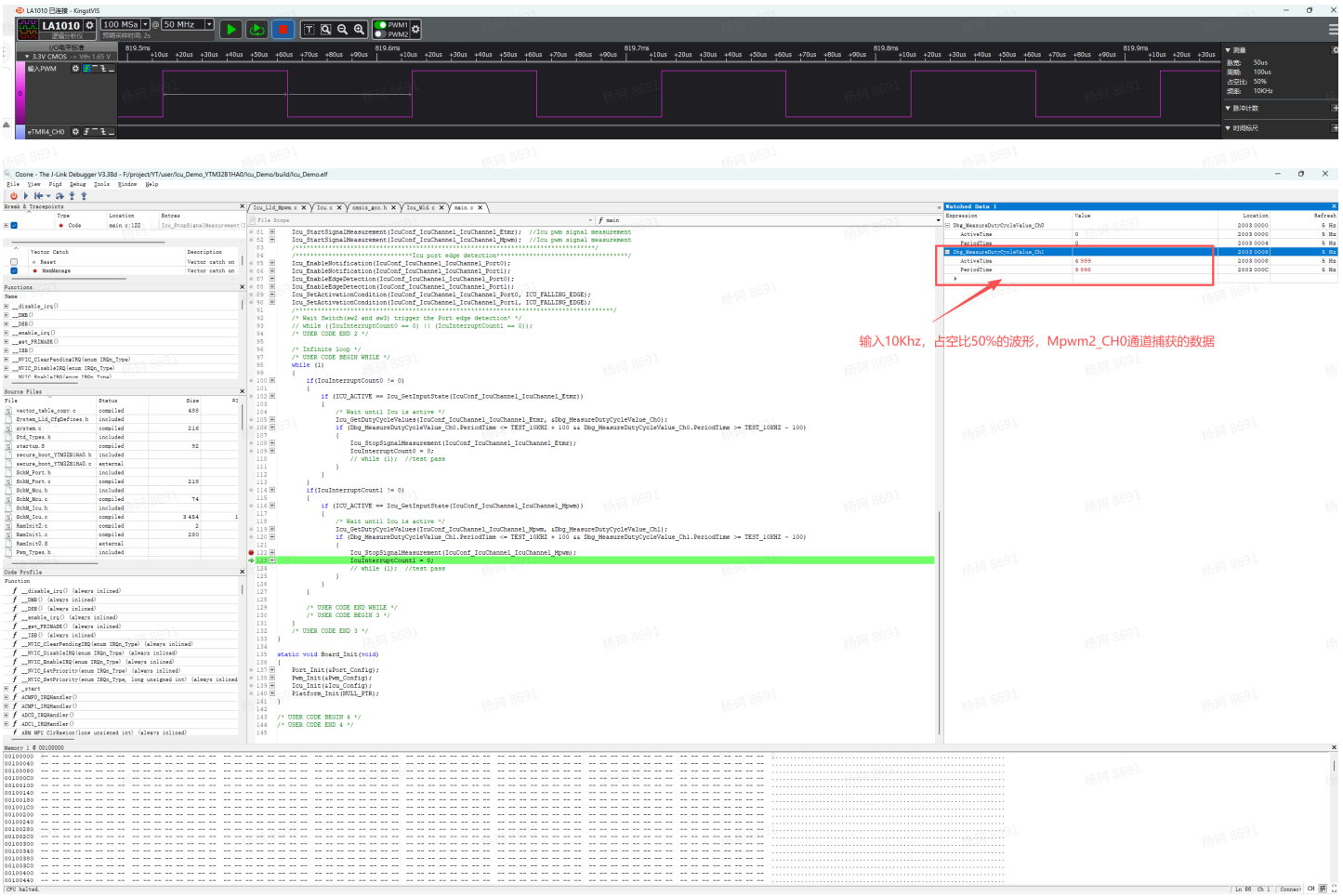
开始测量
按下SW2
Etmr做为输入捕获，判断是否读取到Active
获取DutyCycle
按下SW3
Mpwm作为输入捕获，判断是否读取到Active
获取DutyCycle

2.3.6 现象

根据Port配置的通道，在板子上选中相应的通道引脚。

周期：1/(100M) * 9998 = 99.98us

占空比：1/(100M) * 4999 = 49.99us



文档历史

版本号	日期	修订记录
V1.0	2026.05.04	初始版本