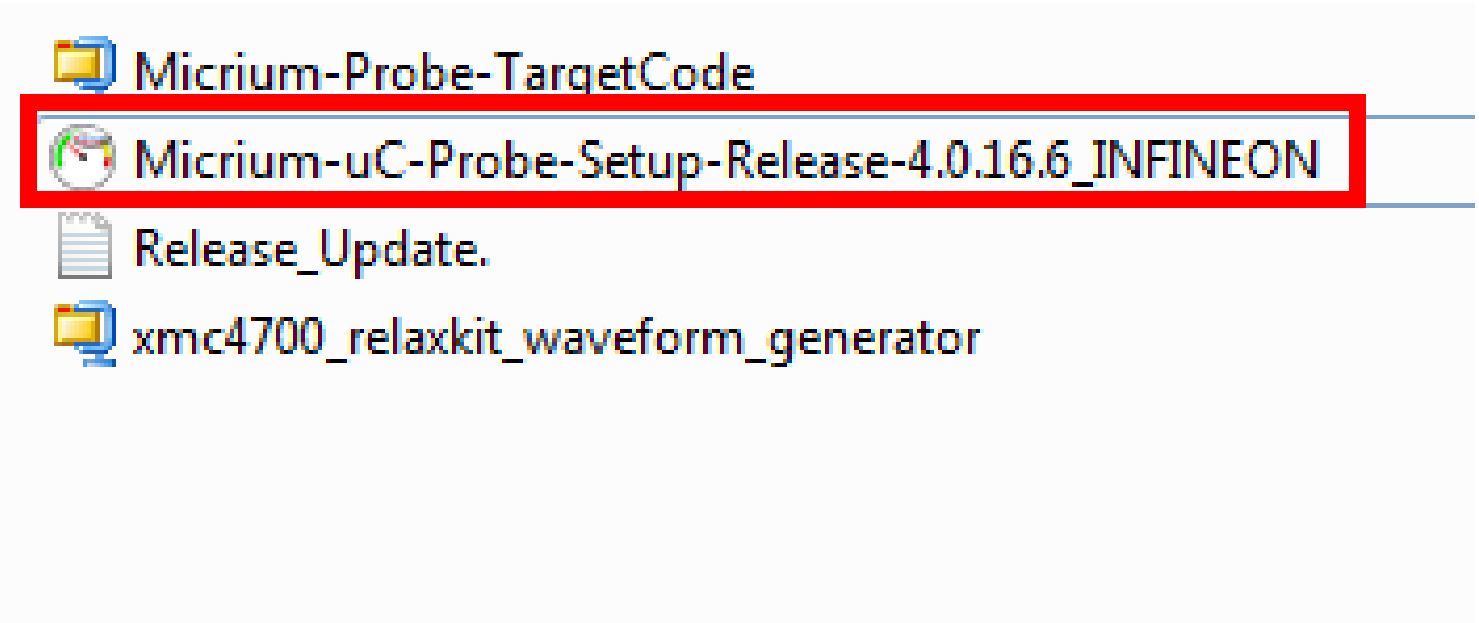
Start using µC/ Probe

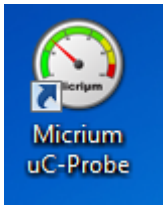
Meng deyun/Edison Zhang
(IFCN PMM SMD AP SH APC POWER)
2016-4-27

安装µC/ Probe

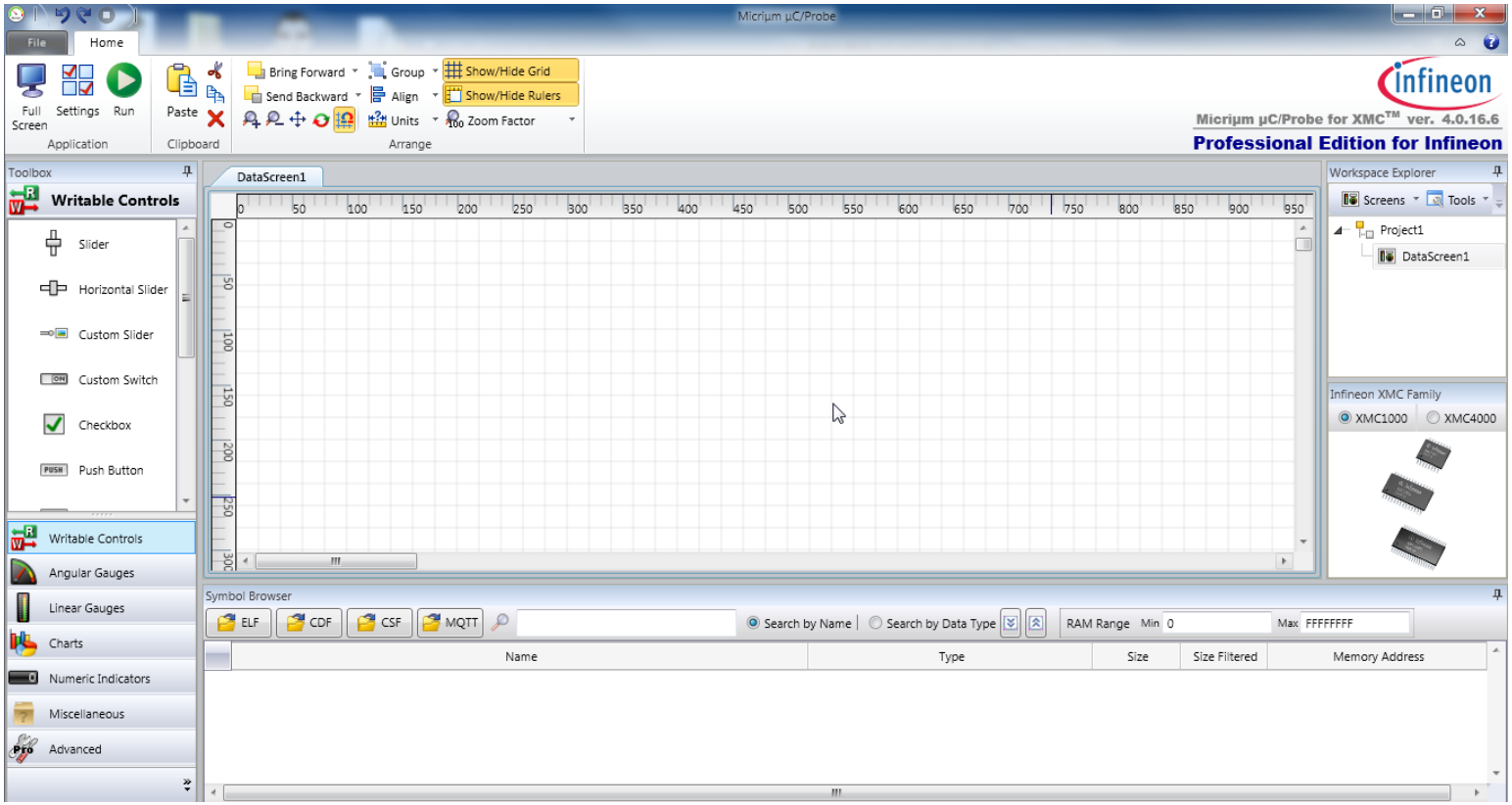


- 执行安装程序，按提示完成
- 压缩包TargetCode是虚拟示波器使用的C文件，后面详述
- 后一个压缩包XMC4700kit的样例

进入uC_Probe

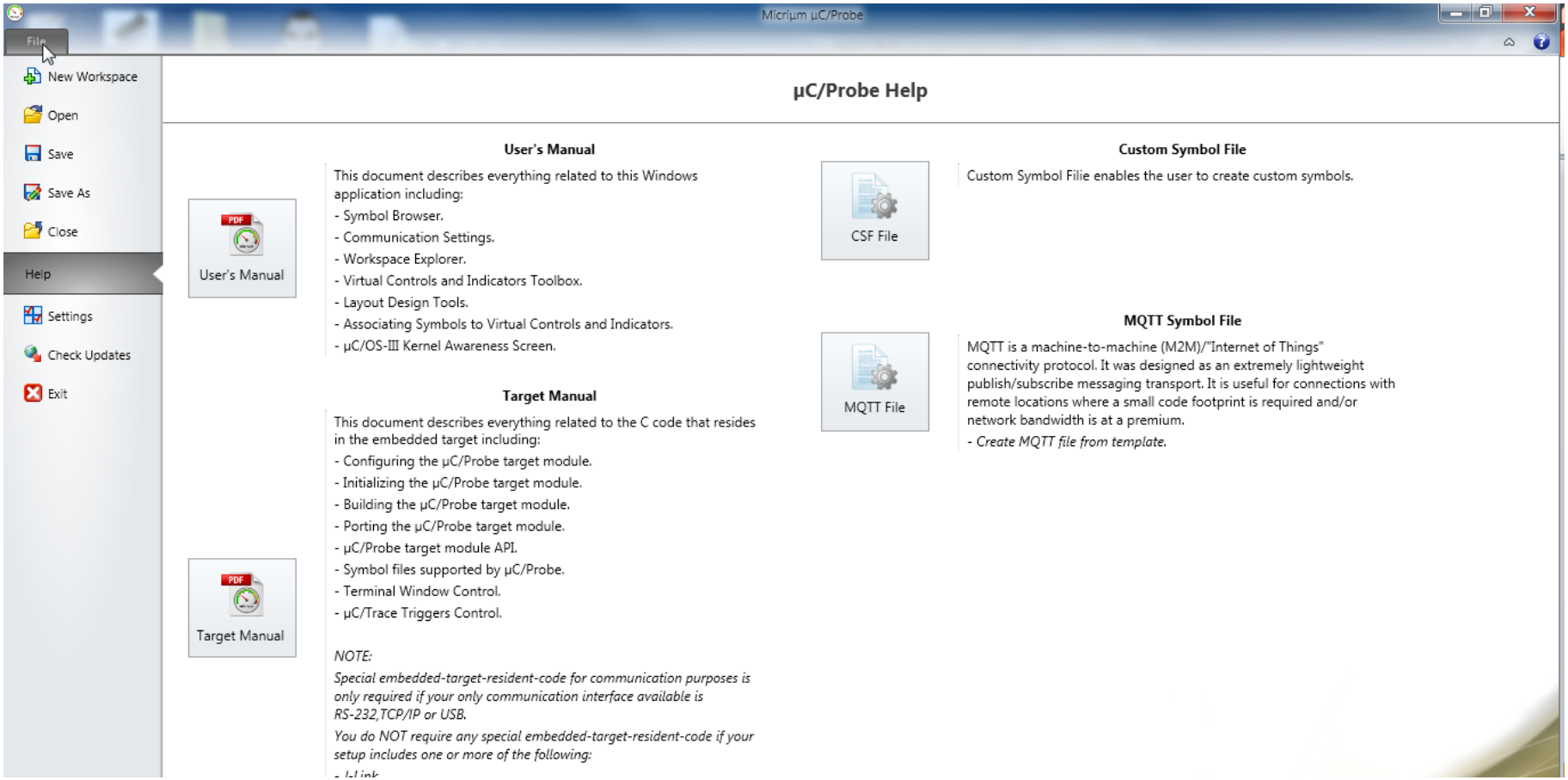


› 点击桌面图标执行uC_Probe



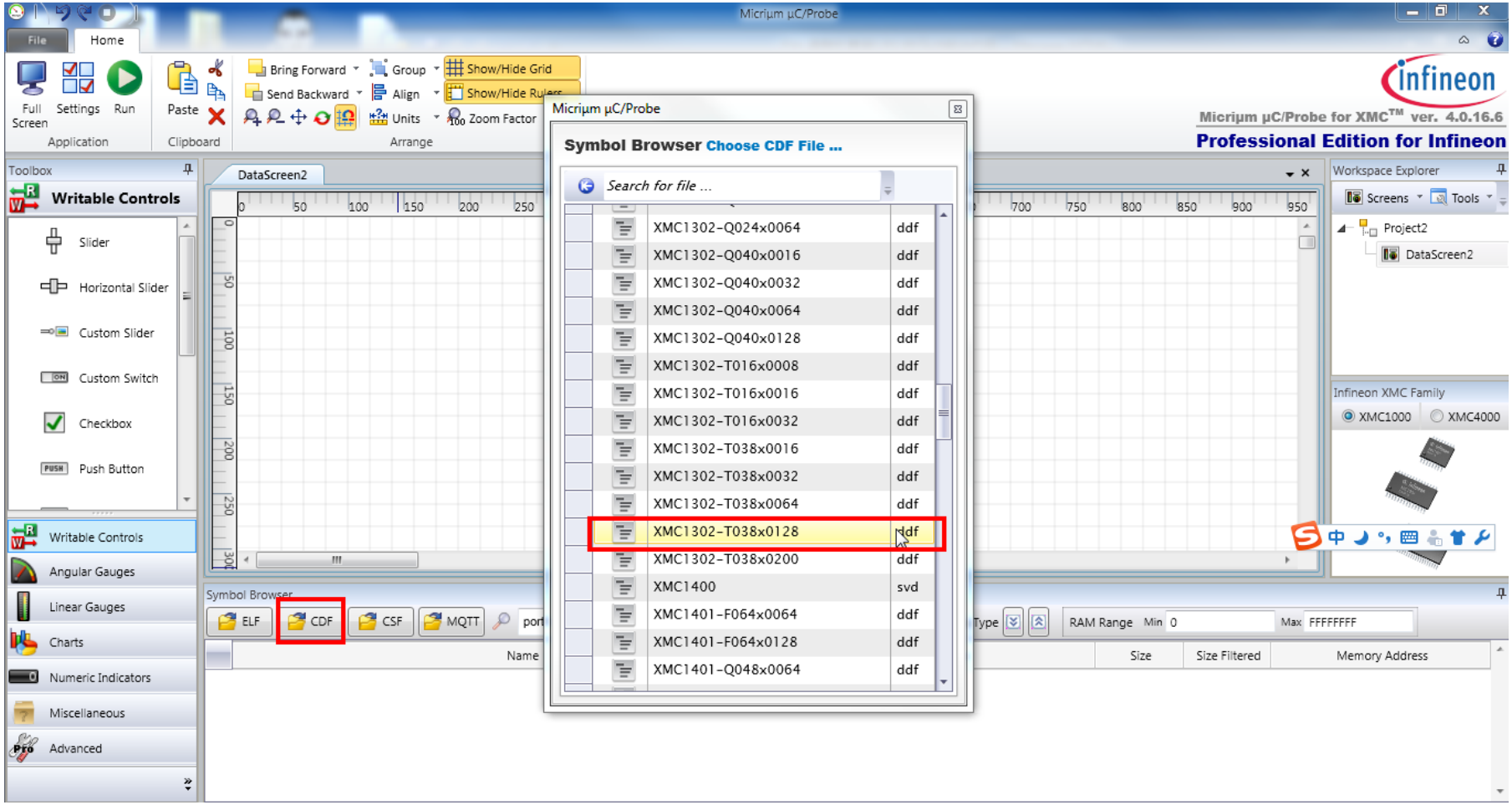
参考文件

› 鼠标点击左上角‘File’，可以看到两个pdf手册描述细节，再点一下File返回



载入芯片文件

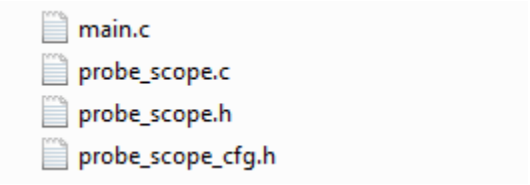
› 点击CDF，在列表里选择对应MCU型号

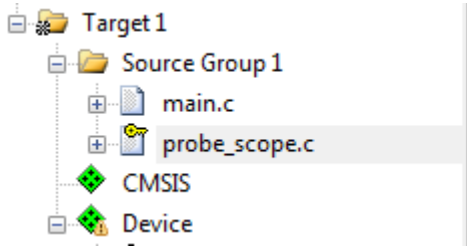


以KEIL例程为例

- 生成一个KEIL的XMC1300的Blinky工程，做测试用途

- 解压缩安装文件里附带的Micrium-Probe-TargetCode.Zip,找到下面路径
  - Micrium-Probe-TargetCode\Micrium\Software\uC-Probe\Target\Scope
- Copy scope下三个文件到测试工程目录，添加C文件到 Keil workspace
- 打开option, 添加路径
  - .\
- Include两个头文件






```
34  #include <xmc_gpio.h>
35
36  #include "probe_scope.h"
37  #include "probe_scope_cfg.h"
38
39  #define LED1 P0_6
40
```

配置cfg文件

› 打开cfg头文件，配置示波器通道，根据自己芯片的内存大小选择通道数和采样数，这里把默认的8通道改成2通道。

probe_scope_cfg.h | main.c* | WDT.c | WDT.h | XMC1300.h | startup_XMC1300.s | probe_scope.c | GPIO.h

```
 8  *
 9  * File     : PROBE_SCOPE_CFG.H
10  * By       : JJL
11  * Version  : V1.00.00
12  ************************************************************************************************************
13  */
14
15
16 /*
17  ************************************************************************************************************
18  *                                                CONFIGURATION
19  ************************************************************************************************************
20  */
21
22  #define  PROBE_SCOPE_MAX_CH                    2     /* The maximum number of channels: [1,8].
23  #define  PROBE_SCOPE_MAX_SAMPLES            1000     /* The maximum number of samples per channel.
24  #define  PROBE_SCOPE_16_BIT_EN                 1     /* The maximum size of each sample is 16-bits: [0,1].
25  #define  PROBE_SCOPE_32_BIT_EN                 1     /* The maximum size of each sample is 32-bits: [0,1].
26  #define  PROBE_SCOPE_SAMPLING_CLK_HZ_DFLT   1000     /* Default freq (Hz) to configure the timer at init.
27  #define  PROBE_SCOPE_IPL                      13
28
```

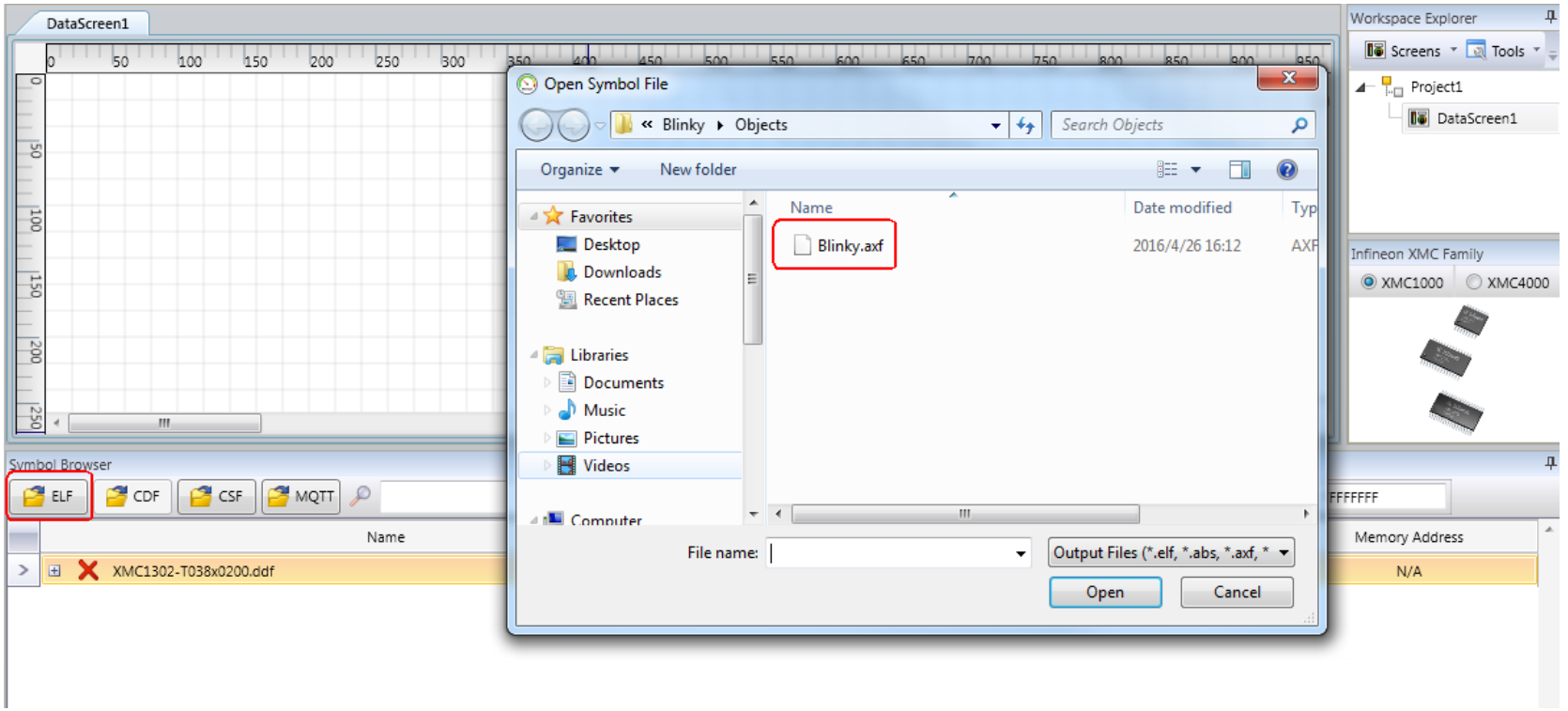
程序内容

- A. 在Main.c里面调用初始化函数ProbeScope_Init(10000).这里的形参10000代表下面的采样程序每0.1mS（10000HZ）就会去采样需要观察的变量。
- B. 修改SysTick_Config()参数，每0.1ms产生一次System Tick的中断
- C. 在System Tick中断服务函数中添加ProbeScope_Sampling()函数
- D. 变量Count和Count1用来测试

```
39  #define LED1 P0_6
40
41  uint16_t Count = 0, Count1 = 0;   D
42
43  void SysTick_Handler(void)
44  {
45    Count++;
46    Count1 += 2;                    D
47    XMC_GPIO_ToggleOutput(LED1);
48    ProbeScope_Sampling();          C
49  }
50
51
52  int main(void)
53  {
54    ProbeScope_Init(10000);         A
55
56    XMC_GPIO_SetMode(LED1, XMC_GPIO_MODE_OUTPUT_PUSH_PULL);
57
58    /* System timer configuration */
59    SysTick_Config(SystemCoreClock/10000);   B
60
61    while(1)
62    {
63      /* Infinite loop */
64    }
65  }
66
```

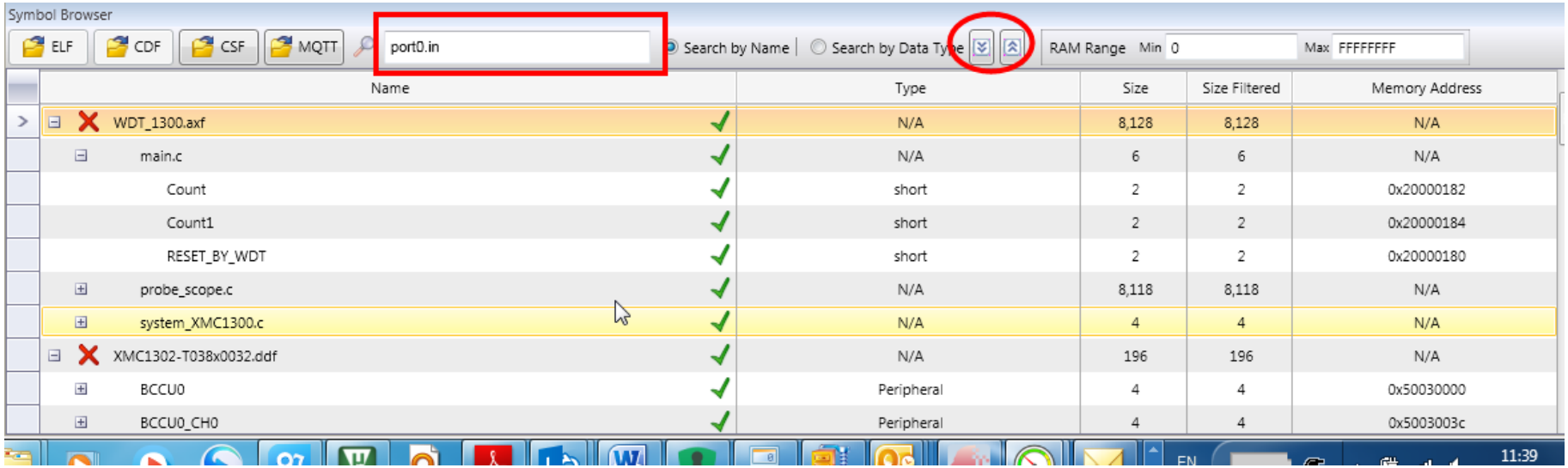
载入对应的编译器仿真文件

- 编译完成后，download到XMC1300BOOTKIT，切换回uC_Probe
- keil用户选择axf文件打开（DAVE用户选择elf，其他找对应的文件后缀）



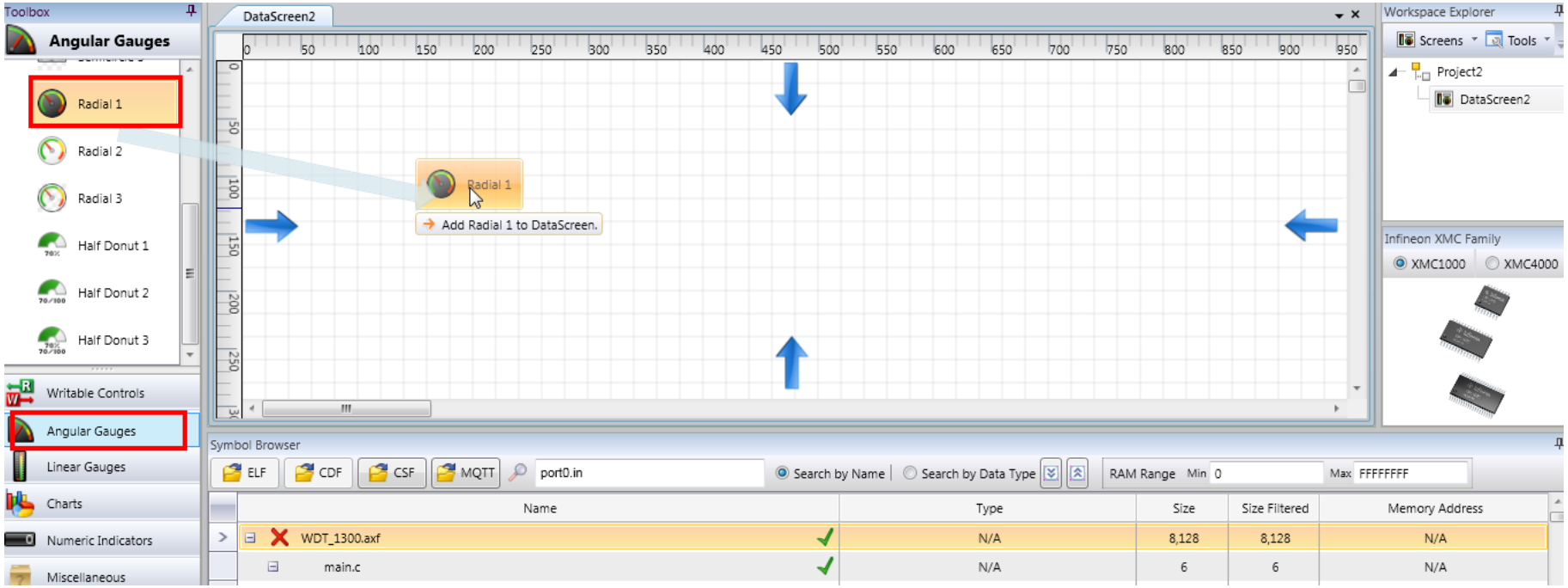
Symbol Browser窗口

- 在Symbol Browser窗口可以看到相应的变量和外设
- 查找窗口输入名字可以快速找到寻找的内容
- 圆圈处可以关闭和展开分支



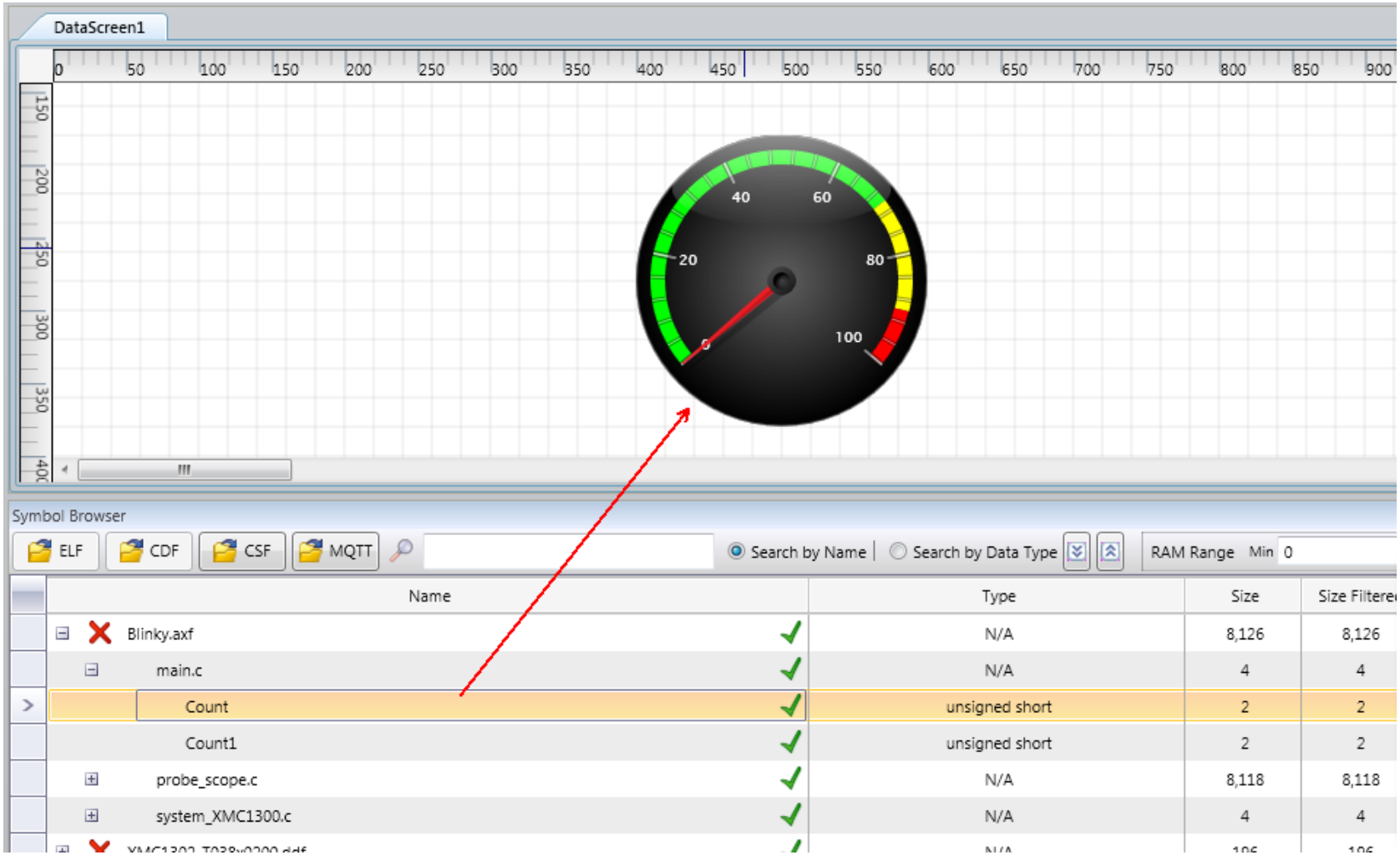
1 增加一个仪表显示

› 选择Angular Gauges，选择一个仪表，拖到datascreen里面



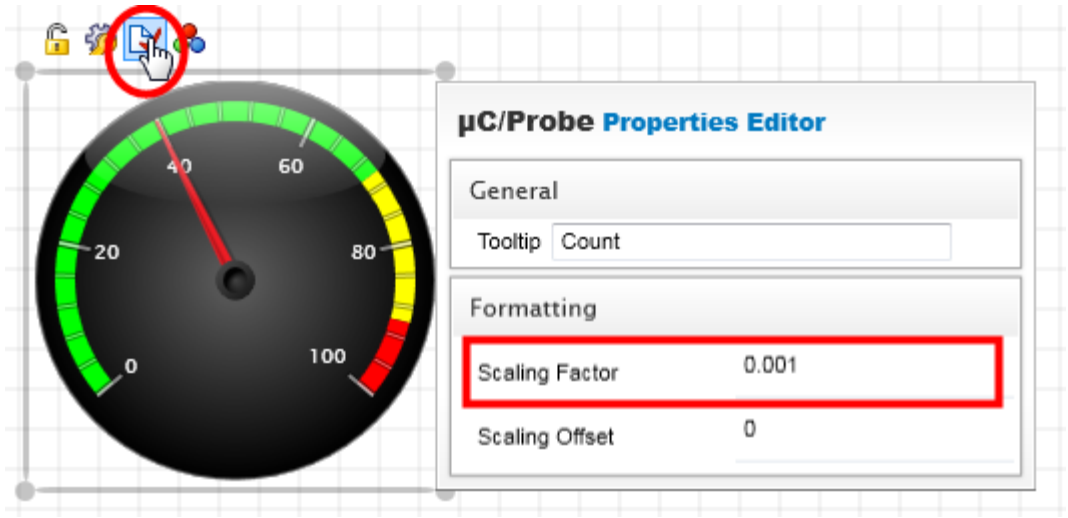
关联变量和仪表

› 在变量里选择一个拖入仪表界面



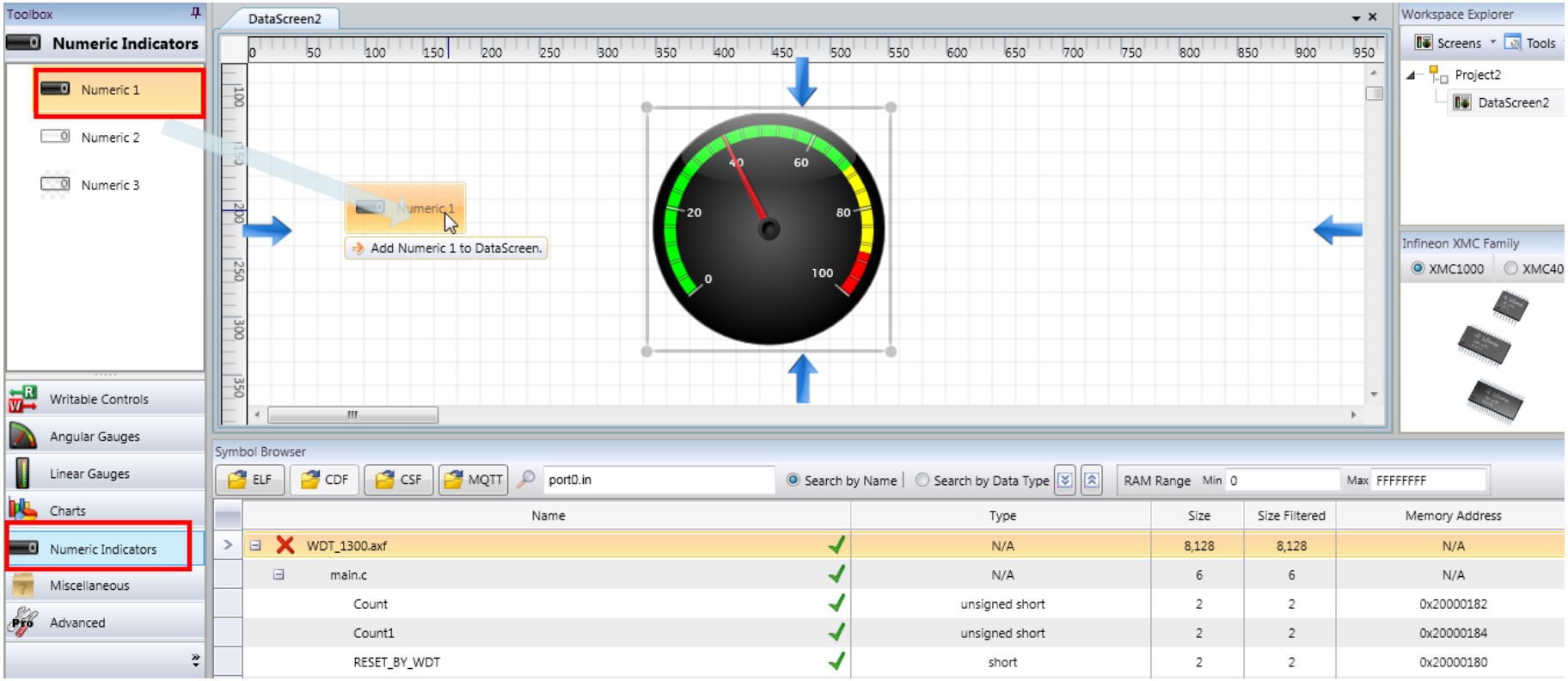
设定显示比例

- 确认Bootkit里面程序已经运行
- 由于count设为uint16_t,从0-65535循环，因此把比例Scaling Factor设为0.001，这样指针转到65到66之间就回到0
- 设定完成后，点击Run，可以看到指针动态从0到65移动



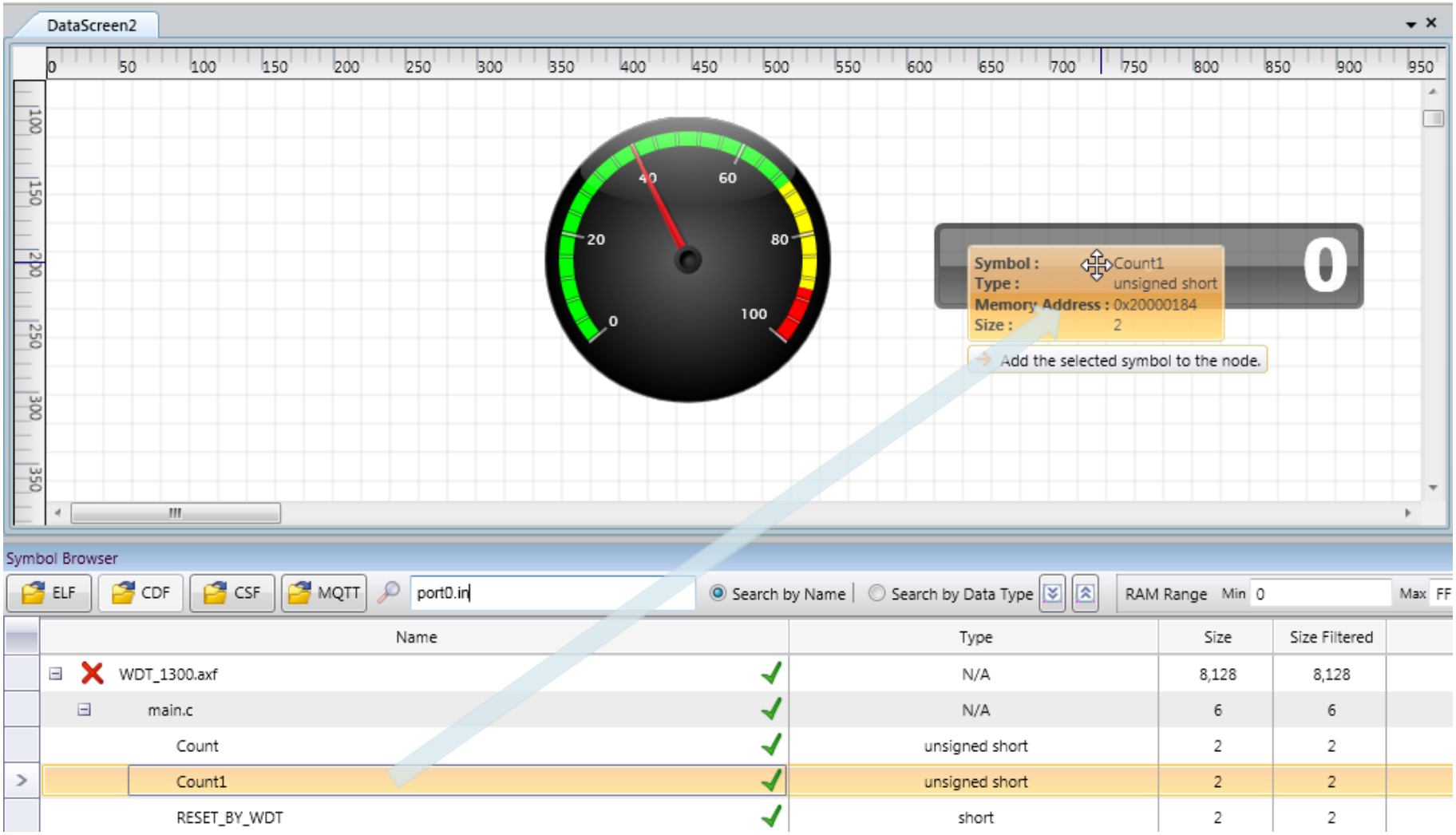
2 增加一个数字表显示

› 选择Numeric Indicators,鼠标拖一个数字表到DataScreen



关联变量到数字表

› 拖变量Count1 到数字表

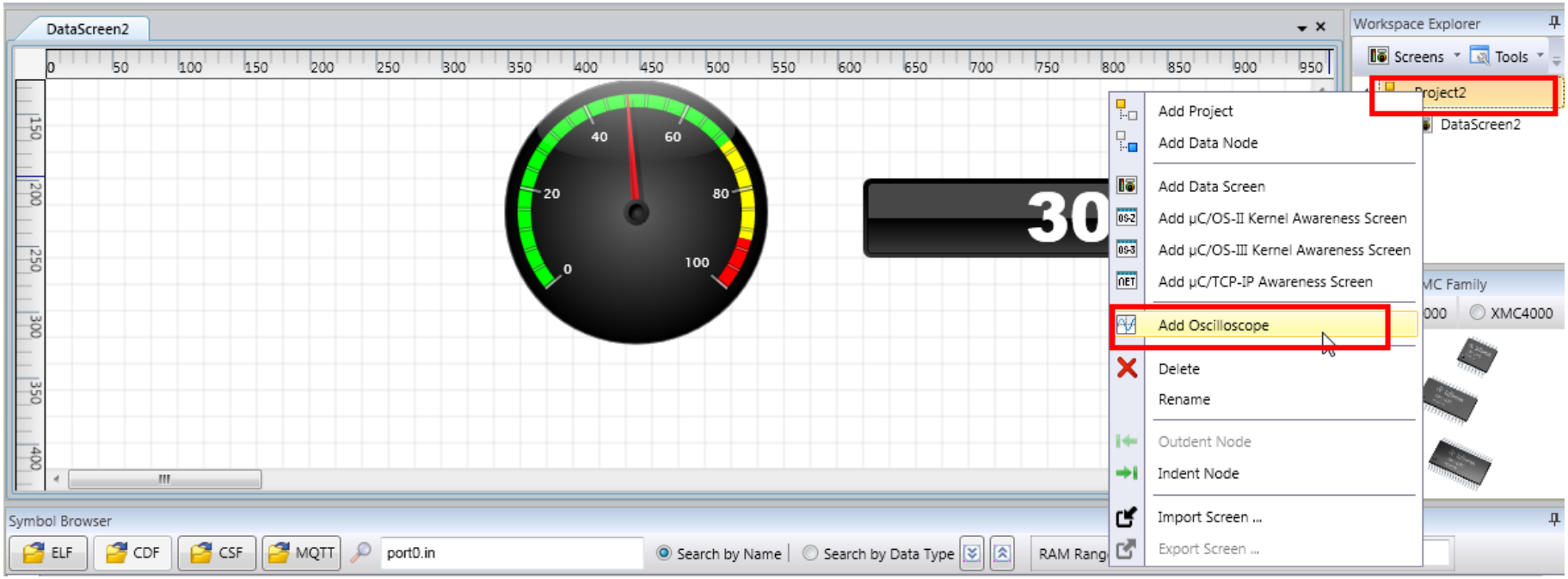


显示结果

› 点击Run后，可以看到变量Count1显示到数字表，不断变化

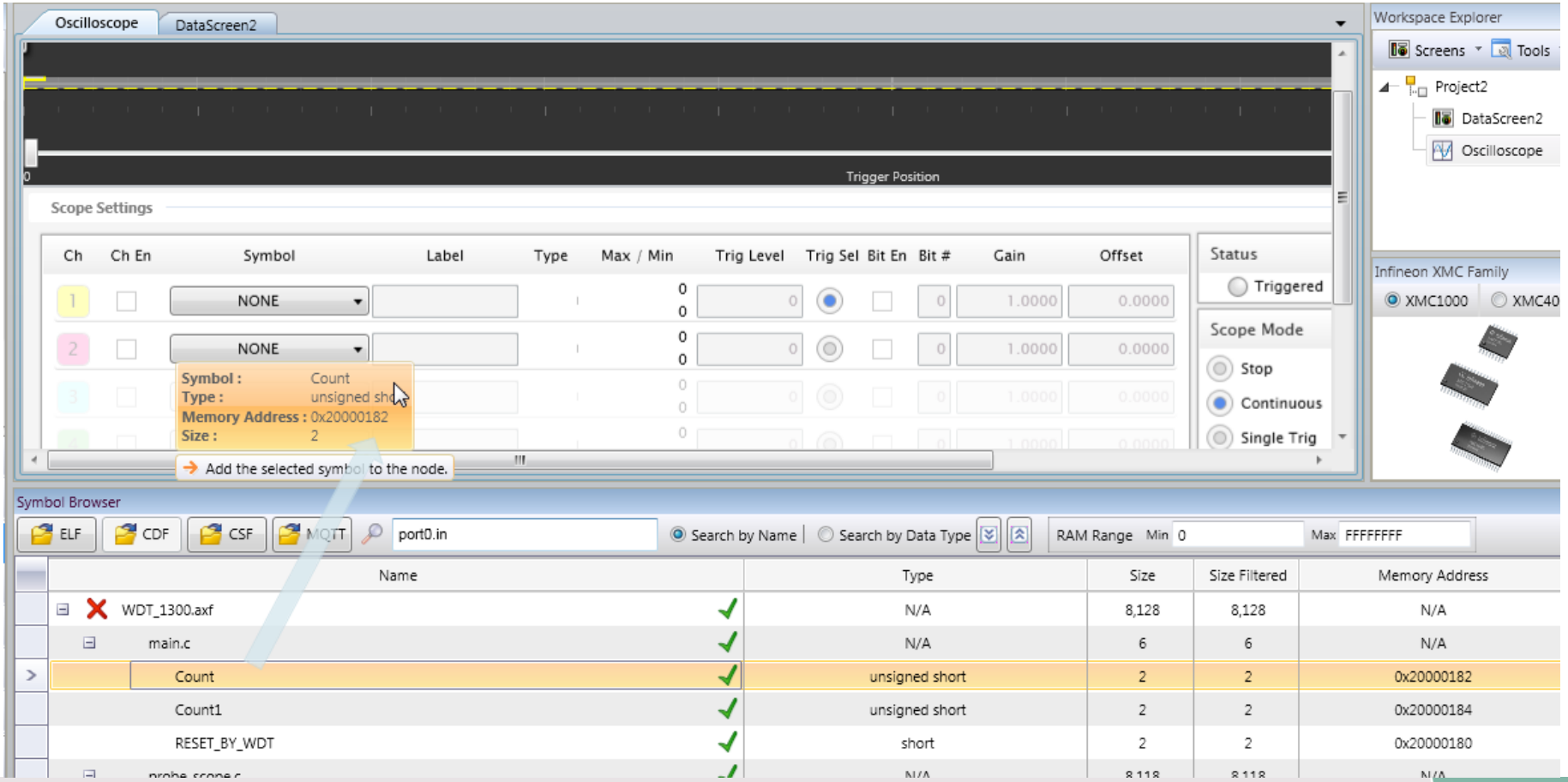
3 增加示波器功能

› 右键点击project，选择Add Oscilloscope



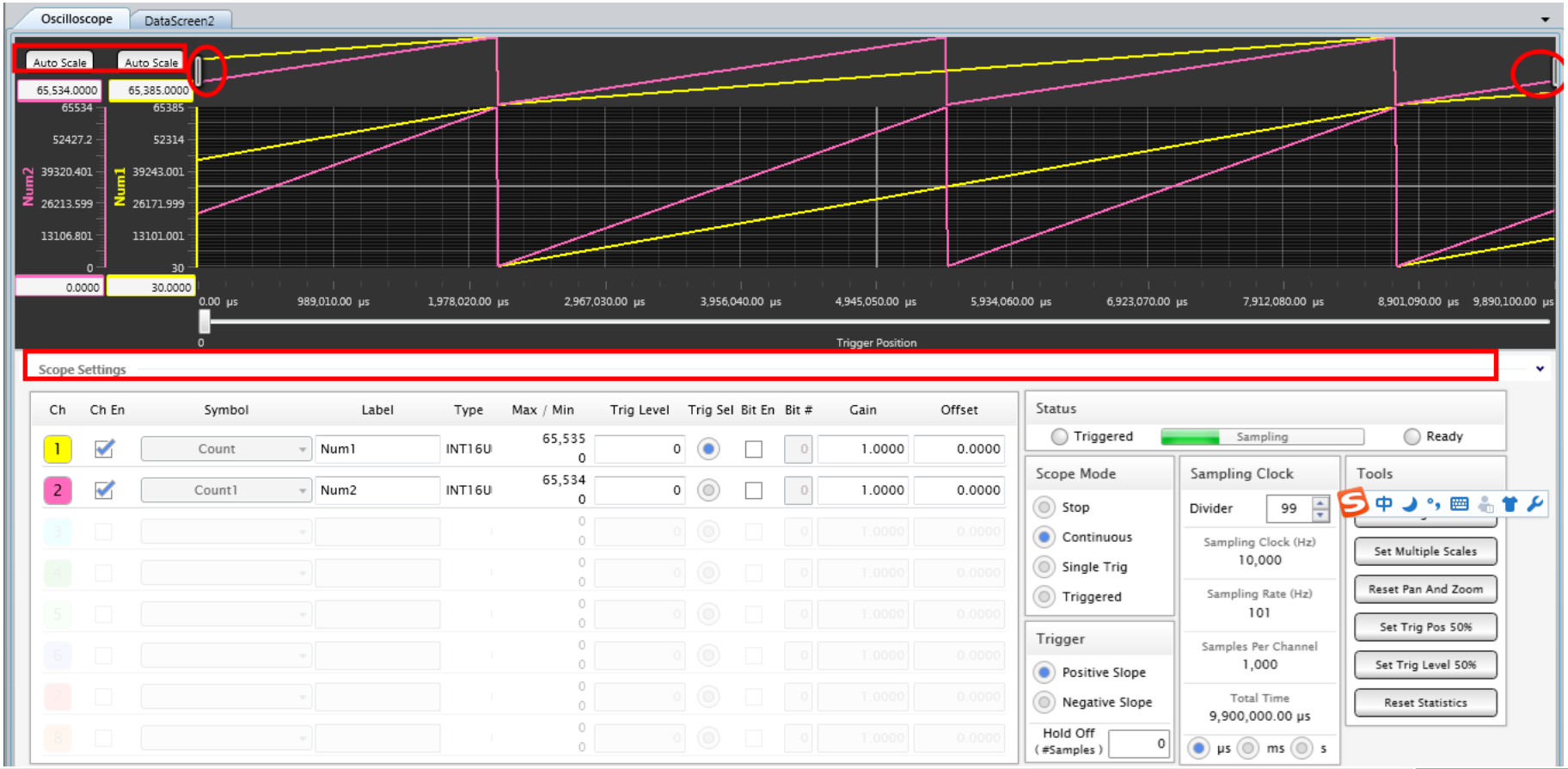
3-1　把变量添加到示波器

› 用鼠标把变量count，count1分别拖到Scope Setting



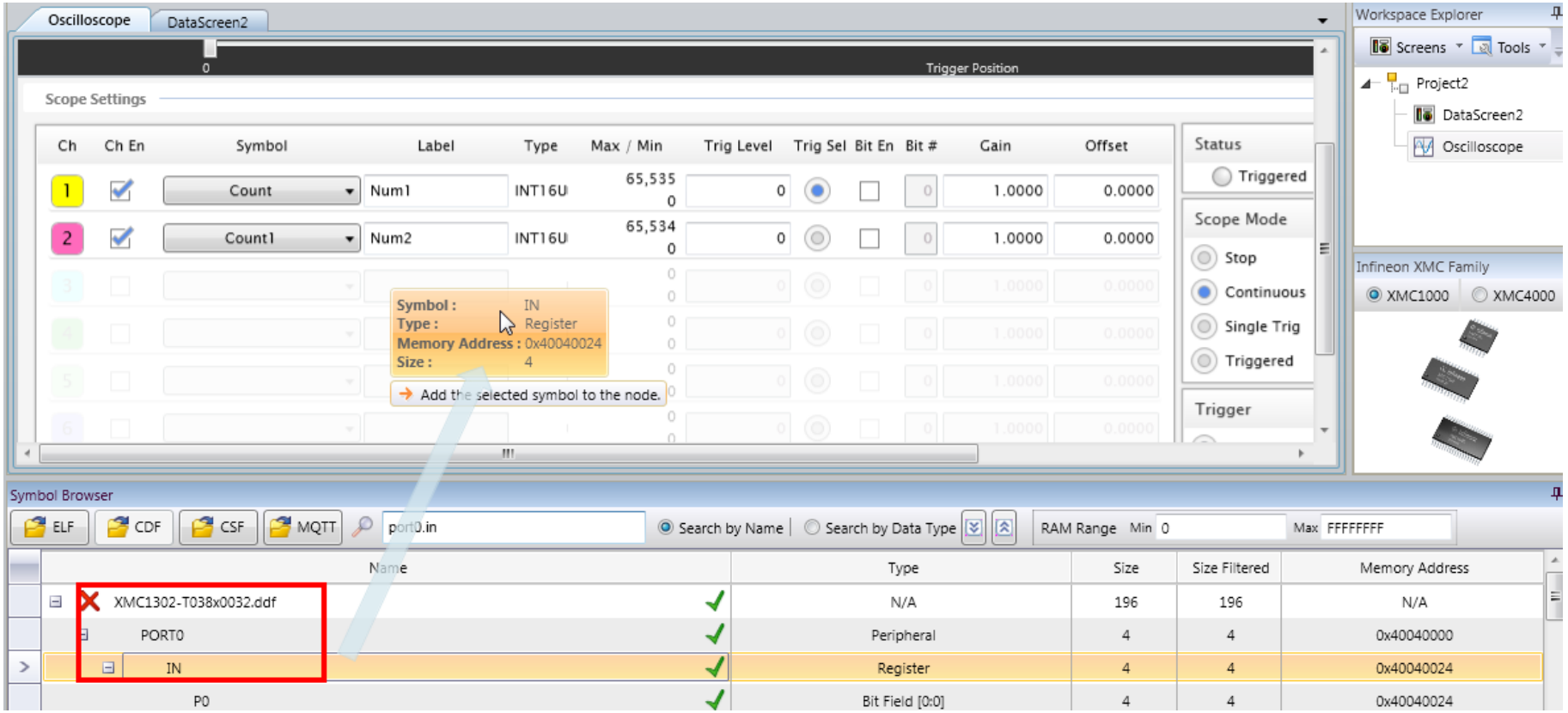
幅度和时间轴调节

- 点击Auto_Scale把幅度调整到合适范围
- 时间轴在Sampling Clock divide调整
- 触发和停止等见操作界面，和正常示波器类似
- 双击scope Setting可以放大示波器到全屏
- 拖拽上方圆圈处可以进行局部放大



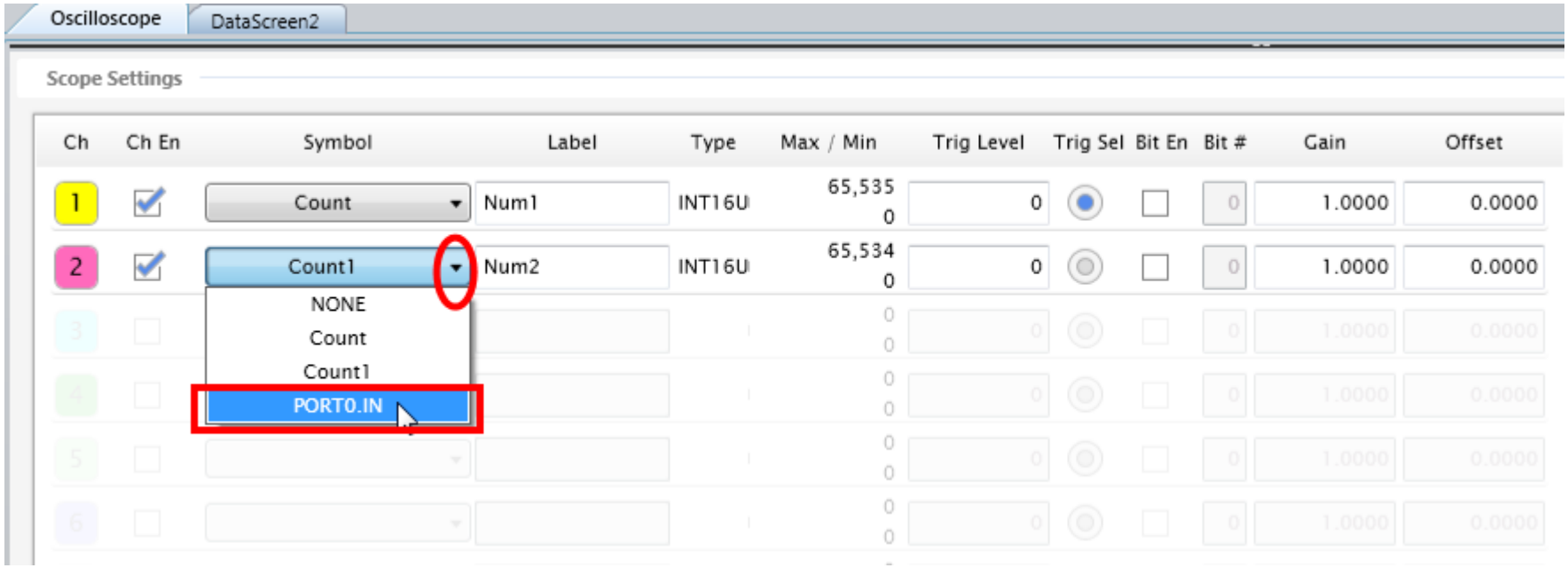
3-2观察寄存器

› 拖拽寄存器PORT0.IN到示波器设置

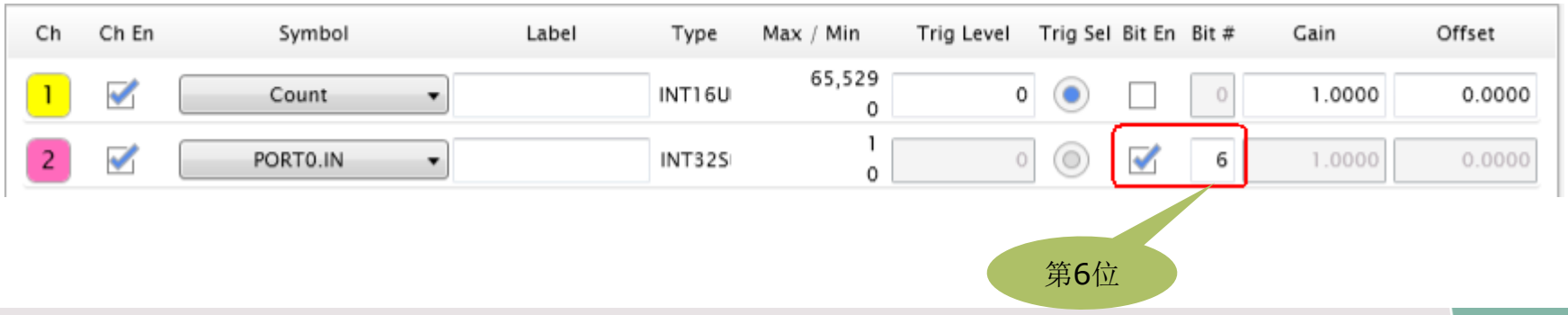


选择通道2观测P0.6

› 下拉菜单选择PORT0.IN

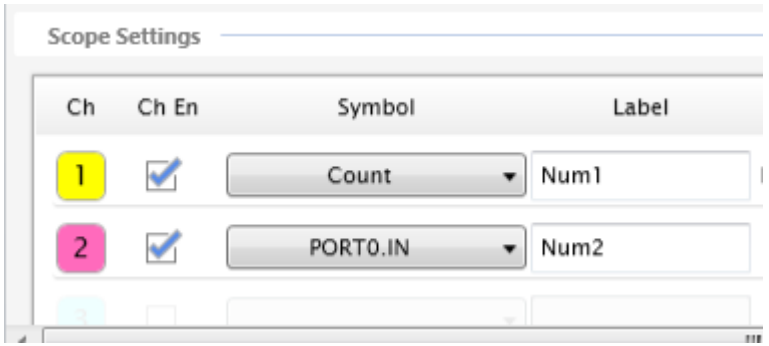


› 选择Bit，第6位



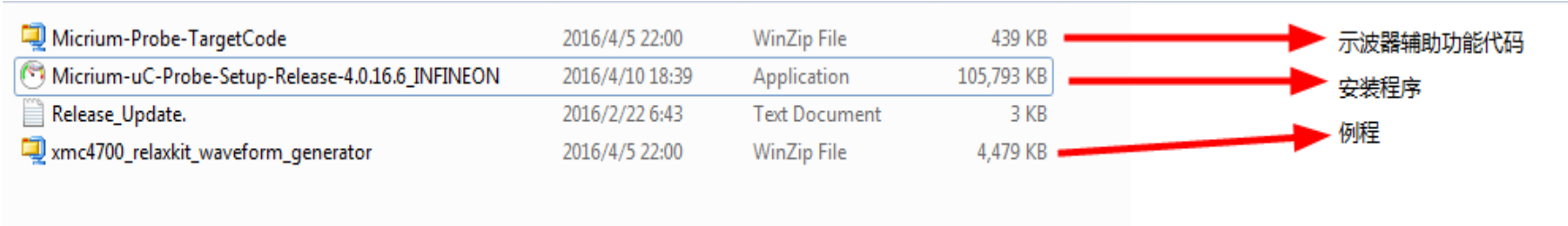
输出结果

```
void SysTick_Handler(void)
{
  Count++;
  Count1 += 2;
  if((Count == 0) || (Count == 32767))
  {
    XMC_GPIO_ToggleOutput(LED1);
  }
  ProbeScope_Sampling();
}
```

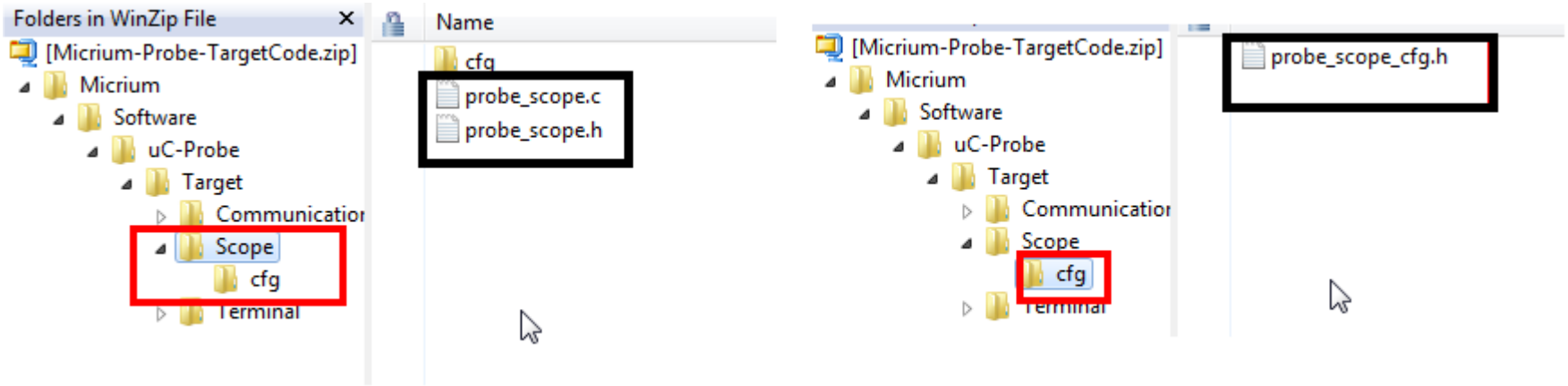


文件资源

› 安装程序包



› 示波器辅助功能代码位置



Part of your life. Part of tomorrow.

Infineon