

实验 10: WWDG and ADC

一、实验器材

硬件: ARM-STM32 开发板, J-Link/St-Link。

软件: Win10, STM32CubeIDE, HAL library

二、实验要求

1. Use the ADC to get the measurement of internal temprature sensor(practice of the last lab);
2. Use KEY1 to refresh the WWDG and output the corresponding data when the early wakeup interrupt is triggered.

三、实验过程

1. 根据 lab 课内容, 新建项目, 添加文件, 添加依赖
2. 在多通道单转换模式下激活 ADC1
3. 在 while 循环里面打印温度和循环次数
4. 实现 HAL_WWDG_EarlyWakeupCallback 函数去刷新 WWDG 在 KEY1 被按下的时候
5. Build, 生成 “.hex” 文件, 用 “FlyMcu” 下载到板子上。
6. 测试, 测试所有情况, 观察实验板是否和预期显示一致。

四、实验结果

软件代码（只需用户实现功能的主要代码部分）：

在 while 循环里面打印温度和循环次数：

```

while (1)
{
    /* USER CODE END WHILE */

    /* USER CODE BEGIN 3 */
    HAL_ADC_PollForConversion(&hadc1, HAL_MAX_DELAY);
    float raw =(float)HAL_ADC_GetValue(&hadc1);
    raw = raw * 3.3 / 4096;
    T = ((1.43 - raw) / 4.3) + 25;

    i++;
    sprintf(msg, "%f %d\r\n", T,i);
    HAL_UART_Transmit(&huart1, (uint8_t*)msg, strlen(msg), HAL_MAX_DELAY);
    HAL_Delay(100);
}

```

实现 HAL_WWDG_EarlyWakeupCallback 函数去刷新 WWDG 在 KEY1 被按下的时候

以及 EXTI

```

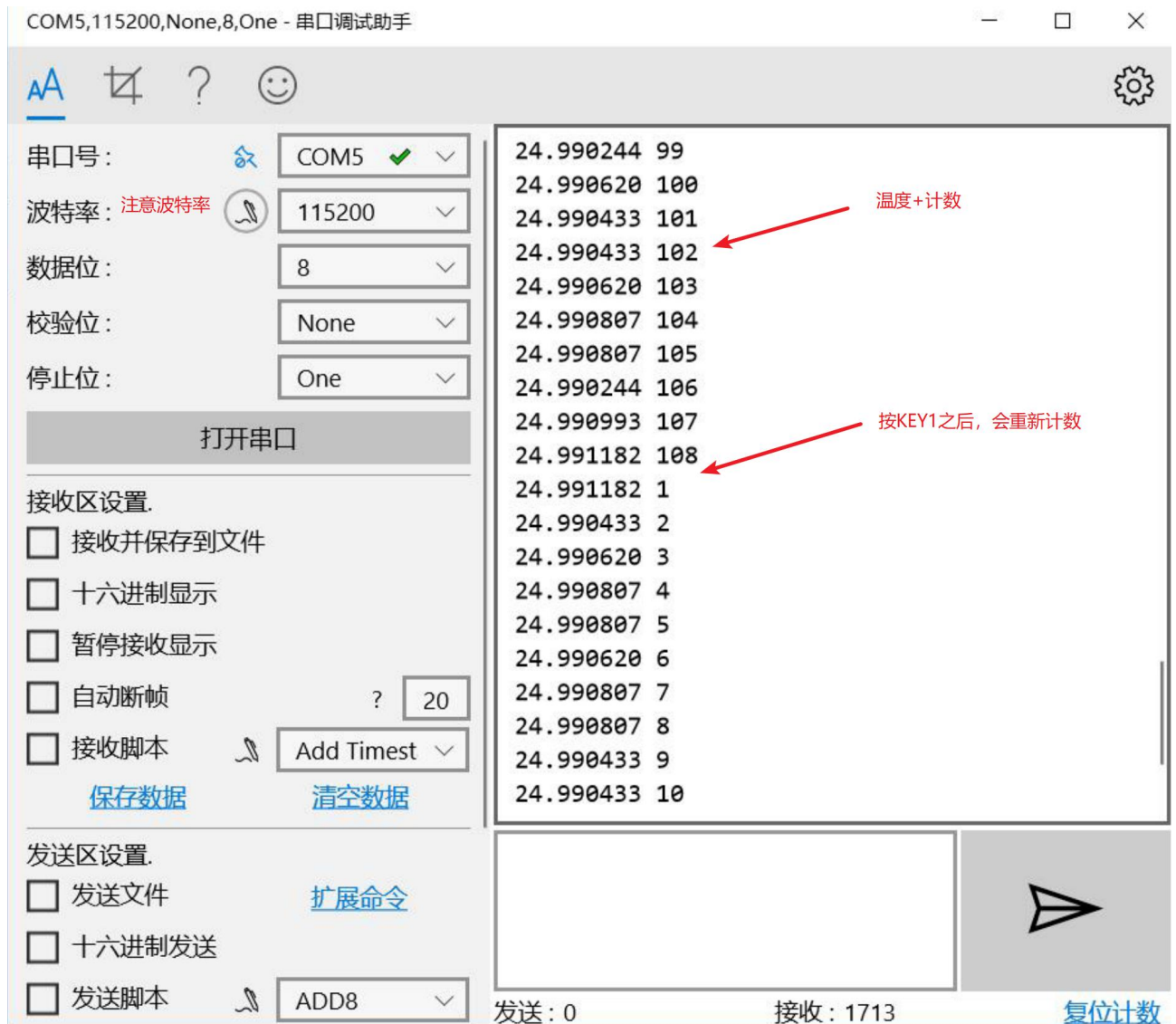
/* USER CODE BEGIN 0 */
void HAL_WWDG_EarlyWakeupCallback(WWDG_HandleTypeDef *hwwdg) { HAL_WWDG_Refresh(hwwdg); }
void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin) {
    HAL_Delay(100);
    switch (GPIO_Pin) {
        case KEY1_Pin:
            if (HAL_GPIO_ReadPin(KEY1_GPIO_Port, KEY1_Pin) == GPIO_PIN_RESET) {
                HAL_WWDG_Refresh(&hwwdg);
            }
            break;
        default:
            break;
    }
}

HAL_StatusTypeDef HAL_WWDG_Refresh(WWDG_HandleTypeDef *hwwdg)
{
    /* Write to WWDG CR the WWDG Counter value to refresh with */
    WRITE_REG(hwwdg->Instance->CR, (hwwdg->Init.Counter));

    /* Return function status */
    return HAL_OK;
}

```

实际验证附图（含开发板状态拍照、仿真截图等）：



五、实验总结

简述在实验过程中出现的问题，解决的过程和结果，及其他需要说明的情况。

1. 每次上板前记得重新 build 一次。

2. Debug 后 Programming 需要在 Post-build steps 的 Command 下加上 arm-none-eabi-objcopy

"\${ProjName}.elf" -O ihex "\${ProjName}.hex"

3. 用“FlyMcu”下载到板子和串口调试工具只能有一个连着，不能同时连着，会报串口占用的错误。

打印浮点数可能会有空行，cubuIDE 会有提示，需要在 CMakeLists 里：添加 `-u _printf_float \`
`SET (COMMON_FLAGS`

`"-mcpu=cortex-m3 ${FPU_FLAGS} -mthumb -mthumb-interwork -ffunction-sections -fdata-sections \ -g`
`-fno-common -fmessage-length=0 -specs=nosys.specs -specs=nano.specs -u _printf_float")` 来修改编译器设置。

4. 如果出现乱码, 先看看串口调试工具的编码格式有没有改成 utf-8; 如果还是乱码, 检查数组是否越界, 长度是否正确, 这也会导致乱码。

5. 一些库函数以及 cubuIDE 的操作知识的时候, 去翻翻之前的课件, 或者去 google 上查找一些关于 eclipse 的解决方案就可以基本解决目前遇到的问题。