

## 前言

本应用指南主要介绍如何使用硬件 CRC 外设及 CRC 原理相关内容。

支持型号列表：

支持型号	AT32F4 系列
------	-----------

目录

1	简介 .....	7
2	CRC 标准 .....	8
3	CRC 计算方法 .....	9
3.1	CRC 算法 .....	9
3.2	AT32 CRC 外设 .....	9
3.2.1	数据格式逆转 .....	9
3.2.2	AT32 CRC 计算 .....	10
4	代码配置解析 .....	12
4.1	函数接口 .....	12
4.2	CRC 配置流程 .....	12
5	案例 CRC 计算 .....	15
5.1	功能简介 .....	15
5.2	资源准备 .....	15
5.3	软件设计 .....	15
5.4	实验效果 .....	16
5.4.1	宏定义选择 CRC32_TEST_1 .....	16
5.4.2	宏定义选择 CRC32_TEST_2 .....	17
5.4.3	宏定义选择 CRC32_TEST_3 .....	18
5.4.4	宏定义选择 CRC32_TEST_4 .....	19
5.4.5	宏定义选择 CRC32_TEST_5 .....	20
5.4.6	宏定义选择 CRC16_TEST_1 .....	21
5.4.7	宏定义选择 CRC16_TEST_2 .....	22
5.4.8	宏定义选择 CRC16_TEST_3 .....	23
5.4.9	宏定义选择 CRC16_TEST_4 .....	24
5.4.10	宏定义选择 CRC16_TEST_5 .....	25

5.4.11	宏定义选择 CRC8_TEST_1 .....	26
5.4.12	宏定义选择 CRC8_TEST_2 .....	27
5.4.13	宏定义选择 CRC8_TEST_3 .....	28
5.4.14	宏定义选择 CRC8_TEST_4 .....	28
5.4.15	宏定义选择 CRC8_TEST_5 .....	30
5.4.16	定义选择 CRC16_TEST_8BIT_DATA(MODBUS).....	31
5.4.17	宏定义选择 CRC32_TEST_8BIT_DATA .....	32
5.4.18	宏定义选择 CRC32_TEST_16BIT_DATA .....	33
6	文档版本历史 .....	34

## 表目录

表 1. 常用 CRC 标准.....	8
表 2. 多项式关系释义 .....	8
表 3. 库函数接口.....	12
表 4. 配置流程示例 .....	13
表 5. CRC 计算案例 data buffer .....	15
表 6. 宏定义配置.....	15
表 7. 参数配置 .....	16
表 8. 参数配置 .....	17
表 9. 参数配置 .....	18
表 10. 参数配置 .....	19
表 11. 参数配置.....	20
表 12. 参数配置 .....	21
表 13. 参数配置 .....	22
表 14. 参数配置 .....	23
表 15. 参数配置 .....	24
表 16. 参数配置 .....	25
表 17. 参数配置 .....	26
表 18. 参数配置 .....	27
表 19. 参数配置 .....	28
表 20. 参数配置 .....	28
表 21. 参数配置 .....	30
表 22. 参数配置 .....	31
表 23. 参数配置 .....	32
表 24. 参数配置 .....	33
表 25. 文档版本历史 .....	34

## 图目录

图 1. CRC 计算过程 .....	9
图 2. 16-bit 数据按字节逆转图示 .....	10
图 3. 16-bit 数据按半字逆转图示 .....	10
图 4. CRC 外设计算流程图 .....	10
图 5. CRC 配置流程图 .....	13
图 6. 上位机配置 .....	16
图 7. 串口打印结果 .....	16
图 8. 上位机配置 .....	17
图 9. 串口打印结果 .....	17
图 10. 上位机配置 .....	18
图 11. 串口打印结果 .....	18
图 12. 上位机配置 .....	19
图 13. 串口打印结果 .....	19
图 14. 上位机配置 .....	20
图 15. 串口打印结果 .....	20
图 16. 上位机配置 .....	21
图 17. 串口打印结果 .....	21
图 18. 上位机配置 .....	22
图 19. 串口打印结果 .....	22
图 20. 上位机配置 .....	23
图 21. 串口打印结果 .....	23
图 22. 上位机配置 .....	24
图 23. 串口打印结果 .....	24
图 24. 上位机配置 .....	25
图 25. 串口打印结果 .....	25
图 26. 上位机配置 .....	26
图 27. 串口打印结果 .....	26
图 28. 上位机配置 .....	27
图 29. 串口打印结果 .....	27
图 30. 上位机配置 .....	28

图 31.串口打印结果..... 28

图 32.上位机配置 ..... 29

图 33.串口打印结果..... 29

图 34.上位机配置 ..... 30

图 35.串口打印结果..... 30

图 36.上位机配置 ..... 31

图 37.串口打印结果..... 31

图 38.上位机配置 ..... 32

图 39.串口打印结果..... 32

图 40.上位机配置 ..... 33

图 41.串口打印结果..... 33

# 1 简介

循环冗余校验（CRC）是一种检测数据错误的技术，主要用在数据通信和数据存储方面，但它只能检测到传输或存储的数据是否有误，没有将错误纠正的功能。

AT32 MCU 片上 CRC 外设采用 CRC-32 标准（多项式为：

$X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^8+X^7+X^5+X^4+X^2+X+1$ ，简记式：0x4C11DB7），每次输入输出 32-bit 数据，并支持输入输出数据按设定格式 bit 逆转。

## 2 CRC 标准

在进行 CRC 计算时会有指定的多项式或标准来对验证方和被验证方进行约束。以下将对标准、多项式和简记式进行描述，常用的标准有以下几种（但不仅限于）：

表 1. 常用 CRC 标准

标准	多项式	简记式
CRC-8	$X^8+X^2+X+1$	0x07
CRC-12	$X^{12}+X^{11}+X^3+X^2+X+1$	0x80F
CRC-16	$X^{16}+X^{15}+X^2+1$	0x8005
CRC-CCITT	$X^{16}+X^{12}+X^5+1$	0x1021
CRC-32	$X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^8+X^7+X^5+X^4+X^2+X+1$	0x4C11DB7

CRC-12 码通常用来传送 6-bit 字符串，CRC-16 码及 CRC-CCITT 码通常用来传送 8-bit 字符串，CRC-32 码通常用来进行点对点的同步传输。

不同的 CRC 标准具有不一样的多项式，其中多项式中  $X^Y$  可以简单理解为 bit Y 为 1，以 CRC-8 举例，多项式的释义如下：

表 2. 多项式关系释义

bit	8	7	6	5	4	3	2	1	0
多项式	$X^8$	0	0	0	0	0	$X^2$	X	1
binary	1	0	0	0	0	0	1	1	1
Hex	0x107								
简记式	0x07								

不同 CRC 标准下其最高位都默认为 1，通常简记式的表示方法会忽略最高位进行标记。多项式非固定某个值或某个标准，只要计算方及校验方合理采用一致的多项式即可。



## 3 CRC 计算方法

### 3.1 CRC 算法

CRC 的计算方法采用异或运算方式。在进行 CRC 计算时可能存在目标数据串位宽比多项式位宽更短，并且计算过程希望保证到目标数据串中的每一位都有被处理到，因此通常的做法是在目标位数据串的后面加上简记式位宽  $W$  个“0”，再进行异或求余，示例如下：

多项式 = 1011（宽度  $W=3$ ）

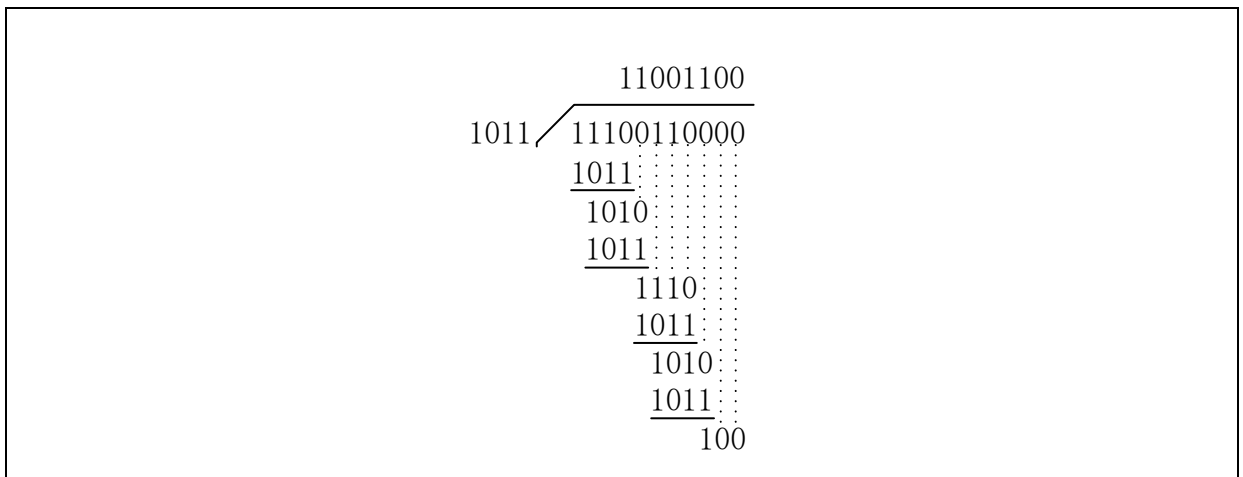
目标数据串 = 11100110

Bit 流 = 目标数据串 +  $W$  个“0” = 11100110000

CRC result = 100

计算过程图示如下：

图 1. CRC 计算过程



### 3.2 AT32 CRC 外设

AT32 CRC 外设特性：

- 采用 CRC-32 标准。
- 一次 CRC 计算需要 4 个 HCLK。
- 输入输出数据格式可逆转。
- 数据写入及结果读出都通过同一个数据寄存器（32 位 CRC\_DT）。

#### 3.2.1 数据格式逆转

AT32 CRC 外设支持输入数据按字节、半字、字格式逆转，输出数据按字节格式逆转的特性可配，默认情况下都不逆转。数据格式逆转指对指定数据按一定的数据格式进行高低位逆转的处理方式。16 位数据按字节和半字的逆转图示如下，其余格式与此类似：

图 2. 16-bit 数据按字节逆转图示

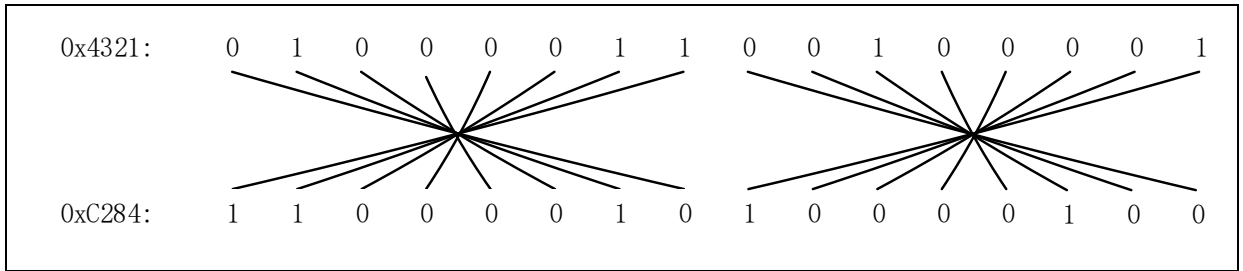
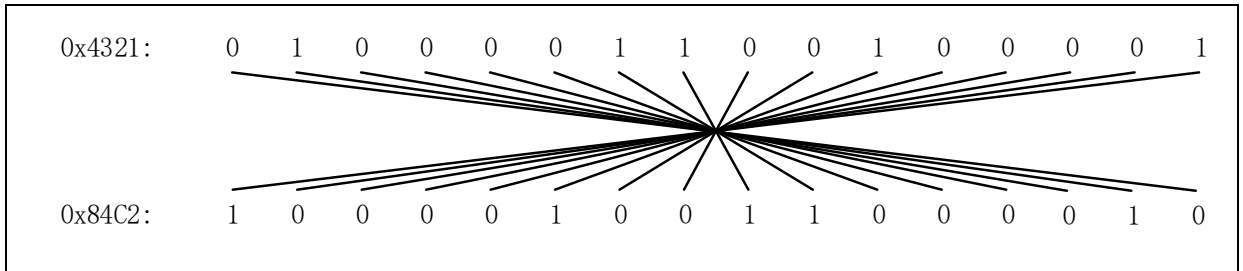


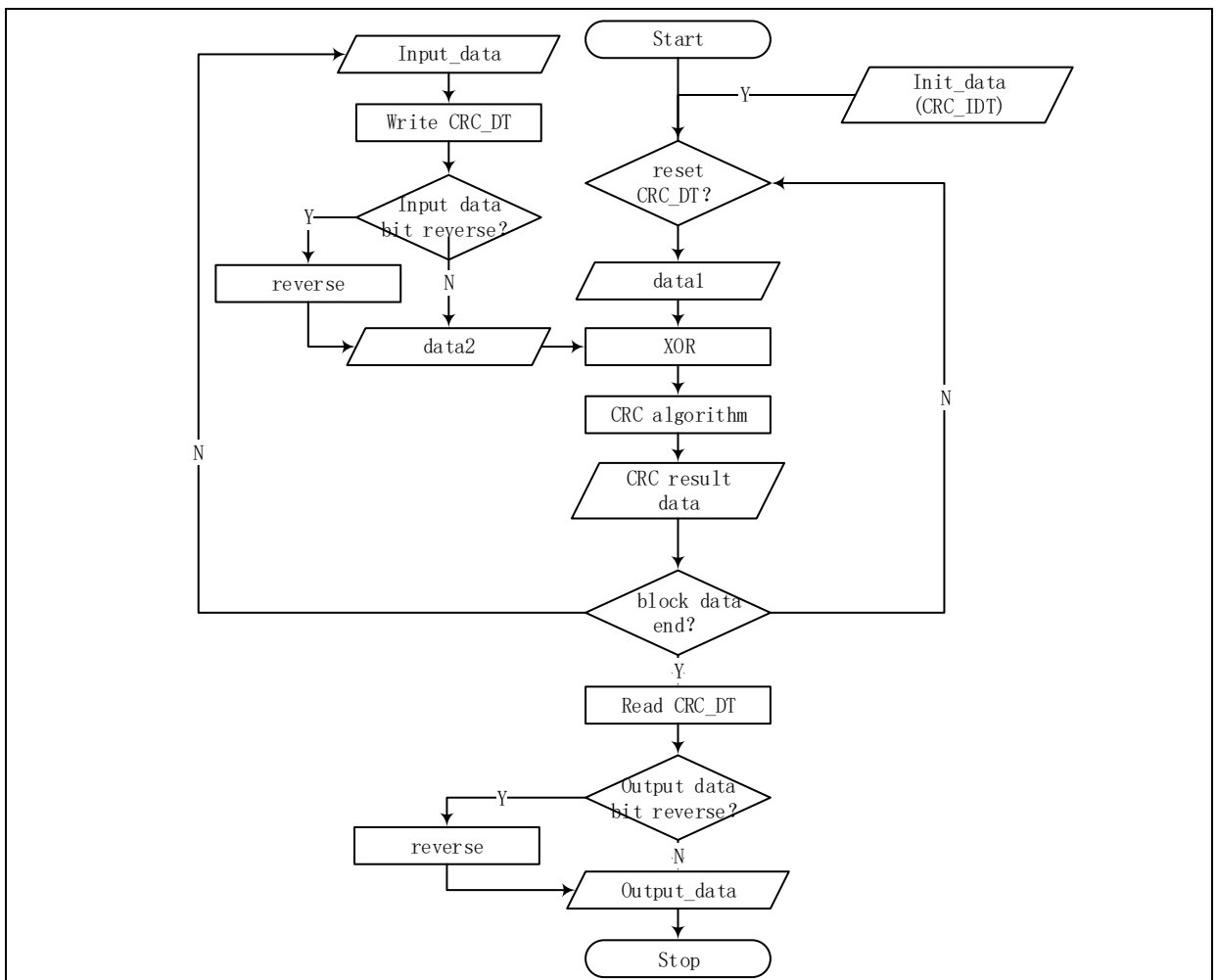
图 3. 16-bit 数据按半字逆转图示



### 3.2.2 AT32 CRC 计算

AT32 CRC 外设在进行计算时的大致流程如下图所示：

图 4. CRC 外设计算流程图



其流程描述如下：

- 首先配置好 CRC 后，复位 CRC\_DT 会将 CRC 初始化寄存器中的值更新到 CRC\_DT 中。
- 将输入数据按 32-bit 数据格式写入到 CRC\_DT 寄存器，硬件会按输入数据逆转设定是否对数据进行逆转。
- 数据逆转处理后的输入数据与 CRC\_DT 中的结果数据进行异或计算。
- 异或计算后的结果再按循环冗余校验算法进行处理，并得出每次计算结果。
- 循环输入数据直到所有数据计算完毕。
- 读 CRC\_DT 获取计算结果，硬件会按输出数据逆转设定是否对数据逆转并给到 CRC\_DT 寄存器。

## 4 代码配置解析

以下将以库函数接口为核心来对 CRC 配置流程和方法进行说明。

### 4.1 函数接口

各系列产品对应提供的 BSP 中对 CRC 外设的设置部分已封装好接口函数以供调用，以下罗列出常用的函数接口，各函数的具体参数及返回值类型等请参考 `at32xxxx_crc.c/h` 文件。

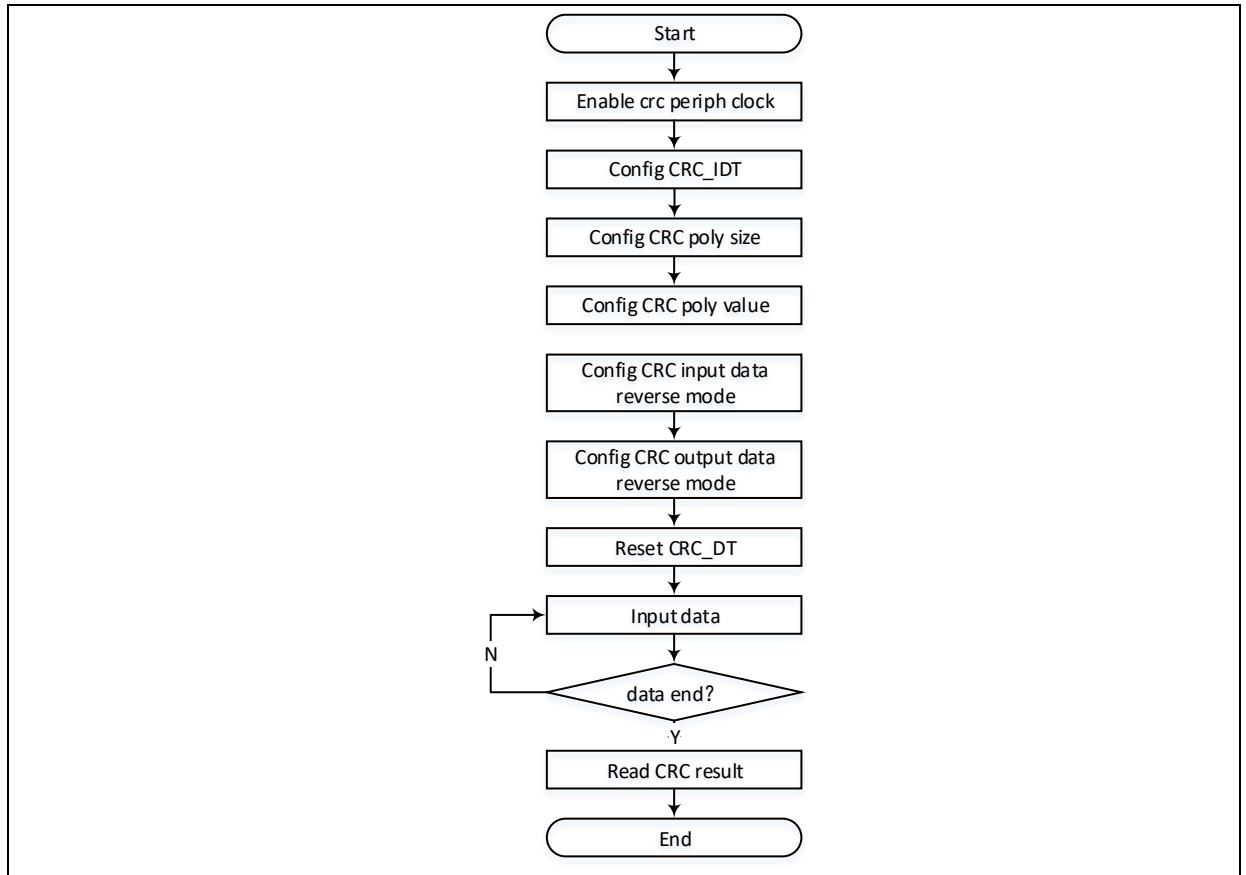
表 3. 库函数接口

<pre>/* 数据寄存器 CRC_DT 的复位函数，默认值为 0xFFFFFFFF */ void crc_data_reset(void);</pre>
<pre>/* 输入一个 32-bit 的数据进行 CRC 计算并返回结果 */ uint32_t crc_one_word_calculate(uint32_t data);</pre>
<pre>/* 按数据块进行 CRC 计算并返回结果 */ uint32_t crc_block_calculate(uint32_t *pbuffer, uint32_t length);</pre>
<pre>/* 读取当前 CRC 数据寄存器 */ uint32_t crc_data_get(void);</pre>
<pre>/* 设置 CRC 通用寄存器，可当做独立的 8-bit 数据存储使用 */ void crc_common_data_set(uint8_t cdt_value);</pre>
<pre>/* 读取 CRC 通用寄存器存储的数据值 */ uint8_t crc_common_data_get(void);</pre>
<pre>/* 初始化寄存器 CRC_IDT 值设置 */ void crc_init_data_set(uint32_t value);</pre>
<pre>/* 按数据类型设置输入数据 bit 逆转 */ void crc_reverse_input_data_set(crc_reverse_input_type value);</pre>
<pre>/* 输出数据 bit 逆转设置 */ void crc_reverse_output_data_set(crc_reverse_output_type value);</pre>
<pre>/* CRC 多项式值设置 */ void crc_poly_value_set(uint32_t value);</pre>
<pre>/* CRC 多项式值读取 */ uint32_t crc_poly_value_get(void);</pre>
<pre>/* CRC 多项式宽度设置 */ void crc_poly_size_set(crc_poly_size_type size);</pre>
<pre>/* CRC 多项式宽度读取 */ crc_poly_size_type crc_poly_size_get(void);</pre>

### 4.2 CRC 配置流程

按常规应用来讲解 CRC 配置流程，其内容可大致分为如下步骤：

图 5. CRC 配置流程图



步骤大致描述如下：

- 首先在进行 CRC 操作之前应开启 CRC 的外设时钟。
- 设置初始化寄存器（CRC\_IDT）值，在对数据寄存器（CRC\_DT）进行软复位时，会将初始化寄存器里的值直接写入到 CRC\_DT 寄存器作为初始值，如不设置，此寄存器默认值为 0xFFFFFFFF。
- 设置多项式宽度。
- 设置多项式值。
- 设置输入输出数据逆转模式。
- Reset CRC\_DT 后将数据按 32-bit 格式逐个写入到 CRC\_DT 进行计算。
- 读 CRC\_DT 寄存器，检查计算结果。

示例代码如下：

表 4. 配置流程示例

```
crm_periph_clock_enable(CRM_CRC_PERIPH_CLOCK, TRUE); // 使能 CRC 外设时钟
crc_init_data_set(0x11223344);                        // 设置初始化寄存器为 0x11223344
crc_poly_size_set(CRC_POLY_SIZE_32B);                 // 设置多项式宽度为 32 位
crc_poly_value_set(0x4C11DB7);                       // 设置多项式值为 0x4C11DB7
crc_reverse_output_data_set(CRC_REVERSE_OUTPUT_DATA); // 配置输出数据逆转
crc_reverse_input_data_set(CRC_REVERSE_INPUT_BY_BYTE); // 配置输入数据字节逆转
crc_data_reset();                                     // 复位 CRC_DT
```

```
crc_value = crc_block_calculate((uint32_t *)data_buffer, BUFFER_SIZE); // block data CRC 计算并返回结果
```

## 5 案例 CRC 计算

### 5.1 功能简介

在 main.c 里选择不同的宏定义，测试不同配置下的 CRC 计算值，通过串口打印最终的 CRC 计算结果。

### 5.2 资源准备

- 1) 硬件环境:  
对应产品型号的 AT-START BOARD
- 2) 软件环境  
AN0109\_SourceCode\utilities\AN0109\_demo

### 5.3 软件设计

- 1) 配置流程
  - 软件选择不同宏定义。
  - 开启 CRC 外设时钟。
  - 配置输入输出数据逆转模式。
  - 输入需要计算的数据块。
  - 判断计算结果。
- 2) 代码介绍  
CRC 计算数据块

表 5. CRC 计算案例 data buffer

```
uint32_t data_buffer_32[] = {0x11223344, 0x55667788};  
uint16_t data_buffer_16[] = {0x1122, 0x3344, 0x5566};  
uint8_t data_buffer_8[] = {0x11, 0x22, 0x33, 0x44, 0x55};
```

宏定义描述

表 6. 宏定义配置

算法	多项式
CRC32_TEST_1	32bit数据，计算CRC32
CRC32_TEST_2	32bit数据，计算CRC32
CRC32_TEST_3	32bit数据，计算CRC32
CRC32_TEST_4	32bit数据，计算CRC32
CRC32_TEST_5	32bit数据，计算CRC32
CRC16_TEST_1	16bit数据，计算CRC16
CRC16_TEST_2	16bit数据，计算CRC16
CRC16_TEST_3	16bit数据，计算CRC16
CRC16_TEST_4	16bit数据，计算CRC16
CRC16_TEST_5	16bit数据，计算CRC16
CRC8_TEST_1	8bit数据，计算CRC8
CRC8_TEST_2	8bit数据，计算CRC8

CRC8_TEST_3	8bit数据, 计算CRC8
CRC8_TEST_4	8bit数据, 计算CRC8
CRC8_TEST_5	8bit数据, 计算CRC8
CRC16_TEST_8BIT_DATA	8bit数据, 计算CRC16
CRC32_TEST_8BIT_DATA	8bit数据, 计算CRC32
CRC32_TEST_16BIT_DATA	16bit数据, 计算CRC32

## 5.4 实验效果

在 main.c 里选择不同的宏定义, 测试不同配置下的 CRC 计算值, 通过串口打印最终的 CRC 计算结果。

### 5.4.1 宏定义选择 CRC32\_TEST\_1

表 7. 参数配置

算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC32	0x4C11DB7	MSB First	0xFFFFFFFF	0x00000000	0x11223344 0x55667788	0xDABFB5CD

图 6.上位机配置

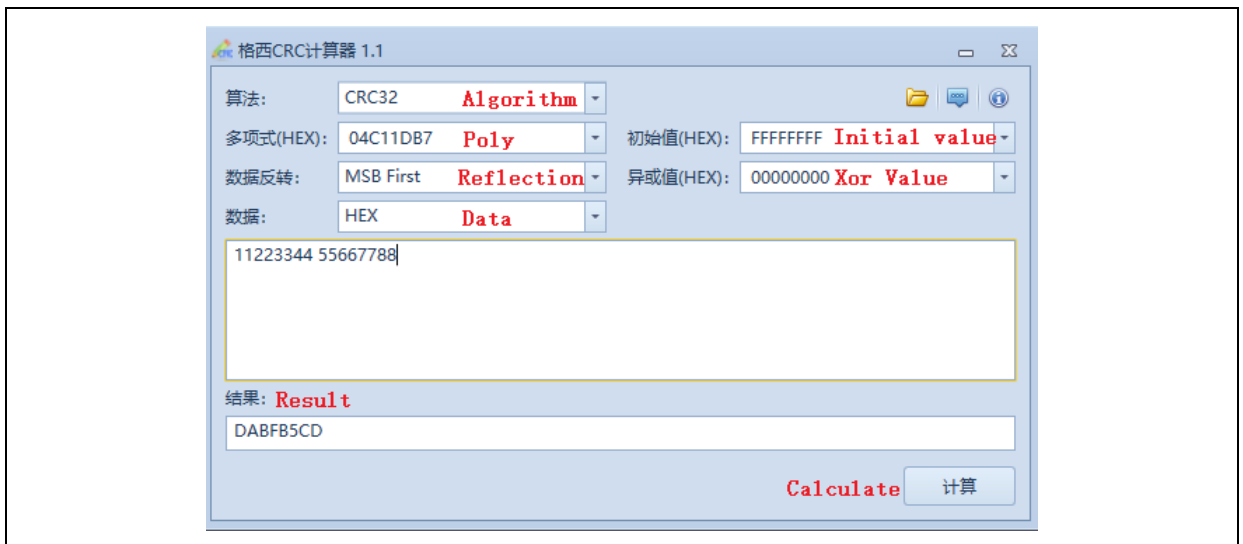
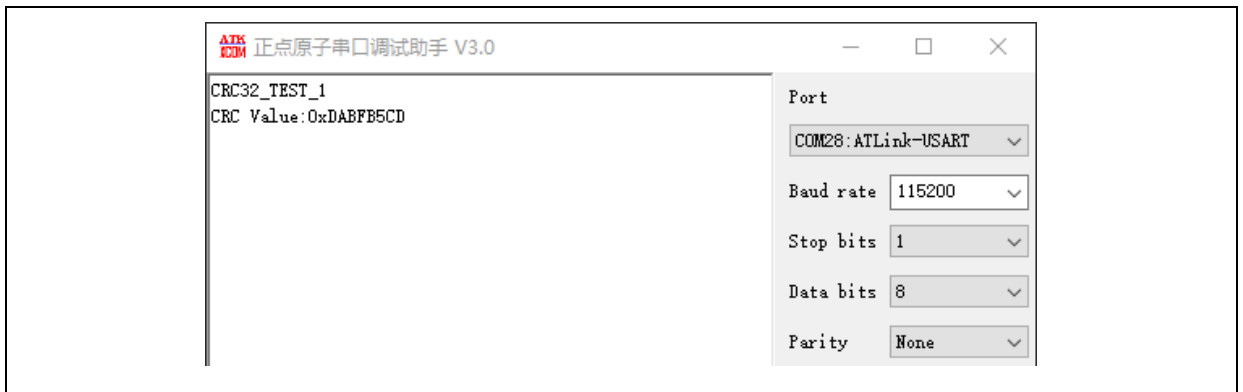


图 7.串口打印结果





### 5.4.2 宏定义选择 CRC32\_TEST\_2

表 8. 参数配置

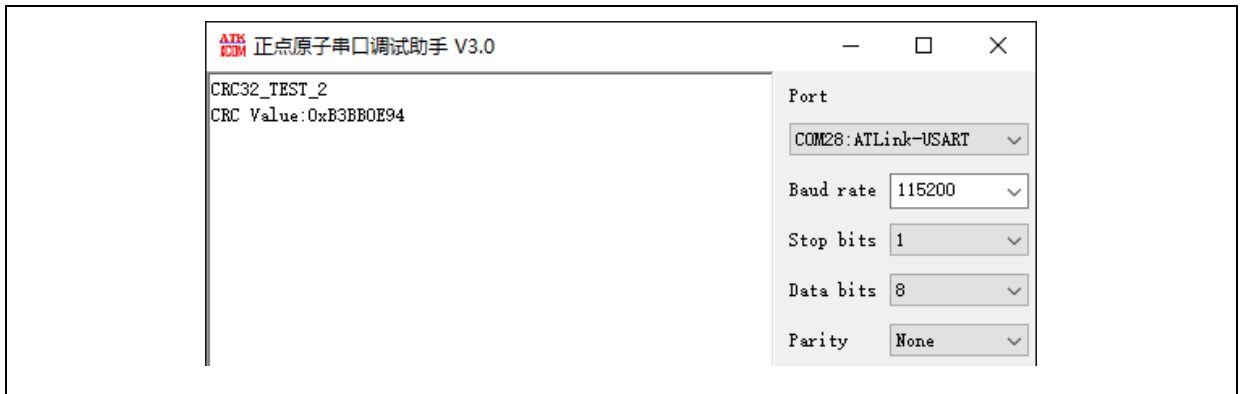
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC32	0x4C11DB7	MSB First	0x00000000	0x00000000	0x11223344 0x55667788	0xB3BB0E94

图 8.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 9.串口打印结果



### 5.4.3 宏定义选择 CRC32\_TEST\_3

表 9. 参数配置

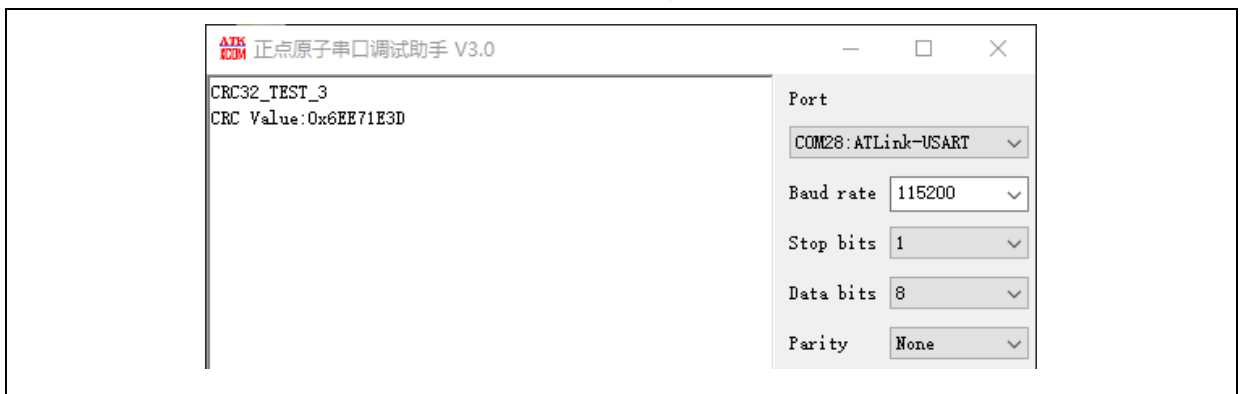
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC32	0x4C11DB7	LSB First	0xFFFFFFFF	0x00000000	0x11223344 0x55667788	0x6EE71E3D

图 10.上位机配置



注: CRC计算器英文界面请参考[5.4.1](#)章节。

图 11.串口打印结果



### 5.4.4 宏定义选择 CRC32\_TEST\_4

表 10. 参数配置

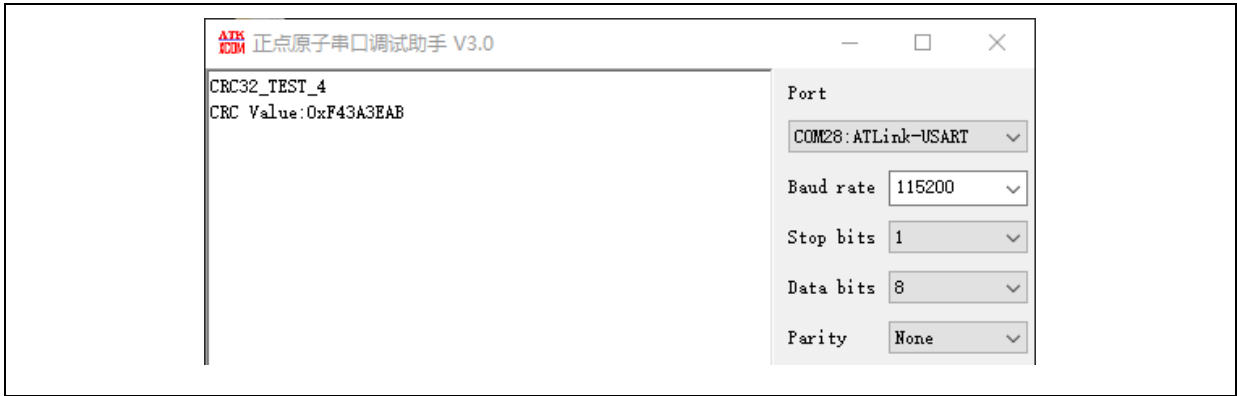
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC32	0x4C11DB7	LSB First	0x00000000	0x00000000	0x11223344 0x55667788	0xF43A3EAB

图 12.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 13.串口打印结果



### 5.4.5 宏定义选择 CRC32\_TEST\_5

表 11. 参数配置

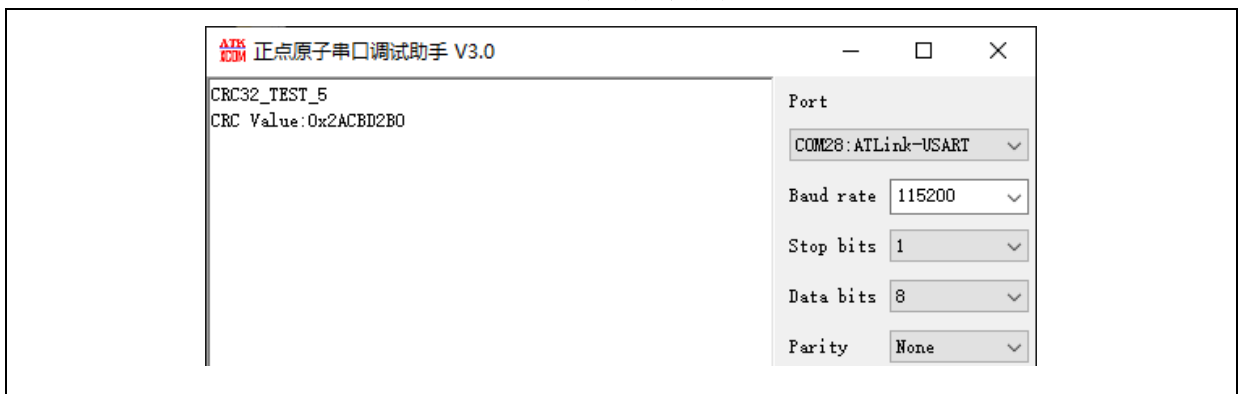
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC32	0xA833982B	MSB First	0xFFFFFFFF	0x00000000	0x11223344 0x55667788	0x2ACBD2B0

图 14.上位机配置



注: CRC计算器英文界面请参考[5.4.1](#)章节。

图 15.串口打印结果



### 5.4.6 宏定义选择 CRC16\_TEST\_1

表 12. 参数配置

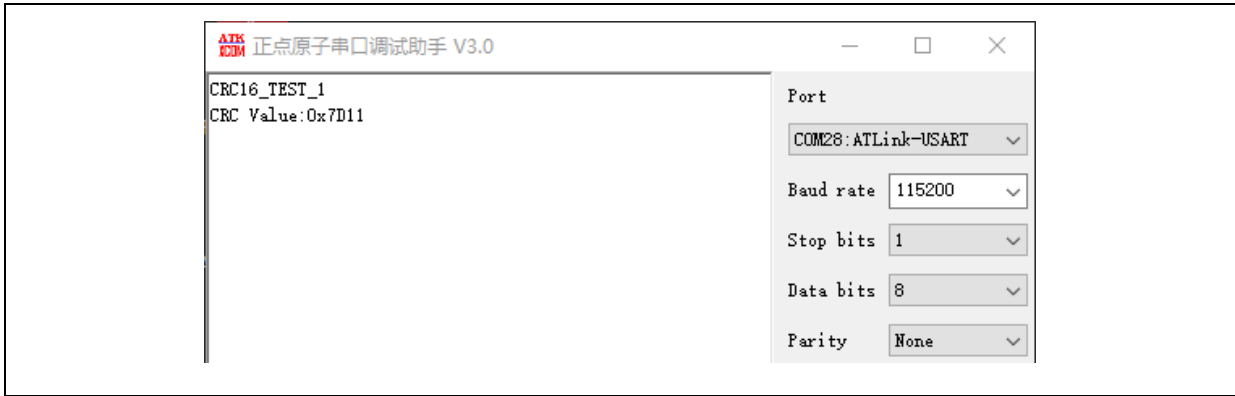
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC16	0x8005	MSB First	0xFFFF	0x0000	0x1122 0x3344	0x7D11

图 16.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 17.串口打印结果

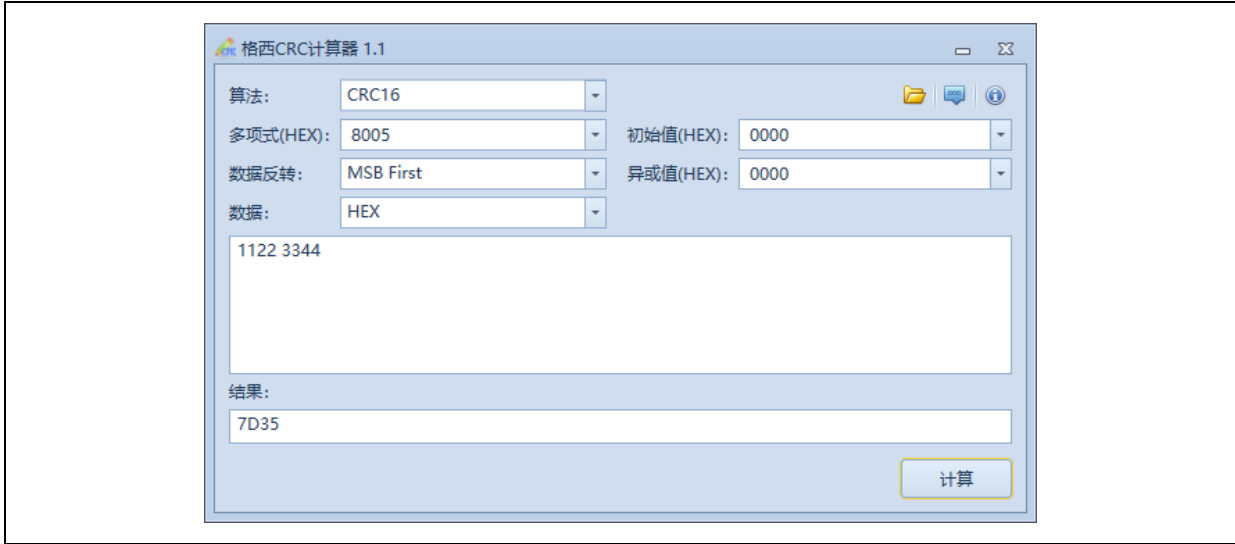


### 5.4.7 宏定义选择 CRC16\_TEST\_2

表 13. 参数配置

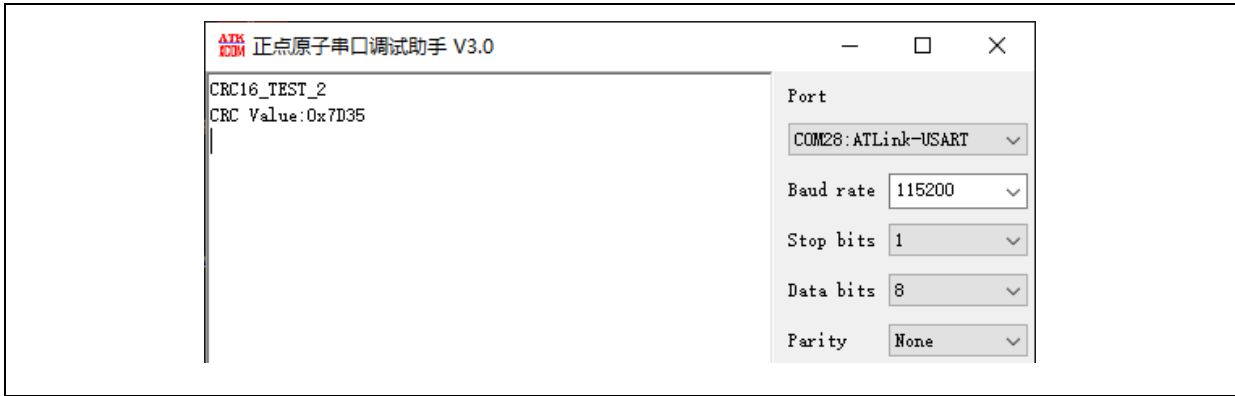
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC16	0x8005	MSB First	0x0000	0x0000	0x1122 0x3344	0x7D35

图 18.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 19.串口打印结果

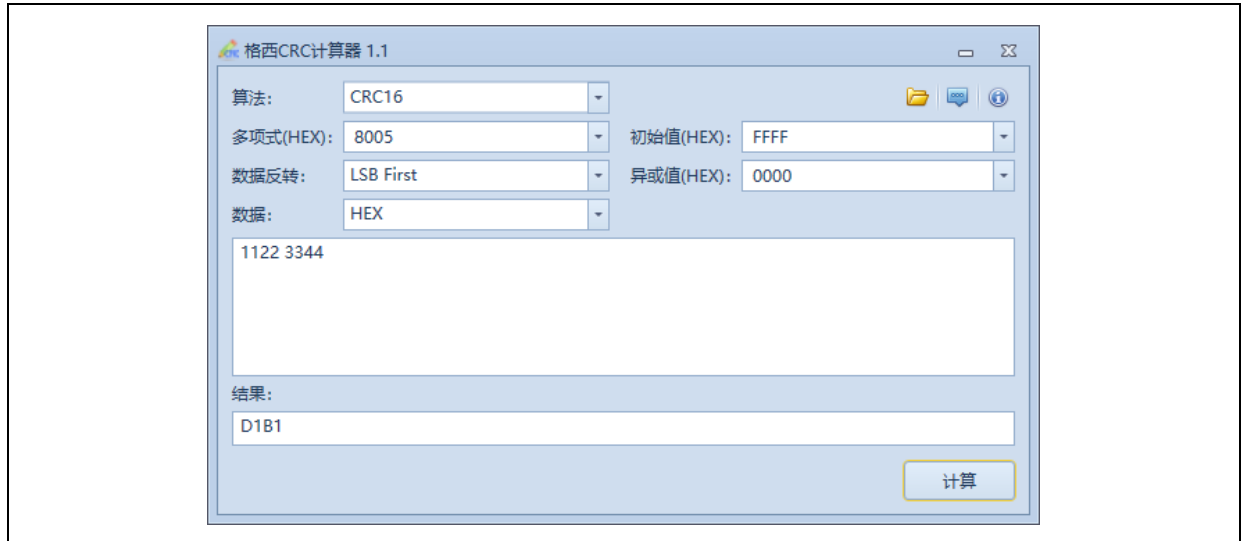


### 5.4.8 宏定义选择 CRC16\_TEST\_3

表 14. 参数配置

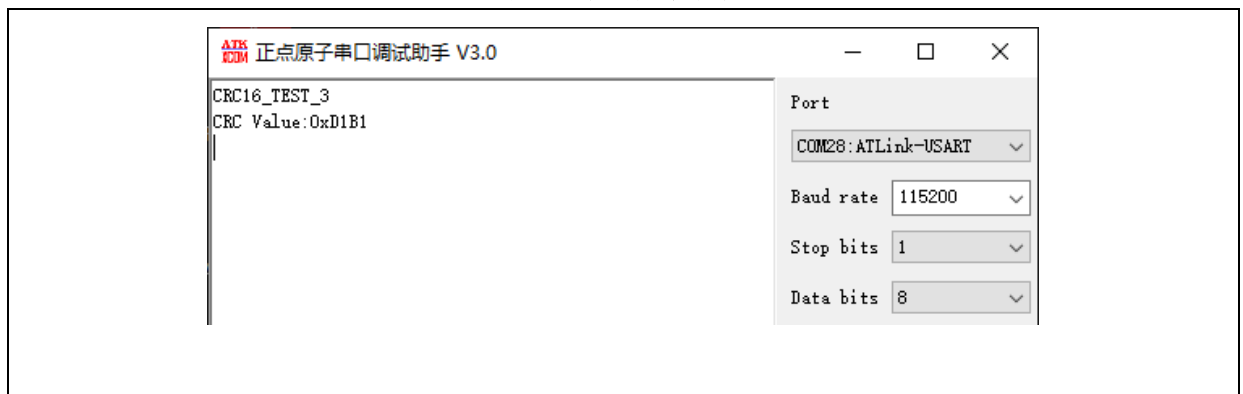
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC16	0x8005	LSB First	0xFFFF	0x0000	0x1122 0x3344	0xD1B1

图 20.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 21.串口打印结果

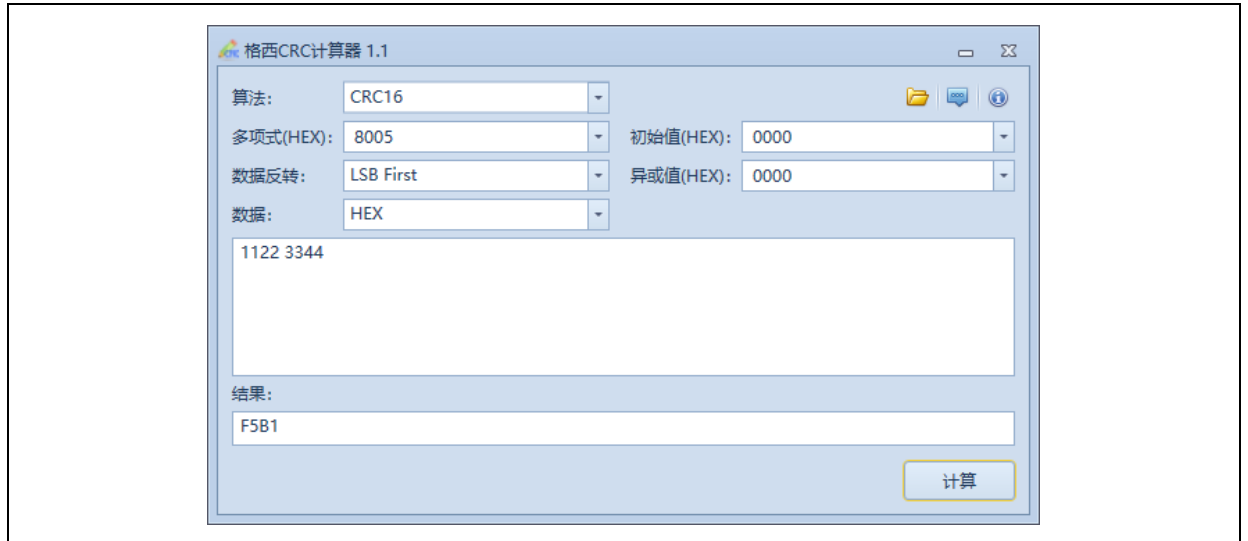


### 5.4.9 宏定义选择 CRC16\_TEST\_4

表 15. 参数配置

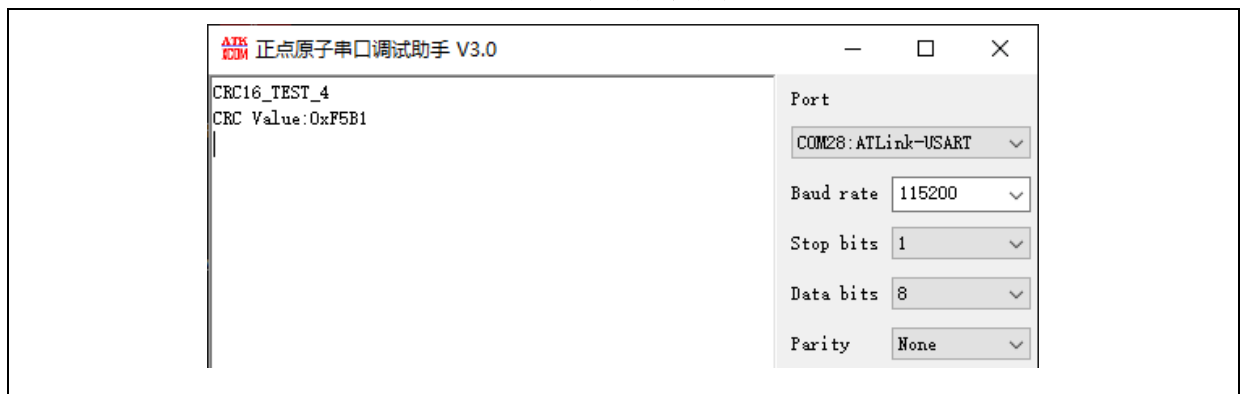
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC16	0x8005	LSB First	0x0000	0x0000	0x1122 0x3344	0xF5B1

图 22.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 23.串口打印结果



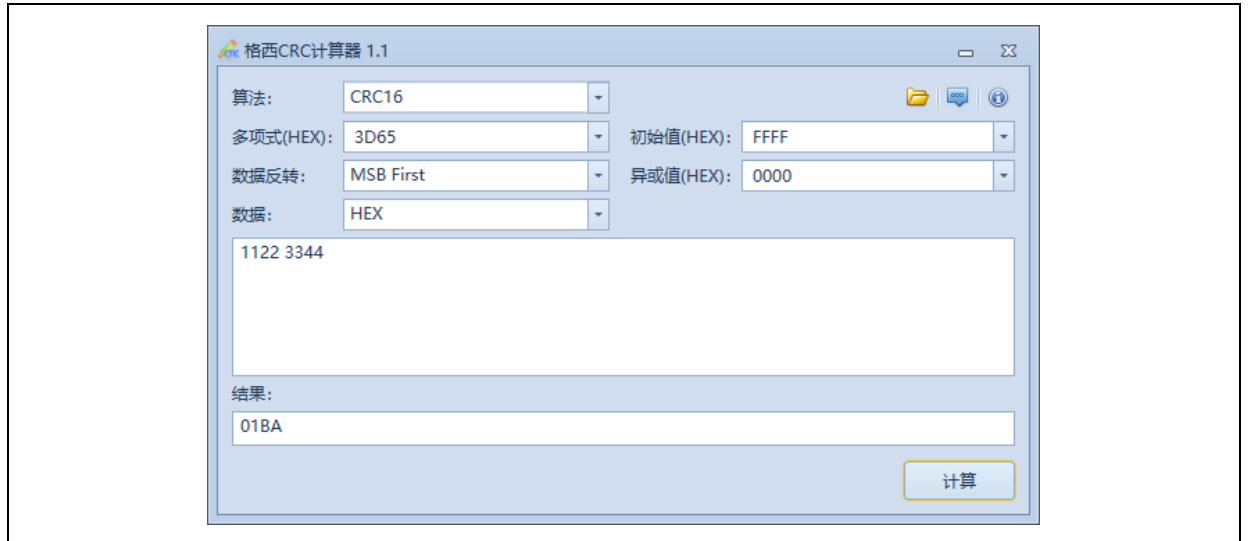


### 5.4.10 宏定义选择 CRC16\_TEST\_5

表 16. 参数配置

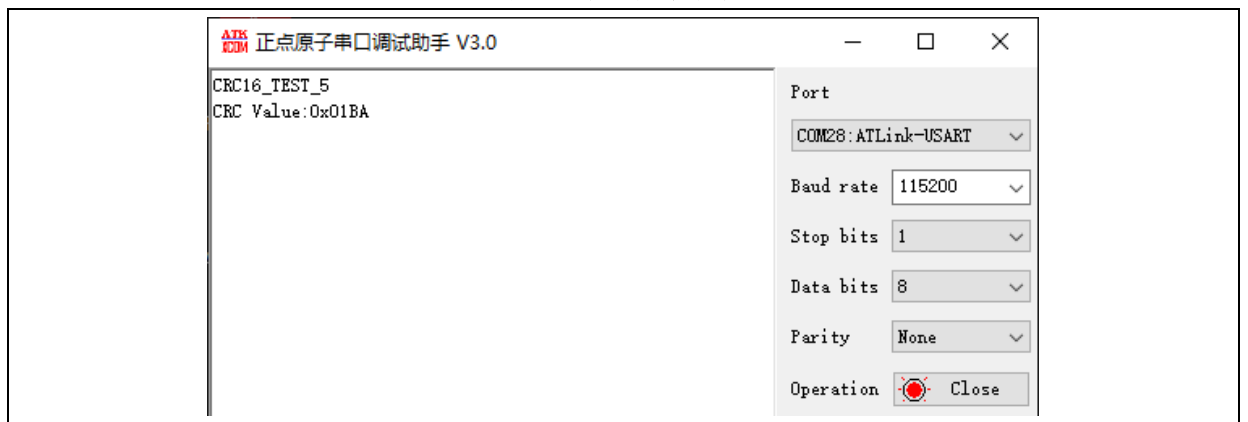
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC16	0x3D65	MSB First	0xFFFF	0x0000	0x1122 0x3344	0x01BA

图 24.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 25.串口打印结果

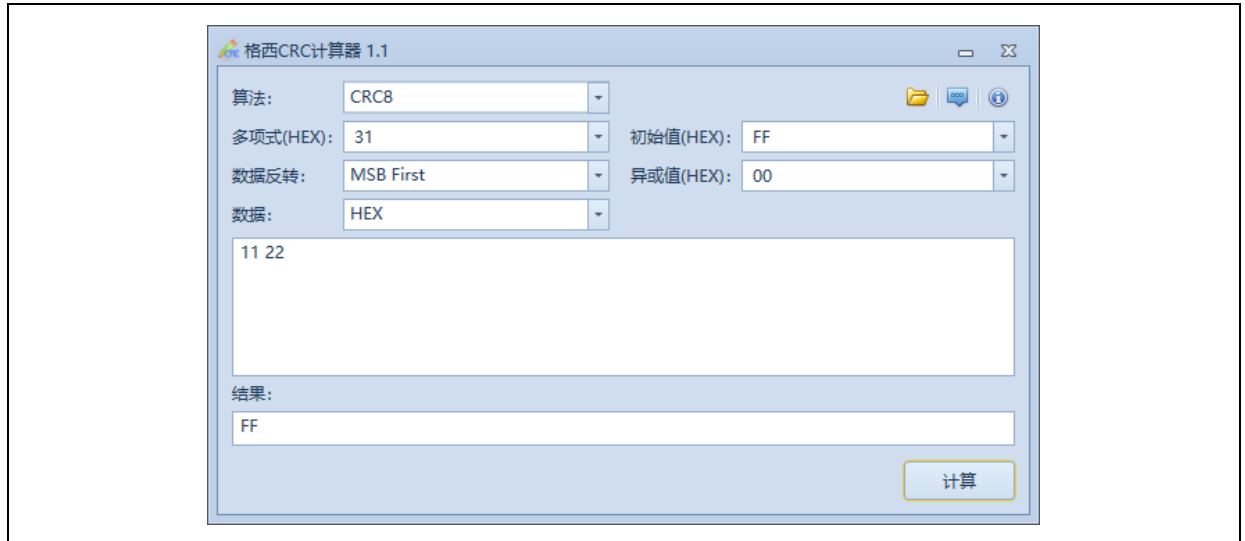


### 5.4.11 宏定义选择 CRC8\_TEST\_1

表 17. 参数配置

算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC8	0x31	MSB First	0xFF	0x00	0x11 0x22	0xFF

图 26.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 27.串口打印结果



### 5.4.12 宏定义选择 CRC8\_TEST\_2

表 18. 参数配置

算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC8	0x31	MSB First	0x00	0x00	0x11 0x22	0x7E

图 28.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 29.串口打印结果

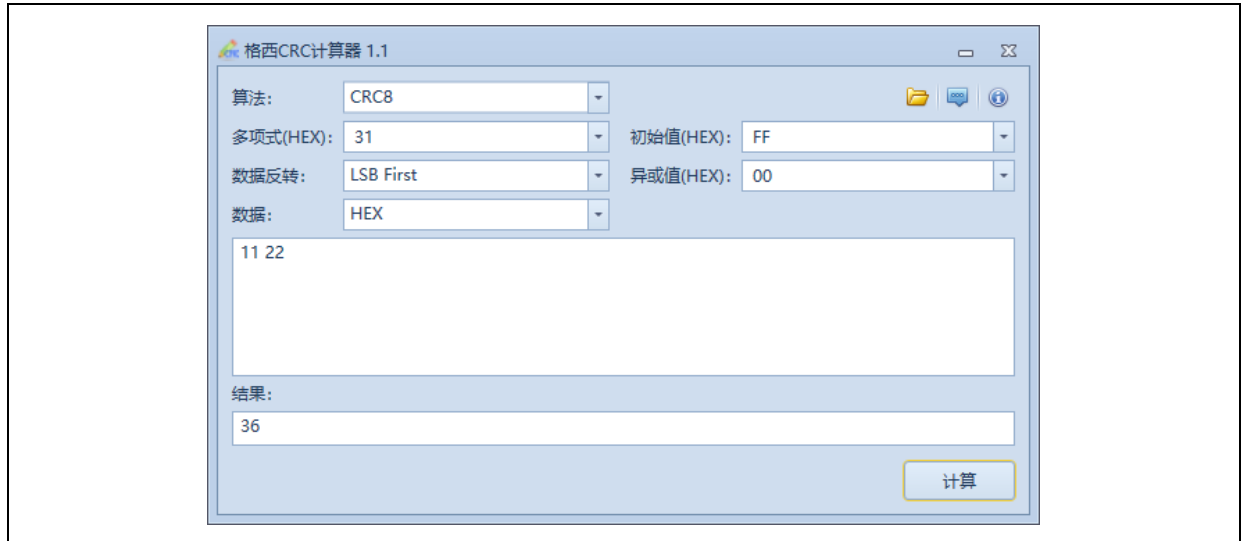


### 5.4.13 宏定义选择 CRC8\_TEST\_3

表 19. 参数配置

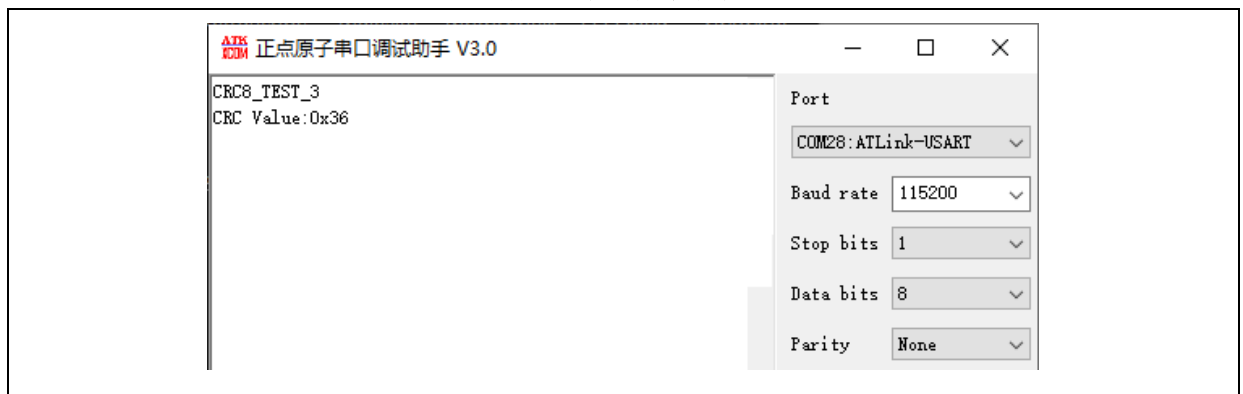
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC8	0x31	LSB First	0xFF	0x00	0x11 0x22	0x36

图 30.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 31.串口打印结果

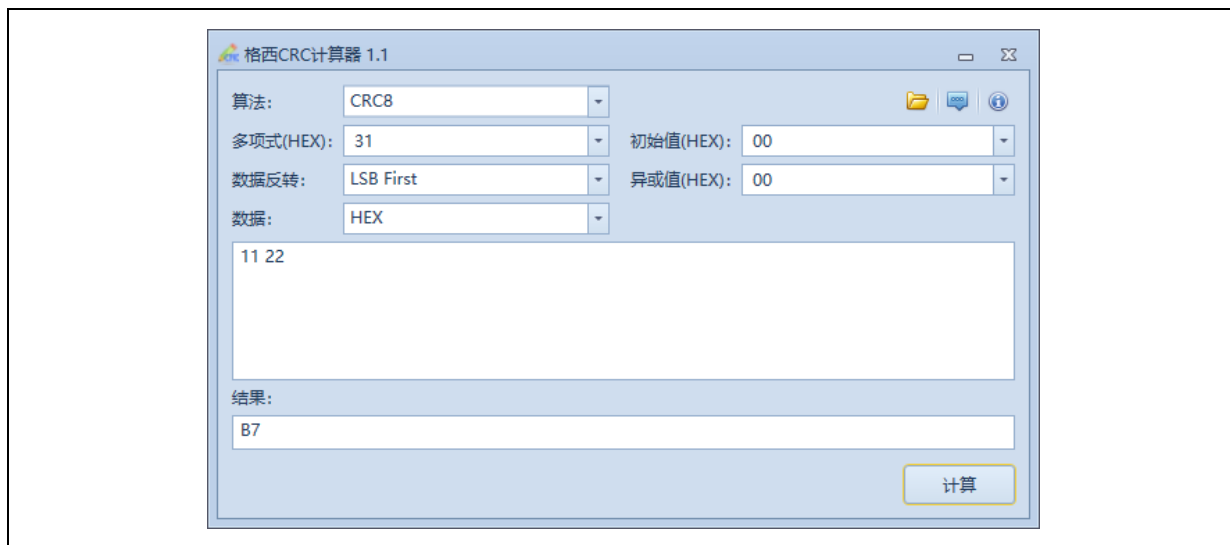


### 5.4.14 宏定义选择 CRC8\_TEST\_4

表 20. 参数配置

算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC8	0x31	LSB First	0x00	0x00	0x11 0x22	0xB7

图 32.上位机配置



注: CRC计算器英文界面请参考[5.4.1](#)章节。

图 33.串口打印结果



### 5.4.15 宏定义选择 CRC8\_TEST\_5

表 21. 参数配置

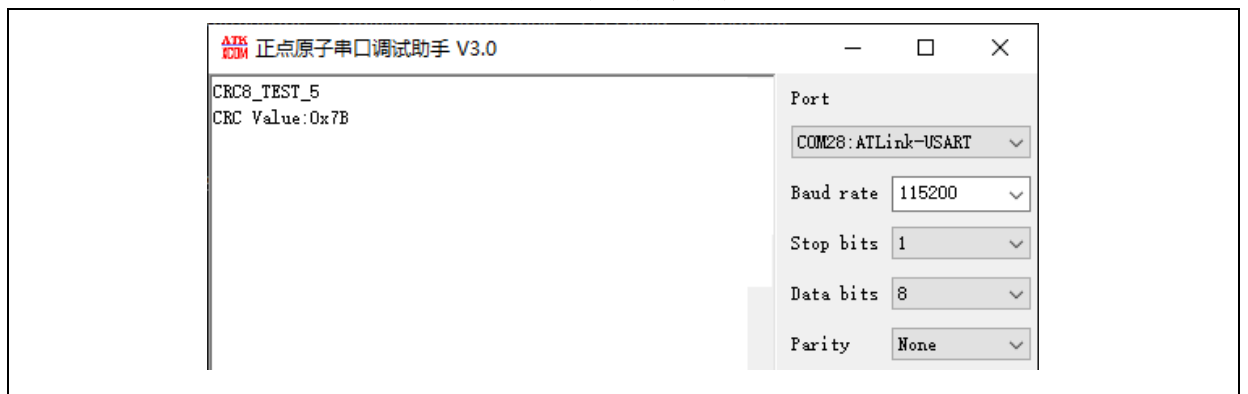
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC8	0x07	MSB First	0xFF	0x00	0x11 0x22	0x7B

图 34.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 35.串口打印结果



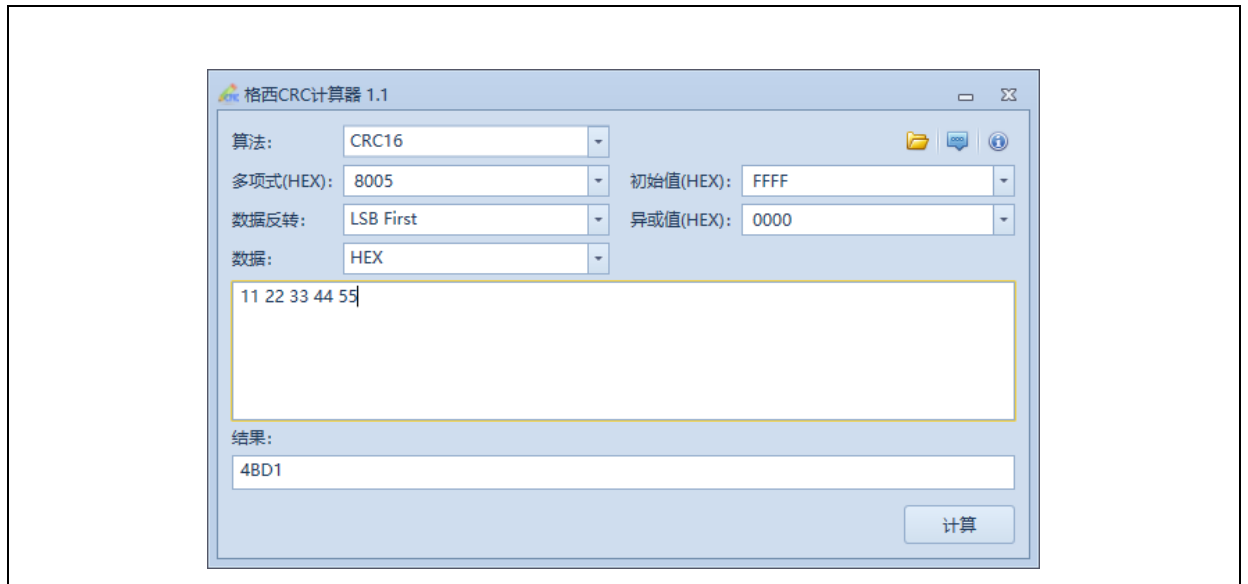
### 5.4.16 定义选择 CRC16\_TEST\_8BIT\_DATA(MODBUS)

使用 CRC16 计算，传入数据为 8bit 数据

表 22. 参数配置

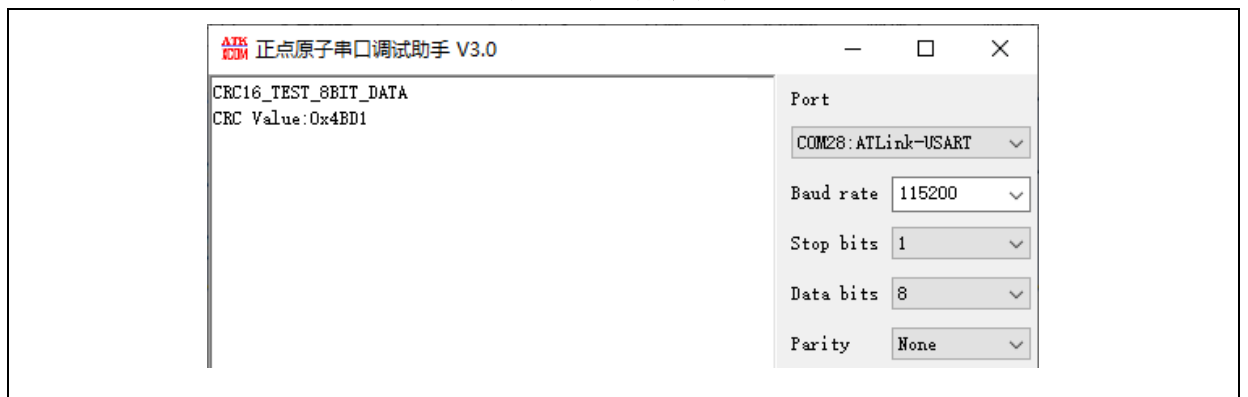
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC16	0x8005	LSB First	0xFFFF	0x0000	0x11 0x22 0x33 0x44 0x55	0x4BD1

图 36.上位机配置



注：CRC 计算器英文界面请参考[5.4.1](#) 章节。

图 37.串口打印结果



### 5.4.17 宏定义选择 CRC32\_TEST\_8BIT\_DATA

使用 CRC32 计算，传入数据为 8bit 数据

表 23. 参数配置

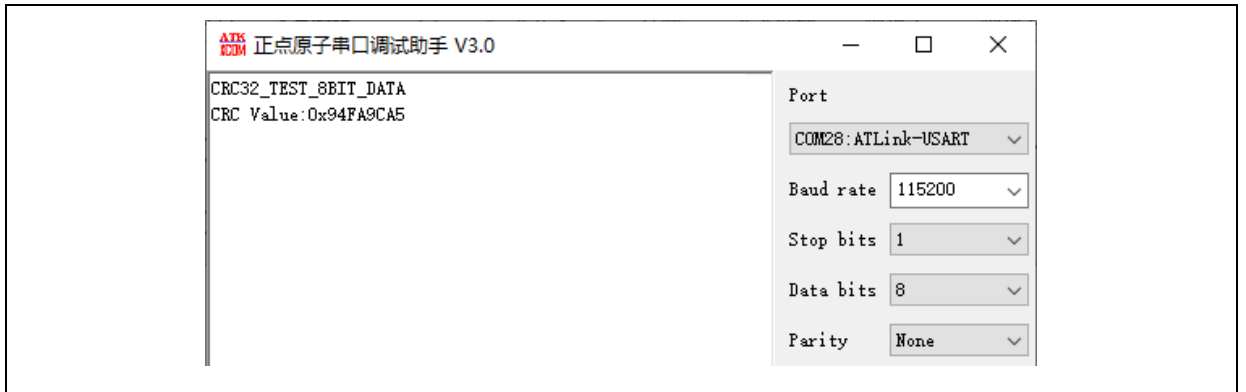
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC32	0x4C11DB7	MSB First	0xFFFFFFFF	0x00000000	0x11 0x22 0x33 0x44 0x55	0x94FA9CA5

图 38.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 39.串口打印结果





### 5.4.18 宏定义选择 CRC32\_TEST\_16BIT\_DATA

使用 CRC32 计算，传入数据为 16bit 数据

表 24. 参数配置

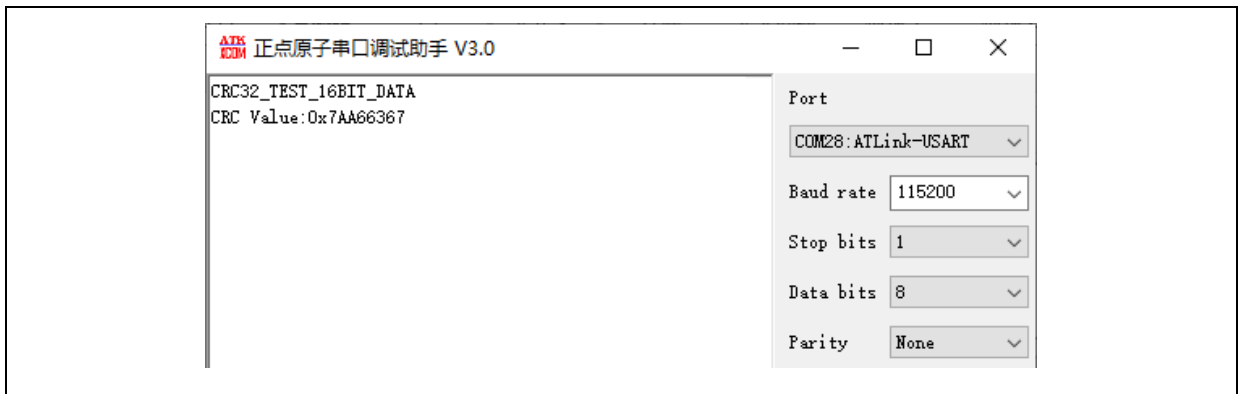
算法	多项式	数据反转	初始值	异或值	测试数据	测试结果
CRC32	0x4C11DB7	MSB First	0xFFFFFFFF	0x00000000	0x1122 0x3344 0x5566	0x7AA66367

图 40.上位机配置



注：CRC计算器英文界面请参考[5.4.1](#)章节。

图 41.串口打印结果



## 6 文档版本历史

表 25. 文档版本历史

日期	版本	变更
2022.01.21	2.0.0	最初版本
2023.11.02	2.0.1	新增CRC多项式宽度和初始值设置描述
2024.01.08	2.0.2	新增CRC计算例程

**重要通知 - 请仔细阅读**

买方自行负责对本文所述雅特力产品和服务的选择和使用，雅特力概不承担与选择或使用本文所述雅特力产品和服务相关的任何责任。

无论之前是否有任何形式的表示，本文档不以任何方式对任何知识产权进行任何明示或默示的授权或许可。如果本文档任何部分涉及任何第三方产品或服务，不应被视为雅特力授权使用此类第三方产品或服务，或许可其中的任何知识产权，或者被视为涉及以任何方式使用任何此类第三方产品或服务或其中任何知识产权的保证。

除非在雅特力的销售条款中另有说明，否则，雅特力对雅特力产品的使用和/或销售不做任何明示或默示的保证，包括但不限于有关适销性、适合特定用途（及其依据任何司法管辖区的法律的对应情况），或侵犯任何专利、版权或其他知识产权的默示保证。

雅特力产品并非设计或专门用于下列用途的产品：（A）对安全性有特别要求的应用，例如：生命支持、主动植入设备或对产品功能安全有要求的系统；（B）航空应用；（C）航天应用或航天环境；（D）武器，且/或（E）其他可能导致人身伤害、死亡及财产损失的应用。如果采购商擅自将其用于前述应用，即使采购商向雅特力发出了书面通知，风险及法律责任仍将由采购商单独承担，且采购商应独立负责在前述应用中满足所有法律和法规要求。

经销的雅特力产品如有不同于本文档中提出的声明和/或技术特点的规定，将立即导致雅特力针对本文所述雅特力产品或服务授予的任何保证失效，并且不应以任何形式造成或扩大雅特力的任何责任。

© 2024 雅特力科技 保留所有权利