# P1060966 HW&APP FPGA Register Map

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **作者** | **日期** | **描述** |
| 0.1 | XYL | 2025/8/1 | 基于原ATP的初始版本 |
|  |  |  |  |

## **概要**

Stand HW FPGA和APP FPGA是完全独立的，各自维护一张寄存器表。

## **HW-FPGA寄存器列表**

|  |  |  |
| --- | --- | --- |
| **地址** | **地址** | **说明** |
| SCRATCHPAD | 0x00000000 | FPGA型号，版本，发布日期等；读写测试寄存器 |
| CLOCK GEN | 0x00010000 | 配置内部时钟，默认不用修改 |
| OSC COUNTER1 | 0x00020000 | 100M时钟频率测试 |
| OSC COUNTER2 | 0x00021000 | 50M时钟频率测试 |
| GPIO | 0x00030000 | GPIO输入/输出，包括通用GPIO和Switches等 |
| FPGA IF | 0x00040000 | HW和APP FPGA之间的接口信号互联测试 |
| EEPROM | 0x00050000 | EEPROM读写测试 |
| PHY | 0x00060000 | PHY通信模块测试 |

### **SCRATCHPAD子模块**

* + 1. **寄存器列表**

这个模块用于测试上位机和FPGA之间的通信，可访问的寄存器列表如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | **偏移地址** | **读写** | **位宽** | **说明** | **复位值** |
| **Version** | **0** | **R** | **32位** | **FW Version** | **0x0000\_0001** |
| **ID** | **1** | **R** | **32位** | **FPGA ID** | **0x0000\_0050** |
| **Date** | **2** | **R** | **32位** | **Build Date** | **0x2025\_0714** |
| **dev\_sp1** | **3** | **R/W** | **32位** | **Scratch Pad 1** | **0x12345678** |
| **dev\_sp2** | **4** | **R/W** | **32位** | **Scratch Pad 2** | **0x9abcbeef** |

* + 1. **说明**
* 该模块定义了一些只读寄存器包含FPGA型号，版本，发布日期等
* 该模块实现了两个32位通用寄存器（Scratch Pad），支持独立读写。
* 复位时，dev\_sp1 和 dev\_sp2 分别初始化为 0x12345678 和 0x9abcbeef。
  + 1. **示例**

Step-1. OPB Read dev\_sp1 @0x0001\_0000

Step-2. OPB Read dev\_sp2 @0x0002\_0000

Step-3. OPB Write dev\_sp1 @0x0001\_0000

Step-4. OPB Write dev\_sp2 @0x0002\_0000

Step-5. OPB Read dev\_sp1 @0x0001\_0000

Step-6. OPB Read dev\_sp2 @0x0002\_0000

### **CLOCK GEN子模块**

**注意：该模块用于配置内部时钟参数，无需修改！**

* + 1. 寄存器列表如下：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **地址宏定义** | **偏移地址** | **读写** | **位宽** | **说明** | **复位值** |
| PULSE\_5US\_ADDR | 1 | R/W | 16位 | 5us脉冲分频系数 | 250 |
| PULSE\_50US\_ADDR | 2 | R/W | 16位 | 50us脉冲分频系数 | 2500 |
| PULSE\_500US\_ADDR | 3 | R/W | 16位 | 500us脉冲分频系数 | 25000 |
| PULSE\_100US\_ADDR | 4 | R/W | 16位 | 100us脉冲分频系数 | 5000 |
| CLK\_2MHZ\_ADDR | 5 | R/W | 16位 | 2MHz时钟分频系数 | 25 |

### **OSC COUNTER1子模块**

该模块用于测试U61 (100MHz Oscollator)。

注意：STN ATP没有这项测试要求。

* + 1. **寄存器列表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **偏移地址** | **读写** | **位宽** | **说明** |
| CNTRL Register | 0x0 | R/W | 3 bits | 控制寄存器  control[0]=Start  control[1]=Reset  control[2]=In Progress |
| Counter Register | 0x1 | R | 16 bits | 计数寄存器  只读，当前计数值 |
| Scratch Pad | 0x2 | R/W | 32 bits | 通用寄存器，可读写 |

* + 1. **详细说明**
* 控制寄存器（CNTRLR, 0x0）
  + [0] Start Measurement（写1启动一次测量）
  + [1] Reset Counter（写1复位计数器）
  + [2] Measurement In Progress（只读，测量进行中）
* 计数寄存器（COUNTR, 0x1）
  + [15:0] 计数值（只读）
* Scratch Pad寄存器（SPR, 0x2）
  + [31:0] 通用数据寄存器（可读写）
    1. **示例**

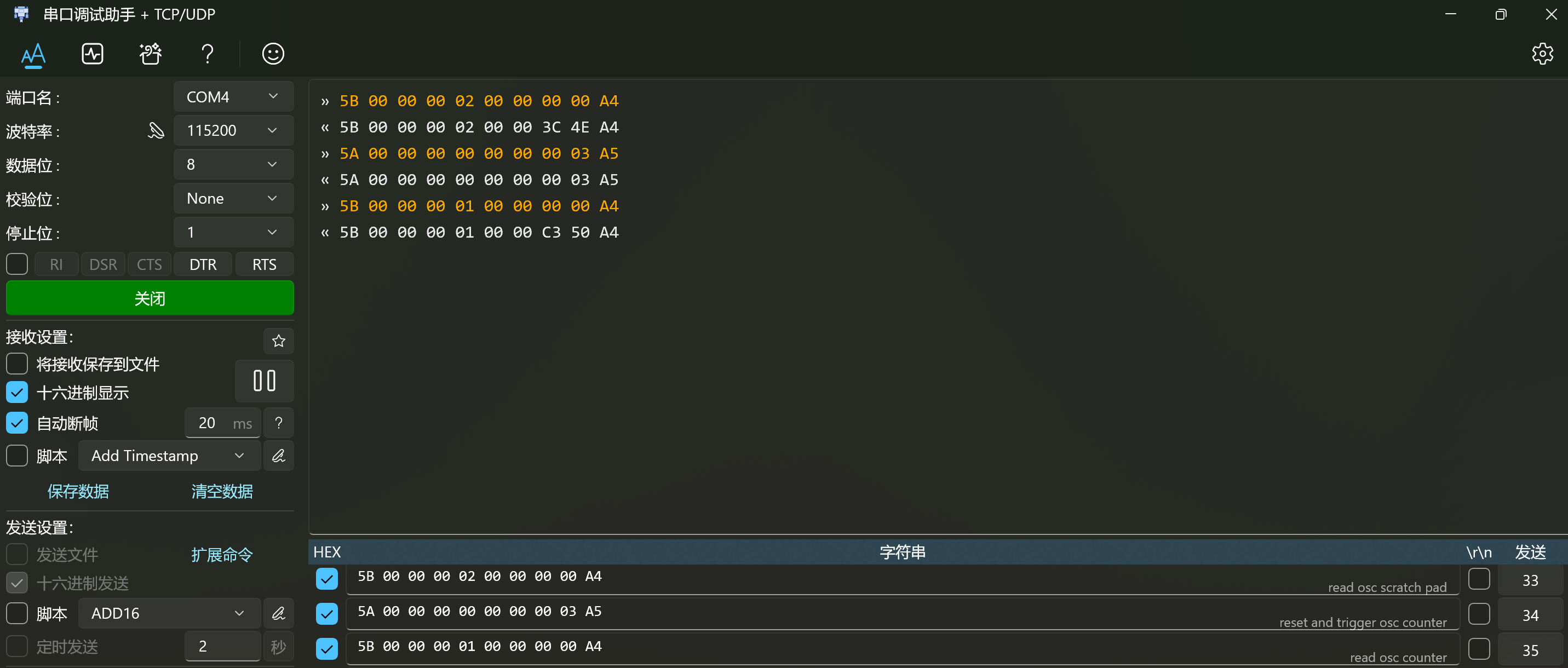
Step-1. OPB Read Scratch Pad Register@(0x00000002), return 32’h00003c4e

Step-2. OPB Write 32’h3 to CNTRL Register@(0x0000000)

Step-3. OPB Read Counter Register @(0x00000001), check return value

Counter Value = 100M/2k = 50,000 (0xC350) (+/ 50）

Freq Value = Counter\_Value \* 2k +/ 0.1M



### **OSC COUNTER2子模块**

该模块用于测试U48 (50MHz Oscillator)。

注意：STN ATP没有这项测试要求。

* + 1. **寄存器列表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **偏移地址** | **读写** | **位宽** | **说明** |
| CNTRL Register | 0x0 | R/W | 3 bits | 控制寄存器  control[0]=Start  control[1]=Reset  control[2]=In Progress |
| Counter Register | 0x1 | R | 16 bits | 计数寄存器  只读，当前计数值 |
| Scratch Pad | 0x2 | R/W | 32 bits | 通用寄存器，可读写 |

* + 1. **详细说明**
* 控制寄存器（CNTRLR, 0x0）
  + [0] Start Measurement（写1启动一次测量）
  + [1] Reset Counter（写1复位计数器）
  + [2] Measurement In Progress（只读，测量进行中）
* 计数寄存器（COUNTR, 0x1）
  + [15:0] 计数值（只读）
* Scratch Pad寄存器（SPR, 0x2）
  + [31:0] 通用数据寄存器（可读写）
    1. **示例**

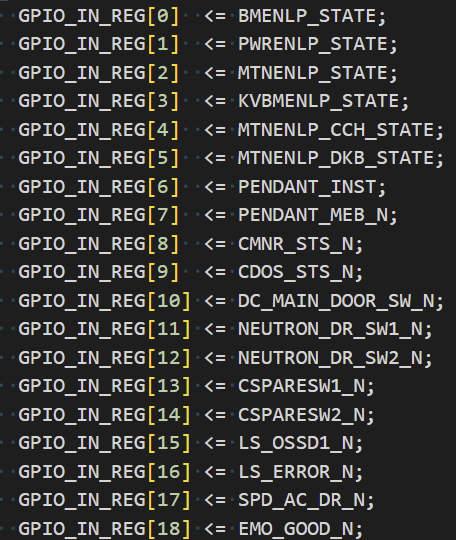
### **GPIO子模块**

该模块用于读写FPGA GPIO。

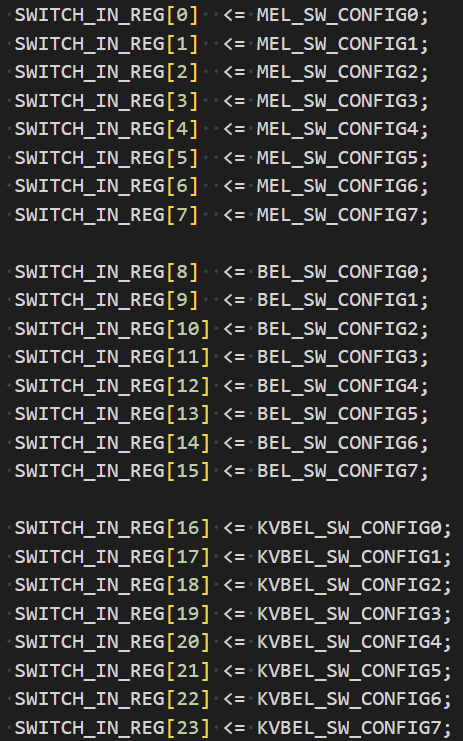
* + 1. **寄存器列表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 地址（宏定义） | 偏移地址 | 读写 | 位宽 | 说明 | 复位值 |
| GPIO\_IN\_ADDR | 0x0 | R | 32位 | 通用GPIO输入 | NA |
| SWITCH\_IN\_ADDR | 0x1 | R | 32位 | 编码开关输入 | NA |
| GPIO\_OUT\_ADDR | 0x2 | R/W | 32位 | 通过GPIO输出 | 32’h0000\_0000 |
| TP\_OUT\_ADDR | 0x3 | R/W | 32位 | 测试点输出 | 32’h0000\_0000 |

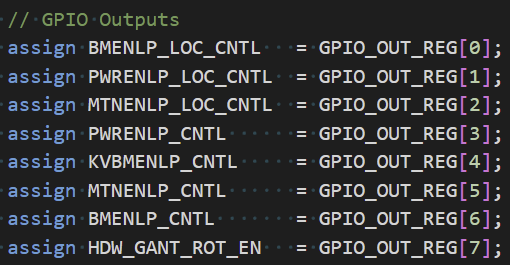
* + 1. **GPIO IN**



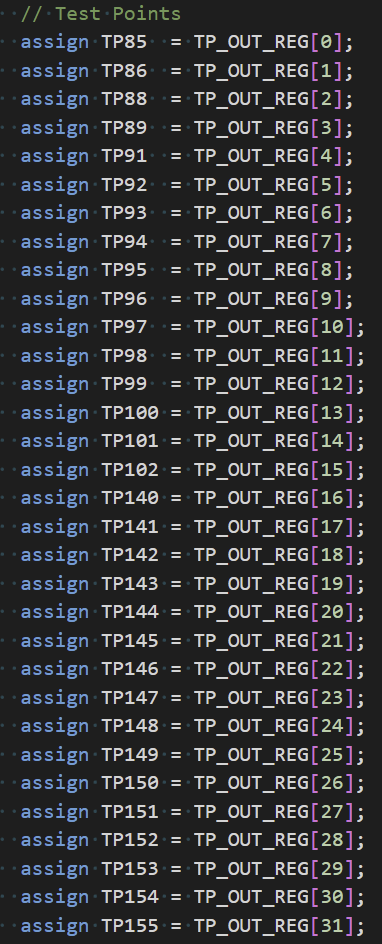
* + 1. **SWITCH IN**



* + 1. **GPIO OUT**



* + 1. **TESTPOINTS OUT**



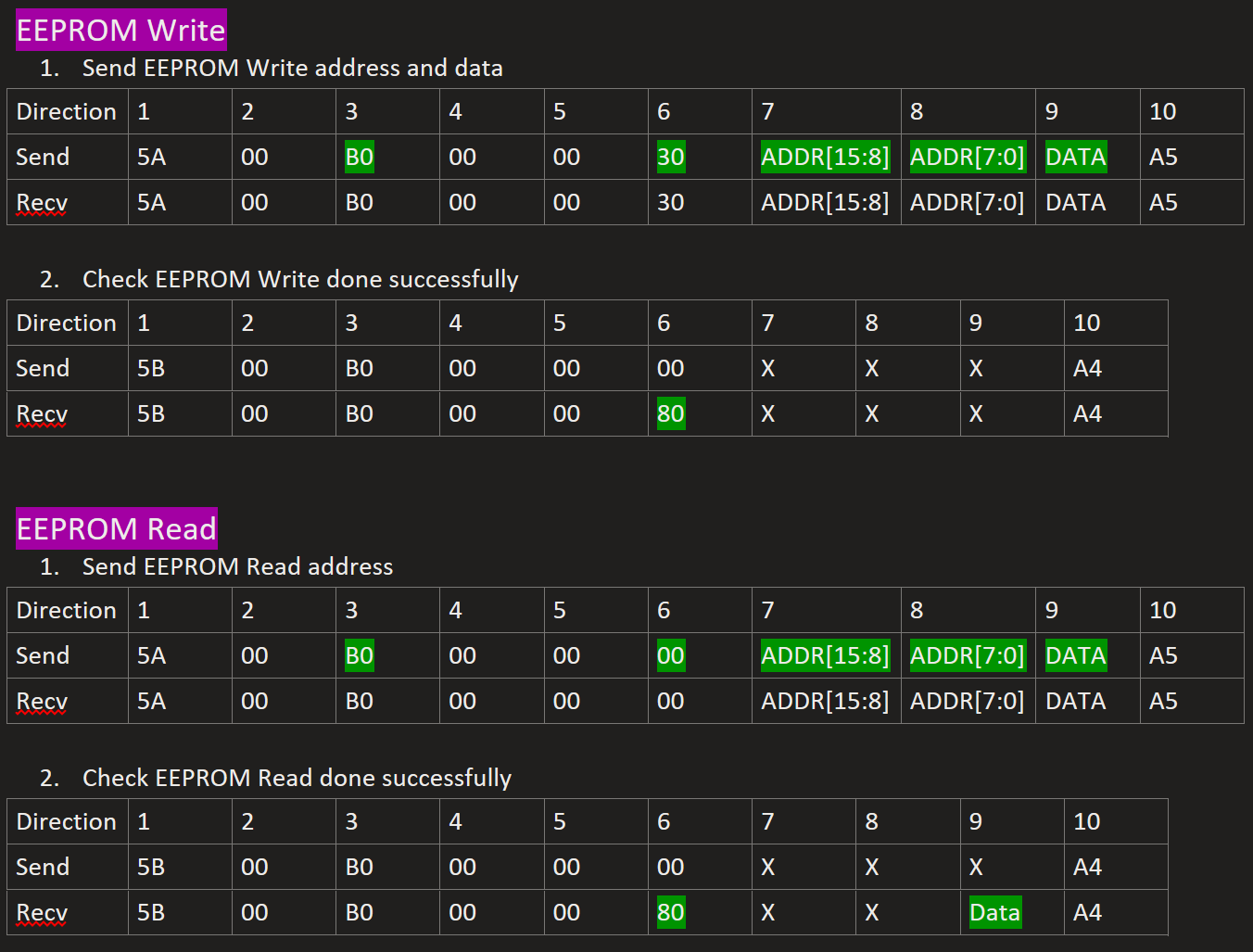
### **FPGA IF子模块 (TODO)**

### **EEPROM子模块**

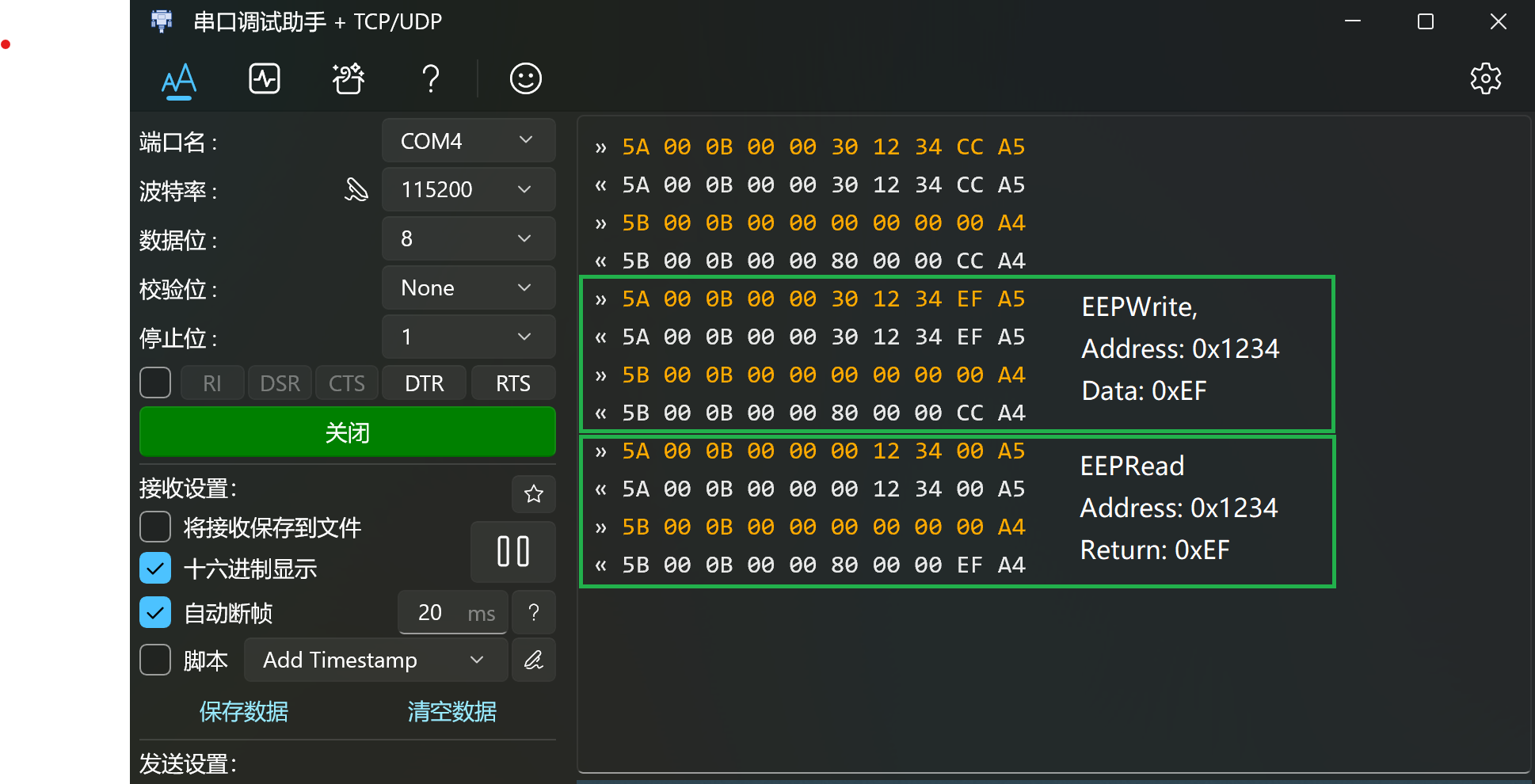
该模块用于EEPROM(U46)的测试。

注意：OPB\_ADDR应该修改为32’H0005\_0000

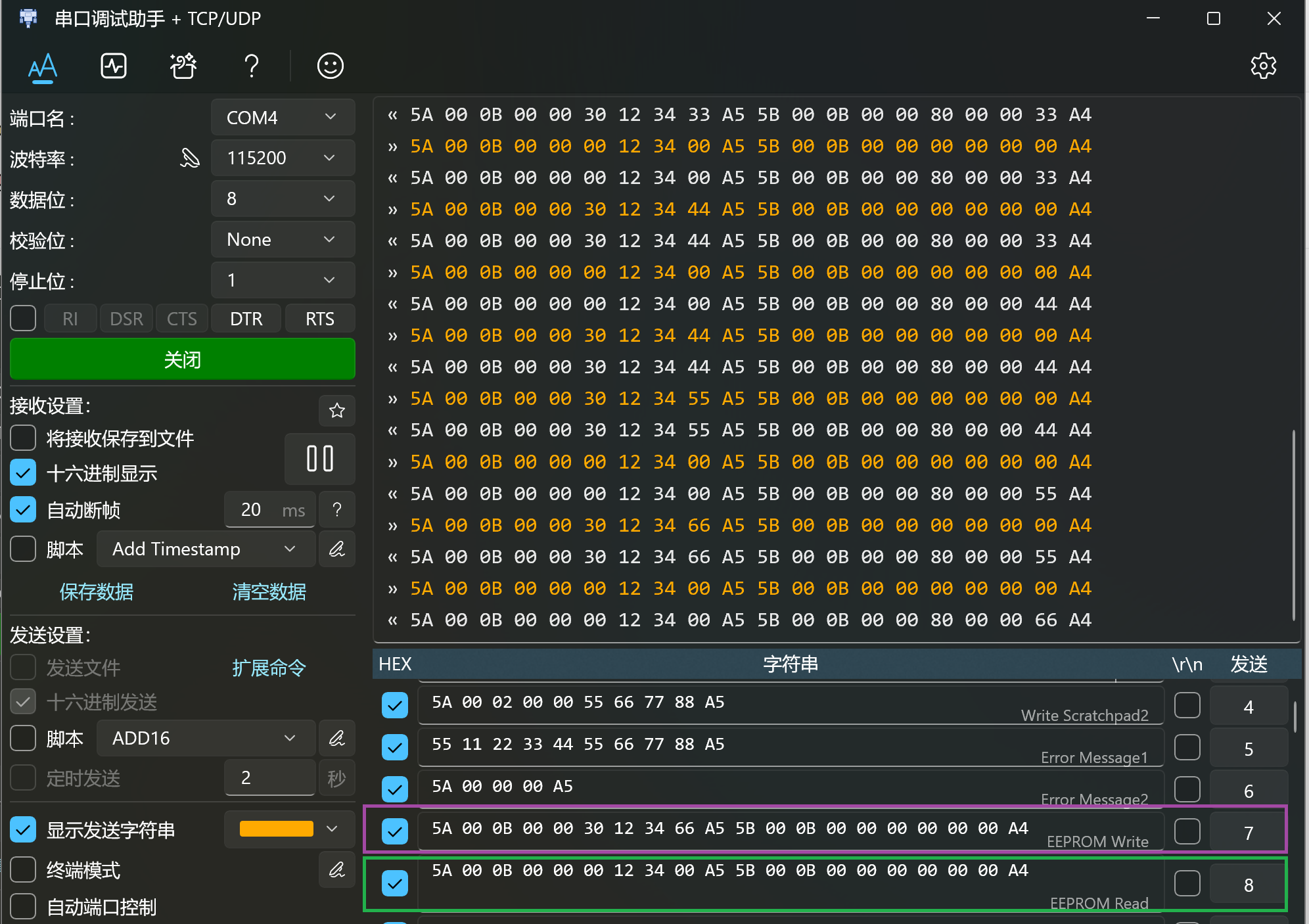




**Example**

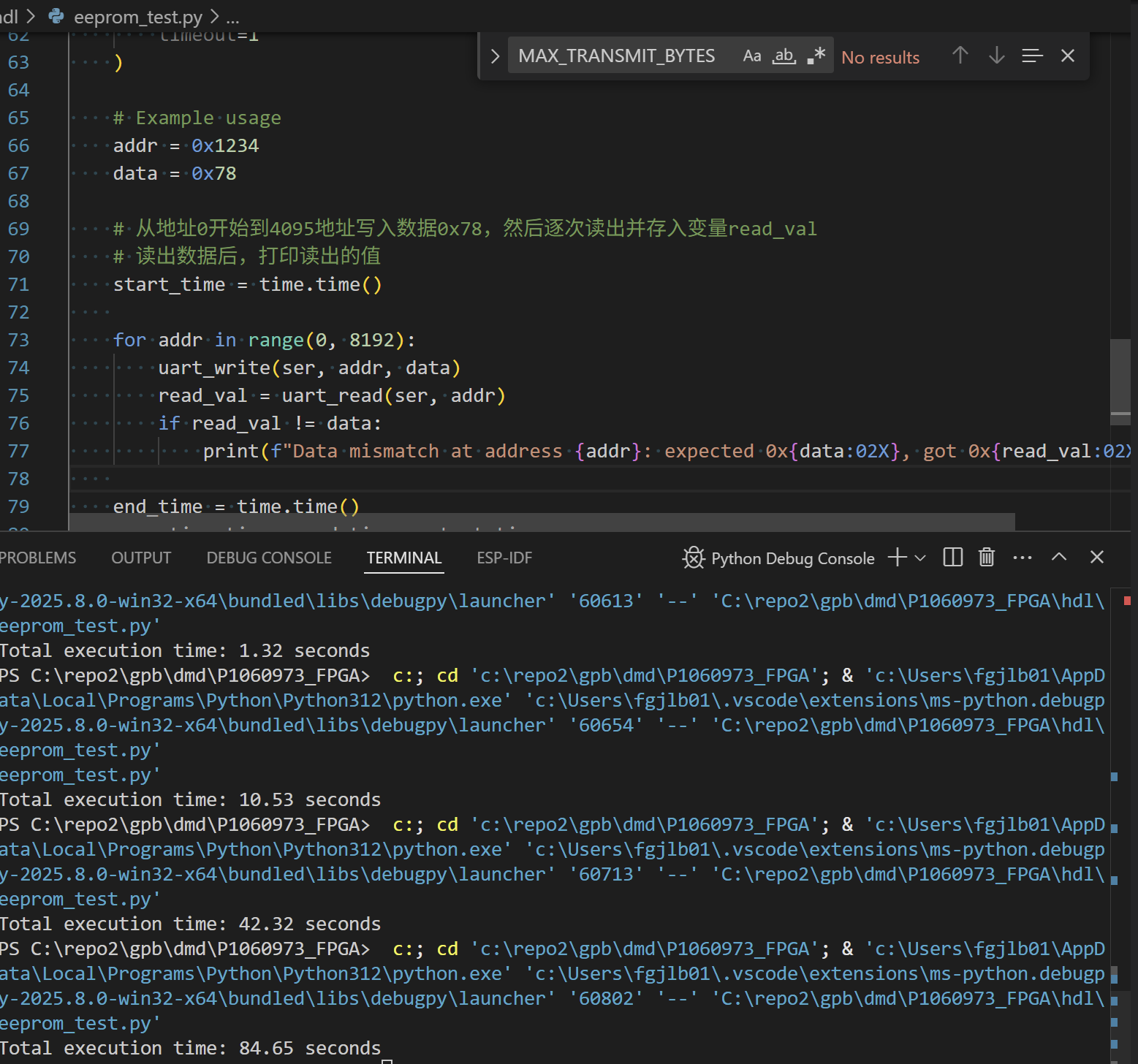


为了节省测试时间，两条命令可以组装成一条命令，如下：



写了一段python测试EEPROM的读写时间，见eeprom\_test.py。先写入数据到一段EEPROM，然后读出并进行校验。注意：EEPROM 25XX640A的最大存储空间是8192 bytes. 测试结果如下：

|  |  |
| --- | --- |
| 数据长度 | 读写时间(second) |
| 128 | 1.32 |
| 1024 | 10.53 |
| 4096 | 42.32 |
| 8192 | 84.65 |



### **PHY子模块 (TODO)**

## **APP-FPGA寄存器列表**

|  |  |  |
| --- | --- | --- |
| **地址** | **地址** | **说明** |
| SCRATCHPAD | 0x00000000 | FPGA型号，版本，发布日期等；读写测试寄存器 |
| CLOCK GEN | 0x00010000 | 配置内部时钟，默认不用修改 |
| OSC COUNTER | 0x00020000 | 100M/50M时钟频率测试 |
| GPIO | 0x00030000 | GPIO输入/输出，包括通用GPIO和Switches等 |
| FPGA IF | 0x00040000 | HW和APP FPGA之间的接口信号互联测试 |
| ADC | 0x00050000 | ADC ADS8864I相关测试 |
| DAC | 0x00060000 | DAC60504相关测试 |
| RS422 | 0x00070000 | RS422/485通信测试 |
| CAN | 0x00080000 | CAN BUS通信测试 |

### **SCRATCHPAD子模块**

这个模块用于测试上位机和FPGA之间的通信。

* + 1. **寄存器列表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | **偏移地址** | **读写** | **位宽** | **说明** | **复位值** |
| **Version** | **0** | **R** | **32位** | **FW Version** | **0x0000\_0001** |
| **ID** | **1** | **R** | **32位** | **FPGA ID** | **0x0000\_0100** |
| **Date** | **2** | **R** | **32位** | **Build Date** | **0x2025\_0714** |
| **dev\_sp1** | **3** | **R/W** | **32位** | **Scratch Pad 1** | **0x12345678** |
| **dev\_sp2** | **4** | **R/W** | **32位** | **Scratch Pad 2** | **0x9abcbeef** |

* + 1. **说明**
* 该模块定义了一些只读寄存器包含FPGA型号，版本，发布日期等
* 该模块实现了两个32位通用寄存器（Scratch Pad），支持独立读写。
* 复位时，dev\_sp1 和 dev\_sp2 分别初始化为 0x12345678 和 0x9abcbeef。
  + 1. **示例**

Step-1. OPB Read dev\_sp1 @0x0001\_0000

Step-2. OPB Read dev\_sp2 @0x0002\_0000

Step-3. OPB Write dev\_sp1 @0x0001\_0000

Step-4. OPB Write dev\_sp2 @0x0002\_0000

Step-5. OPB Read dev\_sp1 @0x0001\_0000

Step-6. OPB Read dev\_sp2 @0x0002\_0000

### **CLOCK GEN子模块**

**注意：该模块用于配置内部时钟参数，无需修改！**

* + 1. **寄存器列表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **地址宏定义** | **偏移地址** | **读写** | **位宽** | **说明** | **复位值** |
| PULSE\_5US\_ADDR | 1 | R/W | 16位 | 5us脉冲分频系数 | 250 |
| PULSE\_50US\_ADDR | 2 | R/W | 16位 | 50us脉冲分频系数 | 2500 |
| PULSE\_500US\_ADDR | 3 | R/W | 16位 | 500us脉冲分频系数 | 25000 |
| PULSE\_100US\_ADDR | 4 | R/W | 16位 | 100us脉冲分频系数 | 5000 |
| CLK\_2MHZ\_ADDR | 5 | R/W | 16位 | 2MHz时钟分频系数 | 25 |

### **OSC COUNTER子模块**

该模块用于测试U58 (100MHz Oscollator)。

注意：STN ATP没有这项测试要求。

* + 1. **寄存器列表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **名称** | **偏移地址** | **读写** | **位宽** | **说明** |
| CNTRL Register | 0x0 | R/W | 3 bits | 控制寄存器  control[0]=Start  control[1]=Reset  control[2]=In Progress |
| Counter Register | 0x1 | R | 16 bits | 计数寄存器  只读，当前计数值 |
| Scratch Pad | 0x2 | R/W | 32 bits | 通用寄存器，可读写 |

* + 1. **详细说明**
* 控制寄存器（CNTRLR, 0x0）
  + [0] Start Measurement（写1启动一次测量）
  + [1] Reset Counter（写1复位计数器）
  + [2] Measurement In Progress（只读，测量进行中）
* 计数寄存器（COUNTR, 0x1）
  + [15:0] 计数值（只读）
* Scratch Pad寄存器（SPR, 0x2）
  + [31:0] 通用数据寄存器（可读写）
    1. **示例**

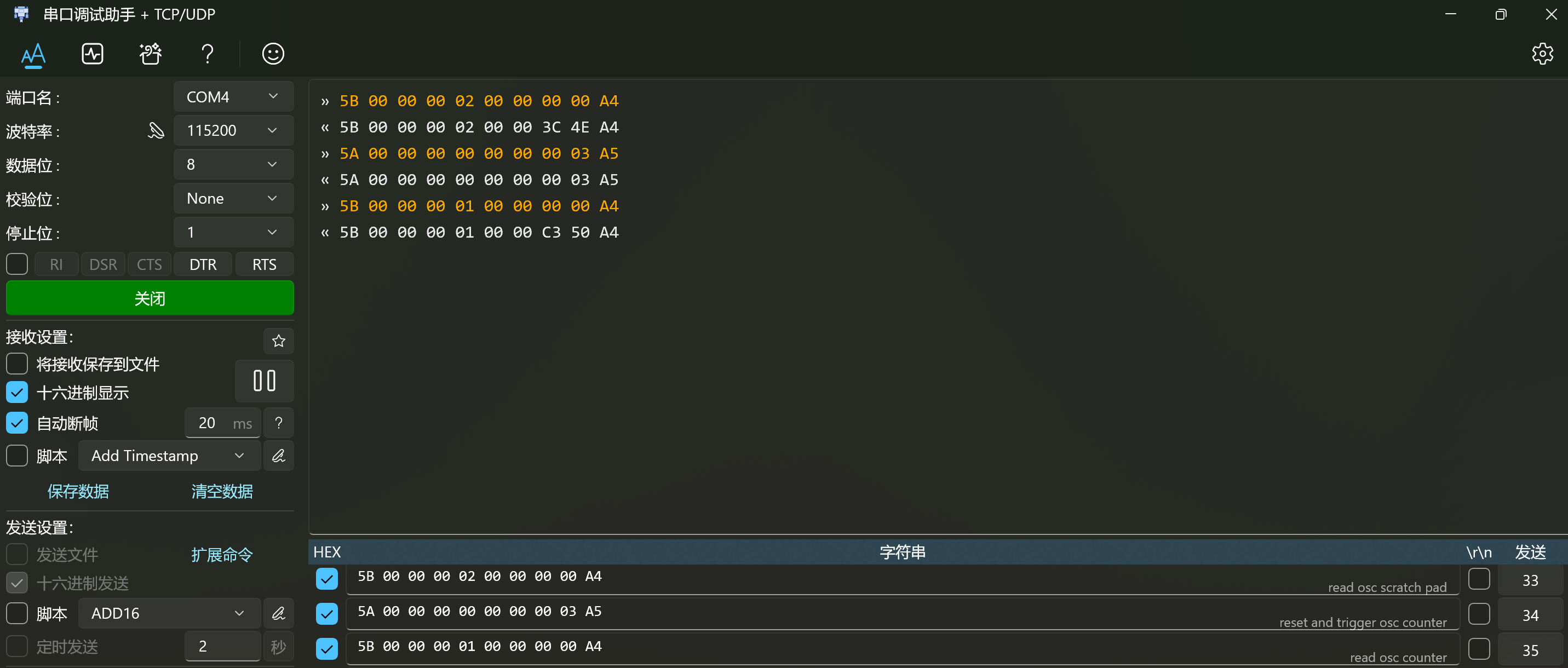
Step-1. OPB Read Scratch Pad Register@(0x00000002), return 32’h00003c4e

Step-2. OPB Write 32’h3 to CNTRL Register@(0x0000000)

Step-3. OPB Read Counter Register @(0x00000001), check return value

Counter Value = 100M/2k = 50,000 (0xC350) (+/ 50）

Freq Value = Counter\_Value \* 2k +/ 0.1M



### **GPIO子模块**

该模块用于读写FPGA GPIO。

* + 1. **寄存器地址与功能列表如下：**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 地址（宏定义） | 地址值 | 寄存器名称 | 位宽 | 读写 | 复位值 | 说明 |
| PWR\_IF | 0x00050000 | POWER\_IF\_REG | 32位 | R/W | 0 | 电源接口 |
| GANTRY\_EMOPS | 0x00050001 | GANTRY\_EMOPS\_IF\_REG | 32位 | R/W | 0 | 龙门EMOPS接口 |
| GANTRY\_96V\_IF | 0x00050002 | GANTRY\_96V\_IF\_REG | 32位 | R/W | 0 | 龙门96V接口 |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

* + 1. **GPIO Output**
    2. **GPIO Input**
    3. **示例：读写Power相关GPIO**

如下命令使能FAN\_EN和P12V\_ISO\_EN

5A 00 05 00 00 00 00 00 03 A5

如下命令读取FAN\_EN和P12V\_ISO\_EN，P24V\_GOOD\_N, FAN\_FAIL\_N

5B 00 05 00 00 00 00 00 00 A4

查找对应的bit得到GPIO的状态

### **FPGA IF子模块 (TODO)**

### **ADC模块**

该模块用于ADC ADS8864的测试, 覆盖ATP测试项**8.6.21-8.6.26等。**

* + 1. **寄存器列表**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **地址宏定义** | **地址范围** | **名称/功能** | **位宽** | **读写** | **说明** | **复位值** |
| D\_RAM\_ADDR | 0x000-0x3FF | 数据RAM | 16位 | R | 采集数据存储区 | 0 |
| CNTRL\_ADDR | 0x800 | 控制寄存器 | 3位 | R/W | control[1]: start, control[2]: reset | 0 |
| CLK\_DIV\_AQ\_ADDR | 0x802 | 采样时钟分频 | 16位 | R/W | 采样时钟分频系数 | 250 |
| CLK\_DIV\_SD\_ADDR | 0x804 | 串行时钟分频 | 16位 | R/W | 串行数据时钟分频系数 | 3 |
| D\_LENGTH\_ADDR | 0x806 | 采样长度 | 10位 | R/W | 采集数据点数, Max=1024 | 5 |
| STATUS\_ADDR | 0x808 | 状态寄存器 | 4位+1 | R | status[0]:busy  [1]:done  [2]:aq timeout  [3]:busy timeou  [31]:ADC\_SDOUT | 0 |
| STATE\_ADDR | 0x80A | 状态机状态 | 5位 | R | 当前状态机状态 | 0 |
| SP\_ADDR | 0x900 | SP寄存器 | 16位 | R | 调试/辅助寄存器 | 0x1122 |

* + 1. **说明**
* **数据RAM**：0x000~0x3FF，采集到的ADC数据，16位宽，只读。
* **控制寄存器**：写入可启动采样、复位模块，读出当前控制状态。
* **分频寄存器**：可配置采样时钟和串行数据时钟分频系数。**无特殊需求，不用修改！**
* **采样长度**：设置一次采集的数据点数。
* **状态寄存器**：包含采集状态、完成标志、超时等信息。
* **状态机状态**：可读出当前状态机的状态码，用于辅助调试。
* **SP寄存器**：辅助/调试用途。
  + 1. **举例**

STEP1. 配置ADC SEL\_MEX，选择哪一路采样信号，参考Section-6.GPIO模块

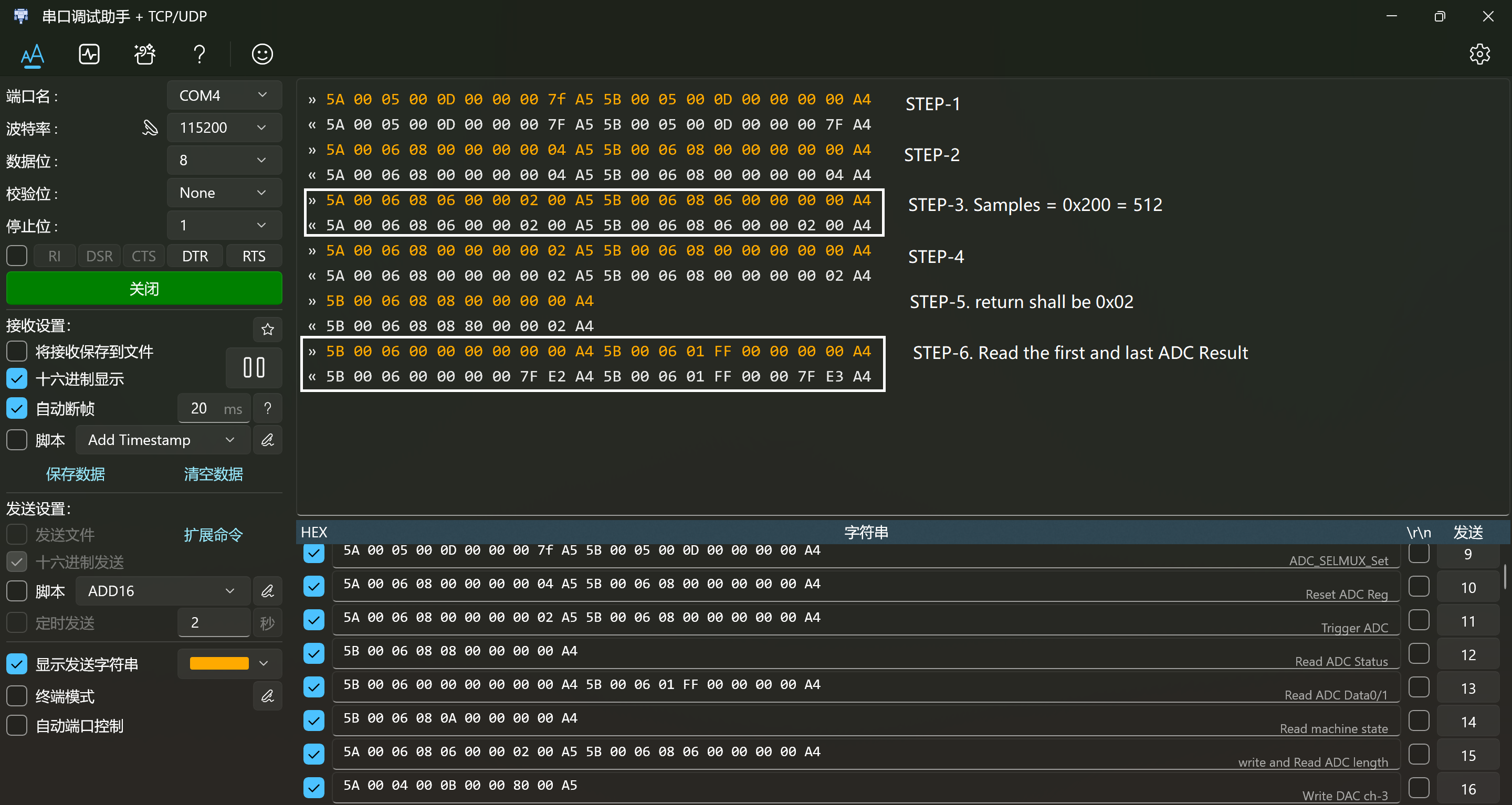
STEP2. 复位ADC寄存器

STEP3. 配置采样长度

STEP4. 触发ADC采数

STEP5. 读ADC状态

STEP6. 读ADC结果



### **DAC模块**

该模块用于DACx0504的测试，覆盖ATP测试项**8.1.9。**

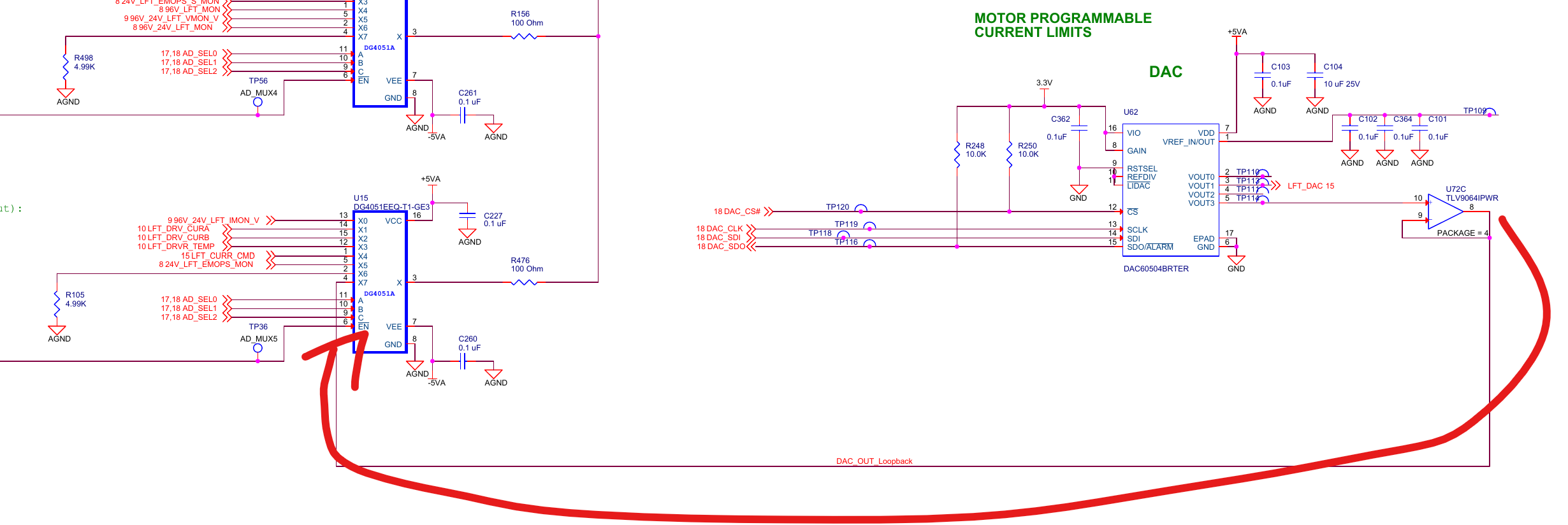
* + 1. **寄存器列表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **地址宏定义** | **地址** | **名称/功能** | **位宽** | **读写** | **说明** |
| DAC0\_ADDR | 0x8 | DAC0数据 | 16bit | R/W | DAC通道0数据 |
| DAC1\_ADDR | 0x9 | DAC1数据 | 16/24 | R/W | DAC通道1数据 |
| DAC2\_ADDR | 0xA | DAC2数据 | 16/24 | R/W | DAC通道2数据 |
| DAC3\_ADDR | 0xB | DAC3数据 | 16/24 | R/W | DAC通道3数据 |

* + 1. **说明**

写DAC0/1/2/3数据寄存器，DAC的输出对应的电平Out = Data/65536\*5

* + 1. **举例：DAC-ADC环回测试**



STEP-1. OPB Write DAC data for VOUT3

STEP-2. OPB Write to select DAC\_OUT\_loopback as ADC input

* Measure Voltage @TP70 (4.99V for DAC data = 0xFFFF )
* Measure Voltage @TP70 (2.50V for DAC data = 0x8000 )

STEP-3. OPB Write to reset ADC Reg

STEP-4. OPB Write to Trigger ADC

STEP-5. OPB Read for ADC Status

STEP-6. OPB Read for ADC Result

* + ADC Data = 0Xff51 /65535\*5=4.98V When DAC data = 0xFFFF
  + ADC Data = 0X7FE0 /65535\*5=2.49V When DAC data = 0x8000

****

### **RS422模块**

1. **MSSB STN模块**

该模块用于MSSB STN(U86)的测试, 覆盖ATP测试项**8.4.10-8.4.12**

1. **寄存器列表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **偏移地址** | **名称/功能** | **位宽** | **读写** | **说明** |
| 0x0 | mssb\_config | 32位 | R/W | [31]:trans\_start，写1启动传输，自动清零；[19:0]:trans\_bytes，传输字节数 |
| 0x1 | mssb\_status | 32位 | R | [1:0]：00=busy，01=完成无错，11=完成有错 |
| 0x2 | mssb\_recv\_bytes | 32位 | R | [19:0]: 实际接收字节数 |
| 0x3 | mssb\_error\_bytes | 32位 | R | [19:0]: 错误字节数 |

1. **说明**
   1. **mssb\_config**

* [31] trans\_start：写1启动传输，自动清零
* [19:0] trans\_bytes：本次要传输的字节数（最大1M，默认10K）
  1. **mssb\_status**
* [1:0] 状态：00=busy，01=完成无错，11=完成有错
  1. **mssb\_recv\_bytes**
* [19:0] 实际接收字节数
  1. **mssb\_error\_bytes**
* [19:0] 错误字节数（接收数据与预期不符的字节数）

1. **举例**

参考MSSB SRV模块测试

1. MSSB SRV模块

该模块用于MSSB SRV(U98)的测试, 覆盖ATP测试项**8.4.10-8.4.12**

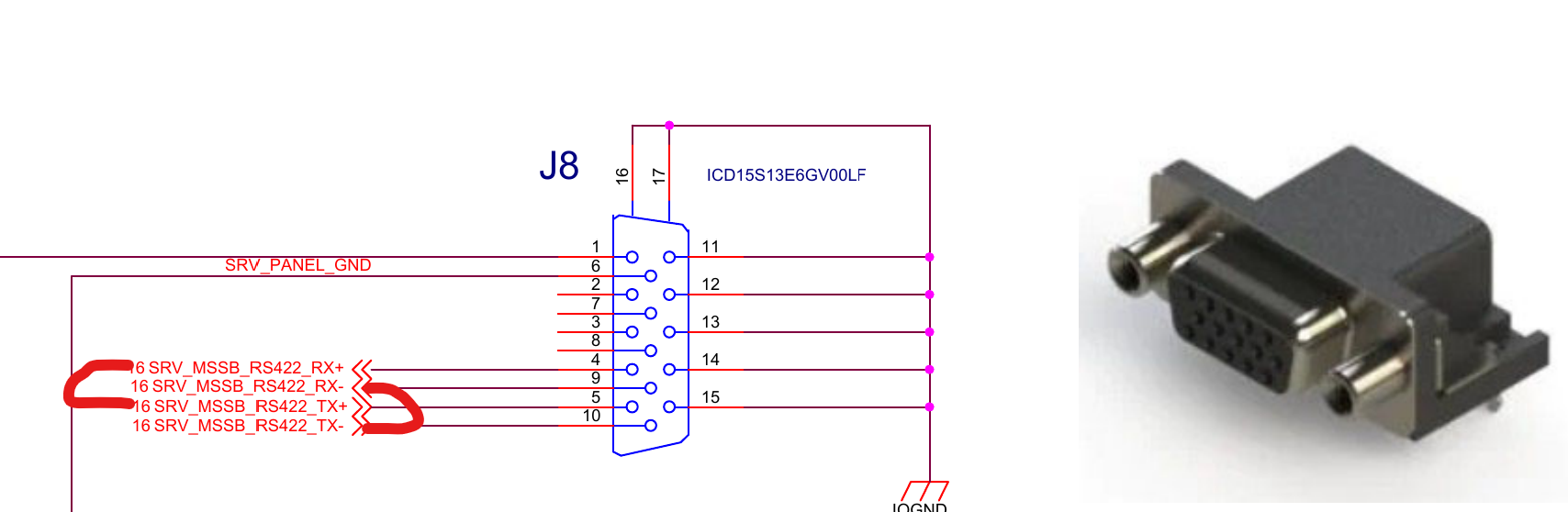
参考MSSB STN模块的测试，注意两者的基地址不同：

`define MSSB\_STN\_ADDR           32'h00090000        /\* MSSB address \*/

`define MSSB\_SRV\_ADDR           32'h00091000        /\* MSSB address \*/

**举例**

* Connect RS422\_RX+ to RS422\_TX+, pin-4 to pin-5
* Connect RS422\_RX- to RS422\_TX-, pin-9 to pin-10

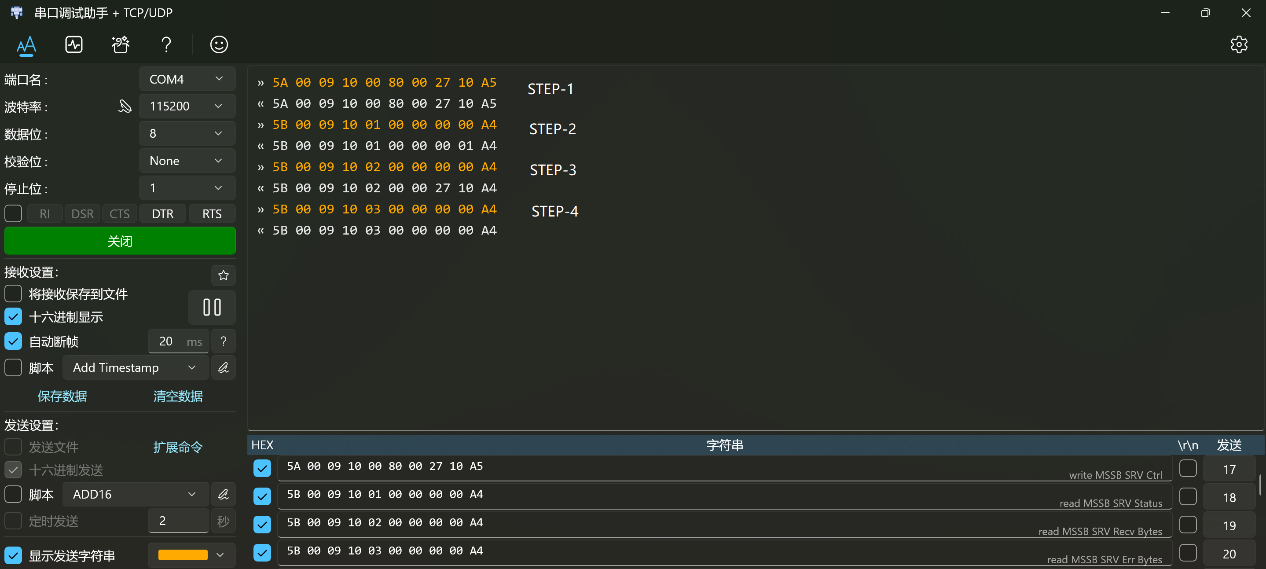


STEP-1. OPB Write to trigger MSSB Test, TRANS\_BYTES = 0x2710 (10K)

STEP-2. OPB Read for MSSB Status, the return value shall be 0x01

STEP-3. OPB Read for MSSB Recv Bytes, it shall be 0x2701 (10K)

STEP-4. OPB Read for MSSB Err Bytes, it shall be 0



### **CAN模块 (TODO)**