

## 实验十六：双口 RAM 实验——基于双口 RAM 的 ARM 与 FPGA 通信

### 一、实验目的与意义

- 1、掌握双口 RAM IP 核的调用及例化方法。
- 2、掌握 RAM 读写时序。
- 3、掌握 QuartusII 的使用方法。

### 二、实验设备及平台

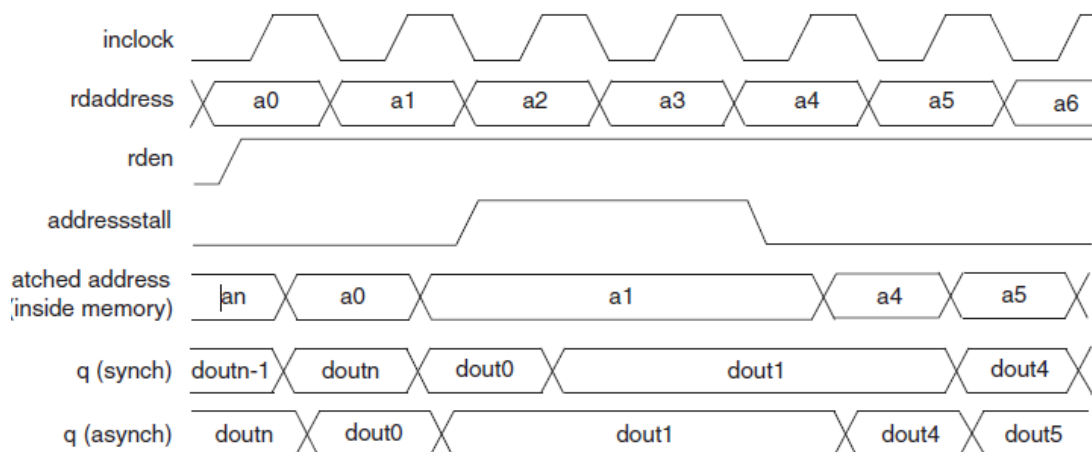
- 1、iCore4 双核心板。
- 2、Blaster（或相同功能）仿真器。
- 3、JLINK（或相同功能）仿真器。
- 4、Micro USB 线缆。
- 5、Keil MDK 开发平台。
- 6、Quartus 开发平台。
- 7、电脑一台。

### 三、实验原理

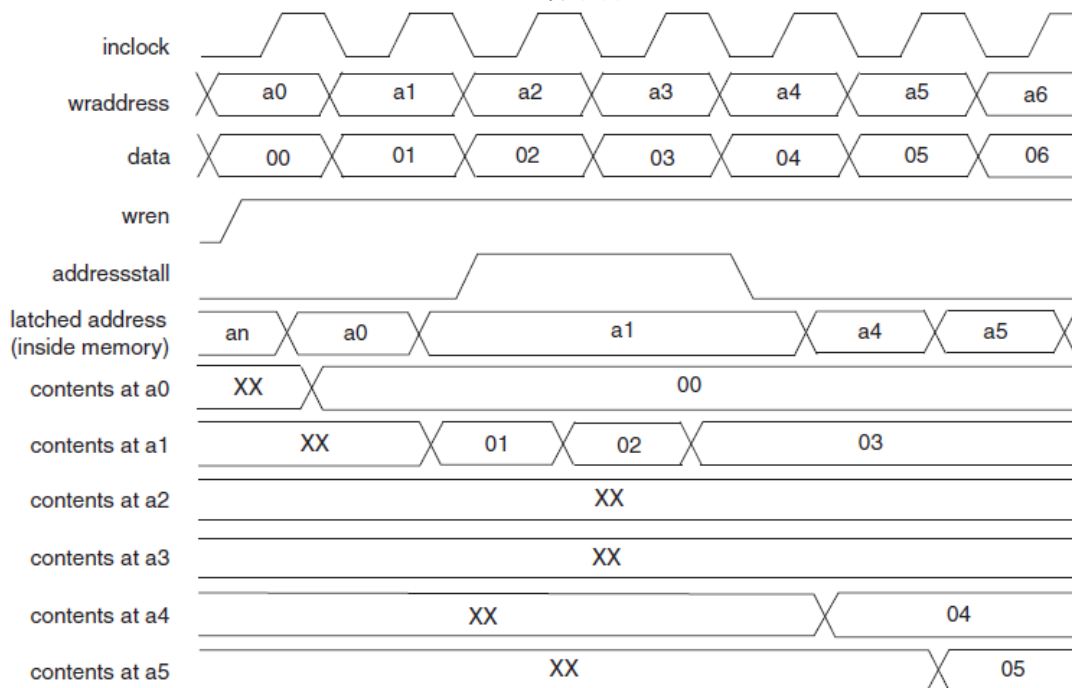
RAM 全称是 Random Access Memory（随机存取存储器），是与处理器进行数据交互的存储器。既可向指定地址单元写入信息也可从指定地址单元中读取信息，掉电后数据丢失，属于易失性存储器。

双口 RAM 分为伪双口 RAM 和真双口 RAM 两种，伪双口 RAM 一个端口只读，另一个端口只写；而真双口 RAM 两个端口都能读写。在 SPI 通信实验中，使用的就是伪双口 RAM，因此，在本实验中只讲真双口 RAM。从整体上讲，真双口 RAM 和伪双口 RAM 都能同时进行读写操作；虽然有两个端口，两套数据线和地址线，但是存储数据的空间单元是共享的，因此存储后的数据也是共享的。其基本工作时序和普通 RAM 一致，都是以时钟上升沿作为触发事件，在上升沿到来前保证地址和数据稳定即可将数据准确的写入对应的地址

单元中，读取时，也是如此。其读写时序如图 16-1 所示。



16-1 (a) 读时序图



16-1 (b) 写时序图

由于双口 RAM 有两条相互独立的地址线、数据线和控制线，能实现数据共享，本实验针对双口 RAM 的工作特点进行实验验证。实验中通过 FPGA 对 RAM 的 A 端口进行读写操作，RAM 的 A 端口先向 RAM 中写入 0 到 255 的数据，然后再读出来，如此循环操作，同时通过 Commix 向 STM32 发送命令，STM32 收到命令后通过 FSMC 总线与 FPGA 实现通信，对 RAM 的 B 端口进行都控制，发送一次读命令 RAM 的 B 端口把相应地址里的数据发送至 ARM，再发送至 Commix 显示出来以验证实验的成功性。再本实验中验证了双口 RAM 的两种工作模式：1、双口 RAM 一个端口写操作，另一个端口读操作。2、双口 RAM 两个端口同时读操作，实验原理图如下图所示。

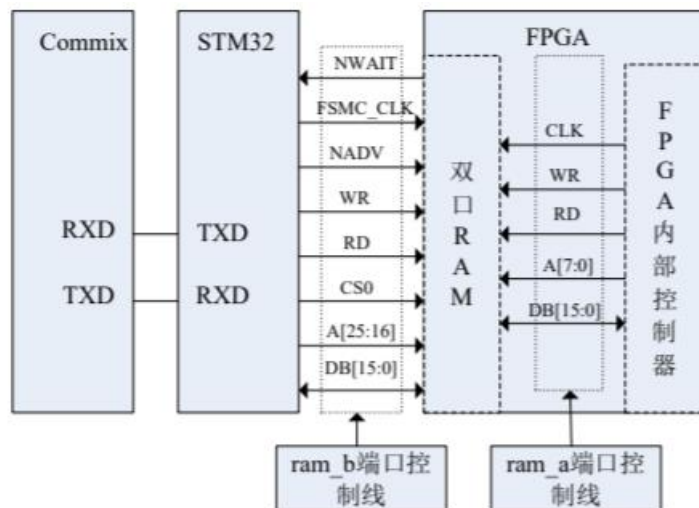


图 16-2

## 四、RAM IP 核调用

1、打开 Tool -> Megawizard Plug-In Manager, 如图 16-3 所示。

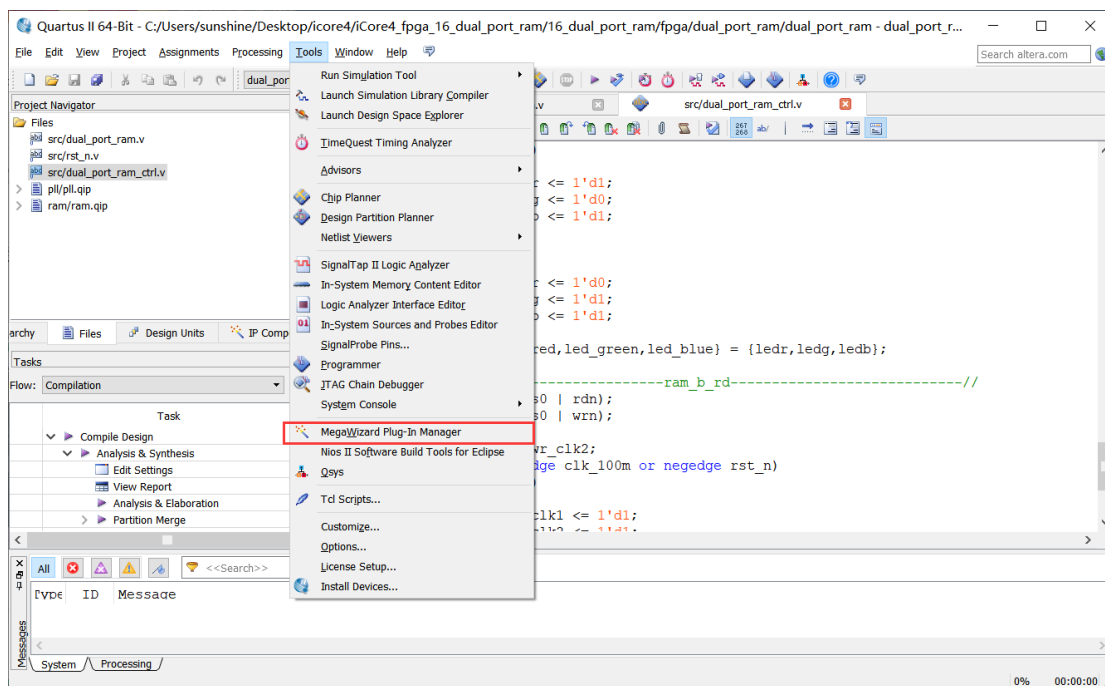


图 16-3

2、在下面界面中保持默认，直接 Next 即可，如图 16-4 所示。

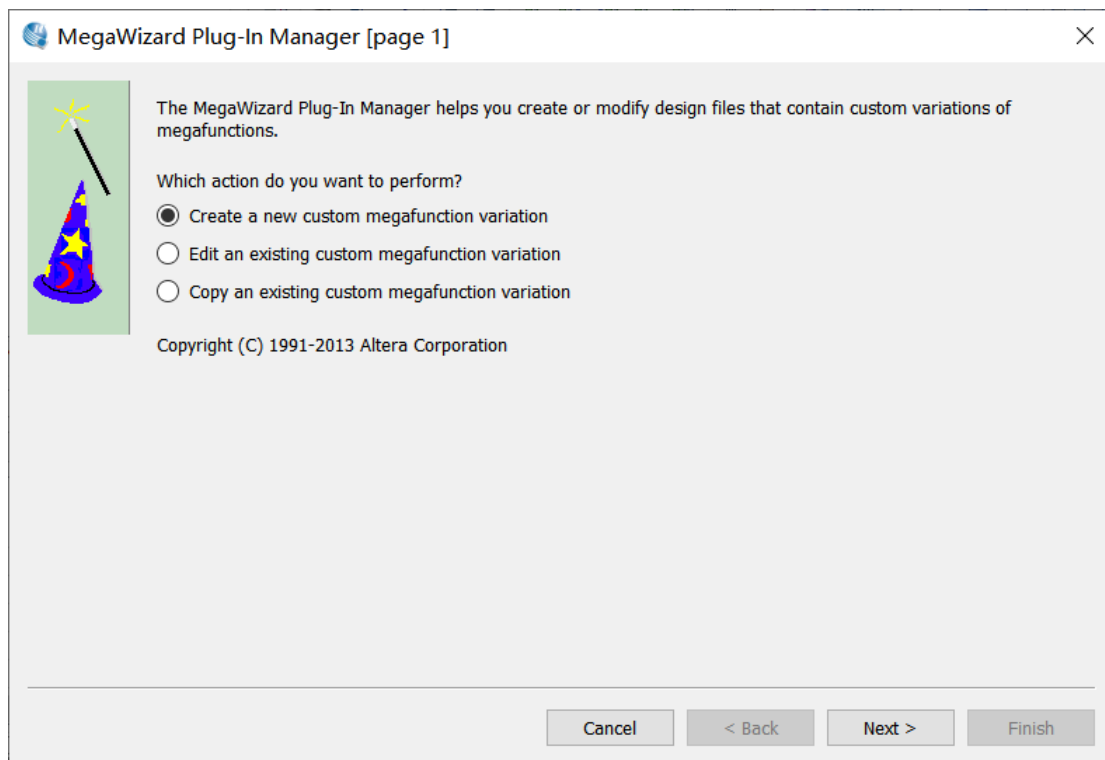


图 16-4

3、在该界面中选择双口 RAM IP 核、芯片类型、硬件描述语言类型及 IP 核相关文件存储位置，如图 16-5 所示。

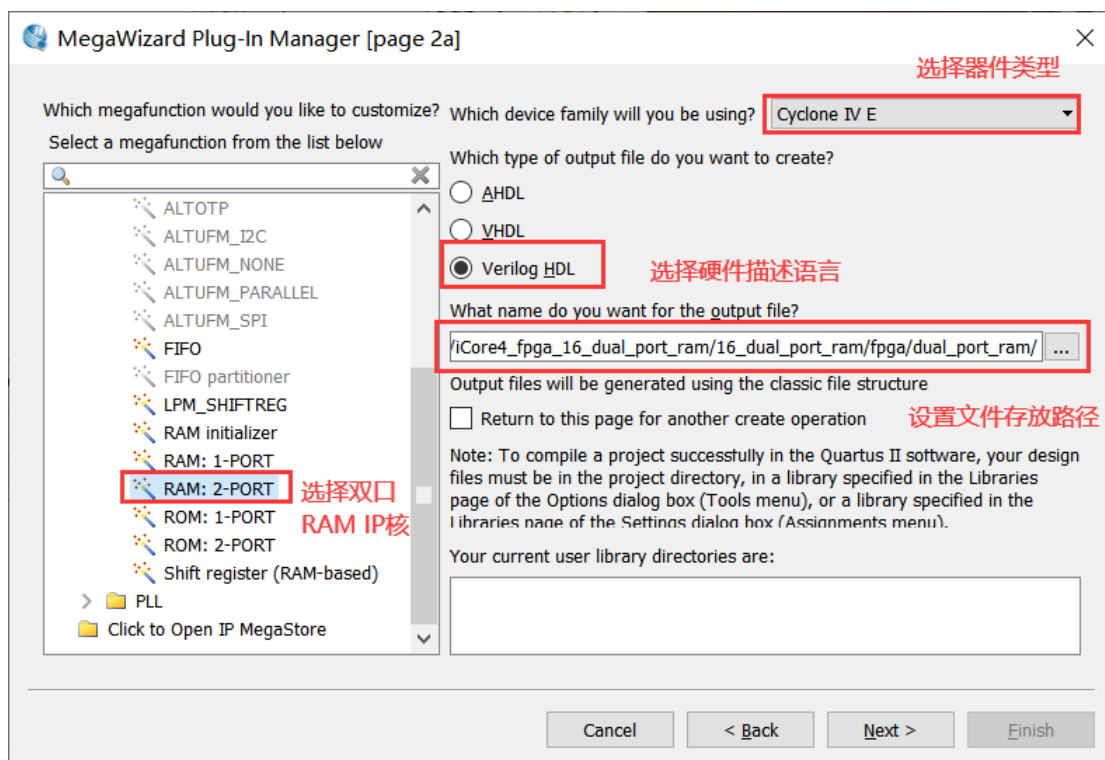


图 16-5

4、在该对话框中选择真双口 RAM（两个输入两个输出），存储深度以字节为单位，如图 16-6 示。

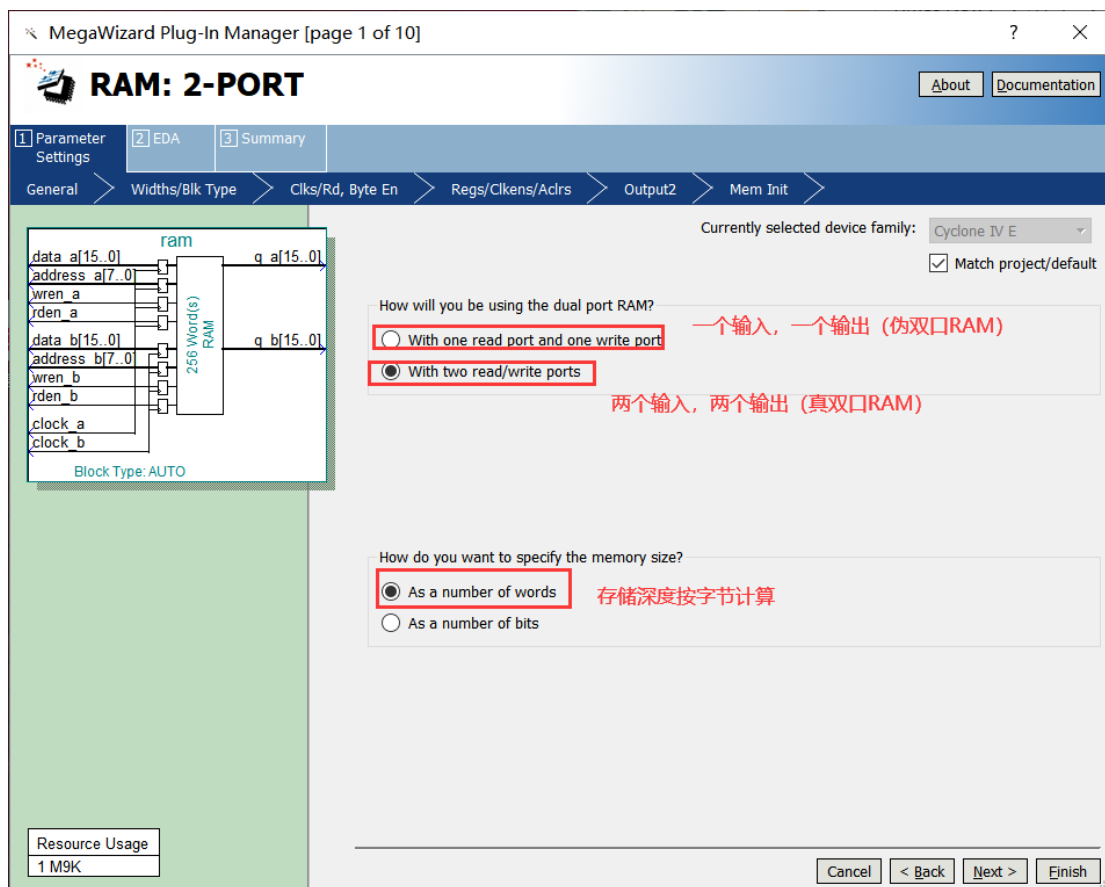


图 16-6

5、在该对话框中设置存储深度，如图 16-7 所示。

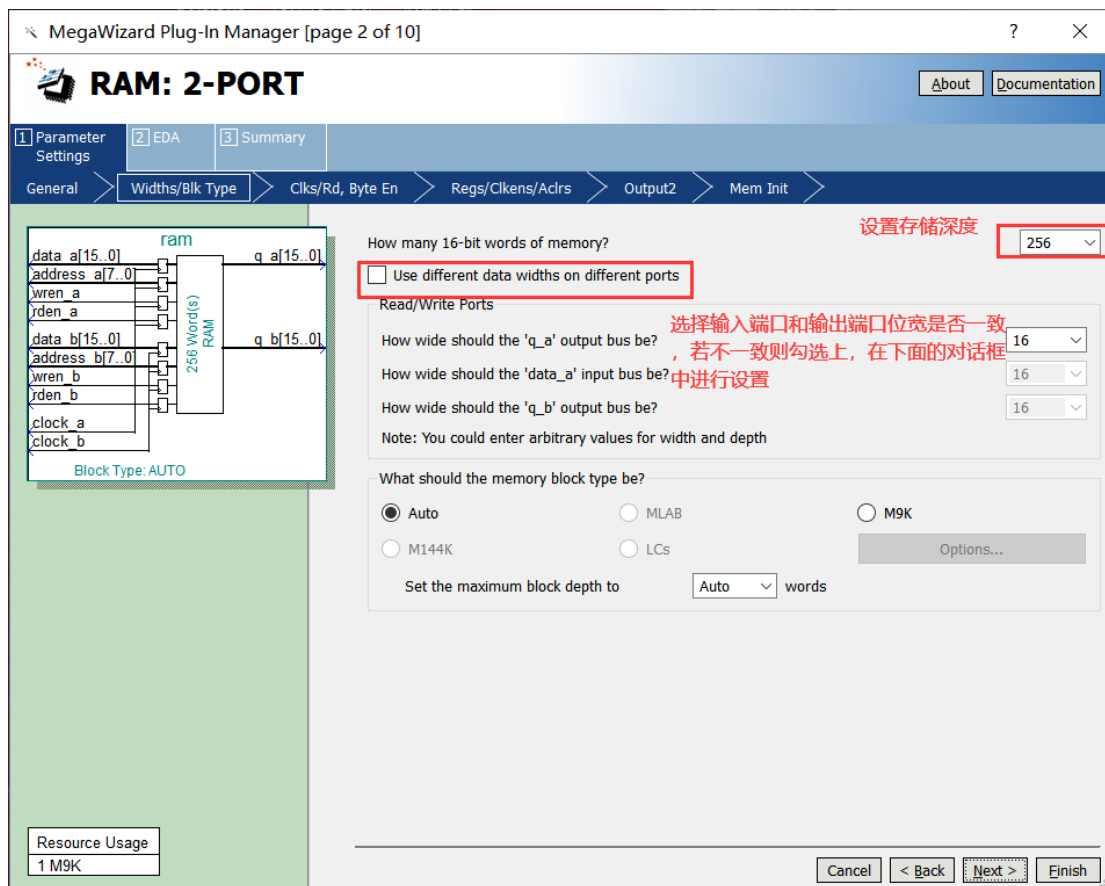


图 16-7

6、在此对话框选择时钟信号输入方式、增加读使能信号，如图 16-8 所示。

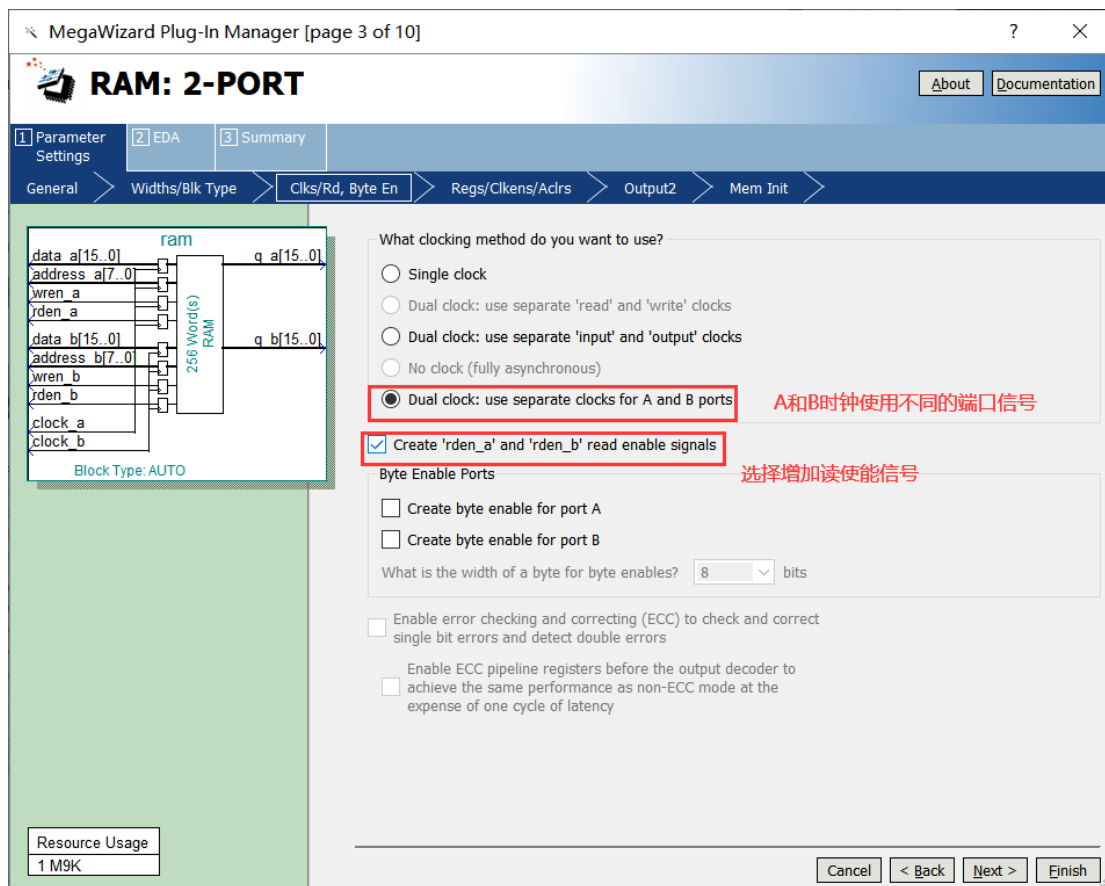


图 16-8

7、直接 Next 即可，如图 16-9 所示。

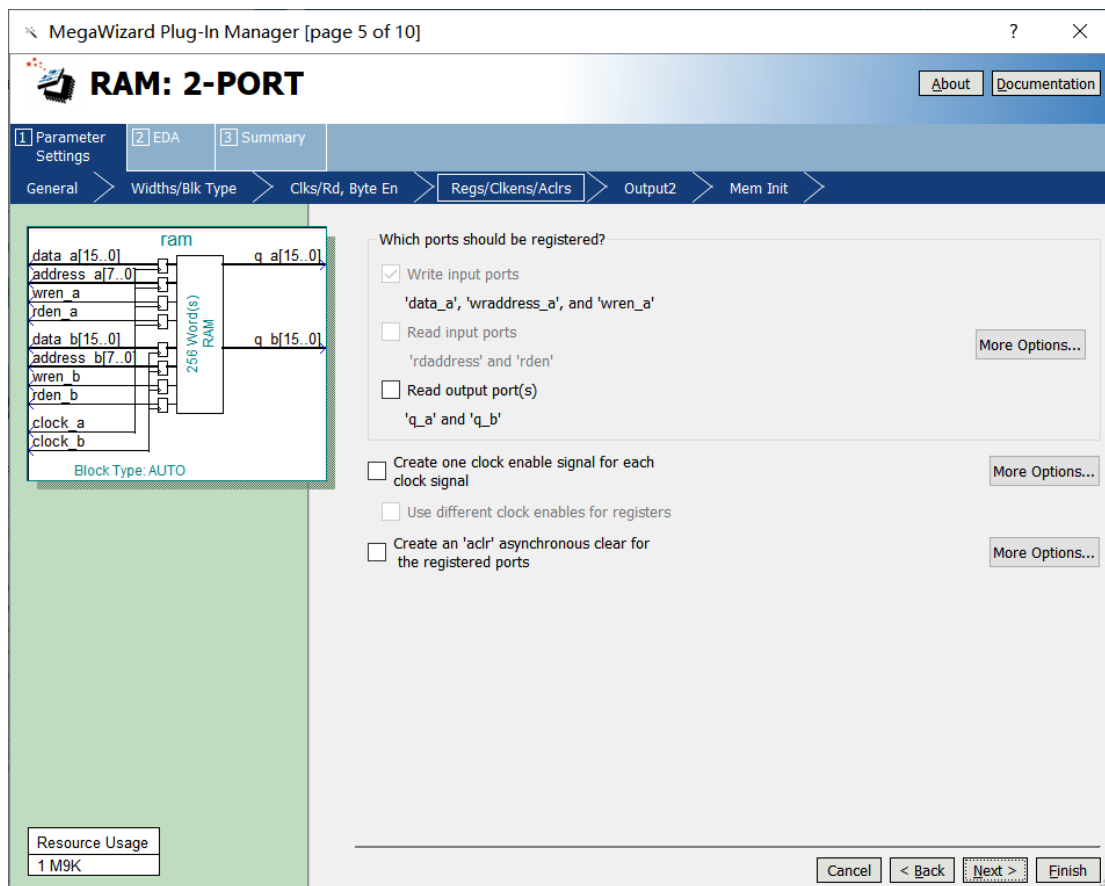


图 16-9

8、该对话框直接 Next 即可，如图 16-10 所示。

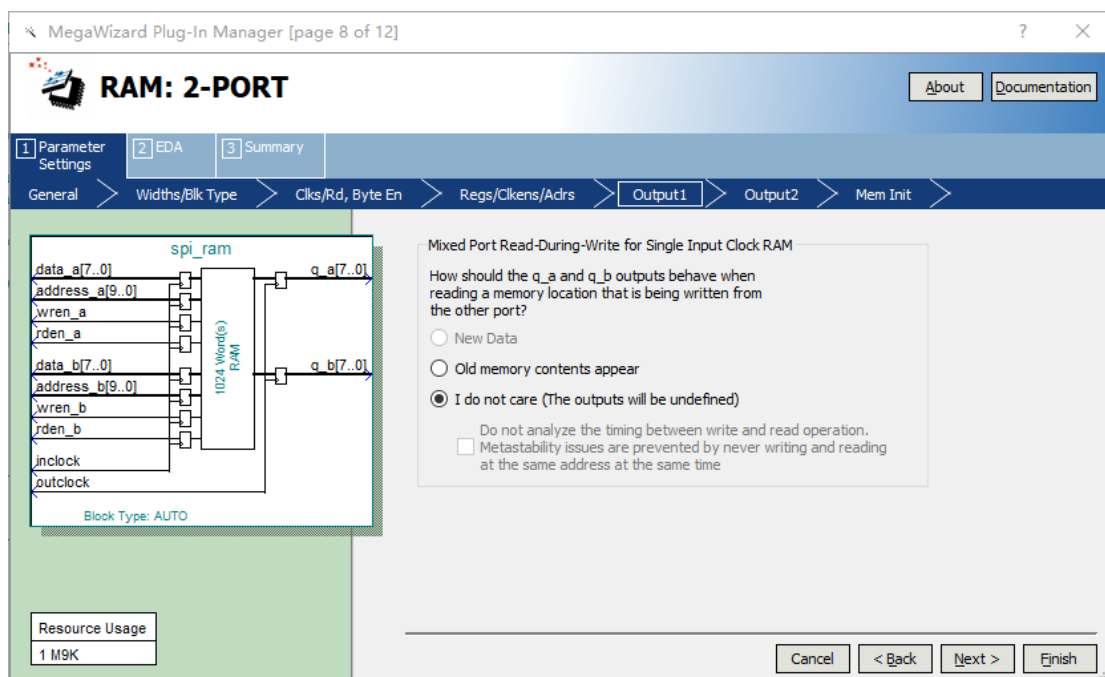


图 16-10



9、选择读出的数据是新数据还是老数据，一般设置新数据（New Data），如图 16-11 所示。

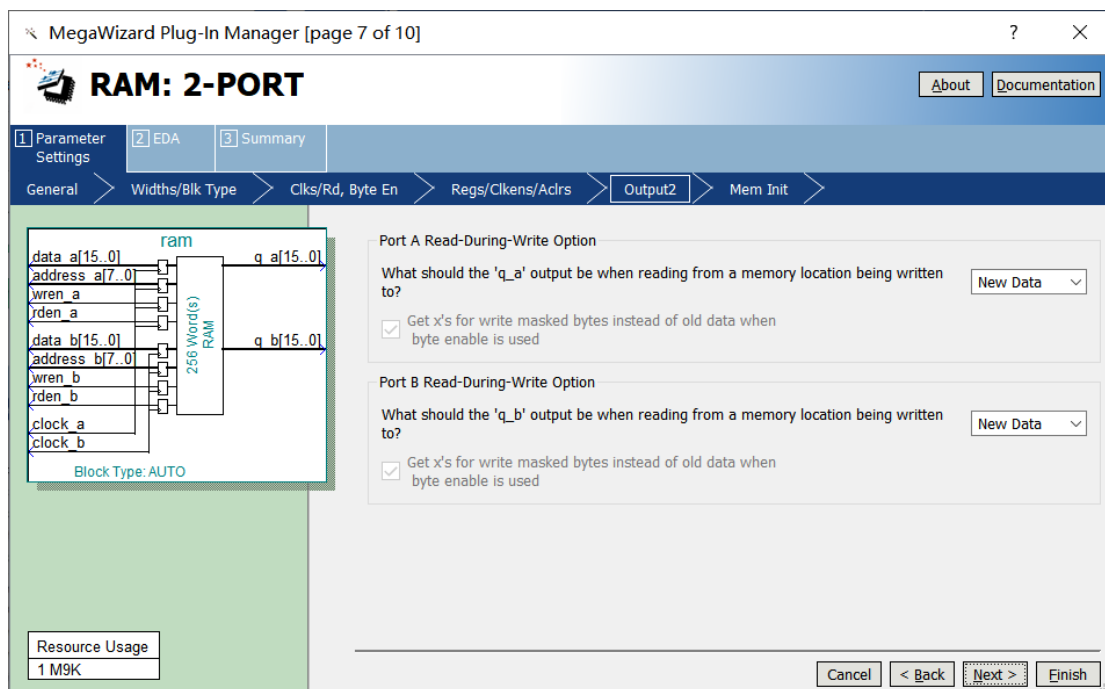


图 16-11

10、该对话框设置 RAM 无初始值，直接 Next，如图 16-12 所示

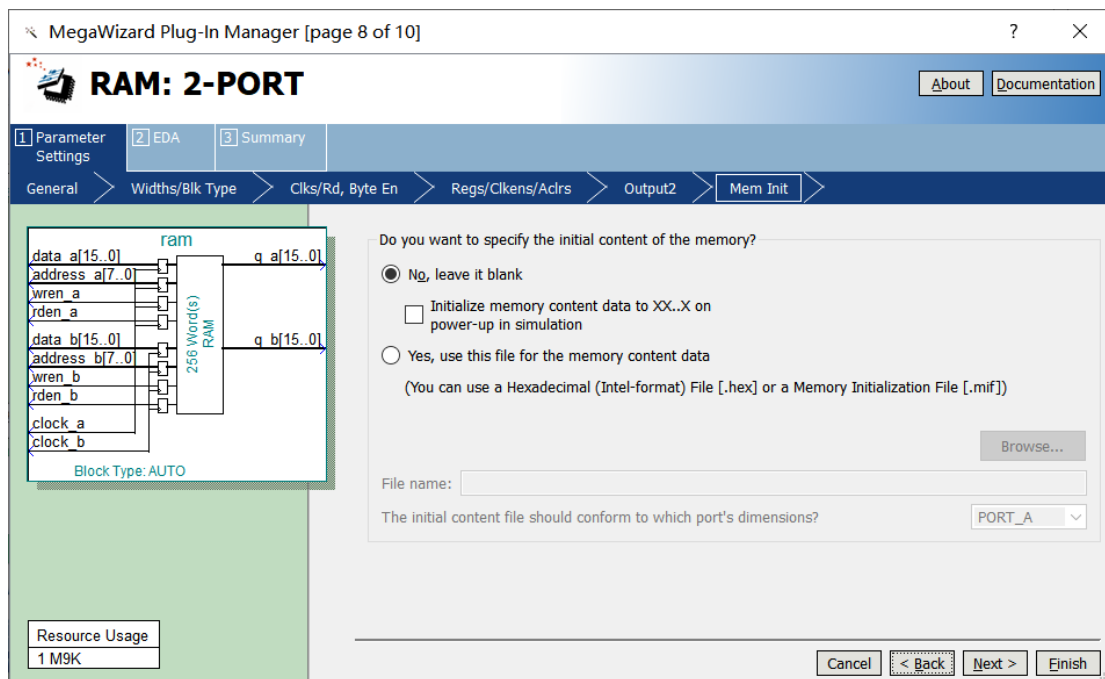


图 16-12

11、直接 Next，如图 16-13 所示。

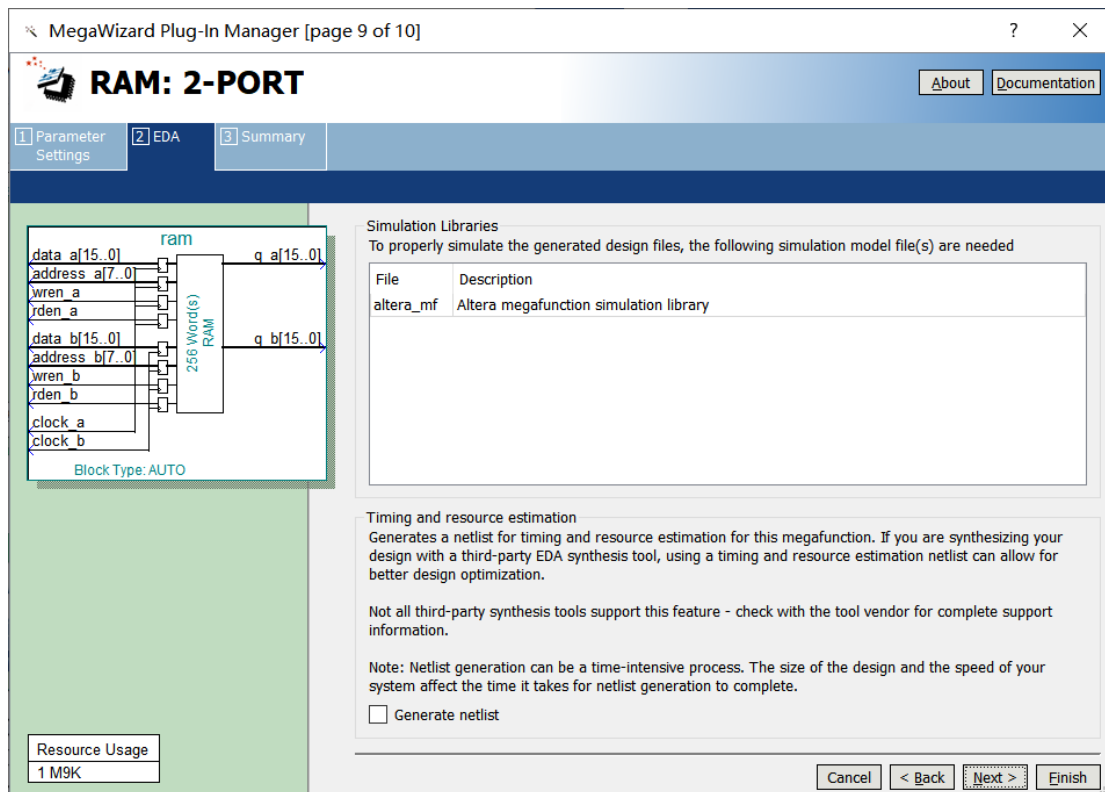


图 16-13

12、选择生成的 IP 核关联文件类型，如图 16-14 所示。

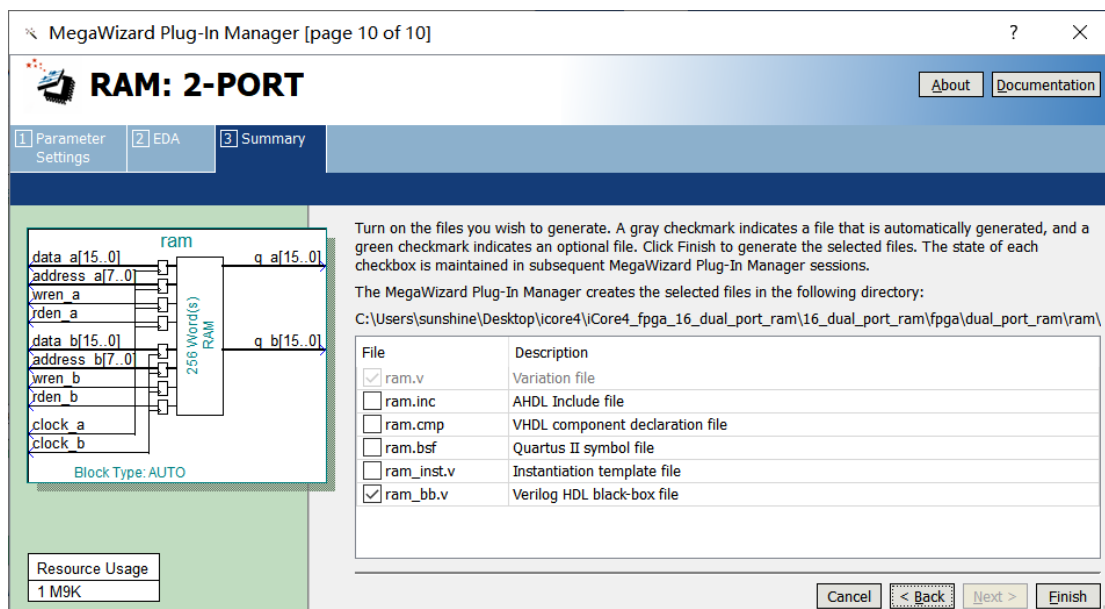


图 16-14

13、IP 核设置完成后出现下面对话框，点击 Yes，否则将无法正常使用 RAM，如图 16-

15 所示。

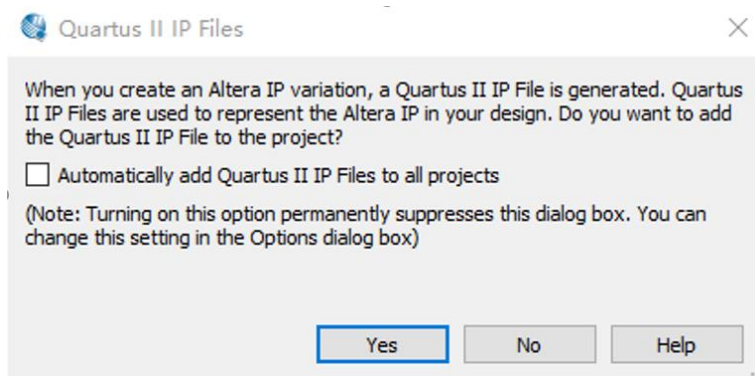


图 16-15

## 五、代码讲解

IP 核调用只是生成相关的模块文件，在应用中要实现存储、读写功能，还需要对 IP 核进行实例化操作，双口 RAM IP 核例化代码如下：

```
//-----//
//双口 RAM 的缺点是与单口 RAM 相比，相同容量的 RAM 占用逻辑资源比较多。
//优点在于具有 2 个输入 2 个输出端口，两个端口都可用于输入输出，可实现同时读写，便于控制。
ram u1 (
    .data_a(data_a),
    .address_a(address_a),
    .wren_a(wren_a),
    .rden_a(rden_a),
    .clock_a(!clk_a),
    .q_a(dataout_a),

    .data_b(db),
    .address_b(ab),
    .wren_b(1'd0),
    .rden_b(!rd),
    .clock_b(clk_b),
    .q_b(dataout_b)
);
```

## 六、实验步骤

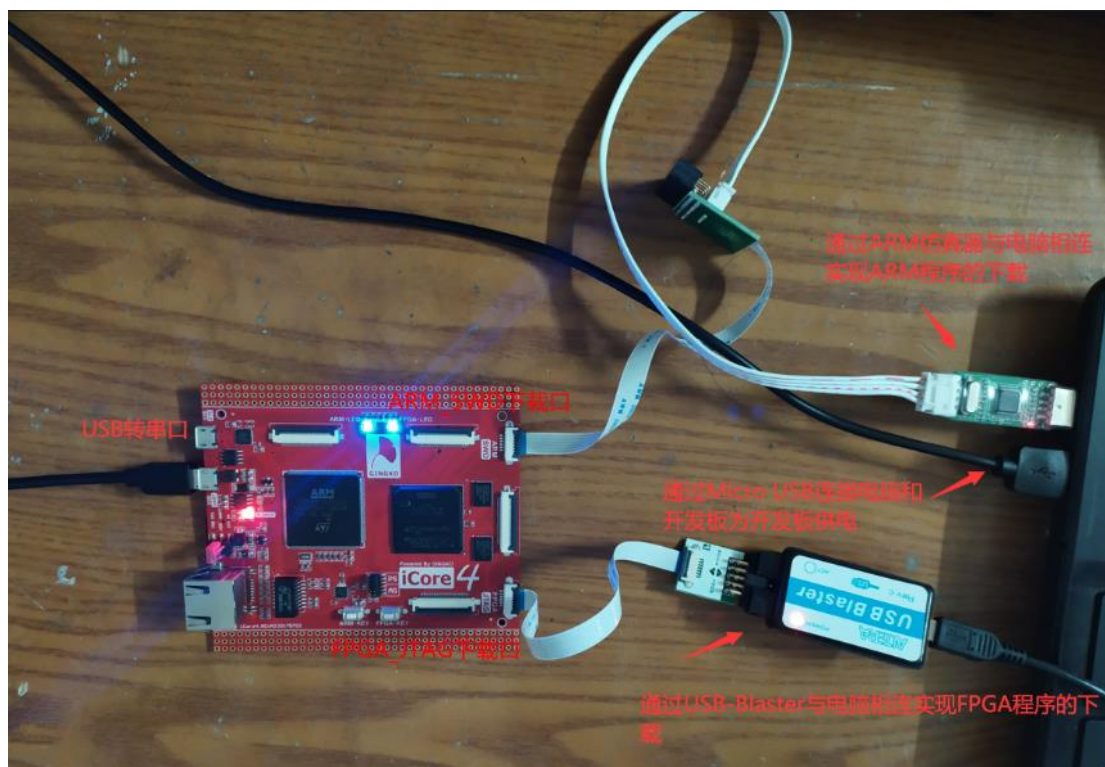


图 16-16

- 1、将硬件正确连接，如图 16-16 所示。
- 2、打开 Commix 串口精灵，找到对应的 COM 口打开。
- 3、打开 QuartusII 开发环境，并打开实验工程。
- 4、将 FPGA 程序下载至 iCore4 上。
- 5、打开 Keil MDK 开发环境，并打开实验工程。
- 6、烧写 ARM 程序到 iCore4 上
- 7、输入串口命令，观察实验现象。

## 七、实验现象

一方面，观察 FPGA\_LED 灯的颜色，FPGA 向 RAM 中先写入数据，然后再读出来，读出数据与其地址相等则绿灯常亮。否则红灯亮。

另一方面，通过 Commix 发送读命令读取 RAM 地址中的数据，对比写入数据与读取的数据，可知道写入与读取的数据相同。

命令格式:

读命令: READ:地址 (0-255) \CR\LF

例如: READ:25\CR\LF

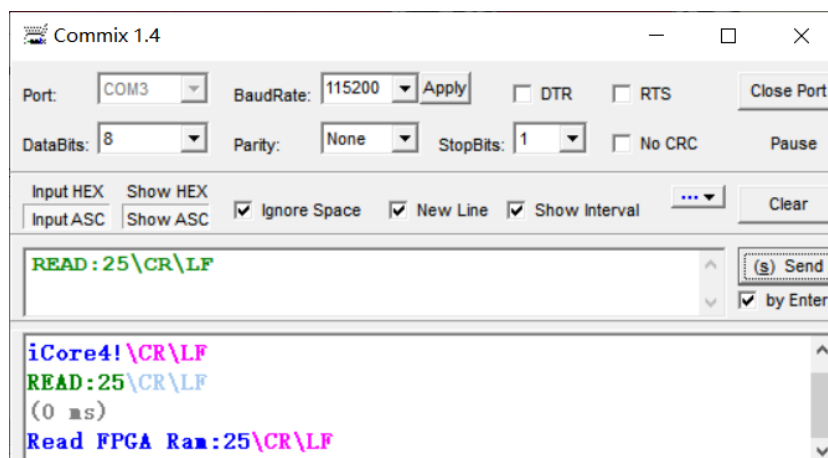


图 16-17