

第八届

## 全国大学生集成电路创新创业大赛

报告类型\*: 设计报告

参赛杯赛\*: 中科芯杯

作品名称\*: 高速多端口共享缓存管理模块

队伍编号\*: CICC3263

团队名称\*: 随便起了个名字

## 一、 赛题分析

随着近代网络技术的高速发展，互联网流量呈爆炸式增长，企业和家庭对网络带宽的要求日益提高。在网络设备对数据包的处理中，存储管理、调度占去了大部分时间，因此大多数存储器的低速缓存能力成为了限制网络处理器进一步提高的瓶颈。支持高速数据存储的 SRAM 管理模块，避免数据包的长度和存储器数据通道的数量对存储器资源的影响可以通过内存回收动态调整空间节约达到存储资源的目的，提高数据存储效率，提升数据校验纠错的能力。

需要设计一款可对 SRAM 进行有效管理的 SRAM 控制器 IP，模块框图如下图 1.1 所示。

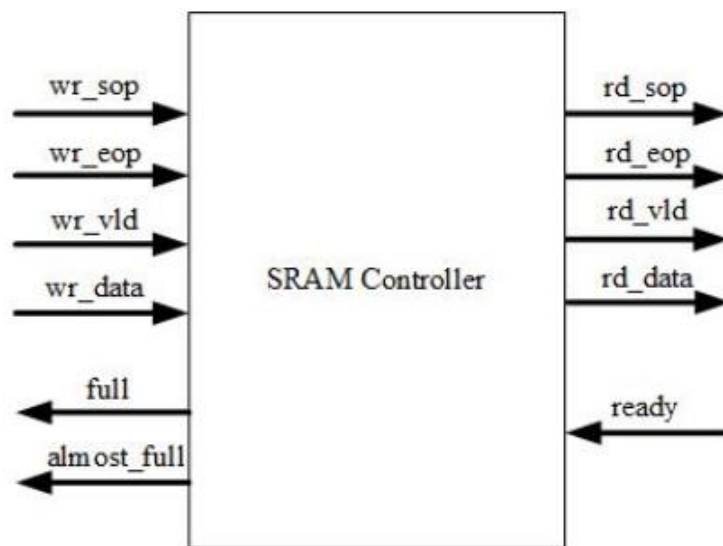


图 1.1 SRAM 控制器 IP 模块框图

该 SRAM 控制器 IP 的具体要求如下所示：

- (1) 支持管理不少于 32 块 256K bit 的 SRAM 单元，满足 8M bit 容量，时钟>250Mhz，支持高速缓存。
- (2) 支持 16 个端口同时读写缓存，每端口数据传输带宽可达到 1G bps。
- (3) 每端口支持 8 个优先级队列，按队列缓存数据。
- (4) 支持按包缓存、调度数据，数据包长 64-1024 字节，每包帧格式含控制帧部分（包括但不限于目的地址、优先级）。

## 二、 详细设计

### 2.1 系统框图

本作品的整体框图如下图 2.1 所示。

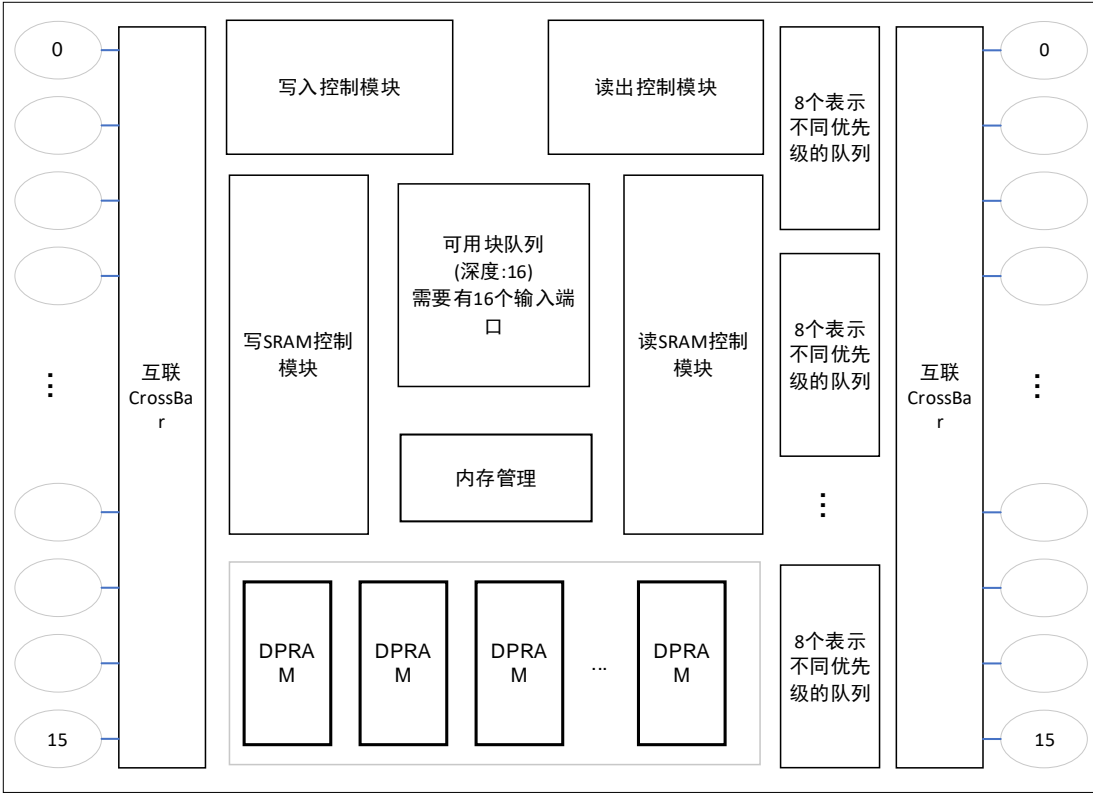


图 2.1 整体框图

系统中核心模块有互联 crossbar、dpram 块、可用块队列、内存管理、写 SRAM 控制模块、读 SRAM 控制模块以及每个端口的不同优先级队列。

crossbar 模块用于将各个端口与 SRAM 的读写端口进行连接, dpram 块为合并后的双端口 SRAM, 可用块队列实际上记录的为每个 dpram 块中一个可以写入位置的地址, 内存管理模块实现 dpram 内存的分配与释放管理, 读、写 SRAM 控制模块实现底层的 dpram 读写功能。

### 2.2 数据位宽设计

赛题要求管理不少于 32 块 256Kbit 的 SRAM, 即总容量不少于 8Mbit, 本作品将 SRAM 的位宽定为 32bit, 每个 SRAM 的深度为 8192。

输入输出端口的位宽设计为 32bit, 接口时钟为 250MHz, 每个端口的数据

传输带宽为 8Gbps，满足赛题的 1Gbps 要求。

## 2.3 数据包格式设计

本作品的数据包格式，如下图 2.2 所示。

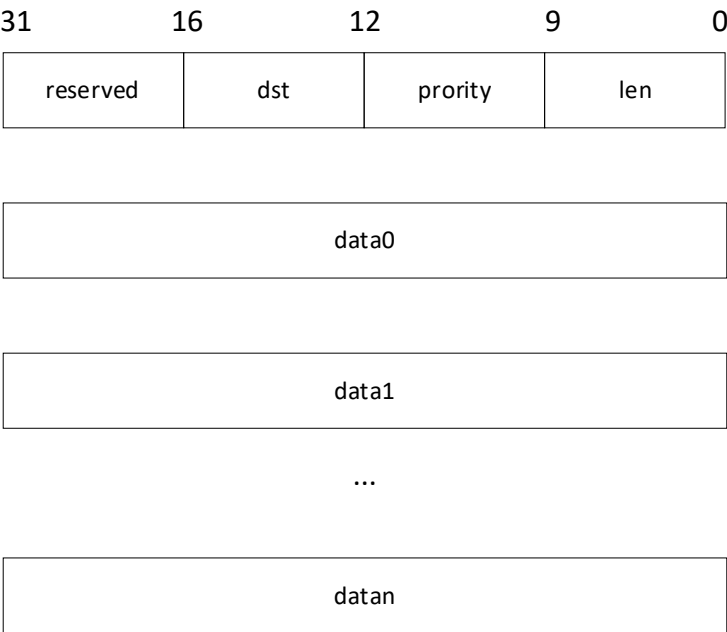


图 2.2 数据包格式

len 字段为数据长度（64-1024Bytes），priority 字段为数据包的优先级（0-7），dst 字段为数据包目的端口，目的端口有 16 个（0-15），reserved 字段暂时保留。

## 2.4 子模块设计

### 2.4.1 DPRAM 模块

真双端口 RAM 指的是有两个读写端口，每个端口都可以独立发起读或者写。两组数据线和地址线，一组是读或者写数据线，另外一组也是读或者写数据线，同一个时刻两个端口可以同时访问，两个端口可以是读和读，也可以是写和读，还可以是写和写。

本作品需要管理至少 32 块 256Kbit 的 SRAM，目前设定为 32 块。将 32 块 SRAM 分成 16 组，每组 2 个 SRAM，称为一个 DPRAM 块。DPRAM 模块的框图如下图 2.3 所示。

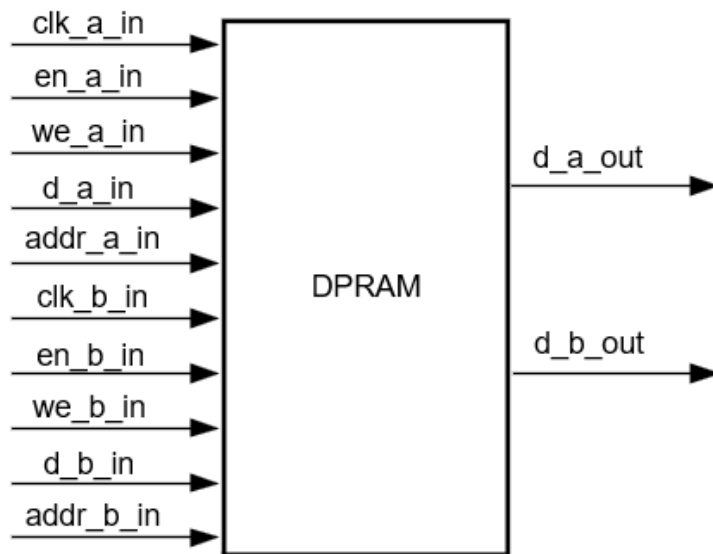


图 2.3 DPRAM 模块框图

DPRAM 模块的端口信号的说明如下表 2.1 所示。

表 2.1 DPRAM 模块的端口信号说明

端口名	方向	端口说明
en_a_in	输入	端口 A 使能信号，高有效
we_a_in	输入	端口 A 写使能信号，高有效
addr_a_in	输入	端口 A 的 RAM 地址
d_a_in	输入	端口 A 的写数据端口
clk_a_in	输入	端口 A 的时钟信号
d_a_out	输出	端口 A 的读数据端口
en_b_in	输入	端口 B 使能信号，高有效
we_b_in	输入	端口 B 写使能信号，高有效
addr_b_in	输入	端口 B 的 RAM 地址
d_b_in	输入	端口 B 的写数据端口
clk_b_in	输入	端口 B 的时钟信号
d_b_out	输出	端口 B 的读数据端口

### 三、 验证平台

设计的验证平台框图如下图 3.1 所示。

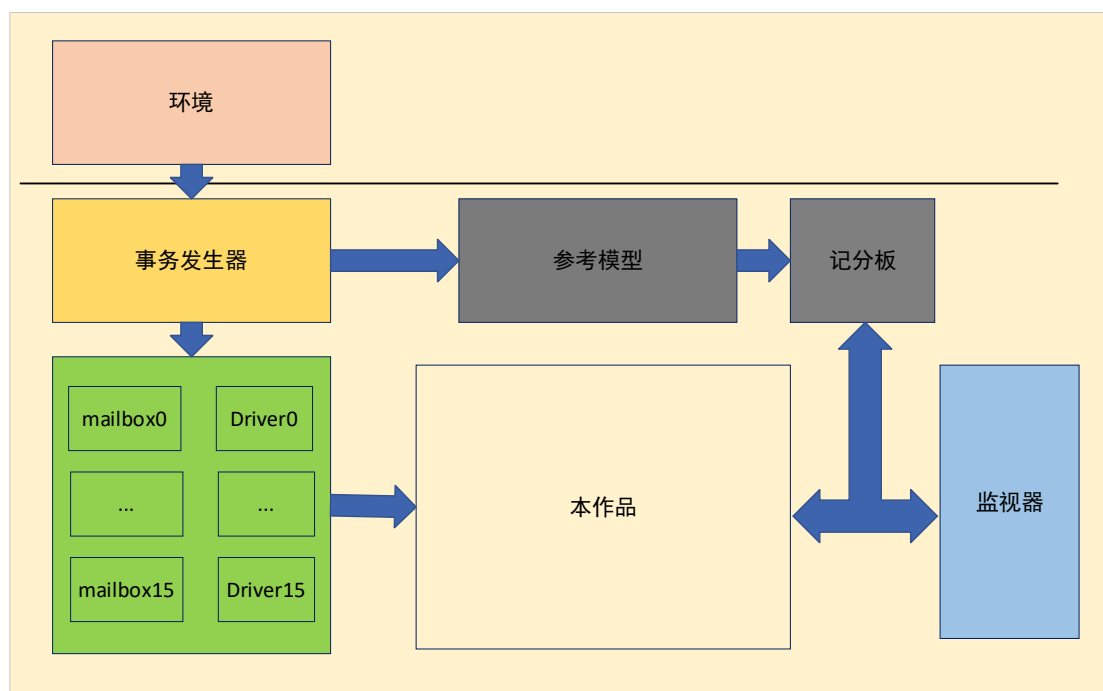


图 3.1 验证平台框图

上图的验证平台中，包含了验证环境，事务发生器，驱动器，监视器，参考模型和记分板模块。图中灰色部分的参考模型和记分板暂时还没有实现。

在环境模块中，对随机化的部分参数进行了设置，并串联起整个验证流程。在事务发生器中，随机化的生成各个数据包数据，并分别放进 16 个和输入端口对应的信箱 mailbox 中。在驱动器中启动了 16 个线程，每个线程和一个输入端口进行绑定，不断从对应的信箱 mailbox 中取数据，然后发送到对应的端口。监视器类对输出端口的数据进行简单的统计和输出展示。

## 四、 仿真测试

### 4.1 DPRAM 模块仿真测试

对 DPRAM 模块的读写进行测试，测试波形图如下图 4.1 所示，测试输出如下图 4.2 所示。

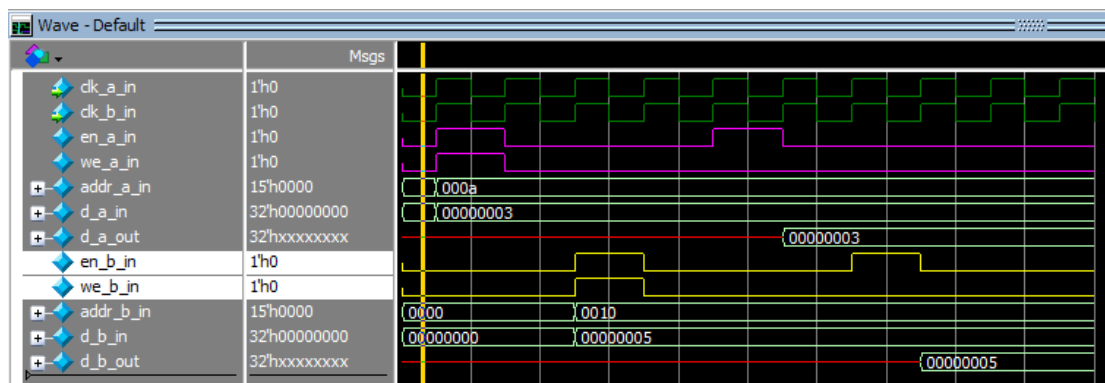


图 4.1 DPRAM 模块仿真波形

```

VSIM 10> run
# @5: PortA write at 000a with value      3 !
# @25: PortB write at 0010 with value     5 !
# @55: PortA read at 000a with value :    3!
# @75: PortB read at 0010 with value :    5!

```

图 4.2 DPRAM 模块测试输出

对 DPRAM 模块进行读写仿真测试，均可以得到正确的结果，符合预期功能。