京微齐力
HERCULES MICROELECTRONICS

HME-P1

DDR2/3 控制器

参考设计指南

2022 年 04 月

京微齐力（北京）科技有限公司

注意

手册版本号

HME-P1ANC02-DDRCR

联系我们

如果您在使用我们的产品过程中有任何疑问或问题，请与京微齐力（北京）科技有限公司或者您当地的代理商联系，或发送邮件至：

sales@hercules-micro.com

环境保护

本产品中包含的某些物质可能会对环境或人体健康有害，为避免将有害物质释放到环境中或危害人体健康，建议采用适当的方法回收本产品，以确保大部分材料可正确地重复使用或回收。有关处理或回收的信息，请与当地权威机构联系。

声明

本手册中包含的信息已经仔细检查并认为是完全可靠的。但是，不对手册中可能或潜在的错误负责。京微齐力（北京）科技有限公司保留停止发布或修改手册而不事先通知的权利。为确保获得最新的产品信息，建议用户及时更新手册版本。

本手册介绍的产品并没有被授权用作为生命保障设备或系统中的关键部件。在此使用到的术语有如下定义：1.生命保障设备或系统是满 足以下条件的设备或系统，(a)被通过手术植入人体内或（b)用来保障或维持生命，当按照标签上的使用说明正确使用时，有理由认为其工作的中断将会给使用者带来巨大的伤害。2.所谓关键部件是指生命保障设备或系统中满足以下条件的部件，即有理由认为该部件中断工作将会导致整个生命保障设备或系统中断工作，或者是影响到后者的安全性和有效性。

版本信息

下表列出了本产品手册的历史版本信息。

发布时间	文档版本	修订信息
2019 年 08 月	1.0	初始版本
2022 年 04 月 18 日	HME-P1ANC02-DDRCR	更新版本

目录

关于本手册

本文档描述基于 HME-P1 器件的 DDR2/3 控制器 SDII、AXI 接口实现的存储器读写测试用例。

HME-P1 系列的信息与文档请访问：http://www.hercules-micro.com/content/details74_351.html。

1. P1 系列的 DDR2/3 控制器简介

本章介绍 P1 系列的 DDR2/3 控制器的特性。

1. P1 系列的 DDR2/3 控制器简介

1.1 控制器简介

P1 是京微齐力最新推出的面向通信、工业、视频处理等中高端应用的中等容量 FPGA 产品，其内置硬件 DDR2/3 存储器控制器，无需消耗珍贵的 FPGA 逻辑及片上存储器资源即可轻松实现大容量高带宽的外部 DDR2/3 存储器访问。

为缩短开发时间、提高开发效率，京微齐力针对该控制器的 SDII、AXI 接口提供了包含存储器读、写及数据比对功能的参考设计，并提供完整工程，该工程可作为进一步设计的基础和出发点。

1.2 特性概要

- DDR2：400-1066 Mb/s，DDR3：800-1333 Mb/s
- DDR 存储器与控制器时钟比例为 2:1
- 支持 DDR BIST 通过或失败标志
- 支持 DDR BIST 是否完成标志
- 支持两个通道 AXI 总线独立访问存储器操作
- 支持两个通道 SDII 总线独立访问存储器操作，AXI 与 SDII 总线是互斥的
- ECC 功能 (32 位数据 + 8 位 ECC)

本章介绍 DDR2/3 控制器的系统架构与实现过程。

2. 系统架构与实现

HME P1 器件的 DDR3 控制器以硬核的形式提供，用户可通过 FPGA 侧的用户逻辑与其对外接口进行交互，从而无需考虑 DDR2/3 颗粒的复杂时序及性能优化问题，该 DDR3 控制器架构如图 1 所示：

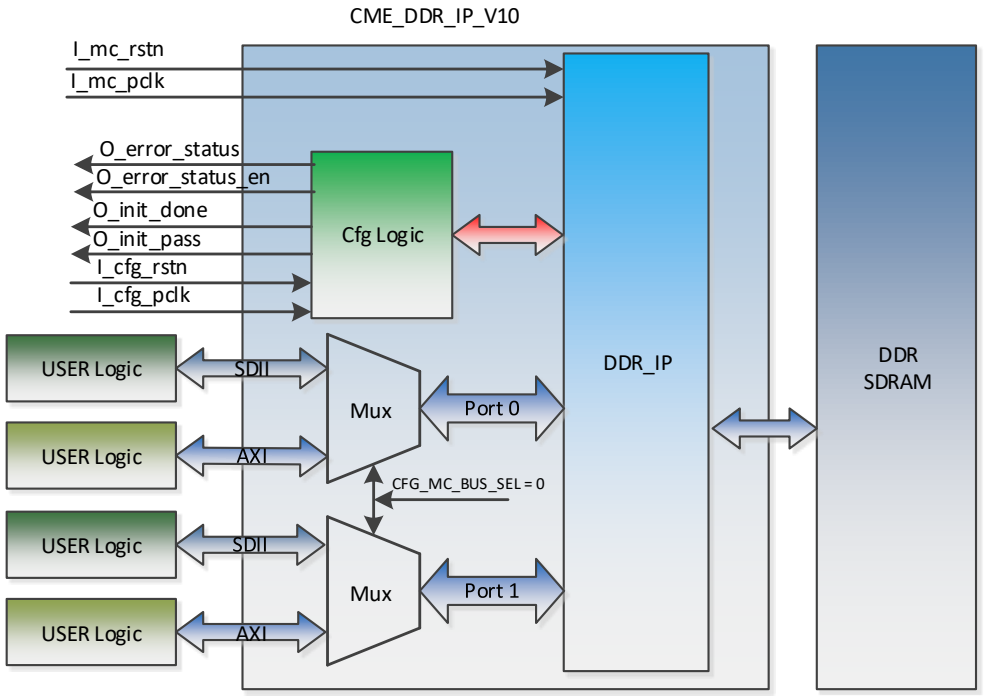

图 1 DDR3 控制器架构

注意：*上图中 AXI 总线和 SDII 总线是互斥的，通过多路选通器与 DDR3 控制器 IP 相连。*

AXI 总线接口符合 ARM AMBA AXI 标准，具体情况可以参阅 ARM AMBA AX 规范文档；而 SDII 是私有总线协议，更加强调接口的简洁明了，可以在保证实现高访问带宽的前提下尽量节省资源消耗，SDII 接口信号详细描述见表 1。

表 1 SDII 接口信号

名称	位宽	方向	描述
sdii_port00_req	1	输入	命令请求信号
sdii_port00_gnt	1	输出	命令应答信号
sdii_port00_rw	1	输入	读写控制信号 0：读命令； 1：写命令；
sdii_port00_addr	32	输入	命令地址
sdii_port00_len	6	输入	突发操作长度 000000：保留

名称	位宽	方向	描述
			000001：1 数据传输 000010：2 数据传输 …. 100000：32 数据传输 其他：保留
sdii_port00_partial_wrt	1	输入	部分写有效控制信号 0：全部写数据有效； 1：部分写数据有效（写屏蔽信号不为零）；
sdii_port00_wdata_pop	1	输出	写数据流量控制信号
sdii_port00_wdata	128	输入	写数据总线信号
sdii_port00_wmask	16	输入	写屏蔽信号总线 0：写数据总线相应字节有效； 1：写数据总线相应字节无效；
sdii_port00_rdata_psh	1	输出	读数据流量控制信号
sdii_port00_rdata	128	输出	读数据总线信号
sdii_port01_req	1	输入	命令请求信号
sdii_port01_gnt	1	输出	命令应答信号
sdii_port01_rw	1	输入	读写控制信号 0：读命令； 1：写命令；
sdii_port01_addr	32	输入	命令地址
sdii_port01_len	6	输入	突发操作长度 000000：保留 000001：1 数据传输 000010：2 数据传输 …. 100000：32 数据传输 其他：保留
sdii_port01_partial_wrt	1	输入	部分写有效控制信号 0：全部写数据有效； 1：部分写数据有效（写屏蔽信号不为零）；
sdii_port01_wdata_pop	1	输出	写数据流量控制信号
sdii_port01_wdata	128	输入	写数据总线信号
sdii_port01_wmask	16	输入	写屏蔽信号总线 0：写数据总线相应字节有效； 1：写数据总线相应字节无效；
sdii_port01_rdata_psh	1	输出	读数据流量控制信号
sdii_port01_rdata	128	输出	读数据总线

SDII 接口支持各种形式的读写事务，其操作时序如图 2、图 3 和图 4 所示：

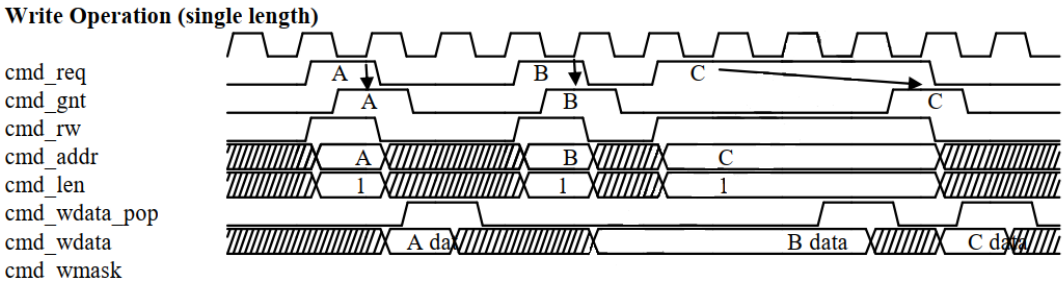

图 2 SDII 单写操作

图 2 为单次写时序，根据 DDR 控制器当时所处的状态，cmd_gnt 信号会在 cmd_req 信号有效的同一个时钟沿或之后的某一个时钟沿由 DDR 控制器拉高进行响应，在此期间 SDII 写操作的发起方应保持 cmd_rw、cmd_addr、cmd_len 信号保持不变，并且直到获得 cmd_gnt 信号响应之后 cmd_req 信号才可以被撤销。

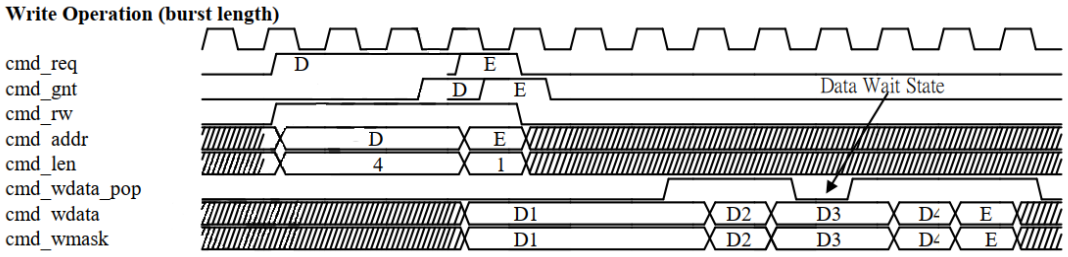

图 3 SDII 突发写操作

突发写操作时序如图 3 所示，上图中展示的是流水线化的两个读操作，前一个是长度为 4 的突发写，紧跟着一个单写操作，DDR 控制器 SDII 接口也支持流水线化的读写、读读等操作。

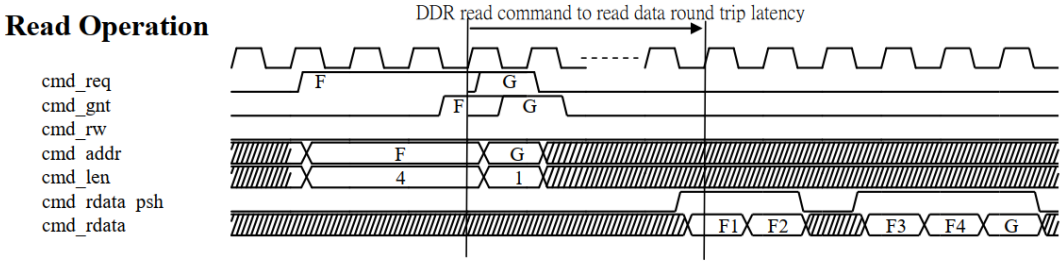

图 4 SDII 读操作

图 4 读操作中，cmd_rdata_psh 拉低表示拉低区间内的 cmd_rdata 无效，同样地，写操作中 cmd_wdata_pop 信号拉低表示 cmd_wdata 无效。

AXI 接口信号详细描述见表 2。

表 2 AXI 接口信号

接口	名称	位宽	方向	描述
Port 0 AXI bus	I_pbus_aclk0	1	输入	时钟信号，所有的 AXI 接口信号都由此时钟上升沿同步采样
	axi_port00_arid	16	输入	读地址 ID

接口	名称	位宽	方向	描述
	axi_port00_araddr	32	输入	读操作地址
	axi_port00_arlen	4	输入	突发读长度
	axi_port00_arsize	3	输入	突发读大小
	axi_port00_arburst	2	输入	突发读类型
	axi_port00_arvalid	1	输入	读地址有效
	axi_port00_arready	1	输出	读地址就绪
	axi_port00_rid	16	输出	读操作 ID
	axi_port00_rdata	128	输出	读通道数据
	axi_port00_rresp	2	输出	读操作响应
	axi_port00_rlast	1	输出	读操作完成信号
	axi_port00_rvalid	1	输出	读操作有效
	axi_port00_rready	1	输入	读操作就绪
	axi_port00_awid	16	输入	写地址 ID
	axi_port00_awaddr	32	输入	写操作地址
	axi_port00_awlen	4	输入	突发写长度
	axi_port00_awsize	3	输入	突发写大小
	axi_port00_awburst	2	输入	突发写类型
	axi_port00_awvalid	1	输入	写地址有效
	axi_port00_awready	1	输出	写地址就绪
	axi_port00_wid	16	输入	写操作 ID
	axi_port00_wdata	128	输入	写通道数据
	axi_port00_wstrb	16	输入	写操作探针
	axi_port00_wlast	1	输入	写操作完成信号
	axi_port00_wvalid	1	输入	写操作有效
	axi_port00_wready	1	输出	写操作就绪
	axi_port00_bid	16	输出	写响应 ID
	axi_port00_bresp	2	输出	写响应
	axi_port00_bvalid	1	输出	写响应有效
	axi_port00_bready	1	输入	写响应就绪
Port 1 AXI bus	l_pbus_aclk1	1	输入	时钟信号，所有的 AXI 接口信号都由此时钟上升沿同步采样
	axi_port01_arid	16	输入	读地址 ID
	axi_port01_araddr	32	输入	读操作地址
	axi_port01_arlen	4	输入	突发读长度
	axi_port01_arsize	3	输入	突发读大小
	axi_port01_arburst	2	输入	突发读类型
	axi_port01_arvalid	1	输入	读地址有效
	axi_port01_arready	1	输出	读地址就绪
	axi_port01_rid	16	输出	读操作 ID
	axi_port01_rdata	128	输出	读通道数据
	axi_port01_rresp	2	输出	读操作响应
	axi_port01_rlast	1	输出	读操作完成信号

接口	名称	位宽	方向	描述
	axi_port01_rvalid	1	输出	读操作有效
	axi_port01_rready	1	输入	读操作就绪
	axi_port01_awid	16	输入	写地址 ID
	axi_port01_awaddr	32	输入	写操作地址
	axi_port01_awlen	4	输入	突发写长度
	axi_port01_awsize	3	输入	突发写大小
	axi_port01_awburst	2	输入	突发写类型
	axi_port01_awvalid	1	输入	写地址有效
	axi_port01_awready	1	输出	写地址就绪
	axi_port01_wid	16	输入	写操作 ID
	axi_port01_wdata	128	输入	写通道数据
	axi_port01_wstrb	16	输入	写操作探针
	axi_port01_wlast	1	输入	写操作完成信号
	axi_port01_wvalid	1	输入	写操作有效
	axi_port01_wready	1	输出	写操作就绪
	axi_port01_bid	16	输出	写响应 ID
	axi_port01_bresp	2	输出	写响应
	axi_port01_bvalid	1	输出	写响应有效
	axi_port01_bready	1	输入	写响应就绪

AXI 总线的地址/控制和数据通道是分离的，共有 5 个独立的通道，支持基于突发的数据传输（burst_based transcation），通过这种连续进行数据传输的方式，多个数据可以被当做一个单元（相当一个数据块）来传送，从而提高了传输效率。图 5、图 6 和图 7 是 AXI 总线操作时序图。

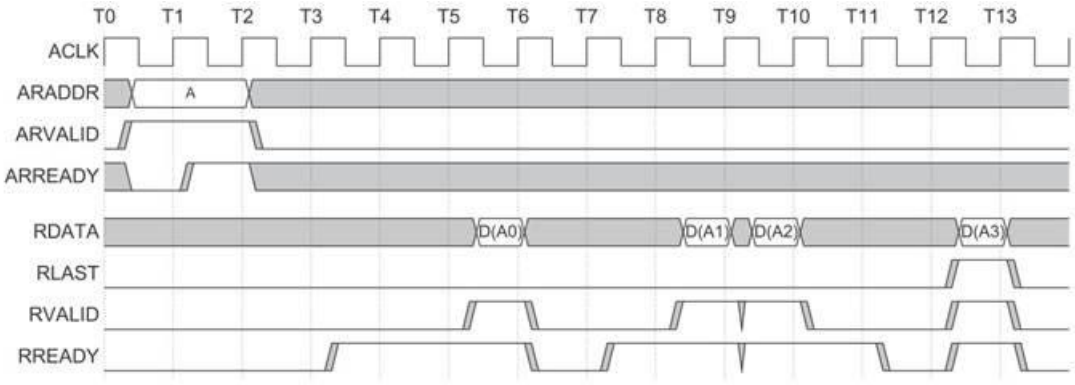

Figure 1-4 Read burst

图 5 突发读操作

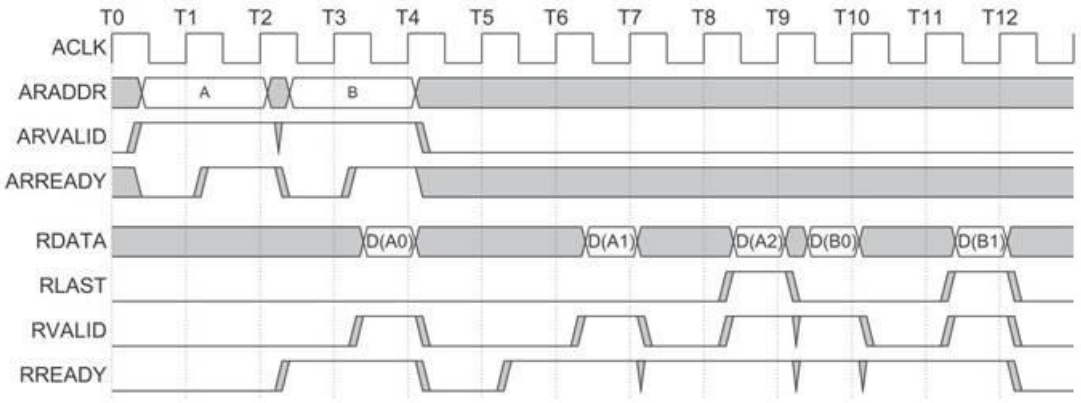

Figure 1-5 Overlapping read bursts

图 6 连续突发读操作

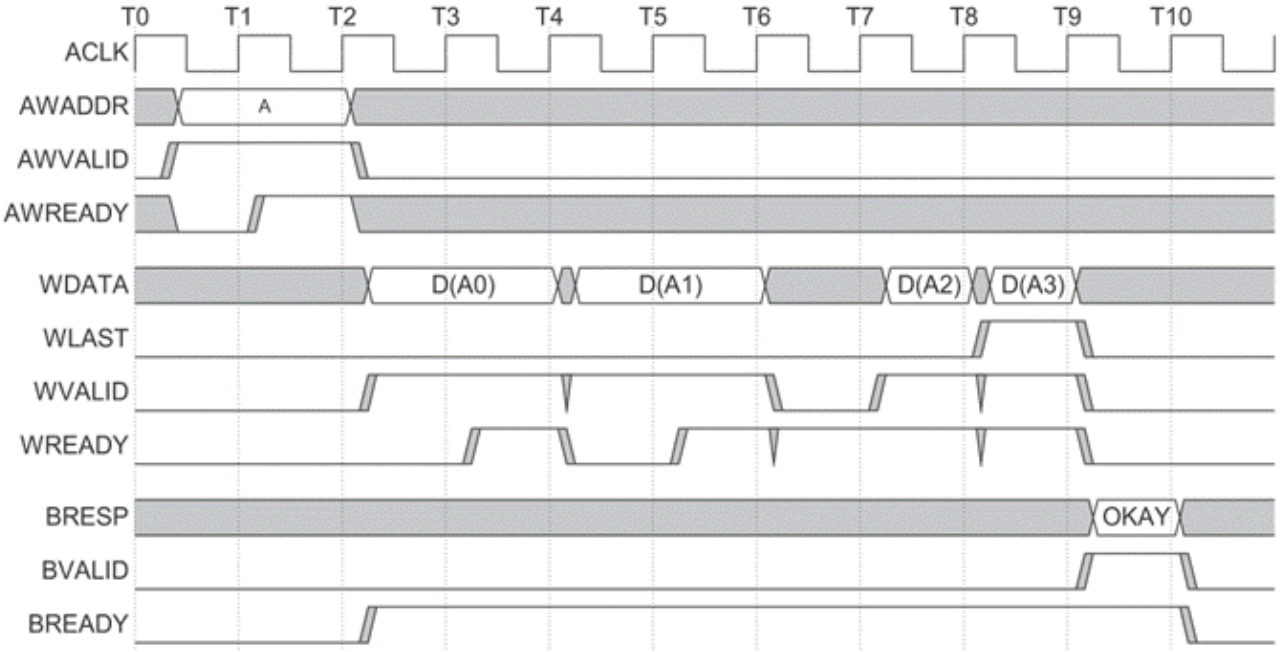

Figure 1-6 Write burst

图 7 突发写操作

基于 SDII 接口的参考设计实现了 DDR3 存储器读写操作及读写数据的比对验证，具体实现架构如图 8 所示。

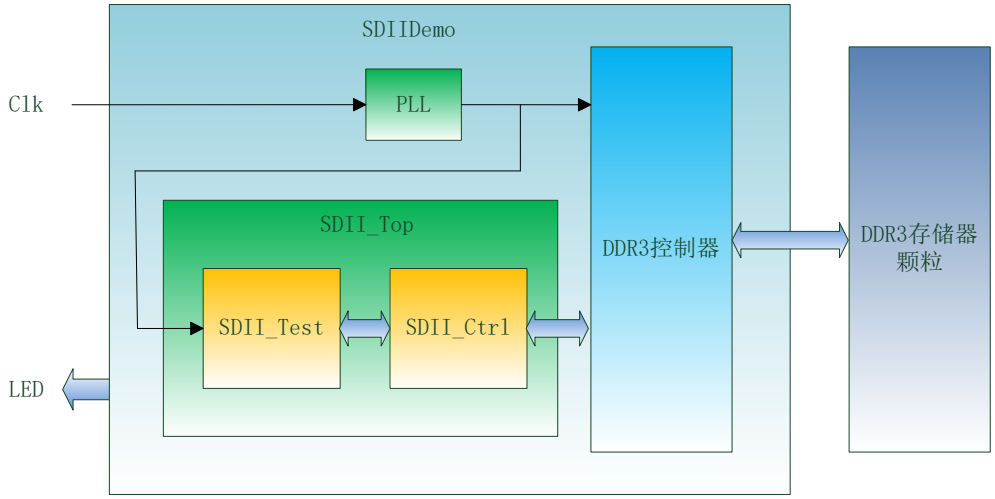

图 8 参考设计整体架构

上图中，DDR3 控制器直接由伏羲软件的 IP 向导管理器生成，设置为 SDII 接口；PLL 输出 200M 主时钟驱动 DDR3 控制器及 SDII 接口操作的顶层逻辑 SDII_Top，SDII_Top 通过 SDII 接口与 DDR3 控制器进行交互，SDII_Top 内部包括两个子模块，分别是 SDII_Test 和 SDII_Ctrl，其中 SDII_Test 实现具体的 DDR3 存储器的地址管理、数据读写、数据比对及状态输出等功能，SDII_Ctrl 实现 SDII_Test 与 SDII 接口的桥接功能。

SDII_Test 是本参考设计中的核心功能模块，该模块采用极简的状态机控制形式，非常简洁，易于理解，可以在此模块的基础上根据具体要求增添用户逻辑实现更为复杂的存储系统。该模块的主状态机各状态定义如下：

```
localparam IDLE          = 3'b000;
localparam WR_REQ        = 3'b001;
localparam SEND          = 3'b011;
localparam SEND_DONE     = 3'b010;
localparam RD_REQ        = 3'b110;
localparam RECIVE        = 3'b111;
localparam RECIEVE_DONE  = 3'b101;
localparam JUDGE         = 3'b100;
```

总体控制逻辑采取先写后读再更新地址的思路，在读数据的同时进行比对，状态机循环遍历完整的 DDR 存储空间逐一进行写读比对，状态转换部分代码如下：

```
IDLE        : S_machine <= ( I_cmd_done ) ? WR_REQ : S_machine ;
WR_REQ      : S_machine <= SEND ;
SEND        : S_machine <= ( I_usr_cmd_gnt ) ? SEND_DONE : S_machine ;
SEND_DONE   : S_machine <= ( I_cmd_done ) ? RD_REQ : S_machine ;
RD_REQ      : S_machine <= RECIVE ;
```

```
RECIVE          : S_machine <= ( I_usr_cmd_gnt ) ? RECIEVE_DONE :   S_machine ;
RECIEVE_DONE    : S_machine <= ( I_cmd_done) ? JUDGE : S_machine ;
JUDGE           : S_machine <= IDLE ;
```

考虑到 SDII 数据总线宽度高达 128 位，读写数据对比部分采取多个 8 位比较器并行以及流水线多级比较的方式以提高这部分的整体性能，具体代码如下：

```
generate
genvar i;
for(i=0;i<16;i=i+1)begin
    always @( posedge I_sys_clk ) begin
      S_cmp_result_0[i] <= ( S_cmp_data_template[ 8*i+7 : 8*i ] == S_cmp_data_src[8*i+7: 8*i ]);
    end
end
endgenerate
```

最终根据比较器的输出，驱动比对失败计数器对出现错误的读写操作进行计数并输出，用户可以将该输出进一步处理货直接驱动片外 LED 进行外部观测。

与基于 SDII 接口的参考设计一样，基于 AXI 总线接口的参考设计实现了 DDR3 存储器读写操作及读写数据的比对验证，采用与 SDII 接口参考设计类似的系统架构，只不过在 IP 向导管理器生成 DDR3 控制器时选择 AXI 接口，为了提高读写数据的复杂度以便得到更多的测试模式，此设计中测试数据作为初始化文件保存在片上的 RAM 块中，通过改变初始化文件可以实现不同的测试模式，读写及比对逻辑思路与 SDII 接口参考设计完全相同，此处不再赘述。

本章介绍 DDR2/3 控制器的开发环境及实现过程，包括资源消耗、时序性能。

3. 开发环境及实现

软件开发环境：Fuxi Version2019.1 Win64

硬件开发环境：HME-P1 EVB Board V1.1

关于上述评估版的详细信息请参考 C1 评估板用户指南。

SDII 接口参考设计资源消耗请见表 3。

表 3 资源消耗

Resource	Usage
Total logic elements	315/36864(0.85%)
Total LUTs(partially or completely used)	315/36864(0.85%)
Total ADD1s(partially or completely used)	112/36864(0.30%)
Total registers	483/74955(0.64%)
I/O pins	9/409(2.20%)
Total PLBs(partially or completely used)	138/4608(2.99%)
Total embedded memory blocks	4/576(0.69%)

SDII 接口参考设计时序性能请见表 4。

表 4 时序分析报告

Clock Name	Type	FMAX	Target Frequency	Slack	Failing Path Count
u0_pll_pll_u0/CO0	Base	231.9MHz	200.0MHz	0.689	0

AXI 接口参考设计资源消耗请见表 5。

表 5 资源消耗

Resource	Usage
Total logic elements	480/36864(1.30%)
Total LUTs(partially or completely used)	480/36864(1.30%)
Total ADD1s(partially or completely used)	202/36864(0.55%)
Total registers	1484/74955(1.98%)
I/O pins	9/409(2.20%)
Total PLBs(partially or completely used)	262/4608(5.69%)
Total embedded memory blocks	33/576(5.73%)

AXI 接口参考设计时序性能请见表 6。

表 6 时序分析报告

Clock Name	Type	FMAX	Target Frequency	Slack	Failing Path Count
u1_pll_pll_u0/CO2	Base	194.6MHz	175.0MHz	0.575	0

综合以上两种接口总线比较可知，P1 器件的 DDR3 硬核控制器性能优良，可以在极大地降低开发难度的前提下为用户提供非常高的存储带宽及存储效率，配合 P1 器件丰富的片上资源可以在很短的时间内帮助用户解决视频、通信、信号处理等各应用领域的大容量高带宽数据存储方面的需求问题。