MC:只负责两件事情：1）下发配置信息； 2）转发请求



状态网 & 配置网

包含一个本地ROM存储器,用于存储配置信息！

这里要注意

配置信息的格式

为设计需要，定义三种模拟簇：数据生成簇、数据计算簇、数据校验簇！其中，数据校验簇又分为PCC校验簇和广播校验簇，区别在于前者通过PCC获得而后者通过广播网获得！

数据生成簇：产生源数据，并写入DDR！

一旦接收到配置信息，立即发送数据写请求，且可以通过配置信息确定写请求的次数，每次写数据量，首个数据的值！

数据计算簇：从DDR中读取源数据，计算后将结果值写回DDR！

数据校验簇：从DDR中读取结果数据，与结果理论值作对比！

**WRITE\_CLUSTER CFG\_INFO**

* CFG0

|  |  |
| --- | --- |
| 31：22 | 21：0 |
| Write\_cycle\_times | Write\_data\_size |

* CFG1

|  |
| --- |
| 31：0 |
| Write\_data\_head |

* CFG2

|  |
| --- |
| 31：0 |
| Write\_Jump\_distance |

**CKECK\_SLUATER CFG\_INFO**

* CFG0

|  |  |
| --- | --- |
| 31：29 | 28：0 |
| ~~Check\_cycle\_times~~ | Check\_data\_size |

* CFG1

|  |
| --- |
| 31：0 |
| Check\_data\_head |

|  |  |  |
| --- | --- | --- |
| **配置项** | **信息宽度** | **代表意义** |
| Write\_cycle\_times | 10-bits | 写操作次数  0值代表无循环  X值代表循环X+1次 |
| Write\_data\_size | 22-bits | 单次写数据批量 |
| Write\_data\_head | 32-bits | 首次写动作的第一个数 |
| Write\_Jump\_distance | 32-bits | 两次写动作的数据增量 |
| ~~Check\_cycle\_times~~ | ~~3-bits~~ | ~~数据校验次数~~  ~~0值代表无循环~~  ~~X值代表循环X+1次~~ |
| Check\_data\_size | 29-bits | 单次校验数据批量 |
| Check\_data\_head | 32-bits | 首次校验数据的第一个数 |

表 1 各个簇与四层网络的链接状态

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Config\_net** | **Status\_net** | **PCC\_net** | **Broadcast\_net** |
| **WRITE\_CLUSTER** | **√** | **√** | **√** | **╳** |
| **ALU\_P\_CLUSTER** | **√** | **√** | **√** | **╳** |
| **ALU\_B\_CLUSTER** | **√** | **√** | **╳** | **√** |
| **CHECK\_P\_CLUSTER** | **√** | **╳** | **√** | **╳** |
| **CHECK\_B\_CLUSTER** | **√** | **╳** | **╳** | **√** |

基本测试：

测试过程中的MC功能极度简化，只负责配置信息的一次性下发、请求信息的转发（滤除通道释放状态信息）！

系统时钟频率：100M Hz

1. 普通读写

1个写 + 1个读 + 1个进程控制中心 + 1个MC(仿真通过，下板通过)

1个写 + 4个读 + 1个进程控制中心 + 1个MC(仿真通过，下板通过)

1个写 + 8个读 + 1个进程控制中心 + 1个MC

一次性写入------------------------------------------(仿真通过，下板通过)

分多次写入------------------------------------------(仿真通过，下板通过)

临时补充：检测通道是否正确释放--------(下板通过)

开启循环：是否正常？-------------------------(下板通过)

大数据量是否正常工作？----------------------(下板通过)

3个写 + 8个读 + 1个进程控制中心 + 1个MC

第一种情况：--------------------------------------(下板通过)

**写通道**：横向优先、3个簇、循环3次、一级3行4列、二级1k行1k列、起始地址为0、数据块大小为1M-64bits

**读通道**：横向优先、8个簇、无 循 环、一级2行4列、二级1k行1k列、起始地址为4M、数据块大小为1M-64bits

第二种情况：--------------------------------------(下板通过)

**写通道**：列向优先、3个簇、循环3次、一级3行4列、二级1k行1k列、起始地址为0、数据块大小为1M-64bits

**读通道**：列向优先、8个簇、无 循 环、一级2行4列、二级1k行1k列、起始地址为4M、数据块大小为1M-64bits

*<网络需要：配置网 + 状态网 + PCC网><所有模拟簇的二级坐标固定为0>*

1. 三方

1个写 + 1个三方 + 1个读 + 1个进程控制中心 + 1个MC(下板通过)

功能：

三个写模拟簇、三个三方簇、1个校验模拟簇；

简单对比（只写一次数据）：

情形一：

只有目标节点请求！

情形二：

只有三方节点请求！

情形三：

目标节点和三方节点皆请求！

针对简单对比，设定两个人工控制的输入标志，分别来控制目标节点和三方节点的功能是否开启，可以达到利用一个设计完成对比过程！

*<网络需要：配置网 + 状态网 + PCC网><所有模拟簇的二级坐标固定为0>*

1. 广播

1个写 + 8个广播 + 1个进程控制中心 + 1个MC()