



GE Fanuc Automation

PACSystems™ RX7i ***内存交换模板***

用户手册

GFK-2300A
July 2004

中国工控网
www.chinakong.com
资料中心

在本出版物中使用的 警告、当心和注意标志

警告

在该出版物中，警告标志被用来强调危险电压、电流、温度或其它条件，这些危险条件可能造成设备或与其使用有关的人身伤害。

在不加注意就会造成人身伤害或设备损坏的情况下，使用“警告”标志。

当心

在不当心就可能损坏设备的场合，使用“当心”标志。

注意

“注意”只是引起对理解和操作设备特别重要的信息的注意。

本手册的内容基于该版本出版时可以得到的信息。内容力求精确，但是难以涵盖软硬件所有细节和变更信息，也不能提供涉及安装、运行或维护的所有可能的偶然情况。文中所描述的一些特点并非所有的硬件和软件系统都有。对于本手册资料在今后所做的变更，GE Fanuc Automation 公司没有义务通知本手册的持有者。

GE Fanuc Automation 公司没有任何表示或保证，明确或暗示，也就是，就法律规定而言，对本资料中所包含信息的准确性、完整性和实用性，本公司不承担责任。不保证它的可做商品或适合应用的目的。

下面是 GE Fanuc Automation 北美公司的一些注册商标。

Alarm Master	Genius	PowerTRAC	Series Six
CIMPLICITY	Helpmate	ProLoop	Series Three
CIMPLICITY 90-ADS	Logicmaster	PROMACRO	VersaMax
CIMSTAR	Modelmaster	Series Five	VersaPro
Field Control	Motion Mate	Series 90	VuMaster
GENet	PowerMotion	Series One	Workmaster
		PowerMotion	

引言.....	1-1
特征	1-2
内存交换网板基本操作	1-4
功能兼容性	1-5
相关出版物	1-5
中国工控网 www.chinakong.com 资料中心	
安装配置.....	2-1
用户特征	2-2
角色开关 (仅RMX)	2-3
光学收发器	2-3
节点号	2-3
冗余传输模式操作	2-3
网络内存偏移	2-4
无赖数据包侦测及移除	2-5
物理安装.....	2-6
安装工具.....	2-6
RX7i机架内安装内存交换板	2-6
光纤及连接器	2-7
光纤连接器	2-7
光缆.....	2-7
62.5/125 μ m MMF	2-7
50/125 μ m MMF	2-8
硬件配置.....	2-9
配置内存交换网板.....	2-9
配置参数	2-10
基本操作.....	3-1
上电及初始化	3-1
BUS_ 函数	3-2
网络内存的多重写入	3-3
下载配置的特殊事项.....	3-4
数据传输时间.....	3-4
估算总的传输时间	3-5
高级操作	4-1
VME 总线中断	4-2
中断初始化逻辑	4-4

Contents

中断处理逻辑	4-5
关联逻辑和中断.....	4-5
中断的动态屏蔽.....	4-6
在内存交换网板上	4-6
在CPU上	4-6
网络中断	4-7
发送网络中断.....	4-7
接受网络中断	4-7
初始化	4-8
网络中断服务	4-8
内存奇偶验.....	4-10
受令内存清除.....	4-12
确定环网完整性.....	4-12
内存交换网板说明	A-1
寄存器定义.....	B-1
区域 2: 主控制状态寄存器.....	B-2
本地控制状态寄存器(LCSR)	B-3
本地控制状态寄存义	B-3
本地中断状态寄存器 (LISR)	B-5
本地中断状态寄存器位定义.....	B-5
本地中断允许寄存器(LIER)	B-7
产生网络中断的寄存器	B-8
网络目标数据寄存器(NTD)	B-8
网络目标节点寄存器(NTN)	B-8
网络中断命令寄存器 (NIC)	B-8
接收网络中断寄存器	B-9
中断发送者数据寄存器.....	B-9
中断发送者节点ID寄存器.....	B-9
中断1发送者数据 (ISD1).....	B-9
中断1发送者节点ID (SID1)	B-9
中断2发送者数据(ISD2).....	B-10
中断2发送者节点ID (SID2).....	B-10
中断3发送者数(ISD3).....	B-10
中断3发送者节点ID (SID3).....	B-10
中断4发送者数据(ISD4).....	B-10
中断4发送者节点ID (SID4).....	B-10

Contents

区域3: 辅控制器状态寄存器	B-11
偏移量 440h	B-11
偏移量441h	B-11
偏移量442h	B-12
偏移量445h	B-12
偏移量446h	B-12
区域4: 中断应答寄存器	B-13
中断应答寄存器 (IAKR)	B-13

本手册介绍了PACSystems RX7i系统中以下类型内存交换模板的特征、安装和操作：

订货号	说明
IC698CMX016	RX7i 控制内存交换（CMX）模板
IC698RMX016	RX7i 冗余内存交换（RMX）模板

内存交换模板使用内存映像技术，它允许PLC之间或其它运算设备之间通过高速光纤网络进行确定性的数据共享。确定性网络允许数据在一个预设的时间周期内进行共享。该网络可以由上述产品或其它任何与VMIC 5565系列产品兼容的内存映像产品混合组成。每个设备都作为网络上的一个单独节点。每个内存映像网最多可有256个节点。任何时候只要有一个节点上的数据被改写，其它节点上的数据将会被自动更新。

网络上的每一个节点用光纤连接到一个菊链环上。第一个节点的发送器连接到第二个节点的接受器上，第二个节点上的发送器连接到第三个节点的接受器上，如此类推，直到环网完成，将最后一个节点的发送器接回到第一个节点的接受器上。1-3页上的示意图给出了一个7节点内存映像网的例子。

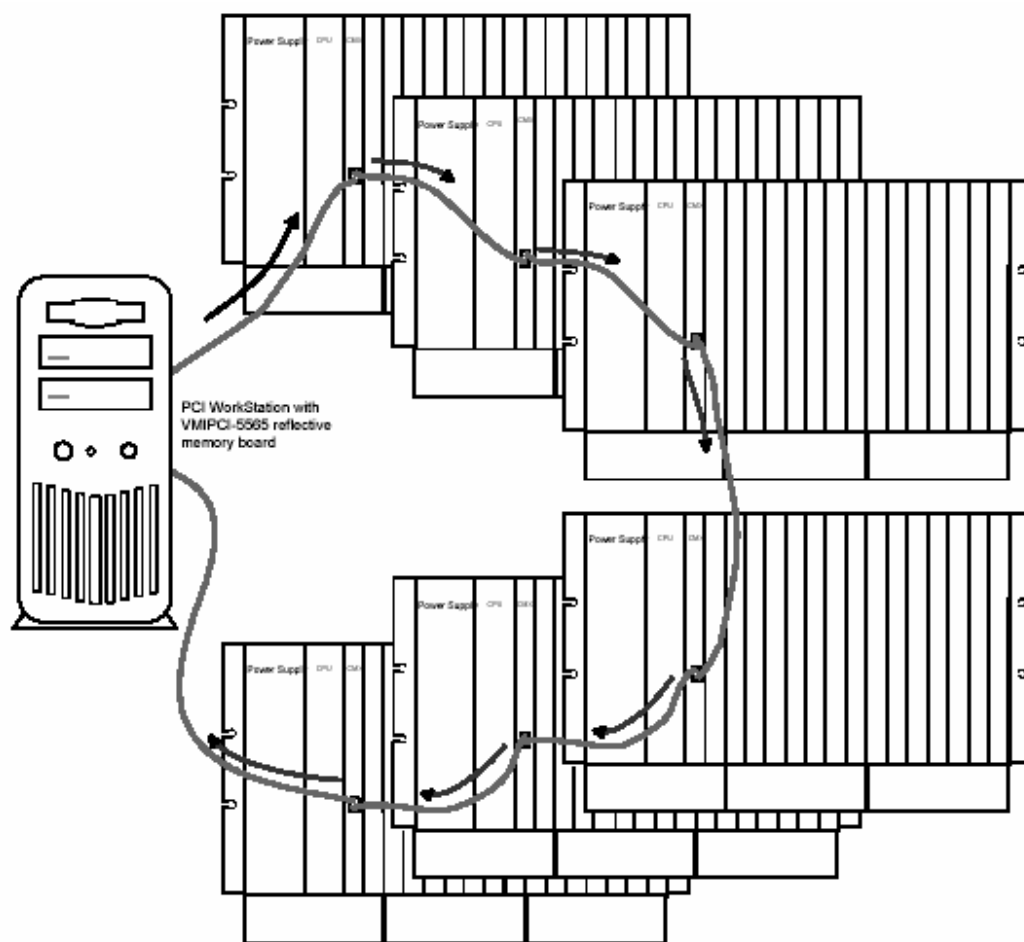
RMX可以作为冗余连接的一部分用于冗余RX7i系统（双CPU）。一个完整的冗余连接包括主单元上的一块RMX和一个从单元上的RMX，以及一条连接它们的高速光纤。它必须是一个两节点环：其它任何内存映像网节点都不能再联到它上面。当RMX不用做冗余连接时，它和CMX的功能相同。

PACSystems热备CPU冗余系统的操作的详细介绍请参考 *PACSystems Hot Standby CPU Redundancy User's Manual, GFK-2308*。

一个PACSystems RX7i主机架最多支持4块内存交换网板，它们可以是RMX和CMX的任意组合。使用冗余CPU时，一个机架内两个以上的RMX可以被配置为冗余连接。

特点

- PACSystems RX7i 单槽模式。
- 16MB带奇偶映像内存。
- 软件配置所有节点参数（没有跳线和开关设置）。
- 高速易用2.12Gb光纤网络。
- 网络操作不需RX7i CPU进行处理。
- 与VMIC 5565系列内存映像网板网络兼容。
- 与多模光纤的连接性能达到30.48m/100ft。
- 动态包长，由网板自动控制。
- 可编程VME总线中断输出。
- 4个通用网络中断，每个32位数据。
- 网络故障监测。
- 每个网络多达256个节点。
- 冗余传送模式操作。这种可选模式减少了数据包丢失的机会。
- 可配置的网络内存偏移允许将网络上的节点根据它们使用的16MB地址空间段的情况化分到不同的组。



七节点内存映像网示例

内存交换基本操作

一旦模板被配置过后，网络的数据传输能够由通过VME总线写内存映像区域开始。网板能够将数据分成4—64 byte的包，通过发送器传递给下一个节点的接收器。只要一个包被接受到，网板将评估该包。然而，如果包不合法或它由本板自身产生的，它将会被丢弃。正在接受的节点将数据写入本地映像内存，并且同时将数据传送到下一个节点。在那里，以上过程将会被重复直到数据回到产生的节点，然后被删除。

基本的操作函数使用Proficy Machine Edition – Logic Developer – PLC配置。

更多的信息，请见以下章节：

- 第2章，“安装配置” — 介绍用户特征，提供物理安装和软件配置的规程
- 第3章，“基本操作” — 介绍基本的内存映像网板的函数以及如何访问它们。
- 第4章，“高级操作” — 介绍如何使用网板的高级功能。
- 附录A，“内存交换网板详细说明” — 提供性能以及环境说明。
- 附录B，“寄存器定义” — 提供模板寄存器的详细定义。

功能兼容性

- PACSystems RX7i CPU硬件版本在2.0或更高。作冗余连接（仅RMX）需要支持冗余的CPU，比如IC698CRE020。
- 编程软件：Proficy Machine Edition Logic Developer – PLC版本4.5或更高。
- 作为通用内存映像网板使用时，它和VMIC5565系列内存映像网板兼容。但是不能和VMIC其它系列的网板同用。

相关出版物

PACSystems CPU Reference Manual, GFK-2222

PACSystems RX7i Installation Manual, GFK-2223

PACSystems RX7i User's Guide to Integration of VME Modules, GFK-2235

PACSystems Hot Standby CPU Redundancy User's Manual, GFK-2308

Proficy Machine Edition Logic Developer-PLC Getting Started, GFK-1918

更多最新的相关文献，请访问GE Fanuc网页 <http://www.gefanuc.com/>

第2章

安装配置

本章提供了内存交换网板用户特征、物理安装和初始化配置的向导。

在使用该网板之前，必须使用**Proficy Machine Edition Logic Developer — PLC**软件对它进行配置。编程软件允许规定网板的基本硬件配置，这允许它在一个内存映像网中或作为一个冗余连接（仅**RMX**）工作。

用户特征

CMX的面板上有4个LED和一个光收发器。RMX有八个LED和一个光收发器。

LED标签	模板	说明
OK	CMX RMX	亮表明模板工作正常
CONFIG LINK OK	CMX RMX	不用作冗余连接时,亮表示模板已经配置。 用作冗余连接时(RMX)亮表示连接工作正常。
LOCAL READY	RMX	亮表示本地单元准备好。
LOCAL ACTIVE	RMX	亮表示本地单元正在工作。
REMOTE READY	RMX	亮表示远程单元准备好。
REMOTE ACTIVE	RMX	亮表示远程单元正在工作。
OWN DATA	CMX RMX	亮表示模板至少从网络上受到过一次它自己发出的数据包。
SIGNAL DETECT	CMX RMX	亮表示接受器侦测到一个光纤信号。



角色开关 (仅RMX)

任务开关是一个弹性复位的双位开关，它一般停在OFF状态，当RMX模板用作冗余连接时，此开关允许手动切换完成从运行的控制器切换到备用的控制器的切换。为了开始角色切换，必须使开关停在ON位置至少1秒。开关状态会自动返回，并且会被过滤（小于1秒），以防止偶发事件。

当RMX模板用作通用内存映像网板的一个节点时（比如不用作冗余连接），角色开关将会不起任何作用。

光学收发器

光学收发器有两个“LC”光纤端口。标为“TX”的端口是发送器，而标为“RX”的是接受器。关于连接光缆的细节，请见“光缆和连接器”（2-7）。

节点号

光纤内存映像网板上的每一个节点必须有一个唯一的节点号，它可以是从0到255。节点号通过软件配置。

冗余传输模式操作

冗余传输模式通过编程软件中的硬件配置功能来打开或关闭。采用冗余传输模式时，每一个包在网络中传输两次。接受节点将判断每个冗余传输。如果在第一次传输时没有侦测到错误，它会用来更新板载内存，第二次传输将会被丢弃。如果第一次侦测到错误，第二次传输用来更新板载内存——如果没有故障的话。如果两次传输都有错误，两次传输都不会被使用，并且数据会从网络上彻底被删除。只要又一次传输出现错误，Bad Data位（Region 2的LCSR的01位）将会被置1。

冗余传输模式极大的减小了网络上数据丢弃的几率。然而，它会使网络传输速率降低50%。

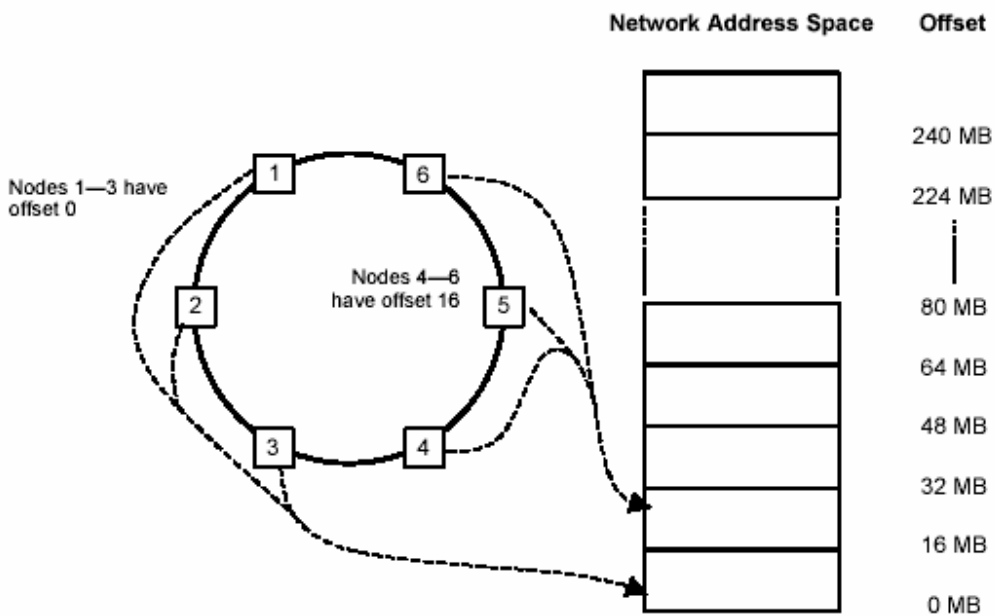
注意：冗余传送模式只适用于网络上数据报传递。它和冗余连接或冗余LED没有任何关系。

网络内存偏移

对于某个特定的节点，一个偏移量可以加到（由本地VME总线写到内存映像网板的）网络数据包的地址上。同样，在本地映像内存使用接受到的数据包时，这个偏移量将会从被减掉。这个偏移量也是通过编程软件来配置。它的范围是0到240MB，步长为16MB。

这个特点可以允许你将网络上的节点分配到以16MB为一段地址空间的组内。在网络上使用同一偏移量地址空间的节点功能上相当于它们在一个自有的网络当中。

下图提供了一个拥有双组的六节点网络的例子。节点1—3使用网络地址空间中首个16MB空间(偏移为0)。节点4—6使用第二个16MB的地址空间(偏移为16MB)。



化分为两段地址空间的节点组网络

无赖数据包侦测及移除

无赖数据包指的是看上去不属于网络上任何节点的数据包。如果数据包在通过非始发节点时发生了改变，始发节点开始不正常工作时，始发节点可能不能识别出它所发出的数据包，这样就不能将该数据包从网络上移除。这种情况下该数据包就会在网络中重复循环。

无赖数据包比较稀少。它们的存在表明因为元件故障或环境过于恶劣而导致某个网板异常。通常的解决办法是隔离更换故障板或改善工作环境。然而，有些情况下只要无赖数据能从网络中被移除，宁可容忍零散的无赖数据，而不愿停止系统来维修。

为了阻止无赖包在网络中不断循环，可以将一块（可有两块）网板配置成两块无赖包管理器。一个无赖包管理器可以改变通过它的每一个数据包。如果一个包第二次回到管理器，它将会被识别并且从网络中被移除。如果有一个无赖包被监测到，无赖包故障标志在本地中断状态寄存器（LISR）中被置位。作为可选，模板能通过编程设置成在出现无赖包时自动生成一个VME总线中断。

每一个内存映像网络支持两个无赖包管理器，管理器0和管理器1，所以它们能够相互检查。

一个网络中的两个板子不能被设置成同样的管理器，否则，当中一个会错误地移除另一个所产生的数据包。

无赖包管理器可以使用编程软件来设置。

物理安装

安装工具

安装前确保准备以下工具：

- 一块PACSystems RX7i 冗余CPU，必须是2.00或更高的硬件版本。
- 一个带电源模板的PACSystems RX7i 机架。
- 编程软件：CIMPLICITY Machine Edition – Logic Developer PLC 版本4.5或以上。
(和一台PC兼容机)
- 光缆。关于连接光缆的细节，请见“光缆和连接器”（2-7）。

注意：带有一块或更多内存映像网板的RX7i系统必须安装在一个符合放辐射标准并且带有有CE标志的金属壳体内。详情见PACSystems RX7i安装手册，GFK-2223的附录A。

在RX7i机架内安装内存交换网板

警告

不要在带电的情况下插入模板，否则会造成CPU停止、模板损坏、或导致人员受伤。

内存交换网板只能安装在RX7i的主机架中，每个主机架最多支持4块内存映像网板。

注意：建议RMX安装在主机架的第3和第4槽。这样可以给RMX模板VME中断请求优先权。但是也可以不必按照建议把RMX安装在3、4槽。

1. 确保机架断电。
2. 按照系统配置情况将模板插入机架相应槽内。
3. 将板子在槽内按牢固，但是不要强按。拧紧面板上下的固定螺丝。
4. 将光纤连接到RX和TX接头上。
5. 将TX光纤接入下一块模板的RX上，然后再重复同样工作，直到最后一块模板的TX接回第一块的RX上，形成环网。
6. 给机架上电。

注意：内存交换网板在得电时处于一个未配置状态，其光学发送器不能工作。直到RX7i CPU将硬件配置发送到模板，它才能在网络中正常工作。更多信息请见“配置”。

光纤及连接器

内存映像产品既可以通过单工（单光纤）也可以通过双工（双光纤）连接。这些线缆必须是多模得。内存交换网板不支持单模光缆。

单工光缆必须是用于2个以上得节点。双工光缆可以用于冗余连接或其它只包含两个节点的网络。

同内存交换网板兼容的带连接器的预制光缆可以使用以下格式订货号来订购：
VMICBL-000-F5-0XX 其中OXX标明长度。

光纤连接器

要求具备以下特点的连接器的。

- LC类，符合IEC 61754-20标准。
- 锆陶瓷箍圈。
- 介入损耗：最大0.35dB。
- 回波损耗：-30 dB。
- 温度范围：-20°C到85°C。

光缆

光缆的结构取决于工作环境。下面描述了两种适合提升器（OFNR）安装的典型光缆。它们的最外面包层直径的推荐尺寸是3.0mm。

62.5/125 μm MMF 光缆

- 多模，带等级指示的玻璃光纤。
- 芯直径：62.5 μm。
- 金属层直径：125μm。
- 缓冲层直径：900μm。
- 衰减率：最大3.75 dB/km。
- 模态带宽：850 nm时160 MHz-km (最小)。
- 工作温度：0到70°C。
- 支持标准：NEC OFNR, CSA FT-4, ICEA S-83-596

50/125 μm MMF Cable

- 多模，带等级指示的玻璃光纤。
- 芯直径：50 μm 。
- 金属层直径：125 μm 。
- 缓冲层直径：900 μm 。
- 衰减率：最大1.5 dB/km，
- 模态带宽：850 nm时160 MHz-km (最小)。
- 工作温度：0到70°C。
- 支持标准：NEC OFNR, CSA FT-4, ICEA S-83-596

硬件配置

在使用内存交换网板前必须使用**CIMPLICITY Machine Edition Logic Developer**—PLC软件对其进行配置。编程软件允许为你的RX7i机架指定一个硬件配置。该硬件配置确定了机架内将会安装的模板，并且配置这些模板的操作参数。

必须下载硬件配置到RX7i的CPU中，以配置内存交换网板。在这些配置运行之前，节点将会保持光学发送器和接受器在无效状态。

对于通用内存映像操作，可以在硬件配置中设置以下参数：节点号，冗余传输模式，无赖数据管理器，网络内存偏移量，以及中断允许。如果冗余连接选择操作，这些参数将会被自动设置，而不需要配置。

配置内存交换网板

使用软件配置RX7i配置的细节，请参考软件的在线手册。以下是配置的主要步骤：

1. 打开RX7i的Tatget下的硬件配置。
2. 右击将要安装模板的槽，选择Add Module。模板目录被打开。
3. 点击Communications标签，选择Memory Xchange module 点击OK。模板就会添加到机架的硬件配置上，并且其参数会在Parameter Editor窗口。
4. 点击想要编辑的参数的标签来编辑参数，然后在点击相应的Value区域。这些区域的细节请见2—10页的“配置参数”。
5. 保存设置并且下载到CPU中，使设置生效。

配置参数

Redundant Link

(仅对RMX，并且仅在一个冗余CPU被配置后才有用)

选择:

- **Disabled:** RMX不是用作冗余连接，而是和通用模板CMX作相同用法时，这个标签上所有的参数都是可配置的。
- **Enable:** RMX用作冗余连接且不能用作通用的内存映像网板。这个标签上所有的参数是不可配的，并且中断参数被设为Disabled。

默认:

- 允许——当Target中有少于两块RMX已经被配置成冗余连接时。
- 不允许——当此Target中已经有两块RMX已经被配置成冗余连接时。

注意: RMX的节点号，冗余传输模式，无赖包管理器，网络内存偏移量只有在它的冗余连接被设置成Disable时才有效。

节点号

内存映像网中识别此节点的唯一号码。

合法范围: 1—255。

默认: 0。

冗余传输模式

决定数据包是传输一次还是传输两次。

注意: 网络上所有的节点必须使用相同的传输模式设定。

选择:

- **Disabled:** 每一个包传输一次。这确保了最大效率的网络传输速率，但是会增加数据丢弃的风险。
- **Enabled:** 每个包传输两次。这很大的降低了数据丢弃的风险，但是会降低级数据的传输速率。更多细节见2—3页。

默认: 不允许。

无赖包管理器

细节见2—5页上的“无赖包侦测及移除”

小心

不要把网络中两个节点配置成同样的无赖包管理器；否则，其中的一个会在数据包送达环网内所有的节点之前错误的将它们删除。

选择：

- **Disabled:** 模板不会侦测到无赖数据包。
- **Rogue Master 0 Enabled:** 内存交换网板设置成无赖包管理器0。
- **Rogue Master 1 Enabled:** 内存交换网板设置成无赖包管理器1。

默认：不允许。

网络内存偏移量（MB）

偏移量被加到由本地VME生成并写入映像内存的网络数据包上。同时，在本地映像内存使用接受来的网络数据包之前，它将会被减去。细节请见2—4页。

合理范围：0—240MB，步长16MB。

默认：0。

中断

（当冗余连接参数设为Enable时此设置为只读）。告诉系统是否要从内存交换网板产生一个VME中断。参数必须设置成Enabled来使中断触发一个逻辑块的执行。

选择：Disabled, Enabled。

默认：不允许。

注意： 本章所描述的功能对用作冗余连接的RMX不适用。当内存交换网板用作冗余连接时，其所有的内存区域都在CPU的控制之下，并且不能被用户逻辑访问。BUS_函数（BUS_RD,BUS_WRT,BUS_TS,和BUS_RMW）访问会失败且状态值会显示为8（区域不允许）。

四个内存区域定义给内存交换网板。

- 区域1：映像内存区
- 区域2：主控制及状态寄存器
- 区域3：辅控制及状态寄存器
- 区域4：中断应答寄存器

区域1和板上整个16MB映像内存相对应。只有使用模板高级功能时才需要访问区域2、3或4。高级功能在第4章描述。

上电及初始化

当内存交换网板第一次得电时，以下工作将会进行：

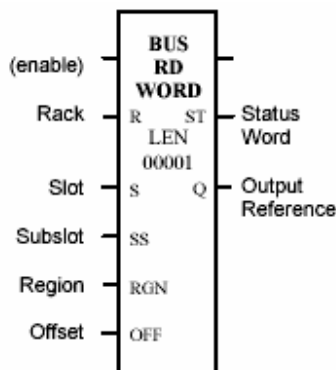
1. 进行一个回送测试。OWN DATA和SIGNAL DETECT LED在测试期间将会打开。
2. 区域1内的内存全部置0。
3. OK LED打开。
4. 如果模板被配置：
 - A. 模板的网络发送器允许。
 - B. CONFIG LED打开。
 - C. 一个测试包会被发出用来确定是否环网内所有节点的发送器都被允许了。LCSR内的OWN DATA状态（0位）指示环网是否完好无损。

BUS_ 函数

以下四个函数允许PAC CPU中的应用程序和机架内安装的内存交换网板（可以是多块）进行通讯：

- 总线读 (BUS_RD)
- 总线写 (BUS_WRT)
- 总线读-修改-写 (BUS_RMW)
- 总线测试及设置 (BUS_TS)

以上所有的函数使用相同的参数设置来指定访问那个网板以及该网板的那个区域。下图展示总线读功能块的参数：



机架号和槽号指明要访问哪一块内存交换网板。子槽号应该保持空白或置0。区域号指的是内存映像网板中四个内存区中的某一个。偏移量是一个从0开始的数字，它指定访问内存区域中哪一部分。

注意：如果不指定一个区域，默认为1，它将对应映射内存区（区域1）。

细节参考PACSystems CPU 参考手册,GFK-2222。

注意：对内存映像网板的读-修改-写（RMW）访问（通过BUS_RMW或BUS_TS函数）不能由内存交换网板本身来保证其对这些地址访问的精确性；它们仅对其它VME访问保证精确性。其它总线控制器不允许在Read和Write操作之间状态向模板写数据。然而，从内存映像网上或从CMX/RMX模板本身写是不被禁止的。所以（RMW的）VME写能覆盖和丢弃写到相同地址上的来自网络或模板写入的任何数据。

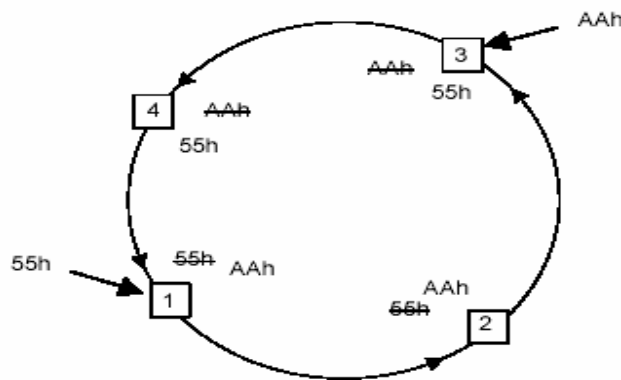


网络内存的多重写入

应用程序两个和两个以上的节点不会同时向相同的内存映像网地址上写数。如果同时写，不同节点上映像内存内的数据值可能会不相同。

例如，按照以下方式连接的包括节点1、2、3和4四个节点的网络。如果在几乎同一时刻内节点1写55h而节点3写AAh到网络地址00h，将会出现以下情况：

- 在节点2上，内存地址00h变成55h，但是然后快速变成AAh。
 - 在节点4上，内存地址00h变成AAh，但是然后快速变成55h。
- 结果就造成了节点2和4在相同的地址00h上拥有不同的值。



多重写到相同RFM网络地址示例

如果应用程序需要不同的节点向同一地址写数，可以使用以下方法来避免在同一时间内写到内存地址上。

- 为所有的节点指定一系列的规则，它们每次只允许一个节点向某个地址上写入。例如，考虑最简单的两节点共享一个内存地址的情形。系列规则可以如下：
 - 在某个共享的内存地址内已经有了值1，并且只能向其中写入值2时，节点1才可以向这个地址上写数。
 - 在某个共享的内存地址内已经有了值2，并且只能向其中写入值1时，节点2才可以向这个地址上写数。
- 当某个内存地址允许写入，使用网络中断来标识。例如：
 - 节点1最初有写到地址0的许可。
 - 当节点2想写到地址0，它发出一个网络中断到节点1来请求许可。
 - 节点1收到中断后，如果有必要它结束自己向地址0写数，然后发一个网络中断到节点2，允许节点2向地址0写数。

下载配置的特殊事项

如果包含内存交换网板改变的硬件配置被下载，先前对该模板寄存器作出的任何逻辑上改变将会被重写。伴随其它的事件，以下的事件也将发生：

- 模板暂时从网络中断开。
- CPU清除网络中断FIFO队列。
- CPU清除LIER。
- CPU清除LISR。
- 模板LCSR（位于区域2中偏移地址为08h）中Latch Sync Loss位（第3位）被置1。
- 模板重新接入网络。
- 一个测试包发出，以确定是否网内所有的节点都已经接入并且发送器都允许工作。LCSR中OWN DATA状态（第0位）将指示环网是否完好。
- CONFIG LED打开。

如果一个硬件配置下载下去，但是以前已经配置并工作的内存交换网板部分的配置没有改变的话，该模板将保持网络连接，并且继续接受来自内存映像网络内其它网板内存写入等。

你可以编制应用程序在第一次扫描时读取LISR中Latched Sync Loss（第11位）的状态，以确定模板是否从网络中断开。如果模板被断开，网络中所有节点的映像内存上的数值都需要被刷新。为了刷新这些值，一个或更多的节点应该重写需要的数值到映像内存上，比如，区域1。

如果一次下载改变了模板内存的偏移地址，下载后该模板上映像内存里的内容将会失去其原先定义。应该刷新这些位于区域1内的数值。为了刷新这些值，一个或更多的节点应该重写需要的数值到映像内存上。

数据传输时间

通过内存交换网板从一个PLC向另一个PLC传输数据的时间很大程度取决于数据交换算法。然而，内存交换网板与PACSystems CPU之间数据读写的时间量将会在以下章节中描述：

以下表格提供了估算PACSystems CPU和CMX模板或RMX模板（不是用作冗余连接）之间读/写时间的公式。实际的读/写时间可能同估算的时间有些轻微出入，并且大部分系统的实际表现会比估算的要好。估算读/写时间基于以下条件：CPU无故障允许，CPU无串行通讯，无Genius总线故障，无背板高中断率。另外，计时是基于单个BUS_RD和BUS_WRT函数功能块，且数据长度在256到131,068字节之间。

CPE020 和CRE020 CPU到内存交换网板读/写传输时间

CPU写入内存交换网板的时间(ms)	$(0.00008079 * (\text{字节数})) + 0.25$
CPU读取内存交换网板的时间(ms)	$(0.00006853 * (\text{字节数})) + 0.22$

CPE010 CPU到内存交换网板读/写传输时间

CPU写入内存交换网板的时间(ms)	$(0.00009831 * (\text{字节数})) + 0.644$
CPU读取内存交换网板的时间(ms)	$(0.00008468 * (\text{字节数})) + 0.631$

估算总传输时间

对于大部分系统光纤网络单数据传输时间要远小于读/写时间。所以，使用一个单独的BUS_WRT或BUS_RD功能块传输一个数据包的总时间可以通过简单的求一定数量数据在CPU和内存交换网板之间读与写的和来粗略的来估计，同时还要把CPU的异步延时考虑进去。

例如，如果传输1024字节，并用最后一个DWORD（4字节）来标识Data Valid，而且接收CPU扫描时间为5毫秒。你可以按照以下步骤来估算数据从一个CPU传输到另一个CPU（注意：本例使用CPE020/CRE020计时值）：

1. 估算写时间：

$$\text{WriteTime} = (0.00008079 * (1024)) + 0.25 = 0.34 \text{ ms}$$

2. 估算读时间

$$\text{ReadTime} = (0.00006853 * (1024)) + 0.22 = 0.29 \text{ ms}$$

3. 估算CPU异步延时。对于大部分应用程序这是一个CPU扫描周期，因为典型的应用程序每周期会检查一次Data Valid Dword。然而，如果应用程序使多CPU同步优先于数据传输，这个延时时间会大大的减少。

$$\text{AsyncDelayTime} = \text{Receiving CPU Scan time} = 5 \text{ ms}$$

4. 估算总传输时间：

$$\begin{aligned} \text{TotalTime} &= \text{WriteTime} + \text{AsyncDelayTime} + \text{ReadTime} \\ &= 0.34\text{ms} + 5\text{ms} + 0.29\text{ms} = 5.63 \text{ ms} \end{aligned}$$

你的算法也许或多或少比这个例子有效。对于大量的数据传输，你可以通过将传输的数据分成更小的包来优化数据传输，这样，在一块CPU在读数的同时，另一块CPU则正在写

注意：本章所描述的功能不能用在一块作为冗余连接的RMX上。当内存交换网板用作CPU冗余系统之间通讯连接时，这些内存区域处于CPU的控制之下，不能被用户逻辑访问。BUS_×××访问将会失败，状态值为8（区域不允许）。

本章叙述如何使用内存交换网板的高级功能。这些功能通过区域2，3和4来访问。附录B提供了这些区域的详细定义。高级功能如下：

- VME总线中断处理
- 网络中断处理
- 内存奇偶校验
- 受令内存清除
- 环网完整性检查

VME 总线中断

内存交换网板有一个单独的可编程VME总线中断输出，它可以用来触发执行一个逻辑块。内存交换网板能对不同的事件产生中断，比如奇偶错误，信号监测错误，或无赖数据包故障。完整的列表见4-3页“VME总线中断事件”。

如果你的应用程序使用VME总线中断，它必须向模板写数据来选择中断源，并且在它们发生时作出响应。为了控制哪个事件会产生一个中断，需要向本地中断允许寄存器（LIER-区域2，偏移14h）写数据。一旦中断产生，逻辑必须读取本地中断允许寄存器（LIER-区域2，偏移10h）来确定中断产生的原因(可能不是一个)并且作出适当的响应。

注意：如果CPU在停止模式下有VME中断产生，相关的中断块不会因为中断的产生而执行。

以下是使用VME总线中断过程中包含的一些步骤的总结：

1. 决定你想为哪个事件产生一个中断。（见“VME总线中断事件”4-3页）
2. 开发中断初始化逻辑。
3. 开发中断处理逻辑块。
4. 将模板硬件配置中中断参数设为允许。
5. 使中断和逻辑块相关联。
6. 下载并测试应用程序。

VME总线中断事件

内存交换网板可以为以下事件的任意组合产生一个VME总线中断。使用LIER来选择哪个事件会产生一个中断。另外，LISR的中断允许（第14位）必须置1。这些寄存器的详细定义请参考附录B。

事件	描述
网络中断1	收到一个类型1网络中断
网络中断2	收到一个类型2网络中断
网络中断3	收到一个类型3网络中断
网络中断4	收到一个类型4网络中断
同步丢失	光纤接受器丢掉了收到的信号。数据可能已经过早的从网络上被移除。很可能的原因包括：模板的配置被改变，接受器光缆断开，或上游节点发送器失效（如掉电，配置改变，或有意使其失效）。
坏数据	接受器回路监测到一个不合法数据包。数据可能过早从网络上移除。
无赖数据包故障	模板监测并移除一个无赖包。
内存奇偶错误	监测到一个奇偶错误
RX FIFO几乎满	接受器FIFO已经几乎满了。这个事件表明接受回路在满负荷运行，这在正常条件下运行使不该发生的。如果它发生，应用程序应该暂时停止所有对模板的访问。
RX FIFO满	接受器FIFO已经满了。这表明内存映像网板进行了不恰当的操作。数据也许已经过早的从网络上被移除。
内存写丢弃	奇偶校验允许时，因为违反了队列或长度的限制一次内存写被丢弃。

中断初始化逻辑

为了给内存交换网板编程产生一个VME总线中断，强烈建议你的逻辑使用下列方法。

第1步.执行一次BUS_RMW_WORD函数将0(“AND”)赋给操作参数，将FEDFh赋给掩码参数，将4赋给区域号，68h给偏移地址(IAKR)。这个操作将0写到8位的IAKR内，它用来确定过去在CPU停止模式下所发生的中断。同时，当IAKR的第7位是1时，这个操作将1写回那一位，这用来清除锁存器。用FEDFh掩蔽当前值会阻止这个寄存器中其它位改变。

第2步.执行一次BUS_RWT_DWORD函数写一个位模式到本地中断允许寄存器(LIER-区域2，偏移14h)。输入参数取决于你想让模板为之产生中断的事件组合。例如，编程为模板产生一个响应4—3页上每一个事件的中断，写00003FC7h到LIER。

第3步.执行一次BUS_RMW_WORD函数，操作数赋1(“OR”)，掩码参数赋4000h，区域赋2，偏移赋10h(LISR)。这个操作将1写到本地中断状态寄存器的第14位，它用来为模板产生一个中断。使用一个读—修改—写来保存LISR中的其它位。

第4步.执行一次BUS_RMW_DWORD函数，操作数赋0(“AND”)，掩码参数赋FFFC038h。区域号赋2，偏移赋10h(LISR)。你可以随意将一个变量地址赋到原始值输出参数上(如%T0001)(用来接收原始值)。这个操作检索LISR中所有的值并且将0写回所有的锁存位来清除它们。

第5步.可选。测试本地LISR拷贝内每一个非网络中断状态位。如，如果第11位(如T0001)值为1，在第4步逻辑运行之前一个同步丢失曾发生。

第6步.可选：针对LISR中可能的四个网络中断中的每一个(第0，1，2和7位)。(另外，你的逻辑可以按照“接受网络中断”(4—7页)中描述的方法清除网络中断FIFO队列)

a) 测试本地LISR拷贝中的网络位(如，测试%T0001看有没有类型1网络中断)，如果该位为1，读相应的网络中断FIFO。

b) 执行一次BUS_READ_DWORD函数，再次读LISR。在第4步后不要再写任何值到LISR。然后重复步骤6。

第7步.执行一次BUS_RMW_BYTE函数，操作数赋1(“OR”)，掩码参数赋1，区域号赋4，偏移赋69h(IAKR)。这个操作将1写到IAKR的第8位，这允许模板产生一个VME总线中断。与1进行“OR”操作防止这个寄存器中其它位被改变。如果在此时，LISR中有一位是1并且它相应的LIER位也是1，模板立刻产生一个VME总线中断。在这种状态下，CPU将安排执行你的中断块。

中断处理逻辑

在开发一个处理内存交换网板VME总线中断块逻辑时，强烈建议您按照以下方法来进行：

- 第1步.执行一次BUS_RMW_WORD函数，操作数赋0(“AND”)，掩码参数赋FEDFh。区域号赋4，偏移赋68h (IAKR)。本操作写0到中断应答寄存器 (IAKR)，它用来应答VME总线中断。另外，IAKR的第7位是1时，本操作将回写1到该位，以清除该位的锁存器。用FEDFh掩蔽当前值会阻止这个寄存器中其它位改变。
- 第2步.执行一次BUS_RMW_DWORD函数，操作数赋0(“AND”)，掩码参数赋FFFFC038h。区域号赋2，偏移赋10h (LISR)。将一个变量地址赋到原始值输出参数上 (如%T0001) (用来接收原始值)。本操作获取本地中断状态寄存器 (LISR) 的当前值并且将0写回所有的锁存位以清除它们。
- 第3步.测试LISR本地拷贝中每一个非网络中断状态位。如，如果第11位 (如T0011) 值为1，一个同步丢失曾发生。
- 第4步. 对于LISR中可能的四个网络中断位中的任何一个 (第0, 1, 2和7位)。
- a) 测试本地LISR拷贝中的网络位 (如，测试%T0001看有没有类型1网络中断)，如果该位为1，读相应的网络中断FIFO。
 - b) 执行一次BUS_READ_DWORD函数，再次读LISR。在第2步后不要再写任何值到LISR。然后重复步骤4。
- 第4步.执行一次BUS_RMW_BYTE函数，操作数赋1 (“OR”)，掩码参数赋1，区域号赋4，偏移赋69h (IAKR)。这个操作将1写到IAKR的第8位，这允许模板产生一个VME总线中断。与1进行“OR”操作防止这个寄存器中其它位被改变。如果在此时，LISR中有一位是1并且它相应的LIER位也是1，模板立刻产生一个VME总线中断。在这种状态下，CPU将安排执行你的中断块。

关联中断和逻辑

使用内存交换网板中断来触发逻辑执行时，必须规定中断与要执行的逻辑块之间的联系。在你能产生这个联系之前，产生中断的模板在硬件配置中必须配置完毕，并且，中断 (Interrupt) 差数必须设置成允许(Enable)。在CIMPLICITY Machine Edition中，打开中断逻辑的属性 (properties)，并且展开编排 (Scheduling) 属性。在编排 (Scheduling) 对话框中，其中，类别 (Type)，选模板中断 (Module Interrupt)。触发 (Trigger)，使用下拉菜单选择一个模板中断。模板使用r.s (#i)来标识，其中 r表示机架，s表示槽号，#i表示中断号。使用中断# 1。

中断的动态屏蔽

内存交换网板上

在运行期间，通过向LISR中的中断允许位（第14位）写数，应用程序能够对来自内存交换网的VME总线中断进行屏蔽或解除屏蔽。

CPU上

在运行期间，使用SVC_REQ函数功能块17，应用程序能够对与内存交换网板相联系的中断块的执行进行屏蔽或解除屏蔽。使用SVC_REQ 17时，内存类型（memory type）赋十进制17，偏移（offset）赋VME中断的id。想获得模板的VME中断ID号，请见PACSystems RX7i User's Guide to Integration of VME Modules, GFK-2235。

如果中断块的执行没有被屏蔽，CPU会处理VME总线中断，并且安排执行和中断相联系的逻辑块。当中断块的执行被屏蔽，CPU会处理中断，但是不会执行与之联系的逻辑块（如，CPU会丢弃中断）。

在CPU从停止（Stop）变到允许（Run）时，中断块的执行将会被解除屏蔽。

使用SVC_REQ #17的细节，请参考PACSystems CPU Reference Manual, GFK-2222。

网络中断

网络上任何一个节点都可以向网络上某特定的节点或广播式的向网络中每一个节点发送一个网络中断包。每一个网络中断包都包含了发送者的节点ID号，目标节点ID号，中断类型信息，以及32位的用户定义数据。总共有4类网络中断，它们都由用户定义。

发送网络中断

为了开始一个网络中断，你的应用逻辑必须向区域2中3个寄存器上写数：

1. 向网络目标数据（NTD）寄存器中写入32位用户数据。
2. 向网络目标节点（NTN）寄存器中写入目标节点ID。
3. 向网络中断命令（NIC）寄存器中写入中断类型信息。

本步骤必须最后执行，因为它实际产生了网络中断包。想确定写到本寄存器中的数值，请参考附录B。

第2、3步能够使用BUS_WRT_WORD函数来完成。

接收网络中断

节点每次产生一个网络中断时，它包含了该节点本身的节点ID作为包的一部分。当内存交换网板接受到发给它的网络中断时，它会将发送者的节点ID和32位的用户定义数据存入一个FIFO（先入先出）队列。每个网络中断类型（1—4）都有一个独立的FIFO。每个FIFO能存储127个中断。

任何时候，只要FIFO中至少有一个中断在等待状态，LISR中相应的位将会被置1。作为可选，通过编程能使模板在它一接受到网络中断就会产生一个VME总线中断。

为了获取FIFO中等待最长的中断，应用逻辑应该做以下工作。

1. 读相应的中断发送者数据（ISD×）寄存器。
2. 读相应的中断发送者节点ID（SID×）寄存器。

当SID×寄存器被读取，整个中断输入（节点ID和用户数据）会从FIFO中被移除。所以，每个发送节点ID只能被读取一次。如果需要发送者数据，在读取中断发送者节点ID（SID×）寄存器之前读取相应的发送者数据（ISD×）寄存器。

初始化

如果应用程序要使用一个中断块来作中断服务，你应该按照4—4页上的“中断初始化逻辑”所描述的步骤来作。

如果应用程序需要丢弃任何不需要服务的网络中断，通过写一个字节的0到区域2的以下偏移地址上来清除各个中断发送者ID（SID）FIFO。

发送者ID FIFO	偏移地址
SID1	24h
SID2	2Ch
SID3	34h
SID4	3Ch

网络中断服务

以下规则给出了服务所有4个网络中断FIFO的必要步骤。这个规则中所访问的所以寄存器都在区域2里。

注意：如果应用程序使用一个中断块监测到一个网络中断被接受，你应该将此规则包含到4—5页描述的“中断处理逻辑”规则当中去。

第1步.获得LISR的值。例如，执行一次BUS_READ_DWORD函数，选择区域2和偏移地址10h。将一个变量地址给输出参数上（如%T0001）。千万不要往LISR中写回任何数。

第2步.如果LISR中四个网络中断位（第0，1，2和7位）为0的话，说明没有任何中断在等待，你可以退出这个过程。

第3步.测试LISR的第0位（如，%T0001）。如果该位是1，则有一个类型1网络中断在等待：

- a. 读中断1发送者数据（ISD1）寄存器（偏移量20h）来获得32位数据。
- b. 读中断1发送者节点ID（SID1）寄存器（偏移量24h）来获得发送者节点ID。

第4步.测试LISR的第1位（如，%T0002）。如果该位是1，则有一个类型2网络中断在等待：

- a. 读中断2发送者数据（ISD2）寄存器（偏移量28h）来获得32位数据。
- b. 读中断2发送者节点ID（SID2）寄存器（偏移量2Ch）来获得发送者节点ID。

第5步.测试LISR的第2位（如，%T0003）。如果该位是1，则有一个类型3网络中断在等待：

- a. 读中断3发送者数据（ISD3）寄存器（偏移量30h）来获得32位数据。
- b. 读中断3发送者节点ID（SID3）寄存器（偏移量34h）来获得发送者节点ID。



第5步.测试LISR的第7位（如，%T0008）。如果该位是1，则有一个类型4网络中断在等待：

- a. 读中断4发送者数据（ISD4）寄存器（偏移量38h）来获得32位数据。
- b. 读中断4发送者节点ID（SID4）寄存器（偏移量3Ch）来获得发送者节点ID。

第7步.可选：为了确定和服务另外的正在等待的中断，从第1步重复整个过程。

内存奇偶校验

如果应用程序需要使用内存交换网板的内存奇偶校验，应用程序必须向模板写数来允许奇偶校验并且对任何奇偶错误作出反应（如，记录一个用户应用故障）。为了允许奇偶校验功能，写1到本地控制状态寄存器（LCSR-区域1，偏移8h）的第27位上。你可以随意通过模板编程来为奇偶故障产生一个VME总线中断（见“VME总线中断”）。

只要允许奇偶校验，写往区域1的数据必须是双字（DWORD，32位）。当奇偶校验有效，字节（BYTE，8位）和字（WORD，16位）写到区域1是禁止的。如果尝试这样写，BUS_函数看上去是成功的完成了，然而，内存地址里的内容不会改变。模板通过将本地中断状态寄存器（LISR-区域2，偏移10h）中内存写丢弃位（Memory Write Discard）置1来报告故障。你可以随意为模板编程来为这个错误产生一个VME总线中断（见“VME总线中断”）。

在允许其发送器工作前，模板在上电时将区域1内的值初始化为0。这个工作为模板上所有的映像内存的奇偶位都做了初始化。

如果一个节点允许奇偶校验，同一网络上所有的节点都要允许。

下表荟聚了用于奇偶校验功能的位。

注意：置1或清除下叙位时，推荐使用BUS_RMW，这样寄存器中其它位不受影响。

本地控制状态寄存器（LCSR），区域2，偏移08h

名称	位	描述
奇偶校验允许	27	置1时，内存奇偶校验允许。

本地中断中断状态寄存器（LISR），区域2，偏移10h

名称	位	描述
LISR奇偶错误锁存	13	置1时，监测到奇偶错误。写0到本位可清除它。通常清除本位前先清IAKR的第7位。
内存写丢弃	12	置1时，在允许奇偶校验的情况下，由于违反了队列或长度限制而监测到一个内存写错误。写0可清除它。

中断应答寄存器（IAKR），区域4，偏移68h

名称	位	描述
IAKR奇偶错误锁存器	7	置1时，监测到一个奇偶错误。写1到本位来清除它。通常在先清本位，再清LISR的第13位。

侦测和清除一个奇偶错误

如果应用程序使用一个中断块监测和清除奇偶错误，你应该使用“中断处理逻辑”中描述的规则来清除奇偶错误（Parity Error）锁存器。

不然，也可以用以下的规则来监测和清除奇偶错误锁存器。也许想每个扫描周期都执行这个逻辑。

1. 执行一次BUS_RMW_BYTE函数，操作数赋0（“AND”），掩码参数赋DFh，区域4，偏移68h（IAKR）。中断应答寄存器的第7位是1时，本操作写1到该位，清除其锁存器。用DFh来屏蔽当前值，防止此寄存器的其它位被改变。
2. 执行一次BUS_RMW_DWORD函数，操作数赋0（“AND”），掩码参数赋FFFFDFFFh，区域2，偏移10h（LISR）。将一个变量地址给原始值输出参数（如，%T0001）。本操作获得本地中断状态寄存器的当前值，并且写0到13位来清除它。（不同的掩码值，如FFFFC038h，这时能用来清除LISR其它的位——如果你愿意的话）。
3. 测试LISR本地拷贝（如%T0013）的第13位。如果值为1，则有一个内存奇偶错误产生。

受令内存清除

内存交换网板支持写0到所有区域1的地址内的命令。只要发送器被允许，这些写（0）还能被发送到网络上。整个操作可能要持续几秒才能完成。执行这个命令时，模板和产生VME总线中断是不同的。

- 为了初始化这个命令，执行一次BUS_RMW_BYTE函数，操作数赋1（OR），掩码参数赋4h，区域3，偏移422h。
- 为了确定内存清除是否执行完毕，执行一次BUS_RD_BYTE函数，区域3，偏移442h。当清理内存（Clear Memory）为0，说明操作完成。

确定环网完整性

任何时候你都可以通过产生一个数据包并确定它是否返回来检查环网的完整性。它表明了是否所有的节点都连在了环网上，并且发送器都被允许。

按照以下步骤检查环网：

1. 通过执行一次BUS_RMW_BYTE函数（操作数赋0（AND），掩码参数赋FFh，区域2，偏移8h）来清除OWN DATA。
2. 通过向区域1写数或产生一个网络中断来初始化网络包。
3. 通过执行一次BUS_RD_BYTE来读LCSR（区域2，偏移08h）。
4. 测试OWN DATA状态（第0位）。
5. 重复第2步直到OWN DATA位变为1，才意味环网是完整的。

附录
A

内存交换网板说明

本附录提供了只对RMX和CMX模板适用的说明。对于所有RX7i模板都适用的通用说明及标准，RX7i控制系统安装要求，以及安装安全指导，请参考PAC系统安装手册（PACSystems RX7i Installation Manual, GFK-2223）。

内存交换网板性能说明

物理尺寸	PCB: 6.299"W×9.187"H
包长度	动态，由内存交换网板自动控制
用户内存	16MB SDRAM
工作电压	+5VDC（来自电源模板）
电流需求	1.8A
连接器	光纤LC类，符合IEC 61754-20标准。细节见第2章中的“光纤及连接器”

附录 B

寄存器定义

本附录给出了模板寄存器的详细定义：

区域2：主控制状态寄存器

区域3：辅控制状态寄存器

区域4：中断应答寄存器

注意：当RMX模板用作冗余连接时，这些内存区域在CPU的控制之下，不能通过逻辑访问。**BUS_×××**访问会失败，返回状态值为**8**（区域）。

注意：修改这些寄存器中独立的位时，建议用**BUS_RMW**函数，这样同一寄存器中的其它位就不会受影响。

区域2：主控制状态寄存器

偏移(hex)	寄存器	描述	访问	注释
00h-03h	--	保留	--	不要修改
04h	NID	节点ID寄存器	只读	显示硬件配置的节点ID号
05h-07h	--	保留	--	不要修改
08-0Bh	LCSR	本地控制状态寄存器	读写	一些位保留，一些位只读。 细节见B-3页
0C-0Fh	--	保留	--	不要修改
10-13h	LISR	本地中断状态寄存器	读写	一些位保留，一些位只读。
14-17h	LIER	本地中断允许寄存器	读写	细节见B-7页
18-1Bh	NTD	网络目标数据	读写	网络中断数据位。细节见B-8页。
1Ch	NTN	网络目标节点	读写	网络中断目标节点ID。细见B-8页。
1Dh	NIC	网络中断命令	读写	产生网络中断，细节见B-8页。
1E-1Fh	--	保留	--	不要修改
20-23h	ISD1	中断1发送者数据	读写	从网络中断1的FIFO中读用户定义数据。细节见B-8页。
24h	SID1	中断1发送者节点	读写	从网络中断1的FIFO中读节点ID。 细节见B-8页。
25-27h	--	保留	--	不要修改
28-2Bh	ISD2	中断2发送者数据		从网络中断2的FIFO中读用户定义数据。细节见B-8页。
2Ch	SID2	中断2发送者节点	读写	从网络中断2的FIFO中读节点ID。 细节见B-8页。
2D-2Fh	--	保留	--	不要修改
30-33h	ISD3	中断3发送者数据		从网络中断3的FIFO中读用户定义数据。细节见B-8页。
34h	SID3	中断3发送者节点	读写	从网络中断3的FIFO中读节点ID。 细节见B-8页。
35-37h	--	保留	--	不要修改
38-3Bh	ISD4	中断4发送者数据		从网络中断4的FIFO中读用户定义数据。细节见B-8页。
3Ch	SID4	中断4发送者节点	读写	从网络中断4的FIFO中读节点ID。 细节见B-8页。
3D-FFFh	--	保留	--	不要修改

本地控制状态寄存器 (LCSR)

偏移08h, 读写, 双字, 字, 字节访问

第07位	第06位	第05位	第04位	第03位	第02位	第01位	第00位
TX FIFO 是空的	TX FIFO 几乎满	锁存的 RX满	锁存的RX FIFO几乎满	同步 丢失	RX信号 监测	坏数据	自有数据 (OWN DATA)

第15位	第14位	第13位	第12位	第11位	第10位	第09位	第08位
保留							

第23位	第22位	第21位	第20位	第19位	第18位	第17位	第16位
保留							

第31位	第30位	第29位	第28位	第27位	第26位	第25位	第24位
保留				奇偶校 验允许	冗余传输 模式允许	无赖数据 包管理器 1允许	无赖数据 包管理器 0允许

本地控制状态寄存器位定义

位	名称	访问	说明
第00位	自有数据 (OWN DATA)	读写	自从上一次本位置0后, 到本位置1时, 模板收到传回来的自己的数据或中断包至少一次。本位同时也控制OWN DATA LED的状态。写0到本位来清除它。
第01位	坏数据	只读	本位为1时, 接收器回路监测到一个不合法包。数据也许过早从网络上被移除。写本地中断状态寄存器的相应位来清除本位。
第02位	RX信号 监测	只读	本位为1时, 模板接收器正在测试光。本位提供了即时状态(不锁存)
第03位	同步丢失	只读	自从上一次LISR相应位置0后, 到本位置1时, 光纤接受器至少有一次已经丢失了接受到的信号。数据也许过早从网络上被移除。很可能的原因是模板的配置改变, 接收光纤断开, 或上游节点发送器不允许(如, 断电, 配置改变, 或有意不允许)。写LISR中相应的位, 来清除本位。
第04位	RX FIFO 几乎满	只读	本位为1时, 接受FIFO几乎满了。这表明接收器回路在满负荷工作, 这在正常环境下是不会发生的。如果它确实发生了, 应用程序应该暂停对改模板的所有访问。写LISR中相应的位, 来清除本位。

位	名称	访问	说明
第05位	锁存的RX FIFO满	只读	本位为1时，接收FIFO已满。这表明了内存交换网板的工作不正常。数据也许过早从网络上被移除。写LISR的相应位来清除本位。
第06位	TX FIFO几乎满	只读	本位为1时，TX FIFO当前几乎已满。本位只提供即时状态（不锁存）。正常情况下要周期性诊断本位。
第07位	TX FIFO空	只读	本位为1时，TX FIFO当前是空的。本位只提供即时状态（不锁存）。
第08-15位	保留	--	写入0。
第16-17位	保留	--	不雅修改。
第18-19位	保留	--	写入0
第20-21位	保留	--	不需要修改。
第22-23位	保留	--	保留，而且应该写入0。
第24位	无赖数据包管理器0允许	只读	本位为1时，这块模板正用作无赖数据包管理器0。
第25位	无赖数据包管理器1允许	只读	本位为1时，这块模板正用作无赖数据包管理器1。
第26位	冗余传输模式允许	只读	本位为1时，冗余传输模式允许。本位为0时，选择的是快速或无冗余模。
第27位	本地总线奇偶允许	读写	本位为1时，内存奇偶校验允许，写入内存数据必须是双字DWORD型，并且偏移地址必须是4的倍数。16位或8位写入是不允许的。
第28-31位	保留	--	不要修改。

本地中断状态寄存器 (LISR)

偏移10h, 读写, 双字, 字, 字节访问

第07位	第06位	第05位	第04位	第03位	第02位	第01位	第00位
网络中断 4标志	无赖包 故障	保留			网络中断 3标志	网络中断 2标志	网络中断 1标志

第15位	第14位	第13位	第12位	第11位	第10位	第09位	第08位
自动清除 标志	中断 允许	LISR奇偶 错误锁存	内存写 丢弃	同步 丢失	RX FIFO满	RX FIFO 几乎满	坏数据

第23位	第22位	第21位	第20位	第19位	第18位	第17位	第16位
保留							

第31位	第30位	第29位	第28位	第27位	第26位	第25位	第24位
保留							

本地中断状态寄存器位定义

位	名称	访问	说明
第00位	网络中断 1标志	--	本位为1时, 至少有一个类型1中断在相应的FIFO中等待。本位保持为1, 直到中断1发送者节点ID (SID1) FIFO空。
第01位	网络中断 2标志	--	本位为1时, 至少有一个类型2中断在相应的FIFO中等待。本位保持为1, 直到中断2发送者节点ID (SID2) FIFO空。
第02位	网络中断 3标志	--	本位为1时, 至少有一个类型3中断在相应的FIFO中等待。本位保持为1, 直到中断3发送者节点ID (SID3) FIFO空。
第03- 05位	保留	--	不要修改。
第06位	无赖包 故障	读写	本位为1时, 模板监测并移除一个无赖数据包。本位是锁存的。一旦置1, 必须通过写0到本位地址上来清除它。
第07位	网络中断 4标志	--	本位为1时, 至少有一个类型4中断在相应的FIFO中等待。本位保持为1, 直到中断4发送者节点ID (SID4) FIFO空。
第08位	坏数据	读写	本位为1时, 接收器回路已经监测到一个不合法数据包。数据可能过早从网络上被移除。本位是锁存的。一旦置1, 必须通过写0到本位地址上来清除它。
第09位	RX FIFO 几乎满	读写	本位为1时, 接收器FIFO几乎要满。这表明接受器回路满负荷运行, 这在正常情况下是不允许的。如果这确实发生, 应用程序应该暂停对这块模板的访问。一旦置1, 它必须通过写0到本位地址来清除。

位	名称	访问	说明
第10位	RX FIFO 满	读写	本位为1时，接收器FIFO已经满了。这表明内存交换网板不正常工作。数据也许过早的从网络上被移除。本位是锁存的。一旦置1，必须通过写0到本位地址上来清除它。
第11位	同步丢失	读写	自从上一次LISR相应位置0后，到本位置1时，光纤接受器至少有一次已经丢失了接受到的信号。数据也许过早从网络上被移除。很可能的原因是模板的配置改变，接收光纤断开，或上游节点发送器不允许（如，断电，配置改变，或有意不允许）。写0来清除本位。如果这时一个发来的数据接受不到，本位会立刻置0。为了阻止连续中断，你可能想暂时将LIER中相应的位设为0，直到同步丢失条件被消除。
第12位	内存写 丢弃	读写	本位置1时，因为违反了队列或长度限制，一个内存写已经被丢弃（奇偶校验允许的情况下）。写0到本位来清除它。
第13位	LISR奇偶 错误锁存 器	读写	本位置1时，监测到一个奇偶错误。写0到本位清除它。在清除本位之前总是先要清除IARK的第7位
第14位	中断允许	读写	在模板产生一个VME总线中断之前，LIER中任何一个（你想要产生）中断标志以及相联的允许位必须置1，同时本位也必须置1。如果LIER中的自动清除（AUTO CLEAR）允许位置1，在LISR寄存器被读取时，中断允许位自动清除。
第15位	自动清除 标志	只读	本位为高阻（1）时，在LISR寄存器被读取时，中断允许位（第14位）自动被清除。
第16- 31位	保留	--	不要修改。

本地中断允许寄存器(LIER)

偏移14h, 读写, 双字, 字, 字节访问

第07位	第06位	第05位	第04位	第03位	第02位	第01位	第00位
网络中断 4允许中 断标志	无赖包故 障中断允 许标志	保留			网络中断 3允许中 断标志	网络中断 2允许中 断标志	网络中断 1允许中 断标志

第15位	第14位	第13位	第12位	第11位	第10位	第09位	第08位
自动清除 允许	保留	内存奇偶 错误中断 允许	阻止内存 写中断允 许	同步丢失 中断允许	RX FIFO 满中断允 许	RX FIFO 几乎满中 断允许	坏数据 中断允 许

第23位	第22位	第21位	第20位	第19位	第18位	第17位	第16位
保留							

第31位	第30位	第29位	第28位	第27位	第26位	第25位	第24位
保留							

这整个寄存器都是可读/写访问。保留位应该一直写为0。

当LISR中相应的位置1时, LIER中每一位都独立的使模板产生一个VME总线中断。为了使VME总线中断被允许, LISR中的中断允许位也必须为1。

位	名称	访问	说明
第15位	自动清除 允许	读写	本位为高阻(1)时, 在LISR寄存器被读取时, 中断允许位(第14位)自动被清除。

产生网络中断的寄存器

NTD,NTN和NIC寄存器用来产生网络中断。

网络目标数据寄存器 (NTD)

NTD寄存器是一个32位寄存器，偏移地址为18h。这是你写32位用户定义数据（它将同网络中断一起被发送）的地方。向这个寄存器中写数不会产生实际的中断。对NTD寄存器可以进行读写访问。

网络目标节点寄存器 (NTN)

NTN寄存器是一个8位寄存器，偏移地址为1Ch。这是写入你想要中断的节点ID的地方。如果你发送一个全局网络中断命令，本寄存器中的值就没有用途。

写NTN寄存器不会产生实际的网络中断。对这个寄存器可以进行读/写访问。NTN寄存器可以和NIC寄存器共同作为一个字（WORD）来同时读写。如执行一次BUS_WRT_WORD就可以对它们进行同时写。

网络中断命令寄存器 (NIC)

偏移1Dh，读写，双字，字，字节访问

NIC寄存器是一个8位寄存器，偏移1Dh。这是你写入你想要发送的中断的类型的地方。为了确定写入什么值，请你查询下表。NIC寄存器是可读写访问的。写NIC寄存器产生网络中断。

值	功能
00h	保留。不要使用。
01h	发送类型1网络中断到NTN中规定的节点
02h	发送类型2网络中断到NTN中规定的节点
03h	发送类型3网络中断到NTN中规定的节点
04h,05h	保留。不要使用。
06h	绕者环网发送一个测试中断，如果这个包能回到始发点，LISR中的自有数据位（OWN DATA）置1。环中的其它节点不受影响。如果你使用这个命令，NTN设为始发点的ID，NTD设为FFFFFFFFh。
07h	发送类型4网络中断到NTN中规定的节点
08h	保留。不要使用。
09h	发送类型1网络中断到其它所有节点。（全局）
0Ah	发送类型2网络中断到其它所有节点。（全局）
0Bh	发送类型3网络中断到其它所有节点。（全局）
0Ch-0Eh	保留。不要使用。
0Fh	发送类型4网络中断到其它所有节点。（全局）
10h-FF	保留。不要使用。

接受网络中断寄存器

SID[4-1]和ISD[4-1]，这4对寄存器是用来获取接受到的网络中断。

中断发送者数据寄存器

节点每次产生一个网络中断时，它都将自己节点ID作为包的一部分包含进去。当内存交换网板接受到一个网络中断后，它将发送者的节点ID和32位用户定义数据存储在FIFO（先入先出）队列中，这个队列与网络中断的类型相联系。FIFO可以存储最多127个中断。

应用逻辑可以通过读中断发送者数据（ISD_x）寄存器来获得FIFO中最早输入项的用户定义32位数。读ISD_x寄存器并不能消除FIFO中的输入项。只有通过访问中断发送者节点ID（SID_x）寄存器才能从FIFO移除相应的输入项。

为了确认FIFO中是否有一个输入项在等待，检查LISR中相应位。

中断发送者节点ID寄存器

节点每次产生一个网络中断时，它都将自己节点ID作为包的一部分包含进去。当内存交换网板接受到一个网络中断后，它将发送者的节点ID和32位用户定义数据存储在FIFO（先入先出）队列中，这个队列与网络中断的类型相联系。FIFO可以存储最多127个中断。

应用逻辑可以通过读中断发送者节点ID（SID_x）寄存器来获得FIFO中最早输入项的节点ID。这个寄存器被读取时，这个输入项（节点ID和用户数据）完全从FIFO中被移除。所以每个发送者ID只能读一次。如果需要发送者节点ID，在读取SID_x之前，一定先要读取ISD_x。

为了确认FIFO中是否有一个输入项在等待，检查LISR中相应位。

中断1发送者数据 (ISD1)

偏移20h，双字访问

像上面描述的那样获取类型1网络中断的32位用户定义数据。

中断1发送者节点ID (SID1)

偏移24h，字节访问

像上面描述的那样获取类型1网络中断的发送者节点ID，并FIFO中相应的整个输入项。

中断2发送者数据 (ISD2)

偏移28h，双字访问

像上面描述的那样获取类型2网络中断的32位用户定义数据。

中断2发送者节点ID (SID2)

偏移2Ch，字节访问

像上面描述的那样获取类型2网络中断的发送者节点ID，并FIFO中相应的整个输入项。

中断3发送者数据 (ISD3)

偏移30h，双字访问

像上面描述的那样获取类型3网络中断的32位用户定义数据。

中断3发送者节点ID (SID3)

偏移34h，字节访问

像上面描述的那样获取类型3网络中断的发送者节点ID，并FIFO中相应的整个输入项。

中断4发送者数据 (ISD4)

偏移38h，双字访问

像上面描述的那样获取类型4网络中断的32位用户定义数据。

中断4发送者节点ID (SID4)

偏移3Ch，字节访问

像上面描述的那样获取类型4网络中断的发送者节点ID，并FIFO中相应的整个输入项。

区域3：辅控制状态寄存器

你的应用程序千万不要访问以下没有做说明的区域3的其它部分。另外，通常使用下面函数中的一个来访问这个内存区域：

`BUS_RMW_BYTE`

`BUS_RD_BYTE`，长度为1

`BUS_WRT_BYTE`，长度为1

偏移量 440h

位	名称	访问	默认状态	描述
第0位	本地准备好LED	读写	0=off	写1打开LED。写0关LED。这个LED只在RMX模板上有。过滤系统失败（Fitered Systme Fail）位变为0时，本位置0。
第1位	本地运行LED	读写	0=off	写1打开LED。写0关LED。这个LED只在RMX模板上有。过滤系统失败（Fitered Systme Fail）位变为0时，本位置0。
第2位	远程准备好LED	读写	0=off	写1打开LED。写0关LED。这个LED只在RMX模板上有。过滤系统失败（Fitered Systme Fail）位变为0时，本位置0。
第3位	远程运行LED	读写	0=off	写1打开LED。写0关LED。这个LED只在RMX模板上有。过滤系统失败（Fitered Systme Fail）位变为0时，本位置0。
第4-7位	保留	--	--	不要修改。

偏移量 441h

位	名称	访问	默认状态	描述
0	信号监测	只读	NA	本位为1时，接收器正在监测光。本位仅提供即时状态（不锁存）。
1	发送器允许	读写	1（允许）	写1到本位允许发送器。写0允许传送器。
2	环回（Loopback）允许	读写	0（不允许）	写1到本位不允许收发器，并且导致发送的信号从模板内部回到发送器。
3	Dark-on-Dark允许	读写	0（不允许）	控制是（1）否（0）允许Dark-on-Dark。如果允许，模板在信号监测（Signal Detect）位为0时，自动关闭发送器；在信号监测位为1时，自动打开发送器。

位	名称	访问	默认状态	描述
4	自动传输不允许	读写	0（无自动不允许）	若本位为1，在过滤系统失败位变为0时，发送器自动不允许。
5	CMX/RMX指示器	只读	NA	本位为1时，模板是一个RMX。为0时，模板是一个CMX。
6	保留	--	0	不要修改。
7	保留	--	0	不要修改。

偏移量 442h

位	名称	访问	默认状态	描述
0	保留	--	0	不要修改。
1	保留	--	0	不要修改。
2	内存清理	读写	0	写1到本位，将区域1内所有的内存清除。一旦所有内存都被清除，本位会被置0。
3-7	保留	--	0	不要修改。

偏移量 445h

名称	访问	默认状态	描述
角色开关过滤时间	读写	64h（1秒）	代表角色开关（Role Switch）过滤时间。在过滤角色开关状态（Filtered Role Switch State）位置前，角色开关必须触发这么长时间。每个计数代表0.01秒。时间范围是0.01到2.54秒。

偏移量 446h

位	名称	访问	默认状态	描述
0	过滤系统失败	只读	1	0表示CPU失败。为0时，所有4个冗余状态LED全部熄灭。或见B-11页自动传输不允位。
1	当前角色开关状态	只读	NA	表明角色开关的当前输入状态。这个开关RMX模板上才有。
2	过滤角色开关状态	读写	0	为1时，角色开关已经在ON位置保持了角色开关过滤时间或更久。本位是锁存的。写1来清除本位。写0无效。
3-7	保留	--	0	不要修改。

区域4： 中断应答寄存器

你的应用程序千万不要访问以下没有做说明的区域4的其它部分。。

中断应答寄存器 (IAKR)

偏移： 68h， 字节， 字访问

本寄存器的正确使用在第4章中作了说明。总是使用BUS_RMW操作来访问这个寄存器。

位	名称	访问	说明
0-4	保留	--	不要修改。
5	保留	--	总是写为0。
6	保留	--	不要修改
7	IAKR奇偶错误锁存器	读写	置1时， 一个奇偶错误被监测到。写1到本位清除它。总是在清除LISR的第13位之前， 先清除本位。
8	主 (Master) VME中断允许	读写	写0到本位， 应答一个VME中断。写1到本位， 允许模板产生后发的中断。
9-15	保留	--	不要修改。

A

Auxiliary control and status registers
辅控制状态寄存器 (Region 3), B-11

B

Basic operation 基本操作, 1-4
BUS_ functions BUS函数, 3-2

C

Cables 光缆, 2-7
Checking ring integrity 检查环网完整性, 4-12
Compatibility 兼容性, 1-5
Configuration 配置, 2-9
Parameters 参数, 2-10
Connector 连接器, 2-7

D

Data transfer time 数据传输时间, 3-4

F

Features 特征, 1-2
Fiber-optic cables 光缆, 2-7
Fiber-optic connector 光纤连接器, 2-7
Functional compatibility 功能兼容性, 1-5

I

Installation, 2-6
cables, 2-7
connectors 安装光缆连接器, 2-7
Interrupt acknowledge register (Region 4),
中断应答寄存器 4-10, B-13
Interrupts 中断
associating with logic 与逻辑相联系, 4-5
masking 屏蔽, 4-6
multiple 多重, 4-9
network 网络, 4-7
VME Bus VME总线, 4-2

L

LEDs, 2-2
Local control and status register (LCSR),
本地控制状态寄存器 4-10, B-3
Local interrupt status register (LISR)
本地中断状态寄存器, 4-2, 4-10, B-5
Local interrupts enable register (LIER)
本地中断允许寄存器, 4-2, B-7

M

Memory clear 内存清除, 4-12
Memory parity checking 内存奇偶校验, 4-10
Multiple writes 多重写, 3-3

N

Network interrupts 网络中断, 4-7, B-8, B-9

O

On-demand memory clear
受令内存清除, 4-12
Optical transceiver 光学收发器, 2-3

P

Parameters 参数, 2-10
Parity 奇偶, 4-10
Primary control and status registers
(Region 2) 主控制状态寄存器, B-2
Publications 出版物, 1-5

R

Read modify write (RMW) access
读修改写访问, 3-2
Redundant transfer mode operation
冗余传输模式工作, 2-3
Region 2 区域 2, B-2
Region not Enabled 区域不允许, 3-1, 4-1, B-1
Registers for generating network
interrupts 产生网络中断寄存器,
B-8
Registers for receiving network interrupts 接
受网络中断寄存器, B-9
Related publications 相关出版物, 1-5
Rogue packet detection and removal 无赖数据包
侦测和移除, 2-5
Role switch 角色开关, 2-3

S

Specifications
performance, A-1

T

Test and set (TS) access 测试及设置访问, 3-2
Test ring 测试环网, 4-12, B-8
Transceiver 发送器, 2-3

U

User features 用户特征, 2-2

V

VME_functions. See BUS_functions

W

Window not Enabled窗口不允许, 3-1, 4-1, B-1

Writes

Multiple多重写入, 3-3