

34 多媒体卡接口 (MCI)

34.1 描述

多媒体卡接口 (MCI) 支持多媒体卡 (MMC) 规范 V2.2 和 SD 存储卡规范 V1.0。

MCI 包括一个命令寄存器，响应寄存器，数据寄存器，暂停计数器和错误检测逻辑，当需要时能自动处理命令发送，接收响应和有处理器开销限制的数据

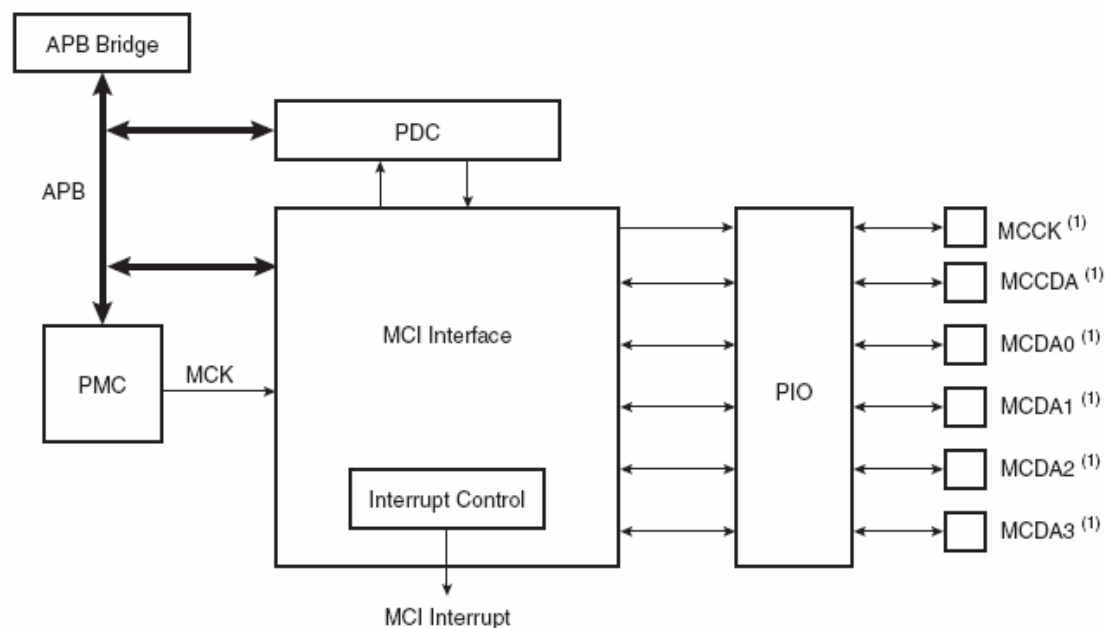
MCI 的最高工作频率为主控时钟 2 分频并支持 1 插槽连接。每个插槽可用于连接多媒体卡总线 (多达 30 个卡) 或 SD 存储卡。每次仅可选择一个插槽 (插槽为复用)。SD 卡寄存器中的一个位域执行此选择

SD 存储卡通信是基于一个 9 针接口 (时钟，命令，四个数据和三个电源线)，多媒体卡基于一个 7 针接口 (时钟，命令，一个数据，三个电源线和一個保留线日后使用)

SD 存储卡接口还支持多媒体卡操作。SD 和多媒体卡之间的主要区别是初始化过程和总线拓扑结构

34.2 框图

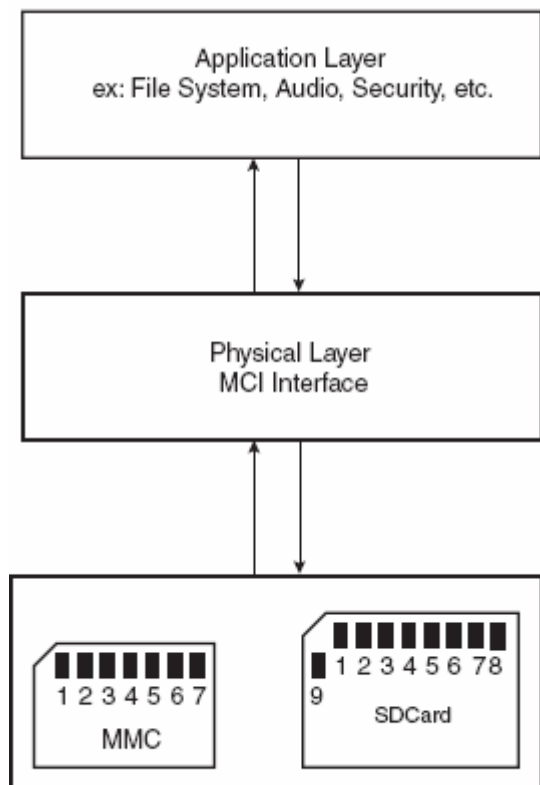
图 34-1 框图



注意：当在产品中嵌入若干 MCI (x MCI)，MCCK 参考 MCI_x_CK，MCCDA 参考 MCI_x_CDA，MCDA_y 参考 MCI_x_DA_y。

34.3 应用框图

图 34-2 应用框图



34.4 引脚名称列表

表 34-1 I/O 口线描述

引脚名称 (2)	引脚描述	类型 (1)	备注
MCCDA	命令/响应	I/O/PP/OD	MMC 或 SD 卡的 CMD
MCKK	时钟	I/O	MMC 或 SD 卡的 CLK
MCDA0 - MCDA3	插槽 A 数据 0...3	I/O/PP	MMC 的 DAT0 SD 卡的 DAT[0...3]

注意：1.I：输入，O：输出，PP：推/拉,OD：开漏

2.当在产品中嵌入若干 MCI (x MCI),MCKK 参考 MCIx_CK,MCCDA 参考 MCIx_CDA,MCDAy 参考 MCIx_DAy.

34.5 附属产品

34.5.1 I/O 口线

用于连接多媒体卡或 SD 卡的引脚可以和 PIO 口线复用。编程者必须首先编程 PIO 控制器来为 MCI 引脚指定外设功能

34.5.2 电源管理

MCI 可通过电源管理控制器 (PMC) 协调同步，所以编程者必须首先配置 PMC 来使能 MCI 时钟。

34.5.3 中断

MCI 接口有一个连接于高级中断控制器 (AIC) 的中断口线
处理 MCI 中断需要在配置 MCI 前编程 AIC

34.6 总线拓扑结构

图 34-3 多媒体存储卡总线结构



多媒体卡通信基于一个 7 针串行总线接口。有三个通信口线和四个电源口线。

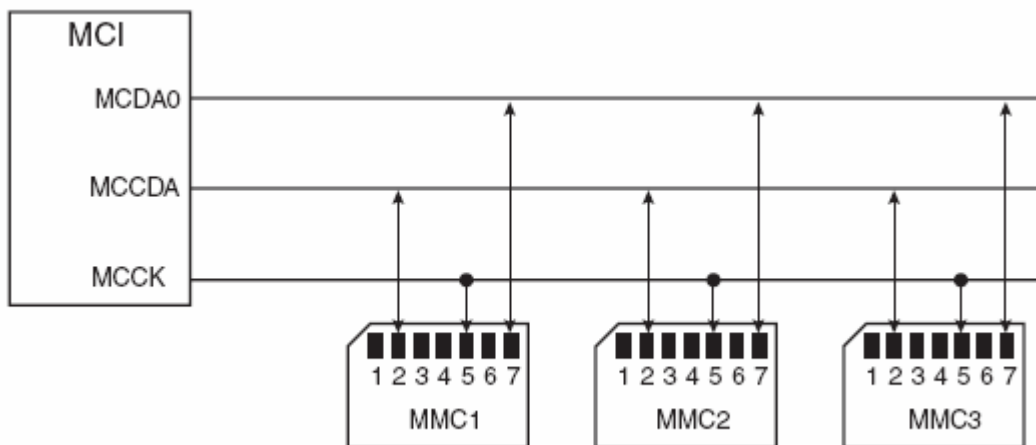
表 34-2 总线结构

引脚编号	名称	类型 (1)	描述	MCI 引脚名称 (2) (Slot z)
1	RSV	NC	未连接	-
2	CMD	I/O/PP/OD	命令/响应	MCCDz
3	VSS1	S	电源地	VSS
4	VDD	S	电源电压	VDD
5	CLK	I/O	时钟	MCCK
6	VSS2	S	电源地	VSS
7	DST[0]	I/O/PP	数据 0	MCDz0

注意：1. I: 输入, O: 输出, PP: 推/拉, OD: 开漏

2. 当在产品中嵌入若干 MCI (x MCI), MCCK 参考 MCIx_CK, MCCDA 参考 MCIx_CDA, MCDAy 参考 MCIx_DAy.

图 34-4 MMC 总线连接 (一个插槽)



注意：当在产品中嵌入若干 MCI (x MCI), MCCK 参考 MCIx_CK, MCCDA 参考 MCIx_CDA, MCDAy 参考 MCIx_DAy.

图 34-5 SD 存储卡总线结构



SD 存储卡总线包括表 34-3 中所列信号

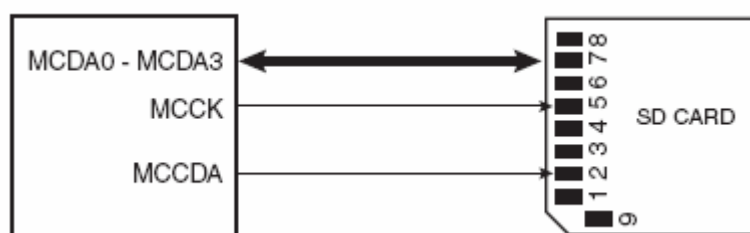
表 34-3 SD 存储卡总线信号

引脚编号	名称	类型 (1)	描述	MCI 引脚名称 (2) (Slot z)
1	CD/DAT[3]	I/O/PP	卡检测/数据口线位 3	MCKz3
2	CMD	PP	命令/响应	MCCDz
3	VSS1	S	电源地	VSS
4	VDD	S	电源电压	VDD
5	CLK	I/O	时钟	MCKK
6	VSS2	S	电源地	VSS
7	DAT[0]	I/O/PP	数据口线位 0	MCDz0
8	DAT[1]	I/O/PP	数据口线位 1	MCDz1
9	DAT[2]	I/O/PP	数据口线位 2	MCDz2

注意：1. I: 输入, O: 输出, PP: 推/拉, OD: 开漏

2. 当在产品中嵌入若干 MCI (x MCI), MCKK 参考 MCIx_CK, MCCDA 参考 MCIx_CDA, MCDAy 参考 MCIx_DAy.

图 34-6 用一个插槽连接 SD 卡总线



注意：当在产品中嵌入若干 MCI (x MCI), MCKK 参考 MCIx_CK, MCCDA 参考 MCIx_CDA, MCDAy 参考
 当配置 MCI 为 SD 存储卡操作时，可在 MCI_SDCR 寄存器中选择数据总线宽度。
 清零此寄存器的 SDCBUS 位表示一位宽度；置位表示四位宽度。对多媒体卡来
 说，仅使用数据口线 0. 其它数据口线可用作独立的 PIO

34.7 多媒体卡操作

上电复位后，一个专用的基于多媒体卡总线协议的信息将卡初始化。每个信息由以下任一记号表示：

- 命令：命令用来启动操作。从主机发送命令到单卡（寻址命令）或到所有已连接的卡（广播命令）。命令在 CMD 口线上串行发送。
- 响应：响应是从定址卡或（同步的）从所有已连接的卡到主机作为对前一个已接收命令做出回答。响应在 CMD 口线上串行传输。
- 数据：数据在卡和主机之间传输。数据由数据口线传输。

在初始化阶段由总线控制器对当前连接的卡进行地址分配，实现卡定址。每个卡都有其唯一的 CID 编号。

在多媒体卡系统规范中有对命令，响应和数据部件结构的描述。见 525 页的表 34-4。

多媒体卡总线传输由这些内容组成。

多媒体卡有不同类型的操作。定址操作通常包括命令和响应记号。另外，有些操作包含数据记号；其它的操作在命令或响应结构内直接传输信息。此时，操作中不出现数据记号。DAT 和 CMD 口线上的位以 MCI 时钟同步传输。

定义两种类型的数据传输命令：

- 序列命令：这些命令初始化一个连续的数据流。它们仅当 CMD 口线上跟随一个停止命令时终止。此模式将命令开销减少到一个极小范围。

- 块定向命令：这些命令由 CRC 位连续的发送数据块。

读/写操作都允许单个或多块发送。类似于序列读，当 CMD 口线上跟随一个停止命令时多块发送终止

MCI 提供一套寄存器来执行所有的多媒体卡操作。

34.7.1 命令-响应操作

复位后，MCI 禁用并且只有在置位 MCI_CR 控制寄存器中的 MCIEN 位后生效。

当总线待用时 PWSEN 位通过 $2^{PWSDIV} + 1$ 分频 MCI 时钟来降低功耗。

卡的命令和响应由 MCI 时钟的上升沿协调同步。

多媒体卡的所有时序都定义在多媒体卡系统规范中。

需要两种总线模式（开漏和推/拉）来处理所有定义在 MCI 命令寄存器中的操作。MCI_CMDR 允许执行命令。

例如，执行 ALL_SEND_CID 命令：

CMD

主机命令					Nid 周期					CID			
S	T	内容	CRC	E	Z	*****	Z	S	T	内容	Z	Z	Z

ALL_SEND_CID 命令和 MCI_CMDR 控制寄存器的域和值见表 34-4 和表 34-5 描述。

表 34-4 ALL_SEND_CID 命令描述

CMD 索引	类型	变量	Resp	缩写	命令描述
CMD2	Bcr	[31: 0] 填充位	R2	ALL_SEND_CID	要求所有卡在 CMD 口线上发送其 CID 编号

注意：bcr 就是带响应的广播命令

表 34-5 MCI_CMDR 命令寄存器的域和值

域	值
CMDNB (命令编号)	2 (CMD2)
RSPTYP (响应类型)	2 (R2: 136 位响应)
SPCMD (专用命令)	0 (非专用命令)
OPCMD (开漏命令)	1
MAXLAT (命令响应的最大延迟时间)	0 (NID 周期 == 5 周期)
TRCMD (传输命令)	0 (不传输)
TRDIR (传输方向)	X (仅在传输命令可用)
TRTYP (传输类型)	X (仅在传输命令可用)

MCI_ARGR 包含命令的变量域。

要发送命令，用户必须执行以下步骤：

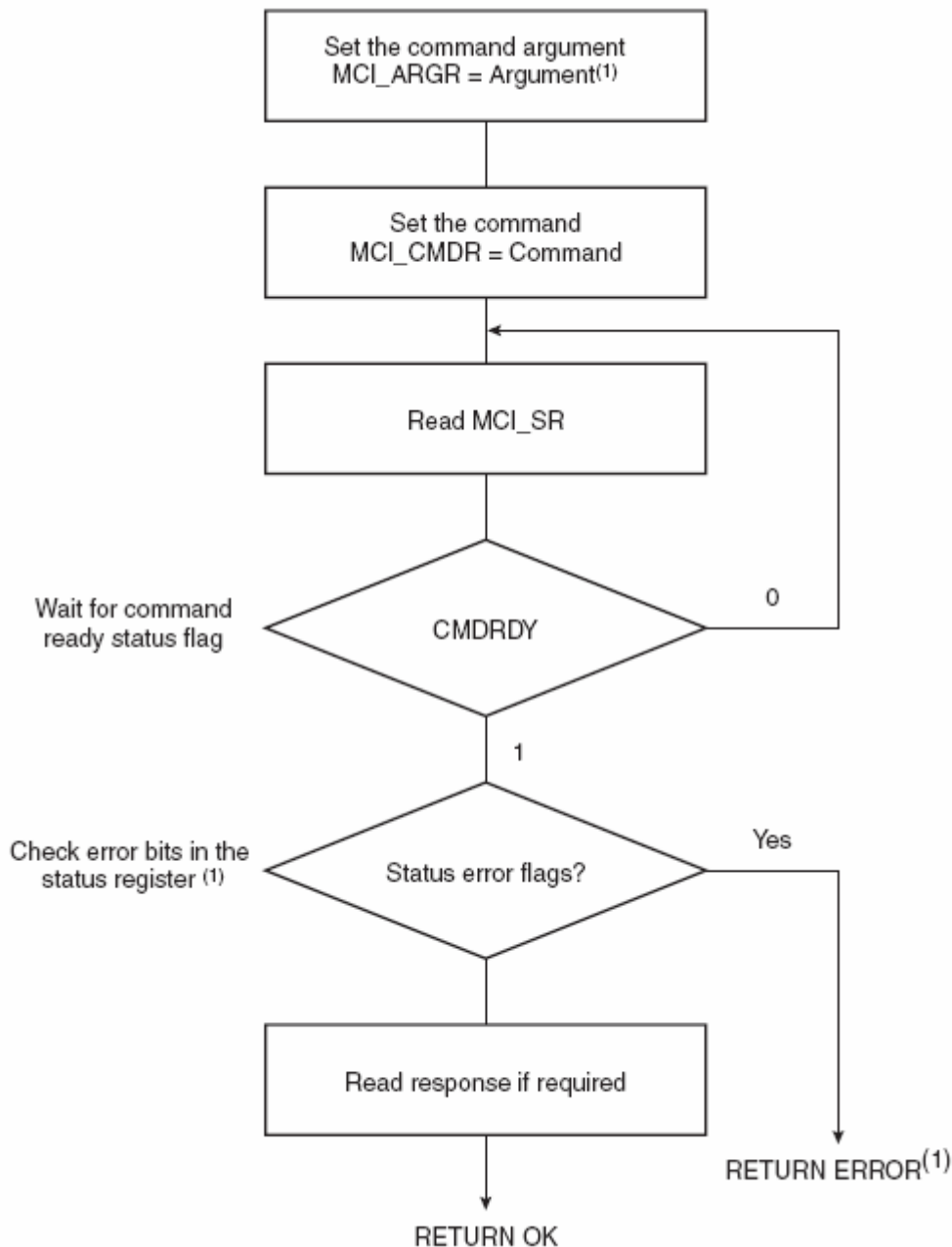
- 将命令变量填入变量寄存器 (MCI_ARGR)
- 设置命令寄存器 (MCI_CMDR) (见表 34-5)

写命令寄存器后立即发送该命令。当命令完成时激活状态寄存器 (MCI_SR) 中的状态位 CMDRDY。若此命令需要响应，可在 MCI 响应寄存器 (MCI_RSPR) 中读取到。根据命令响应大小从 48 位到 136 位。MCI 内置了一个错误检测来防止传

输期间的数据破坏。

下面的流程图展示了怎样发送命令到卡及在需要时读取响应。该例中，对状态寄存器位进行轮询，但是要设置中断使能寄存器中的相应位以允许使用中断方式。

图 34-7 命令/响应功能流程图



注意：1.若为 SEND_OP_COND 命令，则 CRC 错误标志位总会出现（参考多媒体卡规范中的 R3 响应）

34.7.2 数据传输操作

多媒体卡允许多种读/写操作（单块，多块，流等）。通过设置 MCI 命令寄存器 (MCI_CMDR) 中的传输类型 (TRTYP) 域来选择这些传输。

可用外围 DMA 控制器 (PDC) 功能完成这些操作。若 MCI_MR 中的 PDCMODE 位置位，则接下来所有的读/写都使用 PDC 设备

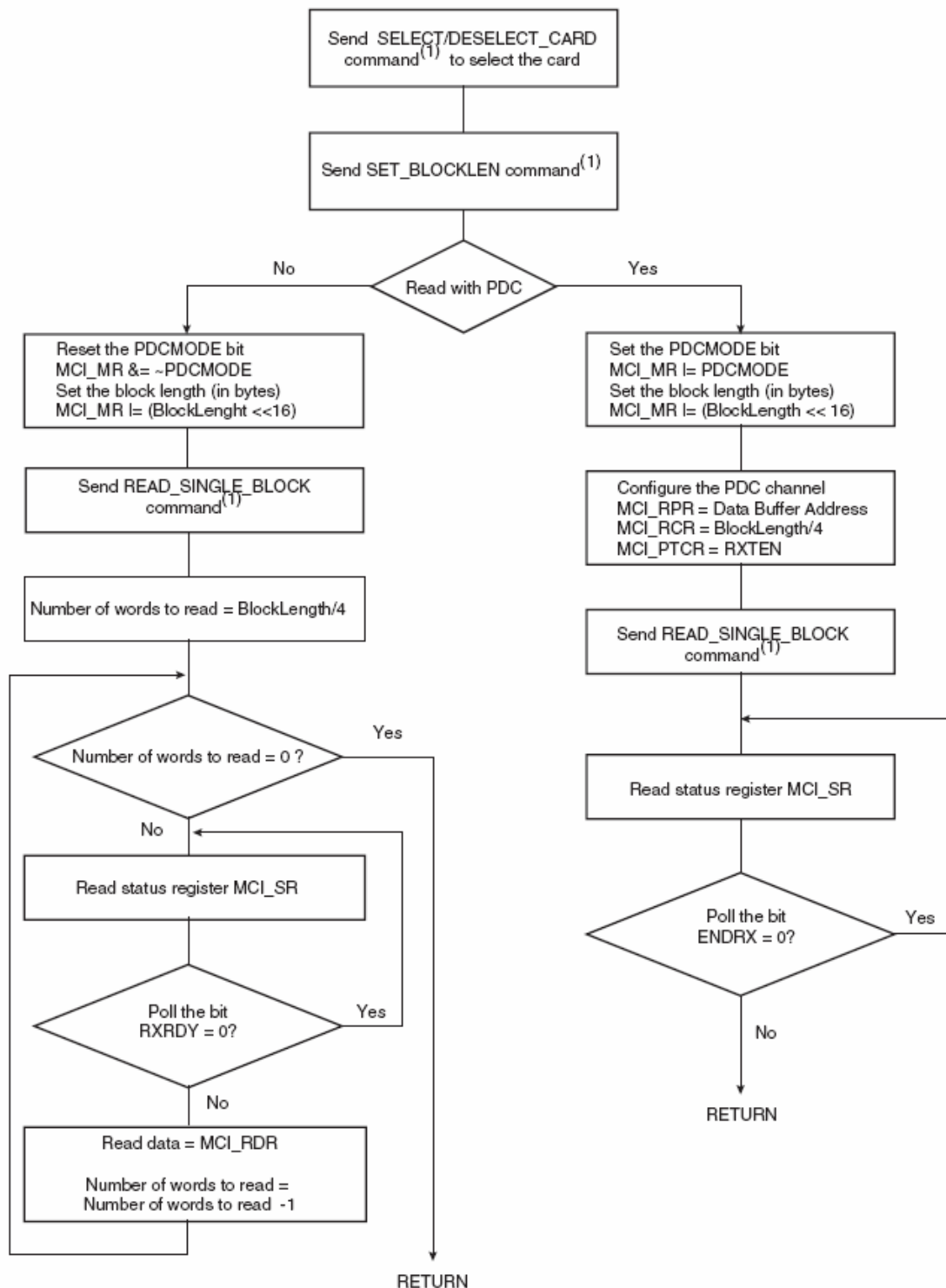
所有情况下，块长度必须在模式寄存器 MCI_MR 中定义。
此域决定数据块的大小

34.7.3 读操作

下面的流程图展示了使用或不用 PDC 设备时怎样读取单时钟。此例中（见图 34-8），使用轮询发等待读取结束。

类似的，用户可配置中断使能寄存器 (MCI_IER) 在读取接收时触发一个中断

图 34-8 读功能流程图



注意：1.假设此命令已正确发送（见图 34-7）

34.7.4 写操作

写操作中，MCI 模式寄存器 (MCI_MR) 用来定义写非多块大小的填充值。

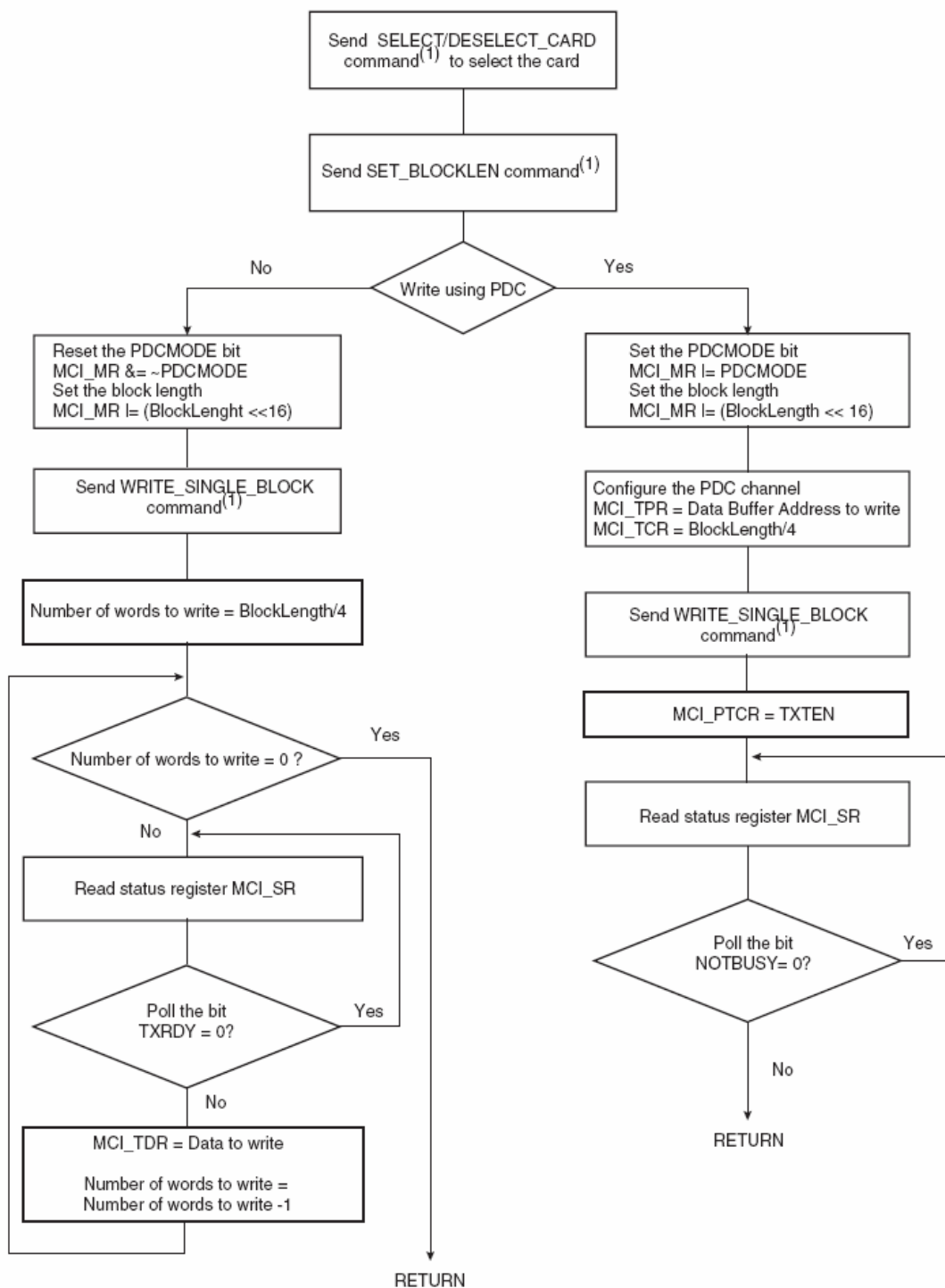
若 PDCPADV 位为 0，则当填充数据时使用 0x00 值，否则使用 0xFF

若置位，PDCMODE 位使能 PDC 传输。

下面的流程图展示了使用或不用 PDC 工具时怎样写一个单块(见图 37-9)。

可使用轮询或中断方法根据中断屏蔽寄存器 (MCI_IMR) 中的内容等待写结束

图 34-9 写功能流程图

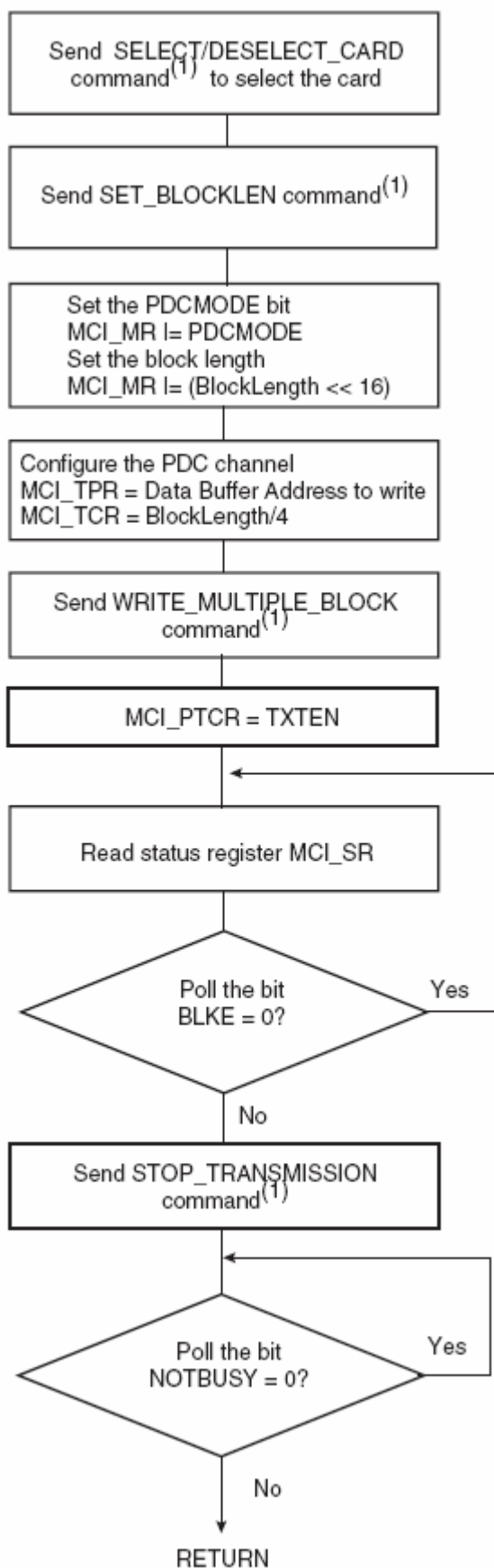


注意：1.假设此命令已正确发送（见图 34-7）

下面的流程图展示了使用 PDC 时怎样管理一个多写块传输(见图 34-10)。

根据中断屏蔽寄存器 (MCI_ 的内容可使用轮询或中断方法等待写结束

图 34-10 多写功能流程图



注意：1. 假设此命令已正确发送（见图 34-7）

34.8 SD 操作

多媒体卡接口能执行 SD 存储（安全数字存储卡）卡命令。

SD 卡基于多媒体卡（MMC）格式，但尺寸稍微大一些并且数据传输速率性能高一些，边上有一个锁定开关来防止意外改写和安全特性。其物理形状因素，引脚分配和数据传输协议附加一些条件就与多媒体卡前向兼容。SD 受众多专利和商标保护，只有通过安全数字卡协会许可才可使用。

SD 卡通信基于 9 针接口（时钟，命令，4×数据和 3×电源线）。规范中定义了通信协议。SD 卡和多媒体卡之间的主要差别是在初始化过程。

SD 卡总线允许动态配置数据口线数量。上电后，对数据传输 SD 卡默认仅使用 DAT0 作为数据传输。初始化后，主机可以改变总线宽度（有效数据口线数）。

34.9 多媒体卡接口（MCI）用户接口

表 34-6 寄存器映射

偏移量	寄存器	寄存器名称	读/写	复位值
0x00	控制寄存器	MCI_CR	写	-
0x04	模式寄存器	MCI_MR	读/写	0x0
0x08	数据超时寄存器	MCI_DTOR	读/写	0x0
0x0C	SD 卡寄存器	MCI_SDCR	读/写	0x0
0x10	变量寄存器	MCI_ARGR	读/写	0x0
0x14	命令寄存器	MCI_CMDR	写	-
0x18-0x1C	保留	-	-	-
0x20	响应寄存器（1）	MCI_RSPR	读	0x0
0x24	响应寄存器（1）	MCI_RSPR	读	0x0
0x28	响应寄存器（1）	MCI_RSPR	读	0x0
0x2C	响应寄存器（1）	MCI_RSPR	读	0x0
0x30	接收数据寄存器	MCI_RDR	读	0x0
0x34	发送数据寄存器	MCI_TDR	写	-
0x38-0x3C	保留	-	-	-
0x40	状态寄存器	MCI_SR	读	0xC0E5
0x44	中断使能寄存器	MCI_IER	写	-
0x48	中断禁用寄存器	MCI_IDR	写	-
0x4C	中断屏蔽寄存器	MCI_IMR	读	0x0
0x50-0xFC	保留	-	-	-
0x100-0x124	保留给 PDC	-	-	-

注意：1.响应寄存器可由 MCI_RSPR 读 N 次或在连续地址（0x20 到 0x2C），N 由响应大小决定

34.9.1 MCI 控制寄存器

名称: MCI_CR

访问类型: 只写

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
SWRST	-	-	-	PWSDIS	PWSEN	MCIDIS	MCIEN

- **MCIEN:** 多媒体接口使能

0=无效

1=若 MCDIS 为 0，使能多媒体接口

- **MCIDIS:** 多媒体接口禁用

0=无效

1=禁用多媒体接口

- **PWSEN:** 节能模式使能

0=无效

1=若 PWSDIS 为 0，使能节能模式

警告: 使能此模式前，用户必须在 PWSDIV 域（模式寄存器 MCI_MR）设置一个非零值

- **PWSDIS:** 节能模式禁用

0=无效

1=禁用节能模式

- **SWRST:** 软件复位

0=无效

1=复位 MCI。执行软件硬件复位 MCI 接口

34.9.2 MCI 模式寄存器

名称: MCI_MR

访问类型: 读/写

31	30	29	28	27	26	25	24
BLKLEN							
23	22	21	20	19	18	17	16
BLKLEN						0	0
15	14	13	12	11	10	9	8
PDCMODE	PDCPADV	-	-	-	PWSDIV		
7	6	5	4	3	2	1	0
CLKDIV							

- CLKDIV: 时钟分频器

多媒体卡接口时钟 (MCCK 或 MCI_CK) 为 $(2 * (\text{CLKDIV} + 1))$ 分频的主控时钟 (MCK)。

- PWSDIV: 节能分频器

当进入节能模式时, $2^{(\text{PWSDIV})} + 1$ 分频多媒体卡接口时钟

警告: 在使能 MCI_CR (MCI_PWSEN 位) 中的节能模式前该值必须为非零。

- PDCPADV: PDC 填充值

0=当填充数据到写传输时使用 0x00 值 (不仅 PDC 传输)

1=当填充数据到写传输时使用 0xFF 值 (不仅 PDC 传输)

- PDCMODE: PDC 定向模式

0=禁用 PDC 传输

1=使能 PDC 传输。此情况下, 在 PDC 传输已完成重新激活 MCI 模式寄存器 (MCI_SR) 中的 UNRE 和 OVRE 标志位

- BLKLEN: 数据块长度

此域决定数据块的大小

位 16 和位 17 必须置为 0

34.9.3 MCI 数据超时寄存器

名称: MCI_DTOR

访问类型: 读/写

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	DTOMUL			DTCYC			

- DTCYC: 数据超时周期数
- DTOMUL: 数据超时乘数

此域决定 MCI 在两个数据块传输之间等待的最大主控周期数

它等于 (DTCYC × 乘数).

乘数如下表所示由 DTOMUL 定义

DTOMUL			乘数
0	0	0	1
0	0	1	16
0	1	0	128
0	1	1	256
1	0	0	1024
1	0	1	4096
1	1	0	65536
1	1	1	1048576

若已超越由 DTCYC 和 DTOMUL 设置的数据时限, 则 MCI 状态寄存器 (MCI_SR) 中的数据超时错误标志位 (DTE) 升高

34.9.4 MCI SD 卡寄存器

名称: MCI_SDCR

访问类型: 读/写

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	-	-	-	-	-
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
SDCBUS	-	-	-	-	-	-	SDCSEL

● SDCSEL: SD 卡选择

SDCSEL		SD 卡插槽
0	0	选择插槽 A
0	1	保留
1	0	保留
1	1	保留

● SDCBUS: SD 卡总线宽度

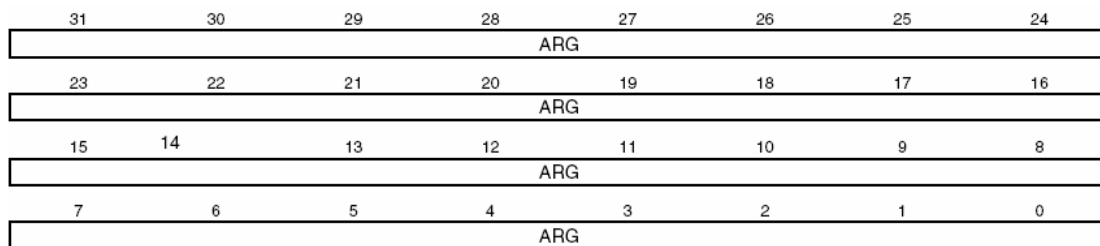
0=1 位数据总线

1=4 位数据总线

34.9.5 MCI 变量寄存器

名称: MCI_ARGR

访问类型: 读/写



- ARG: 命令变量

34.9.6 MCI 命令寄存器

名称: MCI_CMDR

访问类型: 只写

31	30	29	28	27	26	25	24
-	-	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	-	-	TRTYP		TRDIR	TRCMD	
15	14	13	12	11	10	9	8
-	-	-	MAXLAT	OPDCMD	SPCMD		
7	6	5	4	3	2	1	0
RSPTYP		CMDNB					

当 MCI_SR 中 CMDRDY 为 0 时此寄存器写保护。若发送一个中断命令，此寄存器仅可被中断响应 (SPCMD 域) 写。这就是说当前命令不能被中断或修改。

- CMDNB: 命令号
- RSPTYP: 响应类型

RSP		响应类型
0	0	无响应
0	1	48 位响应
1	0	136 位响应
1	1	保留

- SPCMD: 专用命令

SPCMD			命令
0	0	0	无专用 CMD
0	0	1	初始化 CMD: 初始化序列须 74 个时钟周期
0	1	0	同步 CMD: 发送挂起命令前等待当前数据块传输结束
0	1	1	保留
1	0	0	中断命令: 对应中断模式 (CMD40)
1	0	1	中断响应: 对应中断模式 (CMD40)

- OPDCMD: 开漏命令
0=推拉命令
1=开漏命令
- MAXLAT: 命令到响应最大等待时间
0=5 周期最大等待
1=64 周期最大等待
- TRCMD: 传输命令

TRCMD		传输类型
0	0	无数据传输
0	1	启动数据传输
1	0	停止数据传输

1	1	保留
---	---	----

● TRDIR: 传输检测

0=写

1=读

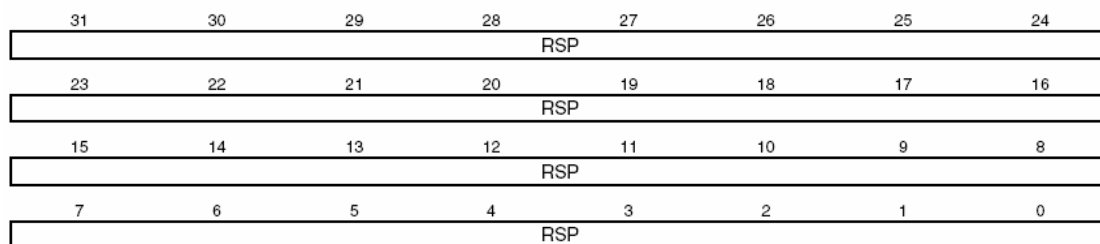
● TRTYP: 传输类型

TRTYP			传输类型
0	0	0	MMC/SD 卡单块
0	0	1	MMC/SD 卡多块
0	1	0	MMC 流
0	0	1	保留

34.9.7 MCI 响应寄存器

名称: MCI_RSPR

访问类型: 只读



- RSP: 响应

注意: 1. 可通过 N 次访问在相同的 MCI_RSPR 或在连续地址 (0x20 到 0x2C) 上读取响应寄存器。

N 取决于响应的大小

34.9.8 MCI 接收数据寄存器

名称: MCI_RDR

访问类型: 只读

31	30	29	28	27	26	25	24
DATA							
23	22	21	20	19	18	17	16
DATA							
15	14	13	12	11	10	9	8
DATA							
7	6	5	4	3	2	1	0
DATA							

- DATA: 读取数据

34.9.9 MCI 发送数据寄存器

名称: MCI_TDR

访问类型: 只写

31	30	29	28	27	26	25	24
DATA							
23	22	21	20	19	18	17	16
DATA							
15	14	13	12	11	10	9	8
DATA							
7	6	5	4	3	2	1	0
DATA							

- DATA: 写数据

34.9.10 MCI 状态寄存器

名称: MCI_SR

访问类型: 只读

31	30	29	28	27	26	25	24
UNRE	OVRE	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	DTOE	DCRCE	RTOE	RENDE	RCRCE	RDIRE	RINDE
15	14	13	12	11	10	9	8
TXBUFE	RXBUFE	-	-	-	-	-	-
7	6	5	4	3	2	1	0
ENDTX	ENDRX	NOTBUSY	DTIP	BLKE	TXRDY	RXRDY	CMDRDY

- CMDRDY: 命令就绪
0=命令正在处理
1=上一个命令已发送。当写 MCI_CMDR 时清零
- RXRDY: 接收器就绪
0=自上次读取 MCI_RDR 还未接收到数据
1=自上次读取 MCI_RDR 还已接收到数据
- TXRDY: 发送就绪
0=上次写入 MCI_TDR 的数据还未传输到移位寄存器
1=上次写入 MCI_TDR 的数据已传输到移位寄存器

- BLKE: 数据块结束

此标志必须仅对写操作使用

0=还未完成数据块传输。当读取 MCI_SR 时清零

1=数据块传输结束，包括 CRC16 状态发送

在 PDC 模式(PDCMODE=1)，当上一数据块的 CRC 状态已发送（已置位 TXBUFE）则标志位置位
否则(PDCMODE=0)，对每一个发送 CRC 状态标志位置位。

对更多关于 CRC 状态的细节参考 MMC 或 SD 规范

- DTIP: 数据传输正在进行

0=无传输数据正在进行

1=当前数据传输正在进行，包括 CRC16 计算。在 CRC16 结束时清零

NOTBUSY: MCI 不忙

此标志必须仅对写操作使用

数据块写操作使用数据口线(DAT0)上写操作持续的采样忙信号：在一个数据块传输期间，若卡没有空闲的数据接收缓冲区，则该卡通过轮询数据口线(DAT0)的 LOW 信号指示此状态。只要此卡至少有一个对已定义数据传输块长度的接收缓冲区变为空闲则该卡就停止轮询数据口线。

NOTBUSY 标志允许处理这些不同状态

0=对新的数据传输 MCI 未就绪。在卡响应结束时清零

1=对新的数据传输 MCI 未就绪。当数据口线上的忙状态结束时置位。这对应于卡的空闲间断数据接收缓冲

对更多关于忙动作的细节参考 MMC 或 SD 规范

- ENDRX: RX 缓冲结束

0=自上次写 MCI_RCR 或 MCI_RNCR 接收计数器寄存器未达到 0

1=自上次写 MCI_RCR 或 MCI_RNCR 接收计数器寄存器已达到 0

- ENDTX: TC 缓冲结束

0=自上次写 MCI_RCR 或 MCI_RNCR 发送计数器寄存器未达到 0

1=自上次写 MCI_RCR 或 MCI_RNCR 发送计数器寄存器已达到 0

注意：可使用 BLKE 和 NOTBUSY 标志来检查数据口线上数据已成功发送或不仅仅从 PDC 传输到 MCI 控制器

- RXBUFF: RX 缓冲区满

0=MCI_RCR 或 MCI_RNCR 为非零值

1=MCI_RCR 和 MCI_RNCR 都为零值

注意：可使用 BLKE 和 NOTBUSY 标志来检查数据口线上数据已成功发送或不仅仅从 PDC 传输到 MCI 控制器

- RINDE: 响应索引错误

0=无错误

1=在命令索引发送和响应索引接收之间检测到误匹配。当写 MCI_CMDR 时清零

- RDIRE: 响应方向错误

0=无误

1=在响应时还未检测到从卡到主机的方向位

- RCRCE: 响应 CRC 错误

0=无误

1=在响应时检测到一个 CRC7 错误。当写 MCI_CMDR 时清零

- RENDE: 响应结束位错误

0=无误

1=还未检测到响应结束位。当写 MCI_CMDR 时清零

- RTOE: 响应超时错误

0=无误

1=通过 MCI_CMDR 中的 MAXLAT 已超越时限来置位响应超时。当写 MCI_CMDR 时清零

- DCRCE: 数据 CRC 错误

0=无误

1=在上一数据块中检测到 CRC16 错误。通过读 MCI_SR 寄存器复位

- DTOE: 数据超时错误

0=无误

1=通过 MCI_DTOR 中的 DTOCYC 和 DTOMUL 已超越时限来置位数据超时。通过读 MCI_SR 寄存器复位

- OVRE: 超限

0=无误

1=至少已丢失一个 8 位已接收数据（未读）。当发送一个新数据传输命令时清零

- UNRE: 欠载运行

0=无误

1=至少一个未带有效信息（未写）的 8 位数据已发送。当发送一个新数据传输命令时清零

- RXBUFF: RX 缓冲区满

0=MCI_RCR 或 MCI_RNCR 为非零值

1=MCI_RCR 和 MCI_RNCR 都为零值

- TXBUFE: TX 缓冲区空

0=MCI_TCR 或 MCI_TNCR 为非零值

1=MCI_TCR 和 MCI_TNCR 都为零值

34.9.11 MCI 中断使能寄存器

名称: MCI_IER

访问类型: 只写

31	30	29	28	27	26	25	24
UNRE	OVRE	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	DTOE	DCRCE	RTOE	RENDE	RCRCE	RDIRE	RINDE
15	14	13	12	11	10	9	8
TXBUFE	RXBUFF	-	-	-	-	-	-
7	6	5	4	3	2	1	0
ENDTX	ENDRX	NOTBUSY	DTIP	BLKE	TXRDY	RXRDY	CMDRDY

- CMDRDY: 命令就绪中断使能
- RXRDY: 接收器就绪中断使能
- TXRDY: 发送就绪中断使能
- BLKE: 数据块结束中断使能
- DTIP: 数据传输正在进行中断使能
- NOTBUSY: 数据不忙中断使能
- ENDRX: 接收缓冲结束中断使能
- ENDTX: 发送缓冲结束中断使能
- RXBUFF: 接收缓冲区满中断使能
- TXBUFE: 发送缓冲区空中断结束
- RINDE: 响应索引错误中断使能
- RDIRE: 响应方向错误中断使能
- RCRCE: 响应 CRC 错误中断使能
- RENDE: 响应结束位错误中断使能
- RTOE: 响应超时错误中断使能
- DCRCE: 数据 CRC 错误中断使能
- DTOE: 数据超时错误中断使能
- OVRE: 超限中断使能
- UNRE: 欠载运行中断使能

0=无效

1=使能相应中断

34.9.12 MCI 中断禁用寄存器

名称: MCI_IDR

访问类型: 只写

31	30	29	28	27	26	25	24
UNRE	OVRE	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	DTOE	DCRCE	RTOE	RENDE	RCRCE	RDIRE	RINDE
15	14	13	12	11	10	9	8
TXBUFE	RXBUFF	-	-	-	-	-	-
7	6	5	4	3	2	1	0
ENDTX	ENDRX	NOTBUSY	DTIP	BLKE	TXRDY	RXRDY	CMDRDY

- CMDRDY: 命令就绪中断禁用
- RXRDY: 接收器就绪中断禁用
- TXRDY: 发送就绪中断禁用
- BLKE: 数据块结束中断禁用
- DTIP: 数据传输正在进行中断禁用
- NOTBUSY: 数据不忙中断禁用
- ENDRX: 接收缓冲结束中断禁用
- ENDTX: 发送缓冲结束中断禁用
- RXBUFF: 接收缓冲区满中断禁用
- TXBUFE: 发送缓冲区空中断禁用
- RINDE: 响应索引错误中断禁用
- RDIRE: 响应方向错误中断禁用
- RCRCE: 响应 CRC 错误中断禁用
- RENDE: 响应结束位错误中断禁用
- RTOE: 响应超时错误中断禁用
- DCRCE: 数据 CRC 错误中断禁用
- DTOE: 数据超时错误中断禁用
- OVRE: 溢出中断禁用
- UNRE: 欠载运行中断禁用

0=无效

1=禁用相应中断

34.9.13 MCI 中断屏蔽寄存器

名称: MCI_IMR

访问类型: 只读

31	30	29	28	27	26	25	24
UNRE	OVRE	-	-	-	-	-	-
23	22	21	20	19	18	17	16
-	DTOE	DCRCE	RTOE	RENDE	RCRCE	RDIRE	RINDE
15	14	13	12	11	10	9	8
TXBUFE	RXBUFF	-	-	-	-	-	-
7	6	5	4	3	2	1	0
ENDTX	ENDRX	NOTBUSY	DTIP	BLKE	TXRDY	RXRDY	CMDRDY

- CMDRDY: 命令就绪中断屏蔽
- RXRDY: 接收器就绪中断屏蔽
- TXRDY: 发送就绪中断屏蔽
- BLKE: 数据块结束中断屏蔽
- DTIP: 数据传输正在进行中断屏蔽
- NOTBUSY: 数据不忙中断屏蔽
- ENDRX: 接收缓冲结束中断屏蔽
- ENDTX: 发送缓冲结束中断屏蔽
- RXBUFF: 接收缓冲区满中断屏蔽
- TXBUFE: 发送缓冲区空中断结束
- RINDE: 响应索引错误中断屏蔽
- RDIRE: 响应方向错误中断屏蔽
- RCRCE: 响应 CRC 错误中断屏蔽
- RENDE: 响应结束位错误中断屏蔽
- RTOE: 响应超时错误中断屏蔽
- DCRCE: 数据 CRC 错误中断屏蔽
- DTOE: 数据超时错误中断屏蔽
- OVRE: 超限中断屏蔽
- UNRE: 欠载运行中断屏蔽

0=相应中断未使能

1=相应中断使能



Powered by Team Mcuzone

QQ:8204136

Website: www.mcuzone.com

2009

