

SJA1000 独立 CAN 控制器

目录:

- 1. 特性
- 2. 总体说明
- 3. 订货信息
- 4. 方块图
- 5. 引脚
- 6. 功能说明
 - 6.1 CAN 控制器的模块说明
 - 6.1.1 接口管理逻辑 (IML)
 - 6.1.2 发送缓冲器 (TXB)
 - 6.1.3 接收缓冲器 (TXB, RXFIFO)
 - 6.1.4 验收滤波器 (ACF)
 - 6.1.5 位流处理器 (BSP)
 - 6.1.6 位时序逻辑 (BTL)
 - 6.1.7 错误管理逻辑 (EML)
 - 6.2 CAN 控制器的详细说明
 - 6.2.1 与 PCA82C200 的兼容性
 - 6.2.2 BasicCAN 和 PeliCAN 模式的区别
 - 6.3 BasicCAN 模式
 - 6.3.1 BasicCAN 的地址列表
 - 6.3.2 复位值
 - 6.3.3 控制寄存器 (CR)
 - 6.3.4 命令寄存器 (CMR)
 - 6.3.5 状态寄存器 (SR)
 - 6.3.6 中断寄存器 (IR)
 - 6.3.7 发送缓冲器列表
 - 6.3.8 接收缓冲器
 - 6.3.9 验收滤波器
 - 6.4 PeliCAN 模式
 - 6.4.1 PeliCAN 地址列表
 - 6.4.2 复位值
 - 6.4.3 模式寄存器 (MOD)
 - 6.4.4 命令寄存器 (CMR)
 - 6.4.5 状态寄存器 (SR)
 - 6.4.6 中断寄存器 (IR)
 - 6.4.7 中断使能寄存器 (IER)
 - 6.4.8 仲裁丢失捕捉寄存器 (ALC)
 - 6.4.9 误码捕捉寄存器 (ECC)
 - 6.4.10 错误报警时序寄存器 (EWLR)
 - 6.4.11 RX 错误计数寄存器 (RXERR)
 - 6.4.12 TX 错误计数寄存器 (TXERR)
 - 6.4.13 发送缓冲器
 - 6.4.14 接收缓冲器
 - 6.4.15 验收滤波器
 - 6.4.16 RX 信息计数器 (RMC)

- 6.4.17 RX 缓冲器起始地址寄存器 (RBSA)
- 6.5 常规寄存器
 - 6.5.1 总线定时寄存器 0 (BTR0)
 - 6.5.2 总线定时寄存器 1 (BTR1)
 - 6.5.3 输出控制寄存器 (OCR)
 - 6.5.4 时钟分频寄存器 (CDR)
- 7. 限值
- 8. 温度特性
- 9. 直流特性
- 10. 交流特性
 - 10.1 交流时序图
 - 10.2 附录交流信息
- 11. 封装概述
- 12. 生命支持系统中的应用

1. 特性

- 和 PCA82C200 独立 CAN 控制器引脚兼容
- 和 PCA82C200 独立 CAN 控制器电气兼容
- PCA82C200 模式（即默认的 BasicCAN 模式）
- 扩展的接收缓冲器（64 字节、先进先出 FIFO）
- 和 CAN2.0B 协议兼容（PCA82C200 兼容模式中的无源扩展帧）
- 同时支持 11 位和 29 位识别码
- 位速率可达 1Mbits/s
- PeliCAN 模式扩展功能：
 - 可读/写访问的错误计数器
 - 可编程的错误报警限制
 - 最近一次错误代码寄存器
 - 对每一个 CAN 总线错误的中断
 - 具体控制位控制的仲裁丢失中断
 - 单次发送（无重发）
 - 只听模式（无确认、无活动的出错标志）
 - 支持热插拔（软件位速率检测）
 - 验收滤波器扩展（4 字节代码，4 字节屏蔽）
 - 自身信息接收（自接收请求）
- 24MHz 时钟频率
- 对不同微处理器的接口
- 可编程的 CAN 输出驱动器配置
- 增强的温度适应（-40~+125℃）

2. 总体说明

SJA1000 是一种独立控制器，用于移动目标和一般工业环境中的区域网络控制（CAN）。它是 PHILIPS 半导体 PCA82C200 CAN 控制器（BasicCAN）的替代产品。而且，它增加了一种新的工作模式（PeliCAN），这种模式支持具有很多新特性的 CAN 2.0B 协议。

3. 订货信息

型号	封装		
	名称	说明	版本
SJA1000	DIP28	塑质双列直插封装；28引脚（600mil）	SOT117-1
SJA1000T	SO28	塑质小型线外封装；28引脚；外宽7.5mm	SOT136-1

4. 方框图

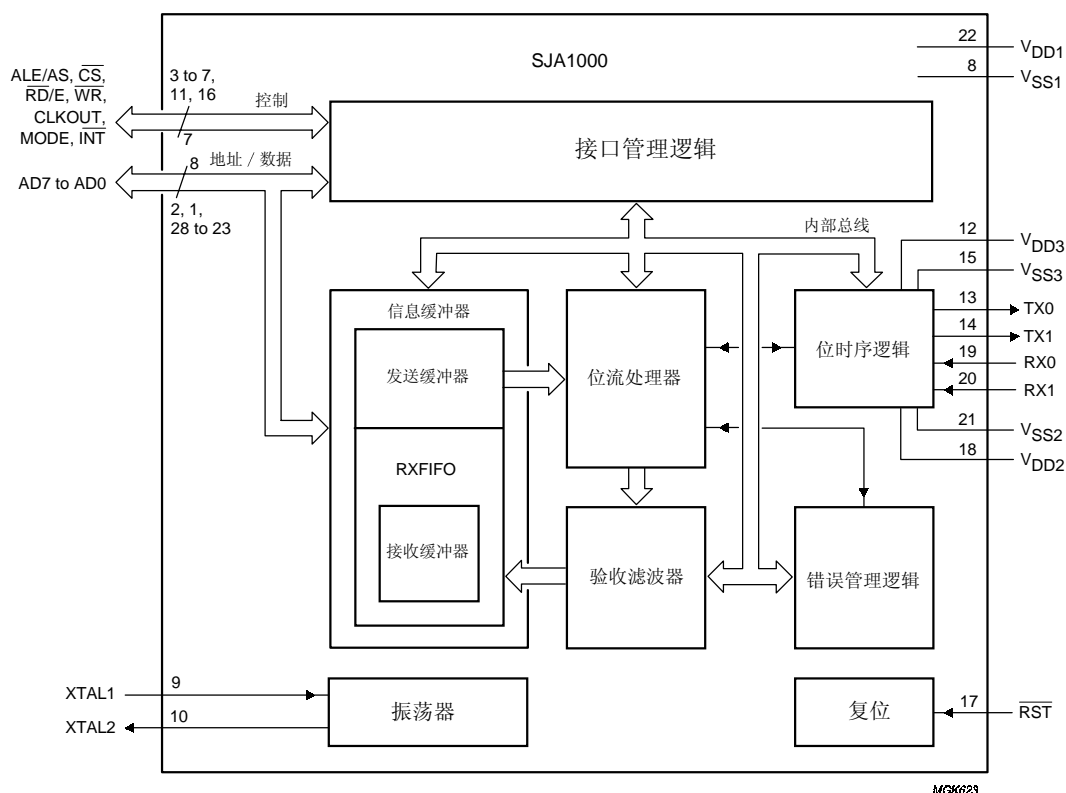


图 1 方块图

5. 管脚排列

符号	引脚	说明
AD7-AD0	2, 1, 28-23	多路地址/数据总线
ALE/AS	3	ALE输入信号 (Intel模式), AS输入信号 (Motorola模式)
/CS	4	片选输入, 低电平允许访问SJA1000
(/RD) /E	5	微控制器的/RD信号 (Intel模式) 或E使能信号 (Motorola模式)
/WR	6	微控制器的/WR信号 (Intel模式) 或RD/(/WR) 信号 (Motorola模式)
CLKOUT	7	SJA1000产生的提供给微控制器的时钟输出信号; 时钟信号来源于内部振荡器且通过编程驱动; 时钟控制寄存器的时钟关闭位可禁止该引脚。
VSS1	8	接地
XTAL1	9	输入到振荡器放大电路; 外部振荡信号由此输入; 注1
XTAL2	10	振荡放大电路输出; 使用外部振荡信号时左开路输出; 注1
MODE	11	模式选择输入 1=Intel模式 0=Motorola模式
VDD3	12	输出驱动的5V电压源
TX0	13	从CAN输出驱动器0输出到物理线路上
TX1	14	从CAN输出驱动器1输出到物理线路上
VSS3	15	输出驱动器接地

/INT	16	中断输出，用于中断微控制器；/INT在内部中断寄存器各位都被置位时低电平有效；/INT是开漏输出，且与系统中的其它/INT是线或的；此引脚上的低电平可以把IC从睡眠模式中激活
/RST	17	复位输入，用于复位CAN接口（低电平有效）；把/RST引脚通过电容连到V _{SS} ，通过电阻连到V _{DD} 可自动上电复位（例如，C=1μF；R=50kΩ）
V _{DD2}	18	输入比较器的5V电压源
RX0, RX1	19, 20	从物理的CAN总线输入到SJA1000的输入比较器；支配（控制）电平将会唤醒SJA1000的睡眠模式；如果RX1比RX0的电平高，就读支配（控制）电平，反之读弱势电平；如果时钟分频寄存器的CBP位（见表49）被置位，就旁路CAN输入比较器以减少内部延时（此时连有外部收发电路）；这种情况下只有RX0是激活的；弱势电平被认为是高而支配电平被认为是低
V _{SS2}	21	输入比较器的接地端
V _{DD1}	22	逻辑电路的5V电压源

注：

1. XTAL1 和 XTAL2 引脚必须通过 15pF 的电容连到 V_{SS1}。

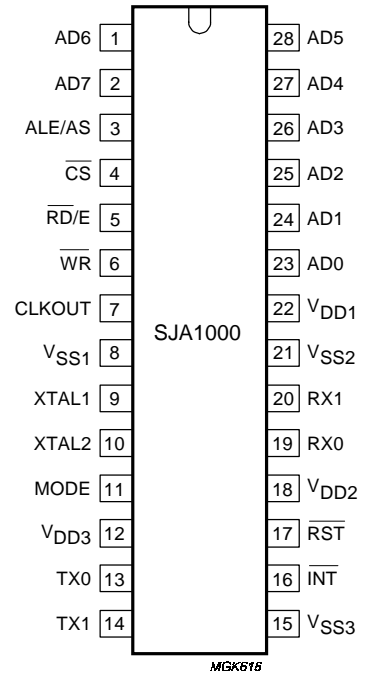


图2 引脚配置（DIP28）

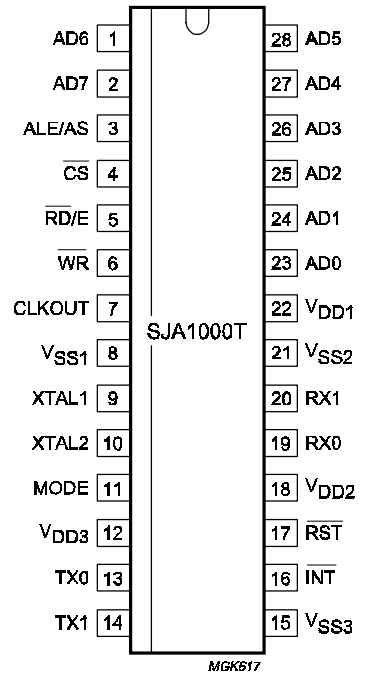


图3 引脚配置（SO28）

6. 功能说明

6.1 CAN 控制模块的说明

6.1.1 接口管理逻辑（IML）

接口管理逻辑解释来自 CPU 的命令，控制 CAN 寄存器的寻址，向主控制器提供中断信息和状态信息。

6.1.2 发送缓冲器（TXB）

发送缓冲器是 CPU 和 BSP（位流处理器）之间的接口，能够存储发送到 CAN 网络上的完整信息。缓冲器长 13 个字节，由 CPU 写入、BSP 读出。

6.1.3 接收缓冲器（RXB, RXFIFO）

接收缓冲器是验收滤波器和 CPU 之间的接口，用来储存从 CAN 总线上接收和接收的信息。接收缓冲

器 (RXB, 13 个字节) 作为接收 FIFO (RXFIFO, 长 64 字节) 的一个窗口, 可被 CPU 访问。

CPU 在此 FIFO 的支持下, 可以在处理信息的时候接收其它信息。

6.1.4 验收滤波器 (ACF)

验收滤波器把它其中的数据和接收的识别码的内容相比较, 以决定是否接收信息。在纯粹的接收测试中, 所有的信息都保存在 RXFIFO 中。

6.1.5 位流处理器 (BSP)

位流处理器是一个在发送缓冲器、RXFIFO 和 CAN 总线之间控制数据流的程序装置。它还在 CAN 总线上执行错误检测、仲裁、填充和错误处理。

6.1.6 位时序逻辑 (BTL)

位时序逻辑监视串口的 CAN 总线和处理与总线有关的位时序。它在信息开头“弱势-支配”的总线传输时同步 CAN 总线位流 (硬同步), 接收信息时再次同步下一次传送 (软同步)。BTL 还提供了可编程的时间段来补偿传播延迟时间、相位转换 (例如, 由于振荡漂移) 和定义采样点和一位时间内的采样次数。

6.1.7 错误管理逻辑 (EML)

EML 负责传送层模块的错误管制。它接收 BSP 的出错报告, 通知 BSP 和 IML 进行错误统计。

6.2 CAN 控制器的详细说明

SJA1000 在软件和引脚上都是与它的前一款—PCA82C200 独立控制器兼容的。在此基础上它增加了很多新的功能。为了实现软件兼容, SJA1000 增加修改了两种模式:

- BasicCAN 模式: 与 PCA82C200 兼容
- PeliCAN 模式: 扩展特性

工作模式通过时钟分频寄存器中的 CAN 模式位来选择。复位默认模式是 Basic CAN 模式。

6.2.1 与 PCA82C200 兼容性

在 Basic CAN 模式中, SJA1000 模仿 PCA82C200 独立控制器所有已知的寄存器。在 6.2.1.1 到 6.2.1.4 节中所描述的特性不同于 PCA82C200, 这主要是为了软件上的兼容性。

6.2.1.1 同步模式

在 SJA1000 的控制寄存器中没有 SYNC 位 (在 PCA82C200 中是 CR.6 位)。同步只有在 CAN 总线上“弱势-支配 (控制)”的转换时才有可能发生。写这一位是没有任何影响的。为了与现有软件兼容, 读取这一位时是可以把以前写入的值读出的 (对触发电路无影响)。

6.2.1.2 时钟分频寄存器

时钟分频寄存器用来选择 CAN 工作模式 (Basic CAN /Peli CAN)。它使用从 PCA82C200 保留下来的一位。象在 PCA82C200 中一样, 写一个 0-7 之间的值, 就将进入 Basic CAN 模式。默认状态是 12 分频的 Motorola 模式和 2 分频的 Intel 模式。保留的另一位补充了一些附加的功能。CBP 位 (见表 49) 的置位使内部 RX 输入比较器旁路, 这样在使用外部传送电路时可以减少内部延时。

6.2.1.3 接收缓冲器

PCA82C200 中双接收缓冲器的概念被 Peli CAN 中的接收 FIFO 所代替。这对软件除了会增加数据溢出的可能性之外, 不会产生应用上的影响。在数据溢出之前, 缓冲器可以接收两条以上信息 (最多 64 字节)。

6.2.1.4 CAN 2.0B

SJA1000 被设计为全面支持 CAN 2.0B 协议, 这就意味着在处理扩展帧信息的同时, 扩展振荡器的误差被修正了。在 Basic CAN 模式下只可以发送和接收标准帧信息 (11 字节长的识别码)。如果此时检测到 CAN 总线上有扩展帧的信息, 如果信息正确, 也会被允许且给出一个确认信号, 但没有接收中断产生。

6.2.2 Basic CAN 和 Peli CAN 模式的区别

在 Peli CAN 模式下, SJA1000 有一个含很多新功能的重组寄存器。SJA1000 包含了设计在 PCA82C200 中的所有位及一些新功能位。Peli CAN 模式支持 CAN 2.0B 协议规定的所有功能 (29 字节的识别码)。

SJA1000 的主要新功能:

- 标准帧和扩展帧信息的接收和传送
- 接收 FIFO (64 字节)
- 在标准和扩展格式中都有单/双验收滤波器 (含屏蔽和代码寄存器)
- 读/写访问的错误计数器
- 可编程的错误限制报警

- 最近一次的误码寄存器
- 对每一个 CAN 总线错误的错误中断
- 仲裁丢失中断以及详细的位位置
- 一次性发送（当错误或仲裁丢失时不重发）
- 只听模式（CAN 总线监听，无应答，无错误标志）
- 支持热插（无干扰软件驱动位速检测）
- 硬件禁止 CLKOUT 输出

6.3 BasicCAN 模式

6.3.1 BasicCAN 地址表

SJA1000 是一种 I/O 设备基于内存编址的微控制器。双设备的独立操作是通过象 RAM 一样的片内寄存器修正来实现的。

SJA1000 的地址区包括控制段和信息缓冲区。控制段在初始化载入是可被编程来配置通讯参数的（例如，位时序）。微控制器也是通过这个段来控制 CAN 总线上的通讯的。在初始化时，CLKOUT 信号可以被微控制器编程指定一个值。

应发送的信息会被写入发送缓冲器。成功接收信息后，微控制器从接收缓冲器中读取接收的信息，然后释放空间以做下一步应用。

微控制器和 SJA1000 之间状态、控制和命令信号的交换都是在控制段中完成的。这个段的列表见表 3。初始载入后，寄存器的验收代码、验收屏蔽、总线定时寄存器 0 和 1 以及输出控制就不能改变了。只有控制寄存器的复位位被置高时，才可以访问这些寄存器。

在以下两种不同的模式中访问寄存器是不同的：

- 复位模式
- 工作模式

当硬件复位或控制器掉线（见表 5，状态寄存器的总线状态位）时会自动进入复位模式（见表 3，控制寄存器的位复位请求）。工作模式是通过置位控制寄存器的复位请求位激活的。

表 1 BasicCAN 地址分配表；注 1

CAN地址	段	工作模式		复位模式	
		读	写	读	写
0	控制	控制	控制	控制	控制
1		(FFH)	命令	(FFH)	命令
2		状态	-	状态	-
3		(FFH)	-	中断	-
4		(FFH)	-	验收代码	验收代码
5		(FFH)	-	验收屏蔽	验收屏蔽
6		(FFH)	-	总线定时0	总线定时0
7		(FFH)	-	总线定时1	总线定时1
8		(FFH)	-	输出控制	输出控制
9		测试	测试；注2	测试	测试；注2
10	发送缓冲器	识别码 (10-3)	识别码 (10-3)	(FFH)	-
11		识别码 (2-0) RTR和DLC	识别码 (2-0) RTR和DLC	(FFH)	-
12		数据字节1	数据字节1	(FFH)	-
13		数据字节2	数据字节2	(FFH)	-
14		数据字节3	数据字节3	(FFH)	-
15		数据字节4	数据字节4	(FFH)	-

16		数据字节5	数据字节5	(FFH)	-
17		数据字节6	数据字节6	(FFH)	-
18		数据字节7	数据字节7	(FFH)	-
19		数据字节8	数据字节8	(FFH)	-
20	接收缓冲器	识别码 (10-3)	识别码 (10-3)	识别码 (10-3)	识别码 (10-3)
21		识别码 (2-0) RTR和DLC	识别码 (2-0) RTR和DLC	识别码 (2-0) RTR和DLC	识别码 (2-0) RTR和DLC
22		数据字节1	数据字节1	数据字节1	数据字节1
23		数据字节2	数据字节2	数据字节2	数据字节2
24		数据字节3	数据字节3	数据字节3	数据字节3
25		数据字节4	数据字节4	数据字节4	数据字节4
26		数据字节5	数据字节5	数据字节5	数据字节5
27		数据字节6	数据字节6	数据字节6	数据字节6
28		数据字节7	数据字节7	数据字节7	数据字节7
29		数据字节8	数据字节8	数据字节8	数据字节8
30		(FFH)	-	(FFH)	-
31		时钟分频器	时钟分频器;注3	时钟分频器	时钟分频器

注:

1. 必须注明的是寄存器在高端 CAN 地址区被重复 (8 位 CPU 地址的最高位是不参与解码的; CAN 地址 32 是和 CAN 地址 0 连续的)
2. 测试寄存器只用于产品测试。正常操作中使用这个寄存器会导致设备不可预料的结果。
3. 许多位在复位模式中是只写的 (CAN 模式和 CBP)

6.3.2 复位值

检测到有复位请求后将中止当前接收/发送的信息而进入复位模式。一旦向复位位传送了 ‘1-0’ 的下降沿, CAN 控制器将返回工作模式。

表 2 复位模式的配置; 注 1 和注 2

寄存器	位	符号	名称	值	
				硬件复位	软件或总线关闭复位CR. 0
控制	CR.7	-	保留	0	0
	CR.6	-	保留	×	×
	CR.5	-	保留	1	1
	CR.4	OIE	溢出中断使能	×	×
	CR.3	EIE	错误中断使能	×	×
	CR.2	TIE	发送中断使能	×	×
	CR.1	RIE	接收中断使能	×	×
	CR.0	RR	复位请求	1(复位模式)	1(复位模式)
命令	CMR.7	-	保留	注3	注3
	CMR.6	-	保留		
	CMR.5	-	保留		
	CMR.4	GTS	睡眠		
	CMR.3	CDO	清除数据溢出		
	CMR.2	RRB	释放接收缓冲器		
	CMR.1	AT	中止传送		

	CMR.0	TR	发送请求		
状态	SR.7	BS	总线状态	0(总线开启)	×
	SR.6	ES	出错状态	0(ok)	×
	SR.5	TS	发送状态	0(空闲)	0(空闲)
	SR.4	RS	接收状态	0(空闲)	0(空闲)
	SR.3	TCS	发送完毕状态	1(完毕)	×
	SR.2	TBS	发送缓冲器状态	1(释放)	1(释放)
	SR.1	DOS	数据溢出状态	0(无溢出)	0(无溢出)
	SR.0	RBS	接收缓冲器状态	0(空)	0(空)
中断	IR.7	-	保留	1	1
	IR.6	-	保留	1	1
	IR.5	-	保留	1	1
	IR.4	WUI	唤醒中断	0(复位)	0(复位)
	IR.3	DOI	数据溢出中断	0(复位)	0(复位)
	IR.2	EI	错误中断	0(复位)	×;注4
	IR.1	TI	发送中断	0(复位)	0(复位)
	IR.0	RI	接收中断	0(复位)	0(复位)
验收代码	AC.7-0	AC	验收代码	×	×
验收屏蔽	AM.7-0	AM	验收屏蔽	×	×
总线定时0	BTR0.7	SJW.1	×同步跳转宽度1	×	×
	BTR0.6	SJW.0	同步跳转宽度0	×	×
	BTR0.5	BRP.5	波特率预设值5	×	×
	BTR0.4	BRP.4	波特率预设值4	×	×
	BTR0.3	BRP.3	波特率预设值3	×	×
	BTR0.2	BRP.2	波特率预设值2	×	×
	BTR0.1	BRP.1	波特率预设值1	×	×
	BTR0.0	BRP.0	波特率预设值0	×	×
总线定时1	BTR1.7	SAM	采样	×	×
	BTR1.6	TSEG2.2	时间段2.2	×	×
	BTR1.5	TSEG2.1	时间段2.1	×	×
	BTR1.4	TSEG2.0	时间段2.0	×	×
	BTR1.3	TSEG1.3	时间段1.3	×	×
	BTR1.2	TSEG1.2	时间段1.2	×	×
	BTR1.1	TSEG1.1	时间段1.1	×	×
	BTR1.0	TSEG1.0	时间段1.0	×	×
输出控制	OC.7	OCTP1	输出控制晶体管P1	×	×
	OC.6	OCTN1	输出控制晶体管N1	×	×
	OC.5	OCPOL1	输出控制极性1	×	×
	OC.4	OCTP0	输出控制晶体管P0	×	×
	OC.3	OCTN0	输出控制晶体管N0	×	×
	OC.2	OCPOL0	输出控制极性0	×	×
	OC.1	OCMODE1	输出控制模式1	×	×

	OC.0	OCMODE0	输出控制模式0	×	×
发送缓冲器	-	TXB	发送缓冲器	×	×
接收缓冲器	-	RXB	接收缓冲器	×;注5	×;注5
时钟分频器	-	CDR	时钟分频寄存器	0000 0000 (Intel); 0000 0101 (Motorola)	×

注:

1. ‘×’表示这些寄存器或位不受影响。
2. 括号中是功能说明。
3. 读命令寄存器的结果总是‘1111 1111’。
4. 总线关闭时错误中断位被置位(此中断被允许情况下)。
5. RXFIFO 的内部读/写指针被设置成初始化值。连续的读 RXB 会得到一些未定义的数据(部分旧信息)。发送信息时,信息并行写入接收缓冲器,但不产生接收中断且接收缓冲区是不锁定的。所以,即使接收缓冲器是空的,最近一次发送的信息也可从接收缓冲器读出,直到它被下一条发送或接收的信息取代。

硬件复位时, RXFIFO 的指针指到物理地址‘0’的 RAM 单元。软件设置 CR.0 或因为总线关闭的缘故 RXFIFO 的指针将被设置到当前有效 FIFO 的开始地址,这个地址不同于物理的 RAM 地址‘0’,而是第一次释放接收缓冲器命令后的有效起始地址。

6.3.3 控制寄存器 (CR)

控制寄存器的内容是用于改变 CAN 控制器的行为的。这些位可以被微控制器设置或复位,微控制器可以对控制寄存器进行读/写操作。

表 3 控制寄存器各位的说明; CAN 地址 0

位	符号	名称	值	功能
CR.7	—	—	—	保留;注1
CR.6	—	—	—	保留;注2
CR.5	—	—	—	保留;注3
CR.4	OIE	溢出中断使能	1	使能;如果置位数据溢出位,微控制器接收溢出中断信号(见状态寄存器;表5)
			0	禁能;微控制器不从 SJA1000 接收溢出中断信号
CR.3	EIE	错误中断使能	1	使能;如果出错或总线状态改变,微控制器接收错误中断信号(见状态寄存器;表5)
			0	禁能;微控制器不从 SJA1000 接收错误中断信号
CR.2	TIE	发送中断使能	1	使能;当信息被成功发送或发送缓冲器又被访问时,(例如,中止发送命令后),微控制器接收 SJA1000 发出的一个发送中断信号
			0	禁能;微控制器不从 SJA1000 接收发送中断信号
CR.1	RIE	接收中断使能	1	使能;信息被无错接收时,SJA1000 发出一个接收中断信号到微控制器
			0	禁能;微控制器不从 SJA1000 接收发送中断信号
CR.0	RR	复位请求;注4	1	当前;SJA1000 检测到复位请求后,中止当前发送/接收的信息,进入复位模式
			0	空缺;复位请求位接收到一个下降沿后,SJA1000 回到工作模式

注:

1. 控制寄存器的任何写访问都将设置该位为逻辑‘0’(复位)。

2. 在 PCA82C200 中这一位是用来选择同步模式的。因为这种模式不在使用了，所以这一位的设置不会影响微控制器。为了软件上的兼容，这一位是可以被设置的。硬件或软件复位后不改变这一位。它只反映用户软件写入的值。

3. 读此位的值总是逻辑‘1’。

4. 在硬启动或总线状态位设置为‘1’（总线关闭）时，复位请求位被置为 1（当前）。如果这些位被软件访问，其值将发生变化，而且会影响内部时钟的下一个上升沿（内部时钟的频率是外部晶振的 1/2）。在外部复位期间，微控制器不能把复位请求位置为 0（空缺）。如果把复位请求位设为 0，微控制器就必须检查这一位以保证外部复位引脚不保持为低。复位请求位的变化是同内部分频时钟同步的。读复位请求位能够反映出这种同步状态。

复位请求位被设为 0 后，SJA1000 将会等待：

a) 一个总线空闲信号（11 个弱势位），如果前一次复位请求是硬件复位或 CPU 初始复位。

b) 128 个总线空闲，如果前一次复位请求是 CAN 控制器在重新进入总线开启模式前初始化总线造成的；必须说明的是，如果复位请求位被置位，一些寄存器的值会被改变的（见表 2）。

6.3.4 命令寄存器（CMR）

命令位初始化 SJA1000 传输层上的动作。命令寄存器对微控制器来说是只写存储器。如果去读这个地址，返回值是‘1111 1111’。两条命令之间至少有一个内部时钟周期。内部时钟的频率是外部振荡频率的 1/2。

表 4 命令寄存器各位的功能说明（CMR）；CAN 地址 1

位	符号	名称	值	功能
CMR.7	-	-	-	保留
CMR.6	-	-	-	保留
CMR.5	-	-	-	保留
CMR.4	GTS	睡眠;注1	1	睡眠;如果没有 CAN 中断等待和总线活动,SJA1000进入睡眠模式
			0	唤醒;SJA1000正常工作模式
CMR.3	CDO	清除数据溢出;注2	1	清除;清除数据溢出状态位
			0	无动作
CMR.2	RRB	释放接收缓冲器;注3	1	释放;接收缓冲器中存放信息的内存空间将被释放
			0	无动作
CMR.1	AT	中止发送;注4	1	当前;如果不是在处理过程中,等待处理的发送请求将取消
			0	空缺;无动作
CMR.0	TR	发送请求;注5	1	当前;信息被发送
			0	空缺;无动作

注：

1. 将睡眠模式位置为 1，SJA1000 进入睡眠模式；没有总线活动，没有中断等待。至少破坏这两种情况之一时将会导致 GTS 的唤醒中断。设置成睡眠模式后，CLKOUT 信号持续至少 15 位的时间，以使被这个信号锁定的微控制器在 CLKOUT 信号变低之前进入准备模式。

如果前面提到的三种条件之一被破坏，SJA1000 将被唤醒：GTS 位被置为低后，总线转入活动或/INT 有效（低电平）。一旦唤醒，振荡器就将启动而且产生一个唤醒中断。因为总线活动而唤醒的 SJA1000 直到检测到 11 个连续的弱势位（总线空闲序列）才能够接收到这个信息。在复位模式中，GTS 位是不能被置位的。在清除复位请求后，且再一次检测到总线空闲，GTS 位才可以被置位。

2. 这个命令位是用来清除由数据溢出状态位指出的数据溢出情况的。如果数据溢出位被置位，就不会产生数据溢出中断了。在释放接收缓冲器命令的同时是可以发出清除数据溢出命令的。

3. 读接收缓冲器之后，微控制器可以通过设置释放接收缓冲器位为 1 来释放 RXFIFO 中当前信息的内存空间。这可能会导致接收缓冲器中的另一条信息立即有效。这样会再产生一次接收中断（使能条件下）。如果没有其它可用信息，就不会再产生接收中断，接收缓冲器状态位被清除。
4. 中止传送位是在 CPU 要求当前传送暂停时使用的，例如：传送一条紧急信息。正在进行的传送是不停止的。要查看原始信息是否被成功发送，可以通过传送成功状态位来检测。不过，这必须在发送缓冲器状态位为 1（释放）或发送中断产生的情况下才能实现。
5. 如果发送请求在前面的命令中被置位，它就不可以通过直接设置为 0 来取消它了。不过，可以通过设置中止发送位为 0 来取消。

6.3.5 状态寄存器（SR）

状态寄存器的内容反映了 SJA1000 的状态。状态寄存器对微控制器来说是只读存储器。

表 5 状态寄存器各位的功能说明： CAN 地址 2

位	符号	名称	值	功能
SR.7	BS	总线状态; 注 1	1	总线关闭;SJA1000退出总线活动
			0	总线开启;SJA1000加入总线活动
SR.6	ES	出错状态;注 2	1	出错;至少出现一个错误计数器满或超过 CPU 报警限制
			0	ok;两个错误计数器都在报警限制以下
SR.5	TS	发送状态;注 3	1	发送;SJA1000在传送信息
			0	空闲;没有要发送的信息
SR.4	RS	接收状态;注 3	1	接收;SJA1000正在接收信息
			0	空闲;没有正在接收的信息
SR.3	TCS	发送完毕状态;注 4	1	完毕;最近一次发送请求被成功处理
			0	未完毕;当前发送请求未处理完毕
SR.2	TBS	发送缓冲器状态;注 5	1	释放;CPU可以向发送缓冲器写信息
			0	锁定;CPU不能访问发送缓冲器;有信息正在等待发送或正在发送
SR.1	DOS	数据溢出状态;注 6	1	溢出;信息丢失,因为RXFIFO中没有足够的空间来存储它
			0	空缺;自从最后一次清除数据溢出命令执行,无数据溢出发生
SR.0	RBS	接收缓冲器状态;注 7	1	满;RXFIFO中有可用信息
			0	空;无可用信息

注:

1. 当传输错误计数器超过限制（255）（总线状态位置 1—总线关闭），CAN 控制器就会将复位请求位置 1（当前），在错误中断允许的情况下，会产生一个错误中断。这种状态会持续直到 CPU 清除复位请求位。所有这些完成之后，CAN 控制器将会等待协议规定的最小时间（128 个总线空闲信号）。总线状态位被清除后（总线开启），错误状态位被置为 0（ok），错误计数器复位且产生一个错误中断（中断允许）。
2. 根据 CAN 2.0B 协议说明，在接收或发送时检测到错误会影响错误计数。当至少有一个错误计数器满或超出 CPU 警告限制（96）时，错误状态位被置位。在允许情况下，会产生错误中断。
3. 如果接收状态位和发送状态位都是 0，则 CAN 总线是空闲的。
4. 无论何时发送请求位被置为 1，发送完毕位都会被置为 0（未完毕）。发送完毕位的 0 会一直保持到信息被成功发送。
5. 如果 CPU 在发送缓冲器状态位是 0（锁定）时试图写发送缓冲器，则写入的字节被拒绝接收且会在无任何提示的情况下丢失。
6. 当要被接收的信息成功的通过验收滤波器后（例如，仲裁后之初），CAN 控制器需要在 RXFIFO 中用一些空间来存储这条信息的描述符。因此必须有足够的空间来存储接收的每一个数据字节。如

果没有足够的空间存储信息，信息将会丢失且只向 CPU 提示数据溢出情况。如果这个接收到的信息除了最后一位之外都无错误，信息有效。

7. 在读 RXFIFO 中的信息且用释放接收缓冲器命令来释放内存空间之后，这一位被清除。如果 FIFO 中还有可用信息，此位将在下一位的时限 (t_{SCL}) 中被重新设置。

6.3.6 中断寄存器 (IR)

中断寄存器允许中断源的识别。当寄存器的一位或多位被置位时，/INT（低电平有效）引脚就被激活了。寄存器被微控制器读过之后，所有位复位，这导致了/INT 引脚上的电平漂移。中断寄存器对微控制器来说是只读存储器。

表 6 中断寄存器各位的功能说明；CAN 地址 3

位	符号	名称	值	功能
IR.7	-	-	-	保留;注1
IR.6	-	-	-	保留;注1
IR.5	-	-	-	保留;注1
IR.4	WUI	唤醒中断;注2	1	置位;退出睡眠模式时此位被置位
			0	复位; 微控制器的任何读访问将清除此位
IR.3	DOI	数据溢出中断;注3	1	设置;当数据溢出中断使能位被置为1时向数据溢出状态位 '0-1' 跳变, 此位被置位
			0	复位; 微控制器的任何读访问将清除此位
IR.2	EI	错误中断	1	置位;错误中断使能时,错误状态位或总线状态位的变化会置位此位
			0	复位; 微控制器的任何读访问将清除此位
IR.1	TI	发送中断	1	置位;发送缓冲器状态从0变为1(释放)和发送中断使能时,置位此位
			0	复位; 微控制器的任何读访问将清除此位
IR.0	RI	接收中断;注4	1	置位;当接收FIFO不空和接收中断使能时置位此位
			0	复位; 微控制器的任何读访问将清除此位

注:

1. 读这一位的值总是 1。
2. 如果当 CAN 控制器参与总线活动或 CAN 中断正在等待时，CPU 试图进入睡眠模式，唤醒中断也会产生的。
3. 溢出中断位（中断允许情况下）和溢出状态位是同时被置位的。
4. 接收中断位（中断允许时）和接收缓冲器状态位是同时置位的。

必须说明的是接收中断位在读的时候被清除，即使 FIFO 中还有其它可用信息。一旦释放接收缓冲器命令执行后，接收缓冲器中还有其它可用信息，接收中断（中断允许时）会在下一个 t_{SCL} 被重置。

6.3.7 发送缓冲区列表

发送缓冲区的全部内容列表如表 7。缓冲器是用来存储微控制器要 SJA1000 发送的信息的。它被分为描述符区和数据区。发送缓冲器的读/写只能由微控制器在工作模式下完成。在复位模式下读出的值总是 'FFH'。

表 7 发送缓冲器列表

CAN地址	区	名称	位							
			7	6	5	4	3	2	1	0
10	描述符	识别码字节1	ID.10	ID.9	ID.8	ID.7	ID.6	ID.5	ID.4	ID.3
11		识别码字节2	ID.2	ID.1	ID.0	RTR	DLC.3	DLC.2	DLC.1	DLC.0
12	数据	TX数据1	发送数据字节1							
13		TX数据2	发送数据字节2							

14	TX数据3	发送数据字节3
15	TX数据4	发送数据字节4
16	TX数据5	发送数据字节5
17	TX数据6	发送数据字节6
18	TX数据7	发送数据字节7
19	TX数据8	发送数据字节8

6.3.7.1 识别码 (ID)

识别码有 11 位 (ID0-ID10)。ID10 是最高位，在仲裁过程中是最先被发送到总线上的。识别码就象信息的名字。它在接收器的验收滤波器中被用到，也在仲裁过程中决定总线访问的优先级。识别码的值越低，其优先级越高。这是因为在仲裁时有许多支配（控制）位开头的字节。

6.3.7.2 远程发送请求 (RTR)

如果此位置 1，总线将以远程帧发送数据。这意味着此段中没有数据字节。尽管如此，也需要同识别码相同的数据帧来识别正确的数据长度。

如果 RTR 位没有被置位，数据将以数据长度码规定的长度来传送。

6.3.7.3 数据长度码 (DLC)

信息数据区的字节数根据数据长度码编制。在远程帧传送中，因为 RTR 被置位，数据长度码是不被考虑的。这就迫使发送/接收数据字节数为 0。总之，数据长度码必须正确设置以避免两个 CAN 控制器用同样的识别机制启动远程帧传送而发生总线错误。数据字节数是 0-8，是以如下方法计算的：

数据字节数 = $8 \times \text{DLC.3} + 4 \times \text{DLC.2} + 2 \times \text{DLC.1} + \text{DLC.0}$

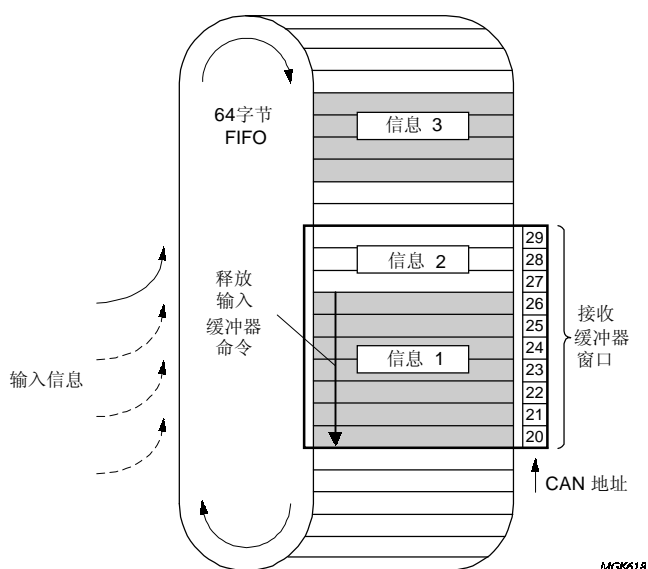
为了保持兼容性，数据长度码不超过 8。如果选择的值超过 8，则按照 DLC 规定的 8 字节发送。

6.3.7.4 数据区

传送的数据字节数由数据长度码决定。发送的第一位是地址 12 单元的数据字节 1 的最高位。

6.3.8 接收缓冲器

接收缓冲器的全部列表和发送缓冲器类似。接收缓冲器是 RXFIFO 中可访问的部分，位于 CAN 地址的 20-29 之间。



接收缓冲器当前的可用信息是信息 1

图 4 在 RXFIFO 中储存信息的例子

识别码、远程发送请求位和数据长度码同发送缓冲器的相同，只不过是在地址 20-29。如图 4 所示，RXFIFO 共有 64 字节的信息空间。在任何情况下，FIFO 中可以存储的信息数取决于各条信息的长度。如

果 RXFIFO 中没有足够的空间来存储新的信息，CAN 控制器会产生数据溢出。数据溢出发生时，已部分写入 RXFIFO 的当前信息将被删除。这种情况将通过状态位或数据溢出中断（中断允许时，如果除了最后一位整个数据块被无误接收也使 RX 信息有效）反应到微控制器。

6.3.9 验收滤波器

在验收滤波器的帮助下，CAN 控制器能够允许 RXFIFO 只接收同识别码和验收滤波器中预设值相一致的信息。验收滤波器通过验收代码寄存器（ACR；见 6.3.9.1 节）和验收屏蔽寄存器（AMR；见 6.3.9.2 节）来定义。

6.3.9.1 验收代码寄存器（ACR）

表 8 ACR 的位分配；CAN 地址 4

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
AC.7	AC.6	AC.5	AC.4	AC.3	AC.2	AC.1	AC.0

复位请求位被置高（当前）时，这个寄存器是可以访问（读/写）的。如果一条信息通过了验收滤波器的测试而且接收缓冲器有空间，那么描述符和数据将被分别顺次写入 RXFIFO。当信息被正确的接收完毕，就会：

- 接收状态位置高（满）
- 接收中断使能位置高（使能）接收中断置高（产生中断）。

验收代码位（AC.7-AC.0）和信息识别码的高 8 位（ID.10-ID.3）相等，且与验收屏蔽位（AM.7-AM.0）的相应位相或为 1。即如果满足以下方程的描述，则被接收：

$$[(ID.10-ID.3) \equiv (AC.7-AC.0)] \vee (AM.7-AM.0) \equiv 11111111$$

6.3.9.2 验收屏蔽寄存器（AMR）

表 9 AMR 位配置；CAN 地址 5

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
AM.7	AM.6	AM.5	AM.4	AM.3	AM.2	AM.1	AM.0

如果复位请求位置高（当前），这个寄存器可以被访问（读/写）。验收屏蔽寄存器定义验收代码寄存器的相应位对验收滤波器是“相关的”或“无影响的”（即可为任意值）。

6.3.9.3 其它寄存器

其它寄存器见 6.5 节。

6.4 PeliCAN 模式

6.4.1 PeliCAN 地址列表

CAN 控制寄存器的内部寄存器对 CPU 来说是以外部寄存器形式存在而作片内内存使用。因为 CAN 控制器可以工作于不同模式（工作/复位；见 6.4.3 节），所以必须区分不同的内部地址定义。从 CAN 地址 32 起所有的内部 RAM（80 字节）被映象为 CPU 的接口。

表 10 PeliCAN 地址分配；注 1

CAN地址	工作模式		复位模式	
	读	写	读	写
0	模式	模式	模式	模式
1	(00H)	命令	(00H)	命令
2	状态	-	状态	-
3	中断	-	中断	-
4	中断使能	中断使能	中断使能	中断使能
5	保留(00H)	-	保留(00H)	-
6	总线定时0	-	总线定时0	总线定时0

7	总线定时1	-	总线定时1	总线定时1
8	输出控制	-	输出控制	输出控制
9	检测	检测;注2	检测	检测;注2
10	保留(00H)	-	保留(00H)	-
11	仲裁丢失捕捉	-	仲裁丢失捕捉	-
12	错误代码捕捉	-	错误代码捕捉	-
13	错误报警限制	-	错误报警限制	错误报警限制
14	RX错误计数器	-	RX错误计数器	RX错误计数器
15	TX错误计数器	-	TX错误计数器	TX错误计数器
16	RX 帧信息 SFF;注3	RX 帧信息 EFF;注4	TX 帧信息 SFF;注3	TX 帧信息 EFF;注4
17	RX识别码1	RX识别码1	TX识别码1	TX识别码1
18	RX识别码2	RX识别码2	TX识别码2	TX识别码2
19	RX数据1	RX识别码3	TX数据1	TX识别码3
20	RX数据2	RX识别码4	TX数据2	TX识别码4
21	RX数据3	RX数据1	TX数据3	TX数据1
22	RX数据4	RX数据2	TX数据4	TX数据2
23	RX数据5	RX数据3	TX数据5	TX数据3
24	RX数据6	RX数据4	TX数据6	TX数据4
25	RX数据7	RX数据5	TX数据7	TX数据5
26	RX数据8	RX数据6	TX数据8	TX数据6
27	(FIFO RAM);注5	RX数据7	-	TX数据7
28	(FIFO RAM);注5	RX数据8	-	TX数据8
29	RX信息计数器	-	RX信息计数器	-
30	RX缓冲器起始地址	-	RX缓冲器起始地址	RX缓冲器起始地址
31	时钟分频器	时钟分频器;注6	时钟分频器	时钟分频器
32	内部RAM地址0(FIFO)	-	内部RAM地址0	内部RAM地址0
33	内部RAM地址1(FIFO)	-	内部RAM地址1	内部RAM地址1
↓	↓	↓	↓	↓
95	内部RAM地址63(FIFO)	-	内部RAM地址63	内部RAM地址63
96	内部RAM地址64(TX缓冲器)	-	内部RAM地址64	内部RAM地址64
↓	↓	↓	↓	↓
108	内部RAM地址76(TX缓冲器)	-	内部RAM地址76	内部RAM地址76
109	内部RAM地址77(空闲)	-	内部RAM地址77	内部RAM地址77
110	内部RAM地址78(空闲)	-	内部RAM地址78	内部RAM地址78
111	内部RAM地址79(空闲)	-	内部RAM地址79	内部RAM地址79
112	(00H)	-	(00H)	-
↓	↓	↓	↓	↓
127	(00H)	-	(00H)	-

注:

1. 必须说明的是在 CAN 的高端地址区的寄存器是重复的(CPU 地址的高 8 位是不参与解码的: CAN

地址 128 和地址 0 是连续的)。

2. 测试寄存器只用于产品测试。正常工作时使用这个寄存器会使设备产生不可预料的行为。
3. SFF=标准帧格式
4. EFF=扩展帧格式
5. 这些地址分配反映当前信息之后的 FIFO RAM 空间。上电后的内容是随机的且包含了当前接收信息的下一条信息的开头。如果没有信息要接收，这里会出现部分旧的信息。
6. 一些位在复位模式中是只写的 (CAN 模式、CBP、RXINTEN 和时钟关闭)。

6.4.2 复位值

检测到复位模式设置位后，中止当前发送/接收信息而进入复位模式。当向复位模式位 ‘1-0’ 跳变时，CAN 控制器回到模式寄存器所定义的模式。

表 11 复位模式配置；注 1 和注 2

寄存器	位	符号	名称	值	
				硬件复位	软件设置MOD. 0 或总线关闭时 复位
模式	MOD.7-5	-	保留	0(保留)	0(保留)
	MOD.4	SM	睡眠	0(唤醒)	0(唤醒)
	MOD.3	AFM	验收滤波器	0(双向)	×
	MOD.2	STM	自检测模式	0(正常)	×
	MOD.1	LOM	只听模式	0(正常)	×
	MOD.0	RM	复位模式	1(当前)	1(当前)
命令	CMR.7-5		保留	0(保留)	0(保留)
	CMR.4	SRR	自接收模式	0(空缺)	0(空缺)
	CMR.3	CDO	清除数据溢出	0(无动作)	0(无动作)
	CMR.2	RRB	释放接收缓冲器	0(无动作)	0(无动作)
	CMR.1	AT	中止发送	0(空缺)	0(空缺)
	CMR.0	TR	发送请求	0(空缺)	0(空缺)
状态	SR.7	BS	总线状态	0(总线开启)	×
	SR.6	ES	出错状态	0(ok)	×
	SR.5	TS	发送状态	1(等待空闲)	1(等待空闲)
	SR.4	RS	接收状态	1(等待空闲)	1(等待空闲)
	SR.3	TCS	发送完毕状态	1(完毕)	×
	SR.2	TBS	发送缓冲器状态	1(释放)	1(释放)
	SR.1	DOS	数据溢出状态	0(空缺)	0(空缺)
	SR.0	RBS	接收缓冲器状态	0(空)	0(空)
中断	IR.7	BEI	总线出错状态	0(复位)	0(复位)
	IR.6	ALI	仲裁丢失中断	0(复位)	0(复位)
	IR.5	EPI	错误消极中断	0(复位)	0(复位)
	IR.4	WUI	唤醒中断	0(复位)	0(复位)
	IR.3	DOI	数据溢出中断	0(复位)	0(复位)
	IR.2	EI	错误警报中断	0(复位)	×;注3
	IR.1	TI	发送中断	0(复位)	0(复位)

	IR.0	RI	接收中断	0(复位)	0(复位)
中断使能	IER.7	BEIE	总线错误中断使能	×	×
	IER.6	ALIE	仲裁丢失中断使能	×	×
	IER.5	EPIE	错误消极中断使能	×	×
	IER.4	WUIE	唤醒中断使能	×	×
	IER.3	DOIE	数据溢出中断使能	×	×
	IER.2	EIE	错误报警中断使能	×	×
	IER.1	TIE	发送中断使能	×	×
	IER.0	RIE	接收中断使能	×	×
总线定时0	BTR0.7	SJW.1	同步跳转宽度1	×	×
	BTR0.6	SJW.0	同步跳转宽度0	×	×
	BTR0.5	BRT.5	波特率预设值5	×	×
	BTR0.4	BRT.4	波特率预设值4	×	×
	BTR0.3	BRT.3	波特率预设值3	×	×
	BTR0.2	BRT.2	波特率预设值2	×	×
	BTR0.1	BRT.1	波特率预设值1	×	×
	BTR0.0	BRT.0	波特率预设值0	×	×
总线定时1	BTR1.7	SAM	采样	×	×
	BTR1.6	TSEG2.2	时间段2.2	×	×
	BTR1.5	TSEG2.1	时间段2.1	×	×
	BTR1.4	TSEG2.0	时间段2.0	×	×
	BTR1.3	TSEG1.3	时间段1.3	×	×
	BTR1.2	TSEG1.2	时间段1.2	×	×
	BTR1.1	TSEG1.1	时间段1.1	×	×
	BTR1.0	TSEG1.0	时间段1.0	×	×
输出控制	OCR.7	OCTP1	输出控制晶体管P1	×	×
	OCR.6	OCTN1	输出控制晶体管N1	×	×
	OCR.5	OCPOL1	输出控制极性1	×	×
	OCR.4	OCRP0	输出控制晶体管P0	×	×
	OCR.3	OCTN0	输出控制晶体管N0	×	×
	OCR.2	OCPOL0	输出控制极性0	×	×
	OCR.1	OCMODE1	输出控制模式1	×	×
	OCR.0	OCMODE0	输出控制模式0	×	×
仲裁丢失捕捉	-	ALC	仲裁丢失捕捉	0	×
错误代码捕捉	-	ECC	错误代码捕捉	0	×
错误报警限制	-	EWLR	错误报警限制寄存器	96	×
RX错误计数器	-	RXERR	接收错误计数器	0(复位)	×;注4
TX错误计数器	-	TXERR	发送错误计数器	0(复位)	×;注4
TX缓冲器	-	TXB	发送缓冲器	×	×

RX缓冲器	-	RXB	接收缓冲器	×;注5	×;注5
ACR0-3	-	ACR0-ACR3	验收代码寄存器	×	×
AMR0-3	-	AMR0-AMR3	验收屏蔽寄存器	×	×
RX信息计数器	-	RMC	RX信息计数器	0	0
RX缓冲器起始地址	-	RBSA	RX缓冲器起始地址	0000 0000	×
时钟分频器	-	CDR	时钟分频器地址	0000 0000Intel; 0000 0101Motorola	×

注:

1. ×表示这些寄存器或位是何值是无任何影响的。
2. 括号中是功能意义的解释。
3. 在相应的中断允许时, 总线关闭则错误报警中断被置位。
4. 若是因为总线关闭而进入复位模式, 接收错误计数器被清 0, 发送错误计数器被初始化到 127 以计数 CAN 定义的包括 128 个 11 位连续隐藏(弱势)位的总线关闭恢复时间。
5. RXFIFO 的内部读/写指针复位到初始化值。连续的读 RXB 口将会得到一些未定义的值(一部分是老的信息)。如果有信息被发送, 就被并行写入接收缓冲器。只有这次传送是自接收请求引起的才会产生接收中断。所以, 即使接收缓冲器是空的, 最后一次发送的信息也可以从接收缓冲器中读出, 除非它被下一条要发送或接收的信息覆盖。硬件复位时, RXFIFO 的指针指向物理 RAM 地址 '0'。通过软件设置 CR.0 或总线关闭会使 RXFIFO 的指针指向当前有效 FIFO 的起始地址(RBSA 寄存器), 这个地址不同于第一次释放接收缓冲器命令后的 RAM 地址 '0'。

6.4.3 模式寄存器 (MOD)

模式寄存器的内容是用来改变 CAN 控制器的行为的。CPU 把控制寄存器作为读/写寄存器, 可以设置这些位。保留位读值为逻辑 0。

表 12 模式寄存器的各位的功能; CAN 地址 '0'

位	符号	名称	值	功能
MOD.7	-	-	-	保留
MOD.6	-	-	-	保留
MOD.5	-	-	-	保留
MOD.4	SM	睡眠模式;注1	1	睡眠;没有CAN中断等待和总线活动时,CAN控制器进入睡眠模式
			0	唤醒;从睡眠状态唤醒
MOD.3	AFM	验收滤波器模式;注2	1	单;选择单个验收滤波器(32位长度)
			0	双;选择两个验收滤波器(每个有16位激活)
MOD.2	STM	自检测模式;注2	1	自检测;此模式可以检测所有节点,没有任何活动的节点使用自接收命令;即使没有应答,CAN控制器也会成功发送
			0	正常模式;成功发送时必需应答信号
MOD.1	LOM	只听模式;注2和3	1	只听;这种模式中,即使成功接收信息,CAN控制器也不向总线发应答信号;错误计数器停止在当前值.
			0	正常模式
MOD.0	RM	复位模式;注4	1	复位;检测到复位模式位被置位,中止当前正在接收/发送的信息,进入复位模式

		0	正常;复位模式位接收到'1-0'的跳变后,CAN控制器回到工作模式
--	--	---	-----------------------------------

注:

- 睡眠模式位设为 1 (sleep), SJA1000 将进入睡眠模式; 没有总线活动和中断等待。至少破坏这两种情况之一时将会导致 SM 产生唤醒中断。设置为睡眠模式后, CLKOUT 信号持续至少 15 位的时间, 以允许主微控制器在 CLKOUT 信号电平变低而被锁住之前进入准备模式。前面提到的三种条件之一被破坏时, SJA1000 将被唤醒: SM 电平设为低(唤醒)之后, 总线进入活动状态或/INT 被激活(变低)。唤醒后, 振荡器启动且产生一个唤醒中断。由于总线活动唤醒的直到检测到 11 个连续的隐藏(弱势)位(总线空闲序列)后才能接收这条信息。注意在复位模式中是不能设置 SM 的。清除复位模式后, 再一次检测到总线空闲时, SM 的设置才开始有效。
- 如果先进入复位模式, MOD.1-MOD.3 是只写的。
- 这种工作模式使 CAN 控制器进入错误消极状态。信息传送是不可能的。以软件驱动的位速检测和“热插”时可使用只听模式。所有其它功能都能象在正常工作模式中一样使用。
- 在硬件复位或总线状态位为 1 (总线关闭)时, 复位模式位也被置为 1 (当前)。如果通过软件访问这一位, 值将发生变化且下一个内部时钟(频率为外部振荡器的 1/2)的上升沿有效。在外部复位期间, 微控制器不能将复位模式位设置为 0 (空闲)。因此, 将复位模式位设为 1 后, 微控制器必须检查此位以确保外部复位引脚上不保持高。复位请求位的改变和内部分频时钟同步。读复位请求位能够反映出这种同步状态。复位模式位为 0 后, CAN 控制器会等待:
 - 一个总线空闲信号(11 个隐藏(弱势)位), 如果上一次复位是硬件复位或 CPU 初始复位。
 - 128 个总线空闲, 如果上一次复位是 CAN 控制器在重新进入总线开启之前初始化复位。

6.4.4 命令寄存器 (CMR)

命令位初始化 CAN 控制器传输层的一个动作。这个寄存器是只写的, 所有位的读出值都是逻辑 0。因处理的需要, 两条命令之间至少有一个内部时钟周期。内部时钟周期的频率是外部振荡器的一半。

表 13 命令寄存器 (CMR) 各位的功能说明; CAN 地址 1

位	符号	名称	值	功能
CMR.7	-	保留	-	-
CMR.6	-	保留	-	-
CMR.5	-	保留	-	-
CMR.4	SRR	自接收请求;注1和2	1	当前;信息可被同时发送和接收
			0	-(空缺)
CMR.3	CDO	清除数据溢出;注4	1	清除;数据溢出状态位被清除
			0	-(无动作)
CMR.2	RRB	释放接收缓冲器;注4	1	释放;接收缓冲器FXFIFO中载有信息的内存空间被释放
			0	-(无动作)
CMR.1	AT	中止发送;注5和2	1	当前;如果不是正在处理,等待中的发送请求被取消
			0	-(空缺)
CMR.0	TR	发送请求;注6和2	1	当前;信息被发送
			0	-(空缺)

注:

- 如果验收滤波器已设置了相应的识别码, 当发送自接收请求信息时同时开始接收。接收和发送中断对自接收是有效的(模式寄存器的自检测模式也有类似情况)。
- 设置命令位 CMR.0 和 CM.1 会立即产生一次信息发送。当发生错误或仲裁丢失时是不会重发的(单次发送)。设置命令位 CMR.4 和 CMR.1 会立即产生一次自接收性质的信息发送。发生错误或仲裁丢失时是不会重发的。设置命令位 CMR.0、CMR.1 和 CMR.4 会立即产生一个信息发送(见 CMR.0

和 CMR.1 的定义)。

一旦状态寄存器的发送状态位被置位，内部发送请求就被自动清除。

同时设置 CMR.0 和 CMR.4 会忽略 CMR.4 位。

3. 这个命令位用于清除数据溢出位指出的数据溢出情况。如果数据溢出位被置位就不会再有数据溢出中断产生。

4. 读接收缓冲器之后，CPU 可以通过设置释放接收缓冲器位为 1 来释放 RXFIFO 的内存空间。这样就会导致接收缓冲器内的另一条信息立即有效。如果没有其它有用的信息，就复位接收中断。

5. 当 CPU 需要当前请求发送等待时，例如：先发送一条比较紧急的信息时。但当前正在处理的传送是不停止的。要想知道源信息是否成功发送，可以通过传送完毕状态位来查看。不过，这应在在发送缓冲器状态位置 1 或产生发送中断后。

要注意的是，即使因为发送缓冲器状态位变为‘释放’而使信息被中止，也会产生发送中断。

6. 如果前一条指令中发送请求被置为 1，它不能通过设置发送请求位为 0 来取消，而应通过中止发送位为 0 取消。

6.4.5 状态寄存器 (SR)

状态寄存器反映 CAN 控制器的状态。状态寄存器对 CPU 来说是只读内存。

表 14 状态寄存器的各位功能说明；CAN 地址 2

位	符号	名称	值	功能
SR.7	BS	总线状态;注1	1	总线关闭;CAN控制器不参与总线活动
			0	总线开启;CAN控制器参与总线活动
SR.6	ES	出错状态;注2	1	出错;至少一个错误计数器满或超过了由错误报警限制寄存器(EWLR)定义的CPU报警限制
			0	ok;两个错误计数器都在报警限制以下
SR.5	TS	发送状态;注3	1	发送;CAN控制器正在发送信息
			0	空闲
SR.4	RS	接收状态;注3	1	接收;CAN控制器正在接收信息
			0	空闲
SR.3	TCS	发送完毕状态;注4	1	完毕;最后一次发送已被成功处理
			0	未完;当前请求的发送未处理完
SR.2	TBS	发送缓冲器状态;注5	1	释放;CPU可以向发送缓冲器中写信息
			0	锁定;CPU不能访问发送缓冲器;信息不是在等待发送也不是正在发送.
SR.1	DOS	数据溢出状态;注6	1	溢出;信息因RXFIFO中无足够的存储空间而丢失
			0	空缺;自从上一次执行清除数据溢出命令以来无数据溢出发生
SR.0	RBS	接收缓冲器状态;注7	1	满;RXFIFO中有可用信息
			0	空;无可用信息

注:

1. 当发送错误计数器超过限制 (255)，总线状态位被置为 1 (总线关闭)，CAN 控制器将设置复位模式位为 1 (当前) 而且产生一个错误报警中断 (相应的中断允许时)。发送错误计数器被置为 127，接收错误计数器被清除。这种模式将会保持直到 CPU 将复位模式位清除。完成这些之后，CAN 控制器将通过发送错误计数器的减 1 计数以等待协议规定的最少时间 (128 个总线空闲信号)。之后总线状态位被清除 (总线开启)，错误状态位被置为 0 (ok)，错误计数器复位且产生一个错误报警中断 (中断允许时)。这期间读 TX 错误计数器给出关于总线关闭修复的状态信息。
2. 根据 CAN 2.0B 协议规定，在接收和发送期间检测到错误会影响错误计数器。至少有一个错误计数器满或超过 CPU 报警限制 (EWLR) 时错误状态位被置位。中断允许时，会产生错误报警中断。

EWLR 硬件复位后的默认值是 96。

3. 如果接收状态位和发送状态位都是 0（空闲），则 CAN 总线是空闲的。如果这两位都是 1，则控制器正在等待下一次空闲。硬件启动后，直到空闲状态到来必须检测到 11 个连续的隐藏（弱势）位。总线关闭后会产生 128 个 11 位的连续隐藏（弱势）位。
4. 一旦发送请求位或自接收请求位被置 1，发送成功状态位就会被置 0（不成功）。发送成功状态位会保持为 0 直到发送成功。
5. 如果 CPU 试图在发送缓冲器状态位是 0（锁定）时向发送缓冲器写，写入的字节将不被接受且在没有任何提示的情况下丢失。
6. 当要接收的信息已经成功通过验收滤波器的时候，CAN 控制器需要在 RXFIFO 中有足够的空间来存储信息描述符和每一个接收的数据字节。如果没有足够的空间来存储信息，信息就会丢失，在信息变为无效时向 CPU 提示数据溢出。如果信息没有被成功接收（例如，由于错误），就没有数据溢出情况提示。
7. 读出 RXFIFO 中的所有信息和用释放接收缓冲器命令释放它们的内存空间之后，此位被清除。

6.4.3 中断寄存器（IR）

中断寄存器允许中断源的识别。当这个寄存器的一位或多位被置位时，CAN 中断将反映到 CPU。CPU 读此寄存器的时候，除了接收中断外的所有位都被复位。

中断寄存器对 CPU 来说是只读存储器。

表 15 中断寄存器（IR）的位功能说明；CAN 地址 3

位	符号	名称	值	功能
IR.7	BEI	总线错误中断	1	置位;当CAN控制器检测到总线错误且中断使能寄存器中的BEIE被置位时此位被置位
			0	复位
IR.6	ALI	仲裁丢失中断	1	置位;当CAN控制器丢失仲裁,变为接收器和中断使能寄存器的ALIE为被置位时,此位被置位
			0	复位
IR.5	EPI	错误消极中断	1	置位;当CAN控制器到达错误消极状态(至少一个错误计数器超过协议规定的值127)或从错误消极状态又进入错误活动状态以及中断寄存器的EPIE位被置位时此位被置1
			0	复位
IR.4	WUI	唤醒中断;注1	1	置位;当CAN控制器在睡眠模式中检测到总线的活动且中断寄存器的WUIE位被置1时此位被置位
			0	复位
IR.3	DOI	数据溢出中断	1	置位;数据溢出状态位有‘0-1’跳变且中断寄存器的DOIE位被置位时此位被置1
			0	复位
IR.2	EI	错误报警中断	1	置位;错误状态位和总线状态位的改变和中断寄存器的EIE位被置位时此位被置1
			0	复位
IR.1	TI	发送中断	1	置位;发送缓冲器状态从‘0-1’(释放)跳变且中断寄存器的TIE位被置位时此位被置1
			0	复位
IR.0	RI	接收中断;注2	1	置位;接收FIFO不空且中断寄存器的RIE位被置位时此位被置1
			0	复位;RXFIFO中无可用信息

注:

1. 如果 CPU 在 CAN 控制器参与总线活动或 CAN 中断正在等待时试图置位睡眠模式位也产生唤醒

中断。

2. 除了 RI 取决于相应的中断使能位 (RIE) 这一点外, 此位的行为和接收缓冲器状态位是等效的。所以读中断寄存器时接收中断位不被清除。释放接收缓冲器的命令可以临时清除 RI。如果执行释放命令后 FIFO 中还有可用信息, RI 被重新置位。否则 RI 保持清 0 状态。

6.4.7 中断使能寄存器 (IER)

这个寄存器能使不同类型的中断源对 CPU 有效。

这个寄存器对 CPU 来说是可读/写存储器。

表 16 中断使能寄存器 (IER) 的各位的功能说明; CAN 地址 4

位	符号	名称	值	功能
IER.7	BEIE	总线错误中断使能	1	使能;如果检测到总线错误,则CAN控制器请求相应的中断
			0	禁能
IER.6	ALIE	仲裁丢失中断使能	1	使能;如果CAN控制器已丢失了仲裁,则请求相应的中断
			0	禁能
IER.5	EPIE	错误消极中断使能	1	使能;若CAN控制器的错误状态改变(从消极到活动或反之), 则请求相应的中断
			0	禁能
IER.4	WUIE	唤醒中断使能	1	使能;如果睡眠模式中的CAN控制器被唤醒,则请求相应的中断
			0	禁能
IER.3	DOIE	数据溢出中断使能	1	使能;如果数据溢出状态位被置位(见状态寄存器;表14),CAN控制器请求相应的中断
			0	禁能
IER.2	EIE	错误报警中断使能	1	使能;如果错误或总线状态改变(见状态寄存器;表14),CAN控制器请求相应的中断
			0	禁能
IER.1	TIE	发送中断使能	1	使能;当信息被成功发送或发送缓冲器又可访问(例如,中止发送命令后)时,CAN控制器请求相应的中断
			0	禁能
IER.0	RIE	接收中断使能;注1	1	使能;当接收缓冲器状态是'满'时,CAN控制器请求相应的中断
			0	禁能

注:

1. 接收中断使能位对接收中断位和外部中断输出/INT 有直接的影响。如果 RIE 被清 0 且没有其它中断被挂起, 外部/INT 引脚电平会立即变高。

6.4.8 仲裁丢失捕捉寄存器 (ALC)

这个寄存器包括了仲裁丢失的位置的信息。仲裁丢失捕捉寄存器对 CPU 来说是只读存储器。保留位的读值为 0。

表 17 仲裁丢失捕捉寄存器 (ALC) 的各位功能说明; CAN 地址 11

位	符号	名称	值	功能
ALC.7-ALC.5	-	保留	值和功能见表18	
ALC.4	BITNO4	第四位		
ALC.3	BITNO3	第三位		
ALC.2	BITNO2	第二位		
ALC.1	BITNO1	第一位		
ALC.0	BITNO0	第零位		

仲裁丢失时，会产生相应的仲裁丢失中断（中断允许）。同时，位流处理器的当前位位置被捕捉送入仲裁丢失捕捉寄存器。一直到用户通过软件读这个值，寄存器中的内容都不会变。随后，捕捉机制又被激活了。

读中断寄存器时，中断寄存器中相应的中断标志位被清除。直到仲裁丢失捕捉寄存器被读一次之后，新的仲裁丢失中断才有效。

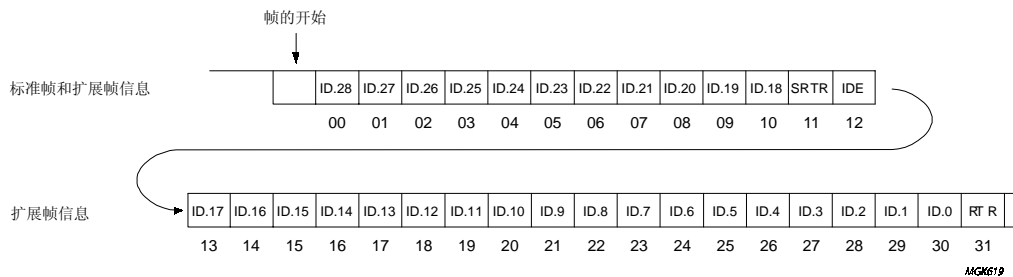


图5 仲裁丢失位解释

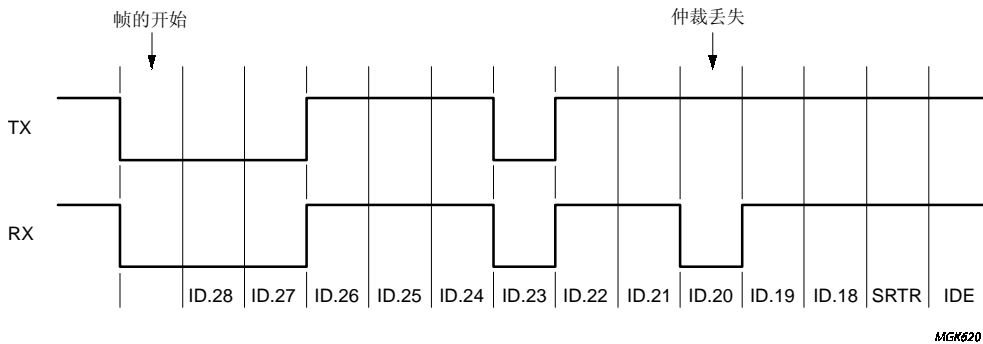


图6 仲裁丢失解释举例

表 18 仲裁丢失捕捉寄存器的 Bit4-Bit0 的功能

位					十进制值	功能
ALC. 4	ALC. 3	ALC. 2	ALC. 1	ALC. 0		
0	0	0	0	0	0	仲裁丢失在识别码的bit1
0	0	0	0	1	1	仲裁丢失在识别码的bit2
0	0	0	1	0	2	仲裁丢失在识别码的bit3
0	0	0	1	1	3	仲裁丢失在识别码的bit4
0	0	1	0	0	4	仲裁丢失在识别码的bit5
0	0	1	0	1	5	仲裁丢失在识别码的bit6
0	0	1	1	0	6	仲裁丢失在识别码的bit7
0	0	1	1	1	7	仲裁丢失在识别码的bit8
0	1	0	0	0	8	仲裁丢失在识别码的bit9
0	1	0	0	1	9	仲裁丢失在识别码的bit10
0	1	0	1	0	10	仲裁丢失在识别码的bit11
0	1	0	1	1	11	仲裁丢失在SRTR位;注2
0	1	1	0	0	12	仲裁丢失在IDE位

0	1	1	0	1	13	仲裁丢失在识别码的bit12;注3
0	1	1	1	0	14	仲裁丢失在识别码的bit13;注3
0	1	1	1	1	15	仲裁丢失在识别码的bit14;注3
1	0	0	0	0	16	仲裁丢失在识别码的bit15;注3
1	0	0	0	1	17	仲裁丢失在识别码的bit16;注3
1	0	0	1	0	18	仲裁丢失在识别码的bit17;注3
1	0	0	1	1	19	仲裁丢失在识别码的bit18;注3
1	0	1	0	0	20	仲裁丢失在识别码的bit19;注3
1	0	1	0	1	21	仲裁丢失在识别码的bit20;注3
1	0	1	1	0	22	仲裁丢失在识别码的bit21;注3
1	0	1	1	1	23	仲裁丢失在识别码的bit22;注3
1	1	0	0	0	24	仲裁丢失在识别码的bit23;注3
1	1	0	0	1	25	仲裁丢失在识别码的bit24;注3
1	1	0	1	0	26	仲裁丢失在识别码的bit25;注3
1	1	0	1	1	27	仲裁丢失在识别码的bit26;注3
1	1	1	0	0	28	仲裁丢失在识别码的bit27;注3
1	1	1	0	1	29	仲裁丢失在识别码的bit28;注3
1	1	1	1	0	30	仲裁丢失在识别码的bit29;注3
1	1	1	1	1	31	仲裁丢失在RTR位;注3

注:

1. 仲裁丢失的二进制编码结构位的号码。
2. 标准帧信息的 RTR 位。
3. 只使用于扩展帧信息。

6.4.9 错误代码捕捉寄存器 (ECC)

这个寄存器包含了总线错误的类型和位置信息。错误代码捕捉寄存器对 CPU 来说是只读内存。

表 19 错误代码捕捉寄存器的位功能说明; CAN 地址 12

位	符号	名称	值	功能
ECC.7 ⁽¹⁾	ERRC1	错误代码1	-	-
ECC.6 ⁽¹⁾	ERRC0	错误代码0	-	-
ECC.5	DIR	方向	1	RX;接收时发生的错误
			0	TX;发送时发生的错误
ECC.4 ⁽²⁾	SEG4	段4	-	-
ECC.3 ⁽²⁾	SEG3	段3	-	-
ECC.2 ⁽²⁾	SEG2	段2	-	-
ECC.1 ⁽²⁾	SEG1	段1	-	-
ECC.0 ⁽²⁾	SEG0	段0	-	-

注:

1. ECC.7 和 ECC.6 的解释见表 20。
2. ECC.4-ECC.0 的解释见表 21。

表 20 ECC.7 和 ECC.6 的功能说明

位ECC. 7	位ECC. 6	功能
0	0	位错
0	1	格式错
1	0	填充错
1	1	其它错误

表 21 ECC.4-ECC.0 的功能说明; 注 1

位ECC. 4	位ECC. 3	位ECC. 2	位ECC. 1	位ECC. 0	功能
0	0	0	1	1	帧开始
0	0	0	1	0	ID.28-ID.21
0	0	1	1	0	ID.20-ID.18
0	0	1	0	0	SRTR位
0	0	1	0	1	IDE位
0	0	1	1	1	ID.17-ID.13
0	1	1	1	1	ID.12-ID.5
0	1	1	1	0	ID.4-ID.0
0	1	1	0	0	RTR位
0	1	1	0	1	保留位1
0	1	0	0	1	保留位0
0	1	0	1	1	数据长度代码
0	1	0	1	0	数据区
0	1	0	0	0	CRC序列
1	1	0	0	0	CRC定义符
1	1	0	0	1	应答通道
1	1	0	1	1	应答定义符
1	1	0	1	0	帧结束
1	0	0	1	0	中止
1	0	0	0	1	活动错误标志
1	0	1	1	0	消极错误标志
1	0	0	1	1	支配(控制)位误差
1	0	1	1	1	错误定义符
1	1	1	0	0	溢出标志

注:

1. 位的设置反映了当前结构段的不同错误事件。

总线发生错误时被迫产生相应的错误中断(中断允许时)。同时,位流处理器的当前位置被捕捉送入错误代码捕捉寄存器。其内容直到用户通过软件读出时都是不变的。读出后,捕捉机制又被激活了。访问中断寄存器期间,中断寄存器中相应的中断标志位被清除。新的总线中断直到捕捉寄存器被读出一才可能有效。

6.4.10 错误报警限制寄存器(EMLR)

错误报警限制在这个寄存器中被定义。默认值(硬件复位时)是 96。复位模式中,此寄存器对 CPU 来说是可读/写的。工作模式中是只读的。

注意,只有之前进入复位模式,EWLR 才有可能被改变。直到复位模式被再次取消后,才有可能发生

错误状态的改变（见状态寄存器；表 14）和由新的寄存器内容引起的错误报警中断。

表 22 错误报警寄存器（EWLR）的位说明；CAN 地址 13

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
EWL.7	EWL.6	EWL.5	EWL.4	EWL.3	EWL.2	EWL.1	EWL.0

6.4.11 RX 错误计数寄存器（RXERR）

RX 错误计数寄存器反应了接收错误计数器的当前值。硬件复位后寄存器被初始化为 0。工作模式中，对 CPU 来说是只读的。只有在复位模式中才可以写访问此寄存器。

如果发生总线关闭，RX 错误计数器就被初始化为 0。总线关闭期间，写这个寄存器是无效的。

注意，只有之前进入复位模式，才有可能由 CPU 迫使 RX 错误计数器发生改变。直到复位模式被取消后，错误状态的改变（见状态寄存器；表 14）、错误报警和由新的寄存器内容引起的错误中断才可能有效。

表 23 RX 错误计数寄存器（RXERR）各位的功能说明；CAN 地址 14

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
RXERR.7	RXERR.6	RXERR.5	RXERR.4	RXERR.3	RXERR.2	RXERR.1	RXERR.0

6.4.12 TX 错误计数寄存器（TXERR）

TX 错误计数寄存器反映了发送错误计数器的当前值。

工作模式中，这个寄存器对 CPU 是只读内存。复位模式中才可以写访问这个寄存器。硬件复位后，寄存器被初始化为 0。如果总线关闭，TX 错误计数器被初始化为 127 来计算总线定义的最小时间（128 个总线空闲信号）。这段时间里读 TX 错误计数器将反映出总线关闭恢复的状态信息。

如果总线关闭是激活的，写访问 TXERR 的 0-254 单元会清除总线关闭标志，复位模式被清除后控制器会等待一个 11 位的连续隐藏（弱势）位（总线空闲）。

表 24 TX 错误计数寄存器（TXERR）的各位功能说明；CAN 地址 15

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TXERR.7	TXERR.6	TXERR.5	TXERR.4	TXERR.3	TXERR.2	TXERR.1	TXERR.0

向 TXERR 写入 255 会初始化 CPU 驱动的总线关闭事件。只有之前进入复位模式，才有可能发生 CPU 引起的 TX 错误计数器内容的改变。直到复位模式被再次取消，错误或总线状态的改变（见状态寄存器；表 14）、错误报警和由新的寄存器内容引起的错误中断才有可能有效。离开复位模式后，就象总线错误引起的一样，给出新的 TX 计数器内容且总线关闭被同样的执行。这意味着重新进入复位模式，TX 错误计数器被初始化到 127，RX 计数器被清 0，所有的相关状态和中断寄存器位被置位。

复位模式的清除将会执行协议规定的总线关闭恢复序列（等待 128 个总线空闲信号）。

如果在总线关闭恢复（TXERR>0）之前又进入复位模式，总线关闭保持有效且 TXERR 被锁定。

6.4.13 发送缓冲器

发送缓冲器的全部列表见图 7。请务必分清标准帧格式（SFF）和扩展帧格式（EFF）配置。发送缓冲器允许定义长达 8 个数据字节发送信息。

6.4.13.1 发送缓冲器列表

发送缓冲器被分为描述符区和数据区，描述符区的第一个字节是帧信息字节（帧信息）。它说明了帧格式（SFF 或 EFF）、远程或数据帧和数据长度。SFF 有两个字节的识别码，EFF 有四个字节的识别码。数据区最长 8 个数据字节。发送缓冲器长 13 个字节，在 CAN 地址的 16-28。

注意，使用 CAN 地址的 96-108 可以直接访问发送缓冲器的 RAM。这个 RAM 区是为发送缓冲器保留的。下面三个字节是通用的：CAN 地址 109、110 和 111。

CAN地址	16	TX 帧信息	CAN 地址	16	TX 帧信息
	17	TX 识别码 1		17	TX 识别码 1
	18	TX 识别码 2		18	TX 识别码 2
	19	TX 数据字节 1		19	TX 识别码 3
	20	TX 数据字节 2		20	TX 识别码 4
	21	TX 数据字节 3		21	TX 数据字节 1
	22	TX 数据字节 4		22	TX 数据字节 2
	23	TX 数据字节 5		23	TX 数据字节 3
	24	TX 数据字节 6		24	TX 数据字节 4
	25	TX 数据字节 7		25	TX 数据字节 5
	26	TX 数据字节 8		26	TX 数据字节 6
	27	未使用		27	TX 数据字节 7
	28	未使用		28	TX 数据字节 8

a. 标准帧格式

b. 扩展帧格式

图 7 标准帧和扩展帧格式配置在发送缓冲器里的列表

6.4.13.2 发送缓冲器的描述符区

发送缓冲器位的列表见表 25-27 (SFF)，表 28-32 (EFF)。给出的配置是和接收缓冲器列表相匹配的。

表 25 TX 帧信息 (SFF): CAN 地址 16

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
FF ⁽¹⁾	RTR ⁽²⁾	X ⁽³⁾	X ⁽³⁾	DLC.3 ⁽⁴⁾	DLC.2 ⁽⁴⁾	DLC.1 ⁽⁴⁾	DLC.0 ⁽⁴⁾

注:

1. 帧格式。
2. 远程发送请求。
3. 不影响；推荐在使用自接收设备（自测）时和接收缓冲器（0）兼容。
4. 数据长度代码位。

表 26 TX 识别码 1 (SFF); CAN 地址 17; 注 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID.28	ID.27	ID.26	ID.25	ID.24	ID.23	ID.22	ID.21

注:

1. ID.×表示识别码的×位。

表 27 TX 识别码 2 (SFF); CAN 地址 18; 注 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID.20	ID.19	ID.18	$X^{(2)}$	$X^{(3)}$	$X^{(3)}$	$X^{(3)}$	$X^{(3)}$

注:

1. ID.X表示识别码的X位。
2. 影响；推荐在使用自接收设备（自测）时和接收缓冲器（RTR）兼容。
3. 不影响；推荐在使用自接收设备（自测）时和接收缓冲器（0）兼容。

表 28 TX 帧信息 (EFF); CAN 地址 16

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
FF ⁽¹⁾	RTR ⁽²⁾	X ⁽³⁾	X ⁽³⁾	DLC.3 ⁽⁴⁾	DLC.2 ⁽⁴⁾	DLC.1 ⁽⁴⁾	DLC.0 ⁽⁴⁾

注:

1. 帧格式。
2. 远程发送请求。
3. 不影响; 推荐在使用自接收设备 (自测) 时和接收缓冲器 (0) 兼容。
4. 数据长度代码位。

表 29 TX 识别码 1 (EFF); CAN 地址 17; 注 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID.28	ID.27	ID.26	ID.25	ID.24	ID.23	ID.22	ID.21

注:

1. ID.×表示识别码的×位。

表 30 TX 识别码 2 (EFF); CAN 地址 18; 注 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID.20	ID.19	ID.18	ID.17	ID.16	ID.15	ID.14	ID.13

注:

1. ID.×表示识别码的×位。

表 31 TX 识别码 3 (EFF); CAN 地址 19; 注 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID.12	ID.11	ID.10	ID.9	ID.8	ID.7	ID.6	ID.5

注:

1. ID.×表示识别码的×位。

表 32 TX 识别码 4 (EFF); CAN 地址 20; 注 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID.4	ID.3	ID.2	ID.1	ID.0	X ⁽²⁾	X ⁽³⁾	X ⁽³⁾

注:

1. ID.×表示识别码的×位。
2. 不影响; 推荐在使用自接收设备 (自测) 时和接收缓冲器 (RTR) 兼容。
3. 不影响; 推荐在使用自接收设备 (自测) 时和接收缓冲器 (0) 兼容。

表 33 帧格式 (FF) 和远程发送请求 (RTR) 位

位	值	功能
FF	1	EFF;CAN控制器将发送扩展帧格式
	0	SFF;CAN控制器将发送标准帧格式
RTR	1	远程;CAN控制器将发送远程帧
	0	数据;CAN控制器将发送数据帧

6.4.13.3 数据长度代码 (DLC)

数据区的信息字节长度由数据长度代码编制。在远程帧发送开始时由于 RTR 位被置位 (远程), 数据长度代码是不被考虑的。这使接收/发送的数据字节数目为 0。如果有两个 CAN 控制器使用同一个识别码同时启动远程帧传送, 数据长度代码必须正确说明以避免总线错误。

数据字节长度范围是 0-8，编码形式如下：

数据字节数= $8 \times \text{DLC.3} + 4 \times \text{DLC.2} + 2 \times \text{DLC.1} + \text{DLC.0}$

为了兼容，大于 8 的数据长度代码是不可用的。如果大于 8，将以 8 个字节计。

6.4.13.4 识别码（ID）

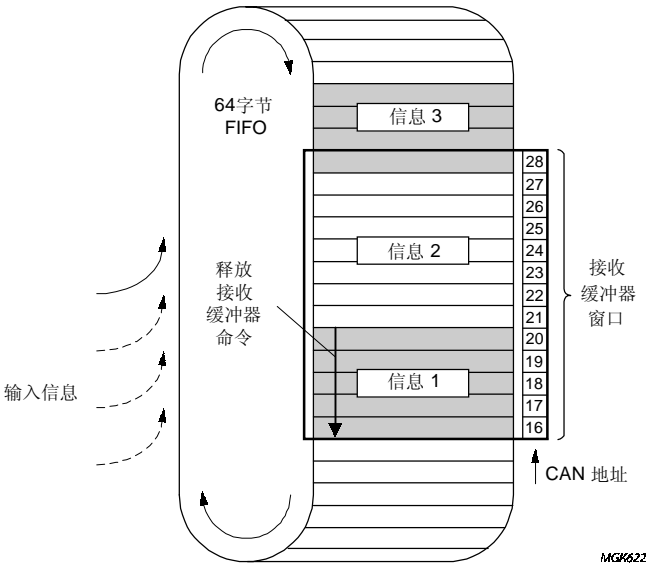
标准帧格式（SFF）的识别码有 11 位（ID.28-ID.18），扩展帧格式的识别码有 29 位（ID.28-ID.0）。ID.28 是最高位，在总线仲裁过程中最先发送到总线上。识别码就象信息的名字一样，使用在验收滤波器中，而且在仲裁过程中决定了总线访问的优先权。识别码的二进制值越低优先权越高。这是由于仲裁时有大量的前导支配位。

6.4.13.5 数据区

发送的字节数取决于数据长度代码。最先发送的是在 CAN 地址 19（SFF）或 21（EFF）的数据字节 1 的最高位。

6.4.14 接收缓冲器

接收缓冲器的列表与前面一节讲述的发送缓冲器很相似。接收缓冲器是 RXFIFO 的可访问部分，位于 CAN 地址的 16 和 28。每条信息都分为描述符和数据区。



接收缓冲器中当前的可用信息是信息 1

图 8 RXFIFO 中的信息存储举例

6.4.14.1 描述符区

接收缓冲器的位列表见表 34-36（SFF）和表 37-41（EFF）。所选配置是与接收缓冲器列表（见 6.4.13.2 节）相匹配的。

表 34 RX 帧信息（SFF）；CAN 地址 16

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
FF ⁽¹⁾	RTR ⁽²⁾	0	0	DLC.3 ⁽³⁾	DLC.2 ⁽³⁾	DLC.1 ⁽³⁾	DLC.0 ⁽³⁾

- 注：
- 1. 帧格式。
 - 2. 远程发送请求。
 - 3. 数据长度代码位。

表 35 RX 识别码 1 (SFF); CAN 地址 17; 注 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID.28	ID.27	ID.26	ID.25	ID.24	ID.23	ID.22	ID.21

注:

1. ID.×表示识别码的×位。

表 36 RX 识别码 2 (SFF); CAN 地址 18; 注 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID.20	ID.19	ID.18	RTR ⁽²⁾	0	0	0	0

注:

1. ID.×表示识别码的×位。
2. 远程发送请求。

表 37 RX 帧信息 (EFF); CAN 地址 16

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
FF ⁽¹⁾	RTR ⁽²⁾	0	0	DLC.3 ⁽³⁾	DLC.2 ⁽³⁾	DLC.1 ⁽³⁾	DLC.0 ⁽³⁾

注:

1. 帧格式。
2. 远程发送请求。
3. 数据长度代码位。

表 38 RX 识别码 1 (EFF); CAN 地址 17; 注 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID.28	ID.27	ID.26	ID.25	ID.24	ID.23	ID.22	ID.21

注:

1. ID.×表示识别码的×位。

表 39 RX 识别码 2 (EFF); CAN 地址 18; 注 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID.20	ID.19	ID.18	ID.17	ID.16	ID.15	ID.14	ID.13

注:

1. ID.×表示识别码的×位。

表 40 RX 识别码 3 (EFF); CAN 地址 19; 注 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID.12	ID.11	ID.10	ID.9	ID.8	ID.7	ID.6	ID.5

注:

1. ID.×表示识别码的×位。

表 41 RX 识别码 4 (EFF); CAN 地址 20; 注 1

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
ID.4	ID.3	ID.2	ID.1	ID.0	RTR ⁽²⁾	0	0

注:

1. ID.×表示识别码的×位。
2. 远程发送请求。

注意：在帧信息字节中的接收字节长度代码代表实际发送的数据长度代码，它有可能大于 8（取决于发送器）。无论如何，最大接收数据字节数是 8。这一点在阅读接收缓冲器中的信息时应当考虑。

见图 8, RXFIFO 共有 64 个信息字节的空間。一次可以存储多少条信息取决于数据的长度。如果 RXFIFO 中没有足够的空間来存储新的信息，CAN 控制器会产生数据溢出条件，此时信息有效且接受检测为肯定。发生数据溢出情况时，已部分写入 RXFIFO 的信息将被删除。这种情况可以通过状态寄存器和数据超限中断（中断允许）反应到 CPU。

6.4.15 验收滤波器

在验收滤波器的帮助下，只有当接收信息中的识别位和验收滤波器预定义的值相等时，CAN 控制器才允许将已接收信息存入 RXFIFO。

验收滤波器由验收代码寄存器(ACRn)和验收屏蔽寄存器（AMRn）定义。要接收的信息的位模式在验收代码寄存器中定义。相应的验收屏蔽寄存器允许定义某些位为“不影响”（即可为任意值）。

有两种不同的过滤模式可在模式寄存器中选择（MOD.3, AFM；见 6.4.3 节）：

- 单滤波器模式（AFM 位是 1）
- 双滤波器模式（AFM 位是 0）

6.4.15.1 单滤波器配置

这种滤波器配置可以定义一个长滤波器（4 字节）。滤波器字节和信息字节之间位的对应关系取决于当前接收帧格式。

标准帧：如果接收的是标准帧格式的信息，在验收滤波中只使用前两个数据字节来存放包括 RTR 位的完整的识别码。如果由于置位 RTR 位而导致没有数据字节，或因为设置相应的数据长度代码而没有或只有一个数据字节，信息也会被接收的。对于一个成功接收的信息，所有单个位的比较后都必须发出接受信号。

注意，AMR1 和 ACR1 的低四位是不用的。为了和将来的产品兼容，这些位可通过设置 AMR1.3、AMR1.2、AMR1.4 和 AMR1.0 为 1 而定为“不影响”。

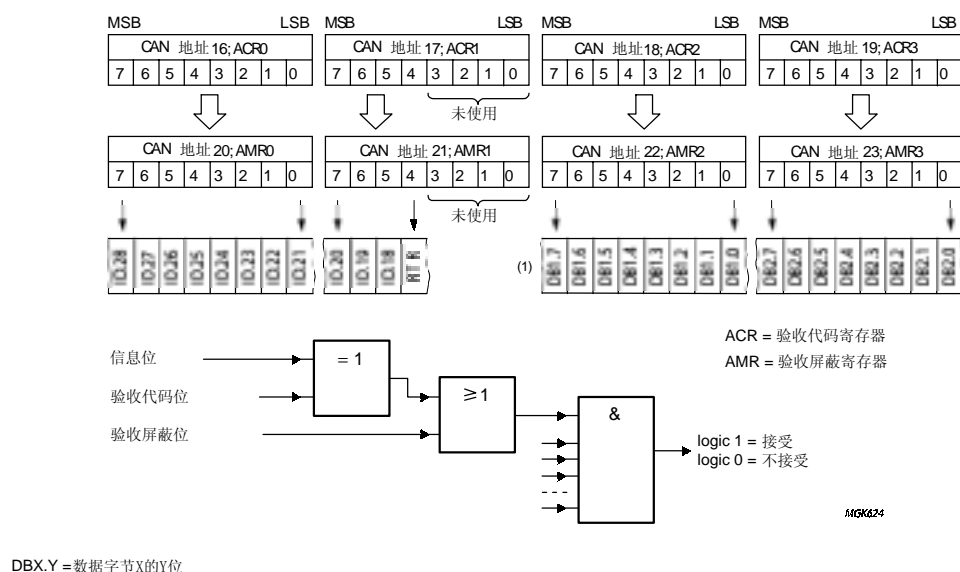


图 9 接收标准结构信息时的单个滤波器配置

扩展帧：如果接收的信息是扩展帧格式的，包括 RTR 位的全部识别码将被接受过滤使用。

为了成功接收信息，每个位的比较后都必须发出接受信号。

必须注意的是，AMR3 的最低两位和 ACR3 是不用的。为了和将来的产品兼容，这些位应该通过置位 AMR3.1 和 AMR3.0 来定为“不影响”。

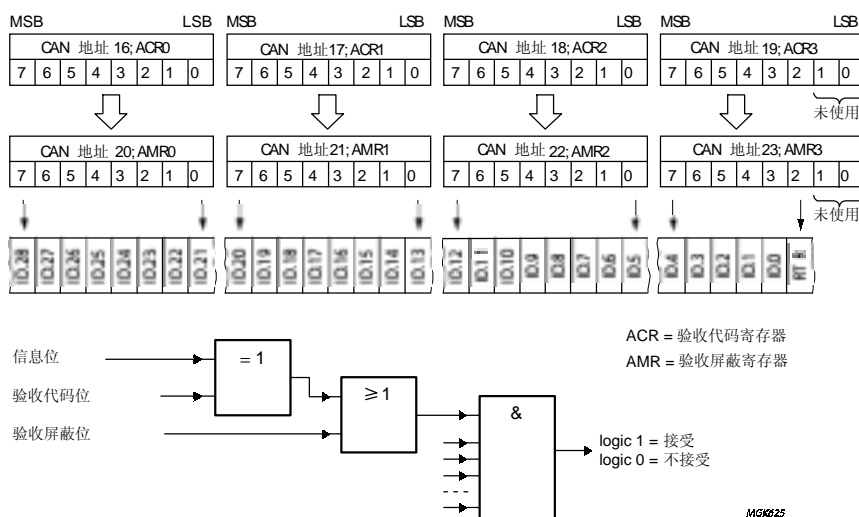


图 10 单滤波器配置，接收扩展帧信息

6.4.15.2 双滤波器的配置

这种配置可以定义两个短滤波器。一条接收的信息要和两个滤波器比较来决定是否放入接收缓冲器中。至少有一个滤波器发出接受信号，接收的信息才有效。滤波器字节和信息字节之间位的对应关系取决于当前接收的帧格式。

标准帧：如果接收的是标准帧信息，被定义的两个滤波器是不一样的。第一个滤波器比较包括 RTR 位的整个标准识别码和信息的第一数据字节。第二个滤波器只比较包括 RTR 位的整个标准识别码。

为了成功接收信息，所有单个位的比较时应至少有一个滤波器表示接受。RTR 位置位或数据长度代码是 0 时表示没有数据字节存在。无论怎样，只要从开始到 RTR 位的部分都被表示接收，信息就可以通过滤波器 1。

如果没有向滤波器请求数据字节过滤，AMR1 和 AMR3 的低四位必须被置为 1（不影响）。当使用包括 RTR 位的整个标准识别码时，两个滤波器都同样工作。

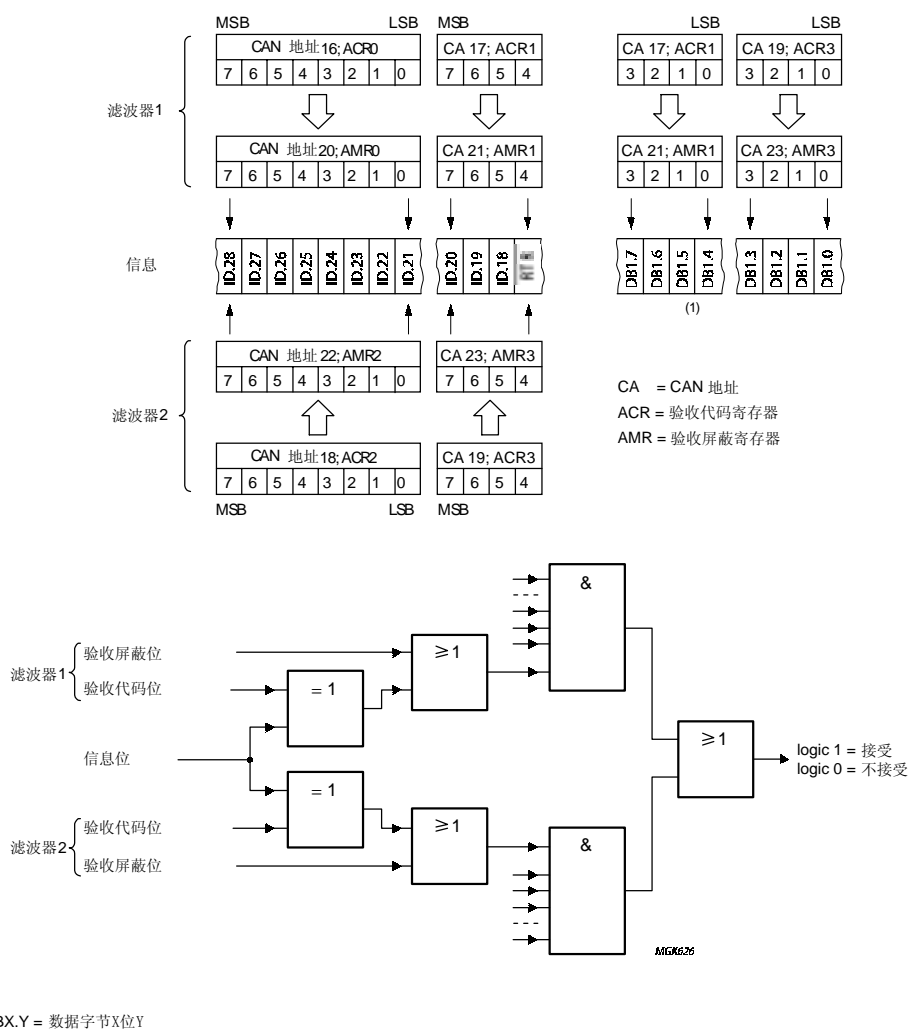


图 11 接收标准帧信息的双滤波器配置

扩展帧：如果接收到扩展帧信息，定义的两个滤波器是相同的。两个滤波器都只比较扩展识别码的前两个字节。

为了能成功接收信息，所有单个位的比较时至少有一个滤波器表示接收。

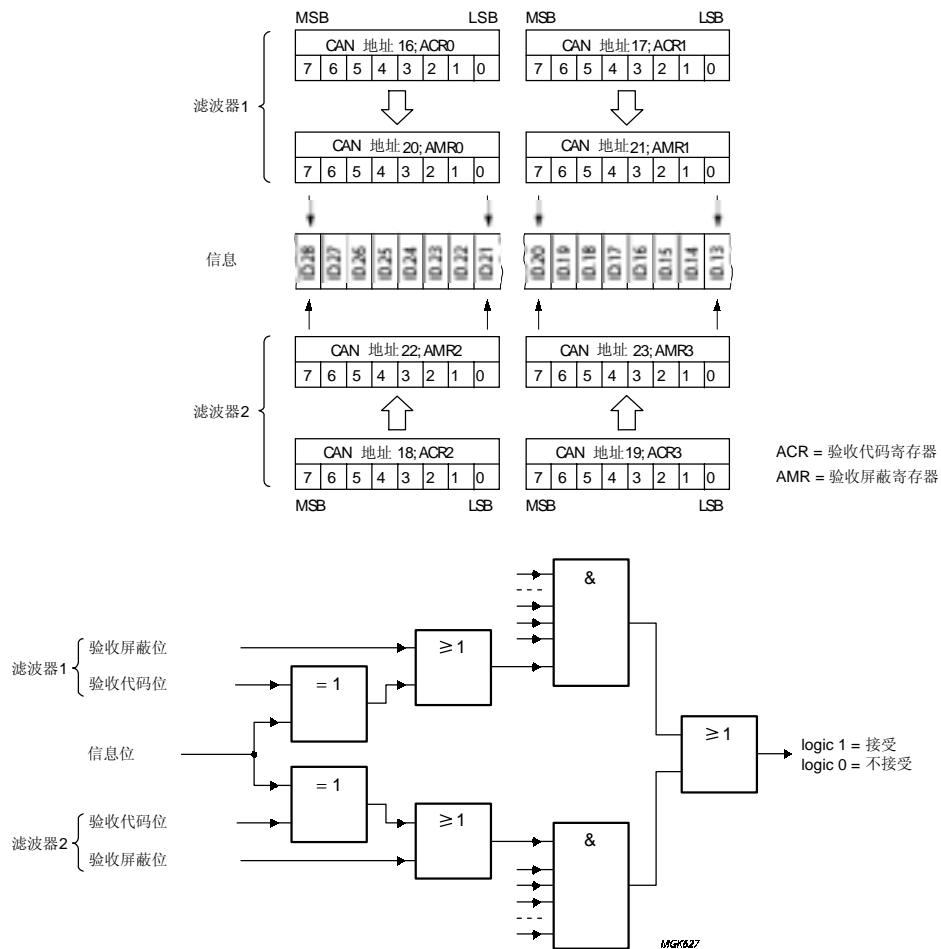


图 12 双滤波器配置，接收扩展帧信息

6.4.16 RX 信息计数器（RMC）

RMC 寄存器（CAN 地址 29）反映了 RXFIFO 中可用的信息数目。其值每次接收时加 1，每次释放接收缓冲器减 1。每次复位后，该寄存器清 0。

表 42 RX 信息计数器（RMC）各位的功能说明；CAN 地址 29

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
(0) ⁽¹⁾	(0) ⁽¹⁾	(0) ⁽¹⁾	RMC.4	RMC.3	RMC.2	RMC.1	RMC.0

注：

1. 此位不能被写。读这个寄存器时结果总是 0。

6.4.17 RX 缓冲器起始地址寄存器（RBSA）

RBSA 寄存器（CAN 地址 30）反映了当前可用来存储位于接收缓冲器窗口中的信息的内部 RAM 地址。这条信息可以帮助说明内部 RAM 的内容。起始于 CAN 地址 32 的内部 RAM 地址区可以被 CPU 读/写访问（复位模式只能写）。

例子：如果 RBSA 被设置为 24（十进制），当前在接收缓冲器窗口中的可视信息被存储在内部 RAM，起始地址 24。因为 RAM 也被直接列入 CAN 地址空间（起始地址 32，等于 RAM 地址 0），所以这条信息也可以用 CAN 地址 56 及随后字节地址访问。

CAN 地址=RBSA+32>24+32=56）。

如果信息超过 RAM 地址 63，会从地址 0 继续。

当 FIFO 中至少有一条可用信息时就会执行释放接收缓冲器命令。RBSA 在下一条信息开始的时候更

新。

硬件复位时，指针初始化为‘00H’。软件复位（设置为复位模式）时，指针保持原值，但 FIFO 被清空；这就意味着 RAM 的内容是不变的，但下一条接收的（或传送的）信息将会覆盖当前在接收缓冲器窗口的可视信息。

RX 缓冲器起始地址寄存器在工作模式中是只读的，在复位模式中是可读/写的。必须注意写访问 RBSA 首次有效是在下一个内部时钟频率的上升沿，内部时钟频率是外部振荡器的 1/2。

表 43 RX 缓冲器起始地址寄存器（RBSA）各位的功能说明；CAN 地址 30

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
(0) ⁽¹⁾	(0) ⁽¹⁾	RBSA.5	RBSA.4	RBSA.3	RBSA.2	RBSA.1	RBSA.0

注：

1. 此位不能写。此寄存器的读出值总是 0。

6.5 命令寄存器

6.5.1 总线定时寄存器 0（BTR0）

总线定时寄存器 0 定义了波特率预设值（BRP）和同步跳转宽度（SJW）的值。复位模式有效时这个寄存器是可以被访问（读/写）的。

如果选择的是 PeliCAN 模式，此寄存器在工作模式中是只读的。在 BasicCAN 模式中总是‘FFH’。

表 44 总线定时寄存器 0（BTR0）的位功能说明；CAN 地址 6

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SJW.1	SJW.0	BRP.5	BRP.4	BRP.3	BRP.2	BRP.1	BRP.0

6.5.1.1 波特率预设值（BRP）

CAN 系统时钟 t_{SCL} 的周期是可编程的，而且决定了相应的位时序。CAN 系统时钟由如下公式计算：

$$t_{SCL} = 2 \times t_{CLK} \times (32 \times BRP.5 + 16 \times BRP.4 + 8 \times BRP.3 + 4 \times BRP.2 + 2 \times BRP.1 + BRP.0 + 1)$$

这里 $t_{CLK} = XTAL$ 的频率周期 $= 1/f_{XTAL}$

6.5.1.2 同步跳转宽度（SJW）

为了补偿在不同总线控制器的时钟振荡器之间的相位偏移，任何总线控制器必须在当前传送的相关信号边沿重新同步。同步跳转宽度定义了每一位周期可以被重新同步缩短或延长的时钟周期的最大数目，

$$t_{SJW} = t_{SCL} \times (2 \times SJW.1 + SJW.0 + 1)$$

6.5.2 总线定时寄存器（BTR1）

总线定时寄存器 1 定义了每个位周期的长度、采样点的位置和在每个采样点的采样数目。在复位模式中，这个寄存器可以被读/写访问。在 PeliCAN 模式的工作模式中，这个寄存器是只读的。在 BasicCAN 模式中总是‘FFH’。

表 45 总线定时寄存器 1（BTR1）的各位功能说明；CAN 地址 7

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
SAM	TSEG2.2	TSEG2.1	TSEG2.0	TSEG1.3	TSEG1.2	TSEG1.1	TSEG1.0

6.5.2.1 采样（SAM）

位	值	功能
SAM	1	三倍;总线采样三次;建议在低/中速总线(A和B级)上使用,这对过滤总线上的毛刺波是有益的
	0	单倍;总线采样一次;建议使用在高速总线上(SAE C级)

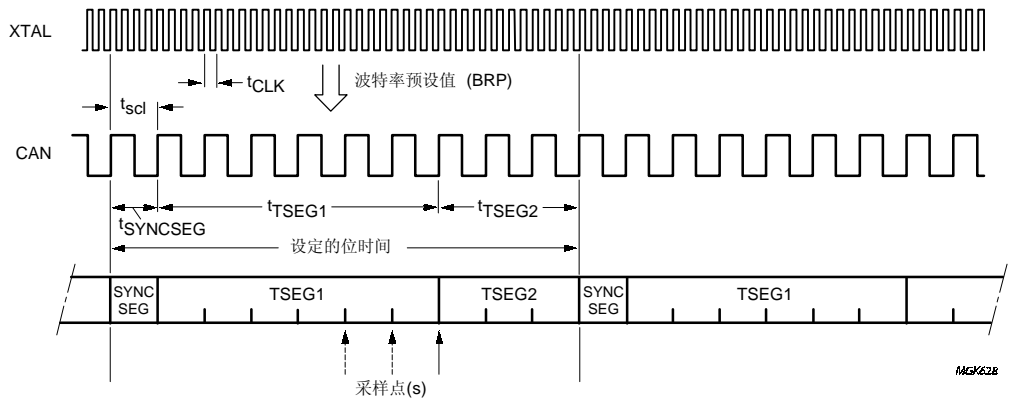
6.5.2.2 时间段 1 (TSEG1) 和时间段 (TSEG2)

(TSEG1) 和 (TSEG2) 决定了每一位的时钟数目和采样点的位置，这里：

$$t_{\text{SYNCSEG}} = 1 \times t_{\text{SCL}}$$

$$t_{\text{TSEG1}} = t_{\text{SCL}} \times (8 \times \text{TSEG1.3} + 4 \times \text{TSEG1.2} + 2 \times \text{TSEG1.1} + \text{TSEG1.0} + 1)$$

$$t_{\text{TSEG2}} = t_{\text{SCL}} \times (4 \times \text{TSEG2.2} + 2 \times \text{TSEG2.1} + \text{TSEG2.0} + 1)$$



可能值是: BRP = 000001, TSEG1 = 0101, TSEG2 = 010.

图 13 一个位周期的总体结构

6.5.3 输出控制寄存器 (OCR)

输出控制寄存器实现了由软件控制不同输出驱动配置的建立。在复位模式中此寄存器可被读/写访问。在 PeliCAN 模式的工作模式中，这个寄存器是只读的。在 BasicCAN 模式中总是 'FFH'。

表 46 输出控制寄存器 (OCR) 的各位功能说明; CAN 地址 8

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OCTP1	OCTN1	OCPOL1	OCTP0	OCTN0	OCPOL0	OCMODE1	OCMODE0

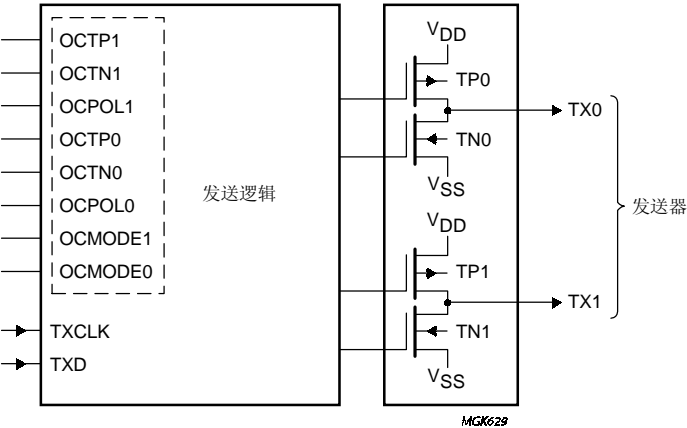


图 14 收发器的输入/输出控制逻辑

当 SJA1000 在睡眠模式中时，TX0 和 TX1 引脚根据输出控制寄存器的内容输出隐性的电平。在复位状态（复位请求=1）或外部复位引脚/RST 被拉低时，输出 TX0 和 TX1 悬空。

发送的输出阶段可以有不同的模式。表 47 列出了输出控制寄存器的设置。

表 47 OCMODE 位的说明

OCMODE1	OCMODE0	说明
0	0	双相输出模式
0	1	测试输出模式;注 1
1	0	正常输出模式
1	1	时钟输出模式

注:

1. 检测输出模式中，TXn 会在下一个系统时钟的上升沿映射在 RX 各引脚检上。TN1、TN0、 TP1 和 TP0 配置同 OCR 相对应。

6.5.3.1 正常输出模式

正常模式中位序列（TXD）通过 TX0 和 TX1 送出。输出驱动引脚 TX0 和 TX1 的电平取决于被 OCTPx,OCTNx（悬空，上拉，下拉，推挽）编程的驱动器的特性和被 OCPOLx 编程的输出端极性。

6.5.3.2 时钟输出模式

TX0 引脚在这个模式中和正常模式中是相同的。但是，TX1 上的数据流被发送时钟（TXCLK）代替了。发送时钟（不翻转）的上升沿标志着一位的开始。时钟脉冲宽度是 $1 \times tscl$ 。

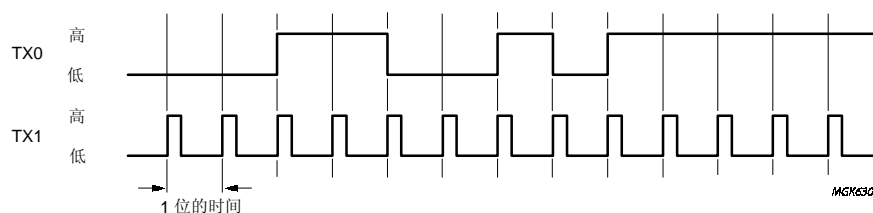


图 15 时钟输出模式举例

6.5.3.3 双相输出模式

相对于正常输出模式，这里的位代表着时间的变化和触发。如果总线控制器被发送器从总线上通电退耦，则位流不允许含有直流元件。这一点的总结见下一个表。在隐性位无效（悬空）期间，支配位轮流使用 TX0 或 TX1 电平发送，例如，第一位在 TX0 上发送，第二位在 TX1 上发送，第三位 TX0 上发送等等，依此类推。

双相输出时序配置的例子见图 16。

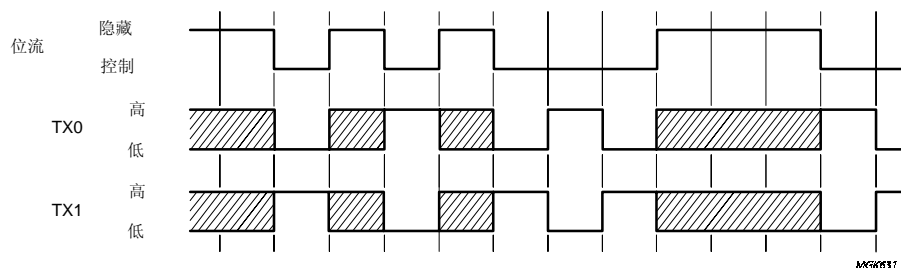


图 16 双相输出模式举例（输出控制寄存器=F8H）

6.5.3.4 测试输出模式

在测试输出模式中 RX 上的电平在下一个系统时钟的上升沿映射到 TXn 上，系统时钟 ($f_{osc}/2$) 与输出控制寄存器中定义的极性一致。

表 48 显示了输出控制寄存器的位和输出引脚 TX0 和 TX1 的关系。

表 48 输出引脚配置；注 1

驱动	TXD	OCTPX	OCTNX	OCPOLX	TPX (2)	TNX (3)	TXX (4)
悬空	×	0	0	×	关	关	悬空
上拉	0	0	1	0	关	开	低
	1	0	1	0	关	关	悬空
	0	0	1	1	关	关	悬空
	1	0	1	1	关	开	低
下拉	0	1	0	0	关	关	悬空
	1	1	0	0	开	关	高
	0	1	0	1	开	关	高
	1	1	0	1	关	关	悬空
上拉	0	1	1	0	关	开	低
	1	1	1	0	开	关	高
	0	1	1	1	开	关	高
	1	1	1	1	关	开	低

注：

1. ×=不影响。
2. TPX 是片内输出发送器 X，连接 V_{DD} 。
3. TNX 是片内输出发送器 X，连接 V_{SS} 。
4. TXX 是在引脚 TX0 或 TX1 上的串行输出电平。当 TXD=0 和 TXD 连续是 1 时，CAN 总线上的输出电平必需是本地的。

位序列 (TXD) 通过 TX0 和 TX1 发送。输出驱动引脚上的电平取决于被 OCTPx,OCTNx (悬空，上拉，下拉，推挽) 编程的驱动器的特性和被 OCPOLx 编程的输出端极性。

6.5.4 时钟分频寄存器 (CDR)

时钟分频寄存器为微控制器控制 CLKOUT 的频率以及屏蔽 CLKOUT 引脚。而且它还控制着 TX1 上的专用接收中断脉冲、接收比较通道和 BasicCAN 模式与 PeliCAN 模式的选择。硬件复位后寄存器的默认状态是 Motorola 模式 (0000 0101, 12 分频) 和 Intel 模式 (0000 0000, 2 分频)。

软件复位 (复位请求/复位模式) 时，此寄存器不受影响。

保留位 (CDR.4) 总是 0。应用软件总是向此位写 0 以与将来可能使用此位的特性兼容。

表 49 时钟分频寄存器 (CDR) 各位的功能说明；CAN 地址 31

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
CAN模式	CBP	RXINTEN	(0) ⁽¹⁾	关闭时钟	CD.2	CD.1	CD.0

注：

1. 此位不能被写。读值总为 0。

6.5.4.1 CD.2-CD.0

复位模式和工作模式中一样，CD.2-CD.0 是可以无限制访问的。这些位是用来定义外部 CLKOUT 引脚上的频率的。可选频率一览表见表 50。

表 50 CLKOUT 频率选择; 注 1

CD. 2	CD. 1	CD. 0	时钟频率
0	0	0	$f_{osc}/2$
0	0	1	$f_{osc}/4$
0	1	0	$f_{osc}/6$
0	1	1	$f_{osc}/8$
1	0	0	$f_{osc}/10$
1	0	1	$f_{osc}/12$
1	1	0	$f_{osc}/14$
1	1	1	f_{osc}

注:

1. f_{osc} 是外部振荡器 (XTAL) 频率。

6.5.4.2 时钟关闭

设置这一位可禁能 SJA1000 的外部 CLKOUT 引脚。只有在复位模式中才可以写访问。如果置位此位, CLKOUT 引脚在睡眠模式中是低而其它情况下是高。

6.5.4.3 RXINTEN

此位允许 TX1 输出用来做专用接收中断输出。当一条已接收的信息成功的通过验收滤波器, 一位时间长度的接收中断脉冲就会在 TX1 引脚输出 (帧的最后一位期间)。发送输出阶段应该工作在正常输出模式。极性和输出驱动可以通过输出控制寄存器编程 (见 6.5.3 节)。复位模式中只能写访问。

6.5.4.4 CBP

置位 CDR.6 可以中止 CAN 输入比较器, 但这只可在复位模式中。这主要用于 SJA1000 外接发送接收电路时。此时内部延时被减少, 这将会导致总线长度最大可能值的增加。如果 CBP 被置位, 只有 RX0 被激活。没有被使用的 RX1 输入应被连接到一个确定的电平 (例如, V_{SS})。

6.5.4.5 CAN 模式

CDR.7 定义了 CAN 模式。如果 CDR.7 是 0, CAN 控制器工作于 BasicCAN 模式。否则, CAN 控制器工作于 PeliCAN 模式。只有在复位模式中是可以写的。

7. 限值

符合 AMR (Absolute Maximum Rating) 系统 (IEC134) 规定; 所有电压都是以 V_{SS} 为参考的。

符号	参数	条件	最小值	最大值	单位
V_{DD}	电源		-0.5	+6.5	V
I_I, I_O	除了 TX0 和 TX1 之外所有引脚的输入/输出电流		-	± 4	mA
$I_{OT(sink)}$	TX0 和 TX1 共消耗的电流	注1	-	30	mA
$I_{OT(source)}$	TX0 和 TX1 源电流的和	注1	-	-20	mA
T_{amb}	工作环境温度		-40	+125	°C
T_{stg}	储存温度		-65	+150	°C
P_{tot}	总功耗		-	1.0	W
V_{esd}	各引脚的放电	注3	-1500	+1500	V
		注4	-200	+200	V

注:

1. I_{OT} 在总线失败情况下是允许的, 因为此时 TX 输出在很短时间后 (总线关闭状态) 被自动关闭。正常操作时 I_{OT} 是一峰值电流, 宽度 $t < 100ms$ 。每个 TX 输出平均输出电流不超过 10mA。
2. 这个值是基于可允许的最大温度和封装热阻, 而不是设备功耗。

3. 人体模型：等效于通过 $1.5k\Omega$ 的电阻给 $100pF$ 的电容放电。
4. 机器模型：等效于通过 25Ω 和正 $2.5\mu H$ 的 RL 电路给 $200pF$ 的电容放电。

8. 温度特性

符号	参数	条件	值	单位
$R_{th(j-a)}$	节点处的温度电阻	任何条件	67	K/W

9. 直流特性

$V_{DD}=5V$ ($\pm 10\%$); $V_{SS}=0V$; $T_{amb}=-40\sim+125^{\circ}C$; 所有电压均参考于 V_{SS} ; 除非有特殊说明。

符号	参数	条件	最小值	最大值	单位
电源					
V _{DD}	电压源		4.5	5.5	V
I _{DD}	工作电流	f _{osc} =24MHz;注1	-	15	mA
I _{sm}	睡眠模式电流	振荡器非激活状态;注2	-	40	μ A
输入					
V _{IL1}	在引脚ALE/AS, /CS, (/RD)/E, /WR和MODE上的低电平输入		-0.5	+0.8	V
V _{IL2}	在引脚XTAL1和/INT上的低电平输入电压		-	0.3 V _{DD}	V
V _{IL3}	在引脚/RST,AD0-AD7和RX0上的低电平输入电压		-0.5	+0.6	V
V _{IH1}	在引脚ALE/AS,/CS,(/RD)/E,/WR和MODE上的高电平输入电压		2.0	V _{DD} +0.5	V
V _{IH2}	在引脚XTAL1和/INT上的高电平输入电压		0.7V _{DD}	-	V
V _{IH3}	在引脚/RST,AD0-AD7和RX0上的高电平输入电压		2.4	V _{DD} +0.5	V
hys _{RST}	在引脚/RST,AD0-AD7和RX0 ⁽⁵⁾ 上的输入滞后		500	-	mA
I _{LI}	除 XTAL1,RX0 和 RX1 外各引脚的漏电流	0.45V<V _{I(D)} <V _{DD} ;注3	-	±2	μA
输出					
V _{OL}	在引脚AD0-AD7,CLKOUT和/INT上的低电平输出电压	I _{OL} =4Ma	-	0.4	V
V _{OH}	在引脚AD0-AD7,CLKOUT和/INT上的高电平输出电压	I _{OH} =-4mA	V _{DD} -0.4	-	V
CAN输入比较器 (见图22)					
V _{th(I)(diff)}	差动输入阈值电压	V _{DD} =5V±10%; 1.4V<V _{I(RX)} <V _{DD} -1.4V; ;注4和6	-	±32	mV
V _{hys}	迟滞电压		8	30	mV
I _I	输入电流		-	±400	nA
CAN输出驱动器					

$V_{OL(TX)}$	在引脚TX0和TX1上的低电平输入电压	$V_{DD}=5V \pm 10\%$ $I_O=1.2mA$;注6 $I_O=10mA$	- -	0.05 0.4	V V
$V_{OH(TX)}$	在引脚TX0和TX1上的高电平输入电压	$V_{DD}=5V \pm 10\%$ $I_O=1.2mA$;注6 $I_O=10mA$	$V_{DD}-0.05$ $V_{DD}-0.4$	- -	V V

注:

1. AD0-AD7=ALE=/RD=/WR=/CS= V_{DD} ;RST=MODE= V_{SS} ;RX0=2.7V;RX1=2.3V;XTAL1=0.5V 或 $V_{DD}-0.5V$;所有输出都是空载。
2. AD0-AD7=ALE=/RD=/WR=/INT=/RST=/CS=MODE=RX0= V_{DD} ;RX1=XTAL1= V_{SS} ;所有输出空载。
3. $V_{I(D)}$ =数据输入引脚的输入电压。
4. $V_{I(RX)}$ =RX0 和 RX1 引脚上的输入电压。
5. 只在比较器忽略模式时。
6. 生产时不检测。

10 . 交流特性

$V_{DD}=5V \pm 10\%$; $V_{SS}=0V$; $C_L=50pF$ (输出引脚); $T_{amb}=-40\sim+125^{\circ}C$;除非另有说明; 注 1。

符号	参数	条件	最小值	最大值	单位
f_{OSC}	振荡器频率		-	24	MHz
$t_{SU(A-AL)}$	对ALE/AS低电平的地址建立时间		8	-	ns
$t_{H(AL-A)}$	ALE为低的保持时间		2	-	ns
$t_{W(AL)}$	ALE/AS的脉冲宽度		8	-	ns
t_{RLQV}	/RD为低, 输出有效数据	Intel模式	-	50	ns
t_{EHQV}	E为高, 输出有效数据	Motorola模式	-	50	ns
t_{RHDZ}	/RD为高, 数据悬空	Intel模式	-	30	ns
t_{ELDZ}	E为低, 数据悬空	Motorola模式	-	30	ns
t_{DVWH}	/WR为高, 输入数据有效	Intel模式	8	-	ns
t_{WHDX}	/WR为高后的数据保持时间	Intel模式	8	-	ns
t_{WHLH}	/WR为高到下一个ALE高		15	-	ns
t_{ELAH}	E为低到下一个AS高	Motorola模式	15	-	ns
$t_{SU(I)(D-EL)}$	E为低的输入数据建立时间	Motorola模式	8	-	ns
$t_{H(I)(EL-D)}$	E为低的输入数据保持时间	Motorola模式	8	-	ns
t_{LLWL}	ALE低到/WR低	Intel模式	10	-	ns
t_{LLRL}	ALE低到/RD低	Intel模式	10	-	ns
t_{LLEH}	AS低到E高	Motorola模式	10	-	ns
$t_{SU(R-EH)}$	RD/ (/WR) 到E高的建立时间	Motorola模式	5	-	ns
$t_{W(W)}$	/WR脉冲宽度	Intel模式	20	-	ns
$t_{W(R)}$	/RD脉冲宽度	Intel模式	40	-	ns
$t_{W(E)}$	E脉冲宽度	Motorola模式	40	-	ns
t_{CLWL}	/CS低到/WR低	Intel模式	0	-	ns
t_{CLRL}	/CS低到/RD低	Intel模式	0	-	ns
t_{CLEH}	/CS低到E高	Motorola模式	0	-	ns
t_{WHCH}	/WR高到/CS高	Intel模式	0	-	ns
t_{RHCH}	/RD高到/CS高	Intel模式	0	-	ns

t_{ELCH}	E低到/CS高	Motorola模式	0	-	ns
$t_{W(RST)}$	/RST脉冲宽度		100	-	ns
输入比较器/输出驱动器					
t_{SD}	输入和输出延迟之和	$V_{DD}=5V \pm 10\%$ $V_{DIF}=\pm 32mV$ $1.4V < V_{I(RX)} < V_{DD}$ -1.4V; 注2	-	40	ns

注:

1. 生产过程中是不检测交流特性的。
2. 如果使用了外部发送接收电路, 模拟输入比较器可以通过时钟分频寄存器的 **CBP** 位的设置来内部忽略掉。这样会减少延时 ($<26ns$)。 $V_{I(RX)}=RX0$ 和 $RX1$ 引脚上的输入电压。

10.1 交流时序图

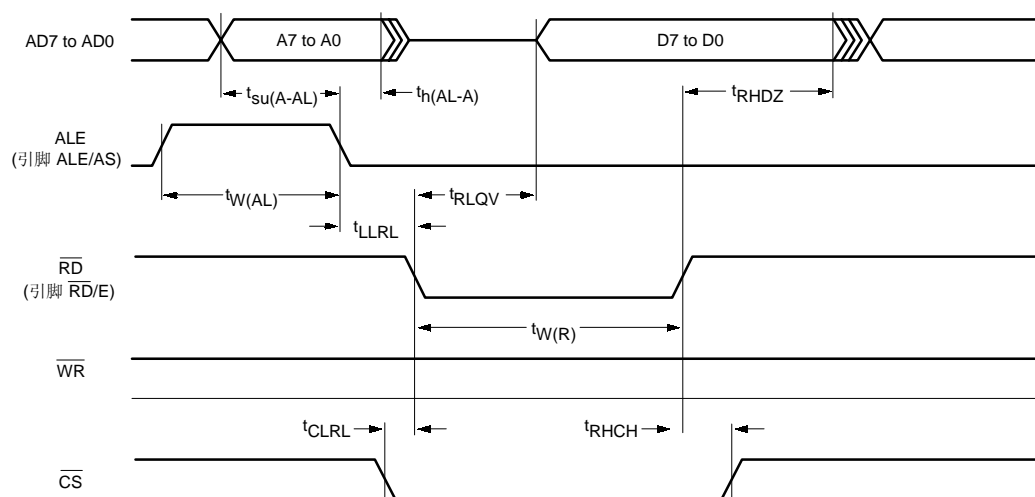


图 17 读周期时序图;Intel 模式

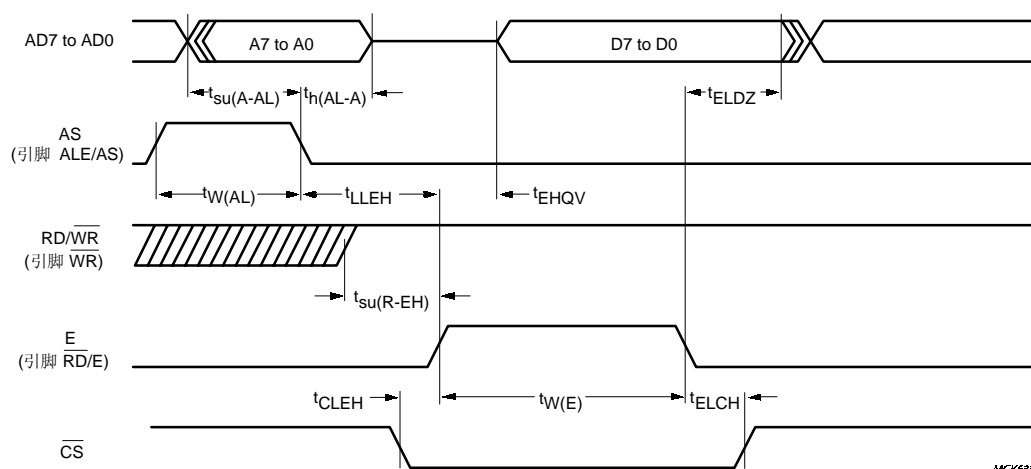


图 18 读周期时序图;Motorola 模式

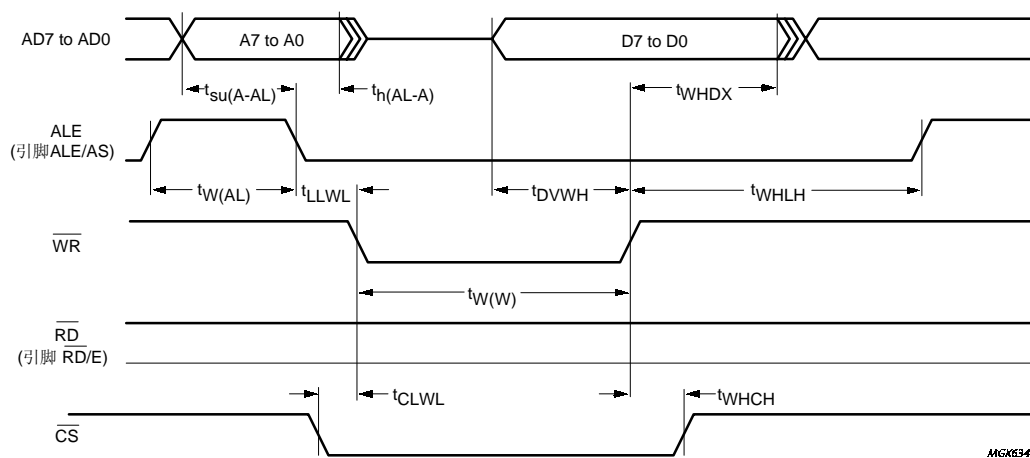


图 19 写周期时序图;Intel 模式

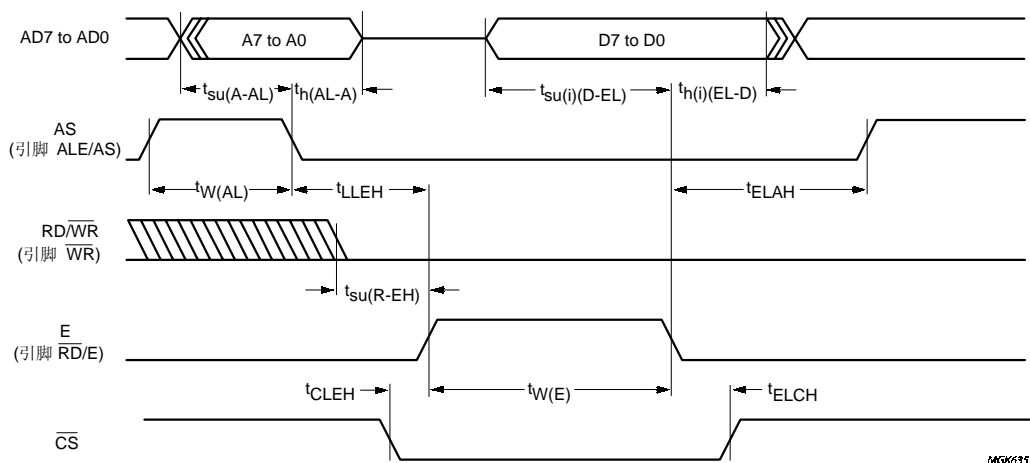


图 20 读周期时序图;Motorola 模式

10.2 附录交流信息

为了在恶劣的情况下优化其噪声消除能力，芯片由三个独立的引脚供电，三个独立的引脚接地。

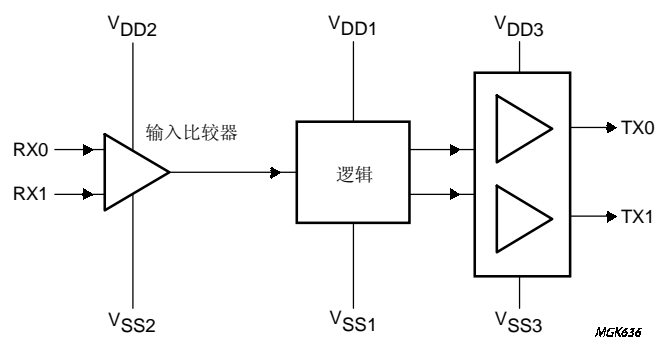
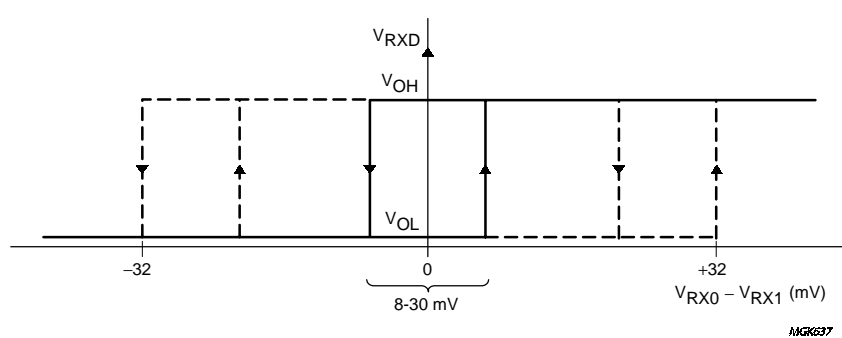


图 21 噪声消除优化图



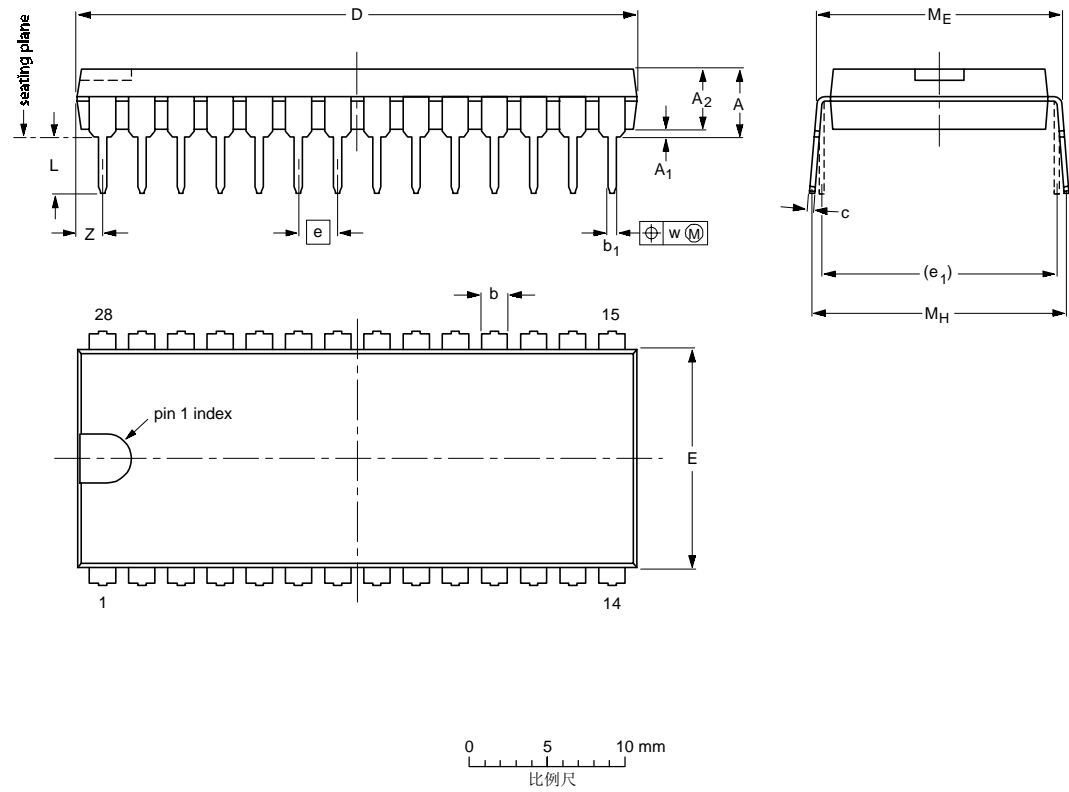
RX引脚的绝对输入电压是: $1.4\text{ V} < V_{RX} < V_{DD} - 1.4\text{ V}$

要在RXD引脚获得确定的输出电平那么由RX引脚的输入的差动电压至少要大于 $\pm 32\text{ mV}$

图 22 输入比较器的定义

11 . 封装概述

DIP28: 塑质双列直插封装; 28 引脚 (600mil)



规格尺寸 (英寸是由毫米换算得出)

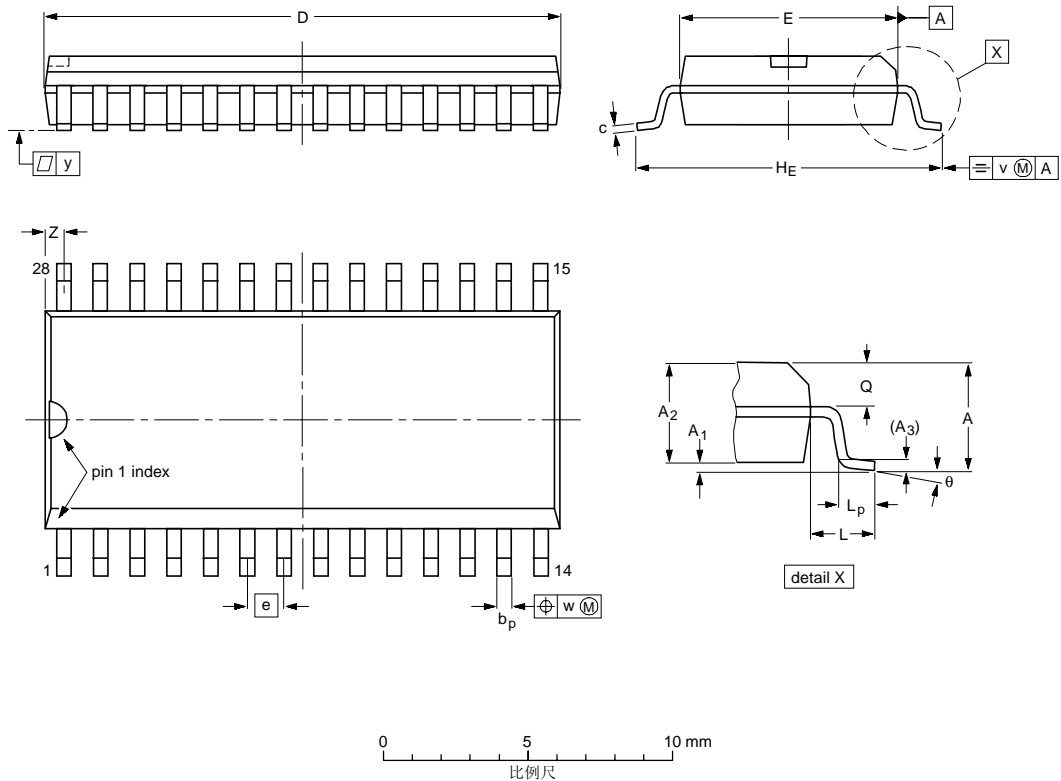
单位	A max.	A ₁ min.	A ₂ max.	b	b ₁	c	D ⁽¹⁾	E ⁽¹⁾	e	e ₁	L	M _E	H	w	Z ⁽¹⁾ max.
mm	5.1	0.51	4.0	1.7 1.3	0.53 0.38	0.32 0.23	36.0 35.0	14.1 13.7	2.54	15.24	3.9 3.4	15.80 15.24	17.15 15.90	0.25	1.7
inches	0.20	0.020	0.16	0.066 0.051	0.020 0.014	0.013 0.009	1.41 1.34	0.56 0.54	0.10	0.60	0.15 0.13	0.62 0.60	0.68 0.63	0.01	0.067

注意

1. 每边最大0.25 mm 的塑质或金属的突出部分不包含在内。

外型 版本	参考				欧洲设计的 投影图	版本日期
	IEC	JEDEC	EIAJ			
SOT117-1	051G05	MO-015	SC-510-28			95-01-14- 99-12-27

S028: 塑质小型外线封装; 28 引脚; 外宽 7.5mm



规格尺寸 (英寸是由毫米转换得到)

UNIT	A _{max.}	A ₁	A ₂	A ₃	b _p	c	D ⁽¹⁾	E ⁽¹⁾	e	H _E	L	L _p	Q	v	w	y	Z ⁽¹⁾	θ
mm	2.65	0.30 0.10	2.45 2.25	0.25	0.49 0.36	0.32 0.23	18.1 17.7	7.6 7.4	1.27	10.65 10.00	1.4	1.1 0.4	1.1 1.0	0.25	0.25	0.1	0.9 0.4	8° 0°
inches	0.10	0.012 0.004	0.096 0.089	0.01	0.019 0.014	0.013 0.009	0.71 0.69	0.30 0.29	0.050	0.419 0.394	0.055	0.043 0.016	0.043 0.039	0.01	0.01	0.004	0.035 0.016	

注意

1. 每边最大0.15 mm 的塑质或金属的突出部分不包含在内。

型号 版本	参考				欧洲设计的 映射图	版本日期
	IEC	JEDEC	EIAJ			
SOT136-1	075E06	MS-013				97-05-22 99-12-27

13. 生命支持系统中的应用

这些产品不是为生命支持工具、设备或系统而设计的，错误使用它们可能会造成人身伤害。购买和使用 PHILIPS 产品的消费者如果不顾风险在这些领域中使用，对因不恰当的使用或购买所造成的损失由用户负全部责任。