

本节将详细介绍 STC32G 系列单片机中所集成的 CAN 模块的结构和功能。STC32G 系列单片机内部集成两组独立的 CAN 总线功能单元,支持 CAN 2.0 协议。该模块内部结构框架如图 16.20 所示,该模块的主要功能如下:

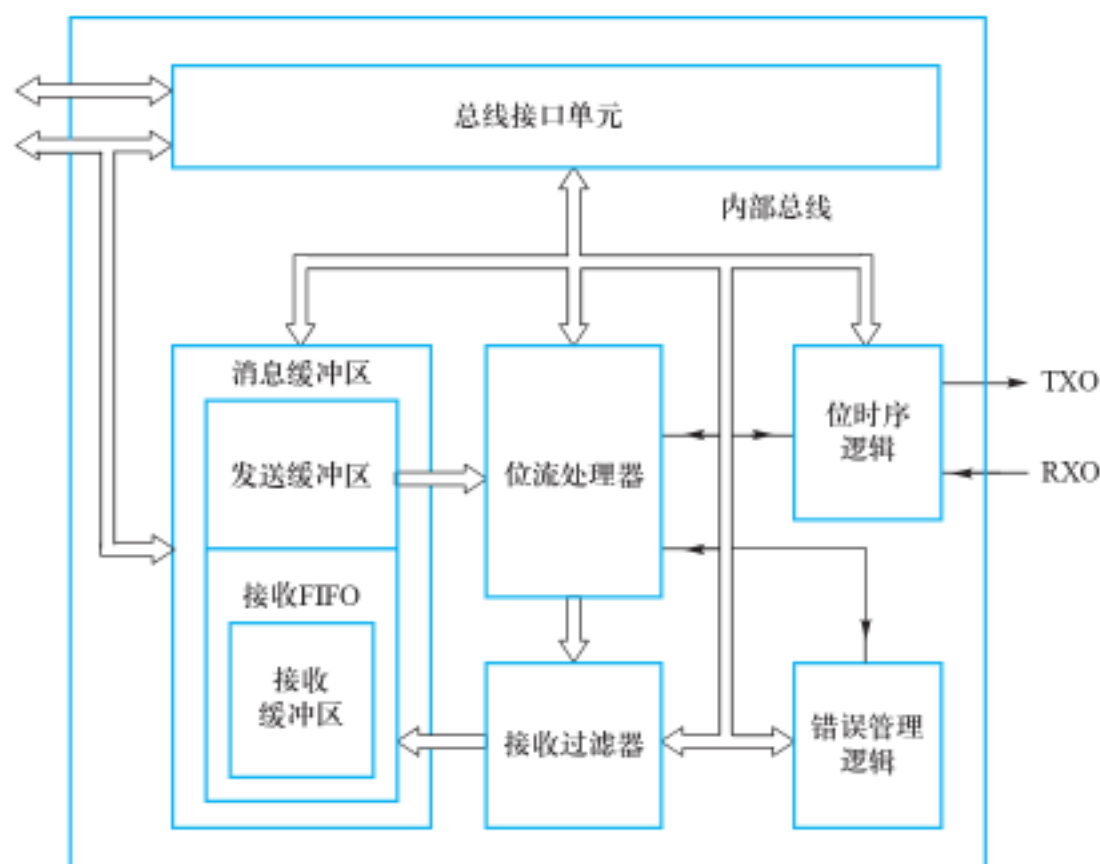


图 16.20 CAN 控制器内部结构

- (1) 标准帧和扩展帧信息的接收和传送
- (2) 64 字节的接收 FIFO
- (3) 在标准和扩展格式中都有单/双验收滤波器
- (4) 发送、接收的错误计数器
- (5) 总线错误分析

16.2.1 CAN 功能脚切换

在 STC32G 系列单片机中集成了两路 CAN 接口控制器。第一路引脚的位置通过 P_SW1 寄存器设置,其具体含义如表 16.2 所示。

表 16.2 P_SW1 寄存器的含义

符号	地址	B7	B6	B5	B4	B3	B2	B1	B0
P_SW1	A2H	S1_S[1:0]		CAN_S[1:0]		SPI_S[1:0]		LIN_S[1:0]	

从上表可知,P_SW1 寄存器第 4 和第 5 位用于选择第一路 CAN 接口所使用的 STC32G 系列单片机的引脚位置,其具体含义如表 16.3 所示。

表 16.3 第一路 CAN 接口引脚位置的设置

CAN_S[1:0]	CAN_RX	CAN_TX
00	P0.0	P0.1
01	P5.0	P5.1
10	P4.2	P4.5
11	P7.0	P7.1

第二路引脚的位置通过 P_SW3 寄存器设置,其具体含义如表 16.4 所示。

表 16.4 P_SW3 寄存器的含义

符号	地址	B7	B6	B5	B4	B3	B2	B1	B0
P_SW3	BBH	I2S_S		S2SPI_S[1:0]		S1SPI_S[1:0]		CAN2_S[1:0]	

从上表可知,P_SW3 寄存器第 0 位和第 1 位用于选择第二路 CAN 接口所使用的 STC32G 系列单片机的引脚位置,其具体含义如表 16.5 所示。

表 16.5 第二路 CAN 接口引脚位置的设置

CAN2_S[1:0]	CAN2_RX	CAN2_TX
00	P0.2	P0.3
01	P5.2	P5.3
10	P4.6	P4.7
11	P7.2	P7.3

16.2.2 CAN 模块相关的寄存器

本节介绍 CAN 模块相关的寄存器,主要包括辅助寄存器 2、CAN 总线中断控制寄存器、CAN 总线地址寄存器和 CAN 总线数据寄存器。

1. 辅助寄存器 2 (AUXR2)

该寄存器的如表 16.6 所示。该寄存器位于 SFR 区域地址为 97H 的位置。复位后,该寄存器的值为“xxxx0000”。表中:

表 16.6 辅助寄存器 2(AUXR2) 的含义

寄存器名	地址	B7	B6	B5	B4	B3	B2	B1	B0
AUXR2	97H	-	-	-	-	CANSEL	CAN2EN	CANEN	LINEN

(1) CANEN:CAN 总线使能控制位。当该位为 0 时,关闭第一路 CAN 控制器。该位为 1 时,使能第一路 CAN 控制器。

(2) CAN2EN:CAN2 总线使能控制位。当该位为 0 时,关闭第二路 CAN 控制器(CAN2)。

该位为 1 时,使能第二路 CAN 控制器(CAN2)。

(3) CANSEL:CAN 总线选择,该位为 0 时,选择第一路 CAN 控制器。该位为 1 时,选择第二路 CAN(CAN2)控制器。

2. CAN 总线中断控制寄存器 (CANICR)

该寄存器的含义如表 16.7 所示。该寄存器位于 SFR 区域地址为 F1H 的位置。复位后,该寄存器的值为“00000000”。表中:

表 16.7 CAN 总线中断控制寄存器 (CANICR) 含义

寄存器名	地址	B7	B6	B5	B4	B3	B2	B1	B0
CANICR	F1H	PCAN2H	CAN2IF	CAN2IE	PCAN2L	PCANH	CANIF	CANIE	PCANL

(1) CANIE:CAN 总线中断使能控制位,该位为 0,关闭 CAN 中断。该位为 1,使能 CAN 中断。

(2) CANIF:CAN 总线中断请求标志位,需软件清零。

(3) PCANH 和 PCANL:CAN 中断优先级控制位,其具体含义如表 16.8 所示。

表 16.8 CAN 中断优先级设置

PCANH	PCANL	优先级
0	0	0(最低)
0	1	1
1	0	2
1	1	3(最高)

(4) CAN2IE:CAN2 总线中断使能控制位,该位为 0,关闭 CAN2 中断,该位为 1,使能 CAN2 中断。

(5) CAN2IF:CAN2 总线中断请求标志位,需软件清零。

(6) PCAN2H 和 PCAN2L:CAN2 中断优先级控制位,其具体含义如表 16.9 所示。

表 16.9 优先级

PCAN2H	PCAN2L	优先级
0	0	0(最低)
0	1	1
1	0	2
1	1	3(最高)

3. CAN 总线地址寄存器 (CANAR)

该寄存器的含义如表 16.10 所示。该寄存器位于扩展 SFR 区域地址为 7EFEBBH 的位置。复位后,该寄存器的值为“00000000”。

表 16.10 CAN 总线地址寄存器 (CANAR) 含义

寄存器名	地址	B7	B6	B5	B4	B3	B2	B1	B0
CANAR	7EFEBBH								

4. CAN 总线数据寄存器 (CANDR)

如表 16.11 所示。该寄存器位于扩展 SFR 区域地址为 7EFEBCH 的位置。复位后,该寄存器的值为“00000000”。

表 16.11 CAN 总线数据寄存器 (CANDR) 含义

寄存器名	地址	B7	B6	B5	B4	B3	B2	B1	B0
CANDR	7EFEBCH								

16.2.3 CAN 模块内部功能寄存器

首先说明两路 CAN 控制器是硬件上完全各自独立的,两组 CAN 的内部寄存器也是各自独立的,只是访问地址是相同的。当需要访问不同组 CAN 的内部寄存器时,需要首先通过 AUXR2.CANSEL 进行选择然后就可以正确访问所对应的寄存器了。

这里需要特别强调,对 CAN 内部功能寄存器进行读写均需要通过 CANAR 和 CANDR 进行间接访问。

1) 读 CAN 内部功能寄存器的方法

- (1) 将 CAN 内部功能寄存器的地址写入 CAN 总线地址寄存器 CANAR。
- (2) 读取 CAN 总线数据寄存器 CANDR。

比如:需要读取 CAN 内部功能寄存器 ISR 的值

```
CANAR = 0x03;    //将 ISR 的地址写入 CANAR
dat = CANDR;     //读取 CANDR 以获得 ISR 的值
```

2) 写 CAN 内部功能寄存器的方法

- (1) 将 CAN 内部功能寄存器的地址写入 CAN 总线地址寄存器 CANAR。
- (2) 将待写入的值写入 CAN 总线数据寄存器 CANDR。

比如:需要将数据 0x5a 写入 CAN 内部功能寄存器 TXBUF0

```
CANAR = 0x08;    //将 TXBUF0 的地址写入 CANAR
CANDR = 0x5a;    //将待写入的值 0x5a 写入 CANDR
```

注:CAN 模块内部功能寄存器的地址应该理解为以 CAN 总线地址寄存器 (CANAR) 为基地址的偏移量。

1. CAN 模式寄存器 (MR)

该寄存器的含义如表 16.12 所示。表中:

表 16.12 CAN 模式寄存器(MR) 含义

寄存器名	偏移地址	B7	B6	B5	B4	B3	B2	B1	B0
MR	0x00	-	-	-	-	-	RM	LOM	AFM

(1) RM:CAN 模块的 RESET MODE。当该位为 0 时,关闭复位模式。当该位为 1 时,使能复位模式。

(2) LOM:CAN 模块的只侦听模式。当该位为 0 时,关闭只侦听模式。当该位为 1 时,使能只侦听模式。

(3) AFM:CAN 模块的接收滤波选择(参见 ACR 寄存器描述)。当该位为 0 时,接受滤波器采用双滤波设置。当该位为 1 时,表示接受滤波器采用单滤波设置。

2. CAN 命令寄存器 (CMR)

该寄存器各位含义如表 16.13 所示。表中:

表 16.13 CAN 命令寄存器(CMR) 含义

寄存器名	偏移地址	B7	B6	B5	B4	B3	B2	B1	B0
CMR	0x01	-	-	-	-	-	TR	AT	-

(1) TR:CAN 模块发送请求,该位为 1 时表示发起一次帧传输。

(2) AT:CAN 模块传输终止,该位为 1 时表示终止当前的帧传输。

3. CAN 状态寄存器 (SR)

该寄存器的含义如表 16.14 所示。表中:

表 16.14 CAN 状态寄存器(SR) 含义

寄存器名	偏移地址	B7	B6	B5	B4	B3	B2	B1	B0
SR	0x02	RBS	DSO	TBS	-	RS	TS	ES	BS

(1) RBS:接收 BUFFER 状态。该位为 0 时,接收 BUFFER 无数据帧;该位为 1 时,接收 BUFFER 有数据帧。

(2) DSO:接收 FIFO 溢出循环标志。该位为 0 时,接收 FIFO 没有溢出循环产生;该位为 1 时,接收 FIFO 有溢出循环产生。

(3) TBS:CAN 模块发送缓冲区状态。该位为 0 时,禁止发送缓冲区,CPU 不可对缓冲区进行写操作;该位为 1 时,发送缓冲区空闲,CPU 可对缓冲区进行写操作。

(4) RS:CAN 模块接收状态。该位为 0 时,CAN 模块接收空闲;该位为 1 时,CAN 模块正在接收数据帧。

(5) TS:CAN 模块发送状态。该位为 0 时,CAN 模块发送空闲;该位为 1 时,CAN 模块正在发送数据帧。

(6) ES:CAN 模块错误状态。该位为 0 时,CAN 模块错误寄存器值未达到 96;该位为 1 时,CAN 模块至少有一个错误寄存器的值达到或超过了 96。

(7) BS:CAN 模块 BUS-OFF 状态。该位为 0 时,CAN 模块不在总线关闭状态。该位为 1

时,CAN 模块在总线关闭状态。

当 CAN 控制器发生错误的次数超过 255 次,就会触发总线关闭错误。一般发生总线关闭的条件是 CAN 总线受周围环境干扰,导致 CAN 发送端发送到总线的数据被总线判断为异常,但异常的次数超过 255 次,总线自动设置为总线关闭状态,此时总线处于忙的状态,无法发送数据和无法接收数据。

4. CAN 中断/应答寄存器 (ISR/IACK)

该寄存器的各位含义如表 16.15 所示。表中:

表 16.15 CAN 中断/应答寄存器 (ISR/IACK) 含义

符号	地址	B7	B6	B5	B4	B3	B2	B1	B0
ISR/IACK	0x03	-	ALI	EWI	EPI	RI	TI	BEI	DOI

(1) ALI:仲裁丢失中断。该位为 1 时,仲裁丢失;写 1 时,将该位清零。

(2) EWI:错误警告中断。当 SR 寄存器中的 ES 或者 BS 值为 1 时,设置该位;写 1 时,将该位清零。

(3) EPI:CAN 模块被动错误中断。当 CAN 错误寄存器操作被动错误计数值时,设置该位;当写 1 时,该位清零。

(4) RI:该位为 1 时 CAN 模块接收中断,CAN 模块接收缓冲区中存在数据帧,用户需要对 RI 写 1,以减少接收信息计数器(RMC)值。

(5) TI:该位为 1 时 CAN 模块发送中断,CAN 模块数据帧发送完成。用户需要对 TI 写 1,复位发送缓冲区的写指针。

(6) BEI:该位为 1 时 CAN 模块总线错误中断,CAN 模块在接收或者发送过程中产生了总线错误。

(7) DOI:该位为 1 时 CAN 模块接收溢出中断,CAN 模块接收 FIFO 溢出。

5. CAN 中断寄存器 (IMR)

该寄存器各位的含义如表 16.16 所示。表中:

表 16.16 CAN 中断寄存器 (IMR) 含义

寄存器名	偏移地址	B7	B6	B5	B4	B3	B2	B1	B0
IMR	0x04	-	ALIM	EWIM	EPIM	RIM	TIM	BEIM	DOIM

(1) ALIM:仲裁丢失中断。该位为 0 时,禁止仲裁丢失中断;该位为 1 时,使能仲裁丢失中断。

(2) EWIM:错误警告中断。该位为 0 时,禁止错误警告中断;该位为 1 时,使能错误警告中断。

(3) EPIM:CAN 模块被动错误中断。该位为 0 时,禁止 CAN 模块被动错误中断;该位为 1 时,使能 CAN 模块被动错误中断。

(4) RI:CAN 模块接收中断。该位为 0 时,禁止 CAN 模块接收中断;该位为 1 时,使能 CAN 模块接收中断。

(5) TI:CAN 模块发送中断。该位为 0 时,禁止 CAN 模块发送中断;该位为 1 时,使能

CAN 模块发送中断。

(6) BEI:CAN 模块总线错误中断。该位为 0 时,禁止 CAN 模块总线错误中断;该位为 1 时,使能 CAN 模块总线错误中断。

(7) DOI:CAN 模块接收溢出中断。该位为 0 时,禁止 CAN 模块接收溢出中断;该位为 1 时,使能 CAN 模块接收溢出中断。

6. CAN 数据帧接收计数器 (RMC)

该寄存器的含义如表 16.17 所示。

表 16.17 CAN 数据帧接收计数器(RMC) 含义

寄存器名	偏移地址	B7	B6	B5	B4	B3	B2	B1	B0
RMC	0x05	-	-	-	RMC[4:0]				

7. CAN 总线时钟寄存器 0 (BTR0)

该寄存器各位的含义如表 16.18 所示。表中：

表 16.18 CAN 总线时钟寄存器 0(BTR0) 含义

寄存器名	偏移地址	B7	B6	B5	B4	B3	B2	B1	B0
BTR0	0x06	SJW[1:0]				BRP[5:0]			

(1) BRP:CAN 波特率分频系数。使用下面的公式计算：

$$BRP = BTR0[5:0] + 1$$

CAN 模块内部时钟周期 $t_q = t_{CLK} \times BRP$ 。其中, $t_{CLK} = 1/f_{XTAL}$ (主频 2 分频)

(2) SJW:重新同步跳跃宽度,使用下面的公式计算：

$$SJW = SJW[1] \times 2 + SJW[0]$$

8. CAN 总线时钟寄存器 1 (BTR1)

该寄存器各位的含义如表 16.19 所示。表中：

表 16.19 CAN 总线时钟寄存器 1(BTR1) 含义

寄存器名	偏移地址	B7	B6	B5	B4	B3	B2	B1	B0
TSCB[R1:0]	0x07	SAM				TSG2[2:0]			

(1) TSG1:同步采样段 1。

(2) TSG2:同步采样段 2。

(3) SAM:总线电平采样次数。该位为 0 时,总线电平采样 1 次;该位为 1,总线电平采样 3 次。

CAN 波特率可用下式计算：

$$CAN \text{ 波特率} = 1 / \text{正常时间位}$$

$$\text{其中,正常时间位} = (1 + (TSG1 + 1) + (TSG2 + 1)) * t_q$$

9. CAN 总线数据帧发送缓存 (TXBUF_n)

该寄存器各位的含义如表 16.20 所示。CAN 总线数据帧发送缓存寄存器分别位于偏移地址为 0x08、0x09、0x0A 和 0x0B 的位置。

表 16.20 CAN 总线数据帧发送缓存 (TXBUF_n) 含义

寄存器名	偏移地址	B7	B6	B5	B4	B3	B2	B1	B0
TXBUF0	0x08	Frame byte n							
TXBUF1	0x09	Frame byte n+1							
TXBUF2	0x0A	Frame byte n+2							
TXBUF3	0x0B	Frame byte n+3							

发送 BUFFER 包含 4 个寄存器:TXBUF0,TXBUF1,TXBUF2,TXBUF3。

每当写 TXBUF3 寄存器时,缓冲区指针自动加 1,将 TXBUF0,TXBUF1,TXBUF2,TXBUF3,写入缓冲区。

10. CAN 总线数据帧接收缓存 (RXBUF_n)

该寄存器各位的含义如表 16.21 所示。CAN 总线数据帧接收缓存寄存器分别位于偏移地址为 0x1C、0x1D、0x1E 和 0x1F 的位置。

表 16.21 CAN 总线数据帧接收缓存 (RXBUF_n) 含义

寄存器名	偏移地址	B7	B6	B5	B4	B3	B2	B1	B0
RXBUF0	0x0C	Frame byte n							
RXBUF1	0x0D	Frame byte n+1							
RXBUF2	0x0E	Frame byte n+2							
RXBUF3	0x0F	Frame byte n+3							

(1) 接收缓冲区包含 4 个寄存器:RXBUF0,RXBUF1,RXBUF2,RXBUF3

每当写 RXBUF3 寄存器时,缓冲区指针自动加 1,将 RXBUF0,RXBUF1,RXBUF2,RXBUF3,写入缓冲区。

一帧 CAN 的数据最长是 16 个字节,因此接收一帧数据需要循环读取接收缓冲区四次。CAN 模块的 RXFIFO 是一个 64 个字节 FIFO。当数据为 1 个字节时,最多能保存 21 帧数据;当数据为 8 个字节时,最多能保存 5 帧数据。通过读取 RMC 寄存器,可得到接收帧的个数。

(2) CAN 总线接受过滤器

在接受过滤器的帮助下 CAN 控制器能够允许 RXFIFO 只接收与识别码和接受过滤器中预设值相一致的信息,通过 ACR 寄存器和 AMR 寄存器来定义接受过滤器。

11. CAN 总线接受代码寄存器 (ACR_n)

ACR_n 寄存器各位的含义如表 16.22 所示。这些寄存器分别位于偏移地址 0x10、0x11、0x12 和 0x13 的位置。

表 16.22 CAN 总线接受代码寄存器 (ACRn) 含义

寄存器名	偏移地址	B7	B6	B5	B4	B3	B2	B1	B0
ACR0	0x10	ACR0							
ACR1	0x11	ACR1							
ACR2	0x12	ACR2							
ACR3	0x13	ACR3							

12. CAN 总线接受屏蔽寄存器 (AMRn)

ACMRn 寄存器各位的含义如表 16.23 所示。ACMRn 寄存器分别位于偏移地址 0x14、0x15、0x16 和 0x17 的位置。

表 16.23 CAN 总线接受屏蔽寄存器 (AMRn) 含义

寄存器名	偏移地址	B7	B6	B5	B4	B3	B2	B1	B0
AMR0	0x14	AMR0							
AMR1	0x15	AMR1							
AMR2	0x16	AMR2							
AMR3	0x17	AMR3							

过滤消息的方式有两种,由模式寄存器中的 AFM(MR.0)位选择。这两种模式为单个过滤器模式 (AFM 位是 1) 和过滤器模式 (AFM 位是 0)。

滤波的规则是:每一位接受屏蔽分别对应每一位接受代码,当该位接受屏蔽位为“1”时 (即设为无关),接收的相应帧 ID 位无论是否与相应的接受代码位相同均会表示为接受;当接受屏蔽位为“0”时 (即设为相关),只有相应的帧 ID 位和相应的接受代码位值相同的情况才会表示为接受。只有在所有的位都表示为接受的时候,CAN 控制器才会接收该报文/消息。

(1) 单个过滤器的配置

这种过滤器配置定义了一个长滤波器 (4 字节、32 位),由 4 个接受代码寄存器和 4 个接受屏蔽寄存器组成的接受过滤器,过滤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。

如图 16.21 所示,接收 CAN 标准帧单个过滤器的配置:对于标准帧,11 位标识符、RTR 位、数据场前两个字节参与过滤;对于参与过滤的数据,所有 AMR 为 0 的位所对应的 ACR 位和参与过滤数据的对应位必须相同才算接收通过;如果由于设置 RTR 为“1”而没有数据字节,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,也会接收报文/消息。对于一个成功接收的报文/消息,所有单个位在滤波器中的比较结果都必须为“接受”。

注:不能使用 AMR1 和 ACR1 的低四位,这些位用于和将来的产品兼容。这些位可通过设置 AMR1.3、AMR1.2、AMR1.1 和 AMR1.0 为 1 而定为“不影响”。



图 16.21 接收 CAN 标准帧时单个过滤器配置

类似的,接收 CAN 扩展帧时单个滤波器配置的结构如图 16.22 所示。



图 16.22 接收 CAN 扩展帧时单过滤器配置

(2) 双过滤器的配置

这种配置可以定义两个短滤波器,由 4 个 ACR 和 4 个 AMR 构成两个短过滤器。只要通过任意一个过滤器,就可以接收总线上的信息。过滤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。

如图 16.23 所示,接受 CAN 标准帧双过滤器的配置:如果接收的是标准帧信息,两个过滤器的定义是不一样的。第一个过滤器由 ACR0、ACR1、AMR0、AMR1 以及 ACR3、AMR3 低 4 位组成,11 位标识符、RTR 位和数据场第 1 字节参与过滤;第二个过滤器由 ACR2、AMR2 以及 ACR3、AMR3 高 4 位组成,11 位标识符和 RTR 位参与滤波。为了成功接收信息,在所有单个位的比较时,应至少有一个过滤器表示接受。RTR 位置为“1”或数据长度代码是“0”,表示没有数据字节存在;只要从开始到 RTR 位的部分都被表示接收,信息就可以通过过滤器 1。如果没有数据字节向过滤器请求过滤,必须将 AMR1 和 AMR3 的低 4 位设置为“1”,即“不影响”。此时,两个过滤器的识别工作都是验证包括 RTR 位在外的整个标准识别码。

类似的,接收 CAN 扩展帧时双滤波器配置的结构如图 16.24 所示。



图 16.23 接收 CAN 标准帧时双过滤器配置

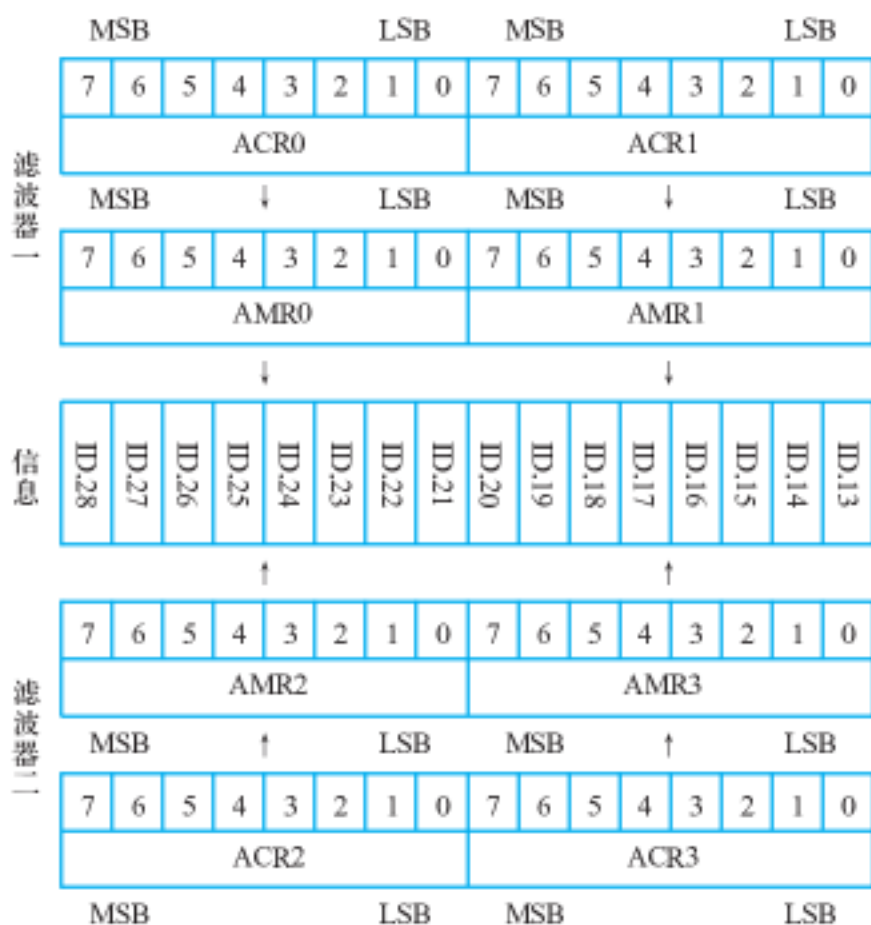


图 16.24 接收 CAN 扩展帧时双过滤器配置

13. CAN 总线错误信息寄存器 (ECC)

该寄存器各位的含义如表 16.24 所示。表中：

表 16.24 CAN 总线错误信息寄存器 (ECC) 含义

符号	地址	B7	B6	B5	B4	B3	B2	B1	B0
ECC	0x18	RXWRN	TXWRN	EDIR	ACKER	FRMER	CRCER	STFER	BER

- (1) RXWRN: 当 RXERR 大于等于 96 时该位置位。
- (2) TXWRN: 当 TXERR 大于等于 96 时该位置位。
- (3) EDIR: 传输错误方向。该位为 0 时, 发送时发生错误; 该位为 1 时, 接收时发生错误。
- (4) ACKER: ACK 错误。
- (5) FRMER: 帧格式错误。
- (6) CRCER: CRC 错误。
- (7) STFER: 位填充错误。
- (8) BER: 位错误。

14. CAN 总线接收错误计数器 (RXERR)

该寄存器各位的含义如表 16.25 所示。

表 16.25 CAN 总线接收错误计数器 (RXERR) 含义

符号	地址	B7	B6	B5	B4	B3	B2	B1	B0
RXERR	0x19	RXERR							

RXERR 表示计数器值代表当前接收错误计数值。该寄存器只读。当关闭总线事件发生后, 该计数器值由硬件清零。

15. CAN 总线发送错误计数器 (TXERR)

该寄存器各位的含义如表 16.26 所示。

表 16.26 CAN 总线发送错误计数器 (TXERR) 含义

符号	地址	B7	B6	B5	B4	B3	B2	B1	B0
TXERR	0x1A	TXERR							

TXERR 表示发送错误计数寄存器反映了发送错误计数器的当前值。在 CAN 控制器工作时, 该寄存器为只读。在硬件复位后, 将该寄存器初始化为 0。

当总线关闭事件发生后, 将错误计数器初始化为 127, 以计算总线定义的最小时间 (128 个总线空闲信号)。该段时间里读发送错误计数器将反映出总线关闭恢复的状态信息。

16. CAN 总线仲裁丢失寄存器 (ALC)

该寄存器各位的含义如表 16.27 所示。

表 16.27 CAN 总线仲裁丢失寄存器 (ALC) 含义

符号	地址	B7	B6	B5	B4	B3	B2	B1	B0
ALC	0x1B	-	-	-	ALC[4:0]				

ALC 说明这个寄存器包括了仲裁丢失的位置的信息。当处理完当前仲裁丢失中断(应答)之后,该值会在下一次仲裁丢失时更新。具体数值如表 16.28 所示。

表 16.28 具体数值对应

ALC[4:0]	数值	描述
00000	0	Arbit lost in ID28/11
00001	1	Arbit lost in ID27/10
00010	2	Arbit lost in ID26/9
00011	3	Arbit lost in ID25/8
00100	4	Arbit lost in ID24/7
00101	5	Arbit lost in ID23/6
00110	6	Arbit lost in ID22/5
00111	7	Arbit lost in ID21/4
01000	8	Arbit lost in ID20/3
01001	9	Arbit lost in ID19/2
01010	10	Arbit lost in ID18/1
01011	11	Arbit lost in ID17/0
01100	12	Arbit lost in SRTR/RTR
01101	13	Arbit lost in IDE bit
01110	14	Arbit lost in ID16 *
01111	15	Arbit lost in ID15 *
10000	16	Arbit lost in ID14 *
10001	17	Arbit lost in ID13 *
10010	18	Arbit lost in ID12 *
10011	19	Arbit lost in ID11 *
10100	20	Arbit lost in ID10 *
10101	21	Arbit lost in ID9 *
10110	22	Arbit lost in ID8 *
10111	23	Arbit lost in ID7 *
11000	24	Arbit lost in ID6 *
11001	25	Arbit lost in ID5 *
11010	26	Arbit lost in ID4 *
11011	27	Arbit lost in ID3 *
11100	28	Arbit lost in ID2 *
11101	29	Arbit lost in ID1 *
11110	30	Arbit lost in ID0 *
11111	31	Arbit lost in RTR