

exCAN

enhanced extended controller area network

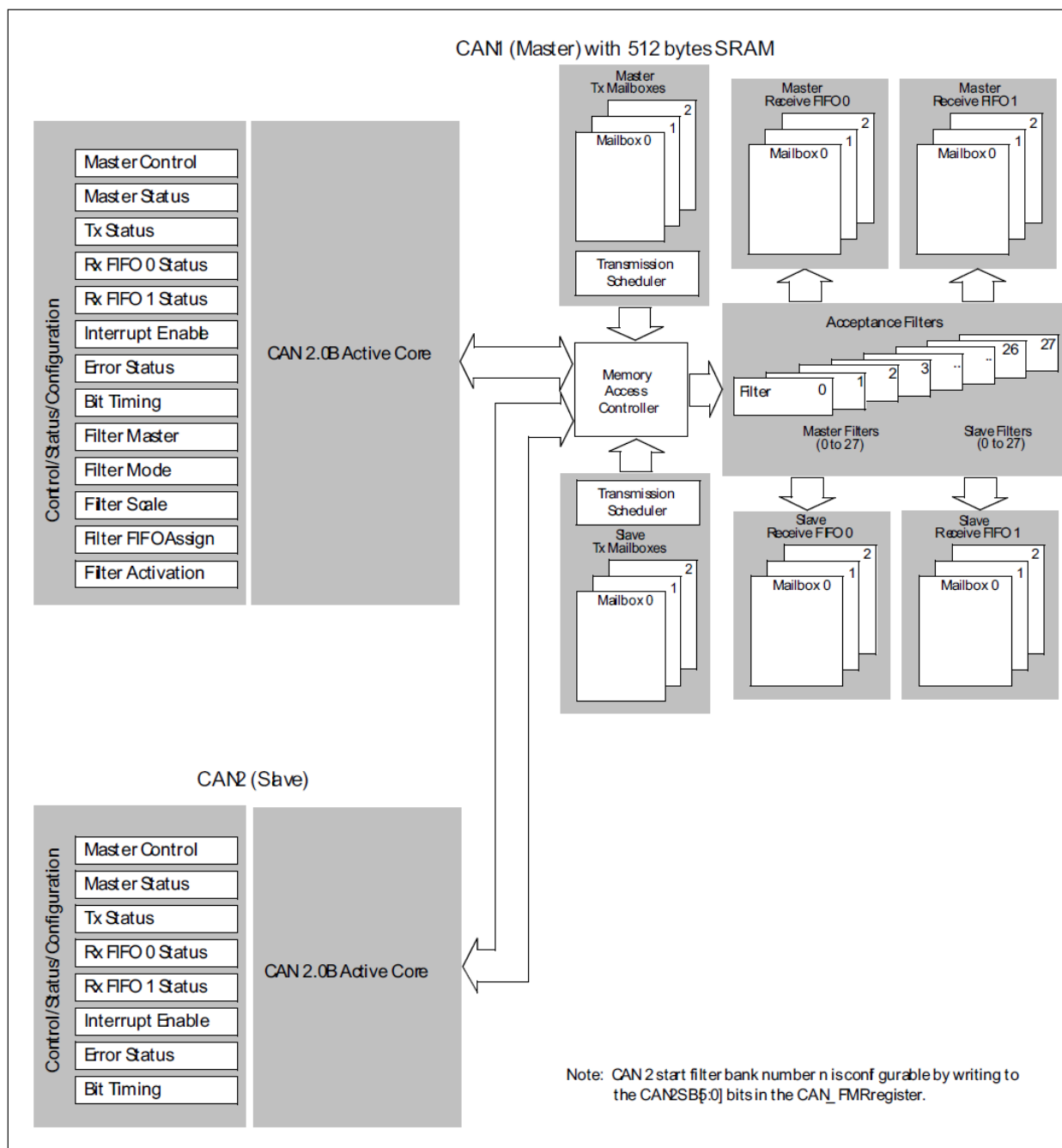
目录

1	简介	3
2	主要特征	4
3	exCAN 引脚.....	4
4	exCAN 地址.....	5
5	exCAN 控制位/寄存器	5
5.1	模块控制位.....	5
5.2	exCAN 通用寄存器.....	5
6	exCAN mailboxes & filters.....	10
6.1	邮箱	10
6.2	过滤器	11
6.3	RAM 组织.....	11
7	exCAN 中断.....	11
8	exCAN 时钟.....	12
9	版本历史	12
10	-	12

1 简介

Enhanced Extended CAN 外设，简称 exCAN，用于 CAN 组网。两个 exCAN 均符合 ISO 11898-1；2015（CAN 协议规范第 2.0 版 A、B 部分）和 CAN FD 协议规范第 1.0 版。

在安全苛求的应用中，exCAN 提供所有的硬件功能用于支持 CAN 的定时触发通讯。



2 主要特征

- 符合 CAN 协议第 2.0 版 A、B 部分
- 不与 USB 共享 SRAM，可与 USB 同时使用
- 支持 ISO 11898-1:2154,-4 (CAN FD)
- 位速率最高 10 Mbit/s
- 支持时间触发

发送

- 3 个发送邮箱
- 中配置发送优先级
- 发送 SOF 时的时间戳

接收

- 2 个接收 FIFO，每个深度为 3 级
- 可扩展的滤波器组：
 - ◇ 单个 CAN 的型号拥有 14 组滤波器
 - ◇ 双 CAN 的型号拥有共用的 28 组滤波器
- 标识符列表功能
- 可配置的 FIFO 溢出
- 接收 SOF 时记录时间戳

时间触发

- 禁止自动重发模式
- 16 位自由运行的计数器
- 最后 2 个字节发送时间戳

管理

- 可屏蔽的中断
- 邮箱映射到唯一的地址空间

双 CAN

- CAN1 和 CAN2 组成双 CAN
- CAN1：主 CAN，管理从 CAN 和 SRAM 之间的访问
- CAN2：从 CAN，不能直接访问 SRAM
- 两个 CAN 共享 SRAM

3 exCAN 引脚

下表列出了 exCAN 的外部引脚。

Signals	Signal type	Description
exCAN1_RX	数字输入	exCAN1 接收引脚
exCAN1_TX	数字输出	exCAN1 发送引脚
exCAN2_RX	数字输入	exCAN2 接收引脚
exCAN2_TX	数字输出	exCAN2 发送引脚

4 exCAN 地址

exCAN 挂在 APB1 总线上，其控制寄存器及 SRAM 地址皆在 APB1 地址空间。

Bus	Boundary address	Size (bytes)	Peripheral
APB1			exCAN3 SRAM
			exCAN1 SRAM
	0x4000 6C00 – 0x4000 6FFF	1 KB	eXCAN3
	0x4000 6800 – 0x4000 6BFF	1 KB	eXCAN2
	0x4000 6400 – 0x4000 67FF	1KB	eXCAN1/USBHD 2 nd 1KB SRAM

注：exCAN 的 SRAM 不提供直接地址访问。

5 exCAN 控制位/寄存器

5.1 模块控制位

RCC_APB1ENR: exCAN 允许

RCC_APB1RSTR: exCAN 复位

	exCAN1EN	exCAN2EN	exCAN3EN	exCAN1RST	exCAN2RST	exCAN3RST
F0		APB1ENR[27]	-		APB1RSTR[27]	-
F1	APB1ENR[25]	APB1ENR[26]	APB1ENR[24]	APB1RSTR[25]	APB1RSTR[26]	APB1RSTR[26]
F3		APB1ENR[27]			APB1RSTR[27]	

5.2 exCAN 通用寄存器

exCAN register map

Offset	Register	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
0x000	MCR																DBF	RESET								TTCM	ABOM	AWUM	NART	RFLM	TXFP	SLEEP	INRQ

	Reset Value																					
0x004	MSR																					
	Reset Value																					
0x008	TSR		LOW[2:0]			TME[2:0]																
	Reset Value	0	0	0	1	1	1	0	0	0	0											
0x00C	RFOR																					
	Reset Value																					
0x010	RF1R																					
	Reset Value																					
0x014	IER																					
	Reset Value									0	0	0										
0x018	ESR																					
	Reset Value	0	0	0																		
0x01C	BTR	SILM	LBCK																			
	Reset Value	0	0																			

0x18C 0x19C 0x1AC	TDHOR TDH1R	DATA7[7:0]								DATA6[7:0]								DATA5[7:0]								DATA4[7:0]									
	TDH2R	Reset Value																																	
0x1B0 0x1C0	RIOR RI1R	STID[10:0]/EXTID[28:18]																EXID[17:0]																IDE	RTR
	Reset Value	Reset Value																																	
0x1B4 0x1C4	RDTOR RDT1R	TIM[15:0]																FMI[7:0]								FDL	R1	BRS	ESI	DLC[3:0]					
	Reset Value	Reset Value																																	
0x1B8 0x1C8	RDLOR RDL1R	DATA3[7:0]								DATA2[7:0]								DATA1[7:0]								DATA0[7:0]									
	Reset Value	Reset Value																																	
0x1BC 0x1CC	RDHOR RDH1R	DATA7[7:0]								DATA6[7:0]								DATA5[7:0]								DATA4[7:0]									
	Reset Value	Reset Value																																	
0x200	FMR																	CAN2SB[5:0]																FINIT	
	Reset Value																	0 0 1 1 1 0																1	
0x204	FM1R	FBM[27:0]																																	
	Reset Value	0 0																																	
0x20C	FS1R	FSC[27:0]																																	
	Reset Value	0 0																																	
0x214	FFA1R	FFA[27:0]																																	
	Reset Value	0 0																																	

0x21C	FA1R					FACT[27:0]																													
	Reset Value					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x240 +(n*8)	FnR1	FB[31:0]																																	
	Reset Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x244 +(n*8)	FnR2	FB[31:0]																																	
	Reset Value	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

注:

- 1) $n = 0 \sim 27$
- 2) **FDCTLR**、**FTDCR**、**DBTR** 仅在初始化模式能够写入
- 3) **FDCTLR**→**FDOE** 必须先配置好，再操作 CAN SRAM（过滤器、邮箱）；**FDOE** 值如果更改，须重新初始化过滤器及邮箱。因为 **FDOE=0** 和 **1** 时 CAN SRAM 中的过滤器、邮箱排列不一致。

TX[1:0]: 发送引脚控制

- 0: CANx_TX 由 CAN 内核控制
- 1: 可在引脚 CANx_TX 监控采样点
- 2: 引脚 CANx_TX 为显性（“0”）电平
- 3: 引脚 CANx_TX 为隐性（“1”）电平

BTCANE: Better CAN Enable

- 0: disable
- 1: Enable。在 FDCAN 模式时，在 BRS 位结束时切换位速率，而不是在 SP 点

R1: FDCAN 时的保留位，仅在 BTCANE=1 时有效

- 0: 发送、接收的 R1 位为显性电平，为 FDCAN 模式
- 1: 发送、接收的 R1 位为隐性电平，在 BRS=1 时，位速率切换发生在 BRS 位结束，必须为支持 BTCAN 的接口才能使用该功能

EHEn: Enhance Enable 增强功能使能

- 0: 允许增强功能，提高 FDCAN 兼容性
- 1: 禁止增强功能

ESIMOD: 错误状态指示模式

- 0: 发送帧的 ESI 位始终显示节点错误状态
- 1: 当节点处于主动错误状态时，通过 TDTxR 寄存器的 ESI 位值决定发送帧的 ESI 位显示的错误状态。当节点处于被动错误状态时，发送帧的该位发送隐性位

TDCMOD: 传输延迟补偿模式

- 0: 测量值加软件偏移
- 1: 仅使用软件偏移

TDCE: 传输延迟补偿使能

- 0: 传输延迟补偿禁止
- 1: 传输延迟补偿使能

NISO: ISO/BOSCH 模式



- 0: ISO
1: BOSCH
- PXHD: 协议异常事件检测禁用
0: 协议异常事件检测使能（当接收到隐性保留位时转为 IDLE 状态）
1: 协议异常事件检测禁用（当接收到隐性保留位时，将此视为帧格式错误）
- BRSE: 位速率翻转使能
0: 禁止位速率翻转
1: 允许位速率翻转，由 TDTxR 寄存器的 BRS 位确定是否使用位速率翻转
- FDOE: FD 操作允许
0: 禁止 CAN-FD 功能
1: 允许 CAN-FD 功能
- PXE: 协议异常事件状态
当协议异常事件被检测到时该位由硬件置 1，软件写 1 清 0
- TDCV[6:0]: 传输延迟补偿值
由硬件控制，显示硬件自动计算的传输延迟补偿值
- TDCO[6:0]: 传输延迟补偿偏移
这些位被用于设置次级采样点（SSP）基于测量的补偿时间的偏移，测量的补偿时间是由硬件计算信号从 CAN_TX 发出到从 CAN_RX 接收到的过程的延迟时间得出的
- TDCF[6:0]: 传输延迟补偿过滤器
这些位定义了次级采样点（SSP）的最小值。当配置的 TDCF 值大于 TDCV 时，这些位的值用于替代 TDCV，以过滤毛刺引起的测量错误导致的 SSP 采样点过早的情况发生
- DJSW[3:0], DTS2[3:0], DTS1[4:0], DBRP[9:0]: 位速率定义
- TDLxR: 当使能 CAN-FD 时，待发送数据通过该寄存器连续写入，最多 16 个字
RDLxR: 当使能 CAN-FD 时，待读取数据通过该寄存器连续读出，最多 16 个字

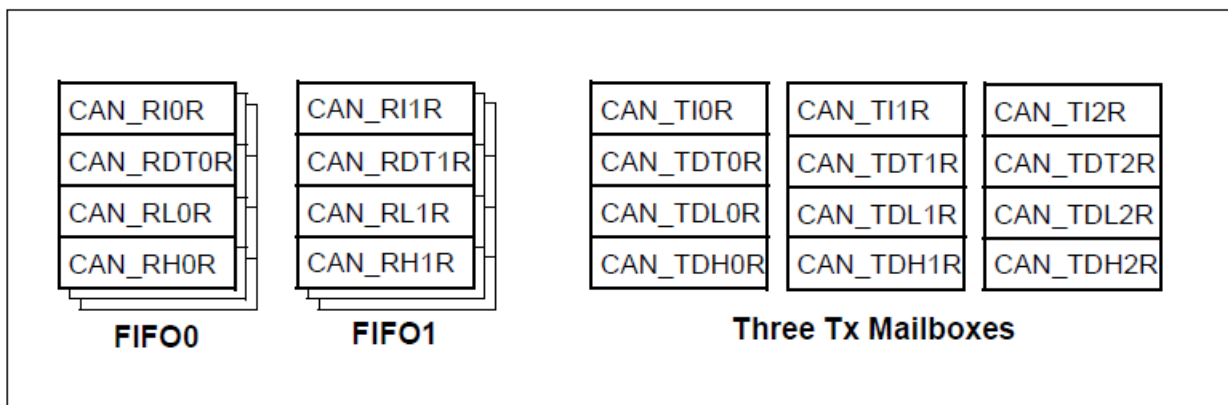
6 exCAN mailboxes & filters

6.1 邮箱

邮箱由发送邮箱和接收邮箱组成，发送邮箱和接收邮箱包含相同的寄存器，只有以下不同：

- CAN_RDTxR 寄存器中的 FMI 位
- 接收邮箱总是写保护（只读）
- 发送邮箱只有为空时才能写入，即 CAN_TSR->TME 位为 1

总共包含 3 个发送（TX）邮箱和 2 个接收（RX）邮箱。每个 RX 邮箱含有 3 级 FIFO，MCU 总是访问最早接收到的消息。每个邮箱由 4 个寄存器组成。



6.2 过滤器

exCAN 包含 28 个过滤器组 (bank)。如果是双 CAN 模式，则 CAN1 和 CAN2 共享这 28 组过滤器。通过 FMR->CAN2SB 来调节各个 CAN 所拥有的过滤器组。以 CAN2SB 为界线，从 CAN2SB 开始的过滤器组，给 CAN2 使用；编号小于 CAN2SB 的过滤器组，给 CAN1 使用。

如果 CAN2SB=0，则所有过滤器组给 CAN2 使用；如果 CAN2SB=28，则所有过滤器组给 CAN1 使用。

6.3 RAM 组织

邮箱和过滤器都位于 CAN 的专用 RAM 中，并按发送邮箱、接收邮箱、过滤器组的顺序存放。
该 RAM 不能通过 MCU 的 APB 总线直接访问，而只能通过 CAN 的 SFR 地址访问。

7 exCAN 中断

exCAN 的中断向量如下所示。

Position	Priority	Type of priority	Acronym	Description	Address
F0 series					
30	37	settable	CEC_exCAN	CEC and exCAN global interrupts (combined with EXTI line 27)	0x0000 00B8
F1/F3 series					
19	26	settable	CAN1_TX	CAN1 TX interrupts	0x0000 008C
20	27	settable	CAN1_RX0	CAN1 RX0 interrupts	0x0000 0090
21	28	settable	USB_HP/ CAN1_RX1	USB High Priority/ CAN1 RX1 interrupts	0x0000 0094
22	29	settable	USB_LP/ CAN1_SCE	USB Low Priority/ CAN1 SCE interrupts	0x0000 0098

63	70	settable	CAN2_TX	CAN2 TX interrupts	0x0000 013C
64	71	settable	CAN2_RX0	CAN2 RX0 interrupts	0x0000 0140
65	72	settable	CAN2_RX1	CAN2 RX1 interrupts	0x0000 0144
66	73	settable	CAN2_SCE	CAN2 SCE interrupts	0x0000 0148

8 exCAN 时钟

exCAN 的时钟为 PCLK 时钟。

9 版本历史

Date	Revision	Author	Changes
2022/10/6	0.10	Dick Hou	初版
2022/11/2	0.11		增加 TS1[4]、TS2[3]、HWSYNCFE 位
2022/11/3	0.12		调整 BTR、DBTR 相关位
2022/11/5	0.13		调整 EHEn 位 增加 BTCANEN、R1 位
2023/4/19	0.14		BTCANEN 重命名为 BTCANE，TDCEN 重命名为 TDCE

10 -