

XC164CM

内嵌 C166SV2 核的 16 位单片微控制器
第 2 卷（共 2 卷）：外设单元

Microcontrollers



Never stop thinking

Edition 2006 - 03

**Published by
Infineon Technologies AG
81726 München, Germany**

**© Infineon Technologies AG 2006.
All Rights Reserved.**

Legal Disclaimer

The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffenheitsgarantie"). With respect to any examples or hints given herein, any typical values stated herein and/or any information regarding the application of the device, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

Information

For further information on technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies Office (www.infineon.com).

Warnings

Due to technical requirements components may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies Office.

Infineon Technologies Components may only be used in life-support devices or systems with the express written approval of Infineon Technologies, if a failure of such components can reasonably be expected to cause the failure of that life-support device or system, or to affect the safety or effectiveness of that device or system. Life support devices or systems are intended to be implanted in the human body, or to support and/or maintain and sustain and/or protect human life. If they fail, it is reasonable to assume that the health of the user or other persons may be endangered.

XC164CM

内嵌 C166SV2 核的 16 位单片微控制器
第 2 卷（共 2 卷）：外设单元

Microcontrollers



Never stop thinking

XC164CM, 第 2 卷（共 2 卷）：外设单元

版本信息：V1.2, 2006 - 03

先前的版本: V1.1, 2005-11
V1.0, 2005-06

页	内容
从版本 V1.1, 2005-11 到版本 V1.2, 2006-02 的主要修正	
19-34	修改 Br8 波特率（表格最后一行）。
21-9	更正图 21-9。
21-10	更正公式。
21-33, 21-34	调换两张图的位置。
从版本 V1.0, 2005-06 到版本 V1.1, 2005-11 的主要修正	
21-83	TwinCAN 的实现：更正对不存在引脚（P4）的错误描述。
17-1...	CAPCOM2：更正对 T0 和 T1（CAPCOM1）的错误描述。
16-4, 16-7	ADC 采样时间 – 更正公式（位域 ADSTC）。

期待您的指正

本手册中如有不当、错误及遗漏之处，敬请批评指正，以便我们不断改进用户手册的质量。请将您的建议（以及本手册的相关参考资料）发送至：

mcdocu-chinesecomments.XIY@infineon.com



声明：本中文用户手册是基于英文版本的翻译，如出现与英文用户手册不符之处，请以英文用户手册为主。

目录:

本用户手册分为两卷：“系统单元”和“外设单元”。为了便于用户的使用，目录（以及关键字索引）将两卷中的章节（以及关键字）统一列出，以便用户快速查阅相关内容（卷[1]或卷[2]）。

1	简介	1-1
1.1	16 位微控制器系列产品	1-3
1.2	基本特性	1-5
1.3	缩写	1-10
1.4	命名规则	1-11
2	体系架构概况	2-1
2.1	基本 CPU 概念及优化	2-2
2.1.1	高指令带宽/快速执行	2-5
2.1.2	功能强大的执行单元	2-6
2.1.3	高性能的分支、调用和循环处理	2-7
2.1.4	统一、优化的指令格式	2-8
2.1.5	可编程多优先级中断系统	2-9
2.1.6	系统资源接口	2-10
2.2	片上系统资源	2-11
2.3	片上外设模块	2-14
2.4	时钟产生	2-28
2.5	功率管理特性	2-29
2.6	片上调试支持（OCDS）	2-30
2.7	保护位	2-31
3	存储器结构	3-1
3.1	地址映射	3-3
3.2	特殊功能寄存器区	3-5
3.3	数据存储区	3-9
3.4	程序存储区	3-11
3.5	系统堆栈	3-13
3.6	IO 区	3-14
3.7	存储器边界越界	3-15

3.8	片上程序 Flash 模块	3-16
3.8.1	Flash 工作模式	3-18
3.8.2	命令序列	3-19
3.8.3	纠错与数据完整性	3-25
3.8.4	保护与安全特征	3-28
3.8.5	Flash 状态信息	3-33
3.8.6	操作控制和错误处理	3-36
3.9	程序存储器控制	3-38
3.9.1	地址映射	3-39
3.9.2	Flash 存储器访问	3-40
3.9.3	IMB 控制功能	3-42
4	中央处理器（CPU）	4-1
4.1	CPU 的组成	4-3
4.2	指令读取和程序流控制	4-4
4.2.1	分支检测和分支预测规则	4-6
4.2.2	预测正确的指令流	4-7
4.2.3	预测错误的指令流	4-9
4.3	指令处理流水线	4-11
4.3.1	通用寄存器引起的流水线冲突	4-12
4.3.2	间接寻址模式引起的流水线冲突	4-15
4.3.3	存储器带宽引起的流水线冲突	4-16
4.3.4	CPU-SFR 更新引起的流水线冲突	4-19
4.4	CPU 配置寄存器	4-26
4.5	通用寄存器的使用	4-30
4.5.1	GPR 寻址模式	4-33
4.5.2	上下文切换	4-35
4.6	代码寻址	4-40
4.7	数据寻址	4-42
4.7.1	短寻址模式	4-43
4.7.2	长寻址模式	4-45
4.7.3	间接寻址模式	4-49
4.7.4	DSP 寻址模式	4-51
4.7.5	系统堆栈	4-57

4.8	标准数据处理	4-61
4.8.1	16 位加法器/减法器、阵列移位器和 16 位逻辑单元	4-64
4.8.2	位操作单元	4-65
4.8.3	乘除单元	4-66
4.9	DSP 数据处理（MAC 单元）	4-68
4.9.1	数字的表示和舍入	4-70
4.9.2	16 位×16 位有符号/无符号乘法器和换算器	4-71
4.9.3	级联单元	4-71
4.9.4	一位换算器	4-71
4.9.5	40 位加法/减法器	4-71
4.9.6	数据限制器	4-71
4.9.7	累加移位器	4-72
4.9.8	40 位有符号累加器寄存器	4-73
4.9.9	MAC 单元状态字 MSW	4-74
4.9.10	重复计数器 MRW	4-76
4.10	常数寄存器	4-78
5	中断与强制中断功能	5-1
5.1	中断系统结构	5-2
5.2	中断仲裁与控制	5-4
5.3	中断向量表	5-9
5.4	外围事件控制器通道的操作	5-18
5.4.1	PEC 源指针与目的指针	5-22
5.4.2	PEC 传送控制	5-25
5.4.3	支持数据链的通道链接模式	5-26
5.4.4	PEC 中断控制	5-27
5.5	中断和 PEC 服务请求的优先级排序	5-29
5.6	上下文切换与状态保存	5-31
5.7	中断节点共享	5-34
5.8	外部中断	5-35
5.9	OCDS 请求	5-41
5.10	中断服务请求的响应延迟	5-41
5.11	强制中断功能	5-43
6	通用系统控制功能	6-1

6.1	系统复位	6-2
6.1.1	复位源和复位阶段	6-3
6.1.2	复位后的状态	6-5
6.1.3	特定应用的初始化	6-8
6.1.4	系统启动配置	6-10
6.1.5	复位行为控制	6-12
6.2	时钟产生	6-13
6.2.1	振荡器	6-14
6.2.2	时钟产生和频率控制	6-16
6.2.3	时钟分配	6-23
6.2.4	振荡器看门狗	6-24
6.2.5	中断产生	6-24
6.2.6	外部时钟信号的产生	6-25
6.3	中央系统控制功能	6-29
6.3.1	状态指示	6-32
6.3.2	复位源指示	6-33
6.3.3	外设关闭握手机制	6-34
6.3.4	调试系统控制	6-35
6.3.5	寄存器安全机制	6-37
6.4	功率管理	6-41
6.4.1	功耗降低模式	6-41
6.4.2	降低时钟频率	6-44
6.4.3	灵活的外设管理	6-44
6.5	看门狗定时器（WDT）	6-46
6.6	ID 控制模块	6-51
7	并行端口	7-1
7.1	输入阈值控制	7-2
7.2	输出驱动器控制	7-3
7.3	端口复用功能	7-7
7.4	P1 口	7-8
7.5	P3 口	7-19
7.6	P5 口	7-37
7.7	P9 口	7-42

8	专用引脚	8-1
9	LXBus 控制器（EBC）	9-1
9.1	时序原则	9-2
9.1.1	基本总线周期协议	9-2
9.1.2	总线周期示例：最快访问周期	9-3
9.2	功能描述	9-4
9.2.1	配置寄存器简介	9-4
9.2.2	时序配置寄存器 TCONCS7	9-4
9.2.3	功能配置寄存器 FCONCS7	9-6
9.2.4	地址窗选择寄存器 ADDRSEL7	9-7
9.2.5	对 TwinCAN 的访问控制	9-9
9.2.6	关机控制	9-9
9.3	EBC 寄存器表	9-10
10	引导程序加载器	10-1
10.1	进入引导程序加载模式	10-2
10.2	加载启动代码	10-4
10.3	退出引导程序加载模式	10-4
10.4	选择 BSL 波特率	10-5
11	调试系统	11-1
11.1	简介	11-1
11.2	调试接口	11-2
11.3	OCDS 模块	11-3
11.3.1	调试事件	11-5
11.3.2	调试动作	11-6
11.4	Cerberus	11-7
11.4.1	功能概述	11-7
12	指令集概述	12-1
13	器件规范	13-1

14	通用定时器单元	14-1
14.1	定时器模块 GPT1	14-2
14.1.1	GPT1 核心定时器 T3 的控制	14-4
14.1.2	GPT1 核心定时器 T3 的工作模式	14-8
14.1.3	GPT1 辅助定时器 T2/T4 的控制	14-15
14.1.4	GPT1 辅助定时器 T2/T4 的工作模式	14-18
14.1.5	GPT1 时钟信号控制	14-28
14.1.6	GPT1 定时器寄存器	14-31
14.1.7	GPT1 定时器的中断控制	14-32
14.2	定时器模块 GPT2	14-33
14.2.1	GPT2 核心定时器 T6 的控制	14-35
14.2.2	GPT2 核心定时器 T6 的工作模式	14-39
14.2.3	GPT2 辅助定时器 T5 的控制	14-42
14.2.4	GPT2 辅助定时器 T5 的工作模式	14-45
14.2.5	GPT2 寄存器 CAPREL 工作模式	14-50
14.2.6	GPT2 时钟信号控制	14-55
14.2.7	GPT2 定时器寄存器	14-58
14.2.8	GPT2 定时器和 CAPREL 的中断控制	14-59
14.3	GPT 模块接口	14-60
15	实时时钟	15-1
15.1	确定 RTC 的时间基准	15-2
15.2	RTC 运行控制	15-5
15.3	RTC 工作模式	15-6
15.3.1	48 位定时器操作	15-9
15.3.2	系统时钟操作	15-9
15.3.3	周期性中断的产生	15-10
15.4	RTC 中断产生	15-11
16	模数转换器	16-1
16.1	模式选择	16-4
16.1.1	兼容模式	16-4
16.1.2	增强模式	16-6
16.2	ADC 操作	16-10
16.2.1	固定通道转换模式	16-13

16.2.2	自动扫描转换模式	16-14
16.2.3	等待读取模式	16-15
16.2.4	通道插入模式	16-16
16.3	自动校准	16-19
16.4	转换时间控制	16-20
16.5	ADC 中断控制	16-23
16.6	ADC 模块接口	16-24
17	捕获/比较单元 CAPCOM2	17-1
17.1	CAPCOM2 定时器	17-4
17.2	CAPCOM2 定时器中断	17-9
17.3	捕获/比较通道	17-10
17.4	捕获模式操作	17-13
17.5	比较模式操作	17-14
17.5.1	比较模式 0	17-15
17.5.2	比较模式 1	17-15
17.5.3	比较模式 2	17-18
17.5.4	比较模式 3	17-18
17.5.5	双寄存器比较模式	17-22
17.6	比较输出信号的产生	17-25
17.7	单次事件操作	17-27
17.8	交错和非交错操作	17-29
17.9	CAPCOM2 中断	17-34
17.10	外部输入信号的要求	17-36
17.11	CAPCOM2 单元接口	17-37
18	捕获/比较单元 6 (CAPCOM6)	18-1
18.1	定时器 T12 模块	18-4
18.1.1	定时器 T12 的操作	18-7
18.1.2	T12 的比较模式	18-12
18.1.3	死区时间产生	18-22
18.1.4	T12 捕获模式	18-25
18.1.5	类磁滞控制模式	18-30
18.2	定时器 T13 模块	18-31
18.2.1	定时器 T13 的操作	18-34

18.2.2	T13 的比较模式	18-39
18.3	定时器模块控制	18-44
18.4	多通道模式	18-51
18.5	霍尔传感器模式	18-54
18.5.1	霍尔序列比较逻辑	18-55
18.5.2	霍尔序列采样	18-56
18.5.3	用定时器 T12 模块实现无刷直流电机控制	18-57
18.5.4	霍尔模式标志位	18-59
18.6	强制中断处理	18-65
18.7	输出调制控制	18-69
18.8	映射寄存器传送控制	18-74
18.9	中断产生	18-76
18.10	暂停模式	18-85
18.11	CAPCOM6 单元接口	18-86
19	异步/同步串行接口（ASC）	19-1
19.1	操作概述	19-3
19.2	异步工作	19-5
19.2.1	异步数据帧	19-6
19.2.2	异步发送	19-9
19.2.3	发送 FIFO 操作	19-9
19.2.4	异步接收	19-12
19.2.5	接收 FIFO 操作	19-12
19.2.6	FIFO 透明模式	19-15
19.2.7	IrDA 模式	19-16
19.2.8	异步模式中 RxD/TxD 数据路径选择	19-18
19.3	同步操作	19-19
19.3.1	同步发送	19-20
19.3.2	同步接收	19-20
19.3.3	同步时序	19-20
19.4	波特率产生	19-22
19.4.1	异步模式下的波特率产生	19-22
19.4.2	同步模式下的波特率产生	19-26
19.5	自动波特率检测	19-27

19.5.1	操作概述	19-27
19.5.2	串行帧自动波特率检测	19-28
19.5.3	波特率选择和计算	19-30
19.5.4	自动波特率检测成功后，改写寄存器	19-34
19.6	硬件错误检测功能	19-35
19.7	中断	19-36
19.8	寄存器	19-40
19.9	ASC 模块接口	19-59
20	高速同步串行接口	20-1
20.1	介绍	20-1
20.2	工作概述	20-1
20.2.1	工作模式选择	20-3
20.2.2	全双工操作	20-9
20.2.3	半双工操作	20-11
20.2.4	连续传送	20-13
20.2.5	波特率产生	20-13
20.2.6	检错机制	20-15
20.2.7	SSC 寄存器总结	20-17
20.2.8	端口配置要求	20-18
20.3	SSC 模块接口	20-19
21	TwinCAN 模块	21-1
21.1	内核描述	21-1
21.1.1	概述	21-1
21.1.2	TwinCAN 控制外壳	21-4
21.1.2.1	初始化处理	21-4
21.1.2.2	中断请求压缩器	21-5
21.1.2.3	全局控制和状态逻辑	21-6
21.1.3	CAN 节点控制逻辑	21-7
21.1.3.1	概述	21-7
21.1.3.2	定时控制单元	21-9
21.1.3.3	位流处理器	21-11
21.1.3.4	错误处理逻辑	21-11
21.1.3.5	节点中断处理	21-12

21.1.3.6	报文中断处理	21-13
21.1.3.7	中断指示	21-13
21.1.4	报文处理单元	21-15
21.1.4.1	仲裁和验收屏蔽寄存器	21-16
21.1.4.2	数据帧和远程帧处理	21-17
21.1.4.3	发送报文对象处理	21-18
21.1.4.4	接收报文对象处理	21-20
21.1.4.5	单数据传送模式	21-22
21.1.5	CAN 报文对象缓存（FIFO）	21-23
21.1.5.1	CAN 控制器对缓存区的访问	21-25
21.1.5.2	CPU 对缓存区的访问	21-26
21.1.6	网关报文处理	21-27
21.1.6.1	正常网关模式	21-28
21.1.6.2	带 FIFO 缓存的正常网关模式	21-32
21.1.6.3	共享网关模式	21-35
21.1.7	TwinCAN 模块编程	21-40
21.1.7.1	CAN 节点 A/B 设置	21-40
21.1.7.2	报文对象的初始化	21-40
21.1.7.3	报文传送控制	21-41
21.1.8	回环模式	21-44
21.1.9	单次发送试验功能	21-45
21.1.10	模块时钟要求	21-45
21.2	TwinCAN 寄存器描述	21-46
21.2.1	寄存器映射	21-46
21.2.2	CAN 节点 A/B 寄存器	21-48
21.2.3	CAN 报文对象寄存器	21-63
21.2.4	全局 CAN 控制/状态寄存器	21-80
21.3	XC164CM 模块的实现	21-82
21.3.1	TwinCAN 模块接口	21-82
21.3.2	与 TwinCAN 模块相关的外部寄存器	21-83
21.3.2.1	系统寄存器	21-84
21.3.2.2	端口寄存器	21-85
21.3.2.3	中断寄存器	21-87

21.3.3	寄存器表	21-88
22	寄存器组	22-1
22.1	PD+BUS 外设	22-1
22.2	LXBUS 外设	22-13
关键字索引		1

14 通用定时器单元

通用定时器单元的 GPT1 和 GPT2 模块具有非常灵活的多功能定时器结构，可用作定时、事件计数、脉宽测量、脉冲生成、倍频及其它用途。五个 16 位定时器组成两个定时器模块 GPT1 和 GPT2。每个模块中的各个定时器均可独立工作在不同模式，如门控定时器模式、计数器模式，或者和同模块中的另一个定时器级联。每个模块具有复用输入/输出功能及专用中断。

模块 GPT1 中包含三个定时器/计数器：核心定时器 T3 和两个辅助定时器 T2 和 T4。最大精度为 $f_{GPT}/4$ 。GPT1 的辅助定时器可被配置为核心定时器的重载或捕获寄存器。寄存器描述请参阅[章节 14.1.6](#)。

- 最大精度 $f_{GPT}/4$
- 3 个独立的定时器/计数器
- 定时器/计数器可级联
- 4 种工作模式
 - 定时器模式
 - 门控定时器模式
 - 计数器模式
 - 增量接口模式
- 重载和捕获功能
- 独立的中断线

模块 GPT2 中包含两个定时器/计数器：核心定时器 T6 和辅助定时器 T5。最大精度为 $f_{GPT}/2$ 。附加的捕获/重载寄存器（CAPREL）支持功能扩展的捕获和重载操作。寄存器描述请参阅[章节 14.2.7](#)。核心定时器 T6 可和 CAPCOM 单元的定时器（T7 和 T8）级联。

模块 GPT2 的特性总结如下：

- 最大精度 $f_{GPT}/2$
- 2 个独立的定时器/计数器
- 定时器/计数器可级联
- 3 种工作模式
 - 定时器模式
 - 门控定时器模式
 - 计数器模式
- 由 16 位捕获/重载寄存器 CAPREL 实现的扩展捕获/重载功能
- 独立的中断线

14.1 定时器模块 GPT1

从编程人员的角度来看，GPT1 由一组特殊功能寄存器（SFR）组成（总结见图 14-1）。端口和方向寄存器中用作 GPT1 功能的部分用阴影标出。

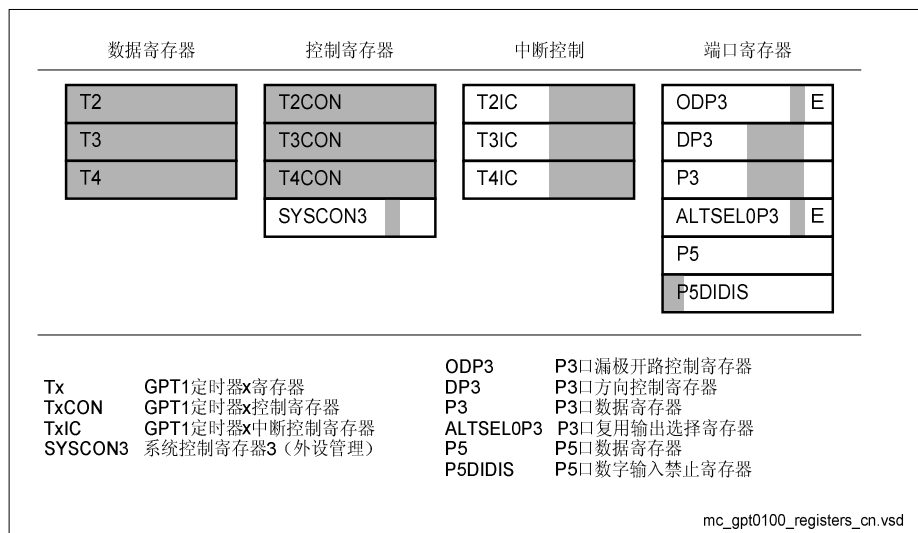


图 14-1 和定时器模块 GPT1 相关的特殊功能寄存器（SFR）

GPT1 的三个定时器（T2，T3，T4）可工作在 4 种基本模式：定时器模式、门控定时器模式、计数器模式或增量接口模式。三个定时器均可递增或递减计数。每个定时器分别由各自的控制寄存器 TxCON 控制。

每个定时器有一个对应的输入引脚 TxIN（复用引脚功能），用作门控定时器模式中的门控信号，或计数器模式中的计数输入。计数方向（递增/递减）可由软件设定，或由外部递增/递减控制输入引脚 TxEUD（复用引脚功能）上的信号动态控制。输出翻转锁存 T3OTL 指示核心定时器 T3 发生上溢/下溢，T3OTL 的状态可从引脚 T3OUT（复用引脚功能）输出。辅助定时器 T2 和 T4 还可以（通过 T3OTL）和核心定时器 T3 级联工作，或者用作核心定时器 T3 的捕获或重载寄存器。

定时器计数寄存器 T2、T3 或 T4 位于不可位寻址的 SFR 区（见章节 14.1.6），CPU 访问这些寄存器，可读取或修改每个定时器的当前计数值。若定时器执行递增、递减、重载或捕获操作时，CPU 恰好正在对（任意）某个定时器寄存器写入，则 CPU 的写操作占优以确保结果正确。

GPT1 的中断由中断控制寄存器 TxIC 控制，这些寄存器不在 GPT1 模块中。GPT1 的输入和输出线与 P3 和 P5 口的引脚相连。端口功能的控制寄存器位于相应的端口模块中。

注：对外部输入信号的时序要求请参阅[章节 14.1.5](#)，包括引脚在内的模块接口信号总结见[章节 14.3](#)。

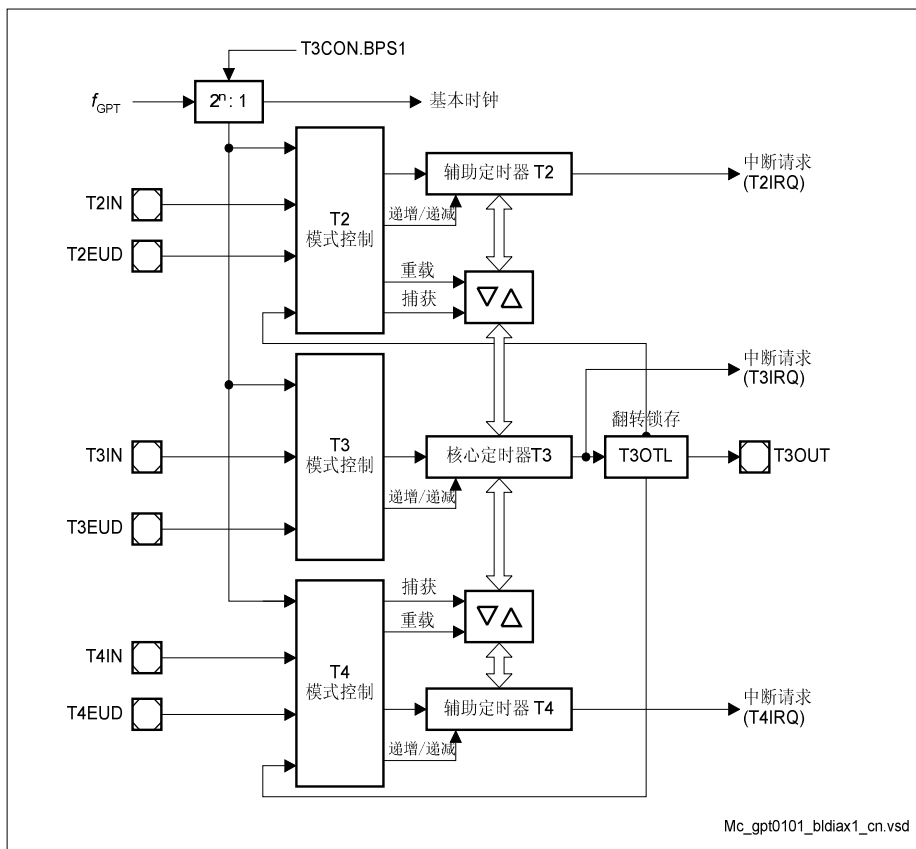


图 14-2 GPT1 框图 (n = 2...5)

14.1.1 GPT1 核心定时器 T3 的控制

可从计数寄存器 T3 中读取核心定时器 T3 的当前值，该寄存器也可由 CPU 写入，例如由 CPU 设置定时器的初始值。

由可位寻址的控制寄存器 T3CON 配置和控制核心定时器 T3。

GPT12E_T3CON

定时器 T3 控制寄存器

SFR (FF42H/A1H)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T3 R DIR	T3 CH DIR	T3 ED GE	BPS1		T3 OTL	T3 OE	T3 UDE	T3 UD	T3R	T3M			T3I		
rh	rwh	rwh	rw		rwh	rw	rw	rw	rw	rw			rw		

符号	位序号	读写类型	功能描述
T3RDIR	15	rh	定时器 T3 旋转方向标志 0 定时器 T3 递增计数 1 定时器 T3 递减计数
T3CHDIR	14	rwh	定时器 T3 计数方向改变标志 每次定时器 T3 的计数方向改变时，该位被置位。T3CHDIR 必须由软件清零。 0 未检测到计数方向改变 1 已检测到计数方向改变
T3EDGE	13	rwh	定时器 T3 计数沿检测标志 每次检测到计数沿，该位被置位。T3EDGE 必须由软件清零。 0 未检测到计数沿 1 已检测到计数沿
BPS1	[12:11]	rw	模块 GPT1 预分频控制 选择模块 GPT1 的基本时钟 （请参阅 章节 14.1.5 ） 00 $f_{GPT}/8$ 01 $f_{GPT}/4$ 10 $f_{GPT}/32$

符号	位序号	读写类型	功能描述
			11 $f_{GPT}/16$
T3OTL	10	rwh	定时器 T3 溢出翻转锁存 每次 T3 上溢/下溢时翻转锁存。可由软件置位或复位。（具体描述请参阅相关章节）
T3OE	9	rw	上溢/下溢输出使能 0 复用输出功能被禁止 1 T3OTL 的状态从引脚 T3OUT 输出
T3UDE	8	rw	定时器 T3 外部递增/递减计数使能¹⁾ 0 输入 T3EUD 被断开 1 计数方向受输入 T3EUD 的控制
T3UD	7	rw	定时器 T3 递增/递减计数控制¹⁾ 0 定时器 T3 递增计数 1 定时器 T3 递减计数
T3R	6	rw	定时器 T3 运行控制位 0 定时器 T3 停止 1 定时器 T3 运行
T3M	[5:3]	rw	定时器 T3 模式控制（基本工作模式） 000 定时器模式 001 计数器模式 010 门控定时器模式，门控信号低电平有效 011 门控定时器模式，门控信号高电平有效 100 保留，不使用该组合 101 保留，不使用该组合 110 增量接口模式 （旋转检测模式） 111 增量接口模式 （边沿检测模式）
T3I	[2:0]	rw	定时器 T3 输入参数选择

符号	位序号	读写类型	功能描述
			对应不同工作模式，输入参数选择请参阅相关内容： 表 14-7 对应定时器模式和门控定时器模式 表 14-2 对应计数器模式 表 14-3 对应增量接口模式

1) 位 T3UD 和 T3UDE 的编码见**表 14-1**。

定时器 T3 运行控制

通过软件设定 T3R（定时器 T3 运行控制位），可启动或终止核心定时器 T3。该控制位对 T3 的所有工作模式均有效。置位 T3R 将启动定时器；清零 T3R 将终止定时器。

门控定时器模式下，只有当 T3R = 1 且门控信号有效（设置为高电平或低电平有效）时，定时器才能工作。

注：当定时器控制寄存器 T2CON 或 T4CON 中的位 T2RC 或 T4RC 被置位时，位 T3R 也将控制辅助定时器 T2 和/或 T4（启动或终止）。

计数方向控制

控制寄存器 TxCON 中的位 TxUD 和 TxUDE 用来选择由软件、或外部输入引脚 TxEUD（定时器 Tx 外部递增/递减控制输入）控制 GPT1 定时器（核心定时器和辅助定时器）的计数方向。由软件控制计数方向时（TxUDE = 0），置位或清零 TxUD 可改变计数方向；位 TxUDE = 1 时，由引脚 TxEUD 控制计数方向。但是，仍可用 TxUD 来改变实际计数方向，如**表 14-1**所示。无论定时器是否运行，均可改变计数方向。

注：当引脚 TxEUD 用作外部计数方向控制输入时，必须被配置为输入引脚（对应的方向控制位必须被清零）。

表 14-1 GPT1 定时器计数方向控制

引脚 TxEUD	位 TxUDE	位 TxUD	计数方向	位 TxRDIR
X	0	0	递增计数	0
X	0	1	递减计数	1
0	1	0	递增计数	0
1	1	0	递减计数	1
0	1	1	递减计数	1
1	1	1	递增计数	0

定时器 T3 输出翻转锁存

定时器 T3 的上溢/下溢信号送入定时器工作模式框图中的“翻转锁存”模块。**图 14-3** 给出该模块的内部逻辑。T3 的上溢或下溢信号是两个锁存器的输入时钟：第一个锁存器代表控制寄存器 T3CON 中的位 T3OTL；第二个锁存器是由 T3OTL 的输出控制的内部翻转锁存器。两个锁存器的输出均送入辅助定时器 T2 和 T4 的输入控制模块。映射锁存器的输出电平与 T3OTL 的输出电平匹配，只是延迟一个时钟周期。当 T3OTL 的值发生变化，将会导致 T3OTL 和映射锁存器的输出电平暂时不同，从而可触发定时器 T2 和/或 T4 的计数事件。

软件写入 T3OTL 时，两个锁存器同时置位或清零。这种情况下，送入辅助定时器的两个信号的电平相同，T2/T4 不会检测到跳变沿。寄存器 T3CON 中的位 T3OE（上溢/下溢输出使能）可使能由外部引脚 T3OUT 监控 T3OTL 的状态。当 T3OTL 连接至外部端口引脚上（必须配置为输出）时，可用 T3OUT 控制外部硬件。若 T3OE = 1，引脚 T3OUT 输出 T3OTL 的状态。若 T3OE = 0，引脚 T3OUT 输出高电平（只要选择此复用端口用作 T3OUT 功能）。

触发信号可用作辅助定时器 T2 和 T4 计数器功能的输入，或者作为 T2 和 T4 重载功能的触发源。

如图 14-3 所示，当软件修改锁存器 T3OTL 的值、以决定输出线状态时，内部映射锁存器也相应被置位或清零。因此 T2/T4 不会检测到触发条件。

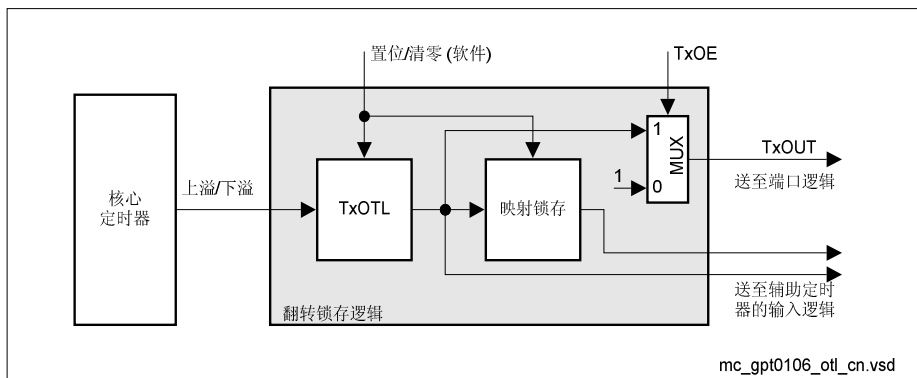


图 14-3 核心定时器 T3 翻转锁存逻辑框图

14.1.2 GPT1 核心定时器 T3 的工作模式

定时器 T3 工作在定时器模式

将寄存器 T3CON 中的位域 T3M 设置为 000_B，核心定时器 T3 工作在定时器模式。定时器模式下，模块输入时钟 f_{GPT} 经两个可编程的预分频因子（由寄存器 T3CON 中的位域 BPS1 和 T3I 控制）分频后，用作 T3 的输入时钟信号。输入时钟选择的详细内容见[章节 14.1.5](#)。

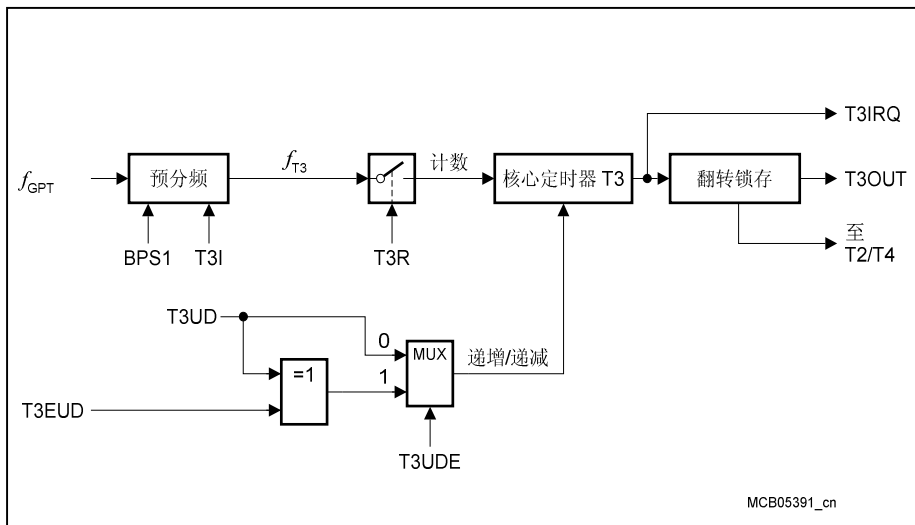


图 14-4 定时器模式下核心定时器 T3 框图

门控定时器模式

将寄存器 T3CON 中的位域 T3M 设置为 010_B 或 011_B，核心定时器 T3 工作在门控定时器模式。位 T3M.0 (T3CON.3) 选择门控输入的有效电平。门控定时器模式下输入时钟的频率选择和定时器模式下相同（请参阅[章节 14.1.5](#)），但该模式下定时器的输入时钟受外部输入引脚 T3IN（定时器 T3 的外部输入）门控控制。

T3 工作在该模式时，引脚 T3IN 必须配置为输入，即对应的方向控制位必须为 0。

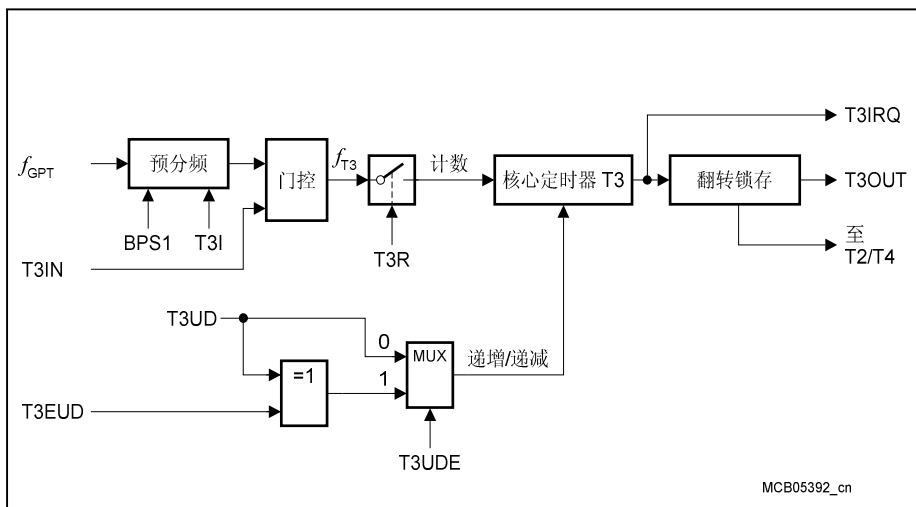


图 14-5 门控定时器模式下核心定时器 T3 框图

若 T3M = 010_B，引脚 T3IN 为低电平时定时器使能，高电平时定时器终止。若 T3M = 011_B，T3IN 必须为高电平才能使能定时器。此外，通过软件修改 T3R 可控制定时器的开启或关闭。只有当 T3R 为 1 且门控信号有效时，定时器才可以工作。若 T3R 为 0 或者门控信号无效，定时器将停止工作。

注：引脚 T3IN 上门控信号的跳变不会引发中断请求。

计数器模式

将寄存器 **T3CON** 中的位域 **T3M** 设置为 **001_B**，核心定时器 **T3** 工作在计数器模式。计数器模式下，外部输入引脚 **T3IN** 上的跳变用作定时器 **T3** 的计数时钟。**T3IN** 上的正跳变、负跳变或任意跳变可触发定时器递增或递减计数。控制寄存器 **T3CON** 中的位域 **T3I** 用来选择触发边沿（见**表 14-2**）。

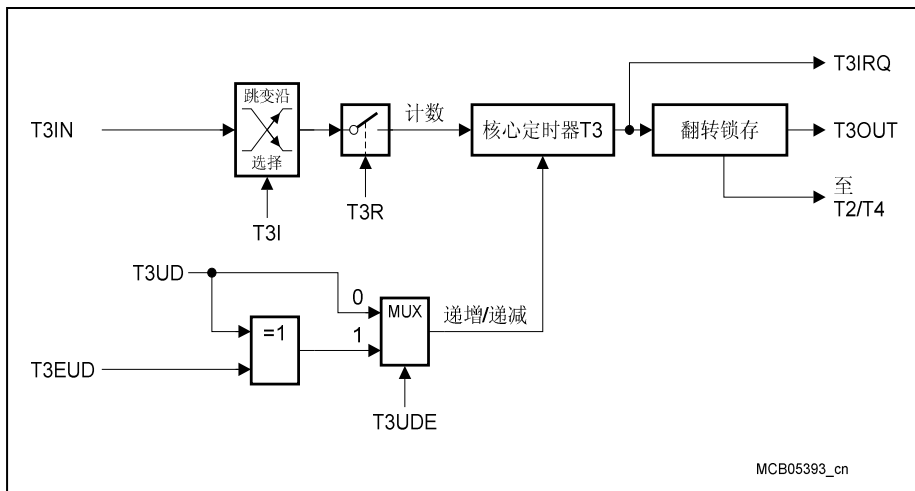


图 14-6 计数器模式下核心定时器 T3 框图

表 14-2 GPT1 核心定时器 T3（计数器模式）输入沿选择

T3I	计数器递增/递减计数触发沿
000	无，计数器 T3 被禁止
001	T3IN 上的正跳变（上升沿）
010	T3IN 上的负跳变（下降沿）
011	T3IN 上的任意跳变（上升沿或下降沿）
1XX	保留，不使用该组合

计数器模式下，必须将引脚 **T3IN** 配置为输入（相应方向控制位 **DPx.y** 必须为 0）。计数器模式下允许的最大输入频率取决于所选择的预分频因子。为了确保能够正

确识别 T3IN 上计数输入信号的跳变，输入电平必须至少保持（高或低）规定数目的模块时钟周期之后才能改变。具体内容请参阅[章节 14.1.5](#)。

增量接口模式

将寄存器 T3CON 中的位域 T3M 设置为 110_B 或 111_B，核心定时器 T3 工作在增量接口模式。增量接口模式下，与核心定时器 T3 相关的两个输入信号（T3IN，T3EUD）用作增量编码器的接口。其中一个或这两个信号上的每次跳变都触发 T3 计数，从而提供了 2 倍或 4 倍于编码器输入的精度。

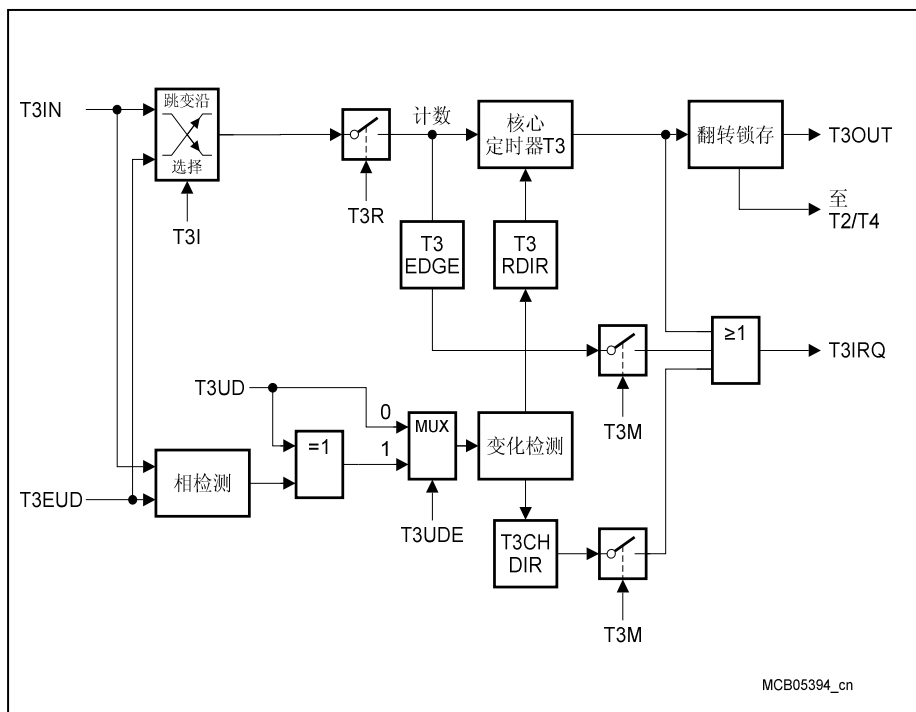


图 14-7 增量接口模式下核心定时器 T3 的框图

控制寄存器 T3CON 中的位域 T3I 控制选择触发沿（见[表 14-3](#)）。两个输入信号的跳变序列经评估后产生计数脉冲和方向信号。因此，根据增量编码器的速度和方向可自动修改 T3 的值，该数值始终表示编码器的当前位置。

产生中断请求（T3IRQ）的方式有：

旋转检测模式下（ $T3M = 110_B$ ），每次 $T3$ 的计数方向改变时产生中断请求；边沿检测模式下（ $T3M = 111_B$ ），每次检测到 $T3$ 的计数沿时产生中断请求。寄存器 $T3CON$ 中的状态位 $T3RDIR$ 、 $T3CHDIR$ 和 $T3EDGE$ 分别监控 $T3$ 的计数方向、计数方向的变化和计数请求。

表 14-3 核心定时器 T3（增量接口模式）输入沿选择

T3I	计数器递增/递减计数触发沿
000	无，计数器 T3 停止
001	T3IN 上的任意跳变（上升沿或下降沿）
010	T3EUD 上的任意跳变（上升沿或下降沿）
011	T3 任意一个输入引脚（T3IN 或 T3EUD）上的任意跳变（上升沿或下降沿）
1XX	保留，不使用该组合

无需外部接口，逻辑增量编码器即可直接和 **XC164CM** 相连。不过，在标准系统中使用比较器将编码器的差分输出（比如 A ， \bar{A} ）转换为数字信号（比如 A ），从而大大增强了抗噪能力。

*注：编码器的第三个输出信号 $T0$ （指示机械零位置）可以和 **XC164CM** 的外部中断输入相连，触发定时器 $T3$ 复位（例如将 **ZEROS** 通过 **PEC** 传送至 $T3$ ）。*

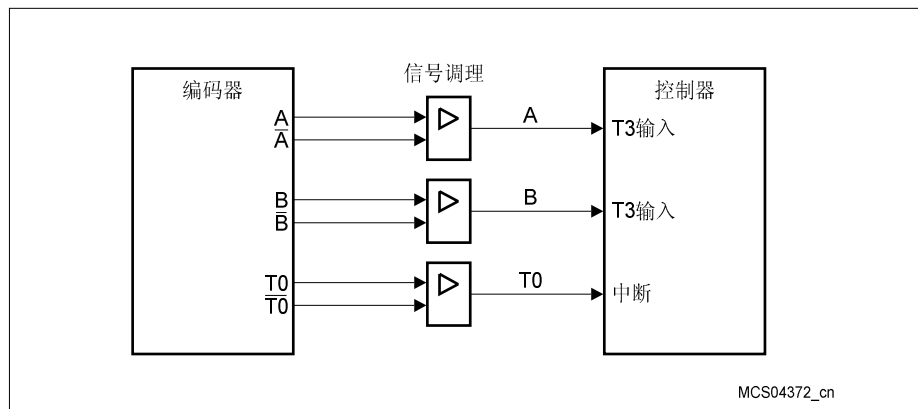


图 14-8 编码器和 XC164CM 的连接

要执行增量接口操作，必须满足以下条件：

- 位域 T3M 必须为 110_B 或 111_B
- 必须将引脚 T3IN 和 T3EUD 配置为输入，即相应的方向控制位必须为 0
- 位 T3UDE 必须为 1，从而使能外部信号自动控制计数方向

增量接口模式下允许的最大计数频率取决于所选择的预分频因子。为了确保能够正确识别任意一个输入信号的跳变，输入电平必须至少保持（高或低）规定数目的模块时钟周期之后才能改变。具体内容请参阅[章节 14.1.5](#)。

由于增量接口模式下，所评估的两个输入信号之间有 90°相移，其最大输入频率可达到最大计数频率的一半。

增量接口模式下，可从输入信号变化的序列中自动获取计数方向信息；该序列和所连接的传感器的旋转方向相对应。各种可能的信号变化组合总结见[表 14-4](#)。

表 14-4 GPT1 核心定时器 T3（增量接口模式）计数方向

另一个输入信号 的电平 (T3EUD/T3IN)	T3IN 输入		T3EUD 输入	
	上升沿	下降沿	上升沿	下降沿
高电平	递减计数	递增计数	递增计数	递减计数
低电平	递增计数	递减计数	递减计数	递增计数

[图 14-9](#) 和 [图 14-10](#) 举例说明计数信号的产生，以及在计数方向的控制下 T3 如何工作，同时说明如何进行输入信号的抖动补偿（传感器位于切换点附近时可能会出现抖动现象）。

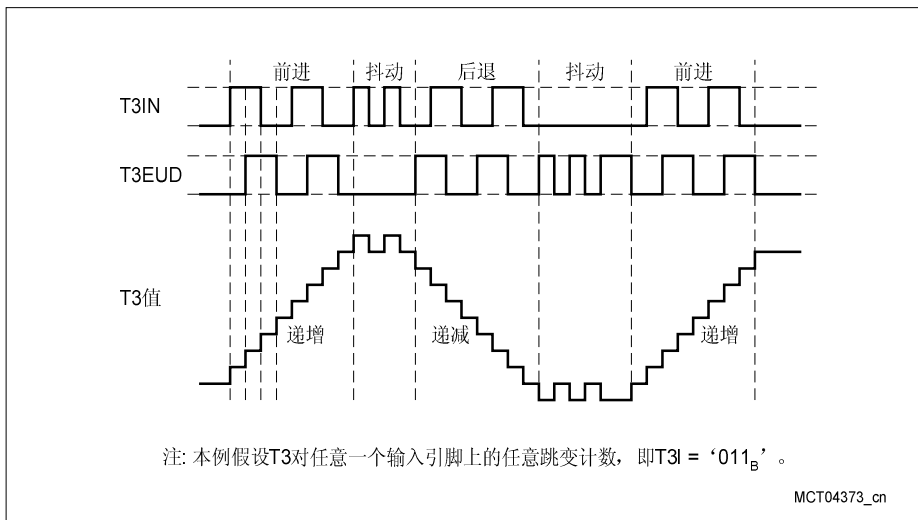


图 14-9 增量编码器信号评估, 2 个计数输入

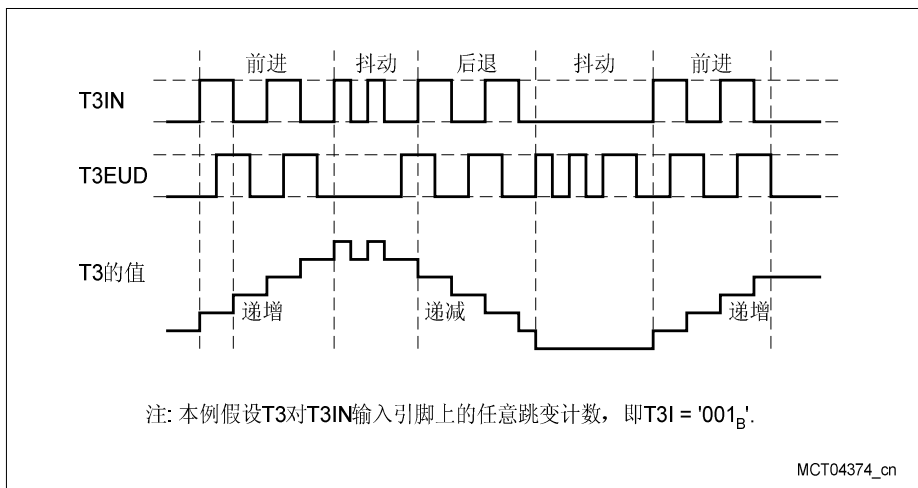


图 14-10 增量编码器信号评估, 1 个计数输入

注: 工作在增量接口模式下的定时器 T3 可自动提供传感器的当前位置信息。通过测量输入信号的周期值可获取传感器的动态信息 (速度、加速度、减速度),

定时器 T5 工作在附加特殊捕获模式下可实现该功能（具体内容请参阅[章节 14.2.5](#)）。

14.1.3 GPT1 辅助定时器 T2/T4 的控制

辅助定时器 T2 和 T4 功能相同。它们可被配置为定时器模式、门控定时器模式、计数器模式或增量接口模式。定时器输入时钟的频率选择以及计数输入信号的选择和核心定时器 T3 相同。除了这四种基本工作模式，辅助定时器还可以和核心定时器级联工作，或者用作核心定时器的重载或捕获寄存器。辅助定时器的启动/终止可由 T3 的运行位远程控制，从而可同步控制多个定时器。

辅助定时器 T2 或 T4 的当前值可分别从计数寄存器 T2 或 T4 中读取。这些寄存器也可由 CPU 写入，例如由 CPU 设置定时器的初始值。

通过可位寻址的控制寄存器 T2CON 和 T4CON（结构完全相同）分别配置定时器 T2 和 T4。请注意：模块 GPT1 中三个定时器均具备的功能由各自的控制寄存器中相同位置的位元/域、以相同的方式进行控制。

注：辅助定时器无输出翻转锁存逻辑，无复用输出功能。

GPT12E_T2CON

定时器 T2 控制寄存器

SFR (FF40_H/A0_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T2 R DIR	T2 CH DIR	T2 ED GE	T2 IR DIS	-	-	T2 RC	T2 UDE	T2 UD	T2R	T2M			T2I		
rw	rw	rw	rw	-	-	rw	rw	rw	Rw	rw			rw		

GPT12E_T4CON

定时器 T4 控制寄存器

SFR (FF44_H/A2_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T4 R DIR	T4 CH DIR	T4 ED GE	T4 IR DIS	-	-	T4 RC	T4 UDE	T4 UD	T4R	T4M			T4I		
rw	rw	rw	rw	-	-	rw	rw	rw	rw	rw			rw		

符号	位序号	读写类型	功能描述
TxRDIR	15	rh	定时器 Tx 旋转方向标志 0 定时器 Tx 递增计数 1 定时器 Tx 递减计数
TxCHDIR	14	rwh	定时器 Tx 计数方向改变标志 每次定时器 Tx 计数方向改变时，该位被置位。TxCHDIR 必须由软件清零。 0 未检测到计数方向改变 1 已检测到计数方向改变
TxEDGE	13	rwh	定时器 Tx 计数沿检测标志 每次检测到计数沿时，该位被置位。TxEDGE 必须由软件清零。 0 未检测到计数沿 1 已检测到计数沿
TxIRDIS	12	rw	定时器 Tx 中断请求禁止控制 0 增量接口模式下，使能 TxCHDIR 和 TxEDGE 产生中断请求 1 增量接口模式下，禁止 TxCHDIR 和 TxEDGE 产生中断请求
TxRC	9	rwh	定时器 Tx 远程控制位 0 定时器 Tx 由自身的运行位 TxR 控制 1 定时器 Tx 由 T3 的运行位 T3R 控制，不由 TxR 控制
TxUDE	8	rw	定时器 Tx 外部递增/递减计数使能¹⁾ 0 输入 TxEUD 被断开 1 计数方向受输入 TxEUD 控制
TxUD	7	rw	定时器 Tx 递增/递减计数控制¹⁾ 0 定时器 Tx 递增计数 1 定时器 Tx 递减计数
TxR	6	rw	定时器 Tx 运行控制 0 定时器 Tx 停止

符号	位序号	读写类型	功能描述
			1 定时器 Tx 运行 <i>注：如果 $TxRC = 0$，该位才可控制定时器 Tx。</i>
TxM	[5:3]	rw	定时器 Tx 模式控制 （基本工作模式） 000 定时器模式 001 计数器模式 010 门控定时器模式，门控信号低电平有效 011 门控定时器模式，门控信号高电平有效 100 重载模式 101 捕获模式 110 增量接口模式（旋转检测） 111 增量接口模式（边沿检测）
Txl	[2:0]	rw	定时器 Tx 输入参数选择 不同工作模式的输入参数选择请分别参阅相关内容： 表 14-7 对应定时器模式和门控定时器模式 表 14-2 对应计数器模式 表 14-3 对应增量接口模式

1) 位 TxUD 和 TxUDE 的编码见**表 14-1**。

定时器 T2/T4 运行控制

可通过两种不同方式（软件）启动或终止辅助定时器 T2 和 T4：

- 由相关的定时器运行位（T2R 或 T4R）控制。此时要求相应的 $TxRC = 0$ 。
- 由核心定时器的运行位（T3R）控制。此时要求相应的远程控制位必须置位（ $TxRC = 1$ ）。

所选择的运行控制位对 T2/T4 的所有工作模式均有效。置位该位，将启动定时器工作；清零该位，将终止定时器工作。

门控定时器模式下，只有当所选择的运行位置位，且门控信号有效（为高或低电平，和设定的有效电平相同）时，定时器才能工作。

注：若选择远程控制辅助定时器，T3R 将同步启动或终止定时器 T3 和所选择的辅助定时器。

计数方向控制

控制 GPT1 模块各定时器（核心定时器和辅助定时器）计数方向的方式相同：由软件控制、或由外部输入引脚 TxEUD 控制。具体内容见表 14-1。

注：引脚 TxEUD 用作外部计数方向控制的输入时，必须配置为输入引脚（对应的方向控制位必须清零）。

14.1.4 GPT1 辅助定时器 T2/T4 的工作模式

除极少数例外情况，基本工作模式下辅助定时器和核心定时器的操作几乎完全相同。此外，T2/T4 可工作在一些组合工作模式下。

定时器 T2 和 T4 工作在定时器模式

将寄存器 TxCON 中的位域 TxM 设置为 000B，辅助定时器 Tx 工作在定时器模式。

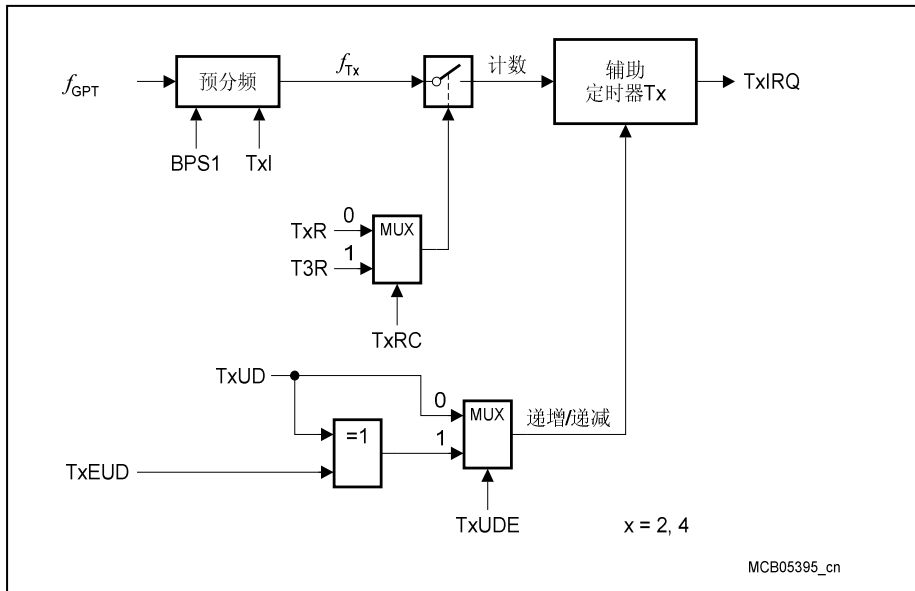


图 14-11 定时器模式下辅助定时器框图

定时器 T2 和 T4 工作在门控定时器模式

将寄存器 TxCON 中的位域 TxM 设置为 010_B 或 011_B，辅助定时器 Tx 工作在门控定时器模式。位 TxM.0 (TxCON.3) 选择门控输入的有效电平。

注：TxIN 上门控信号的跳变不会引发中断请求。

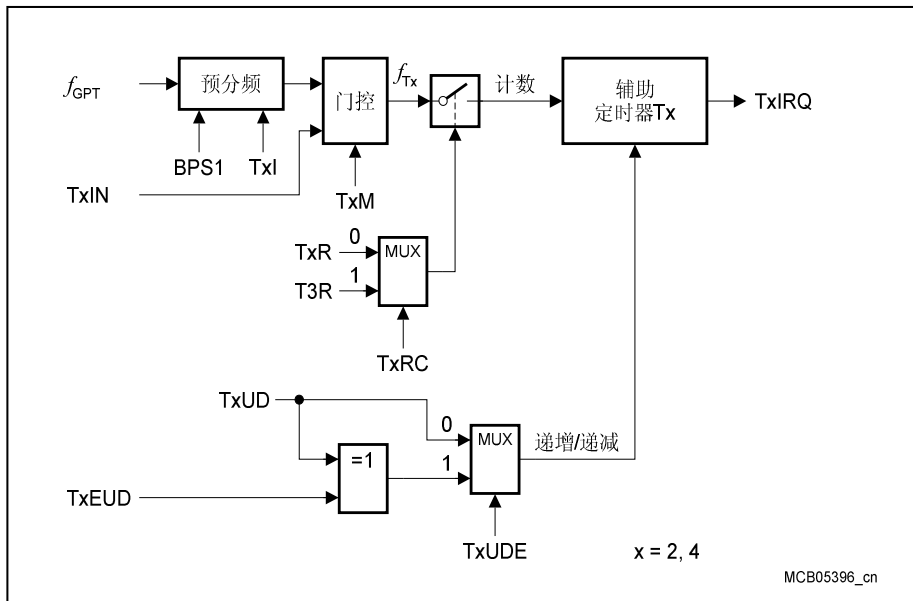


图 14-12 门控定时器模式下辅助定时器框图

注：T2 和 T4 输出不经翻转锁存。

可由 T2/T4 的运行位本地或 T3 的运行位远程控制辅助定时器 T2/T4 的启动/终止。

定时器 T2 和 T4 工作在计数器模式

将寄存器 TxCON 中的位域 TxM 设置为 001_B，辅助定时器 Tx 工作在计数器模式。计数器模式下，外部输入引脚 TxIN 跳变、或定时器 T3 翻转锁存 T3OTL 跳变时，可触发辅助定时器计数。TxIN 或 T3OTL 上的正跳变、负跳变或任意跳变可触发定时器递增或递减计数。控制寄存器 TxCON 中的位域 TxI 控制选择触发沿（见**表 14-5**）。

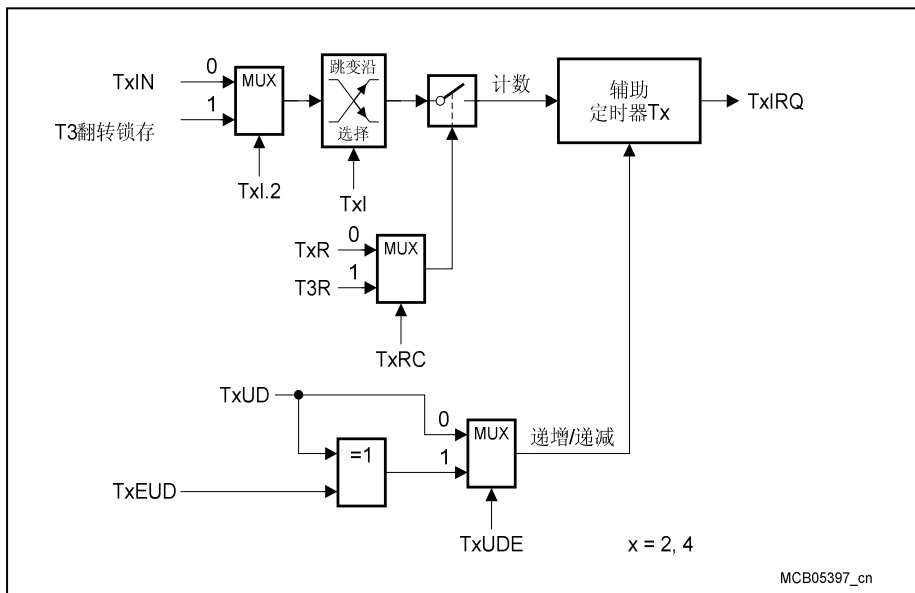


图 14-13 计数器模式下辅助定时器框图

表 14-5 GPT1 辅助定时器（计数器模式）输入沿选择

T2I/T4I	计数器递增/递减计数触发沿
X00	无，计数器 Tx 被禁止
001	TxIN 上的正跳变（上升沿）
010	TxIN 上的负跳变（下降沿）
011	TxIN 上的任意跳变（上升沿或下降沿）
101	T3 翻转锁存 T3OTL 上的正跳变（上升沿）

T2I/T4I	计数器递增/递减计数触发沿
110	T3 翻转锁存 T3OTL 上的负跳变（下降沿）
111	T3 翻转锁存 T3OTL 上的任意跳变（上升沿或下降沿）

注：只有 T3 上溢/下溢引发的 T3OTL 的状态跳变才能触发 T2/T4 计数；软件修改 T3OTL 的状态不会触发 T2/T4 计数。

计数器模式下，必须将引脚 **TxIN** 配置为输入（对应的方向控制位 **DPx.y** 必须为 0）。计数器模式下允许的最大输入频率取决于所选择的预分频因子。为了确保能够正确识别 **TxIN** 上计数输入信号的跳变，输入电平必须至少保持（高或低）规定数目的模块时钟周期之后才能改变。具体内容请参阅[章节 14.1.5](#)。

定时器 T2 和 T4 工作在增量接口模式

将寄存器 TxCON 中的位域 TxM 设置为 110_B 或 111_B，辅助定时器 Tx 工作在增量接口模式。增量接口模式下，与辅助定时器 Tx 相关的两个输入信号（TxIN，TxEUD）用作增量编码器的接口。这两个外部输入引脚上的一个或两个信号的跳变都触发 Tx 计数，从而提供了 2 倍或 4 倍于编码器输入的计数精度。

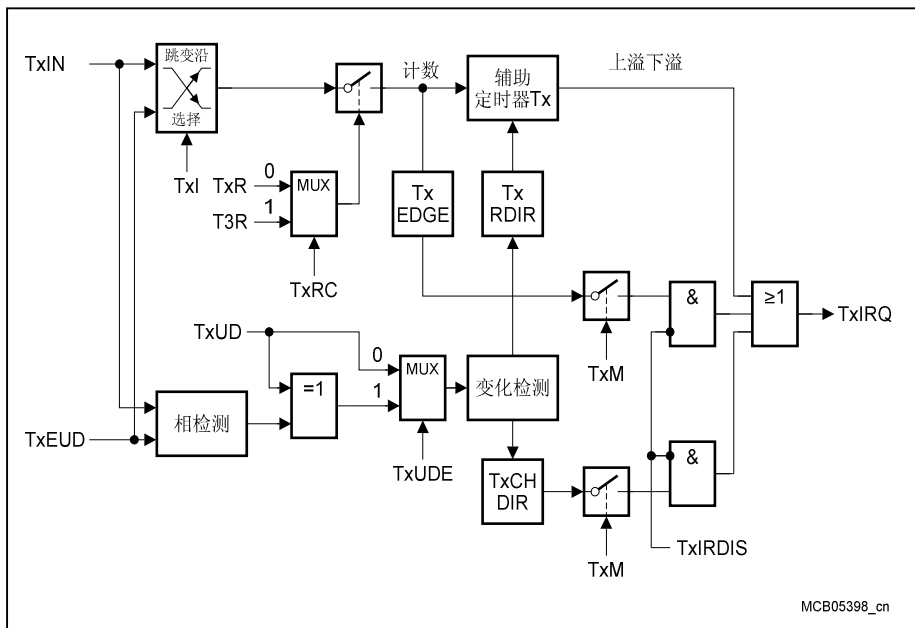


图 14-14 增量接口模式下辅助定时器 Tx 的框图

增量模式下定时器 T2 和 T4 的操作及中断产生和定时器 T3 相同。其运行描述、图和表格也适用。

定时器级联

溢出翻转锁存位 T3OTL 用作计数器模式下辅助定时器的计数时钟源时，核心定时器 T3 和辅助定时器级联工作。定时器级联可构成 32 位或 33 位定时器/计数器，这取决于选择何种 T3OTL 跳变触发辅助定时器计数。

- **32 位定时器/计数器：**若选择 T3OTL 的任意跳变触发辅助定时器计数，核心定时器 T3 在每次上溢/下溢时，辅助定时器计数。因此 T2/T4 和 T3 构成 32 位定时器。
- **33 位定时器/计数器：**若仅选择 T3OTL 的正跳变或负跳变触发辅助定时器计数，核心定时器 T3 在每两次上溢/下溢后，辅助定时器计数一次。因此它们构成 33 位定时器（16 位核心定时器 + T3OTL + 16 位辅助定时器）。

只要 T3OTL 不被软件修改，它即表示内部翻转锁存器的状态，可被视为 33 位定时器的组成部分。

两个级联定时器的计数方向可以不同，从而为用户提供了多种不同的配置组合。

定时器 T3 代表级联定时器的低 16 位，在这种情况下，T3 可工作在定时器模式、门控定时器模式或计数器模式。

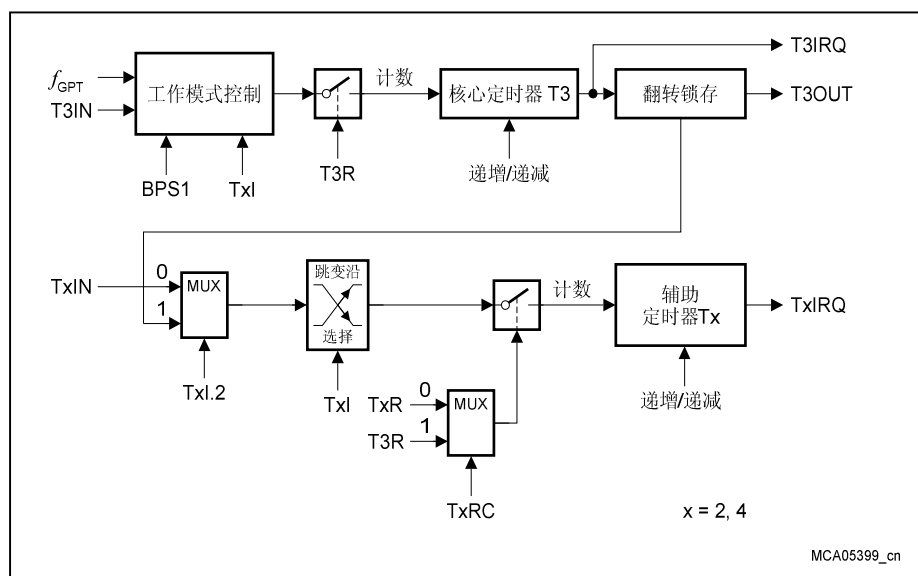


图 14-15 核心定时器 T3 和辅助定时器级联

辅助定时器工作在重载模式

将寄存器 TxCON 中的位域 TxM 设置为 100B，辅助定时器 Tx 工作在重载模式。重载模式下，两个信号中的任意一个可触发重载，将辅助定时器寄存器的内容重新装入核心定时器 T3 中。触发信号的选择和计数模式下计数时钟源的选择方式相同（见表 14-5），即辅助定时器的输入 TxIN 跳变时、或翻转锁存 T3OTL 跳变时可触发重载。

注：辅助定时器（T2 或 T4）工作在重载模式下时，定时器的停止不受自身运行控制位 T2R 或 T4R 控制。

若定时器的输入引脚 TxIN 用于触发重载，必须将其配置为输入。

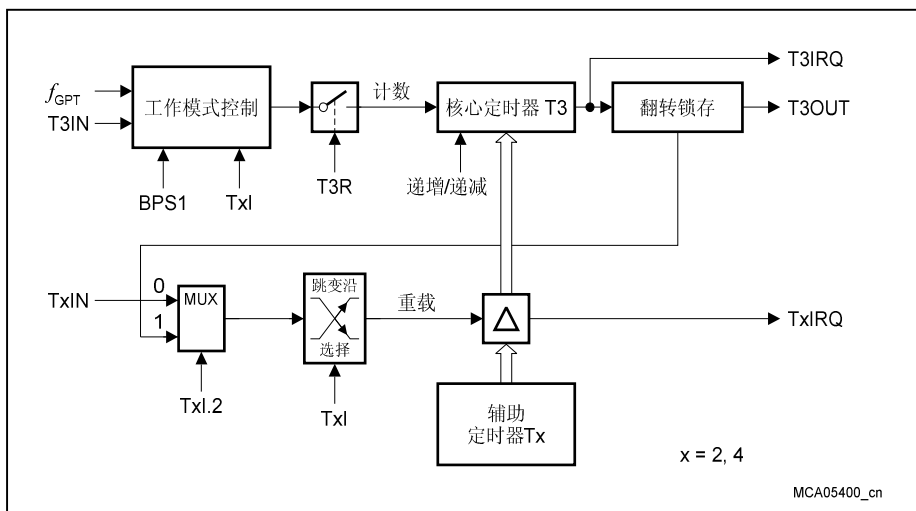


图 14-16 重载模式下 GPT1 的辅助定时器

出现触发信号时，辅助定时器寄存器（T2 或 T4）的内容装入 T3，相应的中断请求标志（T2IR 或 T4IR）被置位。

注：若选择 T3OTL 的跳变作为触发信号，产生触发时，中断请求标志 T3IR 也将被置位，指示 T3 上溢或下溢。软件修改 T3OTL 不会触发 T2/T4 的计数功能。

为了确保能够正确识别 TxIN 上重载输入信号的跳变，输入电平必须至少保持（高或低）规定数目的模块时钟周期之后才能改变。具体内容请参阅章节 14.1.5。

由 T3OTL 触发的重载工作模式可有多种不同的配置。选择不同的有效跳变沿，将执行以下功能：

- 若选择 T3OTL 的任意跳变触发重载，T3 在每次上溢或下溢时，辅助定时器的值重新装入核心定时器。此配置为标准重载模式（上溢/下溢时发生重载）。
- 若选择 T3OTL 的正跳变或负跳变触发重载，T3 在每两次上溢或下溢后，辅助定时器的值重新装入核心定时器。
- 使用“单沿跳变”重载模式可实现非常灵活的脉宽调制（PWM）。即设定一个辅助定时器在 T3OTL 正跳变时重载核心定时器；另一个辅助定时器在 T3OTL 负跳变时重载核心定时器。通过这种组合，核心定时器可轮流从两个辅助定时器中获得重载值。

图 14-17 举例说明如何利用“单沿跳变”重载机制产生 PWM 信号。T2 定义了 PWM 信号的接通时间（正跳变触发重载），T4 定义了 PWM 信号的关闭时间（负跳变触发重载）。若 T3OE = 1，PWM 信号可从引脚 T3OUT 输出。利用该方法，PWM 信号的接通/关闭时间可在较大范围内变化。

注：如有需要，可通过软件改变输出翻转锁存 T3OTL 的值，从而修改 PWM 信号。

但该操作不会触发 T3 重载。

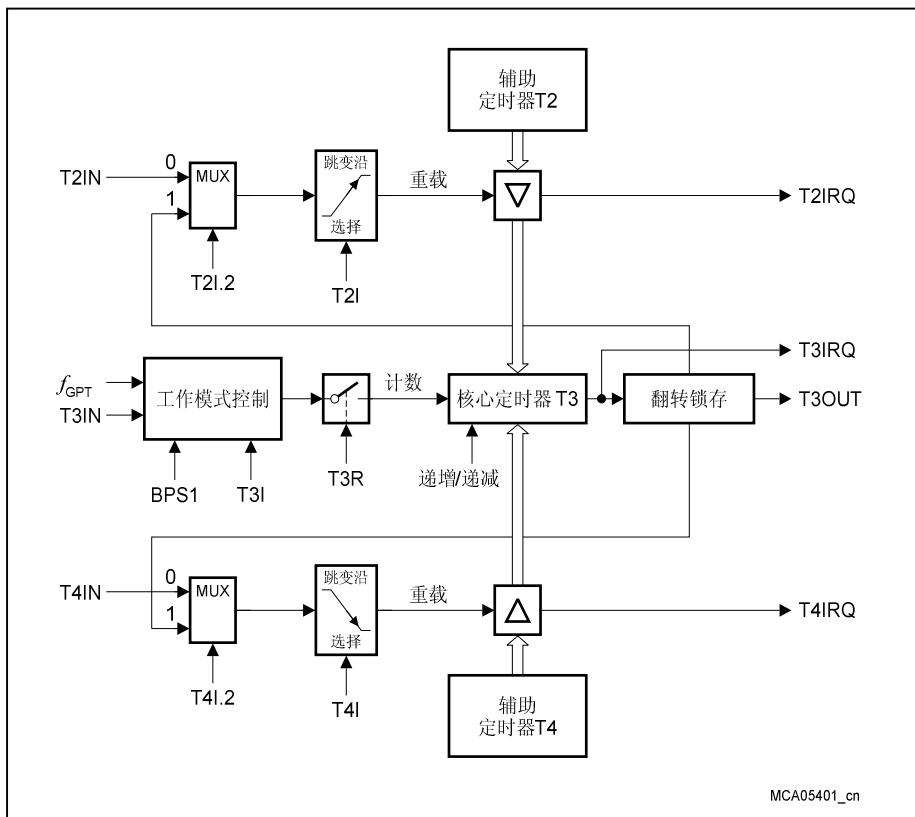


图 14-17 用于产生 PWM 的 GPT1 定时器重载配置

注：应该尽量避免为两个辅助定时器选择相同的重载触发事件（尽管该情况可能发生）。发生该情况时，两个重载寄存器会试图同时加载核心定时器。此时丢弃 T2 的值；将 T4 的值载入 T3。

辅助定时器工作在捕获模式

将寄存器 **TxCON** 中的位域 **TxM** 设置为 **101_B**，辅助定时器 **Tx** 工作在捕获模式。捕获模式下，辅助定时器的外部输入引脚 **TxIN** 发生跳变时，核心定时器 **T3** 的值被锁存到辅助定时器寄存器。捕获触发信号可以是 **TxIN** 的正跳变、负跳变或任意跳变。

位域 **TxI** 中的低两位用于选择有效跳变沿（见**表 14-5**）。**TxI** 中的最高位（位 2）与捕获模式无关，必须被清零（**TxI.2 = 0**）。

*注：辅助定时器（**T2** 或 **T4**）工作在捕获模式时，定时器的停止不受自身运行控制位 **T2R** 或 **T4R** 控制。*

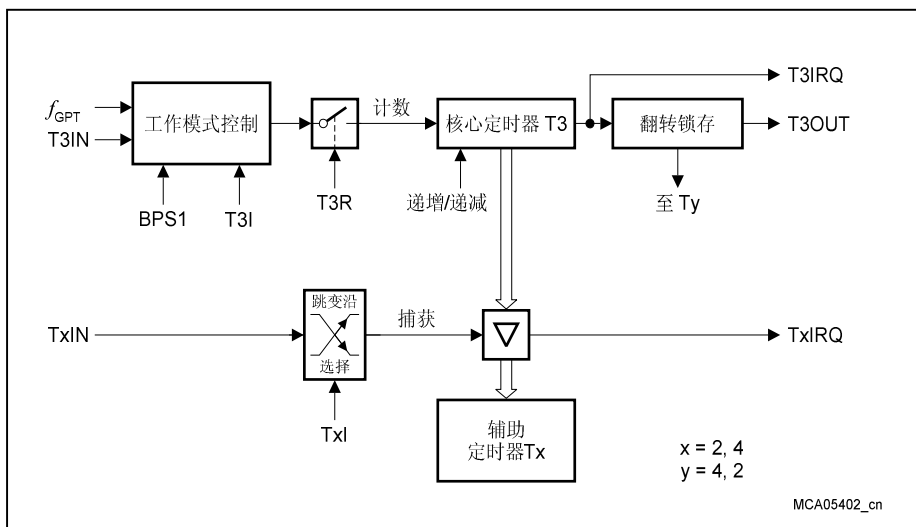


图 14-18 捕获模式下 GPT1 辅助定时器的框图

输入引脚 **TxIN** 上产生触发信号（所选择的跳变）时，核心定时器的内容被锁存到辅助定时器寄存器中，相关中断请求标志 **TxIR** 被置位。

捕获模式下，必须将输入引脚 **TxIN** 配置为输入。为了确保能够正确识别 **TxIN** 上捕获输入信号的跳变，输入电平必须至少保持（高或低）规定数目的模块时钟周期之后才能改变。具体内容请参阅**章节 14.1.5**。

14.1.5 GPT1 时钟信号控制

定时器模块 GPT1 的所有操作均由其基本时钟触发。系统时钟经过基本模块预分频产生该基本时钟，预分频因子由寄存器 T3CON 中的位域 BPS1 控制（见图 14-2）。有两种不同的方式产生计数时钟：

- **内部计数时钟**，经过预分频处理（预分频因子可编程设定）的 GPT1 基本时钟，该时钟用于（门控）定时器模式。
- **外部计数时钟**，取自定时器的输入引脚，该时钟用于计数器模式。

GPT1 模块的基本时钟决定最大计数频率和定时器精度。

表 14-6 模块 GPT1 的基本时钟选择

模块预分频 ¹⁾	BPS1 = 01 _B	BPS1 = 00 _B ²⁾	BPS1 = 11 _B	BPS1 = 10 _B
GPT1 的预分频因子: F(BPS1)	F(BPS1) = 4	F(BPS1) = 8	F(BPS1) = 16	F(BPS1) = 32
最大外部计数频率	f _{GPT} /8	f _{GPT} /16	f _{GPT} /32	f _{GPT} /64
输入信号持续稳定的最短时间	4 × t _{GPT}	8 × t _{GPT}	16 × t _{GPT}	32 × t _{GPT}

1) 请注意位域 BPS1 为非线性编码。

2) 复位后的缺省值。

内部计数时钟产生

定时器模式和门控定时器模式下，GPT1 的每个定时器的计数时钟来自经过预分频处理的 GPT1 基本时钟，该预分频因子可由定时器控制寄存器 TxCON 中的位域 TxI 编程设定。

定时器 Tx 的计数频率 f_{Tx} 和相应计数精度 r_{Tx} 随较低的时钟频率线性变化，计算公式如下：

$$f_{Tx} = \frac{f_{GPT}}{F(BPS1) \times 2^{\langle TxI \rangle}} \quad r_{Tx} [\mu s] = \frac{F(BPS1) \times 2^{\langle TxI \rangle}}{f_{GPT} [MHz]} \quad (14.1)$$

计数频率不仅取决于公共的模块时钟预分频因子 F(BPS1)；还取决于定时器各自的预分频因子 2^{⟨TxI⟩}。GPT1 定时器的总预分频因子总结见表 14-7。

根据总预分频因子和系统频率计算得到定时器的各项参数（如计数时钟频率、计数时钟精度和计数周期），见表 14-8。请注意有些结果已四舍五入。

表 14-7 GPT1 内部计数时钟的总预分频因子

定时器 Tx 的 预分频因子	公共预分频因子 ¹⁾			
	BPS1 = 01_B	BPS1 = 00_B	BPS1 = 11_B	BPS1 = 10_B
Txl = 000_B	4	8	16	32
Txl = 001_B	8	16	32	64
Txl = 010_B	16	32	64	128
Txl = 011_B	32	64	128	256
Txl = 100_B	64	128	256	512
Txl = 101_B	128	256	512	1024
Txl = 110_B	256	512	1024	2048
Txl = 111_B	512	1024	2048	4096

1) 请注意位域 BPS1 为非线性编码。

表 14-8 GPT1 定时器参数

系统时钟 = 10 MHz			总分 频因 子	系统时钟 = 40 MHz		
频率	精度	周期		频率	精度	周期
2.5 MHz	400 ns	26.21 ms	4	10.0 MHz	100 ns	6.55 ms
1.25 MHz	800 ns	52.43 ms	8	5.0 MHz	200 ns	13.11 ms
625.0 kHz	1.6 μs	104.9 ms	16	2.5 MHz	400 ns	26.21 ms
312.5 kHz	3.2 μs	209.7 ms	32	1.25 MHz	800 ns	52.43 ms
156.25 kHz	6.4 μs	419.4 ms	64	625.0 kHz	1.6 μs	104.9 ms
78.125 kHz	12.8 μs	838.9 ms	128	312.5 kHz	3.2 μs	209.7 ms
39.06 kHz	25.6 μs	1.678 s	256	156.25 kHz	6.4 μs	419.4 ms
19.53 kHz	51.2 μs	3.355 s	512	78.125 kHz	12.8 μs	838.9 ms
9.77 kHz	102.4 μs	6.711 s	1024	39.06 kHz	25.6 μs	1.678 s

系统时钟 = 10 MHz			总分频因子	系统时钟 = 40 MHz		
频率	精度	周期		频率	精度	周期
4.88 kHz	204.8 μ s	13.42 s	2048	19.53 kHz	51.2 μ s	3.355 s
2.44 kHz	409.6 μ s	26.84 s	4096	9.77 kHz	102.4 μ s	6.711 s

外部计数时钟输入

用 GPT1 的基本时钟对外部输入信号采样（见图 14-2）。为了确保信号能被正确识别，外部信号的当前输入电平值（高或低）必须至少保持一个完整的采样周期之后才能改变。若输入信号连续两次采样的电平值不同，即可识别到信号发生了跳变。因此，至少需要两个基本时钟周期对外部输入信号采样，故输入信号的最大频率一定不能高于基本时钟频率的一半。

GPT1 外部输入信号的参数限制总结见表 14-9。

表 14-9 GPT1 外部输入信号的参数限制

系统时钟 = 10 MHz		外部输入频率因子	GPT1 分频因子 BPS1	输入信号持续时间	系统时钟 = 40 MHz	
最大输入频率	电平保持最短时间				最大输入频率	电平保持最短时间
1.25 MHz	400 ns	$f_{\text{GPT}}/8$	01_B	$4 \times t_{\text{GPT}}$	5.0 MHz	100 ns
625.0 kHz	800 ns	$f_{\text{GPT}}/16$	00_B	$8 \times t_{\text{GPT}}$	2.5 MHz	200 ns
312.5 kHz	1.6 μ s	$f_{\text{GPT}}/32$	11_B	$16 \times t_{\text{GPT}}$	1.25 MHz	400 ns
156.25 kHz	3.2 μ s	$f_{\text{GPT}}/64$	10_B	$32 \times t_{\text{GPT}}$	625.0 kHz	800 ns

上表中的各项参数限制对 GPT1 的所有外部输入信号均有效，包括计数器模式下和增量接口模式下的外部计数信号、门控计数器模式下的门控输入信号以及外部方向信号。

14.1.6 GPT1 定时器寄存器

GPT12E_Tx

定时器 Tx 计数寄存器 SFR (FE4x_H/2y_H) 复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Txvalue															
rwh															

表 14-10 GPT1 定时器寄存器地址

定时器寄存器	物理地址	8 位地址
T3	FE42 _H	21 _H
T2	FE40 _H	20 _H
T4	FE44 _H	22 _H

14.1.7 GPT1 定时器的中断控制

当定时器（递增计数）从 FFFF_H 至 0000_H 发生上溢，或定时器（递减计数）从 0000_H 至 FFFF_H 发生下溢时，寄存器 TxIC 中的中断请求标志（T2IR, T3IR 或 T4IR）被置位。若相应中断使能位（寄存器 TxIC 中的 T2IE, T3IE 或 T4IE）被置位，将会产生指向定时器中断向量（T2INT, T3INT 或 T4INT）的中断，或者触发 PEC 操作。每个定时器对应一个中断控制寄存器。

GPT12E_T2IC

定时器 T2 中断控制寄存器 SFR (FF60_H/B0_H) 复位值: - - 00_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	GPX	T2IR	T2IE	ILVL			GLVL		
-	-	-	-	-	-	-	rw	rwh	rw		rw			rw	

GPT12E_T3IC

定时器 T3 中断控制寄存器 SFR (FF62_H/B1_H) 复位值: - - 00_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	GPX	T3IR	T3IE	ILVL			GLVL		
-	-	-	-	-	-	-	rw	rwh	rw		rw			rw	

GPT12E_T4IC

定时器 T4 中断控制寄存器 SFR (FF64_H/B2_H) 复位值: - - 00_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	GPX	T4IR	T4IE	ILVL			GLVL		
-	-	-	-	-	-	-	rw	rwh	rw		rw			rw	

注：控制位域的具体解释请参阅中断控制寄存器的描述。

14.2 定时器模块 GPT2

从编程人员的角度来看，GPT2 由一组特殊功能寄存器（SFR）组成（总结见下图）。端口和方向寄存器中用作 GPT2 功能的部分用阴影标出。

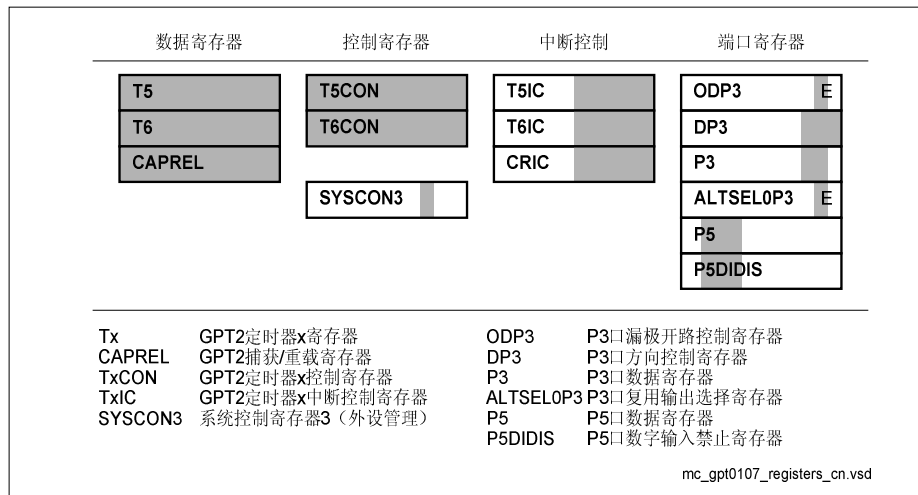


图 14-19 定时器模块 GPT2 的相关特殊功能寄存器（SFR）

GPT2 模块中的两个定时器（T5，T6）可工作在 3 种基本模式：定时器模式、门控定时器模式或计数器模式。所有定时器均可递增或递减计数。每个定时器分别由各自的控制寄存器 TxCON 控制。

每个定时器有一个对应的输入引脚 TxIN（复用引脚功能），用作门控定时器模式中的门控信号，或计数器模式中的计数输入。计数方向（递增/递减）可由软件设定，或由外部递增/递减控制输入引脚 TXEUD（复用引脚功能）上的信号动态改变。输出翻转锁存 T6OTL 指示核心定时器 T6 发生上溢/下溢，T6OTL 的状态可从引脚 T6OUT（复用引脚功能）输出。辅助定时器 T5 可以（通过 T6OTL）和核心定时器 T6 级联工作。

捕获/重载寄存器 CAPREL 可用于捕获定时器 T5 的内容，或者重载定时器 T6。系统支持一种特殊的模式，即通过寄存器 CAPREL 同时实现捕获、重载两种功能，采用该模式可实现倍频。输入引脚 CAPIN，或 GPT1 的定时器 T3 的输入 T3IN 和 T3EUD 触发捕获；定时器 T6 上溢或下溢触发重载。定时器 T6 的上溢/下溢信号还可作为 CAPCOM2 定时器的输入时钟。

定时器计数寄存器 T5 和 T6 位于不可位寻址的 SFR 区，CPU 访问这些寄存器，可读取或修改每个定时器的当前值（请参阅[章节 14.2.7](#)）。若定时器执行递增、递减、重

载或捕获操作时，CPU 恰好正在对（任意）某个定时器寄存器写入，则 CPU 写操作占优以确保结果正确。

由中断控制寄存器 TxIC 控制 GPT2 的中断产生，这些寄存器不在 GPT2 模块中。GPT2 的输入和输出线与 P3 和 P5 口的引脚相连。端口功能的控制寄存器位于端口模块中。

注：外部输入信号的时序要求请参阅[章节 14.2.6](#)，包括引脚在内的模块接口信号归纳见[章节 14.3](#)。

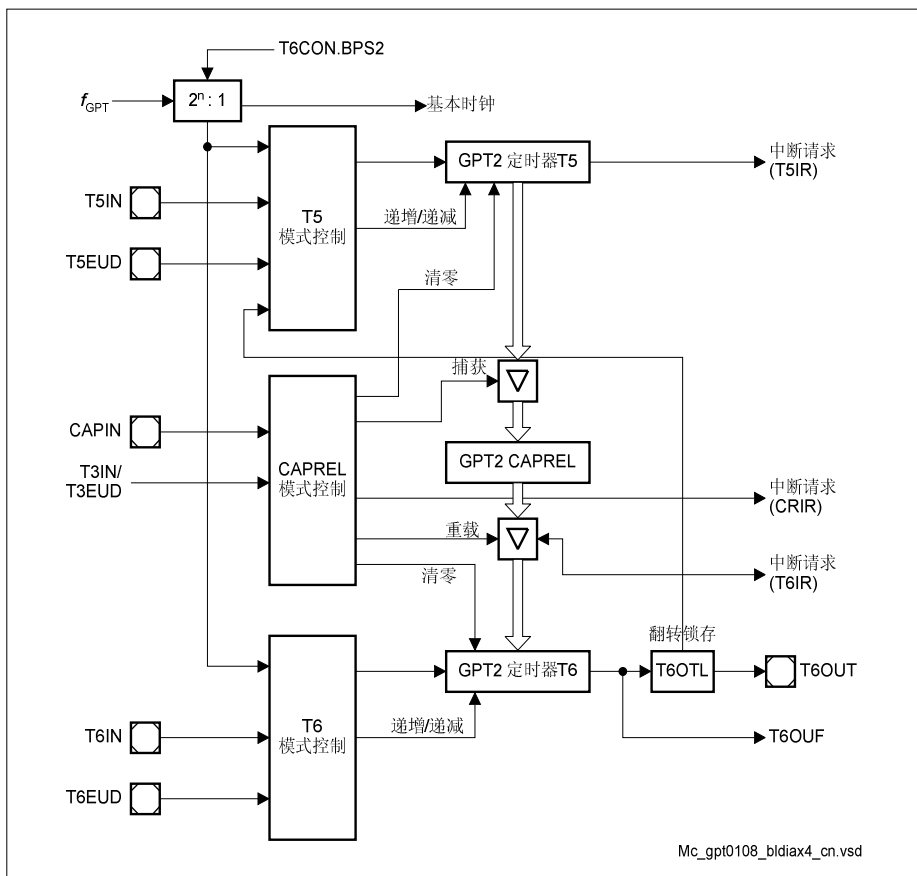


图 14-20 GPT2 框图

14.2.1 GPT2 核心定时器 T6 的控制

可从计数寄存器 T6 中读取核心定时器 T6 的当前值，该寄存器也可由 CPU 写入，例如由 CPU 设置定时器的初始值。

由可位寻址的控制寄存器 T6CON 配置和控制核心定时器 T6。

GPT12E_T6CON

定时器 T6 控制寄存器

SFR (FF48H/A4H)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T6 SR	T6 CLR	-	BPS2	T6 OTL	T6 OE	T6 UDE	T6 UD	T6R	T6M			T6I			
rw	rw	-	rw	rwh	rw	rw	rw	rw	rw			rw			

符号	位序号	读写类型	功能描述
T6SR	15	rw	定时器 T6 重载模式使能 0 由寄存器 CAPREL 重载 T6 被禁止 1 由寄存器 CAPREL 重载 T6 被使能
T6CLR	14	rwh	定时器 T6 清零使能位 0 捕获事件不触发定时器 T6 清零 1 捕获事件触发定时器 T6 清零
BPS2	[12:11]	rw	模块 GPT2 预分频控制 选择模块 GPT2 的基本时钟 （请参阅 章节 14.2.6 ） 00 $f_{GPT}/4$ 01 $f_{GPT}/2$ 10 $f_{GPT}/16$ 11 $f_{GPT}/8$
T6OTL	10	rwh	定时器 T6 溢出翻转锁存 每次 T6 上溢/下溢时翻转。可由软件置位或复位（具体描述请参阅相关章节）。
T6OE	9	rw	上溢/下溢输出使能 0 复用输出功能被禁止 1 T6 翻转锁存器的状态输出至引脚 T6OUT

符号	位序号	读写类型	功能描述
T6UDE	8	rw	定时器 T6 外部递增/递减计数使能¹⁾ 0 输入 T6EUD 被断开 1 计数方向受输入 T6EUD 的控制
T6UD	7	rw	定时器 T6 递增/递减计数控制¹⁾ 0 定时器 T6 递增计数 1 定时器 T6 递减计数
T6R	6	rw	定时器 T6 运行控制位 0 定时器 T6 停止 1 定时器 T6 运行
T6M	[5:3]	rw	定时器 T6 模式控制（基本工作模式） 000 定时器模式 001 计数器模式 010 门控定时器模式，门控信号低电平有效 011 门控定时器模式，门控信号高电平有效 100 保留，不使用该组合 101 保留，不使用该组合 110 保留，不使用该组合 111 保留，不使用该组合
T6I	[2:0]	rw	定时器 T6 输入参数选择 不同工作模式的输入参数选择请分别参阅相关内容： 表 14-16 对应定时器模式和门控定时器模式 表 14-12 对应计数器模式

1) 位 T6UD 和 T6UDE 的编码见表 14-11。

定时器 T6 运行控制

可通过软件设定 T6R（定时器 T6 运行控制位），启动或终止核心定时器 T6 工作。该控制位对 T6 的所有工作模式均有效。置位 T6R 将启动定时器；清零 T6R 将终止定时器。

在门控定时器模式下，只有当 T6R = 1 且门控信号有效（设置为高电平或低电平有效）时，定时器才能工作。

注：若定时器控制寄存器 T5CON 中的位 T5RC 被置位，位 T6R 也将控制（启动或终止）辅助定时器 T5 的运行。

计数方向控制

GPT2 定时器（核心定时器和辅助定时器）的计数方向可由控制寄存器 TxCON 中的位 TxUD 和 TxUDE 控制，选择由软件、或外部输入引脚 TxEUD（定时器 Tx 外部递增/递减控制输入）控制递增/递减计数。由软件控制计数方向时（TxUDE = 0），置位或清零 TxUD 可改变计数方向；位 TxUDE = 1 时，由引脚 TxEUD 控制计数方向。但是，仍可用 TxUD 来翻转实际的计数方向，如表 14-11 所示。无论定时器是否工作，计数方向均可被改变。

表 14-11 GPT2 定时器计数方向控制

引脚 TxEUD	位 TxUDE	位 TxUD	计数方向
X	0	0	递增计数
X	0	1	递减计数
0	1	0	递增计数
1	1	0	递减计数
0	1	1	递减计数
1	1	1	递增计数

14.2.2 GPT2 核心定时器 T6 的工作模式

定时器 T6 工作在定时器模式

将寄存器 T6CON 中的位域 T6M 设置为 000_B，核心定时器 T6 工作在定时器模式。定时器模式下，模块输入时钟 f_{GPT} 经两个可编程的预分频因子（由寄存器 T6CON 中的位域 BPS2 和 T6I 控制）分频后，用作 T6 的输入时钟信号。输入时钟选择的具体内容请参阅[章节 14.2.6](#)。

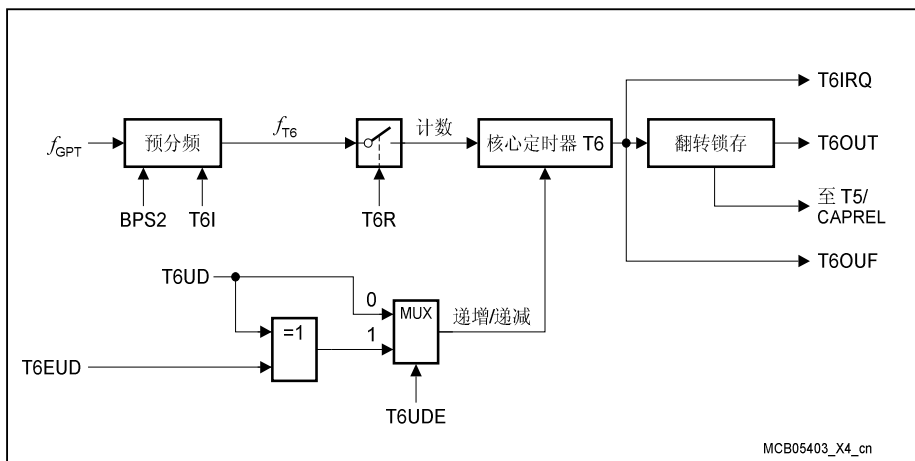


图 14-22 定时器模式下核心定时器 T6 框图

门控定时器模式

将寄存器 T6CON 中的位域 T6M 设置为 010_B 或 011_B，核心定时器 T6 工作在门控定时器模式。位 T6M.0 (T6CON.3) 选择门控输入的有效电平。门控定时器模式下输入时钟的频率选择和定时器模式下相同（请参阅[章节 14.2.6](#)）。但该模式下定时器的输入时钟受外部输入引脚 T6IN（定时器 T6 的外部输入）门控控制。

T6 工作在该模式时，引脚 T6IN 必须被配置为输入（对应的方向控制位必须为 0）。

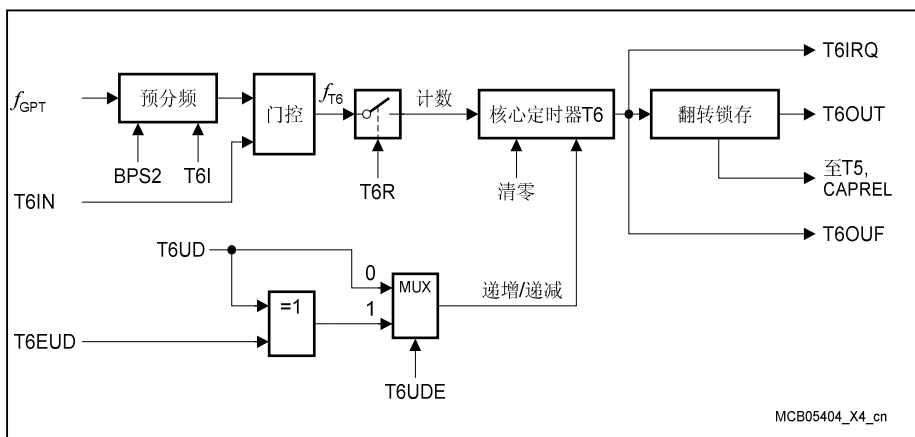


图 14-23 门控定时器模式下核心定时器 T6 框图

若 T6M = 010_B，引脚 T6IN 为低电平时定时器被使能；为高电平时定时器被终止。若 T6M = 011_B，T6IN 必须保持高电平才能使能定时器。此外，通过软件修改 T6R 可开启或关闭定时器。只有当 T6R 为 1 且门控信号有效时定时器才能工作。若 T6R 为 0 或者门控信号无效，定时器将停止工作。

注：引脚 T6IN 上门控信号的跳变不会引发中断请求。

计数器模式

将寄存器 **T6CON** 中的位域 **T6M** 设置为 **001_B**，核心定时器 **T6** 工作在计数器模式。计数器模式下，外部输入引脚 **T6IN** 上的跳变用作定时器 **T6** 的输入时钟。**T6IN** 上的正跳变、负跳变或任意跳变均可触发定时器递增或递减计数。控制寄存器 **T6CON** 中的位域 **T6I** 控制选择触发边沿（见**表 14-12**）。

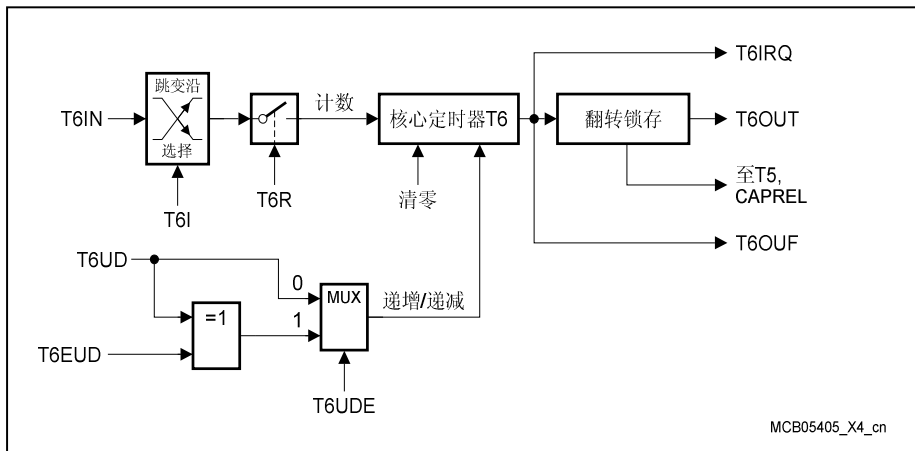


图 14-24 计数器模式下核心定时器 T6 框图

表 14-12 GPT2 核心定时器 T6（计数器模式）输入沿选择

T6I	计数器递增/递减计数触发沿
000	无，计数器 T6 被禁止
001	T6IN 上的正跳变（上升沿）
010	T6IN 上的负跳变（下降沿）
011	T6IN 上的任意跳变（上升沿或下降沿）
1XX	保留，不使用该组合

计数器模式下，必须将引脚 **T6IN** 配置为输入（对应的方向控制位 **DPx.y** 必须为 **0**）。计数器模式下允许的最大输入频率取决于所选择的预分频因子。为了确保能够正确识别 **T6IN** 上计数输入信号的跳变，输入电平必须至少保持（高或低）规定数目的模块时钟周期之后才能改变。具体内容请参阅**章节 14.2.6**。

14.2.3 GPT2 辅助定时器 T5 的控制

辅助定时器 T5 可被配置为定时器模式、门控定时器模式或计数器模式，定时器输入时钟的频率选择以及计数输入信号的选择和核心定时器 T6 相同。除了这三种计数模式，辅助定时器还可以和核心定时器级联，外部或内部事件可触发将 T5 的内容捕获到寄存器 CAPREL 中。辅助定时器的启动/终止可由 T6 的运行位远程控制，从而可同步控制多个定时器的运行。

辅助定时器的当前值可从计数寄存器 T5 中读取。该寄存器也可由 CPU 写入，例如设置定时器的初始值。

通过可位寻址的控制寄存器 T5CON 配置定时器 T5。该寄存器中的某些位元/域也控制 CAPREL 寄存器的功能。请注意：模块 GPT2 中两个定时器均具备的功能由各自的控制寄存器中相同位置的位元/域、以相同的方式进行控制。

注：辅助定时器无输出翻转锁存逻辑，无复用输出功能。

GPT12E_T5CON

定时器 T5 控制寄存器

SFR (FF46H/A3H)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T5 SC	T5 CLR	CI	T5 CC	CT3	T5 RC	T5 UDE	T5 UD	T5R	T5M			T5I			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			rw		

符号	位序号	读写类型	功能描述
T5SC	15	rw	定时器 T5 捕获模式使能 0 捕获到寄存器 CAPREL 被禁止 1 捕获到寄存器 CAPREL 被使能
T5CLR	14	rw	定时器 T5 清零使能位 0 捕获事件不触发定时器 T5 清零 1 捕获事件触发定时器 T5 清零
CI	[13:12]	rw	寄存器 CAPREL 捕获触发事件选择 （与位 CT3 有关） 00 捕获被禁止 01 CAPIN 上的正跳变（上升沿）或 T3IN 上的任意跳变 10 CAPIN 上的负跳变（下降沿）或 T3EUD 上的任意跳变

符号	位序号	读写类型	功能描述
			11 CAPIN 上的任意跳变（上升沿或下降沿）或 T3IN/T3EUD 上的任意跳变
T5CC	11	rw	定时器 T5 捕获校正 0 直接捕获 T5 的值（无校正） 1 T5 的值减 1 之后被捕获
CT3	10	rw	定时器 T3 捕获触发使能 0 输入线 CAPIN 触发捕获 1 T3 的输入线 T3IN 和/或 T3EUD 触发捕获
T5RC	9	rw	定时器 T5 远程控制位 0 定时器 T5 由自身的运行位 T5R 控制 1 定时器 T5 由 T6 的运行位 T6R 控制，不受 T5R 控制
T5UDE	8	rw	定时器 T5 外部递增/递减计数使能¹⁾ 0 输入 T5EUD 被断开 1 计数方向受输入 T5EUD 的控制
T5UD	7	rw	定时器 T5 递增/递减计数控制¹⁾ 0 定时器 T5 递增计数 1 定时器 T5 递减计数
T5R	6	rw	定时器 T5 运行控制位 0 定时器 T5 停止 1 定时器 T5 运行 <i>注：只有 T5RC = 0 时，该位才可控制定时器 T5。</i>
T5M	[5:3]	rw	定时器 T5 模式控制（基本工作模式） 000 定时器模式 001 计数器模式 010 门控定时器模式，门控信号低电平有效 011 门控定时器模式，门控信号高电平有效 1XX 保留，不使用该组合

符号	位序号	读写类型	功能描述
T5I	[2:0]	rw	定时器 T5 输入参数选择 不同工作模式的输入参数选择请分别参阅相关内容： 表 14-16 对应定时器模式和门控定时器模式 表 14-12 对应计数器模式

1) 位 T5UD 和 T5UDE 的编码见 **表 14-11**。

定时器 T5 运行控制

可通过两种不同的方式（软件设定）启动或终止辅助定时器 T5：

- 由相关的定时器运行位（T5R）控制。此时要求 $T5RC = 0$ 。
- 由核心定时器的控制位（T6R）控制。此时要求远程控制位必须被置位（ $T5RC = 1$ ）。

所选择的运行控制位对 T5 的所有工作模式均有效。该位被置位将启动定时器；该位被清零将终止定时器。

门控定时器模式下，只有当所选择的运行位被置位且门控信号有效（设置为高或低电平有效）时，定时器才能工作。

注：若选择远程控制 T5 运行，T6R 将同步启动或终止定时器 T6 和辅助定时器 T5。

14.2.4 GPT2 辅助定时器 T5 的工作模式

除极少数例外情况，辅助定时器在基本工作模式下的操作和核心定时器几乎完全相同。此外，T5 可工作在一些组合作模式式下。

定时器 T5 工作在定时器模式

将寄存器 T5CON 中的位域 T5M 设置为 000_B，辅助定时器 T5 工作在定时器模式。

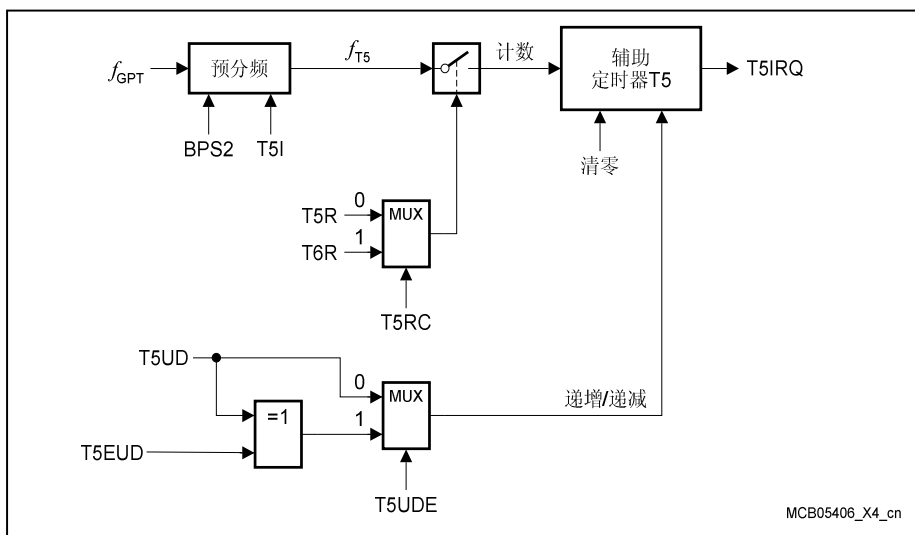


图 14-25 定时器模式下辅助定时器 T5 框图

定时器 T5 工作在门控定时器模式

将寄存器 T5CON 中的位域 T5M 设置为 010_B 或 011_B，辅助定时器 T5 工作在门控定时器模式。位 T5M.0 (T5CON.3) 选择门控输入的有效电平。

注：T5IN 上门控信号的跳变不会引发中断请求。

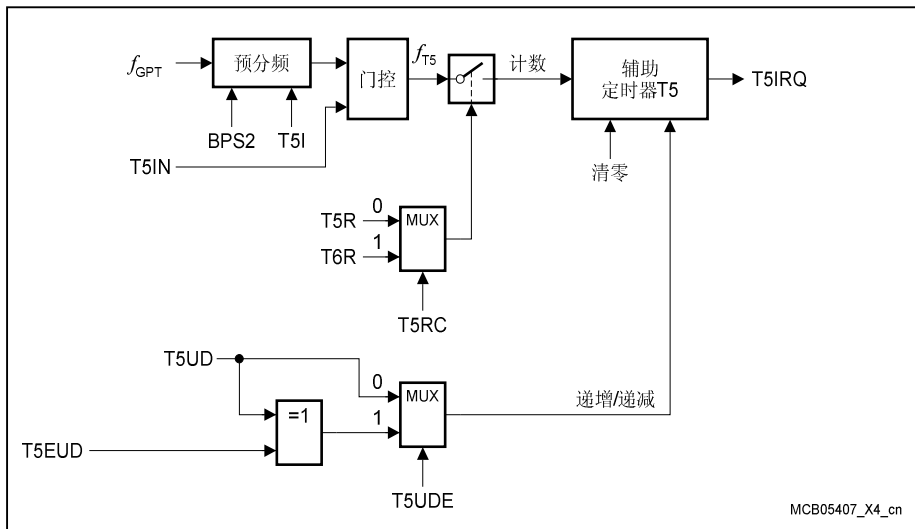


图 14-26 门控定时器模式下辅助定时器 T5 框图

注：T5 无输出翻转锁存逻辑。

可通过自身或 T6 的运行位控制辅助定时器的启动/终止。

定时器 T5 工作在计数器模式

将寄存器 T5CON 中的位域 T5M 设置为 001_B，辅助定时器 T5 工作在计数器模式。计数器模式下，外部输入引脚 T5IN 跳变、或定时器 T6 的溢出翻转锁存 T6OTL 跳变时，可触发辅助定时器的计数。T5IN 或者 T6OTL 上的正跳变、负跳变或任意跳变可触发定时器递增或递减计数。控制寄存器 T5CON 中的位域 T5I 控制选择触发沿（见表 14-13）。

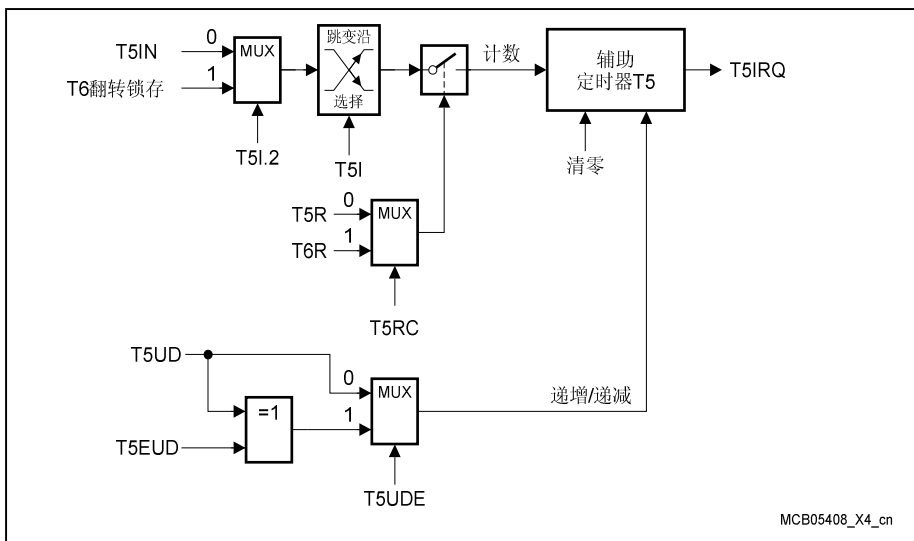


图 14-27 计数器模式下辅助定时器 T5 框图

表 14-13 GPT2 辅助定时器（计数器模式）输入沿选择

T5I	计数器递增/递减计数触发沿
X00	无，计数器 T5 被禁止
001	T5IN 上的正跳变（上升沿）
010	T5IN 上的负跳变（下降沿）
011	T5IN 上的任意跳变（上升沿或下降沿）
101	T6 翻转锁存 T6OTL 上的正跳变（上升沿）
110	T6 翻转锁存 T6OTL 上的负跳变（下降沿）

T5I	计数器递增/递减计数触发沿
111	T6 翻转锁存 T6OTL 上的任意跳变（上升沿或下降沿）

注：只有 T6 上溢/下溢引发的 T6OTL 的状态改变才能触发 T5 计数；软件修改 T6OTL 的状态不会触发 T5 计数。

计数器模式下，必须将引脚 **T5IN** 配置为输入（对应的方向控制位 **DPx.y** 必须为 0）。计数器模式下允许的最大输入频率取决于所选择的预分频因子。为了确保能够正确识别 **T5IN** 上计数输入信号的跳变，输入电平必须至少保持（高或低）规定数目的模块时钟周期之后才能改变。具体内容请参阅[章节 14.2.6](#)。

定时器级联

溢出翻转锁存位 T6OTL 用作计数器模式下辅助定时器的计数时钟时，核心定时器 T6 和辅助定时器 T5 级联工作。定时器级联可构成 32 位或 33 位定时器/计数器，这取决于选择何种 T6OTL 跳变触发辅助定时器计数。

- **32 位定时器/计数器：**若选择 T6OTL 的任意跳变触发辅助定时器计数，核心定时器 T6 在每次上溢/下溢时，触发辅助定时器计数。因此 T5 和 T6 构成 32 位定时器。
- **33 位定时器/计数器：**若仅选择 T6OTL 的正跳变或负跳变触发辅助定时器计数，核心定时器 T6 在每两次上溢/下溢后，辅助定时器计数一次。因此它们构成 33 位定时器（16 位核心定时器 + T6OTL+ 16 位辅助定时器）。

只要 T6OTL 不被软件修改，它即表示内部翻转锁存器的状态，可被视为 33 位定时器的组成部分。

两个级联定时器的计数方向可以不同，从而为用户提供了多种不同的配置组合。

定时器 T6 代表级联定时器的低 16 位，在这种情况下，T6 可工作在定时器模式、门控定时器模式或计数器模式。

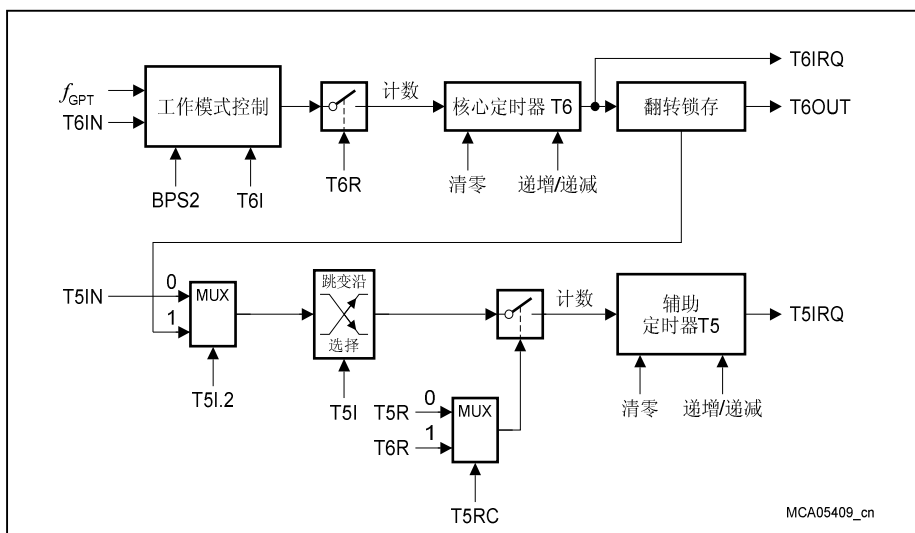


图 14-28 核心定时器 T6 和辅助定时器 T5 级联工作

14.2.5 GPT2 寄存器 CAPREL 工作模式

捕获/重载寄存器 CAPREL 可用于捕获定时器 T5 的内容，或者重载定时器 T6。系统支持一种特殊的模式，即通过寄存器 CAPREL 同时实现捕获、重载两种功能，采用该模式可实现倍频。输入引脚 CAPIN，或 GPT1 的定时器 T3 的 T3IN 和 T3EUD 触发捕获；定时器 T6 上溢或下溢触发重载。

除捕获功能之外，还可用捕获触发信号分别对定时器 T5 和 T6 清零。

寄存器 CAPREL 的功能由定时器控制寄存器 T5CON 和 T6CON 中相关的位/位域控制。

GPT2 的捕获/重载寄存器 CAPREL 工作在捕获模式

置位控制寄存器 T5CON 中的位 T5SC，寄存器 CAPREL 工作在捕获模式（将寄存器 T5CON 中的位域 CI 设置为非零值，选择触发信号）。捕获模式下，用户选择的外部输入引脚发生跳变时，辅助定时器 T5 的值被锁存到寄存器 CAPREL 中。位 CT3 选择由外部输入 CAPIN 或 GPT1 的定时器 T3 的输入 T3IN 和/或 T3EUD 作为捕获功能的触发源：CAPIN 上的正跳变、负跳变或任意跳变可触发捕获；或 T3IN 和 T3EUD 或二者任意一个输入上的跳变可触发捕获。寄存器 T5CON 中的位域 CI 控制选择有效沿。触发沿选择总结见表 14-14。

表 14-14 CAPREL 寄存器输入沿选择

CT3	CI	计数器递增/递减计数触发沿
X	00	无，捕获模式被禁止
0	01	CAPIN 上的正跳变（上升沿）
0	10	CAPIN 上的负跳变（下降沿）
0	11	CAPIN 上的任意跳变（上升沿或下降沿）
1	01	T3IN 上的任意跳变（上升沿或下降沿）
1	10	T3EUD 上的任意跳变（上升沿或下降沿）
1	11	T3IN 或 T3EUD 上的任意跳变（上升沿或下降沿）

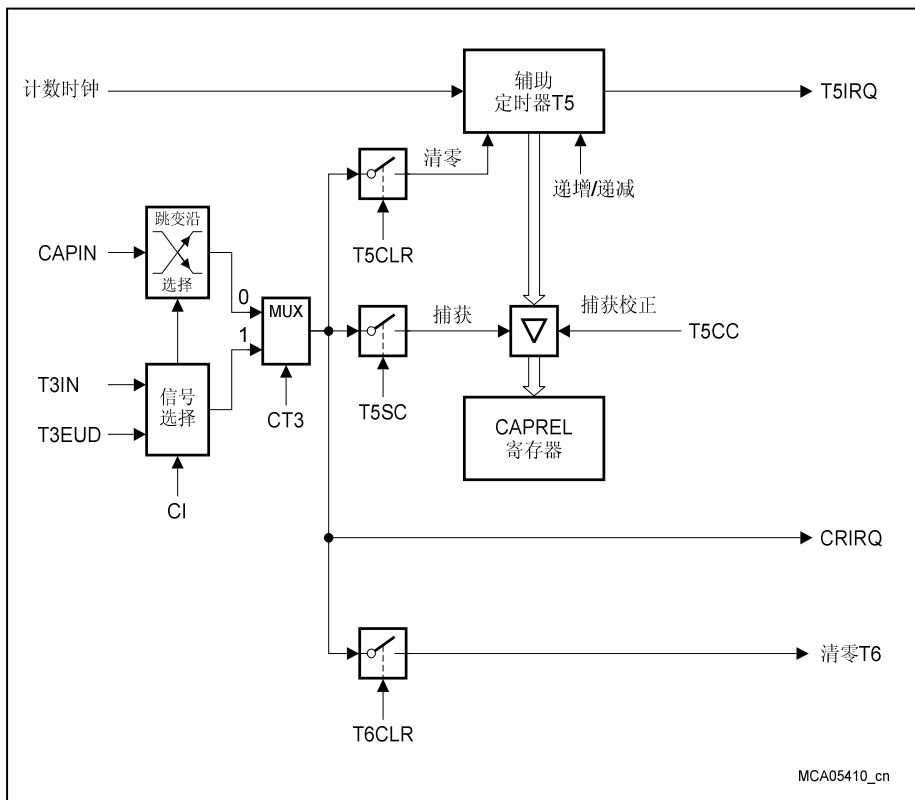


图 14-29 捕获模式下 GPT2 寄存器 CAPREL 框图

检测到触发事件时，辅助定时器 T5 的值被锁存到寄存器 CAPREL 中，中断请求线 CRIRQ 被激活；可通过寄存器 T5CON 中的位 T5CLR 和寄存器 T6CON 中的位 T6CLR 来控制该触发事件是否分别对定时器 T5 和/或 T6 清零。若 TxCLR = 0，捕获事件不影响定时器 Tx 的值；若 TxCLR = 1，捕获结束后定时器 Tx 被清零。

注：位 T5SC 仅控制是否执行捕获操作。若 T5SC 被清零，外部输入信号仍可对定时器 T5 和/或 T6 清零，或者用作外部中断输入。该中断由 CAPREL 中断控制寄存器 CRIC 控制。

若使能 T3IN 或 T3EUD (CT3 = 1) 触发捕获操作, 所选输入信号发生跳变时, T5 的内容被捕获到寄存器 CAPREL 中。可利用这些捕获值测量 T3 的输入信号。这种功能非常有用, 例如当 T3 工作在增量接口模式时, 可以获取输入信号的动态信息 (速度、加速度)。

捕获模式下，必须将引脚 **CAPIN**，**T3IN** 或 **T3EUD** 配置为输入。为了确保能够正确识别触发输入信号的跳变，输入电平必须至少保持（高或低）规定数目的模块时钟周期之后才能改变。具体内容请参阅[章节 14.2.6](#)。

GPT2 的捕获/重载寄存器 CAPREL 工作在重载模式

置位控制寄存器 **T6CON** 中的位 **T6SR**，寄存器 **CAPREL** 工作在重载模式。重载模式下，**T6** 上溢或下溢会触发重载操作，将寄存器 **CAPREL** 的值装入核心定时器 **T6** 中。**CAPREL** 寄存器的相关中断请求线 **CRIRQ** 不会被激活；但中断请求线 **T6IRQ** 会被激活，指示定时器 **T6** 发生上溢/下溢。

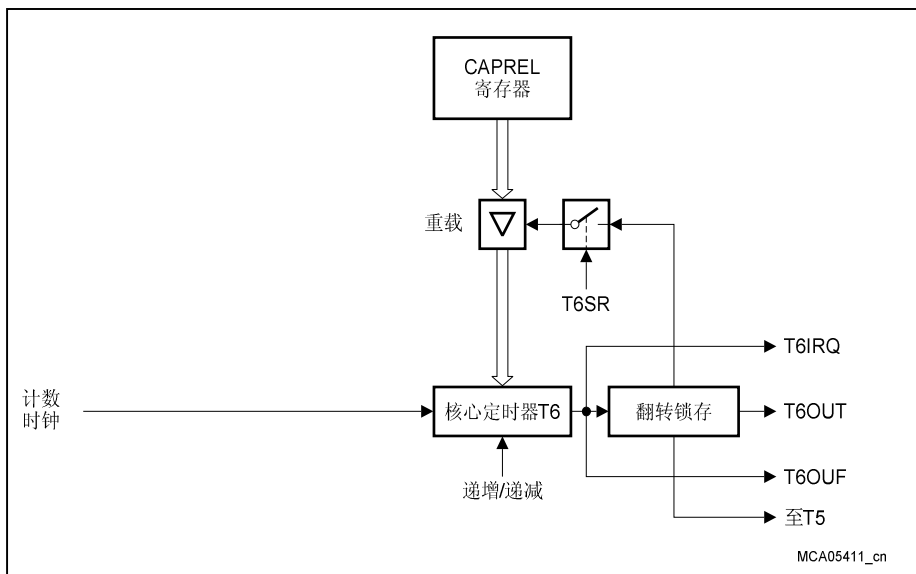


图 14-30 重载模式下 GPT2 寄存器 CAPREL 框图

GPT2 的捕获/重载寄存器 CAPREL 工作在捕获 – 重载模式

T5SC 和 T6SR 可分别使能寄存器 CAPREL 的重载和捕获功能，因此置位 T5SC 和 T6SR，可同时使能这两种功能。利用该特性可对输入频率倍增。

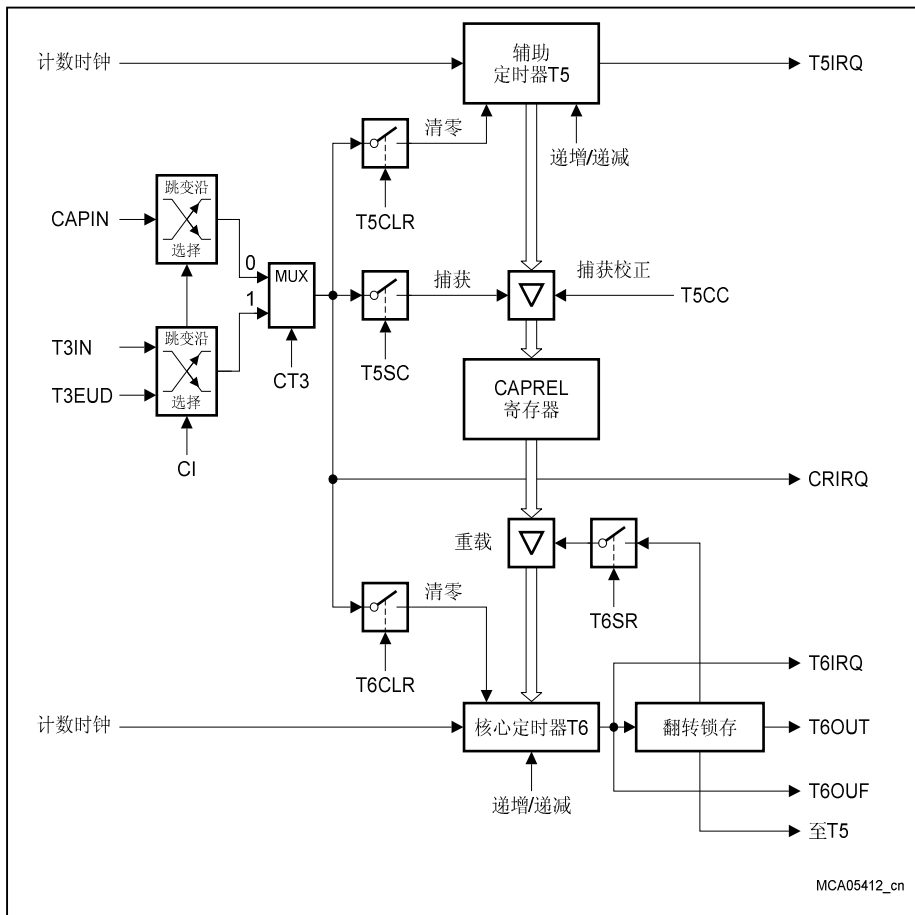


图 14-31 捕获 – 重载模式下 GPT2 寄存器 CAPREL 框图

可用该组合模式检测非周期性的连续外部事件，但需要采用更高的计数精度，这意味着，两次外部事件之间需要更多的“时钟标记”。

为了实现该功能，用定时器 T5 和 CAPREL 寄存器测量外部事件的时间间隔。定时器 T5 工作在定时器模式，以某个频率，如 $f_{GPT}/32$ 递增计数。外部事件从引脚 CAPIN 输入。外部事件发生时，定时器 T5 的内容被锁存到寄存器 CAPREL 中，定时器 T5 被清零（T5CLR = 1）。因此，寄存器 CAPREL 中始终保存两次事件之间的正确时间，用定时器 T5 的增量表示。定时器 T6 工作在定时器模式，以某个频率，如 $f_{GPT}/4$ 递减计数，发生下溢时将 CAPREL 的值重新装入 T6 中。因此，寄存器 CAPREL 的值代表定时器 T6 两次下溢之间的时间，用定时器 T6 的增量表示。由于（此例中）定时器 T6 的计数频率为 T5 的 8 倍，因此 T6 在两次外部事件之间会下溢 8 次，从而定时器 T6 的下溢信号会产生 8 个“时钟标记”。每次下溢时，中断请求线 T6IRQ 被激活，位 T6OTL 翻转。T6OTL 的状态可从引脚 T6OUT 输出。该信号的跳变频率为引脚 CAPIN 上信号频率的 8 倍。

注：定时器 T6 的下溢信号可进一步用作 CAPCOM2 单元定时器的输入时钟，从而用户可设置比外部事件计数精度更高的比较事件。通过 T6OUF 信号实现单元之间的连接。

捕获校正

以下原因会造成输出频率的偏差：定时器 T5 对实际时间计数（例如，工作在 1MHz 的 T5 对 10 kHz 的输入信号会计数到 $64_H/100_D$ ）；但是 T6OTL 只在 T6 下溢时翻转（如从 0000_H 翻转到 $FFFF_H$ ）。依照上面的举例，T6 将从 64_H 开始递减计数，因此 T6 每计数 101 次之后下溢。故实际输出频率为 79.2 kHz，而非期望值 80 kHz。

该偏差可通过激活捕获校正（T5CC = 1）加以补偿。使用捕获校正特性时，T5 值被减 1 之后才进行捕获。这样，误差即被校正（上例中，T5 计数到 $64_H/100_D$ ，但 CAPREL 寄存器会捕获减 1 之后的值 $63_H/99_D$ ，因此 T6 会计数 100 次后溢出，输出频率为 80 kHz）。

14.2.6 GPT2 时钟信号控制

定时器模块 GPT2 的所有操作均由基本时钟触发。系统时钟经过基本模块预分频产生该基本时钟，预分频因子由寄存器 T6CON 中的位域 BPS2 控制（见[图 14-20](#)）。有两种不同的方式产生计数时钟：

- **内部计数时钟**，经过预分频处理（预分频因子可编程设定）的 GPT2 基本时钟，该时钟用于（门控）定时器模式。
- **外部计数时钟**，取自定时器的输入引脚，该时钟用于计数器模式。

GPT2 模块的基本时钟决定了最大计数频率和定时器精度。

表 14-15 模块 GPT2 的基本时钟选择

模块预分频 ¹⁾	BPS2 = 01 _B	BPS2 = 00 _B ²⁾	BPS2 = 11 _B	BPS2 = 10 _B
GPT2 的预分频因子: F(BPS2)	F(BPS2) = 2	F(BPS2) = 4	F(BPS2) = 8	F(BPS2) = 16
最大外部计数频率	f _{GPT} /4	f _{GPT} /8	f _{GPT} /16	f _{GPT} /32
输入信号持续稳定的最短时间	2 × t _{GPT}	4 × t _{GPT}	8 × t _{GPT}	16 × t _{GPT}

1) 请注意位域 BPS2 为非线性编码。

2) 复位后的缺省值。

内部计数时钟产生

定时器模式和门控定时器模式下，GPT2 的每个定时器的计数时钟来自经过预分频处理的 GPT2 基本时钟，该预分频因子可由定时器控制寄存器 TxCON 中的位域 TxI 编程设定。

定时器 Tx 的计数频率 f_{Tx} 和相应计数精度 r_{Tx} 随较低的时钟频率线性变化，计算公式如下：

$$f_{Tx} = \frac{f_{GPT}}{F(BPS2) \times 2^{<TxI>}} \quad r_{Tx} [\mu s] = \frac{F(BPS2) \times 2^{<TxI>}}{f_{GPT} [MHz]} \quad (14.2)$$

计数频率不仅取决于公共的模块时钟预分频因子 F(BPS2)；还取决于定时器各自的预分频因子 2^{<TxI>}。GPT2 定时器的总预分频因子总结见[表 14-16](#)。

表 14-16 GPT2 内部计数时钟的总预分频因子

定时器 Tx 的 预分频因子	公共预分频因子 ¹⁾			
	BPS2 = 01 _B	BPS2 = 00 _B	BPS2 = 11 _B	BPS2 = 10 _B
Txl = 000_B	2	4	8	16
Txl = 001_B	4	8	16	32
Txl = 010_B	8	16	32	64
Txl = 011_B	16	32	64	128
Txl = 100_B	32	64	128	256
Txl = 101_B	64	128	256	512
Txl = 110_B	128	256	512	1024
Txl = 111_B	256	512	1024	2048

1) 请注意位域 BPS2 为非线性编码。

根据总预分频因子和系统频率计算得到定时器的各项参数（如计数时钟频率、计数时钟精度和计数周期），见**表 14-17**。请注意有些结果已四舍五入。

表 14-17 GPT2 定时器参数

系统时钟 = 10 MHz			总预 分频 因子	系统时钟 = 40 MHz		
频率	精度	周期		频率	精度	周期
5.0 MHz	200 ns	13.11 ms	2	20.0 MHz	50 ns	3.28 ms
2.5 MHz	400 ns	26.21 ms	4	10.0 MHz	100 ns	6.55 ms
1.25 MHz	800 ns	52.43 ms	8	5.0 MHz	200 ns	13.11 ms
625.0 kHz	1.6 μs	104.9 ms	16	2.5 MHz	400 ns	26.21 ms
312.5 kHz	3.2 μs	209.7 ms	32	1.25 MHz	800 ns	52.43 ms
156.25 kHz	6.4 μs	419.4 ms	64	625.0 kHz	1.6 μs	104.9 ms
78.125 kHz	12.8 μs	838.9 ms	128	312.5 kHz	3.2 μs	209.7 ms
39.06 kHz	25.6 μs	1.678 s	256	156.25 kHz	6.4 μs	419.4 ms

系统时钟 = 10 MHz			总预分频因子	系统时钟 = 40 MHz		
频率	精度	周期		频率	精度	周期
19.53 kHz	51.2 μ s	3.355 s	512	78.125 kHz	12.8 μ s	838.9 ms
9.77 kHz	102.4 μ s	6.711 s	1024	39.06 kHz	25.6 μ s	1.678 s
4.88 kHz	204.8 μ s	13.42 s	2048	19.53 kHz	51.2 μ s	3.355 s

外部计数时钟输入

用 GPT2 的基本时钟对外部输入信号采样（见 [图 14-20](#)）。为了确保信号能被正确识别，外部信号的当前输入电平值（高或低）必须至少保持一个完整的采样周期之后才能改变。若输入信号连续两次采样的电平值不同，即可识别到信号发生了跳变。因此，至少需要两个基本时钟周期对外部输入信号采样，故输入信号的最大频率一定不能高于基本时钟频率的一半。

GPT2 外部输入信号的参数限制总结见 [表 14-18](#)。

表 14-18 GPT2 外部输入信号的参数限制

系统时钟 = 10 MHz		外部输入频率因子	GPT2 分频因子 BPS2	输入信号持续时间	系统时钟 = 40 MHz	
最大输入频率	电平持续稳定的最短时间				最大输入频率	电平持续稳定的最短时间
2.5 MHz	200 ns	$f_{GPT}/4$	01_B	$2 \times t_{GPT}$	10.0 MHz	50 ns
1.25 MHz	400 ns	$f_{GPT}/8$	00_B	$4 \times t_{GPT}$	5.0 MHz	100 ns
625.0 kHz	800 ns	$f_{GPT}/16$	11_B	$8 \times t_{GPT}$	2.5 MHz	200 ns
312.5 kHz	1.6 μ s	$f_{GPT}/32$	10_B	$16 \times t_{GPT}$	1.25 MHz	400 ns

上表中的各项参数限制对 GPT2 的所有外部输入信号均有效，包括计数器模式下的外部计数信号和门控计数器模式下的门控输入信号。

14.2.7 GPT2 定时器寄存器

GPT12E_Tx

定时器 Tx 计数寄存器 SFR (FE4x_H/2y_H) 复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Txvalue															
rwh															

表 14-19 GPT2 定时器寄存器地址

定时器寄存器	物理地址	8 位地址
T5	FE46 _H	23 _H
T6	FE48 _H	24 _H

GPT12E_CAPREL

捕获/重载寄存器 SFR (FE4A_H/25_H) 复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Capture/Reloadvalue															
rwh															

14.2.8 GPT2 定时器和 CAPREL 的中断控制

当定时器（递增计数）从 FFFF_H 至 0000_H 发生上溢，或定时器（递减计数）从 0000_H 至 FFFF_H 发生下溢时，寄存器 TxIC 中的中断请求标志（T5IR 或 T6IR）被置位。只要检测到引脚 CAPIN 上的信号跳变（由位域 CI 选择跳变沿），寄存器 CRIC 中的中断请求标志 CRIR 被置位。若相应中断使能位（寄存器 TxIC 中的 T5IE 或 T6IE，CRIC 寄存器中的 CRIE）被置位，将会产生指向定时器或 CAPREL 中断向量（T5INT, T6INT 或 CRINT）的中断，或者触发 PEC 操作。两个定时器和 CAPREL 寄存器各对应一个中断控制寄存器。

GPT12E_T5IC

定时器 T5 中断控制寄存器 SFR (FF66_H/B3_H) 复位值: - - 00_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	GPX	T5IR	T5IE	ILVL				GLVL	
-	-	-	-	-	-	-	rw	rwh	rw	rw				rw	

GPT12E_T6IC

定时器 T6 中断控制寄存器 SFR (FF68_H/B4_H) 复位值: - - 00_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	GPX	T6IR	T6IE	ILVL				GLVL	
-	-	-	-	-	-	-	rw	rwh	rw	rw				rw	

GPT12E_CRIC

CAPREL 中断控制寄存器 SFR (FF6A_H/B5_H) 复位值: - - 00_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	GPX	CRIR	CRIE	ILVL				GLVL	
-	-	-	-	-	-	-	rw	rwh	rw	rw				rw	

注：控制位域的具体解释请参阅中断控制寄存器的描述。

14.3 GPT 模块接口

定时器单元模块 GPT1 和 GPT2 以两种基本方式和周围资源接口（见图 14-32）：

- **内部连接：** 定时器和片上资源（如时钟产生单元、中断控制器，或其它定时器）接口。
- **外部连接：** 定时器通过端口引脚和外部资源接口。

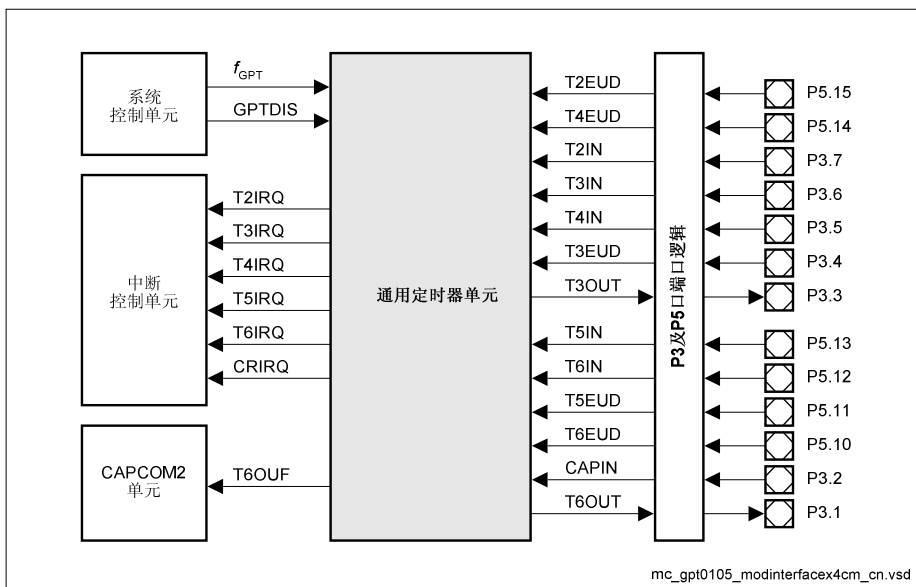


图 14-32 GPT 模块接口

必须将用作定时器输入信号的端口引脚配置为输入引脚，对应的方向控制位必须清零（ $DPx.y = 0$ ）。

必须将用作定时器输出信号的端口引脚配置为输出引脚，对应的方向控制位必须置位（ $DPx.y = 1$ ）。并通过各自的复用功能选择寄存器将这些引脚设置为定时器的输出。（见第 7 章）。

必须使能定时器中断请求的中断节点并设定其中断优先级。

15 实时时钟

XC164CM 的实时时钟（RTC）模块由一组预分频器和定时器级联构成。主振荡器经预分频处理为 RTC 提供计数时钟。RTC 的应用包括：

- 48 位定时器，测量长时间间隔
- 系统时钟，指示当前时间和日期
 （RTC 的结构支持由系统时钟直接指示时间和日期）
- 周期性中断（可由级联链中的任意一个定时器产生）

不同的编程设定和中断请求信号可使 RTC 满足不同的应用需求。即使 XC164CM 工作在省电模式 RTC 仍可继续工作，从而提供了系统的实时信息。

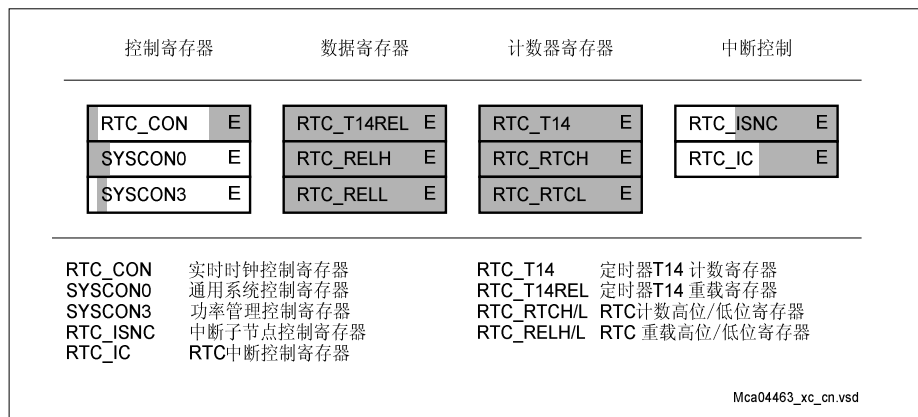


图 15-1 RTC 模块相关的特殊功能寄存器（SFR）

RTC 模块由 3 个分频模块组成：

- 8:1 分频器（可编程选择开启或关闭）
- 可重载的 16 位定时器 T14
- 32 位 RTC 定时器模块（通过 RTC_RTCH 和 RTC_RTCL 进行访问），由以下定时器级联构成：
 - 可重载的 10 位定时器 CNT0
 - 可重载的 6 位定时器 CNT1
 - 可重载的 6 位定时器 CNT2
 - 可重载的 10 位定时器 CNT3

所有定时器均递增计数。五个定时器均可产生各自的中断请求，所有的中断请求合成为一个公共中断节点请求。

注：为了确保始终能够指示正确的系统时间（即使在中间复位的情况下），RTC 寄存器不受系统复位的影响。

15.1 确定 RTC 的时间基准

RTC 的定时器链由时钟信号 f_{RTC} 触发计数，该时钟信号由主振荡器经预分频处理后产生（见 [图 15-2](#) 和 [图 15-3](#)）。 f_{RTC} 可直接或通过 8:1 预分频构成基本的 RTC 时钟。定时器 T14 可提供不同应用情况下的计数增量，从而确定了 RTC 定时器的输入频率，即 RTC 的时间基准（见 [表 15-3](#)）。

XC164CM 的系统时钟 f_{SYS} 也为 RTC 模块提供时钟信号。该时钟信号控制 RTC 的逻辑模块和总线接口。为了能够与计数时钟正确同步，系统时钟必须至少是计数时钟的四倍，即 $f_{SYS} \geq 4 \times f_{CNT}$ 。

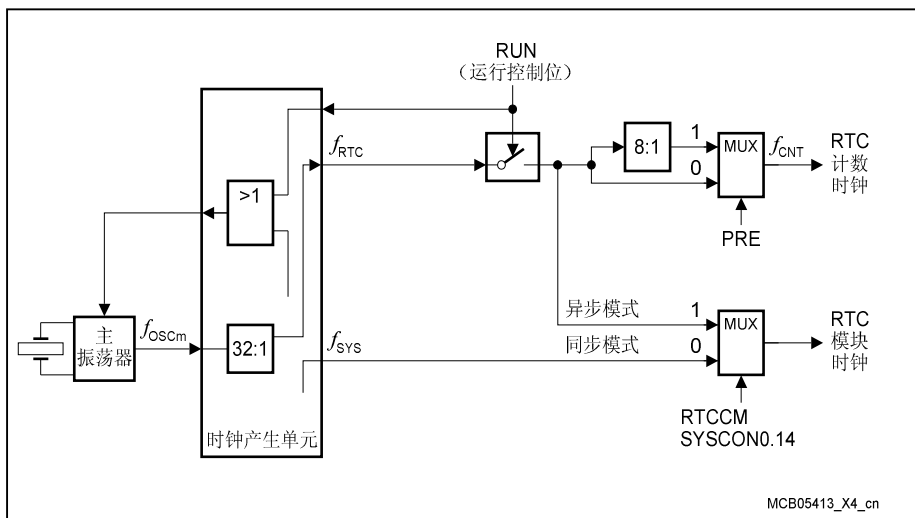


图 15-2 RTC 时钟源框图

表 15-1 例举 T14 中断周期的取值范围和 T14 的重载值（时间基准为 1s 和 1ms）：

表 15-1 RTC 时间基准举例

振荡器 频率	T14 中断周期		重载值 A		重载值 B	
	最小值	最大值	T14REL	时间基准	T14REL	时间基准
4 MHz	8.0 μ s	4.194 s	---- _H / C2F7 _H	1.000 s	FF83 _H / FFF0 _H	1.000 ms/ 1.024 ms

注：根据是否使能 8:1 预分频器，从这对重载值中选择相应的值。

异步工作

当系统时钟频率低于 $4 \times f_{CNT}$ 时，将无法与计数时钟正确同步、计数事件可能会被遗漏；当 XC164CM 进入比如休眠模式，系统时钟会彻底关闭，RTC 将停止计数。

发生上述情况时，RTC 可切换到异步模式（通过置位寄存器 SYSCON0 中的位 RTCCM 实现）。异步模式下，计数寄存器直接由计数时钟控制，和系统时钟无关。RTC 异步工作确保了即使在休眠模式或掉电模式下，仍能正确计时。

但是，在异步模式下，由于无法维持计数寄存器和总线接口之间的同步，RTC 寄存器不能被写入。读操作可能会干扰计数事件，因此必须对读操作进行验证（比如，连续三次读同一个值）。

注：当然，只有在系统时钟未关闭时，考虑异步模式对寄存器读写操作的限制才有意义。

时钟模式切换

由寄存器 SYSCON0 中的位 RTCCM 选择 RTC 的时钟模式（同步或异步）。RTC 复位之后（由寄存器 SYSCON0 中的 RTCRST 触发），实时时钟系统工作在同步模式（RTCCM = 0）且 8:1 预分频被使能。

时钟模式的选择还影响对 RTC 寄存器的访问。寄存器 RTC_CON 中的位 ACCPOS 指示是否可对寄存器进行完全访问（复位后的缺省值 ACCPOS = 1 表示可对寄存器读写；ACCPOS = 0 反之）；ACCPOS 也指明了当前的时钟模式。

请注意：软件应查询位 ACCPOS 的值，以确认是否转移到所期望的时钟模式。

切换至异步模式（RTCCM = 1）后，位 ACCPOS = 0 指示异步模式下的正确操作。在该模式下系统时钟可被关闭或其频率被降低。

切换至同步模式（RTCCM = 0）后，位 ACCPOS = 1 指示同步模式下的正确操作。在该模式下可重新访问（读写）RTC 寄存器。

注：若 8:1 预分频被禁止，RTC 从同步模式切换到异步模式时可能会丢失一个计数事件（ t_{CNT} 的跳变沿）。因此对于此类应用，推荐使能 8:1 预分频。

软件校正提高 RTC 精度

振荡器频率和预分频因子（包括或不包括 T14 和 8:1 预分频器）决定了 XC164CM RTC 的精度。预分频产生的精度限制由二进制计数器的量化引起（平均值为 0）；振荡器频率产生的精度限制由理想频率和实际频率之间的差值引起（因此差值会随时间累加），该影响可被预估并补偿。针对时间精度要求较高的应用，可通过软件进一步提高 RTC 总精度。

提高精度的关键是要知道实际的振荡频率。可通过实际频率和期望频率之间的误差测量 RTC 的偏差。可很容易的计算得出：经过多少（N）周期该误差将累积到 ± 1 个周期。因此，只需在每 N 个周期之后即可 ± 1 校正计数脉冲。比如可通过中断服务程序实现周期性的误差校正；或在需要时对所读取的 RTC 寄存器数值进行评估以实现差值校正（使用这种方式需储存 RTC 的前一个值）。

注：不过对于大多数应用，RTC 所提供的标准精度已远远满足应用需要。

调整 RTC 的当前值需先读取再写入完整的 48 位数值，只有通过连续三次访问方可完成。为了避免多字读取/写入发生冲突，RTC 可增加或减少一个计数脉冲进行纠错。

这可通过置位寄存器 RTC_CON 中的 T14INC 或 T14DEC 实现：下次计数事件到来时增加一个额外的计数脉冲（T14INC），或减少一个计数脉冲（T14DEC）。操作结束前这些位始终保持置位状态，操作结束后硬件自动对相关位清零。

注：同时置位 T14INC 和 T14DEC 将不会影响计数值。

15.2 RTC 运行控制

若启动 RTC，必须置位寄存器 RTC_CON 中的位 RUN（复位后的缺省值）。可以将位 RUN 清零，比如在计时过程中，某些操作阶段 RTC 无需参与工作。置位寄存器 SYSCON3 中的位 RTCDIS 可彻底关闭 RTC 模块。

注：需要有效的计数时钟才可控制 RTC 的正确操作。

软件置位寄存器 SYSCON0 中的位 RTCRST 将触发 RTC 复位。此时 RTC 的所有寄存器均被设置为初始值，位 RTCRST 被自动清零。正常的系统复位不会影响 RTC 寄存器的内容及操作（但 RTC_IC 会被复位）。初始化程序必须确保 RTC 工作在恰当的工作模式下。

RTC 控制寄存器 RTC_CON 选择 RTC 模块的基本操作。

RTC_CON

控制寄存器

ESFR (F110H/88H)

复位值: 8003H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACC POS	-	-	-	-	-	-	-	-	-	-	-	T14 INC	T14 DEC	PRE	RUN
rh	-	-	-	-	-	-	-	-	-	-	-	rwh	rwh	rw	rw

符号	位序号	读写类型	功能描述
ACCPOS	15	rh	RTC 寄存器访问控制 0 不能执行写操作，只能异步读取 1 可执行读写操作
T14INC	3	rwh	定时器 T14 加 1 该位置位时，下次计数事件到来时增加一个计数脉冲，即 T14 加 1。 加 1 操作完成后硬件自动对该位清零。
T14DEC	2	rwh	定时器 T14 减 1 该位置位时，下次计数事件到来时减少一个计数脉冲，即 T14 减 1。 减 1 操作完成后硬件自动对该位清零。
PRE	1	rw	RTC 输入时钟预分频使能 0 预分频被禁止，T14 的输入时钟为 f_{RTC} 1 预分频被使能，T14 的输入时钟为 $f_{RTC}/8$

符号	位序号	读写类型	功能描述
RUN	0	rw	RTC 运行位 0 RTC 停止 1 RTC 运行

15.3 RTC 工作模式

根据应用需求，RTC 可被配置为不同的工作模式。这些工作模式的设置可通过选择恰当的重载值和中断信号来实现。

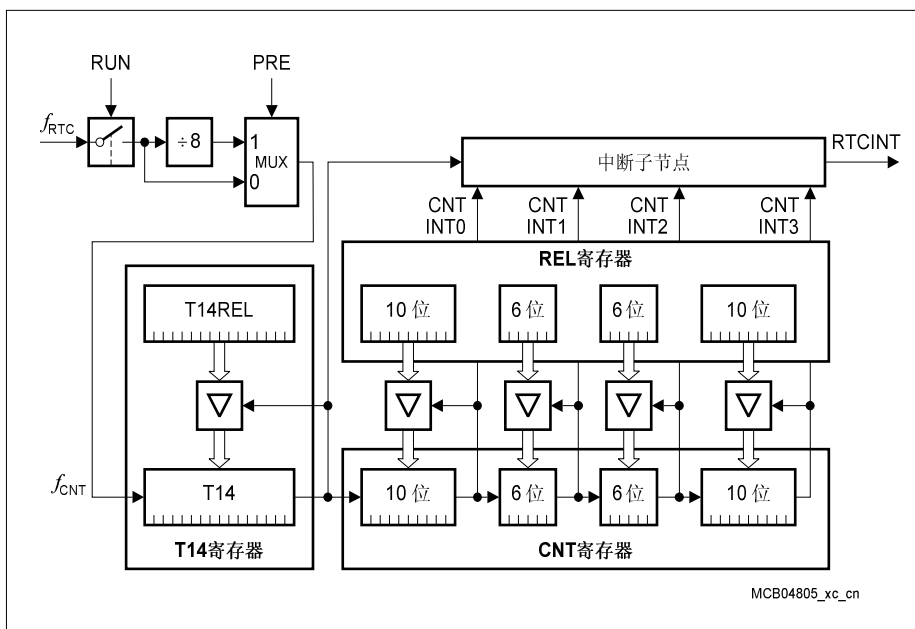


图 15-3 RTC 框图

RTC 寄存器访问

RTC 的实际值由三个寄存器 T14、RTCL 和 RTCH 表示。由于这些寄存器级联构成 RTC 的计数器链，RTC 运行时会产生内部溢出。读取或写入 RTC 值时，必须考虑可能的内部溢出以避免读取/写入错误值。

读取定时器的值时必须谨慎：因为需要对不同的寄存器执行三次读操作，读操作之间存在固有的时间延迟。必须考虑到在两次读取访问之间可能发生的溢出(T14 溢出送至 RTCL 和/或 RTCL 溢出送至 RTCH)，而造成读写错误。

例如，从 RTCH 中读取/向 RTCH 中写入 0000_H后，在访问 RTCL 时会产生错误值，因为此时 RTCL 可能产生溢出。这种情况下，RTCH 将变为 0001_H。访问 T14 和 T14REL 时也应进行同样的处理。

通过专用地址访问定时器 T14 和 T14 重载寄存器：通过两个 16 位 RTC 定时器寄存器 RTCH 和 RTCL 访问 RTC 的四个计数器 CNT3...CNT0；通过两个 16 位 RTC 重载寄存器 RELH 和 RELL 访问四个对应的重载值 REL3...REL0。

表 15-2 定时器 T14 的寄存器地址

寄存器名称	长/短地址	复位值	注
RTC_T14	F0D2 _H /69 _H	0000 _H	16 位定时器，可用作 RTC 模块预分频
RTC_T14REL	F0D0 _H /68 _H	0000 _H	定时器 T14 的重载寄存器

RTC_RTCH

RTC 定时器高位寄存器 ESFR (F0D6_H/6B_H) 复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT3										CNT2					
rwh										rwh					

RTC_RTCL

RTC 定时器低位寄存器 ESFR (F0D4_H/6A_H) 复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT1						CNT0									
rwh						rwh									

符号	位序号	读写类型	功能描述
CNTx (x = 3 ... 0)	[15:6] [5:0] [15:10] [9:0]	rwh	RTC 定时器计数值 CNTx 这些位域溢出时会触发计数脉冲，控制下一个计数器 CNTx+1（CNT3 除外）计数，紧接着将位域 RELx 的值重新装入 CNTx。 此外，溢出会触发中断请求。

RTC_RELH

RTC 重载高位寄存器 **ESFR (F0CE_H/67_H)** **复位值: 0000_H**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REL3										REL2					
rwh										rwh					

RTC_RELL

RTC 重载低位寄存器 **ESFR (F0CC_H/66_H)** **复位值: 0000_H**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REL1						REL0									
rwh						rwh									

符号	位序号	读写类型	功能描述
RELx (x = 3 ... 0)	[15:6] [5:0] [15:10] [9:0]	rwh	RTC 重载值 RELx CNTx 计数溢出时，将该值复制到位域 CNTx 中。

注：只有专用 RTC 复位的情况下，所有的 RTC 寄存器才会复位。该复位操作不受系统复位控制，而是通过软件实现。

15.3.1 48 位定时器操作

定时器 T14 和 COUNT0...COUNT3 级联可构成 48 位定时器，由 RTC 的输入时钟控制计数（时钟的预分频可以选择）。重载寄存器 T14REL, REL1 和 RELH 必须被清零以产生真正的二进制 48 位定时器。不过也允许采用其它重载值。RTC 的各定时器使用非零重载值时，各自的溢出情况不同，因此必须谨慎使用非零重载值。

最大可用的时间跨度为 2^{48} ($\approx 10^{14}$) 个 T14 输入时钟（假定无预分频），振荡器频率为 32 kHz 时，该时间值将超过 200 年。

15.3.2 系统时钟操作

由于 RTC 模块不受系统复位¹⁾的影响，因此可以保持实时系统时钟，即使在省电模式下实时系统时钟仍可正常工作并指示当前时间和日期。

实时系统时钟的精度由定时器 T14 的输入时钟决定。选择恰当的重载值，各级联定时器可分别表示当前时间和/或日期的一部分。T14 可将 RTC 调整到期望的工作范围（时间或日期）。T14REL 的重载值为 0000_H 时对应可用的最大时间跨度，即 T14 将输入时钟 2^{16} 分频。

系统时钟举例

RTC 的计数时钟为 $f_{OSCm}/32$ (8:1 预分频关闭)。当 $f_{OSCm} = 4 \text{ MHz}$ 时，计数时钟 $f_{CNT} = 125 \text{ kHz}$ 。选择恰当的重载值，RTC 的定时器可直接表示当前时间（见 [图 15-4](#) 和 [表 15-3](#)）。

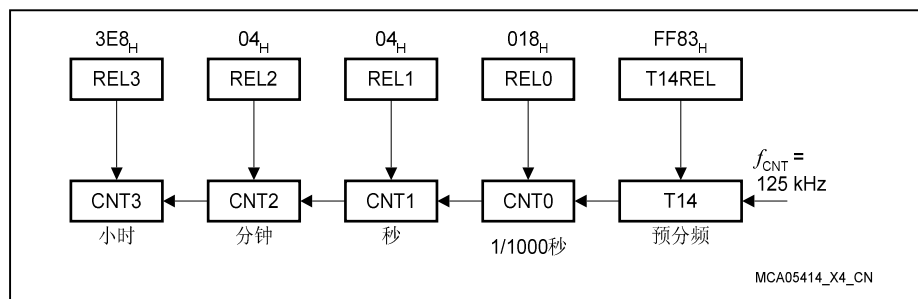


图 15-4 RTC 定时器配置举例

注：该配置可每毫秒、每秒、每分钟、每小时或每天产生一个中断请求。

1) 上电复位后 RTC 寄存器的值未定义。

级联链中的每个定时器递增计数时，将输入时钟（ $2^{< \text{定时器宽度} > - < \text{重载值} >}$ ）:1 分频。

表 15-3 给出在表示一天/一周时间时，各定时器所对应的重载值。

表 15-3 重载值举例

	REL3	REL2	REL1	REL0	T14REL
一天的时间表示（图 15-4）					
公式	$2^{10}-24$	2^6-60	2^6-60	$2^{10}-1000$	$2^{16}-125$
重载值	3E8 _H	04 _H	04 _H	018 _H	FF83 _H
功能	小时	分钟	秒	1/1000 秒	预分频
中断周期	天	小时	分钟	秒	毫秒
一周的时间表示					
公式	$2^{10}-7$	2^6-24	2^6-60	$2^{10}-600$	$2^{16}-12500$
重载值	3F9 _H	28 _H	04 _H	1A8 _H	CF2C _H
功能	天	小时	分钟	1/10 秒	预分频
中断周期	周	天	小时	分钟	100 毫秒

15.3.3 周期性中断的产生

只要 RTC 的任一定时器溢出和被重载，RTC 模块产生中断请求。该中断请求可用作比如：（无需重载通用定时器）提供和 CPU 频率无关的系统时间标记；或者规律性的将系统从休眠模式唤醒。选择恰当的重载值并使能对应的中断请求，可调整中断周期的长度。

该模式下，可和其它模式组合工作。比如，重载值 T14REL = F9C0_H（ $2^{16} - 1600$ ），这样，每隔 50 ms 产生一个 T14 中断请求，规律性的唤醒系统。而级联链中 T14 之后的定时器仍可用于表示时间或构成二进制计数器，只是时间基准不同。

15.4 RTC 中断产生

RTC 定时器链中每个定时器溢出时均可产生中断请求。RTC 的中断子节点控制寄存器 **ISNC** 将这些中断请求信号合并起来激活 RTC 的公共中断请求线 **RTC_IRQ**。

每个定时器溢出时，寄存器 **ISNC** 中对应的中断请求标志被置位。对应的中断使能位则决定该中断请求是否激活公共中断请求线。被使能的中断请求“相或”之后连接到 **RTC_ITRQ** 线上（见图 15-5）。

中断处理器通过查询中断请求标志确定中断请求源，并在执行适当的操作后必须对这些标志清零（不由硬件清零）。CPU 转入执行中断处理器指向的中断服务程序时，公共节点请求位被自动清零。

注：如果只有一个中断请求源被使能，自然不需额外的软件检查。该中断请求和公共中断节点都必须被使能。

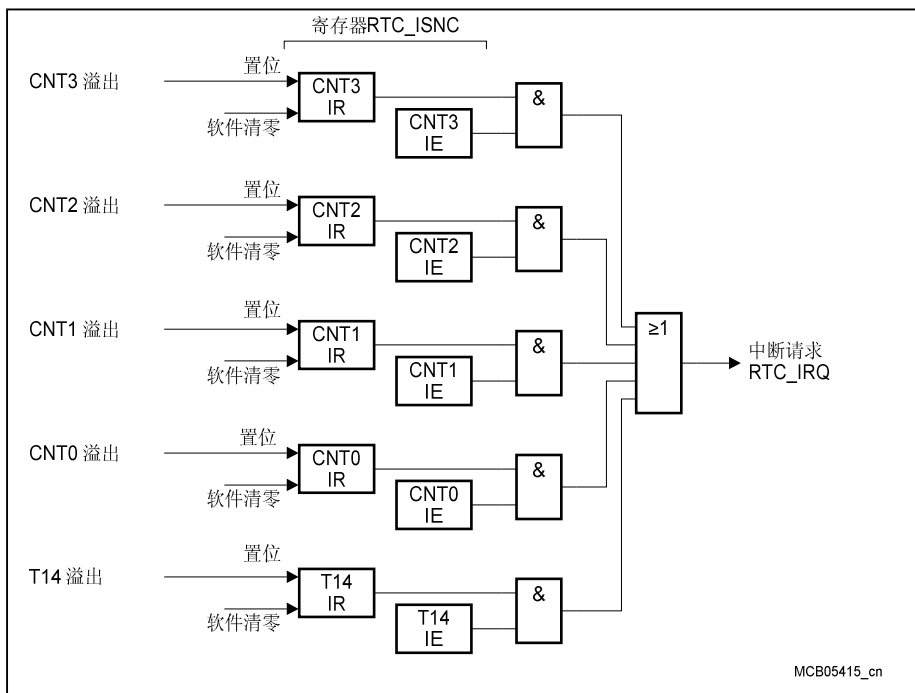


图 15-5 中断框图

RTC_ISNC

中断子节点控制寄存器

ESFR (F10C_H/86_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	CNT 3IR	CNT 3IE	CNT 2IR	CNT 2IE	CNT 1IR	CNT 1IE	CNT 0IR	CNT 0IE	T14 IR	T14 IE
-	-	-	-	-	-	rwh	rw	rwh	rw	rwh	rw	rwh	rw	rwh	rw

符号	位序号	读写类型	功能描述
CNTxIR (x = 3...0)	9, 7, 5, 3	rwh	CNTx 中断请求标志 0 无中断请求挂起 1 已产生中断请求
CNTxIE (x = 3...0)	8, 6, 4, 2	rw	CNTx 中断使能控制位 0 中断请求被禁止 1 中断请求被使能
T14IR	1	rwh	T14 溢出中断请求标志 0 无中断请求挂起 1 已产生中断请求
T14IE	0	rw	T14 溢出中断使能控制位 0 中断请求被禁止 1 中断请求被使能

注：寄存器 ISNC 中的中断请求标志必须由软件清零，进入中断服务程序时这些标志位不会被硬件自动清零。

RTC_IC

RTC 中断控制寄存器

ESFR (F1A0_H/D0_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	GPX	RTC IR	RTC IE	ILVL				GLVL	
-	-	-	-	-	-	-	rw	rwh	rw	rw				rw	

注：控制位域的具体解释请参阅中断控制寄存器的描述。

寄存器 RTC_IC 不在 RTC 模块中，任何系统复位均可复位该寄存器。

16 模数转换器

XC164CM 系列中的几款产品（见**表 1-1**）提供精度为 8 位或 10 位，集成采样和保持电路的模数转换器（ADC）。输入复用器可从多达 14 路模拟输入通道（P5 口的复用功能）中，通过软件（固定通道模式）或自动（自动扫描模式）选择转换通道。

为了满足大多数嵌入式控制应用的需要，ADC 支持以下转换模式：

- **固定通道单次转换**
对选中通道进行转换并产生（一个）转换结果
- **固定通道连续转换**
对选中通道重复转换
- **自动扫描单次转换**
对选中通道组中的每路通道进行转换，各通道产生一个转换结果
- **自动扫描连续转换**
对选中通道组重复转换
- **等待 ADDAT 读取模式**
前次的转换结果读取之后自动启动转换
- **通道插入模式**

当硬件触发时启动转换，可将指定通道的转换插入转换组中（自动扫描）

通过 SFR 寄存器组和端口引脚控制 ADC 的功能并读取转换结果。增强模式寄存器提供更多 ADC 控制功能。

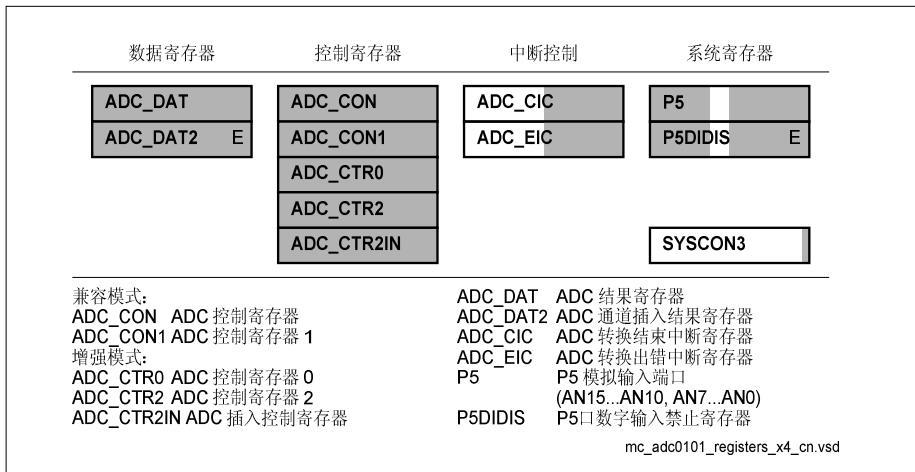


图 16-1 ADC 的相关特殊功能寄存器（SFR）和端口引脚

外部模拟参考电压 V_{AREF} 和 V_{AGND} 固定不变。给 ADC 独立供电降低了对其它数字信号的干扰。在复位校准阶段及整个转换过程中，参考电压必须保持稳定以达到最大精度。

采样时间和转换时间皆可编程设定，因此可根据模拟输入和/或模拟参考电压的内阻不同，调整 ADC 的采样和转换时间（用户可参阅应用指南 AP2428）。

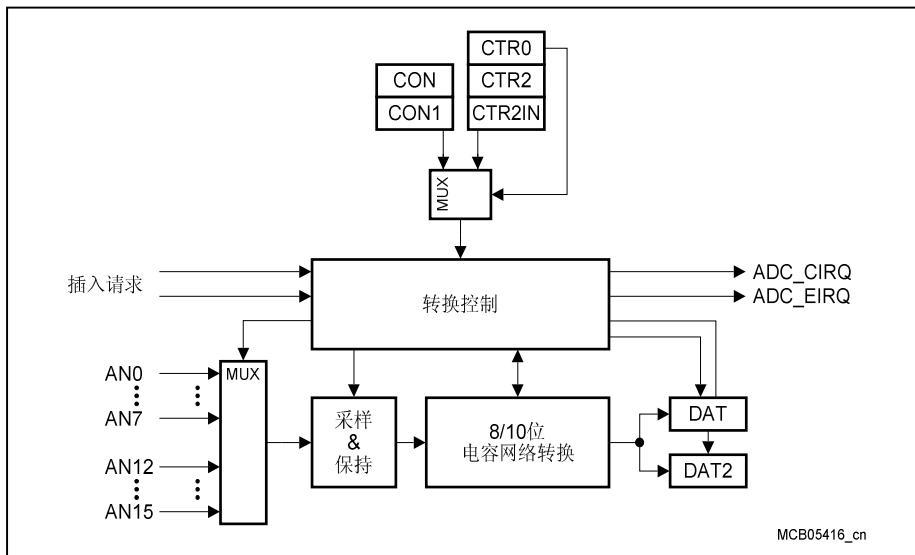


图 16-2 模数转换器框图

ADC 是采用逐次逼近技术、以电容网络实现的。转换过程包括以下 3 个阶段：

- 采样阶段：将电容网络和选中的模拟输入端相连，对电容网络充放电直至达到模拟信号的电压值。
- 实际转换阶段：将电容网络和模拟输入端断开，在逐次逼近过程中通过参考电压 **VAREF** 反复对电容网络充放电。
- （可选的）后校准阶段（调整电容网络以适用于变化的条件，如温度）：将校准后的结果写入结果寄存器，并产生中断请求。

ADC 中有两组分别对应兼容模式和增强模式的控制、数据和状态寄存器。在给定时间内只有一组寄存器有效。由于两组寄存器中的大多数位元或位域控制相同的功能，或者控制方式相似，因此在以下章节中，针对功能（而非针对两个寄存器组）进行描述。

16.1 模式选择

模拟输入通道 AN15...AN10, AN7...AN0 是输入单向口 P5 的复用功能。P5 口可输入模拟或数字信号。用作模拟输入引脚时，建议通过寄存器 P5DIDIS 禁止数字信号输入，从而当（模拟）输入信号的电压在 V_{IL} 和 V_{IH} 之间时，避免了不必要的串扰电流和切换噪声。

由两组可位寻址的控制寄存器控制 ADC 的功能。兼容模式下使用寄存器 ADC_CON 和 ADC_CON1；增强模式下使用寄存器 ADC_CTR0、ADC_CTR2 和 ADC_CTR2IN。寄存器指定转换通道的编号、转换模式、并指示转换器的状态。

16.1.1 兼容模式

兼容模式下 ($MD = 0$)，寄存器 ADC_CON 和 ADC_CON1 选择 ADC 的基本功能。尽管寄存器所提供的功能有限，但其结构已与先前版本的 ADC 模块兼容。

ADC_CON

ADC 控制寄存器

SFR (FFA0H/D0H)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADCTC	ADSTC	ADCRQ	ADCIN	ADWR	ADBSY	ADST	-	ADM	ADCH						
rw	rw	rwh	rw	rw	rh	rwh	-	rw	rw						

符号	位序号	读写类型	功能描述
ADCTC	[15:14]	rw	ADC 转换时间控制 （规定 ADC 的基本转换时钟 f_{BC} ） 00 $f_{BC} = f_{ADC}/4$ 01 $f_{BC} = f_{ADC}/2$ 10 $f_{BC} = f_{ADC}/16$ 11 $f_{BC} = f_{ADC}/8$
ADSTC	[13:12]	rw	ADC 采样时间控制 （规定 ADC 的采样时间） 00 $t_{BC} \times 8$ 01 $t_{BC} \times 16$ 10 $t_{BC} \times 32$ 11 $t_{BC} \times 64$
ADCRQ	11	rwh	ADC 通道插入请求标志

符号	位序号	读写类型	功能描述
ADCIN	10	rw	ADC 通道插入使能
ADWR	9	rw	ADC 等待读取控制
ADBSY	8	rh	ADC 忙碌标志 0 ADC 空闲 1 正在进行转换
ADST	7	rwh	ADC 启动控制位 0 终止正在进行的转换 1 启动转换
ADM	[5:4]	rw	ADC 模式选择 00 固定通道单次转换 01 固定通道连续转换 10 自动扫描单次转换 11 自动扫描连续转换
ADCH	[3:0]	rw	ADC 模拟通道输入选择 选择（第一个）将转换的通道

ADC_CON1

ADC 控制寄存器 1

SFR (FFA6_H/D3_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ICST	SAM PLE	CAL	RES	ADCTC						ADSTC					
rw	rh	rh	rw	rw						rw					

符号	位序号	读写类型	功能描述
ICST	15	rw	改进型转换和采样时间 选择有效的时间控制位域。 0 标准转换和采样时间控制，寄存器 ADC_CON 中的 2 位位域（复位后的缺 省值） 1 改进型转换和采样时间控制，寄存器

符号	位序号	读写类型	功能描述
			ADC_CON1 中的 6 位位域
SAMPLE	14	rh	采样阶段状态标志 0 ADC 不处于采样阶段 1 ADC 正在采样
CAL	13	rh	复位校准阶段状态标志 0 ADC 不处于校准阶段 1 ADC 正在校准
RES	12	rw	转换精度控制 0 10 位精度（复位后的缺省值） 1 8 位精度
ADCTC	[11:6]	rw	ADC 转换时间控制 规定 ADC 的基本转换时钟： $f_{BC} = f_{ADC} / (<ADCTC> + 1)$
ADSTC	[5:0]	rw	ADC 采样时间控制 规定 ADC 的采样时间： $t_S = t_{BC} \times 4 \times (<ADSTC> + 2)$

注：选择 ADCTC 和 f_{ADC} 时，一定不能超过 f_{BC} 的极限值（请参阅数据手册）。

16.1.2 增强模式

增强模式下（MD = 1），寄存器 ADC_CTR0, ADC_CTR2 和 ADC_CTR2IN 选择 ADC 的基本功能。增强模式的寄存器结构和兼容模式有所不同，该模式提供了更多的功能选择。

由寄存器 ADC_CTR2(IN) 选择转换时间，ADC_CTR2 控制标准转换，ADC_CTR2IN 控制插入转换。

ADC_CTR0

ADC 控制寄存器 0

SFR (FFBE_H/DF_H)

复位值: 1000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MD	SAM PLE	ADCTS	AD CRQ	AD CIN	AD WR	AD BSY	AD ST	ADM	CAL OFF	ADCH					
rw	rh	rw	rwh	rw	rw	rh	rwh	rw	rw	rw					

符号	位序号	读写类型	功能描述
MD	15	rw	模式控制 0 兼容模式 1 增强模式 <i>注：正在转换时，禁止修改控制位 MD。用户编程时必须注意。</i>
SAMPLE	14	rh	采样阶段状态标志 0 ADC 不处于采样阶段 1 ADC 正在采样
ADCTS	[13:12]	rw	通道插入触发输入的选择 00 通道插入触发输入被禁止 01 选择触发输入 CAPCOM2 10 选择触发输入 CAPCOM6 11 保留 <i>注：ADCTS 的复位值为 01_B 以实现兼容</i>
ADCRQ	11	rwh	ADC 通道插入请求标志
ADCIN	10	rw	ADC 通道插入使能控制
ADWR	9	rw	ADC 等待读取控制
ADBSY	8	rh	ADC 忙碌标志 0 ADC 空闲 1 正在进行转换
ADST	7	rwh	ADC 启动/停止控制位 0 终止正在进行的转换 1 启动转换

符号	位序号	读写类型	功能描述
ADM	[6:5]	rw	ADC 模式选择控制 00 固定通道单次转换 01 固定通道连续转换 10 自动扫描单次转换 11 自动扫描连续转换
CALOFF	4	rw	校准禁止控制 0 执行校准 1 禁止（关闭）校准 <i>注：兼容模式和增强模式下该位均有效</i>
ADCH	[3:0]	rw	ADC 模拟通道输入选择 选择（第一个）将转换的通道

ADC_CTR2

ADC 控制寄存器 2

ESFR (F09C_H/4E_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	RES			ADCTC						ADSTC					
-	rw			rw						rw					

ADC_CTR2IN

ADC 插入控制寄存器 2

ESFR (F09E_H/4F_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	RES			ADCTC						ADSTC					
-	rw			rw						rw					

符号	位序号	读写类型	功能描述
RES	[13:12]	rw	转换精度控制 00 10 位精度（复位后的缺省值） 01 8 位精度 1x 保留
ADCTC	[11:6]	rw	ADC 转换时间控制

符号	位序号	读写类型	功能描述
			规定 ADC 的基本转换时钟： $f_{BC} = f_{ADC} / (<ADCTC> + 1)$
ADSTC	[5:0]	rw	ADC 采样时间控制 规定 ADC 的采样时间： $t_S = t_{BC} \times 4 \times (<ADSTC> + 2)$

注：选择 *ADCTC* 和 f_{ADC} 时，一定不能超过 f_{BC} 的极限值（请参阅数据手册）。

16.2 ADC 操作

通道选择，ADCH

位域 ADCH 控制输入通道复用器的逻辑操作。单通道模式下，ADCH 指定要进行转换的模拟通道；自动扫描模式下，ADCH 指定要转换的通道组中编号最高的通道。

在转换过程中可以修改 ADCH。固定通道模式下当前转换结束后、或自动扫描模式下当前转换序列结束后，ADCH 的新值生效。

ADC 标志，ADBSY，SAMPLE

ADC 启动（置位 ADST）后 ADC 忙碌状态标志被置位，只要 ADC 在执行转换或者校准，该位始终被置位。

ADC 空闲时 ADBSY 被清零，表明当前没有转换或校准操作。

在采样过程中置位 SAMPLE。

ADC 启动/停止控制，ADST

通过位 ADST 启动或停止 ADC。置位 ADST 将启动单次转换或一个转换序列。

ADC 启动后，忙碌标志 ADBSY 被置位，ADC 选择并采样由 ADCH 指定的输入通道。转换过程中采样电平被保持。该通道转换结束后，转换结果连同转换通道编号被送入结果寄存器，并产生中断请求。转换结果保存在位域 ADRES 中。

ADST 在被硬件或软件清零之前始终保持置位。硬件清零和转换模式有关：

- 固定通道单次转换模式下，指定通道转换结束之后 ADST 被清零。
- 自动扫描单次转换模式下，通道 0 转换结束之后 ADST 被清零。

注：连续转换模式下，ADST 不能由硬件清零。

软件清零 ADST 将终止 ADC。ADC 的终止响应和转换模式有关：

- 固定通道单次转换模式下，ADC 结束当前转换之后停止运行。和 ADST 不被软件清零的操作相同。
- 固定通道连续转换模式下，ADC 结束当前转换之后停止运行。通常采取这种方式结束该转换模式。
- 自动扫描单次转换模式下，ADC 继续对自动扫描序列转换，直到通道 0 转换结束之后停止运行。和 ADST 不被软件清零的操作相同。
- 自动扫描连续转换模式下，ADC 继续对自动扫描序列转换，直到通道 0 转换结束之后停止运行。通常采取这种方式结束该转换模式。

清零之后置位 ADST 可重新启动 ADC，这将中止当前转换、用控制寄存器中设定的新参数重新启动 ADC。

转换模式选择，ADM

位域 ADM 选择 ADC 的转换模式，列于表 16-1 中。

表 16-1 ADC 转换模式

ADM	说明
00	固定通道单次转换
01	固定通道连续转换
10	自动扫描单次转换
11	自动扫描连续转换

在转换过程中，可以改变模式选择位域 ADM 和通道选择位域 ADCH。当前转换结束之后 ADM 的新值生效；当前转换结束之后（固定通道模式）或当前转换序列结束之后（自动扫描模式）ADCH 的新值生效。

转换精度控制，RES

ADC 可产生 10 位（RES = 0）或 8 位（RES = 1）的转换结果。根据应用需求可选择更高的转换速度（8 位转换所需转换时间较短）或更高的转换精度。

转换结果

转换结果存放在结果寄存器 ADC_DAT 中，插入通道的转换结果存放在寄存器 ADC_DAT2 中。

转换结果存放的位置由基本工作模式（兼容模式或增强模式）和选择的精度（8 位或 10 位）决定。

注：转换结果对应的通道编号存放在寄存器 ADC_DAT 的位域 CHNR 中；CPU 将要插入的模拟通道的编号存放在寄存器 ADC_DAT2 的位域 CHNR。

ADC_DAT

ADC 结果寄存器					SFR (FEA0 _H /50 _H)							复位值: 0000 _H			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CHNR				ADRES											
rwh				rwh											

ADC_DAT2

ADC 通道插入结果寄存器					ESFR (F0A0 _H /50 _H)							复位值: 0000 _H			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CHNR					ADRES										
rw					rwh										

符号	位序号	读写类型	功能描述
CHNR	[15:12]	rw[h]	通道编号（指明被转换的模拟通道）
ADRES	[11:0]	rwh	模数转换结果 最近一次的转换结果。 兼容模式下，结果存放位置如下： 8 位：ADRES [9:2] 10 位：ADRES [9:0] 增强模式下，结果存放位置如下： 8 位：ADRES [11:4] 10 位：ADRES [11:2] <i>注：始终将 ADRES 中的未使用位设置为 0。</i>

16.2.1 固定通道转换模式

将模式选择位域 **ADM** 设置为 **00_B** 或 **01_B**，分别对应固定通道单次转换模式或连续转换模式。**ADST** 启动 **ADC** 之后，忙碌标志 **ADBSY** 被置位，开始对位域 **ADCH** 指定的通道进行转换。转换完成后，中断请求标志 **ADCIR** 被置位。

单次转换模式，**ADC** 将自动停止，并复位 **ADBSY** 和 **ADST**。

连续转换模式，**ADC** 将自动开始新的转换（转换通道由 **ADCH** 指定）。每次转换结束后 **ADCIR** 被置位。

若在转换的过程中软件复位 **ADST**，**ADC** 将在完成当前通道转换之后，停止运行并复位 **ADBSY**。

16.2.2 自动扫描转换模式

将模式选择位域 **ADM** 设置为 **10_B** 或 **11_B**，分别对应自动扫描单次转换模式或连续转换模式。该模式自动对一个模拟通道序列进行转换（从位域 **ADCH** 指定的通道开始，到通道 **0** 结束），转换过程无需软件修改通道编号。

置位 **ADST** 启动 **ADC**，忙碌标志 **ADBSY** 被置位，首先对位域 **ADCH** 指定的通道进行转换。转换结束后，中断请求标志 **ADCIR** 被置位，接着 **ADC** 自动开始对下一个通道进行转换。每次转换结束 **ADCIR** 被置位。通道 **0** 转换之后当前转换序列结束。

单次转换模式，**ADC** 将自动终止，并复位 **ADBSY** 和 **ADST**。

连续转换模式，**ADC** 将自动开始新的转换序列（从 **ADCH** 指定的通道开始）。

若在转换的过程中软件复位 **ADST**，**ADC** 完成当前序列（包括通道 **0** 的转换）的转换之后，停止运行并复位 **ADBSY**。

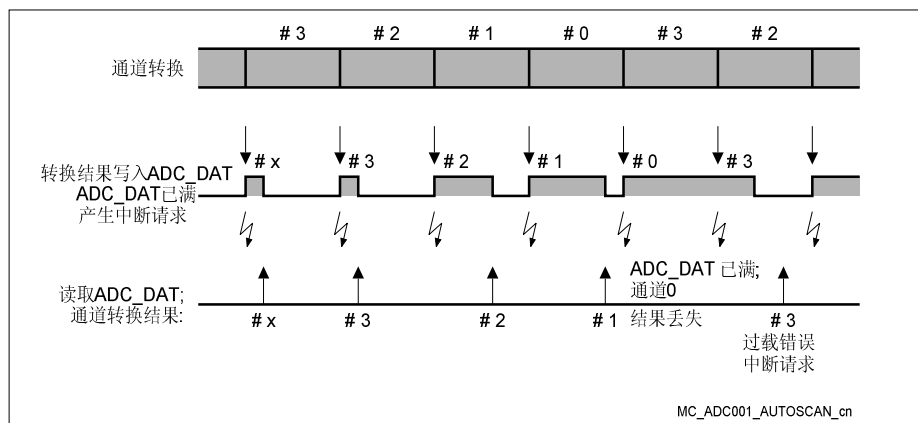


图 16-3 自动扫描转换模式举例

*注：由于通道 9 和 8 未和输入引脚连接，因此当自动扫描序列的起始通道编号大于 7 时，将产生（最多）2 个无效结果。以 **ADCH = E_H** 起始的自动扫描序列会产生如下 15 个转换结果：14, 13, 12, 11, 10, x, x, 7, 6, 5, 4, 3, 2, 1, 0。*

ADCH = 9_H 或 8_H 在转换开始时对应产生 2 个或 1 个无效结果，因此在实际应用中这类设置无意义。

16.2.3 等待读取模式

缺省模式下，若新的转换已完成而前次的转换结果还未从结果寄存器中读出，新的转换结果将覆盖前次的转换结果，前次结果丢失。ADC 的过载错误中断请求标志 ADEIR 被置位。

尤其是在连续转换模式下，为了避免产生出错中断以及转换结果丢失，可通过置位 ADWR 将 ADC 切换到“等待读取模式”。

若当前转换完成时前次结果还未读出，则将新的转换结果保存在临时缓存中，并将下次转换挂起（ADST 和 ADBSY 将保持置位状态，但不会产生“转换结束”中断）。前次结果读出后，临时缓存中的内容被复制到 ADC_DAT 中（产生 ADCIR 中断），开始执行挂起的转换。该机制对单次和连续转换模式均适用。

注：标准模式下，ADC 以固定速率（由转换时间决定）进行连续转换，“等待读取模式”下，由于转换挂起会造成延迟。然而，只有当 CPU（或 PEC）无法跟上转换速率时，才会影响转换。

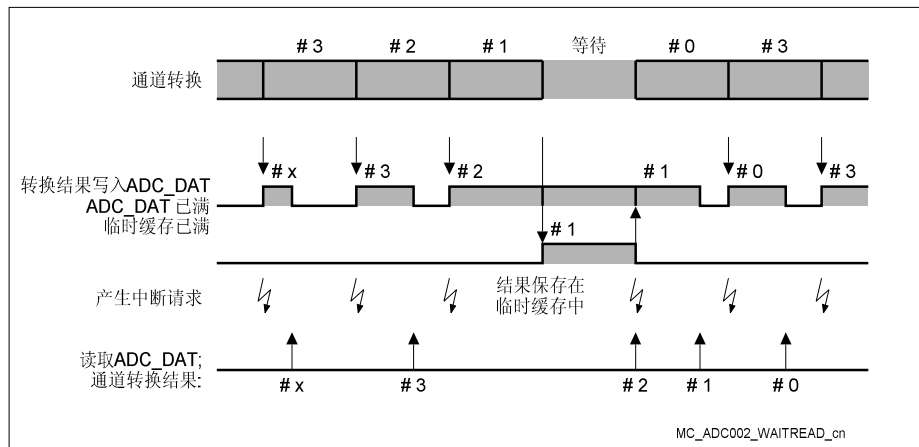


图 16-4 等待读取模式举例

16.2.4 通道插入模式

在不改变当前工作模式（ADC 正工作在连续或自动扫描模式）的情况下，通道插入模式可对插入的特定模拟通道进行转换。结束特定通道的转换之后，ADC 继续工作在原先的工作模式。

置位 **ADCIN** 并要求等待读取模式有效（**ADWR** = 1），通道插入模式被使能。该模式下寄存器 **ADC_DAT2** 中的位域 **CHNR** 指定要转换的通道编号。

*注：ADC 不修改寄存器 **ADC_DAT2** 中位域 **CHNR** 的值，只修改位域 **ADRES** 的值。由于插入转换的通道编号未经缓存，因此在插入转换的采样阶段一定不能修改 **ADC_DAT2** 中的 **CHNR**，否则输入复用器会切换到新的通道。建议只有不进行插入转换时，才改变通道编号。*

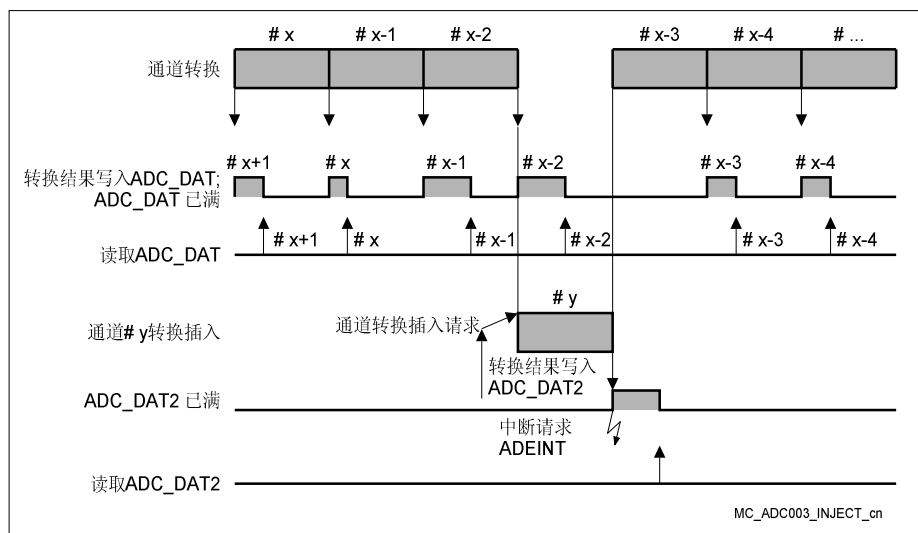


图 16-5 通道插入举例

可通过三种方式触发通道插入：

- 软件置位通道插入请求位 **ADCRQ**
- **CAPCOM2** 单元中捕获/比较寄存器 **CC31** 的比较或捕获事件置位 **ADCRQ**
- **CAPCOM6** 单元中定时器 **T13** 的周期匹配置位 **ADCRQ**

第二种方式在某个特定的时间触发通道插入：**CAPCOM2** 单元的寄存器 **CC31** 中存放着预定义的计数值，计数到该值后触发通道插入。

*注：无论通道插入模式是否被使能，任意被选中的触发方式（**CAPCOM2** 中通道 **CC31** 的中断请求或 **CAPCOM6** 中定时器 **T13** 的周期匹配）会置位通道插入请求位 **ADCRQ**。建议在使能通道插入模式之前总是对位 **ADCRQ** 清零。*

当前转换（若正在进行转换）结束之后启动（插入）指定通道的转换。完成该插入通道的转换之后，结果存放在结果寄存器 **ADC_DAT2** 中，并产生“通道插入完成”中断请求，该中断使用中断请求标志 **ADEIR**（此时要求等待读取模式有效）。

*注：插入转换的结果直接写入 **ADC_DAT2** 中。若前次的转换结果还未被读出，它将被新值覆盖。*

若临时缓存已满，则将标准转换挂起。

转换仲裁

ADC 空闲时会立即响应转换请求，触发相应的转换。若 ADC 正在进行转换时又有新的转换请求，ADC 的操作将取决于转换类型（标准或插入转换）。

注：控制位（ADST 或 ADCRQ）从 0 翻转到 1 时激活转换请求。也就是说，控制位置位之前必须为 0。

各种可能情况下的 ADC 操作归纳见 **表 16-2**。

表 16-2 转换仲裁

正在进行的转换	新转换请求	
	标准	插入
标准	取消正在进行的转换，开始新请求的转换 ¹⁾	当前转换结束之后，开始新请求的转换
插入	当前转换结束之后，开始新请求的转换	当前转换结束之后，开始新请求的转换。但第二次转换时，位 ADCRQ 将为 0

1) 在重新启动标准转换时，若有已挂起的插入转换，应先处理插入转换。

16.3 自动校准

XC164CM 的 ADC 模块具有自校准特性，主要用来纠正由工艺变化引起的增益误差和温度变化引起的偏移误差。

ADC 支持两种校准：

- 复位校准：复位后对 ADC 进行全面的基本校准，尤其在上电复位时需要进行复位校准。
- 后校准：每次转换结束后进行细微校准。

复位校准

复位后 ADC 会自动进行全面的上电校准，以纠正 ADC 的增益误差和偏移误差。为了实现最佳校准，上电校准过程中参考电压和电源电压必须保持稳定。校准过程将执行多个校准周期，其间调整步长逐渐减小。ADC 模块的实际特性决定了校准周期数。上电校准最多占用 11696 个基本时钟（ f_{BC} ）周期。

只要上电校准正在进行，状态标志 CAL 被置位。

注：必须在进入休眠模式、空闲模式或掉电模式之前完成复位校准（CAL = 0）。否则模拟比较器会保持运行，这是省电情况下所不希望的。

后校准

每次转换结束后进行细微校准。对于 8 位和 10 位转换，只要不超过数据手册中规定的总不可调整误差（TUE），并不强制执行后校准。置位寄存器 ADC_CTRL0 中的 CALOFF 将禁止后校准。若后校准被禁止，将不执行后校准，从而缩短了总的转换时间。

注：只有在完成复位校准后才能禁止后校准。

16.4 转换时间控制

启动转换之后，首先通过相应的模拟输入引脚对 AD 转换器的电容器充电，当电容器上的电压值和当前模拟输入的电压相等时充电结束，加载电容消耗的时间称作采样时间。接着将采样的电压值逐步转换为和 ADC 精度相对应的数字值。在转换过程中（不包括采样时间）通过引脚 V_{AREF} 和 V_{AGND} 对内部电容反复充放电。

因为在给定的时间内电容器必须达到其最终电压值（至少接近其最终电压），因此电容器采样和充放电所需的电流大小由转换各步骤所消耗的时间决定。不过采样源所能提供的最大电流取决于其内阻。

转换过程中的采样和转换时间可在一定范围内由软件设定。各转换步骤消耗的绝对时间（实际时间）和控制器的速度无关，从而可使 XC164CM 的 ADC 模块适用于不同特性的系统：

快速转换：将各步骤的时间设定为所允许的最小绝对值可实现快速转换，这适用于扫描高频信号，不过模拟输入和模拟电源的内阻必须足够低。

高内阻：将各步骤的时间设定成较大值、或所允许的最大值可实现高内阻，这适用于：使用具有高内阻的模拟输入和模拟电源以保持尽可能低的电流。不过这种情况下的转换速率可能非常低。

控制位域

有两种机制控制转换和采样时间：

- **标准时间控制：**使用寄存器 `ADC_CON` 中的两个 2 位位域选择转换时间和采样时间。该机制提供精简控制，但限制了预分频因子的取值范围。
- **改进型时间控制：**使用寄存器 `ADC_CON1`（兼容模式）或寄存器 `ADC_CTR2/ADC_CTR2IN`（增强模式）中的两个 6 位位域选择转换时间和采样时间。该机制下预分频因子取值的范围较宽，因此 ADC 可更灵活的适用于内部和外部系统环境。

兼容模式下置位寄存器 `ADC_CON1` 中的位 `ICST` 或者选择增强模式，将采用改进型时间控制。

标准时间控制

由寄存器 **ADC_CON** 中的两个 2 位位域进行标准时间控制。位域 **ADCTC**（转换时间控制）选择用于 **ADC** 操作的基本转换时钟（ f_{BC} ）。采样时间由位域 **ADSTC** 控制，并由上述转换时钟计算得到。采样时间始终为 $8t_{BC}$ 的整数倍。**表 16-3** 列出可能的参数组合。

表 16-3 标准转换和采样时间控制

ADC_CON.15 14 (ADCTC)	ADC 基本时钟 $f_{BC}^{1)}$	ADC_CON.13 12 (ADSTC)	采样时间 t_s
00	$f_{ADC}/4$	00	$t_{BC} \times 8$
01	$f_{ADC}/2$	01	$t_{BC} \times 16$
10	$f_{ADC}/16$	10	$t_{BC} \times 32$
11	$f_{ADC}/8$	11	$t_{BC} \times 64$

改进型时序控制

为了使时序参数的设定精度更高，用更宽的位域进行采样和转换时序控制（此时 **ADC_CON** 中的 2 位位域无效）。

兼容模式下（位 **ICST** = 1），使用寄存器 **ADC_CON1** 中的位域控制所有转换。

增强模式下（位 **MD** = 1），使用寄存器 **ADC_CTR2** 中的位域控制标准转换；使用寄存器 **ADC_CTR2IN** 中的位域控制插入转换。

位域 **ADCTC**（转换时间控制）选择用于 **ADC** 操作的基本转换时钟（ f_{BC} ）。采样时间由位域 **ADSTC** 控制，从转换时钟计算得到。采样时间始终为 $4t_{BC}$ 的整数倍。**表 16-4** 列出可能的参数组合。

表 16-4 改进型转换和采样时序控制

ADCTC	ADC 基本时钟 $f_{BC}^{1)}$	ADSTC	采样时间 t_s
$00'0000_B = 00_H$	$f_{ADC}/1$	$000000_B = 00_H$	$t_{BC} \times 8$
$00'0001_B = 01_H$	$f_{ADC}/2$	$000001_B = 01_H$	$t_{BC} \times 12$
$00'0010_B = 02_H$	$f_{ADC}/3$	$000010_B = 02_H$	$t_{BC} \times 16$
.....	$f_{ADC}/(ADCTC+1)$	$t_{BC} \times 4 \times (ADSTC+2)$
$11'1111_B = 3F_H$	$f_{ADC}/64$	$111111_B = 3F_H$	$t_{BC} \times 260$

1) 选择 **ADCTC** 和 f_{ADC} 时，一定不能超过 f_{BC} 的极限值（请参阅数据手册）。

总转换时间举例

完整的转换时间包括采样时间 t_S ，转换本身（逐次逼近和校准）和转换结果传送到结果寄存器中所需的时间。总转换时间的计算举例如下（标准转换时间）。

以模块时钟周期为基本单位计算转换时间，时钟周期 $t_{ADC} = 1/f_{ADC}$ 。

- 假设： $f_{ADC} = 40 \text{ MHz}$ （即 $t_{ADC} = 25 \text{ ns}$ ）， $ADCTC = 01_B$ ， $ADSTC = 00_B$
- 基本时钟： $f_{BC} = f_{ADC}/2 = 20 \text{ MHz}$ ，即 $t_{BC} = 50 \text{ ns}$
- 采样时间： $t_S = t_{BC} \times 8 = 400 \text{ ns}$

10 位转换：

- 有后校准： $t_{C10P} = t_S + 52 \times t_{BC} + 6 \times t_{ADC} = (2600+400+150) \text{ ns} = 3.15 \mu\text{s}$
- 无后校准： $t_{C10} = t_S + 40 \times t_{BC} + 6 \times t_{ADC} = (2000+400+150) \text{ ns} = 2.55 \mu\text{s}$

8 位转换：

- 有后校准： $t_{C8P} = t_S + 44 \times t_{BC} + 6 \times t_{ADC} = (2200+400+150) \text{ ns} = 2.75 \mu\text{s}$
- 无后校准： $t_{C8} = t_S + 32 \times t_{BC} + 6 \times t_{ADC} = (1600+400+150) \text{ ns} = 2.15 \mu\text{s}$

注：详细规范请查阅相关产品的数据手册。

16.5 ADC 中断控制

每次转换结束后，置位中断控制寄存器 **ADC_CIC** 中的中断请求标志 **ADCIR**。该“转换结束”中断请求会引发指向 **ADCINT** 的中断，或者触发 **PEC** 数据传送，将寄存器 **ADC_DAT** 中数据读出，例如保存到内部 **RAM** 的数据表中供以后使用。

若 **ADC_DAT** 中的前次结果被新的转换结果覆盖（标准模式下的出错中断），或插入转换的结果已存放在 **ADC_DAT2** 中（插入转换结束中断），置位寄存器 **ADC_EIC** 中的中断请求标志 **ADEIR**。该中断请求可引发指向 **ADEINT** 的中断，或者触发 **PEC** 数据传送。

ADC_CIC

ADC 转换中断控制寄存器																SFR (FF98H/CC _H)		复位值: - - 00 _H	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
-	-	-	-	-	-	-	GPX	ADC IR	ADC IE	ILVL				GLVL					
-	-	-	-	-	-	-	rw	rwh	rw	rw				rw					

ADC_EIC

ADC 出错中断控制寄存器																SFR (FF9AH/CD _H)		复位值: - - 00 _H	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
-	-	-	-	-	-	-	GPX	ADE IR	ADE IE	ILVL				GLVL					
-	-	-	-	-	-	-	rw	rwh	rw	rw				rw					

注：控制位域的具体解释请参阅中断控制寄存器的描述。

16.6 ADC 模块接口

ADC 以不同的方式和周围资源接口。

内部连接

CAPCOM2 单元中的捕获/比较信号 CC31IO 和 CAPCOM6 单元中的定时器 T13 周期匹配信号连接至 ADC，这两个信号用作插入转换的触发源。

ADC 的两个中断请求线送给中断控制模块。

外部连接

模拟输入信号通过 XC164CM 的 P5 口（输入单向口）送入 ADC 模块。两个专用引脚（ V_{AREF} 和 V_{AGND} ）为转换提供模拟参考电压。

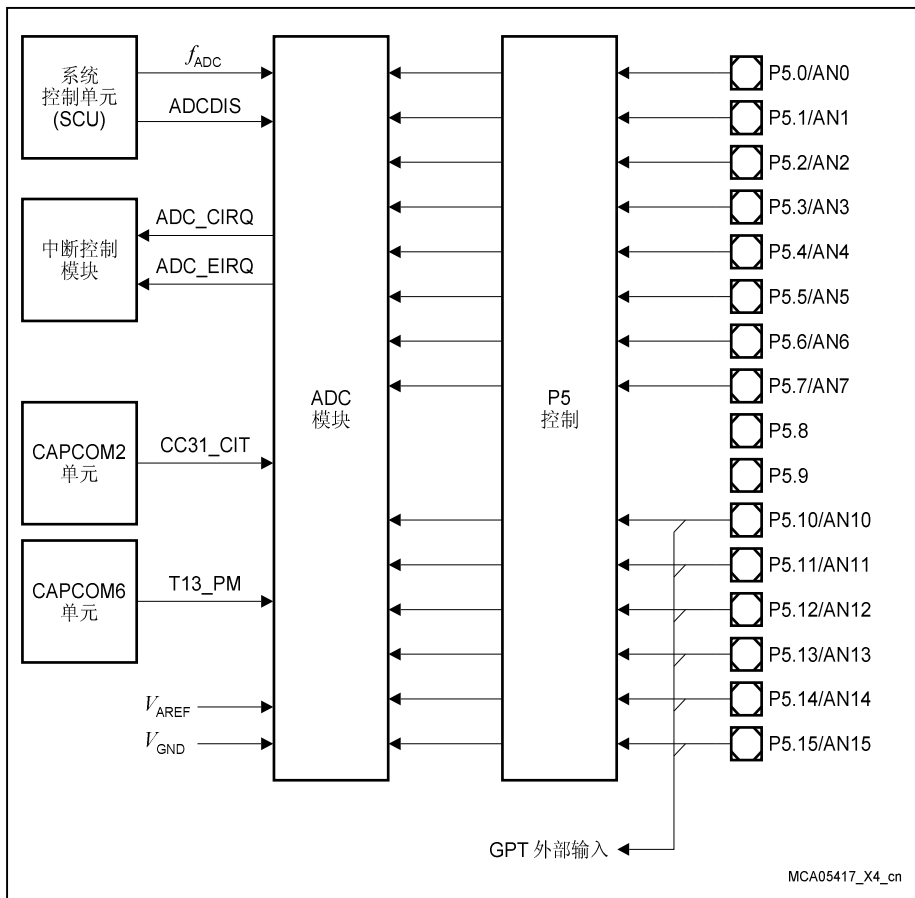


图 16-6 ADC 模块输入输出接口

17 捕获/比较单元 CAPCOM2

XC164CM 有一个捕获/比较（CAPCOM2）单元，该单元提供了和 2 个定时器紧密配合工作的 16 路捕获/比较通道。每个 CAPCOM2 通道可以由特定的内部或外部事件触发**捕获**定时器的内容；或将定时器的值和给定值进行**比较**，匹配时修改输出信号。

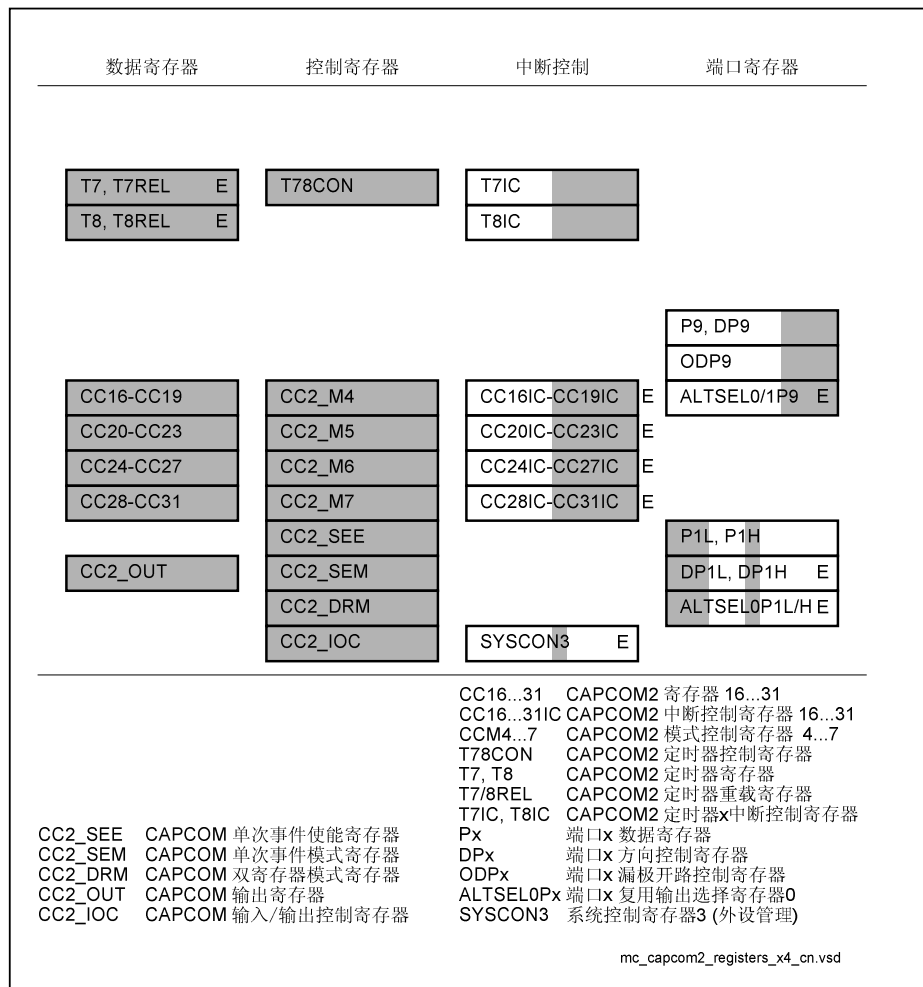


图 17-1 CAPCOM 单元的相关特殊功能寄存器（SFR）

采用该机制，只需最少的软件干预，CAPCOM2 单元就可支持多达 16 路通道上时序的产生和控制。

从编程人员的角度来看，“CAPCOM2 单元”由一组特殊功能寄存器（SFR）组成，其中包括用作捕获/比较单元复用输入/输出功能的端口引脚和端口方向控制位（见图 17-1）。

CAPCOM2 单元用于处理高速 IO 任务，如脉冲和波形产生、脉宽调制、或记录特定事件的发生时间。CAPCOM2 单元还可实现多达 16 个由软件控制的中断事件。

CAPCOM2 单元有两个 16 位定时器（T7/T8）和（一组）16 个双功能 16 位捕获/比较寄存器（CCy）。每个定时器都有相应的重载寄存器（TxREL）。

CAPCOM2 定时器的输入时钟来自经过预分频处理的模块输入时钟（ f_{cc} ），预分频因子可编程设定；或来自定时器 T6 的上溢/下溢信号。T7 也可由外部事件触发、工作在计数器模式（对外部输入计数）。

每个捕获/比较寄存器可单独被设定为捕获或比较操作，并被分配给两个定时器之一。每个捕获/比较寄存器对应一个信号，用作捕获操作的输入或比较操作的输出。

当执行捕获操作时，一旦输入信号跳变，当前定时器的值将被锁存到对应的捕获/比较寄存器中。该事件也将激活相关的中断请求线。

当执行比较操作时，一旦所分配的定时器递增计数至捕获/比较寄存器中的比较值，一个相关的输出信号将跳变。该比较匹配事件也将激活相关的中断请求线。双寄存器比较模式下，由一对寄存器共同控制一个公共的输出信号。

比较输出信号可从专用输出寄存器读取，比较输出信号也可控制相连端口引脚的输出锁存，用户可编程选择输出路径。

可选择两种时序方案（请参阅**章节 17.8**）进行输出信号切换：

交错模式¹⁾，输出信号依次分 8 步连续切换，各切换步骤占用一定时间。交错模式下，最大精度为 $8t_{cc}$ 。

非交错模式下，各输出信号同时立即切换。非交错模式下，最大精度为 $1t_{cc}$ 。

1) 交错模式和先前 16 位微控制器中的 CAPCOM2 单元兼容。

图 17-2 给出 CAPCOM2 单元的基本结构。

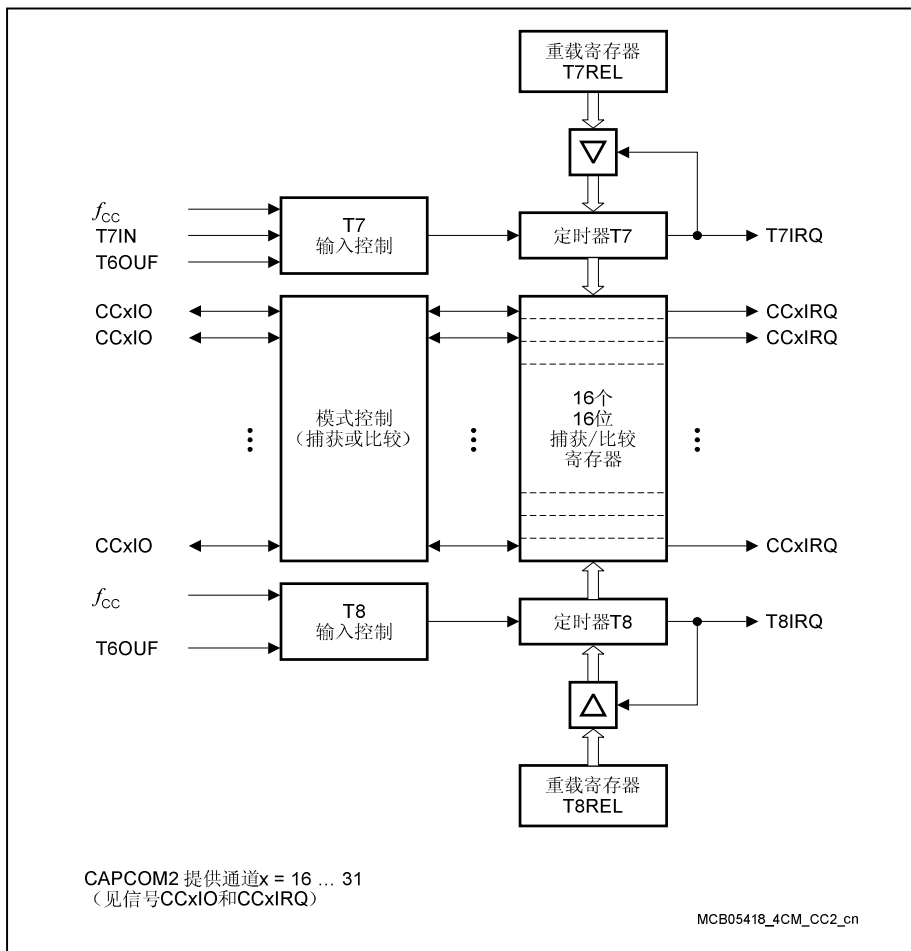


图 17-2 CAPCOM2 单元框图

17.1 CAPCOM2 定时器

定时器 T7 和 T8 主要为 CAPCOM2 的各路捕获/比较通道提供两个独立的时间基准。交错模式下最大精度为 $8t_{cc}$ ，非交错模式下最大精度为 $1t_{cc}$ 。

除输入引脚（图中虚线框标出）外，两个定时器的基本结构完全相同，如图 17-3 所示。

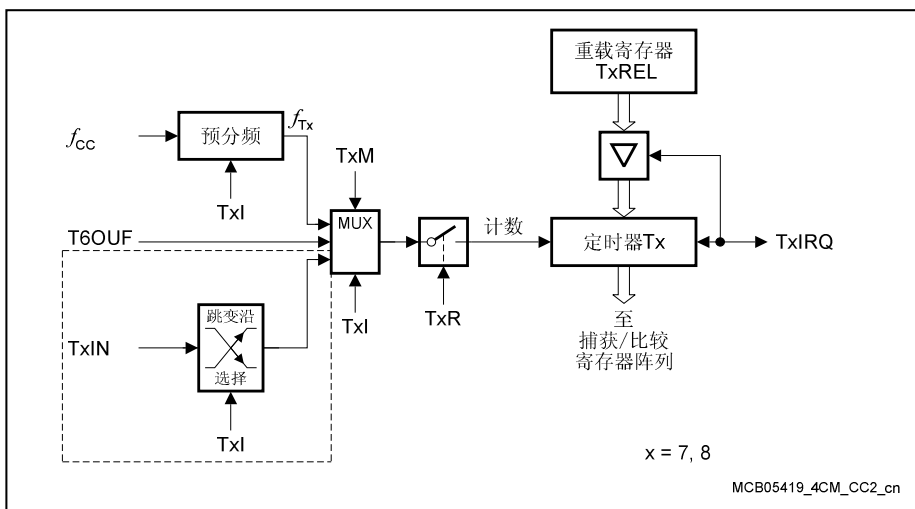


图 17-3 CAPCOM2 定时器框图

CAPCOM2 定时器的功能由可位寻址的寄存器 T78CON 控制。T78CON 的高位字节控制定时器 T8，低位字节控制定时器 T7。两个定时器的控制方式相同（除外部输入之外）。

在所有模式中，定时器始终递增计数。CPU 可从定时器寄存器 Tx 中读取当前定时器的值，Tx 不可位寻址。当 CPU 要对寄存器 Tx 进行写操作时，如果此时定时器正准备加 1 或被重载，CPU 写操作占优，加 1 或重载操作被禁止以保证定时器操作正确。

CC2_T78CON

定时器 7/8 控制寄存器

SFR (FF20_H/90_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	T8R	-	T8M	T8I	-	T7R	-	T7M	T7I						
-	rw	-	rw	rw	-	rw	-	rw	rw						

符号	位序号	读写类型	功能描述
TxR	14,6	rw	定时器/计数器 Tx 运行控制 0 定时器/计数器 Tx 被禁止 1 定时器/计数器 Tx 被使能
TxM	11,3	rw	定时器/计数器 Tx 模式选择 0 定时器模式 1 计数器模式
TxI	[10:8], [2:0]	rw	定时器/计数器 Tx 输入选择 定时器模式 (TxM = 0) : 输入频率 $f_{TX} = f_{CC} / 2^{(<TxI>+3)}$ 或 $f_{CC} / 2^{(<TxI>)}$, 分别对应交错模式和非交错模式, 见表 17-1。 计数器模式 (TxM = 1) : 000 GPT 定时器 T6 上溢/下溢 001 引脚 TxIN 的正跳变 (上升沿) 010 引脚 TxIN 的负跳变 (下降沿) 011 引脚 TxIN 的任意跳变 (上升和下降沿) 1XX 保留, 不要使用该组合! <i>注: 定时器 T8 在计数器模式下的唯一有效设置为 000_B。其它设置下 T8 不工作。</i>

定时器运行标志 **TxR** 控制定时器的启动和停止。以下定时器模式和操作的描述只适用于定时器被使能的状态, 即假定相关的运行标志已被置位。

定时器模式

定时器模式下（ $TxM = 0$ ），CAPCOM2 定时器的输入时钟来自经过预分频处理的时钟信号 f_{CC} ，预分频因子可编程设定。每个定时器有各自的预分频器，预分频的设置由相应定时器控制寄存器 T78CON 中的位域 TxI 控制。

定时器 Tx 的输入频率 f_{Tx} 和相应精度 r_{Tx} 根据以下公式计算：

交错模式：

$$f_{Tx}[MHz] = \frac{f_{CC}[MHz]}{2^{(<TxI>+3)}} \quad r_{Tx}[\mu s] = \frac{2^{(<TxI>+3)}}{f_{CC}[MHz]} \quad (17.1)$$

非交错模式：

$$f_{Tx}[MHz] = \frac{f_{CC}[MHz]}{2^{<TxI>}} \quad r_{Tx}[\mu s] = \frac{2^{<TxI>}}{f_{CC}[MHz]} \quad (17.2)$$

定时器从 FFFF_H 计数至 0000_H 时溢出，对应重载寄存器 TxREL 中的内容重新装入定时器中。重载值决定了 Tx 连续两次溢出之间的周期 P_{Tx} ：

交错模式：

$$P_{Tx}[\mu s] = \frac{(2^{16} - <TxREL>) \times 2^{(<TxI>+3)}}{f_{CC}[MHz]} \quad (17.3)$$

非交错模式：

$$P_{Tx}[\mu s] = \frac{(2^{16} - <TxREL>) \times 2^{<TxI>}}{f_{CC}[MHz]} \quad (17.4)$$

置位运行标志（TxR）启动定时器工作之后，首次加 1 操作的执行时间少于所设定的定时器精度；接下来的所有递增操作的执行时间均等于设定的定时器精度。

模块输入时钟为 40 MHz 时、不同预分频因子所对应的定时器输入频率、精度和周期值列于表 17-1 中，定时器的周期值是假设重载值为 0000_H 的情况下计算得出的。请注意某些数值经过舍入。

表 17-1 定时器模式下定时器 Tx 的输入时钟选择， $f_{CC} = 40 \text{ MHz}$

Txl	预分频因子	输入频率	精度	周期
交错模式				
000 _B	8	5 MHz	200 ns	13.11 ms
001 _B	16	2.5 MHz	400 ns	26.21 ms
010 _B	32	1.25 MHz	800 ns	52.43 ms
011 _B	64	625 kHz	1.6 μs	104.86 ms
100 _B	128	312.5 kHz	3.2 μs	209.72 ms
101 _B	256	156.25 kHz	6.4 μs	419.43 ms
110 _B	512	78.125 kHz	12.8 μs	838.86 ms
111 _B	1024	39.0625 kHz	25.6 μs	1677.72 ms
非交错模式				
000 _B	1	40 MHz	25 ns	1.6384 ms
001 _B	2	20 MHz	50 ns	3.2768 ms
010 _B	4	10 MHz	100 ns	6.5536 ms
011 _B	8	5 MHz	200 ns	13.11 ms
100 _B	16	2.5 MHz	400 ns	26.21 ms
101 _B	32	1.25 MHz	800 ns	52.43 ms
110 _B	64	625 kHz	1.6 μs	104.86 ms
111 _B	128	312.5 kHz	3.2 μs	209.72 ms

计数器模式

计数器模式下（ $TxM = 1$ ），CAPCOM2 定时器的输入时钟来自外部输入引脚 T7IN，或者 GPT 定时器 T6 的上溢/下溢信号。

用连接到 TxIN 引脚上的外部信号作为计数时钟只适用于定时器 T7。计数器模式下，定时器 T8 只能使用 GPT 定时器 T6 的上溢/下溢信号作为计数时钟（ $TxI = 000_B$ ）。

位域 T7I 选择引脚 T7IN 上外部信号的正跳变、负跳变或任意跳变触发定时器 T7 递增计数。计数器模式下，外部信号和端口引脚的设定必须满足某些条件以确保操作正确，具体内容请参阅[章节 17.10](#)。

定时器溢出和重载

若 CAPCOM2 定时器的计数值为 $FFFF_H$ 时发生新的计数触发事件，将产生定时器中断请求，并将重载寄存器 TxREL 的值重新装入定时器中。发生下次计数事件时，定时器从重载值开始递增计数。

重载寄存器 TxREL 不可位寻址，复位值为 0000_H 。

17.2 CAPCOM2 定时器中断

定时器溢出时，相应的定时器中断请求标志 **TxIR** 被置位。若中断使能位 **TxE** 被置位，该中断请求标志可用于产生中断或触发 **PEC** 服务请求。

每个定时器各有相应的可位寻址的中断控制寄存器和中断向量。中断控制寄存器 **TxIC** 和其它中断控制寄存器的结构相同。

CC2_T7IC

CAPCOM T7 中断控制寄存器 **ESFR (F17A_H/BE_H)** 复位值: - - 00_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	GPX	T7IR	T7IE	ILVL			GLVL		
-	-	-	-	-	-	-	rw	rwh	rw	rw			rw		

CC2_T8IC

CAPCOM T8 中断控制寄存器 **ESFR (F17C_H/BF_H)** 复位值: - - 00_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	GPX	T8IR	T8IE	ILVL			GLVL		
-	-	-	-	-	-	-	rw	rwh	rw	rw			rw		

注：控制位域的具体解释请参阅通用中断控制寄存器的描述。

17.3 捕获/比较通道

16 位捕获/比较寄存器 CC16 到 CC31 用作定时器 T7 和 T8 捕获或比较操作的数据寄存器。捕获/比较寄存器不可位寻址。

CAPCOM2 单元中的 16 个捕获/比较寄存器由 4 个结构完全相同的可位寻址的 16 位模式控制寄存器 CC2_M4...CC2_M7 控制（详见以下寄存器描述）。每个模式控制寄存器分别控制四个捕获/比较寄存器的模式选择和定时器分配。

CAPCOM2 单元（CC31...CC16）的捕获/比较寄存器

CC2_M4

CAPCOM 模式控制寄存器 4 **SFR (FF22H/91H)** **复位值: 0000H**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACC 19	MOD19			ACC 18	MOD18			ACC 17	MOD17			ACC 16	MOD16		
rw	rw			rw	rw			rw	rw			rw	rw		

CC2_M5

CAPCOM 模式控制寄存器 5 **SFR (FF24H/92H)** **复位值: 0000H**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACC 23	MOD23			ACC 22	MOD22			ACC 21	MOD21			ACC 20	MOD20		
rw	rw			rw	rw			rw	rw			rw	rw		

CC2_M6

CAPCOM 模式控制寄存器 6 **SFR (FF26H/93H)** **复位值: 0000H**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACC 27	MOD27			ACC 26	MOD26			ACC 25	MOD25			ACC 24	MOD24		
rw	rw			rw	rw			rw	rw			rw	rw		

CC2_M7

CAPCOM 模式控制寄存器 3

SFR (FF28_H/94_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ACC 31	MOD31			ACC 30	MOD30			ACC 29	MOD29			ACC 28	MOD28		
rw		rw		rw		rw		rw		rw		rw		rw	

符号	位序号	读写类型	功能描述
ACCy	15,11, 7, 3	rw	CAPCOM 寄存器 CCy 的定时器分配位 0 指定 CCy 和定时器 T7 配合工作 1 指定 CCy 和定时器 T8 配合工作
MODy	[14:12], [10:8], [6:4], [2:0]	rw	CAPCOM 寄存器 CCy 的模式选择 见表 17-2。

每个寄存器 CCy 可被设定为捕获模式或四种比较模式之一，并被分配给 CAPCOM2 单元的定时器之一（T7/T8）。特殊的双寄存器比较模式将两个寄存器组合起来控制一个公共的输出信号。某个 CCy 寄存器的捕获或比较操作被禁止时，该寄存器可用于保存通用变量。

表 17-2 捕获模式和比较模式的选择

模式	MODy	选中的工作模式
禁用	0 0 0	捕获和比较模式禁用 CAPCOM 寄存器可存储通用变量
捕获	0 0 1	引脚 CCyIO 上的正跳变（上升沿）触发捕获操作
	0 1 0	引脚 CCyIO 上的负跳变（下降沿）触发捕获操作
	0 1 1	引脚 CCyIO 上的任意跳变（上升沿和下降沿）触发捕获操作
比较	1 0 0	比较模式 0：只产生中断 每个定时器周期产生多个中断 可使能 Bank2 寄存器的双寄存器比较模式

模式	MODy	选中的工作模式
	1 0 1	比较模式 1：每次匹配时翻转输出引脚 每个定时器周期产生多个比较事件 可使能 Bank1 寄存器的双寄存器比较模式
	1 1 0	比较模式 2：只产生中断 每个定时器周期只产生一个中断
	1 1 1	比较模式 3：每次匹配时置位输出引脚 每次定时器溢出时复位输出引脚； 每个定时器周期只产生一个中断

捕获和比较模式的详细描述对所有捕获/比较通道均有效，因此各寄存器、各位和各引脚用符号 x 加以区分。

注：若通道未和引脚相连，只有比较模式 0 和 2 有意义。通道 31 的捕获或比较事件可用来触发 XC164CM 模数转换器的通道插入（若该模式被使能）。

17.4 捕获模式操作

捕获模式下，一旦外部事件触发，CAPCOM2 定时器的当前值将被锁存（捕获）到相应的捕获/比较寄存器中。该模式可用于记录外部事件的发生时刻、或测量两个外部事件的时间间隔（用定时器的增量表示）。

可设定（和输入引脚连接的）外部信号的正跳变、负跳变或任意跳变作为捕获触发事件，通过对应的模式控制寄存器中的位域 **MODy** 选择。所选择的外部信号跳变发生时，定时器的内容被锁存到捕获/比较寄存器中，相应的中断请求线 **CCyIRQ** 被激活。若中断被使能，该中断请求会引发中断或 **PEC** 服务请求。

注：捕获输入可用作附加外部中断输入，此时捕获操作不可用。

模式控制寄存器中的定时器分配控制位 **ACCy** 控制选择捕获定时器 **T7** 或 **T8** 的内容。

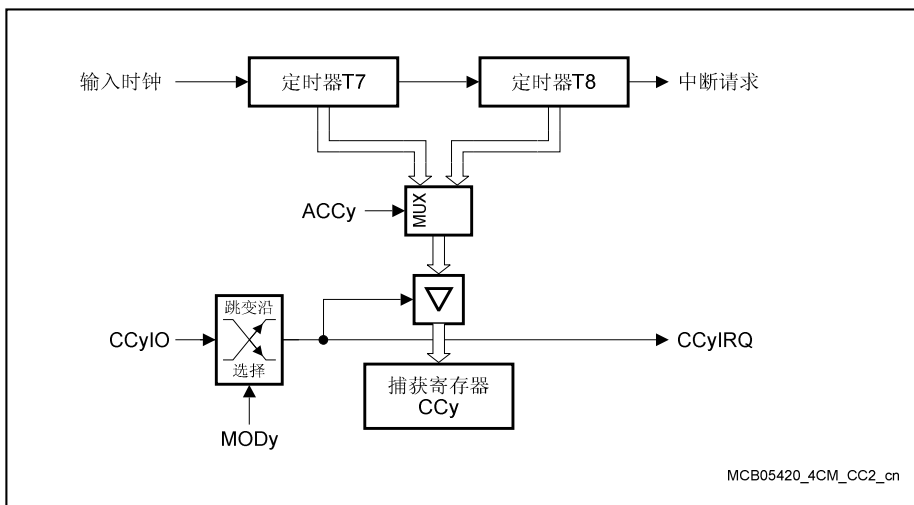


图 17-4 捕获模式框图

必须将捕获操作的相关引脚设定为输入。为了确保能够正确识别输入信号的跳变，输入电平必须至少保持（高或低）规定数目的模块时钟周期之后才能改变。具体内容请参阅[章节 17.10](#)。

17.5 比较模式操作

比较模式可在最低的软件开销下，产生触发事件（中断和/或输出信号跳变）或脉冲序列。所有比较模式中，将保存在捕获/比较寄存器 **CCy** 中的 16 位数值（以下描述中也称为“比较值”）和所分配的定时器（T7 或 T8）的计数值进行连续比较。若定时器的当前值和比较值相匹配，与寄存器 **CCy** 相关的中断请求线被激活，是否在对应的输出引脚 **CCyIO** 上产生输出信号取决于所选择的比较模式。

CAPCOM2 支持四种不同的比较模式。通过模式控制寄存器中的位域 **MODy** 可分别为每个捕获/比较寄存器选择对应的比较模式。模式 0 和模式 2 不影响输出信号。下面将对每种模式进行详细说明。

除这些“单寄存器”模式之外，**CAPCOM2** 还支持“双寄存器”比较模式，该模式下允许两个寄存器共同控制一个输出引脚。由于可设定两个不同的比较值控制同一个信号的跳变序列，“双寄存器”比较模式进一步降低了软件开销。具体内容请参阅[章节 17.5.5](#)。

所有比较模式中，比较器执行“等于”比较。也就是说，只有当定时器的计数值和比较寄存器的值相等时才检测到匹配。此外，只有在定时器硬件执行加 1 操作后紧接着的一个时钟周期内，比较器才被使能，目的在于防止定时器不以最高输入时钟工作时

（定时器模式或计数器模式）出现重复匹配，此时定时器中的一个值会保持多个或多达上千个时钟周期。该操作的副作用在于，软件修改定时器的值无法获得比较匹配。若软件设定定时器的值和某个比较寄存器的值相同，不会检测到匹配。比较寄存器的设定值小于定时器的当前计数值时，不执行任何操作。

端口输出功能的具体操作请参阅[章节 17.6](#)。

若两个或多个比较寄存器设定的比较值相同¹⁾，对应定时器递增计数到该比较值后，对应的各中断请求标志被置位，并产生预先选择的输出信号。定时器再次递增计数或者被软件写入之前，同一个比较值的其它比较事件被禁止²⁾。复位后，只有当定时器开始递增计数、或已由软件写入，并为捕获/比较寄存器 **CCy** 选择了对应的比较模式，**CCy** 的比较事件才被使能。

1) 交错模式下，这些中断和输出信号顺序产生（请参阅[章节 17.8](#)）。

2) 定时器执行加 1 操作之前（定时器工作在较低频率），即使经过多个比较周期，给定的一个比较值只产生一个比较事件。

17.5.1 比较模式 0

比较模式 0 只产生中断，可用作软件定时。该模式下，每次检测到比较寄存器 CCy 中的比较值和对应该定时器的计数值匹配时，激活中断请求线 CCyIRQ。“匹配”意味着定时器的内容和比较寄存器的值相等（“=”）。若在一个定时器周期内更新寄存器 CCy 中的比较值，其间可能产生多个比较事件。该模式下，比较事件不会影响对应的端口信号 CCyIO，因此该端口可用作通用 IO。

注：若 bank2 中的某寄存器设定为比较模式 0，该寄存器的双寄存器比较模式可被使能（请参阅[章节 17.5.5](#)）。

17.5.2 比较模式 1

比较模式 1 和比较模式 0 的基本操作相同，差别仅在于该模式会影响相关的输出信号。该模式下，每次检测到比较寄存器 CCy 中的比较值和对应该定时器的计数值匹配时，激活中断请求线 CCyIRQ。同时，相关输出信号翻转。若在一个定时器周期内更新寄存器 CCy 中的比较值，其间可能产生多个比较事件。

注：若 bank1 中的某寄存器设定为比较模式 1，该寄存器的双寄存器比较模式可被使能（请参阅[章节 17.5.5](#)）。

端口输出信号的具体操作请参阅[章节 17.6](#)。

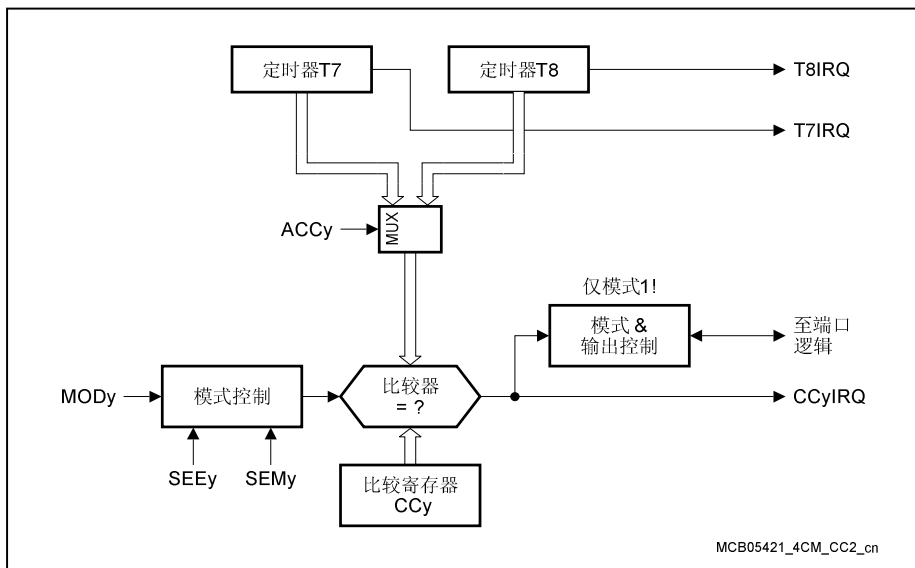


图 17-5 比较模式 0 和 1 框图

注：比较模式 0 输出信号不受影响。

图 17-6 举例说明比较模式 0 和 1 的不同操作情况。

各例中定时器的重载值设定为 $FFF9_H$ 。定时器溢出时，从该值重新开始递增计数。

例 1 中，寄存器 CCy 的值为 $FFFC_H$ 。定时器递增计数到该值时检测到匹配，中断请求线 $CCyIRQ$ 被激活。比较模式 0 的所有操作完毕；比较模式 1，相关的端口输出被翻转，引发输出信号翻转。若寄存器 CCy 的值不改变，定时器每次计数到设定的比较值时执行该操作。

例 2 中，第一次比较值 $FFFC_H$ 匹配之后，软件将 $FFFF_H$ 重新装入比较寄存器 CCy 中。定时器继续递增计数，计数到新的比较值并检测到新匹配后，激活中断请求线（比较模式 0 和 1）并翻转输出信号（比较模式 1）。若比较值不再改变，定时器再次计数到 $FFFF_H$ 时发生匹配。

该例说明在当前的定时器周期内可能出现多次比较匹配（和比较模式 2 和 3 作对比）。

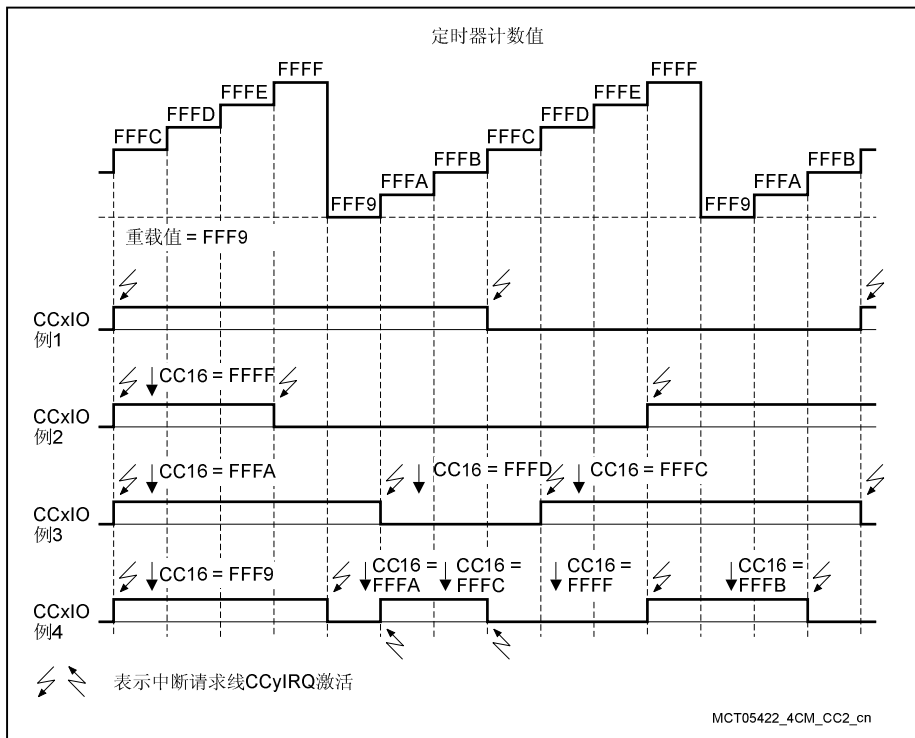


图 17-6 比较模式 0 和 1 时序举例

例 3 中，在当前定时器周期内，只有大于定时器当前计数值的新比较值才会引发新匹配。首次匹配（ FFFC_H ）之后将 FFFA_H 重新装入比较寄存器中，由于定时器计数已超过新比较值，因此直到下个定时器周期，定时器计数到 FFFA_H 时才引发比较匹配。此时将大于定时器当前值的新比较值重新装入寄存器 CCy 中，在本周期内将引发新匹配。

例 4 中，比较值等于定时器的重载值、或者等于最大计数值 FFFF_H 。

17.5.3 比较模式 2

比较模式 2 和比较模式 0 类似，只产生中断。主要区别是在一个给定的定时器周期内，比较模式 2 只发生一次比较匹配，产生一次中断请求。

该模式下，在一个定时器计数周期内首次检测到匹配时，激活中断请求线 CCyIRQ。此外，即使在该周期内将大于定时器当前值的比较值写入比较寄存器中，所有其它的比较匹配被禁止。只有当定时器溢出时才解禁。首次匹配之后写入比较寄存器中的新值只能在下一个定时器周期内生效。

17.5.4 比较模式 3

比较模式 3 的基本操作和比较模式 2 相同，差别仅在于该模式会影响相关的输出引脚。在一个定时器周期内只可能产生一个比较事件。

该模式下，在一个定时器计数周期内首次检测到匹配时，激活中断请求线 CCyIRQ，并将相关的输出信号置 1。此外，即使在该周期内将大于定时器当前值的比较值写入寄存器中，所有其它的比较匹配被禁止。只有当定时器溢出时才解禁。首次匹配之后写入比较寄存器中的新值只在下一个定时器周期内生效。

溢出信号还可将相关的输出信号复位为 0。

必须注意：当所设定的比较值和定时器重载值相等时，比较匹配试图置位输出信号，同时定时器溢出试图复位输出信号。必须避免冲突，这种情况下输出信号的状态保持不变。

注：将比较值从大于定时器的当前值修改为小于定时器的当前值时，新值只有在下一个定时器周期才能被识别。

端口输出信号的具体操作请参阅[章节 17.6](#)。

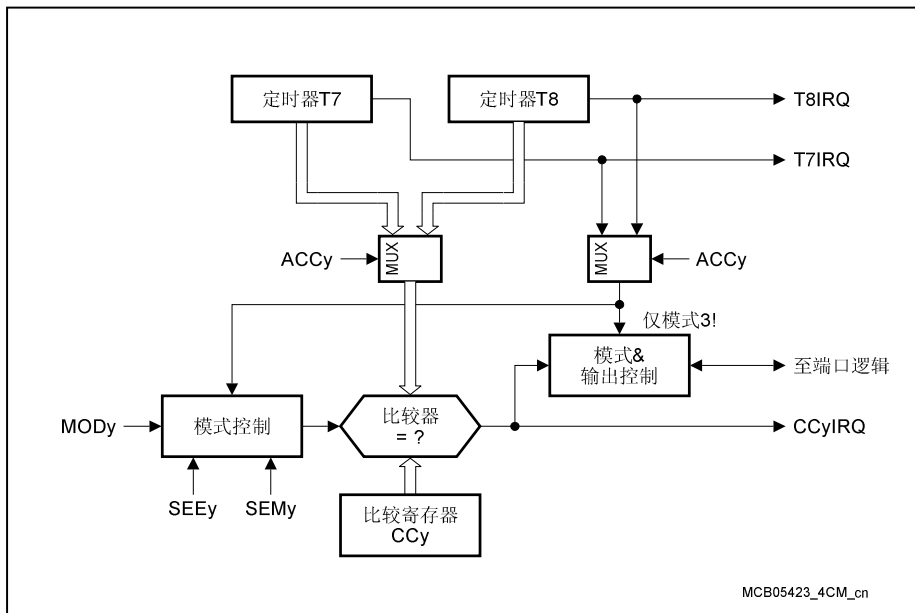


图 17-7 比较模式 2 和 3 框图

注：比较模式 2 的端口锁存和输出信号不受影响。

图 17-8 给出比较模式 2 和 3 的时序举例。

各例中定时器的重载值设定为 $FFF9_H$ 。每当定时器溢出，从该值重新开始递增计数。

例 1 中，寄存器 CCy 的值为 $FFFC_H$ 。定时器递增计数到该值时检测到匹配，中断请求线 $CCyIRQ$ 被激活。比较模式 2 的所有操作完毕；而在比较模式 3 下，相关的端口输出被置 1。定时器继续递增计数直至溢出。溢出时端口输出复位为 0。请注意，尽管图中未标出，定时器的溢出信号也激活相关的中断请求线 $TxIRQ$ 。若寄存器 CCy 中的比较值不改变，端口输出在下一个定时器周期内再次置位，定时器溢出时再次复位。这是产生脉宽调制（PWM）信号的理想操作，此时的软件开销最低。改变比较值相应改变脉冲宽度。

例 2 中，在一个定时器周期内首次匹配之后比较操作被禁止。首次 $FFFC_H$ 匹配之后，产生中断请求并置位端口输出。此外其它匹配被禁止。若此时将新比较值（尽管该值大于定时器的当前计数值）写入寄存器 CCy ，不会产生中断请求、端口输出不受影响。只有在定时器溢出后，比较逻辑再次被使能，在 $FFFF_H$ 处将检测到下次匹配。可见，由于软件可写入任意的比较值（大于或小于定时器的当前值），这是产生 PWM 的

理想操作。该机制确保了新比较值（通常在恰当的中断服务程序中写入比较寄存器）只会在下个定时器周期内生效。

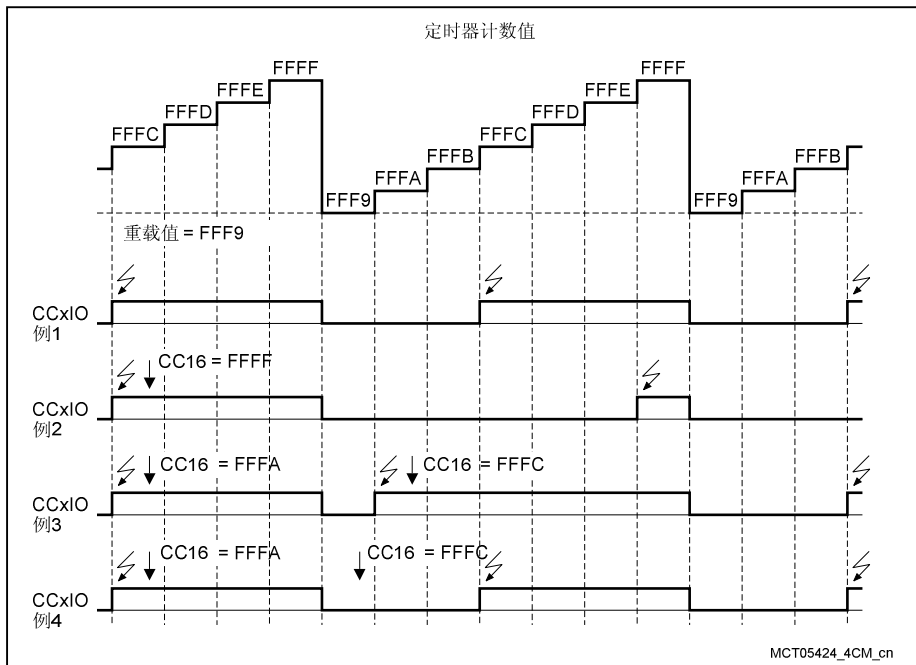


图 17-8 比较模式 2 和 3 时序举例

注：比较模式 2 只产生中断；模式 3 还会产生输出信号。

例 3 中，进一步说明比较匹配操作被禁止。

例 4 中，在一个定时器周期内，首次匹配之前将新比较值写入比较寄存器，显然原先设定的比较匹配（FFFA_H处）不会发生，将在 FFFC_H处检测到首次匹配。但必须注意，重新设定比较寄存器的操作是异步的，即寄存器写操作的时刻和定时器的当前值无关。该操作具有一定风险，因为其结果不易被预估。如果软件将新比较值写入寄存器的时刻，定时器已计数到原先设定的比较值 FFFA_H，那么在 FFFA_H处会检测到匹配，新的设定值只能在下个定时器周期生效。

图 17-9 给出比较模式 2 和 3 的特例。例 1 为比较值和定时器的重载值相等的情况。两种模式均产生中断。模式 3 输出信号不受影响，始终保持高电平。将比较值设置成定时器的重载值，就能产生占空比为 100% 的 PWM 信号。该方法的主要优点在于仍产生比较中断并可用来重载下次的比较值，因此无需特殊处理（见例 3）。

特例 2、4 和 5 中以不同的方式产生占空比为 0% 的 PWM 信号。

在特例 2 中，异步重新设定比较值，其值和重载值相等。当前定时器周期结束时产生比较中断，该中断控制软件设置下次的比较值。该方法的缺点在于新的常规比较值至少在两个定时器周期之后才可生效。和重载值 FFF9_H 的比较匹配将阻止当前定时器周期内的其它比较匹配。特例 4 对此作了进一步说明。

特例 5 就上述问题（新的常规比较值至少在两个定时器周期之后才可生效）给出一种解决方案。比较寄存器中重新装入小于定时器重载值的比较值 FFF8_H。定时器永远不会计数到该值，因此不会检测到匹配。首次定时器溢出后输出信号被复位为 0。第二次溢出之后，软件将一个常规比较值重新装入比较寄存器。由于以前未发生比较阻滞（由于没有比较匹配），新写入的比较值将在当前定时器周期内生效。

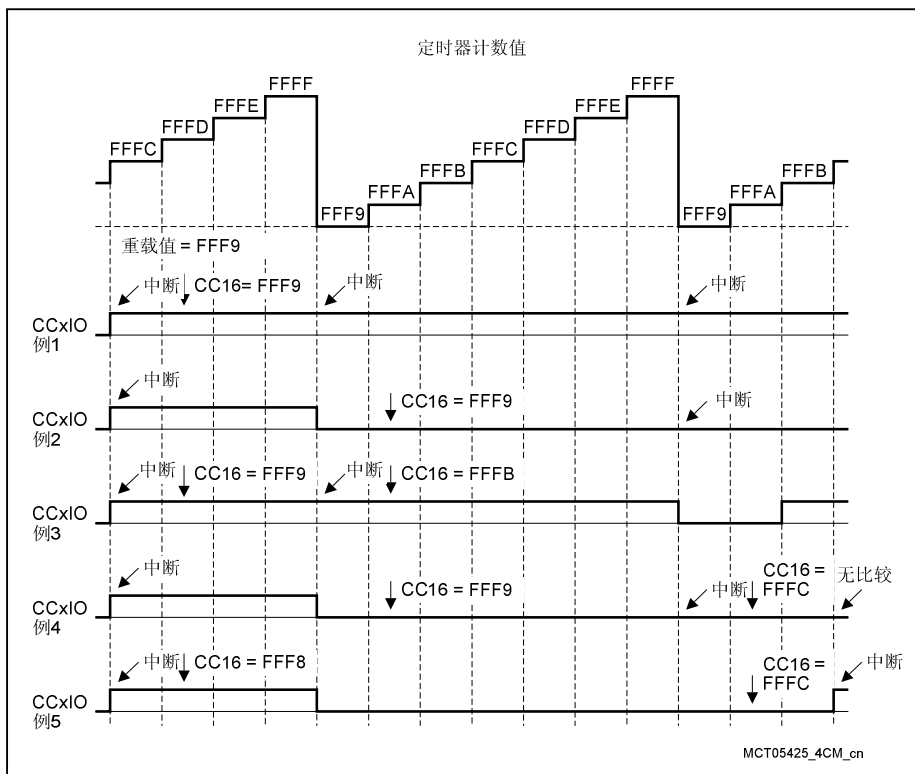


图 17-9 比较模式 2 和 3 的特例

17.5.5 双寄存器比较模式

在某些应用中，双寄存器比较模式可进一步降低软件开销。该模式下，两个比较寄存器共同控制一个输出。可通过 **DRM** 寄存器、或两个寄存器比较模式的特殊组合选择该模式。

双寄存器比较模式下，**CAPCOM2** 单元中的 16 个捕获/比较寄存器被看作两组寄存器，每组由 8 个寄存器组成。低 8 个寄存器构成 **bank1**，高 8 个寄存器构成 **bank2**。双寄存器模式下，**bank1** 的寄存器和 **bank2** 的寄存器一一构成寄存器对。寄存器对中的两个寄存器共同控制和 **bank1** 寄存器相关的输出引脚。

一个寄存器对与 **bank1** 和 **bank2** 寄存器的对应关系和它们所控制的双寄存器比较模式的输出引脚列于表 17-3 中。

表 17-3 双寄存器比较模式下的寄存器对

CAPCOM2 单元			
寄存器对		使用的输出引脚	CC2DRM 中的控制位域
Bank1	Bank2		
CC16	CC24	CC16IO	DR0M
CC17	CC25	CC17IO	DR1M
CC18	CC26	CC18IO	DR2M
CC19	CC27	CC19IO	DR3M
CC20	CC28	CC20IO	DR4M
CC21	CC29	CC21IO	DR5M
CC22	CC30	CC22IO	DR6M
CC23	CC31	CC23IO	DR7M

可分别将每对寄存器设定为双寄存器比较模式。某寄存器对中的两个寄存器的比较模式为某种特定组合时，可选择双寄存器比较模式：**Bank1** 的寄存器必须设定为模式 1（影响端口输出）；**bank2** 的寄存器必须设定为模式 0（只产生中断）。

可通过寄存器 **CC2_DRM** 中的位域 **DRxM** 分别控制（使能或禁止）每对寄存器的双寄存器比较模式。

CC2_DRM

双寄存器比较模式寄存器

SFR (FF2A_H/95_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DR7M	DR6M	DR5M	DR4M	DR3M	DR2M	DR1M	DR0M								
rw		rw		rw		rw		rw		rw		rw		rw	

符号	位序号	读写类型	功能描述
DRxM	[1:0], [3:2], [5:4], [7:6], [9:8], [11:10], [13:12], [15:14]	rw	双寄存器 x 比较模式选择 00 通过比较模式 1 和 0 的组合控制 DRM （兼容模式） 01 禁止 DRM 操作，和比较模式无关 10 使能 DRM 操作，和比较模式无关 11 保留 <i>注：“x”表示 bank 中寄存器对的编号。</i>

可分别控制每对寄存器的双寄存器比较模式。

双寄存器比较模式的框图中（图 17-10），bank2 的寄存器用 CCz 表示，bank1 的寄存器用 CCy 表示。

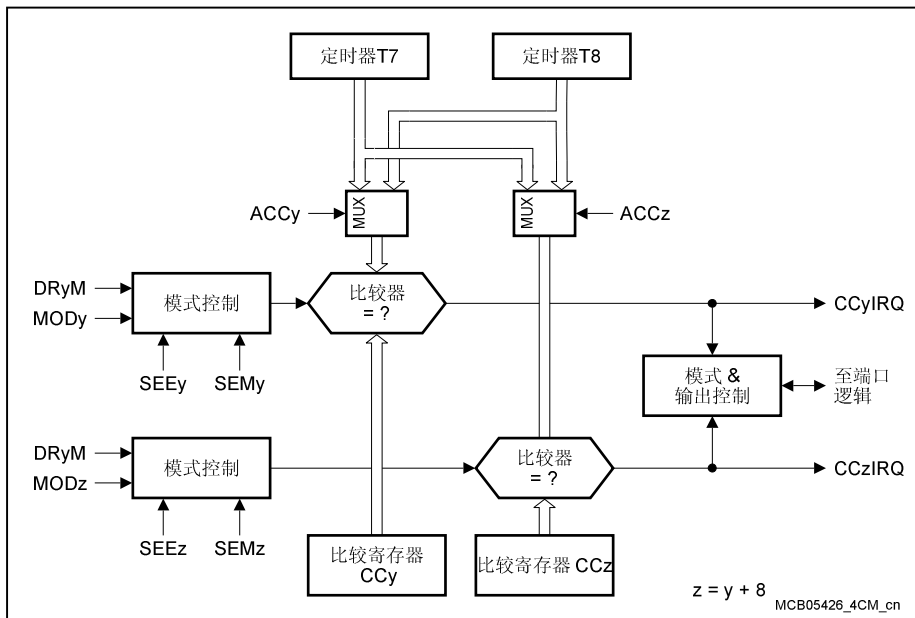


图 17-10 双寄存器比较模式框图

当检测到寄存器对（CCy 或 CCz）中的任意一个寄存器发生匹配时，相关中断请求线（CCyIRQ 或 CCzIRQ）被激活，和 bank1 的寄存器 CCy 对应的引脚 CCyIO 翻转。产生的中断始终和引起匹配的寄存器相对应。

注：若寄存器对中的两个寄存器 CCy 和 CCz 同时发生匹配，引脚 CCyIO 只翻转一次，不过会产生两个独立的比较中断请求。

每个寄存器对中的两个寄存器可分别被指定和 CAPCOM2 单元两个定时器中的任意一个配合作。由于两个定时器能以不同的精度和频率工作在不同的模式下，这种灵活的分配方式提供了广泛的应用。不过，这可能需要复杂软件算法来处理不同的定时器周期。

注：信号 CCzIO（双寄存器比较模式中不使用该信号）可用作通用 IO。

17.6 比较输出信号的产生

本节讨论 CAPCOM2 单元和端口逻辑之间的相互配合。图 17-11 详细描述了图 17-5、图 17-7 和图 17-10 中的模块“模式 & 输出控制”的逻辑结构。

每个输出信号被锁存到输出锁存寄存器 CCx_OUT 中的相关位。每次产生比较事件时对应位被更新。这些输出寄存器中的各位和相应的端口引脚相连，作为该端口的复用输出功能。

比较信号还可以直接影响相关的端口输出锁存 Px。此时必须选择引脚信号来自端口锁存（这种情况下，相应的引脚必须选择端口锁存功能）。在非交错模式下、或置位寄存器 CCx_IOC 中的位 PL 将禁用直接端口锁存功能。

寄存器 CCx_OUT 始终和端口输出锁存同步更新。

CC2_OUT

比较输出寄存器

SFR (FF2C_H/96_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC15 IO	CC14 IO	CC13 IO	CC12 IO	CC11 IO	CC10 IO	CC9 IO	CC8 IO	CC7 IO	CC6 IO	CC5 IO	CC4 IO	CC3 IO	CC2 IO	CC1 IO	CC0 IO
rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh

符号	位序号	读写类型	功能描述
CCyIO	15...0	rwh	通道 y 比较输出 相关端口引脚的复用输出功能

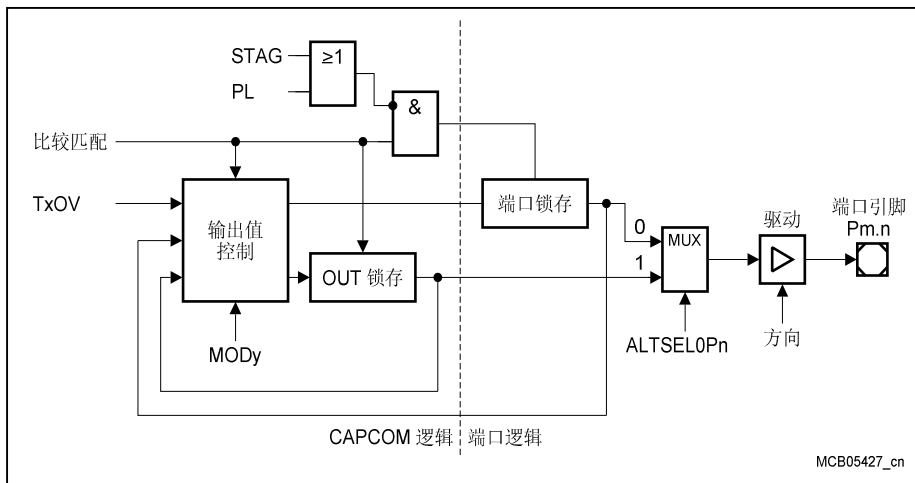


图 17-11 比较模式的端口输出框图

注:只有在模式 1 或模式 3, 比较输出信号才送到引脚上。

比较事件的输出信号可以为 1、0、当前电平反相或者原电平。“输出值控制”模块根据比较事件、定时器溢出信号以及端口和 OUT 锁存的当前状态确定正确的新电平。如果输出需要翻转（如在比较模式 1 中），输出锁存的状态被读出、取反、然后写回。

寄存器 ALTSEL 可选择输出引脚的信号来源：端口锁存信号或者 OUT 信号（见图 17-11）。

注：若软件和比较事件同时修改端口输出锁存，软件修改占优。此时硬件触发的修改无效。

17.7 单次事件操作

若某应用只需要一次比较事件（在某个时间段内），单次事件操作有助于降低软件开销，无需软件在匹配事件后作出快速响应。

若没有该特性，要实现单次事件操作时，在设定的比较匹配发生后，只能通过软件禁止比较模式；或者将定时器计数范围之外的比较值写入比较寄存器。因此，通常需要中断服务程序来执行该操作。若定时器周期非常短，则对中断响应速度要求严格——在定时器再次计数到相同的比较值之前要完成禁止操作。

单次事件操作不需要软件在首次比较匹配之后作出响应。可以在比较事件之前设置好所有的操作，比较事件发生后无需任何操作。硬件负责产生唯一的比较事件，并禁止其后所有的比较匹配。

通过单次事件模式寄存器 **CC2_SEM** 和单次事件使能寄存器 **CC2_SEE** 设定该功能。每个寄存器中的每一位和 **CAPCOM2** 单元中的每个 **CCy** 寄存器一一对应。

CC2_SEM

单次事件模式控制寄存器

SFR (FE2C_H/16_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEM 15	SEM 14	SEM 13	SEM 12	SEM 11	SEM 10	SEM 9	SEM 8	SEM 7	SEM 6	SEM 5	SEM 4	SEM 3	SEM 2	SEM 1	SEM 0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

符号	位序号	读写类型	功能描述
SEMy	15...0	rw	单次事件模式控制 0 通道 y 的单次事件模式被禁止 1 通道 y 的单次事件模式被使能

CC2_SEE

单次事件使能寄存器

SFR (FE2A_H/15_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEE 15	SEE 14	SEE 13	SEE 12	SEE 11	SEE 10	SEE 9	SEE 8	SEE 7	SEE 6	SEE 5	SEE 4	SEE 3	SEE 2	SEE 1	SEE 0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

符号	位序号	读写类型	功能描述
SEEy	15...0	rwh	单次事件使能控制 0 通道 y 的单次事件被禁止 1 通道 y 的单次事件被使能 <i>注：事件发生后由硬件对该位清零。</i>

将 CCy 寄存器设置为单次事件操作：软件首先设定需要的比较操作和比较值，然后置位寄存器 CC2_SEM 中的相应位以使能单次事件模式。最后，置位寄存器 CC2_SEE 中相应的单次事件使能位。

设定的比较匹配发生后，执行比较模式的所有相应操作。此外，硬件自动禁止其后所有比较匹配并复位寄存器 CC2_SEE 中的使能位。只要该使能位被清零，任何比较操作被禁止。设置新事件时，必须首先将该位重新置 1。

17.8 交错和非交错操作

CAPCOM2 单元可工作在两种基本模式：交错模式和非交错模式。通过寄存器 IOC 进行模式选择。

CC2_IOC

I/O 控制寄存器

ESFR (F066_H/33_H)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-												-	ST AG	PL	-
-												-	rw	rw	-

符号	位序号	读写类型	功能描述
STAG	2	rw	交错模式控制 0 CAPCOM 工作在交错模式 1 CAPCOM 工作在非交错模式
PL	1	rw	端口锁定控制 0 比较输出信号控制相关的端口输出锁存 1 比较输出信号对端口输出锁存的直接控制被禁止

注：只要非交错模式被使能（STAG = 1）、或端口锁定被激活（PL = 1），CAPCOM2 单元将无法修改端口输出寄存器的内容。

交错模式下，CAPCOM2 的工作周期由 8 个模块时钟周期组成，不同寄存器的比较事件交错输出，也就是说，具有相同比较值的不同寄存器，发生比较匹配时不会同时切换相应的输出，而是有一个固定延迟。该操作有助于降低由同时切换输出引起的噪声和峰值功耗。

非交错模式下，CAPCOM2 的工作周期等于 1 个模块时钟周期，具有相同比较值的比较事件在同一时钟周期切换所有的比较输出。该模式提供了运行速度更快、精度更高的 CAPCOM2 单元（为交错模式的 8 倍）。

交错模式

图 17-12 说明交错模式的操作。该例中，所有 CCy 寄存器均工作在比较模式 3。

寄存器 CC16、CC17 和 CC18 设定的比较值均为 FFFE_H 。定时器递增计数到 FFFE_H ，比较器检测到这三个寄存器的比较匹配。比较器匹配后的下一个周期，寄存器 CC16 的输出 CC16IO 切换到 1。但输出 CC17IO 和 CC18IO 不会同时切换、而是在随后的两个周期内分别切换到 1。该交错输出模式依次进行下去，直到寄存器 CC23 的输出完成切换。寄存器编号代表以时钟周期为单位的输出信号的时延 – 寄存器 CC23 比寄存器 CC16 的输出切换晚 7 个周期。图示举例中，寄存器 CC23 的比较值为 FFFD_H 。因此，定时器计数到 FFFD_H 时，在 CAPCOM2 工作周期的最后一个时钟周期对 CC23 的输出进行切换。

定时器溢出时，所有比较输出复位到 0（比较模式 3）。从图 17-12 再次看出输出信号复位交错进行。

寄存器 CC24 到 CC31 的输出切换和寄存器 CC16 到 CC23 的输出切换并行进行。事实上，低寄存器组和高寄存器组交错输出并行进行。采用这种方式，可确保双寄存器比较模式下寄存器对的两个比较信号的操作同时进行。

交错模式下可进行直接端口锁存切换（请参阅章节 17.6）。不过，也可用相关端口引脚的复用输出功能输出比较信号。

注：上述为交错模式的基本描述，只适用于和引脚相连的通道。

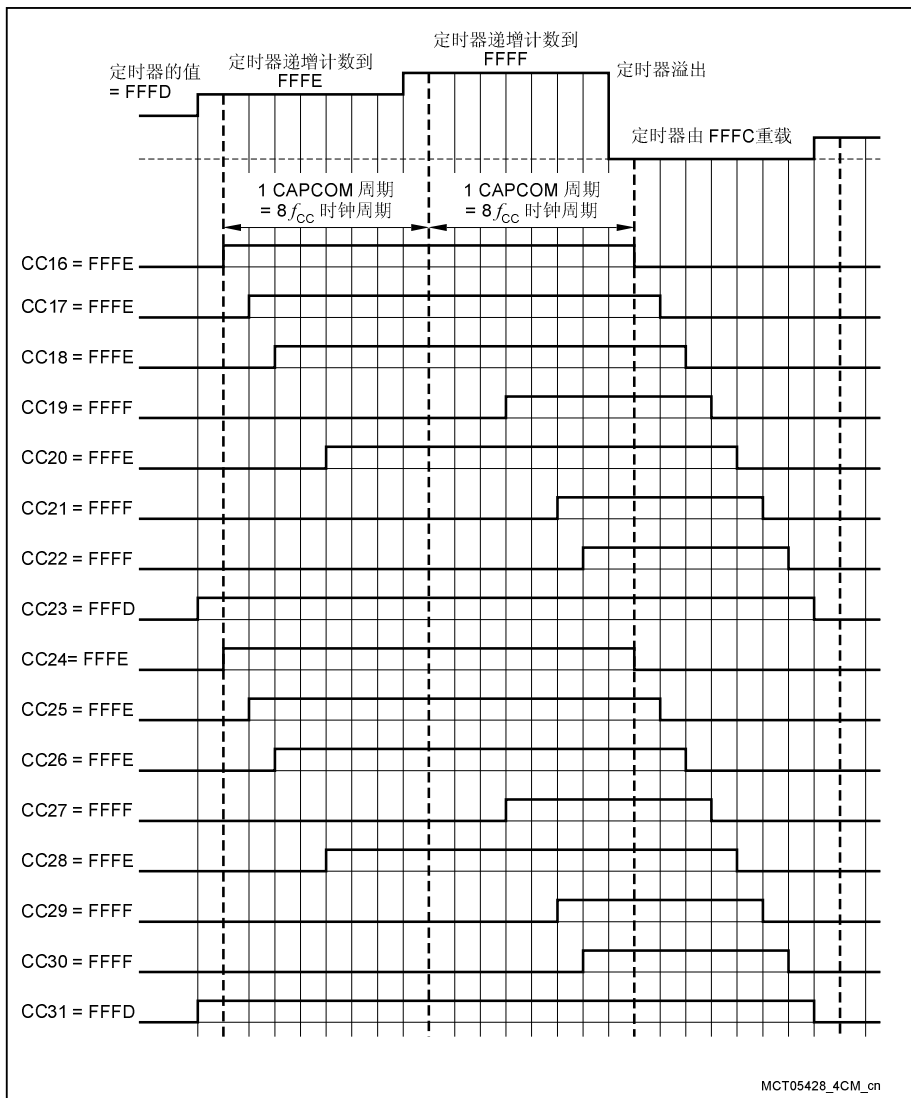


图 17-12 交错模式操作

非交错模式

为了使 CAPCOM2 单元运行速度最快、精度最高，可选择非交错模式。该模式下，CAPCOM2 的一个工作周期等于 1 个模块时钟周期。定时器加 1 操作、新计数值和比较值的比较在一个时钟周期之内完成，相应的输出信号在下个时钟周期切换（和下次定时器加 1 和比较并行进行）。

图 17-13 说明非交错模式的操作。请注意，当定时器溢出时，也需要额外一个时钟周期来切换输出信号。

注：非交错模式下，直接端口锁存切换被禁止。

上述为非交错模式的基本描述，只适用于和引脚相连的通道。

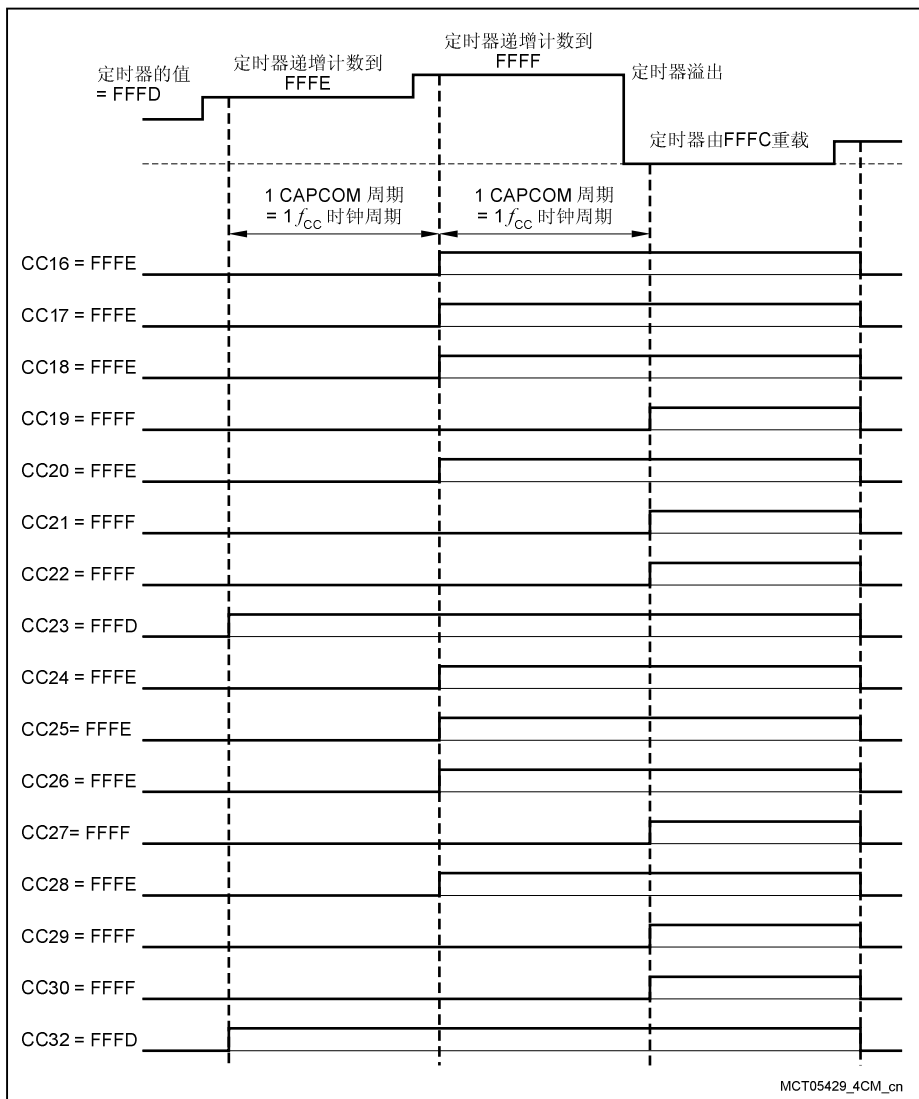


图 17-13 非交错模式操作

17.9 CAPCOM2 中断

发生捕获或比较事件时，相应捕获/比较寄存器 CCy 的中断请求标志 CCyIR 被自动置位。若中断使能位 CCyIE 被置位，该中断标志可被用于产生中断或触发 PEC 服务请求。捕获中断可看作具有附加特性（可记录发生触发事件的时间）的外部中断请求。

每个捕获/比较寄存器有各自的可位寻址的中断控制寄存器和中断向量。这些寄存器和所有其它中断控制寄存器的结构相同。寄存器基本结构如下所示，表 17-4 列出这些寄存器的地址。

CC2_CCyIC

CAPCOM 中断控制寄存器 (E)SFR (表 17-4) 复位值: - - 00_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	GPX	CCy IR	CCy IE	ILVL			GLVL		
-	-	-	-	-	-	-	rw	rwh	rw	rw			rw		

注：控制位域的具体解释请参阅中断控制寄存器的描述。

表 17-4 CAPCOM2 单元中断控制寄存器地址

CAPCOM2 单元		
寄存器	地址	寄存器区
CC2_CC16IC	F160 _H /B0 _H	ESFR
CC2_CC17IC	F162 _H /B1 _H	ESFR
CC2_CC18IC	F164 _H /B2 _H	ESFR
CC2_CC19IC	F166 _H /B3 _H	ESFR
CC2_CC20IC	F168 _H /B4 _H	ESFR
CC2_CC21IC	F16A _H /B5 _H	ESFR
CC2_CC22IC	F16C _H /B6 _H	ESFR
CC2_CC23IC	F16E _H /B7 _H	ESFR
CC2_CC24IC	F170 _H /B8 _H	ESFR
CC2_CC25IC	F172 _H /B9 _H	ESFR
CC2_CC26IC	F174 _H /BA _H	ESFR
CC2_CC27IC	F176 _H /BB _H	ESFR
CC2_CC28IC	F178 _H /BC _H	ESFR
CC2_CC29IC	F184 _H /C2 _H	ESFR
CC2_CC30IC	F18C _H /C6 _H	ESFR
CC2_CC31IC	F194 _H /CA _H	ESFR

17.10 外部输入信号的要求

根据 CAPCOM2 的模块时钟和基本工作模式（交错或非交错），CAPCOM2 模块的逻辑电路对外部输入信号采样。为了确保外部信号能够被正确识别，它的当前电平值（高或低）必须至少保持一个完整的采样周期。

非交错模式下的采样周期为 1 个模块时钟；交错模式下的采样周期为 8 个模块时钟。为了识别信号的跳变，必须对其采样两次。若连续两次采样的电平值不同，即可识别到信号发生了跳变。因此，对外部输入信号采样至少需要两个采样周期，故输入信号的最大频率一定不能高于非交错模式下模块时钟的一半、或者交错模式下模块时钟的 1/16。

外部输入信号参数限制见 [表 17-5](#)。

表 17-5 CAPCOM2 外部输入信号参数限制

	非交错模式	交错模式
输入信号频率的最大允许值	$f_{CC}/2$	$f_{CC}/16$
输入信号电平保持稳定的最短时间	$1/f_{CC}$	$8/f_{CC}$

外部信号用作计数或捕获输入时，和其相连的端口引脚必须设置为输入。

注：测试时，用作计数或捕获输入的引脚可设置为输出。可通过软件或其它外设控制对应信号，从而触发计数或捕获事件。

为了将比较输出信号送出，相关端口引脚必须设置为输出。比较输出信号可直接控制端口锁存、或控制（用作复用端口输出功能的）CC2_OUT 锁存的输出。

17.11 CAPCOM2 单元接口

CAPCOM2 单元（[图 17-14](#)）以不同的方式和周围资源接口。

内部连接

GPT2 定时器 T6 的上溢/下溢信号 T6OUF 连接至 CAPCOM2 单元，为 CAPCOM2 的定时器提供了一个可选时钟源。

CAPCOM2 单元的 18 条中断请求线送至中断控制模块。

注：和 CAPCOM2 单元相连的 P1 口的高 6 条输入线还可用作独立的外部中断输入。

通道 31 的中断请求线和 ADC 相连，用作插入转换的触发源。

外部连接

CAPCOM2 单元的 12 个捕获/比较信号和 XC164CM 的输入/输出端口相连。根据设定的方向，这些端口可提供来自外部系统的捕获触发信号（输入）、或将比较输出信号送到外部电路（输出）。

注：捕获触发信号也可取自输出引脚。例如，在这种情况下，软件可产生触发沿。
定时器 T7 的时钟源可来自外部信号。

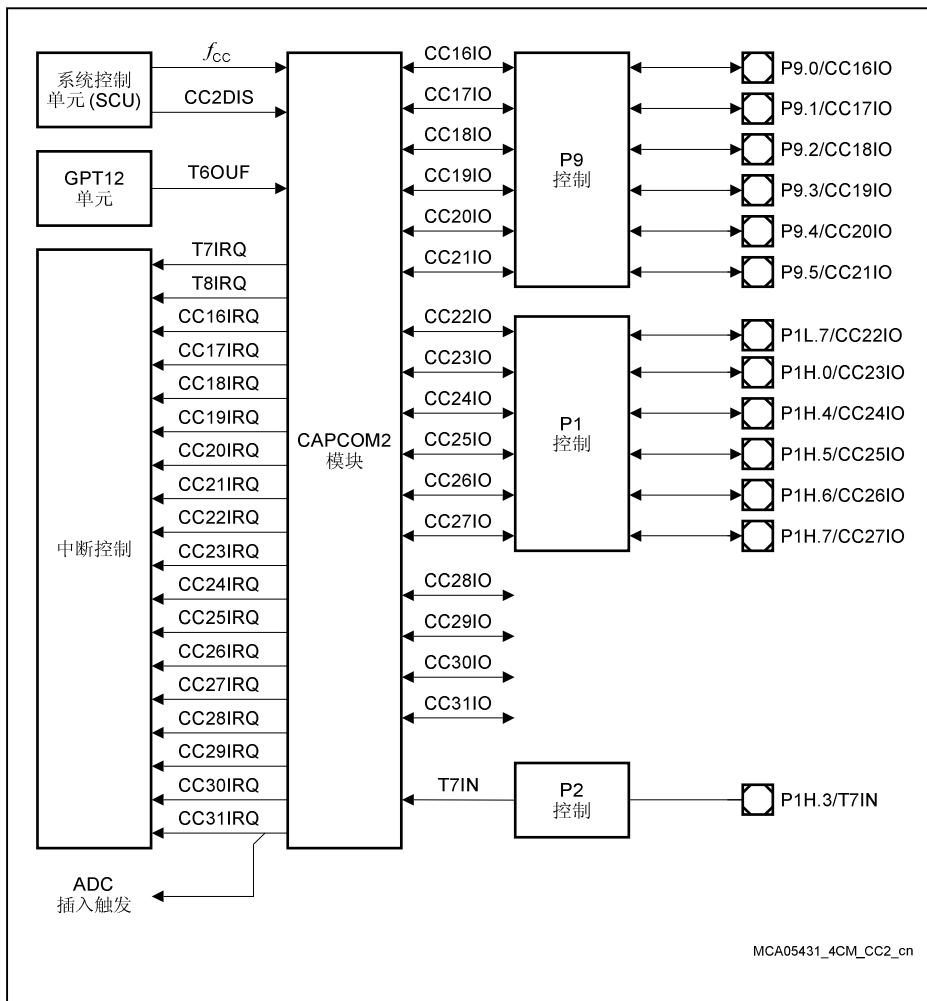


图 17-14 CAPCOM2 单元接口

18 捕获/比较单元 6（CAPCOM6）

XC164CM 系列中的几款产品（见表 1-1）具有捕获比较单元 6（CAPCOM6），该单元由带有三路捕获/比较通道的定时器 T12 和带有一路比较通道的定时器 T13 组成。T12 的各通道既能独立产生 PWM 信号或接受捕获信号，各通道也可共同产生驱动交流电机或逆变器的控制信号序列。CAPCOM6 的特殊工作模式可用于控制无刷直流电机，使用霍尔传感器或利用反电动势检测的控制方案。此外，CAPCOM6 支持块切换和多相电机控制。

数据寄存器		控制寄存器		中断控制		系统寄存器	
T12	X	CMPSTAT	X	CCU6_IS	X	P1L	
T12PR	X	CMPMODIF	X	CCU6_ISS	X	DP1L	E
T12DTC	X	TCTR0	X	CCU6_ISR	X	ALTSEL0P1L	E
CC60R	X	TCTR2	X	CCU6_INP	X		
CC60SR	X	TCTR4	X	CCU6_IEN	X	P1H	
CC61R	X	MODCTR	X			DP1H	E
CC61SR	X	TRPCTR	X	CCU6_T12IC	E	ALTSEL0P1H	E
CC62R	X	PSLR	X	CCU6_T13IC	E		
CC62SR	X	MCMOUTS	X	CCU6_EIC	E	SYSCON3	E
T13	X	MCMOUT	X	CCU6_IC	E		
T13PR	X	MCMCTR	X				
CC63R	X	T12MSEL	X				
CC63SR	X						

T12/T13	CAPCOM6 定时器T12/T13计数寄存器	TRPCTR	强制中断控制寄存器
T12PR/T13PR	CAPCOM6 T12/T13 周期寄存器	PSLR	被动态电平寄存器
T12DTC	CAPCOM6 T12 死区时间控制寄存器	IS	中断状态寄存器
TCTR _x	CAPCOM6 定时器控制寄存器	ISS/ISR	中断状态置位/复位寄存器
T12MSEL	T12 捕获/比较模式选择寄存器	IEN	中断使能寄存器
CC6 _x R	CAPCOM6 捕获/比较寄存器	INP	中断节点指针寄存器
CC6 _x SR	CAPCOM6 捕获/比较映射寄存器	T12IC/T13IC	CAPCOM6 定时器中断控制寄存器
CMPSTAT	捕获/比较状态寄存器	CCU6_IC	CAPCOM6 中断控制寄存器
CMPMODIF	捕获/比较状态修改寄存器	CCU6_EIC	CAPCOM6 出错中断控制寄存器
MODCTR	调制控制寄存器	P1L/P1H	P1口数据寄存器
MCMOUT	多通道模式输出寄存器	DP1L/DP1H	P1口方向控制寄存器
MCMOUTS	多通道模式输出映射寄存器	ALTSEL0P1x	P1口复用功能输出选择寄存器0
MCMCTR	多通道模式控制寄存器	SYSCON3	系统控制寄存器3（外设管理）

mc_capcom60100_registers_cn.vsd

图 18-1 CAPCOM6 相关的特殊功能寄存器（SFR）

CAPCOM6 中丰富的状态位、参数的同步更新（通过映射寄存器实现）、中断请求信号的灵活产生，为用户提供了高效的软件控制方式。

定时器 T12 模块特性：

- 3 路捕获/比较通道，可单独用作捕获或比较通道
- 支持三相 PWM 产生（6 路输出，每相两路信号分别用于控制上桥臂或下桥臂开关）
- 16 位精度，最大计数频率 = 外设时钟频率
- 各路通道的死区时间控制，以避免功率器件短路
- T12/T13 寄存器同步更新
- 支持中心对齐和边沿对齐 PWM 产生
- 支持单次模式
- 支持多中断请求源
- 支持类磁滞控制模式

定时器 T13 模块特性：

- 一个独立的比较通道输出
- 16 位精度，最大计数频率 = 外设时钟频率
- 可与 T12 同步
- 周期匹配和比较匹配时产生中断
- 支持单次模式

附加特性：

- 支持用于驱动无刷直流电机的块切换
- 通过霍尔序列进行位置检测
- 霍尔效应噪声滤波
- 块切换的自动转速测量
- 综合错误处理
- 由外部信号（ $\overline{\text{CTRAP}}$ ）控制快速急停，无需 CPU 干预
- 多通道交流驱动的控制模式
- 输出电平可选，以适配功率级器件

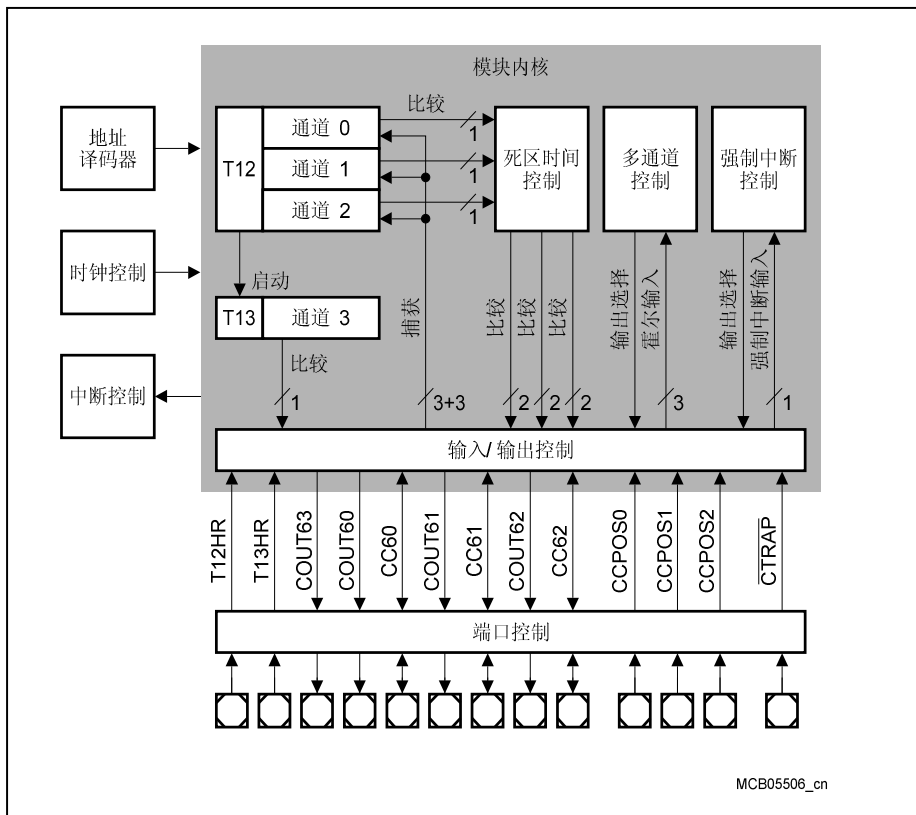


图 18-2 CAPCOM6 框图

定时器 T12 的三路通道可分别工作在捕获和/或比较模式，两种模式也可组合使用。定时器 T13 只能工作在比较模式。多通道控制单元产生可由 T12 和/或 T13 调制的输出序列。信号的调制源可选，并可组合使用。

18.1 定时器 T12 模块

定时器 T12 是产生三相 PWM 信号的主要单元。一个 16 位计数器通过比较器和 3 路通道寄存器相连，计数器的值和某个通道寄存器的值匹配时，产生一个信号。CAPCOM6 单元为 T12 模块提供了多种控制功能，以便适用于不同应用。

除产生三相 PWM 信号之外，T12 模块还支持单通道比较和捕获功能、以及死区时间控制和类滞滞比较模式。

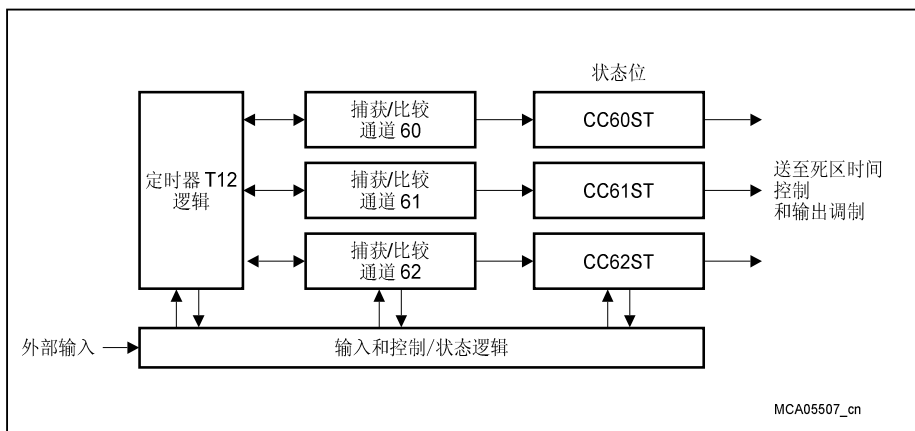


图 18-3 定时器 T12 模块总览

图 18-4 为定时器 T12 的详细框图。定时器 T12 的输入时钟 f_{T12} 来自经过预分频处理的模块时钟 f_{CC6} （预分频因子可编程设定）和 1/256 分频（可选），分频因子分别由位域 T12CLK 和 T12PRE 选择（见**表 18-1**）。T12 可递增或递减计数，计数方向取决于所选择的工作模式。计数方向标志 CDIR 指明当前的计数方向。

定时器 T12 通过比较器和周期寄存器 T12PR 相连，该寄存器的内容决定了 T12 的最大计数值。边沿对齐模式下，T12 计数到周期值后复位为 0000_H。中心对齐模式下，T12 计数到周期值后，计数方向从“递增”变为“递减”（请注意，在该模式下，T12 递增计数到周期值加 1 后转为递减计数）。上述两种情况下，均产生信号 T12_PM

（T12 周期匹配）。周期寄存器的新周期值从映射周期寄存器 T12PS 中获取，T12PS 的值由软件载入。“T12 映射传送”控制信号 T12_ST 控制将新周期值从映射寄存器传送到 T12PR 中的操作（见**章节 18.8**），该信号的产生取决于当前的工作模式和控制位 STE12 的状态。CAPCOM6 单元提供了周期值、以及（和产生 PWM 信号相关的）其它数值的映射寄存器，便于用户编程同步更新所有相关参数。

信号 T12_ZM 和 T12_OM 分别指示计数器的值是否等于 0000_H 或 0001_H。这些信号控制 T12 的计数和切换操作。

捕获/比较单元 6 (CAPCOM6)

T12 基本工作模式：边沿对齐模式（见图 18-5）或中心对齐模式（见图 18-6），由位 CTM 控制选择。单次模式控制位 T12SSC 控制定时器在当前计数周期结束后自动停止运行（见图 18-7 和图 18-8）。

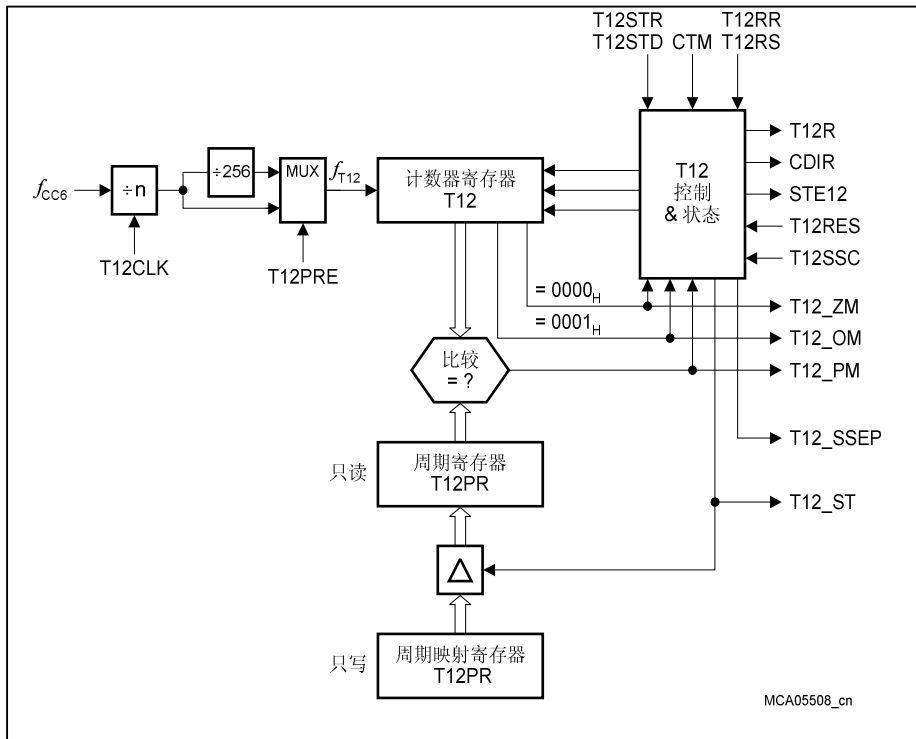


图 18-4 定时器 T12 的逻辑和周期比较器

运行位 T12R 控制 T12 定时器的启动和停止。T12R 可由软件设置：置位 T12RS 则对 T12R 置位，置位 T12RR 则对 T12R 清零；或者根据预先选定的条件硬件复位。

定时器 T12 可由控制位 T12RES 清零。置位“只写”控制位 T12RES 只清零定时器的内容，不会产生其它影响，如不会终止定时器工作。

T12 映射传送控制信号 T12_ST 由位 STE12 控制产生，该控制位可通过相关置位控制位 T12STR，或复位控制位 T12STD 间接设置。

注：控制选择 T12 工作模式的寄存器具体描述请参阅[章节 18.3](#)。

捕获/比较单元 6（CAPCOM6）

寄存器 T12 的内容表示定时器 T12 的计数值。只有当定时器 T12 停止工作时，才可对寄存器 T12 进行写操作；定时器 T12 运行过程中写操作无效。寄存器 T12 始终可由软件读取。

CCU6_T12

定时器 T12 计数寄存器

XSFR（E890H/--）

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T12CV															
rwh															

符号	位序号	读写类型	功能描述
T12CV	[15:0]	rwh	定时器 T12 的计数值 定时器 T12 的 16 位计数值

寄存器 T12PR 中存放着定时器 T12 的周期值。周期值和 T12 的实际计数值进行比较，根据规定的计数规则执行相关操作。

CCU6_T12PR

定时器 T12 周期寄存器

XSFR（E892H/--）

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T12PV															
rwh															

符号	位序号	读写类型	功能描述
T12PV	[15:0]	rwh	定时器 T12 的周期值 T12 的计数值和 T12PV 相等时触发周期匹配。计数到该值后，定时器 T12 被清零（边沿对齐模式）或将其计数方向改为递减计数（中心对齐模式）。

该寄存器对应一个映射寄存器（占用相同的物理地址），位 STE12 控制映射传送。软件读操作读出当前用于周期比较的周期值；而写操作将周期值写入映射寄存器中。映射寄存器结构便于用户同步更新 T12 的所有相关值。

18.1.1 定时器 T12 的操作

定时器 T12 的输入时钟 f_{T12} 来自经过预分频处理的模块时钟 f_{CC6} （预分频因子可编程设定）和 1/256 分频（可选）。总的预分频因子列于表 18-1 中。定时器 T12 不工作时预分频器复位，从而保证产生可重复的时序和延迟。

表 18-1 定时器 T12 输入时钟选择

T12CLK	输入时钟 预分频器关闭（T12PRE = 0）	输入时钟 预分频器开启（T12PRE = 1）
000 _B	f_{CC6}	$f_{CC6}/256$
001 _B	$f_{CC6}/2$	$f_{CC6}/512$
010 _B	$f_{CC6}/4$	$f_{CC6}/1024$
011 _B	$f_{CC6}/8$	$f_{CC6}/2048$
100 _B	$f_{CC6}/16$	$f_{CC6}/4096$
101 _B	$f_{CC6}/32$	$f_{CC6}/8192$
110 _B	$f_{CC6}/64$	$f_{CC6}/16384$
111 _B	$f_{CC6}/128$	$f_{CC6}/32768$

定时器周期由周期寄存器 T12PR 的值和定时器模式决定。

边沿对齐模式下，定时器周期为：

$$T12_{PER} = \langle \text{周期寄存器的值} \rangle + 1; \text{以 } T12 \text{ 时钟频率} (f_{T12}) \text{ 为单位} \quad (18.1)$$

中心对齐模式下，定时器周期为：

$$T12_{PER} = (\langle \text{周期寄存器的值} \rangle + 1) \times 2; \text{以 } T12 \text{ 时钟频率} (f_{T12}) \text{ 为单位} \quad (18.2)$$

定时器 T12 工作时，对计数寄存器 T12 的写操作无效。若 T12 停止工作且死区时间计数器为 0，写入寄存器 T12 的值立刻生效。

如上所述，T12 可工作在边沿对齐或中心对齐模式。两种模式下，由“计数规则”决定 T12 计数器的操作。

边沿对齐模式下，计数规则决定的 T12 操作如图 18-5 所示。

中心对齐模式下（T12 递增和递减计数），计数规则决定的 T12 操作如图 18-6 所示。

定时器 T12 工作在边沿对齐模式下：

- 检测到周期匹配时，计数器在下一个 f_{T12} 时钟复位为 0。T12 始终递增计数（CDIR = 0）。

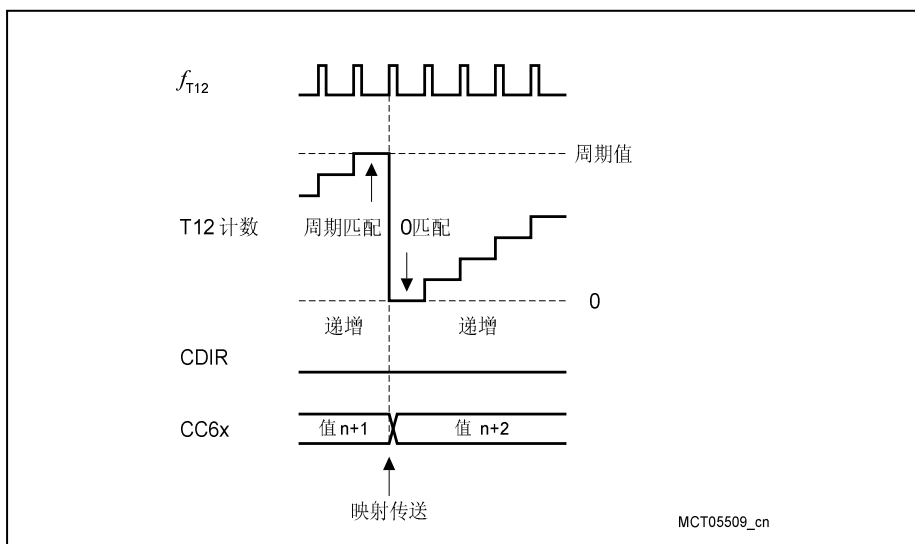


图 18-5 边沿对齐模式下 T12 的操作

定时器 T12 工作在中心对齐模式下：

- 计数器递减计数到 0001_H 时，下一个 f_{T12} 时钟 T12 被设置为递增计数（CDIR = 0）。
- 计数器递增计数，检测到周期匹配时，下一个 f_{T12} 时钟 T12 被设置为递减计数（CDIR = 1）。
- CDIR = 0 时，下一个 f_{T12} 时钟计数器递增计数；CDIR = 1 时，下一个 f_{T12} 时钟计数器递减计数。

注：在 1-匹配或周期匹配后的下一个定时器时钟，位 CDIR 改变。因此，定时器以原先的计数方向再计数一个时钟后才真正改变计数方向（见图 18-6）。

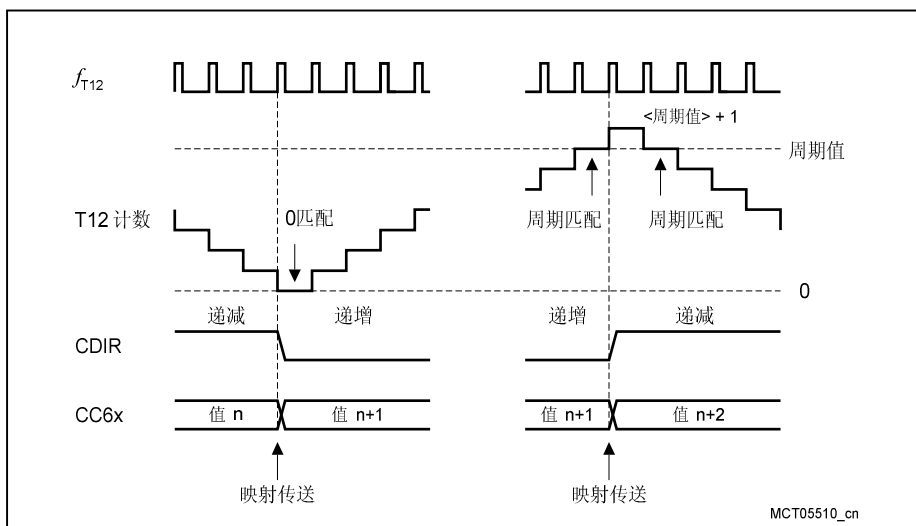


图 18-6 中心对齐模式下 T12 的操作

T12 映射传送信号，T12_ST

采用一个特殊的映射传送信号（T12_ST）便于使周期值、比较值的更新和 T12 操作同步进行。该信号的产生可通过软件设置位 STE12 来请求（“只写”位 T12STR 写入 1 将置位 STE12；“只写”位 T12STD 写入 1 将清零 STE12）。

若请求映射传送（STE12 = 1），以下情况会产生信号 T12_ST：

- 递增计数检测到周期匹配时，或
- 递减计数到 0001_H 时，或
- 定时器 T12 停止工作（T12R = 0）

当信号 T12_ST 有效，在下一个 T12 时钟周期触发映射传送：映射寄存器中的新周期值传送到实际周期寄存器 T12PR 中；映射寄存器中的新比较值传送到实际比较寄存器中（请参阅[章节 18.1.2](#)）。映射寄存器传送之后，位 STE12 自动清零。

T12 启动/停止和复位控制

位 T12R 被置位时，定时器 T12 开始计数。通过设定位 T12R 可控制定时器的运行（“只写”位 T12RS 写入 1 将置位 T12R；“只写”位 T12RR 写入 1 将清零 T12R）。单次模式下 T12R 也可由硬件清零。

“只写”位 T12RES 置 1 将对定时器 T12 清零。该操作只清零定时器（置为 0000_H）、不会引发其它操作，比如不会清零定时器的运行位。

单次模式

单次模式下，运行位 **T12R** 也受硬件影响。可由位 **T12SSC** 使能单次模式。
T12SSC 被置位时，当前定时器周期结束之后定时器停止运行。边沿对齐模式下，定时器计数到周期值后被清零至 **0000_H**，计数周期结束；中心对齐模式下，定时器递减计数到 **0000_H** 时计数周期结束。分别见 [图 18-7](#) 和 [图 18-8](#)。

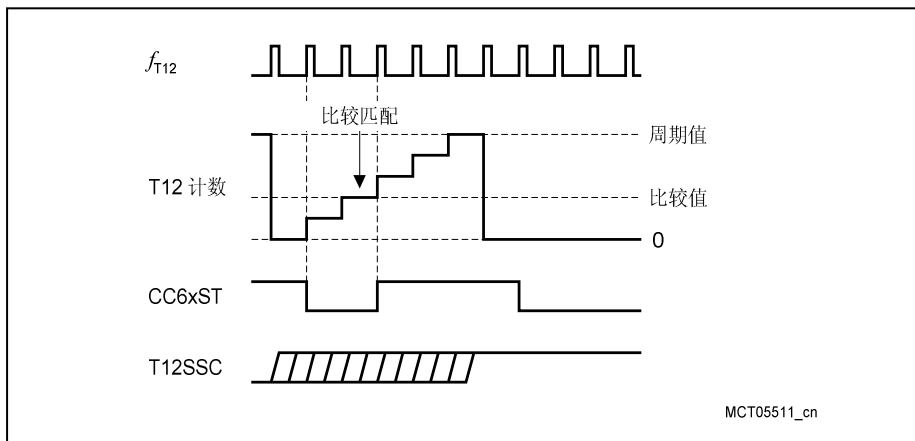


图 18-7 边沿对齐模式下的单次操作

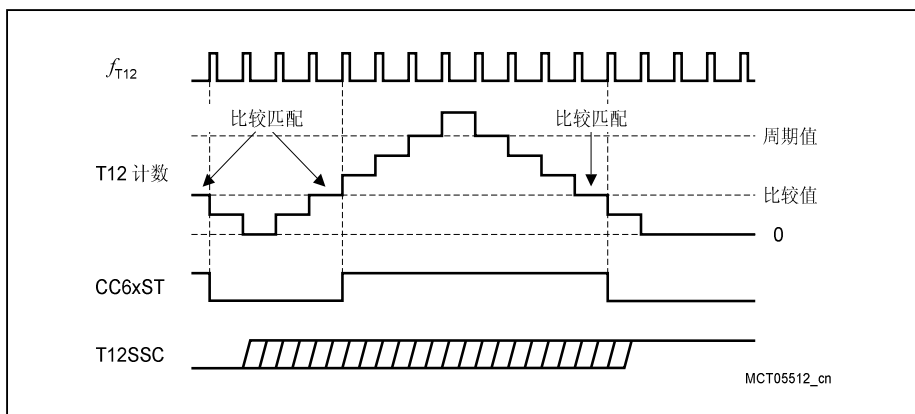


图 18-8 中心对齐模式下的单次操作

18.1.2 T12 的比较模式

定时器 T12 的三路独立的捕获/比较通道可协同 T12 的计数值，分别执行比较或捕获操作。捕获功能的具体内容请参阅[章节 18.1.4](#)。

比较模式下（见[图 18-9](#)），三路通道既可独立工作，也可共同产生三相 PWM 序列。

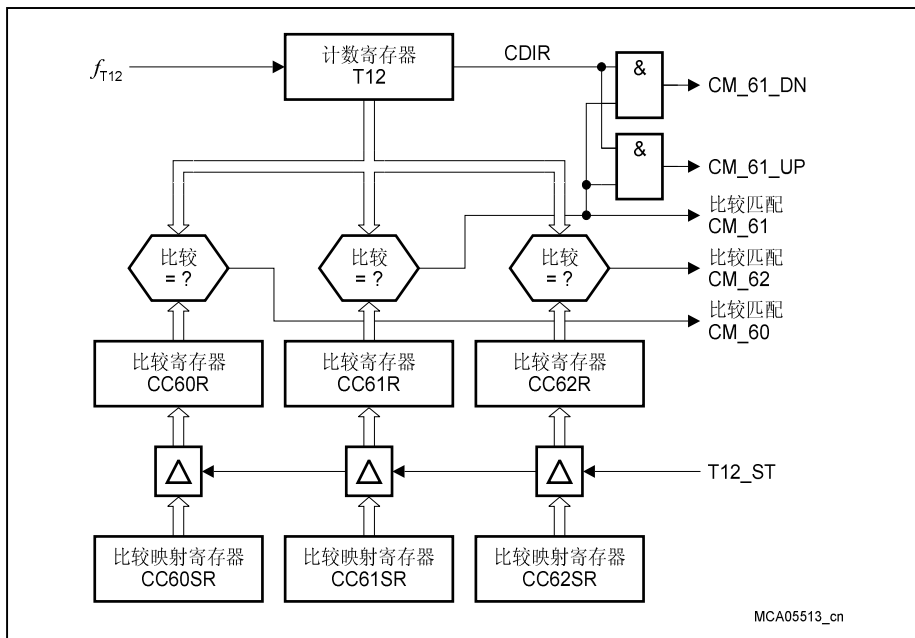


图 18-9 T12 通道比较器

每路通道通过各自的相等比较器和 T12 的计数寄存器相连，当计数器的值和比较寄存器中的值匹配时产生匹配信号。每路通道由比较器和双寄存器结构（实际比较寄存器 CC6xR 和相关映射寄存器 CC6xSR）组成。CC6xR 与比较器直接相连，CC6xSR 由软件预先加载。当 T12 的映射传输信号 T12_ST 有效时，映射寄存器中的内容被传送到实际比较寄存器中。CAPCOM6 单元提供了比较值、以及（和产生 PWM 信号相关的）其它数值的映射寄存器，便于用户软件同步更新所有相关参数（请参阅[章节 18.8](#)）。

和每路通道相关的状态位 CC6xST 保存比较（或捕获）操作的状态（见[图 18-10](#)）。

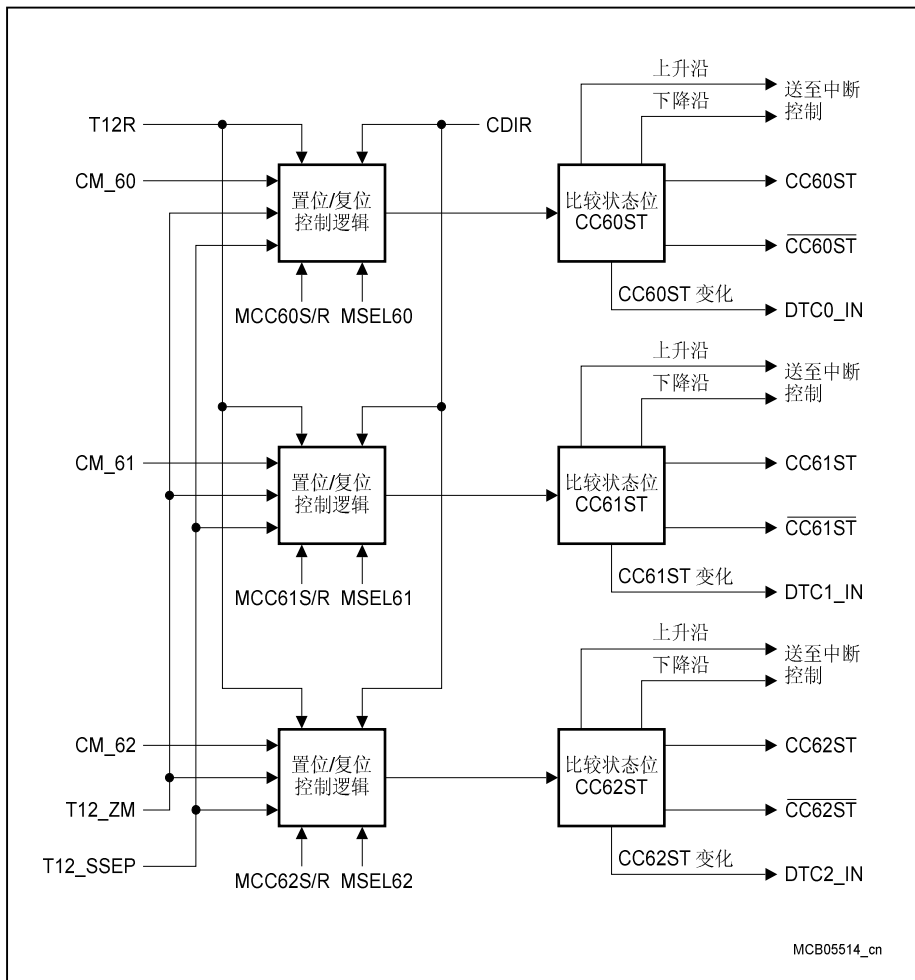


图 18-10 比较模式下的比较状态位框图

状态位 $CC6xST$ 置位/复位控制逻辑的输入包括：计数方向标志位（CDIR），定时器运行位（T12R），定时器 0-匹配信号（T12_ZM），单次模式结束信号（T12_SSEP），各通道的比较匹配信号 CM_6x 以及模式控制位 MSEL6x。此外，可通过设定置位控制位 MCC6xS 和复位控制位 MCC6xR 分别置位或复位各状态位。

注：霍尔传感器模式下，状态位还由附加输入控制（请参阅[章节 18.5](#)）。

只有当定时器 T12 运行时 ($T12R = 1$) 才可硬件修改状态位 CC6xST。硬件修改状态位时，比较模式下应用以下规则置位和复位状态位（如图 18-11 和图 18-12 所示）：

状态位 CC6xST 的置位条件：

- T12 递增计数到比较匹配后，下一个 T12 时钟 (f_{T12}) 置位 CC6xST（即，计数器递增计数到超过比较值）；
- T12 递增计数到 0-匹配且同时为比较匹配时，下一个 T12 时钟 (f_{T12}) 置位 CC6xST。

状态位 CC6xST 的复位条件：

- T12 递减计数到比较匹配后，下一个 T12 时钟 (f_{T12}) 复位 CC6xST（即，计数器递减计数到低于比较值）；
- T12 递增计数到 0-匹配但同时不是比较匹配时，下一个 T12 时钟 (f_{T12}) 复位 CC6xST。

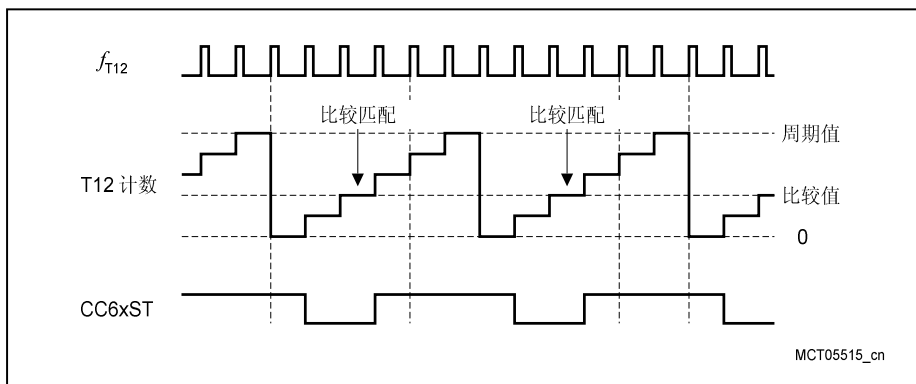


图 18-11 边沿对齐模式下的比较操作

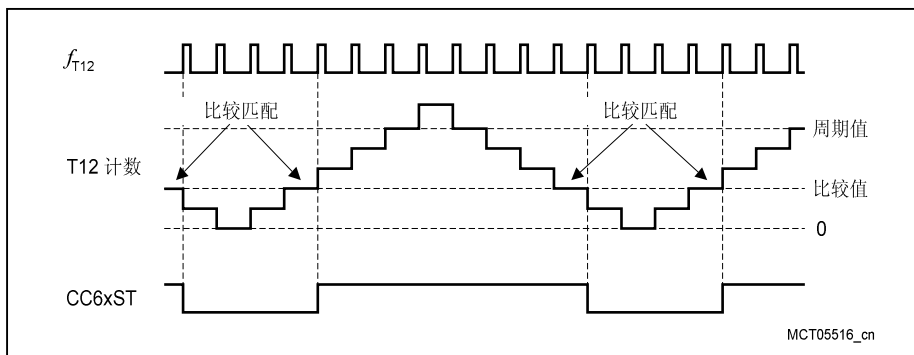


图 18-12 中心对齐模式下的比较操作

图 18-13 例举出更多的比较波形。请务必注意：图中假定定时器计数过程中某些比较值被修改。修改值预先载入映射寄存器 **CC6xSR** 中，当 **T12** 的映射传送信号 **T12_ST**（假定被使能）有效时，该修改值被传送到实际比较寄存器 **CC6xR** 中。

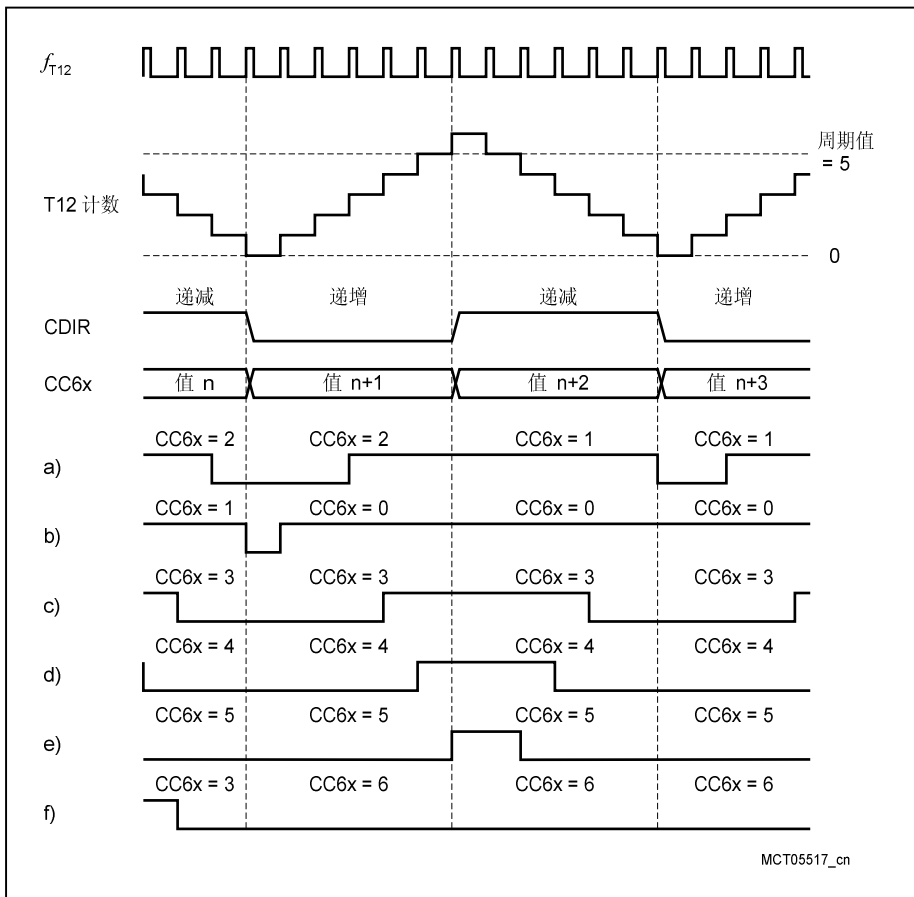


图18-13 比较波形举例

例 b) 图示说明跳变到占空比为 100%的情况。先使用比较值 0001_H，然后变为 0000_H。请注意：在新比较值 0000_H生效的定时器周期内，仍产生长度为一个 T12 时钟周期的低脉冲（该脉冲由前面的比较值 0001_H产生）。接下来的定时器周期，状态位 CCxST 始终保持为 1，产生占空比为 100%的信号。此时，比较规则“0-匹配且同时为比较匹配”生效。

例 f) 图示说明跳变到占空比为 0%的情况。新比较值设定为 <周期寄存器的值> + 1，状态位 CC6xST 始终为 0。

图 18-14 例举三通道比较波形的产生。通过恰当的死区时间控制和输出调制，可产生高效的三相 PWM 信号。

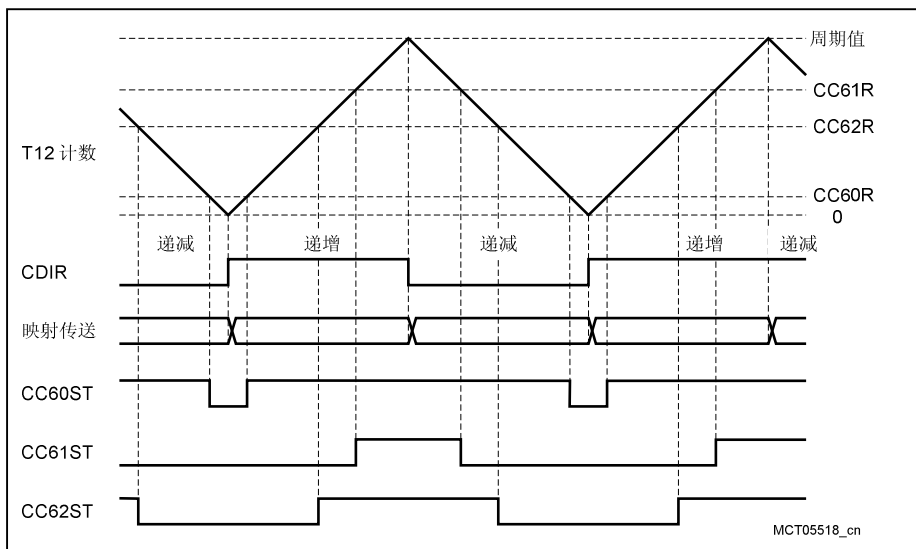


图 18-14 三通道比较波形

比较模式输出路径

图 18-15 给出从通道状态位到其输出引脚的最简信号路径。如图所示，用户可通过多种控制选择，决定当前状态位 **CC6xST** 所对应的（期望）输出信号的切换操作。输出调制的具体内容请参阅**章节 18.7**。

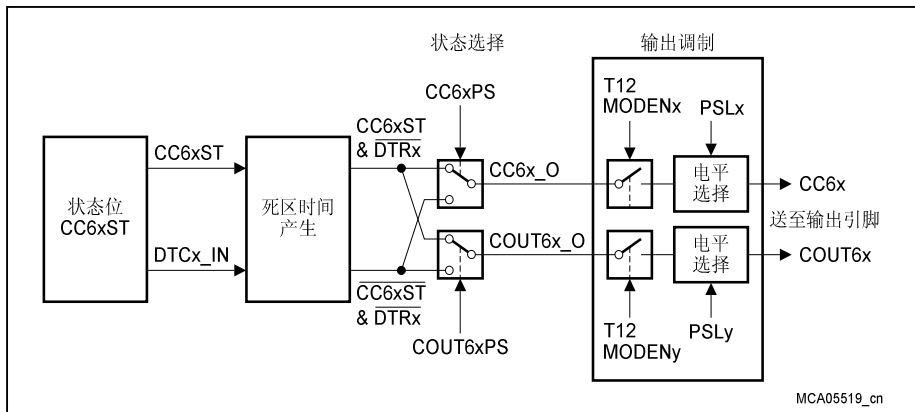


图 18-15 比较模式简化输出路径框图

比较模式寄存器

比较模式下，寄存器 CC6xR (x = 0, 1, 2) 是 T12 的实际比较寄存器。保存在 CC6xR 中的值和 T12 的计数值进行比较（三路通道并行进行）。寄存器 CC6xR 为只读寄存器；若要修改 CC6xR 的值，将映射寄存器 CC6xSR 中的修改值映射传送到 CC6xR 中。映射寄存器可由软件读写。

捕获模式下，当检测到相关的捕获事件时，T12 寄存器的当前计数值捕获到寄存器 CC6xR 或 CC6xSR 中（取决于选择的模式）。

CCU6_CC60R

通道 0 捕获/比较寄存器					XSFR (E898H/--)					复位值: 0000H					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC60V															
rh															

CCU6_CC61R

通道 1 捕获/比较寄存器					XSFR (E89A _H --)					复位值: 0000 _H					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC61V															
rh															

CCU6_CC62R

通道 2 捕获/比较寄存器					XSFR (E89C _H --)					复位值: 0000 _H					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC62V															
rh															

符号	位序号	读写类型	功能描述
CC6xV (x = 0,1, 2)	[15:0]	rh	通道 x 比较值 比较模式下，位域 CC6xV 中存放着和 T12 的计数值进行比较的值；捕获模式下，T12 的捕获值可从这些寄存器读出。

CCU6_CC60SR

通道 0 捕获/比较映射寄存器 XSFR (E8A0H/--) 复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC60S															
rwh															

CCU6_CC61SR

通道 1 捕获/比较映射寄存器 XSFR (E8A2H/--) 复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC61S															
rwh															

CCU6_CC62SR

通道 2 捕获/比较映射寄存器 XSFR (E8A4H/--) 复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC62S															
rwh															

符号	位序号	读写类型	功能描述
CC6xS (x = 0,1, 2)	[15:0]	rwh	通道 x 比较值映射寄存器 比较模式下，映射传送期间，位域 CC6xS 的内容被传送到寄存器 CC6xR 的位域 CC6xV 中；捕获模式下，T12 的捕获值可从这些寄存器读出。

捕获/比较单元 6（CAPCOM6）

寄存器 T12MSEL 用于设置定时器 T12 模块中的三路通道对应的捕获/比较功能。

CCU6_T12MSEL

T12 模式选择寄存器

XSFR（E8C6H/--）

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	MSEL62				MSEL61				MSEL60			
-	-	-	-	rw				rw				rw			

符号	位序号	读写类型	功能描述
MSEL62 MSEL61 MSEL60	[11:8] [7:4] [3:0]	rw	捕获/比较模式选择 这些位域为 T12 的三路捕获/比较通道选择相应的工作模式。每路通道（x = 0, 1, 2）可分别被设定为其中一种模式（霍尔传感器模式除外），见表 18-2。

表 18-2 捕获/比较模式总览

MSEL6x	选择的工作模式
0000 _B	禁止比较输出，引脚 CC6x 和 COUT6x 可用作通用 IO。
0001 _B	比较结果从引脚 CC6x 输出，COUT6x 可用作通用 IO。
0010 _B	比较结果从引脚 COUT6x 输出，CC6x 可用作通用 IO。
0011 _B	比较结果从引脚 COUT6x 和 CC6x 输出。
01XX _B	双寄存器捕获模式，见章节 18.1.4 和表 18-3。
1000 _B	霍尔传感器模式，见章节 18.5。为确保能够正确使能该模式，三组位域 MSEL6x 必须都被设置为霍尔传感器模式。
1001 _B	类磁滞模式，见章节 18.1.5。
101X _B 11XX _B	多输入捕获模式，见章节 18.1.4 和表 18-4。

注：可从通道状态（修改）寄存器中获取通道的状态信息，具体内容请参阅章节 18.3。

18.1.3 死区时间产生

功率逆变器中某一相的上下桥臂开关信号（信号互补）由同一个比较通道产生。例如，若 T12 计数值大于比较值时，上桥臂的控制开关导通（状态位 = 1），那么，当 T12 计数值小于比较值时，下桥臂的控制开关导通（状态位 = 0）。

大多数情况下，相互连接的功率开关的开关特性不对称，即接通和断开所需时间不同。若功率器件的接通时间小于断开时间，通常会引发问题，导致逆变器桥臂短路，从而可能会损坏整个系统。为了通过硬件解决该问题，CAPCOM6 单元中有一个可编程的死区时间产生模块，使信号从被动态到主动态的切换延迟产生（主动态到被动态的切换不延迟）。

T12 三路通道的死区时间产生模块的结构相似，如图 18-16 所示。只要位 CC6xST 变化，即触发对应的死区时间计数器工作。死区时间计数器为单次、6 位递减计数器，该计数器的时钟源和定时器 T12 的相同（ f_{T12} ）。状态位 CC6xST 的变化会激活触发脉冲 DTCx_IN，引发死区时间计数器重载，将保存在寄存器 T12DTC 中的值 DTM 重新装入计数器并启动其计数。

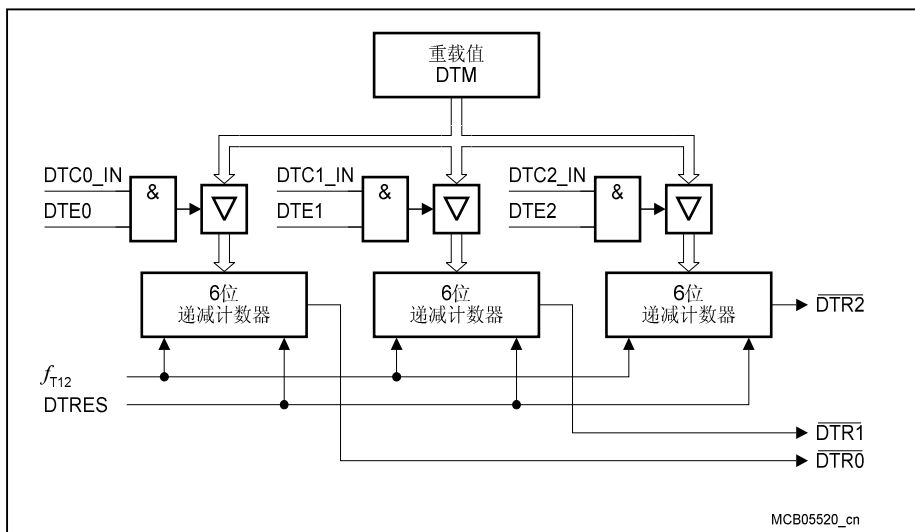


图 18-16 死区时间产生框图

递减计数过程中，输出线 \overline{DTRx} 为 0。因此，重载值决定了输出信号处于低电平（被动态）的时间长短，从而决定了状态位从 0 跳变到 1 的输出时延。输出信号的主动态由 1 表示。

可编程的重载值 DTM 对三路通道均适用。

计数器计数到 0 时停止计数，输出线 $\overline{\text{DTRx}}$ 置 1。 $\overline{\text{DTRx}}$ 为 0 时每相的两路输出均被强制为被动态， $\overline{\text{DTRx}}$ 为 1 时输出能够转入主动态。

每一个死区时间计数器对应有各自的使能控制位 DTEx ，用来使能触发输入 DTCx_IN 。计数器只有在不运行时才能被重载，从而避免了在检测到状态位 CC6xST 变化时，重新触发死区时间计数。图 18-17 说明死区时间波形的产生（被动态 = 0）。下面将对相关的寄存器进行详细说明。

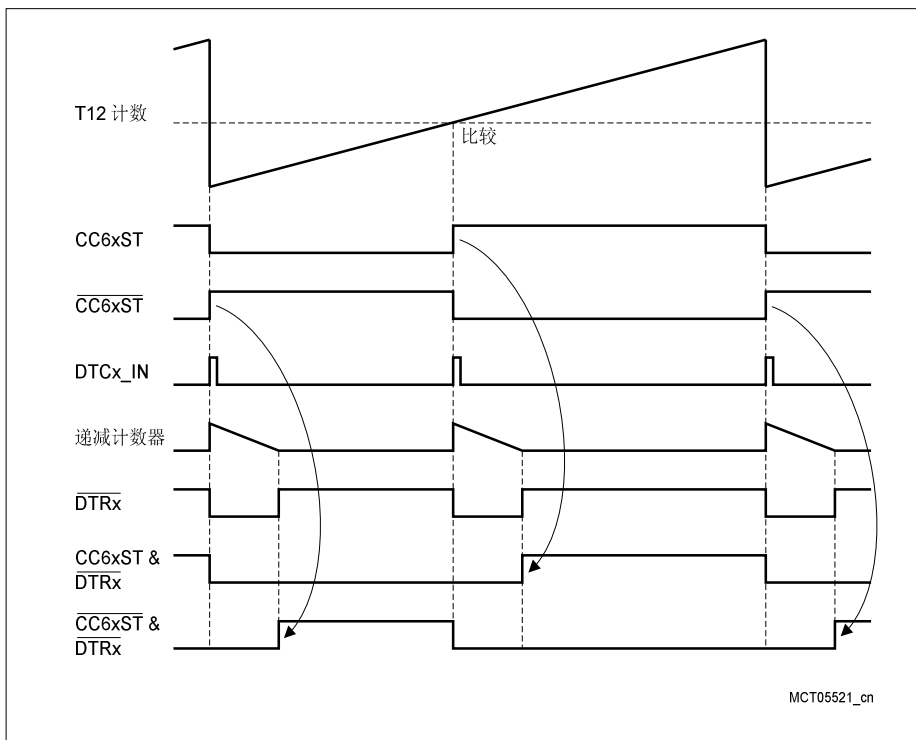


图 18-17 死区时间产生波形

捕获/比较单元 6（CAPCOM6）

寄存器 T12DTC 控制 T12 比较通道的死区时间产生。可分别使能/禁止各路通道的死区时间产生。若被使能，被动态到主动态的跳变时延由位域 DTM 决定。

CCU6_T12DTC

T12 死区时间控制寄存器

XSFR (E894H/--)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	DTR 2	DTR 1	DTR 0	-	DTE 2	DTE 1	DTE 0	-	-	DTM					
-	rh	rh	rh	-	rw	rw	rw	-	-	rw					

符号	位序号	读写类型	功能描述
DTR2 DTR1 DTR0	14, 13, 12	rh	死区时间运行指示标志 指示定时器 T12 各路比较通道（0，1，2）死区时间产生的状态。 0 死区时间计数器停止工作 1 死区时间计数器正在计数（时延有效）
DTE2 DTE1 DTE0	10, 9, 8	rw	死区时间产生使能位 使能/禁止定时器 T12 各路比较通道（0，1，2）的死区时间产生。 0 死区时间产生被禁止 1 死区时间产生被使能
DTM	[5:0]	rw	死区时间值 DTM 规定了输出从被动态到主动态的可编程切换时延。

注：置位寄存器 TCTR4 中的位 DTRES 可对死区时间计数器清零。

18.1.4 T12 捕获模式

T12 模块的三路通道均可响应外部信号，捕获 T12 的时间信息。

CAPCOM6 单元提供了几种不同的捕获模式。所有捕获模式下，均使用各路通道的双寄存器结构（原寄存器和映射寄存器）。这样可降低中断发生率，每两个事件 CPU 才需响应一次。寄存器 T12MSEL 中的位域 MSEL6x 分别为各路通道选择捕获模式。

表 18-3 捕获模式总览

MSEL6x	模式	引脚	有效沿	CC6nSR 保存到	T12 保存到
0100 _B	1	CC6x	上升沿	-	CC6xR
		CC6x	下降沿	-	CC6xSR
0101 _B	2	CC6x	上升沿	CC6xR	CC6xSR
0110 _B	3	CC6x	下降沿	CC6xR	CC6xSR
0111 _B	4	CC6x	任意沿	CC6xR	CC6xSR

图 18-18 说明捕获模式 1 的操作。检测到输入引脚 CC6x 的上升沿（0 到 1 跳变）时，将定时器 T12 的当前计数值捕获到 CC6xR 中；检测到输入引脚 CC6x 的下降沿（1 到 0 跳变）时，将定时器 T12 的内容捕获到 CC6xSR 中。

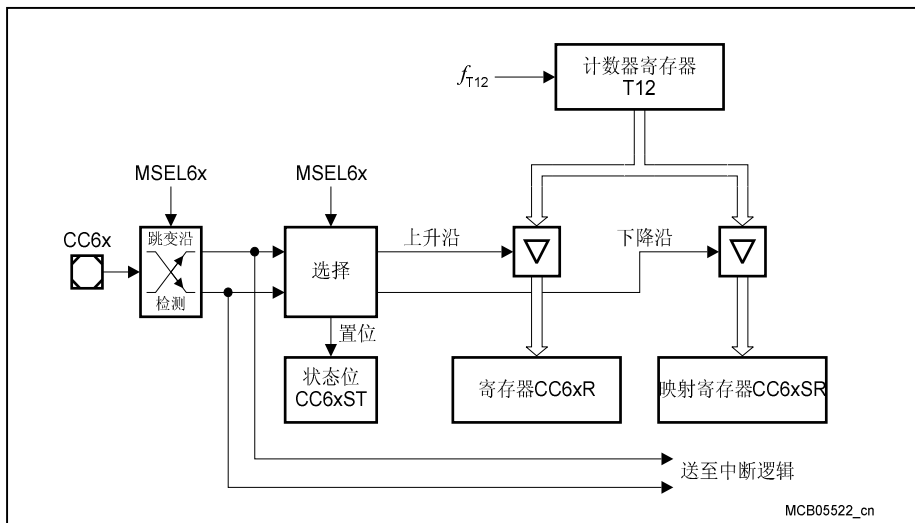


图 18-18 捕获模式 1 框图

捕获/比较单元 6 (CAPCOM6)

捕获模式 2、3 和 4 如图 18-19 所示，各模式的差别仅在于引发捕获操作的有效跳变沿。当处于三种捕获模式中的任意一种，检测到输入引脚 CC6x 出现有效跳变时，对应的映射寄存器 CC6xSR 的当前值被锁存到寄存器 CC6xR 中，定时器 T12 的当前值捕获到寄存器 CC6xSR 中（同时传送）。捕获模式 2 的有效沿为引脚 CC6x 的上升沿；捕获模式 3 的有效沿为引脚 CC6x 的下降沿；捕获模式 4 的有效沿为引脚 CC6x 的上升沿或下降沿，见表 18-3。当输入信号连续两次跳变间隔很短时，这些捕获模式非常有用。

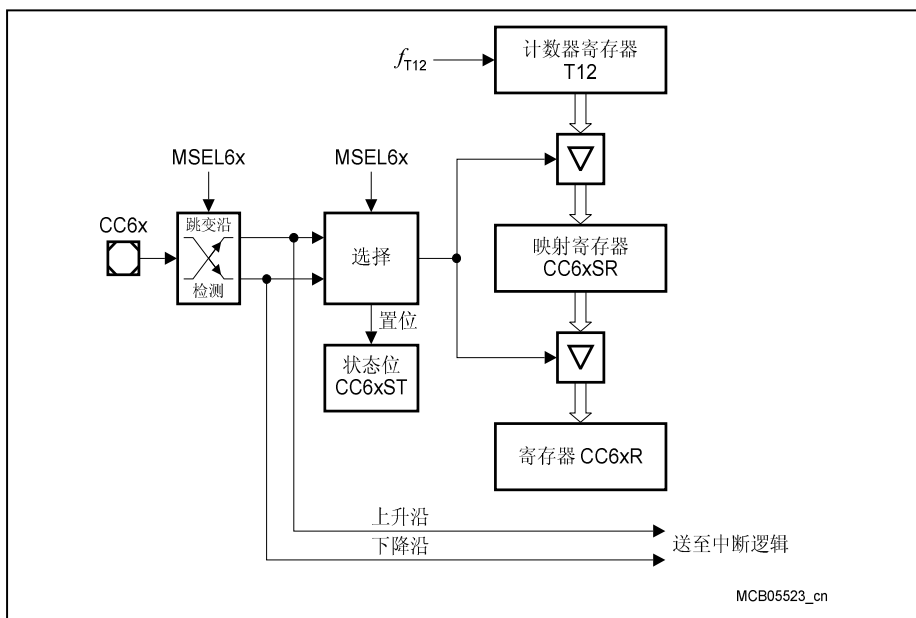


图 18-19 捕获模式 2、3、4 框图

捕获/比较单元 6 (CAPCOM6)

此外，还有五种多输入捕获模式，使用两个外部信号引脚 CC6x 和 CC6POSx 触发捕获。

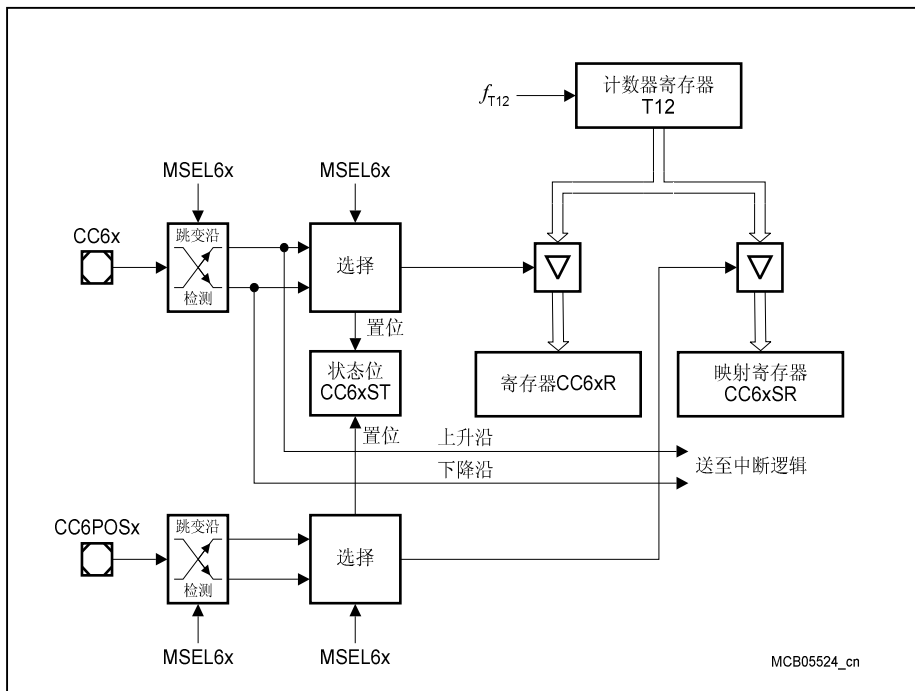


图 18-20 多输入捕获模式框图

多输入捕获模式下，引脚 CC6x 上选定的事件触发 T12 的当前值锁存到寄存器 CC6xR 中；引脚 CC6POSx 上选定的事件触发 T12 的当前值锁存到寄存器 CC6xSR 中。触发事件包括：两个输入引脚上相反的跳变沿、相同的跳变沿、或任意跳变沿。触发事件选择见表 18-4。

各种不同的捕获模式下，当引脚 CC6x 或 CC6POSx 上发生选定的捕获触发事件时，通道状态位 CC6xST 被置 1。该通道状态位必须由软件清零。

此外，引脚 CC6x 上检测到的所有跳变沿均可激活中断请求线，向 CPU 发送中断请求。（请参阅章节 18.9）。

表 18-4 多输入捕获模式总览

MSEL6x	模式	引脚	有效沿	T12 保存到
1010 _B	5	CC6x	上升沿	CC6xR
		CCPOSx	下降沿	CC6xSR
1011 _B	6	CC6x	下降沿	CC6xR
		CCPOSx	上升沿	CC6xSR
1100 _B	7	CC6x	上升沿	CC6xR
		CCPOSx	上升沿	CC6xSR
1101 _B	8	CC6x	下降沿	CC6xR
		CCPOSx	下降沿	CC6xSR
1110 _B	9	CC6x	任意沿	CC6xR
		CCPOSx	任意沿	CC6xSR
1111 _B	-	保留（无捕获或比较操作）		

18.1.5 类磁滞控制模式

类磁滞控制模式（MSEL6x = 1001_B）提供了这样的可能性：如果输入 CC6POSx 为 0，通过复位 CC6xST 可切断 PWM 输出。利用该模式可实现简单的电机控制，例如用比较器指示过流。当 CC6POSx = 0 时，相应通道的 PWM 输出驱动其被动电平。只有当 CC6POSx = 1 时，才可能置位 CC6xST。

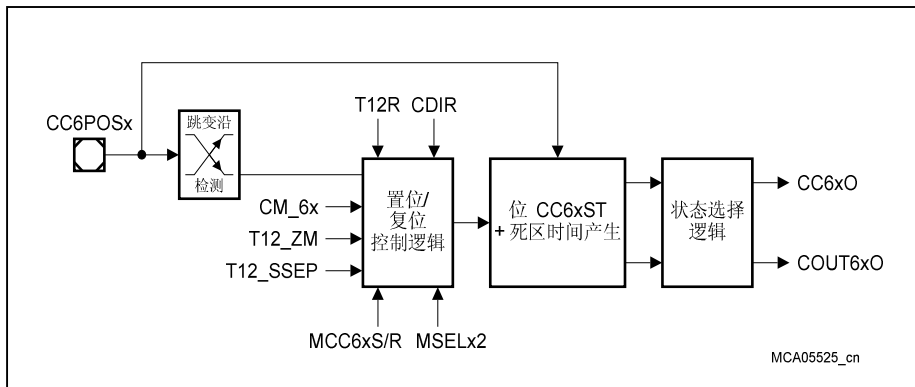


图 18-21 类磁滞控制模式逻辑框图

该模式下，引脚 CC6POSx 发生负跳变时状态位 CC6xST 被复位。只要输入 CC6POSx 为 0，状态位的输出始终为被动态。当 CC6POSx 处于高电平时，输出可进入主动态，由位 CC6xST 决定（见图 18-10 状态位逻辑和图 18-15 输出路径）。

该模式可将和时间相关的控制行为引入类磁滞控制器。标准的类磁滞控制器检测某值是否已超过极限，并根据比较结果切换输出。根据不同运行条件，切换频率和占空比不固定、可不断改变。

若要实现基于类磁滞控制器（控制内环）的时间相关控制环路（控制外环），如果外部环路与内环同步，将会表现出更好的行为特性。因此，可采用本产品提供的类磁滞模式，它结合了按时间切换的控制和类磁滞控制两种功能。例如，在该模式，可在某个固定时刻接通输出；一旦检测到输入引脚 CC6POSx 的下降沿，则立刻切断输出。

类磁滞模式可用于产生带有过流保护的标准 PWM。只要引脚 CC6POSx 上不出现低电平，则以正常方式产生输出信号（先前章节已作说明）；只有当引脚 CC6POSx 上出现低电平，如检测到过流时 CC6POSx 被拉低，才切断输出以避免损坏系统。

18.2 定时器 T13 模块

定时器 T13 和定时器 T12 的唯一差别在于 T13 只有一路比较通道。一个 16 位的递增计数器通过比较器和通道寄存器相连，计数器的值和通道寄存器的值匹配时，输出匹配信号。T13 模块具有多种控制功能，使其适用于不同的应用需求。此外，T13 可由定时器 T12 的事件同步触发。

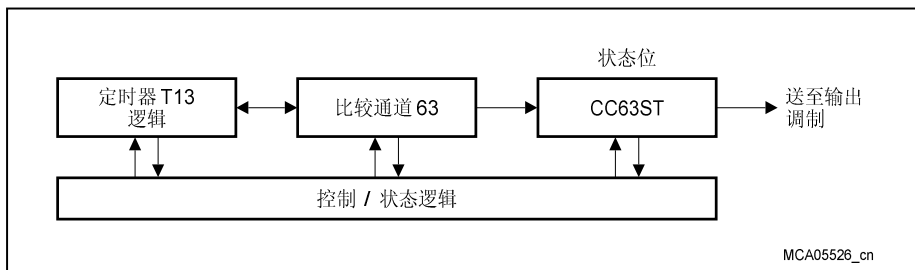


图 18-22 定时器 T13 模块总览框图

图 18-23 为定时器 T13 的内部结构框图。定时器 T13 的输入时钟 f_{T13} 来自经过预分频处理的模块时钟 f_{CC6} （预分频因子可编程设定）和 1/256 分频（可选）。T13 只能递增计数（类似于 T12 的边沿对齐模式）。

T13 通过比较器和周期寄存器 T13PR 相连。该寄存器中存放的值决定 T13 的最大计数值。T13 计数到周期值后，在下一个 T13 的时钟沿，产生信号 T13_PM（T13 周期匹配）并将 T13 复位为 0000_H。周期寄存器的新周期值从映射周期寄存器 T13PS 中获取，T13PS 的值由软件载入。“T13 映射传送”控制信号 T13_ST 控制将新周期值从映射寄存器传送到 T13PR 中的操作，该信号的产生取决于控制位 STE13。CAPCOM6 单元提供了周期值、以及（和产生 PWM 信号相关的）其它数值的映射寄存器，便于用户编程同步更新所有相关参数（请参阅**章节 18.8**）。

信号 T13_ZM 指示计数器的值是否等于 0000_H。

在当前计数周期结束后，单次模式控制位 T13SSC 控制定时器的自动停止（见**图 18-25**）。

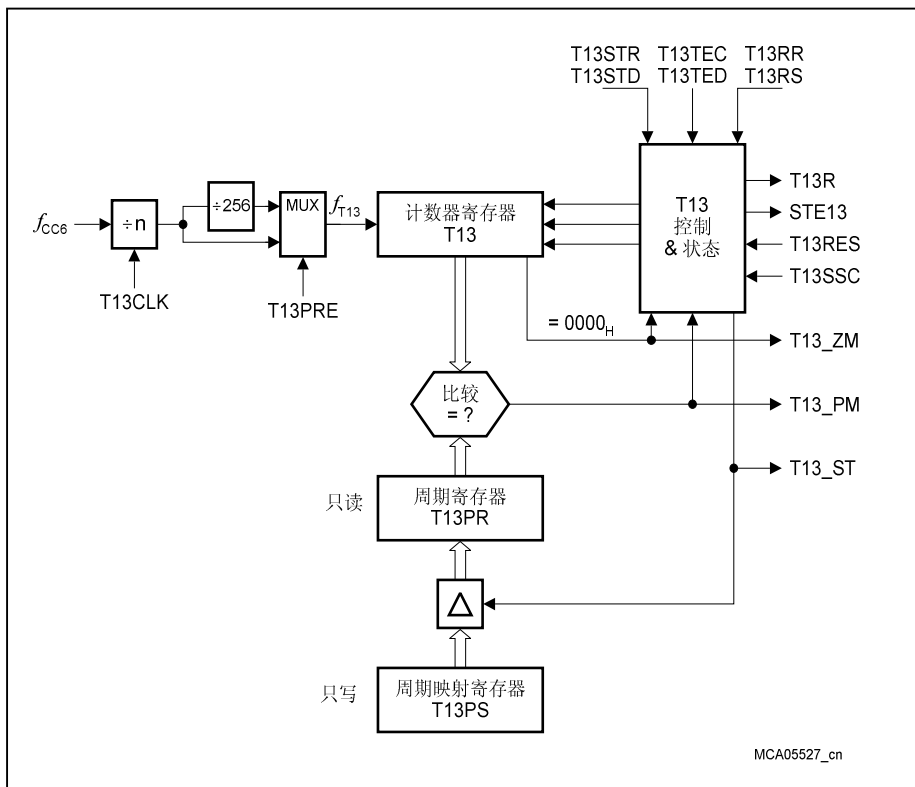


图 18-23 T13 计数器逻辑和周期比较器

运行位 T13R 控制定时器 T13 的启动或终止。T13R 可由置位控制位 T13RS 或复位控制位 T13RR 软件设定；或根据预先选定的条件由硬件复位。

定时器 T13 可由控制位 T13RES 清零。置位“只写”控制位 T13RES 只清零定时器的内容、不会产生其它影响，如不会终止定时器工作。

T13 映射传送控制信号 T13_ST 由位 STE13 控制产生，该控制位可通过相关置位控制位 T13STR，或复位控制位 T13STD 间接设置。

位域 T13TEC 和 T13TED 控制 T13 与定时器 T12 的事件同步。T13TEC 选择触发事件；T13TED 选择使触发事件生效的定时器 T12 的计数方向。

注：T13 的周期寄存器和对应的映射寄存器占用相同的物理地址。写操作对应写入映射寄存器；读操作对应读取实际周期寄存器的值。

捕获/比较单元 6（CAPCOM6）

寄存器 **T13** 的内容显示了当前定时器 **T13** 的计数值。只有在定时器 **T13** 停止工作时，才可对寄存器 **T13** 进行写操作；定时器 **T13** 工作时，写操作无效。寄存器 **T13** 始终可由软件读取。

CCU6_T13

定时器 T13 计数寄存器

XSFR (E8B0H/--)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T13CV															
rwh															

符号	位序号	读写类型	功能描述
T13CV	[15:0]	rwh	定时器 T13 的计数值 定时器 T13 的 16 位计数值。

寄存器 **T13PR** 中存放着定时器 **T13** 的周期值。周期值和 **T13** 的当前计数值进行比较，两值匹配时 **T13** 被复位。该寄存器对应有一个映射寄存器，位 **STE13** 控制映射传送。读操作读取实际周期寄存器的值；写操作写入映射寄存器；映射寄存器结构便于用户同时更新 **T13** 的所有相关值。

CCU6_T13PR

定时器 T13 周期寄存器

XSFR (E8B2H/--)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T13PV															
rwh															

符号	位序号	读写类型	功能描述
T13PV	[15:0]	rwh	定时器 T13 的周期值 T13 的计数值和 T13PV 相等时触发周期匹配。计数到该周期值，定时器 T13 被清零。

18.2.1 定时器 T13 的操作

定时器 T13 的输入时钟 f_{T13} 来自经过预分频处理的模块时钟 f_{CC6} （预分频因子可编程设定）和 1/256 分频（可选）。预分频因子列于表 18-5 中。定时器 T13 不工作时预分频器复位，从而保证产生可重复的时序和延迟。

表 18-5 定时器 T13 输入时钟选择

T13CLK	输入时钟 预分频器关闭（T13PRE = 0）	输入时钟 预分频器开启（T13PRE = 1）
000 _B	f_{CC6}	$f_{CC6}/256$
001 _B	$f_{CC6}/2$	$f_{CC6}/512$
010 _B	$f_{CC6}/4$	$f_{CC6}/1024$
011 _B	$f_{CC6}/8$	$f_{CC6}/2048$
100 _B	$f_{CC6}/16$	$f_{CC6}/4096$
101 _B	$f_{CC6}/32$	$f_{CC6}/8192$
110 _B	$f_{CC6}/64$	$f_{CC6}/16384$
111 _B	$f_{CC6}/128$	$f_{CC6}/32768$

定时器周期由周期寄存器 T13PR 中的值决定，计算公式如下：

$$T13_{PER} = \langle \text{周期寄存器的值} \rangle + 1; \text{以 } T13 \text{ 时钟频率} (f_{T13}) \text{ 计数} \quad (18.3)$$

定时器 T13 工作时，写入计数寄存器 T13 无效。若 T13 停止计数，写入寄存器 T13 的值立刻生效。

上面已提到，T13 只能递增计数，类似于 T12 的边沿对齐模式，因此 T13 计数器的“计数规则”非常简单：

- 检测到周期匹配时，下一个 T13 时钟沿计数器复位为 0。T13 始终递增计数。

T13 的操作如图 18-24 所示。

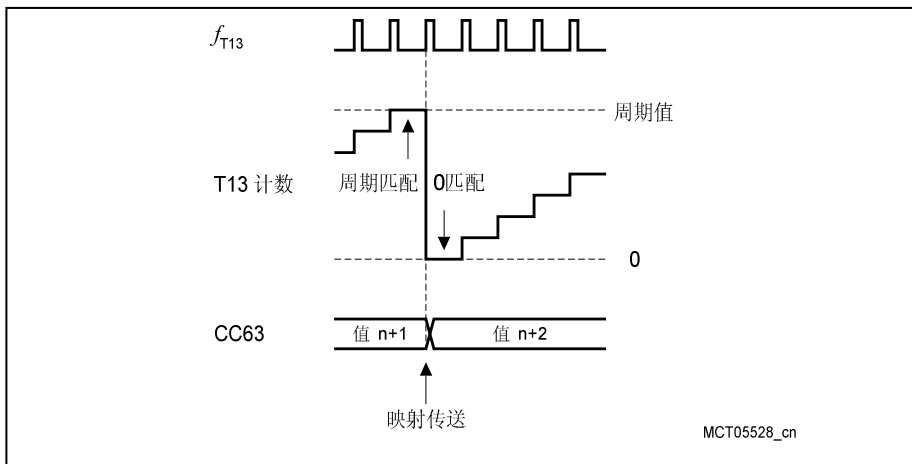


图 18-24 T13 的操作

T13 映射传送信号, T13_ST

采用一个特殊的映射传送信号 (T13_ST) 便于使周期值、比较值的更新和 T13 的操作同步进行。该信号的产生可通过设置位 STE13 来请求 (“只写” 位 T13STR 写入 1 将置位 STE13; “只写” 位 T13STD 写入 1 将清零 STE13)。

若请求映射传送 (STE13 = 1), 以下情况会产生信号 T13_ST:

- 递增计数到周期匹配后计数器复位, 或
- 定时器 T13 停止计数 (T13R = 0)

当信号 T13_ST 有效, 映射寄存器中的新周期值传送到实际周期寄存器 T13PR 中; 映射寄存器中的新比较值传送到实际比较寄存器中 (请参阅[章节 18.1.2](#))。

T13 启动/停止和复位控制

“只写” 位 T13RS 置 1 时启动定时器 T13, 该操作置位定时器运行位 T13R, 定时器开始计数; 要终止 T13 运行, “只写” 位 T13RR 置 1, 该操作对定时器运行位 T13R 清零, 定时器停止计数。

单次模式下也可由硬件清零运行位 T13R。此外, T12 模块产生的触发事件 (置位运行位 T13R) 可启动 T13。下一章节将对此进行详细说明。

“只写” 位 T13RES 置 1 将对定时器 T13 清零。该操作只清零定时器的内容 (至 0000H)、不会引发其它操作, 比如不会清零定时器的运行位。

单次模式

单次模式下，运行位 T13R 也受硬件影响。通过位 T13SSC 使能该模式。T13SSC 被置位时，当前定时器周期结束之后定时器停止运行（见图 18-25）。定时器计数到周期值后被清零至 0000_H，计数周期结束。

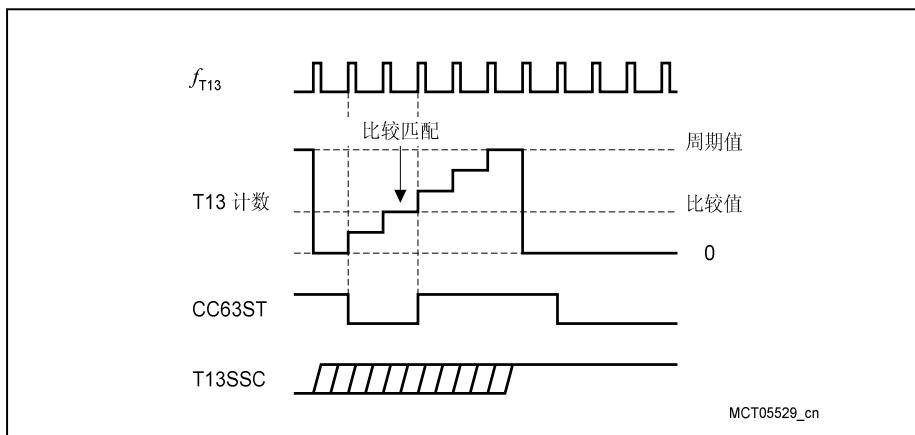


图 18-25 定时器 T13 的单次操作

T13 和 T12 同步

定时器 T13 可与 T12 事件同步。位域 T13TEC 和 T13TED 选择触发事件，启动定时器 T13 工作。触发事件硬件置位 T13R，T13 开始计数。该特性和单次模式相结合，可在 T12 事件之后产生可编程的时延。

图 18-26 举例说明 T13 如何与 T12 的事件同步，图中所选择的事件为递增计数时的比较匹配（比较值 = 2）。T12 和 T13 的输入时钟可以不同（选择其它预分频因子），图中 f_{T13} 为 f_{T12} 的一半。

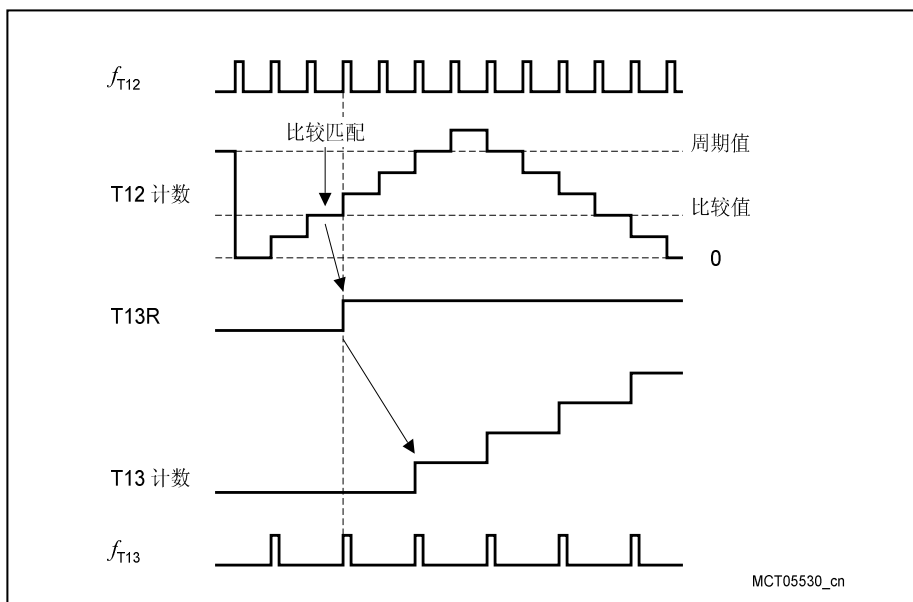


图 18-26 T13 和 T12 同步

位域 T13TEC 选择启动 T13 的触发事件（自动置位 T13R 以与 T12 的比较信号同步），事件选择列于表 18-6 中；另外由 T13TED 选择使触发事件生效的定时器 T12 的计数方向（见表 18-7）。

表 18-6 T12 触发事件选择

T13TEC	事件选择
000 _B	无
001 _B	通道 0 上的 T12 比较事件
010 _B	通道 1 上的 T12 比较事件
011 _B	通道 2 上的 T12 比较事件
100 _B	任意通道（0、1、2）上的 T12 比较事件
101 _B	T12 周期匹配
110 _B	T12 递增计数时 0-匹配
111 _B	霍尔状态的任何变化

表 18-7 T12 触发事件附加条件

T13TED	事件选择附加条件
00 _B	保留，无操作
01 _B	T12 递增计数时选定的触发事件有效
10 _B	T12 递减计数时选定事件的触发有效
11 _B	无论 T12 递增/递减计数，选定的触发事件均有效

18.2.2 T13 的比较模式

定时器 T13 的单路比较通道可执行（和 T13 的计数值有关的）比较操作。

图 18-27 给出比较模式下 T13 的通道操作。T13 通道通过相等比较器和 T13 的计数寄存器相连，当计数器的值和比较寄存器中的值匹配时产生匹配信号。通道由比较器和双寄存器结构（实际比较寄存器 CC63R 和相关映射寄存器 CC63SR）组成。CC63R 与比较器直接相连、CC63SR 由软件预先加载。当 T13 的映射传输信号 T13_ST 有效时，映射寄存器中的内容被传送到实际比较寄存器中。CAPCOM6 单元提供了比较值、以及（和产生 PWM 信号相关的）其它数值的映射寄存器，便于用户编程同步更新所有相关参数（请参阅**章节 18.8**）。

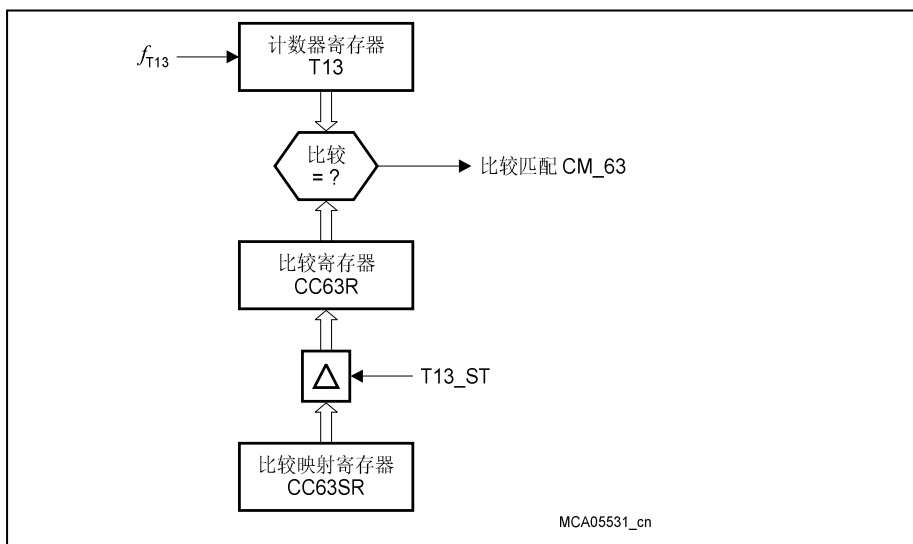


图 18-27 T13 通道比较

T13 通道状态位 CC63ST 保存比较操作的状态。满足某些条件时该状态位将被置位或复位，以下章节将对此进行详细说明。**图 18-28** 给出比较状态位的逻辑操作。

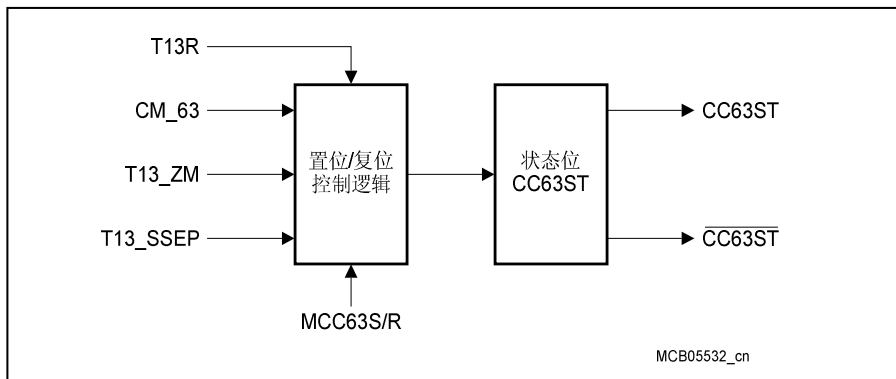


图 18-28 T13 状态位框图

状态位 CC63ST 置位/复位控制逻辑的输入包括：定时器运行位（T13R），定时器 0-匹配信号（T13_ZM），单次模式结束信号（T13_SSEP）以及实际比较匹配信号 CM_63。此外，可通过设定置位控制位 MCC63S 和复位控制位 MCC63R 相应置位或复位 CC63ST。

只有当定时器 T13 运行时（T13R = 1）才可硬件修改状态位 CC63ST。这种情况下，应用以下规则置位和复位状态位：

状态位 CC63ST 被置 1：

- 比较匹配时（T13 始终递增计数），下一个 T13 时钟（ f_{T13} ）置位 CC63ST（即，计数器递增计数到超过比较值）；
- 0-匹配且同时为比较匹配时，下一个 T13 时钟（ f_{T13} ）置位 CC63ST。

状态位 CC63ST 被复位为 0：

- 0-匹配但同时不是比较匹配时，下一个 T13 时钟（ f_{T13} ）复位 CC63ST。

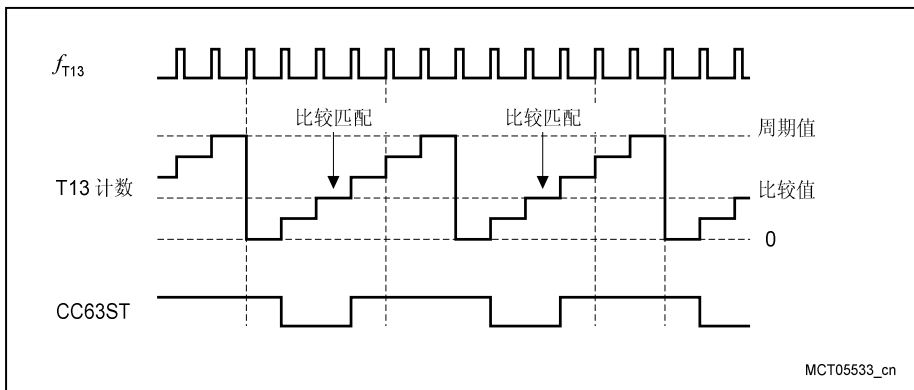


图 18-29 T13 的比较操作

注: 图 18-29 所示波形和图 18-11 所示 T12 的波形相同。

T13 比较模式输出路径

图 18-30 给出从通道状态位到其输出引脚最简信号路径。如图所示，用户可通过多种控制选择，决定当前状态位 **CC63ST** 所对应的（期望）输出信号的切换操作。输出调制控制的具体内容请参阅**章节 18.7**。

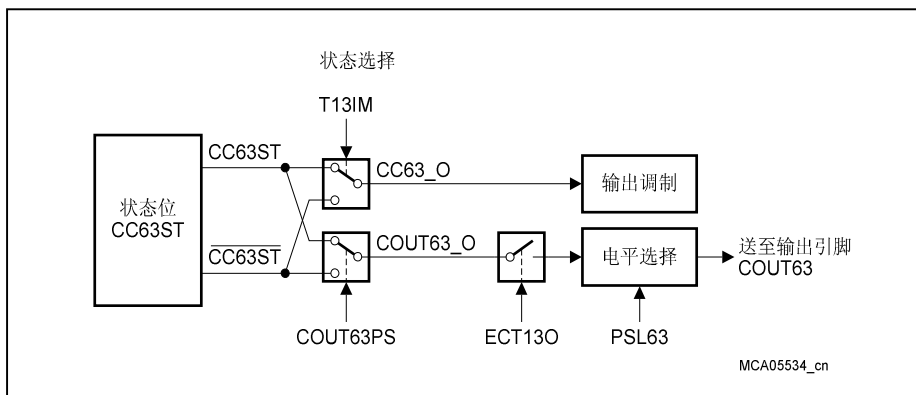


图 18-30 CC63 输出路径

输出线 **COUT63_O** 可产生 T13 的 PWM 信号，从引脚 **COUT63** 输出。信号 **CC63_O** 可输出 T13 的 PWM 信号用来调制 T12 的相关输出信号。为了使 **COUT63** 和内部调制信号无关，可分别由 **T13IM** 和 **COUT63PS** 选择引发有效信号的比较状态。

T13 比较模式寄存器

寄存器 CC63R 是 T13 的实际比较寄存器。保存在 CC63R 中的值和 T13 的计数值进行比较。寄存器 CC63R 为只读寄存器；若要改变 CC63R 的值，将映射寄存器 CC63SR 中的修改值映射传送到 CC63R 中。映射寄存器可由软件读写。

CCU6_CC63R

通道 3 比较寄存器

XSFR（E8B4_H/--）

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC63V															
rh															

符号	位序号	读写类型	功能描述
CC63V	[15:0]	rh	通道 3 的比较值 位域 CC63V 中存放着和 T13 的计数值进行比较的值。

CCU6_CC63SR

通道 3 比较映射寄存器

XSFR（E8B6_H/--）

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC63S															
rwh															

符号	位序号	读写类型	功能描述
CC63S	[15:0]	rwh	通道 3 比较值映射寄存器 映射传送期间，位域 CC63S 的内容被传送到寄存器 CC63R 的位域 CC63V 中。

18.3 定时器模块控制

定时器 T12 和 T13 的大多数特性由定时器控制寄存器 TCTR0、TCTR2 和 TCTR4 控制。

寄存器 TCTR0 控制定时器 T12 和 T13 的基本功能。

CCU6_TCTR0

定时器控制寄存器 0

XSFR (E8AC_H/--)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	STE 13	T13R	T13 PRE	T13CLK			CTM	CDIR	STE 12	T12R	T12 PRE	T12CLK		
-	-	rh	rh	rw	rw			rw	rh	rh	rh	rw	rw		

符号	位序号	读写类型	功能描述
STE13¹⁾	13	rh	定时器 T13 映射传送使能 0 映射寄存器传送被禁止 1 映射寄存器传送被使能
T13R²⁾	12	rh	定时器 T13 运行控制位 T13R 启动和终止定时器 T13。软件置位 T13RS 可置位 T13R，软件置位 T13RR 可复位 T13R；或根据位域 T13SSC 定义的功能硬件复位 T13R。 0 定时器 T13 被终止 1 定时器 T13 在运行
T13PRE	11	rw	定时器 T13 预分频使能位 使能 T13 的附加 1/256 预分频器。 0 附加预分频被禁止 1 附加预分频被使能
T13CLK	[10:8]	rw	定时器 T13 输入时钟选择 定时器 T13 的输入时钟由 CAPCOM6 单元的输入时钟 f_{CC6} 分频得到，根据等式 $f_{T13} = f_{CC6} / 2^{<T13CLK>}$ ，选择定时器 T13 的输入时钟，见表 18-5。

符号	位序号	读写类型	功能描述
CTM	7	rw	T12 的工作模式 0 边沿对齐模式 1 中间对齐模式
CDIR	6	rh	定时器 T12 的计数方向 指示定时器 T12 的当前计数方向。 0 T12 递增计数 1 T12 递减计数
STE12¹⁾	5	rh	定时器 T12 映射传送使能 0 不请求映射寄存器传送 1 请求映射寄存器传送
T12R²⁾	4	rh	定时器 T12 运行控制位 T12R 启动和终止定时器 T12。软件置位 T12RS 可置位 T12R，软件置位 T12RR 可复位 T12R；或根据位域 T12SSC 定义的功能硬件复位 T12R。 0 定时器 T12 被终止 1 定时器 T12 在运行
T12PRE	3	rw	定时器 T12 预分频使能位 使能 T12 的附加 1/256 预分频器。 0 附加预分频被禁止 1 附加预分频被使能
T12CLK	[2:0]	rw	定时器 T12 输入时钟选择 定时器 T12 的输入时钟由 CAPCOM6 单元的输入时钟 f_{CC6} 分频得到根据等式 $f_{T12} = f_{CC6} / 2^{<T12CLK>}$ ，选择定时器 T12 的输入时钟。见 表 18-1 。

1) 发生映射传送时，位 STE12/STE13 被清零。

2) 由 TxSSC, TxRR 或 TxRS) 同时置位/复位 T13R/T12R 无效，位 T13R/T12R 保持不变。

注：只有当定时器 T12/T13 不工作时（T12R/T13R = 0），对位域 T12CLK/T13CLK 或 T12PRE/T13PRE 的写入才有效。

寄存器 TCTR2 控制定时器 T12 和 T13 的单次模式和同步功能。

CCU6_TCTR2

定时器控制寄存器 2

XSFR (E8AE_H/--)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	-	T13 TED		T13 TEC		T13 SSC	T12 SSC	
-	-	-	-	-	-	-	-	-	-	rW		rW		rW	rW

符号	位序号	读写类型	功能描述
T13TED	[6:5]	rW	触发事件计数方向控制 位域 T13TED 附加规定了使选定的触发事件生效的 T12 计数方向（见表 18-7）。
T13TEC	[4:2]	rW	触发事件选择 位域 T13TEC 选择启动 T13 的触发事件（自动置位 T13R 以与 T12 的比较信号同步），见表 18-6。
T13SSC T12SSC	1 0	rW	定时器 T13/T12 单次模式控制 该位控制 T13/T12 的单次模式。 0 单次模式被禁止 1 单次模式被使能

捕获/比较单元 6（CAPCOM6）

寄存器 **TCTR4** 支持软件控制（独立的置位和复位条件）运行位 **T12R** 和 **T13R**。此外，定时器（运行中）可被复位；位 **STE12** 和 **STE13** 可由软件控制，读取这些位始终返回 0。

CCU6_TCTR4

定时器控制寄存器 4

XSFR (E8A6H/--)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T13 STD	T13 STR	-	-	-	T13 RES	T13 RS	T13 RR	T12 STD	T12 STR	-	-	DT RES	T12 RES	T12 RS	T12 RR
w	w	-	-	-	w	w	w	w	w	-	-	w	w	w	w

符号	位序号	读写类型	功能描述
T13STD T12STD	15 7	w	定时器 T13/T12 映射传送禁止 0 无操作 1 STE13/STE12 被复位，不触发映射传送
T13STR T12STR	14 6	w	定时器 T13/T12 映射传送请求 0 无操作 1 STE13/STE12 被置位，请求映射传送
T13RES T12RES	10 2	w	定时器 T13/T12 复位 0 对 T13/T12 无复位操作 1 T13/T12 计数寄存器被复位为 0。根据切换规则切换输出信号。置位 T13RES/T12RES 不影响位 T13R/T12R 。
T13RS T12RS	9 1	w	定时器 T13/T12 运行位置位控制 ¹⁾ 软件置位 T13RS/T12RS 可相应置位 T13R/T12R （启动定时器 T13/T12 ）。 0 T13R/T12R 未被置位 1 T13R/T12R 被置位， T13/T12 开始计数
T13RR T12RR	8 0	w	定时器 T13/T12 运行位复位控制 ¹⁾ 软件复位 T13RR/T12RR 可相应清零 T13R/T12R （终止定时器 T13/T12 ）。 0 T13R/T12R 未被清零

捕获/比较单元 6（CAPCOM6）

符号	位序号	读写类型	功能描述
			1 T13R/T12R 被清零，T13/T12 停止计数
DTRES	3	w	死区时间计数器复位 0 对死区时间计数器无复位操作 1 三个死区时间计数器均被清零、停止计数

1) 同时设置置位控制位和复位控制位无效，相关定时器操作不受影响。

捕获/比较单元 6 (CAPCOM6)

通道状态寄存器 CMPSTAT 中的状态位指示当前的捕获或比较状态，控制位决定比较通道对应主动态还是被动态。

CCU6_CMPSTAT

比较状态寄存器

XSFR (E8A8H/--)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T13 IM	C OUT 63PS	C OUT 62PS	CC 62PS	C OUT 61PS	CC 61PS	C OUT 60PS	CC 60PS	-	CC 63ST	CC POS 2	CC POS 1	CC POS 0	CC 62ST	CC 61ST	CC 60ST
rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	-	rh	rh	rh	rh	rh	rh	rh

符号	位序号	读写类型	功能描述
T13IM	15	rwh	T13 反相调制控制 位 T13IM 将 CC63_O 信号反相，用于调制 CC6x 和 COUT6x (x = 0, 1, 2) 信号。 0 CC63_O 未反相 1 CC63_O 被反相用于进一步调制
COUT63PS¹⁾ COUT62PS CC62PS COUT61PS CC61PS COUT60PS CC60PS	14 13 12 11 10 9 8	rwh	比较输出的被动态选择 位 COUT6xPS/CC6xPS 选择比较通道的被动态。输出处于被动态时，输出引脚驱动（由寄存器 PSLR 定义的）被动电平。 0 CC6xST 为 0 时比较输出驱动被动电平 1 CC6xST 为 1 时比较输出驱动被动电平 <i>注：捕获模式下不使用这些位。</i>
CC63ST²⁾ CC62ST CC61ST CC60ST	6 2 1 0	rh	T13/T12 比较状态位 位 CC6xST 监控捕获/比较通道的状态。 比较模式： 0 定时器计数值小于比较值 1 定时器计数值大于比较值 捕获模式（仅通道 0 ...2）： 0 未检测到选定的跳变沿 1 已检测到选定的跳变沿
CCPOSx	5, 4, 3	rh	霍尔序列采样值

捕获/比较单元 6（CAPCOM6）

- 1) 这些位具有映射位，分别与 T12 和 T13 的捕获/比较寄存器并行更新。读操作读取实际使用值，写操作写入映射位。
- 2) 根据 T12 和 T13 的切换规则置位和复位这些状态位。

比较状态修改寄存器 **CMPMODIF** 支持软件修改（独立的置位和清零条件）通道状态位 **CC6xST**。该特性允许用户编程单独修改输出线的状态，比如比较定时器停止计数时，用户可自行修改相应的比较状态。

CCU6_CMPMODIF

比较状态修改寄存器

XSFR (E8AA_H/--)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	MCC 63R	-	-	-	MCC 62R	MCC 61R	MCC 60R	-	MCC 63S	-	-	-	MCC 62S	MCC 61S	MCC 60S
-	W	-	-	-	W	W	W	-	W	-	-	-	W	W	W

符号	位序号	读写类型	功能描述
MCC63R	14	w	捕获/比较通道 x 状态复位位 0 无操作 1 位 CC6xST 被清零 ¹⁾
MCC62R	10		
MCC61R	9		
MCC60R	8		
MCC63S	6	w	捕获/比较通道 x 状态置位位 0 无操作 1 位 CC6xST 被置位 ¹⁾
MCC62S	2		
MCC61S	1		
MCC60S	0		

1) 同时设置置位控制位和复位控制位将使相关状态位翻转。

18.4 多通道模式

多通道模式可用一条指令调制全部六路 T12 的相关输出信号。位域 MCMOUT.MCMP 中的位规定哪些输出有效。若多通道模式被使能（位 MODCTR.MCMEN = 1），只有位域 MCMP 中被置 1 的位对应的输出才有效。

该位域对应的映射位域为 MCMPS，可由软件写入。可由 T12 或 T13 事件触发将 MCMPS 中的新值传送到位域 MCMP 中，该映射传送操作和 T12 或 T13 事件同步。该结构支持由软件写入新值，在定义好的时刻发生映射传送，使映射传送和 PWM 信号同步。这样可避免由不同步的调制源（T12，T13，软件）造成的多余脉冲。

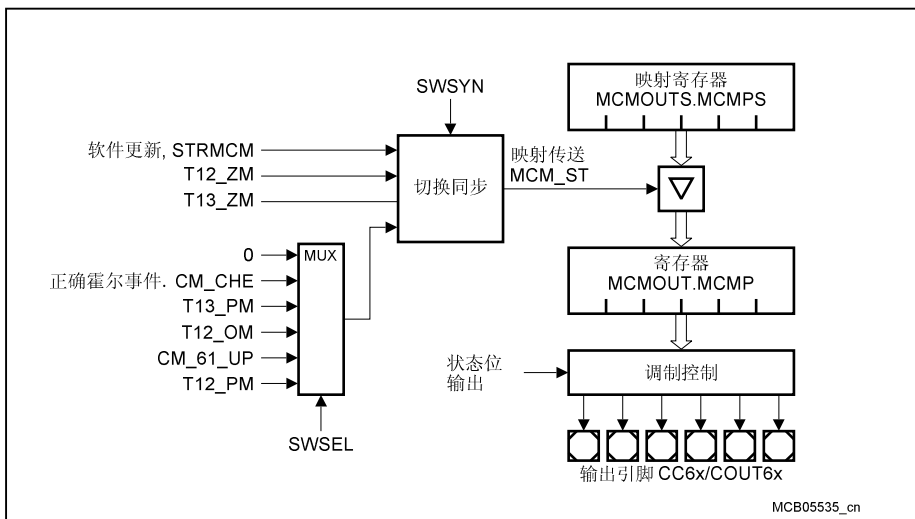


图 18-31 多通道模式框图

图 18-31 所示为多通道模式的调制选择。位域 SWSEL 选择更新位域 MCMP 的触发事件。为了使 MCMP 更新和 T12 或 T13 产生的 PWM 信号同步，由位域 SWSYN 选择引发映射传送的同步事件。该逻辑结构决定了产生新的 PWM 周期时更新 MCMP。选定的触发事件发生时，提示标志 R 被置位；发生映射传送时 R 被复位。软件监控该标志以检查该逻辑的状态。

若选择直接同步模式，当选定的触发事件发生时立刻更新 MCMP。也可由软件请求更新 MCMP，映射传送请求位 STRMCM 被置位时，写入位域 MCMPS 的值会立刻更新 MCMP。若使用直接模式和位 STRMCM，则 MCMP 的更新完全由软件控制。触发事件选择和同步事件选择归纳于表 18-8 和表 18-9 中。

表 18-8 多通道模式触发事件选择

SWSEL	事件选择（见寄存器 CCU6_MCMCTR） ¹⁾
000 _B	无
001 _B	引脚 CCPOS6x 上的正确霍尔事件（CM_CHE）
010 _B	T13 周期匹配（T13_PM）
011 _B	T12 递减计数时 1-匹配（T12_OM）
100 _B	T12 递增计数时通道 1 的比较事件（CM_61_UP；见 章节 18.5 和 图 18-34 ）。相位延迟功能
101 _B	T12 递增计数时周期匹配（T12_PM）
11x _B	保留，无操作

1) 始终可由软件控制。

表 18-9 多通道模式触发事件同步

SWSYN	同步事件（见寄存器 CCU6_MCMCTR）
00 _B	直接模式：触发事件直接引发映射传送
01 _B	T13 的 0-匹配
10 _B	T12 递增计数时的 0-匹配
11 _B	保留，无操作

捕获/比较单元 6（CAPCOM6）

寄存器 MODCTR 控制由定时器 T12 和 T13 产生的 PWM 序列调制相应的输出信号。此外，多通道模式可被使能作为输出信号的附加调制源。

CCU6_MODCTR

调制控制寄存器

XSFR (E8C0H/--)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ECT130	-	T13MODEN					MCMEN	-	T12MODEN						
rw	-	rw					rw	-	rw						

符号	位序号	读写类型	功能描述
ECT130	15	rw	定时器 T13 比较输出使能 0 信号 COUT63 强制为 0 1 信号 COUT63 根据设定输出
T13MODEN T12MODEN	[13:8] [5:0]	rw	定时器 T13/T12 调制使能 该位域使能由 T13 或 T12 产生的 PWM 序列调制相应的比较通道。 0 输出不经 T13/T12 调制 1 输出由 T13/T12 的 PWM 序列调制 T13MODEN[5:0]和 T12MODEN[5:0]（从左至右）分别对应： COUT62, CC62, COUT61, CC61, COUT60, CC60
MCMEN	7	rw	多通道模式使能 该位使能由多通道序列调制输出信号（MCMP 设定的有效输出）。 0 调制被禁止 1 调制被使能

注：寄存器 MCMOUT, MCMOUTS 和 MCMCTR 也控制多通道模式的操作。

18.5 霍尔传感器模式

对于无刷直流电机，通常使用多通道模式产生调制序列以控制电机正确工作。调制序列必须根据电机的角度位置相应产生。通常使用霍尔传感器或反电动势检测来决定转子的角度位置。CAPCOM6 的三个输入，CC6POS0...CC6POS2，可用作霍尔传感器或反电动势检测信号的输入。

电机转动位置和输出调制序列之间紧密关联。当电机转动到某个位置时，由霍尔传感器的输入采样（霍尔序列）表示当前位置，接下来必须输出预先确定的调制序列。电机类型不同，驱动电机的调制序列有所不同。因此，希望能够比较灵活的定义霍尔序列和相应调制序列之间的关系。

CAPCOM6 通过特定的寄存器实现了该特性，该寄存器中存放着当前实际的霍尔序列（CURH）、下次期望的霍尔序列（EXPH）、以及相应的输出序列（MCMP）。当采样的霍尔序列和期望序列（EXPH）匹配时，输出新的调制序列。CAPCOM6 监控霍尔输入的变化以检测电机下一次的转动相位（块切换的某段，两两导通的某个状态）。检测到下一个期望的霍尔序列时，输出下一个相应的调制序列。

CAPCOM6 引入霍尔输入的采样时延，从而可（在一定程度上）抗噪声。此外，霍尔序列采样和当前霍尔序列（CURH）进行比较，从而可容忍短毛刺的存在。

霍尔传感器模式下，通过双寄存器结构控制霍尔和调制序列的产生。寄存器 MCMOUT 中存放着实际使用的值；映射寄存器 MCMOUTS 中存放着取自预定义表的正确的霍尔和调制序列（针对给定电机）。

当检测到正确的霍尔序列变化时，即霍尔序列采样和期望序列相匹配时，映射寄存器的内容映射传送到寄存器 MCMOUT 中。接着用户编程将新值装入 MCMOUTS 中。从 MCMOUTS 到 MCMOUT 的映射传送也可由软件强制执行。

18.5.1 霍尔序列比较逻辑

图 18-32 给出双寄存器结构和序列比较逻辑。软件将新调制序列（MCMPS）和当前霍尔序列（CURHS）和期望霍尔序列（EXPHS）写入映射寄存器 MCMOUTS 中。寄存器 MCMOUT 中存放着实际使用的值。调制序列 MCMPS 被送入输出调制控制模块。当前霍尔序列（CURH）和期望霍尔序列（EXPH）通过比较器和霍尔序列采样（引脚 CC6POSx 上的）进行比较。信号 HCRDY（霍尔比较就绪）控制霍尔输入采样和比较器输出评估，下一节将对此进行详细说明。

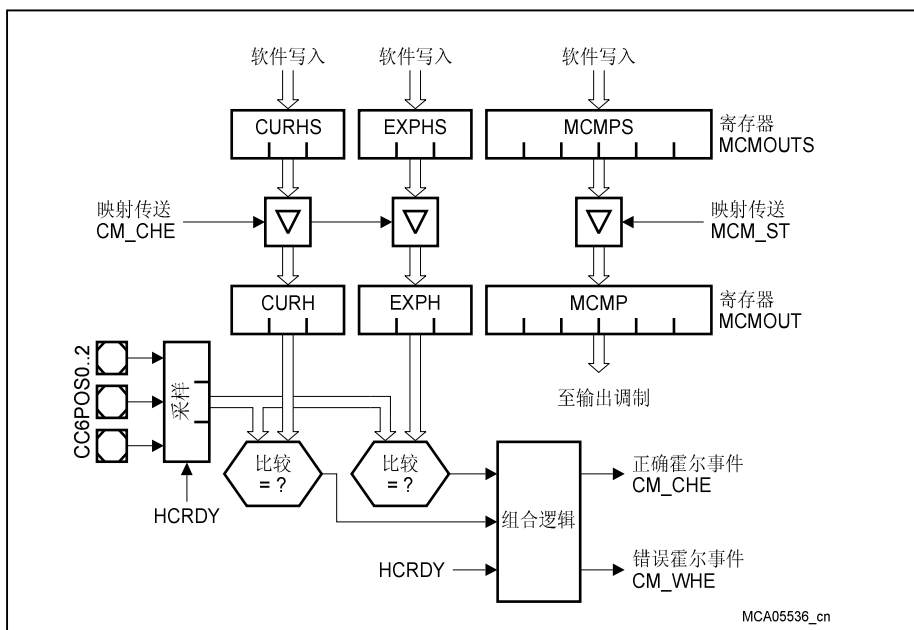


图 18-32 霍尔序列比较逻辑

当霍尔序列采样和期望序列（EXPH）匹配时，产生信号 CM_CHE（正确霍尔事件）。当霍尔序列采样和当前序列（CURH）匹配时不产生信号，这是由输入线上的毛刺引起的跳变（而非霍尔序列改变引起）。若霍尔序列采样既不和 EXPH 匹配、也不和 CURH 匹配，产生信号 CM_WHE，表明这是错误的霍尔事件。

每次产生正确的霍尔事件时（CM_CHE），将映射寄存器 MCMOUTS 中的新霍尔序列（下一个）传送到 MCMOUT 中，新的霍尔序列连同相应的输出序列（取自预定义表）可由软件写入 MCMOUTS 中。信号 MCM_ST 用来触发调制序列的映射传送。该信号可通过信号 CM_CHE 产生（见图 18-31）。将 MCMOUTS 中的位 STRHP 置 1，也

可实现映射传送。当使用相位延迟功能时（由 T12 的通道 1 产生），多通道模式映射传送 MCM_ST（由位 STR 指示）触发新序列载入。

18.5.2 霍尔序列采样

跳变沿检测模块以频率 f_{CC6} 监控霍尔传感器的输入（CC6POSx）。检测到（三个输入中）任意一个输入的电平改变时，产生输出信号。在恶劣的逆变器工作环境中，高 di/dt （电流变化过快）会引起毛刺，可用硬件噪声滤波器来抑制霍尔输入上的毛刺。

使用死区时间计数器 DTC0 实现噪声滤波，此时 T12 通道的模式控制位域 MSELx 必须全部设定为“1000_B”。跳变沿检测模块的输出信号触发 DTC0，DTC0 被重载并开始计数，从而产生一个时延。计数器计数至 1 时产生输出信号 DTC0_O，该信号用作输入采样和比较评估信号 HCRDY（见 图 18-32）。

该特性通过时延实现噪声滤波。大多数干扰，如开关噪声和信号振荡，可通过此方式消除。检测到输入信号跳变时，延迟一段时间之后对其采样，时延长度由 DTC0 的重载值决定。将采样序列和当前霍尔序列（CURH）、期望霍尔序列（EXPH）进行比较。若采样序列和当前序列（CURH）匹配，检测到的信号跳变由噪声毛刺引起（延迟之后消失），故不触发其它操作。若采样序列和期望序列（EXPH）相同，检测到的信号跳变为正确的霍尔事件，产生信号 CM_CHE 以触发其它操作。

但是，如果采样序列和 CURH、EXPH 都不匹配，检测到的信号跳变导致错误的霍尔序列，则产生信号 CM_WHE 以指明该错误并触发其它适当的操作。

图 18-33 为噪声滤波器的逻辑框图。

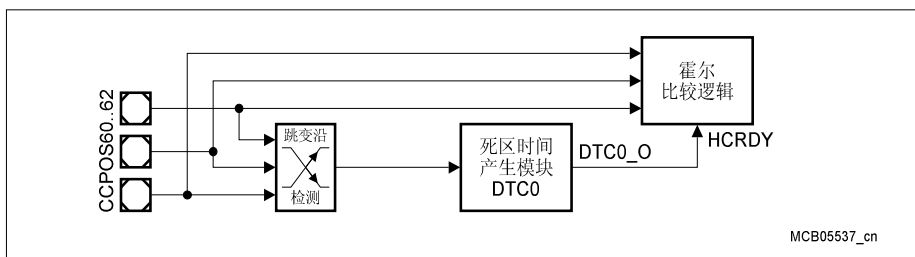


图 18-33 霍尔触发器逻辑框图

18.5.3 用定时器 T12 模块实现无刷直流电机控制

CAPCOM6 提供了一种便于无刷直流电机控制的专用模式，通过定时器 T12 模块实现。将 T12 三路通道的位域 MSELx 全部设定为 1000_B，选择该模式。

该模式下，如图 18-34 所示，通道 0 工作在捕获模式；通道 1 和 2 工作在比较模式。

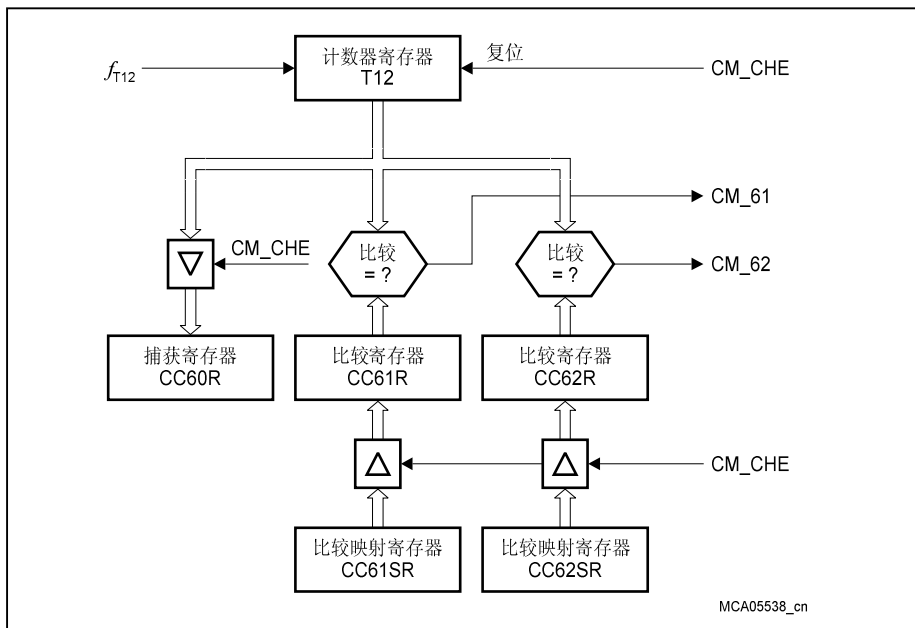


图 18-34 霍尔传感器模式下的 T12 模块

该模式下，正确霍尔事件（CM_CHE）控制从映射寄存器（CC6xSR）到实际比较寄存器（CC6xR）的映射传送。此外，该信号触发捕获操作，将 T12 的当前计数值捕获到寄存器 CC60R 中，然后强制 T12 复位为 0000_H。CM_CHE 还控制 T12 新周期值的映射传送。

注：该模式下，不产生映射传送信号 T12_ST。映射位，如 PSLy，不被传送到主寄存器（实际寄存器）中。只有当定时器 T12 不工作时才可设定主寄存器。此时，映射寄存器和主寄存器均可由软件编程设定。

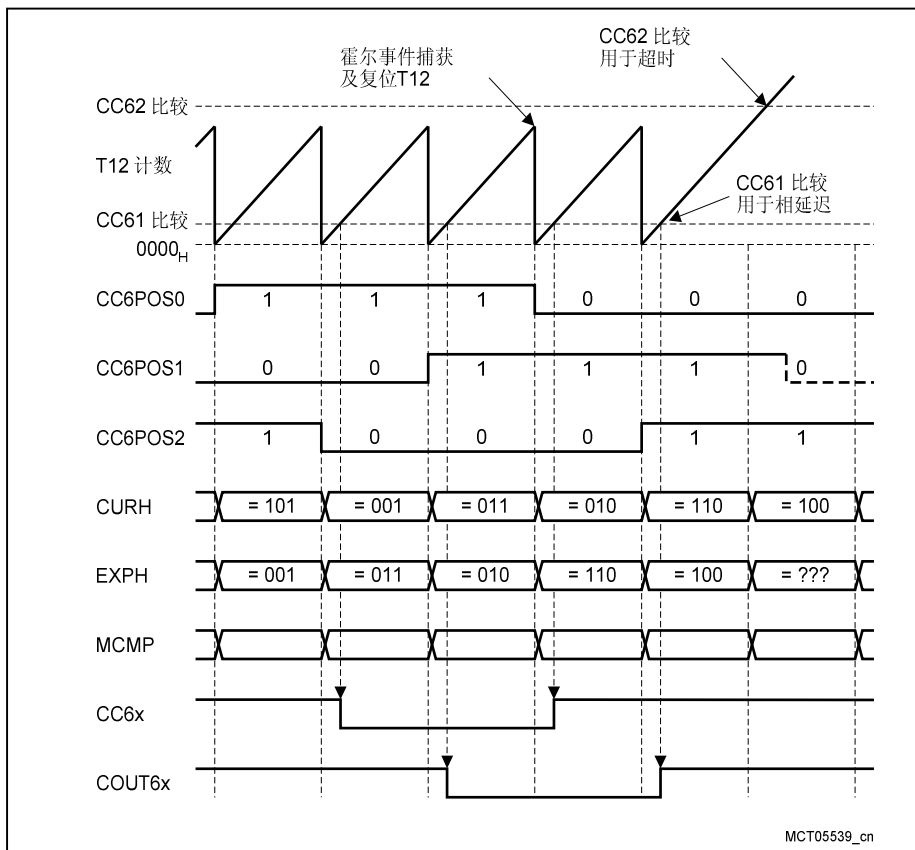


图 18-35 无刷直流电机控制举例（所有 MSEL6x = 1000B）

检测到有效的、期望霍尔序列后，T12 的计数值被捕获到通道 0 中（代表电机实际转速）并复位 T12。当定时器计数至通道 1 的比较值时，通过触发位域 MCMP 的映射传送（若位域 **SWEN** 使能映射传送）来切换到下个多通道状态。该触发事件可和某些条件相结合，用来实现噪声滤波（正确的霍尔事件）并使下个多通道状态与调制源同步（以避免输出毛刺）。在不使用霍尔传感器、而用反电动势检测转子位置的情况下，通道 1 的比较功能可产生从位置传感器的输入信号到输出切换之间的相位延迟。通道 2 的比较值可用作超时触发（中断），指示电机的实际转速远远低于期望转速（可能由异常的负载变化引起）。该模式下，必须禁止由 T12 调制输出（T12MODENx = 0）。

定时器计数值的捕获（捕获至寄存器 CC60R）、寄存器 CC61SR 到寄存器 CC61R 的映射传送、寄存器 CC62SR 到寄存器 CC62R 的映射传送、T12 周期值的映射传送、和 T12 复位事件同步进行。

18.5.4 霍尔模式标志位

根据霍尔序列的比较操作置位相关的标志位，以指示模块状态并触发其它操作和中断请求。

霍尔序列采样和期望序列（EXPH）匹配时，由信号 CM_CHE 置位寄存器 IS 中的标志位 CHE（正确的霍尔事件）。也可通过软件置位寄存器 ISS 中的 SCHE 对该标志位置位。若（寄存器 IEN 中的）位 ENCHE 被使能，CHE 的置位信号还可向 CPU 发送中断请求。可通过置位寄存器 ISR 的 RCHE 对 CHE 清零。

标志位 WHE 指示产生了错误的霍尔事件。该标志位的置位、复位以及中断请求产生方式和标志位 CHE 完全相同。

标志位 STR 的实现和 CHE、WHE 相同，该标志位由映射传送信号 MCM_ST 硬件置位（见图 18-31）。

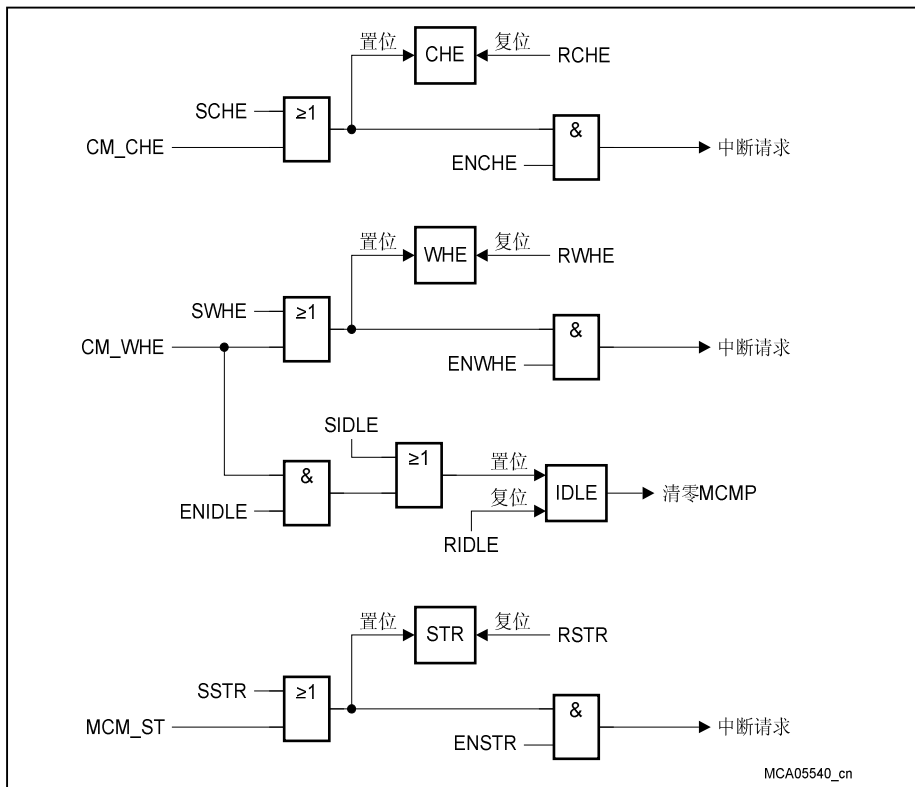


图 18-36 霍尔模式标志逻辑

请注意：标志位 CHE、WHE 和 STR 的中断请求由相应的置位信号触发产生。这意味着，即使标志位已经被置位，仍可产生中断请求，无需复位标志位以使能后续的中断请求。

不过标志位 IDLE 的实现有所不同。若使能位 ENIDLE 被置位，可由信号 CM_WHE 硬件置位 IDLE。也可通过软件置位 SIDLE 置位该标志位。只要 IDLE 被置位，调制序列 MCMP 将被清零、强制输出为被动态。必须通过置位 RIDLE 清零标志位 IDLE，以返回正常操作。要从 IDLE 模式彻底重启，必须通过设定寄存器 MCMOUTS 中的位 STRMCM 和 STRHP 来触发寄存器 MCMOUTS 到 MCMOUT 的映射请求。这种模式下，从 IDLE 模式返回正常模式的操作由软件控制，不过可以和 PWM 信号同步。

霍尔模式寄存器

寄存器 MCMOUTS 中存放着调制序列和霍尔序列的映射位域、以及由软件触发的映射传送控制位。相关映射传送信号被激活时，位域 MCMPS、EXPHS 和 CURHS 中的内容被传送到寄存器 MCMOUT 的对应位域中。

CCU6_MCMOUTS

多通道模式输出映射寄存器

XSFR (E8CA_H--)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STRHP	-	CURHS				EXPHS		STRMCM	-			MCMPS			
w	-	rw				rw		w	-			rw			

符号	位序号	读写类型	功能描述
STRHP	15	w	<p>霍尔序列映射传送请求</p> <p>置位 STRHP 引发立即更新，位域 CURHS 和 EXPHS 中的值立刻传送到位域 CURH 和 EXPH 中。该功能支持软件触发更新。读取该位时始终返回 0。</p> <p>0 根据定义的硬件动作更新位域 CURH 和 EXPH。写入位域 CURHS 和 EXPHS 不会修改位域 CURH 和 EXPH。</p> <p>1 位域 CURH 和 EXPH 由位域 CURHS 和 EXPHS 的写入值更新。</p>
CURHS	[13:11]	rw	<p>当前霍尔序列映射位域</p> <p>CURHS 是位域 CURH 的映射位域。检测到正确的霍尔事件时，该位域的内容传送到 CURH 中。</p>
EXPHS	[10:8]	rw	<p>期望霍尔序列映射位域</p> <p>EXPHS 是位域 EXPH 的映射位域。检测到正确的霍尔事件时，该位域的内容传送到 EXPH 中。</p>
STRMCM	7	w	<p>MCMPS 的映射传送请求</p> <p>置位 STRMCM 引发立即更新，位域 MCMPS 的值立即传送到位域 MCMP 中。</p>

符号	位序号	读写类型	功能描述
			<p>该功能支持软件触发更新。读取该位时始终返回 0。</p> <p>0 根据定义的硬件动作更新位域 MCMP。写入位域 MCMPS 不会修改 MCMP。</p> <p>1 位域 MCMP 由位域 MCMPS 的写入值更新。</p>
MCMPS	[5:0]	rw	<p>多通道 PWM 序列映射位域</p> <p>MCMPS 是位域 MCMP 的映射位域。根据寄存器 MCMCTR 定义的传送条件触发多通道映射传送。</p>

寄存器 MCMOUT 中存放着当前使用的调制序列和霍尔序列。

CCU6_MCMOUT

多通道模式输出寄存器

XSFR (E8CC_H/--)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	CURH		EXPH		-	R	MCMP							
-	-	rh		rh		-	rh	rh							

符号	位序号	读写类型	功能描述
CURH¹⁾	[13:11]	rh	<p>当前的霍尔序列</p> <p>CURH 的值通过映射传送从位域 CURHS 中获取。每次检测到霍尔传感器的输入引脚 CC6POS_x 发生跳变后，将位域 CURH 和霍尔序列采样进行比较。若两序列匹配，则所检测的跳变沿无效（如，由毛刺引起）。若序列采样和 CURH 或 EXPH 都不匹配，则置位错误霍尔事件信号，触发其它相关操作。</p>
EXPH¹⁾	[10:8]	rh	<p>期望的霍尔序列</p> <p>EXPH 的值通过映射传送从位域 EXPHS 中获取。每次检测到霍尔传感器的输入引脚</p>

符号	位序号	读写类型	功能描述
			CC6POSx 发生跳变后，将位域 EXPH 和霍尔序列采样进行比较。若两序列匹配，置位可触发其它操作的正确的霍尔事件信号。
R	6	rh	提示标志 该标志表明选定的触发源已请求位域 MCMP5 到 MCMP 的映射传送。MCMEN = 0 且发生映射传送时，该位被清零。 0 未请求映射传送 1 已产生从 MCMP5 到 MCMP 的映射传送请求，但映射传送还未执行
MCMP²⁾	[5:0]	rh	多通道调制序列 MCMP 中存放着多通道模式的输出调制序列，可将相应的输出设置为被动态。MCMP 的值从位域 MCMP5 映射传送到 MCMP 中。 0 输出设定为被动态 1 可输出 T12 或 T13 产生的 PWM 信号（根据寄存器 MODCTR） MCMP[5:0]（从左至右）分别对应： COUT62、CC62、COUT61、CC61、COUT60、CC60。

1) 位域 EXPH 和 CURH 中的各位和输入引脚 CCPOSx（x = 0, 1, 2）上的霍尔序列对应关系如下：
 （EXPH.2, EXPH.1, EXPH.0），（CURH.2, CURH.1, CURH.0），（CCPOS2, CCPOS1, CCPOS0）。

2) 位 IS.IDLE = 1 时，位域 MCMP 被清零。

寄存器 MCMCTR 中存放着多通道模式的控制位，控制输出调制序列。

CCU6_MCMCTR

多通道模式控制寄存器

XSFR (E8CEH/--)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	-	-	SWSYN	-	-	-	-	SWSEL
-	-	-	-	-	-	-	-	-	-	rw		-	rw		

符号	位序号	读写类型	功能描述
SWSYN	[5:4]	rw	切换同步 若 SWSEL 选定的事件已请求映射传送（标志 R 置位），位域 SWSYN 触发从 MCMP5 到 MCMP 的映射传送。该特性可使输出和调制源（T12 或 T13）同步。见表 18-9。
SWSEL	[2:0]	rw	切换选择 SWSEL 选择从 MCMP5 到 MCMP 映射传送的请求源（下一次的多通道事件）。映射传送和 SWSYN 选定的事件同步发生。见表 18-8。

注：只有位 MCMEN = 1 时，才能使能由硬件产生映射传送请求。

18.6 强制中断处理

强制中断功能使 PWM 输出能够响应输入引脚 $\overline{\text{CTRAP}}$ 的变化。如果强制中断输入有效，则相应输出截止，从而关闭功率器件（如，用于实现紧急停止）。

强制中断标志 TRPF 监控强制中断输入的变化并启动进入强制中断状态。该标志也可由软件置位。强制中断状态位 TRPS 决定输出结果并控制强制中断状态的退出。

检测到强制中断条件时，强制中断标志 TRPF 和强制中断状态位 TRPS 均被置 1，立即进入强制中断状态。TRPS 的输出送入输出调制模块，从而使输出无效（输出设置为被动态）。CAPCOM6 单元分别为 T12 的各路输出和 T13 的单路输出提供了对应的强制中断使能控制位，以便适用于不同的应用需求（请参阅章节 18.7）。

有多种退出强制中断状态的方式，可编程选择针对给定应用的最佳方式。退出强制中断状态的方式有：强制中断条件撤销后立即退出、软件控制退出、或者使退出和（由定时器 T12 或 T13 产生的）PWM 信号同步。

图 18-37 给出强制中断功能的逻辑框图。输入 $\overline{\text{CTRAP}}$ 有效时，置位寄存器 IS 中的强制中断标志 TRPF 和强制中断状态位 TRPS。也可通过软件置位 STRPF 来置位标志位 TRPF。强制中断状态位 TRPS 也可由其置位/复位控制模块设定。只要引脚 $\overline{\text{CTRAP}} = 0$ ，TRPF 和 TRPS 保持置位，不能被清零（假设 $\text{TRPPEN} = 1$ ）。

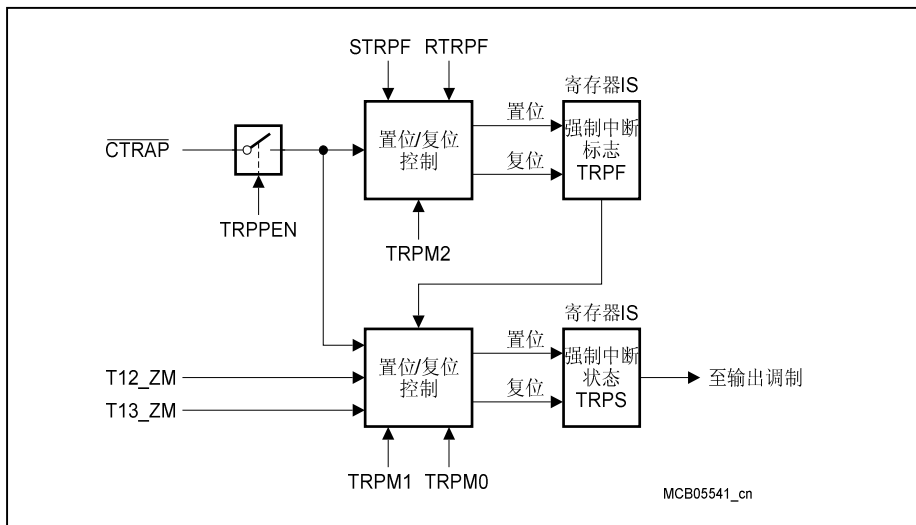


图 18-37 强制中断逻辑框图

捕获/比较单元 6 (CAPCOM6)

模式控制位 $\overline{\text{TRPM2}}$ （位于强制中断控制寄存器 TRPCTR 中）控制 TRPF 复位。若 $\text{TRPM2} = 0$ ， $\overline{\text{CTRAP}}$ 返回无效电平时（ $\overline{\text{CTRAP}} = 1$ ） TRPF 由硬件自动清零；若 $\text{TRPM2} = 1$ ， $\overline{\text{CTRAP}}$ 变为无效之后， TRPF 必须由软件复位。

模式控制位 TRPM1 和 TRPM0 （位于强制中断控制寄存器 TRPCTR 中）控制 TRPS 复位。 TRPS 复位将退出强制中断状态、返回正常操作。有三种方式复位 TRPS ，分别由 TRPM1 和 TRPM0 控制选择：一种方式为当强制中断标志 TRPF 被清零后，立即退出强制中断状态，不和定时器 T12 或 T13 同步；另两种方式使退出强制中断状态分别和定时器 T12 或 T13 的计数周期同步。图 18-38 给出相关操作。

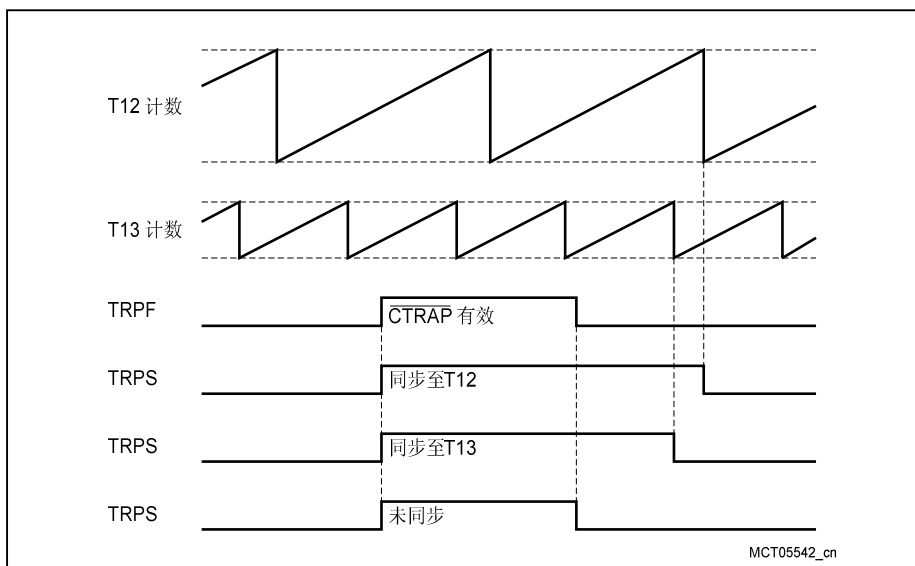


图 18-38 强制中断状态同步（ $\text{TRPM2} = 0$ ）

强制中断处理寄存器

寄存器 TRPCTR 控制强制中断功能。TRPCTR 中存放着各路输出信号的强制中断使能控制位、以及发生强制中断时的操作选择控制位。

CCU6_TRPCTR

强制中断控制寄存器

XSFR (E8C2H/--)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRP PEN	TRP EN13	TRPEN				-	-	-	-	-	-	TRP M2	TRP M1	TRP M0	
rw	rw	rw				-	-	-	-	-	-	rw	rw	rw	

符号	位序号	读写类型	功能描述
TRPPEN	15	rw	强制中断输入引脚使能控制 0 强制中断输入引脚 $\overline{\text{CTR\!AP}}$ 被禁止。可软件置位 TRPF 产生强制中断。 1 强制中断输入引脚 $\overline{\text{CTR\!AP}}$ 被使能。强制中断既可通过软件置位 TRPF 产生，也可由 $\overline{\text{CTR\!AP}} = 0$ 硬件触发。
TRPEN13	14	rw	T13 输出的强制中断使能控制 使能 T13 的输出信号 CC63 的强制中断功能。 0 强制中断功能被禁止。T13 的输出和状态位 TRPS 无关。 1 强制中断功能被使能。位 TRPS = 1 时，T13 的输出被设置为被动态。
TRPEN	[13:8]	rw	T12 输出的强制中断使能控制 使能 T12 的输出信号 CC6x 和 COUT6x 的强制中断功能。 0 强制中断功能被禁止。T12 的输出和状态位 TRPS 无关。 1 强制中断功能被使能。位 TRPS = 1 时，T12 的输出被设置为被动态。 TRPEN[5:0]（从左至右）分别对应： COUT62、CC62、COUT61、CC61、

符号	位序号	读写类型	功能描述
			COUT60、CC60。
TRPM2	2	rw	<p>强制中断模式控制位 2</p> <p>TRPM2 控制选择由硬件还是软件控制退出强制中断状态。</p> <p>0 由硬件控制退出强制中断状态。输入引脚 $\overline{\text{CTRAP}}$ 变为无效（$\overline{\text{CTRAP}} = 1$）时，位 TRPF 由硬件自动清零。</p> <p>1 由软件控制退出强制中断状态。输入引脚 $\overline{\text{CTRAP}}$ 变为无效后（$\overline{\text{CTRAP}} = 1$），必须由软件复位 TRPF。</p>
TRPM1 TRPM0	[1:0]	rw	<p>强制中断模式控制位 1,0</p> <p>这两位控制如何退出强制中断状态。强制中断标志 TRPF 被复位后，选择以下方式复位强制中断状态位 TRPS、退出强制中断状态：</p> <p>00 和 T12 同步：T12 0-匹配（T12_ZM）时复位 TRPS、退出强制中断状态</p> <p>01 和 T13 同步：T13 0-匹配（T13_ZM）时复位 TRPS、退出强制中断状态</p> <p>10 保留，无操作</p> <p>11 不和 T12 或 T13 同步：强制中断标志 TRPF 被复位后，立即复位 TRPS、退出强制中断状态。</p>

18.7 输出调制控制

数据最后送入输出调制控制逻辑模块。该模块将所有的调制源逻辑组合，以控制输出引脚的实际电平。

下面将分别讨论六路 T12 的相关输出（CC6x，COUT6x）和 T13 的相关输出 CC63。

图 18-39 所示为控制 T12 相关输出的六个控制模块和多个控制信号。每个控制模块的输入为四路调制信号及对应的使能控制信号。三个状态位 CC6xST 的输出经过状态选择逻辑产生调制信号 CC6x_O 和 COUT6x_O（见**图 18-15**）。信号 MCMPy 对应多通道模式寄存器 MCMOUT 的六路输出（见**图 18-31**）。状态位 CC63_ST 的输出经过状态选择逻辑产生信号 CC63_O（见**图 18-30**），该信号送入所有 6 个调制控制模块。强制中断状态位对应的输出信号 TRPS 也送入所有 6 个调制控制模块中（见**图 18-37**）。

每个调制控制模块的输入信号 CC6x_O/COUT6x_O，CC63_O 和 TRPS 分别对应有各自的使能控制信号；所有输入信号 MCMPy 只对应有一个通用使能控制信号 MCMEN。

每个调制控制模块的输出和一个电平选择模块相连，该模块根据输出线的状态，决定引脚的实际输出电平。

图 18-40 为单个调制和电平选择模块详细的逻辑框图。该模块将不同的控制信号逻辑组合、使得只有被使能的信号才可最终影响输出线 MCL_OUT。若某个调制信号 CC6x_O/COUT6x_O，CC63_O，或 MCMPx 被使能且处于被动态，则输出 MCL_OUT 也处于被动态，和其它使能信号的状态无关；只有当所有使能信号均处于主动态，输出 MCL_OUT 才是主动态。

若强制中断状态有效（TRPS = 1），所有强制中断被使能（TRPENy = 1）的输出被设置为被动态。

可根据调制控制模块的输出 MCL_OUT，由被动态选择控制位 PSLy 选择输出引脚的实际电平。当 MCL_OUT 处于被动态，PSLy 指定的电平即为输出的实际电平；当 MCL_OUT 处于主动态，PSLy 的反相电平为输出的实际电平。

控制位 PSLy 对应有映射寄存器，从而在更新 PSLy 时避免了输出线上多余的脉冲。T12 的映射传送信号（T12_ST）触发 PSLy 更新。读操作读取实际使用值；写操作写入映射位。CAPCOM6 单元提供了 PSL、以及（和产生 PWM 信号相关的）其它数值的映射寄存器，便于用户编程同步更新所有相关参数（请参阅**章节 18.8**）。

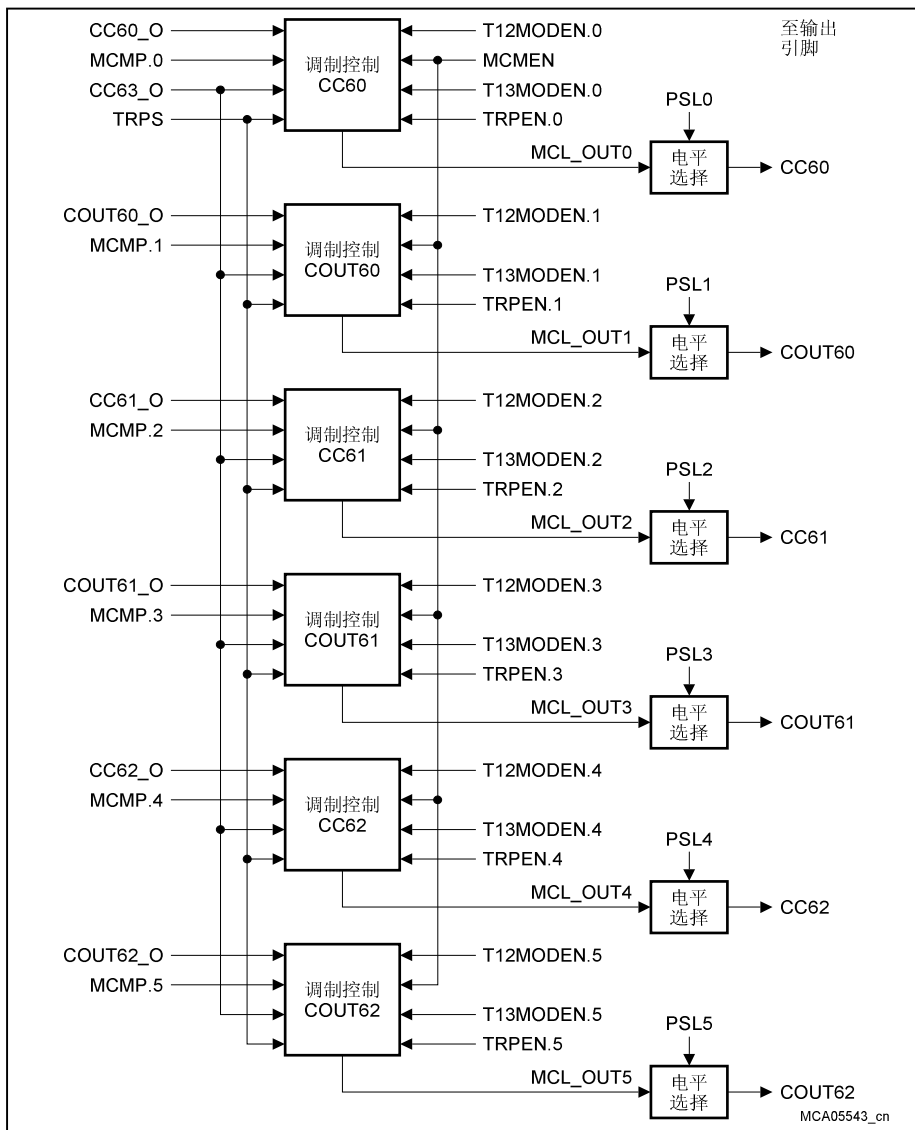


图 18-39 输出调制总览

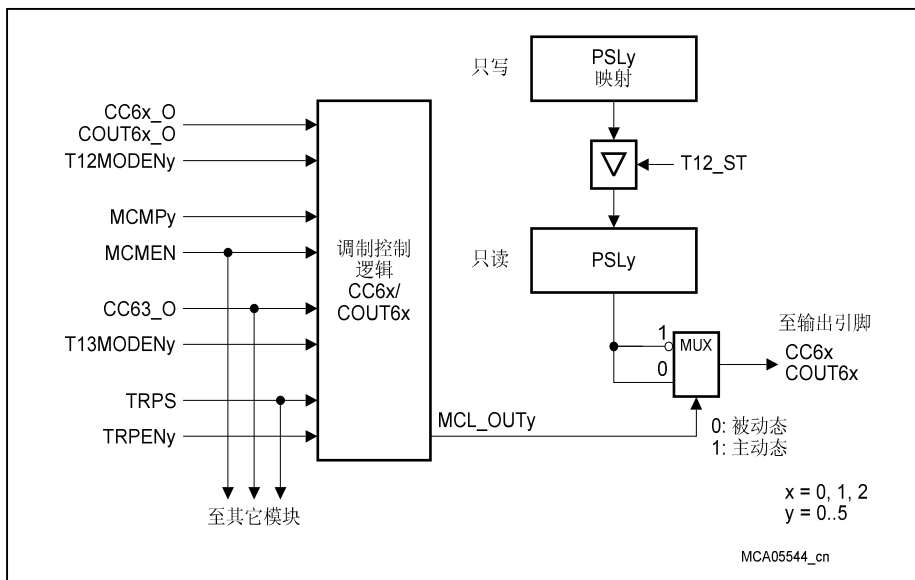


图 18-40 定时器 T12 的输出调制

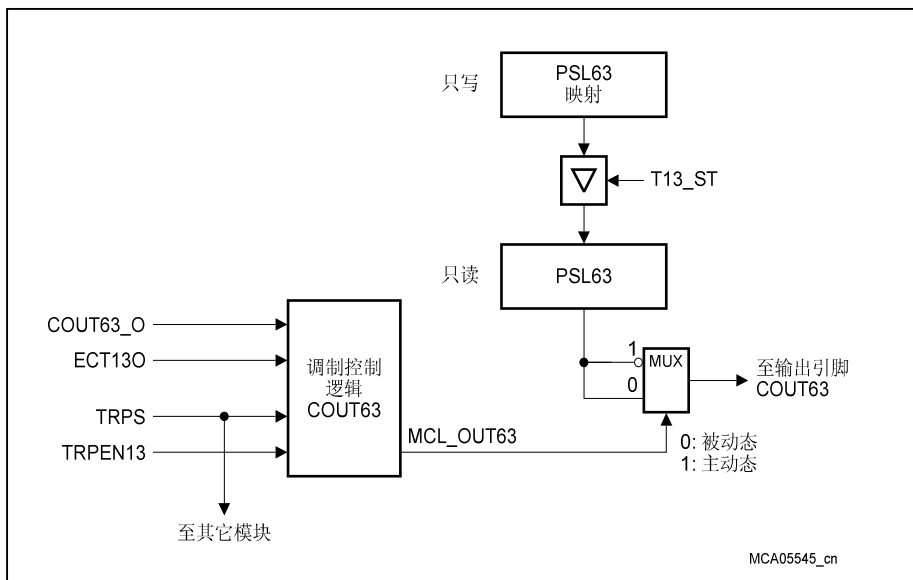


图 18-41 定时器 T13 的输出调制

捕获/比较单元 6（CAPCOM6）

寄存器 **PSLR** 定义了模块输出引脚驱动的被动态电平。输出端口处于被动态时，引脚驱动的电平为被动态电平。输出端口处于主动态时，引脚驱动的电平为主动态电平，该电平和被动态电平反相。被动态电平可使输出电平和所连接的功率器件的驱动极性（反相或不反相）相匹配。

CCU6_PSLR

被动态电平寄存器

XSFR (E8C4H/--)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	PSL 63	-	PSL					
-	-	-	-	-	-	-	-	rwh	-	rwh					

符号	位序号	读写类型	功能描述
PSL63	7	rwh	T13 输出 COUT63 的被动态电平控制 该位域决定输出引脚 COUT63 的被动态电平。 0 被动态电平为 0 1 被动态电平为 1
PSL	[5:0]	rwh	T12 输出的被动态电平控制 该位域决定模块输出为被动态时所驱动的被动态电平。 0 被动态电平为 0 1 被动态电平为 1 PSL[5:0]（从左至右）分别对应： COUT62、CC62、COUT61、CC61、 COUT60、CC60。

18.8 映射寄存器传送控制

图 18-42 和图 18-43 概括了映射寄存器的结构和映射传送信号，以及对不同寄存器的读写访问。CAPCOM6 单元采用映射寄存器结构，用于存储同一个 PWM 周期的参数，便于用户同步更新所有和 PWM 周期信号产生相关的参数。新 PWM 周期可用一组新参数产生。

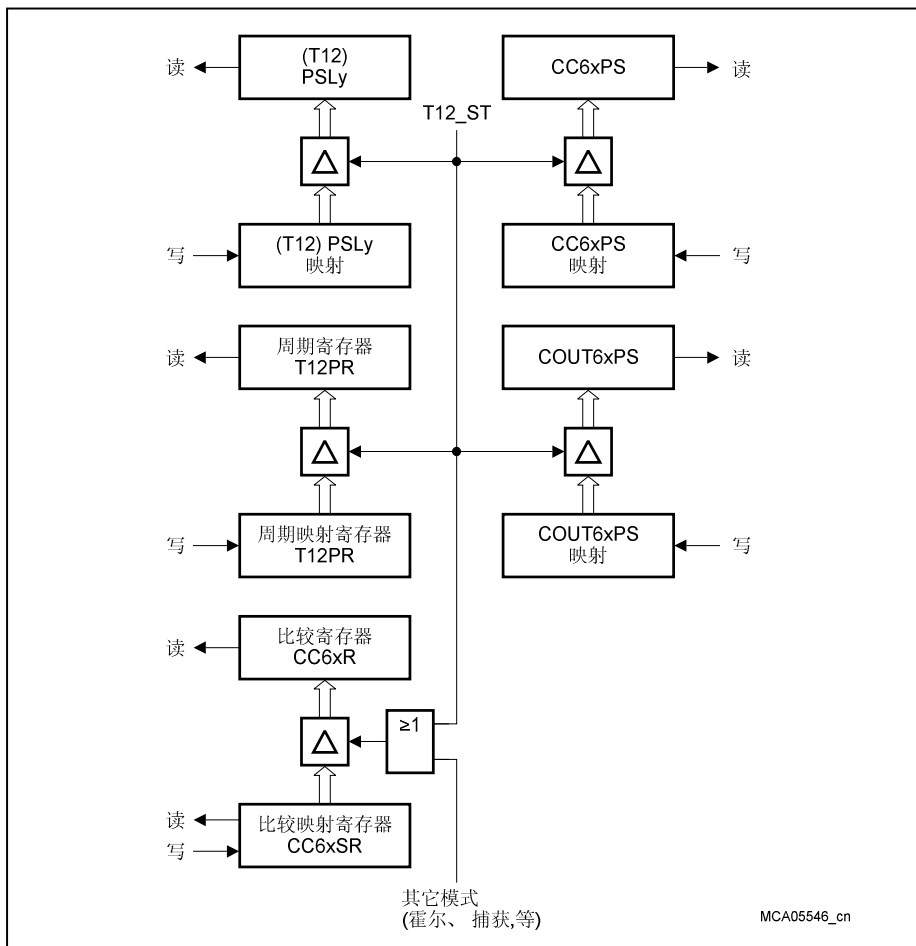


图 18-42 T12 映射寄存器和传送信号总览

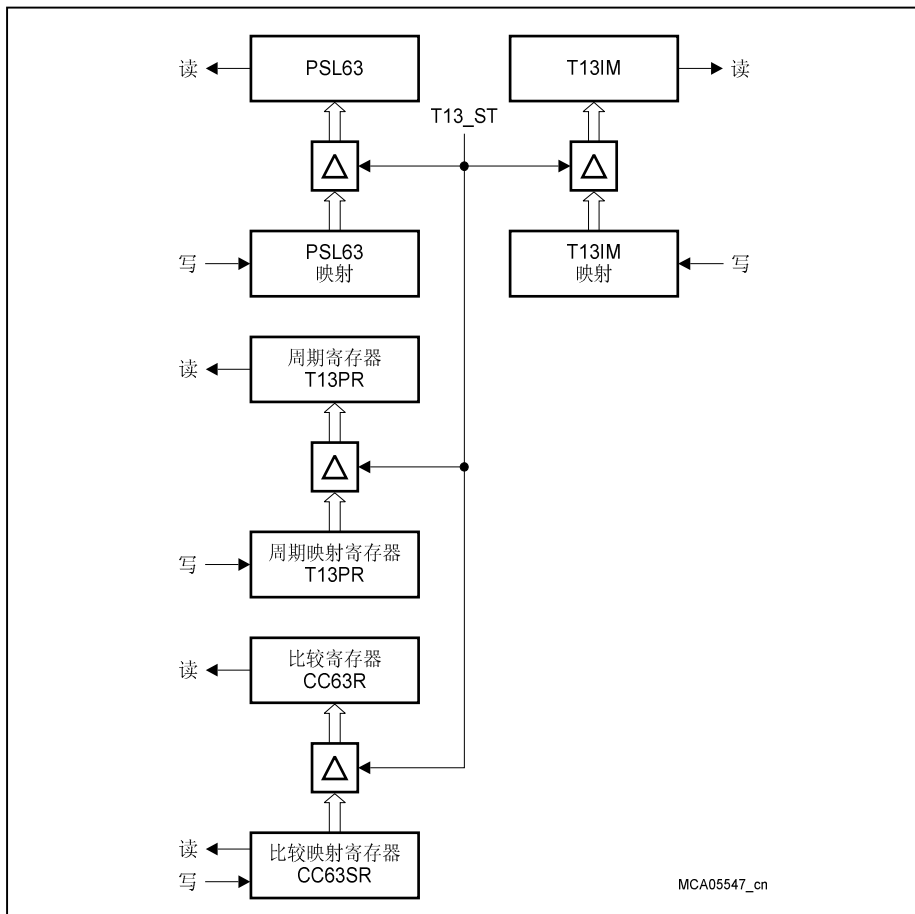


图 18-43 T13 映射寄存器和传送信号总览

18.9 中断产生

中断结构如图 18-44 所示。硬件中断事件、或软件置位（寄存器 ISS 中的）相关中断置位可触发中断产生。中断脉冲的产生和寄存器 IS 中的中断标志无关。对寄存器 ISR 中的相关位置位可软件复位中断标志。

若寄存器 IEN 中的某中断使能位被置位，可在某中断输出线上（CAPCOM6 单元共有四条中断线 I0...I3）产生中断脉冲。若多个中断源和同一个中断节点指针（位于寄存器 INP 中）相连，各中断请求合并为一条公共的中断请求输出线。

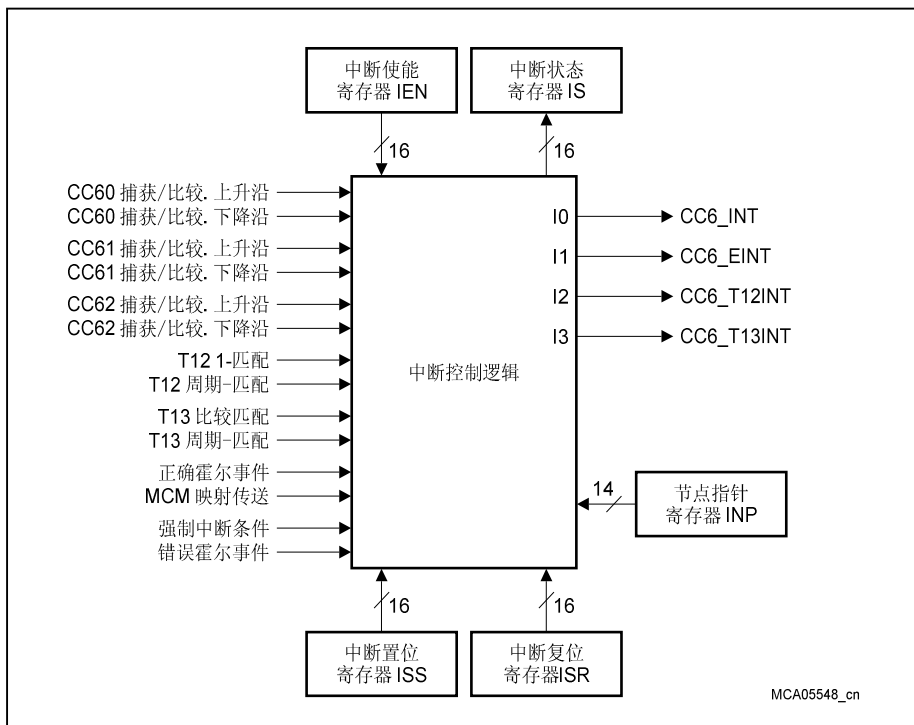


图 18-44 中断结构总览

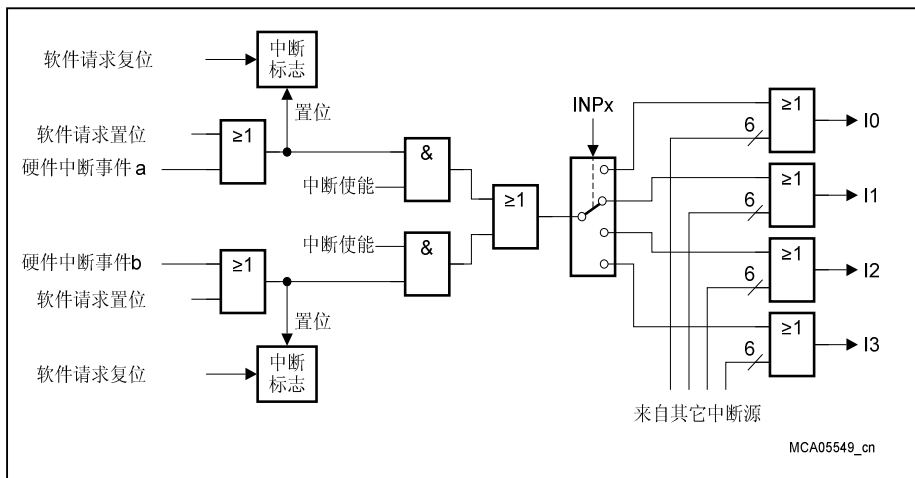


图 18-45 中断内部结构

中断寄存器

寄存器 IS 中存放着各中断请求和状态位。IS 为只读寄存器，写操作不会改变该寄存器的值。置位寄存器 ISS 或寄存器 ISR 可分别置位或复位寄存器 IS 中的对应位。

CCU6_IS

中断状态寄存器

XSFR (E8D0H/--)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	IDLE	WHE	CHE	TRP S	TRP F	T13 PM	T13 CM	T12 PM	T12 OM	ICC 62F	ICC 62R	ICC 61F	ICC 61R	ICC 60F	ICC 60R
-	rh	rh	rh	rh	rh	rh	rh	rh	rh	rh	rh	rh	rh	rh	rh

符号	位序号	读写类型	功能描述
IDLE	14	rh	空闲状态标志位 若空闲状态被使能（ENIDLE = 1），位 WHE 置位时（发生错误的霍尔事件），IDLE 被置位，该位必须由软件清零。 0 无操作 1 位域 MCMP 被清零，选定的输出被设

捕获/比较单元 6（CAPCOM6）

符号	位序号	读写类型	功能描述
			置为被动态
WHE	13	rh	错误霍尔事件标志位 0 还未检测到错误霍尔序列 1 已检测到错误霍尔序列（非期望序列）
CHE	12	rh	正确霍尔事件标志位 0 还未检测到正确（期望的）霍尔序列 1 已检测到期望的霍尔序列
TRPS¹⁾	11	rh	强制中断状态位 0 未处于强制中断状态。 1 处于强制中断状态。位 TRPF = 1 时置位 TRPS。根据寄存器 TRPCTR 中选定的模式复位 TRPS。
TRPF	10	rh	强制中断标志位 若 TRPPEN = 1 且 $\overline{\text{CTRAP}} = 0$ ，硬件置位强制中断标志 TRPF，该标志位也可由软件置位。若 TRPM2 = 0，输入 $\overline{\text{CTRAP}}$ 失效时，硬件复位 TRPF（TRPPEN = 1）；若 TRPM2 = 1，必须由软件复位 TRPF 以退出强制中断状态。 0 还未发生强制中断 1 已发生强制中断（ $\overline{\text{CTRAP}} = 0$ 或由软件控制）
T13PM	9	rh	定时器 T13 周期匹配标志位 0 还未发生 T13 周期匹配 1 已发生 T13 周期匹配
T13CM	8	rh	定时器 T13 比较匹配标志位 0 还未发生 T13 比较匹配 1 已发生 T13 比较匹配
T12PM	7	rh	定时器 T12 周期匹配标志位 0 还未发生 T12 周期匹配

捕获/比较单元 6（CAPCOM6）

符号	位序号	读写类型	功能描述
			1 T12（递增计数时）已发生周期匹配
T12OM	6	rh	定时器 T12 1- 匹配标志位 0 还未发生 T12 1- 匹配 1 T12（递减计数时）已发生 1- 匹配
ICC62F	5	rh	捕获，比较匹配下降沿标志位 比较模式下，T12 递减计数时检测到比较匹配；捕获模式下，检测到输入 CC6x（x = 0, 1, 2）发生下降沿跳变。 0 上述事件还未发生 1 上述事件已发生
ICC61F	3		
ICC60F	1		
ICC62R	4	rh	捕获，比较匹配上升沿标志位 比较模式下，T12 递增计数时检测到比较匹配；捕获模式下，检测到输入 CC6x（x = 0, 1, 2）发生上升沿跳变。 0 上述事件还未发生 1 上述事件已发生
ICC61R	2		
ICC60R	0		

1) 强制中断状态下，选定的输出被设置为被动态。

注：并非寄存器 IS 中所有位均可产生中断，寄存器 IS 中还包含状态位，这些状态位的置位和复位操作和其它标志位相似。

注：比较模式（及霍尔模式）下，只有在定时器运行（TxR = 1）时才能产生和定时器相关的中断。捕获模式下，即使定时器 T12 不工作也会产生捕获中断。

捕获/比较单元 6（CAPCOM6）

只写寄存器 ISS 和 ISR 中的各位分别对应寄存器 IS 中的各中断和状态标志（位 11 除外）。置位 ISS 或 ISR，可分别置位或清零相关标志。读取这些控制位始终返回 0。

寄存器 ISS 中的某位置 1，将置位寄存器 IS 中对应标志，并可能触发中断请求（若该中断被使能并且相应功能可用）。

寄存器 ISR 中的某位置 1，将复位寄存器 IS 中对应的标志。

CCU6_ISS

中断状态置位寄存器

XSFR (E8D2H/--)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	S IDLE	S WHE	S CHE	-	S TRP F	S T13 PM	S T13 CM	S T12 PM	S T12 OM	S CC 62F	S CC 62R	S CC 61F	S CC 61R	S CC 60F	S CC 60R
-	W	W	W	-	W	W	W	W	W	W	W	W	W	W	W

符号	位序号	读写类型	功能描述 ¹⁾
SIDLE	14	w	置位 IDLE 标志位
SWHE	13	w	置位错误霍尔事件标志位
SCHE	12	w	置位正确霍尔事件标志位
STRPF	10	w	置位强制中断标志位
ST13PM	9	w	置位定时器 T13 周期匹配标志位
ST13CM	8	w	置位定时器 T13 比较匹配标志位
ST12PM	7	w	置位定时器 T12 周期匹配标志位
ST12OM	6	w	置位定时器 T12 1- 匹配标志位
SCC62F SCC61F SCC60F	5 3 1	w	置位捕获，比较匹配下降沿标志位
SCC62R SCC61R SCC60R	4 2 0	w	置位捕获，比较匹配上升沿标志位

1) 对这些控制位写 1 将置位对应的标志，写 0 无效。

CCU6_ISR

中断状态复位寄存器

XSFR (E8D4H/--)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	R IDLE	R WHE	R CHE	-	R TRP F	R T13 PM	R T13 CM	R T12 PM	R T12 OM	R CC 62F	R CC 62R	R CC 61F	R CC 61R	R CC 60F	R CC 60R
-	W	W	W	-	W	W	W	W	W	W	W	W	W	W	W

符号	位序号	读写类型	功能描述 ¹⁾
RIDLE	14	w	复位 IDLE 标志位
RWHE	13	w	复位错误霍尔事件标志位
RCHE	12	w	复位正确霍尔事件标志位
RTRPF	10	w	复位强制中断标志位
RT13PM	9	w	复位定时器 T13 周期匹配标志位
RT13CM	8	w	复位定时器 T13 比较匹配标志位
RT12PM	7	w	复位定时器 T12 周期匹配标志位
RT12OM	6	w	复位定时器 T12 1- 匹配标志位
RCC62F RCC61F RCC60F	5 3 1	w	复位捕获，比较匹配下降沿标志位
RCC62R RCC61R RCC60R	4 2 0	w	复位捕获，比较匹配上升沿标志位

1) 对这些控制位写 1 将清零对应的标志，写 0 无效。

注：可通过以下方式置位寄存器 ISS 或 ISR 中的置位/清零位：位操作（如 BSET）、逻辑操作（如 OR）、或 MOV 单指令（如通过 PEC 执行）。

捕获/比较单元 6（CAPCOM6）

寄存器 IEN 中存放着中断使能位和一个特殊控制位，发生错误霍尔序列时该控制位使能自动转入空闲状态。置位寄存器中某位将使能对应的中断请求（或定义的其它功能），清零寄存器中某位将禁止对应的中断请求（或定义的其它功能）。

CCU6_IEN

中断使能寄存器

XSFR (E8D8H/--)

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	EN IDLE	EN WHE	EN CHE	-	EN TRPF	EN T13 PM	EN T13 CM	EN T12 PM	EN T12 OM	EN CC 62F	EN CC 62R	EN CC 61F	EN CC 61R	EN CC 60F	EN CC 60R
-	rw	rw	rw	-	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

符号	位序号	读写类型	功能描述
ENIDLE	14	rw	使能 IDLE 标志置位功能
ENWHE	13	rw	使能错误霍尔事件中断
ENCHE	12	rw	使能正确霍尔事件中断
ENTRPF	10	rw	使能强制中断标志中断
ENT13PM	9	rw	使能定时器 T13 周期匹配中断
ENT13CM	8	rw	使能定时器 T13 比较匹配中断
ENT12PM	7	rw	使能定时器 T12 周期匹配中断
ENT12OM	6	rw	使能定时器 T12 1- 匹配中断
ENCC62F ENCC61F ENCC60F	5 3 1	rw	使能捕获，比较匹配下降沿中断
ENCC62R ENCC61R ENCC60R	4 2 0	rw	使能捕获，比较匹配上升沿中断

捕获/比较单元 6（CAPCOM6）

通过设置中断节点指针寄存器 CCU6_INP，将 CAPCOM6 模块的多个中断源映射到 4 个中断节点上。中断节点指针的位域编码（中断节点指针和中断输出线的对应关系）见**表 18-10**。

CCU6_INP

中断节点指针寄存器

XSFR (E8D6H/--)

复位值: 3940H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	INPT13	INPT12	INPERR	INPCHE	INPCC62	INPCC61	INPCC60							
-	-	rw	rw	rw	rw	rw	rw	rw							

符号	位序号	读写类型	功能描述
INPT13	[13:12]	rw	定时器 T13 中断的中断节点指针 该位域选择中断请求源 T13CM 和/或 T13PM 所激活的中断输出线。
INPT12	[11:10]	rw	定时器 T12 中断的中断节点指针 该位域选择中断请求源 T12OM 和/或 T12PM 所激活的中断输出线。
INPERR	[9:8]	rw	错误中断的中断节点指针 该位域选择中断请求源 TRPF 和/或 WHE 所激活的中断输出线。
INPCHE	[7:6]	rw	CHE 中断的中断节点指针 该位域选择中断请求源 CHE 和/或 STR 所激活的中断输出线。
INPCC62 INPCC61 INPCC60	[5:4] [3:2] [1:0]	rw	通道 x 中断的中断节点指针 该位域选择中断请求源 CC6xR 和/或 CC6xF 所激活的中断输出线。

表 18-10 中断节点指针位域编码

位域 INPxx	被激活的中断输出线
00 _B	I0
01 _B	I1
10 _B	I2
11 _B	I3

中断请求源到中断节点的缺省映射方式、中断源对应的控制寄存器列于表 18-11 中。

所有中断控制寄存器的结构相同。寄存器基本结构如下所示，表 18-11 中列出相关的寄存器地址。

CCU6_xIC

CAPCOM6 中断控制寄存器

ESFR（表 18-11）

复位值: - - 00_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	GPX	CCy IR	CCy IE	ILVL				GLVL	
-	-	-	-	-	-	-	rw	rwh	rw	rw				rw	

注：控制位域的具体解释请参阅中断控制寄存器的描述。

表 18-11 CAPCOM6 缺省中断节点寄存器分配

中断源	中断请求线	中断控制寄存器	寄存器地址
通道 0 中断	I0	CCU6_IC	F140 _H
通道 1 中断	I0		
通道 2 中断	I0		
正确霍尔序列中断	I1	CCU6_EIC	F188 _H
紧急事件中断	I1		
定时器 T12 中断	I2	CCU6_T12IC	F190 _H
定时器 T13 中断	I3	CCU6_T13IC	F198 _H

18.10 暂停模式

暂停模式下，模块时钟 f_{CC6} 停止，CPU 仍可访问寄存器（只读）。该模式可用于调试，如，“冻结”当前的器件状态以获取某些内部值。暂停模式下，定时器 T12 和 T13 停止工作。暂停模式不会影响寄存器，也就是说，进入或退出暂停模式时，硬件不会修改寄存器的内容。软件也无需干预。暂停模式下，可由读指令访问所有寄存器以进行调试。

进入暂停模式的条件为：暂停模式被请求并被使能、并且模块已进入安全、确定的状态（类似单次模式下定时器停止工作的条件）。当模块输出与功率逆变器相连时，该模式可避免出现危急状况。

暂停状态不会直接影响输出信号。

18.11 CAPCOM6 单元接口

CAPCOM6 单元以不同的方式和周围资源接口。

内部连接

CAPCOM6 单元的 4 条中断请求线送至中断控制模块。

定时器 T13 的周期匹配信号 (T13_PM) 和 ADC 模块相连，用作插入转换的触发源。

外部连接

CAPCOM6 单元的信号和 XC164CM 的输入/输出端口相连。这些端口可提供来自外部系统的捕获触发信号、将比较输出信号送至外部电路，或者接受控制输入信号。

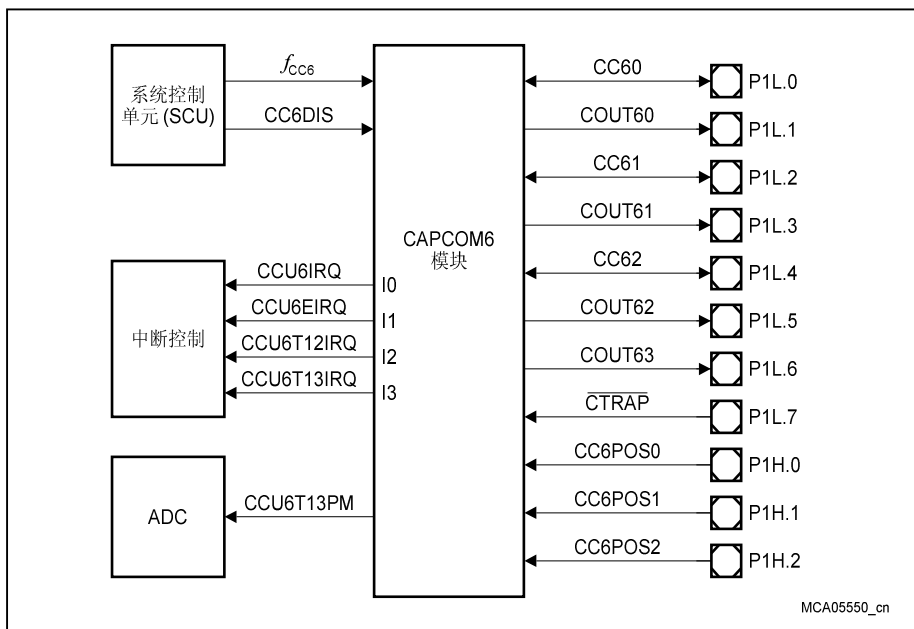


图 18-46 CAPCOM6 单元接口

19 异步/同步串行接口（ASC）

XC164CM 包含 2 个异步/同步串行接口，ASC0 和 ASC1。下面的章节给出了 ASC 模块一般特性和操作。最后一节描述两个 ASC 模块的具体实现，包括 ASC 模块和其它片上模块的相互连接。

ASC 模块支持全双工异步通信和半双工同步通信。ASC 提供下列特性及功能。

特性和功能

- 全双工异步工作模式
 - 8 或 9 位数据帧，LSB 在先
 - 奇偶校验位产生/检查
 - 1 或 2 个停止位
 - 波特率范围 2.5 Mbit/s 到 50 bit/s（@40MHz 模块时钟 f_{ASC} ）
- 带有自动地址/数据字节检测的多处理器通信模式
- 回环能力
- 支持 IrDA 数据传输，数据率最高可达 115.2 kbit/s
- 半双工 8 位同步工作模式
 - 波特率范围 5 Mbit/s 到 202 bit/s（@40MHz 模块时钟 f_{ASC} ）
- 双缓存发送器/接收器
- 中断产生
 - 发送器缓存为空的条件下产生中断
 - 发送一帧最后一位的条件下产生中断
 - 接收器缓存已满的条件下产生中断
 - 错误条件下产生中断（帧错误，奇偶校验错误，过载错误）
- 异步工作模式的自动波特率检测单元
 - 标准波特率检测：1200、2400、4800、9600、19200、38400、57600、115200 和 230400 bit/s
 - 非标准波特率检测
 - 异步模式检测
 - 7 位，偶校验；7 位，奇校验；8 位，偶校验；8 位，奇校验；8 位，无奇偶校验
 - 检测之后自动初始化控制位和波特率发生器
 - 串行 2 字节 ASCII 字符帧检测

- FIFO
 - 8 级接收 FIFO (RXFIFO)，8 级发送 FIFO (TXFIFO)
 - RXFIFO 和 TXFIFO 独立控制
 - 9 位 FIFO 数据宽度
 - 可编程的接收/发送中断触发级别
 - 接收和发送 FIFO 填充级别指示
 - 过载和下溢错误产生

图 19-1 给出所有与 ASC 内核功能相关的接口。

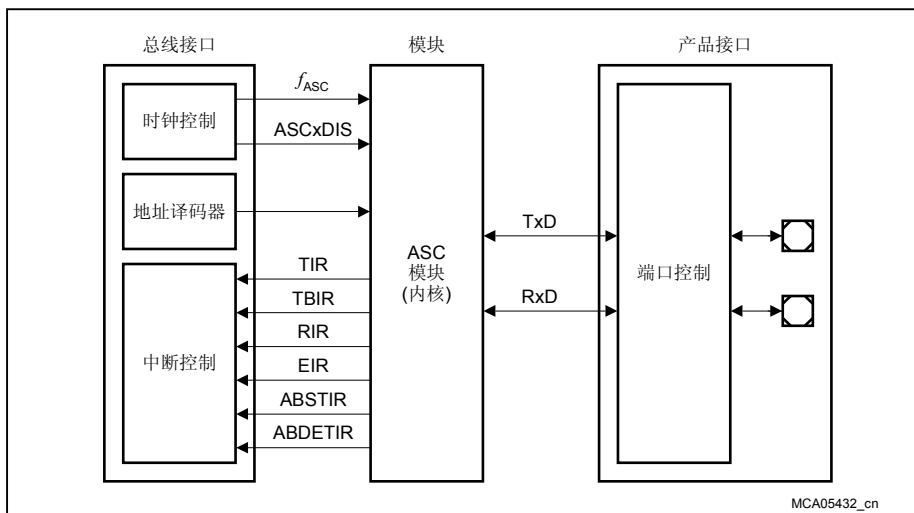


图 19-1 ASC 接口框图

19.1 操作概述

图 19-2 为 ASC 工作在异步和同步模式下的模块框图。

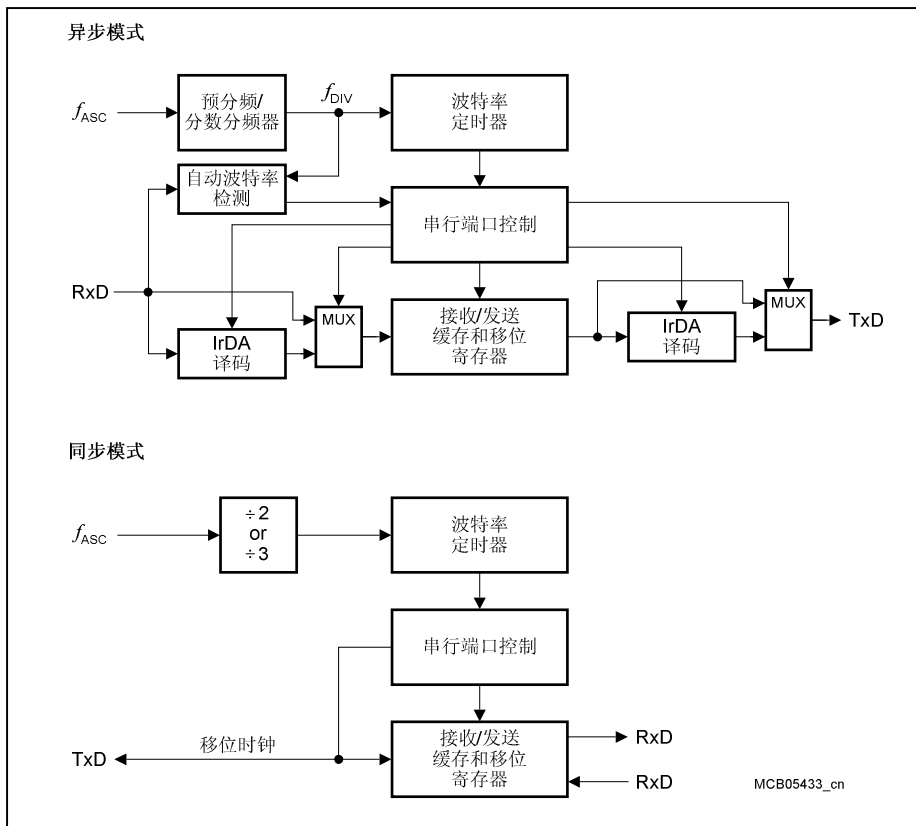


图 19-2 ASC 模块框图

ASC 支持数据率高达 2.5 Mbit/s 的全双工异步通信和数据率高达 5 Mbit/s 的半双工同步通信 (@40MHz 模块时钟)。同步模式下，发送和接收数据由微控制器产生的移位时钟同步。异步模式下，数据传送宽度 (8 位或 9 位)、是否产生奇偶校验位、以及停止位数都可由用户编程选择。ASC 提供奇偶校验错误、帧错误和过载错误检测来提高数据传送的可靠性。数据发送和接收实行双缓存。对于多处理器通信，ASC 具有数据字节和地址字节区分机制。ASC 的循环模式支持测试功能。带有多种输入时钟分频器的 13 位波特率定时器提供串行时钟信号。在特殊异步模式下，ASC 支持 IrDA 数据发送，

数据率高达 115.2 kbit/s，IrDA 脉冲宽度固定或者可编程设定。自动波特率检测功能可检测异步数据帧的波特率及模式，并自动初始化波特率发生器及模式控制位。

通过写入发送缓存寄存器 TBUF 启动数据的发送。选择的工作模式决定了实际发送数据位的个数，因而，对寄存器 TBUF 的位元 9 至位元 15 进行写操作是无意义的。因为数据发送是双缓存，所以可以在前一个字符发送结束之前向发送缓存寄存器写入新的字符。该特性支持无间隙地连续发送字符。

由接收使能位 REN 使能数据接收。在接收完一个字符之后，接收到的数据可从接收缓存寄存器 RBUF（只读类型）中读出；如果所选的工作模式支持奇偶校验，也可读出接收到的奇偶校验位。在所选工作模式下，读取 RBUF 寄存器中无效的高位位元将返回 0。

因为数据接收为双缓存，所以在先前接收到的字符未从接收缓存寄存器中读出之前，已经可以开始接收下一个字符。在所有模式下，可通过位 OEN 选择接收过载错误检测。如果该位使能，当第 9 个字符接收完成时接收缓存寄存器的内容尚未被读出，过载错误状态标志 OE 和错误中断请求线 EIR 被激活。接收缓存内先前接收的字符被覆盖。

回环模式（由位 LB 选择）允许当前发送的数据同时被接收缓存接收。该模式可用于在开发早期测试串行通信子程序，无需提供外部网络。

注：在回环模式下，不需要相关端口引脚的复用输入/输出功能。

注：仅当波特率发生器运行位 R 被置位时，才可能发送或接收串行数据。否则，串行接口空闲。

注：请不要将模式控制位 M 设置为保留组合（110_B），以避免串行接口出现不可预测的行为。

串行通道 ASC 的工作模式由其控制寄存器 ASCx_CON 控制。该寄存器包含用于模式和错误检查选择的控制位，以及用于错误类型识别的状态标志。

19.2 异步工作

异步模式支持全双工通信，发送器和接收器使用相同的数据帧格式和波特率。在 TxD 线上发送数据，在 RxD 线上接收数据。支持 IrDA 数据接收和发送，数据率可达 115.2 kbit/s。图 19-3 给出异步工作模式下的 ASC 框图。

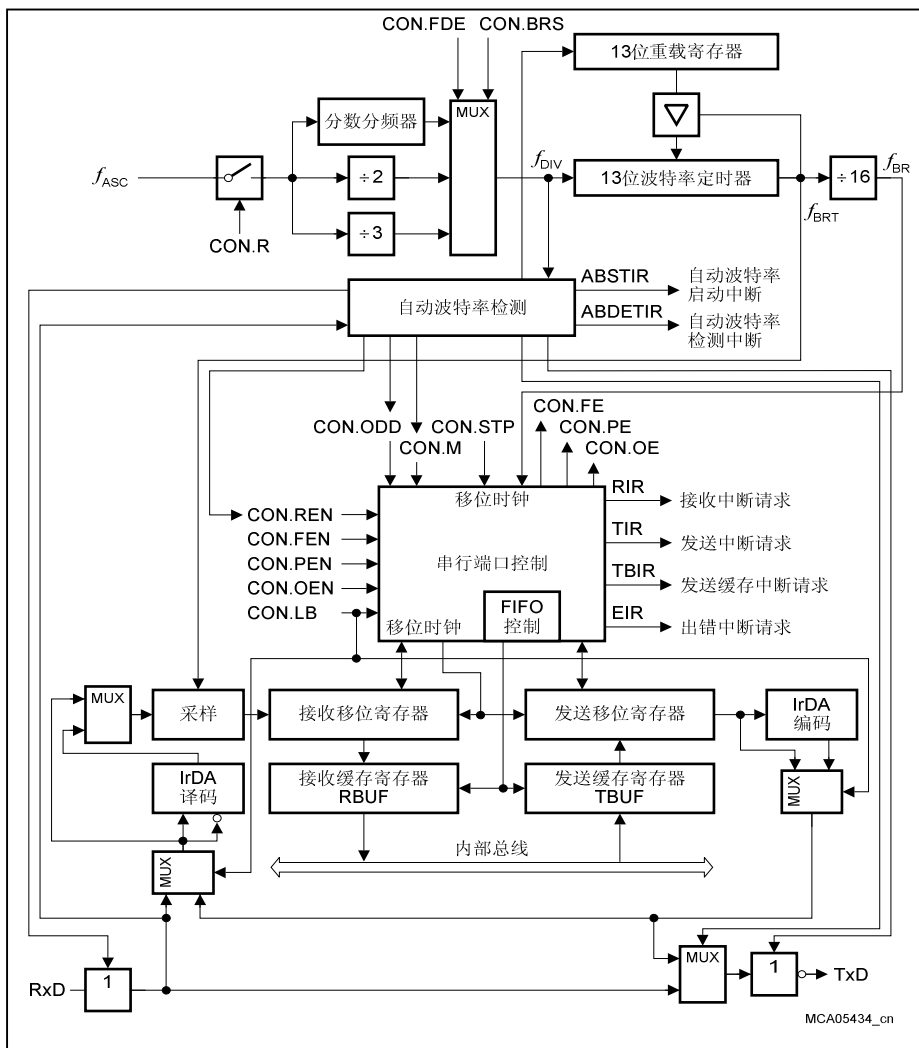


图 19-3 串行通道 ASC 的异步工作模式

19.2.1 异步数据帧

8 位数据帧

8 位数据帧包含 8 个数据位 D7...D0 ($M = 001_B$)，或者包含 7 个数据位 D6...D0 加 1 个自动产生的奇偶校验位 ($M = 011_B$)。由位 ODD 决定进行奇校验还是偶校验。如果 7 个数据位的模二和为 1，那么偶校验位置 1。同样情况下，奇校验位清零。通过位 PEN 使能奇偶校验检查（8 位数据模式下，始终关闭奇偶校验）。如果接收到错误的校验位，总是将奇偶校验错误标志 PE 和错误中断请求标志一起置位。奇偶校验位保存在位 RBUF.7 中。

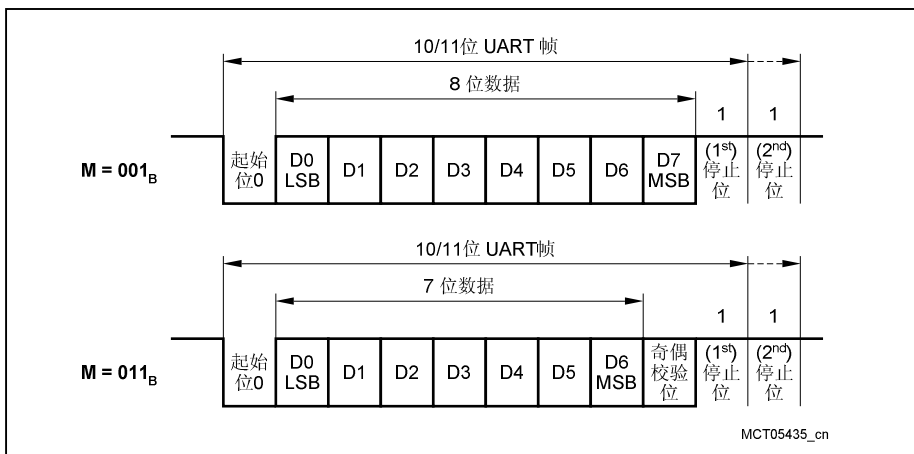


图 19-4 异步 8 位帧

9 位数据帧

9 位数据帧包含 9 个数据位 D8...D0 ($M = 100_B$) 或者 8 个数据位 D7...D0 加 1 个自动产生的奇偶校验位 ($M = 111_B$)，或者 8 个数据位 D7...D0 加 1 个唤醒位 ($M = 101_B$)。由位 ODD 决定进行奇校验还是偶校验。如果 8 个数据位的模二和为 1，那么偶校验位置 1。同样情况下，奇校验位清零。通过位 PEN 使能奇偶校验检查（9 位数据和唤醒模式下，始终关闭奇偶校验）。如果接收到错误的校验位，总是将奇偶校验错误标志 PE 和错误中断请求标志一起置位。奇偶校验位保存在位 RBUF.8 中。

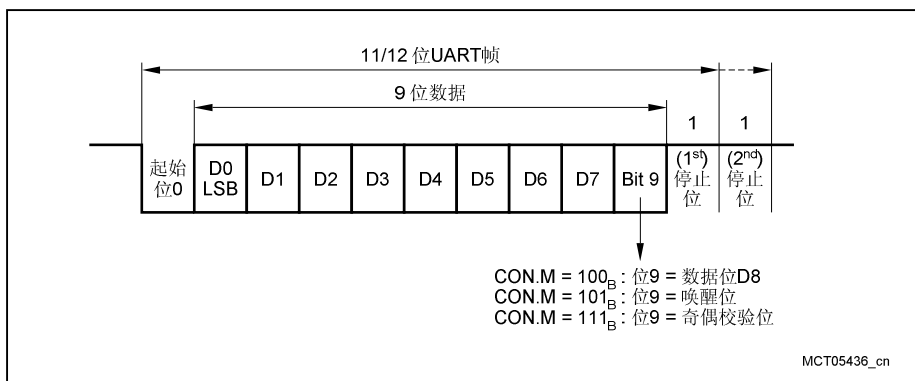


图 19-5 异步 9 位帧

在唤醒模式，仅在第 9 位（唤醒位）为 1 时，才将接收到的帧传送至接收缓存寄存器。如果该位为 0，不激活接收中断请求，不传送任何数据。

该特性可用于控制多处理器系统通信：当主机要给某一从机发送数据块时，主机首先发出地址字节以识别目标从机。地址字节和数据字节的区别是：附加的第九位数据为 1 时的字节为地址字节，为 0 时该字节为数据字节。因此，从机接收到数据‘字节’不会发生中断，而接收到地址字节，所有从机发生中断（工作在 8 位数据 + 唤醒位模式），每个从机都检查接收到字符的低 8 位（地址）。被寻址的从机将切换至 9 位数据模式（如通过清零位 M[0]实现），使该从机能够接收即将到来的数据字节（已经清零唤醒位）。未被寻址的从机仍然保持 8 位数据 + 唤醒位模式，忽略后面的数据字节。

IrDA 帧

IrDA 的调制机制基于标准异步数据发送帧。IrDA 模式（ $M = 010_B$ ）下的异步数据格式定义如下：

1 个起始位/8 个数据位/1 个停止位

对异步数据帧的编码/解码见图 19-6。总的来说，IrDA 发送期间，UART 帧被编码为 IR 帧，反之亦然（将 IR 帧解码至 UART 帧）。IR 帧上的低电平指示“LED 关闭”状态，IR 帧上的高电平指示“LED 开启”状态。

UART 帧中位元值为 0，产生一个高脉冲。UART 帧中位元值为 1，不产生脉冲。高脉冲在位单元的中间开始且宽度固定为位时间的 $3/16$ 。ASC 还可以对 IrDA 的脉宽编程。另外，IrDA 模式中接收到的 IrDA 脉冲的极性可以反相。图 19-6 给出未反相的 IrDA 脉冲产生机制。

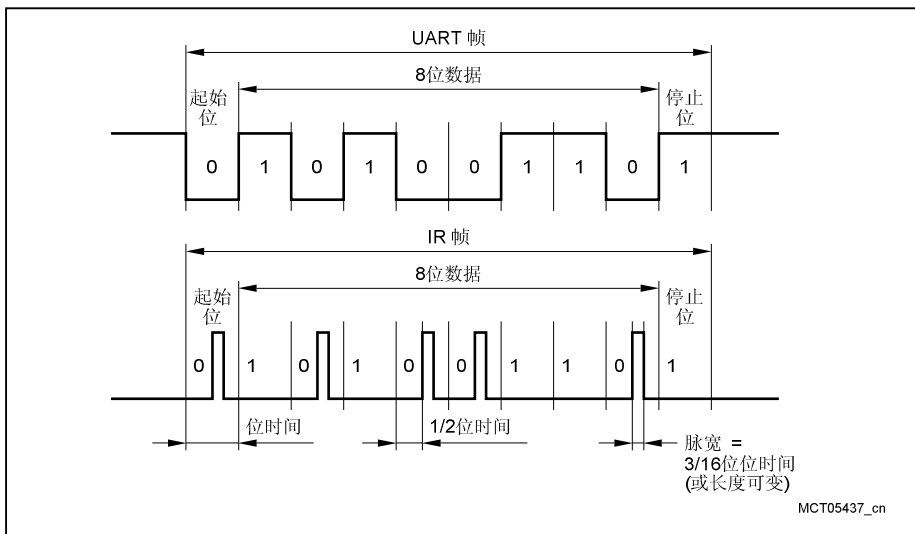


图 19-6 IrDA 帧编码/解码

ASC IrDA 脉冲模式/宽度寄存器 PMW 包含 8 位 IrDA 脉宽值和 IrDA 脉宽模式选择位。仅在 IrDA 工作模式下需要设置该寄存器。

19.2.2 异步发送

如果位 R 置位，且数据已被载入到 TBUF，在波特率定时器 16 分频下一次溢出（波特率时钟 f_{BR} 跳变）时开始异步发送。发送的数据帧包含以下 3 个基本部分：

- 起始位
- 数据域（8 或 9 位，先发送 LSB，奇偶校验位可选）
- 分隔符（1 个或 2 个停止位）

数据发送实行双缓存，当发送器空闲时，立即将发送缓存寄存器中的待发数据送入发送移位寄存器中，从而为下次要发送的数据释放发送缓存。由激活发送缓存中断请求线 TBIR 来指示该状态。连续发送先前数据的同时，可以将下一个数据载入到 TBUF 中。

发送一帧的最后一位之前，也就是说，在第 1 个或第 2 个停止位移出发送移位寄存器之前，激活发送中断请求线 TIR。

注：发送输出引脚 TxD 必须设置为复用数据输出。

19.2.3 发送 FIFO 操作

发送 FIFO（TXFIFO）提供下列功能：

- 使能/禁止控制
- 用于产生发送中断的可编程填充级别
- 填充级别指示
- FIFO 清零（擦除）操作
- FIFO 溢出错误产生

由 TXFCON 控制寄存器控制 8 级发送 FIFO。当置位 TXFEN 时，使能发送 FIFO。由 TXFITL 设定的中断触发级别决定 TXFIFO 的填充级别，在该级别上产生发送缓存中断 TBIR 或发送中断 TIR。当发送 FIFO 的填充级别等于或者低于保存在 TXFITL 中的值时，总会产生上述中断。

FIFO 状态寄存器 ASCx_FSTAT 中的位域 TXFFL 指示实际被写入（有效的）TXFIFO 的字节个数。因此，在中断服务子程序中，可由软件验证，比如，还有几个字节可从寄存器 TBUF 写入到发送 FIFO 中，而不产生过载错误。

不能直接访问发送 FIFO。所有 TXFIFO 的数据写操作都通过写入到 TBUF 寄存器来实现。

通过置位寄存器 ASCx_TXFCON 位 TXFFLU，可以擦除或清零 TXFIFO。在 TXFIFO 擦除操作之后，TXFIFO 为空且发送 FIFO 填充级别设置为 0000_B。接收 FIFO 的擦除操作不会中止正在进行的串行发送操作。

注：ASC 模块复位后和 TXFIFO 先前被使能，又被禁止（复位 TXFEN）两种情况下，TXFIFO 会被自动擦除。

19.2.4 异步接收

如果位 R 和 REN 被置位，由 RxD 线上的下降沿（1 到 0 的跳变）启动异步接收过程。对接收数据线 RxD 进行采样，采样率为所设波特率的 16 倍。采用多数决定法，由第 7、第 8 和第 9 个采样值确定有效位值。这样就避免了可能由噪声引起的错误结果。

当起始位被采样时，如果检测到的值不为 0，接收电路复位并等待 RxD 线上的下一次 1 至 0 的跳变。一旦证明起始位有效，接收电路连续采样，并将接收的数据帧移入接收移位寄存器。

当接收到最后一个停止位，接收移位寄存器的内容被传送到接收数据缓存寄存器 RBUF。与此同时，无论是否接收到有效的停止位，在最后一个停止位时隙（如编程设定的那样）中的第 9 个采样值之后激活接收中断请求线 RIR。随后，在接收数据输入线上，接收电路等待下一个起始位（1 到 0 的跳变）。

注：接收器输入引脚 RxD 必须配置为输入。

清零位 REN 将停止异步接收过程。完成当前帧接收，包括产生接收中断请求和错误中断请求（如果需要的话）。不识别该帧之后的起始位。

注：唤醒模式下，仅在第 9 位（唤醒位）为 1 时，接收到的帧才被传送到接收缓存寄存器。如果该位为 0，不会激活接收中断请求，也不会进行数据传送。

19.2.5 接收 FIFO 操作

接收 FIFO（RXFIFO）提供下列功能：

- 使能/禁止控制
- 用于产生接收中断的可编程填充级别
- 填充级别指示
- FIFO 清零（擦除）操作
- FIFO 溢出错误产生

由控制寄存器 RXFCON 控制 8 级接收 FIFO。当置位 RXFEN 时，使能接收 FIFO。由 RXFITL 设定的中断触发级别决定产生接收中断 RIR 的 RXFIFO 填充级别。当填充级别等于或者大于保存在 RXFITL 中的值时，总会产生 RIR。

FIFO 状态寄存器 ASCx_FSTAT 的位域 RXFFL 指示实际写入 FIFO，并可由用户读出的字节个数。

不能直接访问接收 FIFO。所有对 RXFIFO 的读操作都通过读 RBUF 寄存器实现。

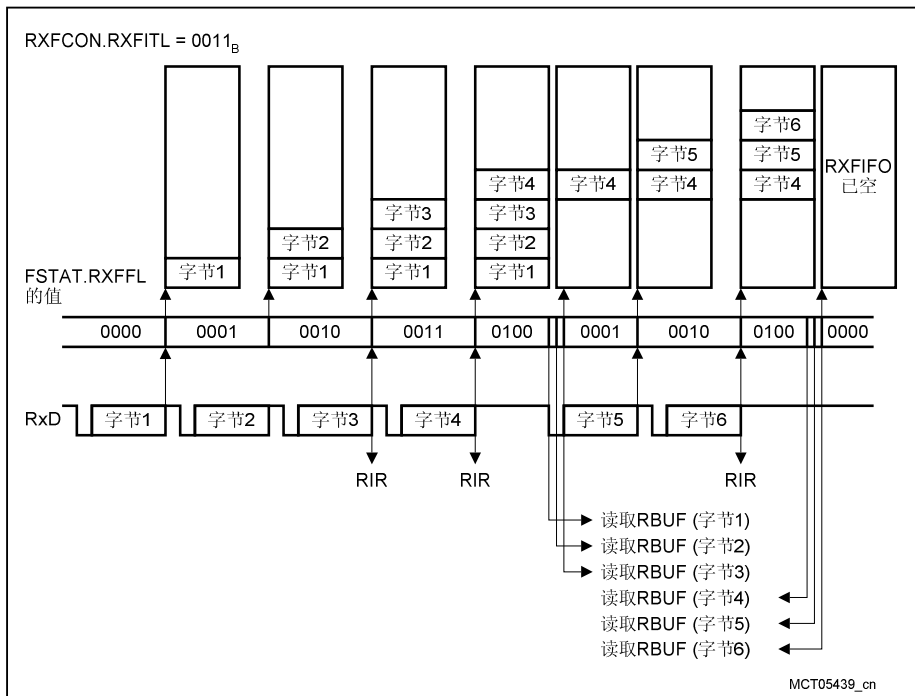


图 19-8 接收 FIFO 操作示例

图 19-8 中的例子为典型的 8 级接收 FIFO 操作。在该例中，通过 RxD 输入线接收 6 个字节。接收 FIFO 中断触发级别设置为 0011_B。因此，在接收字节 3 之后（RXFIFO 中已经填充了 3 个字节），产生第一个 RIR 接收中断。

在接收字节 4 之后，从 FIFO 中读出 3 个字节。该次读操作之后，RXFIFO 中仍有一个字节。在接收另外两个字节（字节 5 和 6）之后（RXFIFO 再次填充有 3 个字节），RIR 再次被激活。最后，在 3 次读操作之后，FIFO 为空。

如果 RXFIFO 填满之后又收到新字节，产生接收中断 RIR 和错误中断 EIR（且位 OE 置位）。这种情况下，最后写入接收 FIFO 的数据字节被覆盖。在过载情况下，接收 FIFO 填充级别 RXFFL 设置为最大。如果 RXFIFO 使能但是为空时，执行 RBUF 读操作，OE 置位并产生错误中断 EIR。在这种情况下，接收 FIFO 填充级别设置为 0000_B。

如果 RXFIFO 可供使用，但是被禁止（RXFEN = 0）而接收操作被使能（REN = 1），异步接收操作功能和 ASC 模块的异步接收操作功能相同。

置位寄存器 RXFCON 中的位 RXFFLU，可擦除或清零 RXFIFO。该操作之后，RXFIFO 为空且接收 FIFO 填充级别置为 0000_B。

ASC 模块的复位操作，以及复位 RXFEN 使 RXFIFO 变为禁止的情况下（先前 RXFIFO 被使能），都会自动擦除 RXFIFO。复位 REN 而不复位 RXFEN 对 RXFIFO 状态无影响（复位）。这意味着 ASC 的接收操作被停止，这种情况不改变 RXFIFO 的内容。再次置位 REN，RXFIFO 的内容再次可供使用。

注：在一次成功的自动波特率检测之后（如果实现），在接收数据之前应当擦除 RXFIFO。

19.2.6 FIFO 透明模式

在透明模式下，用特定的中断产生机制产生接收和发送缓存中断。总的来说，在透明模式下，如果 RXFIFO 内有数据，总会产生接收中断。如果 TXFIFO 不满，则总会产生发送缓存中断。该模式下，与中断产生相关的条件如下：

- FIFO 填充级别
- 数据寄存器 RBUF/TBUF 的读/写操作

接收 FIFO 的中断产生取决于 RXFIFO 填充级别和寄存器 RBUF 的读操作。（见 [图 19-9](#)）。当寄存器 ASCx_RXFCON 中的 RXTMEN 和 RXFEN 被置位时，使能 RXFIFO 的透明模式。

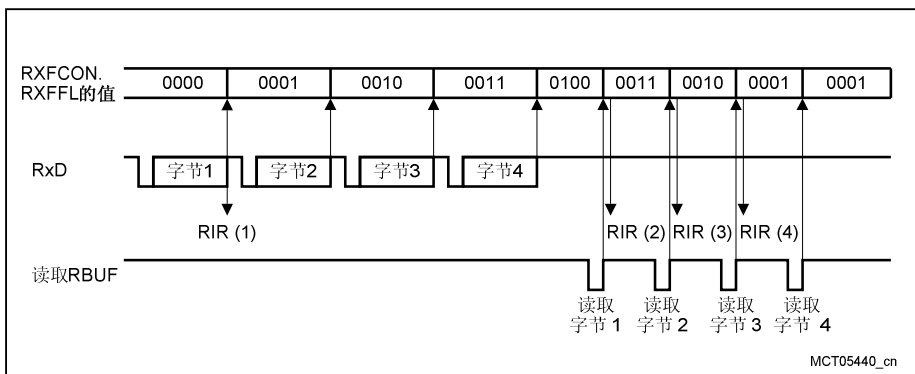


图 19-9 接收 FIFO 操作透明模式

如果 RXFIFO 为空，当第一个字节被写入到空 RXFIFO 中时（RXFFL 从 0000_B 变为 0001_B），总是产生接收中断 RIR。如果 RXFIFO 内至少有一个字节，接收中断的产生取决于寄存器 RBUF 的读操作。如果在一次 RXFIFO 读操作之后，RXFIFO 中仍然有数据（RXFFL 不等于 0000_B），接收中断 RIR 总是被激活。如果在一次 RBUF 读操作之后 RXFIFO 为空，将不再产生接收中断。

如果 RXFIFO 已满（RXFFL = 最大值）且又接收到一个字节，产生错误中断 EIR 并置位 OE。这种情况下，最后写入的数据字节被覆盖。如果 RXFIFO 被使能但是为空（下溢条件）时执行 RBUF 操作，同样会产生错误中断 EIR 并置位 OE。

如果透明模式下，RXFIFO 被擦除，用户必须注意，先前未被处理的接收中断将被忽略。

注：在透明模式下，接收 FIFO 中断触发级别位域 RXFITL 无效。

发送 FIFO 的中断产生取决于 TXFIFO 填充级别和对寄存器 TBUF 写操作的执行。当置位 TXTMEN 和 TXFEN 时，TXFIFO 透明模式被使能。

在给寄存器 `ASCx_TBUF` 写入一个字节之后，`TXFIFO` 仍未满的情况下（`TXFFL` 不等于最大值），总是产生发送缓存中断 `TBIR`。`TXFIFO` 擦除操作之后，或在 `TXFIFO` 先前被禁止，当前又被使能（`TXTMEN` 和 `TXFEN` 置位）的情况下，`TBIR` 也被激活。这些情况下，`TXFIFO` 为空，准备好进行数据填充。

如果 `TXFIFO` 已满（`TXFFL` = 最大值）且又有一个字节写入到 `TBUF` 中，`TBUF` 写操作之后，不会产生发送缓存中断。这种情况下，最后写入发送 `FIFO` 的数据字节被覆盖，且 `OE` 置位，产生一个过载错误中断。

注：在透明模式下，发送 `FIFO` 中断触发级别位域 `TXFITL` 无效。

19.2.7 IrDA 模式

IrDA 脉冲的持续时间通常为 $3/16$ 的位周期。IrDA 标准也允许脉冲持续时间跟波特率或位周期无关。这种情况下，发送脉冲宽度等于 115.2 kbit/s 时 $3/16$ 的脉冲宽度，为 $1.627 \mu\text{s}$ 。用户可编程选择 IrDA 的脉宽为固定值或者和位周期相关。位 `IRPW` 选择 IrDA 脉冲宽度模式。

选择固定脉宽时，寄存器 `PMW` 的低 8 位用于将 IrDA 脉冲宽度调整为一个固定值 $1.627 \mu\text{s}$ 。固定 IrDA 脉冲宽度由可编程的计数器产生，见图 19-10。

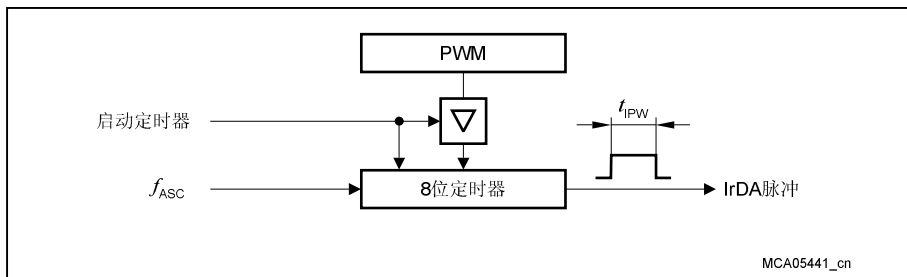


图 19-10 固定 IrDA 脉冲产生

IrDA 宽度根据表 19-1 给出的公式计算。

注：表 19-1 中 `PMW` 代表脉冲模式/宽度寄存器 `PMW` (`PW_VALUE`) 的内容，为 8 位无符号整数。

表 19-1 IrDA 脉冲宽度计算公式

PMW	PMW_IPMW	公式	
1...255	0	$t_{IPW} = \frac{3}{16 \times \text{波特率}}$	$t_{IPW \min} = \frac{(PMW >> 1)}{f_{ASC}}$
	1	$t_{IPW} = \frac{PMW}{f_{ASC}}$	

PW_VALUE 的内容还定义了最小 IrDA 脉冲宽度 ($t_{IPW \min}$)，在接收操作期间仍可被识别为有效的 IrDA 宽度。该功能与位 IRPW 设定的 IrDA 宽度模式（固定或可变）无关。将 PMW 的位域 7-0 右移 1 位，再除以模块时钟 f_{ASC} ，得到最小的 IrDA 脉冲宽度。

注：如果 IRPW 清零（固定 IrDA 脉冲宽度），PW_VALUE 值的选取必须保证

$$t_{IPW} > t_{IPW \min}。$$

表 19-2 给出 f_{ASC} 的典型频率值。

表 19-2 IrDA 脉冲宽度调整至 1.627μs

f_{ASC}	PMW	t_{IPW}	误差	$t_{IPW \min}$
20 MHz	33	1.650 μs	+1.4%	0.8 μs
40 MHz	65	1.625 μs	-0.1%	0.8 μs

19.2.8 异步模式中 RxD/TxD 数据路径选择

由寄存器 CON 和 ABCON 中的几个控制位决定异步模式下串行数据输入和输出的数据路径，见图 19-11。同步操作模式不受数据路径选择的影响。

RxD 输入的信号经过由位 RXINV 控制的反相器。反相器输出的信号用于自动波特率检测，该信号在回声模式（由位 ABEM 控制）下被旁路。另外，RxD 输入信号路径上的两个复用器提供回声模式功能（由位 LB 控制）和 IrDA 接收脉冲反相功能（由位 RxDI 控制）。

根据所设定的异步模式（由位域 M 控制），输出信号或回声模式下的 RxD 输入信号（由位 ABEM 控制）通过一个反相器（由位 TXINV 控制）被切换至 TxD 输出。

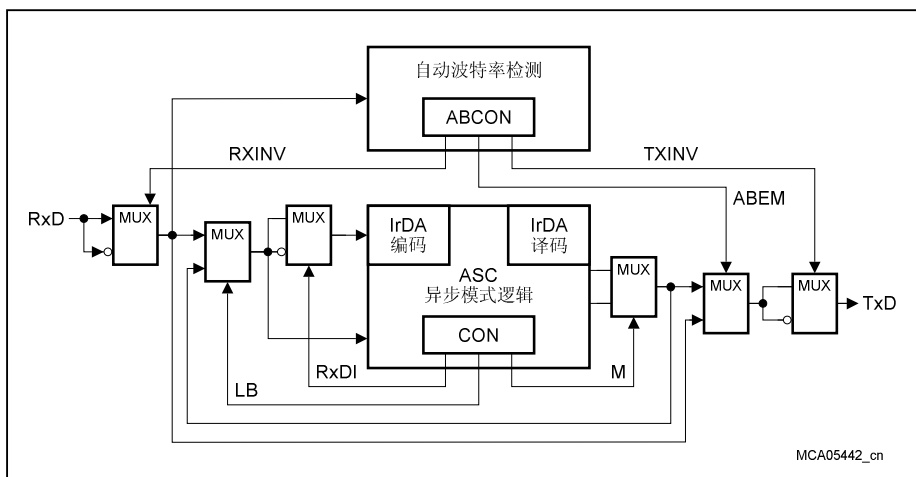


图 19-11 异步模式下 RxD/TxD 数据路径

注：在回声模式，回声模式输出复用器阻止发送输出信号。图 19-11 所示，经 IrDA 编码的接收输入信号不能用于自动波特率检测。

19.3 同步操作

同步操作支持半双工通信，基本上通过移位寄存器进行简单的 IO 扩展。通过线 RxD 接收和发送数据，同时通过 TxD 输出移位时钟。

通过设置位域 $M = 000_B$ 来选择同步模式。

8 个数据位的接收和发送与内部波特率发生器产生的移位时钟同步。只要进行数据发送或接收，移位时钟就一直有效。

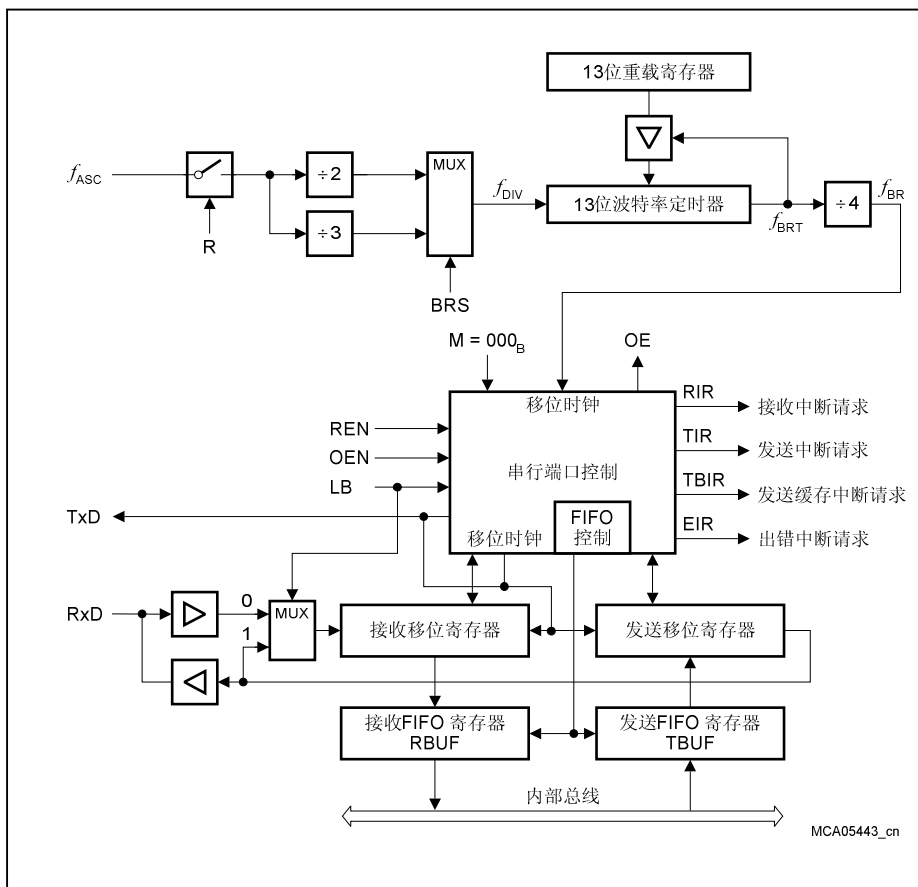


图 19-12 串行通道 ASC 同步模式

19.3.1 同步发送

如果置位 R 且清零位 REN（半双工，无接收），数据被载入 TBUF 之后，在四个状态时间内开始同步发送。例外情况：回环模式下（位 LB 置位），要接收的发送数据字节必须置位 REN。数据发送为双缓存。当发送器空闲，载入 TBUF 中的发送数据立即被移入发送移位寄存器中，从而为更多的数据释放 TBUF 空间。该状态由激活发送缓存中断请求线 TBIR 来指示。TBUF 在连续发送数据的同时，可被载入新数据。数据位的发送和移位时钟同步。在第 8 个数据位的位时间之后，TxD 和 RxD 线都将变为高。发送中断请求线 TIR 被激活，串行数据发送停止。

注：引脚 TxD 必须配置为复用数据输出，提供移位时钟。在发送期间引脚 RxD 必须配置为输出。

19.3.2 同步接收

置位 REN 启动同步接收。如果位 R 被置位，RxD 上的数据以与 TxD 输出的时钟同步的节奏移入接收移位寄存器中。在移入第 8 位之后，接收移位寄存器中的内容被传送到接收数据缓存 RBUF，接收中断请求线 RIR 被激活，接收使能位 REN 被复位，串行数据接收停止。

注：引脚 TxD 必须配置为复用数据输出，提供移位时钟。引脚 RxD 必须配置为复用数据输入。

通过清零位 REN 停止同步接收。如果必要的话，完成当前字节的接收，包括产生接收中断请求和错误中断请求。接收过程中向发送缓存寄存器的写操作对接收过程无任何影响，且不会启动发送过程。

如果已经使能过载检测（置位 OEN），先前接收的字节还未从已满的接收缓存中读出，下一个字节的接收已经完成，那么错误中断请求线 EIR 和过载错误状态标志 OE 将被激活/置位。

19.3.3 同步时序

图 19-13 给出了 ASC 同步模式数据接收和数据发送时序图。在空闲状态下，移位时钟电平为高。开始同步发送数据字节时，在移位时钟的下降沿，数据移出到 RxD 线上。如果通过 RxD 接收数据字节，在移位时钟的上升沿锁存数据。

在连续接收或发送 2 个数据字节之间，插入一个移位时钟周期（f_{BR}）延迟。

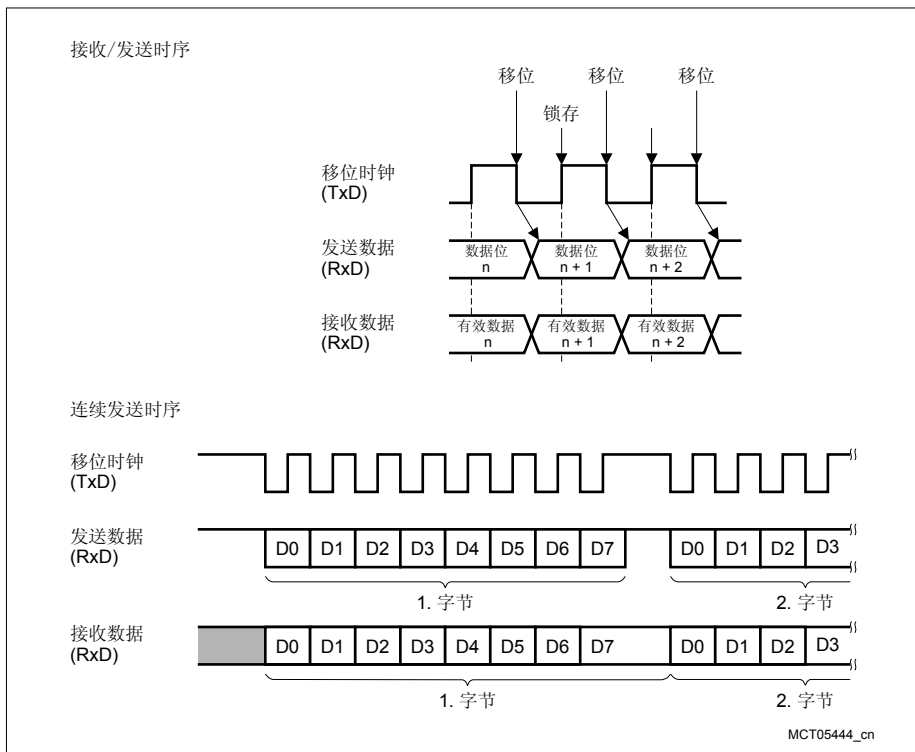


图 19-13 ASC 同步模式波形

19.4 波特率产生

串行通道 ASC 带有专用可重载 13 位波特率发生器，其波特率的产生不依赖于其它定时器。

ASC 输入时钟 f_{ASC} 经过预分频得到波特率发生器的时钟 (f_{DIV})。波特率定时器递减计数，可以通过波特率发生器运行位 **R** 来启动或停止波特率定时器的运行。定时器每次下溢为串行通道提供一个时钟脉冲。每次下溢时由保存在重载寄存器中的 13 位值重载定时器。由此获得的时钟 f_{BRT} 再次被分频（分频因子：异步模式中为 16，同步模式中为 4）得到波特率时钟。由位 **BRS** 和 **FDE** 选择预分频器。除了 2 个固定分频器，在异步模式下还提供一个分数预分频器，分频因子为 $n/512$ ，其中 $n = 0 \dots 511$ 。因此，ASC 模块的波特率由模块时钟，**FDV** 的内容，**BG** 重载值和工作模式决定（同步或异步）。

寄存器 **ASCx_BG** 为双功能波特率发生器/重载寄存器。读 **ASCx_BG** 返回定时器 **BR_VALUE** 的内容（位 15...13 返回 0），对 **BG** 进行写操作总会更新重载寄存器（位 15...13 无意义）。

每次对 **ASCx_BG** 进行写操作都执行定时器自动重载，重载值为重载寄存器的内容。然而，如果执行 **ASCx_BG** 写操作的同时，位 **R** 被清零，那么，直到位 **R** 置位后的第一个指令周期，才能执行定时器重载。为了实现正确的波特率初始化，应当仅在 **R = 0** 时，才进行 **ASCx_BG** 写操作。如果当 **R = 1** 时写入 **ASCx_BG**，ASC 模块的发送和接收操作可能会出现不可预测的行为。

ASC 波特率定时器重载寄存器 **ASCx_BG** 定义了异步和同步模式下波特率定时器的 13 位重载值。

19.4.1 异步模式下的波特率产生

对于异步模式，波特率发生器提供的时钟 f_{BRT} 是所建立的波特率的 16 倍。每位接收数据都在该时钟的第 7，8 和 9 个周期上被采样。用于产生 13 位波特率定时器输入时钟的分频电路，通过一个分数分频电路进一步扩展功能，使得波特率的调整更加精确，从而扩展了波特率的范围。

波特率由下列位和寄存器的值决定：

- 输入时钟 f_{ASC}
- 由位 **FDE** 和 **BRS** 选择波特率定时器输入时钟 f_{DIV}
- 如果 **FDE** 置位（分数分频器），寄存器 **ASCx_FDV** 的值
- 13 位重载寄存器 **ASCx_BG** 的值

带重载寄存器的波特率定时器的输出时钟作为 ASC 异步模式下的采样时钟。对于波特率计算，采样时钟 $f_{DIV}16$ 分频得到波特率时钟 f_{BR} 。

ASC 分数分频寄存器 **ASCx_FDV** 包含 9 位分频值（仅在异步模式下有效）。该寄存器还可用来产生自动波特检测单元参考时钟。

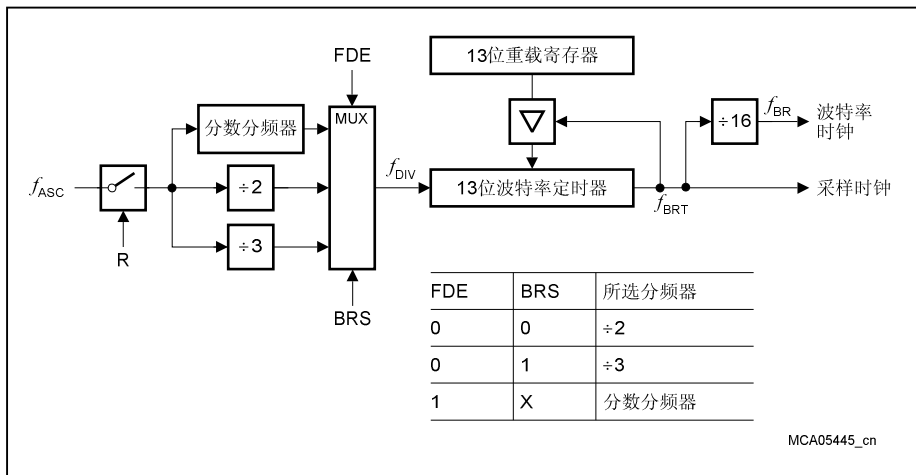


图 19-14 异步模式下 ASC 波特率产生电路

使用固定输入时钟分频器

串行通道 ASC 的异步操作中，当使用固定输入时钟分频因子（ $FDE = 0$ ）时，其波特率和给定波特率所需的重载值由下面的公式决定。

BG 为重载位域 BR_VALUE 的内容，为 13 位无符号整数。

当使用两个固定时钟分频器，模块时钟为 40MHz 时，异步模式所能达到的最大波特率为 1.25 Mbit/s。表 19-4 给出各种常见的波特率和所需的重载值，以及和期望波特率的相对偏差。

注：表 19-3 中 FDE 必须为 0。表中给出的偏差是经过舍入得到的值。利用波特率晶体振荡器将得到更精确的波特率值，而无偏差。

表 19-3 使用固定输入时钟分频器的异步波特率计算公式

FDE	BRS	BG	公式
0	0	0...8191	$\text{波特率} = \frac{f_{ASC}}{32 \times (BG + 1)}$ $BG = \frac{f_{ASC}}{32 \times \text{波特率}} - 1$
	1		$\text{波特率} = \frac{f_{ASC}}{48 \times (BG + 1)}$ $BG = \frac{f_{ASC}}{48 \times \text{波特率}} - 1$

表 19-4 使用固定输入时钟分频器的异步波特率典型值

波特率	BRS = 0, f _{ASC} = 40MHz		BRS = 1, f _{ASC} = 40MHz	
	偏差	重载值	偏差	重载值
1.25 Mbit/s	----	0000 _H	不适用	不适用
19.2 kbit/s	+0.1% / -1.3%	0040 _H / 0041 _H	+0.9% / -1.3%	002A _H / 002B _H
9600 bit/s	+0.1% / -0.6%	0081 _H / 0082 _H	+0.9% / -0.2%	0055 _H / 0056 _H
4800 bit/s	+0.1% / -0.2%	0103 _H / 0104 _H	+0.3% / -0.2%	00AC _H / 00AD _H
2400 bit/s	+0.1% / -0.0%	0207 _H / 0208 _H	+0.0% / -0.2%	015A _H / 015B _H
1200 bit/s	+0.0% / -0.0%	0410 _H / 0411 _H	+0.0% / -0.0%	02B5 _H / 02B6 _H

使用分数分频器

当选择分数分频器时，波特率定时器的输入时钟 f_{DIV} 是模块时钟 f_{ASC} 经分频处理得到（分频因子可编程设定）。如果 FDE 置位，激活分数分频器。其将 f_{ASC} 进行 n/512 分频，n 值可从 0 到 511。如果 n = 0，分频因子为 1，意味着 f_{DIV} = f_{ASC}。总的来说，用分数分频器得到的波特率，比用 2 个固定预分频器所得的波特率精确得多。

注：BG 代表重载位域 BR_VALUE 的内容，为 13 位无符号整数。

注：FDV 代表分数分频寄存器 FD_VALUE 中的内容，为 9 位无符号整数。

表 19-5 使用分数输入时钟分频器的异步波特率计算公式

FDE	BRS	BG	FDV	公式
1	-	0...8191	1...511	$\text{波特率} = \frac{\text{FDV}}{512} \times \frac{f_{\text{ASC}}}{16 \times (\text{BG} + 1)}$
			0	$\text{波特率} = \frac{f_{\text{ASC}}}{16 \times (\text{BG} + 1)}$

表 19-6 使用分数输入时钟分频器的异步波特率典型值

f _{ASC}	期望波特率	BG	FDV	实际波特率	偏差
40 MHz	115.2 kbit/s	04 _H	076 _H	115.234 kbit/s	0.02%
	57.6 kbit/s	04 _H	03B _H	57.617 kbit/s	0.02%
	38.4 kbit/s	0E _H	076 _H	38.411 kbit/s	0.02%
	19.2 kbit/s	0E _H	03B _H	19.206 kbit/s	0.02%

19.4.2 同步模式下的波特率产生

对于同步操作，波特率发生器提供的时钟为波特率的 4 倍（见图 19-15）。

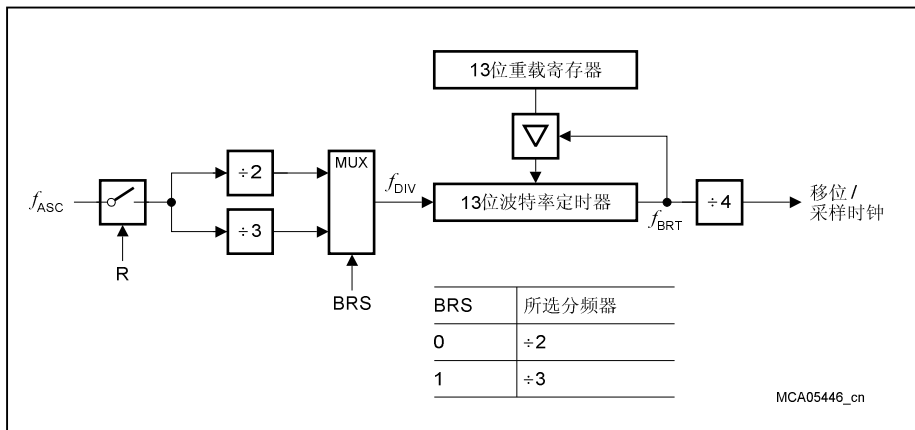


图 19-15 同步模式下 ASC 波特率产生电路

串行接口 ASC 同步操作的波特率由表 19-7 中的公式决定。

表 19-7 同步波特率计算公式

BRS	BG	公式
0	0...8191	$\text{波特率} = \frac{f_{ASC}}{8 \times (BG + 1)} \quad BG = \frac{f_{ASC}}{8 \times \text{波特率}} - 1$
1		$\text{波特率} = \frac{f_{ASC}}{12 \times (BG + 1)} \quad BG = \frac{f_{ASC}}{12 \times \text{波特率}} - 1$

注：BG 代表重载位域 BR_VALUE 中的内容，为 13 位无符号数。

模块时钟为 40 MHz 时，同步模式下所能达到的最大波特率为 5 Mbit/s。

19.5 自动波特率检测

19.5.1 操作概述

自动波特率检测能够识别 RxD 上异步输入信号的模式和波特率。一般而言，需要检测的波特率必须被应用端识别。经过事先定义，自动波特率检测能够识别 9 种常见的波特率格式。自动波特率检测不是为计算未知异步帧波特率而设计的。

图 19-16 示出自动波特率检测如何在异步模式配置下工作。RxD 数据线为自动波特率检测单元的输入。分数分频器产生的时钟 f_{DIV} ，用作自动波特率检测单元的时间基准。成功识别 RxD 上数据输入信号的波特率和异步模式之后，对波特率定时器中的寄存器 ASCx_CON 和寄存器 ASCx_BG 进行相应设置，ASC 模块可以立即启动串行输入数据的接收。

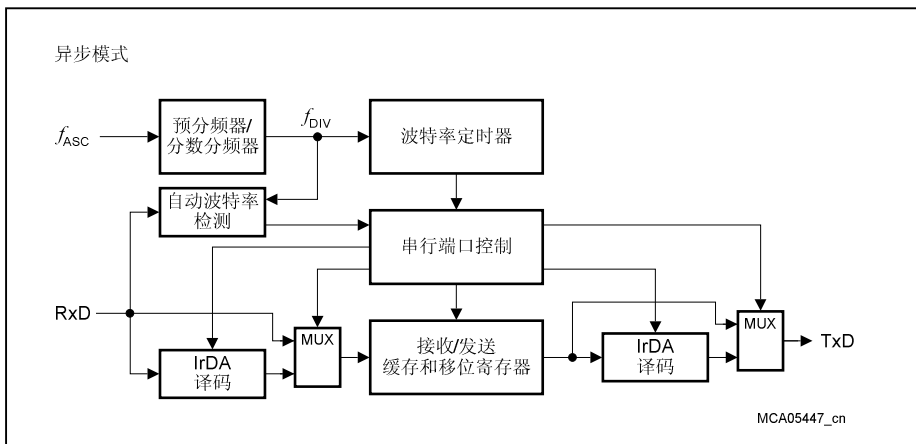


图 19-16 异步模式框图

注：同步模式中自动波特率检测功能不可用。

下面给出自动波特率检测单元启动步骤：

- 定义需要检测的波特率：标准或非标准波特率
- 编程预分频器/分数分频器，选择一个特定 f_{DIV}
- 启动预分频器/分数分频器（置位位 R）
- 中断系统准备
- 使能自动波特率检测（置位 ABCON 的 ABEN，如有需要还可使能中断使能位）
- 查询中断请求标志或等待自动波特率检测中断

19.5.2 串行帧自动波特率检测

自动波特率检测基于一个特定 2 字节串行帧的串行接收。该串行帧由 2 个 ASCII 字节“at”或“AT”（全为大写或全为小写）构成（不允许出现“aT”或“At”大小写混合）。在 5 种类型的异步帧中，这两种字节组合都可检测到。图 19-17 和图 19-18 给出必须检测到的两个串行帧。

注：也将定义其它的 2 字节组合。

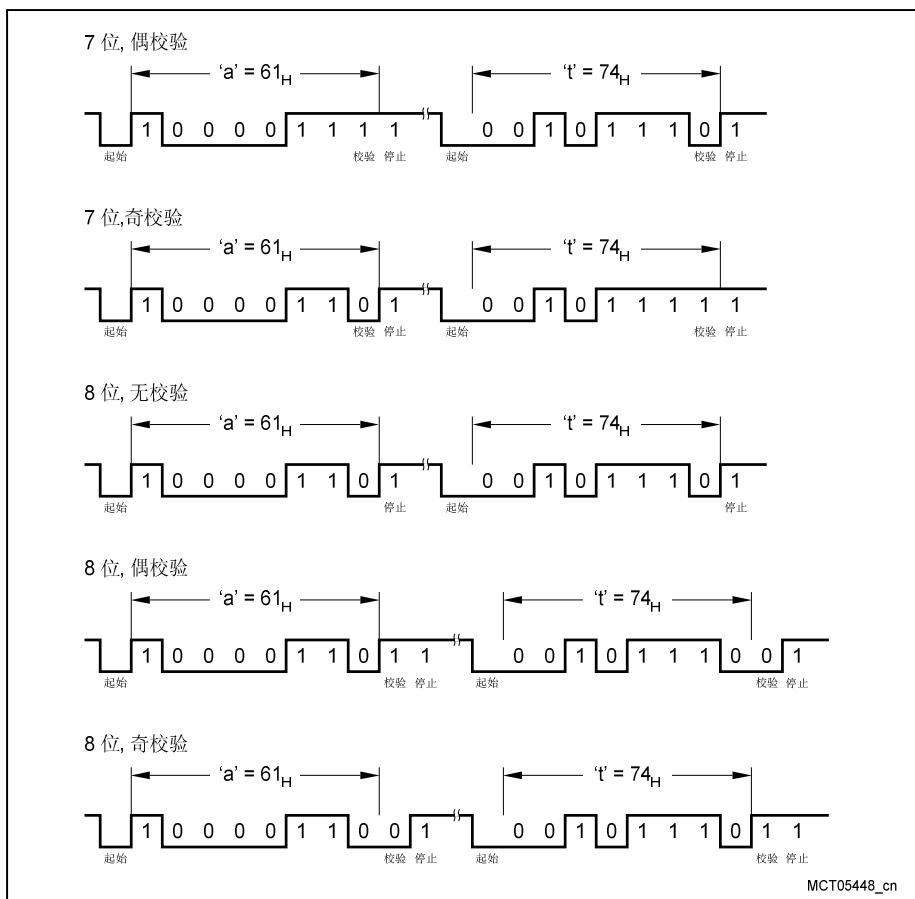


图 19-17 ASCII ‘at’ 2 字节串行帧

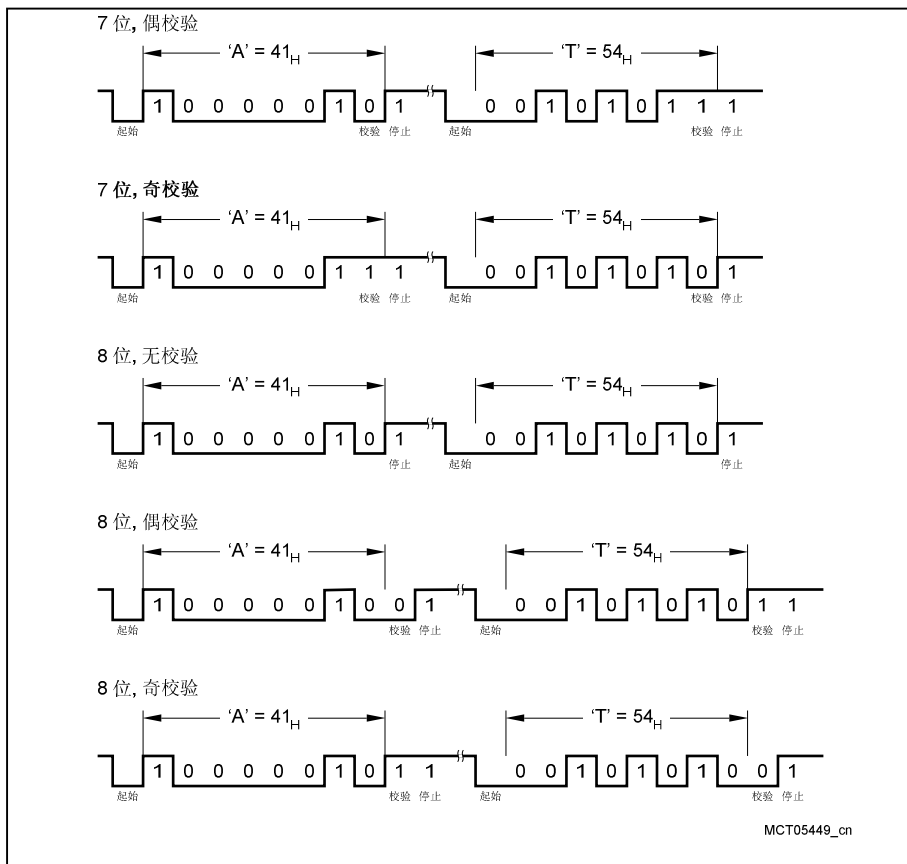


图 19-18 ASCII ‘AT’ 2 字节串行帧

19.5.3 波特率选择和计算

自动波特率检测需要通过一些计算来设置波特率发生器以及将要检测的波特率。必须考虑采用下面的 2 个步骤：

- 定义要检测的波特率
- 编程波特率定时器预分频器 – 设置 f_{DIV}

通常，异步模式下波特率发生器的设置分为两部分（见图 19-14）：

- 时钟预分频部分，从 f_{ASC} 中获得 f_{DIV}
- 波特率定时器部分，产生采样时钟 f_{BRT} 和波特率时钟 f_{BR}

自动波特率检测之前，必须由 CPU 设置预分频部分，在一次成功自动波特率检测之后，波特率定时器（寄存器 $ASCx_BG$ ）会自动加载一个 13 位的值（ BR_VALUE ）完成初始化。下面的计算中，使用分数分频器（ $FDE = 1$ ）。

注：也可以采用固定的 1/2 或 1/3 分频器。不过分数分频器可更加精确地将 f_{DIV} 调整至所需要的值。

标准波特率

对于标准波特率检测，表 19-8 给出了可检测的波特率值。因此，必须设法使从模块时钟 f_{ASC} 获得的波特率发生器的输出频率 f_{DIV} 等于 11.0592 MHz。写入寄存器 FDV 的值是与根据下面公式计算得到的数值最接近的整数：

$$FDV = \frac{512 \times 11.0592 \text{ MHz}}{f_{ASC}} \quad (19.1)$$

$f_{DIV} = 11.0592 \text{ MHz}$ 时，表 19-8 中定义了 9 个可被检测的标准波特率（ $Br0 - Br8$ ）。

表 19-8 利用标准波特率的自动波特率检测 ($f_{DIV} = 11.0592 \text{ MHz}$)

波特率编号	可检测到的标准波特率	分频因子 d_f	检测后 BG 载入值
Br0	230,400 bit/s	48	$2 = 002_H$
Br1	115,200 bit/s	96	$5 = 005_H$
Br2	57,600 bit/s	192	$11 = 00B_H$
Br3	38,400 bit/s	288	$17 = 011_H$
Br4	19,200 bit/s	576	$35 = 023_H$
Br5	9,600 bit/s	1152	$71 = 047_H$
Br6	4,800 bit/s	2304	$143 = 08F_H$
Br7	2,400 bit/s	4608	$287 = 11F_H$
Br8	1,200 bit/s	9216	$575 = 23F_H$

根据表 19-8，假设 f_{DIV} 已被设置为 11.0592 MHz，当寄存器 ASCx_BG 的载入值为 047_H 时，波特率达到 9600 bit/s。

表 19-8 给出了分频因子 d_f 列表，它由下面的公式计算得到：

$$\text{波特率} = \frac{f_{DIV}}{d_f} \quad (19.2)$$

该分频因子 d_f 定义了自动波特率检测期间，预分频输出频率 f_{DIV} 和要检测的波特率之间的**固定**关系。这意味着，改变 f_{DIV} 将得到波特率值完全不同的波特率表。对于要检测的波特率，下列关系始终有效：

$$\text{Br0} = f_{DIV} / 48_D, \text{ Br1} = f_{DIV} / 96_D, \dots, \text{ 直到 } \text{Br8} = f_{DIV} / 9216_D$$

检测高达 230.400 kbit/s 的标准波特率时，需要的 f_{DIV} 最小值为 11.0592 MHz。根据模块时钟频率 f_{ASC} ，通过设置 FD_VALUE 调整分数分频器 f_{DIV} 的值。表 19-9 定义了依据不同模块时钟 f_{ASC} 进行自动波特率检测，所获得的标准波特率偏差（误差）。

表 19-9 标准波特率 – 自动波特率检测偏差

f_{ASC}	FDV	f_{DIV} 的误差
10 MHz	不可能出现此情况	
12 MHz	472	+0.03%
13 MHz	436	+0.1%
16 MHz	354	+0.03%
18 MHz	315	+0.14%
18.432 MHz	307	-0.07%
20 MHz	283	-0.04%
24 MHz	236	+0.03%
25 MHz	226	-0.22%
30 MHz	189	+0.14%
33 MHz	172	+0.24%
40 MHz	142	+0.31%

注：如果自动波特率检测之后偏差过高，如需要得到更精确（误差更小）的波特率，可重新对波特率发生器进行编程。

非标准波特率

根据表 19-8 中 Br0 至 Br8 和分频因子 d_f 的关系，还可以检测非标准的波特率，例如，如果要检测的波特率为 50 kbit/s，比如定义 Br2 对应 50 kbit/s 的波特率选择，进一步可得到下面的结果：

$$f_{DIV} = 50 \text{ kbit/s} \times d_f @ Br2 = 50 \text{ kbit/s} \times 192 = 9.6 \text{ MHz}$$

因此，根据模块时钟频率 f_{ASC} ，本例中必须根据下面的公式设置分数分频器的值（寄存器 FDV）：

$$FDV = \frac{512 \times f_{DIV}}{f_{ASC}}, \quad f_{DIV} = 9.6 \text{ MHz} \quad (19.3)$$

利用该选择（ $f_{DIV} = 9.6 \text{ MHz}$ ），可检测的波特率从 200 kbit/s（Br0）直到 1042 bit/s（Br8）。表 19-10 给出对应的波特率范围。

表 19-10 利用非标准波特率进行波特率自动检测 ($f_{DIV} = 9.6 \text{ MHz}$)

波特率编号	可检测到的非标准波特率	分频因子 d_f	检测后 BG 载入值
Br0	200,000 bit/s	48	2 = 002 _H
Br1	100,000 bit/s	96	5 = 005 _H
Br2	50,000 bit/s	192	11 = 00B _H
Br3	33,333 bit/s	288	17 = 011 _H
Br4	16,667 bit/s	576	35 = 023 _H
Br5	8,333 bit/s	1152	71 = 047 _H
Br6	4,167 bit/s	2304	143 = 08F _H
Br7	2,083 bit/s	4608	287 = 11F _H
Br8	1,042 bit/s	9216	575 = 23F _H

19.5.4 自动波特率检测成功后，改写寄存器

自动波特率检测成功后，寄存器 ASCx_CON 和 ASCx_BG 中的一些位被自动设置为和检测到的串行帧模式及波特率相对应的值（见表 19-11）。控制寄存器 ASCx_CON 中的模式控制位 M 和奇偶校验位 ODD 被改写。寄存器 ASCx_BG 中载入波特率定时器的 13 位重载值。

表 19-11 自动波特率检测后，CON 寄存器重写值

检测到的参数		M	ODD	BR_VALUE
工作模式	7 位，偶校验	011	0	-
	7 位，奇校验	011	1	
	8 位，偶校验	111	0	
	8 位，奇校验	111	1	
	8 位，无校验	001	0	
波特率	Br0	-	-	2 = 002 _H
	Br1			5 = 005 _H
	Br2			11 = 00B _H
	Br3			17 = 011 _H
	Br4			35 = 023 _H
	Br5			71 = 047 _H
	Br6			143 = 08F _H
	Br7			287 = 11F _H
	Br8			575 = 23F _H

注：自动波特率检测中断的描述见[章节 19.7](#)。

19.6 硬件错误检测功能

为了提高串行数据交换的安全性，串行通道 **ASC** 提供了一个错误中断请求标志用来指示是否出现错误，寄存器 **ASCx_CON** 中的 3 个（可选的）错误状态标志指示接收期间检测到的错误类型。接收完成时，如果至少满足下列条件之一，错误中断请求线 **EIR** 与接收中断请求线 **RIR** 将同时被激活。

- 如果置位帧错误检测使能位 **FEN** 且任何期望的停止位不为高，那么置位帧错误标志 **FE**，指示因帧错误产生一个错误中断请求（仅适用于异步模式）。
- 如果置位奇偶错误检测使能位 **PEN**，且对接收到的数据位进行奇偶校验后表明接收数据出错，则置位奇偶校验错误标志 **PE**，指示因奇偶校验出错产生一个错误中断请求（仅适用于异步模式）。
- 如果置位过载错误检测使能位 **OEN**，如果新的一帧接收已完成，接收缓存中接收到的前一个字符还未由软件或者 **DMA** 传送读出，置位过载错误标志 **OE**，指示因过载错误产生错误中断请求（适用于异步和同步模式）。

19.7 中断

串行通道 ASC 共有 6 个中断源。TIR 指示发送中断，TBIR 指示发送缓存中断，RIR 指示接收中断，EIR 指示串行通道错误中断。自动波特率检测单元提供了另外两个中断，ABSTIR 自动波特率检测开始中断和 ABDETIR 自动波特率检测中断。中断输出线 TBIR, TIR, RIR, EIR, ABSTIR, ABDETIR 被激活时（有效状态），需要保持 2 个模块时钟 f_{ASC} 周期。

产生错误中断请求的原因（帧错误、奇偶校验错误、过载错误）可由错误状态标志 FE、PE 和 OE 给出。寄存器 ABSTAT 为两个自动波特率检测中断提供状态信息。

注：与错误中断请求线 EIR 相反，错误状态标志 FE/PE/OE 不会自动复位，必须由软件清零。

对于正常操作（即不出现错误中断），ASC 提供 3 个中断请求控制该串行通道的数据交换：

- 将数据从 TBUF 移入发送移位寄存器时，激活 TBIR。
- 在异步帧的最后一位发送之前，或者同步帧的最后一位发送之后，激活 TIR。
- 将接收到的帧移入 RBUF 时，激活 RIR。

注：由一个中断处理器服务接收任务，由另两个中断处理器服务发送器，这为服务程序提供了便利。

如果只是发送数据，使用发送中断（TIR）已经足够，指示除了异步帧的最后一位，先前载入的数据已经被发送。对于多次连续传送，直到先前帧的最后一位被发送才载入下面的帧，这点是非常必要的。异步模式下，这正好为处理器留下一个位时间用来响应发送中断请求，在同步模式下，这是不可能的。

当先前的数据仍在发送时，可以重载 TBUF，从而利用发送缓存中断（TBIR）重载发送数据，为服务子程序发送完整帧提供时间。

只要自动波特率检测单元使能（ABEN, ABDETEN 和 ABSTEN 置位），并在 RxD 上检测到一个起始位，就会产生自动波特率操作开始中断 ABSTIR。这种情况下，自动波特率检测期间只要检测到一个起始位，都会产生 ABSTIR。

在识别到 2 字节帧的第二个字符之后，总会产生自动波特率检测中断 ABDETIR。这意味一次成功的自动波特率检测。如果 FCDTEN 被置位，在识别到 2 字节帧的第一个字符之后，也会产生自动波特率检测中断 ABDETIR。

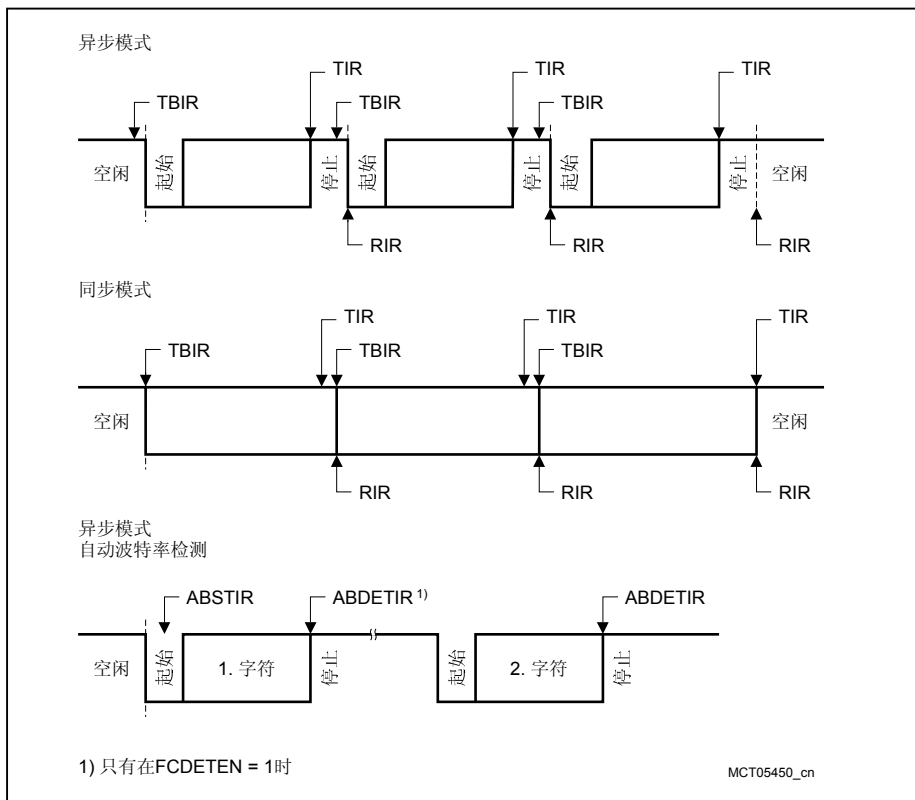


图 19-19 ASC 中断产生

如图 19-19 所示，TBIR 用于重载子程序的早期触发，而 TIR 指示发送完成。因此，在数据块结束时，进行信息发送的程序应当根据 TIR 信号以确保所有的数据都已发送。

由以下中断服务请求控制寄存器控制 ASC0 和 ASC1 模块的六个中断：

- ASC0_ABIC, ASC1_ABIC: 控制自动波特率检测中断
- ASC0_TIC, ASC1_TIC: 控制发送中断
- ASC0_RIC, ASC1_RIC: 控制接收中断
- ASC0_EIC, ASC1_EIC: 控制错误中断
- ASC0_TBIC, ASC1_TBIC: 控制发送缓存为空中断

注：控制位域的解释请参看通用中断控制寄存器的描述。

每个 ASC 模块中的两个自动波特率检测中断请求线（自动波特率检测开始和自动波特率检测结束）“相或”；其输出信号和中断控制寄存器连接，见图 19-20。

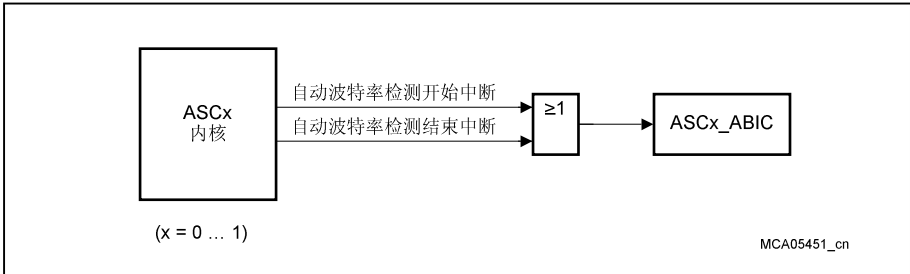


图 19-20 自动波特率检测中断连线

表 19-12 总结了所有的中断请求源。

表 19-12 ASC 中断源

中断	信号	描述
TBUF 操作	TBIR	将发送缓存寄存器 ASCx_TBUF 的内容写入发送移位寄存器的操作。 如果 ASC 配置为 FIFO 且位 TXTMEN 清零，由 TXFITL 定义的 FIFO 填充级别决定何时产生中断。
发送中断	TIR	数据帧的最后一位（第 8 位）通过线 TxD 从发送移位寄存器送出之后，产生中断。 <i>注：仅适用于同步模式</i>
发送中断	TIR	发送移位寄存器通过线 TxD 送出数据帧的最后一位之前，产生中断。如果 ASC 模块配置为 FIFO 且 TXTMEN 清零，由 TXFITL 定义的 FIFO 填充级别决定何时产生中断。 <i>注：仅适用于异步模式</i>
接收中断	RIR	当接收到的帧被从接收移位寄存器复制到接收缓存寄存器中，产生中断。 <i>注：仅适用于同步模式</i>
接收中断	RIR	当接收到的帧被从接收移位寄存器复制到接收缓存寄存

中断	信号	描述
		<p>器中，产生中断。如果 ASC 模块配置为 FIFO 且位 RXTMEN 清零，RXFITL 定义的 FIFO 填充级别决定何时产生中断。</p> <p><i>注：仅适用于异步模式</i></p>
接收出错中断	RIR 和 EIR	<p>当接收到的帧从接收移位寄存器复制到接收缓存寄存器且接收缓存中已经包含有效数据时，产生中断。</p> <p><i>注：仅适用于同步模式</i></p>
接收溢出	RIR 和 EIR	<p>当 FIFO 已填满，如果再接收到数据帧，会出现溢出错误，产生这两个中断，FIFO 中先前接收的帧被覆盖，因而丢失。</p>
读取空 FIFO	EIR	<p>CPU 对空的接收 FIFO 进行读取操作时，产生该中断</p>
透明读操作	RIR	<p>在透明模式下，CPU 读取接收 FIFO，如果 FIFO 不为空，该读操作之后总会产生接收中断。</p>
擦除操作	TBIR	<p>当发送 FIFO 被擦除时，产生一个发送缓存中断。</p>
FIFO 使能	TBIR	<p>先前透明模式下被禁止的发送 FIFO 通过置位 TXTMEN 和 TXFEN 被再次使能时，产生发送缓存中断。</p>
发送溢出	EIR	<p>当发送 FIFO 已填满，如果再写入数据帧，会出现溢出错误，产生中断。FIFO 中先前写入的帧被覆盖，因而丢失。</p>
帧出错	RIR 和 EIR	<p>期望的停止位不为高。</p> <p><i>注：仅适用于异步模式</i></p>
校验出错	RIR 和 EIR	<p>接收到的奇偶校验位和接收数据的奇偶校验不符。</p> <p><i>注：仅适用于异步模式</i></p>

19.8 寄存器

表 19-13 给出用于设置 ASC 模块的所有寄存器，其中包括 ASC 内核寄存器、中断控制寄存器及其地址。

表 19-13 ASC 模块寄存器总结

寄存器名	说明	ASC0 地址		寄存器区	ASC1 地址	
		16 位	8 位		16 位	8 位
ASCx_CON	控制寄存器	FFB0 _H	D8 _H	SFR	FFB8 _H	DC _H
ASCx_TBUF	发送缓存寄存器	FEB0 _H	58 _H	SFR	FEB8 _H	5C _H
ASCx_RBUF	接收缓存寄存器	FEB2 _H	59 _H	SFR	FEBA _H	5D _H
ASCx_ABCON	自动波特率检测控制寄存器	F1B8 _H	DC _H	ESFR	F1BC _H	DE _H
ASCx_ABSTAT	自动波特率检测状态寄存器	F0B8 _H	5C _H	ESFR	F0BC _H	5E _H
ASCx_BG	波特率定时器重载寄存器	FEB4 _H	5A _H	SFR	FEBC _H	5E _H
ASCx_FDV	分数分频寄存器	FEB6 _H	5B _H	SFR	FEBE _H	5F _H
ASCx_PMW	IrDA 脉冲模式和宽度寄存器	FEAA _H	55 _H	SFR	FEAC _H	56 _H
ASCx_RXFCON	接收 FIFO 控制寄存器	F0C6 _H	63 _H	ESFR	F0A6 _H	53 _H
ASCx_TXFCON	发送 FIFO 控制寄存器	F0C4 _H	62 _H	ESFR	F0A4 _H	52 _H
ASCx_FSTAT	FIFO 状态寄存器	F0BA _H	5D _H	ESFR	F0BE _H	5F _H
ASCx_ABIC	自动波特率检测中断控制寄存器	F15C _H	AE _H	ESFR	F1BA _H	DD _H
ASCx_TIC	发送中断控制寄存器	FF6C _H	B6 _H	SFR/ ESFR	F182 _H	C1 _H
ASCx_RIC	接收中断控制寄存器	FF6E _H	B7 _H	SFR/ ESFR	F18A _H	C5 _H

异步/同步串行接口（ASC）

寄存器名	说明	ASC0 地址		寄存器区	ASC1 地址	
		16 位	8 位		16 位	8 位
ASCx_EIC	出错中断控制寄存器	FF70 _H	B8 _H	SFR/ ESFR	F192 _H	C9 _H
ASCx_TBIC	发送缓存中断控制寄存器	F19C _H	CE _H	ESFR	F150 _H	A8 _H

控制寄存器

串行通道 **ASC** 的工作模式由其控制寄存器 **CON** 控制。该寄存器包含用于选择模式和错误检查类型的控制位，以及用于错误识别的状态标志。

ASCx_CON

控制寄存器

SFR（见表 19-13）

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R	LB	BRS	ODD	FDE	OE	FE	PE	OEN	FEN	PEN/RxDI	REN	STP	M		
rw	rw	rw	rw	rw	rwh	rwh	rwh	rw	rw	rw	rwh	rw	rw		

符号	位序号	读写类型	功能描述
R	15	rw	波特率发生器运行控制位 0 禁止波特率发生器工作（ASC 无效） 1 使能波特率发生器工作 <i>注：仅在 R = 0 时，才可对 BR_VALUE 进行写操作。</i>
LB	14	rw	回环模式使能 0 禁止回环模式。标准发送/接收模式 1 使能回环模式
BRS	13	rw	波特率选择 0 选择波特率定时器 2 分频预分频器 1 选择波特率定时器 3 分频预分频器 <i>注：如果 FDE = 1（选择分数分频器），BRS 值无效。</i>
ODD	12	rw	奇偶校验选择 0 选择偶校验（数据流中‘1’个数为奇数，奇偶校验位为 1；数据流中‘1’的个数为偶数，奇偶校验位为 0）。 1 选择奇校验（数据流中‘1’个数为偶数，奇偶校验位为 1；数据流中‘1’的个数为奇数，奇偶校验位为 0）。
FDE	11	rw	分数分频器使能

符号	位序号	读写类型	功能描述
			0 禁止分数分频器 1 使能分数分频器，用作波特率发生器的预分频器（位 BRS 无效）
OE	10	rwh	过载错误标志 出现过载/下溢错误（OEN = 1）时，由硬件置位，必须由软件清零。
FE	9	rwh	帧错误标志 出现帧错误（FEN = 1）时，由硬件置位，必须由软件清零。
PE	8	rwh	奇偶校验错误标志 出现奇偶校验错误（PEN = 1）时，由硬件置位，必须由软件清零。
OEN	7	rw	过载检查使能 0 忽略过载错误 1 检查过载错误
FEN	6	rw	帧检查使能 （仅适用于异步模式） 0 忽略帧错误 1 检查帧错误
PEN/RxDI	5	rw	奇偶检查使能/IrDA 模式下 RxDI 反相 所有非 IrDA 模式的异步模式下（PEN） 0 忽略奇偶校验 1 检查奇偶校验 仅在 IrDA 模式下（RxDI） 0 RxD 输入不反相 1 RxD 输入反相
REN	4	rwh	接收器使能位 0 禁止接收器 1 使能接收器 <i>注：同步模式下，接收一个字节之后由硬件清零位 REN。</i>

符号	位序号	读写类型	功能描述
STP	3	rw	停止位个数选择 0 1 个停止位 1 2 个停止位
M	[2:0]	rw	模式控制 000 _B 8 位数据同步操作 001 _B 8 位数据异步操作 010 _B 8 位数据 IrDA 模式异步操作 011 _B 7 位数据加奇偶校验位的异步操作 100 _B 9 位数据异步操作 101 _B 8 位数据加唤醒位的异步操作 110 _B 保留，请勿使用该组合 111 _B 8 位数据加奇偶校验位的异步操作

波特率寄存器

ASC 波特率定时器重载寄存器 BG 中包含异步或同步模式下波特率定时器的 13 位重载值。

ASCx_BG

波特率定时器/重载寄存器 SFR（见表 19-13） 复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-			BR_VALUE												
-			rw												

符号	位序号	读写类型	功能描述
BR_VALUE	[12:0]	rw	波特率定时器/重载值 对其进行读操作时，返回波特率定时器的 13 位计数值，进行写操作时，加载波特率定时器的重载值。 <i>注：仅当 R= 0 时，才能对 BG 进行写操作。</i>

分数分频器寄存器

ASC 分数分频器寄存器 FDV 包含 9 位分频值（仅适用于异步模式）。还可为自动波特率检测单元提供参考时钟。

ASCx_FDV

分数分频器寄存器						SFR（见表 19-13）						复位值: 0000 _H					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
-						FD_VALUE											
-						rw											

符号	位序号	读写类型	功能描述
FD_VALUE	[8:0]	rw	分数分频器寄存器的值 FD_VALUE 包含 9 位分数分频值，定义分数分频因子 $n/512$ ($n = 0 \dots 511$)。 $n = 0$ 时，分数分频器关闭（输入频率 = 输出频率， $f_{DIV} = f_{ASC}$ ，见 图 19-14）。

IrDA 脉冲模式/宽度寄存器

ASC IrDA 脉冲模式和宽度寄存器 PMW 包含 8 位 IrDA 脉冲宽度值和 IrDA 脉冲宽度模式选择位。仅在 IrDA 工作模式下才用到该寄存器。

ASCx_PMW

IrDA 脉冲模式/宽度寄存器

SFR（见表 19-13）

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-								IRP W	PW_VALUE						
-								rw	rw						

符号	位序号	读写类型	功能描述
IRPW	8	rw	IrDA 脉冲宽度选择 0 IrDA 脉冲宽度为 3/16 位时间 1 IrDA 脉冲宽度由 PW_VALUE 定义
PW_VALUE	[7:0]	rw	IrDA 脉冲宽度值 PW_VALUE 为 8 位值 n，定义 IrDA 脉冲的可变脉冲宽度。 根据 f_{ASC} 的值，可用该值调整不等于 3/16 位时间的 IrDA 脉冲宽度（例如，1.6ms）。

ASC 模块发送缓存寄存器 TBUF 包含异步或同步模式下的发送数据值。

发送缓存寄存器

SFR (见表 19-13)

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-							TD_VALUE								
-							rw								

符号	位序号	读写类型	功能描述
TD_VALUE	[8:0]	rw	<p>发送数据寄存器的值</p> <p>TBUF 包含 ASC 在异步和同步工作模式下要发送的数据。数据发送双缓存。因此，在先前值发送完成之前，可将新值写入 TBUF。</p>

自动波特率检测控制寄存器

ASC 模块的自动波特率检测控制寄存器 **ABCON** 用于控制自动波特率检测的操作，其中包含通用使能位、中断使能控制位和数据通道控制位。

ASCx_ABCON

自动波特率控制寄存器

ESFR（见表 19-13）

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-				RX INV	TX INV	ABEM		-			FC DET EN	AB DET EN	ABS T EN	AUR EN	AB EN
-				rw	rw	rw		-			rw	rw	rw	rw	rwh

符号	位序号	读写类型	功能描述
RXINV	11	rw	接收反相器使能 0 禁止接收反相 1 使能接收反相
TXINV	10	rw	发送反相器使能 0 禁止发送反相 1 使能发送反相
ABEM	[9:8]	rw	自动波特率检测回声模式使能 回声模式下，RxD 上的串行数据被切换至 TxD 输出上。 00 禁止回声模式 01 在自动波特率检测期间使能回声模式 10 始终使能回声模式 11 保留；注意不要使用该组合
FCDETEN	4	rw	2 字节数据帧第一个字符检测使能 0 在识别到 2 字节帧之后，自动波特率检测中断 ABDETIR 变为有效。 1 在检测到 2 字节帧的第一 和 第二个字符之后，自动波特率检测中断 ABDETIR 变为有效。
ABDETEN	3	rw	自动波特率检测中断使能

符号	位序号	读写类型	功能描述
			0 禁止自动波特率检测中断 1 使能自动波特率检测中断
ABSTEN	2	rw	自动波特率检测开始中断使能 0 自动波特率检测开始中断被禁止 1 自动波特率检测开始中断被使能
AUREN	1	rw	CON.REN 的自动波特率检测控制 0 自动波特率检测期间 CON.REN 不受影响 1 当 ABEN 和 AUREN 一起置位时，清零 CON.FEN（禁止接收器） 在一次成功的波特率自动检测之后（检测到第二个字符的停止位），CON.REN 置位（使能接收器）。
ABEN	0	rwh	自动波特率检测使能 0 禁止自动波特率检测 1 使能自动波特率检测 <i>注：一次成功的自动波特率检测之后，由硬件复位 ABEN；（检测到第二个字符的停止位）。</i> <i>如果先前 ABEN 被置位，那么软件复位 ABEN 会中止自动波特率检测。</i>

自动波特率检测状态寄存器

ASC 模块自动波特率检测状态寄存器 ABSTAT 指示自动波特率检测操作的状态。

ASCx_ABSTAT

自动波特率状态寄存器

ESFR（见表 19-13）

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-											DET WAIT	SCC DET	SCS DET	FCC DET	FCS DET
-											rwh	rwh	rwh	rwh	rwh

符号	位序号	读写类型	功能描述
DETWAIT	4	rwh	自动波特率检测处于等待状态 0 字符 ‘a’，‘A’，‘t’ 或 ‘T’ 之一已被检测到 1 自动波特率检测单元等待第一个 ‘a’ 或 ‘A’ 当 FCSDET 和 FCCDET 其中任一个被置位，该位清零（检测到 ‘a’ 或 ‘A’）。该位也可软件清零。当 ABEN 置位时，由硬件置位 DETWAIT。
SCCDET	3	rwh	第二个字符检测到大写字母 0 没有检测到大写字符 ‘T’ 1 已经检测到大写字符 ‘T’ 当 ABEN 置位，如果 FCSDET 或 FCCDET 或 SCSDET 置位，该位由硬件清零。该位也可由软件清零。
SCSDET	2	rwh	第二个字符检测到小写字母 0 没有检测到小写字符 ‘t’ 1 已经检测到小写字符 ‘t’ 当 ABEN 置位，如果 FCSDET 或 FCCDET 或 SCCDET 置位，该位由硬件清零。该位也可由软件清零。
FCCDET	1	rwh	第一个字符检测到大写字母

符号	位序号	读写类型	功能描述
			0 没有检测到大写字符 ‘A’ 1 已经检测到大写字符 ‘A’ 当 ABEN 置位，如果 FCSDET 或 SCSDET 或 SCCDET 置位，该位由硬件清零。该位也可由软件清零。
FCSDET	0	rwh	第一个字符检测到小写字母 0 没有检测到小写字符 ‘a’ 1 已经检测到小写字符 ‘a’ 当 ABEN 置位，如果 FCCDET 或 SCSDET 或 SCCDET 置位，该位由硬件清零。该位也可由软件清零。

注：当识别到第二个字符时，置位 SCSDET 或 SCCDET。在 SCSDET 或 SCCDET 置位之后，复位 ABEN，置位 ABDETIR。

接收 FIFO 控制寄存器

ASCx_RXFCON

接收 FIFO 控制寄存器

ESFR（见表 19-13）

复位值: 0100H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-					RXFITL					-			RX TM EN	RXF FLU	RXF EN
-					rw					-			rw	rw	rw

符号	位序号	读写类型	功能描述
RXFITL	[11:8]	rw	接收 FIFO 中断触发级别 定义接收 FIFO 中断触发级别。当接收 FIFO 填充级别等于或大于 RXFITL 时，在接收一个字节之后，产生接收中断请求（RIR）。 0000 保留；不要使用该组合 0001 中断触发级别设置为 1 0010 中断触发级别设置为 2 0111 中断触发级别设置为 7 1000 中断触发级别设置为 8 <i>注：在透明模式下，不考虑该位域。</i> <i>注：不应当使用大于 FIFO 容量的中断触发级别。</i>
RXTMEN	2	rw	接收 FIFO 透明模式使能 0 禁止接收 FIFO 透明模式 1 使能接收 FIFO 透明模式 <i>注：如果禁止接收 FIFO（RXFEN = 0），该位无效。</i>
RXFFLU	1	rw	接收 FIFO 擦除 0 无操作 1 擦除接收 FIFO <i>注：置位 RXFFLU 清零寄存器</i>

符号	位序号	读写类型	功能描述
			<i>ASC_x_FSTAT 中的 RXFFL。读 RXFFLU 总是返回 0。</i>
RXFEN	0	rw	接收 FIFO 使能 0 禁止接收 FIFO 1 使能接收 FIFO <i>注：复位 RXFEN 自动擦除接收 FIFO。</i>

注：一次成功的自动波特率检测之后，在接收数据之前应该擦除 RXFIFO。

发送 FIFO 控制寄存器

ASCx_TXFCON

发送 FIFO 控制寄存器

ESFR（见表 19-13）

复位值: 0100_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-					TXFITL			-					TX TM EN	TXF FLU	TXF EN
-					rw			-					rw	rw	rw

符号	位序号	读写类型	功能描述
TXFITL	[11:8]	rw	<p>发送 FIFO 中断触发级别</p> <p>定义发送 FIFO 中断触发级别。当发送 FIFO 填充级别等于或低于 TXFITL 时，在传送一个字节之后，产生发送中断请求（TIR）。</p> <p>0000 保留；不要使用该组合</p> <p>0001 中断触发级别设置为 1</p> <p>0010 中断触发级别设置为 2</p> <p>...</p> <p>0111 中断触发级别设置为 7</p> <p>1000 中断触发级别设置为 8</p> <p><i>注：在透明模式下，不考虑该位域。</i></p> <p><i>注：不应当使用大于 FIFO 容量的中断触发级别。</i></p>
TXTMEN	2	rw	<p>发送 FIFO 透明模式使能</p> <p>0 禁止发送 FIFO 透明模式</p> <p>1 使能发送 FIFO 透明模式</p> <p><i>注：如果禁止发送 FIFO（TXFEN = 0），该位无效。</i></p>
TXFFLU	1	rw	<p>发送 FIFO 擦除</p> <p>0 无操作</p> <p>1 擦除发送 FIFO</p> <p><i>注：置位 TXFFLU 清零寄存器</i></p>

			<i>ASCx_FSTAT 中的 TXFFL。读 TXFFLU 总是返回 0。</i>
TXFEN	0	rw	发送 FIFO 使能 0 禁止发送 FIFO 1 使能发送 FIFO <i>注：复位 TXFEN 自动擦除发送 FIFO。</i>

FIFO 状态寄存器

ASCx_FSTAT

FIFO 状态寄存器

ESFR（见表 19-13）

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-				TXFFL				-				RXFFL			
-				rh				-				rh			

符号	位序号	读写类型	功能描述
TXFFL	[11:8]	rh	发送 FIFO 填充级别 0000 发送 FIFO 中填充了 0 字节 0001 发送 FIFO 中填充了 1 个字节 0010 发送 FIFO 中填充了 2 个字节 0111 发送 FIFO 中填充了 7 个字节 1000 发送 FIFO 中填充了 8 个字节 <i>注：在发送 FIFO 擦除操作之后，TXFFL 清零。</i>
RXFFL	[3:0]	rh	接收 FIFO 填充级别 0000 接收 FIFO 中填充了 0 字节 0001 接收 FIFO 中填充了 1 个字节 0010 接收 FIFO 中填充了 2 个字节 0111 接收 FIFO 中填充了 7 个字节 1000 接收 FIFO 中填充了 8 个字节 <i>注：接收 FIFO 擦除操作之后，RXFFL 清零。</i>

19.9 ASC 模块接口

XC164CM 的 ASC 模块和 IO 端口及其它内部模块的连接见图 19-21 和图 19-22。

ASC0 和 ASC1 的输入/输出线和 P3 口引脚相连。每个模块的 6 根中断请求线和中断控制模块相连。

由系统控制单元 SCU 处理 ASC 模块的时钟控制和仿真控制。

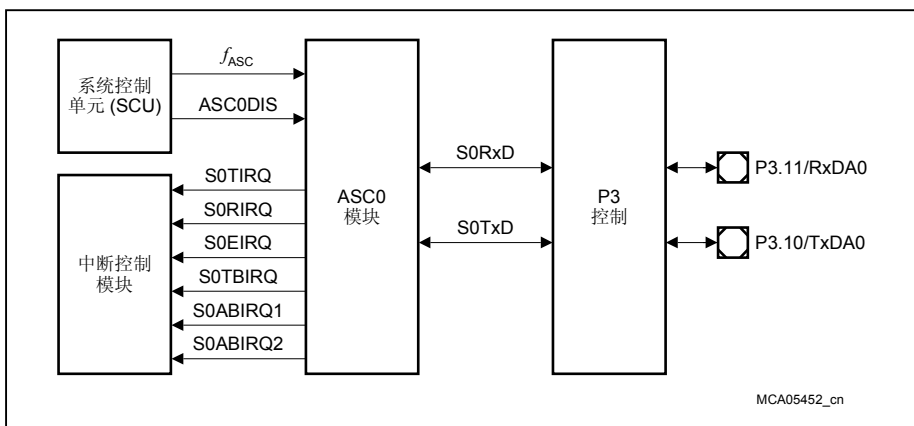


图 19-21 ASC0 模块接口

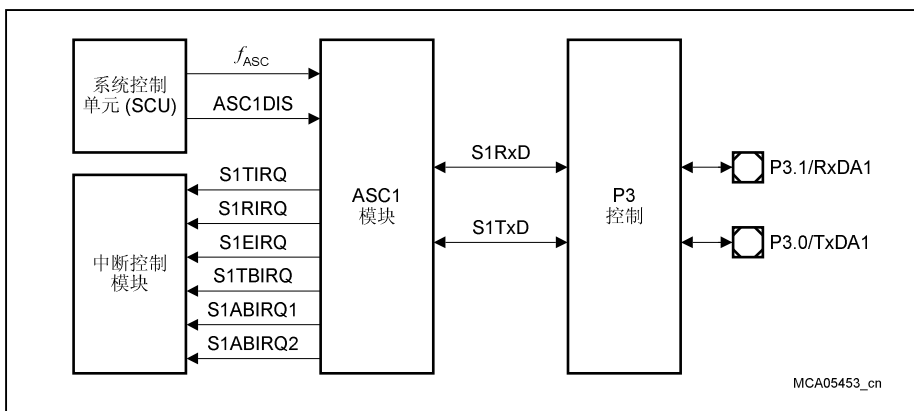


图 19-22 ASC1 模块接口

注：同步工作模式下，不是由 ASC 模块自动设置 RxD 引脚的方向；必须根据所选的工作模式（接收或发送数据），通过寄存器 DP3 中相应的位由软件进行方向切换。

注：P3.1 用作 RxD A1 时，只需置位寄存器 ALTSEL0P3 中位元 1 即可，不需要考虑寄存器 ALTSEL1P3 中的相应位。

20 高速同步串行接口

XC164CM 中有两个高速同步串行接口，SSC0 和 SSC1。以下章节将给出 SSC 模块的一般特性及操作。最后一节说明两个 SSC 模块的具体实现，包括如何和其它片上模块连接。

20.1 介绍

高速同步串行接口（SSC）支持高达 20 Mbit/s（模块工作时钟为 40 MHz）的全双工和半双工串行同步通信。串行时钟信号由 SSC 模块自身产生（主模式），或从外部主机接收得到（从模式）。数据宽度、移位方向、时钟极性和相位均可编程，从而可与 SPI 兼容器件通信。数据发送和接收双缓存。16 位波特率发生器为 SSC 提供独立的串行时钟信号。

特性和功能

- 主模式和从模式操作
 - 全双工或半双工工作
- 灵活的数据格式
 - 数据位个数可编程：2 至 16 位
 - 移位方向可编程：先移位最低有效位（LSB）或最高有效位（MSB）
 - 时钟极性可编程：移位时钟低电平空闲或高电平空闲
 - 时钟/数据相位可编程：在移位时钟的前沿或后沿进行数据移位
- 波特率可变：20 Mbit/s 到 306.6 bit/s（模块工作时钟为 40 MHz）
- 中断产生
 - 发送缓存寄存器已空
 - 接收缓存寄存器已满
 - 出错情况（接收、相位、波特率、发送错误）

20.2 工作概述

用户可灵活的配置高速同步串行接口：用于和其它同步串行接口通信；用于主机/从机或多主机通信；或者与流行的 SPI 接口兼容工作。因此，SSC 可和移位寄存器（IO 扩展）、外设（如 EEPROM 等）或其它控制器（网络）进行通信。SSC 支持全双工和半双工串行通信。通过 MTX/STX 发送数据，通过 MRX/SRX 接收数据，发送和接收线分别和引脚 MTSR（主机发送/从机接收）和 MRST（主机接收/从机发送）相连。时钟信号由 MSCLK（主机串行移位时钟）输出或从 SSCLK（从机串行移位时钟）输入；这两条时钟线和引脚 SCLK 相连。这些引脚为复用端口功能。

SSC 模块的框图如图 20-2 所示。

高速同步串行接口（SSC）

从编程人员的角度来看，SSC 单元由一组与之相关的寄存器（SFR）组成（见图 20-1），其中包括用作 SSC 功能的端口寄存器和端口方向控制寄存器。

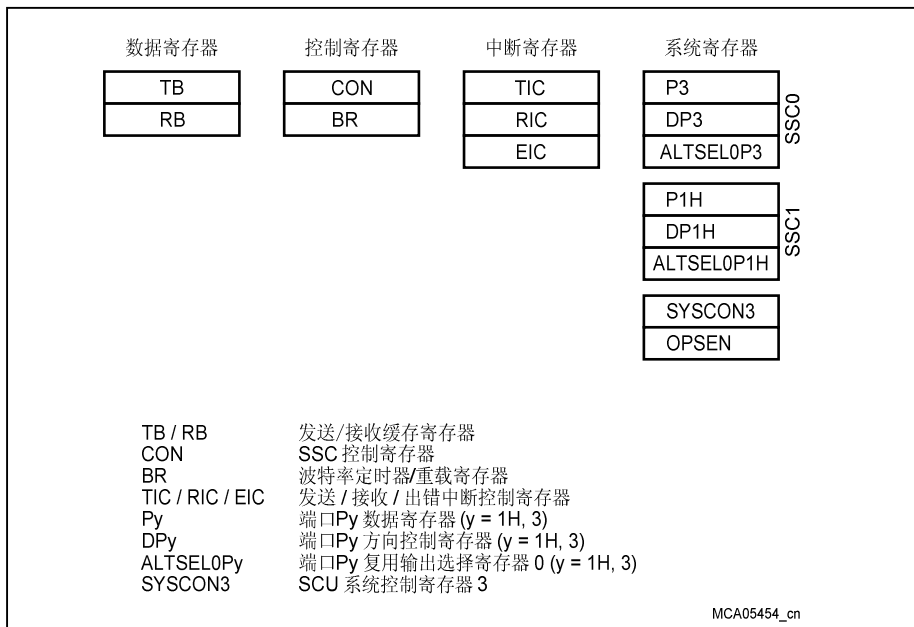


图 20-1 SSC 单元的相关特殊功能寄存器（SFR）

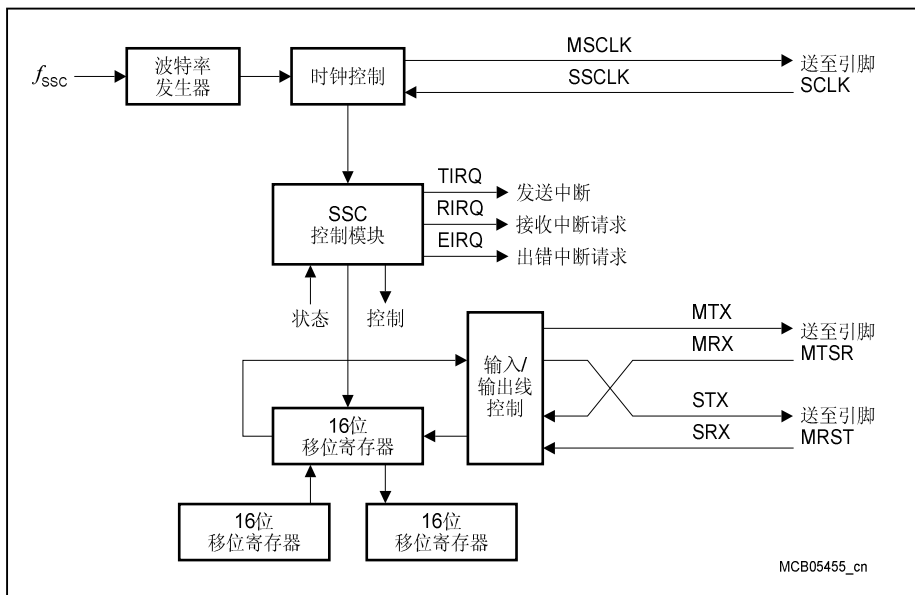


图 20-2 同步串行通道（SSC）框图

20.2.1 工作模式选择

SSC 模块的工作模式由控制寄存器 `SSCx_CON` 选择。该寄存器具有双重功能：

- 编程模式下（通过 `SSCx_CON.EN=0` 禁止 SSC），可访问控制位
- 工作模式下（通过 `SSCx_CON.EN=1` 使能 SSC），可访问状态标志位

下面给出两种功能下寄存器 `SSCx_CON` 的结构。

SSC 控制寄存器（SSCx_CON.EN = 0：编程模式）

SSCx_CON

SSC 控制寄存器

SFR（见表 20-2）

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EN = 0	MS	-	A REN	BEN	PEN	REN	TEN	LB	PO	PH	HB	BM			
rw	rw	-	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			

符号	位序号	读写类型	功能描述
EN	15	rw	使能位 = 0 发送和接收被禁止。可访问控制位。
MS	14	rw	主机选择 0 从模式。从 SCLK 接收移位时钟 1 主模式。产生移位时钟并从 SCLK 输出
AREN	12	rw	自动复位使能 0 波特率出错时无操作 1 波特率出错时 SSC 被自动复位
BEN	11	rw	波特率出错使能 0 忽略波特率错误 1 检查波特率错误
PEN	10	rw	相位出错使能 0 忽略相位错误 1 检查相位错误
REN	9	rw	接收错误使能 0 忽略接收错误 1 检查接收错误
TEN	8	rw	发送错误使能 0 忽略发送错误 1 检查发送错误
LB	7	rw	回环控制位

符号	位序号	读写类型	功能描述
			0 正常输出 1 接收输入和发送输出相连（半双工模式）
PO	6	rw	时钟极性控制 0 时钟线低电平空闲，时钟前沿为低到高的跳变 1 时钟线高电平空闲，时钟前沿为高到低的跳变
PH	5	rw	时钟相位控制 0 在时钟前沿移出发送数据，后沿锁存接收数据 1 在时钟前沿锁存接收数据，后沿移出发送数据
HB	4	rw	头信息控制 0 先发送/接收 LSB 1 先发送/接收 MSB
BM	[3:0]	rw	数据宽度选择 0000 保留。不使用该组合 0001 传送数据宽度为 2 bit 传送数据宽度为 (<BM> + 1) bit 1111 传送数据宽度为 16 bit

SSC 控制寄存器（SSCx_CON.EN = 1：工作模式）

SSCx_CON

SSC 控制寄存器

SFR（见[表 20-2](#)）

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EN = 1	MS	-	BSY	BE	PE	RE	TE	-	-	-	-	BC			
rw	rw	-	rh	rwh	rwh	rwh	rwh	-	-	-	-	rw			

符号	位序号	读写类型	功能描述
EN	15	rw	使能位 = 1 发送和接收被使能。可访问状态标志位和主机/从机控制位。
MS	14	rw	主机/从机选择 0 从模式。从 SCLK 接收移位时钟 1 主模式。产生移位时钟并从 SCLK 输出
BSY	12	rh	忙碌标志位 数据传送过程中被置位。 该位不可写入！
BE	11	rwh	波特率出错标志位 0 未出错 1 从机的实际波特率超过期望波特率的 2 倍或低于 0.5 倍
PE	10	rwh	相位出错标志位 0 未出错 1 接收数据在采样时钟沿附近发生改变
RE	9	rwh	接收出错标志位 0 未出错 1 接收缓存寄存器的内容还未读出，新数据接收已完成
TE	8	rwh	发送出错标志位 0 未出错 1 从机发送缓存寄存器还未更新，数据传送已开始
BC	[3:0]	rh	位计数域 每次移位时更新移位计数器内容。 该位不可写入！

注：访问 **SSCx_CON** 之前，由位 **EN** 的状态决定该寄存器的实际访问对象（访问控制位或标志位）。换句话说，在编程模式下（**EN** = 0）对 **SSCx_CON** 写入 **C057_H** 将启动 **SSC**（**EN** 原先为 0）、使其转入正常工作模式（**EN** = 1）。对 **SSCx_CON** 写操作时，请确保向保留位写 0。

发送缓存寄存器

SSC 发送缓存寄存器 SSCx_TB 中（见**表 20-2**）存放着发送数据。SSCx_TB 中未选中的数据位在发送过程中被忽略。无论先发送 MSB 还是 LSB，发送数据必须右对齐。

接收缓存寄存器

SSC 接收缓存寄存器 SSCx_RB 中（见**表 20-2**）存放着接收数据。SSCx_RB 中未选中的数据位无效、在接收过程中被忽略。无论先接收 MSB 还是 LSB，接收数据必须始终右对齐。

SSC 的移位寄存器通过引脚控制逻辑与发送线和接收线相连（见**图 20-2**）。串行数据的发送和接收同步，它们同时发生，即接收到的数据和发送的数据位数相同。

准备传送数据，将发送数据写入发送缓存寄存器（SSCx_TB），一旦移位寄存器为空立刻把 SSCx_TB 的内容移入。SSC 的主机（CON.MS=1）立刻开始发送；而 SSC 的从机（CON.MS=0）在等到有效的移位时钟之后才开始发送。当开始数据传送，忙碌标志 CON.BSY 被置位，发送中断请求线 TIRQ 被激活以指示寄存器 SSCx_TB 可再次被重载。在传送设定位数的数据（2...16）之后，移位寄存器的内容被移入接收缓存寄存器 SSCx_RB 中，接收中断请求线 RIRQ 被激活。如果不再进行进一步的数据传送（SSCx_TB 已空），CON.BSY 被同时清零。由于标志位 CON.BSY 由硬件控制，故不应由软件修改。

注：在一个串行系统中，给定时间内只有一个 SSC 可用作主机。

可从多方面设定串行数据位的传送：

- 可设定的数据宽度范围：2 位到 16 位
- 可先传送 LSB 或先传送 MSB
- 移位时钟可为低电平空闲或高电平空闲
- 可在移位时钟信号的前沿或后沿移位数据位
- 可设定的波特率范围：306.6 bit/s 到 20 Mbit/s（模块工作时钟为 40 MHz）
- 可产生移位时钟（MSCLK）或接收移位时钟（SSCLK）

这些特性使 SSC 可广泛适用于需要串行数据传送的应用领域。

数据宽度选择支持从 2 位“字符”到 16 位“字符”之间任意长度的数据帧。先传送 LSB（CON.HB=0），可使 SSC 和同步模式下的 SSC 器件通信，或者和 8051 式的串行接口通信；先传送 MSB（CON.HB=1），可使 SSC 和 SPI 接口兼容。

寄存器 SSCx_TB 和 SSCx_RB 中的传送数据始终右对齐，传送数据的 LSB 始终位于寄存器的位元 0，和选定的数据宽度以及先传送 LSB 还是 MSB 均无关。内部移位寄存器逻辑会对传送数据进行重排。SSCx_TB 中未选中的位不予理睬；SSCx_RB 中未选中的位无效，接收中断服务程序对其不予处理。

时钟控制逻辑使 **SSC** 的发送和接收适用于多种不同的串行接口。用一个特定的移位时钟沿（上升或下降沿）移出发送数据；同时用另一个时钟沿锁存接收数据。位 **PH** 选择由前沿或后沿移出数据；位 **PO** 选择移位时钟空闲状态的电平。因此，对于高电平空闲的时钟，时钟的前沿是下降沿，1 到 0 的跳变（见图 20-3）。

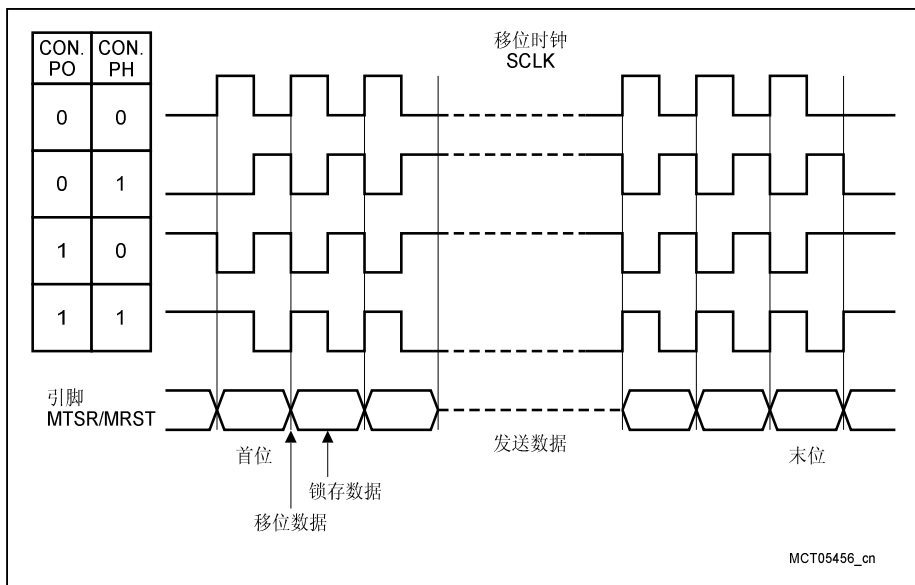


图 20-3 串行时钟相位和极性选择

20.2.2 全双工操作

全双工串口的设置如图 20-4 所示。通过三条线将不同的器件相互连接。始终由主机来定义这三条线：和主机的数据输出线 **MTSR** 相连的为发送线；和主机的数据输入线 **MRST** 相连的为接收线；移位时钟线为 **SCLK**。只有被选作主机的器件产生并从 **SCLK** 线上输出移位时钟。由于所有的从机接收该时钟，它们的 **SCLK** 引脚必须切换到输入模式。主机的移位寄存器输出和外部发送线相连；同时该外部发送线和从机的移位寄存器输入相连。从机移位寄存器的输出和外部接收线相连，从而可使主机接收从机移出的数据。外部连接是硬件连线，这些引脚的功能和方向由各器件的主机操作或从机操作决定。

注：图 20-4 所示的移位方向对 **MSB 在先** 和 **LSB 在先** 的操作均适用。

在该设置下初始化器件，必须设定一个器件工作在主模式；所有其它器件工作在主模式。初始化包括器件 **SSC** 的工作模式选择以及对应端口线的功能定义。

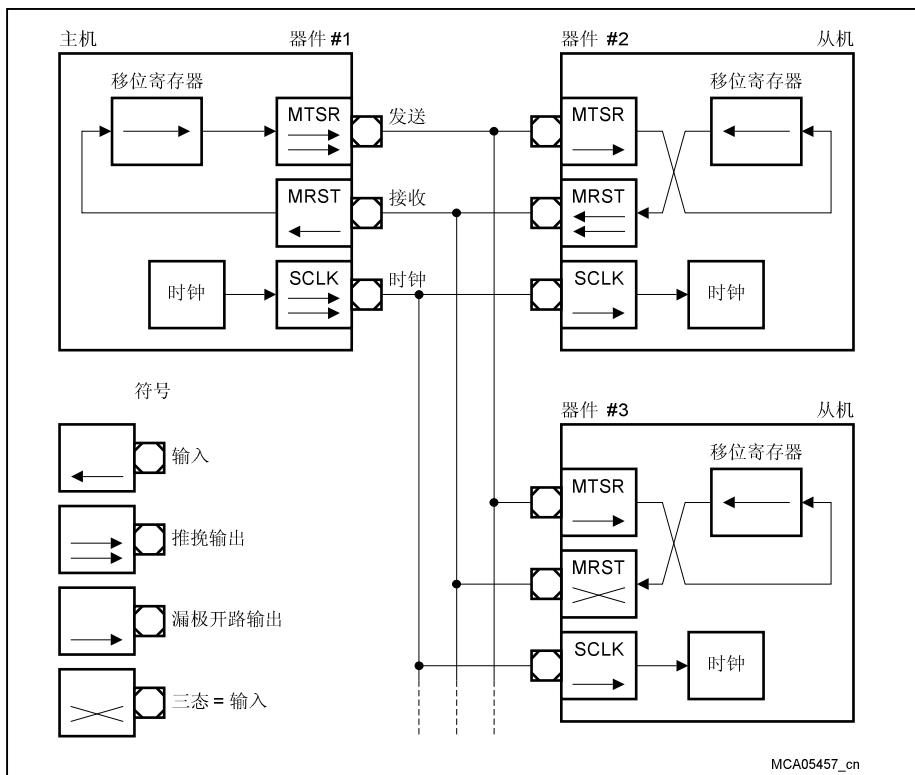


图 20-4 SSC 全双工设置

在如图 20-4 所示的设置中，所有从器件的数据输出引脚 **MRST** 共同连接到接收线上。数据传送时，每个从机从其移位寄存器中移出数据。有以下两种方式可避免不同的从机同时发送数据时，造成接收线上的传送冲突：

- 只有一个从机驱动接收线，即只驱动该从机的 **MRST** 引脚。必须将所有其它从机的 **MRST** 引脚设置为输入，从而只有一个从机可将数据送到主机的接收线上。这些其它的从机只可能接收来自主机的数据。主机通过两种方式选择期望获取数据的从机器件：通过独立的选择线、或向该从机发送特殊指令。选中的从机的 **MRST** 引脚切换为输出，并始终保持输出状态直到接收到取消信号或指令。

在如图 20-4 所示的设置中，器件#2 是可发送数据的从机，其输出驱动器被使能为推挽输出；器件#3 是被暂停发送数据的从机，其引脚被设定为输入模式以关闭其输出驱动器。

- 从机的 **MRST** 引脚采用漏极开路输出，形成线与连接。这种情况下接收线需要一个外部上拉器件。所有未被选中向主机发送数据的从机只发送“1”，从而避免了（选中发送数据的从机发送到）接收线上的数据遭破坏。由于高电平不能被有效的驱动到接收线上，而只能由上拉器件保存；选中发送数据的从机在发送“0”时可有效的将接收线拉至低电平。主机有两种方式选择期望获取数据的从机器件：通过独立的选择线、或向该从机发送特殊指令。

SSC 初始化之后，串行接口被使能。对于主机器件，时钟线 **MSCLK** 此时将对对应被设定的时钟极性。第一次传送开始之前，输出数据线 **MTX** 为 0 或 1。每次传送结束后，数据线 **MTX** 将始终保持最后发送的一位数据的逻辑电平。

若串行接口被使能，发送数据写入 **SSCx_TB** 后，主机器件可启动首次数据传送。该发送数据被复制到移位寄存器中（假定此时移位寄存器已空），在下一个时钟（该时钟由波特率发生器产生），发送数据中的首位被送到输出线 **MTSR** 上（只有当 **EN=1** 时才能开始发送）。根据选择的时钟相位，在 **SCLK** 线上产生时钟脉冲。同时，在相反的时钟沿，主机将输入线 **MRST** 上检测到的数据位锁存并移入移位寄存器。发送数据和接收数据从而进行了“交换”。因为时钟线和所有的从机相连，从机的移位寄存器将和主机的移位寄存器同步移位：移出寄存器中的数据，移入输入线上检测到的数据。在产生设定数目的时钟脉冲之后（由数据宽度决定），主机发送的数据被保存到所有从机的移位寄存器中；选中的从机发送的数据被保存到主机的移位寄存器中。主机和所有从机中，移位寄存器的内容被复制到接收缓存寄存器 **SSCx_RB** 中，接收中断线 **RIRQ** 被激活。

一旦发送缓存寄存器中的内容被复制到从机的移位寄存器中，从机器件会立刻将选中的首位数据（**MSB** 或 **LSB**）送到输出线 **MRST** 上。直到 **SCLK** 上出现第一个时钟沿，位 **BSY** 才被置位。从机器件和主机器件不同，无需等待由波特率发生器产生下一个时钟脉冲。原因在于：根据所选择的时钟相位，主机产生的第一个时钟沿有可能用于锁存第一位数据，因此，从机的第一位数据此时必须已经有效。

注：无论有效数据是否已被发送或接收，SSC 的发送和接收总是同时发生。

注：在初始化主机的 CLK 引脚时需多加注意，以免产生不希望的时钟跳变干扰其它器件。在通过相关的端口方向控制寄存器将时钟引脚切换到输出之前，应由控制寄存器 SSCx_CON 选择时钟的输出电平；由相关的 ALTSEL 寄存器选择复用输出功能；或用时钟空闲电平加载输出锁存器。

20.2.3 半双工操作

半双工模式下只要一条数据线，负责接收和发送数据。该数据交换线同时和每个器件的 MTSR 和 MRST 引脚相连，移位时钟线和 SCLK 引脚连接。

主机器件产生移位时钟以控制数据的传送，同时从机器件接收主机发送的数据。由于所有的发送和接收引脚都连接到同一条数据交换线上，可在任意两个器件之间传送串行数据。

和全双工模式相似，有以下两种方式可避免数据交换线上发生数据传送冲突：

- 只有发送器件可开启其发送引脚驱动器
- 不发送数据的器件采用漏极开路输出，并只发送“1”

由于数据的输入和输出线相连，发送器件会从输入引脚（主机器件对应 MRST，从机对应 MTSR）读回自身发送的数据。若接收数据和发送数据不一致，通过这种方法可以检测到公共数据交换线上的数据遭到破坏。

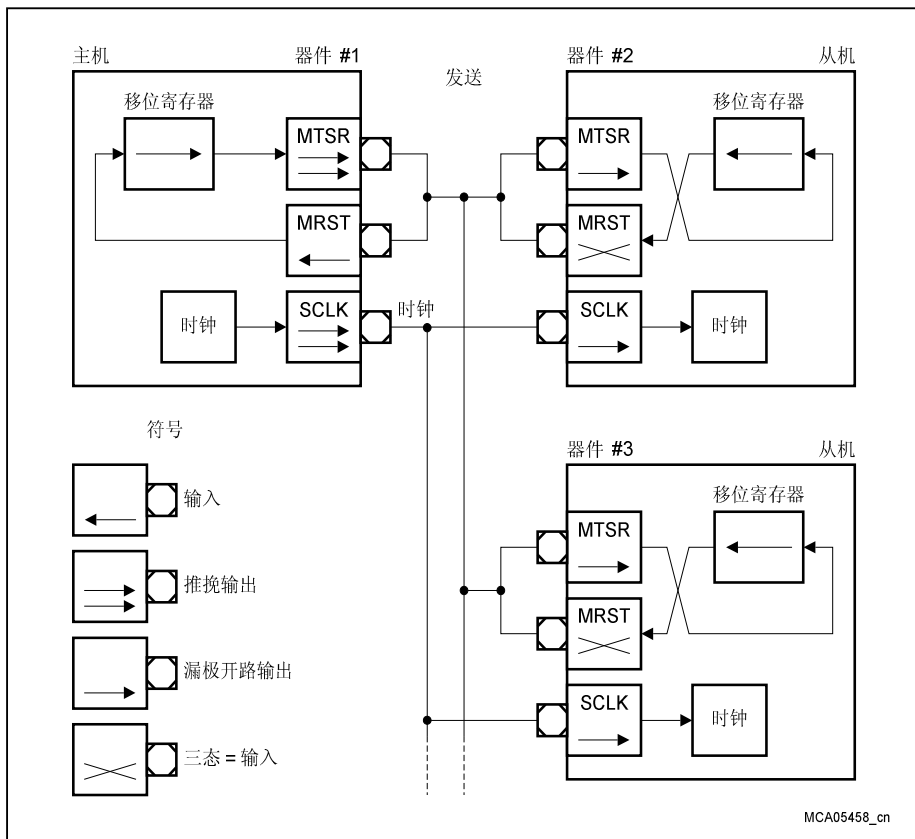


图 20-5 SSC 半双工设置

20.2.4 连续传送

当发送中断请求标志被置位，说明发送缓存寄存器 SSCx_TB 为空、并准备就绪可以装入新的发送数据。如果当前发送结束时 SSCx_TB 已经被重载，新数据立刻被传送到移位寄存器中，没有任何附加延时开始发送该数据。数据线上连续两次传送的数据帧之间没有间隙。例如，传送两个 8 位数据看上去和传送一个 16 位数据相同。该特性可用于和每次传送操作需要 16 位或者多于 16 位的器件接口，只需软件规定数据帧的总长度即可。该特性可使字节宽器件与字宽器件共用一个串行接口通信。

注：由于重新设定基本数据宽度必须在数据传送过程中禁止/使能 SSC，连续传送的数据宽度只能为基本数据宽度（编程设定）的整数倍。

20.2.5 波特率产生

串行通道 SSC 有专用的、具有 16 位重载能力的 16 位波特率发生器，从而无需由定时器产生波特率。图 20-6 所示为 SSC 波特率发生器的逻辑框图。

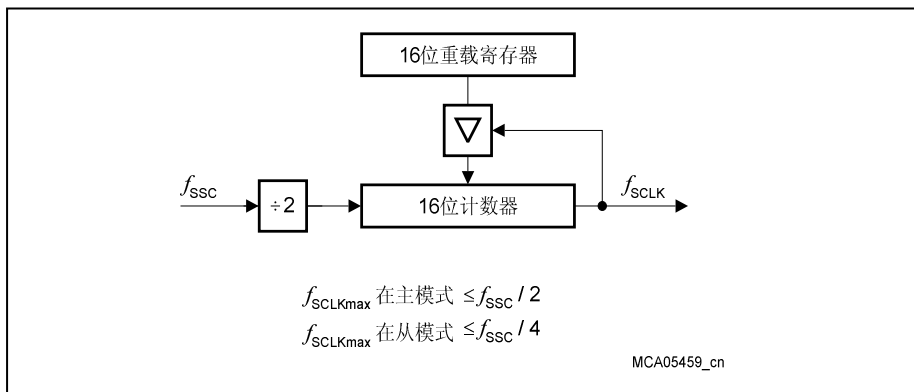


图 20-6 SSC 波特率发生器

波特率发生器的输入时钟为模块时钟 f_{SSC} 。计数器递减计数。通过寄存器 SSCx_BR 控制波特率发生器产生不同的波特率，具体描述如下。

波特率定时器/重载寄存器

SSC 的波特率定时器/重载寄存器 SSCx_BR 具有双重功能。

SSC 被禁止时，SSCx_BR 用作波特率定时器的重载寄存器，在该寄存器中写入需要的重载值。读取 SSCx_BR 返回当前的重载值。

SSC 被使能时，SSCx_BR 中的值代表当前波特率定时器的内容。SSC 被使能时不允许对该寄存器写操作。

波特率计算

定时器中装入重载值，SSC 被使能后立刻开始计数。根据下面的公式，给定重载值计算波特率（Baudrate），或给定波特率计算重载值：

$$\text{波特率} = \frac{f_{\text{SSC}}}{2 \times (
 + 1)} \quad BR = \frac{f_{\text{SSC}}}{2 \times \text{波特率}} - 1 \quad (20.1)$$

代表重载寄存器中的内容，当作 16 位无符号整数；波特率等于 f_{SCLK} ，如图 20-6 所示。

使用 40 MHz 的模块时钟，可获得的最大波特率分别为：主模式 20 Mbit/s（
=0000_H），从模式 10 Mbit/s（
=0001_H）。

表 20-1 列出一些可能的波特率、对应的重载值以及相应的波特率偏差，假定模块时钟为 40 MHz。

表 20-1 SSC 典型波特率（ $f_{\text{SSC}} = 40 \text{ MHz}$ ）

重载值	波特率（= f_{SCLK} ）	偏差
0000 _H	20,000,000 bit/s（只限主模式）	0.0%
0001 _H	10,000,000 bit/s	0.0%
0009 _H	2,000,000 Mbit/s	0.0%
0013 _H	1,000,000 bit/s	0.0%
001A _H	750,000 bit/s	-1.25%
0027 _H	500,000 bit/s	0.0%
0063 _H	200,000 bit/s	0.0%
00C7 _H	100,000 bit/s	0.0%
FFFF _H	306.6 bit/s	0.0%

20.2.6 检错机制

SSC 能够检测四种出错情况。主模式和从模式下均可检测接收出错和相位出错；发送出错和波特率出错只适用于从模式。当检测到某个错误时，寄存器 `SSCx_CON` 中对应的出错标志位被置位，通过激活出错中断请求线 `EIRQ` 产生出错中断请求（见图 20-7）。出错中断处理器会检查出错标志位以确定中断的起因。出错标志位不能被自动复位，中断被响应后必须由软件清零。从而允许通过中断响应某些出错情况；同时软件可查询其它出错标志位。

注：出错中断处理器必须对相关（被使能）的出错标志清零以防止重复的中断请求。

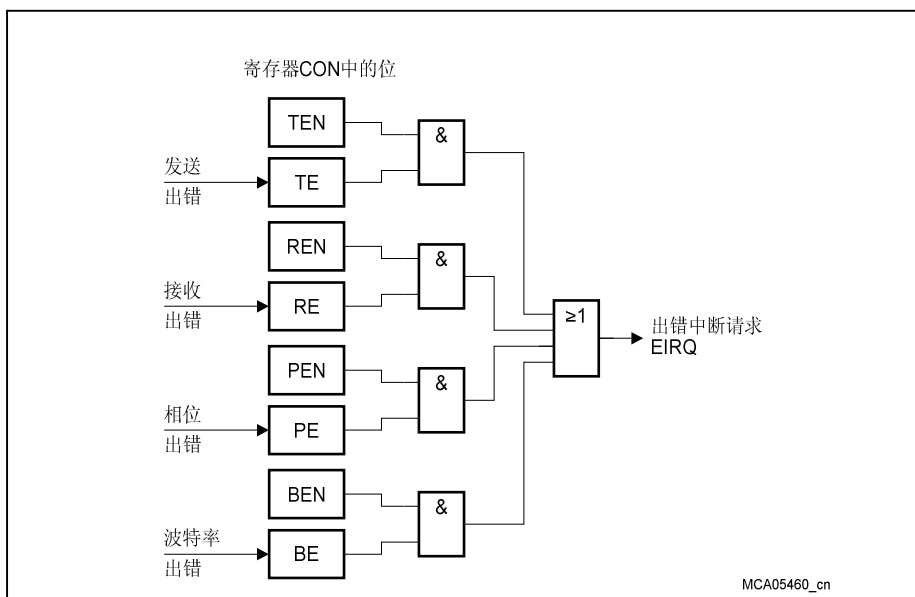


图 20-7 SSC 出错中断控制

接收出错（主模式或从模式）：当新数据帧已接收完毕，但上次数据仍未从接收缓存寄存器 `SSCx_RB` 中读出时，检测到接收出错。此时置位出错标志位 `RE`，并当 `REN` 使能该中断时，激活出错中断请求线 `EIRQ`。接收缓存寄存器 `SSCx_RB` 中的旧数据将被新数据覆盖、且不可恢复。

相位出错（主模式或从模式）：用和模块时钟相同的频率对引脚 `MRST`（主模式）或引脚 `MTSR`（从模式）上的输入数据采样，在移位时钟信号 `SCLK` 的锁存沿的一个周期之前和两个周期之后的采样数据改变，则检测到相位出错。此时置位出错标志位 `PE`，并当 `PEN` 使能该中断时，激活出错中断请求线 `EIRQ`。

波特率出错（从模式）：当输入时钟信号和设定的波特率之间的偏差超过 100%（即超过期望波特率的两倍、或小于期望波特率的一半）时，检测到波特率出错。此时置位出错标志位 **BE**，并当 **BEN** 使能该中断时，激活出错中断请求线 **EIRQ**。使用该检错功能时，要求所设定的从机波特率和主机器件相同。该特性可用于检测时钟线上错误的、多余的或丢失的脉冲（在一帧之内）。

*注：如果波特率出错且位 **AREN=1**，将自动执行 **SSC** 复位。如果检测到过少或过多的时钟脉冲，自动复位将重新初始化 **SSC** 模块。*

发送出错（从模式）：当主机已启动数据传送（**SCLK** 有效），但从机的发送缓存寄存器 **SSCx_TB** 自上一次传送之后还未更新，则检测到发送出错。此时置位出错标志位 **TE**，并当 **TEN** 使能该中断时，激活出错中断请求线 **EIRQ**。如果发送缓存还未被更新就开始进行数据传送，从机会把移位寄存器中的“旧”内容移出，通常为上次传送时接收的数据。如果该从机未被选中发送数据，这可能会导致半双工模式下发送/接收线上数据被破坏（设置为漏极开路输出）。该模式要求未被选中发送数据的从机只送出“1”；也就是说，在任何传送之前，必须将 **FFFF_H** 装入这些从机的发送缓存寄存器中。

注：带有推挽输出驱动的从机在未被选中发送数据时，通常关闭其输出驱动。但为了避免可能的数据冲突或数据错误识别，建议始终在任何传送之前，将值装入从机的发送缓存寄存器中。

可通过控制寄存器 **SSCx_CON** 中的出错状态标志确定中断请求源（接收、相位、波特率或发送出错）。

*注：进入出错中断服务程序时，出错标志位 **TE**、**RE**、**PE** 和 **BE** 不能被自动复位，必须由软件清零。*

20.2.7 SSC 寄存器总结

表 20-2 SSC 模块寄存器总结

寄存器名	说明	SSC0 地址		寄存器区	SSC1 地址	
		16 位	8 位		16 位	8 位
SSCx_CON	控制寄存器	FFB2 _H	D9 _H	SFR	FF5E _H	AF _H
SSCx_BR	波特率定时器重载寄存器	F0B4 _H	5A _H	ESFR	F05E _H	2F _H
SSCx_TB	发送缓存寄存器	F0B0 _H	58 _H	ESFR	F05A _H	2D _H
SSCx_RB	接收缓存寄存器	F0B2 _H	59 _H	ESFR	F05C _H	2E _H
SSCx_TIC	发送中断控制寄存器	FF72 _H	B9 _H	SFR/ ESFR	F1AA _H	D5 _H
SSCx_RIC	接收中断控制寄存器	FF74 _H	BA _H	SFR/ ESFR	F1AC _H	D6 _H
SSCx_EIC	出错中断控制寄存器	FF76 _H	BB _H	SFR/ ESFR	F1AE _H	D7 _H

20.2.8 端口配置要求

SSC 模块工作在主模式或从模式时，相关的端口寄存器设置如表 20-3 所示。

表 20-3 SSC0/SSC1 IO 选择和设置

模块	模式	端口线	复用功能选择寄存器	方向和端口输出寄存器	IO
SSC0	主模式	P3.8/MRST0	ALTSEL0P3.P8 = 1	DP3.P8 = 0	输入
		P3.9/MTSR0	ALTSEL0P3.P9 = 1	DP3.P9 = 1 和 P3.P9 = 1	输出
		P3.13/SCLK0	ALTSEL0P3.P13 = 1	DP3.P13 = 1 和 P3.P13 = 1	输出
	从模式	P3.8/MRST0	ALTSEL0P3.P8 = 1	DP3.P8 = 1 和 P3.P8 = 1	输出
		P3.9/MTSR0	ALTSEL0P3.P9 = 1	DP3.P9 = 0	输入
		P3.13/SCLK0	ALTSEL0P3.P13 = 1	DP3.P13 = 0	输入
SSC1	主模式	P1H.1/MRST1	ALTSEL0P1H.P1 = 1	DP1H.P1 = 0	输入
		P1H.2/MTSR1	ALTSEL0P1H.P2 = 1	DP1H.P2 = 1	输出
		P1H.3/SCLK1	ALTSEL0P1H.P3 = 1	DP1H.P3 = 1	输出
	从模式	P1H.1/MRST1	ALTSEL0P1H.P1 = 1	DP1H.P1 = 1	输出
		P1H.2/MTSR1	ALTSEL0P1H.P2 = 1	DP1H.P2 = 0	输入
		P1H.3/SCLK1	ALTSEL0P1H.P3 = 1	DP1H.P3 = 0	输入

注：根据所选的工作模式（主模式或从模式），寄存器 DP3 或 DP1H 中的端口方向控制位必须由软件置位或清零，不能由 SSC 模块的硬件逻辑自动控制。

20.3 SSC 模块接口

XC164CM 中的 SSC 模块和 IO 端口以及其它内部模块的接口如 [图 20-8](#) 和 [图 20-9](#) 所示。

SSC0 的输入/输出线和 P3 口的引脚相连，SSC1 的输入/输出线和端口 P1H 的引脚相连。每个模块中的三条中断请求线送至中断控制模块。

SSC 模块的时钟控制和仿真控制通过系统控制单元 SCU 进行控制。

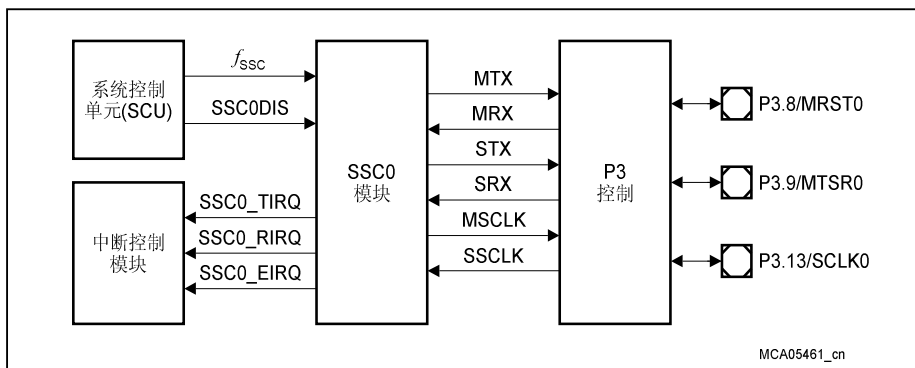


图 20-8 SSC0 模块接口

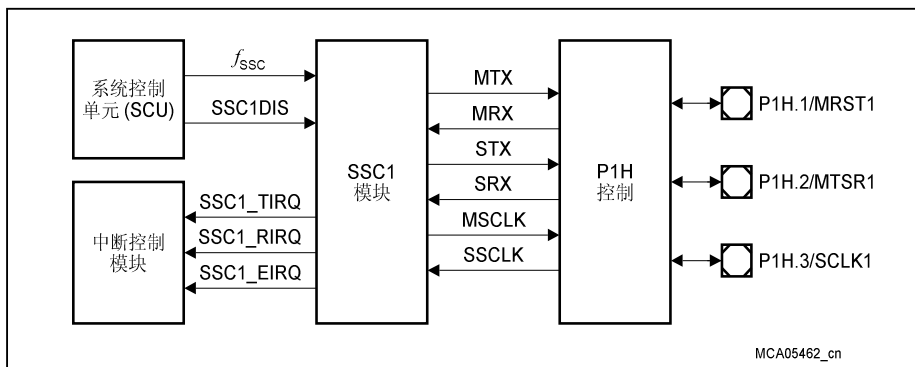


图 20-9 SSC1 模块接口

21 TwinCAN 模块

21.1 内核描述

21.1.1 概述

XC164CM 中的几款产品集成了 TwinCAN 模块（见**表 1-1**）。该模块包括两个全 CAN 功能节点，这两个 CAN 节点可独立工作或者通过网关功能交换数据帧和远程帧。CAN 帧的发送和接收遵循 CAN V2.0B（active）规范。每个 CAN 节点都能接收和发送带 11 个标识符的标准帧和带 29 位标识符的扩展帧。

两个 CAN 节点共享 TwinCAN 模块的资源，目的是优化 CAN 总线通信处理以及使 CPU 负荷最小。全 CAN 功能与 FIFO 结构的灵活组合可满足复杂嵌入式系统的实时要求。CAN 总线监控功能的改进和报文对象个数的增加使得 CAN 总线通信处理更加精确和方便。

根据应用需要，32 个报文对象可独立地分配给两个 CAN 节点之一。网关功能允许两个独立的 CAN 总线系统自动进行数据交换，减少了 CPU 负荷并且改进了整个系统的实时性能。

两个 CAN 节点的位定时源于外设时钟（ f_{CAN} ），可以通过编程使数据速率达到 1Mbps。每个 CAN 节点通过一对接收和发送引脚连接到总线收发器。

特性：

- CAN 功能符合 CAN 规范 V2.0 B
- 每个 CAN 节点都有专门的控制寄存器
- 支持的数据速率高达 1Mbps
- 具有灵活和功能强大的报文传送控制和错误处理能力
- 全 CAN 功能：32 个报文对象可以被分别地
 - 分配给两个 CAN 节点之一
 - 设置为发送或者接收对象
 - 2、4、8、16 或 32 级 FIFO 报文缓存区
 - 设置为处理带 11 位或 29 位标识符的帧
 - 提供用于滤波的可编程验收屏蔽寄存器
 - 通过帧计数器进行监控
 - 可设置为远程监控模式
- 多达 8 个单独的可编程中断节点
- 用于总线监控的 CAN 分析模式

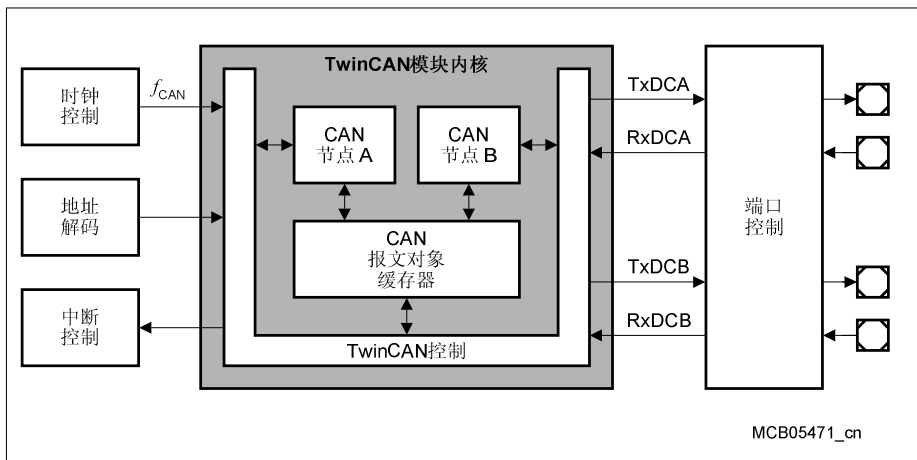


图 21-1 TwinCAN 模块总框图

CAN 内核（图 21-2）可被分为：

- 全局控制外壳，分为初始化逻辑电路，全局控制和状态逻辑以及中断请求压缩器。
 - 初始化逻辑在上电或复位之后，设置所有子模块。在完成 CAN 节点控制逻辑和相关报文对象的初始化之后，相关 CAN 节点和与其相连的 CAN 总线同步。
 - 全局控制和状态逻辑通知 CPU 有关挂起报文对象发送和接收中断以及最近的传送情况。
 - 中断请求压缩器将来自 72 个中断请求源（属于 CAN 节点 A 和节点 B 的），压缩至 8 个中断节点。
- 报文缓存单元，包含报文缓存区，FIFO 缓存区管理，网关控制逻辑和基于报文的中断请求产生单元。
 - 报文缓存单元最多保存 32 个报文对象，每个对象的最大数据长度为 8 字节。每个报文对象有一个标识符和自己的控制和状态位集。初始化之后，报文缓存单元无需 CPU 监控就能够处理数据的接收和发送。
 - FIFO 缓存管理把收到和要发出的报文对象保存在一个环形缓存区中，并确定 CAN 控制器要处理的下一个报文。
 - 网关控制逻辑将报文从 CAN 节点 A 传送到 CAN 节点 B，反之亦然。
 - 中断请求产生单元指示报文对象特定报文的接收或者发送。

- 两个独立的 CAN 节点，分为位流处理器，位定时控制单元，错误处理逻辑，中断请求产生单元和节点控制逻辑：
 - 位流处理器根据 ISO-DIS 11898 标准进行数据、远程、错误和过载帧的处理。该单元控制 CAN 总线，输入/输出移位寄存器和 CRC 寄存器之间的串行数据流，以及 I/O 移位寄存器和报文缓存单元之间的并行数据流。
 - 位定时控制单元根据传播延时和相位漂移误差，定义采样点，并执行重新同步。
 - 错误处理控制逻辑管理接收和发送错误计数器。根据两个定时器的内容，将 CAN 控制器置为：“错误激活”，“错误认可”或“总线关闭”状态。
 - 中断请求产生单元发出报文发送和接收操作成功结束的全局信号，以及所有类型的传送问题，如位填充错误，格式，应答，CRC 或位状态错误，及每个 CAN 总线警告级别的改变或者总线关闭状态。
 - 节点控制逻辑使能和禁止节点特定的中断源，进入 CAN 分析器模式和全局帧计数器的管理。

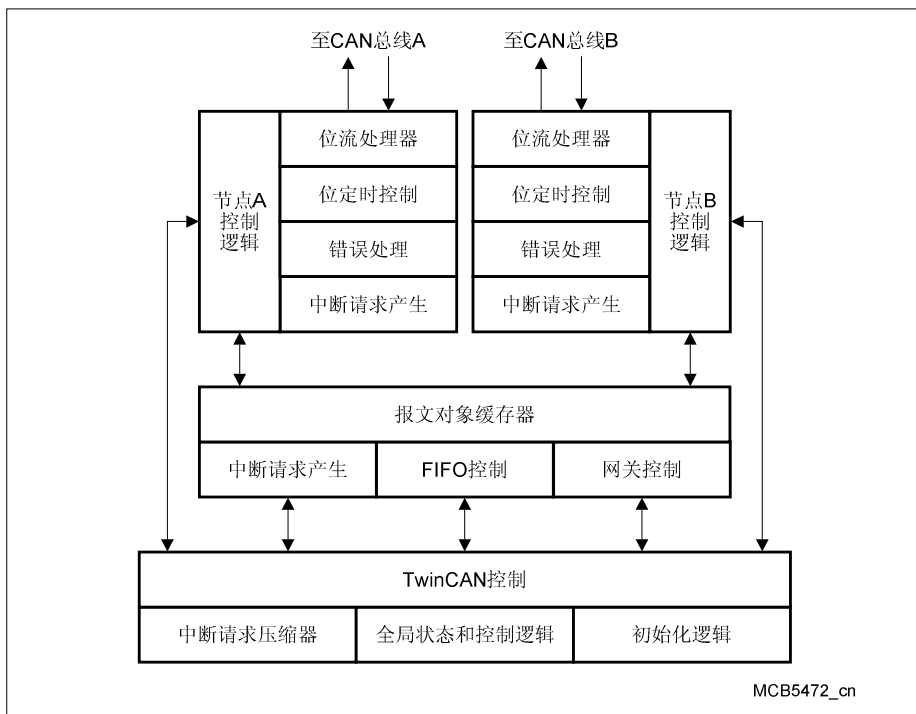


图 21-2 CAN 内核详细框图

21.1.2 TwinCAN 控制外壳

21.1.2.1 初始化处理

外部硬件复位之后或者处于总线关闭状态时，相关的 CAN 控制器节点与相应的 CAN 总线在逻辑上断开，不参与任何报文的传送。该状态由 ACR/BCR 控制寄存器中的位 INIT = ‘1’ 指示，在复位或者总线关闭的情况下该位自动置位。另外，软件置位 INIT 将使 CAN 节点与总线断开。当 INIT 被置位时，受其影响的 CAN 节点控制器和与之相连的 CAN 总线之间的所有报文传送停止，总线输出引脚（TXDC）电平保持为 ‘1’（隐性状态）。

在外部硬件复位之后，所有的控制和报文对象寄存器都复位到相应的值。在“总线关闭”期间或者在向 ACR/BCR 的 INIT 位置 1 的写操作之后，所有相关的控制和报文对象寄存器保持其当前值（错误计数器除外）。

不处于“总线关闭”状态的情况下，复位 INIT 为 ‘0’ 将启动同步序列（=连接到 CAN 总线上），允许该节点再次参加 CAN 通信前，至少需要在与其相连的总线上监测到一个“总线空闲事件”（11 个连续的“隐性”位）。

总线关闭恢复序列期间：

- 复位错误处理逻辑中的接收和发送错误计数器
- 必须检测到 128 个“总线空闲事件”（11 个连续的“隐性”位）之后，才能启动同步序列。进入到总线关闭状态之后，硬件立即开始监控总线空闲事件。由接收错误计数器计数并指示已检测到的总线空闲事件。
- 128 个总线空闲事件之后，重新连接过程通过硬件测试位 INIT。如果 INIT 仍然置位，受影响的 CAN 节点控制器等待，直到 INIT 清零，并且在该节点再次参与 CAN 通信之前，至少在 CAN 总线上检测到一个总线空闲事件。如果 INIT 已经清零，受影响的 CAN 节点控制器和与其相连的 CAN 总线之间的报文传送被立即使能。

21.1.2.2 中断请求压缩器

CAN 模块有 32×2 个报文对象专用中断请求源和 2×4 个节点控制中断请求源。中断请求压缩器将这 72 个请求源压缩至 8 个 CAN 中断节点，报告 CAN 模块的中断请求。每个请求源配置一个中断节点指针，它选择中断节点，用于启动相关的服务子程序，增加了中断处理的灵活性。8 个中断节点中的每一个都可以触发一个独立的中断子程序，拥有自己的中断向量及优先级。

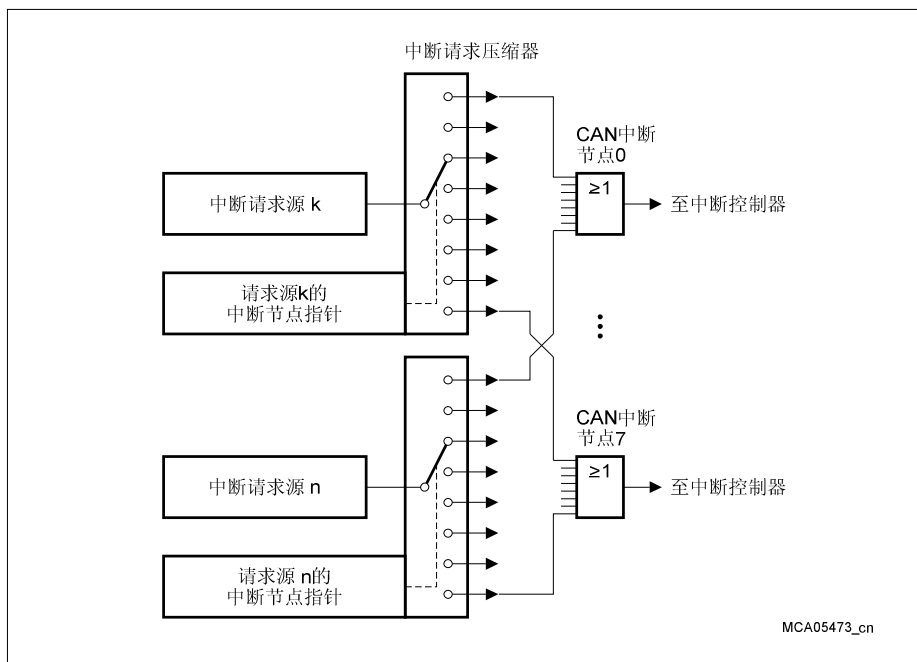


图 21-3 中断节点指针和中断请求压缩器

注：所有中断都是事件导向的。事件设置相应的指示标志且产生一个发向系统的中断。一个中断事件出现时，而其相应的指示标志仍置位状态下，也可产生一个新的中断。

21.1.2.3 全局控制和状态逻辑

接收中断挂起寄存器 **RXIPND** 包含 32 个单独的标志，用来指示相关报文对象挂起的接收中断。如果相应的报文对象已经正确接收一个数据或者远程帧，且相关的中断请求产生已经由 **RXIE_n** = ‘10’ 使能，硬件将置位标志 **RXIPND_n**。通过复位相应的报文对象控制寄存器 **MSGCTR_n** 中的位 **INTPND_n**，可以由软件清零 **RXIPND_n**。

发送中断挂起寄存器 **TXIPND** 和 **RXIPND** 功能相似，为挂起的发送中断提供同样的信息。

21.1.3 CAN 节点控制逻辑

21.1.3.1 概述

每个节点都有专用的节点控制逻辑，用来设置全局行为并提供状态信息。

当 ACR/BCR 寄存器位 CCE 置为 ‘1’，激活设置模式。该模式允许修改 CAN 位定时参数和错误计数器寄存器。

当控制寄存器 ACR/BCR 中的位 CALM 置为 ‘1’，激活 CAN 分析模式。该工作模式下，数据和远程帧被监控，而不参加任何 CAN 传送（CAN 发送引脚保持在隐性电平）。接收的远程帧保存到相应的发送报文对象中，接收的数据帧被保存在与其匹配的接收报文对象中。

CAN 分析模式下，接收帧的整个设置信息被保存到相应的报文对象中，可以由 CPU 评估它们的标识符，XTD 位信息和数据长度码。如果 RMM= ‘1’，远程监控模式有效，接收远程帧时，该信息也可用。接收来的帧将不被应答，也不产生错误帧。如果使能 CAN 分析模式，既不会有相应的数据帧来应答远程帧，也无法通过置位 TXRQ 发送数据帧。对于所有正确接收的帧都可产生接收中断（如果使能），且在接收远程帧情况下，相应远程挂起位 RMTPNd 被置位。

节点中断设置也由节点控制逻辑通过 ACR/BCR 位 SIE、EIE、LECIE 定义：

- 如果控制位 SIE 置为 ‘1’，当 ASR/BSR 寄存器被更新后（由每次成功完成报文传送引发），产生一个状态改变中断。
- 如果控制位 EIE 置为 ‘1’，当识别到总线关闭条件或者已经超过错误警告级别或者发生欠载时，产生一个错误中断。
- 如果位 LECIE 置为 ‘1’，当状态寄存器 ASR 或者 BSR 的位域 LEC 被置成一个错误码时，产生一个最近错误码中断。

状态寄存器（ASR/BSR）提供相关 TwinCAN 节点的当前状态总览：

- 当报文被成功发送且被至少一个其它的 CAN 节点应答时，置位标志 TXOK
- 标志 RXOK 指示 CAN 总线报文的一次无错误接收。
- 位域 LEC 指示 CAN 总线上出现了一个最近错误。也报告位填充错误、格式、CRC 错误和总线仲裁错误（位 0，位 1）。
- 错误处理逻辑中至少一个错误计数器达到错误警告界限时（缺省值为 96），位 EWRN 置 1。
- 当发送错误计数器超过错误界限 255，且相关的 CAN 节点控制器与相应的 CAN 总线逻辑上断开，置位 BOFF。

当从/向 CAN 总线发送或者接收一帧报文时，CAN 帧计数器可用于检查报文对象的传送序列或者获得即时时间信息。由一个 16 位计数器执行 CAN 帧计数，该计数器由寄存器 AFCR/BFCR 控制。位域 CFCMD 定义工作模式和使帧计数器增加的触发事件。

- 在正确发送帧之后

- 在正确接收帧之后
- 在 CAN 总线上出现一个外来帧之后（不是由该 CAN 节点本身发送/接收的）
- 一个新的位时间的开始

在所监测的帧传送结束时，捕获到的帧计数器值被复制到相关的 MSGCTRn 寄存器的位域 CFCVAL 中。帧计数器上溢条件（FFFF_H 到 0000_H）满足时，置位标志 CFCOV，同时如果位 CFCIE 置为 1，产生一个中断请求。

21.1.3.2 定时控制单元

根据 ISO-DIS 11898 标准，一个 CAN 位时间被划分为几个不同的时间段（图 21-4）。每个时间段由几个时间基本单元 t_q 组成。 t_q 的大小由位域 BRP 和位 DIV8X 调整，这两者都控制波特率预分频器（见位定时寄存器 ABTR/BBTR）。波特率预分频器由 CAN 模块时钟 f_{CAN} 驱动。

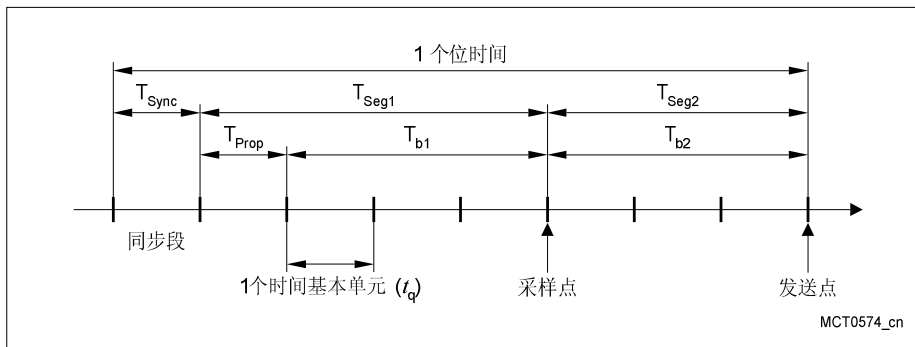


图 21-4 CAN 总线标准位定时图

在同步段 (T_{Sync}) 进行发射器和接收器的时间基准之间的相位同步。同步段长度始终为一个 t_q 。传播时间段 (T_{Prop}) 包括发射器输出驱动器中、CAN 总线上和收发器电路中的物理传播延时。对于工作冲突检测机制来讲， T_{Prop} 必须等于所有的传播延迟量的总和舍入到 t_q 的整数倍之后，再乘以 2 所得的时间值。采样点之前和之后的相位缓冲区间 1 和 2 (T_{b1} 和 T_{b2}) 用于补偿在同步段中检测到的发射器和接收器之间的时钟相位失配。

重新同步所允许的时间基本单元的最大值由位定时寄存器 ABTR/BBTR 的位域 SJW 定义。传播时间段和相位缓冲区间 1 组合形成参数 T_{Seg1} ，由相应的位定时寄存器 ABTR/BBTR 位域 TSEG1 的值定义，ISO 标准要求其最少为 3 个时间基本单元。参数 T_{Seg2} ，由位定时寄存器 ABTR/BBTR 位域 TSEG2 的值定义，包含了相位缓冲区间 2，ISO 标准要求其最少为 2 个时间基本单元。根据 ISO 标准，一个 CAN 位时间是 T_{Sync} 、 T_{Seg1} 和 T_{Seg2} 的总和，必须不少于 8 个时间基本单元。

注：仅在控制寄存器 ACR/BCR 中的控制位 CCE 置 1，即被使能情况下，才能访问位定时寄存器 ABTR/BBTR。

位时间计算：

$$t_q = (\text{BRP}+1) / f_{CAN} \quad \text{如果 DIV8X} = '0'$$

$$= (\text{BRP}+1) / 8 \times f_{CAN} \quad \text{如果 DIV8X} = '1'$$

$$T_{Sync} = 1 \times t_q$$

$$T_{Seg1} = (\text{TSEG1}+1) \times t_q \quad (\text{最小: } 3 t_q)$$

$$T_{Seg2} = (\text{TSEG2}+1) \times t_q \quad (\text{最小: } 2 t_q)$$

$$\text{位时间} = T_{Sync} + T_{Seg1} + T_{Seg2} \quad (\text{最小: } 8 t_q)$$

为了补偿不同 CAN 控制器之间的时钟相移，CAN 控制器必须在从隐性到显性总线电平的任意边沿上进行同步。如果硬件同步使能（在一帧的开始），在同步段重新开始一个位时间。否则，重新同步跳转宽度 T_{SJW} 定义时间基本单元的最大数目，因而重新同步操作可能缩短或拉长位时间。SJW 的值被写入寄存器 ABTR/BBTR 中。

$$T_{SJW} = (\text{SJW} + 1) \times t_q$$

$$T_{Seg1} \geq T_{SJW} + T_{Prop}$$

$$T_{Seg2} \geq T_{SJW}$$

f_{CAN} 的最大相对误差取决于相位缓冲区间和重新同步跳转宽度。

$$df_{CAN} \leq \min(T_{b1}, T_{b2}) / 2 \times (13 \times \text{位时间} - T_{b2}) \text{ 且}$$

$$df_{CAN} \leq T_{SJW} / 20 \times \text{位时间}$$

波特率计算：

$$\text{波特率} = f_{OSC} / ((1 + (\text{TSEG1}+1) + (\text{TSEG2}+1)) (1 + \text{BRP}))$$

21.1.3.3 位流处理器

根据报文缓存区中的报文对象，由位流处理器产生通过 CAN 总线发送的远程帧和数据帧。该单元控制 CRC 发生器并给新的远程帧和数据帧添加校验和信息。在包括帧开始位 SOF 和帧结束位 EOF 之后，位流处理器开始进行 CAN 总线仲裁过程并且发现总线空闲时继续进行帧发送。发送数据的同时，位流控制器连续地监控 I/O 线。如果（总线仲裁阶段和应答间隙之外）在 I/O 线上的电压电平和发送移位寄存器当前发送位的逻辑状态之间检测到失配，产生一个‘最近错误’中断请求，错误码由状态寄存器 ASR/BSR 的位域 LEC 指示。

通过检验相关的 CRC 域，验证当前接收帧的数据一致性。当检测到一个错误时，产生一个‘最近错误’中断请求，错误码由状态寄存器 ASR/BSR 给出。此外，产生一个错误帧并将其发送到 CAN 总线上。将一个无错误的帧分解成标识符和数据部分之后，接收到信息被传送到报文缓存区，执行远程帧和数据帧处理，中断产生和状态处理。

21.1.3.4 错误处理逻辑

错误处理逻辑负责对 CAN 设备进行故障界定。该单元的 2 个计数器，接收错误计数器和发送错误计数器（控制寄存器 AECNT, BECNT）根据来自位流处理器的命令增加和减少。如果在发送操作运行中，位流处理器本身检测到了一个错误，发送错误计数器加 8。当一个外部 CAN 节点通过产生错误帧报告一个错误，节点计数器加 1。为了进行错误分析，被扰乱的报文和节点的传送方向，传送错误的认可，都由控制寄存器 AECNT、BECNT 指示。根据错误计数器中的值，设置 CAN 节点，使其进入到如下状态：“错误激活”，“错误认可”或“总线关闭”。

如果两个错误计数器的值都低于“错误认可”界限 128，CAN 处在“错误激活”状态。如果至少两者之一等于或者大于 128，CAN 节点处在“错误认可”状态。

如果发送错误计数器的值等于或者大于“总线关闭”界限 256，“总线关闭”状态被激活。由状态寄存器 ASR/BSR 中的标志 BOFF 报告该状态。直到完成“总线关闭恢复序列”，器件一直保持在该状态。另外，当至少一个错误计数器等于或超过控制寄存器 AECNT、BECNT 中位域 EWRNLVL 定义的错误警告值时，置位状态寄存器 ASR/BSR 中的位 EWRN。如果两个错误计数器的值再次降到错误警告值以下，位 EWRN 被复位。

21.1.3.5 节点中断处理

每个 CAN 节点都配备有 4 个中断源，支持

- 全局发送/接收逻辑
- CAN 帧计数器
- 出错报告系统

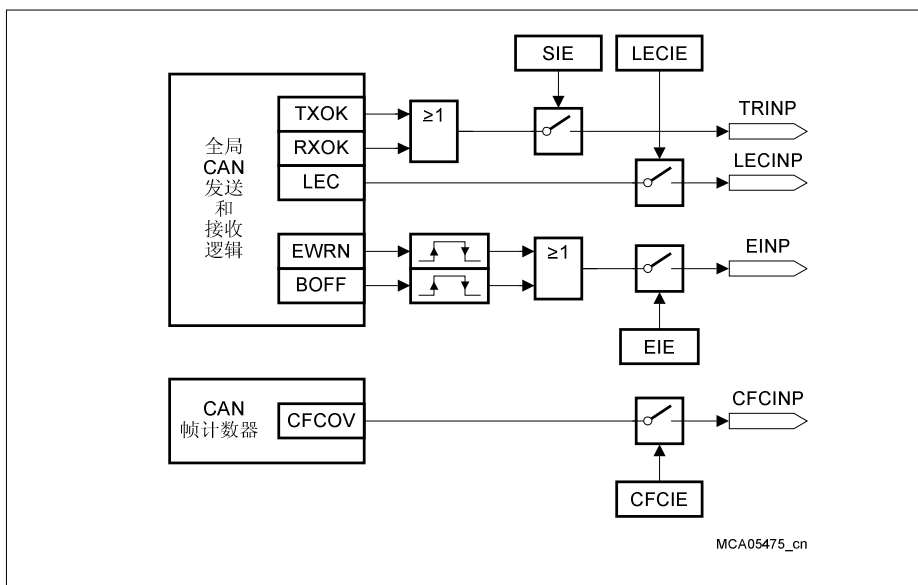


图 21-5 特定节点中断控制

如果由寄存器 ACR/BCR 中位 SIE= ‘1’ 使能，完成一个无错的发送或者接收报文对象之后，节点状态寄存器（ASR/BSR）被更新，全局发送/接收逻辑产生中断请求。相关的中断节点指针由控制寄存器 AGINP/BGINP 中的位域 TRINP 定义。

如果由寄存器 ACR/BCR 中的位 LECIE= ‘1’ 激活，最近错误码中断请求可报告一个错误。相应的中断节点指针由控制寄存器 AGINP/BGINP 中的位域 LECINP 定义。

当 AFCR/BFCR 控制寄存器中位 CFCIE 置为 1 时，CAN 帧计数器溢出时产生中断请求。由寄存器 AGINP/BGINP 中的位域 CFCINP 选择相应的中断节点指针。

错误逻辑监控 CAN 总线错误的数目，根据错误计数器中的值设置或者复位错误警告位 EWRN。如果控制寄存器 ACR/BCR 的位 EIE 置 1，那么位 EWRN 和 BOFF 的任何修改都产生中断请求。相关的中断节点指针由控制寄存器 AGINP/BGINP 中的位域 EINP 定义。

21.1.3.6 报文中断处理

每个报文对象具有 2 个中断请求源，用来指示报文发送或接收的成功结束。

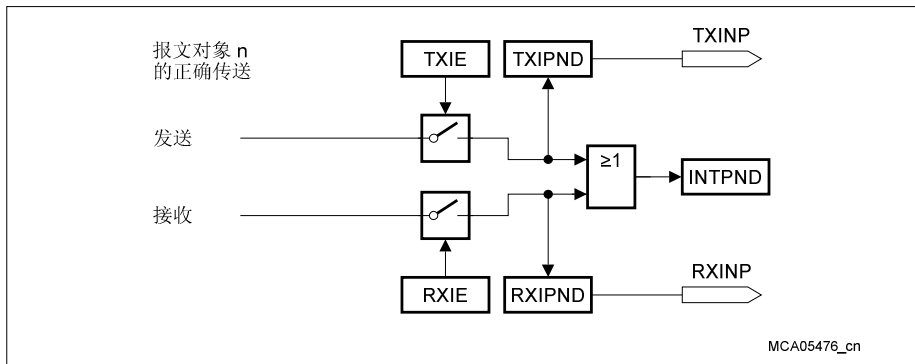


图 21-6 特定报文控制中断

如果相应的报文控制寄存器 MSGCTRn 中的位 TXIE 或 RXIE 置为 ‘10’，报文传送中断源被使能。相关的中断节点指针由报文设置寄存器 MSGCFGn 中的位域 RXINP 和 TXINP 定义。

21.1.3.7 中断指示

AIR/BIR 寄存器提供一个 INTID 位域，用来指示具有最高内部优先级（报文对象号最小）的挂起中断请求源。监控到的中断请求类型，结合位域 INTID，可由寄存器 AIMR0/AIMR4 和 BIMR0/BIMR4 进行选择，这些寄存器中包含对应每个中断源的屏蔽位。如果没有挂起的中断请求，AIR/BIR 中的所有位被清零。中断请求 INTPNDn 必须由软件清零。

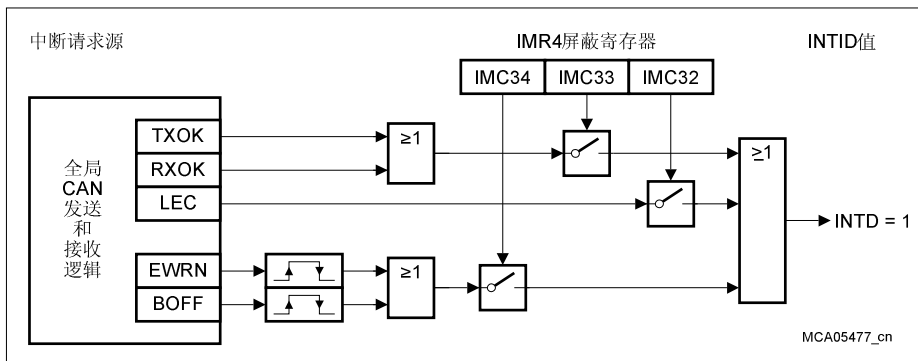


图 21-7 全局中断请求源的 INTID 屏蔽

寄存器 AIMR0/4 和 BIMR0/4 对应每一个中断源包含一个屏蔽位（AIMR0/ BIMR0 对应与报文相关的中断源，AIMR4/ BIMR4 对应与节点相关的中断源）。如果屏蔽位被复位，在产生 INTID 值时，则不考虑相应的中断源。

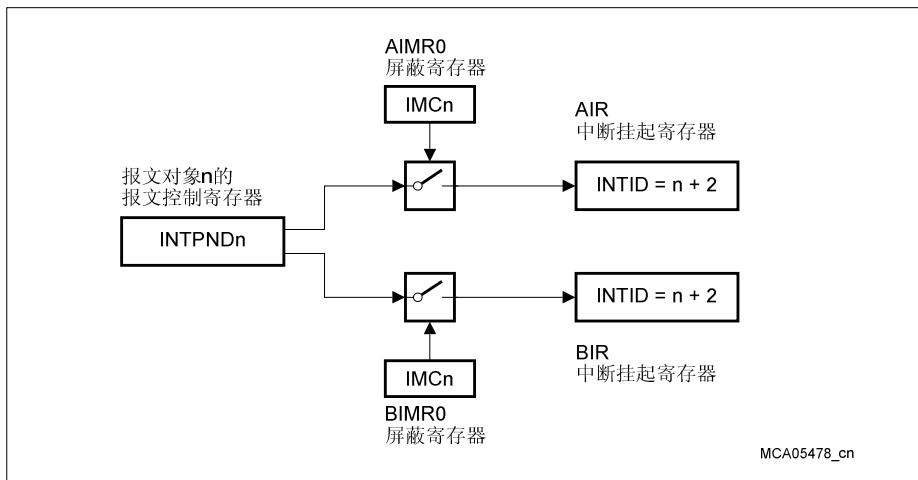


图 21-8 报文中断源请求 INTID 屏蔽

21.1.4 报文处理单元

报文对象是 CPU 和 CAN 控制器之间交换的基本信息单元。CAN 内部存储器提供 32 个报文对象。每个报文对象都有一个标识符，一套控制位和状态位和一个单独的数据区域。每个报文对象包含 32 个字节的内部存储器，被分为如图 21-9 中的控制寄存器和数据存储器部分。

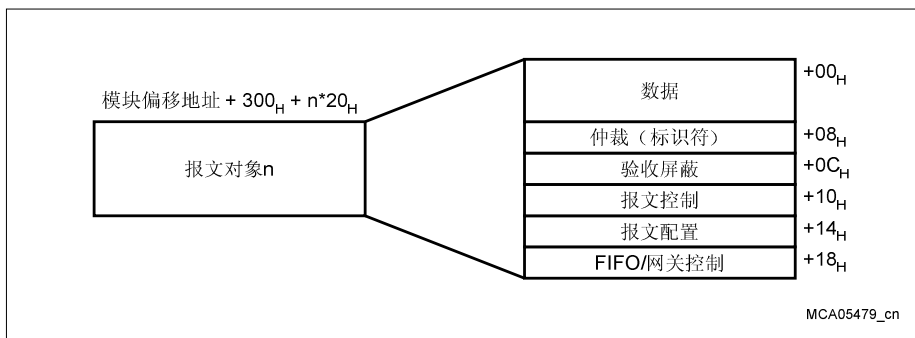


图 21-9 报文对象结构

正常工作模式下，每个报文对象都和一个 CAN 节点相联系。仅在共享网关模式下，两个 CAN 节点都可以访问同一个报文对象（根据相应的位域 NODE）。

报文对象必须在相应的报文控制寄存器（位 MSGVAL）声明有效，才能被相应的 CAN 节点控制逻辑处理。

当 CPU 初始化一个报文对象时，应当复位报文控制寄存器 MSGCTRn 中的位域 MSGVAL，禁止 CAN 节点控制寄存器对相关寄存器和数据缓存区存储器的读/写访问。随后，必须定义报文标识符和工作模式（发送，接收）。如果成功发送和/或接收之后，应当执行一个中断服务程序，相应的位域 TXIE 和 RXIE 必须被置位，并且中断挂起指示器（位域 INTPND）应当被复位。

如果接收来的带有匹配标识符的远程帧不请求自动响应，相应的发送报文对象应设置为 CPUUPD = ‘10’。

一旦位域 MSGVAL 被置为 ‘10’，相应的报文对象可工作，且由相应的节点控制器处理。

21.1.4.1 仲裁和验收屏蔽寄存器

仲裁寄存器 MSGARn 用来对接收报文进行滤波，也给发送报文提供标识符。验收屏蔽寄存器 MSGAMRn 可在验收测试时，将收到的报文的某些标识符位屏蔽起来。

接收到的报文标识符和保存在内部 CAN 控制器存储器中的所有报文对象的标识符比较（按位 XOR）。从报文对象 0 开始且对符合下述条件的所有报文对象都进行比较操作：

- 有效的报文标志（MSGVAL= ‘10’）
- 适当的节点声明（寄存器 MSGCFGn）
- 对于数据帧接收，DIR 控制位已为零（即接收报文对象）
- 对于远程帧接收，DIR= ‘1’（即发送报文对象）
- 匹配的标识符长度声明（XTD= ‘1’ 指示扩展的 29 位标识符，XTD= ‘0’ 指示标准的 11 位标识符）

比较操作的结果和验收屏蔽寄存器中的内容逐位相与（图 21-10）。如果检测到是一致的，接收到的报文存储到 CAN 控制器的报文对象中。比较操作在分析完报文对象 31 之后结束。

注：根据分配的标识符和相应屏蔽寄存器中的内容，可能有多个报文对象可以满足上述选择标准。在这种情况下，接收到的帧保存在这些合适的报文对象中编号最低的报文对象内。

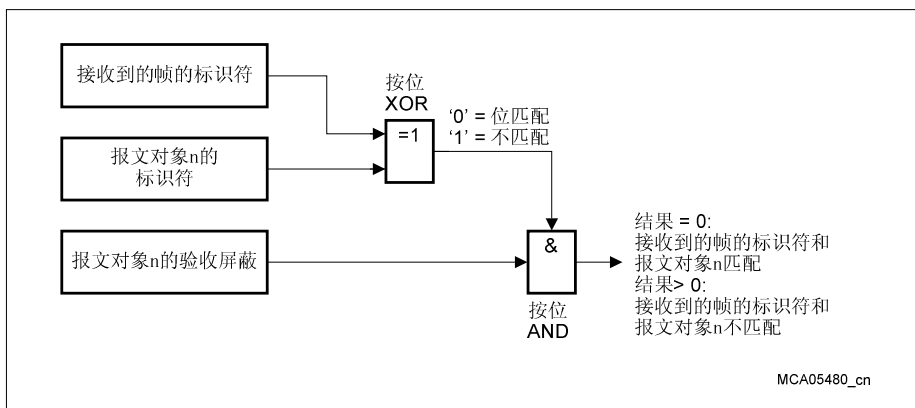


图 21-10 报文对象标识符验收滤波

21.1.4.2 数据帧和远程帧处理

报文对象可以根据控制位 DIR 的值，建立接收或者发送操作。报文对象类型对相关 CAN 节点控制器产生或接收数据帧和远程帧的影响见 [表 21-1](#)。

表 21-1 远程帧和数据帧处理

	给这个报文对象 一个发送请求 (TXRQ=1) 产生...	如果接收到带 有匹配标识符 的数据帧...	如果接收到带有匹配 标识符的远程帧...
接收对象（接收数 据帧，发送远程 帧，控制位 DIR= ‘0’ ）	...一个远程帧。接 收时所请求的数据 帧被保存到这个报 文对象中。	...数据帧保存 在这个报文对 象中。	...忽略这个远程帧。
发送对象（发送数 据帧，接收远程 帧，控制位 DIR= ‘1’ ）	...一个数据帧，它 包含了该报文对象 所保存的信息。	数据帧不被保 存。	...远程帧保存在这个 报文对象中， RMTPND 和 TXRQ 被置为 ‘10’ 。 如果 CPUUPD 被置 为 ‘01’ ，将自动产 生一个数据帧，它包 含了保存在这个报 文对象中的信息。

21.1.4.3 发送报文对象处理

方向标志 DIR= ‘1’（报文设置寄存器 MSGCFGn）的报文对象被作为发送报文对象处理。

位域 MSGVAL= ‘10’ 的所有报文对象都是可以工作的，并参与下述的 CAN 节点控制器操作。

在初始化阶段，发送请求位域（TXRQ），新信息位域（NEWDAT）应该复位为 ‘01’，且寄存器 MSGCTRn 中正在由 CPU 更新的位域 CPUUPD 应当被复位为 ‘10’。要发送的报文字节被写入到报文对象的数据部分（MSGDRn0，MSGDRn4）。要发送的报文字节数目必须写入到寄存器 MSGCFGn 的位域 DLC 中。所选标识符必须写入到寄存器 MSGARn 中。然后，寄存器 MSGCTRn 中的位域 NEWDAT 应该置为 ‘10’，且由 CPU 将位域 CPUUPD 复位到 ‘01’。

使能远程监控模式时（寄存器 MSGCFGn 中的 RMM= ‘1’），如果在对所有 CAN 报文对象做比较和屏蔽操作中发现匹配的标识符，接收到的远程帧的标识符和数据长度码被复制到相应的发送报文对象中。如果 MSGAMRn 中的某些屏蔽位被置为 ‘0’，该复制过程可能会改变发送报文对象中的标识符。

只要寄存器 MSGCTRn 中的位域 MSGVAL 置为 ‘10’，接收带有匹配标识符的远程帧的操作将自动置位位域 TXRQ 为 ‘10’。与此同时，寄存器 MSGCTRn 中位域 RMTPNd 置为 ‘10’，指示接收到一个通过验收的远程帧。或者，另一种方式，由 CPU 写访问寄存器 MSGCTRn 来置位 TXRQ。如果相应的 CAN 控制器节点发现发送请求位域 TXRQ 为 ‘10’（而 MSGVAL= ‘10’ 和 CPUUPD= ‘01’），可根据保存在相应的发送报文对象中的信息产生一个数据帧，并当相应的 CAN 总线空闲时自动传送该数据帧。

如果寄存器 MSGCTRn 中的位域 CPUUPD 置为 ‘10’，禁止报文对象的自动发送，且相应的 CAN 节点控制器不评估标志 TXRQ。CPU 通过清零 CPUUPD，可以解除挂起的发送。这个特性允许用户在软件控制下，侦听总线并应答远程帧。

当发送报文对象的数据部分必须由 CPU 更新时，报文控制寄存器 MSGCTRn 的位域 CPUUPD 应该被置为 ‘10’，以禁止相关 CAN 节点控制器的读或写访问。如果报文对象对存储的数据进行更新期间，收到带有匹配标识符的远程帧，位域 TXRQ 和 RMTPNd 被自动置为 ‘10’，并且相应的数据帧发送被挂起直到 CPUUPD 再次复位。

如果相应的 CAN 节点控制器发现几个带有挂起发送请求的有效报文对象，最先发送编号最低的报文对象的内容。

当所选报文对象数据寄存器中的内容被复制给位流处理器，相关 CAN 节点控制器内部复位位域 NEWDAT。当报文对象已经成功发送，位域 RMTPNd 和 TXRQ 自动复位。

帧计数器的捕获值复制到寄存器 MSGCTRn 中的位域 CFCVAL，如果 TXIE= ‘10’，且发送中断请求被使能（INTPNdn 和 TXIPNDn 被置位），产生发送中断请求。这时如果控制寄存器 AFCE/BFCE 使能，则帧计数器加 1。

当接收到带有匹配标识符的数据帧，相应的发送对象忽略该事件，并且不通过任何中断请求指示该事件。

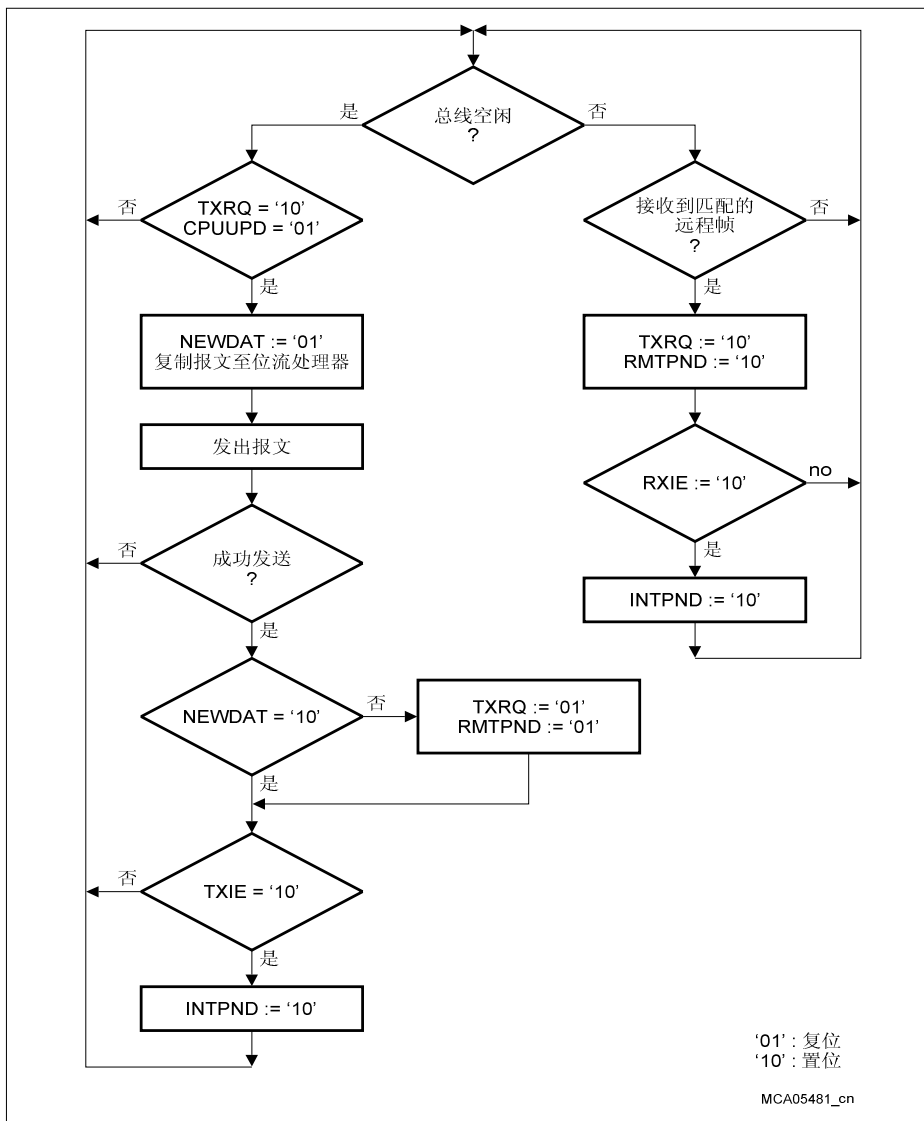


图 21-11 方向 = '1' 的报文对象的处理 = 由 CAN 硬件节点控制器发送

21.1.1.4.4 接收报文对象处理

方向标志 DIR= ‘0’（报文设置寄存器 MSGCFGn）的报文对象被作为接收报文对象处理。

在初始化阶段，寄存器 MSGCTR 中的发送请求位域（TXRQ），报文丢失位域（MSGLST）和 NEWDAT 位域都应该被复位。

位域 MSGVAL= ‘10’ 的所有报文对象都是可操作的，并参与下述的 CAN 节点控制器操作。

当接收到一个数据帧时，报文对象的数据区（MSGDRn0，MSGDRn4）用来保存新信息并用接收到的字节数更新寄存器 MSGCFG 位域 DLC。未用到的报文字节将被不确定的值覆盖。如果寄存器 MSGCTR 的 NEWDAT 仍置位，CAN 控制器认为先前存入的报文被覆盖，并通过设置位域 MSGLST 指示出数据损失。在任何情况下，位域 NEWDAT 自动被置位为 ‘10’，报告 CAN 控制器进行了一次数据寄存器更新。捕获到的帧计数器的值复制到寄存器 MSGCTRn 的位域 CFCVAL 中，且如果 RXIE= ‘10’ 使能时，产生一个接收中断请求（INTPNDn 和 RXIPNDn 被置位）。如果控制寄存器中的 AFCR/BFCR 使能，那么帧计数器加 1。

当接收报文对象被标记为要发送报文（TXRQ= ‘10’），位 MSGLST 自动变为 CPUUPD。如果 CPUUPD 被复位为 ‘01’，CAN 控制器产生一个远程帧，通过 CAN 总线发送给其它的通信参与者。如果 CPUUPD= ‘10’，远程帧传送一直被禁止，直到通过复位 CPUUPD 为 ‘01’，CPU 解除发送挂起。当远程帧被成功发出时，RMTPND 和 TXRQ 被自动复位。最后，如果发送中断被使能（TXIE= ‘10’），可产生一个发送中断请求。

当接收到带有匹配标识符的远程帧时，不对其进行应答，也不通过产生中断请求指示该事件。

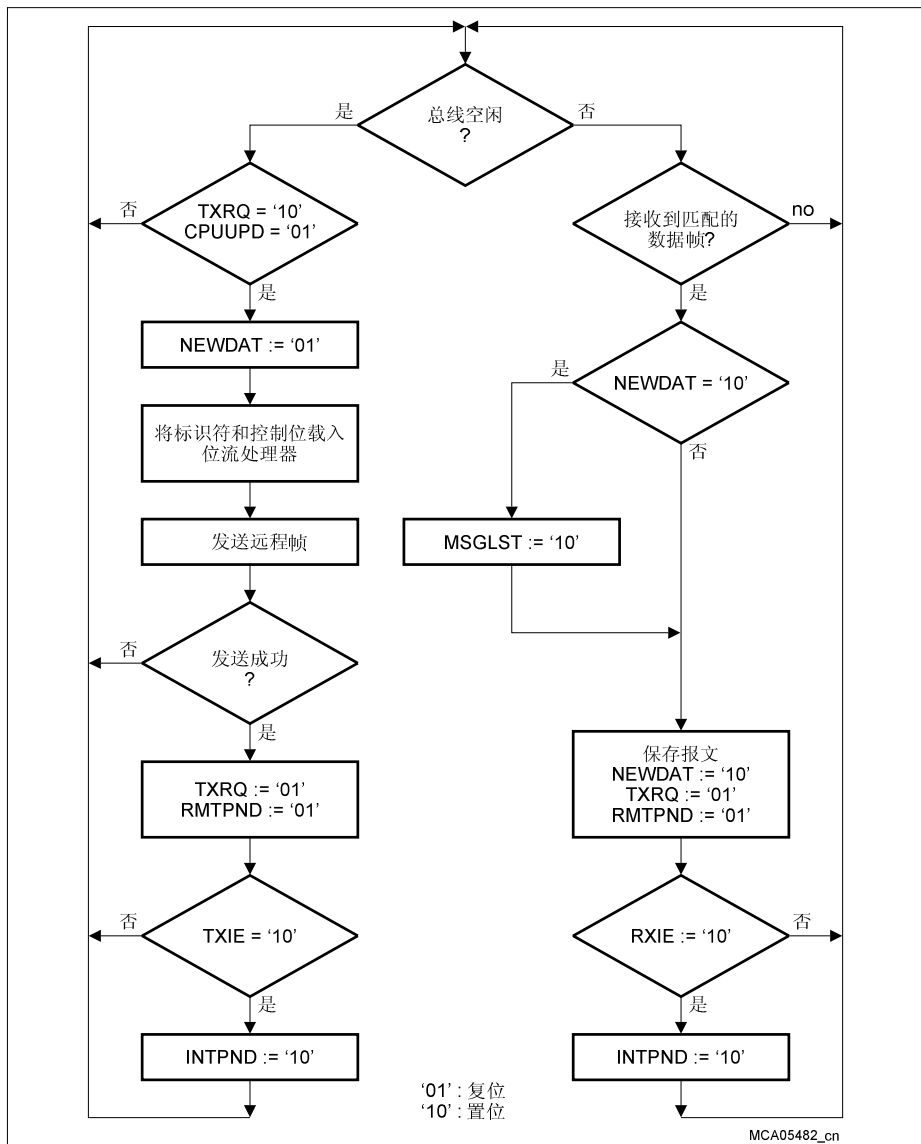


图 21-12 方向 = ‘0’ 的报文对象的处理 = 由 CAN 硬件节点控制器接收

21.1.4.5 单数据传送模式

单数据传送模式是一个有用的特性，可在 CAN 总线上广播数据，且避免无意地复制信息。由 FIFO/网关控制寄存器 MSGFGCRn 中的位 SDT 选择该单数据传送模式。

如果 MSGVAL 置为 ‘10’，接收到的每个带匹配标识符的数据帧都被自动保存在相应的接收报文对象中。当在一个短的时间间隔内，接收到的多个数据帧寻址同一个报文对象时，如果 CPU 没有及时处理报文对象中先前的内容，可能会发生信息丢失（由 MSGLST= ‘10’ 指示）。

根据报文对象的内容，由数据帧应答收到的每个带匹配标识符的远程帧。根据被验收的远程请求个数，该行为可能导致产生和发送多个相同的数据帧。

如果 SDT 置为 ‘1’，报文对象在接收到带有相应的标识符的数据帧之后，CAN 节点控制器自动对位 MSGVAL 复位。所有其它随后的数据帧，因为它们寻址被禁止的报文对象，这些数据帧一直被忽略，直到 CPU 再次置位 MSGVAL。

如果 SDT 置为 1，当相应的数据帧已经被成功发送，CAN 节点控制器自动复位所寻址的报文对象的位 MSGVAL。结果是，随后所有与被禁止的报文对象相关的远程请求都被忽略，直到 MSGVAL 被 CPU 再次置位。该特性允许连续地发送数据，而并不会无意地将信息加倍。

如果 SDT 清零，CAN 节点控制器不会复位位域 MSGVAL。

21.1.5 CAN 报文对象缓存（FIFO）

CPU 高负荷的情况下，在相应的报文对象被 CAN 控制器提供的下一个输入数据流覆盖之前，处理完接收的数据帧可能有些困难。根据具体的应用，有必要保证最小的数据帧产生速率以满足外部的实时要求。

因此，TwinCAN 模块实现了报文缓存区功能，以避免接收时的报文损失，同时使得发送报文的建立时间最短。一些报文对象可被设置为基本对象，利用随后的从属报文对象作为单独的缓存存储器（建立环形缓存区，用做报文 FIFO）。

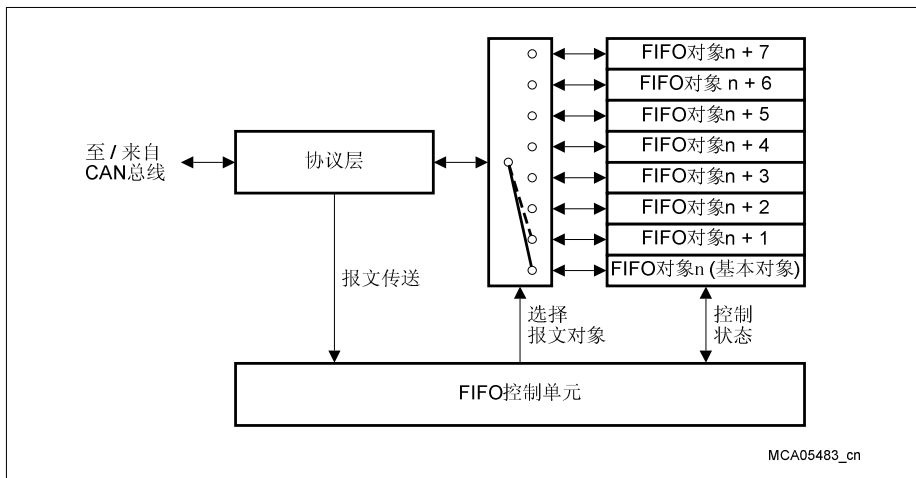


图 21-13 FIFO 缓存区控制结构

组成缓存区基本和从属报文对象的个数，必须是 2 的整数次幂（2、4、8 等），同时缓存区的基地址应当是缓存区长度的整数倍（例如缓存区包含 8 个报文，可以用对象 0、8、16 或 24 作为基本对象，如表 21-2 所示）。

通过将控制寄存器 MSGFGCRn 的位域 MMC 设置为 '010' 来定义基本对象，需要的缓存区大小通过适当地选择 FSIZE 的值来决定。从属对象由设置 MMC 为 '011' 定义。同一个 FIFO 中，所有 FIFO 元素的位域 FSIZE 都必须相等。

表 21-2 具有 FIFO 基本对象功能的报文对象

报文对象 n > FIFO 尺寸	0	2	4	6	8	10	12	14	16	18	...	30
2 级 FIFO	×	×	×	×	×	×	×	×	×	×		×
4 级 FIFO	×	—	×	—	×	—	×	—	×	—		—
8 级 FIFO	×	—	—	—	×	—	—	—	×	—		—
16 级 FIFO	×	—	—	—	—	—	—	—	×	—		—
32 级 FIFO	×	—	—	—	—	—	—	—	—	—		—

作为接收 FIFO 的情况下（DIR = ‘0’），属于同一个缓存区的所有 FIFO 元素的标识符和相应的验收屏蔽位都必须一致。在发送 FIFO 的情况下（DIR = ‘1’），发送时才考虑当前被寻址的报文对象的标识符。

缓存区结构中的每一个元素都保持各自的 MSGVAL, NEWDAT, CPUUPD 或 MSGLST, TXRQ 和 RMTEND 标志，及其单独的中断控制设置。在 FIFO 缓存区内部，所有元素必须满足下列条件：

- 分配给同一个 CAN 节点（寄存器 MSGCFGn 控制位 NODE）
- 编程为同样的传送方向（控制位 DIR）
- 设置了相同的标识符长度（控制位 XTD）
- 设置了相同的 FIFO 长度（位域 FSIZE）且
- 建立了相同的 FIFO 方向（寄存器 MSGFGCRn 的位 FD）
- 从属报文对象的 CANPTR 必须指向 FIFO 基本对象

基本对象的 CANPTR 必须用基本对象的报文编号进行初始化，从属报文对象的 CANPTR 指针必须设置为基本对象中的报文编号。信息传送时，基本对象的 CANPTR 寻址下一个要访问的 FIFO 元素，可根据下述规则计算其值：

$$\text{CANPTRn(新值)} = \text{CANPTRn(旧值)} \& \sim \text{FSIZEn} \mid (\text{CANPTRn(旧值)} + 1) \& \text{FSIZEn}$$

控制位 FD 定义了导致 CANPTR 位域更新的传送行为（接收或发送）。位 FD 和 FIFO 元素的方向位 DIR 相互独立。如果 FD = ‘0’，数据帧的接收（DIR = ‘0’）或者远程帧的接收（DIR = ‘1’）是导致 CANPTR 更新的接收动作。如果 FD = ‘1’，数据帧的发送（DIR = ‘1’）和远程帧的发送（DIR = ‘0’）是导致 CANPTR 增加的发送动作。

注：报文对象总存储容量不受缓存区结构的影响。可用的存储容量可以是不带缓存区结构的 32 个报文对象或者由报文对象组成的深度为 32 的缓存区。另外，根

据 FIFO 规则，只要不超过 32 个报文对象的限制，可以建立带缓存区或者未带缓存区的报文对象任意组合。

21.1.5.1 CAN 控制器对缓存区的访问

由相关的 CAN 控制器管理报文缓存区和 CAN 总线之间的数据传送。每个缓存区都按照 FIFO 算法（先进，先出 = 先改写）存储报文，由 CAN 控制器以环形次序传递报文。

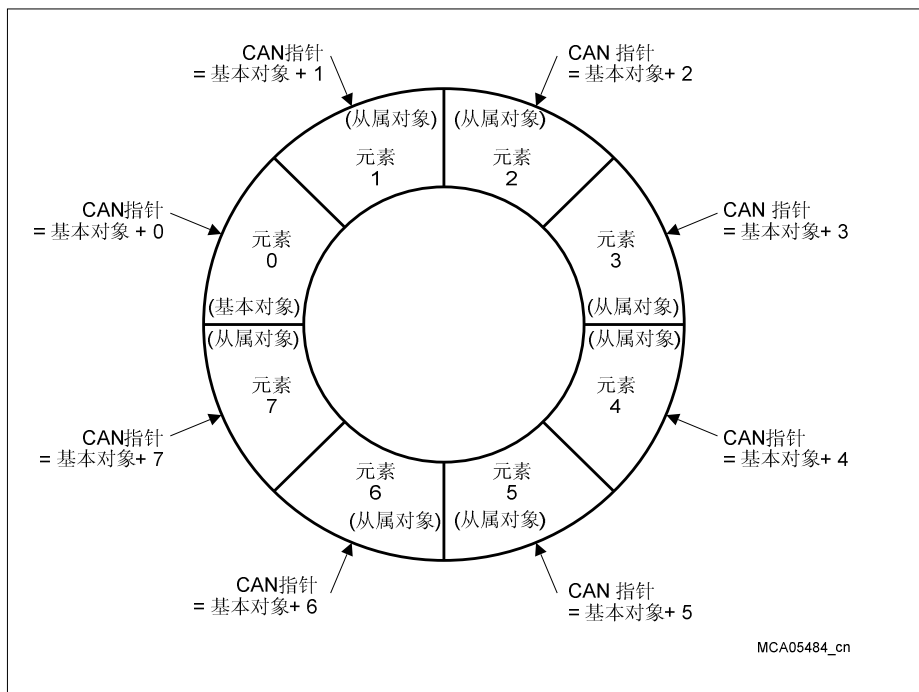


图 21-14 一个基本对象，7 个从属对象的 FIFO 缓存区结构

如果将 FIFO 缓存区初始化为接收，最先通过接收验收的报文保存在基本对象中（编号为 n ），第二个报文被写入到缓存区元素 $(n+1)$ ，依此类推。用于存储下一个输入的报文缓存区元素的编号：由基本对象的控制寄存器 MSGFGCRn 位域 CANPTR 给出。如果保留的缓存区空间已经用完，再次寻址基本报文对象（后面跟随连续的从属报文对象），用基本报文对象保存接收的下一个报文。如果 CPU 没有及时读出报文对

象中的内容，先前的报文被覆盖，该报文丢失情况由相应的 MSGCTR 寄存器标志 MSGLST 指示。

如果 FIFO 缓存区被初始化为发送报文对象，CAN 控制器开始传送缓存区元素 0（FIFO 基本对象）的内容，且控制寄存器 MSGFGCRn 的位域 CANPTR 加 1，指向下一个要发送的 FIFO 元素。

如果当前被基本对象的 CANPTR 寻址报文对象无效（MSGVAL= ‘01’），则禁止该 FIFO 传送数据。这种情况下，不考虑其它 FIFO 元素的位域 MSGVAL（包括当前不被寻址的基本对象）。

如果 FIFO 基本对象的位域 MSGVAL 设置为 ‘10’，当前被寻址的 FIFO 从属对象的位域 MSGVAL 设置为 ‘01’，数据将不会传送给这个从属报文对象，而 FIFO 基本对象的位域 CANPTR 还是按照 FIFO 规则加 1。

如果 FIFO 设置为发送数据帧，并且检测到和 FIFO 中一个元素相匹配的远程帧，则将自动置位相应报文对象中的发送请求和远程帧挂起位。根据 FIFO 规则和 FIFO 基本对象中的位域 CANPTR 的值处理数据帧的发送请求。

21.1.5.2 CPU 对缓存区的访问

必须由软件管理缓存区和 CPU 之间的报文传送。CPU 可以直接访问所有组成缓存区的报文对象。CPU 访问报文对象寄存器的操作不会自动修改控制寄存器 MSGFGCRn 中的位域 CANPTR。

21.1.6 网关报文处理

无需 CPU 干预，CAN 模块支持两个独立的 CAN 总线系统之间的信息自动传送。

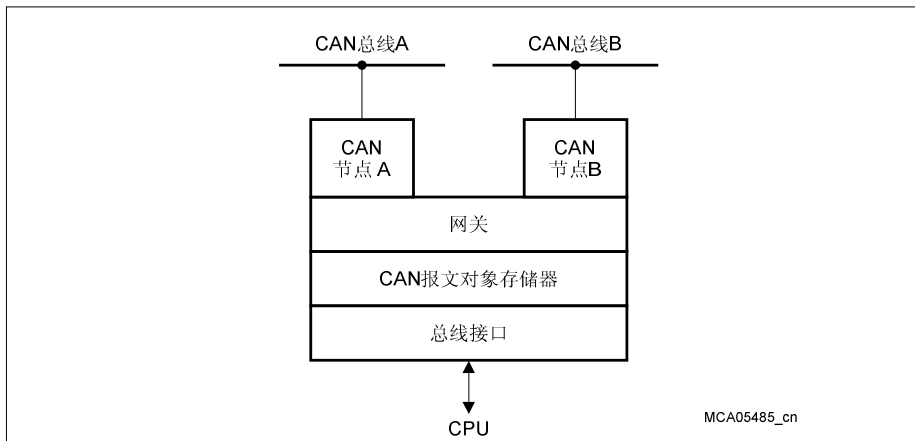


图 21-15 TwinCAN 网关功能

通过两个 CAN 节点共享 CAN 报文对象存储器的方式处理网关功能。报文存储器中的每个对象都通过报文设置寄存器 MSGCFGn 中的位 NODE 和节点 A 或节点 B 相关联。CAN 节点之间的信息交换可通过将两个报文对象结合（正常网关模式）或共享一个报文对象（共享网关模式）的方式来处理。

在下面几段中，接收数据帧的网关方叫做“源”（由<s>表示），通过网关发送数据帧的网关一方，被叫做“目标”（由<d>表示）。为了和该符号一致，通过网关的远程帧在目标方接收，而在源方发送。

由 FIFO/网关控制寄存器 MSGFGCRn 中的位域 MMC 定义报文对象的网关功能和所请求的信息传送模式。

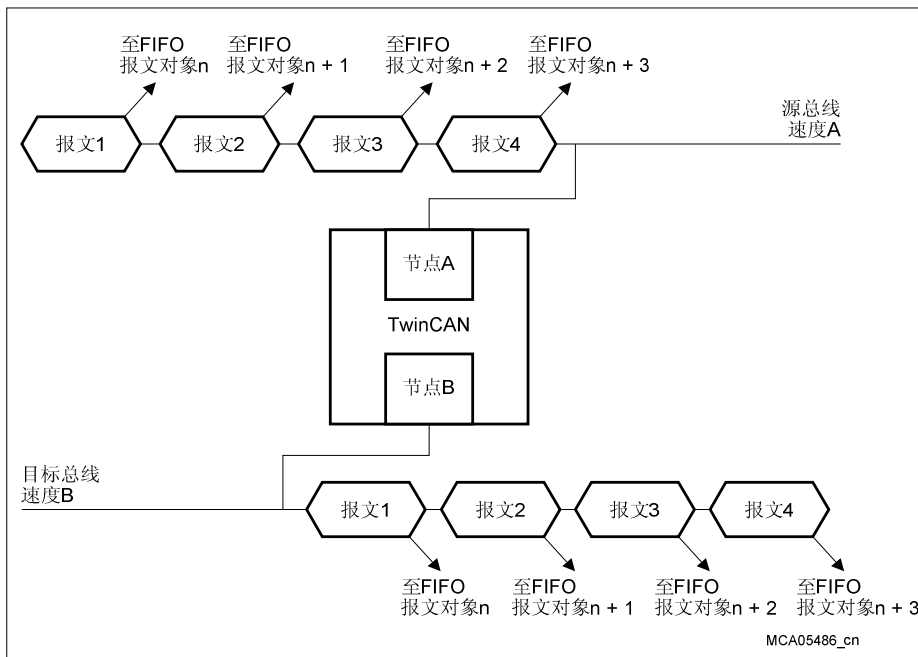


图 21-16 FIFO/网关模式下报文传送次序

21.1.6.1 正常网关模式

正常网关模式中从源向目标节点传送一个报文需要两个报文对象。在该模式下，同样的报文数据可使用不同的标识符。通过正常网关进行报文传送的详细信息由相应的 $MSGFGCR_{<s>}$ 和 $MSGFGCR_{<d>}$ 寄存器控制。来自源报文对象的所有 8 个数据字节（即使不是所有的字节都有效）被复制到目标报文对象。

从源节点接收信息的报文对象必须设置为接收报文对象（ $DIR = 0$ ），且必须通过位 **NODE** 和源 CAN 总线相关联。寄存器 $MSGFGCR_{<s>}$ 按照下列要求初始化：

- 位域 $MMC_{<s>}$ 必须设置为 ‘100’，表示正常网关模式用于接收到来的（数据）帧
- 位域 $CANPTR_{<s>}$ 必须初始化为数据复制过程的目标报文对象编号
- 如果网关目标方不需要 FIFO 功能，位域 $FSIZE_{<s>}$ 必须填充为 ‘00000’。当需要 FIFO 功能时，位域 $FSIZE_{<s>}$ 必须包含该 FIFO 缓存区的长度，且必须和网关目标方 FIFO 基本对象位域 $FSIZE$ 的内容一致。
- 当位 $IDC_{<s>}$ 被置位，源报文对象的标识符被复制到目标报文对象。否则，目标报文对象的标识符不被修改。

- 如果 **DLCC_{<s>}** 被置位，源报文数据长度码被复制到目标对象中。
- 由位 **GDFS_{<s>}** 决定：在完成数据复制过程之后，是否置位目标方的发送请求标志（如果 **GDFS_{<s>} = '1'**，**TXRQ_{<s>} = '10'**）。如果控制位 **CPUUPD_{<d>}** 复位为 **'01'**，则自动发送复制到目标方的数据帧。

CANPTR_{<s>} 寻址的目标报文对象，必须设置为发送操作（**DIR = 1**）。根据所需要的功能，目标报文对象可设置为以下三种工作模式：

- **MMC_{<d>} = '000'**，目标报文对象被声明为标准报文对象。这种情况下，如果被相关的控制位 **CPUUPD_{<d>}** 和 **GDFS_{<s>}** 使能，网关源方接收的数据帧可以自动发送到目标方。目标方接收的远程帧，不能被传送到源方，但是如果 **CPUUPD_{<d>}** 复位为 **'01'**，由目标报文对象直接应答该远程帧。
- **MMC_{<d>} = '100'**，目标报文对象被声明为正常网关模式，处理接收到的（远程）帧。如果使能（**CPUUPD_{<d>}**，**GDFS_{<s>}**），源方接收到的数据帧自动在目标方上发送。如果由位 **SRREN_{<d>} = '1'** 使能，目标方接收到的远程帧在源方自动发送。
- **MMC_{<d>} = '01x'**，目标报文对象被设置为 **FIFO** 的一个元素，用来缓存源方通过网关传送的数据帧。如果 **CPUUPD_{<d>}** 被复位（位 **SRREN_{<d>}** 必须被清零），目标方接收的远程帧，不被传送到源方，而直接由当前寻址的 **FIFO** 元素应答。
- 根据 **FIFO** 规则，远程帧的处理完全由目标方完成。

MMC_{<d>} = '000' :

目标方，标准报文对象的操作如图 21-17 所示：

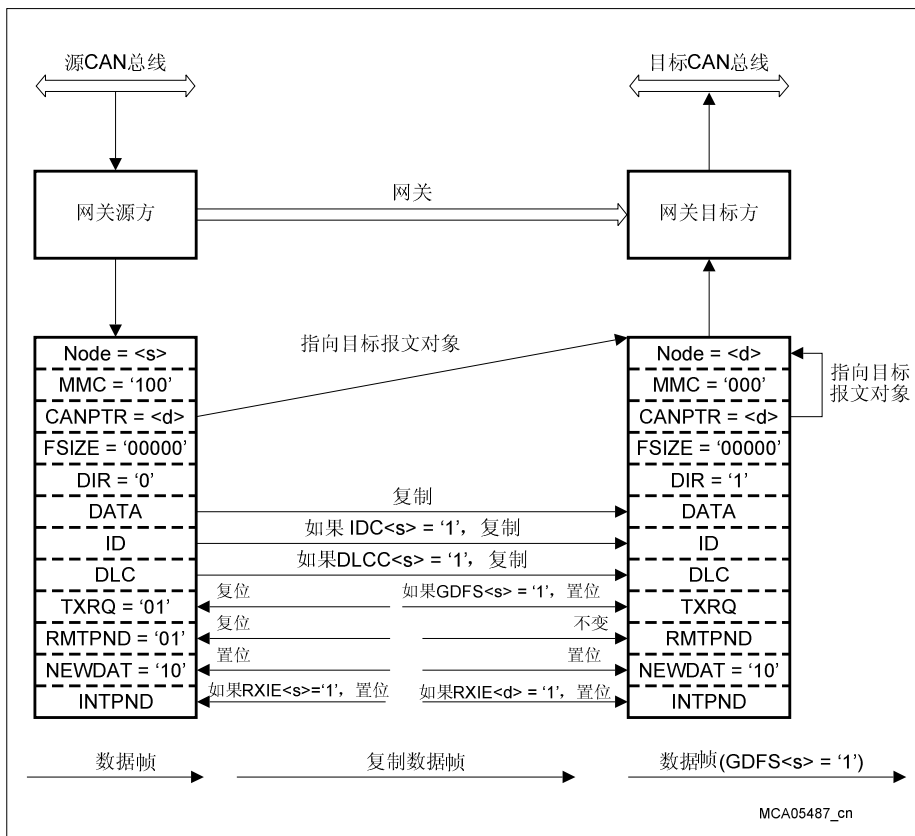


图 21-17 带标准目标报文对象 (MMC_{<d>} = '000') 的正常网关模式下，数据帧接收

到达源节点的匹配数据帧，被自动复制到目标节点中由 CANPTR_{<s>}寻址的报文对象中。位域 CANPTR_{<d>}载入值是目标报文对象编号。不考虑控制位 SRREN_{<d>}，目标节点接收到的远程帧，不会被传送到源节点，而是能够直接由目标报文对象应答。为了实现该特性，如果 CPUUPD_{<d>}复位为 '01'，控制位 TXRQ_{<d>}和 RMPND_{<d>}置为 '10'，在目标方 CAN 总线上立即开始数据帧发送。

MMC_{<d>} = '100' :

正常网关模式下，目标方上接收（远程）帧时报文对象的操作如**图 21-18**所示。

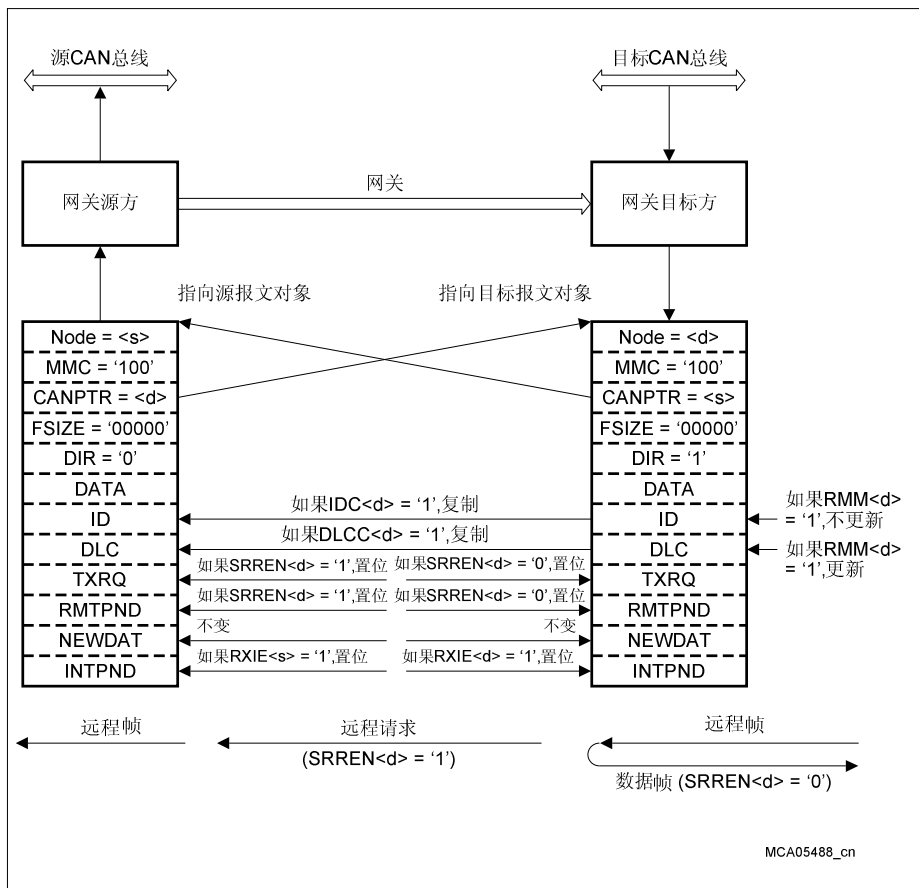


图 21-18 **MMC_{<d>} = '100' 正常网关模式下，远程帧传送**

目标方的网关对象设置为发送对象，可以接收远程帧。如果相关的网关控制寄存器 MSGFGCRn 中位 SRREN_{<d>} 被清零，CAN 目标节点控制器直接应答带有匹配标识符的远程帧。为了实现此功能，如果 CPUUPD_{<d>} 被复位，且控制寄存器 TXRQ_{<d>} 和 RMT PND_{<d>} 设置为 '10'，目标 CAN 总线立即开始发送数据帧。当位 SRREN_{<d>} 置为

‘1’，目标方接收到的远程帧通过网关传送到源方，且被 CAN 源节点控制器再次发送。

通过 CPU 置位 $\text{TXRQ}_{\langle\text{CS}\rangle}$ ，来启动源方网关报文对象的发送请求，该请求总是在源 CAN 总线系统上产生一个远程帧。

21.1.6.2 带 FIFO 缓存的正常网关模式

$\text{MMC}_{\langle\text{d}\rangle} = \text{'01x'}$:

当网关目标对象被设置为 FIFO 缓存时，位域 $\text{CANPTR}_{\langle\text{CS}\rangle}$ 用作 FIFO 元素的指针，用来寻址下一个复制过程的目标对象。必须将 $\text{CANPTR}_{\langle\text{CS}\rangle}$ 初始化为目标方 FIFO 基本对象的编号。当一个数据帧被复制到指定的目标方 FIFO 元素中时，根据 FIFO 规则自动更新 $\text{CANPTR}_{\langle\text{CS}\rangle}$ 。位 $\text{GDFS}_{\langle\text{CS}\rangle}$ 决定，从源方接收的数据帧复制到所选择目标方 FIFO 元素之后，是否置位该元素中的位 $\text{TXRQ}_{\langle\text{CS}\rangle}$ 。

用 $\langle\text{ba}\rangle$ 表示基本报文对象， $\langle\text{sl}\rangle$ 表示从属报文对象。组合成目标方缓存区的基本对象和从属对象的个数，必须是 2 的整数次幂（2、4、8 等等），且缓存区的基地址必须是缓存区长度的整数倍。FIFO 基本元素的位域 $\text{CANPTR}_{\langle\text{ba}\rangle}$ 和位域 $\text{CANPTR}_{\langle\text{CS}\rangle}$ 必须初始化为同样的起始值（FIFO 基本元素的报文对象编号）。所有 FIFO 从属元素的 $\text{CANPTR}_{\langle\text{sl}\rangle}$ 必须初始化为 FIFO 基本元素的报文对象编号。所有 FIFO 元素的位域 $\text{FSIZE}_{\langle\text{d}\rangle}$ 必须包含 FIFO 缓存区长度，且必须和 $\text{FSIZE}_{\langle\text{CS}\rangle}$ 内容一致。

图 21-19 为目标方带 FIFO 缓存区的正常网关操作。

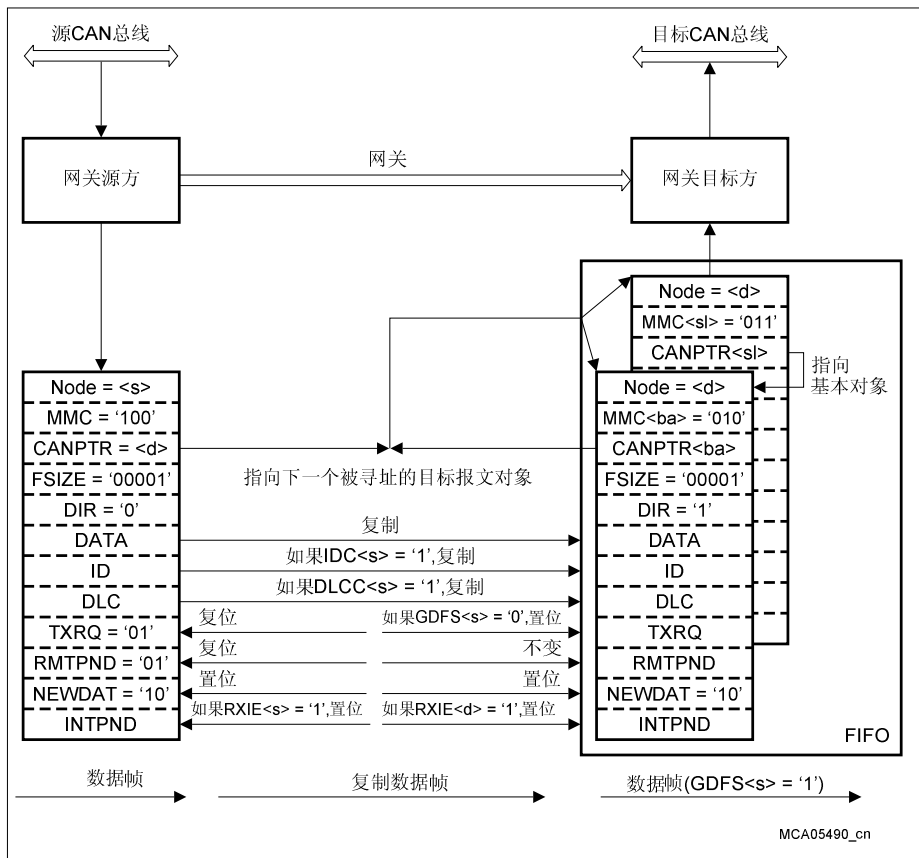


图 21-19 正常网关模式下，目标方带 2 级 FIFO (MMC_{<d>} = '01x') 数据帧传送

目标方 FIFO 元素接收到的远程帧，不能自动传送给源方。因此，和目标方 FIFO 元素关联的 SRREN_{<d>} 控制位必须清零，从而能够直接用数据帧应答接收到的带匹配标识符的远程帧。

从目标方到源方的远程请求以缓存方式传送，可通过一个软件程序处理，该程序设置 FIFO 缓存的网关模式，用来传送数据帧。目标方 FIFO 缓存的元素应当设置为 CPUUPD_{<d>} = '10' 的发送报文对象。接收到的带匹配标识符远程帧应当对所寻址的 FIFO 报文对象，发起一个中断服务请求。相关的中断服务子程序可以将接收到的远程帧的报文标识符和数据长度码复制到和源方 CAN 节点关联的接收报文对象中。在任何情况下，所选择的接收报文对象 TXRQ 必须设置为 '10'，以启动源方的远程帧发送。

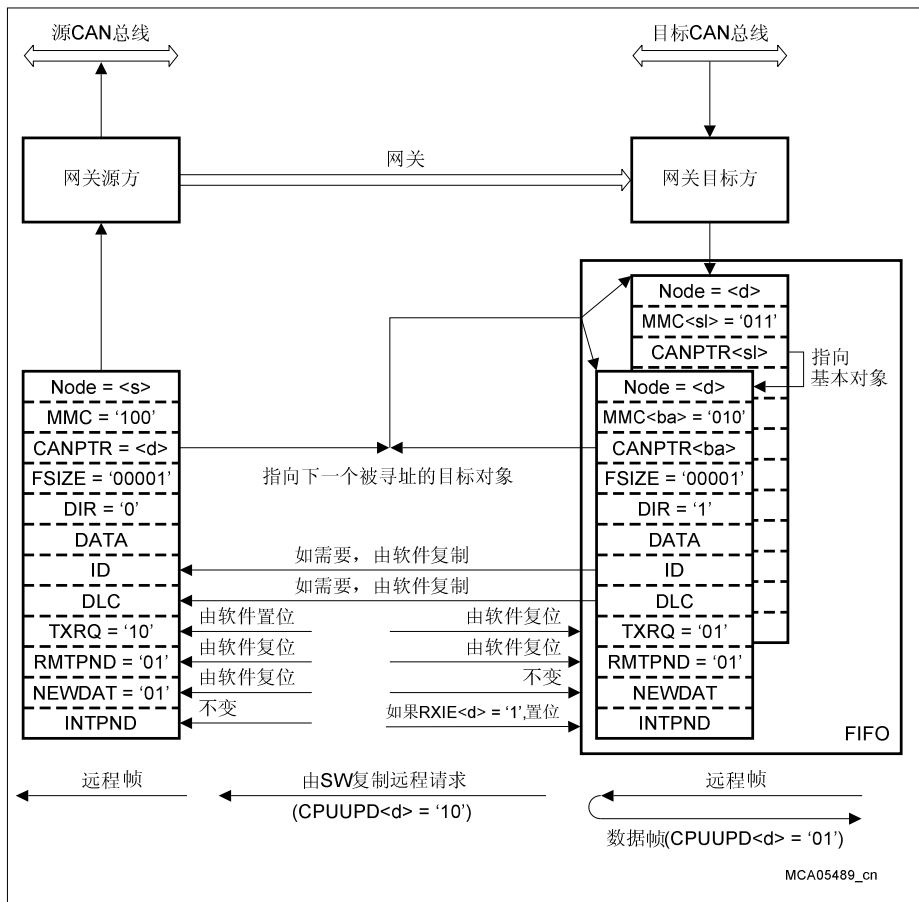


图 21-20 目标方带 2 级 FIFO 的正常网关模式下，远程帧传送

21.1.6.3 共享网关模式

在共享网关模式下，实现网关功能仅需要一个报文对象。该报文对象可以被当作在源和目标 CAN 节点之间切换的正常报文对象，如图 21-21 所示。

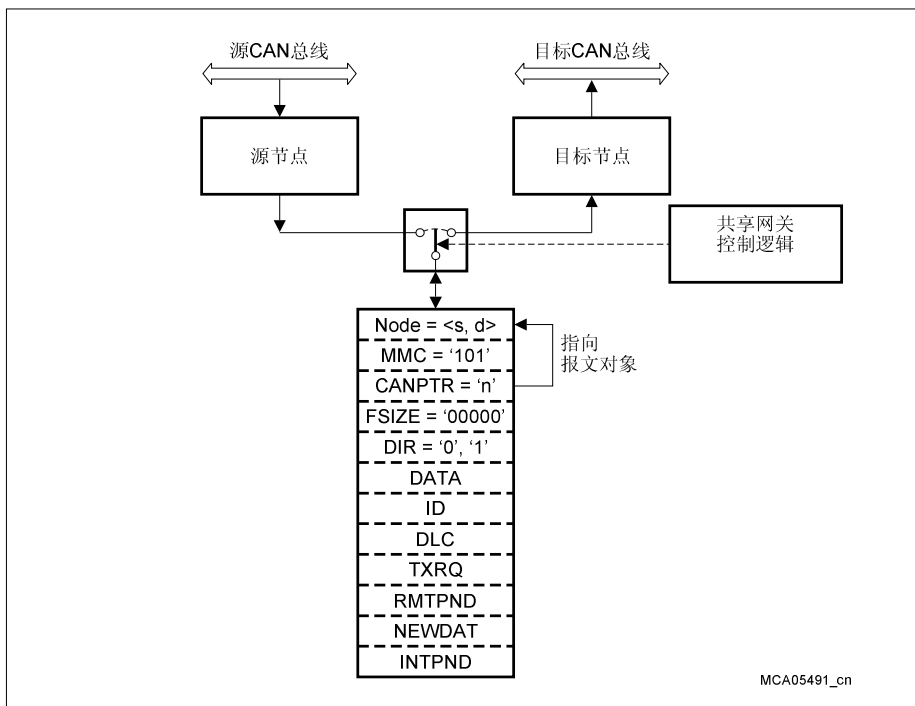


图 21-21 共享网关模式规则

将对应的 MSGFGCRn 寄存器的位 MMC 设置为 ‘101’，则每个报文对象都可用作一个共享网关。当报文配置位 NODE 被清零，CAN 节点 A 用做源节点，传送数据帧至目标节点 B。如果报文配置位 NODE 置 1，CAN 节点 B 作为数据源。利用第二个报文对象，可实现双向网关，共享网关模式下，第二个报文对象的 NODE 设置与第一个报文对象的 NODE 设置相反。位域 CANPTR 必须初始化为共享网关报文对象编号，FSIZE、IDC 和 DLCC 都应当被清零。接收到带匹配标识符的数据帧时，控制寄存器 MSGFGCRn 中的位 GDFS 决定是否自动置位 TXRQ（GDFS= ‘1’）。由位 SRREN 决定，目标方接收到的远程帧是通过网关传送给源节点，还是直接由目标方产生的数据帧应答。

优化的共享网关功能支持下列各种应用场景：

- 和 CAN 节点 A 连接的数据源，该数据源通过 CAN 节点 B 在目标方 CAN 总线上自动连续发送数据帧。
 相应的传送状态转换为：1-2-...
- 和 CAN 节点 A 相连的数据源，在接收到一个来自目标方 CAN 总线上匹配的远程帧后，该数据源通过 CAN 节点 B 连续发送数据帧。
 相应的传送状态转换为：7-4-2-...
- 和 CAN 节点 A 相连的数据源，CAN 节点 B 接收到一个匹配远程帧的操作触发数据帧的发送。相关的数据帧应当由 CAN 节点 B 再次发送在目标方 CAN 总线上。
 相应的发送状态转换为 5-6-1-3-...

根据具体应用，通过适当设置 NODE、DIR、GDFS 和 SRREN，共享网关报文对象可被初始化为源方的接收报文对象或者目标方的发送报文对象。不同的传送状态转换见图 21-22。

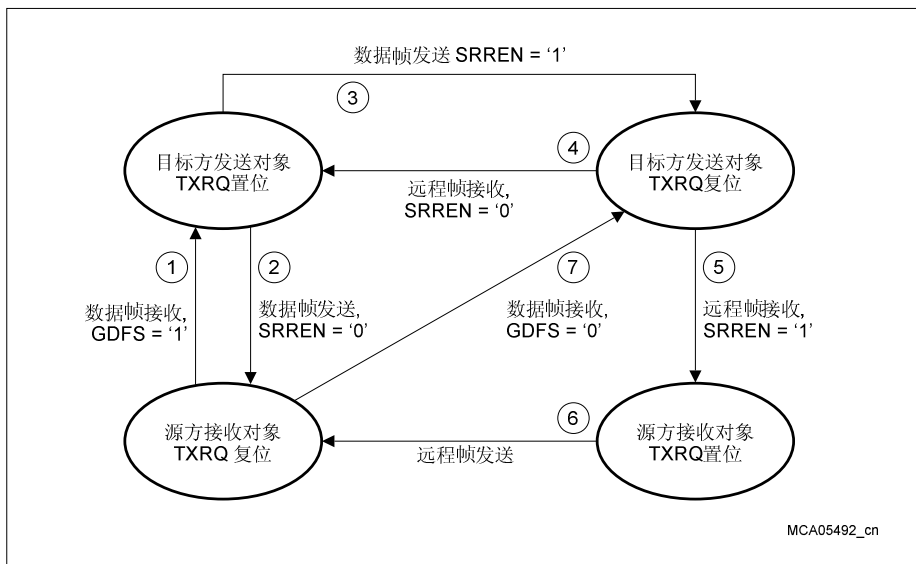


图 21-22 共享网关模式下，传送状态图

当共享网关报文对象被设置为源方接收报文对象（图 21-22 左下方状态圈内），当 **GDFS** 置为 ‘1’ 时接收一个数据帧，通过取反控制位 **NODE** 和 **DIR**，该报文对象被切换为目标方发送报文对象，并发出相应的数据帧而无需 **CPU** 干预（图中左上方状态圈）。

控制位 **SRREN**，决定共享网关报文对象是返回至其最初的功能，作为源方的接收对象（**SRREN**= ‘0’：状态转换 2 至左下方状态圈内，见图 21-22），还是仍分配给目标方等待带匹配标识符的远程帧（**SRREN**= ‘1’：状态转换 3 到右下方的状态圈）。

当共享报文对象被分配给目标方，作为发送报文对象（右上方状态圈），它响应目标方接收到的远程帧。如果位 **SRREN** 被清零，直接根据网关报文对象的内容，产生数据帧来应答远程请求（状态转换 4 到左上方状态圈）。

如果位 **SRREN** 置位且在目标方接收到一个远程帧，通过取反控制位 **NODE** 和 **DIR**，共享的网关报文对象切换为一个源方的接收对象，同时通过设置 **TXRQ** 和 **RMTPND** 为 ‘10’，准备发送接收到的远程帧（状态转换 5 至右下方状态圈）。

然后，该共享的网关报文对象发送相应的远程帧而无需任何 **CPU** 干预（状态转换 6 到左下方的状态圈）。

网关报文对象仍分配给源方，直到接收到一个带匹配标识符的数据帧（图中左下方的状态圈）。然后，共享网关报文对象返回至目标方，且由控制位 **GDFS** 决定，是立即发送相应的数据帧（**GDFS**= ‘1’ 左下方状态图），还是等待 **CPU** 设置 **TXRQ** 为 ‘10’ 的动作（**GDFS**= ‘0’：状态转换 7 到右上方图）。或者另一种情况，带匹配标识符的远程帧到达目标方，可能置 **TXRQ** 为 ‘10’ 并启动数据帧的发送。

带有匹配标识符的共享网关对象切换到了目标方，如果一个数据帧到达源方，那么源方上的这个数据帧会丢失。因为被暂时分配给了目标节点，共享网关报文对象忽略源节点上的该数据帧，也不能通过控制位域 **MSGLST**= ‘10’ 来报告数据丢失。如果通过设置 **GDFS**= ‘0’，禁止目标方的自动数据发送，数据丢失的概率会增加。对于在目标总线上接收的远程帧，必须考虑会发生相应情况。

*注：只要位域 **MSGLST** 有效，到达的数据帧不能自动在目标方发送。因为控制位 **DIR** 的内部翻转，共享网关对象从接收转为发送操作，位域 **MSGLST** 则被解释为 **CPUUPD**= ‘10’，它阻止数据帧的自动传送。*

共享网关模式下，传送状态改变对报文对象中各位域的影响：

表 21-3 共享网关状态转换（第一部分，共两部分）

位域	转换 1： 接收数据帧， GDFS= ‘1’	转换 2： 发送数据帧， SRREN= ‘0’	转换 3： 发送数据帧， SRREN= ‘1’	转换 4： 接收远程帧， SRREN= ‘0’
节点	切换至<d>	切换至<s>	不变	不变
DIR	置位	复位	不变	不变
DATA	接收	不变	不变	不变
标识符	接收	不变	不变	如果 RMM= ‘1’ 接收
DLC	接收	不变	不变	如果 RMM= ‘1’ 接收
TXRQ	置位	复位	复位	置位
RMPND	复位	复位	复位	置位
NEWDAT	置位	复位	复位	复位
INTPND	如果 RXIE= ‘10’，置位	如果 TXIE= ‘10’，置位	如果 TXIE= ‘10’，置位	如果 RXIE= ‘10’，置位

表 21-4 共享网关状态转换（第二部分）

位域	转换 5 接收远程帧， SRREN= ‘1’	转换 6 发送远程帧	转换 7 接收数据帧 GDFS= ‘0’
节点	切换到<s>	不变	切换到<d>
DIR	复位	不变	置位
DATA	不变	不变	接收
标识符	如果 RMM= ‘1’ 接收	不变	接收
DLC	如果 RMM= ‘1’ 接收	不变	接收
TXRQ	置位	复位	复位
RMTPND	复位	复位	复位
NEWDAT	不变	不变	置位
INTPND	如果 RXIE= ‘10’ ， 置位	如果 TXIE= ‘10’ ， 置位	如果 RXIE= ‘10’ ， 置位

21.1.7 TwinCAN 模块编程

必须通过设置特定 CAN 节点的控制寄存器 ACR/BCR 位 INIT 为‘1’，执行软件初始化。当 INIT 置位时，禁止 CAN 控制器和 CAN 总线之间的所有报文传送。

初始化程序必须执行下列任务：

- 相应节点的设置
- 每个相关报文对象的初始化

21.1.7.1 CAN 节点 A/B 设置

每个 CAN 节点都可以通过编程相关的寄存器单独设置。根据 ACR/BCR 控制寄存器的内容，激活正常工作模式或者 CAN 分析模式。另外，可使能或禁止各类中断（状态改变、错误、最近错误等）。

编程 ABTR/BBTR 寄存器来定义位定时。按照连接到相应 CAN 节点的 CAN 总线段的特性，将预分频值，同步跳变宽度和时间段，安排在采样点之前和之后。

全局中断节点指针寄存器（AGINP/BGINP）控制中断请求源的复用器，将中断请求源（错误、最近错误、全局发送/接收和帧计数器溢出中断请求）和 8 条公共的节点中的一个相连。INTID 屏蔽寄存器（AIMR0/4 和 BIMR0/4）的内容决定将那些中断源由 AIR/BIR 中断挂起寄存器报告出来。

21.1.7.2 报文对象的初始化

报文存储空间包含 32 个报文对象，由两个 CAN 节点共享。每个报文对象必须设置与其相关的目标节点及操作属性。总是通过 MSGVAL=‘01’禁止报文对象之后，才能开始该报文对象属性的初始化。

和报文相关的 CAN 节点，是通过 MSGCFGn 寄存器中的位 NODE 定义的。报文对象可以定义为网关，从 CAN 节点 A 到节点 B 传送信息，反之亦然。在这种情况下，必须编程 FIFO/网关控制寄存器 MSGFGCRn，指定网关模式（位域 MMC），目标中断节点和信息移交的更详细的情况。

在寄存器 MSGARn 中建立和报文相关的标识符。寄存器 MSGCFGn 位 XTD 用来指示，是用 29 位扩展标识符还是用 11 位标准标识符，应当根据情况进行相应地设置。根据寄存器 MSGAMRn 中的内容对接收到的报文进行屏蔽滤波。

报文中断处理可单独设置为发送和接收方向。特定方向中断由寄存器 MSGCTRn 的位 TXIE 和 RXIE 使能，而由寄存器 MSGCFGn 中的位域 TXINP 和 RXINP 决定目标中断节点的选择。

报文对象提供 FIFO 缓存区。缓存区容量由 FIFO/网关控制寄存器 MSGFGCRn 的位域 FSIZE 决定。

对于发送报文对象来说，在相应的数据部分被初始化之前，可能已经通过设置 MSGVAL 为‘10’完成报文对象的特性分配。如果 CPUUPD 设置为‘10’，接收到的带匹配标识符的远程帧通过内部置位 TXRQ 被保存下来，但是不立即用一个相应的数据用户手册

帧应答该远程帧。只要 CPUUPD 保持为 ‘10’，保存在寄存器 MSGDRn0 / MSGDRn4 中报文数据可以一直被更新。一旦 CPUUPD 复位至 ‘01’，相关的 CAN 节点控制器就会发送出相应的数据帧。

21.1.7.3 报文传送控制

图 21-23 为一个发送报文对象的处理过程。报文对象特性的初始化过程总是由 MSGVAL= ‘01’ 禁止报文对象开始的。复位一些控制标志（INTPND, RMT PND, TXRQ 和 NEWDAT）之后，定义传送方向和标识符。通过设置 MSGVAL 为 ‘10’ 来完成报文对象的初始化。

为了更新发送报文数据部分，应该先设置 CPUUPD 为 ‘10’，随后对寄存器 MSGDRn0/MSGDRn4 进行写访问。必须由 CPU 通过设置 NEWDAT 为 ‘10’ 来指示数据部分的更新。然后，如果请求自动报文处理，位域 CPUUPD 必须复位为 ‘01’。在这种情况下，寄存器 MSGCTRn 标志 TXRQ 被软件置为 ‘10’，或者因接收到带匹配标识符的远程帧被对应的 CAN 节点硬件设置为 ‘10’，启动数据发送。如果 CPUUPD 保持置位，CPU 必须通过设置 TXRQ 为 ‘10’ 同时复位 CPUUPD 来启动数据发送。如果在报文对象数据存储更新期间，收到一个标识符通过验收的远程帧，位 TXRQ 和 RMT PND 自动置为 ‘10’ 且当 CPUUPD 被再次复位时，由 CAN 控制器自动开始相应的数据帧发送。

图 21-24 为接收报文对象的处理。如上所述，先通过 MSGVAL 禁止报文对象，进行报文对象特性初始化之后，再通过 MSGVAL 使能报文对象。置位 MSGVAL 为 ‘10’ 之后，通过设置 TXRQ= ‘10’，由 CPU 发起远程帧的发送。数据帧的接收由相应的 CAN 节点控制通过 NEWDAT= ‘10’ 来指示。通过 CPU 复位 NEWDAT 为 ‘01’，来启动接收的数据帧（存储在寄存器 MSGDRn0/MSGDRn4 中）的处理过程。扫描标志 MSGLST 之后，指示出先前报文的丢失，应当将接收到的信息复制到一个应用数据缓冲区，为新数据帧释放该报文对象。最后，应当再次检查 NEWDAT 以确保，该过程是在一套一致的数据之上而不是部分新报文和部分旧报文之上进行的。

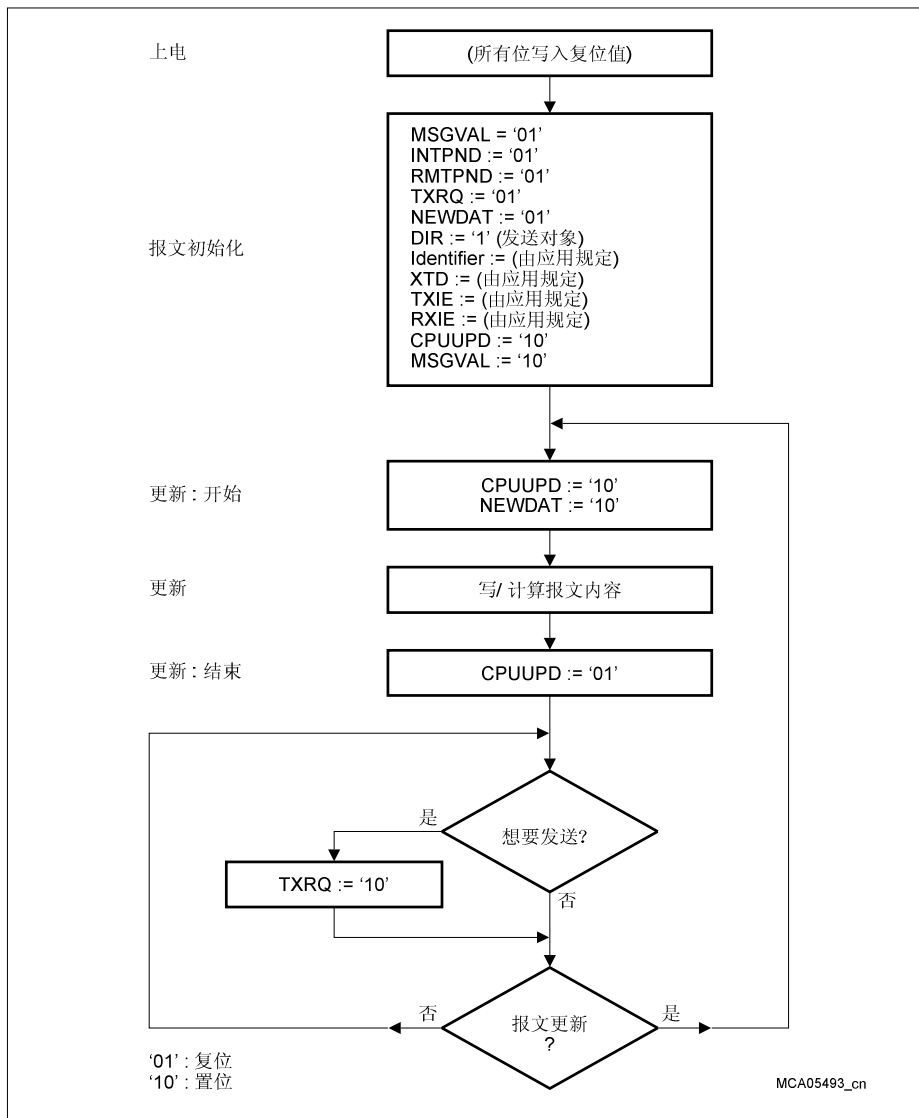


图 21-23 CPU 对发送报文对象的处理

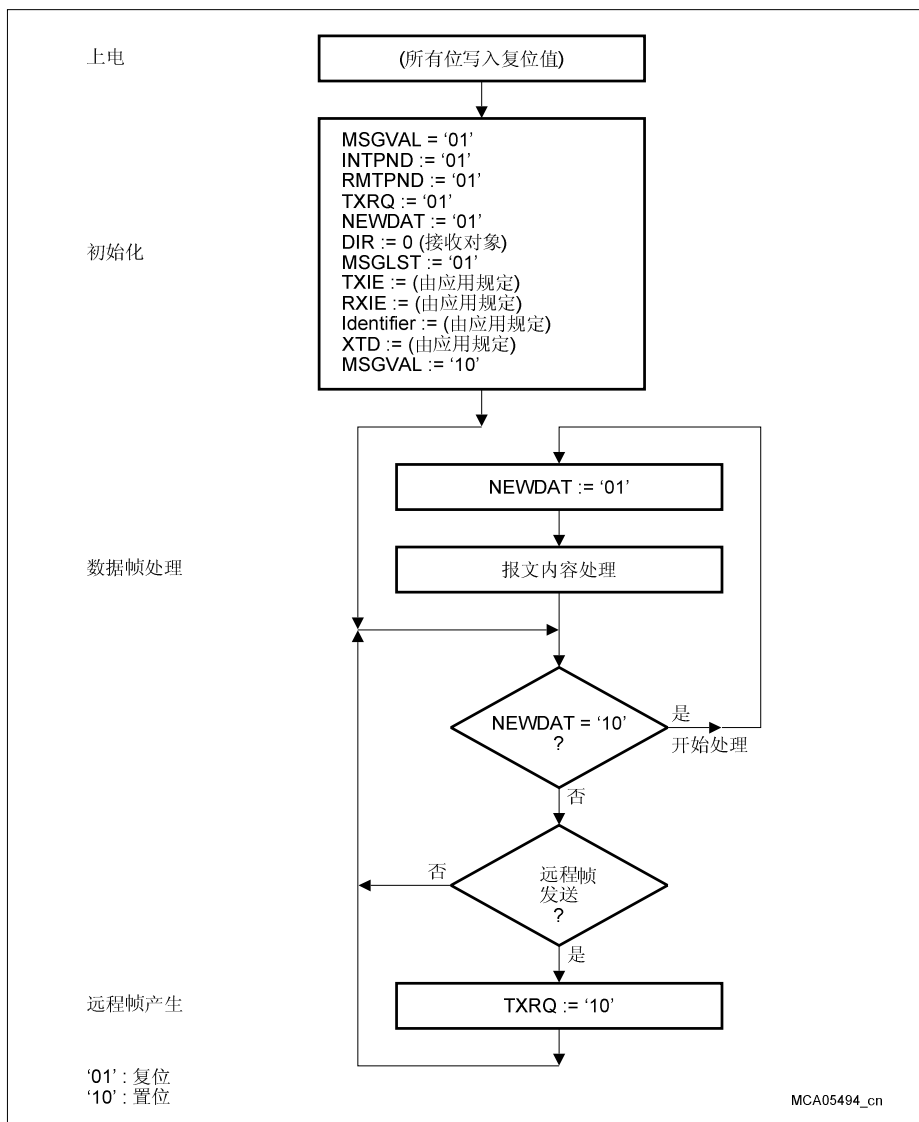


图 21-24 CPU 对接收报文对象的处理

21.1.8 回环模式

TwinCAN 模块的回环模式为 TwinCAN 模块和 CAN 驱动软件的内部测试提供工具。无需和 CAN 总线系统相连，就可以开发和测试 CAN 驱动软件。

在回环模式，发送引脚给收发器发出隐性信号。将发送信号组合在一起并与内部接收信号连接，如图 21-25 所示。回环模式之内不考虑接收输入引脚。

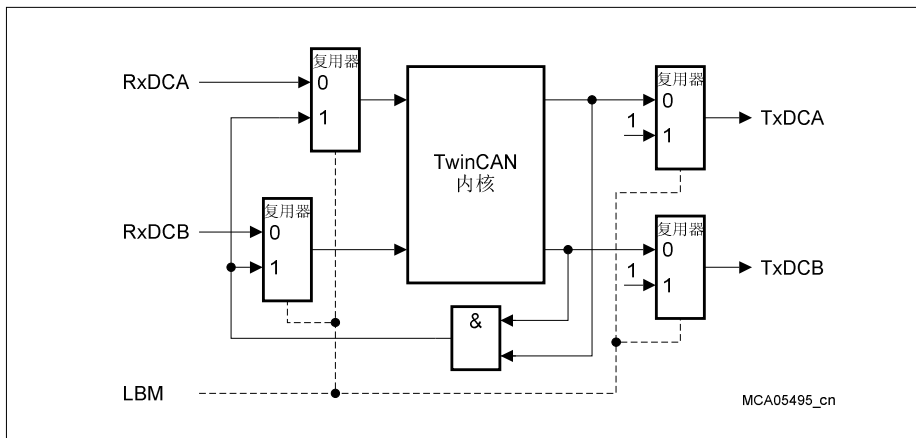


图 21-25 回环模式

根据表 21-5，节点 A 和节点 B 中位定时寄存器的位 LBM 控制回环模式。

表 21-5 回环模式

ABTR.LBM	BBTR.LBM	描述
0	0	禁止回环模式
0	1	禁止回环模式
1	0	禁止回环模式
1	1	使能回环模式

21.1.9 单次发送试验功能

寄存器 MSGFGCRn 中的位 STT 单独地控制每个报文对象的单次发送试验功能。如果使能单次发送试验功能，在该报文对象已经开始相关一帧发送之后，立即复位发送请求标志 TXRQ。因此，即使被错误帧扰乱，发送帧也仅在 CAN 总线上传送一次。

注：报文对象必须由位域 MSGVAL 设置为有效，以确保相关帧的发送。

21.1.10 模块时钟要求

TwinCAN 模块可被设置成许多功能。对于给定的功能，为了工作在特定的波特率上，需要某个最小的模块时钟频率。**表 21-6** 列出了某些设置的例子。这些例子中包含了 CPU 以最快速度连续地访问 TwinCAN 模块的最坏情况。

如果在 CAN 总线上不传送无数据的帧（DLC=0 的数据帧或远程帧），可以降低模块时钟频率（见**表 21-6** 最后一列）。因为传送数据部分的同时，可执行内部操作，所以可能存在这种情况。

表 21-6 对应于 1Mbit/s，所需最小的模块时钟

	1 节点有效 DLC≥0	2 节点有效 DLC≥0	2 节点有效 DLC≥1
FIFO/网关使能	21MHz	36MHz	32MHz
无 FIFO/网关	20MHz	29MHz	26MHz

注：给出的数值为最大 CAN 总线速度 1 Mbit/s 所需要的值。对于低一点的比特率，可线性地降低所需的最小模块频率，也就是，对于 500kbit/s，需要的频率减半。

然而，如果两个节点工作在不同的比特率，必须根据最快的节点选择模块时钟频率。

21.2 TwinCAN 寄存器描述

21.2.1 寄存器映射

图 21-26 给出了所有和 TwinCAN 模块内核相关的寄存器。

CAN节点A 寄存器	CAN节点B 寄存器	CAN报文对象 寄存器 ¹⁾	全局CAN 控制 / 状态 寄存器
ACR	BCR	MSGDRn0	RXIPND
ASR	BSR	MSGDRn4	TXIPND
AIR	BIR	MSGARn	
ABTR	BBTR	MSGAMRn	
AGINP	BGINP	MSGCTRn	
AFCR	BFCR	MSGCFGn	
AIMR0	BIMR0	MSGFGCRn	
AIMR4	BIMR4		
AECNT	BECNT		

ACR	节点A控制寄存器	BCR	节点B控制寄存器
ASR	节点A状态寄存器	BSR	节点B状态寄存器
AIR	节点A中断挂起寄存器	BIR	节点B中断挂起寄存器
ABTR	节点A位定时寄存器	BBTR	节点B位定时寄存器
AGINP	节点A全局中断节点指针寄存器	BGINP	节点B全局中断节点指针寄存器
AFCR	节点A帧计数器寄存器	BFCR	节点B帧计数器寄存器
AIMR0	节点A INTID屏蔽寄存器0	BIMR0	节点B INTID屏蔽寄存器0
AIMR4	节点A INTID屏蔽寄存器4	BIMR4	节点B INTID屏蔽寄存器4
AECNT	节点A错误计数器寄存器	BECNT	节点B错误计数器寄存器

MSGDRn0	报文对象n数据寄存器 0	MSGDRn4	报文对象n数据寄存器 4
MSGARn	报文对象n仲裁寄存器	MSGAMRn	报文对象n验收屏蔽寄存器
MSGCTRn	报文对象n控制寄存器	MSGCFGn	报文对象n设置寄存器
MSGFGCRn	报文对象n FIFO/网关控制寄存器		

RXIPND	接收中断挂起寄存器	TXIPND	发送中断挂起寄存器
--------	-----------	--------	-----------

1) ‘n’ 给出报文对象的编号: n = 0 ... 31.

MCA05496_cn

图 21-26 TwinCAN 内核相关的所有寄存器

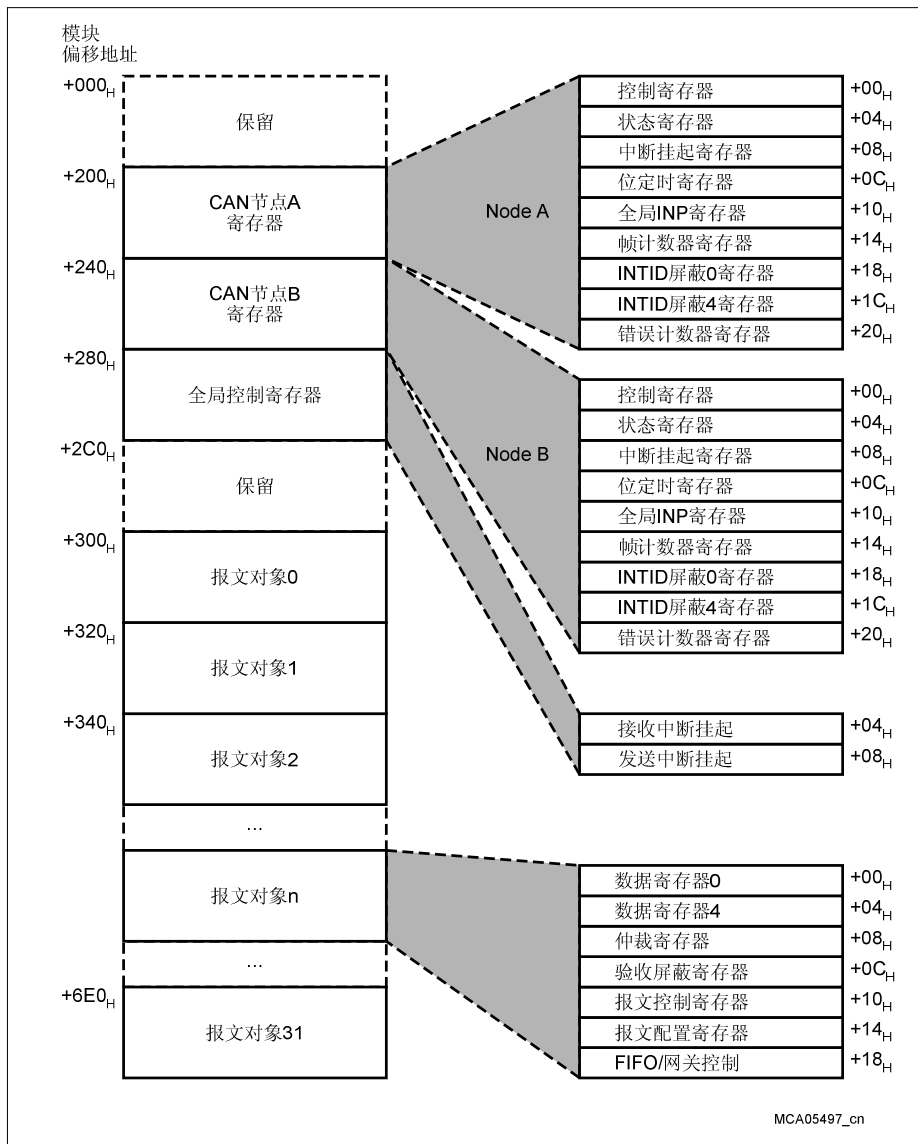


图 21-27 TwinCAN 内核地址映射

21.2.2 CAN 节点 A/B 寄存器

节点控制寄存器用来控制初始化、定义与节点相关的中断处理并选择工作模式。

ACR

节点 A 控制寄存器

复位值: 0001_H

BCR

节点 B 控制寄存器

复位值: 0001_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0								CAL M	CCE	0	LEC IE	EIE	SIE	0	INIT
r								rw	rw	r	rw	rw	rw	r	rwh

符号	位序号	读写类型	功能描述
INIT	0	rwh	初始化 0 复位 INIT 启动与 CAN 总线的同步操作。在同步过程 ¹⁾ 之后，节点参与 CAN 通信。 1 置位 INIT 之后，CAN 节点停止所有的 CAN 总线活动，并可在不影响实际 CAN 总线通信的情况下，将所有的寄存器初始化。当进入总线关闭状态时，INIT 被自动置位。
SIE	2	rw	状态改变中断使能 当报文传送成功（由状态寄存器 ASR 或 BSR 中的标志 TXOK 或 RXOK 指示）结束时，出现状态改变中断。 0 禁止状态改变中断 1 使能状态改变中断
EIE	3	rw	错误中断使能 状态寄存器 ASR 或 BSR 中位 BOFF 或位 EWRN 的改变引发错误中断的产生。 0 禁止错误中断 1 使能错误中断
LECIE	4	rw	最近错误码中断使能

符号	位序号	读写类型	功能描述
			当状态寄存器 ASR 或 BSR 中的位域 LCE 中错误码被置位时，产生一个最近错误码中断。 0 禁止最近错误码中断 1 使能最近错误码中断
CCE	6	rw	设置改变使能 0 对位定时寄存器的访问和错误计数器的修改操作都被禁止 1 对位定时寄存器的访问和错误计数器的修改操作被使能
CALM	7	rw	CAN 分析模式 位 CALM 定义相应节点的报文对象是否工作在分析模式下。 0 CAN 报文对象采用 CAN 协议 1 选择 CAN 分析模式
0	1, 5 [15:8]	r	保留； 读操作返回 0，应写入 0。

1) 在不处于总线关闭状态的情况下（例如，上电之后），软件复位 INIT 之后，在 CAN 模块参与 CAN 通信之前，必须在总线上监测到一个带 11 个连续隐性位（11× ‘1’）的序列。

在总线关闭恢复过程中，必须监测到 128 个带 11 个位连续隐性位（11× ‘1’）的序列。在进入总线关闭状态之后，立即由硬件开始监测隐性位序列。已经检测到的 11× ‘1’ 序列数目由接收错误计数器给出。

总线关闭恢复序列的结束时，由硬件测试位 INIT。如果 INIT 仍置位，受其影响的 CAN 节点控制器等待，直到 INIT 被清零，且在该 CAN 节点再次参与 CAN 通信之前，需在 CAN 总线上检测到 11 个连续的隐性位（11× ‘1’）。如果 INIT 已经被清零，即使能受影响的 CAN 节点控制器和相关的 CAN 总线之间的报文传送。

节点状态寄存器报告错误状态和数据发送的成功结束。为了解除状态改变中断请求，必须读取该寄存器。

ASR

节点 A 状态寄存器

复位值: 0000_H

BSR

节点 B 状态寄存器

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0								B OFF	E WRN	0	RX OK	TX OK	LEC		
r								rh	rh	r	rwh	rwh	rwh		

符号	位序号	读写类型	功能描述
LEC	[2:0]	rwh	最近错误码 位域 LEC 用来指示最新的 CAN 报文传送是否正确（无错误）或者指示检测到的错误类型。错误条件的详细信息见表 21-7。 000 无错误 001 填充错误 010 格式错误 011 应答错误 100 位 1 错误 101 位 0 错误 110 CRC 错误 111 保留
TXOK	3	rwh	报文发送成功 0 自从上一次标志置位，无成功的发送 1 一个报文已经被成功发送（无错误且至少被一个其它 CAN 节点应答） TXOK 必须由软件复位。
RXOK	4	rwh	报文接收成功 0 自从上一次标志复位，无成功的接收 1 一个报文已经被成功接收

符号	位序号	读写类型	功能描述
			RXOK 必须由软件复位。
EWRN	6	rh	错误警告状态 0 没有超过警告界限 1 错误管理逻辑中的一个错误计数器达到错误警告界限 96
BOFF	7	rh	总线关闭状态 0 CAN 控制器不处于总线关闭状态 1 CAN 控制器处于总线关闭状态
0	5, [15:8]	r	保留； 读操作返回 0，应写入 0。

表 21-7 LEC 位域的含义

LEC 错误	描述
无错误	CAN 总线上的最新传送已经成功完成。
填充错误	在接收到的报文中的不允许出现多于 5 个相同位的部分，出现了此种情况。
格式错误	接收帧的固定格式部分格式错误。
应答错误	发送的报文未被另一个节点应答。
位 1 错误	在报文发送期间，CAN 节点要发送隐性电平（‘1’），但是监测到的总线值是显性的（在总截域和应答时隙之外）。
位 0 错误	用该错误编码表示两种情况： <ol style="list-style-type: none"> 在报文发送期间（或者应答位，激活错误标志，过载标志），CAN 节点要发送一个显性位（‘0’），但是监测到的总线值为隐性 在总线关闭恢复期间，每次监测到 11 个隐性位的序列，置位该编码。CPU 可以用该位指示总线不再被连续干扰
CRC 错误	接收到的报文的 CRC 校验和不正确。

中断挂起寄存器包含挂起的优先级最高的中断请求的 ID 号。

AIR

节点 A 中断挂起寄存器

复位值: 0000_H

BIR

节点 B 中断挂起寄存器

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0								INTID							
r								rwh							

符号	位序号	读写类型	功能描述
INTID	[7:0]	rwh	中断标识符 00 _H 无挂起的中断 01 _H LEC、EI、TXOK 或 RXOK 中断挂起 02 _H 报文对象 0 的 RX 或 TX 中断挂起 03 _H 报文对象 1 的 RX 或 TX 中断挂起 21 _H 报文对象 31 的 RX 或 TX 中断挂起 位域 INTID 可被软件写入，在软件操作之后开始更新，并检查这些变化。
0	[15:8]	r	保留； 读操作返回 0。

寄存器 **AECNT/BECNT** 包含接收错误计数器和发送错误计数器的值。一些附加的状态/控制位便于进行错误分析。

AECNTH

节点 A 错误计数器寄存器高位 复位值: 0060_H

AECNTL

节点 A 错误计数器寄存器低位 复位值: 0000_H

BECNTH

节点 B 错误计数器寄存器高位 复位值: 0060_H

BECNTL

节点 B 错误计数器寄存器低位 复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0						LE INC	LE TD	EWRNLVL							
r						rh		rh	rw						

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEC								REC							
rwh								rwh							

符号	位序号	读写类型	功能描述
REC	[7:0] 低位	rwh	接收错误计数器 位域 REC 包含相应节点接收错误计数器值。
TEC	[15:8] 低位	rwh	发送错误计数器 位域 TEC 包含相应节点发送错误计数器值。
EWRNLVL	[7:0] 高位	rw	错误警告级别 位域 EWRNLVL 定义门限值（警告级别，缺省值 60 _H = 96 _D ），达到该门限值，才能置位相应的错误警告位 EWRN。
LETD	8 高位	rh	最近错误的传送方向 0 当相应的节点接收报文时（REC 已增

符号	位序号	读写类型	功能描述
			加），出现了最近错误 1 当相应的节点发送报文时（TEC 已增加），出现了最近错误 不考虑验收滤波的结果，仅指示报文传送期间的错误。
LEINC	9 高位	rh	最近错误计数值增量 0 由 LETD 报告的错误，错误计数器加 1 1 由 LETD 报告的错误，错误计数器加 8
0	[15:10] 高位	r	保留 读操作返回 0，应写入 0。

注：要求寄存器 *ACR/BCR* 中的位 *CCE*= ‘1’，才能修改寄存器 *AECNT/BECNT* 的内容。

位定时寄存器包含调整数据传送波特率和位定时的所有参数。

ABTRH

节点 A 位定时寄存器高位

复位值: 0000_H

ABTRL

节点 A 位定时寄存器低位

复位值: 0000_H

BBTRH

节点 B 位定时寄存器高位

复位值: 0000_H

BBTRL

节点 B 位定时寄存器低位

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0															LBM
r															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIV 8X	TSEG2				TSEG1				SJW	BRP					
rw	rw				rw				rw	rw					

符号	位序号	读写类型	功能描述
BRP	[5:0] 低位	rw	波特率预分频 一个位时间基本单元对应于外部振荡器时钟周期乘以（BRP+1）所得值，还由位 DIV8X 决定。
SJW	[7:6] 低位	rw	（再）同步跳转宽度 允许用（SJW+1）个时间基本单元进行再同步操作。
TSEG1	[11:8] 低位	rw	采样点之前的时间段 考虑到信号传播延时和发送节点与接收节点时钟相位之间失配的补偿，采样点之前有（TSEG1+1）个时间基本单元。 TESG1 的有效值为：2...15。
TSEG2	[14:12]	rw	采样点之后的时间段

符号	位序号	读写类型	功能描述
	低位		考虑到用户定义的延时和发送节点与接收节点时钟相位之间失配的补偿，采样点之后有（TSEG2+1）个时间基本单元。 TESG2 的有效值为：1...7。
DIV8X	15 低位	rw	模块时钟 f_{CAN} 8 分频 0 波特率预分频器直接由 f _{CAN} 驱动 1 波特率预分频器直接由 f _{CAN} /8 驱动
LBM	0 高位	rw	回环模式 0 禁止回环模式 1 如果节点 A 和节点 B 的 BTR 寄存器中的位 LBM 置位，回环模式被使能
0	[15:1] 高位	r	保留 读操作返回 0，应写入 0。

注：寄存器 ACR/BCR 寄存器中的位 CCE= ‘1’，才能修改寄存器 ABTR/BBTR 中的内容。

帧计数器寄存器控制帧计数器并提供状态信息。

AFCRH

节点 A 帧计数器寄存器高位 复位值: 0000_H

AFCRL

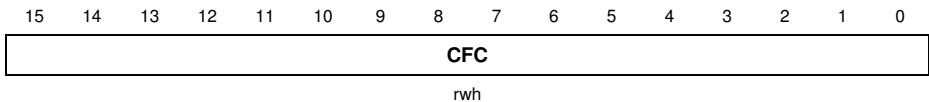
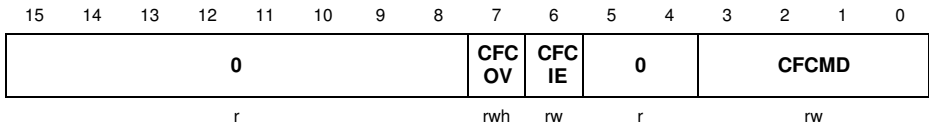
节点 A 帧计数器寄存器低位 复位值: 0000_H

BFCRH

节点 A 帧计数器寄存器高位 复位值: 0000_H

BFCRL

节点 B 帧计数器寄存器低位 复位值: 0000_H



符号	位序号	读写类型	功能描述
CFC	[15:0] 低位	rwh	CAN 帧计数器 该位域包含帧计数器的计数值。 在正确完成一个报文传送后，CFC 的值（在 SOF 位期间捕获到的值）被复制到相应的报文对象控制寄存器 MSGCTRn 的位域 CFCVAL 中。
CFCMD	[3:0] 高位	rw	帧计数模式 该位域定义帧计数器的工作模式。该计数器可工作在帧基准（帧计数）或时间基准（时间戳）上。 0XXX _B 帧计数： ¹⁾ 0XX0 _B CAN 总线上传送一个外来帧之后，CFC 不增加 0XX1 _B 每一次 CAN 总线上正确传送一个

符号	位序号	读写类型	功能描述
			外来帧之后，CFC 增加 0X0X _B 相应的 CAN 节点接收一帧之后，CFC 不增加 0X1X _B 每一次该节点正确接收到一帧之后，CFC 增加 00XX _B 该节点发送一帧之后，CFC 不增加 01XX _B 每次该节点正确发送一帧之后，CFC 增加 1XXX _B 时间戳： 1000 _B 新的位时间开始时，CFC 增加。 该值在 SOF 位期间被采样 1001 _B 新的位时间开始时，CFC 增加。 该值在 EOF 的最后一位期间被采样 其它 保留
CFCIE	6 高位	rw	CAN 帧计数中断使能 置位 CFCIE 使能 CAN 帧计数器溢出。 （CFCOV）中断请求。 0 禁止 CFCOV 中断 1 使能 CFCOV 中断
CFCOV	7 高位	rwh	CAN 帧计数溢出标志 CFC 溢出时（FFFF _H 到 0000 _H ）置位标志 CFCOV。如果相应的中断被使能（CFCIE = ‘1’），产生一个中断请求。 0 还未检测到溢出 1 自从该位上次被复位后，已检测到溢出 必须由软件清零 CFCOV。
0	[5:4] [15:8] 高位	r	保留 读操作返回 0，应写入 0。

1) 如果已经选择了帧计数器功能（CFCMD.3= ‘0’），位 CFCMD.0 使能或禁止外来帧的计数。总线上的不是被该节点自身发送或接收的正确帧被称为外来帧。位 CFCMD.1 使能或禁止已经被相应的 CAN 节点正确接收的帧的计数。位 CFCMD.2 使能或禁止已经由相应的 CAN 节点正确发送的帧计数。

全局中断节点指针寄存器将每个全局中断请求源和 8 个可用的 CAN 中断节点之一相连。

AGINP

节点 A 全局中断节点指针寄存器

复位值: 0000_H

BGINP

节点 B 全局中断节点指针寄存器

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	CFCINP			0	TRINP			0	LECINP			0	EINP		
r	rw			r	rw			r	rw			r	rw		

符号	位序号	读写类型	功能描述
EINP	[2:0]	rw	错误中断节点指针 若 EIE = ‘1’ 使能，报告 “错误中断请求” 的中断节点编号。 000 _B 选择 CAN 中断节点 0 111 _B 选择 CAN 中断节点 7
LECINP	[6:4]	rw	最近错误码中断节点指针 若 LECIE = ‘1’ 使能，报告最近错误中断请求的中断节点编号。 000 _B 选择 CAN 中断节点 0 111 _B 选择 CAN 中断节点 7
TRINP	[10:8]	rw	发送/接收 OK 中断节点指针 若 SIE = ‘1’ 使能，报告发送和接收中断请求的中断节点编号。 000 _B 选择 CAN 中断节点 0 111 _B 选择 CAN 中断节点 7
CFCINP	[14:12]	rw	帧计数器中断节点指针 若 CFCIE = ‘1’ 使能，报告帧计数器溢出中断请求的中断节点编号。

符号	位序号	读写类型	功能描述
			000 _B 选择 CAN 中断节点 0 111 _B 选择 CAN 中断节点 7
0	3, 7 11, 15	r	保留； 读操作返回 0，应写入 0。

中断标识屏蔽寄存器用于禁止寄存器 AIR/BIR 中挂起的中断请求的 ID 通知。中断屏蔽寄存器 AIMR0/BIMR0 用于使能某个报文特定中断源（正确发送/接收），和 INTID 值的产生有关。

AIMRH0

节点 A INTID 屏蔽寄存器 0 高位 复位值: 0000_H

AIMRL0

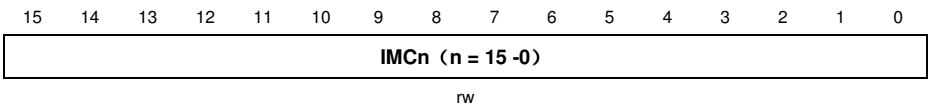
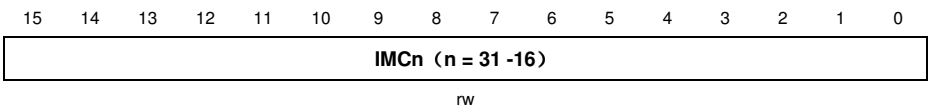
节点 A INTID 屏蔽寄存器 0 低位 复位值: 0000_H

BIMRH0

节点 B INTID 屏蔽寄存器 0 高位 复位值: 0000_H

BIMRL0

节点 B INTID 屏蔽寄存器 0 低位 复位值: 0000_H



符号	位序号	读写类型	功能描述
IMCn (n = 15- 0)	低位	rw	报文对象 n 的 INTID 屏蔽控制
IMCn (n = 31- 16)	高位		0 报文对象 n 和 INTID 值的产生无关 1 报文对象 n 的中断挂起状态和 INTID 值的产生有关

中断屏蔽寄存器 AIMR4/BIMR4 用于确定节点特定的中断源（最近错误，正确接收，错误警告/总线关闭）是否与 INTID 值的产生有关。

AIMR4

节点 A INTID 屏蔽寄存器 4

复位值: 0000H

BIMR4

节点 B INTID 屏蔽寄存器 4

复位值: 0000H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0													IMC 34	IMC 33	IMC 32
r													rw	rw	rw

符号	位序号	读写类型	功能描述
IMC32	0	rw	最近错误中断 INTID 屏蔽控制 0 最近错误中断源与 INTID 值的产生无关 1 最近错误中断源与 INTID 值的产生有关
IMC33	1	rw	TX/RX 中断 INTID 屏蔽控制 0 TX/RX 中断源与 INTID 值的产生无关 1 TX/RX 中断源与 INTID 值的产生有关
IMC34	2	rw	错误中断 INTID 屏蔽控制 0 错误中断源与 INTID 值的产生无关 1 错误中断挂起状态与 INTID 值的产生有关
0	[15:3]	r	保留 读操作返回 0，应写入 0。

21.2.3 CAN 报文对象寄存器

为每个报文对象都提供了一组控制和数据寄存器。相应的寄存器名都附加一个变量 n ，从 0 到 31（例如，MSGDR n 0 意味着数据寄存器 MSGDR300 分配给了报文对象 30）。

报文数据寄存器 0 包含报文对象 n 的数据字节 0 到数据字节 3。

MSGDRH n 0 ($n = 31-0$)

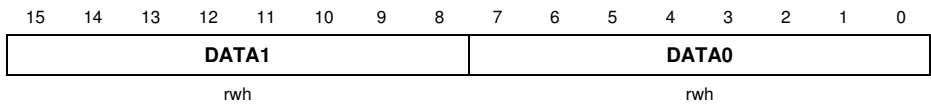
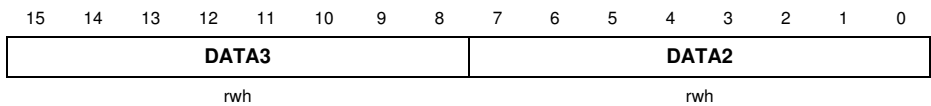
报文对象 n 数据寄存器 0 高位

复位值: 0000_H

MSGDRL n 0 ($n = 31-0$)

报文对象 n 数据寄存器 0 低位

复位值: 0000_H



符号	位序号	读写类型	功能描述
DATA0	[7:0] 低位	rwh	报文对象 n 数据字节 0
DATA1	[15:8] 低位	rwh	报文对象 n 数据字节 1
DATA2	[7:0] 高位	rwh	报文对象 n 数据字节 2
DATA3	[15:8] 高位	rwh	报文对象 n 数据字节 3

报文数据寄存器 4 包含报文对象 n 的数据字节 4 到数据字节 7。

MSGDRHn4 (n = 31-0)

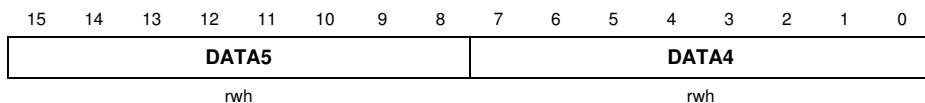
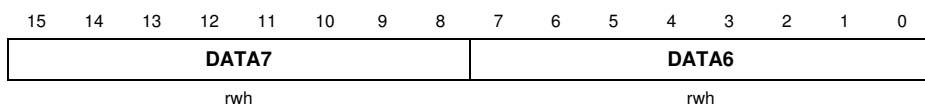
报文对象 n 数据寄存器 4 高位

复位值: 0000_H

MSGDRLn4 (n = 31-0)

报文对象 n 数据寄存器 4 低位

复位值: 0000_H



符号	位序号	读写类型	功能描述
DATA4	[7:0] 低位	rwh	报文对象 n 数据字节 4
DATA5	[15:8] 低位	rwh	报文对象 n 数据字节 5
DATA6	[7:0] 高位	rwh	报文对象 n 数据字节 6
DATA7	[15:8] 高位	rwh	报文对象 n 数据字节 7

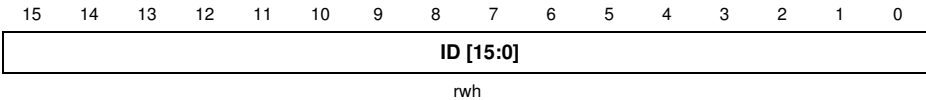
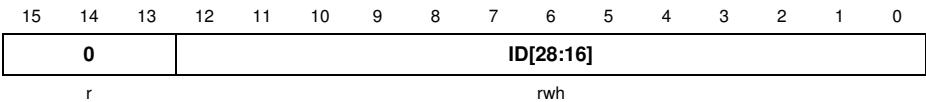
寄存器 MSGARn 包含报文对象 n 的标识符。

MSGARHn (n = 31- 0)

报文对象 n 仲裁寄存器高位 复位值: 0000_H

MSGARLn (n = 31- 0)

报文对象 n 仲裁寄存器低位 复位值: 0000_H



符号	位序号	读写类型	功能描述
ID[15:0]	[15:0]	rwh	报文标识符
ID[28:16]	低位 [12:0] 高位		标准报文标识符（ID[28:18]）或扩展报文标识符（ID[28:0]）。对于标准标识符不考虑 ID[17:0]。
0	[15:13] 高位	r	保留； 读操作返回 0，应写入 0。

寄存器 MSGAMRn 包含报文对象 n 的验收滤波屏蔽位。

MSGAMRHn (n = 31-0)

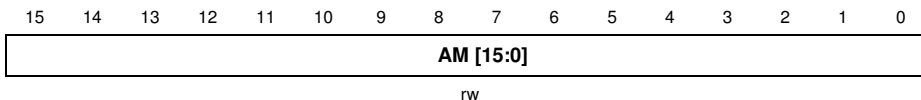
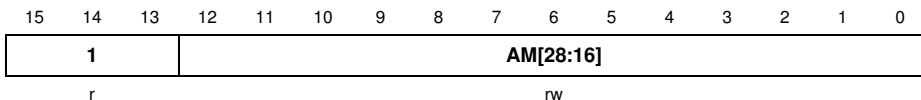
报文对象 n 验收屏蔽寄存器高位

复位值: FFFF_H

MSGAMRLn (n = 31-0)

报文对象 n 验收屏蔽寄存器低位

复位值: FFFF_H



符号	位序号	读写类型	功能描述
AM[15:0] AM[28:16]	[15:0] 低位 [12:0] 高位	rwh	报文验收屏蔽 用标准标识符（AM[28:18]）或扩展标识符（AM[28:0]）对输入的报文进行屏蔽滤波。 对于标准标识符位不考虑 AM[17:0]。 0 在验收测试时，不考虑该标识符位 1 在验收滤波时，考虑该标识符位
1	[15:13] 高位	r	保留； 读操作返回 1，应写入 1。

寄存器 MSGCTRHn 影响 CAN 节点控制器和相应报文对象 n 之间的数据传送，并为捕获的帧计数器值提供存储位域。

MSGCTRHn (n = 31-0)

报文对象 n 报文控制寄存器高位

复位值: 0000_H

MSGCTRLn (n = 31-0)

报文对象 n 报文控制寄存器低位

复位值: 5555_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CFCVAL															
rwh															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RMTPNB	TXRQ	MSGLST CPUUPD	NEWDAT	MSGVAL	TXIE	RXIE	INTPNB								
rwh	rwh	rwh	rwh	rwh	rw	rw	rwh								

符号	位序号	读写类型	功能描述
INTPNB	[1:0] 低位	rwh	报文对象中断挂起 INTPNB 由标志 RXIPNBn 和 TXIPNBn（若 TXIE 或 RXIE 使能）之间的“或”运算产生。INTPNB 必须由软件复位。 复位 INTPNB 也同时复位了相应的 RXIPNB 和 TXIPNB 标志。 01 无挂起的报文对象中断请求 10 报文对象已产生中断请求
RXIE	[3:2] 低位	rw	报文对象接收中断使能 01 禁止报文对象接收中断。 10 使能报文对象接收中断。如果 RXIE 被置位，在成功接收一帧之后，置位 INTPNB 和 RXIPNB。
TXIE	[5:4] 低位	rw	报文对象发送中断使能 01 禁止报文对象发送中断。 10 使能报文对象发送中断。如果 TXIE

符号	位序号	读写类型	功能描述
			被置位，在成功发送一帧之后，置位 INTPND 和 TXIPND。
MSGVAL ¹⁾	[7:6] 低位	rwh	报文对象有效 CAN 控制器仅对有效报文对象进行操作。 当报文对象被改变或者根本不使用该报文对象时，它们会被标记为无效。 01 报文对象无效 10 报文对象有效
NEWDAT ²⁾	[9:8] 低位	rwh	报文对象中新数据可用 01 报文对象中数据未更新 10 报文对象中的数据已经被更新
MSGLST	[11:10] 低位	rwh	报文丢失（仅用于接收） 01 报文对象数据没有丢失。 10 NEWDAT 仍然置位时，CAN 控制器已将新报文保存到该报文对象中，而先前保存的报文丢失。 MSGLST 必须由软件复位。
CPUUPD ³⁾	[11:10] 低位	rwh	CPU 更新（仅用于发送） 指示现在不能发送相应的报文对象。软件设置该位，禁止发送当前正由 CPU 更新的报文或控制对远程请求的自动应答。 01 可由 CAN 控制器自动发送报文对象中的数据 10 报文数据自动发送被禁止
TXRQ ⁴⁾	[13:12] 低位	rwh	报文对象发送请求标志 01 CPU 或远程帧未请求发送任何报文对象数据 10 CPU 或远程帧请求的报文对象数据发送被挂起 通过设置控制位 GDFS= '0'，由 CAN 节点控制器自动置位网关报文对象的 TXRQ 操作被禁止。

符号	位序号	读写类型	功能描述
			当报文已经成功发送时，TXRQ 自动复位。 如果存在几个有效的报文对象有挂起的发送请求，报文编号最低的报文对象将被首先发送。
RMTDND	[15:14] 低位	rwh	远程挂起标志 （用于发送对象） 01 无远程节点请求报文对象数据发送。 10 远程节点请求报文对象数据发送，但数据还未被发送。当 RMTDND 被置位时，CAN 节点控制器也置位 TXRQ。 当报文对象被成功发送时，RMTDND 被自动复位。
CFCVAL	[15:0] 高位	rwh	报文对象帧计数器值 CFCVAL 包含帧计数器内容的复制值，此时为相应报文对象执行最近一次正确的数据发送或接收时帧计数器内容。

- 1) 进行位 XTD, DIR, NODE 和 CANPTR 的更新时，MSGVAL 必须从 ‘01’ 设置为 ‘10’ 。
- 2) 位 NEWDAT 指示新的数据已经被写入到相应报文对象中。对于发送对象，NEWDAT 应当由软件设置，并且当发送开始时由相应的 CAN 节点控制器复位。

对于接收对象，NEWDAT 在接收到带有匹配标识符的数据帧之后，由相关的 CAN 节点控制器置位，该位必须由软件复位。

当 CAN 节点控制器写新数据到报文对象中时，未用到的报文字节将被不确定的值覆盖。通常，在工作之前 CPU 将清零该位域，并且一旦 CPU 完成工作之后还将验证该位域是否仍清零以确保数据的一致性。对于发送报文对象，CPU 应当在将位域 CPUUPD 清零的同时，置位该位域（NEWDAT）。这种做法将确保，如果在报文被 CPU 更新期间，报文数据实际上被发送，CAN 控制器将不会复位位域 TXRQ。通过这种方法，TXRQ 仅在实际数据被正确传送后才被复位。
- 3) 当位域 MSGVAL 被置位（‘10’）时，通过自动置位位域 TXRQ 和 RMTDND 为 ‘10’，将接收到的匹配远程帧考虑在内（独立于位域 CPUUPD/MSGLST）。只有在 CPUUPD 被复位（‘01’）时，才可能进行帧发送。
- 4) 如果接收对象（DIR = ‘0’）被请求用作发送，将发出一个远程帧，目的是向另一个节点请求数据帧。如果发送对象（DIR = ‘1’）请求发送，将发出数据帧。如果位域 NEWDAT 未被置位或者成功发送远程帧之后，位域 TXRQ 和位域 RMTDND 将被 CAN 控制器一起复位。

注：对于发送帧（远程帧或数据帧），位域 CPUUPD/MSGLST 必须被复位。

报文控制寄存器的控制和状态单元由两个补充位实现（帧计数器值除外）。这种特殊机制不需要经过读—修改—写这样的周期，允许对特定单元进行选择性的置位/复位（而不改变其它部分）。表 21-8 给出该位域的使用方法。

表 21-8 控制寄存器中控制和状态单元的置位/复位

2-位域值	写操作的功能	读操作的意义
00_B	保留	保留
01_B	复位该单元	该单元被复位
10_B	置位该单元	该单元被置位
11_B	该单元不被改变	保留

寄存器 MSGCFGn 定义报文对象 n 的设置和相关的中断节点指针。只有在设置位域 MSGVAL 为 ‘10’ 之后，才能由软件改变位 XTD，NODE 或 DIR。从而避免了当报文对象被明确定义为用于更新的时刻，但仍然有效时，而对其无意地进行修改。当 MSGVAL 为 ‘01’ 或 ‘10’ 时，位 XTD，NODE 或 DIR 可被写入，但只有通过设置 MSGVAL 为 ‘10’，才发生更新操作。

MSGCFGHn (n = 31-0)

报文对象 n 报文设置寄存器高位

复位值: 0000_H

MSGCFGLn (n = 31-0)

报文对象 n 报文设置寄存器低位

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0								TXINP			0	RXINP			
r								rw			r	rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0								DLC			DIR	XTD	NO DE	RMM	
r								rwh			rwh	rw	rwh	rw	

符号	位序号	读写类型	功能描述
RMM	0	rw	发送报文对象远程监控模式
	低位		01 禁止远程监控模式。 10 该发送报文对象的远程监控模式被使能。带匹配标识符的远程帧的标识符和 DLC 码被复制到发送报文对象中，以监控输入的远程帧。 位 RMM 仅对发送报文对象有效，对接收对象无影响。
NODE	1	rwh	报文对象 CAN 节点选择
	低位		0 该报文对象被分配给了 CAN 节点 A 1 该报文对象被分配给了 CAN 节点 B
XTD	2	rw	报文对象扩展标识符
	低位		0 该报文对象使用 11 位标准标识符

符号	位序号	读写类型	功能描述
			1 该报文对象使用 29 位扩展标识符
DIR	3 低位	rwh	报文对象方向控制 0 定义该报文对象为接收对象。如果 TXRQ= ‘10’，发送一个带有该报文对象标识符的远程帧。当接收到带有匹配标识符的数据帧时，报文数据保存到相关 MSGDRn0/MSGDRn4 寄存器中。 1 定义该报文对象为发送对象。如果 TXRQ = ‘10’，发送相应的数据帧。接收到标识符匹配的远程帧，RMTPNB 和 TXQR 被置为 ‘10’。
DLC¹⁾	[7:4] 低位	rwh	报文对象数据长度码 0000 _B – 1XXX _B DLC 包含相关报文对象的数据字节的数目 在远程监控模式和网关模式下，位域 DLC 可能被硬件修改。
RXINP	[2:0] 高位	rw	接收中断节点指针 如果寄存器 MSGCTRn 中的位域 RXIE 被置位，由位域 RXINP 决定报文对象接收事件触发哪一个中断节点。 000 _B 选择 CAN 中断节点 0 111 _B 选择 CAN 中断节点 7
TXINP	[6:4] 高位	rw	发送中断节点指针 如果寄存器 MSGCTRn 中的位域 TXIE 被置位，由位域 TXINP 决定报文对象发送事件触发哪一个中断节点。 000 _B 选择 CAN 中断节点 0 111 _B 选择 CAN 中断节点 7
0	[15:8] 低位	r	保留； 读操作返回 0，应写入 0。

符号	位序号	读写类型	功能描述
	3, [15:7] 高位		

- 1) 数据字节的最大编号为 8。由 CPU 写入的大于 8 的值将被内部更正为 8，但是位域 DLC 的内容不被更新。
- 如果接收到的数据帧包含的数据长度码的值>8，仅考虑处理 8 个字节。读访问位域 DLC，返回接收到的数据帧的 DLC 位域的原始值。

FIFO/网关控制寄存器 MSGFGCRn 中的使能和控制位可以使能并控制 FIFO 功能、网关功能和所要求的传送操作。

MSGFGCRHn (n = 31- 0)

报文对象 n FIFO/网关控制寄存器高位

复位值: 0000_H

MSGFGCRLn (n = 31- 0)

报文对象 n FIFO/网关控制寄存器低位

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0					MMC			0			CANPTR				
r					rw			r			rwh				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STT	SdT	FD	0	DL CC	IDC	SRR EN	GD FS	0			FSIZE				
rw	rw	rw	r	rw	rw	rw	rw	r			rw				

符号	位序号	读写类型	功能描述
FSIZE	[4:0] 低位	rw	<p>FIFO 容量控制</p> <p>位域 FSIZE 决定组成 FIFO 缓存区的报文对象个数。偶数号的报文对象可用作 FIFO 基本对象或从属对象，而奇数号的报文对象仅用作从属对象。在网关模式下，FSIZE 决定目标方 FIFO 长度。</p> <p>00000_B 报文对象 n 为 1 级 FIFO 的部分</p> <p>00001_B 报文对象 n 为 2 级 FIFO 的部分</p> <p>00011_B 报文对象 n 为 4 级 FIFO 的部分</p> <p>00111_B 报文对象 n 为 8 级 FIFO 的部分</p> <p>01111_B 报文对象 n 为 16 级 FIFO 的部分</p> <p>11111_B 报文对象 n 为 32 级 FIFO 的部分</p> <p>其它 保留</p> <p>FSIZE = ‘00000_B’ 导致标准报文对象的操作（用作该操作的指针 CANPTR 将不会改变）。如果目标方需要网关传送给单个报文对象（无 FIFO），必须写入该值。</p>

符号	位序号	读写类型	功能描述
			报文对象被配置为标准模式、共享网关模式或 FIFO 从属功能时，不会评估 FSIZE 的值。在这种情况下，FSIZE 应被编程为‘00000B’。
GDFS	8 低位	rw	<p>网关数据帧发出</p> <p>新数据通过网关从源方传送到目标方之后，指明在目标方是否自动产生 CAN 数据帧。</p> <p>0 无附加操作，目标方 TXRQ 将不被置位。</p> <p>1 自动发出相应的数据帧（CANPTR 指向的报文对象 TXRQ，将被硬件置位）。</p> <p>只有在接收到数据帧（DIR_{<S>} = ‘0’）后，才考虑位 GDFS。</p>
SRREN	9 低位	rw	<p>源方远程请求使能</p> <p>指明在报文对象 n 中（产生一个数据帧），或在 CANPTRn 指向的报文对象中（为了在源方总线上产生一个远程帧）发送请求位是否被置位。</p> <p>0 源方总线上将不产生远程帧，而是根据目标对象的内容在目标总线上产生数据帧（TXRQ 将被置位）。</p> <p>1 将不会发出带目标对象内容的数据帧。而是，由位域 CANPTRn 指向的报文对象产生相应的远程帧（TXRQ[CANPTRn] 将被置位）。</p> <p>SRREN 被限制在正常或共享网关模式下（DIR = ‘1’）的发送报文对象。如果收到远程帧，才考虑该位的影响。</p> <p>如果报文对象 n 是 FIFO 缓存的一部分，不能置位位 SRREN。</p> <p>为了在源方产生远程帧，CANPTR 必须指向源报文对象。</p>
IDC	10	rw	复制标识符

符号	位序号	读写类型	功能描述
	低位		<p>通过网关进行帧传送期间，IDC 控制标识符的处理。</p> <p>0 接收报文对象的标识符不被复制到发送报文对象</p> <p>1 接收报文对象的标识符被自动复制到发送报文对象</p> <p>位域 IDC 仅限于设置为正常网关模式的报文对象。</p>
DLCC	11 低位	rw	<p>数据长度码复制</p> <p>通过网关进行数据帧传送期间，DLCC 控制数据长度码的处理。</p> <p>0 源对象提供的数据长度码不被复制到发送对象</p> <p>1 对接收对象有效的数据长度码，被自动复制到发送对象</p> <p>位域 DLCC 仅限于设置为正常网关模式的报文对象。</p>
FD	13 低位	rw	<p>FIFO 方向</p> <p>仅对于 FIFO 基本对象才考虑 FD。（所有 FIFO 元素的 FD 位应具有一样的值）。它定义导致 FIFO 基本对象的 CANPTR 更新的传送操作类型（接收或发送）。</p> <p>0 FIFO 接收：在当前被寻址的报文对象正确接收一个数据帧（DIR= ‘0’）或远程帧（DIR= ‘1’）之后，FIFO 基本对象的 CANPTR 被更新。在任何发送操作之后，CANPTR 不改变。</p> <p>1 FIFO 发送：在当前被寻址的报文对象正确发送一个数据帧（DIR= ‘1’）或远程帧（DIR= ‘0’）之后，FIFO 基本对象的 CANPTR 被更新。在任何接收操作之后，CANPTR 不改变。</p> <p>位 FD 和位 DIR 不相关。</p>
SDT	14	rw	单数据传送模式

符号	位序号	读写类型	功能描述
	低位		<p>在任何传送模式下（FIFO 模式或作为标准对象，接收对象和发送对象）都要考虑该位。</p> <p>0 该报文对象参与一次成功的数据传送之后（接收或发送），控制位 MSGVAL 不被复位。</p> <p>1 该报文对象参与一次成功的数据传送后（接收或发送），控制位 MSGVAL 被自动复位。</p> <p>对于远程帧，不考虑位 SDT。</p> <p>在属于 FIFO 缓存区的所有报文对象中，位 SDT 必须被复位。</p>
STT	15 低位	rw	<p>单次发送试验</p> <p>0 禁止单次发送试验</p> <p>1 使能单次发送试验，在发送开始后 ¹⁾ 相应的 TXRQ 被立即复位</p>
CANPTR	[4:0] 高位	rwh	<p>FIFO/网关功能的 CAN 指针</p> <p>报文对象设置为标准模式（MMC = '000'）：</p> <p>无影响，CANPTR 应该被初始化为相应的报文对象编号。</p> <p>报文对象设置为 FIFO 基本对象（MMC = '010'）：</p> <p>CANPTR 包含被相关的 CAN 控制器下一次发送或接收操作寻址的报文对象编号。</p> <p>CANPTR 应被初始化为相应 FIFO 基本对象的报文编号。</p> <p>报文对象设置为 FIFO 从属对象（MMC = '011'）：</p> <p>CANPTR 应被初始化为相应 FIFO 基本对象的报文编号。</p> <p>报文对象设置为正常网关模式（MMC = '100'）：</p> <p>CANPTR 包含用做网关目标对象的报文对</p>

符号	位序号	读写类型	功能描述
			<p>象编号</p> <p>报文对象设置为不带 FIFO 功能的网关目标对象（MMC = ‘000’）：</p> <p>如果 SRREN 置为 ‘1’，必须将 CANPTR 初始化为用做网关源报文对象的报文编号。将远程帧从目标方传送到源方的操作需要这个后向指针。如果 SRREN 被清零，CANPTR 不被评估，且必须被初始化为相应的报文对象编号。</p> <p>报文对象被设置为共享网关模式（MMC = ‘101’）：</p> <p>无影响，CANPTR 必须被初始化为相应的报文对象编号。</p> <p>对于 FIFO 功能（或以 FIFO 为目标方的网关功能），当 FIFO 模式被激活且正在进行数据传送时，不应当软件写入 CANPTR。该位域可被用于软件复位 FIFO。</p>
MMC	[10:8] 高位	rw	<p>报文对象模式控制</p> <p>位域 MMC 控制报文对象 n 的功能</p> <p>000_B 标准报文对象功能</p> <p>010_B FIFO 功能使能（基本对象）</p> <p>011_B FIFO 功能使能（从属对象）</p> <p>100_B 用于输入帧的正常网关功能</p> <p>101_B 用于输入帧的共享网关功能</p> <p>其它 保留</p>
0	[7:5] 12 低位 [7:5] [15:11] 高位	r	<p>保留；</p> <p>读操作返回 0，应写入 0。</p>

1) 结果是：如果报文失去仲裁或者已经被错误帧破坏，该报文不会被再次发送。

注：仅在位域 **MSGVAL** 被置为 ‘10’ 之后，才考虑发送报文对象的位域 **CANPTR** 的改变。从而避免了当报文对象被明确定义为用于更新的时刻，但仍然有效时，而对其无意的修改。当 **MSGVAL** 为 ‘01’ 或 ‘10’ 时，发送报文对象的位域 **CANPTR** 可被写入。但只有通过设置 **MSGVAL** 为 ‘10’，才进行更新。立即考虑接收对象位域 **CANPTR** 的改变所带来的影响。

21.2.4 全局 CAN 控制/状态寄存器

接收中断挂起寄存器指示报文对象 n 是否有挂起的接收中断。

RXIPNDH

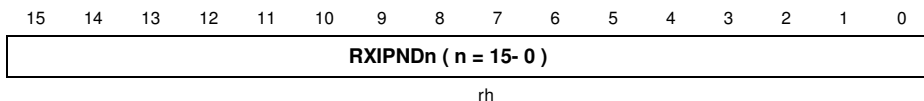
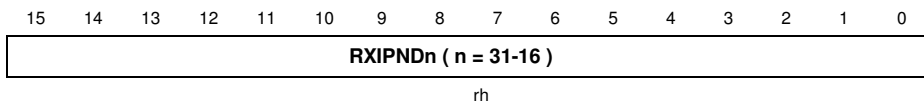
接收中断挂起寄存器高位

复位值: 0000_H

RXIPNDL

接收中断挂起寄存器低位

复位值: 0000_H



符号	位序号	读写类型	功能描述
RXIPNDn (n = 15-0)	0-15 低位	rh	报文对象 n 接收中断挂起 如果报文对象 n 接收到一帧且位 RXIE _n 被置位，那么由硬件置位位 RXIPND _n 。
RXIPNDn (n = 31-16)	n-16 高位		0 报文对象 n 无挂起的接收中断 1 报文对象 n 有挂起的接收中断 通过复位相应的 INT _{PNDn} ，可软件复位 RXIPND _n 。

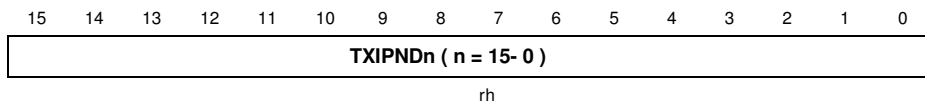
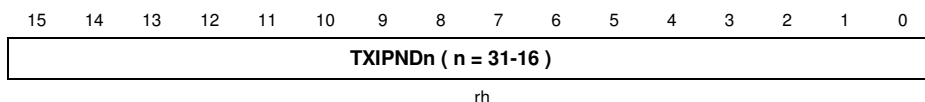
发送中断挂起寄存器指示报文对象 **n** 是否有挂起的发送中断。

TXIPNDH

发送中断挂起寄存器高位 复位值: 0000_H

TXIPNDL

发送中断挂起寄存器低位 复位值: 0000_H



符号	位序号	读写类型	功能描述
TXIPNDn (n = 15- 0)	0-15 低位	rh	报文对象 n 发送中断挂起 如果报文对象 n 发送一帧且位 TXIE _n 已经被置位，那么由硬件置位位 TXIPND _n 。
TXIPNDn (n = 31-16)	n-16 高位		0 报文对象 n 无挂起的发送中断 1 报文对象 n 有挂起的发送中断 通过复位相应的 INT _{PNDn} ，可软件清零 TXIPND _n 。

21.3 XC164CM 模块的实现

本章节描述：

- 和 TwinCAN 模块相关的接口，例如端口连接和中断控制
- 所有和 TwinCAN 模块相关的寄存器以及它们的地址和复位值

21.3.1 TwinCAN 模块接口

在 XC164CM 中，按照图 21-28 将 TwinCAN 模块和 IO 端口相连。

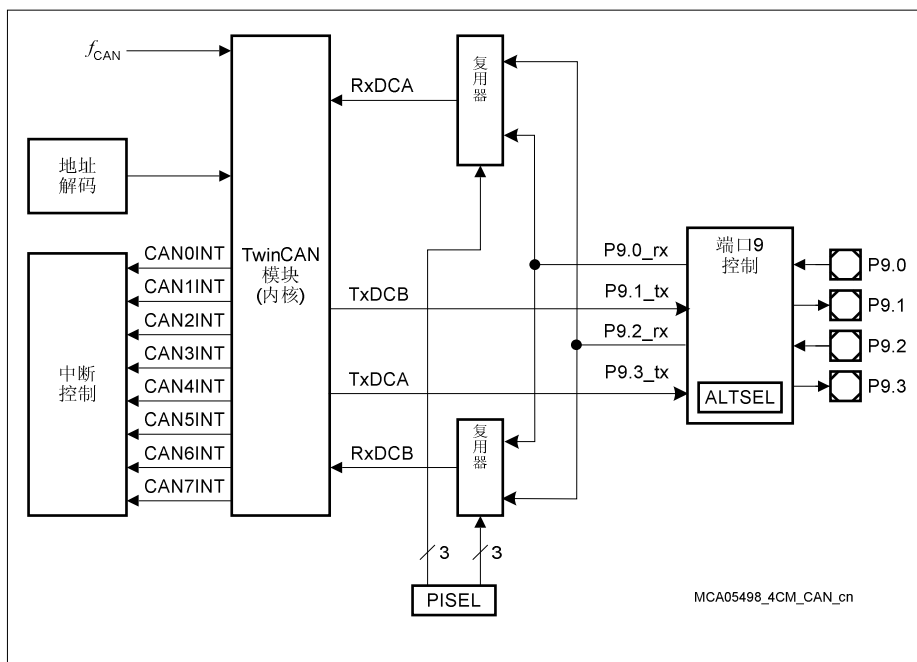


图 21-28 TwinCAN 模块 IO 接口

寄存器 PISEL 中的位域 RISA（对应节点 A）和位域 RISB（对应节点 B）选择输入接收引脚。输出发送引脚由相应的端口 9 的 ALTSEL 寄存器定义。

TwinCAN 模块有 8 条中断请求线。

21.3.2 与 TwinCAN 模块相关的外部寄存器

图 21-29 给出了编程 TwinCAN 模块所需要的外部寄存器。

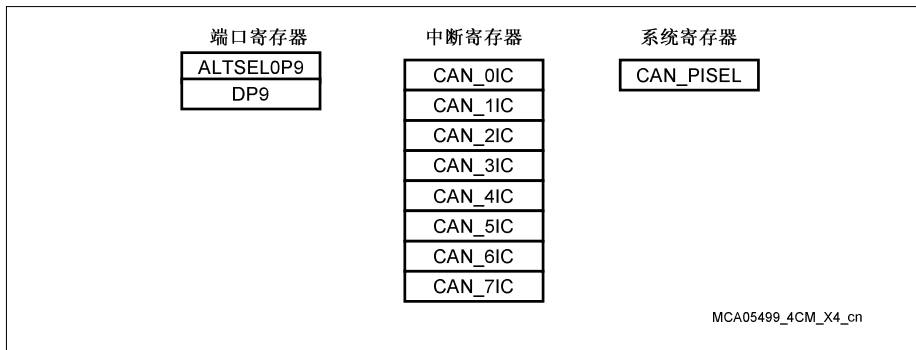


图 21-29 与 TwinCAN 模块应用实现相关的寄存器

21.3.2.1 系统寄存器

寄存器 CAN_PISEL 允许用户为 TwinCAN 接收信号 RXDCA 和 RXDCB 选择输入引脚。

CAN_PISEL

TwinCAN 端口输入选择寄存器

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0										RISB			RISA		
r										rw			rw		

符号	位序号	读写类型	功能描述
RISA	[2:0]	rw	节点 A 的接收输入选择 位域 RISA 为节点 A 的 TwinCAN 输入线 RXDCA 定义输入引脚。 000 保留 001 保留 010 保留 011 RXDCA 的输入引脚是 P9.2 100 RXDCA 的输入引脚是 P9.0 101 保留 11X 保留
RISB	[5:3]	rw	节点 B 的接收输入选择 位域 RISB 为节点 B 的 TwinCAN 输入线 RXDCB 定义输入引脚。 000 保留 001 RXDCB 的输入引脚是 P9.0 010 保留 011 保留 100 RXDCB 的输入引脚是 P9.2 101 保留 11X 保留
0	[15:6]	r	保留； 读操作返回 0，应写入 0。

21.3.2.2 端口寄存器

对 TwinCAN 操作编程所需要的端口寄存器列表如下。

ALTSEL0P9

P9 复用功能选择寄存器 0

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0										0	0	P3	0	P1	0
r										r	r	r	r	r	r

符号	位序号	读写类型	功能描述
ALTSEL0 P9.y	3,1	r	P9 复用功能选择寄存器 0 位 y 0 相关外设输出未选作引脚的复用功能 1 相关外设输出被选作引脚的复用功能

ALTSEL1P9

P9 复用功能选择寄存器 1

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0										0	0	P3	0	P1	0
r										r	r	r	r	r	r

符号	位序号	读写类型	功能描述
ALTSEL1 P9.y	3,1	r	P9 复用功能选择寄存器 1 位 y 0 相关外设输出未选作引脚的复用功能 1 相关外设输出被选作引脚的复用功能

注：阴影部分的位和 TwinCAN 操作无关。

DP9

P9 方向控制寄存器

复位值: 0000_H

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0										P5	P4	P3	P2	P1	P0
r										rw	rw	rw	rw	rw	rw

符号	位序号	读写类型	功能描述
DP9.y	3 ... 0	rw	端口方向寄存器 DP9 位 y 0 端口线 P9.y 为输入（高阻） 1 端口线 P9.y 为输出

注：阴影部分的位和 TwinCAN 操作无关。

表 21-9 TwinCAN 的 IO 线选择和设置

端口线	复用功能 选择寄存器	端口输入 选择寄存器	方向控制 寄存器	IO
-----	---------------	---------------	-------------	----

TwinCAN 节点 A

P9.0/RxDCA	-	CAN_PISEL[5:3]=100 _B	DP9.P0=0	输入
P9.2/RxDCA	-	CAN_PISEL[2:0]=011 _B	DP9.P2=0	输入
P9.3/TxDCA	ALTSEL0P9.P3=1 且 ALTSEL1P9.P3=1	-	DP9.P3=0	输出

TwinCAN 节点 B

P9.0/RxDCB	-	CAN_PISEL[5:3]=001 _B	DP9.P0=0	输入
P9.1/TxDCB	ALTSEL0P9.P1=1 且 ALTSEL1P9.P1=1	-	DP9.P1=1	输出
P9.2/RxDCB	-	CAN_PISEL[2:0]=100 _B	DP9.P2=0	输入

21.3.2.3 中断寄存器

TwinCAN 模块的中断由下列中断控制寄存器来控制：

- CAN_0IC
- CAN_1IC
- CAN_2IC
- CAN_3IC
- CAN_4IC
- CAN_5IC
- CAN_6IC
- CAN_7IC

所有的中断控制寄存器都具有相同的结构。参考用户手册“系统单元”中的相关描述以及中断处理的具体内容。

21.3.3 寄存器表

表 21-10 给出了和 TwinCAN 模块相关的系统寄存器。总结了这些寄存器的地址和复位值。为了简化内核的描述，仅在该寄存器表中加上前缀 ‘CAN’。

TwinCAN 模块的起始地址为 **20'0000_H**，在 TwinCAN 内核描述中给出寄存器偏移地址（相对于该起始地址而言），见 **图 21-27**。所有的 CAN 寄存器列表见 **章节 22.2**。

表 21-10 TwinCAN 模块寄存器总结

名称	描述	地址 ¹⁾	复位值
		16 位	
TwinCAN 模块系统寄存器			
CAN_PISEL	TwinCAN 端口输入选择寄存器	20'0004 _H	0000 _H
CAN_0IC	TwinCAN 中断节点 0 中断控制寄存器	F196 _H	0000 _H
CAN_1IC	TwinCAN 中断节点 1 中断控制寄存器	F142 _H	0000 _H
CAN_2IC	TwinCAN 中断节点 2 中断控制寄存器	F144 _H	0000 _H
CAN_3IC	TwinCAN 中断节点 3 中断控制寄存器	F146 _H	0000 _H
CAN_4IC	TwinCAN 中断节点 4 中断控制寄存器	F148 _H	0000 _H
CAN_5IC	TwinCAN 中断节点 5 中断控制寄存器	F14A _H	0000 _H
CAN_6IC	TwinCAN 中断节点 6 中断控制寄存器	F14C _H	0000 _H
CAN_7IC	TwinCAN 中断节点 7 中断控制寄存器	F14E _H	0000 _H

1) 对于 TwinCAN 模块内核寄存器，8 位短地址不可用。

22 寄存器组

本章总结了所有外设核心寄存器以及与模块相关的外部寄存器。寄存器分两部分列于下表中：第一部分对应 PD+BUS 外设；第二部分对应 LxBus 外设。

22.1 PD+BUS 外设

注：PD+BUS 外设地址指向段 0。

表 22-1 PD+BUS 寄存器列表

寄存器缩写	地址			说明	复位值
	物理地址	8 位地址	存储区域		
异步/同步串行接口 0（ASC0）					
ASC0_CON	FFB0 _H	D8 _H	SFR	ASC0 控制寄存器	0000 _H
ASC0_TBUF	FEB0 _H	58 _H	SFR	ASC0 发送缓存寄存器	0000 _H
ASC0_RBUF	FEB2 _H	59 _H	SFR	ASC0 接收缓存寄存器	0000 _H
ASC0_ABCON	F1B8 _H	DC _H	ESFR	ASC0 自动波特率控制寄存器	0000 _H
ASC0_ABSTAT	F0B8 _H	5C _H	ESFR	ASC0 自动波特率状态寄存器	0000 _H
ASC0_BG	FEB4 _H	5A _H	SFR	ASC0 波特率发生器重载寄存器	0000 _H
ASC0_FDV	FEB6 _H	5B _H	SFR	ASC0 分数分频器寄存器	0000 _H
ASC0_PMW	FEAA _H	55 _H	SFR	ASC0 IrDA 脉冲模式和宽度寄存器	0000 _H
ASC0_RXFCON	F0C6 _H	63 _H	ESFR	ASC0 接收 FIFO 控制寄存器	0000 _H
ASC0_TXFCON	F0C4 _H	62 _H	ESFR	ASC0 发送 FIFO 控制寄存器	0000 _H
ASC0_FSTAT	F0BA _H	5D _H	ESFR	ASC0 FIFO 状态寄存器	0000 _H
异步/同步串行接口 1（ASC1）					
ASC1_CON	FFB8 _H	DC _H	SFR	ASC1 控制寄存器	0000 _H
ASC1_TBUF	FEB8 _H	5C _H	SFR	ASC1 发送缓存寄存器	0000 _H

寄存器组

寄存器缩写	地址			说明	复位值
	物理地址	8 位地址	存储区域		
ASC1_RBUF	FEBA _H	5D _H	SFR	ASC1 接收缓存寄存器	0000 _H
ASC1_ABCON	F1BC _H	DE _H	ESFR	ASC1 自动波特率控制寄存器	0000 _H
ASC1_ABSTAT	F0BC _H	5E _H	ESFR	ASC1 自动波特率状态寄存器	0000 _H
ASC1_BG	FEBC _H	5E _H	SFR	ASC1 波特率发生器重载寄存器	0000 _H
ASC1_FDV	FEBE _H	5F _H	SFR	ASC1 分数分频器寄存器	0000 _H
ASC1_PMW	FEAC _H	56 _H	SFR	ASC1 IrDA 脉冲模式和宽度寄存器	0000 _H
ASC1_RXFCON	F0A6 _H	53 _H	ESFR	ASC1 接收 FIFO 控制寄存器	0000 _H
ASC1_TXFCON	F0A4 _H	52 _H	ESFR	ASC1 发送 FIFO 控制寄存器	0000 _H
ASC1_FSTAT	F0BE _H	5F _H	ESFR	ASC1 FIFO 状态寄存器	0000 _H

同步串行通道 0（SSC0）

SSC0_CON	FFB2 _H	D9 _H	SFR	SSC0 控制寄存器	0000 _H
SSC0_BR	F0B4 _H	5A _H	ESFR	SSC0 波特率定时器重载寄存器	0000 _H
SSC0_TB	F0B0 _H	58 _H	ESFR	SSC0 发送缓存寄存器	0000 _H
SSC0_RB	F0B2 _H	59 _H	ESFR	SSC0 接收缓存寄存器	0000 _H

同步串行通道 1（SSC1）

SSC1_CON	FF5E _H	AF _H	SFR	SSC1 控制寄存器	0000 _H
SSC1_BR	F05E _H	2F _H	ESFR	SSC1 波特率定时器重载寄存器	0000 _H
SSC1_TB	F05A _H	2D _H	ESFR	SSC1 发送缓存寄存器	0000 _H
SSC1_RB	F05C _H	2E _H	ESFR	SSC1 接收缓存寄存器	0000 _H

通用定时器单元（GPT12E）

寄存器组

寄存器缩写	地址			说明	复位值
	物理地址	8 位地址	存储区域		
GPT12E_T2CON	FF40 _H	A0 _H	SFR	GPT12E 定时器 2 控制寄存器	0000 _H
GPT12E_T3CON	FF42 _H	A1 _H	SFR	GPT12E 定时器 3 控制寄存器	0000 _H
GPT12E_T4CON	FF44 _H	A2 _H	SFR	GPT12E 定时器 4 控制寄存器	0000 _H
GPT12E_T5CON	FF46 _H	A3 _H	SFR	GPT12E 定时器 5 控制寄存器	0000 _H
GPT12E_T6CON	FF48 _H	A4 _H	SFR	GPT12E 定时器 6 控制寄存器	0000 _H
GPT12E_CAPREL	FE4A _H	25 _H	SFR	GPT12E 捕获/重载寄存器	0000 _H
GPT12E_T2	FE40 _H	20 _H	SFR	GPT12E 定时器 2 寄存器	0000 _H
GPT12E_T3	FE42 _H	21 _H	SFR	GPT12E 定时器 3 寄存器	0000 _H
GPT12E_T4	FE44 _H	22 _H	SFR	GPT12E 定时器 4 寄存器	0000 _H
GPT12E_T5	FE46 _H	23 _H	SFR	GPT12E 定时器 5 寄存器	0000 _H
GPT12E_T6	FE48 _H	24 _H	SFR	GPT12E 定时器 6 寄存器	0000 _H

实时时钟（RTC）

RTC_CON	F110 _H	88 _H	ESFR	RTC 控制寄存器，低位字	8003 _H
RTC_T14	F0D2 _H	69 _H	ESFR	定时器 T14 寄存器	UUUU _H
RTC_T14REL	F0D0 _H	68 _H	ESFR	定时器 T14 重载寄存器	UUUU _H
RTC_RTCL	F0D4 _H	6A _H	ESFR	RTC 定时器低位寄存器	UUUU _H
RTC_RTCH	F0D6 _H	6B _H	ESFR	RTC 定时器高位寄存器	UUUU _H
RTC_RELL	F0CC _H	66 _H	ESFR	RTC 重载低位寄存器	0000 _H
RTC_RELH	F0CE _H	67 _H	ESFR	RTC 重载高位寄存器	0000 _H
RTC_ISNC	F10C _H	86 _H	ESFR	RTC 中断子节点寄存器	0000 _H

寄存器组

寄存器缩写	地址			说明	复位值
	物理地址	8 位地址	存储区域		
捕获/ 比较单元 2（CAPCOM2）					
CC2_M4	FF22 _H	91 _H	SFR	CAPCOM2 模式控制寄存器 4	0000 _H
CC2_M5	FF24 _H	92 _H	SFR	CAPCOM2 模式控制寄存器 5	0000 _H
CC2_M6	FF26 _H	93 _H	SFR	CAPCOM2 模式控制寄存器 6	0000 _H
CC2_M7	FF28 _H	94 _H	SFR	CAPCOM2 模式控制寄存器 7	0000 _H
CC2_SEE	FE2A _H	15 _H	SFR	CAPCOM2 单次事件使能寄存器	0000 _H
CC2_SEM	FE28 _H	14 _H	SFR	CAPCOM2 单次事件模式寄存器	0000 _H
CC2_DRM	FF2A _H	95 _H	SFR	CAPCOM2 双寄存器模式寄存器	0000 _H
CC2_OUT	FF2C _H	96 _H	SFR	CAPCOM2 输出寄存器	0000 _H
CC2_T7	F050 _H	28 _H	ESFR	CAPCOM2 定时器 7 寄存器	0000 _H
CC2_T8	F052 _H	29 _H	ESFR	CAPCOM2 定时器 8 寄存器	0000 _H
CC2_T7REL	F054 _H	2A _H	ESFR	CAPCOM2 定时器 7 重载寄存器	0000 _H
CC2_T8REL	F056 _H	2B _H	ESFR	CAPCOM2 定时器 8 重载寄存器	0000 _H
CC2_T78CON	FF20 _H	90 _H	SFR	CAPCOM2 定时器 7 和定时器 8 控制寄存器	0000 _H
CC2_IOC	F066 _H	33 _H	ESFR	CAPCOM2 输入/ 输出控制寄存器	0000 _H
CC2_CC16	FE60 _H	30 _H	SFR	CAPCOM2 寄存器 16	0000 _H
CC2_CC17	FE62 _H	31 _H	SFR	CAPCOM2 寄存器 17	0000 _H

寄存器组

寄存器缩写	地址			说明	复位值
	物理地址	8 位地址	存储区域		
CC2_CC18	FE64 _H	32 _H	SFR	CAPCOM2 寄存器 18	0000 _H
CC2_CC19	FE66 _H	33 _H	SFR	CAPCOM2 寄存器 19	0000 _H
CC2_CC20	FE68 _H	34 _H	SFR	CAPCOM2 寄存器 20	0000 _H
CC2_CC21	FE6A _H	35 _H	SFR	CAPCOM2 寄存器 21	0000 _H
CC2_CC22	FE6C _H	36 _H	SFR	CAPCOM2 寄存器 22	0000 _H
CC2_CC23	FE6E _H	37 _H	SFR	CAPCOM2 寄存器 23	0000 _H
CC2_CC24	FE70 _H	38 _H	SFR	CAPCOM2 寄存器 24	0000 _H
CC2_CC25	FE72 _H	39 _H	SFR	CAPCOM2 寄存器 25	0000 _H
CC2_CC26	FE74 _H	3A _H	SFR	CAPCOM2 寄存器 26	0000 _H
CC2_CC27	FE76 _H	3B _H	SFR	CAPCOM2 寄存器 27	0000 _H
CC2_CC28	FE78 _H	3C _H	SFR	CAPCOM2 寄存器 28	0000 _H
CC2_CC29	FE7A _H	3D _H	SFR	CAPCOM2 寄存器 29	0000 _H
CC2_CC30	FE7C _H	3E _H	SFR	CAPCOM2 寄存器 30	0000 _H
CC2_CC31	FE7E _H	3F _H	SFR	CAPCOM2 寄存器 31	0000 _H

捕获/比较单元 6（CCU6）

CCU6_T12	E890 _H	-	IO	定时器 12 计数寄存器	0000 _H
CCU6_T12PR	E892 _H	-	IO	定时器 12 周期寄存器	0000 _H
CCU6_T12DTC	E894 _H	-	IO	定时器 12 死区时间控制寄存器	0000 _H
CCU6_CC60R	E898 _H	-	IO	捕获/比较寄存器（通道 0）	0000 _H
CCU6_CC61R	E89A _H	-	IO	捕获/比较寄存器（通道 1）	0000 _H
CCU6_CC62R	E89C _H	-	IO	捕获/比较寄存器（通道 2）	0000 _H
CCU6_CC60SR	E8A0 _H	-	IO	捕获/比较映射寄存器（通道	0000 _H

寄存器组

寄存器缩写	地址			说明	复位值
	物理地址	8 位地址	存储区域		
				0)	
CCU6_CC61SR	E8A2 _H	-	IO	捕获/比较映射寄存器（通道 1）	0000 _H
CCU6_CC62SR	E8A4 _H	-	IO	捕获/比较映射寄存器（通道 2）	0000 _H
CCU6_T13	E8B0 _H	-	IO	定时器 13 计数寄存器	0000 _H
CCU6_T13PR	E8B2 _H	-	IO	定时器 13 周期寄存器	0000 _H
CCU6_CC63R	E8B4 _H	-	IO	定时器 13 比较寄存器	0000 _H
CCU6_CC63SR	E8B6 _H	-	IO	定时器 13 比较映射寄存器	0000 _H
CCU6_CMPSTAT	E8A8 _H	-	IO	比较状态寄存器	0000 _H
CCU6_CMPMODIF	E8AA _H	-	IO	比较状态修改寄存器	0000 _H
CCU6_TCTR0	E8AC _H	-	IO	定时器控制寄存器 0	0000 _H
CCU6_TCTR2	E8AE _H	-	IO	定时器控制寄存器 2	0000 _H
CCU6_TCTR4	E8A6 _H	-	IO	定时器控制寄存器 4	0000 _H
CCU6_MODCTR	E8C0 _H	-	IO	调制控制寄存器	0000 _H
CCU6_TRPCTR	E8C2 _H	-	IO	强制中断控制寄存器	0000 _H
CCU6_PSLR	E8C4 _H	-	IO	被动态电平寄存器	0000 _H
CCU6_T12MSEL	E8C6 _H	-	IO	T12 捕获/比较模式选择寄存器	0000 _H
CCU6_MCMOUTS	E8CA _H	-	IO	多通道模式输出映射寄存器	0000 _H
CCU6_MCMOUT	E8CC _H	-	IO	多通道模式输出寄存器	0000 _H

寄存器组

寄存器缩写	地址			说明	复位值
	物理地址	8 位地址	存储区域		
CCU6_MCMCTR	E8CE _H	-	IO	多通道模式控制寄存器	0000 _H
CCU6_IS	E8D0 _H	-	IO	捕获/比较中断状态寄存器	0000 _H
CCU6_ISS	E8D2 _H	-	IO	捕获/比较中断状态置位寄存器	0000 _H
CCU6_ISR	E8D4 _H	-	IO	捕获/比较中断状态复位寄存器	0000 _H
CCU6_INP	E8D6 _H	-	IO	捕获/比较中断节点指针寄存器	3940 _H
CCU6_IEN	E8D8 _H	-	IO	捕获/比较中断使能寄存器	0000 _H

模数转换器（ADC）

ADC_CON	FFA0 _H	D0 _H	SFR	模数转换器控制寄存器	0000 _H
ADC_CON1	FFA6 _H	D3 _H	SFR	模数转换器控制寄存器 1	0000 _H
ADC_CTR0	FFBE _H	DF _H	SFR	模数转换器控制寄存器 0	1000 _H
ADC_CTR2	F09C _H	4E _H	ESFR	模数转换器控制寄存器 2	0000 _H
ADC_CTR2IN	F09E _H	4F _H	ESFR	模数转换器插入控制寄存器 2	0000 _H
ADC_DAT	FEA0 _H	50 _H	SFR	模数转换器结果寄存器	0000 _H
ADC_DAT2	F0A0 _H	50 _H	ESFR	模数转换器 2 结果寄存器	0000 _H

中断控制

SSC0_TIC	FF72 _H	B9 _H	SFR	SSC0 发送中断控制寄存器	0000 _H
SSC0_RIC	FF74 _H	BA _H	SFR	SSC0 接收中断控制寄存器	0000 _H
SSC0_EIC	FF76 _H	BB _H	SFR	SSC0 出错中断控制寄存器	0000 _H
SSC1_TIC	F1AA _H	D5 _H	ESFR	SSC1 发送中断控制寄存器	0000 _H
SSC1_RIC	F1AC _H	D6 _H	SFR	SSC1 接收中断控制寄存器	0000 _H
SSC1_EIC	F1AE _H	D7 _H	ESFR	SSC1 出错中断控制寄存器	0000 _H

寄存器组

寄存器缩写	地址			说明	复位值
	物理地址	8 位地址	存储区域		
ASC0_TIC	FF6C _H	B6 _H	SFR	ASC0 发送中断控制寄存器	0000 _H
ASC0_RIC	FF6E _H	B7 _H	SFR	ASC0 接收中断控制寄存器	0000 _H
ASC0_EIC	FF70 _H	B8 _H	SFR	ASC0 出错中断控制寄存器	0000 _H
ASC0_TBIC	F19C _H	CE _H	ESFR	ASC0 发送缓冲中断控制寄存器	0000 _H
ASC0_ABIC	F15C _H	AE _H	ESFR	ASC0 自动波特率中断控制寄存器	0000 _H
ASC1_TIC	F182 _H	C1 _H	ESFR	ASC1 发送中断控制寄存器	0000 _H
ASC1_RIC	F18A _H	C5 _H	ESFR	ASC1 接收中断控制寄存器	0000 _H
ASC1_EIC	F192 _H	C9 _H	ESFR	ASC1 出错中断控制寄存器	0000 _H
ASC1_TBIC	F150 _H	A8 _H	ESFR	ASC1 发送缓冲中断控制寄存器	0000 _H
ASC1_ABIC	F1BA _H	DD _H	ESFR	ASC1 自动波特率中断控制寄存器	0000 _H
GTP12E_T2IC	FF60 _H	B0 _H	SFR	GTP12E 定时器 2 中断控制寄存器	0000 _H
GTP12E_T3IC	FF62 _H	B1 _H	SFR	GTP12E 定时器 3 中断控制寄存器	0000 _H
GTP12E_T4IC	FF64 _H	B2 _H	SFR	GTP12E 定时器 4 中断控制寄存器	0000 _H
GTP12E_T5IC	FF66 _H	B3 _H	SFR	GTP12E 定时器 5 中断控制寄存器	0000 _H
GTP12E_T6IC	FF68 _H	B4 _H	SFR	GTP12E 定时器 6 中断控制寄存器	0000 _H
GTP12E_CRIC	FF6A _H	B5 _H	SFR	GTP12E CAPREL 中断控制寄存器	0000 _H
CC2_T7IC	F17A _H	BD _H	ESFR	CAPCOM 定时器 7 中断控制	0000 _H

寄存器组

寄存器缩写	地址			说明	复位值
	物理地址	8 位地址	存储区域		
				寄存器	
CC2_T8IC	F17C _H	BE _H	ESFR	CAPCOM 定时器 8 中断控制寄存器	0000 _H
CC1_CC8IC	FF88 _H	C4 _H	SFR	CAPCOM 寄存器 8 中断控制寄存器	0000 _H
CC1_CC9IC	FF8A _H	C5 _H	SFR	CAPCOM 寄存器 9 中断控制寄存器	0000 _H
CC1_CC10IC	FF8C _H	C6 _H	SFR	CAPCOM 寄存器 10 中断控制寄存器	0000 _H
CC1_CC11IC	FF8E _H	C7 _H	SFR	CAPCOM 寄存器 11 中断控制寄存器	0000 _H
CC1_CC12IC	FF90 _H	C8 _H	SFR	CAPCOM 寄存器 12 中断控制寄存器	0000 _H
CC1_CC13IC	FF92 _H	C9 _H	SFR	CAPCOM 寄存器 13 中断控制寄存器	0000 _H
CC2_CC16IC	F160 _H	B0 _H	ESFR	CAPCOM 寄存器 16 中断控制寄存器	0000 _H
CC2_CC17IC	F162 _H	B1 _H	ESFR	CAPCOM 寄存器 17 中断控制寄存器	0000 _H
CC2_CC18IC	F164 _H	B2 _H	ESFR	CAPCOM 寄存器 18 中断控制寄存器	0000 _H
CC2_CC19IC	F166 _H	B3 _H	ESFR	CAPCOM 寄存器 19 中断控制寄存器	0000 _H
CC2_CC20IC	F168 _H	B4 _H	ESFR	CAPCOM 寄存器 20 中断控制寄存器	0000 _H
CC2_CC21IC	F16A _H	B5 _H	ESFR	CAPCOM 寄存器 21 中断控制寄存器	0000 _H
CC2_CC22IC	F16C _H	B6 _H	ESFR	CAPCOM 寄存器 22 中断控制寄存器	0000 _H

寄存器组

寄存器缩写	地址			说明	复位值
	物理地址	8 位地址	存储区域		
				制寄存器	
CC2_CC23IC	F16E _H	B7 _H	ESFR	CAPCOM 寄存器 23 中断控制寄存器	0000 _H
CC2_CC24IC	F170 _H	B8 _H	ESFR	CAPCOM 寄存器 24 中断控制寄存器	0000 _H
CC2_CC25IC	F172 _H	B9 _H	ESFR	CAPCOM 寄存器 25 中断控制寄存器	0000 _H
CC2_CC26IC	F174 _H	BA _H	ESFR	CAPCOM 寄存器 26 中断控制寄存器	0000 _H
CC2_CC27IC	F176 _H	BB _H	ESFR	CAPCOM 寄存器 27 中断控制寄存器	0000 _H
CC2_CC28IC	F178 _H	BC _H	ESFR	CAPCOM 寄存器 28 中断控制寄存器	0000 _H
CC2_CC29IC	F184 _H	C2 _H	ESFR	CAPCOM 寄存器 29 中断控制寄存器	0000 _H
CC2_CC30IC	F18C _H	C6 _H	ESFR	CAPCOM 寄存器 30 中断控制寄存器	0000 _H
CC2_CC31IC	F194 _H	CA _H	ESFR	CAPCOM 寄存器 31 中断控制寄存器	0000 _H
CCU6_IC	F140 _H	A0 _H	ESFR	其它中断的中断控制寄存器（模块中断节点 I3）	0000 _H
CCU6_EIC	F188 _H	C4 _H	ESFR	紧急事件中中断的中断控制寄存器（模块中断节点 I2）	0000 _H
CCU6_T12IC	F190 _H	C8 _H	ESFR	T12 中断的中断控制寄存器（模块中断节点 I0）	0000 _H
CCU6_T13IC	F198 _H	CC _H	ESFR	T13 中断的中断控制寄存器（模块中断节点 I1）	0000 _H
ADC_CIC	FF98 _H	CC _H	SFR	模数转换器转换结束中断控	0000 _H

寄存器组

寄存器缩写	地址			说明	复位值
	物理地址	8 位地址	存储区域		
				制寄存器	
ADC_EIC	FF9A _H	CD _H	SFR	模数转换器过载错误中断控制寄存器	0000 _H
CAN_0IC	F196 _H	CB _H	ESFR	TwinCAN 中断控制寄存器 0	0000 _H
CAN_1IC	F142 _H	A1 _H	ESFR	TwinCAN 中断控制寄存器 1	0000 _H
CAN_2IC	F144 _H	A2 _H	ESFR	TwinCAN 中断控制寄存器 2	0000 _H
CAN_3IC	F146 _H	A3 _H	ESFR	TwinCAN 中断控制寄存器 3	0000 _H
CAN_4IC	F148 _H	A4 _H	ESFR	TwinCAN 中断控制寄存器 4	0000 _H
CAN_5IC	F14A _H	A5 _H	ESFR	TwinCAN 中断控制寄存器 5	0000 _H
CAN_6IC	F14C _H	A6 _H	ESFR	TwinCAN 中断控制寄存器 6	0000 _H
CAN_7IC	F14E _H	A7 _H	ESFR	TwinCAN 中断控制寄存器 7	0000 _H
EOPIC	F180 _H	C0 _H	ESFR	PEC 子通道结束中断控制寄存器	0000 _H
PLL_IC	F19E _H	CF _H	ESFR	PLL 中断控制寄存器	0000 _H
RTC_IC	F1A0 _H	D0 _H	ESFR	RTC 中断控制寄存器	0000 _H

端口

PICON	F1C4 _H	E2 _H	ESFR	端口输入阈值控制寄存器	0000 _H
POCON1L	F084 _H	42 _H	ESFR	P1L 输出控制寄存器	0000 _H
POCON1H	F086 _H	43 _H	ESFR	P1H 输出控制寄存器	0000 _H
POCON3	F08A _H	45 _H	ESFR	P3 输出控制寄存器	0000 _H
POCON9	F094 _H	4A _H	ESFR	P9 输出控制寄存器	0000 _H
P1L	FF04 _H	82 _H	SFR	P1 口低位寄存器	0000 _H
P1H	FF06 _H	83 _H	SFR	P1 口高位寄存器	0000 _H
DP1L	F104 _H	82 _H	ESFR	P1L 方向控制寄存器	0000 _H

寄存器组

寄存器缩写	地址			说明	复位值
	物理地址	8 位地址	存储区域		
DP1H	F106 _H	83 _H	ESFR	P1H 方向控制寄存器	0000 _H
ALTSEL0P1L	F130 _H	98 _H	ESFR	P1L 复用功能选择寄存器 0	0000 _H
ALTSEL0P1H	F120 _H	90 _H	ESFR	P1H 复用功能选择寄存器 0	0000 _H
P3	FFC4 _H	E2 _H	SFR	P3 口数据寄存器	0000 _H
DP3	FFC6 _H	E3 _H	SFR	P3 口方向控制寄存器	0000 _H
ODP3	F1C6 _H	E3 _H	ESFR	P3 口漏极开路控制寄存器	0000 _H
ALTSEL0P3	F126 _H	93 _H	ESFR	P3 口复用功能选择寄存器 0	0000 _H
ALTSEL1P3	F128 _H	94 _H	ESFR	P3 口复用功能选择寄存器 1	0000 _H
P5	FFA2 _H	D1 _H	SFR	P5 口数据寄存器	0000 _H
P5DIDIS	FFA4 _H	D2 _H	SFR	P5 口数字输入禁止寄存器	0000 _H
P9	FF16 _H	8B _H	SFR	P9 口数据寄存器	0000 _H
DP9	FF18 _H	8C _H	SFR	P9 口方向控制寄存器	0000 _H
ODP9	FF1A _H	8D _H	SFR	P9 口漏极开路控制寄存器	0000 _H
ALTSEL0P9	F138 _H	9C _H	ESFR	P9 口复用功能选择寄存器 0	0000 _H
ALTSEL1P9	F13A _H	9D _H	ESFR	P9 口复用功能选择寄存器 1	0000 _H

22.2 LXBUS 外设

注：LXBUS 外设地址指向段 32；可由用户程序修改。

表 22-2 LXBUS 寄存器列表

寄存器缩写	物理地址	说明	复位值
TwinCAN			
CAN_PISEL	20'0004 _H	TwinCAN 端口输入选择寄存器	0000 _H
CAN_ACR	20'0200 _H	节点 A 控制寄存器	0001 _H
CAN_ASR	20'0204 _H	节点 A 状态寄存器	0000 _H
CAN_AIR	20'0208 _H	节点 A 中断挂起寄存器	0000 _H
CAN_ABTRL	20'020C _H	节点 A 位定时低位寄存器	0000 _H
CAN_ABTRH	20'020E _H	节点 A 位定时高位寄存器	0000 _H
CAN_AGINP	20'0210 _H	节点 A 全局中断节点指针寄存器	0000 _H
CAN_AFCRL	20'0214 _H	节点 A 帧计数器寄存器低位	0000 _H
CAN_AFCRH	20'0216 _H	节点 A 帧计数器寄存器高位	0000 _H
CAN_AIMRL0	20'0218 _H	节点 A INTID 屏蔽寄存器 0 低位	0000 _H
CAN_AIMRH0	20'021A _H	节点 A INTID 屏蔽寄存器 0 高位	0000 _H
CAN_AIMR4	20'021C _H	节点 A INTID 屏蔽寄存器 4	0000 _H
CAN_AECNTL	20'0220 _H	节点 A 错误计数器寄存器低位	0000 _H
CAN_AECNTH	20'0222 _H	节点 A 错误计数器寄存器高位	0060 _H
CAN_BCR	20'0240 _H	节点 B 控制寄存器	0001 _H
CAN_BSR	20'0244 _H	节点 B 状态寄存器	0000 _H
CAN_BIR	20'0248 _H	节点 B 中断挂起寄存器	0000 _H
CAN_BBTRL	20'024C _H	节点 B 位定时寄存器低位	0000 _H

寄存器组

寄存器缩写	物理地址	说明	复位值
CAN_BBTRH	20'024E _H	节点 B 位定时寄存器高位	0000 _H
CAN_BGINP	20'0250 _H	节点 B 全局中断节点指针寄存器	0000 _H
CAN_BFCRL	20'0254 _H	节点 B 帧计数器寄存器低位	0000 _H
CAN_BFCRH	20'0256 _H	节点 B 帧计数器寄存器高位	0000 _H
CAN_BIMRL0	20'0258 _H	节点 B INTID 屏蔽寄存器 0 低位	0000 _H
CAN_BIMRH0	20'025A _H	节点 B INTID 屏蔽寄存器 0 高位	0000 _H
CAN_BIMR4	20'025C _H	节点 B INTID 屏蔽寄存器 4	0000 _H
CAN_BECLTL	20'0260 _H	节点 B 错误计数器寄存器低位	0000 _H
CAN_BECLTH	20'0262 _H	节点 B 错误计数器寄存器高位	0060 _H
CAN_RXIPNDL	20'0284 _H	接收中断挂起寄存器低位	0000 _H
CAN_RXIPNDH	20'0286 _H	接收中断挂起寄存器高位	0000 _H
CAN_TXIPNDL	20'0288 _H	发送中断挂起寄存器低位	0000 _H
CAN_TXIPNDH	20'028A _H	发送中断挂起寄存器高位	0000 _H

中断控制

CAN_0IC	00'F196 _H ¹⁾	TwinCAN 中断控制寄存器 0	0000 _H
CAN_1IC	00'F142 _H ¹⁾	TwinCAN 中断控制寄存器 1	0000 _H
CAN_2IC	00'F144 _H ¹⁾	TwinCAN 中断控制寄存器 2	0000 _H
CAN_3IC	00'F146 _H ¹⁾	TwinCAN 中断控制寄存器 3	0000 _H
CAN_4IC	00'F148 _H ¹⁾	TwinCAN 中断控制寄存器 4	0000 _H
CAN_5IC	00'F14A _H ¹⁾	TwinCAN 中断控制寄存器 5	0000 _H
CAN_6IC	00'F14C _H ¹⁾	TwinCAN 中断控制寄存器 6	0000 _H

寄存器缩写	物理地址	说明	复位值
CAN_7IC	00’F14E _H ¹⁾	TwinCAN 中断控制寄存器 7	0000 _H

1) 该寄存器位于 ESFR 区。

每个报文对象 n（n = 0 - 31）的基地址列于表 22-3 中。报文对象 n 中每个寄存器的偏移地址在表 22-4 给出。

表 22-3 报文对象基地址

报文对象编号	基地址
报文对象 0	20’0300 _H
报文对象 1	20’0320 _H
报文对象 2	20’0340 _H
报文对象 3	20’0360 _H
报文对象 4	20’0380 _H
报文对象 5	20’03A0 _H
报文对象 6	20’03C0 _H
报文对象 7	20’03E0 _H
报文对象 8	20’0400 _H
报文对象 9	20’0420 _H
报文对象 10	20’0440 _H
报文对象 11	20’0460 _H
报文对象 12	20’0480 _H
报文对象 13	20’04A0 _H
报文对象 14	20’04C0 _H
报文对象 15	20’04E0 _H
报文对象 16	20’0500 _H
报文对象 17	20’0520 _H

报文对象编号	基地址
报文对象 18	20'0540 _H
报文对象 19	20'0560 _H
报文对象 20	20'0580 _H
报文对象 21	20'05A0 _H
报文对象 22	20'05C0 _H
报文对象 23	20'05E0 _H
报文对象 24	20'0600 _H
报文对象 25	20'0620 _H
报文对象 26	20'0640 _H
报文对象 27	20'0660 _H
报文对象 28	20'0680 _H
报文对象 29	20'06A0 _H
报文对象 30	20'06C0 _H
报文对象 31	20'06E0 _H

表 22-4 报文对象寄存器偏移地址

寄存器缩写	偏移地址	说明	复位值
CAN_MSGDRLn0	00 _H	报文对象 n 数据寄存器 0 低位	0000 _H
CAN_MSGDRHn0	02 _H	报文对象 n 数据寄存器 0 高位	0000 _H
CAN_MSGDRLn4	04 _H	报文对象 n 数据寄存器 4 低位	0000 _H
CAN_MSGDRHn4	06 _H	报文对象 n 数据寄存器 4 高位	0000 _H
CAN_MSGARLn	08 _H	报文对象 n 仲裁寄存器低位	0000 _H
CAN_MSGARHn	0A _H	报文对象 n 仲裁寄存器高位	0000 _H
CAN_MSGAMRLn	0C _H	报文对象 n 验收屏蔽寄存器低位	0000 _H
CAN_MSGAMRHn	0E _H	报文对象 n 验收屏蔽寄存器高位	0000 _H
CAN_MSGCTRLn	10 _H	报文对象 n 控制寄存器低位	0000 _H
CAN_MSGCTRHn	12 _H	报文对象 n 控制寄存器高位	0000 _H
CAN_MSGCFGLn	14 _H	报文对象 n 设置寄存器低位	0000 _H
CAN_MSGCFGHn	16 _H	报文对象 n 设置寄存器高位	0000 _H
CAN_MSGFGCRLn	18 _H	报文对象 n FIFO/ 网关控制寄存器低位	0000 _H
CAN_MSGFGCRHn	1A	报文对象 n FIFO/ 网关控制寄存器高位	0000 _H

注：n = 0 ~ 31。

关键字索引

用户可通过关键字方便的查阅其对应 **XC164CM** 架构、功能单元或模块功能的详细描述。检索关键字可帮助用户快速找到 **XC164CM** 相关问题的答案。

本用户手册分为两卷：“系统单元”和“外设单元”。为了便于用户的使用，关键字索引（以及目录）将两卷中的关键字（以及章节）统一列出，以便用户快速查阅相关内容（卷[1]或卷[2]）。

A

Acronyms（缩写），1-10

Adapt Mode（自适应模式），6-11

ADC（模数转换器），2-22, 16-1

ADC_CIC, 16-23

ADC_CON, 16-4

ADC_CON1, 16-5

ADC_CTR0, 16-7

ADC_CTR2, 16-8

ADC_CTR2IN, 16-8

ADC_EIC, 16-23

Address（地址）

Boundaries（边界），3-15

Mapping（映射），3-3

Addressing Modes（寻址模式）

CoREG Addressing Mode（CoREG 寻址模式），4-55

DSP Addressing Modes（DSP 寻址模式），4-51

Indirect Addressing Modes（间接寻址模式），4-49

Long Addressing Modes（长寻址模式），4-45

Short Addressing Modes（短寻址模式），4-43

Alternate Port Functions（复用端口功能），7-7

ALU（算术逻辑运算单元），4-62

Analog/Digital Converter（模数转换器），16-1

Arbitration of conversions（转换仲裁），16-18

ASC（异步/同步串行接口），19-1

ASCx_EIC, 19-37

ASCx_RIC, 19-37

ASCx_TBIC, 19-37

ASCx_TIC, 19-37

Autobaud Detection（自动波特率检测），19-27

Error Detection（错误检测），19-35

Features and Functions（特性和功能），19-1

IrDA Frames（IrDA 帧），19-8

Register（寄存器）

BG, 19-22

RBUF, 19-12, 19-20

TBUF, 19-9, 19-20

Transmit FIFO（发送 FIFO），19-9

ASCx_BG, 19-45

ASCx_CON, 19-42

ASCx_FDV, 19-46

Auto Scan conversion（自动扫描转换），16-14

Autobaud Detection（自动波特率检测），19-27

B

Baudrate（波特率）

ASC0（异步/同步串行接口 0），19-22

Bootstrap Loader（引导程序加载器），10-5

CAN（控制器局域网），21-55

Bit（位）

Handling（处理），4-65

Manipulation Instructions（操作指令），12-2

protected（保护），2-31, 4-65

reserved（保留），2-15

Block Diagram ITC/PEC（中断和 PEC 控制器框图），5-3

BNKSELx, 5-33

Bootstrap Loader（引导程序加载器），6-11, 10-1

Boundaries（边界），3-15

Bus（总线）

ASC（异步/同步串行接口），19-1

CAN（控制器局域网），2-25

SSC（高速同步串行接口），20-1

C

Calibration（校准），16-19

CAN（控制器局域网）

Acceptance filtering（验收滤波），21-16

analysing mode（分析模式），21-7

arbitration（仲裁），21-16

baudrate（波特率），21-55

bit timing（位定时），21-9, 21-55

bus off（总线关闭）

recovery sequence（恢复序列），21-4

status bit（状态位），21-50

- error counters（错误计数器），21-53
- error handling（错误处理），21-11
- error warning level（错误警告级别），21-53
- frame counter/time stamp（帧计数器/时间戳），21-57
- interface（接口），2-25
- single data transfer（单次数据传送），21-22
- CAPCOM12（捕获比较单元 1/2）
 - Capture Mode（捕获模式），17-13
 - Counter Mode（计数器模式），17-8
- CAPCOM2（捕获比较单元 2），2-16
- CAPCOM6（捕获比较单元 6），2-18
- CAPREL, 14-58
- Capture Mode（捕获模式）
 - GPT1（通用定时器单元 1），14-27
 - GPT2（CAPREL），14-50
- Capture/Compare Registers（捕获/比较寄存器），17-10
- CC1_SEE, 17-27
- CC2_DRM, 17-23
- CC2_IOC, 17-29
- CC2_M4-7, 17-10
- CC2_OUT, 17-25
- CC2_SEE, 17-27
- CC2_SEM, 17-27
- CC2_T78CON, 17-5
- CC2_T7IC, 17-9
- CC2_T8IC, 17-9
- CC63R, 18-43
- CC63SR, 18-43
- CC6xR, 18-19
- CC6xSR, 18-20
- CCU6_xIC, 18-84
- CCxIC, 17-34
- Clock（时钟）
 - generation（产生），2-28
 - output signal（输出信号），6-25
- CMPMODIF, 18-50
- CMPSTAT, 18-49
- Command sequences（命令序列），3-19
- Concatenation of Timers（定时器级联），14-23, 14-49
- Configuration（配置）
 - Reset（复位），6-10
- Context（上下文）
 - Pointer Updating（指针更新），4-36
 - Switch（切换），4-35

Switching（切换），5-32
Conversion（转换）
 analog/digital（模拟/数字），16-1
 Arbitration（仲裁），16-18
 Auto Scan（自动扫描），16-14
 timing control（时序控制），16-20
Count direction（计数方向），14-6, 14-37
Counter Mode（计数器模式）（GPT1），14-10, 14-41
Counter（计数器），14-20, 14-47
CP, 4-38
CPU（中央处理器），2-2, 4-1
CPUCON1, 4-26
CPUCON2, 4-27
CRIC, 14-59
CSP, 4-41

D

Data Management Unit（数据管理单元），2-10
Data Page（数据页），4-46
 boundaries（边界），3-15
Data SRAM（数据 SRAM），3-10
Development Support（开发支持），1-8
Direction（方向）
 count（计数），14-6, 14-37
Disable（禁止）
 Interrupt（中断），5-29
Division（除法），4-66
Double-Register Compare（双寄存器比较），17-22
DP1H, 7-9
DP1L, 7-9
DP3, 7-19
DP9, 7-42, 21-86
DPP, 4-46
Driver characteristic (ports)（驱动器特性（端口）），7-4
DSTPx, 5-23
Dual-Port RAM（双端口 RAM），3-10

E

EBC（外部总线控制器）
 Memory Table（存储器表），9-10
Edge characteristic(ports)（边沿特性（端口）），7-5

EMUCON, 6-35
Enable（使能）
 Interrupt（中断），5-29
End of PEC Interrupt Sub Node（PEC 结束中断子节点），5-28
EOPIC, 5-27
Erase command（擦除命令），3-21
Error correction（纠错），3-25
Error Detection（错误检测）
 ASC（异步/同步串行接口），19-35
 SSC（高速同步串行接口），20-15
EXICON, 5-37
EXISEL0, 5-38
EXISEL1, 5-38
External（外部）
 Fast interrupts（快速中断），5-37
 Interrupt pulses（中断脉冲），5-40
 Interrupt source control（中断源控制），5-37
 Interrupts during sleep mode（休眠模式期间的中断），5-39
 Interrupts（中断），5-35

F

Fast external interrupts（快速外部中断），5-37
FINT0ADDR, 5-16
FINT0CSP, 5-16
FINT1ADDR, 5-16
FINT1CSP, 5-17
Flags（标志位），4-61–4-64
Flash（闪存）
 command sequences（命令序列），3-19
 memory mapping（存储器映射），3-16
 memory（存储器），3-12
 waitstates（等待状态），3-41
FOCON, 6-26
Frequency（频率）
 output signal（输出信号），6-25
FSR, 3-33

G

Gated timer mode（门控定时器模式）（GPT1），14-9
Gated timer mode（门控定时器模式）（GPT2），14-40
GPR（通用寄存器），3-7

GPT（通用定时器），2-19
GPT1（通用定时器单元 1），14-2
GPT12E_CAPREL, 14-58
GPT12E_CRIC, 14-59
GPT12E_T2, 14-31
GPT12E_T2CON, 14-15
GPT12E_T2IC, 14-32
GPT12E_T3, 14-31
GPT12E_T3CON, 14-4
GPT12E_T3IC, 14-32
GPT12E_T4, 14-31
GPT12E_T4CON, 14-15
GPT12E_T4IC, 14-32
GPT12E_T5, 14-58
GPT12E_T5CON, 14-42
GPT12E_T5IC, 14-59
GPT12E_T6, 14-58
GPT12E_T6CON, 14-35
GPT12E_T6IC, 14-59
GPT2（通用定时器单元 2），14-33

H

Hardware（硬件）
Traps（强制中断），5-43

I

IDCHIP, 6-51
Idle Mode（空闲模式），6-41
IDMANUF, 6-51
IDMEM, 6-52
IDPROG, 6-52
IDX0, 4-51
IDX1, 4-51
IEN, 18-82
IMB（内部程序存储模块）
 block diagram（框图），3-39
 control functions（功能），3-42
 memories（存储器）
 address map（地址映射），3-39
 wait states（等待状态），3-42
IMBCTR, 3-42
Incremental Interface Mode（增量接口模式）（GPT1），14-11
INP, 18-83

Instruction（指令），12-1
 Bit Manipulation（位操作），12-2
 Pipeline（流水线），4-11
 protected（保护），12-6
Interface（接口）
 ASC（异步/同步串行接口），19-1
 CAN（控制器局域网），2-25
 External Bus（外部总线），9-1
 SSC（高速同步串行接口），20-1
Interrupt Handling（中断处理）
 CAN transfer（CAN 传送），21-5
Interrupt（中断）
 Arbitration（仲裁），5-4
 during sleep mode（休眠模式期间），5-39
 Enable/Disable（使能/禁止），5-29
 External（外部），5-35
 Fast external（快速外部），5-37
 input timing（输入时序），5-40
 Jump Table Cache（跳转表缓存），5-16
 Latency（延迟），5-41
 Node Sharing（节点共享），5-34
 Priority（优先级），5-7
 Processing（处理），5-1
 RTC（实时时钟），15-11
 source control（请求源控制），5-37
 Source（请求源），5-11
 System（系统），2-9, 5-2
 Vectors（向量），5-11
IP, 4-41
IrDA Frames ASC（IrDA 帧 ASC），19-8
IS, 18-77
ISR, 18-81
ISS, 18-80

L

Latency（延迟）
 Interrupt, PEC（中断，外围事件控制器），5-41

M

MAH, 4-73
MAL, 4-73
用户手册

MAR, 3-27
Margin check（裕量检测）, 3-27
MCMCTR, 18-64
MCMOUT, 18-62
MCMOUTS, 18-61
MCW, 4-70
MDC, 4-67
MDH, 4-66
MDL, 4-67
Memory（存储器）, 2-11
 Areas(Data)（区域（数据））, 3-9
 Areas(Program)（区域（程序））, 3-11
 DPRAM（双端口 RAM）, 3-10
 DSRAM（数据 SRAM）, 3-10
 Flash（闪存）, 3-12
 Program Flash（程序闪存）, 3-16
 PSRAM, 3-12
MODCTR, 18-53
MRW, 4-76
MSW, 4-74
Multiplication（乘法）, 4-66

N

NMI（非屏蔽中断）, 5-1, 5-48
Noise filter(Ext.Interrupts)（噪声滤波（外部中断））, 5-40

O

OCDS（片上调试支持）
 Requests（请求）, 5-41
ODP3, 7-20
ODP9, 7-43
ONES, 4-78
Open Drain Mode（漏极开路模式）, 7-3
OPSEN, 6-36
Oscillator（振荡器）
 circuitry（电路）, 6-14
 measurement（测量）, 6-14
 Watchdog（看门狗）, 6-24

P

P1H, 7-8

P1L, 7-8
P3, 7-19
P5, 7-37
P9, 7-42
PEC pointer（PEC 指针）, 3-8
PEC（外围事件控制器）, 2-11, 5-18
 Latency（延迟）, 5-41
 Transfer Count（传送计数）, 5-19
PECCx, 5-19
PECISNC, 5-27
PECSEGx, 5-24
Peripheral（外设）
 Event Controller（事件控制器）, 5-18
 Register Set（寄存器组）, 22-1
 Summary（归纳）, 2-14
Phase Locked Loop（锁相环）, 6-13
PICON, 7-2
Pins（引脚）, 8-1
Pipeline（流水线）, 4-11
PLL（锁相环）, 6-13
PLL_IC, 6-24
PLLCON, 6-18
POCON*, 7-6
Port（端口）, 2-27
Ports（端口）
 Alternate Port Functions（复用端口功能）, 7-7
 Driver characteristic（驱动器特性）, 7-4
 Edge characteristic（边沿特性）, 7-5
Power Management（功率管理）, 2-29
PROCON, 3-29
Program Management Unit（程序管理单元）, 2-10
Programming command（编程命令）, 3-21
Protected（保护）
 Bits（位）, 2-31, 4-65
 features（特性）, 3-28
 Instruction（指令）, 12-6
Protection（保护）
 Commands（命令）, 3-23
PSLR, 18-73
PSW, 4-61

Q

QR0, 4-50
QR1, 4-50
QX0, 4-52
QX1, 4-52

R

RAM
 data SRAM（数据 SRAM）, 3-10
 dual ported（双端口）, 3-10
 program/data（程序/数据）, 3-12
 status after reset（复位后的状态）, 6-7
Real Time Clock（实时时钟）, 2-21, 15-1
Register Areas（寄存器区）, 3-5
Register map（寄存器映射）
 TwinCAN module（TwinCAN 模块）, 21-46
Register Table（寄存器表）
 LXBUS peripherals（LXBUS 外设）, 22-13
 PD+BUS peripherals（PD+BUS 外设）, 22-1
RELH, 15-8
RELL, 15-8
Reserved bits（保留位）, 2-15
Reset（复位）, 6-2
 Configuration（配置）, 6-10
 Source indication（触发源指示）, 6-33
 Values（值）, 6-5
RSTCON, 6-12
RTC（实时时钟）, 2-21
RTC（实时时钟）, 15-1
RTC_CON, 15-5
RTC_IC, 15-12
RTC_ISNC, 15-12
RTCH, 15-7
RTCL, 15-7

S

SCUSLC, 6-39
SCUSLS, 6-38
Security（安全）
 features（特性）, 3-28
Segment（段）

- boundaries（边界），3-15
- Segmentation（分段），4-40
- Self-calibration（自校准），16-19
- Serial Interface（串行接口），2-23, 2-24
 - ASC（异步/同步串行接口），19-1
 - Asynchronous（异步），19-5
 - CAN（控制器局域网），2-25
 - SSC（高速同步串行接口），20-1
 - Synchronous（同步），19-19
- SFR（特殊功能寄存器），3-5
- Sharing（共享）
 - Interrupt Nodes（中断节点），5-34
- Sleep mode（休眠模式），6-43
- Software（软件）
 - Traps（强制中断），5-43
- Source（源）
 - Interrupt（中断），5-11
 - Reset（复位），6-33
- SP, 4-58
- SPSEG, 4-58
- SRAM
 - Data（数据），3-10
- SRCPx, 5-23
- SSC（高速同步串行接口），20-1
 - Baudrate generation（波特率产生），20-13
 - Block diagram（框图），20-3
 - Continuous transfer operation（连续传送操作），20-13
 - Error detection（错误检测），20-15
 - Full duplex operation（全双工操作），20-9
 - General operation（操作概述），20-1
 - Half duplex operation（半双工操作），20-11
 - Interrupts（中断），20-15
- SSCx_CON, 20-4, 20-5
- Stack（堆栈），3-13, 4-57
- Startup Configuration（启动配置），6-10
- STKOV, 4-60
- STKUN, 4-60
- SYSCON0, 6-29
- SYSCON1, 6-30
- SYSCON3, 6-44
- SYSSTAT, 6-32

T

T12, 18-6
T12DTC, 18-24
T12MSEL, 18-21
T12PR, 18-6
T13, 18-33
T13PR, 18-33
T2, 14-31
T2CON, 14-15
T2IC, 14-32
T3, 14-31
T3CON, 14-4
T3IC, 14-32
T4, 14-31
T4CON, 14-15
T4IC, 14-32
T5, 14-58
T5CON, 14-42
T5IC, 14-59
T6, 14-58
T6CON, 14-35
T6IC, 14-59
T7IC, 17-9
T8IC, 17-9
TCTR0, 18-44
TCTR2, 18-46
TCTR4, 18-47
TFR, 5-45
Timer（定时器）, 14-2, 14-33
 Auxiliary Timer（辅助定时器）, 14-15, 14-42
 Concatenation（级联）, 14-23, 14-49
 Core Timer（核心定时器）, 14-4, 14-35
 Counter Mode（计数器模式）（GPT1）, 14-10, 14-41
 Gated Mode（门控模式）（GPT1）, 14-9
 Gated Mode（门控模式）（GPT2）, 14-40
 Incremental Interface Mode（增量接口模式）（GPT1）, 14-11
 Mode（模式）（GPT1）, 14-8
 Mode（模式）（GPT2）, 14-39
Tools（工具）, 1-8
Transmit FIFO（发送 FIFO）, 19-9
Traps（强制中断）, 5-43
TRPCTR, 18-67
TwinCAN
 FIFO（先入先出）

- Base object（基本对象），21-23
- Circular buffer（环形缓存），21-25
- configuration（配置），21-74
- for CAN messages（CAN 报文），21-23
- Gateway control（网关控制），21-74
- Slave objects（从属对象），21-25
- Frames（帧）
 - Counter（计数器），21-8
 - Handling（处理），21-17
- gateway（网关）
 - configuration（配置），21-74
- Gateway（网关）
 - Normal mode（正常模式），21-28
 - Shared mode（共享模式），21-35
 - With FIFO（带 FIFO），21-32
- Initialization（初始化），21-40
- Interrupts（中断）
 - Indication（指示）/ INTID, 21-13, 21-52
 - Node pointer/request compressor（节点指针/请求压缩），21-5
- Loop-back mode（回环模式），21-44
- Message handling（报文处理），21-15
 - FIFO（先入先出），21-23
 - Gateway + FIFO（网关+FIFO），21-32
 - Gateway overview（网关概述），21-27
 - Normal gateway（正常网关），21-28
 - Shared gateway（共享网关），21-35
 - Transfer control（传送控制），21-41
- Message interrupts（报文中断），21-13
- Message object（报文对象）
 - Configuration（配置），21-71
 - Control bits（控制位），21-67
 - Interrupt indication（中断指示），21-13
 - Interrupts（中断），21-13
 - Register description（寄存器描述），21-63
 - Transfer handling（传送处理），21-17
- Node control（节点控制），21-7
- Node interrupts（节点中断），21-12
- Node selection（节点选择），21-71
- Overview（概述），21-1
- register map（寄存器映射），21-46
- Registers（global）寄存器（全局）

- Receive interrupt pending（接收中断挂起），21-80
- Transmit interrupt pending（发送中断挂起），21-81
- Registers（message specific）寄存器（报文专用）
 - Acceptance mask（验收屏蔽），21-66
 - Arbitration（identifier）仲裁（标识符），21-65
 - Configuration（配置），21-71
 - Control（控制），21-67
 - Data（数据），21-64
- Registers（node specific）寄存器（节点专用）
 - Bit timing（位定时），21-55
 - Frame counter（帧计数器），21-57
 - Global interrupt node pointer（全局中断节点指针），21-59
 - Interrupt pending（中断挂起），21-52
 - INTID mask（屏蔽），21-61
 - Status（状态），21-50
- Single transmission（单次发送），21-45
- single-shot mode（单次模式），21-22
- Transfer interrupts（传送中断），21-6
- TwinCAN registers（short names）TwinCAN 寄存器（缩写）
 - ABTRH, 21-55
 - ABTRL, 21-55
 - ACR, 21-48
 - AECNTH, 21-53
 - AECNTL, 21-53
 - AFCRH, 21-57
 - AFCRL, 21-57
 - AGINP, 21-59
 - AIMR4, 21-62
 - AIMRH0, 21-61
 - AIMRL0, 21-61
 - AIR, 21-52
 - ASR, 21-50
 - BBTRH, 21-55
 - BBTRL, 21-55
 - BCR, 21-48
 - BECNTH, 21-53
 - BECNTL, 21-53
 - BFCRH, 21-57
 - BFCRL, 21-57
 - BGINP, 21-59
 - BIMR4, 21-62
 - BIMRH0, 21-61
 - BIMRL0, 21-61
 - BIR, 21-52

BSR, 21-50
MSGAMRHn, 21-66
MSGAMRLn, 21-66
MSGARHn, 21-65
MSGARLn, 21-65
MSGCFGHn, 21-71
MSGCFGLn, 21-71
MSGCTRHn, 21-67
MSGCTRLn, 21-67
MSGDRHn0, 21-63
MSGDRHn4, 21-64
MSGDRLn0, 21-63
MSGDRLn4, 21-64
MSGFGCRHn, 21-74
MSGFGCRLn, 21-74
RXIPNDH, 21-80
RXIPNDL, 21-80
TXIPNDH, 21-81
TXIPNDL, 21-81

V

VECSEG, 5-10

W

Waitstates（等待状态）
Flash（闪存）, 3-41
Watchdog（看门狗）, 2-26, 6-46
after reset（复位后）, 6-7
Oscillator（振荡器）, 6-24
WDT（看门狗定时器）, 6-47
WDTCON, 6-49

Z

ZEROS, 4-78

<http://www.infineon.com>