**B6x**

**BLE芯片使用指南**

**V1.0.0**

©2022

免*责声明：具体实施的描述只是为了说明问题，实际的硬件实施可能有所不同。*

**版本记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **版本** | **日期** | **修订者** | **说明** |
| V1.0.0 | 2023.02.01 | Wanghailong | 初版 |
|  |  |  |  |

**目录**

[1. 系统概述 1](#_Toc5596)

[1.1 概述 1](#_Toc26792)

[1.1.1 特性 1](#_Toc28041)

[1.2 系统地址映射 3](#_Toc29833)

[1.3 系统框图 4](#_Toc13907)

[1.4 端口定义与复用信号 4](#_Toc25992)

[1.5 中断嵌套向量定义 6](#_Toc20318)

[1.5.1 中断优先级 6](#_Toc27629)

[1.5.2 中断向量分配 6](#_Toc1727)

[1.6 事件唤醒 8](#_Toc10423)

[2. 系统配置控制器(SYSCFG) 9](#_Toc24497)

[2.1 概述 9](#_Toc18318)

[2.2 寄存器映射 9](#_Toc32585)

[2.2.1 系统备份寄存器(sys\_backup0) 9](#_Toc2320)

[3. IO功能复用控制(CSC) 10](#_Toc7175)

[3.1 概述 10](#_Toc18608)

[3.2 功能联接 10](#_Toc28342)

[3.3 寄存器映射 10](#_Toc8847)

[3.3.1 CSC输出控制寄存器映射 10](#_Toc22388)

[3.3.2 CSC输入控制寄存器映射 12](#_Toc16613)

[3.3.3 CSC控制寄存器映射 12](#_Toc16351)

[3.4 寄存器说明 14](#_Toc22013)

[3.4.1 CSC寄存器 14](#_Toc19378)

[3.4.2 CSC控制寄存器 14](#_Toc11988)

[3.5 CSC功能选择定义 16](#_Toc13153)

[4. 串行外设接口(SPIM) 17](#_Toc27600)

[4.1 概述 17](#_Toc16496)

[4.2 寄存器映射 17](#_Toc6200)

[4.3 寄存器说明 17](#_Toc24981)

[4.3.1 接收寄存器(rx\_data) 17](#_Toc24141)

[4.3.2 发送寄存器(tx\_data) 18](#_Toc15596)

[4.3.3 启动寄存器(txrx\_bgn) 18](#_Toc17708)

[4.3.4 控制寄存器(ctrl) 18](#_Toc6339)

[4.3.5 清除寄存器(status\_clr) 19](#_Toc3823)

[4.3.6 状态寄存器(status) 20](#_Toc30660)

[4.3.7 数据长度寄存器(dat\_len) 20](#_Toc16549)

[5. 串行外设接口(SPIS) 21](#_Toc27538)

[5.1 概述 21](#_Toc27404)

[5.2 寄存器映射 21](#_Toc30986)

[5.3 寄存器说明 21](#_Toc16960)

[5.3.1 控制寄存器(ctrl) 21](#_Toc963)

[5.3.2 状态寄存器(status) 22](#_Toc17334)

[5.3.3 清除寄存器(info\_clr) 23](#_Toc8854)

[5.3.4 发送寄存器(tx\_data) 23](#_Toc25918)

[5.3.5 接收寄存器(rx\_data) 24](#_Toc3359)

[6. 通用 I/O（GPIO） 25](#_Toc23776)

[6.1 概述 25](#_Toc4768)

[6.2 结构框图 25](#_Toc20684)

[6.3 寄存器映射 25](#_Toc19630)

[6.4 寄存器说明 26](#_Toc30657)

[6.4.1 IO数据设置寄存器(dat\_set) 26](#_Toc5262)

[6.4.2 IO数据清除寄存器(dat\_clr) 26](#_Toc13704)

[6.4.3 IO数据翻转寄存器(dat\_tog) 26](#_Toc14378)

[6.4.4 IO数据屏蔽寄存器(dat\_msk) 27](#_Toc23678)

[6.4.5 IO数据寄存器(dat) 27](#_Toc5130)

[6.4.6 IO状态寄存器(pin) 27](#_Toc17474)

[6.4.7 IO方向设置寄存器(dir\_set) 28](#_Toc8627)

[6.4.8 IO方向清除寄存器(dir\_clr) 28](#_Toc9987)

[6.4.9 IO方向翻转寄存器(dir\_tog) 28](#_Toc18008)

[6.4.10 IO方向屏蔽寄存器(dir\_msk) 29](#_Toc511)

[6.4.11 IO方向寄存器(dir) 29](#_Toc19396)

[7. 基本定时器(BASIC TIMER) 30](#_Toc23345)

[7.1 概述 30](#_Toc32230)

[7.2 结构框图 30](#_Toc7492)

[7.3 寄存器映射 31](#_Toc31078)

[7.4 寄存器说明 31](#_Toc1087)

[7.4.1 控制寄存器1(cr1) 31](#_Toc23589)

[7.4.2 控制寄存器2(cr2) 33](#_Toc10385)

[7.4.3 中断使能寄存器(ier) 34](#_Toc20132)

[7.4.4 中断关闭寄存器(idr) 34](#_Toc25434)

[7.4.5 中断有效状态寄存器(ivs) 34](#_Toc21912)

[7.4.6 中断有效标志寄存器(rif) 35](#_Toc24107)

[7.4.7 中断屏蔽寄存器(ifm) 35](#_Toc5438)

[7.4.8 中断状态清除寄存器(icr) 36](#_Toc29071)

[7.4.9 事件产生寄存器(egr) 36](#_Toc15403)

[7.4.10 计数寄存器(cnt) 37](#_Toc20766)

[7.4.11 预分频寄存器(psc) 37](#_Toc14754)

[7.4.12 重载寄存器(arr) 37](#_Toc14571)

[7.4.13 DMA控制寄存器(dmaen) 38](#_Toc17349)

[8. 通用定时器(COMMON TIMER) 39](#_Toc12163)

[8.1 概述 39](#_Toc5143)

[8.2 结构框图 40](#_Toc18657)

[8.3 寄存器映射 40](#_Toc13600)

[8.4 寄存器说明 41](#_Toc31231)

[8.4.1 控制寄存器1(cr1) 41](#_Toc21052)

[8.4.2 控制寄存器2(cr2) 42](#_Toc18923)

[8.4.3 从机模式控制寄存器(smcr) 44](#_Toc17113)

[8.4.4 中断使能寄存器(ier) 44](#_Toc21644)

[8.4.5 中断关闭寄存器(idr) 45](#_Toc5297)

[8.4.6 中断有效状态寄存器(ivs) 46](#_Toc29812)

[8.4.7 中断有效标志寄存器(rif) 47](#_Toc13507)

[8.4.8 中断屏蔽寄存器(ifm) 49](#_Toc21628)

[8.4.9 中断状态清除寄存器(icr) 50](#_Toc5396)

[8.4.10 事件产生寄存器(egr) 51](#_Toc28406)

[8.4.11 捕获/比较模式寄存器1(ccmr1) 52](#_Toc23937)

[8.4.12 捕获/比较模式寄存器2(ccmr2) 55](#_Toc28603)

[8.4.13 捕获/比较模式使能寄存器(ccer) 57](#_Toc26671)

[8.4.14 计数寄存器(cnt) 59](#_Toc22368)

[8.4.15 预分频寄存器(psc) 59](#_Toc3668)

[8.4.16 重载寄存器(arr) 59](#_Toc29012)

[8.4.17 重复计数寄存器(rcr) 60](#_Toc12399)

[8.4.18 捕获/比较寄存器1(ccr1) 60](#_Toc31674)

[8.4.19 捕获/比较寄存器2(ccr2) 61](#_Toc32049)

[8.4.20 捕获/比较寄存器3(ccr3) 61](#_Toc5942)

[8.4.21 捕获/比较寄存器4(ccr4) 61](#_Toc21656)

[8.4.22 DMA控制寄存器(dmaen) 62](#_Toc1547)

[9. 高级定时器(ADVANCED TIMER) 63](#_Toc20759)

[9.1 概述 63](#_Toc15673)

[9.2 结构框图 63](#_Toc3768)

[9.3 寄存器映射 64](#_Toc8059)

[9.4 寄存器说明 65](#_Toc18591)

[9.4.1 控制寄存器1(cr1) 65](#_Toc14684)

[9.4.2 控制寄存器2(cr2) 67](#_Toc26458)

[9.4.3 从机模式控制寄存器(smcr) 69](#_Toc20149)

[9.4.4 中断使能寄存器(ier) 71](#_Toc11989)

[9.4.5 中断关闭寄存器(idr) 72](#_Toc19792)

[9.4.6 中断有效状态寄存器(ivs) 73](#_Toc28923)

[9.4.7 中断有效标志寄存器(rif) 74](#_Toc5519)

[9.4.8 中断屏蔽寄存器(ifm) 75](#_Toc4678)

[9.4.9 中断状态清除寄存器(icr) 78](#_Toc20795)

[9.4.10 事件产生寄存器(egr) 79](#_Toc22466)

[9.4.11 捕获/比较模式寄存器1(ccmr1) 80](#_Toc4589)

[9.4.12 捕获/比较模式寄存器2(ccmr2) 83](#_Toc29274)

[9.4.13 捕获/比较模式使能寄存器(ccer) 85](#_Toc31670)

[9.4.14 计数寄存器(cnt) 87](#_Toc6507)

[9.4.15 预分频寄存器(psc) 87](#_Toc9698)

[9.4.16 重载寄存器(arr) 87](#_Toc22244)

[9.4.17 重复计数寄存器(rcr) 88](#_Toc27415)

[9.4.18 捕获/比较寄存器1(ccr1) 88](#_Toc31109)

[9.4.19 捕获/比较寄存器2(ccr2) 89](#_Toc27417)

[9.4.20 捕获/比较寄存器3(ccr3) 89](#_Toc1528)

[9.4.21 捕获/比较寄存器4(ccr4) 90](#_Toc21494)

[9.4.22 中断和死线寄存器(bdtr) 90](#_Toc1501)

[9.4.23 DMA控制寄存器(dmaen) 91](#_Toc30068)

[10. 通用异步收发器(UART) 93](#_Toc19453)

[10.1 概述 93](#_Toc30954)

[10.2 结构框图 94](#_Toc16186)

[10.3 寄存器映射 94](#_Toc10482)

[10.4 寄存器说明 95](#_Toc2562)

[10.4.1 UART接收寄存器(rbr) 95](#_Toc20372)

[10.4.2 UART发送寄存器(tbr) 95](#_Toc14922)

[10.4.3 UART波特率控制寄存器(brr) 96](#_Toc9942)

[10.4.4 UART线路控制寄存器(lcr) 96](#_Toc32342)

[10.4.5 UART模式控制寄存器(mcr) 97](#_Toc30431)

[10.4.6 UART控制寄存器(cr) 98](#_Toc19677)

[10.4.7 UART接收超时寄存器(rtor) 99](#_Toc5638)

[10.4.8 UART管道控制寄存器(fcr) 99](#_Toc31570)

[10.4.9 UART状态寄存器(sr) 100](#_Toc25221)

[10.4.10 UART中断使能寄存器(ier) 101](#_Toc6993)

[10.4.11 UART中断关闭寄存器(idr) 101](#_Toc19121)

[10.4.12 UART中断有效状态寄存器(ivs) 102](#_Toc122)

[10.4.13 UART中断标志状态寄存器(rif) 103](#_Toc24534)

[10.4.14 UART中断屏蔽寄存器(ifm) 104](#_Toc6733)

[10.4.15 UART中断清除寄存器(icr) 105](#_Toc4404)

[11. 外部中断/事件控制器(EXTI) 107](#_Toc27077)

[11.1 概述 107](#_Toc22453)

[11.2 寄存器映射 107](#_Toc22707)

[11.3 寄存器说明 108](#_Toc8255)

[11.3.1 外部中断使能寄存器(ier) 108](#_Toc10552)

[11.3.2 外部中断关闭寄存器(idr) 108](#_Toc29155)

[11.3.3 外部中断有效状态寄存器(ivs) 109](#_Toc9679)

[11.3.4 外部中断标志状态寄存器(rif) 109](#_Toc30259)

[11.3.5 外部中断屏蔽寄存器(ifm) 109](#_Toc16057)

[11.3.6 外部中断清除寄存器(icr) 110](#_Toc32128)

[11.3.7 外部中断上升沿触发寄存器(rts) 110](#_Toc26581)

[11.3.8 外部中断下降沿触发寄存器(fts) 110](#_Toc25123)

[11.3.9 外部中断软件触发寄存器(swi) 110](#_Toc17102)

[11.3.10 外部中断定时器触发寄存器(adte) 111](#_Toc13962)

[11.3.11 外部中断消抖使能寄存器(db) 111](#_Toc22028)

[11.3.12 外部中断消抖控制寄存器(dbc) 111](#_Toc5014)

[12. 集成内部总线接口(I2C) 113](#_Toc8761)

[12.1 概述 113](#_Toc3649)

[12.2 寄存器映射 113](#_Toc31045)

[12.3 寄存器说明 114](#_Toc1553)

[12.3.1 控制设置寄存器(cr\_set) 114](#_Toc9364)

[12.3.2 控制清除寄存器(cr\_clr) 115](#_Toc12041)

[12.3.3 状态寄存器(sr) 116](#_Toc32667)

[12.3.4 数据寄存器(dr) 117](#_Toc6210)

[12.3.5 时钟寄存器(clk) 117](#_Toc24742)

[12.3.6 从机0地址寄存器(sadr0) 118](#_Toc31781)

[12.3.7 从机0地址屏蔽寄存器(adm0) 118](#_Toc2200)

[12.3.8 拓展从机地址寄存器(xsadr) 118](#_Toc20380)

[12.3.9 拓展从机地址屏蔽寄存器(xadm) 119](#_Toc23013)

[12.3.10 软件复位寄存器(srst) 119](#_Toc28604)

[12.3.11 从机1地址寄存器(sadr1) 119](#_Toc22547)

[12.3.12 从机1地址屏蔽寄存器(adm1) 120](#_Toc2475)

[12.3.13 从机2地址寄存器(sadr2) 120](#_Toc15489)

[12.3.14 从机2地址屏蔽寄存器(adm2) 121](#_Toc30418)

[12.3.15 从机3地址寄存器(sadr3) 121](#_Toc4691)

[12.3.16 从机3地址屏蔽寄存器(adm3) 121](#_Toc22436)

[13. 独立看门狗(IWDG) 123](#_Toc6820)

[13.1 概述 123](#_Toc18807)

[13.2 寄存器映射 123](#_Toc887)

[13.3 寄存器说明 123](#_Toc14401)

[13.3.1 计数重载值寄存器(load) 123](#_Toc3833)

[13.3.2 当前计数值寄存器(value) 124](#_Toc4926)

[13.3.3 控制寄存器(ctrl) 124](#_Toc30228)

[13.3.4 中断标志清除寄存器(iniclr) 125](#_Toc9163)

[13.3.5 中断标志寄存器(ris) 125](#_Toc26731)

[13.3.6 锁存寄存器(lock) 125](#_Toc32753)

# 系统概述

## 概述

B6x系列芯片是一款超低功率、高性能和高集成的蓝牙5.2 BLE + 2.4G片上系统芯片，设计用于在 2360MHz 至 2520MHz频段上运行。

B6x系列芯片 采用先进的 CMOS 低漏电工艺制造，具有最高的集成度、最低的功耗、最低的漏电电流和降低 BOM 成本，同时简化了整个系统设计。

### 特性

* 蓝牙BT 5.2 + 2.4GHz(私有协议) SoC
* 支持BLE速率：1Mbps、2Mbps
* 灵敏度-95 dBm @1Mbps BLE 模式
* 灵敏度-91.5 dBm @2Mbps BLE 模式
* 发射功率 -40 dBm 至 6 dBm
* 单端天线输出
* 集成balun
* 支持BLE Mesh
* RISC 32位 MCU
* 高性能 (64MHz)
* SMART 8 通道DMA控制器
* CACHE
* CACHE支持范围4KB
* USB
* USB1.1 全速达到12Mbps
* 最大支持5个端点，端点1~4支持同时收发
* 所有端点的FIFO深度64字节
* 片上存储器
* 4KB ROM
* 最大32KB SRAM
* SIP 128KB/256KB QSPI FLASH
* 时钟源
* 支持16MHz晶体振荡器(XTAL)
* 内部高频RC振荡器(IRC) 16MHz +/- 2%
* 内部低频RC振荡器的超低功率时钟(RC32K)
* 内部高频DPLL48M/64M时钟
* 支持外灌时钟
* 系统时钟(RTC)
* 采用超低功耗技术运行
* 由内部超低功耗RC运行
* 看门狗定时器
* 由内部低功率WDT运行
* 计数器/定时器模块
* 1路Advanced 16位计数器/定时器(ATMR)
* 1路Common 16位计数器/定时器(CTMR)
* 支持4个独立通道，用于输入捕获、输出比较、PWM
* 1路Basic 16位计数器/定时器(BTMR)
* 外设接口
* 2路UART，支持RS485/IrDA/ISO7816-3，速率可达3.84Mbps
* 1路SPI Master，最高速率可达16Mbps
* 1路SPI Slave，最高速率可达16Mbps
* 1路I2C总线，可配主/从设备，支持快速模式速率可达400Kbps
* 模拟-数字转换器(ADC)
* 10位精度，1Mbps ADC，10个输入通道
* 语音功能，支持8K采样率
* 通用I/O(GPIO)
* 最大支持20个GPIOs，每个IO都支持中断和唤醒功能
* 电源管理(PMU)
* 集成 Power-On-Reset (POR)
* 集成Low-Voltage-Detect (LVD) ，默认电压阈值1.65V
* 电压范围
* 工作电压 1.8V ~ 5.2V(由芯片型号定义)
* 电流功耗
* 片上LDO稳压器
* 44.7uA/MHz(@3.3V with 16M RC)
* Deep sleep 模式下，70 uA @3.3V
* Power off模式下，2uA @3.3V
* BLE/2.4G 模式下，Tx峰值电流7.2mA (@1.8V 0dBm)
* BLE/2.4G 模式下，Rx峰值电流10.3mA (@1.8V)
* ESD 4500V
* 工作温度-40℃ ~ 105℃

## 系统地址映射



图 1‑1 系统地址映射

系统中总线以及挂载在总线上的各存储器和外设位段分布(Size只代表预留的位置尺寸，不代表实际大小)。

## 系统框图



图 1‑2 系统框图

## 端口定义与复用信号

表 1-1端口定义与复用信号

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **PIN** | | | | **NAME** | **Default** | **Function 0** | **Function 1** | **Function 2** | **Function 3** | **Function Analog** |
| **eSOP**  **8** | **SOP**  **16** | **QFN**  **20** | **QFN**  **32** |  |  | **GPIO** | **CSC** | **SPECIAL** | **TIMER** | **Analog** |
|  | 16 |  |  | GND | -- | -- | -- | -- | -- | -- |
|  |  |  |  | VDD50 | -- | -- | -- | -- | -- | -- |
|  |  | 2 | 4 | VDD42 | -- | -- | -- | -- | -- | -- |
| 7 | 7 | 3 | 5,9 | VDD33① | -- | -- | -- | -- | -- | -- |
| 8 | 1 | 5 | 7 | RFP | -- | -- | -- | -- | -- | -- |
|  | 4 | 6 | 10 | RSTn/PA19 | RSTn | PA19 | CSC | -- | -- | -- |
|  |  | 7 | 11 | PA18 | GPIO | PA18 | CSC | -- | CTMR\_CH4 | -- |
|  | 5 | 8 | 12 | PA17 | GPIO | PA17 | CSC | -- | CTMR\_CH3 | -- |
|  |  | 9 | 13 | PA16 | GPIO | PA16 | CSC | -- | CTMR\_CH2 | ANAIO6 |
|  |  |  | 14 | PA15 | GPIO | PA15 | CSC | -- | CTMR\_CH1 | ANAIO5 |
|  |  |  | 15 | PA14 | GPIO | PA14 | CSC | -- | ATMR\_ETR | ANAIO4 |
|  | 6 | 10 | 16 | PA13 | GPIO | PA13 | CSC | -- | ATMR\_CH3N | ANAIO2 |
| 4 |  |  | 17 | PA12 | GPIO | PA12 | CSC | -- | ATMR\_CH2N | ANAIO1 |
|  |  |  | 18 | PA11 | GPIO | PA11 | CSC | -- | ATMR\_CH1N | ANAIO0 |
|  | 3 | 11 | 19 | PA10 | GPIO | PA10 | CSC | -- | ATMR\_CH4P | ANAIO9 |
|  | 2 | 12 | 20 | PA9 | GPIO | PA9 | CSC | -- | ATMR\_CH3P | ANAIO8 |
|  | 8 | 13 | 21 | PA8 | GPIO | PA8 | CSC | -- | ATMR\_CH2P | ANAIO7 |
| 2 | 9 | 14 | 22 | PA7/DM② | GPIO | PA7 | CSC | -- | ATMR\_CH1P | ANAIO6 |
| 3 | 10 | 15 | 23 | PA6/DP③ | GPIO | PA6 | CSC | -- | ATMR\_BK | ANAIO5 |
|  |  |  | 24 | PA5 | GPIO | PA5 | CSC | -- | CTMR\_CH4 | ANAIO4 |
|  |  |  | 25 | PA4 | GPIO | PA4 | CSC | -- | CTMR\_CH3 | ANAIO3 |
|  |  | 16 | 26 | PA3/MICIN | GPIO | PA3 | CSC | -- | CTMR\_CH2 | MICIN |
|  |  | 17 | 27 | PA2/MICBIAS④ | GPIO | PA2 | CSC | -- | CTMR\_CH1 | ANAIO2 |
|  | 11 | 18 | 28 | PA1/SWDIO | SWDIO | PA1 | CSC | SWDIO | CTMR\_ETR | ANAIO1 |
|  | 12 | 19 | 29 | PA0/SWCLK | SWCLK | PA0 | CSC | SWCLK | -- | ANAIO0 |
| 1 | 13 |  | 30 | VDD12 | -- | -- | -- | -- | -- | -- |
| 5 | 14 | 20 | 32 | XO16M\_O | -- | -- | -- | -- | -- | -- |
| 6 | 15 | 1 | 1 | XO16M\_I | -- | -- | -- | -- | -- | -- |

1. ***QFN32封装有两个VDD33 PIN：PIN5为VDD33\_1，PIN9为VDD33\_2，片内VDD33\_1与VDD33\_2连通***
2. ***USB DM功能的PAD与PA7绑定，使用DM功能时，需要将PA7配置为高阻状态***
3. ***USB DP功能的PAD与PA6绑定，使用DP功能时，需要将PA6配置为高阻状态***
4. ***语音MICBIAS功能的PAD与PA2绑定，使用MICBIAS功能时，需要将PA2配置为高阻状态***

表 1-2 定义说明

|  |  |
| --- | --- |
| **NAME** | **EXPLANATION** |
| PA00~PA19 | GPIOs (High Level) |
| SWCLK | Debug Clock Pin |
| SWDIO | Debug Data Pin (High Level) |
| DP | USB Data Positive |
| DM | USB Data Minus |
| MICBIAS | MIC output voltage |
| MICIN | MIC signal input |
| RFP | RF Antenna Posedge |
| XO16M\_O | Crystal oscillator 16M Output |
| XO16M\_I | Crystal oscillator 16M Input |
| CTMR\_CH1~ CTMR\_CH4 | 4 Channels Common Timer1 PWC Input / PWM Output |
| ATMR\_ETR /CTMR\_ETR | Advanced Timer/Common Timer External Trigger |
| ATMR\_BK | Advanced Timer1 Break Input |
| ATMR\_CH1P~ ATMR\_CH4P | 4 Channels Advanced Timer Positive PWC Input / PWM Output |
| CSC | GPIO Function Multiplexing |
| AINIO0~AINIO9 | ADC Input |
| VDD12 | Digital core voltage output 1.2V |
| VDD33 | voltage input 3.3V |
| VDD42 | voltage input 4.2V |
| VDD50 | voltage input 5.0V |

## 中断嵌套向量定义

### 中断优先级

中断优先级寄存器(Interrupt Priority Register)每个byte的高2位为有效位，支持4个中断优先级设置。

### 中断向量分配

中断向量分配如下表所示(**可设置的优先级默认为0-同级**)：

表 1‑1中断向量分配

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | **优先级** | **名称** | **说明** |
| 0 | -- | -- | 保留 |
| 1 | -14 | NMI\_IRQn | 不可屏蔽中断 |
| 2 | -13 | HardFault\_IRQn | 所有类型的错误 |
| 3 | -- | -- | 保留 |
| 4 | -- | -- | 保留 |
| 5 | -- | -- | 保留 |
| 6 | -- | -- | 保留 |
| 7 | -- | -- | 保留 |
| 8 | -- | -- | 保留 |
| 9 | -- | -- | 保留 |
| 10 | -- | -- | 保留 |
| 11 | -5 | SVCall\_IRQn | 通过 SWI 指令调用的系统服务 |
| 12 | -- | -- | 保留 |
| 13 | -- | -- | 保留 |
| 14 | -2 | PendSV\_IRQn | 可挂起的系统服务 |
| 15 | -1 | SysTick\_ IRQn | 系统定时器 |
| 16 | 可设置 | EXTI\_IRQn | 外部端口中断 |
| 17 | 可设置 | IWDT\_IRQn | IWDT中断 |
| 18 | 可设置 | BLE\_IRQn | BLE中断 |
| 19 | 可设置 | DMAC\_IRQn | DMA中断 |
| 20 | 可设置 | BB\_LP\_IRQn | BB WAKEUP中断 |
| 21 | 可设置 | BTMR\_IRQn | BTMR中断 |
| 22 | 可设置 | CTMR\_IRQn | CTMR中断 |
| 23 | 可设置 | ATMR\_IRQn | ATMR中断 |
| 24 | 可设置 | RTC\_IRQn | RTC中断 |
| 25 | 可设置 | I2C\_IRQn | I2C中断 |
| 26 | 可设置 | SPIM\_IRQn | SPIM中断 |
| 27 | 可设置 | SPIS\_IRQn | SPIS中断 |
| 28 | 可设置 | UART1\_IRQn | UART1中断 |
| 29 | 可设置 | UART2\_IRQn | UART2中断 |
| 30 | 可设置 | AON\_PMU\_IRQn | AON\_PMU中断 |
| 31 | 可设置 | LVD33\_IRQn | LVD33保护中断 |
| 32 | 可设置 | BOD12\_IRQn | BOD保护中断 |
| 33 | 可设置 | USB\_IRQn | USB中断 |
| 34 | 可设置 | USB\_SOF\_IRQn | USB\_SOF中断 |
| 35 | 可设置 | FSHC\_IRQn | FSHC中断 |
| 36 | 可设置 | MDM\_IRQn | MDM中断 |
| 37 | 可设置 | RF\_IRQn | RF中断 |

## 事件唤醒

微控制器支持事件唤醒机制：通过配置外设的中断控制寄存器使能一个中断，但在NVIC中不使能该中断（可通过设置PRIMASK和BASEPRI来禁止），并将内核的系统控制寄存器中的SEVONPEND位使能以允许中断事件唤醒WFI。当外设中断产生后，芯片从WFI唤醒。芯片唤醒后，软件需要清除相应外设的中断标志位和外设在NVIC中断通道上的挂起位。

表 1-3 事件唤醒源

|  |  |
| --- | --- |
| **事件唤醒源** | **描述** |
| GPIO ALL | 选择任意GPIOAxx作为唤醒源 |
| RTC | RTC中断事件 |
| BLE LPTIM | 蓝牙低功耗定时器 |

# 系统配置控制器(SYSCFG)

## 概述

## 寄存器映射

SYSCFG基地址：0x4000\_1000，大小4KB。

表 2-1 寄存器映射(SYSCFG)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 寄存器 | 偏移量 | 读/写 | 描述 | 复位值 |
| SYS\_BACKUP0 | 0x044 | R/W | 只有核电压掉电才能被清零  不受其他复位信号影响 | 0x0000 0000 |

### 系统备份寄存器(sys\_backup0)

表 2-2 系统备份寄存器

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

表 2-3 系统备份寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| USER | 31:16 | 用户自定义 |
| Reserved | 15:0 | 保留 |

# IO功能复用控制(CSC)

## 概述

IO引脚可复用为外设功能端口，例如PWM输出口或UART通信口，每个外设均支持复用到多个引脚上，任意IO端口支持外部中断。

* 可配置为输入或输出
* 输出模式可配置
* 推挽/开漏
* 上拉/下拉
* 输入模式
* 端口浮空
* 上拉/下拉
* 模拟端口
* 支持端口输出数据的复位、置位或取反，可按位操作
* 支持复用为外设功能端口
* 输出驱动能力可配置：两种驱动能力选择
* 支持20个外部输入中断、DMA功能
* 支持端口配置写保护功能

## 功能联接

* 灵活配置，SPI/UART/I2C功能PIN脚可配所有GPIOs

## 寄存器映射

### CSC输出控制寄存器映射

CSC基地址：0x4000\_2000，大小4KB。

表 3-1 输出控制寄存器映射(CSC)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **寄存器** | **偏移量** | **读/写** | **描述** | **复位值** |
| OUT\_CTRL0 | 0x0000 | R/W | CSC Pad Control Output Register for PIOA[00] | 0x0000 0000 |
| OUT\_CTRL1 | 0x0004 | R/W | CSC Pad Control Output Register for PIOA[01] | 0x0000 0000 |
| OUT\_CTRL2 | 0x0008 | R/W | CSC Pad Control Output Register for PIOA[02] | 0x0000 0000 |
| OUT\_CTRL3 | 0x000C | R/W | CSC Pad Control Output Register for PIOA[03] | 0x0000 0000 |
| OUT\_CTRL4 | 0x0010 | R/W | CSC Pad Control Output Register for PIOA[04] | 0x0000 0000 |
| OUT\_CTRL5 | 0x0014 | R/W | CSC Pad Control Output Register for PIOA[05] | 0x0000 0000 |
| OUT\_CTRL6 | 0x0018 | R/W | CSC Pad Control Output Register for PIOA[06] | 0x0000 0000 |
| OUT\_CTRL7 | 0x001C | R/W | CSC Pad Control Output Register for PIOA[07] | 0x0000 0000 |
| OUT\_CTRL8 | 0x0020 | R/W | CSC Pad Control Output Register for PIOA[08] | 0x0000 0000 |
| OUT\_CTRL9 | 0x0024 | R/W | CSC Pad Control Output Register for PIOA[09] | 0x0000 0000 |
| OUT\_CTRL10 | 0x0028 | R/W | CSC Pad Control Output Register for PIOA[10] | 0x0000 0000 |
| OUT\_CTRL11 | 0x002C | R/W | CSC Pad Control Output Register for PIOA[11] | 0x0000 0000 |
| OUT\_CTRL12 | 0x0030 | R/W | CSC Pad Control Output Register for PIOA[12] | 0x0000 0000 |
| OUT\_CTRL13 | 0x0034 | R/W | CSC Pad Control Output Register for PIOA[13] | 0x0000 0000 |
| OUT\_CTRL14 | 0x0038 | R/W | CSC Pad Control Output Register for PIOA[14] | 0x0000 0000 |
| OUT\_CTRL15 | 0x003C | R/W | CSC Pad Control Output Register for PIOA[15] | 0x0000 0000 |
| OUT\_CTRL16 | 0x0040 | R/W | CSC Pad Control Output Register for PIOA[16] | 0x0000 0000 |
| OUT\_CTRL17 | 0x0044 | R/W | CSC Pad Control Output Register for PIOA[17] | 0x0000 0000 |
| OUT\_CTRL18 | 0x0048 | R/W | CSC Pad Control Output Register for PIOA[18] | 0x0000 0000 |
| OUT\_CTRL19 | 0x004C | R/W | CSC Pad Control Output Register for PIOA[19] | 0x0000 0000 |

### CSC输入控制寄存器映射

CSC基地址：0x4000\_2000，大小4KB。

表 3-2 输入控制寄存器映射(CSC)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **寄存器** | **偏移量** | **读/写** | **描述** | **复位值** |
| IN\_CTRL0 | 0x080 | R/W | CSC Pad Control Input Register PIOA[00] | 0x0000 0000 |
| IN\_CTRL1 | 0x084 | R/W | CSC Pad Control Input Register PIOA[01] | 0x0000 0000 |
| IN\_CTRL2 | 0x088 | R/W | CSC Pad Control Input Register PIOA[02] | 0x0000 0000 |
| IN\_CTRL3 | 0x08C | R/W | CSC Pad Control Input Register PIOA[03] | 0x0000 0000 |
| IN\_CTRL4 | 0x090 | R/W | CSC Pad Control Input Register PIOA[04] | 0x0000 0000 |
| IN\_CTRL5 | 0x094 | R/W | CSC Pad Control Input Register PIOA[05] | 0x0000 0000 |
| IN\_CTRL6 | 0x098 | R/W | CSC Pad Control Input Register PIOA[06] | 0x0000 0000 |
| IN\_CTRL7 | 0x09C | R/W | CSC Pad Control Input Register PIOA[07] | 0x0000 0000 |
| IN\_CTRL8 | 0x0A0 | R/W | CSC Pad Control Input Register PIOA[08] | 0x0000 0000 |
| IN\_CTRL9 | 0x0A4 | R/W | CSC Pad Control Input Register PIOA[09] | 0x0000 0000 |
| IN\_CTRL10 | 0x0A8 | R/W | CSC Pad Control Input Register PIOA[10] | 0x0000 0000 |
| IN\_CTRL11 | 0x0AC | R/W | CSC Pad Control Input Register PIOA[11] | 0x0000 0000 |
| IN\_CTRL12 | 0x0B0 | R/W | CSC Pad Control Input Register PIOA[12] | 0x0000 0000 |
| IN\_CTRL13 | 0x0B4 | R/W | CSC Pad Control Input Register PIOA[13] | 0x0000 0000 |
| IN\_CTRL14 | 0x0B8 | R/W | CSC Pad Control Input Register PIOA[14] | 0x0000 0000 |
| IN\_CTRL15 | 0x0BC | R/W | CSC Pad Control Input Register PIOA[15] | 0x0000 0000 |
| IN\_CTRL16 | 0x0C0 | R/W | CSC Pad Control Input Register PIOA[16] | 0x0000 0000 |
| IN\_CTRL17 | 0x0C4 | R/W | CSC Pad Control Input Register PIOA[17] | 0x0000 0000 |
| IN\_CTRL18 | 0x0C8 | R/W | CSC Pad Control Input Register PIOA[18] | 0x0000 0000 |
| IN\_CTRL19 | 0x0CC | R/W | CSC Pad Control Input Register PIOA[19] | 0x0000 0000 |

### CSC控制寄存器映射

CSC控制基地址：0x4000\_2000，大小4KB。

表 3-3 控制寄存器映射(CSC)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 寄存器 | 偏移量 | 读/写 | 描述 | 复位值 |
| PIOA00\_CTRL | 0x0100 | R/W | Pad Control Register (Pad ctrl 00) PIOA[00] | 0x0000 0142 |
| PIOA01\_CTRL | 0x0104 | R/W | Pad Control Register (Pad ctrl 01) PIOA[01] | 0x0000 0182 |
| PIOA02\_CTRL | 0x0108 | R/W | Pad Control Register (Pad ctrl 02) PIOA[02] | 0x0000 0000 |
| PIOA03\_CTRL | 0x010C | R/W | Pad Control Register (Pad ctrl 03) PIOA[03] | 0x0000 0000 |
| PIOA04\_CTRL | 0x0110 | R/W | Pad Control Register (Pad ctrl 04) PIOA[04] | 0x0000 0000 |
| PIOA05\_CTRL | 0x0114 | R/W | Pad Control Register (Pad ctrl 05) PIOA[05] | 0x0000 0000 |
| PIOA06\_CTRL | 0x0118 | R/W | Pad Control Register (Pad ctrl 06) PIOA[06] | 0x0000 0180 |
| PIOA07\_CTRL | 0x011C | R/W | Pad Control Register (Pad ctrl 07) PIOA[07] | 0x0000 0180 |
| PIOA08\_CTRL | 0x0120 | R/W | Pad Control Register (Pad ctrl 08) PIOA[08] | 0x0000 0000 |
| PIOA09\_CTRL | 0x0124 | R/W | Pad Control Register (Pad ctrl 09) PIOA[09] | 0x0000 0000 |
| PIOA10\_CTRL | 0x0128 | R/W | Pad Control Register (Pad ctrl 10) PIOA[10] | 0x0000 0000 |
| PIOA11\_CTRL | 0x012C | R/W | Pad Control Register (Pad ctrl 11) PIOA[11] | 0x0000 0000 |
| PIOA12\_CTRL | 0x0130 | R/W | Pad Control Register (Pad ctrl 12) PIOA[12] | 0x0000 0000 |
| PIOA13\_CTRL | 0x0134 | R/W | Pad Control Register (Pad ctrl 13) PIOA[13] | 0x0000 0000 |
| PIOA14\_CTRL | 0x0138 | R/W | Pad Control Register (Pad ctrl 14) PIOA[14] | 0x0000 0000 |
| PIOA15\_CTRL | 0x013C | R/W | Pad Control Register (Pad ctrl 15) PIOA[15] | 0x0000 0000 |
| PIOA16\_CTRL | 0x0140 | R/W | Pad Control Register (Pad ctrl 16) PIOA[16] | 0x0000 0000 |
| PIOA17\_CTRL | 0x0144 | R/W | Pad Control Register (Pad ctrl 17) PIOA[17] | 0x0000 0000 |
| PIOA18\_CTRL | 0x0148 | R/W | Pad Control Register (Pad ctrl 18) PIOA[18] | 0x0000 0000 |
| PIOA19\_CTRL | 0x014C | R/W | Pad Control Register (Pad ctrl 19) PIOA[19] | 0x0000 0180 |

## 寄存器说明

### CSC寄存器

表 3-4 输出控制寄存器(out\_ctrlxx)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | | | | | CSCFOUT\_EN | -- | | CSCFOUT\_SEL | | | | |

表 3-5 输出控制寄存器(out\_ctrlxx)定义

|  |  |  |
| --- | --- | --- |
| 名称 | 比特 | 描述 |
| Reserved | 31:6 | 保留 |
| CSCFOUT\_EN | 7 | **外设输出功能使能**  0：禁用，1：使能 |
| CSCFOUT\_SEL | 4:0 | **CSC外设输出功能选择**  见表：[CSC0功能选择定义](#_CSC0功能选择定义) |

表 3-6 输入控制寄存器(in\_ctrlxx)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | | | | | CSCFIN\_EN | -- | | CSCFIN\_SEL | | | | |

表 3-7 输入控制寄存器(in\_ctrlxx)定义

|  |  |  |
| --- | --- | --- |
| 名称 | 比特 | 描述 |
| Reserved | 31:6 | 保留 |
| CSCFIN\_EN | 7 | 外设输入功能使能  0：禁用，1：使能 |
| CSCFIN\_SEL | 4:0 | CSC外设输入功能选择  见表：[CSC0功能选择定义](#_CSC0功能选择定义) |

### CSC控制寄存器

表 3-8 CSC控制寄存器(PIOAxx)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | ODE | AE | CE | IE | PUD | | DST | -- | | FSEL | | |

表 3-9 CSC控制寄存器(PIOAxx)定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:12 | 保留 |
| ODE | 11 | 开漏模式  0：禁用，1：使能 |
| AE | 10 | 模拟功能  0：禁用，1：使能 |
| CE | 9 | 电流源功能  0：禁用，1：使能 |
| IE | 8 | 输入模式使能位  0：禁用，1：使能 |
| PUD | 8:6 | 上下拉模式  0：无上下拉  1：下拉(65K，25℃)  2：上拉(65K，25℃)  3：无上下拉 |
| DST | 5 | IO驱动能力选择  0：3mA  1：6mA (MAX) |
| Reserved | 4:3 | 保留 |
| FSEL | 2:0 | IO映射功能选择寄存器  0 : Function 0 GPIO  1 : Function 1 CSC  2 : Function 3 SPECIAL  3 : Function 4 TIMER  ...:保留 |

## CSC功能选择定义

表 3-10 1.1 CSC功能选择定义

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **编号** | **CSC\_OUT** | **CSC\_IN** | **功能输出门值** | **功能输入门值** |
| 0 | UART1\_TXD(IO) | UART1\_TXD(IO) | ‘0’,OE Disabled | ‘1’ |
| 1 | UART1\_RXD(IO) | UART1\_RXD(IO) | ‘0’,OE Disabled | ‘1’ |
| 2 | UART2\_TXD(IO) | UART2\_TXD(IO) | ‘0’,OE Disabled | ‘1’ |
| 3 | UART2\_RXD(IO) | UART2\_RXD(IO) | ‘0’,OE Disabled | ‘1’ |
| 4 | I2C\_SCL(IO) | I2C0\_SCL(IO) | ‘0’,OE Disabled | ‘1’ |
| 5 | I2C\_SDA(IO) | I2C0\_SDA(IO) | ‘0’,OE Disabled | ‘1’ |
| 6 | CTMR\_CH1(IO) | CTMR\_CH1(IO) | ‘0’,OE Disabled | ‘1’ |
| 7 | CTMR\_CH2(IO) | CTMR\_CH2(IO) | ‘0’,OE Disabled | ‘1’ |
| 8 | UART1\_RTS(O) | UART1\_CTS(I) | ‘0’,OE Disabled | ‘1’ |
| 9 | UART2\_RTS(O) | UART2\_CTS(I) | ‘0’,OE Disabled | ‘1’ |
| 10 | UART1\_SCK(O) | SPIM\_MISO(I) | ‘0’,OE Disabled | ‘1’ |
| 11 | UART2\_SCK(O) | SPIS\_CLK(I) | ‘0’,OE Disabled | ‘1’ |
| 12 | SPIM\_CLK(O) | SPIS\_MOSI(I) | ‘0’,OE Disabled | ‘1’ |
| 13 | SPIM\_MOSI(O) | SPIS\_CSN(I) | ‘0’,OE Disabled | ‘1’ |
| 14 | SPIS\_MISO(O) | -- | ‘0’,OE Disabled | ‘1’ |
| 15~31 | 保留 | | | |

**注：CSC OUT功能配置：**CSC->CSC\_OUTPUT[pad].CSC\_FSEL= func;

**CSC IN功能配置：**CSC->CSC\_INPUT[func].CSC\_FSEL= pad;

pad：IO编号，func：CSC功能编号。

如：

1:配置PA00引脚为UART2的TXD输出功能，则(pad=0, func=2):

CSC->CSC\_OUTPUT[0].CSC\_FSEL= 2;

2:配置PA01引脚为UART2的RXD输入功能，则(pad=1, func=3):

CSC->CSC\_INPUT[3].CSC\_FSEL= 1。

# 串行外设接口(SPIM)

## 概述

串行外设接口（SPI）是一种同步串行数据通信协议，以全双工模式运行。由一个主设备和一个或多个从设备组成。主服务器提供SPI时钟，从服务器从主服务器接收SPI时钟。主操作或从操作

* 支持串行外设互连（SPI）主协议
* 支持SPI模式0、1、2、3（基于CPOL和CPHA）
* 支持全双工，支持单发或单收，8位数据传输
* 当内核工作频率降低时，SPI主频率速率以相同的比例降低
* 主模式速度可达16Mbps
* 接收和发送各自独立的4个Byte FIFO缓冲区
* 支持 DMA 传输
* 支持传输已完成的中断
* 支持可配置的MSB/LSB数据传输

## 寄存器映射

SPIM基地址：0x4000\_4000，大小4KB。

表 4-1 寄存器映射(SPI)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **寄存器** | **偏移量** | **读/写** | **描述** | **复位值** |
| RX\_DATA | 0x000 | R | 接收缓存 | 0x0000 0000 |
| TX\_DATA | 0x004 | W | 发送缓存 | 0x0000 0000 |
| TXRX\_BGN | 0x008 | W | 在双工模式或仅为RX模式时使用 | 0x0000 0000 |
| CTRL | 0x00C | R/W | 配置寄存器 | 0x0000 0800 |
| STATUS\_CLR | 0x010 | W | 状态清除寄存器 | 0x0000 0000 |
| STATUS | 0x014 | R | 状态寄存器 | 0x0000 0021 |
| DAT\_LEN | 0x018 | R/W | 数据长度 | 0x0000 0000 |

## 寄存器说明

### 接收寄存器(rx\_data)

表 4-2 接收寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | RX\_DATA | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

表 4-3 接收寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| RX\_DATA | 31:0 | 4 Bytes 接收缓存 |

### 发送寄存器(tx\_data)

表 4-4 发送寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | TX\_DATA | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

表 4-5 发送寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| TX\_DATA | 31:0 | 4 Bytes 发送缓存 |

### 启动寄存器(txrx\_bgn)

表 4-6 启动寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | TXRX\_BGN | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

表 4-7 启动寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| TXRX\_BGN | 31:0 | 在模块化模式或仅为RX模式时使用 |

### 控制寄存器(ctrl)

表 4-8 时钟控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | SPIM\_MSB\_FST | SPIM\_RX\_EN | SPIM\_TX\_EN | SPIM\_INT\_EN | SPIM\_RX\_DMA\_EN | SPIM\_\_DMA\_EN | SPITXM\_CPOL | SPIM\_CPHA | SPIM\_CRAT | | | |

表 4-9 时钟控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:12 | 保留 |
| spim\_msb\_fst | 11 | 1: MSB first  0: LSB first |
| spim\_rx\_en | 10 | Rx启用 |
| spim\_tx\_en | 9 | Tx启用 |
| spim\_int\_en | 8 | 中断启用位 |
| spim\_rx\_dma\_en | 7 | 0: RX作为mcu模式工作  1: RX作为dma模式工作，在配置dma cfg后，将txrx bgn设置为1，以启动dma传输 |
| spim\_tx\_dma\_en | 6 | 0: TX作为mcu模式工作  1: TX作为dma模式工作，在配置dma cfg后，将txrx bgn设置为1，以启动dma传输 |
| spim\_cpol | 5 | 时钟极性位  0: SPI\_CLK在空闲时为低电平  1: SPI\_CLK在空闲时为高电平 |
| spim\_cpha | 4 | 时钟相位  0: 在第一个时钟边沿采样数据  1: 在第二个时钟边沿采样数据 |
| spim\_crat | 3:0 | 时钟速率选择，系统时钟/2^（crat+1） 支持0-11 |

### 清除寄存器(status\_clr)

表 4-10 清除寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | | | | | | | | | | SPIM\_CLR\_INTF | SPIM\_RXDAT\_CLR | SPIM\_TXDAT\_CLR |

表 4-11 清除寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:3 | 保留 |
| SPIM\_CLR\_INTF | 2 | 写入1以清除intf状态 |
| SPIM\_RXDAT\_CLR | 1 | 设置1以清除spi RX fifo状态和内部计数器 |
| SPIM\_TXDAT\_CLR | 0 | 设置1以清除spi TX fifo状态和内部计数器 |

### 状态寄存器(status)

表 4-12 状态寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | SPIM\_BUSY | SPIM\_INTF | SPIM\_RX\_FNUM | | | SPIM\_RX\_FFULL | SPIM\_RX\_FEMPTY | SPIM\_TX\_FNUM | | | SPIM\_TX\_FFULL | SPIM\_TX\_FEMPTY |

表 4-13 状态寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:12 | 保留 |
| SPIM\_BUSY | 11 | spim传输信号 0: 空闲 1: 繁忙，tx/rx正在运行 |
| SPIM\_INTF | 10 | rx/tx中断标志 |
| SPIM\_RX\_FNUM | 9:7 | rx fifo中的字节数 |
| SPIM\_RX\_FFULL | 6 | rx fifo满 |
| SPIM\_RX\_FEMPTY | 5 | rx fifo空 |
| SPIM\_TX\_FNUM | 4:2 | tx fifo中的字节数 |
| SPIM\_TX\_FFULL | 1 | tx fifo满 |
| SPIM\_TX\_FEMPTY | 0 | tx fifo空 |

### 数据长度寄存器(dat\_len)

表 4-14 数据长度寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | DAT\_LEN | | | | | | | | | | | | | | | |

表 4-15 数据长度寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| DAT\_LEN | 15:0 | MCU或DMA模式下RX的数据长度 |

# 串行外设接口(SPIS)

## 概述

* 支持串行外设互连（SPI）从协议
* 支持SPI模式0、1、2、3（基于CPOL和CPHA）
* 支持全双工，8位数据传输
* 从模式速度可达16Mbps
* 接收和发送各自独立的4个Byte FIFO缓冲区
* 支持 DMA 传输
* 支持传输已完成的中断
* 支持可配置的MSB/LSB数据传输

## 寄存器映射

SPIS基地址：0x4000\_5000，大小4KB。

表 5-1 寄存器映射(SPI)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **寄存器** | **偏移量** | **读/写** | **描述** | **复位值** |
| CTRL | 0x000 | R/W | 配置寄存器 | 0x0000 0000 |
| STATUS | 0x004 | R | 状态寄存器 | 0x0000 0000 |
| INFO\_CLR | 0x008 | W | 状态清除寄存器 | 0x0000 0000 |
| TX\_DATA | 0x00C | W | 发送缓存 | 0x0000 0000 |
| RX\_DATA | 0x010 | R | 接收缓存 | 0x0000 0000 |

## 寄存器说明

### 控制寄存器(ctrl)

表 5-2 时钟控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | SPIS\_OTINT\_EN | SPIS\_OT\_EN | SPIS\_OT\_WIN | | | | | | | | SPIS\_EN | SPIS\_RX\_DMA\_MODE | SPIS\_TX\_DMA\_MODE | SPIS\_RXINT\_EN | SPIS\_RX\_EN | SPIS\_CPHA | SPIS\_CPOL | SPIS\_LSBFIRST |

表 5-3 时钟控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:12 | 保留 |
| SPIS\_OTINT\_EN | 17 | SPIS超时中断启用 |
| SPIS\_OT\_EN | 16 | SPIS超时启用 |
| SPIS\_OT\_WIN | 15:8 | 配置超时窗口  单位：位宽时间 |
| SPIS\_EN | 7 | 0: SPIS工作禁用  1: SPIS工作启用  [注：只有当SPIS\_EN为0时，才能配置SPIS相位、极性] |
| SPIS\_RX\_DMA\_MODE | 6 | SPIS RX运行模式控制信号  1: DMA模式  0: MCU模式 |
| SPIS\_TX\_DMA\_MODE | 5 | SPIS TX运行模式控制信号  1: DMA模式  0:MCU模式 |
| SPIS\_RXINT\_EN | 4 | 0: spis\_rxint中断禁用  1: spis\_rxint中断启用 |
| SPIS\_RX\_EN | 3 | spirx控制信号 默认为启用  0: 禁用spis RX  1: spis RX启用  SPIS RX仅在交换程序和交换程序都设置为1时启用 |
| SPIS\_CPHA | 2 | 时钟相位用于样本数据  0: 在第一个时钟边沿采样数据  1: 在第二个时钟边沿采样数据 |
| SPIS\_CPOL | 1 | 当SPIS为IDLE（cs为1）时，spi时钟极性控制位  0: SPI\_CLK在空闲时为低电平  1: SPI\_CLK在空闲时为高电平 |
| SPIS\_LSBFIRST | 0 | 0: MSB first  1: LSB first |

### 状态寄存器(status)

表 5-4 状态寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | SPIS\_OTINT\_ST | SPIS\_RXINT\_ST | RSV\_NOUSE2 | | | SPIS\_TXFIFO\_NUM | | | SPIS\_TXFIFO\_FULL | SPIS\_TXFIFO\_EMPTY | SPIS\_CS | SPIS\_RX\_OVERRUN | RSV\_NOUSE1 | SPIS\_RXFIFO\_NUM | | | SPIS\_RXFIFO\_FULL | SPIS\_RXFIFO\_EMPTY |

表 5-5 状态寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:18 | 保留 |
| SPIS\_OTINT\_ST | 17 | 当SPIS超时发生时，值为1 |
| SPIS\_RXINT\_ST | 16 | 当SPIS RX 1字节时，spis\_rxint为1 由spis\_rxint\_clr清除 |
| RSV\_NOUSE2 | 15:13 | 保留 |
| SPIS\_TXFIFO\_NUM | 12:10 | tx fifo中的字节数 |
| SPIS\_TXFIFO\_FULL | 9 | 0: spis\_txfifo非满  1: spis\_txfifo满 |
| SPIS\_TXFIFO\_EMPTY | 8 | 0: spis\_txfifo非空  1: spis\_txfifo空 |
| SPIS\_CS | 7 | 信号引脚SPIS\_CS  0: 选择  1: 未选择 |
| SPIS\_RX\_OVERRUN | 6 | SPIS rxfifo溢出错误信号  1: rxfifo溢出错误  0: 被SPIS rx溢出的clr清除 |
| RSV\_NOUSE1 | 5 | 保留 |
| SPIS\_RXFIFO\_NUM | 4:2 | rx fifo中的字节数 |
| SPIS\_RXFIFO\_FULL | 1 | 0: spis\_rxfifo非满  1: spis\_rxfifo满 |
| SPIS\_RXFIFO\_EMPTY | 0 | 0: spis\_rxfifo非空  1: spis\_rxfifo空 |

### 清除寄存器(info\_clr)

表 5-6 清除寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | | | | | | | | SPIS\_OTINT\_CLR | SPIS\_RXINT\_CLR | SPIS\_RXDAT\_CLR | SPIS\_TXDAT\_CLR | SPIS\_RX\_OVERRUN\_CLR |

表 5-7 清除寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:5 | 保留 |
| spis\_otint\_clr | 4 | 用于清除SPIS 超时中断标志 |
| spis\_rxint\_clr | 3 | 用于清除SPIS rx中断标志 |
| spis\_rxdat\_clr | 2 | 用于清除SPIS rxfifo点和数据记录 |
| spis\_txdat\_clr | 1 | 用于清除SPIS txfifo点和数据记录 |
| spis\_rx\_overrun\_clr | 0 | 用于清除spis\_rx\_overrun |

### 发送寄存器(tx\_data)

表 5-8 发送寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | TX\_DATA | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

表 5-9 发送寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| TX\_DATA | 31:0 | 4 Bytes 发送缓存 |

### 接收寄存器(rx\_data)

表 5-10 接收寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | RX\_DATA | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

表 5-11 接收寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| RX\_DATA | 31:0 | 4 Bytes 接收缓存 |

# 通用 I/O（GPIO）

## 概述

这些引脚可单独配置为输入或输出。每个引脚可额外地可配置为开漏输出或带滤波输入模式，配置为输出时可选择每个引脚的驱动强度。

* 20个可中断 GPIOs

## 结构框图

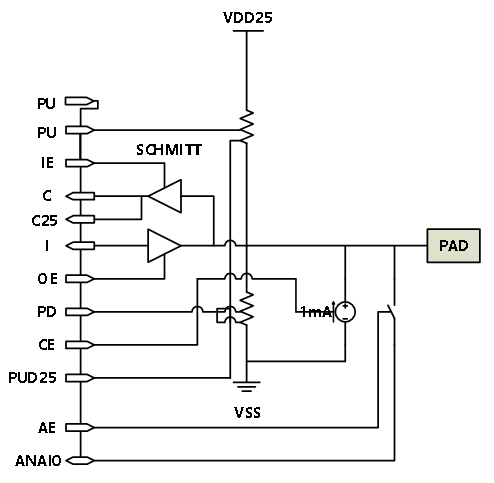


图 6‑1 GPIO结构框图(PBSCUDRT25)

## 寄存器映射

GPIO基地址：0x4001\_0000，大小4KB。

表 6-1 寄存器映射(GPIO)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 寄存器 | 偏移量 | 读/写 | 描述 | 复位值 |
| DAT\_SET | 0x0000 | R/W | GPIO数据集寄存器 | 0x0000 0000 |
| DAT\_CLR | 0x0004 | R/W | GPIO数据清除寄存器 | 0x0000 0000 |
| DAT\_TOG | 0x0008 | R/W | GPIO数据切换寄存器 | 0x0000 0000 |
| DAT\_MSK | 0x000C | R/W | GPIO数据掩码寄存器 | 0x0000 0000 |
| DAT | 0x0010 | R/W | GPIO数据寄存器 | 0x0000 0000 |
| PIN | 0x0014 | R | GPIO引脚寄存器 | 0x000F FFFF |
| … | … | … | 保留 | … |
| DIR\_SET | 0x0020 | R/W | GPIO方向集寄存器 | 0x0000 0000 |
| DIR\_CLR | 0x0024 | R/W | GPIO方向清除寄存器 | 0x0000 0000 |
| DIR\_TOG | 0x0028 | R/W | GPIO方向切换寄存器 | 0x0000 0000 |
| DIR\_MSK | 0x002C | R/W | GPIO方向掩码寄存器 | 0x0000 0000 |
| DIR | 0x0030 | R/W | GPIO方向寄存器 | 0x0000 0000 |
| … | … | … | 保留 | … |

## 寄存器说明

### IO数据设置寄存器(dat\_set)

表 6-2 IO数据设置寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | DAT\_SET | | | | | | | | | | | | | | | | | | | |

表 6-3 IO数据设置寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| DAT\_SET | 19:0 | GPIO数据设置寄存器  IO数据输出  0：无效，1：IO输出高电平 |

### IO数据清除寄存器(dat\_clr)

表 6-4 IO数据清除寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | DAT\_CLR | | | | | | | | | | | | | | | | | | | |

表 6-5 IO数据清除寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| DAT\_CLR | 19:0 | GPIO数据清除寄存器  IO数据清零  0：无效，1：IO输出低电平 |

### IO数据翻转寄存器(dat\_tog)

表 6-6 IO数据翻转寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | DAT\_TOG | | | | | | | | | | | | | | | | | | | |

表 6-7 IO数据翻转寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| DAT\_TOG | 19:0 | GPIO数据翻转寄存器  IO数据翻转  0：无效，1：切换数据寄存器的状态 |

### IO数据屏蔽寄存器(dat\_msk)

表 6-8 IO数据屏蔽寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | DAT\_MSK | | | | | | | | | | | | | | | | | | | |

表 6-9 IO数据屏蔽寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| DAT\_MSK | 19:0 | 0：无效  1：将某位写1将阻止set，clear，tog等任何操作。读取GPIO状态时，MSK设置为1的位段将只会返回0 |

### IO数据寄存器(dat)

表 6-10 IO数据寄存器

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

表 6-11 IO数据寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| DAT | 19:0 | GPIO数据寄存器  Set，clear，tog和mask的操作都将影响该寄存器位段 |

### IO状态寄存器(pin)

表 6-12 IO状态寄存器

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

表 6-13 IO状态寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| PIN | 19:0 | 反映当前IO的高低电平状态，不受MASK寄存器影响 |

### IO方向设置寄存器(dir\_set)

表 6-14 IO方向设置寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | DIR\_SET | | | | | | | | | | | | | | | | | | | |

表 6-15 IO方向设置寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| DIR\_SET | 19:0 | GPIO方向设置寄存器  0：无效，1：输出使能 |

### IO方向清除寄存器(dir\_clr)

表 6-16 IO方向清除寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | DIR\_CLR | | | | | | | | | | | | | | | | | | | |

表 6-17 IO方向清除寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| DIR\_CLR | 19:0 | GPIO方向清除寄存器  0：无效，1：输出禁用 |

### IO方向翻转寄存器(dir\_tog)

表 6-18 IO方向翻转寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | DIR\_TOG | | | | | | | | | | | | | | | | | | | |

表 6-19 IO方向翻转寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| DIR\_TOG | 19:0 | GPIO方向翻转寄存器  0：无效，1：切换方向寄存器状态 |

### IO方向屏蔽寄存器(dir\_msk)

表 6-20 IO方向屏蔽寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | DIR\_MSK | | | | | | | | | | | | | | | | | | | |

表 6-21 IO方向屏蔽寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| DIR\_MSK | 19:0 | GPIO方向屏蔽寄存器  0：无效，1：将某位写1将阻止set，clear，tog等任何操作。读取GPIO方向状态时，MSK设置为1的位段将只会返回0 |

### IO方向寄存器(dir)

表 6-22 IO方向寄存器

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

表 6-23 IO方向寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| DIR | 19:0 | GPIO方向寄存器  Set，clear，tog和mask的操作都将影响该寄存器位段 |

# 基本定时器(BASIC TIMER)

## 概述

1路基本16位计数器/定时器(BASIC TIMER，以下简称BTMR)。

基本定时器(BTMR)由一可编程预除器驱动之16位自动重载计数器构成.其亦可用于多种用途，包含测量输入信号脉冲长度(输入捕获)或产生输出波形(输出比较,PWM).可使用定时预除器与APB时钟控制预除器来调校脉冲长度与波形周期，由数微秒至数毫秒。

高级控制(ATMR),通用(CTMR)与基本(BTMR)定时器皆为完全独立不共享任何时钟源。

* 16 位递增自动重载计数器。
* 16 位可编程预分频器，用于对计数器时钟频率进行分频（即运行时修改），分频系数 介于 1 到 65536 之间。
* 同步化电路可控制定时器含外部信号与链接数个定时器
* 发生如下事件时生成中断/DMA 请求：
* 更新：计数器上溢/下溢，计数器初始化(透过软件或内部/外部触发)
* 外部时钟触发输入或cycle-by-cycle电流管理

## 结构框图



图 7‑1 基本定时器结构框图

## 寄存器映射

BTMR基地址：0x4002\_0000，大小4KB。

表 7-1 寄存器映射(BTMR)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **寄存器** | **偏移量** | **读/写** | **描述** | **复位值** |
| CR1 | 0x000 | R/W | 控制寄存器1 | 0x0000 0000 |
| CR2 | 0x004 | R/W | 控制寄存器2 | 0x0000 0000 |
| … | … | … | … | … |
| IER | 0x00C | W | 中断启用寄存器 | 0x0000 0000 |
| IDR | 0x010 | W | 中断禁用寄存器 | 0x0000 0000 |
| IVS | 0x014 | R | 中断有效状态寄存器 | 0x0000 0000 |
| RIF | 0x018 | R | 中断原始中断标志 | 0x0000 0000 |
| IFM | 0x01C | R | 中断屏蔽中断标志 | 0x0000 0000 |
| ICR | 0x020 | W | 中断清除状态寄存器 | 0x0000 0000 |
| EGR | 0x024 | R/W | 事件生成寄存器 | 0x0000 0000 |
| … | … | … | … | … |
| … | … | … | … | … |
| … | … | … | … | … |
| CNT | 0x034 | R/W | 定时计数 | 0x0000 0000 |
| PSC | 0x038 | R/W | 预定标器 | 0x0000 0000 |
| ARR | 0x03C | R/W | 自动重新加载寄存器 | 0x0000 0000 |
| … | … | … | … | … |
| … | … | … | … | … |
| … | … | … | … | … |
| … | … | … | … | … |
| … | … | … | … | … |
| … | … | … | … | … |
| DMAEN | 0x058 | R/W | DMA触发器事件启用 | 0x0000 0000 |

## 寄存器说明

### 控制寄存器1(cr1)

表 7-2 控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | | | | | ARPE | CMS | DIR | OPM | URS | UDIS | CEN | ARPE |

表 7-3 控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:14 | 保留 |
| ARPE | 7 | Auto-reload preload enable  0: TIMx\_ARR 寄存器没有缓存  1: TIMx\_ARR 寄存器有缓存 |
| CMS | 6:5 | Center-aligned mode selection  00: Edge-aligned mode. 计数器根据方向位(DIR)向上或向下计数.  01: Center-aligned mode 1. 计数器向上和向下计数可选。输出通道(TIMx\_CCMRx 寄存器的 CCxS=00)比较中断标志位仅在计数器递减时设置.  10: Center-aligned mode 2. 计数器向上和向下计数可选。输出通道(TIMx\_CCMRx 寄存器的 CCxS=00)比较中断标志位仅在计数器递增时设置.  11: Center-aligned mode 3. 计数器向上和向下计数可选。输出通道(TIMx\_CCMRx 寄存器的 CCxS=00)比较中断标志位在计数器递增/递减时设置.  注:不允许从边缘对齐模式切换到中心对齐模式，只要计数器是启用的(CEN=1). |
| DIR | 4 | Direction  0:计数器作为递增计数器  1:计数器作为递减计数器  注:此位在计时器配置为中心对齐模式或编码器模式时作为只读寄存器 |
| OPM | 3 | One pulse mode  0: 计数器不会停止计数在产生更新事件时  1: 计数器停止计数在产生下一次更新事件时(清除CEN位) |
| URS | 2 | Update request source  此位由软件设置并清除，以选择 UEV 事件源.  0: 如果使能，以下任何事件都会生成更新中断或 DMA 请求.这些事件可以是:  – 计数器溢出/下溢  – 设置 UG 位  – 更新事件生成通过从模式控制缓冲寄存器加载其预加载值  1: 如果使能，只有计数器溢出/下溢时才会生成更新中断或 DMA 请求. |
| UDIS | 1 | Update disable  此位由软件设置并清除，以启用/禁用 UEV 事件生成  0: UEV 使能. 更新(UEV)事件是由以下事件之一生成:  – 计数器溢出/下溢  – 设置 UG 位  – 更新事件生成通过从模式控制缓冲寄存器加载其预加载值  1: UEV 禁止.更新事件没有生成，影子寄存器将保持它们的值(ARR、 PSC、CCRx)。 然而计数器和预分频器将被重新初始化如果软件对 UG 位进行设置或者接收到硬件复位 |
| CEN | 0 | Counter enable  0:计数器禁止  1:计数器使能  注:外接时钟、门控模式和编码器模式只能在软件设置 CEN 位之后进行工作。而触发模式下硬件可以自动设置 CEN 位 |
| Reserved | 31:14 |  |

### 控制寄存器2(cr2)

表 7-4 控制寄存器2

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

表 7-5 控制寄存器2定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:15 | 保留 |
| MMS | 6:4 | Master mode selection  这些 Bit 位可以对主模式下发送给从计时器进行同步(TRGO)的信息进行选择。组合如下:  000: Reset -TIMx\_EGR 寄存器的 UG 信号作为 trigger 输出(TRGO)。 如果此复位是由 trigger 输入(在复位模式下配置从模式控制器)生成的， 那么TRGO 上的信号与实际复位相比会有延迟.  001: Enable -计数器使能信号 CNT\_EN 作为 trigger 输出(TRGO)。 这对于同时启动多个计时器或控制启用从计时器窗口是很有用的。计数器使能信号是在门控模式下由 CEN 控制位与 trigger 输入进行逻辑或生成的。当计数器使能信号由 trigger 输入控制时， TRGO 上有一个延迟，除非选择了主/从模式(参见 TIMx\_SMCR 寄存器中的 MSM 位描述).  010: Update –事件更新作为 trigger 输出（TRGO） 例如，主定时器可以用作从定时器的预分频器.  011: Compare Pulse – 一旦捕获或比较匹配产生， 在设置 CC1IF 标志位为1 的同时（即使它已经是高电平） trigger 输出将发送一个正脉冲（TRGO） .  100: Compare - OC1REF 信号作为 trigger 输出(TRGO)  101: Compare - OC2REF 信号作为 trigger 输出 (TRGO)  110: Compare - OC3REF 信号作为 trigger 输出(TRGO)  111: Compare - OC4REF 信号作为 trigger 输出 (TRGO) |
| Reserved | 3:0 | 保留 |

### 中断使能寄存器(ier)

表 7-6 中断使能寄存器

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

表 7-7 中断使能寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:2 | 保留 |
| UI | 0 | 更新中断使能  0：无效， 1：更新中断使能 |

### 中断关闭寄存器(idr)

表 7-8 中断关闭寄存器

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

表 7-9 中断关闭寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:2 | 保留 |
| UI | 0 | 更新中断禁止  0：无效， 1：更新中断禁止 |

### 中断有效状态寄存器(ivs)

表 7-10 中断有效状态寄存器

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

表 7-11 中断有效状态寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:2 | 保留 |
| UI | 0 | 更新中断有效状态  0：更新中断禁止状态， 1：更新中断使能状态 |

### 中断有效标志寄存器(rif)

表 7-12 中断有效标志寄存器

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

表 7-13 中断有效标志寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:2 | 保留 |
| UI | 0 | 更新中断标志  0：没有更新发生，  1：更新中断挂起。当寄存器有更新发生时，此位会被硬件置1  触发条件：  当CR1中UDIS=0，重复计数器值的溢出或下流（如果重复计数器=0则更新）  当CR1中的URS=0，UDIS=0时，CNT被软件使用timx\_egr寄存器中的ug位重新初始化  当CR1中的URS=0，UDIS=0时，CNT被一个触发器事件重新初始化 |

### 中断屏蔽寄存器(ifm)

表 7-14 中断屏蔽寄存器

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

表 7-15 中断屏蔽寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:2 | 保留 |
| UI | 0 | 更新中断标志屏蔽  当更新事件中断UI被使能时，此位是由硬件置1  0：没有更新发生，  1：更新中断挂起。当寄存器有更新发生时，此位会被硬件置1  触发条件：  1. 当CR1中UDIS=0，重复计数器值的溢出或下流（如果重复计数器=0则更新）  2. 当CR1中的URS=0，UDIS=0时，CNT被软件使用timx\_egr寄存器中的ug位重新初始化  3. 当CR1中的URS=0，UDIS=0时，CNT被一个触发器事件重新初始化 |

### 中断状态清除寄存器(icr)

表 7-16 中断状态清除寄存器

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

表 7-17 中断状态清除寄存器

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:2 | 保留 |
| UI | 0 | 更新中断清除  0：无效， 1：更新中断清除 |

### 事件产生寄存器(egr)

表 7-18 时钟配置(clkctrl)寄存器定义

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

表 7-19 时钟配置(stcalib)寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31：2 | 保留 |
| UG | 0 | 更新事件产生  这个位可以由软件设置，它是由硬件自动清除的  0：无效  1：重新初始化计数器并生成寄存器的更新。请注意，prescaler计数器也会被清除（无论如何，prescaler比例不受影响）。如果模式为中心对齐模式或dir=0（向上计数）时，CNT被清零，如果dir=1（向下计数）时，则取自动重加载值(timx\_arr)。 |

### 计数寄存器(cnt)

表 7-20 计数寄存器

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

表 7-21 计数寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| CNT | 15:0 | 计数值 |

### 预分频寄存器(psc)

表 7-22 预分频寄存器

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

表 7-23 预分频寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| PSC | 15:0 | 预分频 |

### 重载寄存器(arr)

表 7-24 重载寄存器

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

表 7-25 重载寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| ARR | 15:0 | ARR是要在实际的自动重新加载寄存器中加载的值。  当自动重新加载值为空时，计数器被阻止。 |

### DMA控制寄存器(dmaen)

表 7-26 DMA控制寄存器

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

表 7-27 DMA控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:7 | 保留 |
| UDE | 0 | Update DMA request enable  0: Update DMA 请求禁止.  1: Update DMA 请求使能 |

# 通用定时器(COMMON TIMER)

## 概述

1路通用16位计数器/定时器(COMMON TIMER，以下简称CTMR)，支持4个独立通道，用于输入捕获、输出比较、PWM。

通用定时器由一可编程预除器驱动之32/16位自动重载计数器构成.

其亦可用于多种用途，包含测量输入信号脉冲长度(输入捕获)或产生输出波形(输出比较,PWM)

可使用定时预除器与APB时钟控制预除器来调校脉冲长度与波形周期，由数微秒至数毫秒。

高级控制(ATMR),通用(CTMR)与基本(BTMR)定时器皆为完全独立不共享任何时钟源

* 32/16 位递增、递减、递增/递减自动重载计数器。
* 16位可编程预分频器，用于对计数器时钟频率进行分频（即运行时修改），分频系数介于 1 到 65536 之间。
* 多达 4 个独立通道，可用于：
* 输入捕获
* 输出比较
* PWM 生成（边沿和中心对齐模式）
* 单脉冲模式输出
* 同步化电路可控制定时器含外部信号与链接数个定时器。
* 发生如下事件时生成中断/DMA 请求：
* 更新：计数器上溢/下溢、计数器初始化（通过软件或内部/外部触发）
* 触发事件（计数器启动、停止、初始化或通过内部/外部触发计数）
* 输入捕获(捕获寄存器)
* 输出比较(计数寄存器配对比较寄存器)
* 支持定位用增量（正交）编码器和霍尔传感器电路。
* 外部时钟触发输入或cycle-by-cycle电流管理

## 结构框图



图 8‑1 通用定时器1块图

## 寄存器映射

CTMR基地址：0x4002\_1000，大小4KB。

表 8-1 寄存器映射(CTMR)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **寄存器** | **偏移量** | **读/写** | **描述** | **复位值** |
| CR1 | 0x000 | R/W | 控制寄存器1 | 0x0000 0000 |
| CR2 | 0x004 | R/W | 控制寄存器2 | 0x0000 0000 |
| SMCR | 0x008 | R/W | 从机模式控制寄存器 | 0x0000 0000 |
| IER | 0x00C | W | 中断启用寄存器 | 0x0000 0000 |
| IDR | 0x010 | W | 中断禁用寄存器 | 0x0000 0000 |
| IVS | 0x014 | R | 中断有效状态寄存器 | 0x0000 0000 |
| RIF | 0x018 | R | 中断原始中断标志 | 0x0000 0000 |
| IFM | 0x01C | R | 中断屏蔽中断标志 | 0x0000 0000 |
| ICR | 0x020 | W | 中断清除状态寄存器 | 0x0000 0000 |
| EGR | 0x024 | R/W | 事件生成寄存器 | 0x0000 0000 |
| CCMR1 | 0x028 | R/W | 捕获/比较模式寄存器1 | 0x0000 0000 |
| CCMR2 | 0x02C | R/W | 捕获/比较模式寄存器2 | 0x0000 0000 |
| CCER | 0x030 | R/W | 捕获/比较启用寄存器 | 0x0000 0000 |
| CNT | 0x034 | R/W | 定时计数 | 0x0000 0000 |
| PSC | 0x038 | R/W | 预定标器 | 0x0000 0000 |
| ARR | 0x03C | R/W | 自动重新加载寄存器 | 0x0000 0000 |
| RCR | 0x040 | R/W | 重复计数器寄存器 | 0x0000 0000 |
| CCR1 | 0x044 | R/W | 捕获/比较寄存器1 | 0x0000 0000 |
| CCR2 | 0x048 | R/W | 捕获/比较寄存器2 | 0x0000 0000 |
| CCR3 | 0x04C | R/W | 捕获/比较寄存器3 | 0x0000 0000 |
| CCR4 | 0x050 | R/W | 捕获/比较寄存器4 | 0x0000 0000 |
| … | … | … | … | … |
| DMAEN | 0x058 | R/W | DMA触发器事件启用 | 0x0000 0000 |

## 寄存器说明

### 控制寄存器1(cr1)

表 8-2 控制寄寄存器1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | | | | CKD | ARPE | CMS | | DIR | OPM | URS | UDIS | CEN |

表 8-3 控制寄寄存器1定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:14 | 保留 |
| CKD | 9:8 | Clock division  此位域表示定时器时钟(CK\_INT)频率与死区时间发生器、 数字滤波器(ETR,  TIx)使用的采样时钟(tDTS)之间的除法比  00: tDTS=tCK\_INT  01: tDTS=2\*tCK\_INT  10: tDTS=4\*tCK\_INT  11: 保留 |
| ARPE | 7 | Auto-reload preload enable  0: TIMx\_ARR 寄存器没有缓存  1: TIMx\_ARR 寄存器有缓存 |
| CMS | 6:5 | Center-aligned mode selection  00: Edge-aligned mode. 计数器根据方向位(DIR)向上或向下计数.  01: Center-aligned mode 1. 计数器向上和向下计数可选。输出通道(TIMx\_CCMRx 寄存器的 CCxS=00)比较中断标志位仅在计数器递减时设置.  10: Center-aligned mode 2. 计数器向上和向下计数可选。输出通道(TIMx\_CCMRx 寄存器的 CCxS=00)比较中断标志位仅在计数器递增时设置.  11: Center-aligned mode 3. 计数器向上和向下计数可选。输出通道(TIMx\_CCMRx 寄存器的 CCxS=00)比较中断标志位在计数器递增/递减时设置.  注:不允许从边缘对齐模式切换到中心对齐模式，只要计数器是启用的(CEN=1). |
| DIR | 4 | Direction  0:计数器作为递增计数器  1:计数器作为递减计数器  注:此位在计时器配置为中心对齐模式或编码器模式时作为只读寄存器 |
| OPM | 3 | One pulse mode  0: 计数器不会停止计数在产生更新事件时  1: 计数器停止计数在产生下一次更新事件时(清除CEN位) |
| URS | 2 | Update request source  此位由软件设置并清除，以选择 UEV 事件源.  0: 如果使能，以下任何事件都会生成更新中断或 DMA 请求.这些事件可以是:  – 计数器溢出/下溢  – 设置 UG 位  – 更新事件生成通过从模式控制缓冲寄存器加载其预加载值  1: 如果使能，只有计数器溢出/下溢时才会生成更新中断或 DMA 请求. |
| UDIS | 1 | Update disable  此位由软件设置并清除，以启用/禁用 UEV 事件生成  0: UEV 使能. 更新(UEV)事件是由以下事件之一生成:  – 计数器溢出/下溢  – 设置 UG 位  – 更新事件生成通过从模式控制缓冲寄存器加载其预加载值  1: UEV 禁止.更新事件没有生成，影子寄存器将保持它们的值(ARR、 PSC、CCRx)。 然而计数器和预分频器将被重新初始化如果软件对 UG 位进行设置或者接收到硬件复位 |
| CEN | 0 | Counter enable  0:计数器禁止  1:计数器使能  注:外接时钟、门控模式和编码器模式只能在软件设置 CEN 位之后进行工作。而触发模式下硬件可以自动设置 CEN 位 |

### 控制寄存器2(cr2)

表 8-4 控制寄存器2

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | | | | | TI1S | MMS | | | CCDS | CCUS | -- | CCPC |

表 8-5 控制寄存器2定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:15 | 保留 |
| TI1S | 7 | TI1 selection  0: TIMx\_CH1 作为 TI1 输入源  1: TIMx\_CH1, CH2和CH3的异或逻辑输出作为 TI1 输入源 |
| MMS | 6:4 | Master mode selection  这些 Bit 位可以对主模式下发送给从计时器进行同步(TRGO)的信息进行选择。组合如下:  000: Reset -TIMx\_EGR 寄存器的 UG 信号作为 trigger 输出(TRGO)。 如果此复位是由 trigger 输入(在复位模式下配置从模式控制器)生成的， 那么TRGO 上的信号与实际复位相比会有延迟.  001: Enable -计数器使能信号 CNT\_EN 作为 trigger 输出(TRGO)。 这对于同时启动多个计时器或控制启用从计时器窗口是很有用的。计数器使能信号是在门控模式下由 CEN 控制位与 trigger 输入进行逻辑或生成的。当计数器使能信号由 trigger 输入控制时， TRGO 上有一个延迟，除非选择了主/从模式(参见 TIMx\_SMCR 寄存器中的 MSM 位描述).  010: Update –事件更新作为 trigger 输出（TRGO） 例如，主定时器可以用作从定时器的预分频器.  011: Compare Pulse – 一旦捕获或比较匹配产生， 在设置 CC1IF 标志位为1 的同时（即使它已经是高电平） trigger 输出将发送一个正脉冲（TRGO） .  100: Compare - OC1REF 信号作为 trigger 输出(TRGO)  101: Compare - OC2REF 信号作为 trigger 输出 (TRGO)  110: Compare - OC3REF 信号作为 trigger 输出(TRGO)  111: Compare - OC4REF 信号作为 trigger 输出 (TRGO) |
| CCDS | 3 | 保留 |
| CCUS | 2 | Capture/compare control update selection  0: 当捕获/比较控制位预加载时(CCPC=1)， 它们仅通过设置 COMG Bit 位进行更新  1: 当捕获/比较控制位预加载(CCPC=1)时， 它们通过设置 COMG Bit 位或遇到 TRGI 上升沿进行更新. |
| Reserved | 1 | 保留 |
| CCPC | 0 | Capture/compare preloaded control  0: CCxE, CCxNE 和 OCxM bit 位没有预加载  1: CCxE, CCxNE 和 OCxM Bit 位是预加载的，在写入之后，只有在发生通信事件(COM)时才会更新(根据 CCUS Bit 位的不同，设置 COMG Bit 位或者检测 TRGI 上升边).  注意:这个 Bit 只作用于具有互补输出的通道 |

### 从机模式控制寄存器(smcr)

表 8-6 从机模式控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | | | | | MSM | TS | | | SMS | | | |

表 8-7 从机模式控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| MSM | 7 | Master/slave mode |
| TS | 6:4 | Trigger selection  此位字段选择要用于同步计数器的触发器输入  000: Internal Trigger 0 (ITR0)  001: Internal Trigger 1 (ITR1)  010: Internal Trigger 2 (ITR2)  011: Internal Trigger 3 (ITR3)  100: TI1 Edge Detector (TI1F\_ED)  101: Filtered Timer Input 1 (TI1FP1)  110: Filtered Timer Input 2 (TI2FP2)  111: External Trigger input (ETRF)  注意：这些位只有在不使用时才可以更改  SMS=000)，以避免在过渡时的错误边缘检测。 |
| Reserved | 3 | 保留 |

### 中断使能寄存器(ier)

表 8-8 中断使能寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | CC4OI | CC3OI | CC2OI | CC1OI | - | BI | TI | COMI | CC4I | CC3I | CC2I | CC1I | UI |

表 8-9 中断使能寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:13 | 保留 |
| CC4OI | 12 | 通道4捕获/比较过捕(overcapture)中断使能  0：无效， 1：CC4 overcapture中断使能 |
| CC3OI | 11 | 通道3捕获/比较过捕(overcapture)中断使能  0：无效， 1：CC3 overcapture中断使能 |
| CC2OI | 10 | 通道2捕获/比较过捕(overcapture)中断使能  0：无效， 1：CC2 overcapture中断使能 |
| CC1OI | 9 | 通道1捕获/比较过捕(overcapture)中断使能  0：无效， 1：CC1 overcapture中断使能 |
| Reserved | 8 | 保留 |
| BI | 7 | BREAK中断使能  0：无效， 1：BREAK中断使能 |
| TI | 6 | 触发中断使能  0：无效， 1：触发中断使能 |
| COMI | 5 | COM中断使能  0：无效， 1：COM中断使能 |
| CC4I | 4 | 通道4捕获/比较中断使能  0：无效， 1：CC4中断使能 |
| CC3I | 3 | 通道3捕获/比较中断使能  0：无效， 1：CC3中断使能 |
| CC2I | 2 | 通道2捕获/比较中断使能  0：无效， 1：CC2中断使能 |
| CC1I | 1 | 通道1捕获/比较中断使能  0：无效， 1：CC1中断使能 |
| UI | 0 | 更新中断使能  0：无效， 1：更新中断使能 |

### 中断关闭寄存器(idr)

表 8-10 中断关闭寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | CC4OI | CC3OI | CC2OI | CC1OI | - | BI | TI | COMI | CC4I | CC3I | CC2I | CC1I | UI |

表 8-11 中断关闭寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:13 | 保留 |
| CC4OI | 12 | 通道4捕获/比较过捕(overcapture)中断禁止  0：无效， 1：CC4 overcapture中断禁止 |
| CC3OI | 11 | 通道3捕获/比较过捕(overcapture)中断禁止  0：无效， 1：CC3 overcapture中断禁止 |
| CC2OI | 10 | 通道2捕获/比较过捕(overcapture)中断禁止  0：无效， 1：CC2 overcapture中断禁止 |
| CC1OI | 9 | 通道1捕获/比较过捕(overcapture)中断禁止  0：无效， 1：CC1 overcapture中断禁止 |
| Reserved | 8 | 保留 |
| BI | 7 | BREAK中断禁止  0：无效， 1：BREAK中断禁止 |
| TI | 6 | 触发中断禁止  0：无效， 1：触发中断禁止 |
| COMI | 5 | COM中断禁止  0：无效， 1：COM中断禁止 |
| CC4I | 4 | 通道4捕获/比较中断禁止  0：无效， 1：CC4中断禁止 |
| CC3I | 3 | 通道3捕获/比较中断禁止  0：无效， 1：CC3中断禁止 |
| CC2I | 2 | 通道2捕获/比较中断禁止  0：无效， 1：CC2中断禁止 |
| CC1I | 1 | 通道1捕获/比较中断禁止  0：无效， 1：CC1中断禁止 |
| UI | 0 | 更新中断禁止  0：无效， 1：更新中断禁止 |

### 中断有效状态寄存器(ivs)

表 8-12 中断有效状态寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | CC4OI | CC3OI | CC2OI | CC1OI | - | BI | TI | COMI | CC4I | CC3I | CC2I | CC1I | UI |

表 8-13 中断有效状态寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:13 | 保留 |
| CC4OI | 12 | 通道4捕获/比较过捕(overcapture)中断有效状态  0：CC4 overcapture中断禁止状态，  1：CC4 overcapture中断使能状态 |
| CC3OI | 11 | 通道3捕获/比较过捕(overcapture)中断有效状态  0：CC3 overcapture中断禁止状态，  1：CC3 overcapture中断使能状态 |
| CC2OI | 10 | 通道2捕获/比较过捕(overcapture)中断有效状态  0：CC2 overcapture中断禁止状态，  1：CC2 overcapture中断使能状态 |
| CC1OI | 9 | 通道1捕获/比较过捕(overcapture)中断有效状态  0：CC1 overcapture中断禁止状态，  1：CC1 overcapture中断使能状态 |
| Reserved | 8 | 保留 |
| BI | 7 | BREAK中断有效状态  0：BREAK中断禁止状态， 1：BREAK中断使能状态 |
| TI | 6 | 触发中断有效状态  0：触发中断禁止状态， 1：触发中断使能状态 |
| COMI | 5 | COM中断有效状态  0：COM中断禁止状态， 1：COM中断使能状态 |
| CC4I | 4 | 通道4捕获/比较中断有效状态  0：CC4中断禁止状态， 1：CC4中断使能状态 |
| CC3I | 3 | 通道3捕获/比较中断有效状态  0：CC3中断禁止状态， 1：CC3中断使能状态 |
| CC2I | 2 | 通道2捕获/比较中断有效状态  0：CC2中断禁止状态， 1：CC2中断使能状态 |
| CC1I | 1 | 通道1捕获/比较中断有效状态  0：CC1中断禁止状态， 1：CC1中断使能状态 |
| UI | 0 | 更新中断有效状态  0：更新中断禁止状态， 1：更新中断使能状态 |

### 中断有效标志寄存器(rif)

表 8-14 中断有效标志寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | CC4OI | CC3OI | CC2OI | CC1OI | - | BI | TI | COMI | CC4I | CC3I | CC2I | CC1I | UI |

表 8-15 中断有效标志寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:13 | 保留 |
| CC4OI | 12 | 通道4捕获/比较过捕(overcapture)中断标志  参考CC1OI |
| CC3OI | 11 | 通道3捕获/比较过捕(overcapture)中断标志  参考CC1OI |
| CC2OI | 10 | 通道2捕获/比较过捕(overcapture)中断标志  参考CC1OI |
| CC1OI | 9 | 通道1捕获/比较过捕(overcapture)中断标志  只有在输入捕获模式下配置相应的通道时，硬件才会设置此标志  0：没有检测到overcapture中断  1：当CC1I标志置1时且计数值已经被CCR1寄存器捕获 |
| Reserved | 8 | 保留 |
| BI | 7 | BREAK中断标志  如果启用了中断，则一旦break输入产生，硬件就会设置此标志。  0：没有检测到break事件  1：检测到有活跃的break输入 |
| TI | 6 | 触发中断标志  这个标志是由硬件对触发事件设置的(当从模式控制器在所有模式下启用门控模式时，检测到trgi输入，但如果启用了中断模式，则检测到活动边缘  0：没有发出事件发生  1：触发中断挂起 |
| COMI | 5 | COM中断标志  这个标志是由COM事件上的硬件设置的控制位-CcxE，CCxNE，OCxM-已经更新)  0：没有检测到COM事件  1：COM中断挂起 |
| CC4I | 4 | 通道4捕获/比较中断标志  参考CC1I |
| CC3I | 3 | 通道3捕获/比较中断标志  参考CC1I |
| CC2I | 2 | 通道2捕获/比较中断标志  参考CC1I |
| CC1I | 1 | 通道1捕获/比较中断标志  通道1被配置为输出模式：  当计数器与比较值匹配时，此标志由硬件设置，如果启用了中断，则在中心对齐模式下有些例外(请参阅timx\_cr1寄存器描述中的cms位)。在timx\_icr中写入1来清除原始中断  0：没有匹配  1：计数器timx\_cnt的内容与timx\_ccr1寄存器的内容相匹配。在计数器溢出（向上计数和上/下计数模式）或欠流（向下计数模式）模式时，如果CCR1大于ARR时，此位置1  通道1被配置为输入模式：  这个位是由捕获上的硬件设置的。通过软件或读取timx\_ccr1寄存器进行清除  0：没有输入捕获发生  1：计数器值已在timx\_ccr1寄存器中捕获(在ic1上检测到一条与所选极性匹配的边缘) |
| UI | 0 | 更新中断标志  0：没有更新发生，  1：更新中断挂起。当寄存器有更新发生时，此位会被硬件置1  触发条件：  当CR1中UDIS=0，重复计数器值的溢出或下流（如果重复计数器=0则更新）  当CR1中的URS=0，UDIS=0时，CNT被软件使用timx\_egr寄存器中的ug位重新初始化  当CR1中的URS=0，UDIS=0时，CNT被一个触发器事件重新初始化 |

### 中断屏蔽寄存器(ifm)

表 8-16 中断屏蔽寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | CC4OI | CC3OI | CC2OI | CC1OI | - | BI | TI | COMI | CC4I | CC3I | CC2I | CC1I | UI |

表 8-17 中断屏蔽寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:13 | 保留 |
| CC4OI | 12 | 通道4捕获/比较过捕(overcapture)中断标志屏蔽  参考CC1OI |
| CC3OI | 11 | 通道3捕获/比较过捕(overcapture)中断标志屏蔽  参考CC1OI |
| CC2OI | 10 | 通道2捕获/比较过捕(overcapture)中断标志屏蔽  参考CC1OI |
| CC1OI | 9 | 通道1捕获/比较过捕(overcapture)中断标志屏蔽  只有在输入捕获模式下配置相应的通道时，硬件才会设置此标志  0：没有检测到overcapture中断  1：当CC1I标志置1时且计数值已经被CCR1寄存器捕获 |
| Reserved | 8 | 保留 |
| BI | 7 | BREAK中断标志屏蔽  如果启用了中断，则一旦break输入产生，硬件就会设置此标志。  0：没有检测到break事件  1：检测到有活跃的break输入 |
| TI | 6 | 触发中断标志屏蔽  这个标志是由硬件对触发事件设置的(当从模式控制器在所有模式下启用门控模式时，检测到trgi输入，但如果启用了中断模式，则检测到活动边缘  0：没有发出事件发生  1：触发中断挂起 |
| COMI | 5 | COM中断标志屏蔽  这个标志是由COM事件上的硬件设置的控制位-CcxE，CCxNE，OCxM-已经更新)  0：没有检测到COM事件  1：COM中断挂起 |
| CC4I | 4 | 通道4捕获/比较中断标志屏蔽  参考CC1I |
| CC3I | 3 | 通道3捕获/比较中断标志屏蔽  参考CC1I |
| CC2I | 2 | 通道2捕获/比较中断标志屏蔽  参考CC1I |
| CC1I | 1 | 通道1捕获/比较中断标志屏蔽  通道1被配置为输出模式：  当计数器与比较值匹配时，此标志由硬件设置，如果启用了中断，则在中心对齐模式下有些例外(请参阅timx\_cr1寄存器描述中的cms位)。在timx\_icr中写入1来清除原始中断  0：没有匹配  1：计数器timx\_cnt的内容与timx\_ccr1寄存器的内容相匹配。在计数器溢出（向上计数和上/下计数模式）或欠流（向下计数模式）模式时，如果CCR1大于ARR时，此位置1  通道1被配置为输入模式：  这个位是由捕获上的硬件设置的。通过软件或读取timx\_ccr1寄存器进行清除  0：没有输入捕获发生  1：计数器值已在timx\_ccr1寄存器中捕获(在ic1上检测到一条与所选极性匹配的边缘) |
| UI | 0 | 更新中断标志屏蔽  当更新事件中断UI被使能时，此位是由硬件置1  0：没有更新发生，  1：更新中断挂起。当寄存器有更新发生时，此位会被硬件置1  触发条件：  1. 当CR1中UDIS=0，重复计数器值的溢出或下流（如果重复计数器=0则更新）  2. 当CR1中的URS=0，UDIS=0时，CNT被软件使用timx\_egr寄存器中的ug位重新初始化  3. 当CR1中的URS=0，UDIS=0时，CNT被一个触发器事件重新初始化 |

### 中断状态清除寄存器(icr)

表 8-18 中断状态清除寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | CC4OI | CC3OI | CC2OI | CC1OI | - | BI | TI | COMI | CC4I | CC3I | CC2I | CC1I | UI |

表 8-19 中断状态清除寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:13 | 保留 |
| CC4OI | 12 | 通道4捕获/比较过捕(overcapture)中断清除  0：无效， 1：CC4 overcapture中断清除 |
| CC3OI | 11 | 通道3捕获/比较过捕(overcapture)中断清除  0：无效， 1：CC3 overcapture中断清除 |
| CC2OI | 10 | 通道2捕获/比较过捕(overcapture)中断清除  0：无效， 1：CC2 overcapture中断清除 |
| CC1OI | 9 | 通道1捕获/比较过捕(overcapture)中断清除  0：无效， 1：CC1 overcapture中断清除 |
| Reserved | 8 | 保留 |
| BI | 7 | BREAK中断清除  0：无效， 1：BREAK中断清除 |
| TI | 6 | 触发中断清除  0：无效， 1：触发中断清除 |
| COMI | 5 | COM中断清除  0：无效， 1：COM中断清除 |
| CC4I | 4 | 通道4捕获/比较中断清除  0：无效， 1：CC4中断清除 |
| CC3I | 3 | 通道3捕获/比较中断清除  0：无效， 1：CC3中断清除 |
| CC2I | 2 | 通道2捕获/比较中断清除  0：无效， 1：CC2中断清除 |
| CC1I | 1 | 通道1捕获/比较中断清除  0：无效， 1：CC1中断清除 |
| UI | 0 | 更新中断清除  0：无效， 1：更新中断清除 |

### 事件产生寄存器(egr)

表 8-20 事件产生寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | | | | | | TG | COMG | CC4G | CC3G | CC2G | CC1G | UG |

表 8-21 事件产生寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31：8 | 保留 |
| TG | 6 | 触发事件产生  0：无效  1：TIF标志已在TIMx\_RIF寄存器中设置。相关中断或DMA如果启用了传输，就可以发生传输 |
| COMG | 5 | 捕获/比较控制更新生成  0：无效  1：当设置了CCPC位时，它允许更新CCxE、CCxNE和OCxM位。注意：这个位只作用于具有互补输出的通道。 |
| CC4G | 4 | 通道4捕获/比较事件产生  参考CC1G |
| CC3G | 3 | 通道3捕获/比较事件产生  参考CC1G |
| CC2G | 2 | 通道2捕获/比较事件产生  参考CC1G |
| CC1G | 1 | 通道1捕获/比较事件产生  0：无效  1：在通道1上生成了一个捕获/比较事件  通道1被配置为输出模式：  有中断或者DMA请求时，此位置1  通道1被配置为输入模式：  有输入捕获事件发生时，此位置1；如果此时CC1I已经为1，将CC1OI置1 |

### 捕获/比较模式寄存器1(ccmr1)

OUTPUT:

表 8-22 捕获/比较寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | OC2CE | OC2M | | | OC2PE | OC2FE | CC2S | | OC1CE | OC1M | | | OC1PE | OC1FE | CC1S | |

表 8-23 捕获/比较寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| OC2CE | 15 | 通道2输出比较清除使能  参考OC1CE |
| OC2M | 14:12 | 通道2输出模式  参考OC1M |
| OC2PE | 11 | 通道2输出比较重载使能  参考OC1PE |
| OC2FE | 20 | 通道2输出比较快速使能  参考OC1FE |
| CC2S | 9:8 | 通道2捕获/比较选择  00: CC2 通道配置成输出  01: CC2 通道配置成输入, TI2 作为 IC2 的输入  10: CC2 通道配置成输入, TI1 作为 IC2 的输入  11: CC2 通道配置成输入, TRC 作为 IC2 的输入  只有通过 TS 位(TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |
| OC1CE | 7 | 通道1输出比较清除使能  0：OC1Ref不受ETRF输入的影响  1：一旦在ETRF输入上检测到一个高水平，OC1Ref就会被清除。 |
| OC1M | 6:4 | 通道1输出模式  0：输出比较寄存器timx\_ccr1和计数器timx\_cnt之间的比较对输出没有影响（这种模式用于生成定时基）  1：在匹配时将通道1设置为活动级别。当计数器timx\_cnt与捕获/比较寄存器1(timx\_ccr1)相匹配时，Oc1ref信号被强制变高。  2：在匹配时将通道1设置为非活动级别。当计数器timx\_cnt与捕获/比较寄存器1(timx\_ccr1)相匹配时，oc1ref信号被迫变低  3：oc1ref在timx\_cnt=timx\_ccr1翻转  4：oc1ref被强制低  5：oc1ref被强制高  6：在向上计数中，只要其他timx\_cnttimx\_ccr1是活跃的(oc1ref=1)。  7：在向上计数中，只要timx\_cnttimx\_ccr1是不活动的，通道1就是活动的。 |
| OC1PE | 3 | 通道1输出比较重载使能  0：禁用CCR1的preload。任何时候更新CCR1值都会被马上重载。  1：使能CCR1的preload。CCR1的重载值只有在发生更新事件后才会被重载。 |
| OC1FE | 2 | 通道1输出比较快速使能  0：CC1的行为通常取决于计数器和CCR1的值。当触发器打开时。使能CC1输出的最小延迟时间为一个边沿触发输入是5个时钟周期  1：触发器输入上的活跃边沿类似于CC1上的比较匹配产量然后，将OC设置为独立于结果的比较水平  的比较。延迟采样触发器输入并使能CC1输出被减少到3个时钟周期。OCFE仅在通道配置为PWM1或PWM是才活跃 |
| CC1S | 1:0 | 通道1捕获/比较选择  00: CC1 通道配置成输出  01: CC1 通道配置成输入, TI1 作为 IC1 的输入  10: CC1 通道配置成输入, TI2 作为 IC1 的输入  11: CC1 通道配置成输入, TRC 作为 IC1 的输入  只有通过 TS 位(TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |

INPUT:

表 8-24 捕获/比较寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | IC2F | | | | IC2PSC | | CC2S | | IC1F | | | | IC1PSC | | CC1S | |

表 8-25 捕获/比较寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| IC2F | 15:12 | Input capture 2 filter  参考 IC1F 描述 |
| IC2PSC | 11:10 | Input capture 2 prescaler  参考 IC1PSC 描述 |
| CC2S | 9:8 | Capture/Compare 2 selection  00: CC2 通道配置成输出  01: CC2 通道配置成输入, TI2 作为 IC2 的输入  10: CC2 通道配置成输入, TI1 作为 IC2 的输入  11: CC2 通道配置成输入, TRC 作为 IC2 的输入. 只有通过 TS 位  (TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |
| IC1PSC | 3:2 | Input capture 1 prescaler  00: 没有预分频器，当检测到捕捉输入的边沿时就会进行捕捉  01: 每 2 次事件进行 1 次捕捉  10: 每 4 次事件进行 1 次捕捉  11: 每 8 次事件进行 1 次捕捉 |
| IC1F | 7:4 | Input capture 1 filter  0:没有过滤器， fSAMPLING = fDTS  1: fSAMPLING = fCK\_INT, N = 2  2: fSAMPLING = fCK\_INT, N = 4  3: fSAMPLING = fCK\_INT, N = 8  4: fSAMPLING = fDTS / 2, N = 6  5: fSAMPLING = fDTS / 2, N = 8  6: fSAMPLING = fDTS / 4, N = 6  7: fSAMPLING = fDTS / 4, N = 8  8: fSAMPLING = fDTS / 8, N = 6  9: fSAMPLING = fDTS / 8, N = 8  10: fSAMPLING = fDTS / 16, N = 5  11: fSAMPLING = fDTS / 16, N = 6  12: fSAMPLING = fDTS / 16, N = 8  13: fSAMPLING = fDTS / 32, N = 5  14: fSAMPLING = fDTS / 32, N = 6  15: fSAMPLING = fDTS / 32, N = 8  注意: 当 ICxF[3:0] = 1、 2 或 3 时，必须注意公式中 fDTS 用 CK\_INT 代替 |
| CC1S | 1:0 | Capture/Compare 1 selection  00: CC1 通道配置成输出  01: CC1 通道配置成输入, TI1 作为 IC1 的输入  10: CC1 通道配置成输入, TI2 作为 IC1 的输入  11: CC1 通道配置成输入, TRC 作为 IC1 的输入. 只有通过 TS 位  (TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |

### 捕获/比较模式寄存器2(ccmr2)

OUTPUT:

表 8-26 捕获/比较模式寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | OC4CE | OC4M | | | OC4PE | OC4FE | CC4S | | OC3CE | OC3M | | | OC3PE | OC3FE | CC3S | |

表 8-27 捕获/比较模式寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| OC4CE | 15 | Output Compare 4 clear enable  参考OC1CE |
| OC4M | 14:12 | Output Compare 4 mode  参考 OC1M |
| OC4PE | 11 | Output Compare 4 preload enable  参考OC1PE |
| OC4FE | 20 | Output Compare 4 fast enable  参考OC1FE |
| CC4S | 9:8 | Capture/Compare 4 selection  00: CC4 通道配置成输出  01: CC4 通道配置成输入, TI4 作为 IC4 的输入  10: CC4 通道配置成输入, TI3 作为 IC4 的输入  11: CC4 通道配置成输入, TRC 作为 IC4 的输入. 只有通过 TS 位  (TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |
| OC3CE | 7 | Output Compare 1 clear enable  参考 OC1CE |
| OC3M | 6:4 | Output Compare 3 mode  参考OC1M |
| OC3PE | 3 | Output Compare 3 preload enable  参考 OC1PE |
| OC3FE | 2 | Output Compare 3 fast enable  参考 OC1FE |
| CC3S | 1:0 | Capture/Compare 3 selection  00: CC3 通道配置成输出  01: CC3 通道配置成输入, TI3 作为 IC3 的输入  10: CC3 通道配置成输入, TI4 作为 IC3 的输入  11: CC3 通道配置成输入, TRC 作为 IC3 的输入. 只有通过 TS 位  (TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |

INPUT:

表 8-28 捕获/比较模式寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | IC4F | | | | IC4PSC | | CC4S | | IC3F | | | | IC3PSC | | CC3S | |

表 8-29 捕获/比较模式寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| IC4F | 15:12 | Input capture 4 filter  参考 IC1F 描述 |
| IC4PSC | 11:10 | Input capture 4 prescaler  参考 IC1PSC 描述 |
| CC4S | 9:8 | Capture/Compare 4 selection  00: CC4 通道配置成输出  01: CC4 通道配置成输入, TI4 作为 IC4 的输入  10: CC4 通道配置成输入, TI3 作为 IC4 的输入  11: CC4 通道配置成输入, TRC 作为 IC4 的输入. 只有通过 TS 位  (TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |
| IC3F | 7:4 | Input capture 3 prescaler  参考 IC1F |
| IC3PSC | 3:2 | Input capture 3 filter  参考 IC1PSC |
| CC3S | 1:0 | Capture/Compare 3 selection  00: CC3 通道配置成输出  01: CC3 通道配置成输入, TI3 作为 IC3 的输入  10: CC3 通道配置成输入, TI4 作为 IC3 的输入  11: CC3 通道配置成输入, TRC 作为 IC3 的输入. 只有通过 TS 位  (TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |

### 捕获/比较模式使能寄存器(ccer)

表 8-30 捕获/比较模式使能寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | CC4P | CC4E | CC3NP | CC3NE | CC3P | CC3E | CC2NP | CC2NE | CC2P | CC2E | CC1NP | CC1NE | CC1P | CC1E |

表 8-31 捕获/比较模式使能寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:14 | 保留 |
| CC4P | 13 | Capture/Compare 4 output polarity  参考CC1P |
| CC4E | 12 | Capture/Compare 4 output enable  参考CC1E |
| CC3NP | 11 | Capture/Compare 3 complementary output polarity  参考CC1NP |
| CC3NE | 10 | Capture/Compare 3 complementary output enable  参考CC1NE |
| CC3P | 9 | Capture/Compare 3 output polarity  参考CC1P |
| CC3E | 8 | Capture/Compare 3 output enable  参考CC1E |
| CC2NP | 7 | Capture/Compare 2 complementary output polarity  参考CC1NP |
| CC2NE | 6 | Capture/Compare 2 complementary output enable  参考CC1NE |
| CC2P | 5 | Capture/Compare 2 output polarity  参考CC1P |
| CC2E | 4 | Capture/Compare 2 output enable  参考CC1E |
| CC1NP | 3 | Capture/Compare 1 complementary output polarity  OUTPUT：  0：OC1为高时活跃  1：OC1为低时活跃  INPUT：  CC1NP/CC1P位选择TI1FP1和TI2FP1的有源极性  触发或捕获操作 |
| CC1NE | 2 | Capture/Compare 1 complementary output enable  OUTPUT:  0：禁用通道1输出功能  1：使能通道1输出功能  INPUT:  0：禁用通道1捕获功能  1：使能通道1捕获功能 |
| CC1P | 1 | Capture/Compare 1 output polarity  OUTPUT：  0：OC1为高时活跃  1：OC1为低时活跃  INPUT：  CC1NP/CC1P位选择TI1FP1和TI2FP1的有源极性  触发或捕获操作 |
| CC1E | 0 | Capture/Compare 1 output enable  OUTPUT:  0：禁用通道1输出功能  1：使能通道1输出功能  INPUT:  0：禁用通道1捕获功能  1：使能通道1捕获功能 |

### 计数寄存器(cnt)

表 8-32 计数寄存器

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

表 8-33 计数寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| CNT | 15:0 | 计数值 |

### 预分频寄存器(psc)

表 8-34 预分频寄存器

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

表 8-35 预分频寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| PSC | 15:0 | 预分频 |

### 重载寄存器(arr)

表 8-36 重载寄存器

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

表 8-37 重载寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| ARR | 15:0 | ARR是要在实际的自动重新加载寄存器中加载的值。  当自动重新加载值为空时，计数器被阻止。 |

### 重复计数寄存器(rcr)

表 8-38 重复计数寄存器

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

表 8-39 重复计数寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| RCR | 7:0 | 重复计数寄存器  这些位允许用户设置比较的更新速率寄存器（即从预加载到活动寄存器的定期传输）启用预加载寄存器，以及更新中断生成速率，如果这个中断是启用的。每次REP\_CNT相关下行计数器达到零，将生成一个更新事件并重新启动从REP值计数。因为REP\_CNT重新加载REP值重复更新事件U\_RC，任何写入TIMx\_RCR寄存器的操作都是直到下一次重复更新事件才被考虑到。  它表示在PWM模式下(REP+1)对应于：  -边缘对齐模式下的PWM周期数  -中心对齐模式下的半PWM周期数 |

### 捕获/比较寄存器1(ccr1)

表 8-40 捕获/比较寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | CCR1 | | | | | | | | | | | | | | | |

表 8-41 捕获/比较寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| CCR1 | 15:0 | Capture/Compare value1  如果将通道CCn配置为输出：  CCRn是要在实际捕获/比较n寄存器中加载的值（预加载值）。如果未在TIMx\_CCMRn寄存器(位OCnPE)。否则，预加载值复制到当发生更新事件时，主动捕获/比较n个寄存器。这个活动捕获/比较寄存器包含要进行比较的值计数器TIMx\_CNT和信号在OCn输出。  如果将通道CCn配置为输入：  CCRn是由最后一个输入捕获n个事件传输的计数器值(ICn) |

### 捕获/比较寄存器2(ccr2)

表 8-42 捕获/比较寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | CCR2 | | | | | | | | | | | | | | | |

表 8-43 捕获/比较寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| CCR3 | 15:0 | Capture/Compare value3 |

### 捕获/比较寄存器3(ccr3)

表 8-44 捕获/比较寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | CCR3 | | | | | | | | | | | | | | | |

表 8-45 捕获/比较寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| CCR3 | 15:0 | Capture/Compare value3 |

### 捕获/比较寄存器4(ccr4)

表 8-46 捕获/比较寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | CCR4 | | | | | | | | | | | | | | | |

表 8-47 捕获/比较寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| CCR4 | 15:0 | Capture/Compare value4 |

### DMA控制寄存器(dmaen)

表 8-48 DMA控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | | | | | | TDE | COMDE | CC4DE | CC3DE | CC2DE | CC1DE | UDE |

表 8-49 DMA控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:7 | 保留 |
| TDE | 6 | Trigger DMA request enable  0: Trigger DMA 请求禁止.  1: Trigger DMA 请求使能 |
| COMDE | 5 | COM DMA request enable  0: COM DMA 请求禁止.  1: COM DMA 请求使能 |
| CC4DE | 4 | Capture/Compare 4 DMA request enable  0: CC4 DMA 请求禁止.  1: CC4 DMA 请求使能 |
| CC3DE | 3 | Capture/Compare 3 DMA request enable  0: CC3 DMA 请求禁止.  1: CC3 DMA 请求使能 |
| CC2DE | 2 | Capture/Compare 2 DMA request enable  0: CC2 DMA 请求禁止.  1: CC2 DMA 请求使能 |
| CC1DE | 1 | Capture/Compare 1 DMA request enable  0: CC1 DMA 请求禁止.  1: CC1 DMA 请求使能 |
| UDE | 0 | Update DMA request enable  0: Update DMA 请求禁止.  1: Update DMA 请求使能 |

# 高级定时器(ADVANCED TIMER)

## 概述

高级控制定时器(ADVANCED TIMER，以下简称ATMR)由一可编程预除器驱动之16位自动重载计数器构成。

其亦可用于多种用途，包含测量输入信号脉冲长度(输入捕获)或产生输出波形(输出比较,PWM,互补PWM含dead-time插入)

可使用定时预除器与APB时钟控制预除器来调校脉冲长度与波形周期，由数微秒至数毫秒。

高级控制(ATMR),通用(CTMR)与基本(BTMR)定时器皆为完全独立不共享任何时钟源。

* 16位递增、递减、递增/递减自动重载计数器。
* 16位可编程预分频器，用于对计数器时钟频率进行分频（即运行时修改），分频系数介于1到65536之间。
* 多达4个独立通道，可用于：
* 输入捕获
* 输出比较
* PWM 生成（边沿和中心对齐模式）
* 单脉冲模式输出
* 带可编程死区的互补输出。
* 使用外部信号控制定时器且可实现多个定时器互连的同步电路。
* 重复计数器，用于仅在给定数目的计数器周期后更新定时器寄存器。
* 用于将定时器的输出信号置于复位状态或已知状态的断路输入。
* 发生如下事件时生成中断/DMA 请求：
* 更新：计数器上溢/下溢、计数器初始化（通过软件或内部/外部触发）
* 触发事件（计数器启动、停止、初始化或通过内部/外部触发计数）
* 输入捕获
* 输出比较
* 断路输入
* 支持定位用增量（正交）编码器和霍尔传感器电路。
* 外部时钟触发输入或cycle-by-cycle电流管理

## 结构框图



图 9‑1 ATMR结构框图

## 寄存器映射

ATMR基地址：0x4002\_2000，大小4KB。

表 9-1 寄存器映射(ADMR)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **寄存器** | **偏移量** | **读/写** | **描述** | **复位值** |
| CR1 | 0x000 | R/W | 控制寄存器1 | 0x0000 0000 |
| CR2 | 0x004 | R/W | 控制寄存器2 | 0x0000 0000 |
| SMCR | 0x008 | R/W | 从机模式控制寄存器 | 0x0000 0000 |
| IER | 0x00C | W | 中断启用寄存器 | 0x0000 0000 |
| IDR | 0x010 | W | 中断禁用寄存器 | 0x0000 0000 |
| IVS | 0x014 | R | 中断有效状态寄存器 | 0x0000 0000 |
| RIF | 0x018 | R | 中断原始中断标志 | 0x0000 0000 |
| IFM | 0x01C | R | 中断屏蔽中断标志 | 0x0000 0000 |
| ICR | 0x020 | W | 中断清除状态寄存器 | 0x0000 0000 |
| EGR | 0x024 | R/W | 事件生成寄存器 | 0x0000 0000 |
| CCMR1 | 0x028 | R/W | 捕获/比较模式寄存器1 | 0x0000 0000 |
| CCMR2 | 0x02C | R/W | 捕获/比较模式寄存器2 | 0x0000 0000 |
| CCER | 0x030 | R/W | 捕获/比较启用寄存器 | 0x0000 0000 |
| CNT | 0x034 | R/W | 定时计数 | 0x0000 0000 |
| PSC | 0x038 | R/W | 预定标器 | 0x0000 0000 |
| ARR | 0x03C | R/W | 自动重新加载寄存器 | 0x0000 0000 |
| RCR | 0x040 | R/W | 重复计数器寄存器 | 0x0000 0000 |
| CCR1 | 0x044 | R/W | 捕获/比较寄存器1 | 0x0000 0000 |
| CCR2 | 0x048 | R/W | 捕获/比较寄存器2 | 0x0000 0000 |
| CCR3 | 0x04C | R/W | 捕获/比较寄存器3 | 0x0000 0000 |
| CCR4 | 0x050 | R/W | 捕获/比较寄存器4 | 0x0000 0000 |
| BDTR | 0x054 | R/W | 中断和死时间寄存器 | 0x0000 0000 |
| DMAEN | 0x058 | R/W | DMA触发器事件启用 | 0x0000 0000 |

## 寄存器说明

### 控制寄存器1(cr1)

表 9-2 控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | CMPSEL | | | CMPSELP | CKD | | ARPE | CMS | | DIR | OPM | URS | UDIS | CEN |

表 9-3 控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:14 | 保留 |
| CMPSEL | 13:11 | Compare input selection  000: 输入比较禁止  001: 输入比较=输出比较 0  010: 输入比较=输出比较 1  011: 输入比较=输出比较 2  100: 输入比较=输出比较 3 |
| CMPSELP | 10 | Compare input selection polarity  0: 输入比较低有效  1: 输入比较高有效 |
| CKD | 9:8 | Clock division  此位域表示定时器时钟(CK\_INT)频率与死区时间发生器、 数字滤波器(ETR,TIx)使用的采样时钟(tDTS)之间的除法比  00: tDTS=tCK\_INT  01: tDTS=2\*tCK\_INT  10: tDTS=4\*tCK\_INT  11: 保留 |
| ARPE | 7 | Auto-reload preload enable  0: TIMx\_ARR 寄存器没有缓存  1: TIMx\_ARR 寄存器有缓存 |
| CMS | 6:5 | Center-aligned mode selection  00: Edge-aligned mode. 计数器根据方向位(DIR)向上或向下记数.  01: Center-aligned mode 1. 计数器向上和向下计数可选。输出通道(TIMx\_CCMRx 寄存器的 CCxS=00)比较中断标志位仅在计数器递减时设置.  10: Center-aligned mode 2. 计数器向上和向下计数可选。输出通道(TIMx\_CCMRx 寄存器的 CCxS=00)比较中断标志位仅在计数器递增时设置.  11: Center-aligned mode 3. 计数器向上和向下计数可选。输出通道(TIMx\_CCMRx 寄存器的 CCxS=00)比较中断标志位在计数器递增/递减时设置.  注:不允许从边缘对齐模式切换到中心对齐模式，只要计数器是启用的(CEN=1). |
| DIR | 4 | Direction  0:计数器作为递增计数器  1:计数器作为递减计数器  注:此位在计时器配置为中心对齐模式或编码器模式时作为只读寄存器 |
| OPM | 3 | One pulse mode  0: 计数器不会停止计数在产生更新事件时  1: 计数器停止计数在产生下一次更新事件时(清除CEN位) |
| URS | 2 | Update request source  此位由软件设置并清除，以选择 UEV 事件源.  0: 如果使能，以下任何事件都会生成更新中断或 DMA 请求.这些事件可以是:  – 计数器溢出/下溢  – 设置 UG 位  – 更新事件生成通过从模式控制缓冲寄存器加载其预加载值  1: 如果使能，只有计数器溢出/下溢时才会生成更新中断或 DMA 请求. |
| UDIS | 1 | Update disable  此位由软件设置并清除，以启用/禁用 UEV 事件生成  0: UEV 使能. 更新(UEV)事件是由以下事件之一生成:  – 计数器溢出/下溢  – 设置 UG 位  – 更新事件生成通过从模式控制缓冲寄存器加载其预加载值  1: UEV 禁止.更新事件没有生成，影子寄存器将保持它们的值(ARR、 PSC、CCRx)。 然而计数器和预分频器将被重新初始化如果软件对 UG 位进行设置或者接收到硬件复位 |
| CEN | 0 | Counter enable  0:计数器禁止  1:计数器使能  注:外接时钟、门控模式和编码器模式只能在软件设置 CEN 位之后进行工作。而触发模式下硬件可以自动设置 CEN 位 |

### 控制寄存器2(cr2)

表 9-4 控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | OIS4 | OIS3N | OIS3 | OIS2N | OIS2 | OIS1N | OIS1 | TI1S | MMS | | | CCDS | CCUS | -- | CCPC |

表 9-5 控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:15 | 保留 |
| OIS4 | 14 | Output Idle state 4 (OC4 output)  0: 当 MOE=0 时， OC1=0(在死区时间之后 OC4N 完成)  1: 当 MOE=0 时， OC1=1(在死区时间之后 OC4N 完成)  注意:只要 Lock 级别设定为 1、 2 或 3，这个 Bit 位就不能被修改(在TIMx\_BDTR 寄存器中的 LOCK 位). |
| OIS3N | 13 | Output Idle state 3 (OC3N output)  0: 当 MOE=0 时， 在死区时间之后 OC3N=0  1: 当 MOE=0 时， 在死区时间之后 OC3N=1  注意:只要 Lock 级别设定为 1、 2 或 3，这个 Bit 位就不能被修改(在TIMx\_BDTR 寄存器中的 LOCK 位). |
| OIS3 | 12 | Output Idle state 3 (OC3 output)  0: 当 MOE=0 时， OC1=0(在死区时间之后 OC3N 完成)  1: 当 MOE=0 时， OC1=1(在死区时间之后 OC3N 完成)  注意:只要 Lock 级别设定为 1、 2 或 3，这个 Bit 位就不能被修改(在TIMx\_BDTR 寄存器中的 LOCK 位). |
| OIS2N | 11 | Output Idle state 2 (OC2N output)  0: 当 MOE=0 时， 在死区时间之后 OC2N=0  1: 当 MOE=0 时， 在死区时间之后 OC2N=1  注意:只要 Lock 级别设定为 1、 2 或 3，这个 Bit 位就不能被修改(在TIMx\_BDTR 寄存器中的 LOCK 位). |
| OIS2 | 10 | Output Idle state 2 (OC2 output)  0: 当 MOE=0 时， OC1=0(在死区时间之后 OC2N 完成)  1: 当 MOE=0 时， OC1=1(在死区时间之后 OC2N 完成)  注意:只要 Lock 级别设定为 1、 2 或 3，这个 Bit 位就不能被修改(在TIMx\_BDTR 寄存器中的 LOCK 位). |
| OIS1N | 9 | Output Idle state 1 (OC1N output)  0: 当 MOE=0 时， 在死区时间之后 OC1N=0  1: 当 MOE=0 时， 在死区时间之后 OC1N=1  注意:只要 Lock 级别设定为 1、 2 或 3，这个 Bit 位就不能被修改(在TIMx\_BDTR 寄存器中的 LOCK 位). |
| OIS1 | 8 | Output Idle state 1 (OC1 output)  0: 当 MOE=0 时， OC1=0(在死区时间之后 OC1N 完成)  1: 当 MOE=0 时， OC1=1(在死区时间之后 OC1N 完成)  注意:只要 Lock 级别设定为 1、 2 或 3，这个 Bit 位就不能被修改(在TIMx\_BDTR 寄存器中的 LOCK 位). |
| TI1S | 7 | TI1 selection  0: TIMx\_CH1 作为 TI1 输入源  1: TIMx\_CH1, CH2和CH3的异或逻辑输出作为 TI1 输入源 |
| MMS | 6:4 | Master mode selection  这些 Bit 位可以对主模式下发送给从计时器进行同步(TRGO)的信息进行选择。组合如下:  000: Reset -TIMx\_EGR 寄存器的 UG 信号作为 trigger 输出(TRGO)。 如果此复位是由 trigger 输入(在复位模式下配置从模式控制器)生成的， 那么TRGO 上的信号与实际复位相比会有延迟.  001: Enable -计数器使能信号 CNT\_EN 作为 trigger 输出(TRGO)。 这对于同时启动多个计时器或控制启用从计时器窗口是很有用的。计数器使能信号是在门控模式下由 CEN 控制位与 trigger 输入进行逻辑或生成的。当计数器使能信号由 trigger 输入控制时， TRGO 上有一个延迟，除非选择了主/从模式(参见 TIMx\_SMCR 寄存器中的 MSM 位描述).  010: Update –事件更新作为 trigger 输出（TRGO） 例如，主定时器可以用作从定时器的预分频器.  011: Compare Pulse – 一旦捕获或比较匹配产生， 在设置 CC1IF 标志位为1 的同时（即使它已经是高电平） trigger 输出将发送一个正脉冲（TRGO） .  100: Compare - OC1REF 信号作为 trigger 输出(TRGO)  101: Compare - OC2REF 信号作为 trigger 输出 (TRGO)  110: Compare - OC3REF 信号作为 trigger 输出(TRGO)  111: Compare - OC4REF 信号作为 trigger 输出 (TRGO) |
| CCDS | 3 | Capture/compare DMA selection  0: CCx DMA 请求发送当 CCx 事件发生  1: CCx DMA 请求发送当有 Update 事件发生 |
| CCUS | 2 | Capture/compare control update selection  0: 当捕获/比较控制位预加载时(CCPC=1)， 它们仅通过设置 COMG Bit 位进行更新  1: 当捕获/比较控制位预加载(CCPC=1)时， 它们通过设置 COMG Bit 位或遇到 TRGI 上升沿进行更新. |
| Reserved | 1 | 保留 |
| CCPC | 0 | Capture/compare preloaded control  0: CCxE, CCxNE 和 OCxM bit 位没有预加载  1: CCxE, CCxNE 和 OCxM Bit 位是预加载的，在写入之后，只有在发生通信事件(COM)时才会更新(根据 CCUS Bit 位的不同，设置 COMG Bit 位或者检测 TRGI 上升边).  注意:这个 Bit 只作用于具有互补输出的通道 |

### 从机模式控制寄存器(smcr)

表 9-6 从机模式控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | ETP | ETE | ETPS | | ETF | | | | MSM | TS | | | -- | SMS | | |

表 9-7 从机模式控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| ETP | 15 | External trigger polarity  该位用于对ETR的Trigger 操作极性进行选择  0: ETR 正向，高电平或者上升沿有效.  1: ETR 反向，低电平或者下降沿有效. |
| ETE | 14 | External clock enable  此位用于使能外部时钟模式 2.  0: 外部时钟模式 2 禁止  1: 外部时钟模式 2 使能. |
| ETPS | 13:12 | External trigger prescaler  外部触发信号ETRP的频率最多必须为TIMxCLK的1/4频繁性可以使预调节器能够降低ETRP的频率  00: Prescaler OFF  01: ETRP frequency divided by 2  10: ETRP frequency divided by 4  11: ETRP frequency divided by 8 |
| ETF | 11:8 | External trigger filter  然后，这个位场定义了用于采样ETRP信号的频率  应用于ETRP的数字滤波器的长度。该数字滤波器由一个事件计数器组成，其中有N个连续的事件需要验证输出上的转换：  0000: No filter, sampling is done at fDTS  0001: fSAMPLING = fCK\_INT, N = 2  0010: fSAMPLING = fCK\_INT, N = 4  0011: fSAMPLING = fCK\_INT, N = 8  0100: fSAMPLING = fDTS / 2, N = 6  0101: fSAMPLING = fDTS / 2, N = 8  0110: fSAMPLING = fDTS / 4, N = 6  0111: fSAMPLING = fDTS / 4, N = 8  1000: fSAMPLING = fDTS / 8, N = 6  1001: fSAMPLING = fDTS / 8, N = 8  1010: fSAMPLING = fDTS / 16, N = 5  1011: fSAMPLING = fDTS / 16, N = 6  1100: fSAMPLING = fDTS / 16, N = 8  1101: fSAMPLING = fDTS / 32, N = 5  1110: fSAMPLING = fDTS / 32, N = 6  1111: fSAMPLING = fDTS / 32, N = 8 |
| MSM | 7 | Master/slave mode |
| TS | 6:4 | Trigger selection  此位字段选择要用于同步计数器的触发器输入  000: Internal Trigger 0 (ITR0)  001: Internal Trigger 1 (ITR1)  010: Internal Trigger 2 (ITR2)  011: Internal Trigger 3 (ITR3)  100: TI1 Edge Detector (TI1F\_ED)  101: Filtered Timer Input 1 (TI1FP1)  110: Filtered Timer Input 2 (TI2FP2)  111: External Trigger input (ETRF)  注意：这些位只有在不使用时才可以更改  SMS=000)，以避免在过渡时的错误边缘检测。 |
| Reserved | 3 | 保留 |
| SMS | 2:0 | Slave mode selection  当外部信号被选择时，触发信号的主动边缘  (TRGI)被链接到在外部输入上选择的极性  000：从属模式被禁用-如果CEN=1，那么预调用器是时钟的  直接通过内部的时钟。  001：编码器模式1-TI2FP1边缘上的计数器数量增加/下降取决于TI1FP2水平。  010：编码器模式2-TI1FP2边缘上的计数器数量增加/下降  取决于TI2FP1水平。  011：编码器模式3-计数器计数数在TI1FP1和TI2FP2的边缘取决于其他输入的级别。  100：重置模式-所选触发器输入的上升边缘(TRGI)  重新初始化计数器并生成寄存器的更新。  101：门控模式-当触发器输入时，计数器时钟已启用  (TRGI)高。计数器一旦触发器就停止（但不会重置）  变得低。计数器的启动和停止都被控制了。  110：触发模式-计数器从触发器TRGI的上升边缘开始（但它不会被重置）。只控制计数器的启动位置。  111：外部时钟模式1-所选触发器的上升边缘(TRGI)  钟计数器。 |

### 中断使能寄存器(ier)

表 9-8 中断使能寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | CC4OI | CC3OI | CC2OI | CC1OI | -- | BI | TI | COMI | CC4I | CC3I | CC2I | CC1I | UI |

表 9-9 中断使能寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:13 | 保留 |
| CC4OI | 12 | 通道4捕获/比较过捕(overcapture)中断使能  0：无效， 1：CC4 overcapture中断使能 |
| CC3OI | 11 | 通道3捕获/比较过捕(overcapture)中断使能  0：无效， 1：CC3 overcapture中断使能 |
| CC2OI | 10 | 通道2捕获/比较过捕(overcapture)中断使能  0：无效， 1：CC2 overcapture中断使能 |
| CC1OI | 9 | 通道1捕获/比较过捕(overcapture)中断使能  0：无效， 1：CC1 overcapture中断使能 |
| Reserved | 8 | 保留 |
| BI | 7 | BREAK中断使能  0：无效， 1：BREAK中断使能 |
| TI | 6 | 触发中断使能  0：无效， 1：触发中断使能 |
| COMI | 5 | COM中断使能  0：无效， 1：COM中断使能 |
| CC4I | 4 | 通道4捕获/比较中断使能  0：无效， 1：CC4中断使能 |
| CC3I | 3 | 通道3捕获/比较中断使能  0：无效， 1：CC3中断使能 |
| CC2I | 2 | 通道2捕获/比较中断使能  0：无效， 1：CC2中断使能 |
| CC1I | 1 | 通道1捕获/比较中断使能  0：无效， 1：CC1中断使能 |
| UI | 0 | 更新中断使能  0：无效， 1：更新中断使能 |

### 中断关闭寄存器(idr)

表 9-10 中断关闭寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | CC4OI | CC3OI | CC2OI | CC1OI | -- | BI | TI | COMI | CC4I | CC3I | CC2I | CC1I | UI | -- |

表 9-11 中断关闭寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:13 | 保留 |
| CC4OI | 12 | 通道4捕获/比较过捕(overcapture)中断禁止  0：无效， 1：CC4 overcapture中断禁止 |
| CC3OI | 11 | 通道3捕获/比较过捕(overcapture)中断禁止  0：无效， 1：CC3 overcapture中断禁止 |
| CC2OI | 10 | 通道2捕获/比较过捕(overcapture)中断禁止  0：无效， 1：CC2 overcapture中断禁止 |
| CC1OI | 9 | 通道1捕获/比较过捕(overcapture)中断禁止  0：无效， 1：CC1 overcapture中断禁止 |
| Reserved | 8 | 保留 |
| BI | 7 | BREAK中断禁止  0：无效， 1：BREAK中断禁止 |
| TI | 6 | 触发中断禁止  0：无效， 1：触发中断禁止 |
| COMI | 5 | COM中断禁止  0：无效， 1：COM中断禁止 |
| CC4I | 4 | 通道4捕获/比较中断禁止  0：无效， 1：CC4中断禁止 |
| CC3I | 3 | 通道3捕获/比较中断禁止  0：无效， 1：CC3中断禁止 |
| CC2I | 2 | 通道2捕获/比较中断禁止  0：无效， 1：CC2中断禁止 |
| CC1I | 1 | 通道1捕获/比较中断禁止  0：无效， 1：CC1中断禁止 |
| UI | 0 | 更新中断禁止  0：无效， 1：更新中断禁止 |

### 中断有效状态寄存器(ivs)

表 9-12 中断有效状态寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | CC4OI | CC3OI | CC2OI | CC1OI | -- | BI | TI | COMI | CC4I | CC3I | CC2I | CC1I | UI |

表 9-13 中断有效状态寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:13 | 保留 |
| CC4OI | 12 | 通道4捕获/比较过捕(overcapture)中断有效状态  0：CC4 overcapture中断禁止状态，  1：CC4 overcapture中断使能状态 |
| CC3OI | 11 | 通道3捕获/比较过捕(overcapture)中断有效状态  0：CC3 overcapture中断禁止状态，  1：CC3 overcapture中断使能状态 |
| CC2OI | 10 | 通道2捕获/比较过捕(overcapture)中断有效状态  0：CC2 overcapture中断禁止状态，  1：CC2 overcapture中断使能状态 |
| CC1OI | 9 | 通道1捕获/比较过捕(overcapture)中断有效状态  0：CC1 overcapture中断禁止状态，  1：CC1 overcapture中断使能状态 |
| Reserved | 8 | 保留 |
| BI | 7 | BREAK中断有效状态  0：BREAK中断禁止状态， 1：BREAK中断使能状态 |
| TI | 6 | 触发中断有效状态  0：触发中断禁止状态， 1：触发中断使能状态 |
| COMI | 5 | COM中断有效状态  0：COM中断禁止状态， 1：COM中断使能状态 |
| CC4I | 4 | 通道4捕获/比较中断有效状态  0：CC4中断禁止状态， 1：CC4中断使能状态 |
| CC3I | 3 | 通道3捕获/比较中断有效状态  0：CC3中断禁止状态， 1：CC3中断使能状态 |
| CC2I | 2 | 通道2捕获/比较中断有效状态  0：CC2中断禁止状态， 1：CC2中断使能状态 |
| CC1I | 1 | 通道1捕获/比较中断有效状态  0：CC1中断禁止状态， 1：CC1中断使能状态 |
| UI | 0 | 更新中断有效状态  0：更新中断禁止状态， 1：更新中断使能状态 |

### 中断有效标志寄存器(rif)

表 9-14 中断有效标志寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | CC4OI | CC3OI | CC2OI | CC1OI | -- | BI | TI | COMI | CC4I | CC3I | CC2I | CC1I | UI |

表 9-15 中断有效标志寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:13 | 保留 |
| CC4OI | 12 | 通道4捕获/比较过捕(overcapture)中断标志  参考CC1OI |
| CC3OI | 11 | 通道3捕获/比较过捕(overcapture)中断标志  参考CC1OI |
| CC2OI | 10 | 通道2捕获/比较过捕(overcapture)中断标志  参考CC1OI |
| CC1OI | 9 | 通道1捕获/比较过捕(overcapture)中断标志  只有在输入捕获模式下配置相应的通道时，硬件才会设置此标志  0：没有检测到overcapture中断  1：当CC1I标志置1时且计数值已经被CCR1寄存器捕获 |
| Reserved | 8 | 保留 |
| BI | 7 | BREAK中断标志  如果启用了中断，则一旦break输入产生，硬件就会设置此标志。  0：没有检测到break事件  1：检测到有活跃的break输入 |
| TI | 6 | 触发中断标志  这个标志是由硬件对触发事件设置的(当从模式控制器在所有模式下启用门控模式时，检测到trgi输入，但如果启用了中断模式，则检测到活动边缘  0：没有发出事件发生  1：触发中断挂起 |
| COMI | 5 | COM中断标志  这个标志是由COM事件上的硬件设置的控制位-CcxE，CCxNE，OCxM-已经更新)  0：没有检测到COM事件  1：COM中断挂起 |
| CC4I | 4 | 通道4捕获/比较中断标志  参考CC1I |
| CC3I | 3 | 通道3捕获/比较中断标志  参考CC1I |
| CC2I | 2 | 通道2捕获/比较中断标志  参考CC1I |
| CC1I | 1 | 通道1捕获/比较中断标志  通道1被配置为输出模式：  当计数器与比较值匹配时，此标志由硬件设置，如果启用了中断，则在中心对齐模式下有些例外(请参阅timx\_cr1寄存器描述中的cms位)。在timx\_icr中写入1来清除原始中断  0：没有匹配  1：计数器timx\_cnt的内容与timx\_ccr1寄存器的内容相匹配。在计数器溢出（向上计数和上/下计数模式）或欠流（向下计数模式）模式时，如果CCR1大于ARR时，此位置1  通道1被配置为输入模式：  这个位是由捕获上的硬件设置的。通过软件或读取timx\_ccr1寄存器进行清除  0：没有输入捕获发生  1：计数器值已在timx\_ccr1寄存器中捕获(在ic1上检测到一条与所选极性匹配的边缘) |
| UI | 0 | 更新中断标志  0：没有更新发生，  1：更新中断挂起。当寄存器有更新发生时，此位会被硬件置1  触发条件：  当CR1中UDIS=0，重复计数器值的溢出或下流（如果重复计数器=0则更新）  当CR1中的URS=0，UDIS=0时，CNT被软件使用timx\_egr寄存器中的ug位重新初始化  当CR1中的URS=0，UDIS=0时，CNT被一个触发器事件重新初始化 |

### 中断屏蔽寄存器(ifm)

表 9-16 中断屏蔽寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | - | | | | | | | | | | | | | | | | | | | CC4OI | CC3OI | CC2OI | CC1OI | - | BI | TI | COMI | CC4I | CC3I | CC2I | CC1I | UI |

表 9-17 中断屏蔽寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:13 | 保留 |
| CC4OI | 12 | 通道4捕获/比较过捕(overcapture)中断标志屏蔽  只有在输入捕获模式下配置相应的通道时，硬件才会设置此标志  0：没有检测到overcapture中断  1：当CC4I标志置1时且计数值已经被CCR4寄存器捕获 |
| CC3OI | 11 | 通道3捕获/比较过捕(overcapture)中断标志屏蔽  只有在输入捕获模式下配置相应的通道时，硬件才会设置此标志  0：没有检测到overcapture中断  1：当CC3I标志置1时且计数值已经被CCR3寄存器捕获 |
| CC2OI | 10 | 通道2捕获/比较过捕(overcapture)中断标志屏蔽  只有在输入捕获模式下配置相应的通道时，硬件才会设置此标志  0：没有检测到overcapture中断  1：当CC2I标志置1时且计数值已经被CCR2寄存器捕获 |
| CC1OI | 9 | 通道1捕获/比较过捕(overcapture)中断标志屏蔽  只有在输入捕获模式下配置相应的通道时，硬件才会设置此标志  0：没有检测到overcapture中断  1：当CC1I标志置1时且计数值已经被CCR1寄存器捕获 |
| Reserved | 8 | 保留 |
| BI | 7 | BREAK中断标志屏蔽  如果启用了中断，则一旦break输入产生，硬件就会设置此标志。  0：没有检测到break事件  1：检测到有活跃的break输入 |
| TI | 6 | 触发中断标志屏蔽  这个标志是由硬件对触发事件设置的(当从模式控制器在所有模式下启用门控模式时，检测到trgi输入，但如果启用了中断模式，则检测到活动边缘  0：没有发出事件发生  1：触发中断挂起 |
| COMI | 5 | COM中断标志屏蔽  这个标志是由COM事件上的硬件设置的控制位-CcxE，CCxNE，OCxM-已经更新)  0：没有检测到COM事件  1：COM中断挂起 |
| CC4I | 4 | 通道4捕获/比较中断标志屏蔽  通道4被配置为输出模式：  当计数器与比较值匹配时，此标志由硬件设置，如果启用了中断，则在中心对齐模式下有些例外(请参阅timx\_cr4寄存器描述中的cms位)。在timx\_icr中写入1来清除原始中断  0：没有匹配  1：计数器timx\_cnt的内容与timx\_ccr4寄存器的内容相匹配。在计数器溢出（向上计数和上/下计数模式）或欠流（向下计数模式）模式时，如果CCR1大于ARR时，此位置1  通道1被配置为输入模式：  这个位是由捕获上的硬件设置的。通过软件或读取timx\_ccr4寄存器进行清除  0：没有输入捕获发生  1：计数器值已在timx\_ccr4寄存器中捕获(在ic4上检测到一条与所选极性匹配的边缘) |
| CC3I | 3 | 通道3捕获/比较中断标志屏蔽  通道3被配置为输出模式：  当计数器与比较值匹配时，此标志由硬件设置，如果启用了中断，则在中心对齐模式下有些例外(请参阅timx\_cr3寄存器描述中的cms位)。在timx\_icr中写入1来清除原始中断  0：没有匹配  1：计数器timx\_cnt的内容与timx\_ccr3寄存器的内容相匹配。在计数器溢出（向上计数和上/下计数模式）或欠流（向下计数模式）模式时，如果CCR1大于ARR时，此位置1  通道1被配置为输入模式：  这个位是由捕获上的硬件设置的。通过软件或读取timx\_ccr3寄存器进行清除  0：没有输入捕获发生  1：计数器值已在timx\_ccr3寄存器中捕获(在ic3上检测到一条与所选极性匹配的边缘) |
| CC2I | 2 | 通道2捕获/比较中断标志屏蔽  通道1被配置为输出模式：  当计数器与比较值匹配时，此标志由硬件设置，如果启用了中断，则在中心对齐模式下有些例外(请参阅timx\_cr1寄存器描述中的cms位)。在timx\_icr中写入1来清除原始中断  0：没有匹配  1：计数器timx\_cnt的内容与timx\_ccr1寄存器的内容相匹配。在计数器溢出（向上计数和上/下计数模式）或欠流（向下计数模式）模式时，如果CCR1大于ARR时，此位置1  通道1被配置为输入模式：  这个位是由捕获上的硬件设置的。通过软件或读取timx\_ccr1寄存器进行清除  0：没有输入捕获发生  1：计数器值已在timx\_ccr1寄存器中捕获(在ic1上检测到一条与所选极性匹配的边缘) |
| CC1I | 1 | 通道1捕获/比较中断标志屏蔽  通道1被配置为输出模式：  当计数器与比较值匹配时，此标志由硬件设置，如果启用了中断，则在中心对齐模式下有些例外(请参阅timx\_cr1寄存器描述中的cms位)。在timx\_icr中写入1来清除原始中断  0：没有匹配  1：计数器timx\_cnt的内容与timx\_ccr1寄存器的内容相匹配。在计数器溢出（向上计数和上/下计数模式）或欠流（向下计数模式）模式时，如果CCR1大于ARR时，此位置1  通道1被配置为输入模式：  这个位是由捕获上的硬件设置的。通过软件或读取timx\_ccr1寄存器进行清除  0：没有输入捕获发生  1：计数器值已在timx\_ccr1寄存器中捕获(在ic1上检测到一条与所选极性匹配的边缘) |
| UI | 0 | 更新中断标志屏蔽  当更新事件中断UI被使能时，此位是由硬件置1  0：没有更新发生，  1：更新中断挂起。当寄存器有更新发生时，此位会被硬件置1  触发条件：  1. 当CR1中UDIS=0，重复计数器值的溢出或下流（如果重复计数器=0则更新）  2. 当CR1中的URS=0，UDIS=0时，CNT被软件使用timx\_egr寄存器中的ug位重新初始化  3. 当CR1中的URS=0，UDIS=0时，CNT被一个触发器事件重新初始化 |

### 中断状态清除寄存器(icr)

表 9-18 中断状态清除寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | CC4OI | CC3OI | CC2OI | CC1OI | - | BI | TI | COMI | CC4I | CC3I | CC2I | CC1I | UI |

表 9-19 中断状态清除寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:13 | 保留 |
| CC4OI | 12 | 通道4捕获/比较过捕(overcapture)中断清除  0：无效， 1：CC4 overcapture中断清除 |
| CC3OI | 11 | 通道3捕获/比较过捕(overcapture)中断清除  0：无效， 1：CC3 overcapture中断清除 |
| CC2OI | 10 | 通道2捕获/比较过捕(overcapture)中断清除  0：无效， 1：CC2 overcapture中断清除 |
| CC1OI | 9 | 通道1捕获/比较过捕(overcapture)中断清除  0：无效， 1：CC1 overcapture中断清除 |
| Reserved | 8 | 保留 |
| BI | 7 | BREAK中断清除  0：无效， 1：BREAK中断清除 |
| TI | 6 | 触发中断清除  0：无效， 1：触发中断清除 |
| COMI | 5 | COM中断清除  0：无效， 1：COM中断清除 |
| CC4I | 4 | 通道4捕获/比较中断清除  0：无效， 1：CC4中断清除 |
| CC3I | 3 | 通道3捕获/比较中断清除  0：无效， 1：CC3中断清除 |
| CC2I | 2 | 通道2捕获/比较中断清除  0：无效， 1：CC2中断清除 |
| CC1I | 1 | 通道1捕获/比较中断清除  0：无效， 1：CC1中断清除 |
| UI | 0 | 更新中断清除  0：无效， 1：更新中断清除 |

### 事件产生寄存器(egr)

表 9-20 事件产生寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | | | | | BG | TG | COMG | CC4G | CC3G | CC2G | CC1G | UG |

表 9-21 事件产生寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31：8 | 保留 |
| BG | 7 | BREAK事件产生  0：无效  1：将生成一个中断事件。已清除了MOE位，并设置了BIF标志。如果启用，可能会发生相关的中断或DMA传输。 |
| TG | 6 | 触发事件产生  0：无效  1：TIF标志已在TIMx\_RIF寄存器中设置。相关中断或DMA如果启用了传输，就可以发生传输 |
| COMG | 5 | 捕获/比较控制更新生成  0：无效  1：当设置了CCPC位时，它允许更新CCxE、CCxNE和OCxM位。注意：这个位只作用于具有互补输出的通道。 |
| CC4G | 4 | 通道4捕获/比较事件产生  0：无效  1：在通道1上生成了一个捕获/比较事件  通道1被配置为输出模式：  有中断或者DMA请求时，此位置1  通道1被配置为输入模式：  有输入捕获事件发生时，此位置1；如果此时CC4I已经为1，将CC4OI置1 |
| CC3G | 3 | 通道3捕获/比较事件产生  0：无效  1：在通道1上生成了一个捕获/比较事件  通道1被配置为输出模式：  有中断或者DMA请求时，此位置1  通道1被配置为输入模式：  有输入捕获事件发生时，此位置1；如果此时CC3I已经为1，将CC3OI置1 |
| CC2G | 2 | 通道2捕获/比较事件产生  0：无效  1：在通道1上生成了一个捕获/比较事件  通道1被配置为输出模式：  有中断或者DMA请求时，此位置1  通道1被配置为输入模式：  有输入捕获事件发生时，此位置1；如果此时CC2I已经为1，将CC2OI置1 |
| CC1G | 1 | 通道1捕获/比较事件产生  0：无效  1：在通道1上生成了一个捕获/比较事件  通道1被配置为输出模式：  有中断或者DMA请求时，此位置1  通道1被配置为输入模式：  有输入捕获事件发生时，此位置1；如果此时CC1I已经为1，将CC1OI置1 |
| UG | 0 | 更新事件产生  这个位可以由软件设置，它是由硬件自动清除的  0：无效  1：重新初始化计数器并生成寄存器的更新。请注意，prescaler计数器也会被清除（无论如何，prescaler比例不受影响）。如果模式为中心对齐模式或dir=0（向上计数）时，CNT被清零，如果dir=1（向下计数）时，则取自动重加载值(timx\_arr)。 |

### 捕获/比较模式寄存器1(ccmr1)

OUTPUT:

表 9-22 捕获/比较模式寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | OC2CE | OC2M | | | OC2PE | OC2FE | CC2S | | OC1CE | OC1M | | | OC1PE | OC1FE | CC1S | |

表 9-23 捕获/比较模式寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| OC2CE | 15 | 通道2输出比较清除使能  参考OC1CE |
| OC2M | 14:12 | 通道2输出模式  参考OC1M |
| OC2PE | 11 | 通道2输出比较重载使能  参考OC1PE |
| OC2FE | 20 | 通道2输出比较快速使能  参考OC1FE |
| CC2S | 9:8 | 通道2捕获/比较选择  00: CC2 通道配置成输出  01: CC2 通道配置成输入, TI2 作为 IC2 的输入  10: CC2 通道配置成输入, TI1 作为 IC2 的输入  11: CC2 通道配置成输入, TRC 作为 IC2 的输入  只有通过 TS 位(TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |
| OC1CE | 7 | 通道1输出比较清除使能  0：OC1Ref不受ETRF输入的影响  1：一旦在ETRF输入上检测到一个高水平，OC1Ref就会被清除。 |
| OC1M | 6:4 | 通道1输出模式  0：输出比较寄存器timx\_ccr1和计数器timx\_cnt之间的比较对输出没有影响（这种模式用于生成定时基）  1：在匹配时将通道1设置为活动级别。当计数器timx\_cnt与捕获/比较寄存器1(timx\_ccr1)相匹配时，Oc1ref信号被强制变高。  2：在匹配时将通道1设置为非活动级别。当计数器timx\_cnt与捕获/比较寄存器1(timx\_ccr1)相匹配时，oc1ref信号被迫变低  3：oc1ref在timx\_cnt=timx\_ccr1翻转  4：oc1ref被强制低  5：oc1ref被强制高  6：在向上计数中，只要其他timx\_cnttimx\_ccr1是活跃的(oc1ref=1)。  7：在向上计数中，只要timx\_cnttimx\_ccr1是不活动的，通道1就是活动的。 |
| OC1PE | 3 | 通道1输出比较重载使能  0：禁用CCR1的preload。任何时候更新CCR1值都会被马上重载。  1：使能CCR1的preload。CCR1的重载值只有在发生更新事件后才会被重载。 |
| OC1FE | 2 | 通道1输出比较快速使能  0：CC1的行为通常取决于计数器和CCR1的值。当触发器打开时。使能CC1输出的最小延迟时间为一个边沿触发输入是5个时钟周期  1：触发器输入上的活跃边沿类似于CC1上的比较匹配产量然后，将OC设置为独立于结果的比较水平  的比较。延迟采样触发器输入并使能CC1输出被减少到3个时钟周期。OCFE仅在通道配置为PWM1或PWM是才活跃 |
| CC1S | 1:0 | 通道1捕获/比较选择  00: CC1 通道配置成输出  01: CC1 通道配置成输入, TI1 作为 IC1 的输入  10: CC1 通道配置成输入, TI2 作为 IC1 的输入  11: CC1 通道配置成输入, TRC 作为 IC1 的输入  只有通过 TS 位(TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |

INPUT:

表 9-24 捕获/比较模式寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | IC2F | | | | IC2PSC | | CC2S | | IC1F | | | | IC1PSC | | CC1S | |

表 9-25 捕获/比较模式寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| IC2F | 15:12 | Input capture 2 filter  参考 IC1F 描述 |
| IC2PSC | 11:10 | Input capture 2 prescaler  参考 IC1PSC 描述 |
| CC2S | 9:8 | Capture/Compare 2 selection  00: CC2 通道配置成输出  01: CC2 通道配置成输入, TI2 作为 IC2 的输入  10: CC2 通道配置成输入, TI1 作为 IC2 的输入  11: CC2 通道配置成输入, TRC 作为 IC2 的输入. 只有通过 TS 位  (TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |
| IC1PSC | 3:2 | Input capture 1 prescaler  00: 没有预分频器，当检测到捕捉输入的边沿时就会进行捕捉  01: 每 2 次事件进行 1 次捕捉  10: 每 4 次事件进行 1 次捕捉  11: 每 8 次事件进行 1 次捕捉 |
| IC1F | 7:4 | Input capture 1 filter  0:没有过滤器， fSAMPLING = fDTS  1: fSAMPLING = fCK\_INT, N = 2  2: fSAMPLING = fCK\_INT, N = 4  3: fSAMPLING = fCK\_INT, N = 8  4: fSAMPLING = fDTS / 2, N = 6  5: fSAMPLING = fDTS / 2, N = 8  6: fSAMPLING = fDTS / 4, N = 6  7: fSAMPLING = fDTS / 4, N = 8  8: fSAMPLING = fDTS / 8, N = 6  9: fSAMPLING = fDTS / 8, N = 8  10: fSAMPLING = fDTS / 16, N = 5  11: fSAMPLING = fDTS / 16, N = 6  12: fSAMPLING = fDTS / 16, N = 8  13: fSAMPLING = fDTS / 32, N = 5  14: fSAMPLING = fDTS / 32, N = 6  15: fSAMPLING = fDTS / 32, N = 8  注意: 当 ICxF[3:0] = 1、 2 或 3 时，必须注意公式中 fDTS 用 CK\_INT 代替 |
| CC1S | 1:0 | Capture/Compare 1 selection  00: CC1 通道配置成输出  01: CC1 通道配置成输入, TI1 作为 IC1 的输入  10: CC1 通道配置成输入, TI2 作为 IC1 的输入  11: CC1 通道配置成输入, TRC 作为 IC1 的输入. 只有通过 TS 位  (TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |

### 捕获/比较模式寄存器2(ccmr2)

OUTPUT:

表 9-26 捕获/比较模式寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | OC4CE | OC4M | | | OC4PE | OC4FE | CC4S | | OC3CE | OC3M | | | OC3PE | OC3FE | CC3S | |

表 9-27 捕获/比较模式寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| OC4CE | 15 | Output Compare 4 clear enable  参考OC1CE |
| OC4M | 14:12 | Output Compare 4 mode  参考 OC1M |
| OC4PE | 11 | Output Compare 4 preload enable  参考OC1PE |
| OC4FE | 20 | Output Compare 4 fast enable  参考OC1FE |
| CC4S | 9:8 | Capture/Compare 4 selection  00: CC4 通道配置成输出  01: CC4 通道配置成输入, TI4 作为 IC4 的输入  10: CC4 通道配置成输入, TI3 作为 IC4 的输入  11: CC4 通道配置成输入, TRC 作为 IC4 的输入. 只有通过 TS 位  (TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |
| OC3CE | 7 | Output Compare 1 clear enable  参考 OC1CE |
| OC3M | 6:4 | Output Compare 3 mode  参考OC1M |
| OC3PE | 3 | Output Compare 3 preload enable  参考 OC1PE |
| OC3FE | 2 | Output Compare 3 fast enable  参考 OC1FE |
| CC3S | 1:0 | Capture/Compare 3 selection  00: CC3 通道配置成输出  01: CC3 通道配置成输入, TI3 作为 IC3 的输入  10: CC3 通道配置成输入, TI4 作为 IC3 的输入  11: CC3 通道配置成输入, TRC 作为 IC3 的输入. 只有通过 TS 位  (TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |

INPUT:

表 9-28 捕获/比较模式寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | IC4F | | | | IC4PSC | | CC4S | | IC3F | | | | IC3PSC | | CC3S | |

表 9-29 捕获/比较模式寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| IC4F | 15:12 | Input capture 4 filter  参考 IC1F 描述 |
| IC4PSC | 11:10 | Input capture 4 prescaler  参考 IC1PSC 描述 |
| CC4S | 9:8 | Capture/Compare 4 selection  00: CC4 通道配置成输出  01: CC4 通道配置成输入, TI4 作为 IC4 的输入  10: CC4 通道配置成输入, TI3 作为 IC4 的输入  11: CC4 通道配置成输入, TRC 作为 IC4 的输入. 只有通过 TS 位  (TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |
| IC3F | 7:4 | Input capture 3 prescaler  参考 IC1F |
| IC3PSC | 3:2 | Input capture 3 filter  参考 IC1PSC |
| CC3S | 1:0 | Capture/Compare 3 selection  00: CC3 通道配置成输出  01: CC3 通道配置成输入, TI3 作为 IC3 的输入  10: CC3 通道配置成输入, TI4 作为 IC3 的输入  11: CC3 通道配置成输入, TRC 作为 IC3 的输入. 只有通过 TS 位  (TIMx\_SMCR 寄存器)选择内部 Trigger 输入时， 这个模式才有效 |

### 捕获/比较模式使能寄存器(ccer)

表 9-30 获/比较模式使能寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | CC4P | CC4E | CC3NP | CC3NE | CC3P | CC3E | CC2NP | CC2NE | CC2P | CC2E | CC1NP | CC1NE | CC1P | CC1E |

表 9-31 获/比较模式使能寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:14 | 保留 |
| CC4P | 13 | Capture/Compare 4 output polarity  参考CC1P |
| CC4E | 12 | Capture/Compare 4 output enable  参考CC1E |
| CC3NP | 11 | Capture/Compare 3 complementary output polarity  参考CC1NP |
| CC3NE | 10 | Capture/Compare 3 complementary output enable  参考CC1NE |
| CC3P | 9 | Capture/Compare 3 output polarity  参考CC1P |
| CC3E | 8 | Capture/Compare 3 output enable  参考CC1E |
| CC2NP | 7 | Capture/Compare 2 complementary output polarity  参考CC1NP |
| CC2NE | 6 | Capture/Compare 2 complementary output enable  参考CC1NE |
| CC2P | 5 | Capture/Compare 2 output polarity  参考CC1P |
| CC2E | 4 | Capture/Compare 2 output enable  参考CC1E |
| CC1NP | 3 | Capture/Compare 1 complementary output polarity  OUTPUT：  0：OC1为高时活跃  1：OC1为低时活跃  INPUT：  CC1NP/CC1P位选择TI1FP1和TI2FP1的有源极性  触发或捕获操作 |
| CC1NE | 2 | Capture/Compare 1 complementary output enable  OUTPUT:  0：禁用通道1输出功能  1：使能通道1输出功能  INPUT:  0：禁用通道1捕获功能  1：使能通道1捕获功能 |
| CC1P | 1 | Capture/Compare 1 output polarity  OUTPUT：  0：OC1为高时活跃  1：OC1为低时活跃  INPUT：  CC1NP/CC1P位选择TI1FP1和TI2FP1的有源极性  触发或捕获操作 |

### 计数寄存器(cnt)

表 9-32 计数寄存器

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

表 9-33 计数寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| CNT | 15:0 | 计数值 |

### 预分频寄存器(psc)

表 9-34 预分频寄存器

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

表 9-35 预分频寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| PSC | 15:0 | 预分频 |

### 重载寄存器(arr)

表 9-36 重载寄存器

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

表 9-37 重载寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| ARR | 15:0 | ARR是要在实际的自动重新加载寄存器中加载的值。  当自动重新加载值为空时，计数器被阻止。 |

### 重复计数寄存器(rcr)

表 9-38 重复计数寄存器

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

表 9-39 重复计数寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| RCR | 7:0 | 重复计数寄存器  这些位允许用户设置比较的更新速率寄存器（即从预加载到活动寄存器的定期传输）启用预加载寄存器，以及更新中断生成速率，如果这个中断是启用的。每次REP\_CNT相关下行计数器达到零，将生成一个更新事件并重新启动从REP值计数。因为REP\_CNT重新加载REP值重复更新事件U\_RC，任何写入TIMx\_RCR寄存器的操作都是直到下一次重复更新事件才被考虑到。  它表示在PWM模式下(REP+1)对应于：  -边缘对齐模式下的PWM周期数  -中心对齐模式下的半PWM周期数 |

### 捕获/比较寄存器1(ccr1)

表 9-40 捕获/比较寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | CCR1 | | | | | | | | | | | | | | | |

表 9-41 捕获/比较寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| CCR1 | 15:0 | Capture/Compare value1  如果将通道CCn配置为输出：  CCRn是要在实际捕获/比较n寄存器中加载的值（预加载值）。如果未在TIMx\_CCMRn寄存器(位OCnPE)。否则，预加载值复制到当发生更新事件时，主动捕获/比较n个寄存器。这个活动捕获/比较寄存器包含要进行比较的值计数器TIMx\_CNT和信号在OCn输出。  如果将通道CCn配置为输入：  CCRn是由最后一个输入捕获n个事件传输的计数器值(ICn) |

### 捕获/比较寄存器2(ccr2)

表 9-42 捕获/比较寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | CCR2 | | | | | | | | | | | | | | | |

表 9-43 捕获/比较寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| CCR3 | 15:0 | Capture/Compare value3 |

### 捕获/比较寄存器3(ccr3)

表 9-44 捕获/比较寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | CCR3 | | | | | | | | | | | | | | | |

表 9-45 捕获/比较寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| CCR3 | 15:0 | Capture/Compare value3 |

### 捕获/比较寄存器4(ccr4)

表 9-46 捕获/比较寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | CCR4 | | | | | | | | | | | | | | | |

表 9-47 捕获/比较寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| CCR4 | 15:0 | Capture/Compare value4 |

### 中断和死线寄存器(bdtr)

表 9-48 中断和死线寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | MOE | AOE | BKP | BKE | OSSR | OSSI | LOCK | | DTG | | | | | | | |

表 9-49 中断和死线寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| MOE | 15 | Main output enable  0：MOE只能被软件写  1：MOE可以通过软件写，或者在下一个更新事件(如果BREAK输入被禁用时） |
| AOE | 14 | 保留 |
| BKP | 13 | Break polarity  0：BREAK输入BRK为低时活跃  1：BREAK输入BRK为高时活跃 |
| BKE | 12 | Break enable  0: 禁用BREAK输入  0: 使能BREAK输入 |
| OSSR | 11 | Off-state selection for Run mode  0：当不活动时，OC/OCN输出被禁用（OC/OCN启用输出  signal=0).  1: 当不活动时，将启用OC/OCN输出，其非活动级别为  很快就会成为CCxE=1或CCxNE=1。然后，OC/OCN启用输出信号=1 |
| OSSI | 10 | Off-state selection for Idle mode  0: 当不活动时，OC/OCN输出被禁用（OC/OCN启用输出  signal=0).  1：当不活动时，首先强制OC/OCN输出，其空闲水平为很快成为CcxE=1或CCxNE=1。OC/OCN启用输出信号=1) |
| LOCK | 9:8 | Lock configuration  0：禁用锁  1：LOCK\_level1：DTG，OISx，OISxN，BKE，BKP，AOE寄存器被保护，无法操作  2：LOCK\_level2：LOCK\_level1 + CC Polarity bits  3：LOCK\_level3：LOCK\_level2 + CC Control bits |
| DTG | 7:0 | Dead-time generator setup  此位字段定义了插入在互补的输出。DT对应于这个持续时间。  DTG[7:5]=0xx => DT=DTG[7:0]x tdtg with tdtg=tDTS.  DTG[7:5]=10x => DT=(64+DTG[5:0])xtdtg with Tdtg=2xtDTS.  DTG[7:5]=110 => DT=(32+DTG[4:0])xtdtg with Tdtg=8xtDTS.  DTG[7:5]=111 => DT=(32+DTG[4:0])xtdtg with Tdtg=16xtDTS.  例如 TDTS=125ns(8MHz)，dead-time可能是以下值：  0到15875ns，步进为125ns  16us到31750ns，步进为250ns  32us到63us，不急为1us  64us到126us，步进为2us |

### DMA控制寄存器(dmaen)

表 9-50 DMA控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | | | | | | TDE | COMDE | CC4DE | CC3DE | CC2DE | CC1DE | UDE |

表 9-51 DMA控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:7 | 保留 |
| TDE | 6 | Trigger DMA request enable  0: Trigger DMA 请求禁止.  1: Trigger DMA 请求使能 |
| COMDE | 5 | COM DMA request enable  0: COM DMA 请求禁止.  1: COM DMA 请求使能 |
| CC4DE | 4 | Capture/Compare 4 DMA request enable  0: CC4 DMA 请求禁止.  1: CC4 DMA 请求使能 |
| CC3DE | 3 | Capture/Compare 3 DMA request enable  0: CC3 DMA 请求禁止.  1: CC3 DMA 请求使能 |
| CC2DE | 2 | Capture/Compare 2 DMA request enable  0: CC2 DMA 请求禁止.  1: CC2 DMA 请求使能 |
| CC1DE | 1 | Capture/Compare 1 DMA request enable  0: CC1 DMA 请求禁止.  1: CC1 DMA 请求使能 |
| UDE | 0 | Update DMA request enable  0: Update DMA 请求禁止.  1: Update DMA 请求使能 |

# 通用异步收发器(UART)

## 概述

通用同步异步收发器（UART）提供了一个灵活的方式，使MCU可以与外部设备通过工业标准NRZ的形式实现全双工异步串行数据通讯。UART可以使用分数波特率发生器，提供了超宽的波特率设置范围。

UART支持非同步通讯模式和半双工单线通讯。也支持LIN（本地互联网络），智能卡协议和IrDA（红外数据协会）SIR ENDEC规范和modem流控操作（CTSn/RTSn）同时还支持多机通讯方式。

可以使用DMA实现多缓冲区设置，从而能够支持高速数据通讯。

* 全双工异步通信
* 16-byte接收和发送FIFOs
* 兼容16C550标准
* 可程序设计接收缓冲触发点
* 支持各信道独立波特率可程序设计
* 支持自动波特率检测
* 十二个中断源
* 可与DMA使用
* 利用 DMA 功能将收/发字节缓冲到保留的SRAM 空间
* 内置小数波特率发生器，覆盖范围广的波特率不需要特定值的外部晶体
* 可编程收发波特率高达3MHz，最低可达732Hz(时钟频率为16MHz)
* 支持硬件自动流控制/流控制功能（CTSn、RTSn），RTSn控制流触发点可程序设计
* Modem 硬件自動控制
* RS485 發送始能控制
* 支持CTSn唤醒功能
* 支持IrDA SIR模式
* 支持3/16位周期调制
* 支持RS-485
* 支持9-位模式
* 多处理器通信
* 完全可程序设计的串行接口特性
* 可程序设计数据位个数，即5-,6-,7-,8-,9-位
* 校验位，奇、偶、无校验或者固定校验位生成和检测可程序设计
* 停止位长度可程序设计：1，2位，在智能卡模式中支持0.5，1.5位
* 可设置高位在前或低位在前
* 单线半双工通讯
* 交換Tx/Rx pin 配置
* LIN主机的断开信号发送能力和LIN从机的断开信号检测能力
* 将UART设置为LIN模式时，有13位的断开信号发生器与断开信号检测功能
* 智能卡模式
* 支持ISO/IEC7816-3标准定义的T=0和T=1智能卡异步协议
* 智能卡使用的1.5停止位长度
* 支持ModBus通讯
* 超时检测功能
* CR/LF字符识别
* 杂讯侦测

## 结构框图



图 10‑1 UART框图

## 寄存器映射

UART1基地址：0x4002\_3000，大小4KB。

UART2基地址：0x4002\_4000，大小4KB。

表 10-1 寄存器映射(UART)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **寄存器** | **偏移量** | **读/写** | **描述** | **复位值** |
| RBR | 0x000 | R/W | 接收缓冲寄存器 | 0x0000 0000 |
| TBR | 0x004 | R/W | 发送缓冲寄存器 | 0x0000 0000 |
| BRR | 0x008 | R/W | 波特率寄存器 | 0x0000 0000 |
| LCR | 0x00C | R/W | 线路控制寄存器 | 0x0000 0000 |
| MCR | 0x010 | R/W | 状态控制寄存器 | 0x0000 0000 |
| CR | 0x014 | R/W | 控制寄存器 | 0x0000 0000 |
| RTOR | 0x018 | R/W | 接收器超时寄存器 | 0x0000 0000 |
| FCR | 0x01C | R/W | FIFO控制寄存器 | 0x0000 0000 |
| SR | 0x020 | R/W | 状态寄存器 | 0x0000 0000 |
| IER | 0x024 | W | 中断启用寄存器 | 0x0000 0000 |
| IDR | 0x028 | W | 中断禁用寄存器 | 0x0000 0000 |
| IVS | 0x02C | R | 中断有效状态 | 0x0000 0000 |
| RIF | 0x030 | R | 原始中断标志 | 0x0000 0000 |
| IFM | 0x034 | R | 中断标志已屏蔽 | 0x0000 0000 |
| ICR | 0x038 | W | 中断清除寄存器 | 0x0000 0000 |

## 寄存器说明

### UART接收寄存器(rbr)

表 10-2 UART接收寄存器

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

表 10-3 UART接收寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| RBR | 31:0 | 接收数据寄存器 |

### UART发送寄存器(tbr)

表 10-4 UART发送寄存器

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

表 10-5 UART发送寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| TBR | 31:0 | 发送数据寄存器 |

### UART波特率控制寄存器(brr)

表 10-6 UART波特率控制寄存器

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

表 10-7 UART波特率控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| BRR | 31:0 | 波特率寄存器  只有在UART没有发送和接收时可以执行写操作 |

### UART线路控制寄存器(lcr)

表 10-8 UART线路控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | RTO\_SEL | SWAP | TXINV | RXINV | DATAINV | MSBFIRST | RTOEN | BRWEN | BC | RXEN | PS | PE | STOP | DLS | |

表 10-9 UART线路控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:15 | 保留 |
| RTO\_SEL | 14 | 接收超时选择  0：CPU  1：DMA |
| SWAP | 13 | 交换TX/RX引脚  0：禁用  1：使能 |
| TXINV | 12 | TX引脚主动电平反转  0：TX引脚信号使用标准逻辑电平(VDD=1/idle，Gnd=0/标记）  1：TX引脚信号值被倒置。(VDD =0/mark, Gnd=1/idle). |
| RXINV | 11 | RX引脚主动电平反转  0：RX引脚信号使用标准逻辑电平(VDD=1/idle，Gnd=0/标记）  1：RX引脚信号值被倒置。(VDD =0/mark, Gnd=1/idle). |
| DATAINV | 10 | 二进制数据反演  0：数据寄存器的逻辑数据以正/直接发送/接收逻辑(1=H, 0=L)  1：数据首先通过MSB（位8/7/6/5）传输/接收，紧接着开始b |
| MSBFIRST | 9 | 最高优先  0：数据发送或者接收时，低bit优先  1：数据发送或者接收时，高bit优先 |
| RTOEN | 8 | 超时功能使能  0：禁用，1：使能 |
| BRWEN | 7 | 波特率写使能  0：不可用操作BRR寄存器  1：可以更新BRR寄存器 |
| BC | 6 | Break功能控制  0：禁用，1：使能 |
| RXEN | 5 | 接收使能  0：禁用，1：使能接收 |
| PS | 4 | 奇偶校验选择位  0：奇校验，1：偶校验 |
| PE | 3 | 奇偶校验使能位  0：禁用奇偶校验，1：使能奇偶校验 |
| STOP | 2 | 停止位  正常模式下：  0: 1 Stop bit  1: 2 Stop bit(if DLS[1:0]=11, it will be 1.5 Stop bit)  智能卡模式下：  0: 0.5 Stop bit  1: 1.5 Stop bit |
| DLS | 1:0 | 数据长度选择  00: 8-bit 数据格式  01: 7-bit数据格式  10: 6-bit数据格式  11: 5-bit数据格式t  Note: 9-bit 数据格式用于 RS485 |

### UART模式控制寄存器(mcr)

表 10-10 UART模式控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | HDSEL | CLKEN | SCCNT | | | SCNACK | SCEN | ABRRS | ABRMOD | | ABREN | DMAEN | LINBDL | BKREQ | LINEN | AADINV | AADDIR | AADNOR | AADEN | RTSCTRL | AFCEN | LBEN | IREN |

表 10-11 UART模式控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:23 | 保留 |
| HDSEL | 22 |  |
| CLKEN | 21 | 智能卡模式时钟使能  0：禁用  1：使能 |
| SCCNT | 20:18 | 智能卡自动重试次数  0：禁用重传输-在传输中没有自动重传输方式  1~7：自动重新传输尝试的次数（在令之前错误 |
| SCNACK | 17 | 智能卡NACK模式  0：禁用奇偶校验错误时的NACK传输  1：使能奇偶校验错误时的NACK传输 |
| SCEN | 16 | 智能卡模式使能  0：禁用，1：使能 |
| ABRRS | 15 |  |
| ABRMOD | 14:13 | 自动波特率模式  0：模式0检测下降边缘到第二个下降边缘  1：模式1检测下降边缘到第一上升边缘  2：模式2检测下降边缘到第一上升边缘  3：保留 |
| ABREN | 12 | 自动波特率功能使能  0：禁用， 1：使能 |
| DMAEN | 11 | DMA功能使能  0：禁用，1：使能 |
| LINBDL | 10 | LIN模式break方向长度  0:10-bit break长度  1:11-bit break长度 |
| BKREQ | 9 | LIN模式break请求  写入1到这个位设置请求发送中断，尽快发射机可用。 |
| LINEN | 8 | LIN模式使能位  0：禁用，1：使能 |
| AADINV | 7 | 自动地址检测反转  0：不反转，1：反转 |
| AADDIR | 6 |  |
| AADNOR | 5 |  |
| AADEN | 4 | 自动地址检测操作使能  0：禁用，1：使能 |
| RTSCTRL | 3 | RTSn控制位  0：RTSn电平为1， 1：RTSn电平为0 |
| AFCEN | 2 | 自动流控功能  0：禁用，1： 使能 |
| LBEN | 1 | 回环功能使能  0：禁用，1：使能 |
| IREN | 0 | lrDA模式使能  0：禁用，1：使能 |

### UART控制寄存器(cr)

表 10-12 UART控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | GT | | | | | | | | PSC | | | | | | | | DLY | | | | | | | | ADDR | | | | | | | |

表 10-13 UART控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| GT | 31:24 | 保护时间值 |
| PSC | 23:16 | 预分频值 |
| DLY | 15:8 | 延时时间  包含一个方向控制RTSn延迟值。此寄存器工作在  根据除数位的数量与一个8位计数器结合持续时间 |
| ADDR | 7:0 | 地址匹配值 |

### UART接收超时寄存器(rtor)

表 10-14 UART接收超时寄存器

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

表 10-15 UART接收超时寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| BLEN | 31:24 | Block length |
| RTO | 23:0 | 超时时间设置  23Bit(UART)，8Bit(UART)。 |

### UART管道控制寄存器(fcr)

表 10-16 UART管道控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | TXFL | | | | | RXFL | | | | | TXTL | | RXTL | | -- | TFRST | RFRST | FIFOEN |

表 10-17 UART管道控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:18 | 保留 |
| TXFL | 17:13 | 接收FIFO剩余数据  是由硬件设置，表示发送FIFO中剩余数据 |
| RXFL | 12:8 | 接收FIFO剩余数据  是由硬件设置，表示接收FIFO中剩余数据 |
| TXTL | 7:6 | 发送FIFO触发阈值  0：1-byte  1：2-byte  2：4-byte  3：8-byte |
| RXTL | 5:4 | 接收FIFO触发阈值  0：1-byte  1：4-byte  2：8-byte  3：14-byte |
| Reserved | 3 | 保留 |
| TFRST | 2 | 发送FIFO reset  0：无效，1：复位发送FIFO的指针 |
| RFRST | 1 | 接收FIFO reset  0：无效，1：复位接收FIFO的指针 |
| FIFOEN | 0 | FIFO功能使能  0：禁用  1：使能 |

### UART状态寄存器(sr)

表 10-18 UART状态寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | CTS | DCTS | RFF | RFNE | TFEM | TFNF | BUSY | RFE | TEM | TBEM | BF | FE | PE | OE | DR |

表 10-19 UART状态寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:15 | 保留 |
| CTS | 14 | 发送接触 |
| DCTS | 13 | Delta清除 |
| RFF | 12 | 接收FIFO满 |
| RFNE | 11 | 接收FIFO不空 |
| TFEM | 10 | 发送FIFO空 |
| TFNF | 9 | 发送FIFO不满 |
| BUSY | 8 | UART繁忙 |
| RFE | 7 | 接收FIFO错误 |
| TEM | 6 | 数据发空 |
| TBEM | 5 | 发送BUFF空 |
| BF | 4 | Break中断 |
| FE | 3 | 数据帧错误 |
| PE | 2 | 极性错误 |
| OE | 1 | 超时错误 |
| DR | 0 | 数据状态  0：没有数据，1：数据准备好 |

### UART中断使能寄存器(ier)

表 10-20 UART中断使能寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | CM | EOB | TC | LINBK | ABTO | ABE | BUSY | RTO | MDS | RXS | TXS | RXRD |

表 10-21 UART中断使能寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:12 | 保留 |
| CM | 11 | 字符匹配中断启用  0：无效，1：使能 |
| EOB | 10 | 块结束中断启用  0：无效，1：使能 |
| TC | 9 | 发送完成中断使能  0：无效，1：使能 |
| LINBK | 8 | LIN break方向中断使能  0：无效，1：使能 |
| ABTO | 7 | 自动波特率超时中断使能  0：无效，1：使能 |
| ABE | 6 | 自动波特率结束中断使能  0：无效，1：使能 |
| BUSY | 5 | 繁忙中断使能  0：无效，1：使能 |
| RTO | 4 | 超时中断使能  0：无效，1：使能 |
| MDS | 3 | Modem模式中断使能  0：无效，1：使能 |
| RXS | 2 | 接收的中断状态使能  0：无效，1：使能 |
| TXS | 1 | 发送的中断状态使能  0：无效，1：使能 |
| RXRD | 0 | 接收中断使能  0：无效，1：使能 |

### UART中断关闭寄存器(idr)

表 10-22 UART中断关闭寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | CM | EOB | TC | LINBK | ABTO | ABE | BUSY | RTO | MDS | RXS | TXS | RXRD |

表 10-23 UART中断关闭寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:12 | 保留 |
| CM | 11 | 字符匹配中断关闭  0：无效，1：关闭 |
| EOB | 10 | 块结束中断关闭  0：无效，1：关闭 |
| TC | 9 | 发送完成中断关闭  0：无效，1：关闭 |
| LINBK | 8 | LIN break方向中断关闭  0：无效，1：关闭 |
| ABTO | 7 | 自动波特率超时中断关闭  0：无效，1：关闭 |
| ABE | 6 | 自动波特率结束中断关闭  0：无效，1：关闭 |
| BUSY | 5 | 繁忙中断关闭  0：无效，1：关闭 |
| RTO | 4 | 超时中断关闭  0：无效，1：关闭 |
| MDS | 3 | Modem模式中断关闭  0：无效，1：关闭 |
| RXS | 2 | 接收的中断状态关闭  0：无效，1：关闭 |
| TXS | 1 | 发送的中断状态关闭  0：无效，1：关闭 |
| RXRD | 0 | 接收中断关闭  0：无效，1：关闭 |

### UART中断有效状态寄存器(ivs)

表 10-24 UART中断有效状态寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | CM | EOB | TC | LINBK | ABTO | ABE | BUSY | RTO | MDS | RXS | TXS | RXRD |

表 10-25 UART中断有效状态寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:12 | 保留 |
| CM | 11 | 字符匹配中断有效状态  0：无效状态，1：有效状态 |
| EOB | 10 | 块结束中断有效状态  0：无效状态，1：有效状态 |
| TC | 9 | 发送完成中断有效状态  0：无效状态，1：有效状态 |
| LINBK | 8 | LIN break方向中断有效状态  0：无效状态，1：有效状态 |
| ABTO | 7 | 自动波特率超时中断有效状态  0：无效状态，1：有效状态 |
| ABE | 6 | 自动波特率结束中断有效状态  0：无效状态，1：有效状态 |
| BUSY | 5 | 繁忙中断有效状态  0：无效状态，1：有效状态 |
| RTO | 4 | 超时中断有效状态  0：无效状态，1：有效状态 |
| MDS | 3 | Modem模式中断有效状态  0：无效状态，1：有效状态 |
| RXS | 2 | 接收的中断状态有效状态  0：无效状态，1：有效状态 |
| TXS | 1 | 发送的中断状态有效状态  0：无效状态，1：有效状态 |
| RXRD | 0 | 接收中断有效状态  0：无效状态，1：有效状态 |

### UART中断标志状态寄存器(rif)

表 10-26 UART中断标志状态寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | CM | EOB | TC | LINBK | ABTO | ABE | BUSY | RTO | MDS | RXS | TXS | RXRD |

表 10-27 UART中断标志状态寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:12 | 保留 |
| CM | 11 | 字符匹配中断有效标志  0：无效标志，1：有效标志 |
| EOB | 10 | 块结束中断启用  0：无效标志，1：有效标志 |
| TC | 9 | 发送完成中断有效标志  0：无效标志，1：有效标志 |
| LINBK | 8 | LIN break方向中断有效标志  0：无效标志，1：有效标志 |
| ABTO | 7 | 自动波特率超时中断有效标志  0：无效标志，1：有效标志 |
| ABE | 6 | 自动波特率结束中断有效标志  0：无效标志，1：有效标志 |
| BUSY | 5 | 繁忙中断有效标志  0：无效标志，1：有效标志 |
| RTO | 4 | 超时中断有效标志  0：无效标志，1：有效标志 |
| MDS | 3 | Modem模式中断有效标志  0：无效标志，1：有效标志 |
| RXS | 2 | 接收的中断状态有效标志  0：无效标志，1：有效标志 |
| TXS | 1 | 发送的中断状态有效标志  0：无效标志，1：有效标志 |
| RXRD | 0 | 接收中断有效状态  0：无效状态，1：有效状态 |

### UART中断屏蔽寄存器(ifm)

表 10-28 UART中断屏蔽寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | CM | EOB | TC | LINBK | ABTO | ABE | BUSY | RTO | MDS | RXS | TXS | RXRD |

表 10-29 UART中断屏蔽寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:12 | 保留 |
| CM | 11 | 字符匹配中断标志屏蔽  0：无效，1：屏蔽 |
| EOB | 10 | 块结束中断标志屏蔽  0：无效，1：屏蔽 |
| TC | 9 | 发送完成中断标志屏蔽  0：无效，1：屏蔽 |
| LINBK | 8 | LIN break方向中断标志屏蔽  0：无效，1：屏蔽 |
| ABTO | 7 | 自动波特率超时中断标志屏蔽  0：无效，1：屏蔽 |
| ABE | 6 | 自动波特率结束中断标志屏蔽  0：无效，1：屏蔽 |
| BUSY | 5 | 繁忙中断标志屏蔽  0：无效，1：屏蔽 |
| RTO | 4 | 超时中断标志屏蔽  0：无效，1：屏蔽 |
| MDS | 3 | Modem模式中断标志屏蔽  0：无效，1：屏蔽 |
| RXS | 2 | 接收的中断状态标志屏蔽  0：无效，1：屏蔽 |
| TXS | 1 | 发送的中断状态标志屏蔽  0：无效，1：屏蔽 |
| RXRD | 0 | 接收中断标志屏蔽  0：无效，1：屏蔽 |

### UART中断清除寄存器(icr)

表 10-30 UART中断清除寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | | | | | | | | | CM | EOB | TC | LINBK | ABTO | ABE | BUSY | RTO | MDS | RXS | TXS | RXRD |

表 10-31 UART中断清除寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:12 | 保留 |
| CM | 11 | 字符匹配中断清除  0：无效，1：清除 |
| EOB | 10 | 块结束中断清除  0：无效，1：清除 |
| TC | 9 | 发送完成中断清除  0：无效，1：清除 |
| LINBK | 8 | LIN break方向中断清除  0：无效，1：清除 |
| ABTO | 7 | 自动波特率超时中断清除  0：无效，1：清除 |
| ABE | 6 | 自动波特率结束中断清除  0：无效，1：清除 |
| BUSY | 5 | 繁忙中断清除  0：无效，1：清除 |
| RTO | 4 | 超时中断清除  0：无效，1：清除 |
| MDS | 3 | Modem模式中断清除  0：无效，1：清除 |
| RXS | 2 | 接收的中断状态清除  0：无效，1：清除 |
| TXS | 1 | 发送的中断状态清除  0：无效，1：清除 |
| RXRD | 0 | 接收中断清除  0：无效，1：清除 |

# 外部中断/事件控制器(EXTI)

## 概述

扩展中断和事件控制器(EXTI)通过可配置的和直接的事件输入（行）来管理CPU和系统唤醒。它向电源控制提供唤醒请求，并向CPU NVIC生成中断请求，向CPU事件输入生成事件。对于CPU，需要一个额外的事件生成块(EVG)来生成CPU事件信号。

EXTI唤醒请求允许系统从停止模式中被唤醒。

中断请求和事件请求生成也可以在运行模式中使用。

EXTI的主要功能是：

系统会在任何输入唤醒标志事件上被唤醒

可配置的事件（来自I/O，没有关联中断挂起状态位的外设，或产生脉冲的外设）

-可选择的主动触发边缘

-独立的上升和下降边缘中断等待状态位

-个人中断和事件生成掩码，用于调节CPU唤醒，中断和事件生成

-软件触发可能性直接事件（从外围设备有一个相关的标志和中断等待状态位）

-固定上升边缘主动触发

-EXTI没有中断等待状态位

-个人中断和事件生成面具调节CPU唤醒和事件生成

-没有软件触发可能性

I/O端口选择器

## 寄存器映射

EXTI基地址：0x4002\_8000，大小4KB。

11-1 寄存器映射(EXTI)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 寄存器 | 偏移量 | 读/写 | 描述 | 复位值 |
| IER | 0x000 | W | 中断启用寄存器 | 0x0000 0000 |
| IDR | 0x004 | W | 中断禁用寄存器 | 0x0000 0000 |
| IVS | 0x008 | R | 中断有效状态寄存器 | 0x0000 0000 |
| RIF | 0x00C | R | 原始中断标志状态寄存器 | 0x0000 0000 |
| IFM | 0x010 | R | 中断标志屏蔽状态寄存器 | 0x0000 0000 |
| ICR | 0x014 | W | 中断清除寄存器 | 0x0000 0000 |
| RTS | 0x018 | R/W | 上升的边缘触发器选择寄存器 | 0x0000 0000 |
| FTS | 0x01C | R/W | 落边触发器选择寄存器 | 0x0000 0000 |
| SWI | 0x020 | R/W | 软件中断事件寄存器 | 0x0000 0000 |
| ADTE | 0x024 | R/W | AD触发器启用 | 0x0000 0000 |
| DB | 0x028 | R/W | 输入衰减使能寄存器 | 0x0000 0000 |
| DBC | 0x02C | R/W | 输入取消采样率控制寄存器 | 0x0000 0000 |

## 寄存器说明

### 外部中断使能寄存器(ier)

表 11-2 外部中断使能寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | GPIO19 | GPIO18 | GPIO17 | GPIO16 | GPIO15 | GPIO14 | GPIO13 | GPIO12 | GPIO11 | GPIO10 | GPIO9 | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 | GPIO0 |

表 11-3 外部中断使能寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| GPIO19~GPIO0 | 19:0 | EXTI<n>中断使能位  0：无效，1：中断使能 |

### 外部中断关闭寄存器(idr)

表 11-4 外部中断关闭寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | GPIO19 | GPIO18 | GPIO17 | GPIO16 | GPIO15 | GPIO14 | GPIO13 | GPIO12 | GPIO11 | GPIO10 | GPIO9 | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 | GPIO0 |

表 11-5 外部中断关闭寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| GPIO19~GPIO0 | 19:0 | EXTI<n>中断关闭位  0：无效，1：中断关闭 |

### 外部中断有效状态寄存器(ivs)

表 11-6 外部中断有效状态寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | GPIO19 | GPIO18 | GPIO17 | GPIO16 | GPIO15 | GPIO14 | GPIO13 | GPIO12 | GPIO11 | GPIO10 | GPIO9 | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 | GPIO0 |

表 11-7 外部中断有效状态寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| GPIO19~GPIO0 | 19:0 | EXTI<n>中断有效状态位  0：中断无效，1：中断有效 |

### 外部中断标志状态寄存器(rif)

表 11-8 外部中断标志状态寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | GPIO19 | GPIO18 | GPIO17 | GPIO16 | GPIO15 | GPIO14 | GPIO13 | GPIO12 | GPIO11 | GPIO10 | GPIO9 | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 | GPIO0 |

表 11-9 外部中断标志状态寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| GPIO19~GPIO0 | 19:0 | EXTI<n>中断标志状态位  0：未检测到有效中断，1：检测到有效中断 |

### 外部中断屏蔽寄存器(ifm)

表 11-10 外部中断屏蔽寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | GPIO19 | GPIO18 | GPIO17 | GPIO16 | GPIO15 | GPIO14 | GPIO13 | GPIO12 | GPIO11 | GPIO10 | GPIO9 | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 | GPIO0 |

表 11-11 外部中断屏蔽寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| GPIO19~GPIO0 | 19:0 | EXTI<n>中断屏蔽寄存器  0：未屏蔽中断，1：屏蔽中断 |

### 外部中断清除寄存器(icr)

表 11-12 外部中断清除寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | GPIO19 | GPIO18 | GPIO17 | GPIO16 | GPIO15 | GPIO14 | GPIO13 | GPIO12 | GPIO11 | GPIO10 | GPIO9 | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 | GPIO0 |

表 11-13 外部中断清除寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| GPIO19~GPIO0 | 19:0 | EXTI<n>中断清零寄存器  0：无效，1：中断标志清零 |

### 外部中断上升沿触发寄存器(rts)

表 11-14 外部中断上升沿触发寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | GPIO19 | GPIO18 | GPIO17 | GPIO16 | GPIO15 | GPIO14 | GPIO13 | GPIO12 | GPIO11 | GPIO10 | GPIO9 | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 | GPIO0 |

表 11-15 外部中断上升沿触发寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| GPIO19~GPIO0 | 19:0 | EXTI<n>上升沿触发使能位  0：禁止，1：使能 |

### 外部中断下降沿触发寄存器(fts)

表 11-16 外部中断下降沿触发寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | GPIO19 | GPIO18 | GPIO17 | GPIO16 | GPIO15 | GPIO14 | GPIO13 | GPIO12 | GPIO11 | GPIO10 | GPIO9 | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 | GPIO0 |

表 11-17 外部中断下降沿触发寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| GPIO19~GPIO0 | 19:0 | EXTI<n>下降沿触发使能位  0：禁止，1：使能 |

### 外部中断软件触发寄存器(swi)

表 11-18 外部中断软件触发寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | GPIO19 | GPIO18 | GPIO17 | GPIO16 | GPIO15 | GPIO14 | GPIO13 | GPIO12 | GPIO11 | GPIO10 | GPIO9 | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 | GPIO0 |

表 11-19 外部中断软件触发寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| GPIO19~GPIO0 | 19:0 | EXTI<n>中断软件触发位  0：无操作，1：软件触发 |

### 外部中断定时器触发寄存器(adte)

表 11-20 外部中断定时器触发寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | GPIO19 | GPIO18 | GPIO17 | GPIO16 | GPIO15 | GPIO14 | GPIO13 | GPIO12 | GPIO11 | GPIO10 | GPIO9 | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 | GPIO0 |

表 11-21 外部中断定时器触发寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 |  |
| GPIO19~GPIO0 | 19:0 | EXTI<n>中断AD触发位  0：无操作，1：软件触发 |

### 外部中断消抖使能寄存器(db)

表 11-22 外部中断消抖使能寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Bit** | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| **Name** | -- | | | | | | | | | | | | GPIO19 | GPIO18 | GPIO17 | GPIO16 | GPIO15 | GPIO14 | GPIO13 | GPIO12 | GPIO11 | GPIO10 | GPIO9 | GPIO8 | GPIO7 | GPIO6 | GPIO5 | GPIO4 | GPIO3 | GPIO2 | GPIO1 | GPIO0 |

表 11-23 外部中断消抖使能寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:20 | 保留 |
| GPIO19~GPIO0 | 19:0 | EXTI<n>中断滤波触发位  0：禁止，1：使能 |

### 外部中断消抖控制寄存器(dbc)

表 11-24 外部中断消抖控制寄存器

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

表 11-25 外部中断消抖控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:16 | 保留 |
| DBPRE | 15:8 | 外部中断前置滤波选择位 |
| Reserved | 7:3 | 保留 |
| DBCNT | 2:0 | 外部中断去抖滤波选择位 |

# 集成内部总线接口(I2C)

## 概述

1路I2C总线，支持快速模式速率可达400Kbps。I2C是两线双向的串列传输总线，提供了一种简单有效的方法来实现设备之间的数据交换。I2C标准是一个多主机总线包括衝突检测与仲裁，如果两个或两个以上的主机试图同时控制总线时，其仲裁可以防止数据损坏。在此提供了标准模式(Sm)、快速模式(Fm)与极快速模式(Fm+)供使用者选择。并且也提供SMBus(系统管理总线)与PMBus(电源管理总线)。

* 可配置为主机或从机(Slave and master modes)
* 多主机模式(Multimaster capability)
* 标准模式(up to 100 kHz)(Standard-mode)
* 快速模式(up to 400 kHz)(Fast-mode)
* 7位元与10位元地址模式(7-bit and 10-bit addressing mode)
* 提供2组7位元从机地址(2个地址，其中一个包括遮罩)(Multiple 7-bit slave addresses (2 addresses, 1 with configurable mask)
* 提供所有7位元地址应答模式(All 7-bit addresses acknowledge mode )
* 提供广播模式(General call)
* 可编程的设置时间和保持时间(Programmable setup and hold times)
* 可选择时脉延长(Optional clock stretching)
* 可配置数位滤波器(Programmable digital noise filters)
* 提供深度为8位元组的TX / RX FIFOs
* 提供SMBus标准(SMBus capability)
* 硬体PEC(封包错误检查)产生与ACK控制(Hardware PEC(Packet Error Checking) generation and verification with ACK
* 命令与数据应答控制(Command and data acknowledgecontrol)
* 提供地址解析协定(Address resolution protocol(ARP) support)
* 提供可选择为主机或设备(Host and Device support)
* 提供SMBus警报(SMBus alert)
* 提供超时和空闲状态检测(Timeouts and idle condition detection)
* 提供PMBus rev 1.1标准(PMBusspecification rev 1.1 compatibility)

## 寄存器映射

I2C基地址：0x4002\_9000，大小4KB。

表 12-1 寄存器映射(I2C)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **寄存器** | **偏移量** | **读/写** | **描述** | **复位值** |
| CONSET | 0x000 | R/W | I2C控制集寄存器 | 0x0000 0000 |
| CONCLR | 0x004 | W | I2C控制清除寄存器 | 0x0000 0000 |
| SR | 0x008 | R | I2C状态寄存器 | 0x0000 0000 |
| DATA | 0x00C | R/W | I2C数据寄存器 | 0x0000 0000 |
| CLK | 0x010 | R/W | I2C时钟寄存器 | 0x0000 0000 |
| SADR0 | 0x014 | R/W | 从机地址0寄存器 | 0x0000 0000 |
| ADM0 | 0x018 | R/W | 从机地址掩码0寄存器 | 0x0000 0000 |
| XSADR | 0x01C | R/W | x从机地址寄存器 | 0x0000 0000 |
| XADM | 0x020 | R/W | x从机地址掩码寄存器 | 0x0000 0000 |
| SRST | 0x024 | W | 软复位 | 0x0000 0000 |
| SADR1 | 0x028 | R/W | 从机地址1寄存器 | 0x0000 0000 |
| ADM1 | 0x02C | R/W | 从机地址掩码1寄存器 | 0x0000 0000 |
| SADR2 | 0x030 | R/W | 从机地址2寄存器 | 0x0000 0000 |
| ADM2 | 0x034 | R/W | 从机地址掩码2寄存器 | 0x0000 0000 |
| SADR3 | 0x038 | R/W | 从机地址3寄存器 | 0x0000 0000 |
| ADM3 | 0x03C | R/W | 从机地址掩码3寄存器 | 0x0000 0000 |

## 寄存器说明

### 控制设置寄存器(cr\_set)

表 12-2 控制设置寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | -- | | | | | | | | | | | | | | | | | | | | | | | GCAVAL | IEN | I2CEN | STA | STP | IFLG | AAK | SL10VAL | SL7VAL |

表 12-3 控制设置寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:9 | 保留 |
| GCAVAL | 8 | 广播地址寻址模式  0：禁用， 1：使能 |
| IEN | 7 | 中断使能位  0：无效果， 1：使能 |
| I2CEN | 6 | 0：无效果， 1：使能  当EN=1，I2C收到对应的slave地址时会回复ACK，如果 GCE 位使能，收到广播地址时也会回复ACK。如果 EN=0，I2C不会响应任何总线上的地址。  当 I2C在 master 模式下运行时，EN并不需要设置。但是如果EN为0，当仲裁失败时，不论收到的地址是slave地址还是广播地址，返回的状态代码是38h。收到的地址数据仍然能从数据中读取寄存器。 |
| STA | 5 | 当STA写1时，I2C进入master模式并在总线空闲时发送 start信号。如果I2C已经工作在master模式，并且已经发送了一个或多个字节时，会发送一个restart信号。  如果I2C工作在slave模式，把STA位设置为1，I2C将在slave 模式下完成数据传输然后在总线被释放后进入master模式。发送完启动信号后， STA 位会自动清零 |
| STP | 4 | 当I2C工作在master模式时，设置STP为高，会发送停止信号。如果I2C工作在slave模式，设置STP为高，则slave 会认为接收到了停止信号，但不会发送停止信号。  这一比特会自动清零。 |
| IFLG | 3 | 当I2C进入29个状态(参考 STAT 寄存器状态描述)中任何一个状态时，IFLG都会被拉高。只有在状态F8h时， IFLG 不会拉高。  如果IFLG拉高，且IEN位是设置为 1 时，中断信号拉高。当 IFLG 拉高时，SCL 会被一直拉低，数据传输会暂停。当对 IFLG 写零时，中断清除且SCL释放。 |
| AAK | 2 | 如果满足以下其中一个条件，当 AAK 设置为 1 时， I2C 会发送一个 ACK信号（SDA上的低电平）：  1.收到一个匹配的7位从地址或者10位地址模式时的的第一个字节或第二个字节。  2.收到广播地址，并且GCAVAL设置为 1。  3.在 master 或 slave 模式收到数据。  注意：如果AAK为0时，在master 或 slave模式接收到数据字节时，不会发送ACK。 |
| SL10VAL | 1 | I2C 10bit寻址模式  0：禁用， 1：使能 |
| SL7VAL | 0 | I2C 7bit寻址模式  0：禁用， 1：使能 |

### 控制清除寄存器(cr\_clr)

表 12-4 控制清除寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | -- | | | | | | | | | | | | | | | | | | | | | | | | IEC | I2CENC | STAC | -- | IFLG | AAC | -- | |

表 12-5 控制清除寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| IEC | 7 | 写1清除IE，写0无效 |
| I2CENC | 6 | 写1清除I2CEN，写0无效 |
| STAC | 5 |  |
| Reserved | 4 | 保留 |
| IFLG | 3 |  |
| AAC | 2 | 写1清除AAK，写0无效 |
| Reserved | 1：0 | 保留 |

### 状态寄存器(sr)

表 12-6状态寄存器(sr)

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

表 12-7 状态寄存器(sr)定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| STATUS | 7:3 | 反应当前I2C处于何种状态  该寄存器高 5 比特对应不同状态编码：低 3 比特始终为零。状态代码如下所示：  00H：总线错误(master模式) 08H：起始信号发送  10H：重起始信号发送 18H：地址+写标示位传输，收到ACK  20 H：地址+写标示位传输，未收到ACK  28 H：Master模式发送数据，收到ACK  30 H：Master模式发送数据，未收到ACK  38 H：地址或者数据阶段仲裁失败  40 H：地址+读标示位发送，收到ACK  48 H：地址+读标示位发送，未收到ACK  50 H：Master模式收到数据，发送ACK  58 H：Master模式收到数据，未发送ACK  60 H：Slave模式地址+写标示位收到，发送ACK  68 H：Master模式在地址仲裁失败，收到地址+写标示位，发送ACK 70 H：收到广播地址，发送ACK  78 H：Master模式在地址仲裁失败，收到广播地址发送ACK  80 H：收到Slave地址后，收到数据，发送ACK  88 H：收到Slave地址后，收到数据，未发送ACK  90 H：收到广播地址后，收到数据，发送ACK  98 H：收到广播地址后，收到数据，未发送ACK  A0 H：在slave模式下收到停止或启动条件  A8 H：收到slave地址+读标示位，发送ACK  B0 H：Master模式在地址仲裁丢失，收到slave地址+读标示位，发送ACK  B8 H：slave模式发送数据，收到ACK  C0 H：slave模式发送数据，未收到ACK  C8 H：slave模式发送最后一个字节，收到ACK  D0 H：slave模式发送最后一个字节，未收到ACK  D8 H：未使用  E0 H：第二个地址字节发送，收到ACK  E8 H：第二个地址字节发送，未收到ACK  F0 H：未使用  F8 H：空闲态，IFLG未0 |
| Reserved | 29:24 | 保留 |
| TENMS | 23:0 |  |

### 数据寄存器(dr)

表 12-8 数据寄存器

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

表 12-9 数据寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| DATA | 7:0 | 发送的数据/发送的从地址或刚刚接收到数据  发送时先发送MSB  无论主从模式，只有在IFLG为1时才能读写 |

### 时钟寄存器(clk)

表 12-10 时钟寄存器

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

表 12-11 时钟寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved |  | 保留 |
| N | 6:4 | 采样频率：sample clk = Pclk/(2^N) |
| M | 3:0 | I2C 主机模式scl频率 scl clk = Pclk/(2^N \* (M+1) \* 10) |

### 从机0地址寄存器(sadr0)

表 12-12 从机0地址寄存器

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

表 12-13 从机0地址寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| SADR | 7:1 | 从机地址(7位寻址模式) |
| GCEBAB | 0 | 广播地址  0：无效， 1：该地址是广播地址 |

### 从机0地址屏蔽寄存器(adm0)

表 12-14 从机0地址屏蔽寄存器

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

表 12-15 从机0地址屏蔽寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| MASK | 7:1 | SADR0地址MASK |
| Reserved | 0 | 保留 |

### 拓展从机地址寄存器(xsadr)

表 12-16 拓展从机地址寄存器

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

表 12-17 拓展从机地址寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:11 | 保留 |
| ADDR | 10:1 | 从机地址(10位寻址模式) |
| XGCENAB | 0 | 广播地址  0：无效， 1：该地址是广播地址 |

### 拓展从机地址屏蔽寄存器(xadm)

表 12-18 拓展从机地址屏蔽寄存器

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

表 12-19 拓展从机地址屏蔽寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:11 | 保留 |
| MASK | 10:1 | XSADR地址MASK |
| Reserved | 0 | 保留 |

### 软件复位寄存器(srst)

表 12-20 软件复位寄存器

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

表 12-21 软件复位寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| SOFTWARE RESET | 31:0 | 写1将I2C模块复位 |

### 从机1地址寄存器(sadr1)

表 12-22 从机1地址寄存器

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

表 12-23 从机1地址寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| SADR | 7:1 | 从机地址(7位寻址模式) |
| GCEBAB | 0 | 广播地址  0：无效， 1：该地址是广播地址 |

### 从机1地址屏蔽寄存器(adm1)

表 12-24 从机1地址屏蔽寄存器

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

表 12-25 从机1地址屏蔽寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| MASK | 7:1 | SADR1地址MASK |
| Reserved | 0 | 保留 |

### 从机2地址寄存器(sadr2)

表 12-26 从机2地址寄存器

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

表 12-27 从机2地址寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| SADR | 7:1 | 从机地址(7位寻址模式) |
| GCEBAB | 0 | 广播地址  0：无效， 1：该地址是广播地址 |

### 从机2地址屏蔽寄存器(adm2)

表 12-28 从机2地址屏蔽寄存器

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

表 12-29 从机2地址屏蔽寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| MASK | 7:1 | ADR2地址MASK |
| Reserved | 0 | 保留 |

### 从机3地址寄存器(sadr3)

表 12-30 从机3地址寄存器

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

表 12-31 从机3地址寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| SADR | 7:1 | 从机地址(7位寻址模式) |
| GCEBAB | 0 | 广播地址  0：无效， 1：该地址是广播地址 |
| TENMS | 23:0 | 保留 |

### 从机3地址屏蔽寄存器(adm3)

表 12-32 从机3地址屏蔽寄存器

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

表 12-33 从机3地址屏蔽d寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:8 | 保留 |
| MASK | 7:1 | ADR3地址MASK |
| Reserved | 0 | 保留 |

# 独立看门狗(IWDG)

## 概述

独立看门狗IWDT，当硬件使能时，时钟强制为独立的32KHz LRC时钟，可用于检测软件和硬件异常，如主时钟停振，程序跑飞不再喂狗等。

* 支持硬件使能和关闭看门狗
* 芯片配置位CFG\_WDTEN位配置为1，则硬件使能IWDT
* 芯片配置位CFG\_WDTEN位配置为0，则通过软件可使能IWDT
* 硬件使能后不可通过软件关停
* 硬件使能后IWDT时钟强制为32KHz LRC时钟
* IWDT溢出时间可设定
* 写入IWDT\_LOAD寄存器将重新加载看门狗
* 溢出时产生IWDT复位
* IWDT中断可唤醒STOP1、2模式
* 在Debug时进入halting mode时，由IWDT\_DBGEN关停或使能IWDT （默认为使能），此时原来的控制信号EN不起作用
* **默认8s不喂狗产生复位**

## 寄存器映射

IWDG基地址：0x4002\_A000，大小4KB。

表 13-1 寄存器映射(IWDG)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **寄存器** | **偏移量** | **读/写** | **描述** | **复位值** |
| LOAD | 0x00 | W | 计数器重载寄存器 | 0x0002 0000 |
| VALUE | 0x04 | R | 当前计数值寄存器 | 0x0000 0000 |
| CTRL | 0x08 | R/W | 控制寄存器 | 0x0000 000D |
| INICLR | 0x0C | W | 中断标志清除寄存器 | 0x0000 0000 |
| RIS | 0x10 | W | 中断标志寄存器 | 0x0000 0000 |
| … | … | … | … | … |
| LOCK | 0x100 | R/W | 锁寄存器 | 0x0000 0000 |

## 寄存器说明

### 计数重载值寄存器(load)

表 13-2 计数重载值寄存器

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

表 13-3 计数重载值寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| LOAD | 31:0 | 独立看门狗的计数器重载值  计数范围 0x0000\_0001~0xFFFF\_FFFF。如果为 0， IWDT 不计数。[默认值：0x20000] |

### 当前计数值寄存器(value)

表 13-4 当前计数值寄存器

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

表 13-5 当前计数值寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| VALUE | 31:0 | IWDT 计数器当前值  读取时返回 WWDT 计数器的当前计数值 |

### 控制寄存器(ctrl)

表 13-6 控制寄存器

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Name | -- | | | | | | | | | | | | | | | | | | | | | | | | | | | | CLKSEL | RSTEN | INTEN | EN |

表 13-7 控制寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:6 | 保留 |
| CLKSEL | 3 | IWDG计数时钟选择位  0：PCLK， 1：LSI(RC32K) |
| RSTEN | 2 | IWDG复位使能位  0：禁用， 1：使能，IWDG计数到0时，产生复位信号，将芯片复位 |
| INTEN | 1 | IWDG中断使能位  0：禁用， 1：使能，IWDG计数到0时产生中断标志 |
| EN | 0 | IWDG外设使能位  0：禁用，1：使能 |

### 中断标志清除寄存器(iniclr)

表 13-8 中断标志清除寄存器

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

表 13-9 中断标志清除寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| INICLR | 31:0 | IWDG 中断标志清 0 位  对 IWDG\_INTCLR 寄存器进行任意写操作， IWDG 中断标  志位均被清零，计数器重载 IWDG\_LOAD 寄存器值，继续递减计数 |

### 中断标志寄存器(ris)

表 13-10 中断标志寄存器

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

表 13-11 中断标志寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| Reserved | 31:1 | 保留 |
| RIS | 0 | IWDG 中断标志位  0：未产生中断, 1：IWDG 计数器计数到 0，产生中断  写寄存器 IWDG\_INTCLR，可清除 IWDG 中断标志位 |

### 锁存寄存器(lock)

表 13-12 时钟配置(clkctrl)寄存器

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

表 13-13 时钟配置(stcalib)寄存器定义

|  |  |  |
| --- | --- | --- |
| **名称** | **比特** | **描述** |
| LOCK | 31:0 | IWDG 寄存器保护状态位  0： IWDG 寄存器处于未保护状态  1： IWDG 寄存器处于保护状态  对 IWDG\_LOCK 寄存器写入 0x1ACCE551，被保护的寄存器处于未保护状态；写入其它值，处于保护状态 |