

CMT2310A寄存器说明

概要

本应用文档为使用 CMT2310A 进行产品开发的用户提供寄存器介绍，以方便用户在使用过程查阅各寄存器的说明及用法。

本文档涵盖的产品型号如下表所示。

表 1. 本文档涵盖的产品型号

产品型号	工作频率	调制方式	主要功能	配置方式	封装
CMT2310A	117 - 1050 MHz	(4)(G)FSK/OOK	收发一体机	寄存器	QFN24

用户需要结合阅读以下的应用文档，以了解全部的信息来辅助软硬件开发。

《AN238 CMT2310A 射频参数配置指南》

《AN235 CMT2310A FIFO 和包格式使用指南》

《AN237 CMT2310A 快速上手指南》

《AN239 CMT2310A 自动收发功能使用指南》

目 录

1	芯片寄存器介绍.....	3
1.1	PAGE 0, 控制寄存器 (0x00 - 0x27)	3
1.2	PAGE 0, 配置寄存器 (0x28 - 0x77)	13
1.3	PAGE 1, 配置寄存器 (0x00 - 0x68)	25
2	文档变更记录	26
3	联系方式.....	27

1 芯片寄存器介绍

1.1 PAGE 0, 控制寄存器 (0x00 - 0x27)

这个区域的寄存器，主要用于控制芯片的模式跳转，和操作芯片的中断机制。

表 2. PAGE 0 (0x00 - 0x27) 控制寄存器说明

寄存器名	位数	R/W	比特名	功能说明
CTL_REG_0 (0x00)	7:0	W	PU_BOOT <7:0>	发送 0x03 会让芯片从 IDLE 状态进行上电，上电完成后停留在 SLEEP 状态。
CTL_REG_1 (0x01)	7:0	W	CHIP_MODE_SW<7:0>	芯片切换状态的命令： 00000001: go_sleep 00000010: go_ready 00000100: go_tx 00001000: go_rx 00010000: go_tfs 00100000: go_rfs 除了以上的 6 个值，其余的值无效。
CTL_REG_2 (0x02)	7:0	W	FIFO_RW_PORT<7:0>	这不是一个寄存器，而是进行 FIFO 读写操作的端口，详情请参阅串口操作说明。
CTL_REG_3 (0x03)	7:0	W	REG_CRW_PORT<7:0>	这不是一个寄存器，而是进行寄存器批量读写操作的端口，详情请参阅串口操作说明。
CTL_REG_4 (0x04)	6	RW	TX_DIN_EN	0: 屏蔽发射数据从 GPIO 输入 1: 使能发射数据从 GPIO 输入
	5:3	RW	GPIO1_SEL<2:0>	选择 GPIO1 的功能： 000: DCLK 001: INT1 010: INT2 011: DOUT 其余选项: NA
	2:0	RW	GPIO0_SEL<2:0>	选择 GPIO1 的功能： 000: DOUT 001: INT1 010: INT2 011: DCLK 其余选项: NA
CTL_REG_5 (0x05)	7:6	RW	TX_DIN_SEL<1:0>	选择发射数据从哪个 GPIO 输入： 00: GPIO3 01: GPIO4 10: NA

寄存器名	位数	R/W	比特名	功能说明
				11: NIRQ
	5:3	RW	GPIO3_SEL<2:0>	选择 GPIO3 的功能: 000: INT2 001: INT1 010: DCLK 011: DOUT 101: DIN 其余选项: NA
	2:0	RW	GPIO2_SEL<2:0>	选择 GPIO2 的功能: 000: INT1 001: INT2 010: DCLK 011: DOUT 其余选项: NA
CTL_REG_6 (0x06)	6	RW	DIG_CLKOUT_EN	将 GPIO4 设置为数字时钟输出: 0: 屏蔽 1: 输出 该功能的优先级高于 GPIO4 的其它配置。
	5:3	RW	GPIO5_SEL<2:0>	选择 GPIO5 的功能: 000: RSTn 001: INT1 010: INT2 011: DOUT 100: DCLK 其余选项: NA
	2:0	RW	GPIO4_SEL<2:0>	选择 GPIO4 的功能: 000: DOUT 001: INT1 010: INT2 011: DCLK 101: DIN 其余选项: NA
CTL_REG_7 (0x07)	7	RW	INT1_TYPE_SEL	选择从 GPIO 输出 INT1 的模式: 0: 模式 1, 输出某一个中断源 1: 模式 2, 输出全部中断源相或的结果
	6	RW	INT2_TYPE_SEL	选择从 GPIO 输出 INT2 的模式: 0: 模式 1, 输出某一个中断源 1: 模式 2, 输出全部中断源相或的结果
	5	RW	LFXO_PAD_EN	将 GPIO2 和 GPIO3 设为 LFXO 的两只管脚: 0: 屏蔽 1: 使能

寄存器名	位数	R/W	比特名	功能说明
				该功能优先级高于 GPIO2 和 GPIO3 的其它配置。
	4	RW	API_STOP	让芯片正在执行的 API 停止。 0: API 进行 1: API 停止
	3	RW	SPI_3W_EN	将芯片的 SPI 接口切换为 3 线模式: 0: 4 线模式 1: 3 线模式
	2:0	RW	NIRQ_SEL<2:0>	选择 NIRQ 的功能: 000: INT1 001: INT2 010: DCLK 011: DOUT 101: DIN 其余选项: NA
CTL_REG_8 (0x08)	6:0	W	API_CMD<6:0>	API 命令输入接口
CTL_REG_9 (0x09)	7	R	API_CMD_FLAG	API 命令标志
	6:0	R	API_RESP<6:0>	API 命令执行后的反馈或应答
CTL_REG_10 (0x0A)	7:0	R	CHIP_MODE_STA<7:0>	芯片当前状态: 00000000: IDLE 10000001: SLEEP 10000010: READY 10000100: RFS 10001000: TFS 10010000: RX 10100000: TX IDLE 状态为芯片接通电源后, 无任何操作的状态。其余值无效。
CTL_REG_11 (0x0B)	7:0	RW	FREQ_DONE_TIMES<7:0>	自动跳频完成次数。 有效范围是 0 到 63
CTL_REG_12 (0x0C)	7:0	RW	FREQ_SPACE<7:0>	自动跳频的频道间隔。
CTL_REG_13 (0x0D)	7:0	RW	FREQ_TIMES<7:0>	自动跳频次数。 有效范围是 1 到 64
CTL_REG_14 (0x0E)	1	RW	RX_HOP_PERSIST	RX 自动跳频设置: 0: 完成设定的次数 1: 一直进行
	0	RW	FREQ_SW_STATE	RX 自动跳频时, 每次进入 RX 状态超时后, 用户可选择退回到某个状态, 再重新回到 RX 状态继续下一个频道的接收:

寄存器名	位数	R/W	比特名	功能说明
				0: 退回 READY 状态 1: 退回 RFS 状态
CTL_REG_16 (0x10)	5:0	RW	INT1_SEL<5:0>	INT1 为模式 1 时, 中断源头的选择: 000000: NA 000001: NA 000010: RSSI_PJD_VALID 000011: PREAM_PASS 000100: SYNC_PASS 000101: ADDR_PASS 000110: CRC_PASS 000111: PKT_DONE 001000: SLEEP_TMO 001001: RX_TMO 001010: RX_FIFO_NMTY 001011: RX_FIFO_TH 001100: RX_FIFO_FULL 001101: RX_FIFO_WBYTE 001110: RX_FIFO_OVF 001111: TX_DONE 010000: TX_FIFO_NMTY 010001: TX_FIFO_TH 010010: TX_FIFO_FULL 010011: STATE_IS_READY 010100: STATE_IS_FS 010101: STATE_IS_RX 010110: STATE_IS_TX 010111: DC_FREE_ERR 011000: LBD_STATUS 011001: NA 011010: API_CMD_FAILED 011011: API_DONE 011100: TX_DC_MAX 011101: ACK_RECV_FAILED 011110: TX_RESEND_MAX 011111: NACK_RECV 100000: SEQ_MATCH 100001: CSMA_MAX 100010: CCA_STATUS
CTL_REG_17 (0x11)	7	RW	INT1_POLAR	中断 1 的极性: 0: 高有效 1: 低有效
	6	RW	INT2_POLAR	中断 2 的极性:

寄存器名	位数	R/W	比特名	功能说明
				0: 高有效 1: 低有效
	5:0	RW	INT2_SEL<5:0>	INT2 为模式 1 时, 中断源头的选择: 000000: NA 000001: NA 000010: RSSI_PJD_VALID 000011: PREAM_PASS 000100: SYNC_PASS 000101: ADDR_PASS 000110: CRC_PASS 000111: PKT_DONE 001000: SLEEP_TMO 001001: RX_TMO 001010: RX_FIFO_NMTY 001011: RX_FIFO_TH 001100: RX_FIFO_FULL 001101: RX_FIFO_WBYTE 001110: RX_FIFO_OVF 001111: TX_DONE 010000: TX_FIFO_NMTY 010001: TX_FIFO_TH 010010: TX_FIFO_FULL 010011: STATE_IS_READY 010100: STATE_IS_FS 010101: STATE_IS_RX 010110: STATE_IS_TX 010111: DC_FREE_ERR 011000: LBD_STATUS 011001: NA 011010: API_CMD_FAILED 011011: API_DONE 011100: TX_DC_MAX 011101: ACK_RECV_FAILED 011110: TX_RESEND_MAX 011111: NACK_RECV 100000: SEQ_MATCH 100001: CSMA_MAX 100010: CCA_STATUS
CTL_REG_18 (0x12)	7	RW	SLEEP_TMO_EN	0: 屏蔽 SLEEP_TMO 中断 1: 使能 SLEEP_TMO 中断
	6	RW	RX_TMO_EN	0: 屏蔽 RX_TMO 中断 1: 使能 RX_TMO 中断

寄存器名	位数	R/W	比特名	功能说明
	5	RW	TX_DONE_EN	0: 屏蔽 TX_DONE 中断 1: 使能 TX_DONE 中断
	4	RW	PREAM_PASS_EN	0: 屏蔽 PREAM_PASS 中断 1: 使能 PREAM_PASS 中断
	3	RW	SYNC_PASS_EN	0: 屏蔽 SYNC_PASS 中断 1: 使能 SYNC_PASS 中断
	2	RW	ADDR_PASS_EN	0: 屏蔽 ADDR_PASS 中断 1: 使能 ADDR_PASS 中断
	1	RW	CRC_PASS_EN	0: 屏蔽 CRC_PASS 中断 1: 使能 CRC_PASS 中断
	0	RW	PKT_DONE_EN	0: 屏蔽 PKT_DONE 中断 1: 使能 PKT_DONE 中断
CTL_REG_19 (0x13)	6	RW	PD_FIFO	0: 在 SLEEP 状态下保存 FIFO 内容 1: 在 SLEEP 状态下不保存 FIFO 内容
	5	RW	FIFO_TH<8>	FIFO_TH 的第 8 位。
	4	RW	FIFO_AUTO_CLR_RX_EN	配置进入 RX 前是否自动清除 RX FIFO 的内容: 0: 不清除 1: 清除
	3	RW	FIFO_AUTO_RES_TX_EN	每次发完一个包自动 restore TX FIFO, 如果每次进入 TX 要重复发送超过 1 个包 (TX_PKT_NUM> 0), 这个比特必须设成 1。
	2	RW	FIFO_TX_TEST_EN	0: TX FIFO 只能用 SPI 写, 1: TX FIFO 可被 SPI 读取。该比特只对 TX FIFO 有效, 除了给用户测试时可以使用, 其余时候都应该设成 0。
	1	RW	FIFO_MERGE_EN	0: 分成 2 个独立的 128-byte 的 FIFO, 1: 合并成 1 个 256-byte 的 FIFO。
	0	RW	FIFO_TX_RX_SEL	当 FIFO 为合并模式时, 0: FIFO 用作 TX FIFO 1: FIFO 用作 RX FIFO
CTL_REG_20 (0x14)	7:0	RW	FIFO_TH<7:0>	FIFO 的填入阈值, 单位是 byte, 对 RX 来说, 当未读数据超过这个阈值时, RX_FIFO_TH_FLG 会置 1; 对 TX 来说, 当未发数据小过这个阈值时, TX_FIFO_TH_FLG 会置 0。 当 FIFO_MERGE_EN = 0 时, 有效范围是 1 到 127; 当 FIFO_MERGE_EN = 1 时, 有效范围是 1 到 255。

寄存器名	位数	R/W	比特名	功能说明
CTL_REG_21 (0x15)	6	RW	RSSI_PJD_VALID_EN	0: 屏蔽 RSSI_PJD_VALID 中断 1: 使能 RSSI_PJD_VALID 中断
	5	RW	OP_CMD_FAILED_EN	0: 屏蔽 API_CMD_FAILED 中断 1: 使能 API_CMD_FAILED 中断
	4	RW	RSSI_COLL_EN	0: 屏蔽 RSSI_COLL 中断 1: 使能 RSSI_COLL 中断
	3	RW	DC_FREE_ERR_EN	0: 屏蔽 DC_FREE_ERR 中断 1: 使能 DC_FREE_ERR 中断
	2	RW	LBD_STATUS_EN	0: 屏蔽 LBD_STATUS 中断 1: 使能 LBD_STATUS 中断
	1	RW	LBD_STOP_EN	0: 屏蔽 LBD_STOP 中断 1: 使能 LBD_STOP 中断
	0	RW	LD_STOP_EN	0: 屏蔽 LD_STOP 中断 1: 使能 LD_STOP 中断
CTL_REG_22 (0x16)	4	RW	TX_DATA_INV	0: GPIO 发射数据输入不取反 1: GPIO 发射数据输入取反
	3	RW	PA_DIFF_SEL	0: PA 单端模式 1: PA 差分模式
	2	RW	TRX_SWT_INV	控制 TX/RX 天线开关的两个输出值: 0: 不取反 1: 取反
	1	RW	TRX_SWT_EN	将 GPIO0 和 GPIO1 设置为 TX/RX 天线开关的控制输出。该功能的优先级比 GPIO0_SEL 和 GPIO1_SEL 要高。
CTL_REG_23 (0x17)	7	RW	API_DONE_EN	0: 屏蔽 API_DONE 中断 1: 使能 API_DONE 中断
	6	RW	CCA_STATUS_EN	0: 屏蔽 CCA_STATUS 中断 1: 使能 CCA_STATUS 中断
	5	RW	CSMA_MAX_EN	0: 屏蔽 CSMA_MAX 中断 1: 使能 CSMA_MAX 中断
	4	RW	TX_DC_MAX_EN	0: 屏蔽 TX_DC_MAX 中断 1: 使能 TX_DC_MAX 中断
	3	RW	ACK_RECV_FAILED_EN	0: 屏蔽 ACK_RECV_FAILED 中断 1: 使能 ACK_RECV_FAILED 中断
	2	RW	TX_RESEND_MAX_EN	0: 屏蔽 TX_RESEND_MAX 中断 1: 使能 TX_RESEND_MAX 中断
	1	RW	NACK_RECV_EN	0: 屏蔽 NACK_RECV 中断 1: 使能 NACK_RECV 中断
	0	RW	SEQ_MATCH_EN	0: 屏蔽 SEQ_MATCH 中断 1: 使能 SEQ_MATCH 中断
CTL_REG_24	5	R	SLEEP_TMO_FLG	SLEEP_TMO 中断标志

寄存器名	位数	R/W	比特名	功能说明
(0x18)	4	R	RX_TMO_FLG	RX_TMO 中断标志
	3	R	TX_DONE_FLG	TX_DONE 中断标志
	2	W	SLEEP_TMO_CLR	SLEEP_TMO 中断清零 0: 无动作 1: 清零
	1	W	RX_TMO_CLR	RX_TMO 中断清零 0: 无动作 1: 清零
	0	W	TX_DONE_CLR	TX_DONE 中断清零 0: 无动作 1: 清零
CTL_REG_25 (0x19)	4	W	PREAM_PASS_CLR	PREAM_PASS 中断清零 0: 无动作 1: 清零
	3	W	SYNC_PASS_CLR	SYNC_PASS 中断清零 0: 无动作 1: 清零
	2	W	ADDR_PASS_CLR	ADDR_PASS 中断清零 0: 无动作 1: 清零
	1	W	CRC_PASS_CLR	CRC_PASS 中断清零 0: 无动作 1: 清零
	0	W	PKT_DONE_CLR	PKT_DONE 中断清零 0: 无动作 1: 清零
CTL_REG_26 (0x1A)	5	R	SYNC1_PASS_FLG	SYNC1_PASS 中断标志
	4	R	PREAM_PASS_FLG	PREAM_PASS 中断标志
	3	R	SYNC_PASS_FLG	SYNC_PASS 中断标志
	2	R	ADDR_PASS_FLG	ADDR_PASS 中断标志
	1	R	CRC_PASS_FLG	CRC_PASS 中断标志
	0	R	PKT_DONE_FLG	PKT_DONE 中断标志
CTL_REG_27 (0x1B)	2	W	TX_FIFO_RESTORE	提供用户手动 restore TX FIFO 功能。restore 的意思是复位读指针，维持写指针不变，这样 TX FIFO 又回到未读状态，可以再次重复发射之前填入的数据。
	1	W	RX_FIFO_CLR	0: 无效，1: 清零 RX FIFO。 用户将这个比特设成 1 之后，无需将它再设回 0，这个比特在内部会自动设回为 0。
	0	W	TX_FIFO_CLR	0: 无效，1: 清零 TX FIFO 用户将这个比特设成 1 之后，无需将它再设回 0，这个比特在内部会自动设回为 0。

寄存器名	位数	R/W	比特名	功能说明
CTL_REG_28 (0x1C)	7	R	RX_FIFO_FULL_FLG	指示 RX FIFO 填满的中断。 0: 无效 1: 有效
	6	R	RX_FIFO_NMTY_FLG	指示 RX FIFO 非空的中断标志位。 0: 无效 1: 有效
	5	R	RX_FIFO_TH_FLG	指示 RX FIFO 未读内容超过 FIFO TH 的中断。 0: 无效 1: 有效
	3	R	RX_FIFO_OVF_FLG	指示 RX FIFO 溢出的中断。 0: 无效 1: 有效
	2	R	TX_FIFO_FULL_FLG	指示 TX FIFO 非空的中断。 0: 无效 1: 有效
	1	R	TX_FIFO_NMTY_FLG	指示 TX FIFO 非空的中断。 0: 无效 1: 有效
	0	R	TX_FIFO_TH_FLG	指示 TX FIFO 未读内容超过 FIFO TH 的中断。 0: 无效 1: 有效
CTL_REG_29 (0x1D)	3	W	OP_CMD_FAILED_CLR	OP_CMD_FAILED 中断清零 0: 无动作 1: 清零
	1	W	DC_FREE_ERR_CLR	DC_FREE_ERR 中断清零 0: 无动作 1: 清零
	0	W	LBD_STATUS_CLR	LBD_STATUS 中断清零 0: 无动作 1: 清零
CTL_REG_30 (0x1E)	3	R	OP_CMD_FAILED_FLG	OP_CMD_FAILED 中断标志
	2	R	RSSI_COLL_FLG	RSSI_COLL 中断标志
	1	R	DC_FREE_ERR_FLG	DC_FREE_ERR 中断标志
	0	R	LBD_STATUS_FLG	LBD_STATUS 中断标志
CTL_REG_31 (0x1F)	7	W	API_DONE_CLR	API_DONE 中断清零 0: 无动作 1: 清零
	6	W	CCA_STATUS_CLR	CCA_STATUS 中断清零 0: 无动作

寄存器名	位数	R/W	比特名	功能说明
				1: 清零
	5	W	CSMA_MAX_CLR	CSMA_MAX 中断清零 0: 无动作 1: 清零
	4	W	TX_DC_MAX_CLR	TX_DC_MAX 中断清零 0: 无动作 1: 清零
	3	W	ACK_RECV_FAILED_CLR	ACK_RECV_FAILED 中断清零 0: 无动作 1: 清零
	2	W	TX_RESEND_MAX_CLR	TX_RESEND_MAX 中断清零 0: 无动作 1: 清零
	1	W	NACK_RECV_CLR	NACK_RECV 中断清零 0: 无动作 1: 清零
	0	W	SEQ_MATCH_CLR	SEQ_MATCH 中断清零 0: 无动作 1: 清零
CTL_REG_32 (0x20)	7	R	API_DONE_FLG	API_DONE 中断标志
	6	R	CCA_STATUS_FLG	CCA_STATUS 中断标志
	5	R	CSMA_MAX_FLG	CSMA_MAX 中断标志
	4	R	TX_DC_MAX_FLG	TX_DC_MAX 中断标志
	3	R	ACK_RECV_FAILED_FLG	ACK_RECV_FAILED 中断标志
	2	R	TX_RESEND_MAX_FLG	TX_RESEND_MAX 中断标志
	1	R	NACK_RECV_FLG	NACK_RECV 中断标志
	0	R	SEQ_MATCH_FLG	SEQ_MATCH 中断标志
CTL_REG_33 (0x21)	1:0	R	PAYLOAD_READY<1:0>	在进行自动应答时, 载荷已经收到的标志位。
CTL_REG_34 (0x22)	7:0	R	RSSI_VALUE<7:0>	RSSI 的读取值, 单位是 dbm。
CTL_REG_35 (0x23)	7:0	R	LBD_DATA<7:0>	低电压检测的值。
CTL_REG_36 (0x24)	7:0	R	TEMP_DATA<7:0>	温度检测的值。
CTL_REG_37 (0x25)	7:0	R	FREQ_CHANL_ACT<7:0>	当自动跳频执行时, 当前使用的频道值。
CTL_REG_38 (0x26)	7:0	R	RX_SEQNUM<7:0>	RX 模式时接收到的包序列号低 8 位。
CTL_REG_39 (0x27)	7:0	R	RX_SEQNUM<15:8>	RX 模式时接收到的包序列号高 8 位。

1.2 PAGE 0, 配置寄存器 (0x28 - 0x77)

这个区域的寄存器，主要用于配置芯片的包格式，FIFO，和系统运行机制。

表 3. PAGE 0 (0x28 - 0x77) 配置寄存器说明

寄存器名	位数	R/W	比特名	功能说明
CTL_REG_40 (0x28)	7:3	RW	RX_PREAM_SIZE<4:0>	RX 模式 Preamble 的长度, 可配置为 0-31 个单位长度, 0 表示不检测 Preamble, 1 表示检测 1 个长度单位的 Preamble, 如此类推。
	2	RW	PREAM_LEN_UNIT	Preamble 的长度单位, TX 和 RX 共用: 0: 单位为 8 bits 1: 单位为 4 bits
	1:0	RW	DATA_MODE<1:0>	选择接收和发射的数据模式: 0: Direct 模式 (默认) 1: NA 2: Packet 模式 3: NA
CTL_REG_41 (0x29)	7:0	RW	TX_PREAM_SIZE<7:0>	TX 模式 Preamble 的长度, 可配置为 0-65535 个单位长度, 0 表示不发送 Preamble, 1 表示发送 1 个长度单位的 Preamble, 如此类推。
CTL_REG_42 (0x2A)	7:0	RW	TX_PREAM_SIZE<15:8>	
CTL_REG_43 (0x2B)	7:0	RW	PREAM_VALUE<7:0>	Preamble 的值, TX 和 RX 共用: 当 PREAM_LEN_UNIT =0 时 8bit 有效, 当 PREAM_LEN_UNIT =1 时只有<3:0>有效
CTL_REG_44 (0x2C)	7	RW	SYNC_MODE_SEL	Sync 的检测模式: 0: 兼容 S2LP 模式 1: 兼容 802.15.4 模式
	6:4	RW	SYNC_TOL<2:0>	RX 模式对 Sync Word 检测的容错比特数: 0:不允许有错 1:允许 1bit 接收错误 2:允许 2bits 接收错误 3:允许 3bits 接收错误 4:允许 4bits 接收错误 5:允许 5bits 接收错误 6:允许 6bits 接收错误 7:允许 7bits 接收错误
	3:1	RW	SYNC_SIZE<2:0>	Sync Word 长度:

寄存器名	位数	R/W	比特名	功能说明
				0: 1 byte 1: 2 bytes 2: 3 bytes 3: 4 bytes 4: 5 bytes 5: 6 bytes 6: 7 bytes 7: 8 bytes
	0	RW	SYNC_MAN_EN	Sync Word 的曼切斯特编解码使能: 0: 不使能 1: 使能
CTL_REG_45 (0x2D)	7:0	RW	SYNC_VALUE<7:0>	Sync Word 的值, 根据不同的 SYNC_SIZE 设置来填入不同的寄存器, 详见下表。
CTL_REG_46 (0x2E)	7:0	RW	SYNC_VALUE<15:8>	
CTL_REG_47 (0x2F)	7:0	RW	SYNC_VALUE<23:16>	
CTL_REG_48 (0x30)	7:0	RW	SYNC_VALUE<31:24>	
CTL_REG_49 (0x31)	7:0	RW	SYNC_VALUE<39:32>	
CTL_REG_50 (0x32)	7:0	RW	SYNC_VALUE<47:40>	
CTL_REG_51 (0x33)	7:0	RW	SYNC_VALUE<55:48>	
CTL_REG_52 (0x34)	7:0	RW	SYNC_VALUE<63:56>	
CTL_REG_53 (0x35)	7:0	RW	SYNC_FEC_VALUE<7:0>	Sync_fec 的值, 根据不同的 SYNC_SIZE 设置来填入不同的寄存器, 详见下表。
CTL_REG_54 (0x36)	7:0	RW	SYNC_FEC_VALUE<15:8>	
CTL_REG_55 (0x37)	7:0	RW	SYNC_FEC_VALUE<23:16>	
CTL_REG_56 (0x38)	7:0	RW	SYNC_FEC_VALUE<31:24>	
CTL_REG_57 (0x39)	7:0	RW	SYNC_FEC_VALUE<39:32>	
CTL_REG_58 (0x3A)	7:0	RW	SYNC_FEC_VALUE<47:40>	
CTL_REG_59 (0x3B)	7:0	RW	SYNC_FEC_VALUE<55:48>	

寄存器名	位数	R/W	比特名	功能说明
CTL_REG_60 (0x3C)	7:0	RW	SYNC_FEC_VALUE<63:56>	
CTL_REG_61 (0x3D)	7:0	RW	PAYLOAD_LENGTH<7:0>	包格式里除 PREAMBLE 和 SYNC 外其他域的长度。payload = length（可选）+ address（可选）+ fcs1（可选）+ fcs2（可选）+ data。data 收发目的地和数据源都是 FIFO。这个配置可用于定长包和变长包。
CTL_REG_62 (0x3E)	7:0	RW	PAYLOAD_LENGTH<15:8>	
CTL_REG_63 (0x3F)	7	RW	INTERLEAVE_EN	交织使能： 0：不使能 1：使能
	5	RW	LENGTH_SIZE	可变包的长度选择： 0：1 字节，支持最大长度 255-byte 的可变包 1：2 字节，支持最大长度 65535-byte 的可变包
	4	RW	PAGGYBACKING_EN	自动应答包是否搭载 payload： 0：不搭载 1：搭载
	2	RW	ADDR_LEN_CONF	在可变包中，Node ID 和 Length Byte 的位置关系 0：Node ID 在 length Byte 之前 1：Node ID 在 length Byte 之后
	1	RW	PAYLOAD_BIT_ORDER	0：先对 payload+CRC 每个 byte MSB 进行编解码 1：先对 payload+CRC 每个 byte LSB 进行编解码
	0	RW	PKT_TYPE	包长类型 0：固定包长 1：可变包长
CTL_REG_64 (0x40)	7	RW	SYNC_VALUE_SEL	SYNC_MODE_SEL 为 0 时有效。 0：选择 SYNC_VALUE 1：选择 SYNC_FEC_VALUE
	6	RW	ADDR_SPLIT_MODE	地址分离模式配置 0：仅 DEST ADDR 域。这时 NODE_ADDR 仅用于 DEST ADDR； 1：DEST ADDR 域+SRC ADDR 域。这时 NODE_ADDR 高 16 比特用于配置 DEST ADDR；低 16 比特用于配置 SRC ADDR。
	5	RW	ADDR_FREE_EN	在 RX 模式下，让 ADDR 检测电路独立出来的使能位。 0：不使能

寄存器名	位数	R/W	比特名	功能说明
				1: 使能
	4	RW	ADDR_ERR_MASK	ADDR 检测错误, 会输出 PKT_ERR 中断, 同时可同步复位解码电路, 该比特控制是否进行同步复位。 0: 允许同步复位 1: 不同步复位
	3:2	RW	ADDR_SIZE<1:0>	ADDR 的长度: 0: 1 byte 1: 2 bytes 2: 3 bytes 3: 4 bytes ADDR_SPLIT_MODE 为 1 时, 表示 DEST ADDR 域 + SRC ADDR 域各占 1~2 bytes
	1:0	RW	ADDR_DET_MODE <1:0>	ADDR 的检测模式: 0: 不检测 1: TX 模式发送 ADDR_VALUE 的内容; RX 模式仅识别 ADDR_VALUE 的内容 2: TX 模式发送 ADDR_VALUE 的内容; RX 模式仅识别 ADDR_VALUE 的内容和全 0 3: TX 模式发送 ADDR_VALUE 的内容; RX 模式仅识别 ADDR_VALUE 的内容, 全 0 和全 1
CTL_REG_65 (0x41)	7:0	RW	ADDR_VALUE<7:0>	ADDR 的值, 可配置 1~4bytes 长度内容
CTL_REG_66 (0x42)	7:0	RW	ADDR_VALUE<15:8>	
CTL_REG_67 (0x43)	7:0	RW	ADDR_VALUE<23:16>	
CTL_REG_68 (0x44)	7:0	RW	ADDR_VALUE<31:24>	
CTL_REG_69 (0x45)	7:0	RW	ADDR_BITMASK<7:0>	地址比特掩码。设置的掩码比特, 对应的 ADDR 不做比较。
CTL_REG_70 (0x46)	7:0	RW	ADDR_BITMASK<7:0>	ADDR_SPLIT_MODE 为 0 时, 仅用于 DEST ADDR 域;
CTL_REG_71 (0x47)	7:0	RW	ADDR_BITMASK<7:0>	ADDR_SPLIT_MODE 为 1 时, 高 16 比特用于配置 DEST ADDR; 低 16 比特用于配置 SRC ADDR。
CTL_REG_72 (0x48)	7:0	RW	ADDR_BITMASK<7:0>	
CTL_REG_73 (0x49)	7:6	RW	CRC_SIZE<1:0>	CRC 域校验码字节数 0: 1 字节

寄存器名	位数	R/W	比特名	功能说明
				1: 2 字节 2: 3 字节 3: 4 字节
	5	RW	CRC_BYTE_SWAP	CRC 的收发顺序: 0: 先收发高字节 1: 先收发低字节
	4	RW	CRC_BIT_INV	CRC 码是否取反: 0: CRC code 不取反 1: CRC code 逐位取反
	3	RW	CRC_RANGE	CRC 的计算范围: 0: 整个 payload 1: 仅为 data
	2	RW	CRC_REFIN	CRC 计算时输入字节的比特顺序反转。
	1	RW	CRC_BIT_ORDER	CRC 收发的高低比特顺序: 0: 先收发高 bit 1: 先收发低 bit
	0	RW	CRC_EN	CRC 使能 0: 不使能 1: 使能
CTL_REG_74 (0x4A)	7:0	RW	CRC_SEED<7:0>	CRC 多项式的初始值
CTL_REG_75 (0x4B)	7:0	RW	CRC_SEED<15:8>	
CTL_REG_76 (0x4C)	7:0	RW	CRC_SEED<23:16>	
CTL_REG_77 (0x4D)	7:0	RW	CRC_SEED<31:24>	
CTL_REG_78 (0x4E)	7:0	RW	CRC_POLY<7:0>	CRC 计算的多项式
CTL_REG_79 (0x4F)	7:0	RW	CRC_POLY <15:8>	
CTL_REG_80 (0x50)	7:0	RW	CRC_POLY <23:16>	
CTL_REG_81 (0x51)	7:0	RW	CRC_POLY <31:24>	
CTL_REG_82 (0x52)	7	RW	CRC_REFOUT	CRC 计算时输出所有字节的比特顺序反转。 0: 从 MSB 到 LSB; 1: 从 LSB 到 MSB。
	6	RW	WHITEN_SEED<8>	WHITEN_SEED 的最高位

寄存器名	位数	R/W	比特名	功能说明
	5	RW	WHITEN_SEED_TYPE	白化编解码多项式为 PN7 时的种子类型： 0: 按 A7139 的方式计算 PN7 seed 1: PN7 seed 为 whiten_seed 定义的值
	4:3	RW	WHITEN_TYPE<1:0>	白化编解码的方式： 0: PN9 CCITT 编解码 1: PN9 IBM 编解码 2: PN7 编解码 3: 无效
	2	RW	WHITEN_EN	白化编解码的使能： 0: 无 whiten 编解码 1: 有 whiten 编解码
	1	RW	MANCH_TYPE	曼切斯特编解码的方式： 0: 01 表示 1; 10 表示 0 1: 10 表示 1; 01 表示 0
	0	RW	MANCH_EN	曼切斯特编解码的使能： 0: 不使能 1: 使能
CTL_REG_83 (0x53)	7:0	RW	WHITEN_SEED<7:0>	白化编解码多项式的种子 PN9 时取全 9bit PN7 时取低 7bit
CTL_REG_84 (0x54)	7	RW	CRCERR_CLR_FIFO_EN	接收 CRC 发生不匹配时： 0: 不清零 RX FIFO 1: 清零 RX FIFO
	6	RW	FCS2_EN	是否包含 FCS2 域 0: 无 FCS2 1: 有 FCS2
	5	RW	SEQNUM_MATCH_EN	TX ack 模式下发射端是否需要将收到的序列号与本地发射出去的序列号作比较 0: 不需要比较 1: 需要比较
	4	RW	SEQNUM_SIZE	SEQNUM 域的大小。 0: 1 字节; 1: 2 字节。
	3	RW	SEQNUM_AUTO_INC	TX 的 SEQNUM 是否自动递增。 0: 不累加 1: 每包自动加 1
	2	RW	SEQNUM_EN	0: 无 SEQNUM 域 (即 FCS1 域) 1: 使能 SEQNUM 域 (即 FCS1 域)
	1:0	RW	TX_PREFIX_TYPE<1:0>	TX Prefix 是指在 Packet 模式下, 进入发射状态后, 由于 FIFO 数据还没有准备好, 但是 PA 已经开始发射了, 就需要定义预发

寄存器名	位数	R/W	比特名	功能说明
				射的内容，可以定义为： 0: 发送 0 1: 发送 1 2: 发送 Preamble 3: NA
CTL_REG_85 (0x55)	7:0	RW	TX_PKT_NUM<7:0>	TX 模式下每次重复发的包个数： 0-255 表示发送 1-256 个包
CTL_REG_86 (0x56)	7:0	RW	TX_PKT_GAP<7:0>	TX 模式下重复发包时，包与包之间的间隔： 0-255 表示包与包之间的发送间隔为 1-256 个 Symbol
CTL_REG_87 (0x57)	7:0	RW	SEQNUM_TX_IN<7:0>	TX 的 SEQNUM 初始化值。
CTL_REG_88 (0x58)	7:0	RW	SEQNUM_TX_IN<15:8>	
CTL_REG_89 (0x59)	7:0	RW	SEQNUM_TX_OUT<7:0>	
CTL_REG_90 (0x5A)	7:0	RW	SEQNUM_TX_OUT<15:8>	当前 TX 的序列号的值。
CTL_REG_91 (0x5B)	7:0	RW	FCS2_TX_IN<7:0>	TX 侧 FCS2 域发送值。
CTL_REG_92 (0x5C)	7:0	RW	FCS2_RX_OUT<7:0>	FCS2 域使能的情况下，将收到的 FCS2 域的值输出到该寄存器
CTL_REG_93 (0x5D)	6:2	RW	FEC_PAD_CODE<12:8>	FEC 的 padding 码配置高 5 位
	1	RW	FEC_RSC_NRNSC_SEL	FEC 选择 RSC 或者 NRNSC。 0: RSC; 1: NRNSC。
	0	RW	FEC_EN	FEC 的使能位： 0: 无 FEC 编解码 1: 有 FEC 编解码
CTL_REG_94 (0x5E)	7:0	RW	FEC_PAD_CODE<7:0>	FEC 的 padding 码配置低 8 位
CTL_REG_95 (0x5F)	7:6	RW	MAP_4FSK_3_LEVEL<1:0>	4FSK 接收模式下，最高电平代表的码值。
	5:4	RW	MAP_4FSK_2_LEVEL<1:0>	4FSK 接收模式下，第二高电平代表的码值。
	3:2	RW	MAP_4FSK_1_LEVEL<1:0>	4FSK 接收模式下，第三高电平代表的码值。
	1:0	RW	MAP_4FSK_0_LEVEL<1:0>	4FSK 接收模式下，第四高电平代表的码值。
CTL_REG_96	6:4	RW	TX_EXIT_STATE<2:0>	完成发射后自动退出到设定的状态： 1: SLEEP

寄存器名	位数	R/W	比特名	功能说明
(0x60)				2: READY 3: TFS 4: TX 5: RFS 6: RX Others: SLEEP 只在 Packet 模式下发射完成后才会自动退出 TX，否则芯片会等待 MCU 发 go_* 命令来切换状态。
	2	RW	TX_ACK_EN	TX 模式下使能 ACK 功能 0: 不使能 1: 使能
	1	RW	TX_DC_PERSIST_EN	TX 模式下，duty cycle 发射的配置 0: 完成 TX_DC_TIMES 配置的次数就退出 1: 一直进行，直到这个比特配置为 0
	0	RW	TX_DC_EN	TX Duty Cycle 的使能 0: 不使能 1: 使能
CTL_REG_97 (0x61)	7	RW	CSMA_EN	CSMA 的使能 0: 不使能 1: 使能
	6:4	RW	RX_EXIT_STATE<2:0>	完成接收后自动退出到设定的状态， 1: SLEEP 2: READY 3: TFS 4: TX 5: RFS 6: RX Others: SLEEP 只在 Packet 模式下才会在接收完成后自动退出 RX，否则芯片会等待 MCU 发 go_* 命令来切换
	3	RW	RX_TIMER_EN	RX TIMER 的使能 0: 不使能 1: 使能
	2	RW	RX_ACK_EN	RX ACK 的使能 0: 不使能 1: 使能
	1	RW	RX_AUTO_HOP_EN	RX AUTO HOP 的使能 0: 不使能 1: 使能

寄存器名	位数	R/W	比特名	功能说明
	0	RW	RX_DC_EN	RX Duty Cycle 的使能 0: 不使能 1: 使能
CTL_REG_98 (0x62)	7	RW	PKT_DONE_EXIT_EN	芯片成功收到 PKT_DONE 信号时是保持当前状态还是立即退出返回到 RX_EXIT_STATE 对应的状态 0: 芯片保持当前状态 1: 芯片根据 RX_EXIT_STATE 配置返回到对应状态
	6:4	RW	RX_HOP_SLP_MODE<2:0>	低功耗自动跳频接收选项配置 一共有 7 种模式可选, 详见表 4。
	3:0	RW	SLP_MODE<3:0>	低功耗接收选项配置 一共有 14 种模式可选, 详见表 5。
CTL_REG_99 (0x63)	7:0	RW	SLEEP_TIMER_M<7:0>	定义了 SLEEP TIMER 的计时时间, 公式如下: $T = M \times 2^{(R+1)} \times 31.25 \text{ us}$ R 的取值范围是 0-26
CTL_REG_100 (0x64)	7:5	RW	SLEEP_TIMER_M<10:8>	
	4:0	RW	SLEEP_TIMER_R<4:0>	
CTL_REG_101 (0x65)	7:0	RW	RX_TIMER_T1_M<7:0>	定义了 RX T1 TIMER 的计时时间, 公式如下: $T = M \times 2^{(R+1)} \times 20 \text{ us}$ R 的取值范围是 0-21
CTL_REG_102 (0x66)	7:5	RW	RX_TIMER_T1_M<10:8>	
	4:0	RW	RX_TIMER_T1_R<4:0>	
CTL_REG_103 (0x67)	7:0	RW	RX_TIMER_T2_M<7:0>	定义了 RX T2 TIMER 的计时时间, 公式如下: $T = M \times 2^{(R+1)} \times 20 \text{ us}$ R 的取值范围是 0-21
CTL_REG_104 (0x68)	7:5	RW	RX_TIMER_T2_M<10:8>	
	4:0	RW	RX_TIMER_T2_R<4:0>	
CTL_REG_105 (0x69)	5:4	RW	TIMER_RAND_MODE<1:0>	配置 CSMA 模式下, CSMA SLEEP TIMER 的随机模式: 00: 随机 R 值 01: 随机 M 值 10: R 值和 M 值都随机 11: 使用配置的固定值
	3	RW	SLEEP_TIMER_EN	SLEEP TIMER 的使能: 0: 不使能 1: 使能
	1	RW	LFCLK_SEL	低频时钟源的选择: 0: LFOSC 1: LFXO
	0	RW	LFCLK_OUT_EN	低频时钟输出到 GPIO4 的使能: 0: 不使能 1: 使能 该功能的优先级低于 DIG_CLKOUT_EN, 高于 GPIO4_SEL。

寄存器名	位数	R/W	比特名	功能说明
CTL_REG_106 (0x6A)	7:5	RW	CSMA_CCA_MODE<2:0>	CSMA 模式下信道忙的判断条件： 000: 认为信道一直空闲 001: 4 个检测窗口有大于等于 1 次检测到 RSSI 有效 010: 4 个检测窗口有大于等于 1 次检测到 PJD 有效 011: 4 个检测窗口有大于等于 1 次检测到 RSSI 有效或 PJD 有效 100: 检测到 1 次 SYNC_PASS 101: 检测到 1 次 SYNC_PASS, 或 4 个检测窗口有大于等于 1 次检测到 RSSI 有效 110: 检测到 1 次 SYNC_PASS, 或 4 个检测窗口有大于等于 1 次检测到 PJD 有效 111: 检测到 1 次 SYNC_PASS, 或 4 个检测窗口有大于等于 1 次检测到 RSS 有效或 PJD 有效
	4	RW	CSMA_PERSIST_EN	CSMA 运行选项： 0: CSMA 如果达到最大检测次数信道仍繁忙则退出 CSMA 模式 1: CSMA 一直检测信道闲忙情况直至信道空闲将数据发射出去
	3:2	RW	RSSI_PJD_SEL<1:0>	RSSI_PJD_VALID 中断产生条件： 00: PJD 有效 01: RSSI 有效 10: PJD 和 RSSI 都有效 11: NA
	1:0	RW	CSMA_CCA_WIN_SEL<1:0>	CSMA 单个检测时间窗口大小： 00: 32-symbol 01: 64-symbol 10: 128-symbol 11: 256-symbol
CTL_REG_107 (0x6B)	7:0	RW	RX_TIMER_CSMA_M<7:0>	定义了 RX CSMA TIMER 的计时时间，公式如下： $T = M \times 2^{(R+1)} \times 20 \text{ us}$ R 的取值范围是 0-21
CTL_REG_108 (0x6C)	7:5	RW	RX_TIMER_CSMA_M<10:8>	
	4:0	RW	RX_TIMER_CSMA_R<4:0>	
CTL_REG_109 (0x6D)	7:0	RW	LBD_TH<7:0>	配置低电压检测阈值，低于该阈值认为是低电压状态。
CTL_REG_110 (0x6E)	7:0	RW	TX_DC_TIMES<7:0>	TX Duty Cycle 模式下，非 Persistent 模式下，规定的最大发射次数。
CTL_REG_112 (0x70)	7:0	R	TX_DC_DONE_TIMES<7:0>	TX Duty Cycle 模式下已经完成的发射次数。

寄存器名	位数	R/W	比特名	功能说明
CTL_REG_113 (0x71)	7:0	RW	TX_RS_TIMES<7:0>	TX ACK 模式下，规定的最大重发次数。
CTL_REG_114 (0x72)	7:0	R	TX_RS_DONE_TIMES<7:0>	TX ACK 模式下已经重发的次数
CTL_REG_115 (0x73)	7:0	RW	CSMA_TIMES<7:0>	CSMA 模式下，非 persistent 模式下，规定的最大检测次数
CTL_REG_116 (0x74)	7:0	RW	CSMA_DONE_TIMES<7:0>	CSMA 模式下已经检测的次数
CTL_REG_118 (0x76)	7:0	RW	SLEEP_TIMER_CSMA_M<7:0>	定义了 SLEEP CSMA TIMER 的计时时间，公式如下： $T = M \times 2^{(R+1)} \times 31.25 \text{ us}$ R 的取值范围是 0-26
CTL_REG_119 (0x77)	7:5	RW	SLEEP_TIMER_CSMA_M<10:8>	
	4:0	RW	SLEEP_TIMER_CSMA_R<4:0>	

表 4. RX_HOP_SLP_MODE<2:0>定义的 7 种低功耗接收选项

编号	RX 的延长方式	RX 的延长条件
0	如果配置成 0，就不做任何延长，T1 计时结束就离开 RX	无
1	T1 内一旦满足检测条件，就离开 T1，将控制权交给 MCU	RSSI_VLD 有效
2		PREAM_OK 有效
3		RSSI_VLD 与 PREAM_OK 同时有效
4	T1 内一旦满足检测条件，就切换到 T2，T2 内一旦检测到 SYNC 就退出 T2 并将控制权交给 MCU，否则 T2 计时结束后就退出 RX	RSSI_VLD 有效
5		PREAM_OK 有效
6		RSSI_VLD 与 PREAM_OK 同时有效

表 5. SLP_MODE<2:0>定义的 14 种低功耗接收选项

编号	RX 的延长方式	RX 的延长条件
0	如果配置为 0，不做任何延长，T1 计时结束就离开 RX	无
1	若 T1 内一旦满足检测条件，则离开 T1，将控制权交给 MCU	RSSI_VLD 有效
2		PREAM_OK 有效
3		RSSI_VLD 与 PREAM_OK 同时有效
4	若 T1 内只要检测到 RSSI 有效，则退出 T1 并一直处于 RX，直到 RSSI 不满足则退出 RX	RSSI_VLD 有效
5	若 T1 内一旦满足检测条件，则切换到 T2，T2 计时结束后则退出 RX	RSSI_VLD 有效
6		PREAM_OK 有效
7		RSSI_VLD 与 PREAM_OK 同时有效
8		PREAM_OK 或 SYNC_OK 任意一个有效
9		PREAM_OK 或 NODE_OK 任意一个有效
10		PREAM_OK 或 SYNC_OK 或 NODE_OK 任意一个有效
11	若 T1 内一旦满足检测条件，切换到 T2，T2 内一旦检测到 SYNC 就退出 T2 并将控制权交给 MCU，否则 T2 计时结束后退出 RX	RSSI_VLD 有效
12		PREAM_OK 有效
13		RSSI_VLD 与 PREAM_OK 同时有效

1.3 PAGE 1, 配置寄存器 (0x00 - 0x68)

表 6. PAGE 1 (0x00 - 0x68) 控制寄存器说明

寄存器名	位数	R/W	比特名	功能说明
CMT 配置区 (0x00 - 0x0F)	7:0	RW	不开放	CMT 专用寄存器, 内容由 RFPDK 导出, 由出厂测试决定其值。
TX 配置区 (0x10 - 0x27)				TX 参数配置寄存器, 内容有 RFPDK 导出, 根据用户配置生成。
RX 配置区 (0x30 - 0x61)				RX 参数配置寄存器, 内容有 RFPDK 导出, 根据用户配置生成。
RX_RSSI_REG_00 (0x62)	3:2	RW	RSSI_UPDATE_SEL	RSSI 值 (单位是 dbm) 更新的条件: 00: 一直实时更新 01: 收到 PREAM_OK 时更新 10: 收到 SYNC_OK 时更新 11: 收到 PKT_DONE 时更新
RX_RSSI_REG_01 (0x63)	7:0	RW	RSSI_ABS_TH<7:0>	RSSI 有效的判定门限, 单位是 dbm
RX_DOUT_REG_00 (0x64)	4:2	RW	DOUT_ADJUST_SEL<2:0>	占空比调整的百分比: 0: 3.33% 1: 6.66% 2: 9.99% 3: 13.32% 4: 16.65% 5: 19.98% 6: 23.21% 7: 26.64%
	1	RW	DOUT_ADJUST_MODE	占空比调整的方向: 0: 增加 1 的占空比 1: 减少 1 的占空比
	0	RW	DOUT_ADJUST_EN	使能调整解调输出占空比: 0: 不使能 1: 使能
RX_ANTD_REG_00 (0x67)	3:2	RW	ANT_WAIT_PMB<1:0>	天线校正时等待 Preamble 的长度: 00: RX_PREAM_SIZE x 1.5 01: RX_PREAM_SIZE x 2 10: RX_PREAM_SIZE x 2.5 11: RX_PREAM_SIZE x 3
	1	RW	ANT_SW_DIS	使能跳过天线切换: 0: 进行天线切换 1: 不进行天线切换
	0	RW	ANT_DIV_EN	天线分集使能: 0: 不使能 1: 使能

2 文档变更记录

表 7. 文档变更记录表

版本号	章节	变更描述	日期
0.5	所有	初始版本发布	2020-09-17

3 联系方式

无锡泽太微电子有限公司深圳分公司

深圳市南山区西丽街道万科云城 3 期 8 栋 A 座 30 楼

邮编: 518055

电话: +86-755-83231427

销售: sales@cmostek.com

技术支持: support@cmostek.com

网址: www.cmostek.com

版权所有 © 无锡泽太微电子有限公司，保留一切权利。

无锡泽太微电子有限公司（以下简称：“CMOSTEK”）保留随时更改、更正、增强、修改 CMOSTEK 产品和/或本文档的权利，恕不另行通知。非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。由于产品版本升级或其他原因，本文档内容会不定期进行更新。CMOSTEK 的产品不建议应用于生命相关的设备和系统，在使用该器件中因为设备或系统运转失灵而导致的损失，CMOSTEK 不承担任何责任。

CMOSTEK 商标和其他 CMOSTEK 商标为无锡泽太微电子有限公司的商标，本文档提及的其他所有商标或注册商标，由各自的所有人拥有。