

MFRC522

非接触式读卡器 IC

广州周立功单片机发展有限公司

地址：广州市天河北路 689 号光大银行大厦 12 楼 F4

网址：<http://www.zlgmcu.com>

技术支持

如果您对文档有所疑问，您可以在办公时间（星期一至星期五上午 8:30~11:50；下午 1:30~5:30；星期六上午 8:30~11:50）拨打技术支持电话或 E-mail 联系。

网 址： www.zlgmcu.com

联系电话： +86 (020) 22644358 22644359 22644360 22644361

E-mail: zlgmcu.support@zlgmcu.com

销售与服务网络

广州周立功单片机发展有限公司

地址：广州市天河区北路 689 号光大银行大厦 12 楼 F4 邮编：510630

电话：(020)38730916 38730917 38730976 38730977

传真：(020)38730925

网址：<http://www.zlgmcu.com>

广州专卖店

地址：广州市天河区新赛格电子城 203-204 室

电话：(020)87578634 87569917

传真：(020)87578842

南京周立功

地址：南京市珠江路 280 号珠江大厦 2006 室

电话：(025)83613221 83613271 83603500

传真：(025)83613271

北京周立功

地址：北京市海淀区知春路 113 号银网中心 712 室
(中发电子市场斜对面)

电话：(010)62536178 62536179 82628073

传真：(010)82614433

重庆周立功

地址：重庆市石桥铺科园一路二号大西洋国际大厦
(赛格电子市场) 1611 室

电话：(023)68796438 68796439

传真：(023)68796439

杭州周立功

地址：杭州市登云路 428 号浙江时代电子市场 205 号

电话：(0571)88009205 88009932 88009933

传真：(0571)88009204

成都周立功

地址：成都市一环路南二段 1 号数码同人港 401 室(磨
子桥立交西北角)

电话：(028) 85439836 85437446

传真：(028)85437896

深圳周立功

地址：深圳市深南中路 2070 号电子科技大厦 A 座
24 楼 2403 室

电话：(0755)83781788 (5 线)

传真：(0755)83793285

武汉周立功

地址：武汉市洪山区广埠屯珞瑜路 158 号 12128 室(华
中电脑数码市场)

电话：(027)87168497 87168297 87168397

传真：(027)87163755

上海周立功

地址：上海市北京东路 668 号科技京城东座 7E 室

电话：(021)53083452 53083453 53083496

传真：(021)53083491

西安办事处

地址：西安市长安北路 54 号太平洋大厦 1201 室

电话：(029)87881296 83063000 87881295

传真：(029)87880865

目录

1.简介.....	1
1.1 范围.....	1
1.2 概述.....	1
1.3 特性.....	1
1.4 简化MFRC522 框图.....	2
2.订购信息.....	2
3.管脚信息.....	2
3.1 封装.....	2
3.2 管脚描述.....	3
4.功能框图.....	4
5.MFRC522 寄存器集.....	5
5.1 MFRC522 寄存器概述.....	5
5.2 寄存器描述.....	7
5.2.1 PAGE0: 命令和状态.....	7
5.2.2 PAGE1: 通信.....	15
5.2.3 PAGE2: 配置.....	22
5.2.4 PAGE3: 测试.....	29
6.MFRC522 的功能.....	35
7. 数字接口.....	37
7.1 自动检测微控制器接口类型.....	37
7.2 兼容SPI的接口.....	37
7.2.1 概述.....	38
7.2.2 读数据.....	38
7.2.3 写数据.....	38
7.2.4 地址字节.....	39
7.3 UART接口.....	39
7.3.1 连接到主机微控制器.....	39
7.3.2 选择传输速率.....	39
7.3.3 帧格式.....	40
7.4 I ² C总线接口.....	42
7.4.1 概述.....	42
7.4.2 数据有效.....	42
7.4.3 起始和停止条件.....	42
7.4.4 字节格式.....	43
7.4.5 应答.....	43
7.4.6 7 位寻址.....	44
7.4.7 寄存器写访问.....	44
7.4.8 寄存器读访问.....	44
7.4.9 高速模式.....	45
7.4.10 高速传输.....	45
7.4.11 高速模式下的串行数据传输格式.....	46
7.4.12 F/S模式和HS模式之间的切换.....	47

7.4.13 低速模式下的MFRC522.....	47
8. 模拟接口和非接触式UART.....	47
8.1 概述.....	47
8.2 Tx驱动器.....	47
8.3 串行数据变换.....	48
8.4 CRC协处理器.....	49
9. FIFO缓冲区.....	49
9.1 概述.....	49
9.2 访问FIFO缓冲区.....	49
9.3 控制FIFO缓冲区.....	49
9.4 FIFO缓冲区的状态信息.....	49
10. 定时器组件.....	50
11. 中断请求系统.....	51
12. 振荡器电路.....	52
13. 节电模式.....	52
13.1 硬掉电.....	52
13.2 软掉电.....	52
13.3 发送器掉电.....	53
14. 复位和振荡器启动时间.....	53
14.1 复位时序要求.....	53
14.2 振荡器启动时间.....	53
15. MFRC522 命令集.....	53
15.1 概述.....	53
15.2 通用特性.....	54
15.3 MFRC522 命令概述.....	54
15.4 MFRC522 命令描述.....	54
15.4.1 IDLE命令.....	54
15.4.2 CALCCRC命令.....	54
15.4.3 TRANSMIT命令.....	55
15.4.4 NOCMDCHANGE命令.....	55
15.4.5 RECEIVE命令.....	55
15.4.6 TRANSCEIVE命令.....	55
15.4.7 MFAUTHENT命令.....	55
15.4.8 SOFTRESET命令.....	56
16. 测试信号.....	56
16.1 测试总线.....	56
16.2 管脚AUX的测试信号.....	57
16.3 PRBS.....	57
17. 典型应用.....	57
18. 电气特性.....	58
18.1 绝对最大额定值.....	58
18.2 极限值.....	58
18.3 ESD特性.....	59
18.4 温度特性.....	59

18.5 工作条件范围.....	59
18.6 输入/输出管脚特性.....	59
18.6.1 EA, 12C, SIGIN和NRESET输入管脚特性.....	59
18.6.2 D1, D2, D3, D4, D5, D6 和D7 输入/输出管脚特性.....	60
18.6.3 SDA输入/输出管脚特性.....	60
18.6.4 SIGOUT输出管脚特性.....	60
18.6.5 IRQ输出管脚特性.....	60
18.6.6 Rx输入管脚特性.....	61
18.6.7 OSCIN输入管脚特性.....	61
18.6.8 AUX1 和AUX2 输出管脚特性.....	61
18.6.9 TX1 和TX2 输出管脚特性.....	61
18.7 电流消耗.....	62
18.8 RX输入电压范围.....	62
18.9 RX输入灵敏度.....	63
18.10 时钟频率.....	63
18.11 XTAL振荡器.....	64
18.12 典型 27.12MHz晶体的要求.....	64
18.13 SPI兼容接口的时序.....	64
18.14 I ² C时序.....	65
19.封装.....	66
附录A 版本信息.....	67

1. 简介

1.1 范围

本文描述了发送模块 MFRC522 的功能以及功能和电气规范。本文是最早的目标版本，有关模块的所有规范请参考最终版本。

1.2 概述

MFRC522 是高度集成的非接触式（13.56MHz）读写卡芯片。此发送模块利用调制和解调的原理，并将它们完全集成到各种非接触式通信方法和协议中（13.56MHz）。

MFRC522 发送模块支持下面的工作模式：

- 读写器，支持 ISO 14443A / MIFARE®

MFRC522 的内部发送器部分可驱动读写器天线与 ISO 14443A/MIFARE® 卡和应答机的通信，无需其它的电路。接收器部分提供一个功能强大和高效的解调和译码电路，用来处理兼容 ISO 14443A/MIFARE® 的卡和应答机的信号。数字电路部分处理完整的 ISO 14443A 帧和错误检测（奇偶&CRC）。MFRC522 支持 MIFARE® Classic（如，MIFARE® 标准）器件。MFRC522 支持 MIFARE® 更高速的非接触式通信，双向数据传输速率高达 424kbit/s。

可实现各种不同主机接口的功能：

- SPI 接口
- 串行 UART（类似 RS232，电压电平取决于提供的管脚电压）
- I2C 接口

1.3 特性

- 高度集成的模拟电路，解调和译码响应。
- 缓冲的输出驱动器与天线的连接使用最少的外部元件。
- 支持 ISO 14443A/MIFARE®。
- 读写器模式中，与 ISO 14443A/MIFARE® 的通信距离高达 50mm，取决于天线的长度和调谐。
- 读写器模式下支持 MIFARE® Classic 加密
- 支持 ISO 14443 212kbit/s 和 424kbit/s 的更高传输速率的通信。
- 支持的主机接口
 - 10Mbit/s 的 SPI 接口
 - I2C 接口，快速模式的速率为 400kbit/s，高速模式的速率为 3400kbit/s
 - 串行 UART，传输速率高达 1228.8kbit/s，帧取决于 RS232 接口，电压电平取决于提供的管脚电压
- 64 字节的发送和接收 FIFO 缓冲区。
- 灵活的中断模式。
- 低功耗的硬复位功能。

- 软件掉电模式。
- 可编程定时器。
- 内部振荡器，连接 27.12MHz 的晶体。
- 3.3V 的电源电压。
- CRC 协处理器
- 自由编程的 I/O 管脚
- 内部自测试

1.4 简化 MFRC522 框图

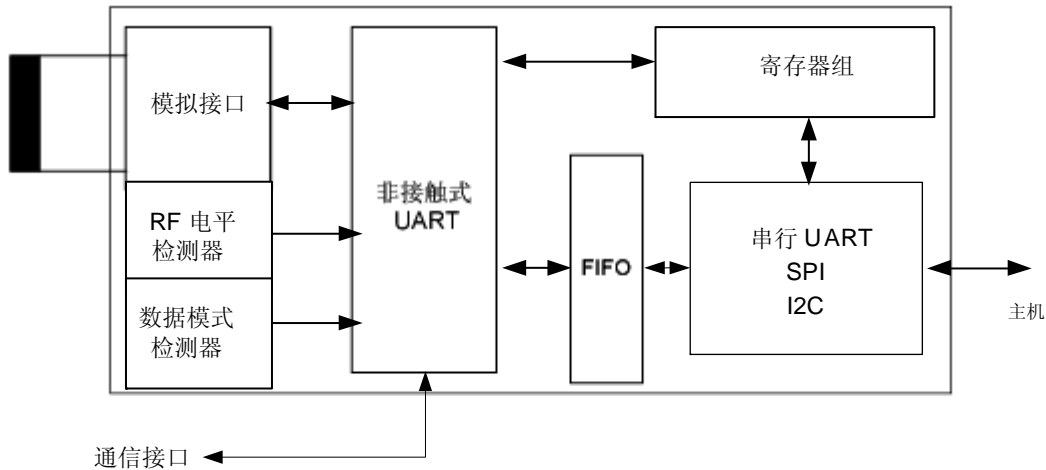


图 1 简化的 MFRC522 框图

模拟接口用来处理模拟信号的调制和解调。

非接触式 UART 用来处理与主机通信时的协议要求。FIFO 缓冲区快速而方便地实现了主机和非接触式 UART 之间的数据传输。

不同的主机接口功能可满足不同用户的要求。

2. 订购信息

待定。

3. 管脚信息

3.1 封装

MFRC522 包含 HVQFN32 的封装形式。

表 1 封装信息

封装	功能注释
HVQFN32	见第 19 章的表面封装

3.2 管脚描述

表 2 管脚描述

注：管脚类型： I—输入； O—输出； PWR—电源

符号	HVQFN32	类型	描述
OSCIN	21	I	晶振输入 ：振荡器的反相放大器的输入。它也是外部产生的时钟的输入（fosc=27.12MHz）。
IRQ	23	O	中断请求 ：输出，用来指示一个中断事件。
SIGIN	7	I	信号输入
SIGOUT	8	O	信号输出
TX1	11	O	发送器 1 ：传递调制的 13.56MHz 的能量载波信号。
TVDD	12	PWR	发送器电源 ：给 TX1 和 TX2 的输出级供电。
TX2	13	O	发送器 2 ：传递调制的 13.56MHz 的能量载波信号。
TVSS	10,14	PWR	发送器地 ：TX1 和 TX2 的输出级的地。
DVSS	4	PWR	数字地 不同接口地数据管脚（测试端口、I ² C、SPI、UART）
D1	25	I/O	
D2	26	I/O	
D3	27	I/O	
D4	28	I/O	
D5	29	I/O	
D6	30	I/O	
D7	31	I/O	
SDA	24	I	串行数据线
EA	32	I	外部地址 ：该管脚用来编码 I ² C 地址。
I2C	1	I	I²C 使能
DVDD	3	PWR	数字电源
AVDD	3	PWR	模拟电源
AUX1	19	O	辅助输出 ：这两个管脚用于测试。
AUX2	20	O	
AVSS	18	PWR	模拟地
RX	17	I	接收器输入 ：接收的 RF 信号管脚
VMID	16	PWR	内部参考电压 ：该管脚提供内部参考电压。
NRSTPD	6	I	不复位和掉电 ：管脚为低电平时，切断内部电流吸收，关闭振荡器，断开输入管脚与外部电路的连接。管脚的上升沿来启动内部复位阶段。
OSCOU	22	O	晶振输出 ：振荡器的反相放大器的输出。
TESTPIN	9		不连接：三态管脚
PVDD	2	PWR	管脚电源
PVSS	5	PWR	管脚电源地

指定封装接口的管脚功能详见第 6 章。

4.功能框图

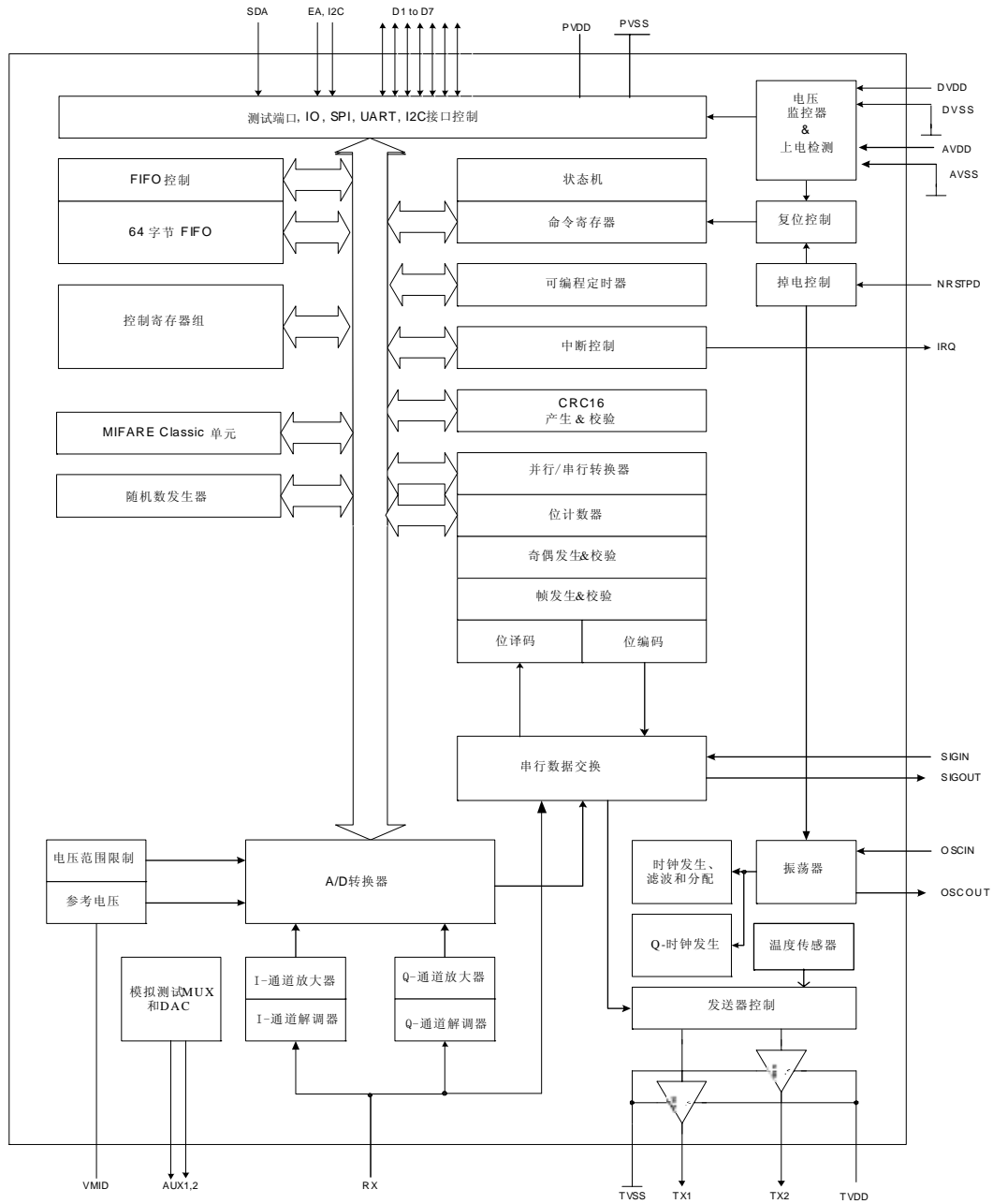


图 2 MFRC522 功能框图

5.MFRC522 寄存器集

5.1 MFRC522 寄存器概述

表 3 MFRC522 寄存器概述

地址 (HEX)	寄存器名称	功能
PAGE0: 命令和状态		
0	RFU	保留为将来之用。
1	CommandReg	启动和停止命令的执行。
2	ComIEnReg	中断请求传递的使能和禁能控制位。
3	DivIEnReg	中断请求传递的使能和禁能控制位
4	ComIrqReg	包含中断请求标志
5	DivIrqReg	包含中断请求标志
6	ErrorReg	错误标志, 指示执行的上个命令的错误状态
7	Status1Reg	包含通信的状态标志
8	Status2Reg	包含接收器和发送器的状态标志
9	FIFODtataReg	64 字节 FIFO 缓冲区的输入和输出
A	FIFOLevelReg	指示 FIFO 中存储的字节数
B	WaterLevelReg	定义 FIFO 下溢和上溢报警的 FIFO 深度
C	ControlReg	不同的控制寄存器
D	BitFramingReg	面向位的帧的调节
E	CollReg	RF 接口上检测到的第一个位冲突的位的位置
F	RFU	保留为将来之用。
PAGE1: 命令		
0	RFU	保留为将来之用
1	ModeReg	定义发送和接收的常用模式
2	TxModeReg	定义发送过程的数据传输速率
3	RxModeReg	定义接收过程中的数据传输速率
4	TxControlReg	控制天线驱动器管脚 TX1 和 TX2 的逻辑特性
5	TxAutoReg	控制天线驱动器的设置
6	TxSeIReg	选择天线驱动器的内部源
7	RxSeIReg	选择内部的接收器设置
8	RxThresholdReg	选择位译码器的阈值
9	DemodReg	定义解调器的设置
A	RFU	保留为将来之用
B	RFU	保留为将来之用
C	MifareReg	控制 ISO 14443/MIFARE 模式中 106kbit/s 的通信
D	RFU	保留为将来之用
E	RFU	保留为将来之用
F	SerialSpeedReq	选择串行 UART 接口的速率

续表 3

地址 (HEX)	寄存器名称	功能
PAGE2: CFG		
0	RFU	保留为将来之用
1	CRCResultReg	显示 CRC 计算的实际 MSB 和 LSB 值
2		
3	RFU	保留为将来之用
4	ModWidthReg	控制 ModWidth 的设置
5	RFU	保留为将来之用
6	RFCfgReg	配置接收器增益
7	GsNReg	选择天线驱动器管脚 TX1 和 TX2 的调制电导
8	CWGsCfgReg	选择天线驱动器管脚 TX1 和 TX2 的调制电导
9	ModGsCfgReg	选择天线驱动器管脚 TX1 和 TX2 的调制电导
A	TModeReg	定义内部定时器的设置
B	TPrescalerReg	
C	TReloadReg	描述 16 位长的定时器重装值
D		
E	TcounterValueReg	显示 16 位长的实际定时器值
F		
PAGE3: 测试		
0	RFU	保留为将来之用
1	TestSel1Reg	常用测试信号的配置
2	TestSel2Reg	常用测试信号的配置和 PRBS 控制
3	TestPinEnReg	D1-D7 输出驱动器的使能管脚（注：仅用于串行接口）
4	TestPin ValueReg	定义 D1-D7 用作 I/O 总线时的值
5	TestBusReg	显示内部测试总线的状态
6	AutoTestReg	控制数字自测试
7	VersionReg	显示版本
8	AnalogTestReg	控制管脚 AUX1 和 AUX2
9	TestDAC1Reg	定义 TestDAC1 的测试值
A	TestDAC2Reg	定义 TestDAC2 的测试值
B	TestADCReg	显示 ADC I 和 Q 通道的实际值
C-F	RFT	保留用于产品测试

不同寄存器的位和标志的操作不同，由它们的功能决定。通常，操作相同的位被分在同一组寄存器。

表 4 寄存器位的操作及其名称

缩写	操作	描述
r/w	读和写	这些位由微控制器写入和读出。由于它们只用作控制用途，它们的内容不受内部状态机的影响，例如，CommIEEn 寄存器可由微控制器写入和读出，也可由内部状态机读出，但状态机不能改变它们的内容。
dy	动态	这些位由微控制器写入和读出。然而，它们也可由内部状态机自动写入，例如，当执行完一个实际的命令后，命令寄存器的内容随之自动更改。
r	只读	这些寄存器保存着大量的标志，其值仅由内部状态来决定，例如，CRCReady 标志不是从外部写入，而是显示了一个内部的状态。
w	只写	读这些只写寄存器时通常返回零。
RFU	-	这些寄存器保留为将来之用，它们的值不应被更改。
RFT	-	这些寄存器保留用于产品测试，它们的值不应被更改。

5.2 寄存器描述

5.2.1 PAGE0: 命令和状态

5.2.1.1 RFU 寄存器

保留为将来之用的寄存器。

表 5 RFUReg

RFUReg	地址: 0x00		复位值: 00000000 (0x00)					
	7	6	5	4	3	2	1	0
	00000000							
访问权限	RFU							

表 6 RFUReg 的位描述

位	符号	功能
7-0	00000000	RFU

5.2.1.2 CommandReg

启动和停止命令的执行。

表 7 CommandReg

CommandReg	地址: 0x01		复位值: 00100000 (0x20)					
	7	6	5	4	3	2	1	0
	00	RcvOff	Power	Command				
			Down					
访问权限	RFU	r/w	dy	dy	dy	dy	dy	dy

表 8 CommandReg 的位描述

位	符号	功能
7-6	00	RFU
5	RcvOff	该位置位时，接收器的模拟电路部分被关断。
4	PowerDown	该位置位时，进入软 PowerDown 模式。 该位清零时，MFRC522 启动唤醒过程。在唤醒过程中该位始终为 1。0 表示 MFRC522 已准备好，可以开始操作。 请参考 13.2 节。 注：如果命令 SoftReset 已经被激活，PowerDown 位就不能再置位。
3-0	Command	根据命令代码来激活命令。通过读该寄存器来了解实际正在执行的命令。见 15.3 节。

5.2.1.3 CommIEnReg

中断请求传递的使能和禁能控制位。

表 9 CommIEnReg

CommIEnReg 地址：0x02 复位值：10000000 (0x80)

	7	6	5	4	3	2	1	0
	IRqInv	TxIEn	RxIEn	IdleIEn	HiAlertIEn	LoAlertIEn	ErrIEn	TimerIEn
访问权限	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

表 10 CommIEnReg 的位描述

位	符号	功能
7	IRqInv	该位置位时，管脚 IRQ 上的信号与寄存器 Status1Reg 的 IRq 位的值相反。 该位清零时，管脚 IRQ 上的信号与 IRq 位的值相同。该位与 DivIEnReg 寄存器的 IrqPushPull 位一起使用，如果是默认值 1，则管脚 IRQ 的输出电平为三态。
6	TxIEn	允许发送器中断请求（由位 TxIRq 来指示）传递到 IRQ 管脚。
5	RxIEn	允许接收器中断请求（由位 RxIRq 来指示）传递到 IRQ 管脚。
4	IdleIEn	允许空闲中断请求（由位 IdleIRq 来指示）传递到 IRQ 管脚。
3	HiAlertIEn	允许高报警中断请求（由位 HiAlertIRq）传递到 IRQ 管脚。
2	LoAlertIEn	允许低报警中断请求（由位 LoAlertIRq）传递到 IRQ 管脚。
1	ErrIEn	允许错误中断请求（由位 ErrIRq）传递到 IRQ 管脚。
0	TimerIEn	允许定时器中断请求（由位 TimerIRq）传递到 IRQ 管脚。

5.2.1.4 DivIEnReg

中断请求传递的使能和禁能控制位。

表 11 DivIEnReg

DivIEnReg 地址：0x03 复位值：00000000 (0x00)

	7	6	5	4	3	2	1	0
	IRQPushPull	00	SignActIEn	0	CRCIEn	00		
访问权限	r/w	RFU	r/w	RFU	r/w	RFU		

表 12 DivlEnReg 的位描述

位	符号	功能
7	IRQPushPull	该位置位时，IRQ 用作标准 CMOS 输出管脚。 该位清零时，IRQ 用作开漏输出管脚。
6-5	00	RFU
4	SiginActIE	允许 SIGIN 有效中断请求传递到 IRQ 管脚。
3	0	RFU
2	CRCIE	允许 CRC 中断请求（由 CRCIRq 来指示）传递到 IRQ 管脚。
1-0	00	RFU

5.2.1.5 CommIRqReg

包含中断请求标志。

表 13 CommIRqReg

CommIRqReg 地址：0x04 复位值：0000100 (0x04)

	7	6	5	4	3	2	1	0
	Set1	TxIRq	RxIRq	IdleIRq	HiAlertIRq	LoAlertIRq	ErrIRq	TimerIRq
访问权限	w	dy	dy	dy	dy	dy	dy	dy

表 14 CommIRQReg 的位描述

位	符号	功能
7	Set1	该位置位时，Set1 定义 CommIRqReg 寄存器中的屏蔽位置位。 该位清零时，Set1 定义 CommIRqReg 寄存器中的屏蔽位清零。
6	TxIRq	该位在发送数据的最后一位发送出去后立刻置位。
5	RxIRq	当接收器检测到一个有效数据流结束后，该位置位。如果寄存器 RxModeReg 的 RxNoErr 位置位，当 FIFO 中有可用的数据字节时，RxIRq 位只能置位。
4	IdleIRq	当一个命令自身终止（例如，当 CommandReg 的值从其它命令变为空闲命令时）该位置位。 如果启动一个未知命令，CommandReg 的值变为空闲命令，并且 IdleIRq 置位。由微控制器启动的空闲命令不会置位 IdleIRq。
3	HiAlertIRq	该位在 Status1Reg 寄存器的 HiAlert 位置位时置位。与 HiAlert 相反，HiAlertIRq 将保存此中断事件，直到得到 Set1 的清零指示时才能被复位。
2	LoAlertIRq	该位在 Status1Reg 寄存器的 LoAlert 位置位时置位。与 LoAlert 相反，LoAlertIRq 将保存此中断事件，直到得到 Set1 的清零指示时才能被复位。
1	ErrIRq	只要 Error 寄存器中任何一个错误标志被设置，该位就置位。
0	TimerIRq	当定时器的 TimerValue 寄存器的值递减到零时，该位置位。

注：

1. CommIRqReg 寄存器的每一位都可通过软件来清零。

5.2.1.6 DivIRqReg

包含中断请求标志。

表 15 DivIRqReg

DivIRqReg	地址: 0x05		复位值: 000x0000 (0Xx0)					
	7	6	5	4	3	2	1	0
	Set2	00		SignActIRq	0	CRCIRq	00	
访问权限	w	RFU		dy	RFU	dy	RFU	

表 16 DivIRqReg 的位描述

位	符号	功能
7	Set2	该位置位时, Set2 定义 DivIRqReg 寄存器中的屏蔽位置位。该位清零时, Set2 定义 DivIRqReg 寄存器中的屏蔽位清零。
6-5	00	RFU
4	SignActIRq	SIGIN 有效时该位置位。当检测到一个上升或下降信号沿时中断被设置。
3	0	RFU
2	CRCIRq	当 CRC 命令有效且所有数据被处理时该位置位。
1-0	00	RFU

注:

1. DivIRqReg 寄存器中的每一位都可通过软件来清零。

5.2.1.7 ErrorReg

错误标志寄存器, 显示执行的上个命令的错误状态。

表 17 ErrorReg

ErrorReg	地址: 0x06		复位值: 00000000 (0x00)					
	7	6	5	4	3	2	1	0
	WrErr	TempErr	0	BufferOfI	CollErr	CRCErr	ParityErr	ProtocolErr
访问权限	r	r	RFU	r	r	r	r	r

表 18 ErrorReg 的位描述

位	符号	功能
7	WrErr	当在 AutoColl 命令或 MFAuthent 命令执行过程中微控制器向 FIFO 写入数据或在发送 RF 接口的最后一位数据和接收 RF 接口的最后一位数据之间的时间内微控制器向 FIFO 写入数据时, 该位置位。
6	TempErr	如果内部温度传感器检测到过热, 则该位置位。这时天线驱动器自动关断。
5	0	RFU
4	BufferOvfI	如果 FIFO 缓冲区已满时微控制器或 MFRC522 的内部状态机 (如接收器) 还向 FIFO 写入数据, 则该位置位。
3	CollErr	该位在检测到位冲突时置位。它在接收器启动阶段自动清除。此标志只在 106kbit/s 的位良好的防冲突机制中有效。在 212 和 424kbit/s 的通信机制下此标志始终为 0。

续表 18

位	符号	功能
2	CRCErr	如果 RxModeReg 寄存器的 RxCRCEn 置位且 CRC 计算出错，则该位置位。此标志在接收器启动阶段自动清零。
1	ParityErr	如果奇偶校验出错，则该位置位。此标志在接收器启动阶段自动清零。它只在 106kbit/s 的 ISO 14443A/MIFARE 通信中有效。
0	ProtocolErr	只要出现以下一种情况，该位就置位： a.) 如果 SOF 出错，则该位置位。它在接收器启动阶段自动清零。此标志只在 106kbit/s 的通信中有效。 b.) 在 MFAuthent 命令执行过程中，如果数据流接收到的字节数错误，则 ProtocolErr 置位。

注：

1. 执行命令时除 TempErr 之外的所有错误标志都将清除。这些标志不能通过软件置位。

5.2.1.8 Status1Reg

包含 CRC、中断和 FIFO 缓冲区的状态标志。

表 19 Status1Reg

Status1Reg	地址：0x07			复位值：00100001 (0x21)				
	7	6	5	4	3	2	1	0
	0	CRCOk	CRCReady	IRq	TRunning	0	HiAlert	LoAlert
访问权限	RFU	r	r	r	r	RFU	r	r

表 20 Status1Reg 的位描述

位	符号	功能
7	0	RFU
6	CRCOk	CRC 结果为 0 时该位置位。发送和接收数据时不定义 CRCOk(使用 ErrorReg 寄存器的 CRCErr) CRCOk 用来指示 CRC 协处理器的状态，在计算过程中其值变为 0；当计算正确执行后，其值变为 1。
5	CRCReady	CRC 计算完成后该位置位。此标志只在 CRC 协处理器计算时才有效。
4	IRq	该位用来显示请求中断的中断源（有关中断使能标志的设置，见寄存器 CommIEnReg 和 DivIEnReg）。
3	TRunning	MFRC522 的定时器单元运行时，该位置位。定时器使 TcounterValReg 的值跟随下个定时器时钟的到来而递减。 注：在门控模式中，如果定时器通过寄存器位使能，则 TRunning 位置位。该位不受门控信号的影响。
2	0	RFU
1	HiAlert	当 FIFO 缓冲区中保存的字节数满足下面的等式时，该位置位。 等式：HiAlert = (64 - FIFOLength) ≤ WaterLevel 举例：FIFOLength = 60, WaterLevel = 4 → HiAlert = 1 FIFOLength = 59, WaterLevel = 4 → HiAlert = 0

续表 20

位	符号	功能
0	LoAlert	当 FIFO 缓冲区中保存的字节数满足下面的等式时，该位置位。 等式: $LoAlert = FIFOLength \leq WaterLevel$ 举例: $FIFOLength = 4, WaterLevel = 4 \rightarrow LoAlert = 1$ $FIFOLength = 5, WaterLevel = 4 \rightarrow LoAlert = 0$

5.2.1.9 Status2Reg

包含接收器、发送器和数据模式检测器的状态标志。

表 21 Status2Reg

Status2Reg	地址: 0x08		复位值: 00000000 (0x00)					
	7	6	5	4	3	2	1	0
	TempSensOff	I2CforceHS	00	MFCrypto1On	Modem State			
访问权限	r/w	r/w	RFU	dy	r	r	r	

表 22 Status2Reg 的位描述

位	符号	功能
7	TempSensOff	内部温度传感器关断时该位置位。
6	I2CforceHS	I2C 输入滤波器设置。 I2C 输入滤波器设置成与 I2C 协议无关的高速模式时该位置位。 I2C 输入滤波器设置成使用的 I2C 协议时该位清零。
5-4	00	RFU
3	MFCrypto1On	该位用来指示 MIFARE®Cypto1 单元接通和因此所有卡的数据通信被加密的情况。 只有成功执行 MFAuthent 命令后，该位才能置位。 该位只在 MIFARE®标准卡的读写器模式中有效。 该位通过软件来清零。
2-0	ModemState	ModemState 用来显示发送器和接收器状态机的状态。 状态描述 000 IDLE 001 等待寄存器 BitFramingReg 的 StartSend 的设置 010 TxWait 011 发送数据 100 RxWait 101 等待数据 110 接收数据

5.2.1.10 FIFODataReg

64 字节 FIFO 缓冲区的输入和输出。

表 23 FIFODataReg

FIFODataReg	地址: 0x09		复位值: xxxxxxxx (0xXX)					
	7	6	5	4	3	2	1	0
	FIFOData							
访问权限	dy	dy	dy	dy	dy	dy	dy	dy

表 24 FIFODataReg 的位描述

位	符号	功能
7-0	FIFOData	内部 64 字节 FIFO 缓冲区的数据输入和输出口。FIFO 缓冲区用作所有数据流输入和输出的并行输入/并行输出转换器。

5.2.1.11 FIFOLevelReg

指示 FIFO 中保存的字节数。

表 25 FIFOLevelReg

FIFOLevelReg	地址: 0x0A		复位值: 00000000 (0x00)					
	7	6	5	4	3	2	1	0
	FlushBuffer	FIFOLevel						
访问权限	w	r	r	r	r	r	r	r

表 26 FIFOLevelReg 的位描述

位	符号	功能
7	FlushBuffer	该位置位时，内部 FIFO 缓冲区的读和写指针以及寄存器 ErrReg 的 BufferOvfl 标志立刻被清除。 该位被读出时返回 0。
6-0	FIFOLevel	指示 FIFO 缓冲区中保存的字节数。写 FIFODataReg 时该字段的值递增；读 FIFODataReg 时该字段的值递减。

5.2.1.12 WaterLevelReg

定义 FIFO 下溢和上溢报警的 FIFO 深度。

表 27 WaterLevelReg

WaterLevelReg	地址: 0x0B		复位值: 00001000 (0x08)					
	7	6	5	4	3	2	1	0
	00	WaterLevel						
访问权限	RFU	r/w	r/w	r/w	r/w	r/w	r/w	r/w

表 28 WaterLevelReg 的位描述

位	符号	功能
7-6	00	RFU
5-0	WaterLevel	该寄存器定义了用于微控制器的 MFRC522 的 FIFO 缓冲区上溢或下溢的报警深度。 如果 FIFO 缓冲区剩余空间可保存的字节数等于或小于定义的 WaterLevel 字节时，Status1Reg 寄存器的 HiAlert 位置位。 如果 FIFO 缓冲区的长度等于或小于定义的 WaterLevel 字节时，Status1Reg 寄存器的 LoAlert 位置位。

5.2.1.13 ControlReg

包含不同的控制位。

表 29 ControlReg

ControlReg	地址: 0x0C				复位值: 00010001 (0x20)			
	7	6	5	4	3	2	1	0
	TstopNow	TstartNow	010		RxLastBits			
访问权限	w	w	RFU		r	r	r	r

表 30 ControlReg 的位描述

位	符号	功能
7	TStopNow	该位置位时，定时器立刻停止运行。该位读出时返回 0。
6	TStartNow	该位置位时，定时器立刻开始运行。该位读出时返回 0。
5-3	010	RFU
2-0	RxLastBits	显示最后接收到的字节的有效位的数目。如果该位为 0，则整个字节有效。

5.2.1.14 BitFramingReg

面向位的帧的调节。

表 31 BitFramingReg

BitFramingReg	地址: 0x0D				复位值: 00000000 (0x00)			
	7	6	5	4	3	2	1	0
	StartSend	RxAlign		0	TxLastBits			
访问权限	w	dy	dy	dy	RFU	dy	dy	dy

表 32 BitFraming 寄存器的位描述

位	符号	功能
7	StartSend	该位置位时启动数据的发送。 该位只在与收发命令一起使用时才有效。
6-4	RxAlign	用于面向位的帧的接收: RxAlign 定义第一个接收到的位在 FIFO 中的存放位置。接收到的其它位依次存放在后面的位置。 例如: RxAlign=0: 接收到的 LSB 位存放在位 0, 接收到的第 2 位存放在位 1。 RxAlign=1: 接收到的 LSB 位存放在位 1, 接收到的第 2 位存放在位 2。 RxAlign=7: 接收到的 LSB 位存放在位 7, 接收到的第 2 位存放在下一个字节的位 0 的位置上。 此标志只在 106kbit/s 的位良好的防冲突机制中有效。其它模式中应设置为 0。
3	0	RFU
2-0	TxLastBits	用于面向位的帧的发送: TxLastBits 定义发送的最后一个字节的位数。000 表示最后一个字节的所有位都应发送。

5.2.1.15 CollReg

定义在 RF 接口上检测到的第一个位冲突。

表 33 CollReg

CollReg	地址: 0x0E			复位值: 101xxxxx (0xXX)				
	7	6	5	4	3	2	1	0
Values AfterColl	0	CollPos	CollPos					
访问权限	r/w	RFU	r	r	r	r	r	r

表 34 CollReg 寄存器的位描述

位	符号	功能
7	ValuesAfterColl	如果该位设置成 0, 则所有接收的位在冲突后将被清除。 该位只在 106kbit/s 的位良好的防冲突过程中使用, 其它情况下该位应当设置成 1。
6	0	RFU
5	CollPosNotValid	如果未检测到冲突或产生冲突的位置超出 CollPos 的范围, 则该位置位。
4-0	CollPos	这些位显示了在接收帧中检测到的第一个冲突的位的位置, 它们只能显示数据位的位置 例如: 0x00 表明位冲突发生在起始位 0x01 表明位冲突发生在第 1 位 0x08 表明位冲突发生在第 8 位 只有位 CollPosNotValid 清零时上面的这些位才能被识别。

5.2.2 PAGE1: 通信

5.2.2.1 RFU 寄存器

保留为将来之用的寄存器。

表 35 RFUReg

RFUReg	地址: 0x10		复位值: 00000000 (0x00)					
	7	6	5	4	3	2	1	0
	00000000							
访问权限	RFU							

表 36 RFUReg 的位描述

位	符号	功能
7-0	00000000	RFU

5.2.2.2 ModeReg

定义发送和接收的常用模式。

表 37 ModeReg

ModeReg	地址: 0x11		复位值: 00111111 (0x3F)					
	7	6	5	4	3	2	1	0
	00	TxWaitRF	1	PolSign	1	CRCPreset		
访问权限	RFU	r/w	RFU	r/w	RFU	r/w	r/w	r/w

表 38 ModeReg 的位描述

位	符号	功能										
7-6	00	RFU										
5	TxWaitRF	如果 RF 场 (RF field) 产生, 则 TxWaitRF 置位, 发送器只能在此时被启动。										
4	1	RFU										
3	PolSign	PolSign 定义 SIGIN 管脚的极性。PolSign 为 1 时, SIGIN 管脚高电平有效; PolSign 为 0 时, SIGIN 管脚低电平有效。 注: 内部的包络信号 (envelope signal) 被编码为低有效。										
2	1	RFU										
1-0	CRCPreset	定义 CRC 协处理器 CalCRC 命令的预置值。 注: 通信过程中, 预置值根据模式的定义自动选择。 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>状态</th> <th>描述</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>0000</td> </tr> <tr> <td>01</td> <td>6363</td> </tr> <tr> <td>10</td> <td>A671</td> </tr> <tr> <td>11</td> <td>FFFF</td> </tr> </tbody> </table>	状态	描述	00	0000	01	6363	10	A671	11	FFFF
状态	描述											
00	0000											
01	6363											
10	A671											
11	FFFF											

5.2.2.3 TxModeReg

定义发送过程的数据速率。

表 39 TxModeReg

TxModeReg	地址: 0x12		复位值: 00000000 (0x00)					
	7	6	5	4	3	2	1	0
	TxCRCEn	TxSpeed			InvMod	000		
访问权限	r/w	dy	dy	dy	r/w	RFU		

表 40 TxModeReg 的位描述

位	符号	功能																		
7	TxCRCEn	如果该位置位，则使能在数据发送过程中产生 CRC。																		
6-4	TxSpeed	定义数据发送的位速率。 MFRC522 的传输速率高达 424kbit/s。 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>状态</th> <th>描述</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>106kbit/s</td> </tr> <tr> <td>001</td> <td>212kbit/s</td> </tr> <tr> <td>010</td> <td>424kbit/s</td> </tr> <tr> <td>011</td> <td>RFU</td> </tr> <tr> <td>100</td> <td>RFU</td> </tr> <tr> <td>101</td> <td>RFU</td> </tr> <tr> <td>110</td> <td>RFU</td> </tr> <tr> <td>111</td> <td>RFU</td> </tr> </tbody> </table>	状态	描述	000	106kbit/s	001	212kbit/s	010	424kbit/s	011	RFU	100	RFU	101	RFU	110	RFU	111	RFU
状态	描述																			
000	106kbit/s																			
001	212kbit/s																			
010	424kbit/s																			
011	RFU																			
100	RFU																			
101	RFU																			
110	RFU																			
111	RFU																			
3	InvMod	该位置位时，发送数据的调制反相。																		
2-0	000	RFU																		

5.2.2.4 RxModeReg

定义接收过程的数据速率。

表 41 RxModeReg

RxModeReg 地址: 0x13 复位值: 00000000 (0x00)

7	6	5	4	3	2	1	0
RxCRCEn	RxSpeed			RxNoErr	000		
访问权限	r/w	dy	dy	dy	r/w	RFU	

表 42 RxModReg 的位描述

位	符号	功能																		
7	RxCRCEn	如果该位置位，则使能在数据接收过程中产生 CRC。 注：该位只能在 106kbit/s 的传输中清零。																		
6-4	RxSpeed	定义数据接收的位速率。 MFRC522 的传输速率高达 424kbit/s。 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>状态</th> <th>描述</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>106kbit/s</td> </tr> <tr> <td>001</td> <td>212kbit/s</td> </tr> <tr> <td>010</td> <td>424kbit/s</td> </tr> <tr> <td>011</td> <td>RFU</td> </tr> <tr> <td>100</td> <td>RFU</td> </tr> <tr> <td>101</td> <td>RFU</td> </tr> <tr> <td>110</td> <td>RFU</td> </tr> <tr> <td>111</td> <td>RFU</td> </tr> </tbody> </table>	状态	描述	000	106kbit/s	001	212kbit/s	010	424kbit/s	011	RFU	100	RFU	101	RFU	110	RFU	111	RFU
状态	描述																			
000	106kbit/s																			
001	212kbit/s																			
010	424kbit/s																			
011	RFU																			
100	RFU																			
101	RFU																			
110	RFU																			
111	RFU																			
3	RxNoErr	忽略接收到的无效数据流（接收到的数据小于 4 位）。接收器仍然保持有效。																		
2-0	000	RFU																		

5.2.2.5 TxControlReg

控制天线驱动器管脚 Tx1 和 Tx2 的逻辑操作。

表 43 TxControlReg

TxControlReg	地址: 0x14				复位值: 10000000 (0x80)			
	7	6	5	4	3	2	1	0
	InvTX2RF On	InvTX1RF On	InvTX2RF Off	InvTX1RF Off	Tx2CW	0	Tx2RFEn	Tx1RFEn
访问权限	r/w	r/w	r/w	r/w	r/w	RFU	r/w	r/w

表 44 TxControlReg 的位描述

位	符号	功能
7	InvTX2RFOn	如果驱动器 TX2 被使能, 则该位置位, TX2 管脚的输出信号反相。
6	InvTX1RFOn	如果驱动器 TX1 被使能, 则该位置位, TX1 管脚的输出信号反相。
5	InvTX2RFOff	如果驱动器 TX2 被禁能, 则该位置位, TX2 管脚的输出信号反相。
4	InvTX1RFOff	如果驱动器 TX1 被禁能, 则该位置位, TX1 管脚的输出信号反相。
3	Tx2CW	该位置位时, TX2 管脚的输出信号不断传递未调制的 13.56MHz 的能量载波信号。 该位清零时, Tx2CW 使能调制 13.56MHz 的能量载波信号。
2	0	RFU
1	Tx2RFEn	该位置位时, TX2 管脚的输出信号将传递经发送数据调制的 13.56MHz 的能量载波信号。
0	Tx1RFEn	该位置位时, TX1 管脚的输出信号将传递经发送数据调制的 13.56MHz 的能量载波信号。

5.2.2.6 RFU 寄存器

保留为将来之用的寄存器。

表 45 RFUReg

RFUReg	地址: 0x15				复位值: 01000000 (0x40)			
	7	6	5	4	3	2	1	0
	01000000							
访问权限	RFU							

表 46 RFUReg 的位描述

位	符号	功能
7-0	01000000	RFU

5.2.2.7 TxSelReg

选择模拟部分的内部源。

表 47 TxSelReg

TxSelReg	地址: 0x16		复位值: 00010000 (0x10)							
	7	6	5	4	3	2	1	0		
	00		DriverSel		SigOutSel					
访问权限	RFU	r/w	r/w	r/w	r/w	r/w	r/w	r/w		

表 48 TxSelReg 的位描述

位	符号	功能
7-6	00	RFU
5-4	DriverSel	选择驱动器 Tx1 和 Tx2 的输入。 状态 描述 00 三态 注: 如果 DriverSel 设置成三态模式, 则在软掉电 (softpower down) 中驱动器只能处于三态模式。 01 内部编码器的调制信号 (包络) 10 RFU 11 高电平 注: 高电平取决于 InvTx1RFOn/InvTx1RFOff 和 Tx2RFOn/InvTx2RFOff 的设置。
3-0	SigOutSel	选择 SigOut 管脚的输入。 状态 描述 0000 三态 0001 低电平 0010 高电平 0011 TestBusBitSel 寄存器定义的测试总线信号 0100 内部编码器的调制信号 (包络) 0101 发送的串行数据流 0110 RFU 0111 TestBusBitSel 寄存器定义的接收到的串行数据流 1000-1011 RFU 1100-1111 RFU

5.2.2.8 RxSelReg

选择内部接收器设置。

表 49 RxSelReg

RxSelReg	地址: 0x17		复位值: 10000100 (0x84)					
	7	6	5	4	3	2	1	0
	UartSel		RxWait					
访问权限	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

表 50 RxSelReg 的位描述

位	符号	功能
7-6	UartSel	选择非接触式 UART 的输入 状态 描述 00 固定的低电平 01 RFU 10 内部模拟部分的调制信号 11 RFU
5-0	RxWait	数据发送后，接收器的启动会延迟 RxWait 个位时钟。在这段‘帧保护时间 (frame guard time)’内，Rx 管脚上的所有信号都被忽略。 接收命令忽略此参数。但所有其它命令（如收发命令、MFAuthent 命令）会考虑该参数。

5.2.2.9 RxThresholdReg

选择位译码器的阈值。

表 51 RxThresholdReg

RxThresholdReg 地址: 0x18 复位值: 10000100 (0x84)

	7	6	5	4	3	2	1	0
	MinLevel				0	CollLevel		
访问权限	r/w	r/w	r/w	r/w	RFU	r/w	r/w	r/w

表 52 RxThresholdReg 的位描述

位	符号	功能
7-4	MinLevel	定义应当接受的译码器输入的最小信号强度。如果信号强度低于此字段的值，则该信号被忽略。
3	0	RFU
2-0	CollLevel	定义译码器输入的最小信号强度，曼彻斯特编码的信号的弱半位必须达到这个强度，来产生与较强半位的幅度相关的一个位冲突。

5.2.2.10 DemodReg

定义解调器的设置。

表 53 DemodReg

DemodReg 地址: 0x19 复位值: 01001101 (0x4D)

	7	6	5	4	3	2	1	0
	AddIQ	FixIQ	0	TauRcv		TauSync		
访问权限	r/w	r/w	r/w	RFU	r/w	r/w	r/w	r/w

表 54 DemodReg 的位描述

位	符号	功能										
7-6	AddIQ	定义接收过程中 I 和 Q 通道的使用。 注：要启用下面的设置，FixIQ 必须为 0。 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>状态</th> <th>描述</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>选择更强的通道</td> </tr> <tr> <td>01</td> <td>选择更强的通道，冻结通信中选择的通道</td> </tr> <tr> <td>10</td> <td>RFU</td> </tr> <tr> <td>11</td> <td>RFU</td> </tr> </tbody> </table>	状态	描述	00	选择更强的通道	01	选择更强的通道，冻结通信中选择的通道	10	RFU	11	RFU
状态	描述											
00	选择更强的通道											
01	选择更强的通道，冻结通信中选择的通道											
10	RFU											
11	RFU											
5	FixIQ	如果该位置位且 AddIQ 的低位为 0，则接收固定使用 I 通道。 如果该位置位且 AddIQ 的低位为 1，则接收固定使用 Q 通道。										
4	0	RFU										
3-2	TauRcv	更改数据接收过程中内部 PLL 的时间常数。										
1-0	TauSync	更改突发过程中内部 PLL 的时间常数。										

RFU 寄存器

保留为将来之用的寄存器。

表 55 RFUReg

RFUReg	地址：0x1A	复位值：00000000 (0x00)
	7 6 5 4 3 2 1 0	
	00000000	
访问权限	RFU	

表 56 RFUReg 的位描述

位	符号	功能
7-0	00000000	RFU

5.2.2.11 RFUReg

保留为将来之用的寄存器。

表 57 RFUReg

RFUReg	地址：0x1B	复位值：00000000 (0x00)
	7 6 5 4 3 2 1 0	
	00000000	
访问权限	RFU	

表 58 RFUReg 的位描述

位	符号	功能
7-0	00000000	RFU

5.2.2.12 RFUReg

保留为将来之用的寄存器。

表 59 RFUReg

RFUReg	地址: 0x1C				复位值: 00000010 (0x02)			
	7	6	5	4	3	2	1	0
	00000010							
访问权限	RFU							

表 60 RFUReg 的位描述

位	符号	功能
7-0	00000010	RFU

5.2.2.13 RFUReg

表 61 RFUReg

RFUReg	地址: 0x1D, 0x1E				复位值: 00000000 (0x00)			
	7	6	5	4	3	2	1	0
	00000000							
访问权限	RFU							

表 62 RFUReg 的位描述

位	符号	功能
7-0	00000000	RFU

5.2.2.14 SerialSpeedReg

选择串行 UART 接口的速率。

表 63 SerialSpeedReg

SerialSpeedReg	地址: 0x1F				复位值: 11101011 (0xEB)			
	7	6	5	4	3	2	1	0
	BR_T0			BR_T1				
访问权限	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

表 64 SerialSpeedReg 的位描述

位	符号	功能
7-5	BR_T0	BR_T0 用来调整传输速率, 相关描述见 7.3.2 节。
4-0	BR_T1	BR_T1 用来调整传输速率, 相关描述见 7.3.2 节。

5.2.3 PAGE2: 配置

5.2.3.1 RFUReg

保留为将来之用的寄存器。

表 65 RFUReg

RFUReg	地址: 0x20				复位值: 00000000 (0x00)			
	7	6	5	4	3	2	1	0
	00000000							
访问权限	RFU							

表 66 RFUReg 的位描述

位	符号	功能
7-0	00000000	RFU

5.2.3.2 CRCResultReg

显示 CRC 计算的实际 MSB 和 LSB 值。

注：该寄存器分成 2 个 8 位的寄存器。

表 67 CRCResultReg

CRCResultReg	地址: 0x21	复位值: 11111111 (0xFF)					
	7 6 5 4 3 2 1 0						
	CRCResultMSB						
访问权限	r	r	r	r	r	r	r

表 68 CRCResultReg 的位描述

位	符号	功能
7-0	CRCResultMSB	该寄存器显示了 CRC 寄存器的高字节的实际值。只有寄存器 Status1Reg 的 CRCReady 位置位时 CRCResultMSB 的内容才有效。

表 69 CRCResultReg

CRCResultReg	地址: 0x22	复位值: 11111111 (0xFF)					
	7 6 5 4 3 2 1 0						
	CRCResultLSB						
访问权限	r	r	r	r	R	r	r

表 70 CRCResultReg 的位描述

位	符号	功能
7-0	CRCResultLSB	该寄存器显示了 CRC 寄存器的低字节的实际值。只有寄存器 Status1Reg 的 CRCReady 位置位时 CRCResultLSB 的内容才有效。

5.2.3.3 RFUReg

表 71 RFUReg

RFUReg	地址: 0x23	复位值: 00000000 (0x00)					
	7 6 5 4 3 2 1 0						
	00000000						
访问权限	RFU						

表 72 RFUReg 的位描述

位	符号	功能
7-0	00000000	RFU

5.2.3.4 ModWidthReg

控制调制宽度的设置。

表 73 ModWidthReg

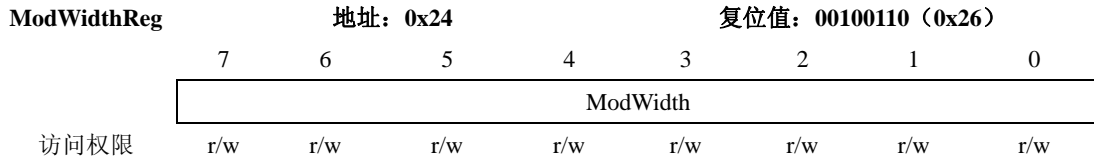


表 74 ModWidthReg 的位描述

位	符号	功能
7-0	ModWidth	这些位将 Miller 调制宽度定义成载波频率的倍数 (ModWidth+1/fc)。最大值是半个位周期。

5.2.3.5 RFUReg

保留为将来之用的寄存器。

表 75 RFUReg

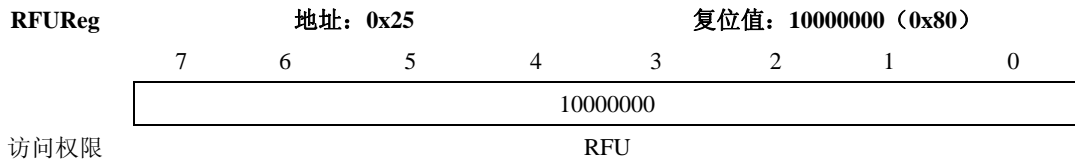


表 76 RFUReg 的位描述

位	符号	功能
7-0	10000000	RFU

5.2.3.6 RFCfgReg

配置接收器增益。

表 77 RFCfgReg

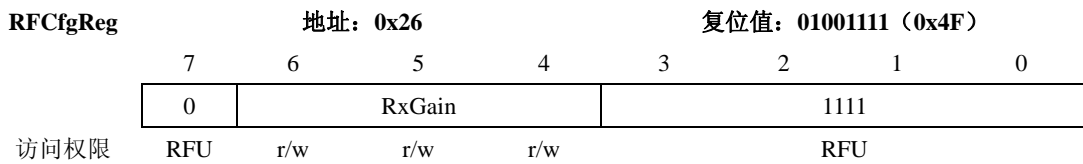


表 78 RFCfgReg 的位描述

位	符号	功能																		
7	0	RFU																		
6-4	RxGain	该寄存器定义了接收器信号电压的增益因子： <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>状态</th> <th>描述</th> </tr> </thead> <tbody> <tr><td>000</td><td>18dB</td></tr> <tr><td>001</td><td>23dB</td></tr> <tr><td>010</td><td>18dB</td></tr> <tr><td>011</td><td>23dB</td></tr> <tr><td>100</td><td>33dB</td></tr> <tr><td>101</td><td>38dB</td></tr> <tr><td>110</td><td>43dB</td></tr> <tr><td>111</td><td>48dB</td></tr> </tbody> </table>	状态	描述	000	18dB	001	23dB	010	18dB	011	23dB	100	33dB	101	38dB	110	43dB	111	48dB
状态	描述																			
000	18dB																			
001	23dB																			
010	18dB																			
011	23dB																			
100	33dB																			
101	38dB																			
110	43dB																			
111	48dB																			
3-0	1111	RFU																		

5.2.3.7 GsNReg

选择天线驱动器管脚 TX1 和 TX2 的 N 驱动器的电导。

表 79 GsNReg

GsNReg 地址: 0x27 复位值: 10001000 (0x88)

7	6	5	4	3	2	1	0
CWGsN				ModGsN			

访问权限 r/w r/w r/w r/w r/w r/w r/w r/w

表 80 GsNReg 的位描述

位	符号	功能
7-4	CWG _s N	此字段定义了输出 N 驱动器的电导。它用来调节输出功率以及后面的消耗电流和工作距离。 注：电导值用二进制数来计量。 注：在软掉电模式中最高位强制为 1。
3-0	ModGsN	此字段定义了输出 N 驱动器的电导，便于时间的调制。它可用来调节调制系数 (modulation index)。 注：电导值用二进制数来计量。 注：在软掉电模式中最高位强制为 1。

5.2.3.8 CWGsPReg

定义 P 驱动器的电导。

表 81 CWGsPReg

CWG_sPReg 地址: 0x28 复位值: 00100000 (0x20)

7	6	5	4	3	2	1	0
0	CWGsP						

访问权限 RFU r/w r/w r/w r/w r/w r/w

表 82 CWGsPReg 的位描述

位	符号	功能
7-6	00	RFU
5-0	CWGSP	此字段定义了输出 P 驱动器的电导。它用来调节输出功率以及后面的消耗电流和工作距离。 注：电导值用二进制数来计量。 注：在软掉电模式中最高位强制为 1。

5.2.3.9 ModeGsPReg

定义驱动器 P 输出电导，便于时间的调制。

表 83 ModGsPRg

ModGsPRg 地址：0x29 复位值：00100000 (0x20)

	7	6	5	4	3	2	1	0
	0		ModGsP					
访问权限	RFU		r/w	r/w	r/w	r/w	r/w	r/w

表 84 ModGsPRg 的位描述

位	符号	功能
7-6	00	RFU
5-0	ModGsP	此字段定义了输出 P 驱动器的电导，便于时间的调制。它可用来调节调制系数。 注：电导值用二进制数来计量。 注：在软掉电模式中最高位强制为 1。

5.2.3.10 TMode 寄存器，TPrescaler 寄存器

定义内部定时器的设置。

注：该寄存器分为 2 个 8 位的寄存器。

表 85 TModeReg

TModeReg 地址：0x2A 复位值：00000000 (0x00)

	7	6	5	4	3	2	1	0
	TAuto	TGated		TAuto Restart	TPrescaler_Hi			
访问权限	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

表 86 TModeReg 的位描述

位	符号	功能								
7	TAuto	该位置位时，定时器在所有速率的发送结束时自动启动。在接收到第一个数据位后定时器立刻停止运行。如果该位清零，表明定时器不受通信协议的影响。								
6-5	TGated	内部定时器工作在门控模式。 注：在门控模式中，当定时器通过寄存器的位被使能时，TRunning 位置位。它不受门控信号的影响。								
		<table border="1"> <thead> <tr> <th>状态</th> <th>描述</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>非门控模式</td> </tr> <tr> <td>01</td> <td>SIGIN 用作门控信号</td> </tr> <tr> <td>10</td> <td>AUX1 用作门控信号</td> </tr> <tr> <td>11</td> <td>A3 用作门控信号</td> </tr> </tbody> </table>	状态	描述	00	非门控模式	01	SIGIN 用作门控信号	10	AUX1 用作门控信号
状态	描述									
00	非门控模式									
01	SIGIN 用作门控信号									
10	AUX1 用作门控信号									
11	A3 用作门控信号									
4	TAutoRestart	该位置位时，定时器自动重新从 TReloadValue 的值开始递减计数，而不是从 0 计数值开始操作。 该位清零时，定时器递减计数到 0，TimerIRq 位设置为 1。								
3-0	TPrescaler_Hi	定义 TPrescaler 的高 4 位。 利用下面的公式来计算 f_{Timer} ： $f_{Timer} = 6.78MHz / TPreScaler$ 详情见第 10 章。								

表 87 TPrescalerReg

TPrescalerReg	地址：0x2B	复位值：00000000 (0x00)					
	7 6 5 4 3 2 1 0						
	TPrescaler_Lo						
访问权限	r/w	r/w	r/w	r/w	r/w	r/w	r/w

表 88 TPrescalerReg 的位描述

位	符号	功能
7-0	TPrescaler_Lo	定义 TPrescaler 的低 8 位。 利用下面的公式来计算 f_{Timer} ： $f_{Timer} = 6.78MHz / TPreScaler$ 详情见第 10 章。

5.2.3.11 TReloadReg

描述 16 位长的定时器重装值。

注：该寄存器分成 2 个 8 位的寄存器。

表 89 TReloadReg (高位)

TReloadReg	地址：0x2C	复位值：00000000 (0x00)					
	7 6 5 4 3 2 1 0						
	TReloadVal_Hi						
访问权限	r/w	r/w	r/w	r/w	r/w	r/w	r/w

表 90 TReloadReg 的位描述

位	符号	功能
7-0	TReloadVal_Hi	定义 TReloadReg 的高 8 位。 当一个启动事件出现时, TReload 的值装入定时器。只有下次启动事件出现时, 此寄存器的内容才会改变, 进而影响定时器。

表 91 TReloadReg (低位)

TReloadReg **地址: 0x2D** **复位值: 00000000 (0x00)**

	7	6	5	4	3	2	1	0
	TReloadVal_Lo							
访问权限	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

表 92 TReloadReg 的位描述

位	符号	功能
7-0	TReloadVal_Lo	定义 TReloadReg 的低 8 位。 当一个启动事件出现时, TReload 的值装入定时器。只有下次启动事件出现时, 此寄存器的内容才会改变, 进而影响定时器。

5.2.3.12 TCounterValReg

定义定时器的当前值。

注: 该寄存器分成 2 个 8 位的寄存器。

表 93 TCounterValReg (高位)

TCounterValReg **地址: 0x2E** **复位值: xxxxxxxx (0xXX)**

	7	6	5	4	3	2	1	0
	TCounterVal_Hi							
访问权限	r	r	r	r	r	r	r	r

表 94 TCounterValReg 的位描述

位	符号	功能
7-0	TCounterVal_Hi	定时器的当前值, 高 8 位。

表 95 TcounterValReg (低位)

TCounterValReg **地址: 0x2F** **复位值: xxxxxxxx (0xXX)**

	7	6	5	4	3	2	1	0
	TCounterVal_Lo							
访问权限	r	r	r	r	r	r	r	r

表 96 TCounterValReg 的位描述

位	符号	功能
7-0	TCounterVal_Lo	定时器的当前值, 低 8 位。

5.2.4 PAGE3: 测试

5.2.4.1 RFUReg

保留为将来之用的寄存器。

表 97 RFUReg

RFUReg	地址: 0x03				复位值: 00000000 (0x00)			
	7	6	5	4	3	2	1	0
	00000000							
访问权限	RFU							

表 98 RFUReg 的位描述

位	符号	功能
7-0	00000000	RFU

5.2.4.2 TestSel1Reg

常用测试信号配置。

表 99 TestSel1Reg

TestSel1Reg	地址: 0x31				复位值: 00000000 (0x00)			
	7	6	5	4	3	2	1	0
	00000					TstBusBitSel		
访问权限	RFU					r/w	r/w	r/w

表 100 TestSel1Reg 的位描述

位	符号	功能
7-3	00000	RFU
2-0	TstBusBitSel	从测试总线选择 TstBusBit。

5.2.4.3 TestSel2Reg

常用测试信号配置和 PRBS 控制。

表 101 TestSel2Reg

TestSel2Reg	地址: 0x32				复位值: 00000000 (0x00)			
	7	6	5	4	3	2	1	0
	TstBusFlip	PRBS9	PRBS15	TestBusSel				
访问权限	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

表 102 TestSel2Reg 的位描述

位	符号	功能
7	TstBusFlip	如果该位置位，测试总线按下面的顺序映射到并行端口： TstBusBit4, TstBusBit3, TstBusBit2, TstBusBit6, TstBusBit5, TstBusBit0, 见第 16 章。
6	PRBS9	根据 ITU-T0150 来启动和使能 PRBS9 序列。 注：所有相关的发送数据的寄存器必须先配置好再进入 PRBS9 模式。 注：定义序列的数据发送通过发送命令来启动。
5	PRBS15	根据 ITU-T0150 来启动和使能 PRBS15 序列。 注：所有相关的发送数据的寄存器必须先配置好再进入 PRBS15 模式。 注：定义序列的数据发送通过发送命令来启动。
4-0	TestBusSel	选择测试总线。见第 16 章。

5.2.4.4 TestPinEnReg

使能测试总线的管脚输出驱动器。

表 103 TestPinEnReg

TestPinEnReg	地址: 0x33							复位值: 10000000 (0x80)
	7	6	5	4	3	2	1	0
	TestPinEn							
访问权限	RFU	r/w	r/w	r/w	r/w	r/w	r/w	r/w

表 104 TestPinEnReg 的位描述

位	符号	功能
7	0	RFU
6-0	TestPinEn	使能 D1-D7 的管脚输出驱动器。 例如： 置位位 1 来使能 D1 置位为 5 来使能 D5 注： 如果使用 SPI 接口，则只有 D1-D4 可以使用。

5.2.4.5 TestPinValueReg

定义测试端口用作 I/O 口时的值。

表 105 TestPinValueReg

TestPinValueReg	地址: 0x34							复位值: 00000000 (0x00)
	7	6	5	4	3	2	1	0
	UseIO	TestPinValue						
访问权限	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

表 106 TestPinValueReg 的位描述

位	符号	功能
7	UseIO	如果使用了串行接口，则置位该位来使能测试端口的 I/O 功能。测试端口的输入/输出操作由 TestPinEnReg 寄存器的 TestPinEn 来定义。输出值在 TestPin 值中定义。
6-0	TestPinValue	如果测试端口用作 I/O 口，则该字段用来定义它们的值。 每个输出必须通过 TestPinEnReg 寄存器中的 TestPinEn 位来使能。 注：如果 UseIO 置位，则读出的该字段值就是 D6-D1 管脚的值。如果 UseIO 清零，则 TestPinValueReg 寄存器的值被读回。

5.2.4.6 TestBusReg

显示内部测试总线的状态。

表 107 TestBusReg

TestBusReg	地址: 0x35				复位值: xxxxxxxx (0xXX)			
	7	6	5	4	3	2	1	0
	TestBus							
访问权限	r	r	r	r	r	r	r	r

表 108 TestBusReg 的位描述

位	符号	功能
7-0	TestBus	显示内部测试总线的状态。测试总线由寄存器 TestSel2Reg 来选择。见第 16 章。

5.2.4.7 AutoTestReg

控制数字自测试。

表 109 AutoTestReg

AutoTestReg	地址: 0x36				复位值: 01000000 (0x40)			
	7	6	5	4	3	2	1	0
	0	1	0	0	SelfTest			
访问权限	RFT	RFT	RFT	RFT	r/w	r/w	r/w	r/w

表 110 AutoTestReg 的位描述

位	符号	功能
7-4	0100	RFT。
3-0	SelfTest	使能数字自测试。自测试由命令寄存器中的自测试命令来启动。自测试通过 1001 来使能。 注：默认条件下自测试被禁止 (0000)。

5.2.4.8 VersionReg

显示版本。

表 111 VersionReg

VersionReg	地址: 0x37				复位值: xxxxxxxx (0xXX)			
	7	6	5	4	3	2	1	0
	Version							
访问权限	r	r	r	r	r	r	r	r

表 112 VersionReg 的位描述

位	符号	功能
7-0	Version	指示当前版本。

5.2.4.9 AnalogTestReg

控制管脚 AUX1 和 AUX2。

表 113 AnalogTestReg

AnalogTestReg	地址: 0x38				复位值: 00000000 (0x00)			
	7	6	5	4	3	2	1	0
	AnalogSelAux1				AnalogSelAux2			
访问权限	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w

表 114 AnalogTestReg 的位描述

位	符号	功能																																		
7-4	AnalogSelAux1	控制 AUX 管脚。注：所有测试信号见第 16 章的描述。																																		
3-0	AnalogSelAux2	<table border="0"> <tr> <td>状态</td> <td>描述</td> </tr> <tr> <td>0000</td> <td>三态</td> </tr> <tr> <td>0001</td> <td>TestDAC1(AUX1)的输出，TESTDAC2(AUX2)的输出 注：电流输出。AUX 使用 1kΩ的下拉电阻。</td> </tr> <tr> <td>0010</td> <td>Testsignal Corr1 注：电流输出。AUX 使用 1kΩ的下拉电阻。</td> </tr> <tr> <td>0011</td> <td>RFU</td> </tr> <tr> <td>0100</td> <td>Testsignal MinLevel 注：电流输出。AUX 使用 1kΩ的下拉电阻。</td> </tr> <tr> <td>0101</td> <td>ADC 通道 I 注：电流输出。AUX 使用 1kΩ的下拉电阻。</td> </tr> <tr> <td>0110</td> <td>ADC 通道 Q 注：电流输出。AUX 使用 1kΩ的下拉电阻。</td> </tr> <tr> <td>0111</td> <td>Testsignal ADC 通道 I 和 Q 结合使用 注：电流输出。AUX 使用 1kΩ的下拉电阻。</td> </tr> <tr> <td>1000</td> <td>产品测试的信号 注：电流输出。AUX 使用 1kΩ的下拉电阻。</td> </tr> <tr> <td>1001</td> <td>RFU</td> </tr> <tr> <td>1010</td> <td>高电平</td> </tr> <tr> <td>1011</td> <td>低电平</td> </tr> <tr> <td>1100</td> <td>TxActive 106kbit/s：在起始位、数据位、奇偶位和 CRC 传输过程中为高电平。 212 和 424kbit/s：在数据和 CRC 传输过程为高电平。</td> </tr> <tr> <td>1101</td> <td>RxActive 106kbit/s：在数据位、奇偶位和 CRC 传输过程中为高电平。 212 和 424kbit/s：在数据和 CRC 传输过程为高电平。</td> </tr> <tr> <td>1110</td> <td>检测到副载波 106kbit/s：不适用。 212 和 424kbit/s：在最后部分的数据和 CRC 传输过程为高电平。</td> </tr> <tr> <td>1111</td> <td>测试总线寄存器定义的测试总线位。</td> </tr> </table>	状态	描述	0000	三态	0001	TestDAC1(AUX1)的输出，TESTDAC2(AUX2)的输出 注：电流输出。AUX 使用 1kΩ的下拉电阻。	0010	Testsignal Corr1 注：电流输出。AUX 使用 1kΩ的下拉电阻。	0011	RFU	0100	Testsignal MinLevel 注：电流输出。AUX 使用 1kΩ的下拉电阻。	0101	ADC 通道 I 注：电流输出。AUX 使用 1kΩ的下拉电阻。	0110	ADC 通道 Q 注：电流输出。AUX 使用 1kΩ的下拉电阻。	0111	Testsignal ADC 通道 I 和 Q 结合使用 注：电流输出。AUX 使用 1kΩ的下拉电阻。	1000	产品测试的信号 注：电流输出。AUX 使用 1kΩ的下拉电阻。	1001	RFU	1010	高电平	1011	低电平	1100	TxActive 106kbit/s：在起始位、数据位、奇偶位和 CRC 传输过程中为高电平。 212 和 424kbit/s：在数据和 CRC 传输过程为高电平。	1101	RxActive 106kbit/s：在数据位、奇偶位和 CRC 传输过程中为高电平。 212 和 424kbit/s：在数据和 CRC 传输过程为高电平。	1110	检测到副载波 106kbit/s：不适用。 212 和 424kbit/s：在最后部分的数据和 CRC 传输过程为高电平。	1111	测试总线寄存器定义的测试总线位。
状态	描述																																			
0000	三态																																			
0001	TestDAC1(AUX1)的输出，TESTDAC2(AUX2)的输出 注：电流输出。AUX 使用 1kΩ的下拉电阻。																																			
0010	Testsignal Corr1 注：电流输出。AUX 使用 1kΩ的下拉电阻。																																			
0011	RFU																																			
0100	Testsignal MinLevel 注：电流输出。AUX 使用 1kΩ的下拉电阻。																																			
0101	ADC 通道 I 注：电流输出。AUX 使用 1kΩ的下拉电阻。																																			
0110	ADC 通道 Q 注：电流输出。AUX 使用 1kΩ的下拉电阻。																																			
0111	Testsignal ADC 通道 I 和 Q 结合使用 注：电流输出。AUX 使用 1kΩ的下拉电阻。																																			
1000	产品测试的信号 注：电流输出。AUX 使用 1kΩ的下拉电阻。																																			
1001	RFU																																			
1010	高电平																																			
1011	低电平																																			
1100	TxActive 106kbit/s：在起始位、数据位、奇偶位和 CRC 传输过程中为高电平。 212 和 424kbit/s：在数据和 CRC 传输过程为高电平。																																			
1101	RxActive 106kbit/s：在数据位、奇偶位和 CRC 传输过程中为高电平。 212 和 424kbit/s：在数据和 CRC 传输过程为高电平。																																			
1110	检测到副载波 106kbit/s：不适用。 212 和 424kbit/s：在最后部分的数据和 CRC 传输过程为高电平。																																			
1111	测试总线寄存器定义的测试总线位。																																			

5.2.4.10 TestDAC1Reg

定义 TestDAC1 的测试值。

表 115 TestDAC1Reg

TestDAC1Reg	地址：0x39				复位值：00xxxxxx (0xXX)			
	7	6	5	4	3	2	1	0
	0	0	TestDAC1					
访问权限	RFT	RFU	r/w	r/w	r/w	r/w	r/w	r/w

表 116 TestDAC1Reg 的位描述

位	符号	功能
7	0	RFT
6	0	RFU
5-0	TestDAC1	定义 TestDAC1 的测试值。通过将 AnalogSelAux1 设置成 0001 可使 DAC1 的输出切换到 AUX1。

5.2.4.11 TestDAC2Reg

定义 TestDAC2 的测试值。

表 117 TestDAC2Reg

TestDAC2Reg 地址: 0x3A 复位值: 00xxxxxx (0xXX)

	7	6	5	4	3	2	1	0
	0	0	TestDAC2					
访问权限	RFU	RFU	r/w	r/w	r/w	r/w	r/w	r/w

表 118 TestDAC2Reg 的位描述

位	符号	功能
7-6	00	RFU
5-0	TestDAC2	定义 TestDAC2 的测试值。通过将 AnalogSelAux2 设置成 0001 可使 DAC2 的输出切换到 AUX2。

5.2.4.12 TestADCReg

显示 ADC I 和 Q 通道的实际值。

表 119 TestADCReg

TestADCReg 地址: 0x3B 复位值: xxxxxxxx (0xXX)

	7	6	5	4	3	2	1	0
	ADC_I				ADC_Q			
访问权限	r	r	r	r	r	r	r	r

表 120 TestADCReg 的位描述

位	符号	功能
7-4	ADC_I	显示 ADC I 通道的实际值。
3-0	ADC_Q	显示 ADC Q 通道的实际值。

5.2.4.13 RFTReg

表 121 RFTReg

RFTReg 地址: 0x3C 复位值: 11111111 (0xFF)

	7	6	5	4	3	2	1	0
	1	1	1	1	1	1	1	1
访问权限	RFT	RFT	RFT	RFT	RFT	RFT	RFT	RFT

表 122 RFTReg 的位描述

位	符号	功能
7-0	11111111	RFT

表 123 RFTReg

RFTReg 地址: 0x3D 复位值: 00000000 (0x00)

	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0
访问权限	RFT	RFT	RFT	RFT	RFT	RFT	RFT	RFT

表 124 RFTReg 的位描述

位	符号	功能
7-0	00000000	RFT

表 125 RFTReg

RFTReg 地址: 0x3E 复位值: 00000111 (0x07)

	7	6	5	4	3	2	1	0
	0	0	0	0	0	1	1	1
访问权限	RFT	RFT	RFT	RFT	RFT	RFT	RFT	RFT

表 126 RFTReg 的位描述

位	符号	功能
7-0	00000111	RFT

表 127 RFTReg

RFTReg 地址: 0x3F 复位值: 01110000 (0x70)

	7	6	5	4	3	2	1	0
	0	1	1	1	0	1	1	1
访问权限	RFT	RFT	RFT	RFT	RFT	RFT	RFT	RFT

表 128 RFTReg 的位描述

位	符号	功能
7-0	00000111	RFT

6. MFRC522 的功能

MFRC522 发送模块支持具有多种传输速率和调制方法的 ISO 14443A/MIFARE®的读写器模式。

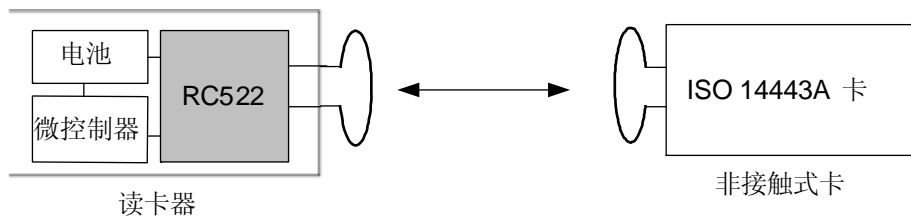


图3 MFRC522 读写器模式

ISO 14443A / MIFARE®的读写器模式是遵循ISO 14443A/ MIFARE®规范的卡通信的常用读卡器。下图描述了物理层上的通信，通信表列出了相关的参数。

- ISO 14443A / MIFARE®读写器功能的通信图

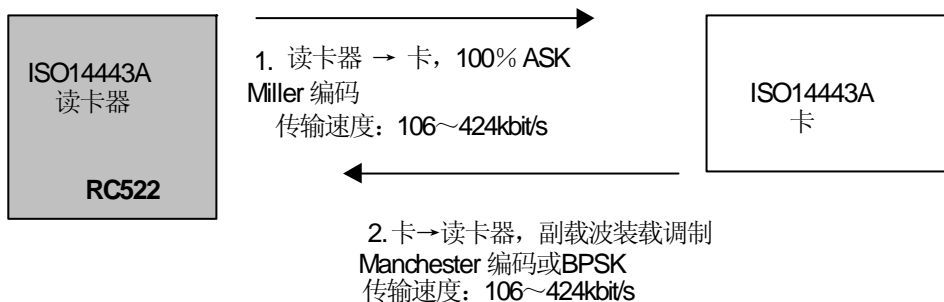


图4 ISO 14443A / MIFARE®读写器通信图

- ISO 14443A / MIFARE®读写器功能通信概述

表 129 ISO 14443A / MIFARE®读写器通信概述

通信方向	传输速率	ISO 14443A / MIFARE®		
		106kbit/s	MIFARE®更高的传输速率 212kbit/s	424kbit/s
读卡器→卡 (数据从MFRC522发送到卡)	读卡器一方的调制	100% ASK	100% ASK	100% ASK
	位编码	改变的 Miller 编码	改变的 Miller 编码	改变的 Miller 编码
	位长度	(128/13.56) us	(64/13.56) us	(32/13.56) us
卡→读卡器 (接收卡的数据)	卡一方的调制	副载波装载调制	副载波装载调制	副载波装载调制
	副载波频率	13.56MHz/16	13.56MHz/16	13.56MHz/16
	位编码	Manchester 编码	BPSK	BPSK

完成整个MIFARE®/ISO 14443A / MIFARE®协议需要使用MFRC522的非接触式UART和专用的外部主机。

- ISO 14443A / MIFARE®的数据编码和帧

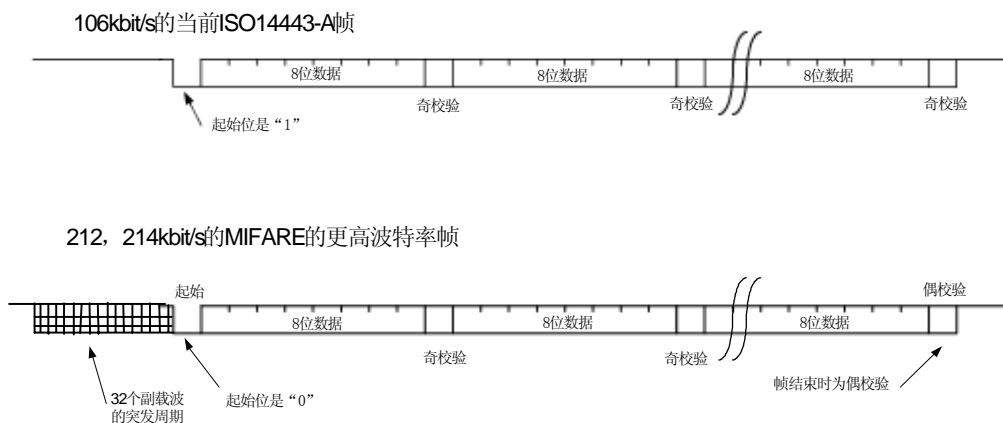


图 5 ISO 14443A 的数据编码和帧

内部 CRC 协处理器根据 ISO 14443A part3 给出的定义来计算 CRC 值。

7. 数字接口

7.1 自动检测微控制器接口类型

MFRC522 支持可直接相连的各种微控制器接口类型，如 SPI、I2C 和串行 UART。MFRC522 可复位其接口，并可对执行了上电或硬复位的当前微控制器接口的类型进行自动检测。它通过复位阶段后控制管脚上的逻辑电平来识别微控制器接口。每种接口有固定管脚的连接组合。下表列出了不同的配置：

表 130 检测不同接口类型的连接方法

MFRC522 管脚	串行接口类型		
	UART	SPI	I2C
SDA	RX	NSS	SDA
I2C	0	0	1
EA	0	1	EA
D7	TX	MISO	SCL
D6	MX	MOSI	ADR_0
D5	DTRQ	SCK	ADR_1
D4	-	-	ADR_2
D3	-	-	ADR_3
D2	-	-	ADR_4
D1	-	-	ADR_5

注：管脚操作简介

管脚操作	输入	输出	输入/输出

7.2 兼容 SPI 的接口

支持串行外围接口(兼容 SPI)来使能到主机的高速通信。SPI 接口可处理高达 10Mbit/s 的数据速率。在与主机微控制器通信时，MFRC522 用作从机，接收寄存器设置的外部微

控制器的数据以及发送和接收 RF 接口相关的通信数据。

7.2.1 概述

兼容 SPI 的接口可启用 MFRC522 和一个微控制器之间的高速串行通信。兼容 SPI 接口的处理与标准 SPI 接口相同。

时序规范请参考 18.13 节。

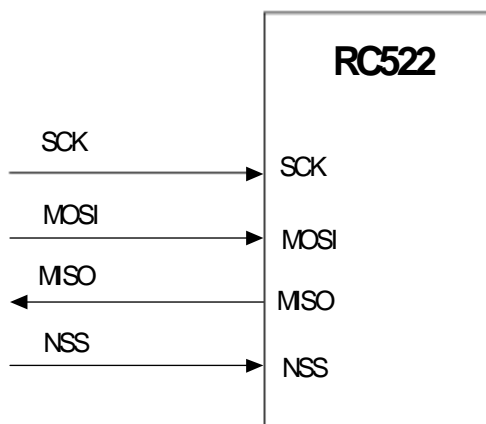


图 6 连接到带有 SPI 接口的微控制器

在 SPI 通信中 MFRC522 模块用作从机。SPI 时钟 SCK 由主机产生。数据通过 MOSI 线从主机传输到从机；数据通过 MISO 线从 MFRC522 发回到主机。

MOSI 和 MISO 传输每个字节时都是高位在前。MOSI 上的数据在时钟的上升沿保持不变，在时钟的下降沿改变。MISO 也与之类似，在时钟的下降沿，MISO 上的数据由 MFRC522 来提供，在时钟的上升沿数据保持不变。

7.2.2 读数据

使用下面的结构可将数据通过兼容 SPI 的接口读出。这样可能读出 n 个数据字节。

发送的第一个字节定义了模式本身和地址。

表 131 MOSI 和 MISO 的字节顺序

	字节 0	字节 1	字节 2	字节 n	字节 n+1
MOSI	地址 0	地址 1	地址 2	地址 n	00
MISO	X	数据 0	数据 1	数据 n-1	数据 n

注：先发送最高位（MSB）。

7.2.3 写数据

使用下面的结构可将数据通过兼容 SPI 的接口写入。这样对应一个地址可以写入多达 n 个数据字节。

发送的第一个字节定义了模式本身和地址。

注：先发送最高位（MSB）。

表 132 MOSI 和 MISO 的字节顺序

	字节 0	字节 1	字节 2	字节 n	字节 n+1
MOSI	地址	数据 0	数据 1	数据 n-1	数据 n
MISO	X	X	X	X	X

7.2.4 地址字节

地址字节按下面的格式传输。第一个字节的 MSB 位设置使用的模式。MSB 位为 1 时从 MFRC522 读出数据；MSB 位为 0 时将数据写入 MFRC522。第一个字节的位 6-1 定义地址，最后一位应当设置为 0。

表 133 地址字节格式

地址 (MOSI)	位 7, MSB	位 6—位 1	位 0
字节 0	1 (读) 0 (写)	地址	RFU (0)

7.3 UART 接口

7.3.1 连接到主机微控制器

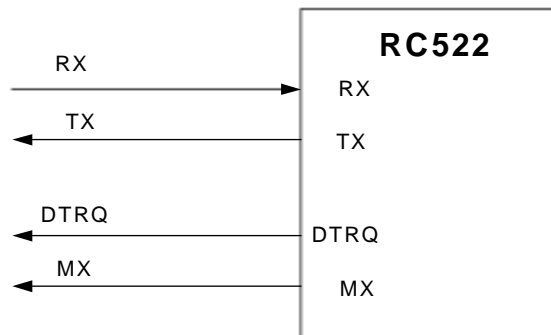


图 7 连接到带 UART 的微控制器

7.3.2 选择传输速率

内部 UART 接口兼容 RS232 串行接口。

表 135 列举了一些传输速率和对应的寄存器设置。

默认的传输速率为 9.6kbit/s。

要改变传输速率，主机控制器必须向 SerialSpeedReg 寄存器写入一个新的传输速率值。位 BR_T0 和 BR_T1 定义的因数用来设置 SerialSpeedReg 中的传输速率。

表 134 描述了 BR_T0 和 BR_T1 的设置。

表 134 BR_T0 和 BR_T1 的设置

BR_T0	0	1	2	3	4	5	6	7
因数 BR_T0	1	1	2	4	8	16	54	64
范围 BR_T1	1-32	33-64	33-64	33-64	33-64	33-64	33-64	33-64

表 135 可选的传输速率

传输速率 [bit/s]	SerialSpeedReg		传输速率精度
	十进制	十六进制	
7.2K	250	FA	-0, 25 %
9.6K	235	EB	0, 32 %
14.4K	218	DA	-0, 25 %
19.2K	203	CB	0, 32 %
38.4K	171	AB	0, 32 %
57.6K	154	9A	-0, 25 %
115.2K	122	7A	-0, 25 %
128K	116	74	-0, 06 %
230.4K	90	5A	-0, 25 %
460.8K	58	3A	-0, 25 %
921.6K	28	1C	1, 45 %
1228.8K	21	15	0, 32 %

表 135 列出的可选传输速率可通过下面的公式计算得到：

如果 BR_T0=0： 传输速率=27,12MHz / (BR_T1+1)

如果 BR_T0 > 0： 传输速率=27,12MHz / (BR_T1+33) / 2^(BR_T0 -1)

注： 不支持大于 1228.8k 的传输速率。

7.3.3 帧格式

表 136 UART 帧格式

	长度	值
起始位	1 位	0
数据位	8 位	数据
停止位	1 位	1

对于数据和地址位，先发送 LSB 位。

注： 发送数据时不使用奇偶校验位。

读数据：

使用下面的结构通过 UART 接口将数据读出。

表 137 读数据示意图

	字节 0	字节 1
RX	地址	
TX		数据 0

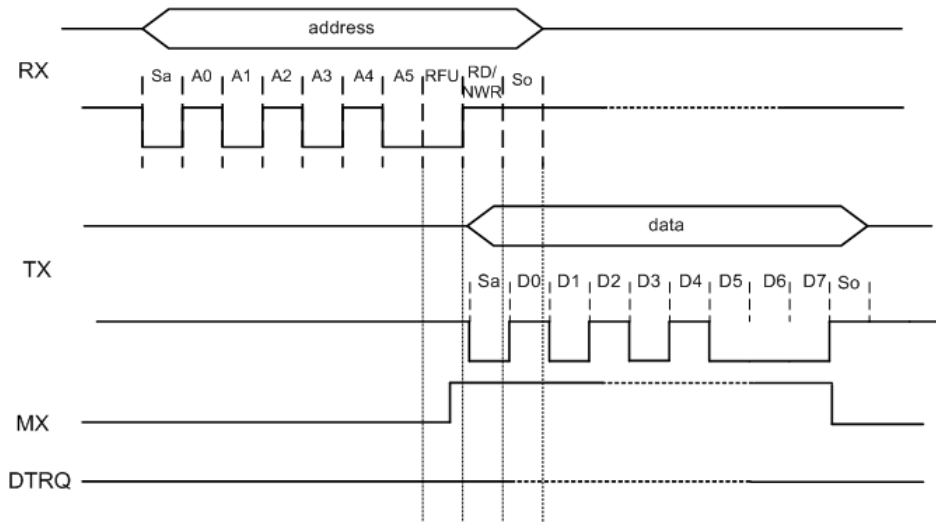


图 8 读数据示意图

写数据:

使用下面的结构通过 UART 接口将数据写入 MFRC522。

表 138 写数据示意图

	字节 0	字节 1
RX	地址 0	数据 0
TX		地址 0

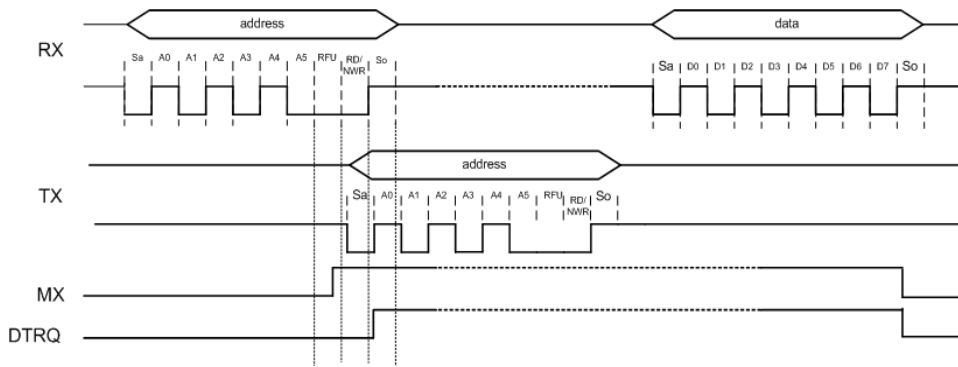


图 9 写数据示意图

注：地址字节到达 RX 线后数据字节可直接发送。

地址字节

地址字节按下面的格式传输。第一个字节的 MSB 位设置使用的模式。MSB 位为 1 时从 MFRC522 读出数据；MSB 位为 0 时将数据写入 MFRC522。第一个字节的位 6 为 RFU，位 5-1 定义地址。

表 139 地址字节

地址	位 7, MSB	位 6	位 5 - 位 0
字节 0	1 (读), 0 (写)	RFU	地址

7.4 I²C 总线接口

支持 I²C 总线接口来使能到主机的低成本、少管脚数的串行总线接口。I²C 接口操作遵循 Philips 半导体 I2C 接口规范 V2.1 (2000 年 1 月)。实现 I²C 接口时只工作在从机模式。因此, MFRC522 不产生时钟, 也不执行访问仲裁。

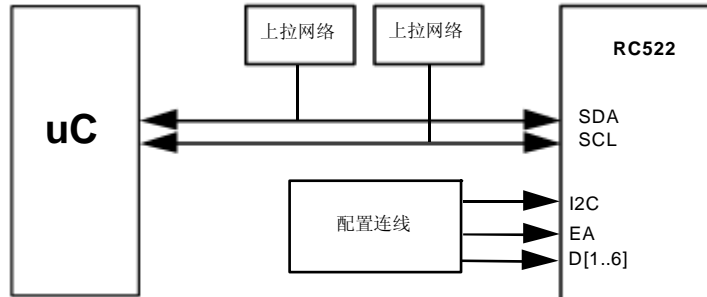


图 10 I²C 接口

7.4.1 概述

接口遵循 I²C 总线规范 V2.1 (2000 年 1 月)。在标准、快速和高速模式中, MFRC522 可用作从接收器或从发送器。

SDA 是一个双向数据线, 通过一个电流源或上拉电阻连接到正电压。如果不传输数据, SDA 和 SCL 均为高电平。MFRC522 有一个三态输出级来执行线与功能。标准模式下 I²C 总线的传输速率为 100kbit/s、快速模式下为 400kbit/s、高速模式下高达 3.4Mbit/s。

如果选择 I²C 接口, 管脚 SCL 和 SDA 管脚都有尖峰脉冲抑制功能, 完全符合 I²C 接口规范。

相关时序请参考 18.14 节。

7.4.2 数据有效

SDA 线上的数据在时钟周期的高电平时间内保持不变。只有当 SCL 的时钟信号为低电平时数据线的高电平或低电平状态才能改变。

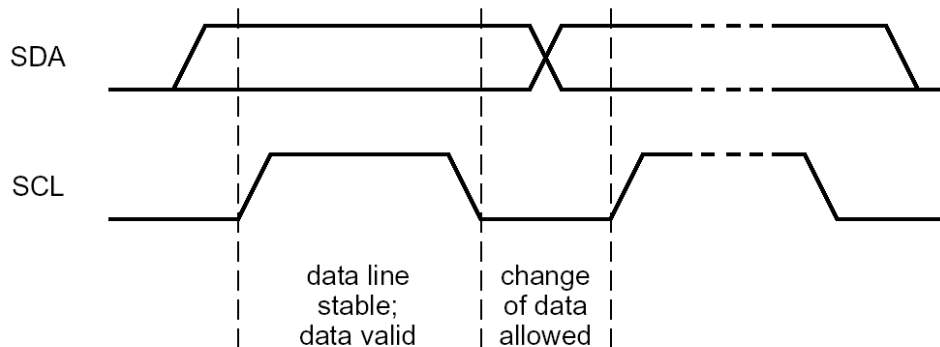


图 11 I²C 总线的位传输

7.4.3 起始和停止条件

要处理 I²C 总线的数据传输, 必须定义起始和停止条件。

起始条件定义成 SCL 高电平时间内 SDA 线上高到低的跳变。

停止条件定义成 SCL 高电平时间内 SDA 线上低到高的跳变。

起始和停止条件通常由主机产生。起始条件后主机被认为处于忙状态；主机在停止条件结束一段时间后被认为重新回到空闲状态。

如果产生的是重复起始 (Sr) 条件而非停止条件，则总线仍处于忙状态。这时，起始条件和重复起始 (Sr) 条件的功能完全相同。因此，S 符号就用作一个常用术语，代表起始和重复起始 (Sr) 条件。

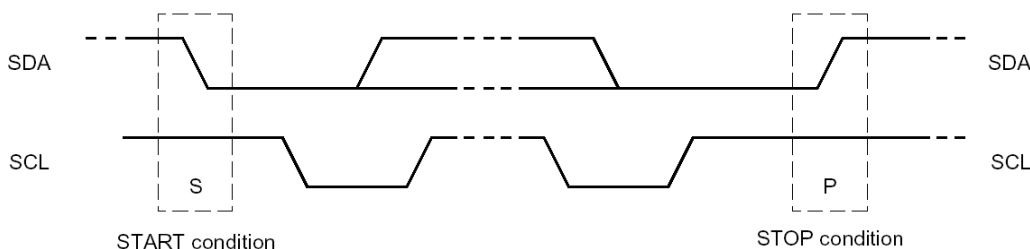


图 12 起始和停止条件

7.4.4 字节格式

一个字节由 8 个位组成。每个字节后面跟一个应答位。数据传输时高位在前，如图 15 所示。一次数据传输发送的字节数不限，但必须符合读/写周期格式。

7.4.5 应答

应答是在一个数据字节结束后强制产生的。应答相应的时钟脉冲由主机产生。在应答时钟脉冲周期内，数据发送器释放 SDA 线（高电平）；接收器拉低 SDA 线使得 SDA 线在这个时钟脉冲的高电平时间内保持低电平。

主机可以产生一个停止条件来终止传输，也可以产生一个重复起始 (Sr) 条件来启动一次新的传输。

主接收器通过在最后一个字节后不产生应答来向从发送器指示数据的结束。从发送器应当释放数据线以允许主机产生停止 (P) 或重复起始 (Sr) 条件。

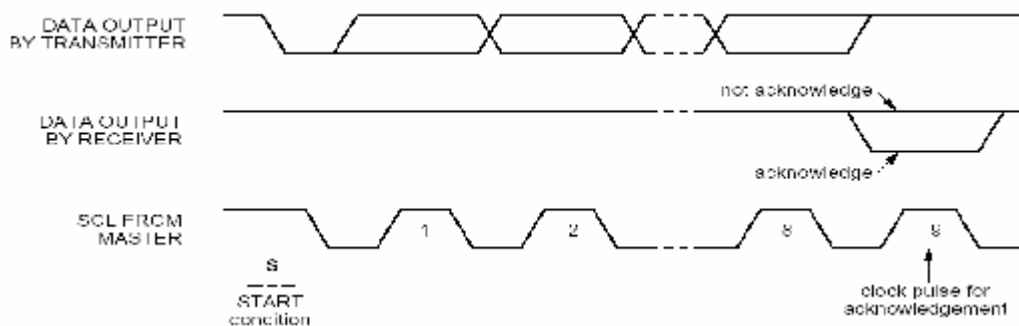


图 13 I²C 总线的应答

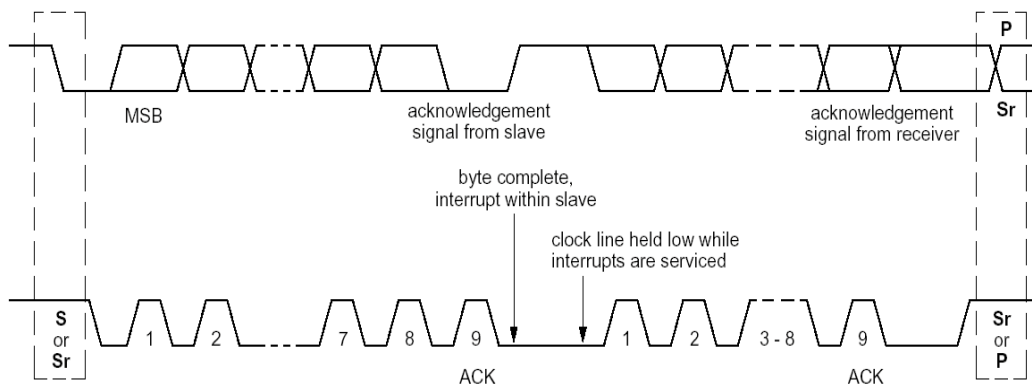


图 14 I²C 总线的数据传输

7.4.6 7 位寻址

在 I²C 总线的寻址过程中，起始条件后的第一个字节用来确定主机选择的通信从机。

有多个保留地址的情况除外。这时，在器件配置过程中，设计者必须确保器件地址不会与保留的地址产生冲突。请检查相应的 I²C 规范中的保留地址一览表。

I²C 地址规范与 EA 管脚的定义有关。在复位管脚释放或上电复位后，器件根据 EA 管脚的逻辑电平来决定总线地址。

如果 EA 管脚为高电平，则对于所有 MFRC522 的器件，器件总线地址的高 4 位由 Philips 保留，设置成 0101（二进制），从机地址剩余的 3 位（ADR_0, ADR_1, ADR_2）可由用户自由配置，这样就可以防止与其它 I²C 器件产生冲突。

如果 EA 管脚设置成低电平，则 ADR_0 – ADR_5 完全由外部管脚来指定（请参考表 131），ADR_6 总是为 0。

在这两种模式下，外部地址编码都在复位条件释放后立即锁定。不考虑数据端口上的进一步变化。通过配置外部连线，数据端口还可用作测试信号的输出。

MSB						LSB	
Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	R/ \overline{W}
从机地址							

图 15 起始条件后的第一个字节

7.4.7 寄存器写访问

使用下面的格式可将数据通过 I²C 接口从主机控制器写入指定的 MFRC522 寄存器中。

帧的第一个字节是器件地址（遵循 I²C 规则），第二个字节是寄存器地址，接下来是 n 个数据字节。在一帧中，所有 n 个数据字节都被写入相同的存储器地址。这种方法可使能 FIFO 的快速访问。

读/写标志应当清零。

7.4.8 寄存器读访问

使用下面的格式可读出指定地址的 MFRC522 寄存器中的数据。

首先，必须写入指定寄存器的地址，格式描述如下。

帧的第一个字节是器件地址（遵循 I²C 规则），第二个字节是寄存器地址。这个写寄存

器地址的操作不需要增加数据字节。

读/写标志应当清零。

写地址操作完成后，开始启动读访问。主机发送 MFRC522 的器件地址。作为回应，MFRC522 将寄存器的内容发回。一帧中所有 n 个数据字节都从相同的寄存器地址读出。这种方法可使能 FIFO 的快速访问或寄存器查询。

读/写标志应当置位。

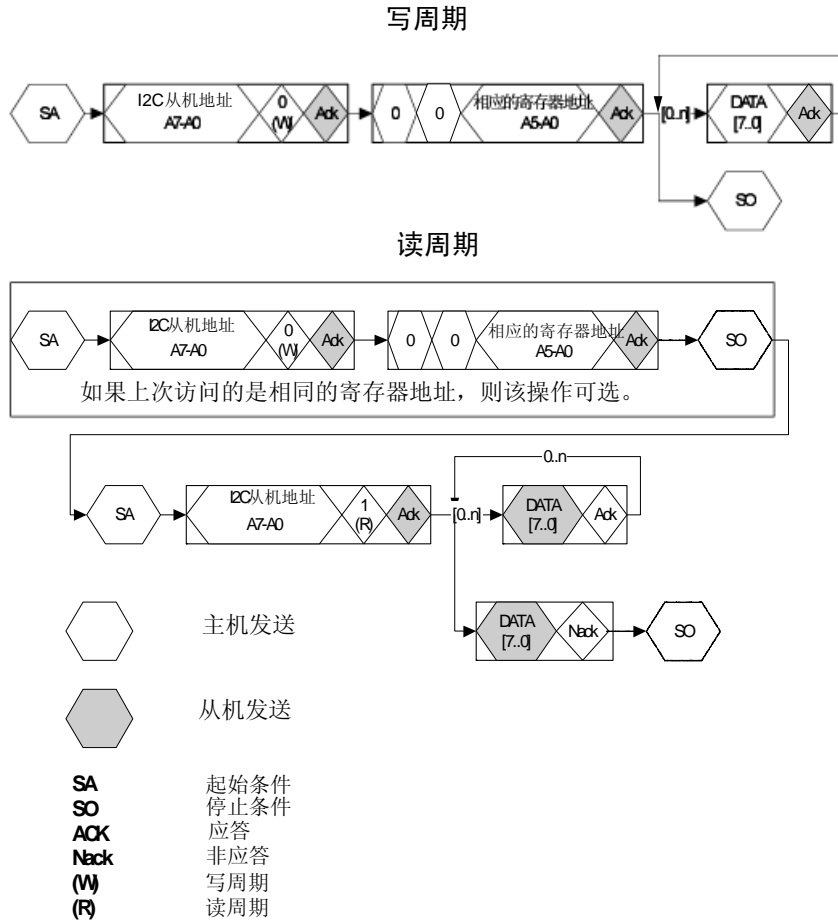


图 16 寄存器读和写访问

7.4.9 高速模式

高速模式下 (Hs 模式) 器件的传输位速率高达 3.4Mbit/s。在混合速率的总线系统中，它保持完全向下兼容快速或标准模式 (F/S 模式) 的双向通信。

7.4.10 高速传输

为了获得高达 3.4Mbit/s 的位传输速率，对常规的 I²C 总线性能作了以下改善。

- 高速模式下器件的输入具有尖峰脉冲抑制功能，SDAH 和 SCLH 输入有施密特触发器，它们与 F/S 模式相比有不同的时序常数。
- 高速模式下器件的输出可控制 SDAH 和 SCLH 信号的下降沿斜率，它们与 F/S 模式相比有不同的时序常数。

7.4.11 高速模式下的串行数据传输格式

高速模式下的串行数据传输满足标准模式的 I²C 总线规范。只有满足以下条件高速模式传输才能启动（所有 F/S 模式下的条件）：

1. 起始条件 (S)
2. 8 位主机代码 (00001XXX)
3. 非应答位 (A)

在 7 位从地址和 R/W 位之后，有效主机再发送一个重复起始条件 (Sr)，从选择的 MFRC522 中接收一个应答位 (A)。

下个重复起始条件 (Sr) 后继续执行高速模式的数据传输，停止条件 (P) 后切换回 F/S 模式。为了降低主机的代码开销，主机可能将大量的高速模式的传输链接到一起，这些传输通过起始条件分隔开来。

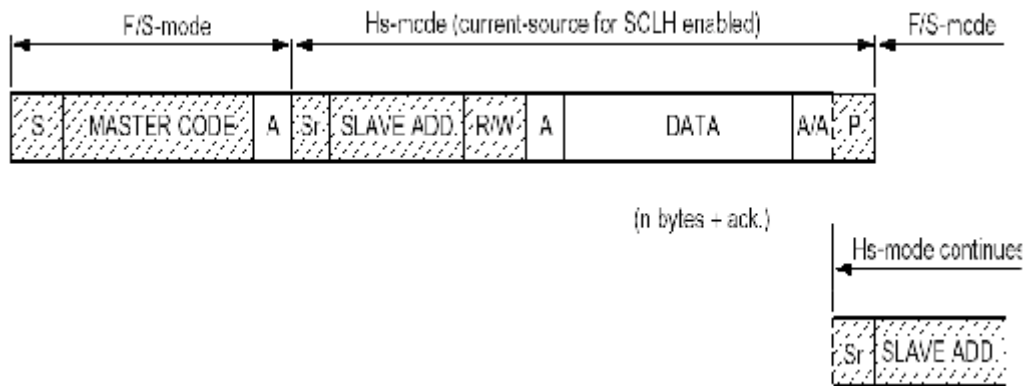


图 17 I²C 高速模式协议转换

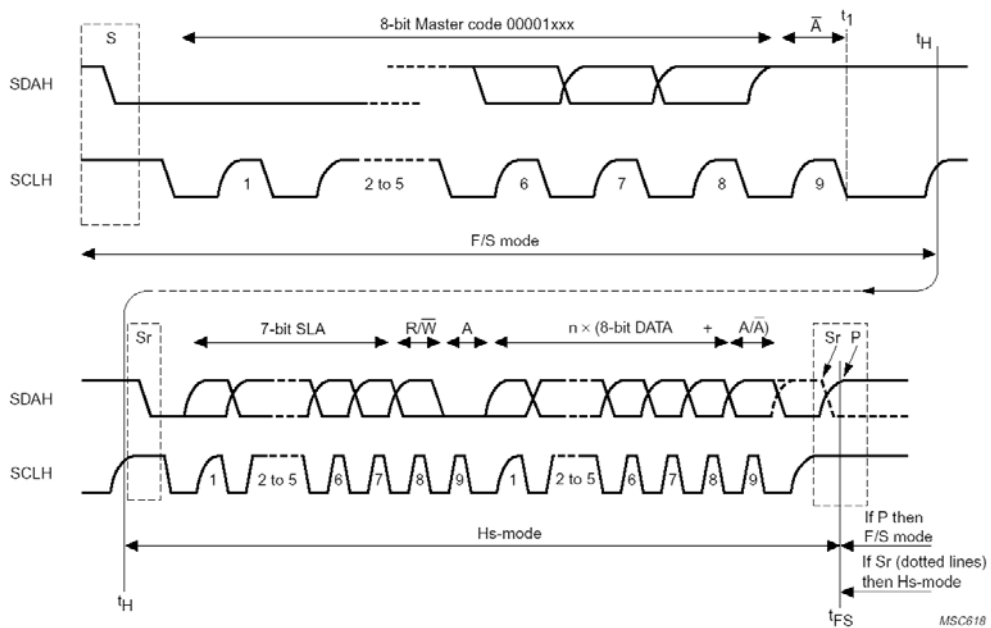


图 18 I²C 高速模式协议帧

7.4.12 F/S 模式和 HS 模式之间的切换

复位和初始化后，MFRC522 工作在快速模式（当快速模式向下兼容标准模式时，它实际上就是 F/S 模式）。连接的 MFRC522 识别到“S00001XXXXA”时将其内部电路从快速模式设置转换成高速模式设置。

MFRC522 执行以下操作：

1. 根据 Hs 模式的尖峰脉冲抑制要求来调整 SDAH 和 SCLH 的输入滤波器。
2. 调整 SDAH 输出级的斜率控制。

对于通信中不含有其它 I²C 器件的系统配置，可以通过另一种方法切换到 HS 模式：置位 Status2Reg 寄存器的 I2CForceHS 位。该位置位将使 Hs 模式永久保持，这就意味着无需再发送主机代码了。这个操作不符合总线规范，只能用在总线无其它器件的情况下。由于尖峰脉冲的减弱抑制，尖峰脉冲可忽略。

7.4.13 低速模式下的 MFRC522

MFRC522 完全向下兼容，可连接到 F/S 模式的 I²C 总线系统。由于此配置中不发送主机代码，因此器件处于 F/S 模式，以 F/S 模式的速率进行通信。

8. 模拟接口和非接触式 UART

8.1 概述

非接触式 UART 可处理与主机之间的通信协议要求。协议本身可产生面向位和字节的帧，也可处理不同的非接触式通信机制的奇偶&CRC 错误检测。

注：天线的规格和调谐以及电源电压都会对通信距离产生重大的影响。

8.2 Tx 驱动器

管脚 Tx1 和 Tx2 发送的信号是经包络信号调制的 13.56MHz 的能量载波。它可用来直接驱动天线，使用一些无源元件进行匹配和滤波，请参考第 17 章。Tx1 和 Tx2 的信号可通过 TxControlReg 寄存器来配置，见 5.2.2.5 节。

调制系数（modulation index）可通过调节驱动器的电阻进行设置。P 驱动器的电阻通过寄存器 CWGsPReg 和 ModGsPReg 来配置；n 驱动器的电阻通过寄存器 GsNReg 来配置。并且，调制系数还取决于天线的设计和调谐。

寄存器 TxModeReg 和 TxAutoSelReg 控制着发送和天线驱动器设置过程中发送数据的速率和帧，以便支持不同模式和传输速率下的不同要求。

表 140 Tx1 的设置

TX1RFEN	INVTX1	ENVELOPE	TX1	GSPMOS	GSNMOS	备注
0	0	0	0		nMod	如果 TX1RFEN=0，管脚 TX1 被设为 0 或 1，由 InvTx1 来决定。
		1	0		nCW	
	1	0	1	pMod		
		1	1	pCW		
1	1	0	0	pMod	nMod	TX1 下拉为 0，与 InvTx1 无关。
		1	RF_n	pCW	nCW	

表 141 Tx2 的设置

TX2RFEN	TX2CW	INVTX2	ENVELOPE	TX2	GSPMOS	GSNMOS	备注		
0	0	0	0	0		nMod	如果 TX2RFEN=0，管脚 TX2 被设为 0 或 1，由 InvTx2 来决定。		
			1	0		nCW			
		1	0	1	pMod				
			1	1	pCW				
	1	0	0	0	0			nCW	TX2CW：总是 gsCW 的值。
				1	0			nCW	
		1	0	1	pCW				
			1	1	pCW				
1	0	0	0	0	pMod	nMod	TX2 下拉为 0（与 InvTx2 无关）。		
			1	RF	pCW	nCW			
		1	0	0	pMod	nMod			
			1	RF_n	pCW	nCW			
	1	0	x	RF	pCW	nCW			
		1	x	RF_n	pCW	nCW			

注：

表中使用了以下缩写：

RF: 13.56MHz 时钟，由 27.12MHz 的石英晶振 2 分频所得。

RF_n: 反相 13.56MHz 的时钟

gspmos: 电导，PMOS 阵列的配置

gsnmos: 电导，NMOS 阵列的配置

pCW: CWGsP 寄存器定义连续波的 PMOS 电导值。

pMod: ModGsP 寄存器定义的调制的 PMOS 电导值。

nCW: CWGsN 寄存器定义连续波的 NMOS 电导值。

nMod: ModGsN 寄存器定义的调制的 NMOS 电导值。

8.3 串行数据变换

MFRC522 主要包含 2 个模块：1 个数字电路和 1 个模拟电路。数字电路由状态机、编码器和译码器逻辑电路组成；模拟电路包括调制器、天线驱动器、接收器和放大器电路。例如，2 个模块之间的接口必须按一定的方式进行配置，以便连接信号可以通过管脚 SIGIN 和 SIGOUT。串行数据的变换由寄存器 TxSelReg 和 RxSelReg 来控制。

下图所示为 TX1 和 TX2 的串行数据变换。

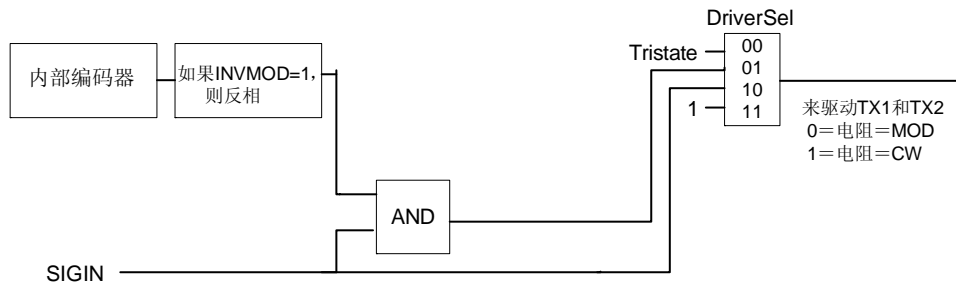


图 19 TX1 和 TX2 的串行数据变换

8.4 CRC 协处理器

CRC 协处理器操作必须对下面的参数进行配置：

表 142 CRC 协处理器参数

参数	值
CRC 寄存器长度	16 位 CRC
CRC 算法	根据 ISO14443A 和 CCITT 来制定算法
CRC 预置值	0000, 6363, A671 或 FFFF, 取决于 CRCPresetReg 寄存器的设置

16 位 CRC 的 CRC 多项式为 $X^{16}+X^{12}+X^5+1$ 。

下面的寄存器可以对协处理器进行配置。

寄存器 CRCPresetReg 定义 CRC 协处理器的预置值。该寄存器只能选择 0000、6363、A671 或 FFFF。

寄存器 CRCResultReg 用来指示 CRC 计算的结果。该寄存器分成 2 个 8 位的寄存器，分别用来代表 MSB 字节和 LSB 字节。

9. FIFO 缓冲区

9.1 概述

MFRC522 包含一个 64×8 位的 FIFO 缓冲区。它用来缓存主机微控制器和 MFRC522 的内部状态机之间的输入和输出数据流。因此，FIFO 缓冲区可能要处理长度大于 64 字节的数据流，但又不考虑时序的限制。

9.2 访问 FIFO 缓冲区

FIFO 缓冲区的输入和输出数据总线连接到 FIFODataReg 寄存器。通过写 FIFODataReg 寄存器来将一个字节的数据存入 FIFO 缓冲区，之后内部 FIFO 缓冲区写指针加 1。读出的 FIFODataReg 寄存器的内容是存放在 FIFO 缓冲区读指针处的数据，之后 FIFO 缓冲区读指针减 1。FIFO 缓冲区的读和写指针之间的间隔通过读取 FIFOLevelReg 得到。

当微控制器发布一个命令后，MFRC522 可以在命令执行过程中根据命令要求来访问 FIFO 缓冲区。通常，只能实现一个 FIFO 缓冲区的操作，该缓冲区可用在输入和输出方向中。因此，微控制器必须小心不要以其它方式来访问 FIFO 缓冲区。

9.3 控制 FIFO 缓冲区

除了读写 FIFO 缓冲区外，FIFO 缓冲区指针还可通过置位寄存器 FIFOLevelReg 的 FlushBuffer 位来复位。从而，FIFOLevel 位被清零，寄存器 ErrorReg 的 BufferOvfl 位被清零，实际存储的字节不能再访问，FIFO 缓冲区可以用来存放下一批 64 字节的数据。

9.4 FIFO 缓冲区的状态信息

微控制器可得到以下 FIFO 缓冲区状态的数据：

- 已经存放在 FIFO 缓冲区中的字节数：寄存器 FIFOLevelReg 的 FIFOLevel 字段
- FIFO 缓冲区已满的警告：寄存器 Status1Reg 的 HiAlert 位
- FIFO 缓冲区已空的警告：寄存器 Status1Reg 的 LoAlert 位

- 指示 FIFO 缓冲区已满时仍有字节写入：寄存器 ErrorReg 的 BufferOvfl 位。BufferOvfl 位可通过置位 FIFOLevelReg 寄存器的 FlushBuffer 位来清零。

当出现以下情况时，MFRC522 可以产生中断信号：

- 如果寄存器 CommIE nReg 的 LoAlertIE n 被置位，当寄存器 Status1Reg 寄存器的 LoAlert 位变成 1 时管脚 IRQ 激活。
- 如果寄存器 CommIE nReg 的 HiAlertIE n 被置位，当寄存器 Status1Reg 寄存器的 HiAlert 位变成 1 时管脚 IRQ 激活。

如果 FIFO 缓冲区中只允许存放 WaterLevel 个（在寄存器 WaterLevelReg 中设置）或更少的字节，则 HiAlert 标志置位。它们满足下面的等式：

$$\text{HiAlert} = (64 - \text{FIFOLength}) \leq \text{WaterLevel}$$

如果实际只有 WaterLevel 个（在寄存器 WaterLevelReg 中设置）或更少的字节存放在 FIFO 缓冲区中，则 LoAlert 标志置位。它们满足下面的等式：

$$\text{LoAlert} = (64 - \text{FIFOLength}) \leq \text{WaterLevel}$$

10. 定时器组件

MFRC522 含有一个定时器组件。外部主机使用该定时器来管理与时间有关的任务。定时器可使用下面的一种配置：

- 超计数器
- 看门狗计数器
- 停止监测
- 可编程一次触发（one-shot）
- 周期性触发器

定时器可用于测量两个事件之间的时间间隔或指示某段时间后指定事件的出现。定时器可由出现的事件来触发（这将在下面进行详细描述），但它本身不会影响任何内部操作（例如，数据接收过程中的定时器超时并不会影响接收过程的自动处理）。并且，定时器还对应几个标志，这些标志可用于产生中断。

定时器有一个 6.78MHz（来自 27.12MHz 的石英晶体）的输入时钟。定时器包含 2 个阶段：1 个预分频器和 1 个计数器。

预分频器是一个 12 位的计数器。TPrescaler 的重装值在寄存器 TModeReg 和 TPrescalerReg 中定义，其值为 0 或 4095。

计数器的 16 位重装值在寄存器 TReloadReg 中定义，取值范围为 0~65535。

定时器的当前值在寄存器 TCouterValReg 中显示。

如果计数值达到 0，则中断自动产生，通过置位寄存器 CommonIRqReg 的 TimerIRq 标志来指示。如果使能，该事件还会激活 IRQ 管脚。TimerIRq 标志由主机来置位和复位。根据配置，定时器可以在计数值达到 0 时停止运行，或者，将寄存器 TReloadReg 的内容作为初始值重新启动。

定时器的状态通过寄存器 Status1Reg 的 TRunning 位来指示。

定时器的启动和停止可分别由寄存器 ControlReg 的 TStartNow 和 TstopNow 来控制。

而且，定时器还可通过置位寄存器 TModeReg 的 TAuto 位自动激活，以满足特定的协议要求。

定时器阶段的延迟时间为重装值+1。

最大时间：TPrescaler=4095，TReloadVal=65535

$$\rightarrow 4096 \times 65536 / 6,78\text{MHz} = 39,59\text{s}$$

举例：

为了得到 100us，需要计数 678 个时钟周期。这就意味着 TPrescaler 的值必须设为 TPrescaler=677。这样，定时器就有了一个 100us 的输入时钟。定时器可计数 65535 个 100us。

11. 中断请求系统

MFRC522 通过置位寄存器 Status1Reg 的 IRq 位或激活 IRQ 管脚来指示中断。IRQ 管脚的信号可用于中断微控制器使用自身的中断处理功能。允许执行有效的微控制器软件。

下表列出了中断标志、相关的中断源以及设置条件。

寄存器 CommIRqReg 的 TimerIRq 标志指示定时器中断。该标志在定时器的值从减 1 到变为 0 时设置。

如果发送器激活且其状态从发送数据变为发送帧结束，则寄存器 CommIRqReg 的 TxIRq 位置位，发送器部件自动设置中断位。

CRC 协处理器在处理完 FIFO 缓冲区的所有数据后置位寄存器 DivIRqReg 的 CRCIRq 标志。这通过设置 CRCReady=1 来指示。

寄存器 CommIRqReg 的 RxIRq 标志用来指示检测到接收数据的结束。

如果执行完一个命令且命令寄存器的内容变为空闲时寄存器 CommIRqReg 的 IdleIRq 标志置位。

如果 HiAlert 位置位，则寄存器 CommIRqReg 的 HiAlertIRq 标志置位，表明 FIFO 缓冲区已经达到 WaterLevel 位指示的长度。

如果 LoAlert 位置位，则寄存器 CommIRqReg 的 LoAlertIRq 标志置位，表明 FIFO 缓冲区已经达到 WaterLevel 位指示的长度。

寄存器 CommIRqReg 的 ErrIRq 标志指示非接触式 UART 在发送或接收过程中检测到一个错误。

表 143 中断源

中断标志	中断源	出现下列情况时中断标志自动置位
TimerIRq	定时器	定时器计数从 1 到 0
TxIRq	发送器	一次数据流发送结束
CRCIRq	CRC 协处理器	已处理完 FIFO 缓冲区的所有数据
RxIRq	接收器	一次数据流接收结束
IdleIRq	命令寄存器	执行完一个命令
HiAlertIRq	FIFO 缓冲区	FIFO 缓冲区已满
LoAlertIRq	FIFO 缓冲区	FIFO 缓冲区为空
ErrIRq	非接触式 UART	检测到一个错误

12. 振荡器电路

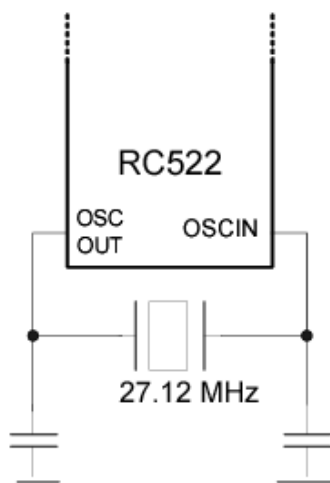


图 20 石英晶体的连接

MFRC522 的时钟可用作同步系统的编码器和译码器的时间基准。因此，时钟频率的稳定是保证系统良好性能的重要因素。为了得到最佳性能，要求时钟抖动尽可能小。最好的方法就是使用一个含有常用推荐电路的内部振荡缓冲器。如果使用外部时钟源，时钟信号就从管脚 OSCIN 输入。这时，要特别注意时钟的占空比，时钟抖动的要求以及检查时钟的质量。

13. 节电模式

13.1 硬掉电

硬掉电通过 NRSTPD 管脚的低电平来使能。硬掉电模式下，包括振荡器在内的所有消耗能量的内部部件全部关闭。所有数字输入缓冲器和输入端分离，只能在内部定义（NRSTPD 管脚本身例外）。输出管脚冻结在某个特定值。

13.2 软掉电

寄存器 CommandReg 的 PowerDown 位置位后立刻进入软掉电模式。所有内部消耗电流的部件都关闭（包括振荡缓冲器在内）。

与硬掉电模式不同，数字输入缓冲器不和输入端分离，功能保持不变。数字输出管脚的状态不变。

所有寄存器的值、FIFO 的内容和配置信息在软掉电模式中都保持不变。

当寄存器 CommandReg 的 PowerDown 被设置成 0 后，还需要 1024 个时钟周期才能退出软掉电模式，由 PowerDown 位来指示。PowerDown 位设置成 0 并不能立刻将其清除。它在退出软掉电模式时自动由 MFRC522 清除。

注：如果使用了内部振荡器，必须考虑到它是由 AVDD 供电的，必须经过 t_{osc} 时间后，振荡器才能稳定运行，内部逻辑才能检测到时钟周期。

注：如果使用了串行 UART 接口，则软掉电模式通过向 MFRC522 发送值 55 (hex) 来复位。为了对寄存器进行进一步的访问，振荡器必须稳定运行。首次读或写访问的必须

是地址 0。

对于串行 UART，建议先发送值 55 (hex)，直到 MFRC522 回应上一个寄存器内容为地址 0 的读命令时才执行对地址 0 的读访问。这样来指示 MFRC522 可以执行进一步的操作。

13.3 发送器掉电

发送器掉电模式切断内部天线驱动器来关闭 RF 场，通过清零寄存器 TXControlReg 的 TX1RfEn 或 TX2RfEn 来实现。

14. 复位和振荡器启动时间

14.1 复位时序要求

复位信号必须经过一个滞后电路和窄带滤波器（抑制小于 10ns 的信号）再进入数字电路。为了实现复位，信号必须至少为 100ns。

14.2 振荡器启动时间

MFRC522 处于低功耗模式或使用 XVDD 对 IC 供电时的振荡器启动时序如下图所示。

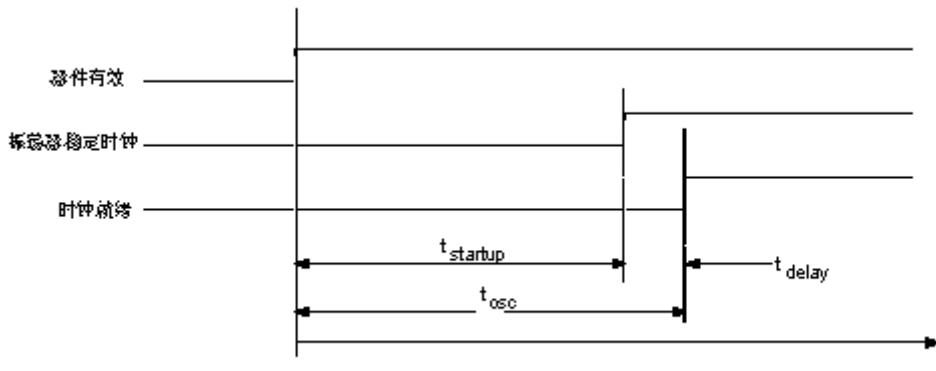


图 21 振荡器启动时间

时间 $t_{startup}$ 定义了晶体振荡器电路的启动时间。该时间由晶体本身来定义。

t_{delay} 定义了时钟信号稳定时 MFRC522 的内部延迟时间，该延迟时间后 MFRC522 才能被寻址。这个时间延时可通过下式计算出来： $t_{delay}[us] = 1024/27.12 = 37.76us$ 。

t_{osc} 定义成时间 t_{delay} 和 $t_{startup}$ 之和。

15. MFRC522 命令集

15.1 概述

MFRC522 的操作由可执行一系列命令的内部状态机来决定。通过向命令寄存器写入相应的命令代码来启动命令。

执行一个命令所需的参数和/或数据通过 FIFO 缓冲区来交换。

15.2 通用特性

- 每个需要数据流（或数据字节流）作为输入的命令在发现 FIFO 缓冲区有数据时会立刻处理，但收发命令除外。收发命令的发送由寄存器 BitFramingReg 的 StartSend 位来启动。
- 每个需要某一数量的参数的命令只有在它通过 FIFO 缓冲区接收到正确数量的参数时才能开始处理。
- FIFO 缓冲区不能在命令启动时自动清除。而且，也有可能要先将命令参数和/或数据字节写入 FIFO 缓冲区，再启动命令。
- 每个命令的执行都可能由微控制器向命令寄存器写入一个新的命令代码（如 idle 命令）来中断。

15.3 MFRC522 命令概述

表 144 命令概述

命令	命令代码	动作
Idle	0000	无动作；取消当前命令的执行。
CalcCRC	0011	激活 CRC 协处理器或执行自测试。
Transmit	0100	发送 FIFO 缓冲区的命令。
NoCmd Change	0111	无命令改变。该命令用来修改命令寄存器的不同位，但又不触及其它命令，如掉电。
Receive	1000	激活接收器电路。
Transceive	1100	如果寄存器 ControlReg 的 Initiator 位被设为 1： 将 FIFO 缓冲区的数据发送到天线并在发送完成后自动激活接收器。 如果寄存器 ControlReg 的 Initiator 位被设为 0： 接收天线的数据并自动激活发送器。
MFAuthent	1110	执行读卡器的 MIFARE 标准认证。
Soft Reset	1111	复位 MFRC522。

15.4 MFRC522 命令描述

15.4.1 IDLE 命令

MFRC522 处于空闲模式。该命令也用来终止实际正在执行的命令。

15.4.2 CALC CRC 命令

FIFO 的内容被传输到 CRC 协处理器并执行 CRC 计算。计算结果存放在 CRCResultReg 寄存器中。CRC 计算无需限制字节的数目。当在数据流过程中 FIFO 变成空时计算也不会停止。写入 FIFO 的下一个字节增加到计算中去。

CRC 的预置值由寄存器 ModeReg 的 CRCPreset 位控制，该值在命令启动时装入 CRC 协处理器。

这个命令必须通过向命令寄存器写入任何一个命令（如空闲命令）来软件清除。如果寄存器 AutoTestReg 的 SelfTest 位设置正确，则 MFRC522 处于自测试模式，启动 CalcCRC 命令执行一次数字自测试。自测试的结果写入 FIFO。

15.4.3 TRANSMIT 命令

发送 FIFO 的内容。在发送 FIFO 的内容之前必须对所有相关的寄存器进行设置。

该命令在 FIFO 变成空后自动终止。

15.4.4 NoCMDCHANGE 命令

该命令不会影响 CommandReg 寄存器中正在执行的任何命令。它可用来修改 CommandReg 寄存器中除命令位之外的任何位，如 RcvOff 位或 PowerDown 位。

15.4.5 RECEIVE 命令

MFRC522 激活接收器通路，等待接收任何数据流。

该命令在接收到的数据流结束时自动终止。根据所选的成帧和速度，通过帧模式结束或长度字节来指示。

15.4.6 TRANSCEIVE 命令

该循环命令重复发送 FIFO 的数据，并不断接收 RF 场的的数据。第一个动作是发送，发送结束后命令变为接收数据流。

发送

接收

发送

接收...

每个发送过程都在 BitFramingReg 寄存器的 StartSend 位置位时启动。TRANSCEIVE 命令通过向命令寄存器写入任何一个命令（如 idle 命令）来软件清除。

15.4.7 MFAUTHENT 命令

该命令用来处理 Mifare 认证以使能到任何 Mifare 普通卡的安全通信。在命令激活前以下数据必须被写入 FIFO:

- 认证命令代码 (0x60, 0x61)
- 块地址
- 扇区密钥字节 0
- 扇区密钥字节 1
- 扇区密钥字节 2
- 扇区密钥字节 3
- 扇区密钥字节 4
- 扇区密钥字节 5
- 卡序列号字节 0
- 卡序列号字节 1
- 卡序列号字节 2
- 卡序列号字节 3

总共 12 字节，应当写入 FIFO 中。

注：当 MFAuthent 命令有效时，任何 FIFO 访问都被禁止。只要访问 FIFO 的操作发生，ErrorReg 寄存器的 WrErr 位就置位。

该命令在 Mifare 卡被认证且 Status2Reg 寄存器的 MFCrypto1On 位置位时自动终止。

当卡未响应时该命令不会自动终止，因此，定时器必须初始化成自动模式。这时，除 IdleIRQ 外，TimerIRQ 也可用作终止的标准。在认证过程中，RxIRQ 和 TxIRQ 被禁止。认证命令结束后（处理完协议后或将 IDLE 写入命令寄存器后）只有 Crypto1On 位有效。

如果认证过程中有错误出现，则 ErrorReg 寄存器的 ProtocolErr 位置位。Status2Reg 寄存器的 Crypto1On 位清零。

15.4.8 SOFTRESET 命令

该命令用来执行一次器件复位。内部缓冲区的配置数据保持不变。

所有寄存器都设置成复位值。命令完成后自动终止。

注：由于 SerialSpeedReg 寄存器被复位，串行数据速率被设置成 9600kbps。

16. 测试信号

16.1 测试总线

测试总线用来执行器件测试。下面的配置可用来改善包含 MFRC522 的系统的设计。测试总线允许使内部信号通向数字接口。测试总线信号通过访问 TestSel2Reg (0x32) 寄存器的 TestBusSel 来选择。

表 145 TestSel2Reg 设置成 0x07

管脚	D6	D5	D4	D3	D2
测试信号	sdata	scoll	svalid	sover	RCV_reset

表 146 测试信号描述

测试信号	描述
sdata	指示实际接收到的数据值。
scoll	指示在实际的位中是否检测到冲突（仅适用于 106kbit/s 的传输）。
svalid	指示 sdata 和 scoll 是否有效。
sover	指示接收器已经检测到一个停止位。
RCV_reset	指示接收器是否复位。

表 147 TestSel2Reg 设置成 0x0D

管脚	D6	D5	D4	D3	D2
测试信号	clkstable	clk27/8	RFU	RFU	clk27

表 148 测试信号描述

测试信号	描述
clkstable	指示振荡器是否产生一个稳定信号。
clk27/8	指示振荡器的输出信号 8 分频
clk27	指示振荡器的输出信号。

16.2 管脚 AUX 的测试信号

表 149 测试信号描述

SELAUX	AUX1/AUX2 的描述
0000	三态
0001	DAC: 寄存器 TestDAC 1/2
0010	DAC: 测试信号 corr1
0011	RFU
0100	DAC: 测试信号 MinLevel
0101	DAC: ADC_I
0110	DAC: ADC_Q
0111	DAC: 测试信号 ADC_I 和 ADC_Q
1001	RFU
1010	高
1011	低
1100	TxActive
1101	RxActive
1111	TstBusBit

通过设置 AnalogTestReg 的 SelAux1 或 SelAux2, 每个信号都可切换到 AUX1 或 AUX2。

注: DAC 有一个电流输出, 建议 AUX1/AUX2 管脚使用一个 1kΩ 的下拉电阻。

16.3 PRBS

根据 ITU-TO150 来使能 PRBS9 或 PRBS15。为了启动定义的数据流的发送, 必须激活命令发送。前导/同步字节/起始位/奇偶位自动产生, 由选择的模式决定。

注: 在进入 PRBS 模式前必须先配置好所有与发送数据相关的寄存器 (遵循 ITU-TO150)。

17. 典型应用

下图所示是一个典型电路图, 图中使用一个互补天线连接到 MFRC522。

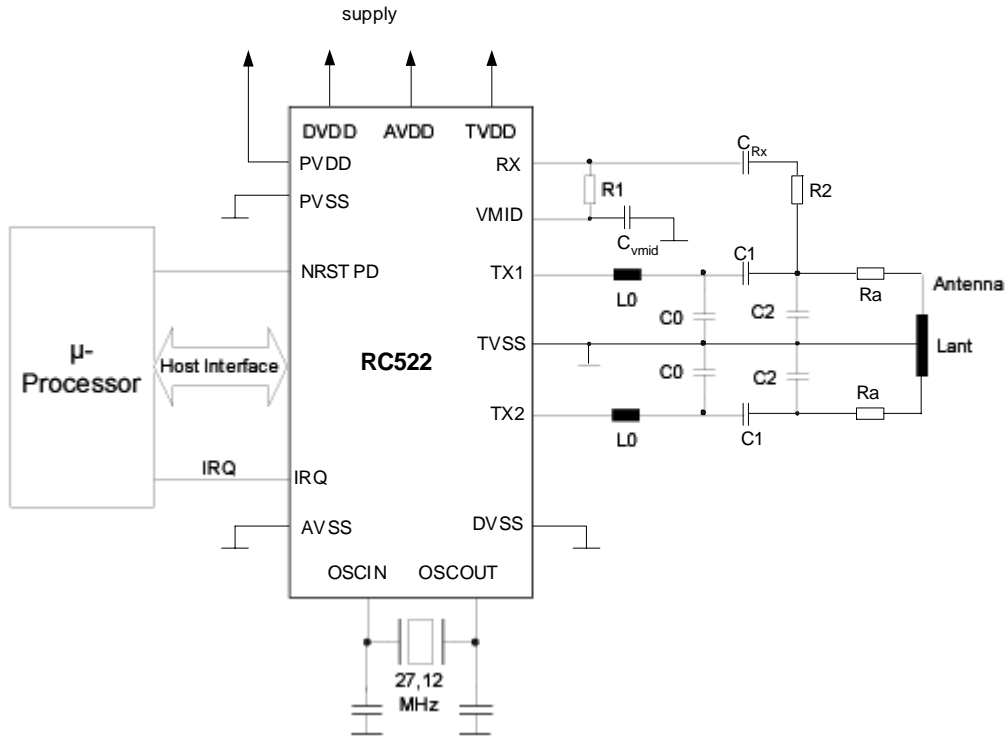


图 22 典型电路图

18. 电气特性

18.1 绝对最大额定值

表 150 绝对最大额定值

符号	参数	最小	最大	单位
AVDD, DVDD, PVDD, TVDD	电源电压	-0.5	4.0	V

18.2 极限值

表 151 极限值

符号	参数	最小	最大	单位
P _{tot}	总功耗		200	mW
T _j	结点温度		100	°C

18.3 ESD 特性

表 152 ESD 特性

符号	参数	最小	最大	单位
ESDH	ESD 敏感度 (人体模式)	1500Ω, 100pF	JESD22- A114-B	2000V
ESDM	ESD 敏感度 (机器模式)	0.75uH, 200pF	JESD22- A114-A	200V
ESDC	ESD 敏感度 (充电设备模式)	场感应 模式	JESC22- C101-A	1000V

18.4 温度特性

表 153 温度特性

符号	参数	条件	封装	值	单位
Rthj-a	结点到周围环境的热 电阻	暴露在静止空气中的四层 PCB 板的焊盘	Jedec HVQFN32	40	k/W

18.5 工作条件范围

表 154 工作条件范围

符号	参数	条件	最小	典型	最大	单位
Tamb	周围环境温度	HVQFN32	-30		+85	°C
AVDD, DVDD,TVDD	电源电压	AVSS=DVSS=PVSS=TVSS=0V, PVDD<=AVDD=DVDD=TVDD	2.5	3.3	3.6	V
PVDD	电源电压	AVSS=DVSS=PVSS=TVSS=0V, PVDD<=AVDD=DVDD=TVDD	1.6	1.8	3.6	V

注:

1. 电源电压在 3V 以下会降低器件的性能 (如, 可实现的操作距离)
2. AVDD, DVDD 和 TVDD 应当总是保持相同的电压。
3. PVDD 应当总是等于或低于 DVDD。

18.6 输入/输出管脚特性

18.6.1 EA, 12C, SIGIN 和 NRESET 输入管脚特性

表 155 EA, 12C, SIGIN 和 NRESET 输入管脚特性

符号	参数	条件	最小	典型	最大	单位
I _{LEAK}	输入漏电流		-1		1	uA
V _{IH}	输入电压高电平		0.7PVDD			V
V _{IL}	输入电压低电平				0.3PVDD	V

18.6.2 D1, D2, D3, D4, D5, D6 和 D7 输入/输出管脚特性

表 156 D1, D2, D3, D4, D5, D6 和 D7 输入/输出管脚特性

符号	参数	条件	最小	典型	最大	单位
I_{LEAK}	输入漏电流		-1		1	uA
V_{IH}	输入电压高电平		0.7PVDD			V
V_{IL}	输入电压低电平				0.3PVDD	V
V_{OH}	输出电压高电平	PVDD=3V, $I_O=4mA$	PVDD-300mV		PVDD	V
V_{OL}	输出电压低电平	PVDD=3V, $I_O=4mA$	PVSS		PVSS+300mV	V
I_{OL}	输出电流驱动低电平	PVDD=3V			4	mA
I_{OH}	输出电流驱动高电平	PVDD=3V			4	mA

18.6.3 SDA 输入/输出管脚特性

表 157 SDA 输入/输出管脚特性

符号	参数	条件	最小	典型	最大	单位
I_{LEAK}	输入漏电流		-1		1	uA
V_{IH}	输入电压高电平		0.7PVDD			V
V_{IL}	输入电压低电平				0.3PVDD	V
V_{OL}	输出电压低电平	PVDD=3V, $I_O=3mA$	-	-	PVSS+400mV	V
I_{OL}	输出电流驱动低电平	PVDD=3V	-	-	4	mA

18.6.4 SIGOUT 输出管脚特性

表 158 SIGOUT 输出管脚特性

符号	参数	条件	最小	典型	最大	单位
V_{OH}	输出电压高电平	PVDD=3V, $I_O=4mA$	PVDD+300mV		PVDD	V
V_{OL}	输出电压低电平	PVDD=3V, $I_O=4mA$	PVSS		PVSS+300mV	V
I_{OL}	输出电流驱动低电平	PVDD=3V			4	mA
I_{OH}	输出电流驱动高电平	PVDD=3V			4	mA

18.6.5 IRQ 输出管脚特性

表 159 IRQ 输出管脚特性

符号	参数	条件	最小	典型	最大	单位
V_{OH}	输出电压高电平	PVDD=3V, $I_O=4mA$	PVDD-300mV		PVDD	V
V_{OL}	输出电压低电平	PVDD=3V, $I_O=4mA$	PVSS		PVSS+300mV	V
I_{OL}	输出电流驱动低电平	PVDD=3V			4	mA
I_{OH}	输出电流驱动高电平	PVDD=3V			4	mA

18.6.6 Rx 输入管脚特性

表 160 Rx 输入管脚特性

符号	参数	条件	最小	典型	最大	单位
$V_{IN,RX}$	输入电压范围		-1	-	AVDD+1V	V
$C_{IN,RX}$	RX 输入电容	AVDD=3V, 接收器有效, $V_{RX}=1V_{pp}$, $1.5V_{DC}$ 的偏移	-	10	-	pF
$R_{IN,RX}$	RX 输入串联电阻	AVDD=3V, 接收器有效, $V_{RX}=1V_{pp}$, $1.5V_{DC}$ 的偏移	-	350	-	Ohm

注：RX 的电压被内部二极管箝位到 AVSS 和 AVDD。

18.6.7 OSCIN 输入管脚特性

表 161 外部时钟的 OSCIN 输入管脚特性

符号	参数	条件	最小	典型	最大	单位
I_{LEAK}	输入漏电流		-1	-	1	uA
V_{IH}	输入电压高电平		0.7AVDD	-	-	V
V_{IL}	输入电压低电平		-	-	0.3AVDD	V
C_{OSCIN}	输入电容	AVDD=2.8V, $V_{DC}=0.65V$, $V_{AC}=1V_{pp}$	-	2	-	pF

18.6.8 AUX1 和 AUX2 输出管脚特性

表 162 AUX1 和 AUX2 输出管脚特性

符号	参数	条件	最小	典型	最大	单位
V_{OH}	输出电压高电平	DVDD=3V, $I_O=4mA$	DVDD-300mV	-	DVDD	V
V_{OL}	输出电压低电平	DVDD=3V, $I_O=4mA$	DVSS	-	DVSS+300mV	V
I_{OL}	输出电流驱动低电平	DVDD=3V	-	-	4	mA
I_{OH}	输出电流驱动高电平	DVDD=3V	-	-	4	mA

18.6.9 TX1 和 TX2 输出管脚特性

表 163 TX1 和 TX2 输出管脚特性

符号	参数	条件	最小	典型	最大	单位
$V_{OH,C32,3V}$	输出电压高电平	TVDD=3V 和 $I_{TX}=32mA$, CWGsP=3F(hex)	TVDD -150mV	-	-	mV
$V_{OH,C80,3V}$	输出电压高电平	TVDD=3V 和 $I_{TX}=80mA$, CWGsP=3F(hex)	TVDD -400mV	-	-	mV
$V_{OH,C32,2V5}$	输出电压高电平	TVDD=2.5V 和 $I_{TX}=32mA$, CWGsP=3F(hex)	TVDD -240mV	-	-	mV
$V_{OH,C80,2V5}$	输出电压高电平	TVDD=2.5V 和 $I_{TX}=80mA$, CWGsP=3F(hex)	TVDD -640mV	-	-	mV
$V_{OL,C32,3V}$	输出电压低电平	TVDD=3V 和 $I_{TX}=32mA$, CWGsN=F(hex)	-	-	150	mV

续表 163

符号	参数	条件	最小	典型	最大	单位
$V_{OL, C80, 3V}$	输出电压低电平	TVDD=3V 和 $I_{TX}=80mA, CWGsN=F(hex)$	-	-	400	mV
$V_{OL, C32, V5}$	输出电压低电平	TVDD=2.5V 和 $I_{TX}=32mA, CWGsN=F(hex)$	-	-	240	mV
$V_{OL, C80, V5}$	输出电压低电平	TVDD=2.5V 和 $I_{TX}=80mA, CWGsN=F(hex)$	-	-	640	mV

18.7 电流消耗

表 164 电流消耗

符号	参数	条件	最小	典型	最大	单位
I_{HPD}^4	硬掉电电流	AVDD=DVDD=TVDD=PVDD=3V, NRESET=LOW	-	-	5	uA
I_{SPD}^4	软掉电电流	AVDD=DVDD=TVDD=PVDD=3V, RF 电平检测器导通	-	-	10	uA
I_{DVDD}	数字电源电流	DVDD=3V	-	6,5	9	mA
I_{AVDD}	模拟电源电流	AVDD=3V, 位 <i>RCVOff</i> =0	-	7	10	mA
$I_{AVDD, RCVOff}$	模拟电源电流, 接收器关闭	AVDD=3V, 位 <i>RCVOff</i> =1	-	3	5	mA
I_{PVDD}^2	管脚电源电流		-	-	40	mA
$I_{TVDD}^{1,3}$	发送器电源电流	连续波	-	60 ⁵	100	mA

注:

1. I_{TVDD} 取决于 TVDD 和连接到 Tx1 和 Tx2 的外部电路。
2. I_{PVDD} 取决于数字管脚的总负载。
3. 典型电路工作期间消耗的总电流在 100mA 以下。
4. I_{SPD} 和 I_{HPD} 是所有电源的总电流。
5. 使用互补驱动器配置和在 13.56MHz 的频率下匹配到 TX1 和 TX2 之间的 40Ω 的天线时的典型值。

18.8 RX 输入电压范围

表 165 RX 输入电压范围

符号	参数	条件	最小	典型	最大	单位
$V_{RX, MinIV, Mill}$	最小输入电压, Miller 编码	AVDD=3V, 106kbit/s	-	150	-	mVpp
$V_{RX, MinIV, Man}$	最小输入电压, Manchester 编码	AVDD=3V, 212 和 424kbit/s	-	100	-	mVpp
$V_{RX, MaxIV, Mill}$	最大输入电压, Miller 编码	AVDD=3V, 106kbit/s	-	4	-	Vpp
$V_{RX, MaxIV, Man}$	最大输入电压, Manchester 编码	AVDD=3V, 212 和 424kbit/s	-	4	-	Vpp

18.9 RX 输入灵敏度

表 166 RX 输入灵敏度

符号	参数	条件	最小	典型	最大	单位
$m_{RX, Mill}$	最小调制系数， Miller 编码	AVDD=3V, 106kbit/s $V_{RX}=1.5V_{pp}$, SensMiller=3	-	33	-	%
$V_{RXMod, Man}^1$	最小调制电压	AVDD=3V, RxGain=7	-	5	-	mV

注 1: 除了 Miller 编码的信号外，最小调制电压对于所有调制机制都有效。

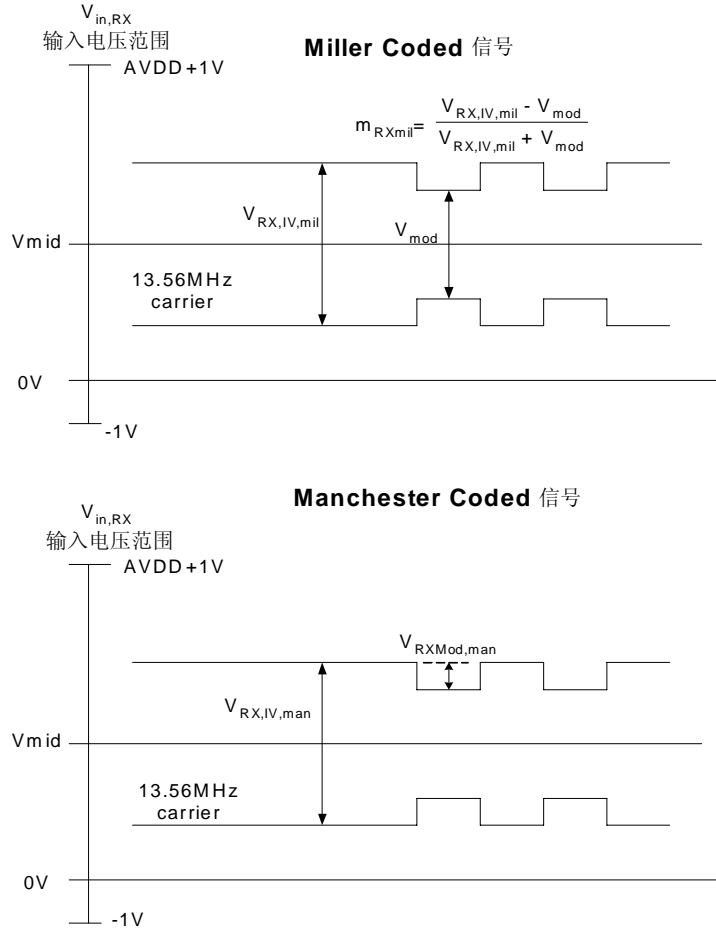


图 23 RX 输入电压范围

18.10 时钟频率

表 167 时钟频率

符号	参数	最小	典型	最大	单位
f_{OSCIN}	时钟频率		27.12		MHz
d_{FEC}	时钟频率的占空比	40	50	60	%
t_{jitter}	时钟边沿的抖动			10	Ps, RMS

18.11 XTAL 振荡器

表 168 XTAL 振荡器

符号	参数	最小	典型	最大	单位
f_{XTAL}	XTAL 频率	27.106	27.12	27.134	MHz
R_{XTAL}	XTAL 串联电阻			tbd	OHM
$C_{L, XTAL}$	XTAL 负载电容			tbd	pF

18.12 典型 27.12MHz 晶体的要求

表 169 XTAL 振荡器

符号	参数	最小	典型	最大	单位
f_{XTAL}	XTAL 频率范围	-	27.12	-	MHz
ESR	XTAL 等效串联电阻	-	-	100	OhM
C_L	XTAL 负载电容	-	10	-	pF
P_{XTAL}	XTAL 驱动电平	-	50	100	uW

18.13 SPI 兼容接口的时序

表 170 SPI 的时序规范

符号	参数	最小	最大	单位
t_{SCKL}	SCK 低电平脉宽	50	-	ns
t_{SCKH}	SCK 高电平脉宽	50	-	ns
t_{SHDX}	SCK 高电平到数据变化	25	-	ns
t_{DXSH}	数据变化到 SCK 高电平	25	-	ns
t_{SLDX}	SCK 低电平到数据变化	-	25	ns
t_{SLNH}	SCK 低电平到 NSS 高电平	0	-	ns

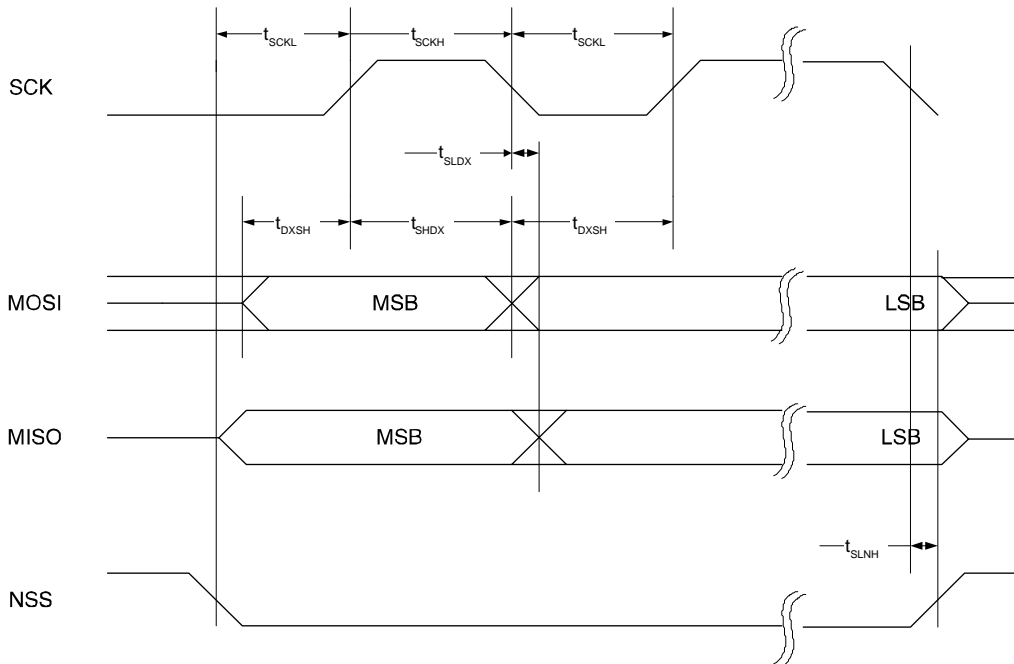


图 24 SPI 的时序图

注：信号 NSS 必须为低以便可在一个数据流中发送多个字节。
 为了发送多个的数据流，NSS 必须在数据流之间设置成高电平。

18.14 I²C 时序

表 171 快速模式下的 I²C 时序

符号	参数	快速模式		高速模式		单位
		最小	最大	最小	最大	
f_{SCL}	SCL 时钟频率	0	400	0	3400	kHz
$t_{HD; STA}$	保持时间（重复的）起始条件。这个周期之后产生第一个时钟脉冲。	600	-	160	-	ns
$t_{SU; STA}$	重复起始条件的建立时间	600	-	160	-	ns
$t_{SU; STO}$	停止条件的建立时间	600	-	160	-	ns
t_{LOW}	SCL 时钟的低电平周期	1300	-	160	-	ns
t_{HIGH}	SCL 时钟的高电平周期	600	-	60	-	ns
$t_{HD; DAT}$	数据保持时间	0	900	0	70	ns
$t_{SU; DAT}$	数据建立时间	100	-	10	-	ns
t_{rsc1}	SCL 信号的上升时间	20	300	10	40	ns
t_{fsc}^1	SCL 信号的下降时间	20	300	10	40	ns
t_{rsda}	SDA 和 SCL 信号的上升时间	20	300	10	80	ns
T_{fsda}	SDA 和 SCL 信号的下降时间	20	300	10	80	ns
t_{BUF}	停止和起始条件之间的总线空闲时间	1.3	-	1.3	-	us

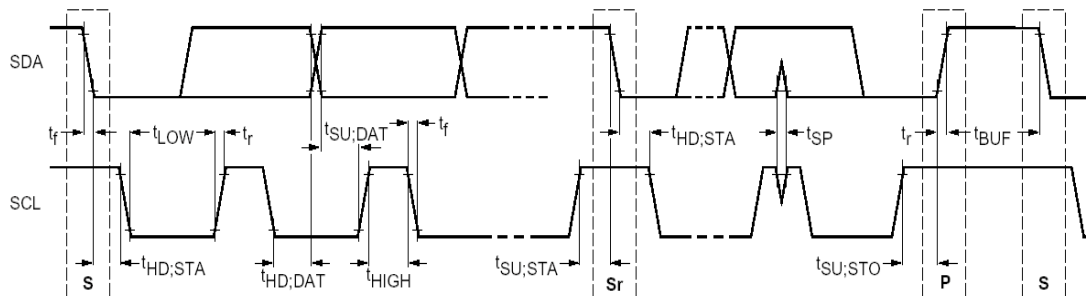
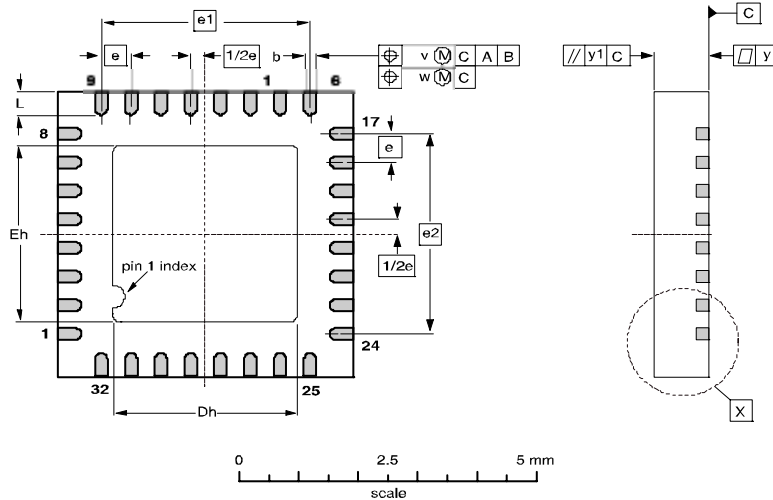
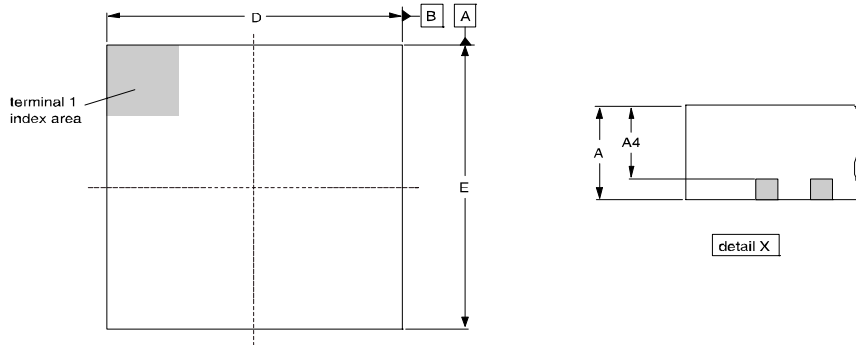


图 25 I²C 总线上 F/S 模式器件的时序

19.封装

HVQFN32



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A4 max.	b	D(1)	Dh	E(1)	Eh	e	e1	e2	L	v	w	y	y1
mm	1.00	0.80	0.35 0.18	5.05 4.95	3.25 2.95	5.05 4.95	3.25 2.95	0.5	3.5	3.5	0.50 0.30	0.2	0.1	0.05	0.1

Note

1. Plastic or metal protrusions of 0.076 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT617-1		MO-220				-01-06-07- 01-08-08

附录A 版本信息

修订版本	修订日期	描述
Rev 1.0	2007 年 9 月	原始版本