

应用资料 2

JF24C工作程序指南

应用资料2 详细描述了JF24C模块芯片MCU的工作程序及工作流程示意图，SPI协议时序图，寄存器设置及各种数据。

应用资料1 详细介绍了JF24C模块的性能与单片机的接口电路及应用指南。

应用资料3 详细介绍了JF24C模块与单片机应用编程指南供参考。

目前2.4G 产品应用比较广泛，有些芯片性能也很不错，但价位都比较偏高，很难进入量产的产品。为降低成本JF24C模块采用裸片绑定，虽然性能指标略低于目前具有代表性的 nRF2401 CC2500 A7105但它的价格要比它们低很多，完全可以满足一般需要双向数据传输及双向遥控的短距离产品应用。

单发单收的产品使用比较简单，加电加信号就发射，收到信号就有输出，纯硬件产品单向传输，不需要软件程序的支持就可以完成收发功能。2.4G产品就比较复杂化了，芯片内有CPU需要软件程序的支持，必须要有单片机的指令才可以完成双向收发功能。单发单收的产品成本低廉应用广泛，但存在着严重的无法避免的同频干扰，2.4G产品具有跳频功能一般都有几十至100多个通道可以避开干扰。但2.4G产品复杂的软件程序也使一些不懂单片机的工程师望而怯步，同时2.4G产品的功耗及成本还有对墙体的穿透性能下降也影响到在低端产品的普及应用。

数据包和FIFO指针定义



1 preamble: 引导码 1 ~ 8个字节 (JF24C模块内部已设置, 用户不可以写入)

2 SYNC: 同步头 32/48/64位地址 (用户可编程写入模块内部寄存器作为地址识别)

当接收模式已收到SYNC 自动设定FIFO写指针= 0
或者发射模式传送过SYNC 自动设定FIFO读指针= 0

3 trailer: 4-18位 (用户可编程, 填充数据包)

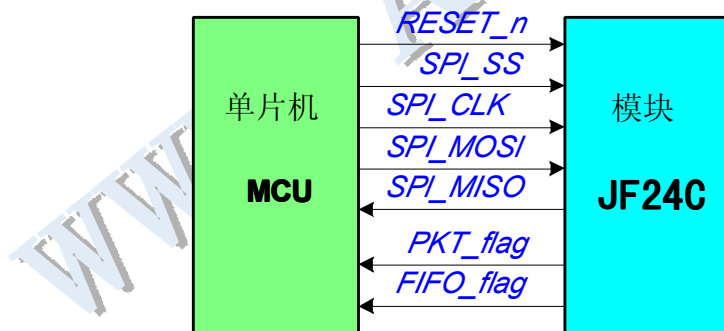
4 payload: 数据, 用户可编程 (RX的数据, 有4个类型: 原始8-10位, 曼彻斯特, 交错, 用FEC)

5 CRC: 16位校验码, 用户不能写入 (模块内部已设置, 发送时自动加入, 接收后自动移除)

注: 对于传送, 在应用中写入数据前必须通过获得reg82(15)的内容明确FIFO的写指针。

数字接口

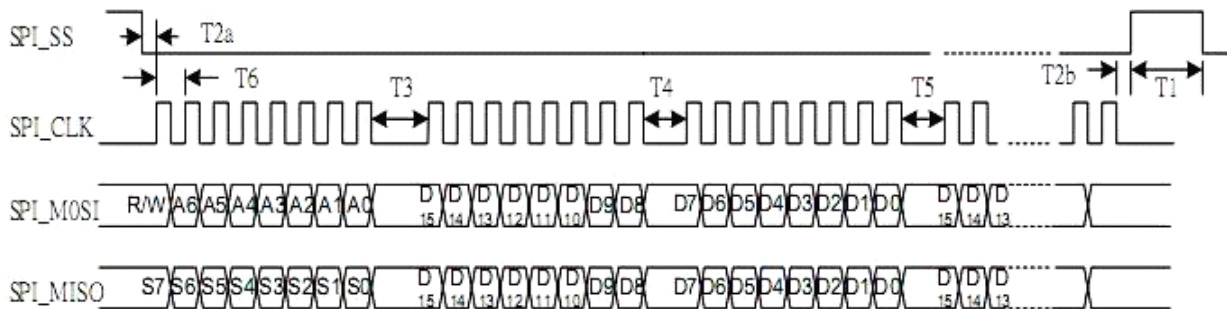
JF24C 采用 SPI 数字接口与单片机连接。它支持两种 SPI 标准格式(CKPHA = 0 CKPHA = 1)
请参考 CKPHA 的数据表以及相关的数据格式。



SPI_CLK	SPI时钟的输入
SPI_SS	SPI从机选择 (输入)
SPI_MOSI	SPI数据输入
SPI_MISO	SPI数据输出
PKT_FLAG	TX / RX标志
FIFO_FLAG	FIFO空/满/标志
RESET_n	复位

SPI时序图

这里是描述 SPI 信号的时序图。以 CKPHA = 0 为例说明（**SPI_MOSI** 数据输入在时钟周期的上升沿有效）



name	Min.	Typ.	Max.	Description
T1	250ns			两次访问SPI的间隔
T2a, T2b	41.5ns			SPI-SS和SPI-CLK之间的关系
T3	* (1)			数据和地址之间的间隔
T4	* (2)			数据高位与低位间的间隔
T5	* (3)			寄存器中两端数据之间的间隔
T6	83ns			SPI_CLK 周期

*注释:

(1) MCU单片机读取寄存器0x00 ~ 0x1f, 至少等待3us (微处理器地址解码)

当MCU读取FIFO数据时,至少等待450ns (微处理器获得正确的FIFO读指针)
T3min = 41.5ns。

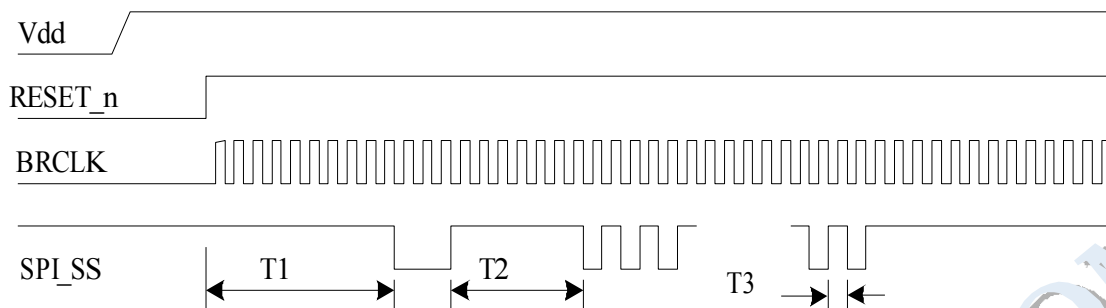
(2)当MCU读取FIFO数据时,至少等待450ns (微处理器获得正确的读指针)

T4min = 41.5ns

(3) MCU写寄存器0x00 ~ 0x1f,至少要等待3us (微处理器编程序寄存器)
如果读FIFO数据,至少450ns等待的时间。T5min = 41.5ns

上电和寄存器初始化序列

下图为上电时序图



接通电源和寄存器序列

1. VDD 上电后, 确认复位引脚在低电平时复位。在置 RESET-n 为 1 后, 逻辑时钟运行在 12MHZ。
 2. 等待 T1(1 ~ 5ms)晶振稳定,单片机使微处理寄存器 (reg48-reg57) 初始化。初始化数据可以被连续写入也可一位一位被写入。
 3. 微处理寄存器初始化后,等待 RFIC 自动校准 T2(2ms),然后进行 RFIC 初始化。对于 RFIC 寄存器初始化, 建议数据间隔 T3 为 3 μ S 一位一位写入。
 4. RFIC 寄存器初始化后, 需要等待 2ms, 使 JF24C 准备好传送或接收。
- 注: 微处理寄存器(reg48 ~ reg57)初始化要在 RFIC 寄存器初始化之前。

进入休眠和唤醒

当单片机通过 SPI-MOSI 给 JF24C 寄存器写 1 同时把 SPI_SS 拉高, JF24C 将进入睡眠状态, 此时电流消耗极低 3.5uA。当 SPI_SS 被拉低时, JF24C 会从睡眠状态自动唤醒。

开始TX / RX和停止TX / RX定义

JF24C 提供了两种方法来处理 TX / RX 数据包的长度。一种是种在 JF24C 微处理器内自动检测数据包 长度。另一种是微处理器保持接收/发射状态直到单片机终止发送和接受。JF24C 微处理器可处理的最高数据包长度为 255 个字节, 如果 RX/TX 数据包长度小于 255 个字节, 我们建议使用 JF24C 微处理器来检测数据包长度。这个选项需要通过配置寄存器的 57[13]。

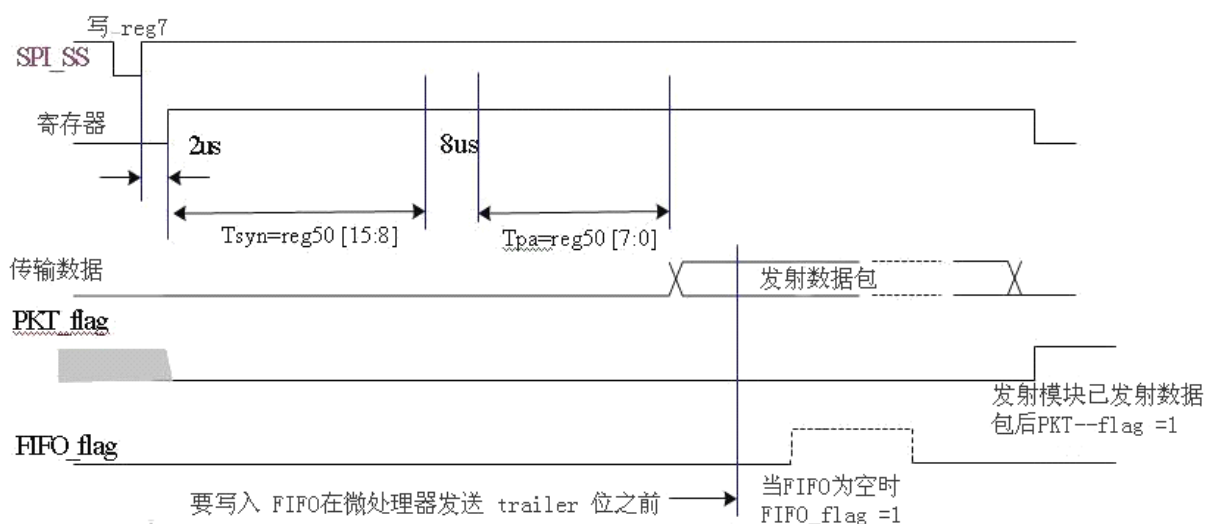
详细的信息,请参阅 JF24C 寄存器的定义。时序图显示细节如下。所有的时序图设置 PKT-flag 和 FIFO-flag 为高电平有效、也可通过寄存器设置为低电平有效。

TX/RX时序图——（JF24C微处理器自动检测数据包的长度）

微处理器处理数据包长度，通过设置Reg57[13]= 1。数据包的第一个字节为数据包长度（这个字节的长度不计入数据包长度）。最大允许包的长度是255字节。微处理器处理TX / RX启动和停止。

TX的时序图

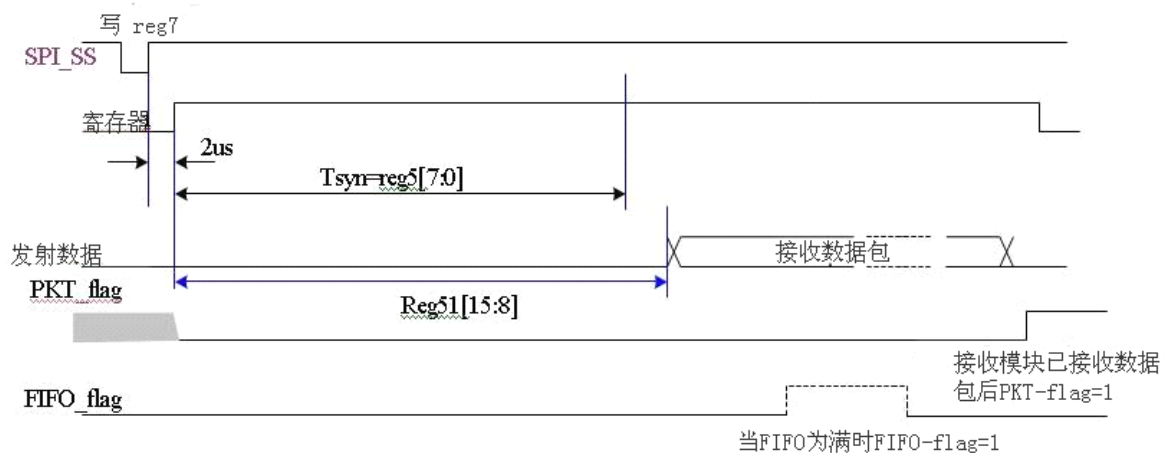
下图 在单片机写Reg7 [8]= 1并且选用传输通道(参阅Reg7定义)后，微处理器会从FIFO中自动传输同步头和数据。单片机需要在微处理器发送trailer位之前填写传输数据。如果数据包的长度超过63个字节，单片机需要多次将数据写入FIFO。FIFO-flag 表明在发射状态FIFO是否为空。



TX时序图 PKT_flag和FIFO_flag 的高电平有效
(微处理器处理数据包长度)

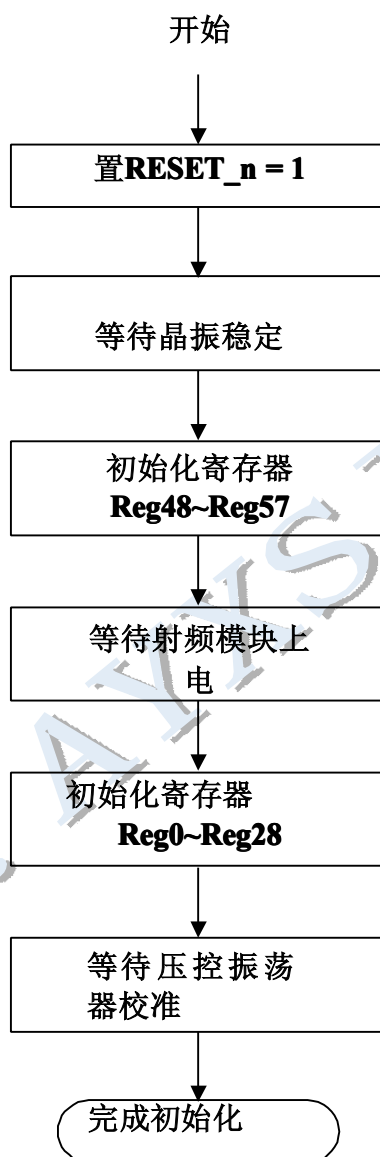
RX时序图

下图 当单片机写Reg7[7]= 1并且选择了接收通道，微处理器将自动检测有效同步头，当它发现有效同步头，微处理器将自动处理包。当接收数据包过程已经完成，微处理器会设置射频进入空闲状态。如果收到包长度超过63个字节，在接收状态FIFO-flag将显示FIFO为满，单片机需要从FIFO中读出数据。

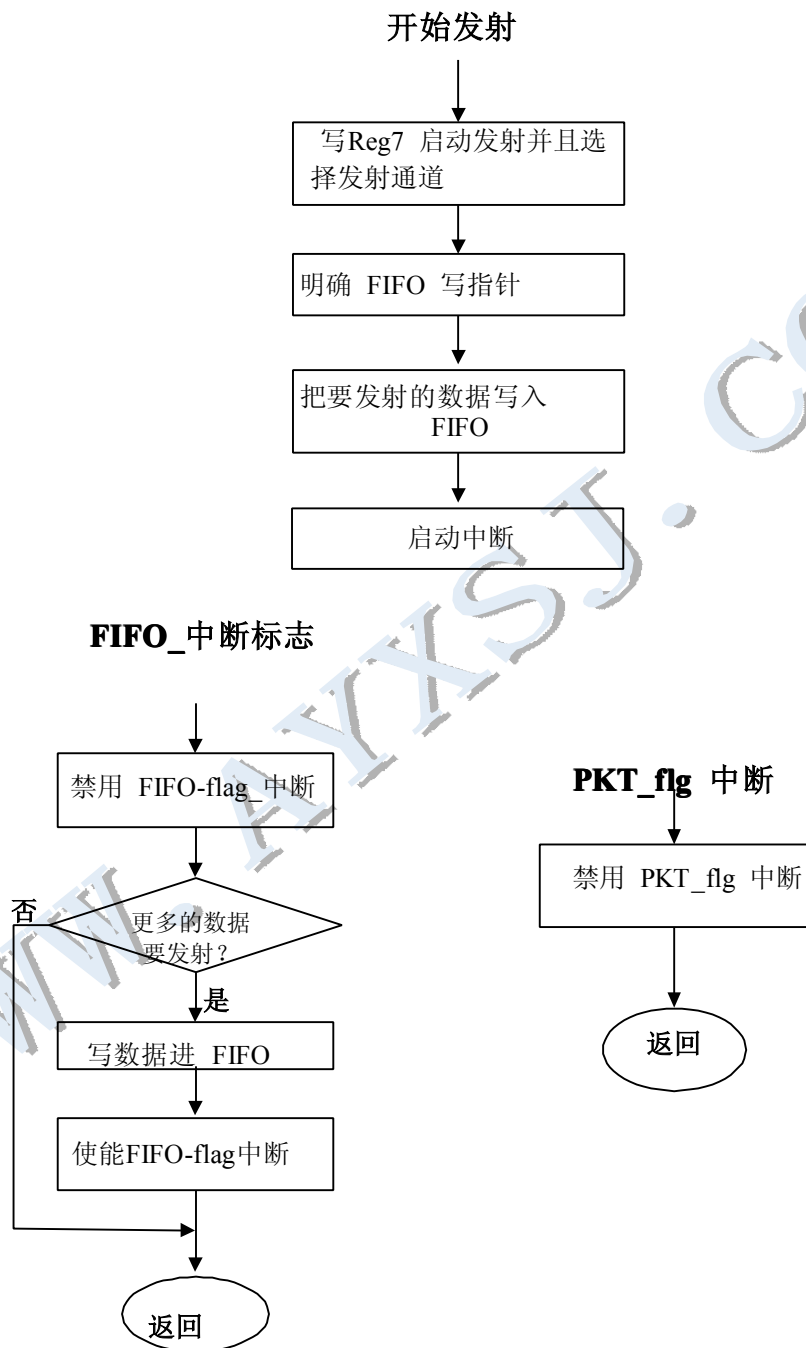


RX时序图 PKT_flag和FIFO_flag 高电平有效
(微处理器处理数据包长度)

1. 射频初始化流程图

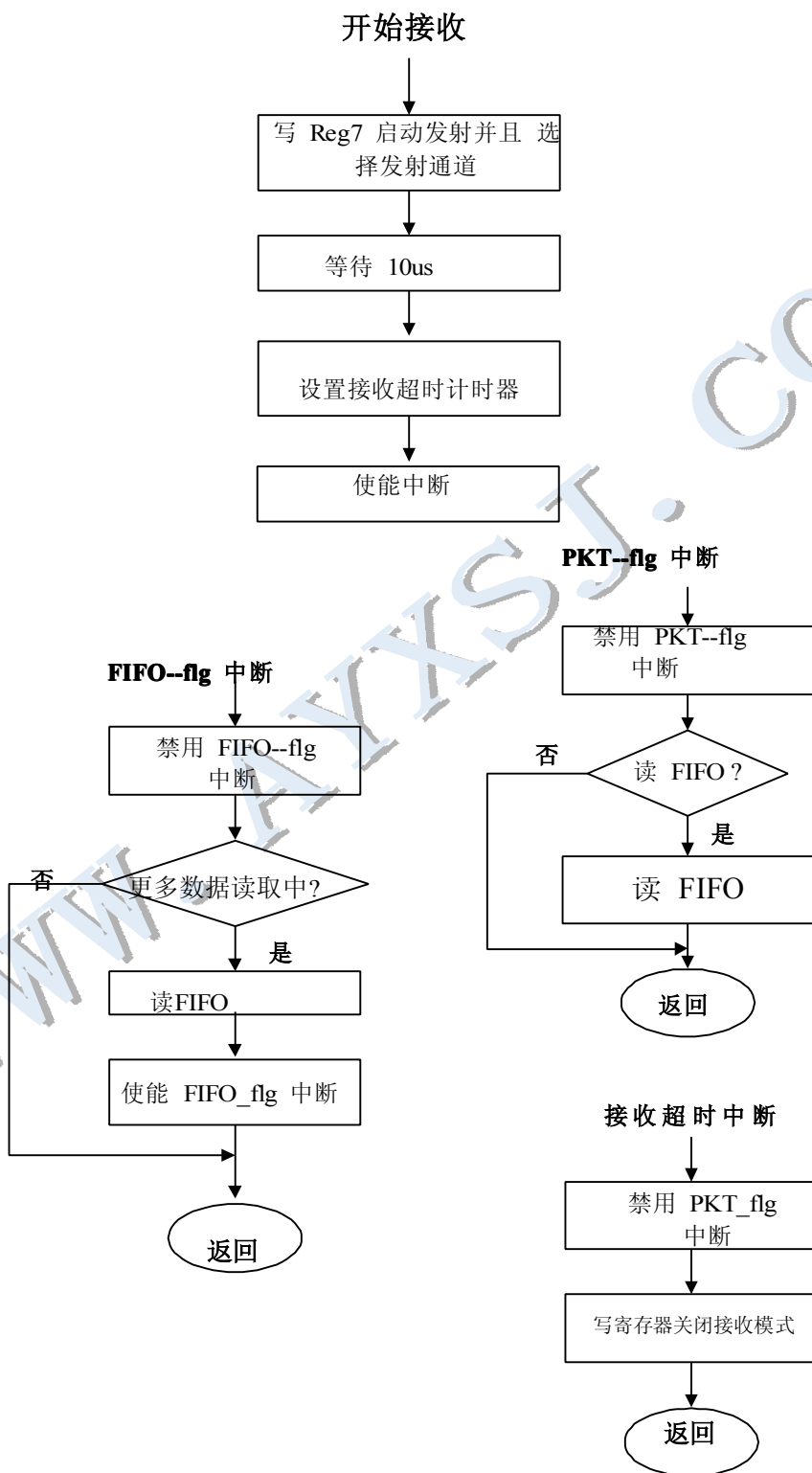


2. 发射流程图 **FIFO_flg** 和 **PKT_flg** 是单片机的中断信号。



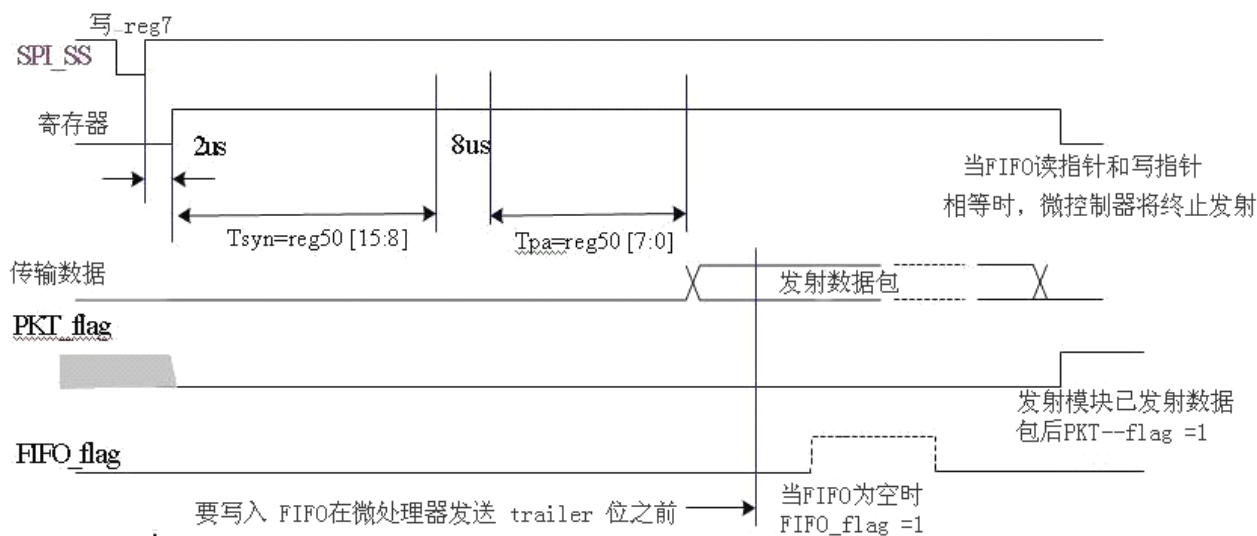
3. 接收流程图

FIFO_flg 和 PKT_flg 是单片机的中断信号。

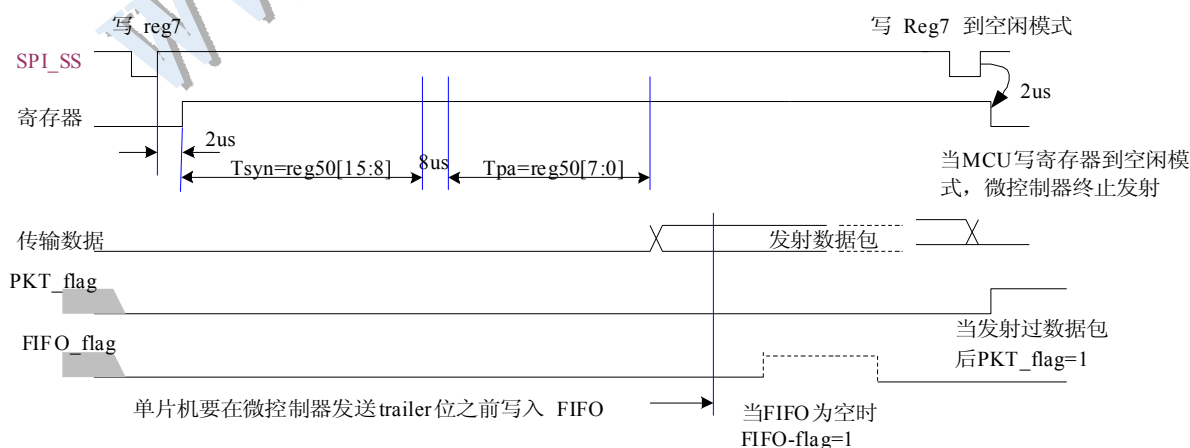


发射/接收时序图——（单片机处理数据包长度）

当 $\text{Reg57}[13] = 0$ ，JF24C 微处理器始终遵循单片机的指令。由单片机来确定发射/接收的启动和停止。但有一种情况 JF24C 微处理器可以停止执行：如果 $\text{Reg57}[8] = 1$ ，微处理器在发射模式保持检测 FIFO 的写指针和 FIFO 的读指针。如果单片机停止写入数据进 FIFO，当微处理器检测到没有数据要传送（这意味着 FIFO 完全空），JF24C 微处理器将自动退出发射模式。时序图见下图所示，如果 $\text{Reg57}[8] = 0$ ，微处理器不停发射，直到单片机写 Reg7 到空闲状态使 FIFO 是完全空白。时序图见下图所示：



发射时序图 PKT_flag 和 FIFO_flag 为高电平有效
（单片机处理数据包长度，JF24C 微处理器确定发射）



发射时序图 当 PKT_flag 和 FIFO_flag 为高电位有效
（单片机处理数据包长度，JF24C 微处理器无法检测 FIFO 的读/写指针）

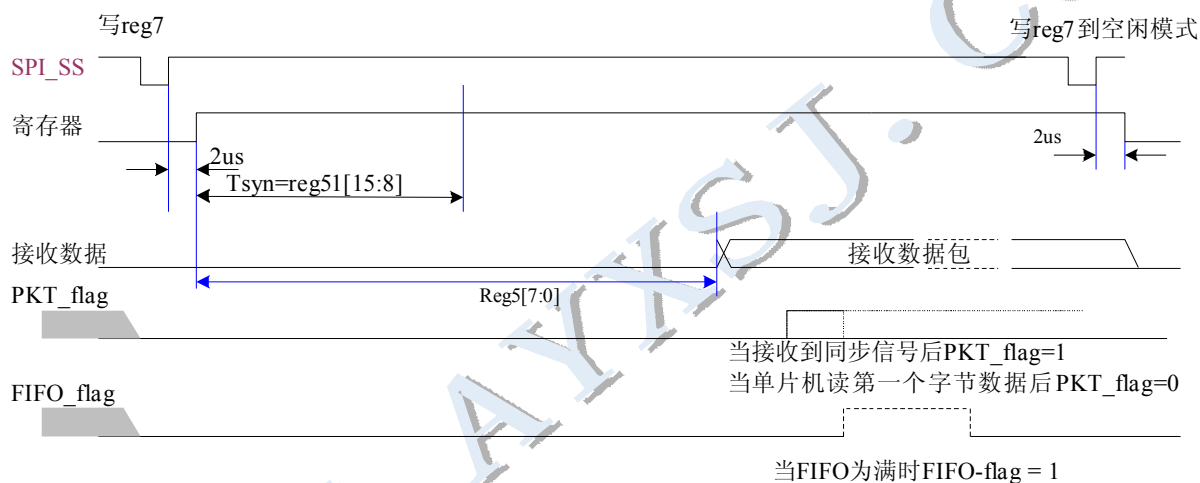
注意：当单片机处理发射/接收数据包长度，绝不能让 FIFO 产生上溢和下溢。它可以通过控制 Reg56 TX_FIFO_threshold 和 RX_FIFO_threshold 来设置。它的值取决于的 SPI 速度。

接收时序图

见下图，当单片机写 Reg7 [7] = 1，在 Tsyn 时间后 JF24C 微处理器将自动开始检测同步。

当检测到有效的同步信号，将与单片机相连的 PKT_flag 设置为高，接着开始填写数据到 FIFO。

PKT_flag 将继续保持高，直至单片机读出第一个字节数据。当单片机读出第一个字节的数据，PKT_flag 就会变成无效状态，直到下一个发射/接收周期。它总是需要单片机写 Reg7 停止接收。



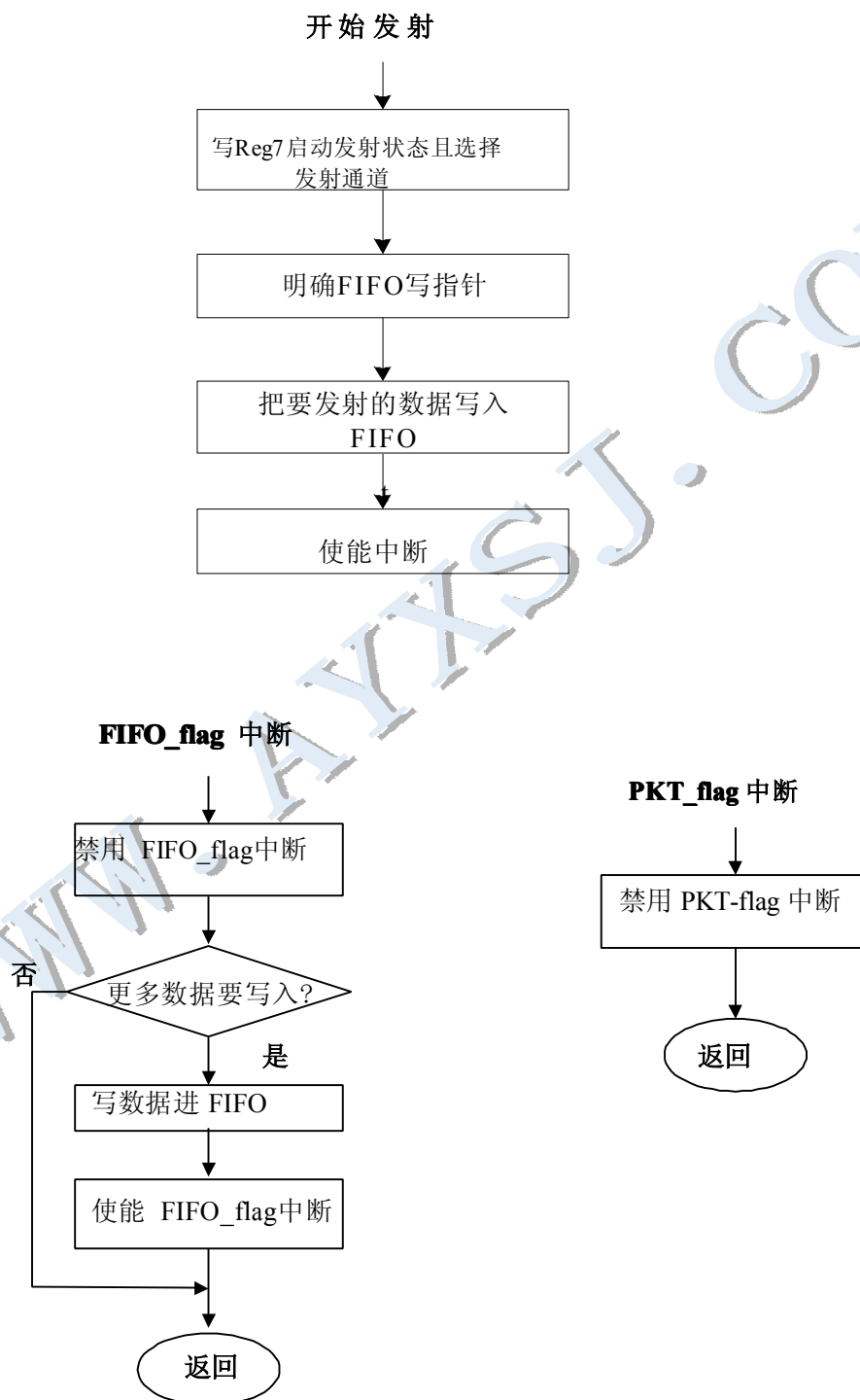
接收时序图（当 PKT_flag 和 FIFO_flag 为高电位有效）

（单片机处理数据包长度）

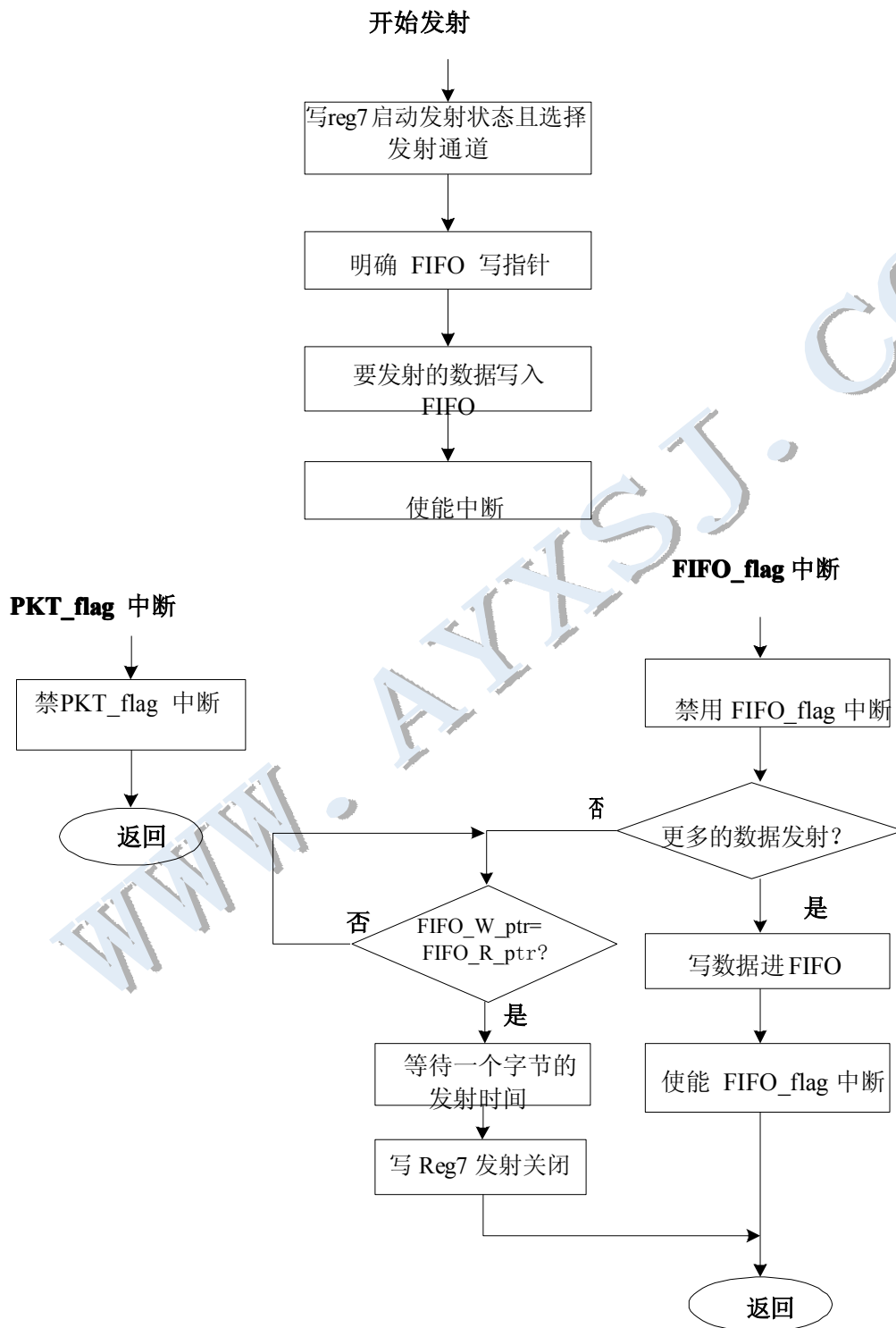
流程图范例：上电和寄存器初始化序列和由微处理器控制数据包长度相同。

TX/RX 流程图, PKT_flag 和 FIFO_flag 由MCU得外部中断源定义。

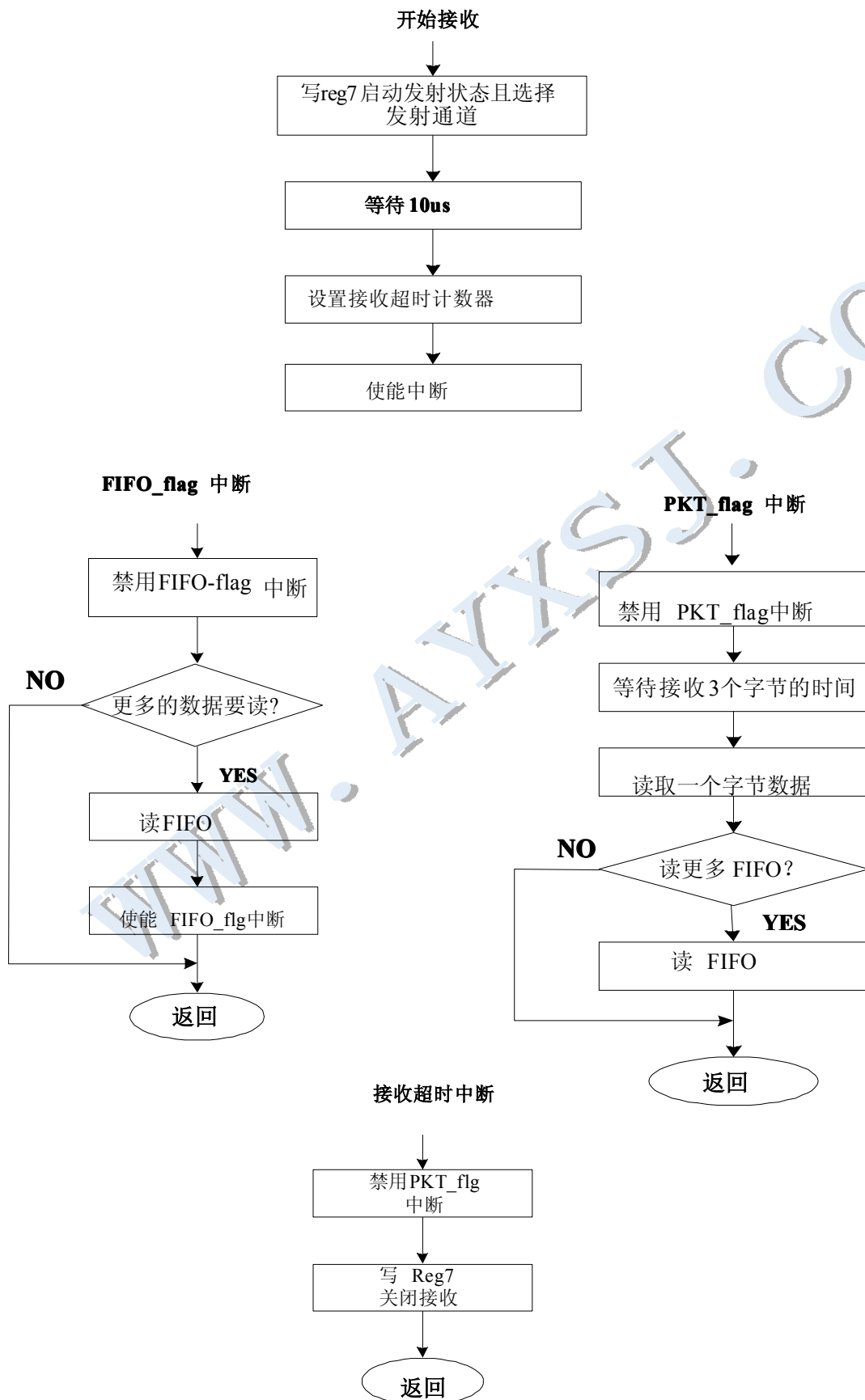
发射流程图 (微处理器检测 FIFO TX/RX 指针)



发射流程图 (微处理器不检测 FIFO TX/RX 指针)



接收流程图



典型寄存器的值

以下寄存器值是建议的大部分客户

Reg. address	Read/Write	Default value (Hexadecimal)	Recommend value (12MHz crystal frequency) (Hexadecimal)
0	R/W	0000	CD51
2	R/W	00C1	0061
4	R/W	0688	3CD0
5	R/W	0041	00A1
9	R/W	0003	3003
14	R/W	6617	6697
16	R/W	0000	F000
18	R/W	FC00	E000
19	R/W	0014	2114
20	R/W	8103	819C
21	R/W	0962	6962
22	R/W	2602	0402
23	R/W	2602	0802
24	R/W	30C0	B080
25	R/W	3814	7819
26	R/W	5304	6704
48	R/W	1800	5800
51	R/W	4000	A000
56	R/W	4407	4407
57	R/W	B000	E000*

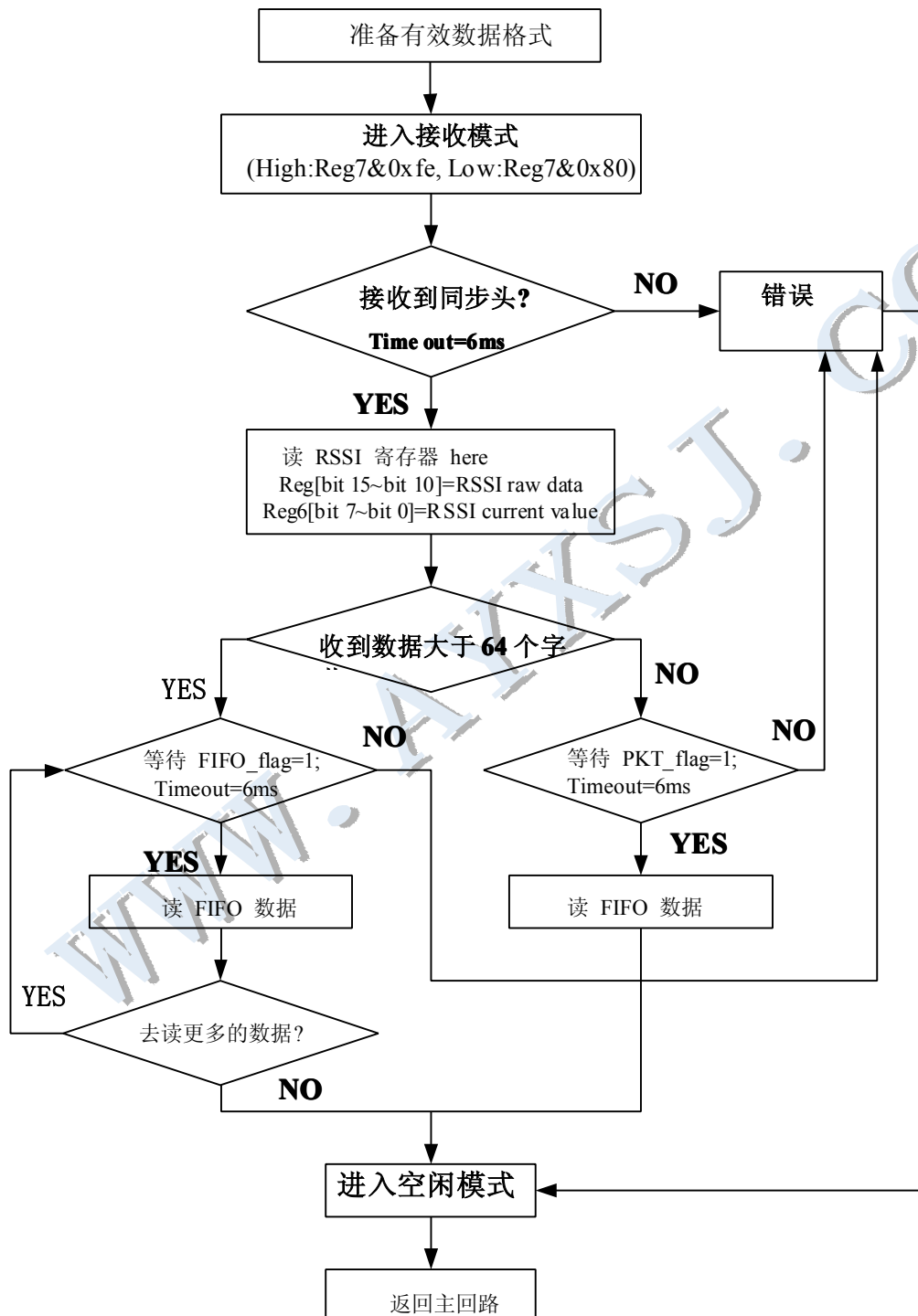
推荐寄存器设置

Reg57, 如果单片机处理数据包长度并且微处理器检测 FIFO 完全空白, Reg57 = 0xC080

Reg57, 如果单片机处理数据包长度和终止 TX Reg57=0xC000

附录1： 读接收信号强度指示值和接收信号强度指示参考表

阅读接收信号强度指示值流程图： （发射方必须在不断传输）



阅读接收信号强度指示值，示例代码如下：(使用 Keil C51)

```
unsigned int RX_packet(bit EnableTimeout,unsigned int bytes,bit rssi)
{
    bit Timex= 0, TimeOut = 0;                unsigned int beri, reg6;
    unsigned char i, j, k,ErrCnt=0;
    unsigned int raw_rssi,d_rssi;
    j=(bytes+1)/64; k=(bytes+1)%64;
    GenerateData();        // prepare the data
    Reg_write16(RF_REG_07, (gReg7_high & 0xfe), (gReg7_low | 0x80)); //enter RX mode
    if(rssi)
    {
        Timer4_Init(CLK_6MS);
        while(((0x0400 & Reg_read16(MAIN_STATUS)) != 0x0400) && (timer4_flag==ING));
        if(timer4_flag)
        {
            Timex = 1;
            goto Err;
        }
        reg6 = Reg_read16(RF_REG_06);
        goto Lb;
    }
    Timer4_Init(CLK_25us);        //make sure PKT has time to go low
    while((timer4_flag==ING) && (PKT==1))
        ;Lb:
    if(j==0)
    {
        if (EnableTimeout)
        {
            Timer4_Init(CLK_6MS);
            while ((PKT==0) && ((Timex= timer4_flag) == ING))
                ;
            if(Timex)
                goto Err;
        }
    }
}
```

else

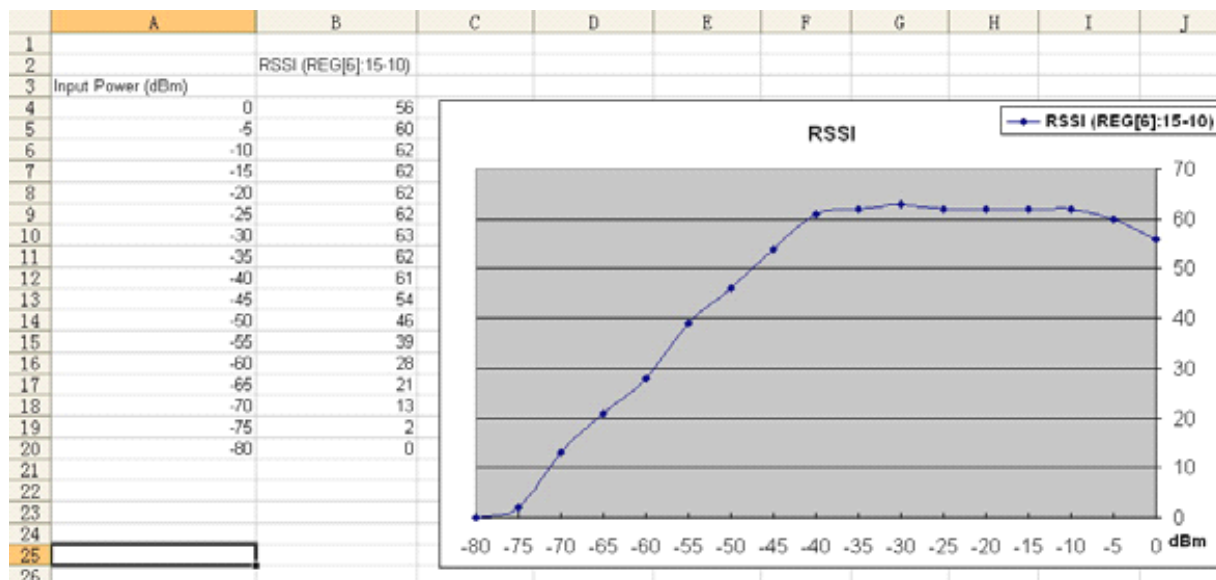
```
        while((PKT==0) && (SW2 == 0))
            ;
        read_FIFO(Temp,(bytes+1));
    }
    else if(j>0)
    {
        if(k==0)
        {
            for(i=0;i<j;i++)
            {
                if(EnableTimeout)
                {
                    Timer4_Init(CLK_6MS);
                    while((FIFO_flag==0) && ((TimeOut=timer4_flag)== ING));
                    if(TimeOut)
                    {
                        ErrCnt = i+1;
                        if(ErrCnt==1)
                        {
                            Timex = 1;
                            goto Err;
                        }
                        else
                        {
                            read_FIFO((Temp+i*64),Temp[0]%64);
                            Timex = 0;
                            goto Err;
                        }
                    }
                }
            }
        }
        else
        {
            while(FIFO_flag==0);
            read_FIFO((Temp+i*64),64);
        }
    }
}
```

```
}  
  
}  
else if(k>0)  
{  
    for(i=0;i<j;i++)  
    {  
        if(EnableTimout)  
        {  
            Timer4_Init(CLK_6MS);  
            while((FIFO_flag==0) && ((TimeOut=timer4_flag) == ING))  
                ;  
            if(TimeOut)  
            {  
                ErrCnt = i+1;  
                if(ErrCnt==1)  
                {  
                    Timex = 1;  
                    goto Err;  
                }  
                else  
                {  
                    read_FIFO((Temp+i*64),Temp[0]%64);  
                    Timex = 0;  
                    goto Err;  
                }  
            }  
        }  
    }  
    else  
        while(FIFO_flag==0)  
            ;  
    read_FIFO((Temp+i*64),64);  
}  
if(EnableTimout)  
{  
    Timer4_Init(CLK_6MS); //This not work at all, because timeout is impossible. So you can  
    del this code
```

```
while((PKT==0) && ((Timex=timer4_flag) == ING))
    ;
    if(Timex)
        goto Err;
    }
else
    while((PKT==0)&&(SW2==0))
        ;
        read_FIFO((Temp+j*64),k);
    }
}
Err:
Reg_write16(RF_REG_07, (gReg7_high & 0xfe), (gReg7_low & 0x7f)); //enter idle mode
if (Timex) beri=
0xffff; else
    beri = verify_FIFO_data( gDAT,Temp, gLength); // == 64, SPI takes 113uS
if(rssi)
{
    raw_rssi = (reg6 & 0xfc00) >> 10;
    d_rssi = reg6 & 0x00ff;
    printf("\rRAW_RSSI= %03u    D_RSSI = %03u",raw_rssi,d_rssi);
}
return beri;
}
```

接收信号强度指示参考表

Table 1. PA V.S. RSSI value



※ Golden range input power is recommend -60dBm.

寄存器设置描述

目录

- 1 JF24C RF寄存器定义
- 2 JF24C 微处理器收发寄存器定义
- 3 JF24C推荐寄存器值设置表

接收射频控制(写/读) – 寄存器0

(默认=0xCD51)

位编号	位名称	描述
15 – 11	MIXER_GAIN[4:0]	设置5位接收混频器增益值
10	MIXER_LP	“1” 使能接收混频器的低功耗模式
9 - 5	RX_BPF_Q[4:0]	为带通滤波器设置5位Q值
4-0	RX_BPF_GN1[4:0]	设置5位为部分1接收器的带通滤波器获得值

射频状态 (只读) – 寄存器1

位编号	位名称	描述
15	VPON	如果正确的VCO曲线已被选中，有两个指标显示，这是其中一个
14	VTFREQ	如果正确的VCO曲线被选择，这是另一个显示的指标；当正确的VCO曲线被选择，“VPON” 和 “VTFREQ” 将是“1”。
13-8	RC[5:0]	由RC计算器电路产生的6位RC带通滤波器校准值
7-0	ADC_DOUT[7:0]	由A/D转换器产生的8位值。

射频合成器/振荡器配置 (写/读) – 寄存器 2

(默认 = 0x137B)

位编号	位名称	描述
15	FORCE_IDLE	当置“1”时重置数字调制解调器进入空闲模式
14	BYPASS_VCO_CAL	在上电或者空闲状态禁用压控振荡器自动校准功能。这一位优先于VCO_CAL_EN (寄存器26的第1位)。
13	Reserved	
12	DEVIATION_SEL	“1”：选择320KHz偏差；“0”：选择160KHz偏差
11-8	VCO_FC[3:0]	为发射调制带宽设4位值；这4位由压控振荡器控制。
7-4	VPON_TUNE[3:0]	设置 VPON 电压
3-0	VTFREQ_TUNE	设置 VTFREQ电压

TX/RX 状态(只读) – 寄存器 3

位编号	位名称	描述
15 – 11	BLUE_RF_STATE[4:0]	显示 BlueRF有限状态机的状态
10	RC_FIN	RC完成状态
9 - 6	VCO_CAL[3:0]	自动校准后表明目前的压控振荡器频段设置
5	VCO_CAL_ERROR	出错标志，表明没有适当的 vco 曲线可以发现。

4	RF_SYNTH_LOCK	说明RF合成器的锁状态
3 - 0	TEMP_SENSOR[3:0]	表明目前的温度范围

BLUE_RF_STATE [4:0]	状态
00000	off
00001	PwrOnWaitXTL
00010	HoldXTL
00011	idle
00100	sleep
00101	SleepWaitXTL
00110	VCO_Sel
00111	VCO_Wait
01000	RXPLLWait1
01001	RXPLLWait2
01010	RXWideFilt
01011	RXNarrowFilt
01111	VCO_PwrOnWait
10000	WaitDataSync1
10001	WaitDataSync2
10010	DataSync
10011	EnblePA1
10100	EnblePA2
10101	TXData
10110	DisablePA1
10111	DisablePA2
11000	DisablePA3

射频发射控制(写/读) – 寄存器 4

位编号	位名称	描述
15-14	RSSI_GN_ADJ[1:0]	RSSI 控制信号
13-10	TXDAC_DC[3:0]	为 TXDAC DC的电压水平置4位值
	RSSI_DIS	“1”:禁用 RSSI特征.
8	Reserved	
7-3	TXDAC_GAIN[4:0]	TXDAC的5位增益值.
2-0	BG_TBIT[2:0]	为控制带隙基准设置3位测试模式

接收定时/功率控制 (写/读)–寄存器 5

位编号	位名称	描述
15 – 11	SYNTH_ON_DELAY_CNT[4:0]	在BLUE RF的等待数据同步状态，射频震荡器将首先被启用。由SYNTH_ON_DELAY_CNT计数器进行时间偏移控制。当计数器计数到零并且SYNTH_IDLE_OFF = “0”，合成器将被使能。每一次增量是1uS。
10 - 9	Reserved	
8	REG_PROTECT	该位是用来保护测试寄存器，除了 Register 7, 8, 9, 30, 31 和 REG_PROTECT，其他的寄存器通过置 REG_PROTECT 为“1”被保护。当 REG_PROTECT 是“0”时，所有的寄存器被通常访问。
7 – 0	RX_DELAY [7:0]	8位接收延迟（从合成程序接收注册到开始传输BDATA1 到 BBIC）.每次增量是1uS。

TX 系数/RSSI 值 (只读) – 寄存器 6

位编号	位名称	描述
15 – 10	RAW_RSSI[5:0]	显示 4 位来自模拟电路的原接收信号强度指示值，达到内部调试的目的
9	AGC	表明目前的 1 位自动增益控制值。（另见寄存器 14 和寄存器 17）
8	RSSI_VALID	说明目前的接收信号强度指示值（本寄存器的 0-3 位）有效
7 - 0	RSSI[7:0]	表明目前的 8 位接收信号强度指示值

射频合成器/ TX-RX控制 (写/读)–寄存器 7 (默认值 = 0x0030)

位编号	位名称	描述
15 - 14	Reserved	
13 - 9	SWALLOW [4:0]	5位合成器计数器，当 RF_PLL_DIRECT置“1”时，这合成器将与 Register7[13:9] 和Register7[6:0]一起被直接编程。这个合成器的频率不会被编程， $f = 2402 + \text{PLL_CH_NO}$
8	DBUS_TX_EN	使能传输序列的状态机控制
7	DBUS_RX_EN	为状态机控制使能接收序列。注意：这 DBUS_TX_EN 和 DBUS_RX_EN 不能同时为“高”，否则，这种配置不能工作。
6 - 0	RF_PLL_CH_NO [6:0] /RF_PLL[6:0]	这 7 位代表的蓝牙射频频道。该频道的频率将是： $f=2402+\text{PLL}+\text{PLL_CH_NO}$ 。当 RF_PLL_DIRECT 设置为“1”，该合成器将与 Register7 [13-9]和Register7 [6: 0]一起被直接编程。Register7[6:0] 被用作 7 位合成器程序计数器。这个合成器的频率将不会被编程， $f = 2402 + \text{PLL_CH_NO}$ 。

RF_TX_EDR 控制(写/读)–寄存器 8

(默认值 = 0x0404)

位编号	位名称	描述
15-13	Reserved	
12-8	ldo_lp_sleep[3:0]	设置 5-bit 低压差休眠电流
7-5	Reserved	
4-0	Ldo_sp-normal[4:0]	设置5-bit低压差工作电流

RFIC控制(写/读) – 寄存器 9

(默认值 = 0x0404)

位编号	位名称	描述
15--12	PA_PWCTR[3:0]	功率放大器高功率版本
11 – 7	PA_GN[4:0]	5位发射功率放大器增益设置
6	TR_SW_POLARITY	当为“0”时, TR_SW的默认极性被选择(TR_SW: “0”是发射; “1”是接收); 当为“1”时, TR_SW的极性相反。
5	APLL_VT_SENSE	APLL 的VT 意义位
4	APLL_VT_FORCE	APLL 的VT动力位
3	APLL_BP	模拟PLL旁路模式. 当为“1”时, Fout = Fin.
2	APLL_PDN	模拟PLL 的掉电模式. 当为“1”时, APLL掉电。
1	BRCLK_SEL	BRCLK的选择引脚. 如果BRCLK_SEL = “1”, BRCLK = crystal_out 或者如果BRCLK_SEL = “0”, BRCLK = TXCLK(1MHz). (默认值 = “1”).
0	BRCLKEN	当 BRCLKEN为高时, BlueRF的输出接口 BRCLK工作(默认值 = “1”).

AMS 测试控制 (写/读) – 寄存器10

(默认值 = 0x0404)

位编号	位名称	描述
15	ENTER_SLEEP	当置“1”时, 芯片进入休眠模式以省电
14	AMS_TST_ENB	当置“1”时, 使能 AMS 测试模式 并且旁路 BlueRF有限状态机
12-13	AMS_TST_MD_SEL	00 :正常运行模式, 01 : 使能检测模数转换器的数字传输I通道 10 : 使能检测模数转换器的数字传输Q通道 11 : 使能检测模数转换器的接收信号强度指示.
11	TXDAC_MOD_MON	使 DAC 的输出检测启用.
10	BPF_TST_PD	当置“1”时表明在AMS_TST_ENB=“1”期间关闭带通滤波器.
9-8	Reserved	
7	Ext_pa_sel	0: RXDATA=EXT_PA_CTRL0, Test1=EXT_TR_SWb, teste=EXT_TR_SWb. 1: RXDATA, Test1, teste 回到以前的定义。
6	Reserved	
5	LNA_TST_PD	当置“1”时,表明在 AMS_TST_ENB=“1”期间掉电低噪声放大器

4	ADC_TST_PD	当置“1”时, 表明在AMS_TST_ENB=“1”期间掉电模数转换器.
3	RF_VCO_TST_PD	当置“1”时,表明在AMS_TST_ENB=“1”期间掉电RF振荡器电路
2	RC_TST_START	在AMS_TST_ENB=“1”期间, 为RC START提供测试值.
1	RC_TST_PD	当置“1”时, 表明在AMS_TST_ENB=“1”期间掉电 RC集成电路
0	MIXER_TST_PD	当置“1”时, 表明在AMS_TST_ENB=“1”期间掉电混频器

AMS 测试控制(写/读)–寄存器11

(默认值 = 0x4041)

位编号	位名称	描述
15	TX_DAC_TST_PD	当置“1”时, 表明在AMS_TST_ENB = “1”期间掉电发射数模转换器
14	AMS_BUF_PD	测试缓冲区的掉电控制, . “1”意味着掉电, “0”意味着测试模式.
13	AMS_BUF_LS	测试缓冲区的信号电平移位控制当置“1”时,信号级别将被转移0.6V; 其他情况, 例如位 “0” 时信号级别无任何变化。

12	AMS_BUF_GN	测试缓冲区增益控制“1”→2V/V; “0” →1V/V.
11	AMS_BUF_SIN	测试缓冲区单端输入控制 “1”→信号终止; “0”→不同的模式.
10	reserved	
9	RSSI_PDN	“1”: 掉电RSSI.
8	reserved	
7	TX_PA_TST_PD	当置“1”时,表明在AMS_TST_ENB= “1”时关闭功率放大器模块
6	TR_TST_SW	当置“1”时, 表明在AMS_TST_ENB = “1”期间把射频开关在接受状态 置 “0” 时表明AMS TST ENB = “1”期间TX的传输路径
5	ADC_TST_CLKEN	当置 “1”时,给 ADC 时钟输入提供使能信号在AMS_TST_ENB = “1” 时
4	SYNTH_TST_PD	当置1时,表明在AMS_TST_ENB =“1”时关闭合成器电路
3	reserved	
2	VCO_PDN	“1” :关闭VCO电源.
1	PDN	“1” :合成器断电.
0	XTAL_OSC_EN	当置“1”时 ,使能内部振荡器电路.

AMS 测试控制(写/读) – 寄存器12 (默认值 = 0x0000)

位编号	位名称	描述
15-6	resv[15:0]	保留寄存器数据接口.
5-0	XI_trim[5:0]	削减晶振

AMS 测试控制 (写/读) – 寄存器13 (默认值 = 0x0000)

位编号	位名称	描述
15-0	AMS_SW_SEL[15:0]	开关选择控制混合信号电路地区。请注意这个选择是独立的控制 AMS_TST_ENB。

AMS TEST Control (Write/Read) – Register 14 (默认值 = 0x0000)

位编号	位名称	Description
15-12	AG_TH[3:0]	为自动增益控制设置高约束值.
11-8	AG_TL[3:0]	位自动增益控制设置低约束值
7	RX_BPF_LP	“1”:使能BPF和带通滤波器的低功率模式.
6	AGC_FORCE	“1”:迫使自动增益控制运行无论信号强度如何变化.(也要看 Register 6 and 17).
5	AGC_DISABLE	“1”: 禁用自动增益电路
4-0	BPF_BW[4:0]	为接收器的带通滤波器设置带宽值

RC 控制 (写/读) – 寄存器 15 (默认值 = 0x017B)

位编码	位名称	描述
15	RC_SEL	从这个寄存器的0-5位手动选择RC的值
14	rst_cnt2_sel	“1”: 选择32, “0”: 选择 64.
13	rst_cnt3_sel	“1”: 选择7, “0”:选择 3.
12	rst_2_en	“1”: 启用额外解调reset2, “0”: reset2 禁用, 回到以前的状态
11-6	RC_OFFSET[5:0]	为 RC CALC 电路设置6位偏移值.
5-0	RC_MANU[5:0]	当RC_SEL被置位时, 手动选择6位RC的值

BPF Control (写/读) –寄存器 16 (默认值 = 0xF000)

位编号	位名称	描述
15-13	RX_BPF_VO_Q[2:0]	设置 3 位值调整调 Q 路径振幅带通滤波器的输出。
12-8	RX_BPF_VO_I1[4:0]	设置5位值调整调I路径振幅带通滤波器的输出（第一阶段）
7-3	RX_BPF_VO_I2[4:0]	设置5位值调整调I路径振幅带通滤波器的输出(第二阶段)
2-0	RX_BPF1_VO[2:0]	设置3位值调整带通滤波器的幅度输出

AGC 控制 (写/读) – 寄存器17

(默认值 = 0x0000)

位编码	位名称	描述
15-0	Reserved	

TX/RX 数据控制(写/读) – 寄存器 18

(默认值 = 0xE000)

位编码	位名称	描述
15	SOFTWARE_CNTL	设置这一位, 使软件控制PA_ON 和SW_ON的时间, 他们分别基于 TX_PA_ON_DELAY 和TX_SW_ON_DELAY 的值。默认设置‘0’ 是选择微处理器状态机去控制他们的时序。
14	RX_DATA_INVERSE	这个控制信号为使用IF信号的上升沿或下降沿提供了方便。当为“1”时, 接收信号BDATA1 或者RXDATA 的极性是相反的。
13	BYPASS_PLL_LOCK	当置“1”时, 发射机在经过发射延时后, 就开始发射将数据, 而不会等待射频锁相环到稳定状态。
12-10	TX_CW[2:0]	这个时间期限, 是在TR-SW开启后发射模式连续传输数据位时设置的。 当这个时间到期时, 发射数据将从BB被发射。
9	BRCLK_SW	“1”: 从APLL的12M路线到BRCLK 引脚。
8	TX_DATA_INVERSE	当置 “1” 时,传输数据BDATA1的极性将与内部相反。
7-0	reserved	

DC Offset控制 (写/读) – 寄存器 19

(默认值 = 0x2114)

位编码	位名称	描述
15	reserved	高功率低噪声放大器。
14	LOBUF_HP	高功率的功率放大器
13	ADC_LP	低功耗的模数转换器
12	Reserved	
11-9	Reserved	
8	Bpktctl_sel	BPKTCTL 控制模式, 以缩小全模式接收
7-4	WIDE_TC[3:0]	为直流偏移宽模式选择时间期限; 在这个偏移内直流偏移电路将利用所确定的2-3位速度位, 跟踪直流偏移。所选择的时间期限到期后直流偏移电路将自动切换到高速的窄模式(这个寄存器的0-1位)。 请注意跟踪速度的宽模式将始终快于窄模式。
3-2	WIDE_TRACK_SPED [1:0]	在全收发模式为直流偏移跟踪速度设置Alpha。
1-0	NARROW_TRACK_ SPEED[1:0]	在接受窄模式, 为直流偏移跟踪速度设置Alpha值。

WIDE_TC

WIDE_TC [3:0]	时间期限 (us)
0000	0
0001	8
0010	16
0011	24
0100	32
0101	40
0110	48
0111	56
1000	64
1001	72
1010	80
1011	88
1100	96
1101	104
1110	112
1111	120

WIDE_TRACK_SPEED

WIDE_TRACK [1:0]	Alpha Value
00	4 (慢)
01	6
10	8
11	10 (快)

NARROW_TRACK_SPEED

WIDE_TRACK [1:0]	Alpha Value
00	1 (慢)
01	2
10	3
11	4 (快)

PLL Synthesizer控制(写/读) – 寄存器20

(默认值 = 0x0003)

位编码	位名称	描述
15	PLL_FREQ_PLUS	设置选择上层边低中频信号或下层边低中频信号被解调
14-8	PLL_RX_FREQ_OFFSET	设置射频锁相环频率以抵消本地振荡器
7-5	A_INIT[2:0]	在断电模式下的 PLL 频率计数初值
4	SYNTH_LP	设置低功耗PLL
3	LNA_LP	设置低功耗 LNA
2	LOBUF_LP	设置低功耗本地振荡缓冲器

PLL Synthesizer Control (写/读) – 寄存器 21

(默认值 0x6962)

位编码	位名称	描述
15	APLL_IDLE_OFF	1: 在空闲模式APLL将进入掉电状态
14	RF_VCO_IDLE_OFF	射频压控振荡器将进入掉电状态, 如果SYNTH_IDLE_OFF为高。否则它将保持活跃当SYNTH_IDLE_OFF为低时。
13	SYNTH_IDLE_OFF	合成器将进入掉电状态, 如果 SYNTH_IDLE_OFF为高。否则 他们将保持活跃当 SYNTH_IDLE_OFF为低时。
12	RF_PLL_DIRECT	当RF_PLL_DIRECT 置“1”时, 这合成器将直接被Register7[11:0]编程。这合成器的频率不能被编程时 $f = 2402 + \text{PLL_CH_NO}$ 。
11 – 0	PLL_RF_FREQ_BASE	设置 RF PLL 的基频 (默认值 = 2402 MHz)

TX RF Timing Control (写/读) – 寄存器 22

(默认值 0x2602)

位编码	位名称	描述
15 – 8	TX_PA_ON_DELAY	这些寄存器是射频功率放大器 (电源控制) 计数器的初始值。计数器开始计数到零后, 在功率放大状态射频发射调制器的电源被打开。当计数器为零时, 功率放大器是活跃的。每次增量是1 uS;这些寄存器设置时间延迟在主张 PA_ON 后BPKTCTL 被断言, 提供 SOFTWARE_CNTL (寄存器18的15位) 设置。
7 – 0	TX_PA_OFF_DELAY	这些寄存器是发射功率放大器断电控制计数器的初始值。当BLUE-RF在功率放大关闭状态, 这个计数器减到0, 发射功率放大将被关闭。每次增量是1 uS。

TX RF/WAKE-UP Timing控制(写/读) – 寄存器23

(默认值 0x0802)

位编码	位名称	描述
15 – 8	TX_SW_ON_DELAY	这些寄存器是 TR_SW选择控制计数器的初始值。当发射功率放大器打开后，计数器开始计数归零。当计数器为零时，收发开关移到发射通道。每次增量是1 uS，这些寄存器也都设置了SW_ON时间延时，同时也提供SOFTWARE_CNTL (寄存器18的第15位) 设置。
7 – 0	WAKE_UP_TIME	这些寄存器是从休眠模式到空闲模式被唤醒时间计数器的初始值。在 BXTLEN信号从低到高后，这个计数器开始计数归零。在计数器为零后开始进入空闲模式。每次时间增量是1 uS。

Fractional-N/VCO Control 1 (写/读) –寄存器 24

(默认值 0xB040)

位编码	位名称	描述
15	APLL_LOCL	只读
14	FORCE_DIT	“1” :无条件的使能抖动.
13	DIT_P_EN	“1”: 使能抖动功能只有当相位插补计数器的值在0-7之间或者在56-63之间。
12	SEL_16_32	“0”: 当抖动功能已启动选择加/减16 在计算sigma-delta输出 “1”: 在计算时选择加/减32 .
11-10	PRE_SCAL_DLY[1:0]	预标MC信号延时时间设定
9-6	VCO_FB[3:0]	压控振荡器频率波段设置.
5	VCO_VT_SENSE	压控振荡器 VTUNE 意义位: “1” 使能 VTUNE 传感在 VTUNE_IO 引脚
4	Reserved	

位编码	位名称	描述
3	TEST_DIV	“1”: SYNTH_SIGOUT 引脚路线的 DIV 信号光强
2	TEST_FREF	“1”: SYNTH_SIGOUT 引脚路线 FREF 信号
1	TEST_NA	“1” SYNTH_SIGOUT 引脚路线时钟的 N/A 计数器
0	TEST1_OUT_EN	“1”: 选择测试引脚作为 SYNTH_SIGOUT输出，启到监视的目的

Fractional-N/VCO Control 2 (写/读) – 寄存器 25 (默认值 0x7819)

位编码	位名称	描述
15-12	CLOSE_LP_DLY[3:0]	选择高斯滤波器的延时输出, 为了弥补Sigma-Delta调制器的延迟
11	SEL_GAU_OUTPUT	“1”: 选择高斯滤波器的输出以馈入在发射模式的模数转换器 “0”: 选择方波输出以馈入发射模式的数模转换器
10	A_INIT_EN	PLL 频率计数初始值
9	SYN_FST_MODE_BP	“1”: 合成器的绕到快速模式.
8	SYN_FIN_SEL	设置内部6MHz或12MHz的参考时钟作为合成器, 以节省电源
7-6	T_RES1[1:0]	为射频控制信号保留寄存器1
5-0	RCP0[5:0]	电荷泵电流设置为 0 。此设置将仅适用于, 当芯片在接收模式时。

CLOSE_LP_DLY

CLOSE_LP_DLY [2:0]	Delay (# of 12MHz clock)
000	4-clock delay
001	5-clock delay
010	6-clock delay
011	7-clock delay
100	8-clock delay
101	9-clock delay
110	10-clock delay
111	11-clock delay

Miscellaneous Control (杂项控制) (写/读) – 寄存器26 (默认值 0x6704)

位编码	位名称	描述
15-10	RCP1[5:0]	电荷泵电流设置1. 这个设置在芯片的发射模式有效
9	VCO_ADJ_EN	“1”: 启用自动调整压控振荡器的逻辑
8	VCO_VT_FORCE	为VCO VT提供控制信号
7	TEST1_DIGI_SEL	“1”: 内部线路信号 UPDATE到TEST1 引脚达到调试的目的 “0”: 内部线路信号 ACCU_IN_MSB到TEST1 引脚. 注意寄存器24的0位优先于这一位。

位编码	位名称	描述
6	BYPASS_GAU	“1” 在发射模式设置数字高斯滤波器旁路逻辑
5-3	VCO_BIAS[2:0]	压控振荡器偏置电流设置
2	VCO_FB_DIVSEL	“1”: 为自动压控振荡器的调整功能, 使能8部分的VCO频段设置 “0”: 为自动压控振荡器的调整功能, 使能4部分的VCO频段设置
1	VCO_CAL_EN	手动启用 RFIC, 使其自动启动 vco 校准曲线程序在空闲状态。

0	AUTO_VCO_SEL	“1”:选择 CAL_VCO (寄存器3的5-8位) 作为有效的VCO频段设置 “0”:选择 VCO_FB (寄存器24的6-9位) 作为有效的VCO频段设置
---	--------------	--

APLL Fractional Bus Value Control (写/读) – 寄存器 27 (默认值 0x0000)

位编号	位名称	描述
15-0	Reserved	

Reference Clock (写/读) – 寄存器 28 (默认值 0x1800)

位编号	位名称	描述
15-14	reserved	
13-0	REF_FQ[13:0]	设置晶体频率; 13-9位 代表晶振频率的整数部分, 二进制表示速率。8-0位代表晶振频率的小数部分, 二进制表示速率。表达(bit 8 =0.5; bit 7 = 0.25; bit 6 = 0.125, ..., and so on).

Manufacture's Revision Code (只读) – 寄存器 29 (默认值0x00?0)

Bit No.	Bit Name	Description
7-4	RF_VER_ID[15:0]	该字段用于确定修改设计 0000意思为没有恢复信号; 0001意味着有恢复信号A; 0010 意味着有恢复信号 B;; 0011意味着有恢复信号C等。
3-0	Reserved	

Manufacture's ID Code LSB (只读) – 寄存器 30 (默认值0x2413)

位编号	位名称	描述
15-0	ID_CODE_L[15:0]	JEDEC JEP106-K低16位的制造的ID代码, 载有制造商, 零件编号, 和版本 The LSB 总为“1”。

Manufacture's ID Code MSB (只读) – 寄存器31 (默认值0x2184)

位编号	位名称	描述
15-12	RF_CODE_ID	
11-0	ID_CODE_M[31:16]	16位的制造的ID代码

JF24C微处理器收发寄存器定义

CONFIGURE_REG 寄存器 48 (默认值 0x5800):congig

位编号	位名称	R/W	描述	默认值
15:13	Preamble_len	R/W	000: 1个字节, 001: 2个字节 010: 3个字节, . . 111: 8个字节 注: 在发射模式,总保持为8位 “1010.” 在BPKTCTL 之前作为同步数据	010B
12:11	Syncword_len	R/W	11: 64 位 {Reg55[15:0],Reg54[15:0],Reg53[15:0],Reg52[15:0]} 10: 48位 {Reg55[15:0],Reg54[15:0],Reg52[15:0]} 01: 32位 {Reg55[15:0],Reg52[15:0]} 00: 16 位{Reg52[15:0]}	11B
10:8	Trailer_len	R/W	000: 4 位, 001: 6位 010: 8 位 011: 10 位 . . 111: 18位	000B
7:6	Data packet type	R/W	00: 归零法数据 01: 曼彻斯特数据类型 10: 8/10 行代码 11: i交错型数据	00B
5:4	FEC type(前向纠错类型)	R/W	00:没有前向纠错 01: 13位前向纠错 10: 23位前向纠错 11: 保留	
3	Power done	W	1: 微处理器设置BnPWR低于RFIC, 接着关闭晶振缓冲区	0B
2	Sleep mode	W	1: 微处理器设置BXTLEN低于RFIC,接着关闭晶振	0B
1	Reset RFIC	W	1: 单片机只是重置射频, once short.	0B
0	Fun_sel	W	1:关闭微处理器 0:开启微处理器	0B

DELAY_REG0 寄存器 49

(默认值0xC00F)

位编号	位名称	R/W	描述	默认值
15:8	BXTLEN delay time	R/W	最大延时时间?定义计算步骤 (Min~Max:0.5~2ms) 1: 8us	C0H
7	Brclk_on_sleep	R/W	0: BRCLK在休眠模式运行	0B
6	reserved			0B
5:0	BDATA1 delay time	R/W	在 BXTLAN BDATA1延时后保持为高 1:1us	0FH

DELAY_REG1 寄存器 50

(默认值 0x9628)

位编号	位名称	R/W	描述	默认值
15:8	TX_dly_tim	R/W	单片机写 reg7 发射使能, 在TX_dly_tim 时间的延时后,微处理器开始发射。单位: 1 means 1uS	96H
7:0	PA_dly_tim	R/W	功率放大延时时间, 在BPKTCTL置高后 单位: 1 means 1uS	28H

DELAY_REG2 寄存器 51

(默认值0x4000)

位编号	位名称	R/W	描述	默认值
15:8	RX_dly_tim	R/W	写单片机 reg7 接收使能, 经过RX_dly_tim延迟,微处理器通过RXCLK启动接收数据.单位: 1 means 1uS	83H
7	Miso_tri-opt	R/W	0: SPI_MISO是终止状态当SPI_SS=1时 1:SPI_MISO 保持输出	0B
6:0	Scramble_data	R/W		00H

SYNC_WORD 1 寄存器 52

(默认值0x0000)

位编号	位名称	R/W	描述	默认值
15:0	SYNC_WORD[15:0]		同步头的低有效位优先, 它符合BTSPEC.? 不同的同步头长度,这个寄存器首先被发送	0000H

SYNC_WORD 4 寄存器53

(默认值 0x0000)

位编号	位名称	R/W	描述	默认值
15:0	SYNC_WORD[31:16]		同步头的低有效位优先	0000H

SYNC_WORD 3 寄存器 54

(默认值0x0000)

位编号	位名称	R/W	描述	默认值
15:0	SYNC_WORD[47:32]		同步头的低有效位优先	0000H

SYNC_WORD 2 寄存器55

(默认值 0x0000)

位编号	位名称	R/W	描述	默认值
15:0	SYNC_WORD[63:48]		同步头的低有效位优先	0000H

Threshold_reg 寄存器56

(默认值0x4407)

位编号	位名称	R/W	描述	默认值
15:12	TX_FIFO_threshold	R/W		0100B
11:8	RX_FIFO_threshold	R/W		0100B
7	Pkt_hint_pority	R/W	1: PKF-flag/FIFO-flag 低电压有效 0:高电压有效	0B
6	addr_match_opt	R/W	地址匹配时间选项	0B
5:0	Syncword_threshold	R/W		07H

RF_CTRL 寄存器57

(默认值 0xB000)

位编号	位名称	R/W	描述	默认值
15	CRC_on	R/W	0: 关闭CRC . 1:开启CRC	1
14	Scramble_on	R/W	0: 关闭scramble . 1: 开启scramble	0
13	Pack_lenth_en	R/W	1: 有效载荷第一个字节为数据包长度	1
12	DIRECT_COTROL_MAS K	R/W	1: RF status 从关闭状态到空闲状态 由 BNPWR_PIN, BXTLEN_PIN, BDATA1_PIN控制 0: 当 REST_n = 1 , 等待一定的时间, 射频地位将自动进入空闲状态	1
11	BNPWR_PIN	R/W		0
10	BXTLEN_PIN	R/W		0
9	BDATA1_PIN	R/W		0
8	Fw_term_tx	R/W	0:单片机处理数据包长度并终止发射 1:当FIFO的写指针和读指针相等时, 微控制器将终止发射 (当单片机处理数据包长度时)	0
7:0	Crc initial data	R/W		00H

Reserve reg 寄存器58

(默认值 0x0000)

Bit	Name	R/W	Description	default
15:0	reserved	R/W		00H

Main_status寄存器 64 (只读)

位编号	位名称	R/W	描述	默认值
15:12	Framer/RFIC_st	R	BnPWR, BXTLEN, TX_EN, RX_EN 0xxx: 关闭状态 1000: 休眠状态 1100: 空闲 1110: 发射 1101: 接收 其他: 非法状态	
11	Error_ok	R	1: 在接收时发生错误 包括FEC CRC检查,它将被清除在下次开始RX/TX时	
10	Syncword_rev	R	1:收到同步头, 他只在接收状态可用 在接收状态以外, 总保持为 '0'	
9:8	reserved			
7:4	Framer TX status	R	来自硬件的发射状态机, 也可以帮助硬件调试	
3:0	Framer RX status	R	来自硬件的接收状态机	

TX_FIFO_REG 寄存器 80

位编号	位名称	R/W	描述	默认
15:0	TXRX_FIFO_REG	R/W	单片机读或写数据的FIFO	00

注意: 单片机按位访问FIFO.

FIFO RD_PTR 寄存器 82

位编号	位名称	R/W	描述	默认
15	Clr_w_ptr	W	1: 当写这一位为“1”时清TX FIFO指针为0。在接受状态不可用。	0
14:8	FIFO_WR_PTR	R	FIFO写指针	
7	Clr_r_ptr	W	1: 当写这一位为“1”时清RX FIFO指针为0。在发射状态不可用。	0
6:0	FIFO_RD_PTR	R	FIFO 读指针。	

JF24C推荐寄存器值设置表

RF 启动

寄存器地址	Read/Write	默认值 (十六进制)	推荐值 (12MHz晶振频率) (十六进制)
0x09	R/W	3003	2001
0x00	R/W	CD51	354D
0x02	R/W	137B	1F01
0x04	R/W	3CD0	BCF0
0x05	R/W	0081	00A1
0x07	R/W	0030	124C
0x08	R/W	0404	8000
0x0C	R/W	0000	8000
0x0E	R/W	6697	169B
0x0F	R/W	017B	90AD
0x10	R/W	F000	B000
0x13	R/W	2114	A114
0x14	R/W	819C	8191
0x16	R/W	0402	0002
0x18	R/W	B040	B140
0x19	R/W	7819	A80F
0x1A	R/W	6704	3F04
0x1C	R/W	1800	5800

微处理启动

寄存器地址	Read/Write	默认 (十六进制)	推荐值 (12MHz晶振频率) (十六进制)
0x30	R/W	5800	9800
0x31	R/W	C00F	FF8F
0x32	R/W	9628	8028
0x33	R/W	8300	8056
0x34	R/W	0000	4EF6
0x35	R/W	0000	F6F5
0x36	R/W	0000	185C
0x37	R/W	0000	D651
0x38	R/W	4407	4444
0x39	R/W	B000	E000

* Reg57, 如果MCU处理数据包长度并且微处理器决定FIFO空满, Reg57=0xC080

* Reg57, 如果MCU 处理数据包长度并且终止 TX , Reg57=0xC000

安阳市新世纪电子研究所有限公司

地 址：中国.河南省安阳市西环城路南 1 号

电 话：86(0372)5968708 5968993

传 真：86(0372)5968993-803

网 址：www.ayxsj.com

市 场 部：

电 话：0372--5968708-801 0372-5968993-801

E-mail: ay5968708@163.com

技 术 部：

电 话：0372--5968708-802 0372-5968993-802

E-mail: ay5968993@163.com

深圳经销商

华强电子世界

深圳圣世佳禾电子公司

地址：深圳市华强电子世界三号楼一楼 A413 柜

电话：0755-83759042 手机：13724309618

联系人：王经理

北京经销商

北京知春电子城

地址：北京市知春电子城二楼 C009 柜台

电话：010-62637297

联系人：杨 飞

郑州经销商

郑州中州商场

郑州新星电子科技

地址：郑州市中州商场三楼东区17号

电话：0371-66961968 手机：13503815523