

HR_C5000 用户手册

特性

- 1) 高性能 4FSK 调制解调
- 2) 符合 ETSI 的 DMR Tire I /II 标准的协议设计，支持物理层、数据链路层和呼叫控制层独立应用
- 3) 采用 TDMA 技术，支持全双工、半双工语音、数据通信，支持中继模式下应用
- 4) 内置高性能 10bit AD/DA
- 5) 支持 FM 调制解调
- 6) 支持模拟语音压缩、解压缩、预加重、去加重
- 7) 支持基带 IQ、可变中频 IQ、可变中频和两点调制等射频接口
- 8) 支持时分射频通道控制
- 9) 内置高性能 CodeC，支持差分 Mic 输入和 Line 输出
- 10) 支持外置 CodeC I²S 接口
- 11) 支持 CMX638 的 SPI 和 C-BUS 接口
- 12) 支持 AMBE3000 的 SPI 和 McBSP 接口
- 13) 支持 SPI 用户数据接口，可用于数字语音录音、回放及提示音输入
- 14) LQFP or VQFN 两种封装可供选择
- 15) 低功耗管理，内核 1.2V，IO 3.3V

应用：

- 1) 公众数字对讲机
- 2) 加密数字移动电台
- 3) 军用单兵数字电台
- 4) 数字集群终端
- 5) 无线本地环
- 6) 无线传感网
- 7) 无线报警系统

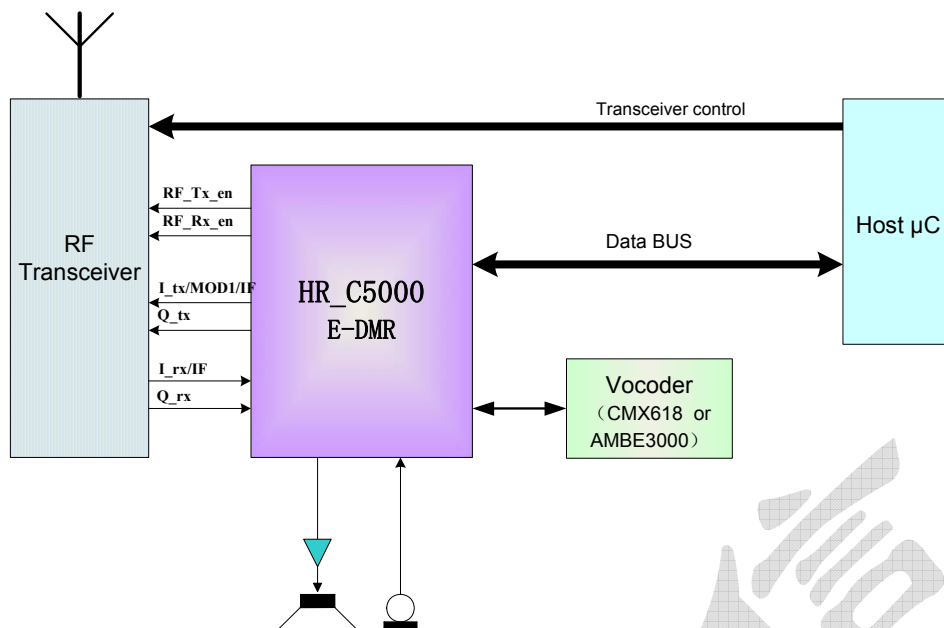


图 1

技术指标

- 系统时钟：9.8304MHz
- 数据传输速率：4.8Kbps、9.6Kbps
- 语音传输速率：4.8Kbps
- 信道间隔：数字：12.5KHz，模拟：12.5KHz/25KHz
- 频率范围：30MHz~1GHz
- 发送信噪比：57db
- 接收灵敏度：-120dbm (0.22uV, BER=5%)
- 误码率： $E_b/N_0=8\text{db}$ $\text{BER}=5 \times 10^{-2}$,
- 频偏范围： $\pm 3\text{KHz}$
- 频率稳定度要求： $\pm 3.75\text{ppm}@400\text{MHz}$
- 工作温度： $-40 \sim 125^\circ\text{C}$
- 待机功耗：20mW

简介

HR_C5000 是宏睿自主研发的符合 ETSI TS102 361 (DMR) 标准的数字对讲专用芯片。

芯片采用 4FSK 调制解调技术，在 12.5K 信道中采用 2 时隙的 TDMA 通信机制实现 2 路数字语音和数据通信传输，支持全双工、半双工通信。

芯片设计采用物理层、数据链路层和呼叫控制层分层设计，独立开放用户接口，充分开放状态信息和配置接口，方便用户进行深度二次开发，支持协议参数配置，适用于数字对讲机、专用集群终端以及低速数据、语音传输终端应用，支持中继和有中心方式下的终端应



用。

芯片采用 C-Bus 和 McBSP 接口无缝对接 CMX638、AMBE3000 等声码器芯片，同时提供标准 SPI 接口，灵活选择声码器，支持加密语音、数据接口，同时为数字语音录音、回放及提示音输入提供接口。

内置 CodeC，实现 Mic 输入和 Line 输出，提供 Mic 增益控制和 LineOut 音量控制，有效减少用户外围器件，并为外部 CodeC 配置标准 I²S 接口。

芯片内置高性能双通道 AD/DA 单元，支持基带 IQ、可配置中频 IQ、可配置中频和两点调制等射频接口，提供独立 IQ 偏置电压可调设计，IQ 路可以进行倒频设置。

采用标准模拟对讲机处理单元，支持 12.5KHz/25KHz 信道通信。

芯片采用 3.3V 供电，内置功耗管理模块，实现低功耗设计。

产品为用户提供 80 PIN LQFP 或 VQFN 封装。

框图

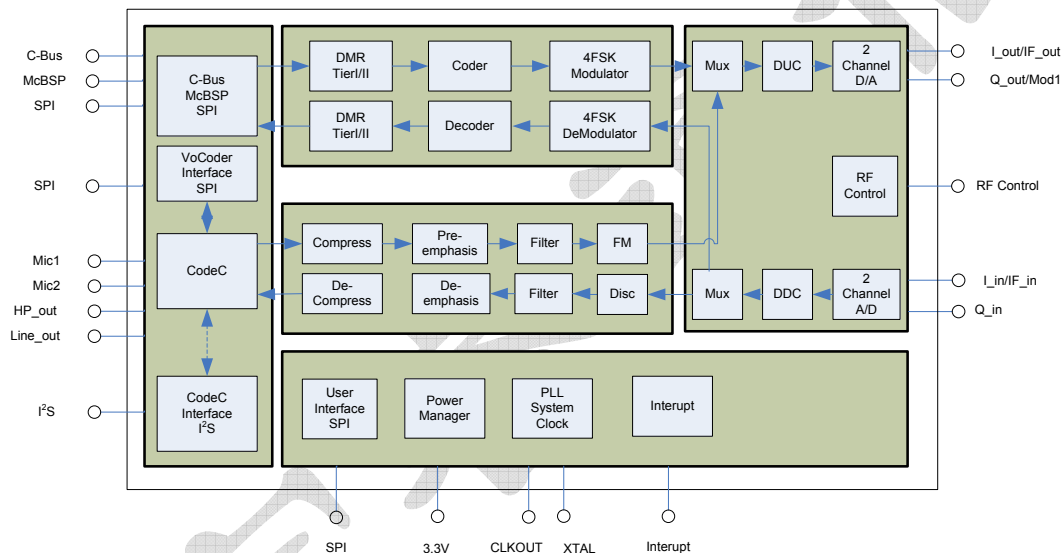


图 2

管脚定义

表 1

管脚	名字	类别	说明
1	HPVCC	AI	HP 电源 3.3V
2	HPOUT	AO	HP 输出
3	HPGND	AI	HP 地
4	CDC_VREF	AI	CodeC 的参考电平
5	MIC2_N	AI	Mic2 输入 N
6	MIC2_P	AI	Mic2 输入 P
7	MIC1_N	AI	Mic1 输入 N



8	MIC1_P	AI	Mic1 输入 P
9	CDC_AVCC	AI	CodeC 的模拟电源 3.3V
10	LINEOUT	AO	Line out
11	MICBIAS	AO	Mic 偏置电压
12	PLL_AVCC	AI	PLL 电源 3.3V
13	PLL_AVSS	AI	PLL 地
14	XTALI	DI	时钟输入
15	CKOut	DO	时钟输出
16	MCLK	DI	CodeC 时钟输入
17	ADCDAT	DI	I ² S ADC 数据（外置 CodeC 时使用）
18	BCLK	DO	I ² S BCLK
19	LRCK	DO	I ² S LRCK
20	DACDAT	DO	I ² S DAC 数据
21	McBSP_RxD	DO	AMBE3000 接口（CMX638 接口：C_CDATA）
22	McBSP_TxD	DI	AMBE3000 接口（CMX638 接口：C_RDATA）
23	McBSP_CLKR	DO	AMBE3000 接口（CMX638 接口：C_CLK）
24	McBSP_FSX	DI	AMBE3000 接口
25	McBSP_CLKX	DI	AMBE3000 接口
26	McBSP_FSR	DO	AMBE3000 接口（CMX638 接口：C_CSN）
27	PKT_RX_WAKE	DO	AMBE3000 接口
28	RTS	DI	AMBE3000 接口
29	TX_RQST	DO	AMBE3000 接口
30	TX_RDY	DI	AMBE3000 接口（CMX638 接口：C_IRQN）
31	STDBY_ENABLE	DO	AMBE3000 接口
32	DVDD	AI	数字电源 1.2V
33	V_SDI	DI	通用声码器的接口 SPI1
34	V_SDO	DO	通用声码器的接口 SPI1
35	V_SCLK	DI	通用声码器的接口 SPI1
36	V_CS	DI	通用声码器的接口 SPI1
37	C_SDI	DI	CodeC 与声码器的接口 SPI2
38	NULL		
39	DVSS	AI	数字地
40	DVCC	AI	数字 IO 电源 3.3V
41	C_CS	DO	CodeC 与声码器的接口 SPI2
42	C_SCLK	DO	CodeC 与声码器的接口 SPI2
43	C_SDO	DO	CodeC 与声码器的接口 SPI2
44	DVSS	AI	数字地
45	TESTMODE	DI	测试模式
46	RESETn	DI	复位
47	PWD	DI	休眠
48	NULL		



49	TIME_SLOT_INTER	DO	30ms 中断
50	SYS_INTER	DO	系统中断
51	RF_TX_INTER	DO	射频发送中断
52	RF_RX_INTER	DO	射频接收终端
53	U_SDO	DO	用户接口 SPI0
54	DVDD	AI	数字电源 1.2V
55	U_SDI	DI	用户接口 SPI0
56	U_CS	DI	用户接口 SPI0
57	U_SCLK	DI	用户接口 SPI0
58	RF_RX_EN	DO	射频收通道开启信号
59	RF_TX_EN	DO	射频发通道开启信号
60	DVCC	AI	数字 IO 电源 3.3V
61	NULL		
62	ADC_VBG_Q	AI	ADC 的 VBG 信号
63	ADC_QVINN	AI	Q 路 ADC 输入 N （中频 Q 路/基带 Q 路）
64	ADC_QVINP	AI	Q 路 ADC 输入 P （中频 Q 路/基带 Q 路）
65	ADC_AVDD33_Q	AI	模拟电源 3.3V
66	ADC_AGND	AI	模拟地
67	ADC_AVDD	AI	模拟电源 3.3V
68	ADC_AGND	AI	模拟地
69	ADC_AVDD33_I	AI	模拟电源 3.3V
70	ADC_IVINP	AI	I 路 ADC 输入 P （中频 I 路/基带 I 路/中频）
71	ADC_IVINN	AI	I 路 ADC 输入 N （中频 I 路/基带 I 路/中频）
72	AVC_VBG_I	AI	ADC 的 VBG 信号
73	DAC_QVOUT	AO	Q 路 DAC 输出 （中频 Q 路/基带 Q 路/中频/两点）
74	DAC_AVSS33	AI	DAC 模拟地
75	DAC_AVDD33	AI	DAC 电源 3.3V
76	DAC_IVOUT	AO	I 路 DAC 输出 （中频 I 路/基带 I 路/中频/两点）
77	DCDC_VDD12	AO	DC-DC 1.2V 输出
78	DCDC_VSS	AI	DC-DC 模拟地
79	DCDC_VDD50	AI	DC-DC 输入电源 3.3V
80	DCDC_SW	AO	DC-DC 开关输出

外围参考

待测

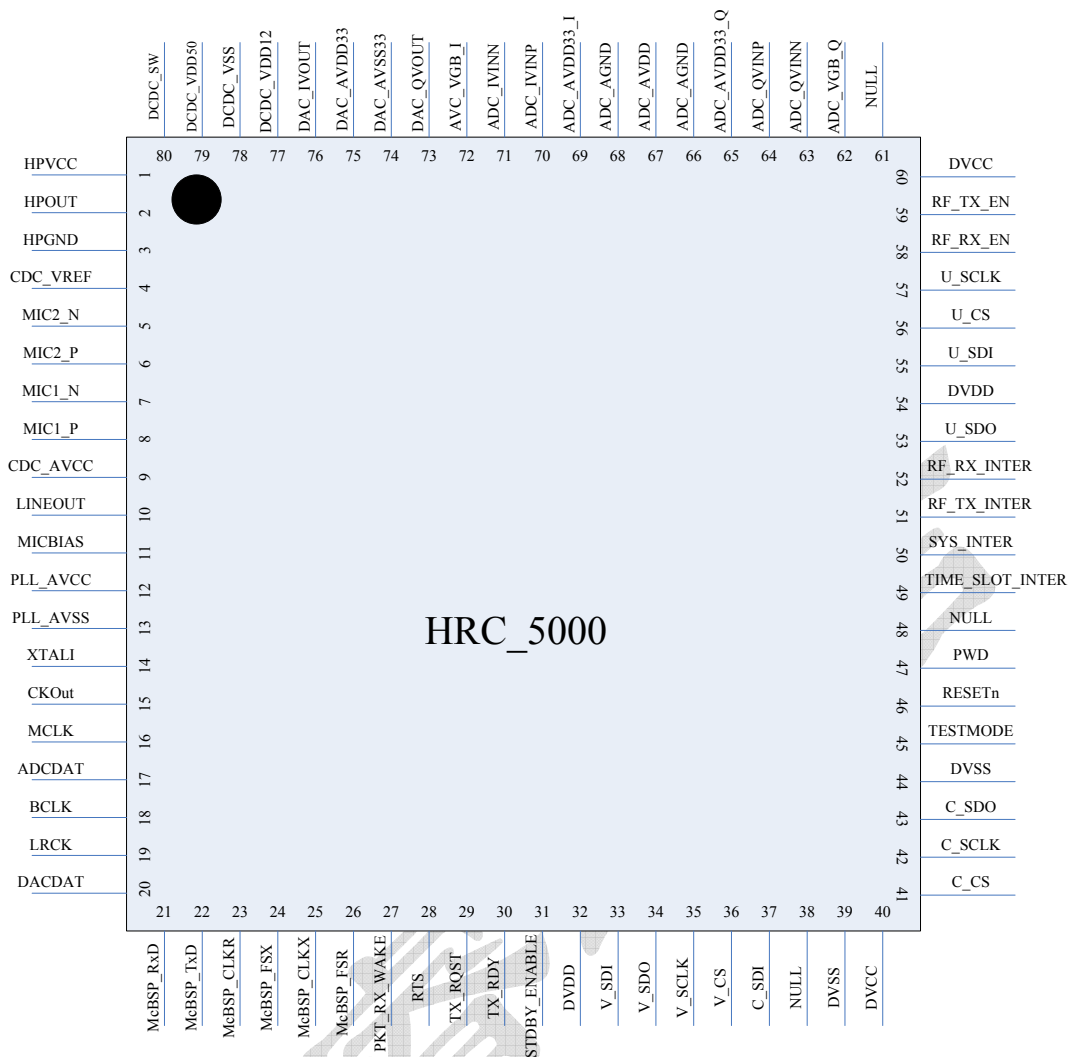


图 3

芯片介绍

1 概要

HR_C5000 芯片设计符合 ETSI 102 361 (DMR) 标准，能够实现 TierI、TierII 通信协议，进行数字语音和数据通信，可应用于数字对讲、有中心小型调度指挥系统应用、专用集群终端以及数据通信节点应用。

芯片设计采用三层分离设计，用户可以灵活应用物理层、数据链路层和呼叫控制层，实现不同的应用需求。

芯片集成高性能通道 A/D 和 D/A，在应用方案上实现与射频通道的多种接口方式，包括基带 IQ、用户可配置中频 IQ、中频以及两点调制信号，实现与模拟对讲机相同的射频方案的兼容；集成高性能语音 CodeC，应用解决方案中直接将 Mic 信号输入，模拟话音输出到外置功放或者耳机，减少用户的外围电路设计，同时也提供用户外置 CodeC 的 I²S 接口；芯片内置多种声码接口，与 CMX638、AMBE3000 实现无缝连接，可以进行芯片参数的直接配置，同时也提供 SPI 接口供用户的其它声码使用；用户可以通过 SPI 接口实现对 HR_C5000 的参数配置、数据读写以及状态参数的读写，完全开放标准的相关参数。

2 参数配置

表 2

类型	地址	读写	名称	缺省值	定义	说明
Reset	0x00	W	DMRnRst	0x00	Bit7	0 DMR 协议 reset
			PHYnRst		Bit6	0 物理层 reset
			CodernRst		Bit5	0 编解码器 reset
			FMnRst		Bit4	0 FM reset
			VoCoderRst		Bit3	0 VoCoder reset
			UDCDDCRst		Bit2	0 UDC/DDC reset
			IISRst		Bit1	0 I ² S reset
			CoderCRst		Bit0	0 CodeC reset
硬件配置	0x01	W/R	RFTransIQMode	0x14	Bit7	0 表示 IQ, 1 表示 QI
			RFRecvIQMode		Bit6	0 表示 IQ, 1 表示 QI
			RFTransMode		Bit5-Bit4	00 表示发送中频模式, 01 表示发送中频 IQ 模式, 10 表示发射基带 IQ 模式, 11 为发送两点调制模式
			RFRecvMode		Bit3-Bit2	00 表示接收中频模式, 01 表示接收中频 IQ 模式, 10 表示接收基带 IQ 模式
					Bit1	IQ 路平衡调试, 1 表示发出信号为 0 加偏置值, 0 表示为正常值
					Bit0	保留
	0x02	W/R	TranslsigCenter	0x00	Bit7-Bit0	发送 I 路的偏置值
	0x03	W/R	RecvlsigCenter	0x00	Bit7-Bit0	接收 I 路的偏置值
	0x04	W/R	TransQsigCenter	0x00	Bit7-Bit0	发送 Q 路的偏置值
	0x05	W/R	RecvlsigCenter	0x00	Bit7-Bit0	接收 Q 路的偏置值
	0x06	W/R	Vocoder638	0x40	Bit7	0 表示 CMX638 接口关, 1 表示 CMX638 接口开
			Vocoder3000		Bit6	0 表示 AMBE3000 接口关, 1 表示 AMBE3000 接口开
			DMRFrom		Bit5	0 表示从 VoCoder 输入, 1 表示从 SPI2 输入
			VocoderFrom		Bit4	0 表示协议层的输出到声码器, 1 表示从协议层的输入到声码器, 包含自环回和自定义声音
			SPIFrom		Bit3	0 表示协议层输出, 1 表示 Vocoder, 表示协议层录音或声码器录音
			CodeCMode		Bit2	0 表示内置, 1 表示外置
			OpenMusic		Bit1	1 开启, 0 关闭



			LocalVoCoderControl		Bit0	0 表示系统自动控制，1 表示 CPU 控制声码器的开关
	0x07	W/R	IFFreq2	0x0B	Bit7-Bit0	中频频率字高 8 位
	0x08	W/R	IFFreq1	0xD9	Bit7-Bit0	中频频率字中 8 位
	0x09	W/R	IFFreq0	0x54	Bit7-Bit0	中频频率字低 8 位
	0x0A	W/R	CLKOutDiv	0x02	Bit7-Bit0	对外输出时钟的分频比
	0x0B	W/R	PLLM	0x0A	Bit7-Bit0	PLL M 寄存器
	0x0C	W/R	PLLBP	0x82	Bit7	0 表示使用 PLL，1 表示 PLL bypass
			PLL SLEEP		Bit6	0 表示正常，1 表示休眠
			PLLDO		Bit5-Bit4	PLL 输出分频数
			PLL N		Bit3-Bit0	PLL 输入分频数
	0x0D	W/R	Codec Ctrl	0xC8	Bit7	选择是否由 MCU 进行 Codec 的部分信号的控制，这些 Codec 的控制信号有： Suspend, Sleep, Switch2 1: CPU 进行控制 0: 内部 Codec 控制模块进行控制
			Suspend		Bit6	CodecIP 是否进入 PowerDown 模式，高有效。 1: IP 进入 PowerDown 模式 0: IP 正常工作模式
			Sleep		Bit5	CodecIP 是否进入低功耗模式 1: IP 进入低功耗模式 0: IP 正常工作模式
			Switch2		Bit4	控制 IP 内部的 K2 开关 1: K2 Switch on 0: K2 Switch off
			AntiPop		Bit3	控制 Codec 控制电路是否开启 IP 模块的消破音处理，消破音处理在 Codec 从 PowerDown/Sleep 模式返回到正常工作模式时启动，避免产生“啪”的声音 1: 表示消破音开启， 0: 表示关闭
					Bit2-0	保留
	0x0E	W/R	HPoutEn	0x40	Bit7	0 表示 off，1 表示 on
			HPMute		Bit6	1 表示 HPout Mute
			HPoutVol		Bit5-Bit4	00 0db



						01 2db 10 4db 6db
			LineOutEn		Bit3	0 表示 off, 1 表示 on
			Mic1En		Bit2	0 表示 off, 1 表示 on
			Mic2En		Bit1	0 表示 off, 1 表示 on
			Switch1		Bit0	0 表示 off, 1 表示 on
	0x0F	W/R	ADLinVol	0xB8	Bit7-Bit3	11111: 12db 11110: 10.5db 11101: 9db ... 00001: -33db 00000: -34.5db
			MicVol		Bit2-Bit1	00: 0db, 01: 6db 10: 12db, 11: 20db
					Bit0	保留
参数配置	0x10	W/R	ModulatorMode	0x30	Bit7	0 表示 DMR, 1 表示 FM
			TierMode		Bit6	0 表示 TierI, 1 表示 TierII
			ContinueMode		Bit5	0 表示 Continue, 1 表示 TimeSlot
			LayerMode		Bit4-Bit3	00 表示物理层模式, 01 表示第二层模式, 10 表示第三层模式
			ISRepeater		Bit2	0 表示非中继, 1 表示中继
			ISAligned		Bit1	0 表示偏移, 1 表示对齐
			RepeaterSlot		Bit0	0 表示 Slot1, 1 表示 Slot2
	0x11	W/R	LocalChanMode	0x80	Bit7	置 1 有效。在中继模式下表示 RepeatedSimplex; 在直接模式下表示 DirectSimplex
					Bit6	置 1 有效。在中继模式下表示 RepeatedDuplex; 在直接模式下表示 DirectDuplex
					Bit5	置 1 有效。在中继模式下表示 RepeatedSimplex WithRC; 在直接模式下表示 DirectSimplexWithRC
					Bit4	置 1 有效。在中继模式下表示 MStoFixedDuplex
					Bit3	置 1 有效表示 SingleFreqSimplex;



					Bit2	置 1 有效。在中继模式下表示 EmbeddedOutboundRC
					Bit1	置 1 有效。表示 DedicatedOutboundRC
					Bit0	保留
	0x14	W/R	LocalSreAddressL	0x01	Bit7-Bit0	本机地址低 8Bit
	0x15	W/R	LocalSreAddressM	0x00	Bit7-Bit0	本机地址中 8Bit
	0x16	W/R	LocalSreAddressH	0x00	Bit7-Bit0	本机地址高 8Bit
	0x17	W/R	LocalGroupAddressL	0x01	Bit7-Bit0	本机所在组地址低 8Bit
	0x18	W/R	LocalGroupAddressM	0x00	Bit7-Bit0	本机所在组地址中 8Bit
	0x19	W/R	LocalGroupAddressH	0x00	Bit7-Bit0	本机所在组地址高 8Bit
	0x1A	W/R	LocalBSAddressL	0x01	Bit7-Bit0	本机所在 BS 地址低 8Bit
	0x1B	W/R	LocalBSAddressM	0x00	Bit7-Bit0	本机所在 BS 地址中 8Bit
	0x1C	W/R	LocalBSAddressH	0x00	Bit7-Bit0	本机所在 BS 地址高 8Bit
	0x1D	W/R	LocalUnaddress	0x00	Bit7-Bit4	无地址呼接收地址
			MaskUnaddr		Bit3-Bit0	无地址呼匹配码
	0x1E	W/R	LocalBroadCast	0x00	Bit7-Bit4	广播接收地址
			MaskBroadCast		Bit3-Bit0	全呼（广播）匹配码
	0x1F	W/R	LocalEMB	0x10	Bit7-Bit4	本地 CC 设置
					Bit3	PI 位设置
					Bit2-Bit0	保留
	0x20	W/R	LocalAccessPolicy	0xAA	Bit7-Bit6	00 表示 impolite_outsync, 01 表示 polite_outsync to all, 10 表示 polite_outsync to cc。
					Bit5-Bit4	00 表示 impolite_Chmonit or, 01 表示 polite_ Chmonitor to all, 10 表示 polite_Chmonitor to cc。
					Bit3-Bit2	00 表示 impolite_unkown sys, 01 表示 polite_unkown sys to all, 10 表示 polite_ unkownsys to cc。
					Bit1-Bit0	00 表示 impolite_findsync, 01 表 示 polite_findsync to all, 10 表示 polite_findsync to cc。
	0x21	W/R	LocalAccessPolicy1	0xA0	Bit7-Bit6	00 表示 impolite_notincall, 01 表 示 polite_notincall to all, 10 表示 polite_notincall to



	0x22	W/R	EncodeStart	0x00	Bit5-Bit4	00 表示 impolite_othercall, 01 表示 polite_othercall to all, 10 表示 polite_othercall to cc。
					Bit3	确认包反馈 polite 策略, 0 表示 impolite, 1 表示 polite
					Bit2-Bit0	保留
					Bit7	写入 1 表示声码器编码开始
					Bit6	写入 1 表示声码器编码结束
					Bit5	写入 1 表示声码器译码开始
					Bit4	写入 1 表示声码器译码结束
					Bit3-Bit0	保留
外置 codec 比例参数	0x30	W/R	BCLK_CNT_H	0x00	Bit7- Bit0	Mclk/BCLK 的分频比例高 8bit
	0x31	W/R	BCLK_CNT_L	0x00	Bit7- Bit0	Mclk/BCLK 的分频比例低 8bit
	0x32	W/R	LRCK_CNT_H	0x00	Bit7- Bit0	Mclk/LRCK 的分频比例高 8bit
	0x33	W/R	LRCK_CNT_L	0x00	Bit7- Bit0	Mclk/LRCK 的分频比例低 8bit
FM	0x34	W/R	FMBpfOn	0xf0	Bit7	0 表示带通滤波器关闭, 1 表示开启
			FMCompressorOn		Bit6	0 表示压缩关闭, 1 表示开启
			FMPreEmphasisOn		Bit5	0 表示预加重关闭, 1 表示预加重开启
			FMBandWidth		Bit4	0 表示 12.5KHz, 1 表示 25KHz
					Bit3- Bit0	保留
	0x35	W/R	FM_dev_coef	0xa0	Bit7-Bit4	FM 调制系数
					Bit3- Bit0	保留
	0x43	R	RSSIlevelH	0x00		RSSI 检测值高 8bit
	0x44	R	RSSIlevelL	0x00		RSSI 检测值低 8bit
DLL	0x50	W/R	LocalDataType	0x00	Bit7-Bit4	每一个发送时隙的 DataType 或 A~F



			LocalVoD		Bit3	0 表示数据，1 表示语音	
					Bit2	保留	
			LocalLCSS		Bit1-Bit0	每一个发送时隙的 LCSS	
	0x51	R	DLLRecvDataType	0x00	Bit7-Bit4	接收到的 DataType	
			DLLRecvPI		Bit3	接收到的 PI	
			DLLRecvCRC		Bit2	接收到的数据校验结果，0 为正确，1 为错误	
			SyncClass		Bit1-Bit0	00 表示无同步头，01 表示语音，10 表示数据，11 表示 RC	
0x52	R	DLLCC	0x00	Bit7-Bit4	接收到的 CC		
				Bit3-Bit0	保留		
CCL	0x60	W/R	TransControl	0x00	Bit7-Bit6	10 表示语音发送请求，01 表示数据发送请求	
					Bit5-Bit0	保留	
	0x61	W	LocalDestAdressL	0x03	Bit7-Bit0	本机呼叫目的地址低 8Bit	
	0x62	W	LocalDestAdressM	0x00	Bit7-Bit0	本机呼叫目的地址中 8Bit	
	0x63	W	LocalDestAdressH	0x00	Bit7-Bit0	本机呼叫目的地址高 8Bit	
	0x64	W/R	LocalVoiceOACSU	0x00	Bit7	0 表示 PATCS，1 表示 OACSU（仅个呼）	
			LocalHeader		Bit6	0 表示数据或语音单头，1 表示双头	
					Bit5-Bit0	保留	
	0x65	W/R	SPEn	0x00	Bit7	SP 数据包使能。	
			LocalDataConfirm		Bit6	0 表示非确认，1 表示确认	
			LocalHeaderType		Bit5-Bit2	发送数据的帧头类型（表 3），与 DPF 一致	
			LocalTransPolicy		Bit1-Bit0	数据发送速率，00 表示 rate1；01 表示 rate3/4；11 表示 rate1/2。	
	LC						
	0x66	W/R	LocalLCPF		Bit7	LC 包中的 PF	
			LocalLCR		Bit6	LC 包中的 R	
			LocalLCFLCO		Bit5-Bit0	LC 包中的 FLCO	
	0x67	W/R	LocalFID		Bit7-Bit0	FID	
	0x68	W/R	LocalLCServOption		Bit7-Bit0	Service Option	
	CSBK						
	0x69	W/R	LocalCSBKPF		Bit7		
			LocalCSBKO		Bit6-Bit1		
					Bit0	保留	
	0x6A	W/R	LocalCSBKBS1		Bit7-Bit0	CSBK 中的高 8bit，用于 Reserved	



0x6B	W/R	LocalCSBKBS2		Bit7-Bit0	CSBK 中的低 8bit，用于 Reserved
0x6C	W/R	LocalCSBKPR1			{Additionnal Information Field,Source Type,Service Type} 、 {Data/CSBK,Group/Indeivdual,Reserved}
0x6D	W/R	LocalCSBKPR2		Bit7-Bit0	Reserved 、 Answer Response、Reason Code、
IP data Header					
0x6E	W/R	LocalHeaderGI		Bit7	GI
		LocalHeaderA		Bit6	A
		LocalHeaderHC		Bit5	HC
		LocalHeaderS		Bit4	S
				Bit3	保留
		LocalLengthH		Bit2-Bit0	发送数据长度的高 3bit
0x6F	W/R	LocalLengthL		Bit7-Bit0	发送数据长度的低 7bit
0x70	W/R	LocalHeaderNS		Bit7-Bit5	N(S)
		LocalHeaderFSN		Bit4-Bit1	FSN
				Bit0	保留
0x71	W/R	LocalHeaderSAP1		Bit7-Bit4	SAP1
		LocalHeaderSAP2		Bit3-Bit0	SAP2
Proprietary data Header					
0x72	W/R	LocalHeaderMFID		Bit7-Bit0	MFID
Short Data Header					
0x73	W/R	LocalHeaderSP		Bit7-Bit5	SourcePort、DD format 高 3bit
		LocalHeaderDP		Bit4-Bit2	DestinationPort、DDformat 低 3bit
		LocalHeaderSP1		Bit1-Bit0	SP 高 2bit、S\F
0x74	W/R	LocalHeaderSP2		Bit7-Bit0	SP 低 8bit
UDT data Header					
0x75	W/R	LocalHeaderUDTF		Bit7-Bit4	UDT Format
				Bit3-Bit0	保留
0x76	W/R	LocalHeaderSF		Bit7	SF
		LocalHeaderPF		Bit6	PF
		LocalHeaderUDTO		Bit5-Bit0	UDTO
DataTerminator LC					
0x77	W/R	TD_LC_PF		Bit7	TD_LC 包中的 PF
		TD_LC_R		Bit6- Bit5	TD_LC 包中的 R
		TD_LC_S		Bit4	TD_LC 包中的 S
		TD_LC_NS		Bit3- Bit1	TD_LC 包中的 NS
				Bit0	保留



	0x78	R	RxLengthL		Bit7- Bit0	接收数据长度低 8 位
	0x79	R	RxLengthH		Bit7- Bit3	保留
					Bit2- Bit0	接收数据长度高 3 位
	0x7A	W/R	RespnStatus		Bit7- Bit6	反馈包头 Class 信息位
					Bit5- Bit3	反馈包头 Type 信息位
					Bit2- Bit0	反馈包头 Status 信息位
	0x7B	R	ConfirmStatus		Bit7- Bit6	提供给 cpu 当前接收的 ACK 状态。00 表示 ACK，01 表示 NACK，10 表示 SACK。
					Bit5- Bit3	提供给 cpu 当前接收的 NI 数目
					Bit2- Bit0	保留
中断						
	0x82	R	InterRequestDeny		Bit7	1 表示发送请求拒绝
			InterSendStart		Bit6	1 表示发送开始
			InterSendStop		Bit5	1 表示发送结束
			InterLateEntry		Bit4	1 表示后接入
			InterRecvData		Bit3	1 表示接收数据
			InterRecvMessage		Bit2	1 表示接收信息
			InterQuit		Bit1	1 表示退出
			InterPHYOnly		Bit0	1 表示物理层单独工作接收中断
	0x83	W	InterClear		Bit7- Bit0	按照比特位清空相应的中断寄存器。
	0x84	R	SendStartVoice		Bit7	语音发送
			SendStartCSBK1		Bit6	OACSU 请求首次发送
			SendStartCSBK2		Bit5	OACSU 请求第二次发送
			Over_VocoderRespon		Bit4	Vocoder 配置返回中断
			SendStartData		Bit3	数据发送
			SendStartDataPart		Bit2	数据部分重传
			SendStartDataFull		Bit1	数据全部重传
					Bit0	保留
	0x85	W/R	SendStartMask		Bit7- Bit0	SendStart 中断的使能
	0x86	R	SendStopVoice		Bit7	语音正常结束
			SendStopVoiceErr		Bit6	语音异常结束（发送超时）
			SendStopOACSUExp		Bit5	语音 OACSU 等待超时
					Bit4	保留
			SendStopData		Bit3	数据正常结束
			SendStopDataErr		Bit2	数据异常结束（发送超时）
			SendStopConfigExp		Bit1	数据确认等待超时
					Bit0	保留

	0x87	W/R	SendStopMask		Bit7-Bit0	SendStop 中断的使能
	0x90	R	RecvDataCRC		Bit7	数据接收中的整个 Fragment 的 32bit CRC 校验结果, 0 为正确, 1 为出错。

3 数据链路层方式

3.1 协议支持

3.1.1. 流程控制

独立支持 direct 和 repeater 模式;

支持 Aligned 模式和 Offset 模式的设定及相应处理;

严格按照 ETSI TS102361-1 设置 MS 的信道接入流程, 包括 direct 和 repeater 模式;

计时器参数可设;

Polite 策略可设;

Hold off 可设;

状态指示;

3.1.2. 时隙组帧

时隙组帧而言, 有 3 种模式: 语音时隙包、数据时隙包和 RC 包。

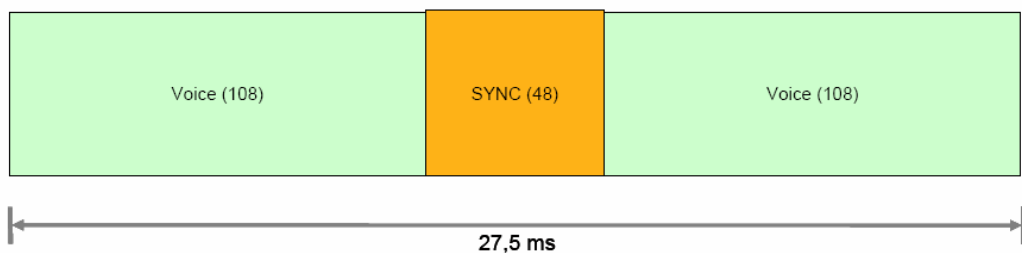


图 4

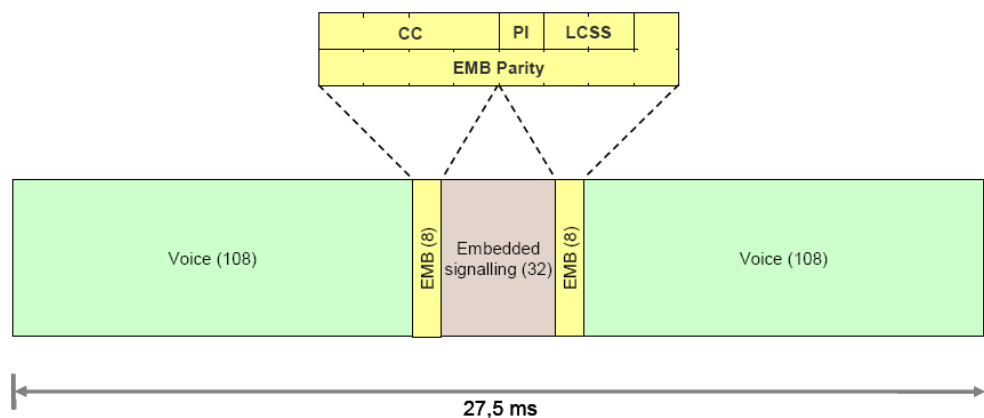


图 5

支持语音时隙包的超帧组帧, 按照 A(SYNC)、B(LC)、C(LC)、D(LC)、E(LC)、F(Null)进行超帧组合。

支持同步头选择加入;

支持 EMB 7bit 加入，对 EMB 进行 QR(16,7,6)编码；

支持 LC 72bit 加入，加入 5bit CS 码，进行可变长度的 BPTC 编码、交织，并加入到 4 个时隙中（128bit）；

支持 Null 时隙加入；

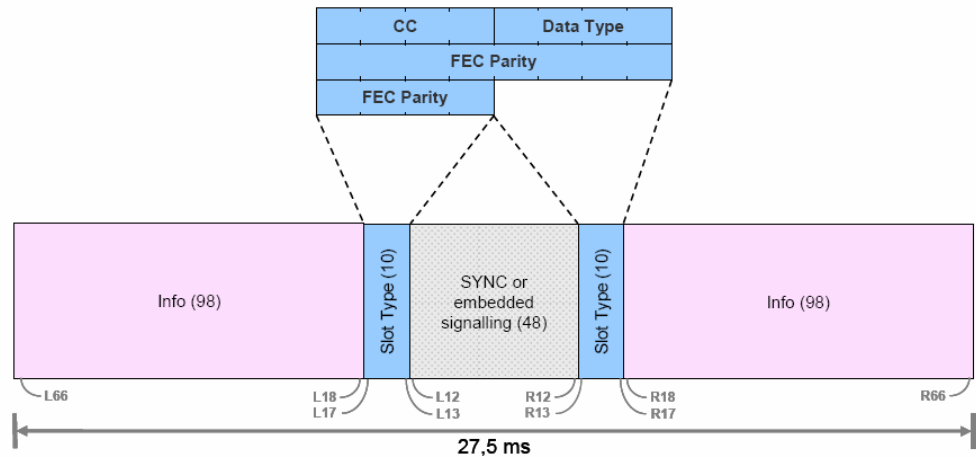


图 6

支持 LC 包、CSBK 包、MBC 包和数据包；

支持加入 Slot Type (20bit)，包括 CC、Data Type，进行 Golay (20, 8) 编码；

支持 SYNC、EMB 加入；

支持加入 72bit LC, CRC24bit 校验，并加入 CRC 掩码(Header 和 Terminator 区别)，进行 BPTC (196, 96) 编码；

支持加入 80bit CSBK, CRC16bit 校验并加入 CRC 掩码，进行 BPTC (196, 96) 编码、交织；

支持加入 96bitIdle，进行 BPTC(196, 96)编码、交织；

支持加入 80bit MBC header, CRC16bit 校验，加入 CRC 掩码，进行 BPTC (196, 96) 编码、交织；

支持加入 96bit MBC 数据，进行 BPTC (196, 96) 编码、交织；

支持加入 80bitMBC lastblock，进行 CRC16bit 校验，进行 BPTC (196, 96) 编码、交织；

支持数据包头，加入 80bit 数据，进行 CRC16bit 校验，加入 CRC 掩码，进行 BPTC (196, 96) 编码、交织；

支持根据应用需求快速生成 Unconfirmed data header、Confirmed data header、Response data header、Proprietary data header、Status/Precoded short data header、Raw short data header、Defined short data header 和 Unified Data transport data header；

支持 Rate 1/2 模式的数据格式，加入 96bit 数据，进行 BPTC (196, 96) 编码、交织；

支持 Rate 1/2 模式的最后时隙数据，加入 64bit 数据，进行 32bitCRC 校验（校验包含所有数据），进行 BPTC (196, 96) 编码、交织；

支持 Rate 3/4 模式的数据格式，加入 96bit 数据，进行 Trellis 编码、交织；

支持 Rate 3/4 模式的最后时隙数据，加入 64bit 数据，进行 32bitCRC 校验（校验包含所有数据），进行 Trellis 编码、交织；

支持 Rate 1 模式的数据格式，加入 96bit 数据；

支持 Rate 1 模式的最后时隙数据，加入 64bit 数据，进行 32bitCRC 校验（校验包

含所有数据)；

支持 3 种速率的 confirmed 数据传输，加入 7bit SN，进行 9bitCRC 校验，加入掩码（不同速率，不同掩码），进行不同速率的编码、交织，对数据加入 32bitCRC；

支持反馈包数据时隙，加入 1—2 个数据反馈包，进行整体 32bitCRC 校验，进行 BPTC (196, 96) 编码、交织；

支持 UDT 的 last block，对数据进行 16bitCRC 校验，进行 BPTC (196, 96) 编码、交织；

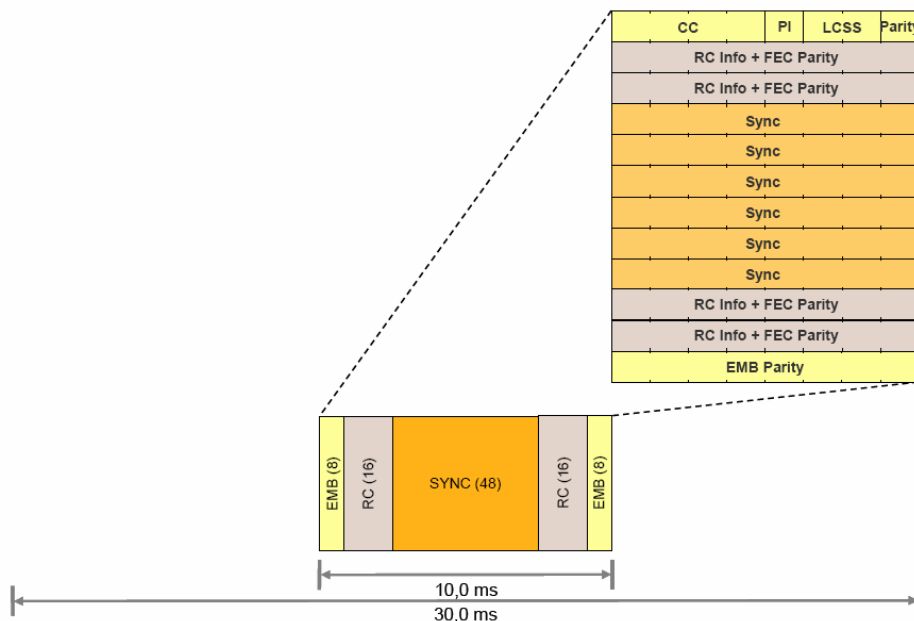


图 7

支持时隙 RC 和 EMB 的 RC 信号；

支持加入 7bitEMB，进行 QR(16,7,6)编码；

支持加入 11bit RC 信号，进行可变长度 BPTC，对 32bit 进行交织，加入到 RC 单元；

接收根据 SYNC 确定帧的内容类型，根据 Slot Type、FLCO、CSBKO、LB、DPF 确定接收帧的类型，然后根据接收到的帧类型进行与发送相对应的解交织、解码、校验。

3.1.3. 模式组帧

连续模式：

支持语音超帧组帧，可设 LC Header、PI Header 或仅 PI Header 模式，结束自动构成超帧并加入 LC Terminator；

支持数据 4.8kbps 组帧，加入数据 LC Header、数据和 LC Terminator；

支持数据 9.6kbps 组帧，加入数据 LC Header、数据和 LC Terminator；

时隙模式：

支持语音超帧组帧，可设 LC Header、PI Header 或仅 PI Header 模式，结束自动构成超帧并加入 LC Terminator；

支持各种数据类型的 4.8kbps 组帧，加入数据 LC Header、数据和 LC Terminator；

3.2 使用方法

表 3

时隙帧类型	编码	是否语音
语音 LC Header	0001	0
语音 PI Header	0000	0
语音 A	0001	1
语音 B	0010	1
语音 C	0011	1
语音 D	0100	1
语音 E	0101	1
语音 F	0110	1
语音 Terminator	0010	0
CSBK	0011	0
MBC Header	0100	0
MBC Intermedia	0101	0
MBC Last	0101	0
DataHeader	0110	0
DataRate1_2	0111	0
DataRate3_4	1000	0
DataRate1	1010	0
Idle	1001	0
Reserved	1011	0
Reserved	1100	0
Reserved	1011	0
Reserved	1110	0
Reserved	1111	0

首先配置基本参数：见表 3 中 DLL。

系统初始化要求处于被动接收状态，DLL 在独立工作时，仅提供 RF_Level 信号，CPU 建立状态机，芯片一直接收信号，如果有同步信息，将会产生同步信号 trans_slot_c。

如果在提出发送请求时芯片尚未同步，CPU 将读取 RSSI 值，判断后决定是否能够发送，如果已经建立同步，CPU 将读取是否可以发送（不可以发送的情况取决于在配置好的 Polite 策略条件下的状态转换）。

如果可以发送，将开启以下情况：

主动模式：系统没有收到任何同步信号，并且 RSSI 低，DLL 建立 trans_slot_c，以此为中断，向 CPU 提供 30ms 中断。

被动模式：系统接收到空中信号，并且允许发送，那么根据收到的 recv_slot_c，开始建立同步，产生 trans_slot_c，并且根据 recv_slot_c 来调整 trans_slot_c，以此提供给 CPU 30ms（近似）中断。

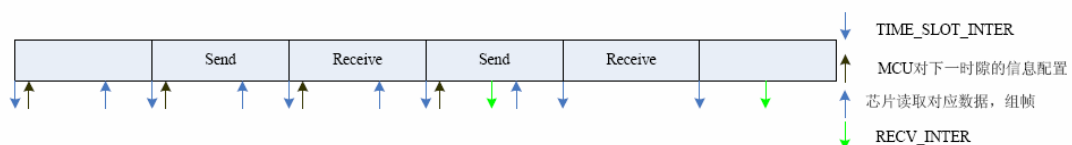


图 8

同步建立后, CPU 会收到 30ms 中断, 由 CPU 决定下一个时隙芯片的收发状态和收发类型, 并且提供每一个时隙的 DataType、PI、LCSS、发送类型和收发使能, 发送类型主要有:

- 语音 LC Header:

CPU 在 0x50 设定发送类型 0x10, 写入 72bit LC 数据, 写入发送使能, 芯片做完 CRC、BPTC、交织和组帧后在下一个时隙发送, 芯片内部开启语音编码, 可以 CPU 控制开启语音编码。

接收则相反, 在接收中断中给出 DataType、CC、PI 和 72bit LC 数据 (或 CRC 校验错误)。

- 语音 PI Header:

CPU 在 0x50 设定发送类型 0x00, 写入 80bit 数据, 写入发送使能, 芯片做完 Header CRC、BPTC、交织和组帧后在下一个时隙发送。

接收中断中给出 DataType、CC、PI 和 80bit 数据 (或 CRC 校验错误)。

- 语音超帧 (实际操作时, 采用 A~F 分开控制即可):

在语音超帧之前, 必须将 LC 72bit 配置, 用于产生超帧中的 EMB, 识别并记录芯片给出的中断, 在每一个定位的 (A~F) 中断中, 给出相应的 CC、PI 和 LCSS, 芯片从内部获取语音编码数据 (也可以通过 CPU 给入语音编码后数据), 并且组帧在下一时隙发送。

接收将从超帧中获得 LC 信息后, 将 CC、PI 和 LC 提供给 CPU, 语音数据可以通过 RAM 送给 CPU, 也可以直接进入译码器的 RAM 中, 在收到 A 帧后, 自动 (或 CPU 控制) 开启语音译码。

语音超帧的 A~F 由 0x50 中设置 0x18~0x68

- 语音 LC Terminate:

CPU 在 0x50 设定发送类型 0x20 (可以不写入 72bit 数据), 写入发送使能, 芯片做完 CRC、BPTC、交织和组帧后在下一个时隙发送。

在接收中断中给出 DataType、CC、PI 和 72bit LC 数据 (或 CRC 校验错误), 并且关闭语音解码 (可以 CPU 控制关闭)。

DLL 状态机制在处于失锁状态后, 也会关闭语音解码 (可以 CPU 控制关闭), 芯片发出异常失锁中断。

- CSBK (80bit):

CPU 在 0x50 设定发送类型 0x30, 写入 80bit 数据, 写入发送使能, 芯片做完 CRC、BPTC、交织和组帧后在下一个时隙发送。

接收中断中给出 DataType、CC、PI 和 80bit 数据 (或 CRC 校验错误)。

- MBC Header:

CPU 在 0x50 设定发送类型 0x40, 写入 80bit 数据, 写入发送使能, 芯片做完 CRC、BPTC、交织和组帧后在下一个时隙发送。

接收中断中给出 DataType、CC、PI 和 80bit 数据 (或 CRC 校验错误)。

- MBC Intermediate/ MBC Last (96bit):

CPU 在 0x50 设定发送类型 0x50, 写入 96bit 数据, 写入发送使能, 芯片做完 BPTC、交织和组帧后在下一个时隙发送。

接收中断中给出 DataType、CC、PI 和 96bit 数据。

- Data Header (80bit):

CPU 在 0x50 设定发送类型 0x60, 写入 80bit 数据, 写入发送使能, 芯片做完 CRC、BPTC、交织和组帧后在下一个时隙发送。

接收中断中给出 DataType、CC、PI 和 80bit 数据 (或 CRC 校验错误)。

- **Data Block** (96/144/192bit 要求包含 Data block SN、9bit CRC 和 Fragment CRC):
CPU 在 0x50 设定发送类型 0x70/0x80/0xA0, 写入 96/144/196bit 数据, 写入发送使能, 芯片做完 BPTC/Viterbi/no、交织和组帧后在下一个时隙发送。
接收中断中给出 DataType、CC、PI 和 96bit 数据。
- **自定义帧:**
CPU 在 0x50 设定发送类型 0xB0~0xF0, 写入 96bit 数据, 写入发送使能, 芯片做完 BPTC、交织和组帧后在下一个时隙发送。
接收中断中给出 DataType、CC、PI 和 96bit 数据。
注: 如果 SlotType 区域校验不通过, 将被列为无效的数据, 丢失, 但是芯片会告诉上层有丢失。

4 呼叫控制层方式

4.1 协议支持

支持协议规定的 SFID 和协议 Layer3 规定的标准字段控制, 主要提供的服务有:
支持采用 CSBK 激活 BS 的过程处理

此过程在 MS 发起中继呼叫时自动处理。

支持 FNS

Feature Not Supported signaling, 采用 Response CSBK, 设定相应的值进行反馈。

组呼 (Group Call)

支持采用 LC 中确定 FID 和 FLCO 的方式进行组帧, 在 CCL 层设计时隙符合标准的群呼状态流转, 配套时隙 PTT 事件、信道申请传输拒绝、信道申请传输接受、语音接收、延迟接入的语音接收、PTT 释放、语音终结 LC 传输、收到 Idle 语音终结、非本身接收语音退出终结和无信号接收终结。

个呼 (Individual Call)

支持直接的个呼和 OACSU 呼叫;

支持 OACSU 呼叫的 CSBK 建立和 CSBK 反馈信号控制流程, 支持反馈信号的异常处理;

无地址呼 (unaddress voice Call)

支持对于无地址呼方式的地址识别, 流程控制;

全呼 (all call)

支持对于全呼方式的地址识别, 流程控制;

广播 (broadcast voice call)

支持广播地址的呼叫方式, 流程控制;

开放信道呼 (open voice channel call)

支持 Group Call 和 Individual Call 下采用 OVCM 模式进行呼叫和接收呼叫;

采用 ETSI TS 102 361-2 标准规定的各个 LC、PDU 控制字的规定, 并严格按照流程设计, 时间参数可设, 采用协议规定的 Polite 策略;

数据信息服务

非确认式信息 (unconfirmed data transmission)

支持 Rate 1/2、Rate 3/4、Rate1 方式的非确认式信息收发, 可以设定用于自定义的数据形式, 加入特定的 Header (Proprietary Header);

确认式信息 (confirmed data transmission)

支持 Rate 1/2、Rate 3/4、Rate1 方式的确认式信息收发, 也可以支持用户自定义的数



据形式，加入特定的 Header，按照协议设计确认式信息收发的流程和异常处理；

支持可变长度的滑动窗口设计；

短信（Short Data）

支持 Defined Data、Raw Data、Status/precoded Data 短息数据的非确认式和确认式的收发方式，参照协议规定 PDU 单元数据；

采用 ETSI TS 102 361-2 标准规定的各个 LC 、PDU 控制字的规定，并严格按照流程设计，时间参数可设，采用协议规定的 Polite 策略；

4.2 使用说明

CCL 模式下的基本配置包括硬件配置，参数配置（其中 0x10 配置为 8'b01x10xxx），主要发起以下请求：

1) 组呼

组呼设计组帧包括 Voice LC Header、Voice A~F、LC Terminator。

用户必须配置的内容：FLCO、Service Option、Group address、Source address。

用户可配置的内容：

时隙： PI

LC： PF、R

掩码： Voice LC Header、Terminator with LC

接入 Polite 策略。

相关寄存器：

0x66: 8'bxx000000

0x68: 参见表 4

表 4

0x61,0x62,0x63: 组呼目的地址

0x64: 8'b0x000000，如果 x=1，则需要向 0x80 地址写入 PI 头的 96bit 内容即 12byte。

0x1F: Bit3 （Option）

0x67: default （Option）

Index 0x10~0x17: default （Option）



0x20, 0x21: default (Option)

用于过程控制的配置：通话请求 (0x60= 8'b10000000)，开始语音发送。可以同时开启声码器 ()，如果允许 CPU 控制声码器 (0x06 Bit0=1)，通话结束 (0x60= 8'b00000000)；

发送端中断：

通话请求拒绝中断。根据发送请求，状态机在经过条件判断后，发出请求拒绝中断，CPU 可以根据此中断决定是否继续进行尝试；

通话开始中断。在状态机接受发送请求后，语音开始进行 LC Header 发送，向 CPU 提出通话开始中断，CPU 可以根据此中断进行通话中显示及计时；

通话结束中断，包括正常、异常情况；异常是指自身内部原因如发送时长超过限制，导致通话结束，正常则指 CPU 发起通话结束请求后，芯片进行 LC Terminator 发送时提出的正常中断。该中断产生后，CPU 可以关闭界面显示，统计时长，可以关闭声码器（如果允许 CPU 控制声码器），该中断也可以自动关闭声码器。

接收端中断：

Voice LC Header 收到中断。该中断提供 DataType、PI、CRC 校验结果 (0x51) 和 CC (0x52)，如果正确则提供 72bit LC 数据，可以从 0x81 地址读取。

后接入中断。后接入中断是在接收到 Voice 中的 EMB 信息，并且整合并且进行校验通过后提供给 CPU 的中断，因此该中断直接提供 72bit LC，可以从 0x81 地址读取。

在收到以上两个中断之一后，CPU 可以开启语音译码器（如果允许 CPU 控制声码器），也可以由芯片自动开启译码器，同时可以进行通话中界面显示，开始计时。

Voice LC Terminator 中断。该中断在接收到一个 Voice LC Terminator 后发出，CPU 可以根据此中断获得 DataType、PI、CRC 校验结果 (0x51) 和 CC (0x52)，如果正确则提供 72bit LC 数据，CPU 根据此中断情况判断结束为正常或者异常，控制界面退出通话界面，结束计时，可以由 CPU 关闭声码器译码（如果允许）。

Lost 中断。该中断为芯片在失锁后提出的，一般为异常通话退出情况，CPU 根据此中断判断通话结束，控制界面退出通话，结束计时，可以由 CPU 关闭声码器译码（如果允许）。

2) 个呼

➤ PATCS

PATCS 组帧包括 Voice LC Header、Voice A~F、LC Terminator。

用户必须配置的内容：FLCO、Service Option (不可广播)、Group address、Source address
用户可配置的内容：

时隙： PI

LC： PF、R

掩码： Voice LC Header、Terminator with LC

接入 Polite 策略。

相关寄存器：

0x66: 8'bxx000011

0x68: 参见表 4

0x61,0x62,0x63: 个呼目的地址

0x64: 8'b0x000000, 如果 x=1, 则需要向 0x80 地址写入 PI 头的 96bit 内容即 12byte。

0x1F: Bit3 (Option)

0x67: default (Option)

Index 0x10~0x17: default (Option)

0x20, 0x21: default (Option)

用于过程控制的配置: 通话请求 (0x60= 8'b10000000), 开始语音发送。可以同时开启声码器(), 如果允许 CPU 控制声码器 (0x06 Bit0=1), 通话结束 (0x60= 8'b00000000);

发送端中断:

与组呼一致。

接收端中断:

与组呼一致。

➤ OACSU

发送端:

OACSU 组帧包括: CSBK、Voice LC Header、Voice A~F、LC Terminator。

用户必须配置的内容: CSBKO、Service Option (不可广播)、FLCO、Target address、Source address

用户可配置的内容:

时隙: PI

CSBK: PF、R

LC: PF、R

掩码: Voice LC Header、Terminator with LC、CSBK

接入 Polite 策略, 确认后, 需要 impolite 策略进行语音发送。

相关寄存器:

0x69: 8'bx0001000

0x6D: 8'b00000000

0x66: 8'bx000011

0x68: 参见表 4

0x61, 0x62, 0x63: 个呼目的地址

0x64: 8'b0x000000, 如果 x=1, 则需要向 0x80 地址写入 PI 头的 96bit 内容即 12byte。

0x1F: Bit3 (Option)

0x67: default (Option)

Index 0x10~0x17: default (Option)

0x20, 0x21: default (Option)

用于过程控制的配置: 通话请求 (0x60= 8'b10000000), 开始语音发送。可以同时开启声码器(), 如果允许 CPU 控制声码器 (0x06 Bit0=1), 通话结束 (0x60= 8'b00000000);

发送端中断:

通话请求拒绝中断。根据发送请求, 状态机在经过条件判断后, 发出请求拒绝中断, CPU 可以根据此中断决定是否继续进行尝试。

CSBK 请求包发送中断。CPU 可以利用该中断显示请求发送中;

接收到反馈包中断 (接受/拒绝)。芯片在该中断中提供 DataType、PI、CC 和 CRC 校验结果, 如果正确则提供 80bit CSBK 数据, CPU 利用该中断显示对方是否接受本机的呼叫。

等待超时, 并进行第二次 CSBK 请求包发送中断。CPU 利用该中断显示发起第二次请求包;

接收到反馈包中断 (接受/拒绝); 芯片在该中断中提供 DataType、PI、CC 和 CRC 校验结果, 如果正确则提供 80bit CSBK 数据, CPU 利用该中断显示对方是否接受本机的呼叫。



第二次等待超时中断；CPU 利用该中断表示对方无应答。

以下与组呼一致。

接收端中断：

CSBK 接收到中断，该中断提供 DataType、PI、CC 和 CRC 校验结果，如果正确则提供 80bit 数据；数据可以从 0x81 读取，如果接收正确，CPU 将利用 CSBK 中的信息，并且根据本机对源地址的接受或拒绝信息来发出反馈包。

以下中断与组呼一致。

接收端：

反馈包中设置内容：P、ServiceOption、Answer Response、Target address、Source address。

相关寄存器：

0x69：8'bx0001010

0x6C：8'bxxxxxxxx

0x6D：8'b0010000x

0x61,0x62,0x63：反馈目的地址

3) 全双工

全双工处理与个呼一致，仅在 Service Option 中设定一种保留的字段用于识别全双工。

作为 OACSU 模式下，全双工仅需要单向 OACSU，在识别通信模式为全双工条件下，反向发起呼叫时 PATCS。

4) 无地址呼

无地址呼为组呼一类，呼叫组呼地址限定为 0xFFFFE0~0xFFFFEF，unaddress (16 个)，在接收机而言，需要设定本机能够接收的 unaddress，其可采用 4bit 匹配码，确定本机可以接收的 unaddress。相关寄存器为 0x1D。

如果本地设置无地址呼地址为 0xFFFFE1，匹配码为 0xF 时，

5) 全呼

全呼为组呼一类，呼叫组呼地址限定为 0xFFFFF0~0xFFFFF7。接收机对于全呼地址，可以采用匹配码进行设定，可以完全匹配或者多个匹配。

6) 广播

广播为组呼一类，呼叫组呼地址限定为全呼地址 0xFFFFF0~0xFFFFF7，在此地址条件下，要求设定 Service Option 为广播。接收机对于广播地址，可以采用匹配码进行设定，可以完全匹配或者多个匹配。

7) OVCM

OVCM 模式是指发送端将本次发送的呼叫（组呼、单工个呼）设定为开放式呼叫，用户接收到 LC 后，识别内部的 OVCM 信息，如果该比特为 1，则可以对信道上存在的呼叫进行监听（不需要匹配地址），如果需要进行回答，同样设置 OVCM 比特为 1，发起方也同样可以听到回复方的通话。

需要注意的是，在双工模式下或者是在无地址呼、全呼模式下以及在数据传输过程中均不支持 ovcm 模式。在这些模式时，用户配置该 bit 未视为无效配置。

通话中产生的中断与组呼一致。

无地址呼、全呼归为特殊地址的组呼，全呼和广播区别在于 Service Option 的设置，OVCM 则在个呼和组呼中均可使用，关键在于设置 ServiceOption 中的 OVCM。

8) BS 激活

BS 激活设计组帧仅一个 CSBK，BS 激活过程是在中继模式下，由发起各种呼叫时，BS



未被激活条件下的一个必然过程，无需软件配置启动和结束。但用户可配置字段中的保留字段：0x1A、0x1B、0x1C 为 BS 的地址，0x6A、0x6B 为 BS 中的 16bit 保留字段。

Pre_CSBK:采用二层技术解决。

NAR CSBK: 采用二层技术解决。

语音 **PI** 的模式未明，可以采用二层技术解决。

MBC 模式未明，可以采用二层技术解决。

保留 **DataType** 问题，因涉及到每一个时隙，因此由二层技术解决。

对于数据部分采用相同的收发机制：

- 非确认，单头
- 确认，单头
- 非确认，双头
- 确认，双头

帧头类型如表 5，字段内容根据表 6 进行用户自定义。每次发送以 **Fragment** 为单位进行控制。

表 5

注：Status/Precoded header 由配合输入的信息长度为 0 决定。

表 6

类型							
公共	G/I	A	SAP1	SAP2	Target address		
IP 数据	HC		F	S	N(S)	FSN	
Status/P		SP	DP	S/P	S/P		
Raw		SP	DP	S/F			
Defined		DD		S/F			
UDT	UDTFormat			SF	PF	UDTO	

9) 非确认 IP:

IP 包设计包括的帧为：Unconfirmed data Header、Proprietary data Header、Non-last data block、Last data block。

用户必须配置的内容：G/I、HC、SAP1、SAP2、Destination address

用户可配置的内容：

Proprietary: 双头时使用，SAP1 在单头时应用到相应的帧头中，在双头时，Proprietary Header 采用 SAP2，Proprietary 帧内容待定。

接入 Polite 策略。

掩码：Data Header。

相关寄存器：

0x65: 8'bx00010xx
0x6E: 8'bx0x00xxx
0x6F: 8'bx00000000
0x70: 8'b00000000
0x71: 8'bx00000000
0x61,0x62,0x63: 目的地址

0x72: 8'bx00000000 (Option)

用于过程控制的配置: 发送请求 (0x60= 8'b10000000), 无需进行发送结束请求。

发送端中断:

发送请求拒绝中断; 根据发送请求, 状态机在经过条件判断后, 发出请求拒绝中断, CPU 可以根据此中断决定是否继续进行尝试;

发送开始中断; CPU 接收到此中断可以显示短信发送中。

发送结束 (包括正常、异常; 异常是指自身内部原因)。CPU 根据发送结果显示相应的发送情况。

接收端中断:

Data Header 接收到中断; 提供 DataType、PI、CRC 校验结果和 CC, 如果校验正确提供 80bit 数据, 从 0x81 地址读取, CPU 可以从 DataHeader 信息中获取信息来源和类型等信息;

Proprietary 接收到中断; 提供 DataType、PI、CRC 校验结果和 CC, 如果校验正确提供 80bit, 从 0x81 地址读取 (注意读取数据长度为 20byte, PI 数据开始位置为 11), 用户可以从信息中获取用户自定义的信息内容;

接收到数据中断; 该中断根据 DataHeader 中的信息, 接收相应的数据长度, 经过整体数据的 CRC 校验后, 提供数据或 CRC32 校验错误;

10) 非确认 Raw data:

Raw data 包设计包括的帧为: Raw short data Header、Proprietary data Header、Non-last data block、Last data block。

相关寄存器:

0x65: 8'bx01110xx
0x6E: 8'bx0x00xxx
0x6F: 8'bx00000000
0x73: 8'bx00000000
0x74: 8'bx00000000
0x61,0x62,0x63: 目的地址

0x72: 8'bx00000000 (Option)

用于过程控制的配置: 发送请求 (0x60= 8'b10000000), 无需进行发送结束请求。

中断与 IP 包一致。

11) 非确认 Status/Precoded data:

Status/Precoded data 包设计包括的帧为: Status/Precoded data Header。

相关寄存器:

0x65: 8'bx01110xx
0x6E: 8'bx0x00000
0x6F: 8'b00000000

0x73: 8'xxxxxxxx

0x74: 8'xxxxxxxx

0x61,0x62,0x63: 目的地址

0x72: 8'xxxxxxxx (Option)

其他与 IP 包一致，在接收中断中，因为 S/P data 没有数据包内容，因此不提供数据到达中断。

12) 非确认 Defined data:

Defined data 包设计包括的帧为：Defined Header、Non-last data block、Last data block。

用户必须配置的内容：G/I、DD Format、Destination address

相关寄存器：

0x65: 8'bx01101xx

0x6E: 8'bx0x00xxx

0x6F: 8'xxxxxxxx

0x73: 8'xxxxxxxx

0x74: 8'xxxxxxxx

0x61,0x62,0x63: 目的地址

0x72: 8'xxxxxxxx (Option)

其他与 IP 包内容一致。

13) 非确认 Unified Data Transport:

UDT data 包设计包括的帧为：UDT Header、Non-last data block、Last data block。

用户必须配置的内容：G/I、DD Format、Destination address、UAB、SF、PF、UDTO

相关寄存器：

0x65: 8'bx00000xx

0x6E: 8'bx0x00xxx

0x6F: 8'xxxxxxxx

0x75: 8'xxxxxxxx

0x76: 8'xxxxxxxx

0x61,0x62,0x63: 目的地址

0x72: 8'xxxxxxxx (Option)

其他与 IP 包一致。

14) 确认式数据:

发送端配置而言，与非确认式一致，仅 Abit 会被设置成 1，即以上相关的寄存器中，0x6E 设置为 8'bx1x00xxx。

对于接收端而言，反馈包的组帧和信息内容由芯片自动完成。

增加发送端中断：

收到反馈包中断，提供 80bit 或校验错误，反馈包的内容基本上由芯片自身利用，除非反馈包表明接收完全正常，信息发送完成；

重传（部分重传、全部重传）中断；CPU 获得该中断后，显示当前进行重传（部分、全部）。

最后发送结束（成功、失败）中断；根据反馈包（或超时）芯片通知 CPU 短信发送成

功或者失败。

增加接收端中断:

接收重传包包头中断, 提供 80bit 或校验错误;

接收到重传包数据, 或提供错误数据;

15) 速率配置:

短数据的速率配置选择寄存器 0x65 LocalTransPolicy 的 Bit1-0。00 表示 rate1; 01 表示 rate3/4; 11 表示 rate1/2。

16) TierI 支持:

配置 TierMode (0x10 Bit6) 值为 0, 实现 TierI 的传输模式, 在该模式下, ContinueMode 和 IsRepeater 均无效, 系统自动配置为连续模式, 并且没有中继模式。语音、数据均为双时隙发送, 两个时隙发送相同的信息。

17) 中继/非中继模式:

中继模式配置选择寄存器 0x10 Bit2 ISRepeater 配置为 0 表示非中继模式, 1 为中继模式。

非中继模式下, 如果 ContinueMode 为 1 (即时隙模式), 芯片采用时隙发, 时隙收, 同时反映到 RF_TX_EN (27.5ms) / RF_RX_EN (27.5ms) 和 RF_TX_INTER (60ms 间隔) / RF_RX_INTER (60ms 间隔);

中继模式下, 如果选择时隙模式, 系统采用时隙发, 连续收, 反映到 RF_TX_EN (27.5ms) / RF_RX_EN (连续) 和 RF_TX_INTER (60ms 间隔) / RF_RX_INTER (1 次), 在全双工通信下, 芯片仍采用时隙发, 连续收, 但反映到射频控制信号为时隙收发 RF_TX_EN (27.5ms) / RF_RX_EN (30ms) 和 RF_TX_INTER (60ms 间隔) / RF_RX_INTER (60ms 间隔)。

18) 连续模式支持:

在 TierII 模式下, 通过配置 0x10 Bit5 为 0 实现连续模式的发送或接收。连续模式仅支持数据传输。数据传输为 9.6kbps 的连续模式。发送机可以通过配置决定, 但是接收机需要利用帧头进行控制, 在获得是连续模式信息后, 将接收设定为连续模式, 从而获得两个时隙的数据并进行联合。

注: 非标准模式下, 仅对个别字段用户能够进行定义, 接收到此类信息后, 将全部直接提交给上层做识别。

5 FM 制式兼容

FM 制式兼容是指针对模拟对讲机的应用, 支持语音带内信号处理包括压缩、解压缩、预加重和去加重应用技术, 完全符合 TIA/EIA-603 标准。

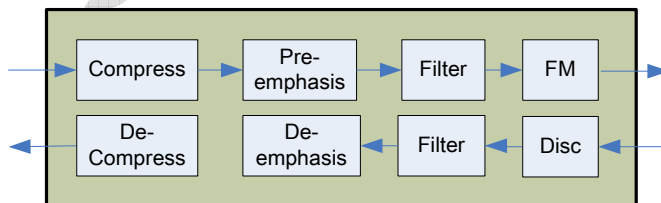


图 9

通过配置 0x34 寄存器, 开关滤波器、预加重/去加重、压缩/解压缩, 实现 FM 的自定义功能。

6 系统时钟

系统内部时钟采用 49.152MHz，为 DMR 数字部分提供 9.8304MHz，FM 部分提供 8.192MHz，CodeC 部分提供 12.288MHz，同时为外部 CodeC 提供可能的 24.576MHz。

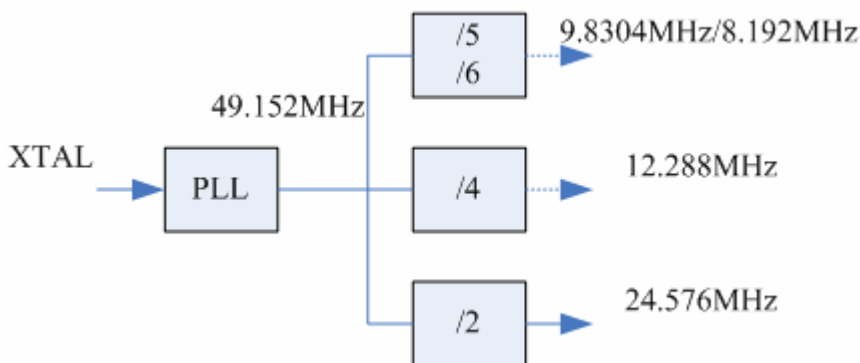


图 10

系统时钟采用 0x18、0x19 进行配置，具体计算公式为：

$$\text{CLK} = \text{XTAL} \times \text{PLLM} / \text{PLLN} / \text{NO}$$

其中 $\text{NO} = 2^{\text{PLLDO}}$

7 中断处理

中断包括 4 个中断，其中包括两个收发切换中断，1 个 30ms 的时钟中断和 1 个系统中断。在正常模式下，如果射频模块不需要进行收发切换的配置，那么只需要 1 个系统中断就可以进行正常的工作处理。

1) 收发切换中断

根据 DMR 标准，在 TDMA 工作条件下，射频通道需要进行 30ms 为单位的收发切换，其产生条件是，开机产生接收中断，在建立同步条件下，开始进行 30ms 的收发切换中断，非发即收。

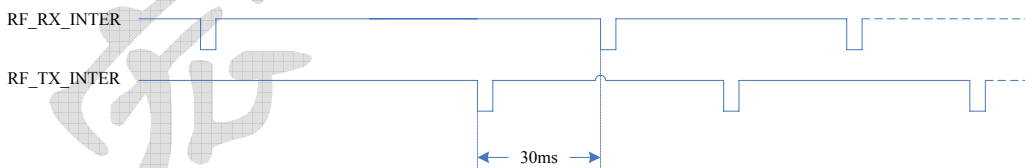
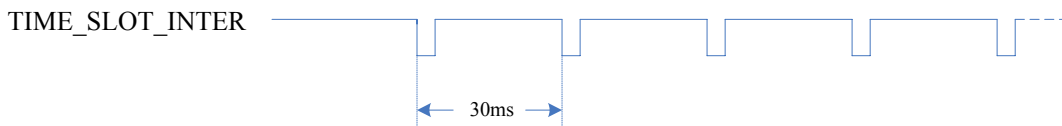


图 11

2) 30ms 中断

30ms 中断产生于工作在物理层模式和数据链路层模式，CPU 利用该中断，决定下一个时隙是否发送，以及发送的内容。



3) 系统中断

系统中断管脚为 SYS_INTER，在收到该中断后，CPU 读取中断状态寄存器 0x82，

获得 8 个类型的中断，包括：

➤ 发送请求拒绝

该中断没有子状态寄存器，表明本次发送请求因为信道忙而被拒绝。

➤ 发送开始

发送开始有子状态寄存器 **0x84**，可以通过 **0x85** 将相应的中断屏蔽掉。子状态寄存器中表明了 7 种产生发送开始的中断，包括：

✧ 语音发送

✧ OACSU 请求首次发送

✧ OACSU 请求第二次发送

✧ **Vocoder 配置返回中断**（本中断为采用 CPU 配置 AMBE3000 时，由 **HR_C5000** 向 CPU 发回的配置完成中断）

✧ 数据发送

✧ 数据部分重传

✧ 数据全部重传

➤ 发送结束

发送结束有子状态寄存器 **0x86**，可以通过 **0x87** 将相应的中断屏蔽掉。子状态寄存器中表明了 6 种产生发送结束的中断，包括：

✧ 语音正常结束

✧ 语音异常结束（发送超时）

✧ 语音 OACSU 等待超时

✧ 数据正常结束

✧ 数据异常结束（发送超时）

✧ 数据确认等待超时

➤ 后接入

后接入中断没有子状态寄存器，在收到该中断后，表明接入的语音通信方式是后接入方式。

➤ 接收数据

接收数据中断没有子状态寄存器，但是其接收数据的对错，接收类型由 **0x51** 寄存器给出，利用 **DLLRecvDataType**、**DLLRecvCRC** 说明收到的数据类型和对错情况，CPU 据此进行相应的状态显示，也可以屏蔽相应的中断。

➤ 接收信息

接收信息中断有子状态寄存器 **0x90**，子状态寄存器仅有一种类型，0 表示整个信息接收校验通过，1 表示整个信息接收校验错误。

➤ 异常退出

异常退出中断没有子状态寄存器，产生异常的原因是状态机内部产生的非预期中断。

➤ 物理层单独工作接收中断

物理层单独工作接收中断没有子状态寄存器，该中断产生于物理层单独工作模式下，接收到数据后产生该中断，通知 CPU 读取相应的寄存器获得接收数据。

除了系统中断外，其它中断均不需要进行清除处理，系统中断的处理方式如下：

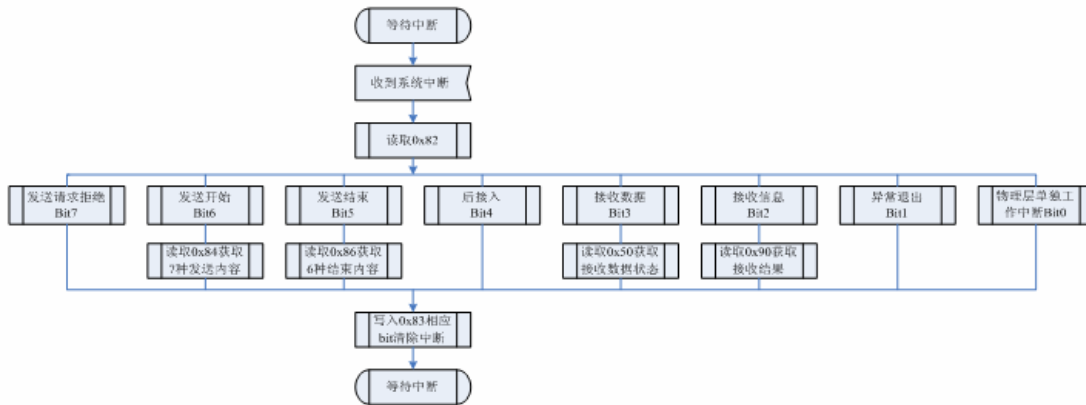


图 12

8 数据读写

芯片配备有 SPI0 和 SPI1 两个用于数据读写的接口，SPI0 功能包括参数配置、状态/数据读写、附属参数配置、CMX638 参数配置、AMBE3000 参数配置、提示音数据写入，SPI1 用于通用声码器语音数据读写。

1) SPI 接口读写时序

Cmd	Addr	Data0	Data1...	Datan
-----	------	-------	----------	-------

图 13

Cmd 是用于识别读写和区分不同功能和选择 Ram 空间。

对于 SPI0

表 7

Cmd	W	IsRead		Bit7	1 表示本次操作为读，0 表示本次操作为写
				Bit6-Bit3	保留
		OPMode		Bit2-Bit0	000 表示操作状态控制 Ram，001 表示操作附属参数配置表，010 表示操作信息数据，011 表示操作提示音，100 表示对 CMX638 参数配置，101 表示对 AMBE3000 参数配置。

对于 SPI1

表 8

Cmd	W	IsRead		Bit7	1 表示本次操作为读，0 表示本次操作为写
				Bit6-Bit3	保留
		OPMode		Bit2-Bit0	011 表示操作声码语音数据

Addr 是本次读写的初始地址，在后续进入（或读取）的数据，将从该初始地址开始，并且逐个累加，在每一次 CS 有效情况下，将持续累加。



Data0-Datan 是每一次读写的数据，最大长度为 2048 Byte。

2) 状态控制 Ram 数据读写

状态控制 Ram 如参数配置表所示，地址空间为 0x00~0xFF，其中包含了硬件参数配置、状态控制参数配置和中断信息读取等。

3) 附属参数配置表读写

附属参数配置表是将一些非常用参数归类为附属参数，包括了协议中的一些常量设置和定时参数设置，其地址空间为 0x00~0xFF。

4) 信息数据读写

信息数据包括物理层独立工作时的 36byte 数据读写，地址空间为 0~35；二层工作模式时，信息数据长度为 9byte (LC)、10byte (CSBK、Header)、12byte (自定义、Rate1/2)、19byte (Rate3/4) 和 25byte (Rate1)，其地址分布空间为 0~47；三层工作模式时，信息数据长度为 9byte (LC)、10byte (CSBK、Header)，其地址空间为 0~15，另考虑到采用双头信息的情况，其最大长度为 12byte (PI)，地址空间为 16~27，三层中的短消息数据存储空间为 48~2047。

表 9

地址	物理层模式	二层模式	三层模式
0x00~0x09	物理层数据	LC	LC
0x0A		CSBK、Header	CSBK、Header
0x0B~0x0C		自定义、Rate1/2	空余
0x0D~0x0F		Rate3/4	
0x10~0x12		Rate1	PI
0x13~0x18			
0x19~0x1C			
0x1D~0x23			
0x24~0x2F			
0x030~0x7FF			短信息

5) 提示音数据写入

提示音是利用 CPU 通过 SPI0 写入数据到 CodeC (内置外置皆可)，要求的 cmd=0x03，addr=0x00，后面跟两个 8bit 数据，写入时间间隔为

6) CMX638 配置

配置前，先等待 HR_C5000 上电后对 CMX638 的初时化配置完成。配置时，HR_C5000 不能对 CMX638 有其他操作，比如读取 CMX638 的编码语音数据。通过 SPI0 与 Cmx638 互通，SPI0 操作格式如下：

表 10

操作	Cmd (1byte)	Addr (1byte)	Data0 (1byte)	Data1 (1byte)	Data2 (1byte)	Data3 (1byte)
写 Cmx 寄存器	0x04	0x00	Cmx 寄存器地址	Cmx 寄存器数据 1	Cmx 寄存器数据 2	Bit7-Bit4=4'b1111, Bit3-Bit0= 要写的 Cmx 寄存器长度 (0/1/2)
读 Cmx 寄存器 第一步	0x04	0x00	Cmx 寄存器地址	0x00	0x00	Bit7-Bit4=4'b1110, Bit3-Bit0= 要读的 Cmx 寄存器长度



器							(1/2)
	第二步	0x84	0x00	Cmx 寄存器数据 1	Cmx 寄存器数据 2		

说明:

1 写 Cmx 寄存器时, 如果这个寄存器只有地址, 则 Data1 与 Data2 都写 0x00, Data3 写 0xF0。

如果这个寄存器数据为 1byte, 则 Data1 写相应数据, Data2 写 0x00, Data3 写 0xF1。

如果这个寄存器数据为 2byte, 则 Data1 与 Data2 写相应数据, Data3 写 0xF2。

2 读 Cmx 寄存器分为两步。首先写 Cmx 寄存器地址, 等待相应的中断产生, 再读 Cmx 寄存器数据。

3 每次操作时间间隔为 2ms。

7) AMBE3000 配置

配置前, 先等待 HR_C5000 上电后对 AMBE3000 的初始化配置完成。初始化配置有: 关闭控制包末尾的奇偶校验, 开启低功耗模式。配置时, HR_C5000 不能对 AMBE3000 有其他操作。通过 SPI 与 AMBE3000 互通, SPI 操作格式如下:

表 11

操作	Cmd (1byte)	Addr (1byte)	Data0 (1byte)	...	Datan (1byte)
写控制包	0x05	0x00	整个控制包数据 n<=19		
读返回的数据包	0x85	0x00	返回的整个数据包 n<=9		

说明:

1 写入的控制包如果有返回的数据包, 则等待相应的中断产生, 可以读取返回的数据包。

2 写入的控制包如果没有返回的数据包, 则中断不产生。

8) SPI1 语音数据读写

语音数据将从 SPI1 中读写, CPU 在知道进行语音通信情况下, 利用声码器部分的线路参数配置, 通过通话开始和结束状态的捕捉, 定时对 SPI1 进行读写操作, 可以获得和声码器一致的接口和数据。

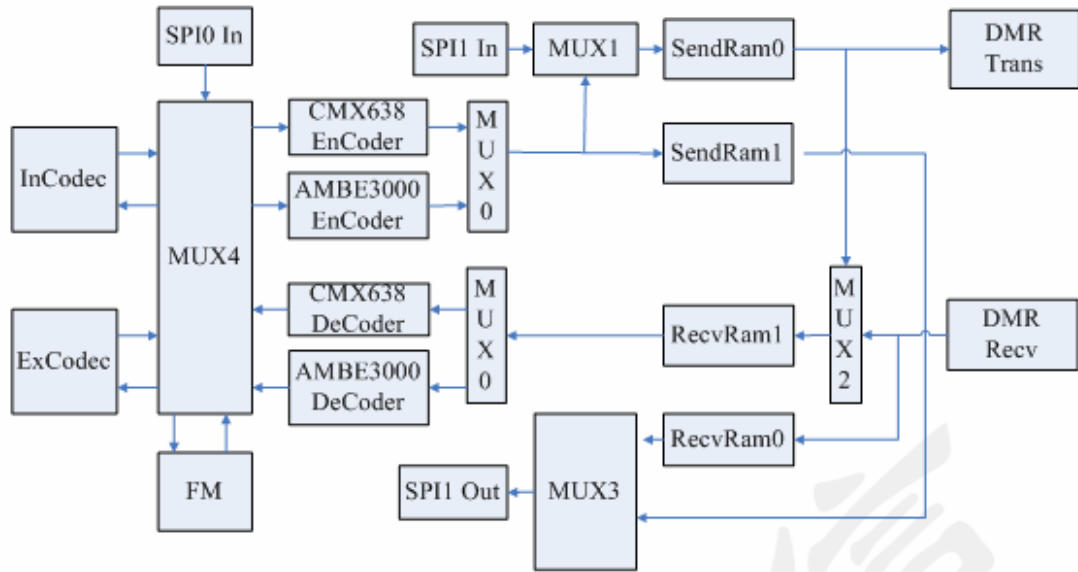


图 14

语音数据读写操作的 cmd=0x03/0x83, addr=0x00。

配置参数通过 SPI0 进行操作。

表 12

类型	地址	读写	名称	缺省值	定义	说明
硬件配置	0x06	W/R	Vocoder638		Bit7	0 表示 CMX638 接口关， 1 表示 CMX638 接口开
			Vocoder3000		Bit6	0 表示 AMBE3000 接口关， 1 表示 AMBE3000 接口开
			DMRFrom		Bit5	0 表示从 VoCoder 输入， 1 表示从 SPI1 输入
			VocoderFrom		Bit4	0 表示协议层的输出到声码器， 1 表示协议层的输入到声码器， 包含自环回和自定义声音
			SPIFrom		Bit3	0 表示协议层输出， 1 表示 Vocoder 输出， 表示协议层录音或声码器录音
			CodeCMode		Bit2	0 表示内置， 1 表示外置
			OpenMusic		Bit1	0 表示关， 1 表示开
			LocalVocoderControl		Bit0	0 表示系统自动控制， 1 表示 CPU 控制
参数配置	0x10	W/R	ModulatorMode		Bit7	0 表示 DMR， 1 表示 FM

通路说明：

MUX0 由[0x06][7:6]控制，其中[0x06][7:6]不能等于 2'b11，即 CMX638 与 AMBE3000 不能同时打开。(如强制等于 2'b11，效果与 2'b00 相同)

MUX1 由[0x06][5]控制，
当=0，MUX0>>MUX1>>SendRam0，
当=1，SPI1_In>>MUX1>>SendRam0

MUX2 由[0x06][4]控制，
当=0，DMRRecv>>MUX2>>RecvRam1，表示环回，
当=1，SendRam0>>MUX2>>RecvRam1，表示自定义声音

MUX3 由[0x06][3]控制，
当=0，RecvRam0>>MUX3>>SPI1_Out，表示协议层录音，
当=1，SendRam1>>MUX3>>SPI1_Out，表示声码器录音

MUX4 由[0x06][2:1]和[0x10][7]控制，其中[0x06][1]的优先级最高，其次是[0x10][7]，
当[0x06][1] = 1，表示开机、来电、来信铃声，
SPI2>>MUX4>>InCodec\ExCodec。
当[0x06][1]=0，[0x10][7]=1，Codec>>MUX4>>FM
与 FM>> MUX4>> Codec
当 [0x06][1] = 0 ， [0x10][7] = 0 ，
Codec>>MUX4>>Vocoder 与 Vocoder>> MUX4>>
Codec

9 声码器外围说明

芯片支持 AMBE3000 和 CMX638，内置声码器和外置可选，语音从 Mic 输入，经过 CodeC 后，通过 SPI2 接口输出给声码器，声码器压缩后的数据通过复用的数据线（McBSP/C-Bus）送入 HR_C5000 发送。接收下来的数据从 HR_C5000 通过复用数据线送给声码器，声码器将解压缩后的数据通过 SPI2 送给 CodeC，CodeC 经过 DA 后输出给 Speaker。

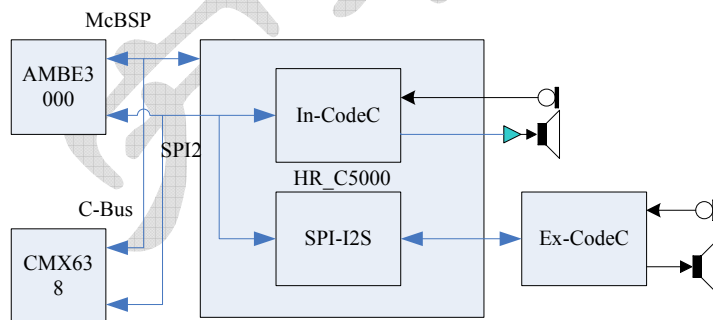


图 15

10 外置 CodeC 的时钟

外置 CodeC 的 I²S 读写时钟速率可以通过 0x30、0x31、0x32、0x33 配置。