无线让生活更美好

Better Life with Wireless

Application Note

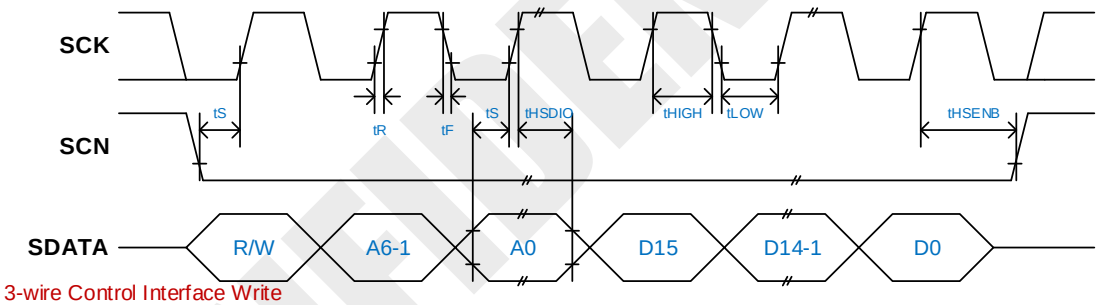
For BK4819 High Performance Walkie-Talkie Transceiver IC

BEKEN CORPORATION

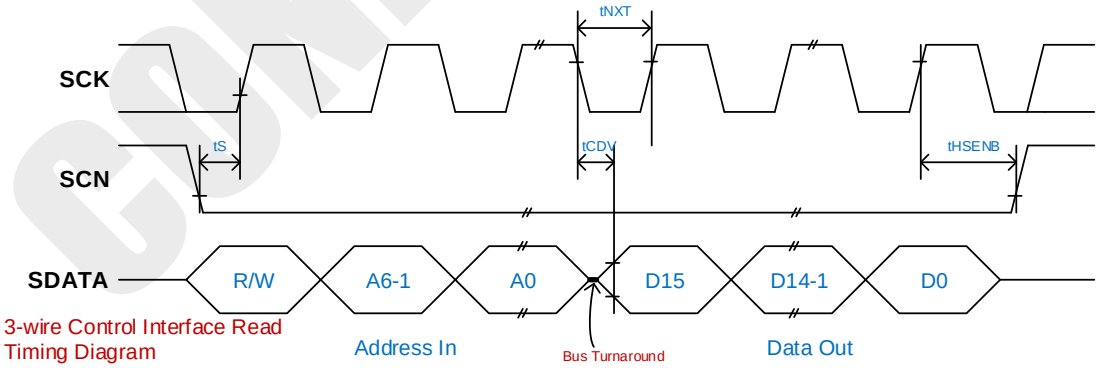
Content Index

1. MCU Interface - 3 Wire SPI

Parameter	Symbol	Min.	Typ.	Max.
SCK Frequency	*fSCK*	*0 MHz*	*—*	*8 MHz*
SCK High Time	*tHIGH*	*25 ns*	*—*	*—*
SCK Low Time	*tLOW*	*25 ns*	*—*	*—*
SDATA Input, SCN to SCK↑ Setup	*tS*	*20 ns*	*—*	*—*
SDATA Input to SCK↑ Hold	*tHSDATA*	*10 ns*	*—*	*—*
SCN Input to SCK↓ Hold	*tHSCN*	*10 ns*	*—*	*—*
SCK↓ to SDATA Output Valid	*tCDV*	*2 ns*	*—*	*25 ns*
SCK↓ to next SCK↑ after Address In	*tNXT*	*1 us*	*—*	*—*
SCK, SCN, SDATA, Rise/Fall Time	*tR,tF*	*—*	*—*	*10 ns*



3-wire Control Interface Write Timing Diagram



3-wire Control Interface Read Timing Diagram

2. Reigster Initialization

芯片上电后进行 Soft Reset、内部 Power Up 和其他初始化设置。使用 RF_Initial()，在 RF_Initail()里根据需求设置静噪门限、接收音量、接收 AGC、MIC 灵敏度、VoX 门限、调制深度、发射功率、亚音频等。

3. Tx/Rx Audio

1) 初始化后默认状态为正常 Speech 模式，可进行语音（300~3kHz）收发。

2) 开启压扩需要设置 RF_EnterCompander()，关闭压扩使用 RF_ExitCompander()

Register	Default	Description
REG_31<3>	*0*	*Enable Compander Function.* *1= Enable; 0=Disable*
REG_28<15:14>	*0b01*	*Expander (AF Rx) Ratio.* *00=Disable; 01=1:2; 10=1:3; 11=1:4*
REG_28<13:7>	*0x56*	*Expander (AF Rx) 0 dB point(dB)*
REG_28<6:0>	*0x38*	*Expander (AF Rx) noise point(dB)*
REG_29<15:14>	*0b10*	*Compress (AF Tx) Ratio.* *00=Disable; 01=1.333:1; 10=2:1; 11=4:1*
REG_29<13:7>	*0x56*	*Compress (AF Tx) 0 dB point(dB)*
REG_29<6:0>	*0x40*	*Compress (AF Tx) noise point(dB)*
REG_6F<6:0>	*Read Only*	*AF Tx/Rx Input Amplitude(dB)*

3) 开启扰频需要设置 RF_EnterScramble()，关闭扰频使用 RF_ExitScramble()

Register	Default	Description
REG_31<1>	*0*	*Enable Scramble Function.* *1=Enable; 0=Disable*
REG_71<15:0>	*0x8517*	*Scramble/Tone1 Frequency Control Word.* *=3300(Hz)* 10.32444 for XTAL 13M/26M or* *=3300(Hz)* 10.48576 for XTAL* *12.8M/19.2M/25.6M/38.4M.* *- The scrambler inversion mixing frequency should be* *kept between 2.6kHz and 3.5kHz*

4) 滤波器开关使能

Register	Default	Description
REG_2B<10>	*0*	*Disable AF Rx HPF300 filter.* *0=Enable; 1=Disable*
REG_2B<9>	*0*	*Disable AF Rx LPF3K filter.* *0=Enable; 1=Disable*
REG_2B<8>	*0*	*Disable AF Rx de-emphasis filter.* *0=Enable; 1=Disable*
REG_2B<2>	*0*	*Disable AF Tx HPF300 filter.* *0=Enable; 1=Disable*
REG_2B<1>	*0*	*Disable AF Tx LPF1 filter.* *0=Enable; 1=Disable*
REG_2B<0>	*0*	*Disable AF Tx pre-emphasis filter.* *0=Enable; 1=Disable*
REG_43<8:6>	*0b001*	*AF Tx LPF2 filter Band Width (Apass=1dB) Selection.* *100 = 4.5 kHz* *101 = 4.25 kHz* *110 = 4 kHz* *111 = 3.75 kHz* *000 = 3 kHz (for 25k Channel Space)* *001 = 2.5 kHz (for 12.5k Channel Space)* *010 = 2.75 kHz* *011 = 3.5 kHz*

5) 音频响应调整使用 RF_SetAfResponse(u8 tx, u8 f3k, u8 db)，其中参数 tx=1 发射/tx=0 接收，f3k=1 调整 3kHz/f3k=0 调整 300Hz，db=调整范围-1~+4dB。例如：
发射 300Hz 增大 2dB：RF_SetAfResponse(1,0,2);
接收 3kHz 减小 3dB：RF_SetAfResponse(0,1,-3);

6) 接收音量设置

Register	Default	Description
REG_48<11:10>	*0b00*	*AF Rx Gain1.* *00=0dB;01=-6dB;10=-12dB;11=-18dB*
REG_48<9:4>	*0x3C*	*AF Rx Gain2.* *-26dB~5.5dB, 0.5dB/step.* *0x00=mute*
REG_48<3:0>	*0b1111*	*AF DAC Gain (after Gain1 and Gain2).* *1111=max; 0000=min; about 2dB/step*

7) 发射调制设置及发射 mute

Register	Default	Description
REG_40<12>	*1*	*Enable RF Tx Deviation.* *1=Enable; 0=Disable*
REG_40<11:0>	*0x4D0*	*RF Tx Deviation Tuning (Apply for both in-band signal and sub-audio signal).* *0=min; 0xFFF=max*
REG_50<15>	*0*	*Enable AF Tx Mute (for DTMF Tx or other applications).* *1=Mute; 0=Normal*

8) MIC 灵敏度设置

Register	Default	Description
REG_7D<4:0>	*0x10*	*MIC Sensitivity Tuning.* *0x00=min; 0x1F=max; 0.5dB/step*

9) AF 输出选择使用 RF_SetAf(u8 mode)，产生本地按键音、提示音可参考 RF_Key()，产生铃音并发射可参考 RF_Call()

4. CTCSS/CDCSS

1) 开启 CTCSS 需要设置 RF_SetCtcss()和 RF_SetCtc2()，其中后者仅用于接收频率 55Hz（或其他 100Hz 以内频率）的 CTCSS 尾音；前者用于接收和发射正常 CTCSS。

2) 开启 CDCSS 需要设置 RF_SetCdcss()，需要设置 134.4Hz 码率和 CDCSS 码。

3) 关闭亚音频使用 RF_ExitSubau()

4) 发射结束时产生尾音使用 RF_GenTail()，相位尾音使用参数 CTC120/CTC180/CTC240，如 RF_GenTail(CTC180)；换频尾音（如 55Hz）使用参数 CTC55，如 RF_GenTail(CTC55)；在 CDCSS 模式下产生 134.4Hz 尾音使用参数 CTC134，如 RF_GenTail(CTC134)

5) 读取 CTCSS 状态使用 RF_GetCtcss()，返回 1 表示收到 CTC1（主 CTC），返回 2 表示收到 CTC2（如 55Hz 尾音）；读取 CDCSS 状态使用 RF_GetCdcss()，返回 1 表示收到 CDC 正码，返回 2 表示收到 CDC 反码；读取相位尾音状态使用 RF_GetTail()，返回 1 表示收到 120°相位变化尾音，返回 2 表示收到 180°相位变化尾音，返回 3 表示收到 240°相位变化尾音。

Register	Default	Description
REG_51<15>	*0*	*1=Enable Tx CTCSS/CDCSS; 0=Disable*
REG_51<14>	*0*	*1= GPIO6 Input for CDCSS; 0=Normal Mode*
REG_51<13>	*0*	*1=Transmit negative CDCSS code* *0=Transmit positive CDCSS code*
REG_51<12>	*0*	*CTCSS/CDCSS mode selection.* *1=CTCSS, 0=CDCSS*
REG_51<11>	*0*	*CDCSS 24/23bit selection.* *1=24bit, 0=23bit*
REG_51<10>	*0*	*1050Hz Detection Mode.* *1=1050/4 Detect Enable, CTC1 should be set to 1050/4 Hz*
REG_51<9>	*0*	*Auto CDCSS Bw Mode.* *1=Disable; 0=Enable.*

Register	Default	Description
REG_51<8>	*0*	*Auto CTCSS Bw Mode.* *0=Enable; 1=Disable*
REG_51<6:0>	*0*	*CTCSS/CDCSS Tx Gain1 Tuning.* *0=min; 0x7F=max*
REG_2E<9:8>	*0x10*	*CTCSS/CDCSS Tx Gain2 Tuning (after Gain1).* *00=12dB; 01=6dB; 10=0dB; 11=-6dB*
REG_07<15:0>		*When <13>=0 for CTC1 or CDCSS 134.4Hz* *<12:0>=CTC1 frequency control word* *= freq(Hz)* 20.64888 for XTAL 13M/26M or* *=freq(Hz)*20.97152 for XTAL* *12.8M/19.2M/25.6M/38.4M* *When<13>=1 for CTC2(Tail 55Hz Rx detection)* *<12:0>=CTC2 (should below 100Hz)frequency* *control word* *= 25391/freq(Hz) for XTAL 13M/26M or* *= 25000/freq(Hz) for XTAL* *12.8M/19.2M/25.6M/38.4M*
REG_08<15:0>		*<15>=1 for CDCSS high 12bit* *<15>=0 for CDCSS low 12bit* *<11:0>=CDCSS high/low 12bit code*
REG_52<15>	*0*	*Enable 120/180/240 degree shift CTCSS or 134.4Hz* *Tail when CDCSS mode.* *0=Normal, 1=Enable*

Register	Default	Description
REG_52<14:13>	*0b00*	*CTCSS tail mode selection (only valid when REG_52<15>=1).* *00= for 134.4Hz CTCSS Tail when CDCSS mode.* *01= CTCSS0 120°phase shift,* *10= CTCSS0 180°phase shift* *11= CTCSS0 240°phase shift*
REG_52<12>	*0*	*CTCSS Detection Threshold Mode,* *1=~0.1%; 0=0.1 Hz*
REG_52<11:6>	*0x0A*	*CTCSS found detect threshold.*
REG_52<5:0>	*0x0F*	*CTCSS lost detect threshold.*
REG_0C<15:14>	*Read Only*	*<14>:CDCSS positive code received* *<15>:CDCSS negative code received*
REG_0C<13:12>	*Read Only*	*CTCSS Phase Shift Received.* *00=No phase shift* *01=CTCSS0 120°phase shift,* *10= CTCSS0 180°phase shift* *11= CTCSS0 240°phase shift*
REG_0C<10:11>	*Read Only*	*<11>:CTC2(55Hz) received* *<10>:CTC1 received*

5. SELCALL

1) 开启 SELCALL(5Tone)模式使用 RF_Enter5tone()，该函数仅对接收频率系数、接收门限、发射通路进行了设置，不会影响到正常音频收听。

2) 退出 SELCALL(5Tone)模式使用 RF_Exit5tone()

3) 发射 SELCALL(5Tone)使用 RF_5toneTransmit()，使用 MCU 计时根据发射码更换发射 SELCALL(5Tone)的频率（Tone1）。

4) 接收 SELCALL(5Tone)使用 RF_5toneReceive()，返回 1 失败，返回 0 成功

6. DTMF

1) 开启 DTMF 模式使用 RF_EnterDtmf()，该函数仅对 DTMF 接收频率系数、接收门限、发射通路进行了设置，不会影响到正常音频收听。

2) 退出 DTMF 模式使用 RF_ExitDtmf()

3) 发射 DTMF 使用 RF_DtmfTransmit()，使用 MCU 计时根据发射码更换发射 DTMF 的频率（Tone1+Tone2）。

4) 接收 DTMF 使用 RF_DtmfReceive()，返回 1 失败，返回 0 成功

Register	Default	Description
REG_70<15>	*0*	*Enable TONE1* *1=Enable; 0=Disable.*
REG_70<14:8>	*0*	*TONE1 tuning gain*
REG_70<7>	*0*	*Enable TONE2* *1=Enable; 0=Disable.*
REG_70<6:0>	*0*	*TONE2/FSK tuning gain*
REG_71<15:0>	*0x8517*	*TONE1/Scramble frequency control word.* *=freq(Hz)* 10.32444 for XTAL 13M/26M or* *=freq(Hz)* 10.48576 for XTAL* *12.8M/19.2M/25.6M/38.4M.*
REG_72<15:0>	*0x2854*	*TONE2/FSK frequency control word* *=freq(Hz)* 10.32444 for XTAL 13M/26M or* *=freq(Hz)* 10.48576 for XTAL* *12.8M/19.2M/25.6M/38.4M.*
REG_50<15>	*0*	*Enable AF Tx Mute (for DTMF Tx or other applications).* *1=Mute; 0=Normal*
REG_0B<11:8>	*Read Only*	*DTMF/5Tone Code Received.*

7. FSK

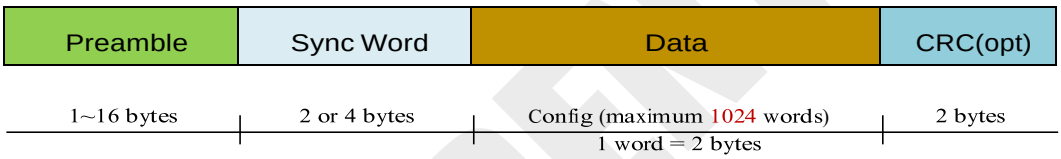
1) 开启 FSK 模式 使用 RF_EnterFsk()， 不会影 响到正 常音频收听， 且可以同 时进入 DTMF/SELCALL 模式进行接收。FSK 速率寄存器与 Tone2 寄存器复用。使用 2400bps 模式 需要开启宏定义 FSK2400

2) 退出使用 RF_ExitFsk()

3) 发射 FSK 使用 RF_FskTransmit()，返回 1 失败，返回 0 成功

4) FSK 帧格式（CRC 为可选），如果要兼容 BK4815/BK4818 则需要在 Data 部分完成 BK4815/BK4818FSK 帧结构里的 Addr/Type/Size/CRCA/Payload/CRCB 数据组帧，并且关掉 BK4819 FSK 帧结构的 CRC 部分，设置相同的 Preamble 和 SyncWord

5) 接收 FSK 使用 RF_FskReceive()，返回 1 失败，返回 0 成功



Register	Default	Description
REG_58<15:13>	*000*	*FSK Tx Mode Selection.* *000 for FSK1.2K and FSK2.4K Tx;* *001 for FFSK1200/1800 Tx;* *011 for FFSK1200/2400 Tx;* *101 for NOAA SAME Tx*
REG_58<12:10>	*000*	*FSK Rx Mode Selection.* *000 for FSK1.2K, FSK2.4K Rx and NOAA SAME Rx;* *111 for FFSK1200/1800 Rx;* *100 for FFSK1200/2400 Rx;*
REG_58<9:8>	*00*	*FSK Rx Gain.*

Register	Default	Description
REG_58<5:4>	*00*	*FSK Preamble Type Selection.* *11=0xAA; 10=0x55; 00=0xAA or 0x55 due to the MSB of FSK Sync Byte 0.*
REG_58<3:1>	*000*	*FSK Rx BandWidth Setting.* *100 for FSK 2.4K and FFSK1200/2400;* *000 for FSK 1.2K;* *001 for FFSK1200/1800;* *010 for NOAA SAME Rx*
REG_58<0>	*0*	*FSK Enable.* *1=Enable;* *0=Disable.*
REG_59<15>	*0*	*Clear TX FIFO, 1=clear*
REG_59<14>	*0*	*Clear RX FIFO, 1=clear*
REG_59<13>	*0*	*1=Enable FSK Scramble*
REG_59<12>	*0*	*1=Enable FSK RX*
REG_59<11>	*0*	*1=Enable FSK TX*
REG_59<10>	*0*	*1=Invert FSK data when RX*
REG_59<9>	*0*	*1=Invert FSK data when TX*
REG_59<7:4>	*0*	*FSK Preamble Length Selection* *0=1 byte; 1=2 bytes; 2=3 bytes; …; 15=16 bytes.*
REG_59<3>	*0*	*FSK SyncLength Selection.* *1=4 bytes (FSK Sync Byte 0,1,2,3) 0=2 bytes (FSK Sync Byte 0,1)*
REG_5A<15:8>	*0x85*	*FSK Sync Byte 0 (Sync Byte 0 first, then 1,2,3)*
REG_5A<7:0>	*0xCF*	*FSK Sync Byte 1*

Register	Default	Description
REG_5B<15:8>	*0xAB*	*FSK Sync Byte 2*
REG_5B<7:0>	*0x45*	*FSK Sync Byte 3*
REG_5C<6>	*1*	*CRC Option* *Enable. 1=Enable;* *0=Disable.*
REG_5D<15:8>	*0x0F*	*FSK Data Length(Byte)* *Low 8bits(Total 11 bits for BK4819). For example, 0xF means 16 bytes length.*
REG_5D<7:5>	*0*	*FSK Data Length(Byte)* *High 3bits(Total 11 bits for BK4819).*
REG_5E<9:3>	*64*	*FSK Tx FIFO (Total 128 Words) Almost Empty Threshold.*
REG_5E<2:0>	*4*	*FSK Rx FIFO (Total 8 Words) Almost Full Threshold.*
REG_5F<15:0>	*x*	*FSK Word Input/Output.*
REG_70<6:0>	*0*	*TONE2/FSK tuning gain*
REG_72<15:0>	*0x2854*	*TONE2/FSK frequency control word* *=freq(Hz)*10.32444 for XTAL 13M/26M or* *=freq(Hz)*10.48576 for XTAL 12.8M/19.2M/25.6M/38.4M.*
REG_0B<7>	*Read Only*	*FSK Rx SyncN Found.*
REG_0B<6>	*Read Only*	*FSK Rx SyncP Found.*
REG_0B<4>	*Read Only*	*FSK Rx CRC Indicator. 1=CRC Pass; 0=CRC Fail.*

8. MDC1200

1) 开启 MDC 模式使用 RF_EnterMdc()，不会影响到正常音频收听，且可以同时进入 DTMF/SELCALL 模式进行接收。MDC 速率寄存器与 Tone2 寄存器复用。默认模式为 1200/1800，若想使用 1200/2400 模式需要开启宏定义 MDC2400

2) 退出使用 RF_ExitMdc()

3) 发射 MDC 使用 RF_MdcTransmit()，返回 1 失败，返回 0 成功

4) 接收 MDC 使用 RF_MdcReceive()，返回 1 失败，返回 0 成功

5) 可支持 HDC1200 模式

9. NOAA SAME

1) 接收 NOAA SAME 码使用 RF_EnterNoaa ()，不会影响到正常音频收听，且可以同时进入 NOAA 模式进行接收。NOAA 速率寄存器与 Tone2 寄存器复用

2) 退出使用 RF_ExitNoaa()

3) 接收 FSK 使用 RF_NoaaReceive()，返回 1 失败，返回 0 成功，MCU 根据收到的码进行协议处理

10. VoX

1) 开启 VoX 使用 RF_EnterVox()；关闭使用 RF_ExitVox()

2) 获取 VoX 状态使用 RF_GetVox()，返回 1 收到 MIC 语音，返回 0 未收到语音。

3) 获取 VoX 幅度使用 RF_GetVoxAmp()，返回值为 MIC 语音幅度，供 MCU 自行 VoX 判断使用。

Register	Default	Description
REG_31<2>	*0*	*Enable VOX detection.* *1=Enable; 0=Disable*
REG_7A<15:12>	*8*	*VoX=0 Detection delay, *128ms*
REG_46<10:0>	*0x50*	*Voice Amplitude Threshold for VOX=1 detect*
REG_79<15:11>	*8*	*VoX Detection Interval Time.*
REG_79<10:0>	*0x40*	*Voice Amplitude Threshold for VOX=0 detect*
REG_64<15:0>	*Read Only*	*Voice Amplitude Out.*
REG_0C<2>	*Read Only*	*VoX Indicator 0: No 1: Yes*

11. Power Saving

1) 进入睡眠使用 RF_Sleep()，唤醒可以使用 RF_WakeUp()，也可以直接进入发射 RF_Txon()或接收 RF_Rxon()
2) 睡眠状态下电流约 200uA~300uA；唤醒后 IDLE 电流约 3mA

Register	Default	Description
REG_37<14:12>	*0b001*	*DSP Voltage Setting.*
REG_37<11>	*1*	*ANA LDO Selection. 1=2.7v, 0=2.4v*
REG_37<10>	*1*	*VCO LDO Selection. 1=2.7v, 0=2.4v*
REG_37<9>	*1*	*RF LDO Selection. 1=2.7v, 0=2.4v*
REG_37<8>	*1*	*PLL LDO Selection. 1=2.7v, 0=2.4v*
REG_37<7>	*0*	*ANA LDO Bypass. 1=Bypass, 0=Enable.*
REG_37<6>	*0*	*VCO LDO Bypass. 1=Bypass, 0=Enable.*
REG_37<5>	*0*	*RF LDO Bypass. 1=Bypass, 0=Enable.*
REG_37<4>	*0*	*PLL LDO Bypass. 1=Bypass, 0=Enable.*
REG_37<3>	*0*	*Reserved.*
REG_37<2>	*0*	*DSP Enable.* *1=Enable,* *0=Disable.*
REG_37<1>	*0*	*XTAL Enable.* *1=Enable,* *0=Disable.*
REG_37<0>	*0*	*Band-Gap Enable.1=Enable, 0=Disable.*

12. Tx/Rx Mode Switch

1) 发射使用 RF_Txon()
2) 接收使用 RF_Rxon()
3) 发射带侧音（如发送铃音）时，使用 RF_Txon_Beep()
4) 做基带接收（中频芯片解调后送至 MICP 或 MICN 脚，经滤波及解码后由 EARO 脚送出至音频功放）时，使用 RF_Rxon_Disc()，可选择是否同时由 PAOUT 送出 VCO 信号供外置混频器使用

Register	Default	Description
REG_30<15>	*0*	*VCO Calibration Enable.* *1=Enable, 0=Disable*
REG_30<13:10>	*0*	*Rx Link Enable (include LNA/MIXER/PGA/ADC).* *1111=Enable, 0000=Disable*
REG_30<9>	*0*	*AF DAC Enable.* *1=Enable, 0=Disable.*
REG_30<8>	*0*	*DISC Mode Disable.* *1=Disable, 0=Enable.*
REG_30<7:4>	*0*	*PLL/VCO Enable.* *1111=Enable, 0000=Disable*
REG_30<3>	*0*	*PA Gain Enable.* *1=Enable, 0=Disable*
REG_30<2>	*0*	*MIC ADC Enable.* *1=Enable, 0=Disable*
REG_30<1>	*0*	*Tx DSP Enable.* *1=Enable, 0=Disable*
REG_30<0>	*0*	*Rx DSP Enable.* *1=Enable, 0=Disable*

13. Squelch, RSSI, Ex-Noise, Gltich

可通过 RF_GetRssi()，RF_GetNoise()，RF_GetGlitch()获取相应的参数，便于设置静噪等级

Register	Default	Description
REG_78<15:8>	*0x48*	*RSSI threshold for Squelch=1, 0.5dB/step*
REG_78<7:0>	*0x46*	*RSSI threshold for Squelch =0, 0.5dB/step*
REG_4F<14:8>	*0x2F*	*Ex-noise threshold for Squelch =0*
REG_4F<6:0>	*0x2E*	*Ex-noise threshold for Squelch =1*
REG_4D<7:0>	*0x20*	*Glitch threshold for Squelch =0*
REG_4E<7:0>	*0x08*	*Glitch threshold for Squelch =1*
REG_4E<13:11>	*0b101*	*Squelch=1 Delay Setting.*
REG_4E<10:9>	*0b111*	*Squelch=0 Delay Setting.*
REG_67<8:0>	*Read Only*	*0.5dB/step, RSSI (dBm) ~= REG_67<8:0>/2 – 160.*
REG_65<6:0>	*Read Only*	*Ex-noise indicator, dB/step.*
REG_63<7:0>	*Read Only*	*Glitch indicator.*
REG_0C<1>	*Read Only*	*Squelch result output.* *1=Link; 0=Loss*

14. AFC, ALC, MIC AGC

MIC PGA 增益自动控制可扩展 MIC 信号动态范围，使较大幅度 MIC 信号不失真发射。用在 DMR 方案中发射时应该关掉此功能（***REG_19<15>=1***），同时 ALC 功能也应关闭（***REG_4B<5>=1***）。

Register	Default	Description
REG_73<13:11>	*0b000*	*Automatic Frequency Correction(AFC) Range Selection.* *000=max; 111=min*
REG_73<4>	*0*	*Automatic Frequency Correction(AFC) Disable.* *1=Disable; 0=Enable.*
REG_19<15>	*1*	*Automatic MIC PGA Gain Controller(MIC AGC) Disable.* *1=Disable; 0=Enable.*
REG_4B<5>	*0*	*AF Level Controller(ALC) Disable.* *1=Disable; 0=Enable.*

15. Frequency Setting

设置频率使用 RF_SetFreq(u16 freq_hi16, u16 freq_lo16)，注意换算公式：
Frequency(Hz)= (freq_hi16<<16 + freq_lo16)*10

Register	Default	Description
REG_38<15:0>	*0x3A98*	*Frequency(Hz)= (freq_hi16<<16 + freq_lo16)*10*
REG_39<15:0>	*0x0271*	

如设置 409.75MHz 频点，则 RF_SetFreq((40975000>>16)&0xFFFF, 40975000&0xFFFF)

16. Tx Output Power

Register	Default	Description
REG_36<15:8>	*0*	*PA Bias output 0~3.2V* *0x00=0V* *…* *0xFF=3.2V*
REG_36<7>	*0*	*1=Enable PACTL output; 0=Disable(Output 0 V)*
REG_36<5:3>	*0b111*	*PA Gain1 Tuning.* *111(max)->000(min)*
REG_36<2:0>	*0b111*	*PA Gain2 Tuning.* *111(max)->000(min)*

功率输出表（近似）

Power(dBm)	***PA Gain1***							
PA Gain2	*111*	*110*	*101*	*100*	*011*	*010*	*001*	*000*
111	*7.26*	*7.01*	*6.67*	*6.17*	*5.39*	*4.10*	*1.52*	*-5.02*
110	*6.38*	*6.08*	*5.67*	*6.06*	*4.12*	*2.53*	*-0.72*	*-9.45*
101	*5.65*	*5.30*	*4.82*	*4.13*	*3.03*	*1.18*	*-2.58*	*-13.4*
100	*5.01*	*4.62*	*4.08*	*3.30*	*2.08*	*0.01*	*-4.16*	*-16.9*
011	*4.19*	*3.73*	*3.11*	*2.21*	*0.84*	*-1.48*	*-6.11*	*-20.5*
010	*2.60*	*2.04*	*1.24*	*0.13*	*-1.56*	*-4.25*	*-9.50*	*-22.9*
001	*1.04*	*0.35*	*-0.63*	*-1.98*	*-3.90*	*-6.80*	*-12.3*	*-23.7*
000	*-0.70*	*-1.51*	*-2.65*	*-4.16*	*-6.20*	*-9.35*	*-14.9*	*-24.3*

17. Interrupt

1) 中断信号可由任意 GPIO 送出（见 GPIO 设置函数），也可轮询 *REG_0C<0>*位，高有效，默认低。

2) 中断可由芯片任意 GPIO 口输出，中断通过对 *REG_02* 寄存器写任意值来清除，如
RF_Write (0x02,0x0000); //clear interrupt

3) 中断高电平（或中断寄存器 *REG_0C<0>*为 1）有效，得到中断时，要先清除中断，才能去读取中断向量表。

Register	Default	Description
REG_0C<0>	*Read Only*	*Interrupt Indicator.* *1=Interrupt Request; 0=No Request.*
REG_3F<15>	*0*	*FSK Tx Finished Interrupt Enable.* *1=Enable; 0=Disable.*
REG_3F<14>	*0*	*FSK FIFO Almost Empty Interrupt Enable.* *1=Enable; 0=Disable.*
REG_3F<13>	*0*	*FSK Rx Finished Interrupt Enable.* *1=Enable; 0=Disable.*
REG_3F<12>	*0*	*FSK FIFO Almost Full Interrupt Enable.* *1=Enable; 0=Disable.*
REG_3F<11>	*0*	*DTMF/5TONE Found Interrupt Enable.* *1=Enable; 0=Disable.*
REG_3F<10>	*0*	*CTCSS/CDCSS Tail Found Interrupt Enable.* *1=Enable; 0=Disable.*
REG_3F<9>	*0*	*CDCSS Found Interrupt Enable.* *1=Enable; 0=Disable.*
REG_3F<8>	*0*	*CDCSS Lost Interrupt Enable.* *1=Enable; 0=Disable.*
REG_3F<7>	*0*	*CTCSS Found Interrupt Enable.* *1=Enable; 0=Disable.*

Register	Default	Description
REG_3F<6>	0	CTCSS Lost Interrupt Enable. 1=Enable; 0=Disable.
REG_3F<5>	0	VoX Found Interrupt Enable. 1=Enable; 0=Disable.
REG_3F<4>	0	VoX Lost Interrupt Enable. 1=Enable; 0=Disable.
REG_3F<3>	0	Squelch Found Interrupt Enable. 1=Enable; 0=Disable.
REG_3F<2>	0	Squelch Lost Interrupt Enable. 1=Enable; 0=Disable.
REG_3F<1>	0	FSK Rx Sync Interrupt Enable. 1=Enable; 0=Disable.
REG_02<15>	Read Only	FSK Tx Finished Interrupt.
REG_02<14>	Read Only	FSK FIFO Almost Empty Interrupt Enable.
REG_02<13>	Read Only	FSK Rx Finished Interrupt Enable.
REG_02<12>	Read Only	FSK FIFO Almost Full Interrupt.
REG_02<11>	Read Only	DTMF/5TONE Found Interrupt.
REG_02<10>	Read Only	CTCSS/CDCSS Tail Found Interrupt.
REG_02<9>	Read Only	CDCSS Found Interrupt.
REG_02<8>	Read Only	CDCSS Lost Interrupt.
REG_02<7>	Read Only	CTCSS Found Interrupt.
REG_02<6>	Read Only	CTCSS Lost Interrupt.
REG_02<5>	Read Only	VoX Found Interrupt.
REG_02<4>	Read Only	VoX Lost Interrupt.
REG_02<3>	Read Only	Squelch Found Interrupt.
REG_02<2>	Read Only	Squelch Lost Interrupt.
REG_02<1>	Read Only	FSK Rx Sync Interrupt.

18. GPIO

1) 根据对应的 PIN 及输出的模式使用 RF_SetGpioOut(u8 num, u8 type, u8 val)，其中 num 为内部编号，type 为输出模式，val 为 GPIO 输出模式下输出值。输出模式详见参考代码 driver.c

2) 获取 GPIO 输入值使用 RF_GetGpioIn(u8 num)，其中 num 为内部编号。

Register	Default	Description
REG_0A<6>	*Read Only*	*GPIO6 Input Indicator. 1=High; 0=Low.*
REG_0A<5>	*Read Only*	*GPIO5 Input Indicator. 1=High; 0=Low.*
REG_0A<4>	*Read Only*	*GPIO4 Input Indicator. 1=High; 0=Low.*
REG_0A<3>	*Read Only*	*GPIO3 Input Indicator. 1=High; 0=Low.*
REG_0A<2>	*Read Only*	*GPIO2 Input Indicator. 1=High; 0=Low.*
REG_0A<1>	*Read Only*	*GPIO1 Input Indicator. 1=High; 0=Low.*
REG_0A<0>	*Read Only*	*GPIO0 Input Indicator. 1=High; 0=Low.*
REG_33<14>	*1*	*GPIO6 Output Disable. 1=Disable; 0=Enable.*
REG_33<13>	*1*	*GPIO5 Output Disable. 1=Disable; 0=Enable.*
REG_33<12>	*1*	*GPIO4 Output Disable. 1=Disable; 0=Enable.*
REG_33<11>	*1*	*GPIO3 Output Disable. 1=Disable; 0=Enable.*
REG_33<10>	*1*	*GPIO2 Output Disable. 1=Disable; 0=Enable.*
REG_33<9>	*1*	*GPIO1 Output Disable. 1=Disable; 0=Enable.*
REG_33<8>	*1*	*GPIO0 Output Disable. 1=Disable; 0=Enable.*
REG_33<6>	*0*	*GPIO6 Output Value(if enabled). 1= High; 0=Low*
REG_33<5>	*0*	*GPIO5 Output Value(if enabled). 1= High; 0=Low*
REG_33<4>	*0*	*GPIO4 Output Value(if enabled). 1= High; 0=Low*
REG_33<3>	*0*	*GPIO3 Output Value(if enabled). 1= High; 0=Low*
REG_33<2>	*0*	*GPIO2 Output Value(if enabled). 1= High; 0=Low*
REG_33<1>	*0*	*GPIO1 Output Value(if enabled). 1= High; 0=Low*
REG_33<0>	*0*	*GPIO0 Output Value(if enabled). 1= High; 0=Low*

Register	Default	Description
REG_34<15:12>	*0x0*	*GPIO3 Output Type Selection.* *The Definitions is the same as REG_34<3:0>.*
REG_34<11:8>	*0x0*	*GPIO2 Output Type Selection.* *The Definitions is the same as REG_34<3:0>.*
REG_34<7:4>	*0x0*	*GPIO1 Output Type Selection.* *The Definitions is the same as REG_34<3:0>.*
REG_34<3:0>	*0x0*	*GPIO0 Output Type Selection. 0=High/Low* *1=Interrupt* *2=Squelch 3=VoX* *4=CTCSS/CDCSS Compared Result* *5=CTCSS Compared Result* *6=CDCSS Compared Result* *7=Tail Detected Result* *8=DTMF/5Tone Symbol Received Flag* *9=CTCSS/CDCSS Digital Wave* *Others=Reserved*
REG_35<11:8>	*0x0*	*GPIO6 Output Type Selection.* *The Definitions is the same as REG_34<3:0>.*
REG_35<7:4>	*0x0*	*GPIO5 Output Type Selection.* *The Definitions is the same as REG_34<3:0>.*
REG_35<3:0>	*0x0*	*GPIO4 Output Type Selection.* *The Definitions is the same as REG_34<3:0>.*

19. XTAL

芯片支持 26M，25.6M，13M，12.8M，19.2M 和 38.4M 的晶体或温补。默认 26M，若使用 26M 外的其他频率晶体或温补使用 RF_SetXtal(u8 mode)，如 RF_SetXtal(XTAL19M2)

20. Frequency Scan

1) 扫频使用 RF_FreqScan()可以获取 LNAIN 脚的射频频率（需要较大幅度>-40dBm），返回 1 表示失败，返回 0 表示成功。频率写入到全局变量 *FRQ_HI16* 和 *FRQ_LO16*

2) 扫到频率后，设置接收频率到该频点，使用 RF_CtcDcsScan()可获取 CTCSS 频率或 CDCSS 码，返回 0 表示失败，返回 1 表示收到 CTCSS 且频率写入全局变量 *CtC_FREQ*，返回 2 表示收到 CDCSS 且写入全局变量 *DCS_HI12* 和 *DCS_LO12*

Register	Default	Description
REG_32<15:14>	*0b00*	*Frequency Scan Time.* *00=0.2 Sec; 01=0.4 Sec; 10=0.8 Sec; 11=1.6 Sec*
REG_32<0>	*0*	*Frequency Scan Enable.* *1=Enable; 0=Disable.*
REG_0D<15>	*Read Only*	*Frequency Scan Indicator.* *1=Busy; 0=Finished.*
REG_0D<10:0>	*Read Only*	*Frequency Scan High 16 bits.*
REG_0E<15:0>	*Read Only*	*Frequency Scan Low 16 bits.* *= REG_0D<10:0><<16 + REG_0E<15:0>, unit is 10Hz*
REG_68<15>	*Read Only*	*CTCSS Scan Indicator.* *1=Busy; 0=Found.*
REG_68<12:0>	*Read Only*	*CTCSS Frequency.* *Frequency(Hz)* *= REG_68<12:0>/20.64888 for 13M/26M XTAL and* *= REG_68<12:0>/ 20.97152 for 12.8M/19.2M/25.6M/38.4M XTAL*
REG_69<15>	*Read Only*	*CDCSS Scan Indicator.* *1=Busy; 0=Found.*
REG_69<14>	*Read Only*	*23 or 24 bit CDCSS Indicator.(for BK4819) 1=24 bit; 0=23 bit.*

Register	Default	Description
REG_69<11:0>	*Read Only*	*CDCSS High 12 bits.*
REG_6A<11:0>	*Read Only*	*CDCSS Low 12 bits.*

21. Channel Spacing

芯片支持多种带宽，包括常见的 12.5k/25k/6.25k/20k，使用 RF_SetChnSpace(u8 space)，输入参数 SPACE_12K5/SPACE_25K/SPACE_6K25/SPACE_20K 即可。可根据实际需求设置发射接收带宽。

Register	Default	Description
REG_43<14:12>	*0b100*	*RF filter bandwidth (Apass=0.1dB)* *000 = 1.7 kHz* *001 = 2 kHz* *010 = 2.5 kHz* *011 = 3 kHz* *100 = 3.75 kHz* *101 = 4 kHz* *110 = 4.25 kHz* *111 = 4.5 kHz* *if REG_43<5>=1, RF filter bandwidth *=2;*
REG_43<11:9>	*0b000*	*RF filter bandwidth when signal is weak* *(Apass=0.1dB)* *000 = 1.7 kHz* *001 = 2 kHz* *010 = 2.5 kHz* *011 = 3 kHz* *100 = 3.75 kHz* *101 = 4 kHz* *110 = 4.25 kHz* *111 = 4.5 kHz* *if REG_43<5>=1, RF filter bandwidth *=2;*
REG_43<8:6>	*0b001*	*AF Tx LPF2 filter Band Width (Apass=1dB) Selection.* *100 = 4.5 kHz* *101 = 4.25 kHz*

Register	Default	Description
		110 = 4 kHz *111 = 3.75 kHz* *000 = 3 kHz (for 25k Channel Space)* *001 = 2.5 kHz (for 12.5k Channel Space)* *010 = 2.75 kHz* *011 = 3.5 kHz*
REG_43<5:4>	*0b00*	*BW Mode Selection.* *00=12.5k; 01=6.25k; 10=25k/20k*

22. Digital Walkie-Talkie

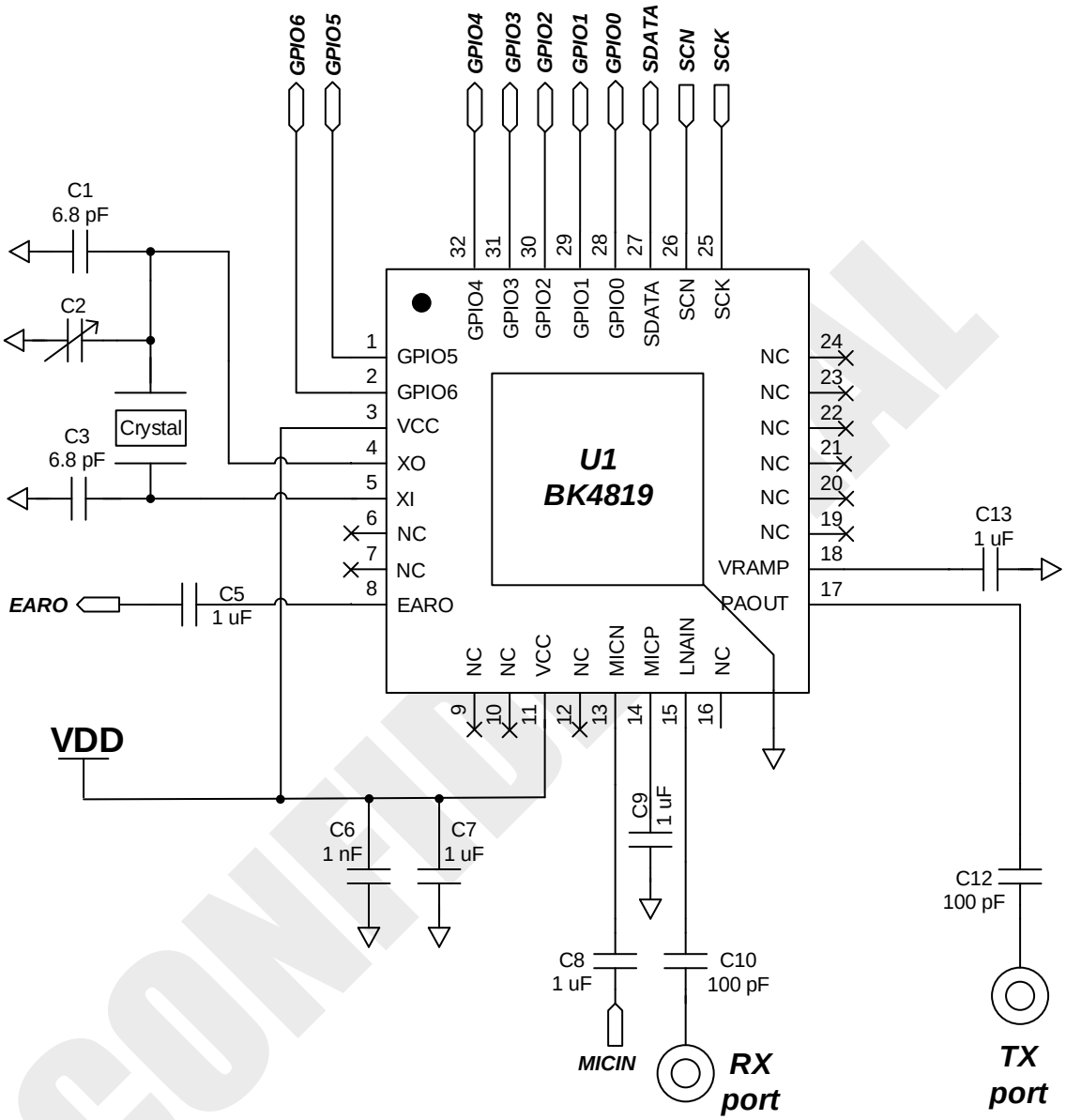
当做数字收发机使用时需要 bypass 所有音频滤波器，使用 RF_EnterBypass()，退出该模式使用 RF_ExitBypass()。以下函数在参考代码中可以找到。

```
void RF_EnterBypass() //for dMR
{
    RF_Write(0x47,REG_47 | 9<<8 | 3); //[11:8]=9 for Rx; [1]=1 for
gain; [0]=1 for Tx
    RF_Write(0x7E,REG_7E & 0xFFC7); //[5:3]=000b for Tx DCC bypass
}

void RF_ExitBypass()
{
    RF_SetAf(MUTE);
    RF_Write(0x7E,REG_7E);

}
```

23. Hardware Design



24. Revision History

Version	Date	Description
1.0	September 18, 2021	Initial release