

## RFM65-69-69H 模块 StarterKit tool 应用示例

1. 应用简介: .....	2
2. RF_DK_PIC16 开发板平台: .....	2
3. StarterKit Tool 安装: .....	5
4. StarterKit tool 与 RF_DK board 的联机: .....	5
5. StarterKit tool 软件简介 .....	6
5.1 菜单工具栏 .....	7
5.1.1 文件菜单 .....	7
5.1.2 操作菜单 .....	8
5.1.3 帮助菜单 .....	8
5.2 窗口工具栏 .....	9
5.3 状态栏 .....	10
5.4 IRQ 状态指示器 .....	10
5.5 操作模式控制栏 .....	11
5.6 寄存器配置标签 .....	11
5.6.1 Common 寄存器配置标签 .....	11
5.6.2 Transmitter 寄存器配置标签 .....	13
5.6.3 Receiver 寄存器配置标签 .....	13
5.6.4 Irq & Mapping 寄存器配置标签 .....	15
5.6.5 Packet Handler 寄存器配置标签 .....	17
6. Packet 模式下 StarterKit 工具的使用方法 .....	18
7. 更多模块的应用示例, 请登陆 <a href="http://www.hoperf.cn">www.hoperf.cn</a> .....	18

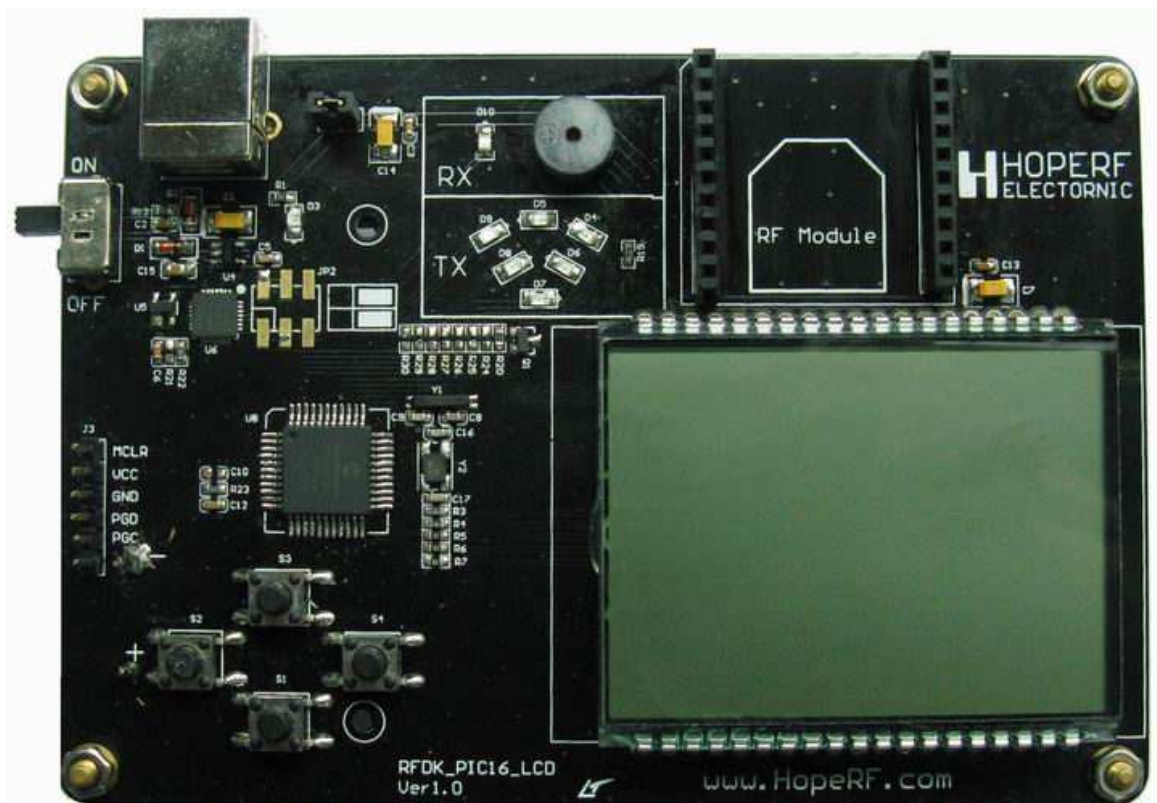
## 1. 应用简介:

射频模块的系统设计应用中，为驱动射频芯片正常工作，必须首先要对射频芯片进行参数配置，即对芯片的各个寄存器进行初始化的赋值，不同的工作方式就需要配置不同的射频参数，为了简化开发者在使用中对不同射频参数值的计算与寄存器参数值的设定，公司特推出了芯片寄存器配置工具 StarterKit tool，通过此寄存器配置工具，开发者可以方便快捷的得到不同射频参数下芯片所有寄存器的配置值。

StarterKit tool 工具是基于华普 RF\_DK\_PIC16 开发板的硬件平台联机使用的，基于此硬件平台，开发者可以直接使用 StarterKit tool 界面得到各种不同射频参数的寄存器配置值，并能够直接作用于 RF\_DK\_PIC16 开发板平台上的射频模块，来检验和验证不同参数下的射频模块的工作效果和参数性能。

开发者只需要购买本公司的 RF\_DK\_PIC16 开发板，并通过公司的网站，下载相应模块的寄存器 StarterKit tool 工具，并将其安装在上位机 PC 上，通过 USB 接口将 RF\_DK\_PIC16 开发板和上位机相连接，即可直接使用 StarterKit tool 配置射频模块工作。

## 2. RF\_DK\_PIC16 开发板平台:

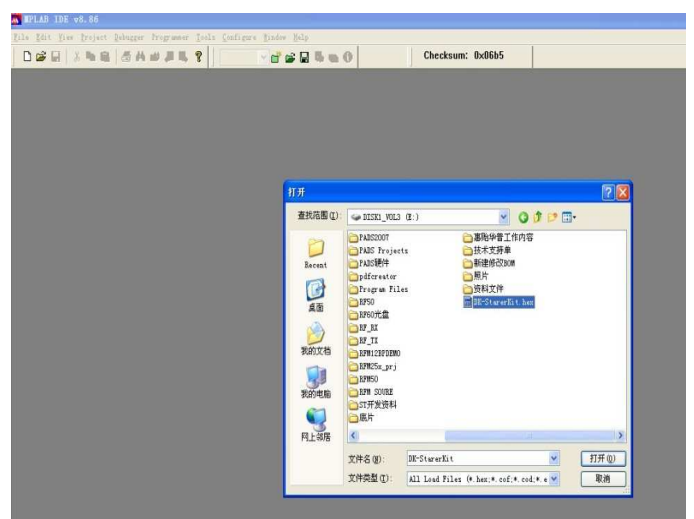
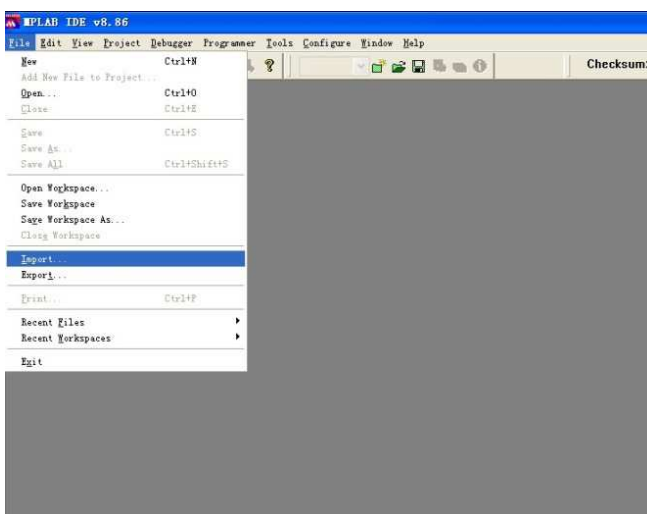
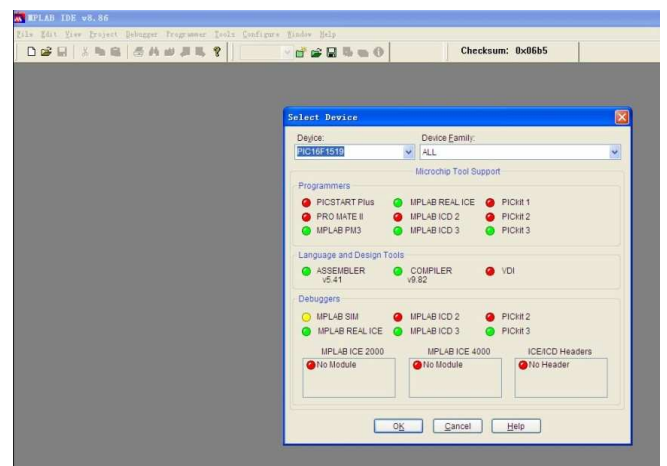
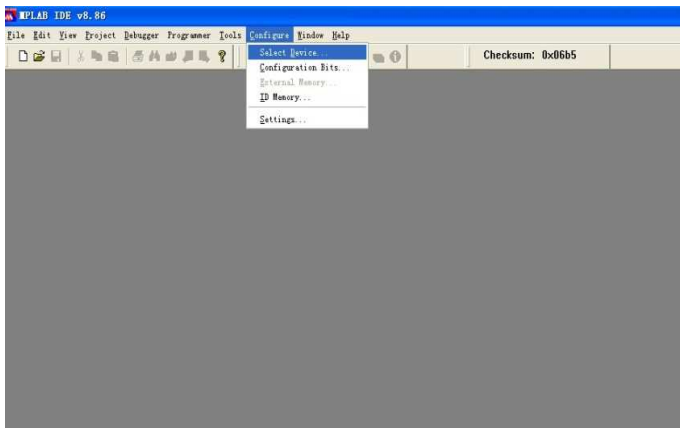


RFDK\_PIC16\_LCD\_Ver1.1 实物图

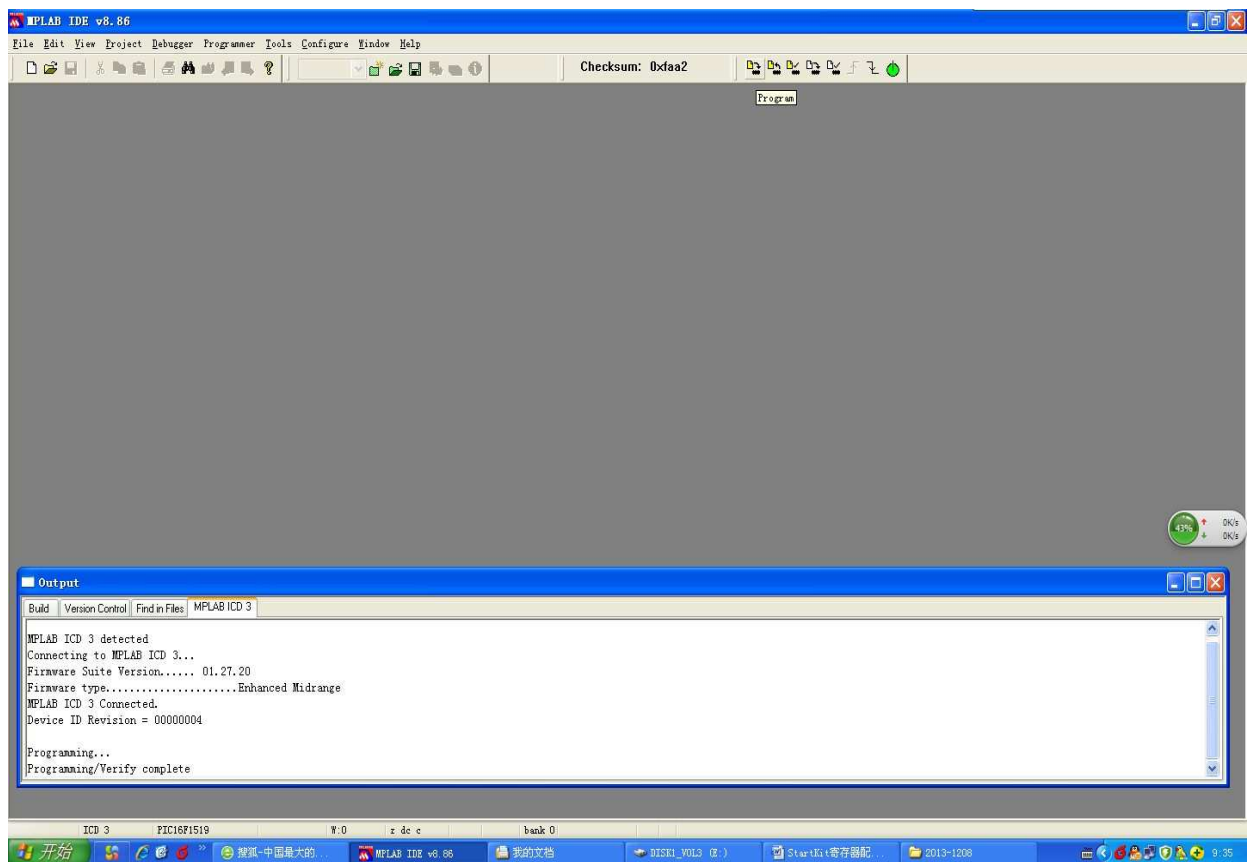
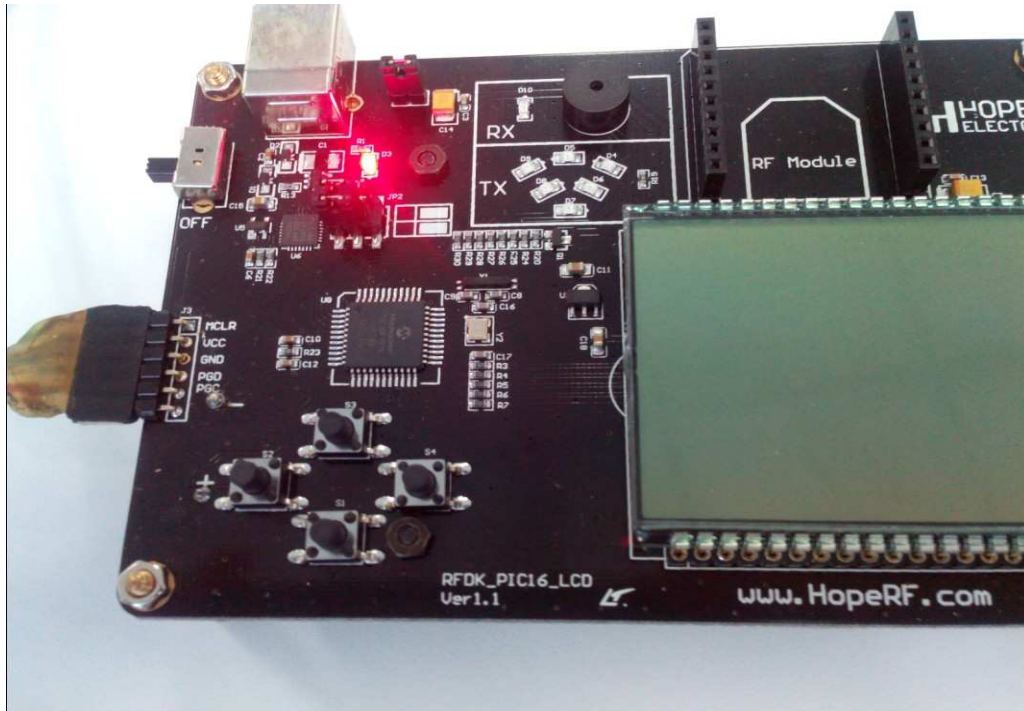
RF\_DK\_PIC16 开发板平台系统是基于 Microchip 公司的 PIC16 系列单片机开发的硬件平台，软件开发环境为 MPLAB IDE v8.86，此 DK 平台是我司针对所有型号射频模块进行快速开发的硬件平台。

由于 RF\_DK\_PIC16 开发板出厂时是作为 DEMO 功能演示板，所以开发板内带有 DEMO 软件系统，此时是无法直接将其联机与 StarterKit tool 工具配合使用的，需要开发者从我司网站上下载相应的固件文件，对开发板的 MCU 进行重新 download，才能使用开发板与 StarterKit tool 进行联机调试。具体步骤如下：


- 首先从我司网站上下载相应的固件文件 DK\_StarterKit.rar，（添加链接网址）并保存于根目录下，解压得到 DK\_StarterKit.Hex 文件。
- 其次打开 MPLAB IDE 开发环境，选择好相应的编程器并勾选上编程器供电的选项，对应芯片选择为 PIC16F1519，然后导入 DK\_StarterKit.Hex 文件，

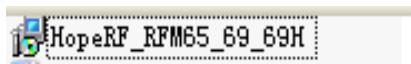


- 将编程器连接到 RF\_DK\_PIC16 开发板的下载接口上，点击编程，重新下载固件文件，编程器提示编程成功即可。

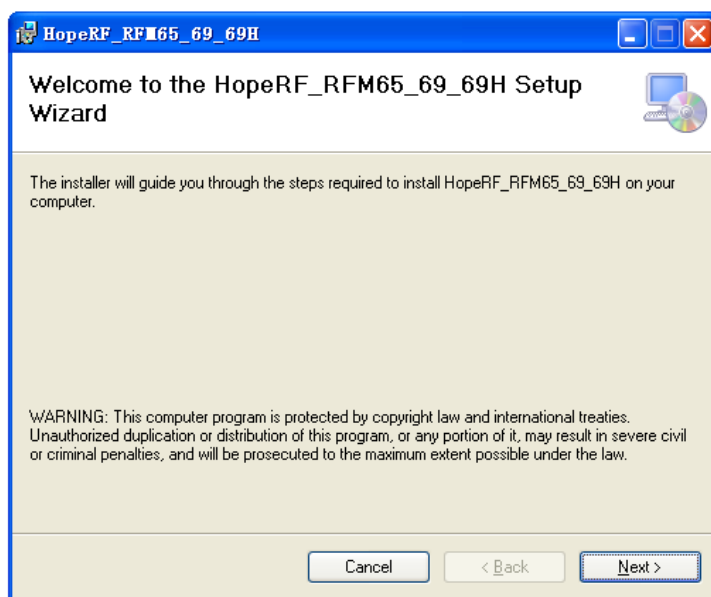


### 3. StarterKit Tool 安装:

StarterKit 寄存器配置工具的安装: 客户需要从公司网站上下载相应模块的寄存器配置工具, 然后进行安装, 以 RF69 芯片系列模块的寄存器配置工具的安装应用为例, 下载后解压得到文件包  HopeRF\_RFM65\_69\_69H\_StarterKit , 进入文件夹, 选中安装文件



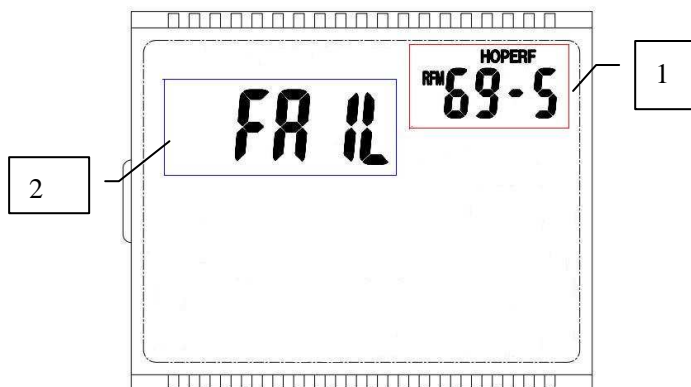
进行安装, 按照提示一步一步操作,



直至安装成功。

### 4. StarterKit tool 与 RF\_DK board 的联机:

- 通过 USB 接口将 RF\_DK board 连接到电脑上, 可以选用 USB 供电, 或则外部电池供电, 拨动电源控制开关, RF\_DK board 即开始工作, 开发板 LCD 屏显示指示如下:




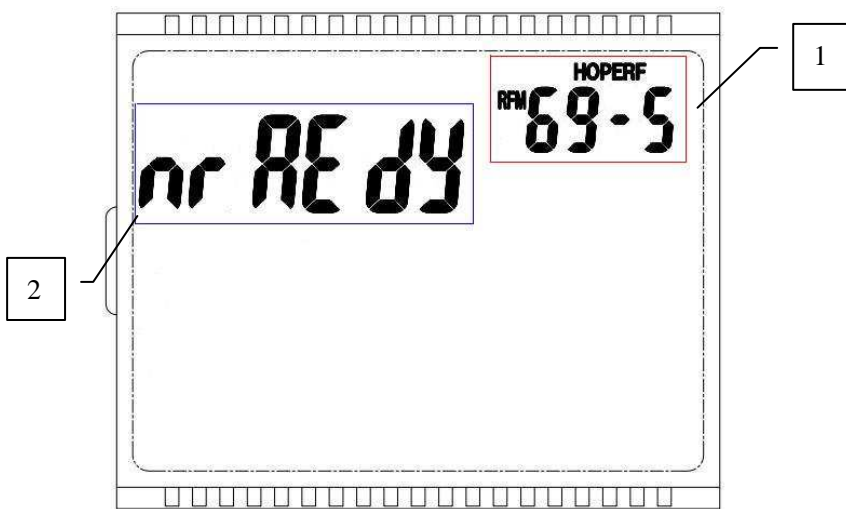


标注 1 部分为基本信息显示，显示有公司的图标和所连接的 StarterKit 型号的信息，如图显示，连接的为“RFM69-S”系列芯片的工具。

标注 2 部分为连接信息显示，此时因未打开 StarterKit 工具，开发板未成功与 StarterKit 联机，故显示“fail”的信息。



- 点击图标 ，打开 StarterKit 寄存器配置工具，StarterKit 会自动扫描串口，与 RF\_DK 开发板进行联机，也可通过显示界面上的“connect”图标进行联机和释放，显示界面如下：



标注 2 显示已联机的信息“nready”，提示 StarterKit 工具与 RF\_DK 开发板已正常联机。

## 5. StarterKit tool 软件简介

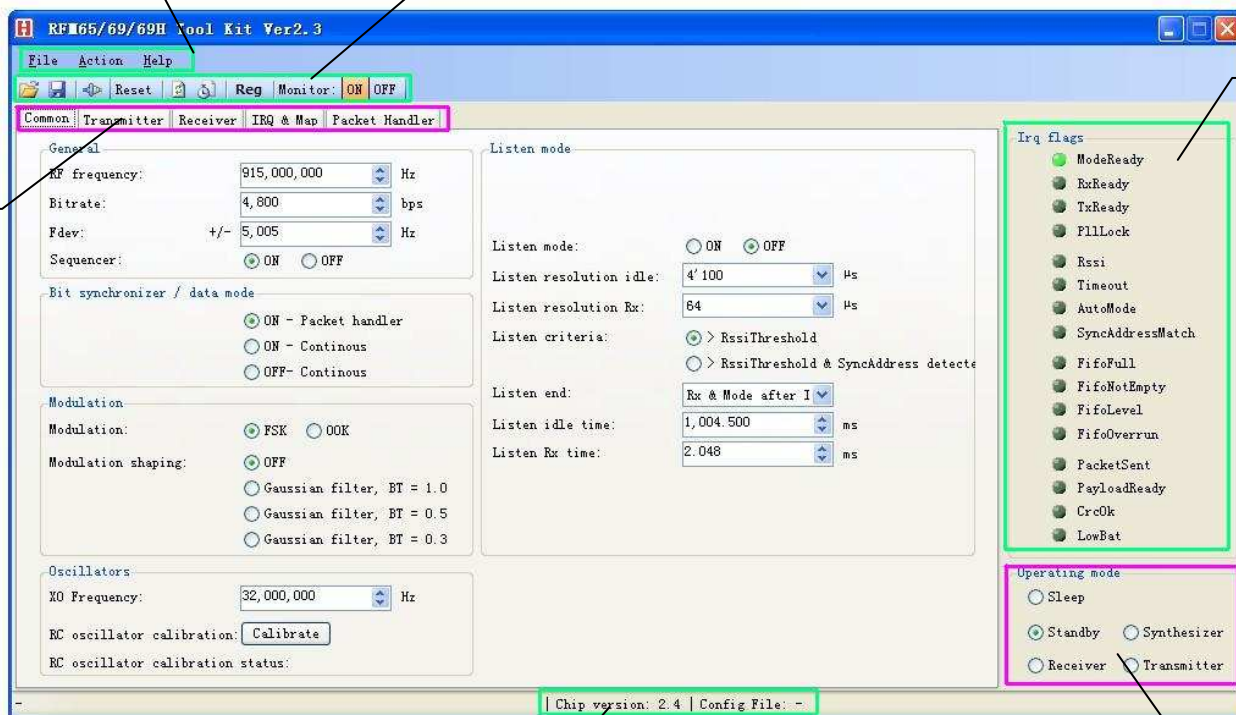
如下图为 StarterKit tool 的操作界面，针对界面中的各个部分的内容和功能，我们在下面的文档中会逐一的进行详解。

5.1

5.2

5.6

5.4



5.3

5.5

## 5.1 菜单工具栏

菜单工具包括了三个下拉菜单：文件菜单、操作菜单和帮助菜单。

### 5.1.1 文件菜单

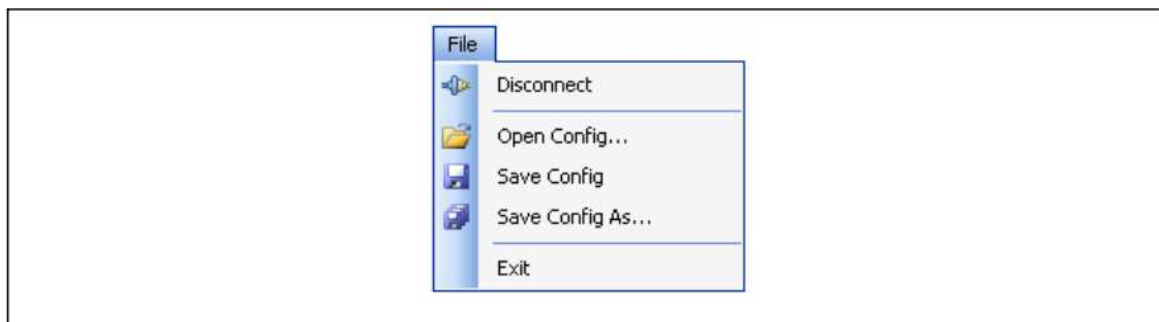
**Connect/Disconnect**：控制寄存器配置界面与 DK 开发板联机或断开联机，这个功能还可以通过窗口工具栏选项中的快捷键实现，如 5.2 节所述。

**Open Config ...**：打开寄存器配置工具所保存的配置文件（.cfg），寄存器工具会通过一个标准的窗口文件对话框读入到工具相对应的文件对话框中，然后将此配置写入到射频模块中，此功能也可通过窗口工具栏的快捷键来实现。

**Save Config**：将寄存器配置工具中当前的射频模块的配置文件（.cfg）进行保存，这是通过一个标准的窗口文件对话框来实现的，默认的文件名是保存的最后一个配置文件。

**Save Config as...**：将寄存器配置工具中的配置文件（.cfg）重新进行保存，提示输入新的文件名保存，可以保存多个配置文件，也可通过工具栏的快捷键来实现。

**Exit:** 关闭当前应用。



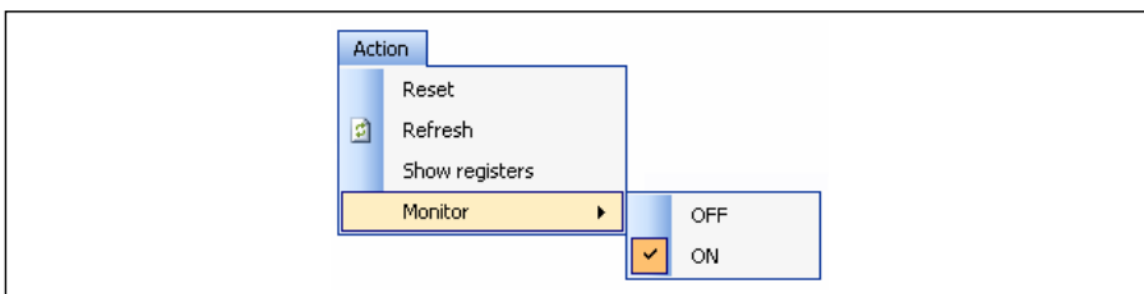
### 5.1.2 操作菜单

**Reset:** 复位，将射频模块的所有寄存器清空配置，恢复成为寄存器默认值。

**Refresh:** 重新读取射频模块的所有的寄存器的值到寄存器配置工具中。

**Show register:** 切换寄存器的显示窗口，并且可以通过工具栏的快捷键来切换，该寄存器窗口显示了当前射频模块的状态下的所有寄存器的配置值，更详细的配置值所代表的状态可以参照规格书中的介绍。

**Monitor On:** 开启寄存器配置工具读取状态标志位的功能，允许配置工具扫描读取射频模块当前的 IRQ 状态寄存器中的状态位，地址为 0x27 的 RegIrqFlag1 和地址为 0x28 的 RegIrqFlag2 的寄存器，并将相应的状态位的结果通过工具上的 IRQ 状态指示灯栏做相应的显示。Monitor Off 关闭此扫描读取状态位的功能。



### 5.1.3 帮助菜单

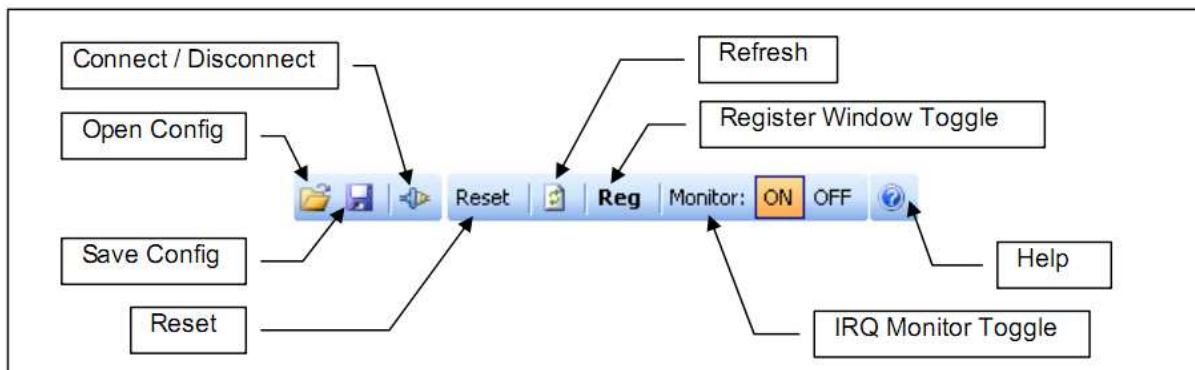
**About RFM.....:** 提供了寄存器配置工具的基本图形信息，可从华普公司的网站上下载 StarterKit tool 的最新版本。



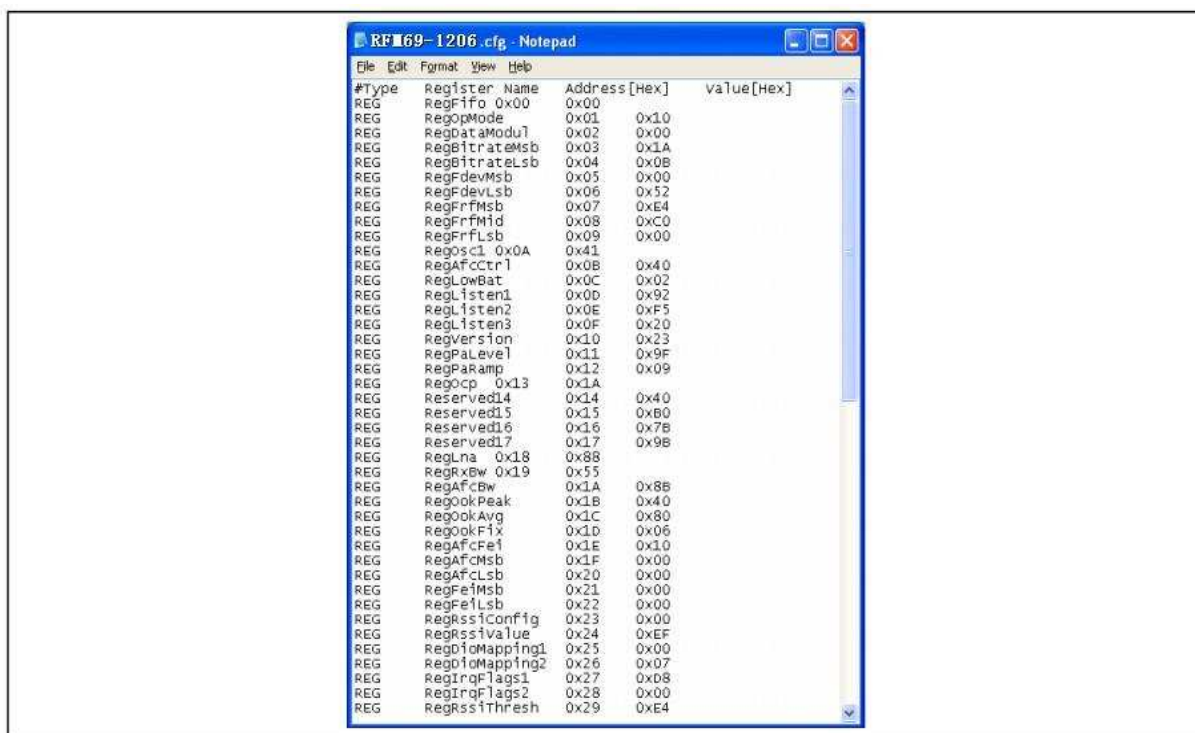


## 5.2 窗口工具栏

窗口工具栏提供了三个按键模块，创建了文件下拉菜单功能的快捷键方式。



**Open Config ...:** 打开寄存器配置工具所保存的配置文件 (.cfg)，寄存器工具会通过一个标准的窗口文件对话框读入到工具相对应的文件对话框中，这是一个带有格式的配置文件，可以直接被寄存器配置工具所调用，然后将此配置写入到射频模块中。这个配置文件可以用任何一个 txt 编辑器打开，里面包含了寄存器名称、寄存器地址和配置值等内容，配置文件为如下格式：



**Save Config:** 保存按钮将保存并提示是否覆盖现有的配置文件。

**Connect/Disconnect :** 控制寄存器配置界面与 DK 开发板联机或断联。

**Reset:** 复位，将射频模块的所有寄存器清空配置，恢复成为寄存器默认值。

**Refresh:** 重新读取射频模块的所有的寄存器的值到寄存器配置工具中。

**Show register:** 切换寄存器的显示窗口，该寄存器窗口显示了当前射频模块的状态下的所有寄存器的配置值。

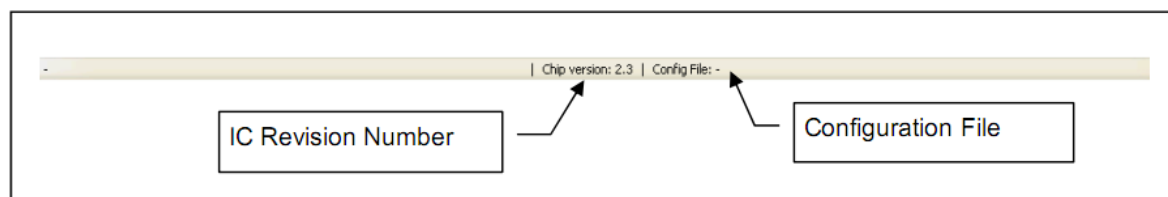
Register	Addr	Value	Register	Addr	Value	Register	Addr	Value	Register	Addr	Value	Register	Addr	Value
RegFifo	0x00	0x00	Reserved15	0x15	0xB0	RegRxTimeout1	0x2A	0x00	RegAesKey2	0x3F	0x00	RegTestAfc	0x71	0x00
RegOpMode	0x01	0x10	Reserved16	0x16	0x7B	RegRxTimeout2	0x2B	0x00	RegAesKey3	0x40	0x00			
RegDataModul	0x02	0x00	Reserved17	0x17	0x9B	RegPreambleMsb	0x2C	0x00	RegAesKey4	0x41	0x00			
RegBitrateMsb	0x03	0x1A	RegLna	0x18	0xB0	RegPreambleLsb	0x2D	0x03	RegAesKey5	0x42	0x00			
RegBitrateLsb	0x04	0x0B	RegRxBw	0x19	0x55	RegSyncConfig	0x2E	0x98	RegAesKey6	0x43	0x00			
RegFdevMsb	0x05	0x00	RegAfcBw	0x1A	0x8B	RegSyncValue1	0x2F	0x01	RegAesKey7	0x44	0x00			
RegFdevLsb	0x06	0x52	RegOokPeak	0x1B	0x40	RegSyncValue2	0x30	0x01	RegAesKey8	0x45	0x00			
RegFrfMsb	0x07	0xE4	RegOokAvg	0x1C	0x80	RegSyncValue3	0x31	0x01	RegAesKey9	0x46	0x00			
RegFrfMid	0x08	0xC0	RegOokFix	0x1D	0x06	RegSyncValue4	0x32	0x01	RegAesKey10	0x47	0x00			
RegFrfLsb	0x09	0x00	RegAfcFei	0x1E	0x10	RegSyncValue5	0x33	0x01	RegAesKey11	0x48	0x00			
RegOsc1	0x0A	0x41	RegAfcMsb	0x1F	0x00	RegSyncValue6	0x34	0x01	RegAesKey12	0x49	0x00			
RegAfcCtrl	0x0B	0x40	RegAfcLsb	0x20	0x00	RegSyncValue7	0x35	0x01	RegAesKey13	0x4A	0x00			
RegLowBat	0x0C	0x02	RegFeiMsb	0x21	0x00	RegSyncValue8	0x36	0x01	RegAesKey14	0x4B	0x00			
RegListen1	0x0D	0x52	RegFeiLsb	0x22	0x00	RegPacketConfig1	0x37	0x00	RegAesKey15	0x4C	0x00			
RegListen2	0x0E	0x55	RegRssiConfig	0x23	0x00	RegPayloadLength	0x38	0x42	RegAesKey16	0x4D	0x00			
RegListen3	0x0F	0x20	RegRssiValue	0x24	0x4F	RegNodeAdrs	0x39	0x00	RegTemp1	0x4E	0x01			
RegVersion	0x10	0x24	RegDioMapping1	0x25	0x60	RegBroadcastAdrs	0x3A	0x00	RegTemp2	0x4F	0x00			
RegPaLevel	0x11	0x3F	RegDioMapping2	0x26	0x07	RegAutoModes	0x3B	0x00	RegTestLna	0x58	0x1B			
RegPaRamp	0x12	0x09	RegIrqFlags1	0x27	0x08	RegFifoThresh	0x3C	0x8F	RegTestPa1	0x5A	0x55			
RegOcp	0x13	0x1A	RegIrqFlags2	0x28	0x00	RegPacketConfig2	0x3D	0x02	RegTestPa2	0x5C	0x70			
Reserved14	0x14	0x40	RegRssiThresh	0x29	0xE4	RegAesKey1	0x3E	0x00	RegTestDagc	0x6F	0x30			

**Monitor On:** 开启寄存器配置工具读取状态标志位的功能，允许配置工具扫描读取射频模块当前的 IRQ 状态寄存器中的状态位，地址为 0x27 的 RegIrqFlag1 和地址为 0x28 的 RegIrqFlag2 的寄存器，并将相应的状态位的结果通过工具上的 IRQ 状态指示灯栏做相应的显示。Monitor Off 关闭此扫描读取状态位的功能。

**Help:** 提供了寄存器配置工具的基本图形信息。

### 5.3 状态栏

状态栏显示了 RF69 芯片的版本信息和当前用户的配置文件，详细的芯片版本信息请参考规格书中关于版本信息的描述。

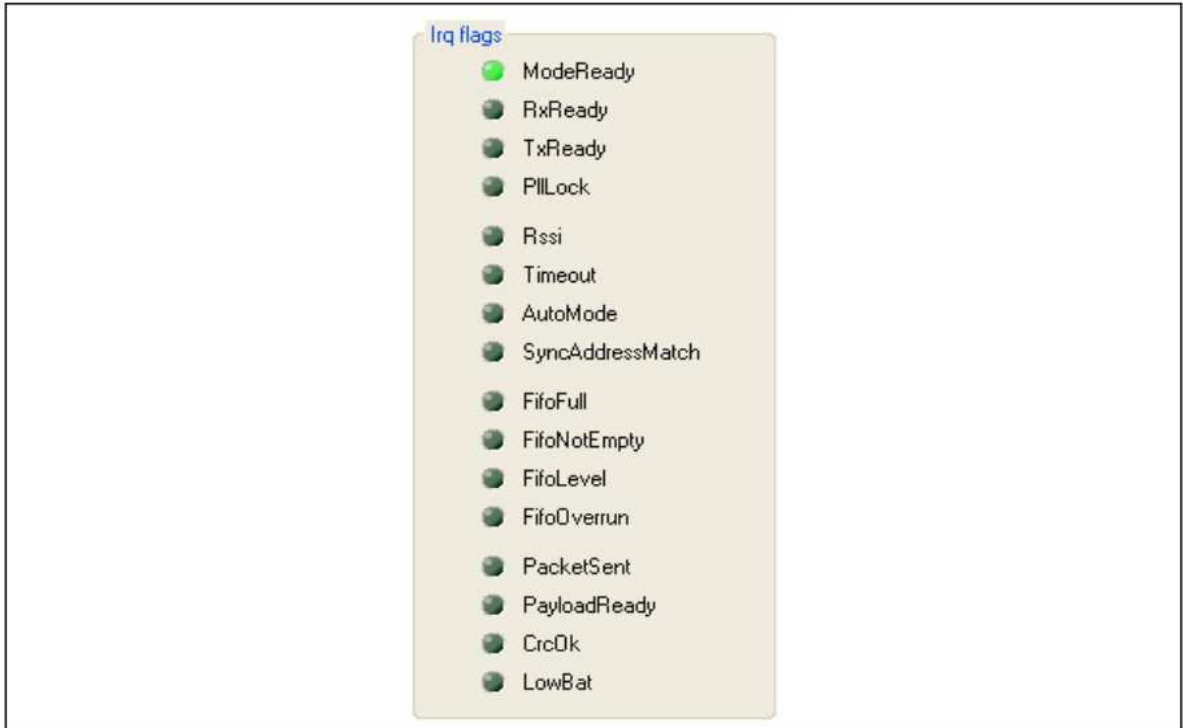


### 5.4 IRQ 状态指示器

IRQ 状态指示器提供 IRQ 寄存器的状态指示，当指示器中的 IRQ 描述亮起时，此时 IRQ

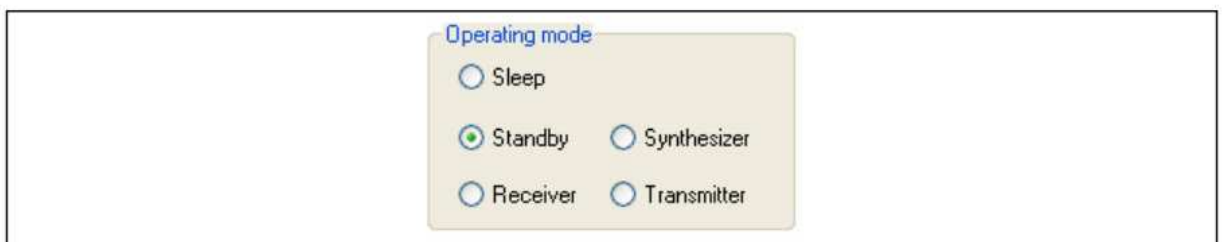
条件为真。

更详细的 IRQ 条件的内容请参照 RF69 规格书中对 IRQ 状态寄存器以及各种 IRQ 条件的描述。



## 5.5 操作模式控制栏

用户可以通过操作模式控制栏来改变模块的工作模式状态，只需要单击工作模式的应用选项，相应的工作模式指示亮起，寄存器工具即会配置模块工作与相应的工作模式之下。



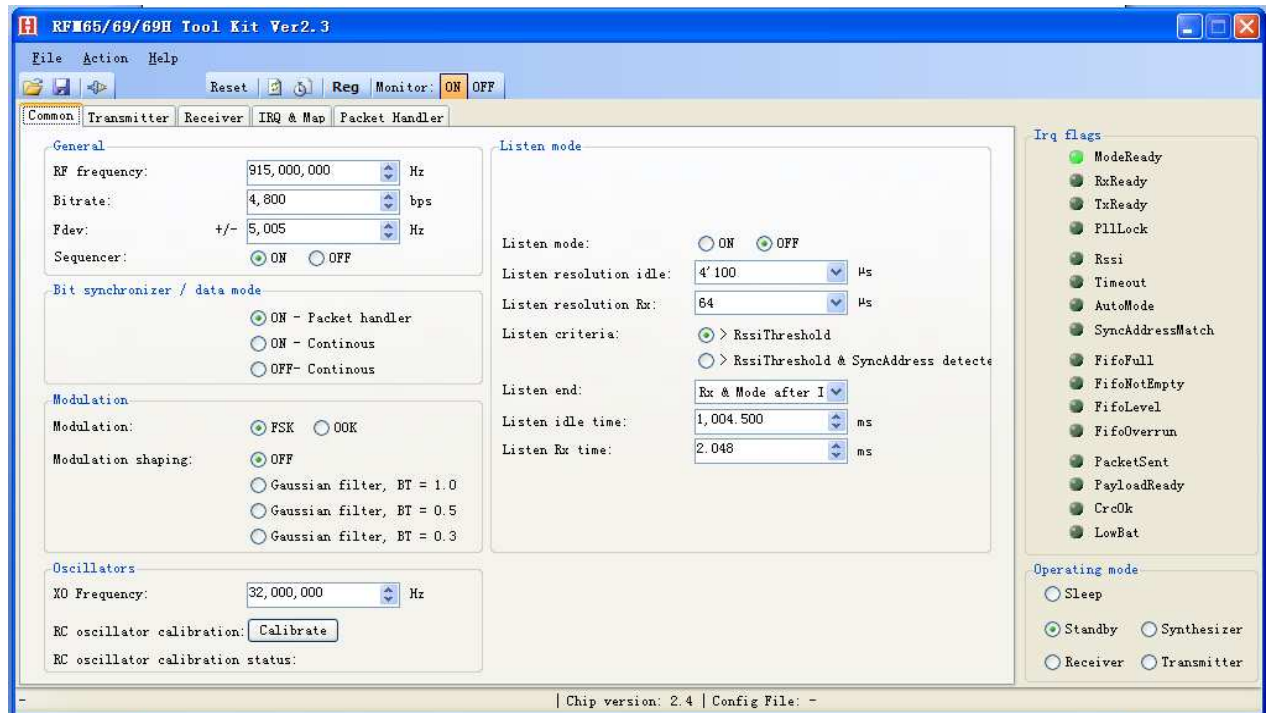
## 5.6 寄存器配置标签

当寄存器配置标签中的内容发生改变时，新设置的寄存器值会被立即更新到射频芯片的寄存器中，建议在射频芯片的待机模式下进行寄存器的重新设定。

### 5.6.1 Common 寄存器配置标签

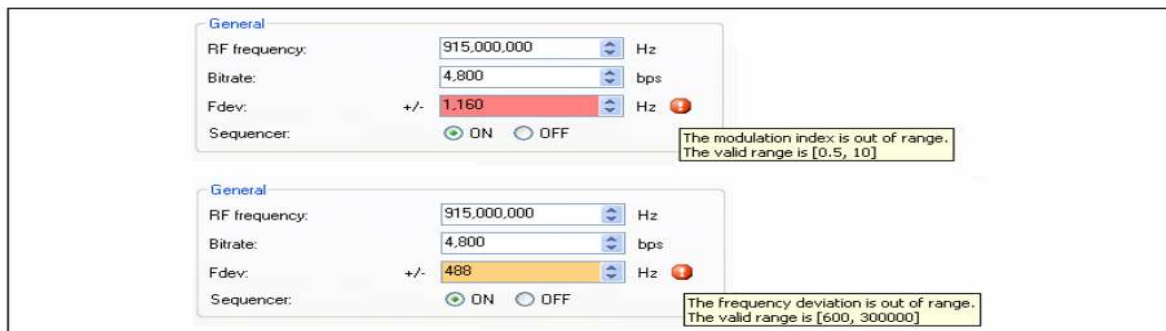
**Common** 通用寄存器配置标签可以设置射频芯片的基本配置，如下详细描述了此通用

寄存器配置的内容和设置参数。

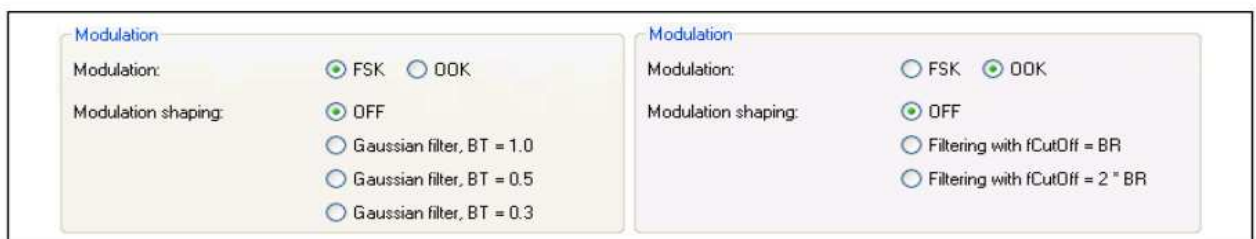


通过此标签中的设置，可以对射频芯片的中心频点、数据速率、发射频偏等参数进行设定，可以通过上下拉选项调节，也可以直接在窗口框内输入所要设置的内容，可以设置基本的调制方式以及数据包的格式，在每一项的设置发生改变时，都会立即将新设定的值更新入射频芯片中去。

由于 StarterKit 工具内部对速率与发射频偏有公式的计算，故有时在设置时可能会出现红色的感叹号提示，此时无需理会此提示即可。



通过调制模式标签可以选择射频模块的调制方式，FSK 调制或者 OOK 调制方式。

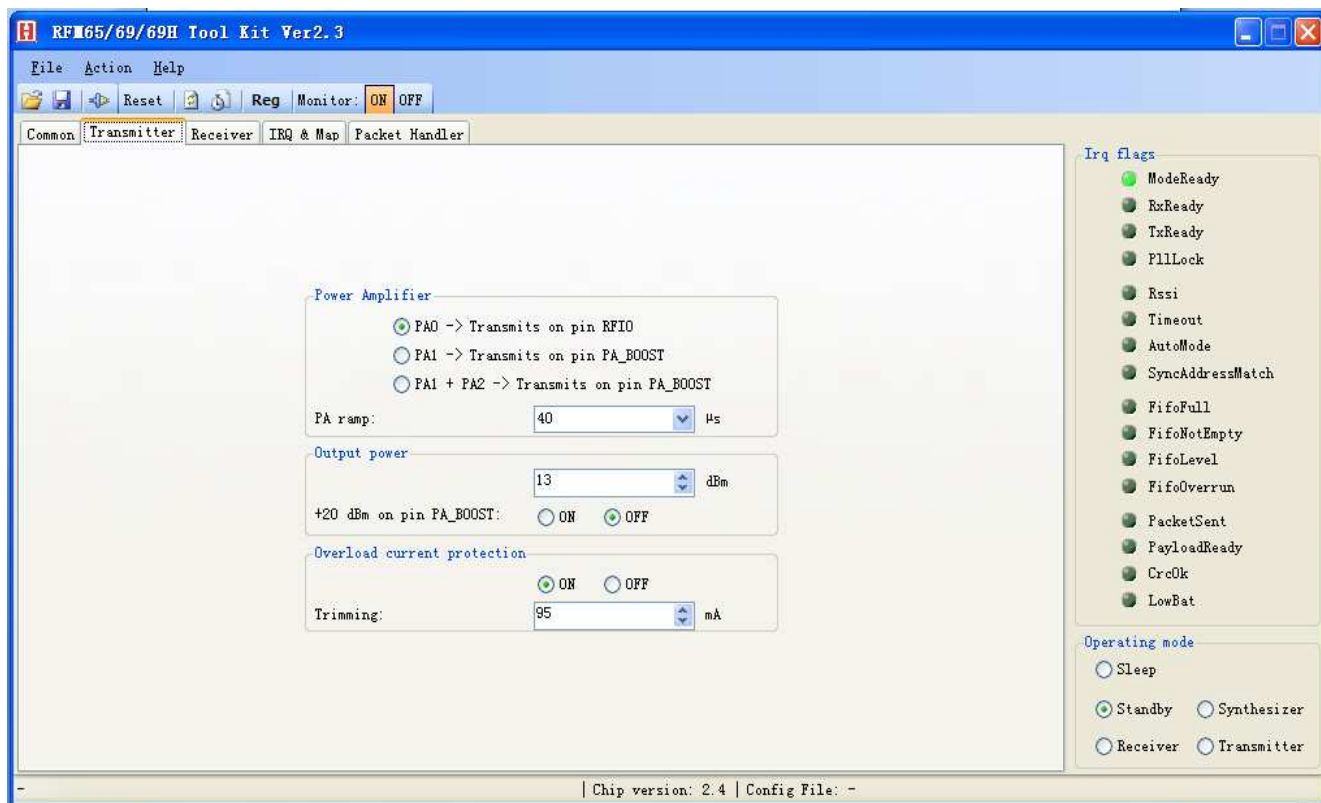




### 5.6.2 Transmitter 寄存器配置标签

**Transmitter** 寄存器配置标签如下图所示，通过此标签内的项目可以选择射频芯片的发射参数，如可以调节功率输出的设置，可以选择 PA 开启响应时间的设置，还可以选择芯片内部的 PA0 或 PA1 或 PA2 输出，通过输出电流保护标签可以设置允许的最大输出电流的设置。

寄存器的设置可以通过上下拉按钮去选择，也可以通过在相应窗口直接输入的方式来来进行不同参数的设置。

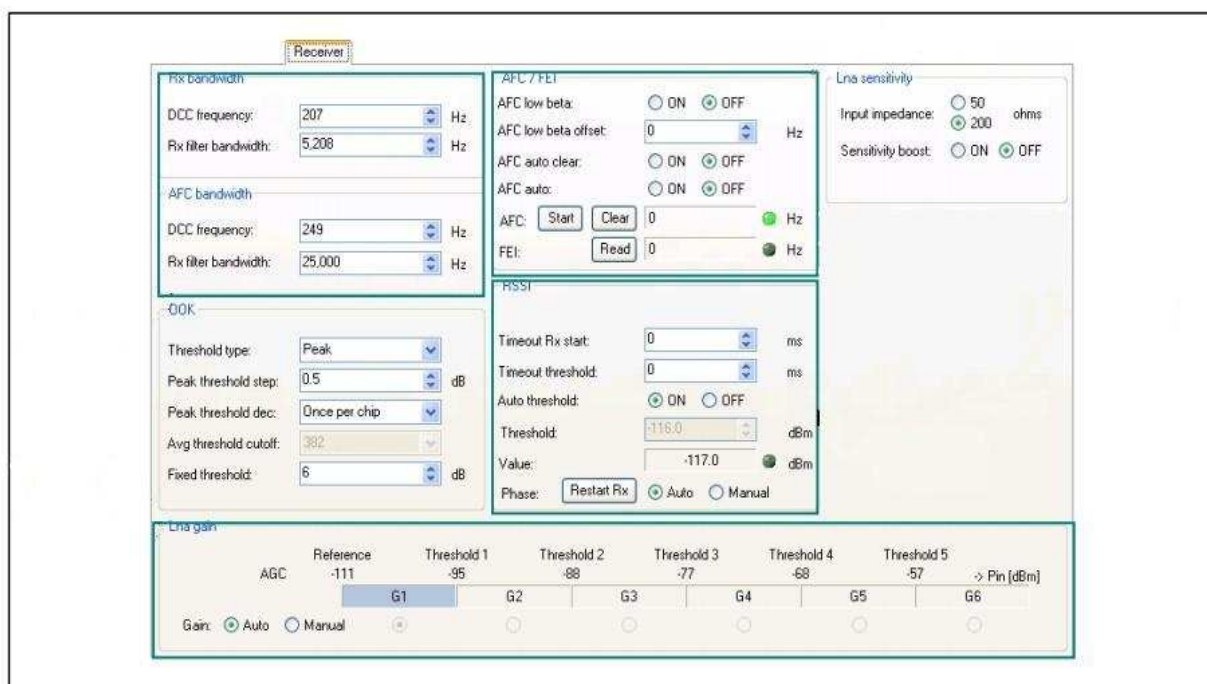


### 5.6.3 Receiver 寄存器配置标签

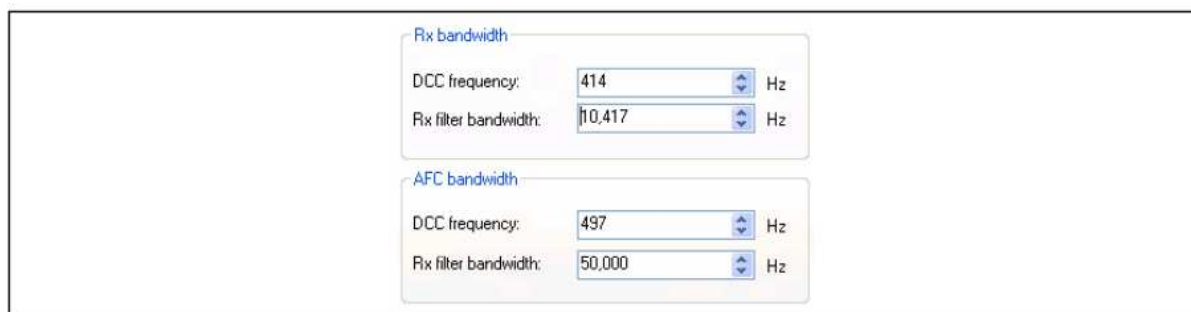
**Receiver** 寄存器配置标签的内容显示如下图所示，通过此配置标签可以设置模块的接收带宽，滤波器带宽等，以及其它与接收相关的参数设置，详见如下介绍。

寄存器的设置可以通过上下拉按钮去选择，也可以通过在相应窗口直接输入的方式来来进行不同参数的设置。





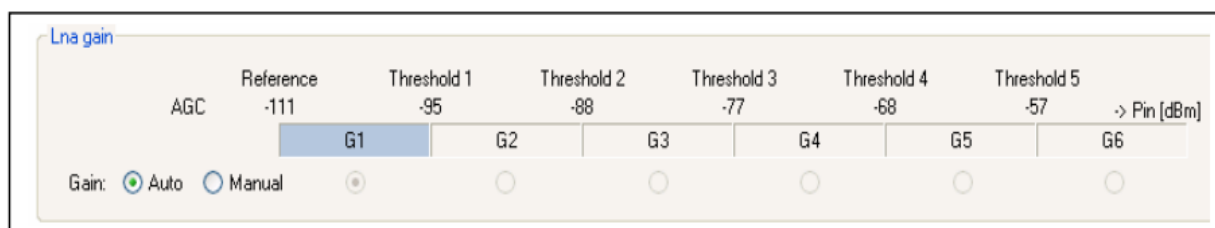
**Rx Bandwidth 窗口:** Rx Bandwidth 窗口标签显示如下，根据规格书上描述，FSK 模式下最佳接收带宽参数设置，可以直接通过窗口输入，DCC 频率约为接收带宽的 4% 比例，此无需设置，StarterKit 工具会根据所设置的接收带宽，自动匹配设置。



**LNA 增益窗口:** LNA 增益窗口提供了配置的接收 LNA 的状态，可以设置为手动调节或者自动调节，在自动模式下 LNA 是基于 RSSI 的值自动设定的，而手动模式下，增益应该根据预期的信号功率值来设定，一般建议选择自动模式。

注意，AGC 的阈值是根据以下公式设定的：

$$AgcRef = [-174 + NF + 10 * \log(2 * RxBw) + DemodSNR + FadingMargin] \text{ dBm}$$



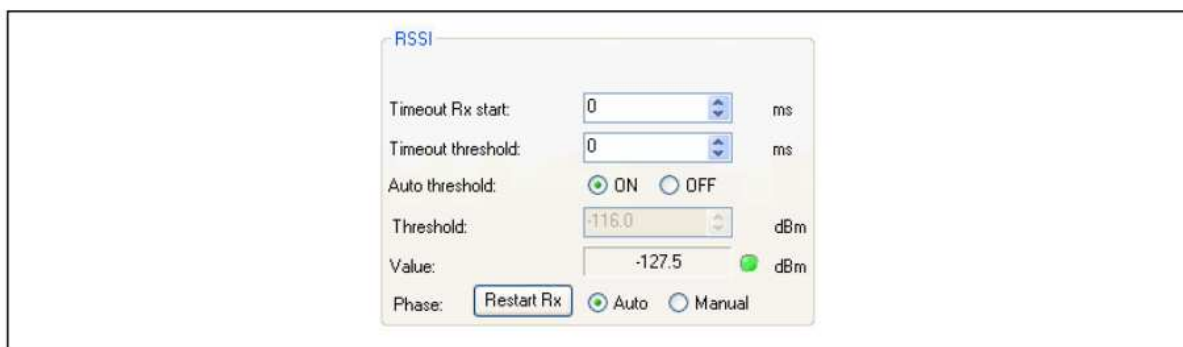
**AFC / FEI 窗口标签:** AFC / FEI (Frequency Error Indicator) 窗口标签的内容如下所示, 请注意当 AFC 被激活时 (无论设置为手动或自动), 在常规的寄存器配置窗口中 RF 频率寄存器表格中的值是没有实时更新的, 尽管收到的信号的频率可能是会有误差的, 可以通过 FEI 窗口的“读”按钮读出频率偏差的值。

使能 AFC low beta 功能实现双 AFC 功能来抵消本振所引起的频率偏差, 更多的参数设置信息请参照规格书中的功能描述。



**RSSI 窗口标签:** RSSI 窗口标签的内容如下所示, 可以访问到 RSSI 超时函数和 RSSI 阈值超时函数, 启用自动或手动 RSSI 阈值检测的编程, 注意不要将 RSSI 阈值与 LNA 或 AGC 阈值相混淆, RSSI 阈值是可以计算出来的, 等同于接收带宽和有效的噪声本底。

$$RSSI_{thres} = [-174 + NF + 10 * \log(2 * RxBw) + DemodSNR] \text{ dBm}$$



#### 5.6.4 Irq & Mapping 寄存器配置标签

**Irq & Mapping** 寄存器配置标签内容如图所示, 请参照 RF69 规格书中对各个 IRQ 中断源的映射内容以及 DIO 口线的相应配置状态, 在寄存器配置工具中可以通过下拉选项框来选择不同模式下的各个 DIO 口的状态映射内容。

注意在射频模块的包模式下, StarterKit 会自动调整 DIO 口线的中断指示状态。

参照下面的表 1 和表 2, 描述了 DIO 口在连续模式和包模式下各个模块状态时的配置状态, 及其端口输出的中断映射内容。

Mode	DIOx Mapping	DIO5	DIO4	DIO3	DIO2	DIO1	DIO0
Sleep	00	-	-	-	-	-	-
	01	-	-	-	-	-	-
	10	LowBat	LowBat	AutoMode	-	LowBat	LowBat
	11	ModeReady	-	-	-	-	ModeReady
Standby	00	ClkOut	-	-	-	-	-
	01	-	-	-	-	-	-
	10	LowBat	LowBat	AutoMode	-	-	LowBat
	11	ModeReady	-	-	-	LowBat	ModeReady
FS	00	ClkOut	-	-	-	-	PllLock
	01	-	-	-	-	-	-
	10	LowBat	LowBat	AutoMode	-	LowBat	LowBat
	11	ModeReady	PllLock	-	-	PllLock	ModeReady
RX	00	ClkOut	Timeout	Rssi	Data	Dclk	SyncAddr
	01	Rssi	RxReady	RxReady	Data	RxReady	Timeout
	10	LowBat	SyncAddr	AutoMode	Data	LowBat	Rssi
	11	ModeReady	PllLock	Timeout	Data	SyncAddr	ModeReady
TX	00	ClkOut	TxReady	TxReady	Data	Dclk	Plllock
	01	ClkOut	TxReady	TxReady	Data	LowBat	TxReady
	10	LowBat	LowBat	AutoMode	Data	LowBat	LowBat
	11	ModeReady	PllLock	TxReady	Data	PllLock	ModeReady

**Table 1: Continuous Mode DIO Mappings**

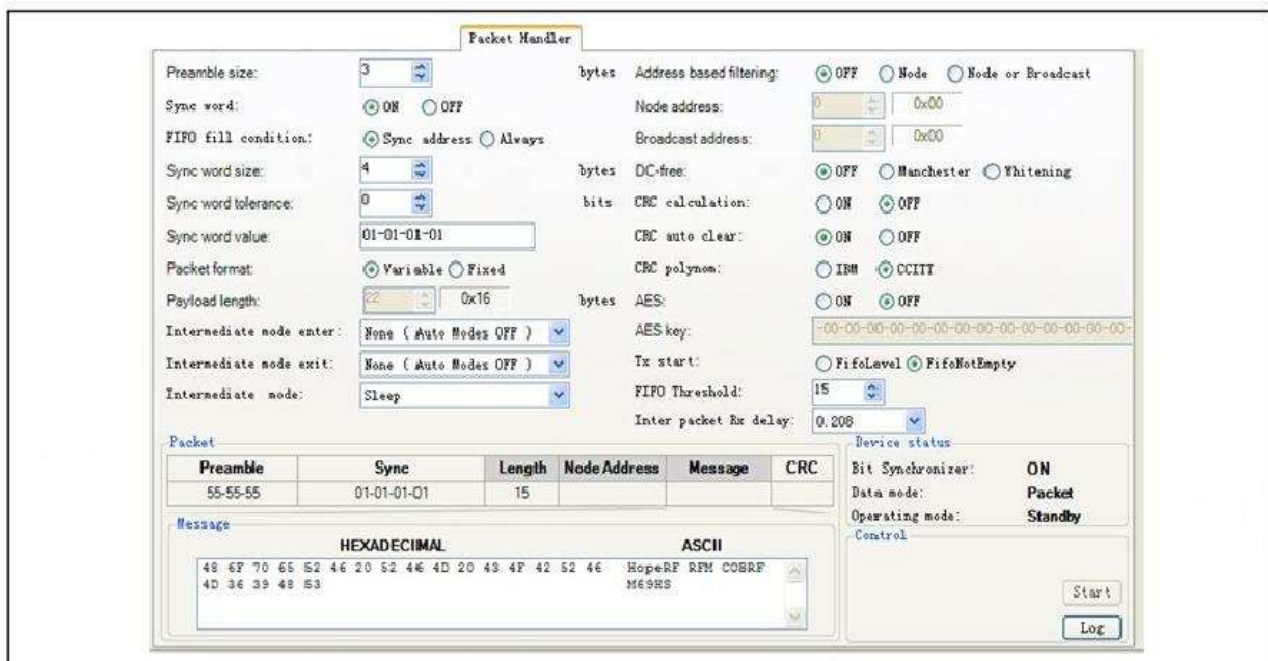
Mode	DIOx Mapping	DIO5	DIO4	DIO3	DIO2	DIO1	DIO0
Sleep	00	-	-	FifoFull	FifoNotEmpty	FifoLevel	-
	01	-	-	-	-	FifoFull	-
	10	LowBat	LowBat	LowBat	LowBat	FifoNotEmpty	LowBat
	11	ModeReady	-	-	AutoMode	-	-
Standby	00	Clkout	-	FifoFull	FifoNotEmpty	FifoLevel	-
	01	-	-	-	-	FifoFull	-
	10	LowBat	LowBat	LowBat	LowBat	FifoNotEmpty	LowBat
	11	ModeReady	-	-	AutoMode	-	-
FS	00	Clkout	-	FifoFull	FifoNotEmpty	FifoLevel	-
	01	-	-	-	-	FifoFull	-
	10	LowBat	LowBat	LowBat	LowBat	FifoNotEmpty	LowBat
	11	ModeReady	PllLock	PllLock	AutoMode	PllLock	PllLock
RX	00	Clkout	Timeout	FifoFull	FifoNotEmpty	FifoLevel	CrcOk
	01	Data	Rssi	Rssi	Data	FifoFull	PayloadReady
	10	LowBat	RxReady	SyncAddr	LowBat	FifoNotEmpty	SyncAddr
	11	ModeReady	PllLock	PllLock	AutoMode	Timeout	Rssi
TX	00	Clkout	ModeReady	FifoFull	FifoNotEmpty	FifoLevel	PacketSent
	01	Data	TxReady	TxReady	Data	FifoFull	TxReady
	10	LowBat	LowBat	LowBat	LowBat	FifoNotEmpty	LowBat
	11	ModeReady	PllLock	PllLock	AutoMode	Plllock	PllLock

**Table 2: Packet Mode DIO Mappings**



### 5.6.5 Packet Handler 寄存器配置标签

**Packet Handler** 寄存器配置标签内容如下所示，关于详细的包模式信息，可以参照规格书中对包模式的详细介绍，如下介绍寄存器配置标签的用法。



**Preamble size** 窗口标签，通过此标签的上下拉选框可以选择前导码的个数，也可以直接在窗口处输入所需要的前导码个数。注意为保证通讯质量前导码最少个数不能小于 2 个。

**Sync Word** 窗口标签，通过此标签可以选择同步字的个数和设置同步字的内容，注意



正确的包模式设定，同步字的个数不能少于 1 个 byte，且同步字的第一个 byte 的内容不能设置为 0x00，StarterKit 工具会自动提示出错当同步字的第一个 byte 被设为 0x00 的时候。

**FIFO fill** 填充条件设为同步字地址匹配才开启 FIFO 填充，数据包格式可设为可变包长或者固定包长的模式。

通过标签中其它选项的设定，还可选择开启广播码或子节地址码的功能，CRC 校验的功能和硬件 AES 加密的功能等。

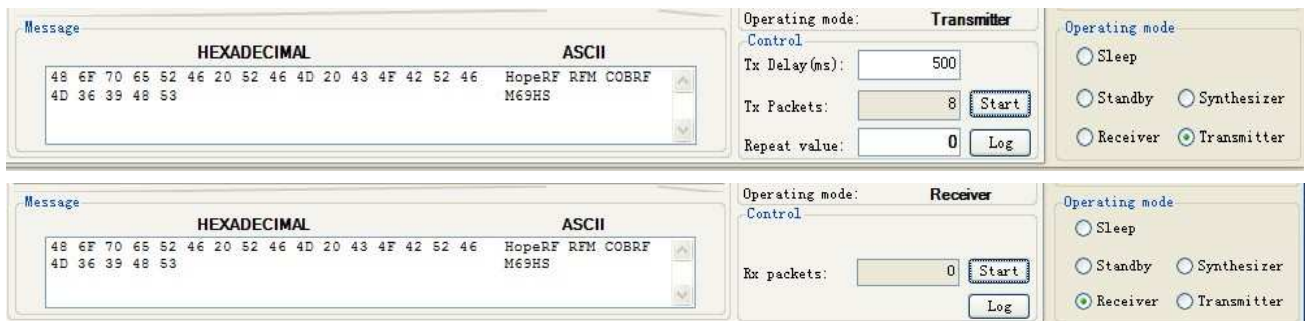
### Packet Handler StarterKit 使用时的限制

当操作 StarterKit 的 packet handler 窗口功能时，由于 StarterKit 软件本身的用法限制，要求在配置一些窗口功能时必须注意以下 2 点：

1. 最小前导码的个数为 2bytes；
2. 最大数据包长度为 66bytes；

## 6. Packet 模式下 StarterKit 工具的使用方法

**包模式下的工作方法：**通过 StarterKit 工具对射频模块的各项参数配置好以及包模式设定完成之后，通过工具右下侧的操作模式控制栏，选择模块所需要的工作状态，如图所设，选中接收或发射模式后，对应模式前的图符为选中状态，此时射频模块已进入相应的工作模式，在 control 控制栏中点击 start 按钮可以开启包模式的接收或发射的功能，并会对处理过的数据包进行计数显示，FIFO 接收到的或发射的数据内容，会显示在左侧的 message 缓冲区框图中：



**Registers Display 寄存器显示窗口：**如前面 5.2 节中介绍，点击菜单中的寄存器显示选项或者快捷键工具栏中的寄存器显示选项，打开寄存器显示窗口查看各寄存器的状态值，当配置寄存器标签中的任何标签选项有所改动时，寄存器显示窗口都会实时的更新到相应的寄存器的配置值中。

**Exit 退出：**使用完该寄存器配置工具后，可直接点击菜单中的退出选项或者直接关闭该工具窗口退出应用程序。

## 7. 更多模块的应用示例，请登陆 [www.hoperf.cn](http://www.hoperf.cn)