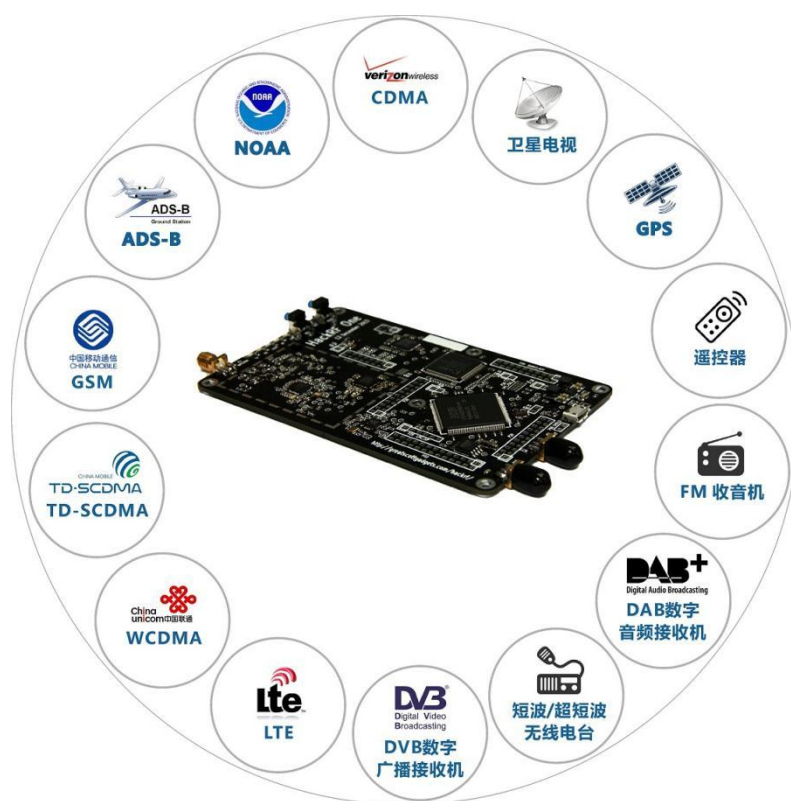


# 通信原理实验

## (SDR 部分)

金向东 龚淑君 王子立 编



浙江大学信息与电子实验教学中心

二〇二一年二月

## 目录

基于 SDR 的无线通信实验简介.....	1
1、 SDR 介绍.....	1
2、 USRP 介绍.....	1
3、 软件平台介绍.....	4
(1) GNU Radio.....	4
(2) Simulink.....	4
(3) LabView.....	4
4、 实验内容简介.....	5
实验讲义.....	11
实验一：LabVIEW 介绍.....	11
实验二：USRP 使用与单边带调制与解调.....	18
实验三：脉冲成形和匹配滤波.....	28
实验四：帧同步和非相干调制.....	34
实验五：ASK 与 FSK 调制.....	39
实验六：信道估算和纠正.....	44
实验 7 载波频率偏移的估算和纠正.....	49

# 基于 SDR 的无线通信实验简介

## 1、SDR 介绍

软件定义无线电（Software Defined Radio, SDR）的主要设计构想是将波形处理，如调制、解调等交给 CPU 实现，而将射频信号的收发，数字上下变频、抽样、插值等高速处理过程交给其硬件设备，该硬件设备也称 USRP（Universal Software Radio Peripheral，通用软件无线电外设），包含两个部分：进行高速信号处理的 FPGA 和覆盖不同频率范围 RF 前端。如图 1-1 所示。

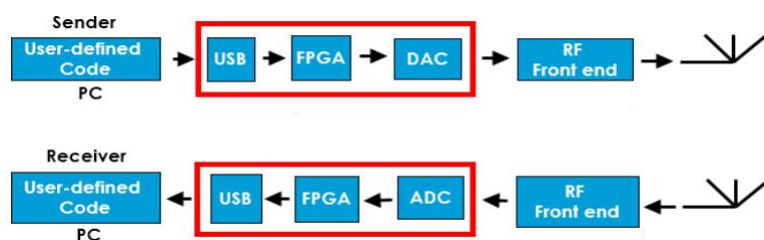


图 1-1 软件定义无线电（SDR）系统

基于软件无线电（SDR）的通信实验平台，能使学生有机会在实验室运用学习到的无线通信理论构建“真实”的通信系统，提供与真实射频信号的交互实验机会，激发学生的学习兴趣。

通过 SDR 的软硬件结合，学生可以创建整个通信系统信号链路中的各个功能模块，并观察各点的信号特征。通过该方法，学生可亲身体验并构建信道编码、调制、定时恢复、随机数生成等现代数字通信系统中常见的功能模块。最后，可在 USRP 射频发射机和射频接收机之间创建完成了一个无线通信链接。学生们仅在学习了射频和通信入门级课程之后，就能够独立地完成构建通信系统的项目。

## 2、USRP 介绍

USRP 可以支持两路并行的发送或者接收。RF 前端，完成射频信号和不同频带信号之间的转换，涵盖从直流到 6GHz 的整个范围，这包括了从调幅广播到超过 Wi-Fi 的所有频率。USRP-B210 是一款应用较广的 SDR 设备，其系统架构如图 2-1 所示。

USRP 的真正价值是它能使工程师和设计师以低预算和最少的精力进行创造。为数不少的开发者和用户贡献了大量的代码库，并为软件和硬件提供了许多实际应用。灵活的硬件、开源的软件和拥有经验丰富用户社区群的强强联合，使它成为您软件无线电开发与应用的理想外设。

USRP-B210 主要特性：

- 高度集成，两路收发，射频范围从 70 MHz–6 GHz；
- 支持高速的 USB 3.0 连接；
- 开源的 UHD 支持多种框架；
- 用户可编写的 FPGA，一个完全集成的混合信号基带直接转换收发器，可提供到 56MHz 的带宽。

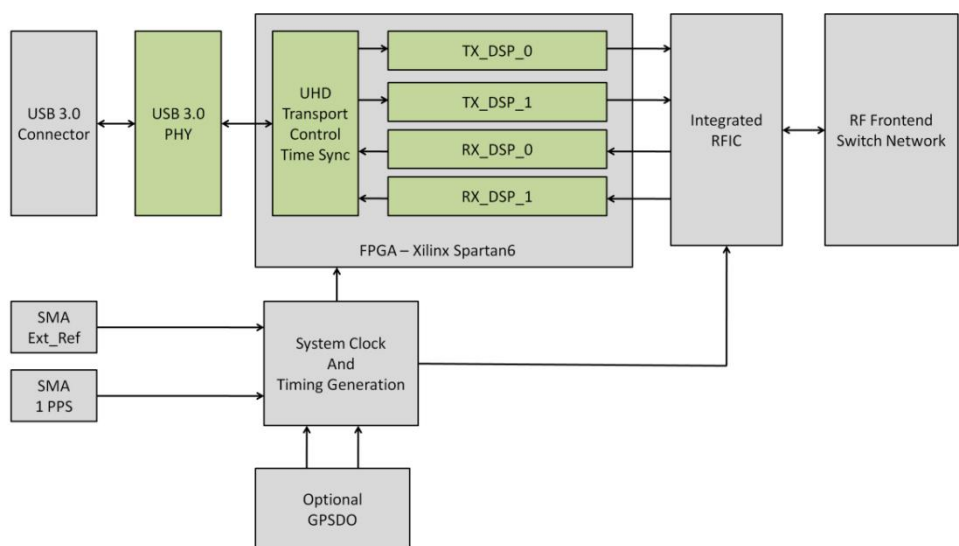


图 2-1 USRP-B210 设备系统架构

USRP- B210 技术参数

参数类别	数值	单位	参数类别	数值	单位
输入\输出			单通道射频性能参数		
直流电压输入	6	V	单边带信号/镜像 抑制	-35/50	dBc
转换模块参数			3.5GHz	1.0	degRMS
ADC 采样速率(最大)	61.44	MS/s	6GHz	1.5	degRMS
ADC 分辨率	12	bits	输出功率	>10	dBm
ADC SFDR	78	dBc	输入三阶截取点	-20	dBm
DAC 采样速率	61.44	MS/s	噪声系数	<8	dB
DAC 分辨率	12	bits	物理属性		
与主机最大速率(16b)	61.44	MS/s			
本振精度	±2.0	ppm			
GPS 未锁定 TCXO 精度	±75	ppb			
GPS 锁定 TCXO 精度	<1	ppb	尺寸	9.7x15.5x1.5	cm
			重量	350	g

B210 采用了射频前端集成芯片 AD9361 设计，AD9361 支持射频端直接上下变频，可实时提供 56MHz 的带宽。采用 AD9361 的两路信号链，提供相干 MIMO 能力。主机通过 USB 3.0

对型号为 Spartan6 XC6SLX150 的 FPGA 一系列操作来实现对 AD9361 中信号的处理和控制。B210 最大提供 61.44MS/s 的吞吐量，利用 UHD 的 API 与 PC 相连。

利用 B210 可以实现十分广泛的应用，包括广播，手机，GPS，WiFi，ISM，FM 和 TV 信号等等。用户可以利用 GNU Radio，LabVIEW 等软件很快的进行一些无线电的开发，或者参加一些开源的软件无线电项目。基于 UHD 的支持，允许用户重用现有的设计，例如 HSDR 和 OpenBTS 两个开源的应用均支持。

基于 USRP 的无线通信实验套件如图 2-2 所示，其中：

- ① USRP-B210 设备
- ② AC/DC 电源转换器
- ③ USB3.0 电缆
- ④ SMA 转 SMA 接口电缆
- ⑤ 30dB SMA 衰减器
- ⑥ 实验指导书

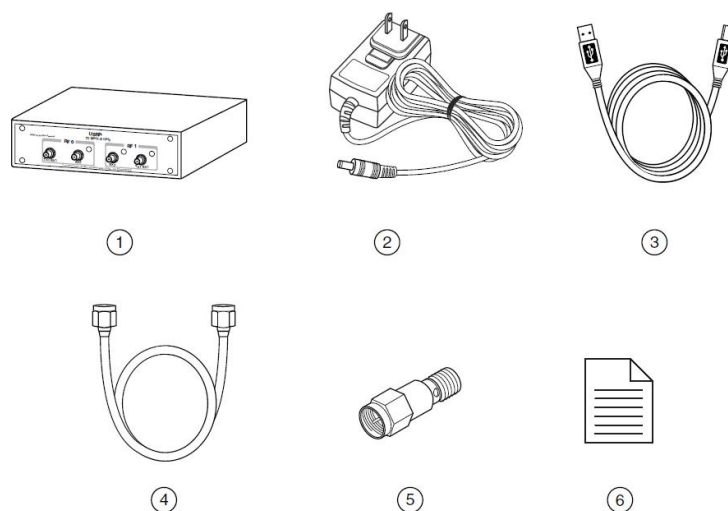
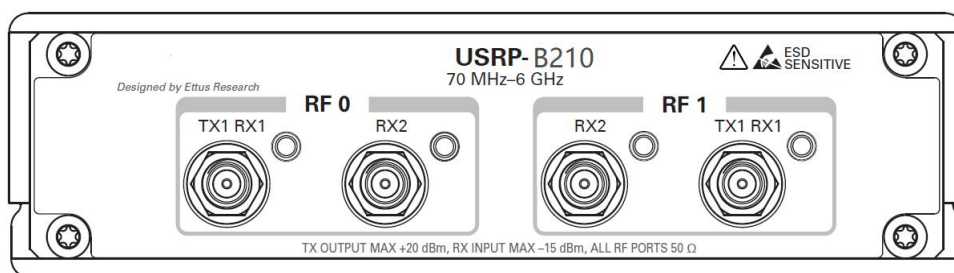


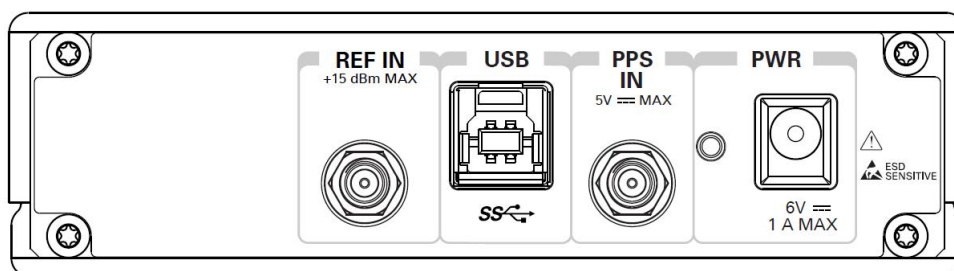
图 2-2 基于 USRP 的无线通信实验套件

USRP-B210 前后面板布局，如图 2-3 所示。前面板中，有 RF0 和 RF1 两个通道。每通道中的 TX1/RX1 为射频输入/输出接口，负载阻抗 50 欧姆，单端口；RX2 为射频输入接口，负载阻抗 50 欧姆，单端口。相应的指示灯可指示各端口的工作状态。

后面板中的 REF IN 是本振参考信号的输入端，USB 接口用于连接电脑，PPS IN 接口可连接外部的秒脉冲（PPS）参考信号，PWR 接口在需要时可连接 6V/3A 的电源适配器（功耗较小时，直接由 USB 供电）。



前面板



后面板

图 2-3 USRP 面板结构图

### 3、软件平台介绍

#### (1) GNU Radio

GNU Radio 是一个开源的软件定义无线电 (SDR) 平台。它有一个世界范围内的开发者和用户社区为其贡献了坚实的基础代码，并提供了许多软件和硬件的实际应用。它提供了一个完整的开发环境，以创建您自己的无线电，为您处理所有的硬件接口、多线程、可移植性的问题。GNU Radio 提供所有通用软件无线电需要的库，包括各种调制方式 (GMSK、PSK、QAM、OFDM 等)、纠错码 (R-S 码、维特比码、Turbo 码)、信号处理模块 (最优滤波器、FFT、均衡器、定时恢复) 和调度。它是一个很灵活的系统，允许用户使用 C++ 或者 Python 开发应用程序。GNU Radio 支持包括 Windows、Linux、Mac OS 等主流平台

#### (2) Simulink

Simulink 是 MATLAB 中的一种可视化仿真工具，是一种基于 MATLAB 的框图设计环境，是实现动态系统建模、仿真和分析的一个软件包，被广泛应用于线性系统、非线性系统、数字控制及数字信号处理的建模和仿真中。Simulink 可以用连续采样时间、离散采样时间或两种混合的采样时间进行建模，它也支持多速率系统，也就是系统中的不同部分具有不同的采样速率。为了创建动态系统模型，Simulink 提供了一个建立模型方块图的图形用户接口 (GUI)，这个创建过程只需单击和拖动鼠标操作就能完成，它提供了一种更快捷、直接明了的方式，而且用户可以立即看到系统的仿真结果。

#### (3) LabVIEW

LabVIEW 是一种程序开发环境，由美国国家仪器 (NI) 公司研制开发的，类似于 C 和 BASIC 开发环境，但是 LabVIEW 与其他计算机语言的显著区别是：其他计算机语言都是采用基于文本的语言产生代码，而 LabVIEW 使用的是图形化编辑语言 G 编写程序，产生的程

序是框图的形式。与 C 和 BASIC 一样，LabVIEW 也是通用的编程系统，有一个完成任何编程任务的庞大函数库。LabVIEW 的函数库包括数据采集、GPIB、串口控制、数据分析、数据显示及数据存储，等等。LabVIEW 也有传统的程序调试工具，如设置断点、以动画方式显示数据及其子程序（子 VI）的结果、单步执行等等，便于程序的调试。

LabVIEW（Laboratory Virtual Instrument Engineering Workbench）是一种用图标代替文本行创建应用程序的图形化编程语言。传统文本编程语言根据语句和指令的先后顺序决定程序执行顺序，而 LabVIEW 则采用数据流编程方式，程序框 LabVIEW 提供很多外观与传统仪器（如示波器、万用表）类似的控件，可用来自方便地创建用户界面。用户界面在 LabVIEW 中被称为前面板。使用图标和连线，可以通过编程对前面板上的对象进行控制。这就是图形化源代码，又称 G 代码 LabVIEW 的图形化源代码在某种程度上类似于流程图，因此又被称作程序框图代码。

特性：

- 尽可能采用了通用的硬件，各种仪器的差异主要是软件；
- 可充分发挥计算机的能力，有强大的数据处理功能，可以创造出功能更强的仪器；
- 用户可以根据自己的需要定义和制造各种仪器。

## 4、实验内容简介

### (1) USRP 边带调制与解调实验

结合 labview 设计软件和 USRP 外设，可以产生并发送任意频率的载波信号；发送和接收单边带调制信号、双边带调制信号。图 3-1 为 LabVIEW NXG 平台设计的系统电路图。

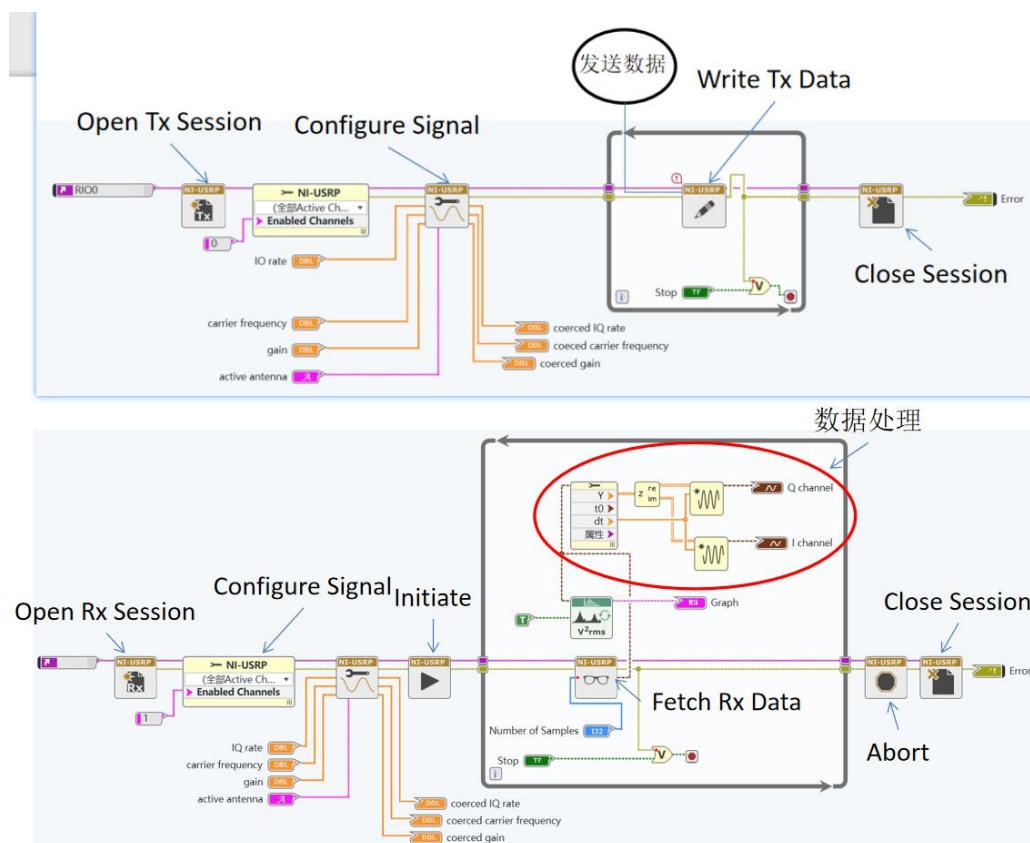




图 3-1 LabVIEW 软件设计的发送/接收电路

使用电缆将 USRP 外设的信号发送端连接到频谱仪，可以观察发送信号的频谱。图 3-2.a 是用频谱仪观测到的载波信号的频谱。图 3-2.b 是用频谱仪观测到的上边带调制信号的频谱。

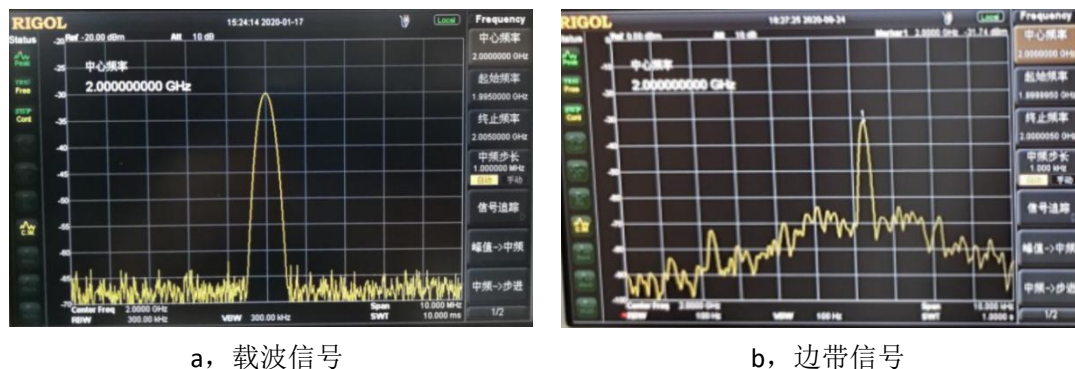


图 3-2 载波和边带信号频谱

运行设计电路，也可以在 labview 前面板得到仿真波形。如图 3-3 所示：是接收到的上边带信号功率谱以及 I/Q 两路正交信号的时域波形。

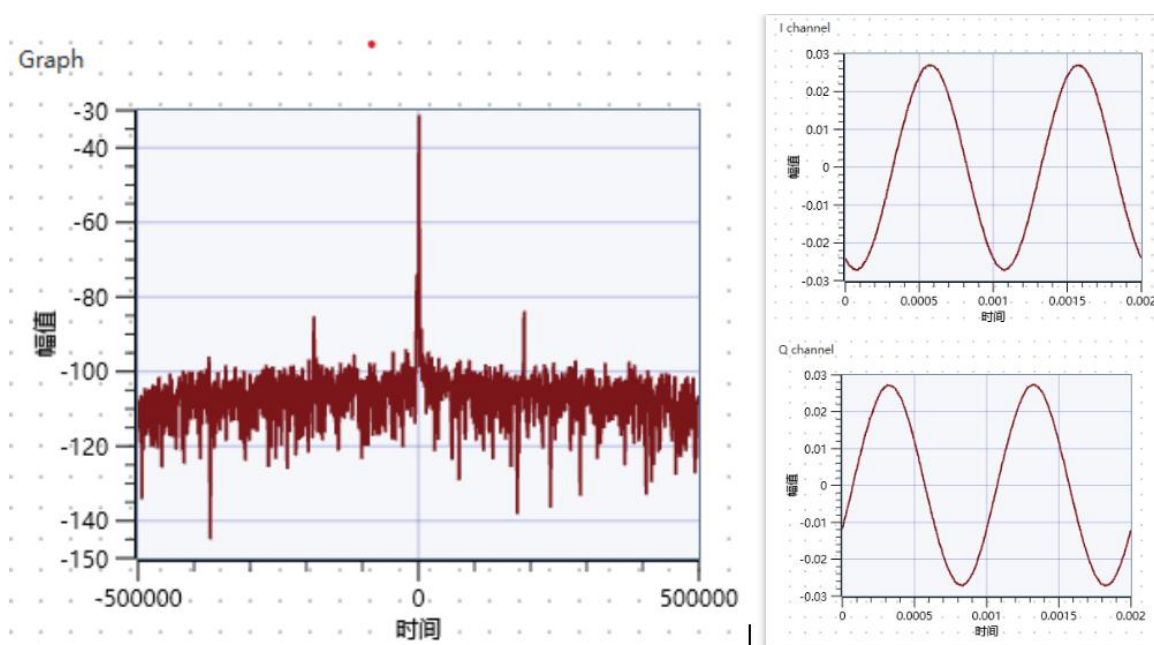


图 3-3 LabVIEW 面板显示的频域时域波形

## (2) 脉冲成形和匹配滤波实验

进行数字信息的脉冲成形发送和匹配滤波接收，数据接收增加脉冲对齐功能。脉冲成形方式有：升余弦、均方、高斯。

图 3-4 是发送端数据经过升余弦脉冲成形后的功率谱和它的眼图。



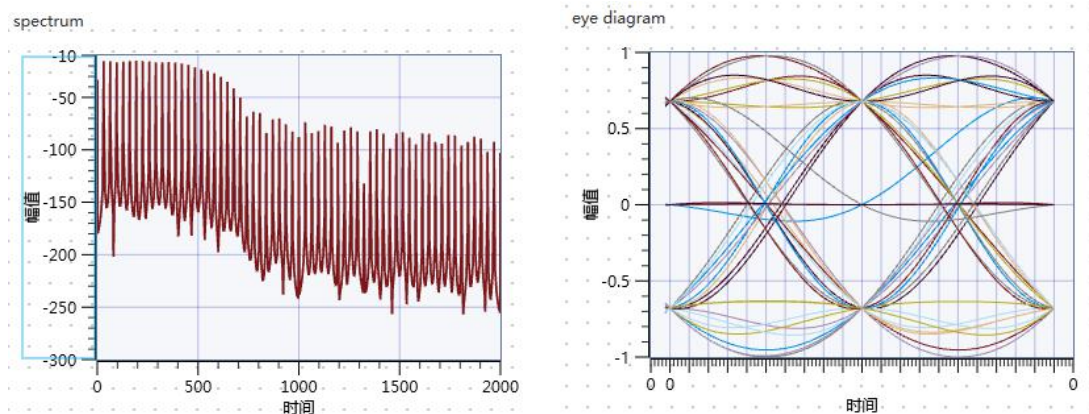


图 3-4 发送端频谱与眼图

图 3-5 是接收端在没有进行脉冲同步和进行脉冲同步后的眼图。

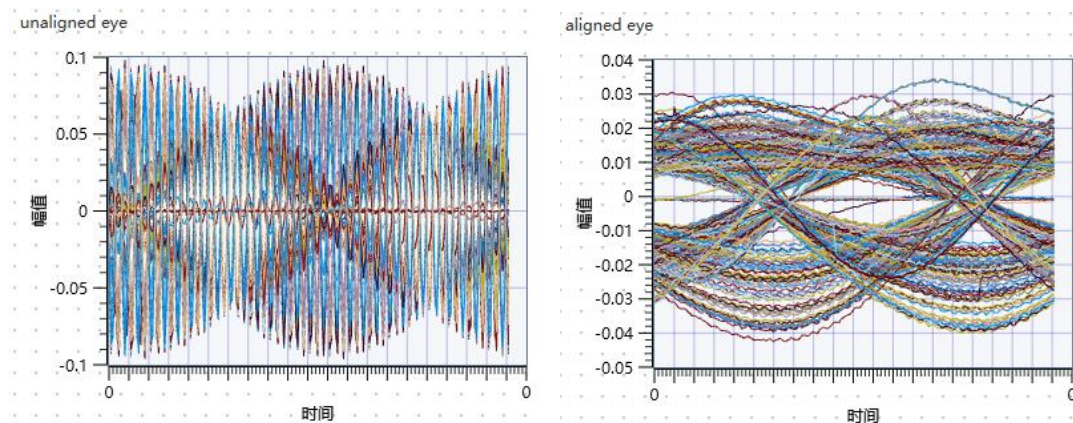


图 3-5 接收端眼图

### (3) 帧同步和非相干调制实验

在发送端，对发送数据进行 DBPSK 编码，再经过脉冲成形发送；在接收端，对数据进行匹配滤波接收，进行帧同步，再进行 DBPSK 解码。

### (4) ASK、FSK 调制实验

ASK 调制发送和接收解调，FSK 调制发送和接收解调。图 3-6 是 ASK 在接收端各个节点处的信号仿真波形。

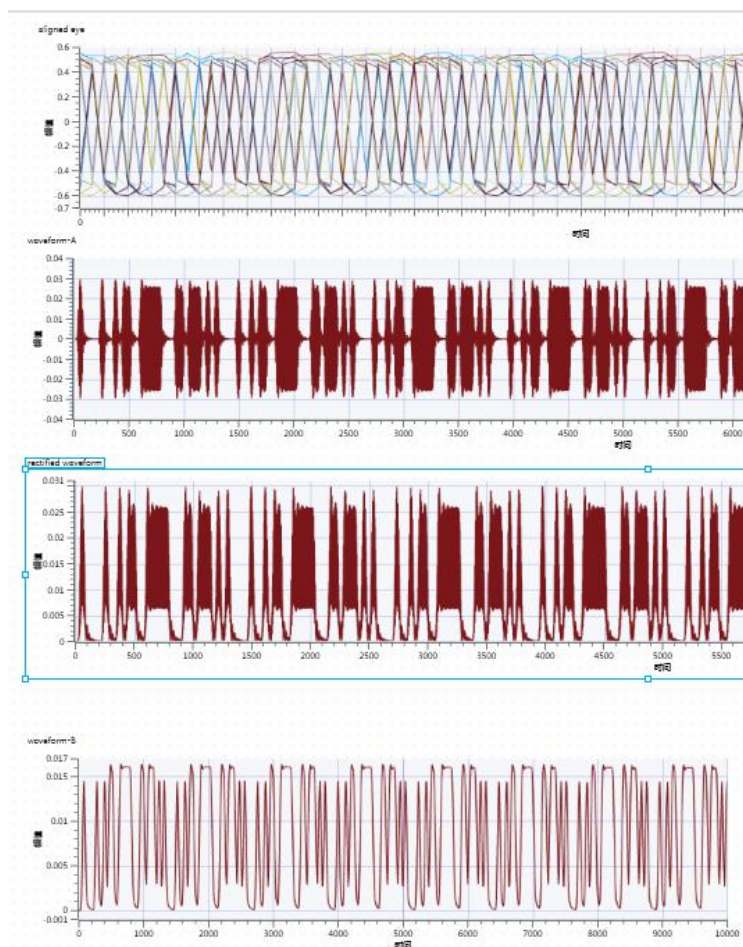


图 3-6 接收端测试点的信号

图 3-7 是在 FSK 发送端的信号频谱，图 3-8 是在接收端各个节点处的信号波形。



图 3-7 FSK 发送频谱

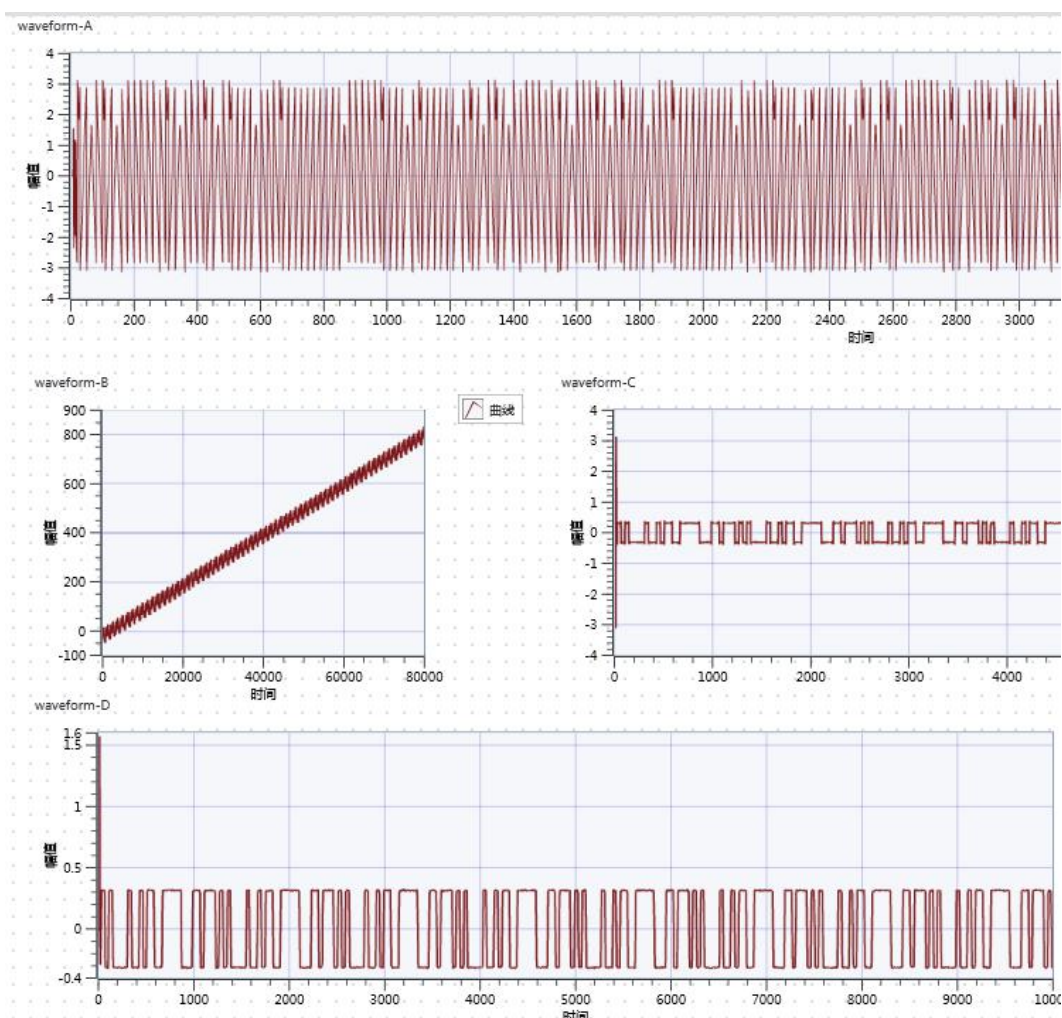


图 3-8 接收端信号波形

### (5) 信道估算和纠正

对信道进行估算和纠正，并计算信噪比。信道估算使用线性最小二乘法估算，以及加性高斯白噪声信道中的最大似然估计。在信道纠正实验中，运行 BPSK 接收电路获取星座图，星座图以一个信道产生的随机相位数旋转。运行窄带相位估算（相位长度  $L=1$ ），纠正信道以补偿相位旋转。

### (6) 载波频率偏移的估算和纠正

主要包括自参考帧同步，载波频率偏移的估算和纠正三部分内容。再将载波频率偏移纠正的结果用于 BPSK 信道纠正中。

### (7) QPSK、MPSK、MQAM 调制实验

对发送数据作 4-PSK、16-PSK、4-QAM、16-QAM 编码，观察信号的星座图，在接收端加入加性高斯白噪声，测量信噪比和误码率。

## (8) OFDM 实验

构建 OFDM 发送和接收各部分模块电路，以及 OFDM 信道估算和纠正电路，搭建一个 OFDM 数据发送接收仿真系统以验证各个模块功能的正确性。最后将 OFDM 应用在 USRP 数据发送接收系统中。

## (9) LoRa 实验

构建 LoRa 发送和接收各部分模块电路，搭建一个 LoRa 发送接收仿真电路以验证各个模块功能的正确性。最后将 LoRa 应用在 USRP 数据发送接收系统中。



# 实验讲义

## 实验一：LabVIEW 介绍

### 一、实验目的

- (1) 熟悉 LabVIEW 通信设计套件
- (2) 会使用 While/For 循环处理数据
- (3) 熟悉多种数据类型
- (4) 会使用数组，访问和处理元素

### 二、实验设备

- (1) 安装 LabVIEW 环境的电脑 1 台

### 三、实验概要

本实验课程的目的是在通用软件无线电外设（USRP）上应用、实践关于软件无线电的数字通信理论。USRP 结合了 LabVIEW 软件，该软件是一个使用模块和连线的图形编程软件。因此有必要熟悉 LabVIEW 语言并能写一些代码。

### 四、实验内容与步骤

#### 1. 开始

- 1) 运行电脑中的软件：NI LabVIEW NXG 2.1
- 2) 新建一个项目：点击文件/新建/项目
- 3) 在项目中添加一个新的 VI：点击文件/新建/VI
- 4) 将项目重命名为 Lab1：在窗口左上角右击未命名项目并点击重命名，窗口如图 1-1 所示：

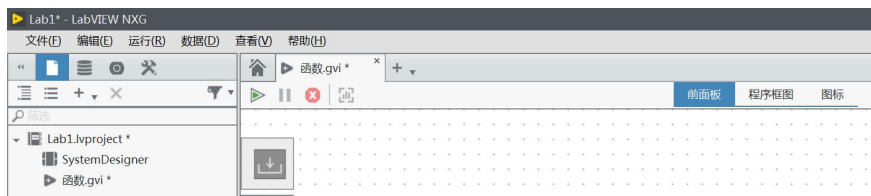


图 1-1 新建项目

- 5) 每一个 VI 有两个窗口：前面板和程序框图。前面板是用户和 VI 进行交互的地方，它包括很多控制输入和显示输出参数端口。程序框图窗口包含从控制输入端接收输入数据，通过连接模块进行处理，并且通过显示输出进行显示的代码（流程图）。

#### 2. While 循环

While 循环是一个控制流语句，用于重复执行一个函数，直到满足给定的条件终端。条件终端定义循环结束的时间。条件终端有两种设置：如果为 True 则继续和如果为 True 则停止。如果设置为 True 继续，则仅当布尔控件返回 True 时才进行循环。如果设置为 True 则停止，布尔值为 True 时，循环将中止。默认情况下，条件终端是“True 则停止”。

##### 2.1 While 循环示例

使用 while 循环，产生一组 0 到 10 之间的随机整数，当产生的数与用户给定的数值匹

配是，停止产生随机数。

- 1) 新建一个 VI
- 2) 在程序框图窗口放置一个 **while** 循环（左侧程序流模块栏里选取）
- 3) 放置随机数模块产生随机数（数据类型/数值/随机数），产生 0 到 1 之间的随机浮点数。
- 4) 为了产生 0 到 10 之间的随机数，放置乘法模块和数值常量输入控件以及最近数取整模块（数据类型/数值）。
- 5) 创建一个数值显示输出端口，显示每一次循环产生的随机数：放置一个数值接线端（数据类型/数值），这个接线端默认的行为特性是“控制”（从前面板输入），需要改变其特性为“显示”（输出到前面板），重命名显示端口为 **Random Integer**。
- 6) 创建一个数值输入控制端口（数值接线端），接收匹配的整数。重命名控制端为 **Number to match**。
- 7) 使用“等于？”模块（数学/比较），比较两个数值的大小。
- 8) 在前面板窗口创建一个停止按钮（按钮/停止按钮），回到程序框图窗口，在未放置项将停止控件拖放在框图中。
- 9) 放置“或”模块（数据类型/布尔），将布尔控制两个输入分别与比较器的输出、停止按钮连接。当输出随机数与输入匹配或者按下停止按钮，循环将中止。
- 10) 回到前面板，将未放置项拖放进窗口界面。前面板和程序框图如图 1-2 所示：

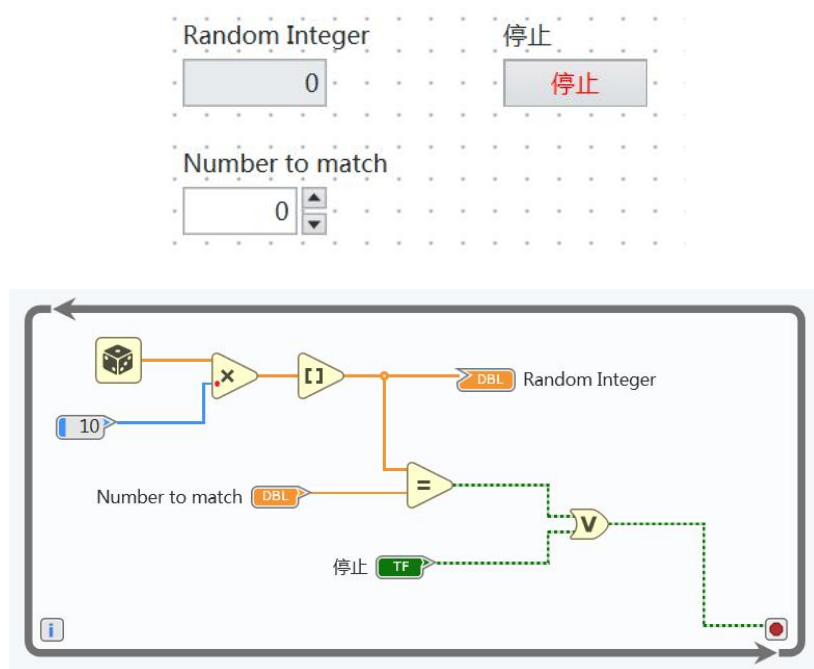


图 1-2

- 11) 在前面板输入一个匹配数值，运行程序，观察结果。
- 12) 在程序框图窗口上方菜单条中，点击“高亮显示执行结果”，运行程序的时候，可以观察到程序框图中数据流的变化。

## 2.2 问题 1

对于以下问题，在实验报告中都需要提供程序图和前面板图。

- 1) 完成以上示例。
- 2) 给 **while** 循环增加一些延时（比如 1 秒钟），以便能观察到数据的变化。



- 3) 增加一个布尔指示器，当随机数大于 5 时打开，小于或等于 5 时关闭。
- 4) 增加一个输出端显示循环的次数。

### 3. 数据类型

在 LabVIEW 中，每一个对象和连线都跟数据类型有关系。LabVIEW 支持很多数据类型，它们由不同的颜色、形状区分。部分数据类型如图 1-3 所示：



图 1-3 部分数据类型

#### 3.1 示例

下面列举簇数据类型的使用：接收一串字符串，将其反转；对给定的数组元素数值增加偏置量。

- 1) 新建一个 VI
- 2) 在前面板，创建一个空白簇，如图 1-4 所示。



图 1-4

- 3) 在空白簇中放置一个字符串输入控件、一个数值输入控件和一个数值数组输入，将数组尺寸改为 5（可通过鼠标拖拉数组外框或直接在属性里修改）。在簇外面再放置一个字符串输入控件和一个数值数组输入，将它们改为显示模式。如图 1-5 所示：
- 4) 在程序框图窗口将未放置项放置在框图界面里，在程序框图中放置簇属性模块（数据类型/簇），以便从簇中读取各个数值。将簇控件与簇属性模块相连，鼠标往下拖拉簇属性模块外框显示更多的属性端口。

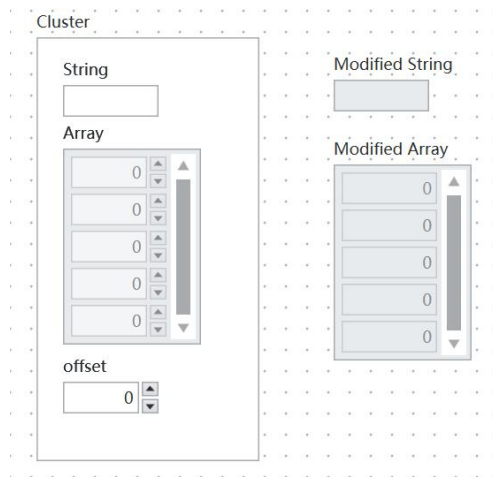


图 1-5

- 5) 使用反转字符串函数（数据类型/字符串）反转字符串，使用“加”函数（数学/数值）将偏置与数组相加。如图 1-6 所示：

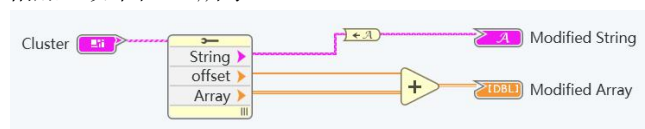


图 1-6

- 6) 在前面板输入控制值，运行程序，通过显示输出确认结果是否正确。  
7) 将改变的值再结合放入簇：复制、黏贴前面板中的簇并将它的属性改为显示。如图 1-7 所示：

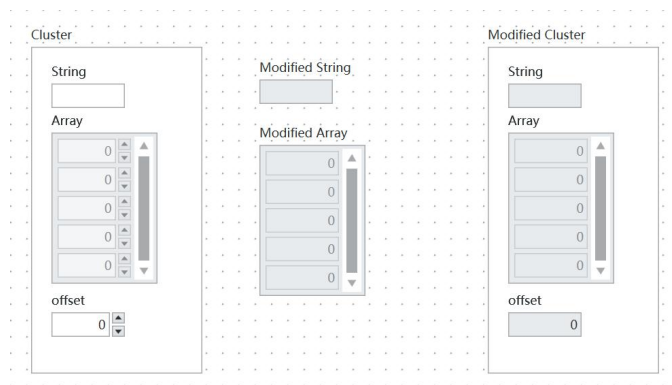


图 1-7

- 8) 将新的簇（未放置项）放置在程序框图中，为了将改变的字符串、数组和新的簇结合在一起，需要再次用到簇属性模块。因为要将数据写入簇，所以需要将簇属性设置为“全部为写入”。最终的程序框图如图 1-8 所示：

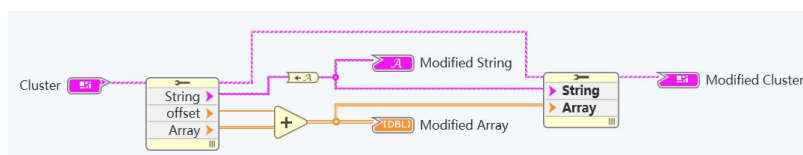


图 1-8

9) 运行程序并比较结果。

### 3.2 问题 2

- 1) 完成上面的示例，在实验报告中提供程序框图和前面板运行结果。
- 2) 是否能将一个整型控制输入与双精度类型显示输出相连？整型输入能否与字符串输出相连？如果不可以，需要使用什么模块进行类型转换？

## 4. 数组

数值中的元素都是相同类型的数据，可以是数值、布尔、路径、字符串、波形和簇。当处理一批类似数据或作重复计算时，可以考虑使用数组。存储波形数据或循环中产生的数据时，使用数组是一个理想选择。

### 4.1 示例

下面举例数值的使用：给定一个数值数组，找出其中的最大元素。

- 1) 新建一个空白 VI
- 2) 在前面板添加一个数值数组输入，通过往下拖拉数组外框将元素数量增加到 5。将数值数组输入重命名为“Array”
- 3) 添加一个数值输入控件，将它改为显示模式并重命名为“Maximum value”。
- 4) 在前面板添加一个 For 循环，将“Array”终端放在循环外面，“Maximum value”终端放在循环里面。
- 5) 将数值终端与 For 循环的左边框相连，在边框上会产生一个“隧道”，隧道使得循环与外面的模块能够进行数据通信。隧道有索引使能功能，当它设置为自动索引使能时，输入循环结构的数组元素在每一次循环的时候送入一个元素。如果索引没有使能，循环结构将一次调用数组中的全部元素。
- 6) 为了在 For 循环中对输入数组的元素逐个跟设定的最大值进行比较，需要在 For 循环中添加一个“大于？”函数（数学/比较）。再放入一个“选择”模块（程序流）。
- 7) 假如当前输入循环的数组元素比最大值大，需要更新最大值。将隧道输出与“大于？”模块的 X 端相连，同时连接到“选择”模块的真值输入端。将“大于？”模块的输出与“选择”模块的选择端连接。
- 8) 我们需要存储当前的最大值以备下次循环迭代的时候使用。移位寄存器是将两次相邻迭代联系起来的数据传输通道。将“选择”模块的输出连接到循环的右侧边框线上，会默认建立一个“隧道”，点击鼠标右键，将它改为移位寄存器模式，这时鼠标光标也变成移位寄存器的图标，点击循环的左边框，完成移位寄存器的放置。（移位寄存器的左侧方框可以看作是来自上一次迭代结果的输入，右侧方框是本次迭代的输出）程序框图如图 1-9 所示：

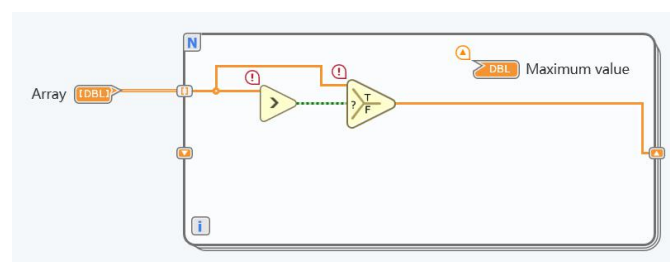


图 1-9

- 9) 移位寄存器的左侧方框是上一次迭代产生的最大值，用来与当前输入的数组元素进行比

- 较。将移位寄存器的左侧方框与“大于？”模块的Y端以及“选择”模块的假值连接。
- 10) 将“选择”模块的输出与 Maximum value 显示控件相连。完整的程序框图如图 1-10 所示：

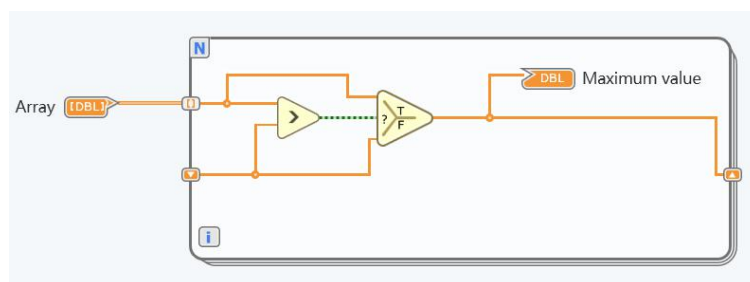


图 1-10

- 11) 在前面板给 array 输入数值，运行程序确认是否正确。

#### 4.2 为题 3

- 1) 完成上面的示例，在实验报告中提供程序框图和前面板运行结果。
- 2) 将数组元素最大值改为 0，再次运行程序，还能找到最大值吗？更改例程，使得每运行一次程序都能找到最大值。
- 3) 在问题 2 的基础上，更改程序框图，能够得到最大值的索引值（在数组中的位置）。结果可参考下图：

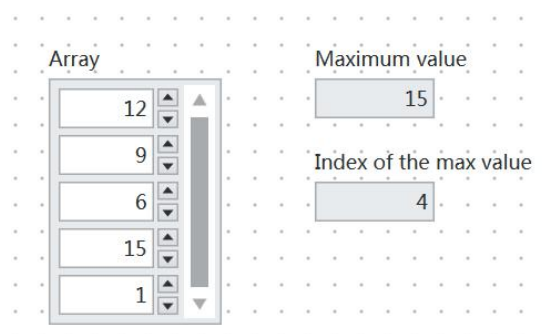


图 1-11

- 4) 完成以下 VI：产生两个数值范围在-0.5 到 0.5 之间的随机数，比较这两个随机数值的大小，如果第一个数比第二个数大，Team A 得分；如果第二个数比第一个数大，Team B 得分。当其中一个队得分达到 50 时，程序停止运行。显示最终的两个随机数、循环的次数、Team A 和 Team B 的得分以及哪一对赢。前面板如下图所示：

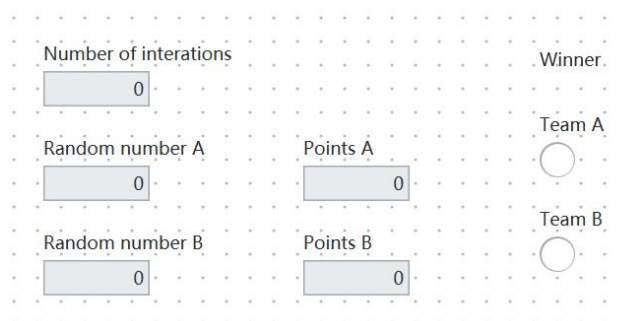


图 1-12

## 五、实验报告要求

- 1) 给出所设计的电路，记录实验数据并加以分析
- 2) 回答所有问题
- 3) 报告在下个实验开始前以书面或电子版形式递交

## 实验二：USRP 使用与单边带调制与解调

### 一、实验目的

- (1) 了解 USRP 设备架构，熟悉其使用方法
- (2) 掌握基于 SDR 的单边带调制与解调实现方法
- (3) 掌握基于 USRP 的发射机与接收机的实现方法

### 二、实验设备

- |                      |     |
|----------------------|-----|
| (2) USRP 设备          | 1 台 |
| (3) 安装 LabVIEW 环境的电脑 | 1 台 |
| (4) 频谱分析仪            | 1 台 |

### 三、实验概要

#### 1. USRP 架构

USRP 硬件平台遵循一个通用的软件定义无线电体系结构，使用高速模拟数字转换器 (ADC) 和数字模拟转换器 (DAC) 实现了一个直接变换模拟前端，该前端具有数字下变频 (DDC) 和数字上变频 (DUC)。接收链以一个高度灵敏的模拟前端开始，该前端能够接收非常小的信号，并且将信号直接下变频到同相 (I) 和正交 (Q) 两路基带信号。下变频之后是高速模数变换和 DDC，以降低采样率并将 I 路和 Q 路基带信号打包传输给主机作进一步处理。发送链从主机开始，在主机上生成 I 路和 Q 路基带信号，并通过以太网电缆传输到 USRP 硬件。信号经由 DUC 送给 DAC，之后再进行 I/Q 上变频混频，产生射频信号，经过放大再进行传输。USRP 硬件结构图如图 2-1 示：

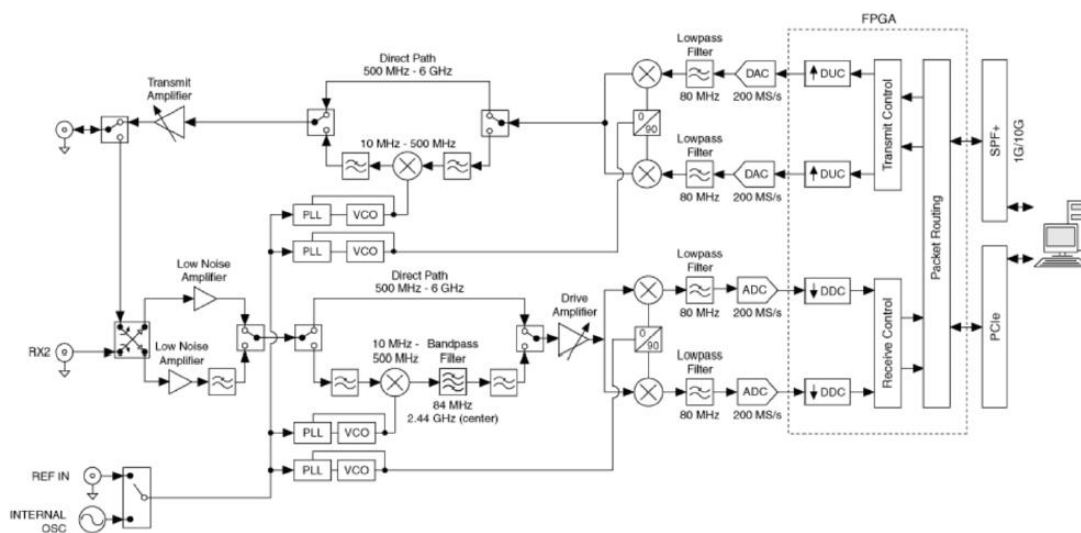


图 2-1 USRP 硬件结构图

#### 2. 基本的发送电路及其模块

LabVIEW 通过 USRP 模块构建发送电路，基本的传输发送电路如图 2-2 所示：



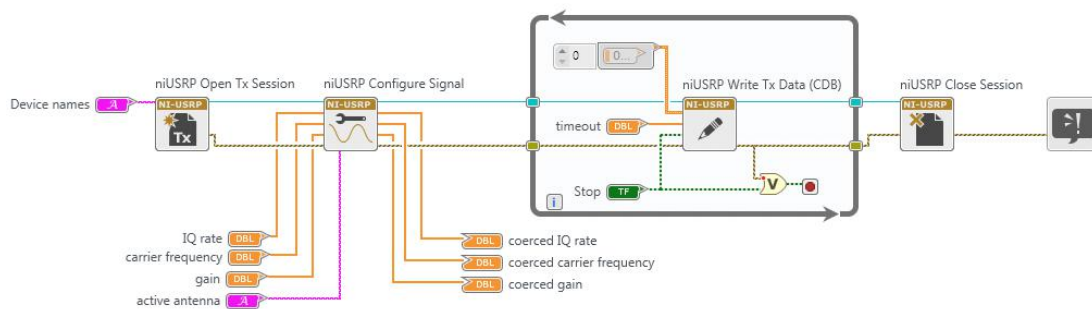


图 2-2 USRP 基本传输电路

发射机相关的模块在程序框图界面左侧菜单条下的硬件接口/Wireless Design and Test/NI-USRP/Tx 中，如图 2-3 所示。包含下列模块：

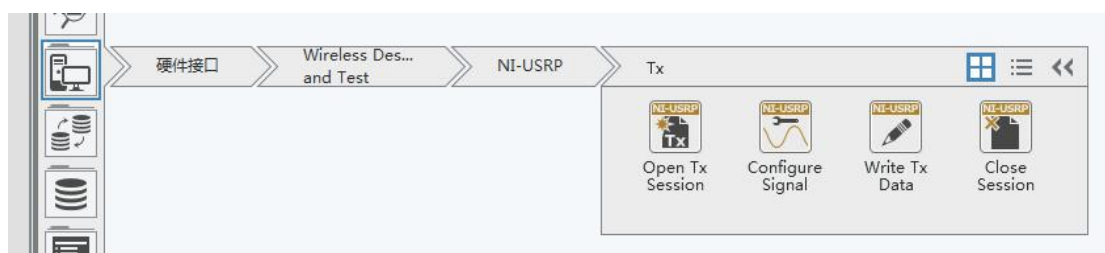


图 2-3 发射机模块提取界面

#### (1) Open Tx Session:

对设备名称输入中指定的设备打开一个传输会话，并将会话句柄作为输出返回。在模块的 device name 输入端需要输入设备名称，用于通知 Labview USRP 的 IP 地址或资源名称。

#### (2) Configure Signal:

配置 Tx 或 Rx 的参数。I/Q rate 是基带 I/Q 数据每秒采样的采样率。carrier frequency 是射频信号的载波频率（以 Hz 为单位）。gain 是射频信号的 Tx 增益（以 dB 为单位）。active antenna 指定传输信道使用的设备天线端口。Coerced IQ rate/carrier frequency/gain 是设备支持的对应的实际值。

#### (3) Write Tx Data:

将 16 位复数有符号整数写入到指定信道。基带采样数据以 16 位复数有符号整数数组形式传输。数据的实部和虚部分别对应同向和正交数据，在数组中，同向和正交数据交替排列。

#### (4) Close Session:

关闭设备的会话句柄。

### 3. 基本的接收电路及其模块

基本的接收电路如下图 2-4 所示：

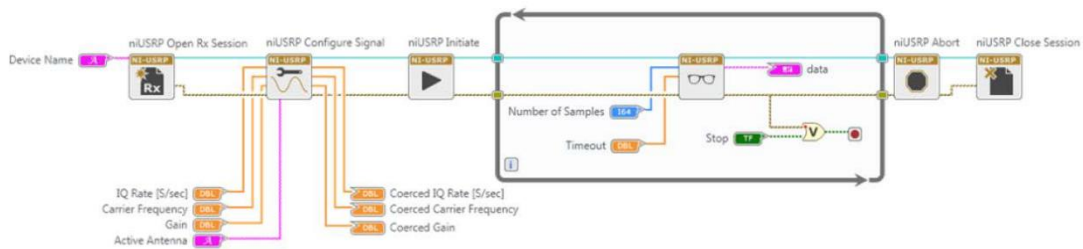


图 2-4 USRP 基本接收电路

接收机相关模块在程序框图界面左侧菜单条下的硬件接口/Wireless Design and Test/NI-USRP/Rx 中，如图 2-5 所示。主要有如下模块：

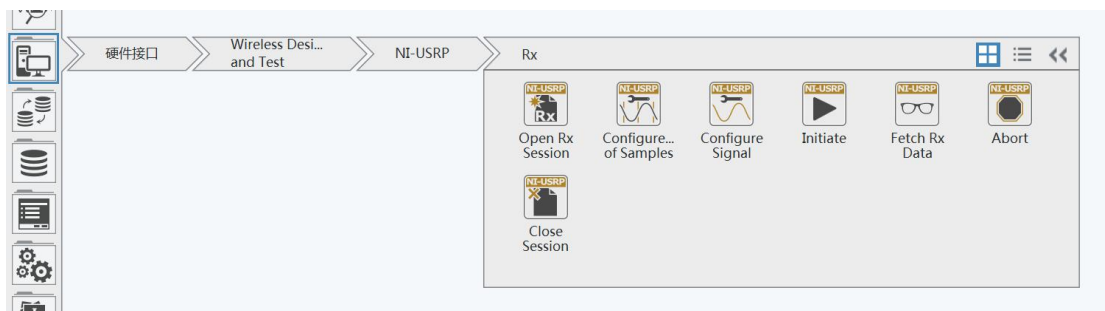


图 2-5 接收模块提取界面

#### (1) Open Rx Session:

对设备名称输入中指定的设备打开一个接收会话，并返回会话句柄输出。在模块的 device name 输入端需要输入设备名称，用于通知 Labview USRP 的 IP 地址或资源名称。

#### (2) Initiate:

开启接收

#### (3) Fetch Rx Data:

从指定的信道获取 16 位有符号整型复数数据。Number of samples 定义了从接收信道获取数据的采样数，data 是接收到的 16 位有符号整型复数数组数据。实部和虚部分别对应同相和正交数据。

#### (4) Abort:

停止接收

### 4. 双边带调制 (DSB)

假设  $\omega(t)$  是携带信息需要传输的信号，将  $\omega(t)$  与一个余弦载波信号相乘，得到：

$$s(t) = \omega(t) \cos(2\pi f_0 t)$$

其中， $f_0$  称为载波频率。根据欧拉特性：

$$\cos(2\pi f_0 t) = \frac{1}{2} (e^{j2\pi f_0 t} + e^{-j2\pi f_0 t})$$

计算信号  $s(t)$  的频谱，对它做傅里叶变化，可得：

$$\begin{aligned}
S(f) &= \int_{-\infty}^{\infty} \omega(t) \left[ \frac{1}{2} (e^{j2\pi f_0 t} + e^{-j2\pi f_0 t}) \right] e^{-j2\pi f t} dt \\
&= \frac{1}{2} \int_{-\infty}^{\infty} \omega(t) e^{-j2\pi(f-f_0)t} dt + \frac{1}{2} \int_{-\infty}^{\infty} \omega(t) e^{-j2\pi(f+f_0)t} dt \\
&= \frac{1}{2} W(f-f_0) + \frac{1}{2} W(f+f_0)
\end{aligned}$$

式中含有差频（下边带）、和频（上边带）两个分量。

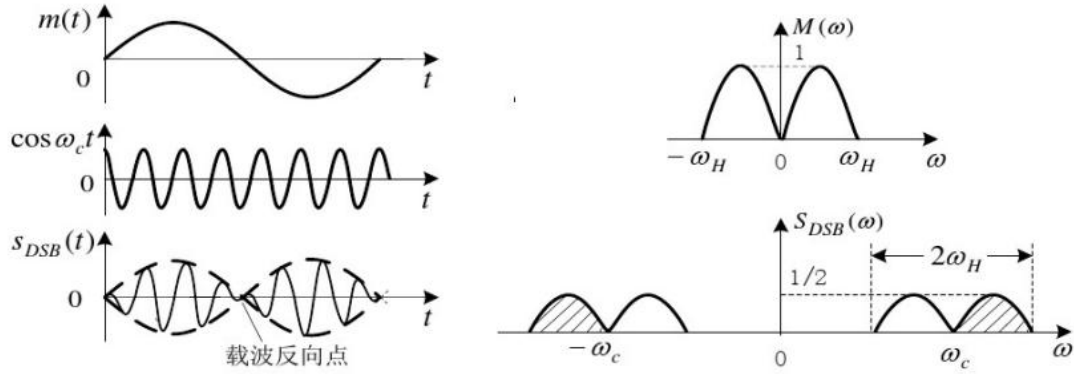


图 2-6 双边带调制的时域波形和频谱结构

## 5. 单边带调制 (LSB or USB)

双边带信号通过相应的窄带滤波电路后，可以保留其中一个边带，这样就可以实现上边带或下边带调制。

本实验中，采用三角函数变换的方法实现单音信号的上变频。将余弦波信号  $\cos \omega t$  作为同相数据，同频率同幅度的正弦波信号  $\sin \omega t$  作为正交数据，送给 Write Tx Data 模块，产生一个与载波信号频率  $\omega_0$  有一个偏置量的上边带信号。具体过程如下式：

$$\begin{aligned}
&\cos \omega t \times \cos \omega_0 t - \sin \omega t \times \sin \omega_0 t \\
&= \frac{1}{2} [\cos(\omega + \omega_0)t + \cos(\omega - \omega_0)t] + \frac{1}{2} [\cos(\omega + \omega_0)t - \cos(\omega - \omega_0)t] \\
&= \cos(\omega + \omega_0)t
\end{aligned}$$

## 四、实验内容与步骤

### 1. 构建基本的 USRP 传输器

基本的传输电路实验步骤如下：

- 1) 将 USRP 模块连接到电脑上
- 2) 运行 NI-USRP Configuration Utility 软件，获取设备 ID，如图 2-7 所示

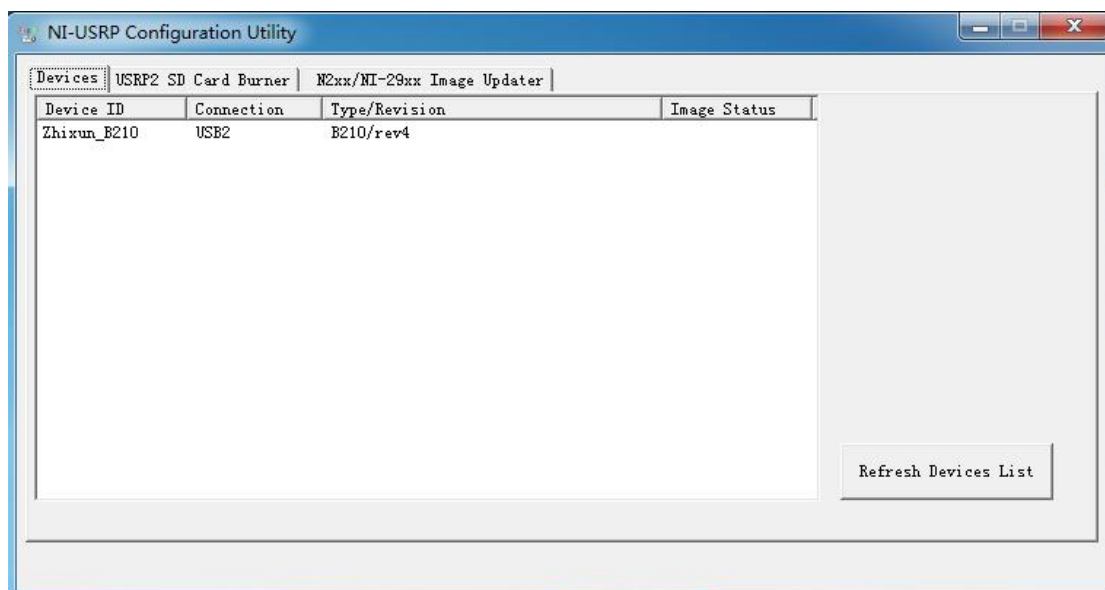


图 2-7 设备 ID 获取

- 3) 在 Labview 中新建一个项目，命名为 Lab2，存放于工作目录。再在项目中新建一个 VI，重命名为 carrier.gvi。
- 4) 在程序框图界面，放置以下模块：Open Tx Session, Configure Signal, Write Tx Data 和 Close Session。
- 5) 点击 Open Tx Session，在右侧的项目面板中，为 device names 创建常量，在程序框图界面给 Open Tx Session 选择设备 ID。

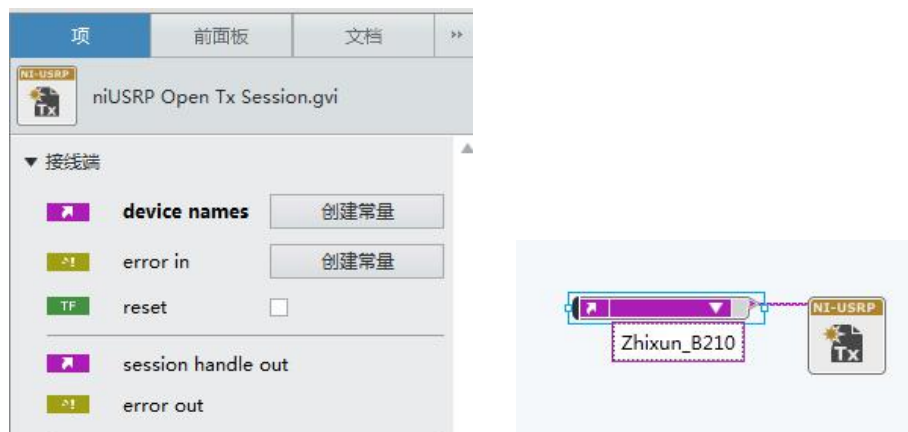


图 2-8 指定设备 ID

- 6) 放置一个 cluster properties 模块，选中点击右键，设置为全部写入。
- 7) 把各个模块的 session handle 端用线连起来
- 8) 左键点击 cluster properties 模块参数条 Start Trigger Type，选择 Configuration/Enabled Channels。右键点击 Enabled Channels 端口，创建一个常量输入，输入“0”，这将在 USRP 前面板上使能 RF0 信道。
- 9) 创建 3 个 DBL 类型的输入控制变量：IQ rate, carrier frequency 和 gain。创建一个字符串类型的输入控制变量 active antenna。创建三个 DBL 类型的输出显示变量：coerced IQ rate, coerced carrier frequency 和 coerced gain。将这些

变量与 Configure Signal 相应的端口连接。

10) 创建一个 while 循环，将 Write Tx Data 模块放在循环里面。

11) 连线模块的 Error out 端，创建一个 Error 显示输出变量，与 Close Session 模块的 Error out 端连线。

12) 创建一个 Stop 按钮（布尔类型变量），放置于 while 循环中。将 Stop 输入和 Write Tx Data 模块的错误输出作为 or 模块的输入，将 or 模块的输出作为循环中断控制。

完成的电路图如图 2-9 所示：

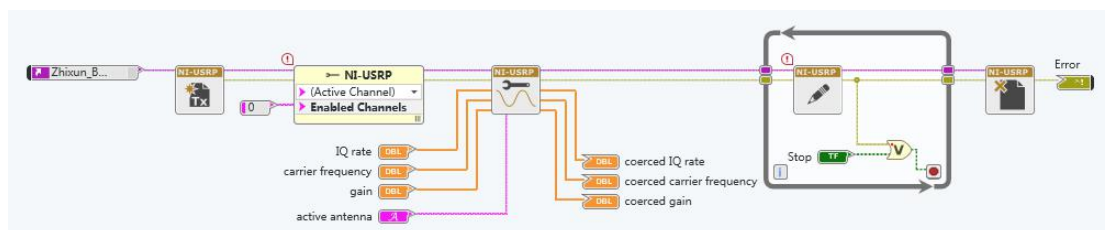


图 2-9 发射机基本电路

USRP 发送接收器是一个正交调制系统，它将同相信号与余弦波（频率值是 carrier frequency）混频，正交信号与正弦波混频。Write Tx Data 模块将输入的 IQ 基带采样信号与两路载波信号进行调制。模块的数据类型设置为 CDB（复数双精度）数组类型。

图 4 是产生正弦载波信号的电路图。使用初始化数组模块，给 Write Tx Data 模块输入一个常数数组作为基带采样信号。

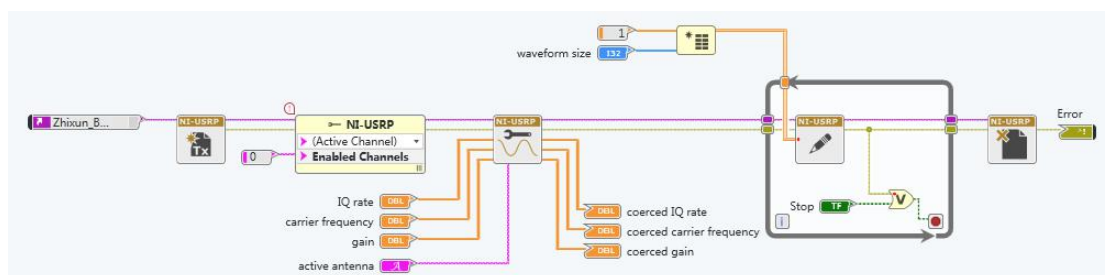


图 2-10 载波信号发送电路

13) 参数设置：

- ① IQ rate = 1M
- ② carrier frequency = 2G
- ③ gain = 0
- ④ active antenna = TX1
- ⑤ waveform size = 1000

14) 在前面板，将输入输出项放置在界面里，根据参数设置要求完成参数输入，运行电路。

15) 使用 SMA 电缆（和一个 30dB 的衰减器）将 USRP 设备的 TX1 输出端口连接到频谱仪，观察是否产生正确的信号频谱。

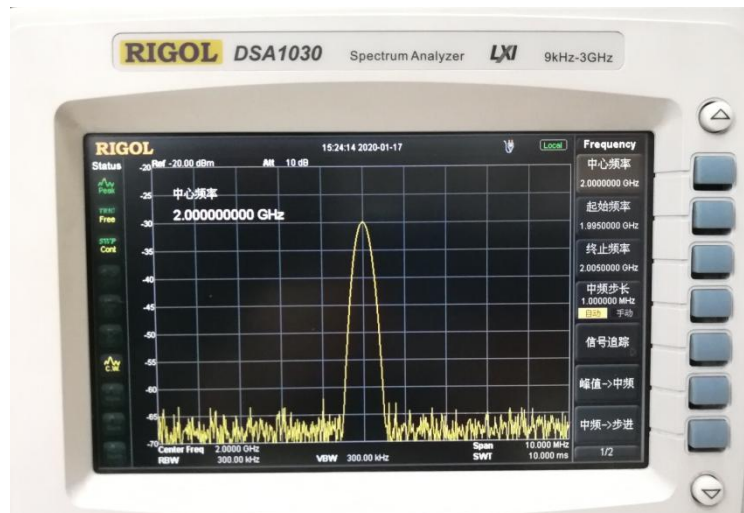


图 2-11 载波信号频谱仪测试

- 16) 改变载波信号频率值，取 1G、2.5G、3G、4G、5G、……，使用频谱仪确认频率是否正确。

## 2. 单音信号上边带传输

上边带电路实验步骤如下：

- 1) 新建一个 VI，命名为 upper-side\_IQ.gvi。复制 carrier.gvi 中的载波信号发送电路到 upper-side\_IQ.gvi 中，将初始化数组，常数输入以及它们与 Write Tx Data 模块的连线删除。
- 2) 创建两个 DBL 类型的参数输入：tone frequency 和 tone amplitude
- 3) 放置两个“波形发生器”模块（分析/信号处理/生成），默认状态下，模块是“Sine”波形和“Waveform”数据类型。
- 4) 将 tone frequency 控制输入连接到两个波形发生器模块的频率端口、tone amplitude 控制输入连接到两个波形发生器模块的幅值端口。将 waveform size 控制输入连接到采样端口，将采样率端口与 coerced IQ rate 输出相连。
- 5) 创建一个数值是“90”的常数输入，与波形发生器模块的相位输入连接。
- 6) 使用“波形属性”模块（数据类型/波形/模拟波形）从波形发生器模块中获取余弦和正弦波形的采样数据
- 7) 使用“实部虚部至复数转换”模块（数据类型/数值/复数）获取余弦和正弦采样并将它们转换成复数。

完成的上边带电路如图 2-12 所示：



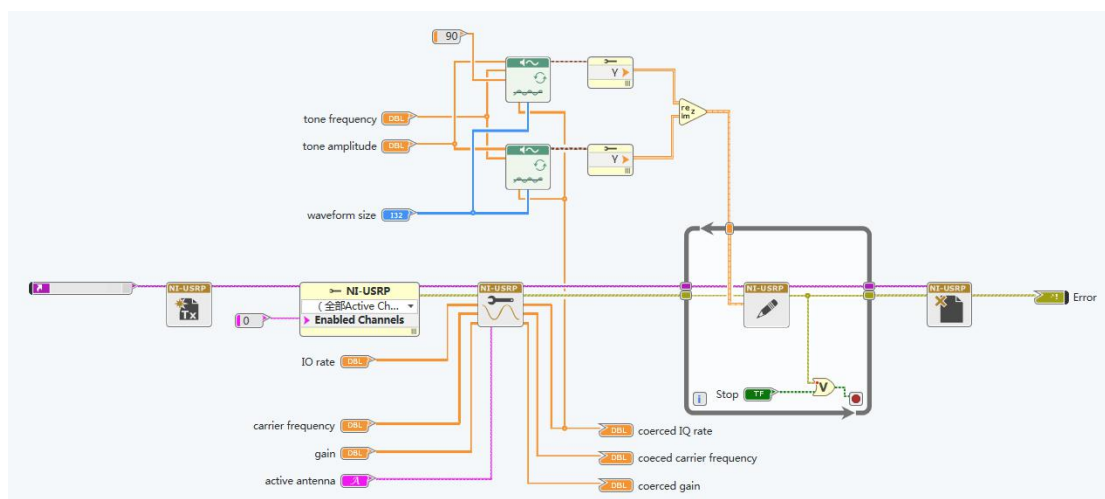


图 2-12 上边带调制发送

#### 8) 参数设置:

- ① IQ rate = 1M
  - ② carrier frequency = 2G
  - ③ gain = 0
  - ④ active antenna = TX1
  - ⑤ waveform size = 1000
  - ⑥ tone frequency = 1K
  - ⑦ Tone amplitude = 1
- 9) 在前面板, 将输入输出项放置在界面里, 根据参数设置要求完成参数输入, 运行电路。
  - 10) 使用 SMA 电缆(和一个 30dB 的衰减器)将 USRP 设备的 TX1 输出端口连接到频谱仪, 观察是否产生正确的上边带信号频谱。
  - 11) 改变 tone frequency 的值, 分别取: 10k、50k、100k, 用频谱分析仪观察上边带信号的频谱变化。

### 3. 问题 1

- 1) 在相同载波频率情况下, 比较载波发生电路和上变频电路的信号频谱。
- 2) 改变 waveform size 参数至 1005, 观察信号的频谱。
- 3) 新建一个 “lower-side\_IQ.gvi”, 完成下变频电路, 观察信号的频谱
- 4) 新建一个 “double-side\_IQ.gvi”, 完成双边带电路, 观察信号的频谱

### 4. 构建 USRP 接收器

USRP 接收器实验步骤如下:

- 1) 新建一个 VI, 并命名为 Rx.gvi。
- 2) 在程序框图界面, 放置以下模块: Open Rx Session, Configure Signal, Fetch Rx Data, Abort 和 Close Session。同发送电路一样, 为参数: IQ rate, carrier

frequency, gain 和 active antenna 创建控制输入和显示输入。模块的 enabled channel 设置为常数 1, 是能 USRP 设备的 RF1 信道。将 Fetch Rx Data 模块放入 while 循环, 点击 Fetch Rx Data 模块, 在函数配置栏, 将数据类型更改为 CDB WDT。完成的电路图如图 2-13 所示。

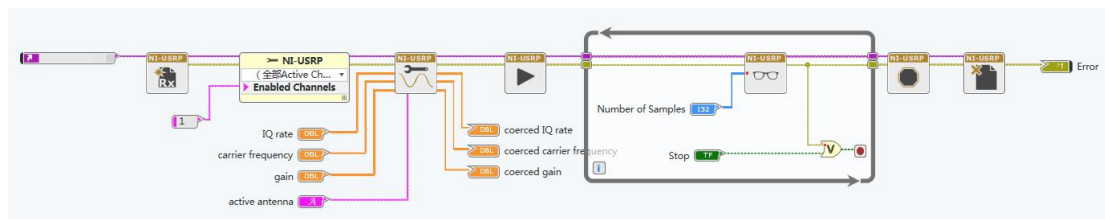


图 2-13 接收机基本电路

- 3) 在电路图中放置“FFT 功率谱和 PSD”模块（分析/信号处理/测量），点击模块，在右侧菜单栏将模块的函数配置设置为“功率谱”和“连续”。在接线端参数设置中，勾选“显示为 dB”。将信号端与 Fetch Rx Data 的数据端相连。为了能够显示频谱，在前面板，从左侧“图形和图表”菜单中选择“图形”显示输出，回到程序框图界面，将“图形”显示输出与功率谱端口相连。
- 4) 放置一个“波形属性”模块（数据类型/波形/模拟波形），将 Fetch Rx Data 模块的数据作为它的输入，使用“复数至实部虚部转换”模块，波形属性模块输出中提取 I 路和 Q 路采样值。使用“创建波形”模块（数据类型/波形/模拟波形）和波形显示输出（在前面板创建），显示 I、Q 两路信号的时域波形。完整的接收电路如图 2-14 所示：

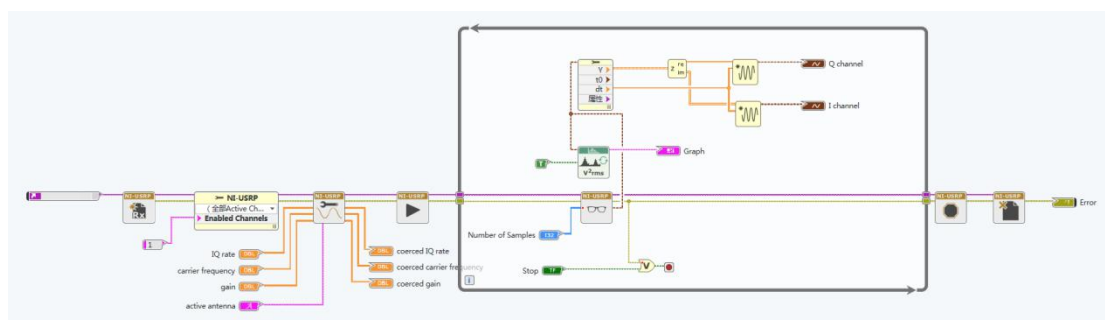


图 2-14 完整的接收电路

- 5) 参数设置：
  - ① IQ rate = 1M
  - ② carrier frequency = 2G
  - ③ gain = 0
  - ④ active antenna = RX2
  - ⑤ number of samples = 100
- 6) 在前面板，将输入输出项放置在界面里，根据参数设置要求完成参数输入。
- 7) 使用电缆（和一个 30dB 的衰减器），将 RF0 的 TX1 端口和 RF1 的 RX2 端口连接起来。

## 5. 问题 2

- 1) 先运行 “upper-side\_IQ.gvi”，然后再运行 “Rx.gvi”，观察接收信号的波形和频谱。
- 2) 分别运行“lower-side\_IQ.gvi”和“double-side\_IQ.gvi”的情况下,运行“Rx.gvi”，观察接收信号的波形和频谱。
- 3) 在 “lower-side\_IQ.gvi” 中，将正弦波信号改为方波信号，观察接收信号频谱的变化。

## 五、实验报告要求

- 4) 给出所设计的电路，记录实验数据并加以分析
- 5) 回答所有问题
- 6) 报告在下个实验开始前以书面或电子版形式递交

## 实验三：脉冲成形和匹配滤波

### 一、实验目的

- (1) 构建基本的数字通信发送、接收系统
- (2) 了解脉冲成形的基本概念
- (3) 了解匹配滤波的基本概念
- (4) 了解脉冲对齐的基本概念

### 二、实验设备

- |                      |     |
|----------------------|-----|
| (5) USRP 设备          | 1 台 |
| (6) 安装 LabVIEW 环境的电脑 | 1 台 |

### 三、实验概要

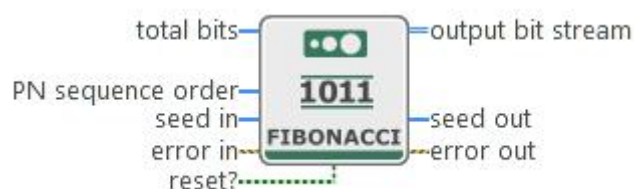
在数字通信系统中，数字信息必须要转换成模拟信号才能被发送出去。这种变换由脉冲成形滤波器完成，它将每一个数据符号转换成一个合适的模拟脉冲信号。由于脉冲信号的频谱决定了整个发送系统的频谱情况，因此脉冲成形滤波器的设计至关重要。为了限制频谱宽度，必须使用慢转换来平滑脉冲，这会导致脉冲信号超出一个符号时间，引入码间干扰。因此，在带宽和码间干扰之间做出权衡。

匹配滤波器负责从接收到的脉冲中捕捉数据符号。匹配滤波器的目标是通过最大化信噪比和最小化码间干扰，减小噪声的影响。

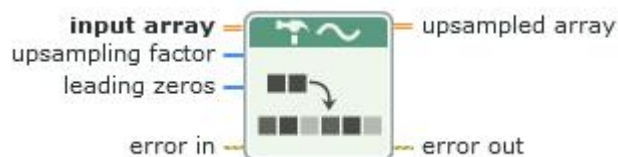
主要模块介绍：

以下是实验中将会用到的主要模块：

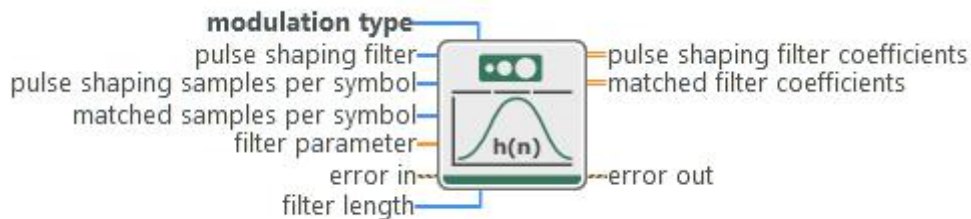
- (1) MT Generate Bits (Fibonacci, PN Order) (分析/Modulation/Digital)：产生 Fibonacci 伪随机序列，使用这个模块产生需要传输的数据。



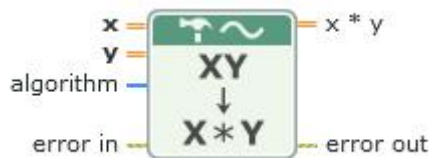
- (2) 升采样 Upsample (分析/信号处理/调理)：根据给定的升采样因子，在一个序列中插入零。



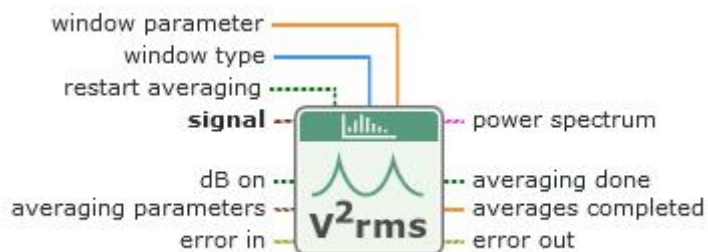
- (3) MT Generate Filter Coefficients (分析/Modulation/Digital/Utilities)：计算滤波系数，用于在数字调制和解调中使用的脉冲成形和匹配滤波器。



(4) 卷积（分析/信号处理/运算）：计算两个序列的卷积。



(5) FFT 功率谱和 PSD（分析/信号处理/测量）：计算时域信号的平均功率谱。



(6) MT Format Eye Diagram（分析/Modulation/Digital/Visualization）：对于一个给定的实数值波形，先将其分成片段，再在波形图上将这些片段以点的形式显示出来。片段的长度是基于符号率和眼图长度参数确定的。



#### 四、实验内容与步骤

##### 1. 脉冲成形

在实验 2 发射机基本电路图 2-9 的基础上，构建一个发送电路。电路中使用脉冲成形和升采样来创建发送波形，使用二进制相移键控（BPSK）调制，BPSK 是最简单的相移键控（PSK）。载波信号的相位携带数据信息，在 BPSK 中，数据“1”对应 0 度，数据“0”对应 180 度。实验步骤如下：

- 1) 创建两个整型输入控件：Message length 和 Symbol rate。升采样因子 upsampling factor 的值由 IQ rate 和 Symbol rate 决定。对数据符号的升采样要在脉冲成形之前完成。
- 2) 产生比特数据：在程序框图中添加 MT Generate Bits 模块来产生一个 Fibonacci 伪随机序列。这个序列由二进制 0 和 1 组成。模块参数“PN sequence order”的默认值是 7。使用默认值。将“Message length”控制输入连接到模块的“total bits”端。
- 3) 比特数据编码成 BPSK 符号：使用乘法和减 1 模块将二进制 0 和 1 映射成符号 -1 和 1（BPSK 编码）。

- 4) 创建一个 DBL 类型的显示变量，并将它命名为 **Upsampling Factor**（升采样因子）。使用合适的数学运算模块计算升采样因子（升采样因子定义为采样率/符号率）。
- 5) 升采样：在程序框图中添加 **Upsample** 模块，使用前面步骤中的 **BPSK** 编码符号和升采样因子，产生升采样符号。
- 6) 脉冲成形：添加 **MT Generate Filter Coefficients** 模块。右键点击模块的“pulse shaping filter”端，选择创建输入控件，创建“pulse shaping filter”端口。同样的，右键点击模块的“modulation type”端并选择创建常量，将它设置为“PSK”。将 **Upsampling Factor** 输入控件与模块的“pulse shaping samples per symbol”端相连。将“pulse shaping filter coefficients”作为输出。“filter length”取默认值 8，“filter parameter”（升余弦或均方根升余弦的滚降因子）取默认值 0.5。滤波器长度 **filter length** 是指在脉冲成形滤波器中符号所需的长度。
- 7) 卷积：使用卷积模块对升采样符号和脉冲成形滤波器系数作卷积运算。
- 8) 创建子 VI 函数 **norm1D**：将符号绝对值的最大值对符号进行归一化，使得符号值在-1 和 1 之间。在项目中创建一个 VI，命名为 **norm1D.gvi**，使用绝对值模块（数学/数值）和数组最大值与最小值模块（数据类型/数组），如下图 3-1 所示。点击“图标”按钮，给新建的子 VI 添加输入、输出端口。

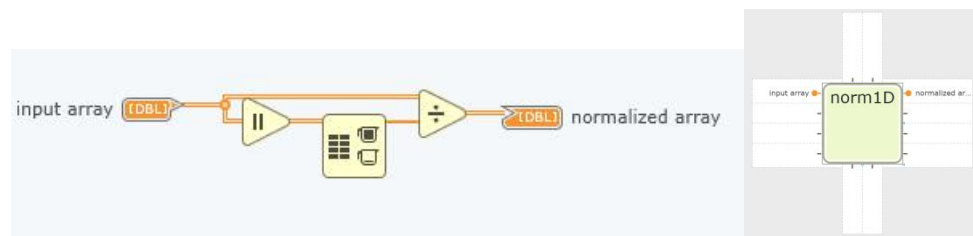


图 3-1 归一化电路

- 9) 回到发送电路程序框图界面，将“**norm1D**”子 VI 放入电路中，对卷积模块的输出进行归一化，再将归一化采样符号与 **niUSRP Write Tx Data** 的数据端相连。
- 10) 测量发送信号的频谱和眼图：使用创建波形模块从发送数据中构建时域波形。使用“倒数”模块从 **IQ rate** 中计算采样周期（dt），并将计算得到的采样周期与创建波形模块相连。添加“**FFT 功率谱和 PSD**”模块，并将它的函数配置为“功率谱”和“连续”。勾选“显示为 dB”。右键点击模块的功率谱端，创建一个显示控件。添加 **MT Format Eye Diagram** 模块（分析/Modulation/Digital/Visualization），将函数配置成“WDT”以接收波形数据。将“eye length”设置成 2，将 **symbol rate** 输入控件与模块相应端相连。右键点击模块的“eye diagram”端创建一个显示控件。
- 11) 保存此发送电路 VI 为“**BPSKTx.gvi**”。
- 12) 参数设置：
  - ①  $\text{IQ rate} = 400\text{k}$
  - ②  $\text{carrier frequency} = 1\text{G}$
  - ③  $\text{gain} = 0$
  - ④  $\text{active antenna} = \text{TX1}$
  - ⑤  $\text{Message length} = 1000$
  - ⑥  $\text{Symbol rate} = 10\text{k}$

## 2. 问题 1

- 1) 如果 **PN** 序列的阶数是 7，它的周期是多少？
- 2) 运行程序的时候，选择两种不同的脉冲成形滤波器：**None** 和 **Root Raised Cosine**，保存



对应的频谱图和眼图。比较不同滤波器情况下的带宽、旁瓣和眼图打开情况。

- 3) 选择 Root Raised Cosine 滤波器，将 Symbol rate 改为 100k，跟 Symbol rate 为 10k 时的频谱和眼图作比较。

### 3. 匹配滤波

在实验 2 图 2-13 的基础上，构建一个完成匹配滤波功能的接收机电路。

实验步骤如下：

- 1) 创建两个整型输入控件：信息长度 Message length 和 Symbol rate。降采样因子由 IQ rate 和 Symbol rate 决定。在发送端，在脉冲成形前对符号作了升采样处理。
- 2) （降采样因子）接收机对接收到的采样信号进行降采样处理，计算降采样因子的方法类似于升采样，这两个数值可以不一样。
- 3) 接收机捕获一帧数据，需要计算捕获到的采样数（提示：需要用到降采样因子和信息长度），并且将采样数与 niUSRP Fetch Rx Data 模块的“number of samples”端相连。移除 while 循环，只作一次数据接收，接收机接收到一帧数据后停止接收。niUSRP Fetch Rx Data 模块的函数配置为 CDB WDT 类型。
- 4) （匹配滤波）添加 MT Generate Filter Coefficients 模块，在模块“pulse shaping filter”端口点击右键创建控制终端，创建“modulation type”常量端口，将它的值设置为“PSK”。将前面步骤中计算的到的降采样因子的值与“matched samples per symbol”相连。将“matched filter coefficients”端作为输出。
- 5) （卷积）使用卷积运算模块，对接收到的采样值和匹配滤波器的输出进行卷积运算。NiUSRP Fetch Rx Data 模块输出的数据是 CDB WDT 类型的，卷积模块处理的数据类型是 CDB 数组类型，所以需要波形属性模块从接收数据中获取 CDB 数组类型数据。添加模型属性模块，并将其行为设置成“全部为读取”。部分电路图如下图 3-2 所示。

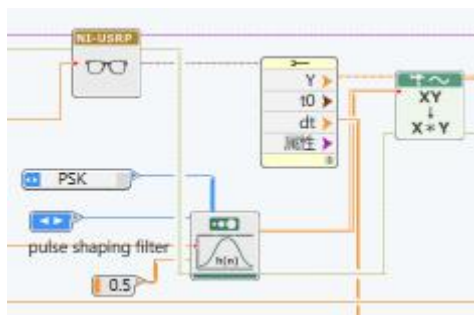


图 3-2 部分发送电路

- 6) （下采样&BPSK 解调）添加降采样模块，将卷积输出连入模块的信号输入端，将前面计算得到的降采样因子数值连入模块的降采样因子端口。使用复数至实部虚部转换模块获取符号的实部值。完成 BPSK 解调（使用大于等于？、布尔值至整数转换模块）：如果符号值小于 0，则比特数据为 0；如果符号值大于等于 0，则比特数据为 1。恢复的比特数据使用数组显示，将它命名为“unaligned bits”。部分电路图如下图 3-3 所示。

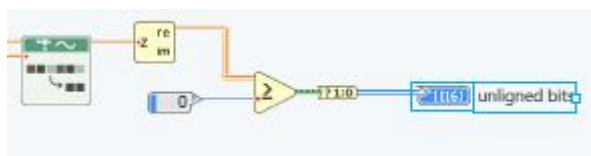


图 3-3 下采样和解调电路

- 7) （眼图）取卷积值得实部数值，使用创建波形和 MT Format Eye Diagram 模块显示眼图。

- 8) 将此 VI 保存为 “BPSKRx.gvi”。
- 9) 参数设置：
  - ① IQ rate = 1M
  - ② carrier frequency = 1G
  - ③ gain = 0
  - ④ active antenna = RX2
  - ⑤ Message length = 1000
  - ⑥ Symbol rate = 10k
- 10) 先运行发送电路代码，再运行接收电路代码，比较匹配滤波前后的波形，测量匹配滤波后的信号波形眼图。

#### 4. 问题 2

- 1) 在接收和发送电路中，将均方根升余弦的滤波器参数改为 0.2 和 1，随着滤波器参数值的增加，码间干扰的数量是增加还是减少了？接收信号的带宽是增加还是减少了？将接收采样值接入功率谱模块测量频谱。
- 2) 多次运行接收机电路代码，观察眼图。每次眼图都对齐了吗？如果没有对齐，请解释原因。如果脉冲没有对齐，会出现什么问题？

#### 5. 脉冲对齐

在接收机电路中，增加脉冲对齐功能。

实验步骤如下：

- 1) 复制 “BPSKTx.gvi” 文件，重命名为 “BPSKTx\_shortPN.gvi”。
- 2) 将 MT Generate Bits 的函数配置改为 “User Defined”。点击右键，为模块的 “user base bit pattern” 创建一个数组常量，输入位序列 “1011100”。在模块 bit 流输出端创建一个显示控件，观察输出 bit 流。局部电路如图 3-4 所示。

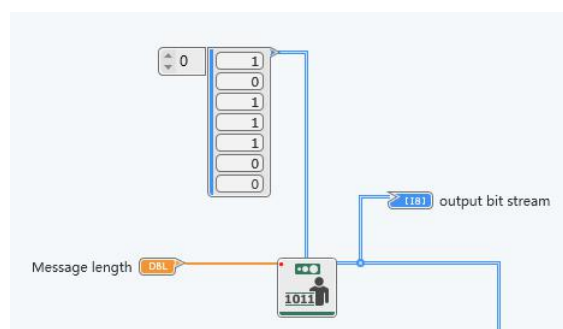


图 3-4 局部电路图

- 3) 完成下面图 3-5 所示的子模块电路，并将其命名为 “PulseAlign.gvi”。点击 “图标” 按钮，给新建的模块添加输入输出端口。

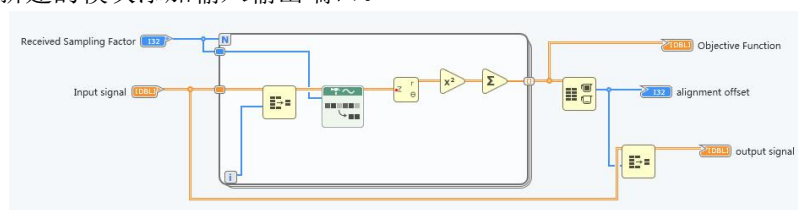


图 3-5 脉冲对齐电路

- 4) 在接收电路中，添加脉冲对齐模块，对匹配滤波后的采样信号进行脉冲对齐，并显示脉

冲对齐后的信号波形眼图（命名为“aligned eye”）。

5) 对对齐后的采样信号进行解调，恢复出位序列（命名为“aligned bits”）。

### 6. 问题 3

1) 解释脉冲对齐电路的工作原理。

2) 对不作脉冲对齐和作脉冲对齐的接收信号眼图作比较。

3) 比较不对齐和对齐后接收到的位序列，那种情况下能检测到传输的数据“1011100”？

### 五、实验报告要求

7) 给出所设计的电路，记录实验数据并加以分析

8) 回答所有问题

9) 报告在下一个实验开始前以书面或电子版形式递交

# 实验四：帧同步和非相干调制

## 一、实验目的

- (1) 了解帧同步的概念和实现方法
- (2) 了解差分二进制相移键控（DBPSK）的编码与解码实现方法

## 二、实验设备

- (1) USRP 设备 1 台
- (2) 安装 LabVIEW 环境的电脑 1 台

## 三、实验概要

判断发送、接收系统是否工作正常，需要对发送和接收到的数据比特作比较，以确认它们的位模式是否一样。在实验三中，接收机能正确接收到短的位序列（例如：1011100）。实际情况中对于长序列，接收机有必要识别出发送序列的起始位置。因此，发送机在每一帧数据的开头，插入一段训练序列（头或前导码），以此来标识有效数据的开始。接收机对接收到的每一帧数据搜索这个训练序列，当接收到的数据和这段固定模式序列的互相关性高的时候，就表明实现了帧同步。

数字调制和解调类型分为两种：相干和不相干。相干调制/解调系统需要载波相位信息监测和确认发送的数据。而非相干系统不需要这些相位信息。在本实验中，将介绍一种非相干调制：差分二进制相移键控（DBPSK）。

## 四、实验内容与步骤

### 1. 帧同步

在这一部分实验内容中，需要完成以下子 VI：

- 1) AddTraining.gvi：在一帧数据中插入训练序列
- 2) FrameSync.gvi：检测训练序列并且实现帧同步
- 3) Sim\_framesync.gvi：对以上子 VI 进行仿真

#### 1.1 添加训练序列

设计“AddTraining.gvi”子 VI，在发送电路中，将训练序列符号插入一帧中。表 4-1 是输入输出端口情况。

表 4-1 输入输出端口情况

	端口名称	类型	描述
输入	Original frame	双精度数组	发送符号中的原始数据帧
	Training sequence	双精度数组	
输出	Frame with training sequence	双精度数组	在起始位置有训练序列的数据帧

使用“数组插入模块”，在在原始帧的起始位置插入训练序列（“数组插入”模块的“索引”端口值应该设置为 0）。

#### 1.2 帧检测和同步

设计“FrameSync.gvi”子 VI，为了对齐和同步帧，将训练序列符号放置在发送机端的发送数据帧起始位置。表 4-2 是输入输出端口情况。

表 4-2 输入输出端口情况

	端口名称	类型	描述
输入	Unsynced frame	双精度数组	同步前接收到的数据帧
	Training sequence	双精度数组	
	Threshold	双精度	检测出的最小峰值
输出	Synced frame	双精度数组	同步数据帧
	Cross-correlation result	双精度数组	接收到的数据帧和训练序列之间的互相关结果
	Peak index	整数数组	互相关结果峰值索引

- 1) 使用“信号相关性”模块（分析/信号处理/运算），在函数设置中选择“互相关”选项，计算接收到的数据帧和训练序列的互相关。
- 2) 使用“阈值检测”模块检测互相关值的峰值和它们的索引位置（提示：“峰值检测模块”检测超出设定阈值的输入序列的峰值，它在“位置”端口输出峰值的索引值，在“计数”端口输出峰值的数量）。
- 3) 从接收到的数据帧中移除训练序列，得到同步数据帧（提示：使用“索引数组”模块获取第一个相关值峰值的索引，使用“数组子集”从“Unsynced frame”中得到“Synced frame”）。

### 1.3 仿真

创建一个子 VI 并将它命名为“sim\_framesync.gvi”，用来确认“AddTraining”和“FrameSync”是否正确。

- 1) 使用“MT Generate Bits”模块生成一个数据比例序列。例如：配置模块为“Fibonacci”PN 序列，total bits 为 50，PN sequence order 为 5。
- 2) 将数据比特映射成符号：1 映射成符号“1”，0 映射成符号“-1”。创建一个显示控件以监测产生的符号序列，它将作为不包含训练序列的原始数据帧。
- 3) 用 13 位 Barker 码创建一个常量数组，作为训练符号序列。13 位 Barker 码为：  
+1+1+1+1+1-1-1+1+1-1+1+1。
- 4) 使用“AddTraining”模块将训练序列插入到原始数据帧的起始位置，并产生加入了训练序列的数据帧。
- 5) 使用“FrameSync”模块确定上一步骤中产生的数据帧中训练序列的位置。需要观察相关值，并基于互相关值选择一个合适的阈值以找出需要的峰值。

### 1.4 问题 1

- 1) 帧同步需要的互相关峰值在哪里？峰值与原始数据帧的起始位置有什么关系？你的“FrameSync.gvi”子 VI 是否产生了同步数据帧？
- 2) 一个好的训练序列应该要能给出一个明显的互相关峰值，以便于检测。因此可能需要增加训练序列的长度。Barker 码的最大长度是 13，可以将两个或更多的 Barker 码连接起来。使用 13 位序列创建一个 26 位训练序列（提示：使用“创建数组”模块，并且勾选其行为特性“连接输入”来连接两个 Barker 码）。阈值将作何改变？此时，互相关的峰值在哪里？原始数据帧的起始位置在哪里？
- 3) 长训练序列的优点和缺点是什么？

### 2. 差分二进制相移键控（DBPSK）

## 2.1 发送电路

在实验 3 “TxBPSK.gvi” 的基础上完成此部分实验，新建一个项目，将此文件拷贝到新项目中，重命名为 “TxDBPDK.gvi”。在这部分实验中，将要构建一个传输电路，完成差分二进制相移键控调制（DBPSK）。同 BPSK 一样，数据信息由载波信号的相位携带。与 BPSK 不同之处在于：DBPSK 传输的符号不仅取决于现在的位信息，也取决于前一个位信息。

在前面板中设置以下参数：

- ① IQ rate = 400k
- ② carrier frequency = 1G
- ③ gain = 0
- ④ active antenna = TX1
- ⑤ Message length = 1000
- ⑥ Symbol rate = 10k
- ⑦ Enabled Channels = 0

1) 参考表 4-3，对添加了训练序列的数据帧进行差分编码。

表 4-3 差分编码参考

输入符号	-1	-1	1	1
前一符号	-1	1	-1	1
当前符号	1	-1	-1	1

2) 重复利用发送电路其余部分电路：脉冲成形滤波器、卷积等。

3) 同时在发送和接收电路中对 NI-USRP 属性模块添加 “LO Frequency” 参数的配置，如图 4-1 所示。

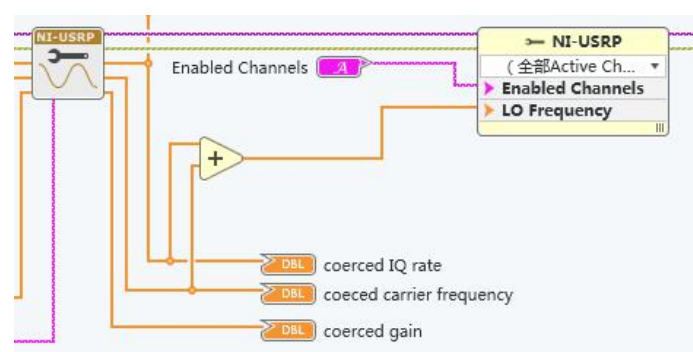


图 4-1 参数配置

## 2.2 接收电路

在实验 3 “RxBPSK.gvi” 的基础上完成这部分实验，将文件重命名为 “RxDBPSK.gvi”。

在前面板中设置以下参数：



- ① IQ rate = 400k
  - ② carrier frequency = 1G
  - ③ gain = 0
  - ④ active antenna = RX2
  - ⑤ Message length = 1000
  - ⑥ Symbol rate = 10k
- 1) 计算捕捉两帧数据所需要的采样数，并将它作为“niUSRP Fetch Rx Data”模块 number of samples 参数端的输入。
  - 2) 使用原有电路中的匹配滤波器、卷积、脉冲成形和降采样模块。
  - 3) 需要重新构建一个归一化模块，命名为“norm1D\_median.gvi”。与实验 3 中使用最大值进行归一化不同，这里用中值对向量进行归一化。创建一个子 VI，如图 4-2 所示，使用它对接收到的符号进行归一化。

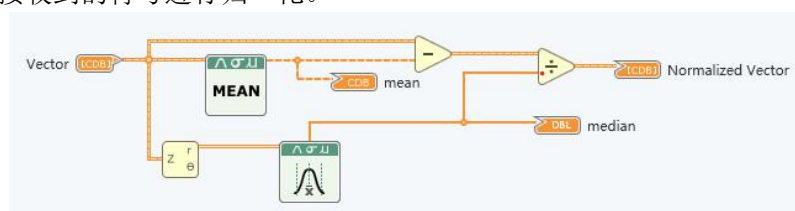


图 4-2 归一化电路

- 4) （差分译码）完成图 4-3 所示电路，对归一化符号进行译码。

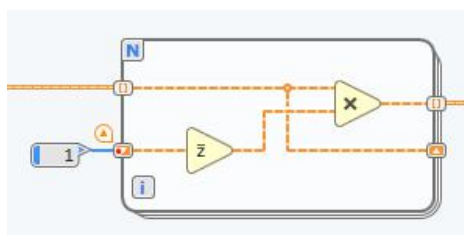


图 4-3 差分译码电路

- 5) 获取译码符号的实部数值，使用“FrameSync”模块以及合适的训练序列对数据帧进行同步。阈值建议设置为 18。
- 6) （符号到位比特）采用与 BPSK 同样的方法，将符号数据映射为比特数据。
- 7) （误比特率）通过与发送的比特数据序列进行比较，来计算误比特率确认译码比特数据的正确性。创建一个和发送电路有相同配置和参数的 PN 序列数据（即：Fibonacci, PN sequence order, total bits, seed in 等）。使用一个 for 循环将译码得到的比特数据与参考数据作比较，计算错误的个数（提示：使用“不等于？”、“布尔值至整数转换”、“加”模块和移位寄存器）。将错误数除以信息长度（Message length）得到误比特率。
- 8) （星座图）在前面板放置一个图形显示控件，将 X 轴名称重命名为 I，Y 轴重命名为 Q。曲线类型选择为“点”。设置自动调整标尺为“无”、标尺最大值设置为 1.5、最小值设置为 -1.5。在图形显示控件中显示译码符号数据。

## 2.3 问题 2

- 1) 给出误比特率和 DBPSK 的星座图。运行接收电路一段时间，观察实验结果是否稳定。
- 2) 给出 BPSK 的误比特率和星座图，与 DBPSK 的结果作比较，并解释原因。
- 3) 理解并解释实验步骤 4 中差分译码的工作原理。

- 4) 对采样符号乘以一个常量（比如 0.01），以减小发送信号的功率，给出新的星座图。

## 五、实验报告要求

- 10) 给出所设计的电路，记录实验数据并加以分析
- 11) 回答所有问题
- 12) 报告在下个实验开始前以书面或电子版形式递交

## 实验五：ASK 与 FSK 调制

### 一、实验目的

- (1) 掌握 ASK 调制的原理和实现方法
- (2) 掌握 FSK 调制的原理和实验方法

### 二、实验设备

- |                      |     |
|----------------------|-----|
| (1) USRP 设备          | 1 台 |
| (2) 安装 LabVIEW 环境的电脑 | 1 台 |
| (3) 频谱分析仪            | 1 台 |

### 三、实验概要

本实验涉及两种非相干调制：幅移键控（ASK）和频移键控（FSK）。在 ASK 调制中，信息通过载波信号的幅度传递。载波信号的幅度跟随信号变化，频率和相位保持不变。假设载波信号用  $\cos 2\pi f_c t$  表示，对于二进制 ASK 调制，比特数据“1”用  $A_1 \cos 2\pi f_c t$  表示；比特数据“0”用  $A_0 \cos 2\pi f_c t$  表示。如图 5-1 所示。

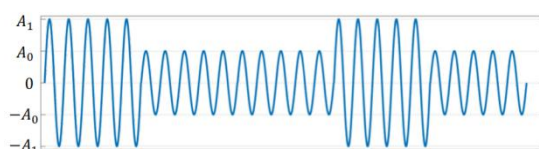


图 5-1 ASK 调制

ASK 调制发送系统框图如图 5-2 所示。数据比特信息先转换成幅度信息，再与载波信号相乘，经功率放大后发送。



图 5-2 ASK 调制发送系统框图

ASK 调制接收系统框图如图 5-3 所示。接收到的信号经过低噪声放大器放大，再通过平方律检波器检波，通过滤波器将不需要的频率分量滤除，得到包含发送信息的直流分量，最后经过比较器，得到比特数据。



图 5-3 ASK 调制接收系统框图

On-Off 键控是最简单的 ASK 调制形式。当  $A_0=0$  时，就是 On-Off 键控。简单的 On-Off 发送/接收系统如图 5-4 所示。

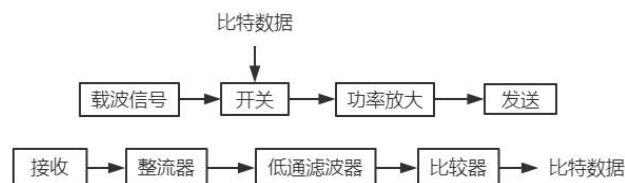


图 5-4 On-Off 发送/接收系统框图

对应的数据变换如图 5-5 所示：

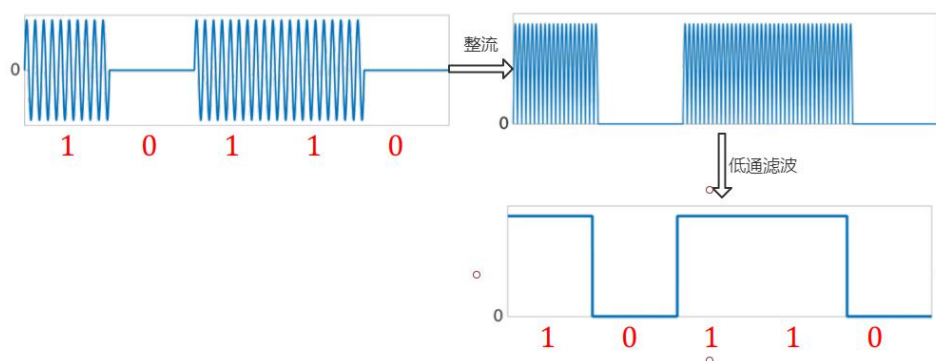


图 5-5 On-Off 键控数据变换

在 FSK 调制中，信息通过载波信号的离散频率变化传送。对于二进制 FSK 调制，比特数据“1”和“0”由两个频率不同的载波信号表示。由于调制信号在两个不同频率间瞬时切换，标准的 FSK 信号相位是不连续的。在实验中，FSK 具有连续的相位，以避免由于不连续相位而产生的宽带频率成份。FSK 调制信号如图 5-6 所示。

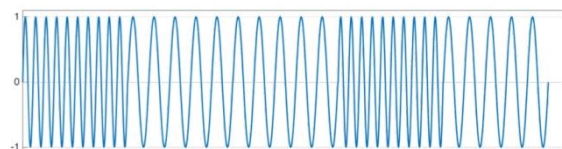


图 5-6 FSK 调制

#### 四、实验内容与步骤

##### 1. ASK: On-Off 键控 (OOK)

###### 1.1 发送电路

在“TxDBPSK.gvi”的基础构建一个实现 OOK 调制的发送电路。将文件重命名为“TxOOK.gvi”。数据通过载波信号的幅度传输，幅度 1 代表数据“1”，幅度 0 代表数据“0”。OOK 调制信号与幅度调制有相似的表达式：

$$y(t) = \frac{1}{2}(1 + m(t)) \cdot A \cdot \cos(2\pi f_c t) \quad 5-1$$

$m(t)$  是脉冲成形符号（大小在 -1 和 1 之间）， $A$  是载波信号的幅度， $f_c$  是载波频率。

在前面板中设置以下参数：

- ① IQ rate = 400k
  - ② carrier frequency = 900.1MHz
  - ③ gain = 0
  - ④ active antenna = TX1
  - ⑤ Message length = 1000
  - ⑥ Symbol rate = 10k
- 1) 继续使用原发送电路中的一些功能模块（比如：位比特数据到符号转换、添加训练序列、脉冲成形、卷积等）。将脉冲成形的调制类型改为 ASK，去除差分编码功能电路。
  - 2) 在图 5-7 中空白区域完成 OOK 调制（提示：参考式 5-1，改变载波信号的幅度）。

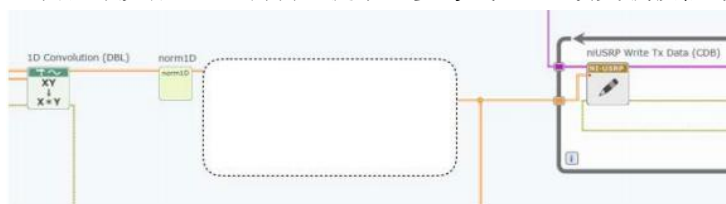


图 5-7

## 1.2 接收电路

在“RxDBPSK.gvi”的基础上构建一个接收电路，完成 OOK 调制信号的接收。将文件重命名为“RxOOK.gvi”。

在前面板中设置以下参数：

- ① IQ rate = 400k
  - ② carrier frequency = 900MHz
  - ③ gain = 0
  - ④ active antenna = RX2
  - ⑤ Message length = 1000
  - ⑥ Symbol rate = 10k
- 1) 继续使用原接收电路中的一些功能模块（比如：匹配滤波器、脉冲对齐、帧同步、符号到比特数据转换、误比特率、星座图等）。将脉冲成形滤波器的调制类型改为“ASK”，去除差分译码功能电路。
  - 2) 进行 OOK 解调，使用带通滤波器滤除不需要的频率分量，需要的接收信号频率在 100KHz。在下图 5-8 空白区域完成一个整流电路（提示：将数值为负的数据取反）。

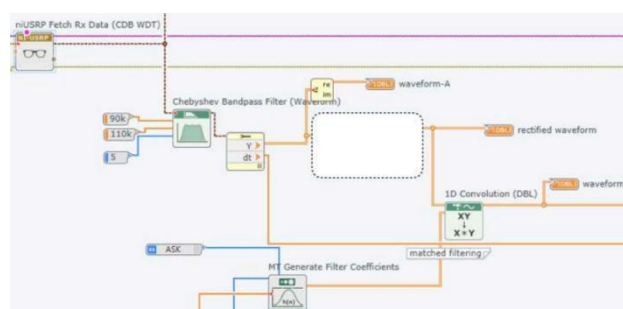


图 5-8

- 3) 给帧同步模块指定合适的阈值
- 4) 确定是否能将数据比特正确的译码出来，并得到正确的星座图和误比特率。

## 1.3 问题 1

- 1) 图 5-8 中，在“waveform-A”、“rectified waveform”和“waveform-B”输出控件端能得到什么波形？将波形画出来并解释原因。
- 2) 假如要实现普通的幅度移频键控（ASK），电路图需要做出怎样的更改？

## 2. FSK

### 2.1 发送电路

在“TxASK.gvi”的基础上，构建一个完成 FSK 调制的发送电路。将文件重命名为“TxFSK.gvi”。数据通过载波信号的频率传递，连续相位 FSK 调制信号与频率调制有相似的表达式：

$$y(t) = A \cdot \cos\left(2\pi f_c t + 2\pi \Delta f \int_0^t m(\tau) d\tau\right) \quad 5-2$$

$\Delta f$  是峰值频率偏移， $m(t)$  是数值介于-1 到 1 之间的脉冲成形符号， $A$  是载波信号的幅度， $f_c$  是载波频率。

在前面板中设置以下参数：

- ① IQ rate = 400k
  - ② carrier frequency = 1GHz
  - ③ gain = 0
  - ④ active antenna = TX1
  - ⑤ Message length = 1000
  - ⑥ Symbol rate = 10k
  - ⑦ Enabled Channels = 0
- 1) 继续使用基本发送电路中的一些功能模块（比如：位比特数据到符号转换、添加训练序列、脉冲成形、卷积等），将脉冲成形的调制类型改为 FSK。
  - 2) （频率调制）完成下图 5-9 中空白区域的电路，实现 FSK 调制（提示：参考式 5-2，其中  $m(t)$  的积分已经由 IIR 滤波器完成）。

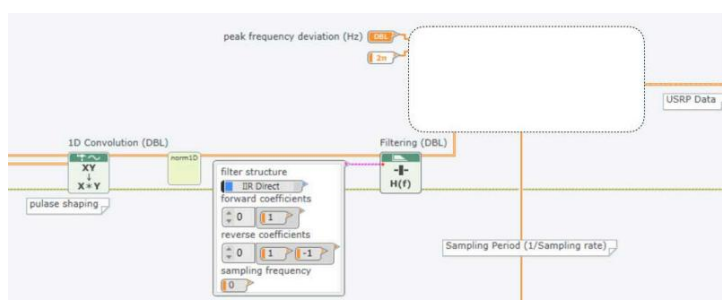


图 5-9

- 3) 峰值频率偏移量设置为 20KHz。

### 2.2 接收电路

在“RxBPSK.gvi”的基础上，构建一个完成 FSK 解调的接收电路。将文件名重命名为“RxFSK.gvi”。

在前面板中设置以下参数：



- ① IQ rate = 400k
  - ② carrier frequency = 1GHz
  - ③ gain = 0
  - ④ active antenna = RX2
  - ⑤ Message length = 1000
  - ⑥ Symbol rate = 10k
- 1) 继续使用基本接收电路中的一些功能模块（比如：匹配滤波器、脉冲对齐、帧同步、符号到比特数据转换、误比特率、星座图等）。
  - 2) （频率解调）完成图 5-10 所示部分原理图实现频率解调。

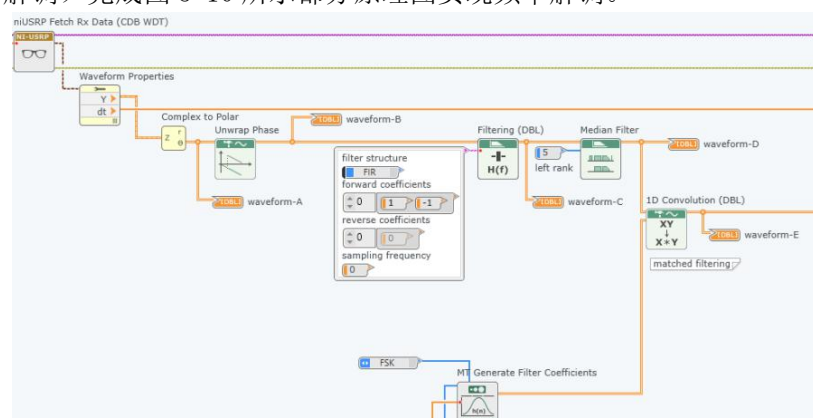


图 5-10

- 3) 给帧同步模块设定一个合适的阈值。
- 4) 确定是否译码得到正确的比特数据，并得到正确的星座图和误比特率。

### 2.3 问题 2

- 1) 对应两个不同的峰值频率偏差：20KHz 和 40KHz，给出并比较发送信号的频谱图，对结果进行解释（提示：利用 Carson 定理）。
- 2) 给出 waveform-A、B、C、D、E 的图形，将信号放大，解释产生这些波形的原因。
- 3) 将峰值频率偏差减小到 1kHz，跟 20kHz 时的星座图进行比较。
- 4) 解释图 5-9 和图 5-10 实现频率调制和解调的工作原理。

## 五、实验报告要求

- 1) 给出所设计的电路，记录实验数据并加以分析
- 2) 回答所有问题
- 3) 报告在下个实验开始前以书面或电子版形式递交

## 实验六：信道估算和纠正

### 一、实验目的

- (1) 了解信道估算的方法
- (2) 了解信道纠正的方法

### 二、实验设备

- |                      |     |
|----------------------|-----|
| (1) USRP 设备          | 1 台 |
| (2) 安装 LabVIEW 环境的电脑 | 1 台 |

### 三、实验概要

本实验中，使用训练序列对窄带单通道信道进行估算。信道估算的方法有很多，比如：最大似然估计、最小均方误差、线性最小二乘法。实验中，使用最简单的线性最小二乘法估算，以及加性高斯白噪声信道中的最大似然估计。信道估算后，取信道参数的倒数，就可以对信道进行纠正。

线性系统用下式表示：

$$Ax = y$$

这里的  $A$  是已知矩阵， $x$  是一个未知向量， $y$  是观察到的输出向量。假设  $A$  是一个满秩矩阵， $x$  的最小平方误差估算是：

$$\hat{x} = (A^H A)^{-1} A^H y$$

这里的  $A^H$  是矩阵  $A$  的共轭转置。 $y$  的平方误差定义为：

$$SE = \|A\hat{x} - y\|_2^2$$

信道可以建模为下面线性方程表示的系统：

$$y[n] = \sum_{l=0}^{L-1} h[l]t[n-l] + \omega[n]$$

$y[n]$  是接收到的符号， $h[n]$  是信道系数， $t[n]$  是  $N_t$  长度的训练序列， $n=0,1,\dots,N_t-1$ 。L 信道长度， $\omega[n]$  是加性高斯白噪声。为了确保训练序列从  $t[0]$  开始，从  $n=L-1$  开始求和。上面的式子可以用矩阵形式表示：

$$\begin{bmatrix} y[L-1] \\ y[L] \\ \vdots \\ y[N_t-1] \end{bmatrix} = \begin{bmatrix} t[L-1] & t[L-2] & \cdots & t[0] \\ t[L] & t[L-1] & \cdots & t[1] \\ \vdots & \ddots & \ddots & \vdots \\ t[N_t-1] & \cdots & t[N_t-L] & \end{bmatrix} \begin{bmatrix} h[0] \\ h[1] \\ \vdots \\ h[L-1] \end{bmatrix}$$

$$y = Ah$$

### 四、实验内容与步骤

#### 1. 信道估算

在这一部分实验中，要求完成以下三个子 VI 函数：**LLS.gvi**：完成线性最小二乘法估算；**Channel\_Estimation.gvi**：估算信道响应函数；**sim\_channel.gvi**：对 ISI 信道进行仿真。

### 1.1 线性最小二乘法估算

设计“LLS.gvi”，计算最小平方误差解并返回误差。输入输出变量如表 6-1 所示。

表 6-1 变量表

	终端名称	类型	描述
输入	$y$	一维双精度复数数组	观察向量
	$A$	二维双精度复数数组	系统矩阵
输出	$X\_hat$	一维双精度复数数组	最小平方误差解
	$Squared\ error$	双精度	平方误差

- 1) 使用“伪逆矩阵”模块（数学/线性代数），模块返回矩阵  $A$  的伪逆矩阵：

$$A_{pinv} = (A^H A)^{-1} A^H$$

- 2) 使用“ $A \times B$ ”模块（数学/线性代数）完成两个矩阵或一个矩阵和一个向量的相乘。
- 3) 使用“向量范数”模块计算一个向量的范数。默认的范数类型是 2。

### 1.2 信道估算

设计“Channel\_Estimation.gvi”，使用“LLS.gvi”估算信道。输入输出变量如表 6-2 所示。

表 6-2

	终端名称	类型	描述
输入	$Training\ sequence(t[n])$	一维双精度复数数组	训练序列
	$Channel\ length(L)$	整型	
	$Received\ Symbols(y[n])$	一维双精度复数数组	在接收机中接收到的符号
输出	$Channel\ Estimate(h[n])$	一维双精度复数数组	估算得到的信道系数
	$Channel\ Estimate\ Magnitude$	一维双精度数组	
	$Channel\ Estimate\ Phase$	一维双精度数组	
	$Rank\ A$	双精度	矩阵 $A$ 的秩
	$Squared\ error$	双精度	信道估算的平方误差

- 1) 完成图 6-1 空白部分电路 **Box1**，构建矩阵  $A$ 。注意矩阵  $A$  是  $(N_t - L + 1)$  行、 $L$  列。每一个 **for** 循环构建一行元素。开启“自动索引”功能，以构建二维矩阵（提示：使用“数组子集”和“反转一维数组”模块）。
- 2) 完成图 6-2 空白部分电路 **Box2**，构建向量  $y$ 。注意向量  $y$  的长度是  $(N_t - L + 1)$ ，索引值从  $L - 1$  开始（提示：使用“数组子集”模块）。

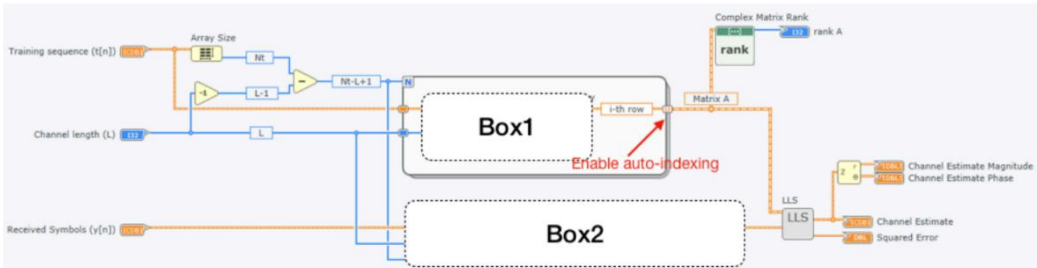
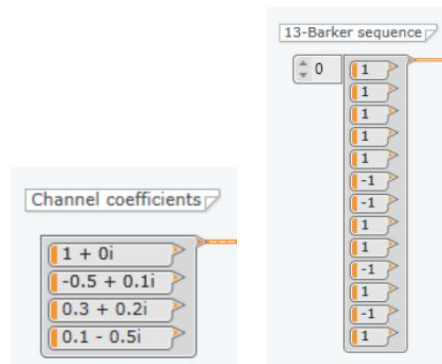


图 6-1

### 1.3 仿真

构建“sim\_channel.gvi”对信道估算进行仿真。使用以下信道系数，并与 13-Barker 序列进行卷积。卷积结果用来仿真接收到的符号。确认“Channel\_Estimation.gvi”能重新构建原来的信道系数。



#### 2. 问题 1

- 1) 使用正确的信道长度（本例为 4）运行仿真文件，给出估算的信道系数。结果是否与原来的信道匹配？
- 2) 如果信道长度估算过大或过小，将出现什么情况。改变信道长度值，大于 4 或小于 4，使用相同的信道系数，进行仿真。
- 3) 尝试不同的训练序列，比如： $+1-1+1-1+1-1+1-1+1-1+1$ 。进行仿真，结果会如何？请解释原因（提示：检查矩阵 A 的秩）。

#### 3. 信道纠正（单信道）

运行 BPSK 接收电路获取星座图，星座图以一个信道产生的随机相位数旋转。在这部分实验中，我们将运行窄带相位估算（信道长度  $L=1$ ），纠正信道以补偿相位旋转。

- 1) 更改帧同步模块，如图 6-2 所示。变化如下：输入输出符号是双进度复数类型；输出同步数据符号的长度是一帧；原始的训练符号（26-Barker 序列）作为输入；从接收符号中获取训练序列符号作为模块输出；相关值经过归一化，“threshold”的范围为[0,1]。完成图 6-2 中空白部分电路 Box1。

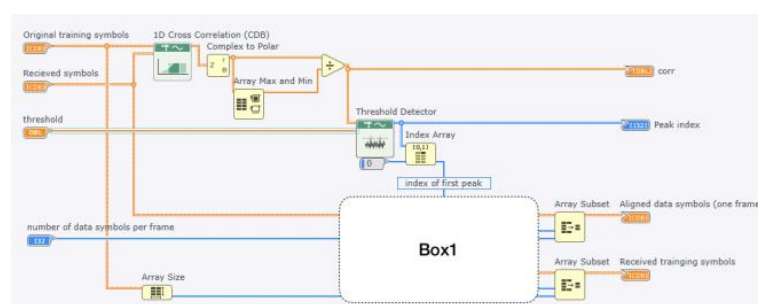


图 6-2

- 2) 使用接收到的训练符号和信道估算模块纠正信道。在 BPSK 接收电路的基础上，完成图 6-3 中空白部分电路 Box1。注意：将下采样后的归一化模块移除。

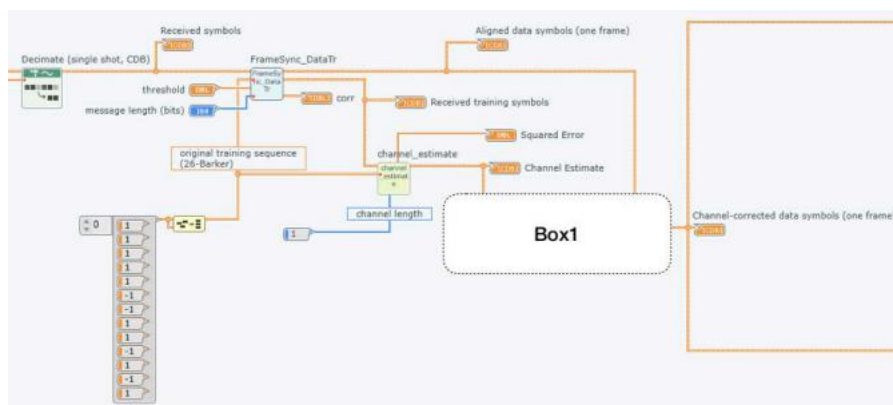


图 6-3

#### 4. 问题 2

1) 参数设置参考 BPSK 发送和接收电路部分实验，运行 BPSK 发送电路发送 BPSK 信号，运行具有信道估算和纠正功能的 BPSK 接收电路。给出信道纠正前后的 BPSK 星座图（分别是“aligned data symbols”和“channel-corrected data symbols”），解释原因。

#### 5. 信噪比

信号  $x[n]$  的平均功率可以由下面的式子计算得到：

$$\sigma_x^2 = \frac{1}{N} \sum_{n=0}^{N-1} |x[n]|^2$$

信噪比经常用 dB 表示为：

$$SNR_{dB} = 10 \log_{10}(SNR) = 10 \log_{10} \frac{\sigma_x^2}{\sigma_w^2}$$

式中  $\sigma_x^2$  和  $\sigma_w^2$  分别是信号和噪声的平均功率。

完成图 6-4 中空白部分电路 Box1，从接收符号中计算噪声。为此需要从接收到的数据符号中提取出原始符号。确定并填写图中平均 BPSK 信号功率的理论值。

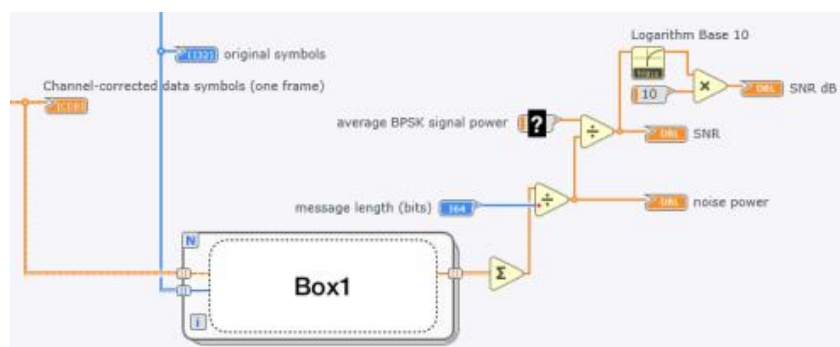


图 6-4

#### 6. 问题 3

1) 减小发送端信号功率（比如 0.01），给出减小功率前后的信噪比和星座图。

## **五、实验报告要求**

- 1) 给出所设计的电路，记录实验数据并加以分析。
- 2) 回答所有问题。
- 3) 报告在下个实验开始前以书面或电子版形式递交。



## 实验 7 载波频率偏移的估算和纠正

### 一、实验目的

- (1) 掌握自参考帧同步的方法
- (2) 了解载波频率偏移的估算
- (3) 了解载波频率偏移的纠正

### 二、实验设备

- |                      |     |
|----------------------|-----|
| (1) USRP 设备          | 1 台 |
| (2) 安装 LabVIEW 环境的电脑 | 1 台 |

### 三、实验概要

在大多数通信系统中，接收端的载波频率与发送端的载波频率不一致，这种情况称为载波发生了偏移。在本实验中，首先需要检测存在载波频率偏移（CFO）情况下的训练序列，然后再进行 CFO 的估算，再纠正 CFO 以接收正确的数据符号。

通常情况下，在存在 CFO 的情况下，使用已知训练序列的相关性来检测训练序列是不可能的。我们使用自参考帧同步的方法。发送端发送的信号中存在两段相邻的训练序列，将第一段  $N_t$  长度的符号与第二段  $N_t$  长度的符号作相关，实现数据帧的同步，检测出训练序列，然后进行 CFO 的估算和纠正。

### 四、实验内容与步骤

在本实验中，我们需要完成以下 4 个子 VI 函数：自参考帧同步 FrameSync\_selfref.gvi; CFO 估算 CFO\_estimation.gvi; CFO 纠正 CFO\_correction.gvi; CFO 仿真 sim\_CFO.gvi。

#### 1. 自参考帧同步

在存在载波频率偏移的情况下，发送端发送的数据序列前面插入两段训练序列。假设  $y(n)$  是接收到的数据符号， $N_t$  是训练序列的长度。在接收端，对接收到的数据，以  $N_t$  为长度单位，作移位相关，如下面表达式所述。设定合适的阈值，对移位相关结果作峰值检测，将超过阈值的第一个数据对应的位置，判定为数据帧训练序列的起始位置。

$$R[n] = \left| \sum_{k=0}^{N_t-1} y^*[n+k] y[n+k+N_t] \right|$$

为本实验新建一个项目文件，在项目文件下新建一个 VI 函数：FrameSync\_selfref.gvi。输入输出变量如表 7-1 所示：

表 7-1

	终端名称	类型	描述
输入	Received symbols	一维双精度复数数组	接收符号
	Training sequence length	双精度	训练序列长度
	Threshold	双精度	阈值
	Number of data symbols per frame	整型	一帧的数据符号个数
输出	Correlation	双精度	自相关值
	Peak indices	一维有符号整型数组	峰值索引
	Aligned data symbols	一维双精度复数数组	同步数据符号
	Received training symbols	一维双精度复数数组	接收到的训练序列

完成图 7-1 中空部分电路 Box1 和 Box2，实现自参考帧同步。

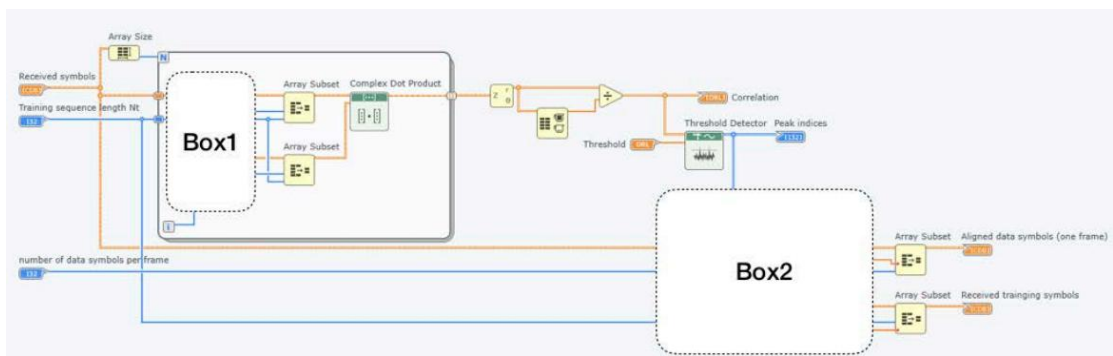


图 7-1

## 2. CFO 估算

假设  $f_e$  是载波频率偏移,  $T_s$  是符号周期,  $N_t$  是一段训练序列的长度,  $y(n)$  是接收到的第一段训练序列:

$$y[n] = e^{j2\pi f_e n T_s} t[n]$$

那么  $y(n+N_t)$  是接收到的第二段训练序列, 可以表示成:

$$\begin{aligned} y[n + N_t] &= e^{j2\pi f_e (n+N_t) T_s} t[n + N_t] \\ &= e^{j2\pi f_e N_t T_s} y[n] \\ &= \alpha y[n] \end{aligned}$$

对  $\alpha$  作最小二乘法估算:

$$\alpha = \frac{\sum_{n=0}^{N_t-1} y^*[n] y[n + N_t]}{\sum_{n=0}^{N_t-1} y^*[n] y[n]}$$

进而可以得到载波频率偏移的估算表达式:

$$\hat{f}_e = \frac{\text{Phase}(\sum_{n=0}^{N_t-1} y^*[n] y[n + N_t])}{2\pi N_t T_s}$$

新建一个 VI 函数: CFO\_estimation.gvi, 估算载波频率偏移  $f_e$ 。输入输出变量如表 7-2 所示:

表 7-2

	终端名称	类型	描述
输入	Received training symbols	一维双精度复数数组	接收到的训练序列
	Symbol rate	双精度	符号率
输出	Estimated CFO	双精度	估算得到的 CFO

完成图 7-2 中空白部分电路 Box1。

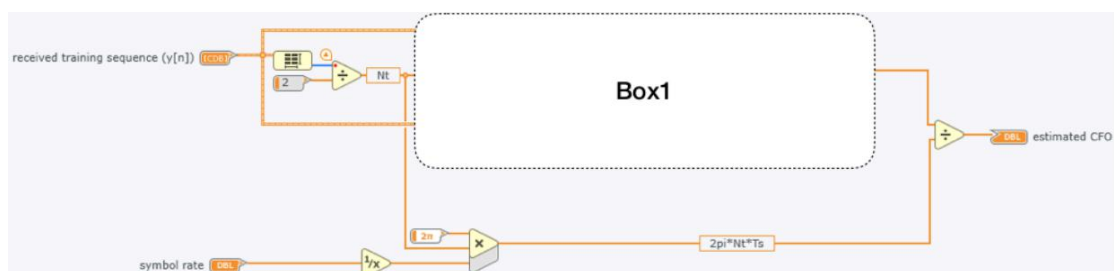


图 7-2

3. CFO 纠正

对载波频率偏移完成估算后，就可以对接收到的数据符号进行载波频率偏移的纠正。纠正的数据符号可以用下面这个表达式表示。

$$\hat{y}[n] = e^{-j2\pi\hat{f}_e n T_s} y[n], \quad n = n_d, n_d + 1, \dots, n_d + N - 1$$

其中的 n 取值从  $n_d$ ,  $n_d+1$ , 一直到  $n_d+N-1$ ,  $\hat{f}_e$  是估算得到的 CFO,  $n_d$  是初始偏移量, N 是符号的个数。

新建一个 VI 函数：CFO\_correction.gvi，将 CFO 估算值作为输入，对符号进行纠正。表 7-3 是输入输出变量表：

表 7-3

	终端名称	类型	描述
输入	Estimated CFO	双精度	估算得到的 CFO
	Symbol rate	双精度	符号率
	Symbols	一维双精度复数数组	符号
	Initial offset	有符号整型	初始偏移量
输出	Corrected symbols	一维双精度复数数组	纠正后的符号

完成图 7-3 中空白部分电路 Box1 和 Box2。

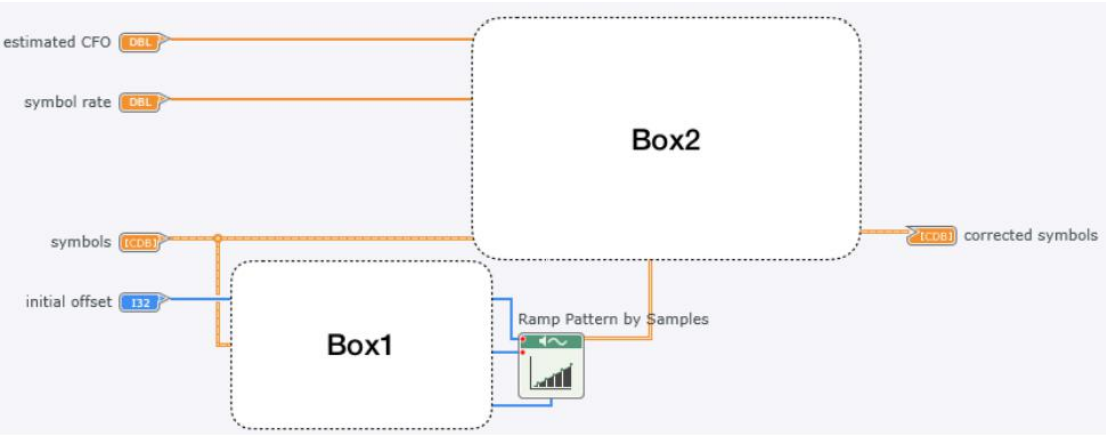


图 7-3

4. CFO 仿真

为了验证载波频率偏移估算、纠正电路功能的正确性，搭建一个载波频率偏移产生、估算和纠正的仿真电路。先由 MT generate Bits 模块生成 5 级伪随机码信号，作为发送的信号源，在它前面再插入两段相同的训练序列：11111-1-111-11-11，再加载波频率偏移，对有载波频率偏移的数据符号作帧同步，提取出训练序列和真正需要的数据，然后作频率估算，最后对接收数据作频率纠正。

使用 CFO\_estimated 模块和 CFO\_correction 模块完成图 7-4 电路空白部分 Box1。

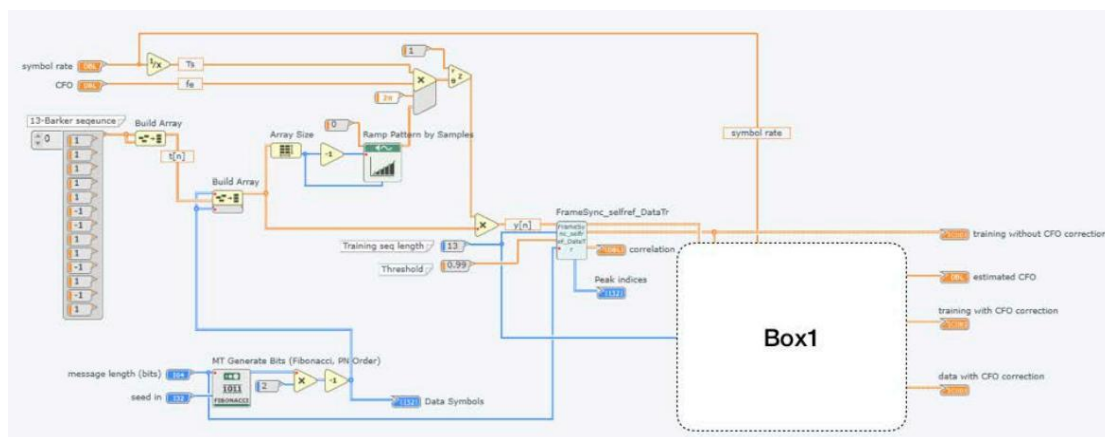


图 7-4

前面板的参数设置如下：

- ① Symbol rate = 100k
- ② CFO = 1kHz
- ③ Message length = 1000
- ④ Seed in = 169

#### 5. 问题 1

- 1) 运行 CFO 仿真电路，获得 CFO 的估算值，CFO 纠正前、后的训练序列星座图，以及帧同步模块的自相关结果。
- 2) 在符号率为 100kHz 的情况下，增加 CFO 值，直到电路 CFO 估算和纠正失败，能估算出的最大 CFO 值是多少？与理论值比较。
- 3) 将“seed in”的值改为 133，给出自相关结果。会出现什么问题？讨论这个问题的解决方法。

#### 6. BPSK 接收系统中的 CFO 和信道纠正

在实验 6 对 BPSK 接收电路作信道估算和纠正的基础上修改电路，将电路中的帧同步模块替换成自参考帧同步模块，在信道估算前添加 CFO 估算和纠正电路。功能框图如图 7-5 所示。使用经过 CFO 纠正过的训练序列进行信道估算，再对经过 CFO 纠正过的数据符号进行信道纠正。

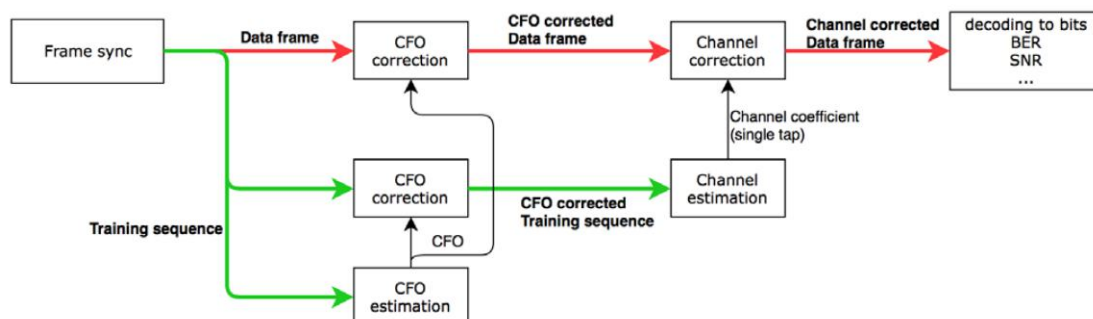


图 7-5

#### 7. 问题 2

- 1) 完成图 7-5 电路图。

参数设置如下，

发送电路：

①IQ rate: 1M

②Symbol rate: 100k

③Carrier frequency: 1GHz

接收电路：

①IQ rate: 1M

②Symbol rate: 100k

③Carrier frequency: 1GHz+1kHz (CFO)

给出下列数据符号的星座图：帧同步后的数据信号；经过 CFO 纠正后的数据符号；经过信道纠正后的数据符号。

- 2) 尝试超出最大能估算出的 CFO 值的载波频率偏移情况，给出估算得到的 CFO 值，以及经过 CFO 纠正后的数据星座图。对实验结果作解释。

## 五、实验报告要求

- 1) 给出所设计的电路，记录实验数据并加以分析。
- 2) 回答所有问题。
- 3) 报告在下个实验开始前以书面或电子版形式递交。