西安交通大学
XI'AN JIAOTONG UNIVERSITY

通信原理实验

2 差错控制

姓　名：张翠翠

办公室：西一楼520

邮　箱：zhangcuicui@mail.xjtu.edu.cn

内容回顾

调制解调
差错控制（信道编码+交织）
同步处理
OFDM
信道均衡
调频收音机

内容

- 纠错编码

  线性码和非线性码

  卷积码和分组码

- 差错控制技术

  交织+纠错编码=FEC前向纠错

- 差错控制编码在星座映射之前还是之后?
- 差错控制编码是在比特流做层的还是符号流层做的？为什么?

内容

- DVB-S2：交织+BCH+LDPC
- 第4代移动通信：交织、咬尾卷积码、Turbo
- 第5代移动通信：交织、Polar码、LDPC

- 线性分组码：汉明码
- 循环码：BCH码、CRC
- 卷积码：Turbo码

提纲

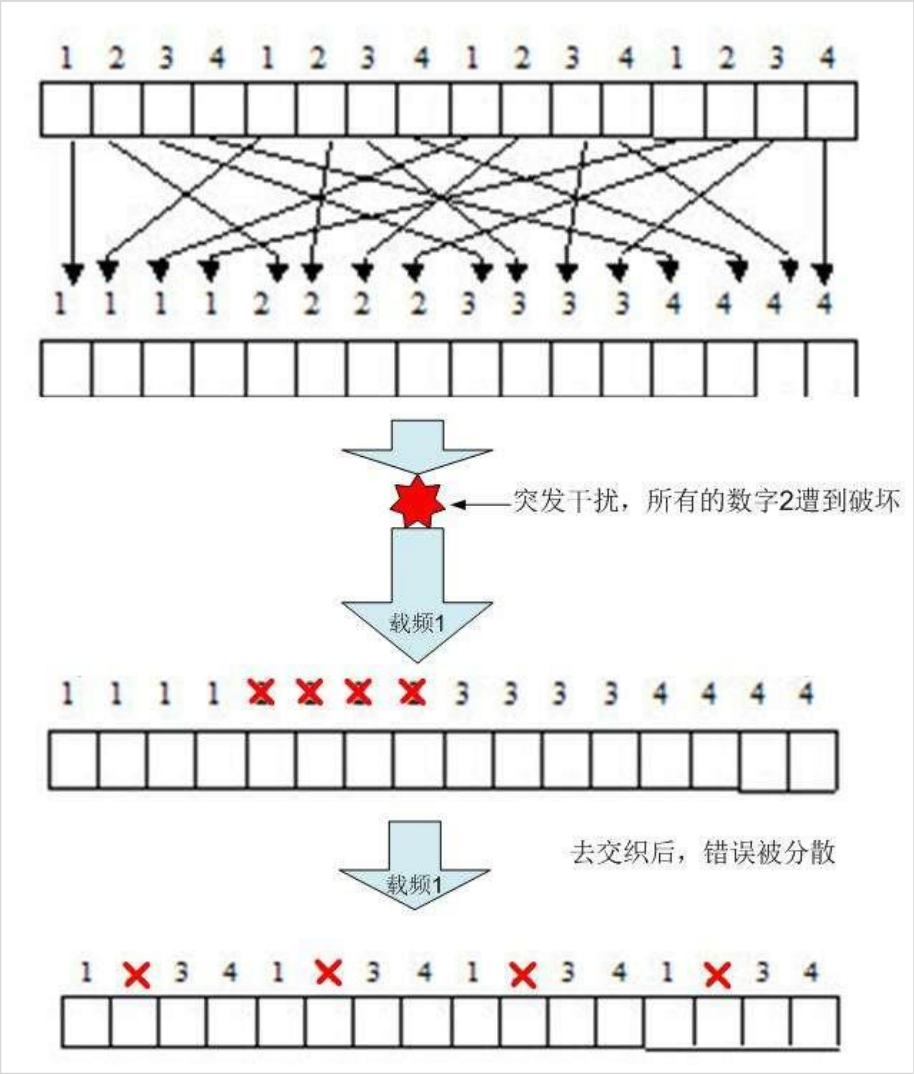
- 交织编码技术
- 线性分组码
- 七四海明码
- 发送端基带处理
- 实验任务
- 实验报告要求

交织编码技术

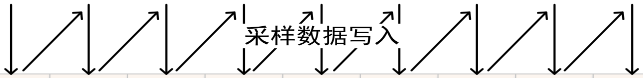
- 交织编码技术：

  抗突发错的一种有效编码技术

  在DVB、4G、5G中都有使用

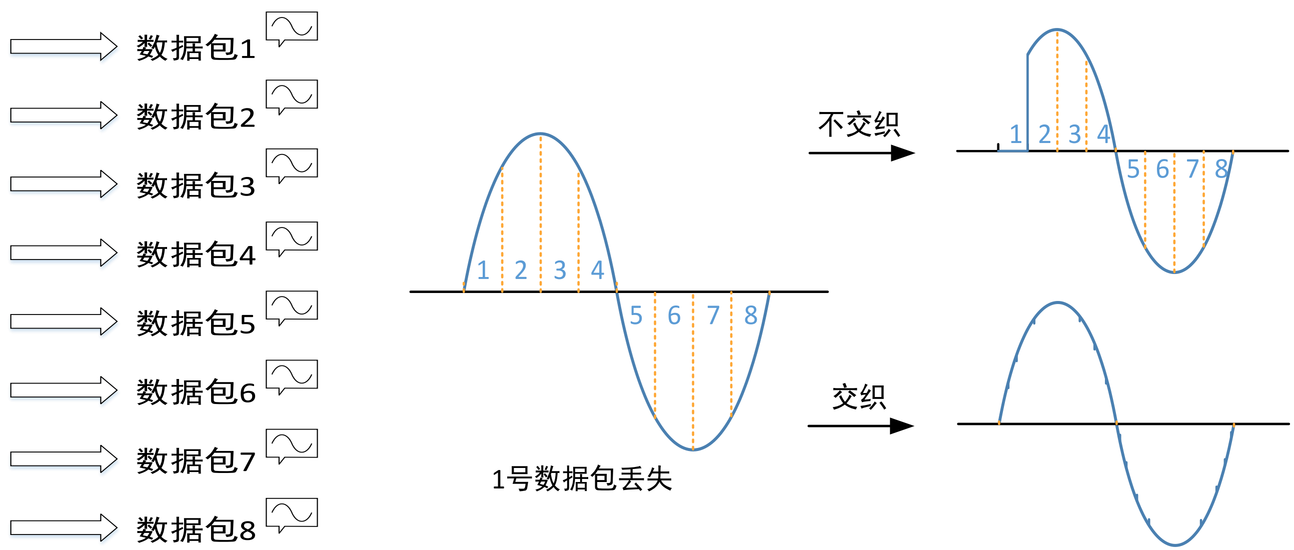


交织编码技术



0	8	16					4080	4088
1	9	17					4081	4089
2	10	...					4082	4090
3	11						4083	4091
4	12						4084	4092
5	13					...	4085	4093
6	14					4078	4086	4094
7	15					4079	4087	4095

RAM存储空间（每个小单元存储一个采样点）



- 具体实现怎么做？Matlab、C、JAVA、Verilog
- 有没有带来冗余？有没有纠错能力？
- 配合信道编解码使用，提高系统的纠错能力

线性分组码

■ 基本概念

- （信息位+监督位）组成，降低有效性、换取可靠性
- 增加冗余，与信源编码的降低冗余相反
- 监督位与信息位之间是线性关系

➢ （n,k）线性分组码: k位信息位+（n-k）位监督位

一个长度为n且具有2^k个码字的二进制分组码，当且仅当，它的2^k个码字在n元组的向量空间中形成k维子空间，称为（n，k）线性分组码

线性分组码

从线性代数向量空间的角度看，线性分组码（n,k）是n元组向量空间V中的k维子空间，存在k个线性独立的码字，形成基向量，张成码字空间C。每一个码字v可以看成是以输入信息u作为系数的基向量的线性组合。

$$u=(u_0,u_1,\cdots,u_{k-1}) \longrightarrow \boxed{\text{信道编码器}} \longrightarrow v=(v_0,v_1,\cdots,v_{n-1})$$

$$v=u_0*\boldsymbol{g_0}+u_1*\boldsymbol{g_1}+\cdots+u_{k-1}*\boldsymbol{g_{k-1}}$$

$$G=\begin{bmatrix}\boldsymbol{g_0}\\ \boldsymbol{g_1}\\ \vdots\\ \boldsymbol{g_{k-1}}\end{bmatrix}=\begin{bmatrix}g_{0,0} & g_{0,1} & \cdots & g_{0,n-1}\\ g_{1,0} & g_{1,1} & \cdots & g_{1,n-1}\\ \vdots & \vdots & \cdots & \vdots\\ g_{k-1,0} & g_{k-1,1} & \cdots & g_{k-1,n-1}\end{bmatrix}$$

$$\boldsymbol{v}=\boldsymbol{u}*\boldsymbol{G}$$

矩阵G为线性分组码的生成矩阵
线性分组码是由矩阵G的行向量张成的子空间

线性分组码

矩阵G的零空间N（G）的维度是n-k。因而可以找出n-k个线性独立的向量，张成G的零空间N（G）。

$$H=\begin{bmatrix}\boldsymbol{h_0}\\ \boldsymbol{h_1}\\ \vdots\\ \boldsymbol{h_{k-1}}\end{bmatrix}=\begin{bmatrix}h_{0,0} & h_{0,1} & \cdots & h_{0,n-1}\\ h_{1,0} & h_{1,1} & \cdots & h_{1,n-1}\\ \vdots & \vdots & \cdots & \vdots\\ h_{n-k-1,0} & h_{n-k-1,1} & \cdots & h_{n-k-1,n-1}\end{bmatrix}$$

$$G*H^T=0_{(k,n-k)}$$

矩阵H称为线性分组码的校验矩阵

$$\boldsymbol{v}*H^T=\boldsymbol{u}*\boldsymbol{G}*H^T=0$$

$$\boldsymbol{v}=\boldsymbol{u}*\boldsymbol{G}$$

G和H是线性分组码设计的核心，G和H确定后即确定了该线性分组码

线性分组码

设接收到的码字为$v' = v \oplus e$

$$S = v' * H^T = (v \oplus e) * H^T = e * H^T$$
$$= [e_0, e_1, \cdots, e_{n-1}] * \begin{bmatrix} h_{0,0} & h_{1,0} & \cdots & h_{n-k-1,0} \\ h_{0,1} & h_{1,1} & \cdots & h_{n-k-1,1} \\ \vdots & \vdots & \cdots & \vdots \\ h_{0,n-1} & h_{1,n-1} & \cdots & h_{n-k-1,n-1} \end{bmatrix}$$
$$= e_0 * h_0 + e_1 * h_1 + \cdots e_{n-1} * h_{n-1}$$

S称为伴随式，S只跟错误位置向量有关，与发送的原码字无关

如果该码字能纠t个错，也就是说e中最多有t个1.
要从S中纠t个错，H的任何2t列必须线性无关。

$$e1_0 * h_0 + e1_1 * h_1 + \cdots e1_{n-1} * h_{n-1} \neq e2_0 * h_0 + e2_1 * h_1 + \cdots e2_{n-1} * h_{n-1}$$

海明码（7，4）

Bit #	1	2	3	4	5	6	7
Transmitted bit	p_1	p_2	d_1	p_3	d_2	d_3	d_4
p_1	Yes	No	Yes	No	Yes	No	Yes
p_2	No	Yes	Yes	No	No	Yes	Yes
p_3	No	No	No	Yes	Yes	Yes	Yes

https://en.wikipedia.org/wiki/Hamming%287,4%29

海明码（7，4）

Bit #	1	2	3	4	5	6	7
Transmitted bit	p_1	p_2	d_1	p_3	d_2	d_3	d_4
p_1	Yes	No	Yes	No	Yes	No	Yes
p_2	No	Yes	Yes	No	No	Yes	Yes
p_3	No	No	No	Yes	Yes	Yes	Yes

$$G = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

$$H^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$G * H^T = 0$$

$$S = e * H^T = e_0 * h_0 + e_1 * h_1 + \cdots e_{n-1} * h_{n-1} = h_i$$

Golay码

Golay（23，12，7）是海明码的一种，是23维空间中的12维子空间，有效码字个数为2^12=4096个，最小码距为7. 可以纠3位错，检7位错。

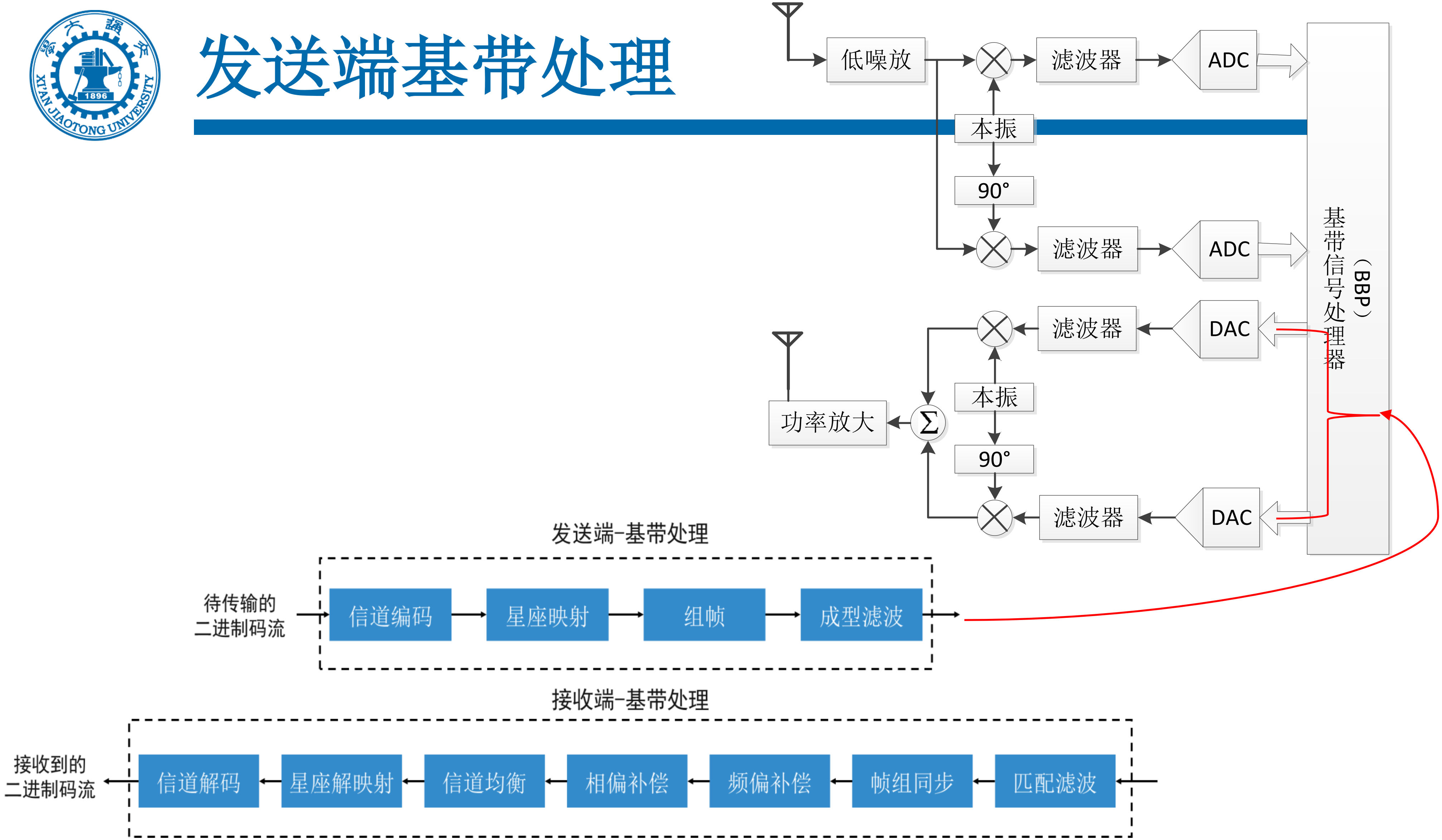
Golay（ 23，12，7 ）是完备码、是循环码

```
G=[1 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 ;
   0 1 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 1 0 0 0 1 0 ;
   0 0 1 0 0 0 0 0 0 0 0 0 1 0 1 1 1 0 0 0 1 0 1 ;
   0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 1 0 1 1 ;
   0 0 0 0 1 0 0 0 0 0 0 0 1 1 1 0 0 0 1 0 1 1 0 ;
   0 0 0 0 0 1 0 0 0 0 0 0 1 1 0 0 0 1 0 1 1 0 1 ;
   0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 1 1 0 1 1 ;
   0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 1 1 0 1 1 1 ;
   0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 1 1 0 1 1 1 0 ;
   0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 1 1 0 1 1 1 0 0 ;
   0 0 0 0 0 0 0 0 0 0 1 0 1 0 1 1 0 1 1 1 0 0 0 ;
   0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 1 1 1 0 0 0 1 ];
```

```
H=[1 1 1 0 1 1 1 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 ;
   1 1 0 1 1 1 0 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 ;
   1 0 1 1 1 0 0 0 1 0 1 1 0 0 1 0 0 0 0 0 0 0 0 ;
   1 1 1 1 0 0 0 1 0 1 1 0 0 0 0 1 0 0 0 0 0 0 0 ;
   1 1 1 0 0 0 1 0 1 1 0 1 0 0 0 0 1 0 0 0 0 0 0 ;
   1 1 0 0 0 1 0 1 1 0 1 1 0 0 0 0 0 1 0 0 0 0 0 ;
   1 0 0 0 1 0 1 1 0 1 1 1 0 0 0 0 0 0 1 0 0 0 0 ;
   1 0 0 1 0 1 1 0 1 1 1 0 0 0 0 0 0 0 0 1 0 0 0 ;
   1 0 1 0 1 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 ;
   1 1 0 1 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 ;
   1 0 1 1 0 1 1 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 ];
```

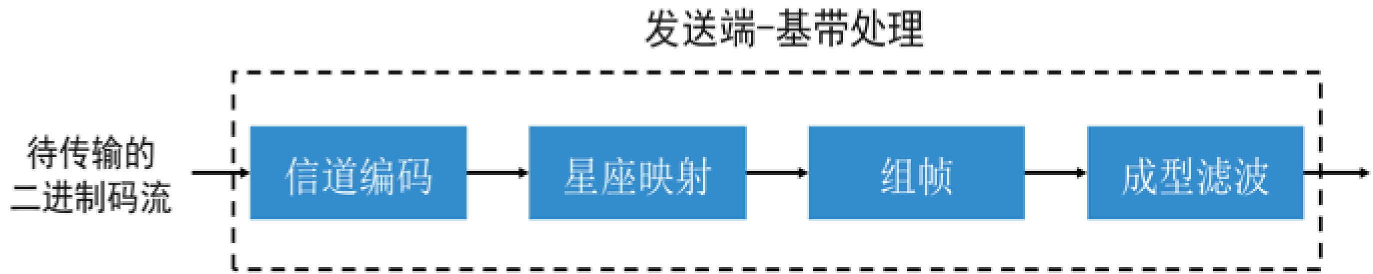
https://en.wikipedia.org/wiki/Binary_Golay_code

发送端基带处理



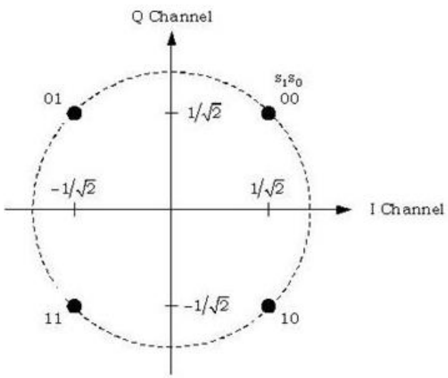
Matlab上基带处理

发送端基带处理



0100110101110010001101010 → 74海明码 → 0100xxx1101xxx0111xxx0010xxx0011xxx0101xxx

0100xxx1101xxx0111xxx0010xxx0011xxx0101xxx



QPSK
星座映射 →

I: -0.707 0.707 …
Q: 0.707 0.707 …

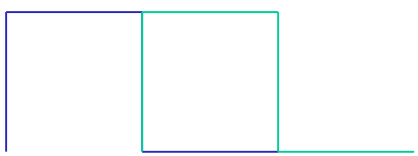
组帧

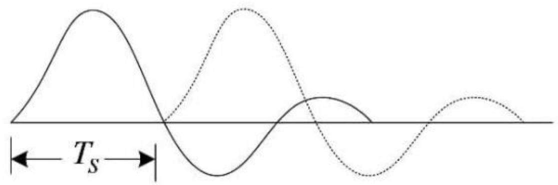
I: 1 1 -1 1 -1 -1 -1 -0.707 0.707 …
Q: 0 0 0 0 0 0 0 0.707 0.707 …

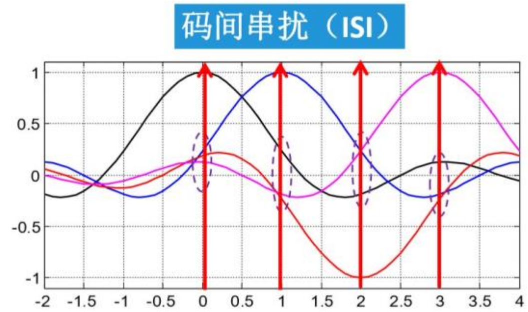
添加M序列对应的调制符号
0010111

发送端基带处理

由于信道的带宽限制，当信号在有限带宽信道中传输时，会导致码元信号在时域上的拖尾，拖尾叠加到其它码元上影响其它码元的正确判决，称之为码间串扰ISI.







发端符号
时域上带限
频域上无限

→

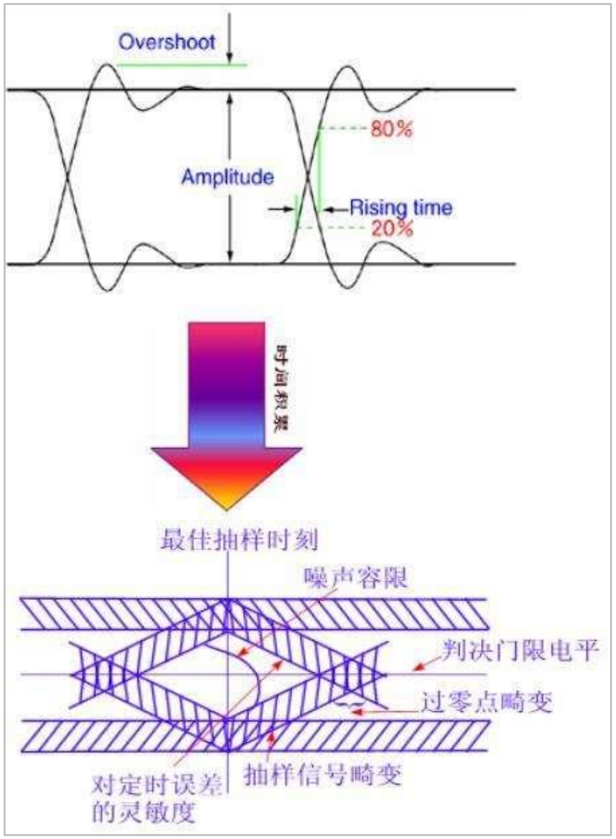
信道
(频域带限)

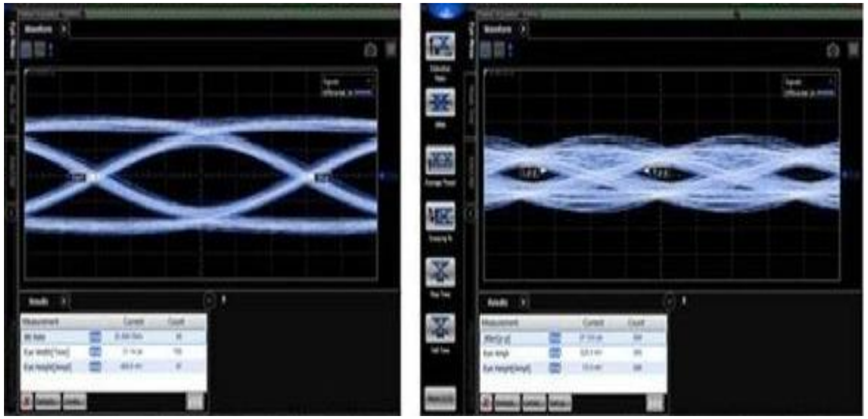
→

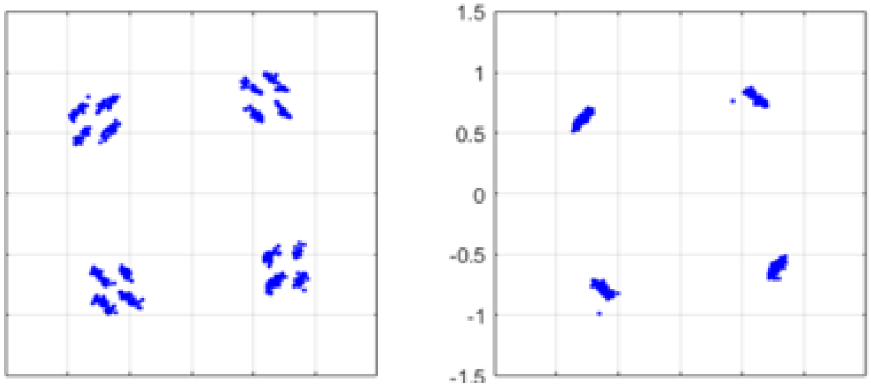
收端符号
时域上拖尾产生ISI
频域上带限

发送端基带处理

- 二元幅度调制系统中，用收端信号的眼图来观察ISI
- 多元幅相调制系统中，用收端信号的星座云来观察ISI







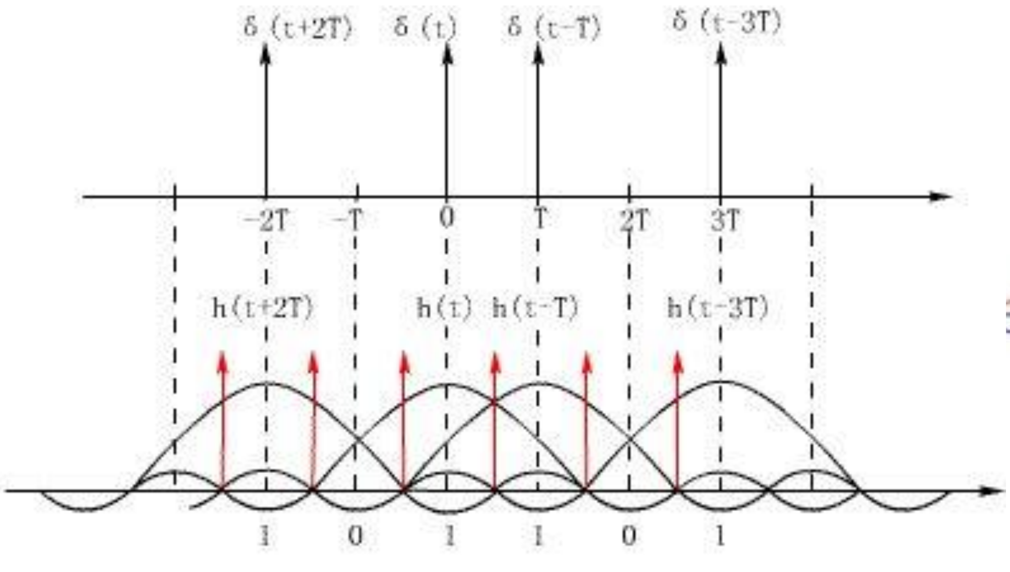
发送端基带处理

https://zhuanlan.zhihu.com/p/439814485

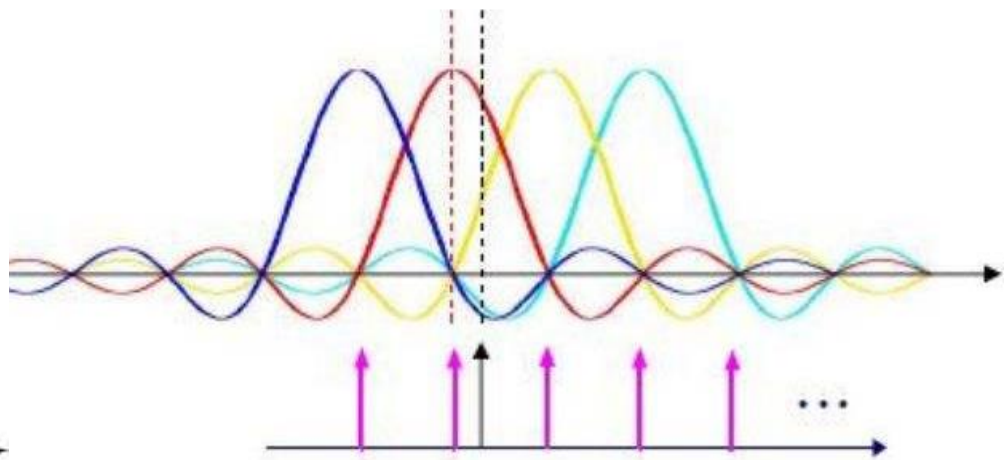
- 能不能完全消除信号的拖尾?

  信道的带限特性是客观实际存在的，实际中都是带限系统

- ➢ 但是可以想办法让拖尾按照我们想要的方式出现—不影响其它码元的正确判决



附图 18



发送端基带处理

- 能不能完全消除信号的拖尾？

  信道的带限特性是客观实际存在的，实际中都是带限系统

➢ 但是可以想办法让拖尾按照我们想要的方式出现

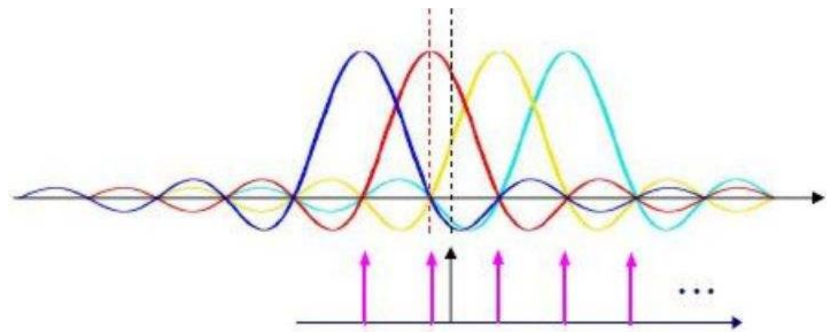
所有其它码元的拖尾和为0

$$x(t) = \sum_{n=-\infty}^{n=+\infty} a_n \delta(t - nT)$$

$$s(t) = x(t) * h(t) = \sum_{n=-\infty}^{n=+\infty} a_n h(t - nT)$$

$$s(t_k) = a_k h(0) + \boxed{\sum_{n \neq k} a_n h(kT - nT)}$$

为0



$$\Rightarrow \quad \boxed{h(kT) = \begin{cases} 1, & k = 0 \\ 0, & k \neq 0 \end{cases}}$$

无码间串扰的时域条件

发送端基带处理

https://zhuanlan.zhihu.com/p/439814485

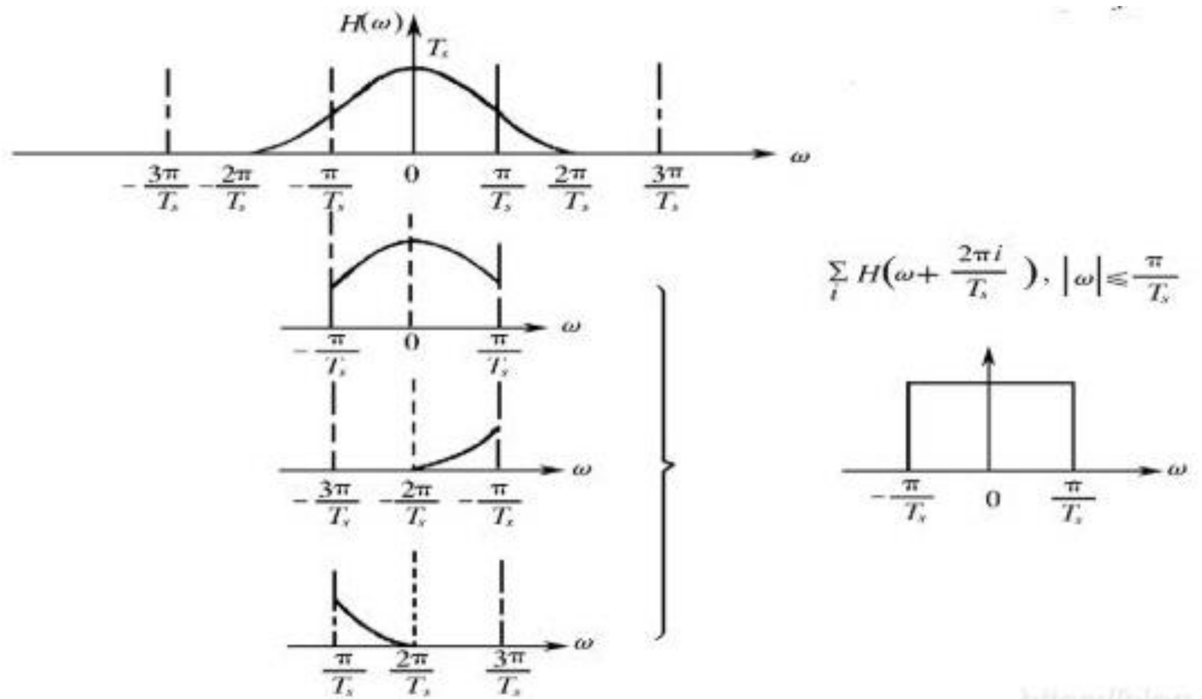
- 所有其它码元的拖尾和在当前码元判决处为0
- $H(\omega)$ 频移相加后，得到某一常数

$$h(kT)=\begin{cases}1, & k=0\\ 0, & k\neq 0\end{cases}$$

$$\boxed{\sum_i H\left(\omega+\frac{2\pi i}{T}\right)=T, \qquad |\omega|\leq\frac{\pi}{T}}$$

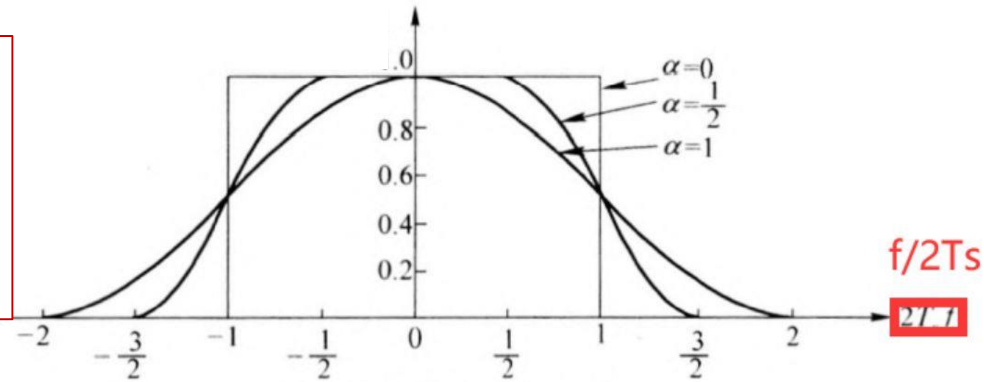
无码间串扰的频域条件

$H(\omega)$ 不唯一



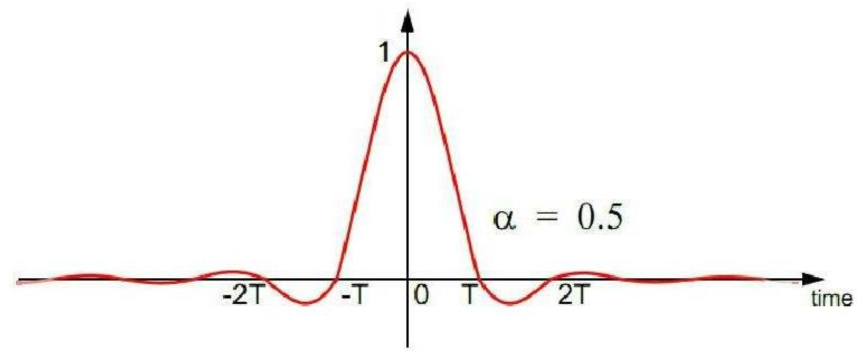
- 升余弦滤波器

$$
\begin{aligned}
G(f) &= T & |f| \le \frac{1-\alpha}{2T} \\
&= T\cos^2\left[\frac{\pi T}{2\alpha}\left(|f| - \frac{1-\alpha}{2T}\right)\right] & \frac{1-\alpha}{2T} < |f| \le \frac{1+\alpha}{2T} \\
&= 0 & \frac{1+\alpha}{2T} < |f|
\end{aligned}
$$



(a) 升余弦滤波器的频率响应

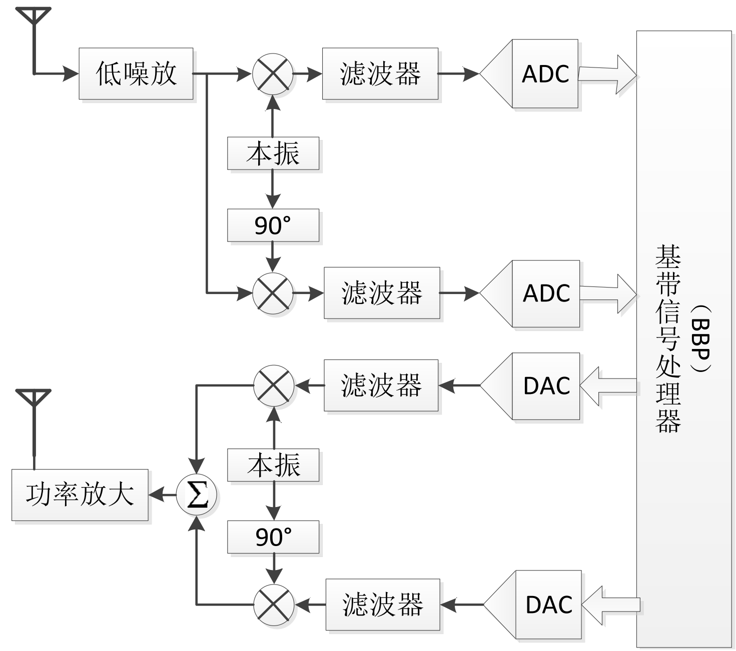
$$
g(t) = \left(\frac{\sin(\pi t/T)}{\pi t/T}\right)\left(\frac{\cos(\alpha\pi t/T)}{1-(2\alpha t/T)^2}\right) \qquad -\infty < t < +\infty
$$



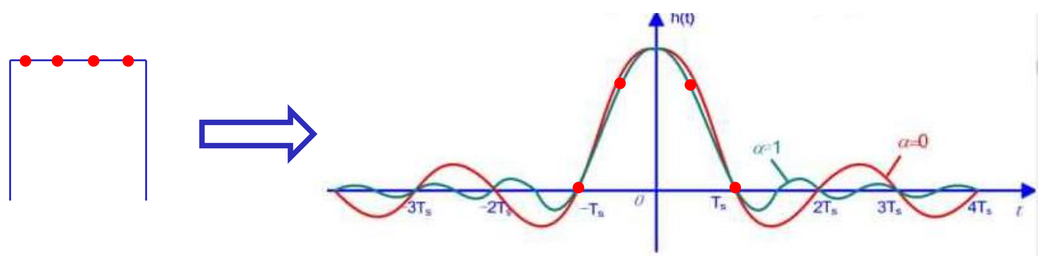
发送端基带处理

星座映射并组帧后 → 成型滤波器（根升余弦） → 信道 → 匹配滤波器（根升余弦） → 收端处理

I:	1	1	-1	1	-1	-1	-1	-0.707	0.707	…
Q:	0	0	0	0	0	0	0	0.707	0.707	…



DAC采样率40MHz
保证一个符号至少4个样点，越多越好，10个足够
以4个样点为例：则符号率为10MHz
具体实现：将I和Q路分别经过根升余弦滤波器



实验任务

- 信道编解码的实现和验证

1.在已有的demo基础上，发送端合适位置加入信道编码（可自由选择编码类型及方式）

2.在接收端合适位置加上对应的信道解码

3.系统测试，保证编解码的正确性，收端能得到发端信息

4.计算误码率，对比添加信道编解码和未添加信道编解码的误码率

实验报告要求

2 信道编解码

一 实验内容

1.1 信道编解码的实现和验证

二 实验原理

2.1 差错控制编码的分类
2.2 线性分组码的编码、校验、纠错原理
2.3 码间串扰产生的原因
2.4 成型滤波、匹配滤波的原理和作用

三 实验结果图示及分析

3.1 信道编解码的具体实现
3.2 从误码率分析系统性能

四 思考题

4.1 符号率和 DAC 采样率的关系
4.2 线性分组码中码距、纠错位数、查错位数的关系