《以太网》

[《以太网》 1](#_Toc185328102)

[第一节：CRC算法 3](#_Toc185328103)

[1.1CRC码 3](#_Toc185328104)

[1.1.1CRC定义 3](#_Toc185328105)

[1.1.2任意CRC硬件实现 4](#_Toc185328106)

# 第一节：CRC算法

CRC是一种常用于高速结构或者总线的信道编码，它具备检错能力，它可以对任意比特数据的进行检查。

CRC= （N,K） ，K为任意bit，监督位为crc码，也可以是任意比特

通过以上可以看出，CRC是一种**扩展性和灵活性较高**的信道分组码，因此在数字通信中得到非常广泛的应用，比如USBx.x，ETH,PCIE,…..。

## 1.1CRC码

Crc校验的核心就是计算监督位，这里监督位我们习惯上叫做CRC码，任何信道分组码主要是计算监督位，这里就是计算crc码。

### 1.1.1CRC定义

CRC码倍定义为，(N,K),监督位为N-K，将Kbit的信息为乘以2N-K,模2除以生成多项式，最后得到的余数即为CERC码。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| N-1 | N-2 | … | N-K | N-K-1 | … | 0 |
|  |  |  |  |  |  |  |

CRC

CRC码（监督位）

**Crc码=Data[K-1:0]\*2n-k%plot**

信息位

例如：crc=(11,8) ,data[7:0]=8’b1001\_1100, plot = 4’b1001

第一步：data[7:0]\*2(N-K) = data[7:0]\*211-8 = {data[7:0],3’b000}= 11’b1001\_1100\_000

第二步：求余数

1 0 0 1 1 0 1

1 0 0 1 1 1 0 0 0 0 0

1001

1 0 0 1

1 0 0 1

0 0 0 0 1 1 0 0

0 1 0 1 0

1 0 0 1

1 0 0 1

0 1 0 1

0 0 1 1 0 0

### 1.1.2任意CRC硬件实现

G1

G0=1

data

G0

data

C1

CN-1

C…

C0

Gn.

G..

G2

练习：G(x )= x5+x3+1 实现crc5的CRC码计算电路实现

G= 6’b101001

G3=1

G1=0

G5=1

G4=0

G2=0

C0

C1

C2

C4

C3

reg [4:0]   crc;

fd= crc[4]^data;

Crc\_nex[0] = fd;

Crc\_nex[1] = crc[0] ;

Crc\_nex[2] = crc[1];

Crc\_nex[3] = crc[2]^fd;

Crc\_nex[4] = crc[3] ;

