

Laboratory

1

## Data Link Layer

### *Implementation of character count and Bit Stuffing method*

#### Objective

One of The biggest things in the world is Internet. The communication can be established using OSI or TCP/IP Models. In both models common layer is Data Link layer.

The data link layer is the second layer in the OSI Model. The three main functions of the data link layer are to deal with **transmission errors**, **regulate the flow of data**, and provide a **well-defined interface to the network layer**.

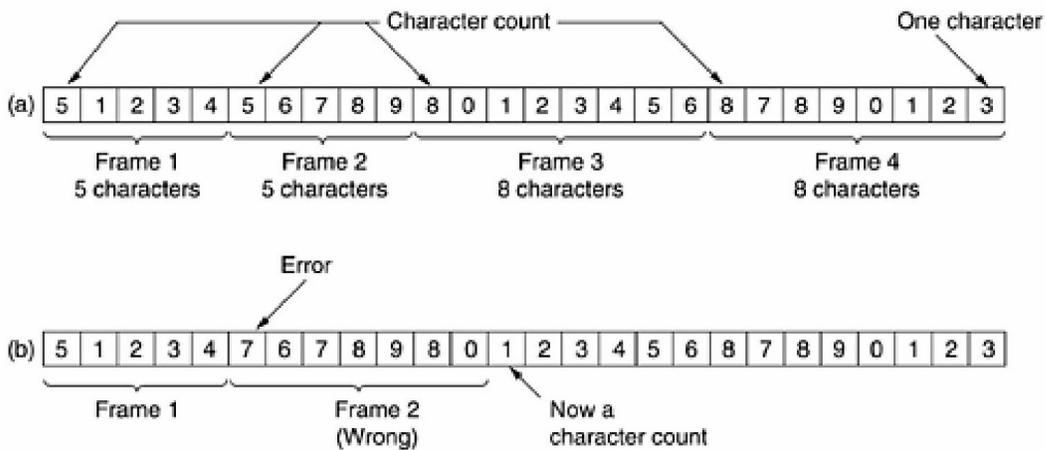
Since the physical layer merely accepts and transmits a stream of bits without any regard to meaning or structure, it is up to the data link layer to create and recognize frame boundaries. This can be accomplished by attaching special bit patterns to the beginning and end of the frame. If these bit patterns can accidentally occur in data, special care must be taken to make sure these patterns are not incorrectly interpreted as frame delimiters. The four framing methods that are widely used are

- Character count
- Starting and ending characters, with character stuffing
- Starting and ending flags, with bit stuffing
- Physical layer coding violations

#### Overview

*Character counting* simply notes the count of remaining characters in the frame's header. This method, however, is easily disturbed if this field gets faulty in some way, thus making it hard to keep up synchronization.

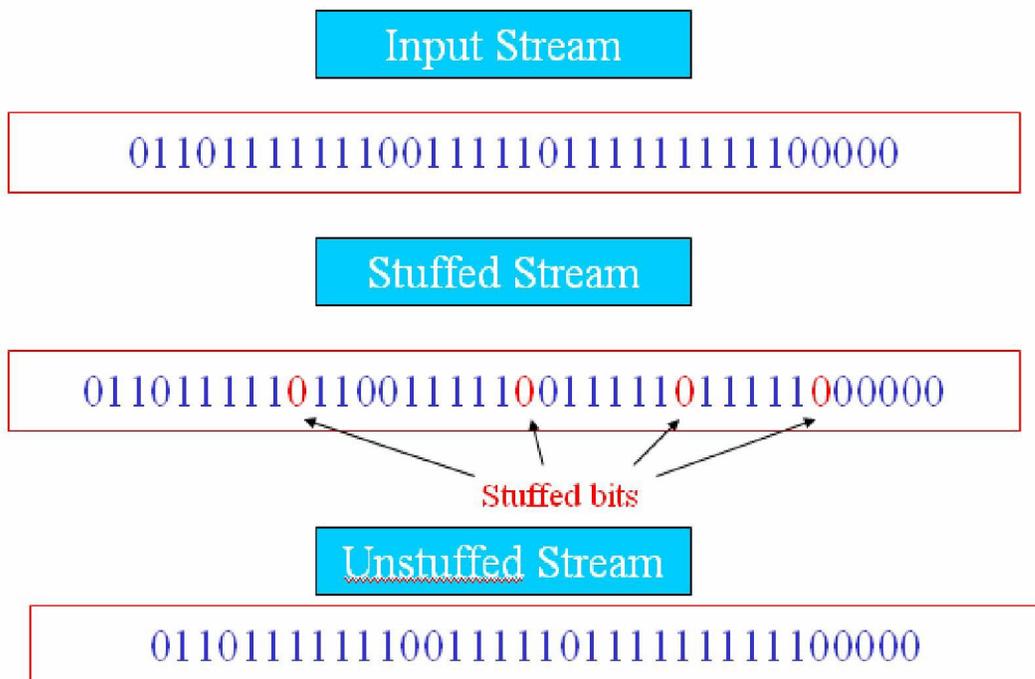
This method uses a field in the header to specify the number of characters in the frame. When the data link layer at the destination sees the character count, it knows how many characters follow, and hence where the end of the frame is. The disadvantage is that if the count is garbled by a transmission error, the destination will lose synchronization and will be unable to locate the start of the next frame. So, this method is rarely used.



A character stream. (a) Without errors. (b) With one error.

### Bit stuffing

The third method allows data frames to contain an arbitrary number of bits and allows character codes with an arbitrary number of bits per character. At the start and end of each frame is a flag byte consisting of the special bit pattern 01111110. Whenever the sender's data link layer encounters five consecutive 1s in the data, it automatically stuffs a zero bit into the outgoing bit stream. This technique is called bit stuffing. When the receiver sees five consecutive 1s in the incoming data stream, followed by a zero bit, it automatically destuffs the 0 bit. The boundary between two frames can be determined by locating the flag pattern.



## Procedure

### 1. Implement the data link layer framing method such as character count

**Include header files `stdio.h`, `string.h`**

**Declare two dimensional array with name “char” with 10 rows and 20 cols.**

```
int get_input();
void make_frames(int);
int count_chars(int s);

void main()
{
int no_of_words=get_input();
make_frames(no_of_words);
}

int get_input()
{
int answer;
int i=0;
do{
printf("\nEnter the Word:");
scanf("%s",input[i]);
fflush(stdin);
printf("\nDo you want to continue: (y: 1/n: 0)?:");
scanf("%d",&answer);
i++;
}while(answer!=0);
return i;
}
void make_frames(int num_words)
{
int i=0;
write the output statement to print “The “

for(i=0;i<num_words;i++)
printf("%d%s", (count_chars(i)+1),input[i]);
printf("\n\n");

}
int count_chars(int index)
```

```
{
int i=0;
while(input[index][i]!='\0')
i++;
return i;
}
```

## 2. Implement the data link layer framing method such as Bit Stuffing.

**Include header files stdio.h, string.h**

**Declare two dimensional array with name data "char" with 4 rows and 100 cols.**

```
#define DELIM_BIT_PATTERN "01111110"
#define SNDR_INPUT 0
#define SNDR_OUTPUT 1
#define REC_INPUT 2
#define REC_OUTPUT 3
```

**Declare two dimensional array**

```
int valid_data(void);
void sender_bit_stuff(void);
void receiver_process_data(void);
```

```
int main()
{
int ans;
do{
printf("\nEnter Data from Network Layer in Binary Form:");
scanf("%s",data[SNDR_INPUT]);
if(!valid_data())
continue;
sender_bit_stuff();
printf("\nSends Physical Layer Data:%s\n",data[SNDR_OUTPUT]);
strcpy(data[REC_INPUT],data[SNDR_OUTPUT]);
receiver_process_data();
printf("\nReceiver's Network Layer Data: %s\n",data[REC_OUTPUT]);
printf("\n\nDo you want to continue?(y: 1/n: 0)");
scanf("%d",&ans);
}while(ans!=0);
}
```

```
int valid_data()
{
```

```

char *p=data[SNDR_INPUT];
if(*p=='\0')
{
printf("\n***Enter Some DATA***\n");
return 0;
}
while(*p!='\0')
{
if(*p!='1' && *p!='0')
{
printf("*** this is not binary data. please Enter 0's and 1's\n");
}
p++;
}
return 1;
}

```

```

void sender_bit_stuff(void)
{
char *src=data[SNDR_INPUT];
char *dst=data[SNDR_OUTPUT];
int count=0;
strcpy(dst,DELIM_BIT_PATTERN);
dst+=strlen(DELIM_BIT_PATTERN);
while(*src!='\0')
{
if(count==5){
*dst='0';
dst+=1;
count=0;
}
if(*src=='1')
count++;
else
count=0;
*dst++=*src++;
}
if(*src=='\0' && count==5){
*dst='0';
dst+=1;
}
strcpy(dst,DELIM_BIT_PATTERN);
dst+=strlen(DELIM_BIT_PATTERN);
*dst='\0';
}

```

```

void receiver_process_data(void)
{

```

```
char *src=data[REC_INPUT];
char *dst=data[REC_OUTPUT];
char *end;
int count=0;
src+=strlen(DELIM_BIT_PATTERN);
end=data[REC_INPUT]+strlen(data[REC_INPUT])-strlen(DELIM_BIT_PATTERN);
while(src<=end)
{
if(count==5)
src+=1;
count=0;
if(*src=='1')
count++;
else
count=0;
*dst++=*src++;
}
*(dst-1)=='0';
return;
}
```

## Output:

### 1.Out Put:

```
Enter the Word:cat
Do you want to continue: (y: 1/n: 0)?:1
Enter the Word:dog
Do you want to continue: (y: 1/n: 0)?:1
Enter the Word:apple
Do you want to continue: (y: 1/n: 0)?:0
The Transmitted Data is:
4cat4dog6apple
```

### 2.Out put:

```
Enter Data from Network Layer in Binary Form: 0111111111110
Senders Physical Layer Data: 01111110011111011111011001111110
Receiver's Network Layer Data: 0111110111110110
Do you want to continue? (y: 1/n: 0)0
```

### **Viva-Voice Questions**

1. What are the data link layer framing methods?
2. Properties of data link layer
3. What is character stuffing
4. What is character count
5. What are advantages and disadvantages of character count
6. What are the data link layer framing methods?
7. Properties of data link layer?
8. What is Bit stuffing?
9. What is Example for Bit Stuffing?
10. What are advantages and disadvantages of Bit Stuffing?

### **Lab Report**

After successful completion of this lab, Student should be able to know about character count and bit stuffing methods.