



DEPARTMENT OF  
COMPUTER SCIENCE AND ENGINEERING

---

## Title: Implementing character Stuffing and De-stuffing

---

DATA COMMUNICATION LAB  
CSE 308



GREEN UNIVERSITY OF BANGLADESH

## 1 Objective(s)

- To implement the data link layer framing method Character stuffing.

## 2 Problem analysis

A technique where each frame starts with the ASCII character sequence DLE STX and ends with the sequence DLE ETX. (where DLE is Data Link Escape, STX is Start of TeXt and ETX is End of TeXt.) This method overcomes the drawbacks of the character count method. If the destination ever loses synchronization, it only has to look for DLE STX and DLE ETX characters. If however, binary data is being transmitted then there exists a possibility of the characters DLE STX and DLE ETX occurring in the data. Since this can interfere with the framing, a technique called character stuffing is used.

In character or byte stuffing; within the frame, replace every occurrence of DLE with the two-character sequence DLE DLE. The receiver reverses the processes, replacing every occurrence of DLE DLE with a single DLE before this data is given to the network layer.

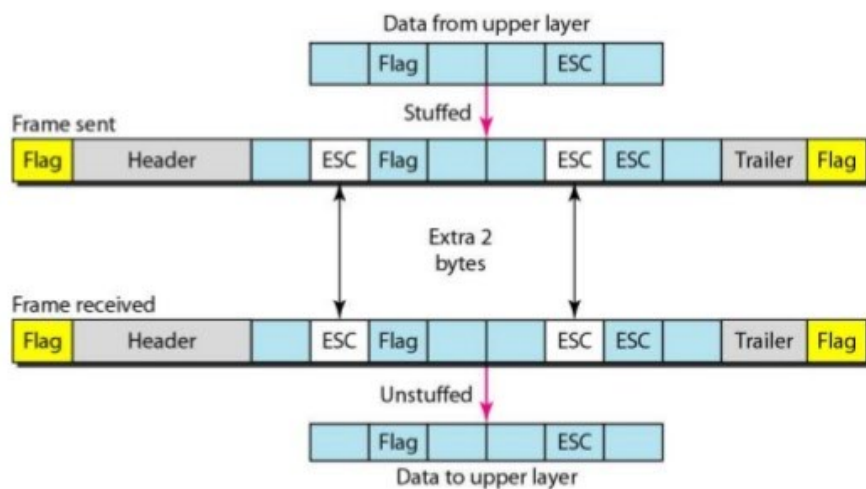


Figure 1: Character stuffing and de-stuffing in HDLC frame structure.

## 3 Algorithm

---

**Algorithm 1:** Character stuffing and de-stuffing

---

**Input:** Given a string of size N, flag character, escape character

- To stuff replace every occurrence of DLE with the two-character sequence DLE DLE.
  - replace every occurrence of DLE DLE with a single DLE in de-stuffing.
- 

## 4 Implementation in C

```
1 #include<stdio.h>
2 #include<string.h>
3 int main()
4 {
5     int i=0, j=0;
6     char d[100], l[]="DLEETX", sd[100], ds[100];
7     printf("Enter Data in UpperCase: ");
8     scanf("%s", d);
9     sd[0]='D', sd[1]='L', sd[2]='E',
```

```

10 sd[3]='S',sd[4]='T',sd[5]='X';
11 j=6;
12 while(d[i]!='\0')
13 {
14     if(d[i]=='D' && d[i+1]=='L' && d[i+2]=='E')
15     {
16         sd[j]='D', sd[j+1]='L', sd[j+2]='E', sd[j+3]='D',
17         sd[j+4]='L', sd[j+5]='E';
18         j+=6;
19         i+=3;
20     }
21     else
22         sd[j++]=d[i++];
23 }
24 sd[j]='\0';
25 strcpy(ds,sd);
26 strcat(sd,l);
27 printf("After Stuffing: ");
28 printf("%s",sd);
29 i=0;
30 j=6;
31 while(ds[j]!='\0')
32 {
33     if( ds[j]=='D' && ds[j+1]=='L' && ds[j+2]=='E' &&
34     ds[j+3]=='D' && ds[j+4]=='L' && ds[j+5]=='E')
35     {
36         d[i]='D', d[i+1]='L',d[i+2]='E';
37         j+=6;
38         i+=3;
39     }
40     else
41         d[i++]=ds[j++];
42 }
43 d[i]='\0';
44 printf("\n\nAfter De-stuffing: ");
45 printf("%s",d);
46 }

```

## 5 Input/Output (Compilation,Debugging & Testing)

**Enter Data in UpperCase: UNITEDLEDSTATES**

**After Stuffing: DLESTX UNITEDLEDLEDSTATES DLEETX**

**After De-stuffing: DLESTX UNITEDLEDSTATES DLEETX**

## 6 Discussion & Conclusion

Based on the focused objective(s) to understand about character stuffing and de-stuffing, the additional lab exercise made me more confident towards the fulfilment of the objectives(s).

## 7 Lab Task (Please implement yourself and show the output to the instructor)

- Implement the Character stuffing and de-stuffing together where the system provides a choice to change the transmitted bit stream before de-stuffing. Consider the followings:

A B C D E F F E F D F E G

Where, F is the flag character and E is the escape character.

## 8 Lab Exercise (Submit as a report)

- Complete the given lab task and submit as a report.

## 9 Policy

Copying from internet, classmate, seniors, or from any other source is strongly prohibited. 100% marks will be *deducted* if any such copying is detected.