**Функция добавление элемента** symble **в** placeToPaste **место последовательности** str**.**

void COM::paste(uint8\_t\* str, char symble, int placeToPaste, int& last) {

for (int i = last + 1; i>placeToPaste; --i) {

str[i] = str[i - 1];

}

last++;

str[placeToPaste] = symble;

}

**Функция преобразования передаваемой информации в пакет и его передача.**

void COM::WritePackage(const uint8\_t\* data) {

uint8\_t temp[200];

int i;

for (i = 0; data[i] != '\0'; ++i) {

temp[i] = data[i];

}

temp[i] = '\0';

int last = ++i;

temp[last] = flag;

paste(temp, flag, 0, last);

for (i = 1; i < last - 1; ++i) {

if (temp[i] == changeSymbol) {

paste(temp, checkUnchange, i + 1, last);

i++;

}

if (temp[i] == flag) {

temp[i] = changeSymbol;

paste(temp, checkChange, i + 1, last);

i++;

}

}

for (i = 0; i <= last; ++i) {

Write((uint8\_t\*)&temp[i], 1);

}

}

**Функция удаления элемента из места** elementToCut **последовательности** str**.**

void COM::cut(uint8\_t\* str, int elementToCut, int&last) {

for (int i = elementToCut; i < last; ++i) {

str[i] = str[i + 1];

}

last--;

}

**Функция получения пакета и его преобразования в исходный вид.**

void COM::ReadPackage(uint8\_t\* str) {

uint8\_t temp = 'a';

while (temp != flag) {

Read(&temp, 1);

}

temp = 'a';

int i = 0;

for (i = 0; temp != flag; ++i) {

Read(&temp, 1);

if (temp == flag) {

break;

}

str[i] = temp;

}

int last = i;

for (i = 0; i <= last; ++i) {

if (str[i] == changeSymbol) {

if (str[i + 1] == checkChange) {

str[i] = flag;

cut(str, i + 1, last);

}

else {

cut(str, i + 1, last);

}

}

}

}