قسمت Reciever :

```
#define F_CPU 8000000UL
#include <avr/io.h>
#include <string.h>
#include <avr/delay.h>
#define LCD_Data_Dir DDRB
#define LCD_Command_Dir DDRA
#define LCD_Data_Port PORTB
#define LCD_Command_Port PORTA
#define RS PC0
#define RW PC1
#define EN PC2

void LCD_Command(unsigned char cmnd)
{
    LCD_Data_Port = cmnd;
    LCD_Command_Port &= ~(1 << RS);
    LCD_Command_Port &= ~(1 << RW);
    LCD_Command_Port |= (1 << EN);
    LCD_Command_Port &= ~(1 << EN);
    _delay_ms(3);
}

void LCD_Init(void)
{
    LCD_Command_Dir = 0xFF;
    LCD_Data_Dir = 0xFF;
    _delay_ms(20);
    LCD_Command(0x38);
    LCD_Command(0x0C);
    LCD_Command(0x06);
    LCD_Command(0x01);
    LCD_Command(0x80);
}
void LCD_Char(unsigned char char_data)
{
    LCD_Data_Port = char_data;
    LCD_Command_Port |= (1 << RS);
    LCD_Command_Port &= ~(1 << RW);
    LCD_Command_Port |= (1 << EN);
    LCD_Command_Port &= ~(1 << EN);
}
void LCD_String(char *str)
{
    for (int i = 0; str[i] != 0; i++)
        LCD_Char(str[i]);
}


char content[200];
int main()
{
    DDRC = 0;
    _delay_ms(100);
    while (PINC == 0xFF){}
    for (int i = 0; i != 200; i++)
    {
        content[i] = PINC;
        _delay_ms(1);
    }
    LCD_Init();
    int lce_length = 16;
    for (int i = 0; i != 200 - lce_length; i++)
    {
        char lcd_text[lce_length];
        strncpy(lcd_text, content + i, lce_length);
        LCD_Command(0x01);
        LCD_Command(0x80);
        LCD_String(lcd_text);
        _delay_ms(10);
    }
    return 0;
}
```

قسمت Sender :

```
#define F_CPU 8000000UL /* Define CPU Frequency e.g. here 8MHz */
#include <string.h>
#include <avr/io.h>
#include <avr/delay.h>
#include <avr/eeprom.h>

char data[200], content[200] = "Hello my name is Bardia.Hello my name is
Bardia.Hello my name is Bardia.Hello my name is Bardia.Hello my name is
Bardia.Hello my name is Bardia.Hello my name is Bardia.Hello my name is Bardia.";

int main()
{
    DDRB = 0xFF;
    PORTB = 0xFF;
    eeprom_busy_wait();
    eeprom_write_block(content, 0, strlen(content)); //Write the content to EEPROM
    eeprom_read_block(data,0,strlen(content)); // Read the content from EEPROM
    _delay_ms(100);
    for (int i = 0; i != 200; i++)
    {
        PORTB = data[i];
        _delay_ms(1);
    }
    PORTB = 0;
    return 0;
}
```