**MINISTERUL EDUCAȚIEI ȘI CERCETĂRII AL REPUBLICII MOLDOVA**

**Universitatea Tehnică a Moldovei**

**RAPORT**

Lucrare de laborator nr. 4

la cursul ***„Interfețe de comunicare”***

**Tema: Cercetarea interfetelor standard de comunicatii in cod serie. Regim Master.**

**A efectuat :**  **St. gr. CR-221FR Serba Cristina**

**A verificat:**  **Ababii Victor**

**Chișinău 2024**

# Scopul lucrării:

Cercetarea, proiectarea și programarea interfețelor standard de comunicare in cod serie (COM, RS, UART, USART, I2C, SPI ). Regim Master. Proiectarea în mediul Proteus sau Fritzing a unui sistem de transfer date în cod serie în baza Kit-ului Arduino UNO. Datele sunt recepționate în cod eserie de la Modulul Terminal si afisate pe Modulul Osciloscop. Datele receptionate sunt afișate pe un

indicator LCD 2\*16 pozitii LM016L.

# Sarcina tehnică:

1. Să se instaleze mediul de dezvoltare Arduino IDE și bibliotecile respective;

2. Să se instaleze mediul de dezvoltare Proteus și bibliotecile respective;

3. Să se asambleze schema electrică de principiu (Proteus) a sistemului pentru achiziția datelor în cod serie prin portul UART (TxD – RxD) de la Modulul Terminal, afișarea disgramei de timp pe Modulul Osciloscop și afișarea datelor pe un Display LCD 2\*16 LM016L;

4. Să se elaboreze programul (Arduino IDE) care efectuează achiziția datelor din portul UART și memorizarea acestora într-o variabilă String;

5. Să se elaboreze programul care afișează datele primite de la Modulul Terminal pe un ecran LCD LM016L;

6. Să se incarce codul HEX al programului elaborat în mediul Proteus pentru testare funcțională;

7. Procesul de simulare sa se înregistreze prin PrintScren-uri.

# Schema electrică de principiu:

# Descrierea sarcinii în parametri tehnici:

Sarcina presupune conectarea terminalului pentru a primi date. Acestea vor fi stocate într-o variabilă string, apoi transmise modulului LCD LM016L. De asemenea, este conectat un osciloscop la ieșirea și intrarea datelor pentru a monitoriza semnalele. Datele sunt transmise prin interfața UART în mod Master.

# Algoritmul de funcționare al sistemului

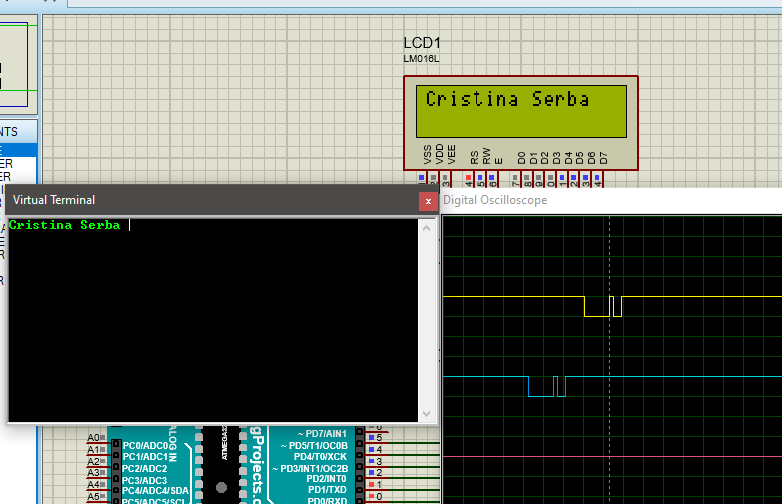
Pentru a opera circuitul în regim master utilizând interfața UART de comunicare, am conectat pin-ul PD0/Rx al plăcii Arduino cu cel Tx al terminalului virtual, iar PD1/Tx al Arduino – cu Rx al terminalului. Display-ul LCD a fost conectat la restul pinilor, maparea fiind specificată în codul sursă. La conexiunea UART dintre Arduino și terminalul virtual am conectat și un osciloscop pentru monitorizarea semnalelor. Modulul LCD a fost utilizat fără reglarea intensității de iluminare, deci pinii VSS și RW au fost conectați la ground.

Rezultatul executării programului a fost încărcat în proprietățile plăcii Arduino.

# Codul sursă C și HEX ale programelor elaborate.

|  |
| --- |
| #include <LiquidCrystal.h>  /\* Initializarea LCD-ului cu pinii corespunzatori conectati la microcontroller \*/  const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;  LiquidCrystal lcd(rs, en, d4, d5, d6, d7);  int charactersDisplayed = 0; /\* folosita ulterior pentru a determina cate caractere sunt afisate \*/  bool firstRowCompleted = false; /\* folosita pentru a determina pe al catalea rand se scrie \*/  void setup()  {    /\* LCD 16x2 \*/    lcd.begin(16, 2);    /\* Terminal \*/    Serial.begin(9600);  }  void loop()  {    while (Serial.available() == 0)    {      /\* permite programului sa astepte date de la terminal \*/    }      String data = "";    while (Serial.available() > 0)    {      /\* Daca sunt date disponibile, umple buffer-ul data \*/      char receivedChar = Serial.read();      data += receivedChar;    }      if (charactersDisplayed + data.length() > 16)    {      /\* Primul rand este plin, va incepe pe al doilea \*/      if (false == firstRowCompleted)      {        firstRowCompleted = true;        lcd.setCursor(0, 1);      }      /\* Al doilea rand este plin, va reseta tot \*/      if (charactersDisplayed + data.length() > 32)      {        lcd.clear();        charactersDisplayed = 0;        firstRowCompleted = false;      }    }      lcd.print(data);    Serial.print(data);      /\* Actualizeaza nr total de caractere afisate \*/    charactersDisplayed += data.length();  } |

# Graficele obţinute în rezultatul simulărilor / măsurărilor efectuate.



# Concluzii

În urma efectuării lucrării de laborator date, care avea ca scop familiarizarea cu interfața de comunicare UART în modul Master și modulul LCD 16x2, am reușit să conectez cu succes o un Display LCD LM016L, un microcontroller de tip Arduino Uno și terminalul virtual. În acest sens, am colectat date în mod dinamic de la tastatură, apoi le-am afișat pe display și terminal folosind interfața UART.   
În concluzie, am acumulat cunoștințe noi despre comunicarea în mod master și am reușit să integrez propriul circuit precum și am adâncit cunoștintele despre modul de lucru în Proteus și Arduino IDE. De asemenea, am putut monitoriza semnalele de intrare și ieșire prin conectarea unui osciloscop.