**MINISTERUL EDUCAȚIEI ȘI CERCETĂRII AL REPUBLICII MOLDOVA**

**Universitatea Tehnică a Moldovei**

**RAPORT**

Lucrare de laborator nr. 3

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

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

**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 Slave. Proiectarea în mediul Proteus sau Fritzing a unui sistem de transfer date în cod serie în baza Kit-ului Arduino UNO. Datele sunt introduse de la tastatura KBD cu 4x4 taste. Vizualizarea datelor are loc pe osciloscopul Terminal.

# 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 de la KBD, transmiterea acestora în cod serie prin portul UART (TxD – RxD) și afișarea diagramei de timp în fereastra de dialog Terminal;

4. Să se elaboreze programul (Arduino IDE) care efectuează achiziția datelor de la KBD cu 16 taste cu scanare în regim dinamic;

5. Să se elaboreze programul care transmite datele în portul UART în cod serie;

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

7. Procesul de transmitere a datelor să se analizeze în baza elementului Terminal.

# Schema electrică de principiu:

# Descrierea sarcinii în parametri tehnici:

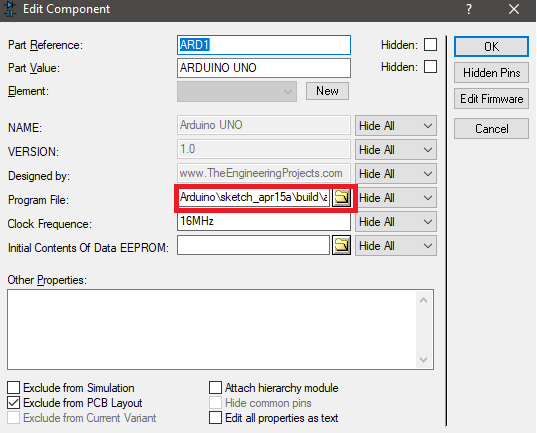
Sarcina presupune conectarea a 3 dispozitive: keypad 4x4, Arduino UNO și terminalul virtual. Pentru transmiterea datelor între microcontroller și terminal este folosită interfața de comunicare UART. Datele vor fi colectate în mod dinamic, în interiorul unei bucle infinite. Aceasta va verifica daca au fost primite date de la keypad, dupa care le va afișa în terminalul virtual.

# Algoritmul de funcționare al sistemului

Pentru a opera circuitul în regim slave 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. Restul pinilor fie n-au fost folosiți, fie au fost conectați cu cei ai keypad-ului 4x4, apoi mapați corespunzător în codul sursă.

Pentru maparea pinilor folosiți la comunicarea cu tastatura am folosit biblioteca Keypad.h din domeniul de proiectare Arduino.

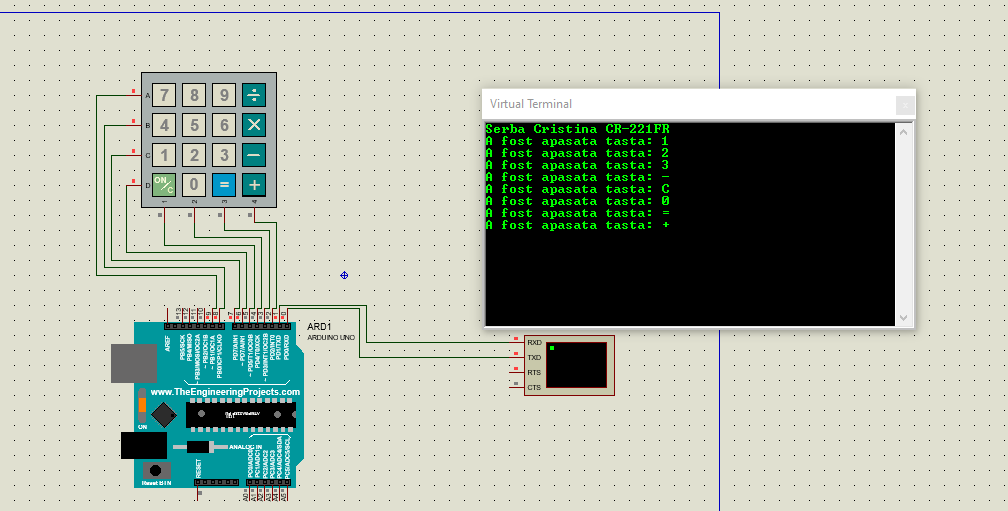
Rezultatul executării programului a fost încărcat în proprietățile plăcii Arduino



# Codul sursă C și HEX ale programelor elaborate.

|  |
| --- |
| #include <Keypad.h>  #define ROWS  4  #define COLS  4  /\* Structura keypad-ului 4x4 \*/  char keys[ROWS][COLS] = {    {'7','8','9','/'},    {'4','5','6','\*'},    {'1','2','3','-'},    {'C','0','=','+'}  };  /\* Maparea la pinii arduino \*/  byte rowPins[ROWS] = {9, 8, 7, 6};  byte colPins[COLS] = {5, 4, 3, 2};  Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );  void setup() {    Serial.begin(9600);    Serial.println("Serba Cristina CR-221FR");  }  void loop()  {    char key = keypad.getKey(); /\* Primeste date de la tastatura \*/    if (key != NO\_KEY)    {      /\* Verifica daca datele primite nu sunt goale \*/      Serial.print("A fost apasata tasta: ");      Serial.println(key);    }  } |

# 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 și mediul de proiectare Proteus, am reușit să conectez cu succes o tastatură 4x4, 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 terminal folosind interfața UART.   
În concluzie, am acumulat cunoștințe noi despre comunicarea în mod slave și am reușit să integrez propriul circuit precum și am adâncit cunoștintele despre modul de lucru în Proteus și Arduino IDE.