

ПРАКТИЧНЕ ЗАНЯТТЯ №2

Організація виведення та візуалізації вимірювальної інформації кириличними символами на платформі Arduino Uno

1.1 Мета роботи

Вивчити схематехніку підключення технічних засобів візуалізації вимірювальної інформації та програмування процедур її виведення на зовнішні пристрої кириличними символами

1.2. Короткі теоретичні відомості

В практичній роботі для виведення та візуалізації вимірювальної інформації на LCD кириличними символами можливо декількома варіантами:

- Підключення бібліотеки для кириличних символів;
- Створення власних символів.

Враховуючи ті фактори, що знайти робочу бібліотеку складно та для відображення інформації зазвичай потрібно вивести одне повідомлення, то оптимальним рішенням буде створити власні символи для виведення.

Для цього нам потрібно використовувати в скетчі змінну *byte* і *функцію* `lcd.createChar ()`. Відразу відзначимо, що обсяг пам'яті на Ардуіно для знаків обмежений

Вивести свій символ або кириличну букв.у на дисплей допоможе таблиця знакогенератора (CGRAM). Такий вид пам'яті в Ардуіно, як CGRAM, може зберігати власні символи, але розмір пам'яті обмежений і може вмістити лише 8 власних символів. Це зазвичай несуттєвий недолік, тому що кириличні символи на 80% збігаються з латинськими. Тому, потрібно прописати тільки не існуючі символи, наприклад, літера велика Б та маленька б.

Щоб отримати свій символ для початку візьміть аркуш паперу і намалуйте на ньому таблицю, де буде 5 стовпців і 8 рядків. Далі заштрихуйте в таблиці клітини (рис.2.1), які повинні висвітлюватися на дисплеї. Справа в тому, що кожен символ на дисплеї складається з пікселів (5 пікселів в ширину і 8 пікселів у висоту). Далі представимо наш символ в вигляді масиву даних, що складається з восьми елементів - восьми рядків.

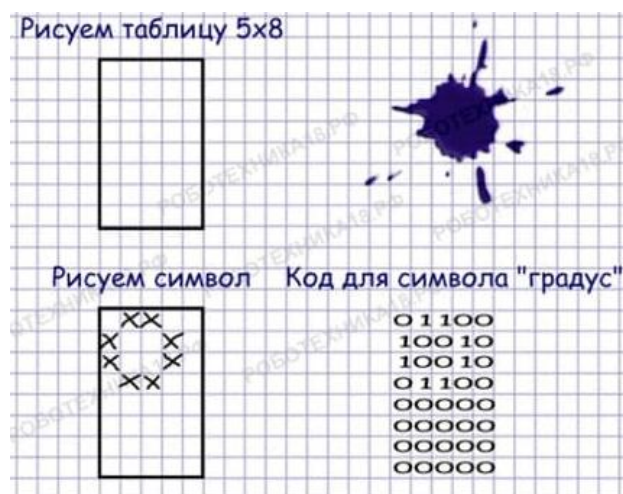


Рисунок 2.1

Підключення дисплею в даній практичній роботі по послідовному інтерфейсу I2C (схема підключення рис.2.2);

- передача даних 4-х розрядним паралельним цифровим кодом (схема підключення розглянута в лабораторній роботі №2).

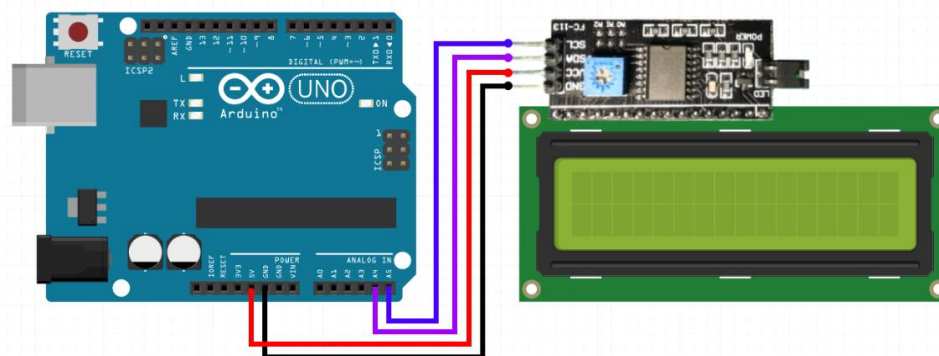


Рисунок 2.2 – Схема підключення LCD дисплея до Arduino по послідовному інтерфейсу I2C

Для роботи з LCD дисплеєм згідно першого підключення вам потрібно буде в скетчі підключити стандартну бібліотеку Wire.h та інтегрувати в середовище Arduino IDE бібліотеку LiquidCrystal_I2C.h, потім також її підключити. Приклад скетчу з підключеними бібліотеками рис.2.3.

```
sketch_lr.2.1 | Arduino 1.8.10
Файл Правка Скетч Инструменты Помощь
sketch_lr.2.1
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,20,4); // Объявляем объект библиотечной функции
// Если надпись не появилась

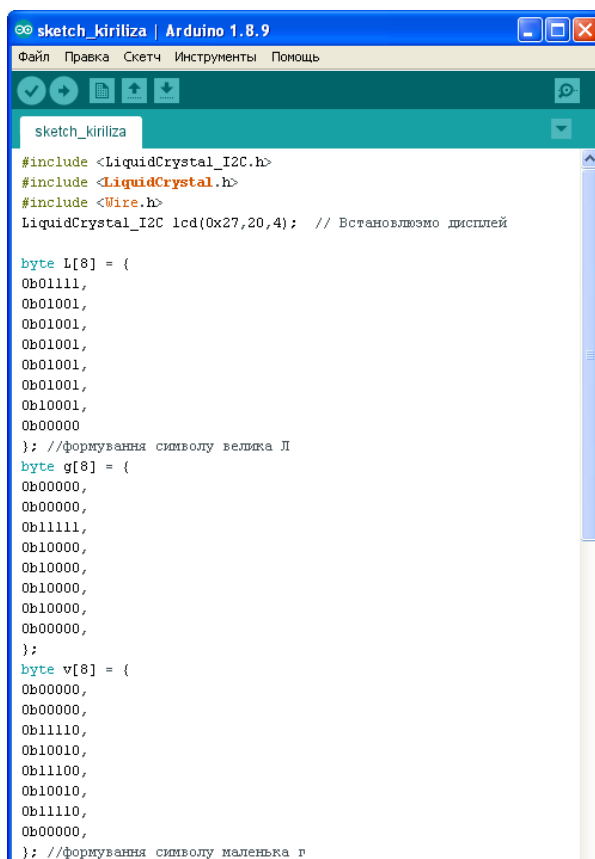
void setup() {
  // put your setup code here, to run once:
  lcd.init(); // Инициализация lcd
  lcd.backlight(); // Включаем подсветку
  // Курсор находится в на 4 знаке 1 строки
  lcd.setCursor(3, 0);
  lcd.print("Hello, world!");
}

void loop() {
  // put your main code here, to run repeatedly:
}

Загрузка завершена.
Скетч использует 2698 байт (8%) памяти устройства. Всего доступно 32768 байт.
Глобальные переменные используют 257 байт (12%) динамической памяти.
11 Arduino/Genuino Uno на COM3
```

Рисунок 2.3

Приклад скетчу для створення символу на LCD I2C рис.2.4.



```
sketch_kiriliza | Arduino 1.8.9
Файл Правка Скетч Инструменты Помощь

sketch_kiriliza

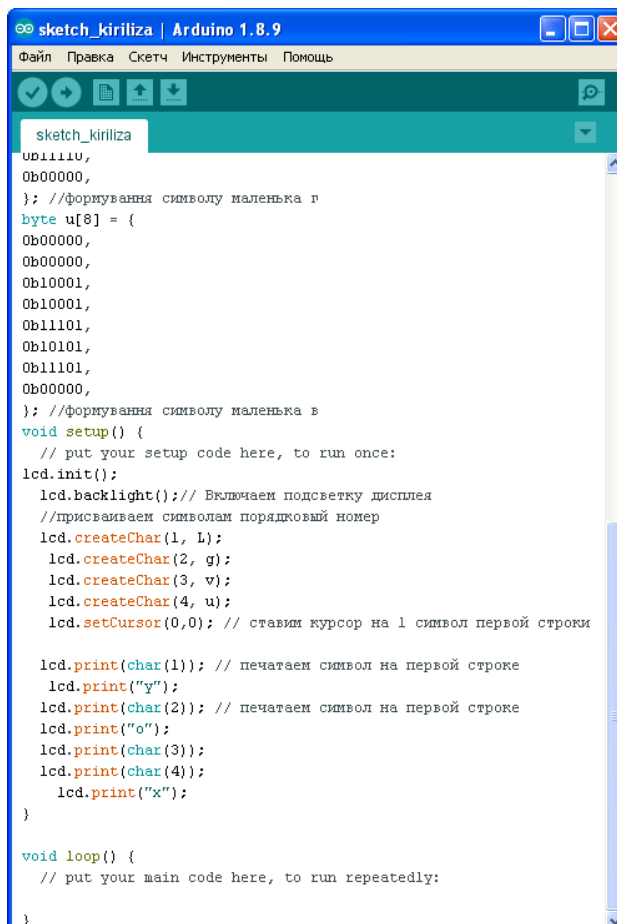
#include <LiquidCrystal_I2C.h>
#include <LiquidCrystal.h>
#include <Wire.h>
LiquidCrystal_I2C lcd(0x27,20,4); // Встановлюємо дисплей

byte L[8] = {
  0b01111,
  0b01001,
  0b01001,
  0b01001,
  0b01001,
  0b01001,
  0b10001,
  0b00000
}; //формування символу велика Л

byte g[8] = {
  0b00000,
  0b00000,
  0b11111,
  0b10000,
  0b10000,
  0b10000,
  0b10000,
  0b00000
};

byte v[8] = {
  0b00000,
  0b00000,
  0b11110,
  0b10010,
  0b11100,
  0b10010,
  0b11110,
  0b00000
}; //формування символу маленька л
```

Рисунок 2.4



```
sketch_kiriliza | Arduino 1.8.9
Файл Правка Скетч Инструменты Помощь

sketch_kiriliza

0b11110,
0b00000,
}; //формування символу маленька л
byte u[8] = {
  0b00000,
  0b00000,
  0b10001,
  0b10001,
  0b11101,
  0b10101,
  0b11101,
  0b00000,
}; //формування символу маленька в

void setup() {
  // put your setup code here, to run once:
  lcd.init();
  lcd.backlight();// Включаем подсветку дисплея
  //присваиваем символам порядковый номер
  lcd.createChar(1, L);
  lcd.createChar(2, g);
  lcd.createChar(3, v);
  lcd.createChar(4, u);
  lcd.setCursor(0,0); // ставим курсор на 1 символ первой строки

  lcd.print(char(1)); // печатаем символ на первой строке
  lcd.print("л");
  lcd.print(char(2)); // печатаем символ на первой строке
  lcd.print("о");
  lcd.print(char(3));
  lcd.print(char(4));
  lcd.print("х");
}

void loop() {
  // put your main code here, to run repeatedly:
}
}
```

Рисунок 2.5

1.3. Підготовка до роботи

При підготовці до роботи необхідно:

- ознайомитись з рекомендованою літературою;
- вивчити короткі теоретичні відомості.

1.4. Порядок роботи:

- Підключити модуль Arduino Uno до LCD дисплея по послідовному інтерфейсу I2C .
- Скачати та підключити в скетчі необхідні бібліотеки для роботи з LCD дисплеєм.
- Вивести повідомлення “Hello world” для того щоб перевірити працездатність зібраного макету.
- Вивести своє прізвище на українській та російській мові позиція задана згідно таблиці 2.1.
- Оформити звіт та зробити висновки.

Таблиця 2.1

№ варіанту	Ряд	Знакомісце
1	2	3
1	1	0
2	2	1
3	3	2
4	4	4
5	1	6
6	2	8
7	3	10
8	4	1
9	1	3
10	2	5
11	3	7
12	4	9
13	1	0
14	2	1
15	3	2
16	4	4
17	1	6
18	2	8
19	3	10
20	4	1