

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**

Інститут **КНІТ**

Кафедра **ПЗ**

**Лабораторна робота № 1**

**На тему:** *“Дослідження рідкокристалічного дисплею **LCD1602** за допомогою середовища **STM32CubeIDE** та плати **STM32F4 Discovery**”*

**З дисципліни:** *“Основи інтернету речей”*

**Львів – 2021**

**Тема роботи:** Дослідження рідкокристалічного дисплею **LCD1602** за допомогою середовища **STM32CubeIDE** та плати STM32F4 Discovery.

**Мета роботи:** Дослідити роботу пристроїв відображення інформації на прикладі символьного рідкокристалічного дисплею .

### **Теоретичні відомості**

Символьний дисплей LCD1602 – рідкокристалічний дисплей (Liquid Crystal Display) екран якого здатний відображати одночасно до 32 символів (16 стовпців, 2 рядки). Символьний дисплей побудований на базі ЖК дисплея типу STN (Super Twisted Nematic) під керуванням контролера HD44780 і має синхронний паралельний 8-бітний інтерфейс. Дисплей оснащений світлодіодним підсвічуванням і здатний одночасно відображати до 32 символів (16 стовпців, 2 рядки) від чого і пішла назва дисплея: LCD1602. Контролер HD44780 має ПЗУ в якій зберігаються цифри, символи латиниці і деякі ієрогліфи японської мови, для їх відображення на дисплеї. Відсутні символи, в т.ч. і символи кирилиці, можна завантажувати в пам'ять ОЗУ контролера, для виведення на дисплей написів українською мовою або нестандартних символів (наприклад «смайликів»).

Підключення дисплея LCD1602 можливе по паралельному інтерфейсу та через конвертер по шині I2C. Шина даних дисплея складається з 8 ліній (D0-D7), але можливо підключити тільки старші 4 лінії (D4-D7). Це не знижує швидкість роботи дисплея. В табл.1 наведені призначення контактів дисплею.

### **Програмування дисплею**

Для роботи з дисплеєм використаємо 4ох бітний інтерфейс. Створимо проект, в налаштуваннях проекту необхідно налаштувати 7 пінів на вихід. В даному прикладі використаємо піни PE7, 10-15 (Рис.1).

Підключимо в наступній послідовності:

Rs Pin	GPIOE Pin 7
Rw Pin	GPIOE Pin 10
En Pin	GPIOE Pin 11
dh4 Pin	GPIOE Pin 12
dh5 Pin	GPIOE Pin 13
dh6 Pin	GPIOE Pin 14
dh7 Pin	GPIOE Pin 15

Таблиця 1. Призначення контактів дисплею

№: Контакту	Назва:	Призначення:
16	<b>K</b> (LED-)	Катод (мінус) LED (світлодіодним) підсвічування
15	<b>A</b> (LED +)	Анод (плюс) LED (світлодіодним) підсвічування
14 ... 7	<b>D7 ... D0</b> (DB7 ... DB0)	Шина даних (Data Bus) складається з 8 ліній. У наведеній схемі використовуються тільки старші 4 лінії, тому що це не впливає на швидкість роботи дисплея
6	<b>E</b>	Сигнал дозволу (Enable)
5	<b>RW</b>	Вибір напрямку (Read / Write) передачі даних: «1» - читання з дисплея / «0» - запис в дисплей
4	<b>RS</b>	Вибір регістру (Register Selection) одержувача інформації: «1» - регістр даних / «0» - регістр інструкцій
3	<b>V0</b> (VEE)	Установка контрастності дисплея: 0 ... +5В постійного струму
2	<b>VDD</b> (VCC)	Харчування логіки дисплея: +5В постійного струму
1	<b>VSS</b> (GND)	Загальний висновок харчування (земля)

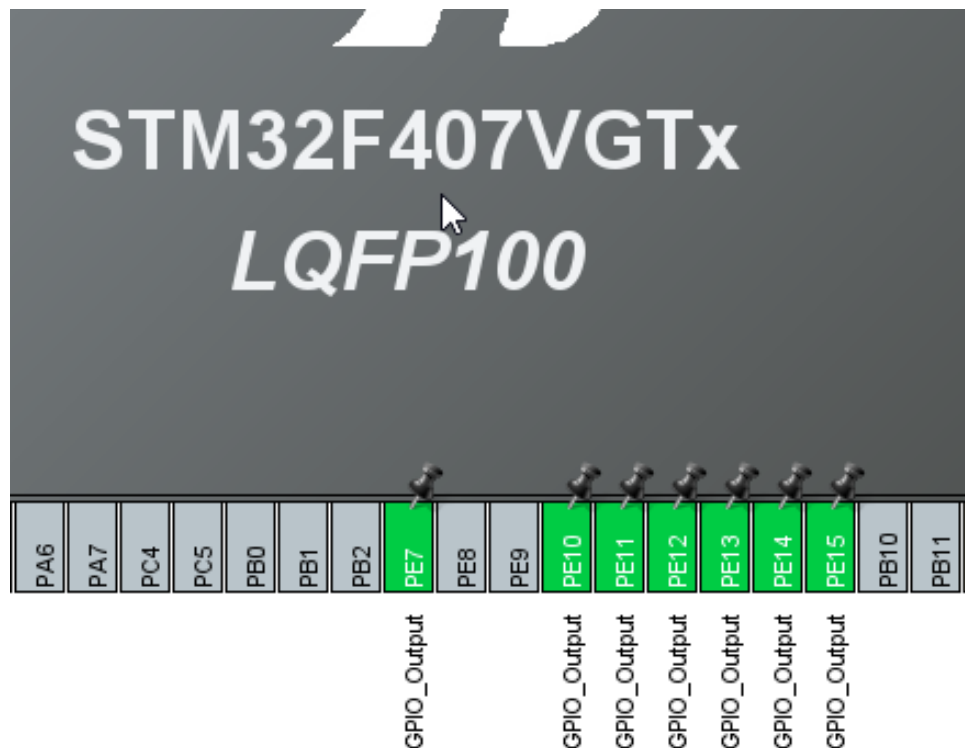


Рис.1 Графічне зображення ініціалізації наявна периферії для роботи з дисплеєм LCD1602

Створимо окремий модуль для роботи з дисплеєм. В файлі .h визначемо наступні дефайни:

```
#define lcdPort  GPIOE
#define rsPin  GPIO_PIN_7
#define rwPin  GPIO_PIN_10
#define enPin  GPIO_PIN_11
#define dh4Pin  GPIO_PIN_12
#define dh5Pin  GPIO_PIN_13
#define dh6Pin  GPIO_PIN_14
#define dh7Pin  GPIO_PIN_15
```

А також визначення для роботи з цими пінами:

```
#define RS(X) HAL_GPIO_WritePin(lcdPort, rsPin, (X))
#define RW(X) HAL_GPIO_WritePin(lcdPort, rwPin, (X))
#define EN(X) HAL_GPIO_WritePin(lcdPort, enPin, (X))
#define DH4(X) HAL_GPIO_WritePin(lcdPort, dh4Pin, (X))
#define DH5(X) HAL_GPIO_WritePin(lcdPort, dh5Pin, (X))
#define DH6(X) HAL_GPIO_WritePin(lcdPort, dh6Pin, (X))
#define DH7(X) HAL_GPIO_WritePin(lcdPort, dh7Pin, (X))
```

Для роботи з початку необхідно написати функцію відправки команди чи даних на контролер дисплею. Дані виставлені на шині DH передаються після зміни стану піна EN на 1 мілісекунду. Визначенням чи це дані, чи це команда відповідає пін RS.

```
void lcdSend(uint8_t isCommand, uint8_t data) {
    RS(isCommand!=0);
    HAL_Delay(1);
    DH7(((data >> 7) & 1)==1);
    DH6(((data >> 6) & 1)==1);
    DH5(((data >> 5) & 1)==1);
    DH4(((data >> 4) & 1)==1);
    EN(1);
    HAL_Delay (1);
    EN(0);
    DH7(((data >> 3) & 1)==1);
    DH6(((data >> 2) & 1)==1);
    DH5(((data >> 1) & 1)==1);
    DH4((data & 1)==1);
    EN(1);
    HAL_Delay (1);
    EN(0);
}
```

```

void lcdCommand(uint8_t cmd) {
    lcdSend(1, cmd);
}

void lcdChar(const char chr) {
    lcdSend(0, (uint8_t)chr);
}

```

Переглянемо список функцій (Табл.1) із документації та реалізуємо деякі із них.

Таблиця 2. Список функцій дисплею.

COMMAND	COMMAND CODE										COMMAND CODE	E-CYCLE f <sub>osc</sub> =250KHz
	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0		
SCREEN CLEAR	0	0	0	0	0	0	0	0	0	1	Screen Clear, Set AC to 0 Cursor Reposition	1.64ms
CURSOR RETURN	0	0	0	0	0	0	0	0	1	*	DDRAM AD=0, Return, Content Changeless	1.64ms
INPUT SET	0	0	0	0	0	0	0	1	I/D	S	Set moving direction of cursor, Appoint if move	40us
DISPLAY SWITCH	0	0	0	0	0	0	1	D	C	B	Set display on/off,cursor on/off, blink on/off	40us
SHIFT	0	0	0	0	0	1	S/C	R/L	*	*	Remove cursor and whole display,DDRAM changeless	40us
FUNCTION SET	0	0	0	0	1	DL	N	F	*	*	Set DL,display line,font	40us
CGRAM AD SET	0	0	0	1	ACG						Set CGRAM AD, send receive data	40us
DDRAM AD SET	0	0	1	ADD						Set DDRAM AD, send receive data	40us	
BUSY/AD READ CT	0	1	BF	AC						Executing internal function, reading AD of CT	40us	
CGRAM/ DDRAM DATA WRITE	1	0	DATA WRITE								Write data from CGRAM or DDRAM	40us
CGRAM/ DDRAM DATA READ	1	1	DATA READ								Read data from CGRAM or DDRAM	40us
	I/D=1: Increment Mode; I/D=0: Decrement Mode S=1: Shift S/C=1: Display Shift; S/C=0: Cursor Shift R/L=1: Right Shift; R/L=0: Left Shift DL=1: 8D DL=0: 4D N=1: 2R N=0: 1R F=1: 5x10 Style; F=0: 5x7 Style BF=1: Execute Internal Function; BF=0: Command Received										DDRAM: Display data RAM CGRAM: Character Generator RAM ACG: CGRAM AD ADD: DDRAM AD & Cursor AD AC: Address counter for DDRAM & CGRAM	E-cycle changing with main frequency. Example: If f <sub>cp</sub> or f <sub>osc</sub> =270KHz  40us x 250/270 =37us

```

void setCursor(uint8_t col, uint8_t row)
{
    lcdSend(1,0x80 | (col | row*0x40));
}

void paramsToggle(uint8_t display,uint8_t cursor,uint8_t blink){
    uint8_t comm = 0x08;
    if(display)comm|=0x04;
    if(cursor)comm|=0x02;
    if(blink)comm|=0x01;
    lcdCommand(comm);
}

void shiftToggle(uint8_t rightShift,uint8_t scrollInc){
    uint8_t comm = 0x04;
    if(rightShift)comm|=0x02;
    if(scrollInc)comm|=0x01;
    lcdCommand(comm);
}

void scrollRight(void){
    lcdCommand(0x10|0x08|0x00);
}

void scrollLeft(void){
    lcdCommand(0x10|0x08|0x04);
}

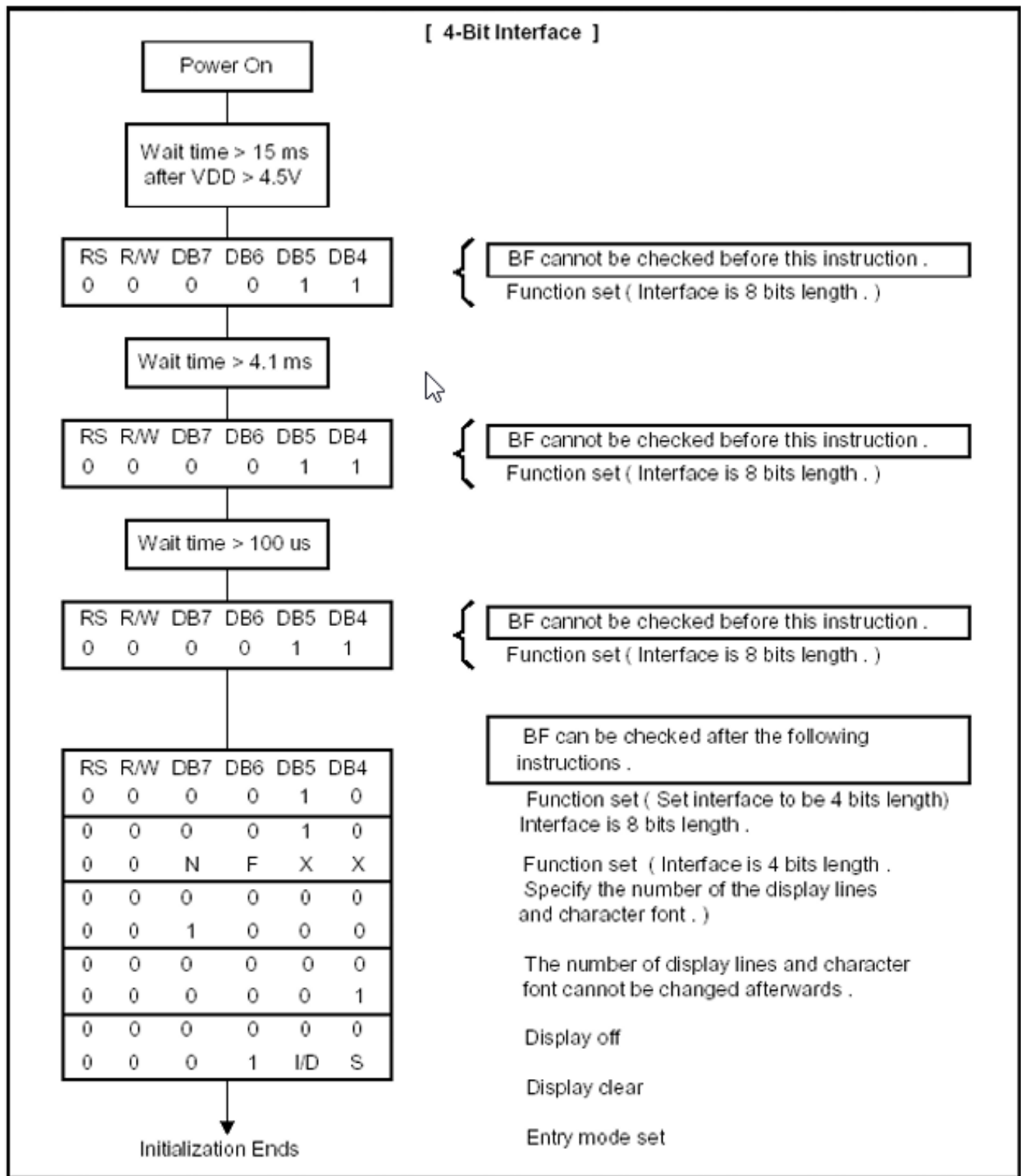
void clear(void)
{
    lcdCommand(0x01);
    HAL_Delay(2);
}

void home(void)
{
    lcdCommand(0x02);
    HAL_Delay(2);
}

```

Для початку роботи необхідно пройти процедуру ініціалізації дисплею за послідовністю поданою в документації (Табл.3).

Таблиця 3. Порядок ініціалізації дисплею.



Переведена в програмний код ініціалізація виглядатиме наступним чином:

```
void initLCD(void){
    HAL_Delay(40);
    lcdCommand(0x02);
    HAL_Delay(1);
    lcdCommand(0x28);
    HAL_Delay(1);
    lcdCommand(0x28);
    HAL_Delay(1);
    paramtrsToggle(1,0,0);
    HAL_Delay(1);
    clear();
    shiftToggle(1,0);
}
```

Для виводу тексту скористаємося функцією передачі даних:

```
void lcdString(char* str){
    for(uint8_t i=0;str[i]!='\0';i++){
        lcdChar(str[i]);
    }
}
```

### **Завдання на лабораторну**

1. Створити проект згідно індивідуального завдання для борду STM32F4 Discovery.
2. Створити проект згідно індивідуального завдання для мікроконтролера STM32F407VG.
3. Виконати індивідуальне завдання.

### **Індивідуальні завдання**

Індивідуальне завдання обирається згідно порядкового номеру в списку підгрупи. У випадку коли порядковий номер у списку перевищує 12 студент обирає варіант за формулою **«порядковий номер» -12**

1. Вивести на дисплей своє прізвище ім'я та по батькові. Вивести на дисплей біжучою стрічкою наступний текст: «Засвічення червоного та синього світлодіодів».



2. Вивести на дисплей своє прізвище ім'я та по батькові. Вивести на дисплей біжучою стрічкою наступний текст: «Вмикання/вимикання червоного та синього світлодіодів».

3. Вивести на дисплей своє прізвище ім'я та по батькові. Вивести на дисплей біжучою стрічкою наступний текст: «Вмикання по черзі червоного та синього світлодіодів».

4. Вивести на дисплей своє прізвище ім'я та по батькові. Вивести на дисплей біжучою стрічкою наступний текст: «Вмикання/вимикання червоного світлодіоду» .

5. Вивести на дисплей своє прізвище ім'я та по батькові. Вивести на дисплей біжучою стрічкою наступний текст: «Вмикання/вимикання червоного та жовтого світлодіодів».

6. Вивести на дисплей своє прізвище ім'я та по батькові. Вивести на дисплей біжучою стрічкою наступний текст: «Почергове циклічне увімкнення жовтого, зеленого, синього та червоного світлодіодів».

7. Вивести на дисплей своє прізвище ім'я та по батькові. Вивести на дисплей біжучою стрічкою наступний текст: «Почергове циклічне увімкнення зеленого, жовтого та червоного світлодіодів».

8. Вивести на дисплей своє прізвище ім'я та по батькові. Вивести на дисплей біжучою стрічкою наступний текст: «Почергове циклічне увімкнення синього червоного світлодіодів».

9. Вивести на дисплей своє прізвище ім'я та по батькові. Вивести на дисплей біжучою стрічкою наступний текст: «Почергове циклічне увімкнення червоного, та синього світлодіодів».

10. Вивести на дисплей своє прізвище ім'я та по батькові. Вивести на дисплей біжучою стрічкою наступний текст: «Перед засвіченням кожного наступного світлодіоду, попередній слід вимкнути».

11. Вивести на дисплей своє прізвище ім'я та по батькові. Вивести на дисплей біжучою стрічкою наступний текст: «Почергове циклічне увімкнення зеленого, жовтого, синього та червоного світлодіодів».

12. Вивести на дисплей своє прізвище ім'я та по батькові. Вивести на дисплей біжучою стрічкою наступний текст: «Почергове циклічне увімкнення червоного та синього світлодіодів».