

ГУАП

КАФЕДРА № 44

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

Доц., канд. техн. наук, доц.

должность, уч. степень, звание

подпись, дата

Т.Н. Соловьёва

инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №3

ОРГАНИЗАЦИЯ ВЗАИМОДЕЙСТВИЯ
МИКРОКОНТРОЛЛЕРА С
ЖИДКОКРИСТАЛЛИЧЕСКИМ ДИСПЛЕЕМ

по курсу: МИКРОКОНТРОЛЛЕРНЫЕ СИСТЕМЫ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №

4142

подпись, дата

К.С. Некрасов

инициалы, фамилия

Санкт-Петербург 2024

Цель работы

Приобретение навыков организации взаимодействия микроконтроллера с устройствами вывода на примере жидкокристаллического знаковосинтезирующего дисплея.

Задание

Требуется разработать программу на языке ассемблера MCS-51 для вывода на экране ЖКИ двух заданных строк. Строки необходимо выровнять по центру экрана, при этом содержимое строк, расположенное в памяти программ, не должно содержать пробелы до первого и после последнего символа.

Задание по варианту 33

1 Строка: номер группы

2 Строка: И.О. Фамилия

Шина управления ЖКИ:

RS: P1.5

RW: P1.7

E: P1.6

Шина данных ЖКИ: P2.4-P2.7

Вид курсора: мигающий

33	P1.5	P1.7	P1.6	P2.4 – P2.7	мигающий
----	------	------	------	-------------	----------

Рисунок 1 – Задание по варианту

Разработка программы

Для начала был подключен ЖК дисплей с подключением портов в соответствии варианту.

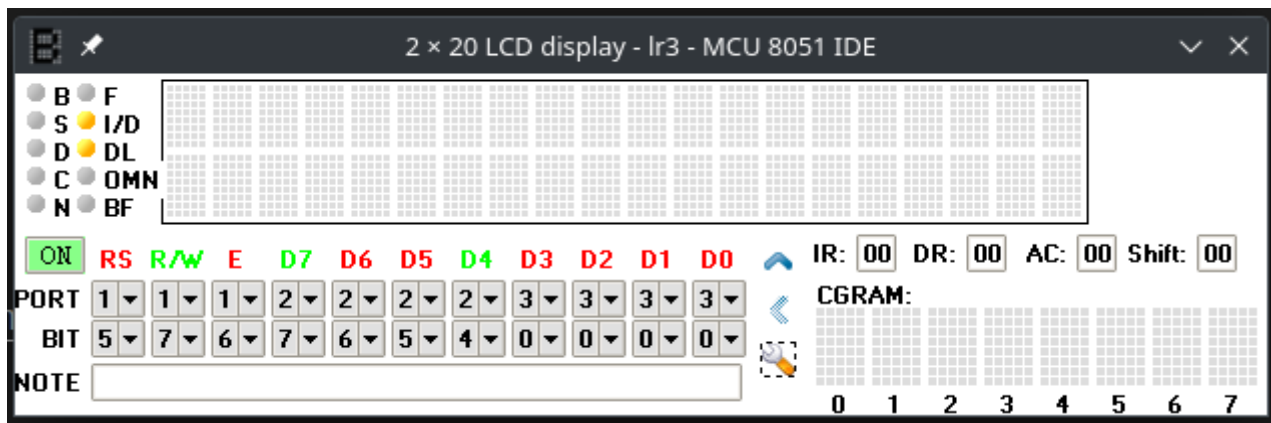


Рисунок 2 – Подключенный ЖК дисплей

Затем был составлен код программы

```
;*****
; *
; Filename: prog-led.asm
; Date: 2024/02/29
; File Version: 1
; Author: Nekrasov K.N.
; Company: SUAI
; Description:
; *
;*****
; Variables
;*****
switch equ 43h ;переключатель «команда-данные» (RS)
bte equ 44h ;выдаваемый на ЖКИ байт
;*****
; Reset Vector
;*****
org 0h ; processor reset vector
ajmp start ; go to beginning of program
;*****
; MAIN PROGRAM
;*****
org 100h
start:
    ; Инициализация ЖКИ
    mov switch, #0 ; переключатель на команду
    mov bte, #020h ; режим 4-х разрядной шины данных
    lcall write_led ; вызов подпрограммы передачи в ЖКИ
```

```

mov bte, #000h    ; активация всех знакомест
lcall write_led
mov bte, #0f0h    ; активация всех знакомест
lcall write_led
mov bte, #000h    ; режим автом. перемещения курсора
lcall write_led
mov bte, #060h    ; режим автом. перемещения курсора
lcall write_led
mov bte, #080h    ; установка адреса первого символа
lcall write_led
mov bte, #080h    ; установка адреса первого символа
lcall write_led

;вывод строк
mov switch, #1 ;переключатель на данные
mov dptr, #0fd0h ;адрес, по которому расположены данные

row1: ;вывод символов первой строки
    clr a
    movc a, @a+dptr
    mov bte, a
    lcall write_led
    clr a
    movc a, @a+dptr
    swap a
    mov bte, a
    lcall write_led
    inc dptr
    mov a, dpl ; младший байт указателя данных
    cjne a, #0d4h, row1 ; пока не введены 4 символа 1ой строки

mov switch, #0 ;команда
mov bte, #0C0h ;установка адреса первого символа второй строки
lcall write_led
mov bte, #040h ;установка адреса первого символа второй строки
lcall write_led
mov switch, #1 ;RS=1 - данные

row2: ;вывод символов второй строки
    clr a

```

```

movc a, @a+dptr
mov bte, a
lcall write_led
clr a
movc a, @a+dptr
swap a
mov bte, a
lcall write_led
inc dptr
mov a, dpl
cjne a, #0e0h, row2 ;d4h+12h=e0h – адр. конца второй стр.

```

```

jmp finish ;переход на конец программы

```

```

write_led: ;подпрограмма передачи в ЖКИ

```

```

mov p2, bte ;передаваемый байт – в P2
setb p1.6    ;E:=1
clr p1.7     ;RW:=0 (запись)

```

```

mov a, switch
mov c, acc.0 ;нам нужен 0-ой бит аккумулятора
mov p1.5, c ;RS:=switch (команда/данные)

```

```

lcall delay ;вызов подпрограммы задержки

```

```

clr p1.6 ;E:=0

```

```

lcall delay

```

```

setb p1.6 ;E:=1
ret

```

```

delay: ;подпрограмма задержки на 40мкс

```

```

push A ;сохраняем аккумулятор в стеке
mov A, #0Ah ; 40 = 2+2+1+A(1+2)+1+2+2
m:

```

```

    dec A
jnz m
nop
pop A ;восстанавливаем значение аккумулятора

```

```

ret

org 0FD0h ;данные располагаем в памяти программ
data:
    db '4142'
    db 'K.S.Nekrasov'
finish: sjmp $ ;конец программы
end

```

Для режима 4-х разрядной шины для начала мы передаём байт 20h, после чего производим настройку, передавая в старшем полубайте сперва старший полубайт настройки, затем младший. С символами поступаем аналогично, записывая сначала верхний полубайт символа, затем меняя местами полубайты командой swap.

Для центрирования текста, при установке начального положения курсора мы прибавляем к нему отступ от левого края (8 для первой строки и 4 для второй)

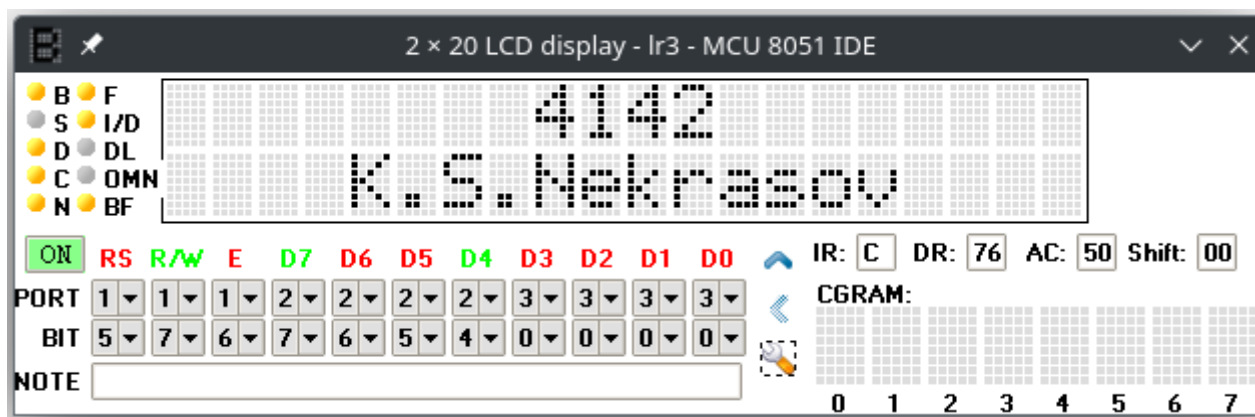


Рисунок 3 – Результат выполнения программы

Вывод

В результате выполнения работы разработана программа на языке ассемблера MCS-51 для вывода на экране ЖКИ двух заданных строк. Проверка работоспособности программы произведена в среде MCU 8051 IDE. Приобретены навыки организации взаимодействия микроконтроллера с устройствами вывода на примере жидкокристаллического знакосинтезирующего дисплея