Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра Информатики

Дисциплина «Программирование»

**ОТЧЕТ**

к лабораторной работе №2

на тему:

**«ЛОГИЧЕСКИЕ КОМАНДЫ. КОМАНДЫ РАБОТЫ С БИТОВЫМИ ПОЛЯМИ. КОМАНДЫ СДВИГОВ. КОМАНДЫ ПЕРЕДАЧИ УПРАВЛЕНИЯ. СПЕЦИАЛЬНЫЕ КОМАНДЫ»**

БГУИР 6-05-0612-02 01

|  |
| --- |
| Выполнил студент группы 353503  АБДУЛОВ Александр Алексеевич |
|  |
| (дата, подпись студента) |
| Проверил ассистент каф. Информатики  РОМАНЮК Максим Валерьевич |
|  |
| (дата, подпись преподавателя) |

Минск 2024

# 1 Индивидуальное задание

**Задание 1. Вариант 1.** Написать тремя способами установку битов 2 и 3 в ячейке памяти $10.

**Задание 2. Вариант 1.** Произвести операцию «логическое И» между битом 0 и битом 4 для ячеек памяти, расположенных по адресам $8200 ... $821f, результат при этом должен быть записан в бит 1 соответствующей ячейки.

# 2 Выполнение работы

Для написания первого задания лабораторной работы были использованы команды пересылки данных и логические команды BSET, BCLR, ORAA, COMA, ROLA. Ниже представлен листинг кода.

org $8000

;способ 1

bset $10,#%00001100

bclr $10,#%11110011

ldaa #$ff

staa $10

;способ 2

bclr $10,#%11111111

ldaa $10

oraa #%00001100

staa $10

ldaa #$ff

staa $10

;способ 3

ldaa $10

asla

asla

asla

asla

asla

asla

asla

asla

coma

asla

asla

asla

asla

asla

asla

rola

rola

asla

asla

staa $10

Способ 1: Cначала устанавливаются биты 2 и 3 в регистре по адресу $10 с помощью команды, которая позволяет установить определённые биты. Затем происходит сброс остальных битов 0, 1, 4 и 5, чтобы гарантировать, что в регистре останутся только нужные биты.

Способ 2: В этом способе сначала все биты в ячейке памяти обнуляются, что позволяет избежать случайных значений. Затем текущее значение из ячейки памяти загружается в регистр A. С помощью операции ИЛИ устанавливаются биты 2 и 3. Далее новое значение из регистра A записывается обратно в ячейку памяти.

Способ 3: Здесь сначала загружается текущее значение из памяти по адресу $10 в регистр A. Значение сдвигается влево несколько раз, чтобы освободить нужные биты и подготовить его к инверсии. Затем происходит инверсия всех битов регистров, что позволяет сменить значения битов. После инверсии выполняются дополнительные сдвиги, чтобы вернуть нужные биты на место. Наконец, окончательное значение записывается обратно в ячейку памяти.

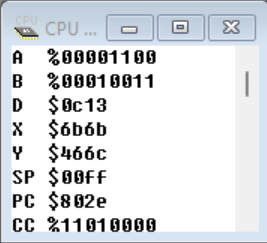


Рисунок 1 – Значения регистров после выполнения программы

На рисунке 2 представлены значения ячеек памяти после выполнения программы.

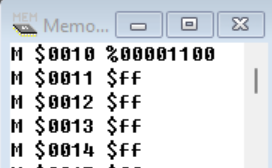


Рисунок 2 – Значения ячеек памяти после выполнения программы

Для написания второго задания, были использованы команды для работы с регистрами, команды для логических операций, а также условные переходы и прерывание. Листинг кода представлен ниже.

org $8000

START:

ldx #$8200

LOOP:

ldaa $0,x

anda #%00000001

beq ZERO

ldab $0,x

andb #%00010000

bne NOTZERO

ZERO:

bclr $0,x,#%00000010

bra NEXT

NOTZERO:

bset $0,x,#%00000010

bra NEXT

NEXT:

ldaa $0,x

adda #%01001011

inx

staa $0,x

clra

clrb

cpx #$821f

bne LOOP

swi

В первых строках кода происходит инициализация регистра X значением $8200, что задает начальный адрес для обработки ячеек памяти. В данном случае регистр X используется для указания на текущую ячейку в диапазоне от $8200 до $821F.

Метка LOOP сигнализирует о начале основного цикла обработки, в котором происходит чтение значения из текущей ячейки памяти с использованием команды загрузки в регистр A. Затем выполняется операция «логическое И» с битом 0. Если результат равен нулю, происходит переход к метке ZERO, что указывает на то, что бит 0 не установлен.

Если бит 0 установлен, значение из той же ячейки загружается в регистр B, и выполняется операция «логическое И» с битом 4. Если бит 4 также не установлен, происходит переход к метке ZERO, где бит 1 в соответствующей ячейке сбрасывается.

Если оба бита 0 и 4 установлены, выполняется команда установки бита 1 в ячейке памяти. После этого программа переходит к метке NEXT, где происходит подготовка для обработки следующей ячейки.

В метке NEXT значение из текущей ячейки загружается обратно в регистр A, к которому добавляется фиксированное значение. Затем адрес в регистре X увеличивается для перехода к следующей ячейке.

В конце выполняется проверка: если текущий адрес не достиг конечного значения $821F, программа возвращается к началу цикла. Если же все ячейки обработаны, вызывается команда прерывания SWI, завершая выполнение программы.

На рисунке 3 представлены промежуточные значения ячеек памяти.

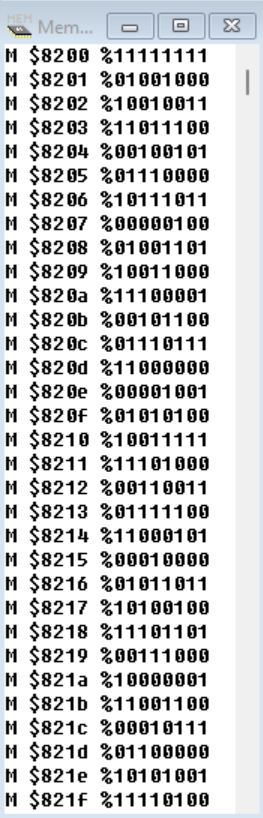


Рисунок 3 – Промежуточные значения ячеек памяти

# Вывод

В ходе лабораторной работы были изучены различные логические команды, команды для работы с битовыми полями, команды сдвигов, передачи управления, в том числе условного, безусловного переходов, команды для работы с подпрограммами и прерываниями. Изучены специальные команды.