

## 6 Лабораторная работа №6. Сложение, вычитание. Логика

**Цель:** изучить действие инструкций ADDWF, ANDWF, IORWF, XORWF, SUBWF. Выполнить сложение, вычитание, применить логические функции.

Задание для внеаудиторной подготовки:

Изучить формат инструкций ADDWF, SUBWF, ANDWF, IORWF, XORWF по приложению Д.

### 6.1 Общие сведения

Операции сложения и вычитания применяются, например, для автоматизации учета количества автомашин на объекте, которые могут свободно въезжать и выезжать с территории через ворота.

В таблице 6.1 представлены логические функции, их таблицы истинности, условные обозначения и интерпретация на электрической схеме. Все логические операции выполняются поразрядно.

Таблица 6.1 – Логические функции  $Y=f(X1, X2)$

Инструкция	Таблица истинности			Условное обозначение	Интерпретация на электрической схеме
	X1	X2	Y		
IORWF Функция OR (ИЛИ)	0	0	0		 <p>При замыкании любого контакта в цепи будет ток</p>
ANDWF Функция AND (И)	0	0	0		 <p>Ток в цепи будет только при замыкании обоих контактов</p>
XORWF Функция XOR (Исключительно ИЛИ)	0	0	0		
	0	1	1		
	1	0	1		 <p>Ток в цепи будет только тогда, когда переключатели X1 и X2 будут в разном положении</p>
	1	1	0		
	0	0	0		
	0	1	1		

Логическая функция «ИЛИ» («OR») может применяться для включения оборудования любой из двух кнопок, расположенными в разных местах помещения.

Логическая функция «И» («AND») может применяться в том случае, если оборудование должно включаться, при срабатывании (включении) двух контактов. Например, освещение в помещении автоматически включается при низкой освещенности и наличия в нем людей.

Логическая функция «Исключительно ИЛИ» («XOR») применяется, например, для управления освещением длинного тоннеля. С любого конца тоннеля можно включить или выключить освещение.

Изучить программу 6.1, которая выполняет различные действия с двумя константами и выводит результаты работы в регистр REZ.

Программа 6.1 – Сложение и вычитание данных.

```
include <p16F877.inc>
```

```
R1 EQU h'21' ; присваиваем регистру по адресу h'21' имя R1
```

```
R2 EQU h'22' ; присваиваем регистру по адресу h'22' имя R2
```

```
R3 EQU h'23' ; R3 – имя регистра по адресу h'23'
```

```
REZ EQU h'24' ; REZ - имя регистра по адресу h'24' для записи результатов
```

```
org h'00' ; инструкция (nop) будет записана по адресу h'00'
```

```
nop ; пустая инструкция будет записана по адресу h'00'
```

```
nop ; пустая инструкция будет записана по адресу h'01'
```

```
nop
```

```
org h'05' ; следующая инструкция будет записана по адресу h'05'
```

START

```
CLRF STATUS ; очищаем регистр STATUS и выбираем нулевой банк
```

```
BSF STATUS, 5 ; переходим в первый банк
```

```
CLRF TRISC ; настраиваем биты PORTC на выход
```

```
BCF STATUS, 5 ; возвращаемся в нулевой банк
```

```
CLRF REZ ; записать в регистр REZ нули
```

```
MOVLW D'5' ; запись константы в W
```

```
MOVWF R3 ; запись содержимого W в регистр R3
```

```
MOVLW D'130' ; запись константы в W
```

```
MOVWF R2 ; запись содержимого W в регистр R2
```

```
MOVLW D'240' ; запись константы в W
```

```
MOVWF R1 ; запись содержимого W в регистр R1
```

```
MOVF R1, W ; копирование константы из R1 в регистр W
```

```
ADDWF R2, W ; сложение W=R2+ W=R2+R1
```

```
MOVWF REZ ; загрузить содержимое W в регистр REZ
```

```
MOVF R1, W ; копирование константы из R1 в регистр W
```

```
SUBWF R2, W ; вычитание W=R2-W=R2- R1
```

```
MOVWF REZ ; переслать содержимое W в регистр REZ
```

```
MOVF R2, W ; копирование константы из R2 в регистр W
```

```
SUBWF R1, W ; W=R1-W=R1-R2
```

```
MOVWF REZ ;
```

MOVF R2, W ;  
 SUBWF R1, W ; вычитание  $W=R1-W=R1-R2$ . Результат оставляем в W  
 MOVWF REZ ; переслать содержимое W в регистр REZ  
 MOVF R2, W ; запись константы в регистр W из R2  
 ANDWF R1, W ;  $W=R1 \text{ AND } W=R1 \text{ AND } R2$ . Результат заносим в W  
 MOVWF REZ ; копируем аккумулятор в регистр REZ.  
 END

## 6.2 Порядок выполнения

Ознакомиться с предложенной выше программой и самостоятельно продолжить программу командами для выполнения операций R2-R2, R2+R3, R1 OR R2, R1 XOR R2.

Выбрать вариант по таблице 6.2 и составить программу.

Таблица 6.2 – Варианты задания

Вариант	Числа R1, R2, R3	Вариант	Числа R1, R2, R3
1	81h, 12h, 1h	6	86h, 67h, 6h
2	82h, 23h, 2h	7	87h, 68h, 7h
3	83h, 34h, 3h	8	88h, 69h, 8h
4	84h, 45h, 4h	9	89h, 6Ah, 9h
5	85h, 56h, 5h	10	8Ah, 6Bh, Ah

Создать окно наблюдения для всех применяемых в программе регистров в необходимом формате чисел. Результаты работы всех операций с числами записать в таблицу 6.3.

Таблица 6.3 - Пример записи результатов работы команд

Числа на входе		Действие	Результат в REZ		Значение в Status
Форма В или D	Форма В или D		Форма В	Форма D	Форма В
R1=D'240'	R2=D'130'	R1+R2	-	114	00011001
R1=D'240'	R2=D'130'	R1-R2	-	110	00011001
R1=D'240'	R2=D'130'	R2-R1	-	146	00011010
R2=D'130'	R2=D'130'	R2-R2	-	0	00011111
R2=D'130'	R3=D'5'	R2+R3	-	135	00011000
R1=b'11110000'	R2=b'10000010'	R1 AND R2	10000000	-	00011011
R1=b'11110000'	R2=b'10000010'	R1 OR R2	11110010	-	00011011
R1=b'11110000'	R2=b'10000010'	R1 XOR R2	01110010	-	00011011

## 6.3 Контрольные вопросы

6.3.1 Как определить с регистрами какого банка работает программа?

6.3.2 Что выполняют инструкции:

ADDWF R1, f; SUBWF R2, f; ?

6.3.3 Что выполняют инструкции:

ANDWF R1, f; IORWF R1, W; XORWF R2, W; ?

6.3.4 Какое назначение директивы ORG h'05'?

6.3.5 Приведите таблицы истинности логических операций.

6.3.6 В каком банке находятся регистры R1, R2 и REZ?

6.3.7 Какой результат мы увидим при сложении 230+60 в МК?

6.3.8 Какой результат мы увидим при вычитании 130-135 в МК?