

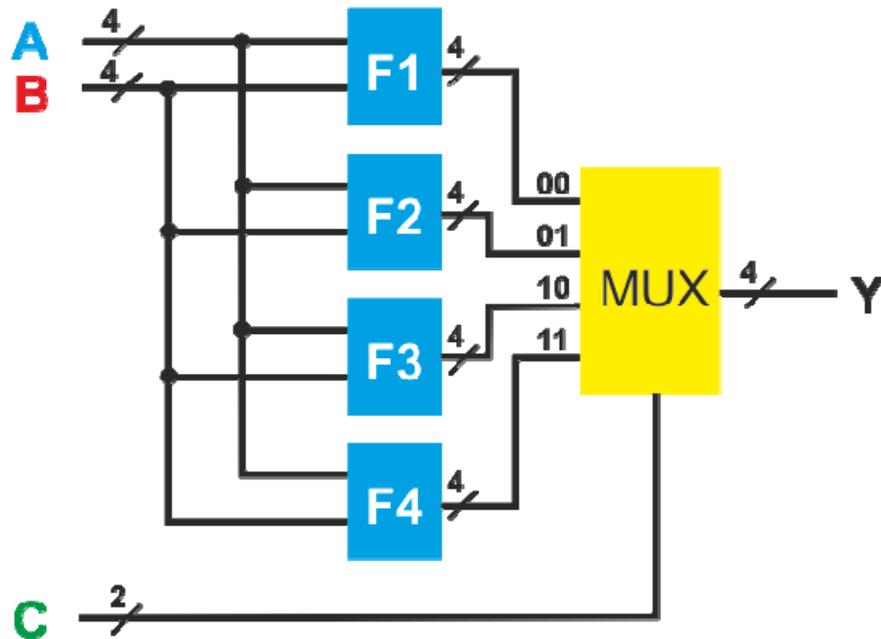
ЛАБОРАТОРНАЯ РАБОТА 2

Задача:

Создать арифметическо-логическое устройство (АЛУ), реализующее заданные функции.

Необходимые элементы найти в элементной базе **DixiCad** или реализовать самому.

4-х битовые операнды должны задаваться таким образом, чтобы было видно, что реализуемая операция работает. В отчете предоставить эти варианты для каждой из операций (в двоичном виде).



F1: $A+B$, $A-B$, etc.

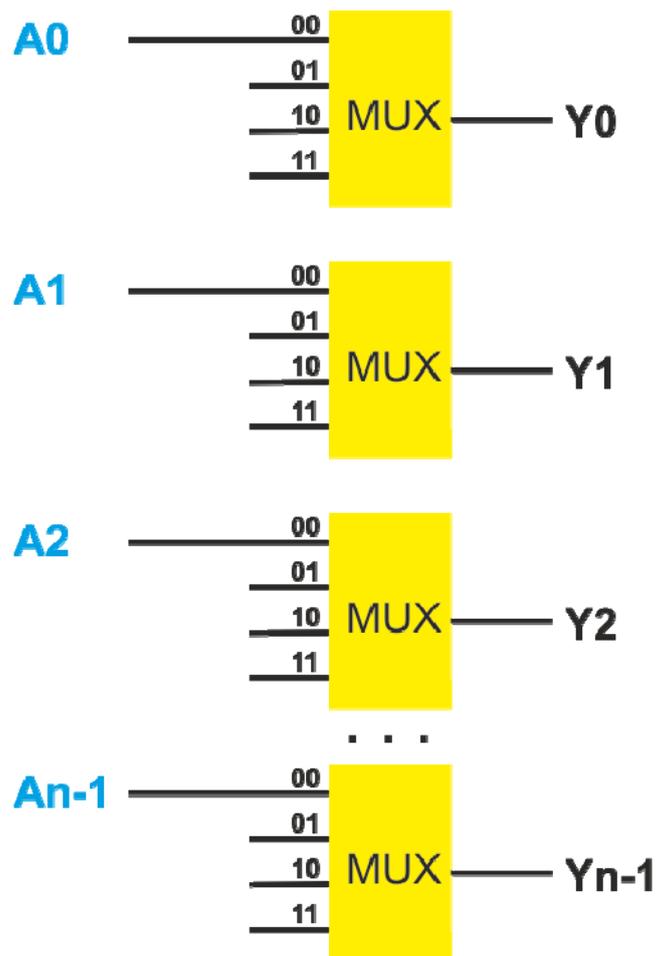
F2: shr A, shr B, etc.

F3: set A,B clr A,B xor A,B etc.

F4: AND, OR,XOR, etc.

2-4MUX	74153
4bitSUM	74S23
Comparator	7485

Схематическое представление работы



Для представления n -битного операнда A в схеме используют n входов $A_{n-1}..A_0$ (A_{n-1} – старший бит, A_0 - младший)

NB! Для реализации сдвига использовать элементы памяти не нужно.

Поясняющие примеры для операций xor, clr, set:

- **xor A, B** (inverteerida sõna A B-nda biti väärtus)

меняет значение бита на противоположное

Пример 1:

A:= $\begin{matrix} 3 & 2 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{matrix}$ B:=0010=2₁₀



A:= $\begin{matrix} 3 & 2 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{matrix}$

Пример 2:

A:= $\begin{matrix} 3 & 2 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{matrix}$ B:=0010=2₁₀



A:= $\begin{matrix} 3 & 2 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{matrix}$

- **clr A, B** (seada sõna A B-nda biti väärtuseks '0')

устанавливает значение бита в '0'

Пример:

A:= $\begin{matrix} 3 & 2 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{matrix}$ B:=0010=2₁₀



A:= $\begin{matrix} 3 & 2 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{matrix}$

- **set A, B** (seada sõna A B-nda biti väärtuseks '1')

устанавливает значение бита в '1'

Пример:

A:= $\begin{matrix} 3 & 2 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{matrix}$ B:=0010=2₁₀



A:= $\begin{matrix} 3 & 2 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{matrix}$

Данные задавать таким образом, чтобы на каждую операцию было бы задано, по крайней мере, по **4 варианта операндов (тесты)**.

4 варианта на каждую из операций должны быть представлены в отчете (Соответственно эти же значения должны использоваться и при симулировании схемы в программе **DixiCad**).

В отчете должны быть подробно показаны вычисления (Например, сложение выполнить в столбик с обозначением возникающих переносов. Если, например, вычитание производится на основе сложения, то это должно отражаться и на приведенных в отчете вычислениях).

Для компаратора тесты выбираются таким образом, чтобы проверить следующие варианты

$$A < B, A = B \text{ и } A > B.$$

Всего минимум 3 теста.

Сдвиг – 4 теста, например, сдвигать числа 1, 2, 4 и 8.

При **циклическом сдвиге** выбирать числа таким образом, чтобы было видно, что произведен циклический сдвиг (то есть выдвигаться должна '1').

Всего 4 теста.

Для операций, изменяющих бит числа, надо выбирать исходные числа таким образом, чтобы результат **отличался** бы от исходного числа, проверить изменения для всех битов числа, показать как будет изменяться число, **если B будет больше, чем 3**.

При задании тестов можно воспользоваться и советами Элмета Орассона по выбору операндов:

Для битовых операций:

– инверсия: можно выбрать следующие значения 1, 2, 4, 8, 0xE, 0xD, 0xB, 0x7

– операции, изменяющие 1 бит: Например, $A=0$, $B=\{0,1,2,3$ и произвольное $B>3\}$

$A=0xF$ $B=\{0,1,2$ и произвольное $B>3\}$.

Всего 8 тестов.

Логические операции:

– во всех случаях 4 теста

$A=0x0$, $B=0x0$; $A=0xF$, $B=0x0$; $A=0xF$, $B=0xF$,

$A=0x0$, $B=0xF$

– AND, NAND: $A=0xF$, $B=\{1,2,4,8\}$, и такие же тесты с другими операндами.

– NOR, OR: $A=0x0$, $B=\{1,2,4,8\}$, и такие же тесты с другими операндами

– XOR – аналогично

Всего 12 тестов.

В отчете должны быть

- схема (нарисованная вручную, но лучше напечатанная)
- тесты (с подробными вычислениями)

Вопросы на защите будут по

- ✓ элементам схемы: сумматор, декодер, мультиплексор (необходимо знать таблицы истинности этих элементов и уметь рисовать по таблицам внутреннюю структуру этих элементов, используя заданные преподавателем логические элементы), логические элементы.
- ✓ уметь пояснять, как в работе реализованы заданные операции.

NB! Использовать MACRO нет необходимости.