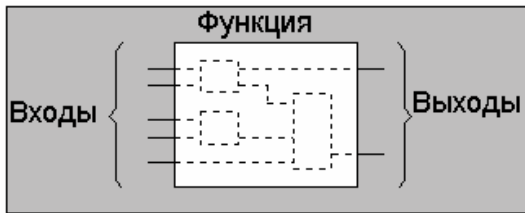


5. Язык функциональных блокковых диаграмм (FBD)

5.1 Функциональные блоки



Каждый функциональный блок графически представляет собой прямоугольник, внутри которого имеется обозначение функции, выполняемой блоком. Выход функционального блока может быть соединен связью с другими блоками. Каждая связь является **переменной**. Один или несколько функцио-

нальных блоков, соединенных связями между собой, образуют программу на языке **FBD**. Язык **FBD** описывает закон преобразования переменных программы.

5.2 Правила языка FBD

Имеются следующие правила языка **FBD**:

- ◆ функциональные блоки могут располагаться произвольно в поле программы;
- ◆ не может быть свободных (несоединенных) входов и выходов функционального блока;
- ◆ любая связь может иметь имя и тип;
- ◆ входы и выходы функциональных блоков, присоединенные к связям, имеющим одинаковые имена, считаются соединенными;
- ◆ очередность выполнения блоков в программе: сверху вниз, слева направо.

В том случае, когда одну и ту же переменную формируют несколько функциональных блоков, последнее значение **переменной** будет присвоено тем блоком, который выполняется **последним**.

Каждый функциональный блок имеет условную точку привязки, которая определяет очередность выполнения блока в программе. Точкой привязки является левый верхний угол блока. В приведенном примере (рис.5-1) цифры внутри блоков показывают очередность выполнения блоков в программе.

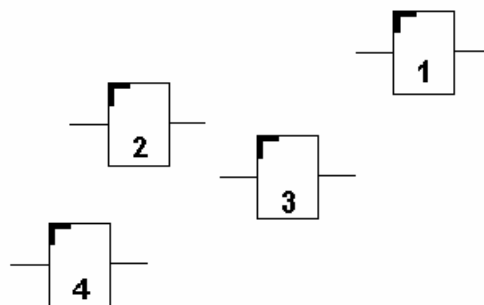


Рис.5-1. Порядок выполнения блоков программы

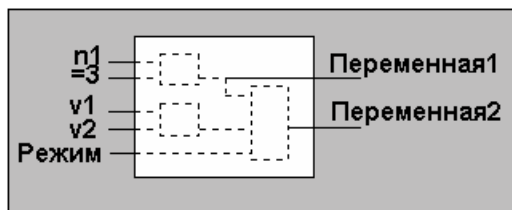
5.3 Базовые типы объектов

Программа на языке **FBD** может содержать следующие базовые типы объектов:

- ◆ функциональные блоки и связи;
- ◆ переходы и метки;
- ◆ комментарии.

5.3.1 Функциональные блоки и связи

Связям, присоединяемым к входам и выходам функциональных блоков, ставятся в соответствие переменные и константы, которые могут иметь атрибуты, описанные в гл.4, п.4.3. Установление соответствия между связью и именем переменной выполняется в процессе присвоения связям символьных имен.



На входе **FBD** блока может быть:

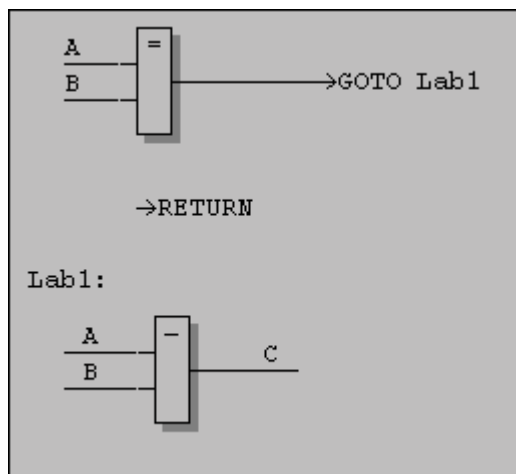
- ◆ константное выражение;
- ◆ любая переменная.

На выходе **FBD** блока может быть любая переменная.

5.3.2 Переходы и метки

Метки и переходы можно использовать для управления процессом выполнения программы. Для этого выход блока соединяют связью с оператором **GOTO** {имя метки}. Если переменная, которая формируется на выходе функционального блока, принимает значение **TRUE**, то программа продолжается с блока, помеченного меткой. Если оператор **GOTO** не соединен с переменной типа **BOOLEAN**, осуществляется безусловный переход к метке.

Для завершения программы существует оператор **RETURN**. Если на входе оператора **RETURN** значение переменной принимает значение **TRUE**, то осуществляется условный



переход к следующей программе. Если оператор **RETURN** не соединен с переменной типа **BOOLEAN**, осуществляется безусловный переход к следующей программе. В случае, когда оператор **RETURN** встретился внутри библиотечного блока, управление передается следующему блоку, но не следующей программе.

В приведенном примере оператор **GOTO** соединен с выходом блока сравнения. В случае равенства переменных **A** и **B** программа перейдет к метке Lab1, иначе будет выполнен оператор **RETURN**.

5.3.3 Комментарии

Комментарии - это произвольный текст в поле программы. Комментарий может быть многострочным и его размер не ограничен.