2.4. Семантический анализатор

Данный этап вводится для осуществления контроля инициализации переменных, используемых в программе. Для выполнения этого действия необходимо разработать алгоритм, выполнить программную реализацию и провести тестирование.

2.4.1. Разработка алгоритма

В разрабатываемом языке инициализацией переменной является связывание переменной с номером порта на плате Arduino через создание директивы препроцессора, а также присваивание переменной какого-то значения. Для упрощения дальнейшей разработки было принято решение о записи номера регистра в таблицу имен вместе с переменной присваивания. Номер регистра определяется свободно доступными регистрами и увеличивается каждый раз, когда происходит присваивание. Тогда проверка инициализации необходима только в параметрах и условиях. Схема алгоритма представлена на рисунке 18.

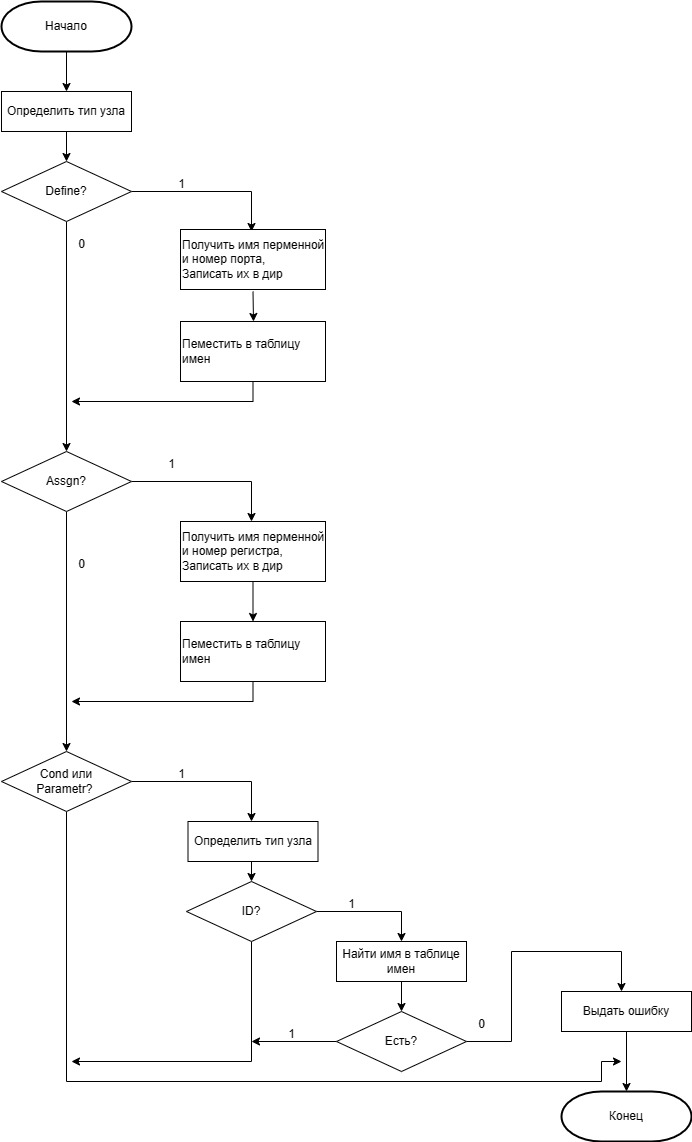


Рисунок 18 – Схема алгоритма семантического анализатора

Первоначально необходимо определить тип узла. Если тип равен «define» или «assgn», происходит создание директивы с записью в нее имени переменной и номера порта или регистра соответственно. Далее созданная директория записывается в таблицу имен. Если же тип равен «cond» или «parametr», то необходимо найти переменную и сравнить ее с таблицей имен. Для ее нахождения необходимо определить узел с типом ID. Сравнение с таблицей имен подразумевает поиск имени в таблице.

2.4.2. Программная реализация

В разрабатываемом языке инициализация переменной должна происходить до момента ее использования через директиву препроцессора или во время ее использования в операции присваивания. Такого порядка следования можно достичь за счет использования алгоритма рекурсивного обхода дерева разбора. Кроме того, использование дерева разбора позволяет обращаться напрямую к узлам для получения значений, так как порядок узлов известен.

Например, для получения значения имени макроса возможно следующее обращение:

node\* macros = current\_node->operand1;

string name\_macros = macros->operand1->value;

Для возможности создания директивы и таблицы имен были разработаны соответствующие классы. Диаграмма классов представлена на рисунке 19.

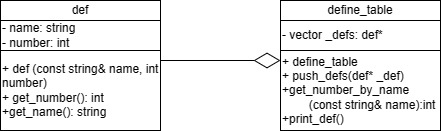


Рисунок 19 – Диаграмма классов семантического анализатора

Листинг программы, содержащий описание классов и алгоритм семантического анализа приведен в приложении В.

2.4.5. Тестирование