

4. Базовые концепции

4.1 Программа управления

Программа управления объектом в контроллере представляется как некоторое логическое устройство, описывающее управление **технологическим процессом** и операции над переменными величинами этого процесса. Программа может состоять из множества других программ, находящихся друг с другом в определенных отношениях, образующих иерархическое дерево. Каждая программа может иметь состояние **Start** или **Stop**. Состояние **Start** означает, что данная программа будет запущена при включении контроллера. Состояние **Stop** означает, что данная программа может быть запущена другими программами при определенных условиях.

4.2 Переменные

Переменным величинам технологического процесса соответствуют *переменные программы*, которые характеризуются именем и типом.

Переменная- это область памяти, в которой находятся данные, с которыми оперирует программа.

Поддерживаются следующие типы переменных:

Переменные типа BOOLEAN

Переменные могут принимать значения - 0(**False**) и 1(**True**).

Переменные типа INTEGER - 16-ти разрядное число со знаком.

Переменные могут принимать целочисленные значения от **-32768** до **+32767**.

Переменные типа SHORTINT – 8-разрядное число со знаком.

Переменные могут принимать целочисленные значения от **-128** до **+127**.

Переменные типа BYTE – 8-разрядное число без знака.

Переменные могут принимать целочисленные значения от **0** до **255**.

Переменные типа LONGINT – 64- разрядное число со знаком.

Переменные могут принимать целочисленные значения от **-2³¹** до **+ 2³¹ -1**.

Переменные типа WORD – 16 - разрядное число без знака.

Переменные могут принимать целочисленные значения от **0** до **65536**.

Переменные типа DWORD – 32 - разрядное число без знака.

Переменные могут принимать целочисленные значения от **0** до **4 294 967 296**.

Переменные типа FLOAT.

Переменные этого типа могут принимать значения от **$\pm 1,18 \times 10^{-38}$** до **$\pm 3,4 \times 10^{+38}$** (стандарт **ANSI/IEEE 754-1985** для двоичной арифметики с плавающей точкой).

Переменные типа **Timer**.

Таймерная переменная принимает значение времени.

Значение переменной не может превышать **248d13h13m56s47**, где:

d - дни, **h** - часы, **m** - минуты, **s** - секунды, **47** – сотые доли секунды.

Цифры после знака **S** всегда означают сотые доли секунды (сантисекунды). Так S10 - десять сантисекунд, S1 - одна сантисекунда.

Типы констант соответствуют основным типам переменных. Признаком константы является знак **=**.

Например:

=2 константа типа **Integer**;
=True константа типа **Boolean**;
=2.21 константа типа **Float**;
=0S1 константа типа **Timer**.

4.3 Атрибуты переменных

Каждая переменная может иметь один или несколько атрибутов. Атрибут означает какое либо свойство переменной независимое от ее типа.

По свойству видимости в проекте переменные могут быть **глобальными** и **локальными**. Глобальные переменные доступны (видимы) всем программам в проекте. Локальная переменная доступна только той программе, внутри которой она объявлена.

Глобальные переменные могут иметь следующие **атрибуты**:

Константа	число, значение которого не может быть изменено программой;
Входная	переменная, логически соединенная с входом контроллера;
Выходная	переменная, логически соединенная с выходом контроллера;
Сетевая	переменная доступная всем узлам сети;
Удаленная	переменная другого узла сети;
Сохраняемая	переменная сохраняется в энергонезависимой памяти. При запуске программы значение переменной считывается из энергонезависимой памяти и записывается в память в конце каждого цикла контроллера. Если по каким-либо причинам значение переменной не может быть корректно считано из энергонезависимой памяти (техническая неисправность, сбой данных и т.п.), тогда переменной присваивается значение по умолчанию, заданное в программе.

4.4 Имена переменных

Имена программам и переменным присваивают по следующим правилам:

- ♦ имя не может содержать более **255** символов;
- ♦ первый символ должен быть буквой, или символом подчеркивания, далее могут следовать цифры, буквы латинского алфавита и символы подчеркивания.

4.5 Циклограмма работы контроллера

Работа контроллера имеет циклический характер. Весь контролируемый технологический процесс разбивается на ряд формальных независимых задач, каждая из которых обслуживается отдельной программой. Программы выполняются поочередно с одинаковым приоритетом. Циклограмма работы контроллера представлена на рисунке 4-1.



Рис.4-1. Циклограмма работы контроллера