

*Инж. С.Я. Куцаков, канд. физ-мат. наук А.О. Мирошник,
канд. техн. наук Б.И. Решмин*

ПРОБЛЕМЫ РАЗВИТИЯ МИКРОСРЕДСТВ ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ В-10

Одно из традиционных направлений деятельности НПП ВНИИЭМ связано с разработкой микросредств вычислительной техники (МСУВТ), предназначенных для автоматизации технологических процессов в промышленности, атомной энергетике, использования в составе массового электротехнического оборудования.

На протяжении ряда лет проводились разработки МСУВТ серии В-10. Была разработана и выпущена широкая номенклатура процессорных модулей, модулей цифрового и аналогового ввода-вывода, плат узкоспециализированного назначения.

Высокая надежность и экономичность заложенных в интерфейсе В-10 архитектурных решений, технологичность конструктивного исполнения промышленных систем на базе МСУВТ В-10 обеспечили изделию длительный срок существования и развития. Номенклатура выпущенных только за последние несколько лет модулей превысила 30 наименований. Вычислительные средства были аттестованы для применения на атомных станциях и получили индекс В-10Р.

Однако длительный срок существования МСУВТ В-10 имел и свои негативные последствия:

1. Традиционно используемые в составе МСУВТ В-10 процессорные модули на базе микропроцессорной серии Intel MCS-51 перестали удовлетворять современным требованиям по быстродействию, производительности и сервисным функциям.

2. Повсеместное применение на верхних уровнях систем автоматизации персональных ЭВМ поставило задачу архитектурной, программной и интерфейсной совместимости с ними технических средств нижнего контроллерного уровня.

3. Усложнение задач, решаемых при управлении технологическими процессами, потребовало привлечения распределенных вычислительных архитектур и, как следствие, обеспечения сетевого взаимодействия между компонентами систем.

4. Используемые для написания прикладных программ традиционные языки программирования (СИ, Паскаль, Ассемблер) требовали высокой квалификации программиста и досконального знания

технических средств и внутренней организации контроллера, ограничивали возможность конечного пользователя по самостоятельному написанию технологических и управляющих программ.

Чтобы привести в сложившихся условиях серию микросредств в соответствие с современными требованиями, сохранив все имеющиеся наработки и обеспечить дальнейшее развитие МСУВТ В-10, было необходимо:

1. Перейти к широкому использованию современных высоко-технологических импортных комплектующих в составе модулей комплекта В-10.

2. При проектировании процессорных плат использовать импортные встраиваемые процессорные модули и специализированную периферию, устанавливаемые на переходных платах-носителях.

3. В качестве носителей информации использовать электронные накопительные устройства большой емкости на базе флэш-памяти не имеющие механических частей и допускающие жесткие условия эксплуатации.

4. Для ускорения разработки и получения универсального программного обеспечения для контроллеров использовать стандартные языки технологического программирования со встроенными средствами сетевого взаимодействия.

После проведения технических, конструктивных и маркетинговых исследований имеющихся на отечественном рынке встраиваемых РС-совместимых процессорных модулей было принято решение об использовании серии модулей стандарта РС-104, разработанных фирмой AMPRO.

Изделия стандарта РС-104 выпускают несколько десятков компаний, объединенных в консорциум РС-104. Консорциум представляет собой неприбыльную организацию, основной целью которой является популяризация стандарта РС-104 в качестве мезонинной шины в средствах промышленной автоматизации. Консорциум открыт для вступления новых фирм, число которых постоянно увеличивается, появились первые отечественные производители плат формата РС-104. Номенклатура имеющихся на рынке модулей РС-104 превышает несколько сотен наименований.

Специально для использования модулей РС-104 в рамках вычислительных средств МСУВТ В-10 была разработана платаноситель ПМИ-06 [1], обеспечивающая интерфейс между РС-совместимой магистралью модулей РС-104 и шиной В-10. Обмен происходит с помощью проекции части пространства ввода-вывода шины ISA на географическое адресное пространство шины В-10. На плате ПМИ-06 конструктивно может быть размещено до четы-

рех модулей РС-104, при этом плата занимает два слота в конструктиве В-10.

Стандартный набор модулей РС-104, устанавливаемый на ПМИ-06 включает в себя:

1. Процессорную плату, производительность микропроцессора которой, в зависимости от сложности решаемых задач и требований по охлаждению системы, может лежать в диапазоне от Intel 80x86 до уровня современных Pentium-процессоров. Для специальных применений возможна установка модулей со специализированными управляющими и сигнальными процессорами.

2. Сетевую плату для обеспечения сетевого взаимодействия и связи с верхними уровнями систем автоматизации. В качестве сетевого интерфейса могут быть использованы как традиционные Ethernet или RS-422/485 модули, так и перспективные модули стандартов ProfiBus.

3. Плату твердотельного накопителя для хранения рабочих программ и данных на базе флэш-памяти объемом от нескольких единиц до нескольких сотен мегабайт информации.

4. Дополнительные платы для решения задач графического представления информации, поддержки функциональных клавиатур и жидкокристаллических или люминесцентных панелей, работы с механическими накопителями на жестких и гибких магнитных дисках и т.д.

При таком подходе к созданию системы на базе МСУВТ В-10 появляется реальная возможность формировать вычислительное ядро необходимой мощности и состава для каждого из классов решаемых задач, использовать весь имеющийся объем конструкторских наработок, всю номенклатуру аттестованных плат ввода-вывода, учитывать все современные веяния в области микропроцессорной и коммуникационной техники без длительного и ресурсоемкого этапа разработок.

По мере возрастания сложности задач возлагаемых на контроллер нижнего уровня при управлении технологическими процессами возрастала роль технолога, знающего все тонкости технологии, осуществляющего постановку задач, и программиста, реализующего эти задачи в виде программного кода. Зачастую сроки и успех реализации конкретного проекта автоматизации оказывались в прямой зависимости от квалификации конкретного программиста. Сам же программист оставался единственным человеком способным разобраться в своей программе, внести в нее изменения или продолжить разработку.

Учитывая влияние человеческого фактора, сложности взаимоотношений программистов и технологов, текучесть программист-

ских кадров легко понять почему каждая новая программа разрабатывалась практически с нуля, с минимальным учетом аналогичного опыта и максимальной затратой времени и сил. Особенно остро вопрос о преемственности, унификации и возможности проверки программного кода встал при переходе к РС-совместимым архитектурам, где «творчество» программиста ограничивалось только его фантазией.

К счастью эта проблема оказалась не типично российской, с ней в той или иной мере столкнулись все промышленно развитые страны. В результате было разработано семейство технологических языков программирования, позволяющих «строить» требуемые технологические программы из стандартных единожды написанных, отлаженных и документированных блоков непосредственно технологами практически без участия программистов. Роль программистов сводилась к написанию и отладке «строительных блоков» и драйверов периферийного оборудования.

Деятельность по разработке языков программирования систем автоматизации технологических процессов была стандартизирована Международной Электротехнической комиссией (ИТС) в виде стандарта IEC-1131. Стандарт определяет пять основных типов технологических языков, из которых наибольший интерес для применения в контроллерах нижнего уровня представляет язык функционально-блоковых диаграмм FDB (Function Block Diagram), берущий свое начало от языка релейных схем и решающий основные проблемы программирования алгоритмов управления технологическими процессами. Этот язык позволяет естественным для инженера-технолога способом, «собирая» программу из «элементарных кубиков», описать алгоритм управления технологическим процессом любой сложности, используя только готовые библиотечные блоки. В качестве библиотечных блоков могут использоваться математические и логические функции, команды управления ходом вычислений, элементарные алгоритмы обработки сигналов (П, ПИ, ПИД – регулирование, цифровая фильтрация и т.п.), специализированные процедуры.

При выборе конкретной реализации технологического языка для штатного использования в рамках РС-совместимого варианта МСУВТ В-10 в расчет принимались следующие соображения:

1. Максимальная простота реализации и доступность для неподготовленного пользователя.
2. Русскоязычный интерфейс и пользовательская документация.
3. Поддержка пакетом всей номенклатуры модулей ввода-вывода, разработанных в рамках В-10, и возможность добавления драйверов для вновь разрабатываемых или модифицируемых плат.

4. Поддержка стандартных протоколов сетевого обмена и распределенных сетевых архитектур.

5. Доступная цена и возможность предоставить пользователю функционирующую демонстрационную версию для ознакомления с пакетом.

В результате была выбрана система объектного визуального программирования отечественной разработки **UltraLogic**, удовлетворяющая всем вышеперечисленным критериям. Структура системы программирования **UltraLogic** приведена на рис.1 [2].

Система максимально ориентирована на конечного пользователя, причем не специалиста-программиста, а на инженера-технолога, хорошо понимающего суть конкретных процессов управления, но не всегда владеющего навыками программирования. При использовании системы программирования **UltraLogic** контроллер В-10, с точки зрения конечного пользователя, превращается в «черный ящик», связанный с объектом управления посредством набора «каналов» ввода-вывода, внутренняя организация и принципы аппаратного функционирования которых более интереса не представляют. Каждому «каналу» ставится в соответствие некоторая переменная, при определении которой задается диапазон значений ввода-вывода, функция преобразования отсчетов ЦАП-АЦП в физические величины, процедура калибровки и усреднения измерений. Вместо детального программирования операций ввода-вывода для выдачи и приема сигналов, пользователь просто присваивает значения переменным при выводе информации или считывает их значения при вводе.

Система программирования **UltraLogic** состоит из двух частей:

1. Среда программирования, имеющей графический интерфейс, функционирующей в среде Windows и включающей графические средства подготовки программ, компиляторы, средства интерактивного диалога и настройки, менеджер проектов, инструменты для автономной отладки программного кода.

2. Системы исполнения, функционирующей в среде DOS на целевом РС-совместимом контроллере, включающей драйверы поддержки сетевого обмена, драйверы плат ввода-вывода, скомпилированный код технологической программы.

Сетевые ресурсы автоматически включаются в состав исполняемого кода и могут поддерживать сетевой обмен как по RS-485, так и с использованием протоколов IPX/SPX по локальной сети.

Организация сетевого обмена информацией в пакете **UltraLogic** полностью аналогична работе с каналами ввода-вывода локального контроллера, единственное отличие состоит в добавлении к имени переменной префикса, определяющего сетевой адрес контроллера с

которым эта переменная связана. Контроллеры, образующие сеть, разбиваются на активные устройства, использующие при работе собственные и сетевые данные и пассивные, поставляющие информацию в сеть. Инициатором сетевого обмена всегда выступает либо активное устройство, либо программа верхнего уровня, запрашивающая данные от контроллерной сети.

В качестве механизмов для передачи данных на верхние уровни иерархических систем управления может быть использован один из следующих механизмов:

1. Прямой вызов функций драйвера сети *UltraLogic* из программы верхнего уровня.

2. Использование прикладной программой встроенных в операционную среду Windows протоколов сетевого DDE-обмена, поддерживаемых сетевым драйвером *UltraLogic*.

3. Применение для связи с современными SCADA - системами верхнего уровня механизмов OPC - серверов, поставляемых с пакетом *UltraLogic*.

Собственно программирование осуществляется в среде разработки с помощью графического редактора и представляет собой создание набора функционально-алгоритмических блоков, связи между которыми определяют порядок их выполнения и передаваемую между блоками информацию. По своему стилю программирование очень напоминает работу в САПР PCAD, что делает *Ultalogic* очень привлекательным инструментом для использования разработчиками оборудования при написании тестовых и отладочных программ.

Разрабатываемая программа может включать в себя произвольное количество подпрограмм и, при необходимости, фрагментов кода, написанного на языках низкого уровня, порядок выполнения которых задается пользователем. Более того, любая законченная программа может быть оформлена в виде функционального блока, сохранена и использована при разработке новых проектов, чем обеспечивается преемственность программирования. В результате инженер-технолог получает возможность создания и накопления собственной специализированной библиотеки функциональных блоков-кубиков, радикально облегчается чтение и понимание программ.

Для поддержки имеющейся в рамках В-10 номенклатуры модулей цифрового и аналогового ввода-вывода на языке С были написаны драйверы, объектный код которых был интегрирован в среду разработки *UltraLogic*.

Готовый проект компилируется с драйверами периферийного оборудования и сетевой поддержки. Полученный исполняемый код

минимального размера легко размещается на флэш-диске платы ПМИ-06 контроллера В-10.

На долю профессионального программиста остаются две основные задачи:

1. Написание на привычных ему языках программирования (Ассемблер, Си, Паскаль и т.п.) и совместная с разработчиком аппаратуры отладка драйверов для новых модулей ввода-вывода.

2. Разработка внешних фрагментов кода добавляемых в программу на функционально-блоковом языке для выполнения задач требующих повышенного быстродействия или специальных вычислений, не реализуемых с помощью штатного набора алгоритмических блоков.

С помощью встроенного в среду разработки отладчика имеется возможность как автономной отладки полученных программ (в режиме эмуляции без реального контроллера), так и отладки на реальном оборудовании, используя пошаговый режим выполнения, визуализацию текущих значений переменных, задание точек останова программ и т.п.

Процессу отладки доступны не только значения переменных одного контроллера, а вся совокупность сетевых данных. Имеется возможность проводить как диагностику работоспособности всей контроллерной сети, так и связи с каждым из контроллеров.

Одним из интересных применений пакета *UltraLogic* является возможность моделирования сложных процессов управления без использования реального оборудования (режим эмуляции) и отображения полученных результатов в табличном или наглядном графическом виде.

В рамках теоретических проработок был заново переработан и существенно расширен набор библиотечных модулей используемых в системах регулирования.

Этап моделирования сложных систем регулирования, таких как векторные системы управления приводами переменного тока, системы управления магнитным подвесом и многие другие, является необходимым перед этапом создания программного обеспечения. Только при такой последовательности выполнения работ можно реализовать системы с высоким качеством регулирования.

В пакете *UltraLogic* имеется раздел библиотеки, содержащий функциональные модули для построения систем регулирования. Несмотря на это, для построения систем и их моделирования была разработана новая библиотека функциональных модулей по следующим причинам:

1. Существующие регуляторы реализованы с некорректным ограничением выходных величин.

2. В представленных модулях нет единообразия построения с точки зрения задаваемых параметров.

3. Существующие виды регуляторов не могут обеспечить все требования при построении и моделировании сложных систем регулирования.

Корректное ограничение выходной величины регулятора предполагает, что интегральная составляющая на момент ограничения сохраняется и при выходе из ограничения. В связи с этим следует отметить, что аналоговые регуляторы на базе операционных усилителей с ограничением выходной величины за счет стабилизаторов или опорных напряжений в цепи обратной связи не обеспечивают корректное ограничение. В этом случае интегральная составляющая регулятора в момент ограничения не сохраняется до выхода из ограничения, а становится равной напряжению ограничения (при длительном ограничении).

При построении цифровых регуляторов не должны использоваться термины «коэффициент интегрирования», «коэффициент дифференцирования», и соответствующие им параметры не должны использоваться как входные параметры регулятора. Для широкого круга инженеров, работавших ранее с аналоговой техникой, более понятны такие термины как «постоянная времени интегрирования», «постоянная времени дифференцирования», «постоянная времени инерционного звена». Единственным новым термином, который вводится при построении цифровых регуляторов, является «шаг квантования по времени dT ». Эти параметры должны быть входными, а коэффициенты алгоритма должны вычисляться при инициализации функционального модуля.

При построении сложных систем регулирования нельзя ограничиться только I , PI , PID -регуляторами. Необходимо иметь регуляторы, реализующие различные функции по заданию и по каналу обратной связи. Например может быть I или PI -регулятор по входу и соответственно PI или PID -регулятор по каналу обратной связи с общим ограничением выходной величины. Эти регуляторы могут быть использованы для адаптации замкнутого контура к меняющимся параметрам объекта или для снятия противоречий между требованием иметь заданное быстродействие и хорошее демпфирование и в то же время малую постоянную времени интегрирования замкнутого контура регулирования. В зарубежной практике название PID -регулятор при более внимательном рассмотрении может означать реализацию PI -функции по заданию и PID -функции по каналу обратной связи.

Для исследования сложных систем регулирования необходимо иметь в виде функциональных модулей объекты систем регулирования, часто встречающиеся в работах института. Это может быть привод постоянного или переменного тока, система магнитов (магнитный подвес), двухмассовая система с переменной жесткостью, зазором и конечным коэффициентом демпфирования и т.д.

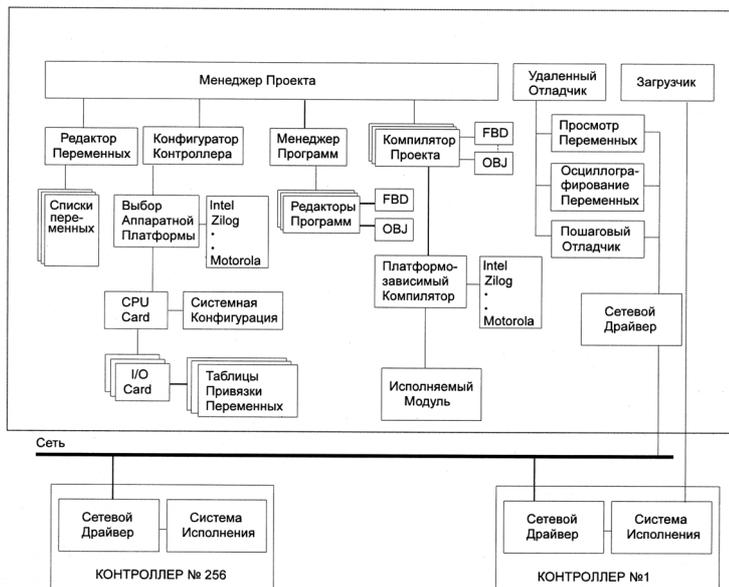


Рис. 1. Система программирования *UltraLogic*

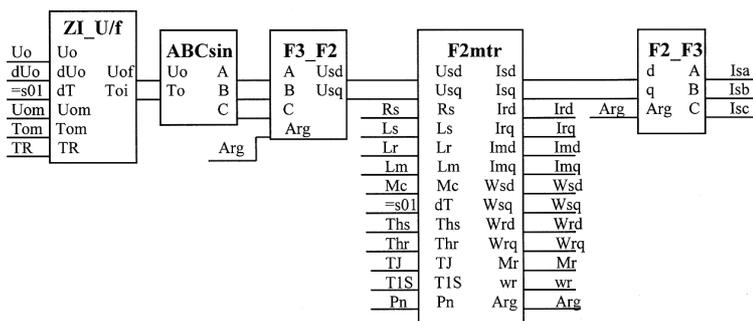


Рис. 2. Схема соединения модулей для пуска асинхронного двигателя от датчика интенсивности при $U/f = \text{const}$

Таким образом, следует наращивать не только модули регуляторов и фильтров, но и модули часто встречающихся объектов, что позволит быстрее исследовать системы регулирования, выбрать оптимальные регуляторы и перенести опыт наладки модели на реальный объект.

Разработанные программные модули новой библиотеки для моделирования и реализации систем регулирования имеют следующие характеристики:

1. Параметры регуляторов задаются в том же виде, как и в аналоговых системах регулирования с введением дополнительно шага квантования по времени.

2. Регуляторы обеспечивают корректное ограничение выходной величины.

3. Все регуляторы имеют признак верхнего и нижнего ограничения, что обеспечивает контроль на размыкание системы при исследовании линейных систем или использование этих сигналов для коррекции работы системы.

4. Модули регуляторов и фильтров позволяют изменять шаг квантования по времени и тем самым реализовывать одновременно быстрые и медленные контуры регулирования (например регулирование тока двигателя и контроль его нагрева), учесть влияние шага квантования на динамику переходных процессов.

5. Функциональные модули реализуются как по методу прямоугольников, так и по методу трапеций. Это отражает тот факт, что для быстродействующих систем регулирования в целях сокращения времени выполнения программы функциональные модули реализуются по методу прямоугольников. Для моделирования объекта можно использовать модули, реализованные по методу трапеций.

6. Функция дифференцирования во всех модулях выполняется совместно с фильтрацией.

Например PID - регулятор имеет передаточную функцию:

$$W(p) = K + \frac{1}{T_i p} + \frac{T_d p}{1 + T_f p},$$

где K - коэффициент пропорциональной части; T_i - постоянная времени интегрирования; T_d - постоянная времени дифференцирования; T_f - постоянная времени фильтра.

1. В зависимости от способа реализации модули имеют в своем обозначении индексы P (прямоугольник) или T (трапеция).

2. Модуль с набором математических функций с ограничением выходной величины определяется как регулятор и имеет в своем обозначении индекс R .

3. Регуляторы реализуют любые сочетания функций, определенных в их названии.

Ниже приведен список модулей новой библиотеки.

I T, I P - интегратор.

DF T, DF P - дифференциатор с фильтром.

IR T, IR P - интегральный регулятор.

PIR T, PIR P - пропорционально-интегральный регулятор.

PIDFR T, PIDFR P - пропорционально-интегрально-дифференциальный регулятор с фильтром на *D*-части.

PI_PIDFR T, PI_PIDFR P - пропорционально-интегральный регулятор по входу и пропорционально-интегрально-дифференциальный регулятор с фильтром на *D*-части по каналу обратной связи.

OGR - модуль ограничения входной величины.

Имеет уставки верхнего и нижнего ограничения на входе, признаки верхнего, нижнего и просто ограничения на выходе. Используется при построении регуляторов и других модулей.

NOTCH - заградительный (режекторный) фильтр.

Для качественной фильтрации реализуется по методу трапеций. Передаточная функция в аналоговом эквиваленте имеет вид:

$$W(p) = \frac{p^2 + \omega_0^2}{p^2 + 2\xi\omega_0 p + \omega_0^2} .$$

Входными параметрами являются коэффициент демпфирования ξ и частота недемпфированных колебаний ω_0 .

BNDPS - полосовой фильтр.

Реализуется на базе заградительного фильтра. Передаточная функция в аналоговом эквиваленте имеет вид:

$$W(p) = \frac{2\xi\omega_0 p}{p^2 + 2\xi\omega_0 p + \omega_0^2} .$$

Asin, Bsin, Csin - генераторы синусоидальных колебаний для фаз *A, B, C*.

Реализованы на базе консервативного звена с переводом в колебательное при стремлении выходного сигнала превысить заданное амплитудное значение.

ABCsin - генератор синусоидальных сигналов для фаз *A, B, C* с использованием модуля вычисления синуса общей библиотеки.

AshimS, BshimS, CshimS - ШИМ-синусоиды для фаз *A, B, C*.

Могут быть использованы для анализа влияния частоты коммутации на токи в машинах переменного тока.

GENP - генератор пилообразного сигнала.

Используется при построении модулей ШИМ-синусоид.

ZI - задатчик интенсивности входного сигнала.

Темп изменения сигнала определяется заданием максимального входного сигнала и временем его достижения.

F2mtr - модель двухфазного асинхронного двигателя с короткозамкнутым ротором в системе координат ротора d, q .

Модель не учитывает влияние насыщения машины и глубины паза ротора при меняющейся частоте на параметры машины и это будет выполнено при совершенствовании модели. В то же время при реализации систем векторного управления возможна адаптация замкнутых контуров регулирования к меняющимся параметрам двигателя при правильном выборе типов регуляторов.

Модель может быть использована для анализа переходных процессов при пуске асинхронного двигателя от сети с использованием модулей **ABCsin, F3-F2**, при пуске двигателя от задатчика интенсивности с поддержанием $U/f = const$ с использованием модулей **ZI_U/f, ABCsin, F2_F3** и самостоятельно при построении систем векторного управления с поддержанием потокосцепления ψ_d , регулирования тока I_q и скорости.

Выходными параметрами являются токи статора, ротора и цепи намагничивания по осям d, q , момент ротора, частота вращения ротора и электрический угол поворота. Для получения токов трехфазной машины необходимо на выход включить модуль **F2_F3**.

На рис.2 в качестве примера приведена схема соединения функциональных модулей для исследования переходных процессов при разгоне асинхронного двигателя от задатчика интенсивности с поддержанием постоянного отношения напряжения к частоте.

F3_F2 - модуль приведения параметров трехфазной машины (напряжений, токов) к параметрам двухфазной машины в системе координат ротора d, q . На вход подаются напряжения (токи) трехфазной машины и электрический угол поворота.

F2_F3 - модуль приведения параметров двухфазной машины в системе координат ротора d, q (напряжений, токов) к параметрам трехфазной машины. На вход подаются напряжения (токи) двухфазной машины по осям d, q и электрический угол поворота.

I_PWS - интегратор со сбросом при достижении выходной величиной значения 360° (полного оборота). Используется модулем **F2mtr** для вычисления электрического угла поворота.

ZI_U/f - задатчик интенсивности частотного сигнала с поддержанием постоянства отношения амплитуды к частоте. Выдает амплитуду и период частоты на вход модуля **ABCsin**. Для коррекции внешней характеристики асинхронного двигателя на низких частотах предусмотрено введение дополнительного сигнала напряжения dU при $f = 0$.

Таким образом, при использовании в качестве средства для создания технологических и управляющих программ системы программирования *UltraLogic* разработчик получает возможность абстрагировавшись от конкретного аппаратного состава используемого контроллера и технических трудностей программирования сосредоточить усилия на практической реализации своих задач. С другой стороны, использование системы программирования *UltraLogic* в качестве штатного средства программирования для МСУВТ В-10 превращает поставляемые НИИ ВНИИЭМ контроллеры в законченный программно-аппаратный комплекс, имеющий повышенную привлекательность для потенциальных пользователей.

ЛИТЕРАТУРА

1. Применение РС-совместимых модулей в промышленных контроллерах МСУВТ В-10 / Сазонов С.В. // Приборы и системы управления. 1997. №4.
2. *UltraLogic* – система подготовки программ для промышленных контроллеров / Шакиров С., Биосов Р., Якубович Б., Журавлев В. // Современные технологии автоматизации. 1997. №3.