

ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА

УДК 681.5

С. М. Вертешев, В. А. Коневцов, И. А. Полетаев

ОБРАТНАЯ ПОЛЬСКАЯ ЗАПИСЬ В КОМПЛЕКСЕ ПСПСЦУ

Опыт создания языка высокого уровня, встроенного в язык блочного проектирования схем цифровых систем автоматического управления.

Ключевые слова: цифровая система автоматического управления, язык блочного проектирования, запись символьного описания схем, блочный редактор, постфиксная запись.

Грамматика языковых средств комплекса ПСПСЦУ (блочного проектирования, символьного описания схем, блочного редактирования, (модулей) вычисления выражений) подробно описана в [1, 2]. Язык блочного проектирования [1, С. 61] схем предназначен для использования на главном этапе проектирования цифровой САУ — постановке задачи управления и разработке функциональных схем и схем соединений [1, С. 197] квалифицированными проектировщиками и наладчиками. Грамматика символьного описания схем используется монтажниками при преобразовании схем соединений в кроссировочные таблицы и схемы [1, С. 198] на этапе технического проектирования. Блочный редактор схем используется квалифицированными проектировщиками и наладчиками на этапах рабочего проектирования, отладки и сдачи в эксплуатацию, эксплуатационным персоналом во время промышленной эксплуатации и модернизации цифровой САУ [1, С. 206–214]. Функциональные модули вычисления выражений [1, С. 67–76, 103–106] используются на всех этапах жизненного цикла цифровой САУ от постановки задачи управления до модернизации цифровой САУ. Особенность комплекса ПСПСЦУ состоит в относительно новом подходе (прерванном известными обстоятельствами 1991 года) к созданию подобных систем проектирования цифровых САУ. Язык комплекса ПСПСЦУ обладает следующими основными отличительными свойствами:

1) высокая изобразительность и наглядность языка блочного проектирования в форме традиционных схем систем автоматического управления;

2) наличие функциональных модулей реализации всех основных задач в цифровых системах автоматического управления с явным заданием всевозможных связей в системе управления; в распоряжение пользователя комплекса ПСПСЦУ предоставляется средство реализации с помощью микропроцессорной техники не только отдельных задач, но и целого множества математических методов теории автоматического управления;

3) встроенный язык высокого уровня, грамматически полностью совместимый с языком блочного проектирования и не уступающий по функциональным возможностям формульным языкам (Фортран, Бэйсик), но отличающийся от них высокой изобразительностью в форме блочных схем.

Основные конструктивные характеристики в комплексе ПСПСЦУ, как и при создании микросхем разработчиками-схемотехниками, при проектировании цифровой САУ: не используются метки, операции условных и безусловных переходов, ветвления, выбора, цикла, составные операторы, макросы (действия), которые применяются в языках программирования всех уровней. Все эти операции скрыты в подпрограммах функциональных модулей и операций в стеке при разработке собственно комплекса ПСПСЦУ. Отсутствуют рекурсия, перегрузка, наследование при разработке подпрограмм функциональных модулей и операций.

Технологические характеристики — комплекс ПСПСЦУ является чисто блочным языковым средством без каких-либо добавлений и описаний с помощью других языковых средств типа «усеченного» Ассемблера (IL), «усечённого» Си или Паскаль (ST), языков подобных SFC (Sequential Function Chart), CFC (Continuous Flow Chart), LD (Ladder Diagram), FCL (Fuzzy Control Language) и т. д. При этом сохраняется возможность проектирования всего многообразия задач цифрового управления, формально обеспечиваемая наличием многих языковых средств описания задач управления. Преобразующие средства комплекса ПСПСЦУ (транслятор и блочный редактор) не генерируют машинные команды. Они создают байтовые массивы базы данных, которые могут обрабатываться на любом микропроцессорном комплексе, имеющем исполнительную систему комплекса ПСПСЦУ (интерпретатор, программный интерфейс, блок обработки прерываний, библиотека подпрограмм функциональных модулей с библиотекой подпрограмм операций, реализованные для процессора соответствующей цифровой УВМ). Это означает отказ от процесса компиляции при проектировании цифровых систем автоматического управления, упрощение трансляции системы управления, структура которой представляет собой фактически последовательность вызовов подпрограмм (CALL(), CALL(), ... , CALL()).

Технико-эксплуатационные характеристики — подход к созданию комплекса ПСПСЦУ позволяет значительно сокращать затраты на основные процессы жизненного цикла программного средства автоматизации. Вместо разработки, эксплуатации и сопровождения пяти и более трансляторов, редакторов, библиотек и т. д. по стандарту МЭК, ведётся разработка одного транслятора, редактора, библиотеки ... простой архитектуры комплекса ПСПСЦУ. Сокращение затрат на сопровождение проектируемой цифровой САУ на базе комплекса ПСПСЦУ по сравнению с затратами на сопровождение этой же САУ средствами стандарта МЭК. Специалист по автоматизации при использовании комплекса ПСПСЦУ имеет возможность решать свои задачи от постановки до внедрения и сдачи в промышленную эксплуатацию без участия квалифицированного программиста, кроме того имеется возможность эффективного разделения труда между различными специалистами, участвующими в процессе автоматизации технологического процесса (проектировщики, монтажники, наладчики, «киповцы» — персонал эксплуатации САУ, технологи, математики).

Комплекс ПСПСЦУ относился к категории перспективного и серийно выпускаемого оборудования [3, 4] ГСП СССР как система автоматизированного проектирования цифровых систем автоматического управления (САПР цифровых САУ). Этот комплекс прошёл успешные промышленные испытания на многих заводах

СССР. Основной областью применения комплекса ПСПСЦУ является автоматизация сложных промышленных объектов. Основными пользователями комплекса ПСПСЦУ могут быть проектировщики цифровых САУ, владеющие математическими методами различных разделов теории автоматического управления.

Почти во всех известных программных средствах проектирования цифровых САУ имеются возможности реализации различных математических зависимостей, которые неизвестны с ранних этапов проектирования цифровых САУ до ввода в действие и промышленной эксплуатации и которые пользователь вынужден проектировать самостоятельно. Чтобы обеспечить пользователю такую самостоятельность наряду с языками блочного проектирования (или наряду с трафаретами обработки сигналов) в ряд систем автоматизации включались средства описания цифровых САУ от мнемокодов до языков «высокого» уровня и специализированных языков. Так, в системе GPCP имеются шесть подобных средств [1, 2]: бланки переменных, регулирования, унифицированного выражения, встроенного ассемблера, констант и начальных значений переменных, программ Ассемблера и Фортрана. В системе SIMATIC S7 используется языки AWL (IL) список инструкций, KOP (LD) релейно-контактные схемы, FBS (FBD) блочное проектирование, текстовый язык SCL (ST), AS (SFC). При использовании этих и подобных средств автоматизации предполагается наличие у пользователя навыков программирования на упрощённом ассемблере. Таким средством можно достигнуть наибольшей гибкости в организации каналов обработки сигналов и их модификации (кроме обработки прерываний, программирования УСО и т. д.). Однако это влечёт значительное снижение производительности труда в проектировании цифровых САУ.

Иной подход к решению задачи повышения производительности труда в проектировании цифровых САУ предложен в своё время системой SIMAT [1, 2] предоставлением в распоряжение пользователя так называемого свободно программируемого многофункционального модуля, позволяющего встраивать различные математические зависимости в работающую цифровую САУ без компиляции. Этот подход представляется достаточно рациональным, поскольку он не вынуждает пользователя работать на двух языках (например, на языке блочного проектирования и на текстовом языке), различающихся по наглядности представления схем цифровых САУ, по синтаксису и семантике, по организации подстановки фактических параметров. Такой многофункциональный модуль является, по существу, встроенным языком языка блочного проектирования. С помощью этого модуля можно практически достигнуть того же эффекта в проектировании выражений, что и, скажем, с помощью Фортрана. Применяя такой модуль пользователю надо всего лишь поменять «акцент». Подобный подход к реализации математических зависимостей, как испытано практикой [3, 4], позволяет повысить производительность труда пользователя в «разы» за счёт исключения необходимости переключения с одной грамматики на другую (третью, четвёртую, пятую и т. д.) за счёт применения не двух и более грамматик, а за счёт применения «диалекта» языка блочного проектирования.

Исходя из изложенных соображений, в комплексе ПСПСЦУ принят аналогичный, но более эффективный подход к созданию многофункционального модуля

как встроенного языка в язык блочного проектирования. Грамматики языка блочного проектирования, символьного описания схем и блочного редактора подробно описаны в [1, 2]. Здесь большее внимание уделяется разработке одного или нескольких многофункциональных и других модулей языка блочного проектирования схем с использованием различных форм записи математических выражений [1, С. 67–76] и топологий функциональных модулей. Для создания функциональных модулей комплекса ПСПСЦУ используются различные формы записи выражений. Различают три формы записи выражений: функциональная запись (префиксная форма записи), выражения, содержащие скобки (инфиксная форма записи) и выражения в бесскобочной форме записи. Первые две формы широко известны и использовались в комплексе ПСПСЦУ для разработки большинства стандартизированных функциональных модулей.

Выражения в бесскобочной форме записи имеют две нотации: префиксную (знак операции предшествует связанным с ним операндам, польская запись) и постфиксную (знак операции следует за связанными с ним операндами, обратная польская запись, нотация польского логика Яна Лукашевича). Для достижения «свободного программирования» в разработке комплекса ПСПСЦУ применялась постфиксная нотация, основанная на многоместных операциях там, где это имеет смысл.

При использовании в выражениях n -местных операций постфиксная форма записи определяется следующим образом:

- 1) Всякая переменная или константа есть выражение.
- 2) Если F есть символ одноместной операции, а A есть выражение, то $AF1$ есть выражение.
- 3) Если F есть символ двуместной операции, а A и B есть выражения, то $ABF2$ есть выражение.

....

$N+1$) Если F есть символ N -местной операции, а A, B, \dots, E есть совокупность из N выражений, то $AB \dots EFN$ есть выражение.

$N+2$) Других выражений не существует.

Характерной особенностью постфиксной формы записи является возможность применения для интерпретации выражений механизма обработки данных в стеке (магазине). Выражения, представленные в постфиксной форме записи, можно рассматривать как программу, содержащую команды трёх видов:

- 1) чтения операндов $AB \dots E$ из адресуемой памяти и записи их значений в вершину стека;
- 2) вызова операции F с выполнением соответствующей подпрограммы при последовательном извлечении значений операндов $E \dots BA$ в обратном порядке из вершины стека и заключительной записью в вершину стека результата;
- 3) чтения результата из стека и записи результата в адресуемую память.

При использовании польской обратной записи отпадает необходимость учёта старшинства операций. Это значительно упрощает обработку выражений трансляторами и вообще необходимость в самих трансляторах подготовки обработки выражений. Трансляторы можно заменить одним или несколькими много-

функциональными модулями, использующими стековые операции. Эти операции позволяют освободиться при проектировании от скобочных выражений, а, следовательно, от преобразования таких выражений в стековую форму обработки сигналов, от процесса компиляции при проектировании цифровой САУ. В ряде языковых средств механизм стековой обработки сигналов либо не используется, либо используется при преобразовании компиляторами скобочных выражений в бесскобочные, но не всегда непосредственно доступен пользователю. Одно их исключений представляет собой язык Форт, в грамматике которого полностью принята постфиксная форма записи выражений, состоящих в основном из одно- и двухместных операций. Полное определение польской обратной записи выражений, т. е. возможность реализации n -местных операций без определения новых слов (выражений в терминах Форты) в языке Форт не используется. Так, выражение $Y = A + B + C + D + E$, записанное средствами комплекса ПСПСЦУ, т. е. в виде $Y = ABCDE + 5$ будет на 2 символа короче выражения $Y = AB + C + D + E +$, написанном на языке Форт. Длина записей средствами комплекса ПСПСЦУ оценивается относительно длины записей выражений на языке Форт по следующим правилам:

- если используется выражение с одно- или двухместной операцией, то длина записи на Форте короче на один символ;
- если используется выражение с трёхместной операцией, то длины записей одинаковы;
- если используется выражение с n -местной операцией при $n > 3$, то длина записи на Форте больше на величину $n - 3$.

Программа, написанная средствами комплекса ПСПСЦУ, будет не длиннее программы, написанной на Форте при $n \geq 3$, но потребует большую глубину стека. В средствах комплекса ПСПСЦУ приняты следующие основные соглашения:

- 1) все математические зависимости, создаваемые пользователем, представляются в постфиксной форме записи выражений (обратная польская запись);
- 2) в последовательности символов записи указывается, кроме операндов и операций число операндов не независимо от того, является ли данная операция одноместной, двухместной, трёхместной и т. д. в данном контексте применения, причём число операндов n -местной операции не должно превышать 255;
- 3) интерпретация выражений осуществляется двумя функциональными модулями вычисления выражений: модулем МВВ с одним или двумя выходами (второй инверсный выход для выражений с логическим результатом) и модулем вычисления выражений многовыходным МВВМ;
- 4) модуль МВВ предназначен в основном для проектирования однозначных линейных и нелинейных зависимостей;
- 5) модуль МВВМ предназначен для проектирования как однозначных, так и многозначных линейных и нелинейных зависимостей, систем линейных и нелинейных функций, для чего в составе стековых операций предусмотрены операции размножения вершины стека, прямого доступа в стек, сдвига стека «вниз»;
- 6) все выражения представляются в блочной форме в виде схем или отдельных блоков с указанием в левом поле каждого блока модуля операндов и числа

операндов (на схемах соединений) или их адресов (на кроссировочных схемах) выражений, а в правом — результатов выражений.

Применение функциональных модулей МВВ и МВВМ особо предпочтительно при реализации нестандартных и нетрадиционных различных математических зависимостей [2, С. 226–245].

Основное отличие применения операций и модулей комплекса ПСПСЦУ заключается в ориентации на специалистов в области автоматизации технических систем. Основное отличие применения операций и модулей по стандарту МЭК заключается в ориентации на специалистов в области программирования. Особое отличие осуществления операций комплекса ПСПСЦУ — в реализации постфиксной нотации [1]. Пользователем этих операций является «непрограммирующий» специалист (проектировщики, монтажники, наладчики, специалисты по КИПиА, технологи, математики), который для программирования должен научиться всего лишь писать все математические выражения в постфиксной форме записи [1] в блочной форме. Эти записи и есть программы, которые «прямо» используются, интерпретируются без предварительного преобразования, то есть компиляции. Тем самым сохраняется контролируемость реализации выражений специалистом по автоматизации.

Пользователем операций по стандарту МЭК 61131-3 является «классический» программист. Специалист (проектировщики, монтажники, наладчики, специалисты по КИПиА, технологи, математики) в случае применения операций по стандарту МЭК теряет контролируемость реализации после передачи своего математического выражения «классическому» программисту, кодировщику. Особое отличие осуществления функциональных модулей комплекса ПСПСЦУ — в реализации всё более сложных и стандартизуемых математических зависимостей различных разделов теории автоматического управления. Пользователем функциональных модулей комплекса ПСПСЦУ, как и пользователем операций комплекса ПСПСЦУ, является «непрограммирующий» специалист (проектировщики, монтажники, наладчики, специалисты по КИПиА, технологи, математики).

«Устойчивой тенденцией промышленности, выпускающей интегральные схемы [5, С. 380], является создание всё более сложных, более насыщенных кристаллов... Интегральные схемы малой степени интеграции (МИС) заменяются интегральными схемами средней степени интеграции (СИС), большой степени интеграции (БИС) и сверхбольшой степени интеграции (СБИС)». Дальнейшим повышением степени интеграции элементов микроэлектроники являлось создание нового класса интегральных микросхем — микропроцессоров (МП), микропроцессорных комплектов (МПК), микро-ЭВМ, программируемых логических контроллеров (ПЛК). К МИС [5, С. 395] отнесены схемы, содержащие до 10 элементов и компонентов булевой алгебры (И, И-НЕ, ИЛИ, ИЛИ-НЕ, И-ИЛИ, И-ИЛИ-НЕ и т. д.). К СИС [6, С. 5] отнесены схемы, содержащие до 100 элементов и компонентов реализации более сложных схем (шифраторы и дешифраторы, мультиплексоры и компараторы, одноктактные и многотактные устройства, триггеры и регистры, счётчики и сдвигающие регистры различной разрядности, «пакетности», «расширяемости», «многоместности»). К БИС отнесены схемы, содержащие до 1000 элементов и ком-

понентов реализации более сложных схем... МПК (микропроцессоры, таймеры, параллельные и последовательные интерфейсы, каналы прямого доступа в память, контроллеры обработки прерываний) относятся к «полуфабрикатам» по сборке микро-ЭВМ и ПЛК, а последние — цифровыми программируемыми средствами автоматизации технологических процессов. Схемотехника постоянно двигалась прогрессируя от разработки и применения МИС к СИС, к БИС, к СБИС, ..., к МПК, к микро-ЭВМ,

В программировании (применении микропроцессорной техники в автоматизации технологических процессов) наблюдается тенденция по принципу «шаг вперёд, два шага назад». «Шагом вперёд» было стремление разработчиков САПР цифровых САУ (системы ADC и SIMAT фирмы SIEMENS [1, 2], программные комплексы СЦУ и МСЦУ Грозненского НПО ПРОМАВТОМАТИКА [3, 4] ...) учёта при разработке языковых средств терминологии теории и практики управления, терминологии специалистов по автоматизации технологических процессов, традиционную форму наглядного (блочного) представления схем систем управления. Это позволяло эффективное разделение труда между разработчиками цифровой САУ на разных этапах создания АСУ ТП (проектирование, монтаж, наладка, эксплуатация, модернизация, смена технологии). «Двумя шагами назад» было возвращение к «многоязычию» систем GPCP (упрощённый Ассемблер) и VICEPS (язык VPL, упрощённый Фортран). Это означает «смещение языков», т. е. значительное снижение производительности труда при проектировании цифровых САУ сложных объектов управления. С 1993 года стандарт МЭК 61131-3 не сдвинулся в сторону создания более сложных, более насыщенных функционально и архитектурно функциональных модулей и операций, как это наблюдается и по настоящее время в тенденции развития схемотехники. Реализация более сложных функций возложена на плечи пользователей программных систем. Наиболее «заметными шагами» стандарта МЭК 61131-3:2003 по отношению к стандарту МЭК 61131-3:1993 являются требования так называемой «расширяемости» операций (многоместные функции) и явного представления блочных схем средствами языка FBD. Явное представление блочных схем средствами МЭК 61131-3:2003 обеспечивается для «непрограммирующих специалистов» фактически на уровне четырёх типов (вместо нескольких требуемых десятков) функциональных модулей (бистабильные элементы, триггеры фронта логического сигнала, датчики времени, счётчики) языков релейно-контактных и блочных схем (LD и FBD). Попытка проектирования задач других математических методов, [2], влечёт потребность новых функциональных модулей либо новых языков (например, FCL, язык матричных операций, язык полиномиальных уравнений и т. д.), [2, С. 199–245].

Выводы

Характерной особенностью создания комплекса ПСПЦУ являлось применение так называемого метода аппаратной разработки программного обеспечения. Сущность этого метода заключается в следующих ниже определениях.

- 1) При проектировании цифровых САУ не используются метки, операции условных и безусловных переходов, ветвления, выбора, цикла, составные операторы, макросы (действия), которые применяются в языках программирования всех

уровней. «Непрограммирующий специалист» не пользуется этими понятиями, они изъяты у него из обихода. Все эти операции скрыты в подпрограммах функциональных модулей и операций в стеке при разработке собственно комплекса ПСПСЦУ. Отсутствуют рекурсия, перегрузка, наследование при разработке подпрограмм функциональных модулей и операций.

- 2) Стековая реализация всех операций, применение при проектировании цифровых САУ постфиксной формы записи выражений (польской обратной нотации выражений) как явной формы языка блочного проектирования.
- 3) Векторная адресация на основе кроссировочных таблиц монтажников систем автоматического управления.
- 4) Применение всевозможных топологических конфигураций (неизменяемые жёсткие, изменяемые в жёсткой топологии, циклически изменяемые, произвольно изменяемые) функциональных модулей.
- 5) Отсутствие компиляции при проектировании цифровых САУ, упрощённая грамматика, принцип интерпретации при проектировании цифровых САУ.
- 6) Отсутствие неявных связей в схемах (размножение сигналов, альтернативное подключение сигналов, обратная связь в блоке, в схеме, в системе управления).
- 7) Возможность подключения системы к выключенному блоку, к блокам выключенных схем.
- 8) Явное управление устройствами связи с объектом.
- 9) Пакетные реализации функциональных модулей.

На примере реализации общих задач автоматизации и автоматизации биохимического реактора в [2, С. 199–245] очевидна сложность проектов на базе МЭК, с точки зрения «классического» программиста, и простота их реализации, с точки зрения «классического» специалиста по автоматике и телемеханике, при использовании в проектировании комплекса ПСПСЦУ.

Литература

- 1 Коневцов В. А. САПР цифровых САУ. Концепция: Монография. Псков: Издательство ППИ, 2011. 256 с.
- 2 Коневцов В. А. САПР цифровых САУ. Концепция: монография. Изд-е второе, дополн. и испр. Псков: Издательство ПсковГУ, 2012. 307 с.
- 3 Коневцов В. А., Казаченко А. П., Бабаянц А. В. МикроДАТ. Программные средства цифрового управления. М.: ЦНИИТЭИ приборостроения, Каталог ГСП, 1985, Том 4, вып. 5, 6, С. 1–70.
- 4 Коневцов В. А., Казаченко А. П., Литвинова Л. М., Бунин А. Б. Модифицированные средства цифрового управления. М.: Информприбор, Каталог ГСП, 1987, Том 4, вып. 10, 11, 12, С. 1–112.
- 5 Применение интегральных схем: Практическое руководство в 2-х кн. Кн. 1. / Под ред. А. Уильямса. М.: Мир, 1987. 432 с.
- 6 Применение интегральных схем: Практическое руководство в 2-х кн. Кн. 2. / Под ред. А. Уильямса. М.: Мир, 1987. 413 с.

Об авторе (ах)

Вертешев Сергей Михайлович — доктор технических наук, профессор, заведующий кафедрой «Информационные системы и технологии», факультет информатики, Псковский государственный университет, Россия.

Конеццов Владимир Александрович — кандидат технических наук, доцент, доцент кафедры «Информационные системы и технологии», факультет информатики, Псковский государственный университет, Россия.

E-mail: kafedravn-ist@mail.ru

Полетаев Игорь Алексеевич — старший преподаватель кафедры «Вычислительные технологии», факультет информатики, Псковский государственный университет, Россия.

E-mail: kafedravn-ist@mail.ru

S. M. Verteshev, V. A. Konevcov, I. A. Poletaev

REVERSE POLISH NOTATION IN KOMPLEX ПСПЦЦ

Experience in creating high level programming language built in block project. Language of circuits of digital automatic control systems.

Key words: *digital automatic control systems, block project language, notation of symbolic description of circuit, block editor, postfix notation.*

About the author(s)

Verteshev Sergey Mikhaylovich, Doctor of Engineering Sciences, Professor, Head of the Department of Information systems and technologies, Faculty of Computer Science, Pskov State University, Russia.

Konevtsov Vladimir Aleksandrovich, Candidate of Engineering Sciences, Associate Professor, Associate Professor of the Department of Information systems and technologies, Faculty of Computer Science, Pskov State University, Russia.

E-mail: kafedravn-ist@mail.ru

Poletayev Igor Alekseevich, Senior Lecturer of the Department of Computing technologies, Faculty of Computer Science, Pskov State University, Russia.