

3. Пятлин О. А. и др. Проектирование микрoeлектронных цифровых устройств.— М.: Сов. радио, 1977.
4. Ачасова С. М., Бандман О. Л. Матричный метод синтеза комбинационных схем и логических преобразователей конечных автоматов.— Изв. АН СССР. Сер. Техн. кибернетика, 1975, № 6.
5. Гладштейн М. А., Баскаков В. А., Комаров В. М. О реализации переключательных функций на микросхемах средней степени интеграции.— Изв. АН УССР. Сер. Кибернетика, 1980, № 1.
6. Hurst L. S. A Survey of Published Information on Universal Logic Arrays.— Microelectronics and Reliability, 1977, vol. 16, N 6.

Поступила в редакцию 18 февраля 1981 г.

УДК 681.3.06

М. А. БОГДАНОВ

(Москва)

ЯЗЫК КОК ДЛЯ ПРОГРАММИРОВАНИЯ В СИСТЕМАХ КАМАК

1. **Введение.** В настоящей статье описывается язык программирования КОК, а также приводятся сведения об интерпретаторе КОК для ЭВМ типа «Электроника-60». Язык и интерпретатор КОК предназначены в основном для тестирования аппаратуры КАМАК, но могут использоваться для программирования работы некритичных ко времени систем КАМАК.

Язык КОК, подобно языку SICL [1], является самостоятельным языком программирования для КАМАК. В языке КОК имеются средства, позволяющие управлять работой аппаратуры КАМАК, а также вести простейшую обработку данных.

Интерпретатор КОК функционально состоит из трех частей: редактора, транслятора и собственно интерпретатора. Редактор вводит программы в ЭВМ, при этом он позволяет исправлять ошибки, обнаруженные параллельно работающим транслятором, а также производить построчное редактирование текста вводимой программы. Транслятор выполняет трансляцию и запись программы в интерпретируемом формате в оперативную память ЭВМ. Интерпретируемый формат представляет собой последовательность данных для обращения к подпрограммам, содержащимся в собственно интерпретаторе и моделирующим синтаксические элементы языка КОК. Данные для обращения к подпрограмме включают в себя номер подпрограммы и набор параметров, передаваемых в подпрограмму.

Управление работой интерпретатора КОК осуществляется с терминального пульта в интерактивном режиме.

Разработаны три варианта интерпретатора КОК: первый — для ЭВМ «Электроника-60», второй — для ЭВМ СМ-3, СМ-4 с дисковой операционной системой (ДОС), третий — для ЭВМ СМ-4 с операционной системой реального времени (ОС-РВ). Вариант интерпретатора для ЭВМ «Электроника-60» функционирует без операционной системы и может работать на базовом комплекте ЭВМ с объемом оперативной памяти 4 К слов. Этот вариант реализует упрощенную версию языка КОК, в которой не используются обращения к магнитному диску, а второй вариант — полную версию языка КОК. В третьем варианте, в отличие от двух первых, не реализованы средства языка КОК, позволяющие производить обработку прерываний от аппаратуры КАМАК. Все три варианта интерпретатора КОК разработаны для крейт-контроллера КАМАК, входящего в состав ИВК-1, ИВК-2 [2].

2. **Язык КОК.** Описание языка ведется с помощью формул Бэкуса — Наура с использованием расширений Лукаса [3].

2.1. <основной символ> ::= A|B|C|D|E|F|G|H|I|J|L|M|N|O|P|Q|R|S
|T|V|W|X|Z| <десятичная цифра>|:|'|"|"~|+|-|*|/|&|α|=|>|<|[[|(|)|\$
<десятичная цифра> ::= 0|1|2|3|4|5|6|7|8|9

2.2. <число> ::= <десятичное число>|<шестнадцатеричное число>

<десятичное число> ::= {<десятичная цифра>} ...

<шестнадцатеричное число> ::= H|H <шестнадцатеричная форма>

<шестнадцатеричная форма> ::= {<шестнадцатеричная цифра>} ...

<шестнадцатеричная цифра> ::= <десятичная цифра> |A|B|C|D|E|F

Десятичное число может содержать от 1 до 5 десятичных цифр, шестнадцатеричная форма — от 1 до 6 шестнадцатеричных цифр. Шестнадцатеричное число H эквивалентно шестнадцатеричному числу HFFFFFFF.

2.3. <переменная> ::= <переменная адресации КАМАК> |F|R|W|S|L|
<переменная V>

<переменная адресации КАМАК> ::= C|N|A

<переменная V> ::= <прямая переменная V>|<косвенная переменная V>

<прямая переменная V> ::= V<номер переменной>

<косвенная переменная V> ::= V(<номер переменной>)

<номер переменной> ::= <десятичное число>

Переменные C, N, A и F задают соответственно номера крейга, станции, субадрес и функцию КАМАК. Переменные R и W используются при обмене данными между ЭВМ и аппаратурой КАМАК в роли буферных регистров. Во время выполнения команды чтения переменная R получает значение, соответствующее прочитанному коду. Во время выполнения команды записи записываемый код соответствует значению переменной W. Переменные R и W представлены в ЭВМ словами длиной по 24 двоичных разряда.

Переменная S соответствует регистру управления и состояния, а переменная L — регистру маски и запросов крейт-контроллера КАМАК.

Переменные V применяются для хранения произвольных данных. Эти переменные различаются по номерам. Прямые переменные V представляются в ЭВМ словами длиной по 16 двоичных разрядов. Номер косвенной переменной V задает номер прямой переменной, которая используется как индексная переменная. Значение индексной переменной в момент обращения к косвенной переменной определяет номер прямой переменной V, к которой происходит фактическое обращение. Переменные V используют свободную часть оперативной памяти ЭВМ.

2.4. <индикатор> ::= X|Q

Индикаторы соответствуют сигналам X и Q КАМАК.

2.5. <программа> ::= \$ |{<заголовок цикла>|<инструкция сброса цикла>|<конец цикла>|<метка>|<общая инструкция КАМАК>|<разделитель>|<инструкция управления прерываниями>|<разделитель>|<инструкция присваивания>|<разделитель>|<инструкция адресации КАМАК>|<инструкция сдвига>|<разделитель>|<инструкция F>|<инструкция запуска команды КАМАК>|<разделитель>|<инструкция вывода>|<инструкция распечатки>|<разделитель>|<инструкция перехода>|<разделитель>|<инструкция возврата>|<разделитель>|<условная инструкция>|<декларативная инструкция>|<вызов макроопределения>}|<программа>

<разделитель> ::= , |<конец цикла>

<конец цикла> ::=]

2.6. <заголовок цикла> ::= DO <указатель значения>[[[

<указатель значения> ::= <число>|<переменная> |T|P|D

Заголовок цикла [эквивалентен заголовку цикла DO 1 [.

Указатель значения в заголовке цикла задает кратность выполнения цикла, т. е. участка программы, заключенного между данным заголовком и соответствующим этому заголовку концом цикла. Указатели значения T, P, D задают ввод значения во время выполнения программы: T — с клавиатуры терминального пульта, P — с перфоленты, D — с магнитного диска.

Цикл может содержать другие (вложенные) циклы. Уровень вложенности циклов не должен превышать 18.

2.7. <инструкция сброса цикла> ::= J

Инструкция сброса цикла используется при выходе из цикла до завершения его выполнения.

2.8. <метка> ::= <метка перехода>|<метка прерывания>

<метка перехода> ::= M <номер метки перехода>

<номер метки перехода> ::= <десятичное число>

<метка прерывания> ::= D <номер метки прерывания>

<номер метки прерывания> ::= <десятичное число>

Номер метки прерывания может принимать значение в пределах от 1 до 8.

2.9. <общая инструкция КАМАК> ::= Z|C|I|IC

Инструкции Z и C запускают соответствующие сигналы в крейте КАМАК, номер которого определяется значением переменной C. Инструкция I устанавливает сигнал запрета в крейте КАМАК, а инструкция IC снимает его.

2.10. <инструкция управления прерываниями> ::= X|XC|D|DC

Инструкция X разрешает прерывание по отсутствию сигнала X во время выполнения команды КАМАК, а инструкция XC запрещает это прерывание. Инструкция D разрешает, а инструкция DC запрещает любые прерывания от крейта КАМАК.

2.11. <инструкция присваивания> ::= <переменная> = <выражение>

<выражение> ::= <указатель значения>|<указатель значения><знак операции><выражение>

<знак операции> ::= +|-|*|/|&|α

Знак операции «&» задает логическую операцию И, а знак «α» — ИЛИ.

Вычисление выражения производится слева направо без учета старшинства операций, при этом все числа рассматриваются как целые неотрицательные. Все вычисления выполняются над числами длиной по 24 двоичных разряда, а значения переменных, представимых в ЭВМ меньшим числом разрядов, дополняются в старших разрядах нулями. Соответственно при присваивании значений таким переменным старшие разряды выражения теряются.

Пример: V2 = R&H3FF/50 + V2,

2.12. <инструкция адресации КАМАК> ::= <переменная адресации КАМАК> (<указатель значения>)

Эта инструкция используется для присваивания значений переменным адресации КАМАК аналогично инструкции присваивания.

2.13. <инструкция сдвига> ::= <переменная> <знак сдвига> <выражение>

<знак сдвига> ::= >|<

Знак «>» задает логический сдвиг вправо, а знак «<» — влево. Код значения переменной сдвигается на число разрядов, определяемое величиной выражения.

2.14. <инструкция F> ::= F(<указатель значения>)

Инструкция F присваивает значение переменной F и осуществляет запуск команды КАМАК, задаваемой значениями переменных C, N, A, F.

2.15. <инструкция запуска команды КАМАК> ::= EX

Инструкция EX производит запуск команды КАМАК, определяемой значениями переменных C, N, A, F.

2.16. <инструкция вывода> ::= <инструкция печати>|<инструкция вывода на перфоленту>|<инструкция вывода на магнитный диск>

<инструкция печати> ::= <инструкция печати на АЦПУ>|<инструкция печати на терминальном пульте>

<инструкция печати на АЦПУ> ::= OL (<список печати>)

<инструкция печати на терминальном пульте> ::= OT (<список печати>)

<список печати> ::= '<текст>'|<указатель значения> <формат печат-

ти>|/|{'<текст>'<указатель значения><формат печати>|/}, <список печати>
 <формат печати> ::= <пусто>|Н
 <инструкция распечатки> ::= OL|OT
 <инструкция вывода на перфоленту> ::= OP (<список вывода>)
 <инструкция вывода на магнитный диск> ::= OD (<список вывода>)
 <список вывода> ::= <указатель значения>|<указатель значения>,
 <список вывода>

В текстах, выводимых на печать, после сочетаний символов C(|N(|A(|F(дополнительно печатаются текущие значения соответственно переменных C, N, A, F. В текстах можно использовать буквы как латинского, так и русского алфавита.

Пустой формат печати определяет вывод на печать соответствующего значения в виде целого числа со знаком, при этом учитываются только младшие 16 разрядов кода значения. Формат печати Н задает вывод в шестнадцатеричной форме, при этом учитываются 24 разряда кода значения.

Символы, выводимые на печать, предварительно накапливаются интерпретатором в двух специальных буферах печати, один из которых предназначен для печати на АЦПУ, другой — для печати на терминальном пульте. Оба буфера вмещают по 80 символов. Содержимое буфера печати распечатывается в виде одной строки, когда этот буфер заполняется полностью. Инструкции распечатки OL и OT, а также символ «/» в списке печати предназначены для распечатки содержимого соответствующего буфера печати независимо от его заполнения.

Пример: OT('R =', R, RH, /)

2.17. <инструкция перехода> ::= <инструкция безусловного перехода>|<инструкция перехода на подпрограмму>
 <инструкция безусловного перехода> ::= G <номер метки перехода>
 <инструкция перехода на подпрограмму> ::= GR <номер метки перехода>

2.18. <инструкция возврата> ::= <инструкция возврата из подпрограммы>|<инструкция возврата из прерывания>
 <инструкция возврата из подпрограммы> ::= R <номер метки перехода>

<инструкция возврата из прерывания> ::= RD

Подпрограммой является участок программы, начинающийся меткой перехода и заканчивающийся инструкцией возврата из подпрограммы, номер метки перехода которой равен номеру метки, стоящей в начале подпрограммы.

Пример:

M6 : W = W + 1, F(16)R6,

Подпрограммой обработки прерывания является участок программы, начинающийся меткой прерывания и заканчивающийся инструкцией возврата из прерывания.

Пример:

D2 : N(5) A(0) F(25) L = LαN200, D, RD,

Обработка прерываний от аппаратуры КАМАК осуществляется интерпретатором на двух уровнях: интерпретатора и выполняемой программы. Прерывание от аппаратуры КАМАК на уровне интерпретатора может возникнуть как следствие прерывания на машинном уровне в любой момент времени при выполнении программы интерпретатором. В случае возникновения такого прерывания интерпретация программы прерывается и управление передается на содержащуюся в интерпретаторе специальную подпрограмму обработки прерываний от аппаратуры КАМАК на уровне интерпретатора. Эта подпрограмма читает содержимое регистра маски и запросов крейт-контроллера КАМАК (номер крейта при этом определяется текущим значением переменной C), запоминает номер возникшего прерывания и устанавливает маску на этот номер. На этом обработка прерывания на уровне интерпретатора заканчивается,

и управление возвращается на интерпретацию программы. Прерываниями на уровне выполняемой программы управляет интерпретатор. Они являются средством управления последовательностью действий, которое реализуется чисто программным путем. Такие прерывания интерпретатор может инициировать только в моменты времени между выполнением подпрограмм интерпретатора, которые моделируют синтаксические элементы языка КОК. Прерывание на уровне выполняемой программы возникает после обработки соответствующего прерывания на уровне интерпретатора, если в выполняемой программе имеется подпрограмма обработки прерывания, номер начальной метки которой равен номеру прерывания от аппаратуры КАМАК, зафиксированному во время обработки этого прерывания на уровне интерпретатора. При переходе на выполнение подпрограммы обработки прерывания значения переменных С, N, A, F, R, W и индикаторов сохраняются интерпретатором в стеке. Все прерывания от аппаратуры КАМАК обрабатываются на уровне выполняемой программы без приоритетов, в порядке их возникновения на уровне интерпретатора.

2.19. $\langle \text{условная инструкция} \rangle ::= \langle \text{инструкция IF} \rangle | \text{EL}$
 $\langle \text{инструкция IF} \rangle ::= \text{IF} \langle \text{условие} \rangle$
 $\langle \text{условие} \rangle ::= \langle \text{условие индикатора} \rangle | \langle \text{условие значения} \rangle$
 $\langle \text{условие индикатора} \rangle ::= \langle \text{индикатор} \rangle = \langle \text{значение индикатора} \rangle$
 $\langle \text{значение индикатора} \rangle ::= 0 | 1$
 $\langle \text{условие значения} \rangle ::= (\{ \langle \text{указатель значения} \rangle | \langle \text{указатель значения} \rangle \langle \text{знак условия} \rangle \langle \text{указатель значения} \rangle \}) \langle \text{указатель значащих разрядов} \rangle$

$\langle \text{знак условия} \rangle ::= = | / = | > | < | > = | < =$
 $\langle \text{указатель значащих разрядов} \rangle ::= \langle \text{пусто} \rangle | \& \langle \text{указатель значения} \rangle$
 Конструкция $\langle \text{указатель значения} \rangle$ в условии значения эквивалентна конструкции $\langle \text{указатель значения} \rangle / = 0$. Пустой указатель значащих разрядов эквивалентен $\&N$.

При выполнении инструкции IF с условием значения производится сравнение двоичных кодов, задаваемых указателями значения. Перед сравнением выполняется логическая операция И между сравниваемыми кодами и кодом, задаваемым указателем значащих разрядов.

Инструкция (цикл), следующая сразу за инструкцией IF, будет исполняться только в том случае, если выполняется условие данной инструкции IF. Инструкция (цикл), следующая сразу за инструкцией EL, будет исполняться только в том случае, если не выполнялась инструкция (цикл), предшествующая данной инструкции EL.

Примеры:

IF Q = 1 [F(0) JG3] EL G2,
 IF(R < 5)V5 = V5 + R, EL V6 = V6 - R,

2.20. $\langle \text{декларативная инструкция} \rangle ::= \langle \text{декларация} \rangle | \langle \text{ссылка} \rangle$
 $\langle \text{декларация} \rangle ::= /(\langle \text{имя декларации} \rangle) = \langle \text{указатель CNAF} \rangle /$
 $\langle \text{ссылка} \rangle ::= (\langle \text{имя декларации} \rangle)$
 $\langle \text{указатель CNAF} \rangle ::= \langle \text{указатель C} \rangle | \langle \text{указатель C} \rangle \langle \text{указатель NAF} \rangle | \langle \text{указатель NAF} \rangle$
 $\langle \text{указатель NAF} \rangle ::= \langle \text{указатель N} \rangle | \langle \text{указатель N} \rangle \langle \text{указатель AF} \rangle | \langle \text{указатель AF} \rangle$
 $\langle \text{указатель AF} \rangle ::= \langle \text{указатель A} \rangle | \langle \text{указатель A} \rangle \langle \text{указатель F} \rangle | \langle \text{указатель F} \rangle$

$\langle \text{указатель C} \rangle ::= C(\langle \text{указатель значения декларации} \rangle)$
 $\langle \text{указатель N} \rangle ::= N(\langle \text{указатель значения декларации} \rangle)$
 $\langle \text{указатель A} \rangle ::= A(\langle \text{указатель значения декларации} \rangle)$
 $\langle \text{указатель F} \rangle ::= F(\langle \text{указатель значения декларации} \rangle)$
 $\langle \text{указатель значения декларации} \rangle ::= \langle \text{десятичное число} \rangle | T | D$
 Имя декларации может быть пустым, но не должно содержать более 6 символов. В имени декларации допускается использование букв как латинского, так и русского алфавита.

Указатели значения декларации T и D задают ввод значения перед выполнением программы в специальном режиме работы интерпретатора: T — с клавиатуры терминального пульта, D — с магнитного диска.

Пример:

$$/(АЦП) = С(T) N(T) A(0) F(0)/$$

Ссылка осуществляет присваивание значений переменным С, N, А, F в соответствии с декларацией с таким же именем, при этом если в декларации задан указатель F, то после присваивания значений переменным производится запуск команды КАМАК. Ссылка может стоять в программе только после соответствующей декларации.

2.21. <вызов макроопределения> ::= " <имя макроопределения> "

Макроопределением является любая программа на языке КОК, содержащая вызовы макроопределений, записанная на магнитный диск. Имя макроопределения является именем файла, содержащего данное макроопределение.

2.22. Язык КОК имеет свободный формат. Пробелы, включенные в текст программы, не имеют значения (за исключением пробелов в текстах инструкций печати). В произвольные места текста программы могут включаться комментарии. Комментарием является текст, заключенный между символами «!».

3. Интерпретатор КОК. Ниже кратко описывается управление работой в основном второго и третьего вариантов интерпретатора КОК. Управление работой первого варианта интерпретатора незначительно отличается от описываемого ниже. Отличия определяются тем, что первый вариант не использует магнитного диска и работает без операционной системы.

Интерпретатор КОК может функционировать в трех режимах: ввод программы, ввод значений деклараций, выполнение программы. Режим работы интерпретатора задается оператором с клавиатуры терминального пульта (КТП). Во время работы интерпретатора в режиме ввода программы работают редактор и транслятор, входящие в состав интерпретатора. Редактор может вводить программу с КТП, перфоленты и магнитного диска. Одновременно с вводом программы редактор может производить вывод отредактированного текста программы на печать, перфоленту и магнитный диск. Режим ввода значений деклараций является вспомогательным, в котором задаются значения для указателей SNAF в соответствии с указателями значения декларации T и D.

В процессе работы интерпретатор может печатать на терминальном пульте (ТП) ряд сообщений и запросов. Текст сообщения начинается с двух символов подчеркивания, а текст запроса — с одного. Запрос требует от оператора ответа, который вводится в ЭВМ с КТП. Запросы бывают трех типов: управления, значения и имени файла. Ответ на запрос управления может состоять из одного или нескольких управляющих символов. Каждый управляющий символ воспринимается как определенная директива. Управляющие символы печатаются для подсказки в тексте запроса в круглых скобках, где символ «/» разделяет взаимоисключающие в ответе управляющие символы, а запятая — управляющие символы, которые могут одновременно присутствовать в ответе.

Запрос режима интерпретатора имеет вид:

_MODE? (I/D/R/F):

Управляющие символы в ответе на данный запрос задают: I — режим ввода программы, D — режим ввода значений деклараций, R — режим выполнения программы. Управляющий символ F вызывает завершение работы интерпретатора.

В начале работы в режиме ввода программы интерпретатор печатает запрос на вывод отредактированного текста программы:

_OUT? (L/T, P, D):

Управляющие символы в ответе на данный запрос задают: L — печать на АЦПУ, T — печать на ТП, P — вывод на перфоленту, D — вывод на магнитный диск.

Если во введенной программе имеется хотя бы одна инструкция печати на АЦПУ, то в начале работы в режиме выполнения программы интерпретатор печатает запрос на управление печатью:
_OL? (L/T/N):

При ответе L на данный запрос все имеющиеся во введенной программе инструкции печати будут выводить информацию на АЦПУ. При ответе T все такие инструкции будут производить печать на ТП (а не на АЦПУ!). При ответе N эти инструкции печати выполняться не будут.

Запрос значения в режиме ввода значений деклараций соответствует указателю T и имеет вид:

_? <переменная декларации> =

Здесь <переменная декларации> ::= C|N|A|F

Значение, вводимое в ЭВМ в ответ на данный запрос, задается десятичным числом. Аналогичный запрос в режиме выполнения программы (соответствующий указателю значения T) —

_?T =

В ответе на данный запрос значение может задаваться как десятичным, так и шестнадцатеричным числом. Ответ может состоять из одного символа «\$», который в этом случае интерпретатор воспринимает как директиву прекращения работы в текущем режиме.

Запрос имени файла для записи на магнитный диск программы (в режиме ввода программы) или данных (в режиме выполнения программы) имеет вид:

_OUT FILE?:

Запрос имени файла для ввода с магнитного диска программы (в режиме ввода программы) или данных (в режимах ввода значений деклараций и выполнения программы) имеет вид

_IN FILE?:

Имя файла может состоять не более чем из 6 символов. К введенному имени интерпретатор добавляет расширение КОК в режиме ввода программы и расширение DDD в режимах ввода значений деклараций и выполнения программы. Ответ может состоять из одного символа «\$» который в данном случае интерпретатор воспринимает как директиву отмены соответственно записи на магнитный диск или ввода с магнитного диска.

Редактор, входящий в состав интерпретатора, вводит программу в ЭВМ построчно, при этом строка может содержать не более 72 символов и должна заканчиваться символом «Возврат каретки». В начале работы интерпретатора в режиме ввода программы редактор настроен на ввод с КТП. Редактор запрашивает ввод очередной строки с КТП печатью номера этой строки на ТП. Управление вводом и построчным редактированием текста программы осуществляется с помощью специальных управляющих символов. Ввод программы с перфоленты или магнитного диска запускается вводом с КТП строки, содержащей управляющий символ «%» или «^» соответственно. Построчное редактирование вводимого текста программы производится по номерам строк. Номер строки, подлежащей редактированию, задается вслед за управляющим символом «#», перед управляющим символом «%» или «^» (пример: # 27%). Редактор вводит программу с перфоленты или с магнитного диска до строки с заданным номером включительно, при этом из последней введенной строки удаляется символ «Возврат каретки». После этого редактор переключается на ввод с КТП. Последнюю введенную строку можно корректировать с КТП, при этом удаление символов из строки осуществляется вводом символа «Возврат на шаг» или «Забой». Вводом символа подчеркивания строку можно удалить целиком. Вслед за откорректированной строкой с КТП можно вставить новые строки. После завершения редактирования строки ввод с перфоленты или магнитного диска можно продолжить вводом соответствующего управляющего символа, предварительно задав, если это необходимо, номер следующей строки, подлежащей редактированию.

При вводе программы в ЭВМ транслятор, входящий в состав интерпретатора и работающий параллельно редактору, анализирует программу символ за символом. При обнаружении в строке ошибочного символа транслятор сигнализирует об этом редактору, который останавливает ввод, удаляет из строки ошибочный символ и выдает сообщение:

--ERR 'ошибочный символ'

Вслед за этим редактор распечатывает текст строки до ошибочного символа и переключается на ввод с КТП. Обнаруженную ошибку можно исправить корректировкой строки с КТП, после чего ввод можно продолжить обычным образом. Такая совместная работа транслятора и редактора способствует более быстрому освоению языка КОК.

После завершения ввода программы интерпретатор выдает сообщение:

--VO: <десятичное число>

Десятичное число в данном сообщении задает адрес первого слова области переменных V в оперативной памяти ЭВМ. Это слово соответствует переменной VO.

Во время выполнения программы интерпретатор может обнаружить в программе ошибку. В этом случае выполнение программы прекращается и на ТП печатается сообщение об ошибке. При обращении к несуществующей переменной V сообщение об ошибке имеет вид

--ERR V

Если уровень вложенности циклов превысит 18, выдается сообщение:

--ERR DO

Если в программе встречается инструкция перехода или инструкция возврата из подпрограммы с таким номером метки перехода, что в программе отсутствует соответствующая метка, то выдается сообщение:

--NO M <номер метки перехода>

При обращении к несуществующему крейту КАМАК или к несуществующей ячейке оперативной памяти обработке ошибки на ЭВМ СМ-3, СМ-4 производит операционная система, при этом работа интерпретатора КОК прекращается. На ЭВМ «Электроника-60» подобную ошибку обрабатывает интерпретатор, который прекращает выполнение программы и печатает сообщение:

--ERR ADDR <десятичное число>

В данном сообщении десятичное число задает адрес инструкции интерпретатора КОК, при выполнении которой была зафиксирована ошибка.

Если в процессе выполнения программы производилась запись данных на магнитный диск, то после завершения выполнения программы интерпретатор печатает сообщение:

--OUT D: <число записанных на диск значений>

На ЭВМ «Электроника-60» выполнение программы может быть прекращено в любой момент времени вводом в ЭВМ с КТП символа \$.

4. Заключение. Язык КОК ориентирован в основном на специалистов по электронике, которым по роду своей работы приходится писать программы для тестирования и наладки аппаратуры КАМАК. Язык КОК достаточно прост и, как показала практика, вполне доступен для специалистов по электронике.

ЛИТЕРАТУРА

1. Бредихин С. В., Песляк П. М. Простая система программирования для САМАС.— Автометрия, 1976, № 1.
2. Выставкин А. Н. и др. Проблемно-ориентированные измерительно-вычислительные комплексы на базе ЭВМ СМ-3, СМ-4 и аппаратуры КАМАК для автоматизации научных исследований.— Автометрия, 1980, № 3.
3. Оллонгрен А. Определение языков программирования интерпретирующими автоматами.— М.: Мир, 1977.

*Поступила в редакцию 4 февраля 1981 г.;
окончательный вариант — 12 ноября 1981 г.*