

Диспетчер внешних прерываний обрабатывает прерывания от внешних пнициа-тивных источников и активизирует задачи, связанные с этими источниками внеш-них прерываний.

Получая управление от интерпретатора после выполнения очередного операто-ра текущей задачи, основной диспетчер выбирает из числа активных задач старшую по приоритету и сравнивает ее приоритет с приоритетом выполняющейся задачи. Если приоритет выполняющейся задачи выше, диспетчер дает указание интерпрета-тору продолжить ее выполнение. В противном случае диспетчер передает интерпре-татору номер задачи, подлежащей выполнению.

В отличие от ввода исходной программы, ее редактирования, трансляции во внутреннюю форму, компиляции и ретрансляции выполнение внутренней программы осуществляется в реальном масштабе времени.

Многопультная работа. ДМСРВ обеспечивает многопультную работу, т. е. не-сколько пользователей могут одновременно и независимо готовить и выполнять свои программы, используя ДМСРВ. Это достигается путем генерации и компоновки многораздельной операционной системы, в каждый раздел которой загружается ДМСРВ. При этом каждый раздел работает со своим пультом. Этот режим работы возможен только на ВК М-7000 и СМ-2, представляющих возможность многораздель-ной работы.

Генерация и компоновка ДМСРВ. Создание ДМСРВ происходит в два этапа. На первом этапе с помощью макрогенератора, транслятора с МНЕМОКОДа и компонов-щика программ генерируется операционная система реального времени АСПО.

На втором этапе производится генерация и компоновка ДМСРВ. На специаль-ном языке (языке макрокоманд) пользователь составляет программу генерации, в которой описывает включаемые в данную систему библиотечные подпрограммы поль-зователя, указывает максимальное количество выполняющихся одновременно задач, а также устройства или группы устройств, по прерываниям от которых должны за-пускаться на выполнение задачи. Здесь же задается тип системы (дисксовая либо бездисксовая). Эта программа последовательно обрабатывается макрогенератором и транслятором с МНЕМОКОДа. При этом используется библиотека макроопределений. Результат их работы — рабочие таблицы для ДМСРВ, оформленные в виде подпро-граммы.

Затем при помощи компоновщика программ создается загрузочный модуль, со-держащий следующие задачи: интерпретатор, диспетчер прерываний от внешних сигналов, диспетчер прерываний от внешних устройств.

Инициализация и запуск ДМСРВ осуществляются оператором при помощи спе-циальных команд операционной системы АСПО.

ЛИТЕРАТУРА

1. Кетков Ю. Л. Программирование на БЕЙСИКе.— М.: Статистика, 1978.
2. Айзенберг А. Б. и др. Агрегатная система программного обеспечения М-7000 АСВТ-М.— УСиМ, 1976, № 6.

*Поступило в редакцию 2 июня 1980 г.;
окончательный вариант — 13 октября 1980 г.*

УДК 681.3.06

А. П. КАЗАНЦЕВ

(Пуцино Московской)

АВТОМАТИЗИРОВАННОЕ РАБОЧЕЕ МЕСТО ИССЛЕДОВАТЕЛЯ

В последнее время все более широкое распространение находит концепция ав-томатизированного рабочего места (АРМ). Большое значение имеет автоматизация труда научных работников и инженеров — в общем, исследователей, занимающихся научными экспериментами, разработкой и испытаниями новой техники. АРМ иссле-дователя по своей специфике должно обеспечивать дистанционное управление объек-тами исследования, автоматизировать рутинные операции по сбору данных, выпол-нять расчеты, представлять обработанные данные в удобной для обозрения форме. АРМ исследователя (АРМИ) необходимо для повышения производительности труда исследователя при управлении экспериментом, обработке данных, математическом моделировании.

В нашей статье представлено АРМИ на основе мини-ЭВМ в малой конфигурации для автоматизации малого научного эксперимента общезначимого характера. В качестве универсального канала ввода-вывода в предлагаемой системе АРМИ применяется стандартный интерфейс КАМАК. Модули КАМАК обеспечивают связь АРМИ с объектом исследования и отчасти связь АРМИ с самим исследователем. Благодаря широкой номенклатуре модулей и гибкости структуры КАМАК имеется возможность приспособлять АРМИ к любой исследовательской задаче [1].

Основной элемент программного обеспечения в нашей системе — интерпретатор языка высокого уровня, обеспечивающий интерактивность и наибольшие удобства для пользователя, не имеющего профессиональной подготовки по программированию. Известно, что интерпретаторы языков высокого уровня (например, БЕЙСИК) нашли широкое применение в системах на основе микро-ЭВМ индивидуального пользования [2].

Эффективным средством общения пользователя с ЭВМ является машинная графика. В графической форме, наиболее удобной для восприятия человека, отображаются зарегистрированные и обработанные экспериментальные данные. Так же в графической форме представляются результаты моделирования. И наоборот, графическая информация в закодированном виде может вводиться в ЭВМ. В нашем АРМИ имеется графопостроитель, а также интерактивный графический дисплей. Интерактивность графики и интерпретатора языка обеспечивает работу АРМИ в диалоговом режиме.

Разработанный АРМИ построен на основе ЭВМ М400 из ряда программно совместимых малых ЭВМ серии СМ ЭВМ. К этому ряду примыкает и микро-ЭВМ «Электроника-60». Данное АРМИ реализовано на базовом комплекте УВК М400 с памятью 16 Кбайт. УВК расширен за счет алфавитно-цифрового дисплея, одного крейта КАМАК, дискретного графопостроителя и точечного граф-дисплея на основе ЭЛТ размером 16 см по диагонали. Графопостроитель и граф-дисплей управляются модулями КАМАК.

Наряду с аппаратным оборудованием, важной частью АРМИ является диалоговая система ДС СМ ЭВМ (далее ДС) [3], представляющая собой интерпретатор языка высокого уровня ФОКАЛ. Поскольку ФОКАЛ — проблемно ориентированный язык, программы пользователя, хранящиеся в ОЗУ ЭВМ, реализуют алгоритмы, разрабатываемые для своих целей исследователем, редактируемые и исполняемые в диалоговом режиме в едином процессе. Поскольку интерпретация значительно замедляет прохождение программ для некоторых процессов (в частности, для управления модулями КАМАК), требуется использование дополнительных программ, написанных на Ассемблере и исполняемых непосредственно в кодах. Эти программные модули встраиваются в ДС в виде дополнительных функций языка ФОКАЛ, вызываемых интерпретируемой программой так же, как и стандартные функции языка. Дополнительные функции получают свои имена и, будучи таким образом полностью включенными в ДС, расширяют и приспособляют ее к данному применению. Для разработки дополнительных функций в ДС имеется возможность обращения к стандартным функциям и подпрограммам ДС на уровне ассемблерных программ.

Собственно ДС СМ ЭВМ, помимо программ пользователя и дополнительных функций, требует всего 5,6 Кбайт от всего ОЗУ ЭВМ. Это свойство, наряду с краткостью записи программ пользователя и некоторыми другими особыми свойствами ДС, выгодно отличает ее, например, от интерпретатора БЕЙСИК М400, что и предопределило выбор ДС в качестве основного программного обеспечения АРМИ.

В нашей системе АРМИ в состав ДС включены стандартные функции управления модулями КАМАК, специальные функции управления контроллерами графических устройств, а также функции манипуляции экспериментальными данными. Эта система функций является развивающейся. Базовые функции АРМИ занимают 1708 байт ОЗУ.

Для регистрации экспериментальных данных в ОЗУ выделен буфер данных, куда записываются массивы в целочисленном однословном формате (длина слова 16 бит). Имеется возможность изменять размер буфера в зависимости от максимальной длины массивов данных. Ввод данных в буфер осуществляется функциями управления модулями АЦП, входными регистрами и т. д.

Далее описывается система базовых функций АРМИ.

В ДС включены следующие стандартные функции управления модулями КАМАК (следует различать понятия «функция интерпретатора» и «КАМАК-функция»): FADR(N, A) — адресация; FF(CA, F, D) — общая функция; FI(CA) — ввод данных; FO(CA, D) — вывод данных. В качестве аргументов функций здесь декларированы переменные: N — номер станции в крейте, A — субадрес регистра, F — КАМАК-функция, D — данное, CA — КАМАК-адрес.

Функция FADR определяет КАМАК-адрес (CA) выбранного регистра в произвольном модуле КАМАК, установленном в крейте. Далее CA будет использоваться для адресации этого регистра при выполнении требуемых КАМАК-функций (F). Любая F может быть исполнена общей функцией FF. Однако наиболее часто встречающиеся КАМАК-функции чтения или записи удобно задавать, используя имена быстро интерпретируемых функций FI и FO соответственно.

Имеются две функции управления граф-дисплеем: FG(I) — функция отображения и FCTL(K, N, M) — функция управления параметрами. Первая из них выводит

на экран ЭЛТ последовательность точек, заданную двумерным массивом чисел или одномерным, если отображается однозначный процесс с постоянной дискретизацией независимой переменной (массив данных должен быть предварительно записан в буфер, упоминаемый выше). Функция отображения реализует однократную развертку изображения и задается в цикле программы на языке ФОКАЛ. Испытания показали, что частота повторения такой развертки в зависимости от программного цикла может быть от 16 до 50 Гц, что обеспечивает хорошее качество изображения. Функция отображения имеет единственный аргумент — номер точки, где должна быть высвечена яркостная метка, иначе говоря, точечный курсор для интерактивного общения пользователя с машиной в режиме графического диалога. При последовательном изменении значения аргумента функции FG в цикле курсор пробегает по последовательности точек со скоростью, меньшей скорости реакции пользователя, позволяя последнему отмечать нужные точки прерыванием программы. Дальнейший ход программы выбирается пользователем, и отмеченные курсором и прерыванием данные обрабатываются программой. Таким образом, реализуется графический диалог.

Функция управления параметрами FCTL (K, N, M) служит для модификации изображения на граф-дисплее. Первая переменная из списка аргументов задает показатель масштаба изображения. В соответствии со значением этого параметра масштаб изображения на граф-дисплее может быть увеличен (уменьшен) в 2, 4, 8 и т. д. (по степеням 2) раз. Вторая переменная определяет число точек изображения. В случае отображения процессов N задает масштаб изображения по времени протекания процесса. Третий аргумент принимает логические значения «1» и «0» и указывает на наличие или отсутствие точечного курсора на изображении.

В различных системах машинной графики [4, 5] общепринята определенная структура функций графического отображения. В описываемом АРМИ используется ряд функций управления графопостроителем, входящим в систему: перемещение поднятого пера в точку — FM(X, Y) (функция MOVE); черчение вектора — FD(X, Y) (функция DRAW); взятие начальной точки — FORN (X0, Y0) (функция ORIGIN); масштабирование — FSCL(MX, MY) (функция SCALE); черчение символа — FC (X, Y, C) (функция CHARACTER). Здесь декларированы переменные; X, Y — координаты точки; MX, MY — масштабные коэффициенты; X0, Y0 — координаты начальной точки; C — код символа.

Первоначально координаты X0, Y0 имеют нулевые значения, а коэффициенты MX, MY — единичные. Перед началом черчения в программе пользователя задаются требуемые значения параметров X0, Y0, MX и MY исполнением функций FSCL и FORN, в которых аргументы получают соответствующие значения. Для выхода на начальную точку какой-либо линии применяется функция FM, а для черчения прямолинейных отрезков (линейная интерполяция) — функция FD. Для черчения в выбранной точке символа, задаваемого десятичным кодом, используется функция FC. Координаты X и Y в этих функциях задаются без предварительного масштабирования и сдвига, так как эти операции выполняются автоматически с помощью введенных параметров MX, MY, X0 и Y0.

Функции манипуляции данными осуществляют перенос данных из целочисленного буфера в область переменных с плавающей запятой интерпретатора, где они могут обрабатываться программами пользователя. В процедуре преобразования массивов данных в формат с плавающей запятой первоначально используется функция взятия адреса начала буфера. Общий вид функции — FB(0). Эта функция представляет адрес начала буфера данных в виде числа из области переменных интерпретатора. Далее применяется функция чтения данных. Общий вид функции — FR(A, D). В списке аргументов первый аргумент — адрес, второй — данное. Адрес читаемого слова задается на основе значения полученного ранее адреса начала буфера как базы. После выполнения функции чтения второй аргумент получает значение данного, указанного первым аргументом функции.

В некоторых случаях, например для функции отображения обработанных данных на граф-дисплее, требуется обратный перенос данных из области переменных интерпретатора в целочисленный буфер. Для этих целей служит функция записи. Общий вид функции — FW(A, D). Переменные имеют тот же смысл, что и переменные из списка аргументов функции чтения.

ВЫВОДЫ

Предложенная система АРМИ позволяет:
вести управление экспериментом и обработку данных по программам, разрабатываемым самим экспериментатором;
применять в едином исследовательском процессе математическое моделирование совместно с экспериментальными наблюдениями;
легко расширять или изменять систему, приспособивая ее к новым применениям.

Автор выражает признательность за содействие в работе Л. А. Пронину и Л. И. Подольскому.

ЛИТЕРАТУРА

1. Колпаков И. Ф. Электронная аппаратура на линии с ЭВМ в физическом эксперименте.— М.: Атомиздат, 1973, с. 124—125.
2. Дур Дж. Дешевые вычислительные устройства на базе микропроцессоров.— ТИИЭР, 1978, т. 66, № 2.
3. Перфоленточная диалоговая система программирования ДС, СМ ЭВМ: Описание языка.— М.: ИНЭУМ, 1976.
4. Plot-10. Terminal Control System. User's Manual. Tektronix.— Beaverton, Oregon, USA, 1976.
5. Versaplot Graphics Programming Manual. Versatec.— Santa Clara, Cal., USA, Feb. 1975.

Поступило в редакцию 5 марта 1980 г.
