

14. Штейн М.Е., Штейн Б.Е. Методы машинного проектирования цифровой аппаратуры. М., "Советское радио", 1973, 288 с.

М.А. Шамашов

ФОРМАЛЬНОЕ ОПИСАНИЕ ПРОЦЕССА ЭМУЛЯЦИИ СУВМ

Все более широкое использование специализированных УВМ (СУВМ) для решения задач управления сложными техническими системами ставит целый ряд задач по совершенствованию разработки вычислителей подобного класса. Следует отметить, что в то время как стоимость аппаратного обеспечения СУВМ снижается, стоимость программного обеспечения (ПО) проявляет тенденцию к непрерывному росту. Одним из путей снижения стоимости ПО является использование методов автоматизации программирования. Однако, как правило, ограниченный состав внешних устройств и малый объем оперативной памяти не могут обеспечить функционирование подобных систем на СУВМ. Это приводит к необходимости разработки эмуляторов СУВМ на больших универсальных ЭВМ. (В дальнейшем будем называть их инструментальными ЭВМ).

Эмуляция (от англ. *emulation* - соревнование, состязание) - термин, введенный специалистами фирмы IBM для определения процесса совмещения входных языков ЭВМ. Первоначально аппаратно-программные эмуляторы использовались преимущественно для ускорения процесса реализации программ написанных для машин второго поколения на машинах третьего поколения [1]. Внимание при этом акцентировалось на аппаратных методах эмуляции. Резкое различие в архитектуру моделируемых СУВМ и инструментальных ЭВМ вынуждают отказаться от использования дополнительного комплекса аппаратуры и более внимание уделять программным методам. За последнее время привнес целый ряд работ по совершенствованию программных методов эмуляции. В частности, общепринятую схему "идентификация - интерпретация" [2]-[4] заменяют двухэтапная схема [5,6] и метод "открытых подпрограмм" [7], позволяющие значительно ускорить процесс отладки ПО СУВМ. Исследования существующих и оригинальных методов эмуляции и практическая реализация конкретных эмуляторов

позволяет сделать ряд обобщений и предложить возможное формально описание процесса эмуляции.

Пусть $X = \{x_1, x_2, \dots, x_i, \dots\}$ - множество состояний моделируемого СУВМ. Заметим, что каждое x_i (где в общем случае $i \in \overline{1, \infty}$) - вектор включающий в себя состояния элементов массива "память-регистры" СУВМ, периферии и т.д.:

$$x_i = \{x_{i1}, x_{i2}, \dots, x_{in}\}.$$

Определим конечное множество различных команд СУВМ через

$$A = \{\alpha_1, \alpha_2, \dots, \alpha_m\}.$$

Выполнение программы на реальной СУВМ можно представить как некоторое преобразование G множества X :

$$x_{i+1} = G_k(x_i), \text{ где } i \in \overline{1, \infty}; k = f(x_i).$$

Преобразование G_k над вектором x_i соответствует выполнению команды α_k СУВМ. Определив $k = f(x_i)$, мы характеризуем тот факт, что конкретная команда выполняемая машиной является одно из составляющих ее вектора состояния.

Пусть множество Y - множество состояний инструментальной ЭВМ. Выделим из этого множества подмножество Y' , моделирующее множество X :

$$Y' = \{y'_1, y'_2, \dots, y'_i, \dots\}$$

$$y'_i = \{y'_{i1}, y'_{i2}, \dots, y'_{ip}\}.$$

Здесь y'_{ij} ($i \in \overline{1, \infty}; j \in \overline{1, p}$) элемент из массива "память-регистры" инструментальной ЭВМ, моделирующей определенное подмножество вектора x_i .

Тот факт, что y'_i моделирует состояние x_i , обозначим

$$y'_i \leftrightarrow x_i.$$

Определим конечное множество команд инструментальной машины:

$$B = \{\beta_1, \beta_2, \dots, \beta_q\}.$$

Введем множество семантических эквивалентов:

$$\Gamma = \{\gamma_1, \gamma_2, \dots, \gamma_m\}.$$

Каждый элемент этого множества - вектор, содержащий определенный набор команд инструментальной ЭВМ:

$$\gamma_i = \{\beta_{i1}, \beta_{i2}, \dots, \beta_{ik}\},$$

где $i \in \overline{1, m}; j \in \overline{1, k}; ij \in [1, q]$.

Выполнение набора команд, содержащихся в семантическом эквиваленте \mathcal{P}_κ , приводит к P_κ преобразованию элемента множества $Y'_{i+1} = P_\kappa (y'_i)$.

где $i = \overline{1, \infty}$; $\kappa = f(x_i)$.

Условие функционирования эмулятора можно представить в виде выражения

$$\forall x_i \exists \mathcal{P}_\kappa (y'_{i+1} = P_\kappa (y'_i) \leftrightarrow x_{i+1} = G_\kappa (x_i)),$$

где $i = \overline{1, \infty}$; $\kappa = f(x_i)$.

Используя тот факт, что $Y' \subset Y$, эмулятор E можно полностью определить набором множеств и преобразований:

$$E = \{X, Y, A, B, T, G, P\}.$$

Л и т е р а т у р а

1. М а л и н о в с к и й Б.М., П о г о р е л ь и й С.Д. Методы эмуляции ЭВМ. "УСим", 1974, 4.
2. Д е д е р е р С.Ю. Система интерпретации автоматизированной ЭВМ. М., Издательство ИТМ и ВТ АН СССР, 1974.
3. В я т к и н Г.В., Р ы ч к о в В.А. Моделирование малого цифрового вычислителя на универсальной ЭВМ. В сб: Система автоматизации научных исследований. Рига, "Знание", 1973.
4. Б е л я в с к и й Е.И. Об отладке программ бортовых ЦВМ с помощью интерпретаторов. В сб: Кибернетическая техника. Киев, "Наукова думка", 1971.
5. Б е л я в с к и й Е.И., М и х л и н Г.З., Ф р у м к и н В.А. О построении интерпретаторов специализированных ЭВМ. "УСим", 1975, 5.
6. Б у д я ч е в с к и й И.А., К о в а р ц е в А.Н., Ш а м а ш о в М.А. Некоторые вопросы моделирования систем сбора и обработки данных научного эксперимента. Межвузовский сборник Автоматизация экспериментальных исследований. Вып. 9. Куйбышев, КуАИ, 1976.

