

ПРИЛОЖЕНИЕ НА ЕДИН МЕТОД ЗА МОДЕЛИРАНЕ НА АСМ С ПОМОЩТА НА УНИВЕРСАЛНА АСМ

Румяна Киркова, Аврам Ескенази

Настоящата работа разглежда един метод за моделиране на АСМ (реално съществуваща или проектирана) с помощта на универсална АСМ и приложението му в две конкретни разработки. Върху създадения модел могат да се извършват изследвания, които по ред причини (например големи разходи, загуба на време, невъзможност за проверка и др.) в действителност не могат да бъдат извършени, както и да се направят необходимите изводи.

Тези предимства на моделирането могат да се използват от инженерните науки в областта на проектирането на АСМ както в етапа на конструирането, така и в етапа на експлоатацията им.

В Математическия институт с Изчислителен център при БАН беше разработена [1] и изследвана методика за моделиране на една АСМ с помощта на универсална АСМ (в случая МИНСК-22).

1. ДЕФИНИЦИИ

За да дефинираме АСМ, ще използваме елементи от теорията на автоматите [2].

АСМ ще наричаме съвкупност от следните елементи: централно устройство, командно устройство, оперативна памет и външни устройства.

1.1. Централно устройство (ЦУ) ще наричаме краен автомат с определен набор от състояния в зависимост от броя и дължината на регистрите. Всяка промяна на поне един бит от произволен регистър ще означава състояние, различно от предходното. Всеки вход на този краен автомат ще наричаме микрооперация на АСМ.

1.2. Командно устройство (КУ) — това е библиотека от команди. Състои се от дешифратор на командите и памет, която съхранява реда на изпълнение на микрооперациите за всяка отделна команда. Функцията на дешифратора на командите е да определя коя последователност от микрооперации трябва да се възбуди за конкретния входен сигнал, т. е. за конкретната команда.

1.3. Оперативна памет (ОП) е съвкупност от адресирани клетки за съхранение на информация; тази информация може да се използва непосредствено от ЦУ за извършване на операции.

1.4. Външни устройства (ВУ) — това са външна памет, входни и изходни устройства. Последните две осъществяват връзката човек — машина, а външната памет обикновено съхранява информация, чийто обем надхвърля капацитета на ОП.

2. ОБЩИ ПРИНЦИПИ НА МЕТОДА ЗА МОДЕЛИРАНЕ

2.1. Ще се спрем отначало на най-общите особености на моделирането с разглежданата методика, а след това ще отбележим специфичните черти на конкретните реализации.

За двете машини не се правят никакви специални предположения. Моделираната машина означаваме с M_1 , а моделиращата — с M_2 . Моделирането се извършва с помощта на моделираща програма, написана на един от езиките на M_2 . Такава програма трябва да съдържа всички регистри на M_1 , подпрограми за реализирането на отделните микрооперации на M_1 , подпрограми за реализирането на отделните операции на M_1 и всички необходими константи.

В M_2 се въвежда моделиращата програма и програма, написана в кодовете на M_1 , наречена работна. С помощта на моделиращата програма тази работна програма се изпълнява така, както би работила в M_1 .

2.2. Моделиране на централното устройство. Регистрите на M_1 се моделират като думи от оперативната памет на M_2 . Тук различаваме два случая:

2.2.1. Дължината на регистъра е по-малка или равна на дължината на думата в M_2 . В този случай естественият път е всеки регистър да се моделира в една дума в паметта на M_2 .

2.2.2. Дължината на регистъра е по-голяма от дължината на думата в M_2 . В този случай регистърът се моделира с няколко думи, като се прави подходящо съответствие.

2.3. Моделиране на командното устройство. Всяка команда се моделира с подпрограма, в която има последователно обръщане към определени от алгоритъма ѝ микрооперации.

Първата функция на командното устройство е дешифриране на командата, което се извършва от така наречения дешифратор на командите. При моделирането това става така: ако M_1 има K на брой команди, в K клетки от паметта на M_2 се записват команди за безусловен преход към първия адрес на подпрограмата, моделираща съответната команда от M_1 . Предаването на управлението към този първи адрес моделира изработването на управляващ сигнал, което е втората функция на командното устройство.

2.4. Моделиране на оперативната памет. В зависимост от обема на паметите на M_1 и M_2 имаме два случая:

2.4.1. M_1 има значително по-малка по обем памет от M_2 . Тогава паметта на M_1 може да се моделира изцяло в паметта на M_2 .

2.4.2. Паметта на M_1 е сравнима или по-голяма по обем от тази на M_2 . Тогава може да се моделира само част от нея в паметта на M_2 .

В зависимост от дължината на думата имаме следните случаи:

2.4.3. Дължината на думата на M_1 е значително по-малка от тази на M_2 . Тогава е възможно моделирането на няколко думи от M_1 в една

дума от M_2 . Този начин е икономичен, но създава трудности при обръщане към думата от модела.

2.4.4. При равни дължини на думите на всяка дума от M_1 се съпоставя дума от M_2 .

2.4.5. Когато дължината на думата в M_1 е по-голяма от тази на M_2 , тя се моделира чрез подходящ брой думи от M_2 .

2.5. Моделиране на външните устройства. По принцип външните устройства на M_1 се моделират чрез външните устройства на M_2 . Входно-изходните процедури се моделират чрез подходящи подпрограми.

Въз основа на тези общи принципи бяха разработени следните две моделиращи програми:

а) програма за моделиране на ЦУ, КУ и част от ОП на проект за българска изчислителна машина;

б) програма за моделиране на аритметиката с фиксирана запетая и част от системата за прекъсване на IBM/360.

При по-нататъшните описания първата машина ще означаваме с M_{11} , а втората — с M_{12} .

3. РЕАЛИЗИРАНЕ ВЪРХУ МИНСК-22

3.1. Най-общите характеристики на моделираните машини са: за M_{11} :

обем на паметта — $4K$ клетки;
дължината на думата — 37 разреда;
едноадресна, двукомандна.

за M_{12} :

обем на паметта — възможност до $2^{14}K$ байта;
дължина на байта — 8 разреда;
променлива дължина на думата;
двуадресна, с четири формата на командите.

3.2. Централно устройство. Всички регистри на M_{11} , както и общите регистри и тези за защита по ключ на M_{12} имат дължина, по-малка или равна на дължината на думата в M_2 . Поради това всеки от тях беше моделиран с една дума от паметта на M_2 . Управляващите думи и регистърът на командите на M_{12} поради по-голямата им дължина бяха моделирани с по две думи от M_2 .

Отделните микрооперации се реализираха като подпрограми, написани на машинния език на M_2 .

3.3. Командно устройство. Каталогът на командите, който се използва от дешифратора, може да се моделира по следните два начина:

3.3.1. Ако означим с C кода на командата, в клетка с адрес $A+C$, където A е подходящо избран адрес от паметта на M_2 , се записва команда за безусловен преход към началото на подпрограма за C .

3.3.2. В K последователни клетки в произволен ред са нанесени кодовете C на командите и началните адреси на съответните подпрограми.

Първият начин беше използван в M_{11} . Той дава възможност за по-бързо обръщане към подпрограмата, реализираща C , но тъй като кодовете на командите не са последователни числа, необходимо е да се отделят повече от K на брой клетки от паметта на M_2 .

При втория начин имаме точно обратното -- печели се памет, но се увеличава времето за обръщане към съответната подпрограма. Този начин беше приложен в М12. За да се ускори в известна степен търсенето на подлежащия за обработка код, кодовете бяха разделени в четири подкаталога въз основа на съществуващите в М12 формати на командите.

И при двата случая беше необходимо създаването на управляваща програма, която да осъществи намирането на началото на съответната под програма и да ѝ предаде управлението.

3.4. Оперативна памет. При моделирането на ОП на М11 и М12 бяха обхванати повечето от изброените в 2.4 случаи.

Паметта на М11 е сравнима с паметта на М2. Поради целите на заплануваните експерименти, изискващи малка памет, както и поради това, че трябва да се предвиди място и за моделиращата програма, беше моделирана само част (в обем от 1000_8 клетки) от оперативната памет на М11.

Паметта на М12 в почти всички машини на системата е по-голяма от паметта на М2. Поради това в $10\,000_8$ клетки от паметта на М2 бяха моделирани байтовете с адреси от 0000_{16} до 3777_{16} на М12.

Разбира се, може да се използва и външната памет на М2, но това би било свързано с чувствително увеличение на времето за изпълнение на всяка отделна команда.

Тъй като думите от паметта на М11 са 37-разредни, както и в М2, на всяка дума от паметта на М11 беше съпоставена по една дума от М2. В М12 най-малката адресуема информационна единица е 8-разредният байт. Във всяка клетка на М2 се моделираха по 4 байта. Останалите 5 разреда от клетката служеха за съхранение на информация при междинни операции.

3.5. Външни устройства. Целите на разглежданите две разработки, както беше отбелязано по-горе, не изискваха моделиране на входно-изходните процеси за М1. И в двата случая беше достатъчно вместо моделиране да се извърши имитиране, т. е. подадената на каквото и да е входно устройство информация (работна програма и данни), написана в кода на М1, да бъде внесена в модела на оперативната памет на М1. При такъв подход ни интересуваха само началните данни и крайните резултати и затова входно-изходните процеси бяха реализирани по най-удобния за нас начин, без да се ангажираме с действителното протичане на процесите в М1. Такова решение беше най-подходящо за М12 и поради голямата сложност на входно-изходните ѝ процедури.

Със спомагателни функции и вън от модела е служебният печат. Той беше използван и в двете разработки, което даде възможност за получаване на разнообразна информация по състоянието на всички компоненти на модела. Така например при М11 служебният печат позволяваше да се провери правилният резултат след изпълнение на алгоритъма на определена команда. При М12 след края на работната програма или при прекъсване се отпечатваха произволни участъци от модела на оперативната памет и съдържанието на определени регистри. Освен това съществуваха възможност за отпечатване на текущата управляваща дума и друга актуална информация по време на изпълнението на работната програма.

3.6. Разпределение на паметта на М2. Описаните дотук компоненти на модела се разполагат в паметта на М2, както следва:

	M11	M12
Оперативна памет	1000 ₈ клетки	4000 ₁₆ байта
Моделираща програма		
управляваща програма	47 ₈ клетки	130 ₈ клетки
каталог за дешифратора	100 ₈ клетки	145 ₈ клетки
подпрограми за команди и микрооперации	1700 ₈ клетки	1520 ₈ клетки
регистри	14 ₈ клетки	34 ₈ клетки
константи	100 ₈ клетки	200 ₈ клетки

4. ИЗВОДИ

В процеса на разработката на моделите на M11 и M12 и в стадия на експериментирането им бяха направени сравнения и изводи.

Установени бяха известни различия в подхода при проектирана АСМ (M11) и при реално съществуваща АСМ (M12). Така при M12 точността на моделирането, т. е. степента на съответствие между M1 и M2, в голяма степен зависеше от правилността на направените хипотези за различните елементи (регистри за защита на ОП, ред на действие на прекъсванията и др.) поради недостатъчна информация. Очевидно при проектирана АСМ необходимостта от такива хипотези отпада. В замяна на това задачата на моделиращата програма за реална АСМ се улеснява от известна свобода при боравенето с набора от микрооперации. Като се намали дълбочината на съответствието, тази програма може за удобство да реализира една микрооперация с няколко различаващи се (в известни граници) подпрограми при използването на тази микрооперация в отделните команди. Поради естеството на целите при моделиране на проектирана АСМ такова улеснение не съществува.

От направените експерименти се потвърдиха в общи линии предварителните концепции, които имахме за M12. Моделиращата програма беше организирана така, че включването на допълнителни команди в нея не е трудно — анализът показва, че за повечето от тях могат да се използват построените вече подпрограми на микрооперации. Известни операции обаче (EDIT и др.) биха изисквали твърде големи и сложни програми. По-леко би било реализирането на останалите програмни прекъсвания. Изследванията върху съотношението между времето за изпълнение на една команда в реалната машина и в модела показаха, че последният е пригоден за експерименти, но използването му за изпълнение на работни програми, написани на език на M12, не е удачно. Малко по-добри резултати се получават при премахването на защитата по четене и запис, които са необходими само при мултипрограмен режим на работа. Главните причини за големия разход на време са две:

- голямата разлика в структурата на M12 и M2;
- съществено по-бързият цикъл на паметта на M12.

При разработване на модела на M11 въз основа на направени статистически изследвания се установи, че времето за изпълнение на операция „събиране“ надхвърля предварително зададените граници. Това наложи известни корекции в алгоритъма ѝ. Освен това се оказа необходимо допълването на набора от микрооперации с още една — „дешифриране кода на операцията“.

Описаните две разработки показват практическото приложение на този нов подход [1] при проектиране и изследване на една АСМ.

ЛИТЕРАТУРА

1. Dobrev, D., R. Aslanian, K. Boyanov. Computer simulation. In: Elektronische Informationsverarbeitung und Kybernetik, 2, 1966, № 4, 207—211.
2. Глушков, В. М. Синтез цифровых автоматов. Физматгиз, Москва, 1962.
3. Principles of Operation IBM/360 System, Form A22—6821-7.

Поступила на 18. VI. 1971 г.

ПРИМЕНЕНИЕ ОДНОГО МЕТОДА МОДЕЛИРОВАНИЯ ЭВМ ПРИ ПОМОЩИ УНИВЕРСАЛЬНОЙ ЭВМ

Румяна Киркова, Аврам Ескенази

(Резюме)

Рассматривается метод моделирования электронной вычислительной машины (ЭВМ) при помощи универсальной ЭВМ, разработанный и опробованный в Математическом институте с Вычислительным центром Болгарской академии наук. Реализованы две модели:

а) на центральном устройстве, на командном устройстве и на части оперативной памяти спроектированной болгарской вычислительной машины;

б) на арифметике с фиксированной запятой и на части системы прерывания IBM/360.

Приводятся проделанные в стадии экспериментирования сравнения и выводы.

APPLICATION OF A METHOD OF MODELLING OF A COMPUTER REALIZED BY MEANS OF A GENERAL-PURPOSE COMPUTER

Rumjana Kirkova, Avram Eskenazi

(Summary)

A method of modelling of a computer realized by means of a general-purpose computer is considered. The method under consideration was worked out and experimented at the Mathematical Institute of the Bulgarian Academy of Sciences. Two models have been realized.

a) a model of the central unit, of the control unit and of a part of the main storage of a designed Bulgarian computer.

b) a model of the IBM/360 fixed point arithmetics as well as of part of its interruption system.

Some comparisons made and results obtained in the process of experimentation are exposed in the paper.