



TEHNIKA I INFORMATIKA U OBRAZOVANJU

3. Internacionalna Konferencija, Tehnički fakultet Čačak, 7–9. maj 2010.

TECHNICS AND INFORMATICS IN EDUCATION

3rd International Conference, Technical Faculty Čačak, 7–9th May 2010.

UDK: 004.4

Stručni rad

VIZUELIZACIJA SABIRANJA BINARNIH BROJEVA KORIŠĆENJEM MEMORIJE

Nebojša Stanković¹, Gordana Marković²

Rezime: U radu je opisana aplikacija koja omogućava vizuelizaciju rada aritmetičko logičke jedinice (ALU), na primeru sabiranja dva broja. Ovaj vizuelni alat omogućava korisnicima da razumeju tok podataka: memorija-registri-ALU-registri-memorija. Aplikacija je realizovana korišćenjem programskog jezika MS Visual Basic 6.0.

Ključne reči: Edukacija, simulacija, aritmetičko-logička jedinica.

THE VISUALISATION OF THE BINARY NUMBERS ADDITION WITH THE MEMORY USAGE

Summary: In this paper is explained an application which allows the visualisation of arithmetical-logical unit (ALU) work with an example of two numbers addition. This visual tool allows users to understand the stream of data through memory-registries-ALU-registries-memory. The application is written using the programming language MS Visual Basic 6.0.

Key words: Education, simulation, ALU.

1. UVOD

Razvoj računara uslovljava upoznavanje korisnika na različitim nivoima (od korisničkog pa do dizajnerskog) sa funkcionalnostima koje računari poseduju. Paralelno sa razvojem računara razvijala se i tehnika simulacije računarskog sklopa i pojedinih njegovih delova i to sve sa ciljem da se vizuelno dočara kako to računar radi. Zahvaljujući simulacijama računarskog sistema korisnici se edukuju kako bi razumeli šta se u sistemu dešava. Takođe, simulacije omogućavaju istraživanja određenih situacija koje su od interesa za računarski sklop. Samim tim, istorijat razvoja simulacije računarskih sistema tesno je vezan za razvoj samih računara.

Rad opisuje aplikaciju koja omogućava vizuelizaciju rada aritmetičko-logičke jedinice (ALU) kao sastavnog dela procesora. Na primeru sabiranja dva binarna broja, korisniku se vizuelno prikazuje tok podataka: memorija-registri-ALU-registri-memorija.

¹ Mr Nebojša Stanković, Tehnički fakultet, Svetog Save 65, Čačak, E-mail: jack@tfc.kg.ac.rs

² Gordana Marković, Tehnička škola, Čačak, E-mail: branko333@nadlanu.com

2. EDUKACIONI ZNAČAJ RAČUNARA

Intenzivna primena Informaciono komunikacionih tehnologija u raznim oblastima ljudske delatnosti, uticala je na to da računari postanu sve prisutniji i u procesu nastave i učenja i zahvaljujući prednostima koje ima nad ostalim sredstvima, pripada mu vodeće mesto u procesu uvođenja inovacija u nastavi.

Dijapazon primene računara u obrazovanju praktično je neograničen i u uslovima «eksplozije znanja» računar sa svojim hardverskim i softverskim mogućnostima predstavlja jedno od najboljih, najbržih, najpreciznijih i najpouzdanijih sredstava za dolaženja do informacija i njihove primene. U obrazovanju njegova primena praktično može doprineti rešavanju problema «informacione barijere», tj. podići nastavni rad na viši, kvalitetniji nivo; učiniti ga efikasnijim, delotvornijim i savremenijim.

U visokoškolskom obrazovanju, prednosti nastave uz pomoć računara takođe su sadržane u mogućnostima veće misaone mobilnosti, aktiviranja i samostalnog rada studenata. Savremeni računari pružaju mogućnost simultanog gledanja slike, slušanja govora i korišćenja multimedijjskih izvora saznanja, što, svakako, doprinosi bržem i potpunijem usvajanju gradiva, trajnijem pamćenju naučenog, efikasnijem korišćenju i kreativnijoj primeni usvojenih znanja. Računar sa studentom komunicira pismeno i usmeno, vodi dijalog, pruža neophodne informacije, predstavlja grafikone, slike, filmove, stranice knjiga, projekcije, simulacije, daje objašnjenja pokazanog, upućuje na rešavanje problema, po potrebi daje dopunska uputstva, ispravlja greške i ocenjuje rezultate učenja.

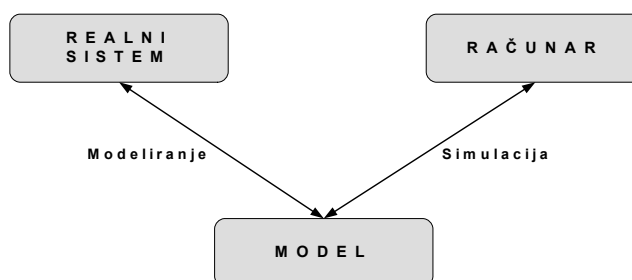
Ali, primena računara u nastavi još uvek nije na zadovoljavajućem nivou upravo zbog nedovoljnog znanja i iskustva nastavnika u ovoj oblasti. Računari mogu unaprediti nastavu i obrazovanje ukoliko se primene na pravom mestu, u pravo vreme, sa adekvatnim sadržajem i metodičkim osmišljenim tehnikama i postupcima. To, zapravo, znači da bi bila potrebna posebna metodika primene računara u nastavi, a nova funkcija predavača u savremenim tehničko-tehnološkim uslovima pretpostavlja i radikalnu promenu filozofije obrazovanja, korenite promene psihološkog i pedagoškog obrazovanja, osavremenjivanje metodičke spreme, potpunije poznavanje i adekvatnu upotrebu savremenih obrazovnih tehnologija.

3. SIMULACIONO MODELIRANJE

U današnje vreme vrlo često se pristupa kreiranju modela za određeni realni sistem kao i simulaciji funkcija tog sistema. Modeliranje je kreativan proces ljudskog uma kojim se na klasičan način formiraju funkcije (koje su od interesa) nekog realnog sistema. Na bazi modela realizuje se i simulacija koja se može posmatrati i kao relacija između modela i samog računara na kome se odvijaju odgovarajuća izračunavanja.

Modeliranje predstavlja jedan od osnovnih procesa ljudskoga uma i usko je vezano za način ljudskog razmišljanja i rešavanja problema. Ono predstavlja svakodnevnu aktivnost i veliki deo onoga što nas čini ljudskim (inteligentnim) bićima i izražava našu sposobnost da mislimo i zamišljamo, da koristimo simbole i jezike, komuniciramo, da vršimo generalizacije na osnovu iskustva, da se suočavamo sa neočekivanim. Upravo zato, modeliranje se najčešće posmatra kao najznačajnije konceptualno sredstvo koje čoveku stoji na raspolaganju.

Reč simulacija u svakodnevnoj upotrebi može da označi veći broj različitih aktivnosti. Ako se proces izgradnje apstraktnih modela za neke sisteme realnog sveta i obavljanje eksperimenata nad njima odvijaju na računaru, tada se govori o računarskom modeliranju i simulaciji. Danas je modeliranje nezamislivo bez računara. Računar se u modeliranju koristi za dve svrhe: za razvoj modela i za izvođenje proračuna na osnovu stvorenog modela. Modeliranje i simulacija predstavljaju složenu aktivnost koja uključuje tri elementa: realni sistem, model i računar. Na slici 1. dat je uprošćen prikaz ove aktivnosti.



Slika 1. Relacije modeliranja i simulacije

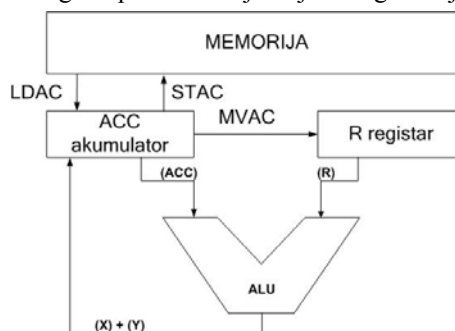
Realni sistem je izvor podataka za specifikaciju modela, tj. izvor podataka o ponašanju. Kako se za simulaciju najčešće koristi računar, to se pod modelom može podrazumevati skup instrukcija (program) koji služi da se generiše ponašanje simuliranog sistema. Računar je uređaj koji je sposoban da izvrši instrukcije modela, koje na bazi ulaznih podataka generišu razvoj modela u vremenu.

Relacija modeliranja odnosi se na proces utvrđivanja stepena slaganja podataka o realnom sistemu sa podacima modela, odnosno ispravnost modela. Relacija simulacije odnosi se na proveru da li simulacioni program verno prenosi model na računar kao i na tačnost kojom računar izvršava instrukcije modela.

4. SINTEZA ARHITEKTURE RAČUNARA

Problem sinteze arhitekture računara podrazumeva definisanje potrebnih aritmetičkih i logičkih operacija, realizacije ulaza i izlaza, rada sa memorijskim lokacijama, registrima i definisanje ostalih funkcija koje omogućavaju manipulaciju podacima. Da bi se analizirao rad arhitekture računara može se poći od ograničenog skupa instrukcija koje omogućavaju osnovne operacije. U ovom radu su korišćene instrukcije koje su neophodne za realizaciju sabiranja dva binarna broja.

Za realizaciju aritmetičke operacije sabiranja potrebna su dva binarna broja nad kojim se izvršava ova operacija, dva registra i aritmetičko-logička jedinica (ALU) opšte namene. Brojevi se unose u sistem preko memorije, dok se rezultat sabiranja šalje takođe u memoriju. Neophodna hardverska struktura pomoću koje se obavlja aritmetička operacija sabiranja data je na slici 2.



Slika 2. Unošenje podataka, pomeranje, sabiranje i generisanje rezultata

Prikaz programske sekvence pomoću koje je realizovan unos, pomeranje, sabiranje i slanje u memoriju dat je u tabeli 1.

Tabela 1. Programska sekvenc

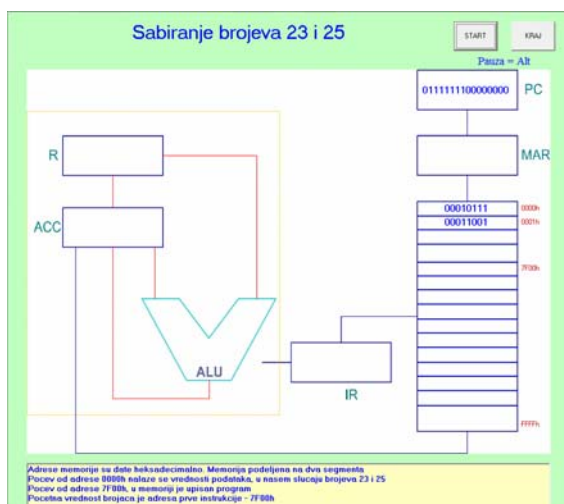
<i>LDAC 0</i>	$(ACC) \leftarrow \text{Memorija (prvi broj)}$
<i>MVAC</i>	$(R) \leftarrow (ACC)$
<i>LDAC 1</i>	$(ACC) \leftarrow \text{Memorija (drugi broj)}$
<i>ADD</i>	$(ACC) \leftarrow (ACC) + (R)$
<i>STAC 2</i>	$\text{Memorija (rezultat)} \leftarrow (ACC)$

Brojevi koje treba sabrati nalaze se u memoriji na lokacijam 0 i 1. Da bi se prvi broj uneo iz memorije u sistem potrebno je koristiti mašinsku instrukciju LDAC 0 (iz memorijske lokacije 0, učitava se prvi broj i šalje u akumulator ACC). Da bi drugi broj uneli u akumulator potrebno je sadržaj akumulatora (prvi broj) prebaciti u pomoćni registar R. To se postiže instrukcijom MVR (sadržaj akumulatora se kopira u registar R). Potom se drugi broj, na isti način, unosi u sistem instrukcijom LDAC 1 (iz memorijske lokacije 1, učitava se drugi broj i šalje u akumulator ACC). Sada su brojevi spremni da se nad njima izvrši operacija sabiranja, pomoću ALU i mašinske instrukcije ADD (sadržaj akumulatora se sabira sa sadržajem registra i smešta se u akumulator). Na kraju, rezultat šaljemo u memoriju, instrukcijom STAC 2 (rezultat sabiranja koji je u akumulatoru, smešta se u memoriju, u memorijsku lokaciju 2)

5. REALIZACIJA PROGRAMSKE SEKVENCE

Aplikacija koja je opisana u ovom radu demonstrira korisnicima na primeru sabiranja dva broja, simulirajući korak po korak, tok podataka. Program je urađen na operativnom sistemu Microsoft Windows XP Professional korišćenjem programskog jezika Microsoft Visual Basic 6.0. Za demonstraciju sabiranja dva broja koriste se brojevi, 23 i 25. Njihovi binarni zapisi su 10111 i 11001. Po pokretanja aplikacije dobija se prozor kao na slici 3.

Prozor ove aplikacije osim dva tastera, <START> i <KRAJ> sadrži:



Slika 3. Početni prozor aplikacije

- Dva 8-bitna registra: akumulator ACC i registar R.
- Memoriju, koja je podeljena na dva segmenta. U prvom segmentu, počev od adrese 0000h (h-heksadecimalno) nalaze se vrednosti podataka, osmobitnih brojeva 23 (00010111) i 25 (00011001). U drugom segmentu, počev od adrese 7F00h (0111111100000000), u memoriju su upisane 16-bitne instrukcije. Kako je dužina memorijske reči 8 bita (1 bajt), to se instrukcije moraju dohvatiti u okviru 2 memorijska ciklusa (2 bajta). Koriste se dva tipa instrukcija:
 - Instrukcije sa operandom (LDAC i STAC). Ove instrukcije koriste direktno memorijsko adresiranje (obraćaju se memoriji) i prvi bajt ovih instrukcija je kôd instrukcije, dok je drugi bajt rezervisan za adresni deo instrukcije.
 - Instrukcije bez operandada (MVAC i ADD). Ove instrukcije koriste registarsko

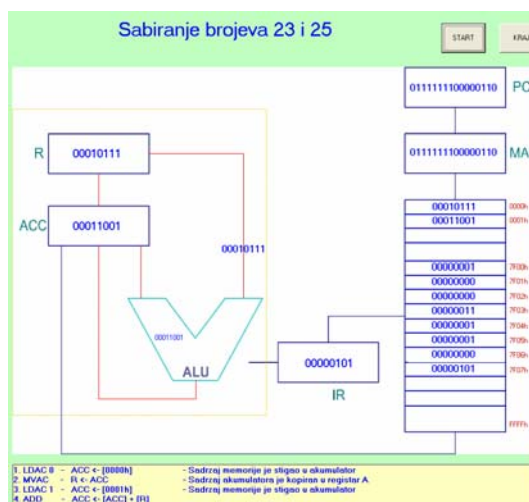
adresiranje i prvi bajt ovih instrukcija je instrukcija bez dejstva, dok je drugi bajt kôd instrukcije.

U memoriji je zapisan program dat u tabeli 1:

- LDAC 0 – iz memorijske lokacije 0, učitava se broj i šalje u akumulator ($ACC \leftarrow [0000h]$), kôd instrukcije je 00000001
- MVAC – sadržaj akumulatora se kopira u registar ($R \leftarrow ACC$), kôd instrukcije je 00000011
- LDAC 1 - iz memorijske lokacije 1, učitava se broj i šalje u akumulator ($ACC \leftarrow [0001h]$)
- ADD – sadržaj akumulatora se sabira sa sadržajem registra i smešta se u akumulator ($ACC \leftarrow [ACC] + [R]$), kôd instrukcije je 00000101
- STAC 2 – rezultat sabiranja koji je u akumulatoru, smešta se u memoriju, u memorijsku lokaciju 2 ($[0002] \leftarrow [ACC]$), kôd instrukcije je 00000010
- 16-bitni programski brojač PC, početne adrese 7F00h, adresa prve instrukcije. Uvećava za 1 za svaki novi memorijski ciklus i sadrži adresu instrukcije koja čeka na izvršenje.
- Memorijsko adresni registar MAR, koji čuva adresu memorijske lokacije koja se adresira. Njegova dužina je 16 bita, što znači postoji mogućnost adresiranja 2¹⁶-1 KB, tj. kapacitet memorije je 64 KB.
- 8-bitni registar instrukcija IR interpretira instrukciju koja je zadnja doneta, tako što prihvata prvi (drugi) bajt instrukcije i u zavisnosti od sadržaja aktivira odgovarajući izlaz.
- Aritmetičko – logičku jedinicu (ALU), koja u ovom primeru, izvršava operaciju sabiranja ADD.

Nakon potvrđivanja tastera <START>, kreće izvršavanje simulacije. Početna vrednost programskog brojača je adresa prve instrukcije (LDAC 0). MAR prihvata adresu iz PC-a i pronalazi odgovarajuću memorijsku lokaciju. Sadržaj memorijske lokacije se šalje u IR, dekodira se i u zavisnosti od instrukcije izvršava se akcija. Kako je prva instrukcija LDAC 0 treba memorijski sadržaj iz lokacije 0 proslediti u akumulator. Instrukcija se izvršava u dva memorijska ciklusa. Nakon završetka ove instrukcije prvi broj (23) je u akumulatoru.

Da bi sabrali dva broja koristi se akumulator ACC i registar R. Prvi broj, koji je u akumulatoru, treba sačuvati i sabrati sa drugim brojem. Zato se prvi broj prosleđuje registru R, instrukcijom MVAC. Potom se akumulator puni drugim brojem (25), instrukcijom LDAC 1. Na taj način, prvi broj se nalazi u registru R, dok je drugi broj u akumulatoru. Adresira se sledeća instrukcija (ADD), IR šalje signal ALU-u da treba da sabere brojeve koji se nalaze u registrima (ACC i R). Na slici 4 dat je trenutak izvršavanja instrukcije sabiranja ADD, kada se ALU-u prosleđuju sadržaji akumulatora ACC i registra R, a potom se izvršava operacija sabiranja i rezultat će nakon toga biti smešten u akumulator.



Slika 4. Izvršavanje instrukcije ADD

Nakon izvršenja instrukcije sabiranja rezultat treba poslati, instrukcijom STAC 2, u memoriju na lokaciju 2. Ovo je poslednja instrukcija programa zapisanog u memoriji. Slika 5 daje završni prikaz nakon realizacije programa. Simulacija se može privremeno prekidati (pauzirati) potvrđivanjem na tastaturi tastera Alt.

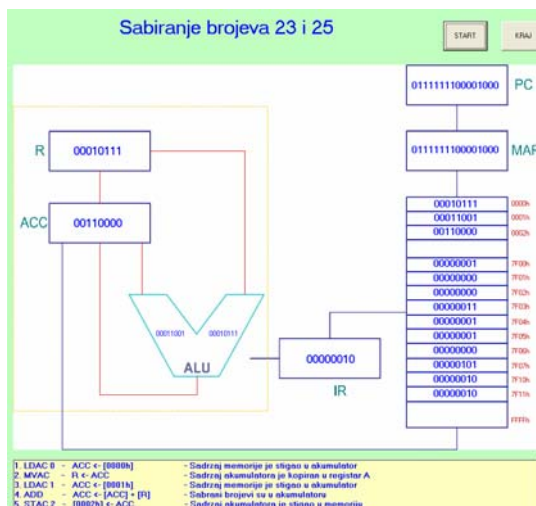
6. ZAKLJUČAK

Složenost i varijacije hardvera računara svakodnevno se povećavaju, pa se za direktno proučavanje određene arhitekture smanjuju mogućnosti i interes. Zbog toga umesto ka realnom sistemu, mnogi se okreću ka simulatorima kao pomoćnim sredstvima u podučavanju o arhitekturi i funkcijama računara. Postoje mnogi simulatori koji su dostupni na Internetu. Analizom dostupnih radova dolazi se do zaključka da su postojeći simulatori projektovani u različite svrhe i za različite kurseve.

U ovom radu je pokazano kako se na vizuelan način, kroz elementarne korake, može prikazati postupak izvršavanja aritmetičke operacije sabiranja. Cilj rada je bio da se korisnicima pruži kompletna informacija i obuka na najnižem nivou.

7. LITERATURA

- [1] Stanković N.: Prilog simulaciji računarskih arhitektura, Magistarski rad, Tehnički fakultet, Čačak, 2009.
- [2] Stanković, N., Marković G., Marković B.: Simulacija rada ALU-a kao osnovne komponente CPU-a, V međunarodni naučni skup „Tehnologija, informatika i obrazovanje za društvo učenja i znanja“, Novi Sad, Zbornik radova Simpozijum TIO5, Fakultet Tehničkih nauka, Novi Sad, 2009.
- [3] Stanković, N., Randić, S.: Simulation of the TFaCo processor, 13th International Research/Expert Conference "Trends in the Development of Machinery and Associated Technology", TMT 2009, Hammamet, Tunisia, 16-21 October 2009. pp. 353-356, Tunisia, 2009.
- [4] Radenković, B., Stanojević M. i Marković A.: Računarska simulacija, Fakultet organizacionih nauka i Saobraćajni fakultet, Beograd, 1999.
- [5] Stojčev, M.: RISC, CISC i DSP procesori, Elektronski Fakultet, Niš, 1997.
- [6] Maxwell, T., Scott, B.: Visual Basic Super Bible, Corte Madera, California, 1992.
- [7] Đorđević, J.: Arhitektura računara, Edukacioni računarski sistem, Arhitektura i organizacija računarskog sistema, ETF, Beograd, 2003.
- [8] Simulation of Little Man Computer, Illinois State Univesity, 2004., <http://www.acs.ilstu.edu/faculty/javila/lmc/>



Slika 5. Program je izvršen