

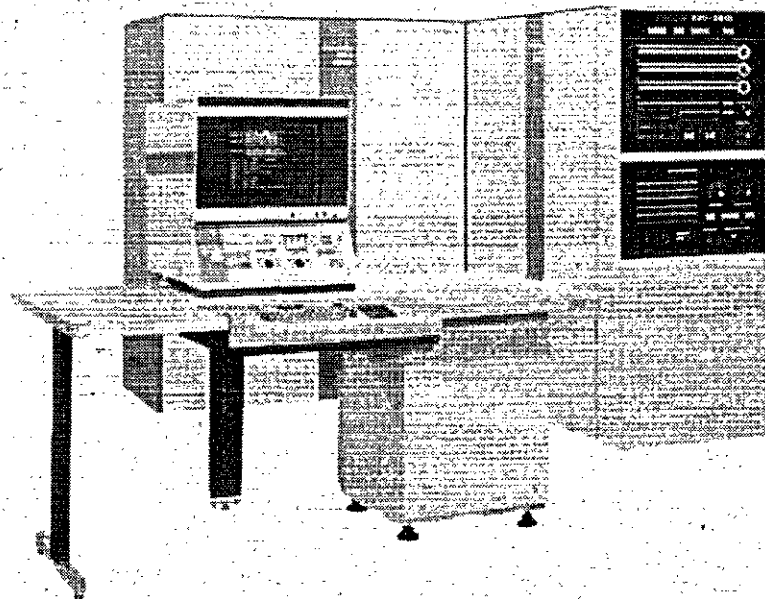
INFORMATICA

YU ISSN 0350-5596



1978

porodica facom firme fujitsu



Radeći bez mnogo buke, ali marljivo posljednje četiri godine, FUJITSU je zajedno sa svojim zastupnikom ZPR-om (Zavod za primjenu elektroničkih računala) uspješno sklopila ugovore za više od 50 FACOM računala u Jugoslaviji.

Iznenadjeni? Ne morate biti. FUJITSU, povrh toga što je vodeći proizvođač sistema za elektroničku obradu podataka na Japanskom tržištu, vrlo brzo preuzima jedno od vodećih mjesta i na svjetskom tržištu. Tajna uspjeha firme FUJITSU je u tome što ima vodeću tehnologiju u kombinaciji sa velikom pouzdanošću sistema i dobro organiziranom službom za održavanje i stručnu pomoć. Ne smijemo zaboraviti ni konkurentne cijene koje će vam dati najbolji mogući odnos cijena/performance.

Ako razmišljate o uvođenju elektronske obrade podataka u vašoj organizaciji ili želite da poboljšate svoj sadašnji sistem, obratite se predstavnicima firme FUJITSU da vas upoznaju sa svim novostima.

FUJITSU proizvodi sve - sastavne dijelove, memorije, off i on-line uredjaje za prikupljanje podataka, inteligentne terminale, micro procesore i micro računala, malih, srednjih, velikih i super velikih kompjutera, uključujući najsnaznije kompjutere za opću svrhu koji se mogu kupiti na tržištu.

FUJITSU je poznata i u području telekomunikacija. To je razumljivo zbog toga što je FUJITSU jedan od vodećih proizvođača telefona i telekomunikacija u Japanu.

FUJITSU je u Jugoslaviji izabrana da snabdije i pomogne kod razvoja najveće on-line real-time mreže koja je do sada ugovorena, uključujući oko 300 terminala.

Mislimo da biste sebi i svojoj organizaciji učinili mnogo, ako saznate više o tome što Vam firma FUJITSU može ponuditi.

Servisni centri i uredi su u Ljubljani, Mariboru, Beogradu i Zagrebu

ZPR
ZAGREB, Savska 56
tel. 518-706
Telex 21689 YU ZPR FJ



LJUBLJANA, Topniška 45
tel. 311-059

FUJITSU LIMITED
Communications and Electronics

INFORMATICA

časopis za tehnologijo računalništva in
probleme informatike
časopis za računarsku tehnologijo i pro-
bleme informatike
spisanie za tehnologija na smetanjeto i
problemi od oblasta na informatikata

YU ISSN 0350-5596

Časopis izdaja Slovensko društvo INFORMATIKA,
61000 Ljubljana, Jamova 39, Jugoslavija

UREJNIŠKI ODBOR:

Člani: T. Aleksić, Beograd, D. Bitrakov, Skopje, P. Dra-
gojlović, Rijeka, S. Hodžar, Ljubljana, B. Horvat,
Maribor, A. Mandžić, Sarajevo, S. Mihalić, Va-
raždin, S. Turk, Zagreb.

Glavni in odgovorni urednik: A.P. Železnikar

TEHNIŠKI ODBOR:

Uredniki področij:

- V. Batagelj - programiranje
- I. Bratko - umetna inteligenca
- D. Čeček-Kecmanović - informacijski sistemi
- M. Exel - operacijski sistemi
- A. Jerman-Blažič - novice založništva
- B. Jerman-Blažič-Džonova - literatura in srečanja
- L. Lenart - procesna informatika
- D. Novak - mikro računalniki
- N. Papić - študentska vprašanja
- L. Pipan - terminologija
- B. Popovič - novice in zanimivosti
- V. Rajković - vzgoja in izobraževanje
- M. Špegel, M. Vukobratović - robotika
- P. Tancig - računalništvo v humanističnih in družbo-
nih vedah
- S. Turk - materialna oprema

Tehnični urednik: R. Murn

ZALOŽNIŠKI SVET

- T. Banovec, Zavod SR Slovenije za družbeno planiranje,
Ljubljana
- A. Jerman-Blažič, Slovensko društvo Informatika,
Ljubljana
- B. Klemenčič, ISKRA, Elektromehanika, Kranj
- S. Saksida, Institut za sociologijo in filozofijo pri
Univerzi v Ljubljani
- J. Vifrant, Fakulteta za elektrotehniko, Univerza v
Ljubljani

Uredništvo in uprava: 61000 Ljubljana, Institut "Jožef
Stuflar", Jamova 39, telefon (061) 263261, telegram:
JOSTIN, telex: 31 269-YU JOSTIN.

Letna naročnina za delovne organizacije je 300,00 din,
za posameznika 100,00 din, prodaja posamezne številke
50,00 din.

Žiro račun: 50101-678-51841

Stališče uredništva se lahko razlikuje od mnenja avtorjev.

Na podlagi mnenja Republiškega sekretariata za prosveto
in kulturo št. 4210-151/77 z dne 4.5.1977, je časopis
INFORMATICA strokovni časopis, ki je oproščen temelj-
nega davka od prometa proizvodov.

Tisk: Tiskarna Krestija, Ljubljana

Grafična oprema: Eltonir Simončič

Letnik 2, 1978 - številka 1

VSEBINA

- B. Hladnik
B. Jarc 6 Proces izgradnje velikih telekomu-
nikacijskih sistemov s pomočjo
tehnik projektne vodjenja
- D. Novak
M. Kovačević
A. Železnikar 14 Mikro računalnik "VITA" s proce-
sorjem 6800
- D. Davčev 21 Prilog kon metodelo na vrednuvanje
na APL sistemite
- A.P. Železnikar 25 Uporaba časovnikov in števnikov v
mikroprocesorskih sistemih s pro-
cesorjem Z-80
- F. Košir 34 Avtomatska obdelava receptov v
SR Sloveniji
- R. Murn
D. Peček 42 Gibki disk
- J. Eržen 46 Računalniški model železniškega
prometa na enotirni progi
- A. Hadži
M. Kovačević
A. Železnikar 51 Audio kasetofon kot cenena vhodno/
izhodna enota za mikro računalnike
- A. Jerman-
Blažič 57 Domaća proizvodnja računarske opre-
me i opreme za prenos podataka
- R. Reinhardt 59 Republiška tekmovanja srednješolcev
iz računalništva
- 67 Novi časopis "Information & Mana-
gement"
- 69 Enostavni stabilizirani usmernik za
mikro računalnike
- 70 Mikroročunalniška integrirana vezja
- 74 Literatura in srečanja
- 77 Novice in zanimivosti

INFORMATICA

Journal of Computing and Informatics

Published by INFORMATIKA, Slovene Society for Informatics, 61000 Ljubljana, Jamova 39, Yugoslavia

EDITORIAL BOARD:

T. Aleksić, Beograd, D. Bitrakov, Skopje, P. Drugojlović, Rijeka, S. Hodžar, Ljubljana, B. Horvat, Maribor, A. Mandžić, Sarajevo, S. Mihalić, Varaždin, S. Turk, Zagreb.

EDITOR-IN-CHIEF:

A. P. Železnikar

TECHNICAL DEPARTMENTS EDITORS:

V. Batagelj - Programming
I. Bratko - Artificial Intelligence
D. Čeček-Kecmanović - Information Systems
M. Exel - Operating Systems
A. Jerman-Blažič - Publishers News
B. Jerman-Blažič-Džonova - Literature and Meetings
L. Lenart - Process Informatics
D. Novak - Microcomputers
N. Papić - Student Matters
L. Pipan - Terminology
B. Popović - News
V. Rajković - Education
M. Spiegel, M. Vukobratović - Robotics
P. Tancig - Computing in Humanities and Social Sciences
S. Turk - Hardware

EXECUTIVE EDITOR:

R. Murn

PUBLISHING COUNCIL:

T. Banovec, Zavod SR Slovenije za družbeno planiranje, Ljubljana
A. Jerman-Blažič, Slovensko društvo informatika, Ljubljana
B. Klemenčič, ISKRA Elektromehanika, Kranj
S. Saksida, Institut za sociologijo in filozofijo pri Univerzi v Ljubljani
J. Virant, Fakulteta za elektrotehniko, Univerza v Ljubljani

Headquarters: 61000 Ljubljana, Institut "Jožef Stefan", Jamova 39, Phone: (061) 263261, Cable: JOSTIN Ljubljana, Telex: 31 269 YU JOSTIN

Annual subscription rate for abroad is US \$ 18 for companies, and US \$ 6 for individuals.

Opinions expressed in the contributions are not necessarily shared by the Editorial Board.

Printed by: Tiskarna Kreslja, Ljubljana

DESIGN: Tihomir Simončič

Volume 2, 1978 - no. 1

CONTENTS

- | | |
|---|--|
| B. Illadnik
B. Jarc | 6 Process of Building-up Extensive Telecommunication Systems by Application of Project Control |
| D. Novak
M. Kovačović
A. Železnikar | 14 The Microcomputer "VITA" Using 6800 Processor |
| D. Davčev | 21 Contribution to the Methods of Evaluation for APL Systems |
| A. P. Železnikar | 25 Usage of Timers and Counters in Z-80 Microprocessing Systems |
| F. Košir | 34 Automatic Prescription Processing (APP) in SR Slovenia |
| R. Murn
D. Poček | 42 Floppy Disk Systems |
| J. Eržen | 46 Traffic Computer Simulation of Single Track Railway |
| A. Hadži
M. Kovačović
A. Železnikar | 51 Application of an Audio Cassette Recorder for a Microcomputer I/O Unit |
| A. Jerman-Blažič | 57 Production of Computer and Data Communication Equipment in Yugoslavia |
| R. Reinhardt | 59 Computer Science Contests for Students of Schools at Medium Level |
| | 67 "Information & Management", a New Journal |
| | 69 A Simple Regulated Power Supply for Microcomputers |
| | 70 Microcomputer Integrated Circuits |
| | 74 News |
| | 77 Literature and Meetings |

navodilo za pripravo članka

Avtorje prosimo, da pošljejo uredništvu naslov in kratek povzetek članka ter navedejo približen obseg članka (število strani A 4 formata). Uredništvo bo nato poslalo avtorjem ustrezno število formularjev z navodilom.

Članek tipkajte na priložene dvokolonske formularje. Če potrebujete dodatne formularje, lahko uporabite bel papir istih dimenzij. Pri tem pa se morate držati predpisanega formata, vendar pa ga ne vrišite na papir.

Bodite natančni pri tipkanju in temeljiti pri korigiranju. Vaš članek bo s foto postopkom pomanjšan in pripravljen za tisk brez kakršnihkoli dodatnih korektur.

Uporabljajte kvaliteten pisalni stroj. Če le tekst dopušča uporabljajte enojni presledek. Črni trak je obvezen.

Članek tipkajte v prostor obrobljen z modrimi črtami. Tipkajte do črt - ne preko njih. Odstavek ločite z dvojnimi presledkom in brez zamikanja prve vrstice novega odstavka.

Prva stran članka:

- v sredino zgornjega okvira na prvi strani napišite naslov članka z velikimi črkami;
- v sredino pod naslov članka napišite imena avtorjev, ime podjetja, mesto, državo;
- na označenem mestu čez oba stolpca napišite povzetek članka v jeziku, v katerem je napisan članek. Povzetek naj ne bo daljši od 10 vrst.
- če članek ni v angleščini, ampak v katerem od jugoslovanskih jezikov izpusite 2 cm in napišite povzetek tudi v angleščini. Pred povzetkom napišite angleški naslov članka z velikimi črkami. Povzetek naj ne bo daljši od 10 vrst. Če je članek v tujem jeziku napišite povzetek tudi v enem od jugoslovanskih jezikov;
- izpusite 2 cm in pričnite v levo kolono pisati članek.

Druga in naslednje strani članka:

Kot je označeno na formularju začnite tipkati tekst druge in naslednjih strani v zgornjem levem kotu,

Naslovi poglavij:

naslove ločuje od ostalega teksta dvojni presledek.

Če nekaterih znakov ne morete vpisati s strojem jih čitljivo vpišite s črnim črnilom ali svinčnikom. Ne uporabljajte modrega črnila, ker se z njim napisani znaki ne bodo preslikali.

Ilustracije morajo biti ostre, jasne in črno bele. Če jih vključite v tekst, se morajo skladati s predpisanim formatom. Lahko pa jih vstavite tudi na konec članka, vendar morajo v tem primeru ostati v mejah skupnega dvokolonskega formata. Vse ilustracije morate (nalepiti) vstaviti sami na ustrezno mesto.

Napake pri tipkanju se lahko popravljajo s korekcijsko

folijo ali belim tusem. Napačne besede, stavke ali odstavke pa lahko ponovno natipkate na neprozoren papir in ga pazljivo nalepite na mesto napake.

V zgornjem desnem kotu izven modro označenega roba oštevilčite strani članka s svinčnikom, tako da jih je mogoče zbrisati.

Časopis INFORMATICA

Uredništvo, Institut Jožef Stefan, Jamova 39, Ljubljana

Naročam se na časopis INFORMATICA za leto 1978 (štiri številke). Predplačilo bom izvršil po prejemu vaše položnice.

Cenik: letna naročnina za delovne organizacije 300,00 din, za posamezniku 100,00 din.

Časopis mi pošiljajte na naslov stanovanja delovne organizacije.

Príimek.....

Ime.....

Naslov stanovanja

Ulica.....

Poštna številka _____ Kraj.....

Naslov delovne organizacije

Delovna organizacija.....

Ulica.....

Poštna številka _____ Kraj.....

Datum..... Podpis:

instructions for preparation of a manuscript

Authors are invited to send in the address and short summary of their articles and indicate the approximate size of their contributions (in terms of A 4 paper). Subsequently they will receive the outor's kits.

Type your manuscript on the enclosed two-column-format manuscript paper. If you require additional manuscript paper you can use similar-size white paper and keep the proposed format but in that case please do not draw the format limits on the paper.

Be accurate in your typing and through in your proof reading. This manuscript will be photographically reduced for reproduction without any proof reading or corrections before printing.

Use a good typewriter. If the text allows it, use single spacing. Use a black ribbon only.

Keep your copy within the blue margin lines on the paper, typing to the lines, but not beyond them. Double space between paragraphs.

First page manuscript:

- a) Give title of the paper in the upper box on the first page. Use block letters.
- b) Under the title give author's names, company name, city and state - all centered.
- c) As it is marked, begin the abstract of the paper. Type over both the columns. The abstract should be written in the language of the paper and should not exceed 10 lines.
- d) If the paper is not in English, drop 2 cm after having written the abstract in the language of the paper and write the abstract in English as well. In front of the abstract put the English title of the paper. Use block letters for the title. The lenght of the abstract should not be greater than 10 lines.
- e) Drop 2 cm and begin the text of the paper in the left column.

Second and succeeding pages of the manuscript: As it is marked on the paper, begin the text of the second and succeeding pages in the left upper corner.

Format of the subject headings: Headings are separated from text by double spacing.

If some characters are not available on your typewriter write them legibly in black ink or with a pencil. Do not use blue ink, because it shows poorly.

Illustrations must be black and white, sharp and clear. If you incorporate your illustrations into the text keep the proposed format. Illustration can also be placed at the end of all text material provided, however, that they are kept within the margin lines of the full size two-column format. All illustrations must be placed into appropriate positons in the text by the author.

Typing errors may be corrected by using white correction paint or by retyping the word, sentence or paragraph on a piece of opaque, white paper and pasting it nearly over errors

Use pencil to number each page on the upper-right-hand corner of the manuscript, outside the blue margin lines so that the numbers may be erased.

Časopis INFORMATICA
Uredništvo, Institut Jožef Stefan, Jamova 39, Ljubljana

Please enter my subscription to INFORMATICA for the volume 1978 (four issues), and send me the bill. Annual subscription price: companies 300,00 din (for abroad US \$ 18), individuals 100,00 din (for abroad US \$ 6)

Send journal to my home address company's address.

Surname.....

Name.....

Home address

Street.....

Postal code _____ City.....

Company address

Company.....

.....

Street.....

Postal code _____ City.....

Date..... Signature

.....

v drugo leto izhajanja časopisa informatica

Časopis Informatica je z zadovoljivim vsebinskim, finančnim in naročniškim uspehom končal svoje prvo leto izhajanja. Na 288 straneh štirih števil je bilo objavljenih 48 strokovnih prispevkov (originalnih člankov, preglednih strokovnih poročil), vrsta recenzij o strokovnih knjigah in člankih, velika količina obvestil o strokovnih prireditvah doma in po svetu (simpoziji, koference, šole, srečanja) ter o razvojnih in raziskovalnih projektih, poročila iz gospodarskih organizacij, najnovejši dosežki na področju tehnologije (novice o izdelkih in proizvajalcih, materialna in programska oprema, tehnološki postopki), naslovi računalniških klubov po svetu, podatki o avtorjih in sodelavcih časopisa Informatica ter oglasi uglednih podjetij in ustanov na področju računalniške tehnologije in informatike. V tem okviru je bil v celoti udejanjen začetni načrt časopisa, ki je predvideval izdajo štirih števil s po 72 stranmi ter usmeritev časopisa v empirične (praktične, tehnične, pregledne) prispevke z najširšega (tudi multidisciplinarnega) delovnega področja.

Pravkar ugotovljena uspešnost pa seveda ne pomeni, da ni bilo ali da ne bo nekaterih izredno perečih problemov. Vprašanje posebnega interesa je premalo intenzivno sodelovanje ter pridobivanje novih, dodatnih avtorjev iz obsežnega problemskega prostora tehničnih in gospodarskih dejavnosti, zlasti tistih v proizvodnji, ki so povezane z razvojem novih izdelkov domače računalniške in druge industrije ter njenega usmerjanja. Vrsta izdelkov, ki se že proizvajajo, ni bila strokovno ocenjena in prikazana v časopisu, ki bi v zadovoljstvo uporabnikov časopisa objavil take prispevke. Premalo je tudi prispevkov, zopet iz industrijske dejavnosti, ki bi obravnavali najnovejši lasten razvoj, pa organizacijske, načrtovalne, tehnološke in druge pomožne postopke. Skratka časopisu manjka inovativni faktor oziroma inovativna vsebina, ki bi iz proizvodnje vzpodbujala razvoj novih izdelkov in obratno. Brez dodatnega prizadevanja strokovnjakov iz industrije tega problema ne bo moč rešiti v tem letu izdajanja časopisa Informatica.

Založniški svet časopisa Informatica je na svoji zadnji seji v maju analiziral in ocenil z uspehom dosedanje izdajanje časopisa ter priporočil uredništvu naslednje dodatne naloge:

- Časopis Informatika naj objavlja prispevke, novice in povzetke iz drugih, sorodnih strokovnopravnih časopisov (izdaje republiških in pokrajinskih biltenov in časopisov s področja informatike).

- Občasno naj se objavijo tudi prispevki s področja planiranja in analize informacijskih sistemov, obstoječih in tistih v izgradnji.

- Občasno se v časopisu objavijo povzetki, pregledi, analize in strokovne recenzije opravljenih in tekočih raziskovalnih in razvojnih nalog, še posebej istih, ki se financirajo ali sofinancirajo z družbenimi sredstvi.

- Pri oglasih gospodarskih organizacij se doda izresek za neposredno komuniciranje s proizvajalcem-ponudnikom.

- Predlaga se honoriranje prispevkov v letu 1978.

Uredništvo časopisa Informatika bo izdajatelju časopisa predlagalo začasno honoriranje vsakršnih objavljenih prispevkov v znesku

600 din

na avtorsko polo (20 000 znakov). To pomeni, da bodo strokovni in vsi ostali (informativni) objavljeni prispevki (recenzije, novice, poročila) honorirani že s prvo številko Informatica 1978.

V naslednjih številkah časopisa si bo uredništvo prizadevalo, da se prikažejo proizvodni načrti ter podatki o asortimanu jugoslovanske računalniške in spremljajoče industrije (EI, Iskra, R. Čajevec, Elektrotehna, Impuls, Unis, TRS, Digitron, Triso, Gorenje, Energoinvest, Selk, RIZ itn.), o industrijskem sodelovanju s tujimi partnerji, o lastnem razvojnem delu, vlaganjih in organizaciji. V tej zvezi bodo bržkone zanimive t.i. celovite ponudbe, ki jih dajejo združeni partnerji, s pregledom materialne in programske opreme, razvojne sistemske podpore, servisnih uslug, šolanja strokovnjakov in tržnih dosežkov. Še zlasti bodo konec leta zanimivi podatki o doseženi finančni, proizvodni, kadrovske, razvojni in tržni realizaciji te veje domače industrije.

K vsemu temu dodajmo na koncu še tole: Časopis Informatica bo tudi v naslednjem obdobju namenjen delavcem na področju računalniške tehnologije in informatike. Vsebinsko časopisa bo moč dopolniti, popestriti in aktualizirati samo z aktivnim sodelovanjem večjega števila uporabnikov časopisa; to sodelovanje bo odslej honorirano. Iz konteksta časopisa nikakor ne bodo izvzeti prispevki drugih interesentov, kot so amaterji, ljubitelji, študenti, učitelji, tehnični strokovnjaki in javni delavci; njihovi prispevki bodo deležni vse potrebne pozornosti.

Vaš urednik:

L. Felcman

proces izgradnje velikih telekomunikacijskih sistemov s pomočjo tehnike projektnega vodenja

b.hladnik
b.jarc

UDK.621.39

Iskra Elektromehanika Kranj

Zaradi vse večjega obsega in razvejanosti proizvodnje elektronskih telefonskih central se je Iskra Elektromehanika Kranj odločila za vpeljavo projektnega vodenja oblikovanja elektronskih telefonskih central. Z dosedanjimi metodami planiranja ne moremo več kvalitetno opravljati planskih funkcij in koordinativne dejavnosti. Organiziranje projektne službe je pravzaprav proces, ki zahteva aktivno sodelovanje služb, ki so udeležene pri oblikovanju projekta. V nadaljevanju je ta proces obširneje razčlenjen. Začetek predstavlja analiza obstoječega sistema pregleda nad odvijanjem projektov, nadalje je obdelana faza logičnega funkcionalnega zaporedja, zatem naloge službe za vodenje projektov, na kraju teče beseda še o komisiji, ki bo naloge službe obdelala do operativnega nivoja in zagotovila pogoje za njeno aktivno in plodno delovanje.

PROCESS OF BUILDING-UP EXTENSIVE TELECOMMUNICATION SYSTEMS BY APPLICATION OF PROJECT CONTROL TECHNIQUE. As the production of electronic telephone exchanges has now become more increased and complex, Iskra Elektromehanika Kranj decided to introduce the project control of building-up the electronic telephone exchanges. Both planning functions and co-ordination activities can not be properly performed by applying the present methods of planning. Organization of the project service in fact requires active collaboration of services which participate at the project building-up. The paper is further analysing this process more in detail. The analysis of the existing system of control at processing of projects is performed first, then the logical functional sequence and further on the activities that are to be performed by the project control service are treated. Finally this paper deals with the commission which would elaborate the tasks of the service up to operational as well as organizational level and assure the conditions for active and fruitful work.

1. Razlogi, ki so vplivali na uvajanje projektnega vodenja v naši delovni organizaciji

Proizvodnja telekomunikacijskih naprav se je, obogatena z najnovejšimi dognanji in sodobno tehnologijo, razvila v proizvodnjo kompleksnih telekomunikacijskih sistemov. Vsak posamezni sistem ima specifičnosti, ki zahtevajo bolj ali manj obsežen razvoj in mu dajejo pečat unikatata. Zaradi tega se močno poveča obseg faz do pričetka same proizvodnje, pa tudi proizvodnja je bolj zahtevna. Zaradi prilagajanja sistema investitorjevemu omrežju in zaradi neprestanega vnašanja novih dognanj, ki pomenijo kvaliteto in visoko konkurenčnost,

je podvržen stalnim spremembam, kar vsekakor povzroča vlaganje velikih naporov v pravo proizvodnjo in v proizvodnjo samo. Končni proizvod je izgubil značaj izdelka, česar smo bili do nedavnega vajeni, metode planiranja, primerne za izdelčno proizvodnjo, so bile nesprejemljive in so nam narekovale nekoliko drugačno reševanje planskih funkcij.

Proces oblikovanja elektronskih telekomunikacijskih sistemov ima vse značilnosti projekta v širšem smislu, saj pri njegovi realizaciji sodeluje večje število udeležencev s še večjim številom aktivnosti, ki so logično povezane in usmerjene k skupnemu cilju. Skupen cilj je jasno določen, iz

njega izhajajo parcialni cilji udeležencev procesa, ki so postavljeni v skladu z njihovimi lastnimi močmi. Pomembnost opredelitve parcialnih ciljev predstavlja koordinacija dejavnosti posameznih udeležencev.

Tehnika vodenja projektov je bila že velikokrat preizkušena pri načrtovanju in izgradnji velikih energetskih, gradbenih, procesnih in tehnoloških sistemov, tako v tujini kot doma. Napor, ki je bil vložen v tak način izgradnje objektov, se je močno obrestoval, predvsem pa manifestiral v veliko boljši koordinaciji dejavnosti oziroma krajših rokih zaključnih del. Te reference potrjujejo pravilnost naše usmeritve - organiziranja projektnega vodenja.

2. Fotek oblikovanja projekta

Fri oblikovanju telefonske centrale sodeluje več služb. Njihovo vključevanje v projektni proces je opisano v tem poglavju. Glavna teža dejavnosti odpade na prodajno službo, projektivo, razvoj, gospodarsko pripravo proizvodnje in na koncu še na montažno službo.

Telefonska centrala je zelo kompleksen projekt, zato zahteva natančno opredeljene poti prenosa informacij in pravilno koordinacijo dejavnosti. Fri tem pa se večkrat pojavljajo težave, ki v večji ali manjši meri vplivajo na čas dobave. Celotni proces izdelave centrale lahko razdelimo v dve glavni fazi:

- predpogodbeni cikel /sl. 1/
- popogodbeni cikel /sl. 2/

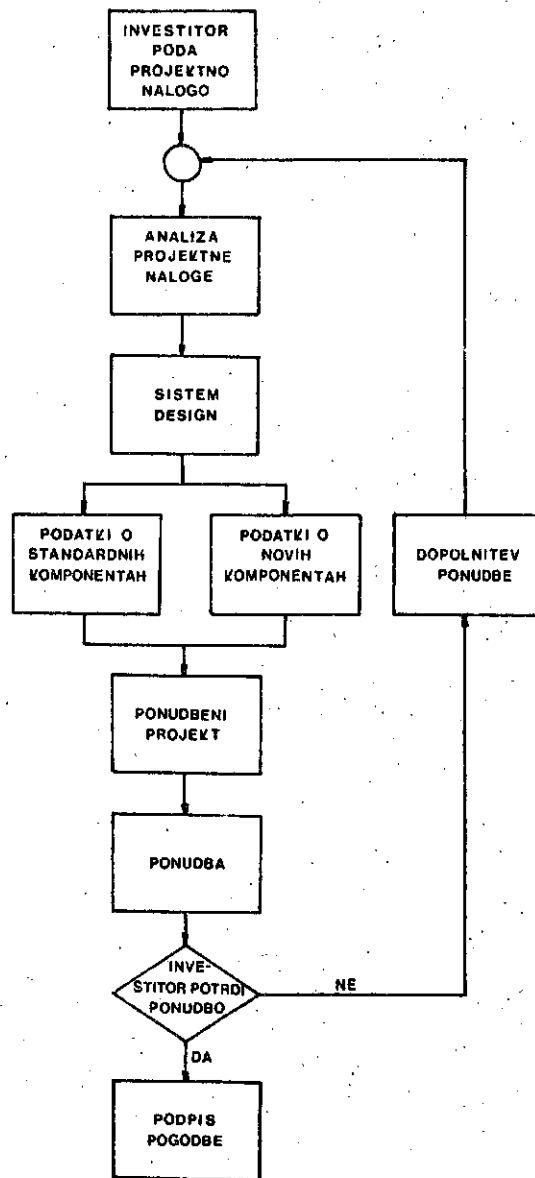
Po drugi strani je možna delitev dejavnosti tudi glede na vrsto obravnavanih elementov:

- standardne komponente, ki imajo že svoje normative in dokumentacijo in
- nerazvite komponente, ki zahtevajo nov razvoj, katerega rezultat je dokumentacija, ki služi za določanje normativov, cene in za naročanje materiala.

Prva delitev časovno določa zaporedje dejavnosti, zato je funkcionalni potek izdelave telefonske centrale obravnavan po tem kriteriju. Poudarek je le na glavnih dejavnostih, ki v grobem nakazujejo celotni funkcionalni potek.

Predpogodbeni cikel

Ta del se nanaša na čas od prvega kontakta kupca s prodajno službo, pa vse do pogodbe. Ko kupec sporoči svoje zahteve prodajni službi, mu pripravi ponudbo. Ponudbeni



sl. 1. Predpogodbeni cikel

cikel ima dve paralelni veji, torej dejavnosti, ki se hkrati izvajajo. Prva veja se nanaša na standardne komponente, drugi del pa obravnava še nerazvite komponente.

Na podlagi podatkov o standardnih in novih komponentah sestavi ponudbo, ki jo kupec pregleda in potrди ali pa zahteva spremembe oziroma dopolnitve. V zadnjem primeru je treba ponudbo ustrezno spremeniti, oziroma dopolniti in potem ponovno predložiti kupcu, da jo potrди ali pa zahteva dodatne spremembe.

Če sprememb ni, sledi podpis pogodbe, ki jo prejme služba za vodenje projektov in jo pošlje vsem prizadetim. Prvi del je s tem končan.

V predpogodbenem ciklu igra glavno vlogo prodaja, ki je v stalnem kontaktu s kupcem in je nosilec dejavnosti v tem obdobju.

Pogodbeni cikel

Po pogodbi poteka izgradnja projekta telefonske centrale po več paralelnih vejah. Ogledajmo si jih po vrsti.

Glavni projekt izdelava Projekktivni biro ali pa PTT sam in je pogoj za pridobitev gradbenega dovoljenja. Prostorski načrt preskrbi prodaja od kupca in je osnova za izdelavo grobe specifikacije montažnega materiala, ki ga prično nabavljati in pripravljati za montažo. V tej veji poteka tudi priprava izvedbenega projekta, ki vsebuje več vrst dokumentacije, potrebne za montažo in pa testiranje.

Paralelno poteka veja, ki se nanaša na standardne komponente. Zanje v projektivi pripravijo specifikacije, zatem pa v gospodarski pripravi proizvodnje prično disponirati oziroma pripravljati podatke za nabavo.

Poleg tega planirajo delo v proizvodnji glede na razpoložljive kapacitete. Ko je material na razpolago, pričnejo proizvodnjo standardne opreme, ki jo potem skladiščijo in pošiljajo na gradbišče.

Tretja paralelna veja se nanaša na opremo, ki zaradi zahtev kupca ali raznih dopolnitev potrebuje razvojno obdelavo, tej pa sledita priprava ustrezne dokumentacije in

proizvodnja opreme.

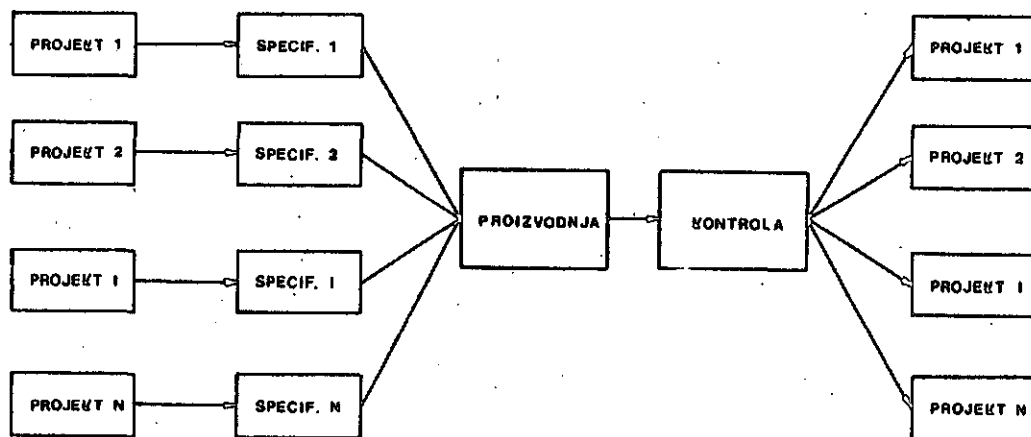
V fazi montaže se vse te veje združijo. Preskusu aparturne in programske opreme sledita tehnični in kvalitativni prevzem ter vključitev centrale v promet.

V našem projektnem procesu želimo zasledovati konkreten projekt skozi vse faze izdelave, vendar se pri tem srečujemo z več problemi. Eden izmed njih je to, da v sami proizvodnji konkreten projekt zapade v anonimnost, ker je trenutno praktično nemogoče zasledovati, kako so realizirani delovni nalogi za posamezne sestavine projekta, ki nas zanima. To problematiko prikazuje slika 2.

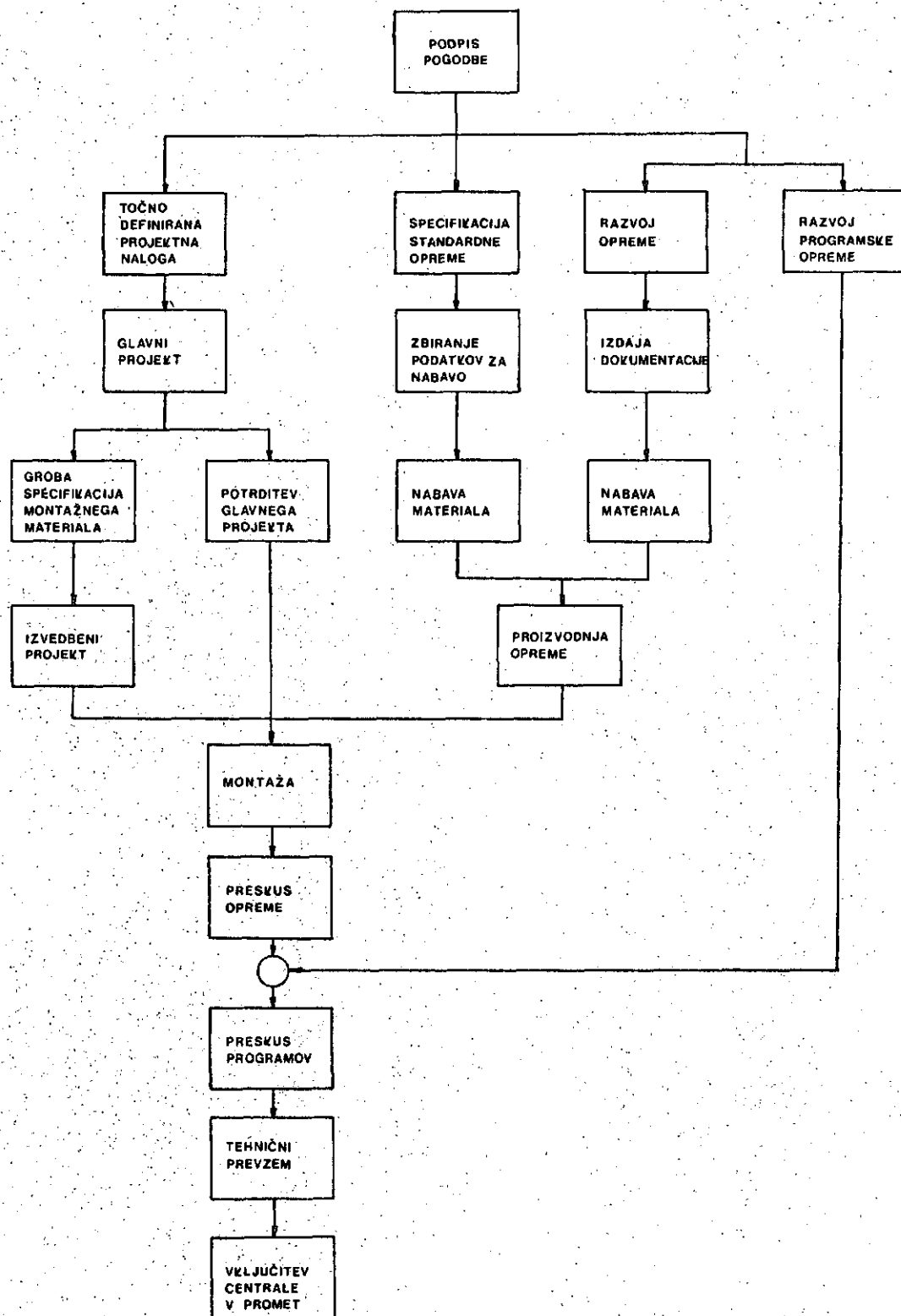
Za projekte v gospodarski pripravi proizvodnje dobijo specifikacije, na osnovi katerih potem planirajo proizvodnjo in združujejo izdelavo posameznih komponent za različne projekte v seriji, tako da pripadnost posameznega izdelka projektu v tej fazi ni znana. Ko pa izdelana komponenta pride v skladišče, dobi svoje "ime" in s tem pripadnost danemu projektu.

Ta problem še rešujemo, ker želimo tudi proizvodno fazo bolj aktivno vključiti v naš zasledovalni proces.

V samem funkcionalnem poteku motnje niso bile omejene, vendar se lahko pojavijo na številnih mestih in povzročajo ponovitev sklopa aktivnosti, kar vse vpliva na dobavni rok. Motnje je težko predvideti vnaprej, zato mora celotna organiziranost projektne procesa omogočati, da jih čim hitreje odpravimo. Tudi glede tega smo pripravili določene rešitve, vendar jih še dopolnjujemo. To sta le dva problema, kar pa ne pomeni, da sta edina.



sl. 2. Združevanje projektov



sl. 5.1. Pogodbeni cikelus

3. Mrežni plan

Za upravljanje in vodenje projektov potrebujemo določen instrumentarij /1/. Izraz instrumentarij združuje več pojmov: metode, algoritme, pomagala in vse ostalo, kar omogoča učinkovito delovanje po principu upravljanja in vodenja projektov. Na izbor instrumentarija vpliva več faktorjev. Navedimo glavne:

- kompleksnost sistema,
- vrsta projekta,
- organiziranost sistema, v katerem projekt nastaja, in organiziranost samega sistema za vodenje in upravljanje projektov.

Pri kontroli izvajanja projekta nam je v pomoč tudi tehnika mrežnega planiranja, ki pa ni edini element instrumentarija.

V zvezi z mrežnim planom za naše projekte je več problemov. Ti projekti so, kot je bilo že večkrat omenjeno, zelo kompleksni, zato se je pojavil glavni problem, kako določiti stopnjo detajlnosti mrežnega plana. Pri tem moramo paziti predvsem na to, da zagotovimo vse potrebne informacije za dejavnosti, ki jih bomo vključili v mrežni plan. Odločili smo se, da ne bomo preveč drobili dejavnosti in smo na začetku uvedli le glavne, torej tiste, za katere lahko zagotovimo vse potrebne podatke.

Problem predstavljajo tudi motnje, kajti v našem projektnem procesu se lahko večkrat pojavljajo motnje in povzročajo ponovitev celega sklopa dejavnosti. Števila ponovitev ni možno predvideti vnaprej. Zato je treba ustrezno modificirati mrežni plan, da ob ažuriranju lahko upoštevamo tudi vse te spremembe.

Odločili smo se za "precedence" - tehniko mrežnega planiranja. Ta tehnika se je izkazala za ugodno zlasti pri dejavnostih, katerih začetki in konci imajo določene časovne premike, takih pa je v našem primeru veliko.

Naloge službe za vodenje projektov v zvezi z mrežnim planom bodo sledeče:

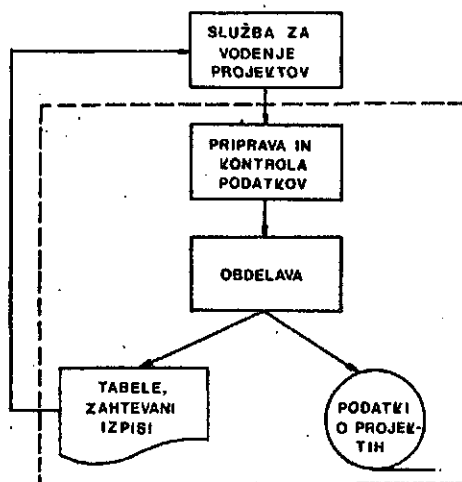
- oblikovanje novega mrežnega plana za zasledovanje novega tipa projektov /modeli/;
- zbiranje podatkov za nov projekt, za katerega je model mrežnega plana že pripravljen;
- ažuriranje mrežnega plana za projekt, ki je že v delu.

Služba za vodenje projektov kontaktira s centrom za ACP in od tod dobiva informacije, ki jih potem posreduje posameznim službam in vodstvenim organom.

Tu gre za informacije o novih projektih, o stopnji realizacije projekta in o rezultatih podatkov ažuriranja.

Za izračun mrežnega plana uporabljamo IBM-paket PCS, vendar smo pripravili še nekaj tabel, ki jih sam paket ne vsebuje.

Povezave med službo za vodenje projektov in centrom za obdelavo podatkov prikazuje slika 4.



sl. 4. Obdelava podatkov o projektih

4. Služba za vodenje projektov

Oblikovanje elektronskega telekomunikacijskega sistema, ki je predmet naše obravnave, imenujemo projektni proces, saj posamezni projekti sestavljajo strnjeno zaporedje odvisnih dejavnosti. Vsi projekti so si po strukturi podobni /tehnologija izvajanja projekta je pri vseh projektih tega procesa poznana, menja se samo objekt/. Ker v projektni proces neprestano vstopajo novi oziroma izstopajo zaključeni projekti, je temu primerno potrebno organizirati tudi skrbništvo.

Odločili smo se, da organiziramo stalno službo za vodenje projektov, ki deluje v okviru planske funkcije delovne organizacije. Naloge službe lahko v grobem razdelimo na:

- plansko-koordinativno in,
- informativno dejavnost.

4.1. V okviru plansko-koordinativne dejavnosti so naslednje naloge:

4.1.1. Sodeluje pri sestavljanju dolgoročnih, srednjeročnih osnovnih in operativnih planov TOZD-ov - udeležencev procesa, iz te naloge neposredno sledi vodenje evidence o zasedenosti projektivnih, razvojnih, proizvodnih in montažnih kapacitet. Ob upoštevanju optimizacije vseh dejavnikov izračunava roke za izvedbo posameznih vrst projektov.

4.1.2. Po sklenitvi pogodbe razčleni

pogodbo do nivoja posameznih organizacijskih enot ob upoštevanju rokov in na osnovi specifikacije. Razčlenitve posreduje TOZD-om kot seštevke večjega števila projektov v skladu z velikostjo in termini posameznih planov. Obdeluje spremembe specifikacij in ugotavlja, ali jih je možno realizirati brez motenja rednega proizvodnega procesa, oziroma predvidi potrebne ukrepe za čim bolj nemoten potek proizvodnje v primeru izjemnega poseganja v to dejavnost.

4.1.3. Tekoče zasleduje realizacijo dejavnosti izvajanja projektov. Analizira odstopanja, predvidi ukrepe in jih predlaga pristojnim organom.

4.1.4. Na osnovi primerjav med plani in dejansko realizacijo predlaga pri sestavljanju planov razširitev posameznih kapacitet, da bi odpravili ozka grla v procesu.

4.1.5. Ob zaključevanju del na objektu se vključuje tudi v kvalitativni in tehnični prevzem ter ob sodelovanju s TOZD-i, ki so udeleženi, določi postopke in roke za odpravo motenj. Naloge, ki slede iz teh predlogov, imajo najvišjo prioriteto.

4.1.6. Koordinira dejavnost pri različnih projektih.

4.1.7. Pripravi in obdela predlog za prioriteto nekega projekta ter ob potrditvi prioritete skrbi za realizacijo do predvidenega termina.

4.1.8. Sistematično vnaša spremembe, ki se pojavljajo v različnih obdobjih in oddelkih.

4.1.9. Obravnava spremembe, daljnosežnost njihovega vpliva in posledice.

4.1.10. Obdeluje mrežni plan projektov.

4.2. Informativna dejavnost obsega:

4.2.1. Ponudbeno in pogodbeno dejavnost prodaje, podatke o zasedenosti kapacitet udeležencev procesa, ki jih bo pridobivala na dva načina:

- črpala iz obstoječih dokumentov in izpisnih tabel,
- vpeljala nove dokumente za specifične podatke.

4.2.2. Ko bodo izdelani mrežni plani za različne vrste projektov /modeli/, bo služba skupno s TOZD določila operativne oblike za pridobivanje informacij.

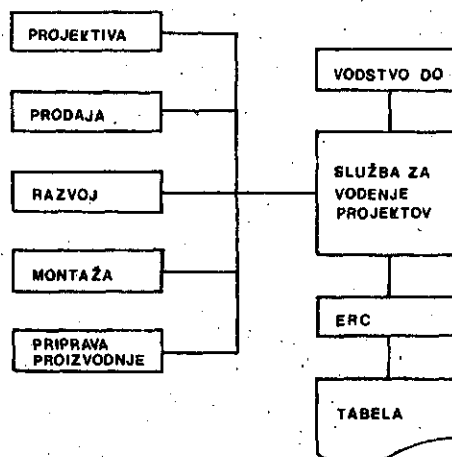
4.2.3. Iz dobljenih in obdelanih podatkov bo posredovala informacije

- mesečno o stanju posameznih objektov TOZD-om in odgovornim v DO,
- podatke o možnih začetkih posameznih faz prizadetim službam,
- podatke o zaključenih objektih,

- Montažno-servisni organizaciji bo posredovala informacijo, da je kupec izpolnil vse potrebne pogoje za pričetek montaže.

Za ažurno periodično zbiranje podatkov je treba vzpostaviti informacijske kanale med službo za vodenje projektov in udeleženci procesa za oblikovanje projektov. V vsakem TOZD deluje nosilec posredovanja informacij, ki je v celoti odgovoren za kvaliteto in predpisano dinamiko posredovanja. Ker deluje v sredini, za katere podatke odgovarja, je njen predstavnik oziroma strokovni sodelavec službe za vodenje projektov za področje svoje sredine /principi matrice organizacije/.

Groba predstavitev informacijskih kanalov je prikazana na sliki 5, iz katere je razvidno, da se vsi informacijski kanali križajo v službi za vodenje projektov. Odtod pa se cepijo na dve veji in sicer gre en kanal v vodstvo DO in drugi v računski center, kjer se informacije strojno obdelujejo. Kanali so za informacije prehodni v obeh smereh.



sl. 5. Informacijski kanali

Na sliki 6 so informacije, ki se prelivajo po zgoraj navedenih kanalih, tudi vsebinsko in časovno opredeljene. Na levi strani so vhodne, na desni izhodne informacije, stebri v sredini slike pa predstavljajo časovno odvijanje procesa. Vrstni red informacij je zaradi tega tudi podan v časovno logičnem zaporedju.

Služba za vodenje projektov je popolnoma nova struktura v organizacijski shemi DO. Da bi se čimbolj vključila v operativno delo in zaživela s polno močjo ter začela uspešno delovati, smo formirali komisijo za uvajanje službe za vodenje projektov. Sestav komisije je tak, da so vanjo vključene vse službe, v katerih sfere bo posegala služba za vodenje

projektov.

Člani komisije so polno odgovorni za dejavnosti in informacije, ki se nanašajo na njihovo službo /matrična organizacija/. Ta njihova angažiranost v času delovanja komisije se prenaša tudi naprej, ko bo pričela delovati služba, sami pa postanejo odgovorni nosilci posredovanja informacij. Komisija je torej začasna struktura, ki bo s formalno postavitvijo službe vse bolj izgubljala vlogo.

Komisija ima določene naloge, lahko pa njihov obseg razširi, če meni, da se bo s tem izboljšala učinkovitost same službe za vodenje projektov. Osnovne naloge so naslednje:

- vzpostavljanje informacijskih kanalov,
- koriščenje obstoječe dokumentacije
- vpeljava novih dokumentov,
- opredelitev detajlnosti informacij,
- periode in obseg informacij
- preizkus postavljenega modela in njegovo izpopolnjevanje,
- normativi posameznih dejavnosti.

5. Povzetek

Zaradi boljšega pregleda nad stanjem projektov bomo lažje preverjali dinamiko odvijanja planskih funkcij in s tem dosegali

tudi enakomernejše zasedanje kapacitet.

S pomočjo službe bomo dosegli boljše koordiniranje udeležencev projektnega procesa in popolnejše informiranje vseh udeleženi struktur. Vsi naštetih faktorji pa se bodo izražali v bolj doslednem izpolnjevanju pogodbenih rokov.

Literatura:

/1/ A.Hauc: Upravljanje projektima, Informator Zagreb 1975

/2/ A.Vila, Z.Leicher: Planiranje proizvodnje i kontrola rokova, Informator Zagreb 1976

/3/ I.Gorenc: Informatika in uporaba avtomatske obdelave podatkov, Moderna organizacija, Kranj 1976

/4/ J.Brandenberger, R. Konrad: Tehnika mrežnog planiranja, Tehnička knjiga, Zagreb 1970

/5/ A.Vila: Planiranje proizvodnje in kontrola rokova /seminarsko gradivo/, Bled 1978

/6/ PCS/360 Program Description and Operations Manual, IBM Application Program

mikroračunalnik "vita" s procesorjem 6800

d.novak
m.kovačević
a.p.železnikar

UDK 681-181.4 VITA

Odssek za računalništvo in informatiko
Institut "Jožef Stefan"

V članku je opisana zgradba mikro računalnika s procesorjem 6800, ki ga lahko štejemo med tipične sisteme za razvoj programov. Računalnik je realiziran na eni plošči dvojnega evropa formata in vsebuje: 16 K zlogov dinamičnega pomnilnika, 4 K zlogov EPROM pomnilnika, 16 vhodno-izhodnih linij, tri časovnike in en serijski komunikacijski kanal.

THE MICROCOMPUTER "VITA" USING 6800 PROCESSOR - This article describes a microcomputer using 6800 processor which represents a typical software development system. This computer is on one board system (double European format) and includes 16 K RAM, 4 K EPROM, two 8 bit parallel ports, three timer channels and one serial I/O channel.

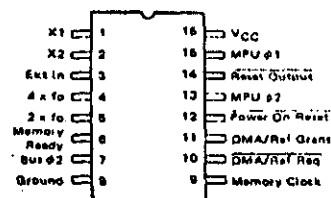
1. UVOD

Mikro računalnik "VITA" predstavlja tipičen sistem za razvoj programov za računalnike s procesorjem 6800. Razvit je bil v želji, da bi z uporabo najmodernejših integriranih vezij dobili zelo bogat sistem na eni sami plošči tiskanega vezja (dvojni evropski format). Pri oblikovanju sistema je bil poudarek na minimizaciji krmilnega vezja za dinamični pomnilnik. Klasični taktni generator za serijski vhodno-izhodni kanal je zamenjalo modernejše integrirano vezje s časovniki, ki jih je mogoče programirati. S tem je bilo pridobljeno precej prostora, poleg tega pa so odpadla stikala za nastavljanje hitrosti. Ker je dal ta manj tradicionalen pristop oblikovanja dobre rezultate, smo smatrali, da bi zgradba mikro računalnika "VITA" utegnila zanimati širšo strokovno javnost. Odločili smo se, da objavimo popolni stikalni načrt. Vse zanimive podrobnosti bodo natančno opisane v tem članku.

Računalnik "VITA" vsebuje: mikro procesor 6800, 16 K zlogov dinamičnega pomnilnika, 4 K zlogov EPROM pomnilnika, 16 vhodnih-izhodnih linij in en serijski komunikacijski kanal. Trenutno ima računalnik monitor, ki obsega 1 K zlogov EPROM pomnilnika. V fazi testiranja pa sta tudi že zbirnik in urejevalnik.

2. TAKTNI GENERATOR XC 6875 (Motorola)

Za generiranje obeh faz sistemskega takta je uporabljen taktni generator 6875 (slika 1.). Ta ima nekaj bistvenih prednosti pred taktnim generatorjem 6871. Taktni generator 6875 je standardno 16-nožično integrirano vezje z razliko od 6871, ki je hibridne izvedbe. Frekvenco je mogoče nastaviti s kvarčnim kristalom (serijska resonanca) ali z RC členom. Najvažnejša prednost pred ostalimi generatorji pa je v tem, da sta na razpolago še signala z dvakratno in štirikratno frekvenco osnovnega sistemskega takta. Te dodatne definicije namreč omogočajo enostavno krmilno vezje za dinamični pomnilnik. Na zivna frekvenca kvarčnega kristala je tedaj štirikrat višja od zahtevane sistemske frekvence. Poleg omenjenih lastnosti omogoča taktni generator XC 6875 še podaljševanje obeh faz takta. Neaktivno fazo ($\Phi 2$ je nizek) podaljšujemo s signalom DMA/Ref Req-, aktivno ($\Phi 2$ je visok)

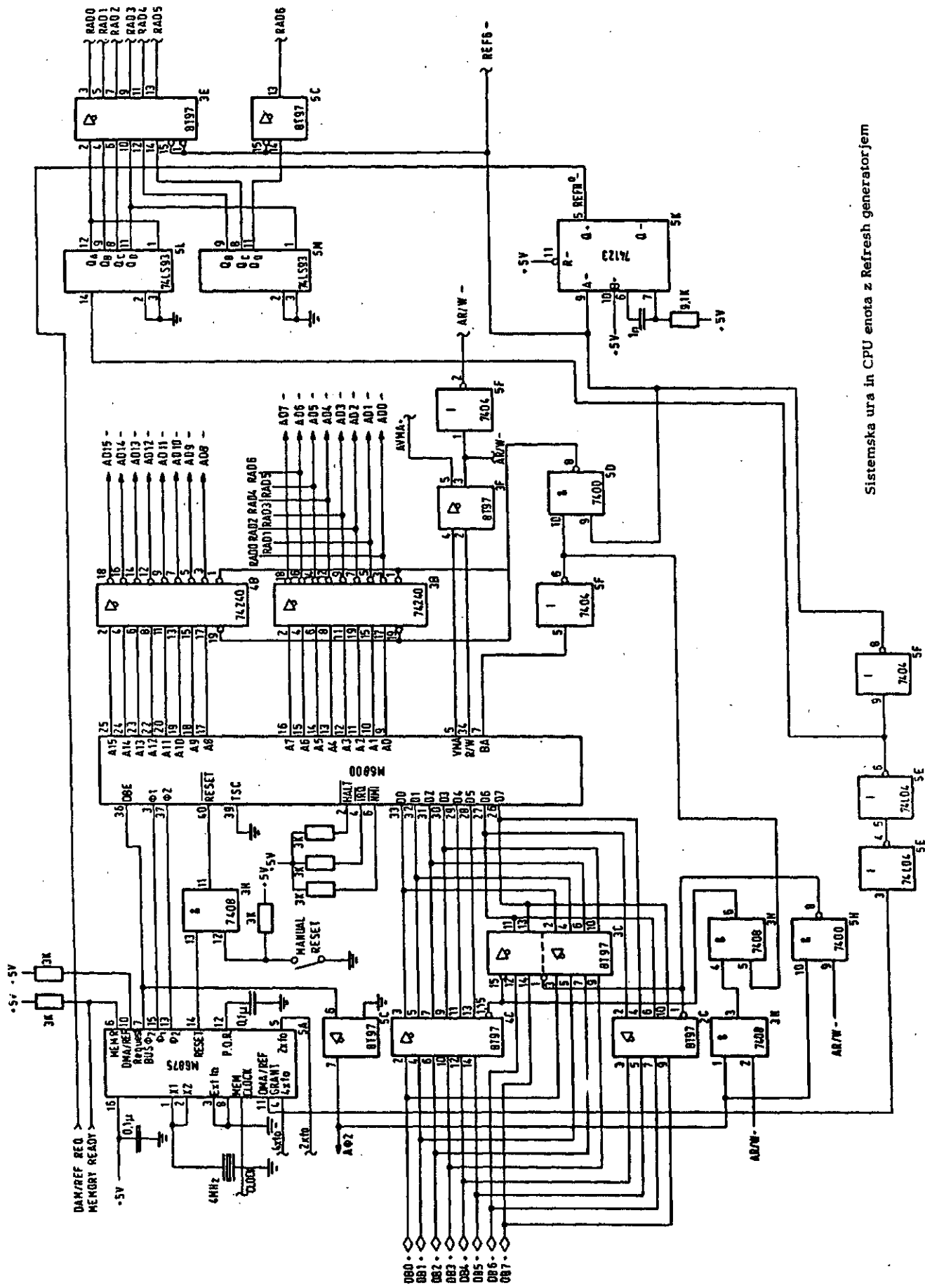


Slika 1. Razporeditev nožic za taktni generator XC 6875

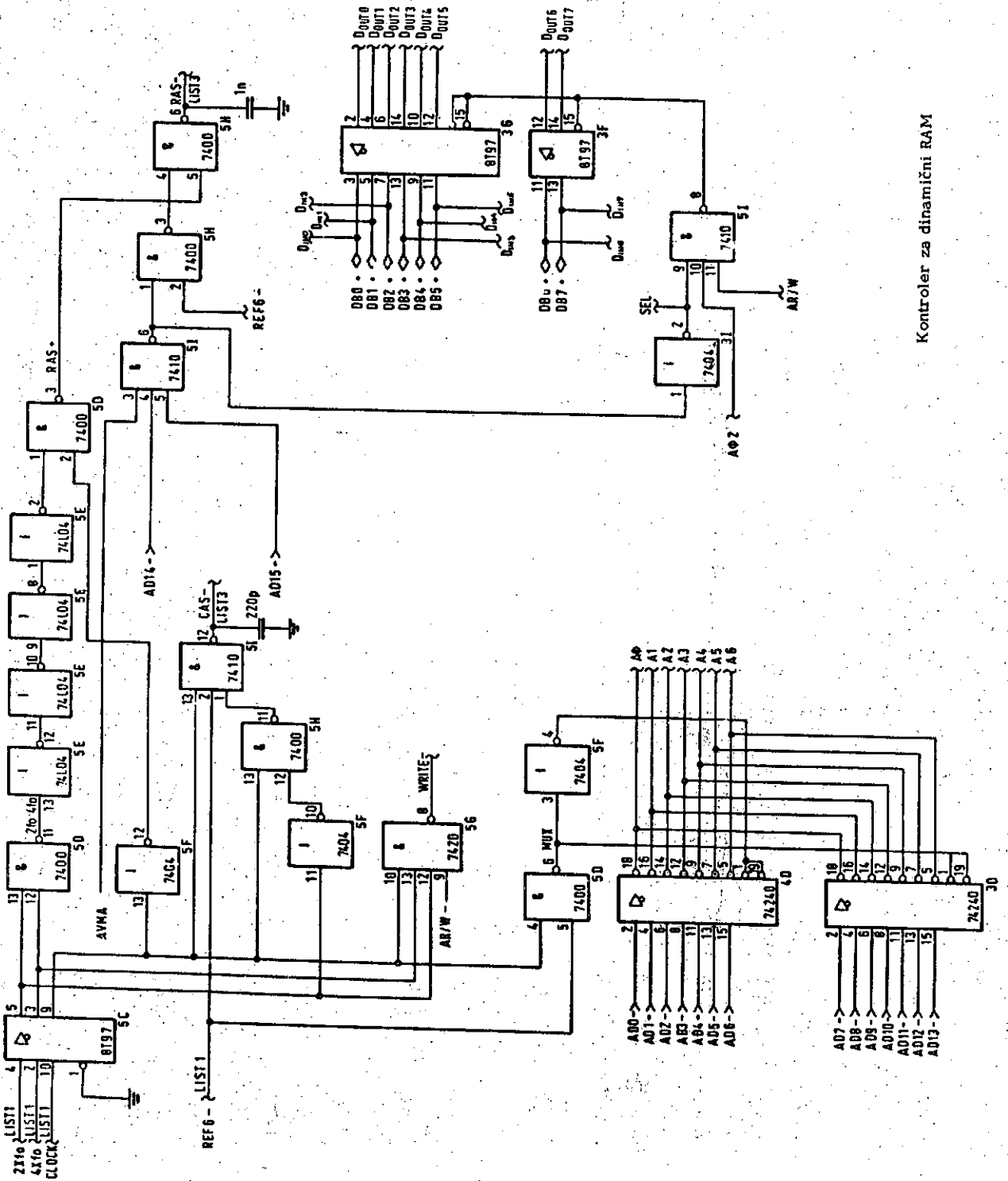
pa s signalom "Memory Ready". Prvi primer podaljšanja uporabljamo pri osveževanju dinamičnih pomnilnikov. Torej pri tem računalniku ne moremo uporabljati DMA tehnike prenosa podatkov. S priključitvijo kondenzatorja med sponki "Power-On-Reset-" in maso je mogoče enostavno realizirati avtomatsko resetiranje ob vklopu.

3. KRMILNO VEZJE ZA DINAMIČNI POMNILNIK

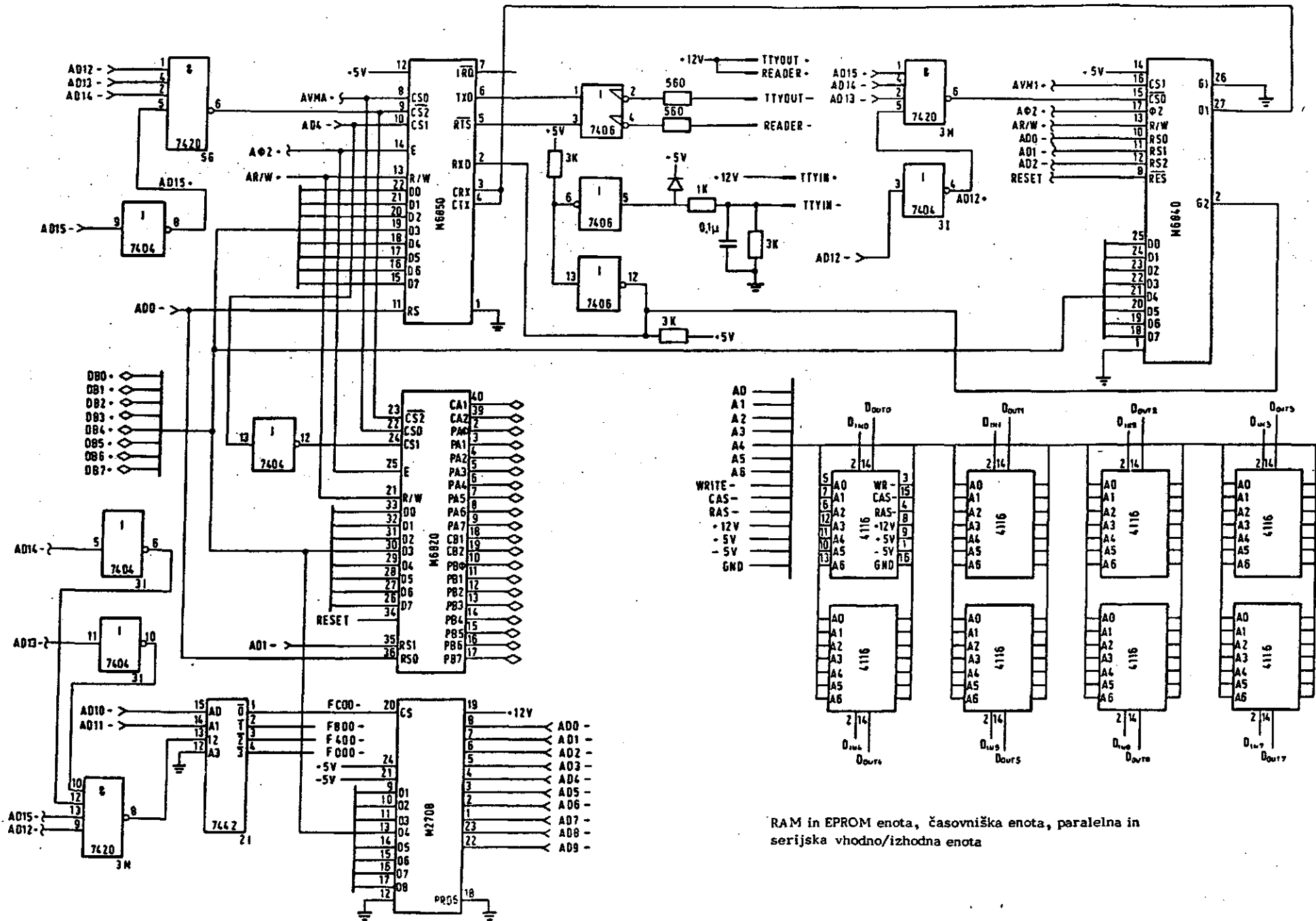
Delovni pomnilnik računalnika predstavlja osem pomnilniških integriranih vezij, ki vsebujejo po 16.384 pomnilniških celic. Slika 2 prikazuje razporeditev nožic in zahtevane časovne diagrame za čitanje, pisanje in za osveževanje pomnilnika. Krmilno vezje mora uskladiti časovne poteke signalov mikro procesorja 6800 z zahtevanimi poteki signalov dinamičnega pomnilnika. Glede na časovne poteke signalov pri pisanju in čitanju (slika 3) je najugodnejši "Read-Modify-Write" režim in osveževanje samo z RAS- (Row Address Strobe) signalom. Oblikovanje krmilnikov za dinamične pomnilnike je zelo izčrpno in podrobno opisano v članku "Razvoj dinamičnih pomnilnikov za mikro računalnike" (Informatica št. 4, letnik 1977). Kot že rečeno nam omogočajo višje derivacije osnovnega takta relativno enostavno generiranje vseh potrebnih signalov (RAS-, CAS-, WRITE-, MUX). Krmilno vezje sestavlja šest običajnih integriranih vezij. Na sliki 4 so prikazani časovni poteki omenjenih signalov glede na sistemski takt.



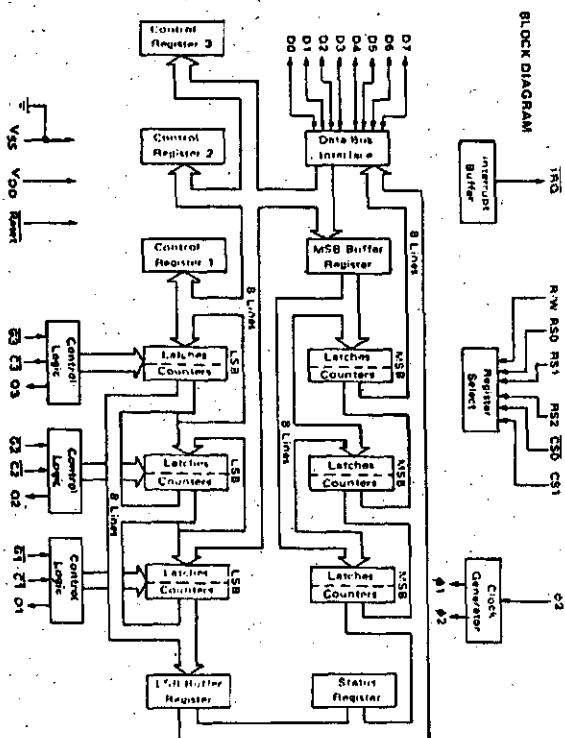
Sistemska ura in CPU enota z Refresh generatorjem



Kontroler za dinamični RAM



RAM in EPROM enota, časovniška enota, paralelna in serijska vhodno/izhodna enota

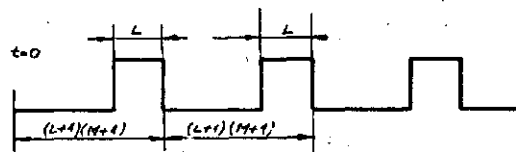


Slika 5. Bločna shema integriranega vezja MC 6840

1. normalen 16-bitni način



2. dvojni 8-bitni način



N = 16-bitna vrednost v časovniku
 L = 8-bitna vrednost v spodnji polovici registra
 M = 8-bitna vrednost v zgornji polovici registra

Slika 6.

c) Tretji način delovanja obsega merjenje periode in dolžin posameznih faz periodičnega signala na vodu G. V tem primeru sproži ustrezna fronta glede na tip merjenja odštevanje začetne vrednosti števnika v časovniku. Odštevanje se konča zopet s fronto. Pri merjenju frekvence nastopita dva dogodka pri enakih frontah, pri merjenju dolžine posameznih faz pa pri različnih frontah (prvi in zadnji).

6800 ASM V01

PAGE 1

```

006 *1000
CONTR2 EQU *1006 WRITE CONTROL REG. 2 & READ STATUS REG.
CONTR1 EQU *1007 WRITE CONTROL REG. 1
TIML02 EQU *1002 READ LSB BUFFER REG. 2
TIMH12 EQU *1003 READ TIMER #2 COUNTER
TIML01 EQU *1004 READ LSB BUFFER REG. #1
TIMH11 EQU *1005 READ TIMER #1 COUNTER
*
1000 06 10 LDA #10
1002 07 90 06 STAA CONTR2 GATE #2 DURATION MEASUREMENT MODE
1005 4F CLRA
1006 07 90 07 STAA CONTR1 ALL TIMERS ALLOWED TO OPERATE
1009 06 90 06 LOOP LDA CONTR2 WAIT UNTIL MEASUREMENT IS
100C 04 02 AND #02 FINISHED
100E 27 F9 BEQ LOOP
1010 06 90 03 LDA TIMH12 MSB OF INTERVAL
1013 F6 90 02 LDAB TIML02 LSB OF INTERVAL
1016 43 COMA
1017 43 COMA
1018 44 LSRA COMPUTE THE DURATION
1019 56 RORB OF AN HALF PERIOD
101A 44 LSRA
101B 56 RORB
101C 44 LSRA
101D 56 RORB
101E 44 LSRA
101F 56 RORB
1020 44 LSRA
1021 56 RORB
1022 07 90 05 STAA TIMH11 SET THE HALF PERIOD
1025 07 90 04 STAA TIML01
1029 06 02 LDA #02
102A 07 90 07 STAA CONTR1 CONTINUOUS OF OPERATION
END
    
```

Slika 7.

Dva časovnika smo v našem računalniku uporabili za merjenje hitrosti vhodnega signala in za nastavljanje oz. generiranje takta za serijski vhodno-izhodni kanal. Takoj po vklopu računalnika moramo poslati (pritisnemo na ustrezno tipko na teleprinterju) v računalnik en serijski ASCII znak, ki ima sodi kod. Vhodni signal je pripeljan tudi na vhod drugega časovnika. Z njim izmerimo dolžino startnega impulza. Nato izračunamo ustrezno frekvenco (upoštevati moramo, da dela ACIA s 16-krat višjo frekvenco) in s prvim časovnikom, nato generiramo periodični signal ustrezne frekvence, ki služi kot takt za regularnost delovanja vhodno-izhodnega kanala (ACIA). S to metodo smo odpravili zahtevo po preklapljanju hitrosti, saj se računalnik avtomatično prilagodi hitrosti periferne naprave. Program za urejanje dolžine štartnega impulza in za generiranje takta je podan na sliki 7.

5. SKLEP

V članku je opisana zgradba mikro računalnika "VITA" zelo na kratko, ker smatramo, da so detajlni stikalni načrti dovolj zgovorni za strokovnjaka s tega področja. Nekoliko izčrpnije so opisani manj običajni detajli in modernejša integrirana vezja. Opisani primer računalnika naj bi opogumil oz. vzpodbudil razvijalce digitalnih sistemov h konstruiranju dovolj močnih in kompatibilnih mikro računalnikov z uporabo najmodernejših tehnoloških dosežkov in k uvajanju manj konvencionalnih vendar učinkovitih detajlov.

6. LITERATURA

- (1) A.P. Železnikar, M. Kovačević, D. Novak, Razvoj dinamičnih pomnilnikov za mikro računalnike, Informatica 1 (1977), šte. 4, str. 9-21.
- (2) M. Kovačević, D. Novak, A.P. Železnikar, Monitorji za mikro sisteme sa procesorom 6800, Informatica 1 (1977), šte. 2, str. 20-25.

prilog kon metodite na vrednuvanje na apl sistemite

d.davčev

UDK 681.3.06

Elektrotehnički fakultet
Skopje

Trudot dava prilog kon metodite na vrednuvanje na APL sistemite. Poradi složenosta na eden APL sistem i različnite verzii na APL ,metodite na vrednuvanje i merenja stanuvaaat mnogu važni kako za korisnikot ,taka i za proektantot na sistemot. Celta na ovoj trud e da dade ednostavni ,no značajni istražovanja vo vrska so funkcioniranjeto na APL sistemite.

CONTRIBUTION TO THE METHODS OF EVALUATION FOR APL SYSTEMS - This paper gives the contribution to the methods of evaluation for APL systems. Because of the complexity of APL ,methods of evaluation and measurement are becoming very important to the user and the designer of the system. The purpose of this paper is to give simple but meaningful investigations into the activity of APL systems.

I. UVOD

APL e interpretativen ,konverzacioni jazik od visoko nivo. Pričinite za radjanjeto na APL bea poveće pati dadeni od K.Iverson i A.Falkoff vo poveće publikaciji. Prvata namena beše potrebata od formalen sistem sposoben da gi opišuva lesno algoritmito vo takvi oblasti kako što se linearno programiranje ,numerička analiza itn. Togaš ,beše uvideno deka takov sistem e isto taka mnogu efikasen i koncizan programski jazik. Vo mnogu zemji vo svetot APL brzo stanuva najvažen vremenski-podelen sistem.

Vo APL nema deklariranje ni na tip ni na forma. Promenlivite se od tipot celobrojni ,karakterni ,realni ili bulovi i vo forma na skalar ,vektor ,matrica ili so opšta struktura. (G.l.). Sepak izvršuvanjeto na primitivnite funkcii traži ispitivanje na tipot i na formata od nivnite argumenti. Sekoja primitivna funkcija poseduva simbol i može da bide edno- ili dvoargumentna , vo funkcija od brojot na napišane argumenti ,što znači da e toa određeno preku sintaksata. Verifikacijata na eden APL izraz za vreme na izvršuvanjeto se sostoi od verifikacijata na kompatibilnosta i domenot (rang ,dolžina ,indeks ,domen) i verifikacija na inicijalizacijata (vrednost).

Vo APL postojat dva načina na rabota :
-vo izvršivanje -sekoja instrukcija vnese na terminalot e interpretirana vednaš i prosledena so odgovor.

-vo definicija -dozvoluva definiranje na funkcii koi možat potoa da bidat povikani. Kako e APL sistem so bogati i širiki vozmožnosti ,od edna strana i kako denes vo svetot postojat poveće verzii na APL ,od druga strana očigledno e da se pojavuva potreba od razvijanje na metodologija za ocenuvanje na eden ovakov sistem.

Dosega se napraveni neкои obidi za voopštuvanje na APL sistemot i negovo vrednuvanje (A.S.,B.l.) i dr. Celta na ovoj trud e da prikaže edna šema kako doprinos kon metodite na vrednuvanje na ovie sistemi. Kako taa

ne zavisi mnogu od specifičnosti na APL ,istata će može da se koristi za vrednuvanje i na drugi interaktivni jazici. Šemata se sostoi od četiri dela : Utvrđivanje karakteristiki na jazikot ,utvrđivanje na okolinata , generiranje i eksploatacija i efikasnost.

II. UTVRĐIVANJE KARAKTERISTIKI NA JAZIKOT

Imajći vo vid deka se ušte ne postoi celosen formalen opis na APL ,kako i faktot deka postojat poveće verzii na APL ,da bi se utvrdile karakteristike na ovoj sistem potrebno e da se definira eden referenten APL. Toj referenten sistem može da bide APLSV (A.l.,A.2.,F.l.,F.2.) vo odnos na koj bi se utvrdile različni varijanti.

Eden lingvistički sistem može da se analizira so ogleđ na negovata sintaksa ,pragmatika i semantika. Sintaksata se bavi so odnosite medju elementite na jazikot. Semantikata se bavi so odnosite medju elementite na jazikot i nadvorenite elementi na koi tie se odnesuvaat. Pragmatikata gi opišuva odnosite na elementite na jazikot so korisnicite na istiot.

II.1. Sintaksa
Sintaksičnata analiza vo APL ,koja e krajno ednostavna , se sostoi vo nabrojuvanjeto na karakterite primeni od sistemot ,leksikata ,organizacijata na kategorii ,izborot na postapka za analiza ,kako i organizacijata na sintagmata posle analizata. Karakterite primeni od sistemot možat da se podelat na podgrupi ,što bi bilo korisno pri vrednuvanjeto. Leksikata e sostavena od avtonomni simboli kako što se 7,100,φ itn. i heteroindikator t.e. iminja ,kako što e na primer APLSV. Avtonomnite simboli možat da bidat ednokarakterni ,povećekarakterni fiksni ili povećekarakterni -promenlivi. Što se odnesuva do nadvoreniot aspekt na sintaksičnata organizacija ,posle neispravno formiran izraz sistemot dava poraka :SYNTAX ERROR .

II.2. Semantika

Naprotiv prostata sintaksa, semantikata na APL e mnogu bogata. Dodeka sintaksata gi tretira odnosite medju celinite na jazikot (na pr. odnosi funkcija-argument), semantikata go tretira nivnoto funkcioniranje. Ako se ima vo vid osnovnata šema : podatoci-algoritam-rezultati, na sekoja celina od jazikot može da i se dodeli edna karakteristika narečena "red". Na ovoj način će postojat celini od redot 0, 1 i 2. Celina od red 0 e celina koja može da bide samo argument ili rezultat na eden algoritam. Celina od redot 1 e celina koja može da ima uloga na algoritam ako ima za argument i rezultat celini od redot 0. Celina od redot 2 e celina koja može da ima uloga na algoritam ako ima kako argumenti celini od redot 1. Rezultatite možat da bida od redot 0 ili 1. Vtoreta karakteristika e slaboda, koja može da ima dve vrednosti 0 i 1 za konstanti i promenlivi soodvetno.

II.2.1. Celini od redot 0 -Vo ovaa grupa spadaat podatocite i dva atributa se tuka od važno značenje: tip i struktura. APL gi poseduva slednite tipovi : bulov (B), celobroen (C), realen (R) i karakteren (K). Vo slučaj na realen tip postojat dve grupi : fiksna i promenliva. Što se odnesuva do strukturata vo APL postojat skalari, vektori i matrici (tabeli). Strukturata vo APL e merliva veličina so pomoš na primitivnata funkcija ζ (A.1.). Ureduvanje na celinite može da bide tabelarno, razgranato ili od "opšta struktura" (G.1.). Posledniot slučaj se odnesuva na strukturite kaj koi točkite se skalari, vektori ili matrici (tabeli). Specijalni funkcii se upotrebuvaat za manipulacija so ovie strukturi.

II.2.2. Celini od redot 1 -fiksni (primitivni funkcii) -Često vo APL se vrši slednata klasifikacija na primitivnite funkcii : valentnost i ednorodnost, po što se razlikuvaat skalarni i mešani funkcii. Valentnost na nekoja funkcija pretstavuva broj na argumenti na taa funkcija. Primitivnite funkcii vo APL imaat valentnost 1 (edno-argumentni funkcii) i 2 (dvo-argumentni funkcii). Definirante funkcii možat da imaat valentnost 0. Funkcija f so valentnost pogolema od 2 se pišuva konvencionalno kako $f(a, b, c, d)$. Ova može da se interpretira kako edno-argumentna funkcija f primeneta na vektorski argument a, b, c, d i ovaa interpretacija se koristi vo APL. Vo APLSV argumentite a, b, c, d moraat da delat ista struktura. Implementacijata na opštite strukturi go odstranuva ova ograničuvanje. Namesto ednorodnost pokorisno bi bilo vo vrednuvanjeto na APL da se koristi eden atribut narečen složen tip koj će gi opati odnosite na tipot i strukturata, medju funkcijata i nejninite argumenti. Ako se koristi znakot U za unija na tipovi, znakot ; za proizvod na tipovi, togaš preku složeniot tip može da se izrazi tipot na argumentite prifateni od edna primitivna funkcija, kako i rezultatot od taa operacija.

Na ovoj način pokraj drugite specifičnosti potrebno e da se ispitaat site četiri vida na primitivni funkcii : skalarni funkcii so eden argument, skalarni funkcii so dva argumenta, mešani funkcii so eden i mešani funkcii so dva argumenta. Sekoj sistem poseduva različen broj na primitivni funkcii so određeni karakteristiki, pa potrebno e pri vrednuvanjeto posebno da se ispitaat ovie funkcii za dadeniot sistem. Zatoa se predlaga slednata notacija : P=BUC, Q=PUR, U=QUK kade se B-bulov, C-celobroen, R-realen i K-karakteren tip, dodeka znakot U pretstavuva unija na tipovi. Za opis na strukturata će se koristat simbolite : S-za skalar, V-za unija na S i vektori i T-za unija na V i tabeli. Na pr. primitivnata funkcija so dva argumenta ζ (vidi A.1.) ima složen tip (U;U)•B.

Kaj mešanite primitivni funkcii ušte edna klasifikacija e od značaj, a taa e prema funkcijata koja tie ja obavuvaat : indikativni (pr. $\zeta, \epsilon, \nu, \Delta$) koi davaat nekoja informacija vo odnos na nekoj APL objekt, strukturalni koi ja modifitsiraat organizacijata na nekoj APL objekt (pr. $\phi, \lambda, \downarrow, \uparrow$) i kreativni funkcii koi se najblisku po efektot od skalarnite funkcii (pr. $t, ?$). Za sekoja od ovie funkcii potrebno e da se ispita pored složeniot tip i dimenziite i rangot na argumentite i rezultatot. Posebna vrsta na mešani primitivni funkcii se dodeluvanjeto (\leftarrow), skokot (\rightarrow), izvršuvanje (Δ), format (∇), zagradi ($[]$) koi treba posebno da se ispitaat.

II.2.3. Celini od redot 1 -promenlivi (definirani funkcii) -Vo APL postojat dva načina na rabota : izvršuvanje i definicija. Vtoriot način doveduva do zgolemuvanje na sintaksata i semantikata na jazikot. Taka se pojavuvaat termini kako program, vodač, etiketa i linija. Semantički imame nova struktura na podatoci : program koj pretstavuva vektor čii elementi se liniite na programot (G.1.). Razmenata na informaciji e važen element od funkcioniranjeto na prostorot za rabota koj e daden na raspolaganje na eden korisnik. Ovaa razmena se obezbeduva preku tabelite koi ja sodržat listata na identifikatorite. Razlikuvame globalni promenlivi, poznati i koi važat za celiot prostor na rabota, lokalni promenlivi koi se definirani vo naslovot na funkcijata i ne se poznati za nea, i etiketi (labeli) koi se globalni vo odnos na numeričkata vrednost i lokalni vo odnos na skokot.

II.2.4. Celini od redot 2 (funkcioneli i operatori) -Vo APL postojat tri primitivni funkcioneli so eden argument : / (redukcija), \ ("scan") i nadvorešen proizvod. So dva argumenti e vnatrešniot proizvod.

II.2.5. Identiteti -Mnoštvo na primitivni funkcii ovozmožuva pojava na relaciji medju funkcii i funkcionalite koi se izrazuvaat preku identitetite. Ovie identiteti možat da bidat indikacija za izborot na korisnikot vo vrška so procena na kvalitetot. Postojat strukturalni, aritmetički, logičko-aritmetički, algoritamski i funkcionalni identiteti.

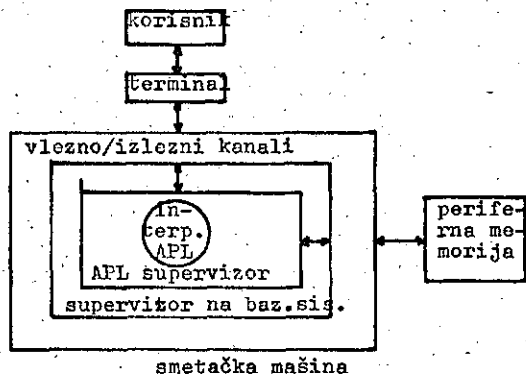
II.3. Pragmatika

Vo delot vo koj beše izvršena semantička analiza na APL beše spomnata i edna posebna vrsta na mešani primitivni funkcii ($\leftarrow, \rightarrow, \Delta, \nabla, []$). Ovie funkcii imaat ne "standardni" domani za razlika od drugite funkcii koi se odnesuvaat na domeni kako što se broevite, karakterite itn., i davaat rezultati koi pripadjaat na ovie isti domeni. Definiranjeto na program vo APL se vrši so pomoš na znakot ∇ i ako ovoj znak se posmatra kako eden funkcionel od posebna vrsta, togaš negovoto izvršuvanje pretstavuva stvaranje na objekti koi ne se direktno izvršeni, tuku samo rezervirani za idni izvršuvanja.

Primitivnata funkcija RC (return) koja go započnuva izvršuvanjeto na operacite može da se smatra kako dvo-argumentna funkcija čii argumenti se linijata od instrukcii koja treba da se izvrši i interpretatorot. Rezultatot može da pripadja na najrazlični domeni.

Znači so postoenjeto na funkcionalni domeni koi se ne "matematički" se manifestira na prvo mesto postoenjeto na edna nova dimenzija na lingvističkata analiza - pragmatikata. Pragmatičkata analiza ne e možna bez celosen opis na konkretni sistem. Ako se posmatra opštata šema od sl.1. se zabeležuva eden određen broj na hardverski delovi (konzola, periferni uredi itn.) i softverski delovi (supervizor, interpretator itn.). Tuka e i korisnikot na sistemot. Nekoi od ovie hardverski i softverski delovi imaat vremenski

karakter (RC, poraki, izvrševanje na nekoja akcija), a nekoji specijalen (rabotniot prostor WS, tabelata na simboli, listata na etiketi itn.). Potrebno e znači segda da se izvrši studija na jazikot vo ova poopšto svetlo, gledan kako del na ovoj kompleksen sistem.



Sl.1.

Poveće od celinite so pragmatička priroda imaat specijalen status vo APL. Tie se razlikuvaat po sintaksa, po leksika ili po ufrlvanje na dodatni karakteri. Po nivniot red postojat pragmatički promenlivi, funkcii i funkcionali.

II.3.1. Pragmatički promenlivi - celini imenuvani so eden rezerviran identifikator (prvot znak kaj ovie celini e karakterot □ -vidi A.2.) Toa se promenlivi koi se odnesuvaat na okolinata na korisnikot i na aktivniot WS. Koga korisnikot dava nekoja vrednost na ovie promenlivi toj predizvikuva modifikacija na edna posebna zona vo svojoj WS. Nekoi od ovie promenlivi (onie koi korisnikot gi deli so sistemot) se reafektirani avtomatski od sistemot pred sekoja nova upotreba.

II.3.2. Pragmatički funkcii - poinaku tie se vikaat i sistemski funkcii vo APLSV. Za ovie funkcii će stane zbor vo delot III.

II.3.3. Pragmatički funkcionali - toa se sistemskite komandi vo APL koi počnuvaat so znakot). Vo ova grupa se nekoi od funkciiite vrzani za tabelata na simbolite, funkcii za prenos na informacii medju WS i bibliotekata itn.

II.3.4. Posebni pragmatički objekti - toa se primitivnite funkcii →, ←, ∇, znakot ∇ za način na definicija, funkcijata I, A.

III. UTVRDUVANJE NA OKOLINATA

APL sistemot može da se posmatra kako del od edna smetačka mašina i nejinata okolina. Za negova realizacija potrebni se: bazičen jazik za redakcija na sistemot (najčesto assembler-ski), programi koi go sočinuvaat sistemot, komunikacii medju APL i bazičniot sistem, upravuvanje so sistemot vo podeleno vreme, ekonomsko upravuvanje so sistemot, uslovi za generiranje i održuvanje. Da bi se utvrdila sredinata vo koja funkcionira APL potrebno e da se odredi softverskata, hardverskata i administrativnata sredina (gi definira uslovite na eksploatacija na sistemot).

III.1. Hardverska sredina

Taa treba da bide potpuno definirana i specificirana: identifikacija na CPU, na kanalice i edinicite za kontrola koi gi podnesuvaat perifernite uredi, perifernite uredi, konzolite. Vsušnost se raboti za ocenuvanje na minimalna konfiguracija za daden tip na instalacija koja će može da go podnese APL.

III.2. Softverska sredina

Se odnesuva na bazičniot sistem: identifi-

cija na sistemot, identifikacija na verzijata itn.

III.3. Softverska APL sredina

Se raboti za specificacija na interakciite medju bazičniot sistem i APL sistemot: vnesuvanje na APL vo bazičniot sistem, specificacija na sistemskite resursi okupirani od APL, metod na presmetka na resursite, interakcija so drugi konverzacioni sistemi, grafički programi APL za vizuelnite terminali. Vsušnost neophodnite interakcii medju APL i negovata okolina može da se realiziraat preku niz na sistemski promenlivi koi pretstavuvaat "interface" medju APL i bazičniot sistem. Pored sistemskite promenlivi vo APL postoi možnost da se koristat funkcii koi ne možat samite da bidad eksplicitno raspoloživi, pa se baziraat na upotrebata na sistemskite promenlivi. (A.2.) Ovie funkcii se narečeni vo APL sistemski funkcii. Tie možat da bidad upotrebeni i vo definiranite funkcii i možat da imaat eden ili dva argumenta. Vo pogolem broj slučai tie imaat impliciten rezultat so toa što nivnoto izvrševanje pričinuva promena vo okolinata. Sistemskite promenlivi se primeri na podeleni promenlivi i tie se podeleni medju WS i APL procesorot. Od druga strana dva inače nezavisni operativni procesori možat da komuniciraat medju sebe ako tie delat edna ili poveće promelivi. Ovaka podeleni promenlivi stvaraat "interface" medju procesorite. Posebno, promenlivate možat da bidad podeleni medju dva aktivni APL WS, ili izmedju eden APL WS i nekoj drug procesor koj e del od celokupniot APL sistem. Pri vrednuvanjeto na dadeniot sistem treba posebno vnimanje da se posveti vo analizata na ovie promenlivi.

IV. GENERIRANJE I EKSPLOATACIJA

IV.1. Generiranje

Da bi se izvršilo generiranje potrebno e da se raspolaga so magnetni traki so moduli, so pričačnici za instalacija itn. Neophoden e celosen opis na etapite na generiranje, kako i možnite sistemski poraki.

IV.2. Eksploatacija

Za korektna eksploatacija potrebni se opisi na: rabotata na operatorot APL, na korisnicite, na komunikaciiite medju bazičniot sistem i APL, na organizacijata na komunikacii na bazičnata smetačka mašina i na ostanatite vo mrežata na smetački mašini. Tuka se i sistemskite komandi koi mu ovozmožuvaat na korisnikot da ja kontrolira cpštata okolina. Tie gi kontroliraat WS, terminalot i bibliotekite i go ovozmožuvaat sistemot na poraki medju korisnicite i operatorot. Na krajot treba da se spomne i analizata koja treba da se sprovede vo vrska so održuvanje na sistemot. Sekoj APL sistem ima određen sistem na zaštita i sigurnost koj vo uslovi na vremenski podelena sredina ima mnogo važna uloga.

V. EFIKASNOST

Povećeto verzii na APL se razlikuvaat vglavnom po efikasnošta t.e. po upotrebata na glavnite resursi: vreme i prostor. Koga se raboti za multiprogramaska sredina performansata na eden APL sistem e pred se funkcija od: brzinata na CPU, operativnata memorija, upotrebata na diskovite, prioritetot, brojot na WS rezidentni vo operativnata memorija (što zavisi od instalacijata), brojot na aktivni terminali i specificnata rabota koja se vrši od korisnikot.

V.1. Organizacija na vremeto

Organizacijata na vremeto e dosta važna vo vrska so efikasnošta na eden APL sistem ako se ima vo vid da e možno da se organizira podelba na vremeto medju poveće korisnici što otvara nova možnost na podelba na promen-

livi medju korisnicite. Imajći vo vid da ovie možnisti se koristat na različn način vo različni sistemi potrebno e postapkata na vrednuvanje da ja dađe specifičnata organizacija na vremeto za dadeniot sistem.

V.2. Organizacija na prostorot

Organizacijata na prostorot e vrzana voglavnom za organizacijata na vremeto (na pr. sistem so edna ili poveče konzoli). Vo site slučaj APL sodrži specijalen koncept na WS. No ,ovoj WS implicira postoenje i na drug prostor ,sistemski prostor (vo ovaa analiza sistemski prostor APL). Ovoj sistemski APL prostor se koristi za skladiiranje na supervizorot ,interpretatorot i dadenite programi ,za delovite od sesiite vo podeleno vreme. WS može da bide aktiven i WS vo rezerva smesten vo bibliotekata na korisnikot ili vo javnite biblioteki. WS ja ima slednata struktura : tabela na simboli , podatoci pridruženi na ovie simboli , blok na izvršuvanje , informacii i parametri na WS ,sistemska zona kade se zapazuvaat registririte. Vo ovoj del od ocenata potrebno e da se definiraat za dadeniot sistem parametrite za ocena na memoriskite potrebi.

V.3. Efikasnost na sintaksično nivo

Ovaa efikasnost e povrzana so algoritmite za konstrukcija i analiza na tabelite na simboli i za sintaksičnata analiza.

V.4. Efikasnost na semantičko nivo

Ovde efikasnosta e povrzana so preciznosta na presmetkite koi se vršat ,što e funkcija od koristeniot aseblerski jazik ,no i od koristenite algoritmi.

V.5. Pragmatička efikasnost

Ovaa efikasnost zavisi od konkretnata organizacija na resursi na sistemot kako , što se : zoni na memorija ,sistemski rutini ,sintaksički i semantički rutini itn.

V.6. Merenje na brzinata na izvršuvanje na primitivnite funkcii

Potrebno e da se izvršat merenja i grafički da se pretstavat za brzinata na izvršuvanje vo funkcija od količinata na informacii so koi se manipulira. Rezultatite možat da bidat dosta različni i da variraat od sistem do sistem.

V.7. Merenje na brzinata na izvršuvanje na definiranite funkcii

Ovoj element e mnogu važen vo ocenuvanjeto : imajći go vo vid izvršuvanjeto na primitivnite funkcii ,metodite na organizacija na WS i na parot supervizor-interpretator ,interesno e da se presmeta brzinata na izvršuvanje na složenite funkcii koi realiziraat eden ist algoritam. Taka možat indirektno da se izmerat : istražuvanjata od tabelata na simbolite ,povikuvanjata na funkcii ,eksploatacijata na strukturite na podatocite ,eksploatacijata na tipot na podatoci itn.

V.8. Odreduvanje na statičkata i dinamičkata učestalost na primitivnite funkcii

Zemajći nekolku reprezentativni definirani funkcii može da se izvršat merenja so cel da se dobijat dinamičkata i statičkata učestalost na primitivnite APL funkcii. Odovde procentualno može da se izvede zaključok kolku često se koristat poedini primitivni funkcii pa na toj način da se izvrši određeno ureduvanje po važnost na primitivnite funkcii. Koga veće gi imame rezultatite od merenjata po točka V.6. možeme polesno da izvršime vistinska ocena na dadeniot APL sistem.

V.9. Odreduvanje na "Instruction Mix"

Imajći ja vo vid dinamičkata učestalost na primitivnite funkcii ,kako i merenjata od točka V.6. ,možno e da se najde "Instruction Mix" za određen APL sistem. Tuka se pojavuva problemot vo vrska so dimenziite i veličinata na argumentite na dadenata primitivna funkcija ,no so određeni ograničuvanja bi moželo da se dojde do približno točnata vrednost za "Instruction Mix" za dadeniot APL sistem.

V.10. Odreduvanje na dolžinata na programot , potrebnoto vreme za programiranje i nivoto na jazikot za daden APL sistem.

VI. ZAKLJUČOK

Trudot gi analizira i dava ocena na поваžnite osobini na eden APL sistem gledan na dvoen način : kako lingvistički sistem i kako sistem del od edna smetačka mašina i nejzinata okolina. Ocenata i analizata se raboteni za da poslužat pri konkretnoto vrednuvanje na APL sistemite.

VII. LITERATURA

- A.1. APL, Manuel de référence IBM ,GHF2-0056-0
- A.2. APL, Shared variable system ,APLSV user's manuel ,programming RPQ WE 1191 ,5799-AJF
- A.3. APL Congress -Copenhagen ,1973
- A.4. Abrams, P.S., An APL machine ,SLAC Report n° 114 ,1970 ,Stanford University
- B.1. Brown, J., A generalization of APL ,Ph.D. , Syracuse University ,1971
- B.2. Bard, Y., Performance criteria and measurement for a time-sharing system ,IBM Syst.J. ,n° 3 ,1971
- C.1. Colloque APL ,IRIA ,Paris ,9-10sept. ,1971
- F.1. Falkoff, A.D., Iverson, K.E., The design of APL ,IBM J. Res. Develop. ,July 1973
- F.2. Falkoff, A.D., Iverson, K.E., APL/360 User's manuel ,1968
- G.1. Glandour, Z., Mezci, I., General arrays ,operators and functions ,IBM Res. Develop. ,July 73
- H.1. Halstead, M.H., Elements of Software Science ,Purdue University ,Els. Pub. Com. ,1977
- S.1. Spirn, J.R., Program Behavior: Models and Measurements ,The Pennsylvania St. University ,Els. Pub. Com. ,1977

uporaba časovnikov in števnikov v mikroprocesorskih sistemih s procesorjem z-80

a.p. železnikar

UDK 681.3-181.4:621.374.32

Odsek za računalništvo in informatiko, Institut Jožef Stefan, Ljubljana

Članek opisuje uporabo, programiranje, vgraditev in funkcijo časovnika/števnika v mikroročunalniškem sistemu. V konkretnih primerih je pokazana dovolj široka uporaba časovnika/števnika Z-80-CTC iz procesorske družine Zilog Z-80. Najprej je opisana materialna zgradba vezja tipa CTC in programiranje vezja: nastavitve operacijskega načina, prekinitvenega vektorja in časovne konstante. Pokazana je vključitev vezja CTC v mikroročunalniško vezje, programiranje prekinitvev, vezanih na časovne konstante, prikazana sta pa tudi dva primera, in sicer programirana ura realnega časa in program za avtomatično nastavljanje hitrosti serijskega kanala.

Usage of Timers and Counters in Z-80 Microprocessing Systems. This article deals with use, programming, interfacing and functioning of counter/timer circuits (CTC) in microprocessing systems. Examples of Z-80 timer/counter application are shown. First, the hardware construction of CTC is described and programming of operation mode, interrupt vector and time constant is presented. A simple circuit for CTC interfacing with microprocessor system is shown. Two examples of CTC programming are described in detail: programmed real time clock and a program for automatic setting of baud rate in serial channel.

1. Uvod

Programiranje procesov v realnem času zahteva nekoliko več znanja o računalniškem sistemu in njegovih možnostih, kot ga navadno ima programer na velikem sistemu; ta pozna in gleda računalniški sistem predvsem globalno, abstraktno ali površinsko. Optimalne možnosti dane računalniške konfiguracije je moč doseči le s temeljitim poznavanjem njene materialne in programske strukture. Tako sodi reševanje določenih problemov z računalnikom v okvir sistemskega programiranja.

Mikroročunalniška uporaba nam večkrat ne omogoči strogega razločevanja med sistemskim in uporabniškim programiranjem. Računalnik, ki je povezan z določenim zunanjim procesom v sistem, mora imeti "sistemske" lastnosti, kot so npr. spremenljivi vhodni/izhodni kanali, časovniški in števniki mehanizmi, možnosti prekinjanja uporabniških programov, kompleksne povezave med kanali, vključevanje dodatnih procesorjev itn. Programer tako oblikovanega procesnega sistema mora obvladati programiranje časovno sovisnih podprocesov v celotnem procesu, torej mora dovolj podrobno poznati tudi mikroročunalniški sistem, s katerim proces krmili.

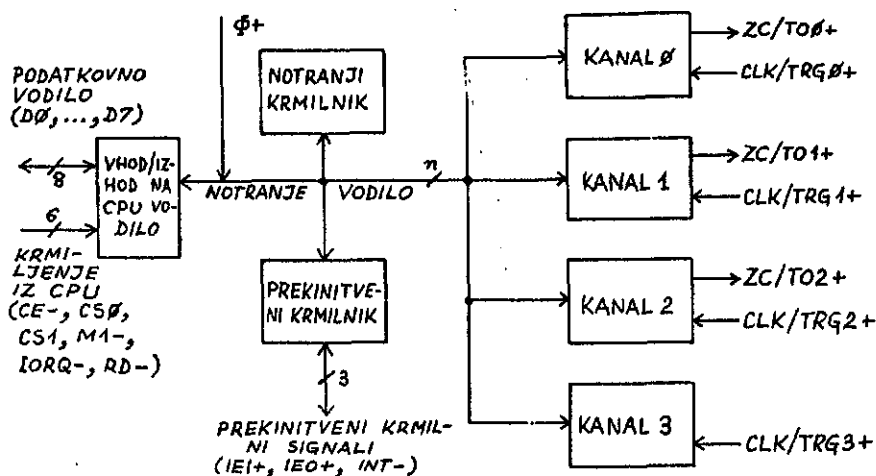
Programiranje procesov, ki zahtevajo referenco na relativni ali absolutni čas, je v mikroročunalniških sistemih omogočeno z uporabo posebnih integriranih vezij, ki jih imenujemo števniki/časovniki (npr. CTC iz družine Z-80 pomeni Counter/Timer Circuit). Mikroročunalniške družine 8080, 6800, F8 in Z-80 imajo taka vezja, kot so npr. intervalni časovnik 8253 (za 8080), programirljivi časovnik 6840 (za 6800), števniki/časovniki CTC ali 3882 (za Z-80) ter ROM/časovnik 3851 in pomnilniški vmesnik/časovnik 3853 (za F8).

V nadaljnjem bomo uporabljali okrajšavo CTC za števnikiško/časovniško vezje. Programirljivost vezij tipa CTC omogoča, da uporabimo to vezje za vrsto različnih funkcij, kot so npr. štetje, odštevalno štetje, generiranje impulzov in prekinjanje programov v različnih časovnih intervalih. Števniki in časovniki so navadno izvedeni v obliki paralelnih kanalov, ki imajo po 16 števnikiških bitov ter so nastavljeni za števrne operacije s poljubnimi moduli (npr. moduli od 1 do 256 za 8-bitni segment). Štetje se dostikrat izvaja s pomočjo sistemskega takta ali njegove derivacije. Pri tem imamo dve vrsti vhodnih podatkov (zlogov) za CTC: krmilni (nastavitveni, načinovni) in vrednostni (količinski) podatek. Krmilni podatek specializira dani kanal v CTC za določeno funkcijo, vrednostni podatek pa določi startno ali ponavljalno vrednost števnega segmenta, kanala ali pa prekinitveni vektor, tj. naslov, kjer se nahaja podatek za skok ob prekinitvi. Krmilni podatek osposobi tako dani kanal za prekinitve, ko se pojavi sistemski prekinitveni signal ali pa tudi signal kanalne ničle (odštevalnega segmenta).

Članek opisuje uporabo elementa CTC v mikroročunalniškem vezju ter pripadajoče programe, ki omogočajo sprejemanje in oddajanje signalov v serijskih V/I kanalih. Opisan je tudi program za generiranje realnega časa z uporabo enega kanala vezja CTC in prekinjevanja poljubnega uporabniškega programa.

2. Zgradba števnikiškega in časovniškega vezja tipa CTC

Oglejmo si arhitekturo števnika/časovnika z oznako Z-80-CTC (ali MK 3882), ki je integrirano vezje (ima 28 nožic). Osnovne lastnosti tega vezja so:



Slika 1. Globalna shema števnikova/časovnika, ki ima štiri kanale (npr. Z-80-CTC); kanal 3 nima izhoda za indikacijo ničle (zero count); vezje CTC se poveže na vodila mikro računalnika (npr. na Z-80-CPU) in ga je mogoče programsko krmiliti in odčitavati.

- vezje je programsko nastavljivo za določeno funkcijo;
- kanali vezja delujejo kot števnik ali časovnik;
- odštevalni števnik je čitljiv programsko in odšteva do ničle;
- časovniški kanal lahko uporablja predštevnik z izbiro faktorja 16 ali 256;
- časovniška operacija se lahko sproži z negativno ali pozitivno signalno fronto;
- trije kanali imajo števniški/časovniški izhod ničle (odštevanje do ničle) za zunanjo uporabo;
- vezje ima krmilnik za prekinitveno prednost posameznih kanalov.

Vezje Z-80-CTC je programirljiva, štirikanalna enota, ki opravlja funkciji štetja in časovnega merjenja v sistemih z mikroprocesorjem Z-80; ta enota je seveda uporabljiva tudi v vezjih z drugimi mikro procesorji.

Na sliki 1 je narisana blokovni diagram števnikova/časovnika, ki prikazuje bistveno funkcijo vezja. Enota se poveže na vodila sistema s procesorjem Z-80 ter ima svojo notranjo logiko, štiri števniške kanale ter prekinitveni mehanizem. Vsakemu kanalu je prirejen prekinitveni (naslovni) vektor za avtomatični skok ob prekinitvi, prekinitvena prednost pa je določena s kanalskim indeksom: kanal 0 ima najvišjo, kanal 3 pa najnižjo prednost.

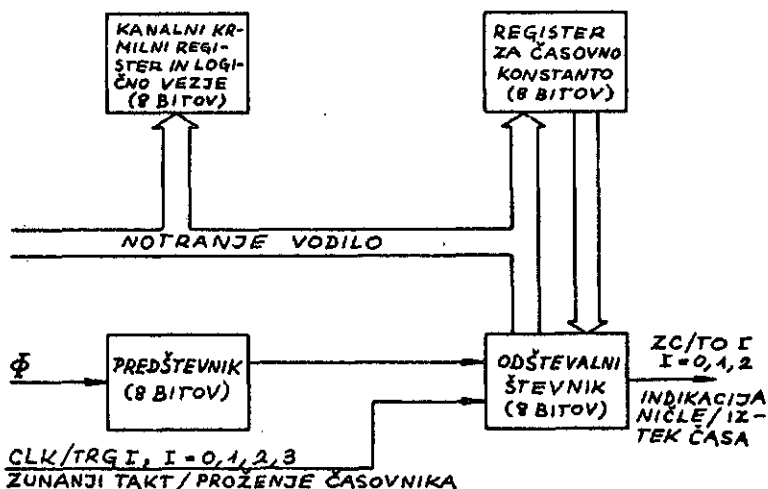
Kanalna logika je sestavljena iz dveh registrov, dveh števnikov ter iz kontrolne logike, kot kaže slika 2. Imamo 8-bitni register za nastavljanje časovnih konstant ter 8-bitni krmilni register za izbiranje funkcij (načinov) v posameznih kanalih. Kanalni števni mehanizem je sestavljen iz 8-bitnega predštevnik in 8-bitnega odštevalnega (navzdolnjega) števnik; slednji je programsko čitljiv. Predštevnik je nastavljen in deli vhodni takti signal s faktorjem 16 ali 256.

Vsebina iz 8-bitnega registra za časovno konstanto, ki smo jo vstavili v ta register preko vodila, se prenese v odštevalni števnik; ko je števnik dosegel ničlo, se vsebina iz registra za časovno konstanto lahko avtomatično ponovno prenese v števnik.

Programsko se preko vodila nastavi tudi vsebina 8-bitnega kanalnega krmilnega registra, s katero se določijo način in pogoji delovanja kanala.

Odštevalni števnik se korakoma dekrementira skozi predštevnik v časovniškem režimu in preko vhoda CLK/TRG v števniškem režimu. Ta števnik se lahko naloži s programskim navodilom ter vsakokrat, ko je dosegel svojo ničlo; števnik je čitljiv v svojem procesu odštevanja.

Slika 2. Bločna shema kanala v vezju CTC; predštevnik je nastavljen (pri Z-80-CTC samo na 16 ali 256), odštevalni števnik pa je nastavljen in čitljiv.



3. Programiranje števnika/časovnika

3.1. Nastavitev operacijskega načina

Funkcija števnika/časovnika je odvisna od vsebine, vstavljen: v kanalni krmilni register KKR (slika 2). Kadar želimo preko sistemskega vodila vstaviti zlog v KKR, mora biti zadnji bit zloga D0 enak 1, kot kaže slika 3. Z vrednostmi preostalih bitov D1, ..., D7 v KKR je natanko določena funkcija. Kanal, na katerega se nanaša podatek o njegovi funkciji v KKR, je določen z bitoma CS0 in CS1 (nožici 18 in 19 integrirane vezja Z-80-CTC). Izbira kanala se preko posebnega vezja doseže z ukazom izhoda in navedbo vrat (npr. ukaz OUP (DD); tu je DD heksadecimalni naslov vrat).

Slika 3 kaže, kako je mogoče določiti funkcijo posameznega kanala z biti D7, ..., D1 registra KKR. Imamo tale pomen:

- D7 = 0: Kanalna prekinitve je onemogočena.
 D7 = 1: Kanalna prekinitve je omogočena in nastopi vedno tedaj, ko odštevalni števnik doseže ničlo. S postavitvijo D7 v KKR na vrednost 1 se za prekinitve ne upošteva prejšnji prehod odštevalnega števnika skozi ničlo.

D7	D6	D5	D4	D3	D2	D1	D0
OMOGOČITEV PREKINITVE	ŠTEVNIK ALI ČASOVNIK	OMOGOČITEV PREDŠTEVNIKA	FRONTA SIGNALA CLK/TRG	PROŽENJE ČASOVNIKA	NALOŽITEV ČASOVNE KONST.	USTAVITEV ŠTEVILKA	1

UPORABA V ČASOVNIŠKEM REŽIMU

Slika 3. Nastavitev kanalnih funkcij z biti kanalnega krmilnega registra (KKR)

- D6 = 0: Imamo časovniški režim. Odštevalni števnik se dekrementira skozi predštevnik. Perioda odštevalnega števnika je določena z izrazom
 TAKT.PREDŠTEVNIK.ČASOVNA KONSTANTA
 kjer je TAKT perioda signala 0 v mikro sekundah, PREDŠTEVNIK je faktor z vrednostjo 16 ali 256 in ČASOVNA KONSTANTA je faktor v intervalu med 1 in 256.
 D6 = 1: Imamo števnikiški režim. Odštevalni števnik se dekrementira z zunanjim taktom CLK/TRG.
 D5 = 0: Sistemski takt 0 se deli v predštevniku s faktorjem 16, toda samo v časovniškem režimu.
 D5 = 1: Sistemski takt 0 se deli v predštevniku s faktorjem 256, toda samo v časovniškem režimu.
 D4 = 0: Padajoča signalna fronta vhodnega signala CLK/TRG začne časovniško operacijo v časovniškem režimu in dekrementira odštevalni števnik v števnikiškem režimu.
 D4 = 1: Naraščajoča signalna fronta vhodnega signala CLK/TRG povzroči delovanje števnika/časovnika, kot je bilo opisano za primer D4 = 0.
 D3 = 0: Če je D2 = 0, potem začne časovnik svoje delovanje v začetku naslednjega strojnega cikla. Če je D2 = 1, potem začne časovnik svoje delovanje v začetku tistega strojnega cikla, ki sledi ciklu, v katerem je bila časovna konstanta vstavljen v register časovne konstante. Poljubna vrednost D3 se upošteva le v časovniškem režimu.

D3 = 1: Če je D2 = 0, potem začne časovnik svoje delovanje po izteku trenutnega strojnega cikla, ko je nastopil specifični prehod (signalna fronta) signala CLK/TRG. Če je D2 = 1, začne časovnik svoje delovanje v strojnem ciklu po vstavitvi časovne konstante, ko je nastopil specifični prehod (signalna fronta) signala CLK/TRG. Poljubna vrednost D3 se upošteva le v časovniškem režimu.

- D2 = 0: Nobena časovna konstanta ne bo sledila kanalnemu kontrolnemu zlogu. Toda neka časovna konstanta mora biti oziroma je vpisana v kanal, da se štetje začne.
 D2 = 1: Časovna konstanta za odštevalni števnik bo naslednji zlog, ki bo vpisan v izbrani CTC kanal. Če se časovna konstanta vstavi med kanalnim odštevanjem, se trenutno odštevanje še konča in šele nato se upošteva (naloži) nova časovna konstanta v odštevalni števnik.
 D1 = 0: Štetje v kanalu se nadaljuje.
 D1 = 1: Štetje v kanalu se ustavi, dokler se ne naloži neka časovna konstanta. Izhod ZC/TO je medtem neaktiven. Kanal ne more sprožiti prekinitve.
 D0 = 1: Vse zapisano o bitih D7, ..., D1 velja le pri pogoju D0 = 1. Pri D0 = 0 se nalaga prekinitveni vektor.

3.2. Naložitev prekinitvenega vektorja

Štirim kanalom v vezju CTC lahko priredimo različne prekinitvene vektorje. Celotni prekinitveni naslov se oblikuje v CPU ter se pojavlja na naslovnem vodilu. Različnost vektorjev za posamezne kanale je odvisna od vrednosti, ki jo naložimo v kanal CHAN0, ko mora biti vrednost bita D0 enaka 0. Biti D7, D6, ..., D3 predstavljajo prekinitveni vektor skupaj s vsebino registra I v CPU, dočim bita D2 in D1 ne vplivata na vrednost vektorja. Ko vezje CTC reagira na prekinitveno zahtevo, se v bitih D2 in D1 pojavi binarni kod (številka) kanala z najvišjo prednostjo, torej D2.D1 = 00, 01, 10, 11 (za štiri kanale). Kanal CHAN0 ima najvišjo prednost.

V naših primerih bo CHAN0 zaseden za serijski vhodni/izhodni kanal, tako da bomo npr. za generiranje realnega časa uporabljali preostale kanale CHAN1, CHAN2 in CHAN3. Naslovi, na katerih se nahajajo naslovi za skok ob prekinitvi, so določeni takole:

stik vsebina (I); vsebina (D7, ..., D3 naloženo v CHAN0 pri D0 = 0); vsebina (D2, D1 v CHAN0, ko imamo za kanale 0, 1, 2, 3 vrednosti D2.D1 = 00, 01, 10, 11); 0

Tako dobimo vselej 16-bitni vektor ob prekinitvi, kjer je D15, D14, ..., D8 = vsebina (I), D7, D6, ..., D3 = vsebina (vstavljen v CHAN0), D2, D1 = številka kanala in D0 = 0. V tabeli 1 imamo kombinacije različnih možnosti za izbiro lokacij, na katerih se lahko nahajajo naslovi za subrutinske prekinitvene pozive. Iz tabele je razvidno, da je mogoče s štirimi kanali v vezju CTC enakomerno zasesti vse sode lokacije pomnilnika; na te lokacije shranjujemo 16-bitne naslove, s katerimi so določeni subrutinski skoki ob prekinitvah.

Primer: Naj bo (I) = 1FH ter naj bo v kanal CHAN0 naloženo 58 (naložimo lahko seveda tudi 59, 5A, ..., 5F, vendar se te vrednosti upoštevajo kot vrednost 58). Ob prekinitvi v tretjem kanalu CHAN3 se naslov za subrutinski skok nahaja v pomnilni celici z naslovom

$$1F00 + 58 + 6 = 1F5E$$

kjer je 6 indeks tretjega kanala (indeksi za kanale 0, 1, 2, 3 so 0, 2, 4, 6).

(I)	naloženo v CHANØ	lok. prekin. CHANØ	naslova za: CHAN1	CHAN2	CHAN3
00	00	0000	0002	0004	0006
00	08	0008	000A	000C	000E
00	10	0010	0012	0014	0016
00	18	0018	001A	001C	001E
..
00	F0	00F0	00F2	00F4	00F6
00	F8	00F8	00FA	00FC	00FE
01	00	0100	0102	0104	0106
01	08	0108	010A	010C	010E
01	10	0110	0112	0114	0116
01	18	0118	011A	011C	011E
..
01	F0	01F0	01F2	01F4	01F6
01	F8	01F8	01FA	01FC	01FE
..
FF	00	FF00	FF02	FF04	FF06
FF	08	FF08	FF0A	FF0C	FF0E
FF	10	FF10	FF12	FF14	FF16
FF	18	FF18	FF1A	FF1C	FF1E
..
FF	F0	FFF0	FFF2	FFF4	FFF6
FF	F8	FFF8	FFFA	FFFC	FFFE

Tabela 1

3.3. Naložitev časovne konstante

Osembitno časovno konstanto $\check{C}K$ naložimo v register za časovno konstanto (slika 2); ta naložitev mora časovno slediti naložitvi kanalne kontrolne besede, v kateri je bil bit D2 postavljen na vrednost 1. Kadar imamo

$$\check{C}K = \emptyset\emptyset H$$

je njena odštevalna vrednost največja, tj. 256; za ostale vrednosti pa velja razmerje:

$$\emptyset1H :: 1, \emptyset2H :: 2, \dots, 1\emptyset H :: 16, \dots \\ \dots, FFH :: 255$$

4. Števniki/časovniki v mikroročunalniškem vezju (sistemu)

Števniki/časovniki, ki ga bomo uporabljali, pripada določeni mikroprocesorski družini, kar pomeni, da je neposredno povezljiv z vodili zadavnega mikroprocesorskega sistema (naslovno, podatkovno, kontrolno in prekinitveno vodilo). Nekaj takih posebnih integriranih vezij smo navedli že v uvodu; na tem mestu si bomo ogledali

vstavitve (vgraditve) števnika/časovnika (ali Z-80-CTC ali MK 3882) v mikroročunalniško vezje s procesorjem Z-80-CPU.

Na sliki 4 vidimo porazdelitev signalov na podnožju integriranega vezja CTC. Na sponke CLK/TRG j (j = 0,1,2,3) lahko pripeljemo zunanje taktne ali časovniške impulze za posamezne kanale j. Sponke ZC/TO j (j = 0,1,2) imajo signal 1 v trenutku, ko je kanal j dosegel ničlo (odštel do ničle). Z bitoma CS1 in CSØ izbiramo kanal (CS1.CSØ = 00,01,10,11) in D7,...,DØ so signali podatkovnega vodila. Nadalje pomeni: CE- signal aktiviranja integriranega vezja (osposobitev), Ø+ je sistemski takt; MI-, IORQ- in RD- so signali kontrolnega vodila, imamo pa tudi tri signale za krmiljenje prekinitve IEI+, IEO+ in INT-. S signalom RESET- se ustavi štetje v vseh kanalih ter se anulirajo osposobitveni biti v vseh kontrolnih registrih.

Oglejmo si, kako vgradimo vezje CTC v konkretni mikroročunalniški sistem, ki uporablja procesor Z-80-CPU. Za nastavljanje funkcij v kanalih vezja CTC uporabimo ukaz

OUT (VRATA),A

kjer je A vsebina akumulatorja in VRATA ime (naslov) vrat, ki se pojavljajo v dani konfiguraciji sistema. Operand VRATA je osembitna beseda, ki se pojavi na naslovnem vodilu, in sicer na vodih AØ,A1,...,A7. Hkrati se pojavi na naslovnem vodilu, in sicer na vodih A8,A9,...,A15 še vsebina akumulatorja A. Zlog (A) se nato pojavi tudi na podatkovnem vodilu in se vpiše v izbrani periferni kanal (z imenom VRATA). V tem primeru lahko ima konfiguracija 256 različnih vrat oziroma perifernih kanalov.

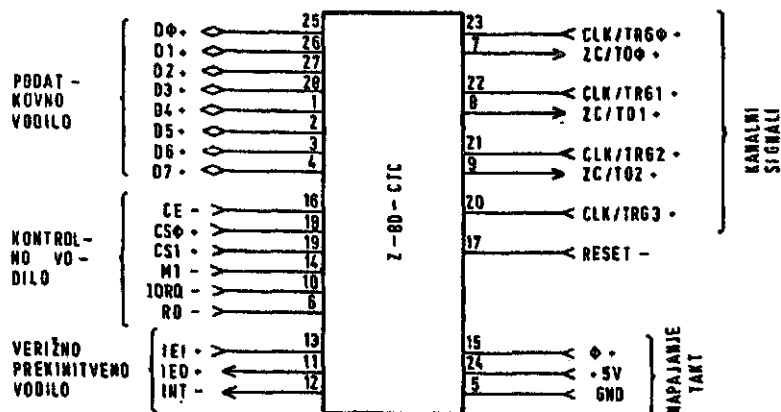
Vzemimo, da so naslovi vrat (spremenljivka VRATA) za kanale 0,1,2,3 v vezju CTC določeni z vrednostmi konstant CHANØ, CHAN1, CHAN2 in CHAN3, ko imamo

CHANØ EQU DBH;
CHAN1 EQU D9H;
CHAN2 EQU DAH;
CHAN3 EQU DBH;

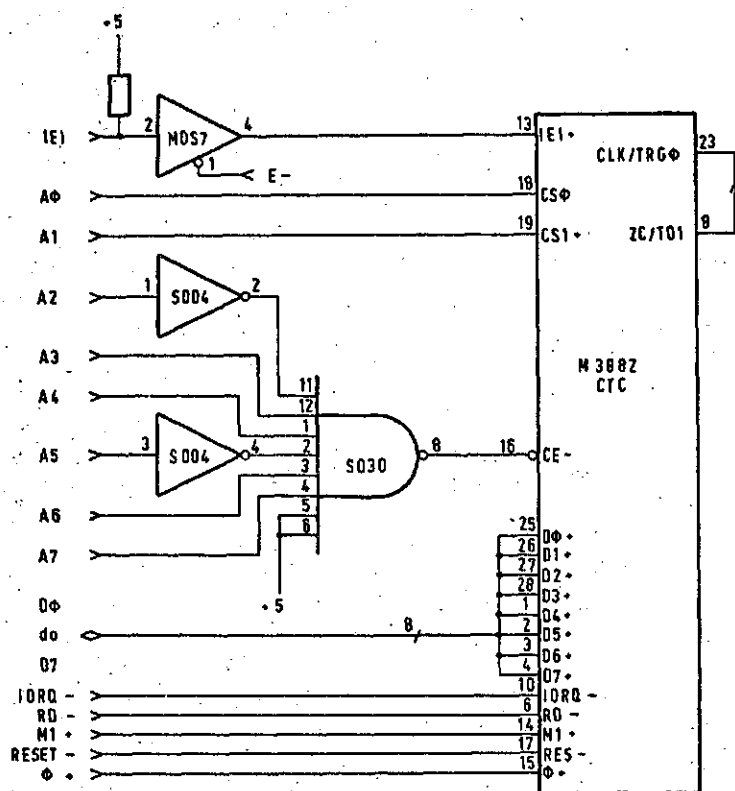
Tako so D8, D9, DA, DB konkretni naslovi vrat v vezju CTC. Ko se npr. z ukazom

OUT (ØDBH),A ;

pojavi D8 na linijah A7,A6,A5,A4,A3,A2,A1,AØ kot binarni zlog 11011000, se mora zgoditi tole: CSØ = AØ = Ø, CS1 = A1 = Ø; s tem je izbran kanal številka Ø (CHANØ) v vezju CTC; s preostalimi šestimi bitmi A7,A6,...,A2 moramo dobiti signal CE- (ko je A7.A6.A5.A4.A3.A2 = 110110). Tako dobimo primer za priključitev vezja CTC na vodilo, kot kaže slika 5. Če povežemo točko ZC/TO 1 s točko CLK/TRG Ø, bomo pri indikaciji ničle v kanalu 1 odštel enico v odštevalnem šte-



Slika 4. Shema podnožja števnika/časovnika Z-80-CTC



Slika 5. Priključitev integriranega vezja CTC na sistemsko vodilo, ko imamo naslove kanalov D8, D9, DA in DB (heksadecimalno)

vniku kanala \emptyset .

Kratko povzamemo: Izbira kanalov v vezju CTC, ko vstavljamo preko podatkovnega vodila različne podatke v CTC in s tem funkcionalno določujemo številniško/časovniško delovanje vezja CTC, je odvisna le od signalov CE-, CS Φ + in CS1+!

5. Programiranje časovnih konstant in prekinitev mikro računalnika

Programiranje serijskih vhodnih/izhodnih kanalov je dostikrat povezano z ustrezno nastavitvijo časovnika, ki ga uporabimo za generiranje (točneje delitev) takta vhodnoizhodnega elementa (npr. takt za UART, USART, ACIA itn.) ali pa za periodično prekinjanje delovanja (izvajanja) uporabniškega programa, ko se ob prekinjavih aktivirajo določene servisne subrutine za obdelavo vhodov in izhodov. V prvem primeru preoblikujemo npr. sistemski takt, ki ima frekvenco 2,4 MHz, v takt s frekvenco 800 Hz (tj. 16-krat višja frekvenco, kot je potrebna za hitrost 50 Baud). V tem primeru uporabimo vezja CTC kot časovnik brez kanalne prekinjitve in pobiramo dobljeni takt s sponke ZC/T01 (glej pod poglavje 3.1). Predhodno moramo v časovnik vstaviti še delilno konstanto

$$2400/0,8 = 3000$$

ko nastavimo predštevnik v CTC na vrednost 16 in odštevalni števec (tj. register za časovno konstanto) na vrednost 187 (lahko izberemo tudi par: predštevnik = 256, CK = 12, ki je pa manj natančen). V drugem primeru pa uporabimo vezje CTC kot časovnik s kanalno prekinjitvijo, ko se nam npr. vsakih 20 ms aktivira subrutina za obdelavo bita serijskega signala (50 Baud). V tem primeru nastavimo s predštevnikom in registrom za CK delilni faktor $2400/0,05 = 48000$ (izberemo par pš = 256 in CK = 187) (glej pod poglavje 3.1 in 3.2).

Prvi primer realiziramo s programskim segmentom

```
LD A,5          ;push control word into channel control register number
; OUT (0DBH),A ; zero when interrupt is disabled and prescaler will
;              ; divide by 16
LD A,187       ;push time constant into channel control register number
OUT (0DBH),A
```

drugi primer pa s segmentom

```
LD A,0A5H      ;push control word into channel control register number
; OUT (0DBH),A ; 0 when interrupt is enabled and prescaler will divide
;              ; by 16
LD A,187       ;push time constant into channel time constant register number
OUT (0DBH),A
```

Z drugim segmentom je povezano prekinjevanje uporabniškega programa, ko se ob vsaki prekinjavi aktivira t.i. servisna subrutina za obdelavo prekinjitve. Zato moramo k temu segmentu dodati še nadaljevalni segment, ki določa prekinjitveni vektor, torej:

```
LD A,LOWINT    ;load lower part of interrupt vector into CTC
OUT (0DBH),A
LD A,UPINT     ;load upper part of interrupt vector into register I
LD I,A
```

Ob prekinjavi bo CPU pobral vsebine iz naslovov takole:

prekinitev v kanalu števil.	naslov	vsebina
0	UP.LOW	(UP.LOW+1).(UP.LOW)
1	UP.LOW+2	(UP.LOW+3).(UP.LOW+2)
2	UP.LOW+4	(UP.LOW+5).(UP.LOW+4)

Vsebine teh naslovov so naslovi za subrutinske (prekinitvene) skoke. Npr.: če je UP = FF in LDW = 12 ter imamo (PPI2) = 00 in (PPI3) = 01, do za kanal 0 ob prekinitvi poklicana subrutina ki začneja na lokaciji 0100.

6. Primer programirane ure realnega časa

Uro realnega časa lahko realiziramo na več načinov; s posebno materialno opremo, ki daje na določenih sponkah podatke o času; z mesečno dodatno materialno in delno programsko opremo, ko dobimo v določenih pomnilnih lokacijah podatke o času; naposled imamo uro realnega časa, ki jo na obstoječi računalniški konfiguraciji realiziramo izključno s programom (brez dodatne materialne opreme). To možnost si bomo ogledali v tem poglavju.

Za programirano uro potrebujemo določene osnovne lastnosti računalniške konfiguracije, in sicer: programirljiv časovnik, ki povzroči sistemsko prekinitve, z mehanizmom subrutinskega (prekinitvenega) skoka, tako da se aktivira servisna subrutina; uporabniški program (ki je na določen način omejen) mora pri tem nemoteno teči dalje. Osnovniška prekinitve, povezana z delovanjem ure, lahko ima najnižjo prednost, važno pa je, da se vsakokratni zahtevani servis opravi prej ali slej.

Pogoji, pri katerih deluje programirana u-
ra nemoteno, ko uporabljamo med njenim delova-
njem različne direktive operacijskega sistema
(npr. za SDB-80), so file:

(1) Nastavitve I := 00H in IF := FFH na u-
streznih lokacijah sistema SDB-80;

(2) program, ki se izvaja, ne sme spremiti vsebine (I) = 00H in ne sme biti uporabljen ukaz DI (Disable Interrupt), ki bi onemogočil prekinitve pri nižji tretjega kanala;

(3) tipka "reset" ne sme biti uporabljena, ker se s tem anulirajo tudi vrednosti registrov v elementu CFC.

Oglejmo si najprej listo za prekinitveno servisno subrutino ure realnega časa na sliki 6. Začelnost te liste je, da je kod programa premestljiv (relativni naslov) in povezljiv preko globalne spremenljivke TIAL, v kateri se nahajajo vsakokratni podatki o času (ure dneva in minute). Povezovalni naslovnik lahko takšen heksadecimalni modul preneša in povezuje, upoštevajoč zasedenost pomnilnika.

Rutino na sliki 6 lahko pojasnimo s temle psevdo kodom:

SUBROUTINE REAL_TIME_CLOCK

deklariraj globalno spremenljivko TIAL in začetni vektor;
onemogoči prekinitve;
shrani v sklad vrednosti tistih registrov, ki bodo v subrutini uporabljeni (A, F, H, L);
vstavi v register HL naslov (kazalec) notranje dvozlčne ure TIA (225-tinke desetinke minute);
za eno enoto (225-tinko desetinke minute);
IP (v spodnjem delu TIA je nastičih 225 pre-
kinitvenih pozivov subrutine, kar us-
treza eni desetinki minute)

THEN

korigiraj celotno štirizlčno uro z ve-
likostnimi rezredi: 225-tinke desetinke

Y11 REAL TIME CLOCK
ADDR OBJECT ST #

```

0001 ; REAL TIME CLOCK INTERRUPT SERVICE
0002 ; SUBROUTINE
0003 ORG 0000H ;ORIGIN VECTOR
0004 GLOBAL TIAI ;VAR TIAI IS GLOB
0005 DEFW 3+2 ;BEGIN VECTOR
0006 ; SUBROUTINE STARTING POINT:
0007 DI ;DISABLE INTERRUPT
0008 PUSH AF ;SAVE AF
0009 PUSH HL ;SAVE HL
0010 LD HL,TIA ;SET HL TO TIME I-
0011 LD A,(HL) ; TEM AREA, INCRE-
0012 INC A ; MENT ITEM1
0013 CP 225D ;TEST ITEM1 ON 225
0014 JK 2,110-S ;IF 225 GO TO 110
0015 LD (HL),A ;ELSE STORE INCR A
0016 EXIT: ;
0017 POP HL ;RELOAD HL
0018 POP AF ;RELOAD AF
0019 EI ;ENABLE INTERRUPT
0020 RETI ;RETURN FROM INT
0021 XOR A ;CLEAR ACC
0022 LD (HL),A ;CLEAR TIME ITEM1
0023 INC HL ;TAKE TIME ITEM2
0024 LD A,(HL) ; AND INCREMENT
0025 INC A ; IT
0026 CP 10D ;TEST ITEM2 FOR 10
0027 JR NZ,EXIT-S ;IF NZ, GO TO EXIT
0028 XOR A ;CLEAR ACC
0029 LD (HL),A ;CLEAR ITEM2
0030 INC HL ;TAKE MINUTE ITEM
0031 LD A,(HL) ; LOAD IT TO ACC
0032 INC A ; AND INCREMENT
0033 DAA ;ADJUST ACC
0034 CP 60H ;TEST ITEM FOR 60
0035 JR NZ,EXIT-S ;IF NZ GO TO EXIT
0036 XOR A ;ELSE CLEAR ACC
0037 LD (HL),A ;CLEAR MINUTE ITEM
0038 INC HL ;TAKE HOUR ITEM
0039 LD A,(HL) ; LOAD IT TO ACC
0040 INC A ; AND INCREMENT
0041 DAA ;ADJUST ACC
0042 CP 24H ;TEST ITEM FOR 24
0043 JR NZ,EXIT-S ;IF NZ GO TO EXIT
0044 XOR A ;ELSE CLEAR ACC
0045 LD (HL),A ;CLEAR HOUR ITEM
0046 JR EXIT-S ;GO TO EXIT
0047 TIA DEFV 0000H ;TIME ITEMS 1 & 2
0048 SET DEFV 0000H ;MIN & HR ITEM
0049 TIAI DEFV 0000H ;MIN & HR GLOBAL
END

```

** PROGRAMMED BY A.P.ZELEZNIKAR, FEBRUARY, 1978, FOR MOSTEK
** SDB-80

Slika 6. Subrutina Y11, ki opravi servis ob prekinitvi v kana-
lu 3. Ura se požene s programom Y12 (slika 7).

minute, desetinke minute, minute in ure;

ELSE

ENDIF

obnovi vsebine uporabljenih registrov iz sklada (A,F,H,L);
omogoči prekinitev;
vrni se k izvajanju prekinjenega uporabniškega segmenta (to se opravi avtomatično)

ENDSUBROUTINE

Uro realnega časa, ki je realizirana s programom na sliki 6, aktiviramo takole:

Najprej naložimo (premestimo) subrutino tako, da ima svoj začetek na lokaciji 0006; to pomeni, da dobimo v celicah 0006 in 0007 skladno z listo na sliki 6 vsebini 08 in 00, ki predstavljaata naslov za skok 0008 (vsebinsko čitamo v obratnem vrstnem redu).

Ko smo subrutino naložili, jo aktiviramo takole:

(1) v kanal 0 elementa CTC naložimo spodnji del prekinitvenega vektorja, tj. 00F;

(2) v register I naložimo vrednost 00H; tako dobimo pri prekinitvi v kanalu 3 podatek za skok na naslovu 0006 (glej tabelo 1);

(3) vstavimo kontrolno besedo (zlog) v kanal 3 elementa CTC (npr. zlog 0A5H);

(4) v kanal 3 vstavimo časovno konstanto; v tem trenutku se začne odštevanje, torej urateče; subrutina REAL TIME CLOCK bo odslej poklicana vsakokrat, ko bo odštevalni števec v kanalu 3 dosegel ničlo; v tem trenutku bo nastopila prekinitev;

(5) predpostavljamo, da je računalnik v režimu MODE 2 (ukaz IM 2) in da je prekinitev omogočena (ukaz EI).

Pri izpolnjenih pogojih (1) do (5) lahko uro odčitavamo z direktivo:

```
M 3D ; naslov 3D = 37(gl.listo) + 6
003D 00 ; 00 je vrednost za minute
003E 11 ; 11 je vrednost za ure
```

Kadar nas zanima čas, preprosto odčitamo vsebinski lokaciji 3D in 3E, na katerih lahko trenutni čas tudi modificiramo.

Seveda pa si lahko zgradimo tudi program, ko z direktivo T (time) dobimo celotno sporočilo o času, npr.

```
TIME IS 1025 1025 1025 GMT ...
```

Lista tega programa je prikazana na sliki 7 (premestljivi modul za naložitev). S programom poženemo uro in začetni čas ji lahko nastavimo v celicah 3D in 3E; časovno direktivo T uporabimo za izpis časovnega sporočila kadarkoli. Globalna spremenljivka TIAL vsebuje vedno točen čas, ker program preprečuje prenos časa iz TIA v TIAL tedaj, ko bi se lahko pojavil aritmetični prenos v spremenljivki TIA (npr. prehod iz stanja 0159 v stanje 0200). Iz komentarja k programu na sliki 7 je razvidna semantika programa.

Prekinitveno subrutino in program naložimo tesno eden poleg drugega in program aktiviramo z direktivo E 41. Naložitev obeh modulov opravimo tako, da naložimo subrutino z direktivo L 6 in nato program z L, ko imamo:

```
.L 6
BEG ADDR 0006
EXECUTE 0006
END ADDR 0040
UNDEF SYM 00
*L
BEG ADDR 0041
END ADDR 00AB
UNDEF SYM 00
*
```

Sporočila v različnih trenutkih (s tastaturo vtiskavamo T) so npr.:

```
TIME IS 1923 1923 1923 1923 1923 GMT ...
TIME IS 1924 1924 1924 1924 1924 GMT ...
TIME IS 1927 1927 1927 1927 1927 GMT ...
```

7. Avtomatična nastavitvev časovnika na hitrost (Baud) serijskega kanala

Vrsta mikro računalnikov vsebuje pretikalo za nastavljanje različnih hitrosti serijskih kanalov, npr. za 45,45, 50, 56, 75, 100, 110, 150, 300, 600, 1200, 2400, 4800, 9600 itn. Baud. Temu stikalu se lahko izognemo tako, da časovnik izmeri hitrost kanala in da se nato vstavi ustrezna časovna konstanta v register časovne konstante, ki osposobi kanal za sprejem in oddajo pri določeni hitrosti.

Imamo dve možnosti: izmerjena hitrost lahko ima kar absolutno (ali višjo derivacijo absolutne) vrednost hitrosti kanala; v tem primeru so lahko hitrosti kanala poljubne (zvezne) v danem intervalu (npr. poljubna vrednost med 110 in 9600 Baud, npr. vrednost 7777 Baud). V drugem primeru se konstanta zaokroži na neko vrednost iz dane množice dopustnih (standardnih) vrednosti. Program za drugo različico bi izgledal takole:

```
LD A,5 ;load appropriate control word
; to enable channel 0 for down
; counting when the time con-
; OUT (0DSH),A ; start will be loaded
LD C,0DEH ;prepare the observation of
LD DE,01 ; start bit at port number DE
; and initialise DE-counter
L1:
IN A,(C) ;wait for start bit and proceed
JP P,L1 ; to L2 at its appearance
L2:
INC DE ;wait for the end of start bit
IN A,(C) ; and proceed to L3 when the
JP M,L2 ; end appeared
L3:
LD SP,BTAB-1 ;take the baud rate table and
; compare value of DE-counter
L4:
INC SP ; with appropriate table value
POP HL ;
SCF ;
SBC HL,DE ;
JR C,L3-3 ;
;
DEC SP ;load time constant from baud
POP AF ; rate table into channel num-
OUT (0DSH) ; ber 0
```

Pripadajoča tabela je pa tale:

vsebine		
0000 BTAB:	DEFW 0DH	; 9600 baud
01	DEFB 01H	; 9600 baud
1A00	DEFW 1AH	; 4800 baud
02	DEFB 02H	; 4800 baud
3400	DEFW 34H	; 2400 baud
04	DEFB 04H	; 2400 baud
6800	DEFW 68H	; 1200 baud
08	DEFB 08H	; 1200 baud
D000	DEFW D0H	; 600 baud
10	DEFB 10H	; 600 baud
A001	DEFW A0H	; 300 baud
20	DEFB 20H	; 300 baud
FF7F	DEFW 7FFFH	; 110 baud
57	DEFB 57H	; 110 baud

Iz programa je razvidno, da merimo vselej dolžino startnega impulza (bita). Ta meritev je npr. omogočena tako, da najprej resetiramo računalnik, nato pa pritisnemo na tastaturi na poljuben znak, ki ima najnižji bit enak 0 (to so znaki A,C,E,....,0,2,4,....).

Y12 REAL TIME CLOCK (Y11) ACTIVATIO

```

ADDR OBJECT ST #
0002 NAME Y12
0003 ; REAL TIME CLOCK ACTIVATING AND
0004 ; READING PROGRAM
0005 ; PROGRAMMED BY A.P.ZELEZNIKAR
0006 ; FOR MOSTEK SDB-80 (Z 80)
0007 ORG 0000H ;PROGRAM ORIGIN
0008 GLOBAL TIA1 ;VAR TIA1 IS GLOB
>0008 0009 CHAN0 EQU 0D8H ;CHAN0 PORT ADDR
0010 ;*****
>000B 0011 CHAN3 EQU 0DEH ;CHAN3 PORT ADDR
>00A5 0012 CNTLW EQU 0A5H ;CTC CONTROL WORD
>0000 0013 RATE1 EQU 0 ;CTC COUNTER RATE
>0000 0014 LOWINT EQU 0 ;LOW INT VECTOR
>0000 0015 UPINT EQU 0 ;UPP INT VECTOR
>0005 0016 REPEAT EQU 5 ;TIME REPEAT RATE
0017 ;*****
>E522 0018 RDCHR EQU 0E522H ;READ CHARACTER
>E3C7 0019 PTXT EQU 0E3C7H ;PRINT TEXT MESS
>E604 0020 PADD0 EQU 0E604H ;PRINT (HL)
>E11D 0021 MONITR EQU 0E11DH ;MONITOR REENTRY
0022 ; PROGRAM STARTING POINT:
0023 CTCINI:IM 2 ;SET INT MODE 2
0024 LD A,LOWINT ;SET LOW INT VECT
0025 OUT (CHAN0),A ; IN CTC CHANNEL 0
0026 LD A,UPINT ;SET UPP INT VECTOR
0027 LD I,A ; IN RREGISTER 1
0028 LD A,CNTLW ;SET CONTROL WORD
0029 OUT (CHAN3),A ; IN CTC CHANNEL 3
0030 LD A,RATE1 ;SET CTC COUNT RA-
0031 OUT (CHAN3),A ; TE IN CHANNEL 3
0032 EI ;ENABLE INTERRUPT
0033 READ: LD E,I ;INITIALIZE
0034 BACK: CALL RDCHR ;READ CHARACTER
0035 CP 'T' ;TEST FOR CHAR 'T'
0036 JR NZ,EXIT1-S ;IF NZ,GO TO EXIT1
0037 LD E,I ; ELSE TYPE MESS
0038 LD HL,TIMEM1 ;PRINT 1-ST PART
0039 CALL PTAT ; OF MESSAGE
0040 LD IX,TIA1 ;PREPARE TRANSFER
0041 WAIT: IN A,(CHAN3) ;PROTECTION AGAIN-
0042 CP 0C0H ; ST TIME CARRY IN
0043 JR NC,WAIT-S ; TIME ITEM AREA
0044 LD A,(IX-2) ;TRANSFER MINUTE
0045 LD (IX*0),A ; ITEM
0046 LD A,(IX-1) ;TRANSFER HOUR
0047 LD (IX*1),A ; ITEM
0048 LD HL,(TIA1) ;SET HR & MIN LOC
0049 LD B,REPEAT ;SET TIME REPEAT
0050 GMT: CALL PADD0 ;PRINT TIME
0051 DJNZ GMT-S ;REPEAT PRINT
0052 LD HL,TIMEM2 ;PRINT REST OF
0053 CALL PTXT ; THE MESSAGE
0054 JR BACK-S ;WAIT FOR NEW 'T'

```

Y12 REAL TIME CLOCK (Y11) ACTIVATIO

```

ADDR OBJECT ST #
004C FE0D 0056 EXIT1: CP 0DH ;TEST FOR CR
004E 2018 0057 JR NZ,EXTEND-S ;IF NZ,GO TO BACK
0050 C31DE1 0058 JP MONITR ; ELSE GO TO MONR
0053 0D0A 0059 TIMEM1:DEFW 0A0DH ;MESSAGE BEGIN
0055 54494D45 0060 DEFW 'TIME IS' ;"TIME IS "
005E 03 0061 DEFB 03 ;ETX CHARACTER
005F 20474D54 0062 TIMEM2:DEFW ' GMT ...' ;" GMT ..."
0067 03 0063 DEFB 03 ;ETX CHARACTER
0068 C31500 0064 EXTEND JP BACK ;SYSTEM EXTENSION
0065 END

```

Y12 REAL TIME CLOCK (Y11) ACTIVATIO

```

ADDR OBJECT ST #
BACK 0015 CHAN0 00D8 CHAN3 00DB CNTLW 00A5
CTCINI 0000 EXIT1 004C EXTEND 0068 GMT 003F
LOWINT 0000 MONITR E11D PADD0 E604 PTXT E3C7
RATE1 0000 RDCHR E522 READ 0013 REPEAT 0005
TIA1 (EXT) 003B TIMEM1 0053 TIMEM2 005F UPINT 0000
WAIT 0028
ERRORS=0000
ERRORS=0000

```

Slika 7. Program Y12 (na tej listi) aktivira uro realnega časa, čita in tiska realni čas v obliki posebnega sporočila z uporabo direktive "T" in omogoča lastno razširitev na dodatne direktive (mnemonično izbrani znaki) preko razširitve (dodatka) segmenta EXTEND (glej konec lista). Najprej se s programom Y12 določi prekinitveni vektor v kanalu 3 vezja CTC, usposobi kanal 3 za določeno (ustrezno) funkcijo (kontrolni zlog) in vstavi časovna konstanta v kanal 3. Takoj po vstavitvi časovne konstante se začne odštevanje v kanalu 3. Ko bo v kanalu 3 odštevanje doseglo ničlo, se bo skladno z vsebino lokacije, ki jo določa prekinitveni vektor, sprožila subrutina Y11 (glej prejšnjo listo) za obdelavo prekinitve. Program Y12 čaka medtem na znak T iz tastature in izpiše časovno sporočilo, ko se je T pojavil. Z znakom "CR" (vrnitev glave na teleprinterju) skočimo iz tega programa, vendar se Y11 (servisna subrutina) še nadalje aktivira ob vsaki prekinitvi v kanalu 3. Ker je začetek programa v našem primeru na lokaciji 0041, lahko ponovno vstopimo v program Y12 tudi z direktivo E 53 (reentry point) namesto z E 41. Program Y12 je predviden za razširitev na dodatne uporabniške direktive, s katerimi bi se izpisovala še različna druga, časovno odvisna sporočila oziroma realizirale druge specifične uporabniške akcije (imamo odprt uporabniški sistem). Na koncu liste je izpisana še simbolna tabela, v kateri je TIA1 zunanja spremenljivka, katere lokacija se določi s kompilacijo rutine Y11.

```

-M 6.AB.
0006 08 00 F3 F5 E5 21 3B 00 7E 3C FE E1 2B 06 77 E1
0016 F1 FB ED 4D AF 77 23 7E 3C FE 0A 20 F1 AF 77 23
0026 7E 3C 27 FE 60 20 E7 AF 77 23 7E 3C 27 FE 24 20
0036 DD AF 77 18 D9 00 00 00 00 00 ED 5E 3E 00 D3
0046 D8 3E 00 ED 47 3E A5 D3 DB 3E 00 D3 DB FB 1E 01
0056 CD 22 E5 FE 54 20 30 1E 01 21 94 00 CD C7 E3 DD
0066 21 3F 00 DB DB FE C0 30 FA DD 7E FE DD 77 00 DD
0076 7E FF DD 77 01 2A 3F 00 06 05 CD 04 E6 10 FB 21
0086 A0 90 CD C7 E3 18 C9 FE 0D 20 18 C3 1D E1 0D 0A
0096 54 49 4D 45 20 49 53 20 20 03 20 47 4D 54 20 2E
00A6 2E 2E 83 C3 56 00

```

8. Sklep

Opisani primeri kažejo, da je uporaba števnikov/časovnikov v mikroročunalniških vezjih lahko zelo pestra in je dostikrat odvisna od iznajdljivosti in znanja uporabnika. Kadar vsebuje računalniška konfiguracija več integriranih vezij tipa CTC, je mogoče programirati medsebojno časovno odvisne akcije. Časovnik/števniki omogoča tako vsestransko programiranje postopkov, ki potekajo v realnem času.

Na sliki 8 je izpisan računalniški kod za absolutno naložitev (med naslovoma 6 in AB).

Slika 8. Računalniški kod za listi iz slike 6 in slike 7, ki je bil naložen z nalegalnikom za povezovanje in premeščanje

Mikroprocesorske družine razpolagajo z različnimi CTC element, ki imajo različne funkcionalne lastnosti (npr. štetje navzgor in navzdol, primerjava s poljubno vrednostjo in ne samo z ničlo, poljubna nastavitve predštevnik itn.). Inženirska praksa potrebuje konfiguracije s časovniki pri večini aplikacij z mikro računalniki.

Literatura

- (1) A.P.Železnikar, I. Ozimek, M.Kovačević, D. Novak, Programiranje mikro računalnikov s procesorjem Z 80, Informatika 1(1977), št. 2, 5-12.

kaj je mikroročunalništvo

Mikro računalništvo je pojav, ki se začneja s tehnološkim razvojem visoke integracije polprevodniških sestavnih delov in z elementom, imenovanim mikro procesor (kratko μP). μP je tako centralni računalniški element, izdelan v obliki integriranega vezja. Mikro računalništvo se nato nadaljuje z uporabo sistemov z mikro procesorji, tj. z mikro sistemi (kratko μS), z njihovim programiranjem, proizvodnjo, raziskovanjem, metodologijo in z ustreznim izobraževanjem.

Mikroročunalniška, polprevodniška tehnologija se ne omejuje zgolj na μP : pojavljajo se spremljevalni elementi, tj. integrirana vezja, kot so: statični in dinamični pomnilniki (RAM, ROM, PROM, EPROM) do 16 tisoč bitov v enem vezju, vhodno/izhodni povezovalni elementi (A-CIA, UART, USRT, USART, PIA, PIO), programirani časovniki in intervalni (časovni) števniki (CTC), vezja za računalniški takt (ure z več fazami in z možnostmi krmiljenja faz), sistemi ojačevalnih vmesnikov s tremi stanji (stanja 0, 1 in visoka izhodna impedanca), naslovni pomnilni vmesniki za povezavo μP s statičnimi in dinamičnimi pomnilniki, prekinjevalna vezja, vezja za pogon periferije (pogon diska, kasetnega sistema, video prikazovalnika), A/D in D/A pretvorniki, multipleksorji, kodirniki in vezja s srednjo stopnjo integracije (tipa TTL, H, LS, L, MOS, CMOS itn.).

Uporaba in izgradnja μS zahteva obvladovanje naslednjih osnovnih veščin:

1. Poznavanje osnovnega μP materiala (različnih vrst osnovnih integriranih vezij, mikroprocesorskih družin ter kompleksnih povezovalnih in komunikacijskih integriranih vezij) ter sposobnost materialnega konfiguriranja, tj. načrtovanja in realizacije v okviru konkretnega mikroročunalniškega sistema.

2. Obvladovanje problemov μS programske opreme, tj. systemskega in uporabniškega programiranja (mikrooperacijski sistemi, razvojna podpora, uporabniški programi). Pri razvojni podpori, ko sistem razvijamo (načrtujemo, gradimo in programiramo) je še posebej pomembna t.i. prečna programska oprema (cross software), ki omogoča uporabo obstoječih mini in makroročunalniških sistemov (prečni zbirniki, simulatorji, emulatorji, prevajalniki, interpretorji itn.).

3. Poznavanje problematike širšega sistema, v katerega bo μS vključen (tehnološki, procesni, časovni, človeški parametri uporabniškega, industrijskega, avtomatiziranega sistema).

Uporaba mikro računalnikov (kratko μR) se danes iz masovnega industrijskega, gospodarskega in poslovnega območja vse bolj širi v medicinsko, šolsko, izobraževalno in tudi v osebno oziroma domače okolje. Tu se μR pojavlja kot bistveni regulacijski in nadzorni element, ki omogoča izdatnejšo izrabo, učinkovitost in natančnost tehničnih naprav (avtomobil, gospodinjstvi stroji, domača administracija, prenos informacij, medicinske tehnična oprema, obdelovalne naprave), pojavlja se pa tudi kot učni in rekreacijski pripomoček v okviru osebnega računalništva (lasten, domači razvoj μS , uporaba μR v gospodinjstvih procesih, v drugih domačih tehničnih napravah, pri lastnem izobraževanju in v igrah doma) ter kot pomagalo in vzpodbudjevalnik kreativnega razmišljanja na področju vzgoje (tehnična in obča kultura) in izobraževanja (od predšolske vzgoje do šolanja na visokih šolah). Skratka μR in njegova uporaba bistveno spreminjata načine dosežanja tehničnega razmišljanja ter vplivata na občutek kreativnega doprinosanja pri razvoju in prizvodnji novih tehničnih sistemov. Zlasti v industrijskih okoljih narašča z uporabo μR zainteresiranost in s tem produktivnost tehničnih delavcev. μR transformacijska tehnologija prinaša tedaj občutek novega, vzpodbudnega in zadovoljujočega mlajšim in starejšim generacijam tehniških strokovnjakov in zlasti mladini.

A.P.Železnikar

avtomatska obdelava receptov v sr sloveniji

f.košir

UDK 681.3:61

REGIONALNA ZDRAVSTVENA SKUPNOST
LJUBLJANA

Avtomatska obdelava receptov je le ena izmed kompletnih in enotno organiziranih obdelav, ki so potrebne v realizaciji zdravstveno informacijskega sistema republike. V njenem izvajanju neposredno sodelujejo zdravstveni delavci v vseh organizacijskih oblikah nudenja zdravstvenega varstva. Z enkratnim in enotnim zajemanjem podatkov na receptih je omogočeno, da z ustrežno organizacijo in uporabo sodobne tehnologije obdelave podatkov zagotavljamo predvsem temeljit vpogled v predpisovanje in potrošnjo zdravil s strokovno medicinskega in ekonomskega vidika obenem pa lekarniškimi enotam omogočamo obdelavo finančnega poslovanja. Prisotna je velika količina podatkov, v Sloveniji predpišejo zdravniki mesečno nad 1 milijon receptov.

V članku je prikazan nastanek in razvoj projekta avtomatske obdelave receptov. Povdarek je zlasti na organizaciji in izvedbi projekta AOR. Obdelani so organizacijski elementi projekta kot tudi nekateri problemi in težave. Prikazana je nadalje II. faza oz. perspektive avtomatske obdelave receptov - zlasti neposredna uvedba računalniških obdelav v lekarnah, ki zagotavljajo vodenje materialne evidence blaga in obračunavanje prodaje in realizacije v lekarnah.

Na koncu so na kratko obravnavani doseženi rezultati avtomatske obdelave receptov v SR Sloveniji.

AUTOMATIC PRESCRIPTION PROCESSING (APP) IN SR SLOVENIA. APP is only one among the complete and unitary organised processings in development of the health information system in republic region. On this subject realization are taking part experts from all branches of fealth insurance. Uniform data input from the prescriptions at one place, enables us:

- through the proper organization and the modern technology, we offer an accurate insight in drug prescription, from medical or economical point of wiew,
- also we automatically process financial dealing of the Slovenian pharmacies.

At the input we gather a great population of data, in Slovenia doctors prescribe more than a million prescriptions, monthly.

In the paper we illustrate the beginning and development of APP, we outline organisation and execution of the projekt. There are described organizational elements with certain problems and difficulties. Article will also informate you about second phase and future in development of APP - use of immediate electronic data processings in the pharmacies to assure fast and efficient book-keeping and accountancy (of the drug sale).

At the end of the article we shortly discuss achieved results of APP in SR Slovenia.

U v o d

S perečo problematiko potrošnje zdravil se tako v svetu, kot pri nas v Jugoslaviji ukvarja mnogo organov in strokovnih institucij. Že bežen pregled strokovne zdravstvene literature nam omogoča spoznanje, da je ta problematika izredno temeljito in široko obravnavana s ciljem, da se ugotovijo subjektivni, objektivni medicinski in demografski faktorji, ekonomski faktorji ter faktorji, povezani z lokalnimi pogoji, ki bi lahko kakorkoli vplivali na potrošnjo. Vsem tem raziskovanjem je skupno to, da skušajo ugotoviti neko "dobro osnovo", na podlagi katere bi se lahko podvzeli ustrezni ukrepi za zmanjšanje potrošnje tako s strokovno medicinskega kot ekonomskega vidika. Toda, na podlagi podatkov, ki so nam na razpolago, ni v svetu države, v kateri bi se lahko na osnovi podatkov iz predpisanih receptov dobil vpogled v potrošnjo zdravil vse populacije, temveč so obdelave določenih podatkov omejene le na območje, kategorijo zavarovanja, posamezna zavarovanja kapitalističnega tipa itd. Nekaj kompleksnejših analiz imamo v socia-

lističnih državah posebno v ČSSR in NDR, kjer se izvajajo sistemi spremljanja potrošnje zdravil, ki zajemajo vse zavarovane osebe po vseh karakteristikah, z diagnozo bolezni in finančnimi podatki, vendar pa so ti sistemi omejeni na določene manjše teritorije. Z druge strani pa smo priča povečanemu interesu farmacevtske industrije, da dobi točnejši pregled o potrošnji zdravil povezan z rezultati svoje dokaj uspešne propagande, kar vse ima namen ugotoviti zgolj uspešnost prodaje določenega zdravila. Pri tem se trošijo ogromna sredstva, ki jih je farmacevtska industrija pripravljena vložiti za čim uspešnejšo prodajo. Tudi v Jugoslaviji imamo primere, da nekatere farmacevtske tovarne vlagajo znatna sredstva za takšna komercialna raziskovanja.

V Jugoslaviji je potrošnja zdravil v zadnjem desetletju skokovito naraščala na kar so opozarjali ne samo naši najvidnejši strokovni forumi, temveč je bila ta problematika obravnavana tudi na nivoju predstavniških teles

Zvezne skupščine in Skupščin posameznih republik in pokrajin. Socialno zdravstveni zbor Skupščine SR Slovenije je že v letu 1967 razpravljajal o problemih zdravstvenega varstva ter zadolžil Republiški zdravstveni center, da pripravi "Navodila za nadzor in racionalno predpisovanje zdravil, da prouči pozitivno listo zdravil ter poišče ustrezne rešitve na tem področju". S tem gradivom so bile načelno tudi začrtane osnovne smernice za AOR, ki so dozorevale in se dopolnjevale na osnovi razprav in sklepov: socialno zdravstvenega zbora zvezne skupščine leta 1970, socialno zdravstvenega zbora Skupščine SR Slovenije leta 1972, v odboru za zdravstvo in socialno varstvo v letih 1972 in 1973, itd in ne nazadnje zahtev in predlogov mnogih zdravstvenih delovnih organizacij in posameznikov. V vseh teh razpravah je bila prisotna ugotovitev, da predstavljajo izdatki za zdravila v zdravstvenem varstvu večji izdatek kot za celotno osnovno zdravstveno službo, da zdravniki letno napišejo v Sloveniji nad 10 milijonov receptov, vendar dokler nimamo primerjav ne moremo oceniti ali je to preveč ali premalo, da pišejo zdravniki na recepte številne podatke, ki pa jih zaradi velike količine na klasičen način ni mogoče obdelati, kljub temu, da so se v zdravstvenih skupnostih številni strokovni delavci ukvarjali predvsem s finančno realizacijo prejetih receptov.

Te in še druge ugotovitve so narekovala potrebo po temeljitejši proučitvi načina obdelave potrošnje zdravil, upoštevajoč vsa dognanja in izkustva doma in v svetu na področju uvajanja novih tehnologij obdelave podatkov v zdravstvu.

V tem sestavku ni mogoče obdelati vso širino tega projekta, želim na kratko orisati organizacijske priprave in prizadevanja za uresničitev, njegovo tehnično realizacijo, ki je bila kljub nekaterim kriznim obdobjem in problemom realizirana, predstavlja pa prvi in doslej edini uresničen korak v uveljavljanju avtomatske obdelave podatkov v zdravstvu za vso populacijo na območju SR Slovenije.

I. ORGANIZACIJSKE PRIPRAVE

1. Izhodišča programa AOR

Kot smo že v uvodu omenili je potrebno po temeljitejši obdelavi in kompleksni potrošnji zdravil narekovala predvsem prekomerna potrošnja in ugotovitve, ki so temeljile na domnevah, da kljub takšni potrošnji pacient ne dobi vedno ustreznega zdravila, ki bi bilo indicirano z njegovim obolenjem. Poleg te splošne ugotovitve pa smo ob izdelavi programa avtomatske obdelave receptov izhajali iz naslednjih konkretnih ugotovitev:

- v Sloveniji smo že v letu 1972 porabili za zdravila nad 35 milijard starih dinarjev, kar je pomenilo 16,7 odstotkov vseh izdatkov za zdravstveno varstvo in zavarovanje. Ti izdatki so z leta v leto nekontrolirano naraščali, kar je obenem pomenilo odtokanje zdravstvenega dinarja v druge gospodarske dejavnosti,
- letno so zdravniki napisali nad 11 milijonov receptov z 22 podatki na vsakem receptu. V Sloveniji smo imeli pet različnih vrst receptnih obrazcev za posamezne kategorije zavarovanih oseb,
- v Sloveniji je v skladu z zakonom o zdravstvenem varstvu zavarovano na takšen ali drugačen način vse prebivalstvo, kar lahko daje kompleksno obdelavo podatkov za vso populacijo vseh kategorij in bioloških starostnih skupin,

- receptov in podatkov, ki so se arhivirali na tedanjih komunalnih zavodih ni bilo mogoče zaradi velike količine niti finančno niti strokovno analizirati, posamezne vzorčne analize so predstavljale zaradi premajhne količine in določene neažurnosti neustrezno osnovo za ukrepanje,

- mesečno je posamezen zdravnik v Sloveniji predpisal tudi za 25 milijonov starih dinarjev receptov v breme sklada zdravstvenega zavarovanja, toda kljub temu terapija ni bila najustreznejša. Takšno stanje je bilo popolnoma v nasprotju z ugotovitvijo angleškega znanstvenika Brodie-ja, ki je na osnovi analize določene količine receptov zapisal: "Kolikor višji je strokovni nivo zdravnika in kolikor celovitejša je njegova orientacija proti pacientu kot osebnosti; toliko manjši je kvantum predpisanih zdravil vseh vrst",

- izreden porast potrošnje antibiotikov vseh vrst, ki jih naravnost "vsiljujejo" farmacevtske tovarne, kar pomeni za območje Slovenije nad 20% vseh zdravil in s tem tudi povsem nekontrolirano uporabo antibiotikov,

- zagotoviti in uvajati takšne pristope sodobne tehnologije v obdelavi podatkov v zdravstvenem varstvu, ki bodo zagotovili takojšnje in kompleksne podatke, s čimer bo omogočeno vplivanje na potrošnjo zdravil in finančnih sredstev ter dane nadaljnje možnosti za druga področja zdravstvenega varstva (ambulantne storitve, zobne, specialistične itd.),

- ne nazadnje uveljavljati in zagotavljati z ustavo opredeljeno manjšavo dela v zdravstvenih skupnostih, posebej še na področju lekarništva ter s tem prispevati k večji odgovornosti zdravstvenih delavcev za opravljene storitve uporabnikom zdravstvenega varstva.

Tako opredeljena izhodišča in smotri za avtomatsko obdelavo so odpirali številne probleme in vprašanja, ki jih je bilo potrebno rešiti v organizacijskem kot strokovno vsebinskem pogledu. Predvsem je bilo nujno izdelati vsebinski program projekta, določiti nosilca in organizatorje projekta, zagotoviti potreben strokovni kader za avtomatsko obdelavo, strokovno proučiti ustrezno uporabo računalniške tehnologije za takšno masovno obdelavo itd. V zvezi s tem so se predstavniki vseh 9 regionalnih skupnosti v Sloveniji (Celje, Nova Gorica, Koper, Kranj, Ljubljana, Maribor, Murska Sobota, Novo mesto, Ravne) dogovorili v februarju 1973 za izhodišča izdelave projekta AOR:

- nosilci organizacije avtomatske obdelave receptov v Sloveniji so Regionalne zdravstvene skupnosti, v katerih se vsklajujejo interesi uporabnikov in izvajalcev zdravstvenega varstva,
- strokovna služba regionalne zdravstvene skupnosti Ljubljana pripravi in izdela projekt enotne organizacije za centralno obdelavo receptov za vso Slovenijo. Vsaka skupnost si z razvojem lastne računalniške tehnologije zagotovi dodatne obdelave,
- vsebina programa avtomatske obdelave receptov mora zasledovati interese in koristi:
 - a) zdravstveni službi nuditi kvantitativno in kvalitativno nove podatke za strokovno medicinske analize

- b) strokovnim lekarniškim enotam omogočiti enotno realizacijo Zakona o evidencah v zdravstvu (materialno knjigovodstvo) in posodobiti finančno poslovanje na relaciji skupnost - lekarna,
- c) zdravstvenim skupnostim zagotoviti takojšen in celovit vpogled v potrošnje zdravil in s tem omogočiti realne osnove za samoupravno usmerjanje potrošnje v skladu z možnostmi in strokovnimi opredelitvami

- program za realizacijo pripraviti v dveh fazah:

1. faza mora zagotoviti podatke za zdravstveno službo in zdravstvene skupnosti
2. faza naj vsebinsko uresniči potrebe lekarn

- zaradi zagotovitve potrebnih osnovnih podatkov o zdravnikih, zdravstvenih delovnih organizacijah in lekarnah je k sodelovanju pritegniti Zavod SRS za zdravstveno varstvo, ki deloma razpolaga s kadrovske evidenco.

2. Opredelitev podatkov na receptu

Da bi uresničili zastavljene cilje in izhodišča za avtomatsko obdelavo potrošnje zdravil, je bila posebno pomembna opredelitev podatkov na receptu, ki morajo predstavljati osnovne

10	DDZ	OPZ	UP	K	KB	S	KVC	SMP	O	DRŽAVA		
11	REGISTRIRNA ŠTEVILKA									OG	M	Z
12	SZZ		ŠT. ZDRAVNIKA			ŠT. Z.D.O.						
13	141		14111111			112311111						
14	ŠT. LEKARNE		PHISP			9577111						
15	G.O.		GRUPA		SK.		PODSK.		CENA			
16	DATUM IZDAJE		T.LABORUM									
17	LETO ROJ.		SPEC.		E		D		P			
18	PRIMEN IN ABE											
19	IME DRUŽINSKEGA ČLANA (Mesec rojstva za otroke do 9 let)											
20	NASLOV BOLNIKA											
21	Rp.											
22	STAMP. Z.D.O.											
23	DATUM											
24	PODPIŠI ŠTAMP. ZDRAVNIKA											
25	ADRV											
26	Povabilo pripravil											

"vhodne" informacije za vse koristnike podatkov. Ob tem smo ugotovili, da nam stari recept nudi zadovoljive podatke, potrebno ga je bilo dopolniti z nekaterimi šifriranimi podatki (šifra zdravila, zdravnika, zdravstvene delovne organizacije) in se ob tem opredeliti, da na en recept pišemo le eno zdravilo. Ni nepomembno, če omenim, da je sedanji recept celo 14. varianta, kar kaže na to, da smo temu vprašanju ob ugotavljanju realnih pogojev za uvedbo nekega podatka posvetili izjemno pozornost. Vsebina podatkov je priložena v tabeli in razvidna iz posnetka recepta.

Verjetno je potrebno, da že na tem mestu odgovorim na tri vprašanja:

- zakaj na receptu ni diagnoze bolezni,
 - na receptu ni količine izdanega zdravila,
 - na receptu ni številke pacienta.
- a) oznaka diagnoze na receptu bi po mnenju avtorjev projekta AOR pomenila v prvi fazi uvajanja pravo "revolucijo" med določenimi zdravstvenimi delavci v Sloveniji predvsem zaradi njihovih subjektivnih stališč v zvezi z etiko in tajnostjo diagnoze, tako, da bi to vprašanje lahko ogrozilo celoten projekt AOR. V zadnjem času so bili tudi na tem področju napravljeni ustrezni pozitivni premiki, tako, da pričakujemo, da bo diagnoza zasedla že pripravljeno mesto na receptu
- b) glede podatka o količini predpisanega zdravila pa je potrebno naglasiti, da je bil od vsega početka predviden za drugo fazo realizacije programa AOR, ki se že izvaja za 17 lekarn v Sloveniji o čemer bo še nekaj povedanega. Poleg tega pa je mogoče za določene potrebe izračunati iz skupne cene ustrezno količino predpisanega zdravila.
- c) številke pacientov (zavarovanih oseb) v Sloveniji še nimamo uveljavljene. Ko bo to vprašanje organizacijsko-strokovno rešeno, bo ta številka na receptu.

PODATKI NA RECEPTU - VHODNI PODATKI

- kategorije zavarovanih oseb (DDZ, OPZ, UP, K, KB, S, KVC, SMP, O)
 - država (konvencije) (država)
 - registrska številka (zaposlitev zavarovane osebe - občina)
 - družinski član (JC)
 - spol (M, Ž)
 - regionalna zdravstvena skupnost (RZS)
 - številka zdravnika (št. zdravnika)
 - številka zdravstvene delovne organizacije (št. Z.D.O.)
-
- številka lekarne (št. lekarne)
 - prispevek (participacija) (prisp.)
 - številka recepta
 - galenska oblika (G.O.)
 - grupa, skupina, podskupina zdravila (grupa sk. zdravilo)
 - datum izdaje zdravila (datum izdaje)
 - taksa laborum (t. laborum)
-
- leto rojstva (leto roj.)
 - specialist (spec.)
 - vrsta programa zdravstvenega varstva (EDP)

3. Zagotovitev drugih osnovnih podatkov za AOR

Avtomatska obdelava receptov temelji predvsem na osnovi podatkov, ki jih zagotavljajo zdravstveni delavci v ustreznih zdravstvenih delovnih organizacijah: zdravstvenih domovih, bolnišnicah in lekarnah. Zaradi zagotovitve cilja o celovitosti obdelave podatkov na receptu je bilo potrebno v ta namen organizacijsko in programsko uresničiti formiranje osnovnih datotek s podatki:

- datoteka zdravnikov v SR Sloveniji
- datoteka strokovnih enot osnovne zdravstvene službe in hospitalov v SRS
- datoteka strokovnih lekarniških enot v SRS
- izdelati šifrant oziroma datoteko zdravil registriranih v Jugoslaviji

a) Datoteka zdravnikov za SRS

Osnovana je na podlagi podatkov Zavoda SRS za zdravstveno varstvo in dopolnjena z dodatnimi podatki ob popisu zdravnikov. Zajema vse zdravnike v republiki Sloveniji in pomeni tako istočasno centralno kadrovske evidenco. Zdravniki, ki imajo pravico predpisovati recepte v breme zdravstvenega varstva imajo v datoteki posebno oznako. Podatki se dopolnjujejo (ažurirajo) mesečno na osnovi sporočil regionalnih zavodov za zdravstveno varstvo, ki morajo po posebnih predpisih sporočati vse spremembe in nove primere Zavodu za zdravstveno varstvo SRS.

Vsak zdravnik je identificiran z dvema številčkama, to je redno šifro, ki jo ima ne glede na zaposlitev oziroma delovno mesto. Ta številka se vpisuje na receptni obrazec. Druga šifra je "klasifikacijska", ki opredeljuje zdravnika v zaposlitvi glede na regijo, zdravstveni dom ali hospital in njegovo specialnost v skladu s pravilnikom o specializaciji zdravstvenih delavcev (Ur. list SRS, 6/74). Zadnje osmo mesto šifre je rezervirano za označitev "subsposialnosti", kar pa v Sloveniji še ni enotno urejeno.

b) Datoteki strokovnih enot zdravstvene službe

Datoteki zajemata vse osnovne podatke za zdravstvene domove, hospitale z oddelki in strokovne lekarniške enote v SRS. Da bi zagotovili čimpopolnejšo kontrolo nad porabo receptov in ugotovili tudi potrebne podatke o predpisovanju zdravil v posameznih strokovnih enotah zdravstvene službe je določeno, da se za vsak zdravstveni dom ali hospital tiskajo posebni recepti (tromestna predtiskana šifra na receptu), ki se ob predpisu zdravila še dopolni s šifro enote ali oddelka. Zdravniki lahko uporabljajo le recepte svoje delovne organizacije, zamenjava je nedopustna.

c) Šifrant zdravil

Avtomatska obdelava potrošnje zdravil v Sloveniji je zasnovana na govoreči sedem-mestni šifri zdravila, kar pomeni, da nam prepoznavanje določenega znaka ali več znakov da tudi že določeno informacijo o zdravilu. Šifra je formirana na osnovi JKL v sodelovanju s strokovnjaki Centra za zdravila iz Zagreba. Posamezne šifre pomenijo:

1	0	5	6	2	2	2
---	---	---	---	---	---	---

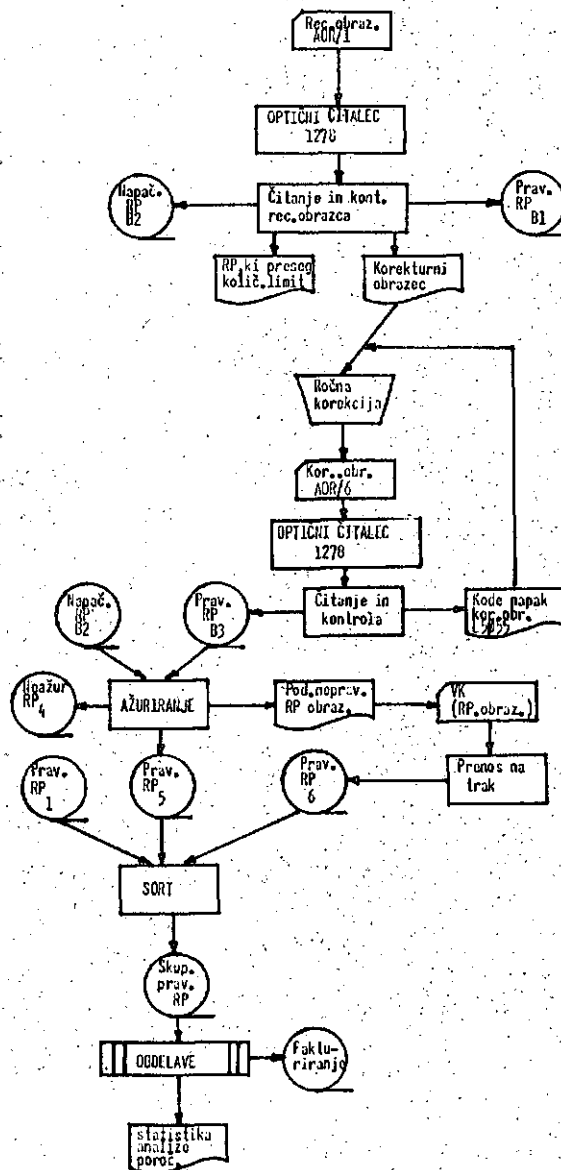
galenska oblika
farmakodinamska grupa
po JKL
podskupina
definicija zdravila
(proizvajalec)

Šifre so opredeljene po oмотih (ambulantnih in kliničnih) po jakosti zdravila in po proizvajalcih, tako domačih kot tujih.

K taki opredelitvi šifre so nas predvsem vodili znanstveni dosežki in stališča Svetovne zdravstvene organizacije, katera je priporočila, da se pri šifriranju zdravil koristijo pozitivni dosežki široko uporabljene MKB (mednarodne klasifikacije bolezni). Označitev zdravila tudi po proizvajalcu je potrebna za spremljanje asortimana njegovih proizvedenih zdravil (raziskava tržišča). Takšno revidirano jugoslovansko klasifikacijo zdravil je osvojil tudi stalni strokovni svet za farmakoterapijo pri Zveznem komiteju za zdravstveno in socialno varstvo v juliju 1974. Kompletan šifrant zdravil je bil tiskan v posebni izdaji in posredovan vsem zdravstvenim delavcem za dnevno uporabo za šifriranje receptov. Za novo registrirana zdravila posredujemo potrebne dopolnitve.

II. PROGRAMSKA IZVEDBA PROJEKTA AOR

1. Optično čitanje vhodnih podatkov



Izredno pomembno in odločujoče vprašanje je predstavljalna opredelitev o uporabi tehnologije obdelave oziroma ugotovitev kateri vhodni računalniški medij je primeren za takšno masovno obdelavo podatkov za republiko. Pri opredelitvi za optični čitalec IBM 1287 smo izhajali iz naslednjih predpostavk:

- da uvedemo sistem, ki bo lahko služil za zajemanje podatkov tudi pri drugih aplikacijah v dejavnosti zdravstvenega varstva (ambulantne storitve, zobozdravstvene itd.),
- kompleksna obdelava podatkov zdravstvenega varstva mora v perspektivi zagotoviti tudi obdelave za plačilo opravljenega dela (kjer ne sme biti vmesnega člana kot so luknjane kartice in p. zaradi točnosti finančno-plačilnih odnosov) ter da Rp predstavlja zdravstveno listino, pri kateri med izvorom podatkov (zdravnik, medicinska sestra, farmacevt) in koristnikom podatkov (uporabniki v zdravstveni skupnosti) ni posrednika, ki je lahko zmotljiv oz. je zagotovljena tajnost podatkov,
- da bo z ozirom na velike količine vhodnih podatkov zagotovljena ustrezna hitrost in ažurnost v obdelavi in njihovi takojšnji uporabnosti za ukrepanje. Na podlagi normativov za luknjane kartice bi namreč za obdelavo receptov potrebovali 40 luknjačev in verificirk v dveh izmenah, kar bi še vedno pomenilo samo pripravo za obdelavo receptov!

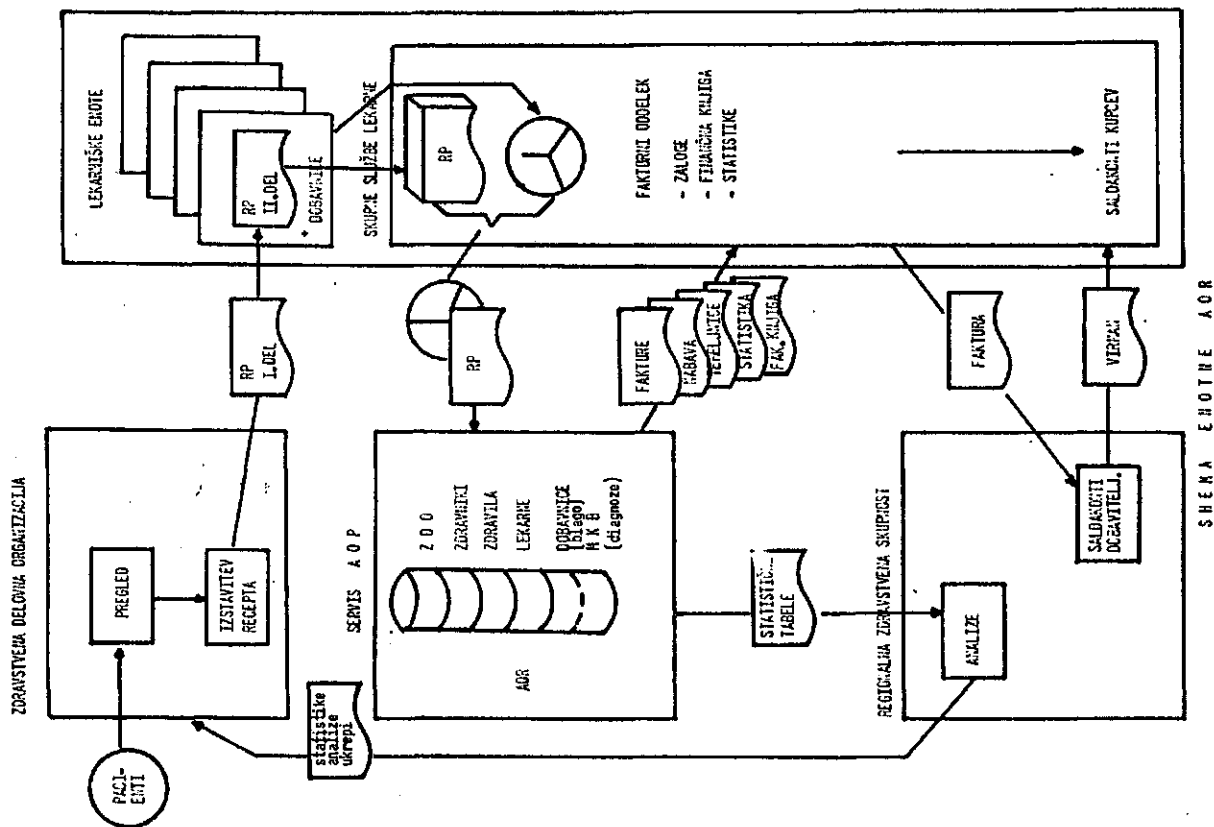
Tehnologija zajemanja vhodnih podatkov z optičnim čitanjem je v svetu že močno uveljavljena posebno v ZDA, ZR Nemčija in Švedska. Uporablja se za vse masovne obdelave, kjer mesečna količina vhodnih dokumentov presega sto tisoč. Čitalnik predstavlja le eno izmed vhodnih enot računalnika, deluje na principu on-line in ima sposobnost, da poprečno v sekundi prečita 2,5 izpolnjena recepta, ki se

prenašajo na periferni pomnilnik in dalje obdelujejo z željenimi programi na kateremkoli računalniškem sistemu. Zaradi posebne tehnologije optičnega čitanja, je kreacija dokumenta za zapis podatkov (recepta) strokovno izredno pomembno in zahtevno opravilo. Dokument mora ustrezati posebnim standardom (velikost polj, podatkov, harva, kvaliteta papirja itd.), kar je vse podvrženo strogim testiranjem na posebnih aparaturnah. Zaradi pričakovanih napak v izpolnjevanju receptov, kjer sodeluje v Sloveniji 12 tisoč zdravstvenih delavcev (tudi drugod po svetu je 10 % napak normalen pojav) smo uvedli "korekturni obrazec", ki zagotavlja, da se katerokoli polje na receptu ročno popravi in s ponovnim čitanjem tega obrazca dopolnijo manjkajoči ali nepravilni podatki na receptu. Drugi način popravljanja napak (samo zaradi neustrezne pisave) je on-line popravljanje nepravilnih znakov na tastaturi optičnega čitalnika, kar pomeni takojšnjo zagotovitev pravih podatkov.

Za kompleten projekt AOR je bilo do sedaj napisanih že nad 150 programov, ki zagotavljajo številne možnosti obdelav po željenih vidikih in zahtevah.

2. Izvedba druge faze

V zvezi z realizacijo zakona o evidencah na področju zdravstva predstavlja druga faza projekta AOR program vodenja materialne evidence v lekarnah in kompletnega avtomatskega obračuna realizacije poslovanja, ki ga je po dogovoru pripravil Inštitut za javno upravo v Ljubljani - Center za informatiko v javni upravi - skupaj s Celjskimi lekarnami.



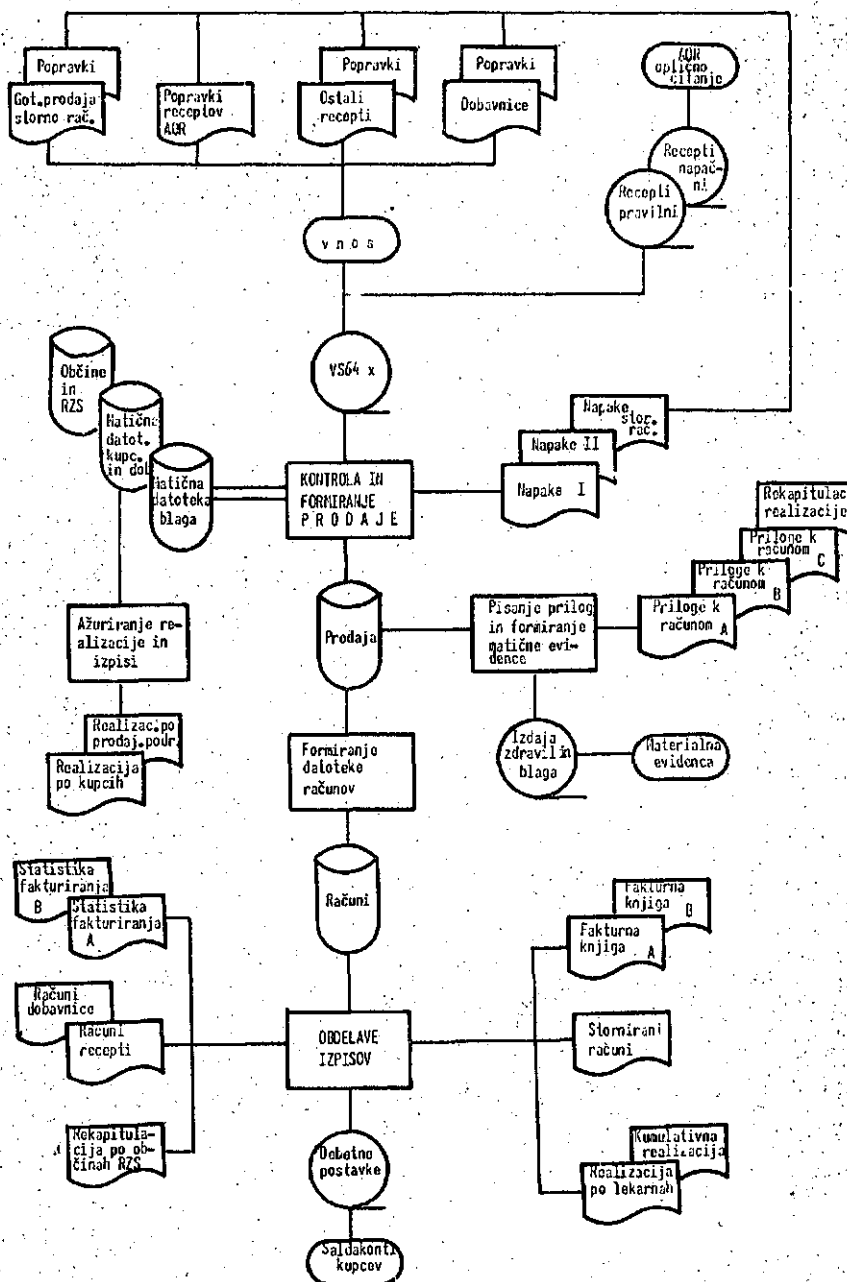


DIAGRAM OBDELAVE

Programska naloga vsebinsko zajema:

- vnos informacij izdaje zdravil za druge republike, posebne kategorije zavarovanih oseb, po naročilnicah organizacij in ustanov na računalniške pomnilnike,
- prevzem informacij o izdaji zdravil v SR Sloveniji iz obdelave AOR,
- združitev teh informacij v skupno datoteko prodaje zdravil in ostalega blaga,
- formiranje datotek:
 - a) faktur in realizacije,
 - b) obračunov za Regionalne zdravstvene skupnosti v SR Sloveniji,
 - c) faktur za skupnosti zdravstvenega zavarovanja iz drugih republik,
 - d) faktur za druge kategorije zavarovanih oseb in delovne organizacije ter ustanove,
 - e) vse izhodne informacije oziroma tabele.

Projekt je programiran kot pilotska študija, ki se je začel izvajati januarja 1976 na območju Celjskih lekarn in se postopoma aplicira tudi v drugih lekarnah na območju SR Slovenije. Potek obdelave je naslednji:

- a) recepti, realizirani v posameznih lekarniških enotah (Celjske lekarne imajo 13 enot) se v vsaki enoti dopolnijo s podatki o šifri zdravila in količini. Lekarniške enote z ozirom na njihovo velikost, oddajo recepte v zbirno mesto (skupne službe) dnevno ali pred zaključkom dekade opremljene z vodečim obrazcem, ki vsebujejo šifro lekarniške enote in datum dekade.
- b) zbirno mesto oddaja recepte direktno v obdelavo 12., 22. in 2. v mesecu. Poleg tega se na posebni zajemalni enoti zajemajo še podatki iz receptov za zavarovance v drugih republikah, dobavnice za izdana

zdravila zdravstvenim delovnim organizacijam in drugim TOZD ter recepti, ki so bili "poškodovani" v obdelavi prejšnje dekade ali zaradi pomanjkljivosti izpušeni v korekturnem postopku. Količina teh podatkov za vnos se giblje med 5 in 8 % od skupno realiziranih receptov v dekadi,

- c) obdelave na računalniku so razdeljene v posamezne faze, ki je vsaka v vsebinski, logični in časovni soodvisnosti od uspešne izvedbe predhodne. Osnova za nadaljnje obdelave je zajemanje podatkov z optičnim čitanjem receptov, ki obsega:
- odčitavanje podatkov s kontrolo in vključevanjem datotek: kupci-dobavitelji, zdravila (vrsta, cena, količina, časovni normativi za magistralna zdravila), zdravniki, zdravstvene delovne organizacije, lekarne, občine,
 - zapisovanje podatkov na: trak prodaje, trak pravilnih po JKZ (Jugoslovska klasifikacija zdravil), trak napačnih, izpis korekturnih obrazcev za podatke, ki so bili spoznani kot formalno in logično nepravilni,
 - fizično ločevanje receptov na optičnem čitalniku na tri dele: pravilni, napačni s tekočo numeracijo (zdravniški del), napačni (lekarniški del).

Trak prodaje vsebuje vse elemente za izdelavo faktur, v tej fazi priključujemo tudi vhodne podatke o dobavnica in receptih iz prejšnje dekade (točka b). Fakture se takoj izpišejo kot tudi dodatni dokumenti za finančno poslovanje lekarne (finančna realizacija po enotah, fakturna knjiga, temeljnica ...). Vsa izhodna dokumentacija se vrne v zbirno mesto za pregled in nadaljnjo distribucijo faktur,

- d) korekturni postopek poteka po dveh tirih:
- za napake v zdravniškem delu podatkov se izpolnjujejo korekturni obrazci,
 - recepti z nepopolnimi podatki v lekarniškem delu se vračajo v zbirni center in se po izvršenih popravkih priključujejo v naslednji dekadi za ponovno obdelavo,
- e) podatki iz receptov, popravljeni v korekturnem postopku se združujejo s trakom prodaje nove dekade, kar pomeni izstavitev faktur in drugih ustreznih dokumentov za lekarno.
- V mesečni obdelavi so tako zajeti vsi recepti, razen 5 do 8 % zadnje dekade, ki se fakturirajo v naslednjem mesecu,
- f) poleg mesečnih tabel za lekarne, se za potrebe skupnosti in zdravstvenih delovnih organizacij izdelujejo vse informacije in analize, ki izhajajo iz 1. faze programa AOR (podatki o predpisanih receptih za zdravnike, zdravstvene delovne organizacije, po farmakodinamskih skupinah in podskupinah, po specialnostih zdravnikov itd. - 14 tabel in analiz), ki se neposredno razpošiljajo po vseh regionalnih zdravstvenih skupnostih,
- g) za lekarne so računalniško rešena naslednja področja, ki so med seboj integralno v obdelavi povezana:

- fakturiranje in obračun realizacije
- finančno knjigovodstvo z glavno knjigo in bruto bilanco
- obračun osebnih dohodkov

- evidenca o izdelanih galenskih pripravkih in porabljenih ingrediencah
- druge pomožne obdelave, potrebne, da zagotovijo nemoteno in povezano strojno obdelavo (statistike, itd.).

V zaključni testni obdelavi sta še aplikaciji:

- saldakonti kupcev in dobaviteljev
- evidenca nabave, prodaje, zalog zdravil in ostalega blaga.

Posebno rešitev terja ročna prodaja zdravil in ostalega blaga v lekarni, ki jo je mogoče vključiti v do sedaj realizirani del obdelav. V strokovni pripravi so nekatere rešitve, vendar pa to vprašanje predstavlja poseben organizacijsko strokovni problem in ne nazadnje določene nove tehnične rešitve.

III. IZVAJANJE IN DOSEZENI REZULTATI AOR

S 1. marcem 1974 se je dejansko začelo izvajanje programa AOR s tem, da so začeli zdravniki predpisovati zdravila zavarovanim osebam na novih receptnih obrazcih. Ugotoviti je potrebno, da so tako zdravstveni delavci kot pacienti sprva ustrezno sprejeli novosti, ki so bile potrebne za uvajanje programa AOR, v kasnejših mesecih pa so se pojavili nekateri posamezniki, ki so poskušali program AOR prikazati kot "muho enodnevnico", kot "zablodo" idealistov, kot megalomanijo itd. To so bili predvsem tisti, ki niso mogli ali hoteli razumeti potrebe po uvajanju avtomatske obdelave podatkov v zdravstvo. Z nemogočimi argumenti kot uporaba svinčnika, kaligrafska pisava, dodatne obremenitve in s tem zastoji v ambulantah so želeli opravičiti strah in odpor do individualne kontrole zdravnika (njegovga številka). K takšnemu razpoloženju so pripomogli tudi nekateri dnevni časniki, ki so povdarjali ogromne stroške za pripravo, uvoz dragih strojev, papirja itd., kar so bile popolnoma neresnične informacije, saj nismo imeli in še nimamo lastnih strojnih kapacitet, vse organizacijske priprave, vključno z recepti za prve tri mesece so stale le 70 starih milijonov din. Ta v določeni meri krizna situacija je bila zaključena z razpravo v Socialno zdravstvenemu zboru skupščine SRS, kjer je republiški sekretar za zdravstvo in socialno varstvo podal ustrezno obrazložitev in stališče o programu AOR v katerem med drugim pravi: "Avtomatska obdelava receptov je le eden prvih pristopov h kompleksnejši avtomatski obdelavi podatkov za področje zdravstva. Le s sprotnimi in po potrebah obdelanimi podatki bo moč do kraja izpeljati svobodno menjavo dela, spremljati zdravstveno stanje prebivalstva ter uresničiti družbene interese na področju zdravstvenega varstva".

Večkrat smo že zapisali, da je program AOR dolgoročen, ki ne more takorekoč čez noč v nekaj mesecih dati vse tisto, kar smo si na strokovnem in finančnem področju delovanja želeli preteklih 30 let. Zato lahko preteklo obdobje, ki ni bilo lahko in vse polno tako objektivnih kot subjektivnih kriznih situacij, imenujem komaj začetek začetka, ki naj zagotovi možnosti za nadaljnje delovanje in izpolnjevanje programa AOR. Zagotavljanje potrebne vsebine je predvsem naloga samega projekta AOR, ki mora v postopnem razvoju dati ustrezne rezultate in informacije za ukrepanje.

Prve rezultate avtomatske obdelave podatkov lahko opredelimo v tri osnovne kategorije:

- pravilno zagotavljanje vhodnih podatkov za obdelavo
- posredni rezultati ekonomsko finančne narave
- konkretni kvalitativno in kvantitativno

novi podatki za strokovno medicinske analize.

1. Dejansko je izpisovanje novega recepta v prvem obdobju terjalo več časa od zdravstvenih delavcev, posebno še, ker je ta predstavljal popolno novost v izvajanju zdravstvenega varstva in ne nazadnje tudi novost na področju tehnologije avtomatske obdelave podatkov. Te začetne težave so seveda pogojevale tudi določen odpor, ki je bil večkrat tudi rezultat neustreznega pristopa različnih združenj in forumov do AOR nasploh. Prizadetost med zdravstvenimi delavci se je najbolj kazala na tistih področjih, kjer je bilo zanemarjeno ustrezno strokovno informiranje vseh sodelujočih in niso bili jasno predočeni dolgoročni cilji obdelav. V teku samega delovnega postopka ob izpisovanju receptov pa so sami zdravstveni delavci ugotovili, da lahko tudi to postane delovna navada kot vsaka druga in jo je moč z različnimi organizacijskimi prijemi tudi ustrezno organizirati. Tako lahko ob obdelavi receptov ugotovljamo, da je večina lekarn v Sloveniji ustrezno pristopila k izpolnjevanju podatkov in da so rezultati izredno ugodni glede števila napak. Tako je večje število lekarn, ki dosegajo komaj 4% napak, so pa tudi nekatere, kjer se število napak povzpne tudi do 30% pri enaki količini in vrsti zdravil. Iz tega jasno izhaja določen odnos posameznikov, ki še niso mogli "spremeniti" mišljenja do AOR. Po podatkih, ki so na razpolago zunaj naših meja, znaša lahko povprečen odstotek napak, ki se zagotavlja dobre rezultate od 7-8%.

Večje število napak ugotovljamo pri podatkih, ki jih izpisujejo zdravniki in drugo osebje ambulant, kjer se giblje odstotek od 25% - 35%, kar pa ne pomeni, da ga ni mogoče občutno znižati. Te napake so namreč predvsem posledica logično nepravilnega izpolnjevanja podatkov in ne toliko formalno nepravilnega načina pisanja znakov, čeprav se v nekaterih ambulantah tudi to dogaja. Določene napake so posledica neinformiranosti zdravstvenih delavcev o sami vsebini recepta, ki zahteva točno določene podatke v skladu z navodili in ne morebitno improvizacijo podatkov. Posebej želimo naglasiti, da se vsi podatki v procesu obdelave popravijo že direktno pri čitanju ali s korekturnimi obrazci, tako, da končne obdelave zagotavljajo vključitev vseh predpisanih receptov, s pravilnimi podatki, obenem pa je točno znano kakšne napake in kdo jih povzroča. S potrebno ažurnostjo v obdelavah za pretekli mesec opozarjamo posameznike na te napake v skladu s samoupravnimi dogovori pa bo potrebno tudi ustrezno ukrepati.

2. Čeprav je osnovni namen avtomatske obdelave receptov strokovno medicinski aspekt, ki naj po določenem obdobju vpliva tudi na ekonomsko finančni, lahko zapišemo, da je ta podan že v dosedanem obdobju, saj so se izdatki za zdravila v Sloveniji umirili in zmanjšali. Iz tabele so razvidni odstotni deleži sredstev, porabljenih za osnovno, specialistično zdravstveno varstvo in za zdravila v obdobju 1971-1977 v odnosu na sredstva za celotno zdravstveno varstvo:

ODSTOTNI DELEŽI V %	1971	1972	1973	1974	1975	1976	1977
ZDRAV. VARSTVO	100	100	100	100	100	100	100
- OSNOVNO	15,37	15,70	15,67	15,87	15,57	15,92	15,83
- SPECIALISTIČ.	7,68	8,28	9,00	9,92	10,44	10,92	12,56
- ZDRAVILA	16,79	15,89	15,32	14,44	14,49	13,97	13,33

3. Podatki, ki izhajajo iz AOR, predstavljajo novo kvaliteto osnovo za strokovne analize in kriterij za delo zdravstvenih delavcev. Žal so bili ti podatki po izkušnjah iz preteklih let najmanj korišćeni v "bazi", iz katere izhajajo. Vendar se tudi ta situacija sčasoma popravlja. Namreč povsod tam, kjer so medicinski sveti in zdravstveni delavci sami aktivno sodelovali pri analizi informacij, se je predpisovanje zdravil strokovno bistveno izboljšalo. Na osnovi strokovnih kazalnikov se je pristopilo tudi k reševanju farmakoterapevtskih doktrinarnih opredelitev (S. Derčevi pediatrični dnevi - junij 1977, XIX Tavčarjevi dnevi - november 1977) in k realizaciji priporočil IV. evropskega simpozija o klinično farmakološki evalvaciji zdravil glede racionalnega predpisovanja zdravil z ozirom na kritično izbiro zdravila.

Posebej naročene obdelave so koristile razne strokovne institucije in združenja. Tako je bila za ljubljansko podružnico Slovenskega zdravniškega društva za letni občni zbor v letu 1977 obdelana FS 08-psihofarmaka - s podskupinami. Nadalje je bila za diabetološko službo prikazana poraba nekaterih podskupin FS 04 - zdravil, ki delujejo na endokrine bolezni in to: insulina ter bigvanidinskih in sulfonamidnih antidiabetičnih sredstev po regijah in za republiko. Celovit pregled najvidnejših rezultatov terja poseben članek oziroma strokovno medicinsko obdelavo!

IV. ZAKLJUČEK

1. Vse organizacijske in programske rešitve so v praksi preizkušene, obdelave dajejo bogate rezultate.

2. Bistvene prednosti uvedbe AOR so:

- zagotavljamo celovit in sproten pregled predpisovanja zdravil za regije in republiko
- omogočeno je vplivanje na gibanje potrošnje zdravil v strokovnem in ekonomskem smislu v okviru zdravstvenih skupnosti
- dana je pomembna vzpodbuda strokovnim zdravstvenim institucijam za oblikovanje farmakoterapijske doktrine
- zastavljena je posodobitev in racionalizacija administrativno finančnega poslovanja v lekarni in s tem sprostitev farmacevtov za strokovno delo
- omogočeno enostavno in efektivno vodenje materialne evidence v lekarnah
- zastavljen je prvi korak v uresničevanju zdravstveno informacijskega sistema republike, ki naj enotno in celovito povezuje zdravstvo na tem območju.

Literatura:

1. Ugotovitve SZO Uporaba in zloraba zdravil in medikamentov. Publikacija SZO: 1975; 20. zasedanje skupščine SZO, Ženeva, maj 1975
2. KOŠIR F., MIKUŽ Razvoj avtomatske obdelave receptov v Sloveniji, T., KRAVANJA B. Pharmaca 14: 27-36, 1976
3. MIKUŽ T., KOŠIR Suvrenena analiza potrošnje lijekova obradom receptata pomoću F., KRAVANJA B., elektroničkog računala u SR Sloveniji, Zdravstvo 10, HOLEŠEK K.: 58-71, 1974
4. KOŠIR F.: Iskustva avtomatske obrade receptata u Sloveniji, Savetovanje o potrošnji lekova u Beogradu, Zbornik radova 60-63, Beograd 1976

gibki diski

r.murn
d.peček

UDK 681.327.63

Institut "Jožef Stefan"
Ljubljana

V članku želimo seznaniti bralce z lastnostmi in značilnostmi gibkih (floppy) diskov. Najprej sta opisana mesto in vloga gibkih diskov, nato je opisana izvedba disket. Podrobno je opisan način kodiranja ter standardni format sledi in sektorja. Na koncu so podane osnovne značilnosti pogonskih enot in navedeni sodobni vmesniki za funkcionalno povezavo pogonskih enot z gostiteljskim računalnikom.

FLOPPY DISK SYSTEMS - In the paper the features of floppy disk are discussed. The significance of floppies, the realization and the main characteristics of standard diskettes are presented. The coding form and standard format of tracks and sectors are discussed in detail.

Finally, the basic characteristics of floppy disk drivers and controllers to enable the interconnection between floppy disk and the host computer are given.

1. UVOD

Gibki (Floppy) diski ali diskete postajajo vedno bolj pomembno sredstvo za shranjevanje digitalnih podatkov. Gibki diski so slično kot ostali diski pomnilniki s sorazmerno hitrim naključnim dostopom. V bistvu so serijski pomnilniki s časom dostopa nekaj desetink sekunde. V primerjavi z ostalimi masovnimi pomnilniškimi sistemi razvrščamo gibke diske po svojih lastnostih in ceni med običajne diske (s čvrsto površino) in digitalne kasetne sisteme. Posebna prednost disket je v hitrosti v primerjavi s ceno, v zanesljivosti in prenosljivosti. Vse omenjene lastnosti narekujejo vedno večjo uporabo disket v mikroročunalniških sistemih. V razpredelnici 1 je prikazana primerjava nekaterih tehničnih podatkov različnih pomnilniških sistemov.

	IBM2315 disk	IBM3740 disketa	Digitalna kasetna	Tonska kasetna	Enota
kapacit. podat.	48	3	6	0,84	Milj.bit. (ne format.)
pov. č. dostopa	0,035	0,45	20	120 (roč.k.)	sekund
hitr. prenos. podat.	2500	250	10	0,3	Kbit/s

Razpredelnica 1.

Gibke diske so prvi uporabili pri trvdki IBM v sredini leta 1960, medtemko je industrijska proizvodnja sistema 3740 stekla v začetku 1973 leta. IBM 3740 se smatra kot standard. Poleg omenjene trvdke obstaja danes še vrsta

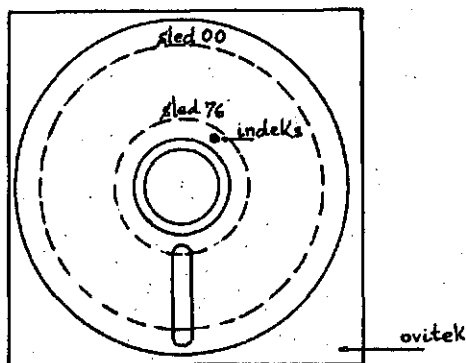
trvdk, ki proizvajajo IBM 3740 združljive (kompatibilne) sisteme. Po nekaterih ocenah /2/ je bilo v letu 1975 od vseh prodanih disketnih sistemov približno 67% IBM 3740 združljivih, tako po izvedbi pogonske enote kot po formatu. Večina preostalih pa se je od IBM standarda razlikovala predvsem po načinu formatiranja ali po velikosti sektorjev. Navajamo nekaj pomembnejših proizvajalcev IBM 3740 združljivih pogonskih enot: CALIFORNIA COMPUTER PROD., CONTROL DATA CORP., GENERAL SYSTEMS INTERNATIONAL., HELITRON, MEMOREX, ORBIS SYSTEMS, PERTEC, SAGEM (Francija), SHUGART ASSOCIATES. Izdelki trvdk CALIFORNIA COMP. PRODUCTS in DYNASTOR niso združljive z IBM standardom.

2. IZVEDBA DISKETE

Sama disketa (vloček) je iz lahkega gibkega materiala. Premer standardne diskete je 7.8" (19,8 cm) z osno odprtino 3,81 cm. Podatki so shranjeni na površini, ki je prevlečena z magnetnim materialom.

Disketa je vstavljena v kvadratni ovitek (cartridge) velikosti 8x8" iz papirja ali plastike. Notranjost ovitka je taka, da omogoča majhno trenje pri vrtenju in dobro ščiti površino diskete. Ovitek ima navadno tri odprtine: osno odprtino za pogon diskete, odprtino za čitalno/pisalno glavo in indeksno odprtino. Pri nekaterih sistemih npr. pri SA 901 trvdke SHUGART imamo še odprtino za zaščito zapisanih podatkov. Obe zadnji omenjeni odprtini omogočata zaznambe svetlobnih impulzov. V zadnjem času so se pojavile na tržišču diskete, ki so na obeh straneh prevlečene z magnetno snovjo tj. dvostranske diskete. Primer dvostranske diskete predstavlja sistem SA 850 trvdke SHUGART.

Podatki so shranjeni na koncentričnih sledih (Tracks), teh je pri IBM standardu 77. Podatki so torej shranjeni na površini med sledjo 00 in sledjo 76 (slika 1). Optična indikacija pokáže pozicijo čitalno/pisalne glave na sledi 00.



Slika 1. DISKETA

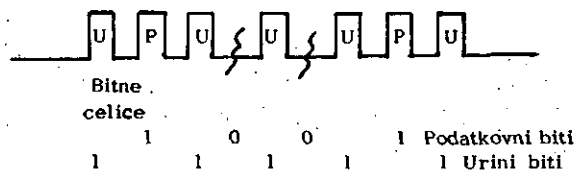
Vsaka sled je razdeljena na enake sektorje, ki so ustrezno adresirani. Razlikujemo dva načina sektoriranja. Prvega imenujemo mehko ali programsko sektoriranje, drugega pa trdo ali stalno sektoriranje. Pri slednjem načinu identificiramo vsak sektor z odprtiniami v ovojnici in disketi (ena luknjica za sektor) ter svetlobnim pretokom skozi te odprtine. Običajno imamo 32 sektorjev in prav toliko odprtin. Oba načina imata indeksno odprtino za identifikacijo začetka sledi.

Značilne vrednosti standardne diskete so:

Premer diskete :	7,8"
Ovitek :	8x8"
Število sledi :	77
število sektorjev na sledi :	26
število bytov v sektorju :	128 bytov
Krožna hitrost :	360 obr/min
Gostota bitov :	3200 b/inc
Čas ene sledi :	166,67 ms
Povprečen čas, da se disketa zavrti za željeno adresno (Average Latency) :	83,33 ms
Hitrost prenosa podatkov :	250 Kbitov/s
Čas dostopa od sledi do sledi :	10 ms
Način zapisa :	frekvenčna modulacija (FM)

3. ZAPISNI FORMAT

Za zapis podatkov na disketo se pri standardni izvedbi uporablja FM način ("frekvenčna modulacija"). Podatkovni impulzi se kodirajo med urine impulze, tako ima niz enic drugačno frekvenco impulzov kot niz ničel. Bitna celica je definirana kot perioda med prednjo fronto urinega impulza (bita) in prednjo fronto naslednjega urinega impulza. Med obema impulzoma v bitni celici se lahko nahaja podatkovni bit (logična 1, če je impulz prisoten). Byte je definiran kot zaporedje osmih bitnih celic. Najbolj značilna bitna celica je označena kot celica 0 in najmanj značilna kot bitna celica 7. Pri zapisu na disketo se zapiše najprej bitna celica 0 in kot zadnja celica 7. Pri čitanju se prenese iz diskete najprej bitna celica 0 in nazadnje celica 7. Na sliki 2 je prikazan primer zapisa podatkov v FM načinu (binarna vrednost podatkov 1001).

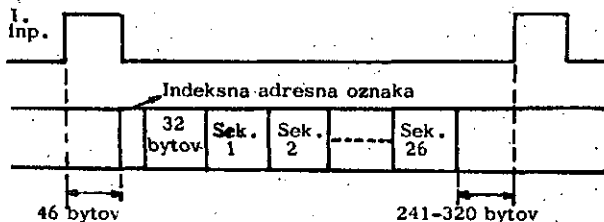


Slika 2. Način kodiranja podatkov
U-urini biti, P-podatkovni biti

Večina današnjih disketnih sistemov uporablja opisani način zapisa podatkov. S ciljem, da se izboljšajo lasnosti disket, so pričeli uporabljati tudi druge zapisne načine. Omenili bomo dva taka načina in to MFM (modificiran FM) in M2FM ali način z dvojno gostoto. Podatki se kodirajo z "grupno" kodo namesto z FM. Ti načini zahtevajo bolj komplicirano elektroniko in so tudi nekoliko manj zanesljivi od zgoraj opisanega načina.

V naslednjem odstavku si bomo ogledali format IBM standarda, ki predstavlja najbolj razširjen način "mehkega" sektoriranja. S pomočjo formata omogočimo točen zapis ter sinhronizacijo prenosa podatkov. Format sestavlja posamezna polja kot so polje podatkov, adresna polja in krmilna polja ter vrzeli (gaps). Vsak sektor vsebuje identično zaporedje polj. Vrzeli predstavljajo varnostno področje med posameznimi polji in preprečujejo, da bi variacije v hitrosti pogonskega motorja in v sinhronizaciji vplivale na točnost. Byti vrzeli so kodirani tako, da vsebujejo samo urine bite (podatkovni biti so vsi 0). Na sliki 3 je podan standardni format sledi. Po zaznamvi indeksne odprtine, ki služi kot startna referenca, ima vsaka sled enako razporeditev podatkovnih adresnih in krmilnih polj. Približno 46 bytov vrzeli za prednjo fronto indeksnega impulza se začne indeksna adresna oznaka (mark) dolžine 1 byta. Ta oznaka služi kot mejnik, ki določi, da se od njega naprej po točno 32 bytov začne prvi byte prvega sektorja. Nato po vrsti sledi 26 identičnih sektorjev in vrzel do začetka indeksnega impulza.

Zadnja omenjena vrzel nima točno predpisane dolžine in zaradi tolerance nekoliko varira. Običajno vsebuje od 241 do 320 bytov. Omenjeni format sledi dopušča uporabniku za zapis podatkov okoli 60% prostora (neformatirana kapaciteta sledi je 41K bitov).



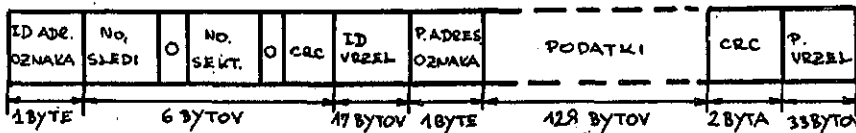
Slika 3. FORMAT SLEDI

Poglejmo še format sektorjev. Vsak sektor je razdeljen na štiri glavna polja: na identifikacijsko polje (ID) in identifikacijsko vrzel ter na podatkovno polje in podatkovno vrzel. S pomočjo identifikacijskega polja je omogočena popolna identifikacija sektorja. To polje se začne z ID adresno oznako dolžine 8 bitov, nato sledi 8 bitna adresa sledi, nato 8 ničel, potem 8 bitna adresa sektorja,

ki ji sledi 8 ničel in na koncu polja 16 bitna beseda za preverjanje -CRC (Cycle Redundancy Check). Adresa sledi in sektorja je definirana s fizičnim zapisom in zato običajno uporabnik ne more spreminjati ID polja. ID polju sledi ID vrzel dolžine 17 bytov. S podatkovno adresno oznako dolžine 16 byta se začne najvažnejše polje tj. podatkovno polje. Za uporabnika je na voljo 128 bytov, dočim sta zadnja dva byta za preverjanje pri čitanju -CRC. Zadnja vrzel, ki sledi temu polju ima 33 bytov.

Posebno funkcijo pri formatiranju imajo adresne oznake. Vse so enake dolžine 1 byta in nam služijo za identifikacijo začetka ID in podatkovnega polja ter za sinhronizacijo vezja pri serijsko paralelni pretvorbi. Adresne oznake razpoznamo po tem, da nam manjkajo nekateri urni biti (pri ostalih podatkovnih bytih so urni biti prisotni pri vseh bitnih celicah). Razlikujemo štiri adresne oznake, tri smo že opisali, četrta pa je adresna oznaka izbrisanih podatkov. Posamezne adresne oznake imajo naslednjo kodo:

Indeksna adresna oznaka:	1 1 1 1 1 1 0 0
	1 1 0 0 0 1 1 1
ID adresna oznaka :	1 1 1 1 1 1 1 0
	1 1 0 0 0 1 1 1
Podat. adres. ozn.:	1 1 1 1 1 0 1 1
	1 1 0 0 0 1 1 1
Adresna oz. izbris. podatkov:	1 1 1 1 1 0 0 0
	1 1 0 0 0 1 1 1



Slika 4. FORMAT SEKTORJA

V sektorskem formatu je 128 bytov od tega je na razpolago za uporabnika 128 bytov tj. 68%. Pri "trdem" sektoriranju nimamo predpisanega standarda, sektor pa omogoča shranjevanje večjega števila bytov. Seveda obstaja še več načinov formatiranja, večina od teh ima za cilj uporabniku povečati podatkovni prostor.

4. POGONSKA ENOTA DISKETE

Običajna pogonska enota vsebuje: čitalno/pisalno glavo, pogonski inehinazem, mehanizem za pozicioniranje sledi, čitalno/pisalno elektroniko, krmilno elektroniko ter sistem za vstavitve diskete v enoto. Vsi ti elementi omogočajo naslednja opravila: čiščenje in zapisovanje podatkov, nastavitve čitalno/pisalne glave na željeno sled in sektor ter generiranje in interpretiranje krmilnih signalov.

Čitalno/pisalna glava se razlikuje od glave, ki se uporablja pri običajnih čvrstih diskih. Le-ta "lebdi" nad diskom in omogoča zaradi zanemarljivo majhnega trenja visoke obrate. Takšno glavo je težko izdelati in vzdrževati. Čitalno/pisalna glava pri gibkih diskih je v neposrednem kontaktu z disketo in seveda podvržena trenju. Z namenom, da se poveča življenska doba glave in diskete ima večina pogonskih sistemov vgrajeno napravo za sprostitev pisalno/čitalno glavo v trenutkih, ko ne zahtevamo čitanja ali zapisovanja.

Vrtenje diskete omogoča ali sinhroni izmenični motor ali enosmerni motor s servo krmiljenjem. Konstantna hitrost (360 obr/min) je zelo pomembna; motor se stalno vrti ne glede na to ali je disketa v uporabi ali ne.

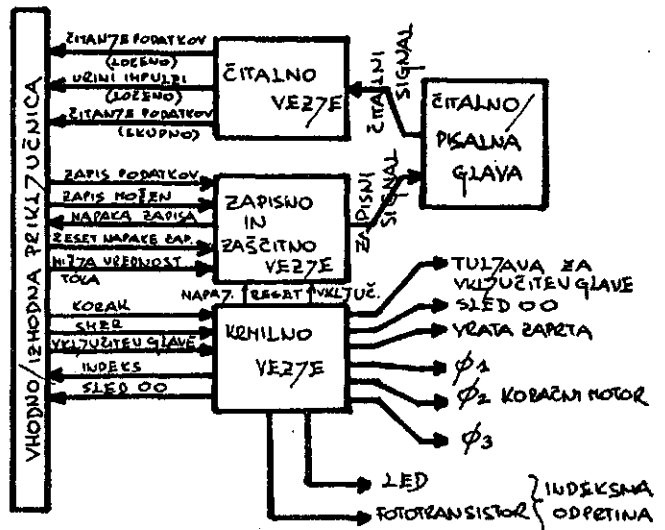
Za premikanje čitalno/pisalne glave uporabljajo v večini današnjih pogonskih enot koračne motorje, ki skupaj s posebnim vijajnim mehanizmom nastavljajo glavo na željeno sled (track). Za povečanje hitrosti nastavljanja čitalno/pisalne glave se uporabljajo tudi linearni motorji (npr. tvrdka Helitron). Funkcionalna blok shema značilne pogonske enote je podana na sliki 5.

Sistemi s "trdim" sektoriranjem imajo še eno izhodno linijo za zaznavo sektorjev. Premik glave na željeno sled izvršimo s pomočjo linije smer (step direction select) in s pomočjo impulzov na liniji korak (step). Vsak impulz na tej liniji omogoči premik glave na sosednjo sled. Na plošči s tiskanim vezjem je še pomembno vezje za demodulacijo frekvenčno moduliranih signalov tako, da lahko pošiljamo iz pogonske enote ločeno podatkovne in urine impulze. Pomembno je tudi zaščitno vezje, ki javi napako preko linije napaka zapisa.

Poleg izmeničnega napajanja potrebujemo še tri enosmerne napetosti: +5V, -5V in +24V.

Opisana pogonska enota je združevala v enem ohišju sistem z eno disketo. Na tržišču obstajajo seveda tudi ohišja z več disketnimi enotami in z različnim številom pogonskih sistemov. Sistem tvrdke Shugart SA902 se sestoji iz dveh neodvisnih mehanizmov za nastavljanje sledi (dva koračna motorja), dveh čitalno/pisalnih glav

in enega motorja za vrtenje diskete. Nekateri sistemi pogonskih enot so prirejeni tako, da jih lahko povežemo z več enakimi pogonskimi sistemi v obliki "radialne" povezave, drugi pa v serijski (daisy chain) povezavi. Seveda pa obstajajo tudi sistemi, ki dopuščajo oba načina povezav (npr. ORBIS 74).



Slika 5. FUNKCIONALNA SHEMA

5. VMESNIK IN KRMILJENJE POGONSKE ENOTE

Za povezavo pogonske enote gibkih diskov z gostiteljskim računalnikom potrebujemo posebno krmilno elektronsko vezje, ki ga upravljamo s pomočjo različnih podprogramov. Le-ti so namenjeni predvsem naslednjim opravilom: določanje adres sledi in sektorja željene datoteke, pozicioniranje na željeno sled, formatiranje, čitanje in zapisovanje podatkov ter detekcija in korekcija napak. Kompleksnost elektronskega vezja in programske podpore zavisi predvsem od namena uporabe diskete in posebnih zahtev uporabnika. V primerih, ko želimo, da je gostiteljski računalnik kar najmanj časa zaposlen z opravili okrog diskete, vpeljemo npr. dodatni mikro računalnik, ki prevzame vsa ta opravila. Dodatno programsko opremo moramo predvideti v primerih, ko moramo shraniti daljše podatkovne nize. Pri tem je pomembno, kako učinkovito porazdelimo večje bloke podatkov po disketi.

V zadnjem času se uporabljajo krmilna elektronska vezja v LSI tehniki v enem čipu. Taki "kontrolerji" so sorazmerno poceni in potrebujemo le malo dodatnih elektronskih komponent za povezavo pogonske enote diskete z gostiteljskim računalnikom (največkrat mikro računalnikom). V praksi dobro preizkušen je npr. kontroler tvrdke National Semiconductor Corp. INS 1771, ki je popolnoma zamenljiv s kontrolerjem FD1771 tvrdke Western Digital. Dobro uporabljiv je tudi kontroler Japonske firme NIPPON Electric Co. z oznako uPD372 D.

Poznani mikroročunalniški tvrdki Intel in Motorola sta tudi najavili svoj kontroler. Kontroler prvo omenjene tvrdke ima oznako 8271 in drugo omenjene MC6843. Imenovane disketne kontrolerje (Floppy Disk Controller) lahko priključimo na gostiteljski računalnik neposredno preko vhodno izhodnih neot ali v povezavi z DMA (Direct Memory Access) kontrolerjem (npr. Intelov 8257 in

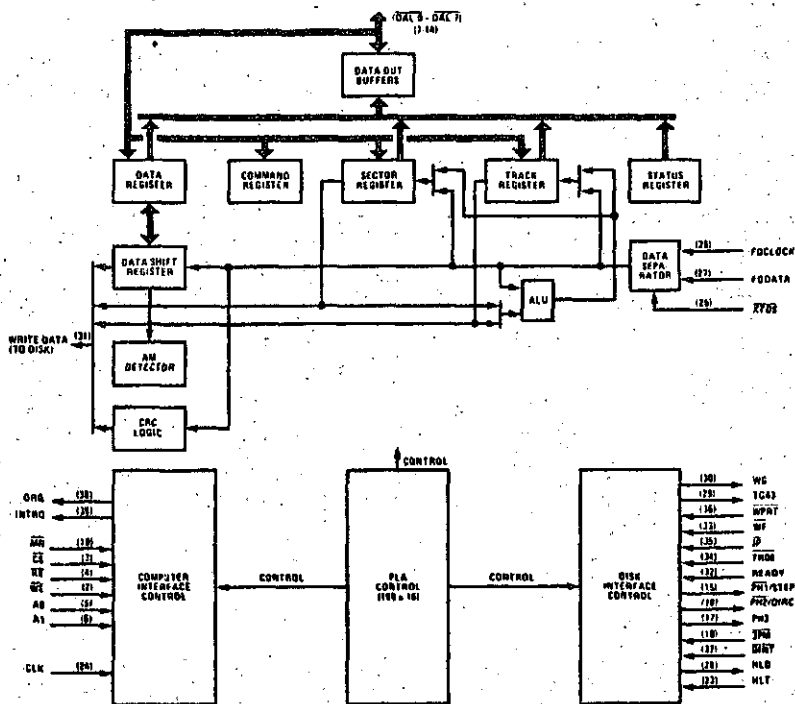
Motorolin MC6844 čip).

Navedimo nekaj lastnosti čipa INS1771:

Kontroler je v enem čipu (40-pin), omogoča "mehko" sektorsko formatiranje, ki je lahko združljivo z IBM 3740 standardom ali pa izbrano po želji uporabnika. Čip se programira s pomočjo sistemskih programov, pri tem se prenašajo vsi podatki, statusne informacije in krmilne besede preko 8 bitnih vodil. Omogoča avtomatsko pozicioniranje sledi z verifikacijo, čitanje in zapis s avtomatskim iskalcem sektorjev, začetni zapis sledi za inicializacijo, omogoča izbiro časa za prehod od sledi do sledi, izbiro časa vključitve glave, je TTL združljiv itd. Blok shema je prikazana na sliki 6.

6. ZAKLJUČEK

Namen pričujočega članka je predvsem seznaniti bralca z osnovnimi lastnostmi in značilnostmi gibkih diskov. Le-ti postajajo vedno bolj zanimivi v kombinaciji z mikro računalniki, tako za organizacije kot za posameznike. Detajlnim opisom elektronskih vezij in organizacije programov smo se izognili, ker bi vsebina preseгла okvir članka. Prav tako nismo omenili t.i.m. mini disket, ki se v zadnjem času zelo uveljavljajo. Vendar bi na tem mestu vseeno omenili nekaj značilnosti slednjih. Mini diskete (prvi so jih izdelali pri tvrdki Shugart) so manjše od standardnih disket in cenejše, zato so zlasti primerne za manjše in cenejše sisteme. Seveda pa je hitrost prenosa podatkov približno za polovico manjša od standardnih disket. Vzrok je predvsem v zmanjšani gostoti bitov in hitrosti vrtenja (300 obr/min). Celotna kapaciteta podatkov pa je manjša za približno eno tretjino. V članku je podana osnova za razumevanje in uporabo tega sorazmerno novega pomnilniškega medija.



Slika 6. BLOK SCHEMA ČIPA INS1771

LITERATURA

- 1) Users Manual; LSI Floppy Disk Controller Chip -uPD372D, NEC MICROCOMPUTERS, INC. (1977).
- 2) S.A. Caswell; Floppy disk Systems, Modern Data, August 1975.

- 3) D. Allen; A Minifloppy Interface, BYTE, February 1978.
- 4) M6800 Microprocessor Applications Manual, Motorola Inc., 1975.
- 5) I. Rampil; A Floppy Disk Tutorial, BYTE, December 1977.

računalniški model železniškega prometa na enotirni progi

j. eržen

UDK 681.3:656.2

PROMETNI INSTITUT ŽG LJUBLJANA

Opisana je simulacija železniškega prometa na enotirni progi, to je simulacija gibanja vlakov z upoštevanjem medsebojnega vpliva vlakov in vpliva motenj na SV napravah in v prometu.

TRAFFIC COMPUTER SIMULATION OF SINGLE TRACK RAILWAY: Digital computer simulation of railway traffic on a single-track railway is described, it involves a simulation of train movement and takes into account the interactive influence among trains and the influence of failures of signalling equipment and disturbance of traffic.

1. Uvod

Železniška vozila se od cestnih razlikujejo predvsem v dveh lastnostih. Prva je ta, da so vezana na svoj tir (tračnice). Zaradi tega se lahko železniška vozila srečujejo le na dvo-tirnih progah ali na postajah, kjer se enotirna proga razcepi v več tirov, ki so med seboj povezani s kretnicami, ki omogočajo prehajanje vozil s tira na tir in odrejaajo, na kateri tir se bo vozilo usmerilo.

Druga posebnost je dolga zavorna pot (1000 metrov) in počasno pospeševanje vlakov. Kjer strojevodja ne vidi 1000 metrov daleč, ga je treba obveščati o vsem, kar se dogaja na progi pred njim vsaj do razdalje 1000 metrov. Ta obvestila posredujejo signalno-varnostne naprave. Zaradi počasnega pospeševanja težkih vlakov (1000 ton) pa je potrebno vožnjo vlakov uravnati tako, da se ne ustavljajo ali zmanjšujejo hitrost po nepotrebnem, saj pri ponovnem pospeševanju porabijo veliko energije in čas potovanja se zelo podaljša.

V grobem ločimo dve grupi ljudi, ki skrbita za varnost in pravilno odvijanje železniškega prometa. Prvo grupo tvori osebje po postajah in dispečerskih centrih. Ti skrbijo za to, da je s signali ob progi enoveljavno definirana zasedenost proge in eventualno usmerjanje vlaka na drugi tir. Druga skupina ljudi je

vožno osebje na vlakih. Ti, upoštevajoč signale uravnavajo hitrost vlaka.

Obe skupini ljudi se danes na nekaterih železnicah že poslužujeta računalnikov, saj tako pri upravljanju svojega dela upoštevajo več dejstev, ki so važna za vodenje prometa. Poveča se še varnost prometa, ki postane bolj tekoč. Naloga, ki jo opravlja prva skupina, se imenuje vodenje prometa, druga skupina pa opravlja manj zahtevno delo - vodenje vlakov.

Računalniški sistemi, ki prevzemajo obe skupini nalog, so sestavni del informacijskega sistema za vodenje železniškega transporta. Tudi ti sistemi razbremenjujejo človeka, marsikje pa ga celo nadomestijo in spadajo med največje računalniške sistema sploh.

2. Namen modela

Na Prometnem inštitutu ŽG Ljubljana so bili izdelani programi za vodenje prometa na enotirni progi (4). Ker mora programski paket za vodenje prometa ob nastopu motenj na SV napravah ali ob drugih izjemnih situacijah tudi zagotoviti varnost prometa, običajno testiranje progamov ne zadostuje. Potrebno je še testiranje na modelu ali pa v povezavi s sistemom za daljinsko vodenje prometa na progi.

Za to variantno preizkušanja se nismo odločili zato, ker takega sistema v ŽG še nismo imeli, poleg tega pa bi morali še nemudoma nakupiti procesni računalnik. Ker procesnega računalnika nismo mogli kupiti, je odpadlo tudi testiranje na modelu proge z majhnimi vlakci. Imeli pa smo pogoje, da programe preizkusimo v povezavi z računalniškim (matematičnim) modelom, saj imamo na inštitutu zmogljiv računalnik IBM 360/30.

Programski paket za vodenje železniškega prometa je precej obsežen in je sestavljen iz več blokov, ki jih v grobem razdelimo v dve grupi: delovne programe, ki delajo odločitve in postavljajo vozne poti in organizacijske programe, ki nadzirajo tek delovnih programov in skrbijo za pravočasno prenašanje potrebnih podatkov z diskov v pomnilnik računalnika in narobe. Pri vodenju prometnega procesa se vsi programi prepletajo in vključujejo eden v drugega. Izhodni podatki enega bloka so v naslednjem trenutku vhodni podatki za nek drugi programski blok. Pri običajnem testiranju pripravljamo testne podatke za vsak programski blok posebej in vsak blok posebej se tudi testira. Tako ni mogoče nič reči o obnašanju programskega paketa kot celote, ne vemo, kako programski paket reagira na različne čakalne čase med posameznimi programi, niti nam ni znana dolžina teh čakalnih časov, kakršni nastopajo v praksi. Tudi pri še tako dobri izbiri testnih podatkih za posamične programe in bloke lahko zanesljivo trdimo, da nismo pokrili prav vseh možnosti, ki jih utesnimo srečati v praksi, tako torej ni preverjeno obnašanje programov prav v vseh situacijah.

Vse te dileme razčistimo s testiranjem programov za vodenje prometa v povezavi s simulacijskim modelom, kakršnega opisuje sestavek. Tak model ima še to prednost, da ga je mogoče hitro in enostavno dopolnjevati in spreminjati. Če se pojavijo kakšne spremembe na progi (novi tiri na postajah, prestavitve signalov, nova postajališča, ukinitve postaj) ali na vozilih (močnejše ali hitrejša lokomotive in hitrejši vagoni), podatke o novostih brez težav vnesemo v model za preizkušanje na nove razmere prilagojenih programov za vodenje prometa. Tudi izkušnje z dela na progi nas lahko spomnijo na nove rešitve, ki jim je brž treba prilagoditi programe za vodenje prometa, po vsaki spremembi pa je treba te programe spet testirati. Z novimi podatki dopolnjen model je tudi

v tem primeru prikladnejši od drugih.

Pomanjkljivost računalniškega modela je v tem da ne moremo simulirati z njim prevelikih sistemov, saj je potem model tako obširen, da mora teči na svojem računalniku. Ker je ta računalnik ponavadi oddaljen od računalnika, kjer tečejo programi za vodenje prometa, rabimo še dobre naprave za prenos podatkov med računalnikoma. Taka konfiguracija nam je pa redko na raspolago.

3. Organizacija modela in vhodni podatki

Programski paket je razdeljen na naslednje bloke:

- programe za pripravo vhodnih podatkov,
- simulacijski program,
- program za generiranje motenj v prometu,
- programe za generiranje javljanj.

a) Programe za pripravo vhodnih podatkov tvorijo:

- program za pripravo splošnih vhodnih podatkov,
- program za pripravo vhodnih podatkov o lokomotivah,
- program za pripravo vhodnih podatkov o objektih na odprti progi,
- program za pripravo vhodnih podatkov o objektih na postajah,
- program za pripravo vhodnih podatkov o uporabi proge.

b) Simulacijski program ima naslednje dele:

- blok za vpisovanje podatkov o voznem redu,
- blok za formiranje zloga podatkov v posameznih vlakih,
- blok za simulacijo gibanja vlaka,
- blok za določanje signalnih znakov,
- blok za simulacijo reagiranja na signal,
- blok za vpis podatkov ob prehodu vlaka na nov prostorni odsek in ob prihodu na postajo ali postajališče,
- blok za vpisovanje javljanj v začasno datoteko.

c) Program za generiranje motenj vsebuje generator naključnih števil in program za generiranje javljanj o motnjah ter program za vpis teh javljanj v določena polja.

d) Program za generiranje javljanj vsebuje programski blok za vpis javljanj v datoteko javljanj, od koder jih potem jemlje program

za vodenje prometa.

Vhodni podatki simulacijskega modela so:

- 1) Splošni podatki,
- 2) Podatki o upornosti proge,
- 3) Podatki o objektih na progi,
- 4) Podatki o lokomotivah,
- 5) Podatki o voznem redu.

1) Splošni podatki so:

- dolžina koraka,
- pojemki za različno hitrost in zavorno pot 1000 metrov,
- omejitve hitrosti,
- začetek proge,
- konec proge,
- vidna razdalja,
- začetek simulacije,
- omejitev hitrosti na kretnicah,
- poprečni čas postavljanja vozne poti (če je PVN v redu),
- poprečen čas obračanja smeri APB.

2) Podatki o upornosti proge so podatki o dolžini in specifični upornosti (Kp/mt) za elemente s konstantno specifično upornostjo.

3) Podatki o objektih na progi:

- Za vsako postajo imamo dane kilometraže (položaje, razdalje od Beograda) za
- kretnice na obeh straneh,
 - kretniške izolirke na obeh straneh,
 - uvozne in izvozne signale na obeh straneh,
 - postajnega poslopja,
 - števecv osi na obeh straneh.

4) Simulacijski program upošteva zaenkrat lokomotivi serije 342 in 362 in elektromotorno štiričlensko garnituro 311. Novih lokomotiv serije 363 še ni v programu, ker jih ob pripripi podatkov še ni bilo pri nas. Za vsako lokomotivo je podana:

- teža,
- dolžina,
- največja hitrost,
- podatki o pospeševanju (v odvisnosti od hitrosti, teže vlaka in specifičnega upora proge).

5) Podatki o vlakih so:

- številka vlaka,
- smer,
- največja hitrost,
- teža,
- serija lokomotive.

6) Podatki o voznom redu:

- odhodi in prihodi vlakov s posameznih postaj
- minimalni postanki na postajah,
- minimalni postanki na postajališčih,
- vožnje v odhodu.

4. Simulacija

Posamične naprave na progi so v simulaciji zajete kot sledi:

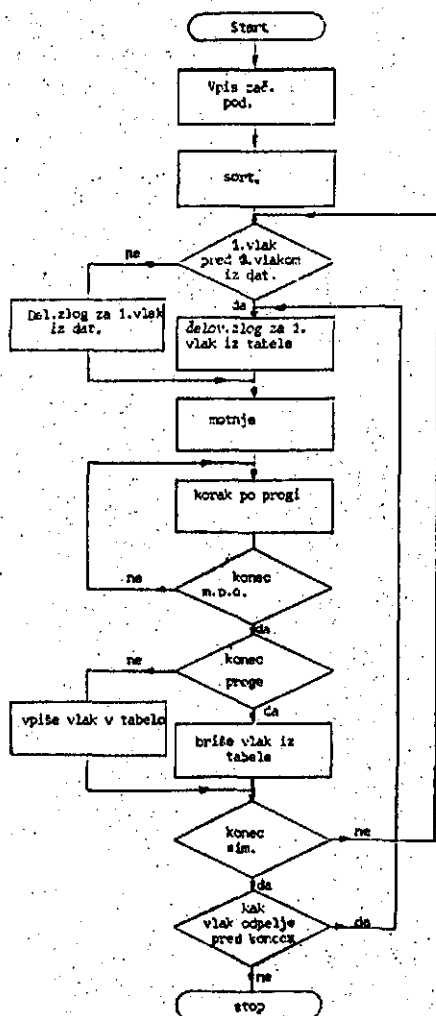
- postajna varnostna naprava: ustrezní program upošteva vse signalne znake, ki jih dajejo signali, vključeni v postajno varnostno napravo in simulira reagiranje vlakov na te signalne znake. Signalni znaki se postavijo z ozirom na zasedenost tirov in način vožnje vlakov (odklon ali premo) kot ga zahtevajo vozni redi in okoliščine;
- signalni znaki, ki jih kažejo signali APB-ja se določajo odvisno od zasedenosti prostornih odsekov in eventuelnih motenj na APB-ju. Simulira se reakcija vlaka na vsak signalni znak;
- javljalnik vlakovnih števil vpliva na simulacijo tako, da skrajša čase, potrebne za postavljanje vozne poti;
- radijske dispečerske zveze so zajete v simulacijskem programu tako, da se vlaku določi hitrost, ki bo prepeljala vlak pred signal takrat, ko mu ne bo treba pred signalom ustavljati;
- naprava za daljinsko vodenje prometa je upoštevana v simulacijskem programu predvsem s tem, da v primeru njene uporabe odpade sporazumevanje prometnikov med seboj (krajši čas pri obračanju APB-ja in podobno).

Simulacijski program omogoča, da simuliramo izpad ali nepopolno delovanje ene ali več naprav kot se to zgodi pri dejanskih motnjah.

Predno steče simulacija je treba posortirati datoteko s podatki o vlakih in voznom redu vlakov po časih odhoda vlakov z začetne postaje. Nato steče faza "polnjenja proge". Z ozirom na čas začetka simulacije se datoteka vozečih vlakov napolni z ustreznimi podatki, s katerimi se ponazori situacija na progi, kakršno zahteva vozni red ob času začetka simulacije.

En programski cikel predstavlja simulacija vožnje enega vlaka na odseku med postajama, na postaji se namreč vlaki lahko prehitijo in tako se zamenjata vlogi predvozečega (ovirajočega)

in sledečega (za varnost odgovornega) vlaka. Ciklus se začne s primerjavo datoteke "vozečih vlakov" (ki se zanje simulira vožnja) in datoteke vlakov, ki se zanje vožnja ni simulirala. Če se mora simulirati vožnje kakega vlaka iz druge datoteke predno se simulira vožnja kateregakoli vlaka iz prve datoteke, se zlog za ta vlak prenese (z ustreznimi dopolnitvami) v prvo datoteko in se simulira vožnja za ta vlak. Če pa ne, se simulacije izvede za vlak iz prvega zloga druge datoteke, tam so namreč zlogi urejeni po času začetka simulacije vožnje na medpostajnem odseku. Zlogi v datoteki "vozečih vlakov" se urejajo sproti, se pravi, ko je končana simulacija vožnje za nek vlak na medpostajnem odseku, se zlog za ta vlak vpiše v datoteko na mesto, kamor sodi glede na čas odhoda na naslednji, medpostajni odsek.



Grobi diagram poteka

Ko se konča simulacija vožnje na zadnjem odseku, se zlog za ta vlak briše iz datoteke "vozečih vlakov".

Vožnja vlaka na medpostajnem odseku se simulira po korakih. Korak je prirastek poti, na katerem smatramo, da se razni parametri ne spreminjajo. Korak je lahko dolg 25 ali 250 metrov. Kratak korak se uporablja, če je potrebna simulacija vožnje z manjšo hitrostjo ali pa večja natančnost zaradi simuliranja vožnje v bližini kritičnih mest (signal, števec, osi, kretnice, postajališče). Vsekakor takemu prirastku poti ustrezna časovni korak (prirastek) ne daljši od dveh sekund. Na dve sekundi se tudi vplisujejo javljanja, ki enolično definirajo situacijo na progi, v datoteko javljanj. Ta javljanja so izhodni podatki programa, gre za točno iste informacije, ki jih prenašamo pri daljinskem upravljanju prometa v komandni center.

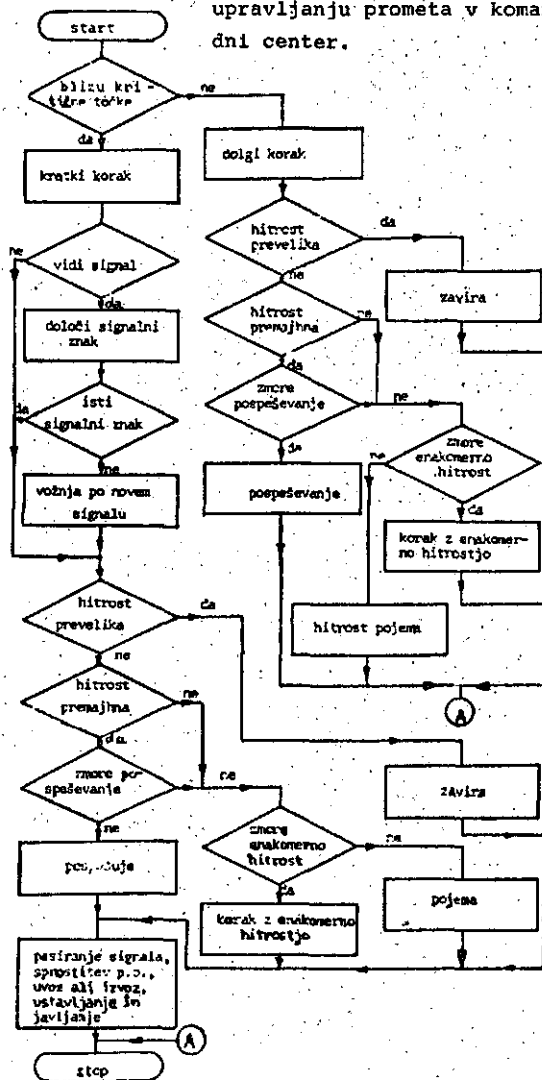


Diagram poteka za en korak vožnje

Na začetku vsakega cikla se vključi tudi generator motenj. Na osnovi izsledkov raziskav o zanesljivosti železniških SV naprav in varnosti železniškega prometa (2, 3) je bil napisan program, ki simulira nastop motenj (okvar in drugih izrednih situacij) tako pogosto, kot se to dogaja na progi v resnici. Program tako pripomore k res verni sliki realne situacije na progi.

5. Izhodni podatki

Izhodni podatki so javljanja s proge. Iz njih se da enolično določiti položaj vsakega vlaka ob danem času in stanje naprav na progi (morebitni nastop motenj). To so javljanja postajnih varnostnih naprav, javljanja APB in javljanja vlakovnih števil. Javljanja se izpisujejo vsaki dve sekundi, kar je pri voznem redu, ki je podan na pol minute natančno in s kakršnim operira tudi računalnik, dovolj. Javljanja za motnje izpisuje program na osnovi nastopa motenj, ki jih generira generator motenj. Druga javljanja pa se izpisujejo na osnovi gibanja vlakov.

Izpisujejo se tale javljanja:

- a) Vozeči vlaki in vodenje prometa generirajo tale javljanja:
- lege kretnic (odklon, prema, prestavljanje),
 - zasedenost kretnic (zasedena, prosta),
 - zasedenost tirnih izolirnih odsekov (zaseden, prost),
 - signalni znak na uvoznih signalih (samo "stoj" ali "vožnja"),
 - signalni znak na izvoznih signalih (samo "stoj" ali "vožnja"),
 - postavljanje vozne poti,
 - pritrditev vozne poti,
 - dajanje privolitve za APB,
 - smer APB.
- b) Generator motenj generira naslednja javljanja:
- motnje na postavljalnem mehanizmu kretnic,
 - motnje na kretniških izolirkah,
 - motnje na signalih (pregorele žarnice),
 - motnje na napajalnih napravah SV naprav.

6. Zaključek

Programi za vodenje prometa imajo blok programov, ki obdelujejo javljanja naprav s proge in iz njih dobiva potrebne informacije za vodenje

prometnega procesa. Informacije grobo razdelimo v dve skupini:

a) Informacije o gibanju vlaka so v bistvu natančno določeni vozni časi vlakov ob upoštevanju karakteristik proge in vlaka, slučajnih vplivov in medsebojnih vplivov vlakov. Vhodni podatki za testiranje brez modela so bila odstopanja od fiksnega voznega reda, ki so bila neodvisne slučajne spremenljivke, odvisnosti namreč ni bilo mogoče podati. Tako ni bila upoštevana odvisnost vlakov med seboj.

b) Tudi podatki o stanju proge so bili pri testiranju brez modela neodvisne slučajne spremenljivke in tudi od njih niso bila odvisna odstopanja od voznega reda.

Testiranje na modelu bo omogočilo, da se programi testirajo ob upoštevanju vseh odvisnosti.

Druge potrebne informacije za vodenje prometa so konstantne in so shranjene v raznih datotekah.

Poleg testiranja obstoječih programov bi nam model, če bi v prihodnosti imeli na izbiro več programskih sistemov za vodenje prometa, pomagal izbrati najboljšega. Lahko pa bomo tudi z njim preverjali učinkovitost različnih SV naprav, kar bo v veliko pomoč pri raznih nakupih.

L i t e r a t u r a

- Eržen Janez, Vodenje železniškega prometa z računalniki in zanesljivost sodobnih signalno-varnostnih naprav (magistrsko delo), Ljubljana, Fakulteta za elektrotehniko, 1977.
- Eržen Janez, Jelaska Miroslav, Uporaba računskih sistemov za vodenje prometa na odsekih magistralnih prog, ki so opremljene s sodobnimi signalno-varnostnimi napravami
- Eržen Janez, Jelaska Miroslav, Gyergyek Ludvik Uporaba računalniških sistemov za vodenje prometa na odsekih magistralnih prog, ki so opremljene s sodobnimi signalno-varnostnimi napravami II.faza., ŽG Ljubljana, Prometni inštitut 1974
- Jelaska Miroslav, Computer elaboration of time table for single railway line. Proceeding 7 th IFIP Conference, Nice, September 8-12, 1975. Springer Vlg. 1976. Pt.1.

audio kasetofon kot cenena vhodno-izhodna enota za mikro računalnike

UDK 681.327.64

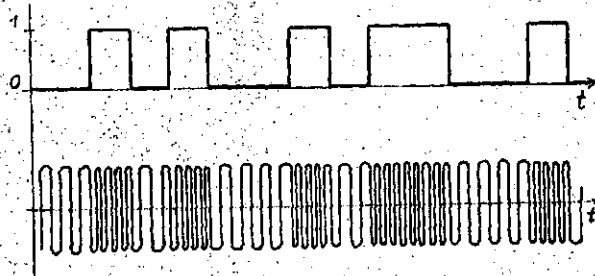
a.hadži
m.kovačevič
a.p.železnikar

Odsek za računalništvo in informatiko
Institut "Jožef Stefan"

Povzemamo izkušnje, ki smo jih dobili pri izdelavi kasetofske pomnilniške enote za mikroprocesorski sistem 6800.

APPLICATION OF AN AUDIO CASSETTE RECORDER FOR A MICROCOMPUTER I/O UNIT- Some Useful hints experienced while designing a cassette recorder peripheral for the 6800 system.

Dejstvo je, da je cenenost, enostavnost in hitra možnost nabave potrebnega materiala odločujoč faktor pri načrtovanju sistemov, ki jih uporabljamo pri raziskovalnem delu ali pri amaterski gradnji. Uporaba cenene audio kasetofona kot pomnilniške enote za mikro računalnike vsekakor ustreza zahtevam. Na navadni kaseti (2x30 min.) lahko shranimo 1 M bit informacije. Hitrost prenosa pa je 300 bit/sek. Odločili smo se za dvotonski zapis. Logično "1" predstavlja ton s frekvenco približno 5 KHz, logično "0" pa za polovico nižja frekvenca.



Slika 1. Oblika zapisa

Frekvenca in hitrost zapisa sta dana z lastnostmi kasetofona. Najvišja frekvenca, ki jo sprejme kasetofon je 5KHz. Če zahtevamo, da je en bit opisan s približno 10 nihaji (to nam določa dolžina bita na 10 % natančno) pridemo do hitrosti prenosa 300 bit/sek \sim 500 boudov.

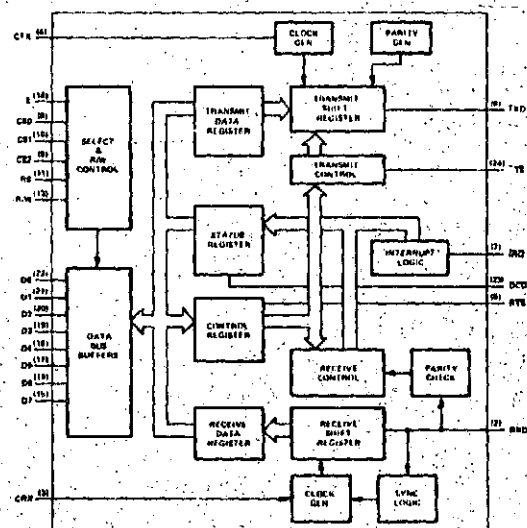
Pozorni moramo biti na izzvenevanje logične "1" pri zapisovanju in pri branju. Saj se nam lahko podaljša tudi za periodo modulacijskega vala, kar prinese v našem primeru 10 % napako v dolžini bita.

Vmesnik smo priključili na mikro procesor preko ACIAe (Asynchronous Communication Interface Adapter). ACIA pretvarja 8 kanalov paralelne informacije v serijsko in obratno.

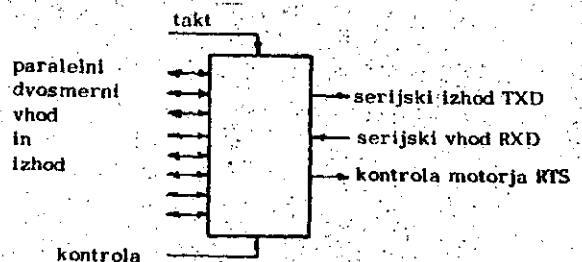
Za mikro procesor pa predstavlja celico z dvema nastavoma, v kateri lahko pišemo in beremo. Prva celica je statusni register, druga pa podatkovni.

Z vpisom v statusni register določimo obliko serijske besede (št. bitov, stop bitov, parnost), signal RTS, ki ga v našem primeru uporabimo za kontrolo motorja in še nekatere parametre. Iz njega pa prečitamo pripravljenost za čitanje ali spreminjanje besede. ACIA ima dvojen spomin za oddajno in sprejemno besedo, tako lahko med

časom oddajanja ali sprejemanja ene besede ($t \sim 35$ ms v našem primeru), centralni procesor zaposlimo z drugimi opravili.

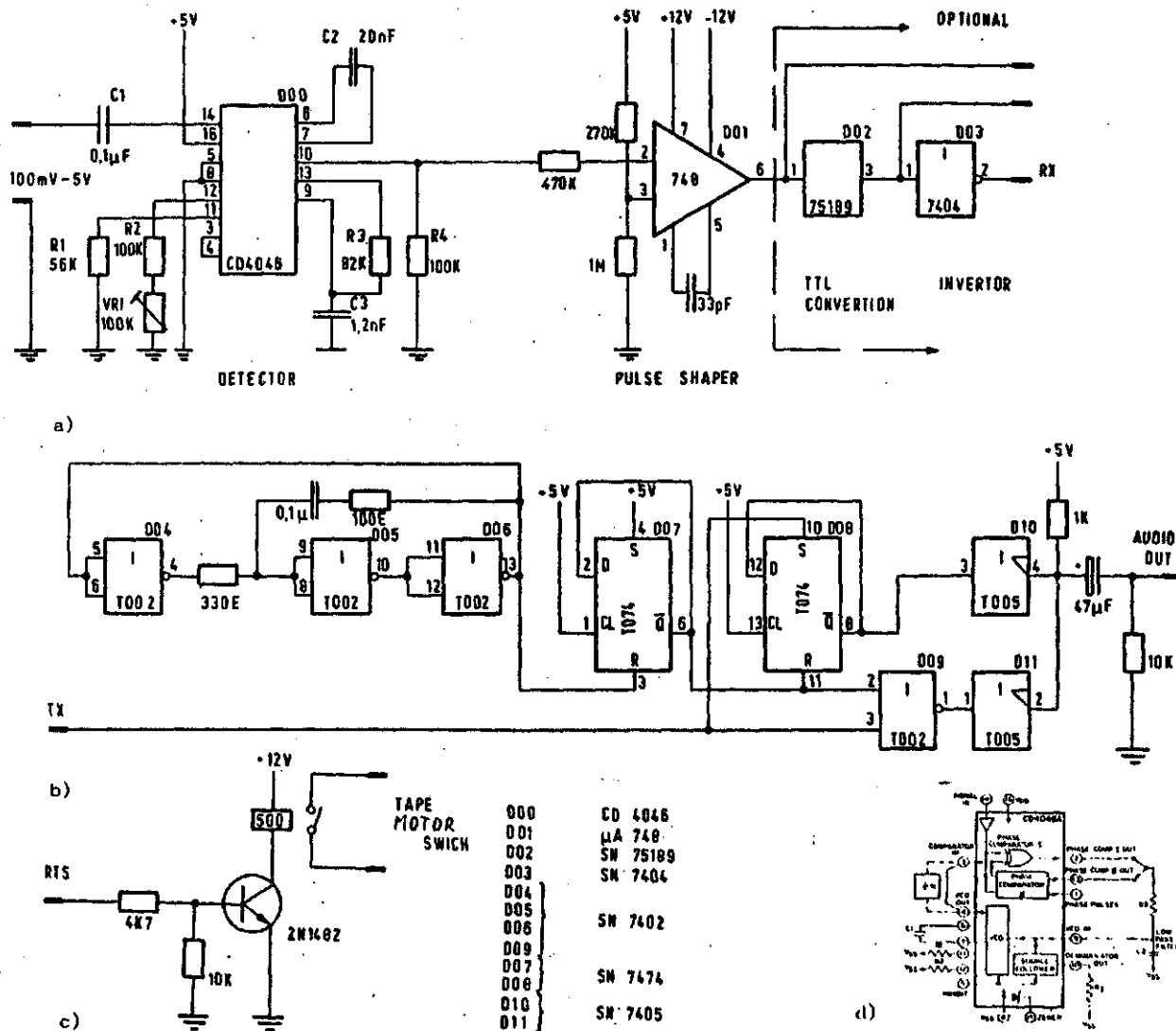


Slika 2. Blok diagram ACIA 56850



sl. 3. Blok diagram vezave ACIA

Vmesnik ima tri neodvisne dele: oddajnik, sprejemnik in kontrolo motorja. Osnova oddajnika sta multivibratorja s frekvenco 5 KHz in 2.5 KHz. Frekvenca na izhodu se izmenjuje glede na logično "1" ali "0" na vhodu.



Slika 4. Shema dvotonskega vmesnika za audio kasetofon.

a) Sprejemnik. Frekvenco prehoda določa VR1.
b) Oddajnik

c) Kontrola motorja
d) Blok shema fazno vkleščene zanke CD 4046

Sprejemnik je zasnovan na fazno vkleščeni zanki, ki višji frekvenci priredi logično "1" nižji pa logično "0". S spreminjanjem elementov, ki določajo frekvenco prehoda, lahko uporabimo sprejemnik za dekodiranje poljubnega dvotonskega zapisa. Za obliko izpisovanja smo izbrali standardno Motorola obliko:

S1, aa, bbbb, d β , d1, ..., d β , cs, β D, β A
:
:
S9

S1 začetek zapisa
aa število besed v zapisu
bbbb začetni naslov
d β , d1, ... d β podatki
cs kontrolna suma
 β D, β A konec zapisa in nova vrstica
S9 konec datoteke

Celotno datoteko razbijemo v zapise s 16 besedami. Uporabili smo 7 bitno ASCII kodo. Takšna oblika zapisovanja še zdaleč ni optimalna, kar se tiče hitrosti, enostavnosti in zanesljivosti prenosa, je pa zato kompatibilna s standardnimi normami zapisovanja.

Programski paket za shranjevanje podatkov iz spomina na kaseto in obratno TAPE MONITOR je kompatibilen z osnovnim orskim programom MIKBUG in je vanj vključen kot podprogram. Tvorita ga dva neodvisna dela: čitalnik in pisalnik (TLOAD in TPUNCH).

Razvili smo tudi preprost program za urejanje datoteke po kaseti. PNAME izpiše pred vsako datoteko ime v obliki:

NM, xy Ime iz poljubnih dveh znakov
x, y FNAME pa pošče izbrano ime.
Bolj praktično je, če so posamezne datoteke na različnih kasetah.


```

26 F4      BRA    TLOAD3
*
FF 05 06  TLOAD4 STX    ENDR    STORE LAST ADDRESS
*
7C 05 00      INC    CKSM    CHECKSUM CONTROL
27 C2      BEQ    TLOAD1
92 3F 01      LDA  A  #7     ERROR
8D FD 0F      JSR    OUTCH   PRINT !?:
*
8D 06 0C  TLOAD5 JSR    TSTOP  STOP TAPE MOTOR
*
CE 05 04      LDY    #BEGA  WRITE FIRST ADDRESS ON MONITOR
8D FD 1C      JSR    OUT2H
8D FD 1C      JSR    OUT2H
8D FD 20      JSR    OUTS    WRITE BLANK
CE 05 06      LDY    #ENDR   WRITE LAST ADDRESS
8D FD 1C      JSR    OUT2H
8D FD 1C      JSR    OUT2H
*
7E FD 4F      JMP    CONTR
*
*TBADDR BUILDS 16 BIT ADDRESS FROM TWO SUBSEQUENT 8 BIT ENTRIES
*
8D 09      TBADDR BSR    BYTE
87 05 02      STA  B  XHI    STORE LEFT SIDE OF ADDRESS
8D 04      BSR    BYTE
87 05 03      STA  A  XLOW   STORE RIGHT SIDE OF ADDRESS
8D      RTS
*
*BYTE ADDS TWO SUBSEQUENT HEX ENTRIES INTO 8 BIT BYTE
*
8D 05 1A  BYTE JSR    INHEX  ENTER FIRST LOWER HALF BYTE
48      ASL  A
48      ASL  A      SHIFT INTO HIGH POSITION
48      ASL  A
48      ASL  A
16      TAB
8D 05 1A  JSR    INHEX  ENTER SECOND LOWER HALF BYTE
*
84 04      AND  A  #FF    CLEAR LEFT SIDE OF ACC.
16      ORA  A      ADD ACC. A AND ACC. B
*
16      TAB    CREATE CHECK SUM
FE 05 00  ADD  B  CKSM  ADD NEW BYTE TO CHECK SUM
F7 05 00  STA  B  CKSM  STORE NEW CHECK SUM
8D      RTS
*
*
*
*
*
*
*

```

```

* OUTPUT CR AND LF
*
86 00  TCR LF  LORA  #100
8D 21      BSR    TOUT
86 00      LORA  #F0A
8D 1D      BSR    TOUT
4F      CLRA
8D 1A      BSR    TOUT.
8D 1A      BSR    TOUT
39      RTS
*
*OUT2H OUTPUT DATA INDICATED BY INDEX REG.
* IN TWO ASCII CODED HEX. CHARACTERS
*
86 00  TOUT2H LDA  A  X      OUTPUT HIGHER HALF BYTE
8D 05      BSR    OUTHL
86 00      LDA  A  X      OUTPUT LOWER HALF BYTE
8D 05      BSR    OUTHR
39      RTS
*
*OUTHL OUTPUT HIGHER HALF BYTE CONTAINED IN ACC. A
*
44      OUTHL LSR  A      SHIFT INTO LOWER HALF OF ACC. A
44      LSR  A
44      LSR  A
44      LSR  A
*
*OUTHR OUTPUT LOWER HALF BYTE CONTAINED IN ACC. A
*
84 0F  OUTHR AND  A  #FF    CLEAR LEFT SIDE OF ACC.
88 30      ADD  A  #F20   CHANGE INTO ASCII CODE
81 3A      CMP  A  #F20
23 02      BLS  TOUT   0 - 9 EXIT
88 07      ADD  A  #F7    A - F EXIT
*
* TOUT WRITES CONTENTS OF ACC. A ON TAPE
*
* OUTPUT ACCA VIA ACIA
37      TOUT  PSH  B
FE F8 0E  LOOP  LDA  B  ACIADS ACIA CONTROL REG. CHECK
57      BSR  B      WAIT UNTIL TRANSMIT DATA REG. (BIT1) IS SET
57      BSR  B
24 F9      BCC  LOOP
*
87 FD 0F      STA  A  ACIADA WRITE DATA
*
33      PUL  B
39      RTS
*
*
*
*
*
*
*

```

54


```

*PUNCH WRITES DATA ON TAPE IN ASCII CODE IN FORM
* S1, AA, BBBB, DD, D1, ..., DF, CS
* S1
* S9
*
* FILE IS BROKEN INTO 16 DATA BYTE RECORDS
* S1 - START OF RECORD
* S9 - END OF FILE
* AA - NUMBER OF BYTES IN RECORD
* BBBB - ADDRESS OF FIRST DATA IN RECORD
* DD, DN - DATA
* CS - CHECK SUM
*RADDR BUILDS 16 BIT ADDRESS FROM TWO SUBSEQUENT 8 BIT ENTRIES VI
*
8D FD 28 TPUNCH JSR     OUTFS  WRITE BLANK
8D FC 7C         JSR     RADDR  READ FIRST ADDRESS
FF 05 04         STX     BEGA    STORE FIRST ADDRESS
FF 05 08         STX     TN      ENTER FIRST ADDRESS
*
8D FD 28         JSR     OUTFS  WRITE BLANK
8D FC 7C         JSR     RADDR  READ LAST ADDRESS
8D FD 28         JSR     OUTFS  WRITE BLANK
FF 05 06         STX     ENDA    STORE LAST ADDRESS
*
8D 06 01         JSR     FNAME   WRITE FILE NAME
*
8E 05 07 TP1    LDA  A  ENDA+1
8D 05 09         SUB  A  TN+1   CALCULATE NUMBER OF BYTES IN FILE STILL LEFT
F6 05 06         LDA  B  ENDA
F2 05 08         SBC  B  TN      TN IS FIRST ADDRESS OF NEW RECORD
2E 04           BNE  TP2    MORE THAN 16 BYTES LEFT
81 10           CMP  A  #16
25 02           BCS  TP2    LESS THAN 16 DATA BYTES LEFT
8E 0F           LDA  A  #5
8D 04           ADD  A  #4
87 05 08         STR  A  #COUNT NUMBER OF ALL BYTES IN RECORD
8D 03           SUB  A  #2
87 05 0A         STR  A  TEMP  NUM. OF DATA BYTES + CHECK SUM BYTE
*
8E 03           LDA  A  #5
8D 05 0C         JSR     TOUT
8E 31           LDA  A  #1
8D 05 0C         JSR     TOUT
*
7F 05 08         CLR  CKSM  CLEAR CHECK SUM
* ALL FOLLOWING EXITS THROUGH PUNT2
* ARE ADDED TO CHECK SUM
CE 05 08         LDX  #ACONT
8D 2F           BSR  PUNT2  WRITE NUMBER OF BYTES IN RECORD
*
CE 05 08         LDX  #TN    WRITE FIRST ADDRESS OF DATA
8D 3A           BSR  PUNT2
CE 05 09         LDX  #TN+1
8D 25           BSR  PUNT2
    
```

```

*
FE 05 08         LDX  TN    SET INDEX REG. ON FIRST ADDRESS
8D 30           BSR  PUNT2  WRITE DATA
08
7A 05 0A         DEC  TEMP  COUNT DATA BYTES
2E F8           BNE  TP4
*
FF 05 08         STX  TN    STORE FIRST ADDRESS OF NEXT RECORD
*
73 05 0A         COM  CKSM  COMPLEMENT CHECK SUM
CE 05 08         LDX  #CKSM
8D 05 05         JSR  TOUT2H WRITE CHECK SUM
8D 05 07         JSR  TCRLF  OUTPUT CR AND LF
*
FE 05 08         LDX  TN    CHECK IF LAST ADDRESS HAS BEEN WRITTEN
8D 09           DEX
8C 05 0E         CPY  ENDA
2E A2           BNE  TP1    IF NOT CONTINUE
**
8E 03           LDA  A  #5
8D 05 0C         JSR  TOUT
8E 39           LDA  A  #9
8D 05 0C         JSR  TOUT
8D 05 07         JSR  TCRLF  OUTPUT CR AND LF
*
8D 2C           BSR  TSTOP  STOP TAPE MOTOR
7E FD 4F         JMP  CONTR
*
F6 05 0A PUNT2  LDA  B  CKSM  MAINTAIN CHECK SUM
E8 0A           ADD  B  X
F7 05 0A         STR  B  CKSM  STORE CHECK SUM
7E 05 05         JMP  TOUT2H WRITE DATA
*
*TSTART RESETS ACIA AND ENGAGES TAPE MOTOR
* WAIT LOOP IS REQUIRED FOR MOTOR ACCELERATION
*
36 TSTART  PSH  A
8E 03           LDA  A  #A03  ACIA MASTER RESET
87 F8 0E         STR  A  ACIACS
8E 02           LDA  A  #A02  ACIA INIC. AND START MOTOR
87 F8 0E         STR  A  ACIACS
32 PUL  A
*
*WAIT TIME DELAY LOOP
*
CE FF FF WAIT  LDX  #FFFFF TAPE MOTOR ACCELERATION TIME DELAY
* NUMBER OF CYCLES IS SET BY INDEX REG.
* 39 MPU CYCLE LOOP
8D 00           DEY
8F 00           TST  X
8D 00           TST  X
8D 00           TST  X
8C 00 00         CPY  #FFFF
2E F2           BNE  WAIT1
    
```

55

```

39          RTS
40 STOP STOPS TAPE MOTOR
41 STOP RST JUMP TO WAIT LOOP
42          *
43          LDR A #882  STOP TAPE MOTOR
44          STA A #1100
45          RTS
46          *
47          *NAME WRITES NAME OF FILE ON TAPE IN FORM
48          *      NM XY
49          *      XY - SELECTED NAME
50          *
51          *
52          *
53          *
54          *
55          *
56          *
57          *
58          *
59          *
60          *
61          *
62          *
63          *
64          *
65          *
66          *
67          *
68          *
69          *
70          *
71          *
72          *
73          *
74          *
75          *
76          *
77          *
78          *
79          *
80          *
81          *
82          *
83          *
84          *
85          *
86          *
87          *
88          *
89          *
90          *
91          *
92          *
93          *
94          *
95          *
96          *
97          *
98          *
99          *
100         *
101         *
102         *
103         *
104         *
105         *
106         *
107         *
108         *
109         *
110         *
111         *
112         *
113         *
114         *
115         *
116         *
117         *
118         *
119         *
120         *
121         *
122         *
123         *
124         *
125         *
126         *
127         *
128         *
129         *
130         *
131         *
132         *
133         *
134         *
135         *
136         *
137         *
138         *
139         *
140         *
141         *
142         *
143         *
144         *
145         *
146         *
147         *
148         *
149         *
150         *
151         *
152         *
153         *
154         *
155         *
156         *
157         *
158         *
159         *
160         *
161         *
162         *
163         *
164         *
165         *
166         *
167         *
168         *
169         *
170         *
171         *
172         *
173         *
174         *
175         *
176         *
177         *
178         *
179         *
180         *
181         *
182         *
183         *
184         *
185         *
186         *
187         *
188         *
189         *
190         *
191         *
192         *
193         *
194         *
195         *
196         *
197         *
198         *
199         *
200         *
201         *
202         *
203         *
204         *
205         *
206         *
207         *
208         *
209         *
210         *
211         *
212         *
213         *
214         *
215         *
216         *
217         *
218         *
219         *
220         *
221         *
222         *
223         *
224         *
225         *
226         *
227         *
228         *
229         *
230         *
231         *
232         *
233         *
234         *
235         *
236         *
237         *
238         *
239         *
240         *
241         *
242         *
243         *
244         *
245         *
246         *
247         *
248         *
249         *
250         *
251         *
252         *
253         *
254         *
255         *
256         *
257         *
258         *
259         *
260         *
261         *
262         *
263         *
264         *
265         *
266         *
267         *
268         *
269         *
270         *
271         *
272         *
273         *
274         *
275         *
276         *
277         *
278         *
279         *
280         *
281         *
282         *
283         *
284         *
285         *
286         *
287         *
288         *
289         *
290         *
291         *
292         *
293         *
294         *
295         *
296         *
297         *
298         *
299         *
300         *
301         *
302         *
303         *
304         *
305         *
306         *
307         *
308         *
309         *
310         *
311         *
312         *
313         *
314         *
315         *
316         *
317         *
318         *
319         *
320         *
321         *
322         *
323         *
324         *
325         *
326         *
327         *
328         *
329         *
330         *
331         *
332         *
333         *
334         *
335         *
336         *
337         *
338         *
339         *
340         *
341         *
342         *
343         *
344         *
345         *
346         *
347         *
348         *
349         *
350         *
351         *
352         *
353         *
354         *
355         *
356         *
357         *
358         *
359         *
360         *
361         *
362         *
363         *
364         *
365         *
366         *
367         *
368         *
369         *
370         *
371         *
372         *
373         *
374         *
375         *
376         *
377         *
378         *
379         *
380         *
381         *
382         *
383         *
384         *
385         *
386         *
387         *
388         *
389         *
390         *
391         *
392         *
393         *
394         *
395         *
396         *
397         *
398         *
399         *
400         *
401         *
402         *
403         *
404         *
405         *
406         *
407         *
408         *
409         *
410         *
411         *
412         *
413         *
414         *
415         *
416         *
417         *
418         *
419         *
420         *
421         *
422         *
423         *
424         *
425         *
426         *
427         *
428         *
429         *
430         *
431         *
432         *
433         *
434         *
435         *
436         *
437         *
438         *
439         *
440         *
441         *
442         *
443         *
444         *
445         *
446         *
447         *
448         *
449         *
450         *
451         *
452         *
453         *
454         *
455         *
456         *
457         *
458         *
459         *
460         *
461         *
462         *
463         *
464         *
465         *
466         *
467         *
468         *
469         *
470         *
471         *
472         *
473         *
474         *
475         *
476         *
477         *
478         *
479         *
480         *
481         *
482         *
483         *
484         *
485         *
486         *
487         *
488         *
489         *
490         *
491         *
492         *
493         *
494         *
495         *
496         *
497         *
498         *
499         *
500         *
501         *
502         *
503         *
504         *
505         *
506         *
507         *
508         *
509         *
510         *
511         *
512         *
513         *
514         *
515         *
516         *
517         *
518         *
519         *
520         *
521         *
522         *
523         *
524         *
525         *
526         *
527         *
528         *
529         *
530         *
531         *
532         *
533         *
534         *
535         *
536         *
537         *
538         *
539         *
540         *
541         *
542         *
543         *
544         *
545         *
546         *
547         *
548         *
549         *
550         *
551         *
552         *
553         *
554         *
555         *
556         *
557         *
558         *
559         *
560         *
561         *
562         *
563         *
564         *
565         *
566         *
567         *
568         *
569         *
570         *
571         *
572         *
573         *
574         *
575         *
576         *
577         *
578         *
579         *
580         *
581         *
582         *
583         *
584         *
585         *
586         *
587         *
588         *
589         *
590         *
591         *
592         *
593         *
594         *
595         *
596         *
597         *
598         *
599         *
600         *
601         *
602         *
603         *
604         *
605         *
606         *
607         *
608         *
609         *
610         *
611         *
612         *
613         *
614         *
615         *
616         *
617         *
618         *
619         *
620         *
621         *
622         *
623         *
624         *
625         *
626         *
627         *
628         *
629         *
630         *
631         *
632         *
633         *
634         *
635         *
636         *
637         *
638         *
639         *
640         *
641         *
642         *
643         *
644         *
645         *
646         *
647         *
648         *
649         *
650         *
651         *
652         *
653         *
654         *
655         *
656         *
657         *
658         *
659         *
660         *
661         *
662         *
663         *
664         *
665         *
666         *
667         *
668         *
669         *
670         *
671         *
672         *
673         *
674         *
675         *
676         *
677         *
678         *
679         *
680         *
681         *
682         *
683         *
684         *
685         *
686         *
687         *
688         *
689         *
690         *
691         *
692         *
693         *
694         *
695         *
696         *
697         *
698         *
699         *
700         *
701         *
702         *
703         *
704         *
705         *
706         *
707         *
708         *
709         *
710         *
711         *
712         *
713         *
714         *
715         *
716         *
717         *
718         *
719         *
720         *
721         *
722         *
723         *
724         *
725         *
726         *
727         *
728         *
729         *
730         *
731         *
732         *
733         *
734         *
735         *
736         *
737         *
738         *
739         *
740         *
741         *
742         *
743         *
744         *
745         *
746         *
747         *
748         *
749         *
750         *
751         *
752         *
753         *
754         *
755         *
756         *
757         *
758         *
759         *
760         *
761         *
762         *
763         *
764         *
765         *
766         *
767         *
768         *
769         *
770         *
771         *
772         *
773         *
774         *
775         *
776         *
777         *
778         *
779         *
780         *
781         *
782         *
783         *
784         *
785         *
786         *
787         *
788         *
789         *
790         *
791         *
792         *
793         *
794         *
795         *
796         *
797         *
798         *
799         *
800         *
801         *
802         *
803         *
804         *
805         *
806         *
807         *
808         *
809         *
810         *
811         *
812         *
813         *
814         *
815         *
816         *
817         *
818         *
819         *
820         *
821         *
822         *
823         *
824         *
825         *
826         *
827         *
828         *
829         *
830         *
831         *
832         *
833         *
834         *
835         *
836         *
837         *
838         *
839         *
840         *
841         *
842         *
843         *
844         *
845         *
846         *
847         *
848         *
849         *
850         *
851         *
852         *
853         *
854         *
855         *
856         *
857         *
858         *
859         *
860         *
861         *
862         *
863         *
864         *
865         *
866         *
867         *
868         *
869         *
870         *
871         *
872         *
873         *
874         *
875         *
876         *
877         *
878         *
879         *
880         *
881         *
882         *
883         *
884         *
885         *
886         *
887         *
888         *
889         *
890         *
891         *
892         *
893         *
894         *
895         *
896         *
897         *
898         *
899         *
900         *
901         *
902         *
903         *
904         *
905         *
906         *
907         *
908         *
909         *
910         *
911         *
912         *
913         *
914         *
915         *
916         *
917         *
918         *
919         *
920         *
921         *
922         *
923         *
924         *
925         *
926         *
927         *
928         *
929         *
930         *
931         *
932         *
933         *
934         *
935         *
936         *
937         *
938         *
939         *
940         *
941         *
942         *
943         *
944         *
945         *
946         *
947         *
948         *
949         *
950         *
951         *
952         *
953         *
954         *
955         *
956         *
957         *
958         *
959         *
960         *
961         *
962         *
963         *
964         *
965         *
966         *
967         *
968         *
969         *
970         *
971         *
972         *
973         *
974         *
975         *
976         *
977         *
978         *
979         *
980         *
981         *
982         *
983         *
984         *
985         *
986         *
987         *
988         *
989         *
990         *
991         *
992         *
993         *
994         *
995         *
996         *
997         *
998         *
999         *
1000        *

```

domaća proizvodnja računarske opreme i opreme za prenos podataka

a. jerman-blazič

UDK 681.3

Republički komite
za društveno pla-
niranje in informa-
cijski sistem,
Ljubljana

Autor daje kraći prikaz stanja u SFR Jugoslaviji na području priprema i razvoja domaće proizvodnje računarske opreme. Na osnovu ocene stanja i mogućnosti proizvođača, te sadašnjih i budućih potreba informacionih sistema posebno u okviru razvoja integriranih informacionih sistema republika, pokrajina i federacije dati su odgovori: šta i kako proizvoditi te kako organizirati drugačiji razvoj i nezavisnu proizvodnju računarske opreme u SFR Jugoslaviji.

PRODUCTION OF COMPUTER AND DATA COMMUNICATION EQUIPMENT IN YUGOSLAVIA - The paper is a brief survey of Yugoslav computer technology and production trends.

Domaća proizvodnja računarske opreme i opreme za prenos podataka.

(Dani Informatike, Novi Sad, 11.4.1978, saopštenje)

U dogovoru o osnovama društvenog plana SFR Jugoslavije za period 1976-1980 su prihvaćeni zadaci na području razvoja i proizvodnje računarske opreme. Većina radnih organizacija elektronske industrije je do pre kratkog vremena bila u zaostanku sa svojim obavezama. Što više, tokom dvogodišnje zabrane i sadašnje verifikacije uvoza računarske opreme te iste organizacije nisu ponudile tržištu niti elemente jednostavnije aparaturne opreme, niti kompleksnije programske opreme. Pored uzroka, kao što je nesposobnost potencijalnih proizvođača da sklope dogovor je za to bila kriva i neelastičnost i neprilagođenost naših propisa o proizvodnoj kooperaciji sa inostranim partnerima što je imalo za posledicu nedovoljnu spremnost renomiranih inostranih firmi za ravnopravnu podelu rada, prava i obaveza. Ovu nespremnost inostranih proizvođača računarske opreme za kooperaciju je bila delimično motivisana i nedoslednošću i brojnim izuzecima pri sprovođenju zabrane odnosno verifikacije uvoza.

Pregled stanja priprema za razvoj i proizvodnju računarske opreme u SFR Jugoslaviji je uprkos tome pokazao da imamo trenutno evidentiranih 10 potencijalnih domaćih proizvođača (ISKRA, Elektrotehna, Gorenje, Digitron, Tvornica računarskih strojeva, Radio Industrija Zagreb, Jugoturbina, Unis, Rudi Čajavec i Elektronska Industrija Niš).

Koji su već ili su upravo pred zaključivanjem ugovora o saradnji odnosno odstupanju proizvodnih prava (licence) sa skoro jednakim brojem priznatih inostranih firmi (CDC, DEC, FUJITSU, IBM, UNIVAC, NCR, BURROUGHS, HONEYWELL, KIENZLE, SAGEM, NIXDORF i HEWLETT PACKARD). Pri tome jedino Institut Mihailo Pupin istupa sa potpuno domaćim razvojem i proizvodnjom - hibridnih računara.

Tako u SR Sloveniji dva proizvođača (ISKRA i ELEKTROTEHNA) nude isporuku domaćih računara već za kraj godišne, dok Tovarna meril iz Slovenj Gradca garantuje isporuku domaćih računarskih video terminala za 20. junij 1978.

Do sada, u SFR Jugoslaviji pa ni u pojedinim republikama nije došlo do bilo kakvog ozbiljnog sporazumevanja i

dogovaranja među nabrojanim proizvođačima, koji u glavnom nude računarsku opremu sličnih karakteristika. Pri tome ni jedan od proizvođača ne predviđa proizvodnju velikih računara, što je i opravdano. Ako se uzme u obzir naše mogućnosti i potrebe, svi potencijalni proizvođači garantuju, da će njihovi mali i srednji računari kao i ostala periferna oprema biti sposobni da se povežu odnosno priključe na bilo koji veliki računar.

Uspostavljanje i razvoj integralnih informacionih sistema republika, pokrajina i federacije zavisit će sve više od rezultata domaćeg stvaralaštva a posebno na području proizvodnje računarske opreme koje je tehnička osnova svih savremenih informacionih sistema. Mogućnosti za uvoz te opreme jesu a bit će i u buduću ograničene zato je opredeljenje za domaću proizvodnju (u početnoj fazi na licenčnoj ili kooperantskoj osnovi) više nego nužno. Time ćemo rešavati i pitanje naše politike nezavisnosti i nezavisnosti kao i pitanje naše odbrambene doktrine.

Predlagani koncepti razvoja informacionih sistema zahtevaju specializirane i univerzalne velike računarske centre kao i distribuirane i integralne obrade podataka, baza podataka i informacija u realnom vremenu, funkcionalno povezane i organizovane u jedinstvenoj informacionoj mreži pojedinih republika, pokrajina i federacije. Pri tome planske projekcije potrebne računarske opreme u SFRJ s kojim bi dostigli status informaciono odnosno računarsko razvijene zemlje po sadašnjim normativima pokazuje da bi morali imati:

1. 100 velikih sistema (< 1Mb) ukupne vrednosti	0,3.10 ⁹ g
2. 1000 srednje velikih sistema (< 1Mb) ukupne vrednosti	1,5.10 ⁹ g
3. 10000 malih sistema (< 0,5 Mb) ukupne vrednosti	3.10 ⁹ g
4. 100 000 terminala ukupne vrednosti	3.10 ⁹ g
Ukupno za	7,8.10⁹ g

računarske opreme.

Vrednost 3. i 4. razreda iznosi 6.10⁹ g ili 77 % vrednosti celokupne opreme. Ukupna vrednost velikih sistema iznosi samo 3,8 % vrednosti celokupne opreme.

Sve ovo nam daje dovoljno podataka o tome gde treba da je težište strateške odluke: šta proizvoditi u Jugoslaviji.

Pri usmeravanju domaće proizvodnje moramo uzeti u obzir još i činjenicu, da već sada programska oprema iznosi više od 75 % vrednosti celokupne računarske opreme i kao i da periferna oprema iznosi više od 75 % vrednosti celokupne aparature opreme. Pri tome je razvoj proizvodnje najsavremenijih memorija, komponenti i sklopova veoma težak i skup ali ipak konačni proizvod zbog masovne proizvodnje postaje sve jeftiniji.

Stoga je pravilna odluka većine domaćih proizvođača da se kod proizvodnje aparature opreme povežu sa priznatim inostranim proizvođačima računarske opreme njenih sklopova i komponentama dobavljenih od specializiranih inostranih proizvođača koji, te elemente zbog masovne proizvodnje nude po veoma povoljnijim cenama. Takav koncept proizvodnje kojeg koriste i mnoge renomirane svetske firme, omogućava da se ostvari u vlastitoj radnoj organizaciji i do 85 % vrednosti od ukupne vrednosti konačnog proizvoda, što je od velikog značaja kad se imaju u vidu stalne uvozne (devizne) teškoće naše zemlje. Pri tome domaći proizvođači ostaju u pogledu savremenosti, kvalitete i cene jednakopravni inostranim proizvođačima a pre svega nezavisni pri izboru dobavljača kao i pri nastupanju na inostranom tržištu.

U slučaju koncepta proizvodnje koji se zasniva na kupovini licence nužno je držati se načela nezavisnosti, kvaliteta i konkurentnosti. Kupljenu licencu treba iskoristiti pre svega za brzo osvajanje savremene tehnologije te postepeno osamostaljivanje pomoću vlastitih rešenja i razvoja.

Domaću proizvodnju treba dakle usmeriti na sav spektar potrebne programske opreme a proizvodnju aparature opreme ograničiti na razvoj i proizvodnju malih i srednjih računara, procesnih računara, aktivnih i pasivnih terminala, mikroracunara, jednostavniju perifernu opremu, opremu za prenos podataka, pomoćnu opremu i potrošni material. Pri tome bi trebalo da domaći proizvođači ponude kompletan inženjering za tehnički realizaciju pojedinih informacionih sistema.

Nužno je da proizvođači prilagode proizvodnju predloženom konceptu razvoja i povezivanja informacionih sistema u okviru integralnih informacionih sistema (društvenog sistema informisanja) pojedinih republika, pokrajina i federacije odnosno potrebama najšireg kruga korisnika.

Potrebno je još odgovoriti na pitanje kako realizovati kompleksni zadatak razvoja i ispostavljanja domaće proizvodnje računarske opreme imajući u vidu znatnu deficitarnost sredstava kadrova i znanja na ovom području.

Postavljeni zadatak je moguće rešiti jedino putem širih dogovaranja i sporazumevanja proizvođača i naučno-istraživačkih i pedagoških organizacija te nosioca i korisnika informacionih sistema i AOP.

Pri tome je zadatak svih republika i pokrajina da se založe za donošenje dogovora o jugoslavenskom programu odnosno projektu razvoja proizvodnje računarske opreme, pre svega sa ciljem obezbeđivanja jedinstvenog jugoslavenskog tržišta, uslova kooperacije sa inostranim partnerima i oblika i načina uključivanja u međunarodnu podelu rada. Time će biti obezbeđen jedinstveni nastup prema inostranim proizvođačima koji su zastupljeni u našoj zemlji kao i uspešniji nastup u nesvrstanim zemljama i

zemljama u razvoju. Pri tome ne smemo zaboraviti obaveze koje su Jugoslavija i Alžir preuzele u pogledu razvoja informatike u okviru zajednice nesvrstanih zemalja.

Zadatak Savozne komore je da u programu svoje aktivnosti unese oblik organiziranja, koji će omogućiti da se zainteresirani domaći proizvođači usklade, dogovore i sporazumeju o podeli rada, specializacije, usklađivanje i doprinjavanje proizvodnih programa kao i o zajedničkom planiranju dugoročnog razvoja. Kod ovog dogovaranja, koje će zaći u pojedina ključna infrastrukturna područja društvene reprodukcije je potrebno uzeti u obzir realne komparativne prednosti posebno u kadrovima, sredstvima i znanju te onemogućiti oduzimanje sredstava i rada.

Odgovarajućim propisima i zakonskim odredbama je nužno zaštititi i istovremeno obavezati domaće proizvođače kao i regulisati obaveze inostranih dobavljača u pogledu kompenzacije i kooperacije.

U Jugoslaviji uveliko zaostajemo pri uvođenju računarske tehnologije zato podrška uvođenju domaće proizvodnje računarske opreme ne sme ni u tom slučaju zadržati razvoj informacionih sistema odnosno uvođenje računarske tehnike. Ostaje još da se pripremimo i savladamo sve teškoće koje će doneti preusmeravanje na domaću opremu.

republiška tekmovanja srednješolcev iz računalništva

r. reinhardt*

Institut "Jožef Stefan",
Ljubljana

Prispevek daje osnovne informacije o tekmovanjih srednješolcev iz računalništva, ki smo ju organizirali v preteklih dveh letih v SR Sloveniji. Dodane so tudi vse naloge z rešitvami in kratka analiza rezultatov.

The article gives the essential information about two contests in computer science for students of schools at medium level. Complete set of problems with their solutions and a short overview of contest results are also given.

Slovensko društvo Informatika je v sodelovanju s Fakulteto za elektrotehniko in Institutom Jožef Stefan organiziralo že dve republiški tekmovanji srednješolcev iz računalništva.

Na potrebo po organizaciji tekmovanja kaže solidna tradicija pouka računalništva v slovenskih srednjih šolah, rastoče število srednješolcev, ki obiskujejo ta pouk, potreba po izboljšanju poklicnega usmerjanja in informiranja o študiju računalniških poklicev, nenazadnje pa je tekmovanje tudi dodatna priložnost za srečanje učiteljev računalništva in izmenjavo izkušenj in mnenj.

Prvega tekmovanja se je udeležilo 47 učencev iz 18 šol (14 gimnazij, 3 tehnične srednje šole in ena ekonomska srednja šola). Na drugem tekmovanju pa je tekmovalo že 52 tekmovalcev po prvem letu pouka računalništva in 27 tekmovalcev po dveh letih pouka. Tekmovalci po prvem letu pouka so bili iz 16 šol (10 gimnazij, 4 tehnične in 2 ekonomske srednje šole), po drugem letu pa iz 10 šol (vse gimnazije).

Organizatorji so v okviru obeh tekmovanj pripravili še pogovor o kadrovske problematiki na področju računalništva, izmenjavo mnenj o ciljih, vsebini in metodah pouka računalništva, kakor tudi informativen pogovor o študiju računalniških poklicev. K sodelovanju so povabili širok krog strokovnjakov iz gospodarstva in višjih ter visokih šol, ki so se temu vabilu tudi odzvali.

ZASNOVA TEKMOVANJA

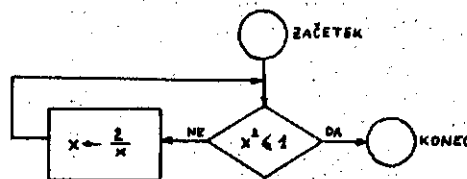
Tekmovalna komisija je ob pripravljanju in izbiranju nalog upoštevala neizenačenost učnih programov, različno znanje dijakov in različne možnosti za praktično delo na računalniku. Zato je poskušala izbrati naloge, ki ne zahtevajo posebnega znanja, niti ne temeljijo preveč na praktičnih izkušnjah učencev. Naloge poskušajo pokriti najvažnejše učne smotre, zato od tekmovalca zahtevajo predvsem razvit smisel za algoritmično razmišljanje.

Na prvem tekmovanju, leta 1977, so bile naloge skupne za obe skupini učencev (po enem letu in po dveh letih pouka računalništva). Rezultati so pokazali (tabela 3), da imajo učenci, ki računalništvo obiskujejo dve leti občutno več izkušenj in znanja. Za drugo tekmovanje je komisija pripravila ločene naloge.

TEKMOVALNE NALOGE PRVEGA REPUBLIŠKEGA TEKMOVANJA IZ RAČUNALNIŠTVA, LJUBLJANA, 16. aprila 1977

Čas reševanja je 2 uri 30 minut, uporabljate lahko vsako literaturo.

- 1) Na številski premici označi množico števil, za katera se naslednji postopek konča:



Oznaka $x \leftarrow \frac{2}{x}$ pomeni, da se vrednost izraza $\frac{2}{x}$ vnese v spremenljivko x .

- 2) Funkcija dveh spremenljivk $f(a, n)$ je definirana z naslednjim predpisom:

$$f(a, n) = \begin{cases} 1, & \text{če je } n = 0; \\ (f(a, \frac{n}{2}))^2, & \text{če je } n \text{ sodo število;} \\ a \cdot f(a, n-1), & \text{če je } n \text{ liho število;} \end{cases}$$

Pri tem je a poljubno realno število, n pa nenegativno celo število.

* Avtor je le zbral in uredil gradivo o tekmovanju. Mnogo mnenj o reševanju nalog in nekaj rešitev so prispevali sodelavci, ki so naloge pripravili in popravljali rezultate: M. Martinec, M. Bohanec, M. Gams in B. Mohar.

- a) Izračunaj $f(2, 5)!$
 b) Kaj izračuna $f(a, n)$? Obrazloži!

- 3) V računalnik vložimo paket kartic (program v visokem programskem jeziku, podatke in potrebne kontrolne kartice). Kaj vse je lahko vzrok, da ne dobimo pričakovanih rezultatov?
- 4) Napiši program, ki bere cela števila, od katerih je vsako na svoji kartici. Branje se konča, če preberemo število 0 ali pa če smo že prebrali sto števil. Program naj izračuna in izpiše aritmetično sredino pozitivnih števil.

- 5) Eden izmed načinov za zapis aritmetičnih izrazov je, da zapišemo operator neposredno pred operanda (v običajnem zapisu stoji med njima). Nekaj primerov:

običajni zapis	novi zapis	rezultat
$5 + 2$	$+ 5 2$	7
$4 * (9 - 2 * 3)$	$* 4 - 9 * 2 3$	12

- a) Izračunaj $* 2 * + 2 2 + 2 * 2 2$
 b) Zapiši algoritem, ki izračuna vrednost izraza, zapisanega v taki obliki.

Operandi so nenegativna cela števila, operatorji pa so $+$, $-$, $*$, $/$.
 Če sta števili eno zraven drugega, je med njima posledek.

2. REPUBLIŠKO TEKMOVANJE SREDNJEŠOLCEV IZ RAČUNALNIŠTVA

NALOGE ZA UČENCE PO PRVEM LETU POUKA

Čas reševanja 2 uri in 30 minut. Ena naloga je neobvezna.

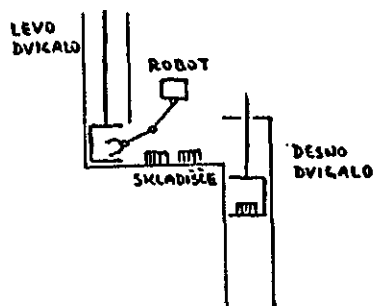
1. Vzemimo, da imamo programski jezik, ki nima logičnih (boolovih) spremenljivk in operatorjev, radi pa bi vseeno take spremenljivke uporabljali. V ta namen bomo vzeli celoštevilčne spremenljivke in imeli vrednost 0 za nepravilno (ne, false) in vrednost 1 za pravilno (da, true). Prevedli logične (boolove) operacije in (konjunkcija, and), ali (disjunkcija, or) in ne (negacija, not) v aritmetične izraze, v katerih uporabljamo samo $+$, $-$, $*$ in $/$.

2. Blagajnik izplačuje osebne dohodke stotim delavcem. Za vsakega delavca ve, kakšen znesek mu mora izplačati. Zapiši postopek, po katerem bi računalnik izračunal, v kakšni obliki mora blagajnik dvigniti denar v banki, da bo vsakemu delavcu njegov znesek izplačal v najmanj bankovcih/kovancih. Računalnik mora torej povedati, koliko kakšnih bankovcev/kovancev naj blagajnik zahteva v banki. Vrednosti bankovcev/kovancev so v polju V dimenzije 13.

3. Transport zabojev poteka z dvema dvigalom. Med dvigalom je robot, ki prelaga zaboje iz levega (dozvoznega) dvigala na desnega (odvoznega). Vsako dvigalo lahko naenkrat pelje samo en zaboj. Robot ima na razpolago še priročno skladišče, v katerega lahko spravi največ deset zabojev. Sestavi postopek, po katerem bi računalnik vodil robota pri opravljanju svoje naloge, če veš, da robot zmora naslednje akcije:

- STANJE (X) ugotovi stanje dvigala X, stanje je lahko odsotno, prazno in polno.
- POŠLJI (X) odpošlje dvigalo X.
- PRENESI (Y, Z) prenese zaboj iz Y v Z.

X je lahko eno od dvigal; Y in Z pa sta ali eno od dvigal ali pa priročno skladišče. Predpostavi, da je v trenutku, ko robota vklopimo, priročno skladišče prazno.



4. Funkciji f in g sta za naravna števila definirani takole:

$$f(a, b) = \begin{cases} 0 & \leftarrow b=0 \\ 1(n, b-1)+a & \leftarrow b>0 \end{cases}$$

$$g(a, b) = \begin{cases} 0 & \leftarrow b=0 \\ g(2a, b/2) & \leftarrow b>0 \text{ in } b \text{ je sod} \\ g(2a, (b-1)/2)+a & \leftarrow b \text{ je lih} \end{cases}$$

- a) Izračunaj $f(5, 3)$ in $g(3, 5)!$
 b) Kakšno matematično operacijo predstavljata f in g ? Utemeljitev!
 c) Koliko ključev funkcije potrebujemo za izračun $f(m, n)$ in koliko za $g(m, n)$, če ju računamo po definiciji? Katera funkcija je učinkovitejša?

5. Napiši program, ki bere zaporedje realnih števil (s kartic, na vsaki kartici je eno število) in izračuna dolžino najdaljšega nepadajočega podzaporedja, npr.:

v zaporedju 3, 8, 9, 2, 1, 8, 7, 9, 9, 10, 11, 3, 2 so nepadajoča podzaporedja

3, 8, 9
 2,
 1, 8
 7, 9, 9, 10, 11
 3 in
 2.

Najdaljše nepadajoče podzaporedje je dolgo 5 členov.

2. REPUBLIŠKO TEKMOVANJE SREDNJEŠOLCEV IZ RAČUNALNIŠTVA

NALOGE ZA UČENCE PO DRUGEM LETU POUKA

Čas reševanja je 2 uri in 30 minut. Ena naloga je neobvezna.

1. Trikotna shema ima obliko:

$$\begin{matrix} a_{1,1} & a_{1,2} & a_{1,3} & \dots & a_{1,20} \\ & a_{2,2} & a_{2,3} & \dots & a_{2,20} \\ & & a_{3,3} & \dots & a_{3,20} \\ & & & \dots & \dots \\ & & & & a_{20,20} \end{matrix}$$

Zapišemo jo v eni vrsti

$a_{1,1}, a_{1,2}, a_{1,3}, \dots, a_{1,20}, a_{2,2}, \dots, a_{19,20}, a_{20,20} =$
 $b_1, b_2, b_3, \dots, b_{20}, b_{21}, \dots, b_{209}, b_{210}$

Na katerem mestu tu v polju b_i se pojavi element $a_{j,k}$?
 Na katerem mestu v shemi se pojavi element b_i (kateri $a_{j,k}$ je b_i)?

2. Kaj izpišeta programa

```

INTEGER F, N, M, I
DATA M /21/
F = 2*M
I = 1
1 CONTINUE
IF (F.EQ.0) GOTO 2
I = I + 1
F = 2*F - 2*I
GOTO 1
2 CONTINUE
WRITE(3,100) I
FORMAT(18)
100 STOP
END
    
```

```

PROGRAM NICLA(OUTPUT)
CONST M=21
VAR I:INTEGER
FUNCTION F(N:INTEGER):INTEGER
BEGIN
IF N>1 THEN
F := 2*F(N-1) - 2*N
ELSE
F := 2*M
END
BEGIN
I:=1
WHILE F(I)<>0 DO I := I + 1
WRITELN(I)
END
    
```

(V Pascalskem programu pomeni \uparrow potenciranje.
 $2 \uparrow n$ pomeni torej 2^n .) Oba programa iščeta ničlo funkcije:

$$f(n) = \begin{cases} 2m & \Leftarrow n=1 \\ 2f(n-1)-2^n & \Leftarrow n>1 \end{cases}$$

3. Imamo krožno cesto, na kateri je poljubno razporejenih n avtomobilov. V rezervoarju imajo skupno toliko goriva, kolikor ga potrebuje eno vozilo za en krog. Vemo (dokazano je), da je mogoče vedno najti avtomobil, ki lahko prevozi cel krog, če lahko ob srečanjih pretoči vse gorivo iz vsakega avtomobila v svoj rezervoar. Zapiši algoritem, ki bo poiskal tako vozilo. Podatki za tvoj algoritem so: za vsak avto njegov položaj na cesti (dolžina loka od neke določene točke na krogu za vse avtomobile v isti smeri) in razdalja, ki jo lahko prevozi s svojim gorivom.

4. Napiši postopek, ki ciklično premakne elemente v polju dolžine n za k mest na desno ($k \leq n$). Na voljo imamo operacijo ZAMENJAJ (1,j), ki zamenja elementa na i -tem in j -tem mestu med seboj. Poleg tega ne smemo uporabljati nobene druge spremenljivke razen števecv.
 Primer: polje 1,2,3,4,5,6,7,8 ciklično premaknemo za tri mesta in dobimo 6,7,8,1,2,3,4,5.
 Oцени število operacij (t.j. ključev programa ZAMENJAJ) v odvisnosti od n in k ! Poskusi najti algoritem, ki pri svojem izvajanju čim manjkrat pokliče podprogram ZAMENJAJ!

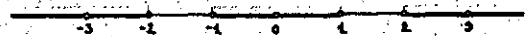
5. Napiši program, ki izpiše na tiskalnik na 41 vrsticah lik oblike:

```

      *
     ***
    *****
   *****
  *****
 *****
*****
*****
 *****
  *****
   *****
    *****
     ***
      *
    
```

REŠITVE NALOG IZ 1. REPUBLIŠKEGA TEKMovanja SREDNJEŠOLCEV IZ RAČUNALNIŠTVA (16. april 1977)

1. Označimo z x začetno vrednost spremenljivke x . Zanka se konča že pri prvem prehodu, če je $-1 \leq x \leq 1$. V drugem prehodu se zanka konča, če je $-1 \leq 2/x_0 \leq 1$, torej $x_0 \leq -2$ ali $x_0 \geq 2$. Če se zanka v teh dveh primerih ne izteče, se ne bo nikoli, saj dobi v drugem prehodu x spet vrednost x_0 ($x \leftarrow 2/(2/x_0)$). Postopek se torej konča, če x_0 leži na označenih intervalih:



2. a) $f(2,5) = 2 \cdot f(2,4) = 2 \cdot (f(2,2))^2 = 2 \cdot ((f(2,1))^2)^2 = 2 \cdot (f(2,1))^4 = 2 \cdot (2 \cdot f(2,0))^4 = 2 \cdot (2 \cdot 1)^4 = 32$

b) Trdimo za vsak a in vsak $n \geq 0$ velja $f(a,n) = a^n$ (*)

Trditve dokazemo z indukcijo:
 $f(a,0) = 1$; pri $n = 0$ torej trditev (*) velja.
 - naj trditev (*) velja za vse $n < k$.
 Dokazimo, da velja tudi za $n = k$. Tedaj ločimo dva primera:

1. k je liho število:
 $f(a,k) = a \cdot f(a,k-1)$
 Ker je $k-1 < k$, uporabimo indukcijsko hipotezo $f(a,k) = a \cdot a^{k-1} = a^k$.

2. k je sodo število:
 $f(a,k) = (f(a, \frac{k}{2}))^2$
 Ker je $\frac{k}{2} < k$ lahko spet uporabimo indukcijsko hipotezo in dobimo
 $f(a,k) = (a^{k/2})^2 = a^k$

3. Vzroki so lahko naslednji:
 napaka v krmilnih karticah
 slovnična napaka v programu
 pomenska napaka v programu
 napačni podatki
 ne pričakujemo pravih rezultatov
 napačno prebran paket
 napaka v prenosu podatkov
 napaka v sistemskem softwaru
 napaka v hardwaru

```

PROGRAM MEC(INPUT,OUTPUT)
CONST NUM=1001
VAR M,N,S,X:INTEGER
BEGIN N:=0; M:=0; S:=0;
REPEAT
READLN(X); N:=N+1;
IF X>0 THEN
BEGIN S:=S+X; M:=M+1; END
UNTIL (X=0) OR (N=NUM);
IF M<>0 THEN WRITELN(S/M)
END.
    
```

```

INTEGER X
S=0.0
N=0
M=0
1 CONTINUE
  READ(2,100) X
  FORMAT(16)
  N=N+1
  IF(X.LE.0)GOTO 2
  S=S+X
  M=M+1
2 CONTINUE
  IF((X.NE.0).AND.(N.LT.100))GOTO 1
  IF(M.EQ.0)STOP
  S=S/M
  WRITE(3,101)S
  FORMAT(F15.6)
101 END

```

5. a) vrednost izraza izračunamo

$$\begin{aligned}
 & \times 2 \times + 2 \times 2 + 2 \times 2 \times 2 = \\
 & = \times 2 \times + 2 \times 2 + 2 \times 4 = \\
 & = \times 2 \times + 2 \times 6 = \\
 & = \times 2 \times 4 \times 6 = \\
 & = \times 2 \times 24 = \\
 & = 48
 \end{aligned}$$

b) Postopek za izračun vrednosti:

dokler ni v vrstici eno samo število ponavljaj

Poišči zadnji operator in ga uporabi nad številoma, ki mu sledita;

Vrednost izraza je število, ki ostane v vrsti;

REŠITVE NALOG IZ DRUGEGA REPUBLIŠKEGA TEKMOVA-NJA SREDNJEŠOLCEV IZ RAČUNALNIŠTVA (po prvem letu pouka)

1. Če si napišemo pravilnostne tabele hitro ugotovimo

x and y se prevede v $(x \times y)$
 x or y se prevede v $(x + y' - x \times y)$
not x se prevede v $(1 - x')$

Pravilnostne tabele (pravilno= $\text{da}=\text{true}$ okrajšamo s T; nepravilno= $\text{ne}=\text{false}$ okrajšamo z F)

xy	$x \text{ and } y$	$x \text{ or } y$	<u>not</u> x	$x' y'$	$x y'$	$x' y$	$x y$	$1 - x'$
FF	F	F	T	0 0	0 0	0 0	0 0	1
FT	F	T	T	0 1	0 1	1 1	1 1	1
TF	F	T	F	1 0	0 1	1 1	1 0	0
TT	T	T	F	1 1	1 1	2 1	1 0	0

2. Vsebina polja V naj bo taka kot v nalogi, vpeljemo še polje K, ki bo vsebovalo število posameznih kosov bankovcev/kovancev.

- Postavimo začetne vrednosti v V in K.
- Stokrat prečitamo znesek, ga razdelimo v bankovce/kovance in število kosov prištejemo ustreznim k-jem
- Rezultate (k) izpišemo.

Predpostavimo, da so vrednosti v polju V urejene po velikosti, tako da je največja vrednost na prvem mestu, najmanjša pa na zadnjem. Vzemimo tudi, da so zneski in vrednosti v polju V podani v parah. Podroben postopek je tedaj:

- Postavi vrednosti v V : $V_1 \leftarrow 100000$; $V_2 \leftarrow 50000$; $V_3 \leftarrow 10000$; ...; $V_{13} \leftarrow 5$;
- Za vse i med 1 in 13 ponovi $k_i \leftarrow 0$;

2. stokrat ponovi

2.1. prečitaj znesek

2.2. za $i = 1, 2, 3, \dots, 13$ ponovi

2.2.1. naj bo m znesek div V_i

2.2.2. k_i -ju prištej m : $k_i \leftarrow k_i + m$

2.2.3. znesku odštej $V_i \times m$: znesek \leftarrow znesek - $V_i \times m$

3. za $i = 1, 2, 3, \dots, 13$ ponovi

3.1. izpiši k_i

3. Dvigala in transport uporablja računalnik takole: (razvoj algoritma je zaradi prostorskih omejitev opuščeno)

$k \leftarrow 0$;

za vedno ponavljaj

če je STANJE(D):

prazen: če $k > 0$ potem

PRENESI(S,D);

$k \leftarrow k - 1$;

ODPOŠLJI(D);

poln: ODPOŠLJI(D);

odsoten: ;

če je STANJE(L):

prazen: ODPOŠLJI(L);

poln: če $k < 10$ potem

PRENESI(L,S);

$k \leftarrow k + 1$;

ODPOŠLJI(L);

odsoten: ;

$$4.a) f(5,3) = f(5,2)+5 = (f(5,1)+5)+5 = ((f(5,0)+5)+5)+5 = 0 + 5 + 5 + 5 = 15$$

$$g(3,5) = g(6,2)+3 = g(12,1)+3 = (g(24,0)+12)+3 = 0 + 12 + 3 = 15$$

b) Tako f kot g zmnožita svoja argumenta: torej

$$f(a,b) = g(a,b) = a \cdot b \quad (*)$$

Dokaz (utemeljitev): dokazovali bomo s popolno indukcijo po b

b.1. $b = 0$ $f(a,b) = 0$; $g(a,b) = 0$ in tudi $a \cdot b = 0$, torej je pri $b = 0$ trditev (*) veljavna.

b.2. predpostavimo, da (*) velja za vse b, ki so strogo manjši od n. Dokažimo, da velja potem tudi

$$f(a,n) = g(a,n) = a \cdot n$$

$f(a,n) = f(a,n-1) + a$; upoštevamo induksijsko hipotezo in $f(a,n) = a(n-1) + a = a \cdot n$

$g(a,n) =$ če je n liho število
 $= g(2a, (n-1)/2) + a$; upoštevamo induksijsko hipotezo in $g(a,n) = 2a \cdot (n-1)/2 + a = a(n-1) + a = a \cdot n$

Če pa je n sodo število, je $g(a,n) = g(2a, n/2)$, to je po induksijski hipotezi enako $2a \cdot n/2$, kar je isto kot $a \cdot n$. S tem je trditev dokazana!

c) Vrednost funkcije f izračunamo tako, da b zmanjšamo za 1 in jo znova pokličemo. Število klicev funkcije je torej kar enako b.

Ko računamo vrednost funkcije g, pokličemo funkcijo vedno z argumentom, ki je manjši ali enak $b/2$, saj je $b/2 \leq b/2$ in $(b-1)/2 < b/2$. Z vsakim klicem se torej vrednost drugega argumenta vsaj razpolovi. Račun se konča, ko je drugi argument enak 0.

Zaporedni b-ji so torej manjši ali enaki naslednjim številom:

$$b, b/2, b/4, b/8, \dots, b/2^n,$$

kjer je n število klicev. Lihim argumentom vsakokrat odštejemo enico, tako da se deljenje povsod izide, torej lahko v zaporedju gledamo povsod celi del kvocienta:

$$\lfloor b \rfloor, \lfloor b/2 \rfloor, \lfloor b/4 \rfloor, \lfloor b/8 \rfloor, \dots, \lfloor b/2^n \rfloor.$$

Celi del števila je enak 0, če je število manjše od 1

$$b/2^n < 1 \Rightarrow b < 2^n \Rightarrow \log b < n \log 2 \Rightarrow n > \frac{\log b}{\log 2}$$

Prejšnji člen zaporedja pa mora biti večji ali enak 1, saj bi se sicer zaporedje končalo že pri njem:

$$b/2^{n-1} \geq 1 \Rightarrow b \geq 2^{n-1} \Rightarrow \log b / \log 2 + 1 \geq n. \text{ Za število korakov velja torej neenakost:}$$

$$\log b / \log 2 < n \leq \log b / \log 2 + 1.$$

Ker logaritemska funkcija mnogo počasneje raste kot linearna funkcija, sklepamo, da je g učinkovitejša.

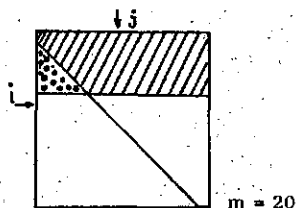
5.

```
PROGRAM CETE (INPUT,OUTPUT);
VAR X,PREJ:REAL;
D,MAX:INTEGER;
BEGIN
  READLN(PREJ); D:=1; MAX:=0;
  WRITELN(PREJ:15:3);
  WHILE NOT EOF DO
  BEGIN
    READLN(X);
    WRITELN(X:15:3);
    IF X>PREJ THEN
      D:=D+1
    ELSE
      BEGIN
        IF D>MAX THEN MAX:=D;
        D:=1
      END;
    PREJ:=X
  END (*WHILE*);
  WRITELN(# NAJDALJ7E NEPADAJODE
  END.          PODZAPOREDJE IMA#,MAX:4,# #LENOV.#)
```

```
REAL X,PREJ
INTEGER D,MAX
FORMAT(F10.0)
READ(2,100)PREJ
D=1
MAX=0
X=PREJ
1 CONTINUE
IF (X.EQ.0)GOTO 2
READ(2,100)X
IF (X.LT.PREJ)GOTO 3
D=D+1
GOTO 4
3 CONTINUE
IF (D.GT.MAX)MAX=D
D=1
4 CONTINUE
PREJ=X
GOTO 1
2 CONTINUE
101 WRITE(3,101)MAX
FORMAT(38# NAJDALJ7E NEPADAJODE
STOP          PODZAPOREDJE IMA,I4,8# #LENOV.)
END
```

REŠITVE NALOG IZ DRUGEGA REPUBLIŠKEGA TEKMOVANJA IZ RAČUNALNIŠTVA (dve leti pouka)

1.



Da določimo zaporedno število i, j -tega elementa je potrebno najprej prešteti, koliko elementov je v šrafiranem delu in temu številu prišteti zaporedno število i, j -tega elementa v i -ti vrstici.

V šrafiranem delu je $(i-1) \cdot m - \frac{(i-2)(i-1)}{2}$ elementov, saj jih je v celem zgornjem pravokotniku $(i-1) \cdot m$, v pikčastem trikotniku pa jih je $\frac{i(i-1)}{2}$.

i, j -ti element je na $j - (i-1)$ -tem mestu v i -ti vrstici, saj v i -ti vrstici manjka ravno $(i-1)$ elementov na začetku. Zaporedna številka i, j -tega elementa je torej:

$$l = j + \frac{(i-1)(2m-i)}{2}$$

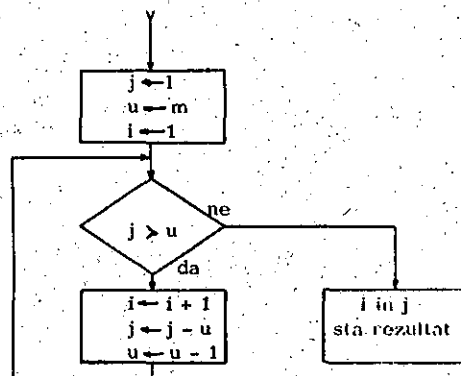
V obratno smer je naloga dosti težja, če se je hočemo lotiti z izrazom. Poskusimo najprej zapisati algoritem, ki iz l izračuna i in j :

$j \leftarrow 1; u \leftarrow m; i \leftarrow 1; (* u$ je dolžina i -te vrste*)

dokler je $j > u$ ponavljaj

```
i ← i + 1;
j ← j - u;
u ← u - 1;
```

i, j je mesto l -tega elementa



Navedimo le še formuli za izračun i in j iz l (brez izpeljave, ki je nekoliko daljša):

$$i(l) = m - \left\lfloor \frac{\left\lfloor \sqrt{1 + 8 \left(\frac{m(m+1)}{2} - l \right)} \right\rfloor - 1}{2} \right\rfloor$$

$$j(l) = 1 - \frac{(i(l)-1)(2m-i(l))}{2}$$

Kjer pomeni $\lfloor \cdot \rfloor$ celi del izraza e .

2. Oba programa očitno iščeta ničlo funkcije

$$f(n) = \begin{cases} 2m & \Leftarrow n = 1 \\ 2f(n-1) - 2^n & \Leftarrow n > 1 \end{cases}$$

$$f(1) = 2m$$

$$f(2) = 2^2 m - 2^2$$

$$f(3) = 2^3 m - 2^3 - 2^3$$

$$f(4) = 2^4 m - 2^4 - 2^4 - 2^4$$

.....

$$f(n) = 2^n m - (n-1) 2^n = \underline{2^n(m-(n-1))} \quad (\ast)$$

Dokaz z indukcijo:

$$a) f(1) = 2^1(m-(1-1)) = 2m$$

b) naj trditev (\ast) velja pri vseh $n < k$. Dokažimo tudi veljavnost pri k .

$$\begin{aligned} f(k) &= \text{po definiciji} = 2f(k-1) - 2^k = \text{po indukcijski hi-} \\ &\text{potezi} = 2 \cdot 2^{k-1} (m-(k-2)) - 2^k = \\ &= 2^k (m-(k-1)) \end{aligned}$$

S tem je trditev (\ast) dokazana za vsak n . Zdaj pa ni več težko poiskati ničle funkcije f . Očitno je $f(n) = 0$ tedaj, ko je $\underline{n = m + 1}$.

Oba programa torej izpišeta 22.

3. Nalogo lahko izvedemo na dva načina:

1. Z navadnim preiskovanjem (tega tu ne bomo opisovali).
2. Tako, da pri preiskavi upoštevamo tisto, kar smo že izračunali pri prejšnjih preizkusih.

Ideja postopka:

Ker je dovolj, da poiščemo en avto, ki lahko prevozi cel krog, si lahko smer vožnje predpišemo vnaprej. Izberimo si avtomobil;

Dokler z "izbranim avtom" ne obkrožimo ceste ponavljaj.

Če lahko "izbrani avto" s svojim gorivom pripelje do sosedaj, potem sosedaj zbrišemo s ceste, vse njegovo gorivo pa pretočimo v "izbrani avto".

Če "izbrani avto" ne more pripeljati do svojega sosedaj, potem si izberemo nek drug avtomobil, ki doslej še ni bil izbran (npr.: drugega sosedaj izbranega avtomobila).

Algoritem bo uporabljal razdalje med avtomobili (in ne njihovih položajev, zato najprej izračunamo razdalje). S $P(i)$ označimo položaj i -tega avtomobila, $D(i)$ naj bo doseg i -tega avtomobila, $R(i)$ pa bo razdalja med i -tim in $i+1$ -im avtomobilom. Avtomobile štejemo od 0 do $n-1$. Predpostavimo, da vedno velja $P(i) \leq P(i+1)$ (za $i = 0, 1, \dots, n-2$) – če to ne velja, preuredimo podatke!

Izračunajmo obseg kroga in R :

obseg $\leftarrow 0$;

za vse i med 0 in $n-1$ ponovi
obseg \leftarrow obseg + $D(i)$;

za vse i med 0 in $n-2$ ponovi
 $R(i) \leftarrow P(i+1) - P(i)$;

$R(n-1) \leftarrow$ obseg - $P(n-1) + P(0)$;

Preden se lotimo podrobnega postopka, ugotovimo še, na kakšen način bomo brisali avtomobile s ceste. Opreмимо vsak avtomobil poleg razdalje do naslednjega avtomobila in dosega še s podatkom M_i , ki pove koliko avtomobilov naprej po cesti je naslednji (neizbrisan) avto. Na začetku bo seveda za vsak avto $M(i)=1$. Ko izbršemo avtomobil s ceste, napravimo to tako, da

vse j -jeve podatke prištejemo njegovemu neposrednemu predhodniku. Podrobnosti naj pojasni dokončen postopek:

obseg $\leftarrow 0$;

za vse i med 0 in $n-1$ ponovi
obseg \leftarrow obseg + $D(i)$;

za vse i med 0 in $n-2$ ponovi
 $R(i) \leftarrow P(i+1) - P(i)$;

$R(n-1) \leftarrow$ obseg - $P(i+1) + P(0)$;

za vse i med 0 in $n-1$ ponovi
 $M(i) \leftarrow 0$;

$i \leftarrow n$;

ponavljaj

$i \leftarrow i-1$;

dokler $R(i) \leq D(i)$ in $M(i) < n$ ponavljaj

$j \leftarrow (i+M(i)) \bmod n$;

$R(i) \leftarrow R(i) + R(j)$;

$D(i) \leftarrow D(i) + D(j)$;

$M(i) \leftarrow M(i) + M(j)$;

dokler ni $D(i) \gg$ obseg;

izpiši 1;

4. Nalogo bomo rešili na dva načina:

- a) Zelo hitro uganemo, kako napraviti ciklični premik za eno mesto. Če to k -krat ponovimo dobimo ciklični premik za k mest. Ciklični premik za eno mesto napravimo tako, da zamenjujemo po dva soseda in sicer od zadaj naprej. Primer:

```

1 2 3 4 5 6 7
1 2 3 4 5 7 6
1 2 3 4 7 5 6
1 2 3 7 4 5 6
1 2 7 3 4 5 6
1 7 2 3 4 5 6
7 1 2 3 4 5 6

```

za vse i med $n-1$ in 1 v padajočem vrstnem redu zamenjaj $(i, i+1)$;

$I = N-1$

DO 1 $I = 2, N$

ZAMENJAJ $(I, I+1)$

$I = I - 1$

1 CONTINUE

for $i := n-1$ downto 1 do

zamenjaj $(i, i+1)$;

Vse skupaj k -krat ponovimo. Za ciklični premik za eno mesto smo potrebovali $(n-1)$ ključev podprograma ZAMENJAJ, ker to k -krat ponovimo je celotno število operacij enako $k(n-1)$.

b) algoritem razdelimo na 3 dele:

1. obrnemo celo tabelo
2. obrnemo prvih k elementov
3. obrnemo zadnjih $n-k$ elementov

Pri tem "obrnemo" pomeni, da elemente prestavimo tako, da pride prvi na zadnje mesto, drugi na predzadnje, ..., zadnji na prvo mesto.

Tabelo obrne takle postopek:

prvi element na p -tem mestu, obračamo q elementov: obrni (p, q) ;

za vse i med 0 in $\lfloor \frac{q}{2} - 1 \rfloor$ ponovi

zamenjaj $(p+i, p+q-i-1)$;

Očitno je, da za obračanje q dolgega kosa tabele potrebujemo $\lfloor \frac{q}{2} \rfloor$ ključev podprograma zamenjaj.

Cel algoritem:

1. obrni (1,n);
2. obrni (1,k);
3. obrni (n+1, n-k);

št. operacij:

$$\left\lfloor \frac{n}{2} \right\rfloor + \left\lfloor \frac{n-k}{2} \right\rfloor + \left\lfloor \frac{n-k}{2} \right\rfloor \leq n$$

5.

```

INTEGER PRESL,ZNAK,I,J,A,B,V1,VRST,STOLP
DATA PRESL,ZNAK/1H ,1H*/
DATA VRST,STOLP/21,41/
FORMAT(130A1)
FORMAT(1H1)
WRITE(3,101)
B=1
A=STOLP
V1=VRST-1
DO 2 I=1,V1
  WRITE(3,100) (PRESL,J=1,A), (ZNAK,J=1,B)
  A=A-1
  B=B+2
2 CONTINUE
DO 1 I=1,VRST
  WRITE(3,100) (PRESL,J=1,A), (ZNAK,J=1,B)
  A=A+1
  B=B-2
1 CONTINUE
STOP
END

```

```

PROGRAM KARO(OUTPUT);
CONST VRST=21; STOLP=41; ZNAK=###;
VAR I,J,A,B:INTEGER;
BEGIN PAGE(OUTPUT);
  A:=STOLP; B:=1;
  FOR I:=1 TO VRST-1 DO
    BEGIN
      WRITE(# #:A);
      FOR J:=1 TO B DO WRITE(ZNAK:1);
      WRITELN;
      A:=A-1; B:=B+2
    END;
  FOR I:=1 TO VRST DO
    BEGIN
      WRITE(# #:A);
      FOR J:=1 TO B DO WRITE(ZNAK:1);
      WRITELN;
      A:=A+1; B:=B-2
    END
  END.

```

ANALIZA USPEHA PO NALOGAH

1977:

1. naloga: Učenci so pogosto pozabljali ugotoviti, pri kakšnih x se zanka izteče po drugem, tretjem, itd. koraku. Hud problem pa je bil tudi rešiti pravilno zapisano neenačbo.

2. naloga: Pri tej nalogi je bilo največ ničel. Razlog za to je verjetno v tem, da mnogi še niso videli rekurzivne funkcije. Tisti pa, ki so nalogo rešili, so jo večinoma rešili pravilno, le na dokaz so pozabili. Korekten dokaz, da predstavlja funkcija potenciranje, je napravilo zelo malo učencev.

3. naloga: Naloga je zgrešena, saj ni mogoče dati izčrpnega odgovora. Mnogi so se izgubili v naštevanju vseh mogočih sintaktičnih napak, na druge pa so pozabili.

4. naloga: Vsi programi zapisani v pascalu so zasnovani pravilno, čeprav so tekmovalci poslušali le eno leto računalništva. Fortranski programerji so po dveh letih bistveno slabši in po nepotrebnem zapletajo. V pascalskih programih je največkrat napačno sestavljen pogoj v **while** stavku (je negiran, tudi sintaktično napačen). V fortranskih programih pa so mnogokrat po nepotrebnem uporabljali polja (tega ni napravil nihče v pascalu), zelo pogosto se pojavlja pogoj $(a = 0)$ v **if** stavku, precej je skokov iz **if** stavka na **goto**. Opazimo grdo tradicijo aritmetičnih **if** stavkov.

5. naloga: Očitno je, da učencev nihče ni učil strukturiranega pristopa k reševanju problema. Pozna se, da so le redki navajeni zapisati algoritem v slovenščini, opazimo tudi zaslepljenost s fortranom. Opazimo lahko tudi, da učenci v Šoli niso navajeni reševati nenumeričnih problemov. Kljub vsemu pa je presenetljivo število pravih rešitev, saj je naloga precej težka.

1978/ po prvem letu pouka

1. naloga: Večina je nalogo pravilno rešila, saj je bila precej enostavna. Tisti, ki je niso rešili, očitno o logičnih spremenljivkah še niso slišali.

2. naloga: Z razumevanjem naloge ni bilo težav. Pogosto so jo reševali tako, da so obdelali vsak kovanec posebej, tj. brez uporabe polja, kar je vodilo do dolgih in kompliciranih algoritmov. Drugih izrazitih napak ni bilo, razen morda tu in tam pozabljena kakšna prireditev (pri programih) ali kakšen važnejši člen pri algoritmihi. Zanimivo je, da so bile mnogo boljše rešitve podane v obliki programov in ne v drugih opisnih oblikah. V prostem zapisu je bila res odlična ena sama rešitev. Kakšni trije odgovori so bili narejeni v stilu: v računalnik vložim kartico, on pa mi natipka rezultat.

3. naloga: Čeprav je bila naloga nestandardna in je zahtevala razmišljanje o vzporednem procesu (oz. dveh asinhronih vzporednih procesih), so dijaki nalogo kar uspešno rešili. Vsaj idejno je nalogo rešila velika večina. Tisti, ki so upoštevali navodilo, naj narišejo tabelo stanj in vanjo vpišejo potrebne akcije, so imeli veliko lažje delo, ker so lahko razmišljali bolj svobodno. Kdor pa je nalogo skušal rešiti na standarden način - z algoritmom oz. diagramom poteka, je večinoma ustvaril zelo nepregleden in redundanten diagram poteka (čeprav navadno pravilen). Pogrešali smo rešitve v obliki lepega algoritma ali programa, čeprav bi bil npr. program v Pascalu lahko zelo preprost in pregleden. Priporočamo, da se tudi prihodnje leto zastavi podobna naloga, ki zahteva razmišljanje o krmiljenju procesa v realnem času.

4. naloga:

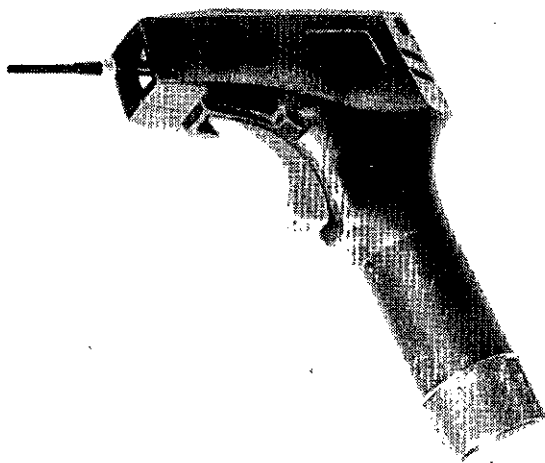
- a) Izračunati f in $g(5,3)$: V glavnem uspešno, napačni rezultati pa so bili predvsem posledica računskih napak oz. napačne ocenitve, kdaj je konec rekurzije.
- b) kakšna matematična operacija sta f in g : Precej trši oreh, predvsem funkcija g . Mnogo je bilo trditvej apriori; brez dokaza. Napaka pri točki a) je vodila do napake pri tem podvprašanju.
- c) Malo pravih odgovorov, napake pa varirajo predvsem za 1. Pohvalno je, da so nekateri računali število klincev za poljubni b . Na vprašanje katera funkcija je učinkovitejša, sta odgovorila dva (ali še kakšen več). Posebnih napak v načinu reševanja te naloge nismo opazili. Nalogo 4 je v primerjavi z nalogo 2 reševalo precej manj tekmovalcev. Od podvprašanja a) proti c) je število odgovorov vidno padalo.



OK MACHINE AND TOOL CORPORATION / 3455 CONNER STREET, BRONX, NEW YORK 10475, U.S.A.

Phone: (212) 994-6600 • Telex: 12 5091 • Telex: 23 2395

HOBBY-WRAP TOOL



OK MACHINE AND TOOL CORPORATION
3455 CONNER STREET, BRONX, N.Y. 10475 U.S.A. PHONE: (212) 994-6600
TELEX NO: 125091 TELEX NO: 232395

AMATERSKO ORODJE ZA OŽIČEVANJE

Model BW-630 je orodje na baterijski pogon za ožičevanje žice tipa 30 AWG na standardne trne, ki so med seboj oddaljeni 1,65 mm. Orodje je opremljeno s kompletom, ki omogoča izdelavo "modificiranega" načina ožičevanja. Vgrajena je tudi naprava, ki preprečuje nategovanje žice. Konstrukcija je prilagojena delu resnih amaterjev; teža orodja je 40 dkg in se napaja preko standardnih ali akumulatorskih baterij velikosti "C". Ohišje pištole je izdelano iz hrupave površine in zavarovano pred udarci. Baterije niso vključene v komplet.

ORODJE ZA OŽIČEVANJE-ODVIJANJE IN SNEMANJE IZOLACIJE

Ceneno orodje, ki opravlja funkcijo treh orodij, s podobno ceno. Z orodjem je mogoče ožičevati, odvijati in snemati izolacijo, s posebnim rezilom, vgrajenim v ročaj. Orodje primerno za delo z žico tipa 30 AWG (0,25 mm), katero se ovije na standardne (0,6 mm) trne podnožij za integrirana vezja. Uporabe se naučimo v nekaj minutah, žico pa ovijemo v nekaj sekundah ne da bi uporabili spojko. K orodju je priloženo tudi navodilo za uporabo.

HOBBY-WRAP-30



STRIP

WRAP

UNWRAP

OK MACHINE & TOOL CORPORATION

3455 CONNER STREET, BRONX, NEW YORK, N.Y. 10475 U.S.A. • PHONE: (212) 994-6600
TELEX: 125091 TELEX: 232395

Ko pišete proizvajalcu, omenite časopis INFORMATICA.

novi časopis: "information and management"

Dubravka Čežez-Kecmanović

Elektrotehnički fakultet, Sarajevo

Obično se u mladim i nedovoljno razvijenim disciplinama ili oblastima događa da nova saznanja, istraživački rezultati i praktična iskustva, mijenjaju ne samo važeće teorijske koncepte ili čitave teorijske sisteme, već i sama shvatanja o tome šta ta disciplina jest i šta ona obuhvata, na koji način, nerijetko, problematizuju i njeno postojanje i njenu važnost uopšte. Ovakvi događaji na putu konstituisanja jedne discipline manifestuju se kao krizne tačke, u kojima se ili pokaže da je dalji razvoj bezperspektivan, nakon čega interes za disciplinu naglo opada, ili se pokažu više ili manje obećavajući prvci daljeg razvoja.

Čini se da oblast informacionih sistema zadnjih godina prolazi kroz jednu krizu ove druge vrste. Okviri računarskih nauka, u kojima se radila oblast "obrade podataka" (kod nas popularno zvana AOP), već odavno su postali tjesni. Uloga i uticaj informacione tehnologije na kompleksne ekonomske i društvene sisteme znatno su proširili spektar problema "obrade podataka" i doprinijeli izdvajanju oblasti informacionih sistema i porastu stručnog i naučnog interesa za ovu oblast. O tome svjedoči i pojava novog časopisa u ovoj oblasti pod naslovom "Information & Management", (North-Holland Publ. Comp.*) umjesto predhodnog IAG-ovog časopisa "Management Datamatics" (raniji naziv "Management Informatics"), glavni urednik je prof. E.H. Sibley. Ovdje se ne radi samo o promjeni imena, forme i uredničkog kolegija časopisa, već o novoj orijentaciji, sa preciznijim profilom i novim ciljevima časopisa, što je, kako kažu njegovi urednici, rezultat svijesti o nastalim promjenama u oblasti informacionih sistema.

"Informacija nije sama sebi cilj, ona je sredstvo za odvijanje neke aktivnosti; u većini slučajeva informacije su potrebne za upravljanje - zadatak koji pokriva širok spektar ljudskih aktivnosti. Gdje god se obavlja upravljački zadatak, ... potrebno je prikupljati, povezivati, korelirati i analizirati informacije. Ime novog časopisa upravo reflektuje ova shvatanja" - kaže se u uvodniku prvog broja časopisa "Information & Management".

Osnovni ciljevi časopisa su:

- prikupljanje i širenje informacija o novim i značajnim rezultatima u oblasti primjene informacionih sistema
- pružanje materijala za obrazovanje i obuku u oblasti informacionih sistema,
- stimulacija usavršavanja metodologije razvoja informacionih sistema i primjene,
- razmatranje problema razvoja i korištenja informacionih sistema za upravljanje - donošenje politika, strateških odluka i drugih upravljačkih aktivnosti - u privredi, nepri-

vrednim organizacijama, državnim organima i internacionalnim organizacijama i institucijama,
- djelovanje kao formalno glasilo IAG-a (IFIP Applied Information Processing Group) u njegovoj naučnoj i obrazovnoj ulozi.

Časopis je namijenjen i korisnicima informacionih sistema i onima koji ga kreiraju i upravljaju njegovim radom kao i istraživačima u oblasti informacionih sistema. Svakome od njih časopis treba da pruži relevantne sadržaje.

Ovakvi ciljevi časopisa odredili su i njegovu strukturu. U časopisu se predviđa šest kategorija priloga:

- sažeti prikazi (briefings) sa ciljem informisanja o novim idejama, konceptima, metodama, itd.; ovo je također i prostor u kome eminentni stručnjaci mogu izložiti mišljenje o stanju u oblasti razvoja i primjene informacione tehnologije i predviđati buduća kretanja,
- aplikacije, za kritičko razmatranje metoda odnosno tehnika, koje se primjenjuju,
- tehnike, za prikaz tehnoloških inovacija u analizi, projektovanju, implementaciji i upravljanju informacionim sistemima,
- prikazi slučajeva (case studies), za detaljno razmatranje realnih praktičnih problema i njihovih rješenja,
- istraživanje, za prikaz rezultata istraživanja, koja se mogu (ili će se brzo moći) primijeniti u praksi,
- značajni članci i knjige, za kritičke prikaze radova koji su ocijenjeni kao vrijedni.

Izložena struktura časopisa otvara mogućnosti ostvarenja ciljeva časopisa. Za sada jedino ostaje otvoreno kako i u kojoj formi će se izlagati problemi obrazovanja i obuke u oblasti informacionih sistema, budući nisu eksplicitno obuhvaćeni u navedenim kategorijama.

U prvom broju časopisa, koji se pojavio krajem prošle godine, zastupljeni su prikazi iz svake kategorije:

Sažeti prikaz: Hammer C. "A Forecast of the Future of Computation",

Aplikacija: Schewe C.D., Wiek J.L. and R.T. Dann, "Marketing the MIS",

Tehnike: Bally L., Britton J. and K.H. Wagner, "A Prototype Approach to Information System Design and Development",

Analiza slučaja: Hauen C., "An Inter-Related Processing Network Architecture",

Istraživanje: Barkin S.R. and G.W. Dickson, "An Investigation of Information System Utilization",

Prikazi knjige: "Management Information and Control Systems" by R.I. Tricker, prikazoo J. Moanta, "Software Metrics" by T. Gilb, prikazoo D. Teichroew.

* Adrese uredništva:

Dep. Inform. Syst. Managem.
University of Maryland
College Park, MD 20742, USA

IAG Headquarters
40, Paulus Potterstraat
Amsterdam, The
Netherlands

Opšti nivo časopisa je zavidno visok, kako u pogledu svježine ideja i informativnosti sadržaja, tako i u pogledu stručnosti i naučne relevantnosti priloga.

Posebnu pažnju zaslužuje Hammer-ov članak o budućnosti informacione tehnologije: tendenciji razvoja hardware-a i predviđanja razvoja do 1990 g., promjenama u arhitekturi računara, distribuiranim sistemima i software-u, koje se mogu očekivati.

Trideset godina stara (ili mlada) računarska tehnologija, u trci da nam proži što brže procesore, veće memorije i što pouzdanije komponente, intenzivno se razvija. Posljednja dostignuća u znaku su "čudo zvanog čip". Počevši od integriranih kola, koja su se pojavila šezdesetih godina, povećanjem stepena integracije postižu se sve kompleksniji čipovi, dok troškovi ostaju praktično isti. Pojavljuju se i usavršavaju nove tehnike, koje omogućuju sve veću gustinu čipa (bit/čip) paralelno sa smanjenjem debljine slojeva (mjereno atomskim dijametrima). Za monolitike čipove Hammer daje slijedeće memorijske karakteristike:

	bita/čipu	atomska debljina
1970	300	200 K
1975	15 K	20 K
1980	500 K	3 K (procjena)
1985	20 M	500 -"
1990	800 M	100 -"

Znači li to da ćemo za manje od 20 god. imati "pravi računar na čipu"?

Predviđanja mogu izgledati optimistička, ali tendencija razvoja je očigledna i nema indikacija da se približava granici rasta.

Tehnološke promjene imaju uticaj na arhitekturu računara. Iako se u osnovnim konceptima arhitekture ne očekuju radikalne promjene u narednih desetak godina, tendencija porasta paralelizma i modularnosti procesora postaću sve izrazitija. Posebno značajno za poslovni svijet je upotreba minija, makroa, multiminija u mrežama, što neosporno vodi stvarnom paralelnom radu i neminovnoj promjeni shvatanja obrade podataka kao serijskog izvršenja instrukcija.

Budućnost software-ske industrije postaje veoma neizvjesna. Da li će ona doživjeti "debakl mehaničkih satova", kako to kaže Hammer, teško je oboriti ili dokazati, ali se sa velikom sigurnošću mogu očekivati hardware-ske realizacije "ne samo kompajlera nego i sistema za upravljanje hijerarhijskim i relacionim bazama podataka, uključujući i operacione sisteme". Visoki troškovi i nezadovoljavajuća pouzdanost software-a doprinijele će sve većoj upotrebi "Chipped software"-a.

Rezimirajući promjene koje informaciona tehnologija bliske budućnosti nosi, Hammer postavlja pitanja njenih implikacija u konkretnim primjenama računara i na širem društvenom planu. Promjene tehnologije, pod cijena hardware-a traže reviziju mnogih konceptata primjene računara i razvoja informacionih sistema, traže novu filozofiju upotrebe računara u svakodnevnim ljudskim aktivnostima. Tehnologija će, kaže Hammer, pružiti mogućnost čovječanstvu da bolje upravlja svojim ograničenim resursima. Da li će čovječanstvo realizovati ovu mogućnost, koju Hammer predviđa, i ostvariti Win-nerovu viziju svijeta "ljudske upotrebe ljudskih bića", zavisi od upotrebe tehnologije, od ciljeva kojima ona bude služila.

Tehnologija gotovo svakodnevno pruža nova dostignuća, otvara nove mogućnosti. Da se u oblasti primjene mnogo sporije napreduje svjedoče prilozi o aplikacijama, tehnikama razvoja sistema i istraživanjima u oblasti informacionih sistema. Danas aktuelni problemi u oblasti informacionih sistema, lociraju se u sferi upotrebe informacija i informacionih sistema u okviru ljudskih sistema obrade informacija. Izbor članka u ovom prvom broju časopisa pokazuje namjeru urednika da na stranicama časopisa reflektuje aktuelne probleme teorije i prakse i relevantne doprinose njihovom rješavanju.

Kritičnost prema važećim konceptima i teorijskim shvatanjima i stvarna otvorenost prema različitim mišljenjima i problemima, nacionalnog i internacionalnog karaktera, za koje se zalažu urednici i u uvodniku i u samom sadržaju prvog broja časopisa, neophodna su pretpostavka da časopis "Information & Management" bude stimulirajući faktor u razvoju oblasti informacionih sistema i ostvari ciljeve koji su mu postavljeni. Ako to, makar i samo djelomično, bude ostvareno, to će opravdati njegovo postojanje, shvatanja i filozofiju koji stoje iza naziva

"Information & Management"

enostavni stabilizirani usmernik za mikro računalnike

Pri uporabi integriranih napetostnih regulatorjev (stabilizatorjev) lahko povečamo tok za dano stabilizirano napetost tako, da dodamo vezju t.i. premostitveni (ojačevalni) močnostni tranzistor in povečamo tok iz stabiliziranega vira za 20-krat ali več. V takšnem vezju želimo zagotoviti naslednje delovne pogoje:

(1) hlajenje premostitvenega tranzistorja T1 naj bo čimbolj izdatno; to dosežemo tako, da spojimo njegov kolektor neposredno s hladilno maso (brez vmesnega sljudnega vložka) in to maso dodatno ohlajujemo (z ventilatorjem);

(2) na premostitvenem tranzistorju T1 moramo imeti pri njegovem največjem dopustnem toku (kolektorskem toku) najnižji dopustni napetostni padec U_{CE} , kar pomeni, da moramo ustrezno dimenzionirati U_{VH} (z upoštevanjem notranje upornosti usmerjevalnega dela);

(3) imeti želimo avtomatično omejitev toka skozi tranzistor T1, kjer tokovna vrednost ne sme preseči največji dopustni tok I_C ;

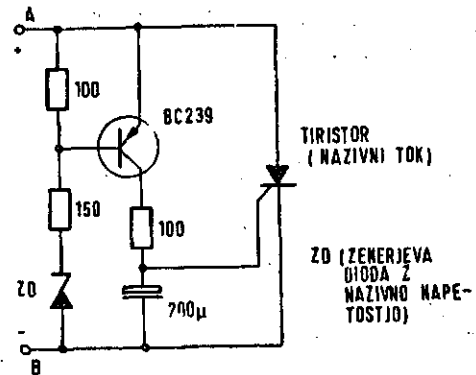
(4) zaradi zaščite računalniških vezij moramo imeti učinkovito napetostno zaščito, ki pri določenem povečanju nazivne napetosti (npr. pri 5,5 V ali 12,5V) sklene izhodni sponki stabiliziranega usmernika;

(5) vezje mora biti sestavljeno tako, da se ne zmanjša faktor stabilizacije integriranega regulatorja (npr. regulatorja tipa 7905 ali 7912);

(6) pri navedenih lastnostih usmernika želimo obdržati njegovo nizko ceno.

Vezje, ki izpolnjuje navedene lastnosti, je narisano na sliki 1, napetostna zaščita pa posebej na sliki 2. Tranzistor T1 je tipa NPN, npr. 2N3055 ($I = 15$ A) 2N3773 (16 A), 2N3442 (10 A) itn. Tranzistor T2 skrbi za tokovno zaščito tranzistorja T1. Ko naraste tok skozi upor R1, se tranzistor T2 odpre in tako zmanjša celokupno upornost v povezavi z uporom R2; s tem se omeji tok I_C skozi T1 napetost na izhodu usmernika pade (zaradi prevelike obremenitve). Tranzistor T2, ki je tipa NPN, mora imeti dopustni tok 1 do 2 A (npr. BC 138, 140, 141, 286 itn.).

Stabilizator ST1 je iz serije 7900, npr. 7905 (za -5 V), 7912 (za -12 V) itn. in ima negativno vhodno in izhodno napetost (nasproti sponki COMMON). Vrednosti C1 in C2 sta 10 μ F.



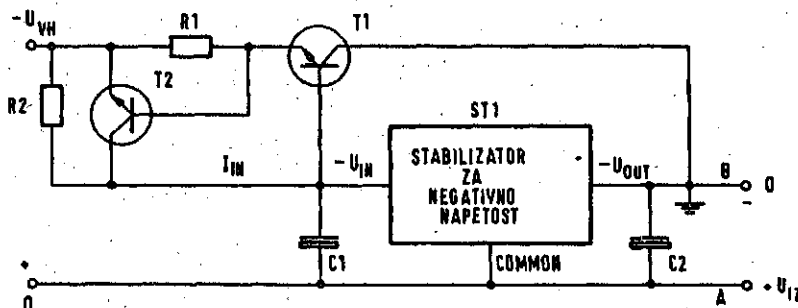
Slika 2. Shema napetostne zaščite za vezje na sliki 1, kjer ima ZD Zenerjevo napetost enako nazivni napetosti usmernika, tiristor pa prevaja nazivni tok (ali večji tok)

Upora R1 in R2 določimo takole: naj bo $I_{IZ} = 15$ A; za T2 imejmo $U_{BE, odprtja} = 0,65$ V; tedaj je $R1 = 0,65/15 = 0,043$ Ohm. Nadalje bodi $I_{IN} = 1$ A in $U_{R2} = 3$ V, torej $R2 = 3/1 = 3$ Ohm. Moč na uporih je $P_{R1} = 9,75$ W in $P_{R2} = 3$ W. Tako imamo empirično $U_{VH} = 9$ do 12 V pri $U_{IZ} = 5$ V. Upornosti R1 in R2 določimo natančneje s poskusom pri danih T1 in T2.

Na sliki 2 vidimo še navadno napetostno zaščito, ki je povezana v točki A in B vezja na sliki 1.

Shema na sliki 1 je ugodna zlasti za pozitivne izhodne napetosti (npr. +5, +12, +15 V itn.), saj so premostitveni tranzistorji T1 tipa NPN dokaj ceneni (pri nas 2N3055) ter jih tudi imamo v maloprodaji. Ker so nazivni tokovi pri negativnih napetostih (-5, -12 V itn.) navadno manjši, lahko vezje na sliki 1 uporabimo tudi v teh primerih, če namesto točke B ozemljimo točko A; v tem primeru izoliramo T1 s sljudno podložko od hladilne mase.

A.P.Železnikar



Slika 1. Vezje stabilizatorja napetosti, ki uporablja premostitveni močnostni tranzistor T1 (npr. 2N3055), katerega kolektor je povezan z maso (zaradi učinkovitejšega hlajenja)

mikroročunalniška integrirana vezja v številkah

317	Kasetni kontroler (NEC)	2914	PIC (Priority Interrupt Controller); kontroler za uravnavanje prekinitev s prednostjo (AMD)
372	Kontroler za floppy disk (NEC)	3000	Družina 2-bitnih mikroprocesorskih gradnikov (Intel)
0400	4-bitni gradnik tipa 12L (TI)	3001	Mikroprogramska krmilna enota (Intel)
464	65536-bitni serijski pomnilnik, organiziran v 16 paralelnih 4K-kanalov, vezje ima 16 pinov (Fairchild)	3002	Element za centralno procesiranje, ki je 2-bitni gradnik (Intel)
481	4-bitni gradnik (TI)	3003	Vnaprejšnji generator prenosa pri soštevanju (Intel)
601	16-bitno integrirano vezje (Data General)	3850	Mikro procesor iz družine F8 Fairchild (Mostek)
1000	4-bitni mikro procesor (TI)	3851	PSU (Program Storage Unit) je dodatno vezje za procesor 3850, ki vsebuje ROM (1k x 8), časovnik, prekinjevalnik ter dodatne vhodno/izhodne kanale (Fairchild, Mostek)
1070	Glej 1000	3852	Vmesnik za dinamični pomnilnik v družini F8 Fairchild, Mostek)
1100	Glej 1000	3853	Vmesnik za statični pomnilnik (SM) v družini F8 (Fairchild, Mostek)
1200	Glej 1000	3854	DMA kontroler za F8 (Fairchild, Mostek)
1270	Glej 1000	3856	Časovnik z V/I vrati in ROMom (Fairchild)
1300	Glej 1000	3857	Časovnik, vmesnik za statični pomnilnik in ROM (Fairchild)
1600	Družina integriranih vezij za emulacijo sistema PDP 11/34 (Western Digital)	3861	PIO s časovnikom za F8 (Fairchild, Mostek)
1702	Ultravijolično izbrisljivi ROM, ki ima 256 besed s po osmimi biti (Intel)	3870	Mikro računalnik z osmimi biti v enem vezju, ki je nadgradnja za 3850, saj vsebuje še 4032-zložni ROM in 128-zložni RAM (Mostek)
1802	8-bitni CMOS mikro procesor (RCA)	3871	PIO s časovnikom (Fairchild)
2102	Statični RAM s 1024 besedami po en bit	3876	8-bitni mikro računalnik v enem vezju, ki je nadgradnja za 3870 in ima 4032-zložni ROM in 256-zložni RAM (Mostek)
2650	8-bitni mikro procesor (Signetics)	3880	Mikro procesor Z-80 (Mostek)
2651	UART za 2650 (Signetics)	4004	4-bitni mikro procesor (Intel)
2652	Integrirano vezje za SDLC (Signetics)	4027	4096 x 1-bitni dinamični RAM, zamenljiv s 16384 x 1-bitnim dinamičnim RAMom tipa 4116 itn. (Mostek)
2655	PIO za 2650 (Signetics)	4040	4-bitni mikro procesor, ki je nadgradnja procesorja 4004 in vsebuje več registrov ter izvaja večje število ukazov (Intel)
2702	Glej 1702	4096	4096 x 1-bitni dinamični RAM, klasičnega tipa (poravnane pozitivne signale fronte signalov RAS- in CAS-)(Mostek)
2708	Ultravijolično izbrisljivi ROM, ki ima 1024 besed s po osmimi biti (Intel)		
2716	Ultravijolično izbrisljivi ROM, ki ima 2048 besed s po osmimi biti (Intel)		
2716	Ultravijolično izbrisljivi ROM, ki ima 2048 besed s po osmimi biti, vendar ni zamenljiv z Intelovim ROMom		
2901	4-bitni mikroprocesorski gradnik (AMD)		
2902	Vnaprejšnji generator prenosa pri reševanju (AMD)		
2903	Izboljšana različica za 2901 (AMD)		
2909	Mikroprogramski sekvenčnik (AMD)		
2911	Mikroprogramski sekvenčnik (AMD)		

4102	Glej 2102 (Mostek)	7400	Družina integriranih vezij tipa TTL(TI)
4104	4096 x 1-bitni statični RAM (Mostek)	8008	8 bitni mikro procesor (Intel)
4116	16384 x 1-bitni dinamični RAM s časom dosega od 150 do 250 ns, zamenljiv s 4k-bitnim RAMom tipa 4027 (Mostek)	8048	8-bitna družina mikro računalnikov v enem integriranem vezju (Intel)
4264	PIO za 4040 (Intel)	8080	8-bitni mikro procesor, ki je nadgradnja procesorja 8008 ter ima različno zalogo ukazov toda podobno arhitekturo kot 8008 (Intel)
4308	ROM (1024 x 8 bitov) ter I/O vrata za 4040 (Intel)	8085	8-bitni mikro procesor, ki je nadgradnja procesorja 8080, združuje funkcije več članov družine 8080 ter ima dva nova ukaza in štiri prekinitevne ravni (Intel)
5701	4-bitni gradnik, predhodnik gradnika 2901 (MMI)	8086	16-bitni, zlogovno usmerjeni mikro procesor, ki je razširitev procesorja 8085, ima razširjeno zalogo ukazov ter možnost 16-bitne aritmetike (Intel)
6100	12-bitni mikro procesor, ki emulira PDP 8 (Intersil)	8212	Paralelni register (latch) in vmesnik družine 8080 (Intel)
6502	8-bitni mikro procesor (MOS Technology)	8224	Taktni generator družine 8080 (Intel)
6503	... itd. Družina 8-bitnih mikro procesorjev (MOS Technology)	8228	Sistemeski kontroler družine 8080 (Intel)
6520	PIO za družino 6500 (MOS Technology)	8251	USART družine 8080 (Intel)
6530	RAM, ROM, I/O in časovnik (MOS Technology)	8253	Programirljivi intervalni časovnik družine 8080 (Intel)
6604	4096-bitni dinamični RAM, 16 nožic (Motorola)	8255	Programirljivi paralelni vmesnik družine 8080 (Intel)
6605	4096-bitni dinamični RAM, 22 nožic (Motorola)	8259	Kontroler prekinitev družine 8080 (Intel)
6701	Podobno kot 5701, le da je ta tip komercialna različica (MMI)	8271	Kontroler za floppy disk družine 8080 (Intel)
6800	8-bitni procesor (Motorola)	8273	Kontroler za sinhrono podatkovne povezave družine 8080 (Intel)
6801	8-bitni mikro računalnik v enem integriranem vezju (Motorola)	8275	CRT kontroler družine 8080 (Intel)
6802	8-bitni mikro računalnik v dveh integriranih vezjih, ki je nadgradnja procesorja 6800, tako da ima funkcije več članov prejšnje družine 6800 (Motorola)	8279	Kontroler za tastaturo in prikaz družine 8080 (Intel)
6809	8-bitna, kvalitetna nadgradnja procesorja 6800, ki ima razširjeno zalogo ukazov ter možnost procesiranja 16-bitnih besed (Motorola)	8708	Glej 2708
6820	PIO za družino 6800 (Motorola, Mostek, Fairchild, Fujitsu)	8748	Procesor 8048 z EPROMom v enem vezju (Intel)
6821	Glej 6820	9080	Procesor 8080 iz drugega vira (AMD)
6828	Kontroler za prednostno prekinjanje - PIC (Motorola)	9400	Bipolarna makrologična družina (Fairchild)
6840	Programirljivi časovnik, s tremi 16-bitnimi kanali, prirejen za družino 6800 (Motorola)	990	16-bitni mikro procesor, ki je združljiv z družino mini računalnikov tipa 990 (TI)
6842	Taktni vmesnik za CPU družine 6800 (Motorola)	9904	Taktnik za družino 9900 (TI)
6843	Kontroler za floppy disk družine 6800 (Motorola)	9940	16-bitni mikro računalnik v enem vezju z EPROMom in RAMom (TI)
6844	Kontroler za neposredni pomnilniški dostop (DMA) družine 6800 (Motorola)	9980	16-bitni, zlogovno usmerjeni mikro procesor (TI)
6845	Kontroler za prikazovanje na katodni elektroniki (CRT) družine 6800 (Motorola)	10696	PID za mikroprocesorsko družino PPS (Rockwell)
6846	ROM, V/I in časovnik družine 6800 (Fairchild)	10706	Taktnik (Rockwell)
6850	UART za družino 6800 (Motorola)	10800	4-bitni gradnik tipa ECL (Emitter-Coupled Logic) (Motorola)
6860	Modem (Motorola, AMI, Fairchild)	10817	DMA kontroler za PPS8 (Rockwell)
6870	Taktnik z omenjenimi funkcijami (taktni generator, ura), (Motorola)	10936	Kontroler za floppy disk pri PPS8 (Rockwell)
6871	Taktnik z vsemi funkcijami (fazami), ki so potrebne družini 6800 (Motorola)	11806	PPS8, 8-bitni mikro procesor (Rockwell)
6875	Taktni generator z vsemi funkcijami, signalnimi derivacijami višjega reda (4f, 2f, β 1, β 2, MR, DMA)	12660	PPS4, 4-bitni mikro procesor (Rockwell)
		57001	Novo ime za 5701 (MMI)
		67110	Mikroprogramski sledilnik (MMI)
		8T26	Vmesnik s tremi stanji za podatkovno vodilo

	(2x4 vmesniki) (Motorola)	F-8	Glej 3850, 3851
8T95	Vmesnik s tremi stanji (šestkratni), z dvojno negirano osposobitvijo (Motorola)	LSI-11	PDP-11 v izvedbi LSI (DEC)
8T96	Vmesnik s tremi stanji in negiranimi izhodi (šestkratni), z dvojno negirano osposobitvijo (Motorola)	PPS-4	4-bitni mikro procesor (Rockwell)
8T97	Vmesnik s tremi stanji (šestkratni) in z ločenima negiranimi osposobitvama (4+2=6) (Motorola)	PPS-8	8-bitni mikro procesor (Rockwell)
8T98	Vmesnik s tremi stanji in negiranimi izhodi (šestkratni) ter z ločalima, negiranimi osposobitvama (4+2=6) (Motorola)	SC/MP	Okrajšava za "Simple Cost Effective Micro-processor". 8-bitni mikro procesor (National Semiconductors)
8X300	Mikro kontroler (Signetics, SMS)	TMS-1000	4-bitna družina mikro računalnikov v enem integriranem vezju (TI)
		Z-80	8-bitni mikro procesor. Nadgradnja procesorja 8080, ki ima dodatne registre ter razširjeno zalogo ukazov (Zilog)

APŽ

knjige

A. Lesea, R. Zaks: Microprocessor Interfacing Techniques, Založba: Sybex-Europe, 313 rue Lecourbe, 75015-Paris, France, cena: 66 FF, leto izdaje: 1977.

Delo je namenjeno inženirjem in praktikom, ki načrtujejo, gradijo in razvijajo mikroročunalniške sisteme; delo ima svoje težišče na vezjih oz. povezavah v samem mikroročunalniškem sistemu ter izven njega, tj. v povezavah s periferijo.

Podobno kot programiranje računalnikov lahko postane večšina, temelječa na izkušnjah in osebni domiselnosti, je tudi računalniško povezovanje, ki zajema načrtovanje in preizkušanje vezij zlasti za periferijo, spretnost, ki zasluži vso pozornost resnega razvijalca. Delo nakazuje metodologijo in sestavne dele, potrebne za zgraditev popolnega mikroročunalniškega sistema, od osnovnih procesorskih enot do sistema z vso navadno periferijo, tj. od tastature do majhnega diska.

Knjiga ima tale poglavja:

1. Uvod: cilji, od umetnosti k tehniki, razmerje materialne in programske opreme, osnovni mikroprocesorski sistem, krmilni signali μP ;

2. Zgradba osrednje procesne enote (CPU): uvod, arhitektura sistema, linearna izbira, v celoti dekodirano naslavljanje, pomnilna integrirana vezja (IV), vmesniki za vodila, sistem z 8080, taktik, sistemski krmilnik, povezava z ROM-om in RAM-om, kompletni sistem z 8080, sistem z 6800, sistem z Z-80, procesor 8085, povzetek;

3. Osnovni vhod in izhod (V/I): Pomnilniški koncept V/I, posebni (ali pravi) koncept V/I, paralelni V/I, PIA 6820, PPI 8255, serijski V/I, programirani serijski V/I (8080), UART in USART, ACIA 6850, USART 8251, metode V/I: programirna, prekinitvena, DMA (direktni pomnilniški dostop);

4. Povezava s periferijo: uvod, tastature: odskakovanje, tastatura brez prekodiranja, tehnika križanja linij, prekodirana tastatura, prekodirnik, pregledovalno IV, tastatura za ASCII; prikazovalniki z LEDi: 7 segmentni LED, matrični LED, pregled prikazovalnikov; priključitev teleprinterja: materialna in programska; čitalnik papirnega traku, linijski tiskalnik, kartični čitalnik, kasetni vmesnik za KIM in za SC/MP (s programom), kasetni

krmilnik v enem IV: NEC UPD 371 D, vmesnik za CRT prikazovalnik: generiranje znakov, pretvorba v serijski video signal, video pomnilnik, kontroler za CRT prikazovanje v enem IV (6845, 8275); majhen disk: pregled, standardi, pogon diska, formatiranje, trdi sektorji, zaznavanje napak in korekcija, preizkušanje s ciklično redundanco, primer kontrolerja SA 4400, kontrolerji FD 1771 D, NEC UPD 372, 6843, 10936;

5. A/D in D/A konverzija: uvod, D/A konverzija; A/D konverzija: vzorčevanje, aproksimacija A/D konverzije, integracija; neposredna primerjava, IV za A/D konverzijo, A/D podatkovni zbirni sistem: povezovanje, primeri, več kanalov, uporaba vezja MP21, povečanje resolucije; 6. Standardi in tehnika vodil: paralelna vodila, vodilo S-100, sistemsko vodilo za 6800, vodilo IEEE-488-1975, CAMAC; serijski standardi: EIA-RS-232-C, RS-422 in 433; asinhronske komunikacije, sinhronske komunikacije, plošča za vodilo S-100: materialna in programska oprema, A/D konverzija;

7. Multiplekserji: uvod, specifikacije, arhitektura, programska oprema, 32-kanalni multiplekser, moduli;

8. Testiranje: uvod, napake sestavnih delov, programska oprema, šum, meritve, dinamični problemi: uporaba osciloskopa, logični analizator, emulacija, signaturna analiza; tehnika testiranja s programsko opremo: primerjalno testiranje, lastna diagnostika, simulacija, algoritmično generiranje vzorcev: testiranje s stalnimi vzorci, testiranje s spremenljivimi vzorci; simulacija in emulacija: v vezju; procesorske napake;

9. Evolucija: tehnološki razvoj, programirljivi vmesniki in povezovalniki, stroški, plastična programska oprema;

Dodatek A: seznam proizvajalcev;

Dodatek B: seznam proizvajalcev enot in seznam enot, ki uporabljajo vodilo S-100;

Indeks.

A. P. Železnikar

PROIZVAJALEC				PROIZVAJALEC			
	ANGLIJA	FRANCIJA	NEMČIJA		ANGLIJA	FRANCIJA	NEMČIJA
AMD AH 2801A AH 5800A	Advanced Micro Devices, Room 322 Ebury Gate, 23 Lower Belgrave Street, London SW1.	Advanced Micro Devices SA, Sila 314, Immeuble Essen, 20 Rue Saarinen, 94588 Rungis Cedex.	Advanced Micro Devices GmbH, 8000 Munich 2, Herzog-Heinrichstr 3.	Motorola MC10800 MC6600 MC6802 MC14100	Motorola Ltd, York House, Empire Way, Wembley, Middlesex.	Motorola Semiconducteurs SA, 15-17 Avenue de Segur, 75326 Paris, Cedex 07.	Motorola GmbH, Heinrich-Hertzstr 1, 6204 Taunusstein-Neuhof 5.
AMI S 5800 S 58A00 S 68B00 S 9900 S 2000	American Microsystems, 108A Commercial Road, Swindon, Wilts. or Distronic Ltd, 50-51 Burnt Mill, Elizabeth Way, Harlow, Essex.	Tekelac Airtronic, BP2 Cite des Bruyeres, Rue Carle Vernet, 92310 Sevres.	Microtec GmbH, Johannesstr 91, D-7000 Stuttgart 1.	Mullard 2650 8X300 M2901-1 M3002	Mullard Ltd, Mullard House, Torrington Place, London WC1.	RTC Laradiotechnique-Complec, 130 Avenue Ledru Rollin, F-75540 Paris 11.	Valvo, US Bauelemente der Philips GmbH Valvo Haus, Burchardstrasse 19, D-2 Hamburg.
Data General Hn 601	Data General Ltd, Westway House, 320 Rustip Road East, Greenford, Middlesex.	Data General France, La Boursiere RN 186, Immeuble M-BP 78.	Data General GmbH, Frankfurter Allee 27, D-6236 Eschborn.	National SC/MP Semiconductor SC/MP11 PACE	National Semiconductor, 19 Goldington Road, Bedford.	National Semiconductor SARL, Expansion 10000, 28 Rue de la Redoute, 92260 Fontenay-aux-Roses.	National Semiconductor GmbH, Cosimarstr 4/1, 8000 Munich 81.
Fairchild F& serija FS440	Fairchild Camera and Instrument Ltd, 230 High Street, Potters Bar, Herts.	Fairchild Semiconducteurs SA, 121 Avenue d'Italie, 75013, Paris.	Fairchild Camera and Instrument (Deutschland) GmbH, 8000 Munich 80, Truderinger Str 13.	NEC uCOM 42 uCOM 43	NEC Electronics (Europe), 43 Civic Square, Motherwell, Scotland.	NEC Electronics (France), 27-29 rue Poissonieres, 92200 Neuilly sur Seine.	NEC Electronics (Europe) GmbH, Karlsruhe 123-127, 4000 Dusseldorf 1.
Ferranti F100-L	Ferranti Ltd, Digital Systems Division, Western Road, Bracknell, Berks.	Volume users supported direct from UK	Ferranti GmbH, Widenmayerstr 5, 8000 Munich 22.	Pelco PPS4/1 PPS4/2 PPS4 PPS8 R 6500	Pelco (Electronics) Ltd, Enterprise House, 93-95 Western Road, Hove, East Sussex.	REA, 9 rue Ernest Cognacq, 92 Levallois-Perret.	System Kontakt, Siemensstrasse 5, 7107 Bad Friedrichshall.
GIM 98A PIC 1650 90070 CP1600	General Instrument Microelectronics, Regency House, 1-4 Warwick Street, London W1.	PEP, 4 Rue Barthelemy, 92120 Montrouge.	General Instrument Deutschland GmbH, Nordendstrasse 3, 6000 Munich 40.	Plessey 16AS	Plessey Microsystems, Water Lane, Towcester, Northamptonshire.	Plessey France SA, 16-20 rue Petrarque, 75016 Paris.	Plessey GmbH, 8000 Munich 40, Motorstrasse 56.
Intel 8021 8748 8048 8035 8049 8039 8080A 8085A 8086	Intel Corporation (UK) Ltd, 4 Between Towns Road, Cowley, Oxford.	Intel Corporation SARL, 5 Place de Vallance, SILIC 223, 94528 Rungis Cedex.	Intel Semiconductor GmbH, Seidstrasse 27, 8000 Munich 2.	RCA CDP 1802	RCA Solid State, Lincoln Way, Windmill Road, Sunbury-on-Thames, Middlesex.	RCA SA, 32 rue Fessart, 92100 Boulogne.	RCA GmbH, Justus-von-Leibig Ring 10, 2018 Quickborn, Hamburg.
Intersil IN 6180	Intersil Inc, 8 Tessa Road, Richfield Trading Estate, Reading, Berks.	Intersil Inc, 3 Rue de Marty, 78000 Versailles.	Spezial-Electronic KG, 8000 Munich, Orlerstr 8.	Rockwell		6lej Pelco	
Memory Devices MH 6701	Memory Devices Ltd, Central Avenue, East Molesey, Surrey.	Monolithic Memories, SILIC 463, 94613 Rungis Cedex.	Monolithic Memories GmbH, 800 Munich 80, Mauerkircherstrasse 4.	Signetics		6lej Mullard	
Monolithic Memories		6lej Memory Devices		Texas TMS9900 Instruments SP89900 9980A 9981 9940E	Texas Instruments Ltd, Manton Lane, Bedford, Beds.	Sesosem 50 rue Jean Pierr Timbaud, 92403 Courbevoie.	Thomson CSF GmbH, Fallstrasse 42, 8000 Munich 70
Mostek MK 3850 MK 3880	Mostek (UK) Ltd, Masons House, 1 Valley Drive, Kingsbury Road, London NW9.	Mostek France SARL, Bat Liege, 1 Place des Etats-Unis, SILIC 217, F-94518 Rungis Cedex.	Mostek GmbH, 7024 Bernhausen, Talstrasse 172.	Thomson SFF 95880	Thomson CSF, Rightway House, Bell Road, Oanes Hill, Basingstoke, Hants.		
				Zilog 280A	Zilog UK Ltd, Nicholson House, Maidenhead, Berks.	A2M, 40 rue des Tilleuls, 92100 Boulogne.	Kontron Electronic GmbH, Oskar von Militar Strasse, 8057 Eching Bei Munich.

literatura in srečanja

Založniška hiša Integrated Computer Systems se poleg svoje založniške dejavnosti ukvarja tudi z organizacijo vočdnevni intenzivnih tečajev s področja računalniških znanosti. Za drugo polovico leta 1978 je pripravila naslednje tečaje:

1. Data Communications - Digital techniques and System Design. Tečaj bo vodil prof. dr. Donald Schilling. (Copenhagen, 5-8 september)

2. Modern Methods of Digital Signal Processing. Tečaj bo vodil prof. William E. Schoff (Copenhagen, 19-22 september in München, 24-27 oktober)

3. Computer Graphics - State of the Art Techniques and Applications. Tečaj bo vodil dr. Stephen R. Levine. (Stockholm, 26-29 september)

4. Microprocessors and Microcomputers (Stockholm, 19 september)

5. Hands-on Microcomputer Programming and Interfacing Workshop (stockholm, 20-22 september). Oba tečaja bodo vodili strokovnjaki tvrdke Integrated Computer System.

Poleg tega tvrdka Integrated Computer System ponuja takoimenovane mikroročunalniške pakete za studij in delo na domu. Vsak paket vsebuje: mikro računalnik z mikroprocesorjem 8080A razen tega pa še naslednje module:

- 1024 Bytni PROM
- 1024 Bytni RAM
- I/O naprava s tremi 8-bitnimi vrati
- DMA
- 8-števlično LED
- tastaturo s 25 tipkami za programe in podatke

In spremeljevalno knjigo. Cena paketa se giblje okrog 300 DM. Za natančnejše informacije pišite na naslov: ICS- Deutschland, ballu Electronic, Burchardplatz 1, D-2000 Hamburg 1, Germany.

VABILO K SODELOVANJU / CALL FOR PAPERS

V skrajšani obliki objavljamo vabilo k sodelovanju mednarodne organizacije za obravnavo podatkov IFIP za leto 1979. Kongres IFIP-a za dežele Evrope bo v Londonu od 25-28 septembra 1979 leta. Kongres EuroIFIP 1979 je osrednje srečanje teoretikov in praktikov s področja obravnavanja podatkov v Evropi med kongresoma IFIP 1974 in IFIP 1983. V vabljenih predavanjih in predavanjih po sekcijah, bodo priznani strokovnjaki predstavili dosežke s področja računalniških znanosti in aktivnosti vseh IFIP-ovih skupin, kot je IFIP Applied Information Processing Group (IAG), IFIP Technical Committees ipd. To skupine bodo prispevale del kongresnega programa.

Deadline dates

15 September 1978 - Receipt of notification of intent
 31 October 1978 - Receipt of submitted paper (4 copies)
 15 January 1979 - Notification of acceptance/rejection
 31 March 1979 - Receipt of final paper typed ready for printing

Programme Areas

(Key words indicate topics, but do not exclude others within the general extent of the Programme Area)

1. Development, operation and management of computer-based Systems
2. Computer-supported administrative processes in business and government
3. Computer control of technical and environmental processes
4. Computer communication and coordination
5. Computer aid to the creative and analytical mind
6. Computer in society
7. The basic tools and fundamental methods to meet users' needs in data processing

Four (4) copies of the paper, complete with illustrative material should be submitted, comparing:

1. Cover page containing:
 - (a) Title of paper
 - (b) Name, country, affiliation and mailing address of the author(s)
 - (c) Programme Area best fitting to paper (only one area to be selected)
 - (d) The following statement: Neither this paper nor any version close to it has been, or is being, offered for publication. If accepted, the paper will be presented orally at Euro IFIP 79 by the author or one of the co-authors.
 This statement should be signed by the author.

2. Abstract page, containing an abstract not exceeding 100 words.
3. Text of paper, typewritten double-spaced (in English).
4. Illustrations (draft or copies of originals - do not send original photographs or artwork).

All pages must show the author's name in the upper left-hand corner and be numbered consecutively with the cover page as page 1. If there is more than one author, the first listed will be the main author. The total length of the paper, including illustrations and references must not exceed 4,000 words (approximately 16 double-spaced typed pages). An illustration should be counted as equivalent to 250 words. Please note the normal time allotted for oral presentation of paper will be 30 minutes including discussion.

Please contact the Programme Committee if you need advice. Notifications of intent, submitted papers and all correspondence should be addressed to:

EuroFIP 79, Programme Committee Secretariat,
IFIP Foundation, Paulus Potterstraat 40,
Amsterdam 1007, The Netherlands.

Your paper will be refereed and if it is accepted you will be requested to re-type in a form suitable for final photo-reproduction in the EuroFIP 79 Proceedings. At this stage, you will be able to make minor improvement or updates to your text; also the Programme Committee reserves the right to make minor editorial changes.

STROKOVNA SREČANJA

LETO 1978

28 avg. - 6 sept., Torremolinos-Málaga, Španija
INTERGOVERNMENTAL CONFERENCE ON STRATEGY AND
POLICIES FOR INFORMATICS
Organizator: IBI, UNESCO
Informacije: IBI Headquarters, POB 10253, 00144 Rome,
Italy.

4-7 sept., Pisa, Italija
FIRST INTERNATIONAL CONFERENCE ON THE AUTOMATIC
ELABORATION OF ART HISTORY DATA AND DOCUMENTS
Organizator: Villa I Tatti, Università degli Studi di Siena,
Scuola Normale Superiore di Pisa, Ministero per i Beni
Culturali e Ambientali, CNUCE
Informacije: First International Conference on the Auto-
matic Elaboration of Art History Data and Documents,
7 Piazza dei Cavalieri, 56100 Pisa, Italy.

4-8 sept., Manila, Filipini
SOUTH EAST ASIA REGIONAL COMPUTER CONFERENCE
1978 (SEARCC 78)
Organizator: Singapore Computer Society
Informacije: Robert Iau, President, Singapore Computer
Society, c/o Central Provident Fund Board, Robinson
Road, Singapore 1, Republic of Singapore.

4-9 sept., Palaiseau, Francija
FIRST MEETING AFCET-SMF ON APPLIED MATHEMATICS
Organizator: AFCET, Mathematics Society of France-SMF
Informacije: Secretariat, Organization Committee, First
Meeting AFCET-SMF, Applied Mathematics Centre, Poly-
technic School, 91128 Palaiseau, France

6-8 sept., Mannheim, ZRN

3RD SYMPOSIUM ON OPERATION RESEARCH
Organizator: Universität Mannheim
Informacije: prof. dr. Steffens, Universität Mannheim,
Schloss - D 6800 Mannheim

5-6 sept., München, ZRN

SECOND INTERNATIONAL SYMPOSIUM ON DATA PROCESSING
IN SOCIAL SECURITY
Organizator: International Social Security Association
Informacije: Dr. V. Velimsky, ISSA Data Processing
Consultative Service, Boulevard de l'Empereur 7,
B-1000 Brussels, Belgium

5-8 sept., Washington, DC, ZDA

COMPCON '78 (fall): COMPUTER COMMUNICATIONS
NETWORKS
Informacije: COMPCON Fall 78, PO Box 639, Silver Spring,
MD 20901, USA.

11-15 sept., London, Velika Britanija

MINI AND MICRO FORUM CONFERENCE
Organizator: Online Conference Limited
Informacije: Jean Seago, Online, Cleveland Road,
Uxbridge UB8 2DD, Middlesex, UK

10-12 sept., Stanford, Kalifornija, ZDA

COMPUTERS IN CARDIOLOGY
Organizator: IEEE Computer Society
Informacije: Mrs. Droni Moo, Conference Coordinator,
Cardiology Division Stanford University School of Medicine,
701, Welch Road, Suite 3303, Palo Alto, CA 94304

12-15 sept., Berlin

THIRD INTERNATIONAL CONGRESS ON ELECTRONIC INFOR-
MATION PROCESSING (IKD)
Organizator: AMK-Berlin, IKD Professional Commission
Informacije: AMK-Berlin, Ausstellungs-Mess-Kongress-
GmbH, Messedamm 22, D-1000 Berlin 19, Federal Re-
public of Germany

13-15 sept., West Berlin, ZRN

FOURTH INTERNATIONAL CONFERENCE ON VERY LARGE
DATA BASES
Informacije: Herbert Weber (Gen. Chm), Hahn-Meitner
Institut, 1 Berlin 39, Postfach 390128, FRG

18-22 sept., Pariz, Francija

CONVENTION INFORMATIQUE 78
Informacije: Convention Informatique, Secrétariat,
6 place de Valois, 75001 Paris, France

19-22 sept., Canterbury, Kent

CONFERENCE ON MICROPROCESSORS IN AUTOMATION AND
COMMUNICATIONS
Organizator: Institution of Electronic and Radio Engineers
Informacije: Conference Secretariat, Institution of Electro-
nic and Radio Engineers, 99 Gower Street, London WC1 6AZ
UK

21-23 sept., Bologna, Italija

INTERNATIONAL CONFERENCE ON INTERACTIVE TECHNI-
QUES IN COMPUTER-AIDED DESIGN
Informacije: Ira Cotton, Institute for Computer Science
and Technology, National Bureau of Standards, Washington,
DC 20234, USA

26-29 sept. Kyoto, Japonska

ICCC-78: FOURTH INTERNATIONAL CONFERENCE ON COMPUTER COMMUNICATIONS

Organizator: ICCC

Informacije: ICCC, Bell-Northern Research, PO Box 3511-Station C, Ottawa, Ontario, Canada K1Y 4H7

27-29 sept., Snowbird, Utah, ZDA

FIFTH DATA COMMUNICATIONS SYMPOSIUM

Organizator: IEEE Computer Society

Informacije: IEEE Computer Society, PO Box 639, Silver Spring, MD 20901, USA

27-29 sept., Wien, Austrija

IMACS-AICA SYMPOSIUM ON SIMULATION OF CONTROL SYSTEM

Organizator: IMACS, AICA

Informacije: University Professor Dr Inge Troch, Institut für Techn. Mathematik, Gusshausstrasse 27-29, A-1040 Vienna, Austria

2-7 oktober, Bled, Jugoslavija

INFORMATICA 78: THIRTEENTH YUGOSLAV INTERNATIONAL SYMPOSIUM ON INFORMATION PROCESSING

Organizator: Informatika, Slovene Computer Society in cooperation with Josef Stefan Institute and Faculty of Electrical Engineering-University of Ljubljana
Informacije: INFORMATICA 78, Institut Jožef Stefan, 61001 Ljubljana, pp. 199, Yugoslavia

10-12 okt., Venezia, Italija

SECOND CONFERENCE OF THE EUROPEAN COOPERATION IN INFORMATICS ON INFORMATION SYSTEMS METHODOLOGY

Informacije: Prof. dr. P. Lockemann, Institut für Informatik II, Universität Karlsruhe, Postfach 6380, D-7500 Karlsruhe 1, Federal Republic of Germany; or Luciano Lippi, IBM Centro Scientifico, Dorsoduro 3228, 30124 Venice, Italy

17-19 okt., München, ZRN

EUROMICRO 78: FOURTH SYMPOSIUM ON MICROPROCESSING AND MICROPROGRAMMING

Informacije: Dr. H. Berndt (Gen. Chm), Siemens AG, Dv WS PZ 1, Postfach 700078, D-8000 Munich 70, FRG

18-20 okt., Philadelphia, Pennsylvania, ZDA

1978 JOINT AUTOMATIC CONTROL CONFERENCE

Informacije: R.C. Stankey, Instrument Society of America, 400 Stanwix Street, Pittsburgh, PA 15222, USA

25-27 okt., København, Danska

COPENHAGEN CONFERENCE ON COMPUTER IMPACT-78

Organizator: Danish IAG

Informacije: CCI-78, DIAG, The Danish EDP-Council, 58 Bredgade, DK-1260 Copenhagen K, Denmark

25-27 oktober, Santa Monica, Kalifornija, ZDA

FOURTH IEEE SYMPOSIUM ON COMPUTER ARITHMETIC

Organizator: IEEE

Informacije: A. Avizienis, Computer Science Department, University of California at Los Angeles, 3731 Boelter Hall, Los Angeles, CA 90024, USA

November, Seattle, Washington, ZDA

IFIP WORKING CONFERENCE: MICROPROGRAMMING FIRMWARE AND RESTRUCTURABLE HARDWARE

Organizator: IFIP TC.2

Informacije: IFIP Secretariat, 3 rue du Marché, CH-1204 Geneva, Switzerland

7-10 nov., Kyoto, Japonska

FOURTH INTERNATIONAL JOINT CONFERENCE ON PATTERN RECOGNITION

Informacije: Prof. T. Sakai, Kyoto University, Department of Information Science, Kyoto, Japan

13-15 nov. Gif-sur-Yvette, Francija

AF CET CONGRESS: INFORMATICS 1978

Organizator: AFCET

Informacije: AFCET, 156 boulevard Péréire, BP 571, 75826 Paris Cédex 17, France

20-22 nov. Milano, Italija

BIAS'78: 15th INTERNATIONAL CONFERENCE ON AUTOMATION AND MICROCOMPUTER

Informacije: Automation and Microcomputer, 15th BIAS International Conference FAST, P.le R. Morandi 2, 20121 Milano, Italia

4-6 dec., Washington, DC, ZDA

1978 ANNUAL CONFERENCE OF THE ASSOCIATION FOR COMPUTING MACHINERY

Organizator: ACM

Informacije: Gerald L. Engel, Dept. of Mathematics and Computing Systems, Old Dominion University, Norfolk, VA 23508, USA, or Dennis M. Conti, Systems and Software Division, National Bureau of Standards, Washington, DC 20234, USA

13 dec., Washington, ZDA

COMPUTER NETWORKING SYMPOSIUM

Organizator: IEEE Computer Society and National Bureau of Standards, Dept. of Commerce

Informacije: Dr. George Cuan, Computer Sciences Corporation, 6565 Arlington Boulevard, Falls Church, Virginia 22046

leto 1979

29-31 jan., San Antonio, ZDA

6th ACM SIGACT-SIGPLAN TRINITY UNIVERSITY SYMPOSIUM OF PROGRAMMING LANGUAGES

Organizator: ACM

Informacije: Prof. Aaron H. Konstam, Dept. of Computing and Information Sciences Trinity University, San Antonio TX 78284, USA.

29-31 jan., Wien, Avstrija

II ASA: FOURTH INTERNATIONAL SYMPOSIUM ON COMPUTER SYSTEM MODELLING AND PERFORMANCE EVALUATION

Organizator: IIASA

Informacije: Dr. A. Butrimenko, IIASA - A. 2361 Laxemburg, Austria

12-14 feb., Paris, Francija

INTERNATIONAL SYMPOSIUM ON FLOW CONTROL IN COMPUTER NETWORKS

Organizator: IRIA, Institut de Recherche d'Informatique et d'Automatique
 Informacije: IRIA, Secrétariat du Symposium, Domaine de Voluceau - Rocquencourt - B.P. 105, 78150 Le Chesnay, France

marec, Toulouse, Francija

IFAC/IFORS CONFERENCE OF AUTOMATICS AND OPERATION RESEARCH TECHNIQUES APPLIED TO LARGE SYSTEMS ANALYSIS AND CONTROL

Organizator: IFAC, IFORS
 Informacije: IFAC Secretariat, c/o EKONO Oy, PO Box 27, SF-00131 Helsinki, Finland

23-27 april, High Tatras, Češkoslovaška

ALGORITHMS '79

Organizator: The Czechoslovak Scientific and Technical Society, The Union of Slovak Mathematicians and Physicists, The Institute of Technical Cybernetics of the Slovak Academy of Sciences, Dept. of Cybernetics-EF of the Slovak Technical University
 Informacije: Czechoslovak Scientific and Technical Society, Central Council, Prague, PO box 20, Czechoslovakia

21-23 may, Ann Arbor, Michigan, ZDA

4th IFIP/IFAC CONFERENCE ON PROGRAMMING RESEARCH AND OPERATIONS LOGISTICS IN ADVANCED MANUFACTURING TECHNOLOGY (PROLAMAT-79)

Organizator: IFIP, IFAC
 Informacije: Society of Manufacturing Engineers, 20502 Ford Road, POB 930, Dearborn, MI 48128, USA

5-7 jun., Barcelona, Španija

CIL 79: CONVENTION INFORMATICA LATINA

Organizator: Asociación de Técnicos de Informática (ATI), Cámara Oficial de Comercio, Industria y Navegación de Barcelona, Centro de Estudios y Asesoramiento Metalúrgico (CEAM), Facultad de Informática de la Universidad Politécnica de Barcelona, Feria Internacional de Barcelona (FIB)
 Informacije: CIL 79, Pza. Comercial 5, Barcelona-3, Spain

10-16 jun., Prague, Češkoslovaška

SECOND IFAC/IFIP SYMPOSIUM ON SOFTWARE FOR COMPUTER CONTROL (SOCOCO 79)

Organizator: IFAC, IFIP
 Informacije: IFAC Secretariat, c/o EKONO Oy, PO Box 27, SF-00131 Helsinki, Finland

29-31 avg., Zürich, Švica

IFAC SYMPOSIUM ON COMPUTER-AIDED DESIGN OF CONTROL SYSTEMS

Organizator: IFAC
 Informacije: IFAC Secretariat, c/o EKONO Oy, PO Box 27, SF-00131 Helsinki, Finland

25-28 sept., London, Velika Britanija

EURO-IFIP 79

Organizator: IFIP
 Informacije: EURO-IFIP 79, Programme Committee Secretariat, IFIP Foundation, Paulus Potterstraat 40, Amsterdam 1007, The Netherlands

17-21 sept., Dusseldorf, ZRN

5th IFAC SYMPOSIUM ON IDENTIFICATION AND SYSTEM PARAMETER ESTIMATION

Organizator: IFAC
 Informacije: IFAC Secretariat, c/o EKONO Oy, PO Box 27, SF-00131 Helsinki, Finland

novice in zanimivosti

Mikroprocesorski terminal za testiranje sistemov, ki temeljijo na procesorju 8080

Tvrška Hewlett-Packard je zgradila mikroprocesorski terminal, ki omogoča razvijanje in testiranje programov, ki so pisani za sisteme, ki temeljijo na procesorju 8080. Programi imajo lahko največjo kapaciteto 44K zlogov. Zbirna programska oprema omogoča uporabniku, da najde, pregleda in zamenja poljubno programsko vrstico, ustavi program med izvajanjem ali sledi programu med izvajanjem. Programi so lahko shranjeni na dveh tračnih enotah, ki sta vgrajeni v terminal. Naslov: Hewlett-Packard 7, Rue du Bois-du-Lain, P.O. Box, 1217 Meyrin 2, Switzerland.

DP

Osciloskop za testiranje

Tvrška Biomation je poslala na tržišče instrument, ki je kombinacija osciloskopa, naprave za primerjalno testiranje in časovnega logičnega analizatorja. Digitalna vezja testiramo z logičnim analizatorjem, analogna vezja pa z osciloskopom. Primerjalno testiranje sloni na množici testnih funkcij, ki so vgrajene v instrument. Naprava je enostavna za uporabo, ima prožilno sondo in eno sondo za vhodni signal. Naslov: Biomation Corp., 4600 Old Iron Side Drive, Santa Clara, CA 50505, USA.

DP

COBOL RØ2

Najavljena je nova verzija COBOLA, RØ2 za računalnike ranga interdata. Kvalitete RØ2 pridejo do izraza predvsem v hitrosti izvajanja operacij, ki so 3 do 150 krat hitrejše kot v sedanji verziji. Tudi v jeziku samem so spremembe; tako so npr. dodana kvalifikacijska imena za podatke in paragrefe in primerjalna polja 18 bitov, ki so združljiva z IBM formatom. Naslov: Parking-Elmer Data Systems 50/56 Rue de la Procession, 75737 Paris Cedex 15, France.

DP

Mikroračunalniški sistem Z-80 z majhnim diskom

AID-80F je zaključen računalnik z diskom (firme Mostek), ki je prirejen za razvoj materialne in programske opreme (vključno za popraviljanje programov). Osrednja enota sistema AID-80F je plošča SDB-80 (Software Development Board), ki ima procesor Z-80 ter 16K zložni pomnilnik. Plošča RAM-80 vsebuje nadaljnjih 16K zlogov pomnilnika ter V/1. Fleksibilni diskovni krmilnik FLP-80 povezuje SDB-80 z največ štirimi diskovnimi enotami (ki imajo spremljivi sektorski format). Dodatna plošča AIM-80 omogoča emulacijo vezij v realnem času s popraviljanjem, sledenjem ter diagnostiko. Programska oprema vključuje: monitor, tekstni urejevalnik, zbirnik, premeščevalni in povezovalni nalogalnik, popraviljalnik ter periferni izmenjevalni program, ki oblikuje neformatirane diskete s spremljivim sektorjem v formate IBM 370 in zapiše sistemsko programsko opremo. Ta program kopira tudi zbirke podatkov iz diska na disk, iz diska na periferno ali iz periferije na periferijo. Cena sistema AID-80F znaša \$5995; posamezne plošče se prodajajo posebej. Naslov proizvajalca je Mostek Corp., 1215 W. Crosby Rd., Carrollton, TX 75006, USA.

APŽ

Intel 8085, ki uporablja jezik PASCAL

Model 85/P je programirni sistem, ki vsebuje: procesor 8085 s 54K zložnim statičnim pomnilnikom; megazložni pomnilnik v dveh majhnih diskovnih enotah, 24 krat 80 znakovni prikazovalnik z visoko ločljivostjo (na zaslonu), dvoje serijskih vrat za tiskalnik in drugi terminal (tiskalnik ali modem). Sistem razpolaga s prevajalnikom in interpreterjem za PASCAL, ki prevede 700 vrstic (stavkov) v minuti ter s programi za obdelavo naključnih in sekvenčnih podatkovnih zbirk ter z interaktivnim popraviljalnikom izvirnega teksta. Cena sistema 85/P je \$7495, naslov proizvajalca pa: Northwest Microcomputer Systems Inc., 121 E. 11th, Eugene, OR 97401, USA.

APŽ

Mikro računalniki v letu 1981

V treh letih bo osebni mikro računalnik sestavljen iz procesorja (en čip), iz videoprikazovalnega in tastaturnega vezja (v enem čipu), krmilnika za majhne diske (en čip), 8 čipov s 64K-bitnim pomnilnikom ter iz 1 čipa operacijskega sistema v ROMu (ta čip bo vseboval monitor, zbirnik, prevajalnik, urejevalnik); skupno število čipov v osebni mikro računalniku bo tedaj 12. Cena takega računalnika z dvema majhnima diskovnimi enotama bo pod \$2000. Število naslovnih linij pri procesorjih se bo povečalo iz današnjih 16 na 23. Intel 8086 zmora že sedaj 20 linij (pomnilnik s preko milijon zlogov), procesor Z 8000 pa bo imel 23 linij (preko 8 milijonov zlogov pom-

nilnika) ter popolnoma novo funkcionalno obliko.

APŽ

Mikro računalnik v šoli

Mikro računalnik predstavlja novo kvaliteto v osnovni in srednji šoli. Konkretna demonstracija računalniške funkcije, šolski razvoj materialne opreme in programiranje omogočajo učencu učinkovit sestop od abstraktnih, verbalnih in lingvističnih konstrukcij k uporabniški empiriji oziroma praksi. Učitelj lahko tako napreduje uspešneje in hitreje, učenci pa spremljajo pouk z osebnim interesom in prizadevno. Tako so npr. v Marycrest College (Deavenport, Iowa) zamenjali terminal za računalnik IBM 360/65 z mikro računalnikom KIM-1. Uspeh je bil presenetljiv: majhen računalnik ni samo izrinil uporabo terminala za veliki računalnik, marveč je odprl vrsto novih možnosti in prednosti, kot so:

1. mikro računalnik se nahaja v razredu in je učencem dostopen;
2. mikro računalnik je majhna naprava in učencev ne strahuje;
3. učenci se z računalnikom temeljito seznanijo, programirajo in sami izvajajo svoje programe in poskuse;
4. finančno šibke šole si lahko mikro računalnik izposodijo;
5. s primerno programsko opremo je mikro računalnik mogoče uporabiti (npr. v konfiguraciji z mehurčnim pomnilnikom) za računalniško podprto učenje (CAI)...

Mikro računalnik predstavlja novo dimenzijo v poučevanju. Pri pouku se programi izvajajo korakoma (ukaz za ukazom) ter se opazujejo stanja registrov, naslovi, operacijski kodni in podatki po izvedbi ukaza. Uporabiti je mogoče tudi voltmetre (digitalne) in osciloskop. Tako postane predmet računalništvo empiričen in učenje kvalitetnejše, polnejše, razumljivejše. Poučevanje programiranja zgubi tako priokus abstraktne matematične teorije in postane tehniško (tehnika programiranja). Takšna metodologija poveča število učencev, ki lahko postanejo dobri programerji in inženirji.

APŽ

Mikroprocesorska proizvodnja v Italiji

Iz novega kataloga komponent podjetja SGS-ATES povzema mo, da to podjetje proizvaja popolni mikroprocesorski družini F8 in Z-80. Komponente so: 3850 (CPU), 3851 (PSU), 3852 (DMI), 3853 (SMI), 3854 (DMA), 3856 (PSU), 3857 (PSU/SMI), 3861 (PIO), 3870 (mikro računalnik), 3871 (PIO) ter Zilog Z-80 (CPU, CTC, PIO, SIO, DMA). Te komponente je mogoče nabaviti v prodajnih centrih podjetja, in sicer: SGS-ATES Componenti Elettronici SpA, 50127 Firenze, Via Giovanni del Pian del Carpi 96/1 ali tudi pri: 20149 Milano, Via Correggio 1/3.

APŽ

raziskovalne naloge prijavljene na rrs v letu 1978

V tej rubriki bomo objavljali kratke povzetke raziskovalnih nalog, ki jih financira Področna raziskovalna skupnost za avtomatiko, računalništvo in informatiko, ki so s področja računalništva in informatike.

Naslov naloge: Metode hevrističnega programiranja in obravnavanja tekstov

Projekt: Informacijski sistemi

Nosilec naloge: I. Bratko, IJS Ljubljana

Program raziskave:

- Razvoj splošnih metod hevrističnega programiranja in metod za računalniško predstavitev znanja v hevrističnih programih.
- Izdelava in implementacija hevrističnih algoritmov za (a) reševanje nekaterih kombinatoričnih problemov iz avtomatskega razvoja digitalnih vezij, (b) za problem prirejanja transportnih kapacitet transportnim zahtevam, (c) za računalniško generiranje kemijskih struktur v kemijskih informacijskih sistemih.
- Modeliranje sintakse slovenskega jezika po transformacijski gramatiki.
- Implementacija ATN analizatorja in prilagoditev za slovenščino.
- Generativni model morfologije slovenskega jezika.
- Dopolnjevanje in vzdrževanje programske opreme za numerično programiranje na računalniku CDC CYBER in PDP-11 (jezikl s področja umetne inteligence, softverska podpora za procesiranje tekstov in robotiko).

Naslov naloge: Računalniško omrežje

Projekt: Informacijski sistemi

Nosilec naloge: T. Kalin, IJS Ljubljana

Program raziskave:

Okvirni program večletne raziskave:

Program dela je opisan v priloženem "Raziskovalnem programu IJS za področje informatike za obdobje 1977-1980", kamor je vključena tudi ta naloga.

- Nadaljevanje zasledovanja razvoja na področju omrežij za preklapljanje paketov.
- Nadaljevanje dela na vmesnem mrežnem jeziku.
- Nadaljevanje dela na multimikroprocesnem sistemu za komunikacijske svrhe in simulacija dela takšnega sistema.
- Raziskava možnosti vključitve obstoječih računalnikov v javno omrežje za prenos podatkov (s poudarkom na IBM sistemih).
- Projektna študija za določitev topologije in hierarhije protokolov v meteorološko-ekološki mreži.

Naslov naloge: Mikroprocesorji v sistemih nadzora in upravljanja - III. del

Projekt: Sistem doljinskega nadzora in upravljanja

Nosilec naloge: B. Stiglic, VTO Elektrotehnika, Visoka Tehniška šola Univerze v Mariboru

Program raziskave:

Ta naloga predstavlja 3. fazo raziskav, s katero bodo raziskave predvidoma zaključene do te mere, da bo nadaljna naloga predvsem razvojna.

V tej fazi raziskav bo olikovana in izdelana preskusna končna postaja, povezana preko sistema prenosa informacij v elektrogospodarstvu v središče upravljanja in nadzora, kjer bo povezana ali s procesnim računalnikom ali mikroprocesorsko centralno postajo. V nadaljevanju dela bomo preverjali funkcionalnost in zanesljivost prenosa informacij, lokalne obdelave in hranjenja informacij in občutljivost na motnje, ki izvirajo iz obratovanja energetskih objektov.

Naslov naloge: Razpoznavanje funkcij z računalniško analizo

Projekt: Individualne naloge

Nosilec naloge: M. Ribarič, IJS Ljubljana

Program raziskave:

Preučevanje metod za kvalitativno analizo funkcijskih odvisnosti eksperimentalnih podatkov s pomočjo dosegljive strokovne literature in s pomočjo stikov z drugimi skupinami doma in v svetu, ki se ukvarjajo s podobnimi problemi. Razvoj novih metod in računalniških programov za njihovo preizkušanje na primerih iz prakse. Izdelava programskega paketa za analizo eksperimentalnih podatkov, ki bo napisan v standardnem jeziku, tako da bo uporabljen na različnih računalnikih. Vzdrževanje programov in konzultacije pri njihovi uporabi, razširjanje programskega paketa na osnovi stikov z uporabniki. Izdelava in razširjanje dokumentacije in priročnikov. Poročanje na domačih in tujih kongresih in seminarjih.

INFORMATICA 78

Bled, 2. - 7. oktober 1978

simpozij

13. jugoslovanski mednarodni simpozij o obravnavanju podatkov

Bled, 2. - 7. oktober 1978

seminarji

izbrana poglavja računalniških znanosti

Bled, 3. - 6. oktober 1978

razstava

računalniška oprema in literatura

Bled, 1. - 7. oktober 1978

organizator

Slovensko društvo INFORMATIKA v sodelovanju z Institutom Jožef Stefan in Fakulteto za elektrotehniko, Ljubljana

roki

20. marec 1978 - zadnji rok za sprejem formularja s prijavo in 2 izvodov razširjenega povzetka

30. april 1978 - pošiljanje rezultatov recenzije in avtorskega kompleta

20. junij 1978 - zadnji rok za sprejem končnega teksta prispevka

nadaljnje informacije:**INFORMATICA 78**

Institut Jožef Stefan, 6001 Ljubljana, pp. 199

telefon : (061) 263 261, telegram : JOSTIN, Ljubljana

telex : 31 296 YU JOSTIN

S tradicionalnim posvetovanjem "Simpozij in seminarji INFORMATICA" slovensko društvo INFORMATIKA nadaljuje aktivnost Zveznega strokovnega odbora za obravnavanje podatkov pri Jugoslovanskem komiteju za ETAN.

To posvetovanje je postalo tako po udeležbi kot po tehtnosti objavljenih del osrednje jugoslovansko srečanje teoretikov in praktikov s področja obravnavanja podatkov.

Lanskomu simpoziju, na katerem je bilo predstavljenih 247 tujih in domačih del, je prisostvovalo 443 strokovnjakov. Zaradi hitrega vzpona znanstvenih in strokovnih računalniških moči v Jugoslaviji in sosednjih deželah pričakujemo, da bo na letošnjem simpoziju in seminarjih sodelovalo še večje število predavateljev in poslušalcev. Tudi letošnji simpozij bo mednarodnega značaja. V vabljenih uvodnih predavanjih in na seminarjih, ki bodo potekali vzporedno s simpozijem, bodo priznani tuji in domači strokovnjaki pregledno predstavili najnovejše dosežke iz izbranih področij računalništva in informatike.

Za naše strokovnjake bo simpozij priložnost, da v teku uradnega dela simpozija in v neformalnih srečanjih podvržejo svoja dognanja kritični oceni priznanih tujih in domačih strokovnjakov. Upravičeno smemo pričakovati, da bo simpozij pomemben prispevek k medsebojnemu povezovanju ter izmenjavi izkušenj na področju računalništva.

Vabimo vas, da se s prispevkom aktivno udeležite letošnjega posvetovanja.

jezik simpozija

Zaradi mednarodnega značaja simpozija vabimo avtorje, da predložijo in predstavijo svoja dela v angleškem jeziku, kar zagotavlja večji krog poslušalcev in bralcev. Seveda bodo v program uvrščena tudi dela, napisana v kateremkoli izmed jugoslovanskih jezikov.

zbornik del

Vsak udeleženec bo prejel zbornik del ob prihodu na simpozij, vsak avtor pa bo poleg tega prejel še pet kopij svojega prispevka.

prijavnina

Za simpozij : 2.000 din

Za seminarje : 500 din (velja za cikel treh seminarjev)

Za študente velja 50 % popust

Avtorji plačajo enak znesek kot drugi udeleženci.

INFORMATICA 78

Bled, October 2 - 7, 1978

Symposium

13th Yugoslav International Symposium on Information Processing

Bled, October 2 - 7, 1978

Seminars

Selected Topics in Computer Science

Bled, October 3 - 6, 1978

Exhibition

Computer Equipment and Literature

Bled, October 1 - 7, 1978

Organizer

INFORMATIKA, Slovene Computer Society in co-operation with Jožef Stefan Institute, and Faculty of Electrical Engineering, University of Ljubljana

Deadlines

March 20, 1978 - submission of the application form and 2 copies of extended summary

April 30, 1978 - mailing out of the summary reviews and author kits

June 20, 1978 - submission of the full text of contribution

Further information**INFORMATICA 78**

Institut Jožef Stefan, 61001 Ljubljana, pp. 199, Yugoslavia

Phone : (061) 263 261, Cable: JOSTIN Ljubljana

Telex : 31 296 YU JOSTIN

The traditional Bled computer conference, "Symposium and Seminars INFORMATICA", carries on its role as the major Yugoslav meeting of professionals in the field of computers and information processing.

Last year, 443 experts from Yugoslavia and abroad attended the meeting and presented 247 papers.

In view of the recent upsurge of computer-related activities in Yugoslavia and in its neighbouring countries, even stronger participation is expected this year.

As in previous years, this will be an international symposium. In their invited papers and at the seminars, eminent foreign experts will present surveys of the latest achievements in selected fields of information science and technology.

The sponsors and the organizing committee believe that INFORMATICA 78 will be a good opportunity for the exchange of ideas and experiences in the field of information processing. As a meeting of people from East and West it will also be an opportunity to further mutual understanding and co-operation.

You are invited to take part in the symposium with submitted papers.

General Information

In view of the international character of the symposium, authors are invited to write and present their contributions in English in order to benefit from broader audiences. However, papers written in any of the Yugoslav languages will also be included in the program.

Proceedings

Participants will receive the proceedings upon arrival. Each author will also receive five copies of his paper.

Registration Fee

For symposium : 2.000 din

For seminars : 500 din (for cycle of 3 seminars)


Student registration will be half price.

Authors pay full registration fee.



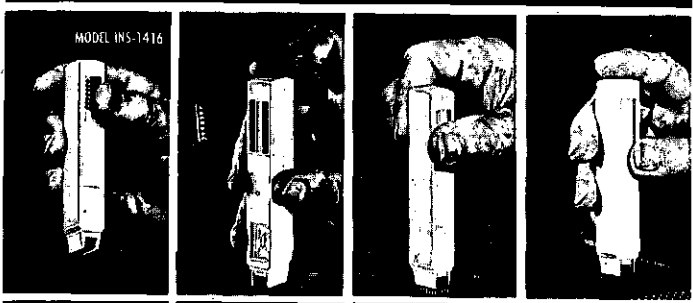
OK MACHINE AND TOOL CORPORATION / 3455 CONNER STREET, BRONX, NEW YORK 10475, U.S.A.

Phone: (212) 994-6600 • Telex: 12-5091 • Telex: 23-2395

IN ELECTRONICS  HAS THE LINE...

DIP/IC INSERTION TOOL WITH PIN STRAIGHTENER

MODEL INS-1416



STRAIGHTEN PINS RELEASE PICK-UP INSERT

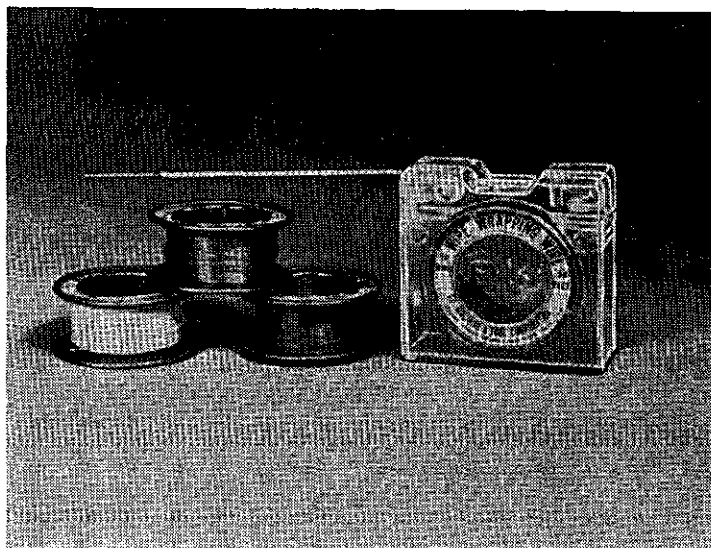
OK MACHINE AND TOOL CORPORATION
3165 CONNER STREET, BRONX, NEW YORK, N.Y. 10475 U.S.A.
 PHONE: (212) 994-6600 • TELEX: 12-5091

INS 1416 je orodje za vstavljanje 16- ali 14-kontaktnih integriranih vezij v podnožja ali izvrtane luknje tiskanega vezja. Posebnost je zoženi profil, ki omogoča vstavljanje vezij, ki so na plošči tesno skupaj. V držalo sta vrezani vodili za ravnanje deformiranih kontaktov integriranega vezja. Vezje potisnemo v vodilo, pri tem se poravnajo deformirani kontakti, izvlečemo pa ga s pritiskom držala navzdol.

TULEC Z ZAMENLJIVIMI KOLUTI ŽICE ZA OVIJANJE

Prednost tulca je v tem, da ne potrebujemo dodatnega orodja za rezanje žice in snemanje izolacije. Iz tulca se izvleče zelena dolžina žice, katero se vstavi v vodilo na vrhu tulca. S pritiskom na vgrajeni gumb, se žico odreže. Potem se žico potegne preko posebnih vgrajenih vilic, ki snemajo izolacijo; isti postopek se ponovi tudi z drugim koncem žice.

Tulec vsebuje kolut s 15 m dolgo žico tipa 30 AWG (0,25 mm), s posebno industrijsko izolacijo in posrebrno bakreno žico. Na razpolago so koluti z belo, modro in rdečo izolacijo.



Ko pišete proizvajalcu, omenite časopis INFORMATICA

nadaljevanje s strani /

5. naloga: Zelo malo pascalskih programov je bilo vsaj za silo pravilnih, nekaj fortranskih pa je bilo solidnih. Večina tekmovalcev po enem letu pouka računalništva še ni sposobna rešiti malo težje (logično zahtevnejše) naloge, večina tistih, ki so nalogo rešili, pa programira zelo nerodno. Primer: za iskanje največjega števila, zložijo vsa števila v polje in šele nato v polju poiščejo največje število. Nikjer ni bilo opaziti pravega strukturiranega postopka. Po tej strani je bila naloga pravo razočaranje.

1978/ po drugem letu pouka

1. naloga: V prvem delu naloge so tekmovalci prišli do prave ideje, niso pa znali sešteti aritmetičnega zaporedja. Drugi del naloge je bilo treba rešiti s postopkom, ki ga je pravilno zapisalo le nekaj tekmovalcev. Večinoma so postopek zapletli in se niso več znašli. Opazimo tudi pomanjkanje sistematičnosti.

2. naloga: Mnogi tekmovalci so zapisali nekaj členov zaporedja in nato (neutemeljeno) sklenili, da funkcija nima ničle. Nekateri so uganili rezultat, vendar ga niso utemeljili. Le zelo malo tekmovalcev je pravilnost rezultata zares dokazalo.

3. naloga: Prav vsi tekmovalci so prišli do ugotovitve, da je treba poskusiti z vsakim avtomobilom. V opisu postopkov so bili premalo natančni in premalo podrobni. Boljši so rezultati tistih, ki so postopek opisovali v slovenščini, kot tistih, ki so to poskušali napraviti z diagramom poteka; v diagramih poteka se učenci vse prehitro spuste na nivo programskega jezika tako, da postane problem zato nepregleden.

4. naloga: Najpogosteje so tekmovalci izbrali primer in na primeru pokazali, kako naj bi njihov postopek deloval. Mnogo jih je primer tako nesrečno izbralo, da je postopek veljal samo za tisti primer. Le izredno redki tekmovalci so algoritem zapisali precizno.

5. naloga: Tekmovalci so bodisi nalogo rešili povsem pravilno ali pa so se izgubili. Mnogi so tudi napisali celo sliko v format stavek. Očitno se pozna razlika med tekmovalci v njihovih praktičnih izkušnjah na stroju.

ZAKLJUČEK

Obe tekmovalni sta bili tako vsebinsko, kot organizacijsko na visokem nivoju. Pokazali sta, da je pouk računalništva naletel na širok odmev v slovenskih srednjih šolah. Opazimo lahko tudi, da je znanje učencev zelo različno (čeprav večinoma na kar zavirljivi ravni - posebej, če ga primerjamo npr. z znanjem študentov računalništva). Različnost znanja je seveda posledica različne učne snovi, ki jo na posameznih srednjih šolah obravnavajo pri računalništvu, in tudi različnega števila ur, ki jih posvečajo temu predmetu.

Na nekaterih šolah sta tako kvaliteta pouka kot njegov obseg kritična, to pa je navadno posledica nerešenih kadrovskih težav in neurejenega dostopa do računalnika za praktične vaje.

Tekmovanja srednješolcev iz računalništva bodo lahko še naprej mesto, kjer lahko primerjamo kvaliteto pouka med šolami in tudi spremembe v kvaliteti pouka, ki bodo posledica rednega izobraževanja učiteljev računalništva, rešenih problemov v zvezi s praktičnimi vajami učencev na računalniku in uveljavitve modernejšega in bolj natančnega učnega načrta, ki je v pripravi.

TABELA 1: Udeležba in uspeh tekmovalcev po šolah.

ŠOLA	1977	1978/1. leto	1978/2. leto
Gimnazija Brežice			3 - 38.67%
Gimnazija Celje	1 - 68.00%		
Ekonomška srednja šola Crnomelj	1 - 13.00%		
Gimnazija Crnomelj	2 - 9.50%	2 - 33.50%	
Gimnazija J. Vege Idrija	4 - 52.75%		2 - 21.00%
Ekonomška srednja šola Izola		3 - 16.67%	
Gimnazija Jesenice	2 - 29.00%		
TSS Jesenice	4 - 21.00%	1 - 25.00%	
Gimnazija Koper		9 - 27.89%	
Gimnazija Kočevje	3 - 52.67%		
Gimnazija I. Cankarja Ljubljana	1 - 56.00%		
I. gimnazija Lj.-Bežigrad	10 - 69.50%	8 - 61.25%	4 - 50.75%
Gimnazija Lj. Sentvid	5 - 48.80%	2 - 15.00%	6 - 20.67%
Gimnazija Lj. Viš	2 - 50.00%	4 - 27.50%	2 - 14.00%
Elektrotehniška šola Ljubljana	2 - 38.00%	5 - 53.80%	
Strojna tehniška šola Ljubljana	2 - 38.00%		
I. gimnazija Maribor	1 - 58.00%		3 - 24.33%
Gimnazija M. Zidanška Maribor	2 - 49.50%	3 - 60.67%	2 - 22.00%
El. in strojna teh. šola Maribor		3 - 58.00%	
Gimnazija Novo mesto	1 - 52.00%	4 - 23.50%	3 - 24.33%
El. in strojna teh. šola Piran		4 - 21.75%	
Ekonomška srednja šola Ptuj		1 - 19.00%	
Gimnazija D. Kvedra Ptuj		1 - 35.00%	
Gimnazija Ravne na Koroškem	2 - 57.00%	1 - 31.00%	1 - 18.00%
Gimnazija Skofja Loka	2 - 39.00%		1 - 27.00%
Gimnazija Trbovlje		1 - 68.00%	

TABELA 3: Priznanja po letih pouka (1977).

	1. leto	2. leto	neznano
1. nagrada	1	4	-
2. nagrada	2	2	1
3. nagrada	1	3	1
pohvala	2	6	2

TABELA 2: Uspeh po nalogah (povprečno število točk od 20 možnih).

	1977	1978/1. leto	1978/2. leto
1.	12.30	11.46	5.89
2.	7.72	6.46	6.12
3.	6.91	10.40	6.19
4.	11.34	8.78	5.56
5.	10.02	5.42	4.19
skupaj	48.30%	42.53%	27.95%
min	3%	0%	0%
max	87%	97%	95%

P r i z n a n i a :

1 9 7 7 :

1. nagrada Luka Sušteršič, I. gimnazija Lj. Bežigrad;
 1. nagrada Dušan Novak, I. gimnazija Lj. Bežigrad;
 1. nagrada Vido Vouk, I. gimnazija Lj. Bežigrad;
 1. nagrada Mirko Skof, Gimnazija Kočevje;
 1. nagrada Peter Reinhardt, I. gimnazija Lj. Bežigrad;

2. nagrada Tomaž Vrhovec, I. gimnazija Lj. Bežigrad;
 2. nagrada Boris Kuntarič, I. gimnazija Lj. Bežigrad;
 2. nagrada Ivan Cibelj, Gimnazija J. Vege Idrija;
 2. nagrada Igor Kononenko, I. gimnazija Lj. Bežigrad;
 2. nagrada Matjaž Nikošč, I. gimnazija Lj. Bežigrad;

3. nagrada Rajko Sveško, Gimnazija Ravne na Koroškem;
 3. nagrada Miran Martinšek, Gimnazija Celje;
 3. nagrada Matjaž Hirsman, Gimnazija Lj. Sentvid;
 3. nagrada Igor Kraševac, Gimnazija M. Zidanška Maribor;
 3. nagrada Goran Devide, I. gimnazija Maribor;

Pohvala Peter Kušar, Gimnazija Lj. Viš;
 Pohvala Bojan Pešek, Gimnazija Lj. Sentvid;
 Pohvala Stanko Vrščaj, Gimnazija Kočevje;
 Pohvala Pajo Čar, Gimnazija I. Cankarja Ljubljana;
 Pohvala Miran Tomljanovič, Gimnazija Novo mesto;
 Pohvala Jure Bkrlj, I. gimnazija Lj. Bežigrad;
 Pohvala Dušan Lapajne, Gimnazija J. Vege Idrija;
 Pohvala Janez Zarnovnik, Gimnazija Lj. Sentvid;
 Pohvala Gorazd Zanoškar, Gimnazija Lj. Sentvid;
 Pohvala Silvij Močnik, Gimnazija J. Vege Idrija;

1 9 7 8 / 1 . l e t o

1. nagrada Marko Ahčan, Gimnazija Lj. Viš;
 1. nagrada Mark Pleško, Gimnazija Lj. Viš;
 1. nagrada Cveto Gregorc, I. gimnazija Lj. Bežigrad;

2. nagrada Zarko Lenardič, Elektrotehniška šola Ljubljana;
 2. nagrada Kazimir Gomilšek, Gimnazija M. Zidanška Maribor;
 2. nagrada Matjaž Ovek, Gimnazija Lj. Viš;

3. nagrada Rudi Ponikvar, Gimnazija Lj. Viš;
 3. nagrada Roman Blenkuš, I. gimnazija Lj. Bežigrad;
 3. nagrada Darko Janžet, I. gimnazija Lj. Bežigrad;

1 9 7 8 / 2 . l e t o

1. nagrada Igor Grličarev, I. gimnazija Lj. Bežigrad;
 1. nagrada Janez Setina, Gimnazija Lj. Sentvid;
 1. nagrada Vido Vouk, I. gimnazija Lj. Bežigrad;

2. nagrada Metod Saje, Gimnazija Novo mesto;
 2. nagrada Goran Salamun, Gimnazija Brežice;
 2. nagrada Andraž Rumpret, Gimnazija Brežice;

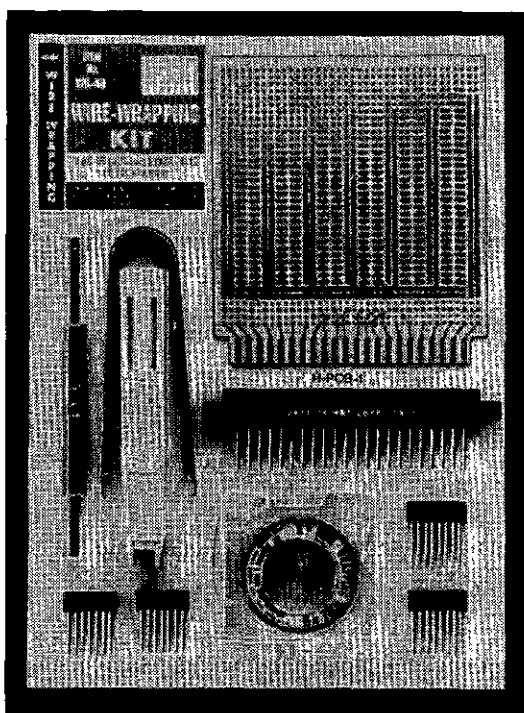
3. nagrada Gorazd Divjak, Gimnazija Brežice;
 3. nagrada Igor Kraševac, Gimnazija M. Zidanška;
 3. nagrada Maja Moran, I. gimnazija Maribor;



OK MACHINE AND TOOL CORPORATION / 3455 CONNER STREET, BRONX, NEW YORK 10475, U.S.A.

Phone: (212) 994-6600 - Telex: 12 5091 - Telex: 23 2395

WK-4B WIRE-WRAPPING KIT

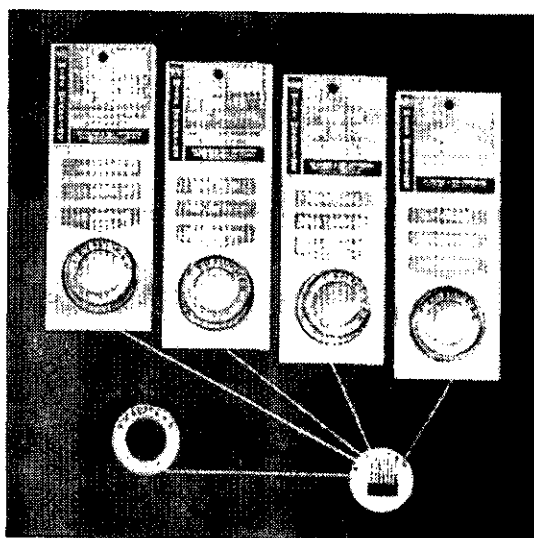


OK MACHINE & TOOL CORPORATION
3455 CONNER ST., BRONX, N.Y. 10475 U.S.A.
TELEX 125091

WK-4B OŽIČEVALNA SESTAVLJENKA

Sestavljenka vsebuje orodje in dele, ki so potrebni za izdelavo prototipne ali amaterske ploščice. Deli, ki jih ima sestavljenka, so: univerzalna plošča s tiskanim vezjem, standardni 2 x 22-polni konektor z izvodi za ožičevanje, dve 14-in dve 16- kontaktni podnožji za integrirana vezja z izvodi za ožičevanje, orodje za vstavljanje in izvlečenje integriranih vezij, tulec s 15 m žice in posebno orodje WSU-30, ki je kombinacija orodja za ožičevanje in odvijanje žice na trnih s premerom 0,63 mm; v ročaju je vgrajeno rezilo za snemanje izolacije.

WIRE-WRAPPING WIRE



OK MACHINE AND TOOL CORPORATION
3455 CONNER STREET, BRONX, N.Y. 10475 U.S.A. PHONE: (212) 994-6600
TELEX NO: 125091 TELEX NO: 232395

ŽICA ZA OVIJANJE

Žica ima najvišjo industrijsko kvaliteto z oznako AWG 30 (0,25 mm), ki je navita v 15 m zvitkih. Žica je primerna za manjše proizvodne serije, razvojna dela, izdelavo prototipov ali za amaterske projekte. Žica je prevlečena s plastjo srebra in je izolirana s posebno plastjo, ki prenese velike mehanske in električne obremenitve.

Na razpolago so štiri barve izolacije: bela, modra, rumsena in rdeča. Žica je navita na 40 mm kolutih, ki omogočajo boljše rokovanje in skladiščenje.

POJMOVI I SKRAĆENICE NA PODRUČJU RAČUNARSKE TEHNIKE

- RAM, Random access memory - memorija sa slučajnim pristupom.
- Random access (slučajni pristup) - pristup do podataka u memoriji gdje adresa sljedećeg podatka ne zavisi od trenutne adrese.
- ROM, Read only memory - tip memorije koja ne dozvoljava upisivanje podataka.
- Read operations, operacija čitanja - događaj kada računar ili druga naprava čita podatak iz drugog računara ili naprave preko vodiča.
- Real-time clock - časovnik realnog vremena - naprava koja slijedi realnom vremenu i pruža mogućnost učitavanja realnog vremena u mašinu.
- Record, zapis - podatkovna jedinica, obično sastavljena od niza znakova.
- Recursive - ponavljajući se proces kod kojeg je rezultat svakog koraka odvisan od rezultata prethodnog koraka.
- Redundancy - upotreba posebnih znakova ili bitova uz podatak u cilju kontrole grešaka i njihove korekcije (parnost)
- Re-entrant, ponovljeni ulaz - osobina rutine da omogući prekid od strane druge rutine koja sadrži poziv prekinute rutine.
- Registar, spremnik - memorijska naprava koja može sadržati jedan ili više bitova ili riječi. Registar se upotrebljava kao privremeno memorijsko područje unutar centralnog procesora.
- Relative addressing - tehnika adresiranja kod koje se stvarna adresa dobije sabiranjem vrijednosti u odgovarajućem registru i sadržaja adresnog polja naredbe.
- Relocatable - rutina kod koje su instrukcije i memorijski pozivi definirani tako da mogu biti postavljeni i izvršeni u bilo kojem dijelu glavne memorije.
- Reset - sadržaj registara ili memorije se vrati na specifične početne uslove.
- Scratch area, zapiske - lokacije, u programu, upotrebljene za privremeno pamćenje podataka, međurezultata.
- Scratch pad - memorijsko područje upotrijebljeno za pamćenje lokacija prekinutog programa te za omogućavanje regularnog nastavljanja programe po izvršenom servisu. To može biti dio glavne memorije ili neodvisna memorija.
- Segmentation, segmentiranje - tehnika dijeljenja velikih programa, koji ne mogu biti u cijelosti naloženi u glavnu memoriju, u manje programe od kojih je uvijek samo jedan u memoriji, a poziva druge sa vanjske memorije. Ta tehnika je poznata također kao overlaying.
- Select code - koda koja odabire perifernu napravu na koju je podatak adresiran.
- Sense instruction - računarska instrukcija koja testira stanje naprave (busy, ready itd.).
- SERIAL - način rada pri kojem se bitovi obrađuju jedan za drugim.
- Serial parallel conversion - konverzija podataka iz serijskog niza u paralelni oblik.
- Service routine - rutina namijenjena za izvršavanje specificiranih I/O operacija ili rutina koju aktivira prekid programa (interrupt).
- Shift - pomijeranje, na lijevo ili na desno, sadržaja registra.
- Simulate - imitacija djelovanja naprave, sistema ili procesa.
- Single shot - metoda djelovanja računara u ručnom režimu gdje se jedna instrukcija izvrši kao odgovor na jednu ručnu akciju.
- Source program - program napisan u jeziku različitom od mašinskog koda.
- Stack - niz memorijskih lokacija gdje se podatak upisuje u lokaciju koja je iznad vrha tog niza, postavši sama vrh niza. Podatak se isključuje iz niza tako da sljedeća niza lokacija postaje vrh.
- Stand alone - računarski sistem ili periferna naprava koja djeluje neodvisno od neke druge naprave.
- Status - odnosi se na jedan od nekoliko mogućih statusa naprave ili programa.
- String - konsekventni niz znakova ili podatkovnih riječi.
- Syntax - skup pravila koja definiraju konstrukciju programskog jezika.
- Task - grupa instrukcija sa specifičnim zadatkom.
- Teletype - Periferna jedinica male brzine, koju čine tastatura, printer, čitač i bušać papirne trake.
- Terminal - periferna jedinica, obično teletype, povezana sa udaljenim računarom.
- Three address - instrukcijski format koji ima specificirane lokacije dvoju operandata te lokaciju za rezultat.
- Time sharing - tehnika koja omogućava upotrebu računara istovremeno od strane više terminala.
- Trace - rutina koja prati izvršavanje programa te ispisuje informacije koje se odnose na izvršavanje pojedine naredbe. Time je omogućeno lakše otkrivanje grešaka.
- Transfer Rate - brzina kojom se bitovi, znaci ili riječi prenose iz jedne naprave na drugu.
- Truth table - tabelarična metoda predstavljanja rezultata buleve operacije.



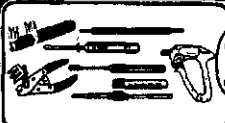



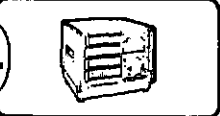
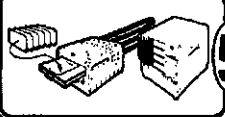
- TTL, Transistor-Transistor Logic
- Two address - Instrukcijski format koji ima specificiranu adresu jednog operanda te adresu za rezultat.
- Two's complement - negiranje binarnog broja tako da kompletiramo svaki bit i nato dodamo 1.
- Unibus - kombinirano I/O i memorijsko vodilo koje sreće-mo kod PDP 11 serije.
- Utility - programi pisani sa namjenom da "pomažu" pri programiranju, diagnosticiranju, upotrebi diska i slično.
- VDU - Visual Display Unit
- Vectored interrupt - prekid koji je automatski identificiran, a kontrola prenešena na rutinu specifičnu za napravu koja vrši prekid.
- Virtual memory - memorijski pozivi u programu se pretva-raj u apsolutne adrese preko specijalnog hardwarea.
- Volatile (store) - memorija koja izgubi svoju sadržinu čim se prekine dotak energije.
- Word orientated - če je riječ osnovni element koji se adresira u memoriji kažemo da je računar word oriented.

Pregled pripremio

M. Kovačević

U ovom broju završavamo kraći pregled pojmova i skraćenica na području računarske tehnike. Pregled je predstavljen sa ciljem da se doprinese širenju "računarske kulture" kod čita-laca drugih zanimanja te da se predloži unifikacija tumače-nja pojedinih pojmova sa područja računarske tehnike. Uredništvo poziva čitaoce da se svojim prilozima uključi u našu akciju.

Uredništvo

 <p>INDUSTRIAL WIRE WRAPPING TOOLS</p>	<p>IN WIRE-WRAPPING  HAS THE LINE..</p>  <p>1</p> <p>MANUAL WIRE WRAPPING TOOLS</p>	 <p>INDUSTRIAL WIRE WRAPPING TOOLS</p>
	<p>ELECTRIC & PNEUMATIC WIRE-WRAPPING TOOLS</p> <p>2</p> 	
	 <p>3</p> <p>SEMI-AUTOMATIC WIRE WRAPPING SYSTEMS</p>	
	<p>SELF-PROGRAMMING CONTINUITY TESTING SYSTEMS</p> <p>4</p> 	
	 <p>5</p> <p>DATAMASTER PROBING FIXTURES</p>	
<p>OK MACHINE & TOOL CORPORATION</p>		

AVTORJI IN SODELAVCI

Rudolf MURN (1930), diplomiral leta 1957 na Fakulteti za elektrotehniko v Ljubljani. Magistriral leta 1966 in doktoriral leta 1974 s področja teorije diagnostike večkratnih napak v digitalnih vezjih. Leta 1976 je prejel skupaj z dvema sodelavcema nagrado Sklada Borisa Kidriča. Je redni sodelavec Instituta "Jožef Stefan", dela na raziskovalnih in aplikativnih nalogah na področju simulacije in testiranja ter načrtovanja digitalnih sistemov. V časopisu Informatica opravlja funkcijo tehničnega urednika.

Franc KOŠIR (1944), diplomiral leta 1967 na Pravni fakulteti v Ljubljani. Zaposlen v Regionalni zdravstveni skupnosti Ljubljana kot vodja centra za avtomatsko obdelavo podatkov od leta 1971 dalje. Poleg organizacijsko operativnih opravil se ukvarja predvsem z načrtovanjem in realizacijo informacijskih sistemov v zdravstvu in aplikativne uporabe računalniške tehnologije v zdravstveni informatiki.

Janez ERŽEN (1948), diplomiral leta 1972 na Fakulteti za elektrotehniko v Ljubljani. Leta 1977 je tam tudi magistriral. Zaposlen je na Železniškem gospodarstvu Ljubljana - Prometni institut. Do sedaj se je ukvarjal z računalniško simulacijo vleke in železniškega prometa, sedaj pa opravlja raziskave s področja zanesljivosti železniških signalno-varnostnih naprav in varnosti železniškega prometa.

Dušan PEČEK (1952), diplomiral leta 1977 na Fakulteti za elektrotehniko v Ljubljani, smer računalništvo in informatika, s tematiko iz področja simulacija digitalnih sistemov. Zapošlen je na Institutu "Jožef Stefan" v Ljubljani, Oddelek za elektroniko. Ukvarja se s problematiko simuliranja poljubnih digitalnih struktur. Iz tega področja je objavil več del v jugoslovanskih časopisih.

CENIK OGLASOV

Ovitek - notranja stran (za letnik 1977)
2 stran ----- 16.000 din
3 stran ----- 12.000 din

Vmesne strani (za letnik 1977)
1/1 stran ----- 8.000 din
1/2 strani ----- 5.000 din

Vmesne strani (za posamezno številko)
1/1 stran ----- 3.000 din
1/2 strani ----- 2.000 din

Oglas o potrebah po kadrih (za posamezno številko)
----- 1.000 din

Razen oglasov v klasični obliki so zaželjene tudi krajše poslovne, strokovne in propagandne informacije in članki. Cena objave tovrstnega materiala se bo določala sporazumno.

ADVERTIZING RATES

Cover page (for all issues of 1977)
2nd page ----- 16.000 din
3rd page ----- 12.000 din

Inside pages (for all issues of 1977)
1/1 page ----- 8.000 din
1/2 page ----- 5.000 din

Inside pages (individual issues)
1/1 page ----- 3.000 din
1/2 page ----- 2.000 din

Rates for classified advertizing:
each ad ----- 1.000 din

In addition to advertisements, we welcome short business or product news, notes and articles. The related charges are negotiable.



**"HOBBY"
WIRE
WRAPPING
TOOLS**

ok wire wrapping center ok

<p>ANOTHER UNIQUE PRODUCT DESIGNED, MANUFACTURED AND MARKETED WORLDWIDE BY OK MACHINE & TOOL CORPORATION</p>	<p>1 </p>
<p>2 </p>	<p>ANOTHER UNIQUE PRODUCT DESIGNED, MANUFACTURED AND MARKETED WORLDWIDE BY OK MACHINE & TOOL CORPORATION</p>
<p>3 </p>	<p>4 </p>
<p>5 </p>	<p>WHAT'S? NEXT</p>

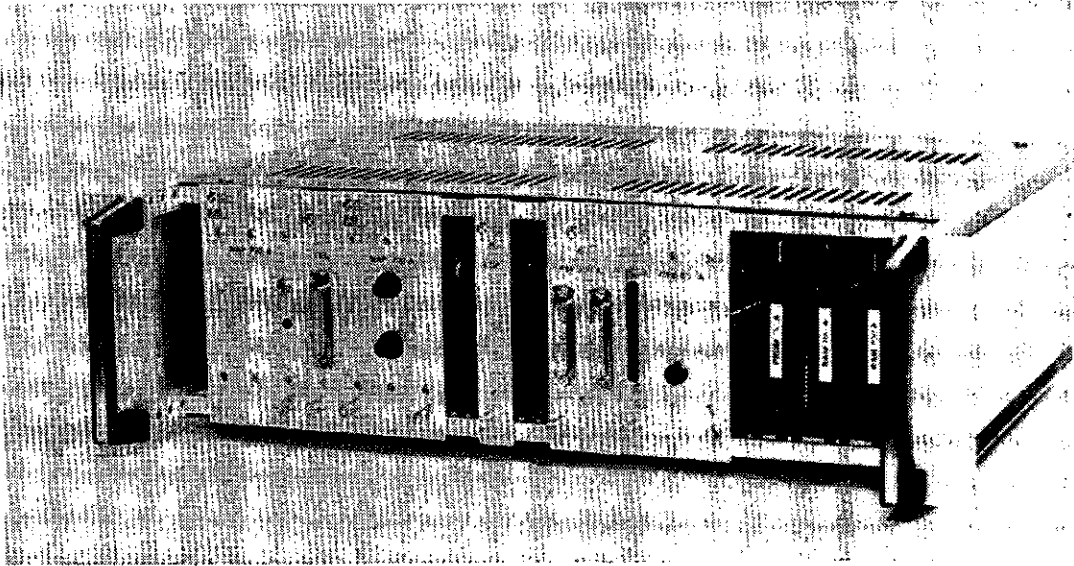
OK MACHINE & TOOL CORPORATION



**"HOBBY"
WIRE
WRAPPING
TOOLS**

●● modularni sistem instrumentacije mikro - m na osnovi mikroročunalnika

Vaš problem avtomatizacije upravljanja procesov in zbiranja podatkov lahko hitro in ekonomično rešite z modularnim sistemom instrumentacije "mikro-m". Iz modulov, ki smo jih razvili in preizkusili, lahko sami ali z našo pomočjo sestavite instrument, ki ustreza vašim zahtevam.



VHODNO/IZHODNE ENOTE:

- PROM-programator za EPROM 1702A (PRP-702)
- programirana ura (PKU-600)
- paralelni vhod/izhod s prekinitvenimi linijami, ojačanimi izhodi ali optično izolirani vhodi/izhodi (PVI-200, PVI-201, PVI-202)
- serijski vhod/izhod, 20 mA zanka, optično izoliran (SVI-220, SOE-250, TEL-001, TEL-010)
- A/D pretvornik, točnost 8 bitov, preklon na sistemsko analogno vodilo (ADP-100)
- izbiralnik analognih kanalov z ojačevalnikom (IKA-400, IKA-410)
- mostični ojačevalnik (MOS-300)

PROCESOR IN SPOMINSKE ENOTE:

- centralna procesna enota s procesorjem tipa 8080 A, 1 K RAM-a, 256 besed PROM-a, prekinitveno strukturo in 14 pozivnimi vodi (CPE-800, CPE 800/1, CPE 800/2)
- aktivni spomin (RAM), statični, 1 K ali 8 K (RAM-700, RAM-701)
- modul za mrtvi spomin (ROM), prostor za 2 K ali 16 K EPROM-a (ROM 750, ROM 751)

OHIŠJE IN NAPAVALNIKI:

- 19" ohišje po DIN 41494, kompletno z vtičnicami in sistemskimi povezavami (OHI-504)
- napajalnik, katerega moč omogoča polno zasedbo ohišja (NAP-700).

Na razpolago so tudi naslednji SISTEMI:

- modularni mikroročunalnik MKR-80: vsebuje enote in programsko opremo za vpisovanje programov s pisalnika v aktivni spomin in izvajanje programov. Možna je razširitev z vsemi vhodno/izhodnimi enotami.
- univerzalni digitalni programator UDP-80: omogoča razvoj in izvajanje funkcij digitalnega avtomata. Osnovne funkcije digitalnega avtomata so razširjene z vgrajenimi programskimi časovniki in števcji, ki so nastavljivi s kodirnimi stikali.
- sistem za izravnavanje konic električne energije SIK-80: izravnava porabo električne energije tako, da v obdobjih velike porabe programirano izklaplja razpoložljiva bremena.

Primeri realiziranih sistemov: avtomatizacija stiskalnice, avtomatizacija reaktorja, mobilni sistem za toplotne meritve in drugi.

INSTITUT "JOŽEF STEFAN", 61000 Ljubljana, Janova 39
 ODSEK ZA REAKTORSKO IN PROCESNO TEHNIKO
 Telefon: (061) 313-022 int. 35

mikroračunalniki

**STE SE ŽE ODLOČILI ZA UPORABO MIKRORAČUNALNIKA V PROIZVODNJI,
ALI PA ŽE IMATE MIKRORAČUNALNIK IN ŽELITE IZVESTI REŠITVE PO
VAŠI ZAMISLI?**

NUDIMO VAM:

- NAJSODOBNEJŠE TEHNIČNE REŠITVE Z UPORABO NOVE TEHNOLOGIJE**
- PROJEKTIRANJE VEČJIH SISTEMOV (MULTIPROCESORSKIH)**
- DIAGNOSTICIRANJE PROCESOV**
- IZDELAVO IN TESTIRANJE UPORABNIŠKIH PROGRAMOV S POMOČJO
VELIKIH SISTEMOV**
- KONZULTACIJE ZA UPORABO MIKRORAČUNALNIŠKIH SISTEMOV V
INDUSTRIJI IN GOSPODARSTVU**
- IMAMO IZKUŠNJE Z UPORABO MIKROPROCESORJEV (Z 80, 6800,
F-8, 8080, 2650, PFL 16, SC/MP), DINAMIČNIH POMNILNIKOV, PERI-
FERIJE ITN.**

**Institut Jožef Stefan, Ljubljana, Jamova 39
ODSEK ZA RAČUNALNIŠTVO IN INFORMATIKO
Telefon (061)63-261 Int. 305**

MIKRORAČUNALNIKI