

# MOJ MIKRO

november 1986 št. 11 / letnik 2 / cena 400 din

## **Priloga:**

*Učimo se programirati MC 68000  
in njegove bližje sorodnike*

## **Tiskalniki:**

NEC PINWRITER P6  
BROTHER M-1509

## **Urejanje teksta:**

Besedila in slike  
končno skupaj

WordStar,  
prednosti  
in težave



## **Programski jeziki:**

GfA basic za atari ST  
Beta Basic 3.0

## **Hardverski nasveti:**

C 64  
kot voltmeter  
Zamenjajmo  
ROM  
z EPROM



**Računalniška animacija**

# ORION

emona commerce  
**tozd globus**  
Ljubljana, Šmartinska 130

## Video kasetnik (player) VP-200

Idealen aparat za tiste, ki že imajo video rekorder, vendar želijo z dodatnim aparatom presnemavati video kasete. Ta aparat ima iste funkcije kot video rekorder, le snemanje z njim ni možno. Majhne dimenzije, priročna uporaba, preprosti priključki (2 cinch RCA) in kakovost!



**IDEALNA KOMBINACIJA!**



## TV 2142 RC

Barvni TV sprejemnik z diagonalo zaslonu 42 cm, daljinsko upravljanje, 16 prednastavitvev, kabelski tuner, idealen aparat tudi za tiste, ki ga želijo prenašati, dodatna teleskopska antena in vdelan ročaj ga uvrščata tudi med prenosne televizorje



### Prodajna mesta:

NOVO MESTO: Emona Dolenjka, Kidričev trg 1, 068 22-395  
ZAGREB: Emona Commerce, Prilaz JNA 8, 041 430-132  
REKA: Emona Commerce, F. Supila 2, 051 36-570  
BEOGRAD: Muzička robna kuća Pro musica, Čika Ljubina 12, 011 634-022, 634-699  
SARAJEVO: Foto-Optik, JNA 50, 071 24-491  
SKOPJE: Centromerkur, Leninova 29, 091 211-157  
ČAKOVEC: Robna kuća Međimurka, Trg republike 6, 042 811-111 interna 213

ISP  
Ljubljana, Titova 21  
061/324-786, 326-677

## Vsebina

<b>Računalniška animacija</b> Računalnik, novo orodje filmskih delavcev	4
<b>Tiskalniki</b> Od plastičnega dirkača do matričnega orkestra	6
<b>Mikropanorama</b> Z računalnikom proti zahrbtnim vetrovom	8
<b>Informacijski sistemi</b> Lokalne računalniške mreže	10
<b>Urojevalniki teksta</b> WordStar: prednosti, težave in izzivi	18
Besedilo in slike: končno skupaj	20
<b>Programski jeziki</b> GIA basic	22
Beta Basic 3.0	24
<b>Hardverski navodi</b> C 64 kot voltmeter	26
Zamenjava ROM z EPROM	28
<b>Numerične metode</b> Matrike (2)	40
<b>Rubrike</b> Mimo zaslona	14
Priloga	29
Mali oglasi	44
Vaš mikro	53
Recenzije	56
Nagradna uganaka	68
Pomagajte, drugovi	90
Igre	61

MOJ MIKRO izdaja in tiska ČGP DELO, Izd. Revije Titova 35, Ljubljana • Prizidevniki skupščine ČGP Delo JAK KO-PRIVC • Glavni urednik ČGP Delo BOŽO KOVAČ • Direktor Izd. Revije BERNARDA RAKOVČEC • Nenaročnega gradiva ne vračamo • MOJ MIKRO je opremljen plačilnega posebnega davka po mnenju republiškega komisija za informiranje, dopis št. 421-1/72 z dne 25. 5. 1984.

Glavni in odgovorni urednik revije Moj mikro VILKO NOVAK • Namestnik glavnega in odgovornega urednika ALUČOŠA VREČAR • Štirokovna urednika CIRIL KRAŠEVEC in ŽIGA TURK • Poslovni sekretar FRANC LOGONDER • Tajnica ELICA POTOČNIK • Oblikovanje in tehnično urejanje ANDREJ MAVŠAR, FRANCI MIHEVC • Redni zunanji sodelavci: ZVONIMIR MAKOVEC, JURE SKVARC, ROBERT SRAKA.

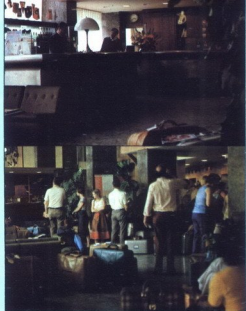
Izdajateljski svetovi: Alenka MIŠČ (Dospodarska zbornica Slovenije, praedraščina, Ciri BEZLAJ (Gorenje – Procesa na oprema, Titova Velenje), prof. dr. Ivan BRATKO (Fakulteta za elektrotehniko, Ljubljana), prof. Aleksander ČOKAN (Dolžana založba Slovenije, Ljubljana), Borislav HADŽIABIĆ (Ivo Lota Ribar, Beograd Zateznik), Miroslav KEK (IK ZSM), inž. Miroslav KOBIL (Iskra, Ljubljana), dr. Beno LUKMAN (S GIS), mag. Ivan GERLIČ (Zveza organizacij za tehniško kulturo, Ljubljana), Tone POLJENEC (Mladinska knjiga, Ljubljana), dr. Marjan SPEGL (Inštitut Jožef Stefan, Ljubljana), Zoran ŠTRISBAČ (Mikrohit, Ljubljana).

Nastavni uredništvo: Moj mikro, Ljubljana, Titova 35, telefon h. c. 315-366, 319-798, telex 31-255 YU DELO • Oglasi: STIK, oglasno trženje, Ljubljana, Titova 35, telefon 318-570 • Prodaja in naročnine: Ljubljana, Titova 35, telefon h. c. 315-366.

Plačila na širo račun: ČGP Delo, Izd. Revije, za Moj mikro, 50102-603-48914.

## VAŠ DELOVNI ČAS JE DRAGOČEN

### NE ZAPRAVLJAJTE GA S SEŠTEVANJEM UR NA ŽIGOSNIH KARTICAH



Na Odsedu za računalništvo in informatiko INSTITUTA JOŽEF STEFAN vam skupaj z GORENJEM iz Titovega Velenja ponujamo:

- namesto žigosnih kartic magnetne kartice;
- namesto ur za žigosanje mreže elektronskih postaj za registracijo;
- namesto »ročnega« seštevanja minut sproten obračun delovnega časa in vrsto urejenih izpisov.

Zakaj je ta sistem zanimiv za vas? Zato, ker je tehnična novost? Ne. Zato, ker je sistem žigosnih kartic tako drag, da si ga bomo vedno težje privoščili. Je drag zaradi visoke cene naprav? Ne. Zaradi izgubljenih delovnih ur pri računanju podatkov na karticah.

Zato prepustite računanje računalniku!

Postopek registracije je preprost: pri prihodu in odhodu potegnemo magnetno kartico skozi zarez v postajici in pritisnemo na tipko. Na podoben način registriramo tudi nadure, službeno in bolniško odsotnost, dopust...

Mrežo postajic za registracijo lahko priključite na računalnik. Za vrsto različnih tipov računalnikov smo pripravili paket programov, ki vam bo omogočil (s pooblastilom!) pregled in urejen izpis obračunanih podatkov. Pri vsakem delavcu bo upošteval fiksen ali dirseč delovni čas, izmene, sobote, nedelje in praznike, na postajice pa bo pošiljal kratka sporočila (npr. DELAVSKI SVET OB 15:30).



univerza e. kardelja  
institut "jožef stefan" ljubljana, jugoslavija  
Odsed za računalništvo in informatiko

61111 Ljubljana, Jamova 39/p.p. (P. O. B.) 153/Telex: 1061214-399/Telegraf: JOSTIN LJUBLJANA/Telex: 31-296 YU JOSTIN

GORAN DEVIDE

**R**ačunalniško animirani filmi si počasi, a zanesljivo utirajo pot tudi v festivalске programe in ne le na filmska platna oziroma male zaslone. Tako so organizatorji svetovnega festivala v Kanadi, letos od 29. septembra do 4. oktobra v rudarskem mestecu Hamiltonu, uvedli posebno tekmovalno kategorijo za računalniško animirane filme. Pri nas so ljubitelji animiranega filma mogli strmeti ob zmogljivosti novih tehnologij in naprav na 7. svetovnem festivalu animiranega filma v Zagrebu (23. do 27. junija). Medtem ko so si na prejšnjem festivalu mogli ogledati program Računalniška animacija le kot spremno manifestacijo, so letos poslušali tudi predavanje profesorja Dana McLaughlina, animatorja, ki poučuje na oddelku za gledališko umetnost kalifornijske univerze UCLA. Inekaj gradiva iz njegovega opisa računalniške in video animacije v ZDA je uporabljenega v tem zapisu. In, kajpada, obiskovalci so si mogli v festival-skem prostoru koncertne dvorane Vatroslav Lisinski ogledati sodobno opremo. Recimo Quick Action Recorder firme NEC, z visoko integracijo pomnilniških vezij kar je še leta 1982 obsegalo celo omaro, danes postavimo na mizo kot podstavek video monitorja. Ta računalniški sistem obsega digitalizator video signala, pomnilnik do 4 Mb, video monitor in komandno ploščo. Pomni do tisoč risb, ki jih v realnem času prikazuje na ob zaslono v zaporedju, ki ga lahko poljubno spreminjamo. Skratka, idealen pripomoček pri klasični animaciji, saj moremo v hipu testirati sekvence narisanih sličic.

Vemo, kako drage video spote si danes privoščijo največje pop zvezde (spomnimo se samo Mickey Jaggerja in njegove Hard Woman: računalniško animirani video spot so izdelali pri Digital Productions in Kaliforniji na računalniku cray X-MP z lastno programsko opremo za simulacijo scen, ki omogoča generiranje podob fotografijske kvalitete). Toda danes je na voljo tudi poceni računalniška animacija. Zato nekaj abecede o tej zvrsti ne bo odveč.



## RAČUNALNIK, novo orodje filmskih delavcev

Proizvodnja računalniško animiranih filmov poteka v treh fazah.

1. **Vnos podatkov v računalnik** je možen na več načinov.

- digitalizacija slik iz video kamere oziroma video rekorderja
- risanje z elektronskim svinčnikom (tablica, mška, svetlobni svinčnik, igralna palica...)
- vnos s tipkovnico računalnika

2. **Manipulacija podatkov v računalniku.** Običajno jo delimo na 2D in 3D animacijo. 2D animacija obsega računalniško simulirano animacijo s folijami in abstraktno poskovno animacijo.

3D se nanaša na objekte z globino, katerih ploskve so strnjene (solid) ali mrežasto upodobljene (wire frame) in se premikajo v prostoru. Ker simulira gibanje realnih predmetov, ji velika večina animatorjev posveča največjo pozornost. Koraki v 3D animaciji:

- Oblikovanje podobe poteka

običajno z veliko grafično tablico, običajno opišemo 2ični model predmeta. V tej fazi izberemo tudi barvo, osvetlitev in teksturo ploskve predmeta.

- V fazi animacije načrtujemo gibanje in časovna razmerja, ki jih sproti preverjamo z animacijo žičnih modelov v realnem času. Kot zadnji del te faze pogosto uporabljamo test gibanja s prikazom celotne slike v nizki resoluciji (hitrosti).

- Popolni prikaz (izračun) posameznih slik in snemanje slike po sliki (frame by frame) na film ali video. Izračun posamezne slike včasih traja precej časa, tudi več kot 20 minut. Zato je pogosto potrebnih 10 ali več dni za snemanje 30 sekund računalniško animiranega filma. Običajno snemajo ob urah, ko računalniški sistem ni zaseden (pogoč) ali pa na poseben računalniku z minimizirano grafično periferijo

3. **Izpis informacije iz računalnika** je možen na več načinov:

- tiskanje/risanje na papir
- projekcija animirane slike s krmiljenjem laserskih zarkov
- snemanje direktno na film oziroma video trak v realnem času
- snemanje direktno na film oziroma video sliko po sliko.

### Računalniško podprta animacija

O njej govorimo, kadar je računalnik le element v procesu animacije. Področja uporabe računalnika v animaciji so:

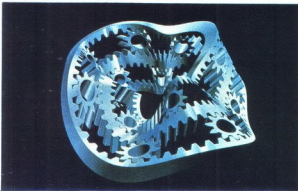
- sistemi za barvanje
- sistemi za direktno snemanje na video medij
- kontrola gibanja
- interaktivni video disk
- sistemi za specialne video efekte
- laserske projekcije

Uporaba video traku za snemanje animacije se je začela leta 1977, ko je tvrdka Lyon Lamb ponudila tržišču 1/2-palčni čb kolonni single frame recorder za testno animacijo skic risanege filma. Japonska družba NEC je razvila »Quick Action Recorder«, računalniški sistem, ki pomni in manipulira (vrstni red, število prikazanih sličic) čb digitalizirane slike.

Pomnilniški sistem, ki pomni računalniško generirane slike, preden jih posnamemo na film oz. video trak (long term digital hard disk recording, kapacitete več

»Dreams«, »Power«, »Magic«, prvonagrajenec med TV reklamami na letošnjem zagrebškem festivalu. Tehnološko vrhunski niz treh filmov, izdelanih pri Abel Image Research, producent Nancy St. John, režija Kenny Mirman in Steven Beck (zgoraj).

»Gears«: prva nagrada med eksperimentalnimi filmi (Cranston / Casuri Productions, Ohio). Dvostrana 3D upodobitev z računalnikom VAX 11/780 in opremo Pyramid Technologies, vsebinsko žal dolgočasno.



Gb) omogoča kontrolo in enostavno spremembo v animaciji. Najbolj znana sta Abekas A-62, ki lahko pomni do 100 sekund animacije za prikaz v realnem času, in Quantelov »Harry«, ki pomni 84 sekund animacije z naključnim dostopom v realnem času. Interaktivne laserske diske in digitalno snemanje še razvijajo in to bo pomenilo bistvene izboljšave v kvaliteti snemanja animacije. V prihodnosti bodo nadomestili video sisteme vrste single frame. Najbolj znan elektronski sistem barvanja uporabljajo v animacijskem studiu Hanna Barbera. V procesu animacije z video kamero digitalizirajo vsako sličico (folio), ki jo, prikazano na zaslonu računalnika, obarvajo s svetlobnim svinčnikom, tablico in menujem barv. Računalniški sistem pomni vse sličice, gibanje kamere nadomesti računalnik. Slike ozdaja so izdelane na klasičen način in s pomočjo shranjene v računalnik. Vse podatke zabeležijo na trdi disk (6 Gb), preden posnamejo animiran film na enopalčni video trak. V tem trenutku razvoja animacije predstavlja elektronski sistem barvanja najbolj vezni člen med tradicionalno ročno risano animacijo in računalnikom. Ponavljajoče in dolgotrajno opravilo barvanja folij in snemanja množice folij sliko za sliko je prevzel računalnik. Animatorji, osvojevni večine garaškega dela, so ohranili fleksibilnost ročno izdelanih risb.

### Sistemi na ključ

Računalnik predstavlja osnovno orodje v procesu animacije. Sisteme za računalniško animacijo delimo na:

- zaprte sisteme na ključ (turnkey systems)
- odprte računalniške sisteme iz več komponent (component unit systems)

*Velike pop zvezde si lahko privoščijo tudi zelo drage video spot. Prveci daleč je »pripravil«* Mick Jagger s pesmijo »Hard Woman«. Računalniško animirani video spot so izdelali v Digital Productions, Los Angeles, California, na računalniku cray X-MP z lastno programsko opremo za simulacijo scen, ki omogoča generiranje podob fotografske kvalitete.



Sistemi na ključ so kompletni paketi strojne in programske opreme za računalniško animacijo. Tako jih imenujemo zaradi tega, ker jih poženemo, ko poženemo avtomobil, ko obrnemo ključ.

Sistemi so optimizirani za operacije, ki jih podpirajo, vendar so omejene fleksibilnosti, saj jih nadzoruje zaprt operacijski program in ne operater. Primerjamo jih lahko z mehaničnim klavirjem. Eden prvih sistemov na ključ, »Ampex Video Art«, je izdelal Ampex leta 1979 za ceno 200.000 dolarjev. Izraz »low end« pomeni manjšo ceno, manj funkcij; sisteme low end turnkey uporabljajo za obogatitev TV programa, poceni reklame, napise, industrijski, medicinski in poslovni video Ce-

nje, 2. Mirage – enota za 3D animacijo, 3. Harry – digitalni rekorder na magnetni disk. Cena sistema od 150.000 do 300.000 dolarjev. Uporabljajo jih za ceneno video animacijo. Cena uporabe kompletne sistema se giblje med 600 in 900 dolarjev na uro), Symbolics sistem, pisan v programskem jeziku Lisp, reklamna cena za božične praznike je bila 149.000 dolarjev, sedaj velja 250.000 dolarjev).

### Računalniški sistemi iz več komponent

Ker jih sestavljajo iz različnih strojnih in programskih enot, ponujajo večje zmogljivosti in večjo



»Kiss Me You Fool!«: Tanya Weinberger (Telesis Productions Rochester, New York) se je najbolj odrezal med neodvisnimi animatorji. Spročena risba je računalniško animirani film navdahnjen z več življenja kot najbolj izpljena 3D animacija. Uporablja so Artronix BSA Paint System.

fleksibilnost kot sistemi na ključ. Zaostajajo v hitrosti in niso tako udobni za uporabo. Če je sistem na ključ analogen mehaničnemu klavirju, potem lahko sistem iz več komponent primerjamo s klavirjem. Potrebujemo izkušnega pianista, torej animatorja ali tandem animatorja in računalniškega programerja oziroma oboje hkrati.

Vsi ti sistemi zagotavljajo popolne 2D in 3D zmogljivosti. Sestavljajo jih različne strojne in programske enote:

- računalnik
- »frame buffer« (video pomnilnik)
- ustrezna programska oprema
- RGB monitor visoke resolucije
- velika grafična tablica oz. druge vhodne enote
- sistem za snemanje sliko po sliko
- filmska kamera oziroma video rekorder, eventualno še enota genlock za sinhronizacijo z vu-

nanjim video sistemom in korektor časovne faze.

V splošnem področju je najbolj razširen Cubicomp. Sestavlja ga video vmesnik in programski sistem, ki teče na računalniških IBM PC AT. Cena osnovnega paketa je 30.000 dolarjev »High end« oziroma sistema z visoko cenno so Abel Image Research (od 80.000 dolarjev dalje) in Wavefront (od 55.000 dolarjev dalje), oba sistema obsegata le programsko opremo, in delovne postaje Evans and Sutherland (200.000 dolarjev), IMI 500 (100.000 dolarjev), Iris (50.000 dolarjev). Delovna postaja običajno obsega računalnik, video vmesnik in osnovno programsko opremo. Za računalniško animacijo najpogosteje uporabljajo računalnike VAX 11/750 oziroma 11/780, SUN, DEC, Solidary, IBM PC ali RT. Najbolj razširjen je video vmesnik Raster Tech.

Sistem za računalniško animacijo ocenimo po naslednjih kriterijih:

- programska oprema mora biti popolna, kajti preteče lahko tudi več let, preden se izgotovi poslednji manjkajoči del programa
- koliko časa potrebujemo, da oblikujemo kompleksen predmet po naši zamisli
- ali računalnik more generirati vmesne faze
- koliko barv lahko hkrati prikazemo na zaslonu
- ali je senčenje gladko, številni virov svetlobe, kvaliteta sijaja
- ali računalnik generira fraktale (naključne vzorce, zelo uporabno npr. pri upodobitvi planin)
- ali zna prikazati prosojne ploskve
- koliko poligonov lahko prikaze
- koliko časa potrebuje za prikaz slike
- ali zmore wire frame oz. popolno animacijo v realnem času
- enostavnost uporabe, koliko operaterjev potrebuje, kvaliteta vzdrževanja s strani proizvajalca

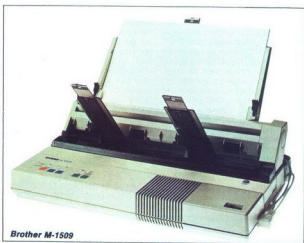
### Prihodnost računalniške animacije

Ekstremni vizionari zagotavljajo, da bo vse, kar vidimo v filmu, vključno z igralci, ustvarili umetnik z računalnikom. Nekateri celo trdijo, da bo tudi kreatorja nadomestil računalnik z umetno inteligenco – za izdelavo celotnega filma naj bi zadostoval le en ukaz. Drugi vizionari napovedujejo računalniški sistem s povezavami

## Brother M-1509

JONAS ŽNIDARŠIČ

**N**ajnovjši Brotherjev tiskalnik M-1509 je precej drugačen od malih brotherjev M-1009, ki so zaradi nizke cene in majhnih dimenzij privlačili jugoslovanske kupce. Brotherjevi tiskalniki so zaradi tega najcenejšega modela postali sinonim za poceni, a na žalost tudi nevaljiten, počasen tiskalnik z majhnimi zmogljivostmi. Mali M-1009 ni bil sposoben niti definiranja YU znakov brez mučnega prehajanja v grafični način tiskanja, kaj šele



Brother M-1509

Nenavaden je tudi položaj velike gumba za ročno obračanje valja: je namreč na levi strani tiskalnika in ne na desni kot pri vseh drugih. Zraven njega (na levi strani) je presenetljivo tudi konektor za povezavo z računalnikom. To pomeni, da sem vam kabel ne bo zapletal v papir, pomeni pa tudi, da bo tiskalnik zasedel na mizi več prostora.

Ko končno ugotovite, kam kaj spada, vas čaka še ena preizkušnja vstavljanje papirja. Zaradi plastičnega pregibljivega vodila je vstavljanje posameznih listov zares preprosto, prava muka pa je vstavljanje perforiranega papirja.

Kontrolnih gumbov ima M-1509 več kot njegovi bratje. Kar pet jih je, trije imajo enake funkcije, kot smo jih pri tiskalnikih že navajeni, dodana pa sta gumba za vklop in izklop načina

# Od plastičnega dirkača do matričnega

proporcionalne grafike ali znakov NLQ. Ker ni imel niti standardnih mednarodnih naborov znakov (švedskih, danskih, japonskih, španskih itd.), je vanj onemogočena tudi vdelava YU znakov v EPROM po YU standardu, ne da bi izgubili znake »7[7]«. YU standard zahteva namreč 10 prostih znakov, ki pa jih pri tiskalnikih najdemo le v švedskem mednarodnem naboru znakov.

M-1009 je pod svojo nalepko in v črni barvi prodajal tudi Schneider (z oznako NLQ 401), vendar s spremenjenim operacijskim sistemom, v katerega so bili vgrajeni tudi znaki NLQ. Hardversko sta tiskalnika popolnoma enaka, tako da si lahko pri znaku, ki ima NLQ 401, prekopirate EPROM in ga vtaknete v M-1009. Stvar bo delala brez težav.

Brother je kmalu poslal na trg novo različico z oznako M-1109, ki pa je bistveno hitrejši, je skoraj stodoostotno združen s standardno ESC/P (EPSON), ima nabor znakov IBM IN NLQ ter vse mednarodne nabore. Prijetna novost je tudi ta, da ima M-1109 4 K vmesnega pomnilnika, ki ga lahko uporabi tudi za definiranje uporabniških znakov z YU znaki v načinu draft teroj ne bo težav, za šumnike NLQ pa bo potrebno posebej po EPROM programatorju.

Kot EPSON FX zna tudi M-1109 risati kroge, kot se spodobijo, in ne več sploščenih elips, kot ji tiska njegov starejši brat M-1009.

M-1109 je na žalost zaprt v isto plastično škatlico kot M-1009, kar mu daje videz krhkosti in nezanesljivosti, ki ostaja nezahvalni zaščiti znakov firme Brother. Tudi M-1509 ni izjema.



Brother M-1109

## Zunanost

Novi Brotherjev tiskalnik M-1509 pa je kljub podobni oznaki povsem nov izdelek. Gre za izredno hiter tiskalnik (180 znakov na sekundo), ki pa je navkljub hitrosti presenetljivo tih. Čeprav je le malo večjih dimenzij od Epsonovega FX-85, je M-1509 tiskalnik s širokim valjem (136 pica znakov v vrstici). Skrivnost njegove majhnosti leži v preimšijeni razporeditvi ploščic tiskanega vedra, valja in transformatorja. Predvsem presečneča višina tiskalnika: 7 centimetrov in pol.

Kot njegovi starejši bratje daje tudi M-1509 krkeh in nič kaj zanesljiv videz. «Krasni» ga tudi velikost števila sestavnih delov: pokrov tiskalne glave iz dveh delov, vodilo za posamezne liste papirja

je pregibljivo, traktor za perforiran papir pa se pritrudi na zadnjo stran tiskalnika. Traktor ima spet poseben plastičen pokrov, na katerem je k sreči odsotnih nekaj sličic, ki ponazarjajo pritrditve traktorja. V pokrov je zatakajena posebna kovinska »štangica«, s katero si pomagamo pri montaži traktorja. Skratka: brez navodil mi ni uspelo pogurkati, kam kaj spada; zatčev, zagod, vodil, kolešč je kratkoma lopeč!

Poseben užitek je montaža kasete s pisalnim trakom, ki ima dodatno vodilo, ki ga je treba pritruditi na pisalno glavo. Kako to storiti, ne da bi si umazali prste, ne vem. Zanimivo je, da ima pisalna glava poseben ročico, s katero lahko premikamo glavo – kadar želimo npr. doseči stikalo DIP, ki so pod njo – ne da bi si opekli prste.

NLQ ter gumb, s katerim izbiramo način vstavljanja papirja: posamezni list, perforiran papir, avtomatsko vstavljanje posameznih listov. Za zadnjo možnost je potrebno seveda dokupiti poseben dodatek (cut sheet feeder), ki s kupa jemlje posamezne liste papirja in jih vstavlja v tiskalnik.

## Notranjost

Če je zunanji del M-1509 vreden mnogo očitkov, pa me je toliko bolj razveselila notranjost M-1509 premore oba vmesnika, vzporedni Centronics in zaporedni RS-232C. Slednji je sicer tiskalnik dodan na posebni ploščici, zato se je dobro ob nakupu prepričati, ali je ploščica vdelana ali ne. Oba vmesnika sta na levi strani, RS-232C tik nad Centronicsom.

Operacijski sistem je vdelan v EPROM 27256 (32 K), v njem pa so definirani vsi znaki. Zanimivo je, da M-1509 preverja ta EPROM, (checksum), zato vanj ni mogoče vdelati YU znakov.

Na srečo ima M-1509 predviden prostor za dodano kartico NLQ, ki tiskalniku doda še dva nabora znakov NLQ v dveh EPROM 27128, ki jih sistem ne preverja. Tako je YU znake mogoče uporabljati v načinu NLQ, sveda z uporabo enega od dveh dodatnih naborov NLQ. Ker je kartica vstavljen v notranjost tiskalnika, je bolj težko v trgovini preverjati, ali je vdelana ali ne. V tiskalniku, ki smo ga dobili za testiranje, je kartica bila že vstavljen, z njo pa je M-1509 dobil dva nova nabora znakov NLQ (gothic in anelia proportional) ter dodatnih 16 K vmesnega pomnilnika. Kartico (z oznako LQ-200) je vsekakor vredno

imeti. 16 K vmesnega pomnilnika pride še kako prav, saj bo računalnik hitro opravil svoje delo, medtem ko bo tiskalnik še kar tiskal.

Na osnovni plošči je kar 24 stikal DIP, ki določajo konfiguracijo tiskalnika ob vklopu. Stikala so nameščena zelo priročno. Pod tiskalno glavo je plastičen pokrovček, ki ga enostavno snamemo in že se pokažejo stikala. Pohvaliti je treba tudi tabelo z oznakami funkcij posameznega stikala, ki je na pokrovčku stikal DIP, tako da nam ni treba brskati po priročniku vsakokrat, ko želimo kaj spremeniti.

### Softver

Kot skoraj vsi novejši tiskalniki, je tudi M-1509 združljiv z obema standardoma, ki veljata v svetu tiskalnikov: IBM in ESC/P (EPSON). Standarda se v naboru ukazov le neznatno razlikujeta, bi-

## Orkestra

stvena razlika je v znakih ASCII nad 128. IBM ima na tem mestu mednarodne znake, grške črke, posebne matematične simbole itd., Epson pa poševne črke. Da se ne bi bilo potrebno odločati, so načrtovalci v M-1509 vdelali kar oba standarda, med njima pa izbiramo s stikalom DIP. Ker je to salomonsko rešitev uporablja že Epson v svojih tiskalnikih FX-85 in VX-105, ni odveč pričakovati isto od ostalih proizvajalcev.

Kaj pomeni združljivost z Epsonom in IBM, vemo. Za M-1509 ni potrebno predelovati printerskih driverjev, stvar poženés iz katelegičnega programa na katerikoli mašini in bo delovala tako, kot od nje pričakujete. M-1109 zna črke podčrtovati, raztegovati, stiskati; ima vse standardne grafične načine – enojna, dvojna, četverna gostota, proporcionalna grafika – ima NLQ znake, indekse, potence in vse logične mešanice različnih tipov črk.

### Kupiti ali ne?

M-1509 je tiskalnik, o katerem se splicja razmišljati. Njegove odlike so široki valji, različni nabori znakov NLQ, hitrost in predvsem presenetljivo tiho tiskanje. Pomankljivost tega tiskalnika je v bistvu samo ena, zelo krhka konstrukcija z veliko premičnimi deli, ki bi lahko bili bolj zasnovani. Kasete s pisalnim trakom so ne-standardne, zato jih bo za začetku potrebno kupiti več nankrat.

Odločen faktor bo seveda cena. Na žalost vam s tem podatkom zaenkrat še ne moremo postrčiti. Najbrž pa bo cena zelo nizka, kot smo pri tiskalnikih te firme že navajeni.

## Nec Pinwriter P6

MATEVŽ KMET

Foto: ŽIGA TURK

V skupino matričnih tiskalnikov visokega razreda (eden od njih – star NL 10 – je bil predstavljen v juljski številki MM) spada tudi serija firme NEC P6 in P7. Seriji sestavljajo štiri tiskalniki, ki se med seboj razlikujejo po širini valja in vdelanem vmesniku, vsi pa so združljivi z Epsonovim tiskalnikom LO 1500. To so: P660 (80 znakov, paralelni), P665 (80 znakov, serijski), P760 (136 znakov, paralelni in P765 (136 znakov, serijski). Poleg teh obstajajo še isti modeli z oznako «C» pred imenom modela, kar pomeni, da je z njimi možno tudi tiskanje v barvah. Seveda ti tiskalniki ne spadajo le v najvišji kakovostni razred, ampak tudi v najvišji cenovni, kar je za našega kupca žal velikokrat odločilen dejavnik, ko se odloča za ali proti nakupu. Preizkusili smo manjšega od obeh tiskalnikov – Pinwriter P6, vi pa boste morali presoditi, ali ste za tako kvaliteto pripravljeni odšteti skoraj še enkrat več kot za star NL 10.

### Tehnični podatki

NEC TYPEWRITER P6 je matrični tiskalnik z matriko 17 x 9 v načinu «draft», 17 x 32 v lepopsnem in 17 x 37 v proporcionalnem načinu. Na papirju ima natisnjena točka 0,2 mm premera. S trakom, ki je spravljen v kaseti, lahko tiskalnik tiska na original in še tri kopije. (Ko smo tiskalnik testirali, je pisal lepo tudi na četrto kopijo.) Proizvajalec jamči, da bo glava tiskalnika trajala vsaj 200 milijonov znakov (več kot 100 tisoč popolnoma popisanih strani formata A4), ena kasetna s trakom pa naj bi zadostovala za 2,2 milijona znakov (okrog 1500 strani). Hitrost pisanja je največja v za to posebej izdelanem naboru znakov, ki izkorišča posebnosti tiskalnika in znaša 216 znakov v sekundi, kar je skoraj tri vrstice teksta. V običajnem načinu «draft» je hitrost tiskanja 180 znakov na sekundo, v lepopsnem načinu pa 60 znakov na sekundo, kar je bila večših kar dobra hitrost.



že za način «draft». Že pri navadnem načinu je tiskalnik kar tih (56 dBA), lahko pa se odločimo tudi za tihi način (quiet mode), ki je za 3 dBA tiši, kar doseže tiskalnik tako, da vsako vrstico tiska dvakrat. To seveda upočasnjuje delo tiskalnika za tiste z nehekerskimi sosedi, ki nimajo razumevanja za glasno piskanje sredi noči. Tiskanje olajša tudi vmesni pomnilnik, velik 8 K (okrog pet tipkanih strani). Za konec še mere: 410 x 335 x 125 mm in teža 8,5 kg.

### Paperware

Ob nakupu tiskalnika dobimo dva priročnika. V prvem so navodila za uporabo in vzdrževanje, v drugem pa so podrobno opisane vse funkcije s kratkimi primeri v bascu. Manjkajo le slike standardnih znakov, kar bi uporabniku prišlo še kako prav pri definiranju lastnega nabora znakov, ki je predvsem v proporcionalnem načinu zaradi velike matrike zelo zamudno.

### Hardware

Že ob prvem srečanju s tiskalnikom nam postane jasno, da imamo opravka s profesionalnim iz-

delkom. Kljub lepemu dizajnu vzbuja tiskalnik vtis, da bi nam lahko brez kakšnih posebnih predelav padel na tla. Na desni strani so štiri tipke:

Tipka SELECT za začetek oziroma prekinitve tiskanja (ON/OFF LINE). FEED nam rabi za premik papirja za eno vrstico, če pa pritiskamo na tipko dalj časa, gre tiskalnik na začetek nove strani. Če je tipka pritisnjena, ko vključimo tiskalnik, dobimo t.i. «self-test».

S tipko FONT lahko izbiramo med nabori znakov; kateri nabor je trenutno izbran, pa nam kaže zaslon LED, ki je pritrjen na ohišje pod masivno pisalno glavo. Če nam le-ta zakriva zaslon, jo lahko z roko premaknemo na katerokoli lego. Izbiramo lahko med desetiimi nabori znakov; lahko se odločimo tudi za nabor znakov, ki smo ga sami definirali.

Tipka QUIET preklopi tiskalnik na tiši način tiskanja. Če je tipka pritisnjena, ko vključimo tiskalnik, izpiše tiskalnik najprej podatke o stanju sistema, nato pa izpiše sprejete podatke v šestnajstškem formatu.

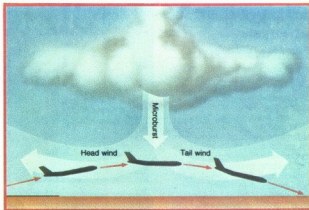
Tiskalnik ima dvakrat po osem «dip switch»-ov. Z njimi nastavimo dolžino strani, način pisanja, skoke čez konec listov, obliko ničle

Nadaljevanje na 13. strani

**D**rugega avgusta 1985 je Lockheed L-1011 TriStar (N726DA), letalo družbe Delta, na poletu 191 prešel v zaključni del prileta na letališču Fort Worth pri Dallasu. Vreme je bilo zelo slabo, pihal je močan veter z dežjem. V letalu je bilo 155 potnikov in 8 članov posadke. Po predvideni proceduri je pilot zmanjšal hitrost na 160 mph (cca 290 km na uro). Minimalna hitrost za tovrstna letala je sicer 112 mph (cca 200 km na uro). Nenadoma je nastal močan čelni veter in kazalec merilnika hitrosti je hitro poskočil (hitrost vetra, približno 40 vozlov, se je vektorsko seštelila s hitrostjo letala). Da bi pilot ohranil načrtovano hitrost in da hkrati ne bi pokvaril ravnine spuščanja (t. i. glide path), je zmanjšal moč motorjev in znova stabiliziral hitrost na 160 mph. Toda v vsega nekaj sekundah je navzdolnik sunek vetra (windshear) pokazal visoko zahrbtnost. Čelni veter hitrosti 40 vozlov se je sprevrgel v veter, ki je z isto hitrostjo pihal v rep. V teh picilih sekundah je hitrost TriStara glede na okolno ozračje padla za približno 80 milj, globoko pod normalo. Krmilne ročice so se zatresle in oglasile so se alarmne naprave. Pilot je nemudoma potisnil naprej ročice za plin, toda nos letala se je med tem, ko so trje močni motorji Rolls-Roycea z 19 t potiska nabirali moč, povesil in treščil v tla, še preden je TriStar nabral dovolj moči, da bi ostal v zraku. Vse to imenovalnega prevlečenega leta (ang. stalling) ni bilo mogoče preprečiti zaradi majhne višine in premalih rezerv moči v motorjih. V nesreči je umrlo 128 potnikov in vseh 8 članov posadke.

**A**li se je podobnim katastrofom mogoče izogniti? Danes tri firme samostojno razvijajo sisteme za opozarjanje pilotov na nevarnost navzdolnega učinka vetra in za pomoč pri letenju na kritičnih območjih. Takšni sistemi so nujna, saj je navzdolnik sunek vetra po podatkih ameriške ustanove National Transportation Safety Board v zadnjih desetih letih samo v ZDA povzročil štiri velike katastrofe in terjal 401 življenje. Pri tem ni upoštevana veliko katastrof manjših letal, ki so jih močne spremembe hitrosti in smeri vetra polomile in razmetale njihove ostanke na površini nekaj kvadratnih kilometrov, pa tudi na številne nesreče in katastrofe, katerih vzroki so bili drugačne narave, vendar povezani z učinki vetra.

Z učinkom vetra imamo v mislih vsako hitro spremembo hitrosti in smeri vetra na majhnem prostoru oziroma v kratkem času, posledica pa je hitra sprememba hitrosti letala glede na okolno ozračje. Najbolj nevarna oblika učinka vetra so močni navzdolni tokovi zraka (ang. downdraft ali microburst, ker se pojavljajo na majhnem prostoru), ki nastajajo v aktivnih kumulonimbusih, nevihtnih oblakih, pogosto hkrati z močnim dežjem (kot v primeru



**Najbolj nevaran je veter, ki nastane zaradi močnih navzdolnih sunkov zračnih gmot na majhnem prostoru; takšno strujanje izvira iz nevihtnega oblaka in se na nizki višini širi v vse smeri. Zaradi hitre spremembe smeri vetra utegne letalo izgubiti vzgon.**  
(Legenda: head wind – čelni veter; tail wind – veter v rep, microburst – ozko, močno navzdolno strujanje zraka.)

## Z računalnikom proti zahrbtnim vetrovom

Delte 191), točo in gremenjem. Močni navzdolniki nastajajo takrat, kadar postane kondenzirana voda, ki jo nosijo navpični tokovi, pretežka in strmoglavijo proti tlom. Microburst, ožji od pol milje, je včasih močan kot tornado. Meteorološki radarji posredujejo pilotom dovolj podatkov, da obidejo te silovite in nevarne padajoče zračne gmete – vendar le takrat, kadar letijo v velikih višinah, težave pa se pojavijo pri pristajanju oziroma vzletu, ko ni mogoče spreminjati smeri poleta. Podobno je pri poletih na majhni višini, kadar se pojavijo navzdolniki navpičen zračni tok se ob trčenju z zemljo deformira in se radialno širi v vse smeri (glej sliko).

Poznamo dva sistema za odkrivanje nevarnih sunkov vetra. Prvi je opt na kopensko mrežo meteoroloških postaj, s katerimi meteorologi odkrivajo in natančno locirajo območja z močnimi navzdolniki. Drugi sistem temelji na instrumentih v letalu, povsem neodvisnih od kopenskih naprav. Ti instrumenti opozarjajo na tovrstno nevarnost in so povezani z instrumenti za pilotiranje. Ameriška Federal Aviation Administration (FAA), zvezna letalska uprava za zdaj uporablja LWSAS – Low Level Wind Shear Alert System, sistem za opozarjanje na navzdolne sunke vetra na majhnih višinah, ki ga sestavlja mreža senzorjev za merjenje hitrosti in smeri vetra na raznih točkah v bližini letališča. Senzorji nenehno pošiljajo zbrane podatke centralnemu procesorju, ki

prejete podatke analizira in jih na zaslonu monitorja prikazuje operaterju. Takšni sistemi so učinkoviti v neposredni bližini letališča, ne pa vzdolž naletnih in odletnih smeri daleč od praga letalske steze, torej območjih, kjer so letala še zlasti občutljiva na učinke vetra, ker piloti ne morejo močnega spremeniti kurza, sicer bi pokvarili proceduro pri pristajanju oziroma vzletanju. Za odkrivanje sunkov vetra je potrebno tudi veliko časa. Včasih tečečeta tudi dve minuti, preden močni veter, ki se širi iz jedra nevihtnega oblaka, zajame dovolj postaj in preden lahko z gotovostjo odkrijejo in locirajo navzdolnik. In celo tedaj mora pilot – na pamet – presojati položaj in sam odločiti, kako bo nadaljeval predpisano proceduro oziroma ali jo bo sploh nadaljeval. Za ponazoritev povemo, da je sistem LWSAS med približevanjem Delte 191 na letališču Fort Worth deloval FAA podpira nadaljnji razvoj zemaških sistemov, vendar sistemi, vdelani v letalo, omogočajo hitrejša preprečevanja katastrof.

**T**ese sisteme sestavljata računalnik in merilnik pospeškov, ki odkrjeta sleherni nepredvideni premik in vsako nepričakovano odstopanje od predvidene smeri oziroma spremembo kota glede na horizont in vsako nepričakovano spremembo hitrosti, pilotu v hipu posredujejata ključne podatke in mu

pomagata iz kritičnega položaja. Posadka potrebuje več kot navaden indikator nevarnosti, kajti polet skozi takšna območja zahteva takšno raven pilotiranja, ki močno presega izkušnje in izurjenost velikega števila pilotov. V normalnih razmerah je zračni tok ob krilih ponesostavljeno mogoče obravnavati kot tok, ki struja paralelno s površjem tla, nad katerim leti letalo. Zato je vpadni kot med krilom in zračnim tokom v grobem enak kotu med krilom in tlemi. Navzdolnik pa ima močno vertikalno komponento hitrosti in v hipu zmanjša vpadni kot krila. Med potekom skozi navzdolne zračne tokove mora pilot zato dvigniti nos letala in leteti z visoko močjo motorjev.

Nekaj družb, vsehvi Boeing Aircraft (Seattle), Safe Flight (White Plains, New York) in Sperry (Phoenix), razvijata takšne sisteme za opozarjanje pilota na srečanje z navzdolniki in za vodenje letala skozi nevarna območja. Safe Flight je konstruiral sistem Wind Shear Warning/Recovery Guidance (WSW/RG), pri katerem računalnik uporablja letalska merilnika vertikalnih in longitudinalnih pospeškov za računanje parametrov letla glede na tla kot referenčno ploskev. Računalnik sprejema tudi podatke senzorja hitrosti in vpadnega kota, podatke, izmerjene glede na tokove okolnega zraka. Če sistem WSW/RG odkrije nenadno povečanje hitrosti glede na ozračje, ne da bi se pri tem ustrezno povečala tudi hitrost glede na tla (pojavi se razlika med hitrostjo, iz-



merjeno v Pittotovi cevi, in hitrostjo, ki jo je izračunal računalnik na temelju sprememb hitrosti, registriranih do tedaj z longitudinalnim akcelerometrom, in sicer je razlika v koeficientu hitrosti zračnega toka v Pitotovi cevi, tega je edini možni vzrok pojava navedene hitrosti čelnege vetra, to pa navaja nevarno učinkovanje zračnih tokov. Sistem sproži zvočni alarm, še preden je nevarnost takšna, da se je bati hudih posledic, izračuna nov kot med vzdolžno osjo letala in horizontom (t. i. pitch) in opozori posadko, naj se pripravi za polet skozi navzdolne zračne tokove. Pilot mora kajpada hitro reagirati in zato skušajo kar najbolj omejiti lažne alarme (elegantni trik – uporaba avtopilota – utegne biti zelo nevarna, kajti naprava bo skušala ohraniti prejšnji »pitch« in kazalec merilnika hitrosti bo »omahnil«).

Zato je Safe Flight polni dve leti preučeval svoj sistem na enem od boeingov 727 družbe United Airlines in skušal kar najbolj raziskati vpliv zračnih turbulenc, glavnih vzrokov za lažne alarme. Meritve so pokazale, da turbulenca pri velikih letalih redkokdaj povzroča obremenitve, večje od 0,12 g. Čeprav se

ričivo in za zmanjšanje vzdrževalnih stroškov Kadar računalnik odkrije navzdolne zračne tokove, se prižge signalna lučka jantarjeve barve (algoritem temelji na kotu med vzdolžno osjo letala in horizonta, hitrosti, vertikalnih in longitudinalnih pospeških). Sperry ta sistem še izpolnjuje, da bi podobno kot sistem Safe Flighta posređeval še navodila za nadaljnje postopke. Do konca leta naj bi Sperry dobil dovoljenje za komercialno uporabo svojega sistema.

Boeingov sistem z instrumenti za pilotiranje opozori, da se letalo približuje nevarnemu območju, nato ga vodi skozi navzdolne zračne tokove. Boeing pričakuje dovoljenje za namestitve tega sistema v svojih letalih tipa 737 do konca leta, v načrtu pa sta še različici za 757 in 767.

Katastrofi Delta 191 bi se mogli izogniti, če bi bilo letalo opremljeno s tovrstnim sistemom. Posadka bi pravočasno dobila obvestilo o navzdolnih sunkih vetra in pilot motorjem ne bi bil odvezel plina, da bi ohranil načrtovano hitrost, temveč bi maksimalno dodal plin in poskrbel za kar največji vpadni kot, nato pa bi izvedel postopek za neuspe-



*Instrument Sperryjevega računalnika, s katerim je mogoče odkrivati in meriti navzdolne sunke vetra.*

morajo piloti boriti z obremenitvami, ki so veliko večje, je Safe Flight kalibriral svoje naprave na 0,15 g. Sistem je testiral in ga sprejeti tudi FAA, in doste so ga vdelali v več kot 40 velikih letal Družbe Eastern, United in Boeing so z letalskimi simulatorji letenja opravile še niz dodatnih testov, ki so pokazali, da sistem zanesljivo reagira na sunke vetra.

Langskega novembra je Sperry spovil lasten program za odkrivanje negativnih učinkov vetra in alarmiranje posadke. Teste so opravili že med razvijanjem softvera, pozneje pa so elektriko in programsko podporo preskusili še v praksi na letalih družbe Piedmont Airlines. Program, ki ga uporabljajo posebni računalniki te letalske družbe, vdelani v boeing 737-200, je zdaj sestavni del Sperryjevega sistema Performance Management System, sistema za varčevanje z go-

šen nalet (t. i. missed approach) in poskusil znova pristati (ker se »microbust« nenehno premika skupaj s komulnibusom) oziroma bi zbral alternativno letališče.

Leonard Green, predsednik Safe Flighta, pri tem opozarja na neko ironijo usode. Ker so se opisane naprave pojavile šele pred kratkim, jih pri FAA še testirajo in jih letalske družbe ne smejo nameščati brez dovoljenja agencije. »Prava ironija je,« pravi Green, »ker se je v Dallasu ubilo šest strokovnjakov IBM, medtem ko cela flota poslovnih letal IBM že uporablja naprave za odkrivanje nevarnih učinkov vetra.«

# KUPUJTE

# MOJ MIKRO

# CENEJEJE!

- Bralec Mojega mikra ponujamo priložnost, da se zavaruje pred inflacijskimi »presenečenji«. Kako?
- Preprosto: postanite naš redni naročnik in podražitve vas ne bodo prizadele. Kako dolgo?
- Pol leta, če boste naročili Moj mikro za pol leta oziroma celo leto, če ga boste naročili za celo leto. Kaj storiti?
- Izpolnite spodnjo naročilnico in jo pošljite na naslov: Moj mikro (za naročnine), Titova 35, 61000 Ljubljana. Začeli boste prejemati Moj mikro, pozneje pa boste dobili tudi položnico in ko boste poravnali naročnino, si boste zagotovili stalno ceno, neodvisno od zanesljivih podražitev, ki nas čakajo v novem letu.

OMENJENE UGODNOSTI VELJAJO SEVEDA TUDI ZA STARE NAROČNIKE! NAROČNINO ZA PRIHODNJE LETO JIM BOMO AVTOMATSKO PODALJSALI ZA POL LETA, ČE PA ŽELJIVO PLAČATI ZA VSE LETO, NAJ TO SPOROČUJO NA GORNJI NASLOV!

**Važno:** za zveste naročnike, stare in nove, pripravljamo še nekaj drugih presenečenj. O tem v prihodnji številki!  
**Pot do cenejšega Mojega mikra:** izrežite spodnjo naročilnico in nam jo izpolnjeno pošljite (še nočeta z izrezovanjem pokvariti revije, se lahko naročite tudi s pisemcem ali dopisnico oziroma preprosto zavrtite telefon: (061) 319-798).

Podpisani \_\_\_\_\_ (čitljiv priimek in ime)

naročam slovensko-srbohrvaško izdajo Mojega mikra (nepotrebno prečrtajte)

na naslov \_\_\_\_\_ (navedite točen naslov, vključno s pošto številko) za dobo 6 mesecev – 12 mesecev (nepotrebno prečrtajte)

Podpis \_\_\_\_\_

## PRIMO POGAČNIK

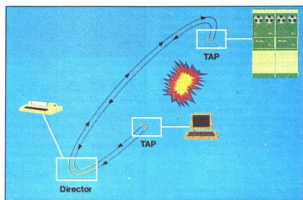
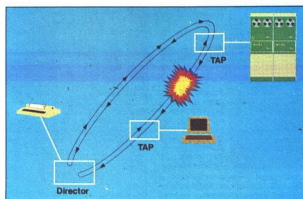
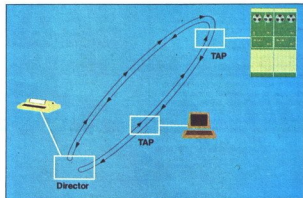
**D**je ko uporabljamo računalnike, pogosteje ugotavljamo, da je računalnikovo »ozko grlo« prav izmenjava podatkov. Nekdaj so za prenos računalniških podatkov uporabljali perforirane kartice, kasneje magnetne trakove, diske. Na ta način so se podatki prenašali zelo počasi, težave so bile tudi z združljivostjo sistemov (še posebno velja to za diske). V novejšem času podatke izmenjujemo z modemi po javnem telefonskem omrežju. Hitrost prenosa in dostopnost podatkov pa se nista bistveno povečala. Po drugi strani se srečujemo z dejstvom, da cena osebnih računalnikov nenehno pada in da ne potrebuje vsak uporabnik računalnika tudi svoj lepissimi tiskalnik oziroma svoj trdi disk ali risalnik. Proizvajalci računalniške opreme so zato začeli razvijati posebne sisteme, ki so omogočali, da je več uporabnikov uporabljalo iste periferne enote, npr. diskovni multiplekser (Corvus Constellation) ali systemizer (Data-trade). To je privedlo do tega, da so uporabniki računalnikov začeli razmišljati o povezovanju računalnikov. Analize so pokazale, da je za 90 odstotkov komunikacij potrebna večja hitrost prenosa kot 300 K bit/sek in da se 60 odstotkov pisnih komunikacij opravi znotraj podjetja. Na podlagi omenjenih podatkov in zamisli o avtomatizirani pisarni se je porajala ideja o lokalnih mrežah.

## Značilnosti lokalnih računalniških mrež

Lokalna računalniška mreža pokriva omejeno območje v mejah od nekaj 100 m do približno 10 km. Po zasnovi niso podvržene omejitvam, pod katerimi delujejo PTT in drugi javni komunikacijski sistemi. Prenos podatkov je relativno hiter, 100 k bit/sek in do več kot 10 M bit/sek. V primerjavi s terminali uporabljajo pri lokalnih računalniških mrežah poceni prenosne medije: koaksialne kablice, razne dvožilne kablice. Vsak vozil v računalniški mreži lahko komunicira s katerikoli drugim in tako sama mreža ne potrebuje nadzornega računalnika. Sporočila in informacije se prenašajo tako, da oddajna postaja odda tudi naslov uporabnika, kateremu je namenjeno sporočilo. Vse postaje poslušajo sporočilo, toda le naslovljena ga uporabi in tudi ustrezno reagira. Tajnost zvez na ta način ni zajamčena in še daleč ni tako velika, kot je

Če se posamezne postaje v obročju zaprejo, lahko pride do popolne prekinitve zveze. Da bi se temu izognili, uporabljamo dvojno zanko. Ko postaje v obročju od sosednje ne dobi potrditve prejema, sama zaključijo tok podatkov. Ustrezno reagirajo tudi druge postaje v obročju. Ta lastnost omogoča, da med samim e anjem obroča dodajamo nove elemente v kablju brez škodljivih posledic za prenos podatkov. Takšno možnost dopušča tudi Recal-Milgvo PLANET.

## LOKALNE RAČUNALNIŠKE MREŽE



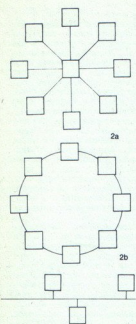
pri direktni povezavi med dvema uporabnikoma. Tajnost podatkov se poveča šele s filtriranjem sporočil ali pa, kar danes čedalje uporabljajo, s posebnim nadzornim računalnikom, ki nadzira prenos podatkov.

## Topologija povezav

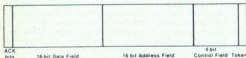
Prve mreže so imele zvezdasto obliko (slika 2). Centralna enota je bila povezana z vsakim posameznim uporabnikom. Uporabniki so lahko sedeli pred relativno enostavnimi terminali, medtem ko je imel centralni računalnik zelo obsežno programsko opremo. Tovrstna mreža je imela tudi veliko pomanjkljivost: če je prišlo do okvare v centralnem računalniku, potem posamezni uporabniki niso več mogli vzpostaviti medsebojne zveze. Kljub temu zvezdasto omrežje še vedno uporabljajo. Najbolj značilen primer je PBX (private branch exchange) ali Juvex, ki uporablja obstoječe telefonske napeljave. V obeh primerih se informacije prenašajo počasi, pogosto celo počasneje kot 1200 bit/sek.

V zadnjem času pogosteje uporabljajo mrežo v obliki obroča. Informacije krožijo po obroču, najpogosteje v eni smeri, in se ojačijo na vsakem vozilu. Toda spet lahko pride do prekinitve zveze, če odpove ena izmed postaj. Zaradi tega pogosteje uporabljajo dve paralelni zanki in ob prekinitvi linije ali okvari postaje se informacijski tok zaključijo tako, kot prikazuje slika 1.

Na področju komunikacijskih mrež je bilo v preteklosti opravljenih več eksperimentov, najbolj znanega med njimi so naredili na Havajski univerzi. Slednji poskus je imel tudi močan vpliv na razvoj računalniških mrež. Na že omenjeni univerzi so želeli povezati vse terminale na Havajskih otokih z lokalnim računalnikom, od tod dalje pa z drugimi računalniškimi mrežami. Ta sistem se imenuje ALOHA; povezava med terminali je bila zagotovljena z radijskimi oddajniki. Vsak terminal ima za oddajo in sprejem na razpolago radijski kanal širine 100 Hz. Zveze so ustvarili takole: vsaka postaja je naprej »poslušala«, ali morda že oddaja katera druga. Če je bil kanal prost, je postaja začela oddajati sporočila. Seveda pa ne moremo vedeti, ali je bilo sporočilo sploh sprejeto oziroma ali je bilo sprejeto brez napak. Zato je morala vsaka postaja, ki je sprejela sporočilo, nato oddati tudi potrditev prejema. Če oddajna postaja ni sprejela potrditve, je sporočilo ponovno oddala. Obstajala je tudi možnost, da sta dve postaji ugotovili, da je kanal prost in sta tako istočasno pričeli oddajati sporočila. Sprejeti podatki so bili seveda napacni in zato sprejema postaja ni oddala potrditve prejema. Sledila je ponovna oddaja sporočila in ta postopek se je ponavljal toliko časa, dokler ni bilo sporočilo sprejeto brez napak. Sistem je zadovoljivo deloval, dokler je bilo sporočil malo, večje je bilo tudi število sporočil, večja je bila tudi verjetnost, da so se »prekrivala«, temu pa je sledila daljša čakalna doba.



Prikazane so najpogostejše topologije mrež. Zvezdasto omrežje (slika 2 a) je značilno za vse zgodnejše tipe mrež, takšno je npr. telefonsko omrežje. Zvezda se vzpostavlja preko centralnega vozila, od katerega je tudi odvisna zanesljivost zveze. Obročni ali zanke (sl. 2b) so danes zelo pogosti pri konstruiranju posameznih mrež. Če posamezni vozil odpravno, lahko odpravno povzročijo prekinitev zveze med drugimi vozili. Sl. 2 c prikazuje linjsko obliko zveze. Vozila lahko poljubno dodajamo ali odvajamo in to popolnoma brez vpliva na zvezo med drugimi vozili. Takšno obliko ima npr. Ethernet.



Paketna struktura (glej članek).

Kot rečeno, je bila ALOHA osnova več poznjšim sistemom. Najpomembnejšega med njimi so razvil leta 1970 pri Xerox Palo Alto Research Centru, imenoval pa se je Ethernet. Za razliko od komunikacijskega sistema ALOHA je bil kot "vodnik" informacij uporabljen koaksialni kabl. Vsaka postaja je stalno spremljala pretok informacij po kablu in če je zaznala, da so informacije popačene, je nemudoma prenehala oddajati. Da ne bi ponovno prišlo do sočasnega oddajanja, je posamezna postaja pričela oddajati ob naključnem času. Na koncu oddaje je vsak oddajnik oddal naključno niz podatkov – šum, kar je v bistvu enaka popačeni informaciji; to so zaznale vse postaje, priklopljene na vodnik in po naključnem času je pričela oddajati tista postaja, ki je imela informacijo, pripravljeno za oddajo. Sam postopek v mnogočem

slopinja na pogovor večje skupine ljudi, v kateri ni koordinatorja, ki bi določal vrstni red govora vsakega posameznika. To je tudi ena od prednosti Etherneta, sistem je namreč preprost in v primerjavi z drugimi relativno zanesljiv in poceni. Proizvajalci so razvili tudi več integriranih Ethernetovih kontrolerjev, npr. Intel 82586, Mostek 68590.

## Obroči in zanke

V obroči in zankah kroži informacija po obroču v eni smeri. Da ne pride do trkov informacij, uporabljajo terminali kontrolo z žetonom. Žeton je posebno sporočilo, ki dovoljuje posamezni postaji/terminalu oddajo. Postaja, ki želi oddajati, zažre žeton toliko časa, dokler ne zaključijo oddajo podatkov, nato pa odda žeton naslednjemu uporabniku. Nekoliko drugače se prenašajo informacije v obroču z vagončki. Po obroču krožijo paketi (vagončki) določene dolžine in zgradbe. Vsaj en bit paketa rabi signalizirano, ki je »vagonček« poln ali prazen. Vsak

vozel prekontrolira, ali je vagonček poln ali prazen in če je prazen, nalozijo svojo informacijo ter ga označi kot polnega. V nasprotnem primeru, torej kadar je vagonček poln, ga vozel – če je namenjen njemu – prebere in odda dalje, drugače pa ga nespremenjeno posreduje naslednjemu vozlu. Zelo znan obroč, ki se često uporablja, je Cambridge ring, ki so ga razvili na univerzi v Cambridgeu. Razvili je že nekaj komercialnih različic Cambridge ringa, npr. Polynet, Trasnring (Sci-

trolirano izdelave predmetov v proizvodnji. Prasy tako lahko z računalniškimi mrežami povežemo več osebnih računalnikov z enim tiskalnikom ali moderm, trdim diskom... Seveda moramo vedeti, kdaj se instalacija računalniške mreže že splača. Nesampetno jo je uporabili le za povezavo več računalnikov s kakšno dražjo periferno opremo, npr. laserskim tiskalnikom, saj so na voljo tudi mnogo cenejše rešitve.

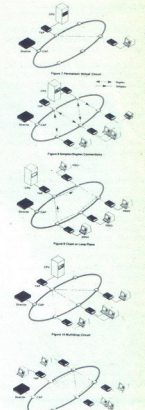
Prednost lokalne računalniške mreže je tudi ta, da potrebujemo je za povezavo računalnikov le en kabl. Nasprotno pa potrebujemo za »klastično« povezavo terminalov z računalnikom po en par žic za vsak terminal.

Vsak proizvajalec ponuja svojo priredbo lokalne računalniške mreže. Večinoma so izpeljanka osnovnih variant, pogosto Cambridge ringa. Z izpeljanko PLANET (Private Lokal Area Network) bomo predstavili tipično računalniško mrežo.

PLANET je primer uporabe Cambridge ringa. Po obroču kroži konstantno število paketov. Največje število paketov v obroču je 64. Paket, ki je konstantne dolžine, je sestavljen iz bitov, ki signalizirajo podrtitve sprejema: 16 bitov predstavlja podatke, 16 naslov, 4 biti so namenjeni kontroli in odpravljanju napak. Sledi bit, ki označuje, ali je paket poln ali prazen (sl. 3).

Povezava med vozili je narejena z dvojnimi, naključnim koaksialnim kablom, v zadnjem času pa vse pogosteje uporabljajo tudi optična vlakna. Vstop v obroč je mogoče prek kabelskega priključka (CAP, cable access point), ki ni nič drugega kot standardni priključek BNC, na katerega je možno enostavno priključiti terminalski vmesnik (TAP, terminal access point). Proizvajalec ponuja tudi posebne izvedbe CAP-priključev z optičnega vlakna na koaksialni ali pa CAP za povezavo na optični kabel. TAP je aktivni del sistema PLANET in omogoča priključitev terminalov preko konektorjev V24/28. Z nadzornim kontrolerjem tvori inteligentni vmesnik. Vsak TAP ima svoj naslov, ki ga uporabnik lahko promembno izmenjuje, to, da deluje neodvisno od uporabniškega protokola, kar pomeni, da ni treba modificirati obstoječih komunikacijskih programov.

Srce sistema PLANET je nadzorni kontroler, imenovan tudi direktor. Direktor vzpostavlja zveze med posameznimi TAP, prav tako nadzira pretok informacij in stanje v obroču. Vse podatke shranjuje in jih po želji posreduje uporabniku. Ima vedno TAP in priključke za terminali oziroma tiskalnik. Če zazna napako na liniji, pretrgana linija to napako na poseben način lokalizira. Pri tem je prišlo do izgube samo tistih informacij, ki so bile trenutno v obroču. Direktor lahko vzpostavlja več različnih tipov zvez, kot na primer enosmerne (simplex) dvosmerne (duplex) zveze med dvema določenimi uporabnikoma (point to point), zveze med enim glavnim in več podrejenimi uporabniki (multidrop), zanko (chain), konferenčne zveze. Slika 4 prikazuje posamezne tipe teh povezav. Obliko zveze lahko



Različni tipi povezav: Simplekska, duplexka med posameznimi uporabnikoma (a, b); vertikalna enojna povezava med uporabniki (c); multidrop (d); konferenčna povezava, za katero je značilno, da oddajo enega uporabnika poslušajo vsi drugi (e).

spremenimo med samim delovanjem mreže, seveda če poznamo geslo, ki nam le spremembe dovoljuje. Tako je zagotovljena tajnost in prioriteta prenašanih informacij, saj direktor ne bo vzpostavil zveze med dvema uporabnikoma, če poprej ni bil programiran, da takšna zveza obstaja.

V nasprotju z običajnimi capljanjem za svetlom si bodo lahko tudi naše delovne organizacije omislile Recal-Milgov Planet. Opisani sistem bo pri nas prodajala Metalka, TOZD, Računalniški inženiring, tel. (061) 317-654, kjer posredujejo tudi natančne informacije o instalaciji, uporabi in ceni sistema.

entifics and Electronics Ltd), Planet (Recal-Milgo).

## Uporaba lokalnih računalniških mrež

Različni računalniki uporabljajo različne operacijske sisteme in zato jih praktično ni mogoče elegantno povezati med seboj, težko je verjetno, da bi ena tovarna uporabljala le en tip računalnika, saj za urejalnik teksta ne potrebujemo računalnika z izredno kvaliteto grafiko, drugače pa je pri načrtovanju vezji. Prav tako se računalniki nenehno spreminjajo, vsakdo bo kupil najnovejši model ali pa listega, ki mu je trenutno finančno dostopen. Nezdravljivost računalnikov neprestano raste, s tem pa se zmanjšuje možnost hitrega prenosa podatkov.

Še posebno pride do izraza povezava računalnikov z računalniško mrežo v odnosu med prodajno-proizvodno-nabavno službo nekega podjetja ali pa pri krmiljenju ali kon-

## Zgodovina računalniške animacije

nadaljevanje s strani 5

po vsem svetu, sistem, ki bo omogočil ljudem izdelavo in gledanje lastnih video filmov kar doma.

Del prihodnosti bi lahko bila računalniška animacija v realnem času. Animacija, ki bi omogočala nadzor in ustvarjanje lastnih 3D risb s trenutnim odzivom. Za ta namen bi potrebovali pomnilnike z večjo zmogljivostjo (npr. laserski disk), zmogljivejšje računalniške in programske opreme z umetno inteligenco.

V igranih filmih računalnik uporablja pri pisanju scenarija, nadzoru produkcije, raziskujejo na področju načrtovanja filmskih kadrov in celotnih filmskih sekvenc. Nekateri producenti razmišljajo o digitalizaciji obrazov umrlih rokovskih zvezd, npr. Jima Morrisona. Tako bi lahko izdelali nove video spote, kjer bi igralcem namodelili obraz z digitalizirano sliko

V praktičnem pogledu bo računalniška animacija prav gotovo vplivala na animacijo v tolikšni meri, kot se je to zgodilo pri vplelavi računalnika v produkcijo glasbe. Tako je npr. v Hollywoodu ostala večina tradicionalnih studiovskih glasbenikov brez dela, ker se je razvil nov tip glasbenika – programerja-aranzerja.

Prvi računalniško animirani film je verjetno »Bouncing Ball«, izdelan leta 1950 v Massachusetts Institute of Technology (MIT), prikazan leta 1951 na ameriški TV. Leta 1957 se je John Whitney, ki se mu je kasneje pridružil brat James, lotil z analognim računalnikom raziskav na področju kontrole gibanja. Whitney je produkcijo računalniško animiranih filmov pričel leta 1966 s filmom »Permutations«.

Leta 1963 je ključno v zgodovini računalniške animacije: Ivan E. Sutherland je na MIT objavil doktorsko disertacijo z naslovom »Sketchpad: A man-machine graphical communication system«. Projekt je omogočil umetniško risanje in animacijo na zaslonu računalnika s svetlobnim svinčnikom. Računalniška animacija je postala interaktivna.

Kenneth Knowlton (Bell Labs) in Stan Van Beek (avtor eksperimentalnih filmov) sta v letih 1964 – 1970 izdelala 9 računalniško animiranih filmov. Leta 1974 je Peter Foldes (National Film Board of Canada) izdelal »La Faim« (lakota), katerega angažirana vsebina in efektna tehnika izdelave sta prispelvala prvi nominaciji računalniško animiranega filma za oskarja.

New York Institute of Technology (MIT) je leta 1978 začel proizvodnjo prvega računalniško animiranega celovečernega filma, ki pa še vedno ni posnet. Računalniško animacijo so v filmih pogosto uporabljali za posebne efekte oziroma sekvence. Film »2001, A Space Odyssey«, izdelan 1968, je zlasti pomemben, saj je uveljavil uporabo računalnikov v filmski produkciji (gibanje vesoljskih ladij in učinke nadzoruje računalnik). Prve računalniške animacije so bile enostavne črtaste risbe in tiskane besede v celovečernem »The Andromeda Strain« (1971). Leta 1973, v filmu »Westworld«, je v računalnik vnesla sekvence digitaliziranih slik žive akcije za potrebe nadaljnje manipulacije. V nadaljevanju filma »Futureworld« (1976) je manipulirana digitalizirana slika glavnice Petra Fonda. Istega leta je v filmu »Demon Seed« prikazan prvi računalniški orgazem, ko superračunalnik Proteus 4 oplodi Julie Christie. Animacija je izdelana na analognem računalniku scanimate tvrdke Computer Images. Film »Tron« (1982) sestavlja 15 minut čiste računalniške animacije za 235 prizorov s povprečno ceno 1200 dolarjev na sekundo. Dodatnih 200 scen vsebuje računalniško generirana ozadja. Film je zaradi šibke zgodbe zadetek v prazno, njegov neuspeh je korak nazaj v

uporabi računalniške animacije v celovečernih filmih.

Se nekaj celovečercer, ki vsebujejo računalniško animacijo: »Time after Time«, »Star Trek, The Motion Picture«, »Star Wars«, »Buck Rogers«, »Alien« in »The Black Hole« v letu 1979. »Looker« leta 1981, »Star Trek II« leta 1982, »Superman III«, »War games«, »2010« leta 1983, »The Last Star Fighter« (s 30 minutami animacije za ceno 4,5 milijonov dolarjev) ter »Electric Dreams« leta 1984. »The Explorers«, »Weird Science« (računalnik kot seksualni objekt), »Young Sherlock Holmes« 1985.

Uporaba računalniške animacije je prav gotovo najbolj popularna na TV, predvsem v reklamah, kar zadeva sprejemljivost in število gledalcev.

Računalniško animacijo, predvsem sistem na kjuč, vse bolj uporabljajo v poročilih in športnih programih.

Zakaj je računalniška animacija tako popularna med oglaševalci? Pred uporabo računalniške animacije je izdelek, npr. hladilnik ali toaletno školjko, prodajala seksi ženska, znana osebnost ali figurica iz risanke. Toda z računalniško animacijo lahko hladilnik ali WC školjka letita, žarita, pojeta in plešeta. Naročniki so navdušeni, njihov predmet zaživi.

## MOJ MIKRO

## Slovenija MIKORARAČUNALNIK ZA SAMOGRADITELJE

Druga serija osnovnih kompletov nam je pošla že pred pričetkom šolskega leta. Medtem se je nabralo kar nekaj novih samograditeljev, ki bi radi kupili osnovni komplet. Toda vmes so se cene na domačem tržišču povzpelle tako strmo navzgor, da bomo poskušali sproti naročiti potrebno količino. Račun je pokazal, da bo znašala cena osnovnega kompleta (dokumentacija, dva pomnilnika tipa eprom in tiskano vsebo) od 80.000 do 90.000 dinarjev. Prosimo vse, ki jih zanima nakup osnovnega kompleta, da nam pisno javite (na naslov uredništva revije Moj mikro), da ste pripravljene kupiti osnovni komplet v okviru navedene cene. Večina samograditeljev je v zaključni fazi dela. To so seveda najbolj napeti trenutki, zato je vsaaka čakanje nevzdržno. Zato vam dajemo novi telefonski številki, na kateri lahko kličete vsak dan od 20. ure naprej. Obe sta v Ljubljani: (061) 344-697 in 332-591. Pokličite, če imate kakšno koli vprašanje v zvezi s projektom Moj mikro Slovenija.

Mikroročunalnik »Moj mikro Slovenija« (MMS) je namenjen samograditeljem. Izdelava mikroročunalnika MMS je projekt, ki sočasno ponuja tudi dovolj strokovnega razvedrila. Za relativno nizko ceno je možno izdelati računalnik z izredno dobrimi tehničnimi karakteristikami. Bogata programska oprema, ki jo je možno izvajati v okviru operacijskega sistema CP/M, je zagotovila, da je računalnik MMS uporaben na vseh področjih: računovodstvo, bančništvo, grafika, podatkovne baze in njihovo urejanje, vzgoja, študij računalništva, znanost in vodenje projektov, proizvodnja in statistika, osebni dohodki in različni programski jeziki.

### TEHNIŠNE KARAKTERISTIKE OSNOVNA VERZIJA: tiskano vezje dimenzije 350x215 mm z naslednjimi tehničnimi podatki:

- procesor: Z80 (2.5MHz, 4MHz ali 5MHz)
- pomnilnik: ROM 8K, RAM 64K zlogov
- izpisi: slikovni krmilnik za 24x80 znakov (video)
- zunanji pomnilnik: 4x disketna enota DSSD (WD 1771 za 8, 5.25 ali 3.5-palčne diskete)
- vmesniki: 2x RS232 sinhroni ali asinhroni vmesnik (SIO) 50 do 19200 baudov, paralelni vmesnik PIO (možna simulacija vmesnika Centronics), vmesnik za tipkovnico
- programska oprema: monitor (v EPROM), generator video znakov (v EPROM), operacijski sistem CP/M 2.2 (disketa)

- ### RAZŠIRITVE:
- pomnilnik: RAM 256 K zlogov
  - RAM disk: 256 K ali 1 M zlogov s potrebno programsko opremo
  - univerzalni krmilnik za gibke diske: 4x disketna enota DSSD (WD 2791 za 8, 5.25 ali 3.5-palčne diskete)
  - krmilni modul za trdi disk: 2x (5M – 200 M zlogov) s priključitvenim konektorjem po standardu ST506
  - barvna grafika: 512x512, 16 barv, »look up« tabela, zvok, ura realnega časa z baterijsko zaščito, hitrost risanja 1.5 M piksel/s
  - univerzalni EPROM programator: za vse tipe EPROM, EEPROM, mnoge »single chip« mikroročunalniške z vdelanim EPROM
  - programska oprema: operacijski sistem CP/M 3+

## Nadaljevanje s strani 7

(prečrtane ali ne) in izberemo nacionalni nabor znakov. Teh pozna NEC 12 in najbrž je odveč poudarjati, da jugoslovskega ni med njimi, da pa ima Danska celo dva nabora znakov.

S posebno ročico na valju za pomikanje papirja zelo enostavno vstavljamo posamezne liste papirja. Tiskalnik nastavi list tako natančno, da popravljanje sploh ni potrebno.

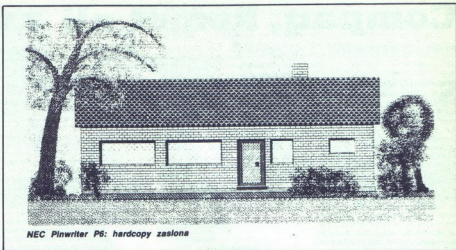
Traktor za brezkončni papir žal ni vdelan in ga je treba kupiti posebej. To je skoraj nujno, saj so kabli na zadnji strani tiskalnika razporejeni tako nerodno, da ovirajo papir. Sicer pa je delo tiskalnika brez traktorja natančnejše, kar je predvsem pomembno v grafičnem načinu.

Tiskalnik ima vdelan tudi ventilator, ki skrbi, da delovna temperatura pisalne glave ni previsoka. Če zraste temperatura na več kot 90 stopinj C, začne tiskalnik pisati le v eni smeri, ko pa senzor izmeri temperaturo višjo od 105 stopinj C, se tiskalnik ustavi, dokler temperatura ne pade pod 90 stopinj C. Tiskanje se nato samo nadaljuje.

Škoda je tudi, da ima tiskalnik vdelan le en vmesnik. Za tako cenno bi si proizvajalec lahko morda duše privoščil oba vmesnika v istem tiskalniku in mu ne bi bilo treba prodajati dveh modelov, kupa pa ne bi skrbelo, kaj se bo zgodilo, če bo zamenjal računalnik.

## Nabori znakov

NEC zna tiskati v treh osnovnih načinih: navadnem (draft), lepopisnem (LQ) in proporcionalnem (črke zavzamejo na papirju različne širine). V vseh načinih lahko uporabljamo klasične načine pisanja (krepko, poševno, podčrtano...). Uporabnik lahko definira do 128 svojih znakov, vendar veljajo le za tisti način pisanja, za katerega jih definiramo. Če torej



NEC Pinwriter P6: hardcopy zaslona

med tiskanjem spreminjamo način pisanja, moramo ustrezno spremeniti tudi svoj nabor znakov.

## Ubežne sekvence

O običajnih ubežnih sekvencah, ki jih poznajo vsi tiskalniki, nima smisla izgovarjati besed, zato se raje ustavimo pri bolj neobičajnih.

NEC zna premikati papir v obeh smereh, tako da lahko naredimo pomik za vrstico naprej ali nazaj, če pa nam to ne zadostuje, lahko pomaknemo papir za n/180 inčev naprej ali nazaj ( $n < 127$ ). Izpisane znake lahko povečamo horizontalno do dvakrat. Pri izpisu računalniške črke lepo oblikuje in niso tako oglate kot pri NL-10. Poleg razmaka med vrsticami lahko določimo tudi razmik med posameznimi znaki in tako pišemo besedilo bolj ali manj stisnjeno. Tiskalnik pozna tudi t.i. «elongated print», s katerim tiskamo snako visoke črke, ki pa so horizontalno razširjene.

Če bi radi izpisali tekst, ki ni urejen za urejevalnikom besedil, zna namesto njega vse delo opraviti kar tiskalnik. Z ubežnimi sekvencami lahko nastavimo urejanje po levem ali desnem robu, centriranje vrstic, ali pa popolno urejanje besedila na levem ali desnem robu. Ker je zaenkrat malo urejevalnikov besedil, ki znajo urejati v proporcionalnem načinu, si lahko za izpisovanje takih tekstov pomagamo z enim od načinov urejanja. Če imamo model, ki zna tiskati v barvah, lahko določimo tudi barvo izpisa. Zanimiva je tudi možnost izpisovanja blokov besedila. Z ubežnimi sekvencami označimo začetek in konec bloka AA/velik je lahko do 2 K) in ga nato izpisujemo na poljubnih mestih v besedilu. Možnost je pravna predvsem za izpisovanje naslovov in podpisov na tekstih, ki jih ne obdelamo z urejevalnikom besedil.

## Grafični

Tiskalnik podpira tako 8 kot tudi 24-pinsko grafiko. Prvi način je primeren predvsem za programe, ki so narejeni za običajne tiskalnike. Ker je matrika široka 24 točk, se pri tem načinu uporablja le vsaka tretja iglica. V 8-pinski grafiki imamo na voljo šest, v 24-pinski pa pet različnih gostot risanja. Gostota točk se giblje od 60 do 360 točk na inč.

## Kupiti ali ne, to je sedaj vprašanje

NEC P6 je vsekakor tiskalnik, ki zna več kot večina matričnih tiskalnikov na trgu. Ne le, da je izredno hiter in tih, tudi kvaliteta izpisa je praktično ista, kot pri marjetičnih tiskalnikih. Ponuja obilo možnosti za igranje z obliko izpisa. 24-bitna grafika pa zmora risanje zelo natančnih slik, kar nam da skupaj z urejevalnikom

besedil, ki zna v tekste vključevati tudi slike, res močno orodje za pisanje raznih publikacij, dokumentacije, poročil...

Tiskalnik stane v Veliki Britaniji 500 funtov, v ZRN pa okrog 1600 mark. Cena je velikostnega reda cen najboljših Epsonovih tiskalnikov, kar pomeni, da je NEC skoraj še enkrat dražji od Starovega NL 10, pri tem pa je treba posebej kupiti še traktor za brezkončni papir. Za povprečnega uporabnika, ki tiskalnik večinoma potrebuje za izpisovanje programov ter tu in tam kakšnega teksta, za katerega ni nujno, da je izpisan res vrhunsko, je razlika v cenah tolikšna, da najbrž vsaj ne bo razmišljal. NEC P6 pa je najbrž idealna rešitev za delovne organizacije, saj je izredno kvaliteten in trpežen, njegova cena pa je še vedno mnogo nižja od cene marjetičnih tiskalnikov, ki se jih da kupiti pri nas. Edina težava je ta, da večina programov (tiskalnik samo testirali na atariju 520) tiskalnika ne podpira in je potrebno kar precej dela za prireditve grafičnih funkcij... Pravo, jo dobimo »za veliko denarja veliko muzike«. Vprašanje je le, ali imate dovolj denarja, da si privoščite izvrstni orkester NEC.

Z individualnim uvozom bo zaradi visoke cene verjetno nekaj težav. Rešitev je uvoz v več delih ali pa igra avanture, ki jo večina naših ljudi rada igra. Da ne bo pomote! – Kontrabanta tokrat ne boste igrali z mavrico, ampak z našimi obojnim organi.

Naslovi  
ZRN: NEC Business Systems (Deutschland) GmbH, Klausenberg Strasse 4, 8000 Muenchen 80; 089 93 20 41

Velika Britanija: NEC Business Systems (Europe) Ltd., 35 Oval Road, Londond NW1 7EA; telefon: 01-267-7000

- To je razmak med znaki 1
- To je razmak med znaki 2
- To je razmak med znaki 3
- To je razmak med znaki 4
- To je razmak med znaki 5
- To je razmak med znaki 6
- To je razmak med znaki 7
- To je razmak med znaki 8
- To je razmak med znaki 9
- To je razmak med znaki 10

To je navadna 10 CPI oblika tiskanja  
Dvakrat horizontalno  
Trikrat horizontalno  
Spet 10 CPI

Dvakrat vertikalno

2 X vert. & horiz.  
2 X vert. & 3 X hor.

# Compaq, komet ali zvezda?

Predrba: LOJZE ZADRAVEC

Še januarja 1982 ga ni bilo nikjer, leta 1986 pa bo imel že več kot 600 milijonov dolarjev prodaje. Zgodba o Compaqovem začetku je še ena od nadih ameriških zgodovino uspehu. Nič več in nič manj kot štirje emigranti so bili zaslužni za štirje milijone firme. Niso sicer pobegnili iz Sovjetske zveze, samo tlaque pri Texas Instruments so se naveličali in februarja 1982 ročno ustanovili novo podjetje z nemogočim imenom Compaq: Joseph „Rod“ Canion, James Harris, William Murto in Steven Flannigan so enostavno menili, da zmorejo več in bolje. Našli so še somišljenika Benjamin Rosena, ki mu ni bilo problem razvežati mošnjake. Tvo je že pomagali pri rasti Lotusovih cvetov in pravkar skrbi tudi za paradoksalno Anso Da se pri Lotusu ni zmotil, je znana stvar. Ansa pa sicer še ni tak hili, da bi sodila na lestvico American Top 5, vendar tudi v manjših razmerah dela dobro. Še najbolj pa mu je uspelo prav s Compaqom. Še negodno dete je namreč že v prvem polnem letu delovanja prodalo morje računalnikov, vrednih 111 milijonov dolarjev – ameriški all-time record!

## Rojstvo

V prvih dneh januarja 1982 je bilo Canion, obklovačec, in Harris sta osnutek za njun prvi prenosni računalnik narisala kar na prtiček v priljubljeni houstonški slaščičarni House of Pies. Skupaj z Murtom so potem razvili strategijo. Hitro so namreč ugotovili, da trgovci niso zainteresirani za prodajo računalnikov, za katere ni nobenih programov. Nič več časa jim ni zvela

tudi ugotovitev, da tudi programski čarovniki ne mislijo pisati programov za še enega novinca. Kje pa je na voljo veliko programov? Pri IBM. In ker je IBM pol leta, preden se je začelo njihovo razmišljanje, dal na trg mnogo obetajočih PC, so se odločili nekako predati in dodati izdelek IBM

Rekrutirali so ljudi, ki so lahko ne le živeli v ječi kompatibilnosti, ampak so bili v takoj uspešnosti celo kreativni. Niso jemali mlečnozobe, pa če bi bili še tako dobri. Podpredsednik Steven Flannigan razlaga: »Vsi ovisetniki bi radi detali za IBM. Štridesetniki so sicer že debeli in plašni, vsekakor pa nimajo tako norih idej.« Programerji, ki delajo za Compaq, imajo v povprečju 15 let izkušenj!

Pa tudi s tako posadko narediti PC kompatibilnega ni bilo lahko. Že res, da mikroprocesor in operacijski sistem nista povzročala težav. Vzeli so pač Intel 8088 in MS-DOS Gatesovega Microsofta. Ampak BIOS je IBM zaščitil, kako pa brez njega? Compaqovi inženirji, ekipa, ki je medtem narasla na 15 programerjev, so se trudili devet mesecev in porabili za delo milijon dolarjev.

Inženirski posli so bili končani oktobra 1982. Takrat je stopil v akcijo William Murto, strokovnjak za marketing. Povezal se je z verigo veletrgovin Sears Roebuck in verigo specializiranih

računalniških trgovin ComputerLand ter tamkajšnje odgovorne poskušal navdušiti za stvar. Oktobra je letel po Ameriki s prototipom in nalogo, prepričati odgovorne pri ComputerLandu o uporabnosti prenosnega PC kompatibilnega. V hotelski sobi v New Orleansu, kjer bi se moral srečati z morebitnimi bodočimi prodajalci, je ugotovil, da je edina primerna vtičnica v kopalnici. Ni bilo druge pomoči, računalnik je postavil na stranično školjko, povabljence stisni v malo kopalnico in predstavitev se je začela. Na koncu je bil uspeh izjemni!

Za prodajne uspehe je še najbolj zaslužna Compaqova distribucijska strategija, njegovo naslanjajo na trgovce. Že res, da je Compaq enake proizvode kot IBM ponujal po nižji ceni, vendar je bilo važnejše to, da Compaq ni poskušal prodajati preko svojih ljudi ali pa neposredno velikim podjetjem. In še nekaj je precej važno, pri prodaji po priporočeni ceni dobi trgovec pri Compaqu 36 odstotkov, pri IBM pa le 33 odstotkov. Da bi tudi v prihodnosti bil Compaq v dobrih odnosih s trgovci, je kupil »Sparky-Sparka, 52-letnega veterana, vzornika in tekmeča IBM, ki je bil pri IBM odgovoren za odnose s trgovci. Sparky ni bil poceni. Ponudil si mo 100.000 Compaqovih delnic po desetkrat nižji ceni na trg, 100.000 dolarjev za prestop in še za 30.000 višjo plačo, kot jo je imel pri IBM – skupaj 150.000

dolarjev. Videti pa je, da se je izdelek sploščal. Konec koncev je Compaq močnejše najti v večini ameriških prodajnih PC.

## Najstnikši Compaq

Compaqova rasti v prvih mesecih je bila tako hitra predvsem zaradi ponudnje resolucijski PC, onih iz tovarni Compaq, vendar pa se je povzdrževanje nadaljevalo tudi leta 1984, ko je bilo dodajati tudi IBM PC. Da utegne biti Compaq več kot muha enodnevna, pa se je pokazalo, da je tudi IBM prišel na svetlo s prenosnimi PC. Imel je oranžni zavesa (veliko uporabnikov ga ne mara), na njem je takoj manj programov za PC in za »poslastico« je bil še težji.

Junija 1984 je Compaq rodil tudi prve neprensne modele, seriji Deskpro. Tudi računalniki Deskpro niso bili na svetlo s prenosnimi PC. Imel je pogovorni s trgovci in iz drugih tržnih razgovorov. V reklamno je vložil 19 milijonov dolarjev, toda zaveza je komaj začela vračati vložen denar, ko je IBM predstavil PC AT. Da bi bila stvar še malce težja, je Compaq po devetih mesecih razvoja namiznega, tudi laptop, imenovanega računalnika, načrte spravil nazaj v predal. Canion razlaga: »Podjetje bi rado piccolo računalnik, ki bi bil poleg tega še perseno lahek. Ampak narediti kaj takega za sprejemljivo ceno ne gre. Zal ne!«

Torej so se morali lotiti še razvoja AT kompatibilnega. Danes že vemo, da so bili tudi tu uspešni.

## Odrasli Compaq

Pri tako uspešnem in samozavestnem Compaqu smo pričakovali, da bo



## Compaq Deskpro 386

ČRT JAKHEL

Deskpro prinaša trgu PC nove razsežnosti. V primerjavi s PC/XT (8088, 4 MHz) je že AT (80286, 8 MHz) nekaj drugega, novi stroj pa predstavlja premerio 80386 (32 bitov) na 16 MHz proizvajalec trdi, da je Deskpro vsaj dvakrat hitrejši od AT. Trdi disk spravi 40 ali 130 Mb, dostopni časi so pod 30 ms (AT: 40, XT: 85–90). V pomnilniku je 1 M protora, brez težav gre do 10 Mb, največ pa do 14 Mb. Od tega se da

8 Mb uporabljati po Lotusovi/Intelovi/Microsofovi normi razširjenega pomnilnika, kar pomeni plodna tla za programe, kot je 1-2-3. Da je to mogoče, ima operacijski sistem (sicer MS-DOS 3.1) nekaj posebnosti, prav pa pride tudi to, da 80386 brez težav dela z virtualnim pomnilnikom. Po želji se

dobi matematični koprocesor 80287 za 4 ali 8 MHz. Možnosti se zdijo sanjske, a še ne pomenijo polne izkoriščenosti dobrot, ki jih prinaša uporaba 80386 – temu se ne godi nič bolje kot starejšemu bratu 80286, oba sta namreč podložna 8086, da se obrani združljivost. In čeprav se lahko ob moč-

nejša stroja spustita na nivo XT, se med sabo slabo razumejo. Nekateri programi, ki dobro izkoristajo prednosti AT, torej na Deskpro ne bodo tekli. To velja tako za načrtovani MS-DOS 5 kot za uporabniške programe. Težavo bi se sicer dalo odpraviti z operacijskim sistemom, ki bi na 386 emuliral 286, pa je, da se bo prihajajoča produkcija omejila na zadnje.

Hkrati z 386 je Compaq pokazal grafično kartico, ki deluje kot Hercules (720/348, mon) ali EGA (do 640/350 točk, 16 ali 64 barv). Kartica prinaša tudi Microsoftovo tehnično novost, t. i. Inport – integrirano vezje, ki predstavlja celoten vmesnik za delo z miško.

Deskprojeva cena je premo sorazmerna zmoglostim: računalnik z 40 Mb prostora na trdem disku stane brez operacijskega sistema, grafične kartice in monitorja skoraj 20.000 Dm, z 130 Mb pa čez 25.000 Dm. Sicer pa: tisti, ki Deskprojeve potenciale potrebuje, najrjuma prepriha v denarnici.

prej ali slej stori kako bolj smelo poslovno potezo. In jo je res! 8. septembra je na svetlo spravil Deskpro 386, prvi na 32-bitnem procesorju Intel 80386 temelječi računalnik. Veliko govoric nas je pripravljalo na novost. Pa jim kar nekako nismo hoteli verjeti. Kako naj bi Compaq naredil nov računalnik, ko pa od IBM nismo slišali nič najmanjše novice o tem, da bi tudi sami delali kaj takega?

Kako je bilo na predstavitvi? Zelo živahno! Povabljeni so se zbrali v najljepši newyorški diskoteki Palladium in ob ugodnih taktih skladike Neila Diamonda «Headed for the Future» so bra se pred gosposi poslušali izvršni posnetki prvih poletov bratov Wright vse do uspehov NASA, od konja do elegantnega Ferrarija, od abakusa do zvezde večer, Compaqovega novega računalnika Deskpro 386. Nato se se prizgale luči in v areno je stopil pod Canion ter začel s svojim delom skrbno pripravljene nastopa. Vrhunec njegove monodrame je bil stavek: «Danes smo vas vpejali v tretjo generacijo osebnih računalnikov.»

Rod sploh ni dosti pretinalav. Lahko bi trdili, da je bil prvi osebni računalnik z apple iz prejšnjega desetletja, začelni druge generacije pa slavni IBM PC izpred petih let. V primerjavi z njima je deskpro nekajkrat boljši, hitrejši, zmogljivejši... Pri Compaqu pravilo: 386 je brezopogovna prihodnost. Vse ostalo je, kar se bo pač spremenilo. «Vsi brez izjeme se strinjajo. Vprašanje je, kdaj se bo pač prihodnost začela.»

## Kako najraj

Resnični preboj bo šele takrat, ko (še) ga bodo videli proizvajalci programske opreme priznali za industrijski standard. Prvi, tjer pa utegne biti najtežje. V industriji, kjer je dostoj v ogromni večini standard določala IBM, ne j bi ta kar senkrat nek prepotentnejši iz Houstona. In še preden na tem področju stori kaj IBM! Je to sploh mogoče?

Compaq se sicer sploh ni tako neznan. In da ne bi mogel uspeti. Že v začetku leta 1985 ga je Intel zaprosil za sodelovanje pri zagotavljanju kompatibilnosti novega procesorja 80386 z vsemi stariimi. Industrija namreč priznava, da je Compaqov test kompatibilnosti najboljši od obstoječih. Za povrh je zelo zgodaj ugotovil, kaj 80386 omogoča, tako zgodaj, da mu potencialna konkurenca zagotovo ni mogla slediti.

Razvoj 386 je zahteval nekako 8 milijonov dolarjev, stroški za marketing pa znašajo 5 milijonov. 13 milijonov dolarjev je najmanj za polovico več, kot je Compaq vlagal v razvoj novih strojev, dosedaj, pa tudi 2,5 milijona dolarjev, zapravljenih v dnevne predstavitve, evforiji, je več, kot bi si si kak laični predstavljali. Stroški se tukaj še ne nehaajo. Nakupili so si za najmanj 100 nahrbtnikov procesorjev 80386 (po 300 dolarjev kos) in kar nekaj kontajnerjev diskovnih pogonov, da bi zadovoljili povpraševanje. Če pa Compaqov 386 pride, kaj potem? Za podjetje ne bo tako strašno. Nekaj mesecev zmanjšanih zaslužkov, izgubljeno leto rasti – nimita morda o nezmotljivem Compaqu... Težav je pač preveč, da bi se ubadali še s polnim novincem. Podpora si je Compaq skušal zagotoviti tudi z najbolj slavimi imeni računalniške industrije, povabljenimi v Palladium. Prišli so



## Amiga se širi...

Amiga je dobila Turbo Chassis, škatlo, ki vsebuje 68020 v taktu 14,28 MHz in 68881, matematični koprocesor s plavajočo vejico. Na ploščici so 32-bitne podatkovne, naslovne in DMA zveze. AmigaDOS baje popolnoma podpira oba dodana procesorja. V škatli je še 512 K/32 bit statični RAM, 20 Mb trdi disk s kontrolerjem SCSI in napajalnik. Lahko dobite tudi disk z 40 Mb. Zaveza se priključi na računalnik preko 100-polne razširivne magistrale. Turbo Chassis ima še dve vrati, ki so predvidena za bodoče razširitve (kdor z velikim ni zadovoljen, dobi še več). Chassis stane 5475 US dolarjev, lahko kaj kupite same ali pa z amigo in monitorjem. Za prospekte pišite Computer Systems Associates, 7564 Trade St., San Diego, CA 92121, USA. Kaže, da amigo sploh ne gre tako slabo; pač odvisno, na kateri strani leži stojiš.

## YES v prenosni različici

predsednik Intelja Gordon Moore, soustanovitelj Microsofta Bill Gates, predsednik Ashton-Tate in Lotus in še deseterica manj važnih moč

Trgovci so 386 že dokončno vzeli za svojega. Ni čudno, ko pa jim zagotavlja vsak prodani model celih 3.000 dolarjev profita. Še najbolj pa je uspeh odvisen od Microsofta. Za vsak programov za novince potrebujemo programske hiše nov operacijski sistem in ne doseganja Microsoftova proizvođa MS-DOS 3.0 ali Xenix System V-286, ki ju uporabljajo sedajni računalniki. Microsoft upa, da bo naredil do konca prvega četrtileta leta 1987 operacijski sistem Xenix 386, posebej namenjen za 80386. Sicer pa za prodaja razvojne sisteme za 80386, ki utegnejo priti pri prepisovanju programov za sposobne 80386.

Še važnejši bo novi, za 80386 prirejeni MS-DOS. Če se bo izkazal za prijetno do uporabnikov, tega ne ve niti Bill Gates. Kmalu gotovo ne, ko pa niti MS-DOS za 80286 še ni narejen. Zamujata že dve leti, za še vedno ni napovedane predstavitve. Zaradi Microsoftovih zamud in težavosti pisanja novih programov se utegne zgoditi, da jih ne bo še meseca ali celo leta. Daljša ko bo zamuda, daljši utegne biti Compaqov boj za priznanje.

## YES v prenosni različici

Philipsov ves je ima v kratkem pojaviti v prenosni verziji, s plazmatiskim zaslonom (kot Ericsson PC) in imenom Lap-Top-Top. Tastaturo se bo, da bo zaveza res silno prenosna, dalo spravit po osnovno anoto. Zaslon je širok kot sam računalnik, le da je 3 cm debel. Sedi na vrhu glavne škatle in se da lepo položiti nanjo, ko pa se vrnete s poti, ga vzdignete in se pretvarjate, da je vaš računalnik normalen «namizni» PC. Cena novotarjeje in datum, ko se bo pojavila v trgovinah, trenutno ništa znana. Pač pa je Philips ponil Yes: z dvema gibkima diskoma in 640 K RAM stane zdaj 6700 namesto 8000 DM. Za ta denar priložijo vmesnike za miš, terdise disk, zraven pa še uro na bateriji in programski paket Open Access. Kdor je pripraviln plačati 8200 DM, dobi še 20 Mb trdi disk. S 512 K RAM, dvema gibkima diskoma, monitorjem, vendar brez Open Accessa je cena 5700 DM.

## Eppur si muove!

Po tolikih letih nespolna vladavine IBM PC z MS-DOS so stvarniki operacijskega sistema pripravljali iskati nove alternative. Posebno vero v prihodnost jim vtilva 80286 – temu na kožo naj bi bil pisan MS-DOS 5 (nekoč je obstajal tudi 4, vendar se Microsoft z njim ni preveč hvalil). 80286 pozna dva načina delovanja: «real» in «protekted». V prvem postane zelo zvesta gotuja 8088 (ta lahko naslovi max. 1 Mb), v drugem pa. prepozna 16 Mb pomnilnika, vendar ni več omejen na stare ukaze. Rezultat: programi zanj so – odvisno od OS – združljivi s tistimi za 8088 ali pa zmorejno veliko več, a na drugačen način,

nr. Xenix. Na prvi pogled očitna rešitev je preklapljalne obeh načinov, vendar je to, kot pravi Bill Gates (Microsoft ustanovitelj) in predsednik, «zelo, zelo kompleksno, ampak našli smo način». Tako naj bi Microsoft božič proslavil v delovnem vzdušju: MS-DOS 5 naj bi se na trgu pojavil v začetku 1987. Objubljajo, da bodo mislili tako na lastnike 8088 PC kot na 80386. Poleg tega predvidevajo, da bodo Windows (večopravilna alternativa GEM Deskopu) postala popularna s formulo MS-DOS = Windows + DOS. Bomo videli. Kogar zanimajo podrobnosti, naj odpre oktobrski Chip na 82. strani.

Na Interbiuroju smo prvič v Jugoslaviji videli IBM RT PC, na tehnološki RICS temelječi «tehnični računalnik. Tačas pa so ga v ZDA še drastično pocenili (z 11.700 na 7.900 dolarjev) in predstavili še zmogljivejšo verzijo. Pocenilo so tudi operacijski sistem, na 2.995 dolarjev. Uporabniki pa kljub temu niso zadovoljni. Premalo zmogljiv je za ceno povprečnega ameriškega avtomobila!

Da ne bi slučajno mislili, da je štrikratno zvišanje cen možno samo v Jugoslaviji. V slabih treh tednih se je nekaj tak groznega zgodilo v ZDA. Da, v Ameriki, kjer se je cena 256 K DRAM čipov povečala z dveh na osem (1) dolarjev. Ni nam znano, kako je to vplivalo na število zvišanih zlomov in samomorov v ZDA.

## Nekoč so bili megabyti

Se še spominjate časov, ko smo spoštljivo govorili o Mb, gigabyti pa so bili sinonim za nedosegljivo? Prihajajo TERAByTi. 1 TB = 1000 Gb = 1.000.000 Mb. Ufff. Novi pojem uvaja nizozemska DOCdata, sistem se imenuje DOCwheel. Obroč, ki pokriva 3 kvadratne čevlje, vsebuje 128 optičnih kaset. Predvidoma je poljuben zapis na poljubni kaseti dosegljiv v 10 sekundah. Kasete so približno tako velike kot navadne zvočne (4,4 x 2,2 x 0,6 palcev) in spravijo do 8 Gb; torej je razmerje volumen/kapaciteta 30-krat ugodnejše kot pri optičnih diskih. Kasete so v hermetično zaprtih kovinskih ohišjih. DOCwheel naj bi uporabljali predvrem veliki sistemi, priključki pa bodo enaki kot pri večini obstoječih tračnih enot. Na kasete se bo dalo pisati le enkrat, vendar pri DOCdata menijo, da to ni problem, saj se že zdaj magnetni trak večinoma uporablja na tak način. DOCdata bo napadla v letu 1987. Vas zanima? DOCdata N. V., Box 1021, Maaskede 11, 5900 BA Venlo, Nizozemska.

## Problemi s papirjem za tiskalnik

Imate tiskalnik in nimate nekondžnoga papirja. Tudi v službi so vas že siti, ker neprestano nosite domov vrećke in polne torbe. Problem, ki vas traže, je cena papirja. Neskončne obrazce prodaju je u škatala po 2000 kosov, kar pa je za računalničarja – ljubitelja prevēć. Previsoka pa je tudi čena, saj škatala papirja stane okrog 30.000 din.

V prodajalnic Narodnih novina iz Zagreba lahko manjši porabnik papirja prav pocene priđe do reprodromateriala, ne da bi izgubili dober glas. Računalnički papir, pakiran v škatala po 200 listov, lahko naročite za vsega 500 din. Preprost račun pokaže, da list papirja stane samo 2,5 din, kar je šestkrat ceneje, kot će bi papir kupovali na "debelo". Računalnički obrasci so na eni strani beli, na drugi strani pa so potiskani z modrim linijama, ki se pri fotokopiranju izgubju. Format je standardni A4 ali 12 inč. Papir lahko naročite na naslovij: Narodne novine, Ul. Proleterskih brigada 60, 41000 Zagreb ali Narodne novine, Beogradski trg 15, 51000 Rijeka.

Čudeži so možni tudi v računalništvu. Najpovēšje se je zgodil pri IBM. Še letos bo Big Blue zmanjšal število zaposlenih za 4000 delavcev, naslednje leto pa še za dodatnih 8000! IBM odvećnih ne bo odpuščal, pać pa jih bo le prej upokojil. Zadeva sploh ne bi povzročila nobenih komentarijev, če ne bi bil IBM doslej znan kot podjetje, ki celo v ZDA (kjer je to sicer normalno) ne odpuščal delavcev. Viške so ponavadi zaposlovali drugje in tako delali po tipičnem japonskem vzorcu – "doživljenske" zaposlenosti.

Tudi ameriška vlada je bolj počasne narave. Tako je šele pred kratkim prišlo na dan, da so zvezni ameriški organi leta 1985 kupili več kot 67.000 mikroročunalnikov, najveć Zenitovih in IBM. Leta 1984 so jih npr. samo 37.000, leta 1983 pa slabih 8.000.

## SH 204: Atarijev pogon za trdi disk

TOMAŽ SAVODNIK

**K**malu potem, ko je Jack Tramiel objavil moć in kvaliteto skoraj zastoj, je začel objubite tudi izpolnjevali. Atari 520 ST+ preprićljivo vodi na lestvij najbolje prodajanih osebnih računalnikov, atari 260 ST pa se bori za prvo mesto v konkurenci hišnih računalnikov. Med drugim je firma pred enim letom objavila trdi disk. O njem je takrat vsak pošten heker, navajen na disketno enoto s 720 K, mislil, da je le nepotrebnorazkošje, ki je povrh še predragor.

Do danes pa so se stvari nekoliko spremenile, da se naš heker počuti utesjenega še na 1 Mb. Ko bi imel vsaj 20-megabytni trdi disk, sanjarji in se v teh kritičnih trenutkih spet obrne na Tramiela, ki je nekdo objavljaj trdi disk. In glej čudež, v trgovinah (ne trditve se pri nas) se je pojavilo pravo 20-megabytno čudo. Poln mešanij občutkov se odvpravi heker v trgovino in za 1990 mark odnese iz nje srednje velik paket. V spominu ima leto 1980, ko je njegovo podjetje za veliko več denarja kupilo 10-megabytni trdi disk za računalnik Apple II. To je bil 8-palčni pogon velikosti dveh škatal za čevlje, vendar je potreboval še transformator, ki po velikosti ni dosti zaostajal. Delo z njim ni bilo ravno prijetno, saj ste imeli občutek, da vam okoli glave letajo muhe. Glasnost, velikost in hitrost niso bile ravno vrline istega trdega diska, vendar je bil takrat "čudo tehnike". Kljub temu pa se kmalu preprićate, da ste se po nepotrebnem balij. Iz zajetne embalaže vzamemo čvrsto kovinsko ohišje sive barve. V tem 80x180x380 mm velikem ohišju je pogon s transformatorjem in kontrolerjem. Na sprednji strani

opazimo le rdeće svetleće se diodo. Na hrbtni strani so prikljućek za napetost, stikalo, varovalka in prikljućek za vmesnik. Poleg tega dobite zraven še kabel za vmesnik, kabel za prikljućitev na omrežje 20 V, 3,5-palčno disketo in nemška in/ali angleška navodila.

### Hardver

Ko odvijemo nekaj vijakov, smamemo pokrov ohišja. Če stvari pogledamo nekoliko pod kožo, ugotovimo, da v škatali ni nič revolucionarnega. Tudi disk je 5,25-palćen (Seagate ST 25). Kontroler je izdelala firma Adaptec. Izdelan je okoli mikroprocesorja 8085 in nekaj TTL solate. S hardverskega stališća je na kontroler možno prikljućiti še en pogon. Ne glede na to, ali so Atarijevi strokovnjaki to možnost spredelovali ali ne, je to zelena luć za vse, ki jim hardverska žilica ne da miru. Ostanenam še mala Atarijeva ploščica in hlajenje. Ta ploščica je vmesnik med kontrolerjem in Atarijevem vodilom DMA. Za hlajenje skrbi ventilator, ki je pritrjen kar na kovinsko ohišje.

### Prikljućitev

Pri prikljućevanju se postavlja vprašanje, zakaj dela Atari tako kratke kablj. Zaradi kratkega kabla nam ne ostane nič drugega, kot da postavimo pogon levo od računalnika. Drugaće pa prikljućevanje ne pomeni hobene teževe in napake skoraj ne morete narediti. Paziti moramo le na to, da vkljućimo disk pred računalnikom. Da je pogon vkljućen, boste ugotovili (slišali) po brenenju ventilatorja. Če se vam pri prikljućevanju karkoli zatakne in ne veste kako naprej, vam o prikljućevanju, formatiranju, razdeljevanju in uporabi pogona "vse" povedo na osmih straneh nemških navodil. Ta navodila so vzor skrevćanja na najnujnejše in vam prav gotovo ne bodo zadostovala. Na srećo so priložena tudi angleška uporabniška navodila (Owner's Manual), v katerih je na štiridesetih straneh zares napisano vse o SH 204.

### Brez ROM ni veselja

Če imate TOS na disketi in ne v ROM in ste s tem zadovoljni, vas bo to odstavek razočaral. Ko preberete navodila, namreć ugotovite, da pogona brez TOS

v romih ni mogoće instalirati. Morda bi se ta problem vendarje dalo rešiti, vendar nam to ni uspelo. V najboljšem primeru nam je računalnik sporoćil napako, da se diska ne da instalirati in da bi bili morebitni podatki na trdem disku unićeni ali pokšodovani. Zakaj Atari ni mislil na lastnike računalnikov v TOS na disketah, mi ni jasno, precej verjetno pa je, da so tako poskusiš zaslužiti denar, saj ste prisiljeni kupiti TOS v romih. Ko je računalnik opremljen po zadnji modi (z romi), instaliranje pogona ne pomeni več nobenega problema. Naprava vsaj deluje brez težav in na Desku se pojavje simbol trdega diska.

### Uporaba

Na priloženi uporabniški 3,5-palčni disketi so poleg pogonskega programa tudi program "SHIP", ki prenese glavo pogona v neuporabljeno področje, in program HDX ter zanj potrebne datoteke. Formatiranje in razdelitev pogona v oddelke (partitions) je s programom HDX hitro in enostavno. Pred vsako kritično operacijo in v programu HDX se moramo večkrat preprićati, da zares vemo, kaj delamo. Pri normalni uporabi diska se na Desku pojavi kot pogon C. Če pa smo pogon razdelili na primer na tri oddelke (partitions), se pojavijo kot pogoni C, D in E.

### Hitrost

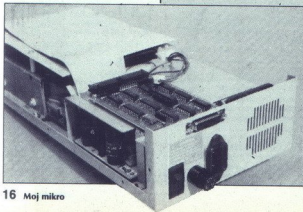
Seveda pričakujemo pri uporabi trdega diska bistveno povećanje hitrosti dostopa do zunanjih podatkov. Pri Atariju se hitrost v najboljšem primeru, to je pri nalaganju slike, poveća petkrat. Nalaganje programa 1st Word z diskete traja kar 24,5 sekunde, medtem ko se isti program naloži iz trdega diska v 5,5 sekunde. Tisk datoteke, dolga 280 K, se disk naloži v 126 sekundah, s trdega diska pa v 78,5 sekunde. Za primerjavo naj navedem še podatke za RAM disk. Program 1st Word se naloži v 3,5 sekunde, datoteka pa v 72 sekundah.

### In pri nas ...

Vse tiho je bilo. Tako bi lahko nadaljevali ta mednaslov, saj v Mladski knjigi trdega diska SH 204 ne bodo prodajali. Za vse, kar se jim zdi Nemćija predaleć, pa velja objuba, da bodo konec leta začeli prodajati 40-megabytni trdi disk. Cena? Cena objubljenega diska še ni znana.

**Hvalimo:**  
– UGODNA ČENA  
– ENOSTAVNO  
PRIKLJUĆEVANJE  
– HITROST

**Grjamo:**  
– Glasnost ventilatorja  
– Prekratke prikljućne kable





## TIT TURNŠEK, Iskra Delta

Na letošnjem sejmu Interbio se ISKRA DELTA ni predstavila samo z novim imenom IDC in z novim, poenotenim designom aparature opreme, pač pa tudi kot proizvajalec široke palete kompatibilnih gradnikov informacijske tehnologije: od aparature opreme do aplikacijskih rešitev.

Boj kot v prejšnjih letih je mogoče razpoznati dolgoročno usmerjenost konvergentnega razvoja IDC, katerega plod je enotna arhitektura informacijskih sistemov.

**Enouporabniški sistem TRIGLAV - MODEL PARTNER** se je z novo generacijo, kompatibilno s standardom VT100 in možnostjo enostavnejše grafike, razvil v delovno postajo, kompatibilno z večjimi sistemi DELTA: TRIGLAV, DELTA 800, DELTA 4850 in GEMINI. PARTNER kot inteligentni terminal znatno povečuje moč in prepusnost takih sistemov v primerjavi z rešitvami na neinteligentnih terminalih. Skupaj z ustreznim sistemsko programsko opremo, ki definira povezavo delovne postaje z večuporabniškim sistemom, se polno uveljavlja enouporabniški sistem kot integralni del informacijskega sistema.

Povezava enouporabniške delovne postaje PARTNER z večuporabniškimi sistemi je podprt z vrsto aparaturnih in programskih proizvodov, od katerih naj omenimo LAN-F, ki omogoča povezovanje sistemov PARTNER v lokalne mreže, paket za enostavnejšo grafiko ter generator aplikacij FORMATIX, ki - vrnosen z večjih sistemov DELTA - teče sedaj tudi na PARTNERju in omogoča manj izkušenemu uporabniku delovne postaje samostojno generiranje aplikacij.

**TRIGLAV - večuporabniški modeli**, zasnovani na vodilu VME, so tehnološki dosežek, ki sodi v sam vrh svetovne tehnologije, kar je razvidno tudi iz odmevov v tujini in izvornih uspehov.

Na TRIGLAV smo prenesli informacijska orodja IDA, tako da lahko sedaj pod operacijskim sistemom DELTA/M na TRIGLAVU z modulu J-11 tečejo vse aplikacije kot na sistemih DELTA 800, s tem da je uporaba TRIGLAVA smotrna, če ni več kot 8 uporabnikov. Na operacijski sistem UNIX in XENIX se je razvila vrsta grafičnih paketov, tako da je tudi TRIGLAV močna grafična postaja in emulira vrsto grafičnih terminalov.

V IDC posvečamo posebno pozornost operacijskemu sistemu UNIX, načrtujemo njegove portacije na druge sisteme DELTA, hkrati pa tudi prilagoditve aplikativnih rešitev. Na zgornjem nivoju se je dosežanim družinam sistemov DELTA pridružil nov sistem **GEMINI**. Gre za večprocesorski sistem s skupno bazo podatkov, na katerega lahko priključimo od 64 do 100 uporabnikov. Na sistem priključujemo 300 in 600 Mb diske. Prepusnost celotnega sistema lahko primerjamo s sistemi, za katere se tipično navaja moč 3 do 5 mips.



## IDC, novo ime in nova poenotena arhitektura

*ISKRA DELTA je na letošnjem sejmu Interbio predstavila širok spekter kompatibilnih gradnikov informacijske tehnologije - od aparature opreme do aplikacijskih rešitev. Najvažnejše od njih so opisane v naslednjem članku.*

**Informacijskim orodjem IDA smo** zadnjem letu dodali **podatkovni slovar IDA-LEKSIKON**, ki je postal močno orodje za razvoj informacijskih sistemov, pomeni pa bo tudi osrednjo oporno točko v nadaljnjem razvoju informacijskih orodij IDA

**Nova verzija IDA-BAZA** je prilagojena za uporabo v sistemih CLUSTER in računalniških mrežah ter predstavlja prvo uporabno verzijo v smeri distribuiranih baz.

Na letošnjem sejmu Interbio smo predstavili tudi nova domača **VT220 in VT100 kompatibilna terminala** v novem obliju. Razvita je izboljšana verzija šalterske delovne postaje, podana pa je tudi usmeritev v vrsto različnih, posebnim zahtevam prilagojenih terminalov (POS).

S ponudbo novih in izpolnjenih aplikacijskih rešitev - aplikacij v IDC dopolnjujemo ponudbo gradnikov informacijskih sistemov, konstruiranih po enotnih principih ARHITEKTURE IDC.

Gradniki na aplikativnem nivoju so zasnovani na funkcionalnem razčlenjevanju informacijskega sistema na podсистeme, aplikativna področja in funkcije.



Prepletajo se van\*

IDC ARHITEKTURA definira osnovne podсистeme:

- POSLOVNO-PROIZVODNI
  - UPRAVLJALSKO-KOMUNIKACIJSKI
  - SAMOUPRAVNO-INFORMACIJSKI
- Znotraj vsakega od teh treh osnovnih podsystemov so definirana funkcionalna področja.

Tako so znotraj **POSLOVNO-PROIZVODNEGA** podsystema definirana naslednja funkcionalna področja:

- RAZVOJ IZDELKOV IN TEHNOLOGIJE (CAD/CAM)

- UPRAVLJANJE PROIZVODNJE
- PROCESNO VODENJE
- PRODAJA IN SERVIS
- SPREMLJANJE UČINKOV PROIZVODNJE

Vsa zgoraj našeta funkcionalna področja lahko definiramo v sklopih vseh organizacijah ne glede na njihovo dejavnost. Tako lahko npr. -vodenje proizvodnje- identificiramo tudi v tipično hotelski, trgovski ali pa, recimo, zdravstveni organizaciji. Konkretno izvedbo posameznih aplikativnih gradnikov se lahko od panoge do panoge znatno razlikujejo.

**UPRAVLJALSKO-KOMUNIKACIJSKI** podsystema zajema:

- MEDESEBNO KOMUNICIRANJE delavcev na različnih nivojih vodenja in odločanja
  - obveščanje delavcev, le-to pa pomeni usmerjeno distribucijo in dostopnost različnih informacij posameznim delovnim skupinam.
- Običajno gre za tri vrste informacij:
- TEHNOLOŠKE INFORMACIJE
  - PRÁVNE INFORMACIJE
  - MARKETIŠKE INFORMACIJE
  - PROJEKTI MANAGEMENT
  - PREGLEDI STANJ (proizvodnje, poslovanja)

**SAMOUPRAVNO-INFORMACIJSKI** sistem zajema:

- SPLOŠNO OBVEŠČANJE DELAVCEV, ki omogoča distribucijo in dostopnost informacij vsem delavcem brez omejitve. Gre za:
- splošne samoupravne akte
- sklepe samoupravnih organov
- materiale za delavske svete

Za podporo upravljaškemu informativnemu in samoupravnemu sistemu smo razvili dva paketa: - ELEKTRONSKA POŠTA, ki rabi za podporo medosebno komuniciranju in usmerjenemu obveščanju

- INTERNI VIDEOTEXT, ki rabi za podporo informiranju

Osnovna ideja aplikativnega področja je grupiranje tesno povezanih aplikacij s skupno bazo podatkov, ki jih implementiramo na istem računalniškem sistemu oziroma jih ni smotno razporediti po različnih sistemih. Celoten informacijski sistem na logičnem nivoju dobimo s povezavo posameznih aplikativnih sistemov, kar je na fizičnem nivoju realizirano s povezavo računalniških sistemov v mrežo. Tako dobimo tipično arhitekturo IDC informacijskega sistema, značilnega za posamezne panoge.

Danes v IDC razpolagamo z vrsto gradnikov na aparaturnem, sistemskem in aplikativnem nivoju. V naslednjem obdobju bomo obstoječemu pridružili še nove in tako v sodelovanju z uporabniki dogradili značilne nivoje za posamezne informacijske sisteme.

\* Strani, namenjene našim poslovnim partnerjem, ki želijo predstaviti svojo dejavnost na področju informatike

# WordStar: predradnosti, težave in izzivi

PRÉDRAG SIMIĆ

**W**ordstar je verjetno najbolje zastopan urejevalnik tekstov na današnjih mikroračunalnikih. Z njim so bile naslovene tisoče knjig (med njimi tudi »2010: druga odiseja« Arturja Clarka); o njem je napisanih več deset knjig – od »WordStara za tajnice« do »WordStara za specialiste«, na Zahodu od strojeplisk praviloma zahtevajo usposobljenost za delo s tem programom; z njim napisanih tekstov ni treba ročno zlagati, saj je večina računalniško podprtih tiskarn podprta prav z njim. Skratka: WordStar je postal neuradni industrijski standard pri urejevanju besedil in to je tisto, zaradi česar bo še dolga leta aktualen. Ali to pomeni, da je najboljši? Med programi pod CP/M najbrž, na IBM PC pa verjetno ne, saj tu vlada ostra konkurenca bolj modernih urejevalnikov, kot so Microsoftov WORD, PC Text 2.0, Zanimivo je, da njegov naslednik, WordStar 2000, s katerim je MicroPro (33 San Pablo Avenue, San Rafael, CA 94903, USA) poskušal odgovoriti na te izzive, ni dosegel niti približno take popularnosti.

## WordStar na hišnih računalnikih

Razlog, da si bomo podrobno ogledali ta program, je pojav nove generacije hišnih računalnikov, opremljenih s sistemom CP/M in hitrimi disketnimi enotami (amstrad, commodore, atari idr.), ki je omogočila večjo uporabo CP/M programov – tudi WordStara – v Jugoslaviji. Govorilo bi namreč o verziji 3.0 (1981), ki jo največkrat srečamo.

Hekeji, ki so se prvič seznanili z urejevanjem besedil ob delu s Taswordom in Visaworitom, bodo najbrž dobili vtis, da gre za kompliciran in za spoznanje starmodern, a zagotovo močan in profesionalno narejen program, katerega uporaba zahteva precej znanja (WordStar ima več kot 140 ukazov, priročnik pa največkrat več kot 200 strani gostega teksta) – vendar ima velike zmogljivosti, sploh ob uporabi pomožnih programov, kot so MailMerge, SpellStar, Starindex in drugi. Po drugi strani pa so nekatere strani programa svojevrstni anahronizmi – WordStaru, na primer, manjka opcija »program customize«, ki je sicer razširjena – to za uporabnika, ki ne uporablja papir dolžine 11 palcev (pri nas v večini drugih evropskih držav uporabljamy 12-palčna) in ne želi imeti 66 vrstic na stran, pomeni vnašanje

»točkastih ukazov« na začetku vsakega novega besedila (z njimi določijo format strani); program je usmerjen k uporabi marjetičnih tiskalnikov; število kontrolnih kodov je omejeno itd. Res je, da WordStar pri instalaciji, pa tudi kasneje, lahko priredimo zahteve uporabnika, vendar ta postopek (patching) zahteva poseg poznavalca strojnega jezika – v nekaterih drugih urejevalnikih to tudi laiki z lahkoto opravijo, vse po tebe skozi enostavne menije.

Prav tu se pojavi potrebuje. Je preveč zapleten, da bi ga uporabljali za priročnostno pisanje pisem in kratkih besedil – če pa potrebujete urejevalnik z več možnostmi ali daljša besedila (preko 20/30 Km), če želite v svoje tekste vnašati datoteke ASCII, dobljene z drugimi programi CP/M (podatkovne baze, preglednice . . .), če pišete številna serijska pisma ali potrebujete kvaliteten urejevalnik za številne prevajalnike, ki delujejo pod CP/M, potem je WordStar prava izbira. Poleg tega je program v primerjavi z drugimi CP/M urejevalniki (dBASE II, Multiplan, Supercalc 2.0), glede hardvera zelo skromen in ga je mogoče uporabiti na sistemih z minimalno količino uporabniškega pomnilnika (potrebujete manj kot 40 km TPA), 40-stolpičnim ekranom in disketnimi enotami skromnih zmogljivosti (160–200 Km). Vendar pa je, kot pri vseh CP/M programih, hitrost take enote odločilna – to je verjetno razlog, zakaj WordStar ni postal popularen na C 64, opremljenih z modulom Z-80.

Osnovna prednost CP/M urejevalnikov besedil je ta, da omogoča delo s t. i. »virtualnim pomnilnikom« na 8-bitnih računalnikih oz. možnost obdelave tekstov, obširnejših od prostega pomnilnika. To bodo znali ceniti vsi tisti, ki so na Taswordu poskušali pisati tekste s 30–40 stranmi. Čeprav je teoretična dolžina datoteke anaka kapaciteti diskete, v praksi ni tako. Prvo omejitev pomeni sam način dela: WordStar rezervno kopijo (backup) briše šele potem, ko je posnel novo, torej so na disketi hkrati vsaj dve verziji urejevanega teksta. Razen tega WordStar pri izvajanju nekaterih ukazov (»poidi na konec«, »poidi na začetek«, »poidi« in druge) odpre posebne datoteke (začete, »\$\$\$«), ki požrejo nekaj prostora – to pa še ne pomeni, da potrebujemo prostor za trikratno dolžino urejevanega teksta, ker bo v takih primerih program sam izbrisal staro verzijo. V zvezi z izvajanjem teh ukazov je še ena omejitev: pri dolgih datotekah delo dolgo traja in vam ne preostane nič drugega, kot da si

dobite RAM – disk ali pa razdelite svoje knjižgo na poglavja, ki ne bodo daljša od 30–40 strani z dvojnimi presledkom (približno 60–80 km). V prid takemu delu je tudi dejstvo, da pomožni programi prav lepo delajo s tako razdrobljenimi datotekami. Torej: če gre za sistem z eno samo disketno enoto majhne kapacitete, bodo dolžine besedil omejene na polovico prostora, ki ostane po snemanju WordStara. Če imate močnejšo enoto ali celo dve, boste lahko izkoristili različne kombinacije, saj WordStar omogoča urejanje teksta na dveh disketah. Kaj se zgodi, če slučajno prekoračimo mogočevneje? Največkrat to pomeni, da je odtipkan tekst dokončno izgubljen, včasih pa računalnik celo zablakira!

Ko včitate WordStar, se na zaslonu najprej pojavi uvodni (nizilne) menu z osnovnimi ukazi (obzira delovnega diska, direktorij, izbira nivoja pomoči – t. i. števila menuev, ki naj bodo vidni med delom), ukazi za odpiranje, tiskanje, preimenovanje, kopiranje in brisanje datotek, sistemski ukazi (zacasen izstop iz WordStara zaradi izvirne podprograma ali dokončen prehod v operacijski sistem) in nalogi za opcije WordStara (MailMerge in SpellStar). Zanimivo je, da MicroProjev programer ni izdelal potrebno v pregrit, ki so mu tako pomembne dolžine datotek in količina prostega pomnilnika na disku, vdelati rutino, enakovredno STAT.COM – slednjo se sicer da poklicati, vendar le iz uvodnega, ne pa tudi iz delovnih menuev, kjer bi jo bolj potrebovali. Za silo pomagal tudi Ctrl-OP: v statusni liniji se bo dolžina teksta nasteno v straneh izpisala v bytih. Poleg uvodnega in glavnega urejevalnega menueja je mogoče med delom poklicati še pet »prefiksni« menuev: HELP (Ctrl-J), ki vsebuje kratek opis vseh ukazov in lahko do precejšnje mere zamenja priročnik – v njem lahko tudi določimo nivo pomoči; BLOK (Actr-K), iz katerega je moč manipulirati z datotekami na delovnih disketah; QUICK (Ctrl-D), ki vsebuje dodatne ukaze za premikanje kurzorja in ukaze tipa »poidi« – »zamenjaj«; PRINT (Ctrl-P) z ukazi za delo s tiskalnikom – poleg njega obstaja še posebej menu, ki ga dobimo z ukazom »IZPIŠI DATOTEKO« iz uvodnega menueja; FORMAT (Ctrl-O) z ukazi za formatiranje teksta, poravnavanje, word-wrap itd.).

Čeprav se tehnika vnašanja in urejevanja teksta z WordStarom ne razlikuje preveč od drugih podobnih programov, delo z diskom povzroča nekatere omejitve, občasno, in sicer odvisno od prostega RAM (na CPC 464/664 manj od

ene strani, na CPC 6128 in C 128 veliko več), se avtomatično vključijo disketna enota in takrat je treba prenehati s tipkanjem oz. vsaj upočasniti tempo. V originalni verziji ni predvideno delo s kurzorskimi tipkami (njihovo funkcijo opravljajo kombinacije Ctrl-A.S, D.F.E in X), vendar se takšni ali podobni ukazi lahko sestavijo s spreminjanjem sistemskih sledi diskete z WordStarom z uporabo programov SETUP, SETKEY, DEFKEY in podobnih. Opazno so tudi razlike v načinu, kako program »skrolira« tekst na ekranih računalnikov z malo prostega pomnilnika. Namesto da bi vse vrstice premikali hkrati, jih premika drugo za drugo po nekem svojestvenem vrstnem redu in tako ustvarja čudne učinke. Čeprav se na vrhu zasлона pojavlja statusna linija podatki o strani, vrsti in stolpcu, v katerem je kurzor, je škoda, da kurzor ne utripa kot pri nekaterih drugih prevajalnikih in ga je včasih težko najti, sploh po večkratni uporabi funkcij »poidi« (Ctrl-QF) in »zamenjaj« (Ctrl-QA). Avtorji WordStara so se ravnali po logiki »Kaj vidis, to dobiš« (what you see is what you get – WYSIWYG), vendar je treba to – kot pri vseh urejevalnikih in 9 (bitnih) računalnikih – jemati s pridržki, saj je ekran po daljšem urejanju prekrit z različnimi kontrolnimi kodmi; da je stvar še huja, niso vse prikazane inverzno. Resnici na ljubo je treba povedati, da obstajajo opcije »skrij« in »pokaži« (kontrolne kode), vendar je vse to daleč od tistega, kar omogočajo moderni urejevalniki na IBM PC, macintoshu in stariju ST.

Pri delu z daljšimi teksti manjka tudi ukaz »poidi na stran . . .«, kar do neke mere nadomesti možnost postavljanja markerjev, a veljajo le, dokler teksta ne spravimo na disketo, t. j. samo enkrat. Od verzije 3.0 dalje zna WordStar tekst vodoravno premikati in z njim, kot tudi proizvajalec, mogoče pisati tekste za mikrofime (32.000 stolpcev – običajna širina na A4 je 64); vendar nekateri ukazi ne delujejo pravilno po 240. stolpcu. Reformiranj odkrije eno od pomembnih lastnosti programa: po moči pri razstavljanju besed na zloge (hypo-help). Se važnejše je, da v primeru, če se pri naslednjem preurejanju beseda sept sestavi, črtica med zlogi ne bo izpisana – za razliko od tistih, ki so mi jih vnesli neodvisno od WordStara (hard hypen – trdi prelom). Veja je pa podariti, da so majhne napake pri preurejanju (Ctrl-B) dokaj pogoste, posebno kadar gre za kakšnega od naših jezikov (WordStar črtico za prelom praviloma vstavi v »nj-ij-lj«). Zelo uporabni so blokovi ukazi, s katerimi lah-

ko WordStar pošilja in sprejema najrazličnejše datoteke ASCII vključno s tistimi iz Tasworda, Cambasa, Supercalca itd., v kombinaciji s »točkastimi ukazi« MailMerge pa postane pravi procesor informacij, čigar uporaba presegajo običajno pisanje serijskih pisem.

Za razliko od drugih urejalnikov, kjer blokovni ukazi zajemajo le cele vrstice, WordStar omogoča delo s stolpci, kar pomeni veliko prednost pri delu s tabelami, včitanimi iz Multiplana ali Supercalca. Izbrani blok WordStar podariti tako, da ga prikaže inverzno na monitorjih, ki so tega zmorni (začuda to velja tudi za ceneje Amstradove monitorje GT-64/65, ne pa za nekatero veliko dražje monitorje druge izdelave). Lastniki računalnikov s majhnim pomnilnikom imajo težave z dolžinami blokov – na CPC 464 in 664 se dajo premeščati le bloki dolžine do 65–70 znakov (izpizovano v čitavanju datotek z diska ni omejeno).

Osnovni namen »točkastih ukazov« (tipa »PL«, »MT«, »MB.ipd.«) je formatiranje strani na ekranu in tiskalniku. Poleg običajnih opcij (dolžina strani, robovi, glava, rep, položaj paginacije ipd.), pa najdemo tudi nekaj nenavadnih, toda koristnih. Na primer, ukaz »pogojna stran« (CP n) omogoča prelom strani na sredi tabele ali neposredno po kakem mednaslovu: če pred tabelo 10 vrstič postaviš (CP 10), se bodo prelomi vršili pred tem ukazom ali na 10 vrstic za njim. Z ukazom za paginacijo (PN) lahko dosežemo, da se vsi zipsi na tiskalniku začne s poljubno številko – ne pa tudi na zaslonu, kjer se vsaka tekstna datoteka začne s številko 1, kar je določena pomanjkljivost.

Kot že rečeno, je ena od najslabših strani programa komunikacija s tiskalnikom, posebno z modernimi matricnimi tiskalniki, ki praviloma nudijo veliko več možnosti, kot jih WordStar zna izkoristiti. Poskusite npr. tekst, napisan brez presledkov med vrsticami, natisniti z dvojnimi presledki. Če imate marjetični tiskalnik (daisy writer), boste izvedli iz kazalov »LH«. Ta pa ne deluje na večini matricnih, vsaj na Epsonovih in kompatibilnih tiskalnikih. Če poskusite dvojne presledke vnesti s formatiranjem (ukaz OS), boste videli, da se urejanje v takih primerih ne obnese in verjetno boste odnehali. Ostane vam le še to, da potrebne kode pošiljate tiskalniku pred delo z WordStarom, kar je danes, kod od poslovnih programov pričakuje prijazenost (user friendliness), precej velika pomanjkljivost. Komunikacija s tiskalnikom vsebuje tudi druge nelagodnosti (poskusite ukaz formfeed), ki zaokrožajo relativno neugodno silo in to aspektu programa.

Na računalnikih z minimalnim

pomnilnikom se bodo pojavile še druge težave: po vsakih 10–15 izpisanih vrsticah se bo WordStar ustavi, da prebere tekst z diska, kar precej podaljšuje čas tiskanja. Prav tako ne bo dostopna sicer zelo atraktivna opcija, ki omogoča tiskanje enega besedila in istočasno urejanje drugega. Na prvi pogled se to ne zdi kritično za poliprofionalno uporabo zunaj strojepisnih uradov. Vse pa je drugače, če želite izpisati 150–200 strani teksta v načinu NLO – takšna operacija lahko traja več ur in ves ta čas računalnika ne boste mogli uporabljati za nič drugega.

Čeprav je program zelo dobro dokumentiran (dokumentacijo sestavljajo trije zajetni priročniki: »WordStar – General Information Manual«, »WordStar Reference Manual« in »WordStar Installations Manual« – vse tri dobite pri nas preko malih oglasov), je o njem napisanih mnogo knjig in priročnikov. Znana nemška založba Markt & Technik Buchverlag (Hans-Pinsel Strasse 2, Haar bei München) ima v svojem katalogu petnajst knjig, posvečenih različnim področjem uporabe WordStara (med drugim tudi posebne priročnike za uporabo tega programa na C 128, računalnikih amstradschneider, apple, atarijih ST in IBM PC).

## WordStarove opcije in kloni

Programski paket za urejanje besedil, čigar osnova je WordStar, dopolnjuje programi MailMerge, SpellStar in StarIndex, proizvođa ista firma (vse štiri prodajajo pod imenom »WordStar Professional« za okoli 400 funtov), kot tudi številni programi nedovish proizvajalcev.

Prvi med pomožnimi programi, kot pove že ime, je namenjen pisanju serijskih personaliziranih pisem. Torej so lahko pozivi za sestanke ali prospekte namesto v stilu »Dragi tovariš, draga tovarica« oz. »Dragi/a tovariš/ica« naslovljeni vsakemu naslovniku posebej, če pa je treba, se lahko v tekst vrinejo tudi sporočila samo nekaterim ljudem z vnapišer določenege spiska (seznama). MailMerge omogoča tudi avtomatski izpis več kopij istega teksta, izpis več datotek z ene ali več disket in druge uporabne opcije. Če imate npr. na nekem disku več datotek in vsaka vsebuje po eno poglavje knjige, ki bi jo radi natisnili, boste z MailMerge sestavili ukazno datoteko naslednje vsebine:

- FI POGLAJVE 1
- FI POGLAJVE 2
- FI POGLAJVE 3

Ali, pri čemer so različna poglavja lahko na različnih disketih. Če gre za diskeje majhne kapacitete, ki ne morejo zajeti vseh datotek, lahko napišemo naslednji ukaz:

● FI B. POGLAJVE KCHANGE itd., zaradi katere bo WordStar odtiskal željeno poglavje in potem počakal, da operater vstavi novo disketo. Ta operacija je možna le na sistemih z dvema disketnima enotama, saj morajo WS.COM, WSMSSG.OVR in WSOVLY.OVR vse čas biti v A..

Drugi program – SpellStar – spada med programe, ki preverjajo pravilnost vnešenega teksta (spelling checkers), tako da besede z datoteke primerjajo s tistimi iz lastnega slovarja. Čeprav gre za kvaliteten program s slovarjem približno 20.000 besed, ki se jih da zlahka menjati in tako uporabljati tudi zunaj angleškega jezikovnega področja, pri nas ni posebno popularen zaradi specifičnosti slovnice jugoslovanskih jezikov.

Tretji program, StarIndex, je namenjen izdelavi različnih pojmovnih, predmetnih in drugih indeksov, ki so lahko urejeni po abecedi in imajo lahko več nivojev. Program je namenjen predvsem avtorjem strokovnih knjig in učbenikov, kjer so indeksi zelo koristni. Po trditvah proizvajalcev lahko program dela tudi s teksti, razdrobljenimi v več datotek, z vdelano opcijo »Style« pa je možno definirati potrebne formate za izpis dobjenih rezultatov Program bajne ne zahteva dodatnega pomnilnika.

Vsekakor lahko ob WordStaru uporabljamo tudi druge Micro-Projeve programe: DataStar, InfoStar, SuperStar, CorrectStar, StarBurst in CalcStar.

Popularnost in visoka cena WordStara (v Angliji okoli 300 funtov, tj. približno 160.000 din po tečaju maja 1986) sta bili izvis številnim softverskim hišam, ki so lansirale bolj ali manj uspešne kopije po znatno nižjih cenah. Skupina programerjev, ki je sodelovala pri izdelavi WordStara, je pred kratkim osnovala firmo, katere glavni produkt je urejalnik New Word – ta je, z manjšimi spremembami, na las podoben originalu: uporablja iste ukaze in dela z vsemi pomožnimi programi in datotekami, stane pa le 69 funtov (CP/M+ verzija za CPC6128/8256). Ima »mail merge« rutino, omogoča zaščito teksta pred spremembami, ima ukaz »poidi na stran...«, kurzor ostane kjer je bil po ukaz »posneni in se vrni« (Ctrl-KS, ni potrebno tipkati Ctrl-QP kot pri WS), dajo se definirati funkcije tipke, uporabljaj lahko 90x32 znakov na PCW 8256 in podobnih, manjka pa WordStarova možnost izvajanja drugih programov iz programa samega in hkratno tiskanje urejanje. V ceno programa na 3-palčnih disketah, insriranega za Amstradove CP/M računalnike, je vračunan spell checker WordPlus firme Oasis Systems s slovarjem 25.000 besed, ki velja za enega od najboljših programov te vrste. Ista firma je pred kratkim lansirala

New Word 3, urejalnik za 16-bitne računalnike v različnih za PC DOS, MS-DOS, CP/M 86 in TURBO DOS; ta ima vdelan slovar, podprogram za indeksiranje, matematične funkcije in možnost dela v barvah, stane pa 249 funtov, kar velja za konkurenčno v kategoriji IBM PC. Med proizvajalci ceneneh kopij WS se je vrnil tudi sam MicroPro s Pocket WordStarom, skromnejšo verzijo, ki jo prodajajo po trikrat manjši ceni.

## Kako do YU – WordStara?

Uporaba WS pri nas odpira vrsto problemov, katerih reševanja pa ne moremo napreth MicroProjevim programerjem, a jih je treba rešiti, če želimo program uporabljati za pisanje v kakem izmed jugoslovanskih jezikov.

Domáci hekerji in črna borza so, kaže, bili uspešnejši od naših proizvajalcev CP/M računalnikov. Najprej se je pojavila nemška verzija, potem pa prvi amaterski in kasneje vse uspešnejši prevodi, narispravljeni s 1.1. disk – sektor – editorji, s katerimi zaobidemo vse obstoječe zahtebe in vse prevedemo kar na disku. Če imate priročnik za instaliranje WS, lahko na ta način vnesete tudi kode za nadzor tiskalnika.

Naslednji korak je bila vdelava YU črk, kar ni predstavljalo posebnega problema za avtorje in programirke jezikov in programskih orodij, čeprav lahko nekateri take rutine tipa YUSCI.COM povzročijo nepravilno delovanje programa. Ena od neškodljivih posledic je to, da se v vse meniji namesto puščice nazvorg (ASCII 94), ki označuje tipko CTRL, pojavja črka »C«. Če vas to moti, pomislite na dejstvo, da so meniji na nekaterih domačih CP/M računalnikih prekriti še s črko »D« namesto znaka »I«, ki se v CP/M programih pogosto uporablja za navpične črte. Tako preurejen WordStar je verjetno najboljši urejalnik besedil, ki ga pri nas srečamo na hišnih računalnikih, a ne za dolgo. Z valom 16/32-bitnih mikroručunalnikov so se standardi spremenili: v modi so okna, ikone, miši, delo z več datotekami in grafiko, polnim izkoristanjem možnosti modernih tiskalnikov, trdnih diskov in podobnega; odpadne so tudi težave pri delu s CP/M 2.2 in CP/M+. To pa ne pomeni, da bo WordStar kmalu pozabljen – njegova široka razširjenost, nedovna kvaliteta in spremljajoča programska oprema, ki je odgovor na mnoge zahteve profesionalne uporabe, bodo poskrbeli, da ta urejalnik besedil še dolgo ostane industrijski standard na svojem področju.

# Besedilo in slike: končno skupaj

ZIGA TURK

**V** večino knjig, skript, diplomskih nalog, člankov za časopis, poročil, priložnic, ... ne sestavljajo samo besedila, ampak tudi večje ali manjše število slik, skic ali diagramov. Vedno več ljudi pripravljajo besedila z mikroročunalnikom in jih natisne na domačem tiskalniku, tiskarji pa jih le še preslikajo, pomanjšajo in razmnožijo. Vključevanje slik v takšna besedila je posele, ki zahteva nekaj rokodelske spretnosti s škjarjami in lepilom in dober fotokopirni stroj, da se na kopiji ne pozna, kje smo lepili. V zadnjem času so tudi za dostopnejše mikroračunalnike razvili urejevalnike, ki dovoljujejo vključevanje z računalnikom narisanih slik in tako omogočajo, da so naši izdelki lepši in jih je enostavneje natisniti. Predstavljamo dva izdelka za Atari ST Ker tečeta pod operacijskim sistemom GEM, ju bo verjetno kmalu mogoče kupiti tudi za računalnike tipa PC (predvsem amstrad in philips PC, kjer je GEM, že vključen v ceno)

Tako je besedilo, ko smo ga izpisali s tiskalnikom.

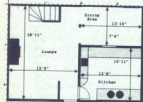
Boffin - Comb and Shells, see page 104.

General Camera Apple

Apple Computer, Inc.  
We are pleased to forward to you details of the Video Capture Board, which is now available in the USA. Please contact us for more information.



Printing in monochrome or color. This software prints out pictures, text and graphics. It is designed to be used with the Atari ST and the Video Capture Board. The software is designed to be used with the Atari ST and the Video Capture Board. The software is designed to be used with the Atari ST and the Video Capture Board.



We have tried this software with the Atari ST and the Video Capture Board. We have tried this software with the Atari ST and the Video Capture Board.

## 1st Word +: prva beseda, drugi poskus

Urejevalnik je našim bralcem že znan (Moj mikro 4/1986, str. 18). Od takrat do danes so uporabniki atarija ST preizkusili še nekaj verzij, kjer je bila marsikatero pomankljivost začetne različice že odpravljena. Tale tekst nastaja s 1st Word Plus, verzija 1.24. Splošno ime programa je precej podoben sistemu, ki je bil opisan v omenjenem članku, in tam zapisanega ne bom ponavljal. Program je po novem dolg približno 180 KB in se v strojih s 520 KB brez ROM verjetno težko uporablja. Od urejevalnika besedil pričakujemo:

1. vnosi in popravljanje besedila
2. oblikovanje natisnjenega teksta
3. izpis z različnimi vrstami tiskalnikov
4. shranjevanje in branje teksta iz zunanjih pomnilniških naprav.

## Vnos in popravljanje

Spremembe je malo in niso bistvene. Odstranili so vmesni pomnilnik med programom in tipkovnico, ne dogaja se več, da bi se zaradi nestrpnega tlačenja kazalcev za pomik premaknili dlje, kot ste si želeli. Ob začetku dela se nova verzija vedno postavi v način WP. To je bilo prej prepuščeno naključju. Namesto ravnika je v prvi vrstici okna mogoče pokazati tudi pozicijo v besedilu (stran, vrstico, stolpec). Še natančnejše informacije o dolžini besedila in zasedenosti diska daje posebna opcija v meniju, ki pove, koliko je našega besedila (strani, vrstice, besede, bytov), koliko pomnilnika je še prostega v RAM in koliko na disku.

Pomikanje po besedilu je hitrejše. Iz enega na čisto drugo konec se z navigacijskim stolpcem (scrollbar) premaknemo tako rekoč v hipu. Nova je tudi možnost za skok na natanko določeno stran (ali vrstico, če nismo v načinu WP). Polepnejše je tudi premikanje pikov besedila. Oznake za način pisanja (mastno, podčrtano ...) se pri tem ne izgubljajo več. Iskanje je nespremenjeno, zato še vedno pogrešam možnost, da bi iskal kontrolne znake. Iskanje je zadaljovito hitro.

## Oblikovanje besedila

Atari ST daje avtorjem urejevalnikov besedil možnost, da so teksti že na zaslonu zelo podobni izdelku na papirju. 1st Word je znal to že v prvi različici dobro izkoristiti. V plusu je mogoče izbirati med normalnim in dvojnim razmikom med vrsticami. Ker se le redko zgodi, da bi nekateri deli besedila pisali razmaknjeno, druge pa z običajno gostoto, se mi zdi boljše, da bi razmaknjeno besedilo vključili kot opcijo pri tiskanju. Koncepte pač tiskamo bolj razmaknjeno, da imamo kam pisati popravke, končni izpis pa je lahko izpisan bolj skupaj.

Ob vnosu se besedilo avtomatsko poravnava, kot želimo. Če ga pozneje popravljamo, ga je treba preurediti s posebnim ukazom. Ta je zdaj zboljšal in omogoča preurejanje odstavka, dela besedila ali vsega teksta. Konec odstavka je še vedno definiran napačno. Zadnji znak v odstavku ne sme biti presledek. Pri poravnavanju zdaj urejevalnik svetuje, kje naj bi se beseda delila. Program je sicer pisan za angleški pravopis, zanimivo pa je, da v kakih 90 odstotkih primerov deli pravilno tudi za slovenska besedila. Po novem je mogoče za isto besedilo izbrati več ravnin, t.j. različnih širin besedila, in različnih znakov (pica, elite, condensed). Žal pa ni mogoče izbrati več različnih glav in podpisov strani. Še vedno smo tudi omejeni na eno samo vrstico glave in eno samo vrstico podpisa. Nova različica naj bi znala vključevati pripomočke pod črto, a vse še ne dela čisto tako, kot so si zamislili.

## Izpis s tiskalnikom

Program za izpis je bil že pri prvi verziji zelo prožen: enostavno se je dal prilagoditi praktično vsakemu tiskalniku. Po novem je vključen v osnovni program, tiskanje pa lahko teče v ozadju, medtem ko pišemo kakšno drugo besedilo. Zaradi tiskanja slik je dodanih nekaj novih ubežnih kečev. Če uporabimo Epsonov način 4 (CRT grafika - 1B, 2A, 04, ...), se bodo slike na papirju natanko ujemale s tistimi na zaslonu. To je pomembno takrat, ko s programom za risanje nekaj narišemo, z urejevalnikom pa v sliko vpisujemo podatke, oznake, kote, komentarje... Kontrolnih programov, s katerimi tiskamo, je lahko več (in ne samo 1stPRINT.DOT kot doslej).

## Shranjevanje na disk

Nova je možnost »save and resume«, torej da shranimo besedilo na disk in urejamo naprej. Še vedno pa program nima ukaza

»shrani vse in odnehaj«, ki bi prišel še posebej prav, kadar urejamo več besedil hkrati in bi radi na hitro zapustili program. Pisanje na disk in včitavanje besedil nista nič hitrejši. Škoda.

## Grafika

Z urejevalnikom ne moremo risati slik, ampak jih moramo v posebnem formatu pripraviti na disku in jih potem vključiti v besedilo. Žal 1stword ne shranjuje besedila skupaj z vključenimi slikami, ampak na določena mesta v datoteki zapiše ime datoteke, v kateri je zapisana ustreza slika. Pri tiskanju in ponovnem urejanju besedila mora biti slika na dostopnem področju (mapi) na disku.

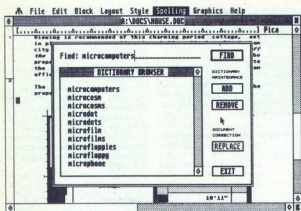
Če besedilo urejamo v grafičnem načinu, se črke znizajo, tako da je na zaslonu 27 vrstic, slike in besedilo pa so po višini in širini proporcionalno temu, kar bo zapisano na tiskalnik (pod pogojem, da je razmik med vrsticami 1/8 palca). Slike, ki jih želimo vključiti, morajo biti zapisane v formatu, ki ga daje program SNAPSHOT.ACC. Tega instaliramo v meni Desk in potem lahko iz vsakega programa, ki dela pod GEMom, vozimo slike ali dele slik v urejevalnik. Tu jih lahko le še premakemo. Večati, manjšati ali kako drugače spreminjati pa jih ne smemo

## BOFFIN: posrečena kombi-nacija

Firma SOFTWARE PUNCH je na sejmu PCW predenila z dvema izdelkoma. Na tem mestu predstavljamo urejevalnik besedil BOFFIN, njihova lokalna vrsta pa bo, upamo, prišla na trg kdaj pozneje. Boffin je krajši kot Prva beseda, poleg zmogljivega urejevalnika besedil pa vključuje program za risanje. Udobje je nekaj manj in program daje nasploh bolj robusten in neotesan videz, ima samo eno veliko pomankljivost: odprto je lahko samo eno okno s tekstom.

### Urejanje teksta

To je podobno, kot pri drugih urejevalnikih in ne prinaša presenečenj. Po besedilu se lahko pomikamo z miško sama s kontrolnimi tipkami. Kdor je navajen Wordstarja, bo cenil, da so kombinacije CTRL enake kot pri tem znanem urejevalniku. Program



V urejevalnik je vdelan korektor pravopisa z dokaj bogato zbirko besed, ki jo lahko še urejamo. Na sliki je del programovega slovarčeka.

več. Besedilo lahko pišemo čez slike. Pri tiskanju se bodo take vrstice tiskale v dveh prehodih tipalne glave, najprej grafična informacija in potem znakovna.

Novi urejevalnik ima vdelan tudi korektor pravopisa (spelling checker) s slovarjem blizu 90.000 angleških besed. Slovar omogoča dodajanje in odzemanje besed.

dela po načelu »kot na zaslonu, tako na papirju«. Kdor pa bi se rad zares prepričal, kateri »nevidni« znaki so tudi vključeni v tekst, si jih s posebno opcijo lahko kar naprej ogleduje. Na zaslonu je mogoče prikazati tudi masten, podčrtan ali poševen tip črk, indekse in potence pa si moramo oblikovati sami. Dva tipa znakov lahko določimo sami. Tip črk določamo šele potem, ko smo jih že napisali, in ne sproti.

Oblikovanje strani je po eni strani slabše, po drugi boljše kot pri 1stWordu. Boljše zato, ker lahko definiramo kar po 7 vrstic podpisov ali glav, ki se bodo izpisovali na vrhu in na koncu strani. Žal ni mogoče dojeti, da bi se npr. številna strani na neparnih straneh izpisovali na desni, na parni pa na levi strani. Izbirati je mogoče med formatoma A4 in A5 in etiketami, nastavljamo število vrstic na strani in razmik med vrsticami.

Tiskanje ni urejeno tako elegantno kot pri Prvi besedi. Nastavljamo lahko samo nekaj ubeznih sekvenc, predvsem pa ni mogoče prirrediti poljubne sekvence poljubnemu znaku. Znake drugih abeced avtomatsko tiska kot grafiko. Tudi pri tem programu se da tiskati v ozadju.

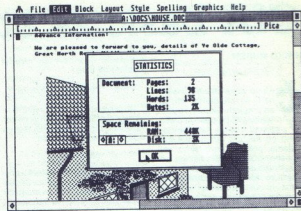
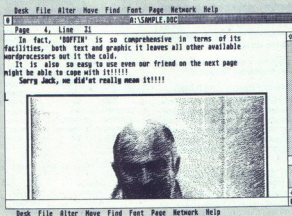
## Sklep

GST je iz že tako dobrega izdelka naredil boljšega. Odpravil je vse napake, ki jih je Jure Skvarč naštel v naši aprilski številki. Kljub nekaterim novim programom ostaja 1stWord urejevalnik št. 1. Komunikacija z uporabnikom je zgledna. Navodila so potrebna le za nastavitve tiskalnika, drugače se s programom dobro znajde tudi popoln začetnik, ki zna vsaj malo angleško. Program omogoča pisanje več besedil hkrati.

## Grafika

Z Boffinu lahko slike tudi rišemo. Nastajajo v posebnem oknu. Če želimo sliko vključiti v besedilo, jo moramo najprej posneti na disk in jo potem po potrebi kot pri Prvi besedi vključiti. Tudi tukaj se besedila shranjujejo posebej in slike posebej. V tekstovnem oknu se nam slike pokazuje, rišemo pa se čudno in okorno, po stopnjah. Teksta v grafične vrstice ni mogoče zapisati.

Grafični urejevalnik je tipa »point-«, torej bitno orientiran.



Takle je zaslon med urejanjem besedila. Na sredini je informacija o zasedenosti pomnilnika.

Uporabljamo lahko znake iz vseh mogočih zahodnoevropskih abeced, s programom YUTILITY.ACC pa tudi z YU znaki.

Nekaj pomanjklivosti ostaja. Kdor se namerava lotiti pisanja knjige ali skript, bo zahteval prožnejše podpise in glave, spreminjanje tipa črk, števila vrstic, vsebine... Pomagal si bo pač tako, da bo knjigo razbil na nekaj datotek. Škoda je tudi, da nekaterih pog-

sto uporabljenih funkcij, ki so našteje v meniji, ni mogoče pognati s kombinacijami kontrolnih tipk (iskanje, indeksi, potence). Morda pa bodo to popravili, ko bodo dodali kakšen nov plus.

gama. V program za risanje lahko nalozimo slike, narisane s programoma Degas ter meladatekote, kot jih definira GEM. Tudi shranjevanji jih je mogoče v vsakem od zgornjih formatov. Če jih želimo vključiti v besedilo, pa morajo biti v posebnem, lastnem formatu. Slik, kot jih posname Snapshot, program ne zna brati.

## Sklep

Boffin je posrečena kombinacija programa za risanje in urejevalnika besedil. Program je v mnogocem boljši od 1stWord žal pa ni tako skrbno dodelan in oblikovan, kot izdelek firme GST. Na ekranu pogosto ostajajo kakšne packe, odvisno iz izbrano točko menija je včasih presenetljiv, a to marsikomu ne pomeni veliko. Če vaše delo zahteva sprotno risanje (npr. formul) in se vam upira, da bi za vsako skico zapiščali program, potem boste z Boffinom zadovoljni. Pa še nekaj nas je pri odnosu Software Punch prijetno presenetilo. Če bi radi, da so meniji in ostala ropotija v vašem materinem jeziku, jim pošljite prevod in brezplačno vam bodo poslali prevedeno verzijo programa. Tudi o manjših popravkih se lahko pogovorite z njimi. Tole zadnje seveda velja, če ste program kupili, kar pa, priznajte, ni vaša navada.

# GfA basic za atari ST

HINKO MUREN

Če smo vajeni delati v basicu na C 64, sharpu ali PC, se ob pregledu ukazov, napotkov in funkcij, ki jih obsega basic ST v sedanjih izvedbi, kakršno dobimo ob nakupu računalniškega sistema, prvi hip navdušimo, saj ta nabor preseneča celo tistega pri PC. Morda bomo pogrešali le oba ukaza za izpis in vnos časa in datuma, ki sta kar precej koristna. Prvo navdušenje pa nas bo najbrž kaj kmalu minilo, morda že takoj na začetku ob nenavadnem načinu dela s štirimi okni, zaradi katerih bomo brez prave potrebe izubuglji več časa, kakor ga bomo pridobili. Po vrh vsega v editorju ne deluje avtomatska numeracija pri neposrednem načinu v tako imenovanem komandnem oknu pa ni mogoče uporabiti kurzorskih tipk in je največkrat najbolje pri napaki ponovno napisati kar celo vrstico. Snovalci basica ST so morda imeli z okni dober namen, vendar je v njih šla po zlu glavna prednost basica – preprosto delo. Če smo začetnik ali novinar in zapremo vsa okna, pa nas čaka novo presenečenje – vse dotedanje delo bo šlo po zlu, ker bo za nadaljevanje treba računalnik resetirati ali ugasniti. Zelo malo tudi, zlasti pri delu z grafiko, da »hardcopy prenese na papir vedno tudi zgorjelo letvico z seznamom menijev.

Podobnih »presenečenj« vseh čaka še cela vrsta. Za začetek poskusimo računati z dvojno natančnostjo. Poskusimo z naslednjim preprostim računom:

```
PRINT 1234567899#++
```

Pričakovali bi, da bo rezultat pravega vsaj lju do štirinajstega mesta. Ob dobljenem rezultatu pa utegne občutljivo programerja z visokim tlakom zadeti kap!

Tudi pri izpisu tekstov ne bo šlo brez zve. Poskusimo v tekstu uporabiti znak za paragraf ali, če imamo nemško tipkovnico, črke s preglesi. Pri pisanju programa ali ukaza se bodo ti znaki lepo izpisali na zaslonu, v izvedbenem oknu pa se bodo iz neznanega vzroka izgubili.

Vsaj v začetku motijo tudi nekatere posebnosti, ki jih v drugih dialektih basica ni najti. Poskusimo z naslednjo vrstico:

```
100 INPUT A,B,C,
```

Pri izvedbi programa je treba vrednotiti za vse tri spremenljivke vnesti drugo za drugo, ločene z vejico, in čele na koncu pritisniti na tipko RETURN.

Povsem nepričakovan rezultat dobimo na ukaz:

```
PRINT ""Tekst v narekovajih""
```

Pričakovali bi, da bo prva dva in zadnja dva narekovanja in zadnje razumel kot dva prazna znakovna zve, v resnici pa dobimo:

```
"Tekst v narekovajih"
```

Podobnega bi se našlo še precej, vendar dovolj o basicu ST. Od julija letošnjega leta je mogoče kupiti zelo zanimivo verzijo basica za ATARI ST z imenom GfA basic, ki utegne biti zelo zanimiva tudi za zahtevnejše programerje. Dokončne sodbe brez daljših izkušenj o tem interpretatorju še ni mogoče podati, saj nedvomno ni brez napak, vendar so prvi poskusni izpisi zelo obetavni, tudi dokumentacija je zadovoljiva, seveda pa ni namenjena popolnemu začetniku.

## Zunanji vtis

GfA basic dela z dvema »oknom«, ki nista izdelana z GEM. V prvem oknu, ki se odpre takoj po vnosu interpretatorja, imamo na voljo dokaj dober editor, ki dovoljuje delo z vsemi kurzorskimi tipkami ali z miško. Zgornji vrstici vsebujeta menu z najbolj pogosto uporabljanimi ukazi, ki jih lahko »poklikamo« z miško, vrsto editorskih ukazov pa dobimo s kombinacijami kurzorskih in funkcijskih tipk. Teh ukazov je toliko, da si jih je kar težko zapomniti. V drugem oknu lahko dajemo ukaze v neposrednem načinu, v njem dobimo tudi izpise oziroma rezultate programov. Zanimivo je, da lahko napišemo nov ukaz brez brisanja tudi, če je v tem oknu od poprej ostala kaka grafična slika. Po brisanju tega okna ne ostane v zgornjih vrsticah nobena moteca letvica z meniju. Programa v tem oknu namreč ne pozna števil na začetku vrstic!

Če smo v oknu za neposredno delo, napišemo za začetek:

```
PRINT FRE(O) <RETURN>
```

Nad rezultatom bomo navdušeni, saj interpretor zavzame samo tretjino prostora v primerjavi z basicom ST. Ta ukaz izvede obenem tudi tako imenovani »garbage collection« in naredi red v pomnilniku.

Pri številskih operacijah dela GfA basic vedno z natančnostjo na enajst mest, ne pozna pa števil z dvojno natančnostjo. Sicer pa so na voljo vsi tipi spremenljivk, ki jih poznajo »normalni« basic.

Preseneča hitrost dela, saj je skoraj takšna kot pri prevajalniku. Preskus s testi benchmark, ki jih je objavil Moj mikro v lanski avgustovski številki, so dali naslednje rezultate:

Test benchmark	Porabljeni čas
1	0,1
2	0,4
3	1,2
4	1,0
5	1,2
6	1,8
7	2,8
8	30,0
	[s]

Razen pri osmem testu so to od deset do dvajsetkrat krajši časi kakor pri vseh drugih, tudi dražjih

računalnikih, za katere je bil takrat narejen primerjalni test. Povedati pa je treba, da primerjava za osmi test ni objektivna, kar funkcije drugi računalniki računajo na sedem mest, ATARI ST z GfA basicom pa na enajst! Za meritve je bilo celo treba v objavljenem programu vse zanke podesteteriti, drugače bi v prvih petih testih dobili rezultati 0.

## Editor

Editor ima na voljo 47 ukazov, če sem štejejo tudi delo s kurzorskimi in funkcijskimi tipkami. Posebnost interpretatorja je, da je v eno vrstico dovoljeno napisati samo en ukaz, kar bo povzročilo nekaj težav zlasti tistim, ki so vajeni kilometrskih vrstic. Dveh uk-

Tabela 1. Ukazi in funkcije za normalno delo

ABS	DFREEE	INPUT	DN...	GOSUB	RMDIR
ADD	DIM	INPUT #	DN BREAK		RND
ALERT	DIM?	INPUT#	DN ERROR		RSET
ARSPR/FILL	DIR	INSTR	DN ERROR GOSUB		RUN
ASC	DIV	FILL	DN MENU GOSUB		SAVE
ATN	DO...LOOP	LEFTS	OPEN		PSAVE
BIN#	DRAW	LEN	QUIT		SETCOLOR
BLOAD	EDIT	LET	PAUSE		SETTIME
BSAVE	ELLIPSE	LINE	P(R)BOX		SGN
BOX	END	LINE INPUT #	P(CIRCLE		SIN
CI	EOF	LINE INPUT #	P(ELLIPSE		SOUND
CALL	ERASE	LIST	PEEK		SPACE#
CHAIN	ERR	LLIST	DPEEK		SPEEK
CHDIR	ERROR	LOAD	LPEEK		SPOKE
CHDRIVE	EXIST	LOC	PI		SPLDPE
CHR#	EXIT	LOCAL	POINT		SPLDPE
CIRCLE	EXP	LOF	PLD		SPRITE
CLEAR	FATAL	LOG	POKE		SQR
CLEAR#	FIELD	LOG10	DPOKE		STOP
CLOSE	FILES	LPOS	LPOKE		STR#
CLOSE#	FILESELECT	LPRINT	LSLET		STRINGS
CLS	FILL	LIST	POLYLINE		SUB
COLOR	FIX	MAX	POLYMARK		SWAP
CONT	FOR...NEXT	MENU	POS		SYSTEM
COS	FORM INPUT	MID#	PRINT		TAB
CVI	FRACT	MIN	PRINT #		TAN
CVL	FRE	MKDIR	PRINT ( #) USING		TEXT
CVS	FULLW	MK#	PROCEDURE		TIMES
CVF	GET	MK#	PUT		TIMER
CVD	GET (RL,DT)	MK#	PUT (RL,DT)		TITLEW
DATA	GOSUB	MK#	POLYMARK		UPPER
DATES	GOTO	MKD#	RANDOM		VAL
DEC	GRAPHMODE	MOUSE	READ		VAL#
DEFILL	HARCOPY	MOUSE#	RELEASK		VAL?
DEFBN	HEX#	MOUSEY	REM		VARPTR
DEFINE	IF	MOUSEX	REPEAT...UNTIL		WAYE
DEFLIST	INC	MUL	RESTORE		WHILE...WEND
DEFMARK	INFOW	NAME	RESUME		WRITE (#)
DEFMOUSE	INKEY#	NEW	RETURN		
DEFTXT	INP	OC#	RIGHT#		

Tabela 2. Ukazi in funkcije za delo z operacijskim sistemom

ADDRIN	GB	GINTIN	FTSIN
ADDROUT	GCONTROL	GINTOUT	PTSOUT
BIOS	GENDOS	INTIN	VBISYS
CONTROL	GENSYS	INTOUT	XDOS

zov ločiti z dvojičjem ni dovoljeno. Pisati smemo z velikimi ali malimi črkami, po vnosu vrstice <RETURN> bodo vsi ukazi napisani z malimi črkami in veliko začetnico, vse spremenljivke pa z velikimi črkami. Po želji pa lahko dobimo izpise ukazov z velikimi črkami, zato bodo pa spremenljivke pisane z malimi črkami. Presledek je treba pisati na koncu vsake rezervirane besede (vendar ne vedno), če ni v nadaljevanju narekavaj ali oklepaj. Pri sprejemu vrstice vrine interpreter sam potrebne presledke in spremeni črke, pri strukturah pa vrstice tudi sam umakne ustrezno od levega roba, tako da izpis programa spominja na lepo napisan program v pascalu. Programerju na to ni treba nič paziti.

Pri vnosu vsake vrstice opravi interpreter kontrolo sintaktične pravilnosti in morebitno napako javi v vrstici tik pod letvico z menuji. Dokler napake ne popravi, je vsako nadaljnje delo nemogoče. Začetnik, ki napake ne najde, bo kar v zadregi, saj mu bo celo miška odpovedala pokročino.

Zelo prijetno je, da skoraj vsak ukaz lahko napišemo skrajšano (brez pike), pri tem pa je vseeno, če napišemo tudi kako črko več, kot je nujno potrebno. Po pritisku na tipko <RETURN> se bo ukaz sam podaljšal na pravilno dolžino. Seveda se je treba tudi tega navaditi, sicer se utegne pripetiti, da na primer po pomoti v neposrednem načinu vpisemo <RETURN>, nakar računalo odpove, če nimamo slučajno vključenekega pisalnika. Za skrivnostni ukaz je namreč okrajšava za HARDCOPY! Možnosti, ki jih ponuja GFA basic, lahko najhitreje ocenimo, če si ogledamo nabor ukazov in funkcij, ki so zbrani v tabelah 1 in 2.

čunalnikih PC), je tu precej novega. Nekaj ukazov in funkcij je takih, ki bi jih z lahko pogrešali, vendar prihranijo precej časa pri programiranju ali pri obdelavi programov. Najprej bi lahko omenili ukaze ADD, SUB, MUL IN DIV, ki pristejajo, odštejajo k navedeni oziroma spremenljivko z nameri množijo ali delijo, dodatno pa sta predvidena tudi ukazi INC in DEC, ki pristejajo oziroma odštejajo vrednost 1. Vsi ti na videz morda nepotrebni ukazi prihranijo pri večjem številu ponavljanj kar precej časa. Pisanje programa precej skrajša na primer ukaz RANDOM, ki daje celostevilno naključno število, za razliko od znanega RND, ki generira decimalna števila z vrednostmi med 0 in 1. Kdor rad dela s POKI in PEKI, bo zelo vesel dodatnih ukazov DPOKE, LPOKE, DPEEK in LPEEK, ki prinašajo 2 oziroma 4 byte. Dodatno so na voljo še ukazi SPOKE, SDPOKE in SLPOKE, s katerimi je mogoče posesti tudi v zaščiteni področja pomnilnika. Nadalje je na voljo cela vrsta ukazov za delo z disketami. Ukazi DIR, FILES, DIRS, CHDIR, MKDIR, RMDIR, CHDIR, DFRFE, EXIST, FILESELECT, KILL in NAME omogočajo dva različna načina izpisa direktorija, izpis, zamenjavo, odpiranje in zapiranje poddirektorijev (map), izpis prostega prostora na disketi, preverjanje, ali je datoteka na disketi, prikaz okvira za izbrano datoteko, brisanje in preimenovanje datotek. Manjša torej samo možnost preimenovanja diskete, ki pa očitvidno ni predvidena v operacijskem sistemu atarija.

Zanimivo je, da ima GFA basic vrsto ukazov, ki omogočajo uporabo rutin operacijskega sistema tudi programerjem, ki ne obvladajo skokov v sistem. Z ukazom ALERT lahko programiramo svarilni okvir enake oblike, kakor ga sicer poznamo iz raznih rutin operacijskega sistema (na primer opozorilo, da bodo izbrisani vsi podatki pri formatiranju), le da tekst v njem lahko določimo sami.

Nadalje je mogoče programirati štiri okna, jim posebej določiti naslov in posebej pojasnilo (ukazi OPENW, MENU in INFOW). Skoraj samo po sebi se razume, da so na voljo tudi ustrezni ukazi za brisanje, zapiranje, povečanje prek celotnega zaslona in določitev lege (CLEARW, CLOSEW, FULLW, s parametri pri OPENW). Razvajalci našega basica niso pozabili niti na miško, saj je zanjo na voljo kar pet ukazov (DEFMOUSE, MOUSE, MOUSEK, MOUSEY in MOUSEK).

Za delo z interno urino in koledarjem so na voljo štiri ukazi oziroma sistemske funkcije, ki jih pri basicu ST pogrešamo v celoti. Za izpis časa in datuma so namenjene funkcije, ki jih pri basicu ST pogrešamo v celoti. Za izpis časa in datuma so namenjene funkcije TIMEXS, TIMER in DATES, za nastavitve pa je predviden poseben ukaz SITTIME. Zadnji v začetku povzroča nekaj težav, dokler se ne navadimo na pravi format. Če nastavljamo samo čas, ne pa tudi datum, ne smemo na koncu pozabiti vejice, sicer računalo vztrajno javlja SYNTAX ERROR in nas ne spusti iz napačne vrstice, čeprav bi najraje nadaljevali delo z napačno nastavljenim časom. Kaj hočemo, basic GFA je pač trmast in svojevglav, dokler se ga ne naučimo v vseh podobnostih (avtor članka je imel v začetku nemalo težav s preprostimi IF stavki...).

GFA basic je posebno odličen za strukturano programiranje; tu je zelo blizu pascalu. Pozna naslednje strukture:

```
IF ... THEN ... ELSE ... ENDIF (z možnostjo vključevanja zanke v zanko)
```

```
FOR ... STEP ... NEXT (za NEXT obvezno navesti spremenljivko-števec)
```

```
DO ... LOOP
REPEAT ... UNTIL
WHILE ... WEND
```

Izredno prijetno je, da interpreter pri teh strukturah po vnosu

vrstice sam poskrbi za ustrezen odmik od levega roba. Za primer prikazimo tale kratek izvleček iz programa:

```
Input A
If A=80 Then
Print "POZITIVEN VNOS"
Else
If A=0 Then
Print "VNOS = 0"
Else
Print "NEGATIVEN VNOS"
Endif
Endif
```

Da bi dobili boljše sliko, kako udoben je naš interpreter, povejmo, da dobimo zgornji izpis v prikazani obliki ne glede na to, ali smo pisali z velikimi ali malimi črkami, ali smo detali presledke ali ne, tudi pomisljaj na koncu znakovnih nizov smo lahko izpustili. Nikakor pa ne smemo pozabiti na ENDIF za vsakim IF. Na to napako pa ne opozoril interpreter šele pri izvajanju programa, pa seveda s tem, da bo ostank program za manjkajočim endif umaknjen.

Posebnost za programerje, ki poznajo samo basic, je delo z ukazi GOTO, GOSUB, ON GOSUB, ON BREAK, (CONT, GO-SUB), ON ERROR, ON MENU GOSUB, ker vrstice niso označene s številkami. Mesto skoka za GOTO je treba označiti s poljubno besedo (ali tudi številko), ki si sledi določijo. Subrutine za GOSUB pa se morajo pričeti z rezervirano besedo PROCEDURE, ki si sledi ime (brez narekavajev). Zadnji ukaz obenem nadomešča tudi END, ki je običajno potreben pred prvo subrutino, če so priloženi na koncu programa. V subrutini dokumentaciji je priporočilo, naj bodo subrutine za začetku programa, to priporočilo, ki si sledi ime, najbrž nepremišljeno, saj je potem pred njimi potreben GOTO, sicer se obdelava programa ustavi pred prvo subrutino. Omenimo, da interpreter javi konec programa vsakokrat s posebnim opozorilnim okvrom, ki pa navadno moti, saj prekrjije izpise. Temu se je mogoče izogni-

## Ukazi in funkcije

Poleg ukazov in funkcij, ki jih obsega basic ST (ali basic pri ra-

```
Defline 1,6,0,0
Circle 320,200,190
Line 320,20,320,40
Line 320,360,320,380
Line 140,200,140,200
Line 480,200,500,200
For I=1 To 12
  X=320+Sin(30#Pi/180#I)*175
  Y=200-Cos(30#Pi/180#I)*175
  If <=320 Or Y=200
    Goto 10
  Endif
  PCircle X,Y,5
  10:
Next I
Xen%320
Yen%200
H=Val(Left$(Time$,2))*30
If H>=360 Then
  HH=360
```

```
Endif
M%320
V%200
Gosub Minuta
PCircle 320,200,9
Do
  T=Time
  Sek=Pi/30#Val(Right$(Time$,2))
  X1=320+Sin(Sek)*158
  Y1=200-Cos(Sek)*158
  Defline 1,2,0,1
  Line 320,200,X1,Y1
  If Flag1 And Sek=Pi/30#55+0.1
    Gosub C
  Endif
  If Val(Right$(Time$,2))=0
    Gosub Minuta
  Endif
  If Flag1 Then
    Pause 35
```

```
Else
  Pause 49
Endif
Color 0
Line 320,200,X1,Y1
Gosub Popravek1
Color 1
PCircle 320,200,9
X2=320+Sin(Sek+Pi/30)*158
Y2=200-Cos(Sek+Pi/30)*158
Defline 1,2,0,1
Line 320,200,X2,Y2
If Flag1 And Sek=Pi/30#55+0.1
  Gosub C
Endif
Repeat
Until T<Time
Color 0
Line 320,200,X2,Y2
Gosub Popravek1
```

ti je z neskončno zanko, ki jo je po končanem delu treba prekiniti s pritiskom kar na tri tipke hkrati. Dokaj populn, obširnejši od standarda, je nabor ukazov za grafiko. Za vsak lik je dodatno mogoče predpisati debelino in tip črt, za polnjenje zaključeni likov je za strojno risanje bi bilo prijetno, če bi imeli poleg šrafure na desno tudi šrafuro na levo). Na voljo je tudi nekaj „ekzotičnih“ ukazov, na primer za pravokotnik z zaokroženimi robovi. Manjka pa pravi ukaz za brisanje posameznih likov. Ukaz GRAPHMODE to sicer omogoča do neke mere, vendar ni vedno najboljši. Največkrat je najbolje ponoviti ukaz za lik, pred tem pa z ukazom COLOR izenačiti barvo črt z barvo azotca. Pred naslednjim likom seveda moramo z novim ukazom COLOR vrniti barvo. Pri ukazih GRAPHMODE ponavljanje sicer ni potrebno, vendar se liki ne smejo prekrivati, ker se na prekritih mestih brišejo.

Za programiranje zvoka obstajajo več ali manj standardni ukazi, ki jih poznamo tudi pri drugih dialektih basica. Dokumentacija je tu silno skromna. V naboru ukazov ni zelo pripravne BEEP, namoščesa ga pa lahko PRINT CHR\$(7), ki daje enak zvok. Avtorji dokumentacije so to možnost spredlagali.

Ostaja tudi možnost dela s sprajli, dokumentacija je tu več kakor skromna.

Veliko ukazov se ujema z ukazi pri basicu ST (in pri basicah računalnikov PC) in jih zato ne bi posebej omenjali. Ostane pa še nekaj dokaj „ekzotičnih“ ukazov, kakor na primer ARRAYFILL, ARRAYPR, DIM\*, FATAL, SEE, RELSEEK, UPPERS. Nekateri od njih utegujejo biti tudi koristni.

Za test uporabnosti GFA basica je bil brez poglobljenega študija dokumentacije in brez izkušenj izdelan priložni program, ki izriše na zaslon uro, kakršno vidimo na televiziji pred poročili. Izpis programa je namenoma brez kome-

tarjev. Da bi bil čim krajši in da bi se bolje videla struktura GFA basica. Pri programiranju ni šlo brez težav, morda bi bil lahko program tudi znatno boljok, ko bi bilo več izkušenj. Prva težava je bila ta, da hitrejši kazalec pri prehodu če počasnejšega slednjega zbrise. Verjetno je napaka v operacijskem sistemu, da pri brisanju kazalca ostane nekaj drobnih pikic na mestu puščice. To je bilo mogoče preprečiti samo tako, da je bil nevidni kazalec, ki briše, daljši od vidnega. Zelo neoporno pa je, da se interna ura računalnika premakne naprej samo na vsaki dve sekundi, sekundni kazalec pa se mora premakniti vsako sekundo. Z ukazom PAUSE, ki ga GFA basic na srečo ima, se to lahko reši dokaj elegantno, vendar mora imeti pred celo uro, ko ura daje zvočni signal, pavza dolgo dolžino, sicer kazalec poškodki za dve sekundi.

Po razmeroma kratkih izkušnjah lahko trdimo, da je GFA basic v primerjavi z basicom ST pravo odkritje, čeprav ni povsem brez napak. Morda se bo zanj navdušili celo kak zagrižen nasprotnik basica, zlasti če bo treba na hitro napisati kak ne preveč zahteven program. V takem primeru je prednost interpreterja pred prevajalnikom nedvomna. Marsikomu bo tudi zelo všeč, da na začetku vrstic ni števil, ne smemo pa pozabiti, da utegujejo biti včasih tudi koristne ali celo potrebne. Za take posebne primere pa bi si kot alternativo želeli tudi povsem preprost basic, brez oken in drugih posebnosti, da bi imel le preprost zaslonski editor.

# Beta Basic 3.0

IGOR BIZJAK

**O** Beta Basicu 1.8 je v Mojem mikru že pisalo, zato ga verjetno vsi poznate. Za tiste, ki pa se še niso srečali z njim, tole: Beta Basic je dodatek basicu, ki ga ima mavrica v ROM. Torej ga je treba naloziti v RAM. Z njim lahko pisamo programe, pri katerih bi morali drugoče uporabiti znanje strojnega jezika. Tako lahko že zelo zanimivo igrice napišete v basicu, učinek pa bo enak ali mogoče malo slabši, kar se tiče hitrosti, kot če bi jo napisali v strojnem jeziku.

Toliko za uvod. Z Beta Basicom 3.0 dobite na račun dobrih 18 K okoli 50 novih ukazov in 26 novih funkcij. Program dobite na kaseti skupaj z obsežnimi navodili (88 stram). Na kaseti je poleg BB 3.0 tudi program TURTLE, ki vam s primerom želve grafike prikazuje zmogljivosti Beta Basica.

V uvodu vas najprej seznanijo, kako posneti rezervo (backup) na kaseto in mikrotračnik. Izboljšana je tudi hitrost izvajanja pri daljših programih. Hitrost pridobi pri hitrejšem izvajanju GOTO, GOSUB, RETURN in zank FOR-NEXT, kadar v pomnilnik spravljamo naslov lokacije in ne programske vrstice. Zgolj zaradi primerjave smo oba basica (tistega v ROM in BB 3.0) testirali (benchmark test) in spodaj lahko vidite primerjavo.

## Ukazi

### EDIT <številka vrste>

Vrste, ki so zelo oddaljene od kurzorja, ki označuje tekočo programsko vrstico, dobimo, če po pritisku na ENTER pritisnemo 0 in odtipkamo številko vrstice, ki jo želimo editirati.

### KEYWORDS številka

Z izbrano številko od 0 do 4 dosežemo:

- 0 – izberemo karakterje UDG
- 1 – izberemo ukaze BB 3.0
- 2 – izberemo vnašanje ukazov na

običajen način  
 3 – izberemo vnašanje ukazov mešano, znak po znaku, ali na običajen način  
 4 – izberemo vnašanje ukazov samo znak po znaku  
**LIST FORMAT številka**  
 Z izbrano številko od 0 do 5 dosežemo:  
 0 – listing enak ZX basicu  
 1 – listing, pri katerem je vsak ukaz v novi vrstici

BENCHMARRK TEST		
	ZX BASIC	BB 3.0
BH1	4' 58	2' 38
BH2	8' 70	9' 78
BH3	21' 10	24' 10
BH4	20' 40	21' 55
BH5	24' 00	25' 00
BH6	55' 30	43' 50
BH7	59' 70	74' 70
BH8	253' 00	220' 50
Σ	58' 50	52' 57

2 – enako kot pri 1, le da so stavki zamaknjeni glede na ukaze FOR, DO, DEF PROC, IF, ON  
 3 – enako kot pri 1, le da ni števil programskih vrstic  
 4 – enako kot pri 2, le da ni števil programskih vrstic

**CSIZE širina <višina>**  
 Lahko povečate ali zmanjšate velikost znakov. Lahko dobite 64 znakov v vrstici.  
**JOIN in SPLIT**  
 Z JOIN lahko dve vrstici združite v eno, če pa postavite znak < > za vrstici in pritisnete ENTER, dobite dve programske vrstici.

## Snemanje

**DEFAULT = m/n/b številka**  
 m – mikrotračnik  
 t – kasetofon  
 n – mreža spectrumov  
 b – RS232, byte kanal  
 S tem ukazom lahko določimo napravo, s katero bomo delali. Tako je npr. dovolj, če hočemo posneti program na mikrotračnik 1, da napišemo **DEFAULT = M1: SAVE "IME";**  
**SAVE <del programa>; <mikrotračnik>; ime**

Posnamo lahko samo del programa. S **SAVE 30 TO 100: "ime"** bomo posneli samo program od vrstice 30 do vrstice 100.  
**SAVE DATA <mikrotračnik>; ime**

S tem ukazom posnamo samo spremljalnikve.  
**MERGE**

Pri mikrotračniku lahko sedaj lepimo tudi avtomatsko tekočo (auto-run) programe.  
**MOVE**

Sedaj lahko premikamo tudi programe, strojno kodo in podat-

```

Color 1
Pcrlcrlc 320,200,9
If Sek=Ura*P/15 And Sek=Ura*P/30 Then
  Gosub Popravek2
EndIf
Loop
Procedure Minuta
  Define l,1,0,0,1
  Color 0
  Line 320,200,Xm%*Sin
  (Min)*3,Ym%-Cos(Min)*3
  Line 320,200,Xh%*Sin
  (Ura)*3,Yh%-Cos(Ura)*3
  Min=P/180+Val(Plot(Times,4,2))
  Xm%=320+Sin(Min)*156
  Ym%=200-Cos(Min)*156
  Define l,1,5,0,1
  Color 1
  Line 320,200,Xm%*Ym%
  If Min=0
    Gosub E
    EndIf
    Flag=1
  Else
    Flag=0
    EndIf
    H=Val(Left$(Times,2))*30
    If H=360 Then
      H=H-360
    EndIf
    Ura=P/180+H/Min/12
    Xh%=320+Sin(Ura)*120
    Yh%=200-Cos(Ura)*120
    Line 320,200,Xh%*Yh%
    Pcrlcrlc 320,200,9
    Return
  Procedure Popravek1
    Xh%=320+Sin(Min)*156
    Yh%=200-Cos(Min)*156
  
```



kovne baze, le da moramo pri tem imeti vsaj dva mikrotlačnika.

## Delo s podatki

**JOIN aS <del> TO bS <smr>**  
**JOIN a <del> TO b <smr>**  
**COPY a <del> TO b <smr>**

Z JOIN in COPY lahko prepisemo ali prekopiramo del ali vso spremenljivko ali drugo.

**DELETE a - <del> ali b <del>**  
Pobrišemo lahko del ali vso vsebino spremenljivke.

**SORT a - ali b <del> <del>**  
Sort nam zelo hitro sortira po abecedi ali po številkah.

**INARRAY (a - <začetni element <del> -), b -)**  
INSTRING (<začetek, a -, b -)

S tema funkcijama iščemo po nizu določen niz.

**LENGTH (številka, -ime niza)**  
Pove dolžino niza. Če je številka 1, nam pove dolžino prve dimenzije, če je 2, druge dimenzije. Ne dela z več kot dvema dimenzijama

**CHAR - (številka)**  
**NUMBER (niz iz dveh znakov)**

Prva funkcija pretvori celo število med 0 in 65536 v niz, sestavljen iz dveh znakov. Druga pa stori ravno nasprotno.

**USING <format>**  
**USING a -, številka)**

Omogoči lepši izpis rezultatov.  
Npr. PRINT USING "###.##";

33.1234 bo izpis 33.1.

**EDIT a - ali EDIT <del>**  
Editorimo lahko tudi spremenljivke.

**EOF (številka kanala)**  
Znaka, kdaj je zadnji element prebran z mikrotlačnikove datoteke.

## Delo z grafiko

**ALTER <atribut> TO atribut**  
Spreminjamo lahko attribute na zaslonu na hitrejši in bolj zanimiv način.

**DRAW TO x, y, <z>**  
Risaneje daljše do določene točke (x, y).

**GET B -, x, y, širina, dolžina**  
S tem ukazom shranimo v aS področje na ekranu, ki je podano v nadaljevanju ukaza. Koordinati x in y sta v točkah, širina in dolžina pa v znakih.

**PLOT x, y, <a>**  
Kar smo popelj z GET shranili, lahko sedaj narišemo na ekran z ukazom PLOT. Siliko lahko prenesemo tudi večjo, npr. ukaz PLOT CSIZE 2; 10,10;A - bo na 10,10 narisal vsebino a - trikrat večjo.

**CSIZE širina < dolžina >**  
Poveča ali zmanjša znak, znake UDG, slike, shranjene z GET, itd.

**POKE naslov, niz**  
Poke nam omogoči da večji del spomina naenkrat popokamo v pomnilnik.

**FILL <INK ali PAPER barva; <x, y>**  
Zapolni področje z izbrano barvo.

**ROLL smer <pikselov><x, y; širina, dolžina>**

Izbrani del ekrana lahko premikamo (scrolliramo) pri čemer se nam vsebina vrne na zaslon, v želeno smer z izbrano hitrostjo.

**SCROLL smer <pikselov><x, y; širina, dolžina>**  
Enako kot pri ROLL, le da izgubimo vsebino ekrana.

**SCRN - (vrstica, stolpec)**  
Prepoznata tudi znake UDG in ne samo znake ASCII

**WINDOW številka <, x, y, širina, dolžina >**  
Okno za izpis teksta. Vsako definirano okno ima svoje izpisne (print) pozicije, barvo, OVER, BRIGHT, FLASH in je oštevilčeno od 1 do 127.

**XOS, YOS, XRG, YRG**  
Spremenljivke, s katerimi določimo začetek koordinatnega sistema in dolžino pri x in y.

**OVERZ**  
Z OVER 2 lahko sedaj rišemo čez ribno po načinu OR.

**SINE (številka), COSE (številka)**  
Hitrejši funkciji sinus in cosinus.

**FILLED ()**  
Pove nam površino področja, ki smo ga zapolnili s FILL.

**MEMORY - ()**  
Vrne del pomnilnika kot niz.

## LIST ali LLIST REF reference

Izpis programske vrstice, v kateri je podana reference.

**REF reference**

**REF spremenljivka**  
Išče po programu dano referenco ali spremenljivko. Ko je najde, izpiše programske vrstico in postavi kurzor na začetek iskane reference.

**RENUM <:, > <del> <LINE številka > <STEP številka >**

Brez asterisk (\*) na začetku preštevilči, z njim pa prekopiira del programa ali vse program.

**MEM()**  
Izpiše, koliko je še razpoložljivega pomnilnika.

## Strukturirano programiranje

**DEF PROC ime <parameter>**  
**<.REF parameter>**  
**END PROC**

Ko želimo definirati novo proceduro, na začetku na začetek ukaz DEF PROC; določimo ime, parametre (če je potrebno) in na koncu damo END PROC.

**<PROC> <ime <parameter>**  
**<parameter>...**

Z imenom procedure proceduro pokličemo in jo tudi izvršimo

**LOCAL spremenljivka <.spremenljivka>...**

Spremenljivke v okviru Procedure določimo kot lokalne.

**DEFAULT spremenljivka = izraz <.spremenljivka = izraz>...**

Spremenljivkam lahko v Procedure tudi določimo izraz, če spremenljivka pri vohodu v proceduro ni definirana.

**REF spremenljivka**

Dosedaj smo aktualne parametre, ki smo jih navedli pri klicu procedure, posredovali formalnim v sami definiciji procedure.

Če pa želimo dobiti kak rezultat iz procedure in ta rezultat shraniti v kako spremenljivko, potem v def procedure uporabimo REF.

npr.: DEF PROC zamenjaj REF aS,REF bS

**LOCAL IS**  
**LET tS=aS, aS=bS, bS=tS**

**END PROC**  
pokličemo proceduro z: LET xS="dober dan", yS="lahko noč"

zamenjaj xS, yS

PRINT xS, yS

**ITEM()**  
Funkcija nam pove informacije o naslednjem podatku, ki ga bomo prebrali z ukazom READ. Funkcijske vrednosti so naslednje:

0 - če so vsi podatki prebrani v tekočem stavku DATA

1 - če je naslednji podatek niz

2 - če je naslednji podatek številski

**DO... LOOP**  
**DO WHILE pogoj... LOOP**  
**DO UNTIL pogoj... LOOP**  
**EXIT IF pogoj**

DO LOOP je zanka, podobna FOR...NEXT, le da ima nekaj prednosti. Vgnezdenje je enako

kot pri zanki FOR NEXT. Izhod iz zanke je možen z DO WHILE, DO UNTIL, in EXIT IF, če ima zanka samo DO BRE WHILE in UNTIL.

npr.: LET vsota = 0  
DO UNTIL vsota > 100 ali DO WHILE vsota <= 100

INPUT "Vstavi številko":x  
LET vsota = vsota + x  
PRINT vsota

LOOP  
PRINT "Vsota je čez sto"

**ELSE v povezavi z IF... THEN**

Če je pogoj v stavku IF - THEN napačen, potem se bo izvajanje nadaljevalo v naslednjem stavku.

Če pa je na koncu stavka IF - THEN ELSE, pa se bo izvajanje nadaljevalo v istem stavku, seveda če bo pogoj v stavku IF - THEN napačen.

**ON kot GOTO ali GOSUB ON številko; št. vrstice, št. vrstice, ... ali ON številko: ukaz; ukaz;...**

Vrednost številka za ON nam pove, na katero številko vrstice moramo program skočiti. V drugem primeru pa se bo izvajanje nadaljevalo v ukazi, ki je določen v vrednosti številka.

npr.: INPUT "Vstavi 1 do 4": izbrana; GOTO ON izbrana; 90, 120, 30, 5

10 INPUT x

20 ON x: PRINT "ena": PRINT "dva": PRINT "štiri"

30 GOTO 10

**ON ERROR št. vrstice**  
**ON ERROR: ukaz; ukaz;...**

V prvem primeru se izvajanje programa, če pride do napake, nadaljuje v vrstici, ki je podana za ukazom ON ERROR. V drugem primeru pa v isti vrstici.

Vrednost napake se vsakokrat shrani v spremenljivko ERROR. Obstajata še dve spremenljivki, ki se med izvajanjem programa ažurirata in ju lahko uporabimo, ko ju potrebujemo. To sta LINO, ki nam da število stavka, ki naj bi se izvršilo, in STAT, ki nam da ukaz v stavku, ki naj bi se izvršilo.

# C 64 kot voltmeter

MIRAN VOZLIČ

**S**tem člankom si boste z malimi stroški napravili dodatek, s katerim bo vaš računalnik postal digitalni voltmeter. O uporabi takega voltmetra je bilo v *Mojem mikro* že veliko napisanega. Za osvežitve spomina vam priporočam, da preberete številke, ki so izšle od julija do novembra leta 1984. V teh številkah je inž. Mitja Borko zelo nazorno opisal osnovne možnosti povezave mikroračunalnika z zunanjim svetom. Za izdelavo našega voltmetra bomo uporabili A/D (analogno/digitalni) pretvornik oznake CA 3162E proizvajalca RCA.

oznake CA 3161E, ki krmlji tri sedemsegmentne displeje, tri tranzistorje, dva trimera in integracijski kondenzator. Za priključitev A/D pretvornika na računalnik pa poleg tega potrebujemo samo še dva trimera, integracijski kondenzator in tri kondenzatorje, ki imajo nalogo, da blokirajo molimolni impulz, ki nastajajo pri multiplexerju. Nalogo integriranega vezja CA3161E, treh tranzistorjev in sedveda displejev bo opravljal računalnik z ustreznim programom. Glavna prednost A/D pretvornika pred drugimi je ta, da potrebuje za delovanje samo pozitivno napetost +5 V, ki nam je na voljo iz računalnika. Pretvornik sam ni namenjen za priključitev na računalnik in podatki na izhodu niso v binarni obliki kot pri pretvornikih, ki so namenjeni za računalnike. To je hkrati dobra in slaba stran. Slaba zato, ker potrebujemo program, ki bo prebral informacijo na izhodu in dobri, ker je resolucija merilnega obsega precej večja. Pri osemitnih pretvornikih imamo 256, pri našem pretvorniku pa 1098 odčitkov, od -99 do +999. Izhodni podatki pri IC (Integrated Circuit) CA 3162E so v obliki multipleksne kode BCD. Kaj več o tme pozneje.

Pretvornik deluje na principu dvojnega nagiba (DUAL-SLOPE). Samo načelo je nekoliko bolj komplicirano, zato ga na tem mestu ne bi opisoval. Literature, v katerih je princip opisan, pa je tudi pri nas veliko. Pretvornik lahko odvisno od priključitve šestega priključka opravi 4 ali 96 pretvorb na sekundo. Če je šest priključek priključen na napetost +5 V, pretvornik opravi 96 pretvorb; ko je priključen na maso ali nepriključen, pa 4 pretvorbe. Posebno stanje je, ko na šestih priključkih prejemo napetost od +0,8 do +1,6 V. Takrat se zadrži zadnja pretvorba «HOLD». V našem primeru bomo šest priključek priključili na +5 V.

podatki na izhodu BCD pojavljajo drug za drugim. V našem primeru, ko imamo trimeristi pretvornik, multiplexer predstavlja trije priključki. Izhod BCD predstavlja 6 štirje priključki, s katerimi lahko predstavimo v binarnem formatu 16 različnih stanj. Pri našem pretvorniku se lahko pojavi na izhodu BCD 12 različnih stanj. Vse možne kombinacije so prikazane v tabeli. Skupno nam torej izhod predstavlja sedem priključkov oz. bomo za dekodiranje podatka na izhodu potrebovali sedem bitov.

Poglejmo si sedaj sliko št. 4, ki nam prikazuje prej omenjeno tabelo in razporeditev priključkov IC ter njihov pomen. Podatki na izhodu IC se pojavljajo na naslednji način. Najprej leži na izhodu BCD vrednost najpomembnejše cifre (MSD), oca. 5 ms, kar nam signalizira z nizkim impulzom priključek št. 4. Po preteku 5 ms se istočasno menja informacija na izhodu BCD in na izhodu multiplexerja. Sedaj je izhodu multiplexerja na priključku št. 5, ki predstavlja najmanj pomembno cifro (LSD). Informacija na izhodu BCD velja torej za enice. Po naslednjih 5 ms se ponovno menja informacija na izhodu BCD in nizki impulz je na

## A/D pretvornik CA3162E

CA 3162E je monolitiko integrirano vezje, namenjeno pretvarjanju analognih velikosti v digitalno informacijo. CA 3162E je pravzaprav namenjen za izdelavo posebnih trimeristnih digitalnih voltmetrov z minimalnim številom dodatnih elementov. Vezje vsebuje poleg omenjenega pretvornika še sedemsegmentni pretvornik BCD

## Izhod pri IC CA 3162E

Da napišemo program, ki bo pravilno prebral vrednost podatkov na izhodu pretvornika, moramo poleg malo znanja o strojnem programiranju poznati še, kako se pojavljajo podatki na njegovem izhodu. Izhod pri tem A/D pretvorniku in še veliko drugega je zapisano v obliki multipleksne kode BCD. Multipleksno pomeni, da se

1000	100	*****		0356 20 78 03	400	JSR ASCC0D	PRET.V ASCII COD
1000	110	JA		0359 05 F8	500	STA LSD	SHRANI LSD
1000	120	JA	3 DIGIT VOLT METER CA3162	0358	510	J	
1000	130	JA	DVM V1	0359 20 69 03	600	LOOPE	JSR READPB /READ PORT B
1000	131	JA		035E 4A	530	LSR A	BIT 2 V CARY
1000	132	JA	BIT ON USER PORT	035F 4A	535	LSR A	
1000	133	JA		0360 4A	540	LSR A	
1000	134	JA	0 -- LEAST SIGNIFICANT DIGIT	0361 00 F8	600	BCL LOOP2	BCS LOOP2
1000	135	JA	1 -- MOST SIGNIFICANT DIGIT	0362 20 78 03	570	JSR ASCC0D	PRET.V ASCII COD
1000	136	JA	2 -- NEXT SIGNIFICANT DIGIT	0366 05 F8	500	STA NSD	SHRANI NSD
1000	137	JA	3 -- BCD INFORMATION 210	0369	530	J	
1000	138	JA	4 -- BCD INFORMATION 211	0369 60	600	RTS	
1000	140	JA	5 -- BCD INFORMATION 212	0369 AD 01 DD	610	READPB LDA PORT B	PREBERI PORT B
1000	150	JA	6 -- BCD INFORMATION 213	036C 05 FA	620	STA MDEG	SHRANI V MDEG.
1000	160	JA	7 -- FREE	036E AD 01 DD	630	LDA PORT B	PREBERI PORT B
1000	200	JA		0371 C5 FA	640	CPF MDEG	ENKAD KOT PREJ
1000	210	JA	***033C-8038C V.M. 12/85	0373 00 4A	650	BCS READPB	JE POTEH READPB
1000	220	JA		0375 29 7F	660	AND #0B1111111	IBRISI BIT 7
1000	240	J		0377 60	670	RTS	JA POTEH NAZAJ
1000	245	J		0378	680		
033C	250	***033C	INŠLOV ZACETA PRG.	0378 C3 0A	690	6SC0D CPF #80	7PB=10
0063	250	DOR0	#DOR0	037A F0 00	700	BCF MINUS	POTEH ZNJK (-)
0081	270	PORTB	/DATA DIREC. REG. B	037C	710	J	
00F7	280	M5D	#M5	037C 03 00	720	CPF #80B	7PB=11
00F8	290	M5D	#M5D	037E F0 00	730	BCF PLUS	POTEH ZNJK (+)
00F9	300	L5D	#F5	0380	740	J	
00FA	310	MDEG0	#FA /MDEG0	0380 10	750	CLC	CLEAR CARY FLAG
033C	320			0381 69 30	760	AND #90B	ASCII ŠTEVILA
033C A9 00	330	START	LDA #800	0380 60	770	RTS	INZAJ
033E 03 03 DD	340	STH DOR0	/PORT B INPUT	0384 10	780	MINUS	J
0341	350	J		0385 69 23	800	ADC #823	ASCII ZNJK (-)
0341 20 69 03	360	LOOP3	JSR READPB /READ PORT B	0387 60	810	RTS	INZAJ
0344 4A	380	LSR A	/BIT 1	0388	820	J	
0345 4A	390	LSR A	/V CARY	0388 10	830	PLUS	CLEAR CARY FLAG
0346 00 F8	400	BCL LOOP3	/BIT 1+1 READPB	0389 69 20	840	AND #80B	ASCII ZNJK (+)
0348 4A	405	LSR A		0388 60	850	RTS	INZAJ
0349 20 78 03	410	JSR ASCC0D	/PRET.V ASCII COD	038C	860	END	
034C 05 F7	420	STH M5D	SHRANI M5D	038C	870		
034E	430	J		LINE\$13	SYMBOL\$14	ERROR\$16	
034E 20 69 03	440	LOOP1	JSR READPB /READ PORT B	ASC0D=#37F	DOR0	#D03	LOOP1
0351 4A	450	LSR A	/BIT 0 V CARY	MDEG0=#0FA	MINUS	#384	M5D
0352 00 FA	460	BCL LOOP1	/BIT 0+1 READPB	READPB=#369	START	#33C	LOOP2
0353 4A	470	LSR A	/SELEC. BCD	LOOP2=#038	LOOP3	#341	L5D
0355 4A	480	LSR A		NSD	#0F8	PLUS	#368
					PORT B	#00F	

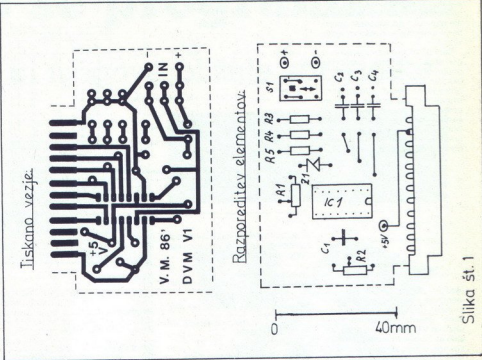
priključku št. 3 (NSD) in tako naprej. Za lažje razumevanje si ogledajte še časovni diagram (slika št. 3).

Če se na izhodu BCD pojavi naslednja binarna informacija 1011, desetistihi 11, pomeni, da je vhodna analogna velikost prevetlika (pozitivna preobremenitev). Negativna preobremenitev pa je takrat, ko se na vseh treh cifrah mestih pojavi binarna kombinacija 1010, desetistihi deset. Negativna napetost do -99 mV se označuje tako, da najvišja cifra MSD dobi binarno kombinacijo 1010.

## Izdelava in priključitev modula

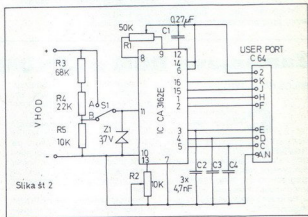
Poleg že opisanih elementov naše vezje vsebuje še tri upore, ki nam rabijo za razširitev merilnega območja. Z navedenimi upori smo razširili merilno območje za desetkrat. Če je preklopnik S1 v položaju A, lahko merimo napetosti od -99 do 999 mV in ko je v položaju B, napetosti od -0,99 do 9,99 V. Če želite drugačna merilna območja, morate navedene upore nadomestiti z drugimi. Toleranca uporov pa naj ne bo večja od enega odstotka. Zener dioda na vhodu ima nalogo da štiti vhod A/D pretvornika prevetlike napetosti. Zener dioda lahko tudi izpustite, če ste prepričani, da vhoda ne boste preobremenili za več kot desetkrat; po nekaterih podatkih tudi do petnajstkratna preobremenitev ne poškoduje pretvornika.

Tiskano vezje izdelajte po sliki št. 1. Luknje izrtajte s svredrom 0,8 mm, vstavite podnožje in druge elemente ter jih prispajkajte. Konektor prispajkamo direktno na tiskano vezje in sicer tako, da prispajkamo samo spodnje priključke. Od zgornjih priključkov pa povežite z žico priključek št. 2 (+5V) s ponko +5V na tiskanem vezju. Ko ste vse skupaj pregleda-



li, lahko vstavite v podnožje IC. Pri tem pazite, da se ne boste dotikali priključkov IC, saj ga lahko s svojo statično elektriko uničite. Prav tako bi vas opozorili, da bodite pri nakupu tega IC previdni. IC mora biti shranjen v antistatični embalaži.

Trimerja nastavite približno na polovico in sedaj lahko vstavite modul pri IZKLJUČENEM računalniku v razširitevna vrata (USER-PORT). Modul vstavimo tako, da so elementi zgornji! Naložite in startajte program. V zgornjem levem kotu se bo pojavila trimesna številka. Sedaj moramo



A/D pretvornik pravilno umeriti. Najprej kratko sklenemo vhod A/D (priključek št. 11 povežemo z maso) pretvornika in s trimerjem R1 nastavimo, da bo vrednost pokazanega števila 000 mV. Nato priključimo na vhod neko znano in čimbolj točno napetost, npr. 900 mV, in s trimerjem R2 nastavimo tako dolgo, da bo na zaslonu prav tako 900 V; pri tem je stikalo S1 v položaju A. Pri umerjanju si lahko pomagamo tudi z drugim digitalnim voltmetrom razpada 0,1 odstotka. Vhodna upornost pretvornika je zelo velika, razreda 100 Mohmov, in če imamo odprte sonke, lahko dobimo na izhodu

neko vrednost, ki je posledica statičnih nabojev na vhodu A/D pretvornika.

## Softver

Še nekaj besed o samem programu. Strojni podprogram je zapisan v izvirni obliki s komentarjem in je tudi disasembliiran. Za vse tiste, ki ne obvladajo strojnega programiranja, pa je strojni del podprograma napisan v glavnem programu v obliki stavkov data. Program nam omogoča samo prikazovanje posameznih cifler na ekranu. Kako ga boste uporabili, pa prepustimo vam. Priporočam ▶

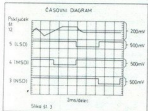
```

100 REM *****
110 REM *
120 REM * DVM V1 VOZELIC MIRAN RIJBB *
130 REM *
140 REM *****
150 I
160 PRINTCHR(147);REM IZBRISJE EKRAH
170 VS=BI:REM VSOTA
180 I
190 FOR I=000 TO 999:REM VPIS STROJNE RUTINE
200 REM X:VS=VS+X
210 POKE I,X
220 NEXT I
230 IF VS<3844 THEN PRINT"NAKVA V DATR STAVKIH I";:END
240
250 BE:~
260
270 SYS28:REM KLIC STROJNE ROTINE
275 FOR I=007 TO 348:REM BRISAJE POSAMEZNIH CIFER
280 AB=CHR(PEEK(I));
290 BE=CHR(LEFT(AB,1));REM ZDRUZEVANJE CIFER V NIZ SPREHVALJIVO
300 REM X:~
310 PRINT"Q";:REM PONIK KURSORJA IN IZPIS SPREHVALJIVE
320 GOTO 250
330
331 DAT189,0,141,3,021,32,105,3,74,74,176,249,74,32,100,3,133,047,32,105
332 DAT34,74,176,236,74,74,38,100,3,133,249,32,100,3,74,74,74,176,249,32
333 DAT176,133,249,96,179,1,021,133,250,173,1,021,197,250,090,244,41,107
334 DAT306,291,108,249,49,291,11,040,0,24,105,49,96,24,105,35,06,24,105,32
335 DAT346

```

vam, da v program vključite merilno območje, grafično prikazovane napetosti itd. Možnost uporabe je omejena le v vašo iznajdljivostjo.

Sam potek programa je takle. Iz glavnega programa pokličemo podprogram, strojno rutino, z ukazom SYS 828. Naloga strojne rutine je, da dekodira vrednost na izhodu A/D pretvornika in shrani posamezne cifre na nastave F7-F9 šestnajstično oz. desetiško 247-249 in se vrne nazaj v glavni program. Z ukazom PEEK (n) v zanki prebere posamezne cifre in jih z



ukazom LEFT\$ združi v niz spremenljivke z imenom b\$. ki jo pa v primeru, da želimo z njo matematično operirati, spremenimo z ukazom VAL(b\$) v neko drugo realno ali celoštevilčno spremenljivko, npr. u. Negativno preobremenitev označujejo znaki --, pozitivno pa +++.

Pri uporabi tega dodatka in pisanju programa vam želimo velik

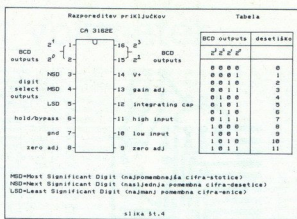
## Zamenjava ROM z EPROM

MILOŠ NOVKOVIČ

**Č**e pride do okvare ROM, je spectrum neuporaben, dokler mu ne vstavimo nov ROM ali programiran EPROM. Drugi razlog za zamenjavo je, da mu prazen prostor zapolnimo z lastnimi programi in mu s tem razširimo možnosti. Prednost razširitev je, da imamo možnost izbire željenega EPROM.

Zato je treba izvesti nekaj sprememb, ki so predvidene že pri izdelavi tiskanega vezja. To je potrebno zaradi razlik v hitrosti delovanja ROM oz. EPROM, kakor tudi ULA.

ULA je posebej projektirano integrirano vezje, namenjeno delu z vso periferijo (tipkovnico, kasetofonom) in za delo z video spominom. Zlasti pomemben je način dela z video spominom. ULA komunicira z video spominom istočasno, ko mikroprocesor komunicira z ROM. Ker se preko ULA dovajajo signali A14 in A15 ter MREQ in MREQ, izvaja ULA komunikacijo z ROM, obenem pa osvežuje video RAM



uspeha. Če imate kakšna vprašanja v zvezi s to temo, vam bom rad odgovoril v reviji.

Uporabljena literatura.  
Linear Integrated Circuits, RCA USA/11-78  
Intern 64, DATA BECKER  
Der Commodore 64 und Rest der Welt, DATA BECKER  
Programieren in Maschinsprache mit dem C-64, C. Lorenz

### Seznam materiala

Trimer potenciometra (precizna)

- R1..... 50 Kohm
- R2..... 10 Kohm

- Upori**  
R3.... 68 Kohm  
R4.... 22 Kohm  
R5.... 10 Kohm

- Kondenzatorji**  
C1.... 0,27 uF  
C2-C4... 3,3 ali 4,7 nF

- Polprevodniki**  
IC1... CA 3162E  
Z1... ZPD 3,6-8,2 V

- Drugi elementi**  
S1..... Preklopnik za tiskano vezje  
Konektor za razširilivna vrata s kontakti 2+22 in medsebojno razdaljo 3,96 mm

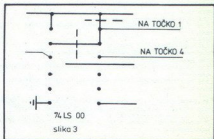
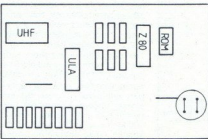
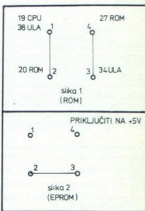
**BARVNI TV SPREJEMNIKI**

emona commerce  
**tozd globus**  
Ljubljana, Šmarinska 130

Krednagracija prodaja  
ISP  
Tovarna 21  
Ljubljana  
(01) 324-766, 324-677

**ORION**

Made in Japan



# Učimo se programirati

## MC 68000 in njegove bližnje sorodnike

MILAN SLUNEČKO

*Dolgo sta v naših krajih kraljevala Z 80 in 6502. Daljše zime 1984 pa je tudi k nam zabodli prvi križanec med 32, 16 in 8-bitnim procesorjem. Skrit je bil v črni škattli z napisom Sinclair QL in se je imenoval MC 68008. Morda ravno zaradi tega, ker je prišel pozimi, ni mogel preveč ogreti src naših komodorjevcev in mavričarjev. Več sreče je imel Jack Tramiel. Čeprav so njegova računalniki iz serije ST prav tako prišli pozimi, je njihova čistokrvna 16-bitna arhitektura močno ogrela marsikoga z ne preveč globokim žepom.*

**K**ako je pravzaprav nastala serija MC 68000? Motorola je že zelo dolgo znana po svojih polprevodniških elementih. Poleg vseh vrst tranzistorjev, triacov, diacov in diod izdeluje digitalne elemente, med katerimi je najbolj pomembna centralna procesna enota Motorolin prvi mikroprocesor je bil MC 6800: osembitni, z dvema akumulatorjema, šestimi načini naslavljanja in 72 močnimi ukazi. Ta procesor je zaslovel, ker je bil najbolj enostaven za uporabo. K tej generaciji sodita še procesorja MC 6802 in MC 6808. Programsko sta enaka kot prvi, vendar so v MC 6802 že vdrali pomnilnik in uro.

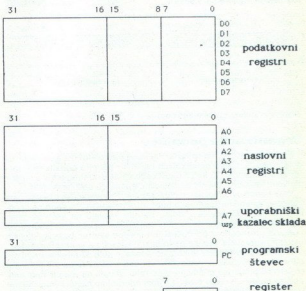
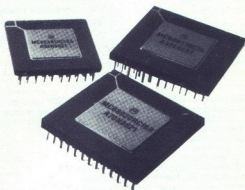
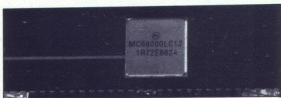
V drugi generaciji so MC 6801, MC 6803 in MC 6805. V vse so že vdrali RAM, ROM in nekaj VII vodi. To so v bistvu mikroročunalniki na enem čipu. Tako smo hitro prišli tudi do tretje generacije, katere predstavnika sta MC 6809 in MC 68000. MC 6809 je 8-bitni mikroprocesor s 16-bitno notranjo strukturo. Ima pet 16-bitnih registrov in je prirejen za učinkovito obdelavo višjih programskih jezikov. Navzven je popolnoma združljiv z vezji in s periferejama, narejenimi za MC 6800.

Procesorja MC 68000 in MC 68008 imata 32-bitno notranjo arhitekturo in sta programsko skoraj enaka. Edina razlika je v tem, da lahko MC 68000 naslavlja 16 Mb pomnilnika (24-bitno naslovno vodilo), MC 68008 pa le 1 Mb (20-bitno naslovno vodilo). MC 68010 jima je zelo podoben, le da ima nekaj instrukcij in registrov več. Najmočnejši iz serije M 68000 je MC 68020: ta je popolnoma 32-bitni, ima nekaj novih načinov naslavljanja in razširjen nabor ukazov, ki obsega popolno 32-bitno aritmetiko, delo s koprocesorji, softverskimi moduli in operacije nad bitnimi polji.

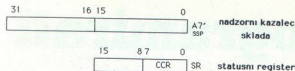
MC 68000 lahko deluje v uporabniškem (user) ali nadzornem (supervisor) modusu. Slednji je v glavnem namenjen operacijskim sistemom in sistemskim programom. Razlog za to je v zgradbi procesorja. Nadzorni modus dopušča uporabo nekaterih dodatnih instrukcij in privilegijev. Že ime pove, da je uporabniški modus namenjen predvsem programom, ki jih napišemo sami.

Kot kaže slika 1, nam MC 68000 ponuja šestnajst 32-bitnih registrov za osnovne namene (d0–d7, a0–a7), 32-bitni programski števec in 8-bitni register stanj. Registri d0–d7 so uporabljeni kot podatkovni registri. Lahko jih izkoristimo za 8-bitne (byte, zlog), 16-bitne (word, beseda) in 32-bitne (long word, dolga beseda) operacije. Naslednjih 7 registrov (a0–a6) je kazalec sklada (USP ali a7) lahko uporabljamo kot programske kazalce sklada in osnovne naslovne registre. 2 naslovni registri lahko operiramo le s 16 ali z 32 biti. Vse te registre lahko uporabimo tudi kot indekse.

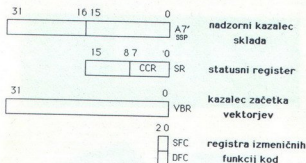
V nadzornem modusu je nekaj dodatnih registrov, ki jih kažeta sliki 2 in 3. Pri MC 68010 sta poleg nadzornega kazalca sklada (SSP) in statusnega registra na voljo dva 3-bitna registra, ki dajeta nadzorniku dostop do uporabniškega pomnilnika ali emulacijo praznih ciklov procesorja. Dodatni 32-bitni register nam pove, kje naj se začne tabela sektorjev za notranje in zunanje dogodke, kot so npr. prekinitev, pasti (traps) in nekateri drugi. O njih bomo govorili nekoliko pozneje.



Slika 1. Registri, namenjeni uporabniku



Slika 2 Registri, ki so dostopni le v nadzornem modusu (MC 68000, MC 68008)

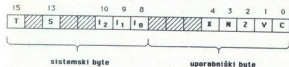


Slika 3 Registri, ki so dostopni le v nadzornem modusu (MC 68010)

Statusni register je sestavljen iz sistemskega (biti od 8 do 15) in uporabniškega dela (biti od 0 do 7). Slednji se imenuje register stanj (CCR - Condition Code Register). Uporabljani so le biti od 0 do 4, njihove funkcije pa so:

- bit 0: carry flag (zastavica prenosa)
- bit 1: overflow flag (zastavica prekoračitve)
- bit 2: zero flag (ničelna zastavica)
- bit 3: negative flag (zastavica negativnosti)
- bit 4: extend flag (zastavica razširjenja)

V sistemskem delu statusnega registra nam biti od 8 do 10 povedo, katera od skupaj 8 stopenj prekinitve je dovoljena. Bit 13 pove, v katerem modusu dela procesor (nadzornem ali uporabniškem). Bit 15 vklopi ali izklopi modus TRACE, s katerim spremljamo ukaze Strukturnega statusnega registra kaže slika 4



Slika 4 Shema statusnega registra

## Organizacija podatkov

Povedali smo že, da je lahko velikost operandov zlog, beseda ali dolga beseda. Osem podatkovnih registrov podpira podatkovne operande dolžine 1, 8, 16 ali 32 bitov, sedem naslovnih registrov skupaj z aktivnim kazalцем sklada pa naslovne operande dolžine 32 bitov.

Vsak podatkovni register je dolg 32 bitov. Pri operacijah z byti se nam ustrezno spreminja spodnjih 8 bitov, pri operacijah z besedo pa s spodnjih 16 bitov. Operacije z dolgo besedo vplivajo na vseh 32 bitov podatkovnega registra. Kadar je ta uporabljen kot operater (source) ali operand (destination) pri operacijah dolžine byta ali besede, se nam ustrezno spreminja spodnji del registra, zgornji pa je nespremenjen.

Vsak naslovni register in kazalec sklada je dolg 32 bitov in hrani 32-bitni naslov, čeprav je naslovnih linij samo 24 oziroma 20. Naslovi registerji ne podpirajo operacij z byti. Tako so nam na voljo le operacije dolžine besede in dolge besede. Kadar uporabimo naslovni register kot operand, se spremeni vseh 32 bitov, ne glede na velikost operacije.

Instrukcije M 68000 vsebujejo dve vrsti informacije. Prva je tip funkcije, ki se mora izvesti, druga pa nam pove, kje so operandi, s katerimi se funkcija izvaja. Splošna oblika ukaza je:

**ukaz.(dolžina I, w ali b) izvir(source), cilj,(destination)**

Kdor je kdaj programiral Z 80, 6502 ali kaj podobnega, je verjetno opazil, da sta tu izvir in cilj zamenjana. Pri naslavljanju bomo v primerih v glavnem uporabljali ukaz **move**, ki je enakovreden ukazu LD pri procesorju Z 80. Njegova naloga je premetavanje podatkov med registri ter med registri in pomnilnikom. Preden se seznanimo z načini naslavljanja, si zapomnimo še eno zelo pomembno pravilo.

**Kadar želimo brati besedo ali dolgo besedo iz pomnilnika, moramo paziti, da je nilen naslov sod. Če tega ne storimo, program po vsej verjetnosti ne bo delal.**

## Naslavljanje

MC 68000 pozna naslednje načine naslavljanja.  
 - **Neposredno podatkovno naslavljanje** (data register direct). Operand je v podatkovnem registru. sintaksa zbirnika je Dn, kjer n pomeni številko podatkovnega registra. Primeri:

**move.l d1,d2** Vsebita vseh 32 bitov registra d1 se prenese v register d2.  
**move.b d3,d7** Vsebita spodnjih 8 bitov registra d3 se prenese v spodnjih 8 bitov registra d7. Pri tem se vsebita zgornjih 24 bitov registra d7 ne spremeni

- **Neposredno naslovno** (address register direct). Operand je v naslovnem registru. Sintaksa zbirnika je An, kjer n pomeni številko naslovnega registra. Kadar je cilj naslovni register, moramo v nekaterih zbirnikih pisati ukaz **move** kot **movea**. Primeri:

**move.l a1,d0** Vsebita vseh 32 bitov registra a1 se prenese v register d0.  
**movea.w d3,a2** Vsebita spodnjih 16 bitov registra d3 se predznačeno razširjeno prenese v register a2. Register a0 postane kazalec sklada, sp je v resnici register a7 (glj sliko 1). Zbirnikom je v glavnem vseeno, ali pisemo sp ali a7.

- **Registrsko posredno** (address register indirect). Naslov operanda je v naslovnem registru. Če imamo npr. na lokaciji 1000 v pomnilniku številko 200 in je vsebita registra a2 1000, potem je operand 200. Sintaksa zbirnika je (An). Primeri:

**move.l(a2),d1** Vsebita, ki jo naslavlja register a2, se prenese v register d1 (v tem primeru 200).  
**move.b d3,(a2)** Spodnjih 8 bitov vsebitine registra d3 se prenese na naslov, ki ga naslavlja register a2.

- **Registrsko posredno s poznejšim povečanjem** (address register indirect with postincrement). Naslov operanda je v naslovnem registru. Po uporabi operanda se vsebita naslovnega registra poveča za 1, 2 ali 4, odvisno od dolžine operanda. Če je npr. beseda na lokaciji 1000 v pomnilniku številko 200 in je vsebita registra An 1000, dolžina operanda pa je beseda, potem je operand 200. Takoj zatem se registru An prišteje 2. Če je naslovni register kazalec sklada (USP ali SSP) in je dolžina operanda byte, se naslov poveča za 2 in ne za 1. Besede in dolge besede je namreč mogoče brati le s sodih naslovov. Sintaksa zbirnika je (An)+. Primeri:

**movea.l(a1)+,a2** Vsebita naslova, na katerega kaže register a1, se prenese v register a2. Takoj zatem se vsebita registra a1 poveča za 4. Operacija je dolga 32 bitov. Spodnjih 16 bitov registra d7 se prenese na naslov, ki ga naslavlja register a4. Vsebita tega registra se poveča za 2. Osem bitov iz sklada se prenese v register d4. Nato se kazalcu sklada prišteje 2 in ne 1.

- **Registrsko posredno s predhodnim zmanjšanjem** (address register indirect with predecrement). Naslov operanda je v naslovnem registru. Pred uporabo operanda se vsebita naslovnega registra zmanjša za 1, 2 ali 4, odvisno od dolžine operanda. Naj bo dolžina operanda npr. 1 (byte) Vsebita registra 999 v pomnilniku je 12, vsebita registra An pa 1000. Najprej se bo register An zmanjšal za 1 in s tega naslova bomo dobili operand 12. Vsebita registra An bo 999. Če je naslovni register kazalec sklada (USP ali SSP) in je dolžina operanda byte, potem je naslov zmanjšal za 2 in ne za 1 (ker je branje besed in dolgi besed možno le s sodih naslovov). Sintaksa zbirnika je -(An). Primeri:

**move.l a3,-(a7)** Vsebitno registra a3 prenesemo v sklad. Pred prenosom se kazalec sklada zmanjša za 4.

**move.l(a3)+,-(a3)**

Ta ukaz nima nobenega učinka. Uporaben je za zapoznavitev.

Posredno naslavljanje naslovnih registrov s predhodnim zmanjšanjem in poznejšim povečanjem je zelo uporabno za uvajanje vrst **lifo** in **fifo**. Pri osebnitnih procesorjih imata tako vlogo ukaz **push** in **pop**.

— **Registrsko posredno z odmikom** (address register indirect with displacement). Naslov operanda dobimo tako, da vsebini naslovnega registra pristoječo 16-bitni nadomestek. Če je vsebina naslovnega registra enaka 1000 in če je nadomestek 100, je operand na naslovu 1100. Sintaksa zbirnika je d1(An), kjer je d16 šestnajstbitni nadomestek. Primeri:

**move.l\$100(a2),d2**

V register d2 se naloži vsebina naslova, ki ga naslavlja vsota registra a2 in nadomestka. Š pri nadomestku pomeni šestnajstško število Besedo, ki jo naslavlja register a3, prenesemo za 2 byta naprej. Dolgo besedo, ki je druga po vrsti v skladu, prenesemo v register a1.

**move.w(a3),2(a3)**

**move.l4(a7),a1**

— **Registrsko posredno z indeksom** (address register indirect with index). Naslov operanda dobimo tako, da vsebini naslovnega registra pristoječo 8-bitni nadomestek in vsebino indeksnega registra. Indeksni register je lahko podatkovni ali naslovni, njegova dolžina pa je 16 ali 32 bitov. Naj bo vsebina An enaka 1000, vsebina Ri (indeksni register) 200 in 8-bitni nadomestek 100. Naslov operanda je vsota teh vrednosti, 1300. Sintaksa zbirnika je d<sub>n</sub>(An, Rn.w) ali d<sub>n</sub>(An, Rn.l) Primeri:

**move.b d1,0(a1,d3.w)**

Spodnjih 8 bitov registra d1 se prenese na naslov, ki je vsota 8-bitnega nadomestka, registra a1 in registra d3, predznačeno razširjenega na 32 bitov Besedo, ki jo naslavlja vsota registra a7, a1 in nadomestka, prenesemo v register d5

**move.w4(a7,a1,l),d5**

— **Kratki absolutni naslov** (absolute short address). Operand je v tem modusu vsebina podanega naslova. Pred uporabo je 16-bitni naslov predznačeno razširjen na 32 bitov Sintaksa zbirnika je xxx.w, kjer je xxx 16-bitni naslov. Primeri:

**move.w\$1A2C.w,-(a7)**

Besedo na naslovu \$1A2C shranimo v sklad Byte na naslovu 4096 prenesemo na naslov \$192. Večini zbirnikov je vseeno, ali pišemo xxx.w ali samo xxx.

**move.b\$1000,\$2000**

**move.l\$0004,a1**

Dolgo besedo na naslovu 4 prenesemo v register a1

— **Dolgi absolutni naslov** (absolute long address). Operand je vsebina podanega 32-bitnega naslova. Pri kratkem absolutnem naslavljanju je naslov predstavljen z besedo, pri dolgem pa z dolgo besedo. Zato je pri kratkem absolutnem načinu doseg le 64 K. Zbirniki se avtomatsko odločijo za ta način, kadar je to mogoče. Sintaksa zbirnika je xxx.l, kjer je xxx 32-bitni naslov. Primeri:

**move.l\$28000,l,d0**

Dolgo besedo na naslovu \$28000 prenesemo v register d0.

**move.b\$12345,d3**

Byte na naslovu \$12345 prenesemo v register d3. Prav tako kot pri kratkem absolutnem načinu je vseeno, ali pišemo xxx.l ali xxx.

— **Programski števec z odmikom** (program counter with displacement). Naslov operanda dobimo tako, da programskemu števcu pristoječo 16-bitni, predznačeno razširjen nadomestek. Sintaksa zbirnika je LABELA(PC), kjer je LABELA 16-bitni nadomestek. Primeri:

**move.wznak(pc),d1**

V register d1 naloži kodo ukaza nop

;

;

;

**znak nop**

— **Programski števec z indeksom** (program counter with index). Naslov operanda je vsota programskega števca, vsebine indeksnega registra in 8-bitnega nadomestka. Indeksni register je lahko podatkovni ali naslovni, njegova dolžina pa je 16 ali 32 bitov. Sintaksa zbirnika je LABELA(PC,Rn.W) ali LABELA(PC,Rn.W), kjer je LABELA 8-bitni nadomestek, Rn pa indeksni register. Primer:

**move.b\$labela(pc,a1,l),d2**

Naslov byta, ki gre v register d1, je vsota programskega števca, 8-bitnega nadomestka in naslovnega registra.

**labela dc.b\$12,34,0,12,23...**

— **Neposredni podatki** (immediate data). V tem načinu je operand podan. Njegova dolžina je 8, 16 ali 32 bitov. Sintaksa zbirnika je #xxxx, kjer je xxxx operand. Primeri:

**move.w#\$1234,d2**

Prenesi besedo \$1234 v register d2.

**move.b#\$10,d0**  
**move.l#\$12345678,d1**

Prenesi byte \$10 v register d0.  
Prenesi dolgo besedo \$12345678 v register d1.

Kadar uporabljamo tako naslavljanje, moramo posebej paziti, da napišemo # pred številko. V nasprotnem primeru dobimo kratki ali dolgi absolutni način.

Pri nekaterih ukazih, kjer je operand podan kot dejanski naslov, niso dovoljeni vsi načini naslavljanja. Ogledimo si nekaj primerov.

**move.w d1,#\$100**

Prepiši spodnjih 16 bitov registra d1 v konstanto.

**jmp #ABCD**

Skoči na konstanto.

**jsr d1**

Pokliči podprogram, ki je v registru d1.

Kot vidite, so ukazi popolnoma nesmiselni ali nezaželeni. Zaradi takih ukazov razvrščamo načine naslavljanja na štiri kategorije. To so:

- podatkovna
  - pomnilniška
  - nadzorna
  - spremenljiva.
- Podatkovni operandi obsegajo vse razen vsebine naslovnih registrov, medtem ko so pomnilniški operandi vse tisto, kar ni shranjeno v registrih. Operand je spremenljiv, če lahko vanj vpišujemo, in nadzorni, če lahko z njim določimo cilj skoka. Kategorije, v katere sodijo posamezni načini, so povzete v tabeli 1.

NAČIN	podatkovni	pomnilniški	nadzorni	spremenljivi
Dn	*	*	*	*
An	*	*	*	*
rel, na P5	*	*	*	*
RP5+index	*	*	*	*
(An)	*	*	*	*
dis(An)	*	*	*	*
d <sub>n</sub> (An)	*	*	*	*
-(An)	*	*	*	*
(An)+	*	*	*	*
absolutni	*	*	*	*
spodetek	*	*	*	*

Tabela 1 Kategorije dejanskih naslovov

## Nabor instrukcij

Omenili smo, da imamo na voljo naslednje zastavice: N – negativno (negative), Z – ničla (zero), V – prekoračitev (overflow), C – prenos (carry) in X – razširjenje (extend). Prvi štirje biti registra stanj (CCR) so prave zastavice in kažejo pogoj rezultatov operacij. Zastavica X je operand za računanje in kažejo natančnostjo. V bistvu je to zastavica prenosa. Ločeni sta le zato, da olajšata programski model. Zastavica X je operand za računanje z dvojno natančnostjo. Zastavica negativnosti se postavi na 1, kadar je najvišji bit rezultat 1, v nasprotnem primeru pa na logično 0. Ničelna zastavica je postavljena na 1 samo, kadar je rezultat 0. Pri aritmetični prekoračitvi se postavi na 1 zastavica prekoračitve. To pomeni, da rezultata ne more predstaviti z velikostjo operanda. Zastavica prenosa je na 1, kadar se prenese najvišji bit pri sestevanju ali odštevanju. Aktivna zastavica razširjenja se postavi prav tako kot zastavica prenosa.

Na sliki 5 je seznam imen za pogoje, ki jih uporabljamo pogojni skoki in druge instrukcije. Test, povezan z vsakim pogojem, je logična formula. Ta preračuna trenutno vrednost zastavic. Če je rezultat formule 1, je pogoj izpolnjen. Pogoj T je npr. vedno izpolnjen, medtem ko je EQ izpolnjen le, kadar je ničelna zastavica 1.

Pri 8-bitnih procesorjih so v zbirnikih ukazi, ki ne sodijo v nabor instrukcij

ORG <naslov> prevedi program na določen naslov  
 EQU nn določi vrednost labeli  
 DEFB n1,n2... vključi podatke dolžine byta v program  
 DEFW n1,n2... vključi podatke dolžine besede v program  
 DEFS nn izpusti nn bytov  
 DEFM 'text' vključi besedilo v program  
 END konec programa

Poleg teh je nekaj ukazov za pogojno prevajanje, obliko izpisa ipd. Za M 68000 se po dogovoru uporabljajo ukazi:  
 ORG <naslov> prevedi program na določen naslov  
 RORG <naslov> prevedi program na prestavljen naslov

EQU nn priredi vrednost nn labeli  
 DC.B n1,n2 vključi podatke dolžine byta v program  
 DC.W n1,n2 vključi podatke dolžine besede v program  
 DC.L n1,n2 vključi podatke dolžine dolge besede v program  
 DS.B nn izpusti nn bytov  
 DS.W nn izpusti nn besed  
 DS.L nn izpusti nn dolgih besed  
 END konec programa

Če želimo v program vključiti besedilo, enostavno napišemo.  
**text dc.b 'Nekaj besedila'**

To je včasih nevarno. Če je npr vsota črk v besedilu iha, bi se moral program nadaljevati z lihega naslova, tega pa M 68000 ne dopušta. Zato namesto DC.B napišemo DC.W ali DC.L. Zbirnik nam bo na koncu dodal eno ali več ničel, da bomo spet prišli na sod naslov

mnemank	pojej	test
T	true	I
F	false	0
H	high	E=Z
LS	low or same	C=Z
CC (HS)	carry clear	C
CS (LD)	carry set	Z
NE	not equal	C
EQ	equal	7
VC	overflow clear	V
VS	overflow set	V
PL	plus	R
MI	minus	R
GE	greater or equal	M*V+R=V
LT	less than	M*V+R=V
GT	greater than	M*V+R=V+Z
LE	less or equal	Z=M*V+R=V

Slika 5. Pogojni testi

### Operacije za prenos podatkov

Osnovni ukaz za prenos podatkov je **move**. Dolžina operacije je byte, beseda ali dolga beseda. Za vse ukaze velja: kadar je operand naslovni register, je lahko dolžina operacije samo beseda ali dolga beseda. Med prenosom se podatek testira in zastavice se postavijo ustrezno. Ker smo si **move** že ogledali pri načinih naslavljanja, ne bomo naštevali primerov

Naslednji ukaz je **movem**. Z njim je mogoče naenkrat prenesti vsebino vseh ali samo nekaterih registrov. Dolžina operacije je lahko samo beseda ali dolga beseda. Register, ki jih bomo prenesli, izbiramo med a0-a7 in d0-d7. Ukaz nima vpliva na zastavice. Primeri:

**movem.l d0-d7/a0-a6,-a7** Prenesi vsebino vseh registrov v sklad  
**movem.l (a7)+,d0-d7/a0-a6** Napolni vse registre z vsebino sklada  
**movem.w d1/d3/a4,\$1000** Prenesi spodnjih 16 bitov vsebine registrov d1, d3 in a4 na naslov \$1000

Z ukazom **movep** prenašamo podatke med periferno enoto in procesorjem. Osnovni obliki ukaza sta

**movep(dolžina) Dx,d(Ay)**  
**movep(dolžina) d(Ay),Dx**

Ukazu moramo določiti še dolžino, ki je lahko beseda ali dolga beseda. X in y sta številki podatkovnega in naslovnega registra. Ukaz ne vpliva na zastavice. Primeri:

**movep.w d1,(a2)** Prenesi spodnjih 16 bitov na naslov, ki ga kaže register a2  
**movep.l \$10(a2),d4** Prenesi podatke, ki se začnejo na naslovu a2+\$10, v register d4  
**movep.w 1(a3),d0** Prenesi podatke, ki se začnejo na naslovu a3+1, v spodnjih 16 bitov registra d0

Ukaz **moveq** je namenjen hitremu prenosu podatka v podatkovni register. Podatek je 8-bitni, vendar se pred uporabo predznačeno razširi. Če je npr. 7. bit podatka 1, se postavijo na 1 vsi biti od 8 do 31. Dolžine prenosa ne določamo, ker je vedno byte. Osnovna oblika ukaza je

**moveq#<data>,Dn**

Operacija vpliva na zastavici N in Z. Zastavici V in C sta vedno 0, na X pa ukaz nima vpliva. Primeri:

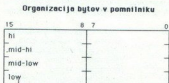
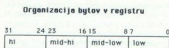
**moveq#\$10,d2** V register d2 se prenese število 16. Biti od 8 do 31 se postavijo na logično 0.

**moveq#-1,d0** V register d0 se prenese število 1. Vsi biti registra d0 so 1.  
**moveq#0,d5** Pobrši vse bite registra d5.

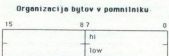
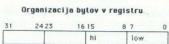
Z ukazom **exg** zamenjujemo vsebino med podatkovnimi, naslovnimi ali mešanimi registri. Osnovna oblika ukaza je

**exgRx,Ry**

Prenos dolge besede na lihi naslov ali z njega



Prenos besede ne sodi naslov ali z njega



Slika 6. Prenos podatkov z ukazom **axg**

Rx in Ry sta katerakoli podatkovna ali naslovna registra. Operacija je vedno dolga 32 bitov. Ukaz ne vpliva na zastavice. Primeri:

**exg d1,d3** Zamenjaj vsebini registrov d1 in d3.  
**exg a3,a4** Zamenjaj vsebini registrov a3 in a4.  
**exg d4,a2** Zamenjaj vsebini registrov d4 in a2. Kadar zamenjujemo naslovni in podatkovni register, je slednji na levi strani.  
**exg a3,d1** Ta primer ni pravilen. Velja bi, če bi bil register d1 na levi, register a3 pa na desni. Pri nekaterih zbirnikih lahko pišemo tudi tako.

Ukaz **lea** napolni naslovni register z dejanskim naslovom. Operacija je vedno dolga 32 bitov. Osnovna oblika ukaza je

**lea <dejanski naslov>,An**

Prednost tega ukaza pred **move** je v tem, da učinkuje tudi pri relativnem naslavljanju. V naslovni register se ne nalozijo vsebina naslova, ampak sam naslov. Ukaz lahko uporabimo tudi za enostavno seštevanje. Zastavice se ne spremenijo. Primeri:

**labela lea labela(pc),a2** V register a2 se bo vpisal naslov ukaza.  
**lea \$20(a2),a3** V register a3 se bo vpisala vsota števila \$20 in vsebine registra a2.  
**lea \$1200,a5** V register a5 se bo vpisal naslov \$1200.

Zelo podoben ukaz je **pea**. Uporablja se tako kot **lea**, le da se rezultat shrani v sklad in ne v naslovni register. Operacija je dolga 32 bitov in ne vpliva na zastavice. Primeri:

**labela pea labela 6pc 7rts** Naprej se bo naslov ukaza **pea** zapisal v sklad. Nato se bo izvedel ukaz **rts**, ki pobere iz sklada naslov in skocij nanj. To pomeni, da se bo program vrtil v neskončni zanki.

**pea (a4)** V sklad se bo shranila vrednost iz registra a4.  
**pea \$10(a2,d3,w)** V sklad se bo shranila vsota števila \$10, vrednosti iz registra a2 in predznačeno razširjene vsebine spodnjih 16 bitov registra d3.

Ukaza **link** in **unlink** lahko uporabimo za shranjevanje povezanih listov lokalnih podatkov in parametrov v skladu. Osnovni obliki ukazov sta



## link An, 12<zamenjava>

link An

Pri ukazu link se vrednost iz statusnega registra shrani v sklad. Potem se naslovni register napolni z vrednostjo kazalca sklada. Nazadnje se vrednosti kazalca sklada prišteje nadomestek. Simbolično je to videti takole.

An → -(sp); sp ⇒ An; sp+nadomestek ⇒ sp

Kadar je nadomestek negativen, pomeni, da želimo v skladu rezervirati prostor. Ukaz ne vpliva na zastavice. **Primer:**

linka1, #-8

Vrednost iz naslovnega registra 1 se bo shrnila v sklad. Potem se bo vsebina kazalca sklada prenesla v register a1. Nazadnje se bo v skladu naredil prostor za dve dolgi besedi. Vsebinska naslovnega registra a2 se shrani v sklad. Nato se vsebina kazalca sklada prenese v register a2. Nazadnje se kazalec sklada zveča za 4. Skoraj enak učinek ima ukaz **movea.l a7,a2**.

link a2,#4

Nasprotno kot link dela ukaz unik. Najprej vsebino naslovnega registra naloži v kazalec sklada. Potem iz njega prebere naslov in ga shrani v naslovni register. Simbolično sta ti operaciji takile.

An ⇒ sp; (sp)+ ⇒ An

Zastavice so po operaciji nespremenjene. **Primer:**

unik a1

V kazalec sklada se vpiše vsebina registra a1. Nato se register a1 napolni s številom v skladu.

V tej skupini je še en ukaz namenjen zamenjavi zgornjih 16 in spodnjih 16 bitov v podatkovnem registru. To je **swap**. **Primer:**

swap d1

Kadar je bit 31 postavljen na 1, se prižge zastavica N. Prižge se tudi zastavica Z, kadar je vsebina registra 0. Zastavica V in C sta vedno 0, na X pa ukaz nima vpliva.

	Izvir				Cilj			
	move	lea	pea	movem <sup>1</sup>	move	lea	pea	movem <sup>2</sup>
Dn	*	*	*	*	*	*	*	*
(An)	*	*	*	*	*	*	*	*
(An)+	*	*	*	*	*	*	*	*
-(An)	*	*	*	*	*	*	*	*
d(An)	*	*	*	*	*	*	*	*
d(An,Xi)	*	*	*	*	*	*	*	*
xxxx.w	*	*	*	*	*	*	*	*
xxxx.l	*	*	*	*	*	*	*	*
d(pc)	*	*	*	*	*	*	*	*
d(pc,Xi)	*	*	*	*	*	*	*	*
*xxxx	*	*	*	*	*	*	*	*

1 – samo, kadar zapisujemo pomnilnik v registre

2 – samo kadar vpisujemo registre v pomnilnik

Pri ukazu **moveq** je izvir vedno podatek, cilj pa podatkovni register. Ukaz **exg** dela samo s podatkovnim in naslovnim neposrednim načinom. **Link** in **unik** operirata samo z naslovnim registrom in s kazalcem sklada. Pri prenosu podatkov z **movep** sta edini možni obliki ukaza **movep Dn,d(An)** in **movep d(An),Dn**.

## Aritmetični ukazi

Prvi med najpomembnejšimi aritmetičnimi operacijami je seštevanje. Za to skrbi ukaz **add**. Osnovna oblika je

**add,(dolžina) <dejanski naslov>,Dn**

oziroma

**add,(dolžina) Dn,<dejanski naslov>**

Izpeljanki tega ukaza sta

**addi,(dolžina) \* <podatek>,<dejanski naslov>**

**addq,(dolžina) \* <podatek>,<dejanski naslov>**

**Addi** je namenjen prištevanju podanega podatka operandu. Prav tako nalogo ima **addq**, le da je tu podatek številno 1 in 7. Vsi ti ukazi vplivajo na vse zastavice. Ker je izpeljank veliko, si oglejmo nekaj primerov več

**add.w (a3),d4**

Prišteje registru d4 vsebino naslova, na katerega kaže register a3. Dolžina operacije je beseda.

**add.l d1,8(a7)**

Prišteje tretji dolgi besedi v skladu vsebino registra d1. Kadar uporabljamo osnovno obliko

**addq.b #4,d3**

ukaza **add**, mora biti ali cilj ali izvir podatkovni register.

Spodnjih 8 bitov registra d3 prišteje 4. Pri seštevanju z **addq** za cilj ne smemo uporabiti načina naslavljanja (d(pc), d(pc,Xi) in \*xxxx. Pri osnovnem načinu seštevanja so dovoljeni vsi načini naslavljanja. Prištejo dolgi besedi na naslovu \$1000 ena.

**addq.l #1,\$1000**

**adda.l nekaj(pc),a2**  
**dc.l \$28000,\$12345**

nekaj

**addi.w +1234,d3**

Prišteje registru a2 številu \$28000. **Adda** pišemo zato, ker je cilj naslovni register.

Prištejemo številu 1234 registru d3. V tem modusu niso dovoljeni načini naslavljanja An, d(pc), d(pc,Xi) in \*xxxx. Operacija je dolga 16 bitov.

Za seštevanje z večkratno natančnostjo uporabljamo zastavico X. To pomeni, da ne seštejemo samo izvira in cilja, ampak tudi zastavico razširitev. Ukaz omogoča le dva načina naslavljanja. To sta

**addx,(dolžina) Dy,Dx**  
**addx,(dolžina) -(Ay),-(Ax)**

Primeri.

**addx.d1,d2**

Sešteje registra d1 in d2. Nato rezultatu prišteje zastavico x in shrani vsoto v register d2.

**addx.w -(a2),-(a3)**

Zmanjšaj registra a2 in a3 za 2. Nato sešteje besedi na naslovih, ki jih kaže ta registra, in prišteje vsebino zastavice x. Rezultat shrani na naslov (a3).

Naslednji zelo pomemben ukaz je seveda odštevanje: **sub**. Tudi zanj so na voljo vse zgoraj opisane izpeljanke:

**sub.(dolžina) <dejanski naslov>,Dn**

**sub.(dolžina) Dn,<dejanski naslov>**

**subi.(dolžina) \*podatek,<dejanski naslov>**

**subq.(dolžina) \*podatek,<dejanski naslov>**

**subx.(dolžina) Dy,Dx**

**subx.(dolžina) -(Ay),-(Ax)**

Kot vidimo, so ukazi ravno taki kot za seštevanje. Zato se ukazu **sub** ne bomo posvetili. Vsi načini naslavljanja in vplivi na zastavice so enaki kot pri seštevanju. Poglejmo si nekaj primerov:

**subi.b #10,d2**

Odstejemo spodnjim 8 bitom registra d2 10.

**suba.l d1,a3**

Odstejemo registru a3 vsebino registra d1.

**suba.l a1,a1**

Zbrši vsebino registra a1. Zastavica Z se postavi na 1, na X pa ukaz ne vpliva. Prepovedani načini naslavljanja so An, d(pc), d(pc,Xi) in \*xxxx. Dolžina operacije je lahko byte, beseda ali dolga beseda. **Primeri:**

**clr.l d1**

Zbrši register d1.

**clr.l (a7)+**

Zbrši dolgo besedo v skladu in ga poveča za 4

**clr.b 10(a3)**

Zbrši byte na naslovu 10+a3.

Podatke lahko med seboj primerjamo s **cmp**. Ta ukaz uporabljamo na tri različne načine:

**cmp.(dolžina) <dejanski naslov>,Dn**

**cmp.l (dolžina) \* <podatek>,<dejanski naslov>**

**cmpm.(dolžina) (Ay)+,(Ax)+**

**Cmpi** je namenjen takojšnjemu primerjanju podatka z operandom. **Cmpm** primerja pomnilniške lokacije. Izvir in cilj ostane po operaciji nespremenjena. Vpliv se pozna le na zastavicih, ki se postavijo tako, kot če bi operanda med seboj odšteli. Pri ukazu **cmp** so dovoljeni vsi načini naslavljanja. Pri **cmpi** pa so pri cilju prepovedani naslednji načini: An, d(pc), d(pc,Xi) in \*xxxx. **Primeri:**

**cmpi.b #10,d2**

Primerja 10 in vsebino spodnjih 8 bitov registra d2. Če je d2=10, se postavi zastavica Z na 1. Če je d2=10, se postavi X na 1 in C na 0. Na X ukaz ne vpliva.

**cmpm.l (a1)+,(a2)+**

Primerja dolgi besedi na naslovih (a1) in (a2) ter povečaj registra a1 na a2 za 4

**cmpa.w d7,a7**

Primerja register d7 s kazalcem sklada.

Za primerjanje z nič imamo na voljo ukaz **test**. Ta ne shrani nobenega rezultata, temveč ustrezno postavi zastavice. Zastavici V in C sta vedno 0, na X pa ukaz ne vpliva. Dolžina testa je lahko byte, beseda ali dolga beseda. Načini naslavljanja An, d(pc), d(pc,Xi) in \*xxxx niso dovoljeni. ▶

Primeri:

**lst.b d1** Poglej, ali je register d1 enak nič  
**lst.l (a3)+** Primerjaj vsebino naslova, ki ga kaže register a3, z nič. Nato ga povečaj za 4.

Podoben ukaz kot **lst** je **tas**. Dolžina primerjanja je byte. Ukaz primerja vrednost operanda z 0 in ustrezno postavi zastavici N in Z. Nato se vijiši bit operanda postavi na 1. Zastavici V in C sta vedno 0, na X pa ukaz ne vpliva. Ta operacija je nedeljiva (ne uporablja cikla read-modify-write) in je ustreza za sinhronizacijo med več procesorji. Prepovedani so listi načini naslavljanja kot pri ukazih **lst**. Primeri:

**tas d2** Poglej, ali je register d2 enak nič. Ustrezno postavi zastavice in nato postavi 7 bit registra d2 na ena

**tas 13(a3)** Poglej, ali je byte na naslovu 13+a3 enak nič. Ustrezno postavi zastavice. Nato še 7. bit naslovljenega byta postavi na ena.

Naslednji ukaz je za predznačeno razširjanje podatkovnega registra. Njegov mnemonik je **ext**. Kadar je podana dolžina operacije beseda, razširimo operand z 8 na 16 bitov. To pomeni, da 7 bit podatkovnega registra kopiramo na bite od 8 do 15. Kadar je dolžina dolga beseda, razširimo operand s 16 na 32 bitov. Kopiramo 15. bit na bite od 16 do 31. Zastavici V in C sta po operaciji nič, zastavica X se ne spremeni, zastavici N in Z pa se postavi odvisno od rezultata. Primeri:

**move.b #57c,d3** Napolni register d3 s 57c.  
**ext.w d3** Predznačeno ga razširi na 16 bitov. V registru d3 dobimo 57c.  
**move.b #85d,d2** Napolni register d2 s 85d.  
**ext.w d2** Predznačeno razširi register d2. V njem dobimo število \$FF85

Število negiramo z ukazom **neg**. Operacija poteka tako, da se operand odšteje od števila 0. Vse zastavice se ustrezno postavijo. Načini naslavljanja An, d(pc), d(pc,Xi) in \*xxxx so spet prepovedani. Dolžina operacije je lahko byte, beseda ali dolga beseda. Primeri:

**neg.b d7** Negiraj spodnjih 8 bitov registra d7.  
**neg.w 54321** Negiraj besedo na naslovu 54321.

Ukaz **negx** je popolnoma enak ukazu **neg**, le da se operandu odšteje še vsebina zastavice X.

Verjetno boste najbolj veselili ukazov **mulu**, **mulv**, **divu** in **divs**, s katerimi nepredznačeno in predznačeno množimo in delimo. Vsi operirajo z operandi dolžine besede in dajo rezultat dolžine dolge besede. Pri deljenju dobimo rezultat v spodnjih 16 bitih in ostanek v zgornjih 16 bitih podatkovnega registra. Cilj pri vseh teh ukazih je namreč lahko le podatkovni register. Pri izviritu uporabimo katerikoli način naslavljanja razen načina An. Pri množenju sta zastavici C in V vedno nič, na zastavico X pa ukaz nima vpliva. Pri deljenju je zastavica C vedno nič. Na X ni vpliva. Zastavica N je 1, kadar je tudi najvišji bit kvocienta 1. Kadar je prekoračitev, se prižge zastavica V. Zastavica Z se postavi na ena, kadar je kvocient nič. Pri prekoračitvi pa sta zastavici N in Z nedefinirani.

Primeri:

**divs d1,d2** Predznačeno deli vsebino registra d2 z vsebino d1. Kvocient dobimo v spodnjih, ostanek pa v zgornjih 16 bitih registra d1.  
**mulu 2(a3),d7** Nepredznačeno množi besedo na naslovu 2+a3 z vsebino registra d7. Rezultat shrani v d7.  
**divu #123,d0** Nepredznačeno deli vsebino registra d0 s številom 123. Kvocient dobimo v spodnjih, ostanek pa v zgornjih 16 bitih registra d0.  
**mulv 8(a2,d3),d4** Predznačeno množi vsebino registra d4 z besedo na naslovu 8+a2+d3. Rezultat dobimo v registru d4.

Kadar želimo množiti ali deliti s konstanto, ki je potenca števila 2, se nam bolj izplača uporabiti ukaze za aritmetični pomik levo ali desno. Maksimalni čas za deljenje je namreč lahko dolg tudi do 158 ciklov procesorja, za množenje pa 70 ciklov. Če to primerjamo s povprečnim časom izvajanja instrukcij pri osembitnem procesorju Z 80, ki je 7-8 ciklov, nam bo kmalu jasno, zakaj.

Sedaj ko že znamo premetavati podatke in nekaj malega z njimi delati, bi bilo lepo, če bi lahko tudi spreminjali potek programa. S takim znanjem bi že lahko pisali prve programe. Naslednja skupina ukazov je namenjena prav temu

## Operacije za kontrolo programa

V to skupino sodijo ukazi, ki uporabljajo vrsto pogojnih in brezpogojnih instrukcij za skoke, skoke v podprograme in vrnitve iz njih. Prvi ukaz za brezpogojni skok je **bra**. Mnemonik izhaja iz besede branch, ki pomeni veja. Iz tega sledi, da je ukaz namenjen relativnim skokom. Relativni skoki so tukaj lahko dolgi do 32 K. Tako lahko vse program napisamo z relativnimi skoki in ga izvajamo s kateregakoli naslova. Za relativni naslov uporabljamo byte (oddaljenost je lahko le do 128 bytov) ali besedo. Osnovna oblika ukaza je

**bra. (dolžina) <labela>**

Primeri:

**bra.s naprej** Skoči naprej do 127 bytov daleč.  
 :  
 :  
 :  
**naprej** **nop** **moveq #10,d0** **Napolni d0 z 10.**  
**naslov** **add.l #100,d0** **Prisrtejo d0 100.**  
 :  
 :  
 :  
**bra naslov** Skoči nazaj do 32 K daleč. Kadar pri ukazu izpustimo dolžino skoka (s), je skok dolg.

Z ukazom **jump** skočimo na natančno določeno naslov. Osnovna oblika je

**jmp <dejanski naslov>**

Po ukazo se izvajanje programa nadaljuje na dejanskem naslovu. Dovoljeni so naslednji načini: (An), d(An), d(An,Xi), xxx.w, xxx.l, d(pc) in d(pc,Xi). Primeri:

**org \$1000** Program se začne na naslovu d1000.  
 :  
 :  
 :  
**jmp \$1000** Skoči na naslov \$1000.  
**jmp (a3)** Skoči na naslov, ki ga kaže register a3.  
**jmp 10(a2,a3.1)** Skoči na naslov, ki ga kaže vsota a2, a3 in števila 10.

Naslednja ukaza sta namenjena vejitvi in skoku v podprogram. Oba sta brezpogojna. Splošna oblika ukazov je

**bsr. (dolžina) <labela>**

**jsr <dejanski naslov>**

Pri ukazih **jsr** so dovoljeni tisti načini naslavljanja kot pri ukazih **jmp**. Ob izvajanju obeh ukazov se naprej shrani v sklad trenutna vrednost programskega števca. Nato program nadaljuje delo tam, kjer kaže labela oziroma dejanski naslov. Iz podprograma se lahko vrnejo z ukazom **rts**. Ta pobere iz sklada dolgo besedo, ki je naslov naslednjega ukaza za kilec v podprogram, in izvaja ukaze naprej. Kako to gre, nam pokaže programček.

Program naj pokliče podprogram, ki testira pomnilnik. Uspešnost testa naj pokaže v registru d0. Če je ta -1, test ni bil uspešen. Če je 0, je test uspel. Uporabili bomo tudi pogojne skoke, ki jih bomo razložili pozneje.

**org 0** Program je na relativnih naslovih  
**equ \$1000** Začetek področja, ki ga bomo testirali  
**zacetek** **equ \$3000** Konec testnega področja  
**konec** **equ \$5000** Bitni vzorec, s katerim bomo izvedli test  
**vzorec** **equ \$6000** Bitni vzorec, s katerim bomo izvedli test  
**start** **moveq #0,d0** DO kaže, da je test uspel  
**test** **bsr.s test** Skoči na podprogram  
**test** **ts1.d0** Poglej, ali je d0 nič  
**test** **bec.s vredu** Če je, izpiši 'Test uspel'  
**test** **lea text1,a0** Kazalec na prvo besedilo  
**test** **jmp izpis** Skoči na izpis  
**test** **lea text2,a0** Kazalec na drugo besedilo  
**test** **jmp izpis** Skoči na izpis  
**test** **Podprogram, ki testira pomnilnik**  
**test** **move.l a0,-(a7)** Shrani register a0  
**test** **movea.l #zacetek,a0** To je začetni naslov testa  
**test** **move.b #vzorec,(a0)+** Polnimo pomnilnik z vzorcem  
**test** **cmpa.l #konec,a0** Ali smo že prišli do konca?  
**test** **bne.s zanka1** Če ne, ponovi  
**test** **bne.s zanka1** Če ne, ponovi  
**test** **Sedaj smo napolnili pomnilnik s testnim vzorcem. Preostane nam le še preskus, ali se je kakšen bit morda spremenil**  
**test** **movea.l #zacetek,a0** Ponovno nastavi kazalec  
**test** **Se je pomnilnik spremenil?**

**zanka2 cmpi.b** **Sporoči napako**  
**vzorec,(a0)+** **Ali smo prišli do konca?**  
**zanka2 cmpi.b** **Če ne, ponovi**  
**vzorec,(a0)+** **Vrremo se v glavni prg.**  
**zanka2 cmpi.b** **bra.s napaka**  
**vzorec,(a0)+** **cmpa.l #konec,a0**  
**zanka2 cmpi.b** **bne.s zanka 2**  
**vzorec,(a0)+** **bra.s vredu2**

**napaka v redu2** subq.l #1,d0  
movea.l (a7)+,a0  
rts  
Na koncu programa izpišemo rezultat testa.  
To naredimo tako, da postavimo kazalec a0 na naslov besede, ki ga želimo izpisati, in skočimo na rutino izpis.

**text1** dc.w 'Nastala je napaka!'  
**text2** dc.w 'Test uspel. Pomnilnik v redu.'

Ta primer ni ravno realen. Pri pravem testiranju vpišemo v pomnilnik bitni vzorec, nato pa pustimo pomnilnik nekaj ur pri miru. V tem času ne smemo z njim delati ničesar, ker se z vsakim branjem ali pisanjem tudi osveži.

V programu smo uporabili pogojne skoke. Sintaksa ukaza je **bcc <labela>**

Kateri pogoj želimo testirati, nam pove ec v ukazu. Pogoje, ki jih lahko uporabimo, vidimo na sliki 5. Tako npr. ukaz **bem** pomeni: Skoči naprej, kadar je zastavica Z prižgana. Če pogoj ni izpolnjen, se program izvaja normalno naprej od naslednjega ukaza.

Naslednji ukaz za vejitev je **dbcc**. Uporabljamo ga s pogojem ali brez njega. Osnovni obliki sta **dbcc Dn,<labela>** in **dbcc Dn,<labela>**

**Prva oblika je pogojna, druga brezpogojna. Ukaz je podoben bcc, le da se tu še prej odšteje od podatkovnega registra 1, kar je zelo ustrezno za zanke.** Primer:

**zanka** move.l #528000,a1  
move.w #51FFF,d0  
clr.l (a1)+  
dbra d0,zanka

Ta zanka bi zbrisala zaslon QL. Verjetno se sprašujete, zakaj potem potrebujemo poleg tega pogoje. Recimo, da imamo 80 bytov dolg vmesni pomnilnik. Uporabljamo ga za vnos stavkov s tipkovnice, ki se končajo s kodo ASCII za pomik v novo vrsto. Sedaj želimo napisati rutino, ki nam bo kot rezultat vrnila dolžino stavka. Če je ta daljši od 80 bytov, je izhodni podatek 80.

**cr** equ \$OD  
move.l #51000,a1  
move.w #79,d0  
**zanka** move.b (a1)+,d2  
cmp.b #cr,d2  
dbeq d0,zanka

**moveq #79,d1**  
sub.w d0,d1

Koda ASCII za line feed  
Začetek bufferja  
Števec do 80  
Preberi prvi byte  
Je že konec stavka?  
Ponavljaj, dokler ne prideš do konca bufferja oziroma do konca stavka.  
Dolžino dobimo tako, da od 79 odštejemo d0.  
Rezultat je v d1.

M68000 nam daje še en ukaz za obdelavo pogojev, **sc**. Osnovna oblika je **sc <dejanski naslov>**

Če je izpolnjen pogoj, nam ukaz postavi vse bite v bytu na dejanskem naslovu na 1, v nasprotnem primeru pa na 0.

**scs 3(a3)**  
**spl \$1000**  
**smi d1**

Kadar je zastavica C ena, postavi vse bite na naslovu a3+3 na ena.  
Kadar je zastavica N nič, postavi vse bite na naslovu \$1000 na ena.  
Kadar je zastavica N ena, postavi bile od 0 do 7 v registru d1 na ena.

Prepovedani so načini naslavljanja: An, d(pc), d(pc,Xi) in #xxxx. Ukazi, ki smo jih doslej opisali v tej skupini, nimajo vpliva na zastavice. Zadnji ukaz, ki je namenjen vrnitvam iz podprogramov, pa ga ima. Najprej se beseda iz sklada prenese v register stanj. Statusni del tega registra se ne spremeni, zastavice se spremenijo odvisno od vsebine besede v skladi. Od tod je ukaz identičen navadni vrnitvi iz podprograma. Mnemonik je rtr.

## Logične operacije

Ukaz **and** izvede logični "in" med izvirom in ciljem. Dolžina operacije je lahko byte, beseda ali dolga beseda. Vsebinsa naslovnega registra se ne more uporabiti kot operand. Zastavici V in C sta vedno nič, na X pa ukaz ne vpliva. Ustrezno se postavlja le zastavici N in Z. Splošni obliki ukaza sta

**and.<dolžina> <dejanski naslov>,Dn**  
**and.<dolžina> Dn,<dejanski naslov>**

Kadar je cilj dejanski naslov, niso dovoljeni načini naslavljanja: Dn, An, d(pc), d(pc,Xi) in #xxxx. Primeri:

**and.b (a1),d3** Izvedi logični "in" med vsebino naslova (a1) in registrom d3. Rezultat shrani v register d3. Dolžina operacije je byte

**and.l \$3000,d7** Izvedi logični "in" med vsebino naslova \$3000 in registrom d7.

**and.w d2,d1** Rezultat shrani v register d3. Dolžina operacije je dolga beseda. Izvedi logični "in" med spodnjimi 16 biti registrov d2 in d1. Rezultat shrani v register d1. Zgornji 16 bitov registra d1 se ne spremeni.

Ukaz **and** je tak kot navadni **and**, le da je tu izvir neposreden podatek.

**Primeri:**

**and.b #100,(a3)** Izvedi logični "in" med številom 100 in vsebino naslova, ki ga naslavlja register a3. Rezultat shrani na ta naslov.

**and.w #1234,d0** Izvedi logični "in" med številom 1234 in vsebino spodnjih 16 bitov registra d0. Rezultat shrani v register d0.

Načini naslavljanja in postavitve zastavic so enaki tudi pri ukazih **or**, **orl** in **eorl**. Or naredi med operandi logični ali, eor pa ekskluzivni ali. Primeri:

**or.l #20,10(a4)** Naredi logični ali med številom 20 in vsebino dolge besede na naslovu a4+10

**eor.w d1,d2** Izvedi ekskluzivni ali med spodnjimi 16 biti registrov d1 in d2. Rezultat shrani v register d2. Pri tem se vsebina zgornjih 16 bitov registra d2 ne spremeni. Izvedi logični ali med vsebino naslova, ki ga kaže register a2, in vsebino naslova \$1000. Povečaj vsebino registra a2 za 1.

**or.b (a2)+,\$1000** Negacija je zadnji ukaz iz te skupine. Dolžina operacije je lahko byte, beseda ali dolga beseda. Prepovedani načini naslavljanja so spet An, d(pc), d(pc,Xi) in #xxxx. Zastavice se postavijo tako kot pri drugih ukazih iz te skupine. Primeri:

**not.b d3** Naredi prvi komplement spodnjih osmih bitov registra d3. Negiraj besedo na naslovu 13+a3.

**not.w 13(a3)** Negiraj dolgo besedo na naslovu 3+a3+a2.

**not.l 3(a3,a2,l)**

## Operacije za premikanje in rotiranje

Aritmetično premaknemo bite operanda v zeleni smeri z ukazoma **asl** in **asr**. Dolžina operanda, ki ga premikamo, je lahko byte, beseda ali dolga beseda. Za koliko bitov bomo premikali, lahko določimo na dva načina:

- Velikost pomika je neposredni podatek. To število je lahko med 1 in 8.
- Velikost pomika je shranjena v podatkovnem registru in je poljubna

Lahko premikamo tudi vsebino pomnilnika, vendar samo za en bit naenkrat, dolžina pa je beseda. Pri **asl** premikamo operand v levo. Pri tem se nizki biti polnijo z ničami, zgornji grejo pa v zastavici C in X. Zastavica V se spremeni, kadar operand spremeni predznak. Pri ukazih **asr** premikamo operand desno. Nizki bit se prenese v zastavici C in X, najvišji bit pa ostane nespremenjen. Osnovne oblike ukazov so

**asd.<dolžina> Dn,Dn**  
**asd.<dolžina> #<podatek>,Dn**  
**asd.<dejanski naslov>**

Primeri:

**asl.w d1,d3** Premakni za d1 bitov vsebino registra d3 v levo.

**asr.b #2,d5** Premakni za dva bita vsebino registra d5 desno.

**asl (a1)** Premakni vsebino naslova, ki ga kaže register a1, za en bit v levo.

Podobna ukaza sta **lsl** in **lsr**. Načini naslavljanja so taki kot pri aritmetičnemu pomikanju. Edina razlika je pri ukazih **lsl**. Operacija poteka tako, da gre najnižji bit v zastavici C in X, na mesto najvišjega pa pride ničla. Primeri:

**lsl.w d2,d6** Logično pomakni za d2 bitov vsebino spodnjih 16 bitov registra d6 desno.

**lsl.b #3,d7** Logično pomakni za 3 bite vsebino spodnjih osmih bitov registra d7 v levo.

**lsl \$1000** Logično pomakni vsebino naslova \$1000 za en bit v levo.

K tej skupini spadata še ukaza za rotacijo bitov levo in desno. Dolžina operacije je lahko vbe, beseda ali dolga beseda. Načini naslavljanja so taki kot pri zgornjih štirih primerih. Pri vrtenju levo se najvišji bit preseli na mesto najnižjega in v zastavici C, drugi bit pa se premaknejo levo. Pri vrtenju v desno je prav nasprotno, biti se premaknejo desno, najnižji bit pa se preseli v zastavico C in na mesto najvišjega. Na zastavico X ukaza nimata vpliva. Primeri:

**rol,d3,d7** Vrta v levo vseh 32 bitov registra d7 za d3 bitov.

**ror (a3)** Vrta v desno vsebino naslova, ki ga kaže register a3, za en bit. Zamenja bile od 0 do 7 z biti od 8 do 15 v registru d1.

**rol,w #8,d1** Vrta v levo vsebino naslova, ki ga kaže register a3, za en bit. Zamenja bile od 0 do 7 z biti od 8 do 15 v registru d1.

Ukaza **rol** in **ror** imata enake načine naslavljanja kot njuna bližnja sorodnika **ror** in **ror**. Razlika je v tem, da se tukaj vrta tudi skozi zastavico X. Tako imamo en bit več. Pri vrtenju v levo se pomaknejo biti v levo, nato gre najvišji bit v zastavici C in X, iz zastavice X pa gre en bit na mesto najnižjega. Pri vrtenju v desno gredo biti v desno, najnižji bit gre v zastavici C in X, iz zastavice X pa se premakne na mesto najvišjega bita. Primeri:

**roxl,w d3,d2** Vrta čez zastavico X vsebino spodnjih 16 bitov registra d2 za d3 bitov v levo

**roxr \$3210** Vrta čez zastavico X vsebino naslova \$3210 za en bit v desno.

## Operacije z biti

Za operacije z biti so v M 68000 štirje ukazi. Njihove funkcije so testiranje, postavitve, brisanje in zamenjava bita. Nmemoniki so **btst**, **baet**, **bclr** in **bchg**. Kadar je cilj podatkovni register, je dolžina operanda dolga beseda, v vseh drugih primerih pa byt. Ukazi učinkujejo samo na zastavico Z, druge se ne spremenijo. Ukaz **btst** ima naslednji osnovni obliki

**btst. (dolžina) Dn, <dejanski naslov>**  
**btst. (dolžina) # <podatek>, <dejanski naslov>**

Cilj ne sme biti neposredni naslovni register. Primeri:

**btst,b d1,\$4** Testiraj bit d1 naslova \$4.

**btst,l #14,d5** Testiraj 14. bit registra d5.

Pri ukazih **baet**, **bclr** in **bchg** so prepovedani načini naslavljanja cilja: An, d(pc), d(pc,Xi) in #xxxx. Primeri:

**baet,b d3,(a7)** Testiraj bit d3 vsebine naslova kazalca sklada in nato ta bit postavi na ena.

**bclr,l #0,d0** Testiraj bit 0 registra d0 in ga nato zbrisi.

**bchg,b d2,7(a5)** Testiraj bit d2 naslova, ki ga kaže vsota a5+7. In ga invertiraj.

**bset,l #22,d1** Testiraj 22. bit vsebine registra d1 in ga postavi na 1.

## Operacije z BCD

Za računanje v BCD (dvojiško kodirani desetiški številci) so na voljo trije ukazi. Prvi je **abcd**. Dolžina operanda je byt. Dva operanda se dobi seštejeta, vsoti se prišteje vrednost zastavice X. Obliki ukaza sta

**abcd Dn,Dn**  
**abcd -(An),-(An)**

Primeri:

**abcd d1,d2** Po BCD pravih šestje vsebini spodnjih osmih bitov registrov d1 in d2. Vsoti prišteje vrednost zastavice X.

**abcd -(a3),-(a6)** Naprej zmanjšaj vsebino registrov a3 in a6 za ena. Nato po BCD pravih šestje vsebini naslovov, ki ju kazeta registra a3 in a6. Rezultatu prišteje vsebino zastavice X.

Pri ukazih **abcd** je vse tako kot pri **abcd**, le da gre tu za odštevanje. Ukaz **abcd** je namenjen BCD negaciji. Operacija poteka tako, da od števila 0 odštejemo vrednost operanda, od dobljene razlike pa še vrednost zastavice X. Dolžina operanda je byt. Prepovedani načini naslavljanja so: An, d(pc), d(pc,Xi) in #xxxx. Primeri:

**abcd d3** Izvedi BCD negacijo nad spodnjimi osmimi biti d3 registra d3. Od razlike odštej vsebino zastavice X.

**abcd 10(a1,d5,w)** Izvedi BCD negacijo nad naslovom, ki ga kaže vsota 10, registra a1 in predznačno razširjenega registra d5. Od razlike odštej zastavico X.

## Operacije za kontrolno sistema

Na začetku naše šole smo omenili, da je nadzorniški modus namenjen predvsem sistemskim programom. Za povezovanje teh programov z uporabniškimi skrbijo naslednji ukazi.

Z ukazom **chk** preverimo, ali je vsebina registra v določenih mejah. Dolžina operanda je beseda. Splošna oblika ukaza je

**chk <dejanski naslov>,Dn**

Processor preverja, ali je vsebina spodnjih 16 bitov podatkovnega registra večja od dejanskega naslova oziroma manjša od nič. Če je, se sproži past. Processor prebere vektor "CHK ukaz", preide v nadzorniški modus in izvede ustrezen dogodek. Dejanski naslov je lahko naslovljen na katerikoli način razen kot neposredni naslovni register An. Če je register Dn manjši od nič, se zastavica N prižge. Če je register Dn večji od dejanskega naslova, se N zbrise. V drugih primerih stanje te zastavice ni definirano. Primeri:

**chk #100,d1** Preveri, ali je vsebina spodnjih 16 bitov registra d1 med 0 in 100. Če ni, sproži past.

**chk d3,d4** Preveri, ali je vsebina spodnjih 16 bitov registra d4 med 0 in vsebino registra d3. Če ni, sproži past.

Z ukazom **trap** softversko izvajamo dogodke. Programski števec in statusni register se shranita v nadzorniški sklad. Nato processor preide v nadzorniški modus in prek vektorjev skoči na ustrežno rutino. Vektorjev je na voljo 16. Na zastavice ni vpliva. Splošna oblika ukaza je

**trap #<vektor>**

Primeri:

**trap #0** Skoči prek prvega vektorja na ustrežno rutino.

**trap #15** Skoči prek zadnjega vektorja na ustrežno rutino.

Z ukazom **trapv** prestrezamo napake prekoračitve. Izvede se le, kadar je zastavica V prižgana. Processor preide prek vektorjev v izvajanje ustreznega dogodka. Drugače se obnaša kot **nop**.

Ukaz **rtc** poganja statusni register in programski števec iz sistemskega sklada. Stari programski števec in statusni register sta izgubljena. Zastavice se postavijo odvisno od vsebine statusnega registra. Ukaz je privilegiran in se uporablja za vrnitve iz dogodkov. To pomeni, da ga lahko uporabljamo le v nadzornem modusu.

Ukaz **reset** je prav tako privilegiran. V uporabniškem modusu processor preide v izvajanje ustreznega dogodka, v nadzorniškem pa ukaz postavi linijo **reset** in povzroči, da se vse zunanje enote postavijo na začetno stanje. Sam ukaz ne vpliva na processor, (ta normalno izvaja ukaze naprej).

Ukaz **stop**: v uporabniškem modusu se izvede ustrezen dogodek. To pomeni, da je ukaz privilegiran. Oblika je

**stop #<podatek>**

V nadzorniškem modusu se neposredni podatek prenese v statusni register. Programski števec se poveča in kaže na naslednji ukaz. Nato se izvajanje ukazov do nadaljnjega ustavi. Nadaljuje se spet ob enem od naslednjih dogodkov:

1. TRACE

2. prekinitevni dogodek, katerega prioriteta je večja od procesorske 3. Zunanji reset.

Prva dogodka se lahko sprožita edino, kadar je vsebina statusnega registra ustrežno postavljenja. Primer:

**stop #S2300** Čakaj na prekinitev, ki ima višjo prioriteto kot 3.

Naslednjih šest ukazov kontrolira vsebino statusnega registra. Vsi so privilegirani. Prvi je **andi** to sr. Njegova oblika je

**andi #<podatek>,sr**

Dolžina operanda je beseda. Zastavice se napolnijo odvisno od podatka. Če je processor v uporabniškem modusu, se ob poskusu, da bi izvedli ta ukaz, sproži past. Primer:

**andi #SF8FF,sr** Postavi najnižjo prioriteto.

Podobna ukaza sta **ori** to sr in **ori** to sr. Prvi je ekskluzivni ali s statusnim registrom, drugi pa logični ali s statusnim registrom. Primeri:

**ori #S8000,sr** Vklopi trace modus.

**ori #S0700,sr** Invertiraj procesorjevo prioriteto.

Z ukazom **move** to sr prenašamo podatke v statusni register. Dolžina operanda je beseda. Zastavice se postavijo v skladu z vsebino operanda. Operand je lahko naslovljen na katerikoli način razen kot neposredni naslovni register An. Primeri:

**move #8700,sr** Postavi prioriteto na najvišjo stopnjo in vključi trace.

**move d3,sr** Prenesi vsebino registra d3 v statusni register.

Ukaz **move from sr** ni privilegiran. To pomeni, da ga lahko kličemo iz uporabniškega modusa. Na zastavice ne vpliva. Prepovedani načini

naslavljanja so: An, d(pc), d(pc,Xi) in \*xxxx. Dolžina operanda je beseda. Primer:

**move sr,\$5432**

Prenesi besedo iz statusnega registra na naslov \$5432.

Za prenašanje vsebine naslovnega registra iz uporabniškega kazalca skladi uporabljamo ukaz **move usp**. Ukaz je privilegiran. Zastavice ostanje nespremenjene. Operand je dolg 32 bitov. Možni sta obliki ukaza

**move usp,An**  
**move An,usp**

Za konec nam ostanejo ukazi z registrom stanj. Prvi trije so logični in, logični ali in ekskluzivni ali z registrom stanj. Dolžina operanda je byte, zastavice pa se spreminjajo v skladu s podatkom. Primeri:

**andi \$F0,ccr**

Zbrši spodnje štiri bite registra stanj.

**ori +\$0F,ccr**

Postavi spodnje štiri bite registra stanj.

**orl +\$FF,ccr**

Invertiraj vse bite registra stanj.

Zadnji ukaz za kontrolo sistema je **move to ccr**. Dolžina operanda je beseda. Zastavice se spreminjajo v skladu z operandom. Dovoljeni so vsi načini naslavljanja razen neposrednega naslovnega registra An. Splošna oblika ukaza je

**move <dejanski naslov>,ccr**

Ukaz, ki ne sodi v nobeno od teh skupin, je **illegal** in je popolnoma namenjen uporabniku. Narediti morate le to, da v tabeli vektorjev spremenite vektor "illegal instruction". Pred vstopom v vašo rutino procesor shrani programski števec in statusni register. V sistem se vrnete z rfe.

Pri MC 68010 je še nekaj izpopolnjenih in novih ukazov. Ker računalnikov s tem procesorjem ni veliko, se ne bomo spuščali še v te podrobnosti.

Sedaj ko smo se naučili vseh ukazov in znamo programirati na papirju, je že čas, da napišemo kakšen programček. Recimo, da bomo napisali enostaven HARDCOPY (tiskalniški zapis zaslona) za računalnik QL. Rutina naj bi bila združljiva z vsemi Eponovimi tiskalniki in njihovimi kloni. Ker tiskanje običajno traja zelo dolgo, se z barnviti odtenki sivine ne bomo ukvarjali.

\*Najprej si nastavimo nekaj koristnih tabel

**ekran equ \$28000**

**lo\_open equ 1**

**lo\_close equ 2**

**lo\_sstr equ 7**

**lo\_sbyte equ 5**

**start moveq #-1,d1**

**moveq +0,d3**

**lea kanal,a0**

**moveq +lo\_open,d0**

**trap +2**

**ts! d0**

**bne.s konec**

**movea.l a0,a5**

**moveq +3,d2**

**moveq #-1,d3**

**lea rfe,a1**

**moveq +lo\_sstr,d0**

**trap +3**

**ts! d0**

**bne.s error**

**lea ekran,a4**

**moveq +35,d7**

**prip1 bsr.s pline**

**bsr.s pline**

**dbra d7,prp1**

**moveq +0,d0**

**error move.l d0,d4**

**movea.l a5,a0**

**moveq +3,d2**

**moveq #-1,d3**

**lea rfe,a1**

**moveq +lo\_sstr,d0**

**trap +3**

**moveq +lo\_close,d0**

**trap +2**

**move.l d4,d0**

**konec rfe**

**pline movea.l a5,a0**

**moveq +10,d1**

**moveq #-1,d3**

**moveq +lo\_sbyte,d0**

**trap +3**

**trap +3**

**trap +3**

**trap +3**

**trap +3**

**trap +3**

**moveq +4,d2**

**moveq #-1,d3**

**lea graph,a1**

**moveq +lo\_sstr,d0**

**trap +3**

**move.w +\$200,d2**

**moveq +lo\_sstr,d0**

**trap +3**

**pline lea buffer,a0**

**moveq +127,d1**

**g!p1 clr.l (a0)+**

**dbra d1,g!p1**

**move.w +128,d5**

**moveq +6,d6**

**g!p2 lea buffer,a0**

**moveq +63,d4**

**g!p3 move.w (a4)+,d0**

**moveq +7,d3**

**g!p4 move.w d3,d2**

**bst.b d2,d0**

**bne.s ggo1**

**addq.b +8,d2**

**bst.l d2,d0**

**beq.s ggo2**

**ggo1 move.b d5,d2**

**or.b (a0),d2**

**move.b (a2)+,a0**

**ggo2 move.b d0+,d1**

**dbra d3,g!p4**

**dbra d4,g!p3**

**lar.b +1,d5**

**dbra d6,g!p2**

**rfe**

**kanal dc.w 4,'ser1'**

**rfe dc.b 27,'A',8,0**

**rfe1 dc.b 27,'A',11,0**

**graph dc.b 27,'K',0,2**

**buffer ds.b 512**

**buffer ds.b 512**

## Obdelava izjemnih dogodkov

Osnovna naloga procesorja je izvajanje ukazov. Vendar se lahko prijeti kakšen nepredviden ali predviden dogodek. Takrat procesor prekine običajno delo in se posveti temu dogodku. Procesor je stalno v enem od treh stanj: normalnem, zgodkovnem ali pavzi. V normalnem stanju izvaja ukaze in shranjuje rezultate. Poseben primer normalnega stanja je, kadar procesor izvede ukaz STOP. Takrat se ustavi. Pri MC 68010 pride v normalno stanje tudi, ko izvede ukaz Dbcc. To je zankovni modus, v katerem procesor bere samo operande in ne tudi ukazov.

Dogodkovno stanje je povezano s prekinitvami, pastmi, slednjem in drugimi dogodkovnimi pogoji. Notranji dogodek lahko generirajo ukazi ali neobičajni pogoji med izvajanjem ukazov. Znanji dogodki pa povzročijo prekinitev, napaka na vodilu ali reset. Tako nam procesor zagotavlja zelo učinkovito zaščito pred nepredvidnimi dogodki.

Pavza je znak katastrofalne hardverske napake. V tem stanju lahko procesor znova požene le zunanji reset. To se lahko zgodi, če procesor npr. oskrbuje dogodke, ki ga je povzročila napaka na vodilu, in medtem nastane tam nova napaka. Procesor sklepa, da sistem ni stabilen, in se ustavi. Pavze ne smemo istovetiti z ustavljanjem (STOP).

Vsi dogodki se obravnavajo v štirih korakih. V prvem se trenutna vsebina statusnega registra prenese v nadzorni sklad in procesor prebere v nadzorni modus. V drugem procesor prebere ustrezni dogodkovni vektor. Tretji korak je shranitev drugih potrebnih parametrov. Na koncu procesor nastavi nekaj registrov in začne izvajati ukaze.

Omenili smo nekaj vrst dogodkov. Nismo pa še povedali, kako procesor ve, kje je reset, ki oskrbuje te dogodke. V prvih 1024 bitih pomnilnika so vektorji, od katerih je vsak dolg 4 byte ali eno dolgo besedo. Tako jih imamo na voljo 255. Njihov pomen kaže tabela 2.

Ob resetu ali vklopu računalnika preide procesor v nadzorni modus in izklopi trace mod. Prioriteto prekinitev postavi na sedmo stopnjo in prebere prva dva vektorja. MC 68010 še prej postavi VBR register na nič. Vektor 0 vzame za nadzorniški kazalec skladi, vektor 1 pa za programski števec. Takoj zatem začne izvajati instrukcije.

MC 68000 in MC 68010 imata 7 stopen prekinitev, MC 68008 pa le tri. Te so 2., 5. in 7. stopnje. Sedma stopnja je najvišja in je nemaskirana. To pomeni, da postavitve prioritete procesorja na sedmo stopnjo ne prepreči prekinitev. To določimo v statusnem registru (biti od 8 do 10). Prekinitev nastane le, kadar je stopnja prekinitev višja od prioritete procesorja. Če med izvajanjem prekinitev nastane napaka na vodilu, procesor izvede nov dogodek, ki se imenuje nepravilni interrupt

Morda najbolj zanimive za programerja so pasti (traps). To so dogodki, ki jih povzročajo nepredvideni pogoji med izvajanjem ukazov ali ukazi same. V resnici ti ukazi tudi učinkujejo kot pasti za napake. Če uporabimo npr. ukaza DIVU ali DIVS, deljenec pa je nič, se bo sprožil poseben dogodek, ki rabi temu namenu. Programerji, ki bi radi imeli lastne ukaze v mikroprocesorju, bodo s pridom uporabili ukaze TRAP #0 - TRAP #15. V grobem bi jih lahko primerjali z restarti v Z 80, le da si tu lahko določimo po vektorjih, katero rutino bodo izvedli. Njihov osnovni namen pa je povezovati uporabniške programe s sistemskimi.

Uporabniku je prav tako namenjen ukaz ILLEGAL, ki po vektorjih omogoča klic kake rutine. Pri vseh procesorjih iz družine M 68000 so tri instrukcijske kode, ki niso uporabljene. Če jih pokličemo, sprožimo dogodek "nepravilni ukaz". Dve od teh treh kod sta rezervirani za Motoroline sistemske izdelke, tretja pa je namenjena uporabniku. Zelo pomembno dejstvo, ki ga še nismo omenili, je to, da je vsak ukaz dolg najmanj 16-n bitov, torej 2 byta. Ukazov, katerih koda se začne s Fxxx ali Axxx, ni. Prav to je zelo močno orodje, ki nam ga daje Motorola. Ob poskusu, da bi izvedli te ukaze, se po ločenih vektorjih sprožijo dogodki, ki to oskrbujejo. Tako emuliramo ukaze, ki jih ni. Pri MC 68020 je linija Fxxx uporabljena za komunikacijo s koprocessori.

Za zaščito sistema je nekaj ukazov privilegiranih. Če poskusimo izvesti katerega od njih v uporabniškem modusu, se sproži dogodek "kršitev privilegijev". Ta je skoraj identičen z nepravilnim ukazom.

Privilegirani ukazi so:

- ANDI to SR
- EORI to SR
- MOVE to SR
- MOVE USP
- ORI to SR
- RESET
- RTE
- STOP
- Pri MC 68010 so še trije ukazi.
- MOVE from SR BX
- MOVES

Lastniki spectrumov, ki obvladajo strojni jezik za Z 80, s pridom uporabljajo MONS, DDT ali kak drug monitor/disassembler. Prav gotovo pa jim je še posebej pri srcu možnost, da sledijo strojnemu programom ukaz za ukazom. Malokdaj pa ve, da tak program niti ni tako enostaven. Konstruktorji M 68000 so se tega dobro zavedali, zato so slednje vedeli v procesor. Uporaba je izredno enostavna. Treba je le nastaviti ustrezen vektor na našo rutino, ki bo izpisovala vsebino registrov, in postaviti sledni bit (trace bit) v statusnem registru na ena. Za vse drugo bo skrbel procesor sam.

Na koncu naše šole povzemimo vse ukaze M 68000.

Ime	Opis	Velikost	N	Z	V	C	X
abcd	Desetiško seštevanje Načina: <b>abcd Dn,Dn</b>	1	N	V	N	S	S
add	Dvojiško seštevanje Načina: <b>add &lt;dn&gt;,Dn</b> <b>add Dn,&lt;ps&gt;</b>	1, 2, 4	S	S	S	S	S
adda	Prištej naslovu Način: <b>adda &lt;dn&gt;,An</b>	2, 4	-	-	-	-	-
addi	Prištej takoj Način: <b>addi #&lt;nep&gt;,&lt;ds&gt;</b>	1, 2, 4	S	S	S	S	S
addq	Prištej hitro Način: <b>addq #&lt;nep&gt;,&lt;s&gt;</b>	1, 2, 4	A	A	A	A	A
addx	Seštej z zastavico X Načina: <b>addx Dn,Dn</b> <b>addx -(An),-(An)</b>	1, 2, 4	S	S	O	O	-
and	Logični 'in' Načina: <b>and &lt;d&gt;,Dn</b> <b>and Dn,&lt;ps&gt;</b>	1, 2, 4	S	S	O	O	-
andi	Logični 'in' Način: <b>andi #&lt;nep&gt;,&lt;ds&gt;</b>	1, 2, 4	S	S	O	O	-
andi s CCR	Logični 'in' Način: <b>andi #&lt;nep&gt;,ccr</b>	1	V	V	V	V	V
andi s SR	Logični 'in' Način: <b>andi #&lt;nep&gt;,sr</b>	2	V	V	V	V	V
asl,asr	Aritmetični pomik levo/ desno Načini: <b>asd Dn,Dn</b> <b>asd #&lt;nep&gt;,Dn</b> <b>asd &lt;ps&gt;</b>	1, 2, 4	S	S	S	S	S
bcc	Pogojna vejitev Način: <b>bcc &lt;labela&gt;</b>	1, 2	-	-	-	-	-
bra	Brezpogojna vejitev Način: <b>bra &lt;labela&gt;</b>	1, 2	-	-	-	-	-
bsr	Vejitev v podprogram Način: <b>bsr &lt;labela&gt;</b>	1, 2	-	-	-	-	-
bchg	Preskusi bit in ga spremeni	1, 4	-	S	-	-	-

	Načina: <b>bchg Dn,&lt;ds&gt;</b> <b>bchg</b> <b>#&lt;nep&gt;,&lt;ds&gt;</b>						
bclr	Preskusi bit in ga zbrši Načina: <b>bclr Dn,&lt;ds&gt;</b> <b>bclr #&lt;nep&gt;,&lt;ds&gt;</b>	1, 4	-	S	-	-	-
bset	Preskusi bit in ga priži Načina: <b>bset Dn,&lt;ds&gt;</b> <b>bset #&lt;nep&gt;,&lt;ds&gt;</b>	1, 4	-	S	-	-	-
btst	Preskusi bit Načina: <b>btst Dn,&lt;ds&gt;</b> <b>btst #&lt;nep&gt;,&lt;ds&gt;</b>	1, 4	-	S	-	-	-
chk	Preveri (po potrebi TRAP) Način: <b>chk &lt;d&gt;,Dn</b>	2	N	N	N	N	N
clr	Izbrši Način: <b>clr &lt;ds&gt;,Dn</b>	1, 2, 4	O	1	O	O	O
cmp	Primerjaj Način: <b>cmp &lt;dn&gt;,Dn</b>	1, 2, 4	S	S	S	S	S
cmpa	Primerjaj z naslovnim reg. Način: <b>cmpa &lt;dn&gt;,An</b>	2, 4	S	S	S	S	S
cmpi	Primerjaj podatek Način: <b>cmpi #&lt;nep&gt;,&lt;ds&gt;</b>	1, 2, 4	S	S	S	S	S
cmpm	Primerjaj pomnilnik Način: <b>cmpm (An)+,(An)+</b>	1, 2, 4	S	S	S	S	S
dbcc	Zmanjšaj, preskusi in razveji Način: <b>dbcc Dn,&lt;labela&gt;</b>	2	-	-	-	-	-
dbra	Zmanjšaj in razveji Način: <b>dbra Dn,&lt;labela&gt;</b>	2	-	-	-	-	-
divs	Predznačeno deljenje Način: <b>divs &lt;d&gt;,Dn</b>	2	S	S	S	O	-
divu	Nepredznačeno deljenje Način: <b>divu &lt;d&gt;,Dn</b>	2	S	S	S	O	-
eor	Logični 'izključni ali' Način: <b>eor Dn,&lt;ds&gt;</b>	1, 2, 4	S	S	O	O	-
eori	Logični 'izključni ali' Način: <b>eori #&lt;nep&gt;,&lt;ds&gt;</b>	1, 2, 4	S	S	O	O	-
eori s ccr	Logični 'izključni ali' Način: <b>eori #&lt;nep&gt;,ccr</b>	1	V	V	V	V	V
eori s sr	Logični 'izključni ali' Način: <b>eori #&lt;nep&gt;,sr</b>	2	V	V	V	V	V
exg	Zamenjaj registra Način: <b>exg Rn,Rn</b>	4	-	-	-	-	-
ext	Razširi s predznakom Način: <b>ext Dn</b>	2, 4	S	S	O	O	-
jmp	Način: <b>jmp &lt;n&gt;</b>						
jsr	Skoči v podprogram Način: <b>jsr &lt;n&gt;</b>						
lea	Naloži dejanski naslov Način: <b>lea &lt;n&gt;,An</b>	4	-	-	-	-	-
link	Podprogramska povezava Način: <b>link An,#&lt;nep&gt;</b>						
lsl,lshr	Logični pomik levo/desno Načini: <b>lsd Dn,Dn</b> <b>lsd #&lt;nep&gt;,Dn</b> <b>lsd &lt;ps&gt;</b>	1, 2, 4	S	S	O	S	S
move	Prenos podatkov Način: <b>move &lt;dn&gt;,&lt;ds&gt;</b>	1, 2, 4	S	S	O	O	-
movea	Prenos v naslovni register Način: <b>movea &lt;dn&gt;,An</b>	2, 4	-	-	-	-	-
movem	Prenos skupinsko Način: <b>movem &lt;sr&gt;,-(An)</b> <b>movem &lt;sr&gt;,&lt;ns&gt;</b> <b>movem (An)+,&lt;sr&gt;</b> <b>movem &lt;n&gt;,&lt;sr&gt;</b>	2, 4	-	-	-	-	-
movep	Prenos periferno Načina: <b>movep Dn,d(An)</b> <b>movep d(An),Dn</b>	2, 4	-	-	-	-	-
moveq	Prenos hitro Način: <b>moveq #&lt;nep&gt;,Dn</b>	4	S	S	O	O	-
moveq v ccr	Prenos v ccr Način: <b>move &lt;d&gt;,ccr</b>	2	S	S	S	S	S
move v sr	Prenos v sr (nadzorni) Način: <b>move &lt;d&gt;,sr</b>	2	S	S	S	S	S
move iz sr	Prenos iz sr Način: <b>move sr,&lt;ds&gt;</b>	2	-	-	-	-	-
move usp	(nadzorni) Način: <b>move usp,An</b> <b>move An,usp</b>	4	-	-	-	-	-
mul	Predznačeno množenje Način: <b>mul &lt;d&gt;,Dn</b>	2	S	S	O	O	-
mulu	Nepredznačeno množenje Način: <b>mulu &lt;d&gt;,Dn</b>	2	S	S	O	O	-

nbcd	Način: <b>mulu</b> <d>,Dn Desetška negacija Način: <b>nbcd</b> <ds>	1	N V N S S
neg	Dvojiška negacija Način: <b>neg</b> <ds>	1, 2, 4	S S S S S
negx	Dvojiška negacija z zast. X Način: <b>negx</b> <ds>	1, 2, 4	S V S S S
nop	Nobena operacija Način: <b>nop</b>	-----	-----
not	Logični komplement Način: <b>not</b> <ds>	1, 2, 4	S S O O -
or	Logični ali Načina. <b>or</b> <d>,Dn or Dn,<ds>	1, 2, 4	S S O O -
ori	Logični ali Način: <b>ori</b> #<nep>,<ds>	1, 2, 4	S S O O -
ori s ccr	Logični ali Način: <b>ori</b> #<nep>,ccr	1	V V V V V
ori s sr	Logični ali (nadzorniški) Način: <b>ori</b> #<nep>,sr	2	V V V V V
pea	Porini dejanski naslov Način: <b>pea</b> <n>	4	-----
reset	Ponovno prižgi (nadzorniški) Način: <b>reset</b>	-----	-----
rol,ror	Pomakni krožno levo/desno Načini: <b>rod</b> Dn,Dn <b>rod</b> #<nep>,Dn <b>rod</b> <ps>	1, 2, 4	S S O S -
roxl,roxr	Pomakni krožno levo/desno Načini: <b>roxd</b> Dn,Dn <b>roxd</b> #<nep>,Dn <b>roxd</b> <ps>	1, 2, 4	S S O S S
rte	Vrni se iz dogodka Način: <b>rte</b>	-----	S S S S S
rtr	Vrni se in postavi stari ccr Način: <b>rtr</b>	-----	S S S S S
rts	Vrni se iz podprograma Način: <b>rts</b>	-----	-----
sbcd	Odštej desetiško Načina. <b>sbcd</b> Dn,Dn <b>sbcd</b> -(An),-(An)	1	N V N S S
sc	Postavi glede na pogoj Način: <b>sc</b> <ds>	1	-----
stop	Čakaj na dogodek Način: <b>stop</b> #<nep>	-----	S S S S S
sub	Odštej dvojiško Načina <b>sub</b> <dn>,Dn <b>sub</b> Dn,<ps>	1, 2, 4	S S S S S
suba	Odštej od naslovnega registra Način: <b>suba</b> <dn>,An	2, 4	-----
subi	Odštej podatek Način: <b>subi</b> #<nep>,<ds>	1, 2, 4	S S S S S
subq	Odštej hitro podatek Način: <b>subq</b> #<nep>,<s>	1, 2, 4	S S S S S
subx	Odštej z zastavico X Načina. <b>subx</b> Dn,Dn <b>subx</b> -(An),-(An)	1, 2, 4	S S S S S
swap	Zamenjaj polovici registra Način: <b>swap</b> Dn	2	S S O O -
tas	Preskusi bit in ga prižgi Način: <b>tas</b> <ds>	1	S S O O -
trap	Skoči v past (dogodek TRAP) Način: <b>trap</b> #<nep>	-----	-----
trapv	Skoči v past pri prekoračitvi Način: <b>trapv</b>	-----	-----
tst	Primerjaj z nič Način: <b>tst</b> <ds>	1, 2, 4	S S O O -
unik	Podprogramska razveza Način: <b>unik</b> An	-----	-----

### Pomen simbolov pri zastavkah:

-	Se ne spremeni
0	Postavi se na logično 0
1	Postavi se na logično 1
A	Spremeni se, če cilj ni naslovni register
S	Spremeni se v skladu z vrednostjo
V	V nekaterih primerih se spremeni v skladu z vrednostjo
N	Ni definirano

### Načini naslavljanja

An	Naslovni register
Dn	Podatkovni register

Rn	Katerikoli register
(An)	Posredno naslovno naslavljanje
d(An)	Posredno naslovno naslavljanje z nadomestkom
-(An)	Posredno naslovno naslavljanje s predhodnim zmanjšanjem
(An)+	Posredno naslovno naslavljanje s poznejšim povečanjem
<dn>	Katerikoli naslovni način
<s>	Spremenljivi naslovni način
<n>	Nadzorni naslovni način
<d>	Podatkovni naslovni način
<ns>	Nadzorno-spremenljivi naslovni način
<ds>	Podatkovno-spremenljivi naslovni način
<ps>	Pomilinski-spremenljivi naslovni način
<sr>	Seznam registrov
<nep>	Neposredni podatki

št. vektorja	dec	hex	prostor	naloga
0	0	000	SP	reset, začetni SSP
1	4	004	SP	reset, začetni PC
2	8	008	SD	napaka vodila
3	12	00C	SD	napaka naslova
4	16	010	SD	neznačna instrukcija
5	20	014	SD	deljenje z nič
6	24	018	SD	instrukcija CHK
7	28	01C	SD	instrukcija TRAPV
8	32	020	SD	kršitev privilegijev
9	36	024	SD	sledenje
10	40	028	SD	vrstični 1010 emulator
11	44	02C	SD	vrstični 1111 emulator
12	48	030	SD	(neuporabljeno, rezervirano)
13	52	034	SD	(neuporabljeno, rezervirano)
14	56	038	SD	formatna napaka
15	60	03C	SD	neimicaliziran prekinitveni vektor
16-23	64	040	SD	(neuporabljeno, rezervirano)
	95	05F	-	-
24	96	060	SD	neprava prekinitev
25	100	064	SD	avtovektor prekinitve 1 stopnje
26	104	068	SD	avtovektor prekinitve 2 stopnje
27	108	06C	SD	avtovektor prekinitve 3 stopnje
28	112	070	SD	avtovektor prekinitve 4 stopnje
29	116	074	SD	avtovektor prekinitve 5 stopnje
30	120	078	SD	avtovektor prekinitve 6 stopnje
31	124	07C	SD	avtovektor prekinitve 7 stopnje
32-47	128	080	SD	vektorji instrukcije TRAP
	191	0BF	-	-
48-63	192	0C0	SD	(neuporabljeno, rezervirano)
	255	0FF	-	-
64-255	256	100	SD	uporabniški prekinitveni vektorji
	1023	3FF	-	-

Tabela 2. Dogodkovni vektorji

Literatura: M68000 16/32-bit Microprocessor; MOTOROLA Semiconductors, MC68000 16-bit Microprocessor, MOTOROLA Semiconductors, Programiranje M 68000, ZOTKS, Ljubljana 1985

Risbe so narejene z Applom računalnikom macintosh

# Matrike (2)

mag. MILKO KEVO, dipl. inž.

V teoriji in uporabi matrik je pomembno v množici definiranih ekvivalentnih matrik najti čimbolj enostavne in uporabne oblike, ki olajšajo algebrske operacije z matrikami. Po potrebi definiramo različne oblike matrik, ki jih imenujemo **normale** ali **kanonske** oblike dane matrike. Tukaj bomo podrobno obdelali dve taki obliki in njima ustrezne transformacije.

## Prevedba matrike v trikotno obliko (triangulacija)

Vsak kvadratno matriko  $A$ , naj bo regularna ali singularna, lahko z elementarnimi operacijami na vrsticah pretvorimo v gornjo ali spodnjo trikotno matriko s t. i. Gaussovo metodo množenja z leve z zaporedjem elementarnih matrik:

$$E_{n-1} E_{n-2} \dots E_1 A = TA = B$$

Pri tem je  $T$  trikotna matrika nasprotnega tipa kot trikotna matrika  $B$ , biti mora vselej regularna in mora zadoščati pogoju  $|T|=1$ .

Če želimo matriko  $A$  transformirati v zgornjo trikotno matriko, začnemo z prvi vrstici (pri predpostavki  $a_{11} > 0$ ). Prvo vrstico, pomnožimo z izrazom  $-a_{ij}/a_{11}$ , prištejemo  $i$ -ti vrstici in dobimo vse elemente pod  $a_{11}$  enake nič. Ta transformacija da enak rezultat, kot če matriko  $A$  z leve pomnožimo z elementarno matriko

$$E_1 = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ -a_{21}/a_{11} & 1 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ -a_{n1}/a_{11} & 0 & 0 & \dots & 1 \end{bmatrix}$$

Po tej transformaciji dobi matrika  $A$  obliko

$$A^{(1)} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ 0 & a_{22}^{(1)} & \dots & a_{2n}^{(1)} \\ \dots & \dots & \dots & \dots \\ 0 & a_{n2}^{(1)} & \dots & a_{nn}^{(1)} \end{bmatrix}$$

kjer je  $a_{ij}^{(1)} = a_{ij} - a_{ij}/a_{11}$ , ( $i,j=2,3,\dots,n$ )

Sedaj naredimo enako s kvadratno matriko reda  $(n-1)$ , ki ima elemente  $a_{ij}^{(1)}$ , se pravi, prištejemo drugo vrstico, pomnožimo z  $-a_{2i}^{(1)}/a_{22}^{(1)}$ ,  $i$ -ti vrstici ( $i=3, \dots, n$ ). Tako dobimo vse elemente pod  $a_{22}^{(1)}$  enake nič. Rezultat je druga transformacija  $A^{(2)}$ . Postopek ponavljamo, dokler po  $n-1$  ciklusu ne pridemo do cilja  $A^{(n-1)}=B$ .

V splošnem poteka  $k$ -ti ciklus s pivotnim elementom  $a_{kk}^{(k)}$  takole:

- Za  $k=1,2,\dots,n-1$  izračunamo faktorje  $m_{ik} = -a_{ik}^{(k)}/a_{kk}^{(k)}$ , ( $i=k+1, \dots, n$ ) za eliminacijo elementa  $a_{ik}^{(k)}$ .
- $K$ -to vrstico matrike  $A^{(k)}$  pomnožimo z  $m_{ik}$ , in jo prištejemo  $i$ -ti vrstici  $A^{(k)}$ :  $a_{ik}^{(k+1)} = a_{ik}^{(k)} + m_{ik} a_{kk}^{(k)}$ , ( $i,j=k+1, \dots, n$ )

Vsi drugi elementi matrike  $A^{(k)}$  ostanejo nespremenjeni.

Kot vidimo iz priloženega programa (vrstice 130 do 200), lahko algoritem zelo enostavno zakodiramo. Ukazi od 30 do 110 poskrbijo za vnos elementov začetne matrike po stolpcih, ukazi od 230 do 290 pa za izpis elementov trikotne matrike (ničel ne izpisujemo).

Rang dobijene ekvivalentne trikotne matrike (s tem pa tudi rang prvotne matrike) je očitno enak številu od nič različnih elementov na glavni diagonali in ustreza številu linearno neodvisnih vrstic začetne matrike. To pomeni, da lahko program 1 uporabimo tudi za določanje ranga matrike in za prevedbo splošnih pravokotnih matrik v trikotno obliko.

Da se dokazati, da pri elementarnih transformacijah samo na vrsticah dobijene trikotne matrike velja:

- Vsak glavni element, različen od nič, lahko pretvorimo v ena.
- Vse elemente vrstice, kjer je glavni element nič, lahko pretvorimo v nič.
- Vse elemente stolpca, kjer je glavni element ena, lahko pretvorimo v nič.

Ustrezno zaporedje transformacij nam da kanonsko obliko matrike, ki se imenuje Hermitova normalna oblika matrike. Očitno je Hermitova normalna oblika regularne matrike enotska matrika.

## Razstavev (razcep) kvadratne matrike na dve trikotni matriki

Katerokoli kvadratno matriko, kjer so glavni diagonalni minorigi različni od nič, lahko predstavimo kot produkt spodnje in zgornje trikotne matrike:  $A=LR$

Ta razcep (znan pod imenom metoda Choleskega) je enoličen, če so diagonalni elementi ene od trikotnih matrik vnaprej dani (npr. lahko so enaki enaj).

Za kvadratno matriko reda 3 imamo

$$A = LR = \begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \begin{bmatrix} 1 & r_{12} & r_{13} \\ 0 & 1 & r_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

Ker je produkt  $L$   $R$  enak matriki  $A$ , morajo biti v splošnem izpolnjeni pogoji

$$\sum_{k=1}^j l_{kj} r_{kj} = a_{ij} \text{ za } i \geq j \quad (j=1,2,\dots,n)$$

$$\sum_{k=1}^i l_{kj} r_{kj} = a_{ij} \text{ za } i < j \quad (i=1,2,\dots,n-1)$$

z naprej definiranimi  $r_{ii} = 1$ , ( $i=1,2,\dots,n$ )

Od tega sledi:

$$l_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{kj} r_{kj} \quad (\text{elementi } i\text{-te vrstice } L \text{ od } l_{i1} \text{ do } l_{in})$$

$$r_{ij} = \frac{1}{l_{ii}} (a_{ij} - \sum_{k=1}^{i-1} l_{kj} r_{kj}) \quad (\text{elementi } i\text{-te vrstice } R \text{ od } r_{i,i+1} \text{ do } r_{in})$$

### PROGRAM 1

```

10 CLS:PRINT"TRIANGULACIJA KVADRATNE MA
TRICE"
20 REM-vnos podatka
30 PRINT:INPUT"RED MATRICE, N=";N
40 DIM A(N,N)
50 FOR J=1 TO N
60 PRINT:PRINT"Stupac "J;"matrice A:"
70 FOR I=1 TO N
80 PRINT:element("I;"J");
90 INPUT A(I,J)
100 NEXT I
110 NEXT J
120 REM-----metoda Gaussa-----
130 FOR K=1 TO N-1
140 FOR J=K+1 TO N
150 A(J)=A(I,K)/A(K,K)
160 FOR J=K+1 TO N
170 A(I,J)=A(I,J)+A(J,K)*A(K,J)
180 NEXT J
190 NEXT I
200 NEXT K
210 REM-----kraj postopka-----
220 REM-ispis rezultata
230 CLS:PRINT"TRKOTNA MATRICA"
240 FOR J=1 TO N
250 PRINT:-----
pac br. J
260 FOR I=1 TO J
270 PRINT A(I,J)
280 NEXT I
290 NEXT J
300 PRINT:PRINT"KRAJ PROGRAMA"-END
element( 3 ) : 7
Stupac 2 matrice A:
element( 1 ) : 7 10
element( 2 ) : 7 8
element( 3 ) : 7 8
Stupac 3 matrice A:
element( 1 ) : 7 7 10
element( 2 ) : 7 7 10
element( 3 ) : 7 7 9
TRKOTNA MATRICA
-----stupac br. 1
9
-----stupac br. 2
10
11111111
-----stupac br. 3
10
11111111
-1
    
```



Pri tem se giblej indeks  $i$  v mejah  $i=1, \dots, n$ , indeks  $j$  pa v mejah  $j=i+1, \dots, n$ . Pri izračunavanju elementov prve vrstice  $R$  imamo deljenje z  $a_{11}$ , kar implicira, da  $a_{11}$  ni enak nič. Če ta pogoj ni izpolnjen, moramo zamenjati vrstice v začetni matriki  $A$ . Metoda Choleskega uporabljamo pri inverziranju matrik in reševanju sistemov linearnih algebrskih enačb.

Program 2 vsebuje opisni algoritem v vrsticah od 130 do 350. Vrstice od 30 do 110 so za vnos elementov matrike  $A$  po stolpcih, vrstice od 370 do 500 pa izpišejo elemente matrik  $L$  in  $R$ , prav tako po stolpcih. Vse elemente matrik  $L$  in  $R$  spravljamo v tabelo  $A$  razen enic na diagonalni  $R$ , ki jih določimo pri izpisu (vrstica 470). V programu ni dela za menjavo vrstnega reda vrstic.

## Inverzija matrike z metodo eliminacije

Če lahko reguliramo matriko  $A$  transformiramo v enotsko matriko  $I$  z množenjem z zaporedjem ustreznih izbranih elementarnih matrik  $E_k$  tipa  $E_i(c)$ ,  $E_{ij}$ ,  $E_{ii}(c)$ , dobimo z množenjem enotske matrike  $I$  z istim zaporedjem elementarnih matrik inverzno matriko  $A^{-1}$ .

Dokaz. Predpostavimo, da so znane elementarne matrike  $E_k$  ( $k=1, K$ ), tako da je  $E_k E_{k-1} \dots E_2 E_1 A = I$ .

Ko obe strani enačbe množimo z desne z  $A^{-1}$ , dobimo

$$E_k E_{k-1} \dots E_2 E_1 A^{-1} = A^{-1}$$

$$E_k E_{k-1} \dots E_2 E_1 I = A^{-1}$$

Problem je v iskanju ustreznega zaporedja  $E_k$ . Inverzijo nesingularne matrike  $A$  ( $n \times n$ ) izvedemo z ( $n$ ) transformacijami, vsaka pa je sestavljena iz dveh korakov. Transformacijo z zaporedno številko  $k$  ( $k=1, 2, \dots, n$ ) izvedemo na naslednji način.

Korak 1:

element  $a_{kk}$  normaliziramo z množenjem  $k$ -te vrstice z inverzno vrednostjo elementa  $a_{kk}$ , če ta inverzna vrednost obstaja. (Če je  $a_{kk}$  enak nič, njevo vrednosti ni. V tem primeru moramo  $k$ -to vrstico zamenjati z neko vrstico  $i$ , ki ima element  $a_{ik}$  od nič različen. V praksi  $k$ -to vrstico zamenjamo z vrstico  $R_{i+k}$ , kjer ima element  $a_{i+k,k}$  največjo absolutno vrednost  $k$ -tem stolpcu na glavni diagonalni ali pod njo. Ta postopek imenujemo pivotiranje elementov matrike.)

Korak 2:

elemente v koloni  $k$ , ki ne ležijo na glavni diagonalni, prevedemo v nič tako, da zamenjamo vrstico  $i$ ,  $i < k$  z ustreznim linearno kombinacijo  $i$ -te in  $k$ -te vrstice.

Tako bo prva transformacija nesingularne matrike  $A=[a_{ij}]$  reda  $n \times n$  dala rezultat

$$A^{(1)} = \begin{bmatrix} 1 & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22} & a_{23} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & a_{i2} & a_{i3} & \dots & a_{in} \end{bmatrix}$$

Tu zgornji indeksi označujejo zaporedno številko transformacije.

Ustrezne operacije na vrsticah matrike  $A$ , ki dajo  $A^{(1)}$ , so:  $R_1^1 = R_1 / a_{11}$  (predpostavka je  $a_{11} < 0$ );  $R_i^1 = R_i - a_{i1} R_1^1$  ( $i < 1$ ); oziroma elementa  $a_{i1}^1$  matrike  $A^{(1)}$  izračunamo z uporabo rekurzivnih formul

$$\begin{aligned} a_{i1}^1 &= a_{i1}/a_{11} \\ a_{ij}^1 &= a_{ij} - a_{i1}a_{1j}^1, (i \neq 1) \end{aligned} \quad (j=1, n)$$

Analno bo druga transformacija naredila matriko  $A^{(2)}$  iz matrike  $A^{(1)}$ :

$$A^{(2)} = \begin{bmatrix} 1 & 0 & a_{13}^2 & \dots & a_{1n}^2 \\ 0 & 1 & a_{23}^2 & \dots & a_{2n}^2 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & a_{i3}^2 & \dots & a_{in}^2 \end{bmatrix}$$

Ustrezne operacije na vrsticah so  $R_2^2 = R_2 / a_{22}^1$ ;  $R_i^2 = R_i^1 - a_{i2}^1 R_2^2$  ( $i < 2$ ), čemur ustrezajo formule za računanje elementov  $a_{ij}^2$  matrike  $A^{(2)}$ :

$$\begin{aligned} a_{ij}^2 &= a_{ij}^1/a_{22}^1 \\ a_{ik}^2 &= a_{ik}^1 - a_{i2}^1 a_{2k}^1, (i \neq 2) \end{aligned} \quad (j=1, n)$$

V splošnem bo  $k$ -ta transformacija ( $k=1, 2, \dots, n$ ) dala matriko

$$A^{(k)} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & a_{1,k+1}^k & \dots & a_{1n}^k \\ 0 & 1 & 0 & \dots & 0 & a_{2,k+1}^k & \dots & a_{2n}^k \\ 0 & 0 & 1 & \dots & 0 & a_{3,k+1}^k & \dots & a_{3n}^k \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 & a_{k,k+1}^k & \dots & a_{kn}^k \end{bmatrix}$$

Tu je  $R_i^k = R_i^{k-1} / a_{kk}^{k-1}$  (ob predpostavki  $a_{kk}^{k-1} \neq 0$ )

$$R_i^k = R_i^{k-1} - a_{ik}^{k-1} R_k^k \quad (i \neq k)$$

Temu ustrezajo rekurzivne formule za računanje elementov  $a_{ij}^k$ :

$$\begin{aligned} a_{ij}^k &= a_{ij}^{k-1} / a_{kk}^{k-1} \\ a_{ik}^k &= a_{ik}^{k-1} - a_{ik}^{k-1} a_{kk}^{k-1}, (i \neq k) \end{aligned} \quad (j=1, n) \quad (1)$$

z definiranimi začetnimi vrednostmi  $a_{ij}^0 = a_{ij}$ . Prav tako moramo definirati začetne vrednosti matrike  $B=i, t, j, b_{ij} = b_{ij}^0$ .

Očitno bo morala biti matrika  $A^{(n)}$ , dobljena v  $n$ -ti transformaciji, enotska matrika  $I$ . Kot smo že povedali: če identično zporočimo operacijo na vrsticah matrik  $A, A^{(1)}, A^{(2)}, \dots$ , hkrati uporabimo na matriki  $I=B, B^{(2)}, \dots$ , bo končni rezultat  $B^{(n)}=A^{-1}$ .

## PROGRAM 2

```

10 CLS:PRINT"TRIJANGULACIJA KVADRATNE MA
TRICE"
20 REM=vnos podatoka
30 PRINT:INPUT"RED MATRICE, N=?":N
40 DIM A(N,N)
50 FOR J=1 TO N
60 PRINT:PRINT"Stupac "J;"matrike A:"
70 FOR I=1 TO N
80 PRINT:element("I");":":
90 INPUT A(I,J)
100 NEXT I
110 NEXT J
120 REM=metoda Choleski:
130 REM=prvi stupac ostane nezmujen
140 REM=ostatak prvog reika
150 FOR J=2 TO N
160 A(I,J)=A(I,J)/A(I,1)
170 NEXT J
180 REM=ostatak I-tog reika
190 FOR I=2 TO N
200 REM=element od L
210 FOR J=2 TO I
220 J=J-1
230 FOR K=1 TO J
240 A(I,J)=A(I,J)-A(I,K)*A(K,J)
250 NEXT K
260 NEXT J
270 REM=element od R
280 I=I+1:I=I-1
290 FOR J=1 TO N
300 FOR K=1 TO I
310 A(I,J)=A(I,J)-A(I,K)*A(K,J)
320 NEXT K
330 A(I,J)=A(I,J)/A(I,I)
340 NEXT J
350 NEXT I
360 REM=-----kraj postopka-----
370 REM=ispis rezultata
380 CLS:PRINT"DONJIA T.M. GORNJA T.M."
390 FOR J=1 TO N
400 PRINT:
pac br.":J
410 FOR I=1 TO N
420 P=A(I,J)
430 IF I < J THEN A(I,J)=0
440 PRINT A(I,J),
450 A(I,J)=P
460 IF I > J THEN A(I,J)=0
470 IF I = J THEN A(I,J)=1
480 PRINT TAB(15);A(I,J)
490 NEXT I
500 NEXT J
510 PRINT:PRINT"KRAJ PROGRAMA"-END
RED MATRICE, N=? 3
Stupac 1 matrike A:
element ( 1 ):? 3
element ( 2 ):? 8
element ( 3 ):? 7
Stupac 2 matrike A:
element ( 1 ):? 10
element ( 2 ):? 9
element ( 3 ):? 8
Stupac 3 matrike A:
element ( 1 ):? 10
element ( 2 ):? 10
element ( 3 ):? 9
DONJIA T.M. GORNJA T.M.
-----stupac br. 1
          3      1
          8      0
          7      0
-----stupac br. 2
          0      1,1111111
          .11111111 1
          .22222222 0
-----stupac br. 3
          0      1,1111111
          0      9,9999999
          -.9999999 1

```

```

PROGRAM 3
10 CLS:PRINT"INVERZIJA I IZRACUNAVANJE
DETERMINANTE MATRICE A(NxN) METODOM ELIMINACIJE"
20 DIM A(50,50),B(50,50)
30 REM=unos podataka
40 PRINT:INPUT"RED MATRICE, N=";N
50 FOR J=1 TO N
60 PRINT:PRINT"Stupac ";J;"matrice A:"
70 FOR I=1 TO N
80 PRINT"element( ";I;"; ";J;");";
90 INPUT A(I,J)
100 REM=inicijalizacija jedinice matr.B
110 B(I,J)=0:IF I=J THEN B(I,J)=1
120 NEXT J
130 NEXT I
140 D=1:IE=1-E
150 REM=pronalazjenje maks. elementa
160 FOR K=1 TO N
170 IF K>N GOTO 330
180 IM=K:IM=A(K,K)
190 KI=K+1
200 FOR I=KI TO N
210 IF AM=A(K,I):AM=IM GOTO 230
220 IM=I:AM=A(K,I)
230 NEXT I
240 REM=zamjena mjesta redaka IM i K ako
je IM razlicito od K
250 IF IM<K GOTO 330
260 FOR J=1 TO N
270 AT=A(IM,J):BT=B(IM,J)
280 A(IM,J)=A(K,J):B(IM,J)=B(K,J)
290 AK,KJ=A(K,J):BT=BT-AK,KJ
300 NEXT J
310 D=D-1
320 REM=test singularnosti
330 IF ABS(A(K,K))<E THEN PRINT:PRINT
MATRICA JE SINGULARNA":END
340 D=AK,KJ
350 REM=djeljenje pivot retka glavnim
dijagonalnim elementom
360 AD=A(K,K)
370 FOR J=1 TO N
380 A(K,J)=A(K,J)/AD
390 B(K,J)=B(K,J)/AD
400 NEXT J
410 REM=zamjena svakog retka linearnom
kombinacijom sa pivot retkom
420 FOR I=1 TO N
430 AK=A(I,K)
440 IF I=K GOTO 490
450 FOR J=1 TO N
460 A(I,J)=A(I,J)-AK*A(K,J)
470 B(I,J)=B(I,J)-AK*B(K,J)
480 NEXT J
490 NEXT I
500 NEXT K
510 REM=ispis rezultata
520 CLS:PRINT"Element, inverzne matrice"
530 FOR J=1 TO N
540 PRINT"-----stu
pac br. ";J
550 FOR I=1 TO N
560 PRINT B(I,J)
570 NEXT I
580 NEXT J
590 PRINT:PRINT"Determinanta .A.=";D
600 PRINT:PRINT"KRAJ PROGRAMA":END
DK
RUN
BINVERZIJA I IZRACUNAVANJE DETERMINANTE
MATRICE A(NxN) METODOM ELIMINACIJE
RED MATRICE, N=7 3
Stupac 1 matrice A:
element( 1 ; 1 ):7 9
element( 2 ; 1 ):7 8
element( 3 ; 1 ):7 8
Stupac 2 matrice A:
element( 1 ; 2 ):10
element( 2 ; 2 ):7 9
element( 3 ; 2 ):7 8
Stupac 3 matrice A:
element( 1 ; 3 ):18
element( 2 ; 3 ):18
element( 3 ; 3 ):9
Element, inverzne matrice
-----stupac br. 1
-1
2
-1
-----stupac br. 2
10
-11
2
-----stupac br. 3
-10
10
-1
Determinanta .A.=-.99999997
    
```

Če označimo  $i$ -to vrstico matrice  $B^{(k)}$  s  $r_i^k$ , so operacije, potrebne, da dobimo  $B^{(k+1)}$ :

$$r_i^{k+1} = r_i^k / a_{kk}^{k+1}$$

$$r_i^{k+1} = r_i^k - a_{ik}^{k+1} r_k^k \quad (i \neq k)$$

Torej lahko elemente  $B^{(k)}$  izračunamo iz  $B^{(k+1)}$  z uporabo rekurzivnih formul:

$$b_{ij}^{k+1} = b_{ij}^k / a_{kk}^{k+1}$$

$$b_{ij}^{k+1} = b_{ij}^k - a_{ik}^{k+1} b_{kj}^k \quad (i \neq k) \quad (2)$$

Ti izrazi pomenijo osnovno rekurzivno formulo za inverzijo matrice z metodo eliminacije, pri čemer (1) reducira matriko  $A$  v  $I$ , (2) pa matriko  $B$  v  $A^{-1}$ .

## Računanje determinante

Hkrati z invertiranjem lahko izračunamo determinanto  $I$   $A$ , tako da izračunamo kumulativni produkt  $|A_k^{(k)}|$ , ( $k=1..n$ ) pri  $k$ -ti transformaciji eliminacije. Po eliminaciji bo vrednost kumulativnega produkta enaka determinanti:
$$|A| = (-1)^n a_{11} a_{22} a_{33} \dots a_{nn}^{k-1} \dots a_{nn}^{n-1}$$
 kjer je  $n$  število zamenjav vrstic pri transformaciji  $A$  v  $I$ . (To sledi iz pravil 1. in 2. transformacije determinante, glej prejšnje nadaljevanje.)

## Fivotiranje elementov matrice

Predstavljeni postopek eliminacije je zaradi velikega števila algebraških operacij precej občutljiv za nabiranje zaokrožitvenih napak. Numerična analiza postopka kaže, da je absolutna napaka približno sorazmerna  $|a_{ij}|/|a_{kk}|$ , to je absolutni vrednosti razmerja  $k$ -tega stolpca v  $i$ -ti in  $k$ -ti vrstici. Od tod sledi, da lahko poiščemo največji element v  $k$ -tem stolpcu na glavni diagonali ali pod njo (imenujemo ga  $a_{i_0 k_0}$ ), nato zamenjamo  $i$ -to in  $k$ -to vrstico in tako dobimo novo  $|a_{ij}| \geq |a_{kk}|$  za  $i > k$ . Elementi nad glavno diagonalo ne pridejo v poštev, ker prštevjanje večkratnika  $i$ -te vrstice za  $i < k$  lahko pripelje do tega, da elementi, ki so že enaki nič, postanejo različni od nič.

Vsi faktorji  $|a_{i_0 k_0}|/|a_{kk}| > k$ , bodo tako manjši od ena, to pa prispeva k zmanjšanju skupne napake rezultata. Menjavo vrstic imax in  $k$  lahko uporabimo tudi pri drugih transformacijah matrik, npr. pri reševanju sistemov linearnih enačb ali računanja lastnih vrednosti matrik; s tem nareč povečamo natančnost.

Sledi algoritem za inverzijo matrice z metodo eliminacije in s pivotiranjem.

Korak 1: Izvodi prihranka prostora spravljamo vse eliminacije matrice  $A$  v isto zvodničnoelementalno tabelo (isto velja za  $B$ ).

Korak 0: vnosi elementov  $a_{ij}$  ( $i, j=1..n$ )

inicijalizacija:

$$b_{ij} = \begin{cases} 1 & \text{za } i=j \\ 0 & \text{za } i < j \end{cases} \quad (i, j=1..n)$$

števci transformacij  $k=1$ 
  
začetna vrednost determinante  $D=1$ 
  
Korak 1: iskanje pivotnega elementa (element z največje absolutno vrednostjo v  $k$ -tem stolpcu na glavni diagonali ali pod njo). Primerjati moramo  $|a_{kk}|, |a_{k-1, k}|, |a_{k+1, k}|$ , da najdemo  $|a_{i_0 k_0}|$ , nato pa zamenjati imax- to in  $k$ -to vrstico matrice  $A$ . Tako menjavo izvedemo tudi v matriki  $B$ . Če imax  $< k$ , potem je  $D = -D$ . Element  $|a_{ij}|$  je zdaj največji v nizu ( $|a_{ij}|, |a_{i, j-1}|, |a_{ij+1}|$ ).

Korak 2: testirati moramo maksimalni element z majhnim številom eps. Če je  $|a_{ij}|$  57eps, prekinimo računanje in sporočimo, da je matrika singularna. V nasprotnem primeru naredimo naslednji korak.

Korak 3: izvedemo  $k$ -ti korak transformacije, se pravi, po vrsti izračunamo

$$D = a_{kk} + D, \quad a_{ij}, \quad b_{ij}, \quad (j=1..n) \text{ in}$$

$$a_{ij}, \quad b_{ij}, \quad (i < k; j=2..n)$$

s pomočjo formul (1) in (2)

Korak 4: testiramo števec transformacij  $k$ . Če je  $k < n$ , potem ije  $k=k+1$  in gremo na korak 1, sicer pa na naslednji korak.

Korak 5: izpis rezultatov  $B = A^{-1}$  in  $D = |A|$ .

Metoda eliminacije za hkratno računanje inverzne matrice in determinante je zelo natančna in učinkovita, pa tudi veliko prostora ne zahteva. Potrebni so približno  $n^3$  množenj in deljenj in  $2n^2$  pomnilniških lokacij. Variante te metode uporabljajo popolno pivotiranje (namesto delnega pivotiranja po stolpcih v priloženem programu). S tem včasih povečamo natančnost rezultata. Pri iniačih, kjer spravljamo elemente  $A^{-1}$  v tabelo  $A$ , približno razpolovimo potrebni pomnilniški prostor. Pri invertiranju matrik višjega reda je priporočljivo uporabiti dvojno natančnost zaradi zmanjšanja skupne zaokrožitvene napake (Priloženemu programu dodajte ukaz 25 DEFDBL A - H).

Obstajajo tudi druge metode za invertiranje matrik (Ralston & Wilf, Mathematical Methods for Digital Computers, John Wiley 1967). Kadar

je začetna matrika slabo pogojena (kadar je skoraj singularna) in/ali kadar je red matrike zelo visok, lahko uporabimo iterativno metodo za večjo natančnost ali razcep začetne matrike v podmatrike. Če je  $B_k$  dober približek za  $A^{-1}$ , lahko definiramo matriko  $R_k = B_k R_k$  še boljši približek za  $A^{-1}$ . Če vstavimo  $R_k$  v prejšnji izraz in izraz uredimo, dobimo sledečo rekurzivno formulo

$$B_{k+1} = B_k (2I - A B_k), k=0,1,2,\dots$$

Z njo lahko (v principu) poljubno povečamo natančnost rešitve. Teoretični pogoj za konvergenco je, da je norma m ali e matrike  $B_k$  manjša od ena. Avtorjevi eksperimenti pa so pokazali, da po nekaj iteracijah proces tudi proces tudi v tem primeru postane divergenten. To pomeni, da je namesto običajno predlaganega kriterija za prekinitev iteracije  $\|B_{k+1}\|, > \|B_k\|$ . Kot najboljši približek  $A^{-1}$  vzamemo  $B_k$ . Ta metoda ni najbolj ustreza za osembitne računalnike, ker zahteva dodatnih  $2n^2$  pomnilniških lokacij.

## Razstavljene matrike (bločne matrike)

Predpostavimo, da lahko dano kvadratno matriko  $n \times n$  s horizontalno in vertikalno razdelitvijo razdelimo v manjše podmatrike (bloke), tako da so podmatrike na glavni diagonali kvadratne, npr.

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} & A_{15} \\ A_{21} & A_{22} & A_{23} & A_{24} & A_{25} \\ A_{31} & A_{32} & A_{33} & A_{34} & A_{35} \\ A_{41} & A_{42} & A_{43} & A_{44} & A_{45} \\ A_{51} & A_{52} & A_{53} & A_{54} & A_{55} \end{bmatrix}$$

A lahko potem razumemo kot nadmatriko, ki ima za elemente matrike:

\* Pri nas najdemo za normo  $\|A\|$  oznako  $\|A\|_1$ , za  $\|A\|_\infty$  pa  $\|A\|^\infty$  (norma neskončno)

$$A = \left[ \begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array} \right]$$

Posebni primer razdeljenih matrik so kvazidiagonalne matrike:

$$A = \left[ \begin{array}{c|c|c} A_1 & * & 0 \\ * & * & * \\ \hline 0 & * & A_k \end{array} \right]$$

Tu so podmatrike  $A^i$  ( $i=1,k$ ) kvadratne matrike, vsi drugi elementi pa so enaki nič. Drugi posebni primer razdeljene matrike je **mejna matrika**:

$$A_n = \left[ \begin{array}{c|c} A_{n-1} & y_{n-1} \\ \hline x_{n-1} & a_{nn} \end{array} \right]$$

kjer je  $A_{n-1}$  matrika reda  $n-1$ ,  $x_{n-1}$  vrstični vektor reda  $n-1$ ,  $y_{n-1}$  stolpec reda  $n-1$ ,  $a_{nn}$  pa skalar.

Razdeljene matrike z enakimi dimenzijami in razdelitvami imenujemo **konformne**. Poudariti moramo, da za vse operacije s konformnimi (ali kompatibilnimi v primeru množenja) razdeljenimi matrikami veljajo iste pravila kot pri navadnih matrikah, če formalno obravnavamo podmatrike kot navadne elemente matrik. Posebej si bomo ogledali primer inverzije razdeljene matrike:

$$\text{Naj bo } A = \left[ \begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array} \right] \text{ i } A^{-1} = \left[ \begin{array}{c|c} B_{11} & B_{12} \\ \hline B_{21} & B_{22} \end{array} \right]$$

$$\text{Potem je } AA^{-1} = \left[ \begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array} \right] \left[ \begin{array}{c|c} B_{11} & B_{12} \\ \hline B_{21} & B_{22} \end{array} \right] \\ = I = \left[ \begin{array}{c|c} B_{11} & B_{12} \\ \hline B_{21} & B_{22} \end{array} \right] \left[ \begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array} \right] = A^{-1} A$$

Z razdelitvijo enotske matrike na štiri konforme podmatrike

$$I = \left[ \begin{array}{c|c} 1 & 0 \\ \hline 0 & 1 \end{array} \right] \text{ in množenjem razdeljenih matrik } A, A^{-1}$$

dobimo naslednje matrične enačbe:

1.  $A_{11}B_{11} + A_{12}B_{21} = I$
2.  $A_{11}B_{12} + A_{12}B_{22} = 0$
3.  $A_{21}B_{11} + A_{22}B_{21} = 0$
4.  $A_{21}B_{12} + A_{22}B_{22} = I$
5.  $B_{11}A_{11} + B_{12}A_{21} = I$
6.  $B_{11}A_{12} + B_{12}A_{22} = 0$
7.  $B_{21}A_{11} + B_{22}A_{21} = 0$
8.  $B_{21}A_{12} + B_{22}A_{22} = I$

Iz 2. sledi  $A_{11}B_{12} = -A_{12}B_{22}$ ,  $B_{12} = -A_{11}^{-1}A_{12}B_{22}$

Ko vstavimo v 4., dobimo  $-A_{21}A_{11}^{-1}A_{12}B_{22} + A_{22}B_{22} = I$ , **odkoder je**  $B_{22} = (A_{22} - A_{21}A_{11}^{-1}A_{12})^{-1}$

Iz 7. sledi  $B_{21} = -B_{22}A_{21}A_{11}^{-1}$

Če 1. z leve pomnožimo z  $A_{11}^{-1}$ , dobimo  $B_{11} + A_{11}^{-1}A_{12}B_{21} = A_{11}^{-1}$ , odkoder je  $B_{11} = A_{11}^{-1} - A_{11}^{-1}A_{12}B_{21}$

Vidimo da je za izračun  $A^{-1}$  treba obrniti le  $A_{11}$ , in matriko, ki da  $B_{22}$  drugi operaciji pa sta samo množenje in seštevanje matrik. Če je  $A_{11} = 0$  ali  $A_{12} = 0$ , so postopek preprostejši.

Zaporedje računanja je vedno  $A_{11}^{-1}$ ,  $B_{22}$ ,  $B_{21}$ ,  $B_{12}$ ,  $B_{11}$ . Posebna varianta gornjega postopka je **metoda zaporednega razcepjanja** (metoda omejevanja). Začnemo pri matriki

$$A = \left[ \begin{array}{cccc} a_{11} & \dots & a_{1n} & \\ \dots & \dots & \dots & \\ a_{n1} & \dots & a_{nn} & \end{array} \right]$$

formiramo zaporedje mejnih matrik

$$S_1 = [a_{11}], S_2 = \left[ \begin{array}{c|c} - & a_{12} \\ \hline a_{21} & a_{22} \end{array} \right], S_3 = \left[ \begin{array}{c|c} S_2 & a_{13} \\ \hline a_{31} & a_{32} & a_{33} \end{array} \right] \dots$$

$$S_n = \left[ \begin{array}{c|c} S_{n-1} & a_{1n} \\ \vdots & a_{2n} \\ \hline a_{n1} & a_{n2} & \dots & a_{n,n-1} & a_{nn} \end{array} \right]$$

Vsako matriko dobimo z razširitvijo prejšnje za eno vrstico in en stolpec. Inverzijo druge od teh matrik lahko izračunamo direktno:

$$S_2^{-1} = \frac{1}{|S_2|} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix}, \text{ gdje je } |S_2| = a_{11}a_{22} - a_{12}a_{21}$$

S substitucijo  $A_{11}^{-1} = S_2^{-1}$ ,  $A_{21} = [a_{31} \ a_{22}]$ ,  $A_{12} = \begin{bmatrix} a_{13} \\ a_{23} \end{bmatrix}$ ,  $A_{22} = [a_{33}]$

lahko izkoristimo prej izpeljane matrične enačbe, da dobimo  $S_2^{-1}$ . Z uporabo  $S_2^{-1}$  na ta način izračunamo  $S_3^{-1}$  itd. do  $S_n^{-1} = A^{-1}$ . Če je kakšna od matrik  $S_i$  pri tem postopku singularna, moramo zamenjati dve vrstici v tej matriki, da lahko računamo naprej.

## Psevdoinverzija pravokotne matrike

Če je  $A$  pravokotna matrika reda ( $m \times n$ ), definicija inverzne matrike nekoliko odstopa od svojega prvotnega pomena. Zato bomo definirali psevdoinverzijo pravokotne matrike,  $A^+$ , ki je identična normalni inverziji, ko je  $A$  kvadratna in nesingularna matrika. V splošnem primeru ima lastnosti:

$$AA^+A = A \text{ i } A^+AA^+ = A^+$$

Če je  $A$  reda ( $m \times n$ ) potem je

$$\text{za } m > n: A^+ = (A^T A)^{-1} A^T, \text{ reda } (n \times m)$$

$$\text{za } m > n: A^+ = A^H (AA^H)^{-1}, \text{ reda } (n \times m)$$

$$\text{za } m = n: A^+ = A^{-1}, \text{ Če je } A \text{ nesingularna}$$

$$\text{za } m = n: A^+ = BA^A, \text{ kjer je } B \text{ matrika katere izpolni enačbo}$$

$$BC^2 = C, C = A^+A, \text{ če je } A \text{ singularna.}$$

Tu je  $A^H$  krajsa oznaka za transpozicijo konjugirane matrike, to je  $A^H = (\bar{A})^T$ . Vidimo, da za računanje  $A^+$  za  $m < n$  potrebujemo eno transpozicijo, dve množenji in eno klasično inverzijo matrike. V primeru realne matrike je  $A^H = A^T$  in lahko za računanje uporabimo programe iz tega prejšnjega nadaljevanja.





**SPEKTROVOCII** Kompleti o 14 programi za samo 900 din + kasete (500 din). Pripravljeno su samo najbolji programi za vas.

**KOMPLET A:** Ping Pong, Friday 13th, Turbo Espirit (izredna vožnja z avionom), Spiffire 40, Visitors, Commando (nesmrtni), The Way of the Tiger 15 (5. program), Bomb Jack, Samantha Fox, Tally Turner.

**KOMPLET B:** Amazon Women (US Gold), Yabba Dabba Doo (pravi risani film), Spellbound, Swords&Sorcery, Frankenstein 2000, YU Stooli Džaka, Green Beret, Fireman, Back to the Future, F. A. Cup Football, Rupert Party, Runestones.

**KOMPLET C:** Barry McGuigan Box, Code-name Mat 2, West Bank, Movie, Blade Runner, Jet Set Willy 4, Mega Fruit, Pentagram (Ujima), Max Headroom, Costa Capers, Starline 2, Schizofrenija, Hubert, Fruity.

**KOMPLET D:** Ghosts n Goblins, Boulderdash 3, Sex Crime, Bounder, Fido, Spitting Images, Babaliba, Equinox, Kirel, Biggles 1, Biggles 2, Zombie, Spiky Harold, Pywcursor.

**KOMPLET E:** Willow Pattern, Twister, Quazatron, Hot Rasputin 2, Spindizzy, Tankals, Redhawk, William Wobler, Devil's Crown, Hocus Focas, Molecule Man, A Tic-Toc to Ride, Flying Formula, On the Edge.

Narodila pošljite na naslov: Tamara Vujišić, Lenjinska 8/II, ulica, 11080 Zemun, ali tel. (011) 552-895, Zoran. 1-5600

**PAKAR PONUJATI!** Dobro pogledajte vse igre v leti številki MM. Komplet, ki vas zanima, mirno naročite pri nas (imamo čisto vse). Naročene programe, ki se vam ne naložijo, postarujemo z novimi. En komplet stane 600 din + ptt + kas. = 1100 din. Profesionalen katalog 50 din. Navajamo znanje: K1: Ghosts n' Goblins, Boulderdash 3, Bounces, K2: Willow Pattern, Cliff Hanger, Porno... K3: Kung-Fu Master, Ace, W. C. Carnival... K4: Hjaek, Exploding Fiat 2, Snodgys... K5: Spitting Images, Babaliba, Kirel... K6: Ninja Master – komplet po želji. Pakar soft, 25. novembra 26, 15000 Šabac, tel. (015) 27-422 od 17. do 19. ure. Ne pozabite, naročite lahko čisto vse komplete! 1-5680

**SCOT** – spektromov software, ki še vedno temelji na tradiciji in kvaliteti, je še vedno tu vami. Vse programe, ki so v YU, dobite tudi pri nas. Katalog. Možna nakup posamezno in v paketih. Tuji ta mesec hit paketi: Paperboy, Mantronic, Ole Toro, Discs of Death, Rupter of the Ice Castle, Force Fighter, Figure Chess, I.C.U.P.S., Universal Hero, Turbo Runner, Superman, Mer-maid Madness, Caves of Mines, Rebel Star 2. Do izida te številke seveda še najmanj dva paketa. Prepričajte se – ne bo vam žal! Matjaz Marinšek, Presejce, Kajuhova 9, 61255 Radomlje, tel. (061) 722-750. 1-5896

**SEKS paket**, dopolnjen in šele zdaj ustrezen zaradi namo za odrasle, prodam. Matjaz Marinšek, Presejce, Kajuhova 9, 61255 Radomlje, tel. (061) 722-750. 9991

**SINCLAIRJE 1** (Formula, Paperboy, Island...), Simulatorje 2 (vojna, vesolje, helikoptar...), 1 za ZX spektrom za samo 800 din + kasete. K2, Zapravka 3, 11080 Zemun, ali tel. (011) 472-425. 1-5462

**FUTURE ORION** ima najdaljšo tradicijo in zagotavlja spektromovcem kvaliteto najpovprešjih iger v kompletu z 12 igrami (1000 din, kasete posebej). Zahvale! »SPEC-TRUM« KATALOG (100 din). Dobava takoj! Rubeticeva 7, 41000 Zagreb, tel. (041) 417-052. Pri večjih naročilih velik popust. 1-5818

**gargamel®**  
soft  
SOFTWARE FOR ZX SPECTRUM

**PRI NAS** imate na voljo vse vrste programov za ZX spektrom! Vpisujemo POKČE za nesmrtnost! Odpravljamo vse vrste začetni in izpolnjujemo preproste softwareje želje! Če imate kakšno željo, nas pokličite ali nam pišite!  
GARGAMELSOFT, Stara c. 40, 61360 Vrhnika, Tel. (061) 752-344. 1-5853

**gargamel®**  
soft  
SOFTWARE FOR ZX SPECTRUM

**PRI NAS**  
– Vsi najnoviji programi za vaš ZX spektrom!  
– Brezplačni katalog!  
– Cene zelo nizke!  
– Vrhuška kvalitata!  
Za katalog ne odlašajte! Naročite ga še danes!  
Naslov: Gargamelsoft, Stara c. 40, 61360 Vrhnika, tel. (061) 752-344. 1-5854

**CLUB M** – ZX spectrum. Ta mesec najpovprešnji programi: The Boxing, Mission Omega, Galvan, Druid, Light Force, Room Ten, Priča o kmetu, Infiltrator in Age of Asia U.S. Golds, Knock Out (Ocean), Hacker II (Activision), Čena samo 800 din za komplet. Ministar Petrovič, II. Zapravka 3, 11000 Beograd, (011) 472-420. 1-5841

**PROFESIONALCI** med amaterji – najhitrejša dobava. Tomislav Petrovič, Setefrova 10, 41000 Zagreb, tel. 323-912. 1-4789

**satan®**  
soft

**SATANSOFT** je eden od redkih, ki redno uvaja programe iz Velike Britanije. Programi, o katerih berete v drugih oglasih, že imamo ali na bode prili vsak hip. Programi, ki jih še nimamo, ne bomo nabavili, ker to počno že drugi. Vse informacije dobite na naslovu. Satansoft, Pod hrasti 8, 61000 Ljubljana. 1-5634

**SEKS KOMPLET** – ZX spectrum. Sex Crime, Sex Mission, Eat It, Strip Poker (US Gold), Strip Poker (Knight Soft), Soho Sex Quest, Samantha Fox Strip Poker, Red Lights, Violent Sex, Mouses, Fuckman, Strip Game, Dirty Movie, Samantha Fox Slide Show, 14 programov, posnetih na kaseti SCNY, skupaj s poštnino – 2500 din. Dobavni rok 1 dan. Satansoft, Pod hrasti 8, 61000 Ljubljana, tel. (061) 331-022. 1-5835

**Coyote**

**PONUJAMO** vam najnovijši turbo komplet za samo 700 din. Komplet Y5: Mindstone, Kidnap, Price of Magic, Atlantic Challenge, Black Arrow, Mafia C, Il. Knight Rider, Dan Dare, Ninja Master, Stainless Steel, 100 Km Race, Sport of Kings. Hitra dobava! Reklamacije upoštevamo! Vsek dan novi programi! Tomaž Leskovec, Drapinska 17, 63000 Celje, tel. (063) 32-693 po 14. uri. 1-5821

**NOBOMET-ŠODARKA**. Najbolje simulacije teh dveh igrarjev za ZX spektrom za samo 800 din + kasete. Club M, Zapravka 3, 11000 Beograd, (011) 472-420. 1-5843

**ŠAR-ŠAR-ŠAR**, 19 izbranih šahovskih programov za ZX spektrom za samo 1000 din + kasete. Club M, Zapravka 3, 11000 Beograd, (011) 472-420. 1-5844

**EKSKLUZIVNO!** Najnoviji kompleti za vaš spektrom. Komplet A: T. Racer, Liridium, Shogun, Paperboy, Superman, Mermaid Madness, A.C.E., Force Fighter, 100 Km Race, Black Arrow, Mafia Contract 2, Komplet B: Dark Scepter, Psi Chess, Light Force, Dynamite Dan II, Dan Dare, Ninja Master, Knight Rider, Hero, SCUPS, Bag Runner, Mantronic, Figure Chess... Cena 1 kompleta 550 din – obeh pa samo 1000 din. Zoran Vasić, Radnička 27, 15000 Šabac, tel. (015) 29-036. 1-5879



**NAJNOVEJŠE, NAJHITREJŠE, NAJVEŠE,** najkvalitetnejše (ni po JUS-u) \* katalog, tuje kasete \*

**EDDIE SOFT**  
Bristol 317  
64000 Kranj  
1-5866

Tel. (041) 36-173.

COMPUTER SHOP \* \* \* COMPUTER SHOP

**NAJVEČJA IZBIRA V NAŠI DEŽELI. PO NAJUGODNEJŠIH CENAH VKLJUČNO TEHNIČNI SERVIS**

COMMODORE C 64  
COMMODORE 128  
COMMODORE 128 D  
SINCLAIR SPECTRUM PLUS  
SINCLAIR SPECTRUM QL  
AMSTRAD CPC 464 ZELEN IN KOLOR MONITOR

AMSTRAD CPC 6128 ZELEN IN KOLOR MONITOR  
DISK DRIVE COMMODORE 1541  
JOYSTICK MAGNUM »SPACE«  
PHILIPS MSX 8020  
PRINTER COMMODORE MPS 803  
PRINTER RITMAN C+ COMMODORE  
PRINTER RITMAN F+ CENTRONICS

Tiskalniki – Programska oprema (software) – drugi različni pripomočki, ki jih lahko uporabite pri vašem računalniku

UL. P. RETI 6, TRST, tel. 993940/61602













# SISTEM, KI RASTE Z VAMI

## OSEBNI RAČUNALNIK INNOTEH PC/XT 640 Kb PROGRAMSKO IN STROJNO 100% ZDRUŽLJIV Z IBM PC/XT

### VSE V ENEM KOVINSKEM OHIŠJU:

- \* mikroprocesor 8088 – 4,77 MHz
- \* osnovna plošča – 256 Kb
- \* 8 slotov – nastavkov za dodatne kartice
- \* 2 vgrajena disketna pogoña TEAC – 5,25" po 360 Kb
- \* vgrajen disketni pogon TEAC – 20 Mb
- \* kontrola delovanja obeh pogonov
- \* večfunkcijska kartica – 384 Kb z vmesniki, ura
- \* hercules monokromatska kartica
- \* dodatni hišalni sistem za drugi disketni pogon
- \* napajalnik 220 V (50 Hz) 185 VA s priključki za ves sistem

### ZUNANJE ENOTE, DODATKI:

- \* AT look tipkovnica z lučsignalnimi znaki – 99 tipk
- \* 12" RGB monitor JVC zelene barve – 22 MHz
- \* operacijski sistem MS DOS 3.1 s priložnikom
- \* licenčni BIOS
- \* kabel za povezavo s tiskalnikom
- \* navodila za uporabo

### MOŽNOSTI RAZŠIRITVE OSNOVNEGA SISTEMA ALI OBSTOJEČE IBM OPREME:

- \* 14" barvni monitor visoke ločljivosti MITSUBISHI – 860.000 din
- \* barvna kartica – 220.000 din
- \* video monokromatska kartica – 390.000 din
- \* turbo osnovna plošča – 1.390.000 din
- \* turbo kartica 1.430.000 din
- \* SN SD CLA kartica – 1.290.000 din
- \* koprocesor 7 MHz – 670.000 din
- \* dodatni disketni pogon 20 Mb – vgradjiv v ohišje – 1.872.900 din
- \* hard disc controller<sup>™</sup> – 474.150 din
- \* večfunkcijska kartica 384 Kb – 468.460 din
- \* BACK-UP TAPE STREAMER 20 Mb – zavarovanje baze podatkov – 2.900.000 din
- \* programska oprema za dinarje: DATA BASE II, III in III+; operacijski sistemi: IBM PC DOS 3.1, MS DOS 3.1, TOP VIEW MULTITASKING & MULTIPROCESSING, GEM (kompilert), XENIX PACKAGE; spread sheet: LOTUS 1-2-3, SYMPHONY, FRAMEWORK, MULTIPLAN; obdelava besedil in podatkov: WORD STAR, WORD STAR 2000+, WORD, WORD PERFECT, BORLAND LINE...
- \* in možnost neposredne uporabe 2,5 milijona IBM programov!



**vse to  
za 5,500.000 din!**

Enoletno jamstvo, servis zagotovljen,  
rok dobave 45 dni  
Dokonačna cena na dan dobave

INNOTEH

**M** mladinska knjiga  
knjižarne in papirnice

Za nakup in informacije se oglašite na naslov:  
**MLADINSKA KNJIGA KIP, GROSISTIČNI ODDELEK,**  
Tilova 3, Ljubljana, tel. (061) 215-358, ali neposredno v  
knjižarnah in papirnicah Mladinske knjige v Ljubljani,  
Mariboru, Celju, Ptuj, Novem mestu, Zagorju ob Savi,  
Tilovem Velenju, Slovenjem Gradcu, Kranju, Tolminu  
in drugod po Sloveniji ter v Zagrebu.







# Ni miru na policah...

ZIG TURK

Dr. Jernej Kozak: Od računalna do urejanja besedil, Državna založba Slovenije, 1986, 190 strani, broširano, 3210 din.

**K**njiga je prva iz serije izdaj, s katerimi namerava DZS popestriti police z računalniško literaturo. Kar nekako logično je, da so za začetek izbrali bolj splošno zastavljeno delo, ki naravnica seznanja z osnovami računalništva.

Razdelimo jo lahko v tri glavne dele: zgodovina, strojna oprema in programska oprema. Težične knjige je seveda na slednjem, saj



je pričakovati, da bo večina bralcev iz armade uporabnikov programov. O zgodovini računalništva pišejo skoraj v vsaki podobni knjigi in nič ni narobe, če preberete tudi ustrezno poglavje v tej knjigi. Spoznali boste, da računalnik ni tako »globok in skrivnosten stroj«, kot ga je leta 1966 označil angleški novinar Snow. Osnove hardvera so razložene dovolj nazorno, da jih bodo razumeli vsi, ki imajo vsaj malo tehnične predstave. V poglavju o predstavitvi informacij ni govor le o raznih številskih sistemih, ampak tudi o tem, kako si računalniki predstavljajo besedila, slike, zvok... Računalniki so namenjeni za reševanje problemov (tudi sekira rabi za to, le da so problemi tam čisto druge vrste). Kako pripravimo računalnik do tega, da bo naredil, kar želimo, temu je posvečena dobra tretjina knjige. Avtor ne uči nobene programske jezika, pač pa razlaga, kako definiramo problem, ki bi ga z računalnikom radi rešili, kako se lotimo izdelave al-

goritma in končno algoritem tudi zapiše. Sicer pa se knjiga ukvarja z uporabo že narejene programske opreme. Avtor razloži, kaj bi operacijski sistem in kako uporabljamo njegove osnovne funkcije, potem pa nas seznanja še z urejevalniki besedil, predvsem z urejevalnikom WordStar. To je pač aplikacija, ki je večini tipkajočih ljudi najbližja. Zanji deli knjige dišijo po MS-DOS, vendar to še zdaleč ne pomeni, da priročnika za ta operacijski sistem ne boste potrebovali.

Če bi bilo v knjigi napisanega še kaj malega več o aplikativni programski opremi (preglednice, baze podatkov in ne le urejevalniki besedil), bi bila knjiga kot nalašč za vse tiste, ki se na delovnem mestu spogledujejo z računalnikom. Tako pa prinaša pravzaprav malo novega. Za programerje je predvsem koristen srednji del, a tudi tisti najbolj nedeljski se bodo morali nekega konkretnega programskega jezika šele naučiti. Knjiga je zelo splošna in v njej se boste naučili malo ali nič takega, kar bi lahko takoj uporabili in preizkusili z računalnikom. Če pa boste knjigo prebrali, boste kasneje veliko lažje razumeli priročnike za razne programske pakete in učbenike za programske jezike. Znalci se boste pogovorili z ljudmi, ki naj bi vam napisali potrebne programe, vaše želje bodo bolj v skladu z realnimi možnostmi računalnikov na današnji stopnji razvoja.

**Kupite, če želite v eni knjigi zvedeti vse o računalnikih, kar od nekoga, ki se na računalnike ne spozna, pričakujemo.**

Veljko Spasić, Dušan Veljković: Basic za mikroročunalne commodore 64; NIRO Tehniška knjiga i Zavod za izdavanje učbenika 1985, 202 strani, broširano, 1250 din.

**K**njig za C-64 je več in več in še kar izhajajo. Škoda, da se naslovi tako simpatično ponavljajo. Če pa je še kje kdo, ki bi se rad naučil programirati C-64 v basicu, potem naj prebere tole oceno.

Začne se z osnovami, t. j. zaslonskim urejevalnikom, interpretiranjem, neposrednim načinom, v prvem delu pa spoznamo še vse osnovne ukaze in funkcije basica za C-64. Zahtevnejše komande (DIM, READ, DATA, RESTORE, GOSUB, PEEK, POKE...) so zbrane v naslednjem poglavju. Ostali deli knjige je posvečen zunanji notam. Najprej kasetofon, disketna enota, tiskalniki, potem še grafika in



zvok. Avtorja imata o bralcu, predvsem na začetku, morda nekoliko boljše mnenje kot kakšen drug pisc priročnika za basic. To lahko razumete tudi tako, da stvari niso razložene začetniško preprosto kot kje drugje.

Za pametnega dovolj, za radovednega premašo, bi lahko označili knjižico v celoti. Basic je razložen in zapisan dovolj natančno in pregledno, da je knjiga lahko edini učbenik in priročnik za to različico basica. Ambicioznejši programerji pa bodo posegli po čem drugem.

**Zakaj kupiti ravno to knjigo o basicu za 64 in ne kakšne druge, o tem boste morali odločiti sami.**

Zdravko Dovedan: BASIC... jezik i programiranje; Ljubljana, ZOTKS, 1986, 398 strani, broširano, 3900 din.

**N**ikoli si ne bi bil mislil, da je mogoče napisati skoraj 400 strani dolge knjigo o basicu. A če se stvari lotite sistematično in pregledno in v več nivojih, dodate uporabne naloge in poveste še kaj o tehniki programiranja...

Knjiga ima pet delov: splošno o računalnikih in računalništvu, trije deli basica v treh različnih nivojih in končno tehnika programiranja. Splošni del je tak, kot v vseh knjigah za začetnike: hardver, softver, jeziki... Basic je razložen v treh nivojih. V najnižjem s primeri razloži, kako kodiramo osnovne operacije, formule, IF in GOTO... Drugi nivo zahteva tudi zahtevnejšega uporabnika. Bralac se mora naučiti, kaj je leksična struktura, sintaksa in semantika in potem, oborožen s tem preciznim orodjem, krene nad funkcije in ukaze v basicu. V tretje gre rado in tako se z vso rotopojto srečamo še na tretjem nivoju. Če smo npr. v prvem obravnavali stavek IF-THEN kar tako, v drugem stavek IF-THEN S sintakso in se-

mantiko, je v zadnjem poglavju govor o IF-THEN-ELSE s sintakso in semantiko. Šele tretjega nivoja so vredni ON-GOTO, WHILE ter ukazji in funkcije za delo z datotekami. Škoda, da ni ostalo pri dveh delih. Če nas avtor v prvih delih knjige uči basica, nas v zadnjem delu uči programirati. Poglavje je, vsaj v primerjavi z drugimi knjigami o basicu, izjemno kvalitetno, avtor vas bo med drugim rešil problemov pri sortiranju in iskanju.

Knjiga je pisana izredno precizno, zato se avtor ni mogel odločiti za péle-méle raznih basicov, ampak je opisal Microsoftov basic, ki je pri osebnih računalnikih najbolj razširjen. Basic je basic in napisano bolj ali manj velja tudi za druge dialekte, vprašanje pa je, če držijo tudi vsi diagrami sintakse, ki jih je v knjigi vse polno.



Knjiga je brez dvoma zelo kvalitetna in če uporabljate PC in nameravate zelo resno razvijati programsko opremo zanj kar v basicu, potem je to knjiga za vas. Priporočam jo tudi vsem, ki so imeli zaradi uporabe basica manj vrednostne komplekse pred kolegi, ki so uporabljali »resne« jezike. Pokažite jim to knjigo, in spoznali bodo, da je tudi basic »resen«, če ga le zagrabite z zadostno mero akademskega besednjaka in železniškimi diagrami sintakse.

**Kupite, če se učite MS-Basic.**

Janez Jereb: Osnove programiranja commodore 64; Tehniška založba Slovenije, 1985, 182 strani, broširano, 2535 din.

**T**udi to je knjiga, ki vas uči osnov programiranja z mikroročunalnikom commodore 64. Je večjega formata od tiste zgornje in preglednejše tiskane, pa napravi zato na prvi pogled boljši vtis.



# UVAŽAMO IZ TAJVANA SESTAVLJIVE RAČUNALNIKE IBM \*

## NUDIMO:

- X T compatible IBM 100% z 2 drive 360 KB i 10 MB H. D.
- A T compatible IBM 100% z 1 drive 1.2 KB i 20 MB H. D.
- enobarvne monitorje
- barvne monitorje
- japonske tiskalnike najboljših proizvajalcev
- video programe, večnamenske tiskalnike
- dodatno opremo za računalnike: floppy disk 8SD 48 TPI in DSD 48 TPI

## ROCCO IMP-EXP COMPUTER DIVISION

Ul. Rossetti 68 - Trst - Tel: 993940/775628

IBM je zaščiteno ime INTERNATIONAL BUSINESS MACHINES.

Na začetku se seznanimo z osnovnimi prijemi na C-64. Spoznamo tipkovnico, naučimo se jo uporabljati v neposrednem načinu, ko v komandno vrstico vpišemo ukaz za ukazom. V naslednjem poglavju napišemo naš prvi program. Poudarek poglavja je ta, da se naučimo uporabljati orodja, ki jih za pisanje programov potrebujemo: urejevalnik, ukaze za shranjevanje na kaseto ali disketo...

Basic V2.0 ni posebno bogata različica tega jezika. Vse ukaze in vdelane funkcije avtor razloži v pospešenem tempu na desetih straneh kar v obliki tabele. Če znate kakšen drug basic, bi ta tabela lahko zadostovala za vse, če bi vsebovala še opozorila pred posebnostmi basica V2.0. Naslednja poglavja so namenjena natančnemu spoznavanju posameznih skupin ukazov. Pohvalno je, da avtor ne govori samo o ukazih, ampak s primeri kaže različne tehnike programiranja in možnosti uporabe, ki začetniku niso vedno takoj jasne. Ukazom za spreminjanje koda programa in vsem, kar je s tem v zvezi, namenja skoraj trideset strani, ki jih zapolnjujejo raznorazne kombinacije stavkov IF in GOTO, ne vedno najbolj pregledne. Podobno so obdelani še nizi, tabele, izpisovanje na zaslono, sortiranje in iskanje, datoteke in grafika. Čisto na koncu je še nekaj besed o vključevanju programov v strojnem jeziku.

Avtorjev pristop, da se ni dolgo zadržal ob vsakem ukazu, ampak jih je podal na začetku kratko in pregledno, je vsekakor zanimiv. Predvsem se je s tem lahko izognil muk, ki so znane tudi avtorjem prvih beril, namreč sestavljanje primerov iz samih znanih črk. Problemsko orientirana poglavja lahko uporabimo tudi kot prilo-

nik. Le redko se kdo vpraša, kako uporabiti npr. funkcijo LEFTS. Če pa se v vpraša, kako deliti mogoč odgovor ob istem vprašanju poglavju.

**Kupite, če še kar ne znate basica za C-64.**

KOMPJUTER  
BIBLIOTEKA

1. **COMMODORE 128 - PRIROČNIK** - Knjiga podrobno razloži delo v vseh treh načinih: C 64, C 128 in CP/M. Preverite, zakaj ima to knjigo vsak lastnik C 128. Cena. 2.500 din.
2. **NAVODILO ZA DISK 1570/1571** - Podrobno razloženo delo z diskom s kopico primerov. Cena 2.000 din.
3. **COMMODORE 128 - PROGRAMERSKI VODNIK** - Za tiste, ki hočejo več. Berite na istih straneh o periferiji, arhitekturi, strojnem programiranju, lokacijah, izid vsak hip. Cena 3000 din.
4. **CP/M PLUS** - Podrobno razloženo delo s tem vse bolj popularnim sistemom. Številne tabele, primeri. Cena 2500 din.
5. **COMMODORE 64 - POMNILNIŠKE LOKACIJE** - Prisilite svoj računalnik, da bo delal, kar želite. S spoznavanjem sieherne lokacije boste spoznali dušo svojega računalnika. Cena 2500 din.
6. **COMMODORE 64/128 TEČAJ PROGRAMIRANJA V ZBIRNIKU** - Končno prva knjiga za programerje v strojnem jeziku V 100 poglavjih je vse posojeno. Kmali. Cena 3000 din.
7. **AMSTRAD CPC 464 PRIROČNIK** - Pôpoinoma razloženo delo z računalnikom, kopico primerov. Cena 2000 din.
8. **AMSTRAD CPC 6128** - V knjigi je podrobno opisano delo v basisu, logo, AMS-DOS, CP/M in še veliko drugega. Cena 3000 din.

VSE KNJIGE SO V KAKOVOSTNEM TISKU, PLATNICE SO PLASTIFICIRANE, TRDA VEZAVA.

Naročam knjige ime in priimek \_\_\_\_\_  
1 2 3 4 5 6 7 8 Ulica in številka \_\_\_\_\_  
(obkrožite št.) Kraj \_\_\_\_\_

KOMPJUTER BIBLIOTEKA  
FILIPA FILIPOVIČA 41  
32000 ČAČAK ali telefon 032-31-20

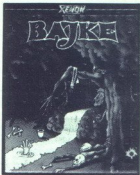
## Kako zaslužiti denar s programi?

1. Prodati program v Veliko Britanijo.
2. Prodajati piratske kopije preko malih oglasov.
3. Kupiti programe ceneje s kuponom v reviji Moj mikro.

*Če ste v vprašalniku obkrožili točko 3, potem ste dojeli bistvo anostavnega poslovanja (beril: najhitreje obrnil denar). Lahko nam verjamete, da smo poskušali čim ceneje prodajati kasete z računalniškimi programi. Tokrat jih prodajamo po polti 10% ceneje. Denar bomo raje pustili svojim kupcem namesto profesionalnim prodajalcem.*

**Na naš naslov: Xenon, pp 60, 61110 Ljubljana, na dopisnici pošljite kupon in svoje podatke.**

**Naročite lahko za ZX spectrum in sicar:**



Bajke  
2500 - 10% - 2250 din.



Smrčki-Štrumpolvi  
2000 - 10% - 1800 din.



Eurorun  
2000 - 10% - 1800 din.

Bratci Mojega mikra  
kupujejo ceneje  
**XENON**

**sinapsa**

**IZBOLJŠAJTE SVOJ  
TELEVIDZAJTE**

**ATARI ST software** – literatura po ugodnih cenah! Telefon (042) 44-844, 45-825, 3x-57

**ATARI 800 XL** Prodaja gotovih kompletov. Delovni Vranec, P. Preradovića 36, 55300 Slava Poljega, tel. (055) 79-202. 1-5377  
**V TISKALNIKE VSEH VRST** vgrajujemo YU znake, Epson, Brother, Star, Schneider itd. Tudi v drugih nepokih. Jonas Znidarčič, Poljevska 9, 61110 Ljubljana. st-1109

**PROFESIONALNI EOP** v vrhunskih mikroskopskih poslovnih cenah! Tel. (042) 44-844, 45-825, 3x-58

**SERVISIRAM RAČUNALNIKE:** ZX spectrum, QL, amstrad, commodore 64/128, apple in servisirajo zanje. Čedo Andrejčič, Omladinskih brigad 87, tel. (011) 162-434 (od 12. do 18. ure). 11070 Novi Beograd. 1-5814

**ATARI ST.** Novi programi, novi priročniki. Cena programa 1000-2000 din. Izbirite 35 programov med 150 najboljšimi programi za 30.000 din. Za podjetja in obrtnike pišemo aplikacije. Za katalog pošljite 150 din. Srečko Bahovec, Pijačeva 31, Ljubljana, tel. (061) 312-046, st-1215  
**ATARI KE, XL.** Novi programi na disketah in kasetah. Poceni kompleti. Za katalog pošljite 150 din. Bahovec, Pijačeva 31, Ljubljana, tel. (061) 312-046. st-1216  
**BARNI MONITOR** Sony KX-14CP1 analog, digital, video, ocarjen, prodaja. Ključič (061) 312-046. st-1222

**IBM PC/XT** in kompetitivne računalniške servisiram. Poleg servisa ponujamo razvoj programske opreme za IBM PC/XT. Servisiramo tudi diskovne enote za vse računalniške Čedo Andrejčič, Omladinskih brigad 87, tel. (011) 162-434 (od 12. do 18. ure). 11070 Novi Beograd. 1-5815

**ATARI 800 XL:** 8 novih kompletov. Franjo Kolar, Sindrovska 104, 21222 Bled, tel. (021) 811-376, 1-5742

**sinapsa**

Priključevanje računalnika na zadnji strani TV sprejemnika je zelo nepraktično, kvari vtičnico, za otroke pa je neizvedljivo (posebno če je televizor v regalu). Monitorje sinapsa. Antenski kabel bo trajno vključeni strani TV sprejemnika. SINAPSJA omogoča trenutni prehod od dela z računalnikom na gledanju TV programa brez menjave priključnih kablov. Cena 2450 din po povzetju. Dragan Čelofiga, Metličke 21, 63325 Šostanj, tel. (063) 882-768, zvečer.

**ATARI SOFT-CLUB ZRENJANIN.** 600 XL/800 XL/130 KE, programi in literatura. Prodaja menjava in sodelovanje. Izjemo: tečaj za učenje basica na kasetah. Za obisk katalog pošljite 100 din. Dejan Lacmanović, Sindrovska 31/A, 23000 Zrenjanin, tel. (023) 66-879. 1-5610

**ATARI ST HARDVER.** Velika izbira naprav vrhunske kvalitete po ugodnih cenah. RAM 1 Mb, romi, TV modulator, disk 720 K, Fast Basic na modulu, literatura, programi. Telefon (056) 569-987, P. N. P. electronic, Jaretova 12, 56000 Split. 1-5730  
**MSX-MSX2-MSX-MSX2.** Velika izbira uporabnih programov in iger. Prodaja in menjava. Izdelava programov po naročilu. Podlogar, C. Tavčarja 1/6, 84270 Jesenice, tel. (064) 82-906.

**COMMODORE 64:** profesionalni prevodi: Priročnik (1300), Programmer's Reference Guide (1500), Matinsko programiranje (1500), Grafika i zvuk (1000), Matematika (1000), Disk sistemi i štampači (1000), Disk 1541 (800), Uporabna za uslužne programe: Simon's Basic (700), Praktični (800), Easy Script (500), Vizivante (600), Pascal (500), MAE (500), Help 64+ (500), Multiplan (800), Superbase (1000), Stat-64 (800), Graf 64 (800), Supergrafik (800).

**SPECTRUM:** profesionalni prevodi: Priročnik CPC 464 (nova vezana izdaja, plastičirane platnice - 2000 din), Locomotive Basic (1400), Matinsko programiranje (1400), V kompletu (4500), Navodila za D01-1 (1400), Navodila za uporabo programa: Depvac, Pascal, Masterfile, Quil, Tasword, Posamezno (800), V kompletu (2700). Vseh pet programov na kaseti (1100).  
**«KOMPUTER BIBLIOTEKA».** Bate Jankovića 79, 32000 Čačak, telefon (032) 30-34. 1-5873

# Nagradna uganka

## Rešitev nagradne uganke iz septembrske številke:

- Zaporedje:  
8 13 18 24 39 54  
ker je 3×8=24, 3×13=39, 3×18=54
- Naloga je res trapasta, pravilen odgovor je 3, ker je to edina cifra, ki ne omejuje zaprtega prostora.
- Rešitev je 0. Drugo število v vsaki vrsti dobimo tako, da izračunamo tretjo potenco prviga in potem odštejemo prvo število. Tretje število v vsaki vrsti dobimo tako, da drugo delimo z 12 in ga potem kvadriramo. Trivialno, kajne! Vseh treh nalog ni pravilno rešil nihče. Tretjo je pravilno rešil samo Aleš (nečitljiv priimek), Cesta v Laško 25, 63000 Celje.

## Nagrade dobijo:

**Slavne Gabrić,** Marksov put 2, 24000 Subotica (original igre Winter Games firme Epyx).

**Polona Zupančič,** Rodine 52, 64274 Žirovnica (original igre Kung Fu firme US Gold).

**Brane Blökar,** Kumiščeva 12, 24000 Subotica (original igre Raid Over Moscow).

**Andreja Molnar,** Kumiščeva 3, 24000 Subotica (Commodore za sva vremena, darilo Mikro knjige iz Beograda).

**Berislav Pač,** Šubičeva 5, 41000 Zagreb (C-64, Memorijske lokacije, darilo Komputer biblioteke, Čačak).

**Sašo Gavič,** Kidričeva 7, 61030 Kočevje (Spectrum priročnik, darilo Mikro knjige Beograd).



# Nova nagradna uganka

## Preračunljiv švecerc

Glavni val računalniškega thotapstva je že mimo, kakšen uporen thotapoc pa se še najde. Pepe Pošten se je v Münchnu znašel pred hudimi težavami. Ponudba velika, a kaj naj kupi, da bo čim bolj zaslužil? Priljubljen njegove stoenke je omejen na 411 litrov. Da bi se laže znašel, si je sestavil naslednje tabelo:

Predmet	Poraba prostora (l)	Zasluzek (DM)
Spectrum	10	30
C-64	15	50
Tiskalnik NL-10	41	150
Amstrad 6128	100	250
Atari 260 ST	57	200
Monitor Philips ČB	71	50
Monitor Philips Color	82	200
PC-Kopija	135	600
Kava Minas à 5 kolor	5	40

Pepe se boji smradu po kavi, zato je bo pripeljal največ 10 kg. Kaj in koliko naj Pepe nakupi, da bo zaslužil kar največ? Na dopisnico zapišite vsebino priljubljen in skupni zaslužek in rešitev do 1. 12. 1986 pošljite na naš vsem dobro znani naslov. Podelili bomo 15 računalniških nagrad, med drugim kvaliteten knjige, tuje in domače kasete in, tokrat prvič, računalniške posterje.

# InTerm

## Konfiguracija za projektiranje centralnega ogrevanja, prežračevanja in klimatizacije

InTerm združuje zanesljivost in široke možnosti uporabe osebnega računalnika APPLE serije II z domačo programsko podporo. Aplikacijo InTerm so razvili strokovnjaki za projektiranje s področja termotehnike in jo preskusili v praksi. Osnovni namen InTerm je izdelava izračunov na področju termotehnike. Programska podpora obsega izračune za:

- ni več možnosti za napake pri izračunu in prepisu
- preprosta uporaba aplikacije

Vse module aplikacije so preverili v projektivnih birojih, kjer jih že več kot eno leto uspešno uporabljajo kot edini način dela. Kakovostni in količinski skok, ki ga pomeni uporaba osebne-



PROGRAM OSGRANJE VEŠE

CELOKUPNI PROGRAMI IZVIRNI

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROJEKCIJSKI

Opis	M	W	H	Q	Pa	Pa	P	M	
1	11000	12	2	20	82	93	109P	182	127A
2	15000	24	3.5	20	17	73	100A	247	2093
3	16000	1A	6	15	24	40	84D	178	818

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

- koeficijent prehoda toplote
  - transmissijskih izgub toplote
  - grelnih teles
  - talnega ogrevanja
  - ventilacijske kanalne mreže
- Vsi ti izračuni upoštevajo veljavne standarde in predpise stroke.
- Prednosti pri uporabi InTerm:
- 3 do 5-krat krajši čas izračunov
  - izpis vseh potrebnih parametrov in sicer v obliki, potrebni za vlaganje v projektno dokumentacijo
  - veliko manj delovnega navora, ker program med izračunavanjem usmerja uporabnika

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

PROGRAMI NADARAVNA INDEKSIRANJE I/O

ga računalnika s programi InTerm med procesom projektiranja, bi mogli primerjati z napredkom, do katerega je prišlo, ko so nazivni kalkulatorji zamenjali logaritemske računalnike. Aplikacijo InTerm uporabljamo z računalnikom APPLE IIc, z dodatkom 80-kolonske kartice pa tudi z računalnikom APPLE Iie.

Proizvodnja in prodaja

**VELEBIT** OUR Informatika

**Apple** kompjutorski centar

Radouševa 3, 41000 Zagreb  
tel 041/219-915, 228-555, tlx. 21512

Apple in logotip Apple sta zaščitna znaka Apple Computer Inc.

Od početka se nam je nabralo nekakj pokov. Objavljamo vse skupaj, bralci z nasveti pa bodo prišli na vrsto prihodnjič.

Kjer ni označeno drugače, velja POKE za neskončno življenj.

**Spectrum**

Benny Hill (spec-mac)

Naloži prvi basic in popravi vrstico 20: CLEAR 24899: POKE 23797,195: RANDOMIZE USR 23760: POKE 33081,0: POKE 33082,0: RANDOMIZE USR 23800

Bobby Bearing POKE 29754,0  
Camelot Warriors POKE 53929,0  
(neranljivost) POKE 50907,195

Equinox

Za uvodnim delom v basicu natipkaj program: 10 CLEAR 65535: FOR N=65400 TO 65433: READ A: POKE N,A: NEXT N: RANDOMIZE USR 65400

20 DATA 62,255,55,221,33,0,64,17,0,250,205,86,5,62,255,55,221,33,0, 91,17,0,250,205,86,5,175,50,116, 120,243,195,12,255

Go to Hell POKE 60193,0  
(neranljivost) POKE 60253,201

Magic Carpet POKE 29530,0  
(čas) POKE 31018,0

(neranljivost) POKE 30955,50

Rambo 2 POKE 27401,0  
Spiky Harold POKE 34813,0

Starstrike II (spec-mac)

Naloži prvi basic in popravi vrstico 20: CLEAR 24899: POKE 23797,195: RANDOMIZE USR 23760: POKE 33696,0: RANDOMIZE USR 23800: RUN (ENTER)

Tapper POKE 33233,0  
Wild West Hero POKE 30092,0

(neranlj. - predmeti) POKE 55668,50  
(neranlj. - sovražniki) POKE 55930,50

**Ervin Kostelec,**  
Ul. Narodne zaštite 2, 61113 Ljubljana

Action Reflex POKE 50770  
(50771,0, 50772,0,  
50964,0, 50965,0, 50966,0)

Alien Highway POKE 39412,201

Babaliba POKE 56749,0  
(bombe) POKE 49732,0  
(čas) POKE 49318,0

Batman POKE 36798,0

Bounder POKE 36610,0

Cauldron POKE 52974,0

Conquest POKE 38577,60

Dynamite Dan POKE 29002,182  
(sovražniki) POKE 29544,201  
(dr. Blitzen) POKE 33170,201

Heavy on the Magick POKE 33222,24

Incredible Shr. Fireman POKE 60217,167

(sovražniki) POKE 59876,0

Jack the Nipper POKE 44325,4

Kidnap POKE 40084,255

Knight Tyme POKE 24584,255  
(24585,255, 45322,255, 45323,255,  
41455,0, 41456,0)

Pyrauruse POKE 33446,201

Spindizzy POKE 51398,0  
(51399,0, 51400,0, 51401,0,  
51402,0, 51403,0, 51404,0)

Turbo Espirit POKE 29839,0

Who Dares Wins II POKE 50833,183  
(bombe) POKE 51847,183

Willow Pattern POKE 47276,201  
**Jože Janžakovič,**  
Kozinova 11, 61000 Ljubljana

Ali Baba

Po naslovni sliki vpišite program:  
1 FOR N=23296 TO 23328: READ A: POKE  
N,A: NEXT N

2 RANDOMIZE USR 23296

3 DATA 221,33,0,128,17,0,50,62, 255,  
55,205,86,5,221,33,106,224,17,74,26,  
62,255,55,205,86,5,175,50,167, 250,  
195,106,224

C.O.R.E.

Del v basicu naložite z MERGE, potem pa  
napišite POKE 23797,195: RUN. Ko računalnik  
naloži sliko in dva bloka, vpišite:

POKE 40408,20: RANDOMIZE USR 23800.

Poženite kaselofon in naložite ostaneke  
programa.

Twister POKE 42411,62: POKE42412,96

Yie Ar Kung-fu (verzija Satansoft)

Po sliki natipkajte:  
1 FOR F=65200 TO 65212: READ A: POKE  
F,A: NEXT F

2 DATA 221,33,0,91,17,0,161,62,255,  
55,195,86,5

Ko se program naloži, natipkajte: POKE  
29587,167 (nesmrtnost), POKE 58355,0  
(nasprotniki vas ne bodo napadali). Program  
poženite z RANDOMIZE USR 27264.

**Miroslav Milošević,**  
76321 Zagoni (Kovačič)

**Commodore 64**

Arc of Yesod POKE 23651,234:  
POKE 23652,234

Boulderdash III POKE 16494,234:  
POKE 16495,234

Crystal Castle POKE 5372,234:  
POKE 5373,160: POKE 5374,8

Cuthbert Tombs (vr.) POKE 9687,234:  
POKE 9688,234

POKE 13202,234: POKE 13203,234

Starquake POKE 11625,234:  
POKE 11626,234

(municija) POKE 4568,234:  
POKE 4569,234

(podlage) POKE 17965,169:  
POKE 17966,120

**Braslav Erpačić,**  
VI. Nazora 8, 43404 Bušetina

Bandits POKE 4759,169

Bruce Lee POKE 7462,165

Commando POKE 2409,173:  
POKE 4854,173

Everyone's a Wally POKE 34461,157

Frak '64 POKE 22048,173

Grog's Revenge POKE 23608,173

Herbert's Dummy Run POKE 4306,165:  
Kung Fu Master POKE 38649,189

Lazy Jones POKE 4251,173

Skool Daze POKE 7553,165:  
POKE 7623,165

Space Taxi POKE 16911,200

Zaxxon POKE 28400,173

**Martin Milinković,**

Žigovcova 5, 41000 Zagreb

**C 16, C 116, plus/4**

Air Wolf POKE5918,100:POKE11680,100

Berks I POKE4468,X

BMX Racers POKE4338,X

Cave Fighter (čas) POKE4184,15

Cuthbert in Space POKE4510,X:SYS4352

Kung-Fu Attack POKE6964,X

Kung-Fu Kid POKE11013,X:SYS8192

Major Blink POKE4471,X

Pacmania POKE 6678,X

Space Fortress POKE14978,X

**Vladimir Turjančin,**

Stevana Mokranjca 8, 78000 Banja Luka

Big Mac POKE12710,255:SYS7000

Exorcist (100 ž.) POKE16214,1:SYS4576

Raffles POKE13516,0:SYS10980

Xargon Wars POKE7302,255:SYS7296

**Marko Hren,**  
Na Korošci 30, 61117 Ljubljana

Commando

Ko se igra naloži, se postavi v monitor  
(SHIFT + reset) in vpiše: 6 134F.

Dark Tower A1FE2 J1P\$1FF0, RETURN  
(2x) 6 1E40 RETURN

Za prehod v naslednji del vpišite 6 201E  
(RETURN) in vključite kaselofon. Če se računalnik  
po nalaganju zablokira, ga resetirajte in poženite z 6 1E40 (RETURN).

**Dejan Ravlić,**  
R. Krstiča 35/1, 37240 Trstenik

To stran smo naredili z računalnikom macintosh,  
programom YU-MacWrite

in tiskalnikom imagewriter.



## Heavy on the Magick

**Tip:** pustolovščina  
**Računalnik:** spectrum 48/128 K, 64/128, amstrad  
**Format:** kaseta  
**Cena:** 9,95 funta  
**Založnik:** Gargoyle Games Ltd, 74 King Street, Dudley, West Midlands DY2 2 OB  
**Povezete:** čarjanje vredno oglada  
**Ocene:** 10/10

### LEON GRABENŠEK

**H**eavy on the Magick kombinira rafiniranost in vnos teksta, ki ga poznamo iz pustolovščin, z animirano grafiko in dogajanjem v realnem času, ki sta značilnosti akcijskih iger. Vsakdo se mora strinjati, da gre tu za najbolj razburljivo zamisel, odkar so se prikazali prvi tvorstveni poskusi (Shadownfire, Lords of Midnight).

V igri je glavna oseba Axil the Able, ki se je kot čarovnik začel nikati znanstvenega gradnja, polnem pošasti, urokov in nevarnosti. Rad bi pobegnili iz gradu skozi anega izmed treh izhodov. Na tej težki poti pa bo moral Axil reševati dodatne naloge, ker sicer nikoli ne bo postal mojster čaranja.

Pri izpolnjevanju naloge ti bo kot Axilu zelo koristila tudi prastara knjiga Grimoire z napotki za čaranja. Vendar je po čudnem naključju nekdo iz nje padel vsežive listov, ki jih v zivirniku ni in so velikanskega pomena za poznejše faze igre.

Nadsvje uporaben je tvoj prijatelj, velikani Apex. Bodi previden, ne vznemirjaj ga s čarovnijo «BLAST», ker boš takoj mrtev. Če pa boš z njim ravnal spoštljivo, ti bo dal napotke za rešitev nekaterih ugan. Včasih ti s svojim velikim telesom zapira izhod iz sobe. Zahvali se mu za pomoč («APEX, THANKS») in zgini bo. Ena izmed najmočnejših strani igre je dialeg Najprej (za narekovanje) navedeš ime osebe, s katero se bi rad pogovarjal, zatem pa stvar, o kateri hočeš informacijo.

Tako («APEX, DOOR» p meni «Apex, bi mi, prosim povedal kaj o tleh vrata?»).

Če imaš že pripravljen načrt, kam boš šel in kaj boš delal, lahko obideš izpise lokacij in skrajšaš čas tipkanja: zapored vneses več ukazov, ločenih z vejicami izvajanje teh ukazov se samodejno prekine, kadar se prikaže kakšna pošast ali če natipkaš «HALT». Ta ukaz ustavi katerokoli Axilovo akcijo (tudi napad).

Ukaz «OPTIONS» te vrne v glavni menu, kjer lahko posnameš igro (razen če si v družbi s katero izmed pošastí).

Celotna slika je povečana, zato je nekatere predmete težko prepoznati. Pomagaš si tako, da natipkaš «EXAMINE OBJECT». Axil bo samodejno prejel predmet, ki mu bo najbolj všeč. Ukaz «EXAMINE» je zelo močan: predmet ti ne samo opiše, ampak ti tudi o A pove, kje in kako se uporabi. Kadar pa na zaslonu več predmetov, sta na voljo ukaza «LEFT» in «RIGHT» Axila premakneta na levo ali desno stran sobe, hkrati pa se spremeni razporeditev predmetov.

Za nekatere ukaz moraš imeti Grimoire. Z ukazom «CALL» pokličeš katerokoli osebo iz igre na svojo lokacijo. Tako imaš vselej pri roki dobrodošnega Apexa. Če se boš pri raziskovanju preveč obriral, bo začela ena izmed oznak za izhode z lokacije utripati. To pomeni, da se ti iz te smeri bližja pošast, ki je po navadi sovražno razpoložena. Lahko poskušaš pobegeti drugam ali pa pogumno uporabiš čarovnijo «BLAST». Z nekaj zaporednimi klci te čarovnija (vedeti moraš tudi ime pošasti) ubiješ sovražnika.

Čarovnijo «FREEZE» boš verjetno uporabljal bolj redko, ker ne učinkuje povsod, vzame ti doberšen del energije in je aktivira zelo kratak čas. Ta čarovnija zamrzne sovražnika in ti omogoči, da se zmuzneš mimo. Z njo tudi izničiš urok, s katerim so zaščiteni nekateri predmeti. S čarovnijo «TRANSFUSION» preliješ tokče za svojo izkušnost (EXPERIENCE POINTS) v energijo.

Ko se ti zatakne, se spomni še najbolj kompleksne čarovnije, «NVOKE»: Z njo prikličiš demone. Če nečeš, da te demon vže v peč (furnace), moraš na tla položiti amulet. Vsak demon ima posebnosti.

— ASTAROT Obrzdaš ga z amuletom «SWORD» Demon te prenese na katerokoli lokacijo, samo njeno ime moraš navesti (npr. «ASTAROT, WOLFDORP»). Dobro premisli, kam se boš transportiral! Amulet moraš namreč pustiti na prejšnji lokaciji — kaj lahko se ti zgodi, da prehoda najazi ni, in igra bo končana.

— BELEZBAR Amulet je «MANTIS». Demon ti pove značilnosti predmeta, ki jih samo z ukazom «EXAMINE» ne zveš. Predmet spustiš na tla in natipkaš «BELEZBAR, predmet».

— MAGOT. Zanj potrebuješ amulet «SUNFLOWER». Demon ima zemljevid glavo in ti bo povedal, na katerem območju je predmet, po katerem sprašuješ («MARGOT, predmet»).

— ASMODEE. Amulet in demonomvši sposobnosti ne poznaj Poskus!

Ponekod ti bodo pot zapirala zaklenjena vrata s strážarjema. Če boš natipkal «GUARDS, DOOR», ti bosta zastavila uganjo. Rešitev uganke je hkrati geslo za odpiranje.

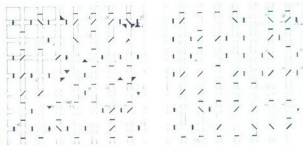
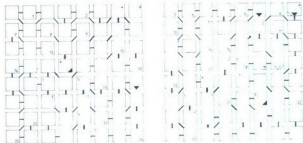
### Karta: JONA JAVORŠEK

#### Legenda:

- bel trikotnik — prehod gor
- črn trikotnik — prehod dol
- prekinjena črta — ni prehoda
- 1 Znak — strelac, 2 Jar of Chrome, 2 Jar of Honey, 3 Pellet — štiti ga urok, 4 Znak — ovan, 5 Znak — bronza, 5 Znak — kozorog, 6 Nest of Phoenix, 7 Bag of gold, 8 Sword, 9 Bag of gold, 10 Mantis, 11 Honey jar, 12 Bag of gold, Garlic, Foot, 13 Znak — skorpion, 14 Key of zinc, 14 Rock snake dead coil, 15 Clasp — Salamander charm, 16 Znak — rak, 17 Leaf of bread, 18 Mirror, 19 Znak — vodnar, 20 Shell, 20 Shell, 20 Jar of hemlock, 21 Jar of honey, 21 Znak — lahnica, 22 Key of brass, 22 Scroll (Transfusion), 23 Slat, Love Pill, 24 Nest of Phoenix, 25 Flask, Cake, 26 Poison Smeared Rock, 27 Cauldron of Cold Iron, Scroll, 28 Snake — Iron clasp, 29 Znak — dvojčka, 30 Key of lithic, 30 Znak — devica, 31 Skull, 32 Poison smeared head, 33 Egg — zaščiten z urokom, 34 Rib, 35 Znak — bik, 36 Key of iron, 36 Nugget, zaščiten z urokom, 37 Thigh, 38 Znak — rib, 39 Key of copper, 39 Jar, 40 Scroll (Call), 41 Ulna, 42 Znak — Delaj po ukazih, 43 Grimoire, Poison Smeared Rock, 44 Bag, Bag of Gold, 45 Znak — lav, 46 Key of nickel, 44 Bag of gold, 47 Zebble — Disguised Erlstone, 48 Ball of copper, 49 Loaf of bread, 50 Sunflower, 51 Reby, 52 Leaf, Bag of gold.

#### Oznake območij:

A — Wolford	P — Secunda Porta	I — Tertia Porta
B — Trollwind	R — Worming	J — Eye Of Heaven
C — Morfang	S — Sothic Complex	K — Wratviale
D — Mioua	T — The Chasm	L — Methos
E — Forburg	U — The Pit	M — Silyome
F — Kilchen Of Ai	V — Quadra Porta	N — Doubt Of Rabak
G — Rook Of Hydra	Z — Lichgate	O — Nainic Complex
H — Pilefoot	A — Wolford	P — Secunda Porta
I — Tertia Porta	B — Trollwind	R — Worming
J — Eye Of Heaven	C — Morfang	S — Sothic Complex
K — Wratviale	D — Mioua	T — The Chasm
L — Methos	E — Forburg	U — The Pit
M — Silyome	F — Kilchen Of Ai	V — Quadra Porta
N — Doubt Of Rabak	G — Rook Of Hydra	Z — Lichgate
O — Nainic Complex	H — Pilefoot	



nje vrata («DOOR, geslo») Nekate- ra druga vrata se odprejo, ko polo- žiš na mizo ustrezne predmet Kateri je to, ti povesta znak na steni ali me sobe. Prečrtan krog (NO ENTRY SING) pomeni, da skozi ta vrata ni prehoda. Za vrata z »zvenekajočim« znakom (TOLL SIGN) je rešitev možna zlata (bag of gold)

Za tretji tip vrata potrebujemo klju- če. V sobi, kjer najdeš ključe, pog- ledaj znak zodiaka na steni. Značil- nost znaka ti je treba samo pove- zati z imenom sobe na zemljevidu in je zmaj ključ za ta vrata. Pove- zave so naslednje: Oven-Flox, Bik-Horns, Dvojčica-Turo, Devica- Purity, Skorpion-Stings, Strelec- Arrows, Kozorog-Nani, Vodnar- Rains, Ribič-Ichty, Lev-Pride, Teh- nica-Scales, Rak-Claws

Zdaj pa k navodilom za igranje! Poberi knjigo magije Grimoire in se odpravi iskati manjkoče li- ste. Naprej poišči tistega z navodi- lo za čarovnjo TRANFUSION. Naslednji je na vrsti list s čarovni- jo CALL. Če jo hočeš pobirati, mor- aš odstraniti urok, ki jo ščiti (FREEZE SCROLL).

Vrni se na začetno lokacijo, nato pa tipkajte E, E, RIGHT. Poberi desno možno zlato (leva je za- strupljena). Pojdi na S in E. Poberi struno kruha (PICK UP LOAF), ki te okrepí. Zdaj si pripravljen za boj, zato pojdí na SW in uniči du- ha (BLAST GHOST). Natipkaj W, W. Spusti možno na mizo Stopi skozi odprta vrata in poberi sonič- no (SUNFLOWER). Tako boš amulet, ki te obvaruje pred jezo demona Magota

Pripravljen si za oživiljenje po- šasti AI. Pojdi na začetno lokacijo (ne pozabi vzeti možnje), nato pa E, N, N, E, RIGHT. Plačaj prehod skozi vrata N, NE, N, W, Poberi ključ E, E, S, S, N, S (ta nelinearna operacija je potrebna zaradi ma- gičnega stopništva). W, NW, uniči zombija (BLAST WRAITH) in po- beri lobanjo (SKULL). SE, S, S, S, Preišči sobo in našel boš stegno (THIGH). N, W, N, SW, S, S, poberi roko (ULNA).

Zdaj imaš vse potrebno. Ostane ti samo še pot na kraj, kjer boš oživil Ajga N, N, NE, E, N, SW, S, W, S, W, LEFT, spusti ključ na mizo. Vrata se bodo odprla in pot bo prosta. Zapeli se skozi vrata, nato pa SW, SE in prisel si na lokacijo s kotlom na sredini. Tu boš »skuhal« dele za pošast AI. Naprej vzemi zvitek pergamenta in oživilj prikáže se ti zanimivo sporočilo. Po vrsti spusti v kotel lobanjo, stegno in roko. Izpusti še čarobno besedo («CAULDRON, ACHAD») in

Vrni se na začetek. Priskrbi si ljubezensko tableto (LOVE PILL). Z njo boš omrežil zelo veliko Picco, ki se od časa do časa prikáže na najbolj nemogočih lokacijah. Najbi njene pomoči igre ne boš nikoli končal, zato se naredi play-

boya in ji izpovej ljubezen («PIC- CA, LOVE»). Izdala ti bo nekaj po- membnega, dobil pa boš tudi mo- ve moči

Čas je, da se opremiš z orožjem. Ko boš izpolnjeval novo na- logo, boš namreč naletel na mno- go pošasti, ki jih samo z «BLAST» ne boš mogli uničiti. Volkodlaka (WEREWOLF) bo pokončno zlato zno (NUGGET), ki ga položiš na tla. Za to potrebujemo slasičico (NOUGAT) Položi jo na mizo namo- sto zlatega zrna, da te duh spusti mimo. Za vampirje potrebujemo če- sen (GARLIC), ki ga najdeš v prvem nadstropju. Protí polizem (SLUG) pomaga kroglica (PEL- LET). Dobí jo, če spustiš na nje- no mesto bakreno žogo (BALL OF COPPER). Drugače ne boš mogli iz sobe, v kateri je kroglica

Zdaj si pripravljen, da obudiš Feniksa (PHOENIX). Pojdi na start, potem pa E, N, NW. Šifra za zaklenjena vrata je »DOOR, WOLF« Prejbi se do naslednjih takih vrst N, NE, W, NE, SE, SW, E, DROP NUGGET (da se zneb- eš volkodlaka), PICK UP NUGGET, NE, E, S, S, W. Šifra za ta vrata je »DOOR, LUNACY« Poišči lupino (SHELL). Poberi jo in poišči Fe- nixovo jajce (EGG). To lahko vzameš žele, ko na njegovo mesto položiš lupino, eniš gnezdi v og- nju. Pred očino te zanjigajo ože- šk ognjenega močarča (SALA- MANDER CHARM). Jajce spusti na srednjo ogarnjo, se umakni v spoštljivo razdaljo in izusti »NEST, PHOENIX»

Spodobí se, da postaneš nekaj- lobo bolj izkušen čarovnik. Zato se odpravi na območje SECUNDA PORTA in poišči zaklenjena vrata. Stražarji ti zastavijo uganko «THE WORD IS NO WORD». Od- govor je preprost, »tišina« (SI- LENCE). Vstopi in se oglej učinek. Na poti te čaka še veliko ovir. Tu je nekaj rešitev: Čez vrezec pri- deš z »WATER, FALL«. Pošast RABAK odženeš z »RABAK, WA- TER«. Čez prepad (CHASM) pri- deš s steklenico (FLASK). Če kakšne ovire ne boš žele odstrani- ti, se lahko še vedno transportiraš naprej s pomočjo demona Astarota

In kako končati igro? Poišči meč (ni ti ga treba pobirati), pokli- čni demona Astarota in reci, »ASTAROT, PILEFOOT!« Odnese te na eno končnih lokacij, kjer ti zleti izhod iz gradu. Čaka te samo še premak na »zhod Natipkaj »DOOR, ELEVEN«. Izhodna vrata se bodo odprla. Vstopi in videl boš, da znajo tudi čarovniki plesa- ti breakdance.

Opozorilo ne poskusi uničevati stvari z »BLAST predmet, FEEZE predmet«, ker igra na skriv- nosten način zablakira.

Še nasvet, kako si pridobiš naj- večjo možno energijo (STAMINA), spretnost (SKILL) in srečo (LUCK). Na začetku poberi Grimoire. Spretnost se ti bo povečala za 5 točk. Zdaj skozi v glavni menu

in pritisni «LOAD AXIL». Nalaga- nje prekini (BREAK) in se vrni v igro. Grimoire bo spet na svojem mestu, tvoja spretnost pa bo tako kot pred nalaganjem. To poravn- lja, dokler ne dosežeš vrednost 99. V glavnem meniju pritisni

«Realign status». Vrednosti vseh treh kazalcev se bodo zamenjale. Kar tako naprej, dokler ne boš imel samih najvišjih vrednosti. Na koncu posmi igro na kaseto, da ti ne bo treba pet minut ponovljati vsega tega vsakič, ko boš igral.



## DRAGOMIR GOJKOVIČ

**N**adajevanje popularne igre Starstrike je ena boljših iger na katere sem naletel v zadnjem letu. Programerji igre Realtime Software so tokrat napravili tako dobro senčno grafiko, da ti kar vza- me sapo.

Igralec se v začetku znajde v po- možnem (podpornem) modulu (Support Module) in lahko odleti v enega od petih sončnih sistemov (označeni so z grškimi črkami), kjer obnovi svoje rezerve goriva. Na zgorjem delu zaslona vidimo čas v desnem vogalu pa rezervo goriva. V spodnjem delu so instrumenti in ko- mande modula. «Laser» kaže segre- vanje lasera (kadar je kazalec na minimumu, moramo laserje spet na- paliti in jih ohladiti), «fuel» je ozna- ka za rezervo goriva, «field» za rezervo varovalnega polja okoli naš- ega vesoljskega plovila, «control» pa kaže hitrost modula. Q pomeni gor, O-leva, A-dol, P-desno, M-ogonj.

Ko po pletu skozi hipervesolje pridedi v sončni sistem, se znajdeš v lokalnem pomožnem modulu. Tu ne moreš obnavljati rezerv goriva, pač pa imaš pred sabo popoln zemljevid tega sončnega sistema. Na zemlje- vidu so zarisane orbite planetov in kaptada sam položaj planetov (vključno s trnico pomožnega mo- dula okrog kroglice sonca). S tipki- ma za gor in dol dobiš podatke o planetu, ki ga raziskuješ. Ojaviti pa- imaš s tremi vrstami planetov.

Na poljedljskih planetih (agricul- tural) je najmanj orožja. Industrijski (industrial) so precej bolj zavaro- vani, vendar jih je moč z malice spretnosti uničiti – njihova oboroži- tev sicer ni kdove kakšna, vendar je nastavljeno veliko pasti. Vojaki pla- neti (military) pa so pravé trdnjave, ki jih je skoraj nemogoče uničiti. Cone, skozi katere se premijaš so zelo dolge in težke, napadajo te zelo agresivne ladje, pasti pa je neslu- tno veliko. Ko si vse planete dobro ogledaš, lahko izbereš tistega, ki ga

**Tip:** akcijsko-strateška igra  
**Racunalka:** spectrum 48 K, C 64, amstrad  
**Format:** kaseta  
**Cena:** 7,95 funta  
**Založnik:** Realtime Software, Prospect House, 32 Sovereign Street, Leeds LS1  
**Povzetek:** kopija Eliza s tehničnimi izboljšavami  
**Ocena:** 9/10

boš napadel. Cilj igre uničiti vse planete v vseh petih sistemih.

Naprej si oglejmo, kako napade- mo poljedelski planet. Prva bobna faza se začne že pri zrhodu iz hiper- vesolja, ko se prebijaš skozi med- planetarne meje postaje. Ovirajo te topovi z izstreki; podobnim zvez- dam, daleč plodše ki se vrtyo okrog osi, in piramide ki se vrtyo okrog plodše Piramide in izstreške lahko uniči, plodša pa se moraš samo izo- gibanjati.

Ko se prebijaš mimo postaj (na- vodno petih) se začne druga faza: To je Orbital insertion, vrtiranje. Bo- rna se proti vesoljskim ladjam ki branjajo planet, katerega napadaš. Na levi strani zaslona je radar za horizontalno navigacijo. Če hočeš odkriti sovražno ladjo, mora biti njen simbol nad tvojim Višinski radar (na desni) kaže kako visoko je sovražna ladja glede na tvoj položaj (oziroma kako nizko). Na tem radar- ju sta še simbol reševane kapsule in simbol reševane ladje (krzhec).

Sovražno ladjo moraš večkrat za- deti (kot v igri Elite). Po zadnjem zadetku se razlezi na tisoče kosov. Toda včasih se pilotu posreči, da se reši s kapsulo in takrat se pojavi lokalnem pomožnem modulu. Če uniči kapsulo s pilotom, ne dobiš nobenih točk, za- to počakaj na reševalno postajo in uniči ob plovilu. Vsega skupaj mor- aš uničiti kakih deset ladij.

Ladij je več vrst. Najpocasnejše so šilaste s poševnimi krili (Krait). Skoraj ves čas leti naravnost in jih zato ni težko zadeti. Šesterokotne ladje z ravnimi krili (ASP Mk II) so dokaj počasne, vendar zelo hitro stromoglavijo in takrat jih ne moreš sestreliti. Izredno manevrirajo, izva- jajo fantastične lupinge in bočna stromoglavjanja. Zelo so nevarne! Četveroletna ladja (Adder) je zelo hitra, šodčke naredi tako rekoč na mestu. Od vseh jih najbolj nevarna. Svetujem ti, da ladjo na zaslona ta- ko dolgo zasleduje, dokler je ne zadenesi in je ne izpreči. Ne dovoli, da bi ti izginila iz nepred očmi in ti prišja za hrbt.

Planetary Descent (spuščanje na planet) je tretja faza na poti do cilja

(reaktorja) V tej fazi letiš po predoru (kot pri Starstirke) in uničujes sračne objekte. To so rudarska postaja, balonska ovira, komandna ladja (navadno sledi predora brani ta pa jo dve topovski bateriji, ki ves čas streljata), vendar je precej počasna, dalje patrolne ladje, ki se gibljejo navpično in v cikcakasti črti, potem pa imamo opraviti še s piramidami. Slednje so pravzaprav skladišča potrebne opreme za vsako, ki jo uničiš, dobiš 1500 točk.

Naj omenim še nekatere objekte, ki jih ne moreš uničiti: topovske baterije, izstrelki (ni prijetno, ko jih kar pet ali šest krene proti tebi) in energetske postaje. Slednje so podobne piramidam. Toda na vsaki dve sekundi izstrelijo energetske žarek. Če tak žarek preletiš, bo tvoje zaščitno polje malce oslabeo.

Četrta in zadnja faza igre je predor v srce planeta. Cilj: uničiti jedrski reaktor. Ko se prebiješ skozi prehod in prides v prostor z reaktorjem se spusti malo nize. Izstrelji nekaj strelcov v reaktor (podoben je prizmi) in preidi skozi odprtino, ki se pojavja na drugi strani. Tako boš uničil reaktor in prejel sporočilo, da je planet nevtraliziran. Potem si spet zagotoviš dostop do menija: lahko izbereš, kateri planet boš sedaj napadel oziroma ali boš raje šel v svoji obojni moduli in obnovil rezerve goriva.

Scenarij za napad na industrijski planet je v bistvu enak, le da je več elementov. V prvi fazi je nekaj razlik. Še preden prides do intergalaktičnih postaj, se znajdeš pred »whorlom« v planetarni pas. Podoben je sesterokotniku, ki se vrti okrog svoje osi, na njegovih krakih pa je šest »piramidic«, ki jih moraš »odpeti«. To napravíš tako, da po vrsti streljaš nanje. Potem letiš skozi medplanetarni pas, ki pa je veliko težje od onega okrog poljedelskega planeta. Nare je prazi veliko več pasti. Začetek tretje faze je povsem enak kot pri poljedelskem planetu. Toda čez čas naladja na vhod v labirint, sestavljen iz raznih geometričnih teles vsakršnih oblik in barv, ki te močno motijo. V tem delu igre imaš na voljo tudi dva nova ukaza: streljanje + dol (pospeševanje hitrosti) M+A, streljanje + gor (upočasnitev) M+O. Hitrost prečitaš na pokazatelju, označenem s črko V.

V samem labirintu so statična telesa in telesa, ki se premikajo. Od statičnih naj omenimo zgornje, središčne in spodnje ščitnike, ovire, postavljene počez itd. Nekatera gibanja telesa se vrtijo okrog svoje osi ali pa se gibljejo po določenih tirnicah. To so razne piramide, trapezi itd. V gornjem delu zaslona vidiš, koliko točk si si nabral (bonus) čim hitreje prides skozi to cono, več točk si priboriš. Čez čas (odvisno od dolžine labirinta) prides do vrat, ki se odpirajo in zapirajo, moraš upaniti, kdaj se bodo odprla in hitro šviginiti skozi. Tako prides v prostor z reaktorjem.

Ko se v četrti fazi znajdeš v tem prostoru, si prisenečen, kajti prostor okrog tebe rotira, reaktorja pa nikjer ne vidiš! Počakaj, da bodo vrata na nasprotni strani nad tabo in se takrat spusti nize. Zagledal boš

reaktor, ki rotira skupaj z vsem prostorom. Dovolj je nekaj strelcov, da uničiš reaktor in da se v nasprotnem zidu odprejo vrata v vesolje. Ko prides skozi ta vrata, prejmeš sporočilo, da je planet uničen.

Pri napadu na vojaški planet je skoraj vse enako, le reaktor je drugačen; trika v zvezi z reaktorjem ti ne bom izdal, kajti igra ko zanimivejša, če se boš sam potrudil.



## Equinox

Tip: akcijska pustolovščina

Računalnik: spectrum 48 K, C 64, amstrad

Format: kaseta

Cena: 9,95 funta

Založnik: Mikro-Gen, 44 The Broadway, Bracknell, Berkshire RG12 1AG

Povzetek: vesoljska komunalna

Ocena: 8/9

### MIROSLAV PETROVIĆ

**S**mo v vesoljski postaji z osmimi nivoji, ki so z medvojvskim teleskopom (translevalni teleskop) povezani v celoto 128 sob. Vsak nivoj (po 16 sob) skriva nestabilno jedrsko posodo, ki jo moramo pravočasno najiti in odstraniti. To ni niti najmanj lahko, kajti čas je omejen, ovire pa vsakršne.

Če igraš s tipkovnico, ti svetujem izbiro tehih tipk: O-levo, P-desno, Q-dviganje pri letu (thrust), A-emananje predmetov (use), 1-pavza.

Na zaslonu so tale pomagala:
 

- Kvadrček v gornjem levem vogalu kaže, kateri predmet trenutno prenašaš (na začetku seveda nimaš nobenega)
- Kvadrček z življenji, v začetku imaš tri življenja, števila življenj pa nikakor ne moreš povečati, tu je takoj tudi tvoj rezultat

- Kvadrček z ostankom časa
 

- Simboli v gornjem desnem kotu kažejo, koliko energije za polet ti še ostalo in koliko imaš še orožja.

- Kvadrček z ostankom časa
 

- Simboli v gornjem desnem kotu kažejo, koliko energije za polet ti še ostalo in koliko imaš še orožja.

Predmet v obliki soda ti poveča energijo za letenje, predmet v obliki akumulatorja pa ti obnovi zalogo orožja. Ta predmeta sta enakomerno razporejena po vsem labirintu. Zato poseži po nji-

ma samo takrat, kadar ti že zmanjkuj rezerv tu so še bombe (na vsakem nivoju po ena) kadar jih aktiviraš, vsi sovražniki v hipu izginejo.

Nekaj predmetov je zelo važnih. Ključ odpirajo na vsakem od osmih nivojev po ena vrata. Ključ pa lahko uporabiš samo enkrat – ko vrata odpreš, ostanejo za vedno odprta. Dinamit (podoben navkrižno položena deska) lahko aktiviraš samo pri vходу, ki je

zasut s kamenjem. Z lopato si utreš pot do labirinta, ki ga potem lahko raziskuješ. Na vsakem nivoju so zelo važni žetoni – brez njih igre ni mogoče končati. Uporabljaš jih kot povratno karto za teleport. Nekalere sobe so namreč povezane izključno s teleskopom in zato moraš paziti, da žetone pravilno uporabljaš. Prepustnica ti omogoči prehod (zaustavi laser), našel pa jo boš na višjih nivojih. Z vrtalcem prebiješ zid in vzameš predmet, ki je za njim S čipi (1-8) pa preideš na naslednji nivo.

Poleg tega so še statični predmeti. Telepote prenašaš z ene na drugo lokacijo (najprej moraš vanj vreči žeton). Medvojvski teleskop (precej je podoben vratom) te premešča z nivoja na nivo, če imaš čip s številko nivoja, na katerega bi rad. Posojevalec (credit dispenser) ti zamenja eno od življenj za žeton, vendar prides vanj samo takrat, kadar nimaš pri sebi nobenega predmeta. V odprtino za odpadke (disposal chute) moraš zmetati vseh osem posod in tako zaključiš igro. Magnetno dvigalo (magnetic lift) te dvigne z enega ali več zaslonov, ne da bi porabil energijo za letenje. Premične stopnice pa ti olajšajo prehod skozi nekatere sobe.

Sovražniki se pojavljajo zelo hitro in so zelo nevarni. Ne morejo streljati, toda preprosto se prilepijo nate in ti izpijejo vso energijo. Vsa kršnih oblik so, skupno pa imajo to, da jih je zelo težko ubiti. Stiskalnice te mimogrede pošljejo v večna lovišča.

Vse predmete jemlješ in uporabljaš s pritskovo na USE. Za podrobnejše informacije pa kliči (011) 472-420

# Fornirad

## INFORMATIKA

TRST – Ul. Cologna 10  
– Tel: 040/572106

hišni računalniki – periferična in splošna oprema – hardware (strojna oprema) – software (programska oprema)

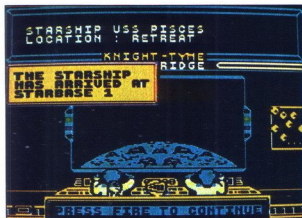
---

# Fornirad

## ELEKTRONIKA

TRST – Ul. Conti 9  
– Tel: 040/733332

elektronski komponenti – antene  
– aparature RTV – CB



## DAMIR ŽAGAR

Če imate radi arkadna pustolovščina, ki so se v svetu iger za spectrum pojavila z Ultimativom Knightrom, če vas tridimenzionalne igre že malce dolgočasijo in če vam je bil pred kakim mesecem ali dvema vseč Mastertronico Spellbound, potem je Knight Tyme kot nalašč za vas.

Meni, ki ga zagledate na zaslonu, morda že poznate iz Spellbounda, vendar o tem ne razmišljajte preveč, temveč se kar lotite igrice, bodisi z gumicami bodisi z igralno palico. Grafika je povprečna, igranje pa sme izbrati, kateri atributi bodo prikazani (igralčevi ali iz ozadja), poleg tega pa lahko pozicijo vsakih hip posname oziroma jo naloži s kasete. Vse to napravite s črko X, kar je edina možnost v začetku igre. Temu delu igre lahko zamerim samo to, da vam kakih deset posnetkov napolni tudi do ene strani kasete.

V Spellboundu si rešil čarovnika Gimbala in zdaj se kot njegov pomočnik in čarobni vitez znajdete na krovu vesoljske ladje USS Prisce v 25. stoletju. Na mirovno-trgovski odpravi morate najti pot nazaj v svoj čas.

Najprej kreni desno od sebe, v kateri si se znašel in ukaži ladijske-mo računalku, naj ti pomaga. Dal ti bo prazno razpoznavno izkaznico (ID). Vzemi jo, prav tako ribo, ki jo tu najdeš. Od servisnega robota S3 E3 vzemi posodo z lepilom in pojdi do komandnega mosta. Saraba, kjer dobiš tablete. S tabletami in s čarom Fortity boš imel neskončno energije, dokler boš oboje nosil s seboj. V oboji za počitek vzemi kamero in odidi do drugega robota, Klinka. Daj mu kamero in film. Sleci plašč, v katerem si neviden in ukaži Klinku, naj ti pomaga. Potem te bo fotografiral in fotografijo boš natepljal na prazno ID. Zdaj imaš razpoznavno kartico, oprni jo na obliko in odidi do Sharon. Od njene boš plinško masko in priročnik. S priročnikom si boš pomagal do predmetov na vrhu vesoljskih modulov. Vrh in zemljevid, ki ju najdeš v enem od njih, odnesi Gordonu. Zdaj lahko vodiš vesoljsko ladjo.

## Knight Tyme

Tip:arkadna pustolovščina

Računalnik: spectrum 48 128 K

Format:kaseta

Cena:2,99 funta

Založnik:Mastertronic

Povzeteč:Spellbound

v vesolju

Ocena:10/10

Odleti na Eden, potem pa na Starbase 1. Ukaži Gordonu, naj ti pomaga in popravi bo teleportiraj. Vzemi priročnik, tablete in lepilo, ki ga ne moreš odvreči, in se teleportiraj na postajo, katere koordinate so X1 Y2 Z3. Prebriži jo in vzemi vrečo s kamenjem in sekuro. Posode z lepilom se znebiš, če jo daš Hectorju. Na vrhu enega od vesoljskih modulov boš našel škornje. Obuj jih in se vrni na ladjo (koordinate X0 Y0 Z0).

Na ladji napolni rezervoarje z gorivom. Vrni se na Eden, z njega pa odleti na Monopol. Poselbi, da boš imel na sebi plinško masko in plašč, ki te naredi nevidnega, ne pozabi pa tudi na sekuro in vrečo s kamenjem. Potem se teleportiraj na Hooperjev asteroid (koordinate X1 Y8 Z4). S stal enega od prostorov poberi del

sončne ure in odidi k Hooperju. Daj mu vrečo s kamenjem in sekuro, v zameno pa boš dobil drugi del ure in talisman. Vrni se v vesoljsko ladjo. V eri od sob sestavi oba dela sončne ure. Sleci plašč in snemi plinško masko, obesi pa nase talisman.

Prek Elena, Starbase 1, Ganrola in nazadnje Nafta odpotuj na Reterat. Tu spet obeci plašč in si nadeni plinško masko. Teleportiraj se na luno (koordinate X8 Y4 Z1). Ko naletiš na banero, jo odstrani s čarom Remove Barrier. Zda ti je odprta pot do Murphya. Od njega boš dobil zadnji del sončne ure. Ko se vrneš na ladjo, stakni ta del s prejšnjim. Snemi masko in sleci plašč, odidi na komandni most.

Na gospodarje časa boš naletel

na Outpostu. Pojdi torej tja, a ko prispes, se spet zavaruj s plaščem in masko, nato pa vrzi do sebe, kjer si pustil dele ure. Vrzi streljo z urokom Lighting Bolt. Tako boš dokončno sestavil uro. Vzemi jo in se teleportiraj na planet (koordinate X8 Y9 Z6). Ker imaš pri sebi sončno uro, se ti ni treba zmeniti za opozorila gospodarjev časa; pojdi kar naprej do konca predora. Ko se bodo gospodarji časa srečali s tabo, bodo na tvojo ladjo prenesli časovni stroj in nato umrli.

Vrni se na ladjo in odleti na Ganrola. Ko se bo ladja spustila do črne luknje, odidi do bove poleg zračne komore. Iz reševalne kapsule se je spremeniš v kontrolno sobo, v njej pa boš našel časovni stroj, ki so ga tu pustili gospodarji časa. Vključuj ga. S sporočilom, ki se izpiše na zaslonu, se je končala tvoja druga, a ne poslednja naloga, čarobni vitez!

## Nasveti za igre:

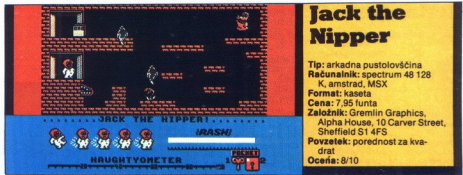
1 Če ti kaj ne gre od rok oziroma če dobiš sporočilo, da imaš zelo malo energije, uporabi urok Fortity na sebi.

2 Če ti kaka oseba noče dati tega ali onega predmeta, če je utrujena oziroma ti noče pomagati, uporabi urok Fortity na njej.

3 Ko odhajaš iz vesoljske ladje, nikar ne pozabi obeci plašč, sicer te bo ujela straža, z masko pa se boš zavaroval pred strupenim plinom.

4 Če ni opazje za vodenje vesoljske ladje, gotovo nisi slekel plašča ali pa si pozabil pripeti razpoznavno kartico.

5 Med poleti od planeta do planeta imej v mislih gorivo napolni rezervoarje, kjerkoli se ti ponudi priročnost.



## ANDREJ TOZON

V igri nastopaš kot Jack, najporednejši otrok na svetu. Lokacija je okoli 50. grafika je enkratna, vse skupaj pa malo spominja na Wallyja in njegovo družino. Jacka usmerjaš s tipkami Z-levo, X-desno, O-gor, K-dol, O-gumb za streljanje, ENTER-skoki vrata, gor-strelj-skok, (levo, desno) +strel streljanje. Igra je narejena dobro, samo rešitev je včasih nelogična, pa tudi tipke ne moreš določiti sam. Scenarij je preprost: bodi tako poreden, da bo kazalec porednosti prišel

do 100 odstotkov. V igri delaš vse mogoče, od razbijanja predmetov do bombardiranja policijske postaje. Za vsako stvar, ki si jo na novo ušpil, ti začnejo loviti prodajalci, milničniki.

Najprej se moraš oborožiti s sebi ustreznim orožjem – pihalnikom. Spravljen je na visoki polici v tvoji sobi. Malo plezanja po omarah in postelji, pa se bo pihalnik kmalu znašel v tvojih rokah. Z njim je najbolje iti dve lokaciji levo v spalnico, kjer strši duh. Postavi se za posteljo in ga skušaj s pihalnikom zadeti. To počni, dokler se ti bodo odstotki

## Jack the Nipper

Tip:arkadna pustolovščina

Računalnik: spectrum 48 128

K, amstrad, MSX

Format: kaseta

Cena: 7,95 funta

Založnik: Gremlin Graphics,

Alpha House, 10 Carver Street,

Sheffield S1 4FS

Povzeteč: porednost za kvadrati

Ocena: 8/10

povečali. Nato skoči na posteljo in spusti pihalnik, da se bo razbil. Prvo delo je opravljeno. V prodajalni igrač razbij medvedka, ki se smejeja s polci, v kitajski prodajalni pa te čakata dva krhika krožnika. Na policijski postaji leži ob mucki baterija. Nesi jo v prodajalno računalku in pojdi v prazni kvadratah na levi strani pulta. Računalniki bodo nehali delati, baterijo lahko razbiješ. Na desni strani banke leži ob napisu ključ, ki odpra skrivne prehode za radiatorje v banki in muzej. Najprej pojdi v banko. Poberi disketo pri vhodu in pazi, da ne pades v



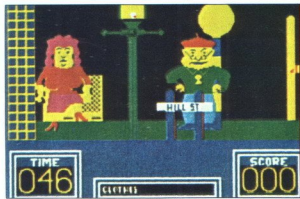
luknjno Disketo nesu in Technology Research and skoči na računalnik v desnem kotu spodaj Razbij disketo Še enkrat v banko Pojdi v luknjno, na vrhu poberi pralni prašek in se napoti v pralniko Sprehodi se mimo pralnih strojev in iz njih bodo začeli lesti mehurčki Razbij prašek in poberi posode, ki je na stroju Z njo umiriš sklepetajoče proteze v Gumbo s Chopping Molars Razbij posodo

Poti nas pelje nazaj v banko Spet je treba v luknjno, toda sedaj poberi list s črko A in ga vrzi dol Gremo v muzej Dudo in kip, ki sta na stojih, lahko mirno razbijš, moraš pa iti spet v skrivni prehod za radiatorjem Zdaš lahko razbijš tudi ključ, saj ga ne boš več potreboval. Ko

prideš iz skrivnega prehoda, poberi trobiljo, ki leži na polici Dva zaslona levo boš na kamenu našel bombo Pojdi iz tega stanovanja na policijsko postajo Mahnu jo desno k ječam in spusti bombo Ko bo eksplozionalna, bodo ujetniki prosti Skoči na stojalo po utež in zbeži levo Postavi se zraven mačke in zatrobi na trobiljo Mačka bo skočila v zrak in se s kremplji zatankala za stroj Utež nisi in prodajalno nogavico in premakni levo ročico (skločil s police) Utež razbij Pojdi skozi šolo skozi vrata Na pultu boš videl sladkorno peno Zatrobi pri mački, pojdi še enkrat v šolo in spusti peno S posodo, ki jo dobiš na isti lokaciji kot peno, pojdi v kitajsko trgovino Spusti posodo V cvetličarni poberi strup na desni

polici Spusti ga pri vrtnarju, ki ima okoli sebe lepe rotice Strup razbij Na pokopališču se za duhom skriva vreča prsti Poberi jo in se vrni na to lokacijo Spusti vrečo Zrastle bodo mesojede rastline Pojdi v hišo ob poti Tam te čakajo zadnja mačka in kročnika, ki ju je treba razbiti Uniči še vse predmete, ki si o stalih, in uspešno boš končal igro.

Pa še POKÉ naložite uvodni BASIC in sliko, nato računalnik resetirajte in vpišite naslednji program  
10 FOR n=16384 TO 16410 READ a POKÉ n,a NEXT n RANDOMIZE USR 16384 20 DATA 49,0,76,221,33,0,91,17,255,164,62,2-55,55,205,86,5,49,0,96,62,201,50,25-2,169,205,168,97



## Benny Hill

**Računalnik:** spectrum 48 K  
**Format:** kaseta  
**Cena:** 7,95 funta  
**Založnik:** DK tronica, Unit 6, Shire Hill Ind. Est., Saffron Walden, Essex CB11  
**Povzetek:** Program Plus  
**Ocena:** 9/10

### VLADIMIR IN MARIO YUKSAN

Praviti imamo z grafiko v slogu Popeya, vendar z nekaj izboljšavami. Bolj je poudarjena trodimenzionalnost in sicer tako, da so oddaljeni predmeti postavljeni nekoliko višje, a bližji nekoliko nižje. Če se gibljete v eno smer, brez težav s pritiskom na tipki za gor ali dol menjate tekoči trak, po katerem skakajete

Igra vsebuje tri stopnje. Na prvi stopnji dobis 100 časovnih enot, na drugi 150, na tretji pa 200. Na vsaki stopnji dobis 20 točk za vsak predmet, ki se ti ga posreči vreči v skatlo. Če te ujamajo čuvaji, izgubis štiri točke, ko pa vržeš v skatlo vse predmete, dobis točke za čas. Na vsaki stopnji je šest predmetov, razen na tretji, kjer jih je devet (toda v skatlo moraš vreči samo šest predmetov). Med igro moraš biti zelo pazljiv, kajti predmeti so zdaj tu, zdaj tam.

Zaslon je razdeljen na dva dela. V zgornjem teče igra, v spodnjem so pa časomer, števec rezultatov, pozkazatelj naloga, ki jo opravljaš in opozorilo na nevarnost (watch out)

Naloga je na prvi stopnji tipično Benny Hillovska. Ukrasti moraš žensko perilo in obleko ter vse to spraviti v skatlo z napisom «clothes» na začetnem zaslonu. Obleka in perilo sta na zadnjem zaslonu, če gledamo od začetnega prsti desni, in sicer vse skupaj visni na vrvi za sušenje perila. Ne gre pa vse gladko in to boš občutil na svoji koži, če prideš v roke kakšni sadistično razpoloženi

ženski Ker pa ni kdo ve kako pametna, jo boš z malo vaje brz pretenal. Toda zapomni si ves čas se moraš premikati, sicer te bo mimogrede ujela. Najlažje se ji boš izmaknil, če boš spreminjal smer, njo pa zavedal, da se bo zaletela v stebel, ograjo ali v kak predmet.

Na drugi stopnji moraš rabutati jabolka in jih poskrviti v skatlo z napisom «apples» na začetnem zaslonu. Ko greš z leve proti desni, pazi na hlev z balami sena! Pot te nato vodi skozi gozd (drevesa so zelo gosta in zato je ta del igre najtežji) Brž ko ukradeš prvo jabolko, te začne preganjati čuvaj. Če te ujame, jo boš dobil s palico po hrbtu. Malce pametnejši je od ženske, vendar se boš tudi njemu izmaknil, če boš valdir prebijanja skozi gozd. Da vse le ne bi bilo prelahko, se v rednih presledkih pojavlja traktor, ki vozi od hleva do dreves. Če se zaletis vanj, boš nekaj časa brez zavesti, vendar jabolka ne boš izgubil. Na traktor te opozori značilni zvok, toda opozorila ne bo, če bo traktor že blizu.

Na tretji stopnji moraš ukrasti sedem (od devetih) igrač iz trgovine in jih skriti v skatlo z napisom «jumps» (skatla je običajno na začetnem zaslonu, igrače pa na polici na zadnjem zaslonu). Ko ukradeš igračo, ti grozita dve nevarnosti. V mestecu namreč skrbita za red in mir dva angleška «bobbyja». Zelo sta pametna, vendar sta tako uvidevna, da ne vtaknejo po ključ, če ti dobita pod roko – vzemeta ti le igračo, jo

vrneta v trgovino, tebe pa pošteno premaltata.

Moraš se hitro premikati in še hitreje razmišljati. Skoraj ni izhoda, če sta policajca skupaj na enem zaslonu! Skušaj ju pretenatati tako, da ju spelješ na kar največ zaslonov, sele potem greš lahko mimo.

Tisti, ki jim ta navodila ne pomagajo dovolj, lahko kličejo na telefonsko številko (041) 712-249.

## Spitfire 40

**Tip:** simulacija letanja  
**Računalnik:** spectrum 48K, amstrad, C 64  
**Format:** kaset, disketa (C 64)  
**Cena:** 9,95 (12,95) funta  
**Založnik:** Mirrorsoft, Purnell Book Centre, Paulton, Bristol BS 18 5LQ  
**Povzetek:** 2. svetovna vojna na evropskem nebu  
**Ocena:** 9/9

### GORAN KLEMENČIĆ

Grafika kot pri simulatorjih, zvok, prilagojen okolju (komodorjevo se bodo morali Book Centre, Paulton, Bristol BS 18 5LQ  
**Povzetek:** 2. svetovna vojna na evropskem nebu  
**Ocena:** 9/9

grafika kot pri simulatorjih, zvok, prilagojen okolju (komodorjevo se bodo morali ne poslušas Mozarta) Nalaganje najprej se moramo odločiti, ali nas

zanimajo posnet status pilota (o tem pozneje), kar pa lahko nalozimo samo v začetku, a posnamemo po vsakem srečnem pristanku, dalje, izberemo si pilotovno ime (na voljo jih je sardani pet) – majhen nasvet izberite vedno isto ime, saj bo o seznamu pilotovnega statusa na kaseti morda zmešnjava!

Potem vzemimo podatke o sebi (tj. statusu pilota) ime, čin, oddelkovanja, ure letanja in številce sestreljenih letal. Program je prijazen omogoči varovanje v letenju oziroma dovolj na nebu ali pa kar pravilno boš, če se odločiš za slednjo možnost, vas na nebu pričakajo sovražniki (banditi), zveste o njihovem številu, višini in koordinatah glede na svoje letališče. Da bi se boljje znašli v kabini svojega «spita», nekaj namigov:

**a) Instrumenti in komande.** Kot v pravem spiftrju, le radijskega aparata ni, da bi se povezali s kontrolnim stolom. Nekako boste vzdržali brez vzorce, paziti morate le na merilnik višine, merilnik hitrosti, merilnik potisne moči motorja, merilnik nagiba (v stopnjah) in klasični merilnik nagiba. Ne spreglejte še treh opozorilnih znamenj: tisto v zgornjem levem kotu vam pove, ali so kolesa dvignjena oz. spuščena, lučki na desni strani zasлона vam kažeta, ali so zavore vključene (brake) in kako je s kolesi (gear). Važen je še pogled iz kabine na vzvratno ogledalo, ki vas opozori, ali se vam je kak Nemeec pripljel za rep in nikar pozabiti na zemljevid, na katerem sta položaja vašega in sovratnikovega letala (sovratnik je črn).

Kolesa: «G» zavore-kolesa «B», večja potisna moč motorja «Q», manjša «W», zakrlica «F», pogled iz kabine «space», zemljevid «M», počevca zemljevida «N», nagib v stopnjah «A» «S» Sicer pa letalo usmerjamo z igralno palico in z njo se lahko gibljemo v osem smeri. S tipkami pr C 64 ne boste nič opravili!

**b) Sovražniki.** So beli, rjavi in črni. Rjave najlažje sestrelite, kar se počasi, neprevditi in vam ne uidejo, če se jim priplje za rep. Beli so sicer tudi počasi, vendar jih je težko opaziti, pa tudi uidejo vam, kadar letite proti njim! Črni so najbolj zveni in hitri, imajo jih na muhi, toda znajo se izmakati, radi se vam pripljejo za rep, v skrajnem primeru pa krenejo naravnost proti vam v samomorilski napad (k sreči ne preveč učinkovit). V zraku so vedno le sovražna letala, zato se vam ni treba bati, da bi sestreljevali prijetje.

Če se vam Nemeec priplje na rep, brez obotavljanja strogolavite in si naberte hitrost. Nobenega smisla nima delati «sodčkov» in «vrvičev», saj niste v akrobatskem letalu. Zvočni signal vas bo opozoril, da je v bližini «bandit» brž ga poiščite, sicer bo po vas! Nigkar se ne podite za Nemeem, ki strogolavje (še sami boste zorali po tleh). Še zlasti, če je bila začetna višina manj, kot 3000 m! Sovratnik na začetku ni zaždel! Kar drži prst na sprožilcu: najprej kratak blisk, potem eksplozija. Če bo Nemeec, ki ga imate na muhi, včasih preprosto izginil z zaslonu, to pomeni, da je treščil na tla – računalnik vam ponavadi prizna zmago.

**c) Status.** Če hočete narediti lepo ▶

vojaško (pilotsko) kariero, upoštevajte tole!

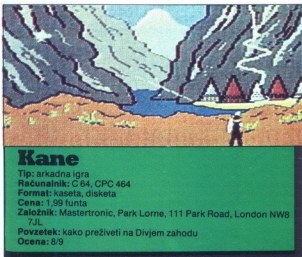
Prav gotovo ne boš daleč prilezel, če se boš po vsakem drugem pristanku polomil oziroma če boš na nebu od devetih Nemcev sestrelil samo dva. Veliko bolj te bodo cenili, če boš imel npr. eno uro intenzivnega letenja in 15 zmag (to je malo), kot pa če boš letel pet ur in sestrelil 30 letal. Zelo hitro upraviči z vsemi sovražniki na nebu in lepo vzletaj! pristajaj!

Medalje so tri (vsaj jah jih doslej nisem dobil več). Za pogum, za posebne zasluge in Viktorijin križec Cinov pa je več, po mojem je najvišji wing commander (letalski polkovnik). Čas poleta je približno realni čas, zato si boš moral vzeti kar precej prostega časa za igrice (največ ti pobere letenje proti sovražniku, še zlasti na višini nad 6000 m).

Se nekaj posvetov iz pilotiranja. Vzgite motor, potisna moč naj bo kar najmanjša, spustite zavore in počasi dodajajte plin, dokler ne dosežete največjega polnisa. Ko letalo doseže hitrost 100 milj na uro (možno so tudi precejšnja odstopanja), potegnite palčko k sebi in počakajte, da se bo letalo odlepilo od tal, nato pa uvlecite kolesa. Opozorilo med vzletom se ne igrajte z zavornimi, kajti to se vam utegne maščevati!



Med samim dvobojem ne opazujte instrumentov, pazite le na merilnik višine. Ko opravite z »banditi«, kar strmoglavite in odzveržete plin (6 ali 12), približno 500 m nad zemljo se zrvanite (še bolj) boste izgubili hitrost, spustite kolesa in zakrlica. Pozor! Če je pri pristajanju hitrost nad 200 milj na uro, vam bo zakrlica in kolesa vzel vrag! In še to: pristane lahko kjerkoli, sploh pa letalskiča ponavadi ni treba iskati, ker je tik pod vami. S spuščeniimi kolesi in zakrlici se spuščajte lepo počasi, ko pa se dotaknete tal, izključite motor in pritisnite na zavore. Letalo bo nekajkrat poskočilo, potem pa se bo umirilo. Če vse to ne pomaga, imate na voljo še telefon (064)-69-950 (zahtevajte Gorana)



## Kane

Tip: arkadna igra

Računalnik: C 64, CPC 464

Format: kasete, disketa

Cena: 1,99 funta

Založnik: Mastertronic, Park Lorne, 111 Park Road, London NW8

7,1.

Povzetek: kako preživeti na Divjem zahodu

Ocena: 8,9

## BORUT HOČEVAR

**K**ane ni ravno najnovejši program, se pa odlikuje z grafiko, zvokom in predvsem animacijo. Ima štiri stopnje. Ste nekje na Divjem zahodu. Obdajajo vas visoke gore, nedaleč od vas pa je indijansko plemo. Po kotlini odmeva donenje bobnov, na srečo pa Indjanci niso izkopal bojne sekire. Vaš junak nestrno drži v roki lok, nad njim pa leti jata divjih rac. Vaša naloga je, da jih sestrelite čim več. Za vsake tri uspešne strele dobite po eno življenje. Včasih letijo race druga vrh druge. Zadedi poskušajte tisto, ki leti najvišje. Če vam bo uspelo, bo race sklatila še vse druge, ki letijo pod njo, poleg tega pa boste za vsako raco dobili po eno nagrado puščice.

V levem spodnjem delu zaslona vas računalnik obvešča o tem, koliko rac ste postrelili (birds killed), v desnem pa, koliko vam jih je ušlo (birdsies home). Vmes so še točke in puščice, ki jih imate na voljo. Ko vam zmanjka puščic ali ko jata divjih rac odleti naprej in nimate več kaj streljati, preidete na naslednjo stopnjo. Le če niste dobili niti enega življenja, boste igrice končali.

Druga stopnja ni preveč težavna. Imate konja, ki se ves čas prestopa in čaka, da ga boste pognali v dir. Prejezditi morate štiri milje dolgo pot do mesteca Kane. Na poti je polno skal, ki jih morate preskakovati. Če se boste s konjem spotaknili ob skalo, bo vaš junak ob močnem tresku pristal na tleh. Če igrice ne boste mogli nadaljevati od tam, kjer ste doživeli »karambol«, pritisnite F1 in commodore vas bo postavil na začetek poti. Konj je izredno dobro animiran, skok je izpiljen do najmanjših podrobnosti. Tu je tudi izvrstna glasba, ki vas spremlja na poti do kažipota. Ta vam pove, da ste prišli do konca druge stopnje. Tretja stopnja je nekoliko težav-

nejša. Stojite predi Kanea. Nikjer žive duše. Pred vami stoji saloon in še nekaj hiš. Nenadoma se izza vogalov prikažejo razbojniki in varčujejo z municijo. Na srečo ste tudi vi oboroženi s koltom, zato se jim hitro zahvalite za dobrodošlico: Z igraino palico prepeljite merilnik na enega od banditov in ustrelite. Če boste natančno pomerili, bo ob prodornem kriku zgnil zaslona. Če ne boste previdni, pa se bo to kaj hitro zgodilo tudi z vami. Zato vam svetujemo, da se čimveč gibljete. Kadar hočete dodobra premisliti o svoji nadaljnji potezi, bo najbolje, če se za nekaj časa umaknete na desno, čez rob zaslona. Tam vam nihče ne bo mogel do živega, poleg tega pa boste lahko v miru napol-

nili orožje. Na levi strani zaslona vidite, koliko banditov ste postrelili (bad guys killed) na desni pa, koliko jih je ostalo (bad guys left). Postreliti jih morate dvannast.

Četrta stopnja zelo spominja na drugo, le da je vaša naloga drugačna: ustaviti vlak, ki je pravkar odpejal s postaje. Odjedzite za njim, tako pazite, da bo vaša hitrost enaka tisti, s katero vozi vlak. Kmalu boste pred seboj zagledali skalo in grm, ki sta ravno toliko oddaljena, da ju z enim skokom ne boste moglo prekosčiti. Malo pred skalo pritisnite tipko in je ne izpuscite. Če boste dobro ocenili mesto odvra, boste prekosčili skalo, konji pa bo priletel na srednje noge in se spet pognal v zrak. Prekosčili boste tudi drugo oviro. Zdjaj lahko tipko izpuscite in povečate svojo hitrost. Ne pozabite pa, da ovir ne bo konec, dokler ne prehitite štirih vagonov in lokomotive. Takrat se bo dogajanje na zaslonu umirilo, to pa bo pomenilo, da ste svojo nalogo tudi tokrat uspešno opravili.

Igre s tem ne bo konec, ampak se bo vse skupaj ponovilo. Če vam bo po četrti stopnji ostalo še kaj življenj, vam v drugem krogu ne bodo koristila – tudi tu bo šlo od začetka, le da boste morali za vsako življenje zagreti po štiri divje race. Druga in tretja stopnja se ne bosta spremenili, na četrti pa bo imel vlak sedem vagonov.

Če vam ne bo uspelo priti tako daleč, nič ne skrbite. Na začetku lahko za vajo sami dolocate, katero stopnjo boste igrali.

## Prvih 20 revije Sinclair User (okt.)



1	New!	Ace	Cascade	£9.95
2	(2)	Jack The Nipper	Gremlin Graphics	£7.95
3	(3)	Kung Fu Master	US DOG	£7.95
4	(1)	Ghosts 'N' Goblins Elite		£7.95
5	(5)	Ninja Master	Firebird	£1.99
6	New!	Video Olympics	Mastertronic	£1.99
7	(4)	Molecule Man	Mastertronic	£1.99
8	(15)	Bobby Bearing	The Edge	£7.95
9	(6)	Knight Time	Mastertronic	£2.99
10	(7)	Green Beret	Imagine	£7.95
11	(8)	Theatre Europe	PSS	£9.95
12	New!	Rebel Star	Firebird	£1.99
13	New!	Full Throttle	2.99 Classics	£2.99
14	New!	Stainless Steel	Mikro-gen	£7.95
15	New!	Cauldron II	Palace	£8.99
16	(10)	Incredible Shrinking Fireman	Mastertronic	£1.99
17	New!	The Boggit	CRL	£7.95
18	New!	Kane	Mastertronic	£1.99
19	New!	Universal Hero	Mastertronic	£1.99
20	(14)	Formula 1 simulator	Mastertronic	£1.99

Tudi naj sodobnejši računalnik  
ne more prodreti v skrivnost

*Mystique*  
parfuma



UNIVERZIADA ET



KRKA KOZMETIKA —  
KOZMETIKA UNIVERZIADJE



KRKA

kozmetika

EPSON

**E**PSON **P**  
TISKALNIKI SO **P**REPROSTO REČENO  
NUJNOST ZA OSEBNE  
**S**OLIDNA IZDELAVA ZAGOTAVLJA  
RAČUNALNIKE.  
**O**MOGOČA NEMOTENO DELO  
LE TISKALNIK,  
KI GA JE IZDELAL EPSON **N.**

LX - 86 Novi  
Epson daje  
širjenje



tiskalnik, ki ga  
tržišću za nadaljnje  
računalništva.

Generalni in izključni zastopnik za Jugoslavijo:

**avtotehna**

LJUBLJANA TOZO Zastopstva, Celovška 175, 61000 Ljubljana  
telefon: 061 552-341, 551-287, 552-182  
telex: 01 629