

5  
Z MIKROKOMPUTEREM NA TY

NR INDEKSU 353965  
PL ISSN 0860-1674

# Bojtek

MIESIĘCZNY DODATEK DO SZKICÓW MŁODYCH

NR 5(17)

MAJ 1987

CENA 100 ZŁ



**INFOSYSTEM'87**

**COMMODORE 16**

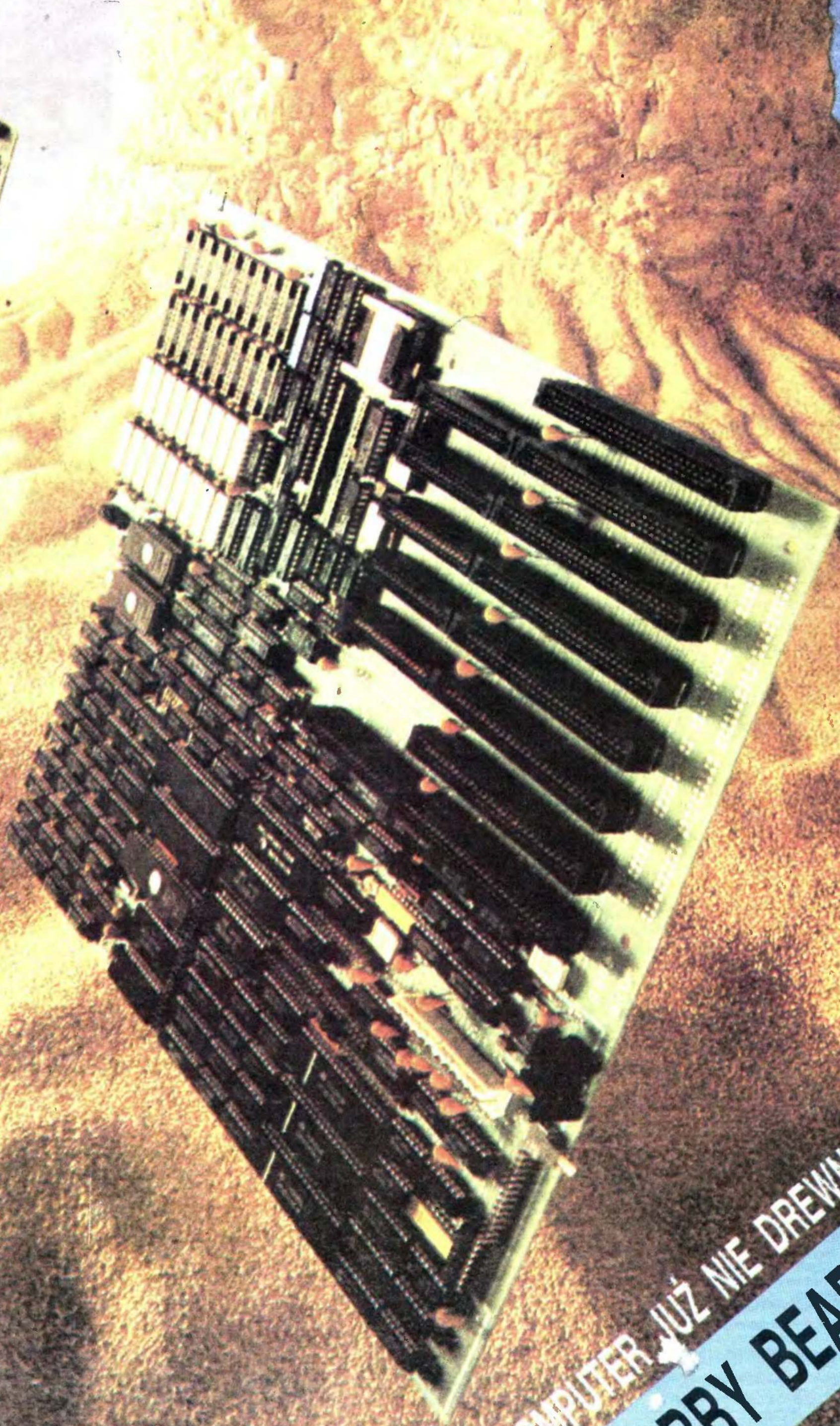
**TURBO  
BASIC CPC**

**CO**

**JEST W ŚRODKU  
ATARI**

**CEBIT'87**

**BIURO  
PRZYSZŁOŚCI**



KOMPUTER JUŻ NIE DREWNIANY  
**BOBBY BEARING**

# KOMPUTER NA KÓŁKACH

Jakie są rezultaty ogłoszonej przez nas półtora roku temu akcji zwalczania analfabetyzmu komputerowego?

Najkrócej mówiąc: jest obecnie lepiej niż mogliśmy to przed półtora rokiem prognozować. Jest więcej komputerów niż się nam wydawało, że będzie. Więcej osób potrafi się nimi w miarę sensownie posługiwać. Rozwinął się również żywiołowo rynek usług i wydawnictw dla ludzi interesujących się informatyką. Są to plusy. Ale obok nich, te pierwsze półtora roku powszechnej edukacji informatycznej w Polsce ujawniło i wyostrzyło wiele zjawisk negatywnych i w największym stopniu niepokojących.

Zaniepokojenie wywołuje fakt, że szeroka edukacja komputerowa odbywa się u nas w przeważającej mierze poza lub obok szkoły. Resort oświaty nie skorzystał z szansy przybliżenia się do życia. I wcale nie z braku wystarczającej liczby szkolnych komputerów leży problem, tylko raczej w braku chęci, wyobraźni i mobilności. Ale to już, zdaje się, naszej oświaty wada nieuleczalna!

Właśnie ta indolencja szkoły spowodowała, że gwałtownie rozszerza się przepaść między dziećmi, które mają dostęp do komputera a tymi, którzy mogą tylko o tym marzyć. Jeśli ktoś pamięta jeszcze, że podstawowym zadaniem systemu oświatowego jest wyrównywanie szans startu życiowego wszystkim dzieciom, to musi przyznać, że udało się uzyskać stan dokładnie odwrotny.

W sytuacji upośledzonych są przede wszystkim dzieci z miast i wsi. Gorsze jest tam bowiem wyposażenie szkół i pracownie komputerowe trafiają się tam jeszcze rzadziej niż w dużych miastach, nie ma klubów komputerów, a i rodzice rzadziej dają się przekonać synowi, że warto kupić komputer.

Żeby choć nieznacznie poprawić szanse startu cywilizacyjnego młodzieży spoza wielkich miast, Związek Młodzieży Wiejskiej postanowił ufundować i przekazać pewną liczbę komputerów dla przodujących kół ZMW. Piękna to inicjatywa, ocale niebawem ważniejsza niż najlepiej nawet przygotowane plenum czy narada. Ale jej zasięg też jest z oczywistych powodów ograniczony.

Dlatego zaczęliśmy się w „Bajtku” zastanawiać, co zrobić aby jak największa liczba

dzieci wiejskich mogła choćby na krótko uzyskać dostęp do komputera? Okazuje się, że jest na to sposób. Z powodzeniem stosują go zresztą już od dwóch lat nasi wschodni sąsiedzi, których również nie stać jeszcze na zafundowanie każdej szkole pracowni informatycznej z prawdziwego zdarzenia. Żeby temu zaradzić utworzono „komputerowe klasy na kółkach”.

Genialne pomysły są jak to zwykle bywa, bardzo proste. Dobrze resorowany autobus, stoliki i krzeselka zamiast ławek, na stolikach komputery osobiste, z przodu autobusu tablica... Zalogę „klasy na kółkach” stanowi kierowca-elektrotechnik (trzeba przecież podłączyć „klasę” do wiejskiej sieci elektrycznej lub uruchomić własny, przenośny agregat) i instruktor-informatyk. Doświadczenia z funkcjonowania takich klas są rewelacyjne, a wykorzystanie każdego komputera osobistego jest kilkakrotnie większe niż w szkole stacjonarnej, gdzie przecież przez większość dni komputery stoją zamknięte na klucz!

Co stoi na przeszkodzie, aby i po polskich wsiach zaczęły jeździć takie komputerowe „klasy na kółkach”? Przynajmniej kilka na początek?

Nie wierzę, aby urzeczywistnieniem tego pomysłu mógł się zająć resort oświaty, bo to wymagałoby przecież aby w ogóle cokolwiek zaczął robić... Ale kierujemy tę inicjatywę do organizacji młodzieżowych. Toż to sama radość umożliwić dzieciom ze wsi pierwszy kontakt z komputerem. Były kiedyś popularne na wsiach studenckie „białe niedziele”. Jeździły po Polsce ruchome pracownie rentgenowskie. Dzisiaj potrzeby czasu są takie, że powinny trafić na polską wieś komputery. Redakcja „Bajtka” zobowiązuje się stanowić załogę merytoryczną pierwszego takiego komputerowego autobusu. Sądzymy, że koledzy z innych pism komputerowych objęliby patronat nad kolejnymi „klasami na kółkach”. Jest to zresztą wspaniałe pole dla studentów informatyki i elektroniki...

Pomysłów i rozwiązań organizacyjnych może być zresztą wiele. Liczymy na nie. Szukamy sojuszników naszej akcji „Komputer na kółkach”. Kto pierwszy nas poprze?

Waldemar Siwiński



## Szanowny Panie Redaktorze

Uważam komputery za wspaniałą pomoc dydaktyczną, szczególnie dla szkolnictwa specjalnego. Jeszcze przed „erą komputerów” w szkołach specjalnych i zakładach wychowawczych wykorzystywane były rozmaite urządzenia mechaniczno-elektryczne (skrzynki sprawnościowe) przy pomocy których dzieci niepełnosprawne zrealizowały różne zadania dydaktyczne. O ile dla dzieci-uczniów szkół masowych zastosowanie najnowszych zdobyczy nauki i techniki (w tym przypadku komputerów) pozwala lepiej realizować program dydaktyczny nie jest natomiast niezbędne i konieczne dla ich pomysłowego funkcjonowania w społeczeństwie, o tyle dla dzieci niepełnosprawnych (np. upośledzonych umysłowo w różnym stopniu) może okazać się zaistniała jedyną szansą integracji z normalnym społeczeństwem.

Z tego też względu wnioskuje o wyposażanie w komputery w pierwszym rzędzie szkół specjalnych i zakładów wychowawczych dla dzieci upośledzonych wraz ze specjalnie opracowanymi różnorodnymi programami w zależności od potrzeb.

Mając to na uwadze jako czynny artysta muzyk przeznaczyłem dochód ze specjalnie dla tego celu zorganizowanych moich koncertów na zakup komputera dla Szkoły Podstawowej Specjalnej Nr 115 we Wrocławiu.

Władysław Salik  
ul. Łączności 6/9  
53-330 Wrocław

było dokładnie odwrotnie. Moją przyszłą (już niedługo) żonę poznałem właśnie przed klawiaturą Commodore 64 u znajomego. Wspólne zainteresowania bardzo nas zbliżyły i spotykaliśmy się coraz częściej, zwykle u kolegi, który miał chyba najmniej powodów do zadowolenia.

To było rok temu. A dzisiaj? Nasz ślub odbędzie się w Święta Wielkanocne.

Zapraszamy Cię Bajtku!

Nie planujemy na razie zakupu mikrokomputera. Prawdę mówiąc i tak spełnił już w naszym życiu sporą rolę.

Jak widzisz Bajtku, są jeszcze takie społeczne funkcje informatyki, o których nie pisałeś na swoich łamach.

Marcin Kalicki  
Gdynia

W imieniu wszystkich członków (34 osób) klubu komputerowego „STOBIT” oraz własnym, przesyłam wyrazy ogromnej wdzięczności za przyjęcie patronatu redakcji „BAJTKA” nad naszym klubem.

Opublikowanie informacji o naszym klubie (w 2 n-rze br.) spowodowało wzrost zainteresowania mieszkańców osiedla naszą działalnością a jednocześnie dało nam poczucie przynależności do ciągle rosnącej rodziny pasjonatów informatyki.

Członkowie klubu tworzą bardzo młody zespół ludzi związanych silną więzią wspólnych zainteresowań.

Przyjęcie patronatu przez redakcję „BAJTKA” jest dla nas ogromnym wyróżnieniem mobilizującym.

Jesteśmy przekonani, że naszą pracą zasłużymy na uznanie nie tylko rodziców członków klubu ale i naszego patrona.

Jan Ejgerd-instruktor  
z upoważnienia członków klubu STOBIT  
przy DK „STOKŁOSY”  
Warszawa ul. Lachmana 5

## WYBIERZ SAM

GRA O JUTRO	
Zawód z perspektywą . . . . .	3
Infosystem'87 . . . . .	4
PROGRAMOWAĆ MOŻE KAŻDY	
Tablice w LOGO . . . . .	5
KLAN ATARI	
Ciągi tekstowe . . . . .	6
Edytory tekstu . . . . .	6
Co jest w środku . . . . .	7
Nie bój się przerwań (5) . . . . .	7
Kasetowy system operacyjny . . . . .	8
KLAN COMMODORE	
Autonumber 64 . . . . .	9
Kopiowanie ekranu na drukarce . . . . .	9
Turbo 16 . . . . .	10
Magnetofon i Commodore Plus/4 . . . . .	11
KLAN SPECTRUM	
Użyteczny hazard . . . . .	12
Zmiana atrybutów . . . . .	13
KLAN AMSTRAD-SCHNEIDER	
Basic CPC-464 . . . . .	14
Co piszczy pod klawiaturą . . . . .	15
CO JEST GRANE	
Bobby Bearing . . . . .	16
Bajtkowa Lista Przebojów . . . . .	18
Silent Service . . . . .	18
Bomb Jack . . . . .	19
JAK TO ROBIĄ INNI	
CEBIT'87 . . . . .	20
Sir Clive powraca . . . . .	21
PODSTAWY	
Komputer już nie drewniany . . . . .	22
NASTĘPNY KROK	
Rozsądna oszczędność . . . . .	23
GIEDŁA . . . . .	26
SAMI O SOBIE	
Komputer Ci pomoże . . . . .	27
DROGI BAJTKU . . . . .	28
TYLKO DLA PRZEDSZKOLAKÓW	
Kubuś ratuje królową . . . . .	30
NIE TYLKO KOMPUTERY	
Biuro przyszłości . . . . .	31

„BAJTEK” — MIESIĘCZNY DODATEK DO „SZTAN-  
DARU MŁODYCH”

ADRES: 00-687 Warszawa, ul. Wspólna 61, Tel. 21-12-05  
Przewodniczący Rady Redakcyjnej: Jerzy Domański-  
redaktor naczelny „Sztandaru Młodych”.

ZESPÓŁ REDAKCYJNY: Waldemar Siwiński (z-ca redaktora naczelnego „SM” — kierownik zespołu „Bajtka”), Roman Poznański (z-ca sekretarza redakcji „SM” — sekretarz zespołu „Bajtka”), Krzysztof Czernek, Sławomir Gajda (red. techniczny), Andrzej Gogolewski, Andrzej Kowalewski, Andrzej Podulka, Sławomir Polak, Wanda Roszkowska (opr. graficzne), Kazimierz Treger, Marcin Wałigórski, Roman Wojciechowski. Zdjęcia w numerze: Leopold Dzikowski.

Klasy redagują:

Commodore — Klaudiusz Dybowski, Michał Silski, Amstrad-Schneider — Tomasz Pyc, Sergiusz Wolicki, Spectrum — Konrad Fedyna, Michał Szuniewicz, Atari — Wiesław Migut, Wojciech Zientara.

Fotoskład — Tadeusz Olczak,

Montaż offsetowy — Grażyna Ostaszewska, Korekta — Maria Krajewska, Ewa Mowińska.

WYDAWCA: RSW „Prasa-Książka-Ruch” Młodzieżowa Agencja Wydawnicza, al. Stanów Zjednoczonych 53, 04-028 Warszawa. Telefony: Centrala 13-20-40 do 49, Redakcja Reklamy 13-20-40 do 49 w. 403, 414.

Cena 100 zł.

Skład techniką CRT-200, przygotowalnia offsetowa i druk PRASOWE ZAKŁADY GRAFICZNE RSW „PRASA-KSIĄŻKA-RUCH” w Ciechanowie, ul. Sienkiewicza 51.

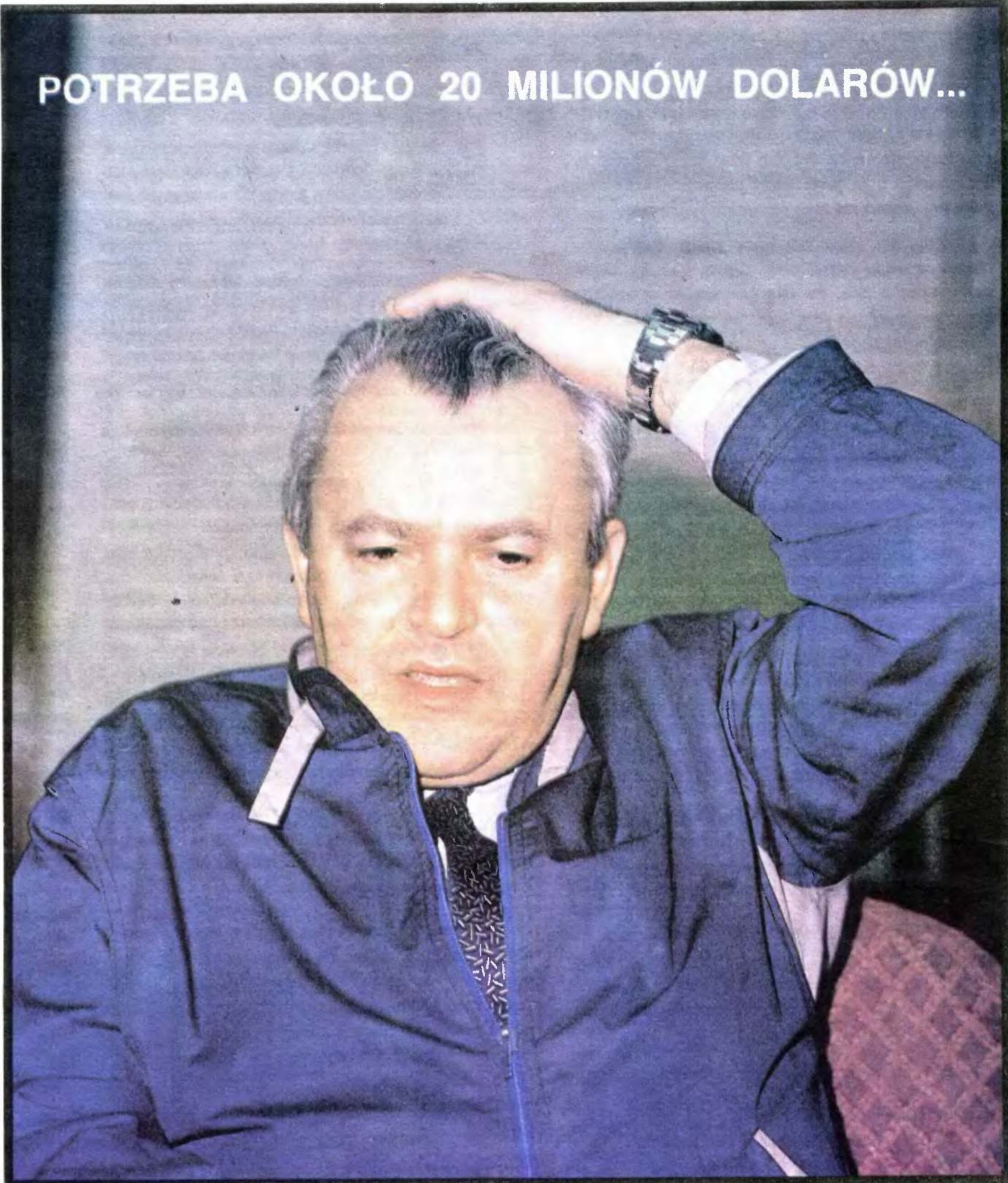
Zam. nr 072247, nakład 250 000 egz., K-109



Bajtek

# ZAWÓD Z PERSPEKTYWĄ

POTRZEBA OKOŁO 20 MILIONÓW DOLARÓW...



## Rozmowa z prof. Wiesławem Grudzewskim, przewodniczącym TMMT, podsekretarzem stanu w Urzędzie Postępu Naukowo- -Technicznego i Wdrożeń

— Nie widzimy, panie ministrze, na pana biurku ani monitora, ani klawiatury...

— Mamy w Urzędzie komputer IBM PC XT. Jest on jednak potrzebny przede wszystkim moim współpracownikom. Dla przykładu: cała dokumentacja współpracy

naukowo-technicznej z zagranicą będzie zarejestrowana, a częściowo już jest na twardych dyskach mikrokomputera. Ja dostaję na biurko gotową analizę lub odpowiedź na pytania. Dlatego, gdy przejdziemy na system komputerów IBM-AT, to będę być może także miał na biurku klawiaturę z monitorem, ale z pewnością najpierw otrzymają ją pracownicy bardziej tego potrzebujący, korzystający z niej w codziennej pracy.

— Komu jak komu, ale z pewnością współkierującemu Urzędem Postępu Naukowo-Technicznego i Wdrożeń musimy zadać pytanie wprost: kiedy na rynku pojawią się mikrokomputery o jakich marzą czytelnicy „Bajtki” — w cenach dostępnych dla ich rodziców?

— Niełatwo odpowiedzieć na to pytanie. Z pewnością pojawią się wkrótce komputery „Elwro 800 Junior” zbudowane głównie z myślą o edukacji informatycznej młodego pokolenia. Nie jestem jednak w stanie odpowiedzieć na pytanie: czy będą one dostępne dla przeciętnej kieszeni? Zdajemy sobie sprawę z potrzeby szybkiego wprowadzenia do produkcji takiego komputera i wydzielenia odpowiednich środków dla Ministerstwa Oświaty, aby mogło zakupić go dla szkół, a także klubów mikrokomputerowych, instytucji i organizacji tak potrzebujących dla uczniów i swoich członków odpowiedniego wyposażenia komputerowego.

— Chcieliśmy jednak usłyszeć coś więcej niż tylko uwagę o konieczności podjęcia produkcji — z tego wszyscy zdajemy sobie sprawę.

— Zarówno „Elwro 800 Junior”, jak i jego profesjonalny odpowiednik „Elwro 800” — to komputery nowoczesne, z tym, że „Junior” akceptować powinien także oprogramowanie Sinclaira. Będziemy go produkować w ilości 100 tys. sztuk; mam na myśli różne jego mutacje. W szkołach wyższych będą potrzebne profesjonalne mikrokomputery o lepszych niż „Junior” parametrach — „Mazovia” i „Elwro 800”. Tych produkować powinniśmy po 30 tys. rocznie. Trudno wymagać jednak ode mnie abym dokładnie podał datę rozpoczęcia produkcji, cenę itd. Samodzielne i samorządne przedsiębiorstwa, które mają produkować mikrokomputery same decydują o tym jaki jest ich program produkcji i termin jego realizacji.

— Ale władze dysponują możliwościami wpływania na profil produkcji przedsiębiorstw, m.in. poprzez system zamówień rządowych.

— Zamówienie, na te komputery zostało złożone. Na razie nie otrzymaliśmy jeszcze założeń techniczno-ekonomicznych tych przedsięwzięć, a więc i umowy między nami a producentami wciąż są jeszcze wstępne. Potrzeba także na uruchomienie projektu poważ-

# GRA O JUTRO

nich środków dewizowych — ok. 20 mln dolarów. Poza tym IBM rozpoczął już produkcję mikrokomputera 32-bitowego. Boimy się, czy dzisiejszy sprzęt klasy XT lub nawet AT nie okaże się wkrótce przestarzały. Podejmowane są zatem również przyspieszone działania mające na celu zapewnienie w przyszłości rozwoju mikrokomputerów na dobrym poziomie europejskim i dla edukacji, i dla zastosowań profesjonalnych. Komputer to przecież niezmiernie ważne narzędzie.

— **Zbyt nieśmiało bierzemy się za unowocześnienie procesu kształcenia. A przecież zgodził się pan chyba z tym, że bez powszechnej edukacji komputerowej nie jest możliwe dokonanie skoku technologicznego?**

— Rzecz chyba nie tylko w informatyce. Wszyscy zastanawiają się, jakie są podstawy sukcesu japońskiego. I dochodzą do przekonania, że jednym z głównych czynników, — może ważniejszym od tradycyjnego stosunku do pracy — jest dobra edukacja, sprawny system edukacyjny. Dużą wagę przykładają się w nim do wykształcenia ogólnego, również wielką do wykonywania przez uczniów samodzielnych zadań w oparciu o nowoczesny sprzęt, laboratoria itd. Zadania z informatyki, wiedzy komputerowej rozwiązuje się w nich w oparciu o wiadomości wyniesione z lekcji fizyki, matematyki, czy nawet języka japońskiego lub angielskiego. A więc system szkolnictwa opiera się na dobrym przygotowaniu z przedmiotów ogólnych, wychowaniu politechnicznym, językach obcych. Wymagania są wysokie a młody człowiek ma wiele możliwości tworzenia. Jest zatem rzeczywiście kształcony w sposób twórczy.

— **Jak z punktu widzenia pana — odbiorcy „produktów” pracy szkół — wygląda przygotowanie naszych uczniów do rozwijania postępu naukowo-technicznego?**

— Nasze szkoły, zarówno średnie jak i wyższe, nie przygotowują wcale źle. Niestety, obniżyła się mocno dyscyplina studiów. Nastąpiło też poważne zubożenie laboratoriów i warsztatów. Są one już mocno przestarzałe.

— **Ale czy to już odbija się na jakości absolwentów?**

— W wielu dziedzinach jesteśmy zapóźnieni, np. w informatyce. I w sprzęcie i w kształceniu kadr.

— **Czy nie wynika to także z niedostosowania struktury kształcenia do potrzeb przyszłości. Jeżeli określamy kilka najważniejszych dla postępu naukowo-technicznego dziedzin, to właśnie w nich głównie powinniśmy kształcić.**

— Nowy program elektronizacji wymaga oczywiście wielu elektroników. Jednak już dotąd kształciliśmy ich też dużo. Oczywiście nowe kierunki muszą się rozwijać. Na ile to się będzie udawać zależy od kadry, wyposażenia laboratoriów itd.

— **Powróćmy do komputerów. Jesteśmy dziś na etapie zabawy. Czy jednak możemy ze spokojnym sumieniem polecać młodym ludziom zajmowanie się informatyką na poważnie? Czy będą np. mogli na tym zarobić?**

— Myślę, że tak. Jest to nowy język porozumienia międzyludzkiego, tak jak wcześniej malarstwo czy znaki stosowane w matematyce. Tym językiem trzeba nauczyć się operować. Wkracza on do medycyny, projektowania inżynierskiego, usług, rolnictwa, humanistyki. Jeżeli mamy ambicje być krajem zajmującym znaczące miejsce na mapie gospodarczej świata nie uciekniemy od informatyki. Trudno powiedzieć jak będzie wyglądało oprogramowanie za lat 20 czy 30 ale na pewno dobre podstawy informatyki pozwolą łatwiej zdobyć tę wiedzę. To przyszłościowa profesja. Równie ważni jak informatycy-specjaliści będą i ci, którzy potrafią korzystać z komputerów w swoich dziedzinach. Taka wiedza wkrótce stanie się przepustką do kultury podobną do tej jaką dziś daje umiejętność czytania i pisanie.

— **Mówiąc o przyszłości, o planach produkcji komputerów przyjęliśmy jakby milcząco, że jesteśmy zdani na własne siły. A przecież możliwa jest współpraca w ramach RWPG.**

— Oczywiście, że nie będzie nas nawet stać na to, by rozwijać informatykę i przemysł komputerowy we wszystkich kierunkach. Potrzebny jest zatem podział zadań między krajami naszej wspólnoty. Trzeba tworzyć i mikrokomputery, i superkomputery, i sieci. Do tego niezbędne jest oprogramowanie. Te zadania trzeba rozwiązywać wspólnie. Tym bardziej, że na wiele produktów i technologii zachodnich nałożone jest embargo.

— **Jakie miejsce dla Polski widziałby Pan w tym**

**podziale pracy? W czym chcemy się specjalizować?**

— W określonych klasach mikrokomputerów, w budowie sieci i drukarek. Mamy tutaj chyba najwięcej doświadczeń i szans.

— **A mikroelektronika: mikroprocesory, pamięci itd?**

— Z pewnością wszystkich podzespołów produkować nie będziemy, ale część trzeba. Już dziś odczuwamy bardzo silnie ich deficyt. Brakuje jednak podstawowych materiałów, jak: polikryształy krzemu, materiały foliowane, półprzewodniki czynne. Produkcja jest bardzo kosztowna i często szkodliwa dla środowiska. Wymagania technologiczne też są bardzo wysokie. Dyskusja z partnerami zagranicznymi na te tematy ciągle trwa. Dla każdego z partnerów w RWPG wygodniej byłoby chyba wytwarzać produkty finalne.

— **Chcieliśmy, aby na zakończenie określił pan profesora zadania dla czytelników „Bajtki”. Czego by pan od nich oczekiwał?**

— Chciałbym, aby od zabawy przeszli do używania mikrokomputerów, przy rozwiązywaniu zadań i problemów z poszczególnych przedmiotów — matematyki, fizyki, wytrzymałości materiałów itd. Niech od prób odpowiedzi na pytanie: jak zbudować dom, jak rozplanować mieszkanie, opracować jakiś katalog dla szkoły przejdą do praktycznych działań jak kierować odcinkiem produkcji jak opracować projekt organizacji budowy, jak analizować koszty wyboru wariantu inwestowania w oparciu o określone przedsięwzięcie naukowo-techniczne. Trzeba nauczyć się formułowania konkretnych zadań i metodyki ich rozwiązywania rozpoczynając oczywiście od uproszczonych sytuacji czy modeli. Uczniowie starszych klas mogą się już podejmować zadań ściśle użytkowych dla swojej szkoły, ośrodka zdrowia, czy jakiegoś zakładu usługowego. Możecie sami, jako redakcja „Bajtki” ocenić ich wysiłki, a ja podejmę się zaproponować nagrodę dla najlepszych.

— **Trzymamy za słowo i dziękujemy za wywiad.**

Rozmawiali:  
Grzegorz Onichimowski  
Waldemar Siwiński

## INFOSYSTEM'87

**Wrocławski INFOSYSTEM to druga już w tym roku po warszawskiej HOME PERSONAL COMPUTER prezentacja firm, komputerów i oprogramowania.**

Wrocławska oferta była bardzo podobna do warszawskiej. Dominowały komputery IBM-kompatybilne oraz ich peryferia. Sensacją tradycyjnie wywoływały komputery produkcji polskiej. Oslawiona MAZOVIA 1016 i komputer szkolny ELWRO Junior to komputery na które czekamy już od ponad roku, a ich producent spółka z o.o. Mikrokomputery prezentuje ciągle te same, lekko już zakurzone egzemplarze.

Według opracowanej przez Instytut Matematyki prognozy, powinniśmy do roku 2000 wyprodukować 300 tys. mikrokomputerów dla nauki, gospodarki i rynku prywatnego oraz 700 tys. dla szkolnictwa. Razem więc okrąży milion. Wydawać by się mogło, że to sporo, choć np. RFN tą liczbą mikrokomputerów dysponowało już w 1985 r.

Czy 300 tys. komputerów dla potrzeb nauki, gospodarki oraz rynku w roku 2000 wystarczy? To pytanie zadawano również w Zakładach Elektronicznych ELWRO, gdzie w czasie trwania wystawy odbyło się wyjazdowe posiedzenie Prezydium Komitetu d/s Nauki i Postępu Technicznego przy Radzie Ministrów na temat oceny stanu i perspektyw rozwoju techniki mikrokomputerowej w naszym kraju.

Kierownik Urzędu Postępu Naukowo-Technicznego minister Konrad Tott stwierdził, że: „prognoza jest kompromisem między potrzebami a możliwościami. Trzeba sobie zdać sprawę z tego, że realizacja tych planów wymaga kilkudziesiętkrotnego zwiększenia produkcji mikrokomputerów w Polsce, do poziomu 80-100 tys. sztuk rocznie.”

W ramach Centralnego Planu Elektronizacji, na który przewidziano w sumie 35 mld zł, nakłady na technikę mikrokomputerową wyniosą 5 mld zł.

Plany na 1988/89 zakładają następującą produkcję urządzeń komputerowych:

### Mikrokomputery:

ELWRO 800, (ELWRO) — 30.000 szt.  
MAZOVIA 1016, (MIKROKOMPUTERY) — 30.000 sztuk.  
MERITUM, (MERA ELZAB) — 5.000 szt.  
KRAK, (KFAP) — 1000 szt.  
MERA 660, (MERA) — 10.000 szt.  
COMPAN 16, (MERA ELZAB) — 1.000 szt.  
Mikrokomputer szkolny, (...) — 100.000 szt.

### Drukarki:

D 100, 50.000 szt.

### Monitory:

Ekranowe (trzeciej generacji) MERA ELZAB — 30.000 szt.

Kolorowe monitory graficzne POLKOLOR — 30.000 sztuk.

### Napędy dysków elastycznych

ED5SL (slim line) — 50.000 szt. w roku 1990

### Plotery i mikrodrukarki graficzne:

Ploter płaski MERA 621 (format A3) 1800 szt. — (1990 r.)

Ploter rolkowy MERA 630 (A3 i A4) 2000 szt. — (1989 r.)

Mikrodrukarka graficzna MDG1 25.000 szt. — (1990 r.)

W ramach Centralnych Programów Badawczo-Rozwojowych i Zamówień Rządowych prowadzi się także prace nad dyskami twardymi (FMIK ERA), monitorami (POLKOLOR) i klawiaturami (MERA-REFA), a w ramach zamówienia rządowego ZRN 8.28 planuje się rozpoczęcie produkcji dyskietek 5 1/4 cala (MERAL) w ilości 6 mln sztuk rocznie.

Jedną z podstawowych barier w wykonaniu tych planów jest zapóźnienie technologiczne w przemyśle półprzewodników. Dyrektor Instytutu Maszyn Matematycznych, Bronisław Piwowar powiedział wprost: „W produkcji układów scalonych jesteśmy na szarym końcu w Europie. Świat wdrożył już dawno mikroprocesory 32-bitowe i przygotowuje kolejne generacje układów, a my ciągle nie możemy uruchomić produkcji procesora 16-bitowego...”

Jest jednak o co walczyć. Dał temu wyraz minister Konrad Tott: „Jestem przekonany, że pomimo dużego zadłużenia nakłady jakie poniesiemy na rozwój elektronizacji przyniosą w latach późniejszych ogromne zyski”.

Stawomir Polak

W języku LOGO dysponujemy wyłącznie dynamicznymi strukturami danych w postaci list i słów (liczby w tym języku są również słowami). Pod tym względem LOGO różni się znacznie od tzw. klasycznych języków programowania, w których najczęściej używaną złożoną strukturą danych są tablice. Okazuje się, że i w LOGO jest to możliwe. Problem rozwiązano w dość zaskakujący z punktu widzenia innych języków programowania, lecz też charakterystyczny dla LOGO sposób.

Na pierwszy rzut oka bardzo łatwo jest przedstawić tablicę jako listę. I tak na przykład tablicę jednowymiarową o  $n$  elementach możemy przedstawić jako listę  $n$ -elementową; z kolei tablica dwuwymiarowa  $m * n$  byłaby listą  $m$  list  $n$ -elementowych, i tak dalej. Wartość wybranego elementu tablicy łatwo jest odczytać przy pomocy operacji ITEM.

Niestety istnieje pewna bardzo poważna niedogodność tego rozwiązania. Mianowicie ma możliwość — w prosty sposób — zmienić wartości wybranego elementu listy. Stosunkowo łatwo można zmodyfikować tylko pierwszy i ostatni jej element — przy pomocy kombinacji odpowiednio BUTFIRST i FPUT lub BUTLAST i LPUT. Modyfikacja elementu wewnątrz listy wymaga wielokrotnego użycia tych operacji oraz tworzenia pomocniczych list, co niepomiarowo wydłuża czas działania programu.

## TABLICE W LOGO

Przedstawianie tablic w postaci list nie jest zatem rozwiązaniem najkorzystniejszym. Przedstawiony program korzysta z całkiem innego sposobu implementacji tablicy — z wykorzystaniem nie list, lecz słów. Jego realizację prześledzimy na przykładzie tablic dwuwymiarowych.

Pomysł polega na spostrzeżeniu, że każdy element dwuwymiarowej tablicy o nazwie  $X$  jest jednoznacznie wyznaczony przez trzy parametry: nazwę tablicy oraz parę współrzędnych elementu. Wszystkie te parametry są słowami — można je zatem skleić w odpowiedni sposób, tworząc jedno nowe słowo, określające element tablicy również w sposób jednoznaczny. Słowo to możemy potraktować jako nazwę zmiennej i nadać mu wartość poprzez MAKE. Cała tablica będzie zatem zbiorem  $m * n$  niezależnych zmiennych.

Tych kilka zdań powyżej możemy streścić w jednej operacji LOGO:

```
to Elt :Tablica :i :j
  op (word :Tablica "# :i ", :j)
end
```

Operacja Elt produkuje właśnie nazwę zmiennej, odpowiadającą wskazanemu elementowi tablicy. Oto przykład jej użycia w odniesieniu do tablicy "Alfa".

```
print Elt "Alfa 1 1
Alfa#1,1
make (Elt "Alfa 1 1) "Bajtek
```

```
print thing Elt "Alfa 1 1
Bajtek
print :Alfa#1,1
Bajtek
```

Jak widać, tworzona zmienna nosi nazwę "Alfa#1,1". Zapis to może niekonwencjonalny, ale starałem się uniknąć używania w nazwie zmiennej nawiasów, będących w LOGO symbolami specjalnymi (można oczywiście zmieniać konwencję zapisu według swego uznania). Widzimy za to, że do elementu tablicy łatwo się odwołać zarówno przy pomocy operacji Elt, jak też poprzez bezpośrednie podanie nazwy.

Powyzsza operacja już wystarcza do skutecznego posługiwania się tablicami w programach LOGO. Zauważmy jednak, że nie daje nam ona możliwości sprawdzenia, czy współrzędne danego elementu mieszczą się w zakresie tablicy. Tego rodzaju kontroli dokonuje np. PASCAL. Możemy i my utworzyć podobny system kontroli.

```
to Dim :Tablica :Zakres_1 :Zakres_2
  make word :Tablica "?1 :Zakres_1
  make word :Tablica "?2 :Zakres_2
end
```

```
to Elt :Tablica :i :j
  if not and (Nalezy :i thing word :Tablica "?1)
    (Nalezy :j thing word :Tablica "?2)
    [Bład se [Indeks poza zakresem tablicy] :Tablica]
  op (word :Tablica "# :i ", :j)
end
```

```
to Nalezy :x :Zakres
  op and (not :x < first :Zakres)
    (not :x > last :Zakres)
end
```

```
to Bład :Komunikat
  print :Komunikat
  toplevel
end
```

Procedura Dim służy teraz do deklaracji rozmiarów tablicy; może ona być indeksowana przez dowolne przedziały liczb całkowitych, np.

```
Dim "Alfa [0 9] [0 9]
Dim "Beta [-1 1] [100 200]
```

Odwołanie do elementu spoza zadeklarowanych zakresów powoduje wskazanie błędu w programie:

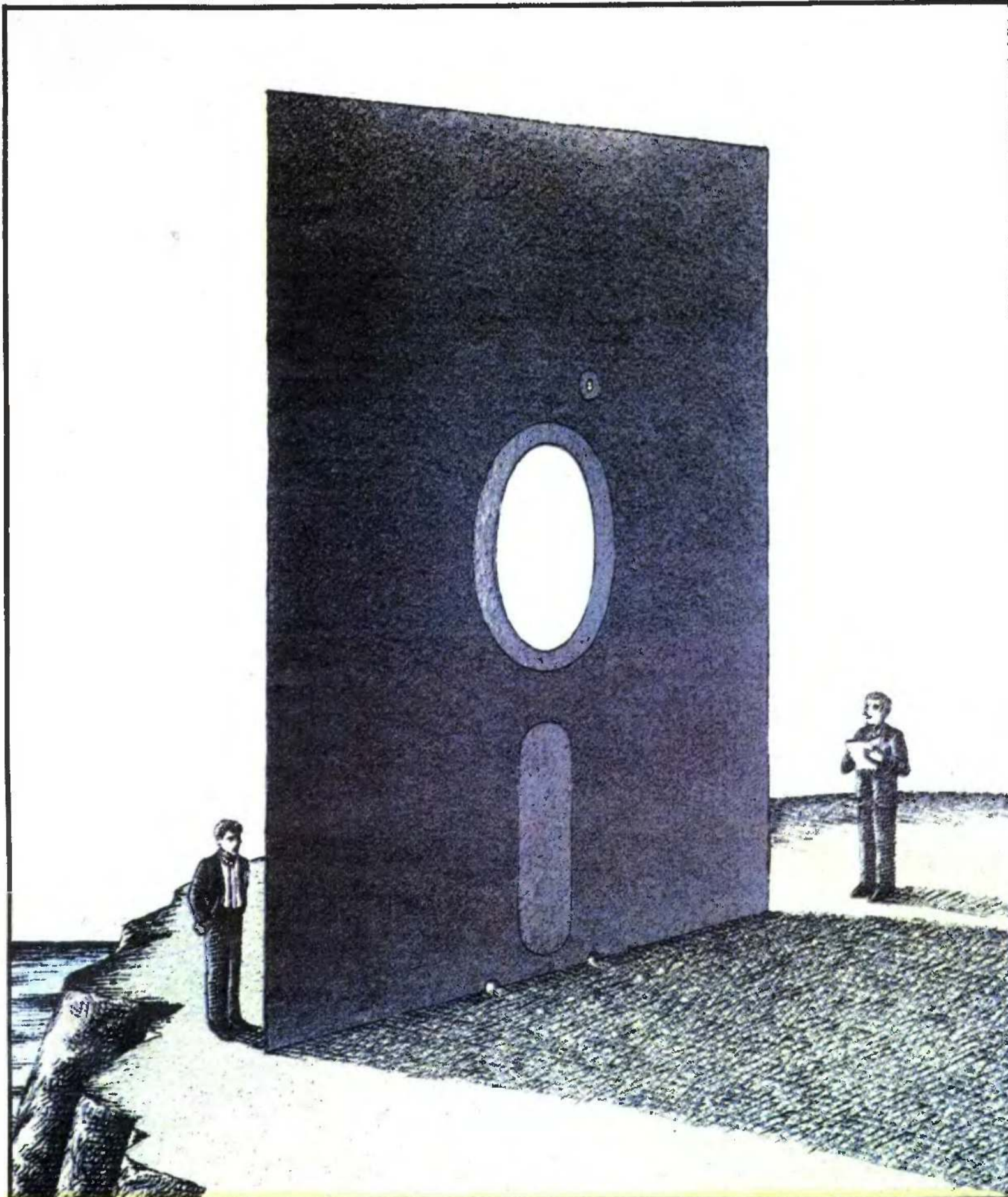
```
make (Elt "Alfa 10 0) -1
Indeks poza zakresem tablicy Alfa
```

Na zakończenie przykład zastosowania naszych procedur: utworzymy tabliczkę mnożenia liczb od 1 do 10.

```
to Tabliczka
  erns
  dim "Iloczyn [1 10] [1 10]
  make "k 0
  repeat 10 [ make "k :k + 1
    make "j 0
    repeat 10 [ make "j :j + 1
      make (Elt "Iloczyn :k :j) (:k # :j)]]
end
```

Po wywołaniu Tabliczka możemy uzyskać wydruk naszej tablicy Iloczyn, wykonując po prostu PONS.

Marcin Waligórski



## CIĄGI TEKSTOWE

**Rysowanie niewielkich, skomplikowanych, kolorowych obrazków przy użyciu instrukcji PLOT i DRAWTO okazuje się bardzo denerwujące. Jeżeli po kilku nieudanych próbach będziesz już na tyle zły, aby rzucić komputer w kąt i powiedzieć „Dość!”, to spróbuj użyć prostej i ciekawej metody „ciągów tekstowych”.**

W przypadku rysowania długich prostych linii nie będzie ona tak wygodna jak PLOT i DRAWTO, ale jeżeli chciałbyś narysować niewielki, kolorowy obrazek w jednym z trybów graficznych od 3 do 7, niewątpliwie „ciągi” okażą się bardzo pomocne i sprawią Ci wiele radości.

Do rysowania używa się instrukcji PRINT (!) oraz ciągów tekstowych składających się w trybach graficznych 3,5 i 7 z czterech liter A B C D, zaś w trybach 4 i 6 z dwóch A i B. Każda z liter „wydrukowana” na ekranie w określonym miejscu zabarwi pixel na inny kolor. W trybach 3,5 i 7 literom odpowiadają następujące rejestry koloru:

- A — COLOR 1 (rejestr 0)
- B — COLOR 2 (rejestr 1)
- C — COLOR 3 (rejestr 2)
- D — COLOR 4 (rejestr 3)

Natomiast w trybach 4 i 6:

- A — kolor pixela
- B — kolor tła

Spróbujmy więc zbudować prosty rysunek w trybie graficznym 5 o wymiarach 10\*10 pixeli. W tym celu musimy napisać dziesięć dziesięcioznakowych ciągów, np.:

```
ciąg 1 CCCCCCAAAA
ciąg 2 DDDDBDDDDA
ciąg 3 DDBDBDADADA
ciąg 4 DDBDDADDDC
ciąg 5 DBDCDADDDC
ciąg 6 BDDADDDDDC
ciąg 7 DBDADCDBDC
ciąg 8 DDADDDDDDC
ciąg 9 DDDDBDDDDC
ciąg 10 DDDDBDDDDC
```

Pozostało już tylko umieszczenie utworzonych ciągów w krótkim, prostym programie:

```
10 GRAPHICS 5+15
20 DIM ST$(100)
30 ST$="CCCCCAAAAADDDDBDDDDADDDDBDDADA
   DDBDDADDDCDBDCDADDDCDBDDDDADDDCDBDADDDCDB
   DDDADDDDDCDDDBDDDDCDDDBDDDDC"
40 X=5:Y=5
80 FOR I=1 TO 10
90 POSITION X,Y+I-1
100 PRINT #5;ST$(I*10-9,I*10)
110 NEXT I
120 GOTO 120
```

Teraz wystarczy chwilę pomyśleć i wszystko będzie jasne. Jeżeli zrozumiałeś, to bardzo dobrze, bo spróbujemy teraz trochę pomanipulować naszym rysunkiem, np. możemy go odwrócić o 90

stopni, wystarczy tylko w naszym programie zmienić linie 20 i 100 oraz dopisać trzy nowe:

```
20 DIM ST$(100),OB1$(100)
50 FOR I=1 TO 10:FOR K=1 TO 10
60 OB1$(I*10-10+K,I*10-10+K)=ST$((10-K)
  )*10+I,(10-K)*10+I)
70 NEXT K:NEXT I
100 PRINT #5;OB1$(I*10-9,I*10)
```

Aby obrócić pierwowzór o 180 stopni wystarczy w utworzonym przed chwilą programie zmienić linie 50, 60 i 70 na:

```
50 FOR K=1 TO 100
60 OB1$(K,K)=ST$(101-K,101-K)
70 NEXT K
```

Natomiast, aby uzyskać lustrzane odbicie trzeba w pierwszym programie zmienić linię 100:

```
100 PRINT #5;ST$((11-I)*10-9,(11-I)*10)
```

Jak widać, manipulacja rysunkami nie jest trudna, może więc spróbujesz sam np. w prosty sposób zamienić kolory poszczególnych pixeli, wykonać obrót o 270 stopni albo połączyć ruch ze zmianą barw (?!). Możesz spróbować także utworzyć przy pomocy ciągów napisy na ekranie. Oczywiście Twoje rysunki nie muszą mieć wymiarów 10\*10 pixeli, ale dowolne np. 4\*6, 8\*8, 4\*12 itp. PRZYJEMNEJ ZABAWY!

*Tomasz Mazur  
Sergiusz Piotrowski*

## EDYTORY TEKSTU

**Artykułem tym rozpoczynamy cykl publikacji o programach użytkowych na komputery Atari. Opiszemy w nich kolejno najpopularniejsze programy edytorskie, graficzne, muzyczne, bazy danych i programy kalkulacyjne.**

Pierwszy odcinek poświęcimy programom służącym do redagowania listów, dokumentów i innych tekstów — tzw. edytorom tekstu.

### **AtariWriter**

(1982 — Atari Inc.)

Jeden z najbardziej popularnych edytorów, zaprojektowany jeszcze dla Atari 400/800. Ma wbudowane możliwości współpracy z tanimi drukarkami Atari (1025, 1027, 1029). Wymaga określenia formatu wydruku na początku dokumentu (późniejsze zmiany są jednak możliwe). Zalety: prosta obsługa, możliwość pisania w dwóch kolumnach oraz przegląd strony przed drukiem. Wady: 40 znaków w wierszu przy przeglądaniu, możliwość pracy tylko z jedną stacją dysków i konieczność potwierdzania dużej części poleceń.

### **AtariWriter Plus**

(1985 — Atari Corp.)

Znacznie rozbudowany w stosunku do poprzedniego, jest już prawie profesjonalnym edytorem. Występuje w dwóch wersjach: na 800XL i 130XE. Umożliwia pracę z dwoma stacjami dysków oraz tworzenie bloków adresowych szczególnie przydat-

nych przy wysyłaniu większej liczby pism o tej samej treści do różnych adresatów. Dodatkowo posiada możliwość kontroli poprawności ortograficznej wprowadzanego tekstu przez porównywanie ze słownikiem (zawartym na dysku).

### **SpeedScript 3.0**

(1984 — Charles Brannon)

Prosty i wygodny w obsłudze, uniwersalny edytor. Nie jest przystosowany do żadnej konkretnej drukarki, a dzięki możliwości definiowania kodów ASCII łatwo jest dopasować go do aktualnie używanej. Przeniesienie go na kasetę jest bardzo proste — praktycznie jest to jedyny edytor stosowany przez posiadaczy magnetofonów. W tworzonym pliku tekstowym nie zapisuje żadnych dodatkowych informacji, dzięki temu może odczytywać wszystkie pliki zapisane w kodzie ASCII. Używany jest w naszej redakcji przy przygotowywaniu do druku listingów programów.

### **Home Word**

(1984 — Target Technologies Inc.)

Bardzo wygodny w obsłudze, wszystkie funkcje wybierane są rysunkami, więc praktycznie instrukcja do niego jest zbędna. W czasie pisania tekstu na bieżąco pokazuje wygląd strony po wydrukowaniu oraz pozostały obszar wolnej pamięci, umożliwia również przegląd gotowego dokumentu w trybie 80 znaków w wierszu. Wadami są: długi czas wykonywania poleceń redakcyjnych, niewielki obszar pamięci dla tworzonego dokumentu i niestandardowy

format zapisu na dysku. Najczęściej przeze mnie używany — na nim powstał m.in. ten artykuł.

### **Lister +**

(1984 — Chet Walters)

Nietypowy edytor. Umożliwia pisanie różnorodnymi zestawami znaków (w zasadzie jedyny edytor pozwalający na stosowanie polskich liter bez ingerencji w program). Druk wykonywany jest w trybie graficznym po jednej linii (maksymalnie 120 znaków w wierszu, jeśli mamy taką drukarkę). W ograniczonym zakresie może pracować z drukarką Atari 1029 (niektóre funkcje wykonywane są przez nią nieprawidłowo, np. druk w podwójnej i poczwórnej szerokości znaków).

### **GT Estate Word Processor**

(1983 — ELCOMP Publishing)

Niezbyt wygodny w użyciu i o mniejszych możliwościach niż pozostałe. Jediną zaletą jest możliwość uzyskania do 250 znaków w wierszu, wymaga to jednak odpowiedniej drukarki. Stosunkowo mało popularny w Polsce.

Specjalną odmianą edytorów jest seria czterech programów firmy X-Ent Software: **Page Designer**, **Typesetter**, **Rubber Stamp** i **Megafont II+**. Stanowią one połączenie edytorów tekstowych z programami graficznymi i umożliwiają łączny druk grafiki i tekstów. Opis tych programów wraz z przykładem zastosowania zamieszczony był w „Komputerze” 8/86.

*Wojciech Zientara*

## Podstawą programowania w języku maszynowym jest znajomość sprzętu. W naszym nowym cyklu postaramy się opisać działanie komputera w możliwie najbardziej przystępny sposób. Zakładamy, że znane wam są już pojęcia takie jak bit i bajt.

Zacznijmy od blokowego schematu budowy wewnętrznej komputera. Na rysunku 1 widzimy układy znajdujące się w komputerze i schemat komunikacji między nimi. Wszystkie elementy niepokazane na rysunku (w tym także klawiatura) to urządzenia zewnętrzne. Pracą całego systemu kieruje jednostka centralna (Central Processor Unit — CPU), którą w tym przypadku jest mikroprocesor MOS6502C. CPU pobiera z pamięci kody kolejnych rozkazów programu i dane do nich, a następnie przesyła polecenia lub informacje do innych elementów systemu. Dodatkowe wejścia procesora służą do odbierania sygnałów od pozostałych układów.

Oczywiste jest już, że muszą istnieć układy, w których magazynowany jest program mikroprocesora i dane do niego. Takimi układami są pamięci. Rozróżniamy dwa rodzaje pamięci: stałą, z której informacja może być tylko odczytywana (Read Only Memory — ROM) i zmienną, tzw. pamięć o dostępie swobodnym (Random Access Memory — RAM), z której można dane odczytywać i do której można je zapisywać. Ze względu na koszt pamięć RAM jest zwykle dynamiczna, tzn. że zapisana w niej informacja znika po wyłączeniu zasilania. Pamięć ROM natomiast zawsze zachowuje swoją zawartość. W związku z tym w pamięci ROM zapisane są procedury umożliwiające uruchomienie komputera po włączeniu zasilania (tzw. zainicjowanie systemu) oraz jego dalszą pracę i komunikację z urządzeniami zewnętrznymi. Zestaw procedur zawartych w ROM-ie zwany jest systemem operacyjnym komputera (Operating System — OS). W pamięci RAM zapisywane są (podczas inicjowania systemu) dane niezbędne do pracy OS oraz programy użytkownika wprowadzane z urządzeń zewnętrznych. Pamięci (RAM i ROM) podzielone są na komórki, z których każda ma swój adres i zawiera jeden bajt informacji. Aby odczytać lub zapisać jakąś wartość, procesor musi podać adres komórki. Służy do tego celu szyna adresowa. Od jej wielkości zależy wielkość obszaru pamięci, jaką

może zaadresować procesor. W komputerach domowych najczęściej jest stosowana szyna 16-bitowa, co daje  $2^{16} = 65536$  możliwych adresów, a więc 64 kilobajty ( $64 \cdot 1024 = 65536$ ). Ponieważ dane są 8-bitowe (1 bajt), to do określenia adresu potrzeba dwa bajty. Dla uproszczenia przyjęto nazywać obszar pamięci, który może być zaadresowany jednym bajtem, stroną pamięci (256 bajtów). Tak więc starszy (HIGH) bajt adresu wskazuje stronę, a młodszy (LOW) komórkę na stronie —  $ADR = 256 \cdot HI + LO$ .

Rysunek 2 przedstawia schemat pamięci najpopularniejszych modeli Atari (600XL, 800XL i 130XE). Widać wyraźnie, że poszczególne modele różnią się tylko wielkością pamięci RAM. Ponieważ procesor może zaadresować bezpośrednio jedynie 64 KB, to niektóre obszary pamięci pokrywają się. Jednocześnie może być użyty tylko jeden z pokrywających się obszarów. O tym, który z nich, decyduje specjalny układ kontrolera pamięci (Memory Management Unit — MMU). Korzystając z informacji otrzymanych od CPU włącza on lub wyłącza odpowiednie obszary pamięci.

Procedura inicjująca zapisuje w pamięci RAM informacje dotyczące organizacji pa-

## CO? JEST W ŚRODKU (CZ. 1)

mieci komputera. Informacje te są potem wykorzystywane podczas pracy systemu operacyjnego oraz mogą być odczytywane i modyfikowane przez użytkownika. Oto najważniejsze z tych komórek:

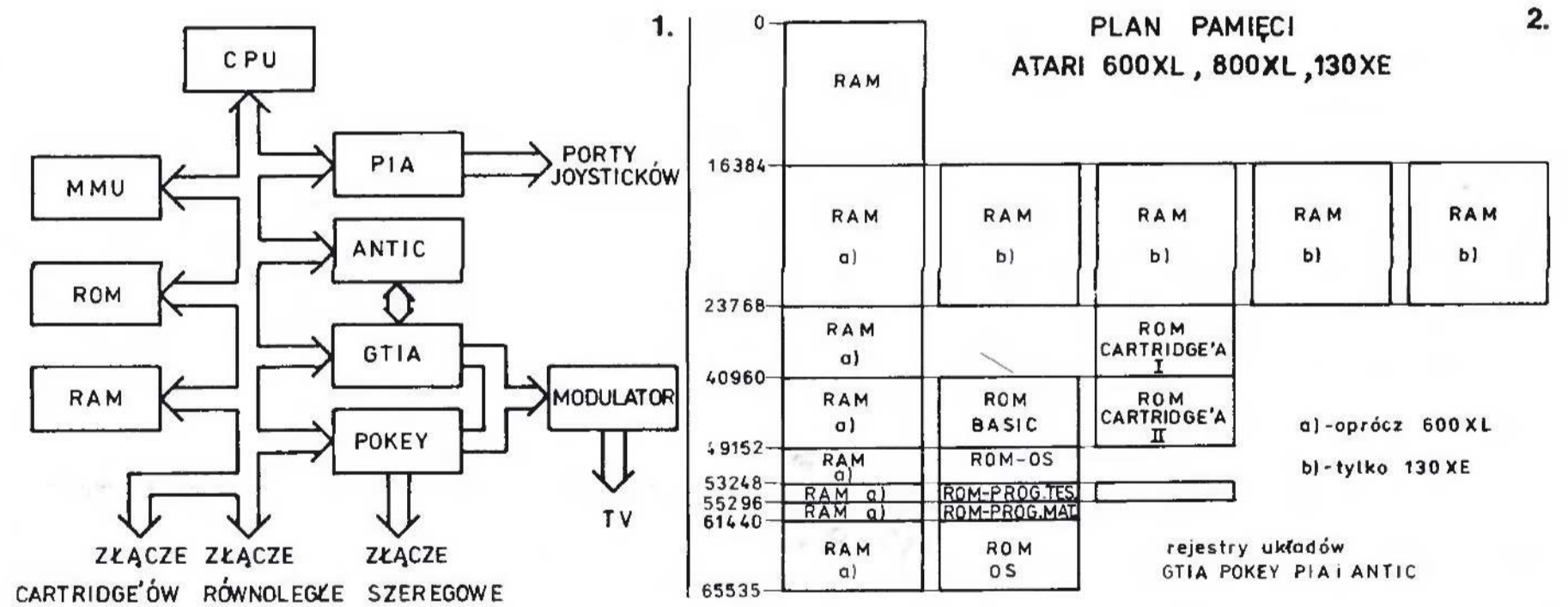
- 106 RAMTOP Wielkość RAM w stronach.
- 740 RAMSIZE Wielkość RAM w stronach.
- 741–742 MEMTOP Koniec wolnej pamięci RAM (powyżej jest zwykle pamięć obrazu).
- 743–744 MEMLO Pierwszy adres wolnej pamięci RAM.
- 14–15 APPMHI Najwyższy adres zajmowany przez program użytkownika.
- 128–129 LOMEM Początek programu BASIC-a.
- 144–145 BASMEMTOP Koniec programu BASIC-a.

Nie wspominałem dotąd o BASIC-u. Rzeczywiście w pamięci ROM jest wbudowany interpreter tego języka. Jest to jednak język dobry jedynie dla producentów. Jego interpretery są stosunkowo proste do za-

programowania i przez to tanie. Niestety z tego powodu większość pracy jest przerywana na użytkownika. Poza tym interpreter ATARI BASIC jest bardzo wolny. Najważniejszą wadą BASIC-a jest to, że każdy komputer ma inną jego wersję. Najlepiej wiedzą o tym Czytelnicy działu „Tylko dla przedszkolaków”. Inne języki, jak PASCAL, LOGO czy C w znacznie mniejszym stopniu zależą od sprzętu i przeniesienie programów źródłowych w tych językach nie stanowi żadnego problemu nawet dla początkującego programisty. Oprócz tego BASIC przez swoją składnię wyrabia złe nawyki, które utrudniają poprawne programowanie. Najlepszym dowodem na powyższe stwierdzenia jest ilość różnych wersji BASIC-a, które poprzez rozbudowę próbują go zbliżyć do języków strukturalnych (np. Pascal). Dlatego też opis ten nie będzie związany z żadnym konkretnym językiem programowania.

W kolejnych odcinkach zostaną przedstawione pozostałe układy komputera oraz wykorzystywane przez nie obszary pamięci.

Wojciech Zientara



# NIE BÓJ SIĘ PRZERWAŃ (5)

Ostatnim przerwaniem jakie chcę opisać jest przerwanie wywołane przez program w języku maszynowym. Tym razem niestety bez przykładu (byłby zbyt długi). Myślę jednak, że mimo to opis będzie zrozumiały.

W zestawie instrukcji mikroprocesora 6502 występuje rozkaz BRK. Jest to polecenie wykonania przerwania. Stosuje się go najczęściej podczas testowania programów napisanych w języku ma-

szynowym. W takim przypadku procedura przerwania wypisuje na ekranie lub drukarce stan wszystkich rejestrów mikroprocesora. Oczywiście procedura ta musi być również napisana przez użytkownika. I to już prawie wszystko.

Prawie, bo jeszcze musimy wiedzieć, gdzie komputer będzie szukał procedury przerwania BRK. Otóż adres tej procedury jest zapisany w rejestrze VBREAK

(518–519 — Vector BREAK). W komórce 518 jest zawarty młodszy bajt adresu procedury, a w komórce 519 starszy. Wystarczy zmienić go tak, aby wskazywał początek naszej procedury i po każdej instrukcji BRK komputer wykona naszą procedurę.

Oczywiście system operacyjny Atari przewiduje znacznie więcej różnych przerwania. Są one używane przez system operacyjny i ingerowanie w nie wymaga dobrej

znajomości komputera i jego systemu. Z tego powodu ograniczyłem się do opisanego przerwania łatwych do wykorzystania przez użytkownika. Na tym kończę ten cykl artykułów. W następnym numerze „Bajtka” pokażę przykład praktycznego wykorzystania przerwania i grafiki graczy i pocisków.

Wojciech Zientara





## WSPÓŁPRACA C-128 W SYSTEMIE CP/M Z IBM PC I DATASETTE

Użytkownicy C-128 posiadający stację dysków 1570, 1571 lub 1572 mogą wykorzystywać system CP/M V3.0 (CP/M+), uzyskując w ten sposób dostęp do wielu programów profesjonalnych. Co więcej, dzięki automatycznemu rozpoznawaniu dyskietek formatowanych na IBM PC pod systemem CP/M-86, mogą też bez kłopotów przenosić swoje zbiory pomiędzy tymi komputerami. Do pełni szczęścia brakuje jedynie możliwości przenoszenia zbiorów na IBM PC pod systemem DOS.

Możliwość taką daje nam system Concurrent DOS. Wystarczy jedynie sformatować dwie dyskietki, używając do tego celu komendy DSKMAINT, jedną po 8 sektorów na ścieżce (dla systemu CP/M), która będzie wykorzystywana do bezpośredniej komunikacji pomiędzy C-128 i IBM PC oraz drugą po 9 sektorów na ścieżce dla systemu DOS. Przesyłanie zbiorów pomiędzy obiema dyskietkami wykorzystując do tego system Concurrent DOS i rozkazy PIP lub COPY nie przedstawia już żadnych trudności, gdyż zbiory zapisane na dyskietce dziewięciosektorowej są już dostępne dla systemu DOS.

Innym mankamentem systemu CP/M jest brak możliwości zapisywania zbiorów na taśmie. Problem ten można rozwiązać poprzez wykorzystanie jako środka komunikacji pamięci RAM, wspólnej dla trybów C-128 i CP/M. W tym celu należy wykorzystać rozkazy MONITOR (tryb C-128) i SID (CP/M) w następujący sposób:

### 1. ZAPIS ZBIORU SYSTEMU CP/M

- a) wczytać rozkaz SID,
- b) wczytać zbiór do pamięci:  
R nazwa zbioru,h

gdzie h jest wartością heksadecymalną liczby bajtów SID, to jest liczby rekordów jaką zajmuje SID na dysku razy 128,

- c) wcisnąć jednocześnie klawisz CONTROL i ENTER (nie myląc go z RETURN — nie należy także używać przycisku RESET), w celu powrotu do trybu C-128,
- d) wywołać MONITOR
- e) zapisać zbiór na taśmie:  
S „nazwa”,1,1p,1k

gdzie p jest wartością h+100 (heksadecymalnie) czyli adresem początku zbioru w pamięci (jedynka przed p oznacza BANK 1), a k jest adresem końca zbioru (tzn. p+d szesnastkowo dla d będącego wartością heksadecymalną liczby bajtów danego zbioru).

- f) wyjść z monitora poprzez X i RETURN (ENTER).

### 2. WCZYTYWANIE ZBIORU

- a) wywołać MONITOR,
- b) wczytać zbiór do pamięci:  
L „nazwa”,1,1p
- c) wyjść z monitora,
- d) wczytać system CP/M za pomocą rozkazu BOOT (nie przez RESET),
- e) wykonać rozkaz SID,
- f) zapisać zbiór na dyskietce:  
W nazwa zbioru,p,k
- g) wyjść z SID.

Jeżeli SID zajmuje 63 rekordy na dysku to h = 1F80, p = 2000, a k nie może przekroczyć D2FF, co umożliwią przesłanie zbiorów do najwyższej 357 rekordów po 128 bajtów; daje to nam możliwość zapisu na taśmie zbiorów zajmujących nieco ponad 44 KB.

Krzysztof Dudziński

## AUTONUMBER 64

```
100 REM   *** AUTONUMBER ***
110 :
120 REM   AUTOR : M. SCHWERTFEGER
130 :
131 :
135 D=0:SA = 53111:EA = 53228
140 FOR I=SA TO EA :READA:POKEI,A:D=D+A:NEXT
150 IF D=12949 THEN 170
160 PRINTCHR$(147) "NIEDOBRE DANE !":END
170 SYS 53111:PRINT"PROGRAM WCZYTANY.":NEW
175 :
180 DATA 169,130,141,008,003,169,207,141,009,003,096
190 DATA 032,115,000,201,095,240,006,032,121,000,076
200 DATA 231,167,032,115,000,208,004,162,010,208,003
210 DATA 032,158,183,142,201,207,169,171,141,004,003
220 DATA 169,207,141,005,003,076,174,167,104,072,201
230 DATA 161,208,057,136,177,122,208,013,169,124,141
240 DATA 004,003,169,165,141,005,003,076,116,164,165
250 DATA 020,166,021,024,105,010,144,001,232,133,099
260 DATA 134,098,162,144,056,032,073,188,032,223,189
270 DATA 032,135,180,032,130,183,132,198,177,034,153
280 DATA 119,002,136,016,248,076,124,165
```

Wprowadzanie dłuższych programów w BASIC-u można sobie ułatwić i jednocześnie przyspieszyć stosując automatyczne numerowanie linii. Niestety Commodore 64 nie daje nam bezpośrednio takiej możliwości i dlatego też proponuję program AUTONUMBER, który pozwoli nam zrealizować tę przydatną funkcję.

Po wczytaniu oraz uruchomieniu programu wpisać:

←X i wcisnąć RETURN. X oznacza tu liczbę z przedziału 1-255 równą krokowi numeracji. Po wykonaniu tej operacji wpisujemy normalnie kolejne linie programu z tym, że ich numery będą pojawiać się na ekranie automatycznie. Jeżeli nie określmy kroku (wpisanie samej strzałki) to będzie on przyjęty jako równy 10.

Po wyzerowaniu komputera (SYS 64738 lub kombinacją STOP i RE-

STORE) program można ponownie wywołać za pomocą SYS 53111. Wciśnięcie klawisza RETURN przy wyświetlonym kolejnym numerze linii powoduje wyjście z programu.

Na podstawie czasopisma „DIE NEUE DATA WELT” 1/85

Fulgenciusz Głowiński

## KOPIOWANIE EKRANU NA DRUKARCE

Opisany poniżej program przeznaczony jest dla Commodore 64 lub 128 (w trybie C-64) oraz drukarek MPS 801, 802, 803 (nr urządzenia 4), a także dla plottera 1520 (nr urządzenia 6) i pozwala na natychmiastowy wydruk zawartości ekranu co może być bardzo pomocne podczas sporządzania programów, dokumentacji, listingów itp.

Program uruchamia się za pomocą SYS 49152 (także po wyzerowaniu komputera np. przez SYS 64738). Sterowanie odbywa się za pomocą klawiszy funkcyjnych F1, F3, F5 i F7. Dwa pierwsze umożliwiają zmianę koloru ramki i tła; F5 powoduje wydruk zawartości ekranu, a klawisz F7 zatrzymuje program (można też wcisnąć jednocześnie STOP i RESTORE). Ponieważ program wykorzystuje wewnętrzny system przerwań komputera, możliwe jest drukowanie zawartości ekranu w dowolnym momencie; także kursor porusza się szybciej niż zwykle.

Użycie klawisza F5 bez włączonej drukarki spowoduje zablokowanie komputera, a więc także programu głównego. Z tego powodu zaleca się sprawdzenie gotowości do pracy samej drukarki przed przystąpieniem do drukowania zawartości ekranu.

Na podstawie czasopisma MICRODOR

Jerzy Żukowski

```
100 REM   KOPIOWANIE EKRANU NA DRUKARCE
110 REM
120 REM   TŁUMACZYŁ DLA BAJTKA
130 REM   JERZY ŻUKOWSKI
140 REM
150 REM   -----
160 :
170 :
180 D=0:SA = 49152 : EA = 49355
190 FORI = SA TO EA :READA:POKEI,A:D=D+A:NEXT
200 IF D<>25819 THEN PRINT"O SPRAWDZ DANE !":END
210 :
220 PRINTCHR$(147)"PROGRAM WCZYTANY."
230 PRINT"MF1 = ZMIENIA KOLOR RAMKI."
240 PRINT"MF3 = ZMIENIA KOLOR TŁA."
250 PRINT"MF5 = KOPIUJE EKRAN."
260 PRINT"MF7 = KONIEC PROGRAMU.":PRINT
270 PRINT"JEZELI MASZ DRUKARKE MPS WCISNIJ 'M'"
280 PRINT"JEZELI MASZ PLOTTER 1520 WCISNIJ 'P'"
290 GETA$:IFA$=""THEN290
300 IFA$<"M"ANDA$<"P"THEN290
310 IFA$="P"THENPOKE49274,6
320 PRINT CHR$(147)
330 INPUT"DRUKARKA GOTOWA (T/N) ";D$
340 IFD$<"T"THENPRINT"URUCHOM DRUKARKE !":GOTO330
350 SYS 49152:NEW
360 :
370 DATA 120,169,013,141,020,003,169,192,141,021,003
380 DATA 088,096,032,159,255,173,119,002,201,133,240
390 DATA 015,201,134,240,027,201,135,240,058,201,136
400 DATA 240,035,076,049,234,032,183,192,172,032,208
410 DATA 200,140,032,208,032,195,192,076,049,234,032
420 DATA 183,192,172,033,208,200,140,033,208,032,195
430 DATA 192,076,049,234,032,183,192,169,049,141,020
440 DATA 003,169,234,141,021,003,032,195,192,076,049
450 DATA 234,032,183,192,169,000,032,249,253,169,018
460 DATA 162,003,160,255,032,000,254,032,192,255,162
470 DATA 018,032,198,255,169,000,032,249,253,169,019
480 DATA 162,004,160,000,032,000,254,032,192,255,162
490 DATA 019,032,080,242,162,000,160,000,024,032,010
500 DATA 229,162,000,160,000,032,087,241,032,210,255
510 DATA 200,192,040,208,245,232,224,025,208,238,032
520 DATA 204,255,169,019,032,195,255,169,018,032,195
530 DATA 255,032,195,192,076,049,234,032,228,255,173
540 DATA 014,200,041,254,141,014,220,096,173,014,220
550 DATA 009,001,141,014,220,096
```

# KLAN COMMODORE



Dostajemy wiele listów z prośbami, pretensjami i groźbami (tak, tak!) na temat zawartości KLANU COMMODORE. Główne zarzuty dotyczą braku jakichkolwiek publikacji dotyczących komputerów C16, C116 oraz Commodore +4. Postaramy się zaspokoić żądania czytelników. Jako pierwszy, publikujemy artykuł dotyczący przyspieszonego ładowania programów dla komputerów C16 i C116. Program jest odpowiednikiem TURBO S/L dla C-64.

TURBO 16 pozwala na przyspieszone wczytywanie programu zwykłą instrukcją LOAD. Program przeznaczony do przyspieszonego ładowania musi być jednak uprzednio odpowiednio zapisany. Służy

do tego poniższy program. Po wpisaniu należy go zapisać na osobnej kasecie. Następnie program uruchamiamy, i jeśli ukaże się napis „O.K.”, oznacza to, że dysponujemy poprawnym programem.

W przypadku pojawienia się komunikatu o błędzie należy sprawdzić linie zawierające dane — program informuje, czy błąd polega na złej ilości linii, czy na pomyłce w pisaniu. Gdy szczęśliwie przebrniemy przez uruchomienie, należy załadować program przeznaczony do późniejszego przyspieszonego ładowania i zapisać go rozkazem SAVE"NAZWA",7. Jeśli chcemy, by program sam się uruchamiał, piszemy SAVE"NAZWA",7,1. Raz załadowane TURBO działa poprawnie do momentu wyłączenia komputera lub załadowania innego programu, mogącego zakłócić jego działanie (np. gry napisanej w języku maszynowym).

Na podstawie Commodore Bulletin 11/1 85/86

(ms)

```

10 REM *****
20 REM * *
30 REM * TURBO 16 *
40 REM * *
50 REM *****
100 DATA1,0,53,0,55,0,52,61,54,61,56,61
110 D=15616:FORI=1TO6:READA,B:POKEA,B:NE
XT
120 READQ:[FQ<>-1]THENQ%=STR$(Q):C=C+Q:D=
D+1:GOTO120
130 IFC<>74750THENPRINT"POMYŁKA W DANYCH
!":END
140 IFD<>16251THENPRINT"ZŁA ILOSC DANYCH
!":END
150 PRINT"OK.":SYS15616
1000 DATA169,11,141,48,3,169,61,141,49,3
,96,72,165,174
1001 DATA201,7,240,4,104,76,164,241,165,
178,141,123,63,165
1002 DATA179,141,124,63,165,157,141,125,
63,165,158,141,126,63
1003 DATA165,173,141,191,62,160,15,169,3
2,153,192,62,136,16
1004 DATA250,164,171,192,17,144,2,160,16
,136,48,8,177,175
1005 DATA153,192,62,76,65,61,160,72,185,
119,62,153,255,6
1006 DATA136,208,247,32,25,227,169,72,14
1,36,3,169,3,141
1007 DATA37,3,169,1,170,168,32,186,255,1
69,188,162,192,160
1008 DATA62,32,189,255,169,0,133,154,169
,3,133,35,169,36
1009 DATA133,34,169,34,160,3,162,38,32,2
16,255,169,75,141
1010 DATA36,3,169,236,141,37,3,169,0,170
,168,133,178,169
1011 DATA7,133,179,133,158,169,176,133,1
57,32,200,61,169,128
1012 DATA133,154,162,0,160,0,173,123,63,
173,178,173,124,63
1013 DATA133,179,173,125,63,133,157,173,
126,63,133,158,32,200
1014 DATA61,76,3,135,32,18,62,165,178,32
,64,62,165,179
1015 DATA32,64,62,165,157,32,64,62,165,1
58,32,64,62,132
1016 DATA180,164,178,169,0,133,178,177,1
78,32,64,62,200,208
1017 DATA2,230,179,196,157,165,179,229,1
58,144,238,165,180,32
1018 DATA64,62,32,84,62,169,27,141,6,255
,141,62,255,169
1019 DATA8,133,1,88,32,132,255,96,120,14
1,63,255,169,0
1020 DATA133,1,169,11,141,6,255,202,208,
253,136,208,250,169
1021 DATA160,141,2,255,169,2,141,3,255,1
69,16,141,9,255
1022 DATA160,64,169,16,32,64,62
1023 DATA136,208,248,169,90,133,171,69,1
30,133,180,169,8,133
1024 DATA172,38,171,32,84,62,198,172,208
,247,96,162,108,144
1025 DATA2,162,255,32,93,62,169,16,44,9,
255,240,251,72
1026 DATA169,0,142,2,255,141,3,255,104,1
41,9,255,165,1
1027 DATA73,2,133,1,96,169,72,141,252,25
5,169,3,141,253
1028 DATA255,141,63,255,238,25,255,24,32
,79,3,141,62,255
1029 DATA165,34,141,25,255,88,32,138,255
,32,132,255,165,158
1030 DATA197,157,208,25,173,71,7,240,14
169,255,141,12,255
1031 DATA141,13,255,32,190,139,76,220,13
9,32,157,138,76,3
1032 DATA135,162,29,76,131,134,0,32,32,3
2,32,32,32,32
1033 DATA32,32,32,32,32,32,32,32,32,1
73,25,255,133
1034 DATA34,56,120,8,32,147,3,32,184,3,1
68,169,0,133
1035 DATA178,32,184,3,133,179,32,184,3,1
33,45,32,184,3
1036 DATA133,46,32,184,3,145,178,69,158,
133,158,200,208,5
1037 DATA230,179,238,25,255,196,45,165,1
79,229,46,144,231,32
1038 DATA184,3,133,157,169,8,133,1,40,17
6,1,96,76,0
1039 DATA7,169,0,133,1,202,208,253,136,2
08,250,132,158,162
1040 DATA0,32,198,3,38,172,165,172,201,1
6,208,245,32,184
1041 DATA3,201,16,240,249,201,90,208,234
,96,169,1,133,172
1042 DATA32,198,3,38,172,144,249,165,172
,96,169,16,36,1
1043 DATA208,252,36,1,240,252,45,9,255,7
2,169,1,162,120
1044 DATA142,2,255,141,3,255,169,16,141,
9,255,173,25,255
1045 DATA24,105,16,141,25,255,104,10,10,
10,10,96,-1

```

## WIĘŚCI

## NAJBARDZIEJ KOMPATYBILNY Z KOMPATYBILNYCH?



Pod takim sugestywnym hasłem francuska spółka TRANS lansuje swój najnowszy wyrób — mikrokomputer JASMIN TURBO HQ, będący odpowiednikiem IBM PC/XT. Wydawałoby się, że na rynku mikrokomputerowym obfitującym w przeróżne odmiany mniej lub bardziej udanych kopii PC nie ma już miejsca na jeszcze jednego „klona”. A jednak JASMIN TURBO HQ dysponuje kilkoma atutami, pozwalającymi na przyciągnięcie potencjalnych nabywców. Niewątpliwym atutem jest oczywiście cena, która dla wersji podstawowej sytuuje się na poziomie cen droższych mikrokomputerów domowych. Równie konkurencyjne są ceny pozostałych trzech wersji HQ, ponieważ wartość

rynkowa oprogramowania i precyzyjnej „myszy”, dostarczanych wraz z mikrokomputerem w ramach ceny kolejnych wersji, przekracza 3000 franków. Oferta spółki TRANS mieści się w granicach od około 4800 franków za HQ CLUB do 11 700 franków za HQ 20.

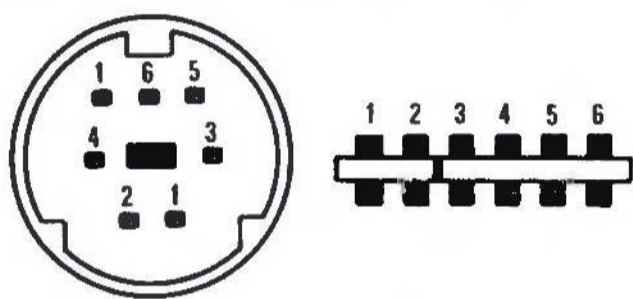
JASMIN TURBO HQ o solidnej konstrukcji, z wyglądu przypominającej bardziej PC/AT, zawiera jednostkę centralną wyposażoną w:

- płytę-matkę z mikroprocesorem 8088-2, pamięcią 640KB RAM (256KB RAM w wersji HQ CLUB) i miejscem na koprocessor arytmetyczny 8087 oraz 8 gniazd, tzw. slotów do podłączania różnych kart, umożliwiających rozbudowę sy-

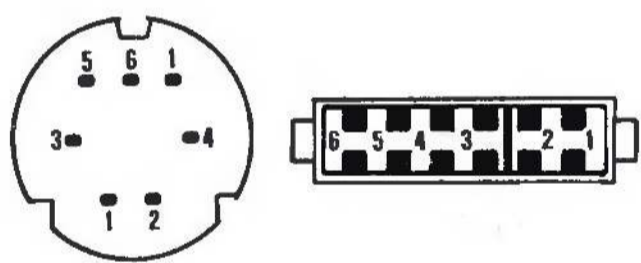
## DATASETTE I COMMODORE PLUS/4

Wprowadzenie nowego typu gniazd i wtyków dla drążków sterowych magnetofonu w komputerach C-16, C-116 oraz PLUS/4 było dość zaskakującym pomysłem firmy Commodore. Zmiana zastosowanego we wcześniejszych modelach standardu jeszcze pogorszyła popularność tej serii. Magnetofony do tych komputerów nie różnią się niczym — prócz nie-szczęśliwych wtyczek — od innych Datasette i mogą współpracować z wszystkimi modelami Commodore, od VIC-20 poczynając i na C-128 kończąc.

Największy problem sprawi nam zakupienie odpowiednich wtyków. O ile sama wtyczka do Commodore 64 może być wykonana z powszechnie dostępnego (na giełdach oczywiście) gniazda krańcowego, o tyle ze zdobyciem wtyków czy gniazd do Commodore PLUS/4 może już być nie-



Rys. 1 Porty magnetofonu w PLUS/4 i C-64 (widok z przodu).

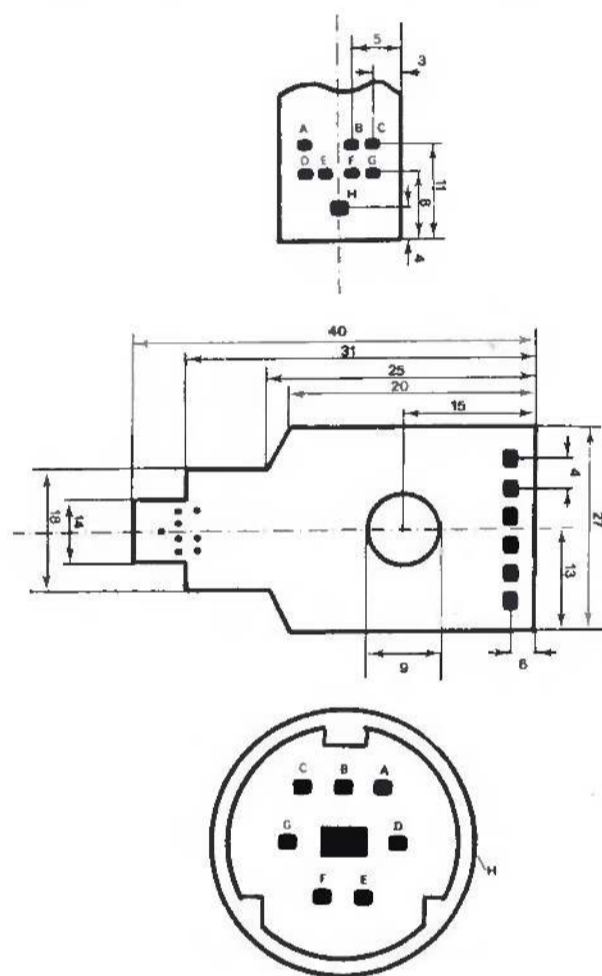


Rys. 2 Wtyki magnetofonowe dla PLUS/4 i C-64 (widok z przodu).

co trudniej. Swego czasu elementy te można było również nabyć na „Giełdzie Bajtka”.

Rysunki 1 i 2 przedstawiają schemat połączeń gniazd i wtyków magnetofonowych dla PLUS/4 (także C-16 i C-116) oraz C-64. Na rysunku 3 przedstawiono złącze pośrednie, które umożliwi korzystanie z DATASETTE od PLUS/4 z Commodore 64, 128 czy VIC-20.

Jeżeli wtyk C-64 zastąpimy odpowiednio przygotowaną płytką, a gniazdo PLUS/4 wtykiem to otrzymamy złącze pośrednie pozwalające na użyt-

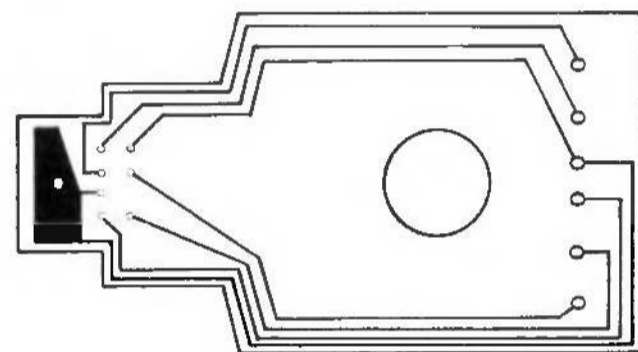


Rys. 3 Płytkę złącza pośredniego (widok od strony druku).

kowanie DATASETTE od C-64 z komputerami PLUS/4, C-16 oraz 116.

UWAGA. Ponieważ port magnetofonu w OBU komputerach jest połączony bezpośrednio z mikroprocesorem, zaleca się ostrożność podczas prób, gdyż grozi to spalaniem układu. Radziłbym wobec tego wszystkim zainteresowanym taką przeróbką, aby dokonać jej pod okiem doświadczonego fachowca.

*Klaudiusz Dybowski*



Rys. 4 Układ ścieżek (widok od strony druku).

### Oznaczenie styków:

- 1 — MASA (GND)
- 2 — +5 V
- 3 — CASSETTE MOTOR
- 4 — CASSETTE READ
- 5 — CASSETTE WRITE
- 6 — CASSETTE SENSE

## WIĘŚCI

### KOMPUTER POLIGLOTA

stemu,  
— szesnastokolorową kartę graficzną typu CGA z wyjściami: całkowitego kolorowego sygnału wizyjnego, sygnału wizyjnego czarno-białego i RGB oraz sprzężeniem dla pióra świetlnego,  
— kartę „MULTI I/O” zawierającą dwa porty szeregowo, interface równoległy drukarki typu Centronics, kontroler dwóch napędów dysków, zegar/kalendarz z zasilaniem baterijnym i gniazdo do podłączenia drążka sterowego.  
Zasilacz 150W z wentylatorem usytuowany jest w jednostce centralnej, co pozwala na alternatywną pracę z różnymi monitorami.  
Programową kompatybilność zapewnia system operacyjny DOS PLUS, akceptujący zarówno MS DOS 2-11, jak i CP/M 86 oraz system GEM firmy Digital Research Incorporation, który umożliwia pracę z wszystkimi, dostępnymi na rynku programami napisanymi pod tym systemem. Np. GEM DES-

KTOP pozwala na pracę HQ z ikonami w stylu Macintosh'a.  
Równie istotnym z punktu widzenia kompatybilności rozwiązaniem jest możliwość pracy z dwoma przełączalnymi z klawiatury częstotliwościami zegara: 8MHz i 4,77MHz. Pracujący normalnie z częstotliwością 8MHz mikrokomputer HQ (stąd w nazwie TURBO) pozwala więc — w przypadku wystąpienia problemów w pracy z programami napisanymi w systemach z zegarem 4,77MHz (np. oryginalny IBM/PC) — ominąć i ten problem. Nie dziwi więc wynik prób kompatybilności przeprowadzanych przy użyciu „Compatest” i MS DOS wg 3 poziomu (kompatybilność sprzętowa i programowa), a mianowicie 98%. A w dodatku TRAN utrzymuje, że w trakcie prób nie znaleziono jeszcze ani jednego programu napisanego na IBM/PC, który by nie działał na JASMIN TURBO HQ! Pozostaje tylko pytanie kto i jak szukał.

(wz)

**Do niedawna** komputery osobiste przeznaczone były głównie dla ludzi posługujących się alfabetem łacińskim. Ostatnio komputerowe firmy brytyjskie i amerykańskie opracowały programy dostosowane do szerokiej gamy języków od arabskiego do zulu.  
Najwięcej problemów stwarza oczywiście język chiński. Składa się on z tysięcy skomplikowanych znaków, z których każdy może być wymawiany na setki sposobów. Chińczycy stworzyli uproszczony system znaków dostosowany do maszyn do pisania i komputerów. Niektóre firmy jak IBM i nieliczne z Chin i Tajwanu próbują komputeryzować język chiński przez przydzielanie każdemu znakowi kodu. Jest to bardzo uciążliwa metoda.  
Prostsze rozwiązanie zastosowała firma Intech Systems Inc. z Mineapolis. Komputer nie ma klawiatury. Pi-

szący wybiera symbole przez dotknięcie odpowiedniego elementu na ekranie. Jeśli piszący chce napisać „an” czyli „pokój” dotyka litery „a” w dole ekranu. Na ekranie ukazują się wówczas chińskie sylaby rozpoczynające się na „a”. Po naciśnięciu sylaby „an” na ekranie wyświetlone zostaną wszystkie znaki tej sylaby różniące się wymową. Użytkownik znający biegle chiński bez trudu wybierze właściwy. Po naciśnięciu sensora odpowiadającego temu znakowi pojawi się on w górnej części ekranu. Tekst może składać się z siedmiu znaków.  
Tą metodą można tworzyć blisko 13 tys. chińskich znaków przy użyciu jednego z czterech fonetycznych alfabetów. Tekst może być pisany pionowo i poziomo.

(do)

## UŻYTECZNY HAZARD — ROZKŁAD DWUMIANOWY

Nie wątpię, że znasz grę w „orła i reszkę”. Gra ta, prawdopodobnie w swojej istocie bardziej sędziwa niż sam pieniądz, polega na rzucaniu monetą. Jeżeli wypadła reszka, a Ty na nią właśnie postawiłeś, to wygrałeś — w przeciwnym razie doznałeś porażki.

Pierwsze pytanie, jakie może się nasunąć, to: co jest naprawdę interesującego w tak prostej grze hazardowej? A jednak — ewidentna dwustanowość (sukces — porażka) elementarnego doświadczenia ma cały szereg nietrywialnych zastosowań. Grę w orła i reszkę odnaleźć można bez trudu w symulacjach metodami Monte Carlo, czy też w wielu problemach fizyki statystycznej, w tym np. fizyki statystycznej magnetyków. Krótko mówiąc: wszędzie tam, gdzie mamy do czynienia z tzw. **schematem doświadczeń Bernoulliego**.

Aby lepiej zrozumieć, na czym polega ta konstrukcja, uruchom program, którego listing przedstawiono na rys. 1. Po wprowadzeniu np. cyfry 0 komputer rozpoczyna wyświetlanie strzałek, przy czym prawdopodobieństwo elementarne tego, że pojedyncza strzałka jest skierowana do góry wynosi p. Inaczej mówiąc, aby narysować pojedynczą strzałkę komputer losuje najpierw liczbę pseudolosową RND z przedziału 0,1. Jeżeli liczba ta jest mniejsza od p, wówczas komputer rysuje strzałkę skierowaną do góry (ewentualnie żeton czarny), w przeciwnym razie strzałkę skierowaną w dół ekranu (ewentualnie żeton biały).

Rysunek 2a, b przedstawia dwa zestawy złożone z takiej samej liczby n (=15) elementarnych eksperymentów. Liczba takich zestawów w obu przypadkach wynosi s = 300, z tym że w pierwszym przypadku p = 0,5, a w drugim p = 0,25. Pytanie zasadnicze jakie nasuwa się w tym miejscu, to: **w jakim celu komputer robi to wszystko?** Odpowiedź jest następująca: komputer buduje statystykę zliczając zestawy elementarnych eksperymentów według następującej reguły. Zestaw, w którym strzałka skierowana do góry pojawiła się j razy, wchodzi do „kieszeni” o numerze j, wykonując to dla każdego j = 0, ..., n. Niech zawartość takiej kieszeni wynosi s<sub>j</sub>. Wówczas prawdopodobieństwo częstotliwościowe pojawia się j sukcesów (tzn. j strzałek skierowanych w górę)

### TABELA I

Exper. i teoret. r. dwumianowy		
j	r. exper.	r. teort.
0	0	.000030517578
1	0	.00045776367
2	0	.0032043457
3	.01	.013885498
4	.043333333	.041656494
5	.11	.091644287
6	.14	.15274048
7	.2	.19638062
8	.18	.19638062
9	.163333333	.15274048
10	.096666667	.091644287
11	.05	.041656494
12	.00333333333	.013885498
13	.00333333333	.0032043457
14	0	.00045776367
15	0	.000030517578

(Dalej dowolny klawisz)

wynosi w przybliżeniu  $p(j) = s_j/s$ . Tak więc w wyniku symulacji komputer wyznacza doświadczalnie tzw. rozkład Bernoulliego, inaczej zwany dwumianowym — rozkład, którego rolę pełnią w fizyce czy matematyce wprost trudno przecenić. Na przykład rozkład Gaussa (zwany także normalnym), czy też rozkład Poissona wyprowadza się właśnie z rozkładu Bernoulliego wprowadzając dodatkowe założenia (np. co do liczby elementarnych doświadczeń n; Czytelników bardziej zainteresowanych tymi zagadnieniami pragnę odesłać do podręcznika M. Fiszera „Rachunek prawdopodobieństwa i statystyka matematyczna”, rozdz. 5 i 6, PWN, Warszawa 1967).

Rysunek 3 przedstawia właśnie porównanie rozkładu doświadczalnego (białe kółka) i teoretycznego (czarne punkty).

Rozkład doświadczalny  $p(j)$  dla każdego j = 0, ..., n komputer buduje w pętli rozpoczynającej się w linii 540, a kończącej w linii 820. Teoretyczny rozkład Bernoulliego ma postać:

$$b(j) = p^j(1-p)^{n-j} \cdot \frac{n!}{j!(n-j)!}$$

W programie b(j) jest obliczane dla każdego j (przy zadanym p oraz n) w bloku zaczynającym się w linii 900, a kończącym w linii 1120.

Przykładowo dla schematu z p = 0,5 (rys. 3a) i p = 0,25 (rys. 3b). Analogiczne porównanie liczbowe przedstawiono w Tabeli I. Ponadto w Tabeli II porównano charakterystyczne momenty (inaczej mówiąc, charakterystyczne wartości średnie) eksperymentalne i teoretyczne.

Zadowalająca zgodność pomiędzy wynikami symulacji komputerowej a przewidywaniami teoretycznymi została osiągnięta dzięki dobrej statystyce zbudowanej przez komputer. Mogę Ci zaproponować, Szanowny Czytelniku, proste ćwiczenie polegające na dopisaniu podprogramu, który porównuje wysumulowany rozkład dwumianowy z odpowiadającym mu rozkładem normalnym lub rozkładem Poissona. To właśnie symulacje komputerowe wspomagają np. fizyków w wyrafinowanych badaniach tam, gdzie m.in. metody analityczne stają się mniej efektywne.

W zakończeniu pragnę podkreślić, że budowanie różnego rodzaju rozkładów stanowi podstawowy cel statystyki matematycznej w ogólności, a fizyki statystycznej w szczególności.

dr hab. Ryszard Kutner  
Uniwersytet Warszawski

### TABELA II

Momenty exper. i teoret.	
m1 = 7.50333333	e1 = 7.5
m2 = 59.8833333	e2 = 60
t2 = 3.5833222	s2 = 3.75
mrl = .0066666673	f1 = 0
amrl = 3.13333333	fa1 = 3.0901936
mrl2 = 14.3333333	f2 = 15
td2 = 14.333289	d2 = 15

**OZNACZENIA**

**parametry eksperymentalne**

m1 — pierwszy moment  
m2 — drugi moment  
t2 — kwadrat dyspersji  
mrl — przesunięcie efektywne  
amrl — przesunięcie absolutne  
mrl2 — kwadratowe przesunięcie efektywne  
td2 — kwadrat dyspersji przesunięcia efektywnego

**parametry teoretyczne**

e1 — pierwszy moment  
e2 — drugi moment  
s2 — kwadrat dyspersji  
f1 — przesunięcie efektywne  
fa1 — przesunięcie absolutne  
f2 — kwadratowe przesunięcie efektywne  
d2 — kwadrat dyspersji przesunięcia efektywnego

```
prawdopodob. element. = 0.5
nr exper. = 6; liczba exper. = 300
liczba elementarnych exper. = 15
```

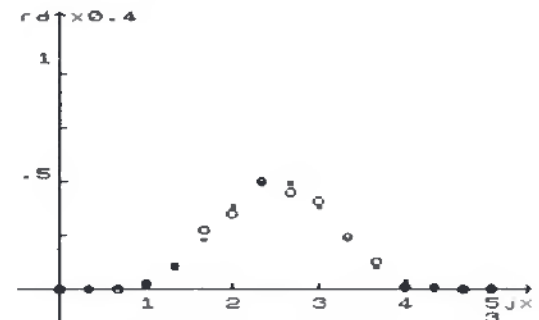
↑ ↓ ↓ ↓ ↑ ↑ ↑ ↑ ↓ ↓ ↓ ↓ ↑ ↑ ↑ ↑

Rys. 2a

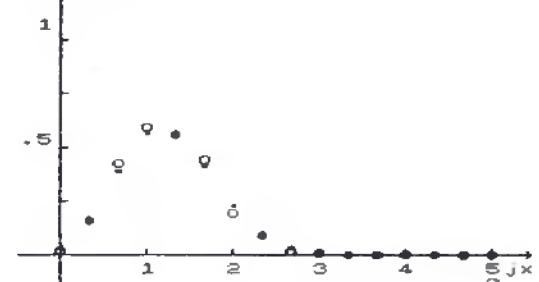
```
prawdopodob. element. = 0.25
nr exper. = 6; liczba exper. = 300
liczba elementarnych exper. = 15
```

↑ ↓ ↓ ↓ ↓ ↑ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↑ ↑

Rys. 2b



Rys. 3a



Rys. 3b

```
10 REM Symulacja schematu Bernoulliego. Rozkład dwumianowy
20 REM Ryszard Kutner
30 REM *****
40 REM Macierze i dane
50 REM
60 DIM p(10): DIM d(10): DIM b(10): DIM s(10): DIM f(10)
70 RESTORE 90
80 READ p,n,s,b,sr
90 DATA .5,15,300,1,17
100 REM
110 INPUT op
120 IF op>0 THEN GO TO 310
130 REM
140 POKE USR "↑", BIN 00010000
150 POKE USR "↑"+1, BIN 00111000
160 POKE USR "↑"+2, BIN 01010100
170 POKE USR "↑"+3, BIN 10010010
180 POKE USR "↑"+4, BIN 00010000
190 POKE USR "↑"+5, BIN 00010000
200 POKE USR "↑"+6, BIN 00110000
210 POKE USR "↑"+7, BIN 00010000
220 REM
230 POKE USR "↓", BIN 00010000
240 POKE USR "↓"+1, BIN 00010000
250 POKE USR "↓"+2, BIN 00010000
260 POKE USR "↓"+3, BIN 00010000
270 POKE USR "↓"+4, BIN 10010010
280 POKE USR "↓"+5, BIN 01010100
290 POKE USR "↓"+6, BIN 00111000
300 POKE USR "↓"+7, BIN 00010000
310 REM
320 POKE USR "0", BIN 00111100
330 POKE USR "0"+1, BIN 01111110
340 POKE USR "0"+2, BIN 11111111
350 POKE USR "0"+3, BIN 11111111
360 POKE USR "0"+4, BIN 11111111
370 POKE USR "0"+5, BIN 11111111
380 POKE USR "0"+6, BIN 01111110
390 POKE USR "0"+7, BIN 00111100
400 REM
410 POKE USR "0", BIN 00111100
420 POKE USR "0"+1, BIN 01000010
430 POKE USR "0"+2, BIN 10000001
440 POKE USR "0"+3, BIN 10000001
450 POKE USR "0"+4, BIN 10000001
460 POKE USR "0"+5, BIN 10000001
470 POKE USR "0"+6, BIN 01000010
480 POKE USR "0"+7, BIN 00111100
490 REM
500 REM *****
510 REM Buduje ansambl statystyczny
520 REM
530 RANDOMIZE sr
540 FOR i=1 TO s
550 PRINT "prawdopodob. element.=",p
560 PRINT "nr exper.=",i;"; liczba exper.=",s; PRINT "liczba elementarnych exper.=",n
570 LET u=0
580 FOR j=0 TO n-1
590 LET r=RND
600 IF r>p THEN GO TO 680
610 LET u=u+1
620 IF op>0 THEN GO TO 650
630 PRINT AT 10+2*INT(j/15),2* j-2*15*INT(j/15),"↑"
640 GO TO 660
650 PRINT AT 10+2*INT(j/15),2* j-2*15*INT(j/15),"↓"
660 PAUSE 50
```

## ZMIANA ATRYBUTÓW

**Wielu użytkowników Spectrum staje przed problemem, jak zmienić dane atrybuty (kolory) na inne bez konieczności czyszczenia ekranu i ponownego jego zapelnienia. Poniżej przedstawiam jeden z najkrótszych sposobów.**

Atrybut pewnego pola na ekranie jest to liczba określająca jego kolor. Oblicza ją się ze wzoru

$a = f * 128 + b * 64 + p * 8 + i$ , gdzie:

- a — atrybut pola
- f — FLASH (0 lub 1)
- b — BRIGHT (0 lub 1)
- p — PAPER (0 — 7)
- i — INK (0 — 7)

Można też sprawdzić atrybuty pola o współrzędnych x, y instrukcją PRINT ATTR (x,y). Dla przykładu pole o wyłączonym miganiu, włączonej jasności, kolorze tła 6 i kolorze znaku 2 będzie miało atrybut

$a = 0 * 128 + 1 * 64 + 6 * 8 + 2 = 114$

lub inaczej

```
PRINT AT 0,0;FLASH 0; BRIGHT 1; PAPER 6; INK 2;,"a"
PRINT ATTR (0,0)
```

Teraz zmiana atrybutu z a na b jest bardzo łatwa:

```
10 INPUT „Stary atr”; a, „Nowy atr.”;b
```

```
20 FOR f=22528 TO 23295
30 IF PEEK f=a THEN POKE f,b
40 NEXT f
```

Liczba 22528 oznacza tutaj pierwszy adres pamięci obrazu dla atrybutów, a 23295 koniec tego obszaru pamięci. Wyrażenie PEEK f to atrybut pola o adresie f. W rezultacie wszystkie atrybuty a zostaną zamienione na atrybuty b. Spróbujmy ten sam program napisać w assemblerze:

```
LD D,st.at,
LD E,nw.at,
LD BC,768
LD HL,22528
START1 LD A,(HL)
CP D
JR NZ,START2
LD (HL),E
START2 DEC BC
INC HL
LD A,B
OR C
JR NZ,START1
RET
```

st.at oznacza stary (zmieniany) atrybut, a nw.at nowy, wstawiany zamiast starego.

Przeanalizujemy program. W rejestrze D mamy atrybut do zmiany, a w E ten, którym należy zastąpić stary. W parę rejestrów BC wpisujemy długość pamięci dla atrybutów, a w parę HL adres początku tej pamięci. Teraz w akumulator wpisujemy liczbę cha-

```
670 GO TO 730
680 IF op>0 THEN GO TO 710
690 PRINT AT 10+2*INT (j/15),2*
j-2*15*INT (j/15);"v"
700 GO TO 720
710 PRINT AT 10+2*INT (j/15),2*
j-2*15*INT (j/15);"o"
720 PAUSE 5
730 NEXT j
740 LET p(u+1)=p(u+1)+1/s
750 IF i=1 OR i=s THEN GO TO 79
0
760 PAUSE b
770 CLS
780 GO TO 820
790 PRINT AT 20,0;"(Dalej dowol
ny klawisz)"
800 PAUSE 0
810 CLS
820 NEXT i
830 REM *****
840 REM
850 REM Konstrukcja teoretycz-
nego rozkładu dwumiano-
wego
860 REM
870 REM Konstrukcja logarytmu
naturalnego z symbolu
Newtona
880 REM
890 PRINT " (Proszę czek
ać)"
900 LET j=0
910 FOR k=1 TO n
920 LET j=j+LN (k)
930 NEXT k
940 LET v=j
950 REM
960 FOR j=1 TO n-1
970 LET u=0
980 FOR i=1 TO j
990 LET u=u+LN (i)
1000 NEXT i
1010 LET w=0
1020 FOR h=1 TO n-j
1030 LET w=w+LN (h)
1040 NEXT h
1050 LET s(j)=v-u-w
1060 LET z=s(j)
1070 LET d(j)=EXP (z)
1080 LET b(j)=d(j)*(p+j)*((1-p)^(
n-j))
1090 LET f(j+1)=b(j)
1100 NEXT j
1110 LET f(1)=(1-p)*n
1120 LET f(n+1)=p*n
1130 CLS
1140 REM *****
1150 REM
1160 REM Tabela - zestawienie
eksperymentalnego i teo-
retycznego rozkładu
dwumianowego
1170 REM
1180 PRINT "Exper. i teoret. r.d
wumianowy"
1190 PRINT
1200 PRINT "j";TAB 4;"r.exper.";
TAB 16;"r.teort."
1210 PRINT
1220 LET m1=0
1230 LET m2=0
1240 LET mrl=0
1250 LET amrl=0
1260 LET mrl2=0
1270 FOR j=1 TO n+1
1280 PRINT j-1;TAB 4;p(j);TAB 16
;f(j)
1290 LET m1=m1+(j-1)*p(j)
1300 LET m2=m2+(j-1)*(j-1)*p(j)
1310 LET mrl=mrl+(2*(j-1)-n)*p(j)
1320 LET amrl=amrl+ABS (2*(j-1)-
n)*p(j)
1330 LET mrl2=mrl2+(2*(j-1)-n)*(
2*(j-1)-n)*p(j)
1340 NEXT j
1350 LET e1=n*p
1360 LET e2=p*n*(1-p+p*n)
1370 LET s2=p*(1-p)*n
1380 LET f1=n*(2*p-1)
1390 LET f2=n+n*(n-1)*(2*p-1)*(
2*p-1)
1400 IF p=.5 THEN LET fa1=5QR (2
*n/PI)
1410 IF p<.5 OR p>.5 THEN LET fa
1=n*ABS (2*p-1)
1420 LET t2=m2-m1*m1
1430 LET d2=4*n*p*(1-p)
1440 LET td2=mrl2-amrl*mrl
1450 PRINT
1460 PRINT "(Dalej dowolny klawi
sz)"
1470 PAUSE 0
1480 CLS
1490 REM
1500 PRINT "OZNACZENIA"
1510 PRINT
1520 PRINT "parametry eksperymen
talne"
1530 PRINT
1540 PRINT "e1 - pierwszy moment"
1550 PRINT "m2 - drugi moment"
```

```
1560 PRINT "t2 - kwadrat dyspers
ji"
1570 PRINT "mrl - przesunięcie e
fektywne"
1580 PRINT "amrl - przesunięcie
absolutne"
1590 PRINT "mrl2 - kwadratowe pr
zesunięcie efektywne"
1600 PRINT "td2 - kwadrat dysper
sjji przesunięcia e
fektywnego"
1610 PRINT
1620 PRINT "parametry teoretyczn
e"
1630 PRINT
1640 PRINT "e1 - pierwszy moment"
1650 PRINT "e2 - drugi moment"
1660 PRINT "s2 - kwadrat dyspers
ji"
1670 PRINT "f1 - przesunięcie ef
ektywne"
1680 PRINT "fa1 - przesunięcie a
bsolutne"
1690 PRINT "f2 - kwadratowe prze
sunięcie efektywne"
1700 PRINT "d2 - kwadrat dyspers
ji przesunięcia ef
ektywnego"
1710 PRINT
1720 REM
1730 PRINT
1740 PRINT "Momenty eper. i teor
et."
1750 PRINT
1760 PRINT "m1=";m1;TAB 17;"e1="
;e1
1770 PRINT "m2=";m2;TAB 17;"e2="
;e2
1780 PRINT "t2=";t2;TAB 17;"s2="
;s2
1790 PRINT : PRINT
1800 PRINT "mrl=";mrl;TAB 17;"f1
=";f1
1810 PRINT "amrl=";amrl;TAB 17;"
fa1=";fa1
1820 PRINT : PRINT
1830 PRINT "mrl2=";mrl2;TAB 17;"
f2=";f2
1840 PRINT "td2=";td2;TAB 17;"d2
=";d2
1850 PRINT : PRINT : PRINT : PRI
NT : PRINT : PRINT
1860 PRINT "(Dalej dowolny klawi
sz)"
1870 PAUSE 0
1880 CLS
1890 REM *****
1900 REM Wyskalowane osie x-y
1910 REM
1920 LET sy=.4: PRINT "rdtx";sy
1930 PRINT : PRINT : PRINT "1"
: PRINT : PRINT : PRINT : PRINT :
PRINT : PRINT : PRINT : PRINT "
.5": PRINT : PRINT : PRINT : PRI
NT : PRINT : PRINT : PRINT
1940 PRINT TAB 29;"")
1950 LET sx=n/5: PRINT TAB 7;"1"
;TAB 12;"2";TAB 17;"3";TAB 22;"4
";TAB 27;"5";TAB 28;"jx": PRINT
TAB 27;sx
1960 PLOT 0,19: DRAW 235,0
1970 PLOT 19,0: DRAW 0,170
1980 FOR i=6 TO 23 STEP 4
1990 PLOT 10+i-1,19: DRAW 0,3
2000 NEXT i
2010 FOR j=5 TO 16 STEP 3
2020 PLOT 19,10+j-1: DRAW 3,0
2030 NEXT j
2040 REM
2050 RESTORE 2070
2060 READ k1,k3,k4,k5,k7,k8,k9
2070 DATA 1,1,1,0,19,300,19
2080 LET k2=5*sx: LET k6=40/sx
2090 LET g0=k7
2100 LET p0=p(1)*k8+k9
2110 LET f0=f(1)*k8+k9
2120 REM
2130 FOR k=k1 TO k2 STEP k3
2140 LET g=k*k4+k5
2150 LET g1=g*k6+k7
2160 LET pn=p(g+1): LET fn=f(g+1)
2170 LET rn=pn*k8+k9: LET dn=fn*
k8+k9
2180 CIRCLE g1,rn,2
2190 CIRCLE g1,dn,1
2200 NEXT k
2210 CIRCLE g0,p0,2
2220 CIRCLE g0,f0,1
2230 REM
2240 PAUSE 0
2250 CLS
2260 REM *****
2270 REM
2280 LET t=INT ((65536*PEEK 2367
4+256*PEEK 23673+PEEK 23672)/50)
2290 PRINT "Czas od chwili start
u=";t;" [s]"
2300 PRINT
2310 PRINT "Reszta RAM-u=";65535
-USR 7962;" [b]"
2320 STOP
```

Rys. 1. Jest to program napisany w BASIC-u ZX SPECTRUM, przygotowany do współpracy z kompilatorem o nazwie ERE, przyspieszającym jego wykonanie.

Jego obsługa jest prosta. Uruchamia się go klawiszem RUN, a w wersji skompiłowanej instrukcją GO TO 10. Następ-

nie należy wprowadzić cyfrę 1, o ile zyczysz sobie, aby program wyświetlał żetonny (np. czarno-białe jeżeli nie włączyłeś kolorów), albo cyfrę 0 jeżeli chcesz mieć do czynienia ze strzałkami symbolizującymi np. momenty magnetyczne. Dalej wystarczy stosować się do poleceń wyświetlanych na ekranie.

Marcin Przasnyski

# BASIC

## CPC 464

CPC 464 różni się od swoich następców (CPC 664 i CPC 6128) nie tylko rodzajem wbudowanej pamięci masowej ale także oprogramowaniem stałym — firmware. Najlepiej widoczne jest to dla użytkowników BASIC-a, który wzbogacony został o kilka nowych rozkazów. Chociaż bogatszy dialekt języka ułatwia programowanie, to jednak utrudnia przenoszenie programów między poszczególnymi modelami. Jeśli program napisany dla CPC 664 lub CPC 6128 używa nowych komend, to nie może być bezpośrednio uruchomiony (a często nawet wylistowany) na CPC 464. Część nowych rozkazów można stosunkowo łatwo przetłumaczyć. Oto kilka propozycji zamian wraz z opisem zastępowanych komend (użytkownicy CPC 464 nie mają ich przecież w instrukcji komputera):

### FRAME

Rozkaz ten powoduje, że komputer zatrzymuje wykonywanie programu do chwili, gdy sterownik monitora wyśle sygnał synchronizacji pionowej. W tym momencie program ma trochę czasu, by wypisać coś na ekranie niezauważalnie dla człowieka. Poprawia się w ten sposób wrażenie ruchu małych obiektów.

Można zastąpić przez **CALL &bd19**

### CURSOR p1,p2

Instrukcja sterująca kursorem. Kursor ma dwa wyłączniki: systemowy i użytkownika. Kursor pojawia się na ekranie tylko wtedy, gdy oba są włączone. BASIC steruje kursorem używając wyłącznika systemowego, tak więc wyłączając przełącznik użytkownika możemy wyłączyć kursor nawet dla instrukcji INPUT. Używając wyłącznika systemowego możemy wyświetlić kursor w chwili, gdy normalnie go nie widać, np. w czasie oczekiwania na wciśnięcie klawisza z zastosowaniem funkcji INKEY\$. (Uwaga: BASIC włącza przełącznik użytkownika w momencie wypisywania „Ready“.)

Wolno opuścić jeden z parametrów np.: **CURSOR, 1** lub **CURSOR 0**.

Znaczenie parametrów:

p1=0 — wyłącznik systemowy wyłączony  
p1=1 — wyłącznik systemowy włączony: analogicznie  
p2=0 lub p2=1 dla wyłącznika użytkownika  
p1=0 zastępujemy przez **CALL &bb84**  
p1=1 zastępujemy przez **CALL &bb81**  
p2=0 zastępujemy przez **CALL &bb7e**  
p2=1 zastępujemy przez **CALL &bb7b**  
np. **CURSOR 1,1 = CALL &bb81:CALL &bb7b**

### GRAPHICS PEN p1,p2

Instrukcja ustala pióro, którym rysowane będą linie i punkty grafiki, oraz wypisywane znaki (przy włączonej funkcji TAG). p1=numer pióra (0-15). Parametr p2 mówi w jaki sposób znaki tekstu mają być wypisywane w czasie działania TAG: jeśli p2=0, to tło znaku będzie w kolorze aktualnego pióra ustalonego przez GRAPHICS PAPER lub CLG natomiast jeśli p2=1, to tło będzie „przezroczyste” (na ekranie umieszczone zostaną tylko piksele właściwego znaku a nie całej matrycy 8 na 8, piksele tła pozostaną nie zmienione).

Ustalenie numeru pióra można zrobić na dwa sposoby:

- dodać p1 jako trzeci parametr najbliższej instrukcji DRAW, DRAWR, PLOT lub PLOTR.
- napisać instrukcję **CALL &bbde,1,1,1,....,1** przy czym ma być p1 jedynek (mało eleganckie ale skuteczne).

Ustawienie rodzaju tła nie może być przetłumaczone tak, by zachować pełną zgodność. Podobny efekt (przy tle rysowanym piórem 0) daje jednak włączenie trybu OR („lub”) określającego sposób wzajemnego oddziaływania kolorów nowego i starego piksela przy umieszczaniu pikseli na ekranie. Tryb OR włączamy rozkazem **PRINT CHR\$(23);CHR\$(3);**

### GRAPHICS PAPER p1

Ustala numer pióra, którym rysowane będzie tło znaków wypisywanych przy włączonej funkcji TAG (ma to również znaczenie dla działania rozkazu MASK). p1=numer pióra (0-15).

Można przetłumaczyć na **CALL &bbe4,1,1,....,1** przy czym ma być p1 jedynek.

Rozkaz **CLG p1** również ustawi pióro, ale jednocześnie wyczyści ekran.

### MASK, FILL

Nie będziemy omawiać tych instrukcji ponieważ do ich przetłumaczenia potrzebne są dość długie procedury w języku maszynowym.

MASK można próbować (ze szkodą dla grafiki) usunąć z programu. Usunięcie FILL prawdopodobnie zbyt pogorszy grafikę, by program warto było w ten sposób przerabiać.

### ON BREAK CONT

Likwiduje możliwość przerywania programu klawiszem ESC.

Można zastąpić przez **ON BREAK GOSUB** z numerem linii, w której znajduje się tylko **RETURN** lub przez komendę **KEY DEF 66,0,0,0,0**. W drugim przypadku należy pamiętać o dodaniu rozkazu **KEY DEF 66,0,&fc,&fc,&fc** przed każdym **ON BREAK GOSUB** i **ON BREAK STOP** w programie (w celu przywrócenia działania klawisza ESC).

### CLEAR INPUT

Czyści bufor klawiatury. Dzięki niej mamy pewność, że w czasie najbliższego działania INKEY\$ lub INPUT nie pojawią się znaki z klawiszy naciśniętych przypadkowo w dotychczasowej pracy programu.

Zastępujemy przez:

**WHILE INKEY\$="" :WEND**

### DERR

Funkcja zwracająca numer błędu powstałego w czasie operacji dyskowych.

Instrukcje zawierające tę funkcję należy usunąć z programu lub zastąpić **DERR** np. przez 0.

### COPYCHR\$(#nr)

Bardzo użyteczna funkcja, której wartością jest znak znajdujący się na pozycji kursora w okienku numer nr. Jeśli znak na tej pozycji jest nierozpoznawalny (np. spacja z jednym pikselem ustawionym przez PLOT), to funkcja zwróci łańcuch znakowy o długości 0.

Przy zastąpieniu tej funkcji będziemy musieli odwołać się do języka maszynowego: zamiast

**10 zm\$=COPYCHR\$(#nr)**

napiszemy

**1 cd\$=CHR\$(&d9)+CHR\$(&d5)+CHR\$(&d9)+CHR\$(&d1)+CHR\$(&dd)+CHR\$(&7e)+CHR\$(&00)+CHR\$(&cd)+CHR\$(&b4)+CHR\$(&bb)+CHR\$(&f5)+CHR\$(&cd)+CHR\$(&60)+CHR\$(&bb)+CHR\$(&21)+CHR\$(&18)+CHR\$(&00)+CHR\$(&19)+CHR\$(&77)+CHR\$(&f1)+CHR\$(&cd)+CHR\$(&b4)+CHR\$(&bb)+CHR\$(&c9)+CHR\$(&00)**

**10 CALL PEEK (@cd\$+1)+256\*PEEK (@cd\$+2), nr:zm\$=MID\$(cd\$,25,1)**

Określenie zmiennej cd\$ wystarczy wykonać raz na początku programu. Oczywiście jeśli wywołanie funkcji nastąpi w innym kontekście, to należy zmodyfikować ostatnią instrukcję z linii 10 np.:

**10 PRINT COPYCHR\$(#1);**

jest równoważne linii 1 jak wyżej, wywołaniu CALL j.w. (z parametrem nr=1) i zamiast ostatniej instrukcji:

**PRINT MID\$(cd\$,25,1);**

Uwaga: parametr nr jest konieczny!

### DEC\$(p1,p2)

Działanie tej funkcji jest bardzo podobne do działania PRINT USING. Wartość wyrażenia p1 zostaje zamieniona na łańcuch znakowy według formatu zawartego w łańcuchu p2 (opis możliwych formatów znajduje się w instrukcji komputera przy opisie PRINT USING). Wartość p1 musi być liczbą.

Przykład:

**DEC\$(&10,"\*\*#####") = "\*\*\*\*\*16"**

Przetłumaczenie tej funkcji jest bardzo proste, gdyż BASIC CPC 464 właściwie ją posiada; przyczyną, dla której jej opis nie został umieszczony w instrukcji komputera, jest mały błąd w procedurze ROMu. CPC 464 wymaga po prostu dwóch nawiasów otwierających pole parametrów, tak więc podany wyżej przykład będzie wyglądał następująco:

**DEC\$(%10,"\*\*#####")**

Oprócz kilku nowych rozkazów BASIC późniejszych modeli Amstrada dopuszcza również dodatkowe parametry niektórych instrukcji:

**PEN#p1,p2,p3**

Poza numerem pióra p2 (dla okna #p1) dopuszczalny jest jeszcze trzeci parametr p3, określający, czy tło wypisywanych znaków jest „przezroczyste” (patrz GRAPHICS PEN) mający znaczenie dla znaków wypisywanych w trybie tekstowym (wyłączona funkcja TAG).

Trzeci parametr zastępujemy przez

**PRINT #p1,CHR\$(22);CHR\$(p3);**

gdzie p3= 0 oznacza tło „nieprzezroczyste”, p3=1 oznacza tło „przezroczyste”.

Uwaga: #p1 może być opuszczone dla okna 0.

**MOVE x,y,p1,p2 MOVER x,y,p1,p2**

Oprócz współrzędnych kursora graficznego można podać numer pióra, którym rysowane będą linie (p1) oraz tryb w jakim kolor piksela znajdującego się na ekranie będzie oddziaływał na kolor piksela wpisywanego na ekran w to samo miejsce. Znaczenie parametru p2 jest następujące:

p2=0 — tryb standardowy: kolor dotychczasowego piksela jest ignorowany

p2=3 — tryb OR: jeśli przedstawimy numery piór piksela dotychczasowego i nowego jako liczby dwójkowe i dodamy te liczby logicznie bit po bicie, to otrzymana liczba dwójkowa będzie numerem pióra nowego piksela.

Przykład:

pióro 5: 0101  
pióro 9: OR 1001  
-----  
pióro 13: 1101

Kolor nowego piksela będzie więc zależał od ostatniej instrukcji INK 13,...

p2=1 tryb XOR: podobnie, lecz nie dodajemy a wykonujemy różnicę symetryczną („albo”)

Przykład:

pióro 5: 0101  
pióro 9: XOR 1001  
-----  
pióro 12: 1100

p2=2 tryb AND: podobnie z mnożeniem logicznym:

Przykład:

pióro 5: 0101  
pióro 9: AND 1001  
-----  
pióro 1: 0001

Parametr p2 tłumaczymy na:

**PRINT CHR\$(23);CHR\$(p2);**

**DRAW,DRAWR,PLOT,PLOTR**

Te instrukcje mają w CPC 664 i CPC 6128 takie same parametry jak MOVE i MOVER. CPC 464 dopuszcza tylko x,y i p1. p2 tłumaczymy więc jak wyżej

Sergiusz Wolicki

# Co piszczy pod klawiaturą?

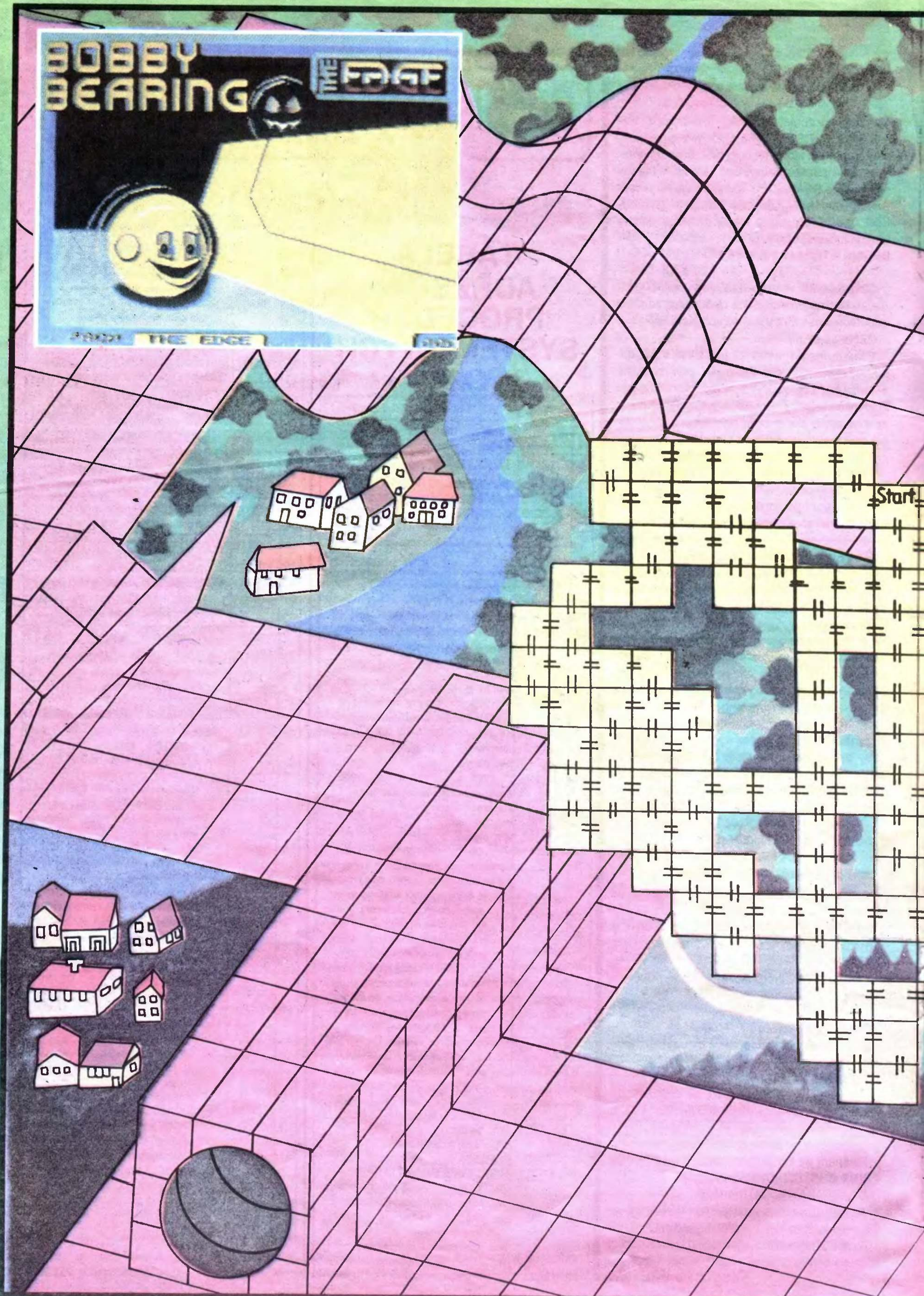
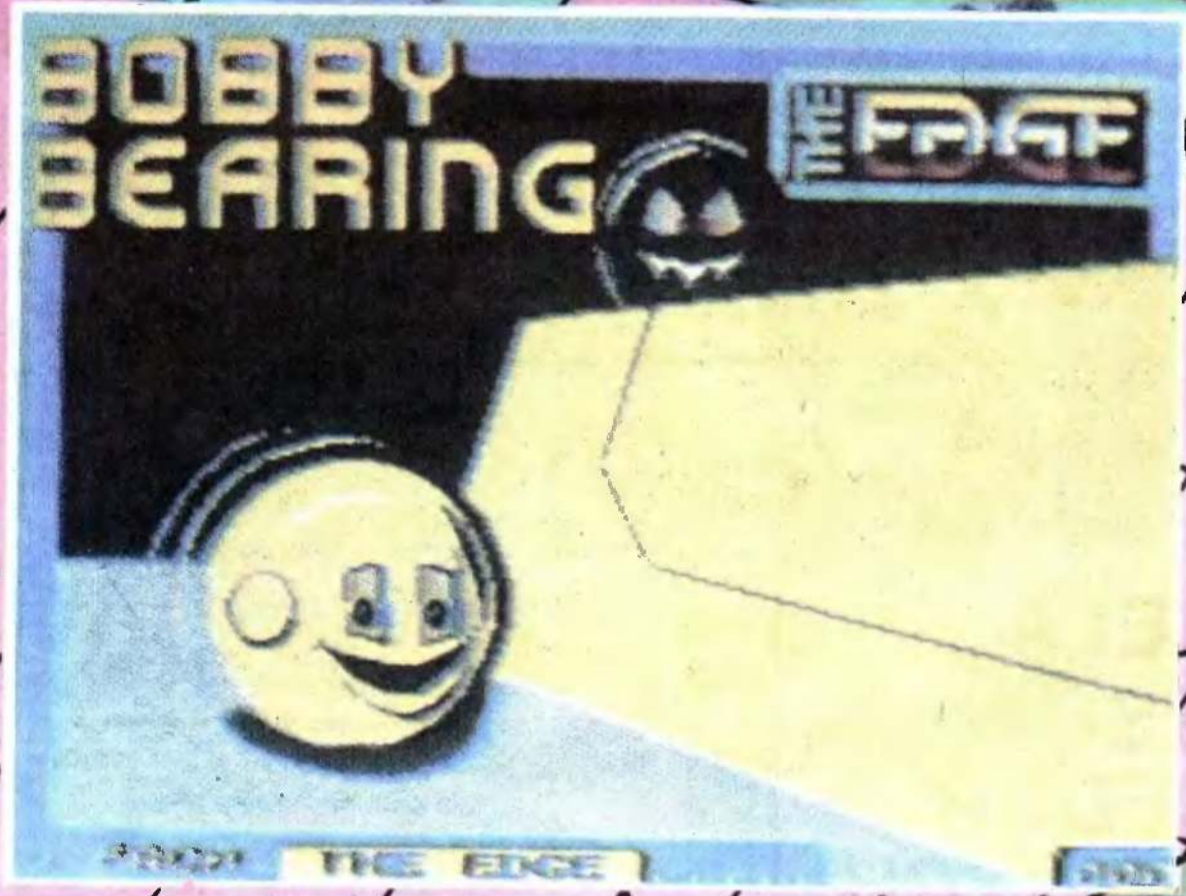
(cz. V)

## TABELA ADRESÓW PROCEDUR SYSTEMOWYCH C.D.

Nr	Adres wektora	Adres rzeczywisty/opis		
		464	664	6128
40	BB78	1180	1178	117C
		Odczytuje pozycję kursora. Wej: nie ma Wyj: H zawiera numer kolumny kursora, L zawiera numer linii kursora a A zawiera licznik przesuwu linii (scroll).		
41	BB7B	1289	1282	1286
		Zezwala na wyświetlanie kursora w trybie tekstowym. Wej: nie ma Wyj: AF są modyfikowane.		
42	BB7E	129A	1293	1297
		Zakazuje wyświetlania kursora w trybie tekstowym. Wej: nie ma Wyj: AF są modyfikowane.		
43	BB81	1279	1272	1276
		Zezwala systemowi na wyświetlanie kursora. Wej: nie ma Wyj: nie ma		
44	BB84	1281	127A	127E
		Zakazuje systemowi wyświetlanie kursora. Wej: nie ma Wyj: nie ma.		
45	BB87	11CE	11C6	11CA
		Sprawdza, czy pozycja kursora znajduje się wewnątrz okna i ewentualnie koryguje. Wej: H zawiera numer kolumny tekstowej pozycji. L zawiera numer linii tekstowej pozycji. Wyj: H zawiera numer kolumny, w której znak ma być drukowany. L zawiera numer linii, w której ma być drukowany. A i F są modyfikowane. Jeśli nie jest wymagane przesuwanie okna (scrolling), wskaźnik CARRY jest "1" i B jest modyfikowany. Jeśli scrolling ma nastąpić, CARRY jest "0" a B zawiera FF. Jeśli ma wystąpić przesuwanie do dołu (inverse), CARRY jest "0" lecz B zawiera 00.		
46	BB8A	1268	1261	1265
		Ustawia kursor na ekranie. Wej: nie ma Wyj: AF są modyfikowane.		
47	BB8D	1268	1261	1265
		Usuwa kursor z ekranu. Wej: nie ma Wyj: AF są modyfikowane.		
48	BB90	12A9	12A2	12A6
		Określa kolor znaku. Wej: A zawiera numer atramentu. Wyj: AF i HL są modyfikowane.		
49	BB93	12BD	12B6	12BA

50	BB96	12AE	12A7	12AB	Odczytuje kolor znaku. Wej: nie ma Wyj: A zawiera numer atramentu. F jest modyfikowany.
		Określa kolor papieru dla tekstu. Wej: A zawiera numer atramentu. Wyj: AF i HL są modyfikowane.			
51	BB99	12C3	12BC	12C0	Odczytuje kolor papieru dla tekstu. Wej: nie ma Wyj: A zawiera numer koloru papieru. A i F są modyfikowane.
52	BB9C	12C9	12C2	12C6	Zamienia kolor znaku z kolorem papieru. Wej: nie ma Wyj: AF i HL są modyfikowane.
53	BB9F	137A	1377	137B	Umożliwia lub zabrania wyświetlanie koloru papieru (tła). Wej: Jeśli tło jest wyświetlane, A jest 0; jeśli nie (tryb przezroczysty) — A jest różne od 0. Wyj: AF i HL są modyfikowane.
54	BBA2	1387	1384	1388	Sprawdza, czy tło (papier) ma być wyświetlane, czy nie. Wej: nie ma Wyj: A = 0 jeśli tło ma być wyświetlane, jeśli nie — A jest różne od 0. DE, HL i F są modyfikowane.
55	BBA5	12D3	12D0	12D4	Odczytuje adres matrycy znakowej. Wej: A zawiera znak poszukiwany w matrycy. Wyj: A i F są modyfikowane. Jeśli matryca jest matrycą zdefiniowaną przez użytkownika, wskaźnik CARRY jest "1". Jeśli matryca znajduje się w pamięci ROM, CARRY jest "0". HL zawiera adres matrycy.
56	BBA8	12F1	12EE	12F2	Ustawia matrycę dla znaku zdefiniowanego przez użytkownika. Wej: A zawiera znak, którego matryca ma być ustawiona a HL zawierają adres matrycy. Wyj: Jeżeli znak jest zdefiniowany przez użytkownika, wskaźnik CARRY jest "1"; w przeciwnym przypadku CARRY jest "0". AF, BC, DE i HL są modyfikowane.
57	BBAB	12FD	12FA	12FE	Ustawia adres tablicy dla matrycy zdefiniowanej przez użytkownika. Wej: DE zawiera pierwszy znak tablicy a HL zawiera adres początku nowej tablicy. (D = 0 jeśli tablica ma być skasowana). Wyj: Jeśli tablica jeszcze nie istnieje, CARRY jest "0" a A i HL są modyfikowane. Jeśli tablica zdefiniowana przez użytkownika istnieje, CARRY jest "1", A zawiera pierwszy znak poprzedniej tablicy, HL zawiera adres poprzedniej tablicy a BC i DE są modyfikowane.

Wojciech Ziółek





# BOBBY BEARING

Jest to jedna z nowszych gier typu SPINDIZZY czy GYROSCOPE. Graficznie zrobiona świetnie — trójwymiarowa staranna grafika. Gra bardzo szybka i wymagająca precyzji.

Główny bohater, BOBBY poszukuje w labiryncie podobnych sobie — białych, uśmiechniętych kul z zamkniętymi oczkami. Zadaniem BOBBY'ego, po odnalezieniu „braci”

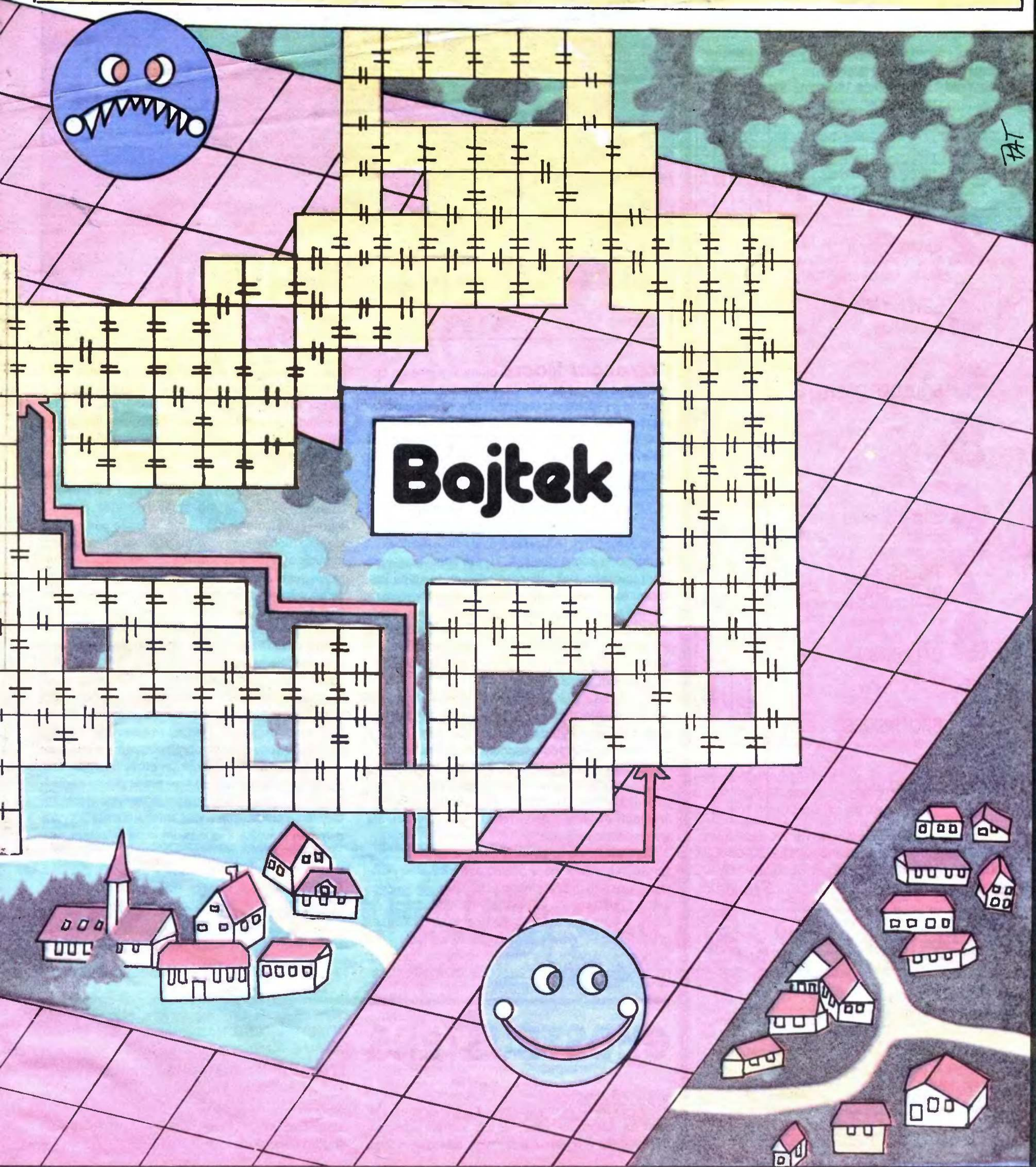
przetransportowanie ich do miejsca oznaczonego na mapie jako „START” czyli do tego miejsca, w którym znajduje się BOBBY na początku gry. „Braci” należy popychać przed sobą. Ich imiona można odczytać na ekranie — pierwszym jest Barnaby.

W wykonaniu zadania przeskadzają „czarne kule”. Odbierają energię, spychają z tra-

sy. Na dowieszenie „braci” masz określony czas. Gdy wpadniesz w pułapkę „bez wyjścia” naciśnij klawisz „Q” — odbierze trochę „czasu” ale będziesz mógł poruszać się dalej.

Po przetransportowaniu wszystkich „braci” spotka cię nagroda. Jaka?... Zagraj, a sam zobaczysz!

M1



# 10

BAJTKOWA LISTA PRZEBOJÓW 5/87

Piąte notowanie Bajtkowej Listy Przebojów przedstawia się bez większych niespodzianek. Głosowało 7.521 osób na 162 tytuły gier.

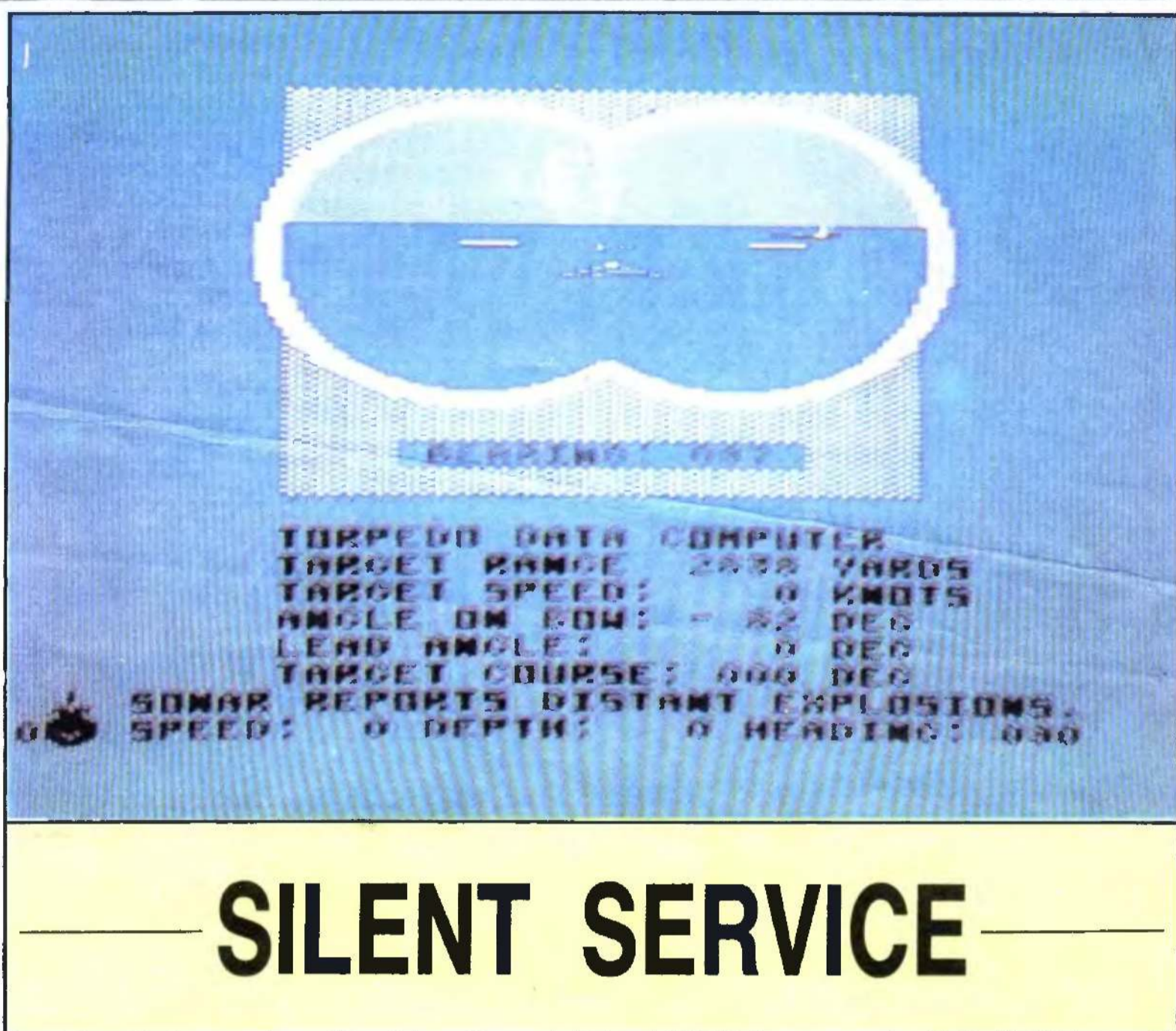
	ATARI	AMSTRAD	COMMODORE	SPECTRUM
1 BEACH HEAD II	x	x	x	x
2 SPY Vs SPY II	x	x	x	
3 SEVEN CITIES OF GOLD	x	x	x	
4 BOULDER DASH	x	x	x	x
5 W.A.R.			x	x
6 SILENT SERVICE	x	x	x	x
7 TIGERS IN THE SNOW		x	x	
8 CAULDRON	x		x	x
9 BROADSIDES		x	x	
10 WARHAWK		x	x	

Nagrodę zestaw programów na ZX Spectrum, ufundowaną przez brytyjską firmę ELECTRONICS EXPORT wylosowała Agnieszka Szczęsna z Warszawy.

*Stawek*



NADAWCA: Jarek Jastrzębski 92-200 Rawa Mazowiecka ul. Mszczonowska 24



## SILENT SERVICE

**Komandor Moore** oderwał wzrok od peryskopu. „Peryskop dół. Ster prawo na burt. Cała naprzód” — padały kolejne rozkazy. Kadłub pochylał się lekko i natychmiast wzmógł się szum silników. Wskazówka logu powoli pięła się w górę. USS „Grayback” kierował się w stronę kolejnego tupa starając się zająć dogodną do ataku pozycję. Po kilku minutach w okularze peryskopu można było bez trudu rozróżnić potężny kadłub transportowca wojsk i podążający za nim statek towarowy. Niestety, pozostałe trzy jednostki okazały się eskortowcami. Mimo to komandor J. A. Moore zdecydował się przeprowadzić atak. Jeszcze tylko rzut oka na kalkulator torpedowy i cztery torpedy opuściły wyrzutnie — trzy pomknęły w stronę transportowca, a jedna w kierunku statku towarowego. Nie czekając na wynik ataku okręt podwodny natychmiast zmienił kurs i zaczął zanurzać się głębiej, aby ująć ewentualnej pogoni.

83, 84, 85... powoli mijaly sekundy. Cała załoga nastuchiwała w napięciu. Nagle dał się słyszeć odgłos dalekiego wybuchu i po chwili drugi. Dwie z wyrzeczonych torped osiągnęły cel. Krótki moment radości, lecz zaraz wszyscy ponownie zamieniają się w słuch. W tej chwili na powierzchni eskortowce całą mocą maszyn ruszają w kierunku przypuszczalnego miejsca pobytu napastnika. Czy uda się wyprowadzić je w pole?

Odpowiedź na to pytanie możesz uzyskać, gdy spróbujesz wcielić się w postać dowódcy amerykańskiego okrętu podwodnego. Umożliwi Ci to program symulacyjny „Silent Service”. Napisany on został z ogromną dbałością o wierne oddanie rzeczywistości — nawet poszczególne warianty akcji są oparte na faktach z historii wojny na Pacyfiku. Grający może wybrać ćwiczebne strzelania torpe-

dowe i artyleryjskie, jeden z sześciu ataków na konwój (w różnorodnych warunkach) lub jeden z pięciu patroli. Szczegółowy jest także wybór poziomu trudności (widzialność, zygakowanie konwoju, wadliwe torpedy itd.). Program nie ogranicza się tylko do widoku przez peryskop. Można „poruszać się” po całym okręcie — wyjść na mostek, kontrolować wskaźniki i przyrządy, oglądać mapę (w jednej z czterech podziałości) lub sprawdzić stan okrętu i jego ewentualne uszkodzenia. Wiernie oddany jest również upływ czasu: w podstawowej skali jednej minucie symulacji odpowiada 15 sekund czasu rzeczywistego. Na przykład załadowanie nowej torpedy do wyrzutni wymaga 2,5 minuty (10 minut czasu symulacji). Upływ czasu można oczywiście przyspieszyć — każde naciśnięcie klawisza F dwukrotnie skraca czas (maksymalnie 32 razy: jednej godzinie odpowiadają wtedy 2 minuty). Ze względu na duży stopień skomplikowania programu nieocenioną pomoc oddaje prawie 50-stronicowa instrukcja, rozpoczynanie gry bez niej nie ma w ogóle sensu. Sama instrukcja jest zresztą rewelacyjna. Oprócz standardowego opisu programu zawiera wiele map, dane taktyczno-techniczne okrętów i ich wyposażenia, opis taktyki stosowanej przez okręty podwodne i eskortowce oraz uzupełnione wykresami zestawienie obrazujące wkład amerykańskiej floty podwodnej w walkę z japońskim imperializmem. Gorąco polecam ten program wszystkim użytkownikom komputerów.

POWODZENIA W ŁOWACH!

**Komputery:** Atari XL/XE/ST, Commodore 64/128, Amstrad 464/664/6128, IBM, Spectrum 48/+  
**Producent:** MicroProse Software Inc.

(ziew)

## GHOSTBUSTERS

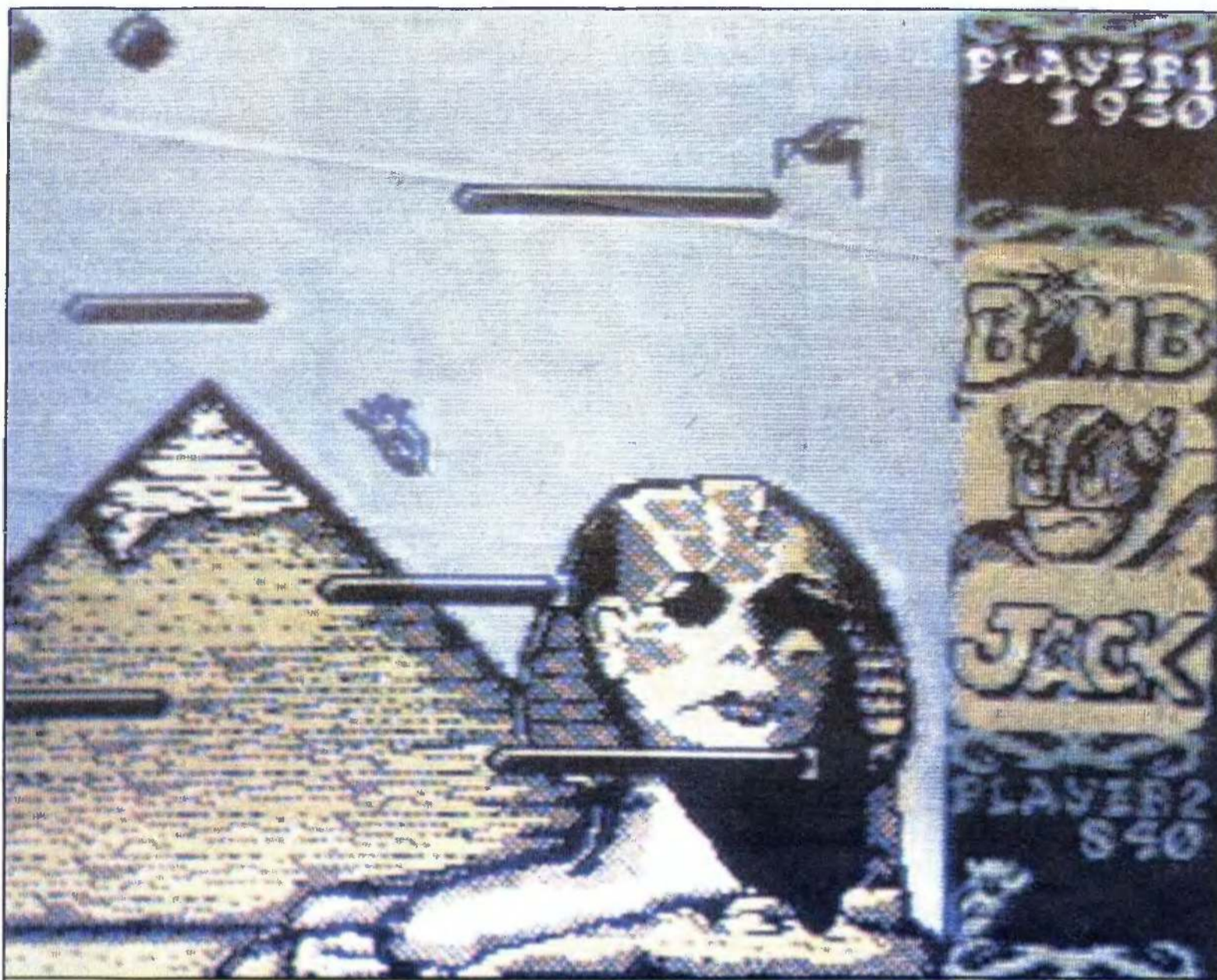
Oto mała niespodzianka dla posiadaczy CHIP-a AY-3-8921A czyli dla wszystkich, którzy mają dostęp do Spectrum 128, +2 lub +3. Po wpisaniu poniższego programu usłyszysz melodię z filmu

GHOSTBUSTERS. Możliwości muzyczne swojego komputera oceń sam.

10 LET a\$ = "M14UX400W0N1C))"

20 LET b\$ = "M1405N9&&3&1CC3ECD\$b5&&1cccc3\$bD5C))"

30 LET c\$ = "M1403N1c&c&\$eeg&\$b&\$b&f&f&f))"  
40 PLAY a\$,b\$,c\$



## BOMB JACK

**Ta gra** nie wymaga myślenia. Pożądana jest natomiast duża spostrzegawczość i — co najważniejsze — mała zręczność. Stajesz się bowiem Bomb Jackiem, legendarnym bioformantem. Zadanie, przed którym teraz stajesz, to zebranie jak największej ilości bomb w ściśle określony sposób. W wykonaniu zadania przeszkadzają mu sympatycznie wyglądające misie i groźny owad z wielkim żądłem. Misie umieją tylko chodzić i gdy spadną na ziemię, przekształcają się w szybko latające pszczoły i nie mniej groźne UFO.

Bomby muszą być zbierane w pewien sposób. Po wzięciu jednej z nich zaczyna palić się sąsiednia. Od tej chwili należy zbierać tylko te bomby, które mają palący się lont. Za to bowiem dostaje się duża ilość punktów.

Po zebraniu wszystkich bomb z ekranu przechodzi się do następnego pomieszczenia. Różnią się one między sobą tłem i rozmieszczeniem platform. Tła powtarzają się co cztery etapy, platformy co kilkanaście.

Co pewien czas pojawia się wirująca płytka z literą. Może to być P, E lub B. Wzięcie płytki z literą P (power) unieruchamia na jakiś czas misie i owada. Można je wtedy zebrać, uzyskując za to pewną ilość punktów. Jeśli nie pospieszysz się, stworci

staną się znowu pełnosprawne i zdolne odebrać Ci życie. Za literkę B (bonus) uzyskasz dodatkową ilość punktów, litera E (extra life) daje dodatkowe życie.

Nasz bohater ma ogromne możliwości poruszania się. Robi to chodząc i skacząc. Do kierowania nim służy pięć przycisków. Wciśnięcie strzału daje skok do 3/4 wysokości ekranu. Aby dolecieć do krawędzi, trzeba wcisnąć przycisk góra + strzał. Strzał w powietrzu powoduje zmianę kierunku lotu z dołu do góry na z góry na dół. Natomiast szybkie przyciskanie strzału wraz z trzymanym klawiszem kierunkowym powoduje lot Jacka w poziomie.

Grę cechuje doskonała grafika i także oprawa dźwiękowa. Można sterować wszystkimi rodzajami joysticków a także z klawiatury. Kilkanaście najlepszych wyników komputer zapamiętuje wraz z imionami ich zdobywców.

Można by powiedzieć, że gra jest prymitywna i dziecinna. Jednak faktem jest, że przyciąga ona każdego.

**Komputer: Amstrad 464/6128, Commodore 64/128, Spectrum 48K/+**

*Marcin Przasnyski*

## KRÓLOWA GIER

Agnieszka Szczęśna, lat 13 uczennica IV klasy Szkoły Podstawowej nr 152 im. Marii Dąbrowskiej zamieszkała w Warszawie przy ul. Korotyńskiego 19 m 70.

**Zainteresowania:** muzyka, gry komputerowe, grafika komputerowa, matematyka.

**Posiadany komputer:** ZX Spectrum +  
**Ulubione gry:** Bomb Jack

**Wymarzony komputer:** Commodore Amiga

**Plany na przyszłość:** Pracować z komputerem

(spo)

# POKE rzysta

Podajemy dziś nową porcję poke'ów do gier na Spectrum. Na początek trzy poke'i, które należy umieścić w ładowaczu (pierwszym segmencie programu). Przypominam, że aby tego dokonać należy wgrać ładowacz za pomocą instrukcji MERGE"", a następnie wpisać podane poke'i po ostatnim LOAD"" CODE, ale przed RANDOMIZE USR (adves).

Teraz obiecane poke'i:

SPINDIZZY

POKE 4872,201

POKE 48401,201

CON-QUEST

POKE 23225,201

PENTAGRAM

POKE 4917,9 (nieśmiertelność)

POKE 50751,0 (nieograniczony skok).

Następne poke'i należy wprowadzić przy pomocy programu COPY-COPY.

FAIRLIGHT

POKE 61893,0, (nieśmiertelność)

POKE 62797,24 (likwiduje ograniczenie na ciężar przedmiotów)

POKE 63478,24 (likwiduje zamknięte drzwi)

POKE 58813,62: POKE 58814,6 (można oglądać rysującą się komnatę)

GHOST'N'GOBLINS

POKE 39857,135: POKE 39858,50: POKE 39859,180

POKE 39860,191

BOBBY BEARING

POKE 29688,175 (zatrzymanie czasu)

SAMANTHA FOX STRIP POKER

POKE 23408,6

JACK THE NIPPER

POKE 44278,58: POKE 44285,58

CAULDRON II

POKE 52133,0

SAI COMBAT

POKE 63364,201: POKE 32421,1

SABOTEUR

POKE 29894,0

SVEEVO'S WORLD

POKE 33219,0: POKE 37008,255

Na koniec podam kilka programów ładujących, które należy umieścić na miejscu oryginalnego programu ładującego.

PYRACURSE

10 CLEAR 24799: LOAD"" CODE 23296

20 POKE 23325,201: RANDOMIZE USR 23299

30 POKE 33446,201: RANDOMIZE USR 29600

PHEENIX

10 CLEAR 24500: LOAD"" CODE 24532

20 LET N=1: POKE 32232,N: LOAD"" CODE

RANDOMIZE USR 30105

Po napisaniu tego programu i wprowadzeniu dwóch segmentów otrzymasz N+1 ludzików i grę rozpocznesz od N+2 screen'u.

Uwaga: Dwa ostatnie programy są opracowane dla kopii dostępnych autorowi.

GRACZ

# JAK TO ROBIĄ INNI



Video Copy Processor P-70B firmy Mitsubishi umożliwia wydruk obrazu z monitora, magnetowidu, kamery video oraz innych urządzeń



Komputer Plautron PT-386: procesor 80386 z zegarem 16 MHz system operacyjny MS-DOS 3.2

## CEBIT '87

**... to prawdziwy raj dla miłośników komputeryzacji, targi komputerowe zaliczane do największych i najbardziej prestiżowych na świecie, organizowane co roku na ogromnych terenach targowych Hannoveru. Tym razem odbyły się w dniach 4-11 marca, zajmując 12 pawilonów wystawowych.**

Trudno byłoby wymienić wszystko to, co było do obejrzenia. Jeszcze trudniej byłoby znaleźć dziedzinę związaną z procesem informatyzacji, która nie byłaby tu reprezentowana. Z naszego punktu widzenia „raj” ten ma jeszcze jedną istotną zaletę — wydaje się być względnie łatwo dostępnym. Lokalizacja geograficzna Hannoveru każe się nam zagłębić w drugi obszar płatniczy jedynie na głębokość ok. 100 km, a więc jest w zasięgu „złotówkowej” benzyny. Dla ewentualnych śmiałków od razu przestroga. Koszty utrzymania w czasie targów trzeba liczyć podwójnie w stosunku do pozostałych dni roku, a absolutny brak miejsc w hotelach już kilkadziesiąt kilometrów od miasta (nie mówiąc o centrum) przypomina rodzime problemy kwatunkowe. W specjalnym komputerowym centrum miejsc noclegowych, najtańszy pokój hotelowy jaki nam po długich poszukiwaniach zaofertowano kosztować miał 120 \$. Nie mogę opisać jego standardu bo ... nie skorzystaliśmy. Istnieje jednak wyjście z tej opresji i to całkiem nam nieobce — kwatery prywatne. W najtańszych (odpowiadających naszej kate-

gorii I) można się przespać „już” za 18 \$. Niestety dorzucają jeszcze obowiązkowe śniadanie za 2 \$. Za mało żeby żyć, za dużo żeby umrzeć.

Atmosfera targów jest wyczuwalna w całym mieście. Już dwadzieścia kilometrów przed miastem pojawiają się na autostradzie targowe tablice informacyjne, a specjalne napisy informują o częstotliwościach, na których można odbierać przez radio informacyjną rozgłośnię targową. Organizatorzy zapewniają zwiedzającym wszelkie możliwe usługi — informacyjne, bankowe, komunikacyjne, zaopatrzeniowe, a nawet religijne różnych wyznań. Tereny targowe, oddalone znacznie od centrum miasta, otaczają olbrzymie puste przestrzenie zamieniane w gigantyczny parking samochodowy. To istotne ułatwienie dla dziesiątek tysięcy odwiedzających targi. Frekwencja jest imponująca, choć koszty samego wejścia na targi nie są bagatelne. Jednokrotny bilet kosztuje 12,5 \$. Jeszcze jedna ciekawostka — wstęp dozwolony jest od 14-tu lat.

Dla zwiedzających dzień targowy zaczyna się o godz. 9.00. Wchodząc wejściem północno-wschodnim, trafiamy na pierwszy i zarazem największy pawilon targowy oznaczony numerem 1. Zlokalizowana w nim jest część administracyjna targów, a liczni wystawcy prezentują tu uniwersalne i specjalistyczne systemy przetwarzania danych, elementy rozbudowywujące możliwości systemów komputerowych, urządzenia biurowe służące komunikowaniu się, dyktafony, maszyny do pisania i do kreślenia, biurowe systemy drukujące i kopiujące, urządzenia do produkcji, obróbki i wykorzystania mikrofilmów do celów biurowych, systemy kasowe i przechowywania mienia i wiele innych urządzeń wspomagających różne specjalistyczne prace biurowe — wszystko podporządkowane systemom sterowanym komputerem. Trudno sobie wręcz wyobrazić jak przyjemna, efektywna i wręcz nobliwa

staje się praca urzędnicza, gdy wykonywać ją przy pomocy tych urządzeń. A jak miło, szybko i kompetentnie byłoby obsługiwani petenci. Można się rozmarzyć.

Niewielki pawilon nr 2 kryje w sobie systemy komputerowe służące wyposażaniu banków, przetwarzaniu danych finansowych, obsłudze klientów, bankowości a także najbardziej przemyślane systemy bezpieczeństwa i ochrony depozytów bankowych. Prezentowano również kompleksowe systemy służące rozwiązywaniu problemów finansowych i kredytowych. Szczególną naszą uwagę zwrócił oferowany za 18000 \$ system komputerowy ze specjalistycznym oprogramowaniem służącym... wprowadzaniu w błąd urzędów skarbowych. Właściciel gwarantował nam jego skuteczność w odniesieniu do 18-tu państw europejskich. Z pewnym zażenowaniem przyznawał, że reklamowany przez niego produkt może nie zdać egzaminu w Belgii i Luksemburgu, ze względu na specyfikę tamtejszych zasad podatkowych. W zupełne osłupienie i bezradność wprawilo go nasze pytanie o skuteczność systemu w konfrontacji z polskim urzędem skarbowym. Odchodziliśmy z triumfalnymi minami.

W kolejnym, trzecim pawilonie przekonujemy się o prawdzie, którą w Polsce ciągle jeszcze odkrywamy. Na programach można zarabiać i to dobrze. Potężne koncerny i zupełnie drobne firmy udowadniają, że proces komputeryzacji ma sens jedynie wtedy, gdy fascynacji sprzętem towarzyszy intensywny rozwój oprogramowania. W najbardziej nawet atrakcyjną postać komputerowej powłoki musi zostać tchnięta software'owa dusza. Praca nad tym, choć na ogół czasochłonna i zawsze związana z pewnym ryzykiem porażki nie jest per saldo marnowaniem pieniędzy. Szkoda, że w tak niewielkim stopniu spotykamy się z efektami tej pracy w tak li-

cznych przecież w Polsce firmach komputerowych. Co pokazano w tej dziedzinie w Hannoverze? Praktycznie wszystko. Kompleksowe rozwiązania służące planowaniu i tworzeniu projektów, jednostki sterowania i kontroli sieciami lokalnymi, systemy informacyjne, wspomagające procesy decyzyjne i zarządzania w rozumieniu rozwiązań całościowych, systemy kontroli transportu, programy kodowania danych, programy doskonalenia systemów, programy użytkowe dla systemów produkcyjnych sterowanych numerycznie, systemy klasyfikacji danych, systemy obrachunku kosztów, systemy użytkowe dla rachunkowości, zakupów i inwestycji i wiele innych.

Ciekawostką samą w sobie jest fakt, że osobny pawilon, oznaczony nr 18 został poświęcony wyłącznie systemom CIM i CAD/CAM. Tak zwane technologie „C” odgrywają stale wzrastającą rolę we współczesnym projektowaniu i wytwarzaniu, wspomagając wymieniane wcześniej programy użytkowe w zakresie opisu graficznego, projektowania i sterowania produkcją. Rozszyfrujmy te angielskie skróty: CIM — computer-integrated manufactory, CAD — computer-aid design i CAM — computer-aid manufacturing.

Organizatorzy CEBIT-u nie ograniczyli się jedynie do prostej ekspozycji prezentowanych towarów. Równolegle realizowany był program informacyjno-produkcyjno-edukacyjny, konferencje i sympozja. Często prelekcje — właściwie nagłościone i wzbogacone obrazem — odbywały się bezpośrednio na stoiskach. Były też krótkie kursy komputerowe. W ramach imprez towarzyszących odbywał się międzynarodowy kongres telekomunikacji, seminarium poświęcone „sztucznej inteligencji”, forum zabezpieczania danych, konferencja nt. funkcjonalności urzędów pocztowych, forum marketingowe itd.

Ze względu na naturalne ograniczenia, przytoczony powyżej opis jest bardzo skróconą relacją z CEBIT 87. Rzucem oka z „lotu ptaka” i to z bardzo wysokiego lotu. Można chyba jednak zaryzykować, nawet na tej podstawie, sformułowanie kilku wniosków z konieczności także uproszczonych.

Komputeryzacja to nie tylko mikrokomputery, a mikrokomputery to nie tylko i nie przede wszystkim PC/XT. A takie wrażenie można było wynieść z wystaw zorganizowanych niedawno w Warszawie w PKiN, w Hotelu „Victoria”. Potrzebne nam są bieżące kontakty z tym co w tej materii dzieje się w najbardziej rozwiniętych krajach. „Nam” tzn. nie tylko specjalistom wysokiej klasy informatykom (oni są na ogół śledzą fachowe czasopisma), ale przede wszystkim potencjalnym użytkownikom, konsumentom komputerowego rynku.

Anegdota o księgowych, którzy wyniki z komputera sprawdzają „na wszelki wypadek” na liczydłe nie rozmija się tak zupełnie z rzeczywistością. Ciągłe przykrwanie postępu cywilizacyjnego do przeciętnego niskiego stanu naszej cywilizacyjnej świadomości rozmija się z sensem. Przelamywanie tych nawyków, stereotypów, przyzwyczajzeń nigdy nie odbywa się bez oporu. Ale opór ten trzeba pokonywać.

Radykalnej zmianie powinno ulec może podejście do kwestii oprogramowania. Szczególnie, że nie mając większych szans w bezpośredniej konfrontacji technicznej, właśnie tworzenie oprogramowania mogłoby się stać naszą polską specjalnością. Nie ograniczajmy się do gier komputerowych czy w najlepszym wypadku tłumaczenia zachodnich programów użytkowych na język polski.

Polski rynek komputerowy to swoiste kuriozum europejskie, ale chwala temu, kto na to pozwolił. Pokonujemy dzięki temu, choć kuchennymi drzwiami, kolejną barierę rozwojową. Ale musimy sobie szybko zdać sprawę, z dużej odpowiedzialności jaka ciąży z tego tytułu na „władcach” tego rynku tak licznych przecież firmach państwowych, prywatnych i polonijnych realizujących proces komputeryzacji kraju. Ich obowiązki powinny być daleko większe od wynikających z prostej rzetelności handlowej. Przecież to właśnie te firmy, sprawując i rozpowszechniając określony sprzęt kształtują nawyki, modę, upodobania. Ustalają swoistą listę rankingową typów komputerów.

Jarosław Pachowski

Komputer Amiga 2000 również pracuje w systemie MS-DOS



## SIR CLIVE POWRACA

**Niepowtarzalny brytyjski fenomen Sir Clive Sinclair zamierza powrócić na rynek komputerowy. Na przeglądzie nowości firm komputerowych, który odbył się w drugiej połowie lutego, w Birmingham, Sir Clive zaprezentował tani (230 funtów szterlingów) komputer przenośny. Jest to, jak twierdzi, idealny komputer osobisty do celów pracy zawodowej, businessu i nauki.**

Niecały rok temu Sinclair zmuszony był sprzedać swe prawa do komputerów marki Spectrum i QL firmie Amstrad. Od czasu zainkasowania pięciu milionów funtów w rezultacie tej nie zaplanowanej transakcji niewiele było o nim słyhać.

O komputerze przenośnym Clive Sinclair myślał już od paru lat. Pandora — tak roboczo nazwano to urządzenie — miała mieć pełnowymiarowy płaski ekran i pamięć nowego typu. W pewnym okresie myślano także o tym, aby była kompatybilna z komputerami Spectrum.

Nowy komputer — Z 88, z którym Sir Clive wchodzi na rynek, niewiele ma wspólnego z pierwotnym pomysłem. Zastosowano w nim tradycyjny, płaski ekran na ciekłych kryształach o pojemności ośmiu wierszy po osiemdziesiąt znaków. Sprzedawane z komputerem oprogramowanie obejmuje edytor tekstów, formularz elektroniczny i notatnik; możliwe jest tworzenie „okienek” pozwalających na jednoczesne wyświetlanie różnych informacji oraz praca w trybie równoległym.

Zalety te wydają się mniej oczywiste, gdy rozważymy je bardziej szczegółowo. Ośmiowierszowy ekran trudno rozsądnie podzielić na mniejsze kawałki, stąd np. jedno z czterech „okienek” służy jedynie do pokazywania stanu baterii. Komputer nie jest wyposażony w stację dysków. Ma pamięć o pojemności 32 KB, z czego jedynie 15 KB dostępne jest dla użytkownika. Ponadto zapowiedziano sprzedaż dodatkowych pamięci — 32 KB w cenie 20 funtów i 128 KB w cenie 50 funtów, a w przyszłości kasety o pojemności 1 MB. Daje to w sumie około 3 MB pamięci, akurat tyle — jak twierdzi Clive Sinclair — aby zapamiętać wszystkie ważniejsze utwory Szekspira. Z podanego wyliczenia wynika więc, że sir Clive ma nienajlepsze zdanie o twórczości tego dramaturga.

Niewątpliwym atutem nowej oferty jest wyjątkowo niska cena, co jednakże pociągnęło za sobą szereg ograniczeń w stosunku do aktualnych tendencji światowych. Np. niewielki ośmiowierszowy ekran nie odpowiada ogólnemu dążeniu do wyposażenia przenośnych komputerów w pełnowymiarowy ekran. Sir Clive jest jednak przekonany, że konkurencyjna cena spowoduje, iż przyszli użytkownicy zaakceptują także fakt, że system operacyjny Z 88 nie odpowiada powszechnie stosowanemu standardom przemysłowym i przyjmować może dane jedynie z IBN i sprzętu z nim kompatybilnego.

Z 88 ma być sprzedawany przez Cambridge Computer, firmę powstałą w zeszłym roku, której jednym z współwłaścicieli jest Clive Sinclair. Ponadto część udziałów posiadają pracownicy, a 10 proc. Sinclair Research. Thorn EMI Datech jest, według oświadczenia Cambridge Computer, „pierwszym producentem” nowego komputera, a Sir Clive oczekuje, że uda mu się osiągnąć poziom sprzedaży równy dziesięciu tysiącom sztuk miesięcznie. Jednakże wloty i upadki Sinclaira pozwalają przypuszczać, że jego nowe osiągnięcie spotka się ze sceptycznym przyjęciem. Kojarzy się on bowiem powszechnie ze stereotypem brytyjskiego badacza, który przy znaczących osiągnięciach technicznych nie bardzo zna się na businessie.

Natomiast fakt, że po ciężkich porażkach w tej dziedzinie znów staje do walki, wzbudziły raczej sympatię wśród ceniących nieugiętość Amerykanów, niż u Brytyjczyków, dla których liczy się przede wszystkim solidność.

Jarosław Barta

# KOMPUTER JUŻ NIE DREWNIANY

**W poprzednim odcinku (Bajtek 3/87) mówiliśmy o tym, że komputer to urządzenie (nawet drewniane), które potrafi zmieniać swój stan. Można zatem wnioskować, że komputer może być zbudowany ze wszystkiego, co przyjmowałoby w sposób stabilny różne stany, dające się zmieniać w sposób kontrolowany.**

Zauważmy, że informacja w naszym urządzeniu może być zakodowana tylko przy pomocy pewnej skończonej liczby pozycji, mogących przyjmować jedynie dwa stany. Najmniejszą liczbą pozycji, kodującą podstawową informację, nazwano bajtem, pojedynczą pozycję bitem, a system logiczny używający tylko dwóch stanów, systemem dwuwartościowym.

W drewnianym komputerze operację dodawania umożliwiła specjalna konstrukcja kanałów i wahadełek. Toczące się kulki ustawiały wahadełka w określonej pozycji. Odczytanie zarejestrowanego stanu wahadełek dawało wynik operacji. W przypadku układów elektronicznych wymagamy aby komputer wykonywał więcej operacji niż tylko dodawanie. Układy, które to potrafią, nazywają się **jednostkami arytmetyczno-logicznymi** (tzw. ALU) lub arytmometrami.

Mówimy już dużo o logice i o układach logicznych, zatem należałoby przybliżyć trochę te terminy. Poznaliśmy już termin logika dwuwartościowa, jako system oparty o dwa stany „prawdę” i „fałsz”. „Prawdę” można utożsamiać z „jedynką logiczną”, a „fałsz” z „zerem logicznym”. Taką logikę nazywano „logiką dodatnią”. Dla układów elektronicznych najczęściej „prawdę” reprezentuje wysoki potencjał, przeważnie 5 V, a „fałsz” niski — 0 V (w praktyce 4,6 — 5,2 V i 0 — 0,7 V).

Prawa logiki, które obowiązują do dziś w konstrukcji komputerów zostały po raz pierwszy przedstawione przez George'a Boole'a w 1847 r. W tzw. „**Algebrze Boole'a**” określone są trzy działania: suma logiczna („lub”, „OR” rys. 1), iloczyn logiczny („i”, „AND” — rys. 2) i negacja („nie”, „NOT”).

W tab. 1-3 przedstawiono odpowiednie tabliczki działań:

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

Tab.1

Suma logiczna  
 $A + B = Q$

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

Tab.2

Iloczyn logiczny  
 $A \cdot B = Q$

A	Q
0	1
1	0

Tab.3

Negacja  
 $A = Q$

Elementy elektroniczne, które wykonują powyższe podstawowe działania nazwano bramkami. Symbole graficzne tych bramek przedstawiono na rys. 3-5.

Pierwsze bramki, produkowane jako układy scalone wykonywane były specjalną techniką, tzw. TTL (ang.: transistor — transistor logic). Stąd ich bardzo popularna nazwa „bramki TTL”. Układem, który zyskał największą popularność, stał się nawet pewnym symbolem okresu technologii TTL, jest bramka typu NAND (rys.6), realizująca już pewne złożone działanie (NOT AND). Tab.4 pokazuje jej funkcje. Jednakże, tu pewne zaskoczenie, budowa jej jest

prostsza od zwykłej bramki AND. Dla bardziej wnikliwych przedstawiam schemat takiej właśnie bramki na rys.7.

A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

Tab.4 działanie „nie i” (NOT AND)

Zobaczmy teraz jak przy pomocy logicznych bramek można zrealizować układy pamiętające i liczące. Zaczniemy od przykładu zwykłego dodawania. Przyjmijmy, że chcemy dodać dwie jednobitowe liczby. Wszystkie możliwe przypadki takiego dodawania przedstawia tab.5. Wynik takiej sumy może być równy dwa (11), zatem dwubitowy, stąd dwa sygnały wyjściowe Q1 i Q2. Na rys.8 znajduje się przykład realizacji powyższego działania, przy pomocy bramek TTL.

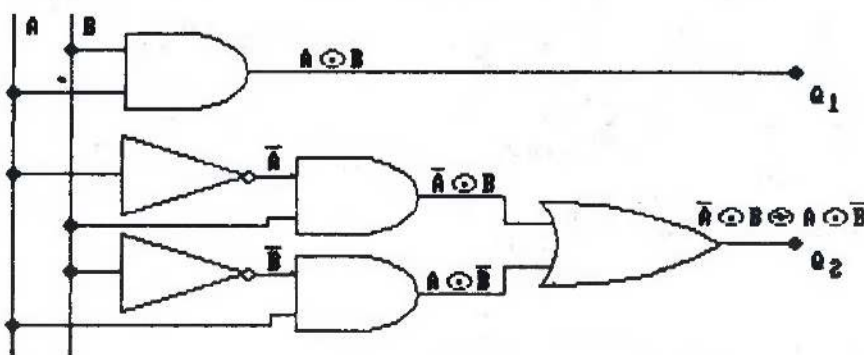
A	B	Q1	Q2	działanie A+B dziesiętnie
0	0	0	0	0+0=0
0	1	0	1	0+1=1
1	0	0	1	1+0=1
1	1	1	0	1+1=2

Tab.5. Wszystkie możliwe przypadki dodawania dwóch jednobitowych liczb

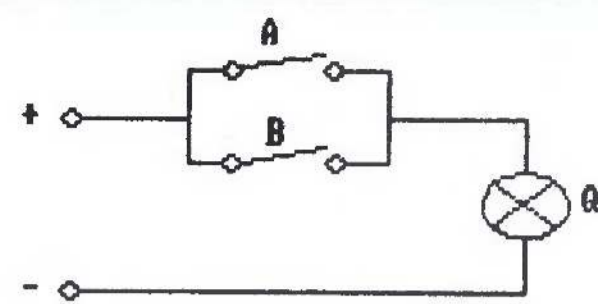
Teraz przyjrzyjmy się drugiego typu układom wchodzącym w skład układów pamiętających czyli rejestrów. Układami tymi są przerzutniki, które definiujemy jako elementy posiadające dwa wyjścia Q i  $\bar{Q}$ . Ich stany zależne są od określonych zmian stanów na wejściach i przechowują informacje pomiędzy dwoma impulsami przełączającymi. Przerzutniki mogą posiadać wejścia synchronizujące i zerujące, lecz nie jest to konieczne. Nazywa się je wtedy **asynchronicznymi**. Takim przerzutnikiem jest **przerzutnik RS**. Posiada on dwa asynchroniczne wejścia informacyjne. Jego działanie obrazuje tab.6, gdzie „Q-” oznacza stan przed przełączeniem, a „Q+” po przełączeniu.

S	R	Q-	Q+
0	0	0	0
0	0	1	1
1	0	0	1
1	0	1	1
0	1	0	0
0	1	1	0
1	1	0	-
1	1	1	-

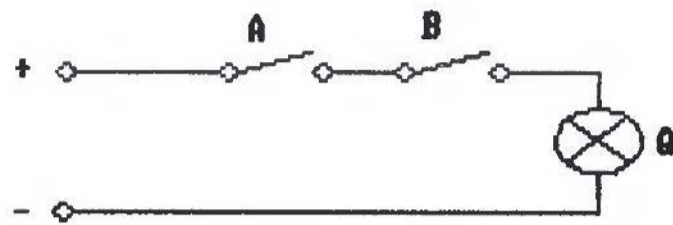
Tab.6. Działanie przerzutnika RS



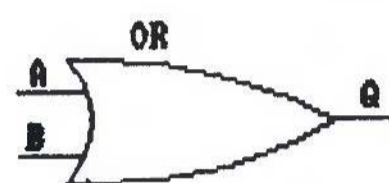
Rys.8 — Schemat ideowy układu sumatora dwóch liczb jednobitowych



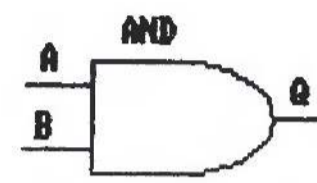
Rys.1 — Działanie funkcji OR



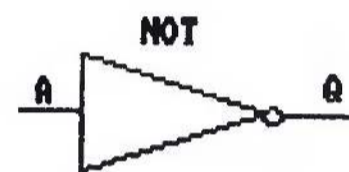
Rys.2 — Działanie funkcji AND



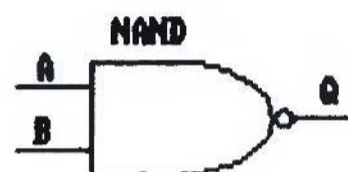
Rys.3 — Bramka OR



Rys.4 — Bramka AND



Rys.5 — Bramka NOT



Rys.6 — Symbol bramki NAND

Działanie przerzutnika jest następujące. Gdy  $R=Q$  i  $S=Q$  stan wyjścia nie zmienia się. Gdy  $S=1$  i  $R=Q$ , wtedy przerzutnik ustawiamy, tzn.  $Q=1$  i  $Q=Q$ . Gdy  $R=1$  i  $S=Q$  a  $Q=1$ . Gdy  $S=R=1$ , nie można przewidzieć czy przerzutnik przełączy się czy też nie.

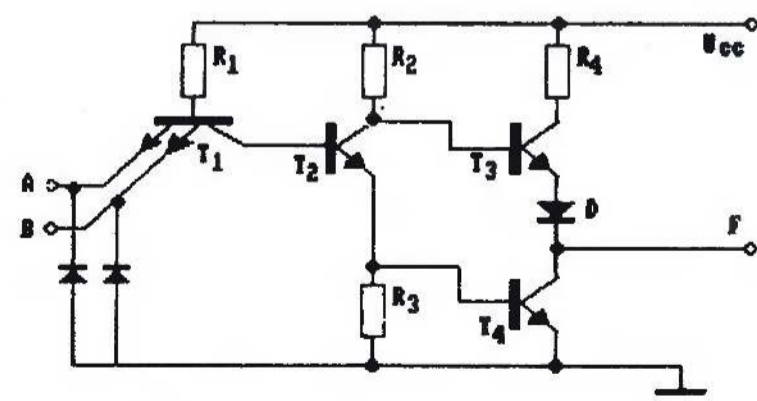
W bardzo dużym skrócie poznaliśmy pewne możliwości podstawowych bramek cyfrowych realizujących działania logiczne. Do zwykłego jednobitowego dodawania potrzebnych było sześć bramek. Do ośmiobitowego już widać, że minimum 48. Do najprostszego elementu rejestru, czyli przerzutnika potrzebne były cztery bramki. Policzymy, dla 48KB Spectrum, to już daje  $49152 \times 8 \times 4 = 1572864$  bramek. A co z innymi układami znajdującymi się w tym towarzystwie...?

O tym za miesiąc.

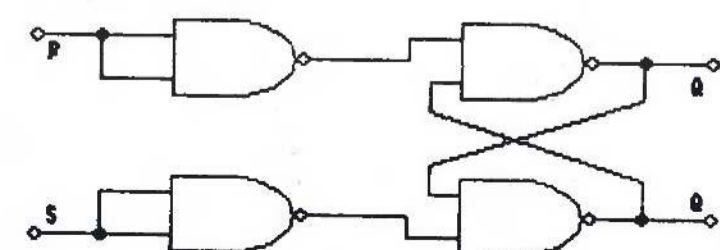
**Krzysztof Czernek**

### LITERATURA

- (1) J. Pieńkos, J. Turczyński: Układy scalone TTL w systemach cyfrowych; WK 1980
- (2) P. Misiurewicz: Układy automatyki cyfrowej; WSP 1981



Rys.7 — Schemat ideowy bramki typu NAND



Rys.9 — Przerzutnik RS

# ROZSĄDNA OSZCZĘDNOŚĆ (2)

**Miesiąc temu zajmowaliśmy się konstruowaniem i oceną algorytmów, czyli metod rozwiązywania zadań.**

**Jeśli ktoś z Was miał problemy z zapisaniem podanych algorytmów w postaci programów, to może pomogą w tym szkice programów realizujących podane miesiąc temu metody.**

Jakie wnioski możemy wyciągnąć? Jeśli dwa programy napisane są w tym samym języku i działają na takim samym komputerze, w identycznych warunkach, to decydującym czynnikiem określającym czas działania jest algorytm każdego z tych programów. Inaczej mówiąc, szybkość działania programu zostaje wstępnie określona jeszcze przed napisaniem pierwszej instrukcji — poprzez wybór metody rozwiązania. Co dalej? Ano to, że warto docenić rolę algorytmu, a więc także rolę i znaczenie przygotowania przemyślanego projektu programu zanim zaczniemy pisać sam program. Warto wreszcie, gdy już wymyślimy metodę rozwiązania danego problemu, zastanowić się, czy jest to metoda wystarczająco efektywna, czy też raczej należy próbować znaleźć jakiś lepszy sposób.

Tutaj niewielka dygresja. Wśród informatyków zawodowych samo zapisanie gotowego algorytmu w postaci programu nie jest zwykle uważane za największy problem. Ten etap często nazywany jest „kodowaniem”, a przecież kodowanie w swoim typowym znaczeniu to czynność prawie mechaniczna, nie wymagająca wiele myślenia. Mamy tutaj do czynienia ze świadomym i celowym oddzieleniem czynności opracowania (wymyślenia nowego lub wybrania z kilku istniejących i opublikowanych) algorytmu od czynności zapisania tego algorytmu we właściwym języku programowania. U początkujących programistów jedną z przyczyn trudności jest to, że zaczynają pisać program niezbyt dokładnie wiedząc jakie czynności musi ten program wykonać, żeby otrzymać rozwiązanie. Na to nakładają się słaba znajomość konstrukcji języka i w rezultacie napisanie nieskomplikowanego programu zaczyna stwarzać duże problemy. Skoro już była mowa o „zawodowcach”, warto dodać że istnieje wiele publikacji zawierających właśnie opisy znanych algorytmów rozwiązywania typowych, spotykanych w codziennej praktyce zadań, a ambicją wielu autorów jest wymyślenie algorytmu rozwiązującego jakiś konkretny problem szybciej niż wszystkie inne znane do tej pory metody.

Ale wróćmy do naszego głównego wątku. Ustaliliśmy, że gdy komputer i język są już ustalone czas działania programu zależy przede wszystkim od metody rozwiązywania problemu. W tej sytuacji nie ma się co łudzić, że radykalnie skrócimy czas działania programu stosując różne dziwne sztuczki, np. stosowanie zmienionych o możliwie najkrótszych nazwach (dotyczy to tych interpreterów BASIC-a, które dają do dyspozycji nazwy wieloznakowe), czy też usuwanie z programów komentarzy. Tak naprawdę, to wśród krążących między mikrofanami metod na przyspieszenie działania programów nastąpiło przemieszczanie pomysłów wartościowych z pomysłami przypominającymi spluwaniu przez lewe ramię i wymawianie magicznych zaklęć, np. obecność komentarzy może mieć naprawdę znaczący wpływ tylko na wielkość programu, ale nie na czas jego działania.

Dlaczego tak zdecydowanie wypowiadam się przeciwko, ogólnie mówiąc sztuczkom, czy nie można ich stosować równoległe z wyborem efektywnego algorytmu? Otóż dlatego, że ogromna większość tych „sposobów” powoduje znaczne zmniejszenie czytelności tekstu programu dla człowieka. A to z kolei znacznie zwiększa nakład pracy potrzebny na zrobienie czegośkolwiek z programem — znalezienie w nim błędu, wprowadzenie modyfikacji itd. Inaczej mówiąc, zyskamy troszkę czasu na skróceniu działania programu, ale za to stracimy go dużo więcej przy okazji pracy z programem. Spróbujcie sobie uświadomić, ile razy programista musi przeczytać tekst dużego programu podczas jego uruchamiania. To, że programy są czytane nie tylko przez komputery, ale również (a może nawet przede wszystkim) przez ludzi jest oczywiste chyba dla wszystkich. Wynikający z tego faktu, równie oczywisty wniosek, że programy należy pisać tak, by ludzie mieli jak najmniej kłopotów z ich odczytaniem jest niestety przez wielu programistów ignorowany. Powiedzmy to sobie szczerze i brutalnie, dotyczy to przede wszystkim programistów słabych.

Aby zademonstrować różnicę w odbiorze tekstu jaką może spowodować różny sposób zapisu, starałem się odpowiednio „preparować” teksty algorytmów i programów zamieszczonych w tym artykule. Efekt pozostawiam Wam do oceny i ewentualnie jako materiał do wyciągnięcia własnych wniosków.

Świetnie, powie ktoś, moje programy piszę sam i sam je czytam, poza tym doskonale wiem, co jest w którym miejscu, nawet bez komentarzy i niezależnie od układu graficznego tekstu. Na pewno jest to prawdą dla programów małych, ale czy rzeczywiście można łatwo zapamiętać wszystkie szczegóły programu mającego kilkaset czy nawet kilka tysięcy linii? Poza tym pamiętajmy, że czasami trzeba wrócić do programu po dłuższej przerwie. Czy za rok nie dojdiesz do wniosku, że jakiś program warto przerebować? I czy przypadkiem nie okaże się wtedy, iż jest on napisany tak, że zupełnie nie widać co się w nim dzieje.

Zbliżyliśmy się w ten sposób do problemu tzw. dokumentacji technicznej, czyli opisu stworzonych programów. Nie będziemy rozwijać tego tematu, gdyż od amatorów programujących dla przyjemności nikt takiej dokumentacji nie będzie wymagał (od zawodowych programistów jest ona wymagana). Może jednak pisząc duży program warto zrobić kilka stron notatek — na pewno przydadzą się w

przyszłości. Zwróćmy wreszcie uwagę, że najprostszą formą takiej dokumentacji — opisu działania programu są właśnie komentarze. Ich dodatkową zaletą jest to, że dopóki mamy tekst programu nie mogą nam zginąć, no i zawsze są pod ręką gdy czytamy program. Pamiętajmy tylko, że jeśli wprowadzamy zmiany do programu, część komentarzy może stać się nieaktualna i zamiast pomagać, będzie wprowadzać w błąd. Czyli, zmieniając program zmieniamy również komentarze — od razu, a nie kiedyś później!

Pisanie dobrych komentarzy, krótkich a równocześnie treściwych i zrozumiałych nie tylko dla autora jest rzeczą trudną — trzeba się tego uczyć tak jak dobrego programowania. Najlepiej zacząć już od dziś.

Jak wspominałem, pisanie programów w sposób czytelny nie ma większego wpływu na szybkość jego działania, może natomiast zwiększyć ilość zajętego przez program miejsca. Co robić, gdy miejsca zaczyna brakować. W większości wypadków wyjście jest następujące: znaczną część miejsca w programie zajmują dane, np. tablice, bardzo długie zmienne tekstowe itp. Na ogół do testowania programu pełna wielkość obszarów danych nie jest potrzebna, co pozwala uzyskać dodatkowe miejsce na dłuższy program. Po ostatecznym przetestowaniu wersję z komentarzami zapisujemy do archiwum — bo może jeszcze kiedyś się przyda, natomiast z wersji roboczej usuwamy komentarze, uzyskując dodatkowe miejsce w pamięci. Możemy to zrobić zupełnie spokojnie, bo od tej pory tę wersję programu będzie czytał tylko komputer, który komentarze i tak ignoruje.

Teraz szata graficzna programu. Trudno ustalić jednoznaczne kryteria rozmie-

szczenia tekstu, zapewniające jego największą czytelność, gdyż wiele zależy tutaj od indywidualnych upodobań, jednak wieloletnia praktyka wykazuje, że są pewne typowe rozwiązania. Można tu zaliczyć:

Umieszczanie instrukcji zmieniających sterowanie w programie (instrukcje warunkowe, pętle) jako pierwszych instrukcji w linii.

Tzw. wcięcia, czyli dodanie przed każdą instrukcją pętli kilku spacji. Powoduje to, że zawartość pętli jest na wydruku przesunięta w prawo, przez co wyróżnia się z treści programu. To samo dotyczy instrukcji wykonywanych warunkowo — zawartych wewnątrz IF THEN.

Oddzielanie od siebie fragmentów programu realizujących odrębne funkcje, np. pustą linią, lub rzędem gwiazdek.

Używanie nazw mnemotechnicznych, tzn. kojarzących się z zawartością nazywanego obiektu (nie na wszystkich mikrokomputerach dysponujemy nazwami o długości odpowiedniej do tego celu).

Nie ma w tej dziedzinie jednolitego, zaakceptowanego przez absolutnie wszystkich standardu, jednak powyższe zasady są akceptowane dość powszechnie i dobrze byłoby nie oddalać się od nich zbyt daleko.

Pora na małe podsumowanie. Nie namawiałem nikogo do pisania nieefektywnych programów, chciałem tylko wskazać gdzie należy szukać efektywności, gdzie zaś oszczędności nie należy robić, gdyż mogą to być oszczędności nieopłacalne. Równocześnie musimy sobie zdawać sprawę, że wybór algorytmu i czytelny zapis to tylko część składników określających jakość programu. Pozostałymi zajmiemy się w niedalekiej przyszłości.

Andrzej Pilaszek

```

1 REM realizacja algorytmu 1
10 DIM TX(1000):PRINT "Dlugosc tekstu"
15 INPUT N:FOR I=1 TO N
20 TX(I)=32:NEXT:FOR I=1 TO N STEP 2
30 TX(I)=100:NEXT:GOSUB 1000:I=1
40 IF I=N GOTO 100
50 IF TX(I) (>) 32 GOTO 90
55 FOR J=I+1 TO N
60 TX(J-1)=TX(J):NEXT
70 N=N-1
80 GOTO 40
90 I=I+1:GOTO 40
100 GOSUB 1000
110 END
1000 FOR I=1 TO N:PRINT TX(I):NEXT
1010 RETURN

10 REM realizacja algorytmu 2
20 DIM TX(1000)
21 REM do testowania wykorzystujemy
22 REM początkowy fragment tablicy T
23 REM o dlugosci N, ktora wczytamy:
24 PRINT "Podaj dlugosc tekstu "
25 INPUT N
30 REM wypelniamy tablice spacjami-
40 REM dane do testow
50 FOR I=1 TO N
60 TX(I)=32
70 NEXT
80 REM wstawienie znakow innych niz
90 REM spacja pozwoli sprawdzic program
100 FOR I=1 TO N STEP 2
102 TX(I)=100
105 NEXT I
160 GOSUB 1000 : REM wyswietl. danych
170 REM *****
180 REM ** zaczynamy program **
190 I=1
200 REM szukamy pierwszej spacji
210 IF TX(I) = 32 GOTO 260
220 I=I+1
230 GOTO 210
240 REM spacja znaleziona,
250 REM ustawiamy w1 i w2
260 W1=I
270 W2=I
280 REM petla przegladania tekstu
290 IF W2 = N+1 GOTO 370 :REM 370=koniec
300 W2=W2+1
310 IF TX(W2)= 32 GOTO 290
320 REM tu w2 wskazuje niespacje -
330 REM przepisanie znaku
340 TX(W1)=TX(W2)
350 W1=W1+1
360 GOTO 290 : REM na poczatek petli
370 N=W1-1
380 GOSUB 1000 : REM wyswietl. wynikow
390 END
1000 REM *****
1005 REM procedura wydruku tablicy
1010 PRINT "TABLICA TX"
1040 FOR I=1 TO N
1050 PRINT TX(I) " ";
1060 NEXT I : PRINT " "
1070 RETURN

```

# WSZYSTKO DLA WSZYSTKICH

**PC XT/AT**

**AMSTRAD**

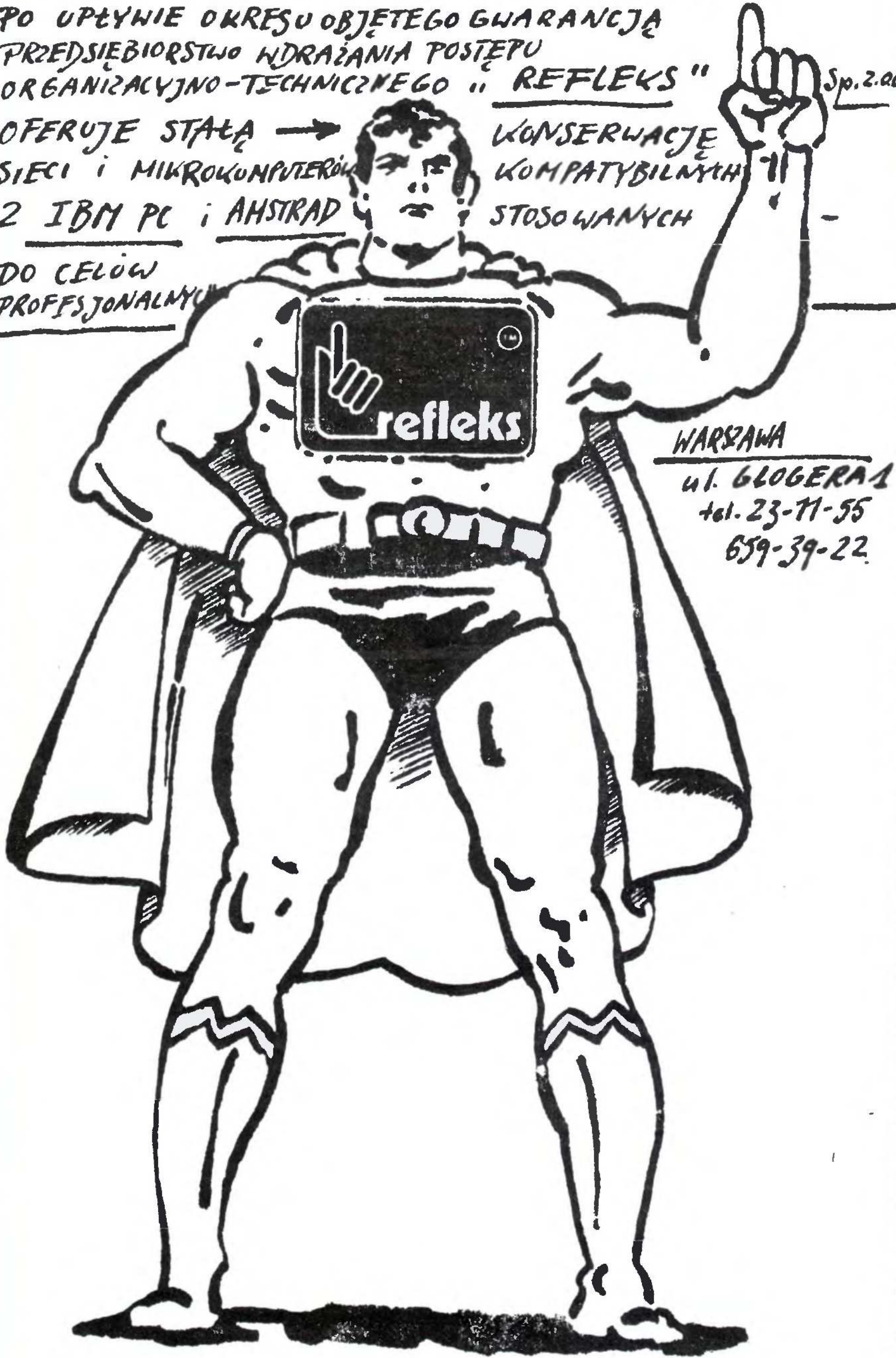
## SUPERSERVIS

PO UPŁYWIE OKRESU OBJĘTEGO GWARANCJĄ  
PRZEDSIĘBIORSTWO WDRAŻANIA POSTĘPU  
ORGANIZACYJNO-TECHNICZNEGO "REFLEKS" Sp. z o.o.

OFERUJE STAŁĄ →  
SIECI I MIKROKOMPUTEROWĄ  
Z IBM PC I AMSTRAD

KONSERWACJĘ  
KOMPATYBILNYCH  
STOSOWANYCH

DO CELÓW  
PROFESJONALNYCH



**WARSZAWA**  
ul. GLOGERA 1  
tel. 23-71-55  
659-39-22.

Sprzedam mysz do ZX - Spectrum. Informacje po nadesłaniu koperty zwrotnej: Anna Wichtowska, ul. Fałata 68 m. 10, 87-100 Toruń.

G-17

ATARI IRATA — SOFTWARE — poleca:  
— ponad 1000 programów  
— instrukcje  
— katalogi i informacje gratis  
Skr. poczt. 160 66—400 Gorzów Wlkp. tel. 249-58

G-43

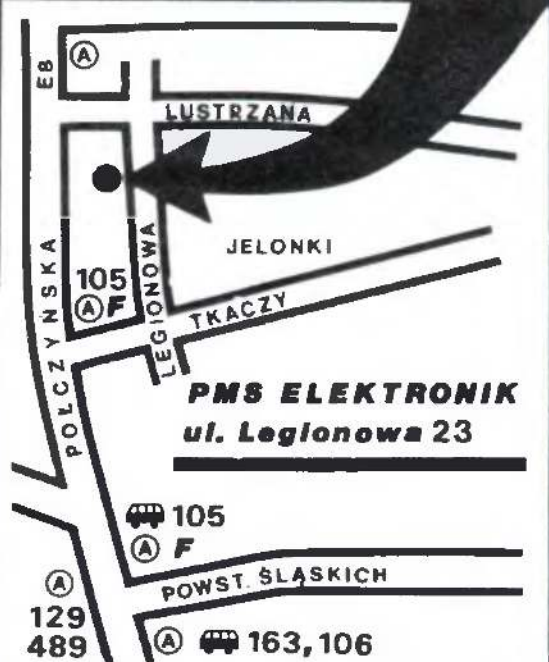
Programy na ATARI i SPECTRUM tanio wypożyczysz na miejscu lub za zaliczeniem pocztowym. Informacje za załączeniem koperty i znaczka. MICROMAN 40—181 Katowice, ul. Osikowa 66, tel. 585-106.

D-43

Programy na ATARI 800 XL wymienię i sprzedam — tanio — 200 zł. Andrzej Ostrowski, ul. Malczewskiego 5/9, 58-309 Wałbrzych.

G-12

### Sinclair ZX Spectrum SERVICE



- Naprawy
  - Programy
  - Interfejsy
  - SP-DOS
- 9<sup>00</sup>-16<sup>00</sup>

PMS elektronik, ul. Legionowa 23, 01-343 Warszawa.

K-79

Studio „RETURN”  
ATARI ● AMSTRAD ● SPECTRUM ● IBM  
wypożyczalnia programów i literatury, Warszawa, ul. Targowa 32, tel. 19-10-34 g. 11—19. Rachunki oraz wysyłka pocztą.

D-56

Commodore — 64  
Poradnik programisty — Basic  
Poradnik programisty — kod maszynowy  
Instrukcja obsługi  
Tłumaczenia  
Sprzedam  
Nawrocki  
Łódź, ul. Pstrągowa 35 m. 18

D-68

ZX SPECTRUM  
Naprawiam komputery. Wykonuję interfejsy do joysticków typu KEMPSTON I SINCLAIR. Il opisane w miesięczniku „Komputer” nr 9/86. Roczna gwarancja. Jerzy Dymecki, Meissnera 14 m. 1, Warszawa.

D-52

REKLAMUJ SIĘ W BAJTKU ● REKLAMUJ SIĘ W BAJTKU



**ENTER**  
computing

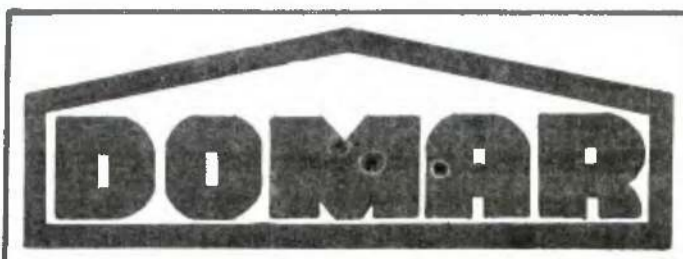
WYSYŁKOWA WYPOŻYCZALNIA  
PROGRAMÓW

02-105 W-wa 21 P-3

**Przypominamy:**

- wszechstronne oprogramowanie ZX SPECTRUM TIMEX 2048 (w tym wszystkie nowości)
- rachunki dla instytucji i osób prywatnych
- termin realizacji zamówienia 5 dni
- wyczerpujące informacje po nadesłaniu koperty zwrotnej.  
**30% zniżki dla młodzieży szkolnej.**

D-189



PRZEDSIĘBIORSTWO  
HANDLU ARTYKUŁAMI  
WYPOSAŻENIA  
MIESZKAŃ W ŁODZI

**PROWADZI SPRZEDAŻ:**

**MIKROKOMPUTERÓW** oraz sprzętu  
**KOMPLEMENTARNEGO**

dla odbiorców indywidualnych i pozarynkowych — w sklepach w Łodzi  
— ul. Piotrkowska 91 tel. 32-20-65  
— ul. Dzierżyńskiego 32a

**ZAPRASZAMY**

K-85

**ATARI**

Programy, literatura, instrukcja  
**POCZTA**

katalogi i informacje bezpłatnie

Termin realizacji 5 dni.

Wysyłamy rachunki.

**ATR-SOFTWARE**  
66-542 Zwierzyń P-1

D-34

**ATARI**

Wysyłka programów na  
cały kraj. Najniższe  
ceny. Katalogi gratis.  
22-600 Tomaszów Lub.,  
woj. Zamość, ul. Króla  
Zygmunta 7/21 tel.25-05.

G-8

**JAK REKLAMOWAĆ  
SIĘ W BAJTKU?**

Reklamy przyjmuje Młodzieżowa  
Agencja Wydawnicza (Redakcja Wy-  
dawnictw Poradniczych i Reklamy),  
04-028 Warszawa. Al. Stanów Zjed-  
noczonych 53, pokój 313. Tel. 10-  
-56-82.

Cena ogłoszeń: 200 zł za 1 cm  
plus dodatki za kolor

Zakład usługowy, Łódź, Roosvelta  
2 wykona na zamówienie **cartridge**  
do C64 i C128 umożliwiające  
kopiowanie programów zabezpie-  
czonych w dowolny sposób. Przys-  
pieszający współpracę z 1541 do 5  
razy. Naprawiamy Spectrum, C64,  
C128, Atari.  
Informacje: zaadresowana koperta  
(może być bez znaczka).

G-31

Agencyjny Zakład Usługowy SPHW  
W-wa, ul. Mokotowska 61 poleca:  
— ZX SPECTRUM SERVICE  
— wypożyczanie i nagrywanie pro-  
gramów na SPECTRUM i ATARI  
— wejście monitorowe w OTV i OTVC  
dla komputerów i video  
— przestrajanie UKF i PAL-SECAM  
Zamiejscowym w tym samym dniu  
gwarancja, rachunki  
ZAPRASZAMY w godz. 12.00—19.00.  
tel. 28-20-27.

D-69

**Firma LUKMART LTD**

uprzejmie informuje państwa, że wprowadziła do  
sprzedaży wysyłkowej najwyższej jakości sprzęt elek-  
troniczny.

- Rewelacyjne PLOTERY HITACHI 672XD,
- DRUKARKI roku 1987 SEIKO-SHA MP/1300  
oraz całą gamę wyrobów komputerowych.

Nasz adres:  
LUKMART LTD.  
58, St. Mary's Road  
London W5 5EX  
Anglia

Wszystkim zainteresowa-  
nym udzielamy informacji  
technicznych i wyjaśnień  
dotyczących zakupu.

tel. 567—7913 telex 922536 lonelk g

K-71

# GIEŁDA (ceny na dzień 1987.04.04)

\* Giełda Bajtka czynna jest w Warszawie przy ul. Grzybowskiej 35 (Szkoła Podstawowa nr 25) w każdą sobotę od 14 do 19 oraz w każdą niedzielę od 10 do 16 przy ul. Saskiej 78

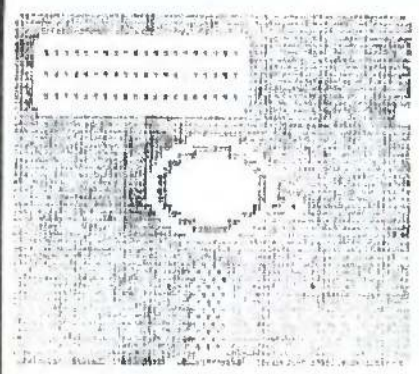
## SINCLAIR

## COMMODORE

## ATARI

## AMSTRAD

	GIĘŁDA BAJTKA (tys. zł)	KOMIS (tys. zł)	AUSTRIA (średnie) (öS)	FRANCJA (średnie) (FF)	RFN (średnie) (DM)	WLK. BRYT. (średnie) (£)
ZX 81	38	—	550	—	49	—
ZX Spectrum 48 KB	109	—	1100-1400	—	150-250	45-65
ZX Spectrum Plus	120-140	140-160	1590	1350	180-300	65-75
ZX Spectrum 128 +2	275-300	—	—	1800-1990	—	110-140
Drukarka SEIKOSHA GP 50S	90-120	160	—	—	199	60-65
Interface Kempston	7-15	—	250	200	35	6-9
Joystick QUICKSHO II	9-11	12	150-160	80	9-15	5-7
C-64	200	—	3400	1900	370-449	90-110
C-128	390-410	580	6500	2890	590	210-230
C-128D	800	—	12000	6850	1250	390-410
Amiga z monitorem kolorowym	—	—	—	—	2900	990
Magnetofon 1531	35-40	45	900	350	49-65	25
Stacja dyskietek 1541	245	—	4500	1950	450	110-150
Stacja dyskietek 1570	—	—	6900	2300	490-540	160
Drukarka MPS 801	200	300	2900	2200	199	—
Dyskietki 5 1/4 (średnia jakość)	0.65-1.5	1.2-2	10-25	7	0.5-1.8	0.8-2
800XL	140	150	1500	900	140-180	60
130 XE	210	—	2100	1400	360	110
Stacja dyskietek 1050	200	—	2200	2150	370	130
Drukarka 1029	240	300	1990	—	—	85
ATARI 520 STM st. dysk. 0.5Mb	800	1.1 mln	—	4000	970	395
464 z monit. monochromat.	230-300	390	6500	2690	520	160
6128 z monit. monochromat.	520	700-800	11000	3990	900	250
6128 z monit. kolor.	600	1 mln	15800	5290	1250	320
PCW 8256	—	1.3 mln	—	5920	—	340
Dyskietki 3 <sup>1/2</sup>	4-6.5	6-7	—	35	7-12	3.5-4
Stacja dyskietek 3 <sup>1/2</sup> do 464	195	295	6000	2600	549	140
PC 1512 SD	—	2.4 mln	—	5920	1189	440



INDYWIDUALNY  
**BANK**  
DANYCH

Nazywam się **Tomek Gawłowski**, jestem uczniem, mam 15 lat. Posiadam mikrokomputer: Philips VG 8020 i Spectrum 48K. Interesuję się informatyką, muzyką, kolarstwem, elektroniką i geografią Polski. Mam różne programy użytkowe i edukacyjne (około 30 graficznych oraz gry). Pragnę nawiązać kontakt z posiadaczami mikrokomputera VG 8020 w celu wymiany literatury i oprogramowania. Adres: ul. Łagiewnicka 80/98 m.57, 91-456 Łódź.

**Krzysztof Petz**, uczeń, 15 lat. Mikrokomputer: Atari 130XE, magnetofon XC-12, monitor. Interesuję się informatyką i wszystkimi dziedzinami z nią związanymi. Oprogramowanie: kilka programów użytkowych (programy kopiujące, bazy danych, „Koala” — wersja kasetowa itp.) oraz kilka gier. Chciałbym nawiązać kontakt z tymi posiadaczami komputera, którzy interesują się jego użytkowaniem. Proponuję wymianę oprogramowania. Adres: ul. Dzierżyńskiego 1/31, 11-200 Bartoszyce.

**Maciej Karczewski**, uczeń, 17 lat. Mikrokomputer Amstrad CPC 664, drukarka i monitor. Interesuję się informatyką i żeglarsstwem. Oprogramowanie: kilkadziesiąt programów, przeważnie użytkowych. Proponuję wymianę programów, także listingów. Adres: ul. Włociańska 16/57, 01-710 Warszawa.

**Jacek Bednarkiewicz**, student, 24 lata. Mikrokomputer Spectrum Plus, GP 500A. Zainteresowania: wykorzystanie komputerów w budownictwie lądowym. Oprogramowanie: kilka własnych programów związanych z budownictwem oraz firmowe programy graficzne, Logo, Data Basic 3.0, kilka gier. Chciałbym wymieniać programy i doświadczenia. Adres: Pl. Piastowski 3/1, 84-300 Łęborg.

**Michał Dylewski**, uczeń, 15 lat. Mikrokomputer Spectrum Plus, magnetofon i pióro świetlne. Zainteresowania: elektronika i informatyka. Oprogramowanie: kilkadziesiąt gier, bazy danych, edytory tekstów, Pascal, Asembler i inne. Chciałbym wymieniać programy, czasopisma, schematy urządzeń peryferyjnych oraz doświadczenia związane z pracą Spectrum Plus. Adres: ul. Wspólna 35 m.5, 00-519 Warszawa.

**Marek Kollbek**, uczeń, 16 lat. Mikrokomputer Commodore C16 z rozbudowaną pamięcią do 64KB, magnetofon. Zainteresowania: informatyka, języki programowania. Posiadam kilkadziesiąt gier oraz dużo własnych programów użytkowych. Chciałbym poprzez wymianę uzyskać ciekawe programy użytkowe. Adres: ul. Krowoderskich Zuchów 11/46, 31-271 Kraków

**Jacek Łowicki**, pracownik Okręgowego Urzędu Miar, 23 lata. Mikrokomputer: SHARP MZ-731 z wbudowanym plotterem i magnetofonem. Interesuję się techniką Audio-Video i informatyką. Dysponuję około 70 programami, przetłumaczoną instrukcją obsługi oraz podręcznikiem Basic'a. Proponuję wymianę oprogramowania i literatury. Adres: ul. Piekarska 13 m. 5, 87-800 Włocławek

**Darek Tomczyk**, uczeń, 14 lat. Mikrokomputery: Amstrad PCP464 i Sega 3000. Interesuję się informatyką i fotografią. Oprogramowanie do Amstrada: kilkanaście gier; do komputera Sega: 10 programów użytkowych i gry. Proponuję wymianę oprogramowania. Adres: ul. Poniatowskiego 3/20, 66-400 Gorzów Wielkopolski

**Konrad Czermarmazowicz**, uczeń, 13 lat. Mikrokomputer Commodore 16. Interesuję się komputerami i sportem. Oprogramowanie: kilkanaście gier i programów użytkowych. Pragnę nawiązać kontakt z posiadaczami C-16 w celu wymiany oprogramowania. Adres: ul. Wybickiego 6, 51-144 Wrocław

**Tomasz Moździk**, uczeń L.O., 16 lat. Mikrokomputer TRS-80 f-my Tandy (64 kB RAM). Zainteresowania: matematyka, mikrokomputery. Chciałbym nawiązać kontakt z posiadaczami tego mikrokomputera w celu zdobycia oprogramowania i wiadomości na temat współpracy TRS-80 z urządzeniami peryferyjnymi. Adres: Podostrówek 5, 16-150 Suchowola

**Marek Suchecki**, uczeń, 16 lat. Mikrokomputer Atari 800XL, magnetofon XC-12. Interesuję się żeglarsstwem, matematyką i informatyką. Oprogramowanie: Kilkanaście gier (w tym kilka własnych), edytor tekstów, program kopiujący. Chciałbym dowiedzieć się czegoś o Asemblerze Atari i wymieniać oprogramowanie. Adres: ul. Kilińskiego 25/4, 58-100 Świdnica

**Filip Hoffman**, uczeń, 12 lat. Mikrokomputer Timex/Sinclair 100 z pamięcią rozszerzoną do 16 kB, magnetofon. Interesuję się matematyką i informatyką. Oprogramowanie: 17 gier i 19 programów użytkowych. Proponuję wymianę programów i doświadczeń. Adres: ul. Czernika 13 m.45, Łódź Widzew

**Jacek Witt**, uczeń, 15 lat. Mikrokomputer Commodore 16 z pamięcią rozbudowaną do 64 kB, firmowy magnetofon. Posiadam ponad 50 programów i proponuję ich wymianę. Adres: ul. Kiedrzyńskiego 3/1, 82-300 Elbląg

**Marcin Kozioł**, uczeń, 10 lat. Mikrokomputer ZX Spectrum, magnetofon. Interesuję się informatyką, filatelistyką i sportem. Oprogramowanie: wybór około 300 najciekawszych gier. Proponuję wymianę oprogramowania. Adres: ul. Oświęcimska 2/78, 32-500 Chrzanów.

## DRUGA GIĘŁDA BAJTKA

Z inicjatywy Komitetu Osiedlowego nr 14, dyrekcji Zespołu Szkół Chemicznych oraz Koła Komputerowego przy Technikum Chemicznym powstała w Warszawie nowa giełda komputerowa firmowana przez „Bajt-

ka”, która zaprasza w każdą niedzielę w godz. 10.00—16.00 do swojej siedziby przy ul. Saskiej 78. Dla wygody uczestników przewidziana jest rezerwacja stolików oraz tablica ogłoszeń. Jesienią organizatorzy oraz redakcja „Bajtki” przewidują przygotowanie wspólnej imprezy mikrokomputerowej połączonej z pokazami najlepszych programów edukacyjnych napisanych przez młodych programistów.



FOT. EUGENIUSZ SOŁODUCHA

## KOMPUTER CI POMOŻE

### Prezentujemy kolejny klub zgłoszony do „Konkursu o Złotą Dyskietkę Bajtka” — Klub Mikrokomputerowy przy Pałacu Młodzieży im. Wielkiej Rewolucji Październikowej w Szczecinie.

Dziś każde dziecko wie, że najprostsza droga do przyjaźnienia się z informatyką wiedzie przez gry i zabawy komputerowe.

— Taka propaganda komputeryzacji przynosi niewiele korzyści — twierdzi mgr inż. Leszek Łukaszewski. — Stukanie w klawisze i wykonywanie poleceń odczytywanych z monitora — denerwuje się — to nie wszystko. W naszym klubie zamiast na gry, uwagę dzieci kierujemy na to co najważniejsze: poznawanie podstaw informatyki i elektroniki, opanowywanie — na początek — języka BASIC, wreszcie — samodzielne układanie programów komputerowych.

Uczestniczą w zajęciach klubu mikrokomputerowego Pałacu Młodzieży w Szczecinie. Zrzesza on 104 dzieci — uczennice i uczniów miejscowych szkół podstawowych i zasadniczych zawodowych. Wiek uczestników — od 8 do 16 lat. Frekwencja na zajęciach niemal stu procentowa.

— Na prasowy anons o zapisach do klubu — opowiada inż. Łukaszewski — jedynego dostępnego uczniom szczecińskich szkół podstawowych, w wyznaczony dzień ustawiała się przed pałacem potężna kolejka dzieci, przybyłych w znacznej części w asyście rodziców. Chętnych było ponad pół tysiąca!

Nowo powstały klub, utworzony na bazie działającego tu od lat Klubu Młodych Elektroników, mógł przyjąć — niestety — tylko ograniczoną możliwościami sprzętowymi i lokalowymi liczbę uczestników. Przyjęto najlepszych, tych, którzy wykazali się pewną wiedzą.

— O dużym zainteresowaniu mikrokomputerami — opowiada mgr Zbigniew Armada dyrektor Pałacu Młodzieży —

dzieży — dowodziło choćby powodzenie z jakim spotkała się wcześniej przeprowadzona przez TNOiK oraz Uniwersytet Szczeciński akcja pn. „Wakacje z Komputerm”. Naszą inicjatywę w tej mierze poparło kuratorium.

Za otrzymaną stamtąd dotacją udało się kupić cztery komputery ZX Spectrum Plus, trzy monitory (czwarty uzyskano poprzez zaadaptowanie telewizora „Unitramtu 617”), trzy magnetofony oraz drukarkę. Sprzęt ten — dodajmy — wykorzystywany jest tu „na okrągło”. Członkowie klubu sami zakładali instalację elektryczną, dorabiali kable połączeniowe, zamontowali do mikrokomputerów wyjścia niskiej częstotliwości na monitory, wykonali cały zestaw plansz poglądowych.

Sebastian Kopaniecki — uczeń klasy VI Szkoły Podstawowej nr 51 z uwagą wpisuje do pamięci komputera ułożoną przez siebie pierwszą część programu pn. „Geografia”. To praca wymagająca skupienia, a przy tym — obok wiedzy geograficznej — także umiejętności kreślarskich, plastycznych, a nawet muzycznych...

Jego koledzy szkolni — rówieśnicy: Daniel Masełko i Marek Dawidowicz, pracujący przy sąsiednim komputerze, proszą dziennikarza o podanie danych personalnych. Chłopcy z wprawą kodują przekazane im informacje, demonstrują zasady korygowania błędów, wyszukiwania danych itp.

Przy kolejnym stoliku pracuje Mariusz Zabraniak, uczeń klasy VII Szkoły Podstawowej nr 45. Mariusz, rocznik 1973, zanim zgłosił się we wrześniu ubiegłego roku do klubu mikrokomputerowego, języka BASIC uczył się „na sucho” z „Młodego Technika”. Teraz uczęszcza tu wraz z młodszym bratem 11-letnim Dariuszem. Obaj, przy współudziale szkolnego kolegi, 13-latką Marka Madęły, są autorami napisanego w BASIC-u programu informacyjnego o Pałacu Młodzieży. Pytanie i odpowiedzi ułożone są w wersjach językowych: polskiej, niemieckiej, angielskiej i hiszpańskiej. Ciekawa oprawa graficzna oraz dźwiękowa uatrakcyjnia korzystanie z programu.

Sporządzenie tego programu zajęło trójce autorskiej blisko pół roku. Oczywiście, nie obyło się bez pomocy instruktora oraz osób znających języki. Mariusz — dodajmy — współuczestniczył także w opracowaniu wersji rosyjskiej programu. Wymagało to zaprogramowania w mikrokomputerze alfabetu rosyjskiego — cyrylicy. Jest on też autorem interesującego, opartego na materiale klasy VI i VII programu pt. „Matematyka”.

Członkowie tego klubu opracowali znacznie więcej programów, 10 najlepszych, wśród których znalazły się uprzednio wymienione, zgłoszono do ogłoszonego przez „Kurier Szczeciński”, PTI oraz Kuratorium — Wojewódzkiego Konkursu dla Młodzieży Szkolnej.

Już nawet krótka wizyta uprzytomnia, że poza potrzebą opanowania zasad programowania, zajęcie to wymaga od autorów rzetelnego poznawania tematów, których dotyczą programy, ułatwia zatem utrwalanie i rozszerzanie wiedzy ogólnej i specjalistycznej. Trudno nie zgodzić się z obowiązującym tu hasłem: „Myśl sam, a komputer ci pomoże”.

*Zygmunt Kowalski*



## KONKURS „O ZŁOTĄ DYSKIETKĘ BAJTKA”

Prezentujemy listę klubów, które zgłosiły się do współzawodnictwa „O Złotą Dyskietkę Bajtka”.

- MERIZAP-Ostrów Wlkp., ul. Poznańska. Klub działa od 1985 r. przy Zesp. Szk. Techn. MERA-ZAP w Ostrowie.
- MNEMONIK-Wrocław, ul. Ostrowskiego 30. Klub działa od 1986 r. przy Zakładach Elektronicznych ELWRO.
- BAJT-Lązy, ul. Szkolna 2. Klub działa od 1986 r. pod patronatem Komendy Hufca ZHP Łązy.

- SYNTAX ERROR-Koszalin, ul. Reytana 15. Klub działa od 1986 r. pod patronatem ZMW.
- BAJTEK-Kraków, ul. Z. Augusta 5. Klub działa od 1986 r. przy Zarz. Dzieln. ZSMP Kraków-Śródmieście.
- COMMODORO-BAJT-CLUB-Bydgoszcz, ul. Kijowska 58. Klub działa od 1987 r. bez opiekuna.
- HARCBAJT-Gdańsk, ul. Za Murami 2/10. Klub działa od 1987 r. przy Komendzie Chorągwi ZHP w Gdańsku.
- MANIAK-Warszawa, ul. Wasilkowskiego 7. Klub działa od 1985 r. przy Klubie Osiedlowym na Ursynowie.
- SINCLAIR CLUB-Kraków, os. Zielona 18/25. Klub działa od 1985 r. bez opiekuna.
- METRO-POST-Piotrków Tryb., ul. Papińskiego. Klub działa od 1986 r. przy Szkole Podst. nr 3.
- BIT-Warszawa, Igańska 26/47. Klub działa od 1986 r. bez opiekuna.
- POLTRONIC-Częstochowa, ul. Iwaskiewicza 3. Klub działa od 1987 r. bez opiekuna.
- TOMBIT-Stegna Gdańska, ul. Jagiełły 3a/9. Klub działa od 1986 r. bez opiekuna.
- INFORMIK-Bydgoszcz, ul. Grunwaldzka 7/1. Działa od 1986 r. bez opiekuna.
- CHOMIK-Warszawa, ul. Nerudy 1. Klub działa od 1986 r. przy Klubie Mieszkańców CHOMICZÓWKA.

- SAVE-Słupca, ul. Powstańców Wlkp. 22. Klub działa od 1985 r. przy Zesp. Szk. Ekonom. im. M. Jackowskiego w Słupcy.
- BAJTEK-Zamość, ul. Akademicka 8. Klub działa od 1986 r. przy I LO im. J. Zamoyskiego w Zamościu.
- KLUB MIKROKOMPUTEROWY-Szczecin, Al. W.P. Klub działa od 1986 r. przy Pałacu Młodzieży w Szczecinie.
- INFORMIK-Olkusz, ul. Kantego 5. Klub działa od 1986 r. przy Szk. Podst. nr 1 w Olkuszcu.
- FENIKS-Recz, ul. Ratuszowa. Klub działa od 1986 r. przy Miejsko-Gminnym Ośrodku Kultury w Reczu.
- ZŁOTY AMSTRAD-Złoty Stok, ul. 1-go Maja 10. Klub działa od 1987 r. pod patronatem Zakładów Tworzyw i Farb Złoty Stok.
- ATARI-Choszczno, ul. Bohaterów W-wy 7. Klub działa od 1986 r. przy Choszczeńskim Domu Kultury.
- LOK-Nowy Sącz, ul. Bieruta 12A. Klub działa od 1986 r. przy Zarz. Woj. LOK.
- Mr ATARI-Kozienice, ul. Konarskiego 6/40. Klub działa od 1985 r. bez opiekuna.
- ATARI-Olsztyn, ul. Murzynowskiego 8. Klub działa od 1986 r. pod patronatem Spółdzielni Mieszk. JAROTY.

## Drogi Bajtku!



*Od dwóch lat posiadam komputer ZX Spectrum +. W tym okresie zbierałem znaczną ilość programów (tj. około 200), a moje zbiory nadal się powiększają. Teraz na rynku pojawił się nowy Spectrum, z wbudowanym magnetofonem i pojemnością pamięci 128 KB. Zamierzam kupić go; mój stary komputer sprzedaję, ale nie wiem czy sprzedać wszystkie programy, które zbierałem. Czy ZX Spectrum + 2 128 KB jest kompatybilny z ZX Spectrum + i czy wszystkie programy pisane na wersję 48 KB wpiszą się do + 2?*

**Artur Bajer**  
ul. Krótka 3/27  
57-540 Łądek-Zdrój

Producent nowego Spectrum nie gwarantuje pełnej zgodności tego modelu ze Spectrum +. Wpływa na to zarówno odmienne oprogramowanie stałe komputera — zmieniona i rozszerzona pamięć ROM zajmuje 32 KB, jak też inna konfiguracja sprzętowa. Błędne działanie programów ze Spectrum + może wystąpić zwłaszcza w przypadku, gdy wykorzystują one niektóre procedury zawarte w pamięci ROM.

Nie należy zatem się spodziewać, że wszystkie programy przez Pana posiadane dadzą się uruchomić na nowym komputerze. Z czasem wyłoni się z pewnością — i będzie lepiej znana — lista programów działających (lub nie) na Spectrum + 2. Do tego czasu radziłbym wstrzymać się ze sprzedażą programów i po prostu... sprawdzić samemu, które warto pozostawić.

*W pierwszym tegorocznym wydaniu Waszego pisma został zamieszczony w dziale „Jak to robią inni” wywiad z Jackiem Tramielą. Była w nim mowa o komputerze 32-bitowym, opartym na mikroprocesorze Motorola 68020. O jaki komputer chodzi?*

**Piotr Lenarczyk**  
ul. Batorego 36/23  
05-400 Otwock

Niestety, na zaspokojenie Pańskiej ciekawości trzeba nieco poczekać. Firma ATARI nie ujawniła bowiem szczegółów swych prac projektowych nie tylko dziennikarzowi „Bajtku”, ale także żadnemu innemu czasopiśmie. Kiedy tylko ów nowy komputer pojawi się na rynku, nie omieszkamy i my zaprezentować go naszym czytelnikom. Zgodnie z wypowiedzią p. Tramiela, powinno to nastąpić jeszcze w bieżącym roku.

*Który z mikrokomputerów: ZX Spectrum 48KB, TIMEX 2048, Commodore VIC-20, C+4, C 16, C 116 najbardziej się nadaje do gier? Ile istnieje gier (z kasy magnetofonowej) na nim?*

*(nazwisko i adres do wiadomości redakcji)*

Z wymienionych urządzeń w zasadzie tylko Spectrum oraz jego nieco solidniej wykonana wersja — TIMEX 2048 spełniają podany warunek. Można szacować liczbę napisanych i wydanych na ten komputer gier na ok. 3000, z czego jakąś wartość dla użytkownika przedstawia być może 300—400.

Życzę udanego zakupu a później dużej ilości wolnego czasu.

*Czy planuje się wydawanie czasopisma „Informatyka w szkole” dla nauczycieli wykładających elementy informatyki lub wykorzystujących komputery na zajęciach przedmiotowych, wzorem np. „Matematyki”, „Fizyki w szkole” itp...*

**mgr inż. Piotr Jurchyszyn**  
Oddział Doskonalenia Nauczycieli  
ul. Dubois 36  
45-067 Opole

W chwili, gdy piszę te słowa, nic jeszcze nie wiadomo ani o zamysłach, ani o tworzeniu takiego periodyku. Nie trzeba chyba nadmieniać, jak jest on potrzebny. Wskazał

Pan na lukę, której nie są w stanie zapłacić wydawnictwa popularne, przeznaczone dla hobbystów. „Bajtek” również takim czasopiśmie stać się w całości nie może; z przyjemnością natomiast powitamy i podejmiemy współpracę z redakcją takiego czasopisma, kiedy ukaże się ono na rynku.

*Uprzejmie proszę o pomoc w zakupie nowego komputera ZX Spectrum 48KB. Jednocześnie proszę o pomoc w uzyskaniu oprogramowania do w/w komputera.*

*Jestem inwalidą I grupy na stałe przykutym do wózka inwalidzkiego, co uniemożliwia mi wyjazdy na giełdy komputerowe i spotkania z innymi użytkownikami.*

*Ponadto chciałbym nawiązać stałą korespondencję z osobami posiadającymi powyższy komputer.*

**Waldemar Kościan (17 lat)**  
ul. Parzęczewska 24A bl. 11 m. 21  
95-100 Zgierz

„Bajtek” nie zajmuje się, niestety, rozprawdaniem sprzętu komputerowego ani oprogramowania; apeluję jednak o pomoc tych, którzy te słowa przeczytają: pomóżcie! To właśnie dziesiątki Waszych serdecznych, bezinteresownych listów sprawiły tyle radości Markowi L. — również niepełnosprawnemu młodemu człowiekowi, zafascynowanemu swoim komputerem. Jeżeli pamiętacie jeszcze podobny apel sprzed tych kilku miesięcy — odezwcijcie się!

Jeżeli Pan pragnie nabyć komputer nowy, sądzę, że interesującą ofertą może być TIMEX 2048, zgodny ze Spectrum,

sprzedawany przez Centralną Składnicę Harcerską w cenie 106 tys. zł. Do tej instytucji przekazałem też Pana pismo.

*Czy Atari 520ST jest kompatybilny z 130XE?*

*Czy na zachodzie Atari 130XE jest sprzedawany z myszką?*

*Czy istnieje program typu GEM na Atari 130XE i czy jest on dołączony do zestawu?*

*Pisaliście w 9/86 numerze „Bajtku”, że nie wolno włączać drukarki do komputera, jeśli którekolwiek z urządzeń jest włączone. Czy można włączyć drukarkę tak, by przy włączonym komputerze włożyć wtyk kabla drukarki, a dopiero potem włączyć drukarkę do sieci?*

*(nazwisko i adres do wiadomości redakcji)*

Modele 520ST i 130XE nie są zgodne.

W krajach zachodnich (RFN, Wielka Brytania) można z reguły nabyć Atari 130XE w zestawie z myszką lub bez niej. Nie jest ona w każdym razie standardowym wyposażeniem komputera, podobnie jak wykorzystujące ją programy.

System GEM w swojej pełnej wersji nie został opracowany dla 8-bitowych komputerów Atari. Istnieje za to szereg programów, wykorzystujących charakterystyczną dla GEM technikę WIMP (Windows, Icons, Mouse, Pull-down Menus).

Nie powinno się podłączać żadnych urządzeń w podany przez Pana sposób. Należy przyjąć zasadę, że wszelkie gniazda lub złącza krawędziowe są w momencie tą-

## ATARI • ZX SPECTRUM

INSTRUKCJA, OPISY, LITERATURA  
KATALOGI — GRATIS  
SZKOŁY I KLUBY — ZNIŻKA  
WYSYŁKA NA CAŁY KRAJ

Wypożyczalnia programów  
D.H. „SEZAM” II p. g. 16.00 — 19.00,  
00-849 Warszawa UPT 66 skr. p. 14.

D-3

**agencja mikrocomputerowa**

**41-200 Sosnowiec P-157**

**UDOSKONALENIA TECHNICZNE KOMPUTERÓW**

**INSTRUKCJE**  
**OPISY**  
**PROGRAMY**  
**poczta**

telefon:  
699-649

**IBM**  
**PC**

**ATARI**  
**AMSTRAD**  
**COMMODORE**

**KATALOG GRATIS**

K-87

czenia szczególnie podatne na zwarcie, w związku z czym nie powinny one być w tej chwili pod napięciem. Sam bez obawy do włączonego komputera podłączam tylko joystick i magnetofon (oczywiście nie magnetofon Commodore).

**Chciałbym kupić komputer Atari 800XL. Czy warto go kupić, czy dalej oszczędzać na Atari 130XE? Jakie są zalety i wady obu komputerów?**

**Krzysztof Sokołowski  
ul. Pińska 52  
71-043 Szczecin**

Oba typy Atari mają zarówno zwolenników, jak przeciwników, pomimo obiegu opinii, jakoby 130XE był lepszą wersją popularnej „osiemsetki”. Można przecież rozszerzyć pamięć 800XL do 256 KB. Klawiatura 130XE, pomimo że starannie wykonana, niektórym użytkownikom odpowiada mniej. Poza tym zarówno możliwości, jak i budowa obu typów są bardzo podobne.

**Czy w komputerze Spectravideo SVI 738 (80 KB RAM) można rozszerzyć pamięć operacyjną do 128 KB? Jeżeli tak, to w jaki sposób dokonuje się instalacji tej dodatkowej pamięci?**

**(nazwisko i adres do wiadomości redakcji)**

Jako wyposażenie dodatkowe komputera Spectravideo produkowane są pakiety 64 KB dodatkowej pamięci RAM. Pakiet taki przyłącza się bezpośrednio do gniazda „expansion slot” komputera — zatem operacja ta nie wymaga od użytkownika żadnych kwalifikacji. Urządzenie to produkowane przez firmę Spectravideo nosi nazwę SVI-747 RAM Expansion Cartridge; można jednak używać pakietu dowolnego innego komputera MSX.

Jak łatwo zauważyć, pojemność otrzymanej w podany sposób pamięci nawet przekracza Pańskie wymagania.

**Od niedawna używam programu BETA-BASIC. Jest on wygodny w użyciu, ale jak używać w nim własnych znaków graficznych jeżeli kryją się tam właśnie komendy BETA-BASIC-a?**

**Napotykam także na inny problem. Jak zlikwidować migotanie animowanego na ekranie znaku? Jeżeli piszę np. jakąś grę, to przy poruszaniu mój bohater miga. Czy mogę wykonać w programie jakąś przeróbkę, która by to migotanie usunęła?**

**Jarosław Juszkiewicz  
ul. Piastów 11/148  
40-868 Katowice**

BETA-BASIC jest wyposażony w standardową instrukcję KEYWORDS (uzyskujemy ją przez wciśnięcie GRAPHICS 8), której wykonanie w trybie bezpośrednim powoduje powrót klawiatury do normalnego trybu graficznego (KEYWORDS 0) lub odwrotnie (KEYWORDS 1). Sama komenda KEYWORDS jest dostępna z klawiatury zawsze, a zatem nawet w zwykłym trybie graficznym nie możemy uzyskać znaku GRAPHICS 8 — spacji graficznej i jej negatywu. W razie koniecznej potrzeby możemy uzyskać je przez użycie funkcji CHR\$.

Migotanie animowanych na ekranie znaków jest trudno usunąć całkowicie z poziomu języka BASIC — interpreter działa zbyt wolno. Możemy jednak starać się je poważnie ograniczyć poprzez zmianę struktury programu.

Program animujący należy pisać tak, aby podczas jego wykonywania czas pomiędzy usunięciem z ekranu znaku a naniesieniem go w nowej pozycji był jak najkrótszy. Oznacza to, że program nie powinien wykonywać żadnych dodatkowych instrukcji pomiędzy tymi dwiema operacjami. Wyliczenie współrzędnych drukowania, sprawdzenie stanu klawiatury itp. czynności muszą być wykonywane w czasie, gdy wszystkie animowane znaki są widoczne na ekranie.

Jeżeli taka przeróbka programu nie przyniesie wystarczających efektów, pozostaje użycie kompilatora.

**Chciałbym zaopatrzyć się w komputer Commodore C-64. Proszę o bliższe dane o tym urządzeniu.**

**Przeczytałem ostatnio na łamach pewnego czasopisma iż BASIC, jako język prosty, powinien jedynie znaleźć zasto-**

**sowanie w kalkulatorach programowanych i w podobny sposób wykorzystywanych mikrokomputerach. Jako język prymitywny o ograniczonych strukturach danych, nie dający możliwości programowania strukturalnego itp. cechach nie powinien być stosowany w edukacji. Podobno LOGO spełniałoby te wymogi, gdyby nie to, iż w obecnym czasie dysponujemy bardzo nikiłą bazą programową w tym języku. Chciałbym dowiedzieć się więcej na ten temat.**

**Ryszard K.  
Oświęcim.**

Szerszy opis Commodore 64 zamieszczony został w numerze 3—4/86 naszego pisma.

BASIC jest faktycznie przestarzałym językiem programowania — powstawał w zasadzie równocześnie z FORTRANEM; tego ostatniego zaś nikt dziś już, na szczęście, nie próbuje używać w szkołach. Równocześnie nie jest on wcale łatwiejszy do przyswojenia od wielu innych języków programowania. Brak możliwości programowania strukturalnego czyni go zaś prawie zupełnie nieprzydatnym do realizacji dużych i złożonych zadań programistycznych. Sądzę, że przytoczona przez Pana opinia jest słuszna — właściwe użycie BASIC-a to proste programy i nieskomplikowane zadania. Coraz więcej słyszy się o krajach, w których usuwa się BASIC ze szkół, lub o firmach programistycznych, które od kandydata do pracy wymagają deklaracji, że nie rozpoczynał nauki programowania od BASIC-a. Stąd też program przedmiotu Elementy Informatyki w naszym kraju milcząco zakłada, że językiem pierwszego kontaktu uczniów z kom-

puterem będzie Logo. Logo wybrano u nas i w wielu innych krajach świata głównie dlatego, że postać tego języka jest wynikiem wieloletnich studiów właśnie nad zastosowaniem komputerów w edukacji — a nie wprowadzeniem nauki programowania do szkół, co warto podkreślić.

Siłą rzeczy jednak w bardzo wielu wypadkach BASIC jest używany jako język publikacyjny w opracowaniach dotyczących mikrokomputerów. Jest to oczywiście — interpreter BASIC-a jest zawsze tym programem, który każdy użytkownik danego typu komputera zawsze ma pod ręką, czego nie można powiedzieć np. o Pascalu lub Prologu. Niekiedy jest też oprogramowaniem bardziej sprawdzonym i niezawodnym od translatorów wczytywanych z dysku czy kasety.

Sądzę, że oprogramowanie, pozwalające szerokiej rzeszy użytkowników korzystać z języka Logo, jest już w Polsce dostępne; problemem dotąd nie rozwiązany pozostaje brak literatury nim się posługującej.

Z drugiej strony istniejące interpretry Logo (zarówno polskiej, jak i angielskiej wersji tego języka) nie są pozbawione wielu poważnych wad. Przykładami mogą tu być wersje na Spectrum i Dr Logo dla Amstradów. Czekamy też na pojawienie się kompilatorów Logo.

*Marcin*

## KLAN NIETYPOWYCH

Jestem posiadaczem nietypowego na naszym rynku komputera firmy COLECO o nazwie ADAM. W USA wyprodukowano sporą liczbę tego typu komputerów (950 tys. szt.), więc może ktoś z rodaków przywiózł go do Polski? Posiadam dwie wersje BASIC-a, SmartLOGO, DataCALC, FastFILLER, PaintMASTER, gry, programy edukacyjne, literaturę. Chciałbym nawiązać kontakt z posiadaczami podobnego komputera lub gry telewizyjnej tej firmy. Mam 30 lat, jestem socjologiem; zainteresowania: fotografia, edukacyjne zastosowanie komputerów.

**Henryk Zyborowicz  
ul. Ślusarska 6 m.2  
82-300 Elbląg**

## NAJTAŃSZE KOMPUTERY W KRAJU PRZEDSIĘBIORSTWO ZAGRANICZNE

# GALLECH

oferuje

kompleksową dostawę sprzętu dostosowanego do użytkowania w ofertowej LOKALNEJ SIECI KOMPUTEROWEJ kompatybilnej z IBM PC NETWORK oraz IBM PC TOKEN-RING NETWORK.

Komputer kompatybilny z IBM PC/XT w cenie od 1.698 tys. zł. w dowolnej konfiguracji.

Komputer kompatybilny z IBM PC/AT w cenie od 6.381 tys. zł w konfiguracji zestawionej na życzenie klienta.

Urządzenia peryferyjne:

- bogaty wybór drukarek „STAR 15”
- pamięci taśmowe (streamer) 60 MB
- digitizer 12" x 12" firmy SUMMAGRAPHICS
- myszy elektroniczne typu MICROSOFT MOUSE
- pióra świetlne
- plottery A-3 firmy ROLAND

Na naszych klientów czeka również bogata oferta oprogramowania.

Na zakupiony u nas sprzęt dajemy 12 miesięczną gwarancję a także zapewniamy odpłatny serwis pogwarancyjny.

Zamówienia prosimy kierować na adres:  
Przedsiębiorstwo Zagraniczne GALLECH  
Miechów ul. Raclawicka 31 tel. 304-57.

K-37

## DATA BOX

pośrednictwo —  
— wypożyczalnia

- komputery
- części
- programy
- audio-video
- RTV

KOMPUTEROWY  
BANK OFERT  
00-750 Warszawa  
ul. Nowosielecka 22  
tel. 40-13-13.

D-57

## Spółdzielnia Rzemieśnicza „PRODUCENT”

18-400 Łomża,

ul. Nowogrodzka 200, tel. 60-62  
oferuje po atrakcyjnych cenach do  
komputera ZX Spectrum:

- Interface systemu Sinclair,
- Interface systemu Kempston,
- Interface systemu Kempston z dodatkowym układem regulacji prędkości działania komputera,
- pióro świetlne z kasetą zawierającą programy:

a) do obliczeń statystyczno-matematycznych,  
b) program graficzny  
oraz Joystick do komputerów ZX Spectrum,  
Commodore, Atari, Amstrad.  
Szczegółowych informacji udziela:  
Zakład Elektroniczny „Magra”  
Jednaczewo 27, 18-400 Łomża  
Marek Kruszewski.

K-66

# KUBUŚ RATUJE KRÓLEWNĘ



## Cześć Maluchy!

Opowiem wam bajkę...

Na samym środku Morza Straszliwego znajdowała się maleńka wyspa. Na niej zły, skrzydlaty Smok więził piękną Królownę. Smutny był los Królowny. Nikt nie mógł odnaleźć wyspy zagubionej pośród fal, a jeśli nawet jakiś śmiałek zdołał osiągnąć jej brzegi, ginął w śmiertelnej potyczce ze Smokiem. Pewnego dnia, młody i dzielny rycerz o imieniu Kubuś usłyszał historię nieszczęśliwej Królowny. Namówił dzikie gęsi o mocnych skrzydłach by przeniósły go nad wzburzonym morzem do Krainy Okrutnego Smoka. Zabrał ze sobą płaszcz swego ojca — Dobrego Czarownika i Kulę z Kryształu Krzemu.

Kula mówiła Kubusiowi w jakiej odległości znajduje się Królowna i Smok, a płaszcz miał umożliwić ucieczkę z wyspy. Czy Kubuś zdoła jednak odnaleźć biedną dziewczynę nim Smok pochwyty go w swe szpony? Przekonajcie się sami.

Pamiętacie przygody Kubusia w Tajemniczym Zamku z numeru 8/86 „Bajtka”? Jeden z moich przyjaciół — przedszkolaków powiedział mi wtedy: „to fajna gra, ale można w nią grać tylko raz: potem już zawsze wiadomo co robić”. To prawda, w tym przypadku chyba większą frajdą było wymyślanie i pisanie programu (świadczy o tym cała masa listów z podobnymi gramami) niż sama zabawa. Nasz dzisiejszy program będzie — mam nadzieję — ciekawszy dla grającego.

Wyspa Smoka w naszym programie to coś w rodzaju wielkiej szachownicy o wymiarach 40x40 kratek. Po niektórych polach Kubuś może poruszać się swobodnie, na inne, z kolei nie może wejść — to np. skały, mury. Komputer przedstawia (w sposób dla nas niewidoczny) postacie na tej szachownicy i sprawdza ich wzajemne położenie.

Zacznijmy od działania podprogramów. Są one umieszczone na końcu, od linii 1000 wwyż.

```

99 REM ***** pozycje startowe *****
100 LET ok=0
110 GOSUB 1000
120 GOSUB 2000
130 GOSUB 3000
140 IF ok=0 THEN GOTO 110
150 LET xrycerza=x
160 LET yrycerza=y
170 GOSUB 1000
180 GOSUB 2000
190 GOSUB 3000
200 IF ok=0 THEN GOTO 170
210 LET xkrolewny=x
220 LET ykrolewny=y
230 GOSUB 1000
240 LET xsmoka=x
250 LET ysmoka=y
299 REM ***** komunikat na ekranie *****
300 PRINT
310 LET x=xrycerza
320 LET y=yrycerza
330 GOSUB 2000
340 PRINT "Wokół ciebie "; obszar$
350 LET x=xrycerza+1
360 GOSUB 2000
370 PRINT "Po wschodniej stronie "; obszar$
380 LET x=xrycerza-1
390 GOSUB 2000
400 PRINT "Po zachodniej stronie "; obszar$
410 LET x=xrycerza
420 LET y=yrycerza+1
430 GOSUB 2000
440 PRINT "Po południowej stronie "; obszar$
450 LET y=yrycerza-1
460 GOSUB 2000
470 PRINT "Po północnej stronie "; obszar$
480 LET dkrolewny=SQR((xrycerza-xkrolewny)^2
+(yrycerza-ykrolewny)^2)
490 LET dsmoka=SQR((xrycerza-xsmoka)^2+
(yrycerza-ysmoka)^2)
500 IF dkrolewny<1 THEN PRINT "Znalazłeś kró-
lewnę. Jesteście uratowani!":END
510 IF dsmoka<1 THEN PRINT "Pochwycił cię sm
ok. Koniec romantycznej przygody!":END
520 PRINT "Królowna znajduje sie w odległosc
i ":INT(10*dkrolewny);" metrów a smok - ";IN
T(10*dsmoka);" metrów."
530 PRINT
540 PRINT " północ- N"
550 PRINT " zachód- W 0 -wschód"
560 PRINT " południe- S"
570 PRINT
580 PRINT "Dokąd idziesz?"
599 REM ***** ruch rycerza *****
600 INPUT odp$
610 LET x=xrycerza
620 LET y=yrycerza
630 IF odp$="N" THEN LET y=y-1
640 IF odp$="S" THEN LET y=y+1
650 IF odp$="W" THEN LET x=x-1
660 IF odp$="O" THEN LET x=x+1
670 GOSUB 2000
680 GOSUB 3000
690 IF ok=0 THEN PRINT "Tam jest "; obszar$:
GOTO 580
700 LET xrycerza=x
710 LET yrycerza=y
799 REM ***** ruch królowny *****
800 LET x=xkrolewny
810 LET y=ykrolewny
820 LET krolewna=INT(RND(1)*4+1)
830 IF krolewna=1 THEN LET x=xkrolewny-0.5
840 IF krolewna=2 THEN LET x=xkrolewny+0.5
850 IF krolewna=3 THEN LET y=ykrolewny+0.5
860 IF krolewna=4 THEN LET y=ykrolewny-0.5
870 GOSUB 2000
880 GOSUB 3000
890 IF ok=0 THEN GOTO 800
899 REM ***** ruch smoka *****
900 LET xkrolewny=x
910 LET ykrolewny=y
950 LET xsmoka=xsmoka-0.5
960 LET ysmoka=ysmoka-0.5
970 IF xrycerza>xsmoka THEN LET xsmoka=
xsmoka+1
980 IF yrycerza>ysmoka THEN LET ysmoka=
ysmoka+1
990 GOTO 300
998 REM ***** PODPROGRAMY *****
999 REM ***** losowanie pozycji *****
1000 LET x=INT(RND(1)*40+1)
1010 LET y=INT(RND(1)*40+1)
1020 RETURN
1999 REM ***** jaki to obszar *****
2000 LET obszar$="las"
2010 IF x<23 THEN LET obszar$="łaka"
2020 IF x=1 OR y=1 OR x=40 OR y=40 THEN LET
obszar$="plaża"
2030 IF y>22 AND y<26 AND x<9 THEN LET
obszar$="skała"
2040 IF y>6 AND y<18 AND x>17 AND x<24 THEN
LET obszar$="staw"
2050 IF (y=12 AND x>8 AND x<18) OR (y=30 AND
x>13 AND x<35) THEN LET obszar$="mur"
2060 IF x<1 OR x>40 OR y<1 OR y>40 THEN LET
obszar$="morze"
2070 RETURN
2999 REM ***** czy jest przejście *****
3000 LET ok=0
3010 IF obszar$="łaka" OR obszar$="las" OR
obszar$="plaża" THEN LET ok=1
3020 RETURN

```

**Losowanie pozycji** (1000—1020) — komputer losuje dwie współrzędne,  $x$  i  $y$ , w zakresie od 1 do 40, a więc pole przez nie określone mieści się na naszej szachownicy. Podprogram ten wykorzystywany jest na początku gry, podczas losowania pozycji początkowych Królewny, Kubusia i Smoka.

**Jaki to obszar** (2000—2070) — na podstawie współrzędnych  $x$  i  $y$  komputer określa co znajduje się w danym miejscu i nazwę tego miejsca umieszcza pod zmienną **obszar\$**. Tutaj jest pole do popisu dla tych, którzy lubią ulepszać programy. Przepisując kilka linii możecie całkiem zmienić krajobraz Krainy Okrutnego Smoka.

**Czy jest przejście** (3000—3020) — na podstawie zmiennej **obszar\$** komputer ustala czy jest to miejsce dostępne (wskaźnik **ok** równy 1) czy niedostępne (**ok**=0) dla Królewny i Kubusia. W naszym przypadku mogą oni chodzić po lesie, łące i plaży.

Program rozpoczynamy od losowego umieszczenia rycerza Kubusia (110—160). Najpierw losowane są współrzędne, następnie komputer sprawdza co znajduje się w danym miejscu, później czy jest to miejsce dostępne (a nie np. środek stawu). Jeśli okaże się, że miejsce jest niedostępne zaczynamy od nowa, w przeciwnym przypadku wylosowane współrzędne stają się pozycją startową Kubusia. Identyczne operacje powtarzają się podczas określania początkowej pozycji królewny (170—220). Co się zaś tyczy smoka, to pomijamy w jego przypadku sprawdzenie na jakim obszarze się znajduje, gdyż jako jegomościowi wyposażonemu w skrzydła jest mu to dokładnie obojętne (230—250).

Teraz komputer przechodzi do wykonywania głównej pętli programu (300—990) a w niej: drukuje komunikat o aktualnej sytuacji, przyjmuje twoją decyzję, przemieszcza królewny i smoka. Najpierw więc komunikat (300—580) — komputer sprawdza gdzie znajduje się w tej chwili Kubuś, bada co jest na sąsiednich polach i wypisuje to na ekranie. Odległość dzieląca naszego bohatera od Smoka i Królewny jest obliczana z twierdzenia Pitagorasa (sądzę, że żadenemu komputerowemu przedszkolakowi nie trzeba przypominać, że „jeśli trójkąt jest prostokątny, to kwadrat długości przeciwprostokątnej jest równy sumie kwadratów długości przyprostokątnych). Przy okazji sprawdzane jest czy nie nastąpiło już spotkanie Kubusia z Królewnią lub paskudnym skrzydłakiem. W obydwu przypadkach gra jest zakończona — choć rzecz jasna z różnym skutkiem.

Sterowanie rycerzem należy do grającego (600—710). Komputer sprawdza jedynie czy nie usiłujesz wprowadzić go np. do morza. Informuje cię wówczas uprzejmie, że tam się nie da iść i prosi o ponowną decyzję.

Ruch Królewny jest całkowicie przypadkowy (800—890). Błąka się biedaczka bez celu. Oczywiście zastrzeżenie dotyczące obszarów zakazanych dotyczy jej również.

Smok porusza się zawsze w kierunku rycerza (900—980). Nie dotyczą go żadne obszary zakazane. Na szczęście olbrzymie cielsko nie pozwala mu poruszać się zbyt szybko i Kubuś może przed nim uciec.

Oto cała mądrość tego programu. Można go było napisać znacznie krócej, lecz nasz sposób ma swoje zalety. Przede wszystkim czytelność i łatwość wprowadzania zmian. A zmieniać można wszystko — nie tylko krajobraz Krainy Okrutnego Smoka. Można uzupełnić informacje drukowane na ekranie, można zmieniać sposób poruszania się postaci, można wreszcie dorobić grafikę (porównaj Skarb Kubusia w numerze 7/86 „Bajtka”). A więc do roboty!

Romek

## MIKRUSEK

Za miesiąc rozwiązanie konkursu „Mikruszek” i lista nagród, dzisiaj natomiast program Szymona Herera (lat 11) z Warszawy. Program wzorowany jest na jeszcze niedawno bardzo popularnych grach telewizyjnych i polega na odbijaniu piłki od ścianki. Pomysł nie jest więc nowy, ale realizacja bardzo ciekawa. Gratuluję!

Romek

```

1 REM *****SCIANA*****
*
2 REM Skok do podprogramu prz
episow.
3 PAPER 7: CLS : GO SUB 4000
5 BORDER 1: PAPER 2: CLS
7 REM Definiowanie piłki
10 FOR n=0 TO 7: READ x: POKE
USR "0"+n,x: NEXT n: DATA BIN 00
011000,BIN 0111110,BIN 1111111
,BIN 10011001,BIN 10011001,BIN 1
111111,BIN 0111110,BIN 0001100
0
15 REM podstawianie zmiennych.
20 LET a$="0": LET b$="1": LET
c$=""
30 LET a=15
40 CLS : LET b=18: LET d=1: LE
T c=10: LET e=31
43 REM Rysowanie ściany
45 FOR n=0 TO 21: PRINT INK 5;
AT n,0;c$: NEXT n
46 REM Rysowanie rakiety
47 PRINT INK 6;AT c,d;a$: PRIN
T INK 4;AT b,e;b$: INK 4;AT b+1,
31;b$
48 REM Uświetlanie ilości doz
wolonych przepuszczeń
50 PRINT AT 0,1;a$ " "
52 IF a=0 THEN FOR m=0 TO 2: F
OR n=50 TO 0 STEP -1: BEEP .009,
n: NEXT n: NEXT m: GO SUB 3000
53 REM *****
55 REM ustalenie kierunku odbi
cia piłki od ściany
56 LET p=RND
60 IF p=1 THEN LET z=.25: GO T
O 70
65 IF p=2 THEN LET z=.75: GO T
O 70
66 LET z=RND
68 REM wycieranie starej piłki
i realizowanie kierunku lotu
70 PRINT AT c,d;" " : IF z>.5 T
HEN LET c=c-1: LET d=d+1
80 IF z<.5 THEN LET c=c+1: LET
d=d+1
90 IF z=.5 THEN LET d=d+1
95 REM Sprawdzenie czy piłka z
ostala odbita czy nie
100 IF d=(e-1) AND (c=b OR c=b+
1) THEN BEEP .1,8: GO TO 1000
110 IF d>30 AND (c<>b OR c<>(b+
1)) THEN BEEP .1,0: LET a=a-1: G
O TO 40
115 REM sprawdzenie czy piłka w
ychodzi za ekran
120 IF c>20 THEN BEEP .1,10: LE
T z=.25: GO TO 70
130 IF c<1 THEN BEEP .1,10: LET
z=.75: GO TO 70
135 REM Rysowanie piłki i skok
do podprogramu ruchów rakiety.
140 PRINT INK 6;AT c,d;a$: GO S
UB 2000: GO TO 70
1000 REM *****
1005 REM Ustalenie kierunku odbi
cia piłki od rakiety i zcieranie
starej piłki
1010 IF c=b THEN LET z=.25
1020 IF c=(b+1) THEN LET z=.75
1030 PRINT AT c,d;" " : IF z<.5 T
HEN LET c=c-1: LET d=d-1
1040 IF z>.5 THEN LET c=c+1: LET
d=d-1
1045 REM Sprawdzenie czy piłka w
ykracza za ekran
1050 IF d<2 THEN BEEP .1,0: GO T
O 60
1060 IF c>20 THEN BEEP .1,10: LE
T p=1: LET z=.25: GO TO 1030
1070 IF c<1 THEN BEEP .1,10: LET
p=2: LET z=.75: GO TO 1030
1075 REM Rysowanie piłki
1080 PRINT INK 6;AT c,d;a$: GO S
UB 2000: GO TO 1030
2000 REM *****
2002 REM Sprawdzenie czy został
naciśnięty klawisz
2005 IF IN 49150=189 OR IN 57342
=190 THEN GO TO 2010
2007 RETURN
2008 REM Sprawdzenie jaki został
naciśnięty klawisz
2010 PRINT AT b,e;" " : AT b+1,e,"
" : IF IN 49150=189 THEN LET b=b
+1
2020 IF IN 57342=190 THEN LET b=
b-1
2035 REM Sprawdzenie czy rakietka
nie wychodzi za ekran
2040 IF b>20 THEN LET b=20
2040 IF b<0 THEN LET b=0
2045 REM Rysowanie nowej rakiety
2050 PRINT INK 4;AT b,e;b$: INK
4;AT b+1,e;b$: RETURN
2999 REM Zakonczenie
3000 PRINT AT 15,16,"KONIEC GRY"
3010 PRINT "I-INSTRUKCJA""S-STA
RT"
3020 IF INKEY$="i" THEN RUN
3030 IF INKEY$="s" THEN RUN 5
3040 GO TO 3000
3999 REM Przepisy
4000 PRINT PAPER 4;"00000000TREN
ING TENISOWY 00000000"
4001 PRINT PAPER 5;"U GRZE GRACZ
ODBIJA PIŁKE W STRONE ŚCIANY.JE
ZELI GRACZ PRZEPUSCI PIETNASCIE
FILEK WTEDY GRA KONCZY SIE."
4010 PRINT PAPER 3;"KLAWISZE:"
4020 PRINT PAPER 2;"GORA:P" PAF
ER 5;"DOL:L"
4030 PRINT PAPER 1; INK 7;AT 21,
0;"NACISNIJ DOWOLNY KLAWISZ."
4040 PAUSE 0: RETURN

```

## BIURO PRZYSZŁOŚCI

*Dokończenie ze str. 32*

gdzie były duże hale maszyn. Dopiero teraz jed-  
nak okazało się, że prawdziwe korzyści wyni-  
kają z kompleksowego podejścia do problemu.  
Główną zaletą automatyzacji prac biurowych  
jest zmiana jakościowa w pracy biura, jakko-  
lwiek pewne korzystne zmiany ilościowe też sa  
nie do pogardzenia. Do automatyzacji zastoso-  
wano nowe podejście, w którym informacje uz-  
nawano za podstawowy surowiec, a technikę  
komputerową — za urządzenia do jego przet-  
warzania.

Skutki takiego podejścia są w różnych dzied-  
zinach odmienne i niekiedy zaskakujące. Jed-  
nym z takich skutków może być wprowadzenie  
nowych metod działania na rynku. Jedną z czo-  
łowych firm ubezpieczeniowych w Wielkiej Bry-  
tanii, Abbey Life dostarczyła swoim agentom  
przenośne mikrokomputery jako pomoc w ich  
kontaktach z klientami. Miało to umożliwić  
agentom dostęp do zbioru informacji przecho-  
wywanych w wielkim komputerze firmy, co z  
kolei ułatwiałoby odpowiedzi na liczne szczegó-  
łowe pytania przyszłego klienta. Agenci, płatni  
od każdej zawartej transakcji, musieli sami ku-  
pić mikrokomputer. Na wydatek ten zdecydowa-  
ło się ponad 25 proc. z nich — eksperyment  
okazał się więc bezdyskusyjnym sukcesem.

Niedawno opracowany raport na temat auto-  
matyzacji w bankach USA stwierdza, że poważ-  
nie myślą one o doprowadzeniu swoich oddzia-  
łów do samowystarczalności w przetwarzaniu  
informacji. To samo opracowanie informuje, że  
w USA około 86 proc. banków zainstalowało  
już w swoich biurach różnorodne systemy au-  
tomatyczne. Analiza przeprowadzona przez fir-  
my komputerowe IBM i Wang, które dostarcza-  
ją 60 proc. urządzeń do przetwarzania tekstów,  
wykazały, że na 1000 zatrudnionych w bankach  
przypada już 26 urządzeń do przetwarzania tek-  
stów. Niewiele mniej było komputerów osobis-  
tych — 21 na 1000 zatrudnionych.

Urządzeniami, które banki najprawdopodob-  
niej wprowadzą wkrótce do swoich biur, będą  
wideo teksty (urządzenia do przesyłania na  
odległość całych stron), końcówki poczty elek-  
tronicznej oraz systemy informacyjne dla kadr  
zarządzających z możliwościami przesyłania  
wykresów.

Dzięki gwałtownemu rozwojowi tendencji do  
automatyzacji prac biurowych tak w USA, jak i  
w Europie Zachodniej, na rynku sprzętu służą-  
cego do tego celu utrzymali się liczni dostaw-  
cy. Przyszłość dla słabszych wytwórców nie  
jest jednak różowa. Podobnie jak z innego typu  
sprzętem komputerowym, największe szanse  
mają silni, jakkolwiek określenie to niekoniecz-  
nie dotyczy tylko IBM czy innych gigantów. Ry-  
nek automatyzacji prac biurowych jest bardziej  
dynamiczny i jeden lub kilka udanych patentów  
mogą zapewnić zaistnienie w nim wytwórcy, o  
którym do niedawna nikt jeszcze nie słyszał.

Różnorodność sprzętu elektronicznego uży-  
wanego do automatyzacji prac biurowych spra-  
wia, że podnoszą się głosy nawołujące do  
stworzenia międzynarodowego standardu  
urządzeń tego typu, co umożliwiłoby organizo-  
wanie współpracy komputerów różnych firm.  
Większość banków amerykańskich używa na  
przykład sprzętu IBM oraz jego tzw. architekту-  
ry systemu sieci. Eksperci nie ryzykują prog-  
noz, kiedy nastąpi opracowanie standardu mię-  
dzynarodowego, który umożliwi łączenie w sie-  
ci elementów różnych systemów, niemniej jest  
to kierunek, w jakim zdąża świat.

Tylko, zapytajmy na koniec, czy rzeczywiście  
zmiana filozofii działania i technicznego wypo-  
sażenia biura, pociągnie również za sobą zmianę  
mentalności pracującego w nim człowieka?

Opr. WS

```

T ***** ATARI *****
90 DIM OBSZAR$(10)
91 DIM ODP$(3)

```

```

***** SPECTRUM *****
820 LET krolewna=INT(RND*4+1)
1000 LET x=INT(RND*40+1)
1010 LET y=INT(RND*40+1)

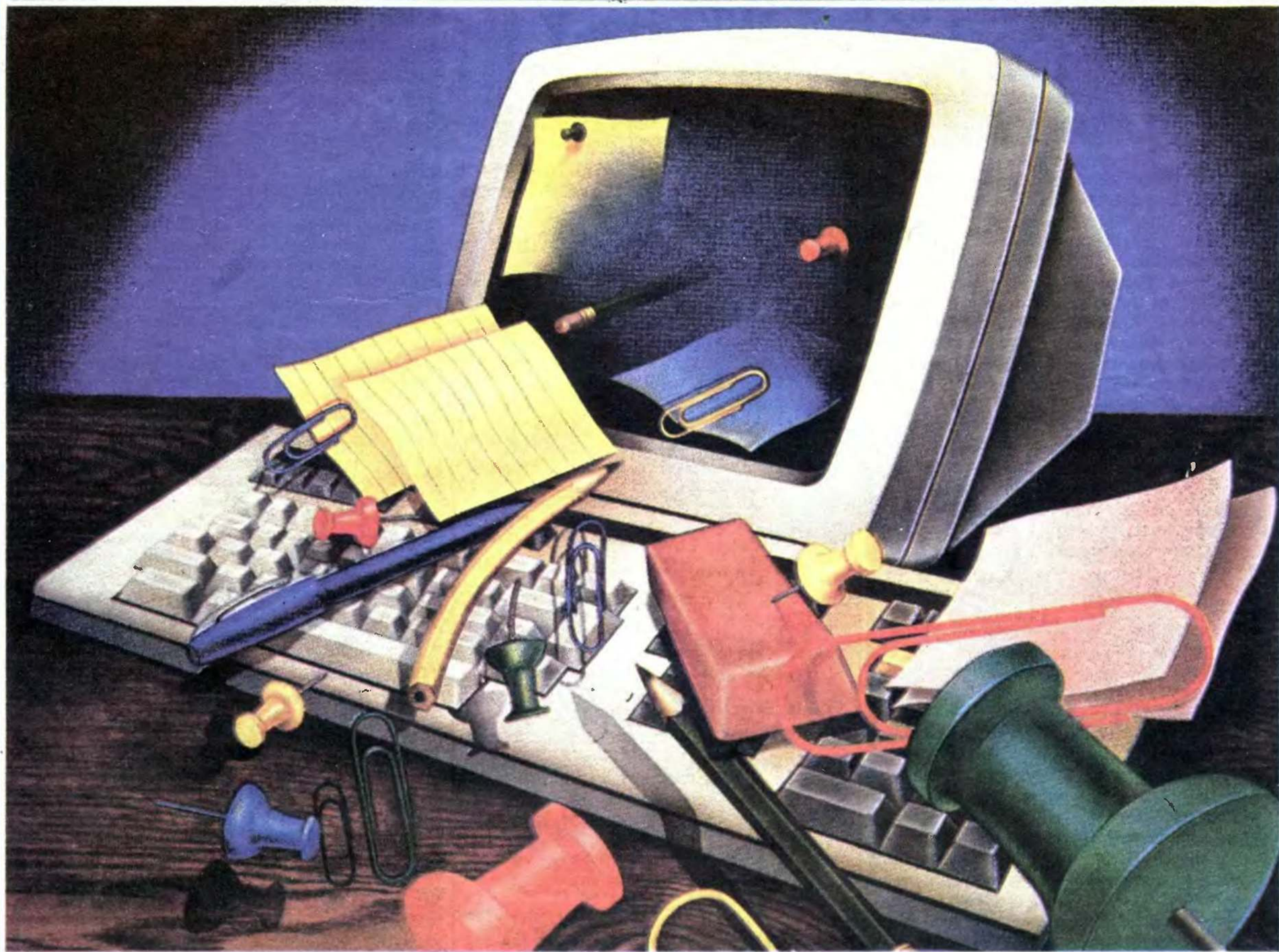
```

```

***** MERITUM *****
820 LET krolewna=INT(RND(0)*4+1)
1000 LET x=INT(RND(0)*40+1)
1010 LET y=INT(RND(0)*40+1)

```

# NIE TYLKO KOMPUTERY



## Inwazja elektroniki na wszelkiego rodzaju działalność biurową przybiera obecnie w najbardziej rozwiniętych krajach rozmiary powodzi.

Setki wytwórców przecigają się w oferowaniu coraz sprawniejszych urządzeń zastępujących tradycyjne atrybuty biura: maszyny do pisania, rozklekotane elektromechaniczne telexy i szafy z aktami. Elektroniczne biuro jest sprawniejsze, tańsze i skuteczniejsze w swoich działaniach.

Na nic chyba nie narzekamy tak często, jak na biurokratów i biurokrację. Urzędnik ukrywający swą niekompetencję za stosami akt, zarządzeń i rozporządzeń, stał się symbolem tego wszystkiego, co hamuje rozwój. Nauka i technika nie znają jeszcze metody, za pomocą której można by przekształcić siedzącego za biurkiem człowieka. Być może uda się to jednak osiągnąć przekształcając najpierw otoczenie, w jakim on działa.

Dziesięć lat temu grupa naukowców z Queen

Mary College z uniwersytetu londyńskiego opracowała raport określający przyszłościową wizję zautomatyzowanego biura. Naukowiec stwierdzili, że w dziesięciolecie 1977—1987 powstaną biurowe systemy informatyczne, które wyeliminują biurka, maszyny do pisania i szafy z aktami. Stało się tak jeszcze szybciej niż przewidywali naukowcy, których w momencie ogłoszenia raportu nazywano marzycielami i fantastami.

Biuurko może być obecnie zastąpione wielofunkcyjnym warsztatem pracy: profesjonalny komputer osobisty połączony z innymi tego typu urządzeniami, przez tanią i wysoce niezawodną sieć lokalną. Warsztat ten może być połączony z głównym komputerem biura, tak, że pracownik może sięgnąć po, lub przetwarzać informacje z głównego zbioru akt przedsiębiorstwa.

Maszyny do pisania mogą być całkowicie zastąpione przez komputery wyposażone w urządzenia do przetwarzania tekstów.

Elektroniczny system akt umożliwi zastąpienie tradycyjnych szaf wypełnionych po brzegi papierami. System ten może być magnetyczny — tak jak urządzenia do przechowywania wielkich ilości akt zbudowane przez IBM, w

których kasety z taśmami są przesuwane przez roboty — lub optyczny, jak maszyna „Megadoc” Philipsa, która jest jakby rozrosniętą wersją cyfrowej maszyny do odtwarzania compact disców, tj. cyfrowych longplayów.

Obfitość systemów elektronicznych oferowanych do zastosowań biurowych spotyka się ciągle z oporami części dyrektorów odpowiadających za prace swoich urzędów. W pewnym stopniu ich rezerwa jest wyłumaczalna. We wczesnych fazach prób zautomatyzowania urzędów popełniono sporo błędów. Entuzjaści, którzy wtedy uwierzyli w zapewnienia producentów elektronicznego sprzętu biurowego, sparzyli się na zapowiedziach znacznego zwiększenia wydajności pracy, a więc obniżenia kosztów papierkowych operacji. Okazało się jednak wkrótce, że wyliczenia i prognozy były błędne.

Biura zaczęto automatyzować od najniższych szczebli personelu urzędniczego i maszynistek, dopiero później zaczęto obejmować tym procesem szczeble pośrednie i zarządzające. Podejście to przyniosło początkowo pewne korzyści, szczególnie w tych organizacjach,

*Dokończenie na str. 31*

# BIURO PRZYSZŁOŚCI