

3-4

Z MIKROKOMPUTEREM NA TY

# Bajtek

MIESIĘCZNY DODATEK DO SZTANDARU MŁODYCH

NR 3-4 KWIECIEŃ 1990 CENA 5000 ZŁ

# CEBIT'90

# 26

## JOYSTICKÓW

## ŚWIAT MAKROASSEMBLERÓW

## TEST SPECTRAVIDEO

## CHIPY W PRZE- STWO- RZACH



KONKURS! KONKURS! KONKURS! KONKURS!

ROBOCOP



Z HANNOWERU

# 40 tysięcy komputerów

Czym dla filmowego świata jest doroczne rozdanie Oscarów w Los Angeles czy festiwal filmowy w Cannes, tym dla specjalistów w branży komputerowej są organizowane w Hannoverze targi „CEBIT”.

„CEBIT” anno domini 1990 to 4 tysiące wystawców z całego świata, ponad 560 tysięcy zwiedzających, setki akredytowanych dziennikarzy i prezentacja tego, co w przemyśle komputerowym jest NAJWIĘKSZE, NAJNOWSZE i NAJLEPSZE. To okazja do zaprezentowania najnowocześniejszych konstrukcji i przekonania klientów, że warto kupić właśnie TEN towar.

„CEBIT'90”, który odbył się w dniach 21–28 marca, stał pod hasłem „Business USA”. Mimo ostrej konkurencji nikt nie kwestionuje prymatu Stanów Zjednoczonych w wyznaczaniu kierunków rozwoju najnowszych technologii. Jednak ojczyzna mikrokomputerów poczuła się zagrożona tworzącym się na jej oczach potężnym rynkiem Zjednoczonej Europy. Dlatego też małe i średnie firmy amerykańskie wsparte dotacjami rządowymi wykupiły kilka hektarów powierzchni wystawowej i zorganizowały show, jakiego Hannover dawno nie widział.

I nie ma tu znaczenia fakt ustawienia w hali numer 6 małej fontanny, czy wspaniałego barku obsługującego gości. Ważne jest to, że poza ciekawą ofertą sprzętową i programową zadbano o szeroką informację. Ekspozycji towarzyszyły seminaria i wykłady na temat możliwości amerykańskiego przemysłu. Do współpracy zachęcał osobiście sekretarz do spraw handlu USA Robert Mosbacher, który, jak powiedział — przyjechał, aby popierać amerykański biznes. Była to zapowiedź szturmowania na jednoczącą się europejską fortecę.

Akcenty europejskie zdominowały w Hannoverze wszystko. Liczne przedstawicielstwa banków działające na terenach targowych podają już aktualny kurs waluty Zjednoczonej Europy — ECU. Nad budynkiem zarządu targów powiewała niebieska flaga z jedenastoma gwiazdami, a większość moich rozmówców czuła się już obywatelami państwa, które stary kontynent ostatni raz oglądał w czasach Cesarstwa Rzymskiego.

Dla mieszkańca Europy Wschodniej przykrym widokiem były mapy tego nowego świata — wszystkie kończyły się na Łabie. Na przykład Polska oznaczona jest jako biała plama.

Dystans widoczny był także na salach wystawowych. Ogromne hale o rozmiarach boisk piłkarskich wypełnione były po brzegi nowoczesnym sprzętem komputerowym. Królował nowy standard Intel — mikroprocesor oznaczony symbolem i486. Zapowiedziano nową konstrukcję — 64 bitowy mikroprocesor oznaczony symbolem 860.



Jonathan Summerton, szef działu reklamy i marketingu firmy ABC Data: ... interesy idą dobrze.

O ile plany konstruktorów się spełnią, to za kilka lat będziemy dysponowali komputerami osobistymi o mocach obliczeniowych dzisiejszych konstrukcji Seymoura CRAY'A. Zapowiedzią skoku w XXI wiek była informacja o prowadzonych przez koncern Motorola wspólnie z Departamentem Obrony USA pracach nad „SuperChipem” — mikroprocesorem o mocy obliczeniowej 200 milionów operacji na sekundę!

Jednocześnie „CEBIT'90” po raz kolejny udowodnił starą prawdę, że produkcja oprogramowania nie nadąza za rozwojem sprzętu. Firmy Novell, Microsoft, Lotus czy Borland dopiero na przełomie 1989/90 przedstawiły nowe programy w pełni wykorzystujące możliwości 32 bitowych mikroprocesorów. Lista oprogramowania specjalnie dla nowej generacji komputerów jest jednak bardzo skromna. Standard IBM PC trzyma się mocno.

Z ciekawostek technicznych warto wspomnieć nowy kolorowy ekran ciekłokrystaliczny o wspaniałej rozdzielczości zaprezentowany przez firmę Toshiba. Hewlett-Packard zaprezentował nową drukarkę typu Ink Jet o bardzo niskiej cenie. Dał się dostrzec znaczny spadek cen drukarek laserowych.

Małe firmy amerykańskie zajmujące się najnowszymi technologiami oferowały rozmaite rozszerzenia sprzętowe i ciekawe oprogramowanie od specjalizowanych kart graficznych (Sota System) do bardzo szybkiego kompilatora języka Pascal (Quick Byte). Zainteresowanie budziła firma Versitron oferująca sieci komputerowe zbudowane w oparciu o światłowody. Jej dyrektor do spraw handlowych Gary Sibiski był zdziwiony, gdy opowiedziałem mu o podobnej konstrukcji powstałej kilka lat temu na Uniwersytecie Mikołaja Kopernika w Toruniu. Pracą zespołu informatyków kierował dr Bronisław Żurawski. Niestety, to nie ci ludzie reprezentowali nasz kraj w Hannoverze. Stoisko Metronexu świadczyło o tym, że jeszcze Polska nie zginęła, choć niewie-

le brakuje. Drukarki Mery Błonie aż proszą się o nowe obudowy, a poza tym nic się od roku ubiegłego nie zmieniło. Może „Dżon Wenczer” pomoże?

Polacy mają dobrą opinię wśród specjalistów. Wielu gospodarzy stoisk nosiło plakietki ze swojsko brzmiącymi nazwiskami. Często w rozmowach pytałem, dlaczego rząd polski nie interesuje się komputeryzacją do tego stopnia, że nawet GUS nie podaje, ile w Polsce jest komputerów. Macie tak wielu dobrych fachowców — mówiono.

Odpowiadałem, że gdy rząd w przeszłości brał się za rozwiązywanie takich czy innych problemów — kończyło się to źle. Miejmy nadzieję, że i tym razem Opatrzność nas ustrzeże. Ludzie sami mogą rozwiązać swoje problemy, o ile nikt im nie będzie przeszkadzał. Na szczęście w Polsce coraz powszechniejsze staje się przekonanie, że bez komputerów nie da się gonić coraz szybciej oddalającego się kontynentu. Stąd tak ogromny popyt na wszystko, co związane jest z informatyką, popyt, który nie maleje i budzi lekkie zdziwienie u Europejczyków.

Wysłannicy „Bajtka” znaleźli się w Hannoverze na zaproszenie firmy ABC DATA z Bonn. Ten znany na polskim rynku dostawca drukarek STAR znacznie rozszerzył zasięg swego działania i firma jest z zyczliwą uwagą obserwowana przez zachodnich finansistów. Dyrektor Lech Matusiak z optymizmem patrzy w przyszłość. Jego specjalista od marketingu i reklamy Jonathan Summerton dodaje, że interesy idą dobrze. Zapotrzebowanie na drukarki komputerowe w Polsce nie maleje. I choć z pewnością komputerowy boom sprzed kilku lat się skończył, to nasz kraj stara się dotrzymać kroku innym. Dobry to prognostyk na przyszłość.

„CEBIT'91” będzie jeszcze bardziej okazały, kolorowy i europejski. A jacy będziemy my?

Marek Czarkowski



# Bajtek

**„BAJTEK” — MIESIĘCZNY DODATEK DO „SZTANDARU MŁODYCH”**

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

**ZESPÓŁ REDAKCYJNY:**

Waldemar Siwiński — kierownik zespołu „Bajtka”  
Marek Czarkowski — sekretarz redakcji „Bajtka”  
Roman Poznański — kierownik działu klanów

Wanda Roszkowska — opr. graficzne

**KLANY REDAGUJĄ:**

Atari — Wojciech Zientara  
Amstrad — Jonasz Mayer  
Commodore — Klaudiusz Dybowski, Dominik Falkowski  
Spectrum — Marian Przasnyski

**STALE WSPÓŁPRACUJĄ:**  
Andrzej Pilaszek, Janusz Jarmoch, Marcin Borkowski, Łukasz Czekański, Waldemar Nowak, Michał Sobieszuk, Maciej Pietras, Marcin Bojko, Jacek Kunowski, Michał Karkucinski

Zdjęcia w numerze Leopold Dzikowski

Montaż offsetowy — Grażyna Ostaszewska,  
Korekta — Maria Krajewska, Zofia Woltńska

Spectrum — Marcin Przasnyski  
szuk, Maciej Pietras, Marcin Bojko,  
Fotoskład — Grażyna Kurzątkowska,  
Korekta — Maria Krajewska, Teresa Rutkowska

WYDAWCA: Oficyna Wydawnicza „Sztandar Młodych”  
Warszawa, ul. Wspólna 61

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

Nr zlecenia 17820 n. 1500 000 egz.



**Wielki konkurs na scenariusz gry komputerowej**

Firma „Karen” razem z redakcjami „Komputera”, „5-10-15”, telewizyjnego programu „Spectrum” i „Bajtkiem” organizuje konkurs dla dzieci na scenariusz gry komputerowej. Jest to wspaniała okazja do sprawdzenia wyobraźni i

co najważniejsze nie trzeba mieć komputera aby wziąć w nim udział. Dla zwycięzców firma „Karen” przygotowała atrakcyjne nagrody. Szczegóły w następnym numerze „Bajtka”.

**KONKURS! KONKURS! KONKURS! KONKURS!**



# 26 JOYSTICKÓW

## W TEŚCIE NON-STOP

18 T-Stick 1



9 Quickshot II Plus



3 Joystick 200-3-Way



23 Commando 4



7 Quickshot I



W oczekiwaniu na wyniki naszego turnieju joysticków przedstawiamy efekty sadystycznych zmagania kolegów z zach.-niemieckiego czasopisma **HAPPY COMPUTER**. Jednocześnie przypominamy o trwającym nadal **WIELKIM TURNIEJU** na najlepszego, najsprawniejszego, najdoskonalszego, jednym słowem **TOTALNY JOYSTICK ROKU** produkcji krajowej.

(Redakcja)

26 spośród najpopularniejszych na zachodnio-niemieckim rynku joysticków zostało poddanych ponad 300 laboratoryjnym próbom, które miały na celu wykazać ich precyzję działania, wykonania, wytrzymałość i wygodę obsługi. Sprawdzano ich odporność na zniszczenie oraz ich ergonomię. Starano się odpowiedzieć na pytanie, który joystick jest najlepszy oraz od jakich elementów zależy jego popularność.

**Zawody odbywały się w trzech kategoriach.**

1. zawartość-przełączniki i ich rodzaje, druty (pomiar długości), inne elementy.
2. zachowanie podczas pracy oraz trwałość — który mikroprzełącznik najlepiej pracuje przy silnym obciążeniu oraz wytrzymałość obudowy na upadek ze stołu.
3. co dany joystick może naprawę dokonać — wielogodzinne testy na grach sportowych typu SUMMER GAMES, nie kończące się strzelaniny.

### JOYSTICK — CO TO JEST

Zwykle jest to prosta konstrukcja składająca się z pałki (stick) i przyjemności (joy) oraz obudowy, czterech styków ułożonych na krzyż, jednego, dwóch lub więcej przycisków strzału (fire), kolorowych drutów i kabla łączącego z komputerem. Z zasady większość joysticków jest do siebie podobna i różni się jedynie kształtem obudowy, położeniem przycisków FIRE, rodzajem przełączników i długością kabli.

### ZAWARTOŚĆ

Do wyboru mamy cztery rodzaje styków:

1. **mikro przełączniki** — precyzyjne, długowieczne, dobrze kontaktujące, pozwalają łatwo wy відчуć siłę nacisku, najlepsze z opisanych

5 Micro Plus



16 Quick Gun Turbo 2 Super



4 Quick Gun Turbo 3



8 Quickshot II



10 Quickshot II Turbo







**20 Match-Pro**

**2. styki** — dwie równoległe blaszki, które pod wpływem nacisku łączą się zamykając obwód, łatwo ulegają zniszczeniu przy długotrwałym użytkowaniu, odkształcają się trwale przy silnym nacisku, ich zaletą jest to, że nie musi się grać tak bardzo siłowo jak w poprzednim przypadku (drugie miejsce)

**3. blachy** — popularne, wymagają mało siły, przez co nie wyczuwa się ich kontaktowania, krótko żyją, blaszki szybko śnieżnią i pękają

**4. gumy** — guma przewodząca, ciekawe rozwiązanie, ale rzadko spotykane

**Drażki** bywają bardzo różnych kształtów, ale dają się podzielić na dwie zasadnicze grupy:

1. mieszczące się całe w dłoni
2. trzymane w dwóch, trzech palcach

**Drażki** pierwszej grupy są lepsze, jeżeli są ergonomicznie wykonane tzn. „pasują do dłoni”. Niektóre z joysticków tej grupy (Joy-200-3-Way, Quick Gun Turbo 3) mają dołączone do kompletu po trzy różne główki. Nie jest to jednak dobre rozwiązanie, bowiem te ostatnie nie pozwalają się zamontować stabilnie i podczas gry kręcą się na wszystkie strony.

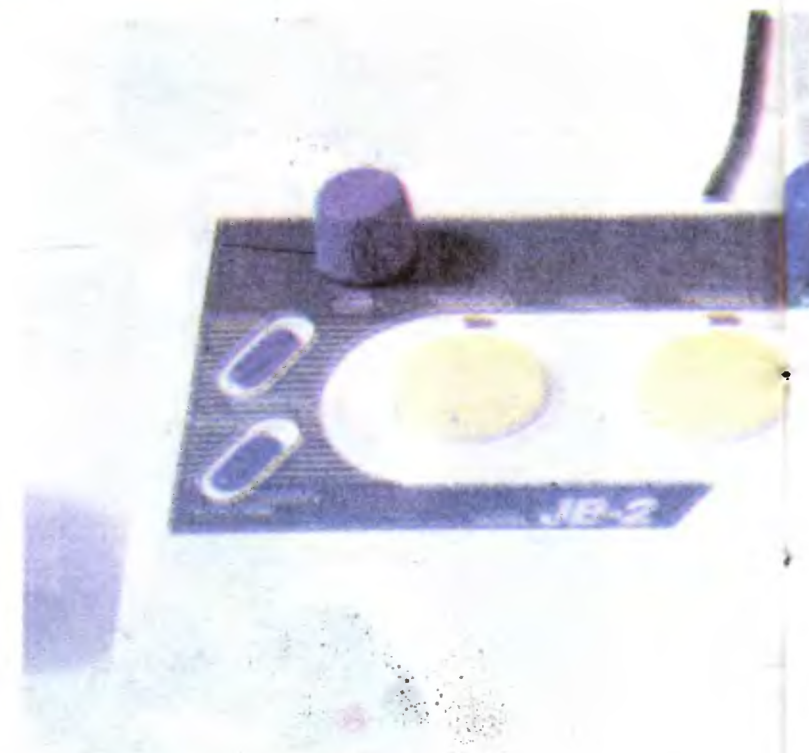
Trzymane w palcach mają swoje plusy i minusy. Dłoń można wprawdzie wygodnie położyć na stole i unikać zmęczenia mięśni, ale potrzebują więcej miejsca do obsługi. Przedstawicielami tej grupy są Multi-Funktion 1001 oraz Multi-Funktion 2002.

Joysticki posiadające kanciastą obudowę (Joy-Board1 i Joy-Board2) są wyjątkowo nieprzyjemne dla dłoni przy długotrwałym użytkowaniu.

**Długość kabla** wpływa nie tylko na wygodę w manewrowaniu joystickiem wokół komputera, ale,



**21 Cruiser**



Joystick numer:	1	2	3	4	5	6
<b>NAZWA FIRMA</b>	MULTI-FUNKTION 2002 KAUFHOF	MULTI-FUNKTION 1001 KAUFHOF	JOYSTICK-200-3-WAY KAUFHOF	QUICK GUN TURBO 3 A-Z ELEKTRONIK	MICRO PLUS KAUFHOF	SWITCH JOY MÜKRA
<b>KONSTRUKCJA:</b>						
<b>PRZEŁĄCZNIKI DRAŻKA</b>	super	super	super	super	super	super
<b>PRZEŁĄCZNIKI FIRE</b>	super	super	super	super	super	super
<b>KSZTAŁT DRAŻKA</b>	b. dobry	b. dobry	dostateczny	dostateczny	b. dobry	b. dobry
<b>AUTO-FIRE</b>	super	b. dobry	dobry	dobry	b. dobry	b. dobry
<b>KABEL</b>	dobry	dobry	dobry	dobry	dobry	dobry
<b>DODATKI</b>	b. dobre	słabe	dostateczne	dostateczne	słabe	słabe
<b>PODSUMOWANIE</b>	super	b. dobry	b. dobry	b. dobry	b. dobry	b. dobry
<b>FUNKCJE:</b>						
<b>CZUŁOŚĆ DRAŻKA</b>	b. dobra (300 g)	b. dobra (200 g)	słaba (720 g)	słaba (600 g)	dobra (340 g)	dobra (360 g)
<b>CZUŁOŚĆ FIRE</b>	b. dobra (200 g)	b. dobra (240 g)	słaba (540 g)	słaba (570 g)	dobra (340 g)	dobra (340 g)
<b>PRZYSSANIE DO OKNA</b>	super	dobre (19 min)	słabe (5 min)	dostateczne (6,34 min)	dobre (17,38 min)	dobre (18,56 min)
<b>SIŁA ODRYWAJĄCA (W PIONIE)</b>	dobry (11 kg)	super (16 kg)	dobry (10 kg)	dobry (10 kg)	dobry (12 kg)	b. dobry (13 kg)
<b>SIŁA ODRYWAJĄCA (W POZYMIE)</b>	dobry (6 kg)	super (10 kg)	słaby (3 kg)	słaby (3 kg)	dostateczny (5 kg)	dostateczny (3 kg)
<b>WYTRZYMAŁOŚĆ OBUDOWY</b>	b. dobra	b. dobra	b. dobra	b. dobra	beznadziejna	słaba
<b>PODSUMOWANIE</b>	b. dobry	super	dostateczny	dostateczny	dobry	dobry
<b>TEST PRAKTYCZNY:</b>						
<b>GRY SPORTOWE</b>	b. dobry	b. dobry	dostateczny	dostateczny	dobry	dobry
<b>STRZELANINY</b>	super	super	dobry	dobry	b. dobry	b. dobry
<b>GEOS</b>	dobry (27 s)	dobry (28 s)	dostateczny (34 s)	dostateczny (32 s)	dobry (26 s)	dobry (28 s)
<b>PODSUMOWANIE</b>	b. dobry	b. dobry	dobry	dobry	b. dobry	b. dobry
<b>CENA W MARKACH RFN</b>	40 DM	30 DM	25 DM	25 DM	20 DM	30 DM
<b>STOSUNEK CENY DO WARTOŚCI UŻYTKOWEJ</b>	dobry	b. dobry	słaby	słaby	dobry	dostateczny
<b>OCENA OSTATECZNA</b>	b. dobry (1)	super (0,75)	dobry (2,5)	dobry (2,5)	b. dobry (1,5)	b. dobry (1,75)
Joystick numer:	14	15	16	17	18	19
<b>NAZWA FIRMA</b>	TURBO S A-Z ELEKTRONIK	QUICK GUN TURBO 1 A-Z ELEKTRONIK	QUICK GUN TURBO 2 SUPER A-Z ELEKTRONIK	QUICK GUN TURBO COBRA A-Z ELEKTRONIK	T-STICK 1 MÜKRA	SPEEDKING KONIX
<b>KONSTRUKCJA:</b>						
<b>PRZEŁĄCZNIKI DRAŻKA</b>	dostateczne	dostateczne	super	super	super	super
<b>PRZEŁĄCZNIKI FIRE</b>	dostateczne	super	super	super	super	b. dobre
<b>KSZTAŁT DRAŻKA</b>	dobry	super	super	dobry	dobry	b. dobry
<b>AUTO-FIRE</b>	słaby	dostateczny	b. dobry	dobry	b. dobry	słaby
<b>KABEL</b>	dobry	dobry	dobry	dobry	dobry	super
<b>DODATKI</b>	słabe	słabe	dobre	słabe	słabe	słabe
<b>PODSUMOWANIE</b>	dostateczny	dobry	super	b. dobry	b. dobry	b. dobry
<b>FUNKCJE:</b>						
<b>CZUŁOŚĆ DRAŻKA</b>	słaba (250 g)	b. dobra (180 g)	słaba (620 g)	super (100 g)	słaba (740 g)	b. dobra (180 g)
<b>CZUŁOŚĆ FIRE</b>	słaba (600 g)	słaba (700 g)	b. dobra (300 g)	słaba (580 g)	b. dobra (160 g)	dobra (320 g)
<b>PRZYSSANIE DO OKNA</b>	super (-)	super (-)	dostateczne (16 min)	brak (-)	super (-)	brak (-)
<b>SIŁA ODRYWAJĄCA (W PIONIE)</b>	dostateczny (8 kg)	dobry (11 kg)	dobry (10 kg)	dostateczny	słaby (6 kg)	b. dobry (-)
<b>SIŁA ODRYWAJĄCA (W POZYMIE)</b>	dostateczny (4 kg)	dostateczny (4 kg)	słaby (3 kg)	dostateczny	słaby (3 kg)	b. dobry (-)
<b>WYTRZYMAŁOŚĆ OBUDOWY</b>	super	beznadziejna	dobra	super	super	beznadziejna
<b>PODSUMOWANIE</b>	dobry	dobry	dobry	dobry	dobry	dobry
<b>TEST PRAKTYCZNY:</b>						
<b>GRY SPORTOWE</b>	dobry	dobry	b. dobry	słaby	dobry	dostateczny
<b>STRZELANINY</b>	dobry	dobry	b. dobry	dobry	b. dobry	dobry
<b>GEOS</b>	dobry (30 s)	dobry (29 s)	dostateczny (31 s)	dobry (29 s)	dobry (29 s)	dobry (28 s)
<b>PODSUMOWANIE</b>	dobry	dobry	b. dobry	dobry	b. dobry	dobry
<b>CENA W MARKACH RFN</b>	30 DM	14 DM	25 DM	30 DM	25 DM	25 DM
<b>STOSUNEK CENY DO WARTOŚCI UŻYTKOWEJ</b>	beznadziejny	b. dobry	dobry	słaby	dobry	dostateczny
<b>OCENA OSTATECZNA</b>	dostateczny (3,25)	b. dobry (1,75)	b. dobry (1,25)	dobry (2,25)	b. dobry (1,75)	dobry (2)





co ważniejsze, na odległość w jakiej siedzimy od monitora. Absolutnym minimum jest 1,2 m. Pod tym względem dwumetrowy kabel Joy-Boardów czyni je absolutnymi liderami w tej klasie.

**AUTOFIRE** był testowany na grze URIDIUM i tu okazało się, że niektóre z joysticków nie pracują prawidłowo (omijanie niektórych przeszkód).

Przycisk AUTOFIRE powinien znajdować się w zasięgu palców i jego włączenie nie powinno odrywać naszej uwagi od ekranu. I tu okazało się, że Joy-Board2 jest najlepszy. Niektóre modele wyposażone są dodatkowo w zmienną częstotliwość ognia (Multi-Funktion 2002, Joy-Board2) co czasami jest przydatne. Niestety, zaledwie połowa z testowanych manipulatorów posiadała zwykły AUTOFIRE.

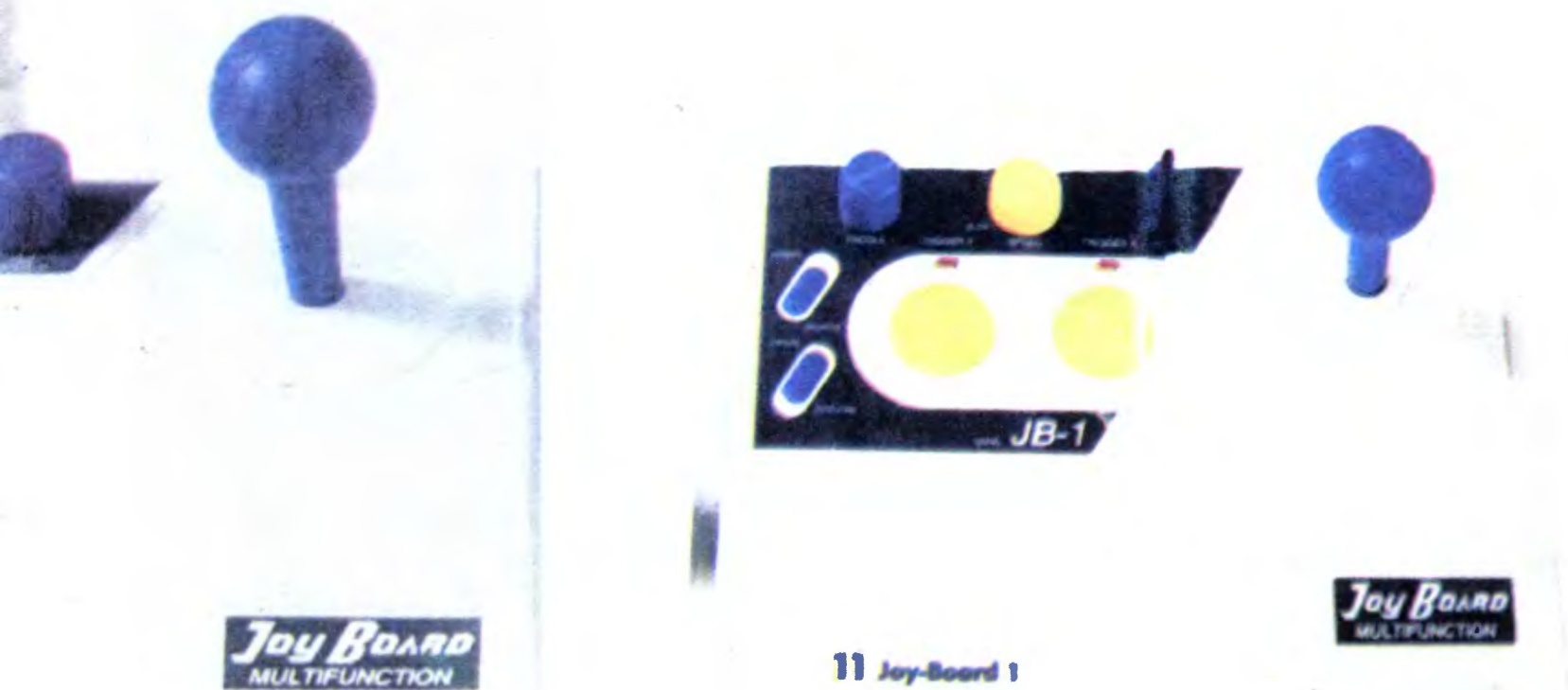
**Dodatki** bywają bardzo różne: począwszy od wymyślnego kształtu obudowy, przez kolorowe lampki, zegarki, czytniki ciekłokrystaliczne, na bardzo kolorowych kartonach skończywszy. Przydatność ich jest zwykle znikoma, a służą głównie nabijaniu ceny.

## MOŻLIWOŚCI (KRYTERIA OCENY)

Wysoko punktowano te, które **lecko działają** i są **precyzyjne**. Twarde zużywające dużo siły nie zapewniają wystarczającego komfortu gry. Utrudnione poruszanie świadczy o tym, że przełączniki, mechanika albo obudowa zostały źle wykonane. W skrajnych przypadkach (Quick-Gun Turbo Cobra i Cruiser), różnica siły uciążu dla uzyskania takiego samego efektu wynosiła 1000 gram. Najkorzystniej w tej dyscyplinie wypadły: stykowy Quickshot i blaszkowy Junior oraz Quick Gun Turbo1. Zupełnie wyjątkowo prezentuje się na tym tle Turbo S.

**Stopy** (ssawki) powinny silnie przylegać do stołu, aby zwolnić drugą rękę, która może w tym czasie obsługiwać klawiaturę. Joysticki posiadające tylko nóżki gumowe (Competition Pro i Competition Pro Extra) są zaledwie zadowolające.

## 1 Multi-Funktion 2002



## 12 Joy-Board 2

## 11 Joy-Board 1

7	8	9	10	11	12	13
QUICKSHOT I SPECTRAVIDEO	QUICKSHOT II SPECTRAVIDEO	QUICKSHOT II PLUS SPECTRAVIDEO	QUICKSHOT II TURBO SPECTRAVIDEO	JOY — BOARD 1 MÜKRA	JOY-BOARD 2 MÜKRA	TURBO JUNIOR A-Z ELEKTRONIK
b. dobre b. dobre b. dobry słaby dobry słabe dobry	b. dobre b. dobre super b. dobry dostateczny słabe b. dobry	super super super b. dobry dobry słabe b. dobry	super super super dobry dobry słabe b. dobry	super super super super super b. dobre super	super super super super super dobre super	dostateczne dostateczne dobry słaby dobry słabe dobry
b. dobra (180 g) super (40 g) super (-) dobry (11 kg) dobry (7 kg) super super	dobra (380 g) słaba (780 g) super (-) b. dobry (13 kg) dobry (6 kg) super b. dobry	dobra (440 g) słaba (780 g) super (-) dobry (12 kg) dobry (7 kg) super b. dobry	b. dobra (280 g) b. dobra (300 g) super (-) b. dobry (13 kg) dostateczny (5 kg) super b. dobry	dobra (340 g) b. dobra (240 g) dostateczne (8,22 min) b. dobry (14 kg) super (10 kg) b. dobra b. dobry	b. dobra (180 g) b. dobra (260 g) dostateczne (6,33 min) dobry (11 kg) dostateczny (5 kg) dostateczna dobry	b. dobra (180 g) dostateczna (500 g) super (-) dobry (11 kg) słaby (3 kg) beznadziejna dobry
b. dobry dobry dobry (28 s) b. dobry 9,95 DM super super (0,75)	b. dobry dobry dobry (30 s) b. dobry 13 DM super super (0,75)	b. dobry super dostateczny (32 s) b. dobry 15 DM super super (0,75)	b. dobry b. dobry dobry (32 s) b. dobry 20 DM super super (0,75)	b. dobry b. dobry dobry (30 s) b. dobry 24 DM super super (0,5)	b. dobry super dostateczny b. dobry 28 DM b. dobry b. dobry (1)	dobry dobry dobry (30 s) dobry 9,95 DM super b. dobry (1,5)
20	21	22	23	24	25	26
MATCH PRO MÜKRA	CRUISER MÜKRA	COMMODORE 1352 COMMODORE	COMMANDO 4 LINDY	COMPETITION PRO DYNAMICS	COMPETITION PRO EXTRA DYNAMICS	ICON TROLLER SUNCOM
super super super b. dobry dobry słabe b. dobry	super super dobry słaby b. dobry słabe b. dobry	dobre dobre dobry słaby b. dobry słabe dobry	dostateczne dostateczne b. dobry b. dobry dobry słabe dobry	super b. dobre b. dobry słaby b. dobry słabe b. dobry	super super b. dobry b. dobry dobry super	dobre dobre dobry słaby nie istotny słabe dobry
dobra (300 g) słaba (760 g) super (-) dobry (10 kg) dostateczny (4 kg) super b. dobry	słaba (1100 g) b. dobra (180 g) dostateczne (7,28 min) dobry (10 kg) dobry (6 kg) dobra dobry	b. dobra (280 g) dobra (360 g) brak (-) dobry (13 kg) słaby (-) super b. dobry	dobra (360 g) słaba (680 g) super (-) dobry (12 kg) b. dobry (8 kg) super b. dobry	dobra (380 g) super (120 g) brak (-) słaby (-) słaby (-) super dobry	dobra (400 g) b. dobra (160 g) brak (-) słaby (-) słaby (-) dobra dobry	b. dobra (160 g) b. dobra (220 g) brak (-) super (mocowany do komputera) super (mocowany do komputera) nie istotna super
b. dobry super dobry (30 s) b. dobry 25 DM b. dobry b. dobry (1)	dostateczny b. dobry dostateczny (31 s) dobry 20 DM dobry b. dobry (1,75)	dobry dobry dostateczny (32 s) dobry 15 DM b. dobry b. dobry (1,5)	b. dobry dobry dobry (28 s) b. dobry 30 DM b. dobry b. dobry (1,25)	dobry b. dobry dobry (28 s) b. dobry 24 DM dobry b. dobry (1,5)	dobry super b. dobry (24 s) b. dobry 39 DM słaby b. dobry (1,75)	dostateczny dobry dobry 39 DM nieuzasadniony dobry (2,25)





14 Turbo S



19 Speedking



15 Quick Gun Turbo 1



25 Competition Pro Extra



6 Switch Joy



2 Multi-Funktion 1001



17 Quick Gun Turbo Cobra



24 Competition Pro



13 Turbo Junior



22 Commodore 1352

Siła przyssania była badana na dwa sposoby.  
 1. na czystej szybie (próba okna) ustawionej w pionie, gdzie wynik 38 min był uznawany za bardzo dobry  
 2. sprawdzano również odporność na odrywanie od blatu stołu po przyłożeniu siły w poziomie i w pionie (odrywanie do góry)

Interesującym wydaje się być fakt, że bardzo dobre wyniki pierwszego testu nie powtarzały się w drugim (Quickshot 1, Quick Gun Turbo 1). Jednocześnie najlepiej wypadł w tej próbie Multi-Funktion 1001.

Niektóre Joysticki (Speedking i Quick Gun Turbo Cobra) wymagają stałego trzymania w ręku. Z uwagi na budowę punktowano jedynie ich ergonomie (wygoda trzymania, łatwość operowania przyciskiem FIRE). W tym miejscu pojawił się problem „krótkich palców” przy posługiwaniu się Speedkingiem.

**Odporność na transport.** Często się zdarza, że jadąc do kolegi zabieramy swojego ulubionego pupila i ten, podróżując z nami w zatłoczonych autobusach razem z książkami w teczce, dociera na miejsce w stanie rozkładu. Pierwsza próba polegała na trzykrotnym zrzuconiu 1,2 kilogramowej książki z wysokości 1,5 m pod działaniem siły grawitacyjnej na wierzchołek i na bok. Metalowe obudowy w Multi-Funktion 1001 i Multi-Funktion 2002 gnioty się, ale nie pękały. U Quick Gun Turbo 2 Super i Joystick-200-3-Way ulegał uszkodzeniu (wylatywał na zewnątrz) FIRE (to się daje naprawić). Joy-Board-2, Cruiser i Competition Pro Extra myliły się. W Micro Plus, Turbo Junior i Quick Gun Turbo 1 uszkodzeniu ulegał drążek sterowy (pękały styki). W przypadku Speedkinga wszystkie części rozsypały się po pokoju. W czasie testów na grach sportowych próbowano sprawdzić odporność na bardzo szybkie machanie „lewo-prawo”, szybkość reakcji na gwałtowną zmianę kierunku oraz skuteczność działania AUTOFIRE.

### PODSUMOWANIE

**Ku ogólnemu zaskoczeniu nie wyróżniono jednego zwycięzcy dla wszystkich kategorii.** Oczywiście jest, że droższe urządzenia są trwalsze (jakość plastiku, kabli, styków) i mają większe możliwości. I tu pojawia się ostatnia pozakonkursowa dyscyplina, czyli cena. Wszystko zależy od tego, czego Państwo oczekujecie, od celu jakimu ma joystick służyć (np. Icontroller jest wyspecjalizowanym joystickiem do obsługi programów używających ikon i daje się połączyć z obudową komputera i, co najważniejsze, od zasobności kieszeni. Wtedy nawet na Quickshota 1 zaczynamy patrzeć przychylnie, bowiem kosztuje zaledwie 9,90 DM. Tak więc wybór należy do Was.

Na podstawie HAPPY COMPUTER opracował Waldemar Nowak



26 IconTroller



# JĘZYKI ATARI XL/XE

Biblioteka książek przeznaczonych dla użytkowników Atari XL/XE wzbogaciła się o kolejną, wartościową pozycję. Jest nią książka Wojciecha Zientary „Języki Atari XL/XE”.

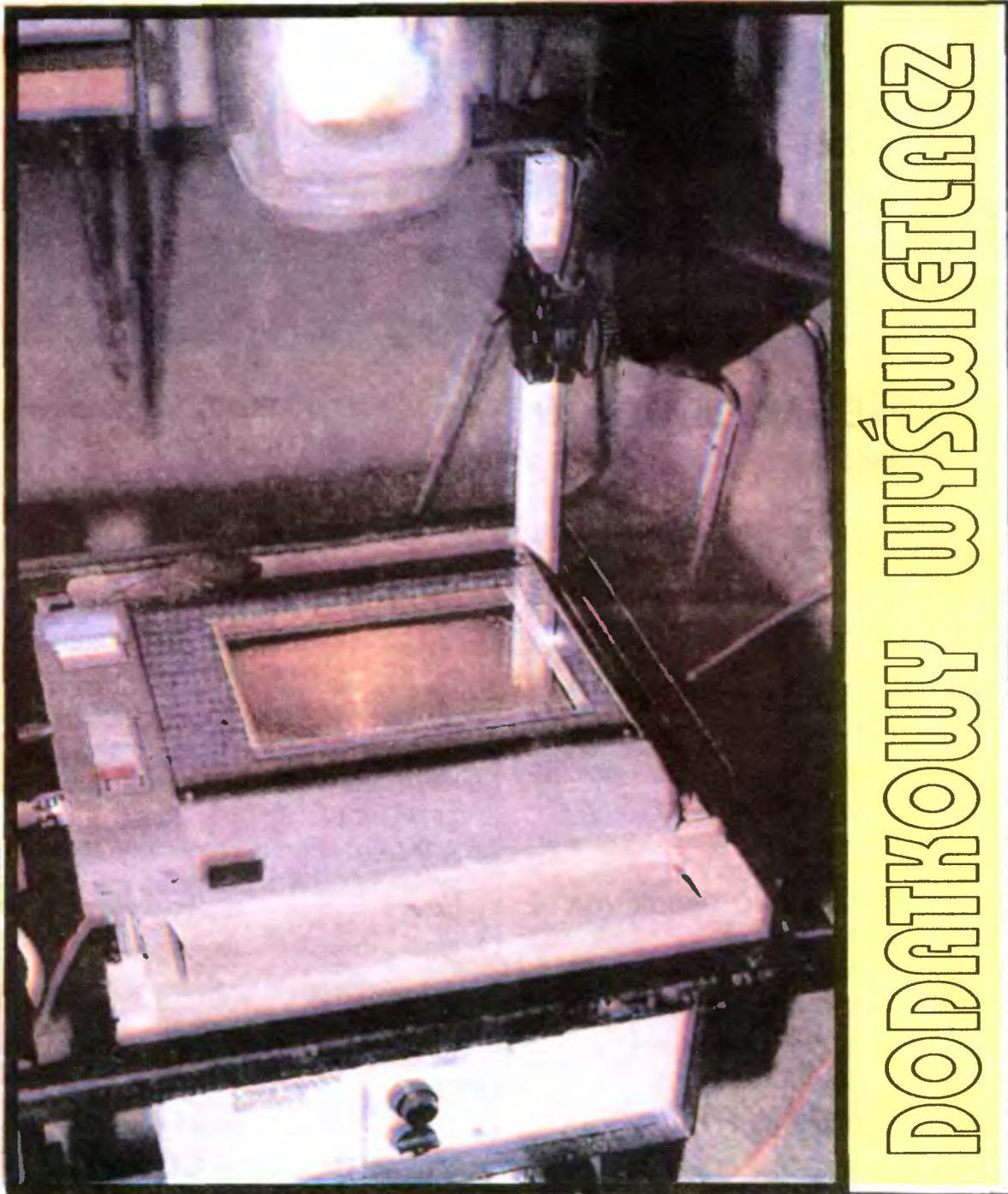
Konstrukcja tego podręcznika jest tak pomyślana, by stanowił on wszechstronną pomoc dla wszystkich, którzy już piszą, lub dopiero myślą o pisaniu programów na 8-bitowe Atari. Początkujący programiści znajdują tu kilka wskazówek odnośnie techniki programowania, a co najważniejsze mogą korzystać z książki jak ze słownika objaśniającego znaczenie każdej komendy języka programowania. Opisano tu nie jeden, lecz kilka najpopularniejszych języków: Turbo Basic XL, Basic XE, Microsoft Basic i Atari LOGO. Ma to ogromną zaletę również i dla bardziej zaawansowanych programistów. Dla nich najużyteczniejszą pomocą jest właśnie zwężone vademecum, w którym podany jest opis instrukcji, występujących w niej parametrów, zmiennych, działania, no i króciutki przykład. Wystarcza to w zupełności, by rozstrzygnąć większość wątpliwości, które mogą nasunąć się podczas pisania programu. Alfabetyczne ułożenie haseł — opisów instrukcji — to również duży atut. Jeśli chcemy sprawdzić tylko, czy prawidłowo użyliśmy jakiegoś wyrażenia, to w zwykłym podręczniku programowania musielibyśmy z pewnością przerzucić kilka kartek.

Ci, którzy posługują się tylko jednym z opisanych w prezentowanej książce języków, na pewno odkryją dzięki niej zalety innych. Myślę, że pomoże ona im wzbogacić technikę programowania.

W wielu czasopismach komputerowych można znaleźć listingi gotowych programów. Nierzadko chcielibyśmy przepisać taki program na własny komputer. Okazuje się, że występują w nim nieznanne instrukcje. Wtedy właśnie powinniśmy sięgnąć po „Języki Atari”. Być może dojdziemy od razu do wniosku, że interesujący nas program w zwykłym Basicu będzie działał zbyt wolno. Wspierani bankiem informacji zgromadzonych w „Językach Atari”, bez żadnych trudności przetłumaczymy ten program na działający znacznie szybciej Turbo-Basic.

(J. J.)

Wojciech Zientara, „Języki Atari XL/XE cz. 1”, Stołeczny Ośrodek Elektronicznej Techniki Obliczeniowej, Warszawa 1989, wyd 1, nakład 5000 egz.



Wyświetlacz ciekłokrystaliczny na projektorze.

## Konstruktorzy urządzeń peryferyjnych do Atari ST pomyśleli o ułatwieniu pracy tym, którzy stosują komputer jako pomoc dydaktyczną w wykładach lub szkoleniach.

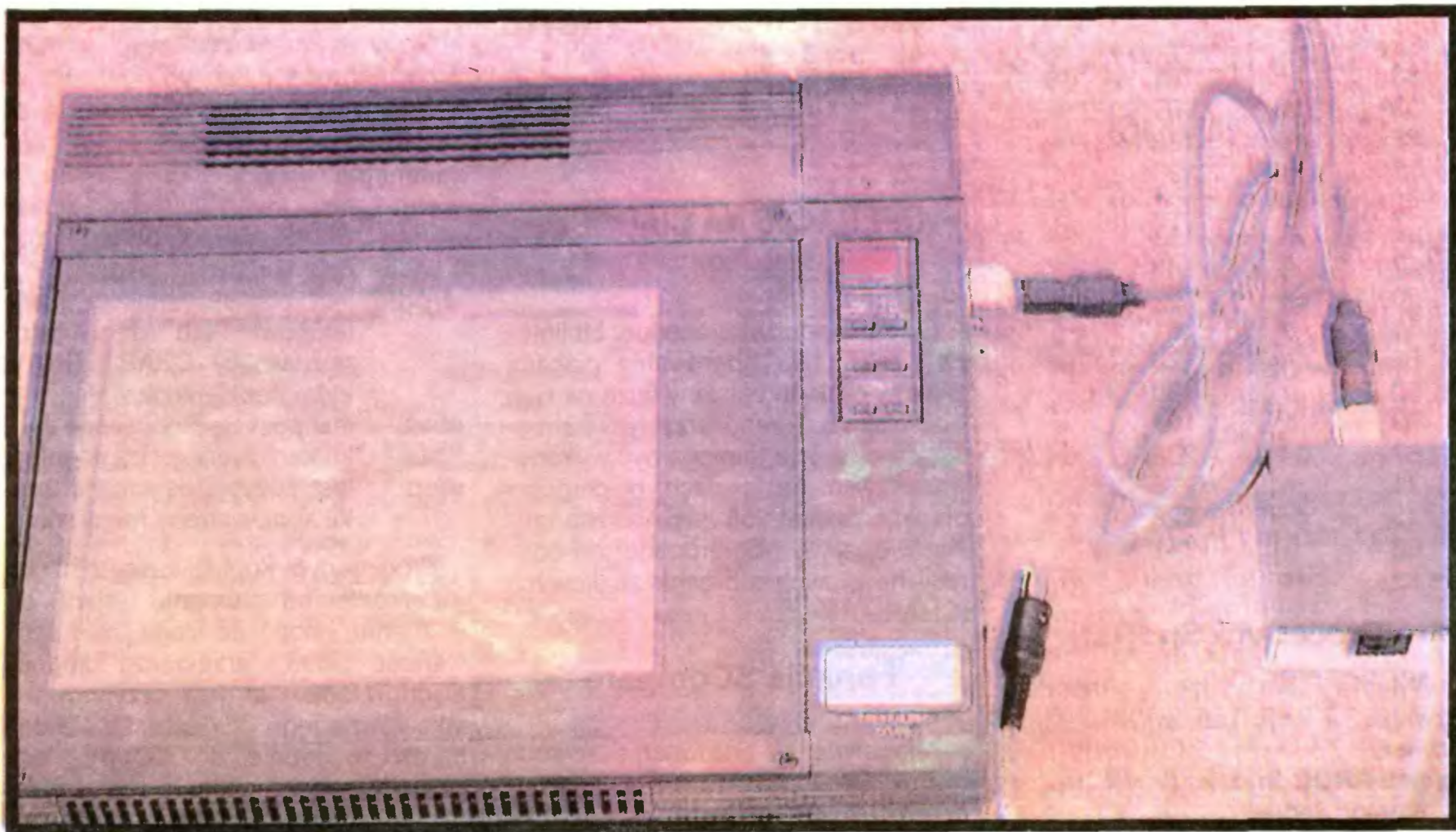
Dużą niedogodnością dla osoby prowadzącej takie szkolenie jest brak możliwości skupienia większej liczby słuchaczy przy jednym komputerze. Nauczyciel mógłby zaoszczędzić wiele czasu, gdyby udało się obraz wyświetlany na monitorze powiększyć i przenieść na duży ekran ścienny.

Pomysł ten można zrealizować praktycznie jeśli sprzęgniemy komputer Atari ST z wyświetlaczem ciekłokrystalicznym QA-50 firmy Sharp. Obraz pojawiający się na ekranie komputera jest przesyłany bezpośrednio na wyjście

monitora lub przez złącze szeregowe. Wyświetlacz przekształca odebrane sygnały i na jego ciekłokrystalicznym ekranie widzimy dokładnie to, co na monitorze. Samo urządzenie ma niewielkie rozmiary, niewiele większe niż strona formatu A3. Z łatwością możemy je umieścić na projektorze, takim jaki używają nauczyciele w prawie każdej szkole do wyświetlania rysunków i tekstu na ścianie. Projektor powiększy tekst lub rysunki z wyświetlacza do praktycznie dowolnych rozmiarów. Teraz obraz z monitora komputerowego jest już wystarczająco duży i wyraźny, by można go odczytać nawet z ostatniego rzędu sali wykładowej. Nikt z uczestników zajęć nie musi się już tłoczyć przy komputerze, każdy słucha objaśnień wykładowcy siedząc wygodnie przy własnym stoliku.

Obsługa wyświetlacza QA-50 jest bardzo prosta. Przyciski sterujące funkcjami urządzenia są umieszczone z

Wyświetlacz wraz z dodatkowym urządzeniem do równoległego przyłączenia monitora SM124.



## KLAN ATARI

prawej strony. Użytkownik ma możliwość rozjaśniania lub przyciemniania obrazu, niezależnie od jasności ustawionej na monitorze komputera oraz regulacji kontrastu. Jeden z klawiszy służy do dostosowania urządzenia do rodzaju i trybu pracy współpracującego z ST monitora (monochromatyczny lub kolorowy). Uzyskuje się w ten sposób obraz o najwyższej ostrości. Funkcja „inverse” pozwala natychmiast zamienić wyświetlany obraz na jego negatyw. Jest to bardzo wygodne, jeśli obserwowany akurat tekst lub obraz jest niewidoczny na jasnym tle. „Clear” przydaje się, gdy na chwilę chcemy przerwać wyświetlanie i wykorzystać projektor do demonstrowania zwykłych klisz. Obraz powraca na ekran wyświetlacza po ponownym przyciśnięciu tego klawisza. Ciekłokrystaliczny ekran wyświetlacza QA-50 ma rozdzielczość 640x400 punktów, tak jak monochromatyczny monitor Atari ST.

Producent wyposaża wyświetlacz QA-50 w kable przyłączeniowe, zasilacz i dodatkowe urządzenie, które pozwala na równoległe przyłączenie monitora SM124. Tym samym prowadzący wykład zyskuje niezwykle dogodną możliwość demonstrowania słuchaczom programów na Atari ST. Wykładowca pracuje w zupełnie normalny sposób, to znaczy tak, jakby komputer znajdował się w biurze lub instytucji naukowej. Jednocześnie objaśnia on bardzo dokładnie wszystkie wykonywane przez siebie czynności oraz komentuje to, co aktualnie dzieje się na monitorze. Uczestnicy, dzięki zastosowaniu wyświetlacza ciekłokrystalicznego i projektora, obserwują wszystko na dużym ekranie ściennym. Mogą oni powtarzać czynności obsługi demonstrowanego programu na swoich własnych komputerach. W ten prosty sposób wykładowca zyskuje bezpośredni kontakt ze wszystkimi uczestnikami prowadzonych przez siebie zajęć.

Dodatkowym udogodnieniem oferowanym przez producenta jest pilot na promienie podczerwone, którym można sterować pracą urządzenia na odległość. Polepsza on komfort obsługi wyświetlacza, a przez to czyni go bardziej atrakcyjnym dla użytkownika.

Czy podobne urządzenia znajdują się kiedyś w wyposażeniu sal wykładowych w naszych szkołach? Na pewno tak. Na razie jedną z przeszkód w szerszym upowszechnieniu będzie stosunkowo wysoka cena, która dla wyświetlacza Sharp QA-50 wynosi 3798 DM. Jest to obecnie zdecydowanie za dużo, jak na skromne możliwości finansowe naszej oświaty.

Janusz Jarmoch



Kontynuujemy opis procedur i funkcji zawartych w bibliotece Action!. Ten odcinek zawiera procedury dźwiękowe, funkcje odczytu manipulatorów oraz operacje na ciągach znakowych.

**PROCEDURY DŹWIĘKOWE**

W odróżnieniu od Basica Action! posiada dwie procedury dźwiękowe: Sound, która uruchamia generatory dźwięku, i SndRst, która je wyłącza.

**Procedura Sound**

Uruchamia wskazany generator dając dźwięk o parametrach określonych przez użytkownika.

format: **PROC Sound (BYTE gen, okres, barwa, siła)**

parametry:

- gen** jest jednym z czterech generatorów dźwięku Atari (0—3);
- okres** wyznacza częstotliwość (wysokość) tworzonego dźwięku; im większa liczba, tym niższy dźwięk;
- barwa** jest miernikiem zniekształceń dźwięku (dozwolone są wartości parzyste od 0 do 14);
- siła** jest to głośność generowanego dźwięku (0—15).

Procedura **Sound** pozwala na sterowanie wbudowanymi generatorami dźwięku Atari i jest bardzo podobna do instrukcji Basica o takiej samej nazwie. Czysty dźwięk można otrzymać jedynie dla **barwy** równej 10 lub 14. Inne wartości służą do uzyskiwania specjalnych efektów, na przykład: dźwięk silników, szum wiatru, wybuchy itp.

Poniższa tabela określa wartości **okresu** niezbędne dla uzyskania dźwięku odpowiadającego poszczególnym nutom:

okres	nuta	okres	nuta
29	c	90	f
31	h	95	e
33	a#	101	d#
35	a	107	d
37	g#	113	c#
39	g	120	c
42	f#	127	h
44	f	135	a#
47	e	143	a
50	d#	152	g#
53	d	161	g
56	c#	170	f#
60	c	180	f
63	h	191	e
67	a#	203	d#
71	a	215	d
75	g#	228	c#
80	g	241	c
85	f#	255	h

**Procedura SndRst**

Wyłącza wszystkie generatory dźwięku, a więc całkowicie wyłącza dźwięk.

format: **PROC SndRst ()**

parametry: żadne

**FUNKCJE MANIPULATORÓW**

Do odczytu położenia manipulatorów (joysticków i potencjometrów) są przewidziane cztery funkcje Action!: Paddle, PTrig, Stick i STrig.

**Funkcja Paddle**

Zwraca wartość określającą położenie wskazanego potencjometru. Wynikiem wykonania tej funkcji jest wartość z zakresu od 0 do 228.

format: **BYTE FUNC Paddle (BYTE port)**

parametry: **port** jest numerem odczytywanego potencjometru (0—3).

**Funkcja PTrig**

Podaje stan przycisku wskazanego potencjometru: zero oznacza naciśnięty, a jeden zwolniony.

format: **BYTE FUNC PTrig (BYTE port)**

parametry: **port** jest numerem odczytywanego potencjometru (0—3).

**Funkcja Stick**

Zwraca wartość określającą położenie wskazanego potencjometru.

format: **BYTE FUNC Stick (BYTE port)**

parametry: **port** jest numerem odczytywanego joysticka (0—1).

Zwracana wartość określa położenie joysticka według takiego schematu jak w Atari Basic.

**Funkcja STrig**

Podaje stan przycisku wskazanego joysticka: zero oznacza naciśnięty, a jeden zwolniony.

format: **BYTE FUNC STrig (BYTE port)**

parametry: **port** jest numerem odczytywanego joysticka (0—1).

**OPERACJE NA CIĄGACH**

Cpisane poniżej procedury biblioteczne służą do operowania ciągami znaków oraz do zamiany liczb na ciąg i odwrotnie. Trzeba przy tym pamiętać, że operacje te mogą być wykonywane tylko na ciągach o długości przekraczającej 255 znaków. Nie można więc użyć tych procedur do operacji na dużych tablicach znakowych (CHAR ARRAY).

**Funkcja SCompare**

Alfabetyczne porównanie dwóch ciągów (tablic CHAR ARRAY) wykonuje funkcja **SCompare**.

format: **INT FUNC SCompare (<ciąg1>, <ciąg2>)**

parametry:

**<ciąg1>** jest ciągiem znaków (stałą tekstową) lub identyfikatorem tablicy CHAR ARRAY;

**<ciąg2>** jest ciągiem znaków (stałą tekstową) lub identyfikatorem tablicy CHAR ARRAY.

W zależności od zawartości ciągów będących parametrami funkcji jej wynikiem jest wartość pokazana w poniższej tabeli:

wynik porównania	wynik funkcji
<ciąg1><ciąg2>	wartość < 0
<ciąg1>=<ciąg2>	wartość = 0
<ciąg1>><ciąg2>	wartość > 0

Porównywanie zawartości ciągów jest wykonywane według wartości kodów ATASCII kolejnych znaków obu ciągów.

**Procedura SCopy**

Służy do kopiowania zawartości jednego ciągu (tablicy) do innego.

format: **PROC SCopy (<cel>, <źródło>)**

parametry:

**<cel>** jest identyfikatorem tablicy CHAR ARRAY, do której będzie kopiowany ciąg;

**<źródło>** jest ciągiem znaków (stałą tekstową) lub identyfikatorem tablicy CHAR ARRAY, która będzie kopiowana.

Ta procedura kopiuje zawartość ciągu <źródło> do ciągu <cel>. Jeśli <cel> ma zadeklarowany wymiar mniejszy niż <źródło>, to kopiowana jest tylko część odpowiadająca zadeklarowanej długości <cel>. W odwrotnej sytuacji kopiowany jest cały ciąg <źródło>, a część ciągu <cel> pozostaje niewypełniona. Jeśli wymiar ciągu <cel> nie zostanie zadeklarowany, to zawsze będzie on równy długości ciągu <źródło>.

**Procedura SCopyS**

Służy do kopiowania części zawartości jednego ciągu (tablicy) do innego.

format: **PROC SCopyS (<cel>, <źródło>, BYTE start, stop)**

parametry:

**<cel>** jest identyfikatorem tablicy CHAR ARRAY, do której będzie kopiowany ciąg;

**<źródło>** jest ciągiem znaków (stałą tekstową) lub identyfikatorem tablicy CHAR ARRAY, która będzie kopiowana;

**start** jest pozycją pierwszego znaku kopiowanego fragmentu;

**stop** jest pozycją ostatniego znaku kopiowanego fragmentu.

Procedura ta kopiuje zawartość ciągu <źródło> od elementu „start” do elementu „stop” do ciągu <cel>. Jeśli wartość „stop” przekracza długość ciągu <źródło>, to jest przyjmowana jako równa jego długości. Działa ona bardzo podobnie do **SCopy**, lecz przepisuje tylko część ciągu <źródło> zamiast całości.

**Procedura SAssign**

Służy do kopiowania zawartości jednego ciągu <tablicy> do części innego.

format: **PROC SAssign (<cel>, <źródło>, BYTE start, stop)**

parametry:

**<cel>** jest identyfikatorem tablicy CHAR ARRAY, do której będzie kopiowany ciąg;

**<źródło>** jest ciągiem znaków (stałą tekstową) lub identyfikatorem tablicy CHAR ARRAY, która będzie kopiowana;

**start** jest pozycją umieszczenia pierwszego znaku kopiowanego fragmentu;

**stop** jest pozycją umieszczenia ostatniego znaku kopiowanego fragmentu.

Procedura ta kopiuje zawartość ciągu <źródło> do ciągu <cel> umieszczając ją od elementu „start” do elementu „stop”. Jeśli wartość „stop” przekracza długość ciągu <cel> to jest przyjmowana jako równa jego długości. Jeśli długość kopiowanego fragmentu określona przez „start” i „stop” jest większa od rzeczywistej długości ciągu <źródło>, to wartość „stop” zostaje odpowiednio zmieniona.

**Procedury Str**

Trzy poniższe procedury zamieniają liczbę podaną jako parametr na ciąg znaków. Każda z nich służy do zamiany liczb innego typu.

formaty:

**PROC StrB (BYTE liczba, <ciąg>)**

**PROC StrC (CARD liczba, <ciąg>)**

**PROC StrI (INT liczba, <ciąg>)**

parametry:

**liczba** jest wyrażeniem arytmetycznym;

**<ciąg>** jest identyfikatorem tablicy CHAR ARRAY

Procedury te zamieniają odpowiednio wartości typu BYTE, CARD lub INT na ciąg znaków złożony z cyfr podanej liczby.

**Funkcje Val**

Trzy poniższe funkcje zamieniają ciąg znaków złożony z cyfr na liczbę. Każda z nich zwraca liczbę innego typu.

formaty:

**BYTE FUNC ValB (<ciąg>)**

**CARD FUNC ValC (<ciąg>)**

**INT FUNC ValI (<ciąg>)**

parametry:

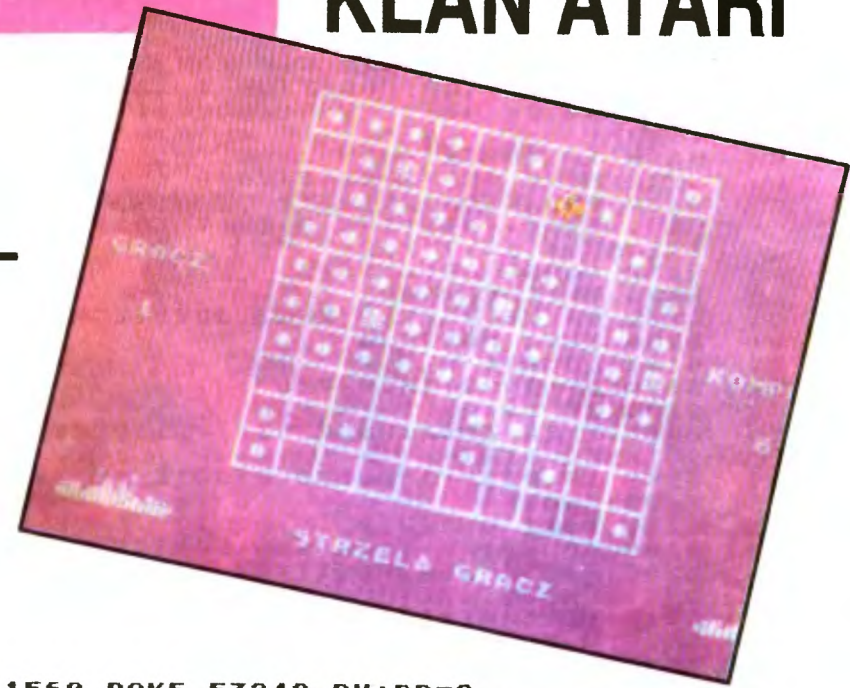
**<ciąg>** jest ciągiem znaków (stałą tekstową) lub identyfikatorem tablicy CHAR ARRAY, złożonej wyłącznie z cyfr („0” — „9”).

Funkcje te zwracają odpowiednią wartość liczbową (BYTE, CARD lub INT — zależnie od typu użytej funkcji) zawartą w podanym ciągu.

Wojciech Zientara



# OKRETY



Znana powszechnie gra w okrety, zwykle rozgrywana przy pomocy kartki papieru, doczekała się już kilku wersji komputerowych. Także Czytelnicy „Bajtka” będą mogli na swoich Atari zrealizować bitwę morską.

Gra została napisana w języku Turbo Basic XL. Zamieszczony obok listing należy dokładnie przepisać, korzystając z pomocy „Edytora Basica”, którego kody kontrolne są umieszczone przed numerem każdego wiersza (nie należy ich, oczywiście, wpisywać). Po zapisaniu programu na kasecie lub dyskietce można przystąpić do zabawy.

Piszemy RUN i naciskamy <RETURN>. Po krótkiej chwili ukazują się plansza tytułowa, a następnie menu. Odpowiedni wariant gry wybierany jest przy pomocy klawiszy konsoli. <OPTION> służy do ustalenia liczby graczy: dwóch graczy przeciwko sobie lub jeden gracz przeciw komputerowi. Klawiszem <SELECT> określa się liczbę okrętów, które wezmą udział w walce (15, 10 lub 5). Sposób sterowania programem wybiera się przy pomocy klawisza <HELP>. Naciśnięcie <START> rozpoczyna grę.

Najpierw trzeba rozstawić na planszy własne okrety. Wszystkie są jednakowe i zajmują po jednym kwadracie (tzw. „jednomasztowce”). Cursor przesuwany jest, zależnie od wybranego sposobu sterowania, joystickiem albo klawiszami ze strzałkami. Umieszczenie jednostki w wybranym miejscu następuje po naciśnięciu przycisku joysticka lub klawisza <RETURN>. Komputer liczy ustawiane okrety i po rozmieszczeniu wszystkich na planszy przechodzimy do właściwej rozgrywki.

Obaj gracze (lub gracz i komputer) oddają na zmianę po jednym strzale. Wybór celu jest wykonywany analogicznie do ustalania położenia jednostek: klawiszami ze strzałkami i <RETURN> albo joystickiem. Oddawaniu strzału towarzyszą odpowiednie efekty akustyczne. Niecelny strzał wywołuje przy tym fontannę wody, a po trafieniu słychać i widać wybuch. Każdy oddany strzał jest zaznaczany na planszy dużą kropką. W miejscu zatopienia okrętu pojawia się natomiast... trupia główka. Ponieważ jednostki nie mogą się ze sobą stykać, to automatycznie zaznaczany jest obszar, w którym nie ma już innych jednostek. Po zatopieniu ostatniego okrętu jednej ze stron gra się kończy i ogłaszany jest zwycięzca.

Życzę przyjemnej zabawy i powodzenia w walce.

Leszek Stróżowski

```

VO 1 REM GRA W OKRETY
CV 2 REM LESZEK STROZOWSKI
XO 3 REM COPYRIGHT (C) BAJTEK
NJ 4 REM
TR 10 POKE 106,164:GRAPHICS 0:POKE 752,1:
    POKE 566,158:EXEC POL:POKE 756,172
UY 15 POKE 82,0:POKE 731,1
WR 50 DIM P1(100),P2(100),A$(12),D$(8),G1
    $(8),G2$(8),K$(8),G$(8),S$(15)
ZL 55 A$="BITWYMORSKIE":G1$="GRACZ 1":G2
    $="GRACZ 2":K$="KOMPUTER":G$="GRACZ"
SB 57 S$="STRZELANIE"
YI 60 EXEC PLAYER1
WY 70 EXEC TYTUL
CA 80 EXEC OPCJE
OR 85 WU=0
QU 90 EXEC PLANSZA
VD 92 EXEC MIG:M=USR(1536)
DS 95 SOUND 3,10,8,1:SOUND 2,50,8,1
WA 100 MOVE 41024,42312,840:MOVE 41024,43
    172,840:POSITION 7,23
BG 103 IF OPT=2 THEN ? "GRACZ 1";
GI 104 IF OPT=1 THEN ? "GRACZ 2";
FD 105 POSITION 16,23:?"-USTAW OKRETY"
FW 107 KJ=0:POSITION 34,13:?" ":POSITION
    3,13:?"0 ";
VZ 110 EXEC KURSOR
QW 130 ON HELP EXEC KLAW,JOYSTICK
FS 135 EXEC POZYCJA
JS 140 LOCATE X,Y,Z
TX 145 IF Z=88 THEN 300
NJ 150 IF Z=20 THEN 310
YM 160 POSITION X,Y:?"X":EXEC MOD:EXEC P
    OLEMARTWE:P1(PT)=1
XR 170 WU=WU+1:POSITION 3,13:?"WU:IF WU=I
    0 THEN 190
PD 180 PAUSE 20:KEY=32:GOTO 130
QR 190 PAUSE 100
HU 195 IF OPT=1 THEN GOS USTKOM
CP 200 POSITION 7,23:?"GRACZ 2 - USTAW
    OKRETY":KJ=1
FK 210 POKE 53277,0:MOVE 42312,41024,840:
    POSITION 3,13:?" ":POSITION 34,13:?"
    0 ";
UN 215 EXEC KURSOR:WU=0
QX 230 ON HELP EXEC F WU,JOYSTICK
FT 235 EXEC POZYCJA
JT 240 LOCATE X,Y,Z
UH 245 IF Z=88 THEN 310
PK 250 IF Z=20 THEN 310
AL 260 POSITION X,Y:?"X":EXEC MOD:EXEC
    OLEMARTWE:P2(PT)=1
DJ 270 WU=WU+1:POSITION 34,13:?"WU:IF WU=
    IO THEN 290
QE 280 PAUSE 20:KEY=32:GOTO 230
NS 290 POKE 53277,0:PAUSE 100:GOTO 5000
KD 300 POSITION X,Y:?"X":PAUSE 20:GOTO 1
    30
HH 310 POSITION X,Y:?"0":PAUSE 20:GOTO 1
    30
LT 320 POSITION X,Y:?"X":PAUSE 20:GOTO 2
    30
IX 330 POSITION X,Y:?"0":PAUSE 20:GOTO 2
    30
ZM 500 PROC KURS
TO 510 POKE 53277,0:MOVE 1700,PMBASE+10 4
    +PY,46:MOVE 1746,PMBASE+1024+21,46:POK
    E 559,62:POKE 54279,176:POKE 704,222
KP 520 POKE 53248,86:POKE 53277,2:PX=86:P
    Y=21
UT 530 ENDPROC
IW 550 PROC POZYCJA
AV 560 X=((PX-86)/8)*2+10:Y=((PY-21)/16)*
    2+1
YK 570 PT=((X-10)/2+1)+((Y-1)/2*10)
ND 580 ENDPROC
SF 600 PROC DRGANIE
AJ 610 DL=DPEEK(560)
QF 620 FOR R=10 TO 30:SOUND 0,100,8,15:50
    UND 1,R,0,15
AB 630 CL=RND*15:SETCOLOR 2,7,CL:SETCOLOR
    4,7,CL
LN 640 POKE DL,16*INT(RND*8):PAUSE 1:NEXT
    R:POKE DL,112:SETCOLOR 2,7,2:SETCOLOR
    4,7,2
RL 650 FOR H=14 TO 0 STEP -1:SOUND 0,100,
    8,H:SOUND 1,R,0,H
MP 660 PAUSE 1:NEXT H
WC 670 ENDPROC
ZS 700 POSITION X,Y:?"0":GOTO 5040
NF 710 POSITION X,Y:?"0":GOTO 5040
BA 720 POSITION X,Y:?"0":GOTO 5140
ON 730 POSITION X,Y:?"0":GOTO 5140
LU 800 PROC ZMAZ
ET 810 POSITION 7,23:?"
    ";
UU 820 ENDPROC
DC 1000 PROC JOYSTICK
KA 1070 S=STICK(KJ)
FR 1080 IF S=11 THEN PX=PX-8:EXEC CEL
FU 1090 IF S=7 THEN PX=PX+8:EXEC CEL
YE 1100 IF PX<86 THEN PX=86
XH 1110 IF PX>158 THEN PX=158
JS 1120 POKE 53248,86:RP=0
NM 1140 IF S=14 THEN PY=PY-16:RP=1:EXEC C
    EL
LH 1150 IF S=13 THEN PY=PY+16:RP=1:EXEC C
    EL
KW 1160 IF PY<21 THEN PY=21
WB 1170 IF PY>165 THEN PY=165
ZS 1180 IF RP=1 THEN MOVE 1746,PMBASE+102
    4+PY,46
PH 1190 IF STRIG(KJ)=0 THEN 1230
QF 1200 GOTO 1070
YP 1230 ENDPROC
LO 1500 PROC KLAW
FJ 1505 POKE 729,8:POKE 730,1
JK 1510 POKE 764,255:KEY=32:GET KEY
LX 1520 IF KEY=43 THEN PX=PX-8:EXEC CEL
JM 1530 IF KEY=42 THEN PX=PX+8:EXEC CEL
YY 1540 IF PX<86 THEN PX=86
YB 1550 IF PX>158 THEN PX=158

```

```

KM 1560 POKE 53248,86:RP=0
XN 1570 IF KEY=45 THEN PY=PY-16:RP=1:EXEC
    CEL
UM 1580 IF KEY=61 THEN PY=PY+16:RP=1:EXEC
    CEL
LN 1590 IF PY<21 THEN PY=21
VQ 1600 IF PY>165 THEN PY=165
ZH 1610 IF RP=1 THEN MOVE 1746,PMBASE+102
    4+PY,46
OY 1620 IF KEY=155 THEN 1640
QE 1630 GOTO 1510
ZA 1640 ENDPROC
LJ 2000 PROC POLEMARTWE
UK 2010 IF X<10 AND Y>1 THEN GOSUB 2100
PG 2020 IF X<10 THEN GOSUB 2110
BP 2030 IF X<10 AND Y<19 THEN GOSUB 2120
KA 2040 IF Y>1 THEN GOSUB 2130
WL 2050 IF Y<19 THEN GOSUB 2140
FB 2060 IF X<28 AND Y>1 THEN GOSUB 2150
YJ 2070 IF X<28 THEN GOSUB 2160
ML 2080 IF X<28 AND Y<19 THEN GOSUB 2170
ZE 2090 ENDPROC
PP 2100 POSITION X-2,Y-2:?"0":RETURN
KU 2110 POSITION X-2,Y:?"0":RETURN
OH 2120 POSITION X-2,Y+2:?"0":RETURN
IC 2130 POSITION X,Y-2:?"0":RETURN
GV 2140 POSITION X,Y+2:?"0":RETURN
OY 2150 POSITION X+2,Y-2:?"0":RETURN
KD 2160 POSITION X+2,Y:?"0":RETURN
NQ 2170 POSITION X+2,Y+2:?"0":RETURN
BI 3000 # USTKOM
ZI 3005 POKE 53277,0:MOVE 42312,41024,840
    :POSITION 7,23:?"OKRETY USTAWIA KUR
    SOR":
KR 3010 WU=0:PAUSE 150:POKE 559,0
UP 3020 X=INT(RND*28+1)
UR 3030 IF X<10 THEN 3020
CX 3040 LP=(X-10) MOD 2
DF 3050 IF LP<0 THEN 3020
VI 3060 Y=INT(RND*19+1)
GR 3070 IF Y<1 THEN 3060
RE 3080 LP=(Y-1) MOD 2
HB 3090 IF LP<0 THEN 3060
XZ 3100 LOCATE X,Y,Z
KD 3110 IF Z=88 THEN 3480
EV 3120 IF Z=20 THEN 3490
KI 3130 PT=((X-10)/2+1)+((Y-1)/2*10)
DM 3150 P2(PT)=1:POSITION X,Y:?"X":EXEC
    POLEMARTWE:WU=WU+1
EB 3160 IF WU=10 THEN 3200
PJ 3170 GOTO 3020
HM 3200 POKE 53277,0:POKE 559,62:GOTO 500
    0
YP 3480 POSITION X,Y:?"X":GOTO 3020
TE 3490 POSITION X,Y:?"0":GOTO 3020
SY 3500 MOVE 43172,41024,840:MOVE 1700,PM
    BASE+1024+PY,46
MF 3510 POSITION 11,23:?"STRZELA KOMPUT
    ER":POSITION 3,13:?"I01:" ":POSITIO
    N 34,13:?"I02:" ";
OR 3520 PX=INT(RND*158+1)
MT 3525 IF PX<86 THEN 3520
QY 3530 LP=(PX-86) MOD 8
HN 3540 IF LP<0 THEN 3520
PF 3550 PY=INT(RND*167+1)
KT 3555 IF PY<21 THEN 3550
AB 3560 LP=(PY-21) MOD 16
KK 3570 IF LP<0 THEN 3550
WI 3580 EXEC POZYCJA
ZI 3590 LOCATE X,Y,Z
IA 3600 IF Z=20 THEN 3980
DI 3610 IF Z=101 THEN 3990
AY 3615 MOVE 1746,PMBASE+1024+PY,46:POKE
    53248,86:POKE 53277,2
ZO 3620 PAUSE 70:POKE 53277,1:EXEC ZMAZ:E
    XEC STRZ:IF P1(PT):GOSUB 4300:ELSE :GO
    SUB 4400:ENDIF
DI 3630 POKE 53277,0
QR 3640 IF IO1=0 THEN GOS KONIEC
KY 3650 PAUSE 100:POSITION 3,13:?" ":P
    OSITION 34,13:?" ";
SK 3660 MOVE 41024,43172,840:GOTO 5020
ZK 3980 POSITION X,Y:?"0":GOTO 3520
QB 3990 POSITION X,Y:?"0":GOTO 3520
UT 4100 QP=76:QK=184:QS=1:WF=1:EXEC LOT
HX 4110 EXEC DRGANIE:POSITION X,Y:?"e"
NO 4120 EXEC POLEMARTWE:IO2=IO2-1:POSITIO
    N 34,13:?"I02:" ";
UM 4130 POKE 53252,0:RETURN
RG 4200 QP=76:QK=166:QS=1:WF=1:EXEC LOT:E
    XEC PLUSK
AP 4210 POSITION X,Y:?"0":POKE 53252,0:R
    ETURN
UX 4300 QP=174:QK=66:QS=-1:WF=0:EXEC LOT
IB 4310 EXEC DRGANIE:POSITION X,Y:?"e"
CF 4320 EXEC POLEMARTWE:IO1=IO1-1:POSITIO
    N 34,13:?"I01:" ";
UR 4330 POKE 53252,0:RETURN
VI 4400 QP=174:QK=84:QS=-1:WF=0:EXEC LOT:
    EXEC PLUSK
AT 4410 POSITION X,Y:?"0":POKE 53252,0:R
    ETURN
TG 5000 IO1=IO:IO2=IO
AV 5020 POKE 77,0:POKE 53277,0:MOVE 42312
    ,41024,840:POSITION 7,23
KY 5022 IF OPT=2 THEN ? "STRZELA GRA
    CZ 1";
XL 5024 IF OPT=1 THEN ? "STRZELA GR
    ACZ 2";
SF 5030 KJ=0:POSITION 3 13:?"I01:" ":P:5
    ITION 34,13 ? I02:" ";
IB 5032 PAUSE 0:KEY=32
LZ 5035 EXEC K RSOR
NR 5040 ON HEL EXEC KL W,JOYSTICK
UR 5050 EXEC F ZYCJA
YR 5060 LOCATE X,Y,Z
JJ 5070 IF Z=20 THEN 700
CH 5080 IF Z=101 THEN 710
ES 5085 EXEC ZMAZ:POKE 53277,1:POKE 53248
    ,0:EXEC STRZ
ZE 5090 IF P2(PT):GOSUB 4100:ELSE :GOSUB
    4200:ENDIF
CV 5092 POKE 53277,0

```











# TETRIS

Serdecznie przepraszamy wszystkich, którzy pokusili się o wpisanie Tetrisa z Bajtka 11/89. Na skutek niewyobraźnego błędu zamiast fragmentu pamięci od adresu 60000 wydrukowany został fragment od adresu 0!

Obok prezentujemy listing ulepszanego i działającego Tetrisa. Aby uzyskać grę, należy:

1. Wpisać Listing 1.
2. Nagrać go na taśmę — SAVE „Tetris” LINE 1

3. Zresetować komputer i wpisać Listing 2.
4. Uruchomić go i po pojawieniu się komunikatu „Start tape...” nagrać plik kodu tuż za nagrany poprzednio programem.

Klawisze sterujące:

1,2 — przesuwanie klocka

8,9 — obrót klocka

SPACE — zrzuć klocka

Wszystkie inne informacje już w programie.

*Maciej Pietraś  
Marcin Przasnyski*

```

10 REM   BAJTEK TETRIS
20 BORDER 7: PAPER 7: INK 0: CLEAR 3e4
: LOAD "tetris.1"CODE
30 CLS : FOR f=64 TO 71 STEP 2: POKE 2
3681,f: LPRINT TAB 11;"T E T R I S": NEX
T f
35 FLOT 0,96: DRAW 255,0
40 PRINT #0,TAB 5;"press any key to st
art"
50 FOR f=0 TO 13: BEEP .01,f: RANDOMIZ
E USR 3583
60 IF INKEY$<>"" THEN GO TO 90
65 NEXT f
66 FOR f=13 TO 0 STEP -1: BEEP .01,f:
NEXT f
70 INPUT ': GO TO 40
90 CLS : REM main program
100 FOR f=2 TO 21: PRINT AT f,18; BRIGH
T 1;" ": NEXT f
110 FOR f=2 TO 21: PRINT AT f,28; BRIGH
T 1;" ": NEXT f
120 PRINT #0;TAB 18; BRIGHT 1;"
": REM 11 spacji
130 POKE 23250,120
135 POKE 23260,120
140 FLOT 31,120
150 DRAW 41,0
160 DRAW 0,-25
170 DRAW -41,0
180 DRAW 0,25
190 FLOT 29,122: DRAW 45,0: DRAW 0,-29:
DRAW -45,0: DRAW 0,29
200 FLOT 20,169: DRAW 100,0: DRAW 0,-27
: DRAW -100,0: DRAW 0,27
210 PRINT AT 1,3;"LEVEL:"
220 PRINT AT 3,3;"SCORE:"
230 PRINT AT 8,10;"<- NEXT"
240 PRINT AT 11,2;"[1] <-"
245 PRINT AT 12,2;"[2] ->"
250 PRINT AT 13,2;"[■] *"
255 PRINT AT 14,2;"[9] <="
260 PRINT AT 15,2;"[0] =>"
265 PRINT AT 16,2;"[S] HALT"
270 PRINT AT 17,2;"[C] CONT"
275 PRINT AT 18,2;"[N] NEXT ON/OFF"
280 PRINT AT 19,2;"[P] LEVEL UP"
285 PRINT AT 20,2;"[G] BEEP ON/OFF"
290 FLOT 10,0: DRAW 130,0: DRAW 0,100:
DRAW -63,0: FLOT 26,100: DRAW -16,0: DRA
W 0,-100
300 POKE 60002,0
310 POKE 60003,0
320 BEEP .5,1: BEEP .2,1: BEEP .2,0: BE
EP .3,4
330 RANDOMIZE USR 6E4
340 PRINT AT 10,19: FLASH 1;"GAME OVER"

350 BEEP .5,1: BEEP .2,1: BEEP .2,0: BE
EP .3,-10
360 PRINT AT 12,18; INVERSE 1;"AGAIN (Y
/N)"
370 IF INKEY$="Y" OR INKEY$="y" THEN G
O TO 90
380 IF INKEY$="n" OR INKEY$="N" THEN G
O TO 30
390 GO TO 370
    
```

```

1 DATA 24,7,255,1,255,0,90,12,0,33,677
2 DATA 0,0,34,103,234,62,15,50,9,92,599
3 DATA 251,62,2,205,1,22,62,16,215,62,898
4 DATA 8,215,62,17,215,62,8,215,62,19,883
5 DATA 215,62,8,215,205,148,234,205,239,235,1766
6 DATA 24,46,245,62,22,215,62,1,215,62,954
7 DATA 13,215,241,33,99,234,14,1,126,183,1159
8 DATA 40,7,14,9,254,10,48,1,79,113,575
9 DATA 197,62,48,129,215,193,62,10,145,135,1196
10 DATA 111,135,135,133,50,102,234,201,205,45,1351
11 DATA 236,205,27,237,40,13,205,227,236,237,1663
12 DATA 75,103,234,62,35,50,9,92,201,205,1066
13 DATA 227,236,205,208,236,202,95,235,254,49,1947
14 DATA 40,72,254,50,40,79,254,32,40,102,963
15 DATA 254,57,40,82,254,48,40,86,230,223,1314
16 DATA 254,83,32,11,58,8,92,230,223,254,1245
17 DATA 67,32,247,24,90,254,78,32,9,58,891
18 DATA 98,234,47,50,98,234,24,77,254,80,1196
19 DATA 32,9,33,99,234,52,205,148,234,24,1070
20 DATA 64,254,71,32,60,58,100,234,47,50,970
21 DATA 100,234,24,51,205,33,237,32,46,33,995
22 DATA 40,238,53,24,40,205,39,237,32,35,943
23 DATA 33,40,238,52,24,29,205,64,237,204,1126
24 DATA 124,237,24,21,205,45,237,204,128,237,1462
25 DATA 24,13,33,41,238,229,205,27,237,225,1272
26 DATA 32,3,52,24,246,205,27,237,202,232,1260
27 DATA 235,205,227,236,58,99,234,42,103,234,1673
28 DATA 95,22,0,25,34,103,234,205,239,235,1192
29 DATA 17,32,0,33,0,4,58,100,234,183,661
30 DATA 196,181,3,62,21,61,202,194,234,71,1225
31 DATA 8,14,19,205,153,237,229,1,9,0,875
32 DATA 62,56,237,177,225,32,3,8,24,231,1055
33 DATA 229,42,103,234,58,99,234,135,95,135,1364
34 DATA 135,131,95,22,0,25,34,103,234,205,984
35 DATA 239,235,17,80,0,33,0,8,58,100,770
36 DATA 234,183,196,181,3,225,8,79,8,121,1238
37 DATA 183,32,13,245,62,56,43,6,11,205,856
38 DATA 203,236,241,8,24,179,229,1,224,255,1600
39 DATA 9,209,229,1,9,0,237,176,225,61,1156
40 DATA 24,224,33,41,238,52,195,215,234,8,1264
41 DATA 245,205,247,235,241,8,201,62,22,215,1681
42 DATA 62,3,215,62,9,215,42,103,234,1,946
43 DATA 16,39,205,34,236,1,232,3,205,34,1005
44 DATA 236,1,100,0,205,34,236,1,10,0,823
45 DATA 205,34,236,125,229,198,48,215,225,201,1716
46 DATA 175,237,66,56,3,60,24,249,9,24,903
47 DATA 239,17,171,237,213,33,235,237,1,56,1439
48 DATA 0,237,176,6,0,14,18,205,153,237,1046
49 DATA 229,62,56,6,11,197,205,203,236,193,1398
50 DATA 225,17,32,0,25,205,203,236,42,45,1030
51 DATA 238,125,230,6,198,2,135,135,50,39,1158
52 DATA 238,38,0,41,41,41,209,25,34,42,709
53 DATA 238,33,23,1,34,40,238,58,120,92,877
54 DATA 230,7,254,7,40,247,50,45,238,1,1119
55 DATA 4,7,205,153,237,62,120,6,5,205,1004
56 DATA 203,236,43,17,32,0,25,6,5,119,686
57 DATA 43,16,252,35,25,6,5,205,203,236,1026
58 DATA 58,98,234,183,40,67,42,40,238,229,1229
59 DATA 33,5,8,34,40,238,42,42,238,229,909
60 DATA 58,45,238,33,171,237,135,135,135,95,1282
61 DATA 22,0,25,34,42,238,205,227,236,225,1254
62 DATA 34,42,238,225,34,40,238,24,24,119,1018
63 DATA 35,16,252,201,175,50,8,92,237,75,1141
64 DATA 101,234,58,8,92,183,192,11,120,177,1176
65 DATA 32,246,201,58,39,238,24,2,62,56,958
66 DATA 8,237,75,40,238,205,153,237,237,91,1521
67 DATA 42,238,26,60,19,8,119,8,61,200,781
68 DATA 245,26,19,35,183,40,19,43,43,254,907
69 DATA 2,40,13,35,213,17,224,255,61,40,900
70 DATA 3,17,32,0,25,209,241,24,222,205,978
71 DATA 232,236,4,24,50,205,232,236,13,24,1256
72 DATA 44,205,232,236,12,24,38,205,232,236,1464
73 DATA 205,128,237,237,75,40,238,205,83,237,1685
74 DATA 245,205,124,237,241,201,205,232,236,205,2131
75 DATA 124,237,237,75,40,238,205,83,237,245,1721
76 DATA 205,128,237,241,201,205,146,237,192,237,2029
77 DATA 91,42,238,26,19,197,8,26,19,12,678
78 DATA 183,40,12,13,5,61,40,7,4,13,378
79 DATA 61,40,2,12,4,205,146,237,32,4,743
80 DATA 8,61,32,228,193,201,14,1,24,2,764
81 DATA 14,255,237,91,42,238,26,71,19,26,1019
82 DATA 129,230,3,18,19,16,248,201,205,153,1222
83 DATA 237,126,254,56,201,197,38,0,104,41,1254
84 DATA 41,41,41,41,6,0,9,1,0,88,268
85 DATA 9,193,201,3,0,1,2,0,0,0,409
86 DATA 0,4,2,0,0,0,0,0,0,4,10
87 DATA 2,0,0,1,0,0,0,4,0,2,9
88 DATA 2,1,0,0,0,4,2,0,1,0,10
89 DATA 0,0,0,4,0,2,1,2,0,0,9
90 DATA 0,5,2,0,0,2,1,0,0,0,10
91 DATA 0,0,0,0,3,3,0,3,0,1,10
92 DATA 2,0,0,0,0,4,2,0,0,0,8
93 DATA 0,0,0,4,2,0,0,1,0,0,7
94 DATA 0,4,0,2,2,1,0,0,0,4,13
95 DATA 2,0,1,0,0,0,0,4,0,2,9
96 DATA 1,2,0,0,0,5,2,0,0,2,12
97 DATA 1,0,0,0,0,0,0,24,23,1,49
98 DATA 211,237,0,1,0,0,0,0,0,0,449
8000 CLEAR 5e4: LET a=60000
8010 FOR n=1 TO 98
8020 LET s=0
8030 FOR m=0 TO 9
8040 READ w: POKE a,w: LET s=s+w: LET a=a+1
8050 NEXT m: READ w: IF w<>s THEN PRINT "Popraw linie
":n: STOP
8060 NEXT n
8070 SAVE "tetris.1"CODE 60000,975
    
```



# PŁYTKA DO AY

**Na gorące prośby Czytelników publikujemy dzisiaj projekt płytki do zmontowania generatora dźwięku AY 3-8910 do ZX Spectrum.**

Gwoli przypomnienia: **AY 3-8910** to czterdziestonóżkowy układ do generacji dźwięku w trzech niezależnych kanałach stereo. Zamontowany jest on m.in. w Spectravideo, Atari ST, Amstrad 6128, Spectrum 128. W niniejszym rozwiązaniu, jako przystawka do Spectrum 48 lub + podłączana do szyny znacznie rozszerzy możliwości muzyczne tego komputera. Większość gier wykorzystuje generator w trakcie działania, oprócz tego istnieją i powstają niezliczone ilości programów muzycznych. Za pomocą **MIDI** można przenosić muzykę z profesjonalnych keyboardów. Możliwość wykorzystania AY-greka jest niezwykle dużo.

W Klanie Spectrum prezentujemy już od pół roku różne programy i informacje dotyczące AY-greka. Będziemy to kontynuować, gdyż zainteresowanie jest ogromne.

Pierwszy schemat generatora, zamieszczony w „Bajtku” 6/89 był błędny. Jest to nasza wina i gdy zorientowaliśmy się, że coś jest nie w porządku, zamieściliśmy (w 11/89) schemat poprawny, wg projektu **Stanisława Winickiego**. Drukowana dzisiaj płytka powstała na bazie tego projektu.

Zanim zaczniecie montować, a nawet kompletować części, pamiętajcie o kilku rzeczach:

— chip AY jest stosunkowo drogi i trudny do zdobycia. Dlatego poszukiwania należy zacząć od niego,

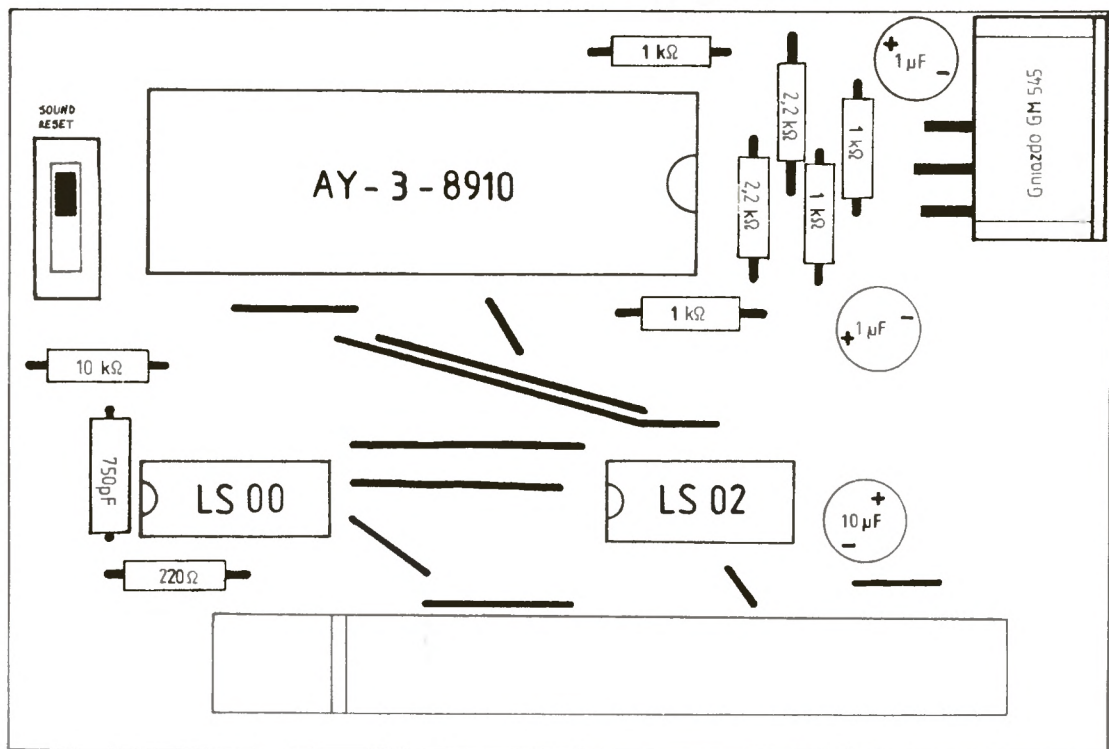
- płytkę i montaż powinien wykonać doświadczony elektronik, korzystając z pewnych części, dokładnie według opisu,
- w przypadku umieszczania płytki w osobnej obudowie, należy zdobyć złącze krawędziowe w dobrym stanie i wyprowadzić na drugą stronę przelot,
- jeśli płytkę umieszczać będziecie wewnątrz innego interfejsu, np. Kempstona lub wewnątrz obudowy Spectrum +, zadbajcie o porządne połączenie płytki z szyną. Oprócz tego, ścieżki oryginalnej płytki Spectrum są wąziutkie, a układy delikatne, więc ostrożnie z lutownicą,
- przełącznik lub przycisk **RESET** (chodzi o wyzerowanie rejestrów układu i wygaszenie dźwięku) należy wyprowadzić na zewnątrz,
- zamiast **AY 3-8910** można zastosować kompatybilny **AY 3-8912**,
- generator należy połączyć ze wzmacniaczem (najlepiej stereo-fonicznym) kablem DIN, więc nie zapomnijcie o tym i nie pomylicie się przy jego podłączaniu.

Te informacje powinny wystarczyć i wyjaśnić wszystkie wątpliwości. Jeszcze raz przypominamy: **generator powinien zmontować tylko doświadczony majsterkowicz.**

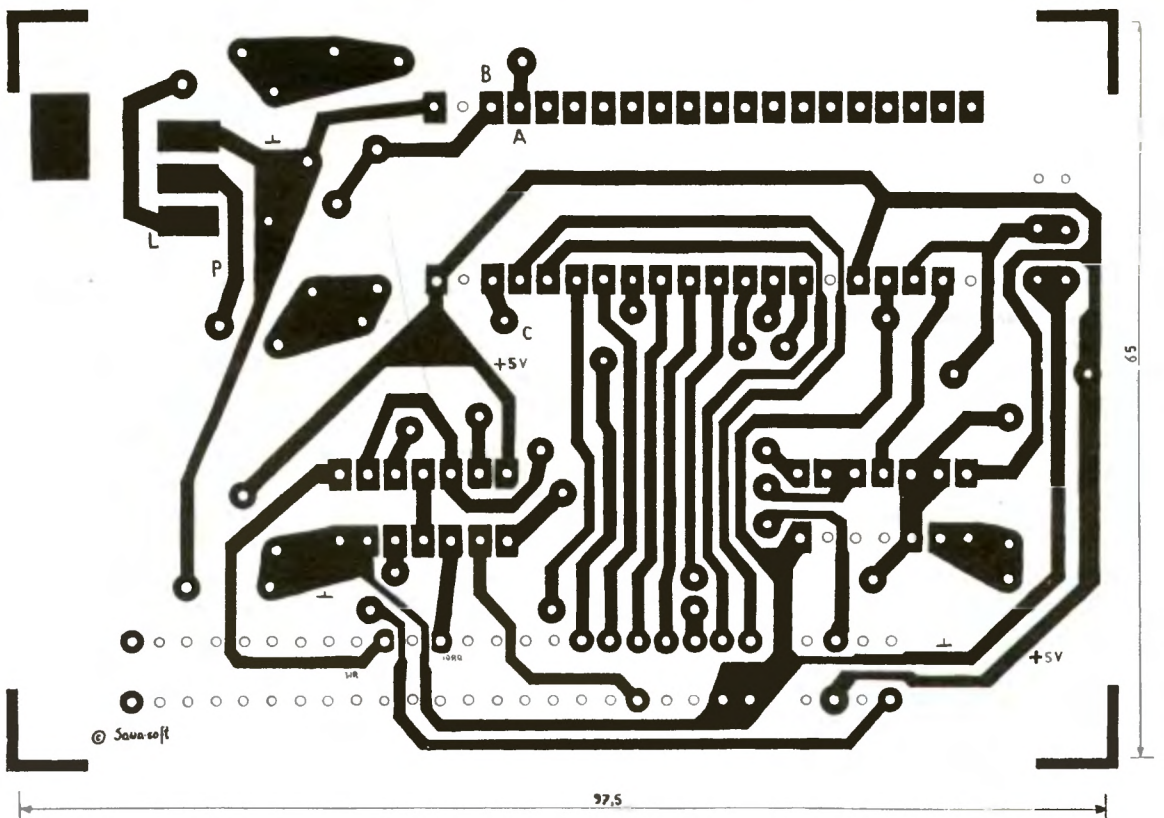
*Marek Sawicki  
Marcin Przasnyski*

**spis części:**  
AY 3-8910 lub 8912

- 74LS00 } — układy scalone
- 74LS02 }
- 750 pF, 10µF, 2x1 µF — kondensatory
- 220Ω, 2x2.2 kΩ, 4x1 kΩ, 10 kΩ — oporniki
- gniazdo GM 545
- przycisk dzwonkowy lub przełącznik ew. złącze krawędziowe



Rys.1 Widok elementów AY-3-8910 SOUND INTERFACE

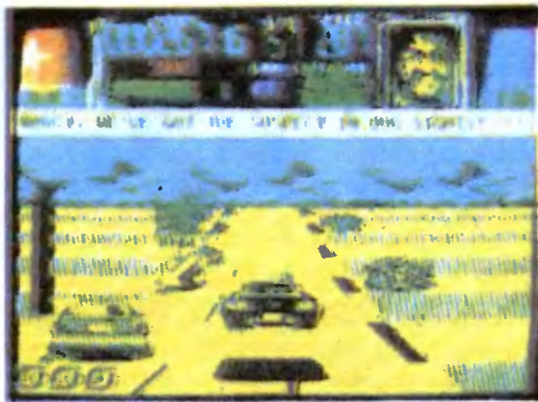


Rys. 2 Widok płytki do AY-3-8910 SOUND INTERFACE

# NOWOŚCI



**ALTERED BEAST**



**CHASE HQ**



**UNTOUCHABLES**



**TUSKER**

**Piszę te słowa w środku grudnia, więc nie wiem nic o ustawie, która ma wejść 1 stycznia, czyli dla Was jest już aktualna. Co za sobą pociągnie? Czy każe płacić ogromne pieniądze i za gry? Czy będziemy musieli skasować swoje zbiory?**

Dla mnie to przyszłość. Dlatego, póki mogę, chcę zaprezentować zwiastuny nowych gier, które przyszyły wraz z listopadowym numerem CRASH-a.

**NIETYKALNI** czyli The Untouchables, kilka razy już zapowiadani. Sześć dogrywanych leveli, podobnie jak w RoboCopie. Akcja ściśle odnosząca się do wydarzeń w filmie. Grafika Martina McDonalda, program napisany w firmie Ocean. Jak tylko zagramy, obiecujemy szerszy opis.

**ALTERED BEAST** — kolejny pomysł serii Ghosts'N Goblins, przeniesiony z automatów. Mnóstwo kolorów, ogromne postacie i szybka akcja. Bohaterem jest wskrzeszony gladiator, który musi uratować swą córkę z rąk czarowników.

**CHASE HQ** firmy Ocean to kolejny samochód do prowadzenia w tłoku rywali. Nowe akcenty, nieznanne dotąd w takich symulatorach, urozmaicona grafika.

**TUSKER** — wieloetapowe i wielowątkowe przygody podróżnika w Afryce. Trzy dogrywane części, kolorowa, ciekawa grafika. Kolejność etapów pomyślana w postaci podróży z południa na północ. Do zdobycia duży skarb, jak przyrzeka firma System 3.

*Gen Martinez*

# SPECTRUMOWA LISTA PRZEBOJÓW

**Nie zdążyliśmy otrzymać jeszcze ani jednego listu, a już przyszedł czas na kolejne notowanie. Znow zebrał się w sile pięciu osób i ustaliliśmy:**

**Grafika:**

1. BATMAN
2. SAVAGE
3. 3D POOL
4. INDIANA JONES
5. TOTAL ECLIPSE

**Użytki:**

1. GENS
2. DEERCOPY
3. MONS 2

4. ZEBRA
5. ART STUDIO

**Muzyka 48:**

1. SAVAGE
2. VECTRON
3. ZANTHRAX
4. FIREFLY
5. GRYZOR

**Muzyka 128:**

1. THE TUBE
2. PLATOON
3. SKATE CRAZY II
4. GLIDER RIDER
5. MIAMI VICE

*Gen Martinez*



# TEST KOMPUTERA SPECTRAVIDEO SUI 738



**Korzystając z uprzejmości firmy PC Plus prezentujemy na łamach Bajtka komputer Spectravideo SVI-738. Ta bardzo udana konstrukcja, niezbyt powszechna na naszym rynku, dotarła do Polski za pośrednictwem Centralnej Składnicy Harcerskiej już trzy lata temu. W większości sprzęt ten trafił do szkół, ale pewna liczba egzemplarzy znalazła się także w rękach osób prywatnych.**

## SPRZĘT

Komputer Spectravideo SVI-738 jest sprzętem 8-bitowym wyposażonym w procesor Z80. Wyświetlanie obrazu wspomaganie jest przez procesor wizyjny Yamaha 9038. Pamięć operacyjna RAM wynosi 64 KB, z możliwością rozszerzenia jej o kolejne 64 KB lub więcej. Dodatkowe 16 KB jest wyodrębnione na pamięć obrazu. Stosunkowo duża jest też pamięć typu ROM — 56 KB, z czego 32 KB przeznaczony są na wbudowany interpreter języka MSX BASIC, 16 KB na procedury obsługi stacji dysków, a 8 KB zawiera procedury złącza RS 232 C.

Olbrymią zaletą komputera firmy Spectravideo jest posiadanie praktycznie wszystkich możliwych interfejsów i gniazd rozszerzających. Drukarkę można podłączyć przez pełne, 8-bitowe złącze równoległe. 9-igłowe gniazdo interfejsu szeregowego, pozwala na wzajemne łączenie komputerów w sieć i korzystanie z modemów.

Pewną niedogodnością jest nietypowy układ sygnałów obu opisywanych złączy. Z tego powodu na rys. 1 i 2 podano ich opis i kable przejściowe do standardu stosowanego w komputerach IBM PC.

Cechą wyróżniającą model SVI-738, na tle innych komputerów domowych, jest wbudowana stacja dysków 3.5" — jednostronnych, o pojemności 360 KB. Bezproblemowe jest także dołączenie zewnętrznego napędu. Przeznaczone do tego celu 25-stykowe gniazdo Canon DB-25 wykonane jest w standardzie stacji 5.25" typu BW 112, oferowanej przez CSH i nadającej się również do przenośnego komputera Bondwell 8. Na rys. 3 podano opis tego gniazda i sposób dołączenia typowego napędu, wyposażonego w 34-stykowe złącze krawędziowe.

Komputer współpracuje także z dowolnym, popularnym magnetofonem, przyłączanym przez zwykłe, 5-stykowe gniazdo. Przy wysokiej cenie dyski i okresowych problemach z ich nabywaniem, rozwiązanie to może być dość atrakcyjne dla niektórych użytkowników.

Następną zaletą Spectravideo jest możliwość korzystania z odbiornika telewizji kolorowej (PAL). Dwa gniazda joystickowe docenia osoby, które spędzają dużo czasu przy grach. Z kolei przy zastosowaniach bardziej profesjonalnych poważnym plusem komputera jest wyjście wizyjne na monitor zarówno monochromatyczny, jak i kolorowy.

Dobrej jakości klawiatura (72 klawisze), z wyodrębnionym blokiem funkcyjnym i dodatkowymi klawiszami sterowania ruchem kursora, dopełnia pozytywny obraz opisywanego sprzętu.

Niestety, nawet w tak zdawałoby się idealnej konstrukcji, znalazły się pewne błędy i niedopracowania, utrudniające wygodne korzystanie z komputera. Podstawową wadą jest niska jakość generowanych sygnałów wizyjnych. Z powodów, prawdopodobnie ekonomicznych, zastosowano nie najlepsze układy współpracujące (mała pamięć RAM i kiepskie generatory sygnałów), które niweczą działanie doskonałego videoprocatora, wykonane go w standardzie MSX II.

Pierwszym, negatywnym rezultatem takiego rozwiązania jest fatalna jakość obrazu, zwłaszcza w trybie 80-kolumnowym. Dłuższa

praca z popularnym monitorem monochromatycznym, np. w systemie CP/M, staje się szybko męką dla oczu. Jakość obrazów kolorowych, pozornie nieco lepsza, również pozostawia wiele do życzenia. Dalszym skutkiem zastosowanych rozwiązań technicznych jest istotne spowolnienie wyświetlania obrazu, ponieważ wszelkie operacje graficzne wykonywane są programowo, bez sprzętowego wspomaganie.

Tych wad nie posiadał egzemplarz udostępniony redakcji do testowania. Zostały one usunięte przez firmę PC Plus. Przeróbka sprowadzała się do zwiększenia pamięci video-RAM do 128 KB i poprawienia układów generacji sygnałów wizyjnych. Wmontowane zostało także dodatkowe gniazdo do monitora kolorowego wysokiej rozdzielczości. Dodanie przycisku RESET, który nie jest standardowym wyposażeniem, istotnie wpłynęło na zwiększenie komfortu pracy.

Oryginalny komputer Spectravideo pozwala uzyskać na ekranie grafikę 256\*192 punktów w 16 kolorach. Przerobiony model zapewnił rozdzielczość 512\*212 i, co ważne, dostarczone oprogramowanie graficzne działało bardzo szybko.

## SYSTEMY OPERACYJNE

Po włączeniu komputera nadzór nad sprzętem przejmują, wbudowany w pamięć ROM, interpreter języka MSX BASIC. Jest to produkt firmy Microsoft wzorowany, na znanym z komputerów IBM PC, GW BASIC'u. Wbudowane komendy dyskowe, graficzne i dźwiękowe, a także pełnoekranowy edytor zapewniają komfortowe warunki programowania. Na tym poziomie, możemy powiedzieć, że dysponujemy bardzo dobrym, 8-bitowym komputerem domowym. Oprogramowanie to głównie gry, niegdyś dostępne w sklepach CSH, a obecnie na gieldzie, zwanej kiedyś bajtkową.

Jeśli granie znudzi się nam wreszcie, to nie musimy wcale pozbywać się sprzętu, bowiem razem z komputerem nabywamy także dwa profesjonalne systemy operacyjne. Jednym z nich jest MSX DOS, od strony organizacji dyski i poleceń systemowych zgodny z MS DOS'em, z którego korzystamy na IBM PC. Z punktu widzenia budowy wewnętrznej MSX DOS jest bardzo podobny do CP/M'u 2.2 i w dużej mierze z nim kompatybilny.

Drugim systemem operacyjnym jest właśnie wspomniany CP/M 2.2. W wersji dostarczonej przez producenta ma on pewne wady i braki. Są to m.in. niedostępność ramdysku i brak tekstowego i graficznego zrzutu ekranu na drukarkę. Niedogodności te zostały usunięte przez firmę PC Plus, która opracowała także prototypową wersję systemu CP/M 3.0, znanego głównie z komputerów Amstrad CPC i PCW. Na rys. 4 i 5 zamieszczono „wizytówki” systemów CP/M 2.2 i CP/M Plus. Ten ostatni jest, ogólnie biorąc, zdecydowanie lepszy, ale niestety dostarczona nam wersja próbna jest niepełna. Brak zainteresowania tym produktem, wykazany przez oświatę, zniechęcił trochę firmę do dalszych inwestycji w tej dziedzinie.

## OPROGRAMOWANIE

Początkowo oferowano wyłącznie gry, ale dość szybko znalazły się na rynku programy narzędziowe i użytkowe; były to głównie adaptacje oprogramowania CP/M'u z innych komputerów.

Bogato reprezentowane są języki programowania:

- Turbo Pascal, Pascal MT+,
- Microsoft BASIC, Mallard BASIC,
- Kompilator FORTRAN IV — F80,
- Asemblerowy pakiet uruchomieniowy — M80, L80, ZSID, HIST, CREF80 i LIB80.

Na szczególną rolę zasługują Turbo Pascal firmy Borland, zastosowany m.in. przez firmę PC Plus do tworzenia ciekawych programów użytkowych i dydaktycznych na Spectravideo. Dostępna implementacja kompilatora została wzbogacona o bibliotekę graficzną. Pewne jej możliwości demonstruje program DEMO. Na rys. 6 znajduje się graficzny zrzut jednego z wielu przykładowych ekranów.

Bardzo atrakcyjne wydaje się wykorzystanie tej biblioteki przy tworzeniu własnych programów. Na listingu 1 przedstawiono króciutki program stosujący procedurę PLOT. W oparciu o ten schemat uruchomiliśmy na Spectravideo pakiet procedur rysujących linie proste, okręgi i elipsy, który był prezentowany w Bajtku 7,88. Otrzymał krzywe, zwłaszcza koła, były znacznie ładniejsze niż te z programu demonstracyjnego.

Przetestowałem także firmowe procedury baz danych oferowane przez Borlanda w pakiecie Turbo Toolbox. Niestety przykładowy program kartoteki nie działał dobrze. Mam jednak nadzieję, że zainwestowanie paru godzin w uruchomienie tego pakietu udostępniłoby

użytkownikom tego komputera doskonałe narzędzie do tworzenia baz danych, znacznie lepszy od tych jakie oferuje dBase II.

CP/M dla Spectravideo sprzedawany jest bez GSX'a (ang. Graphics System extension), w szczególności nie ma programów sterujących (ang. drivers) dla ekranu i pozostałych urządzeń zewnętrznych (drukarki, plotery). Próba uruchomienia GSX'a przeniesionego z Amstrada PCW powiodła się tylko częściowo. Możliwe było wykonanie rysunków na drukarce, ale nie można było sporządzić żadnych napisów. Prawdopodobnie plik DDFXHR8.PRL, obsługujący drukarkę, korzysta z komputerowego generatora znaków ekranu, który w komputerach Amstrad i Spectravideo, znajduje się w innym miejscu. Podobnie, jak i w przypadku toolbox'a do baz danych, sensowne wydaje się zajęcie tym problemem. Standard GSX'a, znany także na IBM PC, wspierany jest przez wiele gotowych programów użytkowych, m.in. przez pakiety DR DRAW i DR GRAPH.

Na Spectravideo dostępny jest też uniwersalny edytor dyskowy HEAVY, wzorowany na słynnym Norton Utilities. Umożliwia on oglądanie i modyfikację dowolnego sektora dyskowego. Ponadto pozwala on na odzyskanie przypadkowo skasowanych plików, podając także ich rozmieszczenie na dyskiecie. Program został napisany całkowicie w Turbo Pascalu przez pana Pawła Ciężkiego. Oferowane są również wersje na inne komputery — Amstrad, Junior. Program HEAVY sprzedawany jest łącznie z programami PCFORMAT i COPY, pozwalającymi formatować dyski obu napędów i kopiować pliki przy wykorzystaniu tylko jednej stacji, z czym nie radzi sobie standardowy PIP.COM.

Razem z komputerem otrzymaliśmy także prototypową wersję XTREE — innego programu znanego z IBM PC. Nie była to pełna implementacja, ale dostępne opcje przeglądania plików tekstowych i binarnych zachwycały szybkością działania. Jest to zasługa specjalizowanego videoprocatora. Mam nadzieję, że obiecana przez PC Plus, wersja CHIWRITERa (edytor tekstowy pracujący w trybie graficznym) na komputer Spectravideo będzie miała parametry nie gorsze niż oryginał. Obecnie najpopularniejszym edytorem na SVI-738 jest WordStar.

Mieliśmy również okazję obejrzeć kilka programów dydaktycznych. Jednym z nich było Liczydło, którego idea działania została zaczerpnięta z programu Eureka firmy Borland. Liczydło wspomaga naukę matematyki na poziomie szkoły średniej, pozwalając na rozwiązywanie dowolnych równań z jedną niewiadomą oraz na sporządzanie wykresów funkcji jednej zmiennej. Obróbkę danych uzyskanych w eksperymentach fizycznych lub chemicznych ułatwia program MINISTAT, sprzedawany w wersjach wysokiej i niskiej rozdzielczości. Umożliwia on dopasowanie wyników do zadanej krzywej lub też wykonanie histogramów. Oba wymienione programy współpracują z typowymi drukarkami. Zaufajmy, że nie mogliśmy obejrzeć w działaniu programu Gwiazda, który jest aktualnie w realizacji. Zawiera on informacje o ponad 8000 gwiazd i będzie doskonałą pomocą na lekcjach astronomii.

## WYMIANA INFORMACJI Z INNYMI KOMPUTERAMI

Dzięki wyposażeniu w złącze szeregowe i gniazdo zewnętrznej stacji dysków 5.25", komputer Spectravideo SVI-738 jest dobrze przygotowany do wymiany programów i danych z innymi komputerami.

W przypadku niewielkich potrzeb w tej dziedzinie, możemy posłużyć się interfejsem RS 232C. Do transferu plików tekstowych w starszący okazał się standardowy program PIP.COM. Wykonanie polecenia `PIP PUN:=nazwa_zbioru, wysyla_zbior a inny komputer, a polecenie PIP nazwa_zbioru =PTR: odbiera go`. Bardziej niezawodny i wygodniejszy jest program KERMIT, który otrzymaliśmy w dwóch wersjach, jedną na Spectravideo, a drugą na IBM PC. Dzięki temu możliwe było przesyłanie z szybkością 9600 bps dowolnych zbiorów między tymi dwoma komputerami. KERMIT jest znanym na całym świecie programem komunikacyjnym, stosowanym głównie na dużych maszynach, ale posiadającym bardzo udane implementacje również na sprzęcie klasy mikro.

Jeśli musimy częściej przenosić pliki między różnymi komputerami, to znacznie wygodniejsze jest zastosowanie stacji dysków. System operacyjny MSX DOS jest zgodny z MS DOS'em jeśli chodzi o strukturę logiczną dysku. Bez żadnych problemów można przesyłać informacje na dyskietkach 5.25". Napotkamy na pewne problemy jeśli będziemy chcieli skorzystać z wbudowanej, jednostron-

nej stacji 3.5". W dalszym ciągu jest to rzadkie wyposażenie komputerów IBM PC dostępnych na naszym rynku. Poza tym dyskietka MSX-DOS'a nie była rozpoznawana przez IBM PC. Konieczne było sformatowanie jednostronne dyskietki na IBM'ie.

Logiczną organizacją dyskietki CP/M'u jest inna niż MSX DOS'a, ale przenoszenie plików między obydwojema systemami umożliwia firmowy program FILECOPY. Podłączenie napędu 3" firmy Hitachi, stosowanego w stacji dysków TIMEX'a do komputera ZX Spectrum, pozwoliło na bezpośrednią wymianę zbiorów między Amstradem PCW (CPC), a Spectravideo. Dyskietkę do transferu należało jednostronnie sformatować na Amstradzie posługując się własnym programem formatującym. Parametry formatu obu stacji dysków na Spectravideo w CP/M'ie 2.2 zebrano w TAB 2. Informacje o obsłudze obcych dyskietek na komputerach Amstrad zostały podane w Bajtku 1,90.

## PODSUMOWANIE

Spectravideo jest starannie i dobrze opracowaną konstrukcją, której niewielkie wady zostały w dużej części usunięte przez firmę PC Plus. Dzięki bogatemu wyposażeniu we wszelkie możliwe interfejsy i dobremu oprogramowaniu jest komputerem, który można polecić szerokiemu gronu użytkowników.

Otrzymał pełnowartościowy zestaw „rekreacyjnego” jest proste i wymaga tylko dołączenia odbiornika telewizyjnego i joysticka. Porównanie na tym poziomie z popularnymi na naszym rynku komputerami Atari, ZX Spectrum czy Commodore C64 wypada na korzyść Spectravideo.

Przygotowanie zestawu „profesjonalnego” nie wiąże się z żadnymi zmianami w samym komputerze, nie musimy kupować dodatkowych interfejsów, aby przyłączyć modem, monitor, drugą stację dysków, czy drukarkę. Możliwość pracy w systemie CP/M, nawet w wersji 2.2, pozwala na porównanie Spectravideo z 8-bitowymi komputerami firmy Amstrad. Przewaga większej pamięci (128, 256, KB) tych ostatnich może być zniwelowana nabyciem dodatkowego Ramdysku (64-512KB) do SVI-738. Zaletą jest natomiast łatwość wymiany zbiorów z IBM PC i wyposażenie w złącze RS 232 C, dokupienie którego do Amstrada jest dość kosztowne.

Na zakończenie chcę stwierdzić, że komputer Spectravideo SVI-738 jest kolejnym bardzo ciekawym i dobrym produktem oferowanym od dłuższego czasu przez Centralną Składnicę Harcerską. Jak w przypadku plotera Sony i stacji Time'a do ZX Spectrum, dziwi polityka handlowa sprzedawcy.

Taktyką CSH jest początkowe ustalenie wysokich, zaporowych cen na sprzęt i brak jakiegokolwiek promocji, czy reklamy. Trzy lata temu komputer Spectravideo był sprzedawany za 560 tys. zł. Za równowartość tej sumy w dolarach można było wtedy myśleć o zakupie prostej konfiguracji tajwańskiego kłona klasy IBM PC. Z kolei stała i szybka inflacja powoduje, że po pewnym czasie ceny w Składnicy stają się bardzo atrakcyjne i zakup czegokolwiek jest po prostu wielką okazją. Tak było i w przypadku Spectravideo, którego pojedyncze egzemplarze można było nabyć w grudniu 1989 w sklepie na Marszałkowskiej za milion złotych. W tym samym czasie i w tym samym stoisku sprzedawano małe Atari z magnetofonem za 1 680 tys. zł. Wniosek jest prosty: mając dobrą reklamę możemy sprzedać znacznie gorszy sprzęt znacznie drożej. Kto na tym trafi to już inna sprawa.

Osobiście żałuję, że tak doskonały sprzęt jakim jest Spectravideo nie przyjął się szerzej w naszym kraju. Abstrahując zupełnie od tego, że jest to komputer standardu MSX o parametrach przewyższających popularne Spectrumy, czy Atari. Dla mnie jest to bardzo dobry sprzęt półprofesjonalny o olbrzymich walorach dydaktycznych, który mógł zdominować nasze szkolnictwo w miejsce nieudanych i skąpo oferowanych Juniorów.

Na szczególne podkreślenie zasługuje też rola niezależnych producentów oprogramowania, co czego praktycznie nie mamy na naszym rynku komputerowym. „Poważne” firmy handlujące głównie sprzętem 16-bitowym w dalszym ciągu sprzedają komputery jak kartofle, zapominając o oprogramowaniu i szkoleniu użytkowników. W przypadku Spectravideo było inaczej. W niedługim czasie po pojawieniu się tego komputera na naszym rynku, firma PC Plus przedstawiła szeroką ofertę zarówno programów, jak i usprawnień sprzętowych. Uważamy, że zjawisko takie, będące odstępstwem od reguły i rzadkie w naszym kraju, godne jest pochwały i powinno ulec upowszechnieniu.

**JONASZ MAYER**

**Testowany komputer łącznie z monitorem kolorowym i oprogramowaniem został nam udostępniony przez pana Mirosława Sobka, reprezentującego firmę PC PLUS, 91-160 Łódź, ul. Mencla 44, tel. 55-75-75. Dziękujemy.**



```
($I Graphic.lib )
var
  i : integer;
  ch : char;
begin
  InitGr(0);
  for i := 10 to 100 do plot (i,i);
  repeat until keypressed; read (kbd,ch);
  EndGr;
end.
```

Krótki program pascalo-  
wy korzystający z biblio-  
teki procedur graficznych  
PC Graf.

Tab. 1. SpectraVideo SVI—738 (dane techniczne)

**Procesory:**  
procesor Z80, zegar 3.58 MHz,  
videoprocessor MSX-VDP (Yamaha 9038)

**Pamięć RAM:**  
użytkownika — 64 KB,  
video — 16 KB (64-128KB) \*  
opcjonalnie: ramdysk — 64 KB (512KB) \*,

**Pamięć ROM:**  
MSX BASIC — 32 KB,  
RS 232C Firmware — 8 KB,  
Disk Firmware — 16 KB,

**Pamięci dyskowe:**  
Napęd 3.5", jednostronny, 80 ścieżek, 9 sektorów na ścieżkę, po-  
jemność 360 KB,  
opcjonalnie: napęd 5.25", dwustronny, 40 ścieżek, 9 sektorów na  
ścieżkę, pojemność 360 KB.

**Ekran:**  
tryb tekstowy: 40\*24, 80\*26.5  
tryb graficzny: 256\*192 (16 kolorów)  
512\*212 (16 kolorów) \*

**Klawiatura:**  
72 klawisze, wydzielony blok funkcyjny i blok sterowania kursorem.

**Dodatkowe złącza:**  
interfejs szeregowy RS 232C,  
interfejs równoległy Centronics,  
50-stykowa szyna systemowa standardu MSX,  
gniazdo drugiego napędu,  
2 gniazda joysticków,  
gniazdo do magnetofonu,  
wyjście na odbiornik TV (PAL),  
wyjście na monitor,  
wyjście sygnału akustycznego,  
dodatkowe wyjście na monitor kolorowy \*

**Systemy operacyjne:**

MSX DISK BASIC  
MSX-DOS  
CP/M 2.2  
CP/M 3.0\*

\* — rozszerzenia i produkty oferowane przez firmę PC PLUS.

Spectravideo SVI - 738 (Amphenol 14)	Canon DB-25
1 STROBE	o ----- o 1
2 DATA 0	o ----- o 2
3 DATA 1	o ----- o 3
4 DATA 2	o ----- o 4
5 DATA 3	o ----- o 5
6 DATA 4	o ----- o 6
7 DATA 5	o ----- o 7
8 DATA 6	o ----- o 8
9 DATA 7	o ----- o 9
10 N/C	o ----- o
11 BUSY	o ----- o 11
12 N/C	o ----- o
13 N/C	o ----- o
14 GND	o ----- o 18-25

Rys. 1 Opis kabla przejściowego z 14 stykowego  
gniazda Amphenol do typowego gniazda Canon DB-25  
stosowanego w komputerze IBM PC dla złącza Centronics.  
(Apostrof oznacza negację sygnału).

Spectravideo SVI - 738 (Canon DB-9,	Canon DB-25
1 GND (PROTECTIVE GROUND)	o ----- o 1
2 TX (TRANSMIT DATA)	o ----- o 2
3 RX (RECEIVE DATA)	o ----- o 3
4 RTS (REQUEST TO SEND)	o ----- o 4
5 CTS (CLEAR TO SEND)	o ----- o 5
6 DSR (DATA SET READY)	o ----- o 6
7 GND (SIGNAL GROUND)	o ----- o 7
8 DCD (DATA CARRIER DETECT)	o ----- o 8
9 DTR (DATA TERMINAL READY)	o ----- o 20

Rys. 2. Opis złącza szeregowego komputera Spectravideo  
SVI-738. Po prawej stronie podano opis przejściowego  
kabla do standardowego gniazda DB-25 stosowanego w IBM PC.

Spectravideo SVI-738 (Canon DB 25)	Stacja dysków (złącze krawędziowe) (34-stykowe)
1 +12V	o
2 +5V	o
3 +5V	o
4 INDEX	o ----- o 8
5 DRIVE 1	o ----- o 12
6 DIRECTION	o ----- o 6
7 STEP	o ----- o 20
8 WRITE DATA	o ----- o 22
9 WRITE GATE	o ----- o 24
10 TRACK 00	o ----- o 26
11 WRITE PROTECT	o ----- o 28
12 READ DATA	o ----- o 30
13 SIDE SELECT	o ----- o 32
14 +12V	o
15 +12V	o
16 +5V	o
17 DRIVE 0	o ----- o 10
18 MOTOR ON	o ----- o 16
19 READY	o ----- o 34
20-25 GND	o ----- o 7,9,...33

Rys. 3. Opis złącza zewnętrznej stacji dysków  
komputera Spectravideo. Opisano także połączenie  
komputera z typowym napędem.

Tab. 2. Tablice DPB napędów dyskowych systemu CP/M 2.2 komputera  
Spectravideo SVI-738

napęd [KB]	adres DPB	LSPT 0	BSH 1	BLM 2	EXM 3	4	DSM 5	6	DRM 7	8	ALD 9-10	CKS 11	12	OFF 13	14
A:[346]	E294	36 00	04	15	01	172 00	63 00	\$B000	16 00	03 00					
B:[358]	E2A3	36 00	04	15	01	178 00	63 00	\$C000	32 00	00 00					
C:[63]	E2B2	32 00	03	07	00	62 00	31 00	\$B000	16 00	00 00					

Uwaga. Opis pól tablic DPB w artykule "Operacje dyskowe", Bajtek  
1,90. W nawiasach kwadratowych podano pojemność dysku przeznaczoną  
na pliki użytkownika.

SPECTRAVIDEO CP/M Revision 2.28 6  
PC plus GRAPHICAL SUPPORT  
Print Screen = SHIFT+COPY  
International keyboard

A>

Rys. 4. Wizytówka systemu CPM w  
wersji adaptowanej przez firmę PC  
Plus.

SPECTRAVIDEO CP/M 3+  
Implemented by Dominik Research  
PC plus GRAPHICAL SUPPORT  
Print Screen = SHIFT+COPY

BIOS3 SPR E600 1400  
BDOS3 SPR C500 2100

48k TPA

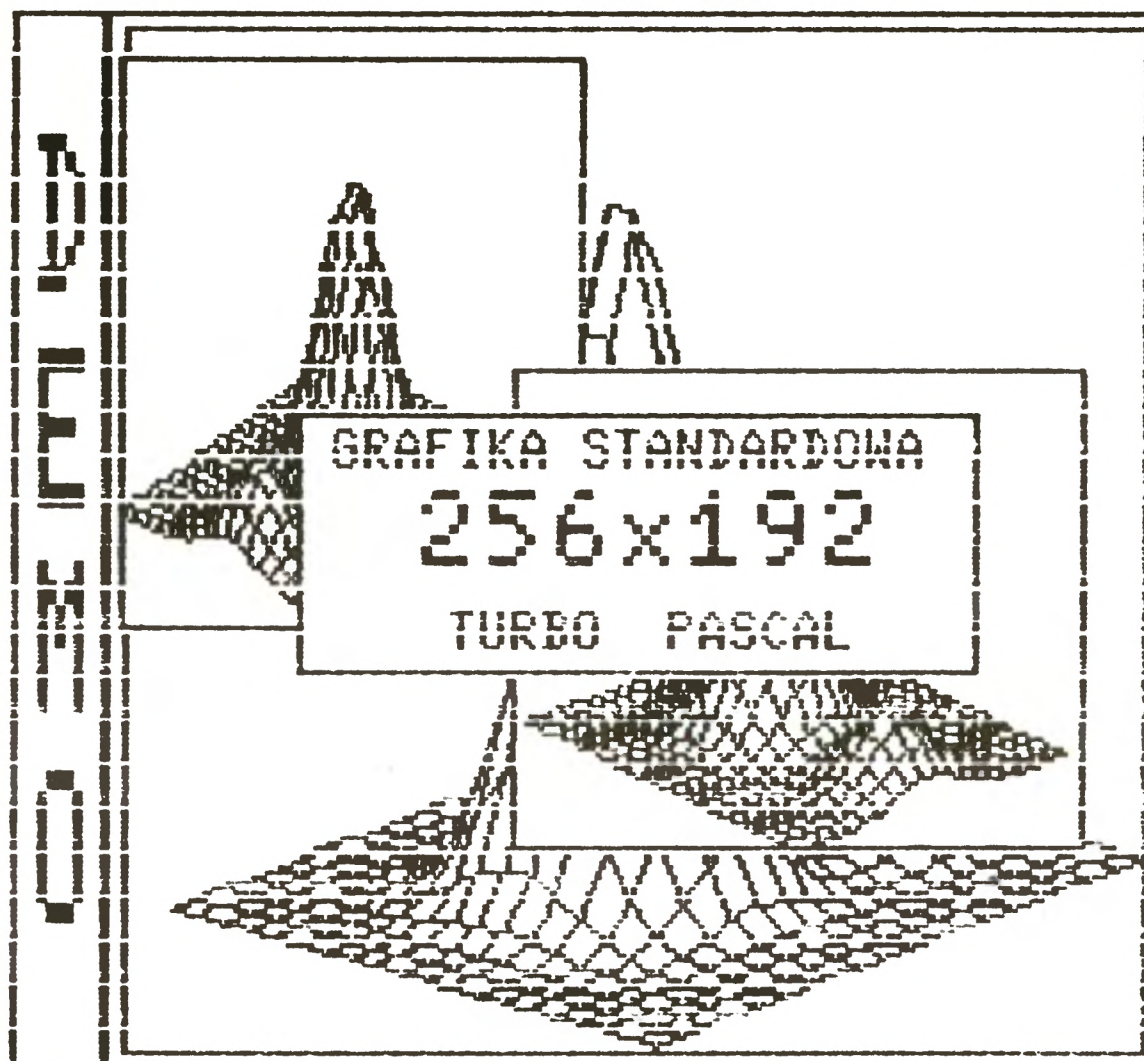
A>setdef \*,a: [order=(com,sub)]

Drive Search Path:  
1st Drive - Default  
2nd Drive - A:

Search Order - COM, SUB

A>

Rys. 5. Wizytówka prototypowej wer-  
sji systemu CP/M Plus adaptowanego  
przez firmę PC Plus.



Rys. 6. Ekran programu demonstrującego możliwości graficzne pascalogo  
pakietu PC Graf.

## Zalety

- + wbudowana stacja dys-  
ków
- + pełne wyposażenie w in-  
terface'y
- + otwartość konstrukcji
- + możliwość pracy w sy-  
stemie CP/M,
- + przenaszalność plików  
dyskowych między Spec-  
travideo a IBM PC.
- + dobry interpreter języka  
MSX BASIC
- + obszerna dokumentacja

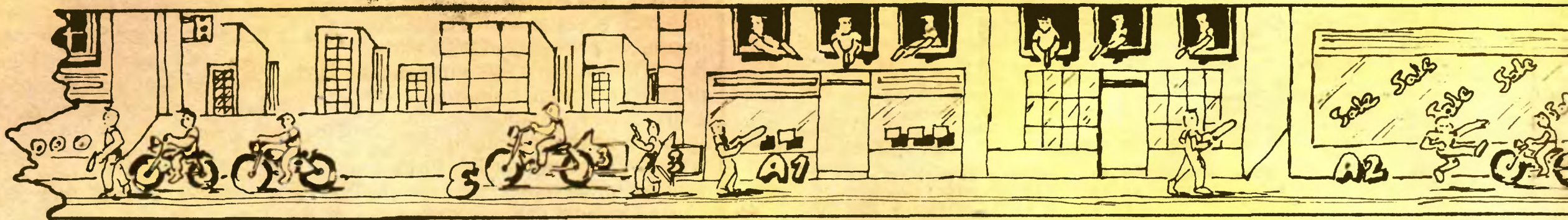
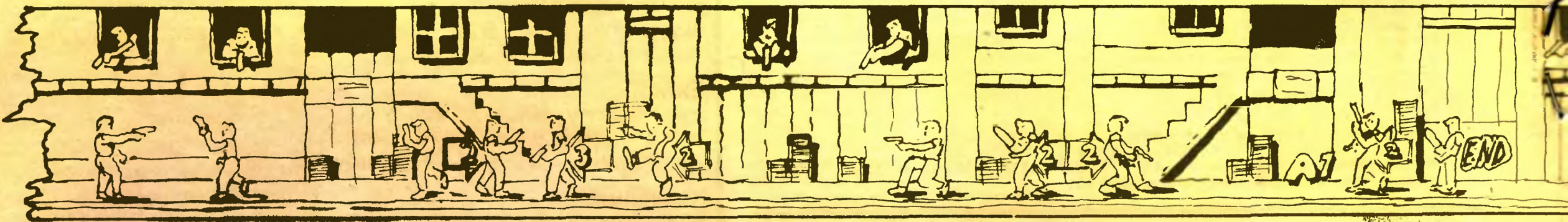
## Wady

- niska jakość sygnałów  
wizyjnych oryginalnego  
sprzętu
- głośna praca napędu dy-  
skowego
- silnie grzejący się zasil-  
acz
- brak przycisku RESET





# ROBOCOP



Szary, stalowy kolos bezszelstnie sunie wąską uliczką. Połyskujący w dłoni pistolet nie pozostawia wątpliwości co do zamiarów dzierżącej go postaci. Już tylko trzy metry dzieli ją od celu. Za chwilę uderzy.

Drewniane drzwi rozpryskują się w drzazgi po uderzeniu równoczesnym ze strzałem w zamek. Stalowa postać w mgnieniu oka wskakuje w powstały otwór. Tam oddaje dwa strzały i na podłogę padają dwa trupy z przestrzelonymi czaszkami. Tak chciało prawo. Robota, skończona, jej wykonawca wraca do samochodu.

Widok rozbryzgujących się na ścianach mózgow nie wzrusza mordercy. Ale pod stalową powłoką bije żywe, ludzkie serce. Nosi je w piersi Superglina — pół robot, pół człowiek. RoboCop.

Z robo-gliniarzem Murphym, kreowanym przez Petera Weller na ekranach kin. Z radością witają go ulubieńcy jako bohatera gry komputerowej. Jest on w niej równie zręczny, co w filmie, a i zniszczenie niesie niemałe.

Zaczynasz, gdy RoboCop udaje się na patrol. Nie jest to zabawa ani ściganie małałów, lecz walka z zastępami karateków, wariatów, snajperów i całego podziemnego śmiecia. On sam przeciwko tej gromadzie nie znaczy wiele, lecz kierowany pewną ręką może ich znacznie przetrzebić. Pistolet, karabin i automat nie ostygną ani na chwilę.

Na patrolu napotykasz zamaskowanego, uzbrojonego mężczyznę, który trzyma przed sobą kobietę — zakładniczkę. Na filmie RoboCop uwolnił ją strzelając między jej nogami, Ty strzel obok jej głowy tak, by trafiać mężczyznę.

Kontynuujesz patrol. Nic ciekawego, tylko amunicja kończy się szybko (znalezione zasobniki pozwalają ją uzupełnić, w razie czego zawsze pozostaje pięć).

W tej części patrolu spotykasz niejakiego Emila, mordercę policjanta, którego serce bije w Twojej stalowej piersi. Aby się upewnić, czy to on, udaj się na komisariat i tam zidentyfikuj go przy pomocy maszyny do portretów pamięciowych.

Udało Ci się? Jeśli tak, to podążaj za Emilem do fabryki. W samym jej centrum spotkasz się oko w oko z robotem ED209, nieudanym wytworem inżyniera, który zaufał elektronice. Przedtem jednak dowiesz się, kto jest szefem gangu przemytników. Tak, to Clarence Botticker. Wiedząc to, musisz udać się do siedziby koncernu OCP, jeśli przepuści Cię ED209.

Tam złóż informacje, wyjdź i czeka Cię już tylko przeprawa z Clarence. Gdy go zabijesz (co nie będzie łatwe), uwolnisz szefa OCP, który sowicie Cię wynagrodzi.

Nie ludź się, nie skończysz tej gry tak szybko, jak jest to tu przedstawione. Być może nie skończysz jej w ogóle. Ale głowa do góry! Już wkrótce film RoboCop II, w którym Superglina pokaże co potrafi!

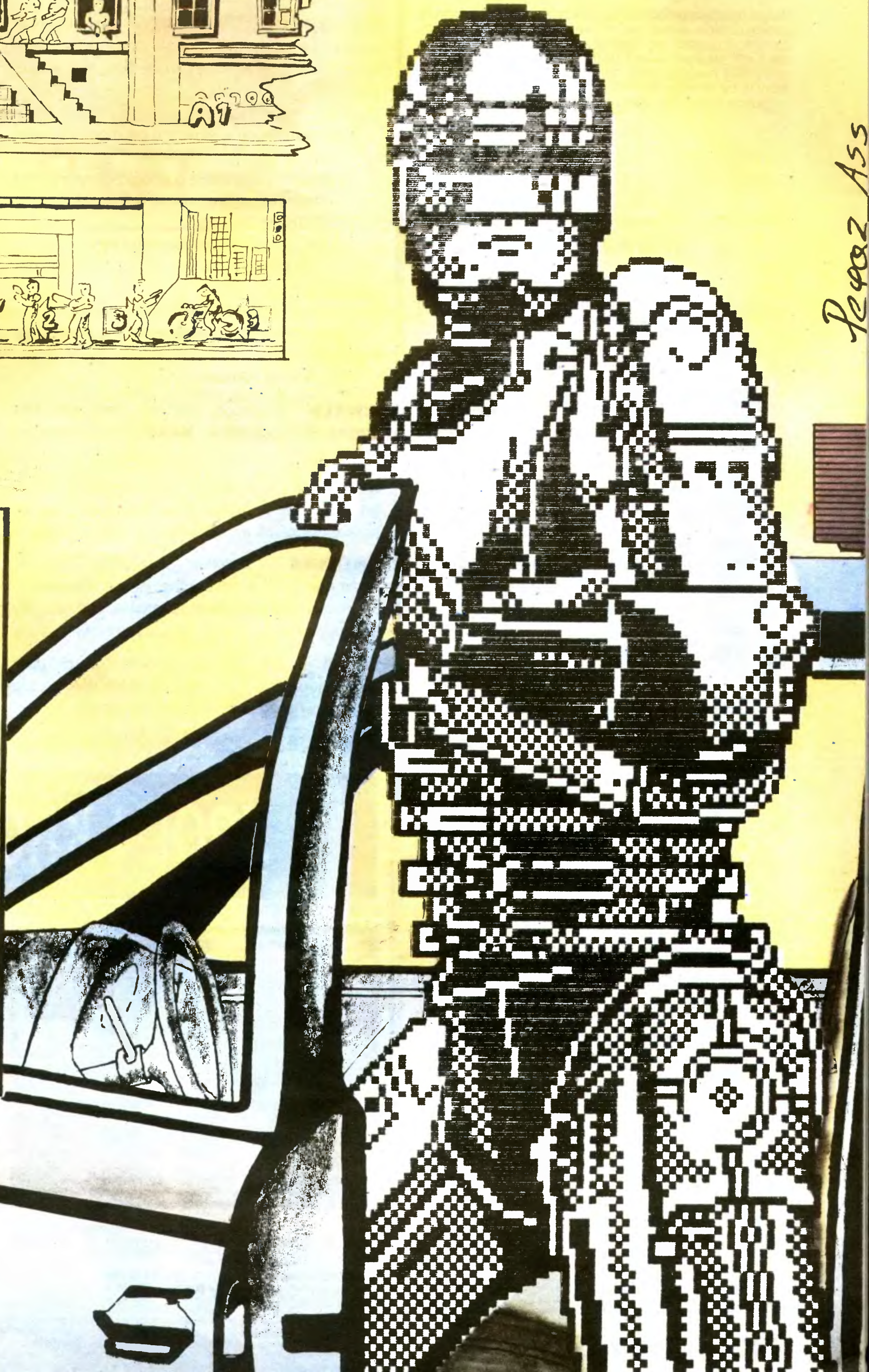
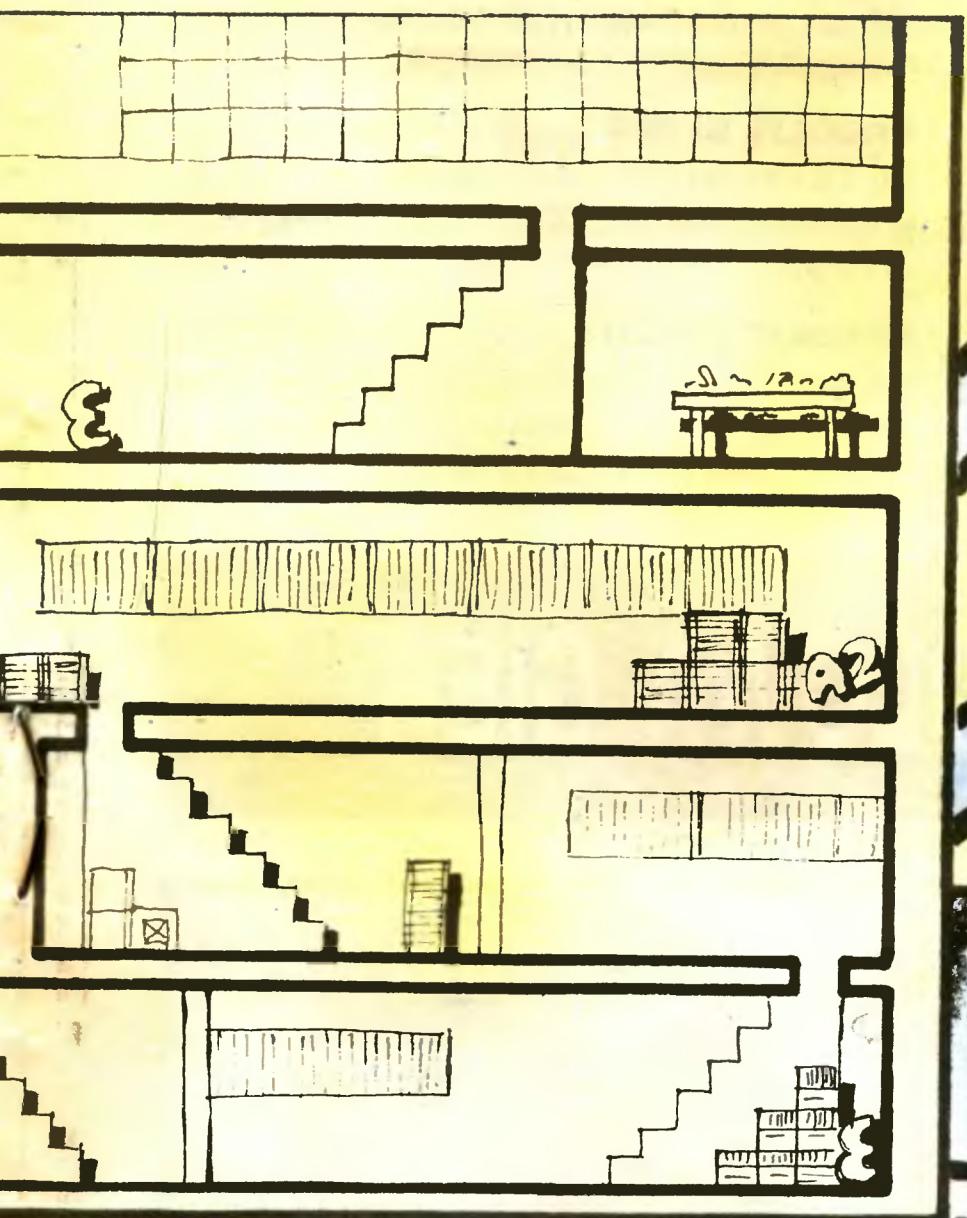
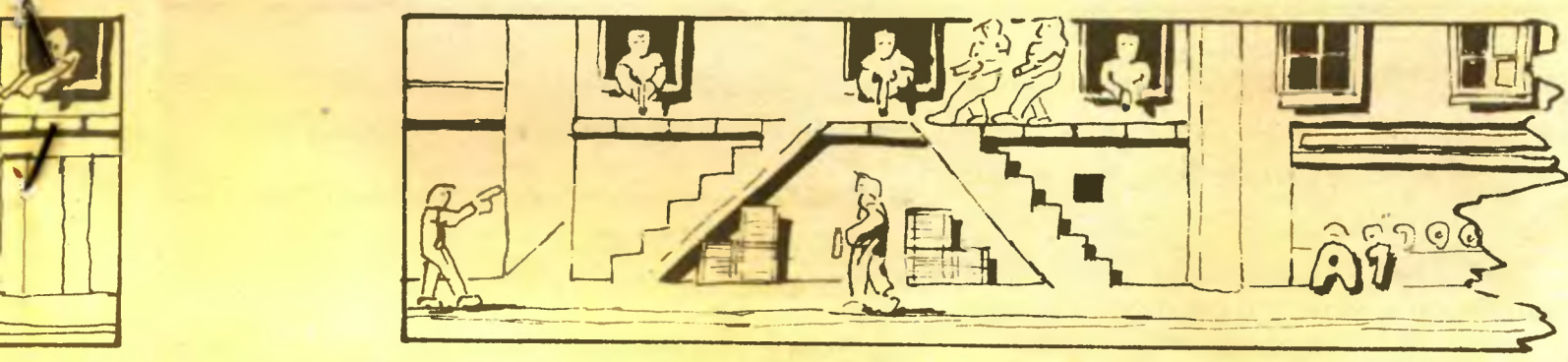
Mapa, którą obok widzicie to tylko niektóre etapy tej gry. Z uwagi na brak miejsca zamieściliśmy rzeczy najpotrzebniejsze. Napewno dacie sobie radę z resztą.

**Firma: Ocean**  
**Komputer: ZX Spectrum**  
**48/129, Commodore 64,**  
**Amstrad CPC**





# BOOOOP



*Revaz Ass*



# 10

BAJTKOWA LISTA PRZEBOJÓW 3-4

W tym miesiącu tendencja średnio zniżkowa. Tylko Indiana Jones dzielnie wspiął się na pierwsze miejsce. Niespodziewanie pojawiła się kilkuczęściowa gra War in the Middle Earth, na podstawie Trylogii Tolkiena. Producenci poszli w niej znacznie dalej niż było to z Lord of the Rings, który to był śmiertelnie nudny i denerwujący.  
Głosów było 1730, oddanych na 81 tytułów.

	ATARI	AMSTRAD	COMMODORE	SPECTRUM
1 INDIANA JONES	>	x	x	x
2 BATMAN	>	<	x	x
3 HOSTAGES	>	x	x	
4 RED HEAT	>	<	x	x
5 WAR IN THE MIDDLE EARTH	>	x	x	<
6 ROBBO	<			
7 4x4 OFF-ROAD RACING	>	x	x	<
8 CABAL	>	<	x	<
9 TOTAL ECLIPSE	>	<	x	<
10 ROBOCOP	>	x	x	x



? NOT	YES NO	HOWDY SEE YOU	GO WANT	TELEPORT GIVE	LIKE SAY	KNOW STRANGER	PLAY SEARCH	RACE VOTE
HELP DISARM	LAUGH SOB	FEAR DESTROY	FREE KILL	PRISON PRISONER	TRAP DANGER	FORBIDDEN RADIOACTIVITY	IMPOSSIBLE BOUNTY	INFORMATION JOKE
RENDEZ-VOUZ TIME	URGENT IDEA	MISSILE CODE	FRIEND ENEMY	MIND BRAIN	WARRIOR PRESIDENT	SCIENTIST GENETIC	SEX REPRODUCTION	MALE FEMALE
IDENTITY POP	RACE DIFFERENT	SMALL GREAT	STRONG BAD	BRAVE GOOD LOOKING	BANANAS POOR	INSULT CURSE	PEACE DEAD	CORXX TROMP
KINGPAK ROB HEAD	CROOLIS VAR CROOLIS ULV	IZWAL MIGRAX	ANTENNA BUGGOL	TRICEPHAL TUBULAR BRAIN	YUKAS SINOX	ONDOYANTE NUMBER	TUTTLE MORLOCK	YOKO MAXON
BLOOD TORKA	SHIP CONTACT	HOME PLANET	TRAUMA ENTRAX	ONDOYA KRISTO	ROSKO CORDO	ULKIAN BOW-BOW	HOOR CO-ORDINATE	= /
0	2	4	6	8	DICTIONARY			
1	3	5	7	9				

## BLOOD, CZYLI KRWAWE

**My spenetrowaliśmy** już galaktykę wzdłuż i wszerz. Nie było to łatwe. Teraz więc chcemy pomóc tym, którzy wciąż tkwią zawieszonymi między nicością a pustką.

- Oto informacje:
- Niebieskie ikony w słowniku nie mogą być użyte,
  - By otrzymać informacje o współrzędnych planet innych stworów, używaj m.in. zwrotu „CODE INFORMATION HELP”.
  - Nie zabijaj stworów, oprócz Numbers
  - Postrasż Obcego, gdy nie możesz się z nim dogadać.
  - Odwiedź wszystkie zamieszkane planety.
  - Przewoź obcych na inne planety, by dowiedzieć się więcej.

Obcy, których spotkasz:

**IZWALS** — to głupek, choć zna wiele słów. Mądrzejszy jest jego ojciec, **MAXON**. Pytaj o kod Sinox.

**KINGPAK** — to głupek niewykształcony. Chce tylko tego, byś go gdzieś przewiózł. Uciekaj, jak tylko poda Ci jakieś współrzędne.

**ANTENNA** — zupełnie nieużyteczny stworek. Może pomóc Ci w odszukaniu innych, ale dopiero po długiej, bezsensownej rozmowie. Boi się, gdy mu pogrozić.

**SINOX** — jeden z mądrzejszych Obcych. Jeśli podasz mu hasło otrzymane od **MAXONA**, będzie miły i da Ci dużo pożytecznych informacji.

**NUMBERS** — wyciśnij go jak gąbkę. Gada jak na-

jęty, trzeba tylko umiejętnie poprowadzić rozmowę. Możesz zniszczyć jego planetę, on zdąży uciec.

**BUGGOL** — inteligentny, ale nieufny. Poprosi Cię o poparcie **MORLOCKA** — zgódź się. Nie mów nic o **YUKASIE**, nienawidzą się.

**MIGRAX** — również inteligentna bestia. Chce, byś go woził. Gdy spotka którąś z sióstr **ONDOYANTE**, chce jechać na planetę Mind.

**ROBHEADS** — niegłupi, ale lekkomyślny. Da Ci informacje, jeśli znajdziesz dla niego inżyniera genetycznego, który pomoże mu.

**TORKA ONDOYANTE** — szuka swego przyjaciela, **TUBULAR BRAINA**. Chce też odbyć z Tobą proces reprodukcji...

**TRAUMA ONDOYANTE** — siostra **TORKI**. Nienawidzi Cię, więc przedstaw się jako **TORKA**. Gada głupoty, trudno ją zmusić do wykrztuszenia czegoś z sensem.

**TUBULAR BRAIN** — z początku gada liczbami. Gdy go „wyczekasz”, zacznie mówić z sensem. Nie wspominaj przy nim o **TORCE**.

**CROOLIS ULVES** — coś w rodzaju antypacyfisty. Lubi się przedrzeźniać i powtarza po Tobie. Nie przedstawiaj się jako **CROOLIS VAREUX**, nienawidzi go.

**CROOLIS VAREUX** — nie zbliżaj się na odległość strzału! Nie wypowie ani jednego sensownego zdania, za to jest bojowy. Dostaje drgawek na dźwięk nazwiska **CROOLIS ULVES**. Gen

## KRÓL I KRÓLOWA GIER



**Ewa Borkowska**, lat 14, uczennica kl. VIII szk. podst. nr 203 w Warszawie  
zainteresowanie: muzyka, języki obce  
komputer: Commodore 64



**Michał Janczak**, lat 12, uczeń kl. IV szk. podst. nr 30 w W-wie  
komputer: Toshiba  
gry: King Quest, Gato,  
hobby: gra na pianinie, angielski

Kopertę nadał Konrad Guzowski z Białegostoku





## STRIKE FLEET

**PHM Pegasus** to znany, przestarzały okręt raketowy, zatapiany po najłabszym nawet trafieniu. Strike Fleet, to dokładne przeciwieństwo PHM Pegasus. Doskonała muzyka i oprawa graficzna, animacja na poziomie Amigi, wielka różnorodność akcji i okrętów stawiają ją bardzo wysoko w hierarchii gier symulacyjnych. Każda chwila spędzona przy tej grze dostarcza niezapomnianych wrażeń oraz dobrej zabawy.

Tym razem masz do wyboru dziesięć scenariuszy, które przygotowują Cię do prawdziwej walki:

— **STARK REALITIES** — patrol po wyznaczonym akwenu, strzelać tylko w samoobronie.

— **ENEMY BELOW** — zniszczyć dwie wrogie łodzie podwodne.

— **ROAD TO KUWEJT** — eskortowanie konwoju tankowców.

— **FALKLANDS DEFENSE** — zlokalizować 79.4 TASK GRUP i zniszczyć ją.

— **DIRE STRAITS** — eskortowanie konwoju tankowców wraz z okrętami zaopatrzenia.

— **OPERATION CORK** — nie dopuścić wrogich jednostek do Atlantyku.

— **SURPRISE INVASION** — trzymać wrogie wojska desantowe z dala od wyspy TRONDHEIM.

— **ESCAPE TO NEW YORK** — ucieczka przed wrogimi wojskami. Kieruj się całą szybkością do Nowego Jorku.

— **WOLFPACK 1990** — eskortowanie piechoty i statków zaopatrzenia na wyspę.

— **MOPPING UP** — zniszczyć wrogie jednostki nawodne oraz podwodne, zanim zdążą dobroić się i nabrać paliwa na wyspie-bazie.

W każdej misji masz do czynienia z innymi typami okrętów, które w sumie formują konwój dowodzony przez Ciebie. Różnią się one przede wszystkim niezatapialnością, uzbrojeniem i szybkością. Z reguły statki poruszają się z szybkością maksymalną 20—34 węzłów, przy czym konwój płynie zawsze z prędkością najwolniejszej jednostki. Niezatapialność zależy zawsze od wielkości i wyporności statku.

Na uzbrojenie składają się:

- 1 — rakiety woda-woda średniego zasięgu (TOMAHAWK)
- 2 — rakiety woda-woda krótkiego zasięgu (HARPOON)
- 3 — rakiety woda-powietrze, przeznaczone do niszczenia obcych rakiet (SMIMA)
- 4 — działo 50 mm, przeznaczone do ostrzeliwania obcych okrętów
- 5 — torpedy
- 6 — torpeda-bomby. Specjalna broń do niszczenia okrętów podwodnych.
- 7 — laser (PHL BURST). Działający na odległości do 2000 m, przeznaczony do niszczenia wrogich rakiet.
- 8 — jeden lub dwa helioplany do niszczenia okrętów podwodnych.

Ostrzeliwanie innych jednostek nie jest zbyt skomplikowane. Należy zbliżyć się do okrętu na odległość radarową lub sonarową i wcisnąć klawisz T (TARGET). Na górze ekranu ukaże się sylwetka statku. Teraz za pomocą klawisza W (WEAPON) ustaw odpowiednią rakietę i wciśnij spację. Jeżeli natomiast chcesz strzelać z dział, wciśnij G (GUN) i dobrze mierząc zakończ epopeję wrogiego okrętu. Pamiętaj jednak, że każda z broni ma ograniczony zasięg działania.

**Inne klawisze:**

- 1-4,0** = prędkość okrętu  
**+, -** = skala czasu, od 1 do 128  
**Q** = koniec misji  
**R** = zasięg radaru/sonaru, od 2 km do 256 km  
**U** = sonar włączony/wyłączony  
**I** = radar włączony/wyłączony  
**P** = zatrzymanie gry

- A** = sterowanie automatyczne/ręczne  
**S** = załoga odpoczywa/gotowa do akcji  
**H** = start helikoptera  
**X** = uaktywniony radar/sonar  
**C** = zmiana statku, którym chcesz kierować  
**U** = mapa akwenu  
**,** = obracanie statkiem

**Firma: Lucasfilm Games**  
**Komputer: Commodore 64, IBM PC**

# S.O.S.

Poszukuję nieśmiertelności do gier COMMANDO, ENDURO RACER, BOMB JACK, ALIENS na Spectrum. Proszę także o przysłanie mi gier BARBARIAN, WEST BANK, KUNG-FU MASTERS i COBRA. W zamian oferuję inne gry.

**Bartek Brańka, ul. T. Boya-Żeleńskiego 14m11 85-858 Bydgoszcz**

Posiadam Spectrum. Nie wiem jak w grze TERRAMEX wziąć tabletki antyradiacyjne. Proszę, pomóżcie!

**Ola Klisiński, ul. B. Chrobrego 17m130 40-883 Katowice**

Posiadam Atari 65XE z magnetofonem. Poszukuję gier BARBARIAN, WINTER GAMES, NIGHT NIGHT.

**Mariusz Omoch, ul. 22 lipca 35 18-420 Jedwabne woj. łomżyńskie**

Co zrobić w 33 komnacie gry ARCANOID, jak pokonać Montezumę w MONTEZUMA's REVENGE? Posiadam Atari 800XL.

**Bartosz Michalczyk, ul. Władysława IV 1-5m32 81-353 Gdynia**

Szukam opisów gier LABIRYNT 202, A.C.E.II, TORPE-DO RUN w wersji na Commodore+4.

**Mariusz Kotecki, Mokiny 13 11-021 Jedzbank woj. olsztyńskie**

Jak uruchomić program ROCKFORD? Pilnie szukam gry THE TRAIN. Posiadam Atari 65XE z magnetofonem.

**Marcin Dylewski, ul. Augustówka 25b 02-981 Warszawa**

Pilnie poszukuję gier ALIENS, ALIEN, THE DETECTIVE, CHIMERA, MIAMI VICE, RETURN of JEDI na Spectrum 48. W zamian wiele innych gier.

**Marek Żuk, ul. 1-go maja 30m4 83-110 Tczew woj. gdańskie**

W zamian za opis i mapę ZORRO udostępnię wiele gier na Atari 65XE z magnetofonem.

**Mariusz Bratko, Jadowniki Mokre 191 33-271, woj. tarnowskie**

Poszukuję opisów do gier GUN SHIP, BARBARIAN, HE-MAN. Pomóżcie! Udostępnię opisy do TRAP DOOR, COMMANDO.

**Leszek Bąkowski, ul. Chyłońska 201m102 81-007 Gdynia**

Mam Atari 800XL z magnetofonem. W zamian za gry ELEVATOR ACTION, MISSION ELEVATOR, INDIANA JOE, KNIGHT LORE, ALIENS oferuję wiele innych gier.

**Michał Wrona, ul. Nowotki 4m7 25-022 Kielce**

Proszę o opisy gier ACE, CRYSTAL CASTLE, DAM BUSTERS, SKY FOX. Posiadam Commodore 64.

**Daniel Żukowski, Wrzesina 63 11-042 Jonkowo**

Bardzo proszę o opisy SPY vs SPY I, II, STRIP POKER. W zamian oferuję opisy do gier SPY vs SPY I i BLUE MAX.

**Adam Makarewicz, ul. Lumumby 31m42 10-207 Olsztyn**

Jestem posiadaczem C-64. Szukam gier ACE OF ACES, GUN SHIP, STEEL THUNDER, SUBMARINE COMMANDER. Za te gry oferuję ACE II, SPITFIRE 40, SOLO FLIGHT, RED ARROWS.

**Andrzej Teszner, os. Wybickiego 26m31 83-300 Kartuzy woj. gdańskie**

Poszukuję kodów do gier KENNEDY APPROACH, ASTEROIDS. Wymienię na opisy innych gier lub zapłacę.

**Jacek Matysiak, ul. Gdynska 20a 82-300 Elbląg**

Uprzejmie proszę o przysłanie mi chipa AY 3-8910. W zamian oferuję około 120 gier na Spectrum (rok 1987—1989).

**Martin Hrušć, ul. Okružná 2060m47 026-01 Ol Dolný Kubín Czechosłowacja.**

Proszę o pomoc w grze FEUD. Ostatnia nadzieja w Czytelnikach „Bajtka”. Mam Atari 800XL.

**Jan Budka, ul. Zalkowicza 36 50-511 Wrocław**

Luke



## AMIGA I GWIAZDY STAR LC-10

**Mam drukarkę. Mała, zgrabna, kremowa obudowa, nawet pasuje kolorystycznie do Amigi. Na biurku przybył jeden kabel, pod biurkiem wolę na wszelki wypadek nie liczyć. Postanowiłem przekonać mój komputer do współpracy z tym — jak zapewniają reklamy — udanym produktem firmy STAR.**

W tym celu przeszukałem dyskietkę systemową. Na dnie jednej z szuflad znalazłem pliki „Init Printer” i „Graphics dump”. Po bożemu „stuknięciem” dwa razy „Init Printer” i wyskoczył ERROR 205 (nie znaleziono zbioru). Widocznie podczas którejś wojny z wirusem usunąłem przypadkowo ten plik. Postanowiłem poszukać gdzie indziej i po kilku minutach udało mi się wydrukować obrazek spod Photon Paint'a. To już był pewniak sukces. Szperając w dyskietkach znalazłem bardzo przydatny program — GrabBit — kopiujący w dowolnym momencie zawartość ekranu na drukarkę po naciśnięciu Ctrl Alt P. Potem postanowiłem wydrukować jakiś tekst. Próba powiodła się, tylko w wydruku nie było ą,ę... itd. — niestety przeróbka Cygnus Edytor'a polegała jedynie na spolszczeniu „fontów”, przez co na ekranie polskie znaki są, ale drukarka ich nie drukuje. Zresztą Cygnus w ogóle nie chciał pracować z drukarką. „Pro Writer” owszem, drukował polskie znaki, i to w kilku krokach i wielkościach czcionki, ale robił to tylko w trybie graficznym, przez co wydruk jednej strony trwał około 15 minut, co przy głupich 4 stronach maszynopisu dawało okrągłą godzinę drukowania. Przewertowałem „Manual” drukarki. Miałem do wyboru cztery możliwości: zmienić ROM w drukarkę, zaprojektować własne znaki dla drukarki jako znaki użytkownika, zaprojektować znaki i drukować je w trybie graficznym lub drukować „ą” jako „a” z przecinkiem itp. Ta ostatnia możliwość była najmniej pracochłonna, ale znaki drukowane w ten sposób są brzydkie, a cofanie głowicy przy każdym polskim znaku zajmuje dużo czasu. Drukowanie znaków w trybie graficznym to znany „chwyt” autorów polskich edytorów tekstu, ale ą i ę są o jedną linię za wysoko, poza tym tak zdefiniowane znaki są odporne na pochylanie, wytłuszczanie, podkreślanie itp. Projektowanie znaków użytkownika jest bardzo czasochłonne, gdyż matryca znaku drukarki ma 11 bajtów, i do tego nie wiadomo dokładnie, który bajt zostanie wydrukowany, a który nie, ale za to znaki tak zaprojektowane są najładniejsze, niemal jak z ROM'u drukarki. Wymiana ROM, która łączyła w sobie zalety poprzednich rozwiązań, bez ich wad, byłaby najłatwiej-

szą, tylko nie bardzo wiedziałem, skąd wziąć nowy ROM. Postanowiłem, że moja drukarka będzie drukować ładne polskie znaki. Zakasałem rękawy, zaprojektowałem znaki dla drukarki, całą sekwencję zamieniłem na znaki ASCII, wysłałem na drukarkę i nic. Uparcie drukuje funty zamiast „ł”, „c” z ogonkiem na dole zamiast „ć” itp. No i bądź tu człowieku mądry i zachowaj zimną krew, skoro godzina projektowania znaków poszła na marne. Po przewertowaniu „ROM Kernel'a” okazało się, że drukarka jest w porządku i to nie ona, a Amiga zignorowała sekwencję znaków sterujących. Otóż do drukarki, z poziomu DOS'u można dobrać się na cztery sposoby:

— Wysłać znaki do „PRT:” czyli poprzez DOS do „driver'a” drukarki.

— Wysłać znaki do „PAR:” czyli bezpośrednio do portu równoległego.

— Wysłać znaki do „SER:” czyli bezpośrednio do portu szeregowego.

— Wysłać znaki do „printer.device” czyli bezpośrednio do „driver'a” drukarki. Wysyłanie znaków do „driver'a” drukarki wiąże się z ich wstępną obróbką przez komputer — gdy w ciągu znaków znajdzie się sekwencja zaczynająca się od ESC, komputer sprawdza w tablicy znaków sterujących, czy jest to któryś z nich, jeśli tak, to zastępuje tę sekwencję odpowiednią dla danej drukarki, pobraną z jej „handler'a”. Gdy sekwencja nie jest żadnym ze znaków sterujących, usuwa z tekstu trzy kolejne znaki, zaczynając od ESC. Wysłanie ciągu znaków do portu szeregowego lub równoległego nie wiąże się z żadną konwersją. Po prostu jest on przekazywany bajt po bajcie do odpowiedniego interfejsu i jeżeli podłączona jest do któregoś z nich drukarka, to już jej sprawa, co wydrukuje. Posiadający tę wiedzę postanowiłem napisać program obsługujący drukarkę. Aby komputer nie czepiał się znaków sterujących wysyłane są one bezpośrednio do interfejsu równoległego („PAR:”). Program najpierw definiuje polskie znaki dla drukarki jako znaki o kodach od 160 do 167 (hex. AO-A7) w kolejności ą,ć,ę,ł,ń,ż,ź,ś. Litera „ó” znajduje się już w ROM'ie drukarki jako znak o kodzie 149 (hex.95). Program napisałem dla drukarki Star LC-10 i prawdopodobnie tylko z nią działa poprawnie. Dla innych drukarek, w których można zdefiniować własne znaki, należy podać odpowiednie sekwencje znaków sterujących (trzeba poszukać w instrukcji). Dla drukarek, w których nie można zaprojektować własnych znaków, trzeba niestety program przerobić tak, aby każdy polski znak był drukowany w trybie graficznym, tzn. każdy znak należy zamienić na sekwencję rozkazu druku 8 bajtów grafiki i ciągu bajtów będących obrazem danego znaku. W programie zdefiniowane są tylko małe litery. Przyjemność zaprojektowania dużych liter pozostawiam potencjalnemu czytelnikowi.

Marcin Bójko

/\* Program drukujący standardową stronę maszynopisu (30 linii po 60 znaków) na drukarce Star LC-10

wersja 1.1 26.12.89

kompilacja: cc -s +i nazwa.c konsolidacja: ln nazwa.o -ic32 /\*

```
#include <exec/types.h>
#include <libraries/dos.h>
#define EOF -1
#define EOL 10
#define IL_LIN 30
#define L_MARG 10
```

```
#define P_MARG 75
#define ODST (int)(11.7*216/(IL_LIN+10))
struct File *plik,*druk;
struct DosBase *DosBase;
int d,i,k,ctr;
char bufor[2048],nazwa[100],ctr[100],c;

BYTE program_drukarki[129]={
0x1b,0x40, /* Wyzeruj drukarkę (reset) */
0x1b,0x3a,0,0,0, /* Skopiuj znaki z ROM do RAM drukarki */
0x1b,0x49,0x34, /* Znaki definiowane będą w trybie draft */
0x1b,0x25,0x31, /* Używaj znaków z RAM a nie z ROM */
0x1b,0x26,0,0xa0,0xa7, /* Definicja znaków od $A0 do $A7 */
0x08,0x08,0x1c,0x54,0x54,0x54,0x54,0x56,0x7b,0x3d,0x04,0x00,
0x88,0x00,0x1c,0x3e,0x22,0x22,0x62,0xe2,0xa2,0x22,0x22,0x00,
0x87,0x00,0x00,0x8a,0x92,0xfe,0xfe,0x12,0x22,0x00,0x00,0x00,
0x88,0x00,0x3e,0x3e,0x20,0x60,0xe0,0xa0,0x20,0x3e,0x1e,0x00,
0x09,0x38,0x7c,0x54,0x54,0x54,0x54,0x56,0x57,0x75,0x35,0x00,
0x87,0x22,0x26,0x26,0x6a,0xaa,0x32,0x32,0x22,0x00,0x00,0x00,
0x87,0x2a,0x2e,0x2e,0x2a,0x3a,0x3a,0x2a,0x2a,0x00,0x00,0x00,
0x88,0x10,0x2a,0x2a,0x2a,0x6a,0xea,0xaa,0x2a,0x24,0x00,0x00,
0x12, /* Wielkość znaku - pica */
0x1b,0x58,L_MARG,P_MARG, /* Ustaw lewy i prawy margines */
0x1b,0x33,ODST, /* Ustaw odstęp między liniami */
0x1b,0x66,0x31,0x02, /* Wsuń 2 linie (wykorzystane w pętli drukującej) */
0x1b,0x61,0x01 /* Centruj linię (j.w.) */
};
BYTE tablica_CED[9]={0xe2,0xea,0xee,0xef,0xeb,0xfa,0xfb,0xf4,0xf3};
BYTE tablica_prog[9]={0xa0,0xa1,0xa2,0xa3,0xa4,0xa5,0xa6,0xa7,0x95};
/* Procedura szukająca polskich znaków w tekście */

char
szukaj(znak)
char znak;
{
int i;
for(i=0;i<9;i++)
if(tablica_CED[i]==znak)
return(tablica_prog[i]);
return(znak);
}

main()
{
DosBase=OpenLibrary("dos.library",0); /* otwórz bibliotekę Dos'u */
printf("Podaj nazwę pliku do wydrukowania ");
scanf("%s",nazwa);
/* Potwierzaj potrzebne ci pliki */
if((plik=Open(nazwa,MODE_OLDFILE))==0)
{
printf("\n Nie znalazłem takiego pliku");
CloseLibrary(DosBase);
exit(0);
}
if((druk=Open("PAR:",MODE_NEWFILE))==0)
{
printf("\n Nie znalazłem zbioru 'paralel.device', lub jest on uszkodzony");
Close(plik);
CloseLibrary(DosBase);
exit(0);
}
printf("Włącz drukarkę\n");
Write(druk,&program_drukarki[0],122); /* zaprogramuj drukarkę */
printf("Włóż papier do drukarki, ustaw co chcesz i wciśnij RETURN");
scanf("%c",c); /* czekaj na dowolny znak */
for(i=0;i<7;i++)ctr[i]=program_drukarki[122+i];

/* pętla drukująca */
while((d=Read(plik,&bufor[0],2048))>0)
{
for(i=0;i<d;i++)
{
c=szukaj(bufor[i]);/* to czasem polski znak */
if(c==EOL) /* gdy pobrany znak jest znakiem końca linii */
k++; /* zwiększ o jeden licznik linii */

if(k==IL_LIN) /* gdy już wydrukowałeś jedną stronę */
{
str++;k=0; /* zwiększ liczbę stron o 1 i wyzeruj liczbę linii */
ctr[6]=1; /* centruj otrzymaną linię */
sprintf(&ctr[7],"Str. %d\n",str);
ctr[99]=12; /* wysuń stronę */
Write(druk,&ctr[0],100); /* wydrukuj numer strony */
ctr[6]=0; /* dosuwaj linie do lewego marginesu */
Write(druk,&ctr[4],3); /* wyślij ostatni rozkaz drukarce */
}
else
Write(druk,&c,1); /* wydrukuj otrzymany znak */
}
}
str++;
ctr[6]=1;
ctr[3]=IL_LIN+2-k;
sprintf(&ctr[7],"Str. %d\n",str); /* wydrukuj numer ostatniej strony */
ctr[99]=12;
Write(druk,&ctr[0],100);
/* przed wyjściem z programu zamknij co trzeba ważne! */
CloseLibrary(DosBase);
Close(druk);
Close(plik);
}
}
```



# ŚWIAT MAKROASSEMBLERÓW — CZ. I

Ten cykl artykułów wiąże się nierozłącznie z „Monitorami języka maszynowego” i w zasadzie jest ich logiczną kontynuacją. Monitor języka maszynowego jest narzędziem pomocnym, lecz układanie dużego programu z jego pomocą jest bardzo uciążliwe. Ten cykl artykułów ma za zadanie wskazać Wam i przedstawić możliwości programów zwanych makroassemblerami — profesjonalnych narzędzi do programowania w języku maszynowym.

Dobry makroassembler jest programem rzadkim. Choć liczba programów dla C-64 nadal rośnie, to tak naprawdę jest ich mniej więcej około dziesięciu. Do najlepszych zaliczają się następujące:

- Commodore 64:**  
**PAL 64, CBM MADS, MERLIN 64, DEVELOP 64, PANTHER C64, MAE64,**  
**Commodore 128:**  
**LADS 128, ELMA**

Oprócz podanych powyżej istnieje jeszcze wiele innych o nie zbadanych przeze mnie możliwościach np.: HIFI C64 i TOP ASS. Częściowo jest to spowodowane brakiem jakichkolwiek instrukcji, częściowo przez brak programów. A może Czytelnicy włączyliby się również w ten cykl artykułów nadsyłając materiały o innych mniej znanych makroassemblerach?

## DLACZEGO MAKROASSEMBLER?

Odpowiedź na to pytanie jest prosta: gdy makroassembler pozwala na stosowanie etykiet. Etykieta jest w zasadzie odpowiednikiem zmiennej numerycznej BASIC-u; różnica polega na tym, że w BASIC-u przypisuje się zmiennym konkretne wartości i liczby, natomiast etykietom — adresy.

Program A	Program B
C000 LDX #S00	C000 CHROUT=\$FFD2
C002 TXA	C002 LDX#S00
C003 LDA	C002 TXA
C006 JSR \$FFD2	C003 LOOP LDA \$8000,X
C009 INX	C006 JSR CHROUT
C00A CPX #S80	C009 INX
C00C BNE \$C003	C00A CPX #S80
C00E RTS	C00C BNE LOOP
	C00E RTS

W powyższym przykładzie przedstawiono dwa programy: jeden wpisujący za pośrednictwem monitora języka maszynowego (A) oraz układany za pomocą makroassemblera (B). W programie B zastosowano dwie etykiety: **CHROUT** i **LOOP**. Pierwsza jest definiowana na początku programu, natomiast drugą zastosowano po prostu dla ułatwienia sobie życia. Zwróć uwagę na linię C00C; w programie A musisz obliczyć adres skoku, natomiast w programie B adres ten (\$C003) zostanie wyliczony automatycznie przez makroassembler. Jeżeli dodam, że nie ma żadnych ograniczeń i możesz stosować dowolną ilość etykiet w swoich programach, to łatwo zrozumieć, dlaczego i w jaki sposób ich stosowanie ułatwia Twoje zadanie. Ponadto program, w którym zamiast adresów występują etykiety, jest znacznie bardziej przejrzysty; jeśli zamiast \$FFD2 użyjesz np. etykiety WYDRUK, to znacznie szybciej potąpiesz się, do czego program służy i co w danym miejscu robi.

Zaletą makroassemblerów jest nie tylko możliwość stosowania etykiet, lecz także automatyczne wpisywanie dowolnych tekstów w dowolne miejsce w pamięci. W miarę omawiania poszczególnych programów będziemy sobie wszystkie początkowo zawile kwestie wyjaśniać.

## PAL 64

Naszą wędrówkę po świecie makroassemblerów rozpoczniemy od programu o nazwie **PAL** — **PERSONAL ASSEMBLY LANGUAGE**. Był to jeden z pierwszych makroassemblerów, jakie pojawiły się na rynku, i stał się on jednocześnie wzorem dla następnych programów tego typu (np. **LADS** dla C-128).

**PAL** ma jedną wprost kapitalną przewagę nad wieloma programami tego typu — pozwala na współpracę z magnetofonem, co sprawia, iż jest on dostępny dla setek posiadaczy Commodore w zestawach uboższych o stację dysków. Choć niektóre możliwości programu wymagają właściwie stacji dysków, to jednak bez żadnych problemów można odczytywać i zapisywać programy źródłowe na taśmie (oczywiście z etykietami).

**PAL 64** zajmuje 4.1 KB pamięci RAM od adresu \$9000 co znaczy, że nie koliduje on z interpreterem BASIC'a. Na dyskietce oryginalnej istnieje też odpowiednia wersja programu (**PAL.R**), którą można wczytać w dowolne miejsce pamięci w zależności od potrzeb użytkownika. W wersji normalnej program uruchamia się za pomocą SYS 700 poprzez specjalnie zmieniony wektor. W tym też momencie następuje zamiana kodu źródłowego na kod wynikowy (tzn. Twój program zawierający np. etykiety jest zamieniany na konkretny program maszynowy). Wskazane jest w związku z tym, aby zawsze w pierwszej linii Twoich programów zawarta była jedynie instrukcja SYS 700.

Dużym ułatwieniem dla początkujących jest fakt, że program, wpisuje się identycznie tak, jak normalny program BASIC. Poszczególne linie są numerowane (zgodnie z zasadami obowiązującymi w BASIC-u). W pewnych sytuacjach w linii może występować kilka poleceń oddzielonych od siebie za pomocą dwukropka.

Zasada wprowadzania i kasowania linii jest również identyczna jak w BASIC-u.

Po wczytaniu **PAL 64** do pamięci wpisz RUN i wciśnij RETURN. Na ekranie pojawi się komunikat (m.in. „[0000—0000]”). Następnie wpisz NEW i wciśnij RETURN. Program jest gotowy do wprowadzenia wersji źródłowej programu.

```
1000 SYS 700
1010 ;
1011 .OPT 00
1012 ;1013 ; PROGRAM DODAWANIE1
1014 ;
1020 SKŁAD1 = $2000
1030 SKŁAD2 = 8193
1040 WYNIK = 8194
1050 ;
1060 * = $C000
1070 ;
1100 CLC ; ZNACZNIK CARRY = 0
1110 LDA SKŁAD1 ; DO PIERWSZEGO SKŁADNIKA
1120 ADC SKŁAD2 ; DODAJ DRUGI SKŁADNIK
1130 STA WYNIK ; ZAPISZ WYNIK
1140 RTS ; KONIEC PROGRAMU
```

Powyższy program dodaje do siebie zawartość komórek 8192 i 8193 i zapisuje wynik w komórce 8194. Zwróć uwagę, że **PAL 64** daje Ci możliwość postępowania się zarówno liczbami w systemie dziesiętnym, jak i w szesnastkowym. Jest to następny plus tego programu ułatwiający znacznie rozpoczęcie układania własnych programów początkującym.

Najważniejsza definicja w tym programie znajduje się w linii 1060. Ustala ona, iż program będzie zawarty w obszarze od adresu \$C000 (49152). W chwili uruchomienia programu **PAL 64** po zmianie kodu źródłowego na wynikowy umieści ten ostani w obszarze od adresu \$6000 (49152) poczynając. Oczywiście możesz sam określić obszar pamięci najbardziej Ci odpowiadający; w naszych przykładach dla C-64 będą jednak ze względów bezpieczeństwa zawsze korzystał z tego właśnie obszaru pamięci RAM. Linia zawierająca definicję obszaru pamięci powinna być **ZAWSZE** umieszczana na początku programu.

Linia 1011 zawiera instrukcję sterującą pracą makroassemblera. Przyjęło się, że w programach tych większość takich instrukcji występuje zawsze po kropce. Instrukcja **OPT** określa parametry wyjściowe **PAL 64**. Dwa zera po tej instrukcji nakazują zapisanie kodu wynikowego w obszarze pamięci zdefiniowanym w linii 1060 (czyli od adresu \$C000). Średnik to odpowiednik instrukcji **REM** w BASIC-u — pozwalają Ci na wpisanie komentarza czy uwag (choć nie zawsze).

Dalej (linie 1020—1040) deklarujemy etykiety. Etykieta nie może być ani instrukcją makroassemblera (np. **OPT**) ani mnemonik (np. **LDA, JSR** czy **STX**). W **PAL 64** etykieta może być dowolnie długa, lecz tylko pierwszych 8 znaków jest faktycznie branych pod uwagę. Po etykietach **MUSI** następować spacja. Ze względu na inne programy, z którymi niebawem się zetkniesz, warto przyjąć zasadę, że etykieta będzie się składała maksymalnie z 6 znaków.

Po wpisaniu tego programu zapisz go na dyskietce lub taśmie za pomocą **SAYE** „jakaśnazwa”, 8 (jeśli masz stację dysków) lub 1, jeśli dysponujesz tylko magnetofonem. Następnie wpisz **POKE 49152,0** i wciśnij **RETURN**. Teraz uruchom program za pomocą **RUN**. Gdy na ekranie pojawi się komunikat (np. [C000—C00B]) to znaczy, że program jest gotowy do uruchomienia. W wypadku

wystąpienia błędu na ekranie ujrzysz stosowny komunikat.

Przed uruchomieniem programu wpisz: **POKE8192,2:POKE8193,2:POKE8194,0:PRINT PEEK(49152)** i wciśnij **RETURN**. W ten sposób do komórek pamięci wymienionych w programie wpiszesz liczby, których wynik dodawania możesz łatwo sprawdzić; ponadto na ekranie zobaczysz, że w obszarze od adresu 49152 coś się zmieniło (został wpisany program wynikowy).

Wykonaj teraz **SYS 49152**, a następnie **PRINT PEEK(8194)**. W rezultacie na ekranie powinieneś otrzymać liczbę 4 stanowiącą wynik działania programu.

**PAL 64** pozwala również na sporą dowolność w zapisie danych. Na przykład zapis **LDX #S80** jest dozwolony w postaciach: **LDX #S80, LDX #128 i LDX #%1000000**. Gdy chcesz wczytać do dowolnego rejestru kod ASCII jakiegos znaku wystarczy wpisać np. **LDY #„Q”**. Możliwe jest również podstawianie wyrażeń arytmetycznych, np. **LDA #\*(\$8000—77)”. Do tego tematu będziemy jeszcze niejednokrotnie wracać w dalszych częściach cyklu.**

**PAL 64** daje użytkownikowi do dyspozycji następujące instrukcje:

- .BYTE** — pozwala na wpisanie jednego lub więcej bajtów
- .WORD** — jak wyżej lecz wpisuje słowo dwubajtowe
- .FILE** — pozwala na łączenie programów źródłowych
- .IF** — instrukcja warunkowa
- .GOTO** — wykonuje skok do określonej linii programu
- .GTB** — powoduje powrót do BASIC
- .ASC** — pozwala na wprowadzenie ciągu tekstu do programu
- .SYS** — odpowiednik SYS w BASIC
- .STM** — określa najniższy adres tablicy symboli
- .SST** — pozwala na wczytanie tablicy symboli
- .LST** — uzupełnienie instrukcji .SST
- .END** — koniec programu lub łączenia programów
- .OPT** — ustala parametry wyjściowe PAL 64
- .BAS** — pozwala na łączenie programów BASIC i maszynowych

Instrukcje te omówimy dokładniej w następnej części cyklu. O jakości **PAL 64** niech świadczy fakt, iż znane wydawnictwo **COMPUTE!** wybrało ten właśnie makroassembler na program, w formacie którego zapisano sto kilkadziesiąt programów źródłowych w książce „MACHINE LANGUAGE ROUTINES FOR THE COMMODORE 64/128”. Książka ta została wydana w roku 1987 (niebawem **KLAN COMMODORE** przedstawi jej recenzję), natomiast **PAL 64** powstał w roku 1979. Jeśli po 8 latach poważne wydawnictwo decyduje się na taki krok, to program musi być naprawdę dobrej klasy.

Jak widać choćby z tego artykułu makroasembler są dość złożonymi programami, których omówienie możnaby ciągnąć prawie w nieskończoność. Proszę przeto o wyrozumiałość Czytelników, ponieważ może się zdarzyć że z braku miejsca będę musiał niektóre informacje pominąć koncentrując się na tych najbardziej istotnych. W miarę możliwości będę starał się także przedstawiać programiki demonstracyjne. (CDN)

*Klaudiusz Dybowski*

## ZAGRAJMY W TO JESZCZE RAZ

### WITAJCIE SZALENI

(i TY, mały Jasiu, też!)

Zgodnie z obietnicą z poprzedniego miesiąca prezentujemy dziś kolejną porcję świeżej krwi na młyn waszej niepohamowanej żądzy zabijania, patroszenia, bombardowania i wysadzania w powietrze wszystkiego, co tylko się da. Każdy z was, obudzony w środku nocy powinien wyrecytować jak w transie „formułę nieśmiertelności”: **RESET; POKE 1234,56 SYS 12345** lub **POKE 1234,56; RUN. NIECH ZŁE MOCE MAJĄ WAS W SWOJEJ OPIECE** i do kolejnego spotkania w pozycji przełącznika **ON**.

*Waldemar Nowak*

CAPTAIN STARLIGHT	13368,173	L	CRACKER'S REVENGE	30190,255	L
CAULDRON 2	40315,221	L	CRAZY CAVE MAN	3332,255	S
	40316,248	L	CRAZY COMETS	37002,169	L
				*37002,0	L
CAVE FIGHTER	35276,255	L		*37004,234	L
CAVELON	23789,255	L		SYS 24881	
	*33789,99	L		*36838,120	L
	*15458,255	L		SYS 26386	
	15485,255	L		*16187,120	L
CAVEMAN	*25485,255	L	CRAZY KONG	30624,173	L
CAVERNS OF ERIBIAN	3332,255	L	CRISIS MOUNTAIN	2665,238	L
	48291,221	L		*3144,238	L
	48292,248	L	CROSSFIRE	27625,173	L
				*5353,44	L
CAVERNS OF KHAFKA	SYS 50333	L	CRYSTAL CASTLES	5643,255	S
	37803,234	L	CRYSTAL CASTLES 2	5714,173	L
	37804,234	L	CURSE OF SHERWOOD	64613,0	L
CAVERNS OF SILAHC	16805,169	L		58257,201	L
	16806,0	L	CUTHBERT IN JUNGLE	2659,5	L
	16807,234	L	CUTHBERT IN SPACE	2726,255	S
CHINA MINER	SYS 5000	W	CYLU	39409,173	L
	34623,234	L			
	34624,234	L	D		
	34625,234	L	DALLAS STAR	3271,0	L
	*34562,10	L	DAN DARE 2	21605,165	L
	*34532,44	L		12634,165	L
	*33301,245	L		SYS 2560	L
	33457,255	L	DARE DEVIL DENNIS	29173,255	L
CHILLER	22501,189	L		17598,x	T
CHOP LIFTER	8011,173	L		17958,0-255	TEMPO
CHUCKIE EGG	16851,255	L	DARE DEVIL DENNIS 2	24683,138	L
	SYS 14848	L		24684,2	L
				SYS 39700	L
CLIFF HANGER	5528,0	L	DECATHLON	9450,173	L
CLOWNS	3566,255	L	DEFENDER	3005,5	L
	*23789,255	L		*2814,255	L
CODE HUNTER	8759,173	L	DEFLECTOR	22709,0	L
	SYS 4700	L		SYS 2088	RESET I
COHEN'S TOWER	7279,0	L		*1023,1	DALEJ
COMBAT	32765,42	L		*1023,4	
COMMANDO	2180,255	L		*1023,5	
	*13590,0	E		*13967,165	
	*2409,44	E		*14073,165	
	9906,x	LI		9715,36	
CONGO BONGO	3655,5	L		11890,x	LN
COSMIC CAUSEWAY	14951,240	L		SYS 9658	
	SYS 8608	L			



# KLAN AMSTRAD

# OPERACJE DYSKOWE W SYSTEMIE CP/M PLUS

## cz. II — Procedury dyskowe niskiego poziomu (Amstrad)

Kontynuując rozważania dotyczące operacji dyskowych w systemie CP/M Plus opiszemy w tym numerze procedury niskiego poziomu pozwalające na obsługę fizycznych napędów komputerów firmy Amstrad.

### Funkcja USERF modułu BIOS

W systemie CP/M Plus przy implementacji modułu BIOS istnieje możliwość wprowadzenia przez producenta sprzętu własnych funkcji, które nie zostały uwzględnione w standardowym szkielecie dostarczanym przez firmę Digital Research. W komputerach Amstrad PCW i CPC funkcja USERF umożliwia dostęp do 36 procedur, z których 18 (TAB. 1) obsługuje fizyczne napędy dyskowe.

Wywołanie procedury dostępnej przez USERF może być zrealizowane w assemblerze przy pomocy następującej sekwencji rozkazów:

```
LD HL, (0001) ; adres modułu BIOS
LD BC, 87 ; 30 funkcja modułu 3*(30-1)
ADD HL, BC
LD (ADR), HL
..... ; instrukcje definiujące parametry
..... ; wejściowe
CALL JPHL ; wywołanie jednej z funkcji USERF
DW ADRES ; adres funkcji USERF wg Tab. 7
RET ; powrót do programu głównego
JPHL: DB 0C3h ; instrukcja skoku bezwarunkowego
ADR: DW 0 ; adres skoku
```

Ponieważ w komputerach Amstrad adres punktu wejściowego dla procedur dostępnych przez USERF jest równy FC5Ah, to powyższy kod można uprościć:

```
..... ; instrukcje definiujące parametry
..... ; wejściowe
CALL FC5Ah
DW ADRES
RET
```

Analogiczne odwołanie przy pomocy instrukcji INLINE w Turbo Pascalu ma postać:

```
INLINE ({.....} $CD/$5A/$FC/$xx/$00);
```

gdzie xx młodszy bajt adresu procedury dostępnej przez USERF.

### Obsługa napędów dyskowych

Chociaż wiele operacji dyskowych jesteśmy w stanie wykonać przy pomocy standardowych funkcji modułu BIOS, to niektóre z nich możliwe są wyłącznie dzięki procedurom dostępnym przez funkcję USERF. Dotyczy to szczególnie formatowania dysku i bezpośredniego programowania sterownika napędów (układ  $\mu$ PD765A).

Przy omawianiu procedur dyskowych pakietu USERF należy zwrócić uwagę na różnicę między tablicami DPB i XDPB (ang. eXtended Disk Parameter Block). Celem uproszczenia opisu tablic dyskowych systemu CP/M Plus posługiwaliśmy się dotychczas wyłącznie pojęciem DPB. W rzeczywistości tablica DPB definiowana jest jako 17 pierwszych bajtów, pełnej tablicy XDPB, zawierającej 27 bajtów.

Niektóre parametry omawianych procedur muszą rezydować we wspólnym, dla wszystkich banków, obszarze pamięci między adresami C000h a FFFFh. Problem ten dotyczy szczególnie tablicy XDPB, której adres przekazywany jest przez rejestr IX.

Błędne wykonanie procedur dyskowych powoduje najczęściej wyzerowanie wskaźnika CARRY i zwrócenie kodu błędu w rejestrze A. Komunikaty odpowiadające tym błędom zebrano w Tab. 2, podając numer błędu, opis angielski i polskie tłumaczenie.

Po tych wstępnych uwagach opiszemy niektóre procedury dyskowe zaimplementowane przez producenta przy pomocy funkcji USERF.

### 3. DD READ SECTOR (0086H)

Czyta sektor dyskowy do pamięci komputera. Parametry wejściowe procedury:

B = bank pamięci (0-systemowy, 1 — TPA)  
C = numer fizycznej stacji (0-pierwszy napęd, 1-drugi napęd)  
D = numer ścieżki logicznej  
E = numer sektora logicznego  
HL = adres bufora w zadanym przez rej. B banku pamięci  
IX = adres tablicy XDPB umieszczonej we wspólnym obszarze pamięci (C000h-FFFFh).

Jeśli operacja została zakończona sukcesem ustawiany jest wskaźnik CARRY. W przeciwnym wypadku wskaźnik ten jest zerowany i rejestr A zawiera kod błędu (0,2,3,4,5,8). Zawartość rejestrów BC, DE, HL jest niszczone.

### 4. DD WRITE SECTOR (0089H)

Zapisuje zawartość bufora na sektor dyskowy. Parametry wejściowe są takie same jak dla operacji DD READ SECTOR. Podobnie, zakończenie procedury powoduje ustawienie wskaźnika CARRY, w przypadku sukcesu i jego wyzerowanie w przeciwnym wypadku. Kody błędów przekazywane w rejestrze A mogą przyjmować wartości: 0,1,2,3,4,5,8. Zawartość rejestrów BC, DE, HL jest niszczone.

### 5. DD CHECK SECTOR (008Ch)

Sprawdza zawartość sektora dyskowego z zawartością bufora dyskowego. Parametry wejściowe są takie same, jak dla procedur DD READ SECTOR i DD WRITE SECTOR. Zakończenie operacji sukcesem ustawia wskaźnik CARRY. Kody błędów przekazywane przy wyzerowanym wskaźniku CARRY, w rejestrze A, są następujące: 0,2,3,4,5,8. Jeśli zawartość bufora w pamięci zgadza się z zawartością sektora na dysku, to ustawiany jest wskaźnik ZERO (zero flag). Zawartość rejestrów BC, DE, HL jest niszczone.

### 6. DD FORMAT (008Fh)

Formatuje jedną ścieżkę dysku posługując się informacją zawartą w buforze nagłówka informacji (ang. header information buffer). Parametry wejściowe są podobne do stosowanych w poprzednich procedurach i wyglądają następująco:

B = numer banku pamięci zawierający „header information buffer”  
C = numer napędu  
D = numer ścieżki logicznej  
E = bajt wypełniający, zwykle E5h (dla IBM PC — F6h)  
HL = adres „header information buffer”  
IX = adres tablicy XDPB (we wspólnym obszarze pamięci)

Format bufora nagłówka informacji jest następujący:  
informacja o pierwszym sektorze  
informacja o drugim sektorze

.....  
informacja o ostatnim sektorze

Format informacji o sektorze:

bajt 0: numer ścieżki fizycznej  
bajt 1: numer głowicy  
bajt 2: numer sektora fizycznego  
bajt 3:  $\log_2$  (rozmiar sektora fizycznego) - 7, zwykle 2 dla sektora o rozmiarze 512 bajtów.

Dla operacji zakończonej sukcesem ustawiany jest wskaźnik CARRY. W przeciwnym przypadku, kod błędu przekazywany jest w rejestrze A: (0,1,2,3,4,5,8). Zawartość rejestrów BC, DE, HL jest niszczone.

### 7. DD LOGIN (0092h)

Wybiera zadany napęd jako aktualny (ang. disc login). Usiłuje ustalić format dysku i w przypadku sukcesu inicjalizuje tablicę XDPB. Nie uwzględnia i nie zmienia zawartości ostatniego bajtu XDPB, który decyduje o tym, czy ustalenie formatu dysku jest dozwolone. Parametry wejściowe:

C = numer napędu

IX = adres XDPB we wspólnym obszarze pamięci

Jeśli operacja kończy się sukcesem to wskaźnik CARRY jest ustawiany i tablica XDPB inicjalizowana, zgodnie z formatem dysku. Rejestr A zwraca typ dysku:

0 — systemowy dysk PCW 8256/8512  
1 — systemowy dysk CPC 6128  
2 — format typu data dla CPC 6128  
3 — format dysku stacji dwustronnej B: w PCW 8256/8512.

W rejestrach DE i HL zwracane są:

DE = rozmiar dwubitowego wektora alokacji

HL = rozmiar tablicy (hash table)

W przypadku błędu wskaźnik CARRY jest zerowany i niszczone jest zawartość rejestrów DE i HL oraz tablicy XDPB. Kody błędów zwracane w rejestrze A: 0,2,3,4,5,6.

Tab. 1. Lista procedur USERF dla komputerów firmy Amstrad

adres	nazwa	wykonywana funkcja
Obsługa napędów dyskowych		
#0080	DD INIT	inicjalizacja napędu
#0083	DD SETUP	zadanie parametrów napędu
#0086	DD READ SECTOR	odczyt sektora
#0089	DD WRITE SECTOR	zapis sektora
#008C	DD CHECK SECTOR	sprawdzenie sektora
#008F	DD FORMAT	formatowanie ścieżki
#0092	DD LOGIN	wybór aktualnego napędu (ang. login)
#0095	DD SEL FORMAT	wybór jednego z formatów
#0098	DD DRIVE STATUS	pobranie statusu napędu
#009B	DD READ ID	odczyt identyfikatora sektora
#009E	DD L DPB	inicjalizacja DPB
#00A1	DD L XDPB	inicjalizacja XDPB
#00A4	DD L ON MOTOR	uruchomienie silnika napędu
#00A7	DD L T OFF MOTOR	odmierzenie czasu wyłączenia silnika
#00AA	DD L OFF MOTOR	wyłączenie silnika napędu
#00AD	DD L READ	obsługa poleceń sterownika napędu
#00B0	DD L WRITE	obsługa poleceń sterownika napędu
#00B3	DD L SEEK	wyszukanie zadanej ścieżki

Tab. 2. Komunikaty błędów dla procedur dyskowych

nr	komunikat angielski	polskie tłumaczenie
0	drive not ready	(napęd nie gotowy)
1	write protected	(dysk zabezpieczony przed zapisem)
2	seek fail	(niemożliwe wyszukanie zadanej ścieżki)
3	data error	(błąd danych)
4	no data	(brak danych)
5	missing address mark	(brakujący znacznik sektora)
6	bad format	(niewłaściwy format)
7	nie używany	
8	media changed	(zmieniono dyskietkę przed zakończ. oper.)



## 9. DD DRIVE STATUS (0098h)

Pobiera słowo statusu danego napędu. Parametrem wejściowym jest rejestr C, którego pierwszy trzy bity są istotne i mają następujące znaczenie:

bit 0,1 — numer napędu (0 lub 1)

bit 2 — głowica.

Status zwracany jest w rejestrze A i ma postać:

bit 7 — niezdefiniowany

bit 6 — zabezpieczony przed zapisem (write protected)

bit 5 — gotowy

bit 4 — ścieżka zerowa

bit 3 — niezdefiniowany

bit 2 — głowica

bit 1,0 — numer napędu

Dla drugiego napędu (numer 1) zerowa wartość bitów 5 i 6 oznacza brak tego napędu.

## 10. DD READ ID (009Bh)

Czyta identyfikator pierwszego znalezionej sektora.

Parametry wejściowe:

C = numer napędu

D = numer logicznej ścieżki

IX = adres tablicy XDPB w obszarze wspólnym pamięci.

Jeśli operacja zakończona powodzeniem, to wskaźnik CARRY jest ustawiony, a rejestr A zawiera numer znalezionej sektora. Rejestr HL wskazuje adres bufora umieszczonego we wspólnym obszarze pamięci.

Format bufora:

bajt 0 — liczba zwróconych bajtów

bajt 1 i następne — zwrócone bajty.

Jeśli operacja DD READ ID kończy się błędem, to jego kod zwracany w rejestrze A ma wartość: 0,2,3,4,5.

Niszczona jest zawartość rejestrów BC, DE.

## Plik DISK3.SYS

Na listingu 1. przedstawiono plik DISK3.SYS zawierający 9 procedur pascaliowych, z których 7 realizuje omówione funkcje USERF obsługi napędów dyskowych. Zbiór ten zostanie wykorzystany później do napisania programu formatującego.

JONASZ MAYER

```
1: (*****)
2: {
3: { Plik DISK3.SYS (C) JM 1989 }
4: {
5: { Biblioteka procedur dyskowych niskiego poziomu korzystająca }
6: { z funkcji systemowych USERF. Przeznaczona na komputery }
7: { Amstrad PCW 8256/8512 i CPC 6128. Umożliwia obsługę obu }
8: { napędów fizycznych, }
9: {
10: { Lista procedur i funkcji: }
11: {
12: { 1. procedure GetDPB (d : char; var DPB : DPBrec); }
13: { 2. procedure SetDPB (d : char; DPB : DPBrec); }
14: { 3. procedure DD_Read (drive : char; track, sector : byte); }
15: { 4. procedure DD_Write (drive : char; track, sector : byte); }
16: { 5. procedure DD_Check (drive : char; track, sector : byte); }
17: { 6. procedure DD_Format (drive : char; track, filler : byte); }
18: { 7. procedure DD_Login (drive : char); }
19: { 8. function DD_DriveStatus (drive : char) : byte; }
20: { 9. procedure DD_ReadID (drive : char; track : byte); }
21: {
22: (*****)
23: const
24: DPBaddr : array [0..1] of integer = ($FF4A,$FF65);
25: type
26: DPBrec = record
27: ( DPB ) LSPT : integer; { sektory logiczne na ścieżce }
28: BSH, BLM : byte; { rozmiar bloku alokacji }
29: EXM : byte; { rozmiar segmentu }
30: DSM : integer; { pojemność dyskiety w blokach }
31: DRM : integer; { liczba pozycji katalogu }
32: ALO : integer; { alokacja katalogu }
33: CKS : integer; { rozmiar sumy kontrolnej }
34: DFF : integer; { zarezerwowane ścieżki }
35: PSH, PHM : byte; { rozmiar sektora fizycznego }
36: ( X-DPB ) SideNess : byte; { "Stronność" }
37: TPS : byte; { ścieżki na stronę }
38: PSPT : byte; { fizyczne sektory na ścieżce }
39: FSN : byte; { numer pierwszego sektora }
40: SS : integer; { rozmiar sektora fizycznego }
41: RWGAP : byte; { przerwa dla zapisu i odczytu }
42: FGAP : byte; { przerwa dla formatowania }
43: MFM : byte; { rodzaj zapisu }
44: ( X-DPB ) AUTO : byte; { autodetekcja formatu }
45: end;
46: SDSrec { Short Disk Specification - krotka specyfikacja dysku }
47: = record
48: DT : byte; { typ dysku 00,01,02,03 }
49: SDN : byte; { "stronność" }
50: sTPS : byte; { ścieżki na stronę }
51: sPSPT : byte; { fizyczne sektory na ścieżce }
52: sSS : byte; { rozmiar sektora fizycznego }
53: sDFF : byte; { zarezerwowane ścieżki }
54: sBSH : byte; { rozmiar bloku alokacji }
55: NDE : byte; { liczba bloków na katalog }
56: sRWGAP : byte; { przerwa dla zapisu i odczytu }
57: sFGAP : byte; { przerwa dla formatowania }
58: end;
59: buffer = array [1..512] of byte;
```

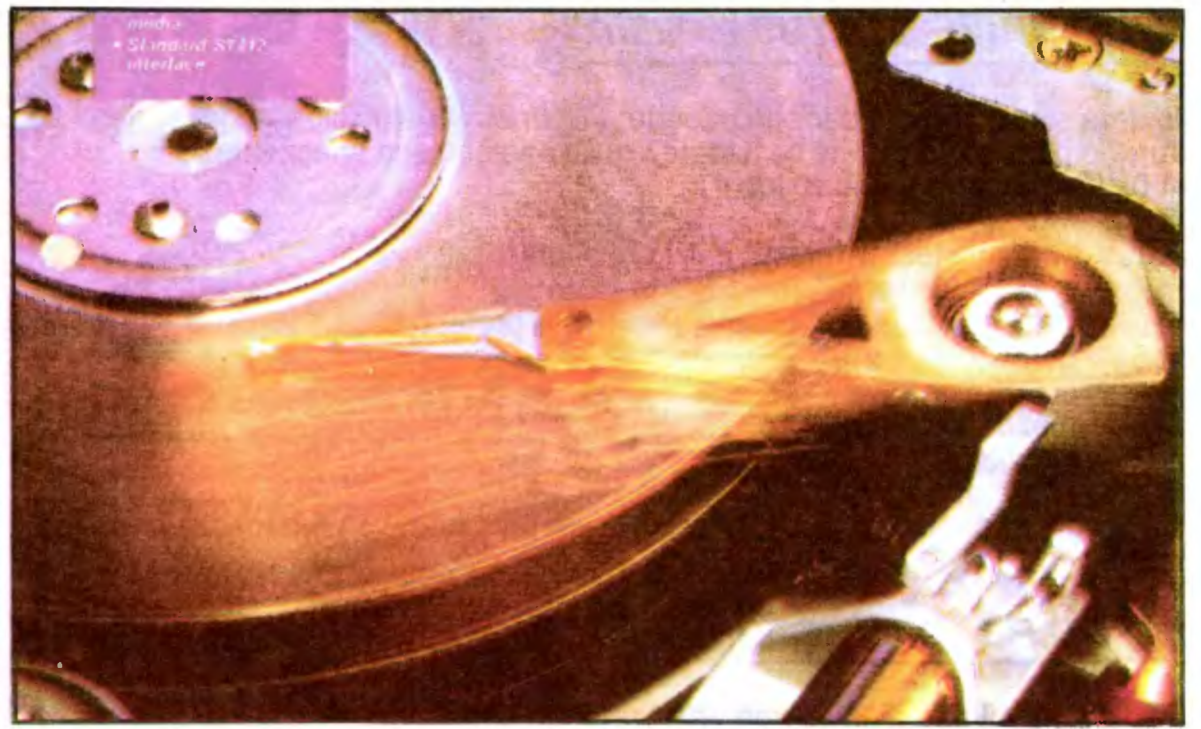
```
60:
61: var
62: error, status, id : byte;
63: ok, matched : boolean;
64: buf : buffer;
65: header : array [1..16,0..3] of byte;
66:
67: procedure GetDPB (d : char; var DPB : DPBrec);
68: (*****)
69: { Pobiera tablice DPB napędu "d" (A,B) do zmiennej DPB, }
70: (*****)
71: begin
72: Move (Mem [DPBaddr[ord(UpCase(d))-65]], DPB, 27);
73: end; { of Get DPB }
74:
75: procedure SetDPB (d : char; DPB : DPBrec);
76: (*****)
77: { Zadaje tablice DPB napędu "d" (A,B) wartosciami zmiennej DPB, }
78: (*****)
79: begin
80: Move (DPB, Mem [DPBaddr[ord(UpCase(d))-65]], 27);
81: end; { of Set DPB }
82:
83: procedure DD_Read (drive : char; track, sector : byte);
84: (*****)
85: { Wczytuje do bufora "buf" zadany sektor "sector" z wybranego }
86: { napędu "unit" i wybranej ścieżki "track". Jesli operacja }
87: { wykonana jest poprawnie, to zmienna "ok" przyjmuje }
88: { wartość TRUE, ewentualny numer błędu przekazywany jest w }
89: { zmiennej "error", }
90: (*****)
91: var unit : byte; address : integer;
92: begin
93: unit := ord (UpCase (drive))-65;
94: address := DPBaddr [unit];
95: inline ($2A/address/ ( HL := addr(DPB) )
96: $E5/$DD/$E1/ ( IX := HL )
97: $06/$01/ ( B := 1 )
98: $3A/unit/$4F/ ( C := unit )
99: $3A/track/$57/ ( D := track )
100: $3A/sector/$5F/ ( E := sector )
101: $21/buf/ ( LD HL,buffer )
102: $CD/$5A/$FC/$96/$00/ ( CALL USERF $0086 )
103: $32/error/ ( LD (error),A )
104: $3E/$00/ ( LD A,$00 )
105: $32/ok/ ( ok := false )
106: $D0/ ( RET NC )
107: $3E/$01/ ( LD A,$01 )
108: $32/ok); ( ok := true )
109: end; { of DD_Read }
110:
111: procedure DD_Write (drive : char; track, sector : byte);
112: (*****)
113: { Zapisuje sektor dyskowy na określonej ścieżce i w okres- }
114: { lonym napędzie. Zmienne "ok" i "error" mają te same znacze- }
115: { nie co dla poprzedniej procedury. }
116: (*****)
117: var unit : byte; address : integer;
118: begin
119: unit := ord (UpCase (drive))-65;
120: address := DPBaddr [unit];
121: inline ($2A/address/ ( HL := addr(DPB) )
122: $E5/$DD/$E1/ ( IX := HL )
123: $06/$01/ ( B := 1 )
124: $3A/unit/$4F/ ( C := unit )
125: $3A/track/$57/ ( D := track )
126: $3A/sector/$5F/ ( E := sector )
127: $21/buf/ ( LD HL,buffer )
128: $CD/$5A/$FC/$99/$00/ ( Call USERF $0089 )
129: $32/error/ ( LD (error),A )
130: $3E/$00/ ( LD A,$00 )
131: $32/ok/ ( ok := false )
132: $D0/ ( RET nc )
133: $3E/$01/ ( LD A,$01 )
134: $32/ok); ( ok := true )
135: end; { of DD_Write }
136:
137: procedure DD_Check (drive : char; track, sector : byte);
138: (*****)
139: { Sprawdza zawartość bufora z zawartością danego sektora }
140: { dyskowego. Wybierany jest napęd i ścieżka. Jesli porównanie }
141: { wypadnie pozytywnie zmienna "matched" przyjmuje wartość }
142: { TRUE, Znaczenie zmiennych "ok" i "error" takie samo jak }
143: { poprzednio, }
144: (*****)
145: var unit : byte; address : integer;
146: begin
147: unit := ord (UpCase (drive))-65;
148: address := DPBaddr [unit];
149: inline($2A/address/ ( HL := addr(DPB) )
150: $E5/$DD/$E1/ ( IX := HL )
151: $06/$01/ ( B := 1 )
152: $3A/unit/$4F/ ( C := unit )
153: $3A/track/$57/ ( D := track )
154: $3A/sector/$5F/ ( E := sector )
155: $21/buf/ ( LD HL,buffer )
156: $CD/$5A/$FC/$8C/$00/ ( CALL USERF $008C )
157: $32/error/ ( LD (error),A )
158: $3E/$00/ ( LD A,0 )
159: $32/ok/ ( ok := false )
160: $32/matched/ ( matched := false )
161: $D0/ ( RET nc )
162: $3E/$01/ ( LD A,$01 )
163: $32/ok/ ( ok := true )
164: $32/matched/ ( matched := true )
```



```

165:      $08/      ( RET z      )
166:      $3E/$00/  ( LD A,$00   )
167:      $32/matched; ( matched := false )
168: end; ( of DD_Check )
169:
170: procedure DD_Format (drive : char; track, filler : byte);
171: (*****)
172: ( Formatuje zadana sciezke wg informacji okreslonych przez )
173: ( zmienna "header", Zadawany jest takze naped "unit" i bajt )
174: ( wypelniajacy "filler", Przy operacji zakonczonej powodze- )
175: ( niem zmienna "ok" przyjmuje wartosc TRUE, Ewentualny blad )
176: ( przekazywany jest w zmiennej "error" )
177: (*****)
178: var unit : byte; address : integer;
179: begin
180:   unit := ord (UpCase (drive))-65;
181:   address := DPBaddr [unit];
182:   inline($2A/address/      ( HL := addr(DPB) )
183:         $E5/$DD/$E1/      ( IX := HL   )
184:         $06/$01/          ( B := 1    )
185:         $3A/unit/$4F/      ( C := unit  )
186:         $3A/track/$57/     ( D := track )
187:         $3A/filler/$5F/    ( E := filler byte )
188:         $21/header/        ( LD HL,header )
189:         $CD/$5A/$FC/$8F/$00/ ( CALL USERF $008F )
190:         $32/error/         ( LD (error),A )
191:         $3E/$00/          ( LD A,$00   )
192:         $32/ok/           ( ok := false )
193:         $D0/              ( RET nc    )
194:         $3E/$01/          ( LD A,$01   )
195:         $32/ok/           ( ok := true  )
196: end; ( of DD_Format )
197:
198: procedure DD_Login (drive : char);
199: (*****)
200: ( Procedura probuje ustalic format dysku, W przypadku )
201: ( operacji zakonczonej sukcesem inicjalizowana jest tablica )
202: ( DPB napędu, )
203: (*****)
204: var unit : byte; address : integer;
205: begin
206:   unit := ord (UpCase (drive))-65;
207:   address := DPBaddr [unit];
208:   inline ($2A/address/      ( HL := addr(DPB) )
209:         $E5/$DD/$E1/      ( IX := HL   )
210:         $3A/unit/$4F/      ( C := unit  )
211:         $CD/$5A/$FC/$92/$00/ ( Call USERF $0092 )
212:         $32/error/         ( LD (error),A )
213:         $3E/$00/          ( LD A,$00   )
214:         $32/ok/           ( ok := false )
215:         $D0/              ( RET nc    )
216:         $3E/$01/          ( LD A,$01   )
217:         $32/ok/           ( ok := true  )
218: end; ( of DD_Login )
219:
220: function DD_DriveStatus (drive : char) : byte;
221: (*****)
222: ( Funkcja zwraca status danego napędu, )
223: (*****)
224: var status, unit : byte;
225:   address : integer;
226: begin
227:   unit := ord (UpCase (drive))-65;
228:   address := DPBaddr [unit];
229:   inline ($2A/address/      ( HL := addr(DPB) )
230:         $E5/$DD/$E1/      ( IX := HL   )
231:         $3A/unit/$4F/      ( C := unit  )
232:         $CD/$5A/$FC/$98/$00/ ( Call USERF $0098 )
233:         $32/status);      ( LD (status),A )
234:   DD_DriveStatus:=status;
235: end; ( of DD_DriveStatus )
236:
237: procedure DD_ReadID (drive : char; track : byte);
238: (*****)
239: ( Czyta identyfikator pierwszego znalezionego sektora, )
240: (*****)
241: var unit : byte; address : integer;
242: begin
243:   unit := ord (UpCase (drive))-65;
244:   address := DPBaddr [unit];
245:   inline ($2A/address/      ( HL := addr(DPB) )
246:         $E5/$DD/$E1/      ( IX := HL   )
247:         $3A/track/$57/     ( D := track )
248:         $3A/unit/$4F/      ( C := unit  )
249:         $CD/$5A/$FC/$9B/$00/ ( Call USERF $009B )
250:         $32/id/           ( LD (id),A )
251:         $3E/$00/          ( LD A,$00   )
252:         $32/ok/           ( ok := false )
253:         $D0/              ( RET NC    )
254:         $3E/$01/          ( LD A,$01   )
255:         $32/ok/           ( ok := true  )
256: end; ( of DD_ReadID )
257:
258: (*****)
259: (
260: Listing 1, Plik DISK3.SYS
261: )

```



# O OBROTACH DYSKÓW MAGNETYCZNYCH CZ. I

**Pamięci masowe na dyskach magnetycznych od wielu lat stanowią podstawę sprawnego działania maszyn cyfrowych. Także od wielu lat stanowią standardowe wyposażenie komputerów osobistych, wyraźna jest też tendencja do wyposażania w nie komputerów domowych. Warto więc przyjrzeć się podstawowym problemom związanym z ich eksploatacją, a zacząć proponuję od krótkiego przypomnienia i przeglądu podstawowych zasad budowy i działania, oraz podstawowych parametrów tych urządzeń.**

Pamięć dyskowa to urządzenie zapisujące zero-jedynkowe dane (bity) na aktywnej magnetycznie powierzchni wirującego krążka (czyli dysku i stąd nazwa). Zapis i odczyt informacji wykonuje głowica magnetyczna, umieszczona w bezpośredniej bliskości wirującej powierzchni. Jeśli głowica stoi nieruchomo, to część powierzchni dysku przesuwaną się pod nią tworzy okrąg, który zwykle nazywamy ścieżką (ang. track). Ustawiając głowicę w innym miejscu uzyskamy inną ścieżkę. Ponieważ na każdej ze ścieżek chcemy wykonywać oddzielne zapisy, więc pomiędzy ścieżkami (czyli między kolejnymi położeniami głowicy) musi być pewien odstęp, wystarczająco duży aby uniknąć zakłóceń. Stąd widać, że ścieżek na dysku nie można mnożyć nieograniczenie — ich liczba zawsze jest zdeterminowana konstrukcją konkretnego urządzenia. Nietrudno się również domyśleć, że im mniejsza głowica i im większy dysk, tym więcej ścieżek da się umieścić a w konsekwencji więcej danych można będzie zapisać. Jednak miniaturyzacja głowic ma swoje granice, a rozmiarów dysku też nie można dowolnie powiększać. Im większy dysk tym trudniej będzie go napędzać, a w skraj-

nym przypadku, gdy stworzymy dysk zdecydowanie za duży to może go rozzerwać na kawałki siła odśrodkowa.

Dlaczego więc zrobiliśmy milczące założenie, że dyski mają się kręcić jak najszybciej? Każda ścieżka zawiera określoną ilość informacji — im szybciej wiruje dysk tym szybciej cała ścieżka przesunie się pod głowicą i jej zawartość może zostać zapisana lub odczytana. A jako użytkownicy jesteśmy przecież zwykle bardzo niecierpliwi.

Powyższe wielkości — pojemność i szybkość to podstawowe dla użytkownika cechy dysku. Pojemność podajemy zwykle jako ilość danych które może pomieścić dysk. Ilość tę mierzymy w kilobajtach (1KB=1024 bajty) lub megabajtach (1MB=1K\*1 KB=1024\*1024 bajty=1048576 bajtów). Szybkość najczęściej określamy podając tzw. czas dostępu. Jest to czas, który upływa od momentu zlecenia transmisji do jej wykonania. Podawany jest w milisekundach.

Z opisanymi cechami wiąże się pierwszy dylemat pamięci dyskowych: jak już zauważyliśmy wymagania dużej szybkości i dużej pojemności domagają się dokładnie przeciwnych rozwiązań. Oczywiście możemy łatwo znaleźć wyjście dla konstruktorów pamięci dyskowych: zwiększyć precyzję zapisu, zmniejszyć rozmiary głowicy i ścieżki. Mówiąc bardziej fachowo chcemy zwiększać gęstość zapisu (ang. recording density), czyli liczbę bitów zapisywanych na jednostce długości ścieżki. Jednak i na tej drodze znajdują się bariery. Pierwszą, zasadniczą, jest fakt, że cząsteczki z których zbudowana jest materia mają określone rozmiary i tych rozmiarów nie da się zmniejszyć (w popularnych urządzeniach dyskowych do tej bariery jeszcze daleko). Dużo większe znaczenie praktyczne ma w tej chwili ograniczenie drugie — koszty. Urządzenie precyzyjniejsze jest przez początkowy okres sprzedaży droższe od tradycyjnych rozwiązań. W ten sposób w naszych rozważaniach pojawił się trzeci czynnik — cena.

Ponieważ odpowiedź na pytanie: który dysk jest lepszy, szybki, pojemny i bardzo drogi, czy wolny, mały i tani, zależy od sytuacji i potrzeb użytkownika, więc producenci oferują całą gamę konstrukcji, a wyboru dokonuje kupujący. Dla przykładu, kompletne



# GENERATOR LICZB PSEUDOLOSOWYCH

urządzenie pamięci typu winchester, o pojemności 20MB (megabajtów) i czasie dostępu 65ms można na Dalekim Wschodzie kupić za ok. 200 dolarów USA, ale za winchester o pojemności kilkuset MB i czasie dostępu rzędu kilkunastu ms trzeba będzie zapłacić nawet kilka tysięcy dolarów. Stacje dysków elastycznych 5.25 cala, średniej jakości można kupić już za około 50—60 dolarów. Podane ceny są cenami detalicznymi — można się spodziewać, że w hurcie ceny są niższe. Są to ceny samych urządzeń pamięci, przewidzianych do zabudowy w zestawach komputerowych, a więc bez własnego zasilacza, czy obudowy.

Rozszyfrujemy jeszcze podane wyżej nazwy. Stacje dysków elastycznych (ang. floppy disk drive) to urządzenia pracujące na nośnikach (dyskietkach) wymiennych. Dzięki wymiennym nośnikom łatwo przenosić dane między maszynami, łatwo też tworzyć archiwa. Za to urządzenia nie mogą być bardzo precyzyjne i stąd pojemność oraz szybkość dyskietek jest niezbyt rewelacyjna. Typ stacji określony jest przede wszystkim przez wielkość używanej dyskietki. Nośniki o średnicy 8 cali praktycznie wychodzą z użycia. Wraz z rodziną IBM PC ogromnie rozpowszechniły się dyskietki 5.25 (inaczej 5 1/4) cala. Mają one pojemność 360KB lub 1.2MB, w zależności od gęstości zapisu. Od kilku lat wzrasta popularność dyskietek 3.5 cala. M. in. zastosowała je firma IBM w swej nowej rodzinie komputerów osobistych — PS/2. Typowe pojemności tych dyskietek to 720KB lub 1.44MB.

Pamięci typu winchester (często używany jest również angielski termin hard disk drive) mają nośnik magnetyczny (zwykle jest to kilka równoległych do siebie dysków osadzonych na wspólnej osi) ukryty w hermetycznie zamkniętej obudowie. Pozwala to znacznie zwiększyć precyzję działania, a więc i poprawić parametry urządzenia. Przykładowe dane dla dysku, którego fotografia (bez fabrycznej obudowy!) została zamieszczona, to prędkość obrotowa 3600 obr/min, gęstość upakowania ścieżek 1033/cal (ang. tpi czyli track per inch), gęstość zapisu na ścieżce 10200 bitów/cal (bpi — bits per inch), średni czas między awariami 30000 (trzydzieści tysięcy!) godzin, planowana żywotność pięć lat. Pojemność tego urządzenia wynosi 53.3megabajta, wymiary 41x146x203mm (jest to wielkość pozwalająca umieścić stację w typowej obudowie IBM PC, w jednej z czterech komór przeznaczonych na pamięci dyskowe), waga 1.4 kg.

Wróćmy teraz do organizacji pracy pamięci. Zarówno to co powiedzieliśmy do tej pory o zasadach działania jak i to co jeszcze przed nami, jest na tyle ogólne, że dotyczy wszystkich typowych pamięci dyskowych.

Pojemność ścieżki jest za duża aby wygodnie było nią operować, dlatego dzielimy ścieżki na równej długości fragmenty nazywane sektorami (ang., sector). Sektor jest jednostką dla urządzenia dyskowego niepodzielną — potrafi ono odczytać lub zapisać tylko cały sektor, nie ma możliwości przeczytania kilku bajtów czy połowy sektora.

Zatrzymajmy się tu na chwilę. Jak w takim razie poprawić dane na dysku? Jeśli mamy np. zapisane w jednym sektorze nazwiska oraz numery telefonów dwudziestu osób i jedna osoba zmienia telefon, to co zrobić? Program, który ma tej poprawki dokonać musi przeczytać do pamięci operacyjnej zawartość całego sektora (kolejne bajty z dysku zajmą kolejne bajty pamięci),

tam już można zmieniać każdą daną z osobna, następnie cały poprawiony obszar pamięci operacyjnej o długości równej długości sektora przesłać z powrotem na dysk.

A jak określane jest właściwe miejsce na zapis i odczyt? Tu właśnie przydaje się podział na ścieżki i sektory. Pozwala on określić jednoznacznie adres transmisji poprzez podanie numeru ścieżki i numeru sektora na ścieżce. To od strony dysku, od strony pamięci operacyjnej podajemy po prostu adres pierwszego bajtu spójnego obszaru przeznaczanego na wpisanie odczytanego z dysku sektora (lub zawierającego dane przeznaczone do zapisu).

Zastanówmy się teraz ile czasu musimy czekać na wykonanie przez pamięć dyskową odczytania wybranego sektora. Otóż po otrzymaniu adresu muszą zostać wykonane następujące czynności: należy przesunąć głowicę nad właściwą ścieżkę, a następnie poczekać aż znajdzie się pod głowicą potrzebny sektor. Cała zawartość sektora jest czytana podczas jednokrotnego przejścia dysku pod głowicą. O czasie wykonania operacji decyduje tu nie czas czytania sektora, lecz przede wszystkim czas przesuwania głowicy ze ścieżki na ścieżkę, w znacznie mniejszym stopniu czas oczekiwania na sektor i dopiero na końcu samo czytanie. Zapamiętajmy ten podział czasochłonności, gdyż będziemy się jeszcze do niej odwoływać.

Skąd biorą się na dyskach ścieżki i sektory? Powierzchnia fabrycznie nowego dysku (czy dyskietki) jest zupełnie jednorodna. Aby właściwie lokalizować odpowiednie miejsca należy nanieść (zapisać) na powierzchni dysku odpowiednie oznaczenia całej omówionej powyżej struktury ścieżek i sektorów. Taka operacja nazywa się formatowaniem dysku. Zwykle system operacyjny mikrokomputera daje użytkownikowi możliwość samodzielnego formatowania dysków. W związku z tą operacją warto zapamiętać dwie rzeczy: wyjęty z fabrycznego opakowania dysk czy też dyskietka nie nadają się do zapisu danych dopóki nie zostaną sformatowane; i po drugie, skoro formatowanie tworzy całą strukturę sektorów na powierzchni dysku zupełnie od nowa, to formatować można również nośniki używane, z zapisanymi danymi. I wtedy cała poprzednia zawartość zostaje bezpowrotnie usunięta! Warto o tym pamiętać, gdy omyłkowe zaformatowanie dysku zawierającego potrzebne informacje jest chyba częstszym zjawiskiem niż utrata danych spowodowana awarią sprzętu.

W powszechnych zastosowaniach pamięci dyskowych istnienie głowic, ścieżek czy sektorów niewiele interesuje użytkownika, gdyż zarządzanie zasobami dyskowymi wykonuje dla niego system operacyjny komputera. Informacje przeznaczone do przechowania na dysku grupowane są w tzw. zbiory (ang. file). Chcąc np. utworzyć zbiór, skopiować go lub usunąć, użytkownik wydaje systemowi operacyjnemu odpowiednie zlecenie, a ten przekłada je na język ścieżek i sektorów, i wykonuje. Jednak problemy związane z użytkową stroną pamięci dyskowych pozostawimy do omówienia za miesiąc.

*Andrzej Krul*



## W wielu programach trzeba korzystać z liczb losowych.

Ponieważ prawdziwy generator liczb losowych musiałby być rozwiązaniem sprzętowym, a nie programowym, w komputerach najczęściej stosuje się generatory liczb pseudolosowych. To „pseudo” oznacza, że znając sposób ich generacji, możemy bez trudu przewidzieć jaka będzie kolejność ich występowania, zwłaszcza że wartość następnej jest liczona na podstawie wartości poprzedniej. Taki jest na przykład generator liczb losowych w Spectrum. Zastosowana tam reguła iteracyjna generuje ciąg wszystkich liczb z zakresu 0-65535 w „losowej” kolejności (funkcja RND zwraca wynik dzielenia kolejnej liczby losowej przez 65536, dlatego uzyskiwane przy jej pomocy liczby mieszczą się w przedziale <0,1). W generowanej serii każda liczba występuje dokładnie jeden raz, potem ciąg się powtarza (jego pseudolosowość można zbadać w bardzo ciekawy sposób, uruchamiając na kilka godzin następujący program — 10 PLOT 255\*RND,175\*RND: GO TO 10 — gdyby liczby były naprawdę losowe, po odpowiednio długim czasie cały ekran byłby czarny.)

Stosowany w generatorze Spectrum schemat iteracyjny (opisany równaniem  $r_{i+1} = (75(r_i + 1)) \bmod 65537 - 1$ ) wymaga prowadzenia obliczeń przy użyciu liczb całkowitych 24 bitowych — największa liczba wchodząca w grę jest bowiem równa 4915275. Z tego powodu trudne byłoby zaimplementowanie tego generatora podczas pisania programu w kodzie maszynowym. Wprowadźmy gdy potrzebna jest tylko jedna liczba losowa, można skorzystać z zawartości rejestru R procesora, ale ulega ona zmianie raz na kilkanaście instrukcji, toteż gdy potrzebne są dwie liczby losowe jedna po drugiej, ryzykujemy że obie będą takie same. Te same uwagi odnoszą się do kompilatora Pascala HP4S, który dysponuje bardzo szybką funkcją RANDOM, opartą właśnie na rejestrze R. Jeszcze gorzej sytuacja wygląda w przypadku kompilatora HiSoft C, który nie dysponuje wbudowanym generatorem, a biblioteczny korzysta ze specjalnie przygotowanych procedur arytmetyki 32 bitowej, skutecznie spowalniających jego działanie.

W przedstawionej sytuacji nie pozostaje nic innego do zrobienia, jak opracowanie nowego generatora, który nie miałby wymienionych wad. Zdecydowałem się na rozwiązanie opisane następującym wzorem:

$$r_{i+1} = (5(r_i + 1)) \bmod 256$$

(przypominam, że mod oznacza resztę z dzielenia pierwszego argumentu przez drugi). Taki generator ma wprawdzie okres równy zaledwie 256, ale do wielu zastosowań jest zupełnie wystarczający, w dodatku bardzo łatwo go zaimplementować we wszystkich językach o których była mowa.

Proponuję trzy wersje generatora — w C, w Pascalu i w kodzie maszynowym (liczba losowa zwracana jest w rejestrze A, może się okazać, że w konkretnym programie po-

trzebne jest przechowywanie wartości rejestrów HL i DE). We wszystkich trzech przypadkach RND jest nazwą samego generatora, a INIRND to procedura inicjująca (INIRND w przypadku asemblera).

Jeżeli okres równy 256 wydaje się komuś zbyt krótki, można się pokusić o jego zwiększenie — zmieniając w podanym wzorze liczby 5 i 256 na inne. Spowoduje to drobne komplikacje, gdy wynik przestanie się mieścić w jednym bajcie, tym niemniej jest możliwe.

Żeby nie być gołosłownym w twierdzeniu że prezentowany generator może (po odpowiednim potraktowaniu) być zupełnie wystarczający, powołam się na wyniki uzyskane w trakcie pisania artykułu „Z chaosu porządek”, który będzie drukowany w Bajtku. Podczas przygotowywania (przy pomocy kompilatora Turbo C 2.0 na IBM PC) rysunków do tego artykułu początkowo korzystałem z wbudowanego generatora liczb losowych — i czekałem na jeden rysunek kilkanaście godzin. Po zastosowaniu prezentowanego generatora, program zaczął działać dwa razy szybciej, przy takiej samej jakości wyników.

*Marcin Borkowski*

```

/*      HiSoft C      */
static char seed;

char rnd(ogr)
char ogr;
{
    seed = 5 * ++seed;
    return( seed % ogr );
}

inirnd()
{
    inline(0xED, 0x5F, 0x32, &seed);
}

{      HiSoft Pascal HP4S      }

VAR
SEED : INTEGER;

FUNCTION RND : INTEGER;
BEGIN
    SEED := (5*(SEED+1)) MOD 256;
    RND := SEED
END;

PROCEDURE INIRND;
BEGIN
    SEED := RANDOM
END;

GENS - wersja assemblerowa.
100 RND    LD    A, (SEED)
110      LD    L, A
120      LD    H, 0
130      INC  HL
140      LD    E, L
150      LD    D, H
160      ADD  HL, HL
170      ADD  HL, HL
180      ADD  HL, DE
190      LD    A, L
200      LD    (SEED), A
210      RET
220 INIRND LD    A, R
230      LD    (SEED), A
240      RET
250 SEED  DEFB  0

```



New

New

New

# Drukarka Star

## to najlepszy przyjaciel Twojego Komputera!



### STAR LC — 10

Najpopularniejsza drukarka świata! Ponad 2.000.000 użytkowników nie może się mylić! Jeśli chcesz mieć drukarkę solidną i niezawodną, wszechstronną i niedrogą, to decyzja jest prosta: LC-10!

Prędkość druku:

144zn/s (draft)  
36zn/s (NLQ)

### STAR LC — 10 colour

Ta sama japońska jakość i niezawodność jak LC-10, z dodatkową możliwością druku w 7 kolorach!

Wystarczy tylko dokończyć barwną taśmę.

Prędkość druku:

144zn/s (draft)  
36zn/s (NLQ)

### STAR LC24 — 10

Drukarka roku 1989 w Europie Zachodniej! Doskonała jakość pisma dzięki nowoczesnej, 24-igłowej głowicy.

Niezastąpiona do korespondencji oraz precyzyjnej grafiki.

Prędkość druku:

170zn/s (draft)  
57 zn/s (LQ)

Wszystkie trzy drukarki posiadają cztery kroje czcionek, wbudowany traktor oraz funkcje „Paper Park”. Można w nich stosować papier z perforacją lub bez, jak również pojedyncze kartki.

**UWAGA: W celu zainstalowania polskich znaków, prosimy skontaktować się z jednym z naszych punktów serwisowych!**

Ceny: LC-10 DM 350, kasetę barwiącą (czarna) DM 7,20  
LC-10 colour DM 450, kasetę barwiącą (kolorowa) DM 14,-  
LC24-10 DM 600, kasetę barwiącą (czarna) DM 11, karta z polskimi znakami DM 100

#### Oto adresy naszych składów celnych i punktów serwisowych:

ABC Data CI CUP, Warszawa, ul. Żurawia 4a, tel. 21.75.08  
ABC Data Service/Take, Warszawa, ul. Konopnickiej 6, tel. 28.92.81  
ECS, Warszawa, ul. Połczyńska 96, tel. 36.82.50  
ZIPO, Gdańsk, ul. Krynicka 1, tel. 41.82.75

Interbit, Kielce, ul. Manif. Lipcowego 4, tel. 441.99  
Sykomat, Kraków, ul. Skawińska 11, tel. 21.95.40  
Techmex, Bielsko Biała, ul. M.C. Skłodowskiej 13, tel. 421.98

**star**  
the ComputerPrinter

**ABC Data**

Przedstawiciel na Polskę: ABC Data GmbH, .....  
Bürgerstraße 12, 5300 Bonn 2. RFN.  
Tel: 0228 354480/90, Tlx: 172283746, Fax: 0228 355635



**Cieślakowski i s-ka**  
PRZEDSIĘBIORSTWO HANDLOWO-USŁUGOWE  
00-446 WARSZAWA UL. FABRYCZNA 2/103

☎ 29-89-31  
OFERUJEMY PERYFERIA

**AMSTRAD**

**ATARI ST  
A MIGA**

Stacje dysków 5.25"

Rozszerzenia pamięci

Modulatory TV

Kontroler stacji dysków CPC 464

Karta EPROM-ów CPC

RS 232 CPC

RS-CENTRONICS PCW

INTERFEJS joystick'a do PCW

8-bitowy CENTRONICS CPC

VIDEO DIGITIZER ST

PROGRAMATOR EPROM-ów

**SUPER  
OFERTA**

**SUPER  
KATALOG**

**ATARI XL/XE**

- największy wybór opisów do gier i programów użytkowych: literatura
- gry, programy narzędziowe, użytkowe, polskie programy edukacyjne
- interfejsy do magnetofonów w systemie standardowym. ATARI SUPER TURBO. TURBO 2000. Przeróbki magnetofonów firmowych
- TOP DRIVE — samodzielny montaż
- interfejsy CENTRONICS do drukarek

**COMMODORE**

- kartridże X, FINAL II, FINAL III
- Sprzedaż wysyłkowa — gwarancja
- Studio komputerowe

**MEGABAJT**

03-945 Warszawa  
skr. poczt. 28, tel. 17-76-16

B14

**ATARI  
XL/XE**

- 2000 programów na kasety i dyskietki
- nowości i trudno dostępne użytki
- rewelacyjnie niskie ceny
- co piąty program bezpłatnie
- bonifikaty dla stałych klientów
- rachunki i trzymiesięczna gwarancja
- katalog z pełnym wykazem — gratis
- wysyłka na cały kraj

**ART-SOFTWARE  
66-542 Zwierzyn P-1**

reklamacja D-46

**MÓZGPROCESOR!**

to rewelacyjna polska gra przygodowa firmy COMPUTER ADVENTURE STUDIO — dla SPECTRUM, TIMEX, JUNIOR (taśma + opis) — dla ATARI XL/XE (dyskietka + opis)

Cena zestawu — równowartość 2 USD

UWAGA! Test w Bajtku nr 10/89  
Zamówienia prosimy kierować:  
COMPUTER ADVENTURE STUDIO  
Bochnia 32-700,  
Kazimierza Wielkiego 37/45  
tel. (0-197) 242-47

B4

**joy**

**Wysyłka natychmiastowa  
za zaliczeniem pocztowym**

**JOYSTICKI** do Atari Commodore, Spectrum, Amstrad; precyzyjny mechanizm specjalne styki, 6 m-cy gwarancja interface do Spectrum

**ELEKTROMECHANIKA**

ul. Cegielniana 17  
32-410 DOBCZYCE

B-9

**ATARI 800 XL,  
65 XE, 130 XE**

Sprzedaż wysyłkowa gier i programów użytkowych na kasetach i dyskietkach. Również w systemie TURBO 2000

Wszystkie nowości!!!  
Instrukcje i literatura.  
Dla zainteresowanych rachunki.

**ANWIKOL**

03-721 Warszawa ul. Jagiellońska 3/28.

(SB 74)

**ELBOX-SERVICE**

poleca naprawy mikrokomputerów ZX SPECTRUM, COMMODORE, TIMEX  
Kraków tel. (0-12) 22-36-39  
w godzinach 11-13 od poniedziałku do środy  
skr. poczt. 536, 30-960 Kraków 1  
(SB 80)

Atari XL/XE najnowsze programy, instrukcje oraz literaturę wysyła studio Mikrobot. Okazja, szybko i tanio. Marek Kąca, Malborska 6/160, 03-286 Warszawa.  
(SB 75)

**MICROMAN**

oferuje:

1. Programy i literaturę dla komputerów Atari XL/XE, Commodore 16/116/+4 Commodore 16/116/4/64 Amige, Spectrum/Timex na miejscu lub za zaliczeniem pocztowym.
  2. Dla użytkowników ELWRO 800 program kopiujący taśma — dysk.
  3. Przystawki UNIWERSAL TURBO dla magnetofonów Atari XC12, umożliwiające zapis i odczyt programów zarówno w systemie BLIZZARD jak i TURBO 2000. UNIVERSAL TURBO to połączenie dwóch najpopularniejszych systemów transmisji w Polsce, to przy tym samym koszcie możliwość wyboru.
  4. Naprawę klawiatur dla komputerów ZX Spectrum.
  5. Dodatkowe akcesoria dla komputerów domowych.
- Informacje na miejscu lub za zaliczeniem koperty zwrotnej i znaczka.  
Adres: **MICROMAN**  
40-181 Katowice  
ul. Osikowa 66  
tel. 585-106

B6

Sklep firmowy „Bajtek” poleca szeroki wybór — mikrokomputerów — oprzyrządowania — oprogramowania — wydawnictw

Prowadzimy również dział sprzedaży komisowej na atrakcyjnych warunkach. **ZAPRASZAMY!**

Bytom ul. Koniewa 6 tel. 81-57-01  
Kraków ul. Pstrowskiego 9 tel. 56-54-52

(SB 78)

**COMPUTER — SERVICE**

Naprawy komputerów Commodore, IBM, Spectrum, Timex oraz serwis i przeróbki zasilacze monitorów, drukarek Instalacje polskich znaków Kraków tel. (012)339651  
Pon.-piątek  
godz. 10-13, 20-21

B15

**Agencyjny Zakład Usługowy SPHW poleca usługi w zakresie:**

- serwis komputerów SPECTRUM, C-64, C+4, Timex, XT, AT, zasilacze komputerowe
  - przestrajanie PAL SECAM
  - wejścia monitorowe TV
- Warszawa ul. Puławska 102 tel. 448789  
czynny 12.00-19.00. Rachunki, gwarancja.**

D-233

„Taśmy w kasetach (czarne i czerwone) do drukarek Commodore 801 Seikosha GP500 i 550A, Atari 1029 za zaliczeniem sprzedam. **Jan Juchec, 47-330 Zdzieszowice ul. Dunikowskiego 7.**  
SB94

Serwis  
Komputerów Spectrum  
Lubiniec  
tel. 18-95 wieczorem

SBx

Mieszkańcy wsi i miasteczek. SDS Atari — programy tanio, szybko (koperta + znaczek) Opole, Krajewskiego 19A/603.  
SA1



	Gielda	Sklep Bajtka	Komis	Pewex	RFN	Baltona	CSH i inne
	tys. zł	tys. zł	tys. zł	\$	DM	\$	tys. zł
<b>SINCLAIR</b>							
ZX 81	200	—	—	—	—	—	—
ZX Spectrum 48	700	800	750	—	80	—	—
ZX Spectrum +	800	900	850	—	90	—	—
Timex 2048	800	950	860	—	—	—	—
ZX Spectrum 128+	1000	—	—	—	—	—	—
ZX Spectrum 128+2	1100	—	—	—	140	—	—
ZX Spectrum 128+3	—	—	—	—	280	—	—
drukarka Seikosha GP 50s	—	—	—	—	—	—	—
Interface Kempston	30	25	—	—	10	—	20

## COMMODORE

Commodore 64	1300	1400	1500	199	290	155	—
VC 20	—	—	—	—	—	—	—
C 16	400	—	—	—	80	—	—
C 116	—	700	—	—	70	—	—
C Plus 4	1000	1100	1200	—	150	—	—
C 128	1800	2000	—	—	399	—	—
C 128 D	2200	2800	—	450	820	—	—
Amiga 500	4500	5000	—	500	899	—	—
Magnetofon 1531	300	330	330	48	30	225	—
Stacja dysków Oceanic	—	—	—	—	320	170	—
Stacja dysków 1571	1800	1900	—	199	460	—	—
Drukarka LC10C	2500	2700	—	—	260	230	—

## ATARI

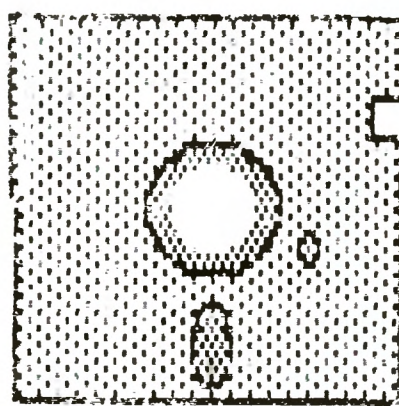
Atari 800 XL	1100	1200	1200	—	—	—	1300
Atari 65 XE	1200	1400	1250	127	—	—	—
Atari 130 XE	1700	1300	—	199	220	—	—
Atari 520 ST	4000	4200	—	—	—	—	—
Atari 1040 ST	—	—	—	—	1140	—	—
Magnetofon XC 12	300	350	—	36	40	—	—
Stacja dysków 1050	1500	—	1700	185	300	—	—
Stacja dysków 520 STM	—	—	—	—	—	—	—
Drukarka 1029	700	770	—	—	—	—	—

## AMSTRAD

Amstrad 464 mono.	1300	1400	—	—	350	—	—
Amstrad 664 mono.	—	2000	—	—	—	—	—
Amstrad 6128 mono.	—	—	—	—	670	—	—
Amstrad PCW 8256	—	—	—	—	—	—	—
Amstrad PCW 8512	—	—	—	—	—	—	—
Amstrad PCW 9512	—	—	—	—	—	—	—
Stacja dysków do 464	—	—	—	—	380	—	—

## SHARP

Sharp MZ 700	—	1000	—	—	—	—	—
Sharp MZ 800	—	1200	—	—	—	—	—
Dyskietki 5.25 cala	3.5	3.5	4	1	0.7	11	6
Dyskietki 3.5 cala	10	12	12	—	5	2.5	20
Dyskietki 3 cale	20	22	—	—	6	3	25
Joystick	70	70	70	5	10	10	70
Monitor Neptun	500	—	—	—	—	—	—



INDYWIDUALNY  
**BANK**  
DANYCH

**Marcin Byszewski**, lat 15. Posiada Amstrada CPC 6128 Oprogramowanie: około 150 gier i programów użytkowych. Zainteresowania: informatyka, muzyka, pisanie programów. Proponuje wymianę programów oraz gier. Adres: 05-807 Podkowa Leśna ul. Helenowska 3.

**Tomasz Bluszcz**, posiada komputer Acorn Compest 128, nawiąże kontakt ze wszystkimi posiadaczami tego typu komputera celem wymiany doświadczeń, literatury i oprogramowania. Adres: 01-489 Warszawa, ul. S. Kaliskiego 217B/214B.

**Piotr Sielecki**, lat 10. Posiada Atari 65XE i magnetofon komputerowy XCA12. Nawiąże kontakt z innymi posiadaczami Atari w celu nauki programowania. Adres: 06-402 Ciechanow, ul. Marii Dąbrowskiej 20.

**Artur Osuchowski**, lat 15. Posiada Atari 520 STFM z wbudowaną stacją 1Mb (około 100 programów). Pragnie nawiązać kontakt z posiadaczami tego komputera w celu wymiany oprogramowania i doświadczeń. Adres: 22-300 Krasnystaw, Plac 1-go Maja 11/7.

**Andrzej Burkowski**, lat 41 posiada Commodore 64 z pełnym osprzętem. Zajmuje się współpracą komputera z elektronicznymi instrumentami muzycznymi. Prosi kolegów o kontakt z kolegami o podobnych zainteresowaniach. Adres: 41-400 Mysłowice, ul. Janowska 40a.

**Wojtek Trzaska**, lat 15. Posiada C64 z magnetofonem i 1500 programów. Pragnie nawiązać kontakt w celu wymiany oprogramowania. Adres: 41-800 Zabrze, ul. Łokietka 5/3.

**Mateusz Korniak**, lat 15. Posiada C-64 wraz z PD-1541 oraz drukarkę MPS-1200. Pragnie nawiązać kontakt w celu wymiany oprogramowania. Adres: 41-800 Zabrze, ul. Michała Archanioła 9/2.

**Kazanow Aleksander**, lat 40. Posiada ATari 130c, stacja dysków magnetofon XC12, TURBO 2000 i Commodore 64 + stację dysków 1541 i magnetofon. Wymiana programów i literatury, opisy gier i schematów elektronicznych i cartridge. Posiada około 3000 programów. Adres: 210-026 Witebsk Białoruska CCP POST BOX 61 210026 kod. (korespondencja w języku rosyjskim, polskim, angielskim).

**Leszek Komendera**, lat 16. Posiada komputer Texas Instruments TI99/4A + firmowy magnetofon oraz literaturę w języku niemieckim i angielskim. Pragnie nawiązać kontakt z innymi posiadaczami tego komputera w celu wymiany doświadczeń i opro-

gramowania. Poszukuje programów dyskowych pod CP/M na ZX Spectrum — stacja FDD3000. Adres: 43-100 Tychy, ul. Rapackiego 16/3.

**Rafał Ristan** lat 17. Posiada Commodore C-64, stację dysków 1541 II, mysz. Zainteresowania: sport i muzyka. Pragnie nawiązać kontakt w celu wymiany oprogramowania. Korespondencja w języku polskim lub niemieckim. Adres: 306Q Stadfhagen, Falkenweg 4, RFN.

**Wiecko Marek**, lat 22. Interesuje się informatyką, posiada Commodore C-64, stację dysków i drukarkę, oraz wiele gier. Proponuje wymianę doświadczeń i oprogramowania. Korespondencja w języku polskim. Adres: Heidebloeck 10, 2000 Hamburg 74 BRD.

**Marek Stanislav**, posiada Spectrum i około 2000 programów, gier. Pragnie nawiązać kontakt w celu wymiany. Adres: Loučovice 108, 382-76 CSRS.

**Martin Bruncko**, lat 14. Posiada Atari 130 XE z magnetofonem. Oprogramowanie, około 200 gier i 80 programów użytkowych. Nawiąże kontakt z posiadaczami Atari w celu wymiany literatury i oprogramowania. Adres: Odbojarov 1956/5-18, 026 01 Dolny Kubin, CSRS.

**Robert Heber**, lat 17. Posiada ZX Spectrum i około 1100 gier i programów użytkowych. Nawiąże kontakt z osobami znającymi Assemblera Z80. Adres: 41-910 Bytom, ul. Chorzowska 20/42.

## ATASERW

43-100 Tychy ul. Lenczewicza 46/3  
tel. 27 69 66

oferuje świetne rozwiązania  
sprzętowe

do ATARI XL/XE:

1. TURBO DOS — wspianały DOS na kartridżu
  2. TOP DRIVE — do stacji 1050, LDW 2000, CALIFORNIA samodzielny montaż — wysyłkowo (rec. INFORMIK III/88)
  3. INTERFEJS CENTRONIKS
  4. ROZSZERZENIA PAMIĘCI
  5. BASIC XE — kartridż
  6. TURBO DOS + BUG65 — kartridż
- 12 miesięcy gwarancji. Informacje i zamówienia telefonicznie (wtorek 8—12, środa, czwartek 16—18 i listownie po otrzymaniu koperty zwrotnej B16

Programy C-16, C-116, C plus/4 wysyłam pocztą. Katalog gratis po otrzymaniu koperty zwrotnej. Nagrywanie Programów Komputerowych. Adres: ul. Lenina 104/3 58-304 Wałbrzych. Bd5

Programy na Atari tanio! Robert Hysa ul. Miła 10/19 41-250 Czeładź. SB95

Mikroservice Commodore-64/128 Amigą, Spectrum-PC/XT/AT- Atari CARTRIDGE 01-911 Warszawa, Anderse-na 3/103. Bd3

Amiga Computer -Studio- skrytka 12 78-401 Szczecinek 3 poleca szeroki wybór programów (gier użytkowych) na Amigę 500 i 2000. Informacje gratis! Napisz do nas jeszcze dziś!!! SB84



SERWIS KOMPUTERÓW

# TEST

40-164 Katowice ul. Modrzewiowa 24/33  
poleca naprawy:

- ATARI 600, 800, 65, 130 XL, XE
- COMMODORE 16, 116, +4, 64, 128
- DISC DRIVE 1541, 1570, 1571, 1050
- MAGNETOFONY COMMODORE
- DRUKARKI

godz. 9-11, 16-18

Rozszerzenie pamięci Commodore 16/116 Atari 600 XL do 64 kilobajtów.

Atari 800 XL 65XE do 130 kilobajtów

(SB 30)

ASL  
2000

# PROSTER

## ANALIZATOR STANÓW LOGICZNYCH

- 16, 32, 48 lub 64 kanały
- sondy typu VAR lub TTL
- pamięć: 2046 stanów/kanał
- zegar wewnętrzny do 80MHz
- wykonanie: terminal PC kompakt z PC

PROSTER

Gliwice 44-101, ul. Zygmunta Starego 6  
tel. 31-48-60, 31-90-21, 32-06-32  
tlx 036315, 036322

ASL  
2000

(SB-2)

## KOMPUTER NATYCHMIAST KUPISZ-SPRZEDASZ

# MAXSOFT

659-44-17 Warszawa

(SB-87)

# TURBO PASCAL

## na ZX Spectrum

Proponujemy Użytkownikom Spectrum i Timexa ze stacją dysków FDD3000 rewelacyjny kompilator języka Pascal — TURBO PASCAL 3.0 w systemie CP/M, opracowany przez profesjonalistów. Dostarczamy bogate biblioteki procedur — obsługi grafiki, muzyki, okien, pamięci zewn., drukarki oraz przykładowe programy. TURBO PASCAL to światowy standard kompilatorów — międzynarodowe narzędzie edukacji. W wersji na Spectrum wzbogacony m.in. o grafikę i dźwięk. To niepowtarzalna okazja programowania współbieżnego — do dyspozycji dwa procesory i 112 K RAM.

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

B12

Najtańsze programy na Spectrum. Zbudniewek Jacek ul. Orkana 10/25 Skierniewice SB92

Spectrum-Timex. Programy tanio. Jarosław Szarski ul. Kilińskiego 27/17 96-300 Żyrardów. SB93

# Drogi Bajtku!

**Mam duży problem z wpisywaniem nieśmiertelności do gier na Spectrum. POKE-i, które podajecie można w łatwy sposób wpisać, gdy loader gry jest w Basicu. A co zrobić, gdy ładowanie jest z poziomu asemblera? (...)**

Jacek Ślęzała, Wrocław

Standardowe procedury ładujące w asemblerze mają postać:

LD IX, start  
LD DE, długość  
LD A, 255  
SCF  
CALL 1366  
JP uruchomienie

start i długość to parametry wczytanego (z taśmy) pliku, liczba w akumulatorze wskazuje na typ pracy procedury — ładowanie, zaś skok pod adres 1366 to wczytanie z taśmy bloku bez nagłówka o długości podanej i od podanego adresu. Następnie wykonywany jest skok pod odpowiedni adres, powodujący wystartowanie wczytanego właśnie programu. Należy zaznaczyć, że w wypadku błędu nastąpi powrót z procedury 1366 z ustawieniem przeniesienia. W naszym przypadku wykonanie ostatniego skoku może spowodować restart systemu na skutek przekłamania kodu programu.

Oczywiście wczytanie bloku z nagłówkiem to dwie procedurki takie, jak powyższa. Pierwsza wczyta 17 bajtów (długość nagłówka), zaś druga resztę pliku, na podstawie zawartych w nagłówku danych, lub też niezależnie od nich. Przed wczytaniem reszty pliku można też sprawdzić nazwę napotkanego pliku itp.

Tak więc bariera do przeskoczenia przy wpisywaniu nieśmiertelności jest ładowanie. Wystarczy pierwszy bajt skoku (195) w loaderze (siedzi on zazwyczaj w zerowej linii i jest przykryty atrybutami — trzeba to sprawdzić!) zamienić na 201 — powrót do Basic-a i jeśli wczytany plik nie przykrył zmiennych systemowych, lub tekstu programu w Basic-u, ani też loader nie zmienił nic istotnego, po wczytaniu nastąpi powrót do Basic-a. Wtedy należy wpisać POKE-a i wystartować program od adresu zakodowanego tuż na naszym wstawionym bajtem 201.

Opisane postępowanie odnosi się do standardowych loaderów asemblerowych. W przypadku większego ich skomplikowania potrzebna jest duża wiedza i doświadczenie. A zdobyć je można zaczynając od rzeczy małych, jak choćby opisana tu procedura LOAD.

**(...) Podczas przeprowadzki zginał mi zasilacz od Spectrum 48. Uniemożliwiło to zupełnie korzystanie z komputera, niedawno zresztą kupionego. W jaki sposób mogę zdobyć potrzebny mi zasilacz?**

Ślawomir Żmudziński  
Bonin 36/4  
76-009 Bonin

W żadnym polskim sklepie nie spotkaliśmy jeszcze zasilacza do Spectrum. Również na giełdach nie było go widać. Prawdopodobnie i sklepy zachodnie rzadko kiedy posiadają zasilacze. Można jednak czasem spotkać w sklepach elektrotechnicznych 9-woltowe zasilacze 1.4 A, 12.6 W. One

nadają się do Spectrum i pracują bez zarzutu. Jeśli ktoś z Czytelników może umożliwić Ślawkowi kupno potrzebnego zasilacza, gorąco o to prosimy.

**Mam kilka pytań związanych z Timexem:**

1. Czy AY to interfejs dołączany do szyny krawędziowej komputera, czy trzeba zamontować go wewnątrz?
2. Czy program „Music Box 128” będzie pracował na Timexie z podłączonym AY-grekiem?
3. Co zrobić, by po wypełnieniu ekranu nie pojawiała się pytanie scroll?

Rafał Staropiętka, Otwock

1. AY-greki spotkać można w postaci gotowych układów zapakowanych w puszkę z szyną krawędziową; wtedy dołączany jest on do szyny krawędziowej. Tak jest też z AY-grekiem, którego projekt prezentujemy w tym numerze. Można natomiast pozbyć się złącza krawędziowego i płytkę zamontować wewnątrz komputera (np. Spectrum +). Trzeba jednak mieć pojęcie o elektronice, by niczego nie popsuć.
2. Timex z interfejsem AY różni się od Spectrum z tym interfejsem tak samo, jak i bez niego. Tak więc nie będzie żadnych przeszkód w pracy „Music Boxa”. Szerzej o różnicach Timex-Spectrum za miesiąc, w Klanie Spectrum.
3. Zmienna systemowa o adresie 23692 przechowuje liczbę linii do przesunięcia bez pytania scroll?

Dla przykładu:

10 POKE 23692,10  
20 LET a=0  
30 PRINT a  
40 LET a=a+1  
50 GO TO 30

Ostatnim numerem będzie 30. Natomiast umieszczenie instrukcji POKE 23692,255 w pętli będzie powodowało „scroll” bez końca, tzn. całkowite pominięcie pytania.

# Atari Turbo 2000 F

Nowy system transmisji danych z magnetofonem przyspieszony do 6700 bodów.

Komplet:

- cartridge
- oprogramowanie
- przeróbka magnetofonu
- instrukcja obsługi
- 12 miesięcy gwarancji

Instalację wykonujemy na oczekaniu.

Interfejs do zwykłego magnetofonu

Duży wybór oprogramowania w standardzie TURBO-2000.

**Informacja:**

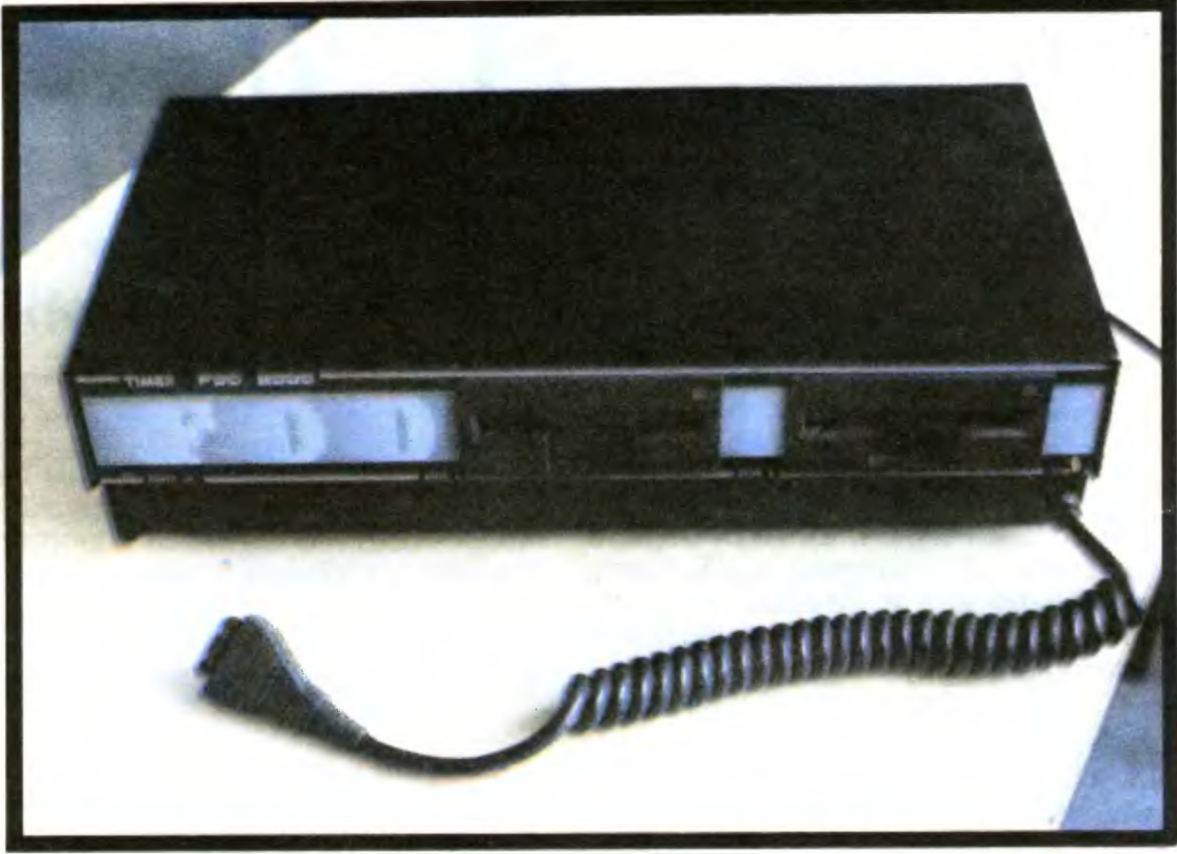
Tel. 33-40-91

**Korespondencja:**

MUEL ul. Cząstkowska 30,  
01-678 Warszawa.

B-31





## STACJA DYSKÓW FDD 3000

„...najlepszą przeróbką magnetofonu jest zamiana go na stację dysków”.

W. Zientara

**Magnetofon, pomyślany jako podstawowa pamięć zewnętrzna Spectrum, u progu XXI wieku jest po prostu niewystarczający.**

Tzw. mikrodrajwy byłyby doskonałą alternatywą, gdyby nie dwie przeszkody: mała trwałość i wyjątkowość. Zupełnie nieznanne u nas waffadrajwy są również ciekawostką i użycie ich równoznaczne jest z postawieniem fabryki kasetek. Co więc zrobić może użytkownik Spectrum, posiadający całą szafę kaset magnetofonowych i myślący szybciej, niż obraca się magnetofon? Powinien zaopatrzyć się w stację dysków.

Już dawno temu, za olbrzymie pieniądze, dostępne były 5.25" stacje specjalne do Spektrum. W większości były to standardowe 40-ścieżkowe napędy w obudowie, z kontrolerem przerobionym dla potrzeb Spectrum. Niestety, w takich przypadkach co stacja, to inny format dyskietki, system zapisu itp.

Najdalej posunęła się firma Timex, wystrzelując zaraz za Timexem 2048 stację Timex FDD 3000. Oba te produkty rozprowadzane są przez Centralną Składnicę Harcerską i znajdują w kraju rzeszę wdzięcznych odbiorców. Znamy już Timexa 2048, jego możliwości i zalety, przyjrzyjmy się więc stacji.

Stacja FDD 3000 ma postać prawie prostopadłościennego pudełka o wymiarach 9x19x38 cm. Od spodu znajdują się cztery gumowe nóżki, z tyłu wyjście kabla zasilania, wyłącznik + bezpiecznik, dwa gniazda RS 232C i wyjście koncentryczne do monitora. Z przodu widnieje przód kieszeni napędu, przycisk podpisany RESET oraz kabel spiralny zakończony 15-bolcowym gniazdem „D”.

W środku znajduje się zasilacz, kontroler i jeden lub dwa napędy „3”. Kontroler jest osobnym systemem

stemem operacyjnym TOS i kilkoma kośćmi trudnymi do zidentyfikowania.

Po podłączeniu stacji do Spectrum, a raczej Spectrum do stacji otrzymujemy naraz dwa twory: pierwszy to Spectrum z rewelacyjną dyskową pamięcią zewnętrzną (160 KB na stronie dyskietki 3"); drugi to komputer z CP/M-em 2.2, gdzie Spectrum „robi” za klawiaturę. Co oznacza CP/M, wiemy wszyscy. WordStar, DBase, Turbo Pascal, Turbo C itd. 64 znaki w wierszu, troszkę nieczytelne w Spectrum, stają się niesłychanie wyraziste przy mariażu Timex 2048-FDD 3000.

Niestety, nasz CP/M nie czyta dyskietek zapisanych np. na Amstradzie pod tym samym systemem operacyjnym. Wystarczy jednak mała programowa przeróbka i prawie wszystko działa bez zarzutu, z wyjątkiem skomplikowanych pakietów „grzebiących” gdzieś w zaułkach systemu.

Pewne kłopoty sprawia na początku postępowanie się klawiaturą. Karkołomne kombinacje klawiszy mogą przyprowadzić o zawrót głowy. Np. CONTROL — CS + SS.ESC — CS + SPACE itd. Nie było jednak wyjścia, klawiatura Spectrum bogata nie jest.

Stacja FDD 3000 dostarczana jest w dwóch wersjach: z jednym lub dwoma napędami 3". Możliwa jest współpraca aż 4 napędów, z których dwa są zewnętrzne. Dla przykładu, podłączenie 40-ścieżkowego napędu 5.25" z IBM-a wymaga przelutowania dwóch kabli w napędzie, 80-ścieżkowy napęd potrzebuje już większej przeróbki, ale otrzymuje się dzięki niemu 620 KB na stronie dyskietki.

Trzycalowe, standardowe dyskietki są drogie i większa biblioteka programów wymaga dużego wkładu finansowego. Dlatego podłączenie zewnętrznego napędu 5.25" daje dużą wyprawkową oszczędność.

CSH rozprowadza dyskietki po zawrotnej cenie 25.000 zł za sztukę. Z prostego dodawania wynika, że za trzy sztuki należy zapłacić 75.000 zł. Tymczasem na półce obok leży pakiet programów pod CP/M-em wraz ze sterłą instrukcji. Programy nagrane są właśnie na trzech dyskietkach Maxell, takich samych, jak te po 25 tys. Pakiet zaś kosztuje jedynie 70.000 zł...

Oprócz kilku programików w Basic-u, do stacji nie ma żadnych programów użytkowych. Tę lukę wypełniają stopniowo rodzimi użytkownicy Spectrum, którzy „oczyścili” CSH ze stacji. Nasz redakcyjny kolega, Maciej Pietraś („Brómba”) przejrzał już FDD 3000 na wylot i pisze masowo kolejne kopie, weryfikatory, „toosy”, di-sectory itp. Tak więc programów nie zabraknie.

Również istniejące użytki, np. GENS, Hisoft Pascal można w prosty sposób przerobić tak, by współpracowały ze stacją. W „Bajtku” podaliśmy

już przykłady takich przeróbek, temat stacji przewijał się będzie jeszcze nie raz.

Komu to nie wystarczy, może napęd 3" podłączyć do IBM-a lub transmitować z nim dane za pośrednictwem złącza RS 232. Możliwości jest wiele.

W redakcji „męczyliśmy” FDD 3000 przez ponad dwa miesiące. W pierwszym tygodniu jeden ze zręczniejszych redaktorów zrzucił ją ze stołu, co spowodowało urwanie transformatora, przyspawanego w środku za śrubę do ścianki. Na szczęście Punkt Napraw CSH przy ul. Okrąg prawie momentalnie (3 tyg.) naprawił szkodę. Od tego czasu bez mechanicznych niespodzianek znęcaliśmy się ujawniając najdziwsze instynkty. Odkryliśmy, że kilkunastokrotne włączenie i wyłączenie jednostki przelutowniczym z tyłu, bez wyjmowania dyskietki powoduje pojawienie się na niej bad-sektorów, nie do odzyskania. Red. Pietraś napisał program, który wprawia głowicę napędu w taki taniec, że urywa się ona. Przycisk RESET użyty odpowiednio szybko pozabawił twarz red. Pietraśa perwersyjnego uśmiechu i uratował głowicę.

Poza tym stacja sprawowała się wspaniale. Szybka i praktycznie niezawodna zaspokajała momentalnie pragnienie kolejnej gry, następnego kompilatora, programu kopiującego.

Na koniec należy zestawić wszystkie wnioski w dwóch słupkach: za i przeciw. Oto one:

#### Za:

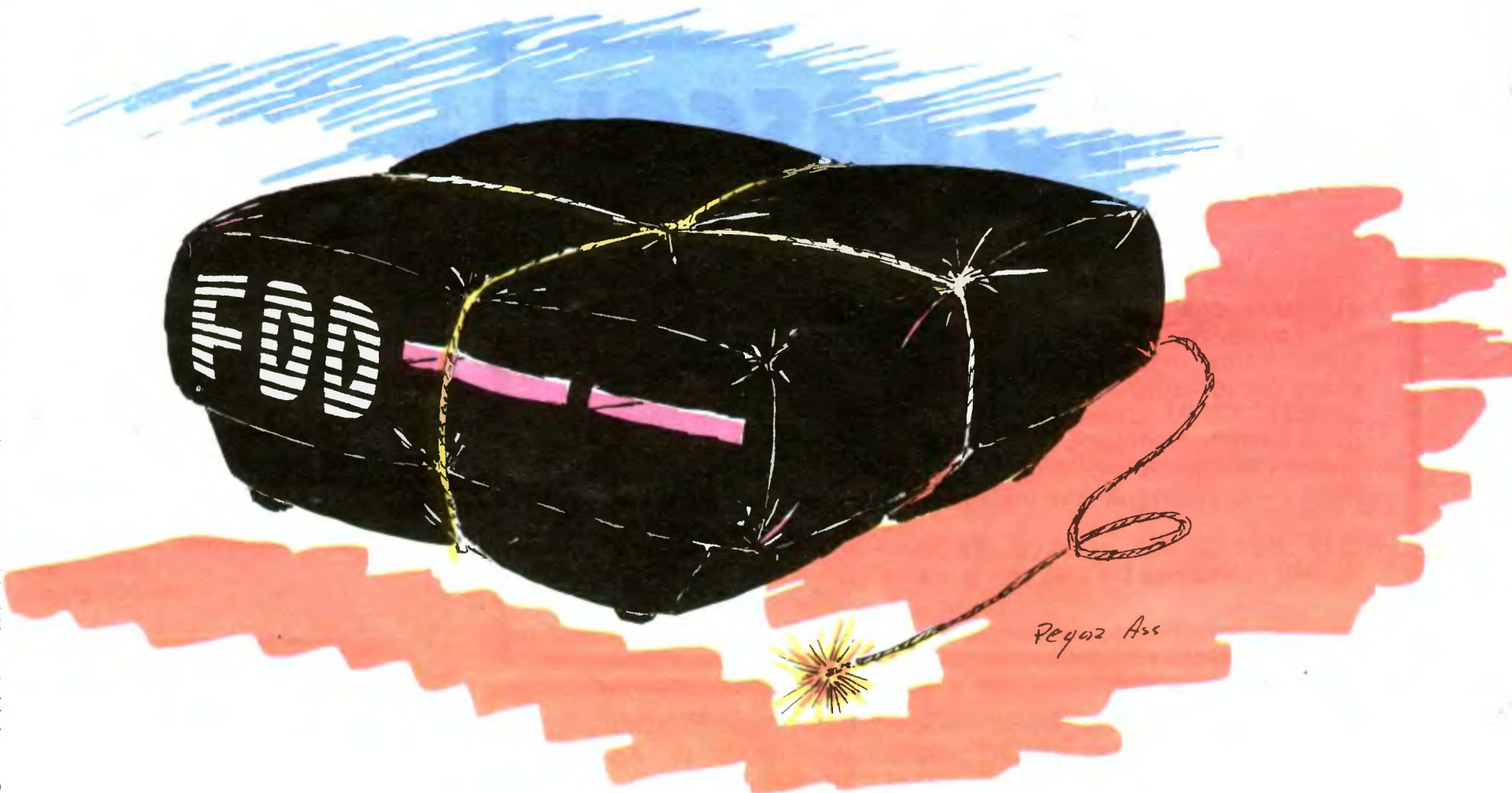
- duża szybkość i niezawodność,
- niska cena (450.000 zł do końca 1989 roku),
- praca pod CP/M-em,
- możliwość podłączenia zewnętrznych napędów,
- wyjście do monitora,
- złącze RS 232C,
- dobrze opracowana, trzyczęściowa instrukcja,
- doskonały system operacyjny TOS (patrz „Bajtek” 1/88”),
- możliwość ustawienia monitora na stacji.

#### PRZECIW:

- drogie dyskietki,
- niezgodność formatu z formatem Spectrum +3,
- CP/M nieznacznie odbiegający od standardu,
- błędy w odczycie dyskietki, gdy na stacji stoi polski monitor Neptun.

Marcin Przasnyski

**Testowaną stację dysków otrzymaliśmy dzięki uprzejmości p. Wojciecha Szantera, dyrektora Centralnej Składnicy Harcerskiej.**





## WUJA TEODORA

WPPUH

„ATARES”

spółka z o.o.

Chorzów ul. Truchana 35,  
tel. 500-797

w godzinach od 18—21

oferuje użytkownikom ATARI najnowsze rozwiązania sprzętowe podnoszące komfort pracy z komputerem m.in.

— **BLIZZARD TURBO** — kasetowy system transmisji danych 6000 bodów (magnetofon po przeróbce czyta 10 razy szybciej, praktycznie bez błędów)— **CRYSTAL SOUND** — system digitalizacji dźwięku— **CARTRIDGE** do obsługi systemu BLIZZARD TURBO (6 typów) i inne— **INTERFACE CBT** do magnetofonów niefirmowych (realizuje transmisję standardową i w TURBO BLIZZARD 6000B)— **INTERFACE STANDARD** do magnetofonów niefirmowych (odczyt — zapis 600 Bodów)— **INTERFACE CENTRONICS** do współpracy z drukarkami i plotterami— **PROGRAMATOR EPROM PE-1**— **TURBO DRIVE 1050 (LDW 2000)** format dysku 180 kB, transmisja 70 kB— **MAXI TURBO DRIVE 1050** — format dysku 180 kB, transmisja 140 kB— **ROZSZERZENIA PAMIĘCI ATARI** do 256 kB łącznie— **SERWIS NAPRAW POGWARANCYJNYCH** sprzętu ATARI, COMMODORE, SPECTRUM— **OPROGRAMOWANIE do ATARI, COMMODORE (w tym AMIGA), SPECTRUM**

## W PRZYGOTOWANIU

— **TELETEKST** na bazie ATARI oraz COMMODORE C-64— **HOME PHOTO** — przetwornik obrazu ATARI— **SUPER SONG** — generator dźwięku do C-16, C+4— **VIDEO LETTER** — system nakładania napisów na obraz video— **UDZIELAMY ROCZNEJ GWARANCJI**, w przypadku magnetofonów ATARI po montażu TURBO przejmujemy GWARANCJĘ po PEWEX-ie— **ZAPRASZAMY PT KLIENTÓW DO WIELOBRANŻOWEGO SKLEPU PRZEMYSŁOWEGO** przy ul. Truchana 35 codziennie w godz. 9—17 Prowadzimy również sprzedaż pozarynkową.

(SB91)

Wojewódzkie Przedsiębiorstwo  
Handlu Wewnętrznego  
Oddział w Tychach

## VIDEOBIT

43-100 Tychy, Al. ZMP 77  
tel. 276975

## poleca między innymi

- sprzęt komputerowy  
Atari ● Commodore ● Amstrad ●  
● IBM PC XT/AT/PS 2
- drukarki STAR, EPSON, AMSTRAD
- Sprzęt audiowizualny
- magnetowidy
- OTV PAL/SECAM
- Videoskopy
- kamery
- anteny satelitarne
- aparaturę badawczo-naukową

— **Udzielamy gwarancji, prowadzimy naprawy pogwarancyjne. Zapewniamy o atrakcyjnych cenach.**

B7

## Cześć Maluchy!

*Wuj Teodor jest konduktorem i jeździ stale na trasie północ—południe. Wszystkie stacje na tej trasie ponumerowane są od 1 (południe) do 50 (północ). Nigdy nie wiadomo, gdzie znajduje się wuj. Na stacjach uzyskujesz jedynie informację, czy pojechał on w kierunku północnym, czy też południowym. W ciągu jednego dnia możesz znaleźć się na dowolnej stacji. Możesz jednak odbyć tylko jedną podróż dziennie. Wuj przebywa w tym czasie najwyżej dwa przystanki. Ciekawe, ile czasu zajmą ci poszukiwania wuja.*

Nasza dzisiejsza zabawa jest odmianą bardzo popularnej gry polegającej na zgadywaniu liczby wylosowanej przez komputer. W tym przypadku pojawia się jednakże dodatkowa komplikacja — poszukiwana liczba zmienia swą wartość w miarę upływu czasu. Spróbujmy prześledzić działanie tego programu.

W pierwszej kolejności komputer losuje miejsce startu (linia 120), czyli numer stacji, z której wuj Teodor rozpoczął swą podróż. Aktualna pozycja wuja przechowywana jest w zmiennej **wuj**. Następną czynnością komputera jest pytanie o stację, do której wybiera się grający (linia 140) i wczytanie odpowiedzi (linia 140) Aktualna pozycja grającego przechowywana jest w zmiennej **stacja**. Teraz komputer przystępuje do wczytania (linia 150) nazwy kolejnego dnia tygodnia znajdujących się w liniach danych **DATA** (linie 500—560). Nazwy dni przechowywane są w zmiennej tekstowej **dzień**. Jeśli przyjmie ona wartość „niedziela” (linia 160) komputer wykonuje komendę **RESTORE**, czyli zaczyna wczytywanie od początku danych zawartych w liniach **DATA** a więc od poniedziałku.

Dzień tygodnia drukowany jest na ekranie (linia 170) a następnie porównywana jest pozycja wuja (zmienna **wuj**) i szukającego (zmienna **stacja**) (linie 180—200) i w efekcie tego porównania wykonywany jest jeden z trzech podprogramów. W przypadku równości obu zmiennych wykonywany jest podprogram zakończenia gry (linie 1000 i 1100) — ukazuje się komunikat „Znalazłeś wuja Teodora!” i program zostaje zatrzymany. W przypadku gdy wartość zmiennej **stacja** jest większa od wartości zmiennej **wuj** wykonywany jest podprogram informujący, że „Wuj Teodor pojechał na południe” (linie 2000 i 2100), a w przeciwnym przypadku, że „Wuj Teodor pojechał na północ” (linie 3000 i 3100).

Teraz komputer przystępuje do przemieszczenia wuja Teodora. W tym celu losuje wartość zmiennej **zmiana** (linia 210). Wartość ta może wynosić -2, -1, 0, 1 lub 2 i zostaje dodana do poprzedniej pozycji

wuja określonej wartością zmiennej **wuj** (linia 220). Jeśli zdarzy się, że w wyniku tej operacji zmienna **wuj** osiągnie wartość wyższą niż 50 lub niższą niż 0, wówczas otrzyma ona wartość, odpowiednio — 50 i 0 (linie 230 i 240). Zamknięcie pętli i powrót do pytania o decyzję grającego (linia 130) następuje w linii 250.

Analizując program warto zwrócić uwagę na konstrukcję rozkazów losujących, a w szczególności losowanie wartości **zmiana** w linii 210. Ci z was, którzy dotychczas nie mogli zrozumieć jak działa instrukcja czytania danych **READ**, mogą dokładnie prześledzić ten mechanizm na przykładzie tego programu.

Strategia gry nie jest bynajmniej prosta. Sporo zależy tu od szczęścia, szczególnie w ostatniej fazie gdy szukający znajduje się już bardzo blisko wuja, ale jeszcze więcej zależy od metody prowadzenia poszukiwań. Przydaje się też bardzo znajomość rachunku prawdopodobieństwa.

W tę grę można bawić się samemu, można też zorganizować zawody. W tym drugim przypadku, jeśli chcemy wyłonić naprawdę najlepszego stratega, musimy dokonać większej liczby poszukiwań i obliczać średnią. Dzięki temu zminimalizujemy wpływ przypadku, czy pecha, którego nie życzy wam w żadnym przypadku

Romek

```

100 REM *** poszukiwania wuja ***
110 REM ***** Teodora *****
120 LET wuj=INT(RND(1)*50)+1
130 PRINT "Dokad jedziesz?"
140 INPUT stacja
150 READ dzien$
160 IF dzien$="niedziela" THEN RESTORE
170 PRINT dzien$
180 IF stacja=wuj THEN GOTO 1000
190 IF stacja>wuj THEN GOSUB 2000
200 IF stacja<wuj THEN GOSUB 3000
210 LET zmiana=INT(RND(1)*5)-2
220 LET wuj=wuj+zmiana
230 IF wuj>50 THEN LET wuj=50
240 IF wuj<1 THEN wuj=1
250 GOTO 130
500 DATA "poniedzialek"
510 DATA "wtorek"
520 DATA "sroda"
530 DATA "czwartek"
540 DATA "piatek"
550 DATA "sobota"
560 DATA "niedziela"
1000 PRINT "Znalazles wuja Teodora!"
1100 END
2000 PRINT "Wuj Teodor pojechał na południe."
2010 RETURN
3000 PRINT "Wuj Teodor pojechał na północ."
3010 RETURN

```





# CHIPY W PRZESTWORZACH

Początki elektroniki w lotnictwie to — formalnie — lata dziesiąte naszego wieku. Pod koniec I wojny światowej po raz pierwszy zastosowano radiostacje w ciężkich samolotach bombowych — stutonowych maszynach o zasięgu kilkudziesięciu kilometrów. Dalej poszło już samo.

Druga wojna światowa przyniosła tyle nowinek w dziedzinie elektroniki, że nie sposób ich tu nawet wymienić. Radiostacje montowano masowo, stosowano dziesiątki systemów naprowadzania, pierwsze kierowane pociski, bomby i rakiety, wewnętrzne telefony, radary naziemne i pokładowe, zdalnie sterowane wieże strzelnicze, noktowizory. Wymieniam tu urządzenia typowo wojskowe, ze względu na „naturalną” awangardowość techniki militarnej.

Mimo nowoczesności, niezawodności i wszechstronności elektroniki, co osiągnięto już w latach czterdziestych, do skoku jakościowego było jeszcze daleko. Wciąż najistotniejszymi cechami samolotu były osiągi, moc silnika, zwrotność itp. Rola systemów elektronicznych z roku na rok rosła, choć na razie tylko ilościowo.

Do wojny wietnamskiej w rozwoju lotnictwa wojskowego stawiano głównie na pułap i szybkość samolotu, nie zapominając oczywiście o tzw. bajerach. Dopiero w latach 70-tych postarano się o stworzenie kompleksowych systemów informatycznych, których cechą jest uniwersalność, tzn. ogarnianie ca-

łej maszyny oraz wyjście swym zasięgiem poza nią. Współpraca ze stacjami naziemnymi doszła z czasem do pewnej granicy — np. w radzieckim Su-15 naprowadzanie na cel odbywa się zupełnie bez udziału pilota; potrzebny jest on tylko do stoczenia walki.

Właściwie każdy z systemów używanych w supernowoczesnych maszynach bojowych — F-16, F-18, B-1, B-2, Mig-29, Su-27, EFA, Rafale — znany był już wcześniej. Istota nowej generacji polega jednak miniaturyzacji oraz całkowitym zintegrowaniu wszystkich urządzeń. Tworzy to układ, którego najsłabszym ogniwem jest człowiek.

Wyrafinowanie elektronicznych urządzeń w samolotach, nie tylko wojskowych, doszło w latach 80-tych do takiego stopnia, że osiągi w wielu przypadkach przestały się liczyć. Najszybszy myśliwiec świata, Mig-25, powstał w połowie lat 60-tych. Po nim żaden samolot seryjny nie przekroczył prędkości 2 Ma (2 razy prędkość dźwięku, czyli ok. 2500 km/h). Dopiero teraz przewidywane jest przekroczenie tej liczby, przez maszyny... pasażerskie i transportowe.

Mig-29 oraz niektóre samoloty cywilne, np. pasażerski Boeing B-7J7.

Zmianom musiało ulec też wyposażenie kabiny pilota. W pracy nowoczesnej maszyny występują już nie dziesiątki, ale setki parametrów, których rejestrowanie i branie pod uwagę przekracza możliwości ludzkiego umysłu. Gdyby stosować zwykłe, mechaniczne „zegary”, zachodziłaby też obawa, że nie zmieszczą się one na desce rozdzielczej. Musiano więc uciec się do ekranów. Na kilku ekranach (HDD — Head-Down Display) pilot obserwuje tylko te dane, które w konkretnym momencie są mu niezbędne. Dla przykładu, w aerobusie A310 są one wybierane nawet ze 160 parametrów — od aktualnej prędkości, przez mapę terenu do zawartości tlenu w powietrzu w WC nr 3. Dane te podawane są zwykle w postaci analogowej, łatwiejszej do błyskawicznego przyswojenia.

Innym systemem wyświetlania danych jest HUD (Head-Up Display), bardzo efektowny, polegający na projekcji obrazów przez przednią szybę. Może być to celownik, zarysy terenu lub wszelkiego rodzaju wskaźniki. HUD stosowany jest w samolotach i śmigłowcach wojskowych już od wielu lat. Wyposażony jest weń np. nienowowy już Mig-23.

Odmianą HUD-a jest rzutowanie celownika na szybę hełmu pilota, co czyniło niesłychane wrażenie na widzach filmu „Błękitny Grom”, a znalazło zastosowanie w Mig-u-29.

Bez „elektronizacji” nie obył się i proces samego kierowania. Zastosowane sterowanie aktywne, drążki sterowe, choć nie zamienione na klawisze, nie są już połączone ze sterami za pomocą linek stalowych, ale szyn cyfrowych. Po mikroinformatycznym można by je nazwać joystickami. Oprócz tego istnieją już systemy sterowania głosem, a koncepcja odczytywania myśli przez samolot przybiera realne kształty.

Niektóre maszyny wyładowane są elektroniką w sposób niewiarygodny. Niemal zabytkowy Il-62 ma na pokładzie 6 komputerów, myśliwski F/A-18 ma ich 13. Rekordzistą jest szwedzki JAS-39 Gripen, który w samym systemie nawigacji zawiera ponad 30 jednostek, a masa urządzeń elektronicznych przekracza 500 kg. Przy startowej masie 8 ton (z paliwem i uzbrojeniem) stanowi to spory procent. Co w zamian? Na przykład to, że Gripen podaje na trójwymiarowej mapie terenu nawet nazwy miejscowości.

Na koniec wypada wspomnieć o projektowaniu statków powietrznych. Tu komputery są jeszcze bardziej niezbędne, niż w samej konstrukcji. Bez maszyny cyfrowej nie do pomyślenia jest zaprojektowanie nawet lotni, nie mówiąc już o bombowcu czy samolocie kosmicznym. Nie wchodzi tu w grę zwykłe obliczenia, lecz różne rodzaje symulacji i modelowania, co obniża koszty i przyspiesza czas budowy prototypu. Skala użycia komputerów jest olbrzymia. Przy projektowaniu B-2 w kalifornijskim centrum Jacka Northropa pracowało naraz 400 terminali w 30 salach. Obniżyło to znacznie cenę samolotu; wynosi ona teraz tylko 500 mln dolarów.

Ładowanie układów scalonych w latające zabawki idzie pełną parą. Koszty są olbrzymie, oszczędności niewyobrażalne. Jak zwykle, lwia część nowoczesnych rozwiązań znajduje zastosowanie w maszynach wojskowych, z czasem przenikają one i do cywila. Ludzkość obserwuje rozwój elektroniki z zapartym tchem. To dzięki niej możemy się już jutro nie obudzić.

PTAKU