

NUMER  
SPECJALNY

TYLKO 0  
**ATARI**

NR INDEKSU 391549  
PL ISSN 0860-1674

# Bojitek

**5** OPISÓW  
GIER

KLUBY ATARI W POLSCE

 **ATARI**<sup>®</sup>  
COMPUTERS

**DOS**  
2.5

**TURBO-COPY**

**LDW SUPER 2000**

**Action!**



# DROGI CZYTELNIKU

Atari to jeden z najpopularniejszych komputerów w naszym kraju. Można się o tym przekonać na każdej giełdzie, w każdym sklepie ze sprzętem komputerowym, przysłuchując się rozmowom w szkole i w pracy. Listy do klanu Atari stanowią zawsze najpokazniejszą paczkę w codziennej poczcie przychodzącej do redakcji „Bajtki”. Cały czas otrzymujemy prośby o zwiększenie liczby stron, na których drukujemy teksty o Atari. Spełniamy tę prośbę wydając specjalny numer poświęcony wszystkiemu, co związane jest z hasłem Atari. Mamy nadzieję, że trafiliśmy w Wasze oczekiwania. Chętnie przeczytamy listy z Waszymi uwagami o tym numerze specjalnym. Pomogą nam również redagować kolejne wydania specjalne. Będą one oczywiście poświęcone Atari, ale również innym komputerom (Commodore, Amstrad, IBM), a także grom.

Numerzy dodatkowe nie są objęte prenumeratą. Pytajcie o nie w kioskach „ruchu” i klubach międzynarodowej prasy i książki. Oczywiście we wszystkich „normalnych”, objętych prenumeratą, numerach „Bajtki” będzie po starciu klan Atari — atrakcyjny jak dotychczas.

Przyjemnej i pożytecznej lektury!

Zespół „Bajtki”

## „BAJTEK-ATARI”

(WYDANIE SPECJALNE „BAJTKA” — POZA PRENUMERATĄ!)

PRZYGOTOWAŁ ZESPÓŁ W SKŁADZIE:

Waldemar Siwiński, Roman Poznański, Wojciech Zientara, Wanda Roszkowska (opr. graficzne), Sławomir Gajda (red. techniczny), Leopold Dzikowski (zdjęcia).

Fotoskład — Tadeusz Olczak

Montaż offset. — Grażyna Ostaszewska

Korekta — Maria Krajewska,

Zofia Wóltańska

ADRES: 00-687 Warszawa, ul. Wspólna 61. Tel. 21-12-05.

WYDAWCA: RSW „Prasa-Książka-Ruch”

Młodzieżowa Agencja Wydawnicza,

al. Stanów Zjednoczonych 53, 04-028

Warszawa. Telefony: Centrala 13-20-40

do 49, Redakcja Reklamy wewn. 403,

414.

Skład techniką CRT-200,

przygotowalnia offsetowa i druk:

PRASOWE ZAKŁADY GRAFICZNE

RSW „PRASA-KSIĄŻKA-RUCH”

w Ciechanowie, ul. Sienkiewicza 51.

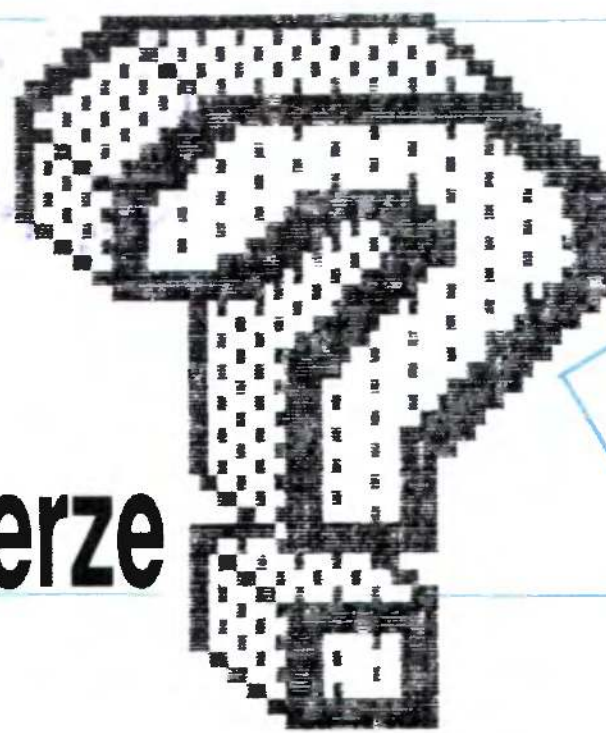
Nr zlecenia 150077

nakład 100 000 + 300 egz., U-113

Cena 180 zł,—



## CO W numerze



## PROGRAMOWANIE

### Action! — nowy język Atari . . . . . 15

Prostszy niż Pascal i C, wielokrotnie szybszy od BASIC-a — strukturalny język programowania. Kto raz spróbuje programowania w Action! ten nigdy już nie wróci do BASIC-a.

### Zabezpieczanie programów w BASIC-u . 19

Piractwo komputerowe jest plagą końca dwudziestego wieku. Każdy programista powinien wiedzieć, jak utrudniać życie hackerom.

## GRY

### Szczebel po szczelbu . . . . . 24

W co grać i jak grać, czyli krótki przegląd komputerowych gier zręcznościowych.

### Nowe i stare . . . . . 24

Opisy gier: „Warship”, „Who Dares Wins II”, „Arkanoid”, „Alley Cat”, i „Chessmaster 2000”.

## PROGRAMY UŻYTKOWE

### Print Shop Converter . . . . . 8

Program umożliwiający przekształcenie rysunków „Print Shop” do formatu DOS 2.5. Pozwoli Ci to na skorygowanie wykonanego rysunku i wydruk na Atari 1029.

### Jak używać Atari 1029? . . . . . 10

Drukarka Atari 1029 wzorowana na Seikoshy GP50 ma możliwości bardzo niewielkie. Odpowiednie oprogramowanie nie zamieni jej co prawda w Epsona, ale trochę do niego zbliży.

### Turbo-Copy . . . . . 14

Jeśli denerwuje Cię długi czas wczytywania programów z magnetofonu, to ten program skróci oczekiwanie oraz umożliwi kopiowanie.

### Pół żartem, pół serio . . . . . 28

Nareszcie będziesz mógł zrezygnować z pomocy kolegów posiadających Commodore'a lub Amstrada. Po obliczeniu biorytmu pomożemy Ci jeszcze w interpretacji.

## SPRZĘT

### Rodzina Atari . . . . . 4

Krótki przegląd 8-bitowych komputerów firmy Atari Corp.

### LDW Super 2000 . . . . . 6

Wkrótce w sklepach Pewexu pojawi się nowa stacja dysków do Atari XL/XE. Już teraz możesz zapoznać się z redakcyjnym testem tego urządzenia.

### Najprostszy modem . . . . . 10

Połączyć komputery między sobą? Nawet początkujący elektronik może wykonać prosty modem telefoniczny i wymieniać z kolegami programy nie wychodząc z domu.

## SZKÓŁKA

### Ładowanie i zapisywanie programów . . 5

Coś dla laików. Pierwsze kroki po nabyciu komputera: jak uruchomić program z dysku lub kasety. Jak zapisać własny program?

## INSTRUKCJE

### DOS 2.5 . . . . . 12

Najpopularniejszy Dyskowy System Operacyjny Atari nie dla wszystkich jest bez tajemnic. Nie posiadasz instrukcji, masz wątpliwości — przeczytaj.

## DZIAŁ ST

### Języki programowania ST . . . . . 29

Ogromne bogactwo języków programowania dostępnych na ST rodzi problem: co wybrać? Pomoże Ci w tym krótki przegląd istniejących kompilatorów i interpreterów.

### MIDI . . . . . 31

Komputery serii ST są wyposażone w złącze MIDI. Służy ono do przesyłania cyfrowych sygnałów dźwiękowych. Szczegóły w numerze.

### Koprocessor arytmetyczny . . . . . 31

Również na tym polu Atari nie ustępuje IBM. Możliwości i zastosowanie koprocessora arytmetycznego w komputerach ST.

### Porozmawiać z ST? . . . . . 32

Po co naciskać klawisze, skoro można polecenia wydawać głosem. Specjalna przystawka umożliwi komputerowi zrozumienie Twoich słów.

## INNE

### CLAUG . . . . . 20

Klub użytkowników Atari w Chicago

### Ocena . . . . . 13

Oceniamy programy Czytelników „Bajtki” — każdy może nadesłać swój program do redakcji i przeczytać opinię o nim.

### Mam Atari . . . . . 3

### List laika . . . . . 22

### Recenzje . . . . . 21

### Edytor BASIC-a . . . . . 3

### Listy do redakcji . . . . . 22

### Ceny sprzętu . . . . . 21

### Kluby Atari . . . . . 20

### Wieści . . . . . 7, 19, 28



# Mam ATARI

**Wiele osób pyta mnie, dlaczego kupiłem Atari. Odpowiedź jest prosta, ale niestety nie krótka. Pracowałem i grałem na różnych komputerach domowych poczynając od ZX-81. Mogę powiedzieć, że znam wszystkie w stopniu umożliwiającym bezproblemową obsługę. Więc dlaczego właśnie Atari?**

Powiedzmy sobie najpierw jakie podstawowe wymagania musi spełniać komputer, który będzie używany przede wszystkim do pracy, a w wolnych chwilach jako narzędzie rozrywki. Powinien on mieć nie mniej niż 48 KB RAM z możliwością późniejszego rozszerzenia co najmniej do 128 KB. Następne wymagania to: praca ze stacją dysków, profesjonalna klawiatura, spore możliwości graficzne i oczywiście dostępność oprogramowania oraz przystępna cena. Nie mniej ważną cechą jest możliwość stopniowej rozbudowy systemu w miarę wzrostu zasobów finansowych.

Wymienione warunki natychmiast wyeliminowały ZX-Spectrum i jego liczne odmiany. Ten komputer powinien być traktowany wyłącznie jako zabawka dla cierpliwych dzieci (niecierpliwie szybko zniszczą). Ograniczenia finansowe wyeliminowały

IBM i jego klony oraz Atari ST i Amigę. Do wyboru pozostało więc sześć komputerów z trzech firm: Amstrad CPC-664, Amstrad CPC-6128, Atari 800XL, Atari 130XE, Commodore C-64 i Commodore C-128.

Zacznijmy od Amstrada. Trzeba od razu szczerze powiedzieć, że jest on mocno przereklamowany. Wszystkie komputery tej firmy stanowią zamknięte (prawie) systemy, co utrudnia stopniową rozbudowę. Zakup musi obejmować z konieczności od razu cały system i siłą rzeczy jest dużym obciążeniem budżetu. Kolejne wady to: niestandardowy rozmiar dyskietki (3 cale), ograniczone możliwości graficzne i stosunkowo niewiele oprogramowania. Włączenie do systemu monitora jest pomysłem dobrym, jeśli zamierza się używać komputera wyłącznie do pracy. W grach oglądanie obrazów w różnych odcieniach zieleni jest niezbyt pasjonujące. Kolejną poważną (z mojego punktu widzenia) wadą jest zastosowanie procesora Z80. Mając pewne doświadczenie w assemblerze twierdzę, że programowanie 6502 wymaga przynajmniej o połowę mniejszego nakładu pracy, a wynik jest znacznie bardziej optymalny. Na koniec zostawiłem najpoważniejsze zastrzeżenie. Jeśli firma wypuszcza co pół roku model komputera i wiele programów nie chce działać na nowym modelu (i odwrotnie), to dowodzi to nieodpowiedniego stosunku do użytkownika. Alan Sugar (właściciel Amstrada) sam zdyskwalifikował się w moich oczach stwierdzeniem: „Nie interesują mnie ci, którzy kupili komputer, lecz tylko ci, którzy zamierzają go kupić”.

Dalej wybór nie był już taki łatwy. Commodore i Atari są urządzeniami tej samej klasy. Atari ma lepszą grafikę, Commodore dźwięk. Commodore ma szybsze procedury matematyczne, Atari bezpo-

średnio obsługę grafiki i dźwięku z Basic-a. Ilość dostępnego oprogramowania jest porównywalna. Trzeba było sięgnąć głębiej. Szybkość transmisji danych ze stacji dysków w Atari wynosi 19 200 bodów, a w Commodore 3800 bodów. Na dyskietce Atari może zapisać 130 KB, Commodore — 170 KB. Trudno się zdecydować. W tym przypadku decydującą rolę odegrały dwa czynniki. Cena C-64 jest około 1,5 raza większa niż 800XL i taka sama jest relacja między C-128 a 130XE. Drugim czynnikiem były moje subiektywne odczucia. Bardzo nie podoba mi się edytor Commodore, a szczególnie znaki kontrolne i praca w trybie wpisywania. Chciałbym tu zaznaczyć, że w chwili podejmowania decyzji dostępność na rynku była podobna — Pewex nie sprzedawał jeszcze Atari. Po sprowadzeniu Atari przez Pewex doszedł jeszcze jeden niebagatelny element, a mianowicie gwarancja i autoryzowany serwis.

Pora teraz przedstawić dalszy ciąg wydarzeń. Ceny w RFN okazały się być „nieco” wyższe niż w reklamach. W efekcie skończyło się na Atari 600XL z magnetofonem. Po upływie około pół roku zmieniłem komputer na 800XL, a po następnych siedmiu miesiącach nabyłem stację dysków. Aktualny „stan” mojego komputera to 256 KB RAM, wbudowany Basic XE i Action! oraz stacja 1050 i drukarka 1029. Plany na najbliższą przyszłość to: modem XM301, interface Centronics i RS232 (produkcja własna: hardware kolega, a software ja), ewentualnie XEP80 oraz trochę drobiazgów (tabliczka graficzna, pióro świetlne itd.). Poznałem już ten komputer dość dobrze i gdybym miał ponownie wybierać, bez wahania wybrałbym znowu Atari.

Marek Zachar

## EDYTOR BASIC-a

**Przy przepisywaniu programów w BASIC-u, szczególnie zawierających dużo liczb w instrukcjach DATA, nie sposób ustrzec się błędów. Aby uniknąć żmudnego wyszukiwania popełnionych omyłek wszystkie programy w BASIC-u będą zamieszczane wraz z kodami kontrolnymi. Poniższy program „Edytor BASIC-a” umożliwi obliczanie i kontrolę kodów wpisywanych linii.**

Najpierw należy dokładnie przepisać zamieszczony tu wydruk i zapisać go na kasecie lub dysku. Przystępując do wpisywania dowolnego programu z naszego pisma trzeba wczytać i uruchomić „Edytor BASIC-a”. Następnie należy wpisywać linie programu. Po wpisaniu linii i naciśnięciu RETURN pojawia się ona w dol-

nej części ekranu wraz z obliczonym kodem kontrolnym. Jeżeli wyświetlony kod jest taki sam jak wydrukowany przed numerem linii, można wpisywać następną linię. Jeśli kody są różne, to ponowne naciśnięcie RETURN powoduje wyświetlenie wpisanej linii w górnej części ekranu i umożliwia dokonanie poprawek. Wpisanie samego numeru linii powoduje wymazanie z pamięci komputera linii programu o tym numerze. Naciśnięcie RETURN wywołuje zawsze ostatnio wpisaną linię. Aby wywołać inną, wcześniej wpisaną linię należy napisać numer tej linii z gwiazdką przed nim (np. \*140) i nacisnąć RETURN.

Po wpisaniu całego programu trzeba przerwać pracę „Edytora” naciśnięciem klawisza BREAK lub RESET. Następnie w celu usunięcia „Edytora” zapisujemy program na taśmie instrukcją LIST "C:",0,31999 lub na dysku instrukcją LIST "D:nazwa",0,31999. Teraz trzeba wpisać rozkaz NEW i odczytać program instrukcją ENTER "C:" lub ENTER "D:nazwa". Po tej operacji w pamięci komputera znajduje się wpisany program bez „Edytora BASIC-a” i można go już ostatecznie zapisać na nośnik.

```
32000 REM EDYTOR BASIC-A
32010 REM wersja 1.0 dla "Bajtka"
32020 CLR :DIM LINIA$(120):CLOSE #2:CLOSE #3
32030 OPEN #2,4,0,"E:":OPEN #3,5,0,"E:"
32040 ? "5":POSITION 11,1: ? "EDYTOR B
ASIC-A"
32050 TRAP 32040:POSITION 2,3: ? "Wpisz
linie programu"
32060 POSITION 1,4: ? " ":INPUT #2;LINI
A$:IF LINIA$="" THEN POSITION 2,4:LIST
B:GOTO 32060
32070 IF LINIA$(1,1)="*" THEN B=VAL(LI
NIA$(2,LEN(LINIA$)):POSITION 2,4:LIST
B:GOTO 32060
32080 POSITION 2,10: ? "CONT"
32090 B=VAL(LINIA$):POSITION 1,3: ? " "
;
32100 POKE 842,13:STOP
32110 POKE 842,12
32120 ? "5":POSITION 11,1: ? "EDYTOR B
ASIC-A":POSITION 2,15:LIST B
32130 C=0:ODP=C
32140 POSITION 2,16:INPUT #3;LINIA$:IF
LINIA$="" THEN ? "LINIA ";B;" USUNIET
A":GOTO 32050
32150 FOR D=1 TO LEN(LINIA$):C=C+1:ODP
=ODP+(C*ASC(LINIA$(D,D))):NEXT D
32160 KOD=INT(ODP/676)
32170 KOD=ODP-KOD*676
32180 KODS=INT(KOD/26)
32190 KODM=KOD-(KODS*26)+65
32200 KODS=KODS+65
32210 POSITION 0,16: ? CHR$(KODS);CHR$(
KODM)
32220 POSITION 2,13: ? "Jeżeli kod sie
nie zgadza, naciśnij RETURN i popr
aw linie.":GOTO 32050
```



Nasza redakcja otrzymuje wiele listów, w których Czytelnicy proszą o dane poszczególnych modeli komputerów Atari oraz pytają o ich kompatybilność. Odpowiadając na te pytania zamieścimy serię artykułów, w których opisemy 8-bitowe komputery firmy Atari oraz najważniejsze urządzenia peryferyjne.



ATARI 400

Wszystkie 8-bitowe komputery Atari posiadają klawiaturę w układzie QWERTY o 57 klawiszach oraz kilka klawiszy funkcyjnych, 16 barw w 16 odcieniach (256 kolorów — jednocześnie można uzyskać na ekranie 128 kolorów) oraz czterokanałowy generator dźwięku. Maksymalna rozdzielczość obrazu w trybach tekstowych wynosi 24 wiersze po 40 znaków, a w trybach graficznych 320x192 punkty. Standardowe złącza to wyjście TV, wyjście monitorowe, złącze szeregowe i gniazdo cartridge'ów. Większość programów działa prawidłowo na wszystkich modelach, a tylko nieliczne programy przeznaczone dla modeli 400/800 działają na pozostałych przy użyciu programów „Translator” lub „Fix XL”.

### ATARI 400

Mikroprocesor 6502 z zegarem 1,8 MHz. Pamięć RAM o wielkości 16 KB, ROM z systemem operacyjnym — 10 KB. Brak wbudowanego na stałe interpretera języka wyższego poziomu — razem z komputerem sprzedawany był interpreter Atari BASIC na cartridge'u. Klawiatura foliowa z czterema klawiszami funkcyjnymi (START, SELECT, OPTION i RESET). Trzy tryby tekstowe i dziewięć graficznych. Wbudowany głośnik. Cztery porty joysticków i drugie dodatkowe gniazdo cartridge'ów. Możliwość rozbudowy pamięci RAM do 32 lub 48 KB.

### ATARI 800

Mikroprocesor 6502 z zegarem 1,8 MHz. Pamięć RAM — 48 KB, ROM — 10 KB (system operacyjny). Interpreter BASIC-a na cartridge'u. Klawiatura profesjonalna z czterema klawiszami funkcyjnymi. Trzy tryby

tekstowe i dziewięć graficznych. Wbudowany głośnik. Cztery porty joysticków i dodatkowe gniazdo cartridge'ów.

### ATARI 800XL

Podstawowy i najpopularniejszy model Atari. Mikroprocesor 65C02 z zegarem 1,8 MHz. Pamięć RAM — 64 KB, ROM — 24 KB (w tym interpreter BASIC-a — 10 KB). Klawiatura profesjonalna z pięcioma klawiszami funkcyjnymi (START, SELECT, OPTION, HELP i RESET). Pięć trybów tekstowych i 11 graficznych. Dwa porty joysticków, wyprowadzona na zewnątrz szyna systemu (równoległa).

### ATARI 600XL

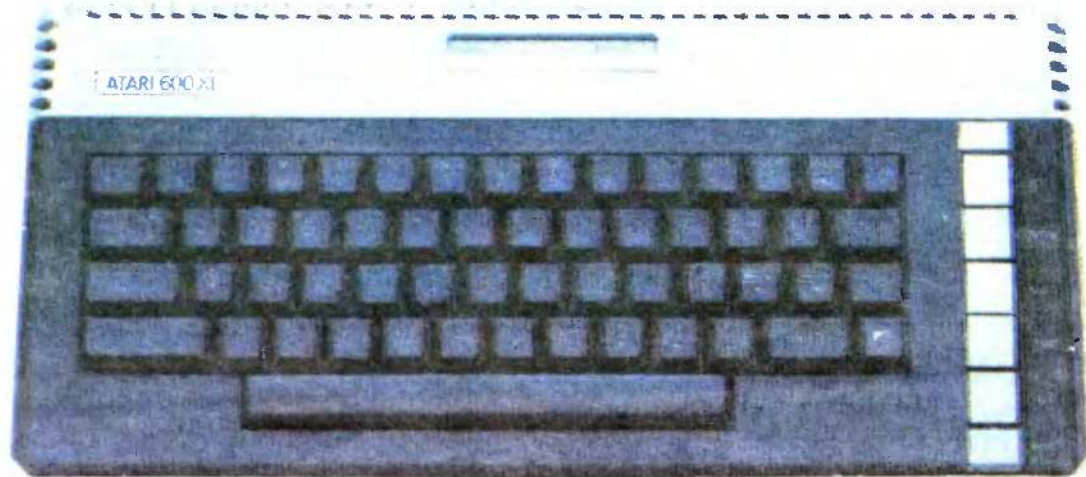
Zmniejszona wersja 800XL. Pamięć RAM o pojemności 16 KB z możliwością przyłączenia zewnętrznego modułu 64 KB. Stosowane są także rozszerzenia pamięci wbudowane do wnętrza komputera na oddzielnej płytce. Pozostałe dane jak 800XL.

### ATARI 1200XL

Nieco rozbudowana wersja 800XL. Klawiatura profesjonalna z dziewięcioma klawiszami funkcyjnymi (dodatkowo F1, F2, F3 i F4). Nie posiada wbudowanego interpretera BASIC-a. Wprowadzone zostały również niewielkie zmiany w systemie operacyjnym. Pozostałe dane jak 800XL. W Polsce praktycznie nie spotykany.

### ATARI 1450XLD

Kolejna odmiana 800XL. Klawiatura profesjonalna z dziewięcioma klawiszami funkcyjnymi. Posiada wbudowaną dwustronną



ATARI 600XL

stację dysków elastycznych pracującą w podwójnej gęstości (po 180 KB na stronę dyskietki). Produkowany w niewielkich ilościach i bardzo mało popularny.

### ATARI 65XE

Praktycznie jest to Atari 800XL w nowej obudowie. Widoczną zmianą w stosunku do 800XL jest brak wyprowadzenia całej szyny systemu. Oprócz tego zastosowano inny układ kontrolera pamięci, co utrudnia rozszerzenie jej pojemności (w poprzednich modelach bardzo proste).

Pamięć jest dostępna programowo nawet z poziomu BASIC-a, a przy pracy ze stacją dysków zakładany jest w niej RAMdysk (499 sektorów). Istnieje także wiele programów wykorzystujących dodatkową pamięć. Pozostałe dane bez zmian. Planowano również wersję 260XE z pamięcią RAM o pojemności 256 KB.

### ATARI 256XT

Wychodząc naprzeciw zapotrzebowaniu użytkowników Atari 800XL kilka firm opracowało rozszerzenie pamięci RAM do 128,



ATARI 800

### ATARI 800XE

W zasadzie jest to 65XE. Jedyną różnicą polega na tym, że na płycie głównej komputera jest już przewidziane miejsce na rozszerzenie pamięci (podstawki dla układów RAM).

### ATARI 130XE

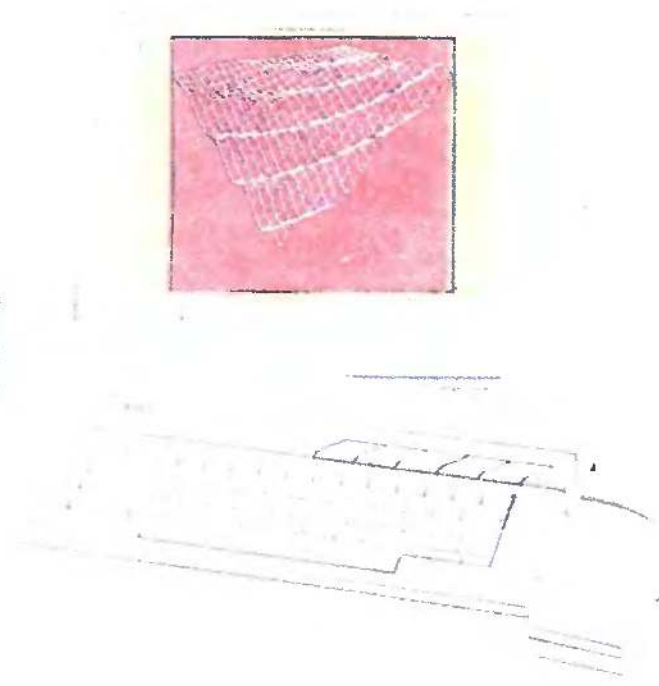
Wersja 65XE o dwukrotnie powiększonej pamięci RAM (do 128 KB). Dodatkowa pa-

mięć 256 lub 512 KB, a nawet 1 MB (Magna Systems). Najpopularniejsze okazało się rozszerzenie do 256 KB. Istnieje kilka wersji tego rozszerzenia, a jako standard został przyjęty system Axlon i takie komputery zyskały nazwę Atari 256XT. Rozszerzenie pamięci do 256 KB jest wykonywane również w Polsce, lecz niestety nie jest ono kompatybilne z Axlon.

Wojciech Zientara



ATARI 800 XL



ATARI 130 XE



## Kupiłeś komputer. Po zmontowaniu według instrukcji całego systemu pojawia się problem. Oczywiście trzeba wczytać jakiś program. Tylko jak to zrobić?

W skład systemu może wchodzić magnetofon lub stacja dysków. Programy mogą być napisane w języku maszynowym lub w BASIC-u. Te ostatnie z kolei mogą być ładowane i uruchamiane automatycznie lub przez użytkownika. Daje to kilka różnych kombinacji. Opiszemy je kolejno.

### ŁADOWANIE Z MAGNETOFONU

Najprostsze jest wczytywanie z magnetofonu programów w języku maszynowym. Ustaw taśmę w kasecie tuż przed początkiem programu, który chcesz załadować. Wciśnij klawisze START i OPTION, a następnie włącz komputer. Klawisze można puścić po kilku sekundach — gdy na ekranie ukaże się obraz. Usłyszysz wtedy pojedynczy dźwięk. Wciśnij klawisz PLAY w magnetofonie i naciśnij dowolny klawisz komputera (oprócz BREAK). Program zapisany na kasecie wczyta się i uruchomi automatycznie.

Podobnie postępujemy z samouruchamiającymi się programami w BASIC-u. Różnica polega na tym, że przy włączaniu komputera trzymamy wciśnięty tylko klawisz START (wciśnięcie OPTION odłącza interpreter BASIC-a i program nie mógłby działać).

Programy w BASIC-u mogą być zapisane oprócz tego trzema różnymi instrukcjami BASIC-a: SAVE, CSAVE i LIST (zobacz opis w dalszej części artykułu). Do odczytania takiego programu należy użyć instrukcji odpowiadającej instrukcji zapisu. Włącz komputer (nie wciskając żadnego klawisza), poczekaj na ukazanie się napisu READY i ustaw taśmę na początku programu. Teraz w zależności od instrukcji użytej do zapisu należy wpisać odpowiednią instrukcję odczytu i naciśnąć klawisz RETURN. Pary odpowiadających sobie instrukcji są następujące:

CSAVE	— CLOAD
SAVE "C:"	— LOAD "C:"
LIST "C:"	— ENTER "C:"

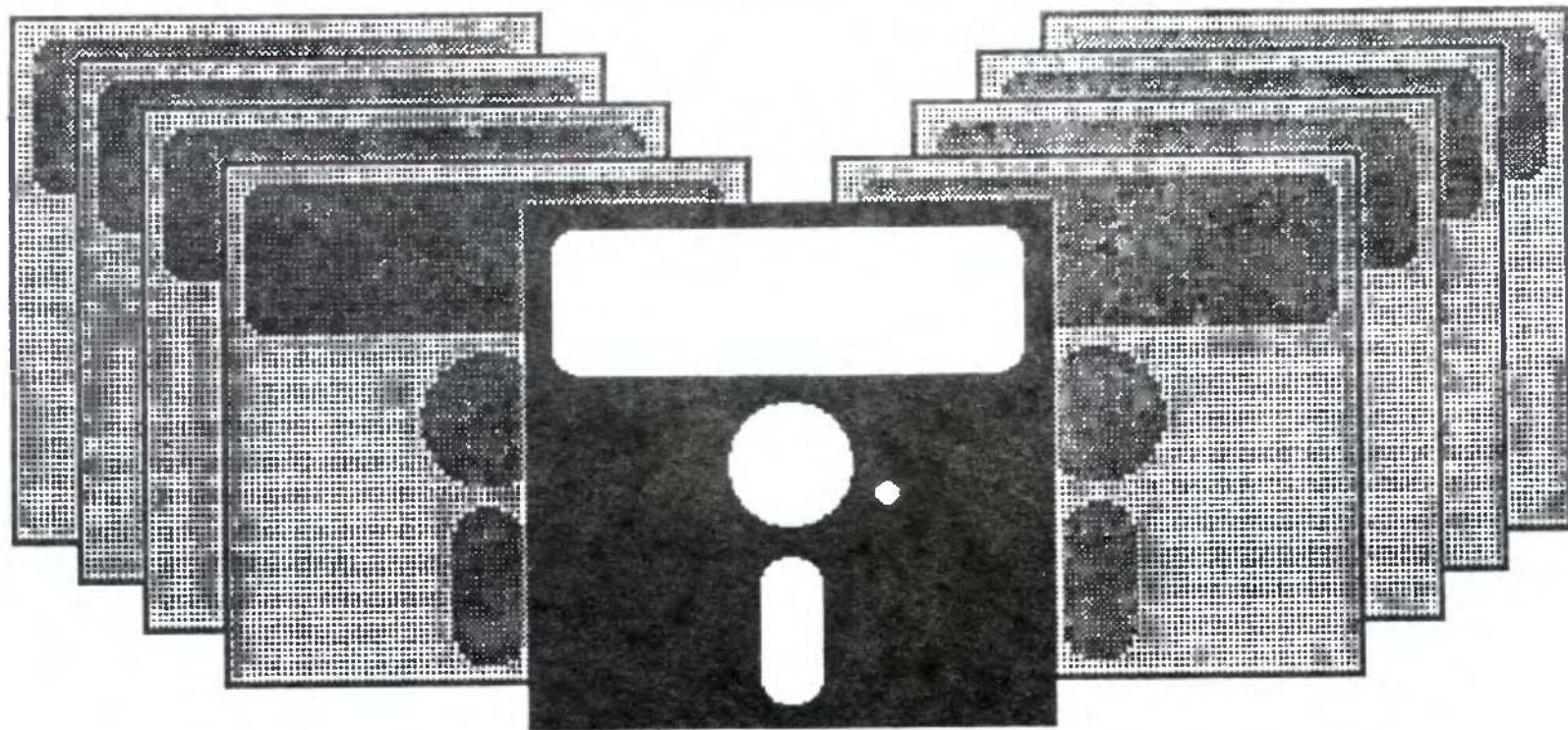
Po naciśnięciu RETURN usłyszysz pojedynczy dźwięk. Wciśnij klawisz PLAY w magnetofonie i naciśnij dowolny klawisz komputera (oprócz BREAK). Program zostanie wczytany z kasy do pamięci komputera. Jeśli użyłeś niewłaściwej instrukcji odczytu, to zostanie wyświetlony raport ERROR — 21. Spróbuj wtedy odczytać program przy pomocy innej instrukcji. Aby teraz uruchomić wczytany program trzeba wpisać polecenie RUN i ponownie naciśnąć RETURN. W przypadku programów zapisanych przez SAVE można uprościć tę procedurę zamiast LOAD "C:" wpisując od razu RUN "C:".

Zapytasz, po co tyle instrukcji do jednej operacji? Instrukcje CLOAD, LOAD i RUN powodują skasowanie zawartości pamięci przed wczytaniem nowego programu, natomiast ENTER dopisuje nowy program do już istniejącego w pamięci. Instrukcja RUN umożliwia także wywołanie i uruchomienie jednego programu z wnętrza drugiego.

### ŁADOWANIE ZE STACJI DYSKÓW

Programy w języku maszynowym mogą występować w formie całodyskowego pliku samouruchamiającego się, pliku uruchamianego z DOS-u oraz pliku uruchamianego przez tzw. inicjalizer.

W pierwszym przypadku wystarczy włożyć dyskietkę z programem do stacji dysków, wcisnąć klawisz OPTION (bez START — powoduje on odczyt z magnetofonu) i włączyć komputer. Program wczyta się i uruchomi samoczynnie.



# ŁADOWANIE I ZAPISYWANIE — PROGRAMÓW —

Początkowe czynności w trzecim przypadku są identyczne, inny jest jedynie efekt końcowy. Zamiast programu wczytuje się inicjalizer, który wyświetla spis programów zawartych na dyskietce. Programy są oznaczone kolejnymi literami alfabetu. Teraz wystarczy tylko nacisnąć klawisz z odpowiednią literą i chwilę poczekać.

Największy kłopot sprawiają programy zapisane „pod DOS-em”. Rozpoczynamy w sposób opisany w pierwszym przypadku. Na ekranie pojawia się menu DOS-u. Jeżeli zobaczysz tylko napis BOOT ERROR, to znaczy, że na dysku nie ma plików DOS-u. Trzeba je wtedy wczytać z dyskietki systemowej i dopiero potem włożyć do stacji dyskietkę z programem. Teraz odczytujemy spis treści funkcją A i wybrany program wczytujemy funkcją L (szczegółowy opis w instrukcji DOS-u).

Podobnie postępujemy przy uruchamianiu programów napisanych w BASIC-u. Jedyna różnica polega na nie wciskaniu klawisza OPTION przy włączaniu komputera. Jeżeli program nie jest samouruchamiający się i na dyskietce nie ma inicjalizera, to po wyświetleniu napisu READY trzeba wpisać instrukcję RUN "D:nazwa" i naciśnąć klawisz RETURN.

### ZAPISYWANIE NA TAŚMIE

Ponieważ początkujący programiści rzadko zaczynają pisanie programów od języka innego niż BASIC, opiszemy tylko zapisywanie programów napisanych w tym języku.

Jak już wcześniej wspomniano zapisu programu można dokonać przy pomocy trzech instrukcji. Wykonywane czynności są we wszystkich trzech przypadkach jednakowe, różnica występuje tylko w sposobie zapisu na taśmie.

Instrukcja LIST "C:" powoduje zapisanie na taśmie programu kodami ATASCII, to znaczy w takiej formie jaką widzimy na ekranie. Pomiędzy poszczególnymi rekordami (blokami danych) zapisywanymi na taśmie pozostawiane są długie przerwy. Ten sposób zapisu zajmuje największe miejsca, lecz pozwala dołączyć program do innego programu znajdującego się w pamięci komputera.

Instrukcja CSAVE zapisuje program w takiej postaci, w jakiej jest przechowywany w pamięci komputera. Każdej instrukcji odpowiada symboliczny kod (tzw. token). Oprócz tego na początku programu zapisywana jest tablica nazw zmiennych i inne parametry programu. Pomiędzy rekordami pozostawiane są krótkie przerwy. Program zapisany w ten sposób zajmuje na taśmie najmniej miejsca.

Niejąko pośrednim sposobem jest użycie instrukcji SAVE "C:". Program jest zapisywany

tokenami jak po CSAVE, lecz przerwy pomiędzy rekordami są długie. Tylko program zapisany w taki sposób można wczytać i uruchomić jedną instrukcją (RUN "C:").

Aby zapisać znajdujący się w pamięci program, wpisz w trybie bezpośrednim (bez numeru linii) instrukcję zapisu (wybraną z trzech wyżej podanych). Usłyszysz podwójny dźwięk. Ustaw taśmę w magnetofonie w miejscu, od którego chcesz rozpocząć zapis. Wciśnij teraz klawisze PLAY i RECORD w magnetofonie i naciśnij dowolny klawisz w komputerze (oprócz BREAK). Program zostanie zapisany w sposób zależny od podanej instrukcji. Warto zapisać stan licznika magnetofonu przed rozpoczęciem nagrywania i po jego zakończeniu. Ułatwi to późniejsze odszukanie początku programu oraz zmniejszy ryzyko zapisania następnego programu na końcu poprzedniego.

### ZAPISYWANIE NA DYSKIECIE

Zapis na dyskietce niewiele różni się od zapisu na taśmie. Dostępne są tylko dwie instrukcje zapisu (LIST "D:nazwa" i SAVE "D:nazwa") i nie można tu mówić o długości pozostawianych przerw.

Podaj odpowiednią instrukcję zapisu w trybie bezpośrednim, a program zostanie zapisany w żądany sposób. I to prawie wszystko. Jak widać współpraca ze stacją dysków jest znacznie prostsza niż z magnetofonem.

Napisałem „prawie”, gdyż pozostał jeszcze problem nazwy programu. Fachowo nazywa się to specyfikacją pliku. W komputerach ATARI składa się ona maksymalnie z trzech części. Pierwszą, która występuje zawsze, jest nazwa urządzenia i następujący po niej dwukropek. Jest to jednoliterowy kod urządzenia zewnętrznego z dodaną ewentualnie liczbą określającą numer, np. C: — magnetofon, D: lub D1: — stacja dysków numer 1, D2: — stacja dysków numer 2. Dwie pozostałe części specyfikacji są używane tylko przy współpracy ze stacją dysków. Jest to nazwa pliku i jego identyfikator oddzielone od siebie kropką. Nazwa pliku może składać się maksymalnie z ośmiu znaków (dużych liter lub cyfr), a identyfikator maksymalnie z trzech znaków. Nazwa pliku jest obowiązkowa, a identyfikator może zostać pominięty. Niektóre Dyskowe Systemy Operacyjne wymagają dodatkowo, aby pierwszym znakiem nazwy pliku była litera.

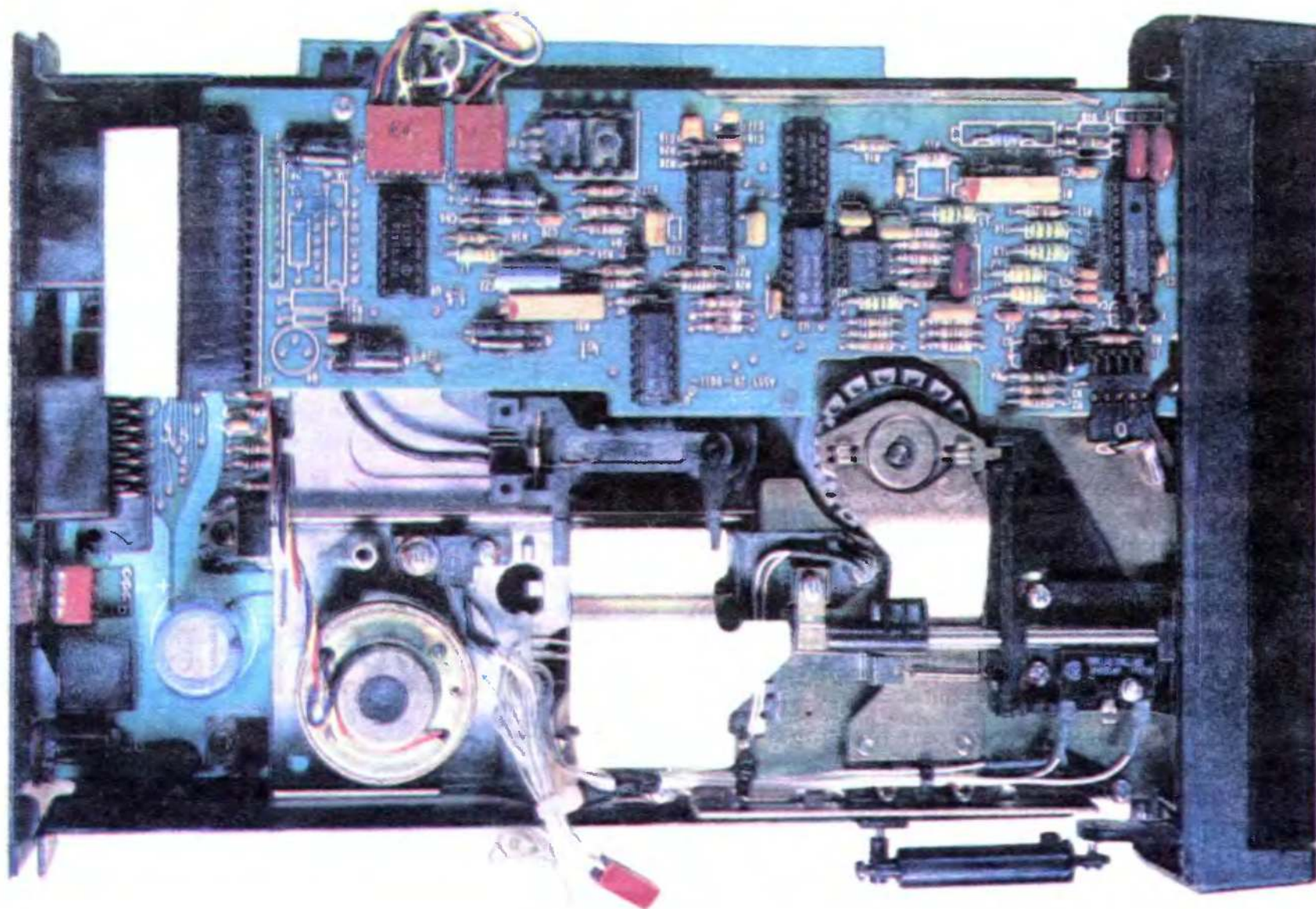
Mam nadzieję, że powyższy opis usunie wątpliwości związane z wczytywaniem i zapisywaniem programów na nośnikach pamięci masowej.

Wojciech Zientara



# LDW SUPER 2000

## 2000



LDW Super 2000 jest jednostronną stacją dysków 5,25 cala przeznaczoną do współpracy z 8-bitowymi komputerami Atari (800XL, 65XE i 130XE). Umożliwia zapis i odczyt dyskietek w pojedynczej (single), rozszerzonej (enhanced) i podwójnej (double) gęstości. Połączona jest z komputerem przy pomocy standardowego złącza szeregowego Atari.

### BUDOWA

Obudowa stacji jest mieszana: tył i przód są wykonane z tworzywa sztucznego, natomiast pozostała część korpusu z barwionego na czarno aluminium. Oprócz tego przód stacji jest chroniony unoszonymi do góry drzwiczkami. Wymiary obudowy są nieco mniejsze niż stacji Atari 1050 i wynoszą 183 x 280 x 72 mm (szerokość x długość x wysokość). Na ścianie tylnej znajduje się wyłącznik zasilania, gniazdo przewodu zasilającego, przełącznik trybów pracy oraz dwa złącza szeregowo. Z przodu oprócz kieszeni dyskietki i zamykającej ją dźwigni umieszczono trzy lampki kontrolne, dwucyfrowy wyświetlacz LED i cztery przełączniki.

Układ mechaniczny jest zapożyczony ze stacji 1050 i składa się z silnika napędowego obracającego dyskietkę silnika krokowego poruszającego głowicę oraz układów przeniesienia napędu.

Znacznie ciekawszy jest układ elektroniczny stacji. Został on zbudowany w oparciu o procesor Z80A i kontroler dysku WD2797A. Program pracy procesora jest zapisany w pamięci EPROM 2732 o pojemności 4 KB. Komunikację z komputerem poprzez złącze szeregowo zapewnia układ CD M6116.

### PRACA

Współpraca z komputerem przebiega pod nadzorem Dyskowego Systemu Operacyjnego.

Do stacji dołączony jest DOS XL 2.35L. Umożliwia on zapis w pojedynczej lub podwójnej gęstości. Przy pojedynczej gęstości dyskietka zawiera 40 ścieżek po 18 sektorów 128-bajtowych, a więc 720 sektorów o łącznej pojemności 92160 bajtów (90 KB). W podwójnej gęstości sektory są 256-bajtowe, co daje pojemność 184360 bajtów (180 KB). Przy zastosowaniu DOS 2.5 można zapisać dyskietkę w rozszerzonej gęstości: 40 ścieżek po 26 sektorów 128-bajtowych (133120 bajtów = 130 KB). Oprócz tego sprawdzałem pracę stacji po kontrolą innych systemów operacyjnych: DOS 4.0, MYDOS 4 2C, Sparta DOS 3.2d i K-DOS. We wszystkich przypadkach praca przebiegała sprawnie i nie było trudności ze zmianą gęstości zapisu.

Przesyłanie danych do i z komputera przebiega z szybkością 19200 bodów. DOS XL 2.35L zawiera program SYNCHROMESH, który dokonuje zmian w systemie operacyjnym komputera, co pozwala na transmisję danych z dwukrotnie większą szybkością. Ponieważ jednak zmodyfikowany system operacyjny wykorzystuje pamięć RAM położoną pod ROM, niemożliwe jest zainstalowanie RAMdysku (nawet w 130XE). Zmusza to do dokonania wyboru: albo szybka praca z zewnętrzną stacją dysków, albo praca z RAMdyskiem. Na korzyść tego pierwszego rozwiązania przemawia dodatkowo znaczne spowolnienie transmisji bez użycia SYNCHROMESH-a z powodu specyficznego sposobu formatowania dyskietki. Z drugiej jednak strony żadna stacja nie może dorównać szybkością RAMdyskowi. Ostateczna decyzja jest więc zależna przede wszystkim od upodobań użytkownika, choć moim zdaniem jest to niedopatrzenie projektantów systemu. Dla porównania podaję wyniki pomiarów czasu odczytu pełnej dyskietki.

DOS 2.5 — pojedyncza gęstość (90 KB) 1'25"

DOS 2.5 — rozszerzona gęstość (130 KB)	1'58"
DOS XL z SYNCHROMESH — pojedyncza gęstość (90 KB)	1'00"
DOS XL z SYNCHROMESH — podwójna gęstość (180 KB)	1'53"
DOS XL bez SYNCHROMESH — pojedyncza gęstość (90 KB)	3'14"
DOS XL bez SYNCHROMESH — podwójna gęstość (180 KB)	3'44"

### DODATKOWE MOŻLIWOŚCI

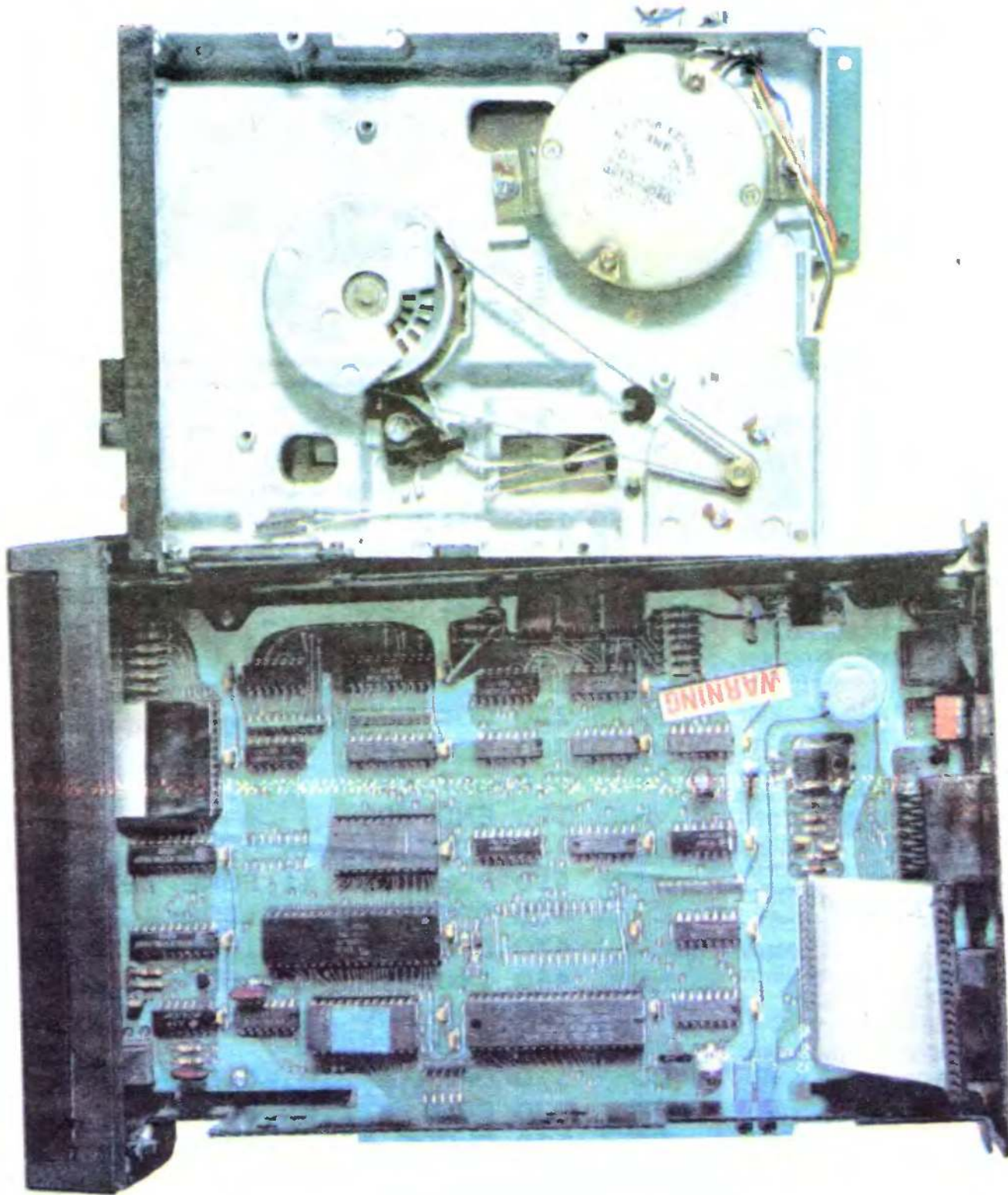
Zastosowany procesor Z80A ma podczas pracy stacji stosunkowo mało do zrobienia. Aby się nadmiernie nie „nudził”, został wykorzystany do przekazywania użytkownikowi informacji o pracy stacji poprzez dwucyfrowy wyświetlacz LED. Rodzaj informacji jest wybierany przez operatora naciśnięciem jednego z przycisków na przedniej ścianie stacji. Przycisk „DRIVE TYPE” powoduje wyświetlenie numeru stacji i gęstości w jakiej aktualnie pracuje. Po naciśnięciu „TRACK” możemy dowiedzieć się, na której ścieżce znajduje się głowica stacji. W przypadku wystąpienia błędu w pracy stacji lub po naciśnięciu „ERROR” wyświetlany jest literowo-cyfrowy kod błędu. Rolę informacyjną pełni także trzy lampki kontrolne. „POWER” sygnalizuje włączenie zasilania, „BUSY” — pracę stacji, a „PROTECT” — zabezpieczenie dyskietki przed zapisem przez zaklejenie wycięcia. Dodatkowy przycisk „PROTECT” uniemożliwia zapis na dyskietce niezaklejonej, co również jest sygnalizowane zapaleniem lampki. Wszystkie te informacje ułatwią zaawansowanemu użytkownikowi korzystanie ze stacji, a początkującemu zajmą czas przy wczytywaniu lub zapisywaniu długich programów.

### EKSPLLOATACJA

Zasadniczą zaletą stacji LDW jest możliwość wyjęcia dyskietki naty-

**Dzięki uprzejmości P.Z. Karen prowadzącego serwis komputerów Atari otrzymaliśmy do przetestowania nową stację dysków do komputerów Atari XL/XE. Stacja dysków LDW Super 2000 znajdzie się w sprzedaży w sklepach Pewex-u na początku grudnia.**





chmiast po zakończeniu zapisu lub odczytu. Jest to szczególnie ważne przy pracy z programami, które wymagają częstej zmiany dyskietek (np. „Print Shop”). Korzystając ze stacji Atari 1050 trzeba za każdym razem odczekać kilkanaście sekund, co zwykle jest bardzo denerwujące. Dobrym pomysłem jest zamykanie stacji oddzielną pokrywą zapobiegającą jej zakurzeniu. Zastosowanie obudowy metalowej znacznie zwiększa odporność na przypadkowe uderzenia i poprawia odprowadzanie ciepła przyczyniając

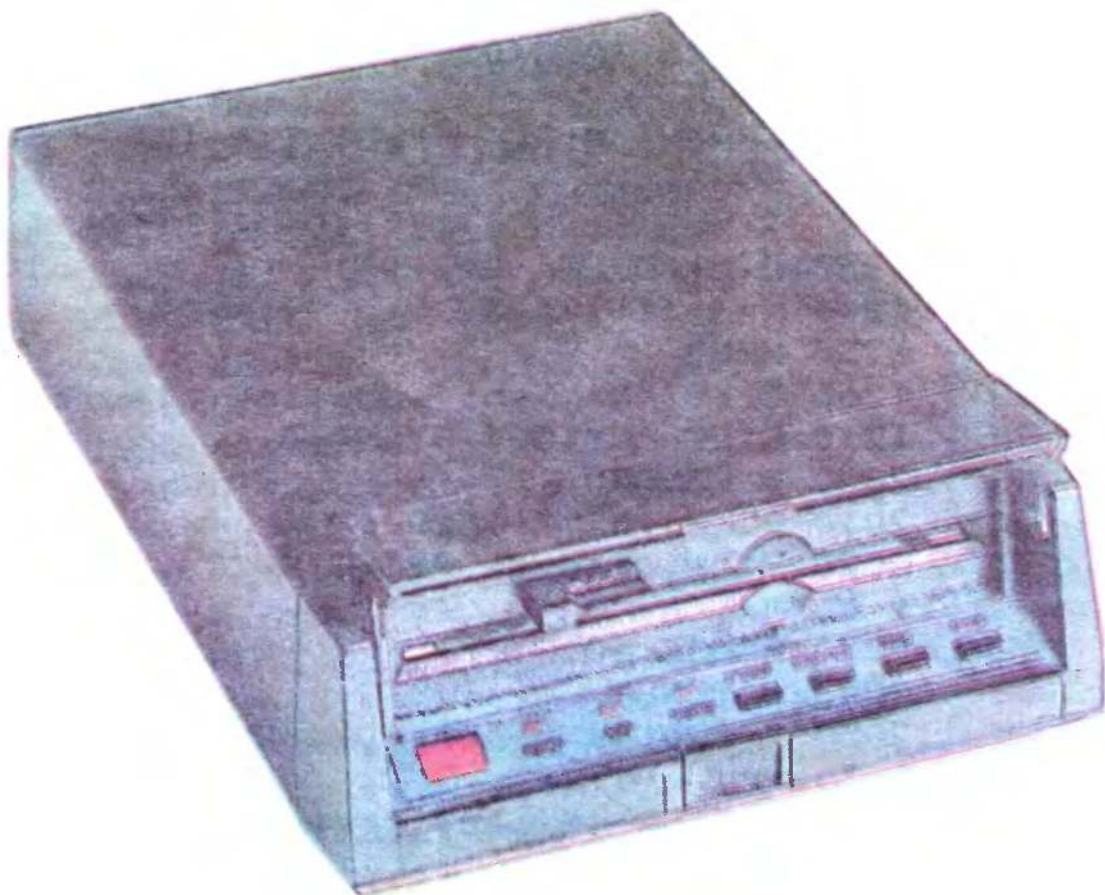
się do przedłużenia żywotności stacji. Niestety warunki obsługi są gorsze niż w 1050. Kieszeń dyskietki jest umiejscowiona nieco głębiej i ma skróconą dźwignię, a uniesione drzwiczki dodatkowo utrudniają dostęp. W efekcie otwieranie i zamykanie stacji wykonuje się dwoma palcami, co jest niezbyt wygodne. Umieszczenie wyłącznika sieciowego z tyłu chroni co prawda przed niespodziewaną „interwencją” dziecka, lecz znajduje się on zbyt blisko gniazda przewodu zasilającego. Każdorazowo powoduje to pocruszenie

przewodu i może spowodować po pewnym czasie obłuzowanie styków i wystąpienie przerw w zasilaniu. Wszystko to świadczy o skupieniu się projektantów na konstrukcji wewnętrznej stacji i nieprzemyśleniu do końca zagadnień eksploatacyjnych.

## PODSUMOWANIE

Stacja LDW Super 2000 jest mimo drobnych wad znacznym krokiem naprzód w stosunku do Atari 1050. Polecałbym ją jednak przede wszystkim bardziej zaawansowanym użytkownikom, gdyż początkujący nie zdołają wykorzystać wszystkich jej możliwości. Brak jest na razie specjalnego oprogramowania dostosowanego do tego typu stacji, takiego jak istnieje dla Happy Warp lub Top Drive. Sama instrukcja, choć bardzo dobrze napisana, nie jest wystarczająca. Z drugiej strony bardzo zachęcająca jest możliwość zaoszczędzenia na dyskietkach dzięki zwiększeniu gęstości zapisu o prawie 50%. Należy tu wrzucić kamyczek do ogródka Pewex-u. Jak mi wiadomo po sprowadzeniu stacji LDW Super zaprzestanie się sprzedaży Atari 1050. Użytkownik nie będzie więc miał możliwości wyboru. Może w ten sposób zuniłkować również inny sprzęt i sprzedawać jeden typ telewizorów, magnetofonów czy radioodbiorników albo jeden fason odzieży i jeden gatunek papierosów.

Wojciech Zientara



**Nie** wszyscy chyba wiedzą, że Atari było pierwszą firmą, która zadebiutowała na rynku amerykańskim konsolami do gier wideo (np. model 2600). W nawiązaniu do tej niejako już tradycji, latem tego roku Atari zaprezentowało nową konsolę, 65 XE Game. Jest to ni mniej, ni więcej, tylko jeden z pierwszych 8-bitowych komputerów Atari w nowym opakowaniu, wyposażony w gniazdo przyjmujące cartridge z grami, ale mogący również dodatkowo być wyposażony w klawiaturę i magnetofon kasetowy. Konsola jest w pełni kompatybilna z około setką gier na Atari. Cena: poniżej 200 dolarów.

**IMG** SCAN — to nazwa nowego skanera współpracującego z Atari ST i jakąkolwiek drukarką. Na zestaw składa się kartridż i pióro świetlne, które zakłada się na głowicę drukarki. Pióro analizuje 256 odcieni szarości, odczytuje również kolory i działa we wszystkich trzech trybach rozdzielczości, oferując aż 9 możliwych zmniejszeń (powiększeń) oryginału. Skaner jest kompatybilny z NEO i DEGAS.

**W** wiadomo, że na monitorze kolorowym SC 1224 nie można uzyskać najwyższej rozdzielczości. Ciągłe podłączanie monitora monochromatycznego SM 124/125, na miejsce odłączonego SC 1224 jest i męczące i niezbyt bezpieczne (ścierają się wtyki). Jednym z wygodnych rozwiązań jest uniwersalny przełącznik firmy SPD, eliminujący tę usterkę.

Przełącznik — podłączony do wejścia monitorowego ST, może przyjąć naraz oba monitory — mono i kolor — a poza tym — dzięki wtykowi DIN — również zestaw HI-FI. Przełączanie monitorów odbywa się ręcznie poprzez przesuw dźwignienki umieszczonej na zewnątrz urządzenia. Eliminuje się w ten sposób wiele problemów. Do jednych z nich należy szybkie zużywanie się styków, co powoduje nieprawidłową inicjalizację po wciśnięciu RESET. Dlatego właśnie niektóre ST startują dwukrotnie zanim zgłosi się system operacyjny.

**T**rack-ball — odmiana joysticka w formie dużej kuli zamontowanej na szerokiej podstawie — dostępny do tej pory dla 8-bitowych komputerów Atari, został ostatnio zaadaptowany również dla potrzeb posiadaczy ST. Główną atrakcją tego urządzenia jest fakt, iż jedynym ruchomym elementem jest obracająca się dookoła kula. Sam track-ball natomiast jest nieruchomy; idealne rozwiązanie dla osób mających mało miejsca na biurku. Track-ball symuluje również joystick oraz myszkę, a dokładniej — jej lewy przycisk, prawy symulowany jest przez Alternate-Home na klawiaturze.



# PRINT

# SHOP

**Jednym z najlepszych programów graficzno-tekstowych dla komputerów Atari jest „Print Shop” firmy Bröderbund Software. Posiada on jednak jedną poważną wadę — niestandardowy format zapisu na dysku. Uniemożliwia to wykorzystanie wykonanych przy jego pomocy obrazków do innych programów graficznych oraz drukowanie ich na drukarce Atari 1029. Zamieszczony poniżej program pozwala przetworzyć obrazki zapisane w formacie Print Shopa na obrazy w trybie GRAPHICS 8 w formacie DOS 2.5.**

Program jest napisany w języku Turbo-BASIC XL, co pozwoliło na jego znaczne skrócenie i zwiększenie szybkości działania. Aby go wpisać, należy najpierw wczytać interpreter Turbo-BASIC XL, następnie Edytor BASIC-a i dokładnie przepisać poniższy wydruk. Po wpisaniu całego programu zapisujemy go instrukcją LIST "D:PSCONV.LST",1,31999 (dysk) lub LIST "C:",1,31999 (kasetka), wpisujemy rozkaz NEW i ponownie wczytujemy program instrukcją ENTER "D:PSCONV.LST" lub ENTER "C:". Teraz można już zapisać gotowy program przez SAVE "D:PSCONV.TXL" lub CSAVE.

Używanie programu jest bardzo proste. Po jego uruchomieniu wyświetlane jest menu. Pierwszą wybraną funkcją musi być zawsze odczyt spisu zawartości dyskietki danych w formacie Print Shopa — jest to konieczne do pracy programu. Spis ten można także wydrukować. Następnie wybieramy opcję odczytu obrazu. Po wybraniu tej opcji komputer pyta o numer obrazu, który chcemy przetworzyć na format DOS 2.5 — chodzi tu o numer kolejny obrazu w spisie zawartości dyskietki Print Shopa. Program automatycznie rozpoznaje, czy to jest obraz całoekranowy (wykonany opcją Screen Magic), czy ikona (na 1/4 ekranu) i wczytuje go do pamięci. Jeżeli zdecydujemy się przekształcić ten obraz, to wymieniamy dyskietkę na inną, zaformatowaną przez DOS 2.5 i wybieramy opcję zapisu obrazu. Znajdujący się w pamięci obraz zostanie zapisany na dysku w standardzie GRAPHICS 8 (62 sektory) lub jako ikona (standard Typesettera — 16 sektorów).

Po zapisaniu wszystkich potrzebnych obrazków w formacie DOS 2.5 możemy przystąpić do ich dalszej obróbki lub drukowania. Można, na przykład, przy użyciu jakiegoś programu graficznego (najlepiej „Design Master”) dorobić „ogonki” w celu uzyskania polskich liter. Ikony w formacie Typesettera można natomiast wykorzystać do tworzenia mieszanych, tekstowo-graficznych wydruków przy pomocy programów „Rubber Stamp” i „Page Designer”.

Wojciech Zientara

**UWAGA: Nie należy wpisywać liter wydrukowanych przed numerem linii programu. NIE SĄ one częścią linii, lecz kodem kontrolnym „Edytora BASIC-a” (zob. str. 3).**

```

KJ 1 POKE 566,158
QQ 10 DIM V$(128),D$(4096),S$(8000)
AY 20 DIM FL$(15),A$(20),B$(10),D(128)
FC 30 OFF=40000
VX 40 S$(%)=" ";S$(8000)=" ";S$(%)=S$
KU 50 D$=S$(%,4096):V$=D$(%,128)
JR 60 V=ADR(V$):D=ADR(D$):S=ADR(S$)
NN 70 # START
VS 80 TRAP OFF:GRAPHICS %0
US 90 ? :? CHR$(127);" PRINT SHOP CONVERT":?
FD 100 ? " (c) 1987 by Wojciech Zientara":?
XI 110 ? :? " Ten program zamienia rysunki lub"
ID 120 ? " ikony PRINT SHOP'a na GRAPHICS 8"
HL 130 ? :? :? CHR$(127);"MENU:"
CQ 140 ? :? "1. PRINT SHOP directory (wymagane)"
DY 150 ? :? "2. Odczyt rysunku PRINT SHOP

```

```

'a"
AM 160 ? :? "3. Zapis rysunku GRAPHICS 8"
RR 170 ? :? "4. Koniec"
HW 180 ? :? :? "Która funkcje wybierasz ?";CHR$(30);
NC 200 # CHOOSE
MJ 210 GET KEY
TS 220 ON KEY-48 GO# PSDIR,PSLOAD,SSAVE,PEND
GX 230 GO# CHOOSE
UM 300 PROC ANY
MK 310 GET KEY
VP 320 ENDPROC
BL 400 PROC RSEC
RI 410 POKE 769,%1:POKE 768,$31
CH 420 DPOKE 772,DB:DPOKE 776,BL
HV 430 DPOKE 778,SN:POKE 770,82
HO 440 I=USR(ADR("hLS[]"))
VW 450 ENDPROC
PR 500 PROC FSAVE
OQ 510 OPEN #%2,8,%0,FL$
PB 520 BPUT #%2,DB,BL
ET 530 CLOSE #%2
VV 540 ENDPROC
HP 1000 # PSDIR
HK 1010 POKE 752,%1:CLS
RH 1020 ? :? "Włoz dysk danych PRINT SHOP a do stacji1 i naciśnij >RETURN<"
GC 1030 EXEC ANY:IF KEY<>155 THEN 1030
PC 1040 DB=V:BL=128:SN=361:EXEC RSEC
GN 1050 IF V$(1,15)<>"PRINT SHOP:CLK!"
PA 1060 ? :? CHR$(253);"To nie jest dysk PRINT SHOPa."
QX 1070 PAUSE 100:GOTO 1020
FQ 1080 ENDIF
UZ 1090 DB=D:BL=128
KC 1100 FOR SN=362 TO 393
HT 1110 EXEC RSEC:DB=DB+BL
TD 1120 NEXT SN
GY 1130 P=%0: ?
RN 1140 ? "Wydruk directory na drukarce T/N)"
LG 1150 EXEC ANY: ?
WD 1160 IF KEY=84 OR KEY=116 THEN P=%1
HN 1170 I=%1:N=%1
FS 1180 WHILE (ASC(D$(I))<>0 AND I<4096)
VA 1190 IF P
SZ 1200 LPRINT N;". ";D$(I,I+15)
EZ 1210 ENDIF
EO 1220 ? N;". ";CHR$(127);D$(I,I+15)
BW 1230 D(N)=DPEEK(D+I+15)
CF 1240 N=N+1:I=I+32
HP 1250 WEND
WH 1260 ? :? "Naciśnij dowolny klawisz"
PH 1270 EXEC ANY:GO# START
NK 2000 # PSLOAD
IC 2010 POKE 752,%1:CLS :?
ZX 2020 ? "Włoz dysk PRINT SHOPa do stacji #1"
KE 2030 PAUSE 10: ?
OS 2040 # PSNUM
LK 2050 ? "Wpisz numer rysunku do wczytania"
SM 2060 ? "lub '#' aby przerwać": ?
OB 2070 # PSNUM2
WK 2080 TRAP #PSNUM:INPUT A$
KR 2090 ON A$="# " GO# START

```



# CONVERTER

```
QK 2100 N=VAL(A$)
SQ 2110 ON N<%1 OR N>128 GO# PSNUM2
TF 2120 TRAP OFF
DV 2130 SN=D(N):?
QI 2140 ON ASC(D$(N*32-12))<>65 GO# ICON
JW 2150 GRAPHICS 24:COLOR %1
WD 2160 SETCOLOR %2,%1,%0
QE 2170 SETCOLOR 4,%1,%0
IV 2180 SETCOLOR %1,%1,14
XM 2190 DB=S:BL=128:LS=61
YD 2200 WHILE LS>%0 AND SN>%0 AND SN<$041
0
WA 2210 EXEC RSEC:LS=LS-%1
RM 2220 DB=DB+126:SN=DPEEK(DB)
HK 2230 WEND
UU 2240 SA=DPEEK(DPEEK(560)+4)
LL 2250 MOVE S,SA,7680
UH 2260 PAUSE 100:TYP=%1
UI 2270 TEXT 100,178,"Nacisnij RETURN..."
WD 2280 EXEC ANY
WY 2290 IF KEY<>155 THEN 2280
ID 2300 GO# START
UU 2310 # ICON
NM 2320 ON ASC(D$(N*32-12))<>88 GO# BAD
JU 2330 GRAPHICS 24:COLOR %1
WB 2340 SETCOLOR %2,%1,%0
QC 2350 SETCOLOR 4,%1,%0
IT 2360 SETCOLOR %1,%1,14
HD 2370 DB=S:BL=128:LS=5
ZD 2380 WHILE LS>%0 AND SN>%0 AND SN<$041
0
XA 2390 EXEC RSEC:LS=LS-%1
RK 2400 DB=DB+126:SN=DPEEK(DB)
HI 2410 WEND
ZB 2420 SA=DPEEK(DPEEK(560)+4)+2815
WO 2430 K=S:L=SA
ND 2440 FOR I=%1 TO 52
NL 2450 MOVE K,L,11
ID 2460 K=K+11:L=L+40
FT 2470 NEXT I
XD 2480 K=S:L=SA
OI 2490 FOR I=%1 TO 80
MV 2500 MOVE L,K,20
HP 2510 K=K+20:L=L+40
FG 2520 NEXT I
PD 2530 PLOT 119,69:DRAWTO 208,69
VX 2540 DRAWTO 208,122:DRAWTO 119,122
MQ 2550 DRAWTO 119,68:DRAWTO 209,68
XP 2560 DRAWTO 209,123:DRAWTO 118,123
HK 2570 DRAWTO 118,68:TYP=%2
UR 2580 TEXT 100,178,"Nacisnij RETURN..."
WM 2590 EXEC ANY
WF 2600 IF KEY<>155 THEN 2280
IM 2610 GO# START
HB 2620 # BAD
BR 2630 ? CHR$(253);"Zly numer, wpisz ponownie"
NV 2640 GO# PSNUM2
GY 3000 # SSAVE
ID 3010 POKE 752,%1:CLS :?
QX 3020 # FILE
YJ 3030 IF TYP=%1 THEN ? "Wpisz nazwe rysunku do zapisania,"
CM 3040 IF TYP=%2 THEN ? "Wpisz nazwe ikony do zapisania,"
TK 3050 ? "lub RETURN dla directory,"
XII 3060 ? "lub '#' aby przerwac.":?
YD 3070 TRAP #FILE:INPUT A$:TRAP OFF
KP 3080 ON A$="#" GO# START
GC 3090 # DRIVE
LH 3100 TRAP #DRIVE:? "stacja # ";
VH 3110 EXEC ANY
QN 3115 IF KEY<49 OR KEY>57
JT 3120 B$="1"
HQ 3125 ELSE
OV 3130 B$=CHR$(KEY)
FZ 3135 ENDIF
DF 3140 ? B$
JD 3150 IF VAL(B$)<1 OR VAL(B$)>9 THEN GO# DRIVE
LZ 3160 FL$="D":FL$(2)=B$
CM 3170 ON A$<>" " GO# NAME
SE 3180 FL$(3)=":*. *"
FY 3190 TRAP #OPER:OPEN #%3,6,0,FL$
DP 3200 # REDIR
CB 3210 TRAP #FRS:INPUT #%3,A$:? A$,
EK 3220 INPUT #%3,A$:? A$:GO# REDIR
TU 3230 # FRS
SI 3240 CLOSE #%3:TRAP OFF
WG 3250 ? :? "Nacisnij dowolny klawisz"
VY 3260 EXEC ANY
NG 3270 CLS :? :GO# FILE
RV 3280 # NAME
ML 3290 FL$(3)=":"
ZL 3300 FOR I=%1 TO LEN(A$)
ZU 3310 IF A$(I,I)<>" " THEN NEXT I
CP 3320 IF I<LEN(A$) THEN A$=A$(I+1):POP
WG 3330 FL$(4)=A$
CK 3340 DB=S
KA 3350 IF TYP=%1 THEN BL=7680
XD 3360 IF TYP=%2 THEN BL=1600
QZ 3370 TRAP #OPER
WU 3380 EXEC FSAVE
UF 3390 TRAP OFF
IG 3400 GO# START
YS 4000 # OPER
PM 4010 ? :? :? CHR$(253);CHR$(127);"BLAD - ";ERR:? :? CHR$(127);
SV 4020 IF ERR=128 THEN ? "Nacisniety BREAK"
GR 4030 IF ERR=130 THEN ? "Urzadzenie nie istnieje"
VL 4040 IF ERR=138 THEN ? "Brak odpowiedzi"
BD 4050 IF ERR=139 THEN ? "Brak komunikacji"
FB 4060 IF ERR=144 THEN ? "Wykonanie niemozliwe"
RW 4070 IF ERR=160 THEN ? "Zly numer stacji"
IP 4080 IF ERR=162 THEN ? "Dysk pelny"
GR 4090 IF ERR=165 THEN ? "Zla nazwa pliku"
RH 4100 IF ERR=167 THEN ? "Plik chroniony"
BK 4110 IF ERR=169 THEN ? "Pamiec stacji pelna"
NI 4120 IF ERR=173 THEN ? "Zly format"
GJ 4130 IF ERR=177 THEN ? "Zly dysk"
WC 4140 ? :? "Nacisnij dowolny klawisz"
VU 4150 EXEC ANY
IT 4160 GO# START
SU 5000 # PEND
KI 5010 CLOSE :DIR :NEW
```



## JAK UŻYWAĆ ATARI 1029

**Według powszechnego mniemania drukarka Atari 1029 jest urządzeniem o niewielkich możliwościach, szczególnie w zakresie drukowania grafiki. Jednak dzięki zastosowaniu odpowiedniego oprogramowania można na niej uzyskać bardzo interesujące wydruki graficzne.**

Najważniejszym programem, który powinien znajdować się w posiadaniu każdego właściciela Atari 1029, jest program „Print 1029”. Umożliwia on drukowanie rysunków zapisywanych w standardzie „Koala” w wersji skondensowanej oraz w standardzie GRAPHICS 8. Jego działanie polega na przekształcaniu zawartości pamięci ekranu na postać zrozumiałą dla drukarki pracującej w trybie graficznym. Drugim programem o analogicznym działaniu jest „Picture Converter V.2.0”. Dodatkowo umożliwia on obracanie rysunków wokół osi pionowej lub poziomej. Mając jeden z tych programów można już wydrukować każdy znajdujący się na dysku rysunek w jednym z podanych standardów.

A jeśli rysunek ma inny standard, np. „Visualizer” lub „Fun with Art”? W tym przypadku pomocny będzie program „Rapid Graphics Converter”, który pozwala na przekształcanie rysunków wykonanych w jednym z 10 standardów na inny. Przy okazji trzeba zwrócić uwagę na znaczne podobieństwo standardu „MicroPainter” i GRAPHICS 8. Jedyna różnica polega na tym, że „MicroPainter” zapisuje na końcu pliku zawartości rejestrów koloru, więc plik „MicroPainter” zawiera

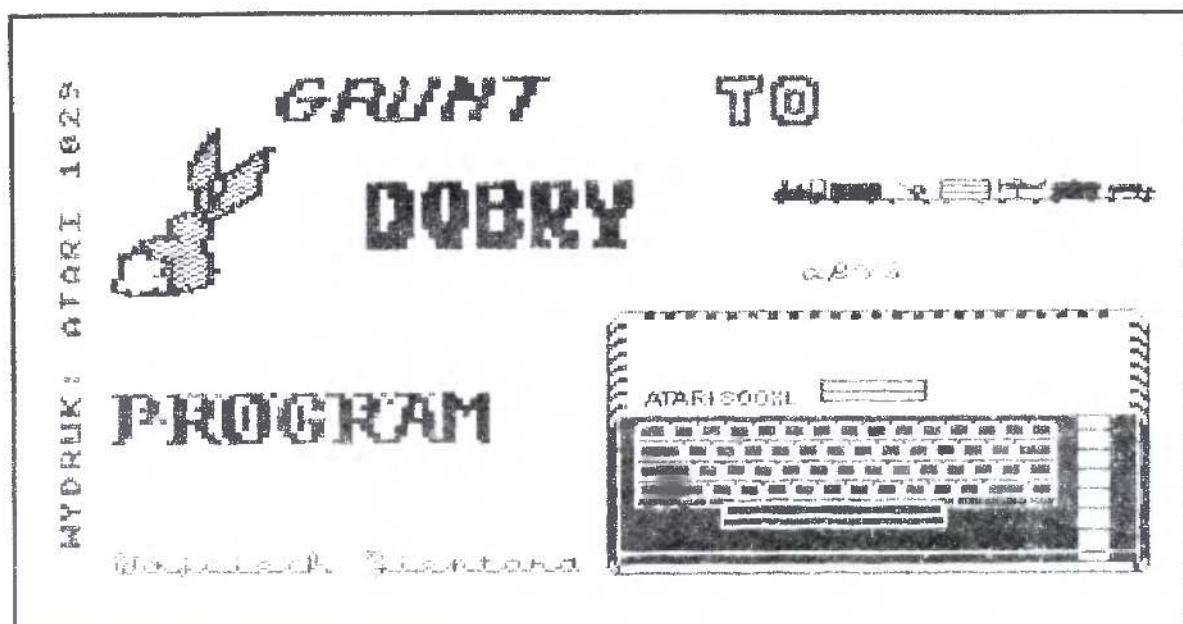
7684 bajty, a plik GRAPHICS 8 tylko 7680 bajtów. Podczas drukowania ostatnie cztery bajty „MicroPainter” są pomijane, można więc założyć identyczność tych standardów (przy drukowaniu).

„Rapid Graphics Converter” nie posiada jednak możliwości odczytania plików utworzonych przez „Print Shop”. Do tego celu należy użyć innego programu — „Print Shop Converter” (jest on opisany w innym miejscu naszego pisma).

Kolejną przeszkodą czyhającą na użytkowników Atari 1029 jest konieczność drukowania obrazów całoekranowych. Uniemożliwia to bezpośrednie drukowanie ikon „Typesettera” i „Print Shopa”. Oczywiście kolejny program — „Rubber Stamp” — pozwoli nam ominąć i to utrudnienie. Posiada on możliwość wczytania ikon „Typesettera” w dowolną ćwiartkę ekranu i wiele różnych funkcji redakcyjnych, które pozwalają je dowolnie przekształcać. Uzyskany obraz jest zapisywany w standardzie GRAPHICS 8 i nadaje się bezpośrednio do druku. Dodatkową funkcją „Rubber Stamp” jest możliwość zamiany ikon „Print Shopa” na ikony „Typesettera” (analogicznie jak „Print Shop Converter”).

Nie są to oczywiście wszystkie programy ułatwiające pracę z drukarką Atari 1029, lecz podany zestaw należy uznać za podstawowy. Posiadanie go pozwoli na znacznie efektywniejsze wykorzystanie drukarki. Nie oznacza to jednak, że będzie ono łatwiejsze. Z przedstawionego opisu jasno wynika, że uzyskanie ciekawych efektów wiąże się z kłopotem lub kilkunastokrotnym przekładaniem dysków i wczytywaniem różnych programów. Niestety jest to cena, jaką płaci się za posiadanie stosunkowo taniej drukarki.

Wojciech Zientara



# N W E W D O M O N A J P R O S T S Z Y

**Przedstawione w niniejszym opisie urządzenie trudno nazwać modemem, gdyż nawet najprostszy modem powinien cechować się pewną inteligencją. Z tego powodu lepszą nazwą dla przedstawionego rozwiązania będzie interfejs telefoniczny.**

Podczas konstruowania przyjęto następujące założenia — najprostszy schemat elektryczny, minimalizacja ilości użytych części. Dzięki takim założeniom interfejs ten będzie co prawda kolejnym, obok magnetofonu, bezmyślnym urządzeniem w domowym zestawie ATARI, ale za to prostym i tanim w budowie, czyli dostępnym dla szerokiego kręgu odbiorców.

Przedstawiony interfejs jest dołączony do złącza szeregowego, na którym można uzyskać sygnał dwutonowy, doskonale nadający się do przesyłania linią telefoniczną z dowolną szybkością.

W Polsce ze względu na jakość linii telefonicznych zalecaną szybkością transmisji jest 300 bodów, jednak przy połączeniach na małe odległości np. w obrębie dzielnicy lub miasta można tę prędkość zwiększyć do 600—900 bodów.

Opisane wyżej czynności w modemach fabrycznych wykonywane są automatycznie tzn. modem przekształca sygnały cyfrowe otrzymane z komputera w sygnały analogowe dwutonowe i wysyła te sygnały do linii telefonicznej z określoną szybkością. W modemach fabrycznych stosowane są trzy podstawowe prędkości transmisji 300, 600 lub 1200 bodów wybierane przez użytkownika lub (w prostych urządzeniach) ustalone przez wytwórcę. Poza opisanymi czynnościami modemy zapewniają dobrą współpracę komputerów dzięki odpowiednim sygnałom „porozumiewawczym”. Modemy do komputerów ATARI budowane są w oparciu o mikroprocesor jednoukładowy typu 8048 firmy INTEL. Program obsługi modemu zapisany jest w dwóch lub trzech częściach. Jedna część w pamięci procesora, druga w DOS-ie, ewentualna część trzecia w specjalnym programie obsługi modemu np. XETERM.COM, HOME-TERM.

W opisanym urządzeniu większość wymienionych czynności powierzona została komputerowi. Do obsługi interfejsu wykorzystano Kasetowy System Operacyjny opisany w 5 numerze „Bajtka”, gdyż pozwala on regulować szybkość transmisji oraz uodparnia komputer na



## BLOK ODBIORCZY

Jest to prosty wzmacniacz mikrofonowy zapewniający odpowiedni poziom napięcia na wejściu bloku przemiany.

## BLOK PRZEMIANY

Następuje tutaj wzmocnienie sygnałów z bloku odbiorczego oraz ich ograniczenie, a następnie sygnały „przechodzą” przez dwa filtry, w których wyodrębniane są sygnały akustyczne odpowiadające zerom i jedynek. Na wyjściu tego bloku znajduje się tranzystor zapewniający uzyskanie sygnałów o poziomie TTL, które wysyłane są do komputera. Dokładny opis interfejsu do magnetofonu znajduje się w KOMPUTERZE nr 9/1986.

## BLOK NADAWCZY

Stanowi go jednotranzystorowy wzmacniacz prądowy. Służy on do wzmocnienia sygnałów elektrycznych uzyskanych ze złącza szeregowego do poziomu zapewniającego wystawienie przetwornika elektroakustycznego (słuchawka telefoniczna W-66).

Układ interfejsu zasilany jest bezpośrednio z komputera, a jego praca sygnalizowana jest świeceniem diody elektroluminescencyjnej.

## URUCHOMIENIE UKŁADU

W interfejsie regulacji wymagają jedynie filtry w bloku przemiany oraz należy tak dobrać głośność, aby zapewnić dobrą jakość połączenia i jednocześnie nie zakłócać spokoju sąsiadom. W zmontowanym interfejsie należy połączyć punkt A z punktem B odcinkiem przewodu. Między masę i nóżkę 9 układu LM 324 należy włączyć woltomierz (o rezystancji wejściowej  $\geq 20$  kOhm/Volt). Następnie łączymy interfejs z komputerem (w złącze szeregowo komputera wkładamy wtyk interfejsu). Teraz włączamy zasilanie komputera i wprowadzamy z klawiatury instrukcję POKE 54018,52. Po naciśnięciu RETURN w interfejsie zaświeci się dioda elektroluminescencyjna. Wpisujemy instrukcję SOUND 0,5,10,4 i naciskamy RETURN. W głośniku powinniśmy usłyszeć wysoki dźwięk. Potencjometrem P1 należy ustawić maksimum wskazań woltomierza. Po zakończeniu regulacji odłączamy woltomierz od nóżki 9 i podłączamy go do punktu C. Wprowadzamy z klawiatury instrukcję SOUND 0,7,10,4 i ustawiamy potencjometrem P2 maksimum wskazań. Na tym kończą się czynności regulacyjne. Należy jeszcze, po wyłączeniu zasilania instrukcją POKE 54018,60 usunąć połączenie punktów A i B. Odpowiednią głośność ustalamy rezystorem R 27.

Jeszcze kilka słów o obudowie interfejsu. Powinna zapewnić zamocowanie mikrofonu i słuchawki na jednej powierzchni, najlepiej górnej, w taki sposób aby był dobry kontakt ze słuchawką telefoniczną, którą będziemy kładli na interfejsie.

## OBSŁUGA INTERFEJSU

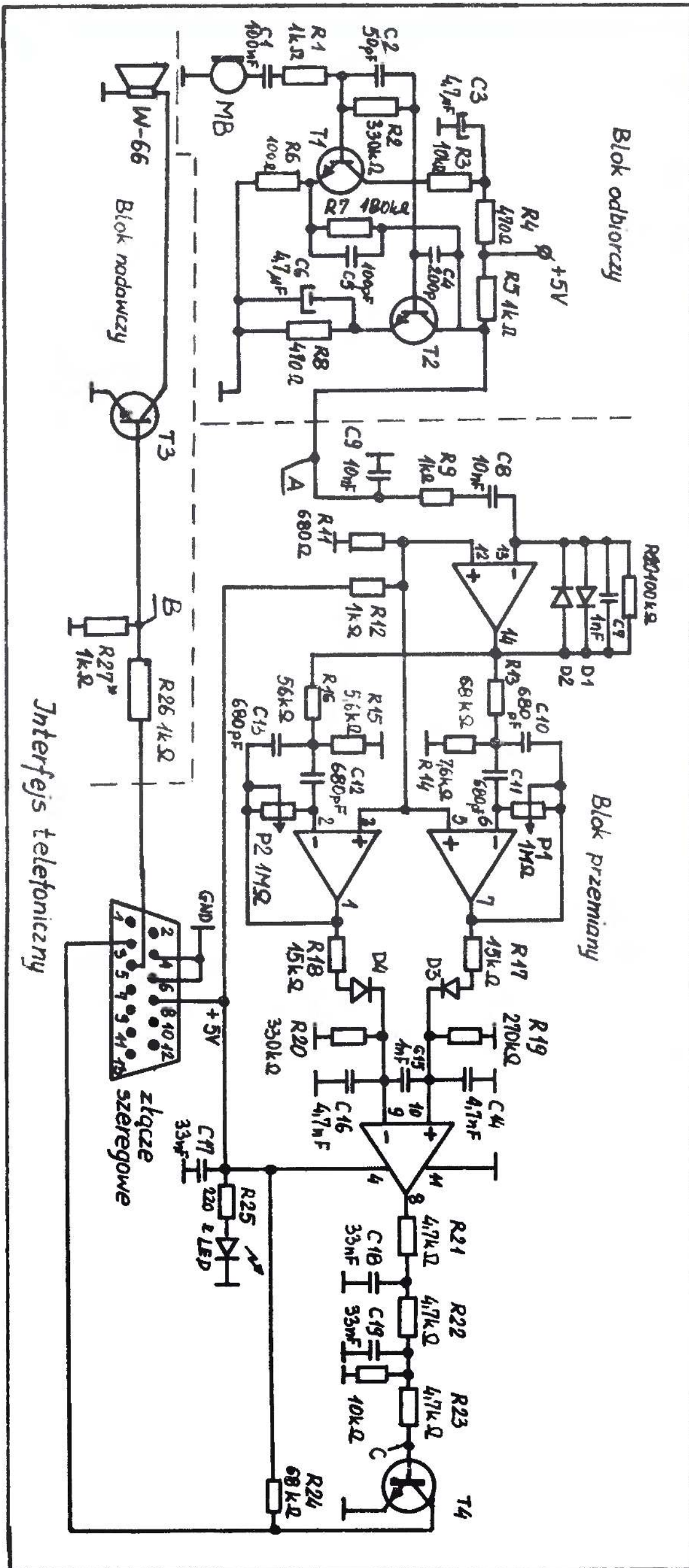
Wczytujemy z kasy KOS oraz program, który chcemy przesłać. Po uzyskaniu połączenia telefonicznego z kolegą lub koleżanką ustalamy kierunek transmisji tzn. kto „nada” i kto „odbiera”. Następnie kładziemy słuchawkę na interfejs (nasz rozmówca czyni to samo) i „uruchamiamy” komputery na transmisję w taki sam sposób jak przy obsłudze magnetofonu tzn. w komputerze „nadawczym” wpisujemy instrukcję SAVE „F:Nazwa”, a w komputerze „odbiorczym” wpisujemy LOAD „F:Nazwa”. Ważne jest, aby najpierw nacisnąć RETURN w „odbiorniku”, a następnie w „nadajniku”. Transmisję można prowadzić wykorzystując bloki transmisji szeregowo (np. OPEN #1,8,128, „F:Nazwa”:? # 1; DANE — w „nadajniku”; oraz OPEN #1,4,128, „F:Nazwa”: INPUT # 1, DANE — w „odbiorniku”).

**Na koniec jeszcze ważna uwaga: na korzystanie z sieci telefonicznej do przesyłania sygnałów cyfrowych („komputerowych”) wymagane jest specjalne zezwolenie Urzędu Telekomunikacji oraz w przypadku urządzeń budowanych samodzielnie badania techniczne. Bliższych informacji udzielają miejscowe Urzędy Poczto-Telekomunikacyjne.**

*Przemysław Strzelecki*

## WYKAZ CZĘŚCI:

1. Układ scalony LM 324	1 szt.
2. Tranzystor T1, 2,4 BC 147 itp.	3 szt.
3. Tranzystor T3 BC 211	1 szt.
4. Dioda BAP 794-795A	4 szt.
5. Dioda CQP 441 lub 442	1 szt.
6. Wtyk ATARI	1 szt.
7. Dowliny mikrofon dynamiczny	1 szt.
8. Słuchawka telefoniczna W-66	1 szt.
9. Potencjometr montażowy 1 M $\Omega$	2 szt.
10. Rezystor MLT 0,125 lub 0,25 W	
100 $\Omega$	1 szt.
220 $\Omega$	1 szt.
470 $\Omega$	2 szt.
680 $\Omega$	1 szt.
1 k $\Omega$	6 szt.
4,7 k $\Omega$	3 szt.
5,6 k $\Omega$	1 szt.
7,5 k $\Omega$	1 szt.
10 k $\Omega$	2 szt.
15 k $\Omega$	2 szt.
56 k $\Omega$	1 szt.
68 k $\Omega$	2 szt.
100 k $\Omega$	1 szt.
150 k $\Omega$	1 szt.
180 k $\Omega$	1 szt.
270 k $\Omega$	1 szt.
330 k $\Omega$	2 szt.
11. Kondensatory (dowolnego typu)	
50 pF	1 szt.
100 pF	1 szt.
200 pF	1 szt.
680 pF	4 szt.
4 nF	2 szt.
10 nF	2 szt.
33 nF	3 szt.
100 nF	1 szt.
4,7 $\mu$ F/10 V	2 szt.



błędy typu TIMEOUT występujące gdy komputer zbyt długo czeka na dane z magnetofonu (lub innego urządzenia). Jest to dość istotna cecha, gdyż w proponowanym urządzeniu nie ma żadnych sygnałów synchronizujących bądź blokujących TIMEOUT. Inne korzyści wynikające z zastosowania KOS-a będą widoczne po przeczytaniu wspomnianego artykułu w 5 numerze Bajtka.

Interfejs telefoniczny jest rozbudowanym o niezbędne elementy interfejsem magnetofonowym. Z tego powodu zrezygnowano z podawania schematu płytki drukowanej,

gdyż jej rysunek będzie zależał od tego czy urządzenie zostanie zbudowane od podstaw, czy też jako uzupełnienie posiadanego interfejsu. Całe urządzenie można podzielić na trzy bloki: odbiorczy — odbiór sygnału z linii telefonicznej; przemiany — zamiana sygnału analogowego na cyfrowy; nadawczy — przesyłanie sygnałów z komputera do telefonu. W proponowanym rozwiązaniu zastosowano akustyczne sprzężenie z linią telefoniczną, tzn. słuchawkę aparatu telefonicznego kładzie się na obudowie interfejsu w miejscu gdzie zainstalowane są głośnik i mikrofon.



# INSTRUKCJE

# 2.5



**Współpraca komputera ze stacją dysków jest możliwa poprzez specjalny program, tzw. Dyskowy System Operacyjny (DOS). W przypadku komputerów Atari DOS jest umieszczony na specjalnej dyskietce. Dzięki temu możliwe jest stosowanie różnych DOS-ów bez zmian sprzętowych. Najczęściej używany jest DOS 2.5, którego krótki opis zamieszczamy poniżej.**

Dyskowy System Operacyjny (Disk Operating System) 2.5 został opracowany w roku 1984 przez firmę Optimized System Software. Jest on rozszerzoną wersją DOS 2.0 tej samej firmy. Umożliwia zapisanie na dyskietce 1010 sektorów po 128 bajtów. Ponieważ DOS wykorzystuje 3 bajty w każdym sektorze dla własnych celów, to ostatecznie mamy pojemność  $1010 \cdot 125 = 126\,250$  bajtów czyli 123,3 KB. Pozostałe 30 z 1040 sektorów znajdujących się na dyskietce jest wykorzystywane przez DOS i jest niedostępne dla użytkownika. W sektorach tych przechowywane są informacje o plikach (programach i zbiorach danych) znajdujących się na dyskietce.

## PLIKI DOS 2.5

Dyskietka systemowa (Master Diskette) DOS 2.5 zawiera sześć plików. Są to:

**DOS.SYS**  
**DUP.SYS**  
**RAMDISK.COM**  
**SETUP.COM**  
**DISKFIX.COM**  
**COPY32.COM**

Plik **DOS.SYS** zawiera procedury komunikacji komputera ze stacją dysków i jest niezbędny do współpracy tych urządzeń. Po uruchomieniu komputera jest on automatycznie wczytywany do pamięci.

**DUP.SYS** (Disk Utility Package) to program umożliwiający użytkownikowi bezpośrednio korzy-

stanie z procedur DOS-u. Ta część DOS-u nie znajduje się stale w pamięci i musi być wczytywana z dysku oddzielnie.

**RAMDISK.COM** jest automatycznie wczytywanym i uruchamianym programem, który służy do zorganizowania w dodatkowej pamięci RAM komputera 130 XE (oraz 256 XT) wirtualnej (wyobrażonej) stacji dysków.

**SETUP.COM** ładuje się funkcją L z DOS-u. Służy on do zmiany konfiguracji systemu i tworzenia plików AUTORUN.SYS uruchamiających programy napisane w BASIC-u.

**DISKFIX.COM** również ładuje się funkcją L. Umożliwia on weryfikację plików zawartych na dyskietce i odzyskanie plików skasowanych.

**COPY32.COM** jest programem umożliwiającym odczytanie plików zapisanych przy użyciu DOS 3 i przeniesienie ich na dyskietkę w formacie DOS 2.5.

## LADOWANIE DOS-u

Przed opisem poszczególnych funkcji DOS 2.5 zajmiemy się jeszcze czynnościami wykonywanymi przez komputer po włączeniu zasilania. Zakładamy przy tym, że stacja dysków jest włączona i znajduje się w niej dyskietka systemowa. Włączenie zasilania komputera powoduje rozpoczęcie pracy przez procedurę inicjowania systemu. Ustawia ona wewnętrzne układy i rejestry komputera, a następnie próbuje odczytu z dysku. Jeżeli na dysku znajduje się plik DOS.SYS, to wczytuje go, w przeciwnym wypadku wyświetla raport „BOOT ERROR” i ponownie podejmuje próbę odczytu (aż do skutku). Gdy DOS.SYS został wczytany, przejmuje kontrolę nad dalszą pracą komputera. W komputerach 130 XE i 256 XT najpierw sprawdza obecność na dysku pliku RAMDISK.COM i jeśli go znajdzie, to wczytuje i uruchamia. W innych modelach ta czynność jest pomijana. Teraz poszukiwany jest na dysku plik o nazwie AUTORUN.SYS. Jeżeli plik o takiej nazwie znajduje się na dysku, to jest ładowany i DOS przekazuje mu dalsze sterowanie komputera. Gdy pliku AUTORUN.SYS nie ma na dysku, sprawdzana jest obecność w pamięci interpretera BASIC-a lub innego cartridge'a. Gdy jest (klawisz OPTION nie był wciśnięty przy włączaniu komputera), to sterowanie zostaje przekazane do niego. Jeżeli trzymaliśmy wciśnięty klawisz OPTION (cartridge odłączony), to wczytywany jest DUP.SYS i na ekranie pojawia się menu DOS-u 2.5. Gdy na dysku nie ma pliku DUP.SYS, komputer zawiesza się i nie pozostaje nam do zrobienia nic oprócz wyłączenia go.

Jeżeli uaktywniony został interpreter BASIC-a (pojawił się napis READY), to dostęp do menu DOS-u uzyskujemy przez wpisanie rozkazu DOS.

## FUNKCJE DOS 2.5

Wszystkie funkcje DOS-u są wywoływane przez naciśnięcie klawisza z literą odpowiadającą wybranej funkcji, a następnie klawisza RETURN. W przypadku naciśnięcia przed RETURN więcej niż jednego klawisza wyświetlony zostanie komunikat „PLEASE TYPE 1 LETTER” (proszę wpisać 1 literę). Naciśnięcie tylko RETURN powoduje ponowne wyświetlenie menu.

**A. DISK DIRECTORY** — odczyt spisu plików zawartych na dyskietce. Po wywołaniu funkcji poja-

wia się komunikat „DIRECTORY — SEARCH SPEC, LIST FILE?” (spis plików — specyfikacja wyszukiwania, plik wyjściowy?). Należy podać klucz, wg którego będą wyszukiwane pliki (np. D2:\*.\*.COM) i po przecinku nazwę urządzenia, na które ma być wyprowadzony spis plików. Naciśnięcie tylko RETURN powoduje przyjęcie standardowych odpowiedzi: D1:\*.\*.E: (spis wszystkich plików zawartych na dyskietce w stacji numer 1 i wyświetlenie ich na ekranie). Pominięcie w specyfikacji nazwy i numeru stacji (\*.\*) spowoduje również odczyt ze stacji numer 1.

**B. RUN CARTRIDGE** — przekaz sterowania do cartridge'a. Gdy nie ma żadnego cartridge'a i przy włączaniu komputera klawisz OPTION był wciśnięty otrzymujemy komunikat „NO CARTRIDGE” (brak cartridge'a).

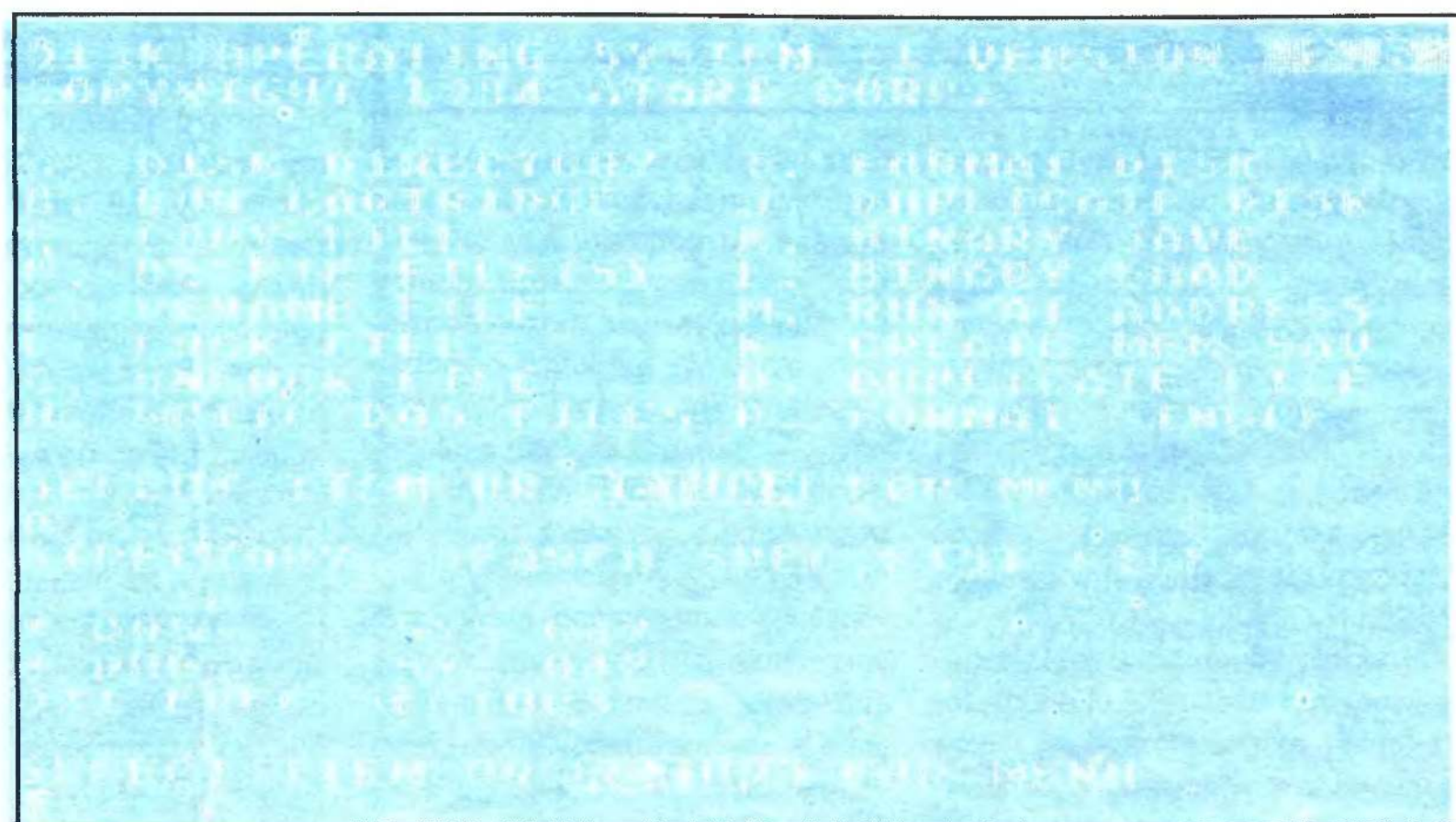
**C. COPY FILE** — kopiowanie plików. W odpowiedzi na pytanie „COPY — FROM, TO?” (kopiowanie — skąd, dokąd?) należy podać specyfikacje plików źródłowego i docelowego, np. D1:\*.\*.COM, D2:\*.\* (D1: można pominąć). Jako plik docelowy może być podana drukarka (P:), a jako źródłowy edytor (E:). Przy kopiowaniu w obrębie jednej dyskietki używanie znaków specjalnych (\* i ?) jest zabronione. Nie dopuszcza się również kopiowania plików DOS.SYS i DUP.SYS, a pozostałe pliki o nazwach z identyfikatorem. „SYS” mogą być kopiowane tylko przez podanie pełnej nazwy (bez znaków specjalnych). Dodatkową możliwością jest łączenie plików przy kopiowaniu w obrębie jednej dyskietki. W tym celu należy na końcu specyfikacji dodać „/A”, np. GRAF1.DAT, GRAF2.DAT/A (dodanie pliku GRAF1.DAT na końcu pliku GRAF2.DAT).

**D. DELETE FILE (S)** — skasowanie pliku/ów. Komunikat „DELETE FILE SPEC” (specyfikacja kasowanego pliku) jest żądaniem nazwy pliku. Pełna nazwa określa konkretny plik do skasowania, a użycie znaków specjalnych umożliwia skasowanie kilku plików jednocześnie. Przed skasowaniem każdego pliku komputer pyta „TYPE „Y” TO DELETE...” (wpisz „Y”, aby skasować...) i wyświetla pełną nazwę pliku. Potwierdzeniem żądania skasowania pliku jest kolejne naciśnięcie klawiszy Y i RETURN. Można uniknąć potwierdzania pisząc na końcu specyfikacji „/N”, np. \*.\*.COM/N spowoduje skasowanie wszystkich plików z identyfikatorem. „COM” zawartych na dysku w stacji numer 1.

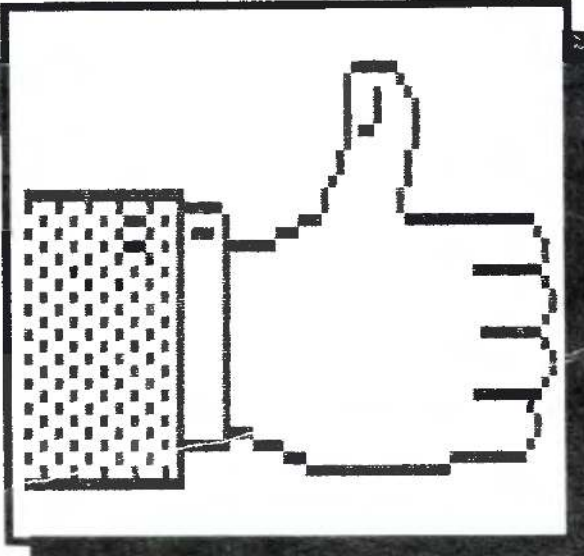
**E. RENAME FILE** — zmiana nazwy pliku. Nazwy — aktualną i nową — wpisujemy po komunikacie „RENAME — GIVE OLD NAME, NEW” (zmiana nazwy — podaj starą nazwę i nową). W nowej nazwie trzeba pominąć nazwę urządzenia, np. D2:STARY.BAS.NOWY.BAS. Dozwolone jest użycie znaków specjalnych do jednoczesnej zmiany nazw kilku plików.

**F. LOCK FILE** — zabezpieczenie pliku. Po wyświetleniu komunikatu „WHAT FILE TO LOCK” (który plik zabezpieczyć?) należy podać właściwą nazwę. Pliki zabezpieczone w ten sposób są oznaczone w spisie zawartości dysku (directory) gwiazdką przed nazwą pliku. Zabezpieczenie to nie chroni plików przed zniszczeniem podczas formatowania dyskietki.

**G. UNLOCK FILE** — odbezpieczenie pliku. Po wyświetleniu komunikatu „WHAT FILE TO UNLOCK?” (który plik odbezpieczyć?) należy podać nazwę pliku. Funkcja ta jest odwrotnością funkcji F.







## OCENA

**H. WRITE DOS FILES** — zapis na dyskietce plików DOS.SYS i DUP.SYS. Funkcja ta pozwala na zapisanie na dyskietce plików DOS-u, których skopiowanie funkcjami C i O jest niemożliwe. Ponieważ pliki te są przepisywane z pamięci komputera, to wprowadzone zmiany (np. przez POKE) są zapamiętywane na dysku. Umożliwia to uzyskanie zmienionych wersji DOS-u. Na wyświetlane komunikaty „DRIVE TO WRITE DOS FILES TO?” (stacja do zapisania plików DOS?) i „TYPE „Y” TO WRITE DOS TO DRIVE ...” (wpisz „Y”, aby zapisać DOS w stacji...) należy wpisać numer stacji i potwierdzić żądanie wpisując Y.

**I. FORMAT DISK** — formatowanie dysku. Każda dyskietka przed użyciem musi zostać sformatowana. Polega to na oznaczeniu na dysku przez DOS granic sektorów, wypełnieniu ich zerami i zapisaniu w niektórych sektorach informacji niezbędnych dla systemu. Formatowanie dyskietki kasuje bezpowrotnie całą zawartą na niej informację. Na wyświetlane komunikaty „WHICH DRIVE TO FORMAT?” (w której stacji formatować?) i „TYPE „Y” TO FORMAT DISK ...” (wpisz „Y”, aby zaformatować dysk ...) należy wpisać numer stacji i potwierdzić żądanie wpisując Y.

**J. DUPLICATE DISK** — skopiowanie całej dyskietki. Komunikat „DUP DISK — SOURCE, DEST DRIVES?” (kopiowanie dysku — stacja źródłowa i docelowa?) jest żądaniem podania numerów stacji, w których będą umieszczone dyskietki, np. D1, D2 lub tylko 1,2. Naciśnięcie RETURN po kolejnym komunikacie „INSERT BOTH DISKS, TYPE RETURN” (włóż oba dyski, naciśnij RETURN) rozpoczyna operację kopiowania, już bez udziału użytkownika. Jeżeli posiadamy jedną stację, to wpisujemy numery stacji D1, D1 lub 1,1. W takim przypadku konieczne będzie przekładanie dysków podczas kopiowania na polecenia: „INSERT SOURCE DISK, TYPE RETURN” (włóż dysk źródłowy, naciśnij RETURN) oraz „INSERT DESTINATION DISK, TYPE RETURN” (włóż dysk docelowy, naciśnij RETURN). UWAGA: dyskietka docelowa jest formatowana przed kopiowaniem.

**K. BINARY SAVE** — zapisanie na dyskietkę zawartości obszaru pamięci komputera jako pliku binarnego. Opis tej funkcji pominiemy, gdyż jest ona przeznaczona dla zaawansowanych użytkowników.

**L. BINARY LOAD** — ładowanie pliku binarnego z dyskietki do pamięci komputera i uruchomienie, jeśli w pliku został podany adres uruchomienia. Funkcja przeznaczona dla zaawansowanych użytkowników.

**M. RUN AT ADDRESS** — uruchomienie programu do podanego adresu (przekazanie sterowania). Adres należy podać w postaci liczby szesnastkowej po wyświetleniu pytania „RUN FROM WHAT ADDRESS?” (od jakiego adresu uruchomić?).

**N. CREATE MEM.SAV** — utworzenie na dysku pliku o nazwie MEM.SAV. W pliku tym zapisywana jest zawartość pamięci komputera podczas wywołania DOS-u. Zabezpiecza to przed utratą wpisywanego programu, jeśli wywołamy DOS bez uprzedniego zapisania tego programu na dysku. Żądanie utworzenia pliku MEM.SAV należy potwierdzić wpisując Y w odpowiedzi na komunikat „TYPE „Y” TO CREATE MEM.SAV” (wpisz „Y”, aby utworzyć MEM.SAV).

**O. DUPLICATE FILE** — kopiowanie pliku. Funkcja ta umożliwia przenoszenie plików z jednej dyskietki na drugą przy użyciu jednej stacji dysków (funkcja COPY kopiuje plik w obrębie jednej dyskietki). Po wyświetleniu komunikatu „NAME OF FILE TO MOVE?” (nazwa pliku do przeniesienia?) wpisujemy nazwę kopiowanego pliku (można użyć znaków specjalnych), a następnie przekładamy odpowiednie dyskietki stosownie do komunikatów „INSERT SOURCE DISK, TYPE RETURN” (włóż dysk źródłowy, naciśnij RETURN) oraz „INSERT DESTINATION DISK, TYPE RETURN” (włóż dysk docelowy, naciśnij RETURN). Gdy w nazwie pliku do kopiowania został użyty znak specjalny, to przy odczytywaniu pliku z dyskietki źródłowej wyświetlana jest pełna nazwa pliku poprzedzona słowem „COPYING” (kopiowanie).

**P. FORMAT SINGLE** — formatowanie dyskietki w pojedynczej gęstości (format DOS 2.0). Funkcja identyczna z funkcją I, lecz dyskietka jest formatowana w standardzie DOS 2.0 (720 sektorów, w tym 707 dostępnych dla użytkownika).

**Brak na naszym rynku podstawowych podręczników dla początkujących programistów powoduje, że wielu z nich pisze programy nie mając możliwości oceny ich poprawności. Spróbujemy zatem zapełnić tę lukę. Uprzedzamy jednak, że nasze oceny będą bezlitosne. Prosimy traktować je jako przyjacielskie porady i nie zniechęcać się. Każdy z nas zaczął podobnie, lecz nie miał nam kto wytknąć błędów. Mamy nadzieję, że dzięki nam unikniecie wielu przykrych rozczarowań. A oto pierwszy list:**

*Mam 12 lat. Posiadam komputer Atari 800 XL, na którym staram się układać programy. Ułożyłem między innymi program „ZEGAR”, który Wam przesyłam, prosząc o ocenę. Zegar każdą minioną godzinę sygnalizuje dźwiękiem.*

*Wojciech Zachariasz  
ul. Ignacego Fika 18/13  
31-214 Kraków*

```

1 GRAPHICS 0
2 POKE 710,0
3 ? "*****"
4 ? "**** ZEGAR ****"
5 ? "**** W. ZACHARIASZ ****"
6 ? "**** KRAKOW - 1987 ****"
7 ? "*****"
8 ? "1? 1? "PODAJ CZAS:"
9 ? "1? 1? 1? 1? 1?"
10 ? "GODZINA"::INPUT A
11 ? "MINUTA"::INPUT B
12 ? "SEKUNDA"::INPUT C
13 H=A
14 Y=B
15 X=C-1
16 IF Y=59 AND X=59 THEN 110
17 IF X=59 THEN X=-1:Y=Y+1
18 IF H=24 THEN 200
19 GOTO 150
20 Y=Y+1
21 GOTO 150
22 H=H+1:Y=0:X=1
23 FOR I=15 TO 0 STEP -0.15
24 SOUND 3.255,10,I
25 SOUND 0,0,0,0
26 SOUND 0,255,10,I
27 NEXT I
28 SOUND 0,0,0,0
29 X=X+1
30 GRAPHICS 2:POKE 710,0:POSITION 5,6
31 #6:H:"":Y:"":X
32 FOR G=1 TO 200 STEP 50:SOUND 0,6,6
33 .10:NEXT G:SOUND 0,0,0,0:GOTO 180
34 FOR I=1 TO 285:NEXT I
35 GOTO 60
36 H=0:Y=0:X=2
37 GOTO 60

```

Nadesłany program działa poprawnie i co najważniejsze wykonuje to, co autor zamierzył. Stanowi jednak niemal idealny przykład, jakie spustoszenia czyni rozpoczynanie nauki programowania od BASIC-a. Wszędzie, gdzie tylko można napotykamy instrukcję GOTO. Aż się prosi, aby sygnał dźwiękowy umieścić w podprogramie. Oprócz tego wielu skoków można uniknąć wpisując po kilka instrukcji w jednej linii.

Kolejny błąd występuje przy przekraczaniu godziny 24. Po godzinie „23:59:59” widzimy

„24:0:2” i dopiero po sekundzie „0:0:3”. Aby tego uniknąć, sprawdzenie warunku H=24 powinno nastąpić przed wyświetleniem aktualnego czasu.

W tak krótkim programie drobiazgiem jest użycie dodatkowej zmiennej G w pętli, lecz w dużych programach może zabraknąć zmiennych (Atari BASIC dopuszcza użycie tylko 128 zmiennych). Ponieważ pętle występujące w programie nie są zagnieżdżone (nie występują jedna w drugiej), to można we wszystkich używać tej samej zmiennej sterującej.

Po tych poprawkach główna część programu (od linii 60) będzie wyglądała następująco:

```

60 IF Y=59 AND X=59 THEN GOSUB 150:GOT
0 80
70 IF X=59 THEN X=-1:Y=Y+1
80 X=X+1
90 GRAPHICS 2:POKE 710,0:POSITION 5,6:
? #6:H:"":Y:"":X
100 FOR I=1 TO 200 STEP 50:SOUND 0,1,6
.10:NEXT I:SOUND 0,0,0,0
110 FOR I=1 TO 285:NEXT I
120 GOTO 60
130 H=H+1:Y=0:X=1:IF H=24 THEN H=0:X=2
140 FOR I=15 TO 0 STEP -0.15
150 SOUND 3.255,10,I
160 SOUND 0,0,0,0
170 SOUND 0,255,10,I
180 NEXT I
190 GOTO 60
200 H=0:Y=0:X=2
210 GOTO 60
220 RETURN

```

Teraz przejdźmy do spraw ważniejszych — do błędów ideowych. Program działa, ale jak i po co? Oczywiście, jeśli ktoś lubi, to może używać komputera jako zegarka. Tylko, że jest to bardzo drogi zegar. Oprócz pomiaru czasu nie można zrobić już nic więcej. Tego typu pomocnicze programy powinny być realizowane w przerwach, aby nie zajmować efektywnego czasu pracy mikroprocesora. Zegar? Tak, ale w taki sposób, aby komputer mógł zrobić coś jeszcze. Tak więc pierwszym pytaniem, jakie powinien zadać sobie każdy programista, musi być: jaki uzyskamy efekt i jakim kosztem? Drugi problem jest innej natury. Autor zadał sobie wiele trudu, aby obliczyć czas trwania pętli dla uzyskania dokładnego wyniku. Efekt jest niestety mizerny. A przecież Atari ma systemowy zegar czasu rzeczywistego w komórkach 18—20. Wykorzystanie go do odmierzenia upływu czasu znacznie zwiększyłoby dokładność zegara. Uprościłoby to również program. Drugie pytanie brzmi więc: jak najprościej zrealizować postawione zadanie na konkretnym komputerze?

Chcę na koniec dodać, że bardzo mnie cieszą nawet takie nieśmiałe próby programowania w wykonaniu bardzo młodych autorów. Szczególnie, gdy widzę, jak „stare konie” łamią joysticki w tempie dwóch na miesiąc i myślą, że jest to jedyna rzecz, do jakiej można wykorzystać komputer.



# TURBO — COPY —

**Wychodząc naprzeciw tym użytkownikom Atari, którzy posiadają wyłącznie magnetofon, zamieszczamy program umożliwiający przyspieszenie transmisji z magnetofonu.**

Zaletą programu jest stosunkowo duża pojemność (ponad 47 KB), która powinna wystarczyć do przekopiowania większości programów. Po wpisaniu i uruchomieniu programu (oraz usunięciu ewentualnych błędów wpisywania) słyszymy sygnał dźwiękowy. Należy teraz ustawić taśmę w magnetofonie, wcisnąć klawisze PLAY i RECORD (w magnetofonie) oraz klawisz RETURN. Na taśmie zostanie zapisana wersja ostateczna programu TURBO-COPY, którą będziemy wczytywać przez włączenie komputera z wciśniętymi klawiszami START i OPTION. Program w BASIC-u (radzimy zachować go na wszelki wypadek) zawiera kontroler błędów, który wykrywa pomyłki popełnione przy przepisywaniu linii DATA, ważna jest przy tym numeracja linii programu i nie należy jej zmieniać. Aby przerobić program na kopiowanie ze standardową szybkością transmisji należy zmienić linie 1410 i 1420 następująco:

```
1410 DATA 15,169,0,141,4,325
```

```
1420 DATA 210,169,6,141,6,532
```

Do korzystania z programu wystarczają informacje zawarte na ekranie po jego uruchomieniu. Program jest zabezpieczony przed skasowaniem przez RESET.

Kilka uwag dla osób zainteresowanych działaniem programu. Na początku znajduje się pętla zapisująca dane do pamięci. Znajdują się tam instrukcje kontroli błędów i wskazania, w której linii DATA został popełniony. Po uruchomieniu i sprawdzeniu poprawności linii DATA następuje otwarcie kanału (linie 100—240) i zapis w postaci pliku samouruchamiającego na kasety.

Właściwy program kopiujący zaczyna się od linii 260. Pierwsze sześć bajtów to nagłówek (header) programu. Po uruchomieniu najpierw zatrzymuje się silnik magnetofonu, a następnie sprawdzana jest wielkość dostępnej pamięci. Jeżeli jest włączony interpreter BASIC-a (nie był wciśnięty klawisz OPTION), to program przekazuje sterowanie do BASIC-a. Jeśli BASIC jest odłączony, to program zostaje uodporniony na RESET przez wykonanie następujących instrukcji:

```
LDA #$AB
STA $02
LDA #$04
STA $03
LDA #$00
STA $0244
LDA #$02
STA $09
```

sany nowy adres programu ANTIC-a (wynosi on 1504) i wektor przerwania VBLK ustawiany jest na adres 1742, gdzie znajduje się procedura obsługująca TURBO — zmiana wartości w generatorach synchronizujących, transmisję oraz sprawdzenie komórki 649, w której znajdują się informacje o rodzaju wykonywanej operacji (zapis — 128 lub odczyt — 0). Kolejną czynnością jest sprawdzenie klawiszy konsoli i zależnie od ich stanu przejście do operacji. Pomijam szczegółowy opis dalszych czynności, gdyż osoby znające system operacyjny łatwo je zrozumieją, a innym nic to nie powie.

*Dariusz Modrinić*

**UWAGA: Nie należy wpisywać liter wydrukowanych przed numerem linii programu. NIE SA one częścią linii, lecz kodem kontrolnym „Edytora BASIC-a” (zob. str. 3).**

```
UW 10 REM *** TURBO-COPY ***
JL 15 REM *** DARIUSZ MODRINIC ***
AN 20 REM *** KRAKOW 1987 ***
HR 25 A=100:B=0:C=0
YF 30 ? CHR$(125);"PROSZE CZEKAC..."
MR 35 FOR P=32768 TO 33462
VJ 40 READ W:B=B+W:C=C+1:POKE P,W
AD 45 IF C=5 THEN C=0:A=A+10:GOSUB 75
LE 50 NEXT P
DV 55 ? CHR$(125);"ZAPIS WERSJI BOOT"
YS 60 ? "WCISNIJ RECORD I PLAY"
TU 65 ? "A NASTEPNIE RETURN"
FH 70 X=USR(32768):END
DR 75 READ S
VN 80 IF S=B THEN B=0:RETURN
EJ 85 ? CHR$(253);? "BLAD W DANYCH";
VE 90 ? " W LINII ";A-10:LIST A-10:END
RU 100 DATA 162,16,169,3,157,507
LR 110 DATA 66,3,169,8,157,403
SJ 120 DATA 74,3,169,128,157,531
XG 130 DATA 75,3,169,72,157,476
XF 140 DATA 68,3,169,128,157,525
PQ 150 DATA 69,3,32,86,228,418
NM 160 DATA 48,36,169,11,157,421
AV 170 DATA 66,3,169,79,157,474
XN 180 DATA 68,3,169,128,157,525
ZD 190 DATA 69,3,169,128,157,526
FY 200 DATA 72,3,169,2,157,403
HZ 210 DATA 73,3,32,86,228,422
UJ 220 DATA 48,8,169,12,157,394
LD 230 DATA 66,3,32,86,228,415
EI 240 DATA 104,96,67,58,158,483
BI 250 DATA 0,0,0,0,0,0
JL 260 DATA 5,160,4,166,4,339
CL 270 DATA 76,247,6,234,234,797
YS 280 DATA 169,171,133,2,169,644
EV 290 DATA 4,133,3,169,0,309
FB 300 DATA 141,68,2,141,198,550
FJ 310 DATA 2,169,2,133,9,315
RM 320 DATA 162,0,169,0,157,488
IR 330 DATA 128,4,224,31,240,627
ME 340 DATA 4,232,76,193,4,509
RO 350 DATA 169,224,141,48,2,584
LT 360 DATA 169,5,141,49,2,366
HY 370 DATA 169,206,141,34,2,552
EI 380 DATA 169,6,141,35,2,353
LB 390 DATA 173,31,208,201,6,619
IF 400 DATA 240,61,201,5,240,747
QV 410 DATA 54,201,3,240,3,501
RD 420 DATA 76,226,4,169,52,527
DN 430 DATA 141,2,211,169,10,533
NS 440 DATA 141,125,6,173,31,476
```

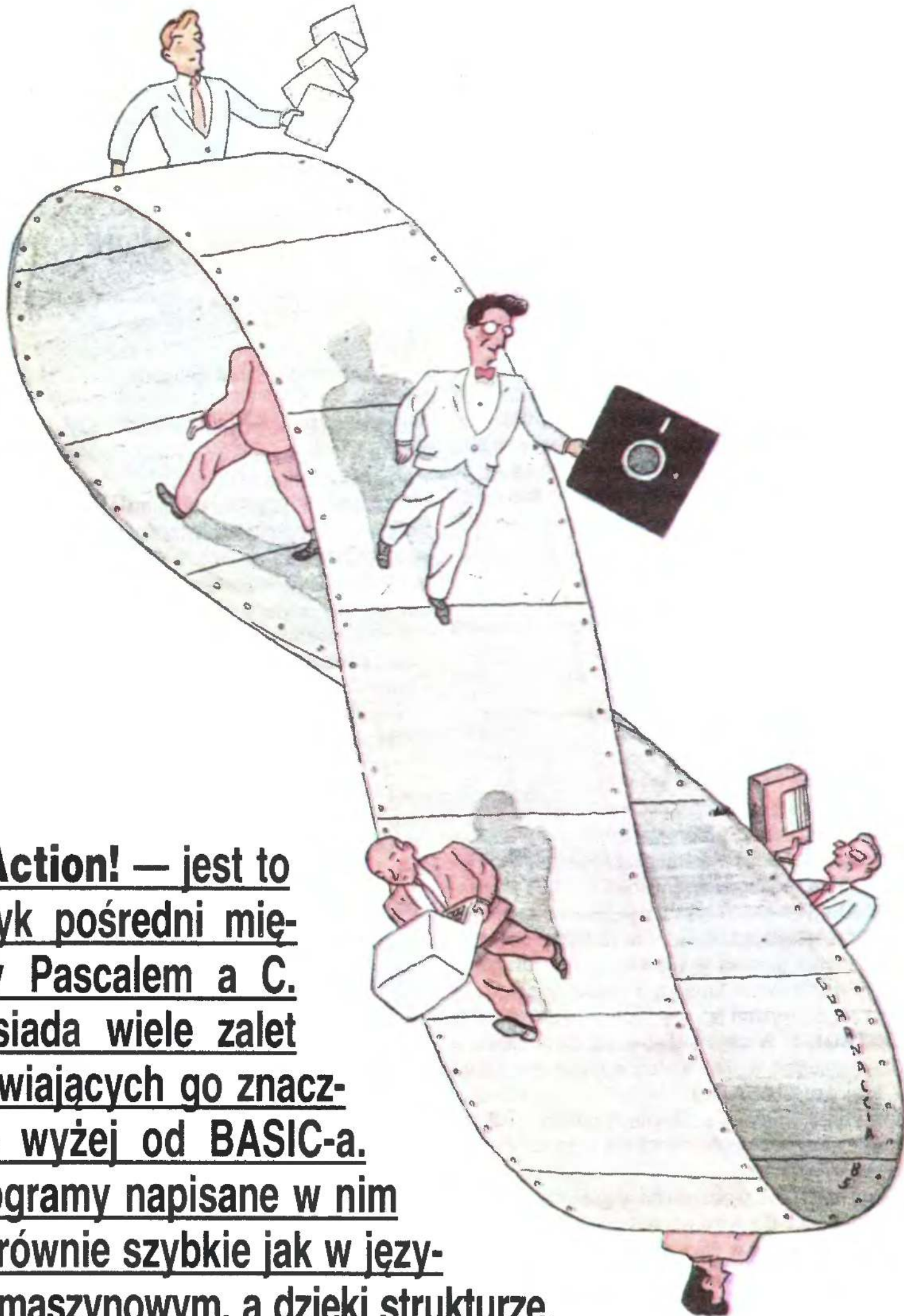
```
PT 450 DATA 208,201,6,240,33,688
MH 460 DATA 201,5,240,26,173,645
DF 470 DATA 220,2,201,17,240,680
NB 480 DATA 3,76,254,4,32,369
UH 490 DATA 191,6,169,60,141,567
ZD 500 DATA 2,211,169,0,141,523
IC 510 DATA 220,2,76,226,4,528
PA 520 DATA 76,135,5,169,10,395
ES 530 DATA 141,252,2,141,84,620
LZ 540 DATA 6,162,16,169,3,356
WP 550 DATA 157,66,3,169,4,399
SE 560 DATA 157,74,3,169,128,531
ZO 570 DATA 157,75,3,169,243,647
LM 580 DATA 157,68,3,169,6,403
RJ 590 DATA 157,69,3,32,86,347
CW 600 DATA 228,169,7,157,66,627
QD 610 DATA 3,169,8,157,68,405
QZ 620 DATA 3,169,7,157,69,405
ZF 630 DATA 3,169,128,157,72,529
EN 640 DATA 3,169,184,157,73,586
VH 650 DATA 3,32,86,228,169,518
DE 660 DATA 12,157,66,3,32,270
BD 670 DATA 86,228,173,88,3,578
PM 680 DATA 141,222,5,173,89,630
IM 690 DATA 3,141,223,5,169,541
JJ 700 DATA 255,141,252,2,32,682
GE 710 DATA 191,6,76,226,4,503
AG 720 DATA 169,10,141,104,6,430
KK 730 DATA 141,252,2,162,16,573
UD 740 DATA 169,3,157,66,3,398
IE 750 DATA 169,8,157,74,3,411
SH 760 DATA 169,128,157,75,3,532
SR 770 DATA 169,243,157,68,3,640
MS 780 DATA 169,6,157,69,3,404
SP 790 DATA 32,86,228,169,11,526
LP 800 DATA 157,66,3,169,8,403
MZ 810 DATA 157,68,3,169,7,404
OR 820 DATA 157,69,3,173,222,624
CG 830 DATA 5,157,72,3,173,410
GQ 840 DATA 223,5,157,73,3,461
UK 850 DATA 32,86,228,169,12,527
MK 860 DATA 157,66,3,32,86,344
TX 870 DATA 228,169,255,141,252,1045
JF 880 DATA 2,32,191,6,76,307
ST 890 DATA 226,4,0,0,112,342
SO 900 DATA 112,112,71,11,6,312
CC 910 DATA 112,66,31,6,112,327
UY 920 DATA 66,0,4,2,2,74
FK 930 DATA 2,112,112,70,71,367
JJ 940 DATA 6,6,6,6,112,136
LS 950 DATA 112,112,66,151,6,447
SY 960 DATA 112,112,112,65,224,625
HN 970 DATA 5,0,0,0,0,5
BV 980 DATA 0,0,0,0,0,0
DQ 990 DATA 0,0,116,117,114,347
IH 1000 DATA 98,111,77,99,111,496
KG 1010 DATA 112,121,0,0,0,233
PJ 1020 DATA 0,0,0,0,0,0
HT 1030 DATA 0,0,34,57,0,91
AJ 1040 DATA 36,33,50,41,53,213
CL 1050 DATA 51,58,0,45,47,201
CJ 1060 DATA 36,50,41,46,41,214
YM 1070 DATA 35,0,43,50,33,161
FK 1080 DATA 43,47,55,0,7,152
BR 1090 DATA 24,23,0,0,0,47
LN 1100 DATA 0,0,115,116,97,328
AL 1110 DATA 114,116,77,111,100,518
QC 1120 DATA 99,122,121,116,0,458
PO 1130 DATA 0,0,0,0,0,0
YT 1140 DATA 0,0,115,101,108,324
CW 1150 DATA 101,99,116,77,122,515
NO 1160 DATA 97,112,105,115,0,429
QA 1170 DATA 0,0,0,0,0,0
DV 1180 DATA 0,0,111,112,116,339
FK 1190 DATA 105,111,110,77,115,518
KR 1200 DATA 105,108,110,105,107,535
PK 1210 DATA 0,0,0,0,0,0
VD 1220 DATA 0,0,104,101,108,313
YK 1230 DATA 112,77,115,116,111,531
MB 1240 DATA 112,0,115,105,108,440
RF 1250 DATA 110,105,107,97,0,419
FZ 1260 DATA 0,0,0,0,0,0
OJ 1270 DATA 0,0,0,48,50,98
BE 1280 DATA 33,55,33,0,33,154
HN 1290 DATA 53,52,47,50,51,253
NZ 1300 DATA 43,41,37,0,58,179
IY 1310 DATA 33,51,52,50,58,244
SD 1320 DATA 37,58,47,46,37,225
PS 1330 DATA 0,0,0,0,0,0
TC 1340 DATA 0,0,169,0,141,310
DA 1350 DATA 84,6,141,104,6,341
QO 1360 DATA 141,125,6,141,220,633
QD 1370 DATA 2,96,173,247,6,524
HS 1380 DATA 141,196,2,141,22,502
OR 1390 DATA 208,238,247,6,173,872
BB 1400 DATA 137,2,201,0,240,580
XB 1410 DATA 15,169,67,141,4,396
GO 1420 DATA 210,169,4,141,6,530
SK 1430 DATA 210,169,96,141,0,616
BA 1440 DATA 3,76,226,192,67,564
UV 1450 DATA 58,158,0,169,60,445
AG 1460 DATA 141,2,211,165,89,608
AL 1470 DATA 201,156,240,3,76,676
BZ 1480 DATA 171,4,96,0,0,271
```

co powoduje po RESET skok pod adres 171+256\*4=1195. W tym czasie jest również ustalany kolor ekranu i tła. Następnie do rejestru DLPTX zostaje wpi-



# Action!

## NOWY JĘZYK ATARI



**Action! — jest to język pośredni między Pascal'em a C. Posiada wiele zalet stawiających go znacznie wyżej od BASIC-a. Programy napisane w nim są równie szybkie jak w języku maszynowym, a dzięki strukturze, edytorowi i bogatej bibliotece procedur programowanie jest znacznie łatwiejsze. Kompilator Action! jest dostępny na wszystkich trzech nośnikach stosowanych w Atari: na kasecie, na dysku i na cartridge'u. Oprócz tego w serwisie Atari można wbudować Action! na stałe do komputera.**

System **Action!** składa się z edytora i kompilatora oraz kontrolującego pracę całości monitora. Program źródłowy piszemy przy pomocy edytora, a następnie kompilujemy. Możliwe jest wykonanie napisanego programu zarówno po skompilowaniu, jak i w wersji źródłowej. W drugim przypadku monitor wczytuje program źródłowy, kompiluje go, a następnie wykonuje. Wadą dostępnego w Polsce systemu **Action!** jest konieczność jego wczytania przed uruchomieniem programu (również skompilowanego). Wynika to z faktu, że kompilator nie dołącza do programu wynikowego biblioteki procedur. Można oczywiście nabyć i taki kompilator, lecz jedynie u producenta.

Programy w **Action!** budowane są ze składników, z których każdy zawiera grupę odpowiednich instrukcji wykonujących jakąś operację. Dzięki temu można łatwo dołączać do pisanego programu wcześniej napisane bloki. Jedynym wymaganiem takiego strukturalnego podejścia jest to, że program musi być złożony z prawidłowych składników. Składnikami tymi są w **Action!** procedury i funkcje. Zwykle program zawiera ich wiele, lecz wymagany jest co najmniej jeden. W programie zawierającym kilka procedur wykonywanie rozpoczyna się od ostatniej, tak więc powinna ona sterować całym programem.

Zanim zaczniemy naukę programowania w **Action!** musimy zapoznać się z działaniem poszczególnych elementów systemu.

### EDYTOR

Edytor służy do tworzenia nowych programów i redagowania starych. Edytor **Action!** jest jednym z najlepiej rozwiązanych edytorów programowych. Można go używać również do innych celów (programowanie w innych językach, redagowanie tekstów itp.).

Po uruchomieniu systemu w dolnej części ekranu pojawia się jasna linia z napisem „ACTION! (c)1983 ACS”. Jest to linia komunikatów, w której wyświetlane są raporty błędów, inne informacje i wprowadzane komendy edytora. Gdy używasz dwóch okien edytora, linia komunikatów umieszczona jest między oknami.

Edytor **Action!** jest zaprogramowany w sposób umożliwiający jak najłatwiejsze pisanie programów. Pozwala on na wpisanie linii o długości do 240 znaków, pomimo, że w oknie edytora widoczne jest tylko 38 znaków. Dzięki temu można pisać programy bardziej przejrzyste. Osiągnięcie maksymalnej dopuszczalnej długości linii edytor sygnalizuje dźwiękiem buczenia. Jeżeli wpisana linia jest dłuższa niż szerokość okna, to znak na krawędzi okna (lewej lub prawej) jest wyświetlany w negatywie (inverse video), aby uwidocznić ten fakt.

Tekst programu wpisuje się do edytora bez żadnych specjalnych poleceń — jak przy użyciu maszyny do pisania. Aby wpisać znak kontrolny należy najpierw nacisnąć klawisz ESC. Mówi to edytorowi, że następny znak ma być interpretowany jako



# PROGRAMOWANIE

tekst, a nie jako polecenie dla edytora. Poleceniami edytora są niektóre kombinacje klawiszy naciskanych równocześnie, np. **SHIFT-CLEAR**.

## TRYBY EDYTORA

Edytor może pracować w dwóch trybach: wstawiania i zastępowania. W trybie zastępowania wprowadzany tekst zastępuje tekst już istniejący, znak po znaku. W trybie wstawiania nowy tekst jest wprowadzany w środek starego bez jego kasowania. Po uruchomieniu edytor pracuje w trybie zastępowania. Do przełączania trybów pracy edytora służy polecenie **SHIFT-CONTROL-I**.

## OBSŁUGA EDYTORA

Aby usunąć tekst z edytora należy użyć polecenia **SHIFT-CLEAR**. Kasuje on nie tylko tekst widoczny na ekranie, lecz całą zawartość edytora. Gdy używane są dwa okna edytora, to polecenie dotyczy tylko tego okna, w którym aktualnie znajduje się kursor. Aby uniknąć omyłkowego zniszczenia redagowanego programu, edytor pyta: „Clear?”. Należy odpowiedzieć „Y” (tak) lub „N” (nie). Jeśli w redagowanym tekście dokonano zmian i nie został on zapisany, edytor pyta ponownie: „Not saved-Delete?” (Nie zapisane, Usunąć?), aby upewnić się, że nie chcesz zapisać poprawionej wersji.

Istnieje tylko jeden sposób opuszczenia edytora (oczywiście oprócz wyłączenia komputera) — **SHIFT-CONTROL-M**. Polecenie to powoduje przekazanie kontroli systemu do monitora, skąd można wywołać inne elementy systemu lub przejść do DOS-u.

Program napisany w edytorze może być zapisany na dowolnym urządzeniu zewnętrznym. Umożliwia to polecenie **SHIFT-CONTROL-W**. Po naciśnięciu tej kombinacji klawiszy pojawia się napis „Write?”. Przy zapisywaniu pliku na dysku należy podać pełną nazwę pliku — D: nazwa.ext. Przy zapisie na magnetofonie lub drukarce wystarczy nazwa urządzenia — odpowiednio C: lub P:

Analogicznie odczytu zapisanego pliku dokonuje się poleceniem **SHIFT-CONTROL-R**, po którym komputer pyta „Read?”. Trzeba teraz podać nazwę pliku w sposób wyżej opisany. Dodatkowo polecenie to umożliwia odczyt spisu zawartości dyskietki (directory). Następuje to po podaniu jako nazwy pliku tekstu ?1:\*.\*. Gdy chcemy odczytać directory z innej stacji dysków, to wystarczy cyfrę 1 zastąpić numerem tej stacji.

## RUCH KURSORA

Poniższe polecenia umożliwiają poruszanie kursora w całym obszarze edytora **Action!**:

<b>CONTROL-←</b>	przesunięcie kursora o jeden znak w lewo;
<b>CONTROL-→</b>	przesunięcie kursora o jeden znak w prawo;
<b>CONTROL-↑</b>	przesunięcie kursora o jedną linię w górę;
<b>CONTROL-↓</b>	przesunięcie kursora o jedną linię w dół;
<b>SHIFT-CONTROL-←</b>	przesunięcie kursora na początek aktualnej linii;
<b>SHIFT-CONTROL-→</b>	przesunięcie kursora na koniec aktualnej linii;
<b>SHIFT-CONTROL-H</b>	przesunięcie kursora na początek pliku;
<b>TAB</b>	przesunięcie kursora do następnej pozycji tabulacji;
<b>SHIFT-TAB</b>	ustawienie pozycji tabulacji;
<b>CONTROL-TAB</b>	skasowanie pozycji tabulacji.

Czasami zachodzi potrzeba podzielenia linii programu na dwie linie. W tym celu należy ustawić kursor na znaku, który ma być pierwszym znakiem w drugiej linii i nacisnąć **SHIFT-CONTROL-RETURN**. Odwrotna operacja jest wykonywana po ustawieniu kursora na pierwszym znaku drugiej linii i naciśnięciu klawiszy **SHIFT-CONTROL-BACKSPACE**.

## REDAGOWANIE TEKSTU

Edytor **Action!** posiada większość funkcji stosowanych w normalnych edytorach tekstu. Jedną z nich jest zamiana pewnych fragmentów tekstu na inne. Umożliwia to polecenie **SHIFT-CONTROL-S**. Po jego użyciu wyświetlany jest napis „Substitute?” i jeśli funkcja ta była już używana, to wyświetlany jest również ostatnio wprowadzony „nowy” tekst. Należy wpisać tekst, który ma być umieszczony w redagowanym programie i nacisnąć **RETURN**, a samo naciśnięcie **RETURN** zachowuje poprzedni tekst. Teraz wyświetlany jest napis „for?”, po którym trzeba wpisać tekst, który ma być zastąpiony (gdy funkcja była już używana należy postąpić analogicznie jak z „nowym” tekstem). Po naciśnięciu **RETURN** edytor znajduje pierwsze wystąpienie podanego „starego” tekstu i zamienia go na „nowy”. Następne polecenie **SHIFT-CONTROL-S** powoduje zamianę wskazanego tekstu w następnym miejscu występowania bez pytań „Substitute?” i „for?” (jeżeli nie została przed tym użyta inna funkcja edytora). Jeżeli podany „stary” tekst nie występuje w redagowanym programie, to wyświetlany jest komunikat „not found” (nie znaleziono).

Możliwe jest także przemieszczanie lub kopiowanie całych bloków tekstu poprzez bufor kopiowania. Za każdym razem, gdy użyte zostanie polecenie **SHIFT-DELETE**, usunięta linia jest umieszczana w buforze kopiowania. Polecenie **SHIFT-CONTROL-P** wstawia usuniętą linię na aktualnej pozycji kursora. Bufor kopiowania jest kasowany przy każdym użyciu **SHIFT-DELETE**, lecz z jednym wyjątkiem. Jeśli **SHIFT-DELETE** jest używane raz po raz (bez innych poleceń lub wpisywania tekstu), to bufor nie jest kasowany. Pozwala to na umieszczenie w buforze całego bloku tekstu i jego wstawienie w dowolnym miejscu poprzez **SHIFT-CONTROL-P**. Ponieważ wstawianie tekstu nie kasuje bufora, to można go kopiować kilkakrotnie.

Jeżeli podczas redagowania linii programu zrobiony został błąd, to można odtworzyć pierwotną postać tej linii poleceniem **SHIFT-CONTROL-U**. Funkcja ta działa tylko pod warunkiem, że kursor nie opuszczał odtwarzanej linii.

## OKNA

Dodatkową funkcją oferowaną przez edytor **Action!** jest możliwość utworzenia drugiego okna i jednoczesnego redagowania dwóch różnych programów. Drugie okno jest tworzone poleceniem **SHIFT-CONTROL-2**. Polecenie to służy także do przeniesienia kursora z pierwszego do drugiego okna. Przeniesienie kursora z okna drugiego do pierwszego wykonuje się poleceniem **SHIFT-CONTROL-1**. W celu skasowania okna najpierw należy umieścić w nim kursor, a następnie nacisnąć klawisze **SHIFT-CONTROL-D**. W linii komunikatów pojawia się napis „Delete Window?”. Dalsze postępowanie jest takie samo jak przy kasowaniu zawartości edytora.

Okno może być przesuwane w górę i w dół przez ruch kursora. Dodatkowe polecenia umożliwiają przesunięcie okna w całości w górę lub w dół. Po poleceniu **SHIFT-CONTROL-↑** okno przemieszcza się w górę tak, że najwyższa linia starego okna jest teraz linią najniższą. Analogicznie polecenie **SHIFT-CONTROL-↓** przesuwa okno w dół. Możliwe jest również przesunięcie całego okna o jeden znak w prawo (**SHIFT-CONTROL-]**) lub w lewo (**SHIFT-CONTROL-[**).

## ETYKIETY

Etykiety pozwalają oznaczyć dowolne miejsce w tekście. W celu ustawienia etykiety na aktualnej pozycji kursora trzeba podać polecenie **SHIFT-CONTROL-T** i po wyświetleniu pytania „tag id:” wpisać

jeden znak identyfikujący etykiety i nacisnąć **RETURN**. Jeżeli etykieta oznaczona podanym znakiem już istnieje, to jest ona usuwana i ustawiana w nowym miejscu.

Polecenie **SHIFT-CONTROL-G** przenosi kursor do etykiety, której znak zostanie wpisany w odpowiedzi na pytanie „tag id:”. Gdy podanej etykiety nie ma w tekście, wyświetlany jest komunikat „tag not set” (etykieta nie ustawiona).

**UWAGA:** Każda operacja, która zmienia zawartość linii, kasuje etykiety w tej linii.

## MONITOR

Monitor **Action!** steruje pracą całego systemu. Linia poleceń monitora znajduje się w górnej części ekranu i zawiera znak > oraz kursor. Pozostała część ekranu stanowi obszar komunikatów. Ma on wiele zastosowań. W czasie działania programu służy do wyświetlania jego wyników. Może być także wykorzystany do śledzenia wykonywania programu. Gdy w programie zostanie wykryty błąd, to w obszarze komunikatów wyświetlany jest numer błędu i część programu z linią zawierającą błąd.

## POLECENIA MONITORA

Monitor rozpoznaje podane mu polecenia tylko po pierwszej literze, nie trzeba więc wpisywać całych słów a tylko literę początkową. Wykonanie wpisanego polecenia następuje po naciśnięciu **RETURN**.

**BOOT (B)** — Powoduje restart systemu i powrót do edytora. Kasuje wszystkie zawarte w pamięci programy i zmienne.

**COMPILE (C)** — Program napisany w **Action!** musi być skompilowany przed wykonaniem. Polecenie **COMPILE** wywołuje kompilator **Action!** i powoduje skompilowanie programu znajdującego się w pamięci edytora. Można także skompilować program zapisany na urządzeniu zewnętrznym. W tym celu trzeba podać nazwę, np C "C:" dla magnetofonu lub C "D1: nazwa.ext" dla stacji dysków. Gdy program ma być odczytany ze stacji numer 1, to można opuścić nazwę urządzenia: C "nazwa.ext". Jeżeli kompilator napotka w programie błąd, to przerywa kompilację i przekazuje sterowanie ponownie do monitora.

**DOS (D)** — Opuszczenie systemu **Action!** i przejście do DOS-u (zawartość edytora i monitora zostaje zniszczona).

**EDITOR (E)** — Przejście do edytora. Jeżeli w czasie kompilacji programu wystąpił błąd, to po przejściu do edytora kursor znajdzie się w linii następującej po błędzie.

**OPTIONS (O)** — Wywołanie menu opcji, które pozwala na zmianę parametrów pracy edytora, kompilatora i monitora. Każda opcja jest wyświetlana w linii poleceń. Jeżeli chcesz ją zmienić, wpisz nową wartość i nacisnij **RETURN**. Samo naciśnięcie **RETURN** pozostawia daną opcję bez zmian. Menu opcji można opuścić naciskając **ESC**.

Poniżej wymienione są wszystkie dostępne opcje. Format opisu; nazwa opcji — tłumaczenie (wartość standardowa — części systemu, na które wpływa).

**Display?** — Ekran (Y — E,K,M)

Ekran może być wyłączany w celu zwiększenia szybkości kompilacji i operacji wejścia/wyjścia. Wpisanie „N” (nie) wyłącza ekran, wpisanie „Y” (tak) pozostawia go włączony cały czas.

**Bell?** — Buczek (Y — E,K,M)

Komputer daje dźwięk bucza zawsze, gdy napotkany zostanie błąd. Można go wyłączyć przez wpisanie tu „N”, wpisanie „Y” włącza buczone ponownie.

**Case sensitive?** — Rozróżnianie liter (N — K)

Gdy ta opcja jest ustawiona na „Y”, to rozróżniane są małe i duże litery (np. „suma”, „Suma” i „SUMA” są różne), a instrukcje języka (np. FOR,



WHILE itd.) muszą być pisane dużymi literami. Jednakże dla ułatwienia pracy początkującym programistom ta opcja jest wyłączona (N) po uruchomieniu systemu.

## Trace? — Śledzenie (N — K)

Dzięki tej opcji można śledzić kompilację programu. Gdy jest ona włączona (Y), kompilator wyświetla w obszarze komunikatów wszystkie wywołane procedury wraz z ich parametrami.

## List? — Listing (N — K)

Ustawienie tej opcji na „Y” powoduje wyświetlanie w obszarze komunikatów aktualnie kompilowanej linii.

## Window 1 size? — Rozmiar okna 1 (18 — E)

Rozmiar okna 1 w edytorze **Action!** jest ustalany bezpośrednio. Wielkość okna 2 jest ustalana pośrednio w zależności od okna 1 — oba okna razem muszą mieć 23 wiersze. Gdy mamy dwa okna, każde z nich może zawierać nie mniej niż 5 wierszy i nie więcej niż 18 wierszy. Wpisanie wartości spoza tego przedziału spowoduje dopasowanie jej do dopuszczalnej wartości.

## Line size? — Rozmiar linii (120 — E)

Rozmiar linii jest liczbą znaków w tej linii. Maksymalny rozmiar linii wynosi 240 znaków. Opcja ta pozwala kontrolować długość linii przy wydruku programu na drukarce.

## Left margin? — Lewy margines (2 — E,M)

Lewy margines jest miejsce, od którego wyświetlane są znaki na ekranie. Może mieć wartość z zakresu od 0 do 39.

## EOL character? — Znak końca linii (spacja — E)

Znak końca linii jest wyświetlany przez edytor **Action!** na końcu każdej linii. Normalnie jest on niewidoczny (pusta spacja), a uwidocznienie go może pomóc w redagowaniu programu. Może tu wpisać dowolny znak (także graficzny) — sugeruję użycie CONTROL-. lub CONTROL-T.

**PROCEED (P)** — Wznawia wykonywanie programu zatrzymanego naciśnięciem klawisza BREAK lub przy użyciu procedury 'Break'. Program jest kontynuowany tak jakby nie nastąpiło żadne przerwanie.

**RUN (R)** — Uruchomienie skompilowanego programu. Może występować w czterech różnych formatach:

RUN	— uruchomienie programu znajdującego się w pamięci edytora;
RUN "nazwa pliku"	— wczytanie i uruchomienie programu z urządzenia zewnętrznego;
RUN adres	— uruchomienie programu od podanego adresu;
RUN nazwa procedury	— uruchomienie jednej procedury z programu lub biblioteki procedur.

**SET (S)** — Polecenie umożliwiające bezpośredni dostęp do pamięci RAM i zmianę jej zawartości. Działa tak samo jak instrukcja SET w języku **Action!**.

**WRITE (W)** — Pozwala na zapisanie skompilowanego programu na urządzeniu zewnętrznym jako pliku binarnego. Program zapisany w ten sposób na dysku może być uruchomiony bezpośrednio z DOS-u (po wczytaniu kompilatora). Przy zapisie na kasecie należy podać nazwę "C:", a na dysku — "D1:nazwa.ext" (dla stacji numer 1 D1: można opuścić).

**XECUTE (X)** — Polecenie umożliwiające wykonanie dowolnej instrukcji języka **Action!** lub dowolnej dyrektywy kompilatora (oprócz MODULE i SET) z monitora. Po poleceniu należy wpisać instrukcję do wykonania, np. XECUTE PrintE("Witaj w Action!").

? — Wyświetlenie aktualnej zawartości pamięci o podanym adresie. Adresem może być dowolna stała kompilatora (opisane są w dalszej części). W obszarze komunikatów wyświetlany jest adres w postaci dziesiętnej i szesnastkowej, znak ATASCII o kodzie odpowiadającym zawartości komórki, zawartość dwóch komórek w postaci szesnastkowej oraz liczby dziesiętne w formacie BYTE i CARD. Na przykład:

```
? $FFFE
65535,$FFFE = s $E6F3 243 59123
* — Wyświetlenie aktualnej zawartości pamięci od podanego adresu. Adresem może być dowolna stała kompilatora. Format wyświetlanego wyniku jest taki jak opisany wyżej. Wyświetlanie przerywa się naciśnięciem klawisza spacji. Naciśnięcie CONTROL-1 chwilowo wstrzymuje wyświetlanie, a ponowne naciśnięcie wznowia je.
```

## URUCHAMIANIE PROGRAMU

Przy uruchamianiu napisanego programu często zachodzi konieczność wyszukiwania popełnionych błędów. Dzięki możliwościom monitora **Action!** jest to czynność łatwa i niekłopotliwa.

Opcja TRACE — Po zezwoleniu tej opcji monitora kompilowany i wykonywany program jest śledzony na ekranie. Nazwa każdej wywołanej procedury jest wyświetlana na ekranie wraz z przekazywanymi parametrami.

Przed wprowadzeniem do uruchamianego programu poprawek konieczne jest zatrzymanie jego wykonywania. W **Action!** jest to możliwe dwoma sposobami: klawiszem BREAK i procedurą 'Break'.

Klawisz BREAK — Zasadniczo klawisz BREAK nie działa w **Action!**. Jednak w dwóch przypadkach można przy jego pomocy przerwać wykonywany program: podczas operacji wejścia/wyjścia i przy wywołaniu procedur, które mają więcej niż 3 parametry.

Procedura biblioteczna PROC Break() — Jeżeli chcemy zatrzymywać uruchamiany program co jakiś czas w celu sprawdzenia poprawności jego działania, to należy w kilku miejscach umieścić wywołanie procedury 'Break'. Działa ona jak instrukcja BASIC-a STOP, a działanie programu można wznowić poleceniem PROCEED.

## SŁOWA KLUCZOWE

Bliższą znajomość z językiem **Action!** zaczniemy od poznania „słów kluczowych”. Są to słowa i symbole, które mają specjalne znaczenie dla kompilatora — instrukcje, deklaracje, operatory i dyrektywy kompilatora. Słowa kluczowe mogą być używane tylko w sposób zdefiniowany w **Action!**, w szczególności nie wolno ich używać jako nazw zmiennych. Oto lista tych słów:

AND	FI	OR	UNTIL	=	(
ARRAY	FOR	POINTER	WHILE	<>	)
BYTE	FUNC	PROC	XOR	#	.
CARD	IF	RETURN	+	>	[
CHAR	INCLUDE	RSH	-	>=	]
DEFINE	INT	SET	*	<	"
DO	LSH	STEP	/	<=	'
ELSE	MOD	THEN	&	\$	;
ELSEIF	MODULE	TO	%	'	
EXIT	OD	TYPE	!	@	

## SYMBOLE

Przy opisie języka będziemy używać pewnych symboli dla uwidocznienia składni. Oto lista tych symboli wraz z objaśnieniem:

**adres** Adres jest to numer komórki w pamięci komputera. Gdy polecamy komputerowi, aby umieścił coś w pamięci, to musimy podać mu adres (jak na poczcie).

**identyfikator** Jest to nazwa określająca zmienną, procedurę itp. Nazwa w **Action!** musi

spełniać następujące warunki:

1. Musi rozpoczynać się od litery.
2. Pozostałe znaki muszą być literami, cyframi lub znakiem podkreślenia (\_).
3. Nie może być słowem kluczowym.

**litera**

Dowolna litera alfabetu, zarówno mała, jak i duża.

**znak**

Litera lub cyfra od 0 do 9.

**MSB, LSB**

MSB oznacza Most Significant Byte (bardziej znaczący bajt), LSB — Least Significant Byte (mniej znaczący bajt).

W systemie dziesiętnym nie ma bajtów znaczących, lecz cyfry. Na przykład w liczbie 54 bardziej znaczącą cyfrą jest 5, a mniej — 4. Podobnie jest w zapisie komputerowym, ale zgodnie ze standardem mikroprocesora 6502 liczby są zapamiętywane i używane w kolejności LSB, MSB.

**\$**

Znak dolara, gdy jest użyty przed liczbą, oznacza liczbę w zapisie szesnastkowym (hexadecymalnym).

**;**

Średnik jest symbolem komentarza i wszystko, co po nim znajduje się w linii jest ignorowane przez kompilator.

**< i >**

Wyrażenie objęte tymi symbolami określa, co powinno znaleźć się w tym miejscu w instrukcji lub linii programu. Np. <identyfikator> oznacza, że musi być użyty prawidłowy identyfikator.

Wyrażenie zawarte między tymi symbolami może być użyte, lecz nie musi. Np. [<identyfikator>] oznacza, że można użyć identyfikatora, ale nie jest to konieczne.

Te symbole oznaczają powtórzenie (jak w muzyce). Np. !:<identyfikator>:! oznacza, że można użyć dowolnej liczby identyfikatorów (również żadnego).

Ten symbol oznacza alternatywę. Np. wyrażenie <identyfikator> \* <adres> należy odczytać: użyj identyfikatora lub adresu, ale nie obu na raz.

## TYPY DANYCH

Przed opisem typów danych stosowanych w **Action!** musimy trochę powiedzieć o zmiennych i stałych, ponieważ są one podstawowymi obiektami, na których operuje komputer.

## ZMIENNE

Dozwolona nazwa zmiennej musi być prawidłowym identyfikatorem. Nie ma żadnych innych ograniczeń dla stosowanych nazw. Aby uzyskać przejrzysty program należy używać znaczących nazw zmiennych np. 'licznik' zamiast 'i'. Nie wpływa to na długość programu wynikowego (skompilowanego), a jedynie zwiększa o kilka bajtów program źródłowy.

## STAŁE

W **Action!** używane są trzy typy stałych: stałe liczbowe, stałe znakowe i stałe kompilatora.

Stałe liczbowe mogą być wprowadzane w trzech różnych formach: jako liczby dziesiętne, liczby szesnastkowe i znaki.

Liczby dziesiętne nie wymagają żadnego specjalnego określenia, np. 2, 46, 324, 65500, -4360 (ujemna). Liczby szesnastkowe są oznaczane znakiem dolara (\$) przed liczbą, np. \$0D, \$300, \$4A00, -\$8C (ujemna). Stałe znakowe są oznaczane apostrofem (') przed znakiem, np. 'A', '@', ' ', 'f'.



Znaki są stałymi liczbowymi, ponieważ są one wewnętrznie reprezentowane przez liczbę bajtową, która jest równa kodowi ATASCII znaku.

Stałe kompilatora różnią się od trzech powyższych typów. Są one używane w czasie kompilacji programu od ustawienia pewnych atrybutów zmiennych, procedur oraz funkcji i nie są stosowane podczas działania programu. Stałe kompilatora mogą mieć następujące formaty:

1. stała liczbowa
2. identyfikator
3. wskaźnik
4. suma dwóch dowolnych wyżej wymienionych

Pierwszy format nie wymaga wyjaśnień, lecz pozostałe trzeba krótko opisać. Gdy używasz identyfikatora (nazwy zmiennej, procedury lub funkcji) jako stałej kompilatora, to używaną wartością jest adres tego identyfikatora. Trzeci format pozwala użyć jako stałych kompilatora wskaźników tablic. Ostatni jest zwykłym dodawaniem dowolnych dwóch z trzech pierwszych typów. Dla lepszego zrozumienia podamy kilka przykładów:

licznik	;adres zmiennej "licznik"
\$8D00	;stała szesnastkowa
numer	;wskaźnik jako stała
5+wsk	;5 plus zawartość wskaźnika "wsk"
\$80+n	;\$80 plus adres zmiennej "n"

## TYPY ZMIENNYCH

W **Action!** występują trzy podstawowe typy zmiennych: **BYTE**, **CARD** i **INT** oraz dodatkowe, które będą opisane później. Wszystkie podstawowe typy zmiennych są liczbowe.

**BYTE** — Dodatnie liczby całkowite mniejsze niż 256. Są one reprezentowane wewnętrznie przez jednobajtową liczbę bez znaku (z zakresu od 0 do 255). Dzięki takiej reprezentacji **BYTE** może być używana jako zmienna znakowa, można więc używać zamiennie słów kluczowych **BYTE** i **CHAR**.

**CARD** — Podobnie jak **BYTE**, lecz o większym zakresie. Reprezentacją wewnętrzną jest liczba dwubajtowa, więc zmienne typu **CARD** mogą mieć wartość z zakresu od 0 do 65535. Liczby typu **CARD** są zapamiętywane w formacie LSB, MSB.

**INT** — Liczby całkowite ze znakiem, reprezentacja i zapis jak **CARD**. Najstarszy bit jest interpretowany jako znak, więc zakres liczb **INT** jest od -32768 do 32767.

## DEKLARACJE

Przed użyciem każdego identyfikatora trzeba go zadeklarować, aby komputer zarezerwował odpowiednie miejsce w pamięci. Format deklaracji jest następujący:

<typ><identyf.> [=<inform.pocz.>]:<identyf.> [=<inform.pocz.>]:!

gdzie <inform.pocz.> jest początkową wartością lub adresem zmiennej: <adres> I[<wartość>].

Przykłady:

BYTE wynik,ilość	;deklaracja zmiennych "wynik" i "ilość" jako typ BYTE
INT num=[0]	;zmienna "num" typu INT o wartości 0
BYTE x=\$8000	;zmienna "x" typu BYTE umieszczona w pamięci pod adresem \$8000
CARD pkt=[0], wsk=\$83D4	;zmienne CARD "pkt" o wartości 0 i "wsk" pod adresem \$83D4. Zmienne nie muszą być w tej samej linii.

Deklaracje zmiennych muszą znajdować się w programie bezpośrednio po instrukcji **MODULE** albo na początku procedury lub funkcji.

Wojciech Zientara

## POLECENIA EDYTORA

Przejdź do monitora <CTRL><SHIFT> M (monitor)

### Polecenia I/O

Odczyt pliku <CTRL><SHIFT> R — nazwa (read)

Odczyt directory <CTRL><SHIFT> R — ?n:\*. \* (read)

Zapis pliku <CTRL><SHIFT> W — nazwa (write)

Wydruk <CTRL><SHIFT> W — P: (write)

### Ruch kursora

Kursor w górę <CTRL> ↑

Kursor w dół <CTRL> ↓

Kursor w prawo <CTRL> →

Kursor w lewo <CTRL> ←

Kursor na początek linii <CTRL><SHIFT> <

Kursor na koniec linii <CTRL><SHIFT> >

Następna linia <RETURN>

Tabulacja <TAB>

Ustawienie tabulacji <SHIFT><TAB>

Skasowanie tabulacji <CTRL><TAB>

### Ruch okna ekranu

Początek pliku <CTRL><SHIFT> H (head)

1 ekran w górę <CTRL><SHIFT> ↑

1 ekran w dół <CTRL><SHIFT> ↓

1 znak w lewo <CTRL><SHIFT> ]

1 znak w prawo <CTRL><SHIFT> [

Utworzenie drugiego okna <CTRL><SHIFT> 2

Przejdź do pierwszego okna <CTRL><SHIFT> 1

Przejdź do drugiego okna <CTRL><SHIFT> 2

Kasowanie okna <CTRL><SHIFT> D (delete)

### Redagowanie tekstu

Wstawianie/wymiana <CTRL><SHIFT> I (insert)

Odtworzenie zmienionej linii <CTRL><SHIFT> U (undone)

Odtworzenie usuniętej linii <CTRL><SHIFT> P (paste)

Zapamiętanie bloku tekstu <SHIFT><DELETE>

Wstawienie tekstu z bufora <CTRL><SHIFT> P (paste)

Wyszukiwanie łańcucha <CTRL><SHIFT> F (find)

Wymiana łańcucha <CTRL><SHIFT> S (substitute)

Podzielenie linii na dwie <CTRL><SHIFT><RETURN>

Połączenie dwóch linii <CTRL><SHIFT><DELETE>

Ustawienie etykiety <CTRL><SHIFT> T (tag set)

Odszukanie etykiety <CTRL><SHIFT> G (go to tag)

## POLECENIA MONITORA

Restart Action! B (boot)

Przejdź do DOS-u D (DOS)

Przejdź do edytora E (editor)

Menu wariantów O (options)

Wykonanie instrukcji X <instrukcja> (xecute)

Kompilacja programu C ["nazwa"] (compile)

Zapis programu W ["nazwa"] (write)

Uruchomienie programu R ["nazwa"] (run)

Kontynuacja programu P (proceed)

Ustawienie wartości (POKE) SET <adres> = <wartosc>

Zawartość komórki (PEEK) ? <adres>

Przegląd pamięci \* <adres>

## KODY BŁĘDÓW Action!

### Kod Znaczenie

- 0 Za mały obszar dostępnej pamięci.
- 1 Brak cudzysłowu (") w łańcuchu.
- 2 Zagnieżdżona dyrektywa DEFINE.
- 3 Brak miejsca w tablicy symboli zmiennych globalnych.
- 4 Brak miejsca w tablicy symboli zmiennych lokalnych.
- 5 Błąd składni w dyrektywie SET.
- 6 Nieprawidłowy format deklaracji.
- 7 Zbyt dużo argumentów w instrukcji lub procedurze.
- 8 Niezadeklarowana zmienna.
- 9 Użyto zmienną w miejscu, w którym wymagana jest stała.
- 10 Niedozwolona instrukcja podstawienia.
- 11 Nierozpoznany błąd.
- 12 Brak instrukcji THEN.
- 13 Brak instrukcji FI.
- 14 Zbyt mały obszar pamięci dla kodu wynikowego.
- 15 Brak instrukcji DO.
- 16 Brak instrukcji TO.
- 17 Nieprawidłowy format wyrażenia.
- 18 Niezamknięty nawias.
- 19 Brak instrukcji OD.
- 20 Nie można przesunąć więcej pamięci.
- 21 Niedozwolona tablica.
- 22 Zbyt duży plik wejściowy.
- 23 Niedozwolone wyrażenie warunkowe.
- 24 Błąd składni w instrukcji FOR.
- 25 Niedozwolona instrukcja EXIT.
- 26 Zbyt głębokie zagnieżdżenie (dozwolone 16 poziomów).
- 27 Błąd składni w instrukcji TYPE.
- 28 Niedozwolona instrukcja RETURN.
- 61 Brak miejsca w tablicy symboli.
- 128 Zatrzymanie programu klawiszem BREAK.



**W**ydaje się, że przełom 1987/88 roku będzie b. znaczący dla ATARI i jego szefa, Jacka Tramiela. W wyniku podpisanej ostatnio umowy między Atari a Commodore, trwającej ponad cztery lata właśnie, kłótnie i swoistego rodzaju „wojna podjazdowa”, zakończyły się, przy czym żadna firma nie poniosła porażki, ani nie odniosła zwycięstwa. Zakopano więc głęboko w Krzemowej Dolinie chip niezgody, powrócono do własnych laboratoriów, wycofując z prasy wzajemne oskarżenia w rodzaju „Obiecanki — cacanki Jacka Tramiela”, „Commodore zdzierca” itp.

Pogodzony więc z Commodore Jacek Tramiel (zresztą jej były szef), ufnie patrzy w przyszłość, czemu raz po raz daje wyraz na wielu konferencjach prasowych i pokazach sprzętu. Realizując politykę anonsowania produkcji swoich wyrobów Tramiel — w przeciwieństwie do firmy Compaq, która lansuje swoje nowości... pierwszego dnia dostawy urządzeń do sklepów USA i Europy Zachodniej — zapowiedział już teraz, iż 1988 rok będzie dla ATARI rokiem przełomowym. Firma chce bowiem wykupić w USA za sumę 67 mln. dol. sieć 67 sklepów, w których będą sprzedawane... oczywiście komputery Atari!

Poza tym na rok 1988 Atari zapowiedział szereg interesujących nowości: przede wszystkim sukcesywnie ma się ukazać na rynku cała gama komputerów kompatybilnych z IBM PC: po modelach PC i PC 2 przyjdzie kolej na komputer AT z procesorem 80286, a następnie nowy, z procesorem 80386. Mega ST będzie posiadał swojego młodszego brata, na razie jeszcze bez nazwy, ale z pełnym, 32-bitowym procesorem Motoroli 68030, ekranem graficznym wysokiej rozdzielczości, systemem eksploatacyjnym Unix i wielodostępem.

Lada dzień ma się ukazać w sklepach CD-ROM, który służy jednocześnie jako... zwykły odtwarzacz płyt CD! Powstaje jedynie pytanie czy znajdzie dużo chętnych melomanów do jego przenoszenia, a przede wszystkim ciągłego włączania i odłączania.

Część z tych urządzeń ma mieć swoją premierę na marcowych targach CEBIT w Hanowerze.

**F**irma Cumana produkuje 5 różnych stacji dysków dla Atari serii ST: stacja 3,5-cala, dwugłowicowa (720 KB po sformatowaniu), stacja podwójna 3,5-cala z dwiema głowicami, stacja podwójna 5,25-cala (720 KB po sformatowaniu, z możliwością wyboru 40 lub 80 ścieżek — ułatwia odczytanie dyskietek IBM PC przy zastosowaniu odpowiedniego emulatora lub przeniesienie programów na dyskietki bardziej ekonomiczne), podwójna stacja 3,5-calowa wspólna w jednej obudowie oraz stacja 3,5- i 5,25-cala, dwugłowicowa, również we wspólnej obudowie. Niżej stacje posiadają własne zasilanie, są szybkie i bardzo ciche. Uwaga: wszystkie stacje podwójne są jedynie dla posiadaczy ST bez wbudowanych stacji ponieważ system eksploatacyjny 520 i 1040 „widzi” tylko dwie stacje: posiadacze Atari STF w żadnym wypadku nie będą mogli użyć drugiej podwójnej stacji.

## Zabezpieczenie programów pisanych w BASIC-u wymaga stosowania różnorodnych sposobów. Poniżej postaram się omówić specyfikę postępowania z programami zapisanymi na taśmie lub na dyskietce — w obu przypadkach stosuje się podobne metody zabezpieczenia.

Poważnym problemem zabezpieczenia BASIC-a jest instrukcja SAVE. Zabezpieczenie przeciw użyciu tej instrukcji jest znacznie bardziej skomplikowane niż mogłoby się wydawać. Głównym powodem takiej sytuacji jest istnienie wielu sposobów wstrzymania wykonywania programu, a następnie zarejestrowania go przy pomocy instrukcji SAVE "D:nazwa", lub SAVE "C:". Aby uniemożliwić potencjalnemu piratowi skopiowanie programu należy pamiętać o następujących rzeczach:

1. Uniemożliwienie postużenia się klawiszem BREAK;
2. Uniemożliwienie postużenia się klawiszem systemowym RESET;
3. Zapobieżenie przerwaniom przez błąd;
4. Zapobieżenie kombinacjom LOAD/SAVE.

Program napisany w BASIC-u ma tę właściwość, że raz załadowany do pamięci jest łatwy do zarejestrowania na dyskietce lub kasecie, jeśli użytkownik programu ma tę możliwość. Rzecz w tym, aby mu tę możliwość odebrać, czyli innymi słowy, program nie może tracić kontroli nad komputerem.

Przejdźmy teraz do pierwszego punktu naszych rozważań, czyli jak uniemożliwić postużenie się klawiszem BREAK. W przypadku użycia klawisza BREAK zatrzymuje się wykonywanie programu bez kasowania zawartości pamięci, co pozwala w prosty sposób zarejestrować kopię programu. Klawisz BREAK blokujemy dwiema instrukcjami:

### **POKE 16,112:POKE 53774,112**

Należy dodać, że powyższe instrukcje trzeba powtarzać po każdej instrukcji GRAPHICS, gdyż niweluje ona nasze zabezpieczenie. Następuje powrót do sytuacji przed wpisaniem zabezpieczenia. Natomiast inne instrukcje mogą zawartość tych komórek np. kasować. Żeby być pewnym skuteczności zabezpieczenia, należy kilkanaście razy powtórzyć te ingerencje w programie. Innym sposobem zablokowania BREAK jest wpisanie instrukcji POKE 566,158 na początku programu (ta instrukcja nie wymaga powtarzania). Można też skorzystać z własnej procedury obsługi przerwania BREAK, jak to zostało opisane w „Bajtku” nr 2/87.

Uniemożliwienie postużenia się klawiszem systemowym RESET. Istnieją trzy proste sposoby zablokowania tego klawisza:

### **a) POKE 580,1**

Instrukcja ta spowoduje w przypadku naciśnięcia RESET tzw. „zimny start”, czyli system automatycznie

## JAK ZABEZPIECZAĆ PROGRAMY W

# BASIC-u?

wystartuje (tak jak przy wyłączeniu i ponownym włączeniu komputera).

### **b) POKE 9,255**

Spowoduje to zablokowanie systemu, jeśli zostanie naciśnięty klawisz RESET (klawisze nie będą działały). Aby uruchomić system trzeba wyłączyć i ponownie włączyć komputer, a następnie załadować program.

### **c) POKE 829,0 lub POKE 830,0 lub POKE 831,0**

Każda z tych instrukcji również powoduje „zimny start” po naciśnięciu klawisza RESET.

Zapobieżenie przerwaniu przez błąd. Aby ustrzec się przed takimi przerwaniem należy stosować instrukcję TRAP, która w momencie wykrycia błędu powoduje skok do określonej linii programu, gdzie np. może nastąpić ponowny start programu. Należy zaznaczyć, że jeżeli raz użyliśmy instrukcji TRAP, to trzeba ją powtarzać, jeśli chcemy, aby program dalej ignorował błędy.

Zabezpieczenie programu przed kombinacją LOAD/SAVE. Zabezpieczenie tego rodzaju opiera się na dwóch czynnościach:

spowodowaniu, aby program uruchamiał się od razu po wprowadzeniu do pamięci komputera;

uniemożliwieniu uzyskania listingu programu.

Pierwsza część tej metody polega na zapisie programu na dyskietce lub kasecie w taki sposób, aby nie można go było wprowadzić do pamięci komputera instrukcją LOAD, gdyż takie wprowadzenie wymaga instrukcji RUN dla uruchomienia programu, a co za tym idzie umożliwia zapisanie go na innym nośniku.

W przypadku dyskietki program musi być wprowadzany instrukcją RUN "D:nazwa". Program uruchomi się automatycznie po wprowadzeniu do pamięci. Jeśli dysponujemy magnetofonem, należy zapisać program na taśmie instrukcją SAVE "C:", a wprowadzać do pamięci komputera — RUN "C:".

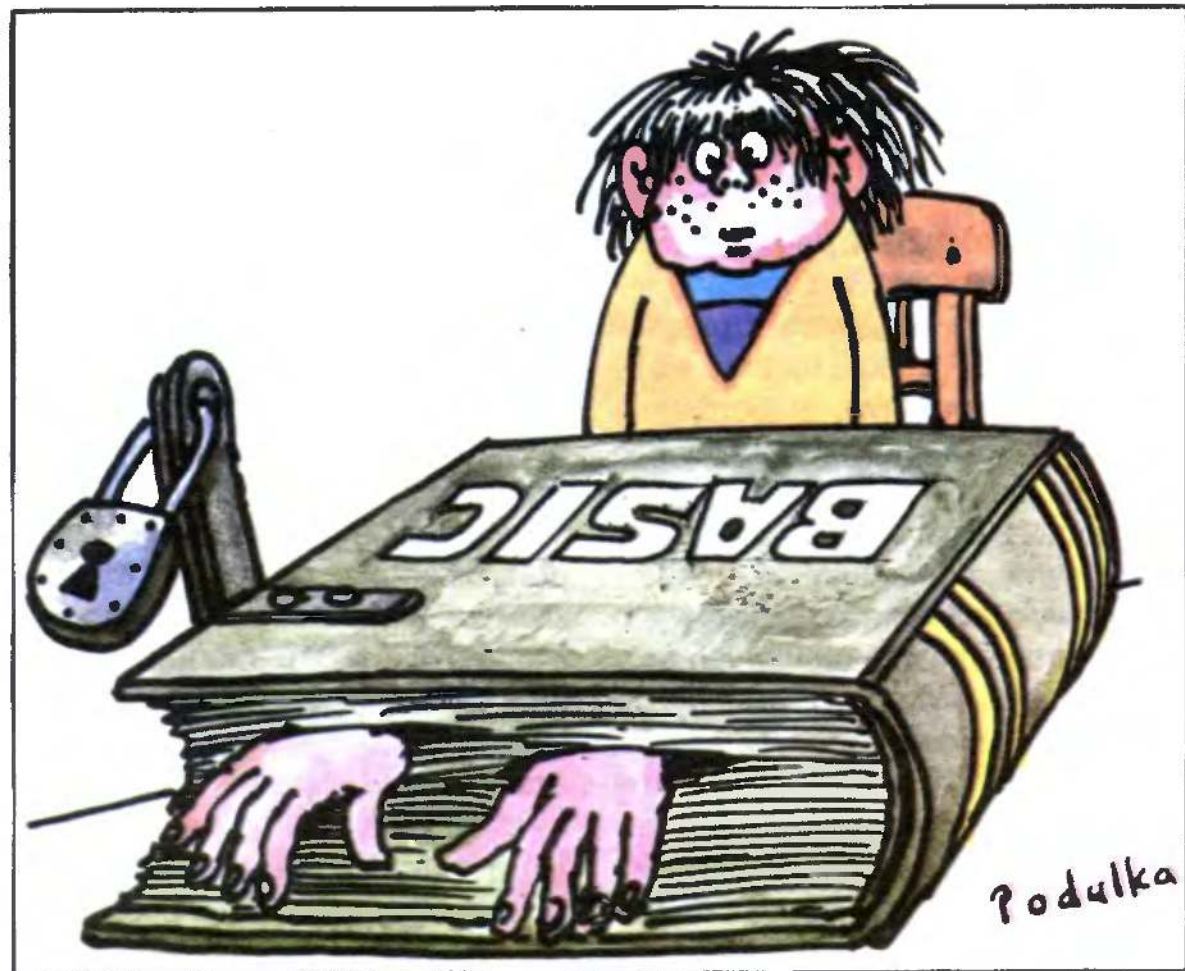
Drugą częścią tej metody jest uniemożliwienie uzyskania listingu programu. Możemy to osiągnąć wpisując jako ostatnią do programu linię: **POKE PEEK(138)+256\*PEEK(139)+2,0:SAVE "C:":NEW.**

W wypadku dyskietki zamiast SAVE "C:" piszemy SAVE "D:nazwa". Jeśli teraz (po wprowadzeniu tej linii do programu) wpisujemy GOTO nr. (nr. — numer linii, gdzie wprowadziliśmy zabezpieczenie) i naciśniemy RETURN, zabezpieczona wersja zostanie zapisana na nośniku (nie zapomnijmy o niezabezpieczoną kopii dla siebie).

Powyższe zabezpieczenie uniemożliwia otrzymanie listingu. Ale oprócz tego program tak zabezpieczony po wprowadzeniu do pamięci instrukcją LOAD nie będzie reagował na żadne instrukcje, także na instrukcję RUN. Nie należy więc zapominać o wprowadzeniu go poprzez RUN "C:", ewentualnie RUN "D:nazwa".

Omówione powyżej metody pozwalają zabezpieczyć program napisany w BASIC-u, tak na taśmie magnetofonowej, jak i na dyskietce. Jednak tylko w przypadku łącznego ich używania możemy mieć pewność, że program jest dobrze zabezpieczony.

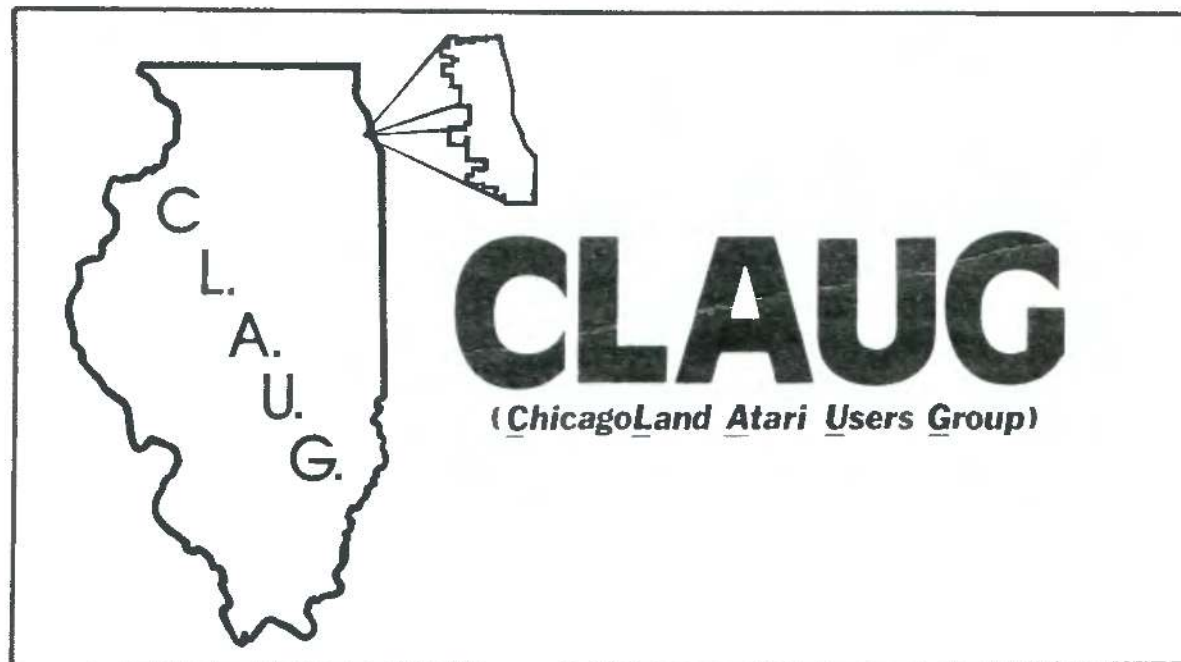
*Piotr Adamczewski*





# CLAUG

**CLAUG — Chicago-Land Atari Users Group (Chicagowska Grupa Użytkowników Atari) jest jedną z największych grup w rejonie Chicago. Została ona założona w 1981 roku jako Atari Computer Users of Chicago (Użytkownicy Komputerów Atari w Chicago).**



„Rozpoczynaliśmy mając tylko około 30 członków” — mówi sekretarz grupy Chuck Schwark — „Teraz zrzeszamy ponad 230 użytkowników, co powoduje, że jesteśmy największą i najstarszą w Illinois grupą ludzi oddanych całkowicie komputerom Atari. Rozwijamy się nadal i nasza grupa powiększa się o 7—15 nowych członków miesięcznie”.

Schwark jest również zarządcą bazy danych grupy opiekującym się kartoteką członków oraz wydawcą comiesięcznego biuletynu. Redaguje on i drukuje matrycę biuletynu. Następnie jeden z jego asystentów wykonuje fotokopie. Biuletyn CLAUG jest wymieniany z biuletynami innych grup użytkowników Atari w całych Stanach Zjednoczonych, jak również w Kanadzie, Szwajcarii, Niemczech Zachodnich, Afryce Południowej, Panamie, Polsce i Hiszpanii.

Od 1985 roku CLAUG jest jedną z grup zapraszanych przez Atari Corp. do prowadzenia własnego stoiska na Consumer Electronic Show, który odbywa się corocznie w czerwcu w Chicago. W tym właśnie roku w uznaniu wkładu pracy członków CLAUG w przygotowanie i przeprowadzenie pokazu szef Atari Sam Tramiel ofiarował grupie system 520ST. Był to pierwszy kompletny system ST przekazany nie producentom oprogramowania, lecz zwykłym użytkownikom. „Ponieważ mieszkamy w Chicago” — mówi Schwark — „i wielu z nas, łącznie ze mną, pracuje na polu elektroniki i komputerów, to normalnie tak czy inaczej obsługujemy CES. Mówimy między sobą, że moglibyśmy pomóc, poprowadzić coś i rozpoczynamy pomaganie w pokazie przez wykonywanie podstawowych prac i uruchamianie niektórych stoisk. Udaje się nam to całkiem nieźle — zwykle pracuje przy tym 10—12 ochotników. Poza tym nie mamy żadnego stałego punktu komputerowego, lecz prowadzimy stoisko demonstracyjne na targach Compu-

ter Central, które odbywają się w Chicago co dwa miesiące.”

Ostatnio grupa przyłączyła się do popierania i finansowania Szpitala-Pomnika Dzieci w Chicago. Korporacja Atari ofiarowała dwa komputery, a Spinnaker i inni producenci dostarczyli oprogramowanie. Aktualnie dwa oddziały szpitala — Child Life i Child Psychiatri — mają kompletne systemy komputerowe. Ochotnicy spośród członków CLAUG uczą teraz dzieci i personel szpitala obsługi komputerów Atari oraz korzystania z oprogramowania. Prowadzą także z dziećmi spotkania, na których odpowiadają na liczne pytania związane z komputerami.

## CZŁONKOWIE

Obszar działania CLAUG pokrywa teren Chicago i sześciu sąsiednich okręgów, są nawet członkowie spoza stanu. Razem obszar ten liczy około 800—900 mil kwadratowych. Grupa liczy około 230 członków, z których każdy płaci rocznie składkę w wysokości 15 \$. Większość członków pracuje lub prowadzi własne interesy. Nie ma wśród nich wielu młodych dorosłych członków, lecz wiele osób posiadających rodziny wprowadziło do grupy swoje dzieci. Nie ma jednak specjalnej sekcji dla dzieci, choć spora grupa młodzieży upoważnia już do odrębnego ich traktowania.

Wielu nowych członków CLAUG jest właścicielami ST lub początkującymi użytkownikami komputerów. Sekcja ST uruchomiła ostatnio publicznie dostępną bibliotekę 16-bitową. Oczywiście CLAUG posiada dużą publiczną bibliotekę 8-bitową oraz bibliotekę sprzętową wyposażoną w komputer Atari 800, dwie stacje dysków 810, modem, tabliczkę graficzną, drukarkę 40-kolumną i interface 850, a także otrzymany od Atari kolorowy system ST.

„Mimo, że istnieją pewne grupy, które twierdzą, że są największe, my mówimy tylko, że jesteśmy jed-

ną z największych, jeśli nie największą grupą w Illinois. Ponieważ średnia rodzina w grupie (według stanu na czerwiec 1986) ma 1,2 dziecka, to możemy zwiększyć liczbę członków rzeczywiście należących do grupy o 20—25%. Natomiast średnie gospodarstwo domowe posiada 1,9 komputera”.

## SERWIS

CLAUG prowadzi całodobowy, publicznie dostępny serwis informacyjny (BBS), który nie wymaga znajomości hasła. Ma on całkowitą pojemność 8,5 MB i pracuje w oparciu o komputer 130XE z RAM-dyskiem 1 MB, BASIC-em XL i systemem operacyjnym MYDOS. Informacja jest przechowywana w osmiu dwustronnych stacjach dysków TEAC o podwójnej gęstości zapisu po 26 sektorów na ścieżkę. Cztery z nich mają format 80 ścieżek, a reszta 77 ścieżek na jednej stronie dyskietki. Całe to wyposażenie jest własnością prezesa grupy Petera Pacione.

Serwis notuje średnio 200—300 wywołań tygodniowo i odpowiadał dotychczas na około 60 000 wywo-

łań. Jest on bardzo często zajęty i niełatwo uzyskać połączenie. Serwis działa samodzielnie przez całą dobę, a sysop kontroluje go tylko od czasu do czasu, aby stwierdzić, że kontrolka modemu stale świeci i stacje dysków ciągle pracują.

## STATYSTYKA

Okolo 30% członków grupy posiada komputery ST, a 10% z nich także 8-bitowe. Najszerzej używanymi komputerami 8-bitowymi są 800XL i 130XE, a najbardziej popularną stacją dysków jest Atari 1050. Na jednego członka grupy wypada średnio 2,1 stacji dysków. Okolo 40% osób wydaje na swoje komputery zainteresowania ponad 500 \$ rocznie, a 26% ponad 250 \$. Epson i Gemini to najczęściej używane drukarki. Najpopularniejszymi modemami są Atari 1030 i XM301 oraz modele MPP/Supra. 94% członków posiada modemy, a 25% korzysta z komercyjnych serwisów telefonicznych.

Programiści stanowią 80—85% członków grupy. Najpopularniejszym językiem ST jest „C”, a najczęściej używane języki 8-bitowe to BASIC, Assembler i Forth.

Rodzaje oprogramowania najszerzej wykorzystywanego przez członków grupy układają się w następującej kolejności:

1. Edytory tekstu
2. Rozrywka
3. Edukacja
4. Interesy i finanse
5. Programowanie
6. Komunikacja
7. Bazy danych
8. Finanse osobiste
9. CAD/CAM

opracowanie w/g „Antic” 6/87

## KLUBY ATARI W POLSCE

<b>MIKROLUD</b>	ul. Łużycka 53/256, 30—658 Kraków
<b>NORA</b>	
<b>MIKROPROCESORA</b>	ul. Jasna 28/8, 85—205 Bydgoszcz
<b>COMPUTERLAND</b>	ul. Teatralna 36a/4, Świdnica
<b>INTEX</b>	ul. Mieszka I 8/4, 08—110 Siedlce
<b>ROMMMM</b>	ul. Waryńskiego 28/48, 00—650 Warszawa
<b>Szkoła Podstawowa nr 1</b>	ul. Jeziorna 12, 74—320 Barlinek
<b>Mr. ATARI</b>	ul. Konarskiego 6/40, 26—900 Kozienice
<b>C.K.M. ATARI</b>	ul. Boh. Warszawy 7, Choszczno
<b>KKMMT BAJTEK</b>	ul. Zygmunta Augusta 5, 31—504 Kraków
<b>KM BAJTEK</b>	ul. Akademicka 8, 22—400 Zamość
<b>ATARI</b>	ul. Murzynowskiego 8, 10—684 Olsztyn
<b>FENIKS (przy M-GOK)</b>	ul. Ratuszowa, 73—210 Recz





**Zaprogramowanie** gry przygodowej nie jest trudne. Wystarczy mieć ciekawy pomysł, odrobinę cierpliwości i wystuchać kilku dobrych wskazówek fachowca, by to pozornie skomplikowane zadanie stało się możliwe do zrealizowania.

Data Becker, spółka z Düsseldorfu znana ze znakomitej serii książek poświęconych komputerowi i informatyce postanowiła ułatwić wszystkim ciekawym poznanie tajemnic gier przygodowych od podszewki. Celowi temu służy wydana w 1984 r. książka „Adventures und wie man sie auf dem Atari 600XL/800XL programmiert”. Jej autorem jest Jörg Walkowiak, wówczas student informatyki i twórca wielu udanych programów.

W fascynujący świat fantazji i przygody wprowadza nas Jörg Walkowiak przedstawiając krótko historię rozwoju i teraźniejszości gier komputerowych. Dzieli się przy tym swoimi refleksjami na temat roli jaką odgrywają w życiu współczesnego człowieka.

W dalszej części znajdujemy pieśsze wskazówki, w jaki sposób przystąpić do tworzenia własnej gry. Okazuje się, że ważne są nie tylko umiejęt-

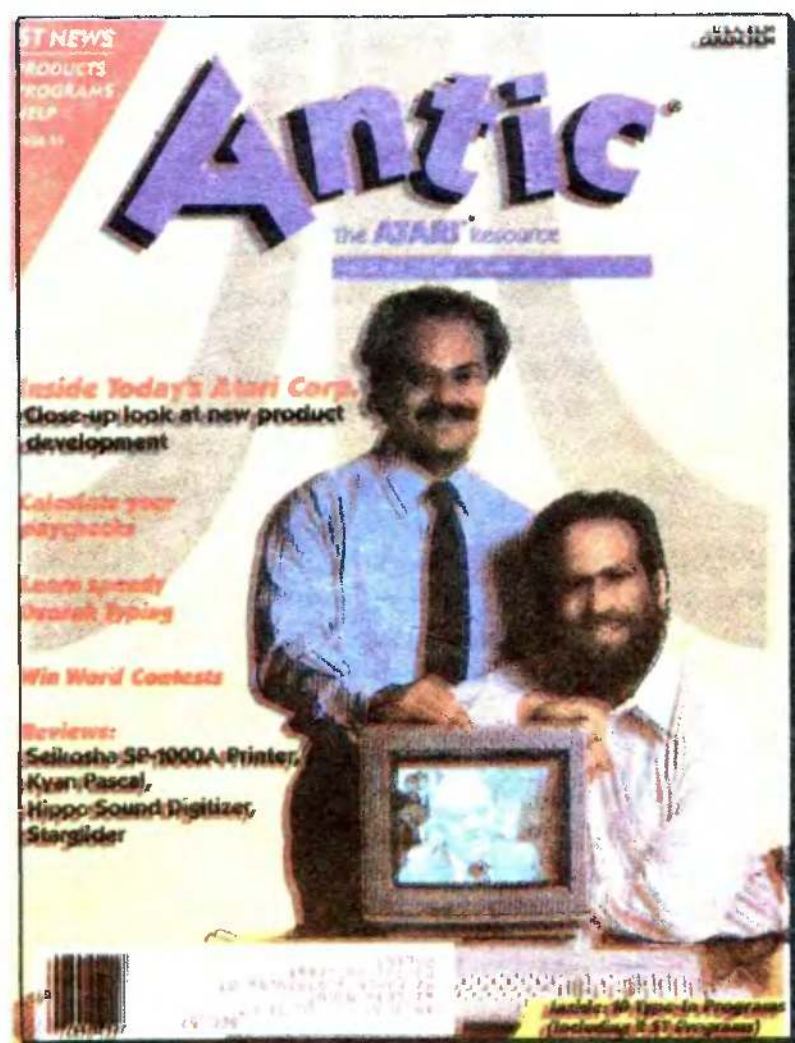
ności programowania. Opracowanie ciekawego scenariusza to połowa sukcesu. Na początek można skorzystać z kilku pomysłów podsuniętych przez autora. Od tej chwili czytelnik asystuje przy powstawaniu nowej gry przygodowej, opartej na razie na specjalnie uproszczonym schemacie. Poznajemy przy tym podstawowe pojęcia opisujące występujące w niej elementy, uczymy się przekładać fantazyjną fabułę na precyzyjne wyrażenia zrozumiałe dla komputera. Stykamy się również z problemem zbudowania odpowiednich struktur danych definiujących mapę pomieszczeń, w których rozgrywa się akcja oraz rozmieszczenie i właściwości różnych przedmiotów. Obok wielu innych łamigłówek rozwiązujemy niełatwe zagadnienia przekazywania poleceń postaciom i informowania gracza o aktualnym stanie gry. Zaskakuje nas pomysłowość doboru słownictwa i reguł składni sprawiające, że język komunikowania się z komputerem jest niezwykle prosty, a jednocześnie zbliżony do naturalnego języka. W książce znajdujemy także odpowiedź na pytanie, w jaki sposób urozmaicić grę poprzez zastawienie różnych pułapek czyhających na bohatera i wprowadzenie punktacji oceniającej jego osiągnięcia.

Na amatorów gotowych programów oczekują trzy kompletne listingi ciekawych gier przygodowych wraz z opisami oraz dwa programy użytkowe: program graficzny oraz program „Venturefix”, który wspomaga generowanie nowych gier przygodowych.

Lektura książki dowodzi, że każdy może stać się twórcą ciekawego programu, przeżywając przy tym niezwykłą przygodę intelektualną. Przystępnie i szczegółowo omówiona metoda programowania wraz z dokładnie wyjaśnionymi gotowymi podprogramami dają solidną gwarancję sukcesu jeżeli zapragnęlibyśmy spróbować własnych sił.

(j.j.)

**Jörg Walkowiak. Adventures und wie man sie auf dem Atari 600XL/800XL programmiert, Data Becker Buch, Düsseldorf 1984. Cena 39,- DM.**



**Dźwięcznie** brzmiąca nazwa ANTIC znana jest wszystkim, którzy mieli styczność z komputerami Atari. Nikt więc nie będzie miał wątpliwości, że pismo mające na okładce właśnie taki tytuł poświęcone jest wyłącznie tematyce związanej z Atari.

Miesięcznik „Antic — The Atari Resource” wydawany jest w San Francisco w Stanach Zjednoczonych, ale dociera również i do Polski. Zawiera

on ponad 100 stron, na których znajdziemy wszystko, co tylko może zaciekać sympatyka Atari. Znajdują się w nim wiadomości o wszelkich nowościach pojawiających się na rynku sprzętu i oprogramowania. Czytelnik sięgający po to czasopismo uzyska dokładne informacje o możliwościach zakupu samego komputera i wszystkich współpracujących z nim urządzeń. Opisywane są także rozwiązania techniczne zastosowane w komputerach rodziny Atari.

Znaczną ilość miejsca przeznaczona na prezentację oprogramowania. W „Antic-u” ukazują się systematycznie materiały dotyczące nowych gier i programów użytkowych przydatnych w różnych dziedzinach życia. Są to dłuższe artykuły opisujące szczegółowo dany program i zakres jego zastosowania, bądź króciutkie wzmianki informujące o jego treści. Przejrzenie stron „Antic-a” pozwala dokonać porównania oprogramowania i ułatwia podjęcie decyzji o zakupie.

Bardzo dużo miejsca w czasopiśmie zajmują wydruki programów do samodzielnego wpisania (8—12 w każdym numerze). Znajdują się wśród nich gry i programy narzędziowe. Część ta adresowana jest do wszystkich interesujących się techniką programowania, ponieważ umożliwia podejrzenie wielu ciekawych sztuczek i zwiększenie możliwości komputera. Jeśli ktoś dojdzie do wniosku, że przepisywanie programów jest zbyt żmudne, to wcale nie musi się męczyć. Wystarczy po prostu zamówić dyskietkę towarzyszącą każdemu numerowi pisma (często zawiera ona dodatkowy program nie umieszczony w piśmie).

Każdy numer „Antic-a” ma oprócz stałych rubryk jeden specjalny temat wiodący, który podawany jest zawsze na grzbiecie okładki. Przyczynia się to bardzo do uatrakcyjnienia pisma i jest dużym ułatwieniem dla Czytelnika poszukującego jakichś konkretnych informacji.

(j.j.)

## CENY SPRZĘTU W USA

(w \$, w sierpniu 87)

### KOMPUTERY:

Atari 800XL	69—80
Atari 65XE	88—100
Atari 130XE	129—140
Atari 520ST	400—516

### MONITORY (do ST):

SM124 (czarno-biały)	129—168
SC1224 (kolor)	298—329

### MAGNETOFONY (do XL/XE):

XC11	38
------	----

### STACJE DYSKÓW:

Atari 1050 (XL/XE)	130—160
Happy 1050 (XL/XE)	285
Indus GT (XL/XE)	180—189
Atari SF314 (ST)	208—220
Atari SF354 (ST)	129—150

### TWARDE DYSKI:

Supra 20 MB (XL/XE)	679
Atari SHD204 20 MB (ST)	580—650
Supra 10 MB (ST)	549—630
Supra 20 MB (ST)	599—640
Supra 30 MB (ST)	779—830
Supra 60 MB (ST)	1790

### DRUKARKI:

Atari 1020 (XL/XE)	25—30
Atari 1025 (XL/XE)	115
Atari 1027 (XL/XE)	100—129
Atari XDM121 (XL/XE)	209
Atari XMM801 (XL/XE)	180—200
Atari SMM804 (ST)	180—200

### INTERFACE:

Atari 850 (XL/XE)	109—119
MPP 1150 (XL/XE)	40—45
Microprint (XL/XE)	30—40
P:R:Connection (XL/XE)	60—63

### MODEMY:

Atari XM301 (XL/XE)	38—45
Atari 835 (XL/XE)	25
Anchor Volksmodem (XL/XE)	129
Supra 1200AT (XL/XE)	139—150
Atari SX300/1200 (ST)	100
Anchor VM520 (ST)	129—139
Supra 1200ST (ST)	139—150

### INNE:

Pióro świetlne Atari	30—38
Tabliczka graficzna Atari	39—43

### ZESTAWY:

800XL, XC11, 2 joysticki	140
800XL, 1050, 1025, joystick	329
130XE, 1050, 1027	378
130XE, 1050, XMM801	448
130XE, 1050, Star NP-10	480
520ST, SF354, SM124	460—569
520ST, SF354, SC1224	725—779
1040STF, SM124	699



Te listy nadeszły do „Bajtka”, do Marcina Waligórskiego. Ponieważ problemy w nich poruszane dotyczą komputerów Atari, więc odpowiadamy na nie w dodatku specjalnym.

Program z artykułu „Nie bój się przerwań (2)” nie chce mi prawidłowo działać. Aby sprawdzić jego działanie dopisałem instrukcję:

```
60 ? 2,9,34:GOTO 60
```

Po kilkakrotnym naciśnięciu BREAK program „zepsuł się”. Dlaczego?

Czy mógłby Pan napisać program, aby po ładowaniu instrukcją CLOAD sam uruchamiał się i nie dał się zatrzymać? Czy trzeba koniecznie przez naciśnięcie klawiszy START i OPTION. Jeśli tak, to proszę o napisanie, jak napisać program ładowany przy naciśnięciu w/w klawiszy, samouruchamiający się oraz niezatrzymawalny.

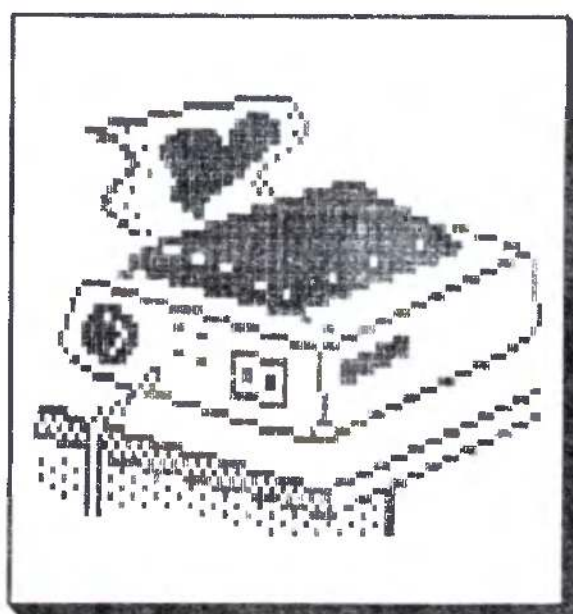
Janusz Baczyński  
ul. Tatrzańska 42  
25-550 Kielce

Program podany w „Bajtku” napisany został w najprostszej możliwej wersji. Nawet przy pobieżnej analizie widać, że podczas przerwania nie zostaje zachowana wartość rejestru Y. Dlatego też po kilkakrotnym wywołaniu przerwania program się zawiesza. Aby temu zaradzić należy w przerwanym dodać na początku rozkazy TYA, PHA i na końcu (przed PLA) rozkazy PLATAY. W programie BASIC-a trzeba więc zmienić linie 10 i 20 następująco:

```
10 FOR I=1536 TO 1559:READ  
A:POKE I,A:NEXT I
```

```
20 DATA 152,72,160,0,185,18,6,145,88,  
200,192,6,208,246,104,168,104,64,38,  
41,39,33,1,0
```

Program zapisany instrukcją CLOAD nie może uruchomić się sam. Aby program BASIC-a sam się uruchamiał, musi być poprzedzony na taśmie krótką procedurą ładującą w języku maszynowym i wczytywany przez włączenie komputera z wciśniętym klawiszem START (bez OPTION — ten klawisz wyłącza interpreter BASIC-a). Procedury tej nie trzeba pisać samodzielnie, można ją skopiować z jakiegoś gotowego



programu, np. „SAM-RECITER” (pierwszy blok). Programy wczytywane przy wciśniętych klawiszach START i OPTION muszą być napisane w całości w języku maszynowym. Do tego celu należy użyć dowolnego assemblera. W obu przypadkach zabezpieczenie przed zatrzymaniem musi być umieszczone w programie.

Mam „w głowie” kilka gier. Niestety utknąłem w pewnym momencie. Dany jest program:

```
10 PRINT "Jeśli masz duszę podróżnika,  
to wyrusz w"  
20 PRINT "—PODRÓŻ PRZEZ  
ŚWIAT—"  
30 PRINT "Naciśnij (A) lub (E)"  
40 PRINT "A=Afryka"  
50 PRINT "E=Europa"  
60 INPUT AS
```

Jak powinny wyglądać linie 70 i 80?

Jacek Jakubiak  
ul. Kopernika 13/26  
66-530 Drezdenko

Przy zastosowaniu tu rozwiązania konieczna jest przede wszystkim deklaracja ciągu AS przed jego użyciem, np. 5 DIM AS(1). Wtedy linie 70 i 80 powinny być następujące:

```
70 IF AS = "A" THEN xxx:REM xxx =  
numer linii  
80 IF AS<>"E" THEN 60
```

Dalej od linii 90 musi znajdować się część programu dotycząca Europy, a od linii xxx — część dotycząca Afryki. Ponieważ instrukcją INPUT wymaga zakończenia wpisywanego ciągu znakiem RETURN, w takim przypadku korzystniejsze jest inne rozwiązanie:

```
60 OPEN #1,4,0,"K":GET #1,X:CLO  
SE #1  
70 IF CHR$(X)="A" THEN xxx  
80 IF CHR$(X)="" THEN 60
```

Przy takiej sekwencji instrukcji wystarczy naciśnięcie jednego klawisza, bez konieczności potwierdzania klawiszem RETURN.

Piszę do Was w imieniu użytkowników stacji dysków Atari 1050. Zakupu tego sprzętu dokonałem w sklepie Pewexu i doznałem pewnego rozczarowania. Dlaczego? Stacja zaopatrzona była w DOS 3.0 — bez opisu komend! Załączona instrukcja zawierała ogólnie opisany DOS 2.5, jako ten lepszy. Droga prywatną wszedłem w posiadanie DOS 2.5, lecz na podstawie załączonej instrukcji trudno, a raczej nie sposób w pełni wykorzystać pamięć dyskową.

Paweł Elminowski  
ul. Chopina 18/5  
62-065 Grodzisk Wlkp.

Stacje dysków rozprowadzane przez Pewex są wyposażone w DOS 3.0 z bardzo prozaiczną przyczyną — jest on najtańszy. Oczywiście brak instrukcji jest poważnym niedopatrzeniem. Trudno mi się jednak wypowiadać w tej sprawie, ponieważ do nabytej przeze mnie stacji był dołączony opis DOS 3.0 w języku angielskim (wystarczająco dokładny). Ponieważ podobne kłopoty mają liczni użytkownicy, zamieszczamy opis funkcji DOS 2.5.

Na zakończenie drukujemy list, jakich chcielibyśmy otrzymywać jak najwięcej.

Po przeczytaniu artykułu „Kruczki i sztuczki” z nr 3/87 postanowiłem podzielić się z Wami moimi osiągnięciami w zakresie Atari. Oto one:

```
10 OPEN #1,4,0,"E":REM OTWARCIE  
EDYTORA EKRAŃOWEGO DO CZYTA  
NIA  
20 INPUT #1,A:REM WPROWADZENIE  
DANYCH
```

Program ten pozwala na wprowadzanie danych instrukcją INPUT bez drukowania???

```
10 OPEN #1,4,0,"E":DIM LINIAS  
(124),RS(3):RS="RUN"  
20 PRINT "":INPUT #1,LINIAS:REM TU  
NALEŻY WPROWADZIĆ LINIE PRO  
GRAMU  
30 CLOSE #1:CLOSE #2  
40 DLUG=LEN(LINIAS):LINIAS$  
(DLUG+2)=RS  
50 OPEN #2,8,0,"C":PRINT #2;LINIAS:  
CLOSE #2  
60 NEW
```

Ten program zapisuje linię w BASIC-u LINIAS\$ na taśmie. Na koniec zmiennej dodawana jest instrukcja RUN w trybie bezpośrednim powodująca samoczynne uruchomienie wpisanej linii po odczytaniu jej z taśmy instrukcją ENTER "C:". Linia jest traktowana jako tekst, a

więc nie jest kontrolowana jej poprawność (trzeba uważnie wpisywać).

POKE 731,1 — wyłączenie „clicku” klawiatury  
POKE 729,0 — wyłączenie powtarzania naciśniętego klawisza  
POKE 703,4 — w trybie "0" dostępne są tylko 4 dolne linie

Mam również kilka pytań:

1. Czy jest możliwe odzyskanie uszkodzonego zapisu na taśmie przy występowaniu błędów numer 143?
2. Czy jest możliwe wyzwanie programu w BASIC-u dłuższego niż 37902 bajty (zapis jest możliwy)?
3. Czy jest możliwość skopiowania gier całodyskowych (Beach Head) na taśmę? Jeśli tak, to co należy wykonać?
4. Czy cena stacji dysków 1050 w Pewexie spadnie w najbliższym czasie?

Maciej Czarnota  
ul. O. Wysockiego 18/142  
03-388 Warszawa

Do tych „sztuczek” chciałbym tylko dodać dwie uwagi. W drugim programiku wpisywana linia musi rozpoczynać się numerem, jak normalna linia programu. Jeśli pominiemy numer linii (tak jak normalnie to się robi w trybie bezpośrednim), to nie jest konieczne dodawanie na końcu instrukcji RUN.

POKE 703,4 działa tak, jak otwarcie ekranu w trybie "0" z oknem tekstowym. Niestety górna część ekranu jest niedostępna zarówno dla instrukcji PRINT, jak i dla PRINT #6. W celu umieszczenia w niej jakiejś treści należy użyć instrukcji POKE lub najpierw coś wpisać i dopiero po tym użyć POKE 703,4.

Teraz kolej na odpowiedzi:

1. Prawdopodobnie nie ma takiej możliwości. Należałoby odczytać w całości uszkodzony program, zapisać go na dysku i dopiero wtedy poprawić błąd. Jedynym znanym mi programem, który potrafi przeczytać uszkodzony program jest CASDUP. Niestety nie ma on możliwości zapisu na dysku.
2. Nie. Wystąpi błąd 19 — PROGRAM TOO LONG (program zbyt długi).
3. Jest to możliwe pod warunkiem, że cały program zmieści się jednorazowo w pamięci komputera (np. „Kennedy Approach”). Wymaga to jednak dokonania znacznych zmian w programie.
4. Z takim pytaniem należałoby się zwrócić do Pewexu. Oczywiście ze zrozumiałych względów (tajemnica handlowa) odpowiedzi się nie uzyska.

Wojtek

W pierwszych słowach mojego listu chciałbym zaznaczyć, że kupiłem Atari 130, stację 1050 i drukarkę 1029. Kupiłem te wszystkie cudowne z oszczędności i snobizmu, a także, żeby sobie ułatwić pracę. Miałem bowiem fatalny zwyczaj trzykrotnego pisania każdego tekstu. Dziś piszę, a raczej drukuję raz a dobrze. Oszczędność czasu — minimum 25 procent. Przeliczając to wszystko na palcach, (na komputerze nie umiem), odnoszę wrażenie, że wydatek zwróci się mniej więcej po 3 latach. Gdyby wszystkie inwestycje były tak rentowne...

Mówiąc szczerze, jest to jedyna prognoza, która mi się sprawdziła. Jako neofita i kompletny laik sądziłem na przykład, że w miarę szybko uruchomię kilka innych programów użytkowych, potrzebnych mnie i mojej żonie w pracy. Niestety, rozkraczyłem się już przy pierwszym podejściu do Synca i Synfile'a. To znaczy prawdopodobnie dałbym sobie radę, ale dokładne rozpoznanie instrukcji zajęłoby pewnie ze dwa tygodnie. Cóż, przyjdzie wynająć fachowca.

Z perspektywy własnych doświadczeń odradzam także kupowanie w firmie wysyłkowej, która nie ma kooperantów w Polsce. Drukarka przyszła uszkodzona i, choć zapewne udałoby się dokonać wymiany w Londynie, korzystając z usług poczty lub Hartwiga, trwałoby to ze trzy miesiące. Za długo; mnie się już zbyt ręce trzęsły do klawiatury. Naprawiłem ją w kraju, bezecnie wykorzystując znajomości, ale zdaję sobie sprawę, iż nie wszyscy je mają.

Nie sprawdziło się jeszcze kilka innych pomysłów, ale mniejsza o to. Sprawdził się natomiast jeden, ale zasadniczy, żeby kupić Atari. Niemala w tym zasługa ludzi z firmy „Karen”, która prowadzi serwis w Polsce. Panowie Trzmiel i Wencel, czy mnie słyszycie?

Ponieważ nadal nie odróżniam bajtów od bitów, sprzęt oceniam zapewne nieco nietypowo w tak fachowym piśmie. A więc po kolei. Najpierw pobłogosławie radzieckiego producenta telewizora Junost, który dzielnie służy mi za monitor. Mimo często wielogodzinnej pracy oczy nie boją, a to najważniejsze.

Teraz uwagi do produktów pana Trzmiela. Stosuję wersję polską Atari Writera Plus i na ekranie wyglądam całkiem przyzwoicie. Gorzej z wydrukiem, drukarka jest niestety mocno amatorska i stanowi zdecydowanie najgorszy punkt zestawu, choć niekoniecznie z winy producenta. Po pierwsze, już wymagała naprawy. Po drugie, są problemy z taśmą. Nie mogę jej kupić w „Pewexie”, taśmy — NIE MA. Barbarzyństwo. Taśmy regenerowane domowymi sposobami trudniej się przesuwają, w związku z czym od czasu do czasu cała maszyna odmawia współpracy. Po trzecie jakość wydruku jest kiepska i doprowadza do pasji maszynistki, które muszą to, co tak pracowicie tworzą, jeszcze raz przepisywać.

Polskie litery wymagają wciskania inwersji i kontroli, i jest to niestety kłopotliwe, ale jak twierdzą fachmani prościej się nie da także w innych komputerach i edytorach. Nie ułatwia pracy klawiatura, która wymaga bardzo precyzyjnego przykładania paluchów do klawiszy, zwłaszcza inwersji.

Stacja dysków działa bez zarzutu, gorzej z krajową energetyką, która już mi raz zafundowała w środku lata wyłączenie prądu.

Kochani, nic więcej wam nie powiem, bo jako laik, nie mam pojęcia, czy przerwania mają to co trzeba i czy zegar tyka właściwie. Na jednym dysku mam edytor, na drugim kilka gier. Z grami było o tyle zabawnie, że tylko jedna dała się zrozumieć i opanować w ciągu dwóch minut — leci sobie samolot, strzela i do niego strzelają. Gratulacje za wypełnienie misji otrzymałem mniej więcej po setnej próbie i czar przysł. Profilaktycznie innymi grami się nie zajmuję, bo to złodziej czasu.

Niestety, moja niespełna dwuletnia córka już wykazuje zainteresowanie „komputelkiem”, co napawa mnie poważnym niepokojem. Zdaje się, że jednak będę musiał się czegoś nauczyć, logo, albo innego bejzyka, wykorzystywania grafiki, grania. Oj, niedobrze, niedobrze.

Pozdrawiam, P.A.



# refleks

**NASZA  
OFERTA!!!**



**ASCOM TECHNOLOGIES**  
(FAR EAST) PTE LTD

## **PWPO-T „Refleks” Sp. z o.o. informuje,**

że działa jako wyłączny przedstawiciel serwisowy na zasadzie zawartego kontraktu z ASCOM TECHNOLOGIES (FAR EAST) PTE LTD. Na zakupiony w tej firmie sprzęt wydawane jest w Polsce świadectwo jakości i udzielana jest roczna gwarancja, w czasie której funkcje gwaranta sprawuje na zasadzie wyłączności PWPO-T „REFLEKS”. Sprzęt zakupiony w ASCOM po odebraniu przesyłki przez użytkownika jest testowany i sprawdzany bezpłatnie w PWPO-T „Refleks” Sp. z o.o.

### **UŻYTKOWNIK OTRZYMUJE TYLKO DOBRY SPRZĘT!**

Ponadto „Refleks” udzieli Państwu wszelkich dodatkowych informacji zarówno handlowych, jak i technicznych (katalogi, cenniki itp.).

Kontakt: **Przedsiębiorstwo Wdrażania Postępu Organizacyjno-Technicznego „Refleks” Sp. z o.o. Dział Importu, 02-051 Warszawa, ul. Glogera 1 tel. (02) 659-20-41, (02) 659-39-22 tlx 817530 ref pl.**

Wysyłkowo z firmy ASCOM TECHNOLOGIES (FAR EAST) PTE LTD otrzymacie Państwo sprzęt mikrokomputerowy wysokiej jakości i w krótkich terminach dostawy:

#### **Oferta po atrakcyjnych cenach:**

- kompletne zestawy mikrokomputerów PC/XT 6/8/10 MHz, PC/AT 8/10/12 MHz, PC/38612/16/20 MHz oraz inne, jak np. mikrokomputery przenośne i najnowsze typy profesjonalnych mikrokomputerów,
- pełny asortyment kart CSKD, wyposażenia i akcesoriów umożliwiających samodzielne zbudowanie mikrokomputera lub rozszerzenie zestawu już posiadanego (karty główne, grafiki, kontrolery, karty obsługi wejść/wyjść, kable, obudowy, klawiatury, zasilacze),
- pełny asortyment urządzeń zewnętrznych, takich jak: monitory monochromatyczne i kolorowe (szeroka gama typów o różnej rozdzielczości), pamięci taśmowe, pamięci na miękkich dyskach i napędy dysków twardej (o bardzo dużej pojemności i krótkim czasie dostępu), różne typy drukarek firm: EPSON, CITIZEN, STAR, PANASONIC, Amstrad, różne typy ploterów i digitizerów,
- **nośniki magnetyczne,**
- **inne wyposażenie w środki techniki biurowej,**
- **urządzenia i przyrządy elektroniczne,**
- **urządzenia techniki video,**
- **elementy i podzespoły elektroniczne.**

ASCOM TECHNOLOGIES/FAR EAST/PTE LTD

Republic of Singapore

45 Genting 05-02 Genting Warehouse Complex Singapore

1334 Republic of Singapore.

Przedsiębiorstwo Wdrażania Postępu Organizacyjno-Technicznego



Sp. z o.o.

K-185



## ATARI?

**Wielu z Was na pewno zdziwi ten tytuł. Odnieść go jednak można do gier typu arkadowego.**

Cechą, która różni je od innych jest skromna grafika. Należy pod tym rozumieć to, że wszystkie dostępne dla gracza poziomy są z reguły na jednakowym tle. Obraz na ekranie telewizora bądź monitora pozostaje stały, a zmieniają się jedynie warunki zewnętrzne takie jak: czas, którym dysponuje gracz; ilość przeciwników, z którymi ma do czynienia itp. Stopień trudności wzrasta wprost proporcjonalnie do wzrostu poziomu, na którym aktualnie się znajdujemy.

Drugą cechą charakterystyczną dla tego typu gier jest to, że komputer uczy nas precyzji i perfekcji. Kiedy rozpoczynamy grę stawia nam zadania proste i łatwe do realizacji. Kiedy już wydaje się, że jesteśmy bardzo dobrzy, przechodzimy szczebelek wyżej i zabawa rozpoczyna się od nowa.

Gry arkadowe oddziałują na cechy, które leżą w charakterze każdego człowieka: ambicje i dążenie do doskonałości. Na pewno trudno jest dorównać maszynie zwłaszcza takiej jak ATARI — ale zawsze można spróbować rywalizacji.

Na początek proponuję bardzo prosty program firmy KAY ENTERPRISES CO. pod tytu-

łem **K-RAZY SHOOTOUT** zajmujący zaledwie 67 sektorów na dyskietce i 1/10 kasety magnetofonowej C-60. Na podstawie tej gry każdy zrozumie na czym polegają gry arkadowe. K-RAZY SHOOTOUT ma nieskomplikowaną grafikę. Składają się na nią plansze z zaznaczonymi na nich fragmentami murów: załamaniami, skrytkami itp. Gra polega na wymianie strzałów między Dzikim Joe, a jego trzema przeciwnikami — rewolwerowcami. Jeżeli choć jeden z nich ginie, natychmiast na jego miejscu pojawia się następny przeciwnik. W każdym etapie jest ich określona ilość. Jeżeli nasz bohater wygra wszystkie pojedynki — przechodzi do następnej rundy. Kolejny etap w górę to większa ilość przeciwników, którzy coraz szybciej do nas strzelają i coraz szybciej poruszają się po planszach, na których toczy się gra. Komputer jest na tyle wyrozumiały, że co 10000 punktów, zdobytych przez nas, daje premię — dwa dodatkowe „życia”. W każdej rundzie dysponujemy jednakowym czasem, aby pozbyć się nieprzyjaciół. Ta prosta z pozoru gra wymaga nie lada zręczności od jej uczestnika i możecie mi wierzyć — naprawdę wciąga.

Inną klasyczną grą arkadową jest **POLE POSITION** firmy NAMCO SOFTWARE. Program ten należy w dalszym ciągu do najlepszych wyścigów samochodowych na ATARI. Daje każdemu możliwość kierowania samochodem wyścigowym i jazdy z prędkością na jaką nigdy by się nie zdobył na jawie. Do wyboru mamy trzy różne tory jazdy i i rywalizujemy z pojazdami sterowanymi przez komputer ATARI. Aby móc uczestniczyć w wyścigu głównym, należy bezwypadkowo i w określonym czasie przejechać okrążenie próbne. Im krótszy będzie czas naszego przejazdu, tym lepszą pozycję wyjściową

zajmiemy na starcie w wyścigu głównym. Komputer pozwala wybrać tor, po którym chcemy jechać oraz ilość okrążeń. Samochodem kieruje się za pomocą joysticka.

Trzecią i ostatnią grą arkadową, którą przedstawię to **Mr Dig** produkcji Microdeal Software. Gra ta kumuluje w sobie wszystkie zalety tak znanych użytkownikom ATARI (i nie tylko) gier jak: Pac-Man czy Boulder Dash. Od gracza wymagana jest jak zwykle, gdy chodzi o gry na ATARI, kompilacja sprawności, ambicji, wytrzymałości oraz szczypty poczucia humoru. Tytułowy bohater czyli Mr Dig ma za zadanie pozbiierać wszystkie wiśnie pojawiające się na ekranie. Przyczyna jego postępowania jest bardzo prozaiczna. On przepada za tymi owocami. Ponadto są mu potrzebne do wyrobu tortów, ciast, puddingów i budyni, które sprzedaje — zarabiając w ten sposób na życie. W realizacji celu przeszkadzają Panu Dig spadające co jakiś czas jabłka, których nadmierny zapas zgromadził w magazynie oraz różnego autoramentu potwory i monstra. Oczywiście Pan Dig nie jest bezbronny. Ma do dyspozycji tajemniczą piłkę POWERBALL, dzięki której może udaremniać nieczne zamiary swoich wrogów.

Program ten doskonale oddaje istotę gier arkadowych. Aby to udowodnić wymienię nazwy poszczególnych poziomów. Są to: dla przedszkolaków, dla dzieci z pierwszej A, bardzo łatwy, łatwy, trochę trudny, zaczynają się kłopoty, nie lubię komputera, jestem coraz lepszy, bardzo ciężki, poziom dla masochistów.

Mam nadzieję, że teraz każdy użytkownik komputera ATARI odróżni bez problemu gry arkadowe od innych.

*Sergiusz Piotrowski*



**Któż** nie słyszał o takich grach jak „Rambo”, „Commando”, „Green Beret”, „Gunlaw”. To gry dla zręcznych, szybkich i „twardych” ludzi.

Także „Who Dares Wins II” reprezentuje ten najpopularniejszy typ gier (gry Arcade). Gra powstała w oparciu o film pod tym samym tytułem, ale nie jest zgodna z nim treściowo. Ci, którzy mają już wprawę nabytą w grze „Gunlaw” mogą być spokojni o swój wynik. Niemniej program jest wyjątkowo trudny i trochę mało dokładny, ale na pewno nie odstraszy amatorów mocnych wrażeń. Na drodze żołnierza (bohaterem gry jest były komandos) staje cała armia świetnie uzbrojonych obrońców. Droga prowadzi poprzez piaszczyste bezdroża, bagniste, podmokłe łąki, przecina tory kolejowe i ruchliwe szosy. Z powietrza cały teren, który jest bazą rebeliantów przeciwnych rządowi, jest broniony przez szwadron kawalerii powietrznej na śmigłowcach Bell UH-1 „IROQUOIS”. Zadaniem por. Marka „Dark Sceptre” Singeltona jest dostanie się za wszelką ceną do głównego magazynu rebeliantów, gdzie składowana jest skradziona z wojskowego transportu broń nuklearna. Nie trzeba wyjaśniać, że broń nuklearna posiadana przez rebeliantów nie tylko zagraża rządowi, ale też pokojowi na całym świecie.

Czy misja „Dark Sceptre” będzie udana zależy tylko od was. Jego niepowodzenia są waszymi, jak i przejście każdej nowej strefy jest też waszym udziałem. Każdy może sprawdzić jak twardym jest człowiekiem.

Zagrać warto, ale UWAGA!, joystick będzie do wymiany.

**Ocena (w skali od 1 do 10):**

GRAFIKA 4

DŹWIĘK 6

SENS GRY 3

(+++)

**Producent ALLIGATA, TYNESOFT (wersja Atari)**  
**Autor: Brian Joblins**



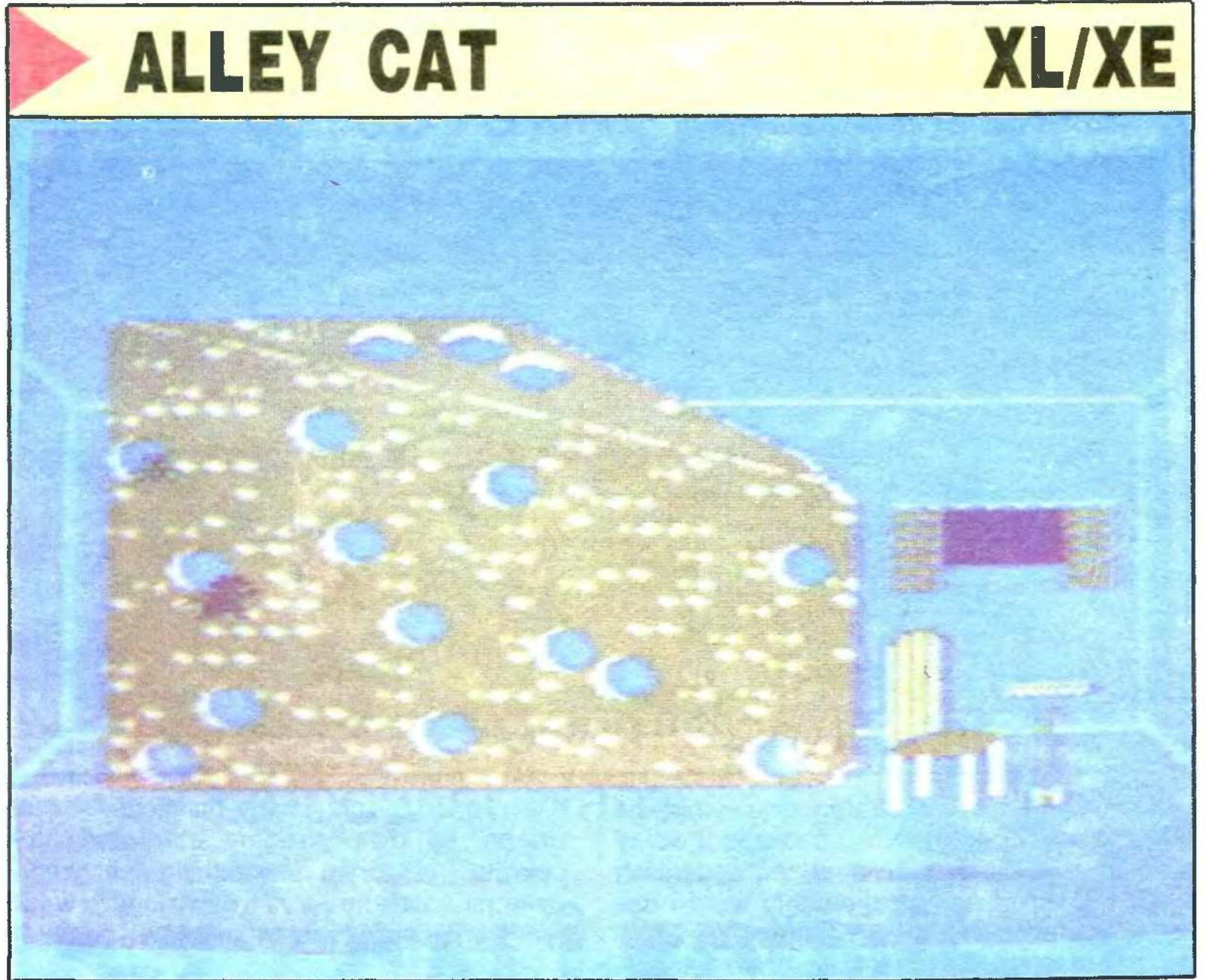
**Jakie nudne** jest w dzisiejszych czasach życie bezdomnego kota, szczególnie, gdy nie ma on przyjaciół a tylko samych wrogów. Ale można to nudne życie jakoś urozmaicić. Który porządny kot odrzuci możliwość bezkarnego rozrabiania wśród ludzi. Każdy lubi gryźć, szarpać, drapać z całych sił, aby wyładować swoją energię. A jeżeli dodać do tego fakt, iż w trakcie zabaw można poznać atrakcyjną kotkę, to znaczy że mamy szczęście.

Jeżeli chcesz spróbować, czy masz szczęście zajrzyj na Kocią Aleję, i zagraj z nami.

Skoki po sznurach z bielizną, łapanie myszy, czy skoki ze śmietnika to tylko preludeum do prawdziwej zabawy. Przy odrobinie szczęścia można wtargnąć do mieszkania człowieka i tam kontynuować zabawę.

W jednym z mieszkań można łapać złote rybki w akwariu (uwaga na glizdowate robaki — są bardzo niesmaczne, a wręcz toksyczne), ganiać się ze szczotką gospodyni i brudzić do woli. W innym można zrzucić wazony z kwiatami na podłogę i bezkarnie uciec przez okno. W jeszcze innym schrupać kanarka na obiad. Można też penetrować wielką górę sera w poszukiwaniu smacznych myszek. Można podjadać psom z misek pyszne mleko (ale tak by ich nie obudzić). Można, można, można wiele, każdy znajdzie zabawę dla siebie i na pewno będzie zadowolony z zaproszenia.

Pamiętajcie więc adres: „Alley Cat” — Kocia Aleja. Ale uwaga: jest strzeżona przez Fre-



da. Stare, ale zawzięte to psisko. Są jeszcze koty z innej dzielnicy, i trochę przeszkadzają, ale to tylko zwiększa atrakcyjność kociego lunaparku.

A gdy poznacie już kotkę Sally, to ... Co wam będę mówił, sami zobaczycie. Wpadnijcie do nas. Wasz stary „Miau”

**Ocena (w skali od 1 do 10):**

GRAFIKA	7
DŹWIĘK	7
SENS GRY	7

(+++)

**Producent: Synapse Software**  
**Autor: Bill Williams**



**Najczęściej** „komputeryzowaną” grą planszową są szachy. Każdy rok przynosi kilka nowych komputerowych wersji tej gry. Ostatnim programem, który ukazał się na giełdzie jest „CHESSMASTER 2000”. Ponieważ nie jestem wystarczająco zaawansowanym szachistą, aby przetestować możliwości tego programu, to mogę tylko powiedzieć, że pokonał on w jednej grze „COLOSSUS CHESS 3,0” i „ODESTA CHESS 7,0”. Jedna gra nie jest wystarczającym dowodem wyższości, lecz na tej podstawie można stwierdzić że „CHESSMASTER 2000” reprezentuje co najmniej taki sam poziom. Po-

przedstawienie gry przez cały czas, także kiedy gracz wykonuje ruch „CHESSMASTER 2000” przyspiesza znacznie rozgrywkę i jest szybkim oraz groźnym przeciwnikiem.

O wartości programu szachowego decyduje obecnie łatwość i przyjemność grania, a nie jego siła i umiejętności — zawsze można z nim przegrać na najwyższym poziomie. „CHESSMASTER 2000” jest tu zaledwie zadowolający. Obraz planszy może być dwu- lub trójwymiarowy, plansza może być także obracana i oglądana pod czterema różnymi kątami. Niestety obraz trójwymiarowy jest użyteczny tylko na dobrym monitorze — nie-

kiedy pokrywające się bierki są trudne do odróżnienia. Również na dwuwymiarowym obrazie mogą wystąpić kłopoty z rozróżnieniem piona i gońca.

Na dodatkowym dysku zawarte jest sto klasycznych rozgrywek i problemów szachowych do powtórzenia i analizy. Cenną zaletą jest także możliwość wydrukowania przebiegu partii na drukarce. Z możliwości opisywanych na opakowaniu na 8-bitowych Atari niedostępny jest zegar szachowy. Ten brak jest dość poważny. Jeżeli chesz rozegrać partię turniejową, to musisz mieć własny zegar i przełączać go za siebie i za komputer. Na plus należy natomiast zaliczyć 20 poziomów trudności zamiast wymienionych dwunastu.

„CHESSMASTER 2000” ma bardzo przyjemną broszurkę o historii szachów i dołączonych klasycznych partiach, lecz instrukcja jest źle napisana. Część problemów wynika ze złej organizacji samego programu. Klawisze nie mają mnemonicznego znaczenia, pozycje w menu są mało opisowe, a submenu nie zawsze są logicznie zestawione. Joystick może być użyty do przesuwania bierek, ale nie do wybierania z menu. Poza tym nie można nim wykonywać ruchów ukośnych.

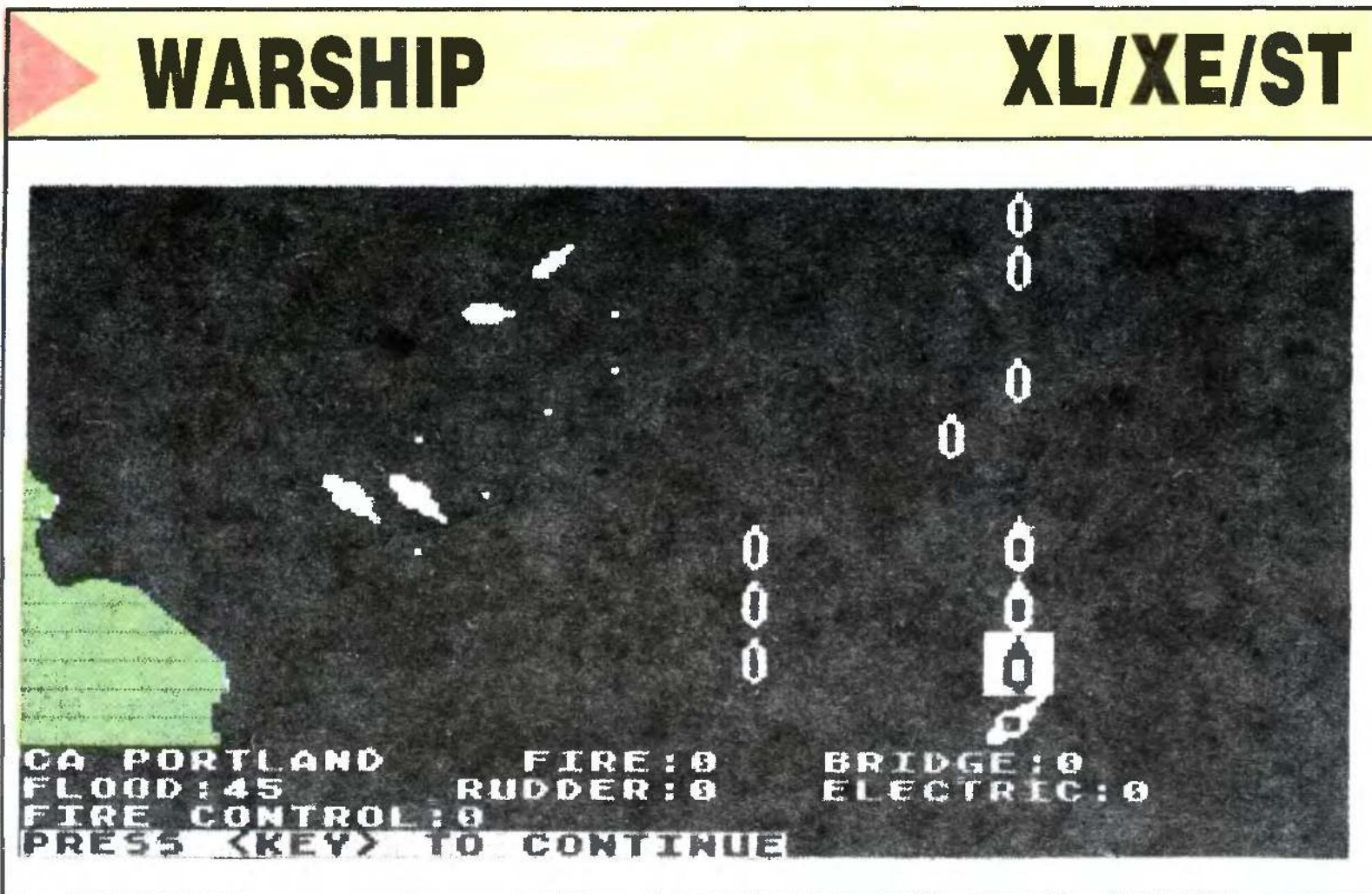
**Ocena (w skali od 1 do 10):**

GRAFIKA	7
OBSŁUGA	4
MOŻLIWOŚCI	8

(ziew)

**Producent: Software Country**





„WARSHIP” jest strategiczną symulacją bitew morskich na Pacyfiku podczas II wojny światowej. SSI umieściła w tej wspaniałej grze wszystkie główne elementy walki i dowodzenia w bitwie. Przy pomocy dodatkowych procedur zaawansowany gracz orientujący się w zagadnieniach wojny na Pacyfiku w latach 1941—45 może skonstruować cały szereg własnych scenariuszy bitew morskich. Będzie to niewątpliwie cenny nabytek do twojej kolekcji. „WARSHIP” jest grą, która wciąga, zmusza do rywalizacji i jeśli tylko zechcesz może być zawsze nową.

W grze znajdują się cztery gotowe scenariusze z II wojny światowej. Trzy z nich odzwierciedlają historyczne bitwy morskie, a jeden zawiera hipotetyczne starcie. Bitwy mogą być rozgrywane przez dwóch graczy lub przez jednego gracza przeciw komputerowi. Można także polecić komputerowi granie za obie strony i tylko śledzić przebieg akcji. Aby gra była bardziej zajmująca, którejkolwiek ze

stron może być przyznany handicap. Przewidziane są dwa poziomy sterowania tryb „okrętu”, w którym dowodzi się poszczególnymi jednostkami i tryb „dywizjonu”, w którym kieruje się zespołem okrętów.

Program uwzględnia wiele czynników wpływających na przebieg walki: obszary lądu, widzialność, uszkodzenia, wielkość okrętów, opancerzenie, manewrowość, uzbrojenie, maskowanie i dokładność ognia. Możesz postawić zastonę dymną dla ukrycia okrętów, lecz radar widzi przez nią równie dobrze, a także powiększa zasięg wykrywania w dzień i umożliwia je w nocy.

Uszkodzenia w bitwie odwzorowują połączone efekty ognia, przecieków, strat sterowności i uzbrojenia oraz zniszczeń mechanizmów napędowych i agregatów prądotwórczych. Trafienie może również spowodować eksplozję komór amunicyjnych, choć zdarza się to bardzo rzadko.

Obie strony mają pod swoim dowództwem siły powietrzne, lecz jedna — nie posiadająca panowania w powietrzu — ma zmniejszoną ilość punktów wsparcia. W grze dotyczącej walki morskiej współczynnik siły lotnictwa nie powinien faworyzować żadnej strony. Na szczęście można wybrać wariant „Neutral” dla stosunku sił powietrznych.

Dzięki dołączonym do programu dodatkowym procedurom można z gotowych elementów zbudować nową grę, lecz nie można zmodyfikować istniejących scenariuszy ani gry przerwanej i zapisanej. Program umożliwia stworzenie mapy dowolnego obszaru świata, który może być przedstawiony w siatce 60x60 pól o boku 1000 jardów. Mapy mogą być zapisywane niezależnie i później modyfikowane.

Każda z walczących flot może mieć do 20 okrętów. Są one wybierane z 53 klas okrętów alianckich i 20 klas okrętów japońskich, lecz wszystkie ich charakterystyki mogą być modyfikowane. Umożliwia to zaprojektowanie dowolnej bitwy morskiej z okresu ostatnich 600 lat. Trzeba jednak wziąć pod uwagę, że gra ma wbudowaną gorszą celność ognia floty „japońskiej”.

To wstyd, że „WARSHIP” nie wykorzystuje graficznych możliwości Atari. Inne programy SSI oferują wysoką rozdzielczość, wielokolorową grafikę i delikatny przesuw obrazu. Ułatwia to orientację i czyni grę bardziej przyjemną. Mimo tej wady „WARSHIP” jest grą godną polecenia.

Podręcznik jest kompletny i dokładny. Oprócz zwykłego opisu programu pokazuje również jak komputer oblicza zależności pomiędzy walczącymi siłami. Gdy jednak zaangażujesz się w walkę, to w ferworze bitewnym lepiej mieć prosty i skuteczny plan akcji, niż dokładnie optymalizować swoje działania.

**Ocena (w skali od 1 do 10):**

GRAFIKA	3
WIERNOŚĆ	8
SENS GRY	9

(ziew)

**Producent: Strategic Simulations Inc.**

**Nie ma chyba gracza,** który nie widział choć raz gry „Ścianka”, „Mur Chiński”, „Break Out”, czy podobnej. Najnowszą i najpopularniejszą grą tego typu jest aktualnie ARKANOID. Swoją formą przypomina nieco pierwowzory, ale jest o wiele bardziej skomplikowany.

ARKANOID stał się z początkiem roku 1987 szlagierem „na automaty” (popularne Coin-Op w salonach gier zręcznościowych). Królował do momentu, gdy wyparty go gry na dysku laserowym takie jak „Space Ace”, „Drason's Lair”, „Master of Univers”.

Oprócz prostego (?) odbijania piłeczki od klocków (u góry ekranu) jest jeszcze w grze kilka niespodzianek nie znanych graczom „ścianki”. Po zbitiu któregoś z klocków może spaść literka, która po złapaniu może ułatwić lub utrudnić grę. A oto znaczenia poszczególnych liter:

- „B” — otwarcie bezpośredniego przejścia do następnego poziomu bez konieczności zbijania reszty klocków.
- „C” — możliwość przytrzymania piłeczki na „tapce” i wystrzelenie jej w dowolnym momencie.
- „D” — potrojenie ilości piłeczek
- „E” — dwukrotne rozszerzenie łapki.
- „L” — możliwość bezpośredniego zbijania klocków za pomocą „lasera”.
- „P” — dodatkowa łapka (nowe życie).
- „S” — zwolnienie ruchu piłeczki.

Każda litera działa do momentu złapania następnej (tylko „S” działa czasowo).

Niektóre klocki należy trafić kilka razy, a niektórych nie można zbić. Z dwu otworów u góry ekranu



mogą wychodzić różne „stwory”, które (po odbiciu od nich) zmieniają bieg piłeczki. Gra kończy się po rozbiciu „maski” na 32 poziomie. Powodzenia.

**Ocena (w skali od 1 do 10):**

GRAFIKA	9
DŹWIĘK	7

SENS GRY 4

(+++)

**Producent: IMAGINE**  
**Autor: Mike Lab**



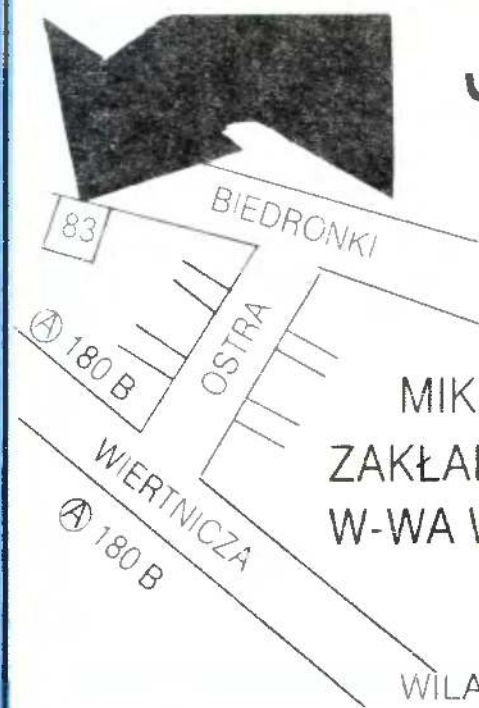
ZAKŁAD DOSKONALENIA ZAWODOWEGO w Łodzi  
ODDZIAŁ TERENOWY w Zduńskiej Woli telefon 61—51 w. 7 (komputery), 63—50 telex 884 257 ZEWAS

**Sklep kupno-sprzedaż Mikrokomputerów i art. elektronicznych, ul. Komisji Edukacji Narodowej 3/5**

- oferuje w ciągłej sprzedaży:
- mikrokomputery różnych typów, w konfiguracjach wg życzeń klienta
  - urządzenia peryferyjne do komputerów, stacje dysków, interfejsy, drukarki itp. wraz z przewodami łączącymi komputer
  - części zamienne do mikrokomputerów
  - programy szkolne i użytkowe
  - sprzęt video, taśmy video i kamery video
  - przyrządy pomiarowe
  - uniwersalne regeneratory kineskopów.

Ponadto informujemy, że produkujemy nowoczesne, funkcjonalne meble na wyposażenie stanowisk komputerowych.

K-218



## JOYSTIK

NAPRAWY PRZERÓBKI NA

MIKROWYŁĄCZNIKI  
ZAKŁAD ELEKTRONICZNY  
W-WA WILANÓW

BIEDRONKI 83  
od 9 do 17

ATARI Cartridge z grami, językami LOGO, BASIC XL, z programami użytkowymi lub zleconymi do 48 KB sprzedam. M.W. Cięci-  
na 37, 34—350 Węg. Górka.

D-181

Programy na ATARI tania wypożyczysz za zaliczeniem pocztowym. „MIKROFAN”  
45—064 Opole 1, skr. poczt. 158.

D-170

Wymienię nagrania i dyski na ATARI 1050. Stanisław Humel, Wieluń,  
Częstochowska 37.

D-185

Bardzo tania — 200 zł — sprzedam programy na ATARI 800 XL, 130 XE, 65 XE. Remigiusz Tywonek, ul. Nowotki 9/2. 58—370 Boguszków-Gorce.

G-129

Zgodnie z wcześniejszymi zapowiedziami

# P.Z. „GALLECH”

z siedzibą w Miechowie serdecznie zaprasza wszystkich zainteresowanych do swojego salonu wystawowego otwartego w każdy dzień roboczy.

Specjaliści naszej firmy prezentują:

- komputery 32-bitowe kompatybilne z IBM PC/AT,
- wielodostęp pod systemem operacyjnym XENIX,
- języki baz danych pod systemem operacyjnym XENIX, (INFORMIX SOL, FOXBASE+ — stu procentowa zgodność ze standardem DBASE III plus)
- oprogramowanie baz danych pracujących w sieciach (SOL BASE, DBASE III plus, CLIPPER AUTUMN 86),
- kompilatory i interpretry języków (C, MS-PASCAL, MS-BASIC, MS-FORTRAN)
- procesor tekstu (Lyrix),
- sieci D-LAN i E-LAN (typu D-LINK i ETHERNET),
- sieciowe systemy operacyjne (IBM PC LAN PROGRAM, D-LINK NETBIOS EMULATOR, D-LINK NETWARE DRIVER, ADVANCED NETWARE 286)

Salon wystawowy mieści się w budynku firmy w Miechowie przy ul. Raclawickiej 31. Prosimy o wcześniejsze telefoniczne uzgodnienie daty przyjazdu, nr tel. 304—57. Miechów.

— SERDECZNIE ZAPRASZAMY —

**UZYTKOWNICY ATARI XL/XE**  
**ATAREX** oferuje TANI  
duży wybór programów  
na kasetach kasetowych oraz dyskietkach  
Szczegółowych informacji po zwołaniu proszę uznać:  
ul: 22 Lipca 17 62-300 WRZESNIA  
**ATAREX**  
ul: 20 Października 42/27 63-000 SRODA WLKP. G-92

## — ATARI —

- przystawki umożliwiające współpracę komputera z dowolnym magnetofonem zapis, odczyt, auto-stop, kopiowanie magnetofon-magnetofon.
- przystawki foniczne z głośnikiem umożliwiające odbiór fonii bez konieczności przestrajania komputera lub telewizora

oferuje  
Zakład Elektroniczny „TURBO”  
Krzysztof Kubiński  
ul. Botaniczna 3/26  
39—300 Mielec

ROCZNA GWARANCJA!  
NAPISZ — ZAOSZCZĘDZISZ DOLARY!  
G-122

## JAK PISAĆ ARTYKUŁY I PROGRAMY

Nie stawiamy jednak żadnych ograniczeń, każdy interesujący materiał na temat Atari, Spectrum i innych popularnych komputerów znajdzie miejsce na naszych łamach. Muszą one spełniać tylko pewne wymagania formalne. Artykuły i opisy programów powinny być w maszynopisie z dużym odstępem między wierszami (ok. 30 wierszy na stronę). Programy należy przysyłać na dyskietkach lub kasetach, gwarantujemy zwrot najpóźniej w ciągu miesiąca. Proszę pamiętać o podaniu sposobu wczytywania — redakcja nie ma etatu dla detektywa.

Programy dotyczące Atari mogą być napisane w dowolnym języku dostępnym dla tego komputera. Ze względu na „Edytor BASIC-

a” programy w tym języku nie mogą zawierać linii o numerach większych od 31999. Dla programów w języku maszynowym oprócz programu ładującego w BASIC-u konieczny jest program źródłowy w formacie asemblera MAC 65. Rysunki mogą być zarówno w tuszu, jak i w ołówku (najlepiej w podziałce 2:1) albo na nośniku magnetycznym w formacie GRAPHICS 8, Micro-Painter lub Koala. Mogą to być także ilustracje np. z katalogów. Pamiętaj o podaniu adresu i telefonu oraz w przypadku opracowań — źródła, z którego korzystałeś.

MASZ POMYSŁ — NAPISZ DO NAS!!!

Wszystkich Czytelników zapraszamy do współpracy przy redagowaniu „Bajtka” i zeszytów dodatkowych. Będziemy zamieszczać nadesłane przez Was gry, programy/użytkowe, sposoby rozbudowy komputera lub niekonwencjonalne sposoby wykorzystania.



**Jak** się dowiadujemy ze źródeł dobrze poinformowanych, na początku przyszłego roku będzie w Polsce seryjnie produkowany modem XM 301 typu "direct connect", tzn. podłączany bezpośrednio do wtyczki gniazda telefonicznego. Szybkość przysyłania informacji: 300 i 1200 bodów/sek. O ile na tę pierwszą szybkość Ministerstwo Łączności wyraziło zgodę, o tyle wartość 1200 nie zostanie chyba w praktyce wykorzystana, głównie z uwagi na słabą jakość naszej sieci telefonicznej.

**R**ównież w 1988 roku planowana jest krajowa produkcja urządzenia XEP 80, które podłączone do Atari 800 XL — pozwala na sprzętowe uzyskanie 80 kolumn (po 24 wiersze) na ekranie, czytelnych również i na monitorze (telewizorze) kolorowym. Już niedługo więc posiadacze Atari będą mieli sposobność uzyskania tego, co do tej pory mogli jedynie otrzymać poprzez żmudne programowanie. Większy opis XEP80 wkrótce.

**Do** 15 lipca Atari oferuje gratisowo przy zakupie 520 ST 4 programy: Neochrome Plus (najnowsza wersja znanego programu graficznego), Animatec (tworzenie krótkich animowanych sekwencji), Macadam Bumper (najlepszy flipper) oraz Wanderer (trójwymiarowa gra zręcznościowa oferowana razem ze specjalnymi stereoskopowymi okularami).

**J**eśli przez przypadek wyłączycie i włączycie szybko ponownie Atari ST, możecie mieć kłopoty. Rozruch „na zimno” może się nie udać, program może się zawiesić, jeśli nie odczekacie na tyle, aby oczyścić pamięć. Dla modelu 520 ST wystarczy sekunda lub dwie, dla 1040 ST — przynajmniej 10. Jeśli po upływie 20 sekund system nadal się nie zgłasza — nie ma rady, naprawa nieunikniona. Oczywiście, nie w każdym egzemplarzu Atari 520/140 ST te objawy występują równie silnie, ale lepiej być ubezpieczonym. Włączajmy komputer powoli!

**C**eny serii ST znowu spadły, co zostało podyktowane wprowadzeniem na rynek z początkiem września nowych Mega ST. I tak: 520 ST kosztuje 550 dolarów (bez monitora), 670 (z monitorem monochromatycznym), 920 (z monitorem kolorowym Philipsa 1424) i 980 (z monitorem kolorowym Atari SC 1224). Model 1040 ST kosztuje odpowiednio drożej: 1230 (monitor mono) i 1340 (monitor kolor).

Przypomnijmy, że Atari 520 ST z monitorem monochromatycznym SM 124, myszką kosztuje w Pewexie 1120 dolarów. Aktualne ceny Atari ST można uzyskać pisząc do firmy Elektronics Export z Londynu, reklamującej się na łamach „Bajtki”.

**Do** firm produkujących sprzęt kompatybilny z IBM dołączyło ostatnio Atari pokazując na targach w Hanowerze model PC 1, kompatybilny wprawdzie w trybie Herkules i EGA, ale posiadający nietypową konstrukcję zewnętrzną, wymagającą podłączenia dodatkowego łącza pośredniego w postaci niedużej skrzyneczki: tylko za jego pomocą komputer może być rozszerzany o dalsze karty.

Poza tym w dalszym ciągu Atari nie zaproponowało taniego — w myśl sloganu „Power without the Price” — monitora EGA. Zapytany przez dziennikarzy dlaczego oferuje technologię (tzn. komputer 16-bitowy klasy IBM), o której zawsze wypowiedział się negatywnie, Jack Tramiel odparł: „Wszyscy wiedzą, że papierosy są szkodliwe dla zdrowia, ale to nie przeszkadza palaczom, by je kupowali.”

## PÓŁ ŻARTEM, PÓŁ SERIO

**Mimo że na łamach „Bajtki” drukowane były programy biorytmów, nie widziałem wersji dla Atari. Proponowany przez mnie program do obliczania biorytmów posiada edycję dla całego żądanego miesiąca.**

Co to są biorytmy, można się dowiedzieć z „Bajtki” nr 12/86 (str. 21), a po szczegóły odsyłam chętnych do książki Jerzego Sikory „Biodiagram prawdę ci powie”.

Program jest prosty i nie wymaga specjalnego objaśnienia.

Teoretycznie, aby obliczyć biorytmy należy policzyć, ile dni upłynęło od dnia urodzenia do żądanej chwili, a następnie liczbę tę podzielić kolejno przez 23, 28 i 33. Reszta z dzielenia da nam kolejny dzień danego biorytmu. W dobie komputeryzacji wpisujemy zamieszczony program, a komputer sam wykona powyższe czynności uwzględniając dzień 29 lutego w latach przestępnych. Obsługa programu polega na podaniu daty urodzenia oraz roku i miesiąca obliczeń biorytmów.

Krótką interpretacją w/g książki J. Sikory „Biodiagram prawdę ci powie”.

BF: BP: BI:

+ + + Dobre samopoczucie, wydajność w pracy fizycznej i umysłowej. Czas udanych transakcji, egzaminów, przedsięwzięcia najwyższej wagi — okres sukcesów.

+ + - Wzmoczona wydajność w pracy fizycznej.

+ - + Nastrój pod pselem — dobra kondycja.

- + + Efektywność umysłowa (uczenie się).

+ - - Skłonność do intryg, upor, wybuchów gniewu.

- + - Bardzo dobry, optymistyczny nastrój, zbudny, ograniczona zdolność twórczego myślenia i logicznego rozumowania. Unikać przemęczenia.

- - + Pełnia sił umysłowych, kiepski nastrój, mała odporność na ból (unikać dentysty).

- - - Regenerować siły, odpoczywać, brać środki wzmacniające.

A więc korzystaj z wyzów!

Stanisław Jawor

**UWAGA: Nie należy wpisywać liter wydrukowanych przed numerem linii programu. NIE SĄ one częścią linii, lecz kodem kontrolnym „Edytora BASIC-a” (zob. str. 3).**

```

CG 4 REM S.J. *** BIORYTMY ***
NE 5 ? CHR$(125)
CR 10 CLR :POKE 710,0
QR 15 DIM A$(11),A(12)
DX 20 POSITION 6,1:?"* * * B I O R Y T M
Y * * *"
QI 25 POSITION 4,4:?" Podaj date urodzenia "
AD 30 POSITION 7,6:?"Rok: ";;INPUT RU
IS 35 POSITION 3,8:?"Miesiac: ";;INPUT N
KD 40 POSITION 5,10:?"Dzien: ";;INPUT D
YD 45 POSITION 6,14:?"BIORYTMY obliczyc:
"
XU 50 POSITION 3,16:?"na"
DU 55 POSITION 6,17:?"Rok: ";;INPUT RB
VV 60 POSITION 2,18:?"Miesiac: ";;INPUT
M
QH 70 ? CHR$(125):POSITION 2,7:?"prosze
czekac..."
VH 80 FOR I=RU TO RB
OR 85 IF INT(I/4)=I/4 THEN C=C+1
IX 90 NEXT I
VU 100 IF INT(RU/4)=RU/4 AND N>2 THEN Q=1
XB 105 IF INT(RU/4)=RU/4 AND M<3 THEN S=1
CB 110 IF INT(RU/4)=RU/4 AND M=2 THEN W=1
IX 115 IF INT(RU/4)=RU/4 AND N=2 THEN H=1
RX 120 FOR I=1 TO 12
CA 130 READ A:A(I)=A
FZ 140 NEXT I
EK 150 FOR I=N TO 12
HW 155 E=E+A(I)
GD 160 NEXT I
CM 165 FOR I=M-1 TO 1 STEP -1
IJ 170 G=G+A(I)
GI 175 NEXT I
MW 180 ND=365*(RB-(RU+1))+E+C+G+1-D-Q-S
IP 190 IF RU>RB OR D>A(N)+H THEN ? CHR$(1
25):POSITION 3,5:?"ZLE DANE":GOTO 510
YT 200 DATA 31,28,31,30,31,30,31,31,30,31
,30,31
NT 210 F=ND-23*INT(ND/23)
VX 220 P=ND-28*INT(ND/28)
RY 230 U=ND-33*INT(ND/33)
KD 235 ? CHR$(125):RESTORE 300
SC 240 FOR I=1 TO 12
HX 250 READ A$
DI 255 IF I=M THEN POSITION 8,3:?"A$
GE 260 NEXT I
LE 265 POSITION 4,1:?"Data urodzenia:
";D;".";N;".";RU;". "
DS 270 POSITION 22,3:?"RB
LH 300 DATA STYCZEN,LUTY,MARZEC,KWIECIEN,
MAJ,CZERWIEC,LIPIEC,SIERPIEN,WRZESIEK,
PAZDZIERNIK,LISTOPAD,GRUDZIEN
QI 305 ? :? :? " 0 0 1 1 2
2 3"
FC 310 ? " 1 5 0 5 0 5
0"
MO 315 ? "-----
---":?"B.F:";
ZX 320 FOR I=1 TO A(M)+W
CB 325 IF F>=23 THEN F=0
ZW 330 IF F>0 AND F<11 THEN ? "+";
PE 335 IF F=0 OR F=11 THEN ? "*";
SL 340 IF F>11 AND F<23 THEN ? ". ";
PP 345 F=F+1
GD 350 NEXT I
LO 360 ? :?"B.P:";
AU 365 FOR I=1 TO A(M)+W
OO 370 IF P>=28 THEN P=0
ME 375 IF P>0 AND P<14 THEN ? "+";
ZX 380 IF P=0 OR P=14 THEN ? "*";
IB 385 IF P>14 AND P<28 THEN ? ". ";
UA 390 P=P+1
HA 395 NEXT I
HQ 400 ? :?"B.I:";
ZW 410 FOR I=1 TO A(M)+W
SA 415 IF U>=33 THEN U=0
RI 420 IF U>0 AND U<16 THEN ? "+";
FW 425 IF U=0 OR U=16 THEN ? "*";
JR 430 IF U>16 AND U<33 THEN ? ". ";
HM 435 U=U+1
GC 440 NEXT I
TB 450 ? :?
UI 460 ? "B.F. - biorytm fizyczny"
RV 470 ? "B.P. - biorytm psychiczny"
NZ 480 ? "B.I. - biorytm intelektualny"
RF 490 ? :? " * - dni krytyczne"
VV 500 ? " + - wzrost":?" "
. - spadek"
DS 510 ? :? :? "W celu uruchomienia progr
amu":?" przycisnij START"
DV 520 IF PEEK(53279)=6 THEN RUN
OA 530 GOTO 520
    
```



# JĘZYKI PROGRAMOWANIA

Czy łatwiej byłoby nam żyć, gdyby wszyscy ludzie porozumiewali się tym samym językiem? Być może dzięki temu nie doszłoby do katastrof takich jak wojny. Esperanto, stworzone przez Ludwika Zamenhoffa jako język powszechny, pozostaje do tej pory ciekawostką, a na świecie w dalszym ciągu królują angielski, niemiecki i francuski. Ponieważ komputer jest dziełem człowieka i odzwierciedlają się w nim ludzkie dążenia do unifikacji — on właśnie może stać się krokiem naprzód na drodze ogólnego zrozumienia się. Egzystencja różnych języków programowania zależy nie tylko od tego, że zostały stworzone przez różnych ludzi, ale również od faktu, że mają one pomóc w rozwiązywaniu wszelkich zadań, które człowiek stawia komputerowi.

Pierwszym poważniejszym zastosowaniem komputerów było sterowanie innymi maszynami. Wymagało to dużej precyzji i szybkości wykonania. W tym momencie niezbędnym stało się rozwinięcie takich form komunikacji, aby wchodzące do komputera zakodowane impulsy były dla niego zrozumiałe. Wraz z postępem technicznym doskonalono również formy przekazywania rozkazów komputerowi.

Przełomowy krok na linii komunikacji między człowiekiem, a maszyną uczynił John von Neuman. Nie narzucał urządzeniu mechanicznych rozkazów tylko przekazywał je poprzez symbole składające się z zer i jedynek, które komputer interpretował jako rozkaz.

Następnym etapem było stworzenie wygodnego i zrozumiałego łącznika z maszyną jakim jest assembler. Uwolnił on programistów od trudnego podwójnego kodu, przy jednoczesnej możliwości stałego kontrolowania własnej pracy. Wraz z upływem czasu assembler jest unowocześniany i chociaż posiada wiele zalet, ma jedną niewątpliwą wadę. Każdy rodzaj procesora rozumie tylko sobie właściwy assembler, co oznacza w praktyce, że dla każdego procesora należy napisać oddzielny assembler.

## Języki do rozwiązywania problemów.

W latach 50-tych pracowano nad stworzeniem lepszej możliwości komunikacji z komputerem. Podjęto próby opracowania języka, który miał pogodzić strukturę problemów do rozwiązywania z wymaganiami człowieka. Szukano języka, który mógłby być użyty w każdym rodzaju komputera.

Wraz z biegiem lat komputer stawał się coraz większym poliglota. Dzisiaj nowe języki programowania w dziedzinie komputerów personalnych pojawiają się wraz z powstawaniem ich nowych generacji. Dzieje się to bardzo szybko. Nie ma jeszcze określonego standardu i na razie nic nie wskazuje na to, aby taki się pojawił. Jedną z przeszkód jest

# ST



fakt, że języki programowania z reguły mają po kilkanaście wersji, które nie zawsze współpracują ze sobą.

Podobnie jest w przypadku komputera ATARI serii ST. Celem artykułu jest omówienie najważniejszych języków programowania na ten komputer. Nie znajdzie nikt podczas czytania wskazówek co jest dobre, a co złe. To zależy od rodzaju problemu do rozwiązania i osobistego doświadczenia. Nie będzie również omówienia wszystkich wersji różnych języków i pod tym względem artykuł ten będzie na pewno niewystarczający.

## FORTAN — ojciec

Na początku lat 50-tych, wraz ze wzrostem udziału komputerów w sferze kierowania życiem gospodarczym, ponadnormatywnie wzrosły koszty rozwoju oprogramowania. Zaczęto kierować uwagę na korektę istniejącej już bazy software'u. Dopiero w 1954 roku udało się koncernowi IBM stworzyć pierwszy większy język programowania. Był to FORTAN (FORmula TRANslator). Od samego początku przeznaczony do pomocy przy rozwiązywaniu skomplikowanych problemów naukowo-technicznych, związanych z dużą ilością trudnych operacji arytmetycznych i algebraicznych oraz odpowiednią obróbką danych. Ponieważ język ten został

opracowany przez duży koncern, stał się jakby z założenia standardem w świecie obróbki danych z dziedziny matematyki. Z czasem tworzono ulepszone wersje tego języka. Najnowszą jest FORTRAN 77. Pomimo dość interesującej oferty sklepów handlujących software'm oraz stałego zwiększania się liczby komputerów personalnych, FORTRAN w dalszym ciągu pozostaje językiem dużych „maszyn”. Dla komputera ATARI ST istnieje wersja FORTRAN-u 77 pod nazwą PROFORTAN firmy Prospero Software i zakres tego języka odpowiada standardowi ANSINN-Norm Fortran 77.

## PASCAL

PASCAL powstał jak gdyby wskutek doświadczeń z różnymi wersjami BASIC-a i FORTRAN-u. Skoki do różnych linii w trakcie realizacji programu czyniły go nieczytelnym i przysparzały kłopotów programistom. I tak w połowie lat 60-tych stworzono język ALGOL, który oparto na zupełnie nowym pomysle-upełnieniu struktur i procedur. Po testach okazało się, że i ten język ma swoje słabe punkty. Wtedy to Niklaus Wirth, który był współautorem jednej z wersji języka ALGOL, stworzył w 1970 roku w Wiedniu język PASCAL. Nazwany tak na cześć znanego filozofa i matematyka Blaise Pascala, a przeznaczony głównie dla informatyków oraz osób zajmujących się naukami ścisłymi. PAS-



CAL ma swoją siłę w zwięzłej i przejrzystej budowie, komfortowych możliwościach rozgałęzień oraz w sposobie organizacji sterowania pętlą. To wszystko buduje podstawy dla strukturalnego sposobu programowania i temu zawdzięcza PASCAL swoją popularność. Inną zaletą tego języka jest możliwość definiowania rejestrów, które obejmują różne rodzaje danych, a następnie rozpatrywania uzyskanych w ten sposób zbiorów.

Pierwszym kompilatorem PASCAL-a dla ATARI ST był GEMDOS-PASCAL firmy GCD. Produkt ten długo pozostawał poza wszelką konkurencją, a jego niedostatki były przez cały czas poprawiane. Istotną pomocą dla użytkowników w pierwszej fazie był podręcznik napisany przez Jurgena Leonharda pod nazwą „EASY PAS”. Obecnie sieć sklepów sprzedających oprogramowanie dostarcza coraz bogatsze wersje tego języka. Na targach CeBit 86 angielska firma Prospero pokazała wersję PROPASCAL, która jest trudniejsza w obsłudze w porównaniu z innymi odmianami tego języka na ATARI ST.

## C — to nie tylko litera

Język C został stworzony w latach 70-tych przez Dennisa Ritchie w firmie AT & T. Przy jego opracowywaniu położono szczególny nacisk na stworzenie języka, który byłby niezależny, a jednocześnie działający w architekturze hardware'u każdego komputera. C ma zastosowanie we wszystkich dziedzinach, od programów maszynowych (gdzie niezbędna jest szybkość), przez tworzenie w CAD, aż do obróbki tekstów. W przypadku ATARI ST język ten jest bardzo licznie reprezentowany, a dzięki bogatej literaturze staje się czołowym językiem tego komputera. Jedną ze starszych wersji C jest produkt firmy Digital Research. Potem pojawiła się wersja GST C-Compiler wykorzystująca w pełni GEM, ale wykazująca pewne braki podczas liczenia.

Język C zyskał szeroki rozgłos wśród użytkowników ATARI ST na całym świecie. Jedną z najlepszych odmian jest tak zwany MEGAMAX C. Nowością na tym rynku jest wersja pod nazwą MARK WILLIAMS C, potrzebująca jednak od 2 do 4 razy dłuższego czasu na kompilację.

## BASIC

BASIC rozwinął się pod koniec lat 60-tych. Pracowali nad nim dwaj Amerykanie Kemeny i Kurtz na uniwersytecie Dartmouth w New Hampshire i na pewno nie byli w stanie przewidzieć, że język ten z czasem zostanie standardem dla wszystkich komputerów domowych i większości personalnych. Pomimo objętości tego języka jest on prosty do nauczenia się, a dialog między człowiekiem a „maszyną” rzeczywiście ma miejsce. Te zalety uczyniły z BASIC-a doskonały język do celów edukacyjnych. Poważnymi wadami są natomiast powolność i mała czytelność. Obecnie na rynku ATARI ST jest kilka wersji interpreterów tego języka m.in. BASIC-Compiler amerykańskiej firmy PHILON INC. Odmiana ta jest uciążliwa w obsłudze, a jedyną jej zaletą jest to, że odpowiada znanej wersji MBASIC firmy MICROSOFT. Poza tym istnieje GFA-BASIC--INTERPRETER oraz niemieckie odkrycie OMIKRON-BASIC, który montowany jako moduł w ROM-ie jest przez cały czas do dyspozycji użytkownika. Nowością na tym rynku jest bardzo dobry amerykański LDW BASIC COMPILER.

## MODULA-2

Jest to kontynuacja języka PASCAL. Duchowym ojcem jest również Niklaus Wirth. MODULA-2 po-

wstała w 1978 roku. Wszystkim posiadaczom ATARI ST firma TDI poleca MODULA-2 SYSTEM wraz z programem narzędziowym (TOOLKIT).

## FORTH — czwarty wymiar

Język ten opracował Charles Moore w latach 60-tych jako język konkurencyjny w stosunku do innych mających małoefektywny sposób liczenia. Pierwszym zadaniem tego języka było sterowanie teleskopem w obserwatorium astronomicznym. Podstawowym elementem, który odróżnia FORTH od innych języków jest uzyskanie od zdefiniowanych elementów języka nowych funkcji. Dzięki temu programista ma nieograniczoną możliwość definiowania różnych funkcji i pojęć. FORTH efektywnie wykorzystuje architekturę procesora, przez co uzyskuje się dużą szybkość obróbki danych.

Pierwszą wersją tego języka dla ATARI ST był 4\*FORTH tzw. Dragon-Group. Inną proponuje firma DATA BECKER pod nazwą FORTH ST. Odmiana ta współpracuje z GEM-em.

## APL — wejście w hieroglify

APL (A Programming Language) — po raz pierwszy zastosowany w legendarnym IBM 360 w dalszym ciągu pozostaje nieznanym szerokim kręgom użytkowników komputerów. Generalną zasadą tego języka jest wykorzystywanie symboli (jak w starożytnym Egipcie) zamiast pseudoangielskich zwrotów typu „ON...GOSUB”.

Dla ATARI ST jest wersja tego języka firmy GDAT pod nazwą APL/68000 oparta na standardzie IBM.

## LOGO — nie tylko dla dzieci

LOGO powstało pod koniec lat 60-tych dzięki profesorowi w Instytucie Technologii w Massachusetts Seymour'owi Papert, który zajmował się w tamtejszym laboratorium problemami sztucznej inteligencji. Papert — matematyk i były student szwajcarskiego psychologa Jean'a Piaget (ojca nowoczesnej psychologii), stworzył LOGO dzięki poszukiwaniom takiego języka, który służyłby jako pomoc pedagogiczna, rozwijająca wśród uczących się zdolności intelektualne i ułatwiająca proces nauki. W ten sposób powstał język będący w stanie przedstawić trudne procesy w sposób obrazowy i plastyczny.

Służą do tego krótkie rozkazy w formie prostych i znanych pojęć. Przy dzisiejszym stanie techniki niemożliwe jest ukazanie pełnych możliwości rozwiązań jakie niesie ze sobą ten język. Elementy rozkazów, które użytkownik ma do dyspozycji można rozwijać poprzez definiowanie funkcji będących częścią składową każdego programu.

LOGO w obszarze sztucznej inteligencji jest poprzednikiem PROLOG-u. ATARI ST posiada wersję LOGO opracowaną przez firmę Digital Research pod nazwą Dr. LOGO. Jest ona uzupełniana przez GEM i prosta do nauki. Pierwsze kroki ułatwia pełen spis treści zawartej w programie i sterowanie przez mysz. Wadą tej wersji jest mała szybkość.

## LISP i PROLOG — języki sztucznej inteligencji

Na początku obydwa miały podobne przeznaczenie jak inne języki programowania. Od kilku lat fachowcy od komputerów coraz więcej uwagi poświęcają tematowi „Komputer-Inteligencja”. W związku z tym języki te zostały odkryte jakby na-

nowo. Należą one do języków deklaratywnych w przeciwieństwie do FORTRAN-u, BASIC-a, PASCAL-a, FORTH-a, które są imperatywne, co oznacza, że program podąża wytyczonym przez programistę szlakiem. Prawdopodobność jego działania zależy od użytych w trakcie przebiegu sytuacji wyjściowych. Jeżeli programista ma mało doświadczenia, jego program będzie nieskuteczny. Deklaratywny charakter oznacza, że program nie posiada algorytmu, tylko składa się z reguł i wniosków końcowych (zasad) obowiązujących w świecie oprogramowania. Natomiast rozwiązaniem dla postawionych pytań jest powracający algorytm (backtracking Algoritmus). LISP jest najstarszym językiem obok FORTRAN-u i jednocześnie egzystującym w wielu odmianach. Powstał początkowo jako kombinacja FORTRAN-u i IPL. John McCarthy — ojciec LISP Procesor (LISP) połączył główną zaletę IPL: budowę struktury danych z wielofunkcyjnością FORTRAN-u. W ten sposób w 1960 roku powstał LISP System.

Także w tej dziedzinie mamy do czynienia z bogatą ofertą na ATARI ST. Jedną z najstarszych wersji jest XLISP napisany w języku C i łatwo dopasowywany do innych komputerów. Firma METACOMCO oferuje inną wersję pod nazwą Cambridge LISP. Kolejną propozycją jest LISPAS II niemieckiej firmy TOMMY SOFTWARE. Odmiana ta dzięki wzajemnym uzupełnianiu się z GEM-em jest krokiem naprzód w świat sztucznej inteligencji.

PROLOG (PROgramming in LOGic) pojawia się na początku lat 70-tych wraz z ideą programowania według logicznego sposobu myślenia. PROLOG został ponownie odkryty kiedy Japończycy wybrali go na język „piątej generacji” i zrobili go częścią składową swojej nowej technologii. PROLOG uwolnił programistów od problemu drogi rozwiązania zadania. Dzięki temu językowi możliwe jest do spełnienia marzenie, żeby zaprogramować komputer bez potrzeby nauki określonego języka programowania. To zupełnie nowa idea.

Dla ATARI ST jest „Prolog Logic-Lab”, który służy jako laboratorium do badania sztucznej inteligencji oraz jest oknem na świat „piątej generacji”. MPROLOG jest kompatybilny ze standardem PROLOG-u opisanym w książce „Programming in Prolog” napisanej przez spółkę autorską Cloksin i Mellisch. Ze swoją szybkością około 800 LIPS (Logical Inferences per Second) może dotrzymać kroku innym wersjom tego języka na różne minikomputery.

## Popatrzmy w przeszłość.

Pierwszym ogniwem łańcucha łączącego człowieka i komputer był assembler. Uwolnił on programistów od kodowania rozkazów, a programy uczynił bardziej czytelnymi i zrozumiałymi. Były one dzięki temu szybsze i doskonale uzupełniały się z komputerem. Jest to niewątpliwą zaletą, lecz zależne od „maszyny” kody czynią assembler językiem nieprzenośnym. Programista pracujący w assemblerze musi uczyć się różnych jego rodzajów, jeżeli ma do czynienia z więcej niż jednym rodzajem komputera. Pomimo tych wad assembler pozostaje jednym z najbardziej efektywnych języków programowania. ATARI ST ma do dyspozycji CP/M 68K ASSEMBLER firmy Digital Research. Jest on bardzo wygodny w obsłudze z racji opanowania standardu MOTOROLI. Inne wersje to IDEAL ASSEMBLER firmy OMIKRON lub GST ASSEMBLER przeznaczony dla ludzi profesjonalnie zajmujących się programowaniem.

na podstawie „ST COMPUTER”  
Sergiusz Piotrowski  
Tomasz Mazur



**Niewątpliwie wielu użytkowników komputera ATARI ST intrygują wciąż dwa niepozorne gniazda MIDI IN oraz MIDI OUT, umieszczone na tylnej ścianie ich maszyny. Z instrukcji obsługi można dowiedzieć się, że jest to Musical Instrument Digital Interface «cyfrowy interfejs instrumentu muzycznego». Brzmi to nieco tajemniczo, a więc zachęcająco! Jeśli więc jakiś program na MIDI wpadnie nam w ręce, uruchamiamy go i widzimy ekran oferujący najrozmaitsze możliwości, a wśród nich bibliotekę zawierającą od kilku do kilkunastu gotowych do odegrania utworów. Wtedy zaczynamy się zastanawiać, kto z naszych znajomych ma syntezator. Kiedy już dojdzie do fuzji syntezatora z komputerem efekt jest niesamowity!**

Nic w tym dziwnego — MIDI powstało dla profesjonalnych muzyków i do profesjonalnych zastosowań.

A musiało powstać bo było potrzebne. Z dwóch zasadniczych powodów:

1. — już kompozytorzy muzyki klasycznej setki lat temu odkryli, że ciekawsze brzmienie pojawia się, gdy ta sama linia melodyczna jest wykonywana przez kilka instrumentów jednocześnie. Podobne efekty zapragnęli uzyskiwać muzycy z kapel rockowych, w skład których z przyczyn ekonomicznych wchodzi z reguły jeden klawiszowiec.

2. — narasta popularność wszelkiego rodzaju automatów i syntezatorów perkusyjnych, które wymagają wzajemnej synchronizacji oraz synchroni-



zacji z innymi instrumentami i syntezatorami — jednolitej i prostej w realizacji.

Dzięki porozumieniu największych producentów elektronicznego sprzętu muzycznego, w 1982 roku doszło do precyzyjnego określenia hardwarowego i softwarowego standardu MIDI. Był to impuls, który wywołał lawinę! W chwili obecnej każdy instrument elektroniczny musi być wyposażony w interfejs MIDI — bez MIDI może być jedynie zabawką.

Dla nowej generacji elektronicznych instrumentów muzycznych — samplerów, syntezatorów, sekwencerów, automatów perkusyjnych, które są już specjalizowanymi komputerami — MIDI to coś równie naturalnego jak CENTRONICS czy RS 232 dla zwykłego komputera.

Upowszechnienie się standardu MIDI stworzyło nową klasę zastosowań dla mikrokomputerów.

Błyskawiczną karierę w roli kontrolera elektronicznego sprzętu muzycznego robi ATARI ST. Stało się to możliwe dzięki umieszczeniu w nim interfejsu MIDI, włączeniu procedur jego obsługi do systemu operacyjnego, dużej pojemności pamięci RAM i szybkiemu mikroprocesorowi. W krótkim czasie od jego wejścia na rynek pojawiło się ponad 30 programów przeznaczonych dla MIDI, co dowodzi zaufania producentów software'u do możliwości tego komputera jako kontrolera MIDI.

Oprogramowanie muzyczne można podzielić na dwie zasadnicze grupy: — sekwencery i edytory brzmień.

Sekwencer — to program umożliwiający zanotowanie w pamięci komputera muzyki aktualnie wykonywanej na syntezatorze. Działa on podobnie jak wielośladowy magnetofon studyjny, ale posiada również zupełnie nowe możliwości, które daje technika

cyfrowa. Istnieją dwa sposoby wprowadzania do komputera dźwięków z klawiatury syntezatora: w „czasie rzeczywistym” — gdy program rejestruje dokładnie to, co w danej chwili „jest grane”, oraz „praca krokowa” — kiedy do pamięci wprowadzana jest oddzielnie każda nuta czy akord. Ten drugi sposób pracy z sekwencerem jest niewątpliwie wolniejszy, ale w efekcie otrzymuje perfekcyjny zapis nawet najbardziej akrobatycznych pomysłów.

Niektóre z sekwencerów przeznaczonych dla komputera ST mają 60 ścieżek oraz możliwości jakie daje bardzo bogato wyposażone studio: programowanie zmian głosów, zmiany tempa, sterowanie zewnętrznymi automatami perkusyjnymi itp. Pamięć RAM o pojemności 1MB umożliwia zapisanie ponad 63 000 nut — jest to więcej niż oferują specjalizowane sekwencery pochodzące z tak renomowanych firm jak Roland czy KORG. Wiele z tych programów symuluje na ekranie monitora tradycyjny magnetofon szpulowy, ale już tylko po to, aby uchronić użytkownika przed ewentualnym szokiem przy kontakcie z tą nową techniką zapisu muzyki.

Edytor brzmień pomaga w programowaniu nowych brzmień na syntezatorze. Wyświetlacz w syntezatorze umożliwia oddzielne pokazanie pojedynczych parametrów programowanego dźwięku, zaś ekran edytora wyświetla Ci je wszystkie w taki sposób, że programowanie nowego dźwięku staje się czystą przyjemnością. Edytor umożliwia również tworzenie banku brzmień na dyskach.

Fascynujące możliwości oferuje nowa generacja programów, wykorzystująca sztuczną inteligencję, która umożliwia tworzenie abosolutnie nowych brzmień oraz sortowanie i zapamiętywanie wg barwy po to, aby w razie potrzeby szybko je odnaleźć i przeprogramować syntezator. Należy jeszcze dodać, że programy Edytorów brzmień są tworzone z myślą o określonym typie syntezatora, więc stosowanie ich do innego typu syntezatora jest niemożliwe.

Coraz popularniejsza staje się jeszcze jedna kategoria programów MIDI — programy umożliwiające wydruk zapamiętanej przez komputer muzyki w postaci standardowej notacji muzycznej (nuty!). Oczywiście, aby korzystać z tej możliwości trzeba dysponować drukarką graficzną, ale co za wyręka dla profesjonalistów!

Krótką historia MIDI ukazuje jeszcze raz, że życie często przeraża oczekiwania, gdyż w chwili narodzin MIDI większość jego obecnych zastosowań nie istniała nawet w umysłach projektantów. Nowe pomysły rodzą się bezustannie, powstają nowe programy, zaś ATARI ST — dzięki MIDI — zdobyło uznanie profesjonalnych muzyków i jest podporą tak renomowanych zespołów jak Dire Straits czy Marillion.

Bez wątplenia, spośród wszystkich komputerów osobistych ST ma największe możliwości, aby spełniać rolę kontrolera muzycznego. Popularności wśród profesjonalnych muzyków przysparza mu także umiarkowana cena.

*Tadeusz Pękacz*

## KOPROCESOR ARYTMETYCZNY

**To, że ATARI ST oparte na mikroprocesorze MOTOROLA 68000 jest bardzo szybkie, nie jest dla nikogo tajemnicą. Dzięki temu komputer ten trafia także „pod strzechy” polskich wyższych uczelni, gdzie skomplikowane zadania matematyczne są na porządku dziennym.**

Sprawność ST w tym obszarze można podnieść przez „dołożenie” koprocatora 68881 do obliczeń arytmetycznych. Procesor MOTOROLA 68000 służy do dyspozycji użytkownika rozkazy niezbędne do: dodawania, odejmowania, mnożenia i dzielenia. W przypadku, gdy zajdzie potrzeba zrobienia obliczeń o wysokim stopniu dokładności, bądź też pojawią się kompleksowe zadania z użyciem sinusów, logarytmów itd. należy nastawić się na „małe kłopoty”.

I w tym miejscu zaczyna się rola koprocatora 68881. Posiada on bardzo wysoką zdolność rachunkową i obejmuje wszystkie najważniejsze funkcje matematyczne. Ponadto uwalnia CPU (właściwy procesor) od czasochłonnych obliczeń i przyspiesza poszczególne kroki operacji liczenia.

Fliesskommaarithmetik-Coprocessor — jak nazwali ten układ Niemcy — stworzony został w firmie

LISCHKA-DATENTECHNIK. Są dwie drogi uzyskania tego układu. Pierwsza to samodzielny montaż 68881, co wiąże się ze skomplikowanymi pracami lutowniczymi oraz druga, łatwiejsza ale i droższa — kupno ATARI ST z zamontowanym już koprocetorem arytmetycznym. Fakt, że ST posiada omawiany układ, nie ogranicza możliwości dalszych rozszerzeń tego komputera, jak też nie blokuje miejsca na płycie głównej. Istnieją dwa sposoby montażu 68881, przez przyłączenie do CPU lub przez włożenie w podstawkę. Mniej eleganckim, ale za to bardziej praktycznym sposobem jest ten pierwszy. Przy zastosowaniu tej metody nie zamyka się szansy dalszych połączeń (LISCHKA-DATENTECHNIK planuje jeszcze tzw. Mini-BUS z 64 KB przestrzenią adresową dla np. drugiego wejścia szeregowego bądź równoległego i ROM-Floppy z 512 KB).

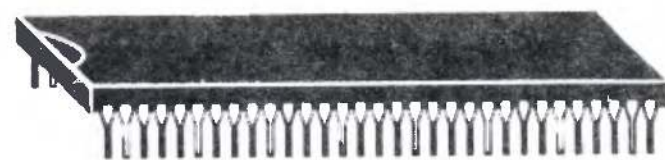
Aby wykorzystać pełne możliwości 68881 należałoby go podłączyć do systemu z układem MOTOROLA 68020. W takim trybie koprocetor pokazuje swoje rejestry jako dodatkowe rejestry procesora. Ten komfort jest jak na razie w ATARI niemożliwy, ale Floating-Point-Processor, jak nazwali go Anglicy, został tak pomyślany, że może współpracować z każdym procesorem serii 680xx.

Wnętrze 68881 to osiem 80-bitowych rejestrów danych (norma amerykańska). Dodatkowo, w stosunku do MOTOROLI 68000, posiada koprocetor arytmetyczny jeszcze trzy 32-bitowe rejestry (Control-Status-, Instruction Address-Register).

Jeśli chodzi o rozkazy matematyczne układowi 68881 nie brakuje nic. Funkcje, które realizuje można podzielić na takie, które mają tylko jeden argument (np. obliczanie sinusa bądź logarytmu) i pozostałe.

Komunikacja koprocatora z CPU przebiega poprzez Coprocessor-Interface-Register, który leży w przestrzeni adresowej MOTOROLI 68000. Praca z ATARI ST wyposażonym w układ 68881 to podobno prawdziwa radość, wynikająca z tego, że można bez problemu rozwiązywać nawet najbardziej złożone zadania matematyczne.

*Sergiusz Piotrowski*







## POROZMAWIAĆ Z ST?

**Na przestrzeni wieków szarlatani i czarodzieje, uczeni i geniusze, ludzie różnych profesji próbowali stworzyć sztuczną inteligencję bądź też nadać sztucznym twórcom cechy czysto ludzkie (miedzy innymi także jak możliwość używania głosu).**

Do próby naśladowania człowieczeństwa należy bez wątpienia zaliczyć przetworzenie ludzkiego głosu za pomocą urządzeń elektronicznych.

Bardzo upraszcza rozwiązanie tego problemu wprowadzanie coraz to nowych i bardziej doskonałych komputerów. Do takich można zaliczyć komputer ATARI serii ST.

Zachodniemiecka firma Schlegel Datentechnik wyprodukowała właśnie dla ATARI ST moduł, który umożliwia wytwarzanie dźwięków przypominających ludzką mowę.

Cale urządzenie składa się z elektroniki (układ scalony przetwarzający mowę i wzmacniacz), głośnika i małego zasilacza (zabezpiecza dopływ prądu). Elektronika umieszczona jest w zgrabnej obudowie, na zewnątrz której znajdują się dwa pokrętki (jedno do regulacji siły głosu i drugie do regulacji szybkości reprodukcji głosu). Podłączenie modułu do ATARI

ST odbywa się przez port (wejście) do drukarki przez oddzielny kabel połączeniowy. Fakt podłączenia „naśladowcy mowy” nie wyklucza możliwości korzystania z drukarki. Dzieje się tak dzięki przełącznikowi, który pozwala na wybór pożądanej opcji. Umieszczony jest on z tyłu obudowy modułu obok wejścia zasilacza i wejścia umożliwiającego podłączenie głośnika.

### Nie tylko elektronika

Opisany moduł mowy jest wyposażony w dyskiety z oprogramowaniem. Użytkownicy mają dwie możliwości, aby wytworzyć dźwięk. Pomocne mogą tu okazać się programy GFA-Basic oraz 1 st Word. W przypadku użycia edytora pożądany tekst zostanie wypowiedziany zamiast wydrukowany. Jakość syntetycznego głosu nie jest co prawda jeszcze rewelacyjna, ale poprawa jest tylko kwestią czasu. Procesor, który jest wbudowany w moduł mowy, a wpływający na jakość przetwarzanego głosu, pozwala przy odpowiednim oprogramowaniu wydobyć szczególnie efektowne i ciekawe dźwięki. W tym właśnie celu dołączona jest do software'u lista z fonetycznie zapisanymi sylabami i należącymi do nich kodami. Dociekliwy użytkownik znajdzie w tym załączniku z pewnością dużą pomoc i oparcie, aby zoptymalizować jakość wydobywanych tonów.

### Co dalej z tym fantem?

Trzeba się zastanowić, do czego może być przydatny opisany imitator dźwięków. Czy należy traktować go jako rozszerzenie jakiejś poważnej dziedziny wiedzy, czy też jako kolejną elektroniczną zabawkę. Jeżeli potraktować to jako zdigitalizowany dźwięk przy codziennym powitaniu z ATARI ST, to jest to oczywiście sztuczka — zabawa dla dużych dzieci.

Patrząc na to z drugiej strony — takie urządzenie może okazać się łącznikiem z obcym światem niepoznanym jeszcze przez człowieka lub też nieodłączną częścią laboratorium językowego. W chwili obecnej przy dostępnym oprogramowaniu moduł mowy ST VOICE nie jest w stanie spełnić pełnych oczekiwań użytkowników. Powinno to zmienić się w przyszłości. Innym problemem jest bardzo rozbudowana sieć połączeń — na przykład reprodukcja głosu przez oddzielny głośnik. Pozostaje więc oczekiwać, aż nosniki ST VOICE zostaną poprawione. To z kolei umożliwi znalezienie poważnego zastosowania dla „naśladowcy mowy”.

**Producent: Schlegel Datentechnik  
Schwarzachstrasse 3  
7940 Riedlingen**

*Na podstawie „ST Computer”  
Sergiusz Piotrowski*