

Paweł Sikorski

TurboXL BASIC

Zbiór artykułów ze strony
<http://turbobasicxlandaction.pl>

©2013 Sikor Soft

1. Wstęp

Każdy użytkownik Atari XL/XE ma do dyspozycji wbudowany język programowania – **ATARI BASIC**. Po wczesnej fascynacji grami lub programami demonstracyjnymi przychodzi u większości z nas, użytkowników maszyn retro, czas, w którym staramy się zaprogramować naszą maszynkę. Oczywiście wybór pada na wbudowany język programowania.

Zazwyczaj szybko okazuje się, że ten wybór nie spełnia naszych oczekiwań. Język ten oprócz tego, że jest stosunkowo wolny (o tym dalej), ma szereg ograniczeń. Szybkie wyszukiwanie zazwyczaj daje nam jeden cel: assembler. Ale tutaj jest pies pogrzebany – szybko okazuje się, że mimo swej prostoty, jest to język, który wymaga dokładnej znajomości nie tylko samej struktury programu, ale także tzw. „*mapy pamięci*” samego komputera nieomal że co do bajtu. A na poznanie tego nie każdy ma czas i chęci, niekiedy zaś po prostu brakuje nam samozaparcia.

Czy w związku z tym mamy się od razu zniechęcić? Nie, na Atari powstał cały szereg języków programowania, można wręcz twierdzić, że pod względem ich ilości (a często i jakości) – jest to jeden z najlepiej „oprogramowanych” komputerów ośmiobitowych. Piszę tu oczywiście o językach natywnych, czyli występujących bezpośrednio na naszym sprzęcie.

Mamy do wyboru zarówno tak zwane języki kompilowane (wikipedia: http://pl.wikipedia.org/wiki/Język_kompilowany), jak i interpretowane (wikipedia: http://pl.wikipedia.org/wiki/Język_interpretowany), do jakich należy Basic. Zaletą języków kompilowanych jest zazwyczaj ich prędkość, za to mają większą złożoność przy tworzeniu kodu – każdorazowe tworzenie kodu przed debugowaniem wymaga kompilacji, a potem często linkowania z biblioteką lub innymi częściami programu. Języki interpretowane są zazwyczaj prostsze, ale i wolniejsze. Co więcej – zazwyczaj wymuszają blokowe myślenie przy pisaniu kodu.

Czy w związku z tym należy się przeczucić na język kompilowany, aby uzyskać większą prędkość? Niekoniecznie, do wielu zadań w zupełności wystarczają języki interpretowane, do jakich zalicza się także Atari Basic. Pewnym rozszerzeniem jego możliwości jest **Turbo Basic XL** – który zostanie opisany na łamach niniejszej strony, zarówno w postaci serii artykułów (moje artykuły będą także umieszczać w formie *.pdf, do łatwiejszego wydruku – kolejne części będzie można dołączać do niniejszego tekstu, tworząc pewną całość, którą będzie można spiąć wspólnie w skoroszycie A5), jak i biblioteki gotowych procedur do wykorzystania (zamieszczanych w formie tekstu) – aby stworzyć pewną ich bazę.

Co przemawia za zastosowaniem Turbo Basica XL? Między innymi jego wsteczna kompatybilność z Atari Basicem, przez co nie tracimy wcześniejszych prac. Większość poprawnie napisanych programów z tego drugiego zadziała w naszym nowym środowisku, trzeba jednak wziąć poprawkę, że tu mogą działać nieco szybciej, co się wiąże ze specyfikacją nowszego podejścia do tematu. Co więcej, do Turbo Basica XL (nazywanego od tej pory w skrócie TB XL) został stworzony bardzo ciekawy kompilator (Turbo Basic XL Compiler) oraz wygodny runtime, uruchamiający skompilowane programy (za pomocą linkera możemy go połączyć bezpośrednio ze skompilowanym programem, tworząc poprawny plik wykonywalny dla DOSa).

W serii artykułów nie zamierzam dokładnie omawiać składni języka, ale pewnie pewne podstawy się mimochodem znajdą. Dla ciekawych polecam lekturę książki W. Zientary „Języki Atari XL/XE – część pierwsza (do znalezienia na www.atarionline.pl w bibliotece atarowca) lub serii artykułów traktujących o TB XL w „Tajemnicach Atari” (do znalezienia na stronie <http://tajemnice.atari8.info/>).

Zapraszam do zapoznawania się z kolejnymi częściami artykułu.