IAR visualSTATE®

Firmy IAR Systems żadnemu elektronikowi nie trzeba przedstawiać. Od lat 80. jest ona jednym z rynkowych liderów w dziedzinie kompilatorów języka C/C++ dla szerokiej gamy procesorów, począwszy od prostych układów 8-bitowych, aż do bardzo rozbudowanych układów 32-bitowych. IAR visualSTATE jest narzędziem umożliwiającym budowanie programów dla mikrokontrolerów w sposób graficzny, za pomocą definiowania stanów urządzenia oraz przejść między nimi. Część Czytelników z pewnością pamięta z przedmiotu „Układy cyfrowe" matematyczny model układu sekwencyjnego, czyli „automat" oraz takie pojęcia jak „funkcja przejść", „funkcja wyjść" itp. Na tym właśnie opiera się idea programowania w IAR visualSTATE, ale na szczęście nie ma potrzeby odgrzebywania notatek z wykładów, gdyż praca w tym pakiecie odbywa się całkowicie intuicyjnie.

IAR visualSTATE jest rozbudowanym zestawem narzędzi, wspomagającym pracę programisty w jednolitym środowisku. Tworzenie programu w *IAR visualSTATE* polega na narysowaniu grafu przejść między zdefiniowanymi stanami, opisującymi stan automatu. Następnym krokiem jest symulacja działania programu w oparciu o graf oraz jego dokładna analiza i testowanie. Etapem kończącym jest generowanie kodu w języku C, odpowiadającego automatowi opisanemu za pomocą grafu. Tak pokrótce wygląda praca z pakietem *IAR visualSTATE*. W dalszej części artykułu zostaną przedstawione narzędzia wchodzące w skład pakietu.

visualSTATE Navigator

Zadaniem *Navigatora* (**rys. 1**) jest zarządzanie projektami pakietu *visualSTATE*. Każdy etap w procesie tworzenia programu ma swój początek w *Navigatorze* i to on odpowiada za utrzymanie spójności projektu. Z poziomu tego modułu są uruchamiane inne narzędzia wchodzące w skład pakietu (*Designer*, *Coder*, *Validator* itd.). W *Navigatorze* są również ustawiane specyficzne opcje dla *Codera* i *Verificatora* oraz jest zapewniony dostęp do pełnej dokumentacji wchodzącej w skład pakietu. Wykorzystanie wszystkich funkcji *Navigatora* wymaga zainstalowania *Adobe Acrobat Readera* oraz *Microsoft Internet Explorera*.

visualSTATE Designer

Jak wspomniano we wstępie programowanie w *visualSTATE* odbywa się za pomocą grafów. Jest to wygodna i czytelna metoda do opisania różnych zjawisk, często spotykana w dokumentacji programów tworzonych w sposób tradycyjny. Dzięki *Designerowi* (**rys. 2**) rysowanie grafów jest niezwykle łatwe, a możliwość korzystania z hierarchicznej struktury budowanego programu pozwala zachować przejrzystość w realizowanym projekcie. Hierarchiczna budowa projektu umożliwia również testowanie prototypu na wczesnym etapie uszczegółowiania projektu oraz analizę działania elementów składowych projektu.

visualSTATE Validator

Validator (**rys. 3**) jest bardzo rozbudowanym debuggerem projektów *visualSTATE*. Umożliwia on symulowanie działania programu oraz jego analizę statyczną i dynamiczną (**rys. 4**). Ręczne wywoływanie zdarzeń umożliwia śledzenie zachowania programu w zależności od występujących okoliczności. Wszystkie zdarzenia z przebiegu symulacji mogą być zapisane w pliku, co ułatwia później analizę działania programu. Możliwe jest również sprawdzenie, jakie zdarzenia powodują określone działanie (np. jakie zdarzenie powoduje włączenie oświetlenia). Działanie programu może być przedstawione jako animacja (**rys. 5**). Podczas budowy większych projektów z pewnością nieocenioną pomocą dla programisty będzie możliwość ustawiania pułapek programowych.

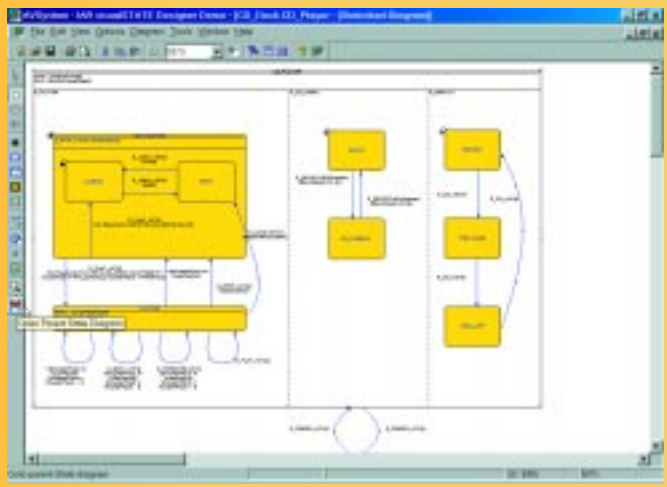
visualSTATE Coder

Na podstawie grafu stworzonego w *Designerze*, *IAR visualSTATE Coder* generuje kod w języku ANSI-C. Generowany kod ten jest niezależny od platformy sprzętowej. Z jednej strony umożliwia to realizację projektu na różnych mikroprocesorach (możliwe jest również testowanie programu na komputerze PC z wykorzystaniem np. *Borland C++*), z drugiej jednak wymaga od użytkownika samodzielnego stworzenia interfejsu pomiędzy programem i sprzętem. Sprawa jest dosyć skomplikowana i wymaga od użytkownika dosyć dobrej znajomości języka C. Generowany kod zawiera standardowe dla projektów *visualSTATE* funkcje interfejsu (API), co umożliwia budowanie przenośnych programów o dobrze zdefiniowanej strukturze. Tak więc ostatecznie program składa się z dwóch części: z części specyficznej dla danej aplikacji (niezależnej od sprzętu) oraz z części definiującej współpracę ze sprzętem (ale za to niezależnej od budowanej aplikacji).

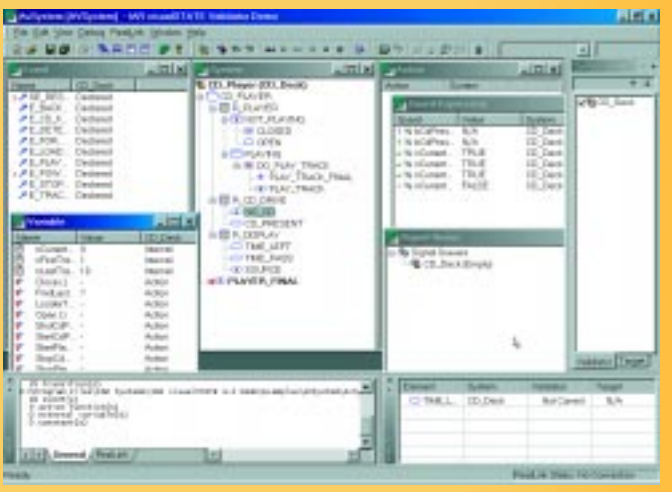
visualSTATE Documenter

Dla większości programistów tworzenie dokumentacji to katorga. Z pomocą przychodzi *Documenter*, który nie tylko automatycznie generuje szkielet dokumentacji, ale w części dotyczącej projektu *visualSTATE'a* również ją wypełnia. Robi to w atrakcyjnej formie graficznej w formacie HTML lub RTF

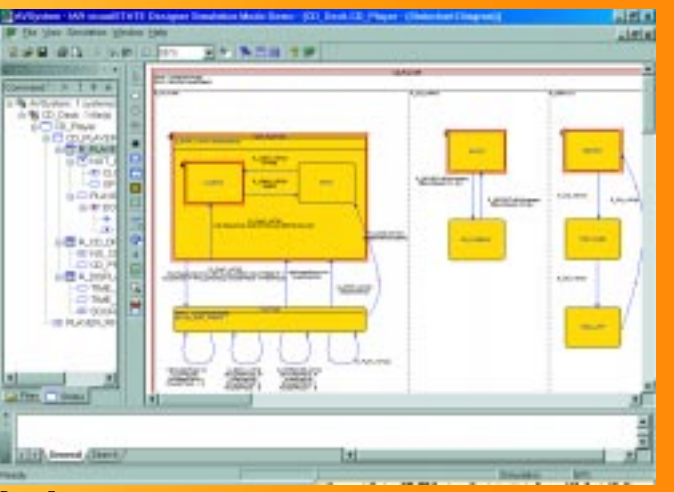
Rys. 1.



Rys. 2.



Rys. 3.



Rys. 5.

(**rys. 6**), tak więc jest możliwa dalsza praca nad dokumentacją w praktycznie dowolnym edytorze. Zawartość generowanego dokumentu można zmieniać (np. można dodać wyniki analizy), ustawiając odpowiednie opcje w *Documenterze*.

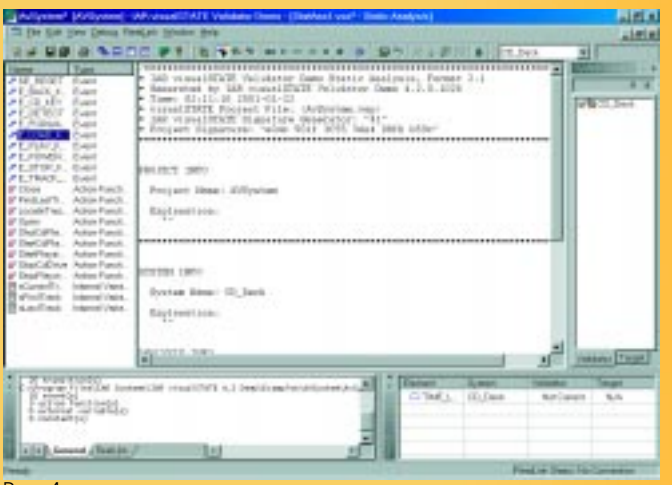
Korzystanie z graficznych narzędzi wymaga od programisty poświęcenia pewnego czasu na oswojenie się ze zwykle nietypowym, specyficznym dla producenta oprogramowania interfejsem. Wielu użytkowników z pewnością będzie miało duże opory w poznawaniu nowego sposobu programowania, ale doświadczenie zdobyte podczas realizacji jednego projektu niemal natychmiast procentuje przy realizacji kolejnych projektów, a praca z wykorzystaniem narzędzi graficznych jest szybsza i łatwiejsza. *IAR visualSTATE* pracuje pod kontrolą systemu operacyjnego *Windows 95/98/Me/NT/2000* praktycznie na każdym komputerze spełniającym wymagania stawiane przez system operacyjny.

IAR visualSTATE wspomaga programistę na każdym etapie procesu tworzenia kodu (oprócz warstwy współpracującej ze sprzętem). Generowany kod pracuje w oparciu o zdarzenia, więc system budowany nawet na prostym mikroprocesorze zapewnia „równoległe" wykonywanie programu. Dzięki temu oraz dzięki rozbudowanym możliwościom *Validatora*, realizowanie systemów czasu rzeczywistego staje się naprawdę łatwe.

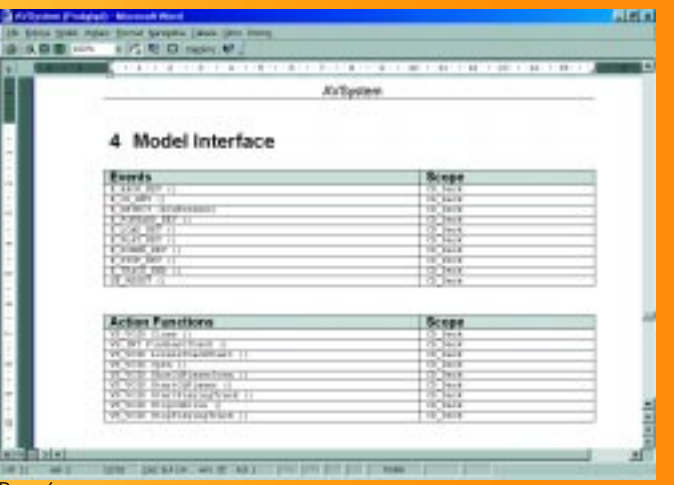
Paweł Zbysiński

IAR visualSTATE 4.2 do testów w redakcji dostarczyła firma RK-System, tel. (0-22) 724-30-39), www.rk-system.com.pl.

Na płycie CD-EP02/2001B znajduje się IAR visualSTATE 4.2 w wersji demonstracyjnej (ograniczenie projektu do 20 stanów).



Rys. 4.



Rys. 6.