

QNX Momentics

Praca z pakietami BSP

Era Momenticsa

Wprowadzenie do sprzedaży w czerwcu 2002 roku pierwszej komercyjnej wersji pakietu QNX Momentics okazało się trafionym posunięciem. Oprogramowanie odniosło duży sukces i obecnie ma spore szanse zostania jednym z najlepszych narzędzi programistycznych w kategorii systemów czasu rzeczywistego. Dynamiczny rozwój poparty szeregiem „latek“, aktualizacji, nowych wersji oprogramowania oraz dodatkowych pakietów kreuje obraz godnego zaufania produktu, i to pomimo tak młodego wieku (na rynku nieco ponad rok).

Obecnie swój rozkwit przeżywiają kompletne, zintegrowane środowiska programistyczne (QNX Momentics, Tornado II...). Trwa nieustający wyścig w ulepszaniu oraz rozbudowywaniu narzędzi, a producenci dokładają wszelkich starań, aby ich programy narzędziowe były jak najbardziej uniwersalne oraz łatwe w użyciu i nauce.

I tak dla przykładu zestaw QNX Momentics zawiera: system operacyjny QNX Neutrino (obecnie w wersji 6.2.1), środowisko graficzne Photon wraz z programem do tworzenia aplikacji okienkowych (phAB), zintegrowane środowisko programistyczne IDE, narzędzia do pisania własnych sterowników DDK, pakiety BSP, biblioteki i narzędzia GNU, instruktażowe klipy wideo, obszerną dokumentację oraz szereg kodów źródłowych. Naszą uwagę, w dalszej części artykułu skupimy na dedykowanych zestawach BSP.

Co to jest BSP?

Pakiet BSP (*Board Support Package*) jest zestawem kodów źródłowych, plików binarnych, konfiguracyjnych oraz kreatorów mających za zadanie uruchomienie systemu operacyjnego wraz z kompletną obsługą wszystkich urządzeń docelowej platformy sprzęto-

wej. Począwszy od QNX Momentics 6.2.0 (17 pakietów dla 25 popularnych platform), liczba zestawów stale rośnie i aktualna ich lista znajduje się pod adresem: http://www.qnx.com/products/ps_bsps. Zanim przejdziemy do omawiania przykładowego pakietu BSP musimy zatrzymać się na chwilę przy analizie sekwencji bootowania systemu operacyjnego.

Bootowanie systemu operacyjnego QNX Neutrino

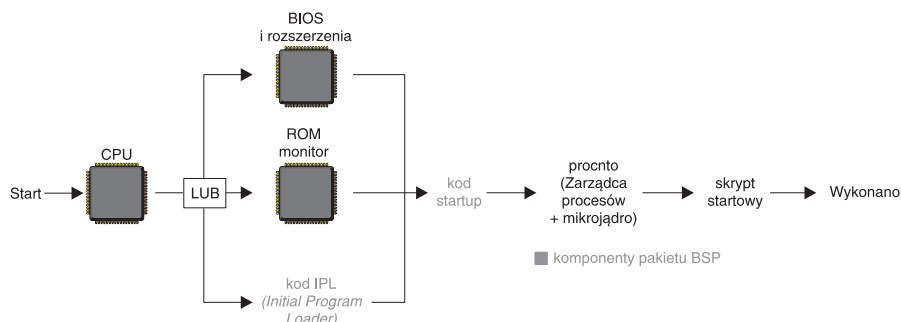
Co dzieje się podczas bootowania? Jakie komponenty są wymagane dla poprawnego załadowania systemu? W odpowiedzi na te pytania pomoże nam schemat inicjacji systemu operacyjnego QNX Neutrino przedstawiony na rys. 1.

Po włączeniu zasilania, resęcie sprzętowym bądź programowym procesor zaczyna wykonywać instrukcje kodu spod specyficznego dla danej architektury adresu (*reset vector*). Kodem tym może być BIOS, ROM monitor, ewentualnie inicjujący program ładujący IPL (*Initial Program Loader*). Pierwszy scenariusz, typowy dla komputerów klasy PC jest najłatwiejszy w realizacji, ponieważ wszystkie czynności konfiguracyjne wykonywane są z poziomu BIOS-u. BIOS zwykle przeprowadza wstępną diagnostykę oraz ustawia poszczególne urządzenia sprzętowe. W następnym etapie BIOS znajduje swoje rozszerzenia (kontroler dysku twardego, boot

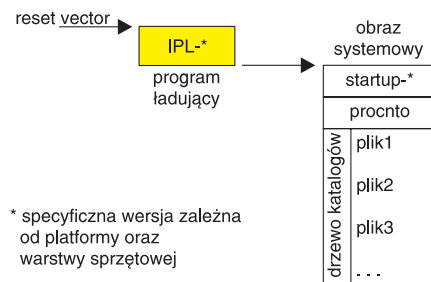
Artykuł ma na celu przybliżenie Czytelnikom tematyki tworzenia wbudowanych obrazów systemowych w oparciu o narzędzia zestawu programistycznego QNX Momentics. Główny nacisk położono przede wszystkim na wykorzystanie pakietów BSP stanowiących niezbędne, fundamentalne wsparcie przy opracowywaniu własnego projektu dla specyficznej platformy docelowej.



ROM karty sieciowej itp.) oraz skacze do nich w celu wykonywania dalszego ciągu programu. Po wykonaniu skoku następuje załadowanie obrazu systemowego do pamięci (np. z dysku twardego) i przekazanie kontroli programowi *startup*. ROM monitor jest specjalnym oprogramowaniem sprzętowym (*firmware*), zapisywanym do pamięci



Rys. 1. Sekwencja inicjowania systemu QNX Neutrino



* specyficzna wersja zależna od platformy oraz warstwy sprzętowej

Rys. 2. Start systemu z wykorzystaniem programu ładującego IPL

Flash w procesie produkcji urządzenia. Przeważnie oprogramowanie sprzętowe obsługuje: transfer danych przez port szeregowy (protokół Xmodem/Ymodem), sieć Ethernet (klient TFTP), proste operacje na pamięci (wyświetlanie, usuwanie i zapis) oraz kilka wariantów debugowania niskiego-poziomu. ROM monitor wgrywany jest z reguły za pomocą programatora pamięci bądź też interfejsu JTAG. Zarówno w pierwszym jak i drugim przypadku większość czynności konfiguracyjnych zostaje wykonana przez BIOS bądź ROM-monitor. Taki rodzaj inicjacji nazywamy gorącym startem.

Start systemu z wykorzystaniem inicjującego programu ładującego IPL

Podczas zimnego startu systemu nic nie jest zainicjowane oraz ustawione. Procesor i cała reszta sprzętu jest po prostu ponownie przywracana do swoich ustawień domyślnych. W jaki zatem sposób dokonywana jest inicjacja systemu?

Procedura konfiguracji oraz ładowania systemu została podzielona na dwa etapy: IPL i *startup*. Pierwszym krokiem wykonywanym przez oprogramowanie jest wgranie obrazu systemu przez program ładujący IPL. Ponadto zadaniem IPL-a jest dokonanie minimalnej konfiguracji sprzętowej, która umożliwi w konsekwencji przejście do drugiego etapu - uruchomienia programu *startup*.

Kod programu IPL został rozбит na dwa programy, które podczas kompilacji są łączone w jeden plik wynikowy. Pierwszy z nich (napisany całkowicie w assemblerze) jest odpowiedzialny za utworzenie środowiska dla programu *startup*,

a mianowicie konfiguracji: zegarów, GPIO, kontrolera pamięci i/lub kontrolera PCI. Ostatecznie program ładujący skanuje pamięć FLASH w poszukiwaniu obrazu, weryfikuje go oraz wgrywa do pamięci RAM. W przypadku nie znalezienia obrazu systemowego IPL umożliwia jego wgranie poprzez port szeregowy (protokół *sendnto*).

Po przekazaniu kontroli do programu *startup*, wykonywane są dalsze czynności konfiguracyjne sprzętu: jednostki MMU, czasomierza systemowego, kontrolera przerwań oraz strony systemowej (czas, data, ilość pamięci, typ procesora, koprocessora i szyny sprzętowej). Gdy już wszystkie urządzenia są poprawnie ustawione *startup* przekazuje kontrolę następnemu programowi obrazu systemowego *procnto* - czyli po prostu uruchamia system operacyjny. Mikrojądro systemu QNX Neutrino wraz z Zarządcą Procesów uruchamiają pozostałe pliki wykonawcze (aplikacje, sterowniki, protokoły itp.) wyszczególnione w skrypcie startowym. Uproszczony proces ładowania systemu pokazano na rys. 2.

Praca z pakietem BSP

Przykładowy pakiet dla platformy Intel a PXA250TMDP BSP zawiera:

- kod źródłowy: IPL, *startup*, sterownika kontrolera dźwięku, systemu plików flash oraz wszystkich dodatkowych bibliotek,
- sterowniki do pozostałych urządzeń (w postaci binarnej): portu szeregowego, karty sieciowej, graficznej oraz ekranu dotykowego,
- szereg kreatorów (pliki *makefile*) oraz dodatkowe pliki konfiguracyjne - np. plik na podstawie którego generowany jest obraz systemowy (*Buildfile*).

W jaki sposób należy rozpocząć pracę z pakietem BSP?

Schemat ideowy pracy z zintegrowanym zestawem programistycznym QNX Momentics przedstawiono na rys. 3.

Przykładowo dla środowiska programistycznego QNX Neutrino 6.2.1 zestawu BSP zainstalowane są w następującym katalogu */usr/src/*

bsp-6.2.1 oraz podzielone są na dwa obszary: specyficznej platformy */usr/src/bsp-6.2.1/processor/platforma* i bibliotek (np. biblioteka IPL, biblioteka *startup*) */usr/src/bsp-6.2.1/libs*. Ponadto dla każdej platformy wyróżniamy:

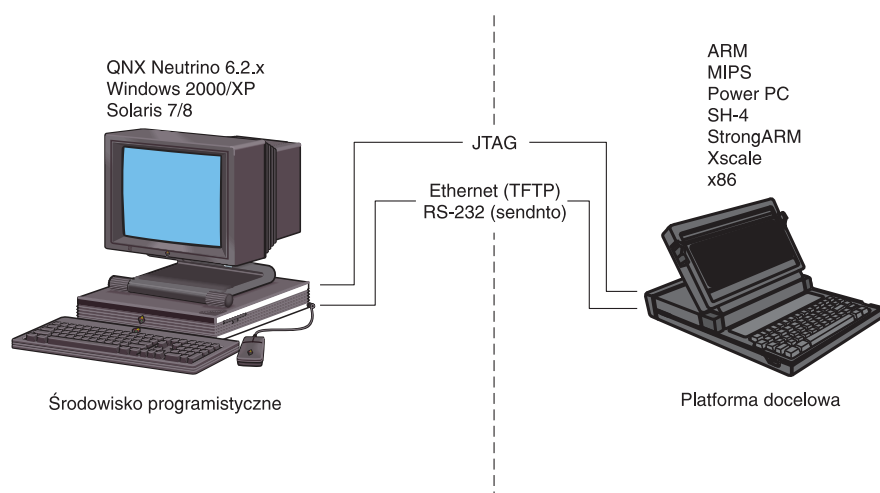
- */usr/src/bsp-6.2.1/processor/platforma/src* -> katalog z kodami źródłowymi: IPL, *startup*, systemu plików Flash, serwera PCI itp.,
- */usr/src/bsp-6.2.1/processor/platforma/scratch* -> katalog ten zawiera pliki wynikowe utworzone poleceniem *make install* wywołanym w katalogu *src*,
- */usr/src/bsp-6.2.1/processor/platforma/images* -> katalog zawierający *makefile*, pliki „budujące“ (*build files*), opisowe oraz dodatkowe skrypty,
- */usr/src/bsp-6.2.1/processor/platforma/prebuilt* -> katalog ten używany jest podczas pierwszego wywołania komendy *make* na poziomie katalogu *processor/platforma*.

Praca z pakietem BSP w zintegrowanym środowisku programistycznym IDE

Korzystając z kreatora należy utworzyć nowy projekt (Standard Make C Projekt) oraz zaimportować do niego całą zawartość katalogu */usr/src/bsp-6.2.1/processor/board/* (w przypadku pracy ze źródłami danej platformy) bądź też */usr/src/bsp-6.2.1/libs* (praca z bibliotekami).

Wynikowe pliki binarne (IPL, obraz systemowy) mogą być przesyłane do urządzenia za pomocą portu szeregowego (protokół *sendnto*). Ponadto IDE posiada wbudowany serwer TFTP umożliwiający transfer danych poprzez sieć Ethernet.

Tworzenie własnych, bootowalnych obrazów systemowych dla pamięci Flash wspomaga w dużym stopniu narzędzie System Builder. Wystarczy kliknąć na żądanym komponencie, a System Builder sprawdzi zależności oraz automatycznie dołączy wszystkie wymagane elementy składowe (biblioteki, itp.). Istnieje również możliwość redukcji rozmiaru danej biblioteki do pliku zawierającego minimalny zbiór funkcji używanych w naszym systemie docelowym.



Rys. 3. Praca z zestawem QNX Momentics

Plany na przyszłość

Na początku lipca ukazał się nowy, poszerzający możliwości systemu pakiet - QNX Momentics 6.2.1B PE. Nowością jest zestaw narzędzi wspomagających zarządzanie energią (*Power Management*). Zcentralizowane sterowanie zasilaniem umożliwia projektantom pełną kontrolę stanu zasilania całego systemu, jaki i poszczególnych jego komponentów składowych. Zestaw specjalnych funkcji pozwala przechwytywać interakcje pomiędzy jednostką zarządzającą a wszystkimi urządzeniami systemu.

Producent systemu QNX Software Systems Ltd. zapowiada agresywną politykę rozwoju pakietów BSP. W III kwartale ukazał się zestaw wspierający nowe procesory Intel (IXCDP 1100, IXDP 425 oraz IXDP 2400), Broadcom (BCM 91125E, BCM 91250E i BCM 5690) Motoroli itp. stworzone z myślą o segmencie sieciowym. Rynek motoryzacyjny również rozwija się w podobnym tempie i wkrótce możemy się spodziewać między innymi następujących zestawów BSP: Hitachi Big Sur/Amanta, Hitachi SH 7760, Hitachi SystemH, Motorola „Redbox“ - Power PC 823e, Motorola MGT 5100.

Na koniec tego roku zapowiada się kolejna wersja systemu QNX Momentics 6.3.0 z nowymi kompilatorami GCC 3.x, obsługą USB 2.0, zewnętrznymi urządzeniami dyskowymi podpinanych przez interfejs USB oraz nowego środowiska programistycznego (Red Hat Linux 7.x/8.x).

Druga połowa roku 2003 zapowiada się naprawdę interesująco.

Zawartość CD

Na CD-ROM-ie dołączonym do czasopisma (wyłącznie w wersji EPo/oL) znajduje się najnowsza, przeznaczona do zastosowań niekomercyjnych wersja zestawu programistycznego QNX Momentics 6.2.1 NC. W skład pakietu wchodzi: system operacyjny QNX Neutrino 6.2.1, środowisko graficzne Photon wraz z programem Photon Application Builder, narzędzia do tworzenia własnych sterowników DDK s (z kodami źródłowymi po jednym na daną klasę), biblioteki ANSI C oraz narzędzia wiersza poleceń GNU dla platformy docelowej x86 oraz ARM (kompilator GCC v2.95x, GDB 5.x, Binutils 2.10.x).

Dla wszystkich zainteresowanych tworzeniem własnych aplikacji dla platformy docelowej ARM interesujące uzupełnienie stanowi pakiet BSP przeznaczony do instalacji na komputerach podręcznych iPAQ (iPAQ Reference Platform) z procesorem StrongARM SA-1110. Sposób instalacji systemu oraz konfiguracji skrosowanej platformy programistycznej został opisany szerzej na stronie: <http://www.qnxzone.com/ipaq>.

Jacek Rudnicki, KTT Quantum
jacek.rudnicki@quantum.com.pl

Dodatkowe informacje

Artykuł powstał na podstawie materiałów udostępnionych przez firmę Quantum, tel. (71) 362-63-56, www.quantum.com.pl.