

Czy bios jest małym systemem operacyjnym?

Robert Wijas, Michał Bażyński

9 stycznia 2004

1 Ogólna charakterystyka systemu BIOS

BIOS (Basic Input/Output System) jest oprogramowaniem najniższego poziomu. Jedną z jego podstawowych funkcji jest inicjalizacja sprzętu. Dostarcza on interfejs pośredniczący pomiędzy sprzętem a systemem operacyjnym. Pozwala także na konfigurację układów scalonych znajdujących się na płycie głównej oraz podstawową konfigurację urządzeń peryferyjnych.

1.1 BIOS ROM

Głównym składnikiem systemu BIOS jest BIOS ROM. Jest to pamięć zawierająca instrukcje systemu BIOS. Obecnie jest to najczęściej pamięć typu EEPROM (Electrically Erasable Programmable Read-Only Memory), która pozwala na zapis za pomocą oprogramowania (flash BIOS).

CMOS. Kolejnym ważnym składnikiem systemu BIOS jest pamięć CMOS. Jej głównym zadaniem jest przechowywanie konfiguracji sprzętu i systemu BIOS. Ustawienia BIOSu są przechowywane w pamięci CMOS (Complementary Metal Oxide Semiconductor). Jest to pamięć nie ulotna (NVRAM, non-volatile RAM). Po wyłączeniu zasilania jest ona podtrzymywana za pomocą niewielkiej baterii. Pamięć tego typu jest niewielkich rozmiarów (zazwyczaj 64 bajty), więc jedna bateria starcza na lata.

Suma kontrolna. Po zmianie parametrów biosu obliczana jest suma kontrolna. Dodawane są wszystkie bajty pamięci CMOS, a następnie zapisywany jest najmniej znaczący bajt tej sumy. Umożliwia to sprawdzenie poprawności danych odczytanych z pamięci CMOS. Taki test przeprowadzany jest zawsze podczas startu komputera. W przypadku stwierdzenia niepoprawności danych pamięci CMOS system BIOS przerywa działanie i wyświetlając stosowny komunikat, zazwyczaj CMOS Checksum Error.

1.2 Warstwowy model oprogramowania

Popularność platformy PC wynika zapewne z ogromnego wyboru spośród rozwiązań sprzętowych i programowych przy zachowaniu wzajemnej kompatybilności. Dodatkowo rozwiązania zastosowane w platformie PC dawały bardzo dobrą

kompatybilność wsteczną. Kluczem do zachowania kompatybilności jest wielowarstwowy model oprogramowania. Najniższą warstwą programową jest właśnie BIOS, który pośredniczy w komunikacji oprogramowania z wyższych warstw ze sprzętem.

1.3 Funkcje BIOSu, przerwania programowe

BIOS dostarcza wiele funkcji, które mogą być wołane przez oprogramowanie wyższego poziomu, czyli np. system operacyjny. Teoretycznie cała komunikacja pomiędzy systemem operacyjnym a sprzętem mogłaby się odbywać za pośrednictwem BIOSu. Jednak obecnie większość systemów pomija tą warstwę komunikując się bezpośrednio ze sprzętem. Powodem takiego postępowania jest chęć zwiększenia wydajności systemu. Funkcje BIOSu są wywoływane za pomocą przerw programowych. Przerwania programowe są podobne do przerw sprzętowych. Różnica polega na tym, że są one generowane przez oprogramowanie, a nie przez zewnętrzne urządzenia. Adresy funkcji obsługujących żądania są przechowywane w tablicy przerw, która jest inicjalizowana przez BIOS podczas startu systemu.

1.4 Program konfiguracyjny

Najpopularniejszym elementem systemu BIOS jest jego program konfiguracyjny. Rynek producentów systemów BIOS jest obecnie zdominowany przez American Megatrends, Inc (AMI) oraz Award Software. Programy konfiguracyjne różnych producentów różnią się interfejsem (tekstowy, graficzny, inny układ opcji itp.). Programy te posiadają jednak bardzo zbliżoną funkcjonalność.

1.5 Ochrona komputera

BIOS dostarcza prosty system ochrony komputera za pomocą haseł. System ten jest dwupoziomowy i składa się z hasła użytkownika i administratora. Hasło użytkownika wymagane jest do uruchomienia komputera. Hasło administratora chroni natomiast przed niepożądanymi zmianami konfiguracji BIOSu. Niestety system ten jest bardzo prymitywny i złamanie tych zabezpieczeń zabiera wprawionemu serwisantowi kilka minut.

1.6 Inne BIOSy

W komputerze PC znajduje się zazwyczaj kilka BIOSów. Oprócz głównego systemu BIOS, o którym traktuje ten dokument, wiele urządzeń peryferyjnych posiada własne (np. karta graficzna, twardy dysk).

2 Uruchomienie komputera

Jedną z najważniejszych funkcji BIOSu jest uruchomienie komputera oraz inicjalizacja sprzętu. Oto typowy scenariusz startu komputera:

- Wewnętrzne źródło zasilania zostaje włączone. Ponieważ źródło zasilania potrzebuje czasu by ustabilizować napięcie i móc dostarczyć energię całemu komputerowi, chipset generuje sygnał reset aż do momentu otrzymania sygnału 'Power Good' (PowerGood, Power OK, PWR OK) od źródła zasilania.
- Komputer jest już gotowy do wykonywania pracy. Niestety w pamięci brak rozkazów do wykonania. Na szczęście procesor wie gdzie szukać pierwszych rozkazów. Znajdują się one w pamięci ROM BIOSu.
- POST (Power-On Self Test). Pierwszą czynnością wykonywaną przez BIOS jest sprawdzenie poprawności działania sprzętu. Testy te trwają tak krótko, że zazwyczaj nie zauważamy ich do momentu wystąpienia problemu. Ponieważ w tej chwili nie została jeszcze zainicjalizowana karta graficzna BIOS informuje użytkownika o problemie za pomocą speakera. Odpowiednia kombinacja sygnałów dźwiękowych identyfikuje rodzaj awarii. Istnieje także dodatkowy kanał informowania o awariach sprzętu, który dostarcza bardziej szczegółowych informacji. Można z niego korzystać za pomocą specjalnej karty ISA.
- Aktywacja karty graficznej. Inicjalizacją zajmuje się BIOS wbudowany na karcie graficznej. Program ten zazwyczaj znajduje się w pamięci pod adresem C000h. Większość kart wyświetla informacje o sobie i jest to pierwsza rzecz pojawiająca się na monitorze.
- Uruchamianie BIOSów pozostałych urządzeń. BIOS szuka w pamięciach ROM pozostałych urządzeń ich BIOSów oraz uruchamia je. Na przykład dla dysków IDE/ATA BIOS znajduje się pod adresem C800h.

- Wyświetlenie informacji o starcie. Ekran ten zazwyczaj zawiera: wersję i nazwę producenta BIOSu, datę, nazwę klawisza uruchamiającego program konfiguracyjny, logo systemu, numer seryjny.
- Wykonywanie dodatkowych testów. Np. test pamięci, którego przebieg możemy zazwyczaj obserwować na ekranie. W przypadku wystąpienia błędów BIOS wyświetli stosowny komunikat tekstowy.
- Wykrywanie sprzętu. BIOS wykonuje kolejne testy by uzyskać jak najwięcej informacji o sprzęcie. Nowoczesne BIOSy potrafią automatycznie skonfigurować wiele elementów komputera np. pamięć, twarde dyski.
- Wykrywanie sprzętu Plug and Play.
- Wyświetlenie podsumowania konfiguracji komputera.
- Wyszukanie napędu startowego. Większość BIOSów pozwala na konfigurację wyszukiwania napędu startowego poprzez określenie sekwencji napędów.
- Po ustaleniu napędu startowego BIOS szuka informacji potrzebnych do rozpoczęcia startu systemu operacyjnego. W przypadku dysków twardej informacji te znajdują się w tzw. MBR (master boot record), który znajduje się na początku pierwszego sektora dysku.
- Start systemu operacyjnego.
- Jeśli BIOS nie znajdzie napędu bootowalnego następuje wstrzymanie działania oraz wyświetlenie komunikatu o błędzie (najczęściej: No boot device available).

Powyższy scenariusz opisuje zachowanie komputera po włączeniu. Ten proces nazywany jest zimnym uruchomieniem (cold boot), w przeciwieństwie do ciepłego uruchomienia (warm boot), które ma miejsce po restarcie komputera (np. Alt+Ctrl+Del). W tym przypadku następują kroki po POST.

3 LinuxBios

3.1 LOBOS

Lobos (Linux Os Boots OS) to linuxowy system call, który ładuje do pamięci inny kernel, i rebootuje system tak że nie jest wykorzystywany w ogóle bios, nie wychodząc w ogóle z 32-bitowego trybu dostępu do pamięci. Jest to zewnętrzny projekt, nie jest on częścią żadnego ze standardowych jąder Linux.

3.2 LinuxBios

LinuxBios to projekt mający na celu użycie linuxa jako biosu. Ogólna idea: skoro już potrafimy (lobosem) zrebootować komputer bez pomocy biosu używając linuxa, to spróbujmy na tyle na ile to możliwe zastąpić całkiem bios linuxem.

Co robi przy starcie zwykły bios?

- bios karty graficznej, inicjalizacja karty graficznej
- wyświetlanie na bieżąco informacji o działaniu
- odpalanie BIOSów innych urządzeń (dyski etc)
- inicjalizacja innych peryferiów (klawiatura, drukarki, cdromy etc)
- POST (power-on-self-test)
- testy pamięci
- dostęp do setupu
- poszukiwanie bootowalnego dysku (dyskietka/dysk/cdrom)

Ideą linuxbiosu jest wykonanie najmniejszej możliwej ilości z powyższych czynności i przekazanie sterowania od razu do kernela zapisanego na chipie biosu. Ten wstaje i sam wykonuje potrzebne operacje (wiele z rzeczy które zwykle robi bios potrafi robić kernel linuxa na przykład PlugAndPlay), po czym odpala LOBOS-em pożądaną system.

Właściwa część bootująca to 500 lini assemblera i 1500 lini kodu w C, sam kernel jest zwykłym linuxowym kernelem (z 10 liniami patchy).

Co zyskujemy?

- aby zbootować komputer nie potrzeba żadnych peryferiów - klawiatury, monitora, dysków, dyskietek czy cdromów. Z łatwo psujących się fizycznie elementów pozostaje więc jedynie wiatraczek procesora i zasilacz.
- dostajemy dużo większą kontrolę nad sposobem bootowania - docelowy system można zdobyć w każdy ze sposobów jakie zna linux - a więc w szczególności przez najróżniejsze rodzaje sieci i protokołów - na przykład przez ssh
- konfigurację biosu można robić zdalnie przez sieć. Ma to olbrzymie znaczenie w przypadku farm komputerów - w sytuacji gdy połączonych jest ze sobą parędziesiąt (czy nawet dużo więcej) komputerów, aby zmienić coś w biosie trzeba zwykle do każdego z nich podejść i podłączyć monitor i klawiaturę
- szybkość - aktualny rekord to zamontowanie root filesystemu w mniej niż sekundę, i pełen boot w mniej niż 3 sekundy. to pozwala myśleć o takich rozwiązaniach jak bootowanie systemu przed odpaleniem każdej aplikacji - każda aplikacja dostaje świeży (więc na pewno nie zepsuty przez poprzednie aplikacje) system, i dodatkowo każda aplikacja może dostać odrobinę inny kernel...

LinuxBios nie dostarcza pełnej biblioteki przerwań (nie dostarcza ich prawie wcale). Wynika to stąd że najnowsze systemy operacyjne właściwie przestały z nich korzystać i same komunikują się z peryferiami. Konsekwencją tego faktu jest to że LinuxBios nie potrafi zbootować dowolnego systemu operacyjnego-a. Aktualnie bootuje Linuxa, FreeBSD, oraz Windows 2000.

4 Czy bios to mały system operacyjny?

Aby odpowiedzieć na to pytanie najpierw trzeba zastanowić się co to jest system operacyjny.

W notatkach z wykładu *Systemy Operacyjne* znajdziemy definicję: „System operacyjny to program, który pośredniczy między użytkownikiem a sprzętem komputerowym”.

Zadania systemu operacyjnego:

- ukrywa szczegóły sprzętowe systemu komputerowego poprzez tworzenie abstrakcji (maszyn wirtualnych)

- zarządza zasobami
- dostarcza przyjazny interfejs: wygoda użycia (ustawianie przełączników, karty perforowane, taśmy perforowane, terminale graficzne z myszką i klawiaturą)

Jak do tej definicji ma się bios?

- pośredniczy pomiędzy użytkownikiem a sprzętem poprzez udostępnianie funkcji biosu służących do pracy ze sprzętem (dużą częścią biosu są funkcje do dostępu do hdd)
- ukrywa szczegóły sprzętowe poprzez tworzenie abstrakcji - dostęp do urządzeń jest taki sam we wszystkich biosach i przy różnych urządzeniach tego samego typu dzięki standardowym funkcjom biosu
- zarządza zasobami - można myśleć o biosie jako o jednozadaniowym OS-ie, w tym sensie zarządza on zasobami dość prostą strategią wyłącznego prawa dostępu. w tym punkcie widać jednak wyraźnie różnicę pomiędzy biosem i systemami operacyjnymi
- bios nie udostępnia środowiska w którym mogły by się wykonywać inne aplikacje w którym zapewnione już by było zarządzanie pamięcią, dostępem do procesora, etc.
- nie do tarcza przyjaznego interfejsu użytkownika

Za czasów DOS-a systemem operacyjnym było wedle powyższej definicji dopiero powiązanie biosu z Dosem. Dostęp do urządzeń schowany za interfejsem swoich funkcji dostarczał bios, a interfejsem użytkownika zarządzającym zasobami był Dos.

W najnowszych systemach operacyjnych odchodzi się od używania funkcji biosu, na rzecz bezpośredniej pracy ze sprzętem za pośrednictwem sterowników. W wyniku rola biosu coraz bardziej ograniczana jest do samego startu komputera, do podania procesorowi pierwszej instrukcji do wykonania. Doskonałym przykładem jest linuxBios - bardzo trudno tam stwierdzić czy biosem jest samo serce systemu (te 500 instrukcji assemblera) ładujące pierwszy kernel z pamięci biosu, czy też biosem należy w tym przypadku nazwać również sam kernel linuxa.

5 Aplikacja w środowisku BIOS

Wyobraźmy sobie, że mamy napisać pewną aplikację, która ma być wywołana bezpośrednio przez BIOS. Porównajmy powyższą sytuację z przypadkiem pisania aplikacji w systemie Linux.

Poniżej zestawione są mechanizmy realizujące podstawowe zadania systemu operacyjnego.

Cecha systemu	Linux	BIOS
Zarządzanie procesami	systemowy scheduler, wielowątkowość, wyłuszczanie	Tylko jeden proces
Zarządzanie pamięcią	Pamięć wirtualna, przestrzenie adresowe	Berpośredni dostęp do pamięci, funkcje BIOS
Przechowywanie danych	Wirtualny system plików	Funkcje BIOS
Obsługa urządzeń peryferyjnych	Sterowniki	Funkcje BIOS
Interfejs użytkownika	Funkcje systemowe, shell	Funkcje BIOS

Jak widać największą różnicą jest ilość procesów. Aplikacja biosowa jest jedynym uruchomionym procesem. Aby wystartować inne procesy lub wątki musi potrafić sama takie procesy/wątki stworzyć, musi być w stanie zarządzać dostępem nowych procesów do procesora i do pamięci. Dobry dla jednej aplikacji system bezpośredniego dostępu do pamięci będzie prawdopodobnie musiał być udoskonalony tak aby procesy-dzieci nie niszczyły pamięci głównej aplikacji i siebie nawzajem. Dostęp do urządzeń zewnętrznych, system plików i interfejs do nich mógłby pozostać w formie procedur biosu, ale bardzo prawdopodobne jest że wygodniej by było udostępniać procesom potomnym wygodniejsze mechanizmy.

Na podstawie tej krótkiej analizy możemy łatwo dojść do wniosku że ta biosowa aplikacja którą piszemy nazywana jest potocznie systemem operacyjnym.

Tak więc ostatecznym wnioskiem jest że bios nie jest systemem operacyjnym. Systemem operacyjnym jest aplikacja która umożliwia tworzenie na komputerze z biosem kolejnych procesów.

6 Linki

- <http://www.pcguides.com/>
- <http://www.bios.pl/>

7 O dokumencie

Dokument ten powstał w ramach zajęć z Systemów Operacyjnych w roku 2003/2004.