

Sprzęt a Linux

- [Płyta główna](#)
- [Katalog /proc](#)
- [Katalog /dev, UDEV, HAL](#)
- [Interfejsy dysków twardych](#)
- [IRQ - przerwania](#)
- [Adresacja I/O](#)
- [Lista urządzeń](#)
- [Katalog /sys](#)
- [UUID](#)
- [Kontrolery USB](#)
- [Moduły jądra](#)
- [Inne](#)
 - [DMA](#)
 - [Coldplug and hotplug devices](#)
 - [D-Bus](#)
 - [MBR \(BIOS\) vs GPT \(UEFI\)](#)
 - [Nazewnictwo prefiksów binarnych i dziesiętnych](#)

Płyta główna

Płyta główna jest bazowym elementem sprzętowym, który stanowi szynę dla wymiany informacji pomiędzy wszystkimi komponentami sprzętowymi na lokalnym serwerze lub PC.

Elementem płyty głównej jest również jej oprogramowanie - tzw. firmware - który obsługuje i pozwala konfigurować podzespoły do niej podłączone.

Firmware obecnych płyt głównych zapewniają:

- podstawową weryfikację poprawności działania sprzętu (POST Power On Self Test) oraz informowanie o błędach;
- obsługę sprzętu w początkowej fazie uruchamiania systemu operacyjnego;
- monitorowanie zasilania oraz temperatury;
- inicjalizację kodu uruchamiającego system operacyjny z zdefiniowanego nośnika;
- możliwość aktualizacji firmware z nośnika USB, z Internetu;
- interfejs dla użytkownika do konfiguracji, na przykład:
 - czasu zegara systemowego,
 - włączanie lub wyłączanie modułów i komponentów sprzętowych;
 - kolejności użycia nośników do uruchomienia systemu,
 - częstotliwości procesora,
 - wzbudzania alarmy przy braku podłączonych komponentów,
 - określenia jak ma zachować się system po utracie zasilania,
 - włączenie zasilania o określonym czasie.

Popularne interfejsy do firmware:

- BIOS - Basic Input/Output System (do 2011 roku)
- EFI - Extensible Firmware Interface (Intel)
- UEFI - Unified EFI (nazwa używana po tym jak rozwój EFI przekazano organizacji Unified EFI Forum złożonej z wielu firm, nie tylko Intela)

Katalog /proc

W systemie Linux istnieje wirtualny katalog `/proc`. Nie reprezentuje on przestrzeni dysku twardego czy innego nośnika danych. Prawie wszystkie pliki w nim zamieszczone mają rozmiar 0 bajtów, mimo to można wyświetlić ich zawartość. Przykładowo:

```
itmz ~ # ls -l /proc/cpuinfo
-r--r--r-- 1 root root 0 pa 25 20:20 /proc/cpuinfo
itmz ~ # head -n 5 /proc/cpuinfo
processor      : 0
vendor_id    : GenuineIntel
cpu family   : 6
model        : 42
model name   : Intel(R) Core(TM) i7-2620M CPU @ 2.70GHz
```

Pliki w `/proc/` reprezentują interfejs dostępu do informacji dotyczących procesorów i sprzętu kontrolowanego przez jądro system. Na przykład:

- `/proc/cpuinfo` - użyte procesory
- `/proc/PID/` - po jednym katalogu dla każdego działającego procesu w systemie, w katalogu znajdują się informacje odnośnie procesu
- `/proc/dma` - zarejestrowane kanały DMA
- `/proc/filesystems` - obsługiwane systemy plików
- `/proc/interrupts` - przypisania przerwania sprzętowych (IRQ)

- `/proc/iomem` - mapa zarezerwowanych zakresów pamięci operacyjnej
- `/proc/ioports` - mapa zarezerwowanych zakresów portów wejścia/wyjścia dla urządzeń
- `/proc/irq/NR/` - katalogi z informacjami dla poszczególnych numerów przerwania
- `/proc/loadavg` - load average systemu
- `/proc/mdstat` - aktualna konfiguracja i status macierzy RAID
- `/proc/meminfo` - szczegóły wykorzystania pamięci operacyjnej
- `/proc/modules` - załadowane i użyte moduły jądra systemu
- `/proc/partitions` - obsługiwane partycje
- `/proc/uptime` - uptime systemu
- `/proc/version` - wersja użytego jądra
- `/proc/vmstat` - statystyki wirtualnej pamięci

Forma informacji pobranych z `/proc` może być czasami nieczytelna, np. `cat /proc/uptime`. Dlatego w takich przypadkach lepiej używać oprogramowania, które poprawnie zinterpretuje informacje przedstawiane przez jądro - komenda `uptime`.

```
cat /proc/uptime
uptime
```

Niektóre pliki zawarte w katalogu `/proc` można modyfikować a przez to sterować funkcjonalnością urządzeń (wpływać na sterowniki zawarte w jądrze systemu). Przykładowo, aby włączyć forwardowanie pakietów pomiędzy interfejsami (gdy Linux ma działać jako router) wystarczy zapisać liczbę 1 w odpowiednim pliku:

```
cat /proc/sys/net/ipv4/ip_forward
echo 1 > \ /proc/sys/net/ipv4/ip_forward
```

Katalog `/dev`, UDEV, HAL

Katalog `/dev`, podobnie jak `/proc` jest wirtualnym katalogiem niepowiązanym z przestrzenią na nośniku danych.

Każde urządzenie obsługiwane przez system Linux (w tym dyski twarde i partycje) jest reprezentowane w systemie jako specjalny plik (ang. special file or device file). Plik ten stanowi interfejs dostępu między oprogramowaniem a sterownikiem obsługującym dane urządzenie. Dzięki temu każdy program może odwoływać się do urządzenia tak jakby był on zwykłym plikiem.

Wszystkie pliki urządzeń w systemie znajdują się specjalnym folderze `/dev` (ang. device). Przykładowo, drugi dysk podłączony do systemu może przyjąć nazwę `/dev/sdb`. Przykładowo, aby pozakładać partycje na tym dysku komenda do partycjonowania (np. `fdisk`) musi odwołać się do pliku `/dev/sdb`.

```
fdisk /dev/sd
```

Obecnie całą budowę specjalnych plików w katalogu `/dev` zajmuje się podsystem **UDEV**. UDEV może być konfigurowany przez administratora i opiera się na regułach (udev rules). UDEV rozpoznaje zmiany stanu urządzeń (np. podłączenie urządzenia USB) i na podstawie reguł definiowanych przez administratora może wykonać określone operacje. Na przykład przy podłączeniu aparatu fotograficznego urządzenie może pojawić się w systemie plików jako `/dev/aparatfoto`, urządzenie może być automatycznie podmontowane pod określony katalog a nawet może być uruchomiony określony skrypt (np. przenoszenia plików z karty aparatu do systemu).

Demon `hal` (Hardware Abstraction Layer Daemon) swego czasu tworzył **warstwę abstrakcji sprzętowej** tak, aby standaryzować interfejs obsługi urządzeń tego samego typu różnych producentów. Obecnie funkcjonalność demona `hal` przejął UDEV - nie ma go we współczesnych dystrybucjach.

Interfejsy dysków twardych

Podstawowe interfejsy dysków twardych podłączanych do płyt głównych:

- IDE, ATA, PATA
 - IDE (Integrated Drive Electronics)
 - ATA (Advanced Technology Attachment)
 - PATA (Parallel ATA) - wcześniejsze ATA, równoległa ATA
- SATA (Serial ATA) - szeregową ATA (obecny standard PC)
- SCSI (Small Computer Systems Interface)
- SAS (Serial Attached SCSI) - obecny standard serwerowy

W zależności od typu kontrolera dyskowego nazwy plików dysków twardych przyjmują następujące nazwy:

- IDE, ATA, PATA: `/dev/hdX`
- SATA, SAS: `/dev/sdX`
- SCSI:
 - hard drive `/dev/sdX`
 - stream tape drive (streamer) `/dev/stX`
- cd-roms `/dev/scdX`

IRQ - przerwania

Urządzenia do komunikacji z CPU i powiadomienia go o tym, że potrzebują obsługi używają przerw. Dawniej urządzenia trzeba było ręcznie przypisywać do określonego numeru przerwy IRQ (Interrupt Request).

Przykładowe podstawowe urządzenia i porty oraz przypisane do nich przerwy (przypisanie to stanowi standard):

- IRQ1 – klawiatura
- IRQ2 - COM2 i COM4
- IRQ3 - COM1 i COM3
- IRQ6 - stacja dyskietek
- IRQ12 - mysz PS/2

Dawniej przypisanie linii przerwy do urządzeń realizowało się przez przestawienie na płycie głównej odpowiednich zworek (jumpers) zgodnie z instrukcją płyty głównej. Później wprowadzono możliwość programowej konfiguracji w interfejsie BIOS. Następnie, dzięki kontrolerowi APIC (Advanced Programmable Interrupt Controller) rozszerzono obsługę do 255 urządzeń jednocześnie (kiedyś tylko 16).

Aktualną listę przerw wraz z liczbą wykonanych wywołań dla każdego procesora można obejrzeć wywołując poniższą komendę:

```
cat /proc/interrupts
```

Adresacja I/O

Urządzenia wejścia/wyjścia (I/O) muszą komunikować się z systemem oraz muszą czasami korzystać z pamięci. Do obsługi tej komunikacji wykorzystywane są porty I/O:

```
cat /proc/ioports
```

Zakres pamięci przydzielona każdemu z urządzeń można obejrzeć na mapie pamięci IO:

```
cat /proc/iomem
```

Przed erą Plug-and-Play zakres adresacji należało ustawiać w BIOS dla każdego urządzenia, aby nie współdzieliły tych samych adresów i nie pojawiły się konflikty.

Lista urządzeń

Do wyświetlenia listy podłączonych urządzeń do płyty głównej czy przez interfejs USB przydają się dwie poniższe komendy:

```
lspci [-v]
lsusb [-v]
```

Katalog /sys

Kolejny katalog wirtualny `/sys` ([Sysfs](#) - sys file system) jest kontenerem w systemie plików zawierającym reprezentantów urządzeń, sterowników oraz innych podsystemów jądra zarządzanych przez niego. W katalogu `/sys` można znaleźć podziały na różne kategorie. Przykładowo `/sys/fs` posiada wewnątrz podkatalogów podział na wykorzystywane obecnie systemy plików.

d-bus – interfejs komunikacji między programami, używany również jako interfejs do powiadamiania o podłączanych lub odłączanych urządzeniach

UUID

Urządzenia są reprezentowane przez unikalny identyfikator UUID (Universally Unique Identifier). Można go odczytać za pomocą komendy

```
blkid DEVICE
```

Kontrolery USB

Obecne kontrolery USB i nazwy modułów w nawiasach:

- Universal Host Controller Interface (UHCI) USB 1.0 f-my Intel
- Open Host Controller Interface (OHCI) USB 1.1
- Enhanced Host Controller Interface (EHCI) USB 2.0
- Extensible Host Controller Interface (XHCI) USB 3.0

Moduły jądra

Jednym z głównych funkcjonalności jądra systemu jest obsługa urządzeń. Obsługa ta jest możliwa dzięki sterownikom, które mogą być wkompiłowane na stałe w jądro systemu albo mogą być „ładowane” do jądra już w trakcie działania systemu. Ta druga opcja jest o tyle wygodna, że samo jądro nie zajmuje wtedy dużo miejsca w pamięci, gdyż użyte są tylko te sterowniki, które rzeczywiście systemowi są potrzebne. Sama operacja ładowania modułów obecnie przeprowadzana jest automatycznie. Administrator ma jednak możliwość załadowania dodatkowego modułu (włączenia określonego sterownika)

```
modprobe MODUL
```

lub wyłączenia modułu z pamięci (wyłączenie sterownika):

```
rmmod MODUL
```

Listę załadowanych obecnie modułów wyświetla komenda

```
lsmod
```

Inne

DMA

DMA (Direct Memory Access) umożliwia bezpośrednie zapisywanie w pamięci danych przez urządzenie bez udziału CPU. Listę zarejestrowanych kanałów DMA można wyświetlić za pomocą:

```
cat /proc/dma
```

Coldplug and hotplug devices

System Linux wspiera urządzenia obu typów:

- coldplug devices – wymagają wyłączenia systemu przed ich podłączeniem/odłączeniem;
- hotplug devices – urządzenia, które mogą być podłączane (a niektóre odłączane) podczas działania systemu (rozwiązania serwerowe wysokiej dostępności).

D-Bus

D-Bus - system komunikacji między procesami - w systemie Linux stanowi zestaw podsystemów złożonych z bibliotek i demona, dzięki którym aplikacje mogą lepiej współpracować ze sobą. D-Bus zapewnia szynę wymiany informacji pomiędzy programami i może na przykład informację pochodzącą z jednej aplikacji przesłać do wielu innych aplikacji równolegle. Jest stosowany w GUI GNOME czy XFCE.

MBR (BIOS) vs GPT (UEFI)

Przez wiele lat obsługę partycji w systemach plików oparta była o Master Boot Record (32-bitowe adresowanie). Charakterystycznymi cechami MBR jest:

- 4 partycje podstawowe (np. /dev/sda1-4)
- wiele partycji logicznych w ramach jednej podstawowej (numerowanie od 5 – np. /dev/sda5)
- tylko jedna partycja aktywna (bootowalna)
- limit obsługi do 2TB / partycja (przy 512 bajtach na sektor)

Narzędzia do zakładania partycji z MBR: `fdisk`, `cfdisk`, `parted`, graficzny `gparted`

GUID Partition Table to nowe, 64-bitowe adresowanie, które charakteryzuje się:

- brakiem podziału na partycje podstawowe/logiczne
- limit obsługi znacznie wyższy niż w MBR - teoretycznie 9.4 ZB (przy 512 bajtach na sektor)

Narzędzia do zarządzania partycjami: `gdisk`, `parted`, graficzny `gparted`

Nazewnictwo prefiksów binarnych i dziesiętnych

kilobajt kB to 10^3 bajtów (1000^1) czyli 8000 bitów

kibibajt KiB to 2^{10} bajtów (1024^1) czyli 8192 bitów

megabajt MB to 10^6 bajtów (1000^2) czyli 8000000 bitów

mebibajt MiB to 2^{20} bajtów (1024^2) czyli 8388608 bitów

[Orders of magnitude](#)