Daemon ACPI w systemie operacyjnym Linux  
Dokumentacja

Paweł Kwieciński

# Czym jest ACPI

ACPI (ang. Advanced Configuration and Power Interface, zaawansowany interfejs zarządzania konfiguracją i energią) - otwarty standard opracowany przez firmy Intel, Microsoft i Toshiba do których później dołączyły HP oraz Phoenix, po raz pierwszy ogłoszony w grudniu 1996.  
  
ACPI zwiększa możliwości systemów zarządzania energią oferowane przez wcześniejsze rozwiązanie APM. Umożliwia systemowi operacyjnemu kontrolowanie ilości energii dostarczanej do poszczególnych urządzeń komputera (napędu CD-ROM, twardego dysku oraz urządzej peryferyjnch) i podobnie jak APM umożliwia ich wyłączanie, gdy nie są używane, a konieczne jest oszczędzanie energii.

Główną cechą różniąca APM i ACPI jest to, że w przypadku APM kontrolę nad urządzeniami miał BIOS komputera. Natomiast ACPI przekazuje możliwość zarządzania urządzeniami systemowi operacyjnemu.

# Stany ACPI

## Stany globalne

Specyfikacja ACPI definiuje następujące siedem tak zwanych stanów globalnych, w których może znajdować się system komputerowy kompatybilny z ACPI:

* + - **G0 (S0)**, Pracuje (Working) jest domyślnym stanem pracy komputera, oznaczającym, ze zarówno system operacyjny, jak i dowolne uruchomione na nim programy działają, a procesor (lub procesory) normalnie wykonuje polecenia.
    - **G1**, Uśpiony (Sleeping) dzieli się na dalsze stany, oznaczane S1 do S4.
      * **S1** to najbardziej "energożerny" stan uśpienia. Podczas przejścia w ten stan, czyszczone są pamięci podręczne procesora, a procesor przestaje wykonywać polecenia. Zachowany jest dopływ energii do procesora i pamięci, a urządzenia, które nie zgłaszają konieczności utrzymania dopływu energii są wyłączane.
      * **S2** jest stanem głębszego uśpienia niż S1. przejście w ten stan oznacza odłączenie zasilania od procesora.
      * **S3** to stan w którym utrzymywany jest dopływ prądu do pamięci RAM w celu zachowania jej zawartości. Stan ten znany jest również jako uśpienie (Sleep), czuwanie (Standby) lub Suspend to RAM (STR)
      * **S4** polega na zapisywaniu zawartości pamięci RAM w pamięci trwałej komputera (np. dysku twardym), tak aby później można było odtworzyć jej stan. Stan ten znany jest powszechnie jako hibernacja (Hiberantion) lub Suspend to Disk
    - **G2 (S5)** Programowo Wyłączony (Soft Off) to stan taki jak G3 (opisany niżej) z tą różnicą, że pod zasilaniem znajduje się co najmniej przycisk power, który pozwala na powrót do stanu G0. Wymaga to pełnej procedury bootowania, ponieważ w G2 stan urządzenia nie jest w żaden sposób zachowywany. Dodatkowo w tym stanie zasilanie może być dostarczane do takich urządzeń jak karta LAN czy USB w celu uruchomienia urządzenia (np. Wake on Lan).
    - **G3** Mechanicznie wyłączony (Mechanical Off)jest stanem w którym zasilanie jest całkowicie odcięte w sposób fizyczny (np. za pomocą przycisku na zasilaczu)

## Stany urządzeń

Stany urządzeń (D0 - D3) definiowane są w następujący sposób. Konkretne znaczenie daneog stanu zależy od urządzenia.

* + - **D0** W pełni działający (Fully On) - urządzenia działa w zwykły sposób
    - **D1** i **D2** to pośrednia stany zasilania, których znaczenie zależu od danego urządzenia
    - **D3** dzielący się na dwa podstany:
      * **D3 Hot** (pod zasilaniem) - urządzenie w tym stanie może przejść w wyższe stany w reakcji na sygnały ACPI
      * **D3 Cold** (brak zasilania) - urządzenie ma fizycznie odcięte źródło zasilania

## Stany procesora

Specyfikacja ACPI definiuje również stany zasilania procesora (C0 - C3):

* + - **C0** to stan standardowe pracy
    - **C1** Zatrzymany (Halt) jest stanem, w którym procesor nie wykonuje rozkazów. Powrót z tego stanu do **C0** jest praktycznie natychmiastowy
    - **C2** Zatrzymanie zegara (Stop-Clock). W tym stanie procesor zachwuje wszystkie dane widoczne dla oprogramowania (cache, rejestry), ale powrót do **C0** trwa dłużej.
    - **C3** Uśpienie (Sleep) - procesor nie musi zachowywać zawartości pamięci cache. Istnieją różne warianty C3, takie jak Deep Sleep, Deeper Sleet, itd.

## Stany wydajności

Stany wydajności (**P0** - **Pn**) określają różne tryby pracu urządzenia lub procesora różniące się od siebie ilością poboru mocy i wydajności pracy. Maksymalnie może istnieć 16 takich stanów dla danego urządzenia. Konkretne znaczenie danego stanu zależy od implementacji.

* + - **P0** - maksymalna wydajność, maksymalny pobór mocy
    - **P1** - wydajność niższa niż P0, pobór mocy niższy niż P0
    - **P2** - wydajność niższa niż P1, pobór mocy niższy niż P1
    - …
    - **Pn** - wydajność niższa niż P(n-1), pobór mocy niższy niż P(n-1)

Przykładem implementacji stanów wydajności są technologie stosowane w dzisiejszych procesorach. Są to m.in. SpeedStep (Intel), czy Cool’n’Quiet (AMD).

# Daemon ACPI (acpid)

W systemach operacyjncyh linux za odczytywanie oraz reagowanie na sygnały odpowiada program **acpid.** Działa on jako daemon systemowy i jest uruchamiany wraz ze startem systemu.

## Zasada działania

Po wystartowania daemona ACPI otwiera on plik **/proc/acpi/event** z którego czytane są wydarzenia ACPI raportowane przez komputer i urządzenia. Plik ten jest czytany linia po pinii przed acpid.

W przypadku, gdy plik ten nie istnieje acpid próbuje połączyć się z kernelem (za pomocą Netlink) i z tamtąd pozyskiwać informacje.

W momencie odczytania wydarzenia, acpid próbuje znaleźć dla niego pasującą **regułę (rule)**, która następnie jest uruchamiana.

## Reguły (Rules)

Reguły acpid zapisane są w plikach konfiguracyjnych, którch acpid domyślnie szuka w folderze **/etc/acpi/events**. Pliki z regułami powinny składać się z małych i dużych liter, podreśleń oraz myślników. Pliki zawierające inne znaki są przez acpid ignorowane.

Każdy z plików reguł powinien zawierać dwa elementy **zdarzenie (event)** oraz **akcję (action)**. Do pliku można również dodawać komentarze poprzedzając daną linię znakiem ‘#’.

Każda linia powinna składać się z trzech elementów:

* + - klucza - do 63 znaków, case-insensitive
    - znaku ‘=’
    - wartości - do 511 znaków, case-sensitive

## Zdarzenie (event)

Jest to wyrażenie regularne, które acpid porównuje do nazwy przechwyconego zdarzenia. W ten sposób reguły dopasowywane są do wydarzeń ACPI.

## Akcja (action)

Wartość akcji definuje co ma się wydarzyć w momencie zajścia danej reguły. Wartość akcji jest interpretowana przez daemona jako polecenie, którye zostanie wykonane za pomocą **/bin/sh**.

Wartość akcji może posiadać specjalny symbol - “%e” za który zostanie podstawiona przez acpid nazwa zdarzenia które uruchomiło daną regułę.

## Socket acpid

Oprócz wspomnianego wyżej mechanizmu przetwarzania reguł na podstawie plików konfiguracyjnych daemon ACPI oferuje również możliwość odczytywania wydarzeń za pomocą socketów.

acpid domyślnie pozwala połączyć się z socketem **/var/run/acpid.socket** na który przesyłane są wszyskie przechwycone zdarzenie.

Pozwala w łatwy sposób uzyskać programowy dostęp do zdarzeń ACPI.

## acpi\_listen

W celu uproszczenia dostępu do eventów ACPI można skorzystać z programu **acpi\_listen**, który odczytuje zdarzenia ze wspomianego wyżej socketa i przekazuje je na standardowe wyjście.

# Przykład - konfiguracja acpid

## Plik konfiguracyjny

Najprostszy możliwy plik ACPI może wyglądać w następujący sposób:

event=button/power

action=/etc/acpi/power.sh "%e"

Pierwsza linia oznacza, że zdarzenie, na które chcemy zareagować to wciśnięcie fizycznego przycisku power na komputerze.

Druga linia mówi jakie polecenie chcemy wykonać w reakcji na ten event. W tym wypadku chcemy uruchomić skrypt **power.sh** do którego jako parametr przekazana zostanie nazwa eventu.

Plik ten powinien zostać zapisany w katalogu **/etc/acpi/events/** (jeśli nie zmieniliśmy domyślnej lokalizacji).

## Skrypt akcji

Przkładowa zawartość pliku pliku **power.sh** może być taka jak poniżej:

#!/bin/bash

/sbin/shutdown -h now "Power button pressed"

Nietrudno się domyślić, że rezultatem będzie wyłączenie komputera z podaniem jako wiadomości przyczyny - wciśnięcia przyisku.

# Przykład - acpi\_listen

$ acpi\_listen

button/power PWRB 00000080 00000004

button/lid LID0 00000080 00000023

button/lid LID0 00000080 00000024

Powyższy przykład przedstawia program **acpi\_listen** w działaniu. Po jego uruchomieniu zostały wykonane trzy akcje. Wciśnięty przycisk POWER, zamknięta, a następnie otwarta pokrywa laptopa. Efekt jest taki jak by się można było spodziewać. Zdarzenia te są wypisywane na wyjście standardowe.