

# 知能情報工学科の講義用端末の節電について

情報工学部 技術部 松元隆二

平成 20 年 8 月 26 日

## はじめに

近年、本学では節電が呼びかけられているが、教員室や研究室では構成や機器の問題で、節電はなかなか進んでいない。しかし、各学科や大学共同設備の情報科学センター/情報基盤室などに設置されている端末は、非常に多くの端末が/同一構成で設置され/定まった時間帯のみに運用される事が多いため、僅かな手間で、節電を行なう事が可能である。今回の発表では、知能情報工学科の学生実験端末などに設定した節電設定について述べる。

知能情報工学科では学生実験および、学生の自習用に端末室を設置している。各端末室の設置台数は下記の通りである。

- 2006 年度導入 (Linux/Windows 共用)
  - － A 教室: 28 台 (日中のみ常時解放)
  - － B 教室: 8 台 (常時解放)
  - － C 教室: 28 台 (講義中のみ解放)
- 2008 年度導入 (Linux 専用)
  - － D 教室: 25 台 (講義中のみ解放)

合計 89 台。利用者には、端末を使い終わった後は電源を落すよう指導しているが、電源を付けたまま退室する利用者が非常に多いため、自動 poweroff を導入した。まず始めに、端末の節電について第 1 章で述べる。

次に、端末は単独で動作しているわけではなく、多くのサーバの支援の基で動作している。例えばファイルサーバやユーザ認証サーバなどが必要である。また知能情報工学科や情報科学センターの端末は管理の効率化のため HDD 無しの特異構成 (diskless 構成) であり、起動時に OS 起動ファイルを提供する起動支援サーバが必要になる。これらのサーバの運用は 24 時間運転の場合が多く、地球温暖化に貢献している。

具体的には、知能情報工学科の端末は下記のサーバの支援の基、動作している。

- A/B 教室用
  - － DHCP サーバ: 1 台 / ファイルサーバ: 1 台 / ユーザ認証サーバ: 2 台
  - － Windows 起動支援サーバ: 3 台 / Linux 起動支援サーバ: 2 台
- C 教室用
  - － Linux 起動支援サーバ: 1 台
- D 教室用
  - － Linux 起動支援サーバ: 2 台

講義中はすべてのサーバを起動させる必要があるが、設計を適切に行えば講義時間外は台数を減らす構成が可能である。サーバの節電について第 2 章で述べる。次に節電型の PC 導入について第 3 章で述べ、最後に HDD の節電について第 4 章で述べる。

## 1 学生実験室の端末の節電

学生実験端末は電源を付けたまま退出する学生が非常に多い。これは知能情報工学科だけの問題だけでなく情報科学センターや教務情報システムのロビー端末で数多く見られるので、教育を行なっても限界があると考ええる。そのため数年前に端末が一定時間未使用の場合は自動的に poweroff を行う機能を追加した。但し Linux のみの対応である。本学科での講義利用 OS は Linux が多いため、実質的にはほとんど問題にならない。

次に端末室の解放時間および講義スケジュールによるタイマー制御も行っている。タイマー制御は Linux だけでなく Windows も制御を行っている。利用時間外になると自動的に poweroff するようになっている。

### 1.1 端末利用中か否かの判断方法

端末利用中か否かの判断基準として、キーボードおよびマウスの利用頻度を採用した。当初はユーザプロセス等の動作状況で判断する方法も検討したが、バックグラウンドで実行して放置されているプロセスがあると判断が難しく、また過去の例として、1997 年～2000 年に導入した Pentium 120MHz の全端末 35 台に学生が無断で「地球外知的生命体探査プロジェクト/ seti@home」<sup>1</sup>をインストールした事があり、常時すべての端末に負荷がかかっている状態になった。同様の事例を考えるとプロセスの動作状況での判断は誤認識の可能性が高いと考え見送った。

なお、過去の事例として、ユーザプロセスの監視による運用がされたことがある。当時のデータは残っていないが、記憶によると、1993 年～1996 年に知能情報工学科で導入した学生実験端末 Sun3/60 (SunOS4) に木村氏らによって導入された。当時のコンピュータは OS 側から電源を切ることができず、OS の終了操作を行なった後に、物理的なスイッチの Off が必要であったため、外付けの電源制御装置を必要とした。

### 1.2 キーボードとマウスの利用頻度による判断の問題点

キーボードの上に荷物等を置いておくと、常にキーが押されている状態になり、常に使っていると判断されるため、キーボードとマウスの両方のデバイスを使っている場合のみ、利用中と判断する事にした。マウスについても微かな震動等で使っていると判断される場合があるようである。秒間の利用頻度で判断すると良い。

次に、講義によっては説明等で 1 時間以上をかける場合がある。当初はキーボードとマウスを 1 時間未使用で poweroff にしていたため、説明中に端末の poweroff が発生するという障害が発生した。説明が 2 時間以上続く事は想定しにくいと考え、講義時間中は 2 時間未使用で poweroff に設定した。具体的には講義が行なわれる可能性がある月～金曜日の 3,4,5 限目は 2 時間で poweroff。それ以外の時間は 1 時間で poweroff を行なうようにした。

次に、身障者の学生はキーボードを一切使わず特殊なマウスで入力を行なう場合がある。つまりマウスだけ使ってキーボードは使わないという状況になったため、講義中に自動 poweroff が発生してしまった。そのため、身障者専用の端末を設け、自動 poweroff を無効にした。

### 1.3 キーボード/マウスの利用中か否かの判断方法

PS/2 キーボードおよびマウスは、割り込み駆動型のデバイス<sup>2</sup>である。キーボード等を操作すると割り込みが発生する。そのため利用中か否かの判断は割り込み数の増加頻度を監視すればよい。今回は利用頻度の監視のためリアルタイムで処理する必要はないため、キーボード等の割り込み発生数をポーリング方式で監視した。Linux では割り込みの発生数は proc ファイルシステム「/proc/interrupts」で取得可能であるため、カーネルモードのドライバの開発の必要はなく、ユーザランドのプログラムで可能である。

<sup>1</sup> インターネットを介して分散コンピューティング環境を構築するプロジェクトである。クライアント実行者の CPU の空き時間を使って電波望遠鏡のデータを解析し、地球外知的生命体からの信号を探すことを目的としたプロジェクト。現在では seti@home は単独のプロジェクトとしては終了し、BONIC プロジェクトに統合された。本プロジェクトへの参加意義については賛否両論あるだろうが、後述の第 3 章で述べるように CPU の空き時間とはいえ、負荷が上がると消費電力が増える。本学は節電が求められているため、研究等で必要な場合を除き、実行は禁止すべきであろう。

<sup>2</sup> デバイスには大分類して割り込み駆動型と、ポーリング型がある。割り込み駆動型の場合は、キーボード等が動いたら、動いたという指示を割り込みを使い OS に与える。ポーリング型の場合は、キーボード等が動いたら動いたという情報をメモリに保存して沈黙し、OS 側で一定間隔でメモリが更新されてないか監視する。

割り込み発生数を表す proc ファイルシステム「/proc/interrupts」(PS/2 キーボード・マウス)

```

CPU0
0: 2888109 XT-PIC timer
1: 887 XT-PIC i8042
2: 0 XT-PIC cascade
5: 0 XT-PIC uhci_hcd:usb4
7: 704 XT-PIC Intel ICH6
8: 1 XT-PIC rtc
9: 1 XT-PIC acpi
10: 575147 XT-PIC eth0, uhci_hcd:usb3, ohci1394
11: 0 XT-PIC uhci_hcd:usb1, uhci_hcd:usb2, ehci_hcd:usb5, sdhci:slot0
12: 7419 XT-PIC i8042
15: 51 XT-PIC ide1

```

上記は一般的な PC の割り込み線の一覧である。一番始めの列 (0 ~ 15) の数字は割り込み線 (割り込みの識別番号/IRQ) である。二列目の数字は割り込みの発生数である。三列目は割り込みの実装、四列目は割り込みを割り当てられているデバイス (デバイスドライバ) である。

PS/2 キーボード (i8042) は割り込み線が 1、PS/2 マウス (i8042) は割り込み線が 12 である<sup>3</sup>。この二つの割り込みの発生頻度を監視すれば、利用頻度が確認出来る<sup>4</sup>。

なお、この割り込み発生数による実装は純粋な PS/2 キーボード/マウスでは容易であるが、USB キーボード/マウスでは割り込み線の区別や割り込みの発生数のカウントが難しい<sup>5</sup>。具体的には USB では次のようになり、

USB キーボード・マウス搭載端末の割り込み発生数の例

```

CPU0
0: 285513 XT-PIC timer
2: 0 XT-PIC cascade
5: 0 XT-PIC uhci_hcd:usb3, ehci_hcd:usb7
8: 3 XT-PIC rtc
9: 3 XT-PIC acpi, uhci_hcd:usb5, ehci_hcd:usb6
10: 15 XT-PIC uhci_hcd:usb2, uhci_hcd:usb4, libata, libata
11: 383690 XT-PIC eth0, uhci_hcd:usb1, HDA Intel

```

割り込み線 1 と 12 が無くなる。加えて、PS/2 のように特定の割り込み線にはならない。ほとんどの PC ではホスト側 (PC 本体側) の USB コントローラーが多数ある。Linux では「uhci\_hcd:usbX」が USB コントローラーのデバイスドライバである。例では USB コントローラーは 6 個 (X が 1 ~ 6) 存在する。USB コントローラーそれぞれが USB-HUB を持ち、キーボード等を接続した USB-HUB により、割り込み線が変化する。そのため割り込み線の判別が難しい。また PC の機種により USB コントローラーの数が異なり、更に一つの割り込み線を複数のデバイスで共有しているため、さらに判別が難しい。現状では USB キーボードへの対応は行っていない。<sup>6</sup>

Linux のデバイスドライバの開発については参考文献 [1, 2] を参照して欲しい。

## 1.4 実際の運用

具体的には Linux は次のような運用スケジュールになっている。

(凡例 S1: 1 時間未使用で停止 / S2: 2 時間未使用で停止 / poweroff: 利用中であっても停止)

平日	1,2 限目	3,4,5 限目	夜間	深夜
A 教室 (日中のみ常時解放)	S1	S2	poweroff	poweroff
B 教室 (常時解放)	S1	S2	S1	poweroff
C 教室 (講義中のみ解放)	S1	S2	poweroff	poweroff
D 教室 (講義中のみ解放)	S1	S2	poweroff	poweroff

<sup>3</sup> 「i8042」というのはキーボードコントローラー。Linux のバージョンによっては「i8042」ではなく「keyboard」・「mouse」と表示。

<sup>4</sup> 原理的には Windows でも Linux 同様に割り込みの監視による利用頻度の確認が可能であると考えられるが、現状では対策を行っていない。

<sup>5</sup> 正確にはわからないが、2006 年頃から発売されている PC では PS/2 コネクタが内蔵されていても内部で USB に変換されている場合がある。その場合は USB と同等の扱いになる。

<sup>6</sup> 現在検討中であるが、Linux の場合、USB キーボード/マウスはデバイスドライバ「usbhid」(USB Human Interface Device) を使うため、これに噛み付く事を検討している。動作状況をユーザランドから取得するには専用の /proc ファイルを定義すれば良い。

Window は次のようなスケジュールになっている。これは市販ソフトウェアによるタイマー制御である。キーボード・マウスが利用中か否かの判断は行なっていない。

平日	1-5 限目	夜間	深夜
A 教室 (日中のみ常時解放)	通常運転	poweroff	poweroff
B 教室 (常時解放)	通常運転	通常運転	poweroff

なお、一部より、5 限目を超えて実験が出来ないという指摘が出ているが、知能情報工学科の規則により学生実験は延長した場合であっても午後 6 時までに終了する事になっているため、対応を断わっている。(実際は常時解放している B 教室に移動するようお願いしている。)

## 2 サーバの節電

はじめに述べたように、講義時間外は閉鎖する教室用のサーバを常時起動する必要はない。具体的には講義時間外は閉鎖している C 教室および D 教室の Linux 起動支援サーバは講義時間外は停止できるよう設計を行った。

なお、本章でこれから述べる方法は追加機器が不要な安上がりな方法である。業者に設計を丸投げ等すると、高価な電源制御装置の導入を求められる可能性が高い。十分業者等と仕様について検討すべきである。また、サーバ停止設定は機器導入前に検討しないと、後から設定変更を行う場合、いわゆる三すくみ状態のためサーバ停止が出来ない場合が多いので、事前に設計するのが望ましい。

### 2.1 PC の自動起動

現在販売されている PC やサーバはほとんどが WoL(Wakeup on LAN) を使う事にネットワーク経由でリモートで電源投入が可能である。

WoL は米国 AMD 社により開発された技術であり、同一 subnet 上の他の PC より Magic Packet と呼ばれる特殊なブロードキャストを送信する事により、PC を起動させる機能である。Magic Packet を送信するソフトウェアは多数フリーソフトで出回っているため入手は容易である。

しかしながら、WoL は未対応機器が存在する。2008 年 3 月に入札で導入した機器については、入札仕様書に WoL 対応を含める事を検討したが、一部メーカーのサーバはカタログで WoL 対応を謳ってなかったため仕様書への記載を見送った。入札の結果、残念ながら WoL 未対応機器が導入される事になったが、実際に試してみると WoL が動作した。カタログで謳ってなくても動作する可能性はある。

また WoL はセキュリティ上の問題で標準で無効にされている場合が多い<sup>7</sup>。そのため、設定変更によりより有効化させる必要がある。

一般に WoL が有効な構成の条件として、

- NIC のネットワークコントローラーが対応している<sup>8</sup>。
- PC が遠隔操作による起動に対応している<sup>9</sup>。増設 NIC の場合は、PC が増設ボードによる遠隔起動に対応している。
- OS が対応している。

の 3 つの条件がそろふ必要がある。OS 側での対応が必要であるため、OS によっては WoL が使えない場合がある<sup>10</sup>。OS が未対応の場合でもオープンソースの OS の場合は、ネットワークコントローラーのレジスタ (WoL レジスタ) を有効にするだけで良い場合が多いので、ネットワークコントローラーのデータシートを取り寄せてネットワークコントローラーのドライバを書き換えれば良い。残念ながらクローズドソースの OS の場合は対応出来ない。

<sup>7</sup>WoL は悪戯にも使える。電源をリモートで投入出来るのであるから、誰も居ない教室の端末を全て起動させるなどの悪戯が出来る。

<sup>8</sup>現時点では PCI/PCI-Express の増設ボードはほぼ対応。Cardbus 等は未対応

<sup>9</sup>WoL はネットワーク経由の起動方法だが、他に電話モデムを介してシリアル経由で起動させる方法もある。

<sup>10</sup>例を挙げると、Sun Micro Systems 社の Solaris は WoL 未対応である。

ソースは 2ch : <http://pc11.2ch.net/test/read.cgi/unix/1208669383/32-49>

Linux で WoL を有効にする方法は、OS 起動後、コマンド「/sbin/ethtool NIC-DEVICE wol g」を発行すれば良い。再起動すると無効になるので、/etc/rc.local 等を書いておけば良い。

— /etc/rc.local の設定例 —

```
if [ -f /sbin/ethtool ]; then
    /sbin/ethtool NIC-DEVICE wol g
fi
```

学科の端末室では、講義がある時間帯の 2 時間前にサーバに Magick Packet を送信し、起動させている。またサーバが正常起動したかを確認するシステムを SNMP(MRTG) で構築し、正常起動に失敗した場合は管理者に警告メールが行くように設定している。

## 2.2 Linux の poweroff

Linux の poweroff は poweroff コマンドを発行する事により可能である。第 1 章で述べた端末の poweroff も poweroff コマンドを発行して実現している。端末の場合は独自開発したソフトウェアで利用頻度およびタイマーを元に poweroff コマンドを発行したが、サーバの場合は OS のタイマー機能 (crontab) を使い実現した。具体的には下記のように午後 8:45 に poweroff コマンドを発行するよう設定した。(X: 日, Y: 月)

— crontab の設定例 —

```
$ crontab -l
45 20 * X Y /sbin/poweroff > /var/tmp/poweroff.log 2>&1
```

なお、古い PC では OS 側から電源を切ることができず、物理的なスイッチの Off が必要であったが、現在ではそのような PC はほとんど無い<sup>11</sup>。

## 2.3 講義時間外のサーバ数の縮小運用

常時解放している教室であっても、講義時間外の利用者は僅かである。同等の機能を提供しているサーバが複数台ある場合は、適切に設定を行えば、稼働しているサーバを減らすことができる。この運用はサーバのソフトウェアにも依存するので、入札仕様策定時に仕様を含めるのが望ましい。

知能情報工学科でも講義時間外の縮小運用が可能であるサーバソフトウェアが導入されているが、一部機能に三すくみが発生しているため、現時点では導入していない。次期システムでは入札仕様を含める事を検討している。

## 3 節電型の PC 導入

D 教室に設置している 2008 年度に導入したシステムでは節電型の PC を導入した。入札当時の Intel 製 CPU で最速は Intel Core2Duo E6000 シリーズであったが、あえて最速を選ばず最低速度の CPU の Intel Celeron 420 で導入した。Core2Duo E6000 シリーズは消費電力 (TDP)<sup>12</sup>が 65W であるが、Celeron 420 は 35W である。約 47%の省電力になっている<sup>13</sup>。

現在では最低速度の CPU であっても、十分高性能であり、高負荷のソフトウェアを使うなどの需要が無ければ高性能の CPU は無用の長物である。今回の導入から導入から半年経過したが、問題等は発生してない。なお、この点について

<sup>11</sup>正確にはわからないが、Windows95 ころのパソコンは OS 側より電源が切れないのが多かったが Windows98 ころのパソコンより電源を切れるようになった。具体的には AT 規格以前の PC はほとんどが電源を切ることができない。ATX 規格以後の PC は OS が対応していたら電源を切ることができる。

<sup>12</sup>TDP(Terminal Design Power), パソコンの熱設計の基準値。最大発熱量では無い。最大発熱量は Intel は公開していないようである。

<sup>13</sup>この値は CPU 単体の消費電力である。PC は電源装置 (AC-DC 変換機)/VGA/マザーボード/HDD/CD ドライブ/モニター等で構成されているため、これらの消費電力を含めた値が PC の消費電力になる。

は学科のカリキュラムにもよるので、学科の講義担当者と十分相談の上、低速 CPU の導入を判断してほしい。知能情報工学科では無用と判断した。

しかしながら、このような設計は評判が悪い。世の常として必ず教員から「予算内で最高速度を目指すべきである」という意見が出る。説得の材料として、「教室が無駄に暑くなり電力のデマンド値が上昇する。制限値をオーバーしても講義を停止するわけにはいかない。」「国家の基本方針である  $CO_2$  削減に反する」「カリキュラムを提示して高性能 CPU が不要であることを示す」などを用意しておけば良いだろう。

### 3.1 CPU の消費電力の歴史

CPU の消費電力は 1995 年に発表された Intel Pentium Pro 200Mhz では 35W 程度であった。その後、動作周波数/回路数の増加に従い消費電力は増大し、2004 年 11 月には 115W(Pentium4 Processore 570J/3.8GHz) の CPU まで登場してきた。この頃の Pentium 4 の開発コードネーム<sup>14</sup>は Prescott であるため、消費電力の多さを皮肉って「Prescott の悪夢」とまで言われた。この頃導入された Intel 製 CPU が PC 向の CPU では一番消費電力が高く、地球温暖化に貢献している

ブレイクスルーは 2006 年 7 月に訪れた。Intel Core2Duo シリーズの登場である。設計の改善により 65W まで低下した。

### 3.2 学内設置の主要 PC の消費電力

表 1 に学内の端末室に設置されている端末の消費電力を挙げる<sup>15</sup>。それぞれ内蔵メモリや内蔵 HDD 等の関係もあり、単純に消費電力を比較する事は出来ないため、一般的な話しと解釈して欲しい。

表 1: 学内の端末室に設置されている端末の消費電力

構成	CPU・コードネーム	TDP	起動時	平常時	負荷時 (増減値)	LCD
A:情報科学センターと同一機種 2004 年購入 MintPC RidottoS	Pentium 4/2.8GHz Northwood	68.4W	41 ~ 86W	42W	72W(30W 増)	24W
B:2006 年度導入端末 MintPC RidottoSP-A	Pentium 4 630/3.0GHz Prescott	84	56 ~ 115	56	100W(44W 増)	23
C:2008 年度導入端末 Dell Optiplex 745	Celeron 420/1.6GHz Conroe-L	32	48 ~ 60	47	56W(9W 増)	24
D: 構成 C と同一機種だが HW 構成違う	Core2Duo E6300/1.86GHz Allendale	65	71 ~ 97	60	68W(8W 増)	24

「平常時」は OS を起動させて何も作業をしてない状態である。「負荷時」は負荷として「 $\sin(10)/\cos(10)$  を連続計算させるプログラム」を一つ実行した状態である。

どの構成も無負荷の平常時は 50W 前後の低消費電力であるが、負荷時にかなりの差が出る。構成 A は負荷時に 30W 増加した。次は構成 B は悪評高い Prescott であるが負荷時に 44W 増加した。構成 C は節電を考えて最低速度の CPU を導入した端末で、負荷時に 9W 増加した。構成 D は構成 C とほぼ同一機種で CPU が異なる構成<sup>16</sup>で、負荷時に 8W 増加した。

<sup>14</sup>Intel 製の CPU は同一製品名であっても、内部実装等に細かい違いがあり、開発コードネームによって区別される。第一世代の Pentium4 は Willamette, 第二世代は Northwood, 第三世代は Prescott。

<sup>15</sup>短時間で 1 台のみ測定した値のため、参考程度にして欲しい。またクランプ型の電流計で測定したため、精度はあまり良くない。また力率測定は行っていないため、電力 (W) ではなく皮相電力 (VA) である。

<sup>16</sup>但し、CPU 以外の HW 構成も異なるため単純に比較はできない。

構成 C は節電を見込んで最低速度の Celeron 420 を導入したのだが、測定の結果は平常時は構成 A が最良になった。しかしながら負荷時は構成 C が最良であるという結果になった。構成 B の端末は現在知能情報工学科の主力端末であるが、平常時も負荷時も成績が悪い。Prescott の悪夢...

### 3.3 高度な省電力機能: EIST / AMD Cool'n'Quiet

次に参考データとして、高度な省電力機能を有する PC の消費電力を表 2 に挙げる。いずれの構成も学科端末のように簡素構成ではなく、高性能 VGA や HDD を複数増設しているため、消費電力が多い。参考程度にしてほしい。

本表に掲載した CPU は高度な省電力機能 (EIST: Enhanced Intel SpeedStep Technology / AMD Cool'n'Quiet) を有し、負荷に比例して動作周波数が増減する機能を有する<sup>17</sup>。一般に同一構成の CPU では動作周波数に比例して性能も増加するが、消費電力も増える。負荷が無い場合は低い周波数に下げることにより節電ができる。本機能はノートパソコン用 CPU では以前より搭載されていたが、近年デスクトップパソコン用 CPU にも搭載されるようになった。

表 2: 参考データ: 高度な省電力機能を有する PC の消費電力

注: TDP は Intel の CPU の基準のため AMD の CPU では未記載

構成	CPU・周波数可変範囲	TDP	起動時	平常時	負荷時 (増減値)
E:自作 PC	Intel Core2Duo E8500 2 ~ 3.16GHz / 2core 内蔵	65W	150W 以下	114W	1 負荷: 132W(18W 増) 2 負荷: 140(26W 増)
F:HP ML115G1	AMD Opteron 1216 1.0 ~ 2.4GHz / 2core 内蔵	不明	133 以下	74	1 負荷: 108(34W 増) 2 負荷: 127(43W 増)

表 2 に挙がっている CPU はいずれもマルチコア仕様のため負荷をかけるプログラムを一つ走らせた場合 (1 負荷) と二つ走らせた場合 (2 負荷) で測定した。高度な省電力機能により負荷により周波数が可変する。平常時は最低周波数で動作し、負荷が一つでもある場合は最高周波数で動作する。

一つ注意点がある。表にデータは挙げてないが、最高周波数に固定した状態で、負荷を与えた場合と無負荷の場合を比較すると、無負荷の場合は消費電力が低い。消費電力は周波数だけに比例するのではなく、負荷によっても比例する。

#### 3.3.1 高度な省電力機能の有効化

EIST 等の高度な省電力設定は標準設定では無効である。有効にするには、BIOS で有効にした後、OS の設定を変更する。比較的最近の OS でないとは対応してない。設定の概要は下記の通りである。詳細については紙面の都合で割愛する。

- Linux: Daemon 「cpuspeed」を有効にする。古い distribution では不可。
- WindowsXP: EIST 等のドライバを導入後、コントロールパネルの電源オプションの電源設定を変更する。
- Windows Vista: 電源オプションの詳細設定で可能 (Vista 搭載機が無いため未確認)。

#### 3.3.2 高度な省電力機能の応用

前述のように動作周波数が低いと省電力となるが、負荷が増加しても、周波数を増加させない設定を行うことで、高負荷時も電力を抑える事が可能である。最近の CPU は低い周波数でも十分高速であるので、低い周波数で固定して運用してもほとんどの演習では問題は無い。表 3/4 に前出の構成 E/F の PC を最低周波数に固定した場合の消費電力を掲載した。二つの負荷をかけた場合で比較すると 16W ~ 43W 程度の省電力になる。

本学では夏季に契約電力がオーバーしそうになると警告メールが發送されるが、講義等は停止することはできない。しかしながら本手法を応用すると、講義等で端末を停止することと無しに節電対策を行う事が出来る。

<sup>17</sup>Intel 製 CPU には EIST の他、C1E(Enhanced Halt State) と呼ばれる機能が提供されている。C1E も無負荷時には周波数を減少させる。但し EIST は OS 側からの制御が可能であるが、C1E は OS 側から制御は出来ない。

低周波数固定する方法は下記の通りである。

- Linux: Daemon 「cpuspeed」の設定を変更する。
- WindowsXP: 標準機能では不可能。フリーソフト「CrystalCPUID」で可能
- Windows Vista: 電源オプションで可能 (Vista 搭載機が無いため未確認)。

表 3: 構成 E の PC で、EIST を通常設定時・最低周波数固定時の消費電力の違い

EIST	周波数	起動時	平常時	1 負荷	2 負荷
通常設定	2 ~ 3.16GHz	150W 以下	114W	132W	140W
最低周波数固定	2GHz 固定	同上	同上	120W(12W 減)	124W(16W 減)

表 4: 構成 F の PC で、Cool'n'Quiet を通常設定時・最低周波数固定時の消費電力の違い

Cool'n'Quiet	周波数	起動時	平常時	1 負荷	2 負荷
通常設定	1 ~ 2.6GHz	133W 以下	74W	108W	127W
最低周波数固定	1GHz 固定	同上	73W	79W(29W 減)	84W(43W 減)

## 4 HDD の節電

HDD は常時内部でモーターが回転しているため、常にモーター分の電流を消費している。しかし OS などの設定で一定時間未使用の場合はモーターを停止する設定が可能である。

知能情報工学科や情報科学センターで導入している端末は HDD が無い diskless 構成であるため、本章で述べる方法は活用できない。しかし、サーバにはバックアップ等の目的のため特定の時間帯しか利用しない HDD が接続されている場合が多々ある。このような HDD には節電設定が有効である。

知能情報工学科では、学生端末の /home の自動バックアップ用サーバと、OS のセキュリティ更新のキャッシュサーバ (yum のキャッシュサーバ) で HDD の節電設定を行っている。どの程度節電に貢献しているかは表 5 を参照。

表 5: HDD の省電力設定

機能	CPU	TDP	HDD 台数	HDD 停止時	HDD 動作時
/home バックアップサーバ	Pentium4 3.2EGHz	89W	2 台 (250Gx2)	86W	93W(7W 増)
yum キャッシュサーバ	Celeron 1.4Ghz	34.8	6 台 (120G ~ 400G)	56W	93W(37W 増)
samba サーバ	Pentium D 2.8GHz	95	4 台 (160G,500Gx3)	100W	122W(22W 増)

なお、確かなソースがあるわけではないが、HDD のモーターの On/Off を頻繁にすると HDD の寿命が縮まるという意見を聞いたことがあるので、頻繁にアクセスがある HDD には省電力設定はお勧めしない。また HDD のモーターが止まった状態から動き出すまで少々待たされるので、サーバの反応が悪くなるためリアルタイム性が求められる場合はお勧めできない。

下記に Linux での設定例を挙げる。

HDD の節電の設定例

```
if [ -f /sbin/hdparm ]; then
    /sbin/hdparm -S180 /dev/sdX
fi
```

ここで「180」は 600 秒間 (5\*180) 未使用で停止を意味し、/dev/sdX は HDD のデバイスの指定である<sup>18</sup>。この設定

<sup>18</sup>詳細は linux のオンラインマニュアル参照 (man hdparm)



を/etc/rc.local 等に記述を行えばよい。なお、RAID カード等の特殊デバイスが導入されている場合は、上記設定は無視される場合がある。

## 5 最後に

ほとんどの学科の端末室で週の 2/3 以上が講義では使われていません。適切に縮小運用設定する事により、節電に十分貢献出来ます。

次に、使わないサーバや PC の電源はこまめに切りましょう。サーバ構築練習と称してほとんどアクセスが無いサーバを構築して、その後存在を忘れて電源を入れたまま放置する人が多いみたいですが、愚の骨頂です。大学だけでなく自宅サーバでも似たような事例を多数聞きます。このようなサーバは地球温暖化に貢献だけならいざ知らず、最悪クラッキングされて、犯罪者の片棒を担ぐ事になりかねません。

## 参考文献

- [1] ジョナサン コルベット 他, Linux デバイスドライバ 第 3 版, オライリージャパン, 2005, ISBN:4873112532.
- [2] Daniel P. Bovet 他, 詳解 Linux カーネル 第 3 版, オライリージャパン, 2007, ISBN:487311313X.