



1 概要

本ソフトウェアは Windows 系ノート PC のバッテリーの詳細情報を取得するソフトウェアです。

2 特徴

a) 情報取得関係

DeviceloControl API と GetSystemPowerStatus API を使用してバッテリーに関する詳細な情報を読み出して表示します。使用している API の名称と読み出す情報は以下のとおりです。

バッテリーを複数個装着している PC ではバッテリー個々に求まる値とまとめて1つとして求まる値があります。

- DeviceloControl - IOCTL_BATTERY_QUERY_TAG

BatteryTag:

- DeviceloControl - IOCTL_BATTERY_QUERY_INFORMATION - BatteryInformation(0)

Capabilities (Hex):

BATTERY_CAPACITY_RELATIVE:

BATTERY_IS_SHORT_TERM:

BATTERY_SET_CHARGE_SUPPORTED:

BATTERY_SET_DISCHARGE_SUPPORTED:

BATTERY_SYSTEM_BATTERY:

Technology:

Chemistry:

DesignedCapacity:

FullChargedCapacity:

DefaultAlert1:

DefaultAlert2:

CriticalBias:

CycleCount:

- DeviceloControl - IOCTL_BATTERY_QUERY_INFORMATION - BatteryGranularityInformation(1)

Entry:

Granularity(0-3):

Capacity(0-3):

- DeviceloControl - IOCTL_BATTERY_QUERY_INFORMATION - BatteryTemperature(2)

BatteryTemperature:

- DeviceIoControl - IOCTL_BATTERY_QUERY_INFORMATION - BatteryEstimatedTime (3)
BatteryEstimatedTime:
- DeviceIoControl - IOCTL_BATTERY_QUERY_INFORMATION - BatteryDeviceName (4)
BatteryDeviceName:
- DeviceIoControl - IOCTL_BATTERY_QUERY_INFORMATION - BatteryManufactureDate (5)
BatteryManufactureDate:
- DeviceIoControl - IOCTL_BATTERY_QUERY_INFORMATION - BatteryManufactureName (6)
BatteryManufactureName:
- DeviceIoControl - IOCTL_BATTERY_QUERY_INFORMATION - BatteryUniqueID (7)
BatteryUniqueID:
- DeviceIoControl - IOCTL_BATTERY_QUERY_INFORMATION - BatterySerialNumber (8)
BatterySerialNumber:
- DeviceIoControl - IOCTL_BATTERY_QUERY_STATUS
PowerState (Hex):
BATTERY_CHARGING:
BATTERY_CRITICAL:
BATTERY_DISCHARGING:
BATTERY_POWER_ON_LINE:
Capacity:
Voltage:
Rate:
- BatteryChargeCompletionTime
FullChargeCapacity, Capacity, および Rate から充電が完了するまでの時間を計算して表示します。ただしかなり精度は悪いです。
ここまでの値は個々のバッテリーごとに求められます。
- GetSystemPowerStatus
ACLineStatus:
BatteryFlag:
High
Normal

Low
Critical
Charging
No system battery
BatteryLifePercent:
BatteryLifeTime:
BatteryFullLifeTime:

この値は複数個のバッテリーに対して1つだけ求められます。

b) 表示保存関係

読み出した情報はフォームに表示します。ファイルに保存もできます。言語は英語/日本語が選べます。

c) 自動更新機能

バッテリー残量など時間とともに値が変化する項目は[更新]または[Refresh]を押すことで内容を更新します。また1秒ごとに自動更新も可能です。

d) その他

複数個のバッテリーが存在する場合、個々のバッテリーの情報を読み出すことができます。

Delphi7のソースが付属します。フリーのコンパイラ Turbo Delphi 2006 Explorer で再コンパイル可能です。

3 動作環境

Windows2000/XPで動作確認してあります。Windows Vistaでも動作すると思いますが動作確認を行っていません。Windows9x系では動作しないと思います。

4 インストール

DISK1フォルダのsetup.exeを実行してください。

もし古いバージョンの kmBattInfo がインストールされている場合は古いバージョンの kmBattInfo を終了しアンインストールしてから、新しいバージョンの kmBattInfo をインストールする必要があります。

5 アンインストール

プログラムの追加と削除またはアプリケーションの追加と削除からアンインストールしてください。

6 手動インストール/アンインストールについて

本ソフトは setup.exe を実行することで自動的にインストールができ、プログラムの追加と削除またはアプリケーションの追加と削除からアンインストールができますが、どうしても手動でインストールしたい場合は適当なフォルダに sources の kmBattInfo.exe をコピーして実行ができます。それ以外のファイルは必要ありません。またレジストリは使用していません。

7 作業フォルダとファイル

kmBattInfo.exe が存在するフォルダに kmBattInfo.ini ファイルが作成されます。ini ファイルはもし存在しない場合デフォルトの値で作成されます。

8 操作方法

8.1 起動

起動するとバッテリーの存在と個数を調べます。バッテリーが複数個存在する場合は、コンボ・ボックスでバッテリーを選択できます。

[ファイル]→[保存]で情報をテキスト・ファイルに保存できます。

8.2 情報の説明

情報は Windows の API を使用して読み出しています。API の仕様に関しては Microsoft の msdn のホームページ(英文)を参照してください。本プログラムで使用している API は以下のとおりです。

- SetupDiGetClassDevs

<http://msdn2.microsoft.com/en-us/library/ms792959.aspx>

バッテリーに関する情報セットのハンドルを取得します

- SetupDiEnumDeviceInterfaces

<http://msdn2.microsoft.com/en-us/library/ms791242.aspx>

バッテリーに関するデバイス情報セットのデバイス・インタフェースを表すコンテキスト構造体

を求めます。

- SetupDiGetDeviceInterfaceDetail

<http://msdn2.microsoft.com/en-us/library/ms792901.aspx>

バッテリーのデバイス・インタフェースに関する詳細情報を求めます。

- DeviceIoControl

<http://msdn2.microsoft.com/en-us/library/aa363216.aspx>

DeviceIoControl はパラメタによって機能が決まります。

- IOCTL_BATTERY_QUERY_TAG

<http://msdn2.microsoft.com/en-us/library/aa372700.aspx>

DeviceIoControl で IOCTL_BATTERY_QUERY_TAG を使用すると BatteryTag が求められます。

- IOCTL_BATTERY_QUERY_INFORMATION

[http://msdn2.microsoft.com/en-us/library/aa372698\(VS.85\).aspx](http://msdn2.microsoft.com/en-us/library/aa372698(VS.85).aspx)

IOCTL_BATTERY_QUERY_INFORMATION はさらに InformationLevel で機能が決まります。

- IOCTL_BATTERY_QUERY_STATUS

<http://msdn2.microsoft.com/en-us/library/aa372699.aspx>

バッテリーの状態を求めます。これは個々のバッテリーの情報を求めます。

- GetSystemPowerStatus

バッテリーの状態を求めます。個々のバッテリーの情報ではなくシステム全体として機能するひとつにまとめたバッテリーの状態を求めます。

8.2.1 BatteryTag

バッテリー・タグです。

msdn の解説ではバッテリーの充電完了、フル充電の容量が変わったとき、バッテリーが交換されたときにこの値が変わると記述されています。このタグの値が変わると各種情報を読み出しなおす必要があります。

kmBattInfo ではまずバッテリー・タグを読み出し、引き続きそのバッテリー・タグを使用して各種情報を読み出していますのでこのバッテリー・タグの値にかかわらず正しく情報を読み出していると思います。

8.2.2 Capabilities

バッテリーの能力です。下記の5ビットが定義されています。

- BATTERY_CAPACITY_RELATIVE

このビットがオンの場合、他の API で読み出すすべての容量に関する情報は相対値になります。何人かに kmBattInfo をためしていただいた結果、実験した PC ではこのビットはオフでした（絶対値）。kmBattInfo はこのビットのオン/オフにかかわらずすべて絶対値と見なして値を表示しています。

- BATTERY_IS_SHORT_TERM

このビットがオンの場合、その PC はバッテリーで駆動することを意図されていません。UPS が付加されたサーバのように、バッテリーは存在するがその駆動時間は短時間であることを示します。

- BATTERY_SET_CHARGE_SUPPORTED

意味がよくわかりません。ためしたノート PC でオンのものはありませんでした。

- BATTERY_SET_DISCHARGE_SUPPORTED

意味がよくわかりません。ためしたノート PC でオンのものはありませんでした。

- BATTERY_SYSTEM_BATTERY

ノート PC のようにバッテリーで駆動される PC のときにオンになるようです。ためしたノート PC はすべてこのビットがオンでした。

8.2.3 Technology

バッテリーが一次電池（充電不可）か二次電池（充電可能）かを示します。

8.2.4 Chemistry

バッテリーの種類を表す最大 4 文字の略号です。msdn には以下の例がありました。

PbAc: 鉛バッテリー

LiION, Li-I: リチウム・イオン・バッテリー

NiCd: ニッケル・カドミウム・バッテリー

NiMH: ニッケル水素・バッテリー

NiZn: ニッケル亜鉛バッテリー

RAM: アルカリ・マンガン二次電池

8.2.5 DesignedCapacity

設計容量。新品のバッテリーの容量です。

8.2.6 FullChargedCapacity

フル充電容量。

劣化してバッテリーの容量が減るとこの値が少なくなる PC もありました。バッテリーが劣化していてもこの値が DesignedCapacity と変化しない PC もありました。

8.2.7 DefaultAlert1

クリティカル・アラームになる容量です。

8.2.8 DefaultAlert2

バッテリー・ロウ・アラームになる容量です。

8.2.9 CriticalBias

バッテリー残量パーセントが 0 になったときにどのくらいの容量が残っているかが読み出せます。これは見かけの容量が 0 になってもまだバッテリーがすこし残っているという使い方をするための値です。実験したすべての PC では 0 でした。

8.2.10 CycleCount

何回充放電を行ったかを示します。実験したすべての PC では 0 でした。0 は非サポートを示します。

8.2.11 BatteryGranularityInformation

容量情報の粒度です。耳慣れない言葉ですが、読み出した Capacity がこの粒度の値の細かさで読み出せる (分解能である) ことを示すらしいです。粒度情報は Granularity と Capacity の組です。最大 4 組です。

- Entry

粒度の情報が何組あるかを示します。

- Granularity(0-3), Capacity(0-3)

0~Capacity(0)が Granularity(0), Capacity(0)~Capacity(1)が Granularity(1)らしいです。

例えば Capacity(0)が 1000[mWh], Granularity(0)が 200[mWh]とすると 0~1000[mWh]は 0, 200, 400, 600, 800, 1000 程度の分解能であることを示すらしいです。

8.2.12 BatteryTemperature

バッテリーの温度です。読み出した値はケルビンなので°Cに変換して表示しています。実験した PC でこの値を返すものはありませんでした。ダミーのデータで検証しています。

8.2.13 BatteryEstimatedTime

バッテリーがあとどのくらいの時間持つかを示します。hh:mm:ss に変換して表示しています。

これはバッテリーの残り時間ですから放電中でないと意味を持たないのですが、実験した PC の中で DELL LATITUDE D600 は充電中にとんでもない値(例:57420:00:00)を返すことがありました。これは PC 側の問題で本ソフトの不具合ではありません。

8.2.14 BatteryDeviceName

バッテリーの型番を表示します。

8.2.15 BatteryManufactureDate

バッテリーの製造日付を表示するようですが、実験した PC でこの値を返すものはありませんでした。

8.2.16 BatteryManufactureName

バッテリーの製造メーカーを表示します。

8.2.17 BatteryUniqueID

バッテリーのユニーク ID 表示します。

8.2.18 BatterySerialNumber

バッテリーのシリアル・ナンバを表示するようですが、実験した PC でこの値を返すものはありません

せんでした。

8.2.19 PowerState

バッテリーの状態です。下記の4ビットが定義されています。

- BATTERY_CHARGING

バッテリーが充電中のときにオンです

- BATTERY_CRITICAL

バッテリーがクリティカルなときにオンです。

- BATTERY_DISCHARGING

バッテリーが放電中のときにオンです

- BATTERY_POWER_ON_LINE

バッテリーが接続されているときにオンです

8.2.20 Capacity

現在のバッテリー容量を示します。充電中ならこの値が増えていきます。放電中からこの値が減っていきます。

8.2.21 Voltage

バッテリーの端子電圧を示します。

8.2.22 Rate

バッテリーの充放電率を示します。充電中なら正の値、放電中なら負の値になります。

実験したPCの中でDELL LATITUDE D600は充電中にとんでもない値(例:-1)を返すことがありました。これはPC側の問題で本ソフトの不具合ではありません。

8.2.23 BatteryChargeCompletionTime

バッテリーが充電完了になるまでの時間です。

この値は API から読み出した値ではありません。

FullChargeCapacity, Capacity, および Rate の値から充電が完了するまでの時間を計算して表示しています。かなり精度は悪いです。

8.2.24 ACLineStatus

バッテリー駆動か AC 電源駆動かを示します。

8.2.25 BatteryFlag

バッテリーの状態を示します。以下の 4 ビットと 2 つの値が定義されています。このどれにも当てはまらない場合、Normal または正常と表示しています。

- High

バッテリーが十分あるときにオンです。

- Low

バッテリーがロウでオンです。

- Critical

バッテリーがクリティカルなときにオンです。

- Charging

バッテリーが充電中にオンです。

- No system battery

バッテリーが存在しないときにこの値になります。

- Unknown

不明なときにこの値になります。

8.2.26 BatteryLifePercent

バッテリーの残量を示します。

8.2.27 BatteryLifeTime

バッテリーの残り時間を示します。

8.2.28 BatteryFullLifeTime

バッテリーの駆動可能時間を示すらしいですがよくわかりません。実験した PC でこの値を正しく返す PC はありませんでした。

9 謝辞

インターネット上で公開されているみなさまのいろいろな情報を参考にさせていただきました。御礼申し上げます。

私も何か貢献したいと思い、ソースを公開させていただきます。

10 バグ報告, 機能追加の提案

ベクターのダウンロードのページにある『コメント・評価』に記入していただければ見るかも知れません。

メールはだいたい読んでいますが 99%が spam なので見逃すかも知れません。

常識的なメールで返事を期待するものにはなるべく返事をするようにはしていますが、返事ができないかもしれません。見落としている場合もありますので催促していただいてもけっこうです。

11 ウイルス・チェック

以下のウイルス対策ソフトでチェック済みです。

- ・ McAfee virusscan バージョン 10.0
- ・ ビルド 10.0.27
- ・ エンジンのバージョン 5200
- ・ DAT のバージョン 5235
- ・ DAT ファイルの作成日 2008-02-21

12 著作権

著作権はキートン増田@普通のサラリーマンが持ちます。

ソースが添付されていますので参考にするのは自由です。

参考にするのでなく、まったくコピーする場合は『キートン増田のソースをコピーした』とわかるように書いていただければご自由にお使いください。

13 雑誌などへの収録

インターネットで取り上げていただける場合には連絡していただければうれしいです。

雑誌に収録していただいた場合には見本誌を1部いただけるととってもうれしいです。

見本誌をいただけない場合には連絡していただけるとうれしいです(自費で買いに行きます-笑)。

14 その他

かなり急いで作ったのでソースが汚いです。同じような処理があったりします。笑って許してやってください。

15 修正履歴

0.5.0.0

初版