

УДК 004.414.22/004.051/519.687.5

И. Б. ТУРКИН, А. В. ВДОВИТЧЕНКО, С. А. САМИР

Национальный аэрокосмический университет им. Н. Е. Жуковского «ХАИ», Украина

ЭНЕРГОСБЕРЕГАЮЩИЕ ФУНКЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ МОБИЛЬНЫХ УСТРОЙСТВ

Приведен обзор современного состояния и анализ перспектив развития, средств оптимизации энергопотребления мобильных устройств. Рассматриваются варианты оптимизации энергопотребления мобильных телефонов. Основное внимание уделено реальному состоянию проблем экологии, значимую роль в которых занимают информационные компьютерные технологии. Подробно рассмотрена роль мобильных устройств в экологии, так же влияние энергопотребления мобильных компьютеров на длительность службы батареи. Рассмотрены подходы к оптимизации питания на уровне прикладных программ и менеджеров управления ресурсами мобильных устройств. Приведены примеры существующих отечественных и зарубежных теорий и их краткие характеристики, приведена библиография по тематике.

Ключевые слова: экология, информационные компьютерные технологии, экологическое программное обеспечение, мобильные устройства, оптимизация, power profiling, обзор.

Введение

Современные информационные технологии не всегда энергетически эффективны. Выбросы, вызванные работой вычислительной техники, превышают выбросы всего авиационного транспорта, а недавно было подсчитано, что среднестатистический сеанс поиска по запросу к поисковой машине Google приводит к выбросу в атмосферу 7 грамм углекислого газа, что соответствует половине эмиссии CO₂ от кипячения чашки воды в «небольшом электрическом» чайнике. Для сравнения: по нормам Евросоюза выбросы углекислого газа у автомобилей не должны превышать 140 грамм на один километр пути.

Интерес у авторов настоящей статьи вызывает совсем не энергетическая эффективность (или неэффективность). Интерес вызывает существующая зависимость такой материальной, физически измеряемой категории, как «энергопотребление» от нематериальных, неизмеряемых в традиционных физических величинах понятий: «байт», «бит», «количество операций в секунду» и т.д.

Для пользователей мобильных коммуникационных устройств (смартфонов, лэптопов, планшетов) этот абстрактный интерес приобретает весьма прагматичный характер. Многие из современных мобильных устройств не уступают по мощности стационарным ПК и одновременно с этим обладают такими свойствами, как портативность и переносимость [1]. Но эти плюсы влекут за собой ряд довольно значительных проблем: проблема с выводом излишков тепла, образуемого при работе таких компонентов как процессор, видеокарта и др., проблема малого срока работы устройства без подключения к

сети электропитания. Обе эти проблемы имеют два пути решения, которые не являются взаимоисключающими – это применение новых технологий для создания аппаратных компонент (процессора, видеокарты, аккумулятора и др.) и внедрение новых подходов к разработке программного обеспечения (ПО) для данных компонентов [2].

1. Анализ методов снижения энергопотребления мобильных устройств

Рассмотрим подробнее проблему быстрого разряда аккумулятора при работе без подключения к сети электропитания. Помимо неудобств в использовании, быстрый разряд несет за собой одну проблему – утилизацию элементов питания после окончания сроков их эксплуатации. Уменьшение суммарного энергопотребления мобильных устройств приводит к увеличению времени разряда батареи мобильного устройства, а следовательно, продлевает срок жизни аккумулятора и снижает общие затраты на утилизацию элементов питания.

Данный подход следует таким принципам [3]:

- считать быстро;
- использовать ресурсы устройства настолько, насколько нужно (нет смысла проводить расчеты в одном потоке, если устройство обладает, к примеру, 4-х ядерным процессором);
- действовать в рамках контекста (работать в различных режимах в зависимости от того, что сейчас является источником питания: сеть или аккумулятор).

На текущий момент большинство разработчиков ПО не уделяют внимание оптимизации энерго-

потребления. К таким выводам можно прийти, воспользовавшись любым стандартным анализатором под любой платформой. Для оценки времени работы инженеры применяют анализ разряда батареи, включающий измерение параметров питаемого устройства, его программного и аппаратного обеспечения и отдельных узлов. Методы анализа включают измерение разрядной характеристики и изучение влияния разряда на разные режимы работы и используемые профили. По результатам анализа инженер может выбрать компромиссную схему управления питанием, максимально продлевающую срок службы батареи [4].

Технологии контроля энергопотребления и рабочих характеристик SmartReflex™ компании Texas Instruments (TI) в настоящий момент значительно усовершенствованы с появлением второго поколения. Технологии SmartReflex™ 2, обладают проверенными на практике интеллектуальными адаптивными возможностями, позволяющими эффективно решить эти задачи и создать пути для будущих решений:

- динамическая подстройка рабочих характеристик транзистора в зависимости от утечки;
- динамическое снижение напряжений блоков памяти, находящихся в режиме ожидания;
- автоматизация применения технологий SmartReflex™ в процессе проектирования [5].

Известно много методов управления энергопотреблением, например, выбор схемы управления питанием:

- сбалансированная. Обеспечивает максимальное эффективное использование батареи, когда это необходимо и экономит энергию, когда устройство находится в состоянии простоя;
- энергосберегающая. Является лучшим способом для максимально долгого использования батареи без перезарядки за счет некоторого снижения производительностью;
- с высокой производительностью. Увеличивает производительность системы за счет снижения времени работы батареи [6].

Оптимизация энергопотребления за счет распараллеливания задачи. В современных устройствах все чаще используются многоядерные процессоры. Использование всех доступных ядер способствует не только повышению производительности, но и экономии электроэнергии. Если добиться одновременного использования наибольшего числа потоков или ядер, то программа будет работать быстрее, расходуя меньше электроэнергии. На приведенном ниже графике (рис. 3) показано потребление электричества тремя разными приложениями при тестировании на процессоре Intel® Core™ второго поколения в операционной системе Windows® 7 [7].

Оптимизация энергопотребления за счет изменения количества системных вызовов. В многопоточном приложении может возникнуть конфликт между потоками, взаимодействующими с ядром системы. Во избежание этих проблем нужно использовать вызов API Windows «EnterCriticalSection()» для синхронизации передачи данных между потоками в пространстве пользователя, а не вызов API Windows «WaitForSingleObject()», выполняющийся в пространстве ядра. Экономия потребления электроэнергии 4-х потоковым приложением при снижении числа блокировок из-за конфликтов может достигнуть 60 % (по результатам тестирования на процессоре Intel Core™ 2-го поколения на платформе Windows 7 [8]).

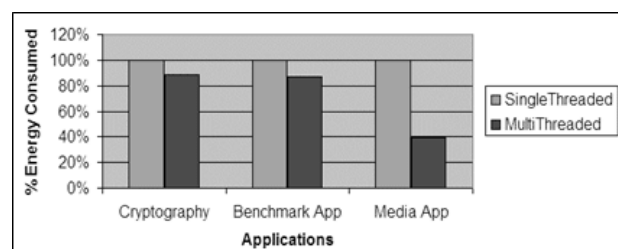


Рис. 3. Экономия электроэнергии трех различных приложений в однопоточном и в многопоточном режиме

Оптимизация энергопотребления за счет объединения операций работы с ресурсами. Если объединить вызовы ресурсов в пакеты, приложение будет потреблять меньше энергии (рис. 4).

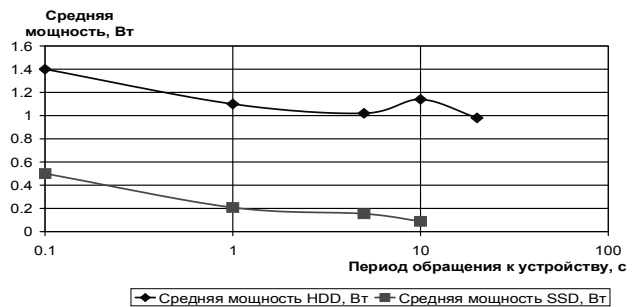


Рис. 4. Зависимость средней мощности энергопотребления от длительности периодов обращений к устройствам

Оптимизация энергопотребления за счет минимизации использования оперативной памяти достигается за счет:

- исключения ненужных преобразований формата графики (из YUV в RGB и обратно) для минимизации числа обращений к графическому и центральному процессорам;
- кэширования часто используемых структур данных;
- ограничения количество перемещений данных между пространством ядра и пространством

пользователя [9].

Оптимизация энергопотребления за счет адаптации под текущий источник питания позволяет продлить срок работы устройств от аккумуляторов, если приложения будут менять режим своей работы в зависимости от используемого в данный момент источника электроэнергии (сеть или аккумулятор) и от уровня оставшегося заряда аккумулятора [10].

2. Инструментальные средства энергетического профилирования приложений

Энергопотребление платформы можно структурировать, разделив его на две части:

- статическая часть включает в себя затраты на поддержание платформы в рабочем состоянии (например, энергия, потребляемая системой охлаждения, блоком питания и т. д.). Она имеет слабую корреляцию с природой исполняемых приложений, поэтому представляет интерес в последнюю очередь из-за невозможности контроля;

- динамическая часть связана с исполнением приложений. Поскольку эта часть сильно зависит от характеристик загрузки, степени утилизации ресурсов и политики энергопотребления, она может сильно варьироваться. Именно в этой области появляется простор для оптимизации, так как ПО вносит серьезный вклад в общее энергопотребление системы [11, 12].

Спецификация стандарта ACPI [1] регламентирует интерфейс регулирования энергопотребления операционной системой. Он объединяет ряд методик по снижению энергозатрат, опирающихся на достаточно очевидную идею: для устройства должен быть реализован набор состояний с пониженными запросами по энергии, но с урезанной в той или иной степени функциональностью. Переключение между состояниями не происходит мгновенно [9, 13].

Для CPU есть два основных набора состояний:

- P-states. Это состояния активной работы. P-state – пара частоты процессора и напряжения. Для каждого ядра P-state может быть выбран индивидуально, но при этом нужно понимать, что в рамках одного package могут существовать аппаратные ограничения на одновременные изменения состояний разными ядрами. Так, например, напряжение в процессоре Pentium 4 не может опуститься ниже, чем самое высокое среди запрошенных [14]. Для управления этими состояниями используются Model Specific Registers IA32_PERF_CTL/IA32_PERF_STATUS;

- C-States. В эти состояния процессор входит при выполнении ассемблерных инструкций HLT/MWAIT (команды остановки активного исполнения). Каждому из них соответствуют различные

уровни сна (оставляем ли напряжение на кэшах, держим ли напряжение на буферах), из более низких дольше просыпаться (от 10 до 100 микросекунд [9]). По этой причине ядро спускается в “состояние глубокого сна” не сразу, а постепенно. Так что излишне частые пробуждения процессора в связи с активностью пользовательского приложения указывают на неэффективность этого приложения с точки зрения энергопотребления.

Процессоры Intel содержат в себе набор эвристических алгоритмов управления энергопотреблением, и у разработчика есть возможность повлиять на их работу записью в MSR IA32_ENERGY_PERF_BIAS. Предполагается существование 16 уровней производительности (0 – максимальная производительность, 15 – максимальное энергопотребление/минимальная производительность).

Процессоры Pentium содержали в себе модуль измерения производительности (Performance Monitoring Unit), который давал возможность собирать статистику по количеству аппаратных событий, происходящих в процессоре. На базе этой статистики долгое время и производились оценки энергопотребления. Происходило это следующим образом: строилось некоторое предположение о корреляции между измеряемыми метриками и энергопотреблением системы (энергетическая модель), и далее, когда профайлер с заложенной моделью попадал на целевую машину, запускался процесс калибровки, связывающий оригинальные метрики и реальное энергопотребление. Далее по полученным коэффициентам можно оценивать энергопотребление.

Такой подход не может гарантировать безошибочный результат, и поэтому в архитектуру Sandy Bridge был заложен RAPL-интерфейс (Running Average Power Limit). Появилась возможность получить информацию о количестве энергии, потребленной системой с момента запуска. Профилировщики, работающие на базе счетчиков производительности, совсем не потеряли значения, поскольку встретить машину, не оснащенную процессором на архитектуре Sandy Bridge и старше, все еще очень несложно, и RAPL позволяет оценивать только энергопотребление CPU, чего в некоторых случаях может быть недостаточно.

Рассмотрим доступные разработчику инструменты, использующие метрики производительности, на базе которых строится оценка затраченной энергии.

PowerTop[2] (Linux) – утилита, предоставляющая информацию по энергопотреблению процессов системы в режиме реального времени. В своих выводах опирается на энергетическую модель, учитывающую потребление CPU, Network-devices, GPU и

HDD [4]. Среди безусловных плюсов стоит отметить то, что эта утилита сможет работать на большинстве существующих CPU от Intel (начиная с Core 2 Duo). Помимо того, инструмент предоставляет простой доступ к ряду параметров управления энергопотреблением системы, а также показывает, какие компоненты наиболее проблемные и приводит конкретные предложения по настройке с целью уменьшения потребления энергии. Примерами таких рекомендаций могут служить – включение энергосбережения Wi-Fi, перевод AC97 в режим энергосбережения, выключение звука, включение режима энергосбережения PCI Express и SATA-устройств, выключение опроса CDROM с помощью HAL, включение экономии энергии в настройках звукового чипа HDA и т.д. Также PowerTop может посоветовать изменения некоторых конфигураций ядра, например – `dirty_ratio`, `dirty_background_ratio`, `sched_mc_power_savings` и т.п.

Joulemeter [6] (Windows) – профилировщик, использующий, как и PowerTop, метрики производительности для оценки затраченной энергии [8]. Учитывает затраты CPU, HDD, GPU, сетевых устройств и экрана. Так же, как и PowerTop, не дает возможности связывать статистику с исходным кодом приложения, хотя и является удобным инструментом для анализа энергопотребления системы в целом. Так же как и PowerTop, Joulemeter полагается на статистику от PMU

Рассмотрим утилиты, которые помимо метрик производительности используют RAPL-интерфейс.

Intel VTune Amplifier 2013 Update 13 [1] (Linux, Windows) – многофункциональный профилировщик производительности, позволяющий также измерять энергопотребление. Может связывать собранную статистику с исходным кодом, собирать стек вызовов и визуализировать полученную информацию. В режиме Advanced Hotspots + collection stack + estimate call counts также дает информацию по количеству потребленной энергии (метрики: Energy Core, Energy GFX, Energy Package). Позволяет получить статистику с точностью до инструкции, что дает очень широкие возможности по анализу для дальнейшей оптимизации.

Activity Monitor [10] (OS X) – монитор производительности, выдающий помимо прочего метрику, характеризующую энергоэффективность работающих приложений. Не дает очень детальной статистики, но позволяет оценить, есть ли смысл подключать более информативные инструменты для получения развернутой статистики.

XCode 5 [16] (OS X). В состав IDE от Apple был включен мощный инструмент по анализу энергопотребления приложения. Он предоставляет агрегированные метрики, которые могут оказаться даже

более информативными, чем абсолютные значения потребления энергии. Среди них можно выделить WakesPerSecond – количество “пробуждений” CPU в связи с активностью приложения. В дополнение к этой метрике данный инструмент выстраивает CPU-usage трассу, на которой выделен CPU Wake Overhead. Такие трассы строятся отдельно для каждого из потоков, что упрощает анализ приложения. В отличие от Intel VTune Amplifier, инструмент от Apple не связывает напрямую измеряемые метрики и код приложения.

Таким образом, мы видим, что для задачи анализа энергопотребления на любой ОС найдется ряд инструментов, предоставляющих различную степень детализации.

Заключение

Рассмотрены текущие проблемы экологии и их дальнейшее развитие. Определена роль компьютерных информационных технологий в проблемах экологии. Рассмотрены мобильные устройства, как часть ИКТ, которая требует особого внимания. Выявлена одна из основных проблем мобильных устройств – это энергопотребление. Проведен анализ методов управления энергопотреблением, включая оптимизацию энергопотребления. Выявлено что большинство разработчиков программного обеспечения не уделяют внимание оптимизации энергопотребления.

В дальнейшем работа может быть развита во многих направлениях, например:

- разработка собственных инструментов оптимизации энергопотребления ПО;
- оценка существующих программных инструментов оптимизации работы ПО, для снижения энергопотребления;
- разработка методологий разработки экологического программного обеспечения.

Литература

1. *Энергосберегающие технологии [Электронный ресурс] / А. Спиридонов. – Режим доступа: <http://www.mobile-review.com/articles/2006/smart-energy.shtml> – 13.12.2013.*
2. *Методы снижения потребления энергии современными портативными устройствами. [Текст] / Е. Бирюков, Д. Василенко. – Режим доступа: http://www.compitech.ru/html.cgi/archiv/05_06/st-at_198.htm – 26.01.2014.*
3. *Производительность и энергопотребление мобильных приложений на примере Windows Phone 7 [Текст] / В. Колесников. – Режим доступа: <http://lib.custis.ru/1b5-performance-and-energy-of-mobile-kolesnikov> – 12.01.2014.*
4. *Брорейн, Э. Анализ разряда батареи для оценки времени работы мобильных устройств [Текст] / Э. Брорейн, Б. Золло // Электронные компоненты. – 2011. – №4. – С. 10 – 11.*

5. Технологии контроля энергопотребления и рабочих характеристик SmartReflex™: снижение потребляемой мощности и оптимизация производительности [Электронный ресурс]. – Режим доступа: http://scanti.ru/journal/ti/components/test/2008/2008-4%2820%29_35.htm. – 29.01.2014.

6. Экономия заряда батареи [Электронный ресурс] / Microsoft inc. – Режим доступа: <http://windows.microsoft.com/ru-ru/windows/conserving-battery-power#1TC=windows-7>. – 23.02.2014.

7. Мэддисон, Э. Контуры мировой экономики в 1-2030 гг. Очерки по макроэкономической истории [Текст] / пер. с англ. Ю. Каптуревского; под ред. О. Филатовой. – М. : Изд. Института Гайдара, 2012. – 584 с.

8. Energy-Efficient Software Guidelines [Электронный ресурс] / Intel inc. – Режим доступа: <http://software.intel.com/ru-ru/articles/partner-energy-efficient-software-guidelines>. – 03.02.2014.

9. Снижение энергопотребления приложений [Электронный ресурс] / Intel inc. – Режим доступа: <http://software.intel.com/ru-ru/articles/conserving-active-power>. – 01.02.2014.

10. Повышение эффективности энергопотребления для приложений [Электронный ресурс] / S. Sinofsky. – Режим доступа: http://blogs.msdn.com/b/b8_ru/archive/2012/02/15/improving-power-efficiency.aspx – 30.01.2014.

11. Green IT и энергоэффективность [Элек-

тронный ресурс] / EcoRussia. – Режим доступа: <http://www.ecorussia.info/ru/ecopedia/it-future>. – 16.12.2013.

12. Abenius, S. Green IT & Green Software - Time and Energy Savings Using Existing Tools [Text] / S. Abenius // Environmental Informatics and Industrial Environmental Protection: Concepts, Methods and Tools – Shaker Verlag 2009. – P. 57-66.

13. Экономия заряда батареи [Электронный ресурс] / Microsoft inc. – Режим доступа: <http://windows.microsoft.com/ru-ru/windows/conserving-battery-power#1TC=windows-7>. – 23.02.2014.

14. Повышение эффективности энергопотребления для приложений [Текст] / Ш. Фараг, Б. Сроуп. – Режим доступа: http://blogs.msdn.com/b/b8_ru/archive/2012/02/15/improving-power-efficiency.aspx. – 10.01.2014.

15. Снижение энергопотребления приложений [Электронный ресурс] / Intel inc. – Режим доступа: <http://software.intel.com/ru-ru/articles/conserving-active-power>. – 01.02.2014.

16. Instruments User Reference [Электронный ресурс] / Apple inc. – Режим доступа: https://developer.apple.com/library/ios/documentation/AnalysisTools/Reference/Instruments_User_Reference/EnergyUsageInstrument/EnergyUsageInstrument.html. – 03.01.2014.

Поступила в редакцию 24.02.2014, рассмотрена на редколлегии 25.03.2014

Рецензент: д-р техн. наук, проф., зав. каф. экономики и маркетинга В. М. Вартамян, Национальный аэрокосмический университет им. Н. Е. Жуковского «ХАИ», Харьков, Украина.

ЕНЕРГОЗБЕРІГАЮЧІ ФУНКЦІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ МОБІЛЬНИХ ПРИСТРОЇВ

І. Б. Туркін, О. В. Вдовитченко, С. А. Самір

Наведено огляд сучасного стану та аналіз перспектив розвитку, засобів оптимізації енергоспоживання мобільних пристроїв. Розглядаються варіанти оптимізації енергоспоживання мобільних телефонів. Основна увага приділена реальному стану проблем екології, значиму роль в яких займають інформаційні комп'ютерні технології. Докладно розглянута роль мобільних пристроїв в екології, також вплив енергоспоживання мобільних комп'ютерів на тривалість служби батареї. Розглянуто підходи до оптимізації харчування на рівні прикладних програм та менеджерів управління ресурсами мобільних пристроїв. Наведені приклади існуючих вітчизняних і зарубіжних теорій і їх короткі характеристики, наведена бібліографія з тематики.

Ключові слова: екологія, інформаційні комп'ютерні технології, екологічне програмне забезпечення, мобільні пристрої, оптимізація, power profiling, огляд.

ENERGY SAVING FUNCTIONS OF THE SOFTWARE OF MOBILE DEVICES

I. B. Turkin, A. V. Vdovitchenko, S. A. Samir

Provides an overview of the modern state and prospects of analysis's development, ways to optimize energy consumption of mobile devices. Discusses options for optimizing energy consumption of mobile phones. Focuses on the real problems of the environment, in which information and computer technology do a significant role. Discussed in detail the role of mobile devices in the environment, also the influence of energy consumption on mobile computers duration of battery life. Approaches to optimize the power level applications and management resource management of mobile devices are considered. The examples of existing domestic and foreign theories and their brief characteristics, a bibliography on the subject are given.

Keywords: ecology, informational computer technology, environmental software, mobile devices, optimization, power profiling, review.

Туркін Ігорь Борисович – д-р техн. наук, проф., зав. каф. інженерії програмного забезпечення, Національний аэрокосмічний університет ім. Н. Е. Жуковського «Харьківський авіаційний інститут», Харьков, Україна, e-mail: nikrutrogi@mail.ru.

Вдовитченко Александр Валерьевич – аспірант каф. інженерії програмного забезпечення, Національний аэрокосмічний університет ім. Н. Е. Жуковського «Харьківський авіаційний інститут», Харьков, Україна, e-mail: kentsanya91@gmail.com.

Аль-Кхшаб Синан Самір – аспірант каф. інженерії програмного забезпечення, Національний аэрокосмічний університет ім. Н. Е. Жуковського «Харьківський авіаційний інститут», Харьков, Україна, e-mail: sinan_alkhshab@mail.ru.