Intraoperation Dynamic Voltage Scaling

# ABSTRACT

Embedded peripherals are often specified with a range of performance characteristics that are affected by their supply voltage. Typically the supply voltage is static and therefore both power consumption and performance traits are known at design-time. With Enhanced Dynamic Power Management (IODVS), we concentrate on reducing the intra-operation power consumption of peripheral devices by dynamically modulating the supply voltage. IODVS is designed to have minimal impact on CPU utilization through the use of peripheral power profiles (PPP) which designate an ideal voltage on a per-state basis. Any peripheral operation seamlessly flows through the pre-determined states and the supply voltage is modulated automatically upon each transition. Peripheral power profiles are unique in that during high-performance states such as data-transmission, peripherals can have the high supply voltage they demand. Likewise, during low-performance states such as mandatory delays, the system can decrease domain voltage and thus reduce power consumption intra-operation. We demonstrate this method on various common peripherals and have found energy savings ranging from 15% - 47%.

# Introduction

We begin by considering an embedded system where the supply voltage to an application MCU is decoupled from the supply voltage of the peripherals that it is controlling. This is becoming more common as modern MCU applications take advantage of Dynamic Voltage and Frequency Scaling (DVFS) and, in effect, IODVS is a natural extension of DVFS to the peripheral domain. The same modulation techniques (DAC, PWM, etc.) that a MCU may use to control its own voltage can be used to control peripheral voltages. We have found that energy can be saved by lowering the domain voltage during low-performance periods.

Consider the Microchip SPI EEPROM [[1](#Mic10)] as a typical peripheral device. A typical write operation of the device has the following state transitions and timings:

|  |  |  |  |
| --- | --- | --- | --- |
| Chip Select, Write Enabled | Write Cmd/Data | **Delay** | Verify Cmd/Data, Chip Deselect |
| 1us | 128us | **5ms** | 128us |

Table : EEPROM Write Cycle

Both of the read and write operations are voltage/frequency dependent in that the 25AA512 can communicate at 20MHz while above 4.5v, 10MHz while above 3.3v and 2MHz while above 1.8v. It follows that one should communicate between the two domains at matched voltages thereby maximizing data transfer while minimizing energy delay product (EDP). The maximum benefit of IODVS can be realized during the longest portion of the transaction: the delay. By decreasing the supply voltage to 1.8v during the delay state, the energy cost of the delay is decreased by 58%.

The IODVS technique is applicable to many peripherals and this investigation considered the peripherals listed in Figure 1 as a representative sample:

|  |  |  |
| --- | --- | --- |
|  | Honeywell HIH-6130 I2C  Temperature / Humidity Sensor | Vmax: 5.5V  Vmin: 2.3V |
| Microchip MCP 25AA512  512Kbit (64KB) SPI EEPROM | Vmax: 5.5V  Vmin: 1.8V |
| Numonyx M25PX16  16Mbit (2MB) SPI Serial Flash | Vmax: 3.6V  Vmin: 2.3V |
| SwissBit S-200u  512MB (SPI Mode) SD Card | Vmax: 3.6V  Vmin: 2.7V (Operating)  Vmin: 2.0V (Idle/Ready) |

Table : Typical External Peripherals

Enabling IODVS requires only an adjustable power supply. An adjustable linear regulator could be used; however in that case one would realize only the benefits of decreased current consumption. This experiment made use of the TPS62240 adjustable switched mode power supply (SMPS) in order to maximize efficiency gains. Peripheral domain voltage modulation is accomplished via DAC output on the STM32F205 MCU signaling into the resistive feedback circuit on the SMPS. In order to measure the results of IODVS, the domain is outfitted with current sense circuitry on both the input to the SMPS and the output to the domain.

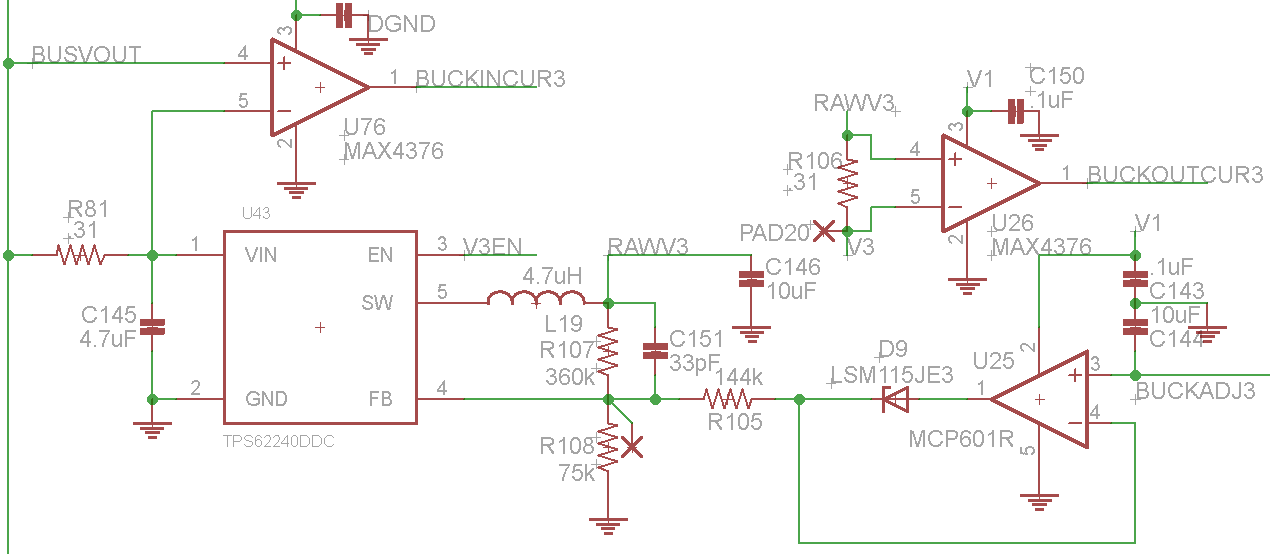


Figure : Peripheral Domain SMPS, Control and Current Sense Circuitry

IODVS is thoroughly tested on each of the four sample peripherals by conducting 1000 pseudo-random tests on each device. The output is analyzed and if any particular operation fails then the test is considered a failure and the PPP is increased (voltage slack is decreased) until all tests complete as expected.

# Related Work

Dynamic Power Management (DPM) and Dynamic Voltage Scaling (DVS) implementations seek to maximize energy efficiency in an embedded system when scheduling the use of external peripherals. DPM policies tend to focus on strict power-state relationships [[2](#Bro03)] while DVS policies tend to incorporate a linear power-performance relationship [[3](#Jej04)]. Most DPM implementations focus on optimal scheduling techniques such that peripherals emerge from shutdown just in time for access by tasks. Generally, the approaches to date can be categorized as a combination of either online [[4](#Hui06)] or offline [[5](#Kum08)] and deterministic [[6](#Swa03)] or probabilistic [[7](#Ira02)].

Offline analysis can aid in the implementation of DPM by analyzing the CFG of a task to determine when a peripheral is likely to be accessed [[5](#Kum08)]. Similar data can be realized online by profiling a task and determining which paths lead to a peripheral access [[8](#You10)]. Both methods enhance the accuracy of predictions regarding the optimal peripheral wakeup time. In fact, all methods must evaluate the cost/benefit of peripheral deactivation with respect to the energy savings gleaned versus the time spent reactivating the device when next needed. This equality is commonly known as the breakeven time [[9](#Edw09)].

Some peripherals provide multiple performance/power states. As such, Mode Dependence Graphs were developed in order to accurately quantify the breakeven time between states [[10](#Dex02)]. Approaches have been explored with respect to optimally scheduling devices with multiple power saving states and in systems where multiple tasks share a common resource (inter-task DPM). Naturally, the decrease in voltage margin along with the decrease in available task slack time also decreases the ability to detect and correct errors as they occur [[11](#Dak06)].

The IODVS technique is different in that we seek to decrease the energy cost of performing peripheral operations as they are performed.

# Assumptions

Create a completely controllable buck power supply via analog input through a DAC. The power source will be supplying voltage for multiple peripherals on a domain separate from the MCU. All digital transactions between the MCU and the peripheral domain will be made at the same voltage. The cost of level translation or isolation is too great to warrant implementation. Also, various sources have cited that the lowest EDP of communication occurs at matched voltage/frequencies. Thus we are left with intra-operation voltage modulation as our means of decreasing energy consumption.

# Methods and Materials

The peripheral power supply (PPS) is outfitted with current sense resistors and amplifiers on both the input to the PPS and the output to the peripheral domain. These signals, along with the input voltage to the PPS and the output voltage from the PPS are fed into the ADC of the STM32F205 microcontroller and sampled at 1MSPS. The MCU has 3 simultaneously sampling ADCs which allows for simultaneous measurement of the output voltage, input current and output current.

Peripheral operations are broken up into states as per an intrinsic state transition diagram. For example, in order to write to EEPROM, the MCU must issue the write command and write the data, wait for a specified delay period and then read the data back in order to verify a correct write. Therefore, the states are delineated as Idle, Writing, Waiting and Verifying.

Each peripheral operation is associated with a specific voltage. For instance, as per our assumptions, data transfers must occur at equal voltages between the domain and MCU. Therefore the Writing and Verifying states voltage must equal that of the MCU (3.3v). This leaves the Idle and Waiting states free for energy optimization.

The set of states and associated voltages creates a power profile per peripheral. Each test designates a power profile to use. Tests were run 1000x and the results were averaged. If any test failed to complete the operation successfully then the power profile was adjusted until operation is guaranteed.

# Results

Here I will provide a brief introduction to all of the results

For instance, the fact that the domain voltage decreases at the rate of:

|  |  |  |
| --- | --- | --- |
|  |  | () |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Figure : EEPROM Write Procedure   |  |  |  |  | | --- | --- | --- | --- | | State | Static | IODVS | Delta | | Idle | 3.10 | 1.16 | -62.52% | | Write | 3.80 | 4.45 | 17.08% | | Wait | 18.09 | 8.90 | -50.79% | | Verify | 4.97 | 6.05 | 21.73% | | Idle | 2.73 | 2.73 | -0.11% | | Total | 32.6852 | 23.2879 | -28.75% |   Table : EEPROM Write Energy Consumption | Macintosh HD:Users:drmoore:Documents:PEGMA:Documentation:Results:EDPM:eepromResult0.png  Figure : EEPROM Test Results |

## Microchip MCP25AA512 EEPROM

Here I will talk about the EEPROM characteristics

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Figure : EEPROM Write Procedure   |  |  |  |  | | --- | --- | --- | --- | | State | Static | IODVS | Delta | | Idle | 3.10 | 1.16 | -62.52% | | Write | 3.80 | 4.45 | 17.08% | | Wait | 18.09 | 8.90 | -50.79% | | Verify | 4.97 | 6.05 | 21.73% | | Idle | 2.73 | 2.73 | -0.11% | | Total | 32.6852 | 23.2879 | -28.75% |   Table : EEPROM Write Energy Consumption | Macintosh HD:Users:drmoore:Documents:PEGMA:Documentation:Results:EDPM:eepromResult0.png  Figure : EEPROM Test Results |

## Numonyx M25PX16 Serial Flash

Here I will talk about the serial flash [[12](#Mic12)]

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Figure : EEPROM Write Procedure   |  |  |  |  | | --- | --- | --- | --- | | State | Static | IODVS | Delta | | Idle | 3.10 | 1.16 | -62.52% | | Write | 3.80 | 4.45 | 17.08% | | Wait | 18.09 | 8.90 | -50.79% | | Verify | 4.97 | 6.05 | 21.73% | | Idle | 2.73 | 2.73 | -0.11% | | Total | 32.6852 | 23.2879 | -28.75% |   Table : EEPROM Write Energy Consumption | Macintosh HD:Users:drmoore:Documents:PEGMA:Documentation:Results:EDPM:eepromResult0.png  Figure : EEPROM Test Results |

## Swissbit S-200U 512MB Micro-SD Memory Card

Here I will talk about the SD Card

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Figure : EEPROM Write Procedure   |  |  |  |  | | --- | --- | --- | --- | | State | Static | IODVS | Delta | | Idle | 3.10 | 1.16 | -62.52% | | Write | 3.80 | 4.45 | 17.08% | | Wait | 18.09 | 8.90 | -50.79% | | Verify | 4.97 | 6.05 | 21.73% | | Idle | 2.73 | 2.73 | -0.11% | | Total | 32.6852 | 23.2879 | -28.75% |   Table : EEPROM Write Energy Consumption | Macintosh HD:Users:drmoore:Documents:PEGMA:Documentation:Results:EDPM:eepromResult0.png  Figure : EEPROM Test Results |

## Honeywell HIH6130 Temperature / Humidity Sensor

Here I will talk about the Honeywell sensor

* Microchip EEPROM: 34% Lower
* Numonyx Serial Flash: 51% Lower
* Swissbit SDCard: 15% Lower
* HIH6130 Temperature / Humidity sensor: 47% Lower

# Conclusions

# References

x

|  |  |
| --- | --- |
| [1] | Microchip Technology Inc. (2010, May) Microchip 25AA512 Datasheet. [Online]. <http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en530926> |
| [2] | B. Brock and K. Rajamani, "Dynamic power management for embedded systems [SOC design]," in *SOC Conference, 2003. Proceedings. IEEE International [Systems-on-Chip]*, 2003, pp. 416-419. |
| [3] | R. Jejurikar and R. Gupta, "Dynamic Voltage Scaling for Systemwide Energy Minimization in Real-Time Embedded Systems," in *Proceedings of the 2004 International Symposium on Low Power Electronics and Design, ISLPED*, 2004, pp. 78-81. |
| [4] | Hui Cheng and S. Goddard, "Online energy-aware I/O device scheduling for hard real-time systems," in *Design, Automation and Test in Europe, 2006. DATE '06. Proceedings*, 2006, pp. 6-10. |
| [5] | C.M. Kumar, M. Sindhwani, and T. Srikanthan, "Profile-based technique for Dynamic Power Management in embedded systems," in *Electronic Design, 2008. ICED 2008. International Conference on*, 2008, pp. 1-3. |
| [6] | V. Swaminathan and K. Chakrabarty, "Energy-conscious, deterministic I/O device scheduling in hard real-time systems," in *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, 2003, pp. 847-858. |
| [7] | S. Irani, S. Shukla, and R. Gupta, "Competitive analysis of dynamic power management strategies for systems with multiple power saving states," in *Design, Automation and Test in Europe Conference and Exhibition, 2002. Proceedings*, 2002, pp. 117-123. |
| [8] | Young-Si Hwang, Sung-Kwan Ku, and Ki-Seok Chung, "A predictive dynamic power management technique for embedded mobile devices," in *Consumer Electronics, IEEE Transactions on*, 2010, pp. 713-719. |
| [9] | Tai-Yi Huang, Cheng-Han Tsai, Jian-Jia Chen, Tei-Wei Kuo Edward T.-H. Chu, "A DVS-assisted hard real-time I/O device scheduling algorithm," *Real-Time Systems*, vol. 41, pp. 222-255, February 2009. |
| [10] | Dexin Li, P.H. Chou, and N. Bagherzadeh, "Mode selection and mode-dependency modeling for power-aware embedded systems," *Design Automation Conference, 2002. Proceedings of ASP-DAC 2002. 7th Asia and South Pacific and the 15th International Conference on VLSI Design. Proceedings*, pp. 697-704, 2002. |
| [11] | Dakai Zhu, "Reliability-Aware Dynamic Energy Management in Dependable Embedded Real-Time Systems," in *Real-Time and Embedded Technology and Applications Symposium, 2006. Proceedings of the 12th IEEE*, 2006, pp. 397-407. |
| [12] | Micron Technology Inc. (2012) M25PX16 Datasheet. [Online]. <http://www.micron.com/parts/nor-flash/serial-nor-flash/m25px16-VMN6P> |
| [13] | Honeywell International Inc. (2013) Honeywell Sensing and Control. [Online]. <http://sensing.honeywell.com/product-page?pr_id=142040> |
| [14] | Swissbit AG. (2014) Swissbit. [Online]. <http://www.swissbit.com/images/stories/pdf2/S-200u_data_sheet_SD-NxBN_Rev111.pdf> |
| [15] | L.B. Hormann, P.M. Glatz, C. Steger, and R. Weiss, "Evaluation of component-aware dynamic voltage scaling for mobile devices and wireless sensor networks," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2011 IEEE International Symposium*, vol. 1, 2011, pp. 20-24. |

x