Intra-Operation Dynamic Voltage Scaling

Daniel R. Moore Alexander G. Dean
Center for Efficient, Scalable and Reliable Computing
Dept. of Electrical and Computer Engineering
North Carolina State University, Raleigh, NC 27695, USA
{drmoore2, agdean}@ncsu.edu

***Abstract*—Embedded peripheral devices are often specified with a range of performance characteristics that are determined by their supply voltage. Recent research explored the benefits of modulating peripheral supply voltage with task-level granularity. With Intra-Operation Dynamic Voltage Scaling (IODVS), we further reduce the energy consumption of peripheral devices by modulating the peripheral supply voltage at critical states occurring during operation of the peripheral device. IODVS is designed to have minimal impact on CPU utilization through the use of a lookup table that designates an ideal voltage on a per-state basis. IODVS is unique in that during high-performance states such as data-transmission, peripherals can have the high supply voltage required to reduce overall energy-delay product. Likewise, during low-performance states such as mandatory delays, the system decreases peripheral domain voltage thus reducing energy consumption without adversely affecting performance or correctness. We demonstrate this method on various peripherals common to wireless sensor nodes and have found total energy savings of up to 40%.**

Keywords—Embedded Systems; Dynamic Voltage Scaling (DVS); Dynamic Power Management (DPM); low-power; low-energy; wireless sensor node (WSN); energy-aware design.

I. Introduction

Consider an embedded system where the supply voltage to an application MCU is decoupled from the supply voltage of its peripherals as shown in Fig. 1. This design 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. We find that energy can be saved by lowering the peripheral domain voltage during periods where low-performance is allowed such as mandatory wait periods and where the application of traditional DVS or DPM techniques would adversely affect operation of either the device or the system.

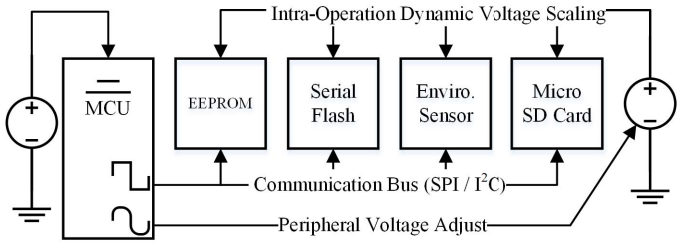


Fig. 1. An IODVS Enabled System

For example, EEPROM is a typical peripheral device that is used to provide non-volatile data storage. They are usually specified for use in systems that require a quick data access time and have low storage capacity requirements. The chips are often specified to operate at multiple voltage levels to achieve compatibility with systems using voltages from 1.8V to 5.0V.

A write operation to the SPI device (and optional verification stage) is typified by the timing diagram shown in Fig. 2. Maximum communication speed scales with slew rate and therefore scales with voltage. It follows that communication between the MCU and peripheral domains should occur at matched voltages, thereby maximizing data transfer, minimizing energy delay product (EDP) and eliminating the need for voltage level translation.

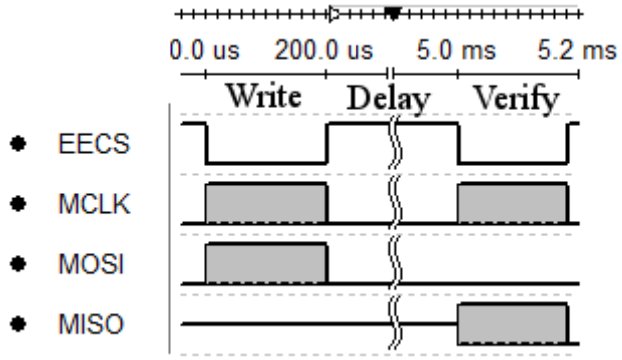


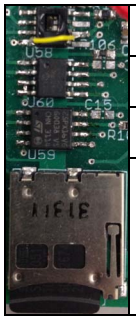
Fig. 2. A SPI EEPROM Write / Verify Cycle

The most distinct benefit of IODVS can be realized during the longest portion of the transaction described in Fig. 2: the delay. IODVS decreases the supply voltage to the chip during this voltage-independent period and we find that the energy cost of the transaction is significantly decreased.

The IODVS technique is applicable to any peripheral that has a voltage/frequency dependence and particularly applicable to those with a wait-state. Our investigation considered the peripherals listed in Table I as a representative sample. The device descriptions and voltage requirements are listed next to their physical location on the test fixture.

Enabling IODVS requires only an adjustable power supply and a means of modulating the output voltage. A switched mode power supply (SMPS) is preferable because it is an efficient means of translating voltage levels. An adjustable linear regulator could be used, but only the benefits of decreased current consumption would be realized.

TABLE I. TYPICAL EXTERNAL PERIPHERALS

	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
	SPI Mode SD Cards: Lexar SDSC: 1.0GB Sandisk SDSC 1.0GB SwissBit: SDSC 512MB Kingston: SDSC: 2.0GB	Vmax: 3.6V Vmin: 2.7V (Operating) Vmin: 2.0V (Idle/Ready)

IODVS was tested on each of the seven sample peripherals listed in Table I. Each device was characterized by a peripheral power profile (PPP) that is effectively a lookup table of state-voltage pairs. The state voltages were derived from the specifications of the device datasheet. A common sequence of operations was performed repeatedly and random input parameters were used on each iteration. The output was analyzed and no failures or unexpected behavior occurred. We observed idle energy decreases of up to 66% and intra-operational energy decreases of up to 40%.

II. RELATED WORK

Dynamic Power Management and Dynamic Voltage Scaling implementations seek to minimize energy consumption in embedded systems. DPM policies tend to focus on strict power-state relationships [1], while DVS policies tend to incorporate a linear power-performance relationship [2]. In fact, DVFS is so useful that hardware is designed specifically to take advantage of it [3]. Most DPM implementations focus on optimal scheduling techniques such that peripherals emerge from low-power states just in time for access by tasks. Generally, the approaches to date can be categorized as a combination of either online [4] or offline [5] and deterministic [6] or probabilistic [7].

Offline analysis can aid in the implementation of DPM by analyzing the control flow graph of an individual task or task set to determine when a peripheral is likely to be accessed [5]. Similar data can be realized online by profiling tasks and determining which paths lead to a peripheral access [8]. 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 relationship is commonly known as the breakeven time [9].

Peripherals can be operated under the same linear [10] and step-wise [11] constraints. Approaches have been explored with respect to optimally scheduling devices with multiple power saving states and with 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 [12]. In fact, DVFS is so useful that dual-output circuits have been developed primarily targeting systems with a SoC [13]. The same circuitry could be reused to implement IODVS.

IODVS is most similar to the Component Aware DVS technique [14] [15] developed for use in wireless sensor nodes (WSN) [16] [17]. An adjustable regulator is used to operate an embedded system at its minimum voltage requirements. However, CADVS operates at the task-level and therefore results in a power / performance relationship typical of DVS. IODVS differs in that it extends the technique into intra-operation granularity.

DPM techniques inherently interfere with the operation of the device and impose a lag in response time. IODVS instead exploits the acceptable operating voltages of the device and does so with no effect on response time.

III. ASSUMPTIONS

The MCU and peripheral voltage domains are decoupled and the peripheral domain is adjustable using an MCU-controlled DAC. All digital signaling between the MCU and the peripheral domain are made at the same voltage. The increased cost and decreased performance of level translation or isolation is too great to warrant implementation [18] for this purpose in most embedded systems. Above all, the lowest communication energy-delay product is found at matched voltage/frequencies.

The current measurements are taken at the output of the peripheral power supply. Thus, the data will indicate the effect of IODVS on the set of peripherals on the domain and not on any one peripheral in particular.

The PPP state-voltage lookup table of each device is constructed solely from the acceptable usage specifications contained within the device datasheet. It was discovered experimentally that many of the devices that we tested operated well below their specified minimum voltage requirements. Although minimizing energy consumption by means of minimizing voltage is the primary goal of this work, it is necessary to ensure functionality of the device is maintained across all environments that may degrade performance.

For instance, the EEPROM under test is specified to operate in the range of -40℃ to +80℃ and with a minimum endurance of 1,000,000 write cycles. As the device nears the edge of its acceptable operating temperature or approaches its lifetime write-cycle limit, the minimum necessary voltage to guarantee completion of a write operation is likely to be that specified by the designers along with an acceptable factor of safety.

IV. METHODS AND MATERIALS

The TPS62240 [19] adjustable switched mode power supply was selected to power the peripheral domain because of its high efficiency at light loads, output capacity and adjustability. Peripheral domain voltage modulation is accomplished using a DAC output on a STM32F205 MCU signaling into the resistive feedback circuit on the SMPS. To measure the results of IODVS, the domain is outfitted with current sense circuitry [20] on both the input to the SMPS and the output to the domain.

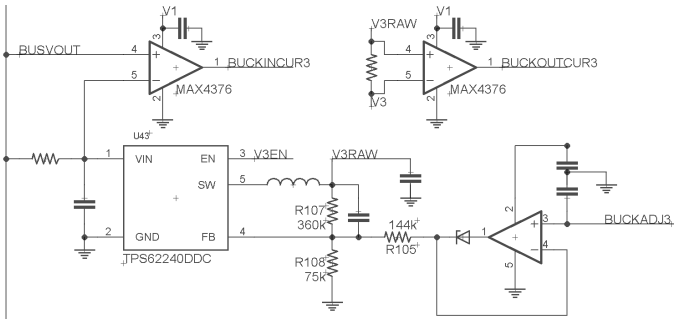


Fig. 3. Peripheral Domain SMPS, Control and Current Sense Circuitry

As shown in Fig. 3, the adjustable peripheral power supply is outfitted with current sense resistors and amplifiers on both the input to the supply and the output to the peripheral domain. These signals, along with the input voltage to the supply and the output voltage to the domain are fed into the ADC of the STM32F205 microcontroller and sampled at up to 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 per an intrinsic state transition diagram. For example, to perform a write to EEPROM, the MCU must issue the write command, 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, per our assumptions, data transfers must occur at equal voltages between the domain and MCU. The voltages of the writing and verifying states must then equal that of the MCU (3.3V). This leaves the idle and wait states available for voltage scaling.

For each device, the pairs of states to voltages form a lookup table (PPP). Each test designates a power profile to use. Peripheral memory was tested with random data and across random memory addresses. Tests were run 2048 times, and the results were averaged. While operating IODVS within the specifications of the device datasheet, no operations failed.

All test results were measured entirely in-system using the three 12-bit simultaneously sampling ADC converters onboard the MCU. The converters are triggered from a timer overflow using a reload value that allows for a complete buffer fill roughly corresponding to the expected length of the test. For example, the duration of the EEPROM test was approximately 10ms with a buffer size of 10240 samples yielded 976.6ns per sample (or a rate of 1.024MHz). Upon a trigger, the state of the peripheral is stored synchronously with the sample. Each test data set was retrieved upon completion and is composed of:

- Time Scale
- 10240 12-bit ADC Samples per channel
- Output Voltage
- Input and Output Current
- 10240 Device State Samples (reading / writing / etc.)
- Bit Resolution (ADC value → Current or Voltage)

The energy consumed throughout a test is calculated using the fundamental relationship shown in (1). The results were calculated offline via (2) and (3), where S is the state of the device, and T_s is the sampling period.

$$P = VI = \frac{E}{t} \tag{1}$$

$$E_s = \sum_{n=0}^{N-1} V_n I_n T_s \tag{2}$$

$$E_{total} = \sum_{S_0}^{S_{n-1}} E_s \tag{3}$$

Separating the energy consumption by state is important because it allows us to consider the effect of duty cycle on the results of IODVS. Each device has an idle state where the voltage applied to the device is the minimum allowed by specification. Applying IODVS to the idle state significantly decreases energy consumption while avoiding the increased response time of DPM.

Likewise, IODVS is applicable to an exploitable sequence of active operations, resulting in decreased energy consumption without the performance impacts of DVFS. We can separate energy consumption into two intervals as shown in (4).

$$E_{total} = E_{idle} + E_{active} \tag{4}$$

$$E_{total} = P_{idle} * T_{idle} + P_{active} * T_{active} \tag{5}$$

A duty cycle of 0% will be dominated by T_{idle} and energy consumption will converge on that of the idle state. On the other hand, a duty cycle of 100% will be dominated by T_{active}. In which case, energy consumption converges on the weighted average of the set of states comprising the active period. In any case, the actual energy decrease due to IODVS will lie in between these two extremes.

V. Results

The effects of domain capacitance are immediately visible. The domain voltage changes at a rate corresponding to (6). This is most discernible as the voltage transitions from high to low because the power supply has a high current drive capability but no current sink circuitry. This is a benefit to IODVS in that peripheral performance is unaffected by higher-than-necessary voltage in the states of concern, which effectively allows peripheral voltage to decay from the higher communicating voltage to the ideal voltage for the operation being performed.

$$V = V_0 e^{\frac{-t}{RC}} \tag{6}$$

$$I = C\frac{dV}{dt} \tag{7}$$

Likewise, on low to high transitions, the output current of the power supply spikes to charge the domain as quickly as possible via (7). There is indeed a response time to IODVS, and it is proportional to the output current sourcing capability of the adjustable peripheral power supply. The supply under test was capable of changing the output voltage of the peripheral domain from 1.8V to 3.3V approximately 75us.

A. *Microchip MCP25AA512 EEPROM*

IODVS uses peripheral power profiles to correlate peripheral voltages with internal state. The PPP specified for the EEPROM under test is derived from the specifications of its datasheet [21]. The EEPROM can communicate at 10MHz at 3.3V, while only 1.8V is required for basic operation. However, the length of the mandatory page-write delay is voltage independent and exploitable by IODVS.

The standard PPP is considered a control group and mandates that all states (writing/waiting/verifying/etc.) should have 3.3V applied to the peripheral. The 1.8VIW (1.8V Idle/Wait) profile mandates that the EEPROM should have 1.8V applied during the idle and waiting states and 3.3V applied on all others. Fig. 5 provides a comparison of both the standard PPP and the 1.8VIW profiles enabled by IODVS.

The state transition diagram of Fig. 4 is known a-priori and is followed throughout the tests illustrated in Fig. 5. The black line indicates device state and is sampled synchronously with the voltage and current measurements.

The test begins with the EEPROM powered up and in the idle state. The WREN (write enable) command is transmitted to the peripheral, along with the write command and address which is followed by 128 bytes of random data (1 page-size). The peripheral and device driver then transition into the page-write delay state and the peripheral voltage is decreased to 1.8V. After the delay, the device driver increases the voltage to the 3.3V necessary for communication and then reads the data back from the device to verify that it was committed properly.

The effects of IODVS are most distinct during the Idle and Wait states. Energy consumption during these states decreased 66.7% and 48.7% respectively. Energy consumption during the Write state increased by 30%. This is primarily as a result of the energy required to charge the domain to 3.3V which is required to complete the transaction.

Note that although the current measurement appears to exceed the graph in Fig. 5, the current spike was indeed measured to be approximately 15mA and the data were integrated accordingly. In fact, charging the domain voltage is responsible for the 29% and 37% increases in the write and verify states respectively.

Two of the SPI lines on the test fixture are multiplexed for I2C communication. This causes the 1mA current swings during the communication phases of the test. The current consumption of the device indicates the behavior of the operation within. Two distinct periods of increased power demand are noticeable, these are likely to be an internal erase operation followed by a write.

The idle time of the test lasted 1ms out of a total test time of 9.475ms. The duty cycle of this test was 89.45%, and energy consumption was reduced by 26.67%. Removing the idle time from the total would yield an energy decrease of 22.36% at a duty cycle of 100%. Realistically, this type of device is used much less frequently owing to its finite number of useable write-cycles. At a duty cycle of 0%, the savings would converge on 66.7%.

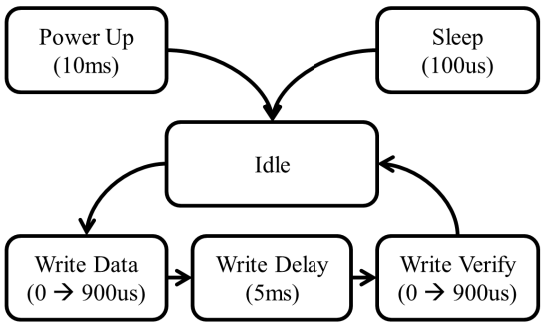


Fig. 4. EEPROM Write Procedure

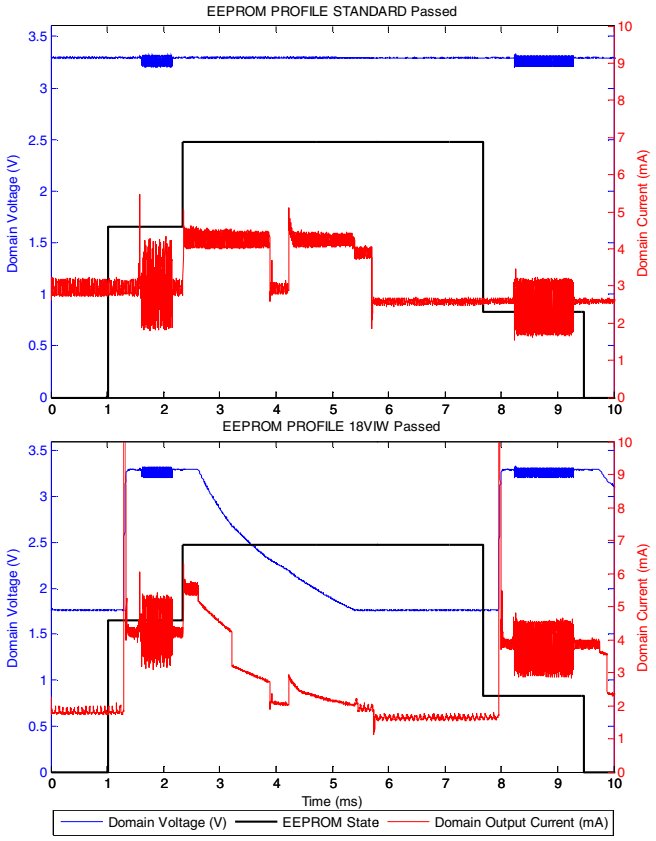


Fig. 5. EEPROM Test Results

TABLE II. EEPROM ENERGY CONSUMPTION

State	Static (µJ)	IODVS (µJ)	Delta
Idle	9.84	3.28	-66.70%
Write	13.28	17.08	28.61%
Wait	62.03	31.83	-48.69%
Verify	16.12	22.08	36.96%
Test Total	**101.27**	**74.26**	**-26.67%**

TABLE III. ENERGY CONSUMPTION VS. DUTY CYCLE

Duty Cycle	Static avg. (µJ)	IODVS avg. (µJ)	Delta
Duty: 0%	9.84	3.28	-66.70%
Duty: 25%	30.24	20.20	-33.19%
Duty: 50%	50.63	37.13	-26.67%
Duty: 75%	71.03	54.05	-23.90%
Duty: 100%	91.42	70.98	-22.36%

B. *Numonyx M25PX16 Serial Flash*

Serial flash modules have a somewhat more complicated state transition diagram than EEPROM. Serial flash chips can only program zeroes to their memory locations. At a simplistic level, this requirement necessitates a complete erase of a subsector before modifying the memory within it. The M25PX16 [22] supports a minimum of 4KB (sub-sector) erase and a maximum of 256B (page) sequential writes. To perform a read-modify-write operation, the transition diagram shown in Fig. 6 is followed.

As with all of the devices under test, the control PPP was standardized at 3.3V throughout, while the 23VIW (2.3V idle/wait) PPP was constructed from the parameters listed within the datasheet. The device has a minimum operational voltage of 2.3V which is used for the idle and wait states. The subsector erase is specified to take a maximum of 150ms, while the page-write completes with a maximum delay of 5ms. Cross-subsector writes were not evaluated because that would simply require two test sequences to occur sequentially.

The device can reach the idle state either 10ms after power up or approximately 30us after the execution of a wake command. As the test begins, the chip is in the idle state; it does not require an initialization routine to execute before entering a functional state. A random sub-sector of memory is read into cache and is modified with the 128 bytes of random data to be committed. The sub-sector erase operation is executed, and IODVS adjusts the peripheral voltage to the wait state (2.3V in this PPP).

Upon completion of the erase cycle, the modified cached data are written back to the flash module one page at a time, resulting in 16 total page-writes. The writes cause a series of alternating "write-wait" states, and the corresponding voltage/state changes are evident in Fig. 7. After the final page-write delay is complete, the data are read back and verified with the cached copy to ensure data integrity.

Table IV summarizes the energy decrease per state yielded through the use of IODVS. As expected, the most significant savings are found in the idle and wait states, while an increase is seen in the active states.

Because the test was in the idle state for 1ms out of the total length of 257ms, the test represents a duty cycle of 99.6%, which is effectively the worst case. As the duty cycle decreases, the idle energy savings begins to dominate and pushes the average toward a limit of 48.66%.

It is noteworthy that this erase-write sequence is common to all flash memory, and so IODVS is applicable to flash memory in general. In high performance parallel NOR and NAND devices, writes complete on the order of micro-seconds. However, erase operations complete on the order of seconds and are easily exploitable by IODVS.

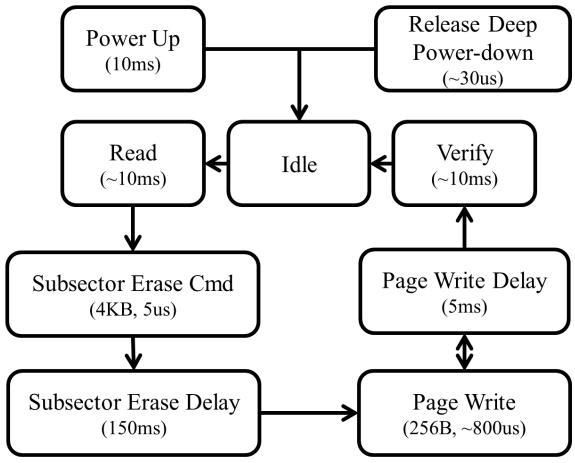


Fig. 6. Serial Flash State Transition Diagram

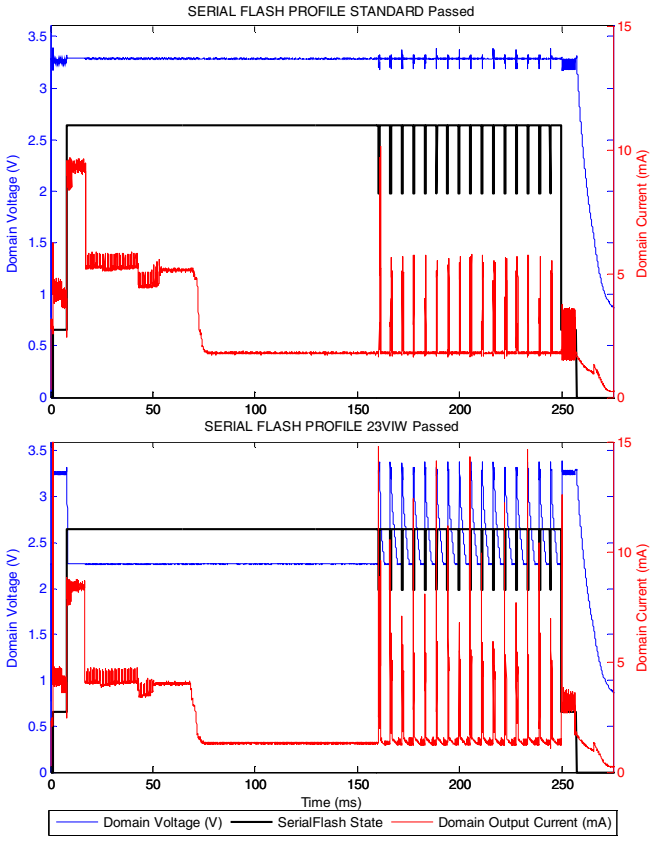


Fig. 7. Serial Flash Read-Modify-Write Test Results

TABLE IV. SERIAL FLASH ENERGY CONSUMPTION

State	Static (µJ)	IODVS (µJ)	Delta
Idle	10.27	5.27	-48.66%
Reading	89.85	90.86	1.13%
Write*	80.35	89.20	11.02%
Wait*	551.18	344.92	-37.42%
Verify	57.52	72.45	25.96%
Test Total	**2517.04**	**1530.27**	**-39.20%**

*sequential write and wait states combined

C. *Micro-SD Memory Card Results*

Micro-SD Cards follow a standard outlined by the SD Association [23]. The standard is a minimum set of electrical and communication specifications that must be met and some vendors exceed those specifications [24] [25]. A few of the variable parameters include clock speed, slew rate, initialization time, block length, read/write timing and power consumption. Additionally, the devices use a MMU which causes access timing to vary. The cumulative effect of these variations results in the SD Card protocol relying heavily on device polling. We avoid polling during write operations by predicting the write completion time and tuning the optimal polling time is a topic for further research.

An SD Card must be initialized after power up as shown in Fig. 8. The MCU communicates with the SD Card via SPI and the initialization process typically takes 250ms. Not all SD Cards support power down modes. IODVS enables the device to transition to the 2.7V "Initialized" state, rather than undergoing a complete power-cycle and incurring the 250ms penalty as would be typical with DPM.

From the initialized state, the device was sent a write command to a random address with random data. The device driver then waits a predetermined amount of time (the prediction) before beginning to poll the device for write complete which can take up to 250ms. After the write finishes, the device driver reads the data back in order to verify that it committed properly before returning to the idle state.

1) Sandisk SDSC 1.0GB Micro-SD Memory Card

Initial experiments with the Sandisk Micro-SD indicated that the majority of write operations completed approximately 150-170ms after they began. Based on this data and as shown in Fig. 9, the card was not polled until the test reached the 180ms mark (which is approximately 165ms after the write command completed successfully). After write-complete polling begins, all of the writes had already completed and were eligible to transition to the verification stage.

Idle energy consumption dropped by 11.5% and the idle duration accounted for 10ms of the 184.1ms test, yielding a duty cycle of 94.6%. A duty cycle of 100% (constant write/verify) would yield an energy decrease of 27.54%.

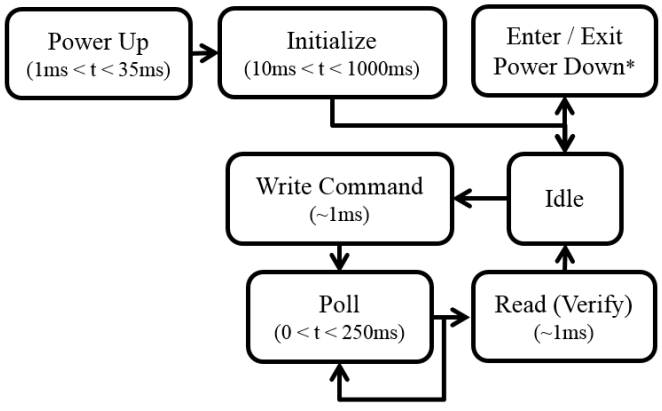


Fig. 8. Typical Micro-SD Card Operation

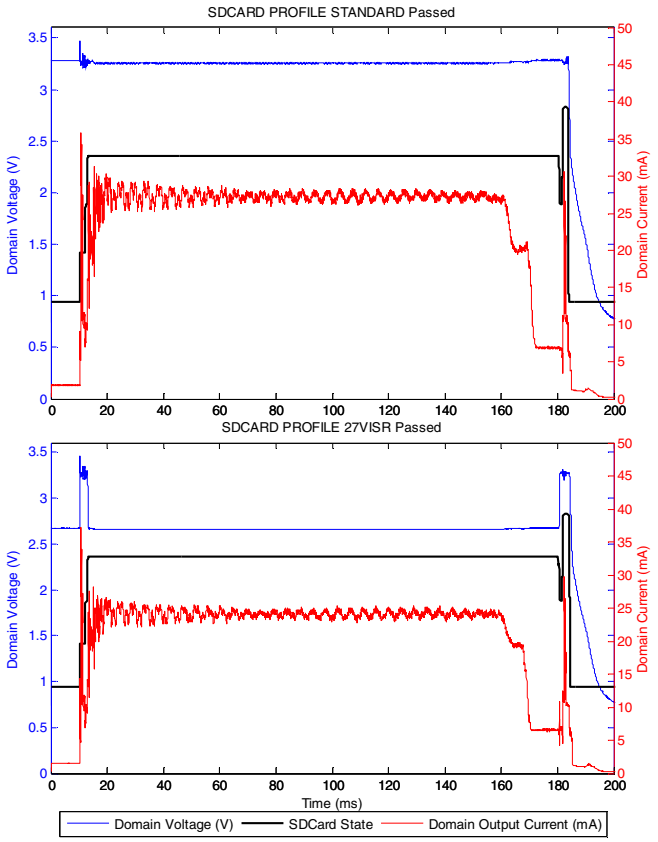


Fig. 9. Sandisk Micro-SD Card Write Operation

TABLE V. SANDISK MICRO-SD CARD ENERGY CONSUMPTION

State	Static (µJ)	IODVS (µJ)	Delta
Idle	157.07	138.95	-11.54%
Write	26.48	26.23	-0.93%
Wait	14021.97	10126.95	-27.78%
Verify	89.86	91.88	2.25%
Test Total	**14295.38**	**10384.02**	**-27.36%**

2) Lexar SDSC 1.0GB Micro-SD Memory Card

The Lexar Micro-SD card had a higher average power draw and a wider write-completion characteristic than the Sandisk Micro-SD Card. The majority of writes completed between 140-180ms after the test began. Polling began at the 160ms mark and nearly 50% of the writes were already complete.

The duty cycle was the same as the Sandisk card at 94.6% yielding an energy decrease of 24.12%. Both the Sandisk and Lexar cards are older technology (manufactured in 2007 and 2009 respectively) and when compared with other cards, show higher energy consumption and slower performance. Newer implementations are better in both categories.

TABLE VI. LEXAR MICRO-SD CARD ENERGY CONSUMPTION

State	Static (µJ)	IODVS (µJ)	Delta
Idle	124.09	102.41	-17.47%
+Write	34.52	34.42	-0.28%
Wait	16608.43	12558.83	-24.38%
Verify	39.45	56.87	44.17%
Test Total	**16806.48**	**12752.54**	**-24.12%**

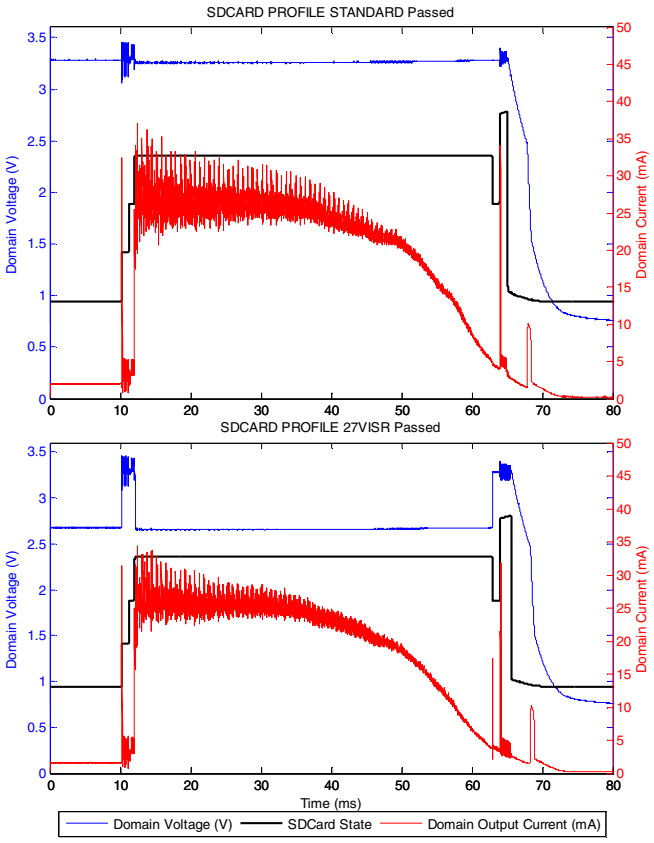


Fig. 10. SwissBit Micro-SD Card Operation

3) Swissbit S-200U 512MB Micro-SD Memory Card

The SwissBit Micro-SD Card is unique in that it uses 4x 4KB buffers to cache reads and writes to the memory card in order to speed up transaction times. The method is effective in that the worst case test time for the SwissBit card is less than half the best case test time for the previous two cards.

The card is equipped with power-fail circuitry that flushes the buffers to non-volatile memory once a voltage threshold has been reached. This functionality is evident at the moment just before the 70ms mark where the peripheral voltage reaches approximately 2.5V coinciding with a current spike of approximately 9mA.

The write-completion time varies much more significantly than the other cards. Current consumption of the device shown in Fig. 10 indicates that writes begin completing at approximately the 35ms mark. A pessimistic prediction was followed and polling began at 65ms.

TABLE VII. SWISSBIT MICRO-SD CARD ENERGY CONSUMPTION

State	Static (µJ)	IODVS (µJ)	Delta
Idle	66.25	43.53	-34.30%
Write	25.01	25.72	2.85%
Wait	3726.20	2839.78	-23.79%
Verify	36.31	31.68	-12.74%
Test Total	**3853.76**	**2940.71**	**-23.69%**

4) Kingston SDHC 2.0GB Micro-SD Memory Card

The Kingston Micro-SD Card was manufactured in 2014. Initial experiments with the device indicated that writes completed nearly 20x faster than the models previously tested. Furthermore, the maximum wait state duration was slightly over 1ms with a very high current consumption throughout the state. The test used a 2us sample time and all writes were completed within 4ms after issue.

TABLE VIII. KINGSTON MICRO-SD CARD ENERGY CONSUMPTION

State	Static (µJ)	IODVS (µJ)	Delta
Idle	24.63	16.89	-31.40%
Write	89.74	91.45	1.90%
Wait	122.44	97.39	-20.46%
Verify	54.00	57.53	6.53%
Test Total	**290.81**	**263.26**	**-9.47%**

D. Honeywell HIH6130 Temperature / Humidity Sensor

The MCU communicates with the temperature and humidity sensor [27] via I^2C. The interface communicates in an open-drain fashion and therefore logic-high levels are accomplished simply by changing the MCU pin direction from output-low to input. The I^2C bus was pulled to match the voltage level of the domain and therefore, when the MCU is sending data to the peripheral, it is not necessary to match the voltage of the MCU and peripheral domain. However, when the MCU is retrieving data from the peripheral, the voltages must be matched in order to ensure that input logic-level requirements are satisfied on the MCU.

The primary benefit of IODVS in this case is that the rate of I^2C communication is highly dependent on the magnitude of the pull-up resistors enabling it and the signaling voltage. By allowing the voltage to increase to 3.3V during the read, larger pull-up resistors can be used, thus decreasing static power dissipation at the same communication frequency.

Because the device operates using open-collector signaling, the PPP is slightly different. Again, the control PPP is 3.3V in all states, but the IODVS PPP is 2.5VIRyTW (idle / ready / transmitting / waiting). Transmitting is denoted as seen from the MCU perspective. So the profile effectively mandates that only when the MCU is receiving data from the peripheral should it raise the device voltage to an MCU compatible level.

This peripheral automatically enters an internal sleep mode described in its data sheet which drops the current consumption when a measurement has completed but has not yet been read. IODVS functions separately and provides additional energy savings. The first state has slightly higher energy power consumption because the device does not have a known measurement available.

TABLE IX. HIH-6130 ENERGY CONSUMPTION

State	Static (µJ)	IODVS (µJ)	Delta
Idle	10.28	6.28	-38.87%
Command	1.68	1.05	-37.60%
Waiting	399.07	245.89	-38.38%
Reading	4.30	4.42	2.62%
Test Total	**415.33**	**257.64**	**-37.97%**

VI. Conclusions and Future Work

IODVS has been shown to decrease energy consumption on a typical group of external peripherals by 10-40% with no decreases in either performance or accuracy. The efficacy of the technique tends to increase with low-duty cycles which is typical of external non-volatile memory. The overhead of performing IODVS is negligible via peripheral power profiles.

Minimal additional circuitry is required to implement IODVS. In many cases the decrease in power budget or increase in performance may offset the additional cost. Our experiment used a DAC because of the flexibility it offered in voltage modulation for a wide variety of devices. Simpler implementations would benefit from switching SMPS feedback resistors into and out of the circuit.

The technique would be most effective if it were used in a system with minimal domain capacitance. This would allow for faster changes in domain voltage which would reduce the response time of the SMPS and reduce the inrush current when charging a domain. Ideal domain capacitance balances IODVS dynamics against varying peripheral loads.

All of the devices tested provide a mechanism for testing operation-complete. We used the timing specifications contained within the device datasheet for all devices except the Micro-SD Cards (where polling was mandatory). Based on the current profiles of the devices, it can be inferred that most operations completed earlier than the datasheets specified. It would be worthwhile to pursue further research combining operation-completion prediction heuristics with IODVS.

Parallel NAND and NOR flash devices have similar timing requirements to the devices we tested. We will continue this research by exploring the benefits of IODVS on those devices as well as expanding the technique to include multiple peripheral voltage domains.

VII. References

[1] 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.

[2] 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.

[3] Andrew Kahng, Rakesh Kumar, Seokhyeong Kang, and John Sartori, "Enhancing the Efficiency of Energy-Constrained DVFS Designs," *IEEE Transactions on VLSI Systems*, vol. 21, no. 10, pp. 1769 - 1782, 2013.

[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] Sungjin Lee and Jihong Kim, "Using Dynamic Voltage Scaling for Energy-Efficient Flash-based," in *ISOCC*, Incheon, 2010.

[11] 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.

[12] 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.

[13] Wei-Chung Chen et al., "A Switchable Digital–Analog Low-Dropout Regulator for Analog Dynamic Voltage Scaling Technique," *IEEE Journal of Solid State Circuits*, vol. 49, no. 3, pp. 740-750, 2014.

[14] 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.

[15] L.B. Hormann, P.M. Glatz, C. Steger, and R. Weiss, "Energy Efficient Supply of WSN Nodes using Component-Aware Dynamic Voltage Scaling," in *European Wireless*, Vienna, 2011.

[16] Ulf Kulau, Felix Busching, and Lars Wolf, "A Node's Life: Increasing WSN Lifetime by Dynamic Voltage Scaling," in *IEEE International Conference on Distributed Computing in Sensor Systems*, Cambridge, 2013.

[17] Waltenegus Dargie, "Dynamic Power Management in Wireless Sensor Networks: State-of-the-Art," *IEEE Sensors Journal*, vol. 12, no. 5, pp. 1518 - 1528, 2012.

[18] Texas Instruments. (2004, June) Selecting the Right Level-Translation Solution. [Online]. http://www.ti.com/lit/an/scea035a/scea035a.pdf

[19] Texas Instruments Incorporated. (2007, September) Texas Instruments Website. [Online]. http://www.ti.com/product/tps62240

[20] Maxim Integrated Solutions. (2012, Oct.) Maxim Integrated Solution Website. [Online]. http://datasheets.maximintegrated.com/en/ds/MAX4376-MAX4378.pdf

[21] Microchip Technology Inc. (2010, May) Microchip 25AA512 Datasheet. [Online]. http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en530926

[22] Micron Technology Inc. (2012) M25PX16 Datasheet. [Online]. http://www.micron.com/parts/nor-flash/serial-nor-flash/m25px16-VMN6P

[23] SD Association. (2013, January) SD Association Web Site. [Online]. https://www.sdcard.org/downloads/pls/simplified_specs/part1_410.pdf

[24] NXP Semiconductors. (2013, April) NXP Semiconductors Web Site. [Online]. http://www.nxp.com/documents/application_note/AN10911.pdf

[25] SuperTalent Corporation. (2012, April) SuperTalent Corporation Website. [Online]. http://www.supertalent.com/datasheets/5_112.pdf

[26] Honeywell International Inc. (2013) Honeywell Sensing and Control. [Online]. http://sensing.honeywell.com/product-page?pr_id=142040