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 peripherals are often specified with a range of performance characteristics that are affected by their supply voltage. Typically the peripheral supply voltage is static and therefore both energy consumption and performance traits are known at design-time. With Intra-Operation Dynamic Voltage Scaling (IODVS), we focus on reducing the energy consumption of peripheral devices by dynamically modulating supply voltage as they perform user-specified operations. IODVS is designed to have minimal impact on CPU utilization through the use of peripheral power profiles (PPP) that designate 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 they demand in order to reduce overall energy-delay product. Likewise, during low-performance states such as mandatory delays, the system can decrease domain voltage thus reducing energy consumption without affecting performance or correctness. We demonstrate this method on various common peripherals and have found total energy savings of up to 40%.

Keywords—Embedded Systems; Dynamic Voltage Scaling; Dynamic Power Management, low power, energy-aware design

# Introduction

Embedded systems are often limited by energy constraints. We seek to increase energy efficiency by limiting power demand through the use of fine grained dynamic voltage scaling.

Consider 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 scenario 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 timeframes where low-performance is allowed such as mandatory wait periods and where the application of traditional DPM techniques would adversely affect operation of either the device or the system.

SPI EEPROM is a typical peripheral device that is used to provide non-volatile data storage. They are typically 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 in order to achieve compatibility with systems using voltages from 1.8V to 5.0V. A write operation to the device (with an optional write-verification stage) has the specifications regarding state transitions and timings shown in Fig. 1:

|  |
| --- |
| 1. A SPI EEPROM Write / Verify Cycle |

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 while minimizing energy delay product (EDP). The maximum benefit of IODVS can be realized during the longest portion of the transaction described in Fig. 1: the delay. IODVS decreases the supply voltage to the chip during this period and we find that the energy cost of the transaction is 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.

1. 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) |

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.

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 were performed 2048 times with random parameters 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%.

# Related Work

Dynamic Power Management (DPM) and Dynamic Voltage Scaling (DVS) implementations seek to minimize energy consumption in an embedded systems. DPM policies tend to focus on strict power-state relationships [2] while DVS policies tend to incorporate a linear power-performance relationship [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 CFG of a task to determine when a peripheral is likely to be accessed [5]. Similar data can be realized online by profiling tasks and thereby 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 equality is commonly known as the breakeven time [9].

Some peripherals provide multiple power/performance states. As such, mode dependence graphs were developed in order to accurately quantify the breakeven time between states [10]. 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 [10].

IODVS is similar to the CADVS technique [12] in that an adjustable regulator is used to operate an embedded system at its minimum voltage requirements. It differs in that IODVS extends the technique into multiple voltage domains and operates at 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.

|  |
| --- |
| 1. Peripheral Domain SMPS, Control and Current Sense Circuitry |

# Assumptions

The MCU and peripheral voltage domains are decoupled and the peripheral domain is adjustable via 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 [13] for this purpose in most embedded systems. Generally, the lowest communication EDP occurs at matched voltage/frequencies. Thus we are left with intra-operation voltage modulation as our means of decreasing energy consumption.

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, not any one peripheral in particular.

The peripheral power profile 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 tested operated well below their specified minimum voltage requirements. While minimizing voltage and therefore energy consumption 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 a marginal factor of safety.

# Methods and Materials

The TPS62240 [1] adjustable (SMPS) was selected to power the peripheral domain due to its high efficiency at light loads, output capacity and adjustability. Peripheral domain voltage modulation is accomplished via DAC output on a 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 [2] on both the input to the SMPS and the output to the domain.

As shown in Fig. 2, 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 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 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. Special care was taken to guarantee that the pins connecting the MCU to peripheral devices were changed to input mode upon transitions into the Idle/Wait states. This ensured that the peripherals were not inadvertently being powered from port pins on the MCU.

The pairs of states to voltages forms a lookup table called the peripheral power profile. 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 within the specifications of the device datasheet, no operations failed.

|  |
| --- |
| **Algorithm 1** Generic IODVS via Device Driver |
| C ← Command received from Application to Device Driver  Sc ← Current state of the selected peripheral  Sn ← Next state of the selected peripheral  Vc ←Voltage corresponding to a specific device state  Vw ←Voltage corresponding to a specific device state  A ← Array of State, Voltage Pairs  SET\_PERIPHERAL\_VOLTAGE(Vc)  SEND\_COMMAND\_TO\_DEVICE(C)  if C requires WAIT\_STATE  SET\_PERIPHERAL\_VOLTAGE(Vw)  WAIT\_FOR\_COMMAND\_COMPLETE(Tw)  SET\_PERIPHERAL\_VOLTAGE(Vi) |

# Results

All test results were measured entirely in-system using the 3 12-bit simultaneously sampling ADC converters onboard the microcontroller. 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 sample rate of 1.024MHz). Upon an ADC trigger, the state of the peripheral is stored synchronously with the sample. Each test data set was retrieved by MATLAB 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 via the fundamental relationship shown in (1). The results were calculated offline via (2), where S is the state of the device, t is time spent in a particular state and Ts is the sampling period.

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

Separating the energy consumption by state is important because it allows us to consider the effect of duty cycle on IODVS. Each device has an idle state where the voltage applied to the device is the minimum allowed by specification. This state allows the device to transition into an active (typically communicating) state immediately while also allowing for a dramatic decrease in power consumption.

The benefits of IODVS can be separated into two groups. First, applying IODVS to the idle state results in decreased energy consumption while avoiding the increased response time of DPM. Second, applying IODVS to a sequence of operations results in decreased energy consumption without the performance impacts of DVFS. As such, we can separate energy consumption into two intervals as shown in (3).

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

Therefore the ratio of active time to idle time dictates the maximum energy savings that IODVS can achieve.

There are two effects that are immediately obvious. One immediately notices the effects of domain capacitance. The domain voltage changes at a rate corresponding to (4). This is most noticeable 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 peripherals to coast from the higher communicating voltage to the ideal voltage for the operation being performed.

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

Likewise, on low to high transitions, the output current of the power supply spikes in order to charge the domain as quickly as possible via (5). This is also beneficial to IODVS in that it allows for very fast transitions from the wait states to the communication states.

## Microchip MCP25AA512 EEPROM

IODVS uses peripheral power profiles (PPP) to correlate peripheral voltages with internal state. The PPP specified for the EEPROM under test is derived from the specifications of its datasheet [15]. 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. 4 provides a comparison of both the standard PPP and the 1.8VIW profiles enabled by IODVS.

The state transition diagram of Fig. 3 is known a-priori and is followed throughout the tests illustrated in Fig. 4. 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 in order to verify that it was committed properly.

The effects of IODVS are immediately noticeable during the Idle and Wait states. The energy consumption during these states decreased 66.7% and 48.7% respectively. Energy consumption during the Write state increased by 30%. This is primarily due to 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 clip the graph in Fig. 4, that 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 erase operation followed by a write operation.

The idle time of the test lasted 1ms out of a total test time of 9.475ms. Thus, 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 the 66.7%.

|  |
| --- |
| 1. EEPROM Write Procedure |
| 1. EEPROM Test Results |

1. EEPROM Energy Consumption

|  |  |  |  |
| --- | --- | --- | --- |
| State | Static (uJ) | IODVS (uJ) | 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%** |

1. Energy Consumption vs. Duty Cycle

|  |  |  |  |
| --- | --- | --- | --- |
| **Duty Cycle** | Static avg. (uJ) | IODVS avg. (uJ) | 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% |

## 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 [13] supports a minimum of 4KB (sub-sector) erase and a maximum of 256B (page) sequential writes. In order to perform a read-modify-write operation, the transition diagram shown in Fig. 5 is followed.

As with all of the devices under test, the 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 function. A random sub-sector of memory is read into cache and modified with the random data to be written. The sub-sector erase operation is executed and IODVS adjusts the peripheral voltage to the wait state (2.3V in this peripheral power profile).

Upon completion of the erase cycle, the modified cache data are written back to the flash module one page at a time, thus 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. 6. 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 and 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 pushes the average toward a limit of 48.66%.

It is noteworthy that this erase-write sequence is common to all flash memory and is thus IODVS is applicable to parallel NOR and NAND devices as well as the serial one tested here. In high performance parallel devices writes complete on the order of micro-seconds, but erase operations complete in the range of 10-100ms and are thus exploitable by IODVS.

|  |
| --- |
| 1. Serial Flash State Transition Diagram |
| 1. Serial Flash Read-Modify-Write Test Results |

1. Serial Flash Energy Consumption

|  |  |  |  |
| --- | --- | --- | --- |
| State | Static (uJ) | IODVS (uJ) | 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%** |

|  |
| --- |
| 1. A Prototypical SDCard Write Operation |

|  |
| --- |
| 1. Typical Micro-SD Card Operation |

## Micro-SD Memory Card Results

Micro-SD Cards follow a standard outlined by the SD Association [20]. The standard is a minimum set of electrical and communication specifications that must be met and some vendors exceed those specifications [17] [18] [19]. 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 accesses 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.

Fig. 7 shows how timing variations affected our testing. The top figure shows one write/verify operation with constant polling for write-complete. The bottom figure shows the average of 2048 operations with polling beginning at 165ms. The non-monotonicity of the device state from the 165ms to 180ms marks indicates that some portion of the writes completed (and advanced to the verify state) before 165ms had elapsed. Also, that the majority of writes completed and advanced around the 170ms mark. Most writes completed before polling began, 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.

The SD Card has the highest current consumption of the devices tested and therefore requires a bulk capacitor at the load in order to ensure sufficient supply at the device. The point at which domain capacitance is detrimental to IODVS is dependent on the demands of the loads. Larger loads allow the domain to transition to lower voltages faster, while larger capacitances cause the domain to transition more slowly.

|  |
| --- |
| 1. Sandisk Micro-SD Card Write Operation |

1. Sandisk Micro-SD Card Energy Consumption

|  |  |  |  |
| --- | --- | --- | --- |
| State | Static (uJ) | IODVS (uJ) | 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%** |

### 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, we find that all of the writes had already completed and were eligible to transition into 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%. The write and verify stages of the test were relatively unchanged, though this could be due to insufficient resolution. Based on previous tests with higher resolution, charging the domain took between 5-10uJ and therefore is negligible compared to the total decrease of 3893uJ.

|  |
| --- |
| 1. Lexar Micro-SD Card Write Operation |

1. Lexar Micro-SD Card Energy Consumption

|  |  |  |  |
| --- | --- | --- | --- |
| State | Static (uJ) | IODVS (uJ) | 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%** |

### Lexar SDSC 1.0GB Micro-SD Memory Card

The Lexar Micro-SD card had a higher average power draw and a different write-completion characteristic than the Sandisk Micro-SD Card. The majority of writes completed between 140-180ms after the test began. This result can also be inferred from the drop in current consumption beginning at the 140ms mark. Polling for the completion did not begin until 160ms after the test began.

Despite the higher current draw, the system still benefited from a decrease in wait state energy consumption by 4049uJ. 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.

|  |
| --- |
| 1. SwissBit Micro-SD Card Operation |

1. SwissBit Micro-SD Card Energy Consumption

|  |  |  |  |
| --- | --- | --- | --- |
| State | Static (uJ) | IODVS (uJ) | 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%** |

### 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 seen 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. 11 indicates that writes begin completing at approximately the 35ms mark.

|  |
| --- |
| 1. Kingston Micro-SD Card Operation |

1. Kingston Micro-SD Card Energy Consumption

|  |  |  |  |
| --- | --- | --- | --- |
| State | Static (uJ) | IODVS (uJ) | 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%** |

### 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 appeared to be slightly over 1ms with a very high current consumption throughout the state. The test proceeded with a 2us sample time.

The write operation appears as a staircase between the 4ms and 6ms mark indicating that the device was ready for the write to a random address immediately in most cases, but after a 1ms delay in others.

Despite the fast characteristics of the device, IODVS was able to decrease the idle energy consumption by 31.4% and the current consumption of the wait state by 20.46%. The device was idle for 4ms out of the total test time of 12ms, yielding a duty cycle of 67%. The energy costs of the write, wait and verify states are relatively close. If the duty cycle were increased to 100%, the energy decrease would converge on 7.45%.

## Honeywell HIH6130 Temperature / Humidity Sensor

The MCU communicates with the temperature and humidity sensor [18] via I2C. 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 I2C 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 the case of this peripheral is that the rate of I2C 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 while maintaining the same communication frequency.

The test begins in the Idle state as shown in Fig. 13 and the MCU issues a “Measure” command to the sensor. The peripheral takes up to 4ms to wake from sleep [22] and then transitions to the temperature measurement and humidity measurement states in sequence. There is a noticeable drop in current in Fig. 14 upon the completion of the measurement and the MCU begins to read the data soon afterward.

This peripheral automatically enters an internal sleep mode described in its data sheet which drops the current consumption when a measurement is not actively being converted. IODVS functions separately and provides additional energy savings. As with the previous tests, there is an immediate drop in idle power dissipation.

The first state has slightly higher energy power consumption because the device does not have a known measurement available. Therefore it sacrifices power savings in order to decrease response time. During this period of time, we observe the best savings that IODVS can bring to bear on the device by savings 38.9% before the measure command is issued.

|  |
| --- |
| 1. HIH-6130 Measurement Sequence Diagram |
| 1. HIH-6130 Temperature / Humidity Sensor Operation |

1. HIH-6130 Energy Consumption

|  |  |  |  |
| --- | --- | --- | --- |
| State | Static (uJ) | IODVS (uJ) | 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%** |

# Conclusions and Future Work

IODVS has been shown to decrease energy consumption on a typical group of peripherals by 30-40% with no decreases in either performance or accuracy. The efficacy of the technique is increased with low-duty cycles. The overhead of performing IODVS is minimal through the use of pre-defined peripheral power profiles. The additional circuitry required to implement IODVS is minimal and in many cases, the performance increase may justify the expense or the power budget decrease may offset the additional cost.

IODVS would be more effective if it were used in a system with zero domain capacitance. This would allow for instantaneous changes in domain voltage and reduce the inrush current when charging a domain. Obviously zero-capacitance is impractical, but minimizing capacitance while maintaining steady voltage to peripheral devices with varying loads would certainly increase the efficiency of IODVS.

The efficacy of IODVS is highly dependent on the type of SMPS used to control domain voltage. It was seen that the input current to the controlling SMPS increased dramatically when changing domain voltage. As such, for a brief period of time, the efficiency of the SMPS is very low. This could be addressed by slowing the rate of change in feedback voltage (and augmented with a predictor), or by using a different type of adjustable SMPS.

Manipulating the voltage across a domain of devices is bound to impact some devices more than others. For instance, if the domain voltage drops below 2.7V, the SD Card reverts from an initialized state to the idle state. Therefore, before adjusting a particular device on the domain, IODVS should determine if that would cause an overall benefit or detriment to devices on the domain. This could be determined by majority vote of devices on the domain. If indeed the voltage is manipulated out of bounds for a particular device, the driver for each device on the domain needs to be notified of the voltage change so that re-initialization can take place if necessary.

# 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. |
| [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. |
| [3] | H. 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. |
| [4] | C. 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. |
| [5] | 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. |
| [6] | 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. |
| [7] | Y.-S. Hwang, S.-K. Ku and K.-S. Chung, "A predictive dynamic power management technique for embedded mobile devices," in *Consumer Electronics, IEEE Transactions on*, 2010. |
| [8] | T.-Y. H. C.-H. T. J.-J. C. T.-W. K. 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. |
| [9] | D. Li, P. 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. |
| [10] | D. 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. |
| [11] | L. Hormann, P. 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*, 2011. |
| [12] | T. Instruments, "Selecting the Right Level-Translation Solution," June 2004. [Online]. Available: http://www.ti.com/lit/an/scea035a/scea035a.pdf. |
| [13] | Texas Instruments Incorporated, "Texas Instruments Website," September 2007. [Online]. Available: http://www.ti.com/product/tps62240. |
| [14] | Maxim Integrated Solutions, "Maxim Integrated Solution Website," 10 2012. [Online]. Available: http://datasheets.maximintegrated.com/en/ds/MAX4376-MAX4378.pdf. |
| [15] | Microchip Technology Inc., May 2010. [Online]. Available: http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en530926. |
| [16] | Micron Technology Inc., 2012. [Online]. Available: http://www.micron.com/parts/nor-flash/serial-nor-flash/m25px16-VMN6P. |
| [17] | N. Semiconductors, April 2013. [Online]. Available: http://www.nxp.com/documents/application\_note/AN10911.pdf. |
| [18] | S. Corporation, May 2011. [Online]. Available: http://www.farnell.com/datasheets/1633579.pdf. |
| [19] | S. Corporation, April 2012. [Online]. Available: http://www.supertalent.com/datasheets/5\_112.pdf. |
| [20] | S. Association, January 2013. [Online]. Available: https://www.sdcard.org/downloads/pls/simplified\_specs/part1\_410.pdf. |
| [21] | Honeywell International Inc., 2013. [Online]. Available: http://sensing.honeywell.com/product-page?pr\_id=142040. |
| [22] | Honeywell International Inc., "Technical Note: Honeywell Sensing and Control," 7 June 2012. [Online]. Available: http://sensing.honeywell.com/i2c%20comms%20humidicon%20tn\_009061-2-en\_final\_07jun12.pdf. |