Intra-Operation Dynamic Voltage Scaling

Maximizing Peripheral Performance and Minimizing Energy Consumption

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; DVS; DPM

# Introduction

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 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 which is used to provide non-volatile data storage. They are generally specified for use in systems that require a quick data access time and low capacity requirements. The backup of RAM to non-volatile memory in the event of a power failure is a typical application. A write operation to the device (with an optional write-verification stage) has the following specifications regarding state transitions and timings per the datasheet [1]:

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

Both 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 any peripheral that has a voltage/frequency dependence and a wait-state. Our investigation considered the peripherals listed in Table I. as a representative sample. The device descriptions 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 | Vmax: 3.6V  Vmin: 2.7V (Operating)  Vmin: 2.0V (Idle/Ready) |

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.

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

IODVS is thoroughly tested on each of the six sample peripherals by conducting 2049 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.

IODVS is noteworthy in that our tests have focused on

# 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] 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 shutdown 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 a task 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 equality is commonly known as the breakeven time [9].

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]. 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].

The CADVS technique [12] is similar to IODVS in that an adjustable regulator is used to operate an embedded system at its minimum voltage requirements. IODVS extends the technique into multiple voltage domains and operates at a much finer granularity. The technique is different in that we seek to decrease the energy cost of peripheral operations as they are performed, which contrasts with DPM in that DPM techniques inherently interfere with the operation of the device.

# 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 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.

In most cases, 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.

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 is specified to operate in the range of -40℃ to +80℃ and for a minimum 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 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. 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 creates a power profile per peripheral. Each test designates a power profile to use. Peripheral memory was tested with random data and across random memory addresses. Tests were run 2049 times and the results were averaged. If any test failed to complete the operation successfully then the test-set was considered to have 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)

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 1. 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 2. 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) correlate peripheral voltages with internal state. The PPP specified for the EEPROM under test is derived from the specifications of its datasheet [1].

According to the device datasheet, the EEPROM can communicate at 10MHz at 3.3V, while only 1.8V is required for basic operation. However, the length of the idle state is voltage independent and IODVS exploits this disassociation.

The standard PPP is considered a control group and indicates that all states (writing/waiting/verifying/etc) should have 3.3V applied to the peripheral. The 1.8VIW (1.8V Idle/Wait) profile indicates 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 test begins with the EEPROM powered up and in the idle state. Next, 2049 tests are ….

|  |
| --- |
| 1. EEPROM Write Procedure |
| test11-1   1. EEPROM Test Results |

1. EEPROM Energy Consumption

|  |  |  |  |
| --- | --- | --- | --- |
| State | Static (uJ) | IODVS (uJ) | Delta |
| Idle | 10.24 | 3.49 | -65.91% |
| Write | 13.79 | 17.85 | 29.47% |
| Wait | 64.59 | 33.71 | -47.80% |
| Verify | 17.58 | 17.40 | -1.02% |
| Idle | 8.81 | 7.96 | -9.57% |
| Total | **115.02** | **80.43** | **-30.07%** |

The effects of IODVS are immediately noticeable during the Idle and Wait states. The energy consumption during these states decreased 66% and 48% respectively. Energy consumption during the Write state appears to have 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 was analyzed accordingly.

## Numonyx M25PX16 Serial Flash

Serial flash modules [13] have a somewhat more complicated state transition diagram than EEPROM. Serial flash chips can only write zeroes to their memory locations. At a simplistic level, this requirement necessitates a complete erase of a subsector before modifying any memory within it. The M25PX16 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 wherein a 4KB sub-sector is erased and 16 page writes follow.

As the test begins, the chip is in the Idle state; it does not require an initialization routine to function. The specified sub-sector of memory is read into cache and modified with the data to be written while the sub-sector erase operation is ongoing. Note that cross-subsector writes were not evaluated because that would simply require two sequences to occur sequentially. Upon completion of the erase cycle, the modified data are written back to the flash module one page at a time. The writes cause a series of alternating “write-wait” states and the corresponding voltage/state changes are evident in Fig. 6.

Table III summarizes the energy saved through the use of IODVS. As expected, the most significant savings are found in the Idle and Wait states and some small increase is seen in the active states.

|  |
| --- |
| **Algorithm 2** Serial Flash Write with IODVS |
| Sc ← Current state of the selected peripheral  Sn ← Next state of the selected peripheral  Vs ←Voltage corresponding to a specific device state  A ← Array of State, Voltage Pairs  Sc ←DEVICE\_READING\_STATE  IODVS\_FIND\_STATE\_VOLTAGE(Vs)  for S in A  if S == Sn  Vs ←As  IODVS\_SET\_VOLTAGE(Vs)  SET\_PERIPHERAL\_VOLTAGE(DEVICE\_READING)  READ\_SERIALFLASH\_SUBSECTOR(Address)  MODIFY\_SERIALFLASH\_SUBSECTOR(Address, Data)  ERASE\_SERIALFLASH\_SUBSECTOR(Address)  SET\_PERIPHERAL\_VOLTAGE(DEVICE\_WAIT)  WAIT\_FOR\_FLASH\_ERASE\_COMPLETE()  for i = 0 to (FLASH\_SUBSECTOR\_SIZE / FLASH\_PAGE\_SIZE)  WRITE\_SERIALFLASH\_PAGE(Address, Data)  WAIT\_FOR\_PAGE\_WRITE\_COMPLETE()  SET\_PERIPHERAL\_VOLTAGE(DEVICE\_WAIT)  VERIFY\_SERIALFLASH\_WRITE(Address)  SET\_PERIPHERAL\_VOLTAGE(DEVICE\_IDLE) |

|  |
| --- |
| 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 | 10.73 | 5.63 | -47.58% |
| Read | 71.51 | 72.36 | 1.19% |
| Erase | 2.47 | 2.47 | -0.13% |
| Write\* | 89.48 | 99.16 | 10.82% |
| Wait\* | 574.84 | 365.75 | -36.37% |
| Verify | 83.78 | 52.77 | -37.02% |
| Idle | 38.92 | 38.20 | -1.85% |
| Total | 2666.18 | 1614.57 | -39.44% |

|  |
| --- |
| 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. [14] [15] [16] [17]

The SD Card protocol heavily relies on polling the peripheral device. As such, there is not much opportunity to apply IODVS like what was shown for the EEPROM and serial flash cases. However, it is very beneficial to apply IODVS to the SD Card after the device has completed its initialization routine.

An SD Card must be initialized before use, the test fixture communicates with the device via SPI and the initialization process takes approximately 200ms. Therefore, when the device is not in use, it can transition to the low-power 2.7V “Initialized” state, rather than undergoing a complete power-cycle as would be typical with DPM.

Thus, the benefits of IODVS with respect to the SD Card are highly dependent on duty cycle. It is shown that the initialized current consumption of the device is reduced by 30% while all other states have no change. Therefore, by using IODVS, the response time of the device is decreased when measured against DPM techniques and the power consumption of the devices is decreased when compared to the static-voltage case.

The SD Card has the highest current consumption of the devices tested and as such causes high droop in domain voltage when active. Decreasing this voltage sag while maintaining or decreasing domain capacitance would increase the efficacy of IODVS and this is further discussed in the future work section.

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

1. Sandisk Micro-SD Card Energy Consumption

|  |  |  |  |
| --- | --- | --- | --- |
| 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% |

### 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 160ms mark (which is approximately 150ms after the write command completed successfully).

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

The Lexar Micro-SD card had 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, whereas polling for the completion did not begin until 160ms after the test began.

After write-complete polling begins at 160ms into the test, we find that approximately half of the writes had already completed and were eligible to transition into the verification stage. On the other hand, half of the writes had not yet completed and polling continued until approximately the 180ms mark at which point all writes completed successfully.

Initiating the polling phase at 160ms into the test seems to be a good first-glance heuristic for energy minimization. It effectively balances the energy cost of keeping the device pending the verification stage against pre-maturely beginning polling the device for write-completion. Heuristic and address-based approaches to optimally scheduling this DPM-like functionality would be appropriate for further research.

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

1. Lexar Micro-SD Card Energy Consumption

|  |  |  |  |
| --- | --- | --- | --- |
| 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% |

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

The Swissbit Micro-SD Card is unique from the others 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 appears to be 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 write-completion time varies significantly more than the other cards. It can be inferred from the current consumption of the device during the test show in Fig. 11 that writes begin completing at approximately the 40ms mark. By the 60ms mark, most writes have completed, but polling does not begin until 65ms into the test. At this point, the vast majority of write operations have completed and can transition to the verification phase. There is a minority of write operations that complete after the 65ms mark that result in smoothing the shape of the state-graph in Fig. 11.

Particularly because of the caching functionality of the card, an address based heuristic to predict optimal polling time of the transition from the Wait 🡪 Verify state would be appropriate for further energy optimization of this device.

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.

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

1. SwissBit Micro-SD Card Energy Consumption

|  |  |  |  |
| --- | --- | --- | --- |
| 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% |

## 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. 12 and the MCU issues a “Measure” command to the sensor. The peripheral transitions to its Wait state where it is internally measuring temperature and humidity. There is a noticeable drop in current in Fig. 13 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 |

|  |  |  |  |
| --- | --- | --- | --- |
| State | Static (uJ) | IODVS (uJ) | Delta |
| Idle | 10.28 | 6.28 | -38.87% |
| Write | 1.68 | 1.05 | -37.60% |
| Wait | 399.07 | 245.89 | -38.38% |
| Read | 4.30 | 4.42 | 2.62% |
| Idle | 17.08 | 19.50 | 14.18% |
| Total | **432.41** | **277.14** | **-35.91%** |

# 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] | Microchip Technology Inc., May 2010. [Online]. Available: 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. |
| [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. |
| [4] | 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. |
| [5] | 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. |
| [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. |
| [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. |
| [8] | 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. |
| [9] | 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. |
| [10] | 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. |
| [11] | 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. |
| [12] | 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. |
| [13] | Micron Technology Inc., 2012. [Online]. Available: http://www.micron.com/parts/nor-flash/serial-nor-flash/m25px16-VMN6P. |
| [14] | N. Semiconductors, April 2013. [Online]. Available: http://www.nxp.com/documents/application\_note/AN10911.pdf. |
| [15] | S. Corporation, May 2011. [Online]. Available: http://www.farnell.com/datasheets/1633579.pdf. |
| [16] | S. Corporation, April 2012. [Online]. Available: http://www.supertalent.com/datasheets/5\_112.pdf. |
| [17] | S. Association, January 2013. [Online]. Available: https://www.sdcard.org/downloads/pls/simplified\_specs/part1\_410.pdf. |
| [18] | Honeywell International Inc., 2013. [Online]. Available: http://sensing.honeywell.com/product-page?pr\_id=142040. |

# Ease of Use

## Selecting a Template (Heading 2)

First, confirm that you have the correct template for your paper size. This template has been tailored for output on the US-letter paper size. If you are using A4-sized paper, please close this file and download the file “MSW\_A4\_format”.

## Maintaining the Integrity of the Specifications

Identify applicable sponsor/s here. If no sponsors, delete this text box (*sponsors*).

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

# Prepare Your Paper Before Styling

Before you begin to format your paper, first write and save the content as a separate text file. Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads-the template will do that for you.

Finally, complete content and organizational editing before formatting. Please take note of the following items when proofreading spelling and grammar:

## Abbreviations and Acronyms

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

## Units

* Use either SI (MKS) or CGS as primary units. (SI units are encouraged.) English units may be used as secondary units (in parentheses). An exception would be the use of English units as identifiers in trade, such as “3.5-inch disk drive”.
* Avoid combining SI and CGS units, such as current in amperes and magnetic field in oersteds. This often leads to confusion because equations do not balance dimensionally. If you must use mixed units, clearly state the units for each quantity that you use in an equation.
* Do not mix complete spellings and abbreviations of units: “Wb/m2” or “webers per square meter”, not “webers/m2”. Spell out units when they appear in text: “. . . a few henries”, not “. . . a few H”.
* Use a zero before decimal points: “0.25”, not “.25”. Use “cm3”, not “cc”. (*bullet list*)

## Equations

The equations are an exception to the prescribed specifications of this template. You will need to determine whether or not your equation should be typed using either the Times New Roman or the Symbol font (please no other font). To create multileveled equations, it may be necessary to treat the equation as a graphic and insert it into the text after your paper is styled.

Number equations consecutively. Equation numbers, within parentheses, are to position flush right, as in (1), using a right tab stop. To make your equations more compact, you may use the solidus ( / ), the exp function, or appropriate exponents. Italicize Roman symbols for quantities and variables, but not Greek symbols. Use a long dash rather than a hyphen for a minus sign. Punctuate equations with commas or periods when they are part of a sentence, as in

*a**b* 

Note that the equation is centered using a center tab stop. Be sure that the symbols in your equation have been defined before or immediately following the equation. Use “(1)”, not “Eq. (1)” or “equation (1)”, except at the beginning of a sentence: “Equation (1) is . . .”

## Some Common Mistakes

* The word “data” is plural, not singular.
* The subscript for the permeability of vacuum **0, and other common scientific constants, is zero with subscript formatting, not a lowercase letter “o”.
* In American English, commas, semi-/colons, periods, question and exclamation marks are located within quotation marks only when a complete thought or name is cited, such as a title or full quotation. When quotation marks are used, instead of a bold or italic typeface, to highlight a word or phrase, punctuation should appear outside of the quotation marks. A parenthetical phrase or statement at the end of a sentence is punctuated outside of the closing parenthesis (like this). (A parenthetical sentence is punctuated within the parentheses.)
* A graph within a graph is an “inset”, not an “insert”. The word alternatively is preferred to the word “alternately” (unless you really mean something that alternates).
* Do not use the word “essentially” to mean “approximately” or “effectively”.
* In your paper title, if the words “that uses” can accurately replace the word “using”, capitalize the “u”; if not, keep using lower-cased.
* Be aware of the different meanings of the homophones “affect” and “effect”, “complement” and “compliment”, “discreet” and “discrete”, “principal” and “principle”.
* Do not confuse “imply” and “infer”.
* The prefix “non” is not a word; it should be joined to the word it modifies, usually without a hyphen.
* There is no period after the “et” in the Latin abbreviation “et al.”.
* The abbreviation “i.e.” means “that is”, and the abbreviation “e.g.” means “for example”.

An excellent style manual for science writers is [7].

# Using the Template

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use the scroll down window on the left of the MS Word Formatting toolbar.

## Authors and Affiliations

The template is designed so that author affiliations are not repeated each time for multiple authors of the same affiliation. Please keep your affiliations as succinct as possible (for example, do not differentiate among departments of the same organization). This template was designed for two affiliations.

### For author/s of only one affiliation (Heading 3): To change the default, adjust the template as follows.

#### Selection (Heading 4): Highlight all author and affiliation lines.

#### Change number of columns: Select the Columns icon from the MS Word Standard toolbar and then select “1 Column” from the selection palette.

#### Deletion: Delete the author and affiliation lines for the second affiliation.

### For author/s of more than two affiliations: To change the default, adjust the template as follows.

#### Selection: Highlight all author and affiliation lines.

#### Change number of columns: Select the “Columns” icon from the MS Word Standard toolbar and then select “1 Column” from the selection palette.

#### Highlight author and affiliation lines of affiliation 1 and copy this selection.

#### Formatting: Insert one hard return immediately after the last character of the last affiliation line. Then paste down the copy of affiliation 1. Repeat as necessary for each additional affiliation.

#### Reassign number of columns: Place your cursor to the right of the last character of the last affiliation line of an even numbered affiliation (e.g., if there are five affiliations, place your cursor at end of fourth affiliation). Drag the cursor up to highlight all of the above author and affiliation lines. Go to Column icon and select “2 Columns”. If you have an odd number of affiliations, the final affiliation will be centered on the page; all previous will be in two columns.

## Identify the Headings

Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

Component heads identify the different components of your paper and are not topically subordinate to each other. Examples include Acknowledgments and References and, for these, the correct style to use is “Heading 5”. Use “figure caption” for your Figure captions, and “table head” for your table title. Run-in heads, such as “Abstract”, will require you to apply a style (in this case, italic) in addition to the style provided by the drop down menu to differentiate the head from the text.

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced. Styles named “Heading 1”, “Heading 2”, “Heading 3”, and “Heading 4” are prescribed.

## Figures and Tables

#### Positioning Figures and Tables: Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation “Fig. 1”, even at the beginning of a sentence.

1. Table Type Styles

| Table Head | Table Column Head | | |
| --- | --- | --- | --- |
| Table column subhead | Subhead | Subhead |
| copy | More table copya |  |  |

1. Sample of a Table footnote. (*Table footnote*)
2. Example of a figure caption. (*figure caption*)

Figure Labels: Use 8 point Times New Roman for Figure labels. Use words rather than symbols or abbreviations when writing Figure axis labels to avoid confusing the reader. As an example, write the quantity “Magnetization”, or “Magnetization, M”, not just “M”. If including units in the label, present them within parentheses. Do not label axes only with units. In the example, write “Magnetization (A/m)” or “Magnetization {A[m(1)]}”, not just “A/m”. Do not label axes with a ratio of quantities and units. For example, write “Temperature (K)”, not “Temperature/K”.

##### Acknowledgment *(Heading 5)*

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

##### References

The template will number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use “Ref. [3]” or “reference [3]” except at the beginning of a sentence: “Reference [3] was the first ...”

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors’ names; do not use “et al.”. Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished” [4]. Papers that have been accepted for publication should be cited as “in press” [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

1. G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*

We suggest that you use a text box to insert a graphic (which is ideally a 300 dpi TIFF or EPS file, with all fonts embedded) because, in an MSW document, this method is somewhat more stable than directly inserting a picture.

To have non-visible rules on your frame, use the MSWord “Format” pull-down menu, select Text Box > Colors and Lines to choose No Fill and No Line.

1. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
2. I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
3. K. Elissa, “Title of paper if known,” unpublished.
4. R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.
5. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
6. M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.