DSP-Based Control of Multi-Rail DC-DC Converter Systems with Non-Integer Switching Frequency Ratios

James Mooney, Simon Effler, Mark Halton, Abdulhussain E. Mahdi
Department of Electronic & Computer Engineering
University of Limerick
Limerick, Ireland
james.mooney@ul.ie

***Abstract*—This paper examines the use of non-integer switching frequency ratios in digitally controlled DC-DC converters. In particular the execution of multiple control algorithms using a Digital Signal Processor (DSP) for this application is analyzed. The variation in delay from when the Analog to Digital Converter (ADC) samples the output voltage to when the duty cycle is updated is identified as a critical factor to be considered when implementing the digital control system. Fixing the delay to its maximum value is found to produce reasonable performance using a conventional DSP. A modification of the DSP's interrupt control logic is proposed here that minimizes the delay and thereby yields improved performance compared with that given by a standard interrupt controller. Applying this technique to a multi-rail power supply system provides the designer with the flexibility to choose arbitrary switching frequencies for individual converters, thereby allowing optimization of the efficiency and performance of the individual converters.**

I. INTRODUCTION

Intermediate bus architectures are a feature of computing and telecommunications devices where the output of an isolated DC-DC converter forms an intermediate voltage bus that provides the input to multiple non-isolated Point-of-Load (POL) converters [1, 2]. The POL converters have different voltage levels as well as being applied to a wide range of load ICs with a variety of specifications, for example ASICs, DSPs, FPGAs and memory devices [3, 4]. The use of digital control is beneficial in such systems as a single digital controller can compensate multiple DC-DC converters as illustrated in Fig. 1. This is in contrast with analog systems which require individual compensators for each converter resulting in a potentially more expensive power supply that consumes more board area. Digital controllers can be categorized into two main architectures. The first uses fixed-algorithm dedicated computational hardware for each individual power converter [5], while the second uses a single DSP that can be time-multiplexed among the different converters [6]. DSPs are commonly used as digital controllers in multi-rail power converter systems. They provide flexibility in terms of the software-based control algorithms that can be implemented and they also incorporate communication and monitoring functionality.

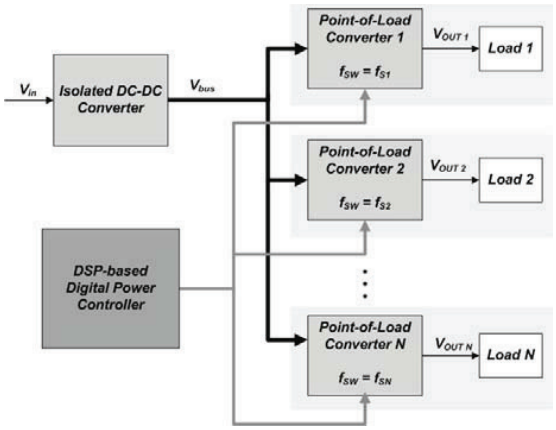

Figure 1. DSP-based controller applied to multiple DC-DC converters

In DSP-based digital power controllers an interrupt signal is used to trigger the execution of a control algorithm when a new ADC sample is available. When multiple power converters are being controlled the interrupt signals are interleaved so that each control loop has its own fixed time slot. This method is only practical when all algorithms are either executed at the same frequency or at different frequencies that are integer multiples of each other. If the switching frequencies have a non-integer ratio, variations occur in the delay between ADC-sampling and Digital Pulse Width Modulator (DPWM) duty-cycle-updating. A detailed analysis of this problem is presented in the next section.

Constraining the switching frequencies to integer multiples of each other can impact the efficiency or performance of the converters because the designer is forced into selecting non-optimal switching frequencies. If an arbitrary switching

frequency can be chosen, the task of meeting the specifications of individual POL converters each with unique efficiency and performance specifications is easier. In designing a buck converter, one of the first specifications to be decided upon is the inductor current ripple, ΔI_L. This is illustrated in the waveform of Fig. 2 for continuous conduction mode operation of a buck converter with switching period, T_S. A simplified representation for the current ripple in inductor L is given by:

$$\Delta I_L = \frac{V_O}{V_{IN}} \cdot \frac{V_{IN} - V_O}{L \cdot f_S}, \qquad (1)$$

where V_O is the output voltage, V_{IN} is the input voltage and f_S is the switching frequency. The value of ΔI_L is thus only dependent on the values of V_O, V_{IN}, L and f_S. The input and output voltages will be specified by the application, therefore the desired inductor current ripple will be determined by choice of L and f_S. If f_S is restricted to an integer multiple of another frequency, this also restricts the value of L that can be chosen. Having to choose a switching frequency that is larger than desired will impact the efficiency of the converter due to the relationship between switching frequency and efficiency [7]. Similarly having to choose an inductor that is larger than desired also affects the efficiency [8]. Removing the restriction of integer multiple switching frequencies thus allows more optimal values of f_S and also L to be chosen.

This paper proposes a solution that minimizes the DSP's variable delay and the associated problematic effects, thereby enabling the benefits of non-integer switching frequency ratios to be obtained in multi-rail applications.

II. Effects of Non-Integer Switching Frequency Ratio

In a typical power control DSP the execution of control algorithms cannot be interrupted. If an interrupt occurs when a control algorithm is already being executed, the interrupt is not serviced until the execution of the algorithm has completed. This results in a delay in the calculation and updating of the duty cycle in the pending Interrupt Service Routine (ISR). When multiple interrupt signals occur simultaneously the algorithms are typically executed according to a pre-defined priority. Thus a delay is introduced between ADC-sampling and duty-cycle-updating for the lower priority algorithm. Consequently the delay between ADC-sampling and duty-cycle-updating can vary each time an interrupt is triggered, depending on whether or not multiple interrupts have occurred simultaneously or an algorithm is already being executed. If the duty cycle has not been calculated by the beginning of the switching cycle, the DPWM will apply the duty cycle from the previous cycle. Additionally if a load transient occurs around this time, the delay in updating the duty cycle will result in a much slower response in the output voltage. The converter could also become unstable if the delay occurs for a number of consecutive cycles.

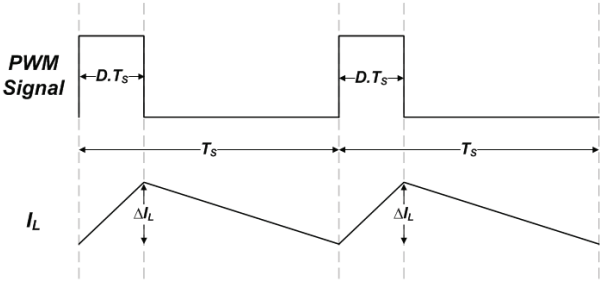

Figure 2. PWM and inductor current waveforms for buck converter

In order to avoid the problems associated with a variable delay, the delay can be fixed at its maximum possible value for each iteration of each algorithm. This is achieved by setting the sampling instant of the ADC at a fixed offset from the beginning of the next switching period. Thus when only one interrupt occurs there will be an 'idle' interval between when the duty cycle is calculated and the beginning of the next switching cycle. Conversely when the maximum number of interrupts occurs simultaneously the duty cycle will be calculated just in time for the beginning of the next switching cycle. The execution of the algorithm thus jitters within a permitted time interval. A problem with using the maximum fixed delay is that it is excessive and therefore degrades the performance of the voltage regulator due to a slower response to load transients. Improved performance can be obtained by reducing this delay [9].

The maximum ADC-sample to duty-cycle-update delay, T_{DMAX} is derived from an analysis of how the control algorithm is executed. In order to calculate the duty cycle as fast as possible 'pre-calculation' operations are performed prior to receiving the ADC sample, so that fewer instructions need to be executed before updating the DPWM. For example in the case of a PID compensator, after the most recent voltage error sample has been obtained it only needs to be multiplied by one coefficient and added to the pre-calculated sum of the other terms in the duty cycle equation, before it can be applied to the DPWM. After this the pre-calculations can be completed in advance of the next iteration. Fig. 3 shows the maximum ADC-sample to duty-cycle-update delays in the situation where three interrupt signals coincide. T_{DMAX} is given by:

$$T_{DMAX} = T_{ADC} + \sum_{i=0}^{HP} \left(T_{DC_i} + T_{PC_i} \right), \qquad (2)$$

where T_{ADC} is the ADC delay, HP is the number of algorithms with higher priority than this algorithm, T_{DC} is the duty cycle calculation time and T_{PC} is the pre-calculation time.

III. Modified Interrupt Method

In order to avoid the effects of variable DSP delays by fixing the delay at its maximum, a modified interrupt controller for DSPs is proposed that reduces T_{DMAX} to an acceptable value. Fig. 4 illustrates the resulting delays if all duty cycle calculations for coinciding interrupts are executed before any pre-calculations for the next iteration are carried out.

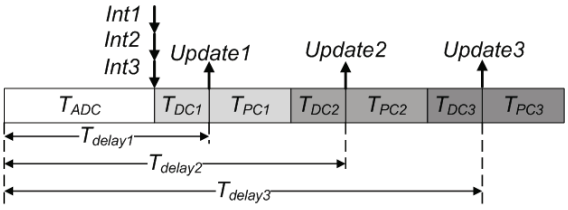

Figure 3. DSP delay with standard interrupt control

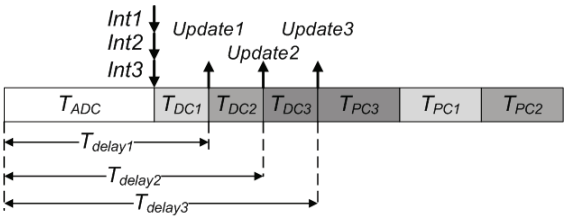

Figure 4. DSP delay with proposed interrupt control

By postponing the pre-calculations until all duty-cycle-updating has been completed, the total ADC-sample to duty-cycle- update delay as given in (2) is reduced. The value of the reduced delay may be obtained from the following equation:

$$T^{*}_{DMAX} = T_{ADC} + \sum_{i=0}^{HP}\left(T_{DC_i}\right). \qquad (2)$$

A. Modified Interrupt Controller Behavior

The modified interrupt controller achieves the behavior illustrated in Fig. 4 by automatically re-enabling all interrupts after the control algorithm has passed a certain stage of execution. This allows interruption of one control algorithm by another control algorithm, i.e. interrupt nesting, during the pre-calculation stage, after the duty cycle has been calculated and the DPWM has been updated.

Each control algorithm has a different interrupt priority level. When multiple interrupts occur the highest priority algorithm is selected first. At this point all other interrupts are disabled, a dedicated counter is loaded with a pre-configured duty cycle calculation time and counting is enabled. Interrupts are re-enabled after the counter determines that the duty cycle calculation time has elapsed. This duration is configurable for each of the algorithms in order to provide the flexibility to execute a different algorithm for each individual power converter. Before the pre-calculations for the highest priority interrupt can begin, execution is interrupted by the next highest priority algorithm. Again all other interrupts are disabled, the counter is reloaded and counting is enabled. The same applies for the next priority interrupt and so on. After no further interrupts are pending, the DSP continues with the execution of the pre-calculations for each of the interrupted algorithms in preparation for their next iterations.

The improved interrupt scheme can be implemented by augmenting a conventional DSP's interrupt controller with minimal additional hardware. The main requirement is the counter to determine when to re-enable interrupts. Some extra counter registers are also required to store the interrupt return addresses and the duty-cycle calculation times for each algorithm in terms of the number of instructions required.

B. Comparison with Existing DSPs

The technique described above is not directly implemented by existing DSPs that are used in power converter control applications [10-12]. Although the interrupts could be manually re-enabled after the duty cycle has been calculated, this requires an additional instruction at the beginning and end of each duty cycle calculation, which adds to the ADC-sample to duty-cycle-update delay. By doing this automatically the proposed scheme frees up more execution time in each switching cycle which could be used to execute instructions of a more complex control algorithm or to perform additional monitoring operations. Some DSPs also only facilitate interrupt nesting where higher priority interrupts are allowed to interrupt the currently executing Interrupt Service Routine, which does not comply with the requirement for each algorithm to be interruptible after a certain number of instructions have been executed. The priority of the interrupts in the proposed method relates to the order in which they will be processed if they occur simultaneously and bears no influence on whether or not they can interrupt other ISRs.

An alternative method that can be implemented using most commercial DSPs is to have separate interrupts for the duty cycle calculation and pre-calculation section of each control algorithm. After the duty cycle calculation has been completed it triggers a lower priority software interrupt for the pre-calculation. This allows other pending duty cycle calculation interrupts to be processed before any pre-calculations take place. The drawback of this method is that a "return from interrupt" instruction must be executed after each duty cycle calculation to exit from the ISR before processing the next pending interrupt. This adds a significant number of additional clock cycles between the duty cycle calculations for each algorithm and thus increases T_{DMAX}. The proposed method avoids this problem by keeping the duty cycle calculation and pre-calculation in the same ISR and only exiting from that routine if another interrupt is pending.

IV. Experimental Verification

In order to evaluate the performance improvement provided by the proposed interrupt controller, it has been compared with the standard interrupt control method, which is illustrated in Fig. 3. The comparison is based on the application of both interrupt control methods to a power converter system consisting of three 12 V - to - 1.5 V buck converters. Rail 0 and Rail 2 operate at a switching frequency of 500 kHz while Rail 1 operates at a switching frequency of 495 kHz, thus the ratio of the Rail 1 switching frequency to the Rail 0 switching frequency is 0.99. The control algorithm for Rail 0 has the highest interrupt priority, followed by Rail 1 and then Rail 2. Each calculation cycle is approximately 30 ns,

corresponding to a 33 MHz clock frequency. The proposed interrupt controller was incorporated into a custom dual-datapath Digital Signal Processor design for power control applications [13] using the Verilog hardware description language. The DSP was implemented on an Altera Cyclone II FPGA device together with the required digital pulse-width modulators. A third order linear compensator was programmed for execution on the dual-datapath DSP to regulate the output voltage of each of the buck converters. The compensator algorithm consisted of six duty-cycle operations and six pre-calculation operations.

Table I summarizes the T_{DMAX} delays for each of the voltage rails for the standard and proposed interrupt methods. It also includes the percentage reduction in delay provided by the proposed interrupt scheme. Although the modified interrupt method does not provide any reduction in T_{DMAX} for the highest priority interrupt i.e. Rail 1, it significantly reduces T_{DMAX} for the lower priority interrupts. It should be noted that the delay for Rail 1 corresponds to the delay that would occur for each of the rails in a system with an integer switching frequency ratio.

The operation of the standard interrupt controller and the modified interrupt controller is illustrated in Fig. 5 and Fig. 6 respectively. The ISR signal indicates when control algorithms are running for Rails 0, 1 and 2, while '3' indicates when the DSP is in background mode during the idle time between the ISRs. For standard interrupt control all ISRs run without interruption, as shown in Fig. 5. During the time interval shown interrupt signals Int0 and Int1 occur close together. In the first instance Int0 occurs before Int1 while in the second instance Int1 occurs just before Int0. ISR0 is therefore not executed until after ISR1 has completed. Fig. 6 illustrates the contrasting operation of the modified interrupt controller. In this case ISR0 is interrupted by ISR1 after the duty cycle has been calculated. After ISR1 has finished, execution returns to ISR0 to complete the pre-calculations for the next iteration. Similarly when Int1 occurs next, ISR1 is interrupted by ISR0 after the duty cycle has been calculated. ISR2 runs without interruption. The resulting difference in performance between the standard and modified interrupt controllers is demonstrated in Fig. 7 and Fig. 8.

The restriction of the long T_{DMAX} delay for the standard method results in a slow response to a load current step as illustrated in Fig. 7. The modified interrupt method with the shorter T_{DMAX} delay provides better performance and thus also facilitates the use of a wider bandwidth compensator. This enables a faster response to the same load step to be obtained as Fig. 8 shows.

TABLE I. MAXIMUM ADC-SAMPLE TO DUTY-CYCLE-UPDATE DELAYS FOR THIRD ORDER COMPENSATOR

Interrupt Method	Rail 0	Rail 1	Rail 2
T_{DMAX} for Standard Interrupt Control	360 ns	720 ns	1080 ns
T^{*}_{DMAX} for Modified Interrupt Control	360 ns	540 ns	720 ns
Reduction in T_{DMAX} (%Reduction in T_{DMAX})	0 ns (0%)	180 ns (25 %)	360 ns (33 %)

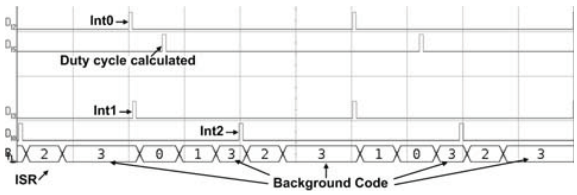

Figure 5. Standard interrupt control operation

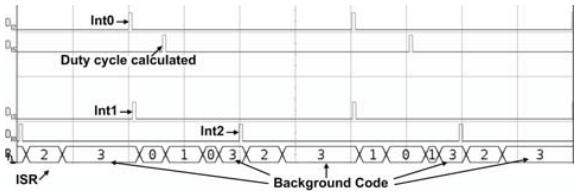

Figure 6. Modified interrupt control operation

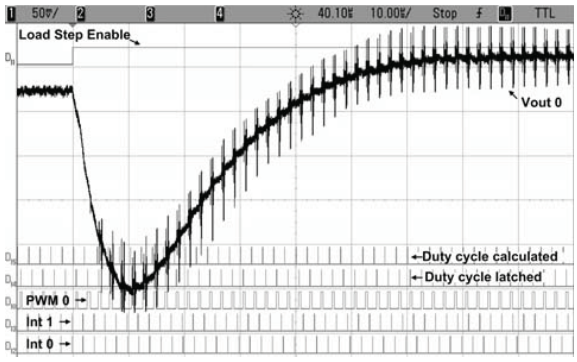

Figure 7. Load transient response for standard interrupt controller

Figure 8. Load transient response for modified interrupt controller with wider bandwidth compensator.

V. CONCLUSIONS

A drawback of using a standard DSP to control multiple power converters is its limitation in dealing with switching frequencies with non-integer ratios. A modified interrupt controller for digital signal processors has been proposed that performs significantly better in such applications. In comparison with the varying or excessive ADC-sample to duty-cycle-update delay in existing DSPs, the proposed interrupt method yields a constant, reduced and hence more desirable delay. The modified DSP thus provides the designer with flexibility in choosing the switching frequency and also the converters' component values. This means that the efficiency and performance of the individual power converters can be improved. The modified DSP also provides a practical method to control multiple power converters with a non-integer switching frequency ratio as an alternative to the more expensive area-intensive option of applying individual digital compensators to each power converter.

REFERENCES

[1] R. Miftakhutdinov, "Power distribution architecture for tele- and data communication system based on new generation intermediate bus converter," in International Telecommunications Energy Conference, 2008, pp. 1-8.

[2] H. Hong, "Coordination of design issues in the intermediate bus architecture," in Applied Power Electronics Conference, 2005, pp. 169-175 Vol. 1.

[3] National Semiconductor. (2010, March). Analog for Altera FPGAs Solutions Guide [Online]. Available: http://www.national.com/appinfo/power/files/NationalAlteraDesignGuide.pdf

[4] Texas Instruments. (2010, Jan.). Power Management Guide [Online]. Available: http://focus.ti.com/lit/sg/slvt145j/slvt145j.pdf

[5] Texas Instruments Data Sheet. (2008, Nov.). UCD9240 Digital PWM System Controller [Online]. Available: http://focus.ti.com/lit/ds/symlink/ucd9240.pdf

[6] J. Mooney, A. E. Mahdi, A. Kelly, and K. Rinne, "DSP-based controller for multi-output/multi-phase high switching frequency DC-DC converters," in IEEE Workshop on Control and Modeling for Power Electronics, 2008, pp. 1-6.

[7] X. Zhang and D. Maksimovic, "Multimode Digital Controller for Synchronous Buck Converters Operating Over Wide Ranges of Input Voltages and Load Currents," IEEE Transactions on Power Electronics, vol. 25, pp. 1958-1965, 2010.

[8] J. A. Abu-Qahouq, O. Abdel-Rahman, L. Huang, and I. Batarseh, "On Load Adaptive Control of Voltage Regulators for Power Managed Loads: Control Schemes to Improve Converter Efficiency and Performance," IEEE Transactions on Power Electronics, vol. 22, pp. 1806-1819, 2007.

[9] L. Corradini, S. Saggini, and P. Mattavelli, "Analysis of a high-bandwidth event-based digital controller for DC-DC converters," in IEEE Power Electronics Specialists Conference, 2008, pp. 4578-4584.

[10] Texas Instruments Reference Guide. (2009, Jan.). TMS320C28x CPU and Instruction Set Reference Guide [Online]. Available: http://focus.ti.com/lit/ug/spru430e/spru430e.pdf

[11] Analog Devices Reference Guide. (2003, Jan.). ADSP-2199x Mixed Signal DSP Controller Hardware Reference [Online]. Available: http://www.analog.com/static/imported-files/processor_manuals/50171261290224ADSP_2199x_Hardware_Reference.pdf

[12] Freescale Semiconductor Reference Guide. (2005, Nov.). DSP56800E Reference Manual [Online]. Available: http://cache.freescale.com/files/dsp/doc/ref_manual/DSP56800ERM.pdf

[13] J. Mooney, M. Halton, and A. E. Mahdi, "Specialized digital signal processor for control of multi-rail/multi-phase high switching frequency power converters," in IEEE Applied Power Electronics Conference, 2010, pp. 2207-2211.