A Communication Protocol with Data Compression for Isolated Digital Power Supplies

Zhiyuan Hu, *Student Member, IEEE*, Yan-Fei Liu *Fellow, IEEE*, Tet Hin Yeap, *Member, IEEE*, and Lusheng Ge

***Abstract*—High-cost isolators and their speed limits are the shortcomings of digital control in isolated power supplies. Different feedback schemes result in different numbers and speed requirements of the isolators, which impact the cost and the decision making. This paper compares existing solutions, and proposes a communication solution featuring data compression and time-division multiplexing, which provides the highest output accuracy and many advanced features of digital control, whereas the cost can potentially be in the range of conventional analog solutions.**

***Index Terms*—data communication, data compression, isolators, protocols, switched-mode power supplies**

I. Introduction

Digitally controlled front-end power converters are gaining popularity in recent years. Many leading power supply manufacturers have introduced "full digital" controlled front-end power modules—a phrase to indicate the use of digital control instead of digitally controlled analog control [1-5]. Power management IC manufacturers also offer a range of digital controllers for front-end conversion applications[6, 7]. In general, digital power supplies are able to provide higher performance and more flexibility at a lower cost [8, 9].

A front-end power module often consists of a non-isolated power factor correction (PFC) stage and an isolated DC-DC stage. There is also an auxiliary power stage with multiple windings, sometimes isolated, to power the primary- and secondary-side logic and MOSFET driver circuits. A microprocessor is often located on the secondary-side to implement power management functions such as fan speed control, monitoring temperature, monitoring several auxiliary power supplies, communicating with upper computers, etc.

In order to reduce safety hazards, isolation is needed between the AC line and the electrical load. Therefore many isolators are needed to transmit signals across the isolation barrier. First, in isolated DC-DC stages, signals must be transmitted across the isolation barrier to complete the closed-loop control. Second, the primary-side monitoring information, such as input under-voltage, over-voltage, over-current and brownout alarms, as well as PFC and fuse statuses, need to be communicated to the secondary-side microprocessor. Third, turn-on/off signals need to be sent from the secondary side to the primary side. Fourth, the feedback signals of the isolated auxiliary power supplies need to be transmitted across the isolation barrier as well. In the analog era, all of these signals require separate isolators, and take a fair amount board space.

In the digital era, digital signal processors (DSPs) are used as power controllers. Since DSPs can process data, a multiplexed digital communication scheme will integrate all the aforementioned signals into one data bus, thus save substantial board space, chip pin-outs, and potentially, cost. Advanced control techniques, e.g. reconfiguring parameters on-the-fly, are also made possible by such a communication bus. For above reasons, this paper is dedicated to a multiplexed data communication solution that is optimized for space, cost, and performance.

The following sections are organized as follows: Section II reviews existing digital feedback solutions; Section III proposes a new communication solution and justifies the rationale of the proposed solution; Section IV describes a Multi-Resolution Feedback method as a part of the proposed solution; Section V describes a Time-division Multiplexing 8B/10B protocol as a part of the proposed solution; Section VI demonstrates the experiment results of a prototype; Section VII discusses the considerations and variations of the proposed solution; Section VIII concludes the paper.

II. Comparison of Existing Digital Feedback Solutions

This section discusses the existing feedback solutions. The commonly used methods are (a) primary-side analog controller with analog isolators, (b) secondary-side analog controller with analog isolators, (c) secondary-side digital controller with analog isolators, (d) secondary-side digital controller with digital isolators, (e) primary-side digital controller with analog isolators, and (f) primary-side digital controller with digital isolators.

A. Primary-side Analog Controller with Analog Isolators

As shown in Fig. 1, when the analog controller is on the primary side, the secondary-side feedback signals are coupled to the primary side using opto-couplers. The analog opto-couplers are of low-cost, usually in the range of tens of cents (in USD). The peripheral resistors, capacitors, and error amplifiers (e.g. TL431) add slightly more cost. The overall cost

Zhiyuan Hu is with Queen's University, Kingston, ON K7L 3N6 Canada (e-mail: zhiyuan.hu@queensu.ca).

Yan-Fei Liu is with Queen's University, Kingston, ON K7L 3N6 Canada (e-mail: yanfei.liu@queensu.ca).

Tet Hin Yeap is with University of Ottawa, Ottawa, ON K1N 6N5 Canada (e-mail: tet@eecs.uottawa.ca).

Lusheng Ge is with Anhui University of Technology, Ma'anshan, Anhui 343002 China (e-mail: lsge@ahut.edu.cn).

This work was supported in part by the Natural Science Foundation of China under Grant 51277003

of this solution is low, but each signal requires a separate opto-coupler, which takes relatively large board space.

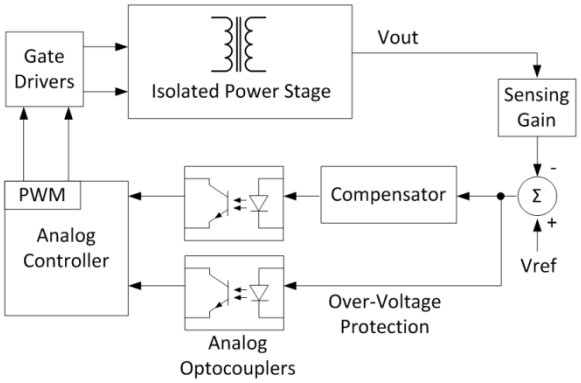


Fig. 1. Primary-side analog controller with analog isolators.

B. Secondary-side Analog Controller with Analog Isolators

As shown in Fig. 2, when the analog controller is on the secondary side, the MOSFET gate drivers are usually on the secondary side as well, and the gate driving power is transferred to the primary side using gate drive transformers. The gate drive transformers in this case are usually of large size due to the considerations of creepage and clearance in order to comply with safety standards, such as UL, VDE, CUL, IEC, and TUV [10]. Other low-voltage signals are still transmitted by opto-couplers. The overall cost and board space are higher than that in Solution A.

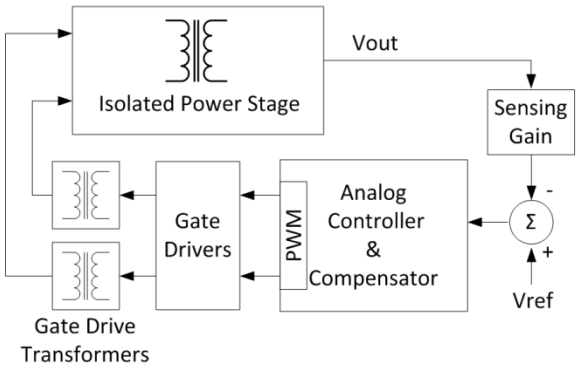


Fig. 2. Secondary-side analog controller with analog isolators.

C. Secondary-side Digital Controller with Analog Isolators

As shown in Fig. 3, when the digital controller is on the secondary side, the feedback loop is closed without crossing the isolation barrier. However high-voltage gate drive transformers are needed to transfer gate driving power to the primary side. An example is in [7]. The cost and the board space of such a solution are similar to that of Solution 0.

If transmitting other less-time-critical information is desired, digital isolators can be used to transmit data. An example is in [11]. Low-cost analog opto-couplers are not suitable for data transmission due to its low speed. Digital isolators that are in the range of 10Mbps typically cost more than analog opto-couplers by an order of magnitude.

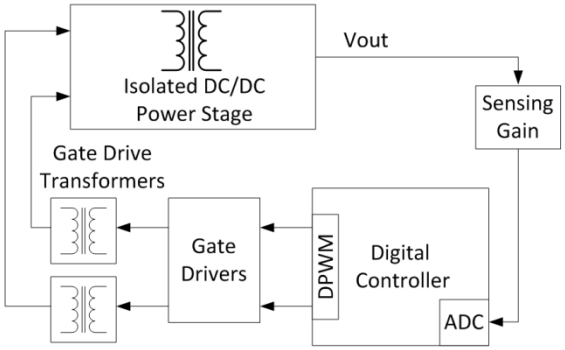


Fig. 3. Secondary-side digital controller with analog isolators.

D. Secondary-side Digital Controller with Digital Isolators

RF-type high-speed digital isolators can directly transmit PWM signals across the insulation barrier, and the gate drivers are placed on the primary side, thus the gate drive transformers can be eliminated to reduce the footprint area, as shown in Fig. 4. Examples are [12, 13]. The problem of this type of solutions is the ultra-high speed requirement for the digital isolator. Since the transmitted signal is the PWM signal, the isolator's speed must be as fast as the DPWM clock, which is the product of the switching frequency and the DPWM quantization levels. For example, if the ADC is 8-bit resolution, the DPWM must be at least 9-bit to avoid limit-cycle oscillation [14]; therefore if the switching frequency is 400*kHz*, the required speed of the digital isolator is:

$$400kHz \cdot 2^9 = 204.8MHz \tag{1}$$

Note that the high-speed isolators in the market are typically limited to 150*MHz* as of today (e.g. Si8440, Silicon Laboratories Inc., and ISO7221MDR, Texas Instruments Inc.), which are expensive and are still too slow for the abovementioned example. Although research has demonstrated a record of 200*MHz* [15], it can hardly accommodate the increasing speed requirement since the desired switching frequency and the ADC resolution are continuously increasing.

Because each PWM signal occupies a channel, four isolator channels are needed in full-bridge configurations. Other system information still needs additional digital isolators to transmit. Therefore, this solution is of the highest cost.

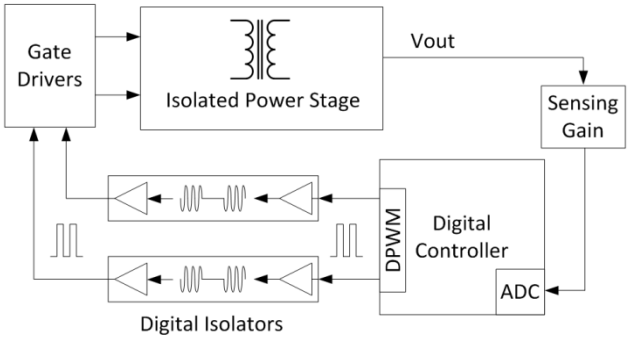


Fig. 4. Secondary-side digital controller with digital isolators.

E. Primary-side Digital Controller with Analog Isolators

When the digital controller and ADC are both on the primary side, the DPWM must not directly use ADC samples to configure duty cycle, otherwise the DPWM and the ADC will have the same amount of quantization levels, and limit-cycle oscillation will occur [14]. This requires the isolator to transmit uncompensated signal from secondary- to primary-side, and the compensation algorithm be performed inside the primary-side digital controller.

However, conventional analog opto-couplers cannot transmit uncompensated signal because of the low bandwidth and the nonlinearity of the current transfer ratio (CTR). In this scenario, linear opto-couplers can be used as analog isolators (e.g. HCNR200/201, Avago Technologies, and IL300, Vishay Semiconductors), as shown in Fig. 5. Also can be used is Isolated Feedback Generator (e.g. UC3901, Texas Instruments Inc.). With these types of isolators, the primary-side ADC can sample uncompensated output voltage signal.

The disadvantage of such solutions is that the cost of the linear opto-coupler and the Isolated Feedback Generator are equally as expensive as high-speed digital isolators. For linear opto-couplers, peripheral amplifier circuits are needed on both primary- and secondary-side [16].

The cost of such solutions is lower than that of Solution D because fewer isolators are needed, but still higher than that of Solution A.

It may be possible to transmit partially compensated error signal (only by an integrator) through a conventional analog opto-coupler, and then perform the rest of compensation algorithm inside the digital controller. But the feasibility of this configuration has not been proved, and significant research effort is required.

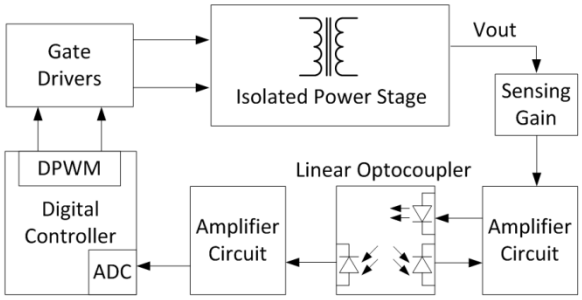


Fig. 5. Primary-side digital controller with analog isolators.

F. Primary-side Digital Controller with Digital Isolators

As it was discussed in Section I, in many front-end conversion applications there is a microprocessor on the secondary side for power management purposes. If this microprocessor equips with an ADC, the output voltage sampling can be performed on the secondary side with no extra cost.

The sampled data can be then sent to the primary side via digital isolators. The required speed of the digital isolator is determined by the switching frequency, the number of ADC bits, and the protocol overhead. This speed requirement is far lower than the direct PWM signal transmission in Solution D, thus lower-cost digital isolators can be used. Examples are in [17-19]. An example structure is shown in Fig. 6.

In [17], two digital opto-couplers are used for data line and clock line, respectively. A customized protocol is implemented. Due to the isolator's speed limit and the inefficient protocol, only 4 least significant bits (LSB) of an 8-bit ADC can be transmitted. The highest ADC accuracy is limited by such an arrangement because if the resolution is finer, the quantization levels provided by the 4 LSB are too few to cover the output voltage range.The effective data bits transmitted in a 16-bit frame is only 4.

In [18], Universal Asynchronous Receiver/ Transmitter (UART) standard is used to transmit feedback data. Only one digital isolator is used for this one-way communication. The disadvantages of UART include: (a) Due to the definition of the protocol, only 8 bits can be transmitted in each packet; thus only 8 bits of a 10-bit ADC can be transmitted; (b) For multiplexed use, or for transmitting data longer than 8 bits, address/sequence bits need to be included in the 8-bit packet, making the payload bits even fewer; (c) It is required that the clock frequency in UART is 16 times the baud rate and the clock mismatch must be less than 3.3% [20]; thus for a reasonable 10 *Mbps* data bus, the clock must be 160 *MHz* which is beyond the spec of most existing power management DSPs; (d) During high-speed operation, problems such as clock drift and send jitter may appear on UART [21].

In [19], two lower-speed digital isolators form a bi-directional data bus, which also facilitates synchronized sampling. The sampling instruction from the primary-side controller is forwarded to the secondary-side ADC as part of the protocol. A state-dependent ADC [22] divides the sampling range into one high-resolution region and two low-resolution regions. Each region has 16 quantization levels thus the ADC sample can be reduced to 4 bits and therefore reduce the speed requirement of the digital isolators.

In [23] and [24], a variable multi-resolution feedback (MRF) method uses 4-bit samples to represent the entire output-voltage regulation range without state-machine mechanism, nor changing the dynamic performance, and provides 10-bit accuracy at the output voltage point. A Time-division Multiplexing 8B/10B (TDM 8B/10B) protocol is highly efficient, low-cost, and can transmit time-critical commands with the highest priority. The integration of these two methods will be discussed in detail in this paper.

The cost of this type of solutions is among the lowest, because fewer isolators are required and the speed requirement is also low. This conclusion is based on the assumption that all the communication codecs are integrated into the DSPs' peripheral thus there is no extra cost. Nevertheless, the cost of the digital isolators is still higher than the analog Solution A.

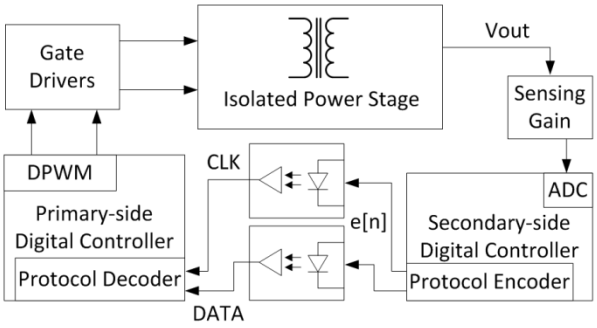


Fig. 6. Primary-side digital controller with digital isolators.

III. The Proposed Solution

The proposed solution includes two components: a data compression method called Multi-resolution Feedback (MRF) method and an optimized communication protocol called Time-division Multiplexing 8B/10B (TDM 8B/10B) protocol. It is an integration of the previously proposed methods in [23] and [24] by the authors of this paper. The proposed solution adopts the scheme in Solution F where the controller is placed on the primary side and the ADC is located on the secondary side. As it was discussed above, such a scheme has potentially the lowest cost among digital solutions. The only disadvantage of this scheme compared to the analog Solution A is still the costly digital isolators. In the proposed solution, this problem is solved by the line code-based TDM 8B/10B protocol, which enables the use of a low-cost pulse transformer as isolator.

Pulse transformer is a type of digital isolator which is widely used in telecom equipment to provide isolated signal transmission. The insulation strength can be designed to meet the double insulation requirement according to IEC60950-1 standard [25]. However, the driving signal of pulse transformers must be DC-balanced to prevent core saturation. A 10*MHz*-range pulse transformer costs roughly only tens of cents (in USD). As a result, the proposed solution will have similar cost to the analog solutions and feature all the advantages of digital control.

The proposed solution is illustrated in Fig. 7, and is described as follows. The secondary-side DSP takes ADC samples of multiple output voltages. The output voltages can be from multiple isolated power stages on the same board or multiple output windings of one power stage, or a mix of the two. The 10-bit ADC samples are compressed into 4 bits using a multi-resolution feedback (MRF) method. Every two 4-bit MRF codes are combined to form an 8-bit byte, which is in turn encoded by an 8B/10B-based protocol and sent to the primary side. The primary-side DSP firstly decodes the 8B/10B code into two 4-bit MRF codes, then decodes the two MRF codes into two 10-bit ADC samples, and then use the ADC samples for corresponding PID calculations. With MRF, the regulated output voltages remain 10-bit accuracy.

The proposed protocol provides the following functions and features: (a) it converts data into a DC-balanced form to enable the use of pulse transformer; (b) it is capable of interrupting the data transmission at any time to push through time-critical commands such as emergency shut-down; (c) it provides enough signal state changes to help clock recovery in order to eliminate a separate isolator for clock line; (d) it multiplexes many feedback loops and other data/commands in one data bus thus reduces the board space and cost; (e) it is highly flexible, configurable, and extendable by users; (f) it is able to detect corrupted data.

In order to justify the cost, the primary-side DSP can be integrated with the PFC controller; and the secondary-side DSP can also perform the tasks of power management functions. Therefore no extra DSP is introduced in the proposed solution.

The 8B/10B codec is not yet a built-in peripheral in any off-the-shelf DSP, however the 8B/10B codec IP cores have been available in many FPGA libraries [26-28]. Its gate count is only about 2200 gates (based on [28] and [29]), similar to that of a UART transceiver [20]. Therefore the cost of having such a peripheral in DSP should be marginal.

The following sections provide the details of the proposed MRF method and the proposed TDM 8B/10B protocol, respectively.

IV. Multi-resolution Feedback Method

When it comes to data communication, the first engineering decision is to determine what to be transmitted. Transmitting the compensated duty cycle value is not preferred because: (a) the resolution of DPWM must be finer than the resolution of ADC to avoid limit-cycle oscillation [14]; therefore DPWM must have more quantization levels than ADC, resulting in more digits to be transmitted per switching period; (b) a bit error in a most significant bit (MSB) may cause a catastrophic failure of the power supply. In comparison, transmitting ADC samples is more reliable against bit errors because the PID block is a low pass filter; and the data volume is smaller. Therefore transmitting ADC samples is preferred.

When transmitting ADC samples, data compression is an

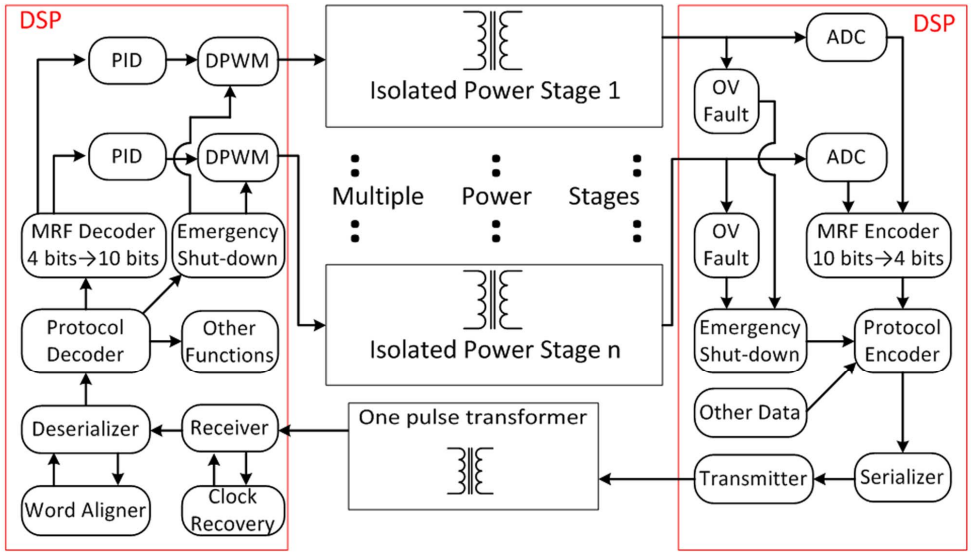


Fig. 7. The proposed solution consists of a Multi-resolution Feedback method (MRF) and a Time-division Multiplexing 8B/10B (TDM 8B/10B) protocol.

effective way to reduce the data volume in order to use lower-speed isolators and thus to reduce the cost.

Lossless data compression methods such as *entropy encoding* and *differential encoding* methods [30, 31] had been considered. *Entropy encoding* methods such as *Huffman encoding* [32] eliminate data redundancy by substituting repeated data patterns by short codes. However these methods only minimize data volume; instead, infrequent data patterns will be encoded into codes even longer than the original data, which is not acceptable for real-time communication. *Differential encoding* can be used in real-time applications; it transmits the incremental values instead of absolute sample values. However the transmission errors will be carried forward to the following samples; and correcting the errors requires excessive protocol overhead. Therefore *differential encoding* is not a robust method.

State-dependent ADC [22] is used in [19] to provide a data compression method to reduce 8-bit samples data into 4 bits: the ADC full scale range is divided into 3 regions; each region is represented by a 4-bit sample. Each region except the steady-state region must cover a wide scale range with only 16 quantization levels. This is an acceptable data compression approach because the sample values distant from the reference level do not require fine resolution. However, in many applications the resolutions may be too coarse and degrade dynamic performances. Also, the state change relies on a state machine, which triggers a state change when sample saturation occurs. This mechanism is susceptible to data transmission errors thus is less reliable.

This paper is derived from principles of information theory, that lossy data compression can be achieved by eliminating unnecessary information from the data. In digital power supply's sample data, unnecessary information can be identified by observing the following two facts:

First, the output voltage will stay in the output regulation range for almost all the time, which is defined in the design specification, usually less than ±5% of the output voltage. For a 10-bit ADC, it has 1024 quantization levels to cover the full scale range (FSR). Each quantization level represents 0.098% of the FSR, which is also the output accuracy of the power supply. If the output regulation range is ±5%, there are only 102 possible quantization levels instead of the full 1024 quantization levels.

Second, high-accuracy ADC samples are not necessary when the output voltage is distant from the reference voltage level. The high-accuracy samples only benefit the performance when the output voltage is near the reference voltage level and fine adjustment is needed to regulate the output voltage to the desired accuracy. However, unnecessary quantization levels must be carefully identified according to the properties of feedback loops, so that a minimum amount of data can provide uncompromised dynamic performance.

Taking advantage of the abovementioned observations, a multi-resolution feedback (MRF) method is proposed to compress ADC samples by removing the unnecessary information from the sample data.

A. Multi-Resolution Feedback Method Description

The multi-resolution feedback (MRF) method is a mapping technique that maps high-resolution ADC samples into a small number of quantization levels, with the finest quantization levels centered near the reference value, and the gradually coarser levels spread out over the output regulation range. The sample values outside the output regulation range are clamped by the upper and lower boundary values. All ADC samples are mapped into MRF levels before transmission, and each MRF level is represented by a MRF code. Because the number of MRF levels is much less than the number of ADC quantization levels, the MRF codes are much shorter than ADC samples, and thus data compression is achieved.

The MRF method uses a regular high-resolution ADC; therefore the secondary-side DSP "knows" the accurate output voltage, and thus it supports closed-loop soft start and over/under-voltage protections in which cases the ADC samples are outside the output regulation range.

A concept map of MRF levels is shown in Fig. 8. The horizontal axis is the sampled error value—the difference of the reference value and the ADC sample value. The vertical axis is the mapped value of the corresponding MRF level. The mapped values are the exact sampled error values at small errors, providing uncompromised output accuracy. At larger errors, the mapped values may deviate from the exact values due to the coarse resolution.

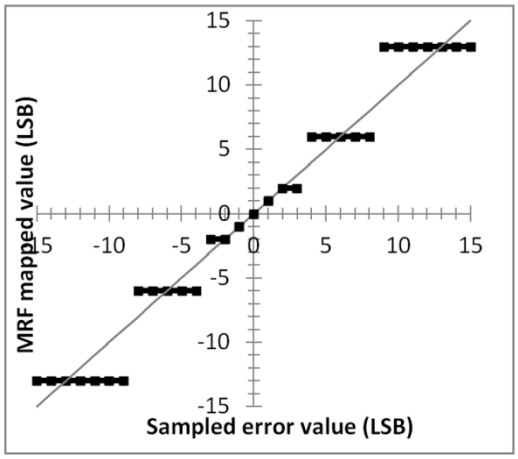


Fig. 8. Concept map of MRF levels. The horizontal axis is the sampled error value, and the vertical axis is the mapped value of the corresponding MRF level.

The error introduced by the MRF's coarse resolution can be considered as a round-off error in quantization effect due to the finite ADC resolution, which does not affect the system stability [33]. Its effect can be considered as a variable gain in the feedback loop, which is the ratio between the mapped value of the MRF level (V_{MRF}) and the exact sampled error value (V_{error}). The system open-loop transfer function can be written as:

$$\hat{H}(s) = M(V_{error}) \cdot H(s) \tag{2}$$

where $H(s)$ is the original open-loop transfer function without MRF, and $M(V_{error})$ is the variable gain which is defined below:

$$M(V_{error}) = \frac{V_{MRF}}{V_{error}} \tag{3}$$

In steady state, the output voltage is regulated at the reference voltage, where the mapped MRF value is the exact sampled error value, thus there is no MRF error, and

$M(V_{error})=1$. Therefore, the MRF method does not change the system performance in steady state.

During dynamics, V_{error} may be large at times and thus are mapped into coarse-resolution MRF levels, resulting in large MRF errors. Nevertheless, when the MRF error is large, V_{error} is also large; and through a proper design, the peak $M(V_{error})$ in each MRF level can be restricted to a small value close to *1* as shown in the following design example. If the gain margin of a system is sufficient, the MRF error should not have influence to the system stability.

B. Design Example

An example of MRF mapping is shown in Table 1. The explanation is as follows.

Table 1. Example of MRF mapping.

V_{error} (V)	V_{error} (LSB)	MRF Value (LSB)	MRF Code
-0.2632 to -0.2131	-42 to -34	-38	1111
-0.2068 to -0.1567	-33 to -25	-29	1110
-0.1504 to -0.1066	-24 to -17	-21	1101
-0.1003 to -0.0564	-16 to -9	-13	1100
-0.0501 to -0.0251	-8 to -4	-6	1011
-0.0188 to -0.0125	-3 to -2	-2	1010
-0.0063	-1	-1	1001
0	0	0	1000
0.0063	1	1	0111
0.0125 to 0.0188	2 to 3	2	0110
0.0251 to 0.0501	4 to 8	6	0101
0.0564 to 0.1003	9 to 16	13	0100
0.1066 to 0.1504	17 to 24	21	0011
0.1567 to 0.2068	25 to 33	29	0010
0.2131 to 0.2632	34 to 42	38	0001

The designed output voltage is *5V* with *±5%* output regulation range. The desired output accuracy is *±0.1%* therefore a 10-bit ADC is chosen which provides *1024* quantization levels. With the FSR set to *6.42V*, each LSB represents a *6.3mV* quantization step. As a result, the output regulation range (*5±0.25V*) is represented in between *±42* LSBs.

The *±42* LSBs are then divided into *15* MRF levels, with the finest level of *1* LSB when V_{error} is *0* LSB and ±1 LSB, a wider level of *2* LSBs when V_{error} is *±2* and *±3* LSB, and gradually widen to the widest level of *9* LSBs when V_{error} is from *34* to *42* LSB, and from *-42* to *-34* LSB. The *15* MRF levels are numbered from *1* to *16* using 4-bit binary codes, "*0001*" to "*1111*".

The 4-bit MRF code is sent to the primary side through the digital isolator. When decode the MRF code to recover the V_{error}, the median value of the corresponding MRF level is used since the original value is not recoverable. For example, the median value *-38* LSB is used when decoding the MRF code "*1111*".

The variable gain introduced by the MRF error in Table 1 is plotted in Fig. 9. The peak variable gain is *1.5*, or *3.5 dB*, which is rather small if the system has sufficient gain margin. In practical design, this extra gain should be considered when completing the loop design.

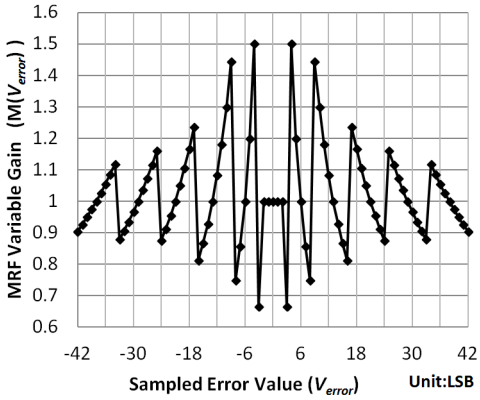


Fig. 9. Example of MRF Variable Gain Plot.

C. Simulation Results

The abovementioned MRF design example is simulated with a *100 kHz* digital Flyback converter in PSIM simulation software.

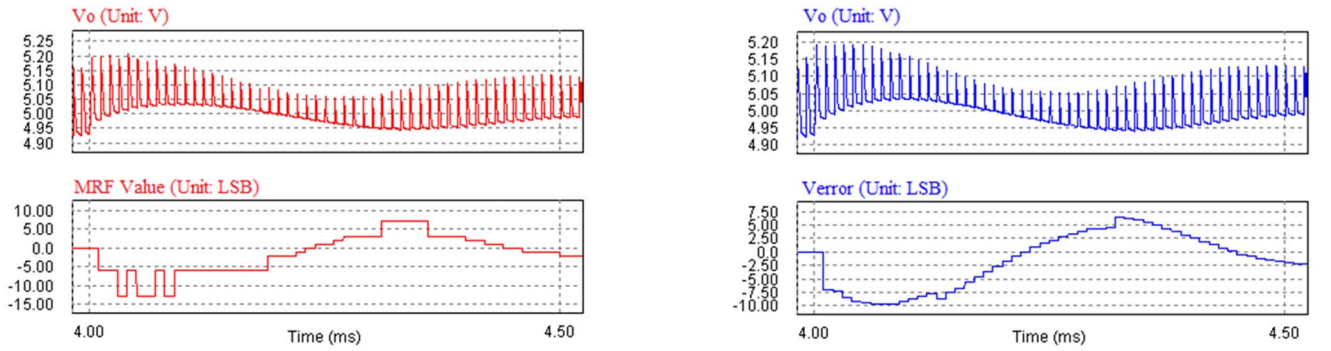
The simulation results are shown in Fig. 10 and Fig. 11. In both figures, the upper waveforms are the output voltage response to a load step change; the lower waveforms are the error signals (accurate or mapped) received by the primary-side controller during the transient. The name V_{error} denotes accurate error values transmitted without MRF; and the name *MRF Value* denotes the mapped error values decoded from MRF codes, which are subject to the MRF error. It can be observed that although the error signal decoded from MRF show coarse resolution when it is large, the output voltage response that with and without MRF are almost identical.

The comparisons in Fig. 10 and Fig. 11 indicated that the MRF method has no visible influence on the system's dynamic response.

D. Design Considerations

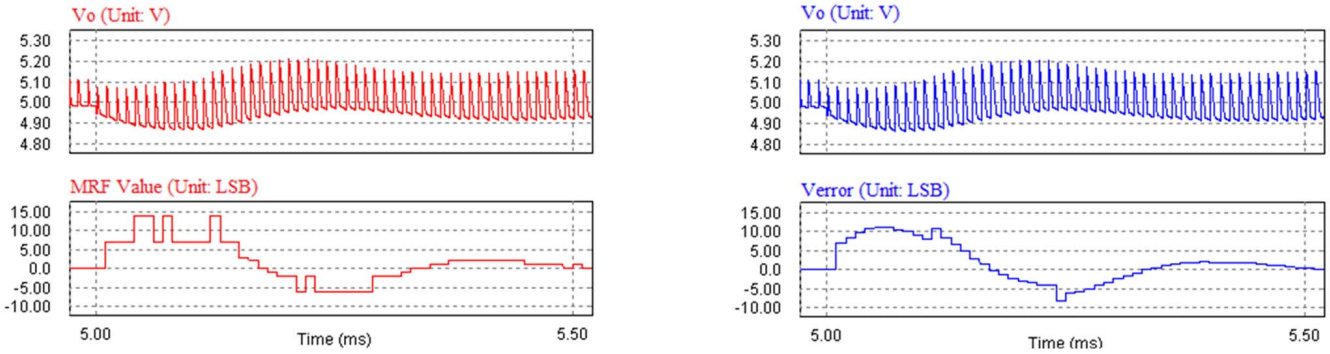
The following aspects should be considered when the MRF method is designed:

1. If the compensator is designed in such a way that steady-state error exists, the output voltage during steady state may deviate from the reference voltage, therefore the MRF levels must have the finest resolution throughout the steady-state error range, which significantly increases the number of MRF levels and the length of the MRF codes. Therefore steady-state error must be eliminated in a MRF-enabled system.



(a) System with MRF (b) System without MRF

Fig. 10. Simulation results (75% to 25% load step).



(a) System with MRF (b) System without MRF

Fig. 11. Simulation results (25% to 75% load step).

2. The finest resolution is only provided within a few LSBs near the reference voltage, therefore the sampling point must be carefully chosen in order to exclude the ripple voltage. In some topologies, synchronizing the sampling point to a certain time instant may reflect the average output voltage. In some other topologies, oversampling may be necessary to averaged out the ripple voltage.

V. Time-division Multiplexing 8B/10B Protocol

Existing protocols, such as UART, I^2C[34], SMBus [35], PMBus [36, 37], SPI [38], and SST, are not designed for real-time communication in digital power supplies. They are either too simple to implement required features, or too complex to be efficient. This section discusses the protocol requirements and proposes a multiplexed protocol that is optimized for isolated digital power supplies.

A. Considerations in Protocol Requirements

The following aspects are considered when search for a desired protocol.

(i) DC-balance

A primary requirement for the desired protocol is to maintain DC-balance in order to use low-cost pulse transformers. DC-balance means the number of "1"s and the number of "0"s are equal within a data stream segment, thus when driving the pulse transformer using low-voltage differential signaling (LVDS) technique, the net magnetization of the core is zero. DC-balance is important to prevent the pulse transformer from saturation.

In digital power supplies, during steady state the ADC samples are of an identical value which is the reference value. If the samples are not DC-balanced data, the pulse transformer will be quickly saturated. This problem can be solved by using line codes.

Line codes encode data into a DC-balanced form before transmission. Commonly used line codes to obtain DC-balance include Manchester code [39], 6B/8B code [40], 8B/10B code [41, 42], and so on.

None of the abovementioned existing protocols ensures DC-balance. Therefore a new protocol is desired which must incorporate one of the abovementioned line codes.

(ii) Multiplexing

A highly desired feature in the desired protocol is multiplexing of the data bus. There are three degrees of multiplexing: (a) in each power stage, all feedback information (e.g. output voltage, load current, etc.) are multiplexed into one data bus; (b) all power stages in the same module are multiplexed into one data bus; (c) the aforementioned feedback data and power management commands, such as emergency shut-down, are multiplexed into one data bus.

The above requirements lead to the following considerations:

1. If a frame of data consists of several packets, sequence

bits may be needed in each packet in order to identify the sequence of the packets and the body of the frames.

2. If frames are sent to different destinations, or if frames contain different types of information, address bits or type bits are necessary to identify the destination and the content of the frames.

3. Command codes must be distinguished from data. An indicator bit may be needed in each packet to tell the decoder whether it is a command code or data.

4. There must be a mechanism to interrupt data transmission prematurely in order to send through higher-priority commands, such as emergency shut-down function for over-voltage and over-current protections.

The length of sequence bits, address bits, and type bits are determined by the number of multiplexed elements; therefore the more the multiplexed elements, the more the protocol overhead, and the less efficient the protocol is.

UART protocol has been used in feedback loop communication, such as in [18]. It does not have an addressing mechanism. Therefore the address bits must be implemented in a higher level protocol, placed in the 8-bit payload, making the effective data bits even fewer.

A better addressing mechanism is desired, which should minimize the protocol overhead when multiplexing more elements. Also, the "high-priority interrupt" feature should be supported without a clock line, so that *clock recovery* can be supported, see below.

(iii) Clock recovery

In order to further reduce the cost by eliminating a separate isolator for the clock line, clock recovery should be used. The clock signal can be recovered and synchronized from the received data stream by using phase lock loop (PLL) or simply detecting the falling edge of the received bits, like it is done in UART. Therefore, the transmitted bit stream should have enough state changes in any condition, or the clock synchronization will be lost.

Unlike in telecommunications where a long run of an identical bit is statistically unlikely, in digital power supply systems, nothing is left to probability, and a long run of an identical bit is bound to happen in some scenarios. Such a long run of an identical bit will cause the recovered clock signal out of sync, and also drive the pulse transformer towards saturation, which was discussed in *(i) DC-balance* early.

Some line codes can be used to support clock recovery. Possible line codes include Bipolar Return-to-Zero (RZ) code [43], Manchester code, 4B/5B code [44], 6B/8B code, 8B/10B code, and so on. Bipolar RZ code and Manchester code guarantee a state change in every bit; however the bandwidth requirement is doubled. 4B/5B code provides at least one state change in every 5 bits. 6B/8B code provides at least one stage change in every 7 bits. 8B/10B code provides at least one stage change in every 7 bits as well, and at least an average of 30 state changes per 100 bits [41].

Among the existing protocols, the SST bus incorporates Bipolar RZ code to attain clock recovery [20]. However, Bipolar RZ code and 4B/5B code do not guarantee DC-balance, so they are excluded from the candidate line codes.

(iv) Error detection

Errors occurred in transmission include bit error and framing error. Bit error refers to incorrect transmission of a bit. Framing error refers to incorrect alignment of a packet or a frame.

Commonly used error detection methods include parity check and cyclic redundancy check (CRC). Parity check adds one parity bit after each packet, which is simple but does not guarantee to detect all the possible errors. CRC can detect both bit error and framing error. It does so by appending a byte-long Packet Error Code (PEC) at the end of each transaction. CRC provides better error detection but has two limitations: (a) it cost more CPU resource in those DSPs that do not have built-in CRC hardware; and (b) the received bytes must wait until the PEC is received at last before can be processed, which increases the requirements for bus speed and register length.

Alternatively, line codes such as 6B/8B code and 8B/10B code encodes data according to certain disparity rules and a codebook. Received codes that are not listed in the codebook or violate the coding rules are recognized as corrupted data. This method provides some error detection without adding any protocol overhead. Besides error detection at the coding layer, using differential signal to drive pulse transformer provides significant immunity to common-mode noise; and the PID block after the decoder can also prevent an undetected error from becoming a catastrophic failure. Therefore, if codebook-type line-codes are used in a protocol, it is possible to omit parity check or CRC in order to minimize the protocol overhead.

Resending the corrupted data is more complicated and is not required in digital power supplies. A more detailed discussion is in *(v) Bus direction*. More importantly, framing errors must be corrected as soon as possible to avoid further damage to the following data. In data buses that have unique "start" and "stop" conditions (e.g. I^2C, SMBus, and PMBs), packet misalignment can be corrected easily; however correcting frame misalignment requires using address/sequence bits, which adds protocol overhead. In other protocols, fixing framing errors require excessive framing bits (e.g. in [17]) or frequent CRC (e.g. in UART in some conditions). The desired protocol should have a better mechanism to detect all framing errors and correct them in time with minimum protocol overhead.

(v) Bus direction

In the real-time communication of digital power supplies, if a packet or a frame is found corrupted, it shall be abandoned; but the receiver needs not to ask the sender to resend the same data, because the data have lost their timeliness. Therefore, the desired protocol only needs to talk one-way, thus the complexity and the protocol overhead are significantly reduced.

In applications where bi-directional communication is desired, the communication of the two directions can run on independent buses. For example, the status monitoring data sent from the primary side to the secondary side are usually not time-critical; therefore the data bus can run at a lower speed, compared to the secondary-to-primary data bus, which carries feedback data and must run at a higher speed. The low-speed bus allows using inexpensive isolators and costs less CPU resources.

(vi) Idling time

The DPWM duty cycle should be updated within the same switching period in which an ADC sample is taken in order to achieve fast dynamic response. The time allowed for data transmission is the switching period less the sampling delay, the computational delay, and the device propagation delay. Therefore there is significant percentage of time when the data bus is idle, which is a waste of resources. Therefore it is once again justified to multiplex several functions onto the data bus in order to utilize the redundant bandwidth.

In the cases that the data bus becomes idle, many protocols hold the line at logic-high (e.g. UART, SMBus, PMBus, etc.), which is unacceptable because the pulse transformer will be saturated. Therefore, a DC-balanced code must be transmitted during idling time, which must be distinctive from data. Setting the LVDS driver to tri-state is an alternative solution but *clock recovery* may become a problem.

(vii) ADC sampling timing

In some applications it is desirable to control the timing of ADC sampling directly from the primary side, so that output voltage ripple can be eliminated in the ADC samples. A solution is provided in [19], which utilizes a primary-to-secondary communication channel to synchronize the sampling timing at the middle of PWM duty cycle. However, in many topologies such as DCM Flyback and most resonant topologies, the desired sampling point which reflects the average output voltage changes with load current, and is difficult to calculate especially when parasitic components are considered. Alternatively, oversampling strategy is simpler, cheaper, and can apply to all topologies. Recent digital power controllers (such as UCD3138, Texas Instruments Inc., and dsPIC33F family, Microchip Technology Inc.) provide oversampling capability without costing excessive CPU resources. The sampling synchronization signal can be obtained from the secondary winding of the power transformer. Therefore, control of sampling timing is not deemed a necessary feature in the protocol requirement.

B. Selection of Line Code

Based on above discussions, a desired protocol should use a line code. The candidate line codes include Bipolar RZ code, Manchester code, 4B/5B code, 6B/8B code, and 8B/10B code. Bipolar RZ code and 4B/5B code are ruled out because they don't provide DC-balance. Manchester code is ruled out because it doubles the bandwidth requirement. Therefore only 6B/8B code and 8B/10B code are left for comparison. They both provide DC-balance and support clock recovery.

(i) 6B/8B code

6B/8B code has been used in many telecom applications. The code encodes 6-bit words into 8-bit codes. Because there are 70 8-bit codes that consist of four "1"s and four "0"s, and there are only 64 6-bit words, all 6-bit words are encoded to a DC-balanced 8-bit code. Four additional DC-balanced 8-bit codes are used as control symbols.

A drawback of 6B/8B code is that it does not have a special "delimiter" code that is distinguishable in a misaligned data stream. For example, the control symbol "*01010101*" can be spelled by the code "*0110**0101***" followed by the code "***0101***0110". Thus in the presence of a framing error, when the receiver "sees" a "*01010101*" pattern in the bit stream, it cannot "tell" whether it is a misaligned control symbol (thus corrects the framing error), or composed by two correct data codes. With the absence of a "delimiter" code, framing error correction must rely on other layer's protocol.

(ii) 8B/10B code

8B/10B code has been used in USB 3.0, HDMI, Gigabit Ethernet, Serial ATA, and many more applications. The code encodes 8-bit words into 10-bit codes. Only those 10-bit codes that consist of five "0"s and five "1"s, or four "0"s and six "1"s, or six "0"s and four "1"s are used in the codebook. The encoding rule of 8B/10B code is briefly described as follows:

Each 8-bit (8B) word can be encoded into a pair of 10-bit (10B) codes. If a 10B code consists of five "0"s and five "1"s, it is disparity neutral; so does its pair code. If a 10B code consists of four "0"s and six "1"s, its disparity is +2; and its pair code must consists of four "1"s and six "0"s, and the disparity is -2.

The system's Running Disparity (RD) is monitored by the encoder: if the previously transmitted 10B code has +2 disparity, then when encoding the next 8B word, the 10B code with -2 disparity will be chosen to neutralize the RD. If the next 8B word is corresponding to a neutral 10B code, the RD does not change; then when the encoder encodes the following 8B word, it will again try to choose a 10B code that can balance the RD.

A profound advantage of 8B/10B code is that it has three "delimiter" codes that can be recognized in any misaligned data stream. In other words, these "delimiters" will not appear in any occasion by two other 10B codes overlapping. Therefore, when the receiver "sees" a "delimiter" pattern in a bit stream, it immediately "knows" this is a "delimiter", and correct the framing error if the data stream is misaligned. The "delimiter" codes are very useful tools for frame synchronization, emergency shut-down, and other critical functions. The 8B/10B code also provides 8 other control symbols which are distinguishable from data.

Above discussions show that 8B/10B code is the preferred line code for the desired protocol. A new optimized protocol for power supply applications is proposed in the following section.

C. Time-division Multiplexing 8B/10B Protocol Description

The proposed protocol is a unidirectional time-division multiplexing (TDM) protocol based on 8B/10B line code; therefore it is named Time-division Multiplexing 8B/10B (TDM 8B/10B) protocol. The TDM mechanism divides each data frame into several channels of fixed length, so that several real-time variables can each take a channel and share one physical data bus. The 8B/10B line code provides DC-balance and clock recovery so as to use only one pulse-transformer as low-cost isolator. The proposed protocol also utilizes the "delimiter" codes and control symbols to minimize the protocol overhead.

The basic structure of the proposed protocol is described as

follows.

(i) Format

Feedback data are transmitted by recurrent frames. The frame rate is determined by the required data update frequency. The format of a frame is shown in Fig. 12. Each frame consists of several packets. The number of packets depends on the number of elements multiplexed. Depending on the functionality, packets are divided into frame header and payload. Frame headers are part of the protocol overhead; and payloads are the feedback data being carried by the protocol. There is no need for a "stop" packet, because the frame length is fixed.

A packet is an 8B/10B code, thus a payload packet has 8 effective data bits. They can be the 8 MSBs or LSBs of an ADC (or the sampled error value), or composed by two 4-bit MRF codes, which was proposed in Section IV.

A frame header packet is a control symbol provided by the 8B/10B codebook, which does not contain any explicit information but flagging the beginning of a frame. However, different control symbols indicate different frame types. Therefore, there is no protocol overhead of having several frame types for *higher-order multiplexing*, which will be discussed further below.

(ii) Addressing

There is no explicit address bit. The addresses are indicated by a fixed sequence of the packets within a frame. For example, the first packet following the frame header contains the output voltage of Power Stages 1; the second packet following the frame header contains the output voltage of Power Stages 2; the third packet following the frame header contains output current of Power Stage 1, and so forth. Different frame types have different address definitions. The decoder firstly recognizes the frame type according to the frame header, and then decodes the addresses according to a pre-defined sequence.

This addressing mechanism can host any number of addresses and does not add any protocol overhead.

(iii) Frame and packet synchronization

The frames and packets are synchronized by the frame headers. At least one frame header should be a "delimiter" code, which is named "sync". When the receiver "observes" a "sync" pattern in its register, it immediately resets the bit counter and the packet counter regardless their current values. As a result, the next 10 bits are considered the first payload packet of a frame, which is correct. This synchronization method guarantees accurate alignment at the beginning of each frame, thus prevents any framing error from carrying forward to the next frame.

The operation of the proposed protocol is described as follows.

(i) Auxiliary functions

Auxiliary functions can be encoded and transmitted in the payload packets same way as data. The decoder will recognize auxiliary function codes by its address information, which is pre-defined and indicated by the packet's sequence.

Alternatively, control symbols can be inserted into data stream every now and then in order to implement infrequent auxiliary functions.

(ii) High-priority interrupts

The "high-priority interrupt" feature is implemented by a "delimiter" code other than the "sync". In the case of an emergency, the DC-balance does not have to be honored; therefore the traffic can be stopped at any time to let the high-priority commands go through.

For example, a "delimiter" code can be named "shut-down". Whenever the receiver "observes" such a pattern in its register, it will recognize the command and shut down the system immediately. Alternatively, this "delimiter" code can be used as a special frame header, followed by a few encoded high-priority commands in the payload. This will allow several high-priority commands to share a "delimiter" since there are only 3 of them.

When the traffic resumes, the "sync" symbol will re-establish the packet and frame synchronization.

(iii) Higher-order multiplexing

The parameters being monitored may have different sampling frequencies. Communicating at a fixed frame rate for all channels may waste significant bandwidth. The capacity to define multiple frame types allows for intelligent adaptation of the sampling frequencies, known as higher-order multiplexing.

For example, a power supply module may have a main power stage operating at *100 kHz*, whose voltage loop is communicating at *100ksps*, in Channel 1; and a standby power stage operating at *50 kHz*, whose voltage loop is communicating at *50 ksps*, in Channel 2. Both power stages' output currents are monitored at *25 ksps*, in Channel 3 and Channel 4, respectively. In this example, all the samples are 8-bit long. If the 4 channels are contained in the same frame, and communicate at *100 kilo frames per second*, the required data bus speed will be $50\ bit/frame \times 10^5\ frame/s = 5\ Mbps$, in which only *60%* of the bits carry useful information.

A better alternative is to use multiple frame types, refer to Fig. 13. Continuing the above example, define three frame types, and each type has two payload packets: Type 1 contains Channel 1 and Channel 2; Type 2 contains Channel 1 and Channel 3; and Type 3 contains Channel 1 and Channel 4. By multiplexing the frame types same as shown in Fig. 13, the

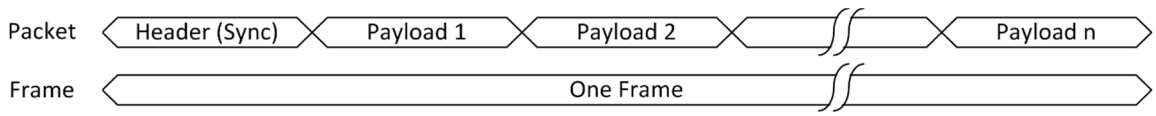


Fig. 12. Format of a frame. A frame starts with a frame header packet, followed by several payload packets.

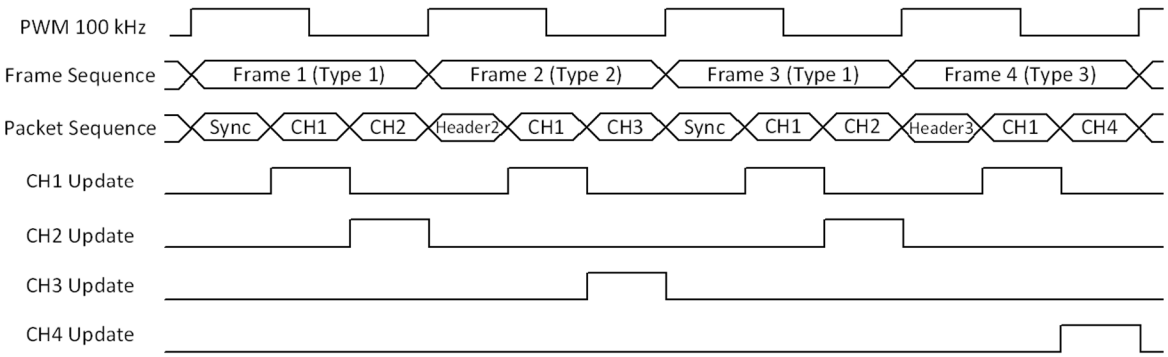


Fig. 13. Example of higher-order multiplexing. CH1 communicates at *100 ksps*; CH2 communicates at *50 ksps*; CH3 and CH4 communicate at *25 ksps*.

required data bus speed is only $30\ bit/frame \times 10^5\ frame/s = 3\ Mbps$. All channels are communicating their own frequency, and no bandwidth is wasted.

(iv) Idle state packet stuffing

The "sync" code is sent repeatedly during the idle state. There are two purposes of the packet stuffing: 1. to maintain DC-balance; 2. to maintain clock synchronization.

Although the receiver expects to receive a payload packet after the "sync" code since it is also a frame header, repeating the "sync" code will only reset the counters time after time, and will not cause confusion. If a different frame header code is received immediately after the "sync" code, the receiver will automatically "understand" the previous "sync" code was a stuffing packet.

(v) Clock mismatching adjustment

The clock frequency on the secondary side may be slightly different than that on the primary side; therefore if the power stage is operating at a fixed switching frequency, which is determined by the primary-side DSP, the packets sent from the secondary side may slightly mismatch the switching cycles.

Several methods can solve this problem.

First, if there is also a data bus sending data from the primary side to the secondary side, the primary-side clock can be acquired by the secondary-side receiver. And the secondary-side transmitter can use the same clock.

Second, the secondary-side DSP may acquire the primary-side switching frequency by detecting signals on the secondary-side windings, and then adjust its clock accordingly.

The third method relies on the protocol: intentionally design the secondary-side transmitter slightly faster than the desired speed, and repeat the frame header in the case that payload data are not ready for transmission.

For example, if a data bus is desired to send 1 frame per switching cycle, but it actually sends 1 frame plus 0.1 packets per switching cycle; then after sending the first frame, the secondary side is *leading* the primary side by 0.1 packets. The consequence is that when the secondary side is about to send the second frame, the payload data is not ready yet if the sampling is synchronized with the switching frequency. In that case, the frame header repeats itself once; as a result the secondary side becomes 0.9 packets *lagging* the primary side. After that, the data will always be ready before needed, until 10 switching cycles later, when the secondary side again leads the primary side by 0.1 packets. In this example, the frame header needs to repeat once every 10 frames. Because the frame header is a control symbol, repeating it will not cause any confusion.

For variable switching frequency topologies, the data transmission is not required to sync with the switching frequency; therefore the clock mismatching will not cause any problem.

(vi) Incorporating MRF

The proposed protocol can incorporate the proposed MRF method. With MRF, each 8-bit payload packet can be composed by two 4-bit MRF codes; therefore the data bus capacity is doubled, or the data bus speed can be reduced.

(vii) Bi-directional communication

Two unidirectional data buses of opposite directions can form a bi-directional data bus. The operations of both directions are independent, allowing a less time-critical data bus to operate at a lower speed, which saves substantial cost and CPU resources.

D. Solution comparison

A comparison of the proposed solution and other solutions is shown in Table 2. It shows the proposed system has the highest output accuracy and the lowest data transfer requirements with many features which reduce the system cost.

VI. Implementation

An experimental prototype is built to verify the feasibility and to demonstrate the advantages of the proposed solution.

The block diagram of the implemented prototype is shown in Fig. 14. The prototype consists of two *5V/5A* Flyback converters that are independently regulated. The switching frequencies are *100 kHz*. The input voltage is *48V*. The transformer's turns ratios are *3:1*. The transformer's magnetizing inductances are *15μH*. Two digital signal controllers (DSCs) (dsPIC30F2020, Microchip Technology Inc.) are used on the primary side and the secondary side,

Table 2. Comparison of the proposed solution and existing solutions.

Solutions	TDM 8B/10B protocol with MRF (the proposed solution)	Sending 4 LSB using a customized protocol [17]	Sending 8 MSB using UART [18]	2-way bus and state-dependent ADC [19]
DC-Balance	Yes	No	No	No
Multiplexed	Yes	No	No	Possible but limited
Number of Isolators	1 pulse-transformer for all power stages	2 digital opto-couplers for each power stage	1 digital opto-coupler for each power stage	2 digital opto-couplers for each power stage
Accuracy	0.098% (10-bit ADC)	0.392% (8-bit ADC)	0.392% (8-bit ADC)	0.392% (8-bit ADC)
Clock Recovery	Support	Do not support	Do not support	Support
Error Detection	Yes	Yes	Yes	Yes
Coding efficiency (incl. protocol overhead)	$10/N \times [ceil(N/2)+1]$ bits per sample. N is the number of multiplexed elements. *	16 bits per sample	12 bits per sample	16 bits per sample/2 (2-way communication)

* Example: For 2 channels, $10/2 \times [ceil(2/2)+1] = 10$ bits per sample. For 4 channels, $10/4 \times [ceil(4/2)+1] = 7.5$ bits per sample. For 9 channels, $10/9 \times [ceil(9/2)+1] = 6.67$ bits per sample. More multiplexed elements result in higher coding efficiency.

respectively. The ADCs, MRF codec, protocol codec, PID controller, and DPWM are implemented in the DSCs. Due to the lack of peripheral support in the DSCs, the serializer, de-serializer, and word aligner are built with discrete logic gates. The multiplexed data in the data bus include the feedback voltage loop data of the two Flyback converters, a secondary-side On/Off command, and Over Voltage Protection commands. The digital isolator is a pulse transformer (H1102NL, Pulse Engineering, Inc.). It is driven by discrete LVDS transceivers.

Due to lack of peripheral support in the DSC, and also for simplicity, a separate clock line instead of a clock recovery block is used in this prototype. However, since the clock recovery feature is available in many FPGA-based codec modules, and since it has been extensively used in almost all 8B/10B-based communications, the ability of the 8B/10B line code to support clock recovery has been thoroughly proved by real-world applications. Therefore this simplification shall not compromise the conscientiousness of this paper.

The data rate of the communication bus is *8 Mbps*, which includes seven payload packets per frame at *100 kilo frames per second*. With this speed, the data bus is capable of multiplexing 14 4-bit-long feedback data at *100 kHz* sampling frequency using the proposed protocol. In this prototype, only the first payload packet is used to carry two 4-bit MRF codes, and the rest payload packets are filled with "*01100 01011*" (corresponding to "0000 0000" in 8B code).

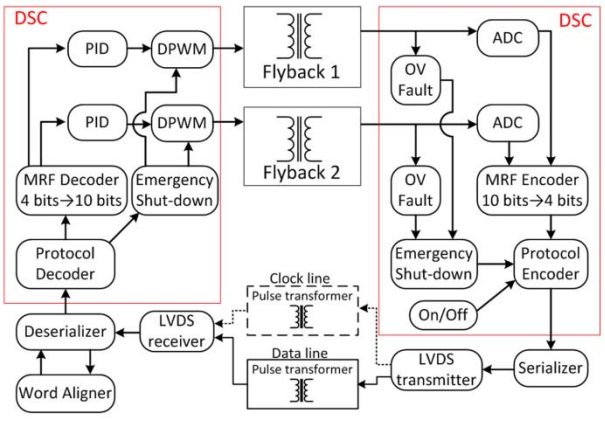


Fig. 14. Block diagram of the implemented prototype.

The example MRF mapping table in Table 1 is used in the prototype. The output regulation range, the ADC FSR, and the LSB resolution are the same as in the example.

The control scheme of the prototype is described as follows: when the secondary-side switch is turned on, the secondary-side DSC repeatedly sends "sync" code to the primary side; the primary-side DSC receives the "sync" code and turns on the power supply. Once the secondary-side DSC detects the output, it stops sending "sync" and begins to send sampled error value using the proposed protocol. The soft-start is achieved by gradually increasing the reference voltage value in the secondary-side DSC program. When the secondary-side switch is turned off, or over-voltage condition is detected, the secondary-side DSC interrupts the data communication and repeatedly sends "shut-down" code to the primary side. The primary-side DSC then shuts down the power stages. The secondary-side DSC will stop sending the "shut-down" code when it detects that the power MOSFET stopped switching.

Fig. 15 and Fig. 16 show the load transient responses. The waveforms do not show any nonlinear behavior, which is consistent with the simulation results. This test proves that the proposed MRF method does not affect the dynamic performances.

Fig. 17 shows the output regulation test. The results show excellent output accuracy at different input voltage and load conditions. This test proves the effectiveness of the proposed MRF method.

In order verify the implementation of the proposed TDM 8B/10B protocol, as well as the DC-balance of the pulse transformer, data stream in 9 successive frames are captured in a single snapshot using a deep-memory oscilloscope, and are zoomed in at each frame to extract the transmitted data. The close-up waveforms of the first frame are in Fig. 18. The extracted data are in Table 3. In Table 3, the Disparity column shows that the ±2 disparity 10B codes are sent alternately to balance the system RD. The MRF Code column shows that the output voltages of both power stages are exactly regulated within 1 LSB of the reference level (1000), which again proves the effectiveness of the proposed MRF method.

A photo of the implemented prototype board is shown in Fig. 19.

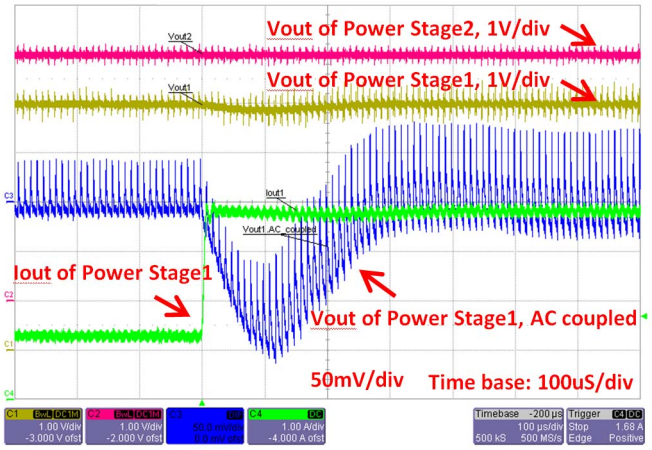


Fig. 15. Load transient response 25% to 75% load. No nonlinear behavior.

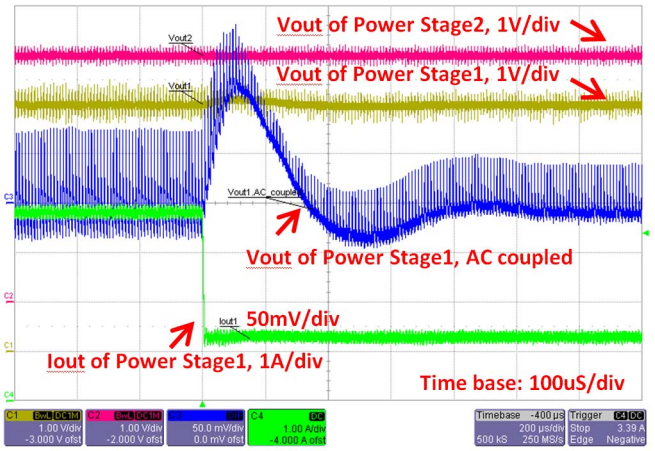


Fig. 16. Load transient response 75% to 25% load. No nonlinear behavior.

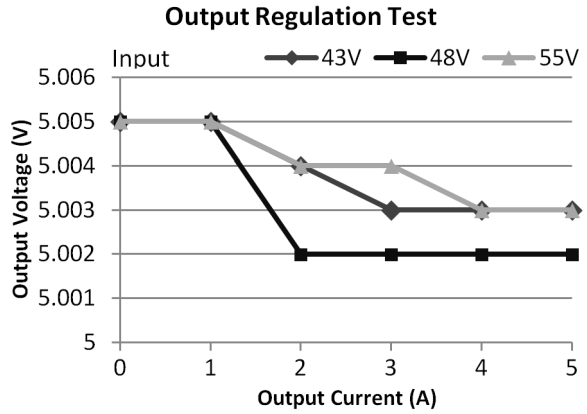


Fig. 17. Output regulation test. The results show excellent output accuracy.

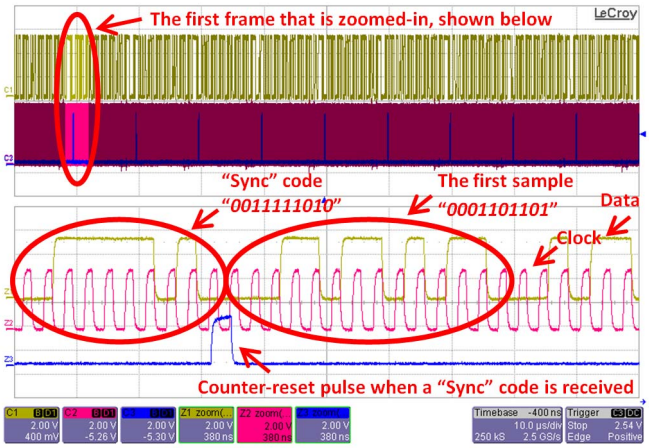


Fig. 18. Data stream in 9 successive frames (zoom-in at the 1st sample).

Table 3. Extracted data from 9 successive frames.

Frame #	Packet #	10B Code	8B Code		Disparity
			CH1	CH2	
1	header	0011111010	Delimiter		+2
	1	0001101101	1000	1000	0
2	header	1100000101	Delimiter		-2
	1	1110010010	1000	1000	0
3	header	0011111010	Delimiter		+2
	1	0001101101	1000	1000	0
4	header	1100000101	Delimiter		-2
	1	1110010010	1000	1000	0
5	header	0011111010	Delimiter		+2
	1	0001101101	1000	1000	0
6	header	1100000101	Delimiter		-2
	1	1001101100	0111	1001	0
7	header	0011111010	Delimiter		+2
	1	0001110010	1000	0111	-2
8	header	0011111010	Delimiter		+2
	1	0001101101	1000	1000	0
9	header	1100000101	Delimiter		-2
	1	1110010010	1000	1000	0

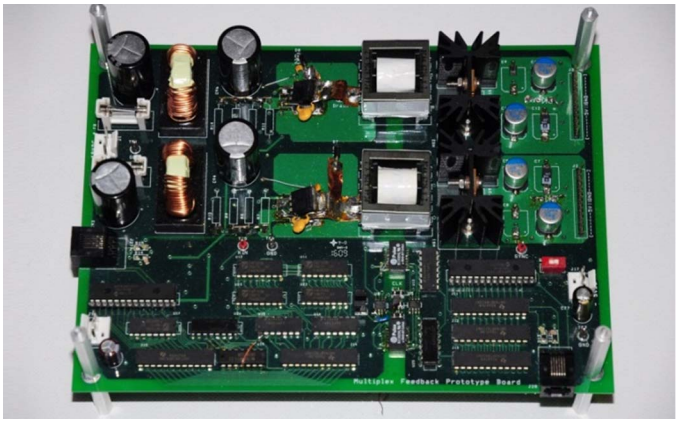


Fig. 19. Photo of the prototype board.

VII. Variations of the Proposed Solution

The proposed solution includes three main ideas: (a) use Multi-resolution Feedback method to compress sample data; (b) use a DC-balanced line code so that low-cost pulse transformers can be utilized; (c) use the "delimiter" codes to implement packet synchronization, frame synchronization, implicit addressing system, high-priority interrupt, higher-order multiplexing, etc., so that the protocol is efficient and feature-rich. For different applications, the implementations may vary. Some of the variations are discussed below.

(i) MRF variations

Depending on the requirements of the output accuracy and output regulation range, the MRF code may have different length.

The MRF method can also fit into other protocols. For example, two 4-bit MRF codes can fit into a UART bus.

(ii) Line code variations

Other line codes such as 12B/14B, 16B/18B, 17B/20B [45] are also available. The selection of line codes is determined by the DC-balance feature and the "delimiter" codes, as well as the length of the data that it carries. For example, two 4-bit MRF codes fit into an 8B/10B line code, whereas two 6-bit MRF codes better fit into a 12B/14B code.

(iii) Bi-directional communication

The proposed solution is for one-way communication. However, it can be easily extended to bi-directional communication by pairing two independent data buses of opposite directions.

A further implementation is to use one isolator for bi-directional communication. Since the frame length is fixed, a transmitter can easily predict when the traffic data from the other side will stop, and then it will take over the data bus. Also, a control symbol can be used to explicit the end of a data stream.

(iv) Multiple access communication

The implemented prototype is point-to-point communication, which means there is only one terminal on each side. However, in some applications, there may be several terminals that require communication. The proposed solution can be extended to a time division multiple access (TDMA) structure. Since each channel has a fixed time slot to transmit data according to a fixed sequence, all the devices on the data bus can "listen" to the traffic—synchronized by the "sync" code—and determine the time slot for them to send/receive data, one after another.

VIII. Conclusion

Footprint area and cost of isolators are the key considerations when choosing a feedback method for isolated digital power supplies, which translate into quantity, type, and speed requirement of the isolators. This paper reviewed possible feedback solutions and proposed a new solution which is optimized through the abovementioned aspects: it minimizes the quantity of the isolators by multiplexing all the feedback information into one data bus, and utilizes clock recovery so that only one isolator is needed; it includes a DC-balanced line code so that this one isolator can be a low-cost pulse transformer. In order to minimize the speed requirement, two measures were taken: by compressing sample data using the proposed MRF method and by minimizing protocol overhead using the proposed TDM 8B/10B protocol. The result of above optimizations is a highly efficient communication solution that provides the highest output accuracy with the lowest speed requirement and the fewest isolators; thus the footprint area can potentially be the smallest and the cost will be comparable to conventional analog solutions. The solution also possesses important features such as detecting bit errors, correcting framing errors within one frame, and interrupting data transmission so as to push through high-priority commands, such as overvoltage shutdown. These features ensure the necessary reliability required by power supplies. The proposed solution also enables users to multiplex any number and types of information into the data bus, thus advanced digital control techniques are made possible. The feasibility of the proposed solution is proved by a prototype. Discussions on possible variations are also provided in the paper.

References

[1] "PFE1100-12-054xA Data Sheet," Power-One Inc., February 2012.

[2] "End-to-End Embedded Power - 2012 Product Selection Guide," GE Power Electronics, Inc., 2012.

[3] "DS1200DC Distributed Power Bulk Front-End," Emerson Network Power, 2011.

[4] "Ericsson Power Modules Selection Guide," Telefonaktiebolaget LM Ericsson, June 2012.

[5] "CFE400M," Data Sheet, TDK-Lambda, May 2012.

[6] "UCD3138 Highly Integrated Digital Controller for Isolated Power," Data Manual, Texas Instruments Incorporated, July 2012.

[7] A. Bersani, A. Dumais, and S. Khare, "DC/DC LLC Reference Design Using the dsPIC DSC," AN1336, Microchip Technology Inc., 2010.

[8] L. Balogh, "A Practical Introduction to Digital Power Supply Control," Seminar, Texas Instruments Incorporated. 2005.

[9] D. Maksimovic, R. Zane, and R. Erickson, "Impact of digital control in power electronics," in *Proc. Int. Symp. Power Semicond. Devices and ICs, (ISPSD '04)*, 2004, pp. 13-22.

[10] P. Scoggins. (2007, January 1) A Guide to Designing Gate-Drive Transformers. *Power Electronics Technology*. pp32-36. Available: http://www.ferroxcube.com/news/gate%20drive%20trafo.pdf

[11] A. Reiter and A. Dumais, "Platinum-rated AC/DC Reference Design Using the dsPIC DSC," AN1421, Microchip Technology Inc., 2012.

[12] K. Leung and D. Alfano, "Design and implementation of a practical digital PWM controller," in *Proc. Appl. Power Electron. Conf. and Expo.(APEC '06)*, 2006, pp. 1437-1442.

[13] W. Feng, X. Yang, H. Ye, X. Wang, and G. Xiao, "Digital PWM Control for Isolated DC-DC Converters with Two Pulses Modulating PWM," in *Proc. Appl. Power Electron. Conf. and Expo.(APEC '09)*, 2009, pp. 269-273.

[14] A. V. Peterchev and S. R. Sanders, "Quantization resolution and limit cycling in digitally controlled PWM converters," *IEEE Trans. Power Electron.*, vol. 18, pp. 301-308, 2003.

[15] B. Chen, "Isolation in Digital Power Supplies Using Micro-Transformers," in *Proc. Appl. Power Electron. Conf. and Expo.(APEC '09)*, 2009, pp. 2039-2042.

[16] "HCNR200 and HCNR201 High-Linearity Analog Optocouplers," Data Sheet, Avago Technologies, 2011.

[17] A. Prodic, D. Maksimovic, and R. W. Erickson, "Digital controller chip set for isolated DC power supplies," in *Proc. Appl. Power Electron. Conf. and Expo.(APEC '03)*, 2003, pp. 866-872 vol.2.

[18] "SMPS AC/DC Reference Design User's Guide," DS70320B, Microchip Technology Inc., 2008.

[19] M. Scharrer, M. Halton, A. Scanlan, and K. Rinne, "Efficient Bi-Directional Digital Communication Scheme for Isolated Switch Mode Power Converters," *IEEE Trans. Circuits Syst. I, Reg. Papers,* vol. 59, pp. 3081-3089, 2012.

[20] R. V. White and D. Freeman, "Data Communications Issues For Power System Management," in *Proc. Appl. Power Electron. Conf. and Expo.(APEC '07)*, 2007, pp. 1188-1199.

[21] W. Elmenreich and M. Delvai, "Time-triggered communication with UARTs," in *IEEE Int. Workshop Factory Commun. Syst.*, 2002, pp. 97-104.

[22] M. Scharrer, M. Halton, T. Scanlan, and K. Rinne, "State-dependent ADC scheme for digitally isolated SMPC," in *Int. Conf. Electron. Circuits, Syst. (ICECS'10)*, 2010, pp. 986-989.

[23] Z. Hu, T. H. Yeap, and Y.-F. Liu, "Multi-resolution feedback to minimize communication data and improve output accuracy," in *Proc. Energy Convers. Congr. and Expo. (ECCE'10)*, 2010, pp. 788-795.

[24] Z. Hu, Y.-F. Liu, and T. H. Yeap, "An efficient communication protocol for single- and multi-module isolated digital power supplies using a single pulse-transformer," in *Proc. Energy Convers. Congr. and Expo. (ECCE'10)*, 2010, pp. 1997-2003.

[25] "5.2 Electric strength," in *IEC 60950-1, Part 1: General requirements*, ed: International Electrotechnical Commission, 2005, p. 367.

[26] "LogiCORE IP Aurora 8B/10B v5.2 User Guide," Xilinx, Inc., July 23, 2010.

[27] "Implementing an 8b/10b Encoder/Decoder for Gigabit Ethernet in the Actel SX FPGA Family," Application Note AC135, Actel Corporation, October 1998.

[28] "8b10b Encoder/Decoder MegaCore Function (ED8B10B)," Data Sheet, Altera Corporation, November 2001.

[29] "Gate Counting Methodology for APEX 20K Devices," Application Note 110, Altera Corporation, September 1999.

[30] K. Sayood, *Introduction to Data Compression, 4th ed.* U.S.A.: Elsevier, Inc., 2012.

[31] D. Salomon, *Data compression: the complete reference, 4th ed.*: Springer-Verlag London Limited, 2007.

[32] D. A. Huffman, "A Method for the Construction of Minimum-Redundancy Codes," *Proceedings of the IRE,* vol. 40, pp. 1098-1101, 1952.

[33] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*: Pearson Education, 2005.

[34] "The I^2C bus specification, revision 2.1," Philips Semiconductors, February 5, 2000.

[35] "System management bus (SMBus) specification, version 2.0," SBS Implementers Forum, August 3, 2000.

[36] "PMBus power system management protocol specification, part I - general requirements, transport and electrical interface, revision 1.1," System Management Interface Forum, Inc.February 5, 2007.

[37] "PMBus power system management protocol specification, part I - command language, Revision 1.1," System Management Interface Forum, Inc.February 5, 2007.

[38] "SPI Interface Specification," VTI Technologies, September 19, 2005.

[39] R. Forster, "Manchester encoding: opposing definitions resolved," *Eng. Sci. and Educ. J.,* vol. 9, pp. 278-280, 2000.

[40] A. X. Widmer, "DC-balanced 6B/8B transmission code with local parity," United States Patent 6,876,315, April 5, 2005.

[41] A. X. Widmer and P. A. Franaszek, "A DC-Balanced, Partitioned-Block, 8B/10B Transmission Code," *IBM J. of Research and Develop.,* vol. 27, pp. 440-451, 1983.

[42] P. A. Franaszek and A. X. Widmer, "Byte oriented DC balanced (0,4) 8B/10B partitioned block transmission code," United States Patent 4,486,739, December 4, 1984.

[43] D. J. Morris, *Pulse Code Formats for Fiber Optical Data Communication: Basic Principles and Applications*: CRC Press, 1983.

[44] D. Barrett and T. King, *Computer networking illuminated*: Jones and Bartlett Publishers, Inc, 2004.

[45] A. X. Widmer, "DC balanced 7B/8B, 9B/10B, and partitioned DC balanced 12B/14B, 17B/20B, and 16B/18B transmission codes," United States Patent 6,614,369 B1, December 4, 2003.