

## Research Article

# Absolute Difference and Low-Power Bus Encoding Method for LCD Digital Display Interfaces

Chia-Hao Fang,<sup>1</sup> I-tao Lung,<sup>2</sup> and Chih-Peng Fan<sup>1</sup>

<sup>1</sup>Department of Electrical Engineering, National Chung Hsing University, Taichung 402, Taiwan

<sup>2</sup>High Concentration Photovoltaic R&D Project, Institute of Nuclear Energy Research, Taoyuan 325, Taiwan

Correspondence should be addressed to Chih-Peng Fan, cpfan@dragon.nchu.edu.tw

Received 31 August 2012; Accepted 20 November 2012

Academic Editor: Chang-Ho Lee

Copyright © 2012 Chia-Hao Fang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Power dissipation has been an inevitable problem of LCD systems for years. To ease the problem, many encoding methods have been developed, such as the methods of transition minimized differential signaling, the most popular one in use for DVI to date, chromatic encoding, and limited intraword transition. In this paper, the authors present the absolute difference and low-power encoding method for the serial transmission of LCD digital DVI display interface. In regard to the LCD digital display interface with UMC 90 nm technology, the proposed method minimizes the architectural complexity and reduces the power dissipation by about 67% and 12%, respectively, compared with the transition minimized differential signaling and limited intraword transition. In short, the proposed method is an efficient bus encoding method to largely decrease the dynamic and total power dissipation of the LCD digital display interfaces.

## 1. Introduction

As display technology evolves, the standard of digital visual interface (DVI) 1.0 [1] has replaced the conventional analog video graphics array (VGA) standard. However, the power dissipation of LCD systems is still an annoying difficulty to deal with. The unit capacitance (up to 50 pF/m for DVI) of LCD systems bus connection through a cable is higher than the one of PCB bus connection, and the power dissipation of LCD panels can easily exceed 1 Watt [2] with the transition minimized differential signaling (TMDS) method. Hence, not only can the performance of LCD panels be impaired by the overheating problem, but designing an efficient connection bus with low power dissipation can cost 10~15% of LCD systems budget as well [3]. Thus, reducing power dissipation becomes an important concern of all contemporary LCD system manufacturers. Many well-known bus encoding techniques [3–10] soften this latent problem, such as TMDS, chromatic encoding (CE), and limited intra-word transition (LIWT). Techniques in [5–7] reduce power dissipation by minimizing switching and coupling activities of on-chip bus, while techniques in [8–10] achieve the same objectives of off-chip bus. Here, in this paper, authors present an efficient

power reduction bus encoding method: absolute difference and low-power (ADALP) for LCD digital display interfaces.

## 2. Encoding Methods of the Digital Display Interfaces

A typical LCD system uses a digital interface to transmit the pixel data from the graphic controller to the LCD display [11]. The pixel data on the display are scanned in a left-to-right and top-to-bottom format resulting in two adjacent pixels mostly with slightly different values. The digital interface in a typical LCD system has the standard protocol, that is, DVI, which is based on the TMDS serial link. In this section, several encoding methods are introduced, such as TMDS, CE, and LIWT. TMDS uses a clock channel and three data channels (0-1-2 or 3-4-5) for R-G-B colors. Each pixel has three colors, respectively, transmitting through three data channels at the same time, and each color has an 8-bit source-word ranging from 0 to 255. The encoded process for the active data can be viewed as a two-stage procedure. At first stage, it uses an additional control bit (i.e., the 9th bit) to determine whether to use the XOR or XNOR function to

minimize the transition activity. When the 9th bit is set to “High,” the first stage uses the XOR function; otherwise, the first stage uses the XNOR function. At the second stage, it produces the 10th bit, inverting the 8-bit words in order to balance the DC signal. For example: three subpixels (i.e., 122, 122, and 123) for the testing data of the TMDS encoding method are transformed into 1010000011, 0001111100, and 1001111100. Numbers of 0 and 1 are even, and both are equal to 50% [1]. CE utilizes one additional control bit in the R, G, or B channels of the serial signal transmission to provide an amount of transition reduction; however, its architecture is relatively complex and considers no DC balance. Therefore, CE is scarcely used for the commercial products. LIWT improves the disadvantage of CE and merits reputations for providing amounts of transition reduction, needing no additional control bit to get the DC balance, and requiring little complexity in architecture. In this paper, the scheme of ADALP takes the form of the limited intra-word transition codes in codewords, and the interpixel correlation of typical images, but transmitting with the absolute difference value to produce an output with unsigned number for obtaining high correlation between adjacent pixels, explained as follows.

### 3. ADALP Encoding Method

*3.1. Algorithm of the Proposed ADALP Encoding Method.* ADALP provides an efficient encoding technique for digital display interfaces. The proposed method reduces the dynamic power dissipation through decreasing the signal transitions. In addition, it uses the same bus width as TMDS does; thus, no additional control bits are required. Figures 1 and 2 show the algorithm of the proposed encoding and decoding method.

In summary, three steps are taken for the detailed algorithm of the proposed encoding method, as follows.

*Step 1.* The two consecutive subpixels are demarcated as  $D_t$  [0:7] and  $D_{t-1}$  [0:7], which determine the 10th bit of the encoded output, that is,  $q\_out$  [9]. When the value of  $D_t$  [0:7] is larger than that of  $D_{t-1}$  [0:7], the signal of  $q\_out$  [9] is set to “High”; otherwise, it is set to “Low.” Furthermore, the values of  $D_t$  [0:7] and  $D_{t-1}$  [0:7] are transmitted into the absolute subtractor for the difference, which is indicated by “Abs\_diff” in Figure 1.

*Step 2.* The absolute difference value of Abs\_diff decides the signal of  $q\_out$  [0]. If the value of Abs\_diff is located at the defined range in the range detector, the signal of  $q\_out$  [0] is set to “High”; otherwise, it is set to “Low.” If the signal of  $q\_out$  [0] is set to “High,” the value of Abs\_diff takes the values of the 2-ADALP codewords. Otherwise, the encoded value takes the value of the source words.

*Step 3.* Finally, ADALP allocates the positions of the signal of  $q\_out$  [0], the encoded value, and the signal of  $q\_out$  [9] in order.

Where  $D_t$  [0:7] and  $D_{t-1}$  [0:7] are two consecutive subpixels,  $q\_out$  is a 10-bit encoded output, that is,  $q\_out$  [0:9], || indicates the absolute value, the value of Abs\_diff

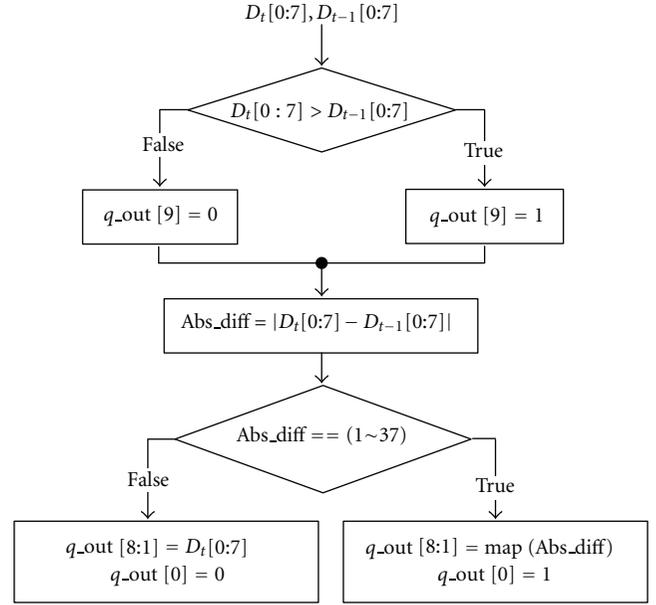


FIGURE 1: Encoder algorithm of the proposed ADALP.

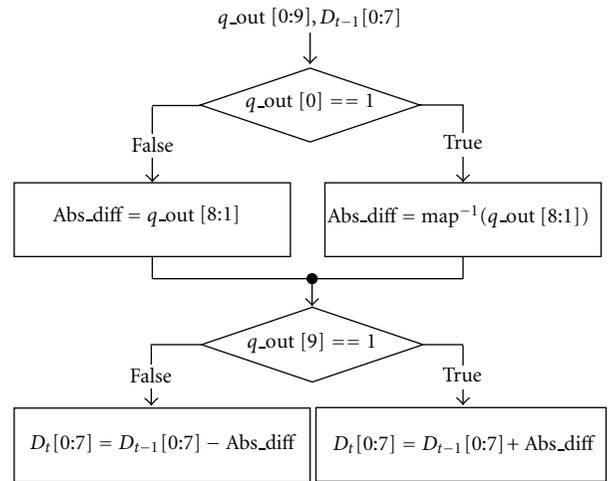


FIGURE 2: Decoder algorithm of the proposed ADALP.

is the absolute difference value of consecutive subpixels, and  $\text{map}(\text{Abs\_diff})$  is the codeword value.

*3.2. Architecture of the ADALP Encoding Method.* The proposed encoding architecture is shown in Figure 3. Digital images exhibit high correlation between adjacent pixels. The signals of  $q\_out$  [9] and  $q\_out$  [0] are used for decoding. ADALP utilizes the complementary function of the subtractor for absolute differences. The values of Abs\_diff range from 0 to 255; however, of them more than 97% in average are distributed between 1 and 37.

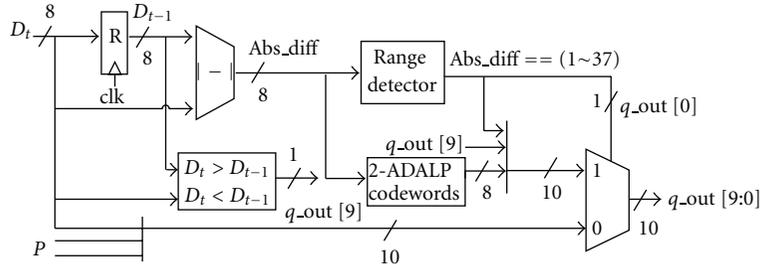


FIGURE 3: The encoder of the proposed ADALP architecture.

Assume that there are  $N$ -bit source words,  $i$  intraword transitions, and a 1-bit control signal. Then the total number of codewords is denoted as follows:

$$\text{the total number of codewords} = \sum \binom{N}{i}, \quad (1)$$

where  $i$  ranges from 0 to  $N - 1$ . For example, let  $N$  and  $i$  indicate 8 and 1, respectively. The 0 intra-word transition has 1 codeword. The 1 intra-word transition has 8 codewords and so does the 2 intra-word transition (i.e., the  $q\_out [9]$  signal is either 1 or 0 because the signal of  $q\_out [9]$  takes an absolute difference of two consecutive data). Hence, the proposed method uses the  $q\_out [9]$  to control the range of codeword; that is, only 9 codewords are used for ADALP, ranging from 1 to 9 (the minus sign is converted to the plus sign), instead of 18 codewords used for LIWT, ranging from 1 to 9 and  $-1$  to  $-9$ ). Thus, the total number of codewords is 9 (i.e.,  $1 + 8 = 9$ ). 1-ADALP encoding method is shown in Table 1. However, 1-bit control signal is not stored among codewords. Therefore, the codeword uses 72-bit registers (i.e.,  $9 \times 8$ -bit) so that the area and power of codec decrease.

*Example.* Suppose that the absolute difference values are 5 subpixels (i.e., 0,  $-1$ , 9, 5, and  $-2$ ) to test the proposed ADALP coding method.

Applying the 1-ADALP method, the transmitted sequence becomes: 0000000000, 0111111111, 1000011111, 1000000111, 0000000001 = 7 transition activities for both intra-word and inter-word transitions.

1-ADALP, 3-ADALP, or above is not used since 1-ADALP codeword has only 9 numbers, and 3-ADALP codewords or above require 93 numbers (i.e.,  $1 + 8 + 28 + 56 = 93$ ) or more only giving a mere improvement. The explanation is as follows. For instance, 1-ADALP codewords located in the range of [1, 9] cover about 80% of hit ratios. Let the scheme of 1-ADALP be compared with that of 2-ADALP, then the hit ratio of 1-ADALP decreases about 17%. The codewords of 3-ADALP located in the range of [1, 93] cover about 99.1% of hit ratios. Let the scheme of 3-ADALP be compared with that of 2-ADALP, then the improvement of the hit ratio for 3-ADALP only increases 2%, but the scheme of 3-ADALP must pay double hardware cost more than the scheme of 2-ADALP. Thus, the scheme of one, three, or above-ADALP does not fit the rule of cost effectiveness, so that the codewords of 2-ADALP are the proper method to be proposed.

TABLE 1: Intraword of 1-ADALP encoding method.

Value	1-ADALP	Value	1-ADALP
9	1.00001111.1	-9	0.00001111.1
8	1.00011111.1	-8	0.00011111.1
7	1.00000111.1	-7	0.00000111.1
6	1.00111111.1	-6	0.00111111.1
5	1.00000011.1	-5	0.00000011.1
4	1.01111111.1	-4	0.01111111.1
3	1.00000001.1	-3	0.00000001.1
2	1.00000000.1	-2	0.00000000.1
1	1.11111111.1	-1	0.11111111.1
0	0.00000000.0		

The codewords of 2-ADALP in Figure 3 use 8-bit source words, 2 intra-word transitions, and a 1-bit control signal. The 0 intra-word transition has 1 codeword. The 1 intra-word transition has 8 codewords. The 2 intra-word transitions have 28 codewords. Therefore, the total number of 2-ADALP codewords is 37 (i.e.,  $1 + 8 + 28 = 37$ ).  $P$  is defined by 2, and authors assume that  $P$  bits of redundancy are allowed. In other words, the proposed ADALP method uses 296-bit registers (i.e.,  $37 \times 8$ -bit) to store the codewords of 2-ADALP, located in the range of [1, 37] covering about 97% of hit ratios in average. The arrangement of the encoded output data is composed of the  $q\_out [0]$  signal, the encoded value, and the  $q\_out [9]$  signal. The encoded values (i.e.,  $q\_out [8:1]$ ), stored at the middle position of the encoded output stream, are obtained from either the source words or the codewords of 2-ADALP. The signal of  $q\_out [9]$  is stored at the foremost position of the encoded output data, while the signal of  $q\_out [0]$  is stored at the aftermost position. Then the encoded output stream is transmitted by the serial transmission. Figure 4 shows the proposed ADALP decoder architecture. If the signal of  $q\_out [0]$  is set to "High," the codewords of 2-ADALP are assigned to the Abs\_diff value; otherwise, the values of  $q\_out [8:1]$  are assigned to the value of Abs\_diff. Next, the decoder uses the signal of  $q\_out [9]$  to obtain the original bus data,  $D_t [0:7]$  at the output of the encoder. For instance, if the signal of  $q\_out [9]$  is set to "High," the value of  $D_{t-1} [0:7]$  adds the value of Abs\_diff, and then the sum is assigned to the value of  $D_t [0:7]$ ; otherwise, when the signal of  $q\_out [9]$  is set to "Low," the value of  $D_{t-1}$

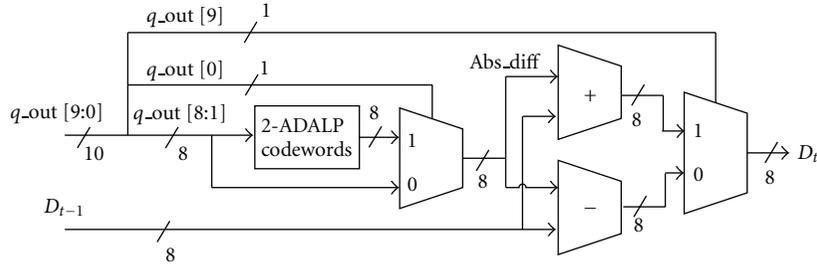


FIGURE 4: The decoder of the proposed ADALP architecture.

TABLE 2: Hit ratios for different bus methods.

Image	LIWT [3] hit %			Proposed hit %		
	R	G	B	R	G	B
Girl_1	95.8	95.8	94.8	98.6	98.4	98.4
Couple	94.3	95.3	95.1	97.8	98.1	98.1
Girl_2	97.7	98.0	98.0	98.8	99.0	99.0
Girl_3	94.5	93.8	94.8	96.9	96.5	97.2
House	97.0	95.6	95.8	98.6	97.4	97.5
Tree	89.2	86	87.4	95.4	93.1	94.6
Jelly Beans_1	96.6	95.6	97.9	98.4	97.5	98.9
Jelly Beans_2	95.5	93.7	95.8	98.1	96.8	97.6
Bird	96.5	96.6	96.6	97.9	97.9	97.9
Butterfly	88.9	90.2	89.9	93.2	94.3	93.8
Average	94.6	94.1	94.6	97.4	96.9	97.3

[0:7] subtracts the value of Abs\_diff, and then the difference is assigned to the value of  $D_t$  [0:7].

## 4. Experimental Results

In this section, images of 256 by 256 pixels from the SIPI database [12] applied the test data streams (i.e., the images of Girls, Couple, House, Tree, Jell Beans, Bird, and Butterfly) to ADALP, TMDS, and LIWT which are uncompressed and stored in the tagged image file format (TIFF). The results of the TMDS method are taken as a reference, and authors take them for a comparison with the ones of ADALP and LIWT. Moreover, some bus encoding performances are compared, such as the hit ratio and the transition activity. The codec area and the power dissipation are also evaluated.

**4.1. Comparison of Simulation Results on Transition Activity Reduction.** The hit ratio is inversely proportional to the transition activity; that is, reducing the transition activity will increase the hit ratios. The explanation is as follows: the LIWT method uses 2-LIWT codewords, ranging from  $-23$  to  $22$ , to reduce the transition activity. It has 46 codewords with 414 bits size (i.e.,  $46 \times 9$ ). Similarly, the proposed ADALP method also uses the codewords of 2-ADALP, ranging from 1 to 37 with 296 bits size (i.e.,  $37 \times 8$ ), to reduce the transition activity. Table 2 shows the hit ratios for two different encoding methods. Therefore, the larger the hit ratio is, the better the encoding method will be. The hit ratios of 3 color

channels of the proposed ADALP method are as follows: the hit ratio of the R color channel is between 93.2% and 98.8%; the hit ratio of the G color channel is between 93.1% and 99%; the hit ratio of the B color channel is between 93.8% and 99%. Moreover, the proposed ADALP method increases more hit ratios, about 2.8% in average, achieving higher hit ratios than the LIWT method does for each image.

To compare ADALP with the LIWT and with the TMDS methods in simulation, respectively, there are ten different test images generated to calculate the averages of the transition activity. Table 3 shows the reduction of the transition activity for two different encoding methods with respect to the TMDS method. The color channels of R, G, and B reduce the transition activity by up to 72.3%, 72%, and 74.9% by using the LIWT method and by up to 78.9%, 77.7%, and 80.3% by using the ADALP method, respectively. In comparison with the LIWT method, the transition activities of the color channels of R, G, and B of the proposed ADALP method, are reduced by up to 30.9%, 26.3%, and 21.7%, respectively, shown in Table 4, that is, by up to 13% in average. As a result, the proposed ADALP method reduces more dynamic power than the LIWT method does. In Tables 3 and 4, both intraword and interword transitions are considered in the simulations of transition activities. As to the DC balance, usually it is not a problem because the maximum cable length of most LCD displays should comply with the specification [3, 4]. In brief, the TMDS method takes the DC balance into consideration and provides the best mechanism to date. Its implementation is simply based on counting zeros and ones according to the image size. Similarly, LIWT and the proposed ADALP encoding methods also consider the DC balance. Comparing with TMDS, the color channels of R, G, and B which obtained the DC balance by using the LIWT method are 50.1%, 49.9%, and 50.2% and by using the ADALP method, are 49.4%, 49.4% and 50.1%, respectively.

**4.2. Comparison of Estimation Results on Total Power Reduction and Area Reduction of Bus Codec.** Bus encoding methods including area of codec and power may affect the total power dissipation and increase overheads. Table 5 shows the actual overheads of different bus encoder and decoder circuits. The codec circuits are modeled by using the Verilog HDL, synthesized by the Synopsys Design Compiler, and laid out by SOC Encounter with the UMC CMOS 90 nm technology, and the power dissipation is estimated by the PrimePower

TABLE 3: Transition activity reduction for different bus methods versus TMDS.

Image	LIWT [3] (reduction %, ones %)						Proposed (reduction %, ones %)					
	R	G	B	R	G	B	R	G	B	R	G	B
Girl_1	59.19	50.04	58.46	49.19	56.12	51.56	61.34	46.55	62.09	47.09	57.69	49.76
Couple	57.36	51.01	56.03	50.63	55.88	50.91	61.31	46.01	61.29	45.81	59.18	49.07
Girl_2	72.31	49.63	70.41	48.22	69.86	47.29	76.63	49.26	73.78	49.23	72.76	49.66
Girl_3	60.09	48.92	60.86	49.11	62.37	49.24	64.77	49.78	62.36	50.57	63.96	50.63
House	64.67	47.61	66.66	49.72	60.68	51.99	66.57	51.12	68.68	51.35	63.76	50.54
Tree	56.43	52.70	54.95	51.72	55.12	52.64	57.36	52.59	57.74	51.85	57.38	52.42
Jelly Beans_1	69.49	50.50	72.01	51.01	74.91	50.26	78.92	49.20	77.53	49.01	80.35	49.06
Jelly Beans_2	69.32	50.89	69.76	50.75	71.31	49.73	78.29	48.88	77.73	48.55	76.31	49.91
Bird	61.31	50.51	63.94	50.21	64.51	49.89	68.11	50.34	70.64	50.41	71.19	50.61
Butterfly	58.72	49.19	60.84	48.84	58.67	49.17	62.99	50.31	65.17	50.13	62.37	50.21
Average	62.89	50.10	63.39	49.94	62.94	50.27	67.63	49.40	67.70	49.47	66.50	50.19

It is noted that “reduction” means the transition activity reduction, and “ones” means the DC balance.

TABLE 4: Transition activity reduction for proposed method versus LIWT.

Image	Proposed (reduction %)		
	R	G	B
Girl_1	5.49	9.23	3.58
Couple	9.26	11.95	7.47
Girl_2	15.62	11.41	9.64
Girl_3	7.08	3.81	4.22
House	5.36	6.09	7.79
Tree	2.15	6.19	5.03
Jelly Beans_1	30.96	19.9	21.7
Jelly Beans_2	29.24	26.32	17.44
Bird	17.55	18.58	18.75
Butterfly	10.32	11.07	8.94
Average	13.30	12.46	10.46

TABLE 5: Area and power dissipation of different bus codecs.

Technology (nm)	Coding methods	Codecs (Encoder + Decoder)	
		Area, $\mu\text{m}^2$	Power, mW
UMC 90 nm	TMDS [1]	3642	0.314
	LIWT [3]	3284	0.236
	Proposed	3245	0.234

EDA tool with actual dynamic test vectors. The total power dissipation of the bus codec includes the dynamic power, the static power, and the short circuit power dissipation.  $V_{DD}$  is 1 Volt. The operation frequency is set to 55 MHz. The area of codec and the power dissipation of the bus codec with the proposed method of simulation are  $3245 \mu\text{m}^2$  and 0.234 mW, respectively. Table 5 indicates that the proposed method required less complexity circuits. It is noted that the bit sizes of the 2-LIWT codewords used in the LIWT method are larger than those of the 2-ADALP codewords used in the proposed ADALP method for the same coding range. Compared with the proposed ADALP bus codec, the hardware

overheads relative to the implementations of TMDS and LIWT are 10.9% and 1.2%, respectively.

The dynamic power dissipation on the coding bus is calculated as follows:

$$P_D = \alpha \times C \times V_{DD}^2 \times f, \quad (2)$$

where  $\alpha$  is the transition activity,  $C$  is the typical mutual capacitance,  $V_{DD}$  is the supplying voltage, and  $f$  is the clock frequency.

A DVI connection consists of one or two TMDS serial transmissions. The voltage swinging between the TMDS differential pairs is either from 0 to 780 mV or from  $-780$  to 0 mV. The maximum frequency is 55 MHz to support the pixel rate of 165 MHz. The length of the DVI cable is limited to 4.6 m. The typical mutual capacitance of high quality DVI cables is around 50 pF/m. With the TMDS method,  $\alpha$  is the average transition activity, which is normalized for comparisons, ranging from 0 to 1. The sources of the total power dissipation include the bus encoder, bus decoder, and dynamic power dissipation on the connection bus (i.e.,  $P_{enc} + P_{dec} + P_{D, coded}$ ). Figure 5 shows the result of the comparison of the total power dissipation for different bus encoding methods in three color channels with UMC 90 nm technology. In Figure 5, both intra-word and inter-word transitions are also considered in the estimations of total power consumption for different bus encoding methods. Comparing with TMDS, the color channels of R, G, and B reduce the total power dissipation by 62.4%, 62.7%, and 62.3%, respectively, if using the LIWT method and by 66.9%, 67%, and 65.8%, respectively, if using the ADALP method. Moreover, comparing with the LIWT method, the proposed ADALP method reduces the total power dissipation by about 4.5%.

## 5. Conclusion

In comparison with TMDS, simulation results show that the proposed ADALP encoding method reduces the transition activity by up to 67.7% and the total power by up to 67% with UMC 90 nm technology. Comparing with LIWT,

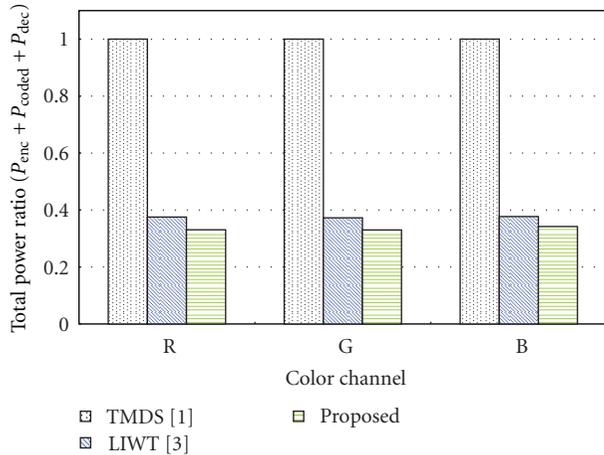


FIGURE 5: Comparison of total power dissipation for different bus encoding methods in three color channels.

the proposed ADALP method reduces the transition activity by about 13% and the total power by about 12% with the same technology.

However, the proposed method does not apply to the 1-ADALP codeword, 3-ADALP or above codewords because the 1-ADALP codeword only contains 9 numbers, the range is too narrow, and 3-ADALP or above codewords only contribute a bit, but cost twice or more price in hardware comparing to the 2-ADALP scheme. Thus, the 3-ADALP or above codewords are not worth developing.

In short, the proposed efficient bus encoding method uses the absolute difference value, needs fewer transition counts, and requires little architectural complexity. Comparing with TMDS and LIWT, it greatly reduces the dynamic and total power dissipation of the LCD digital display interfaces. However, when it comes to the DC balance, it makes slight concession to TMDS and LIWT. Thus, to get the DC highly balanced, reduce more transition activities, and diminish the area of codec, authors and successors need to make more efforts.

## Acknowledgments

This work is supported by the National Science Council, Taiwan, under Grant NSC101-2220-E-005-003. Authors would like to give special thanks to the National Chip Implementation Center (CIC) of Taiwan for providing the related EDA tools.

## References

- [1] DDWG, Digital Visual Interface, V1.0, <http://www.ddwg.org/>.
- [2] I. Choi, H. Shim, and N. Chang, "Low-power color TFT LCD display for hand-held embedded systems," in *Proceedings of the International Symposium on Low Power Electronics and Design*, pp. 112–117, August 2002.
- [3] S. Salerno, E. Macii, and M. Poncino, "Energy-efficient bus encoding for LCD digital display interfaces," *IEEE Transactions on Consumer Electronics*, vol. 51, no. 2, pp. 624–634, 2005.

- [4] W. C. Cheng and M. Pedram, "Chromatic encoding: a low power encoding technique for digital visual interface," *IEEE Transactions on Consumer Electronics*, vol. 50, no. 1, pp. 320–328, 2004.
- [5] W. W. Hsieh, P. Y. Chen, C. Y. Wang, and T. Hwang, "A bus encoding scheme for crosstalk elimination in high performance processor design," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 12, pp. 2222–2227, 2007.
- [6] C. P. Fan and C. H. Fang, "Efficient RC low-power bus encoding methods for crosstalk reduction," *Integration, The VLSI Journal*, vol. 44, no. 1, pp. 75–86, 2011.
- [7] R. B. Lin, "Inter-wire coupling reduction analysis of bus-invert coding," *IEEE Transactions on Circuits and Systems I*, vol. 55, no. 7, pp. 1911–1920, 2008.
- [8] Y. Shin, S. I. Chae, and K. Choi, "Partial bus-invert coding for power optimization of application-specific systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 9, no. 2, pp. 377–383, 2001.
- [9] C. P. Fan and C. H. Fang, "Low-power instruction address bus coding with xor-bits architecture," *Journal of Circuits, Systems and Computers*, vol. 18, no. 1, pp. 45–57, 2009.
- [10] Y. Aghaghiri, F. Fallah, and M. Pedram, "ALBORZ: address level bus power optimization," in *Proceedings of the Quality Electronic Design*, pp. 470–475, 2002.
- [11] *Intel PXA250 and PXA210 Application Processors Developer's Manual*, Intel, 2002.
- [12] A. G. Weber, "USC-SIPI Image Database Version 5," USC-SIPI Report #315, October 1997, <http://sipi.usc.edu/services/database/Database.html>.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

