

# 一种基于图像旋转显示的 DDR2 高效存储方法

刘承禹<sup>1,2</sup>, 田 泽<sup>1,2</sup>, 刘 浩<sup>1,2</sup>, 徐晓梅<sup>1,2</sup>

(1. 中国航空计算技术研究所, 陕西 西安 710068;

2. 集成电路与微系统设计航空科技重点实验室, 陕西 西安 710068)

**摘 要:** 图像旋转是一种常用的数字图像处理技术, 目前已广泛应用于各领域。在图像系统中, 通常采用 DDR2 作为系统的显示缓存。图像显示系统通常采取从左向右、从上到下的扫描刷新方式, 图像数据也以同样的方式进行存储, 而这种图像数据的常规存储方法在大分辨率图像旋转显示时难以满足系统对 DDR2 的带宽需求。为解决这一问题, 图像数据必须采用高效的存储方法。以图像旋转显示原理为依据, 对比常规存储方法, 重点阐述了分块存储方法及其效率, 最后结合项目进行 FPGA 原型验证。验证结果表明, 分块存储方法提高了 DDR2 的带宽利用率, 能够高效满足系统的旋转显示需求, 解决了常规存储方法难以满足大分辨率图像旋转显示的问题。

**关键词:** 图像旋转显示; 存储; 带宽利用率; 效率

**中图分类号:** TP39

**文献标识码:** A

**文章编号:** 1673-629X(2016)04-0016-04

doi: 10.3969/j.issn.1673-629X.2016.04.004

## An Efficient Storage Method of DDR2 Based on Image Rotation Display

LIU Cheng-yu<sup>1,2</sup>, TIAN Ze<sup>1,2</sup>, LIU Hao<sup>1,2</sup>, XU Xiao-mei<sup>1,2</sup>

(1. Aeronautics Computing Technique Research Institute, Xi'an 710068, China;

2. Key Lab of Aeronautics Science and Technology of Integrate Circuit and Micro-system Design,  
Xi'an 710068, China)

**Abstract:** Image rotation is a common digital image processing technology which is widely applied in various fields. In image system, DDR2 is always used as the display memory. Image display system scans to refresh images usually from left to right and from top to bottom, then saving image data in the same order, but for image rotation display of maximum resolution, this general storage method can't meet the requirement of DDR2 bandwidth for system. In order to solve this problem, the high-effective storage method must be used. On the basis of the principle of image rotation display, contrasting the general storage method, the block storage method and efficiency of it is discussed. At last, FPGA verification is made in combination of project. The result indicates this block storage method increases bandwidth utilization of DDR2, which can meet the requirement of rotation display, and solves the problem which general storage method can't meet maximum resolution image rotation display.

**Key words:** image rotation display; memory; bandwidth utilization; efficiency

## 0 引 言

图像旋转是一种常用的数字图像处理技术<sup>[1]</sup>, 目前已广泛应用于各领域。绝大多数显示系统采取从左至右、从上到下的扫描方式<sup>[2-3]</sup>, 图像数据也是以同样的方式进行存储。旋转显示能够使图像旋转 90°、180° 或 270° 后显示在 LCD 屏幕上<sup>[4-6]</sup>。

目前图像显示系统均采用 DDR2 (或 DDR3, 文中以 DDR2 为例, DDR3 原理类似) 作为系统显示缓

存<sup>[7]</sup>。由于图像数据在 DDR2 中的常规存储方式为从左至右、从上到下存储, 与显示系统的扫描方式一致, 因此, 在非旋转显示时, 系统对 DDR2 按行从左至右、从上到下读取图像数据, 此时 DDR2 的带宽利用率最高<sup>[8-9]</sup>。但是, 在 90° 或 270° 旋转显示时, 系统需要对 DDR2 按列读取图像数据, 此时 DDR2 的带宽利用率最低<sup>[10-11]</sup>。在大分辨率图像旋转显示时, 甚至无法满足系统需要。为了提高 DDR2 带宽利用率, 满足系统

收稿日期: 2015-06-16

修回日期: 2015-09-22

网络出版时间: 2016-03-22

基金项目: 中国航空工业集团创新基金 (2010BD63111)

作者简介: 刘承禹 (1982-), 男, 工程师, 研究方向为集成电路设计; 田 泽, 博士, 研究员, 中航首席技术专家, 研究方向为 SoC 设计、嵌入式系统设计、VLSI 设计。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20160322.1517.010.html>

对大分辨率图像的旋转显示需求,必须采用高效的 DDR2 存储方法。

文中结合某 GPU 具体项目实践,从提高 DDR2 带宽利用率角度出发,设计了一种高效的 DDR2 存储方法。该方法可以同时满足显示系统对大分辨率图像的正常和旋转显示需求。

1 图像旋转显示原理

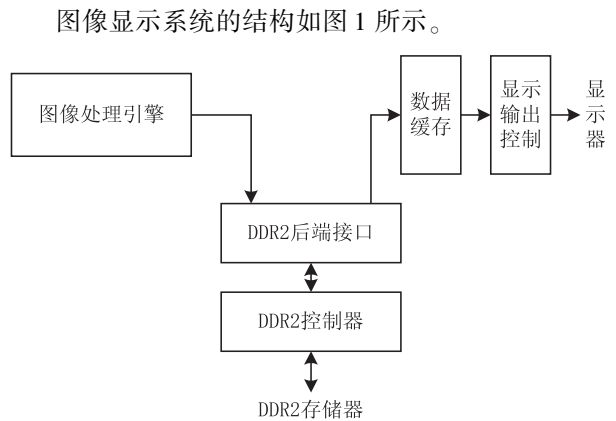


图 1 图像显示系统结构

系统以 DDR2 存储器为系统的显示缓存。图像处理引擎将绘制好的图像通过 DDR2 后端接口和 DDR2 控制器缓存在 DDR2 存储器中。图像显示输出时,DDR2 后端接口通过 DDR2 控制器从 DDR2 存储器中按行读取图像数据,并缓存在数据缓存模块中。数据缓存模块为双缓结构,可以缓存两行图像数据。显示输出控制按行进行数据处理,并且最终输出到显示器<sup>[12]</sup>。

图像显示时,显示器始终是以从左至右、从上到下的顺序按行扫描。图像数据在 DDR2 存储器中的常规存储方式也是以从左至右、从上到下的顺序按行缓存,与图像扫描顺次一致,如图 2 所示,即图像的每一行像素数据在 DDR2 按行存储。

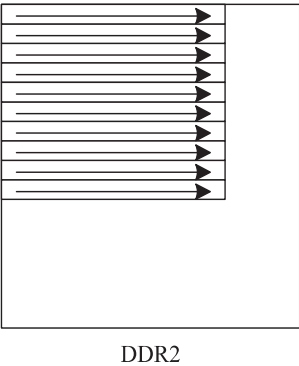


图 2 图像常规存储示意图

图像正常(0°旋转)显示时,显示系统在 DDR2 中以图 3(a)的顺序按行读取数据,即从左上角开始,一行内从左至右读取数据,对行的读取顺序为从上到下;

图像顺时针旋转 90°显示时,显示系统必须以图 3(b)的顺序按列读取数据,即从左下角开始,一列内从下到上,对列的读取顺序为从左至右;图像顺时针旋转 180°显示时,显示系统必须以图 3(c)的顺序按行读取数据,即从右下角开始,一行内从右至左,对行的读取顺序为从下到上;图像顺时针旋转 270°显示时,显示系统必须以图 3(d)的顺序按列读取数据,即从右上角开始,一列内从上到下,对列的读取顺序为从右至左。

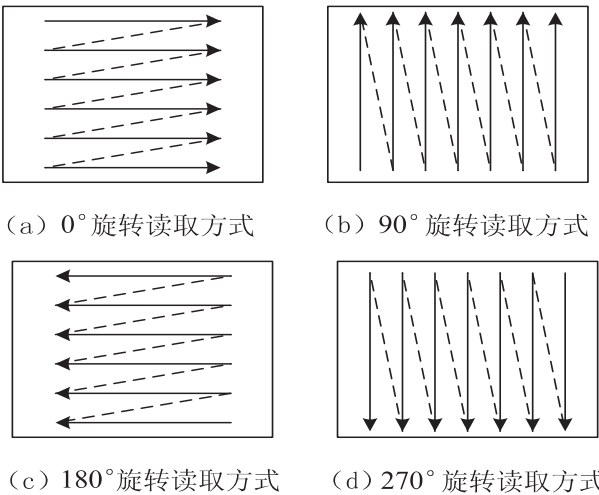


图 3 旋转显示时数据读取方式

2 DDR2 带宽利用率分析

从上节的描述可以看出,图像 0°旋转和 180°旋转显示时,显示系统对 DDR2 的读取方式为按行读取,图像 90°和 270°旋转显示时,显示系统对 DDR2 的读取方式为按列读取。基于这两种读取方式,DDR2 带宽利用率差别很大,其带宽利用率可用公式  $T_1/(T_1 + T_2)$  计算。其中,  $T_1$  为数据周期,  $T_2$  为空闲周期<sup>[13]</sup>。

根据 DDR2 的页面特性,按行读取时,显示系统一次处理一行图像像素数据,此行图像像素数据全在 DDR2 同一行地址内,因此读取此行数据时不需要切换页面,除刷新外,没有空闲周期,  $T_2$  可忽略不计,此时 DDR2 的带宽利用率可以达到 95% 以上。

按列读取时,显示系统一次处理一列图像像素数据,此列图像的每个像素数据均在 DDR2 的不同行地址内,因此读取此列数据时需要多次切换页面,每读取一次,需要关闭上一个页面,重新打开下一个页面,  $T_2$  大于  $T_1$ ,此时 DDR2 的带宽利用率很低,具体利用率如表 1 所示。

表 1 常规存储列读取 DDR2 带宽利用率

突发长度	$T_1$	$T_2$	DDR2 带宽利用率
4	2	13	约 13%
8	4	13	约 23%

从以上分析可以看出,如果按常规方式存储图像



宽利用率达到了 70% 至 80% 左右,有大幅提高。

表 2 分块存储旋转显示 DDR2 带宽利用率

旋转度数	突发长度	$T_1$	$T_2$	DDR2 带宽利用率
0°/180°	4	32	13	约 71%
	8	32	13	约 71%
90°/270°	4	32	13	约 71%
	8	64	13	约 83%

4 分块存储效率验证

在某 GPU 项目中,针对图像的旋转显示,采用了该分块存储方法。其 DDR2 的突发长度固定为 4,数据位宽为 64 位,时钟频率为 333 MHz,支持的最大分辨率为 1 920×1 440@ 60 Hz,每个像素有 32 位数据。

在图像 90°和 270°旋转显示时,系统对 DDR2 中所有地址块进行一次突发读操作将会读取 8 列像素数据,因此在 DDR2 后端接口中增加了 8 个缓存区,每个缓存区可以存储最大分辨率的一列像素数据。

在处理最大分辨率 1 920×1 440@ 60 Hz 时,系统扫描一行需要时间约 11.5 μs。在图像 0°和 180°旋转显示时,系统读取一次需  $1\,920 \times 32 / 64 \times 2 \times 71\% \times 333 = 2.1\,\mu\text{s}$ ,小于 11.5 μs。在图像 90°和 270°旋转显示时,系统读取一次需  $1\,440 \times 2 / 71\% \times 333 = 12.2\,\mu\text{s}$ ,由于一次读取了 8 列像素数据,可供系统扫描 8 行,12.2 μs 远小于  $8 \times 11.5\,\mu\text{s}$ 。

通过上述计算可以看出,采用分块存储,系统 DDR2 缓存效率能满足显示输出带宽需求,并给图像处理引擎的写、绘图等操作预留了足够的带宽。

在项目的实际 FPGA 验证中,采用常规存储时,大分辨率图像无法流畅地 90°和 270°旋转显示,而采用分块存储时,所有分辨率的图像均能流畅、平滑地正常及旋转显示。该对比验证的结果表明,分块存储方法能够解决常规存储难以满足大分辨率图像旋转显示的问题。

5 结束语

文中介绍了图像旋转显示原理和常规存储方法在

图像旋转显示时的低效率缺陷,提出了分块存储方法,重点阐述了其读取方式和效率,并结合项目进行了有效的 FPGA 原型验证。结果表明,该分块存储方法具有很好的效果,提高了 DDR2 的带宽利用率,能够高效满足系统的旋转显示需求,对有类似存储需求的设计具有一定的参考价值。

参考文献:

[1] Wolberg G. Digital image warping[M]. Los Alamitos California:IEEE Computer Society Press,1990:208–209.

[2] 陈 芳. 一种基于错切原理的图像旋转方法[J]. 淮阴师范学院学报:自然科学版,2004,3(4):319–322.

[3] 吴恩华,柳有权. 基于图形处理器(GPU)的通用计算[J]. 计算机辅助设计与图形学学报,2004,16(5):601–612.

[4] Danielsson P E, Hammerin M. High accuracy rotation of images[J]. Graphical Models and Image Processing, 1992, 54(4):340–344.

[5] Luna F D. Introduction to 3D game programming with DirectX 9.0[M]. Texas Plano:Wordware Publishing Inc,2003.

[6] 曹厚德,刘剑波. 一种基于 FPGA 的图像旋转实现技术[J]. 计算机应用与软件,2011,28(7):89–91.

[7] 叶至军. Visual C++/DirectX9 3D 游戏开发导引[M]. 北京:人民邮电出版社,2006.

[8] 汪国有,项国平,孙天春. 利用 FFT 实现图像的快速高质量旋转变换[J]. 华中科技大学学报:自然科学版,2003,31(4):91–93.

[9] 李洪伟. 图像跟踪器中视频处理系统的设计与电子消像旋的实现[D]. 成都:电子科技大学,2006.

[10] Gray K. Microsoft direct X9 programmable graphics pipeline[M]. Washington Redmond:Microsoft Press,2003.

[11] Microsoft. Microsoft DirectX SDK documents[EB/OL]. [2000–04–15]. <http://www.microsoft.com>.

[12] 石 慎,张艳宁,郝润平,等. 基于 Bresenham 画线算法的图像快速高精度旋转算法[J]. 计算机辅助设计与图形学学报,2007,19(11):1387–1392.

[13] 李开宇,张焕春,经亚枝. 基于 FPGA 的高速高质量图像旋转[J]. 中国图象图形学报,2004,9(3):285–289.

[14] 汪国有,王岳环,张天序. 大转角下高质量图像旋转快速算法[J]. 华中科技大学学报,1999,27(6):45–47.