

动态参数模型确定SoC中异步FIFO深度的方法

王 剑^{1,2}, 王 宏¹, 杨志家¹

(1. 中国科学院沈阳自动化研究所 沈阳 110016; 2. 中国科学院研究生院 朝阳区 北京 100039)

【摘要】针对超大规模集成电路和片上系统设计中确定异步FIFO浓度的问题,根据异步FIFO运行时的属性提出FIFO动态参数模型,该模型包括FIFO饱和度、写入端和读出端数据传输率及上溢/下溢频率。在该模型的基础之上,分析异步FIFO的深度与动态参数之间的关系,采用功能仿真方法确定片上系统中异步模块之间数据传输所需FIFO的深度。对典型实例的分析表明,采用这种方法能够在保证系统数据通信性能的前提下,获得最小的FIFO深度,优化系统资源的使用。

关键词 异步FIFO; FIFO深度; 片上系统; 超大规模集成电路

中图分类号 TN47

文献标识码 A

doi.10.3969/j.issn.1001-0548.2009.03.031

New Method for Determining the Depth of Asynchronous FIFO in SoC Based on the Model of Dynamic Parameters

WANG Jian^{1,2}, WANG Hong¹, and YANG Zhi-jia¹

(1. Shenyang Institute of Automation, Chinese Academy of Sciences Shenyang 110016;

2. Graduate School, Chinese Academy of Sciences Chaoyang Beijing 100039)

Abstract Aiming at determining the depth of asynchronous first-in first-out (FIFO) in very large scale integrated (VLSI) circuits and system on chip (SoC) a model of dynamic parameters based on the running-time attribute of the asynchronous FIFO is presented in this paper. This model contains FIFO saturation, writing reading rate and overflow/underflow rate through the function simulation. On the base of this model, the depth of asynchronous FIFO in system on chip (SoC) can be determined by analyzing the relationship between these dynamic parameters and the depth of FIFO through the function simulation. According to the typical example provided, using the presented method can effectively minimize the depth of FIFO and optimize the usage of resource.

Key words asynchronous FIFO; depth of FIFO; system on chip; VLSI circuits

在多时钟域的片上系统(SoC)设计中,普遍使用异步FIFO解决跨时钟域的数据传输问题^[1-2]。使用异步FIFO可以在不同的时钟域之间快速地传输数据,实现数据同步和数据宽度匹配,保证数据传输的稳定性^[3-4]。

异步FIFO的一个重要属性是它的深度,即FIFO最多可以存储的以位为单位的数据的个数,它决定了数据传输的延时和吞吐量,是影响系统性能的一个重要参数。给系统中的异步FIFO设定适当的深度,既可以满足系统的性能需求,也可以优化系统的面积和功耗,因此受到人们的关注。文献[5]提出了计算FIFO深度的经验公式,但是只考虑了理想情况下读写操作等周期间隔发生的情况,很难应用于实际中。文献[6]发明了一种通过周期测量FIFO中存储数据的数量,来避免FIFO溢出的方法,适用于读写速率成比例的应用。文献[7]描述了一种缓冲约束算法,

计算应用了最小FIFO深度,但其只面向特定应用,约束了该算法的使用。文献[8]提出了一种基于排队论的FIFO模型,将随机过程理论引入到FIFO模型建立中,并根据该模型对FIFO的深度进行了研究。但是随机过程只能近似地表示实际应用中FIFO的读写操作序列,对FIFO深度的评估存在误差,所以应用性不强。

本文提出了一种通过分析异步FIFO动态参数,来确定其深度的方法。异步FIFO的动态参数包括FIFO的饱和度、读写速率和溢出情况等,它们反映FIFO的运行状况并且对确定FIFO深度有指导意义。设计者可以根据这些动态参数,分析FIFO深度对系统性能的影响,为异步FIFO确定适当的深度。

1 异步FIFO动态参数

异步FIFO的动态参数反映了FIFO运行时的状

态特性,其中包括FIFO的饱和度、FIFO写入端和读出端的数据传输速率以及出现上溢和下溢的频率,这些动态参数对FIFO深度的确定起指导作用。

1.1 饱和度

FIFO的饱和度定义为某一时刻FIFO中已存储数据的个数相对于FIFO深度的百分比。FIFO的饱和度是瞬时值,反映了当前FIFO中存储单元的使用率,其值为0表示FIFO中没有数据,值为100%表示FIFO已满,存储单元的使用达到饱和。在FIFO运行过程中,如果其饱和度一直很低,则表明FIFO的深度值过大,很多存储单元被浪费了;如果频繁地出现饱和状态,则表明FIFO的深度值过小,存储单元使用紧张。

1.2 写入端和读出端的传输速率

写入端和读出端传输速率分别定义为单位时间内向FIFO中写入数据的字节数和从FIFO中读出数据的字节数。当FIFO的数据宽度给定时,写入端和读出端传输速率描述了发送者和接收者对FIFO进行有效写操作和读操作的频率,因此用单位周期内有效写操作和读操作的次数来分别表示写入端传输速率和读出端传输速率。由于一个周期内,发送者和接收者最多只能执行一次写操作和读操作,所以它们的取值在[0,1]区间。

写入端和读出端的传输速率反映了FIFO所处的工作环境,利用它们,可以推导出当前环境下FIFO的吞吐量。

1.3 上溢频率和下溢频率

当FIFO已满时,若发送端仍然向FIFO中写数据,则FIFO发生上溢;反之,当FIFO为空时,接收端从FIFO中读数据,则发生下溢。上溢频率和下溢频率定义为单位时间内出现上溢和下溢的次数。

FIFO出现上溢,则表示数据流的上游数据传输较快,FIFO中没有多余的空间存储即将到来的数据,这样可能会影响上游的处理性能或造成数据丢失;而FIFO出现下溢,则表明数据流的下游数据传输较快,FIFO中没有数据供下游处理,这样会影响下游的处理性能。单纯增加FIFO的深度,可能会避免上溢的发生,而无法避免下溢的发生。

2 动态参数模型

异步FIFO的接口信号反映了FIFO所处的工作环境,动态参数模型就是在接口信号与动态参数之间建立映射关系,通过对接口信号的分析,可以获得FIFO的动态参数信息。因此,本文先对异步FIFO

的接口信号简要描述,然后介绍FIFO动态参数模型的建立。

2.1 异步FIFO的接口信号

图1给出了异步FIFO的接口信号。 wr_clk 为写时钟, wr_en 为写使能信号。在 wr_clk 的上升沿,当 wr_en 为高时, $data_in$ 中的数据被写入FIFO中。 rd_clk 为读时钟, rd_en 为读使能信号。在 rd_clk 的上升沿,当 rd_en 为高时,FIFO中的相应数据被读出,读出的数据由 $data_out$ 输出。信号 $full$ 为高时,表示FIFO已满,此时FIFO不会接受外部写入的数据;信号 $empty$ 为高时,表示FIFO已空,此时FIFO不响应外部的读操作。为了确保FIFO操作正确,添加一个可配置的保留空间,当FIFO中的剩余空间小于等于设置的保留空间时,使能 wr_hold 信号,指示发送者停止向FIFO发送数据。 rst_n 为全局异步复位信号,低有效,FIFO复位后, $empty$ 为高, $full$ 和 wr_hold 为低。

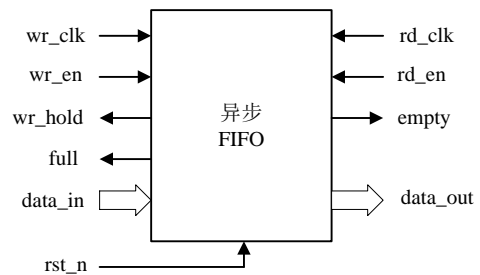


图1 异步FIFO的接口信号

2.2 动态参数建模

对变量符号进行约定: $wr_en(t)$ 和 $full(t)$ 表示第 t 个写时钟有效时,信号 wr_en 和 $full$ 的值; $rd_en(t)$ 和 $empty(t)$ 表示第 t 个读时钟有效时,信号 rd_en 和 $empty$ 的值。定义事件变量 $vaild_wr$ 、 $over_wr$ 、 $vaild_rd$ 和 $under_rd$ 分别表示对FIFO有效的写操作、上溢时写操作、有效读操作和下溢时读操作。当事件变量的值为1时,表示该事件发生,否则该事件未发生。

事件变量的计算公式为:

第 t 个写时钟有效时:

$$vaild_wr(t)=wr_en(t) \& (\sim full(t))$$

$$over_wr(t)=wr_en(t) \& full(t)$$

第 t 个读时钟有效时:

$$vaild_rd(t)=rd_en(t) \& (\sim empty(t))$$

$$under_rd(t)=rd_en(t) \& empty(t)$$

这样,在给出的FIFO动态参数的定义下,就可以对FIFO动态参数进行建模。

2.2.1 饱和度建模

FIFO运行到 T_1 时刻的饱和度为:

$$\text{saturation}(T_1) = \frac{(\text{有效写入次数} - \text{有效读出次数})}{\text{FIFO深度}} \times 100\% = \frac{\left(\sum_{t=1}^n \text{valid_wr}(t) - \sum_{t=1}^m \text{valid_rd}(t) \right)}{\text{depth}} \times 100\%$$

式中 n 和 m 分别表示FIFO运行到 T_1 时刻写时钟和读时钟的周期数; depth 为FIFO的深度。

2.2.2 写入端传输速率和上溢频率的建模

在 $k \sim k+n$ 个写周期内, 写入端传输速率和上溢频率的计算公式为:

$$\text{wr_rate}(k, k+n) = \frac{\sum_{t=k}^{k+n} \text{valid_wr}(t)}{n}$$

$$\text{of_rate}(k, k+n) = \frac{\sum_{t=k}^{k+n} \text{over_wr}(t)}{n}$$

2.2.3 读出端传输速率和下溢频率的建模

在 $j \sim j+m$ 个读周期内, 读出端传输速率和下溢频率的计算公式为:

$$\text{rd_rate}(j, j+m) = \frac{\sum_{t=j}^{j+m} \text{valid_rd}(t)}{m}$$

$$\text{uf_rate}(j, j+m) = \frac{\sum_{t=j}^{j+m} \text{under_rd}(t)}{m}$$

2.3 仿真平台的设计

本文将上述FIFO的动态参数模型集成到SoC仿真平台中, 该仿真平台用于分析SoC中各IP核之间互连结构的数据通信性能, 使用VHDL语言实现。在整个仿真过程中, 异步FIFO的动态参数将被自动记录下来, 设计者可以通过分析FIFO的动态参数, 了解FIFO深度对整个系统数据通信性能的影响, 为每个异步FIFO设定合适的深度。

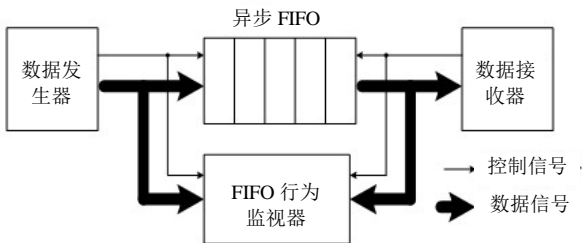


图2 异步FIFO周围的模块连接结构

图2给出了该仿真平台中异步FIFO周围的模块连接结构。平台中的异步FIFO借鉴文献[9-10]中的设计, 支持完全异步的读写时钟。数据发生器和数据接收器模拟FIFO发送端和接收端的数据传输特性, 生成对异步FIFO的写操作序列和读操作序列。设计

者可以使用激励文件来描述异步FIFO发送端和接收端的数据传输特性, 通过读入描述异步FIFO发送端和接收端的数据传输特性的激励文件, 生成对异步FIFO的读写操作序列。因此, 只需改变激励文件, 就可以使该平台用于不同的应用环境中。此外, 平台中还提供了随机模型, 可以生成符合均匀分布、正态分布、泊松分布和指数分布的读写序列。

FIFO行为监视器中实现了对FIFO动态参数的提取, 它对FIFO的接口信号进行监测, 实时统计FIFO的饱和度、写入端和读出端的数据传输速率以及上溢频率和下溢频率, 记录FIFO的运行信息, 并将统计和记录结果存储到文件中。

3 异步FIFO深度的确定

本文给出一个使用该方法确定FIFO深度的实例, FIFO所处的工作环境如图3所示。图中的虚线将整个系统分为2个时钟域, 左边的时钟工作频率为200 MHz, 右边的频率为100 MHz。箭头方向表示数据流的方向, FIFO的数据宽度为32 bit, 并将32 bit宽的数据定义为一个字。

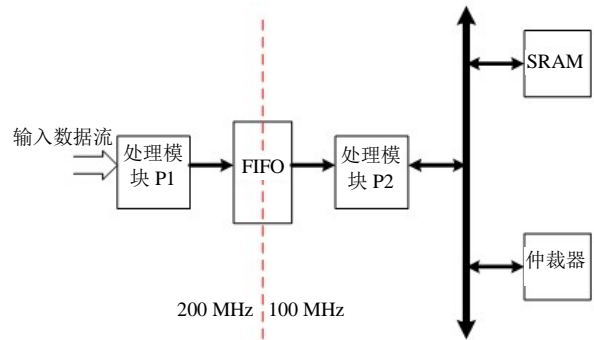


图3 FIFO所处的工作环境

假设处理模块P1对其输入端的数据流进行处理, 将相应处理结果存入FIFO中。输入数据流中相邻两数据帧到达的时间间隔服从泊松分布, P1对每个数据帧处理后, 将32个字写入FIFO中。因此, 在FIFO的写入端执行32个字的burst写操作, 相邻两次burst操作的时间间隔服从泊松分布。处理模块P2将FIFO中的数据读出, 并通过总线存储到SRAM中供总线上其他模块处理。P2对总线的占用率由总线仲裁器控制, 本例中采用基于分时复用的策略, 设P2的总线占用率为32%, 一次总线操作向SRAM中连续写入16个字, 因此, FIFO读出端执行间隔为50个时钟周期的burst读

操作,一次burst操作读出16个字。本例所述的数据传输机制,是目前SOC采用的与外界环境之间进行数据传递的主要方法,对指导实际应用有普遍意义。

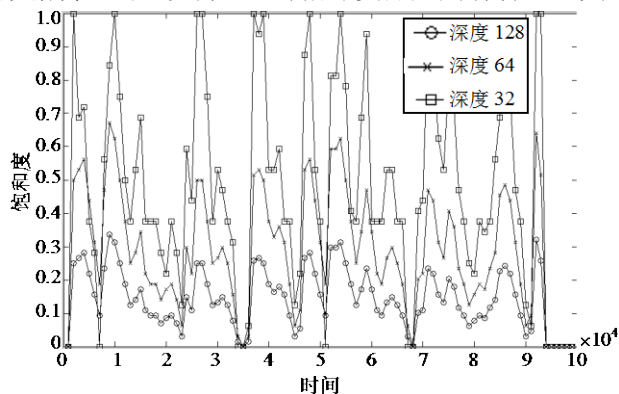


图4 不同深度下, FIFO饱和度的变化情况

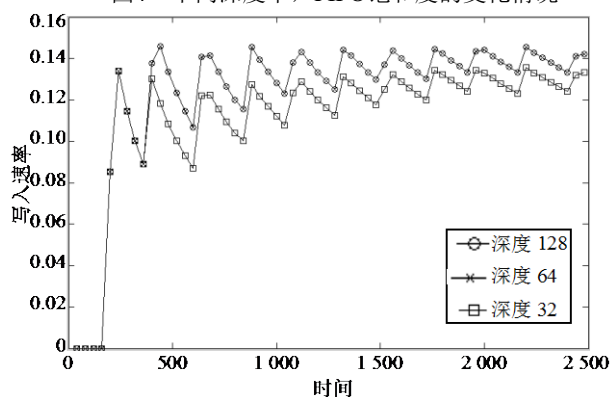


图5 不同深度下, FIFO写入速率的变化情况

为上述应用构建仿真平台,进行仿真,通过对仿真结果的分析确定应用中FIFO的深度。首先考虑FIFO深度对其饱和度的影响。图4描述了不同深度下, FIFO饱和度的变化情况。从图中可以看出,对于特定的FIFO深度,饱和度呈现burst变化,这正确反映了应用中FIFO写入端和读出端的数据传输特性。对于不同的FIFO深度,饱和度的峰值不同。当FIFO深度为32时,饱和度的峰值为1,这表明在FIFO运行过程中可能发生了上溢,因此会出现写操作的丢失,不符合应用的需求。当FIFO深度为64和128时,饱和度的峰值分别为0.67和0.34,表明此时FIFO可以满足写入端和输出端的数据传递,但是深度为128时, FIFO的使用率较低。

其次观察FIFO写入速率与深度之间的关系。图5给出了不同深度下, FIFO写入速率的变化情况。根据图中所示,当运行稳定后, FIFO的写入速率在一个稳定值附近上下波动。由于深度为32时,发生了写操作丢失,所以写入速率的均值比深度为64和128时写入速率的均值低;而深度为64和128时, FIFO的写入速率变化完全一致,这表明FIFO深度足够大时,其对写入速率就没有影响了。

通过上述分析可知,当FIFO深度设为64时,既可以满足应用的要求,也可以使FIFO自身有较高的使用率,所以该应用中FIFO的深度设为64较为合适。

4 结论

通过对异步FIFO动态参数的建模,可以很方便地获得FIFO运行过程中的状态信息,这些信息与FIFO深度密切相关。采用分析异步FIFO动态参数的方法,能够准确地掌握系统中FIFO的使用情况,进而为FIFO确定适当的深度。

参考文献

- [1] DALLY W J, POULTON J W. Digital systems engineering [M]. Cambridge, UK: Cambridge University. Press, 1998.
- [2] BALCH M. Complete digital design[M]. New York: McGraw-Hill, 2003.
- [3] APPERSON R W. A dual-clock FIFO for the reliable transfer of high-throughput data between unrelated clock domains[D]. California: UC Davis, 2004.
- [4] 鲁玲. 多时钟域数据传递的FPGA实现 [J]. 现代电子技术, 2007, 30(21): 130-132.
Lu Ling. An implementation of communicating data between multi-clock domain based on FPGA[J]. Modern Electronic Technique, 2007, 30(21): 130-132.
- [5] 陈征. FIFO缓冲存储器的结构及应用[J]. 汕头大学学报(自然科学版),1998, 13(1): 85-89.
CHEN Zheng. FIFO memory buffers: structure and applications[J]. Journal of Shantou University(Natural Science Edition), 1998, 13(01): 85-89.
- [6] COCHRAN A J, BAILEY P N, CARR L S. FIFO buffer depth estimation for asynchronous gapped payloads: United States Patent. US, 7227876B1 [P]. 2007-06-05.
- [7] RHA Kyoungseok, CHOI Kiyong. Area-efficient buffer binding based on a novel two-port FIFO structure[C]// Proceedings of the Ninth International Symposium on Hardware/Software Codesign. Copenhagen: ACM, 2001: 122-127.
- [8] 宋宇鲲, 王锐, 胡永华, 等. 使用排队论模型对FIFO深度的研究[J]. 仪器仪表学报, 2006, 27(增刊): 2485-2487.
SONG Yu-kun, WANG Rui, HU Yong-hua, et al. Study's depth based on queue-theory model[J]. Chinese Journal of Scientific Instrument, 2006, 27(6): 2485-2487.
- [9] APPERSON R W, YU Z, MEEUWSEN M J, et al. A scalable dual-clock FIFO for data transfers between arbitrary and haltible clock domains[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2007, 15(10): 1125-1134.
- [10] 刘祥远, 陈书明. 一种高性能的异步FIFO结构[J]. 电子学报, 2007, 35(11): 2098-2104.
LIU Xiang-yuan, CHEN Shu-ming. A high-performance asynchronous FIFO architecture[J]. Acta Electronica Sinica, 2007, 35(11): 2098-2104.

编辑 张俊