DMA Controller Using ispLSI® 6192SM

Introduction

This application note outlines the design of a generic four-channel priority encoded DMA controller with a separate on-board memory block using the ispLSI 6192SM from Lattice Semiconductor. The interface to the DMAC consists of a DMA selection line, a 16-bit address bus, a 16-bit data bus, channel selection signals, DMA channel request signals, DMA channel grant signals and a system clock.

DMA Controller Basics

To improve Central Processing Unit (CPU) throughput, microprocessor-controlled systems may employ Direct Memory Access (DMA) to relieve the CPU of bus cycles that would otherwise be taken up with data transfer between I/O devices and memory. In a bus organized system, a DMA controller (DMAC) can reside on the system bus and perform memory transfers in place of the CPU. After the CPU passes the address and block length of data to be transferred, the DMAC takes control of the system data and address busses. The DMAC then transfers data to and from the peripheral I/O devices directly to the memory, freeing the CPU to continue processing other operations. A diagram of a system employing DMA control is shown in Figure 1.

Details of the DMAC Implementation

The ispLSI 6192SM includes a 4000-bit single-port memory module. The memory module interface consists of an eight-bit address bus that is multiplexed with the data bus. The memory block incorporates parity-error checking and signals the CPU by asserting the PARITY_ERR pin when a memory parity error has occurred.

The DMA controller takes advantage of the on-board Register/Counter block of the ispLSI 6192SM for storing word count and starting address information. The Register/Counter block is configured as a four-bank parallel load-up counter with a four-bank register file. The use of this internal block for storing the starting address and word count information simplifies the DMAC design and frees up logic in the ispLSI 6192SM for other uses.

Functionally, the DMA controller consists of channel control logic and a DMA arbitration state machine. The channel control logic contains an address register, a word count register and an address count register for each DMA channel. The DMA arbitration state machine interfaces to I/O devices through the DMA channel request and DMA channel grant signals. Priority is fixed by channel order, giving channel 1 the highest priority and channel 4 the lowest. The memory block interfaces with the CPU through separate address lines and the system data bus. Parity errors are reported through a memory parity error signal. Figure 2 is a functional block diagram of the DMA controller with on-board SRAM showing the use of the Counter/Register block and the memory block of the ispLSI 6192SM.

DMA operations are initialized by the I/O device, sending an interrupt signal to the CPU to indicate that a data transfer is needed from memory. The CPU then loads the address and word count into the DMA controller for the memory locations to be accessed. After releasing mastership of the data bus, the CPU may continue with other processing operations.

Figure 1. Direct Memory Address System Block Design

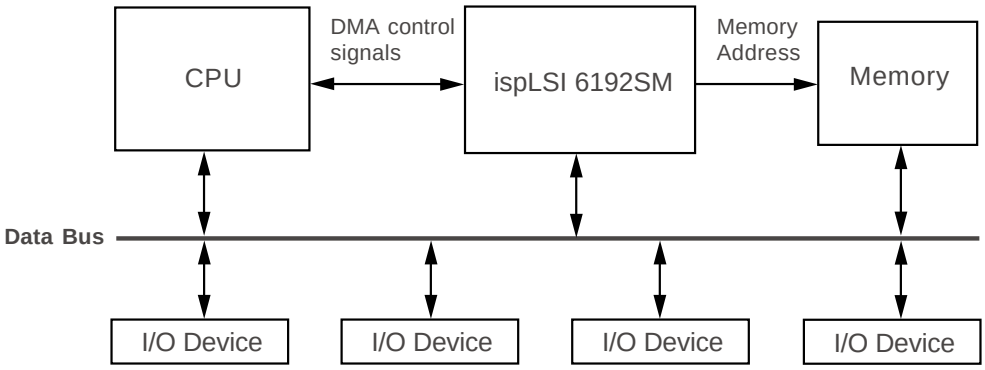
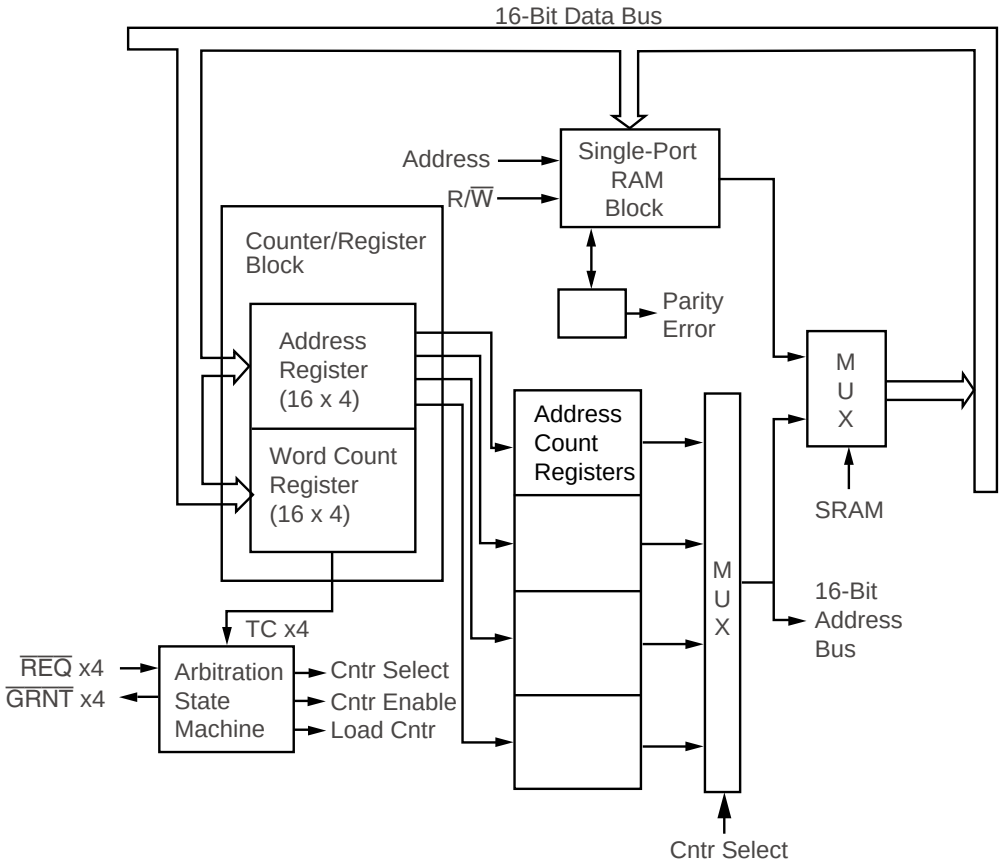


Figure 2. Functional Block Diagram of DMA Controller



After the CPU has loaded the address and word count for a particular DMA channel into the DMA controller, it is ready for I/O handshake signals. The I/O device then asserts its DMA request ($\overline{REQ}$) signal. If the DMA arbitration state machine (Figure 3) is in the Idle state and the I/O device has the highest priority of the asserted request signals, at the next clock cycle it transitions to the Load state for that channel. This state loads the address into the memory address count register and sets the output multiplexers for the correct channel. At the next clock cycle the state machine transitions to the Enable state for the active channel and the Grant ($\overline{GRNT}$) signal for that channel is asserted to indicate to the I/O device that data transfer has begun. The DMA controller then puts the starting address on the address bus and increments the address with every following system clock cycle until the Terminal Count (TC) signal indicates that the final address has been reached. The DMA controller de-asserts the $\overline{GRNT}$ signal to indicate the end of data transfer to the I/O device. The I/O device signals the CPU that transfer is complete with interrupt signals. When the DMA arbitration state machine returns to the Idle state, it is ready to respond to the next DMA channel request. DMA channel waveforms for a data transfer are shown in Figure 4. For device timing information consult the 1996 Lattice Semiconductor Data Book.

In a system without DMA control, the microprocessor must update the memory address for each word that is to be written to or read from memory. This generally takes about six clock cycles per word to be transferred for standard microprocessors. Using DMA control, the CPU uses only six clock cycles for set-up to transfer an entire block of data.

The four-channel, 16-bit DMA controller with memory block in the ispLSI 6192 uses only 56 of the 96 available I/Os in the device. This includes all interface signals to the Single Port RAM block which may also be brought out on separate dedicated pins, saving eight I/O pins. The register and counter block is used to keep track of the starting address and word count. The DMA controller

Figure 3. DMA Arbitration State Machine

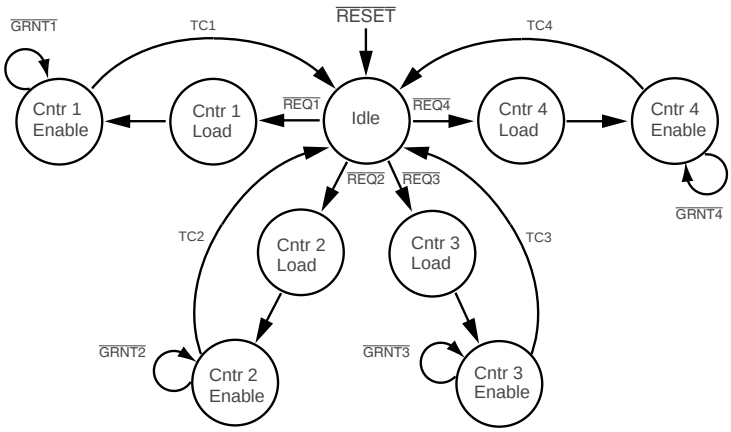
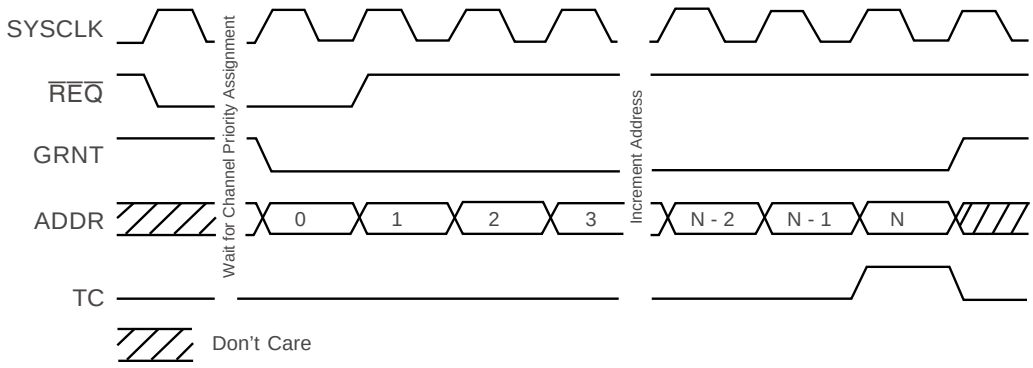


Figure 4. DMA Channel Waveforms for a Data Transfer



logic uses only six of the available 24 twin GLBs in the ispLSI 6192SM. This allows flexibility for implementing other system logic functions in the device in addition to the DMA controller logic.

DMA control can relieve CPU clock cycles that would otherwise be taken up by memory transfer functions, significantly increasing system performance. Because of its in-system programmability, $E^2CMOS^{®}$ flexibility and device functionality, the ispLSI 6192SM is ideally suited for DMA applications. On-board Register/Counter and memory blocks simplify the design cycle and minimize chip count and board space. Using the ispLSI 6192SM from Lattice Semiconductor for memory management maximizes board flexibility and minimizes design effort.

Summary

By taking advantage of the availability of the blocks within the ispLSI6192SM, DMAC implementation makes the most efficient use of block functions. The use of the blocks and the 25 percent of the PLD portion of the device accomplished a custom DMA controller with 75 percent of the PLD left to implement other logic functions within the device.

Source File

The source file for this design is available in text format (file name: an8004.txt) on the ISP Encyclopedia CD-ROM and the Lattice web site. Please note that if you copy or download this file to another directory, this link will not work.

Copyright © 1996 Lattice Semiconductor Corporation.

E^{2}CMOS, GAL, ispGAL, ispLSI, pLSI, pDS, Silicon Forest, UltraMOS, Lattice Logo, L with Lattice Semiconductor Corp. and L (Stylized) are registered trademarks of Lattice Semiconductor Corporation (LSC). The LSC Logo, Generic Array Logic, In-System Programmability, In-System Programmable, ISP, ispATE, ispCODE, ispDOWNLOAD, ispGDS, ispStarter, ispSTREAM, ispTEST, ispTURBO, Latch-Lock, pDS+, RFT, Total ISP and Twin GLB are trademarks of Lattice Semiconductor Corporation. ISP is a service mark of Lattice Semiconductor Corporation. All brand names or product names mentioned are trademarks or registered trademarks of their respective holders.

Lattice Semiconductor Corporation (LSC) products are made under one or more of the following U.S. and international patents: 4,761,768 US, 4,766,569 US, 4,833,646 US, 4,852,044 US, 4,855,954 US, 4,879,688 US, 4,887,239 US, 4,896,296 US, 5,130,574 US, 5,138,198 US, 5,162,679 US, 5,191,243 US, 5,204,556 US, 5,231,315 US, 5,231,316 US, 5,237,218 US, 5,245,226 US, 5,251,169 US, 5,272,666 US, 5,281,906 US, 5,295,095 US, 5,329,179 US, 5,331,590 US, 5,336,951 US, 5,353,246 US, 5,357,156 US, 5,359,573 US, 5,394,033 US, 5,394,037 US, 5,404,055 US, 5,418,390 US, 5,493,205 US, 0194091 EP, 0196771B1 EP, 0267271 EP, 0196771 UK, 0194091 GB, 0196771 WG, P3686070.0-08 WG. LSC does not represent that products described herein are free from patent infringement or from any third-party right.

The specifications and information herein are subject to change without notice. Lattice Semiconductor Corporation (LSC) reserves the right to discontinue any product or service without notice and assumes no obligation to correct any errors contained herein or to advise any user of this document of any correction if such be made. LSC recommends its customers obtain the latest version of the relevant information to establish, before ordering, that the information being relied upon is current.

LSC warrants performance of its products to current and applicable specifications in accordance with LSC's standard warranty. Testing and other quality control procedures are performed to the extent LSC deems necessary. Specific testing of all parameters of each product is not necessarily performed, unless mandated by government requirements.

LSC assumes no liability for applications assistance, customer's product design, software performance, or infringements of patents or services arising from the use of the products and services described herein.

LSC products are not authorized for use in life-support applications, devices or systems. Inclusion of LSC products in such applications is prohibited.

LATTICE SEMICONDUCTOR CORPORATION
5555 Northeast Moore Court
Hillsboro, Oregon 97124 U.S.A.
Tel.: (503) 681-0118
FAX: (503) 681-3037
http://www.latticesemi.com

November 1996