

**Generalized RAD Module Interface Specification  
of the Field-programmable Port eXtender (FPX)  
Version 2.0  
(This document contains only the SDRAM controller interface)**

David E. Taylor, John W. Lockwood, Sarang Dharmapurikar

WUCS-TM-01-15

July 5, 2001

Department of Computer Science  
Applied Research Lab  
Washington University  
Campus Box 1045  
One Brookings Drive  
Saint Louis, MO 63130

Abstract

The Field-programmable Port eXtender (FPX) provides dynamic, fast, and flexible mechanisms to process data streams at the ports of the Washington University Gigabit Switch (WUGS-20). By performing all computations in FPGA hardware, cells and packets can be processed at the full line speed of the transmission interface, currently 2.4 Gbits/sec. In order to design and implement portable hardware modules for the Reprogrammable Application Device (RAD) on the FPX board, all modules should conform to a standard interface. This standard interface specifies how modules receive and transmit ATM cells of data flows, prevent data loss during reconfiguration, and access off-chip memory. Module designers should conform to the standard I/O signal names and take special note of timing diagram references.

Supported by: NSF ANI-0096052

# 1 SDRAM controller interface

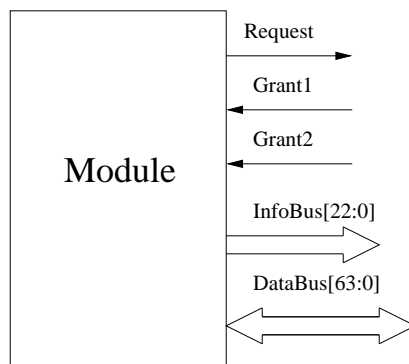


Figure 1: Module Interface for the SDRAM access

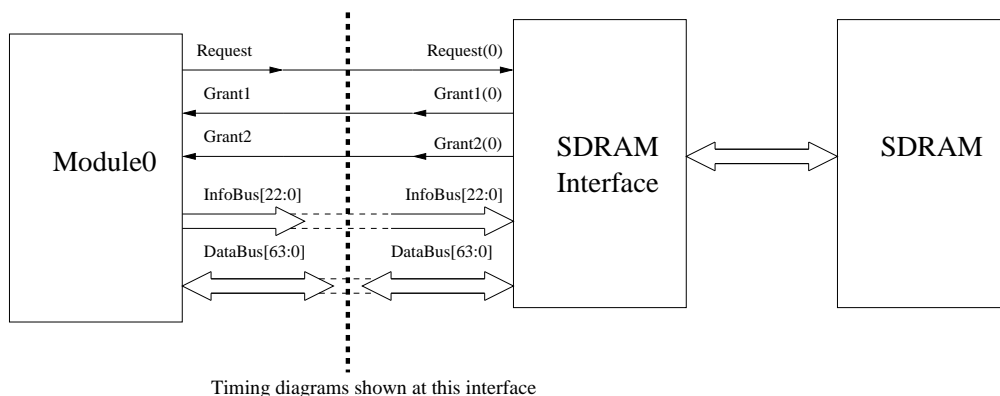


Figure 2: Connecting a module to the SDRAM Controller

Figure 1 shows the module interface for the SDRAM. This interface has two types of signals : one which participate in the arbitration for the SDRAM access and other which actually access the data. The signals Request, Grant1 and Grant2 (as shown in the figure) are used for the arbitration while others are used to access the data. When a module wants to perform a SDRAM operation, it requests the access by asserting the Request signal high. A module can make multiple requests for multiple SDRAM operations by holding the Request signal high for multiple clock cycles. Each request corresponds to one SDRAM operation. After making the request(s), module waits for the Grant1 signal. When the Grant1 signal is asserted high, the module gives all the information regarding the transaction to the SDRAM controller on the InfoBus. This information includes the address, the operation type and the burst length. The SDRAM controller analyzes this information and schedules the actual SDRAM access for this module. At the scheduled time, the SDRAM controller asserts Grant2 high which means that the request (first request in the queue in case of multiple requests) has been granted access to the SDRAM and module can read the data from the DataBus in case of read request or put data on the DataBus in case of a write request. All the requests made by a module are serviced sequentially which means if information regarding request A was sent before the information regarding the request B then A will be serviced before B.

Figure 3 shows the exact protocol and timing for a burst read operation of burst length  $n$  and

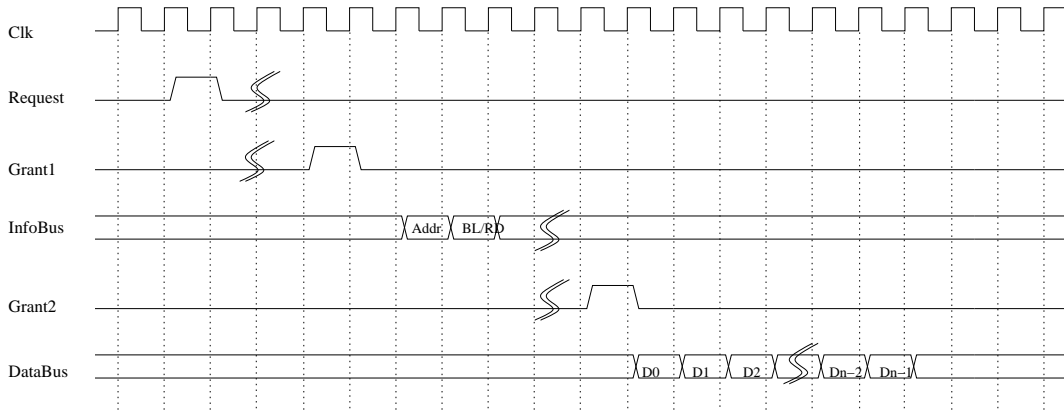


Figure 3: Timing diagram for a burst read operation

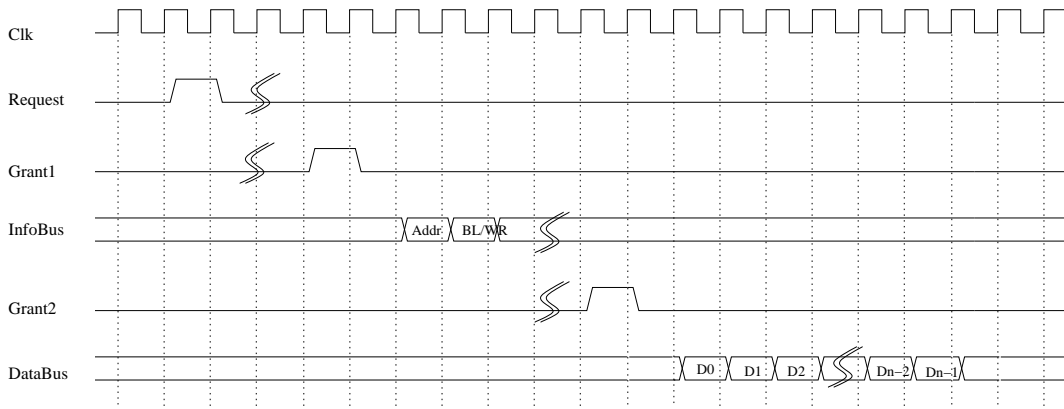


Figure 4: Timing diagram for a burst write operation

Figure 4 shows the timing of a burst write operation of length  $n$ . The discontinuation sign on the signals indicates that there is a variable clock cycles of latency between the Request and Grant1 as well as Grant1 and Grant2.

Figure 2 shows how the module is connected to the SDRAM controller. Request signal of module 'x' is connected to the Request(x) of the SDRAM controller. Similarly Grant1 signal of module is connected to Grant1(x) and Grant2 to Grant2(x) of the SDRAM controller.

Following list details the interface signals and their purpose.

- Request : This signal is used to make a request to the SDRAM controller for SDRAM access. Each request corresponds to one SDRAM transaction. The number of requests made is equal to the number of clock cycles for which the Request is asserted high. Every request gets corresponding grant. *The number of outstanding requests which haven't got the grant can not exceed 3*. Thus, if a module asserts the request signal high for 3 clock cycles then it means it has made 3 requests and if the first request gets a grant then the module can make only one more request.
- Grant1 : This signal is the input to the module from the SDRAM controller. When this signal is asserted high, it means that the module has been given grant to access the InfoBus (described below), on which it can put the information regarding *only one* SDRAM transaction it wants to do. This information includes the SDRAM address, the type of operation (read or write)

and the burst length of this operation. First the 23 bit address should appear on the InfoBus at the time shown in the Figure 4 and Figure 3 then burst length and operation type should appear together on the bus in the next clock cycle. *The number of clock cycles between the Request and Grant1 is variable.*

- InfoBus[22:0] : This is a unidirectional tristate bus which is used to supply information regarding a SDRAM transaction to the SDRAM controller. After Grant1 is given to the module, it should put the 23 bit address on the InfoBus such that InfoBus[22:0] = Address[22:0]. In the next clock cycle it should put the 8 bit burst length and the operation type on the InfoBus such that InfoBus(8) = OperationType and InfoBus[7:0] = BurstLength[7:0]. The convention for the operation type is OperationType = '0' for a *read* operation and '1' for a *write* operation. *Except for these two clock cycles, the module should always drive HiZ on the InfoBus.*
- Grant2 : This signal is a input to the module from the SDRAM controller. For a read operation as shown in Figure 3, when Grant2 signal is asserted high the module should start taking in the data from the next clock cycle. This data corresponds to the read request which was at the head of the queue. Since all the requests made by the module are serviced in the same order in which they appear the module should keep track of the which data corresponds to which request. For a write operation as shown in Figure 4, when Grant2 signal is asserted high, then module should start putting data words on the data bus two clock cycles thereafter.
- DataBus[63:0] : This is a bidirectional tri-state data bus. The module should drive HiZ when it is not used by the module.