

**Arbitration  
and  
Protocol  
subset for the I/O bus**

**Chet Pawlowski  
Low End Systems  
Digital Equipment Corporation**

# 1 pass arbitration

## require

- unrestricted mode (power-up default)
- assign all potential masters to highest priority (1 pass)

## don't allow

- restricted mode (either all nodes use it or none may)
- requirement would change if restricted mode became default

## justification

- each pass is expected to require 200ns
- arbitration priority written to CSR, don't change "on the fly"
- priority also affects arb. settling time
- 2 priorities available, assign high priority to split response if used

# arbitration messages

## require

- power fail

## don't allow

- interrupts by arb. message

## justification

- only power fail is defined
- only asynchronous interrupts could be signalled with this mechanism (no DMA completion) because ordering is lost

# **idle bus arbitration**

**require**

**don't allow**

- **idle bus arbitration**

## **justification**

- **designs must tolerate long latency for high traffic conditions; short latency under light load is not important**
- **coupling of arbitration control with parallel bus is extra complexity**

# **compelled and packet modes**

## **require**

- **compelled mode**
- **packet mode is reserved for future implementation, when more cost effective**

## **don't allow**

## **justification**

- **compelled mode is simplest**
- **gets us 160-200 Mbyte/sec on 64-bit bus**

# **connected and split**

## **require**

- **connected transactions**

**don't allow**

- **split transactions**
- **wait status**

## **justification**

- **arbitration latency is long compared to split hand-shake**
- **single stream bandwidth on I/O bus suffers with split transactions**
- **prefetching allows us to pay the price of latency only once for long data transfer**

# transaction types

## require

- read and write
- read and write partial

## don't allow

- all others

## justification

- I/O bus transactions can and should be simple

# **locked transactions**

**require**

**don't allow**

- any lock transactions
- LK\* bit for creating atomic operations

**justification**

- locks work poorly from remote busses
- protocols exist that don't require locks
- many RISC processors don't provide traditional locks



# **broadcast and broadcast transactions**

**require**

**don't allow**

- **broadcast or broadcast**

**justification**

- **no perceived benefit**

# **intervention and cache coherency**

**require**

**don't allow**

- **cache coherent transactions**
- **intervening status**

**justification**

- **I/O bus is not part of cache coherence domain**

**message passing**

**require**

**don't allow**

- **message passing**

**justification**

- **higher level protocol that is not required**