

Hewlett-Packard Company
 Sunnyvale
 974 East Arques Avenue
 PO Box 3486
 Sunnyvale, California 94086-3486
 408 720 3000



HEWLETT
 PACKARD

7

X3T9.2/92-11

**** DRAFT ****

To: X3T9.2 COMMITTEE (CAM)
From: Chris Runowicz, Hewlett-Packard Company
Subject: DMA data transfer method as outlined in AT Attachment
Date: 12/30/91

BACKGROUND:

In a personal computer, transferring data to and from an AT-type (or IDE) hard disk is commonly via BIOS or software-driver programmed I/O reads or I/O writes, or Programmed I/O. The inclusion of DMA data transfer specifications in the CAM ATA specification document offers an alternative method for data transfer. The primary advantage of DMA over Programmed I/O is increased data transfer performance or throughput. The CAM specifications provide guidelines for both hard disk and system designers for the implementation of DMA-type data transfers in personal computer products.

Drive DMA data transfer cycles will be conducted only after a Read DMA or Write DMA command is successfully received by the drive and the system's DMA channel and controller have been setup. Each complete DMA data transfer cycle involves four phases: 1) the assertion of DMA Request from the drive, 2) the assertion of DMA Acknowledge from the system, 3) the assertion of ~~the~~ I/O Read(s) or I/O Write(s) sent from the host which validates one word of data along the data bus connecting the system and the drive for transfer, and 4) the disertion of DMA Acknowledge from the system.

DMA data transfer cycles have a maximum cycle time, limited by both the system DMA controller and the drive. Assuming the drives are operating at maximum data transfer abilities, the type of system will determine the maximum data transfer rate. Most ISA DMA controllers are limited to 1 microsecond per cycle, or 2Mbyte/s max. This is about equivalent to Programmed I/O maximum transfer rates. However, all EISA DMA (and some ISA) controllers can transfer as fast as 500 nanoseconds per cycle, or 4Mbyte/s max. This is a CAM specification capatible, non-burst mode called B-type. Hence, while ISA implementations offer no significant performance advantages, EISA implementations offer a two-fold increase in maximum data transfer over Programmed-I/O and should considered worthwhile.

PROBLEM:

Upon installing AT-type drives supporting DMA data transfers into various systems, it was found that phase one, or the time it takes for the system to respond to the drive DMA Request with a DMA Acknowledge, is typically one microsecond. This is because there is significant arbitration overhead to ultimately grant the bus to the DMA controller. The time to complete the remaining phases of a DMA data transfer cycle would be in addition to this! Hence, data transfer performance was very poor. Unfortunately the CAM AT Attachment Specification document does not specify this time. Note that both ISA and EISA implementations will suffer from this arbitration overhead, in varying degrees.

Proposed Solution:

The CAM AT Attachment specifies that only word of data will be transferred per DMA data transfer cycle. This is illustrated in Figure 11-3 of the document. With this, system arbitration overhead penalties are disastrous. However with the EISA DMA B-type Demand Mode protocol, more than one word of data can be transferred per DMA data transfer cycle. Hence, the arbitration overhead can be amortized over more data transfers, thereby approaching the 4Mbyte/s limit (see the attached performance comparison chart). It is proposed that CAM specifies this transfer process.

Operation:

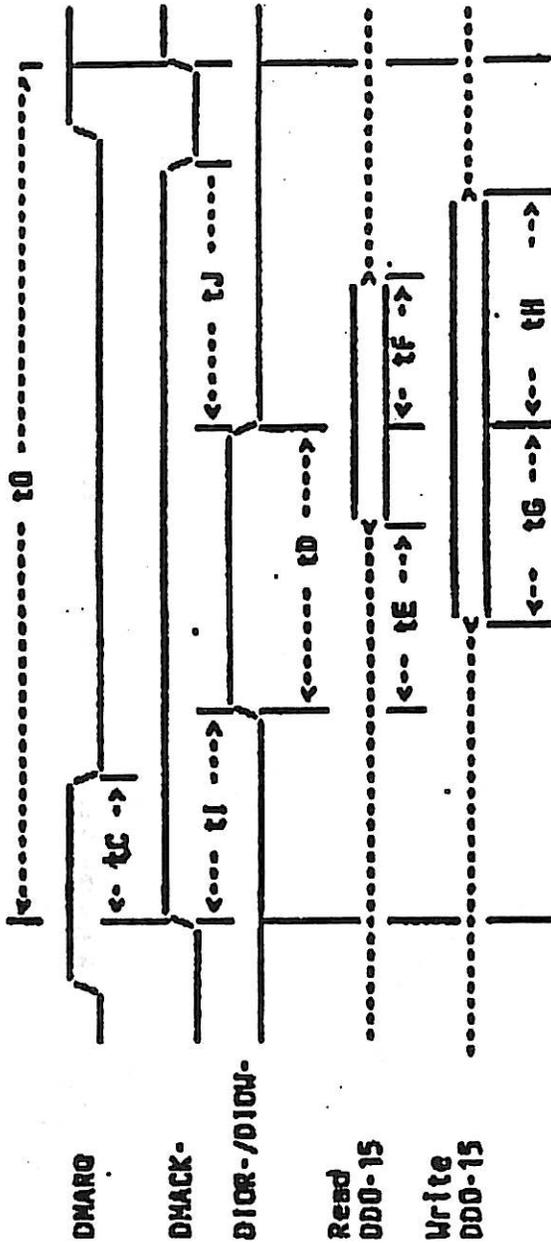
The following details the operation of EISA B-type Demand mode DMA data transfers.

- phase 1: To begin a DMA data transfer cycle, the drive asserts DRQ when data is ready to be transferred. DRQ is held asserted until the entire specified data transfer is completed, or the drive must abort the data transfer flow, for example to refill/flush its data FIFO.
- phase 2: The system asserts ~DACK when the drive's DMA channel has won arbitration. ~DACK is held asserted until the entire specified data transfer is completed, or the system must abort the data transfer flow, for example to refill/flush its data FIFO. If ~DACK is negated to abort the specified data transfer, the drive can either negate or hold DRQ asserted (anticipating the next cycle). Either way, the DMA transfer cycle is completed. It is recommended that the drive not negate DRQ, as this will save time by immediately beginning the next cycle once the system is ready.
- phase 3: While maintaining DRQ and ~DACK asserted, a minimum of one IO Read or IO Write will be asserted. The trailing edge of each IO Read or IO Write shall latch 16-bits of data. Depending on the type of operation, IO Read or IO Write pulses can sequentially come from the system within a minimum of 4 backplanes clock periods (i.e. 8MHz backplane, minimum IOR/W cycle time is 500ns), each transferring 16-bits of data. IO Reads and IO Write pulses will not be mixed together in the same DMA data transfer cycle, and will be determined by type of drive command presently executing, either Read DMA or Write DMA.
- phase 4: The DMA data transfer cycle will end when either the drive or system determines the full completion of the specified data transfer, or if the data transfer must be aborted. To end or abort the data transfer, the drive shall negate DRQ after the leading edge of the assertion of the last valid IO Read or IO Write. To end or abort the data transfer, the system shall negate ~DACK after the trailing edge of the assertion of the last valid IO Read or IO Write. At this point the DMA data transfer cycle is completed.

Specifications:

There are 3 modes of DMA operation presently specified by the CAM AT-Attachment: modes 0, 1 and 2. Mode 0 is compatible with ISA DMA data transfers and should remain unchanged. Modes 1 and 2 are not compatible or functionally appropriate (see performance comparison) with either ISA DMA or EISA B DMA Demand mode. Hence, it is proposed that the operation and specification of mode 1 and mode 2 change to reflect the functionality specified above, and timing specified in the following page.

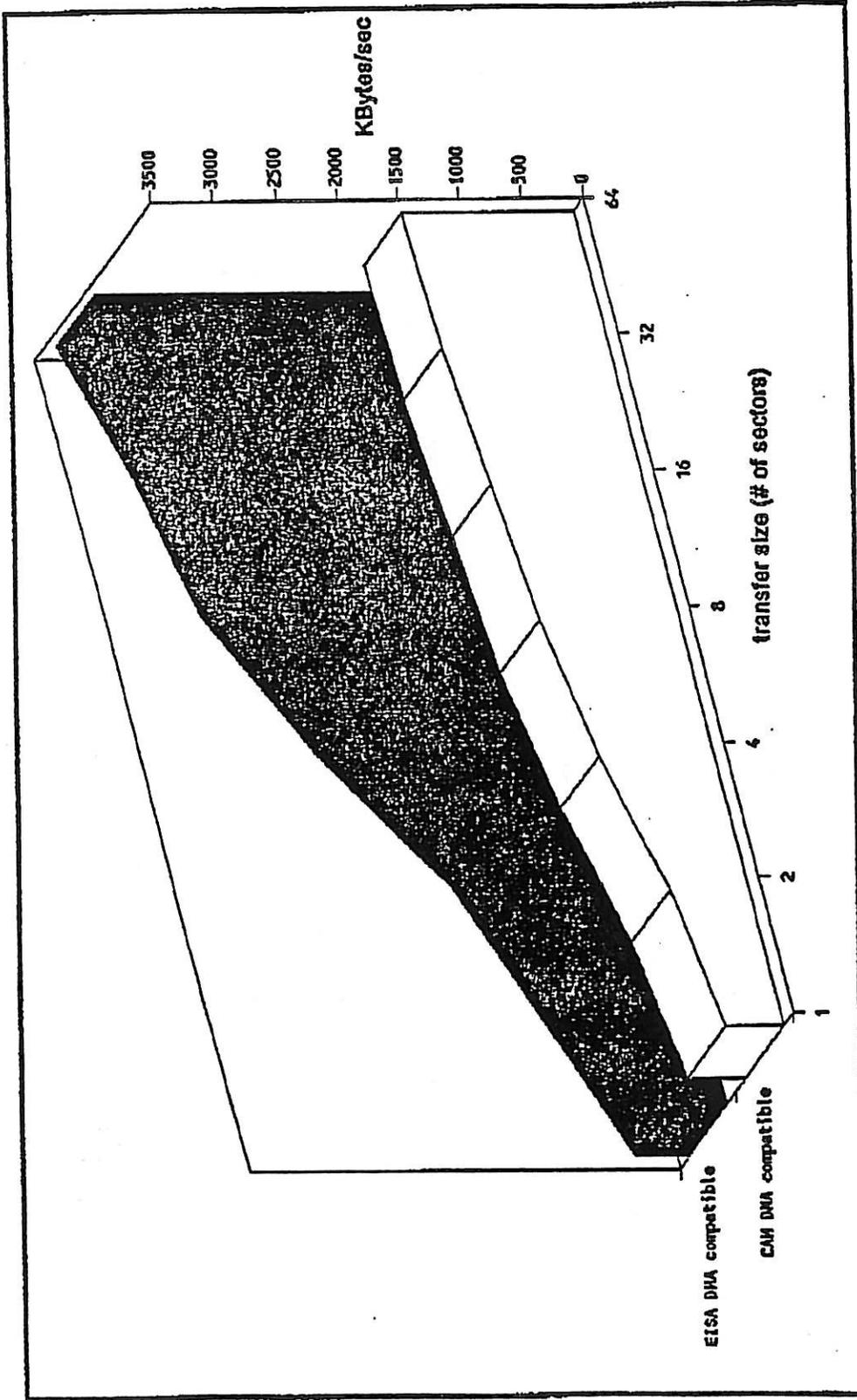
(EXISTING)
 48A DAME TIMING for CAM DMA timing v0.1



| DMA Timing Parameters | Mode 0 | | Mode 1 | | Mode 2 | |
|----------------------------------|--------|-------|--------|------|--------|------|
| | (Min) | (Max) | nsec | nsec | nsec | nsec |
| t0 Cycle Time | 960 | 480 | 480 | 240 | 240 | 80 |
| tC DMACK to DMREQ Delay | 200 | 480 | 100 | 240 | 80 | 120 |
| tD DIOR-/DIOH- 16-bit Data Setup | 480 | 250 | 240 | 150 | 120 | 50 |
| tE DIOR- Data Hold | 250 | 5 | 150 | 5 | 50 | 5 |
| tF DIOR- Data Setup | 5 | 250 | 5 | 100 | 5 | 35 |
| tG DIOH- Data Setup | 250 | 50 | 100 | 30 | 35 | 20 |
| tH DIOH- Data Hold | 50 | 0 | 30 | 0 | 20 | 0 |
| tI DMACK to DIOR-/DIOH- Setup | 0 | 0 | 0 | 0 | 0 | 0 |
| tJ DIOR-/DIOH- to DMACK Hold | 0 | 0 | 0 | 0 | 0 | 0 |

Keep
 Change to mode 1 ERA B
 Change to mode 2 ERA B

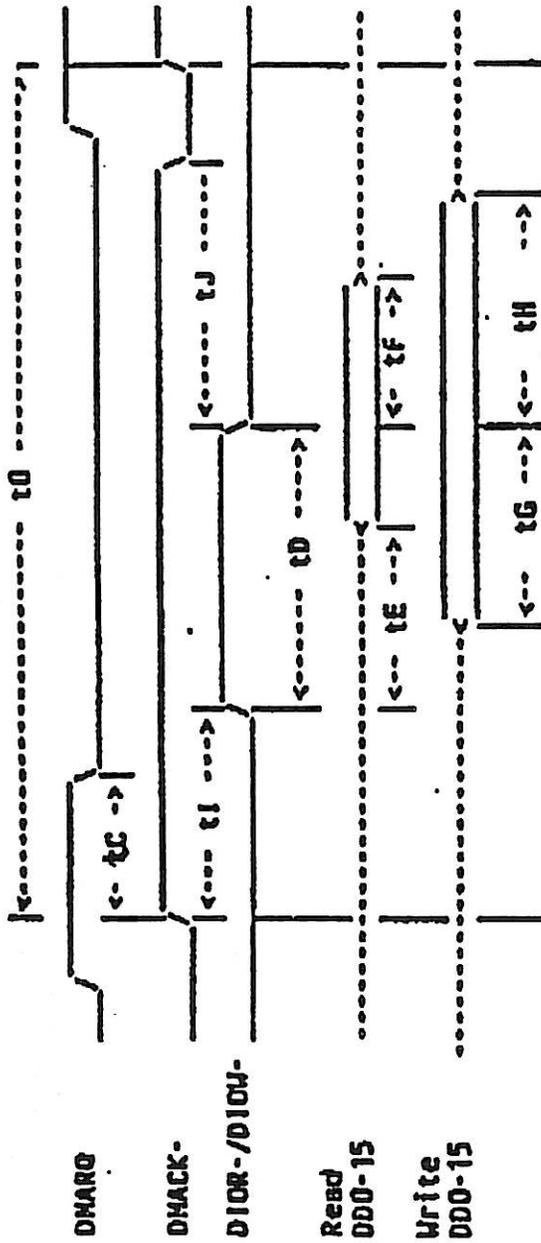
"CORETEST" TRANSFER RATES when installed in an EISA system



CAM DMA compatible
 EISA DMA compatible

Chris Rutowicz, HENLETT-PACKARD COMPANY, 12/30/91, GEN_DHA1-XLC

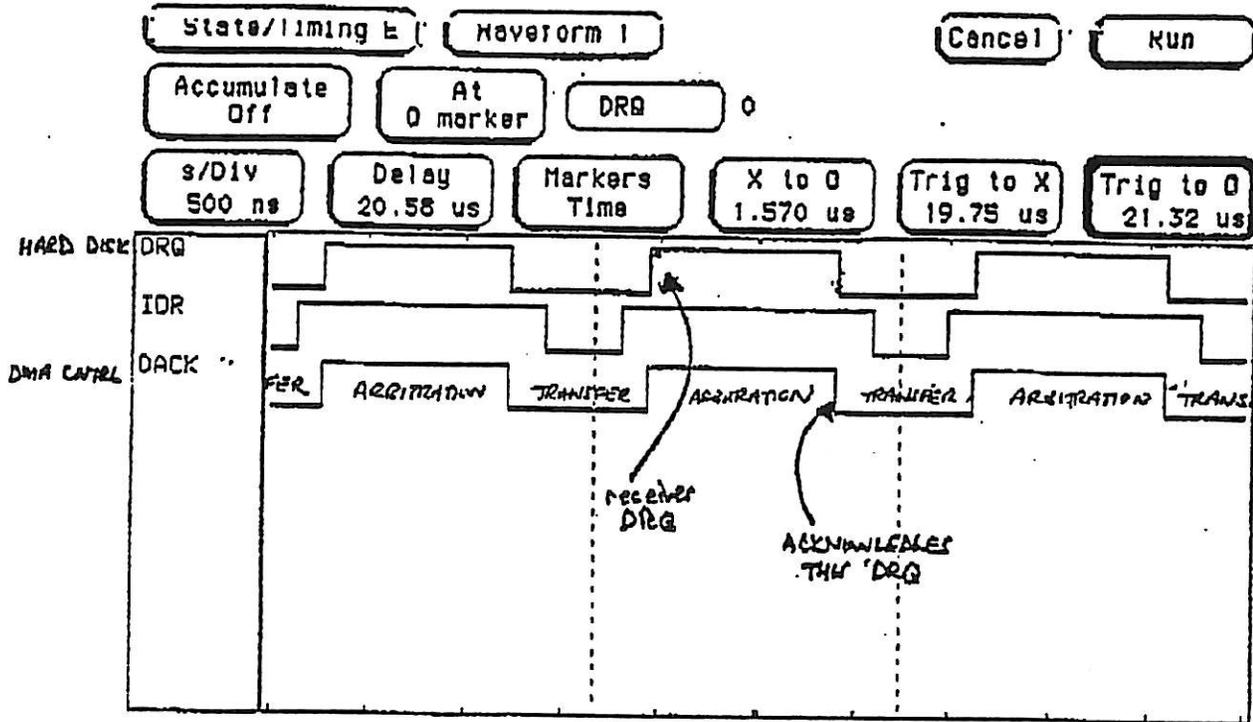
(EXISTING)
 #SADNE TIMING for CAN DMA timing v1.1



| DMA Timing Parameters | Mode 0 | | Mode 1 | | Mode 2 | |
|----------------------------|--------|-------|--------|------|--------|------|
| | (Min) | (Max) | nsec | nsec | nsec | nsec |
| Cycle Time | 960 | 480 | 480 | 240 | 240 | 80 |
| DMACK to DMREQ Delay | 200 | 480 | 100 | 240 | 100 | 120 |
| DIOR-/DIOU- 16-bit | 250 | 250 | 150 | 150 | 50 | 50 |
| DIOR- Data Setup | 5 | 5 | 5 | 5 | 5 | 5 |
| DIOU- Data Setup | 250 | 250 | 100 | 100 | 35 | 35 |
| DIOU- Data Hold | 50 | 50 | 30 | 30 | 20 | 20 |
| DMACK to DIOR-/DIOU- Setup | 0 | 0 | 0 | 0 | 0 | 0 |
| DIOR-/DIOU- to DMACK Hold | 0 | 0 | 0 | 0 | 0 | 0 |

Keep
 Change to mode 1 EISA B
 Change to mode 2 EISA B

EXPERIMENTAL DATA OF CAM DMA:



EXPERIMENTAL DATA OF EISA B-DEMAND DMA

