WHAT DOES THE CUSTOMER WANT ???

PERFORMANCE ???
SSA ?
ATA/IDE ?
SBP ?
PROTOCOL ???
FCP ?
COSTS???
SSA-SP ?

COMPATIBILITY ???
P1394 ?
P1285 ?
SERIAL ???
FC-PH ?

PARALLEL ???
SCSI 2? 3?
IPI-2 ? 3?
GPP ?
DISK LEVEL INTERFACES?
HISTORY LESSON No 100001

IN THE BEGINNING

- ONE CONTROLLER THAT HANDLED MANY DISK’S OR SPINDLES PER SYSTEM
- COSTS OF ECC CIRCUITS, BUFFERS, ENDEC’S, WERE AMORTIZED OVER MANY SPINDLES.
- ARRAYS OF DISKS WERE COMMON (FARMS)
- DISKS HAD VERY LITTLE MICROCODE
DISK LEVEL INTERFACES?

HISTORY LESSON No 100001-2

WHEN WE INHERITED INTELLIGENCE

- Systems became smaller and only needed one or two disks
- Integration techniques became very cost effective
- If only one drive ..., put everything in it... takes cost from system
- Performance was questionable ..., but over-ridden by cost advantages
- The need for better communications between system and peripheral brought us intelligent interfaces
- Disks have large microcode programs...
Development costs are high
APPLICATION DEVELOPMENT
PERIPHERAL DEVICE INTERFACE

DISK LEVEL INTERFACES?

HISTORY LESSON No 100001-3

NOW WHERE ARE WE

- LARGE CAPACITY SPINDLES.... MANY GBYTES UNDER ONE ARM...
  MANY SPINDLES TO OBTAIN AVAILABILITY, THROUGH-PUT

- AVAILABILITY OF DATA MORE CRITICAL THAN EVER

- COST IS STILL CRITICAL BUT PERFORMANCE IS REQUIRED

- ACCESSIBILITY TO DATA FOR INCREASED SYSTEM PERFORMANCE

- FAST - WIDE - SINGLE - OPTICAL - SYSTEM INTERFACES
  TO ACCESS PERIPHERALS

- MORE FUNCTIONS.. MORE MICROCODE.. MORE MORE

MORE MORE

RKR 01/93
APPLICATION DEVELOPMENT
PERIPHERAL DEVICE INTERFACE

ANOTHER INTERFACE !!!

DESIGN GOALS

■ LOW LEVEL ACCESS TO THE MEDIA... IF REQUIRED
■ VARIABLE DATA TRANSFER RATES FOR PERFORMANCE VS COST TRADE-OFFS
■ ALLOW VENDORS TO TAKE ADVANTAGE OF VOLUME PRODUCTION
■ PROVIDE EASY IMPLEMENTATION OF ARRAYS OF DISKS
■ PROVIDE SIMPLIFIED COMMAND STRUCTURES AND FLOW
■ PROVIDE FOR MAXIMUM THROUGH-PUT
■ PROVIDE FOR ERROR FREE TRANSFERS
## Architectural Model

<table>
<thead>
<tr>
<th>System Attachment Level (SCSI, ATA, IPI3, VME, etc., etc.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peripheral Controller</td>
</tr>
<tr>
<td>Basic Device Level Attachment</td>
</tr>
<tr>
<td>Peripheral Device</td>
</tr>
<tr>
<td>Attribute</td>
</tr>
<tr>
<td>------------------------------------------</td>
</tr>
<tr>
<td>System Interface Controls</td>
</tr>
<tr>
<td>Cache Memories (If Required)</td>
</tr>
<tr>
<td>Controls Multiple Devices (No Limits)</td>
</tr>
<tr>
<td>Buffer Areas</td>
</tr>
<tr>
<td>System Specific Knowledge</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
ADVANTAGES OF INTERCONNECTION SCHEME

- PROVIDES LOW LEVEL ACCESS TO MEDIA (if required)
- SIMPLIFIES PERIPHERAL DEVICE Firmware
- LESS COSTLY PERIPHERAL DEVICES
- EASE OF IMPLEMENTATION FOR ARRAYS OF DISKS (RAID's ??)
- PERIPHERAL GUIDED VS SYSTEM AFTER THOUGHT
- PERFORMANCE ORIENTED WITH LESS $ THAN SCSI (PER SYSTEM)
MAXTOR CORPORATION
APPLICATION DEVELOPMENT
PERIPHERAL DEVICE INTERFACE

INFORMATION MOVEMENT PROTOCOL

PERIPHERAL CONTROLLER

4 COMMAND WORDS TRANSMITTED

4 COMMAND WORDS RCV'D & DECODED

4 STATUS WORDS RCV'D

4 STATUS WORDS TRANSMITTED

WAIT FOR INTERRUPT (DO SOMETHING ELSE)

PERFORM SPECIFIED COMMAND

TASK COMPLETED TRANSMIT 4 STATUS WORDS
APPLICATION DEVELOPMENT
PERIPHERAL DEVICE INTERFACE

WHERE DOES IT FIT?
IN THE HIERARCHY OF THE SUBSYSTEM

HOST SYSTEM  SCSI
            ATA etc
PERIPHERAL CONTROLLER  BASIC DEVICE INTF
PERIPHERAL DEVICE

TYPICAL PERIPHERAL SUB-SYSTEM
DATA PATH LAYOUT

- CTL SYNC CKTS
- BUFFER
- SYNC CKTS
- ENDEC
- ECC/EDC
- DATA 2,4,8, or 16 BITS
- PLO CLKS
- WRITE AMPS
- READ AMPS
- HEADS

PERIPHERAL CONTROLLER INTERFACE MEDIA DEVICE