

**Aurora Professional Desktop Computer**

**Main Board Design Specification**

**Revision 0.02**

**3/10/1992**

**Mark Nicol**

**ZDS Desktop Hardware Design**

**CONFIDENTIAL, FOR ZDS INTERNAL USE ONLY**

*THIS DOCUMENT CONTAINS PROPRIETARY AND CONFIDENTIAL INFORMATION, INTENDED ONLY FOR LIMITED DISTRIBUTION WITHIN ZENITH DATA SYSTEMS AND GROUPE BULL.*



# TABLE OF CONTENTS

Project Definition .....	1
Design Philosophy .....	2
Design Constraints.....	2
Options Considered.....	2
Design Information .....	4
Functional Description .....	4
CPU interface.....	4
Memory subsystem .....	6
Local RAM.....	6
Memory Controller .....	6
Parity support .....	6
I/O decode & XBus.....	6
FLASH ROM .....	7
Video .....	7
Audio .....	7
EtherNet.....	7
IDE interface.....	7
Floppy control .....	7
SCSI interface .....	7
Serial & Parallel I/O.....	7
Keyboard & Mouse ports .....	8



## 1. PROJECT DEFINITION

The Aurora project is a high performance desktop computer, designed to be fully configured for all normal user requirements. The machine employs a highly modular construction, which allows it to be easily upgraded with a more powerful CPU, a larger disk drive, or even a larger chassis for more expansion. The CPU will reside on a plug-in module, providing an easy upgrade path from the Intel 80386-SX/20MHz CPU to the anticipated P24/66MHz. The off-board expansion will be provided by a high performance EISA bus backplane with either 2 or 4 available slots. The main features of the system electronics are as follows:

- ? Modular CPU architecture allowing 80386SX @20MHz to P24@66MHz with/without cache
- ? Modular backplane allows either 2 or 4 EISA bus slots for expansion boards
- ? Memory capacity of 4M (minimum) to 128M maximum with 8 banks of 1, 4, or 16M SIMMs
- ? Field upgradeable BIOS in 128K FLASH ROM
- ? Extended VGA video with 1024 x 768 resolution @ 16 colors
- ? EtherNet LAN adapter
- ? SCSI bus interface
- ? IDE drive interface
- ? Floppy drive controller supporting 4M media
- ? Serial I/O port
- ? Parallel I/O port
- ? PS/2 compatible Mouse port
- ? PS/2 compatible Keyboard connector

## 2. DESIGN PHILOSOPHY

The design of the Aurora main board is intended to be highly compatible with past designs done at Zenith Data Systems. As such, it draws upon some of the same support circuitry that was used in past designs. This recycling of established sub-circuits should reduce the risk of the overall design to the sections that are totally new such as Ethernet, SCSI and video.

## 3. DESIGN CONSTRAINTS

The main design constraint of the Aurora main board is the relatively small amount of PCB real estate and cabinet volume available, compared to the large amount of functionality desired. This requires that the highest degree of integration be employed in the chipsets selected.

The EISA based Aurora and the ISA based Orion, both must function with the same CPU modules. This means that the CPU to memory controller interface must be similar enough that the CPU modules will not suffer performance degradation or cost penalty when plugged into either system.

## 4. OPTIONS CONSIDERED

As mentioned above, the requirement to use the same CPU modules as the Orion demands that the memory controller interface be compatible. The most obvious choice is for the two systems to use the same controller. This option, however, does not allow the Orion to realize any significant cost differential compared to the Aurora. Since the only advantage of the ISA based Orion to the EISA based Aurora would be cost, the chipset selection of the Orion must be slanted toward a very cost effective solution. Several chipsets allow the Orion to highly integrate all ISA bus and memory control functions, and at the same time function with either the 386DX or the 486SX/DX. The ability to strap the chipsets for 386 or 486 mode allows the CPU modules to interface quite simply to the memory controller. Since these chipsets are highly integrated and there are numerous vendors competing for the design, the ISA solution can offer a significant cost savings.

The same options aren't available in an EISA solution. There are currently only two EISA bus interface chipsets available, the Intel 82350DT, and the OPTI. The OPTI chipset is new and has not been used in a production consumer machine, or even demonstrated in a working prototype. The additional risk of using the OPTI chipset is too great, since the Aurora is a very schedule sensitive project. This narrows the bus interface selection down to the Intel 82350DT chipset.

The Intel 82350 EISA interface chipset is well understood, although it has numerous errata, and we have used it successfully in the past in both the Piranha and the Viper machines. The 82350DT is the next generation of this chipset, and provides an additional method with which it interfaces to the host memory controller. This new asynchronous mode allows it to be used with another new member of the 82350 family, the 82359 memory controller. This narrows the selection of the Aurora memory controller down to either the Viper ASIC or the Intel 82359.

The Intel 82359 provides an asynchronous interface between the CPU and the memory system. It works with two (2) 82353 Advanced Data Path chips to provide a 128 bit wide data path to local memory. It is highly programmable and appears to comply with all the requirements of the MRD. It interfaces to both the 386 and the 486, but requires a *programmable state tracker* to correctly control the memory system. This PST must be located on the CPU module, since it is peculiar to each type and speed of CPU. The inclusion of the PST on the module, however, would greatly impact the cost of the CPU module for the Orion system. In addition, the 82359/82353 combination is itself quite expensive at this time, and appears to pay a performance penalty at certain CPU speeds, due to its slow response time on certain critical signal paths.

The 644-115 Viper memory controller has already been successfully synchronously interfaced to the 82358 in the Viper design, but with the design criteria of having to support 20MHz CPU modules, the asynchronous mode of the 82358DT might be necessary. This should still be a low risk design. Since the Viper chip interfaces to a 486 CPU only, and the system is to be optimized for the 486 family of CPUs, the CPU modules would all have to present a 486-like appearance to the main board. This would require the 386 CPU modules to carry some extra logic such as cycle translation, parity generation, and 1x clock generation which would offset some of the inherent cost savings of the Orion chipsets. This small amount of logic is relatively inexpensive compared to the multiple PLD PST alternative of the 82359 and presents less of a design challenge.

The most logical solution for the Aurora memory controller seems to be the 644-115 Viper ASIC. This chip is somewhat expensive today, but if it were to be turned by the ASIC group, as probably would be required due to some existing bugs, it could be moved from BiCMOS to advanced CMOS technology. This revision 2 chip would be designed as a drop-in replacement for the existing chip, and could be less than half of the current price. This solution has additional advantages in that we control the performance of our own design, can design today with a proven chip to lower our risk, and can cost reduce it as much as possible. If these future cost reductions are realized, and the added manufacturing and inventory savings of a single main board design are considered, it may be economically possible to phase out the ISA board entirely in the near future.

Another option under consideration is to place all the bus specific logic on the already modular backplane assembly. This would allow the ISA and the EISA boards to share a common main board and the plug-in backplane would determine the bus structure. This has many cost saving advantages for the entire cycle of design through manufacturing, however it doesn't allow the ISA board to realize enough of a cost differential to the consumer against the EISA board. This solution also does not appear to fit within the physical constraints of the 2 high enclosure, since the backplane would need to be extended the full length of the machine in order to place and route all the bus specific circuitry.

## 5. DESIGN INFORMATION

The Aurora main board will be designed with no user configurable switches or jumpers which would require that the cabinet be removed. All configurable settings will be programmable through the firmware. In addition, all firmware changes and updates can be performed without the removal of any hardware, through the use of FLASH ROM for BIOS storage. This allows firmware to be changed by simply booting a floppy disk and downloading the new firmware.

The PCB will employ a multilayer construction allowing power and ground planes and multiple signal routing layers. A minimum of 50% of the discrete passive components will be placed on the bottom side of the board, freeing up prime real estate on the top of the board.

## 6. FUNCTIONAL DESCRIPTION

### 6.1. CPU interface

The interface to the CPU module is through a 200 pin KEL type connector. This connector is a high density, low profile design which allows the CPU module to reside slightly above the plane of the main board. The pinout of the connector is ZDS defined and detailed in diagram **x.x**. All CPU modules will present a 486-like interface to the main board and any individual *personality changes* which need to be done to the CPU will be performed on the module. The individual CPU modules may also provide a *native* mode interface which may be used by the Orion ISA main board for performance reasons. There will be an 8 bit ID port between the CPU module and the main board which will provide information between the two systems. The bits are all driven by the CPU module except bit 7 which is driven by the main board. They are defined as follows:

7	6	5	4	3	2	1	0	Definition
							0	386 CPU
							1	486 CPU
						0		SX variant
						1		DX variant
					0			PWA not used
					1			PWA used
			0					32 bit CPU architecture
			1					64 bit CPU architecture
	0	0	0					66.66 MHz speed
	0	0	1					50.00 MHz speed
	0	1	0					40.00 MHz speed
	0	1	1					33.33 MHz speed
	1	0	0					25.00 MHz speed
	1	0	1					20.00 MHz speed
	1	1	0					16.00 MHz speed
	1	1	1					other speed
0								ISA system board
1								EISA system board

Diagram x.x goes here.

## 6.2. Memory subsystem

### 6.2.1. Memory Controller

The memory controller for the Aurora is the 644-115 Viper controller. This is a ZDS designed LSI device (ASIC), usable in a variety of 486 based designs. It interfaces to the ISA and EISA buses directly, and supports both burst mode accesses from the 486 CPU and the EISA bus. It is also designed to interface directly to the Viper PWA devices, but does not require them to be present in all designs. Additional features of the chip include minimum wait-state penalties, 32, 64, and 128 bit wide data paths which are configured via internal registers, slush ROM support, and EMS support. Some limitations of the controller that would need to be corrected in order to meet all MRD requirements are:

- Dynamic switching of data path widths determined by odd/even SIMM installation.
- DOS 5.0 memory mapping flexibility.
- 4 Megabit DRAM support to allow 128 megabyte local memory using 16 megabyte SIMMs.

### 6.2.2. Local RAM

The main memory of the Aurora will consist of 8 banks of SIMMs (Single Inline Memory Modules). Each module will include one parity bit per byte, and the capacity can be either 1 megabyte (256K x 36 bits), 4 megabytes (1meg x 36 bits), or 16 megabytes (4meg x 36 bits) of FAST PAGE mode 80nS DRAM. The memory controller will access the SIMMs using either a 64 bit or a 32 bit data path, depending on how many SIMMs are installed. The machine will come standard with 4 megabytes of memory installed, and will support up to 128 megabytes after the 644-115 ASIC has been modified.

### 6.2.3. Parity support

The 486 CPU will generate and check parity for all host initiated cycles. This requires that all non-486 based CPU modules must emulate this function on the module. All read accesses of local memory from the ISA/EISA bus will be parity checked in the data path buffers to the bus. Similarly, all bus write accesses of local memory will cause a parity bit to be generated and stored into RAM.

## 6.3. I/O decode & XBus

The I/O decode functions will be handled by a single high integration EPLD device. This will provide a single source for all I/O decodes and also allow the Xbus buffers to be controlled from one device. Additional advantage of a single device are cost savings, board real estate savings, and power savings. The EPLD can be easily converted to a masked part, when the code is firm and volume is high, to further reduce cost.

The Xbus will consist of 20 bits of XA address which is used for address decoding, and 16 bits of XD data. The XD data bus will in addition be split into two separate data busses, one the full 16 bits wide, and another only 8 bits wide. This is needed due to all the peripheral devices present on the main board, which present a significant load to any single bus.

#### **6.4. FLASH ROM**

The system firmware for both the system BIOS, video BIOS, and SCP code will be stored in a 128K x 8 FLASH ROM. This will allow firmware updates to be easily done without opening the machine and removing hardware. This also allows the firmware to be installed as the last operation at the end of the production line, giving the firmware group some additional time for development. The FLASH ROM will contain a non-flashable boot code section which will be used to store the most basic operations needed to load the remainder of the boot code from a floppy drive or I/O port.

#### **6.5. Video**

The video section consists of a standard, but high performance, VGA controller which resides on the Xbus. The VGA will support extended modes such as 1024 x 768 with 16 colors, and 72Hz refresh. Standard video memory will be 512Kbytes, but sockets will be provided for a full 1 megabyte. A bi-directional pass-through feature connector will be installed on the main board to allow connection of an optional *Windows Accelerator Module* (WAM), to be installed in the first EISA bus slot in the backplane. This is essentially a 2/3 length TIGA video board.

#### **6.6. Audio**

The audio support is yet to be determined.

#### **6.7. EtherNet**

The EtherNet LAN support is yet to be determined.

#### **6.8. IDE interface**

The IDE interface will consist of a 40 pin header for 2 internal IDE drives. These drives may be DMA devices allowing high speed operation.

#### **6.9. Floppy control**

The floppy disk controller will be an Intel 82077AA compatible device which incorporates all of the logic required for PC-AT compatible floppy disk and tape drive control including the data separator, drive buffers, and a 16 byte FIFO to avoid bus latency problems.

#### **6.10. SCSI interface**

The SCSI bus interface is yet to be determined, but will include both an internal and an external connector.

#### **6.11. Serial & Parallel I/O**

The I/O ports are contained in a single 16c45x compatible device which contains all the logic needed to directly interface to the 9 pin serial and 25 pin parallel connectors, except for the RS-232C serial drivers and receivers. The I/O decoding and COM1/COM2 mapping are done in the I/O decode EPLD.

## 6.12. Keyboard & Mouse ports

The keyboard and mouse functions are both provided by the combination of MKI (Mouse Keyboard Interface) and 8052 SCP (System Control Processor). The MKI is a ZDS designed ASIC which provides an interface to an 8051/8052 based SCP. The SCP was originally an 8041 based microprocessor, and to remain IBM compatible the SCP must retain this 8041 interface. The 8052 microprocessor has many advantages including faster execution speed, more efficient instruction set, and more specific device variants. The MKI also provides some other features such as GATE A20 and CPU RESET. The use of the standard ZDS desktop MKI/SCP combination allows us to re-use the already developed SCP code with little or no changes. One new concept which will be implemented in this design, is the slushing of SCP code as opposed to having the code embedded in the SCP's ROM. This is not only necessary due to patent restrictions, but also offers all of the advantages of the FLASH ROM previously discussed. The keyboard and mouse ports are both IBM PS/2 compatible.