APPENDIX D

Nx586™ Processor Databook

PRELIMINARY
December 6, 1994

NexGen, Incorporated
1623 Buckeye Drive
Milpitas, CA 95035

Order # NxDOC-DB001-03-W

Contents

Figures

Preface

This databook covers the Nx586™ processor (called *the processor).* The databook is written for system designers considering the use of these devices in their designs. We assume an experienced audience, familiar not only with system design conventions but also with the x86 architecture. The *Glossary* at the end of the book defines NexGen's terminology, and the *Index* gives quick access to the subject matter.

NexGen's Applications Engineering Department welcomes your questions and will be glad to provide assistance. In particular, they can recommend system parts that have been tested and proven to work with NexGen™ products.

Notation

The following notation and conventions are used in this book:

Devices and Bus Names

- *Processor or CPU*—The Nx586 processor described in this book.
- NxVL™ Systems Logic—The NxVL system controller described in the *NxVL System Controller Databook.*
- NxPCI™ Systems Logic—The NxPCI system controller described in the *NxPCI System Controller Databook.*
- NxMC™ Memory Logic—The NxMC memory controller described in the *NxMC Memory Controller Databook.*
- NexBus^5 System Bus—The NexGen system bus, including its multiplexed address/status and data bus (NxAD<63:0>) and related control signals.
- NexBus Processor Bus—The Nx586 processor bus, including its multiplexed address/status and data bus (AD<63:0>) and related control signals.

Signals *and* Timing Diagrams

- Active-Low Signals—Signal names that are followed by an asterisk, such as ALE*, indicate active-low signals. They are said to be "asserted" or "active" in their low-voltage state and "negated" or "inactive" in their high-voltage state.

- Active-High Signals—Signal names, such as GALE, that indicate active-high signals. They are said to be "asserted" or "active" in their high-voltage state and "negated" or "inactive" in their low-voltage state.
- Bus Signals—In signal names, the notation <*n:m*> represents bits *n* through *m* of a bus.
- Reserved Bits and Signals—Signals or bus bits marked "reserved" must be driven inactive or left unconnected, as indicated in the signal descriptions. These bits and signals are reserved by NexGen for future implementations. When software reads registers with reserved bits, the reserved bits must be masked. When software writes such registers, it must first read the register and change only the non-reserved bits before writing back to the register.
- Source—In timing diagrams, the left-hand column indicates the "Source" of each signal. This is the chip or logic that outputs the signal. When signals are driven by multiple sources, all sources are shown, in the order in which they drive the signal. In some cases, signals take on different names as outputs are logically ORed in group-signal logic.
- *Tri-state*®—In timing diagrams, signal ranges that are high impedance are shown as a straight horizontal line half-way between the high and low level.
- *Invalid and Don't Care*—In timing diagrams, signal ranges that are invalid or don't care are filled with a screen pattern.

Data

- *Quantities*—A word is two bytes (16 bits), a dword or doubleword is four bytes (32 bits), and a qword or quadword is eight bytes (64 bits).
- Addressing—Memory is addressed as a series of bytes on eight-byte (64-bit) boundaries, in which each byte can be separately enabled.
- *Abbreviations*—The following notation is used for bits and bytes:

Bits	b	as in "64b/qword"
Bytes	B	as in "32B/block"
kilo	k	as in "4kB/page"
Mega	M	as in "1Mb/sec"
Giga	G	as in "4GB of memory space"

- *Little* Endian Convention—The byte with the address xx...xx00 is in the least-significant byte position (little end). In byte diagrams, bit positions are numbered from right to left: the little end is on the right and the big end is on the left. Data structure diagrams in memory show small addresses at the bottom and high addresses at the top. When data items are "aligned," bit notation on a 64-bit data bus maps directly to bit notation in 64-bit-wide memory. Because byte addresses increase from right to left, strings appear in reverse order when illustrated according to the little-endian convention.
- *Bit Ranges*—In a range of bits, the highest and lowest bit numbers are separated by a colon, as in <63:0>.
- *Bit Values*—Bits can either be *set* to 1 or *cleared* to 0.

- *Hexadecimal and Binary Numbers*—Unless the context makes interpretation clear, hexadecimal numbers are followed by an *h*, binary numbers are followed by a *b*, and decimal numbers are followed by a *d*.

Related Publications

The following books treat various aspects of computer architecture, hardware design, and programming that may be useful for your understanding of NexGen products:

NexGen Products

- *NxVL System Controller Databook*, NexGen, Milpitas, CA, Tel: (408) 435-0202.
- *NxPCI System Controller Databook*, NexGen, Milpitas, CA, Tel: (408) 435-0202.
- *NxMC Memory Controller Databook*, NexGen, Milpitas, CA, Tel: (408) 435-0202.

Bus Standards

- VESA VL-Bus Version 2.0, Video Electronics Standards Association, San Jose CA 1993.
- PCI Local Bus Specification Revision 2.0, Peripheral Component Interconnect Special Interest Group, Hillsboro, Oregon, 1993.

x86 Architecture

- John Crawford and Patrick Gelsinger, *Programming the 80386*, Sybex, San Francisco, 1987.
- Rakesh Agarwal, *80x86 Architecture & Programming*, Volumes I and II, Prentice-Hall, Englewood Cliffs, NJ, 1991.

General References

- John L. Hennessy and David A. Patterson, *Computer Architecture*, Morgan Kaufmann Publishers, San Mateo, CA, 1990.