THE NS16000 MICROPROCESSOR FAMILY
SYSTEM OVERVIEW

REVISION A
NOVEMBER 15th, 1979

COMPANY CONFIDENTIAL
DO NOT REPRODUCE

NATIONAL SEMICONDUCTOR (ISRAEL) LTD

TABLE OF CONTENTS

TABLE OF FIGURES

Page

APPLICABLE DOCUMENTS

- * NS16000 Timing Specifications.
- * NS16032 MPU Architecture Specifications.
- * NS16016/16008 MPU Architecture Specifications.
- * NS16032 MPU Hardware Target Specifications.
- * NS16016/16008 MPUs Hardware Target Specifications.
- * NS16081 Floating Point Processor Target Specifications.
- * NS16082 Memory Management Unit Target Specifications.
- * NS16201 Clock Generator Target Specifications.
- * NS16202 Interrupt Controller Unit Target Specifications.
- * NS16203 DMA Controller Target Specifications.

CHAPTER 1

INTRODUCTION

The NS16000 microprocessor family features advanced microprocessor architecture. The family supports a wide spectrum of system configurations extending from a minimum part-count low-cost system to a powerful 16M byte system. The architecture provides complete upward compatibility from one family member to another.The architecture is expandable and in the future with the help of advancing technology it will be possible to create new CPUs that are downward compatible with the present ones.

The NS16000 microprocessor family is supported by a set of peripherals and slave processors which provide sophisticated interrupt and memory management facilities as well as high speed floating point operations.

CHAPTER 2

FAMILY MEMBERS

2.1 NS16008, NS16016, NS16032 MICROPROCESSOR CHIPS

Initially three microprocessor chips are introduced : the 16032 .- a high-end MPU packaged in a 48-pin DIP, and two medium range MPU's in a 40 pin DIP: THE 16008 and the 16016, using 8 bit and 16 bit Data Bus respectively. MPU options are summarized in Table 1.

MPU	16008	16016	16032
Addressability	64Kbyte	64Kbyte*	16Mbyte
Data Bus Width	8	16	16
Basic Data Types	8/16	8/16	8/16/32
8080 Emulation	Yes	Yes	No
Virtual Storage	No	No	Yes
Slave Processor-Compatibility	No	Yes	Yes
Dedicated Reg.'s	8x16	8x16	2x16, 6x24
G.P. Registers	8x16	8x16	8x32
Pins	40	40	48
H.L.L. Inst.	Yes	Yes	Yes

*16Mbyte with 16082.

Table 1. NS16000 MPU Options

2.2 SLAVE PROCESSORS

The NS16000 architecture is designed to allow future expansion of the instruction set.The new instructions can be executed by special slave-processors which behave like an extension to the MPU. The design of the MPU is such that if no slave-processor exists, a software trap is generated. This way software routines can substitute for non existing slave processors.Some of the more popular slaves may be incorporated into a future high density MPU without requiring any changes in the software.

2.3 NS16081 FLOATING POINT PROCESSOR

The 16081 Floating Point Processor is a member of the 16000 slave processor family. It provides high speed floating point computation for both single and double precision operands. The 16081 is designed to operate either as a slave processor if a 16032 or 16016 MPU is used,or as a peripheral when a 16008 or another non 16000 family MPU is used.When it is used as a slave,its instructions are an extension to the MPU instruction set,using the same symmetrical addressing modes.The 16081 floating point algorithms comply with the proposed IEEE standard.

2.4 NS16082 MEMORY MANAGEMENT UNIT

The 16032 architecture makes advanced memory management schemes feasible. The 16082 Memory Management Unit provides:

* Dynamic Address Translation using Memory Page Tables
* On Chip Cache for the most recently used Page Table Entries.
* Virtual Storage Management
* Memory Protection
* Program Flow Tracing
* Program Breakpointing
* In System Emulation Support

MMU based systems require only one additional clock cycle per memory reference if the Page Table Entry is resident in the On Chip Cache.The cache has a hit ratio of above 95%. If the PTE is not in the cache the MMU automatically accesses the memory tables and inserts the missing information in the cache.Constraints on memory/peripheral timing are the same as in non MMU based systems.All these features are attained while using a standard 48 pin package.

2.5 INTERRUPTS AND THE NS16202 INTERRUPT CONTROLLER UNIT

All MPU chips have the following program-interrupt inputs (in decreasing order of priority):

* Reset	(NRST)
* Abort	(NABRT)
* Non Maskable Interrupt	(NNMI)
* Maskable Interrupt	(NINT)

Reset is used to initialize the microprocessor.

Abort is used to interrupt memory cycles in case of protection violation,bus error etc. The 16032 MPU provides the facility of reexecuting aborted instructions;this is intended for Virtual Memory support.

The Non-Maskable Interrupt (NNMI) should be used for top priority interrupts such as power failure, In System Emulation, etc.

The Maskable Interrupt (NINT) input is software programmed to be either Vectored or non-Vectored. The architecture supports up to 256 Interrupt sources. The 16202 Interrupt Controller Unit is a 40 pin chip designed to handle 16 Interrupt sources(8 of them are software interrupts).The chip is cascadable to allow 256 interrupt sources using one master and 16 slave 16202s. The 16202 provides:

* Programmable priority structure
* Masking
* Automatic-end-of-Interrupt when the Return from Interrupt (RETI) instuction is executed
* Software interrupts
* On-chip real time clock

When the 16202 is used with the 16008, 8 of the 16202's pins serve as a general purpose I/O port.

2.6 NS16203 DMA CONTROLLER

The MPUs are designed to support DMA transfers. The 16203 is an advanced 48 pin DMA controller with the following features:

* Local and Remote Configurations
* Two or four device support(depending on the configuration)
* 8 or 16 bit device compatibility
* Two transfer modes:Direct and Indirect
* Command chaining
* Programmable masked search on the transferred data
* 5Mbyte per second maximum transfer rate

- * Programmable priority
- * Multiple level priority attainable by cascading several 16203s together
- * Autorotating priority capability

2.7 NS16201 CLOCK GENERATOR

The 16201 is a bipolar clock genarator chip for the 16000 family.It requires either a crystal or an external clock and generates the two non overlapping full VCC swing clocks that are required by the 16000 MPUs.In addition it has a Reset input and generates a synchronized Reset-out pulse for the MPU and peripherals. It also syncronizes READY from various sources and generates READY-OUT for the MPU. The 16201 has 5 inputs that are used to enable maximum flexibility in the generation of Wait States when slow memory/peripherals are accessed. These inputs enable insertion of 0 to 15 Wait States or lengthening of the cycle according to the duration of the Continuous Wait State Input.

2.8 SUMMARY OF NS16000 FAMILY MEMBERS

PART	FUNCTION	DATA	ADDR	MPU	PINS
16008	MICROPROCESSOR	8	16	---	40
16016	MICROPROCESSOR	16	16	---	40
16032	MICROPROCESSOR	16	24	---	48
16081	FLOATING POINT	16	--	16032/16016*	28
16082	MEMORY MANAGEMENT	16	24/25	16032/16016	48
16201	CLOCK GENERATOR	--	--	ALL	20
16202	INTERRUPT CONT.	8/16	--	ALL	40
16203	DMA	16	24	16032/16016	48

* Other MPUs including the 16008 can use the 16081 as a peripheral(with either 8 or 16 bit data bus).

Table 2. NS16000 Parts Summary
****

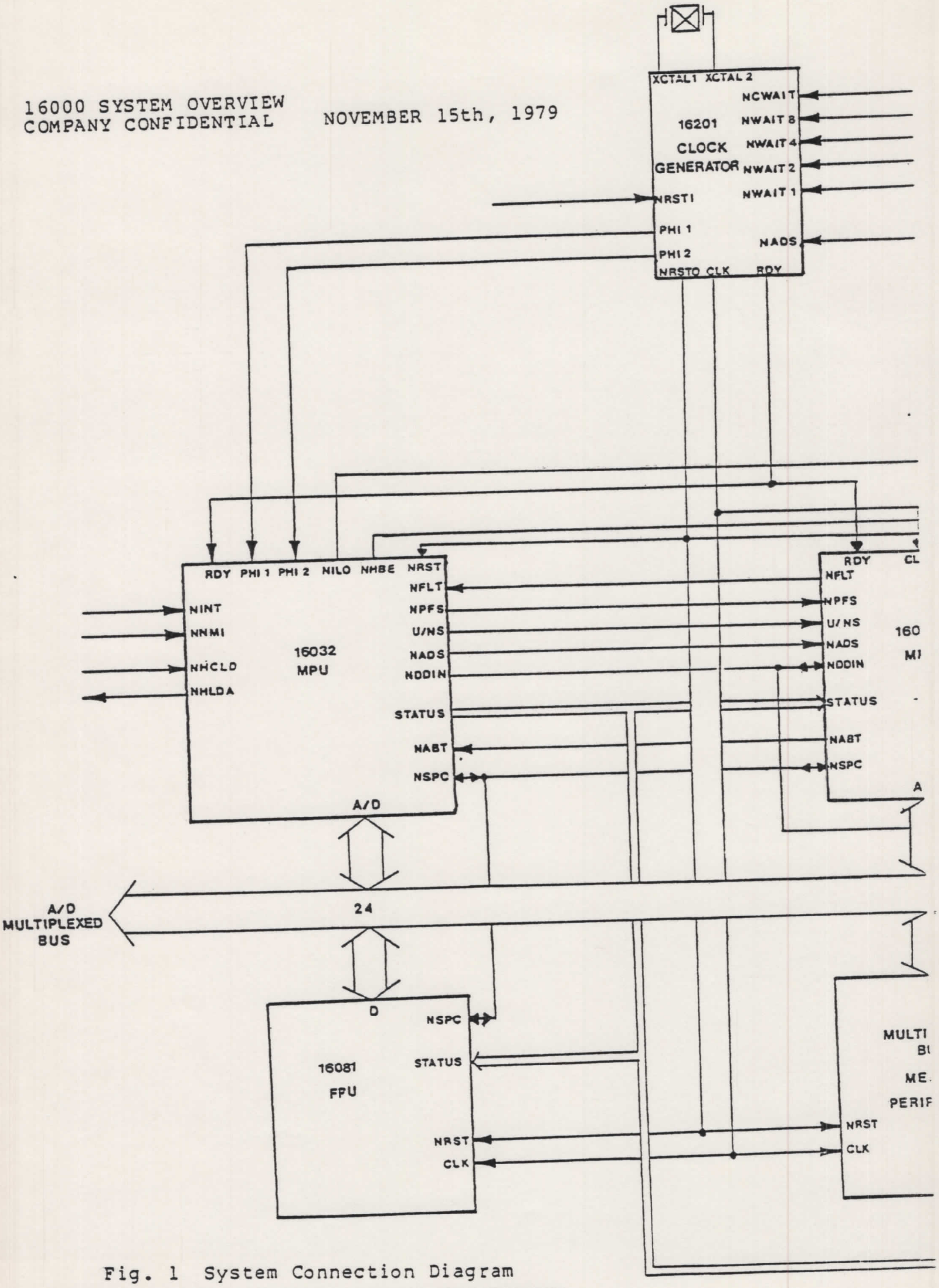
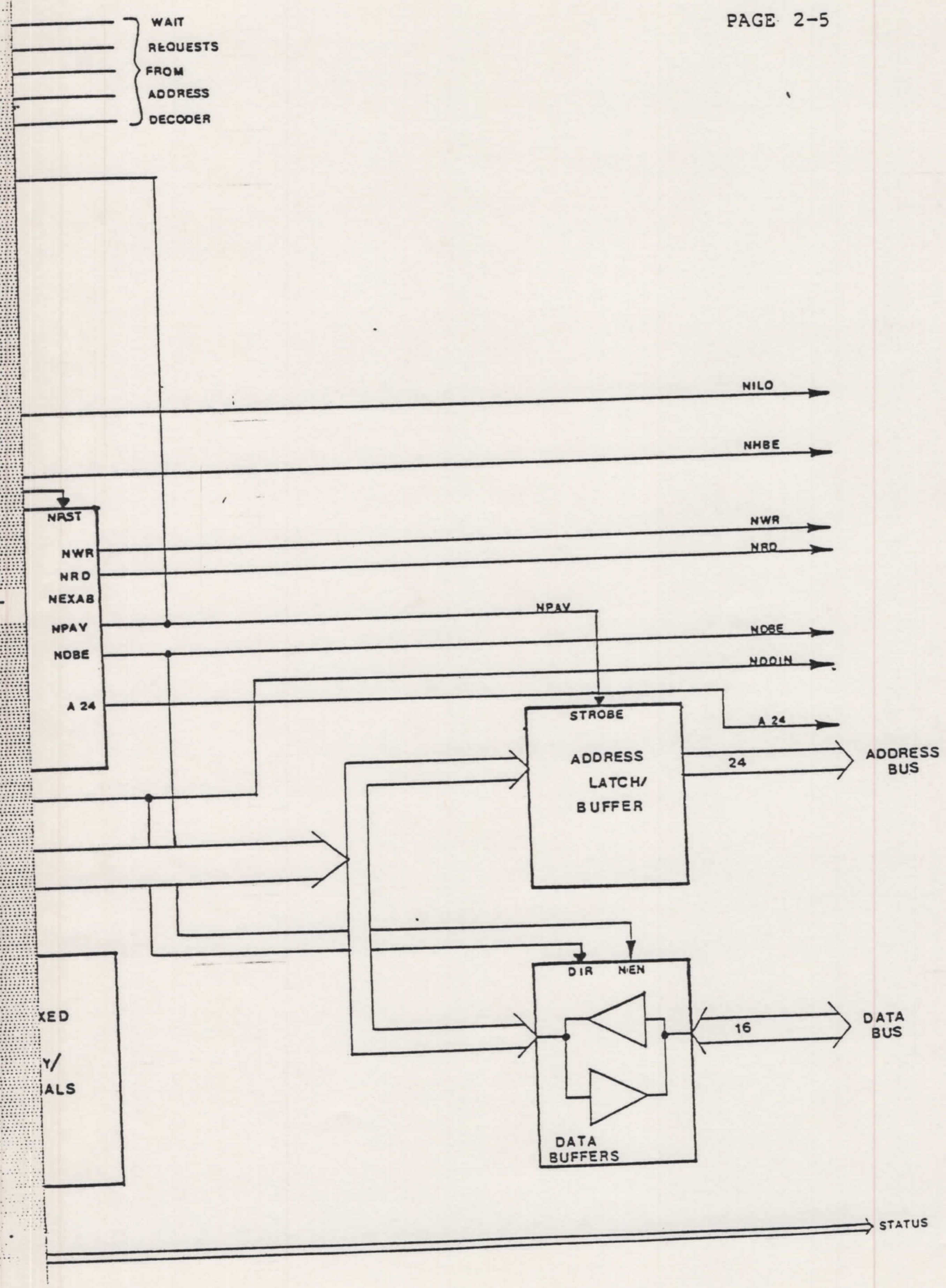


Fig. 1 System Connection Diagram


WAIT
REQUESTS
FROM
ADDRESS
DECODER
NILO
NHBE
NRST
NWR
NRD
NEXAB
NPAV
NDBE
A 24
NWR
NRD
NPAV
NDBE
NDDIN
A 24
STROBE
ADDRESS
LATCH/
BUFFER
24
ADDRESS
BUS
DIR
N/EN
16
DATA
BUS
DATA
BUFFERS
XED
Y/
ALS
STATUS

CHAPTER 3

BUS TRANSFER PROTOCOLS

3.1 GENERAL

There are three types of bus transfers on the 16000 bus:

1. MPU - Memory Transfer
2. MPU - Slave Processor Transfer
3. DMA

3.2 MPU - Memory Transfer

MPU - Memory Transfer is used to transfer data between the MPU and memory or peripherals. The MPU is the initiator and the controller of the transfer. MPU - Memory transfers take four clock cycles (five if address translation is used - see NS16032 Target Specification for details), but the addressed memory or peripheral can extend it by pulling RDY low. The related waveforms are shown in Figures 2,3,4.

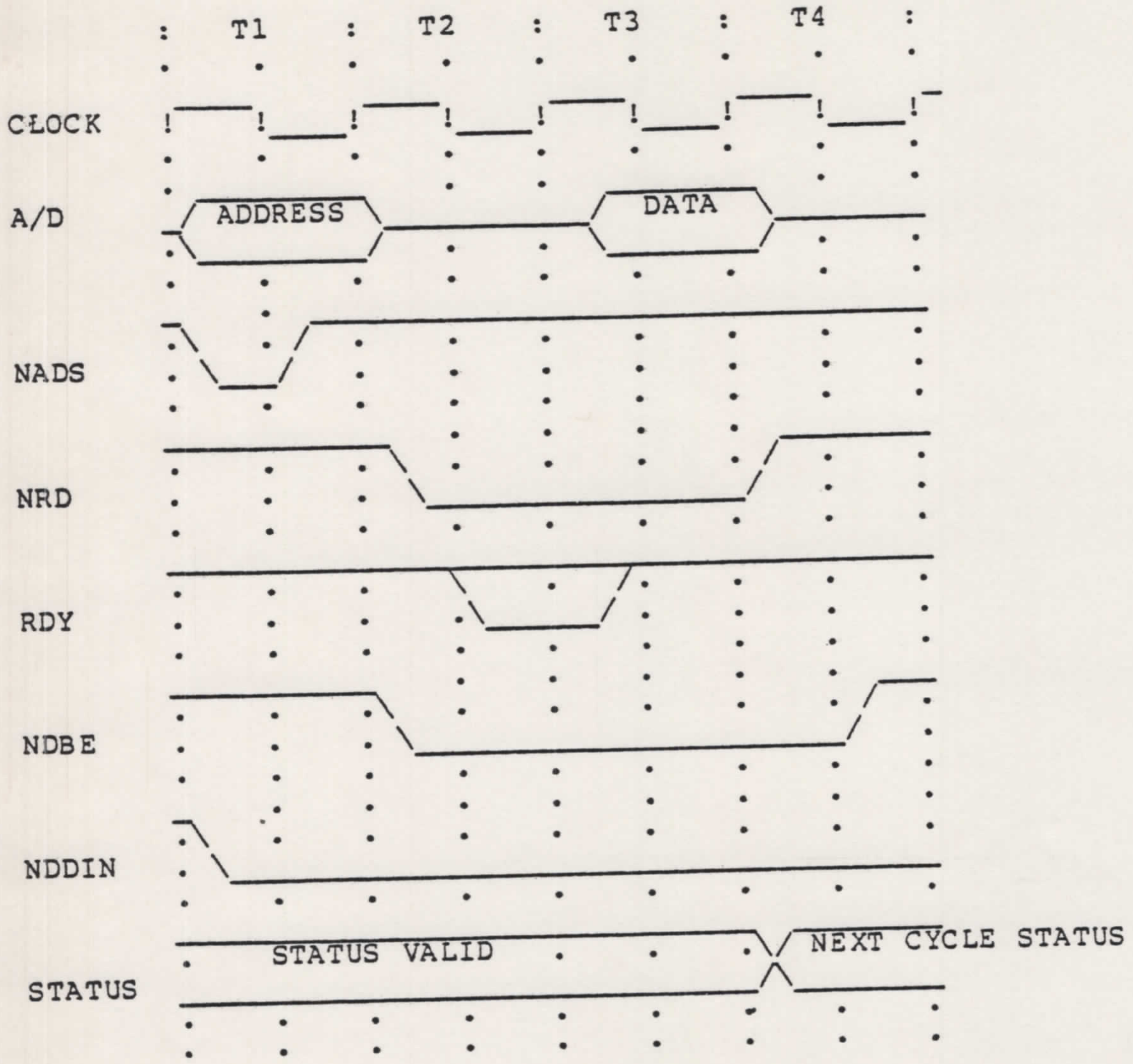


Fig. 2 Read Cycle, No Address Translation

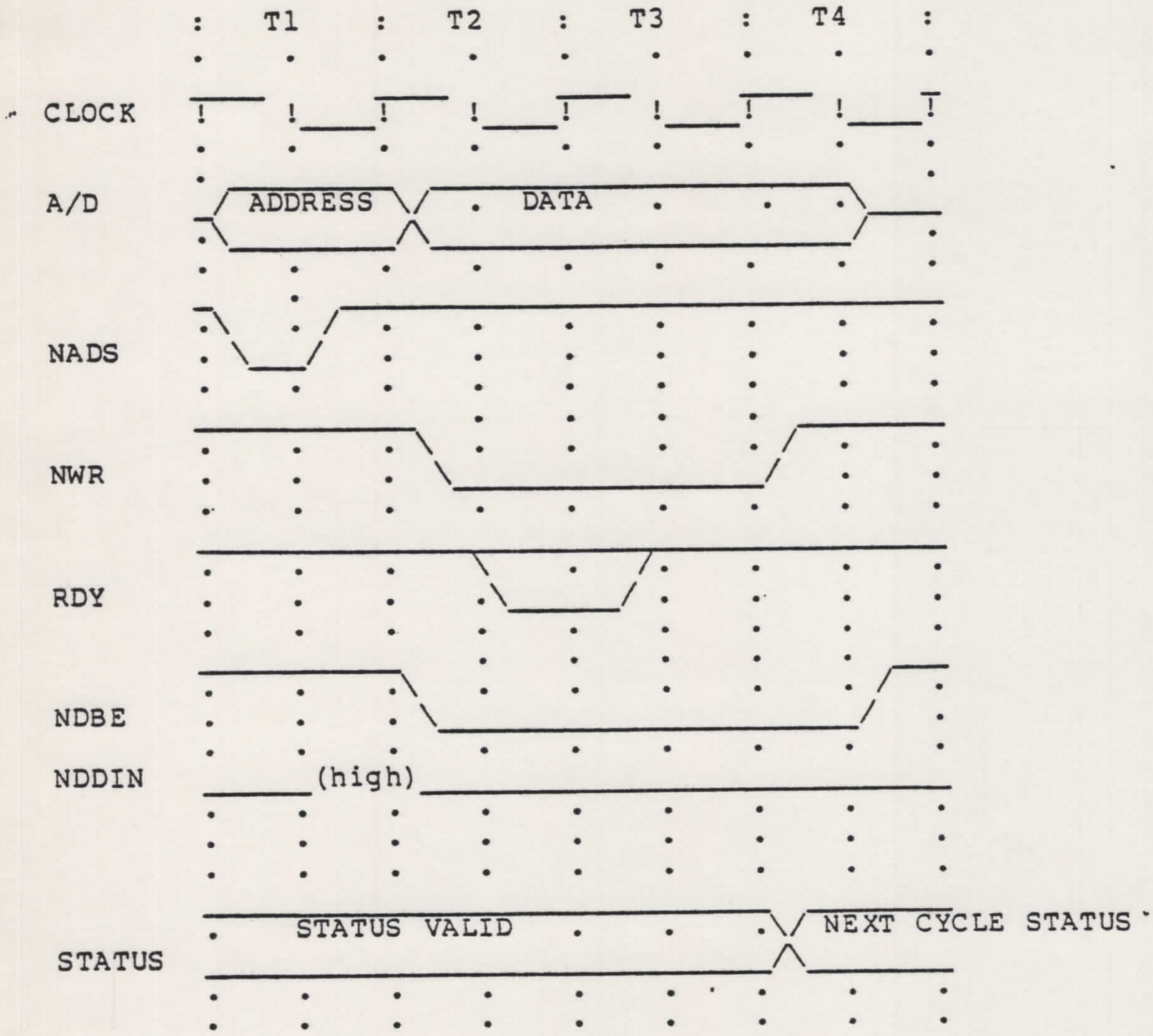


Fig. 3   Write Cycle, No Address Translation

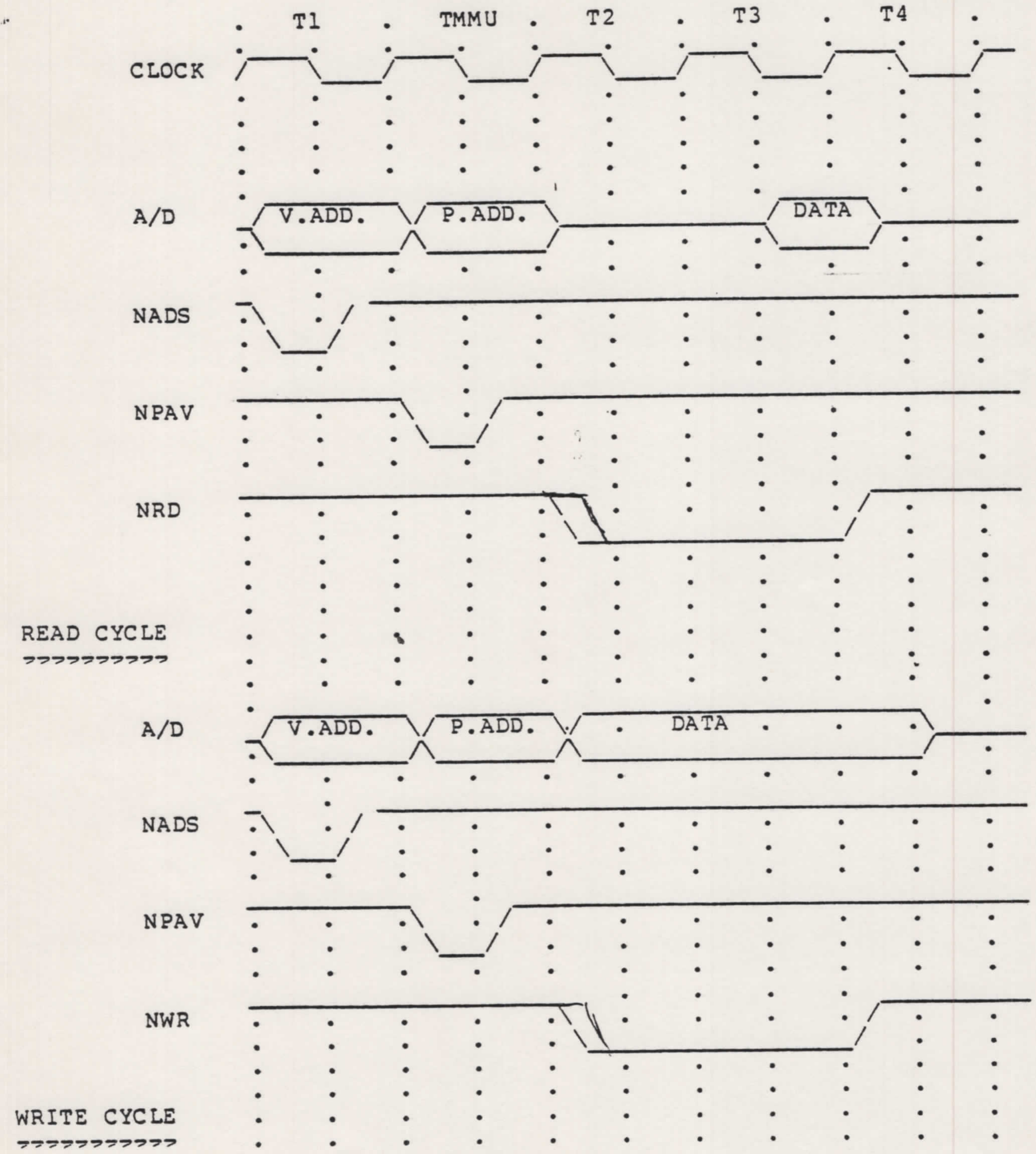


Fig. 4   Read/Write Cycles, Address Translation

3.3 MPU-SLAVE PROCESSOR COMMUNICATION PROTOCOL

When a Slave Processor instruction is executed, the following steps are done:

1. The MPU sends the ID code followed by the OPCODE and all the operands to the SP. If necessary, this sequence is interrupted by memory cycles. The status output is 'Send ID' in the first transfer and 'Send OPC/Operand' in the others. See Fig.5 for the waveforms when data is sent from the MPU to the SP.

2. Other transfers (SP or Memory) may take place depending on the specific SP instruction.

3. The MPU waits for the SP to finish. At the same time it can do memory cycles or grant the bus.

4. When the SP has finished, it signals the MPU by pulling the NSPC line low for a specified time interval.

5. The MPU then reads a status word from the Slave Processor.The LSB of that word is the trap control - if it is set the MPU will do a TRAP.The status output is 'Read SP'. See Fig.6 for the waveforms when data is read from the SP to the MPU.

6. If results are to be sent to the MPU or memory, the MPU will do additional READ cycles from the FPU, with status = 'Read SP Operand'.

If,during the process of reading operands from memory for further transfer into the SP or writing SP results into memory, the MPU gets an ABORT from the MMU, the communication protocol will be interrupted. The SP will wait for the protocol to be resumed, but, instead, it will get the ID again when the instruction is re-executed. This is why the Slave Processors are specified to restart the communication protocol each time a 'Write ID' status with the slave processor's ID is detected.

Since the NSPC is a bidirectional control line, it needs a special I/O buffer in both the MPU and the Slave Processors(see 16032 Hardware Target Specifications for the implementation).

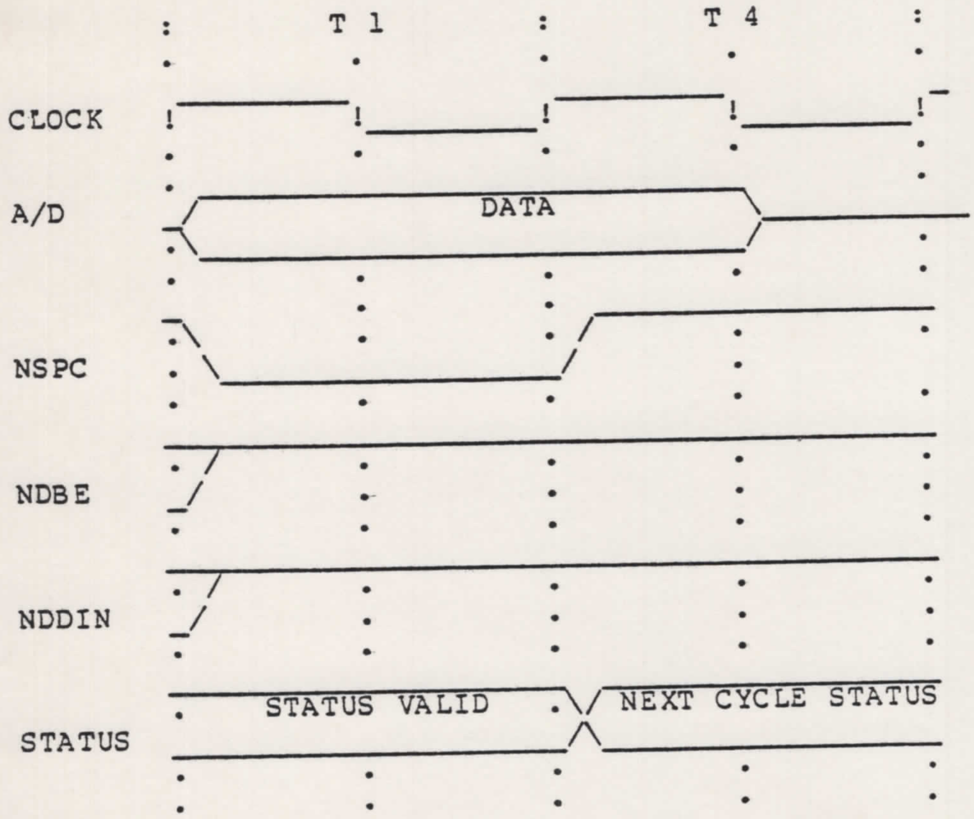


Fig. 5 Write data to Slave Processor

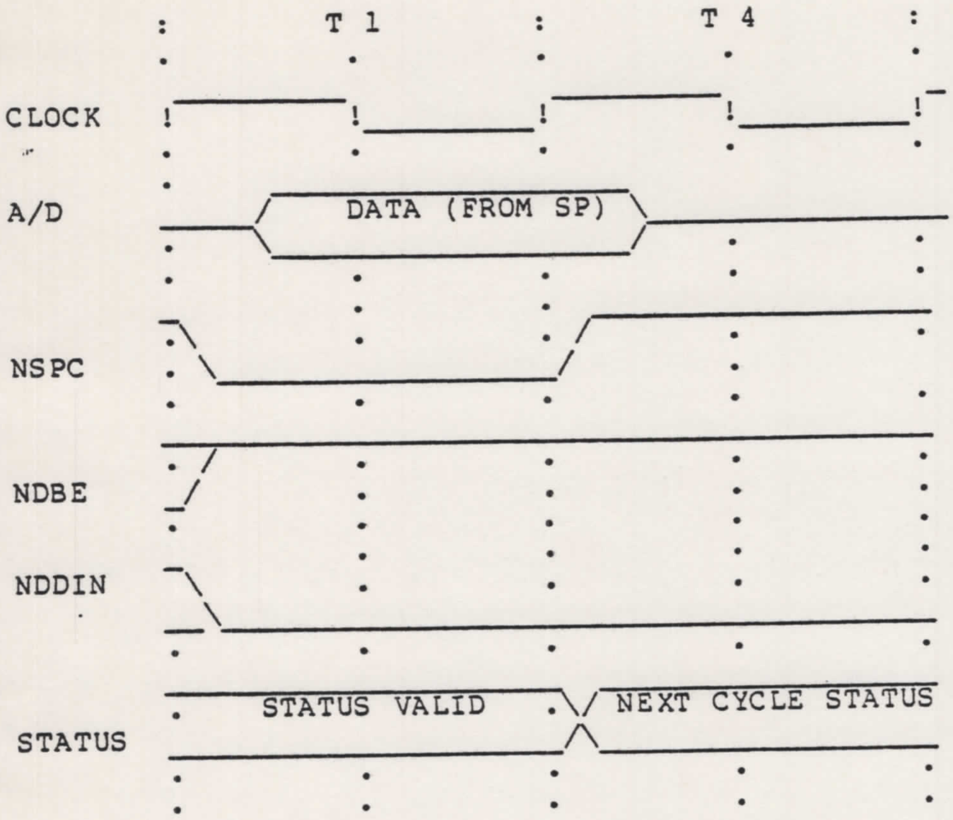


Fig. 6 Read data from Slave Processor

3.4 BUS REQUESTS AND LOCAL BUS ARBITRATION

Local bus arbitration is done using the NHOLD and NHLDA pins of the MPU. The MPU will grant the bus to the requestor after completing its current bus transfer cycle.

Before granting the bus the MPU floats the A/D lines and the NRD,NWR,NDDIN,NDBE and NADS control lines. Thus in a minimum part count system the control bus can be shared, and no external TTL gates are required.

The requestor pulls the NHOLD line low and waits. When the MPU decides to grant the bus, it pulls NHLDA low. When the requestor completes the transfer, it floats off the bus and releases NHOLD. See fig. 7 for the waveforms.

The 16203 DMA Controller is designed to utilize the NHOLD and NHOLDA pins to achieve high speed DMA transfers.

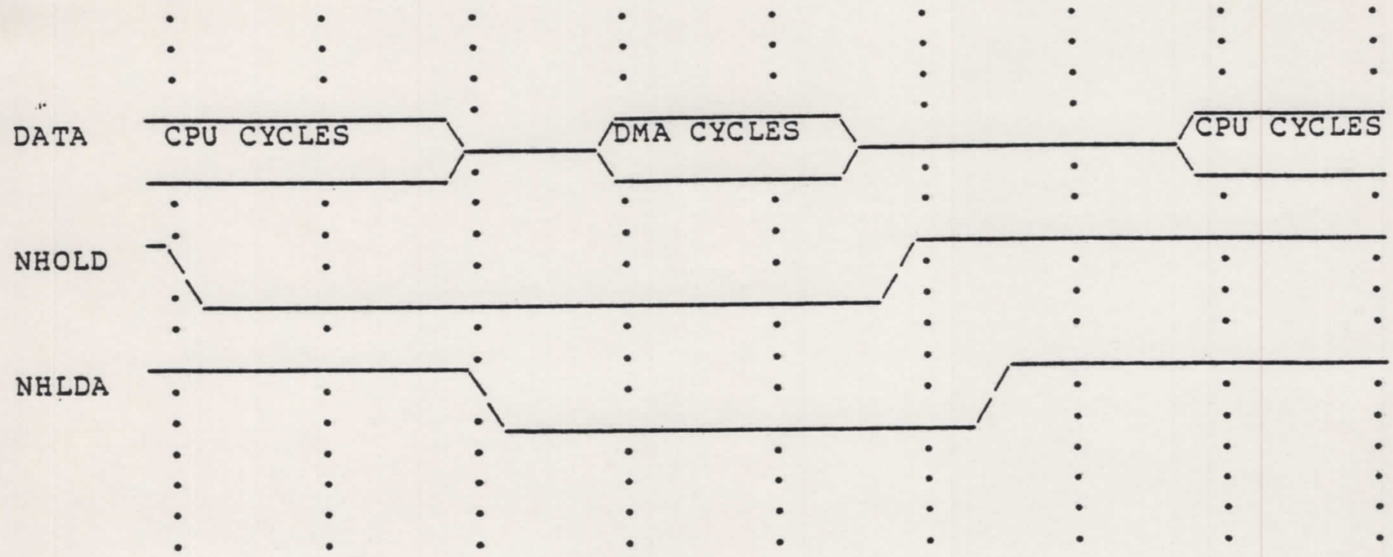


Fig. 7   Hold/Hold Acknowledge Timing

CHAPTER 4

NS16000 BASIC TIMING SPECIFICATION

4.1 Introduction

This chapter contains the basic timing requirements from memories , peripherals and Slave Processors.The requirements from memories and peripherals refer to two basic configurations:Nonbuffered and Buffered (see Figures 8,9).All the timing computations are for a non-MMU system,but it should be noted that inclusion of the MMU will have no adverse effects on timing constraints. For further details see the NS16000 Timing Specifications document.

4.2 Summary of Timing Requirements from Memories/Peripherals.

Setup and Hold Times provided by MPU system

Time	Description	Nobuff	Buff
tCSsRD	Min CS Setup time before RD	33	16
tAsRD	Min Address Setup time before RD	45	18
tCSsWR	Min CS Setup time before WR	33	16
tAsWR	Min Address Setup time before WR	45	18
tCShRD	Min CS Hold time after RD	65	55
tCShWR	Min CS Hold time after WR	65	55
tAhRD	Min Address Hold time after RD	65	55
tAhWR	Min Address Hold time after WR	65	55
tRDw	Min RD Pulse width*	165	158
tWRw	Min WR Pulse width*	165	158
tDsWRia	Min Data Setup time before WR TE*	170	85
tDhWR	Min Data Hold time after WR TE	15	5
tRDn	Min delay from RD to next command	165	155
tWRn	Min delay from WR to next command	165	155
tDsWRv	Min Data Setup time before WR LE	0	---

Peripheral/Memory Responses required by MPU system

Time	Description	Nobuff	Buff
tRDaDv	Max delay from RD Active to Data Valid*	128	68
tDhRDia	Min Data Hold time after RD Inactive	0	0
tRDiaDf	Max delay from RD Inactive to Data Float	80	155
tCSaDv	Max delay from CS Active to Data Valid*	186	119
tAvDv	Max delay from Address Valid to Data Valid*	198	143

LE Leading Edge
TE Trailing Edge
* Assumes no Wait States inserted

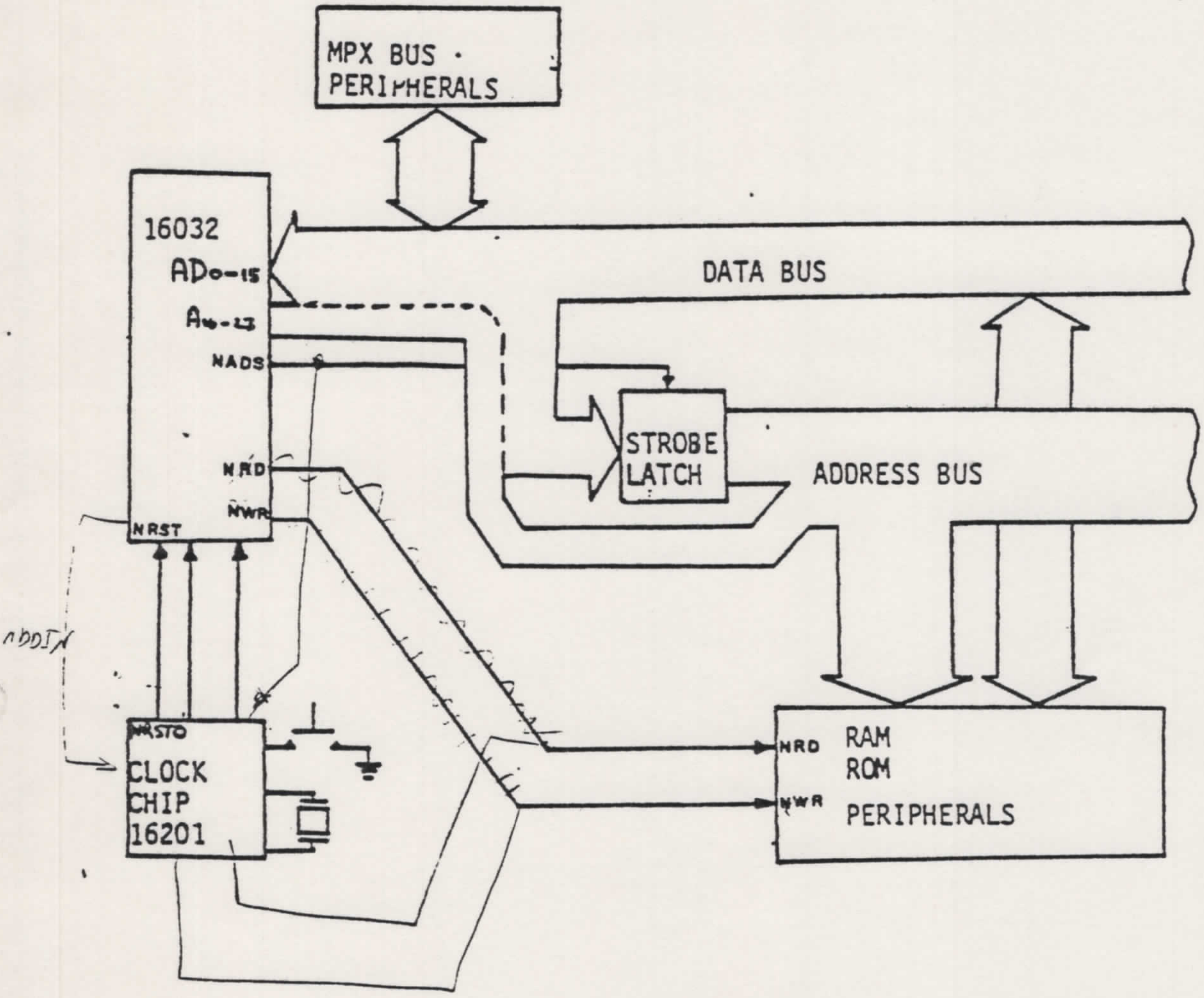


Fig. 8 Basic Nonbuffered System

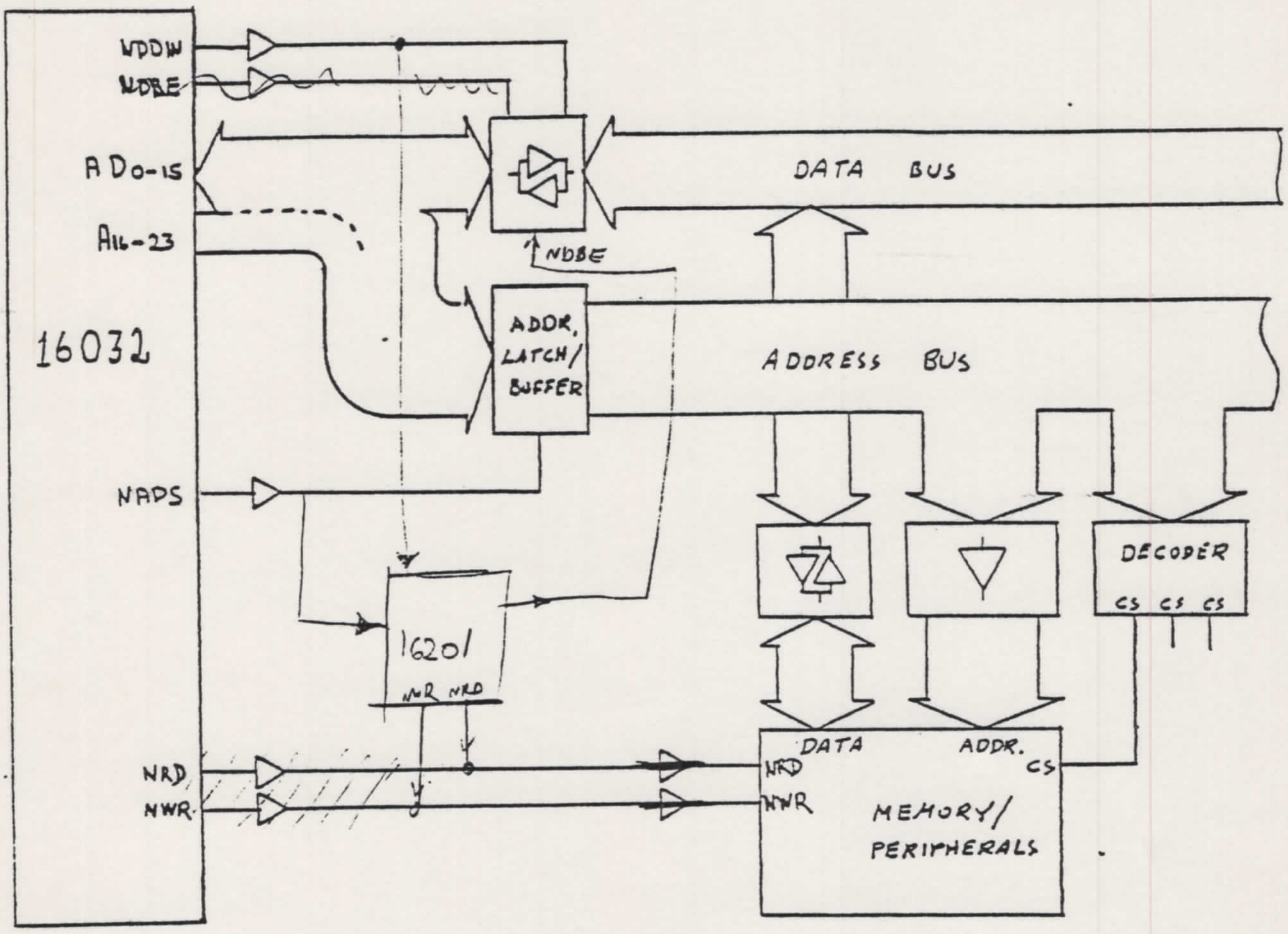


Fig. 9 Basic Buffered System

4.3 Summmary of Timing Requirements from Slave Processors

Setup and Hold Times provided by MPU system

Time	Description		time(ns)
tSsSPCa	Min Status Setup time to SPC active		58
tSPCw	Min SPC width		82
tDsSPCia	Min Data Setup time to SPC TE	[write]	67
tDhSPCia	Min Data Hold time after SPC TE	[write]	30
tShSPCa	Min Status Hold time after SPC LE		87
tSsDv	Min Status Setup time to Data valid	[write]	63
tSsSPCia	Min Status Setup time to SPC inactive		160

Slave Processor Responses required by MPU system

Time	Description		time(ns)
tSPCaDv	Max SPC active to Data Valid delay	[read]	48
tDhSPCia	Min Data Hold time after SPC TE	[read]	0
tSPCiaDf	Max delay from SPC TE to Data Float	[read]	80
tSvDv	Max delay from Status valid to Data valid	[read]	126
tSSPCa	Min SP NSPC active time*		40
tSSPCaSSPCf	Max delay from SP NSPC active to NSPC float*		200

LE Leading Edge
TE Trailing Edge
* Refers to NSPC generated by Slave Processors.