NS16016

~~NS168032~~ MICROPROCESSOR
TARGET - SPECIFICATIONS

REVISION A0
15-SEP-1980

COMPANY CONFIDENTIAL
DO NOT REPRODUCE

NATIONAL SEMICONDUCTOR ISRAEL Ltd.

16016

TABLE OF CONTENTS

16016

APPLICABLE DOCUMENTS

1. NS 16000 uP Family Architecture Description.
2. NS16032 Hardware Target Specifications.
3. The NS16000 Family Overview.
4. The NS16000 Family Timing Specifications.
5. NS16032 Architecture Specificiations
6. NS16081 Floating Point Processor specifications
7. NS16082 Memory Management Unit specifications
8. NS16201 Clock Chip specifications
9. NS16202 Interrupt Control Unit specifications
10. NS16203 DMA Control Unit specifications.
11. NS8080 MANUALS.

CHAPTER 1

INTRODUCTION

The NS168032 is an NS16032 super-set - it is software and pin-to-pin hardware compatible with the NS16032, and, in addition, it can be set in the 8080 EMULATION MODE, where it can execute the NS8080 instruction-set at high speed.

The NS168032 is aimed at the micro-computer user who is willing to move to the NS16000 advanced architecture, but at the same time wants to preserve his old 8080 software, in which he had invested a lot of time and efforts.

CHAPTER 2

HARDWARE

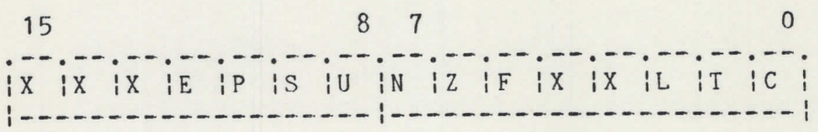
In all hardware respects the NS168032 is identical to the NS16032. Therefore, the NS16032 Hardware Target Specifications can be used for the NS168032 hardware description as well. Moreover, the NS168032 interfaces to other NS16000 family members (NS16201, NS16202, NS16203, NS16081, NS16082 and others) in the same way as the NS16032 does.

CHAPTER 3

NATIVE MODE ARCHITECTURE

When the NS168032 is in Native Mode, its architecture is identical to that of the NS16032 (see NS16032 Architecture Specifications), with the following additions:

1. In the Supervisor part of the PSR there is an additional flag - E , which indicates that Emulation mode is on. The format of the PSR is as follows:



2. Opcode (TBD), which is TRAP UNDEFINED in the NS16032, is TO80 - an instruction which switches the NS168032 to the 8080 Mode. This instruction sets the PSR E bit, and freezes a data and a code segment for the following 8080 instructions (to be detailed in the next chapter). Note that this is the only way in which a user can enter 8080 Emulation mode without Supervisory help.

There are three ways to enter 8080 Emulation mode :

1. Using the TO80 instruction, as was mentioned above.
2. Using a RETTRAP or RETI instruction which modifies the E bit in the Supervisor PSR.

16016

3. Using a BIT-SET PSR or a LPR PSR instruction.

Notes:

1. Only the first way can be used while in User Mode.

2. The second way is used mostly for "Virtual 8080" (see below) and for single-stepping an 8080 program.

3. The third way is illegal - it will put the NS16016 in the 8080 mode, but data is not guaranteed to be fetched from the 64K segment pointed at by the SP (see next chapter).

CHAPTER 4

8080 ARCHITECTURE

4.1 SCOPE OF EMULATION

The Interrupt and I/O structure of 8080 systems is different in terms of hardware from the NS16000 structure. I/O and Interrupts will, therefore, be handled in Native Mode only. The NS16032 Emulation Mode has, consequently, User Mode restrictions i.e.- it cannot execute System-related instructions. (Even if the PSR U bit is reset, which may be the case for a Supervisor-initiated 8080 routine).

The scope of 8080 Emulation is, therefore, the full NS8080 instruction set with the exception of EI, DI, IN and OUT.

4.2 "VIRTUAL" 8080

There are two alternative solutions For those 8080 programs which use the prohibited four instructions:

A. Modify the 8080 program by substituting the four instructions with appropriate Supervisor-calls.

B. use a Virtual Machine technique - since an attempt to execute IN, OUT, EI or DI will cause a TRAP ILLEGAL, and transfer to Native (Supervisor) mode, an appropriate system routine can simulate the desired 8080 instruction. The disassembly should be fairly easy - it is only four instrctions; EI and DI are one byte long, IN and OUT are one byte OPCODE plus one byte for the Port number.

Interrupts also result in a transfer to Native Mode, so that a Virtual 8080 monitor may be able to simulate the 8080 interrupt structure.

Note that just like in Native Mode, traps result in pushing the

current instruction PC, and interrupts result in pushing the PC of the next instruction.

4.3 VIRTUAL MEMORY

Just like the Native mode, the 8080 mode will fully support Virtual Memory - this means that ABORT is supported in such a way as to enable reexecution.

4.4 ADDRESS SPACE SEGMENTATION

8080 Address space is generally 64K byte. The NS16032, however, can address 16M bytes in Native mode. In order to improve 8080 support , the 16032 can provide for each 8080 program 64K for code, and 64K for data. This is done by the following mechanism: when 8080 mode is entered either by the TO80 instruction or by a RETI/RETRAP, SP bits 16 to 23 are frozen, and are used during the emulation for the high order part of all data references (SP,HL and indirect). In the case of RETI/RETTRAP , where the the S bit can also be changed by the same instruction, the SP to be used is the one after the RET is executed.

On the other hand, all Code and Immediates are fetched with PC bits 16 to 23 having the value they had when Emulation Mode was entered. Thus it is possible to have separate segments for code and data.

Self-modifying programs, which cannot have seperate Code and Data segments, can use the same 64K segment.

Multiple non-relocatable 8080 programs can also be supported, by assigning a different 64K byte segment (virtual) for each.

4.5 TARNSFERRING DATA THROUGH REGISTERS

When 8080 mode is on, 8080 registers are the low-order byte of the native registers, with the following translation:

B = R0, C = R1

D = R2, E = R3

H = R4, L = R5

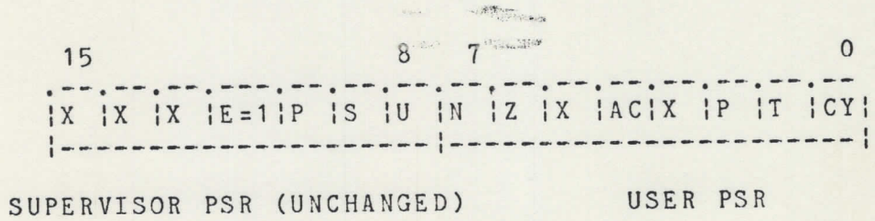
A = R7

R6, FP, SB, MOD, and INTBASE remain unchanged during the emulation.

Note that there is no guarantee to the contents of the higher order part of those user registers which serve as 8080 registers during emulation.

4.6 8080 FLAGS

When the NS168032 is in Emulation mode, it uses the PSR with the following translation:



4.7 GOING FROM 8080 MODE TO NATIVE

OPCODE (TBD), which is NOOP in Native mode, UNDEFINED in NS8080 instruction-set, is defined to be TO16 - an instruction which transfers the NS168032 from Emulation mode to Native mode. All it does is change the E bit in the PSR. The next insrtuction must be a Native mode instruction.