HEWLETT PACKARD

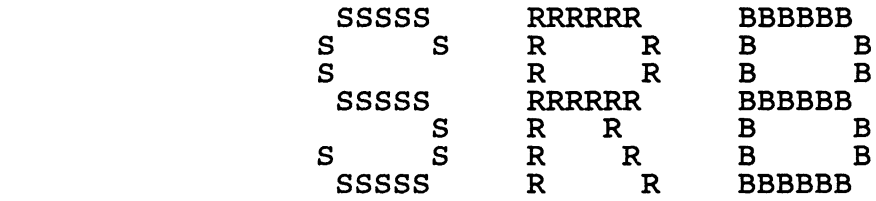
HEWLETT-PACKARD COMPANY
LOGIC SYSTEMS DIVISION

HP 64000 Logic Development System

SYSTEM RELEASE BULLETIN

Part Number: 5958-6019
0287

Printed: FEBRUARY 1987



SYSTEM RELEASE BULLETIN

64000 Logic Development System

MARCH 1987

This System Release Bulletin (SRB) documents all fixes and enhancements that are incorporated in the latest release of software for the 64000 Logic Development System.

The SRB is provided as a benefit of Hewlett-Packard's Software Support Services.

The five sections of the SRB are:

SOFTWARE RELEASE CONTENTS - lists the new revision codes for the 64000 products.

PRODUCT INDEX - lists product names and numbers which are included in this issue.

KPR NUMBER INDEX - sequential list of SR numbers.

KEYWORD INDEX - brief description of each SR.

KNOWN PROBLEM REPORTS - the actual reports.

Software release contents

Product name	Product number	uu.ff
*64000-UX OP-ENV 300	64801S004	01.40
*68000 EMUL 12.5 MHZ	64243	01.01
*68000 EMULATION 300	64242S004	01.00
*68008 EMULATION	64244	01.01
*68010 EMUL 12.5 MHZ	64245	01.01
*80186 EMULATION 300	64224S004	01.00
*80286B ASSEMB	64859	01.01
*80286B ASSEMB 300	64859S004	01.10
*80286B ASSEMB 500	64859S001	01.10
*8051 EMULATION 300	64264S004	01.00
*8085 EMULATION 300	64203S004	01.00
*8086 DQ EMULATION	64220	01.02
*8088 DQ EMULATION	64221	01.02
*INVERSE ASSEMB	64856	01.01
*INVERSE ASSEMB	64856S004	01.00
*OPERATING SYSTEM	64100	02.05
*OP_SYS HP-UX / 300	64883	01.00
*PROM PROGRAMMER 300	64501S004	01.00
*RS-232 TRANSFER 300	64885	01.20
*SOFTKEY EDITOR 300	64790S004	01.10
*SOFTKEY EDITOR 500	64790S001	01.10
*STATE ANALYZER 300	64620S004	01.00
*USER INTERFACE 300	64808S004	01.20
*USER INTERFACE 500	64808S001	01.20
*Z80H EMULATION 300	64253S004	01.00

Product index

Report number index

Keyword index

- 68000 EMUL 12.5 MHZ -

- 68008 EMULATION -

- 68010 EMUL 12.5 MHZ -

- 80286B ASSEMB -

- 80286B ASSEMB -

- 80286B ASSEMB -

Keyword index

- 8086 DQ EMULATION -

- 8088 DQ EMULATION -

- OPERATING SYSTEM -

Number: 5000118455 Product: 68000 EMUL 12.5 MHZ 64243 01.00

One-line description:
Incorrect Setting of the Interrupt Mask while in the Monitor.

Problem:
In the monitor listing at line 162, the instruction "AND #0FEFFH,SR" is used to set the interrupt mask to level 6. This will work only if the mask was set to level 7 upon entry into the monitor. The following monitor entry points do not insure that the mask is preset to 7: SPECIAL_ENTRY, SWBK_ENTRY, and JSR_ENTRY. The following monitor entry points will operate correctly: MONITOR_ENTRY, JUMP_ENTRY, and RESET_ENTRY. For example, if the monitor is entered with the interrupt mask set to level 1, the "AND" instruction will clear the mask to 0, thus permitting all interrupts.

The monitor should be changed to check the value of the mask upon entry, and then use the MOVE, OR, and/or AND instructions to set the mask to level 6.

Temporary solution:
Remove the AND instruction or precede it with an OR instruction to set the mask to level 6.

Signed off 01/12/87 in release 301.01

Number: 5000134627 Product: 68000 EMUL 12.5 MHZ 64243 01.00

One-line description:
Incorrect Inverse Assembly of Address Register Indirect with Index Mode.

Problem:
The assembler syntax of the address register indirect with index addressing mode is d(An,Rn). Rn (the index register) can be either an address or a data register. If the code uses an address register for Rn, the disassembler incorrectly displays the data register of the same number. Refer to the following example:

```
Source Code:        MOVE.W  0[A0,A1].D0
Disassembled Code:  MOVE.W  000H[A0,D1.W],D0
                                ^----------incorrect value
```

Note that the index register was incorrectly disassembled as D1 instead of A1.

Signed off 01/12/87 in release 301.01

Number: 5000136259 Product: 68000 EMUL 12.5 MHZ 64243 01.00

One-line description:
Incorrect Disassembly of the MULU Instruction with Immediate Operand.

Problem:
The MULU instruction (with an immediate operand) disassembles incorrectly if it is preceded by a long word instruction. The following example will illustrate the problem.

1) The assembler listing file:

```
001000  263C       MOVE.L   #10,D3
001002  00000000A
001006  C6FC 0005  MULU     #5,D3
00100A  4E71       NOP
```

2) The disassembled code:

```
001000     MOVE.L  #00000000AH,D3
001006     MULU.W  #000054E71H,D3
```

Note that the immmediate operand of the MULU instruction is incorrect. The disassembler has included the two bytes following the MULU instruction (4E71) as a part of the immediate operand. This could cause the incorrect disassembly of instructions following the MULU instruction because the disassembler might get out of synch.

Signed off 01/12/87 in release 301.01

Number: D200038067 Product: 68000 EMUL 12.5 MHZ 64243 01.00

One-line description:
Inappropriate Error Message When Loading User Program Without Monitor

Problem:
An error message appears on the status line if a user program is loaded into memory without having the monitor linked with it. This error message is:

ERROR: One or more monitor variables not mapped to emulation memory

The load is successful even though this error message appears. An error message should not appear because there is no error in the loading procedure. A more appropriate indicator that the monitor was not loaded into memory is desired.

Signed off 01/12/87 in release 301.01

Number: D200047753 Product: 68000 EMUL 12.5 MHZ 64243 01.00

One-line description:
The "occurs" guided-syntax softkey is missing for certain commands.

Problem:
An error occurs in the guided-syntax softkeys. Specifically, the ‹occurs› softkey is missing when you give the command "trace about data XX". The softkeys that appear are ‹,› ‹status› ‹or› ‹only› ‹counting› & ‹break_on›. The ‹occurs› syntax is an option that is missing after the "data XX" token. Note, if you type "occurs" everything works fine. I would expect to see the following softkeys after the "data XX" token: ‹,› ‹status› ‹occurs› ‹or› ‹only› ‹counting› ‹break_on›.

Temporary solution:
If the "occurs" option is manually typed in the appropriate location, the trace command will operate correctly.

Signed off 01/12/87 in release 301.01

Number: D200055244 Product: 68000 EMUL 12.5 MHZ 64243 01.00

One-line description:
Incorrect Inverse Assembly occurs with a Non-64302A Analyzer.

Problem:
"display memory mnemonic" shows incorrect inverse assembly when using a non-64302A analyzer and displaying memory above address 0FFFFH. Given the code BSR.W 5000H located at address 5006EH, the mnemonic disassembly of memory at address 5006EH will show BSR.W 0000H. This failure occurs because the analyzer i.d. is being defaulted to a 64300A which only uses 16 address lines.

Signed off 01/12/87 in release 301.01

Number: D200038083 Product: 68008 EMULATION 64244 01.00

One-line description:
Inappropriate Error Message When Loading User Program Without Monitor

Problem:
An error message appears on the status line if a user program is loaded into memory without having the monitor linked with it. This error message is:
ERROR: One or more monitor variables not mapped to emulation memory
The load is successful even though this error message appears. An error message should not appear because there is no error in the loading procedure. A more appropriate indicator that the monitor was not loaded into memory is desired.

Signed off 01/12/87 in release 401.01

Number: D200046045 Product: 68008 EMULATION 64244 01.00

One-line description:
Incorrect Setting of the Interrupt Mask while in the Monitor.

Problem:
In the monitor listing at line 163, the instruction "AND #0FEFFH,SR" is used to set the interrupt mask to level 6. This will work only if the mask was set to level 7 upon entry into the monitor. The following monitor entry points do not insure that the mask is preset to 7: SPECIAL_ENTRY, SWBK_ENTRY, and JSR_ENTRY. The following monitor entry points will operate correctly: MONITOR_ENTRY, JUMP_ENTRY, and RESET_ENTRY. For example, if the monitor is entered with the interrupt mask set to level 1, the "AND" instruction will clear the mask to 0, thus permitting all interrupts.

The monitor should be changed to check the value of the mask upon entry, and then use the MOVE, OR, and/or AND instructions to set the mask to level 6.

Temporary solution:
Remove the AND instruction or precede it with an OR instruction to set the mask to level 6.

Signed off 01/12/87 in release 401.01

Number: D200047803 Product: 68008 EMULATION 64244 01.00

One-line description:
The "occurs" guided-syntax softkey is missing for certain commands.

Problem:
An error occurs in the guided-syntax softkeys. Specifically, the ‹occurs› softkey is missing when you give the command "trace about data XX". The softkeys that appear are ‹,› ‹status› ‹or› ‹only› ‹counting› & ‹break_on›. The ‹occurs› syntax is an option that is missing after the "data XX" token. Note, if you type "occurs" everything works fine. I would expect to see the following softkeys after the "data XX" token: ‹,› ‹status› ‹occurs› ‹or› ‹only› ‹counting› ‹break_on›.

Temporary solution:
If the "occurs" option is manually typed in the appropriate location, the trace command will operate correctly.

Signed off 01/12/87 in release 401.01

Number: D200050914 Product: 68008 EMULATION 64244 01.00

One-line description:
"Store memory" command creates 16 bit wordsize rather than 8 bit

Problem:
Using the "store memory <ADDRESS> to <FILE>" command creates a word size of 16 bits. In contrast, the assembler/linker creates the correct wordsize of 8 bits in the absolute file. This causes problems when using the output of the "store memory" command for programming proms.

Temporary solution:
As a workaround you use the "list memory <ADDRESS> to <FILE>" command to create a listing file. Then modify it using a command file that creates exactly the same format as the "list_rom_to <FILE>". Use this modified listing file for programming the prom.

Signed off 01/12/87 in release 401.01

Number: D200060368 Product: 68008 EMULATION 64244 01.00

One-line description:
Incorrect Inverse Assembly of Address Register Indirect with Index Mode.

Problem:
The assembler syntax of the address register indirect with index addressing mode is d(An,Rn). Rn (the index register) can be either an address or a data register. If the code uses an address register for Rn, the disassembler incorrectly displays the data register of the same number. Refer to the following example:

```
Source Code:        MOVE.W  0[A0,A1].D0
Disassembled Code:  MOVE.W  000H[A0,D1.W],D0
                                 ^----------incorrect value
```

Note that the index register was incorrectly disassembled as D1 instead of A1.

Signed off 01/12/87 in release 401.01

Number: D200060384 Product: 68008 EMULATION 64244 01.00

One-line description:
Incorrect Disassembly of the MULU Instruction with Immediate Operand.

Problem:
The MULU instruction (with an immediate operand) disassembles incorrectly if it is preceded by a long word instruction. The following example will illustrate the problem.

1) The assembler listing file:

```
001000  263C       MOVE.L   #10,D3
001002  00000000A
001006  C6FC 0005  MULU     #5,D3
00100A  4E71       NOP
```

2) The disassembled code:

```
001000      MOVE.L  #00000000AH,D3
001006      MULU.W  #000054E71H,D3
```

Note that the immmediate operand of the MULU instruction is incorrect. The disassembler has included the two bytes following the MULU instruction (4E71) as a part of the immediate operand. This could cause the incorrect disassembly of instructions following the MULU instruction because the disassembler might get out of synch.

Signed off 01/12/87 in release 401.01

Number: D200060418 Product: 68008 EMULATION 64244 01.00

One-line description:
Incorrect Inverse Assembly occurs with a Non-64302A Analyzer.

Problem:
"display memory mnemonic" shows incorrect inverse assembly when using a non-64302A analyzer and displaying memory above address 0FFFFH. Given the code BSR.W 5000H located at address 5006EH, the mnemonic disassembly of memory at address 5006EH will show BSR.W 0000H. This failure occurs because the analyzer i.d. is being defaulted to a 64300A which only uses 16 address lines.

Signed off 01/12/87 in release 401.01

Number: D200038075 Product: 68010 EMUL 12.5 MHZ 64245 01.00

One-line description:
Inappropriate Error Message When Loading User Program Without Monitor

Problem:
An error message appears on the status line if a user program is loaded into memory without having the monitor linked with it. This error message is:
ERROR: One or more monitor variables not mapped to emulation memory
The load is successful even though this error message appears. An error message should not appear because there is no error in the loading procedure. A more appropriate indicator that the monitor was not loaded into memory is desired.

Signed off 01/12/87 in release 501.01

Number: D200046060 Product: 68010 EMUL 12.5 MHZ 64245 01.00

One-line description:
Incorrect Setting of the Interrupt Mask while in the Monitor.

Problem:
In the monitor listing at line 166, the instruction "AND #0FEFFH,SR" is used to set the interrupt mask to level 6. This will work only if the mask was set to level 7 upon entry into the monitor. The following monitor entry points do not insure that the mask is preset to 7: SPECIAL_ENTRY, SWBK_ENTRY, and JSR_ENTRY. The following monitor entry points will operate correctly: MONITOR_ENTRY, JUMP_ENTRY, and RESET_ENTRY. For example, if the monitor is entered with the interrupt mask set to level 1, the "AND" instruction will clear the mask to 0, thus permitting all interrupts.

The monitor should be changed to check the value of the mask upon entry, and then use the MOVE, OR, and/or AND instructions to set the mask to level 6.

Temporary solution:
Remove the AND instruction or precede it with an OR instruction to set the mask to level 6.

Signed off 01/12/87 in release 501.01

Number: D200047795 Product: 68010 EMUL 12.5 MHZ 64245 01.00

One-line description:
The "occurs" guided-syntax softkey is missing for certain commands.

Problem:
An error occurs in the guided-syntax softkeys. Specifically, the ‹occurs› softkey is missing when you give the command "trace about data XX". The softkeys that appear are ‹,› ‹status› ‹or› ‹only› ‹counting› & ‹break_on›. The ‹occurs› syntax is an option that is missing after the "data XX" token. Note, if you type "occurs" everything works fine. I would expect to see the following softkeys after the "data XX" token: ‹,› ‹status› ‹occurs› ‹or› ‹only› ‹counting› ‹break_on›.

Temporary solution:
If the "occurs" option is manually typed in the appropriate location, the trace command will operate correctly.

Signed off 01/12/87 in release 501.01

Number: D200060376 Product: 68010 EMUL 12.5 MHZ 64245 01.00

One-line description:
Incorrect Inverse Assembly of Address Register Indirect with Index Mode.

Problem:
The assembler syntax of the address register indirect with index addressing mode is d(An,Rn). Rn (the index register) can be either an address or a data register. If the code uses an address register for Rn, the disassembler incorrectly displays the data register of the same number. Refer to the following example:

```
Source Code:        MOVE.W  0[A0,A1].D0
Disassembled Code:  MOVE.W  000H[A0,D1.W],D0
                                 ^----------incorrect value
```

Note that the index register was incorrectly disassembled as D1 instead of A1.

Signed off 01/12/87 in release 501.01

Number: D200060400 Product: 68010 EMUL 12.5 MHZ 64245 01.00

One-line description:
Incorrect Disassembly of the MULU Instruction with Immediate Operand.

Problem:
The MULU instruction (with an immediate operand) disassembles incorrectly if it is preceded by a long word instruction. The following example will illustrate the problem.

1) The assembler listing file:

```
001000  263C       MOVE.L   #10,D3
001002  00000000A
001006  C6FC 0005  MULU     #5,D3
00100A  4E71       NOP
```

2) The disassembled code:

```
001000      MOVE.L  #00000000AH,D3
001006      MULU.W  #000054E71H,D3
```

Note that the immmediate operand of the MULU instruction is incorrect. The disassembler has included the two bytes following the MULU instruction (4E71) as a part of the immediate operand. This could cause the incorrect disassembly of instructions following the MULU instruction because the disassembler might get out of synch.

Signed off 01/12/87 in release 501.01

Number: D200060426 Product: 68010 EMUL 12.5 MHZ 64245 01.00

One-line description:
Incorrect Inverse Assembly occurs with a Non-64302A Analyzer.

Problem:
"display memory mnemonic" shows incorrect inverse assembly when using a non-64302A analyzer and displaying memory above address 0FFFFH. Given the code BSR.W 5000H located at address 5006EH, the mnemonic disassembly of memory at address 5006EH will show BSR.W 0000H. This failure occurs because the analyzer i.d. is being defaulted to a 64300A which only uses 16 address lines.

Signed off 01/12/87 in release 501.01

Number: D200055343 Product: 80286B ASSEMB 64859 01.00

Keywords: CODE GENERATOR

One-line description:
Obj. code generated for arithmetic instr. are incorrect.

Problem:
The object code produced for the arithmetic instructions FADD,FDIV, FDIVP, FDIVR, FDIVRP, FMUL, FSUB, FSUBP, FSUBR, FSUBRP is not correct. These problems occur only with the code generated for the 80287 coprocessor. The 8087 processor was changed in Feb. 1984. The opcodes generated are for 8087 processors manufactured prior to Feb. 1984.

instruction		opcode - valid prior to FEB. 1984
FADD		DCC1
FDIV		DCF1
FDIV	ST[3],ST	DCF3
FDIVP	ST[4],ST	DEF4
FDIVR		DCF9
FDIVR	ST[4],ST	DCFC
FDIVRP	ST[1],ST	DEF9
FMUL		DCC9
FSUB		DCE1
FSUB	ST[1],ST	DCE2
FSUBP	ST[2],ST	DEE2
FSUBR		DCE9
FSUBR	ST[1],ST	DCE9
FSUBRP	ST[1],ST	DEE9

The pseudo instruction NEW_8087 will cause the correct opcodes to be generated. This assembler should default to the new opcodes without the pseudo instruction.

Temporary solution:
The "NEW_8087" pseudo instruction should be included in any program using the 80287 arithmetic instructions. This pseudo instruction should precede all 80287 instructions.

ex.

```
"80286B"

NEW_8087

PROG

DEC1        FADD
DEF9        FDIV
DCFB        FDIV    ST[3],ST
DEFC        FDIVP   ST[4],ST
DEF1        FDIVR
DCF4        FDIVR   ST[4],ST
DEF1        FDIVRP  ST[1],ST
DEC9        FMUL
DEE9        FSUB
DCE9        FSUB    ST[1],ST
DEEA        FSUBP   ST[2],ST
```

```
DEE1            FSUBR
DCE1            FSUBR   ST[1],ST
DEE1            FSUBRP  ST[1],ST
```

Signed off 01/12/87 in release 901.01

Number: D200055368 Product: 80286B ASSEMB 64859 01.00

Keywords: CODE GENERATOR

One-line description:
FSTENV instruction generates object code without required wait instr

Problem:
The object code for the FSTENV instruction is missing the required wait instruction. The code generated is D936001A, it should be 9BD936001A.

Temporary solution:
Precede all FSTENV instructions with the WAIT instruction.

Signed off 01/12/87 in release 901.01

Number: D200055376 Product: 80286B ASSEMB 64859 01.00

Keywords: CODE GENERATOR

One-line description:
FSTSW/FNSTSW function incorrectly with two-byte memory operand

Problem:
FSTSW/FNSTSW (Store 80287 Status Word) instruction incorrectly results in Invalid Operand Error when used with two-byte memory operand. This instruction should accept a two-byte memory operand.

Temporary solution:
The FSTSW AX or FNSTSW AX versions of the FSTSW/FNSTSW instructions can be used followed immediately by the MOV mem,AX instruction.

Signed off 01/12/87 in release 901.01

Number: D200055426 Product: 80286B ASSEMB 300 64859S004 01.00

Keywords: CODE GENERATOR

One-line description:
FSTSW/FNSTSW function incorrectly with two-byte memory operand

Problem:
FSTSW/FNSTSW (Store 80287 Status Word) instruction incorrectly results in Invalid Operand Error when used with two-byte memory operand. This instruction should accept a two-byte memory operand.

Temporary solution:
The FSTSW AX or FNSTSW AX versions of the FSTSW/FNSTSW instructions can be used followed immediately by the MOV mem,AX instruction.

Number: D200055459 Product: 80286B ASSEMB 300 64859S004 01.00

Keywords: CODE GENERATOR

One-line description:
FSTENV instruction generates object code without required wait instr

Problem:
The object code for the FSTENV instruction is missing the required wait instruction. The code generated is D936001A, it should be 9BD936001A.

Temporary solution:
Precede all FSTENV instructions with the WAIT instruction.

Number: D200055483 Product: 80286B ASSEMB 300 64859S004 01.00

Keywords: CODE GENERATOR

One-line description:
Obj. code generated for arithmetic instr. are incorrect.

Problem:
The object code produced for the arithmetic instructions FADD,FDIV, FDIVP, FDIVR, FDIVRP, FMUL, FSUB, FSUBP, FSUBR, FSUBRP is not correct. These problems occur only with the code generated for the 80287 coprocessor. The 8087 processor was changed in Feb. 1984. The opcodes generated are for 8087 processors manufactured prior to Feb. 1984.

instruction		opcode - valid prior to FEB. 1984
FADD		DCC1
FDIV		DCF1
FDIV	ST[3],ST	DCF3
FDIVP	ST[4],ST	DEF4
FDIVR		DCF9
FDIVR	ST[4],ST	DCFC
FDIVRP	ST[1],ST	DEF9
FMUL		DCC9
FSUB		DCE1
FSUB	ST[1],ST	DCE2
FSUBP	ST[2],ST	DEE2

```
          FSUBR                 DCE9
          FSUBR   ST[1],ST      DCE9
          FSUBRP  ST[1],ST      DEE9
```

The pseudo instruction NEW_8087 will cause the correct opcodes to be generated. This assembler should default to the new opcodes without the pseudo instruction.

Temporary solution:
The "NEW_8087" pseudo instruction should be included in any program using the 80287 arithmetic instructions. This pseudo instruction should precede all 80287 instructions.

ex. "80286B"

```
                    NEW_8087

                    PROG

        DEC1        FADD
        DEF9        FDIV
        DCFB        FDIV    ST[3],ST
        DEFC        FDIVP   ST[4],ST
        DEF1        FDIVR
        DCF4        FDIVR   ST[4],ST
        DEF1        FDIVRP  ST[1],ST
        DEC9        FMUL
        DEE9        FSUB
        DCE9        FSUB    ST[1],ST
        DEEA        FSUBP   ST[2],ST
        DEE1        FSUBR
        DCE1        FSUBR   ST[1],ST
        DEE1        FSUBRP  ST[1],ST
```


Number: D200055400 Product: 80286B ASSEMB 500 64859S001 01.00

Keywords: CODE GENERATOR

One-line description:
FSTSW/FNSTSW function incorrectly with two-byte memory operand

Problem:
FSTSW/FNSTSW (Store 80287 Status Word) instruction incorrectly results in Invalid Operand Error when used with two-byte memory operand. This instruction should accept a two-byte memory operand.

Temporary solution:
The FSTSW AX or FNSTSW AX versions of the FSTSW/FNSTSW instructions can be used followed immediately by the MOV mem,AX instruction.

Number: D200055434 Product: 80286B ASSEMB 500 64859S001 01.00

Keywords: CODE GENERATOR

One-line description:
FSTENV instruction generates object code without required wait instr

Problem:
The object code for the FSTENV instruction is missing the required wait instruction. The code generated is D936001A, it should be 9BD936001A.

Temporary solution:
Precede all FSTENV instructions with the WAIT instruction.

Number: D200055467 Product: 80286B ASSEMB 500 64859S001 01.00

Keywords: CODE GENERATOR

One-line description:
Obj. code generated for arithmetic instr. are incorrect.

Problem:
The object code produced for the arithmetic instructions FADD,FDIV, FDIVP, FDIVR, FDIVRP, FMUL, FSUB, FSUBP, FSUBR, FSUBRP is not correct. These problems occur only with the code generated for the 80287 coprocessor. The 8087 processor was changed in Feb. 1984. The opcodes generated are for 8087 processors manufactured prior to Feb. 1984.

instruction		opcode - valid prior to FEB. 1984
FADD		DCC1
FDIV		DCF1
FDIV	ST[3],ST	DCF3
FDIVP	ST[4],ST	DEF4
FDIVR		DCF9
FDIVR	ST[4],ST	DCFC
FDIVRP	ST[1],ST	DEF9
FMUL		DCC9
FSUB		DCE1
FSUB	ST[1],ST	DCE2
FSUBP	ST[2],ST	DEE2

```
          FSUBR                DCE9
          FSUBR   ST[1],ST     DCE9
          FSUBRP  ST[1],ST     DEE9
```

The pseudo instruction NEW_8087 will cause the correct opcodes to be generated. This assembler should default to the new opcodes without the pseudo instruction.

Temporary solution:
The "NEW_8087" pseudo instruction should be included in any program using the 80287 arithmetic instructions. This pseudo instruction should precede all 80287 instructions.

ex. "80286B"

```
                    NEW_8087

                    PROG

        DEC1        FADD
        DEF9        FDIV
        DCFB        FDIV    ST[3],ST
        DEFC        FDIVP   ST[4],ST
        DEF1        FDIVR
        DCF4        FDIVR   ST[4],ST
        DEF1        FDIVRP  ST[1],ST
        DEC9        FMUL
        DEE9        FSUB
        DCE9        FSUB    ST[1],ST
        DEEA        FSUBP   ST[2],ST
        DEE1        FSUBR
        DCE1        FSUBR   ST[1],ST
        DEE1        FSUBRP  ST[1],ST
```


Number: D200046300 Product: 8086 DQ EMULATION 64220 01.01

One-line description:
Single stepping a noninterruptible instruction causes a display skip.

Signed off 01/12/87 in release 001.02

Number: D200065789 Product: 8086 DQ EMULATION 64220 01.01

Keywords: USER MEMORY

One-line description:
Emulator would not recover from errors during display memory repetitive.

Problem:
The problem occurs when displaying user memory repetitively. An error condition such as slow clock or guarded memory access would cause the 64000 station to reboot or to display extraneous data at the top of the screen. When the screen had been written to at the top, the only action to delete the characters was resetting the station.

Temporary solution:
There is no workaround other than avoiding the error conditions during a repetitive display of user memory.

Signed off 01/12/87 in release 001.02

Number: 5000151043 **Product:** 8088 DQ EMULATION 64221 01.01

Keywords: STATUS

One-line description:
Softkeys for segment status are set up incorrectly

Temporary solution:
Use the binary form for segment status specification instead of the softkeys.

Signed off 01/12/87 in release 101.02

Number: D200062869 Product: 8088 DQ EMULATION 64221 01.01

One-line description:
Store Memory is incorrect for some bock sizes.

Signed off 01/12/87 in release 101.02

Number: D200065763 Product: 8088 DQ EMULATION 64221 01.01

Keywords: USER MEMORY

One-line description:
Emulator would not recover from errors during display memory repetitive.

Problem:
The problem occurs when displaying user memory repetitively. An error condition such as slow clock or guarded memory access would cause the 64000 station to reboot or to display extraneous data at the top of the screen. When the screen had been written to at the top, the only action to delete the characters was resetting the station.

Temporary solution:
There is no workaround other than avoiding the error conditions during a repetitive display of user memory.

Signed off 01/12/87 in release 101.02

Number: D200067553 **Product:** OPERATING SYSTEM 64100 02.04

Keywords: HIGH SPEED LINK

One-line description:
On heavily loaded systems with HSL, DISC DOWN messages appear.

Problem:
On a heavily loaded 64000 system with a High Speed Link, the message "BUS DISC 0 DOWN" may be frequently appearing on the 64000 status line. It happens most frequently with a SS/80 or CS/80 system disc.

Signed off 01/12/87 in release 002.05

5958-6019, February 1987
E0287

Printed in U.S.A.