GRADE A

MESC TECHNICAL NEWS

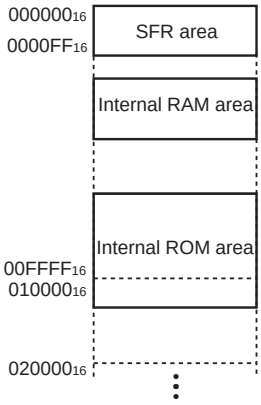
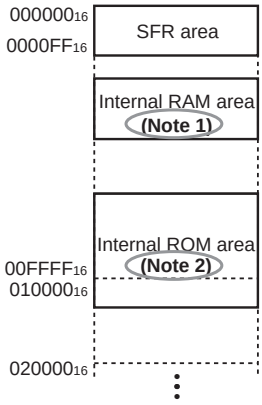
No. M7700-57-9911

Corrections and Supplementary Explanation for “7900 Series Software Manual” (REV.C)

This news includes a few corrections and supplementary explanation for “7900 Series Software Manual” (PDF File, REV.1.00).

And also, this news includes the information previously announced in MESC TECHNICAL NEWS (No. M7700-47-9906, Corrections and Supplementary Explanation for “7900 Series Software Manual” REV. B).
Note: ☆ represents the new information.

Corrections and Supplementary Explanation for "7900 Series Software Manual" (REV.C) NO.1

Page	Error	Correction
P2-7 Line 4 below Figure 2.1.4	This register is cleared to "0000_{16}" at reset.	At reset, DPR0 becomes "0000_{16}", and each of DPR1 to DPR3 becomes undefined.
P2-8 (3) Bit 2 Line 4	Use the **SEI** or **SEP** instruction to set this flag to "1," and use the **CLI** or **CLP** instruction to clear it to "0."	Be sure to use the **SEI** instruction to set this flag to "1"; and be sure to use the **CLI** or **CLP** instruction to clear this flag to "0."
P2-10 Figure 2.2.1	 **Note:** Memory assignment of internal RAM area varies according to the type of microcomputer. Refer to the latest catalogues or datasheets.	 **Notes 1:** When the internal RAM area is followed by an unused area or an external area, do not assign a program to the last 8 bytes of the internal RAM area. **2:** Do not assign a program to the last 8 bytes of the internal ROM area. **3:** The memory assignment of the internal area varies according to the product type. Refer to the latest datasheets or catalogs.
P2-16 Line 8	• Use direct page register 0 (DPR0) only. In this case, specify the offset from DPR0 in length of 8 bits. • Use direct page registers 0 through 3 (DPR0 through 3). In this case, use the high-order 2 bits of the operand (8 bits) to specify the direct page register and the low-order 6 bits to specify the offset.	• When the direct page register select bit = "0" Use direct page register 0 (DPR0) only. In this case, specify the offset from DPR0 in length of 8 bits. • When the direct page register select bit = "1" Use direct page registers 0 through 3 (DPR0 through 3). In this case, use the high-order 2 bits of the operand (8 bits) to specify the direct page register and the low-order 6 bits to specify the offset.
P2-16 Figure	<Direct addressing mode>	<When the direct page register select bit = "0">
P2-18 Figure	<Extension direct addressing mode>	<When the direct page register select bit = "1">

Corrections and Supplementary Explanation for “7900 Series Software Manual” (REV.C) NO.2

<table>
<tr><th>Page</th><th>Error</th><th>Correction</th></tr>
<tr><td>P2-19
P2-22
P2-23
P2-25
P2-28
P2-31
P2-33

Between the description of Function and Figure in the above pages</td><td></td><td>• When the direct page register select bit = “0”
Use direct page register 0 (DPR0) only.
In this case, specify the offset from DPR0 in length of 8 bits.
• When the direct page register select bit = “1”
Use direct page registers 0 through 3 (DPR0 through 3).
In this case, use the high-order 2 bits of the operand (8 bits) to specify the direct page register and the low-order 6 bits to specify the offset.

<When the direct page register select bit = “0”></td></tr>
<tr><td>☆
P2-48
Below the Figure</td><td></td><td>Note: When using the absolute long indexed X addressing mode, make sure that the last code of the instruction does not become the last address of a bank.</td></tr>
<tr><td>P2-56
Function
The last line</td><td>• BBCB and BBSB instructions
:
····· in bank 1₁₆ is specified.</td><td>• BBCB and BBSB instructions
:
••••in bank 1₁₆ is specified.

• When the direct page register select bit = “0”
Use direct page register 0 (DPR0) only.
In this case, specify the offset from DPR0 in length of 8 bits (the contents of the instruction’s third byte for the BBC and BBS instructions, and the contents of the instruction’s second byte for the BBCB and BBSB instructions).
• When the direct page register select bit = “1”
Use direct page registers 0 through 3 (DPR0 through 3).
In this case, use the high-order 2 bits of the operand (8 bits. The contents of the instruction’s third byte for the BBC and BBS instructions, and the contents of the instruction’s second byte for the BBCB and BBSB instructions) to specify the direct page register and the low-order 6 bits to specify the offset.

<When the direct page register select bit = “0”></td></tr>
<tr><td>P4-6
Flag manipulation</td><td><table><tr><th>Instruction</th><th>Description</th></tr><tr><td>SEP</td><td>PS_L (bit n) ← 1 (n = 0 to 7. Multiple operations can be specified.)</td></tr></table></td><td><table><tr><th>Instruction</th><th>Description</th></tr><tr><td>SEP</td><td>PS_L (bit n) ← 1 (n = 0, 1, 3 to 7. Multiple operations can be specified.)</td></tr></table></td></tr>
</table>

Corrections and Supplementary Explanation for “7900 Series Software Manual” (REV.C) NO.3

<table>
<tr><th>Page</th><th>Error</th><th>Correction</th></tr>
<tr><td>P4-6
Unconditional branch

P4-7
Subroutine</td><td>
<table>
<tr><th>Category</th><th>Instruction</th><th>Operation</th></tr>
<tr><td>Conditional branch</td><td>JMP</td><td>PC ← Destination address
PC ← mmll</td></tr>
<tr><td></td><td>JMPL</td><td>PG, PC ← Destination address
PC ← mmll
PG ← hh</td></tr>
<tr><td>Subroutine</td><td>★ BSR</td><td>Stack ← PC
PC ← PC + 2 + REL</td></tr>
<tr><td></td><td>JSR</td><td>Stack ← PC
PC ← Destination address
PC ← PC + 3
M(S, S – 1) ← PC
S ← S – 2
PC ← mmll</td></tr>
<tr><td></td><td>JSRL</td><td>Stack ← PG, PC
PG, PC ← Destination address
PC ← PC + 4
M(S, S – 2) ← PG, PC
S ← S – 3
PC ← mmll
PG ← hh</td></tr>
</table>
</td><td>
<table>
<tr><th>Category</th><th>Instruction</th><th>Operation</th></tr>
<tr><td>Uncondition-al branch</td><td>JMP</td><td>PC ← Destination address</td></tr>
<tr><td></td><td>JMPL</td><td>PG, PC ← Destination address</td></tr>
<tr><td>Subroutine</td><td>★ BSR</td><td>Stack ← PC + 2
PC ← PC + 2 + REL</td></tr>
<tr><td></td><td>JSR</td><td>Stack ← PC
PC ← Destination address</td></tr>
<tr><td></td><td>JSRL</td><td>Stack ← PG, PC
PG, PC ← Destination address</td></tr>
</table>
</td></tr>
<tr><td>P4-12
Contents of M (oprd)</td><td>
<table>
<tr><th>Symbol</th><th>Description</th></tr>
<tr><td>M(oprd)</td><td>Contents of memory location specified by operand</td></tr>
</table>
</td><td>
<table>
<tr><th>Symbol</th><th>Description</th></tr>
<tr><td>M(oprd)</td><td>Contents of memory location specified by operand in ( ), etc.</td></tr>
</table>
</td></tr>
<tr><td>P4-60
BSR
Operation</td><td>Stack ← PC
PC ← PC + 2 + REL</td><td>Stack ← PC + 2
PC ← PC + 2 + REL</td></tr>
<tr><td>P4-82
CMPD</td><td>Operation : E – IMM32

E
□□□□ – IMM32

Description : Subtracts the immediate value from the contents of E in 32-bit length. The result is not stored anywhere.</td><td>Operation : E – M32

E
□□□□ – M32

Description : Subtracts the contents of memory from the contents of E in 32-bit length. The result is not stored anywhere.</td></tr>
<tr><td>P4-94
DIVS
Operation</td><td>When m = “0”
(Figure)
✵ “S” represents MSB of data.

When m = “1”
(Figure)
✵ “S” represents MSB of data.</td><td>When m = “0”
(Figure)
✵ “S” is a sign bit and represents MSB of data.

When m = “1”
(Figure)
✵ “S” is a sign bit and represents MSB of data.</td></tr>
</table>

Corrections and Supplementary Explanation for “7900 Series Software Manual” (REV.C) NO.4

Page	Error	Correction
P4-111 JMP/JMPL **Operation**	• JMP instruction PC ← Specified address PC ← mmll • JMPL instruction PG, PC ← Specified address PC ← mmll PG ← hh	• JMP instruction PC ← Specified address PC ← mmll*1 *1: In the case of (ABS), M(mmll + 1, mmll). In the case of (ABS, X), M(mmll + X + 1, mmll + X). • JMPL instruction PG, PC ← Specified address PC ← mmll*2 PG ← hh*3 *2: In the case of L(ABS), M(mmll + 1, mmll). *3: In the case of L(ABS), M(mmll + 2).
P4-112 JSR/JSRL **Operation**	• JSR instruction Stack ← PC Stack ← Specified address PC ← PC + 3 M(S, S – 1) ← PC S ← S – 2 PC ← mmll • JSRL instruction Stack ← PG, PC PG, PC ← Specified address PC ← PC + 4 M(S to S – 2) ← PG, PC S ← S – 3 PC ← mmll PG ← hh	• JSR instruction Stack ← PC + 3 Stack ← Specified address PC ← PC + 3 M(S, S – 1) ← PC S ← S – 2 PC ← mmll* *: In the case of (ABS, X), M(mmll + X + 1, mmll + X). • JSRL instruction Stack ← PG, PC + 4 PG, PC ← Specified address PC ← PC + 4 M(S to S – 2) ← PG, PC S ← S – 3 PC ← mmll PG ← hh
P4-119 LDX **Description example**	CLM LDX.W #IMM16 ; X ← IMM16 LDX MEM16 ; X ← MEM16 SEM :	CLP x LDX.W #IMM16 ; X ← IMM16 LDX MEM16 ; X ← MEM16 SEP x :
P4-121 LDY **Description example**	CLM LDY.W #IMM16 ; X ← IMM16 LDY MEM16 ; X ← MEM16 SEM :	CLP x LDY.W #IMM16 ; X ← IMM16 LDY MEM16 ; X ← MEM16 SEP x :
P4-132 MPYS **Operation**	When m = “0” (Figure) ❋ “S” represents MSB of data. When m = “1” (Figure) ❋ “S” represents MSB of data.	When m = “0” (Figure) ❋ “S” is a sign bit and represents MSB of data. When m = “1” (Figure) ❋ “S” is a sign bit and represents MSB of data.

Corrections and Supplementary Explanation for "7900 Series Software Manual" (REV.C) NO.5

<table>
<tr><th>Page</th><th>Error</th><th>Correction</th></tr>
<tr><td>P4-145
PEI
Description</td><td>Pushes the contents of the address specified by the sum of the contents of the DPRn and the offset value onto the stack in 16-bit length.
● This instruction is unaffected by flag m.</td><td>Pushes the contents of the address specified by the sum of the contents of the DPRn and the offset value onto the stack in 16-bit length.
• When the direct page register select bit = "0"
The contents of DPR0 are added to the value of operand dd (8 bits), and the result of addition is pushed onto the stack.
• When the direct page register select bit = "1"
In this case, the direct page register DPRn (n = 0 to 3) is specified by the high-order 2 bits of operand dd, and the contents of the specified DPRn are added to the low-order 6 bits of operand dd (8 bits). Then, the result of this addition is pushed onto the stack.
● This instruction is unaffected by flag m.</td></tr>
<tr><td>P4-145
PEI
Description example</td><td>; (S) ← (DPR0 + dd + 1)
; (S – 1) ← (DPR0 + dd)</td><td>; (S) ← (DPR0 + offset + 1)
; (S – 1) ← (DPR0 + offset)</td></tr>
<tr><td>☆
P4-161
PLP,
P4-167
PUL,
Description
Last line</td><td></td><td>● Be sure to keep flag I = "1" when executing this instruction. Also, be sure to use the SEI instruction when setting flag I to "1."</td></tr>
<tr><td>P4-167
PUL
Status flags</td><td>
<table>
<tr><th>Addressing mode</th><th>Cycles</th></tr>
<tr><td>STK</td><td>3 × i + 13</td></tr>
</table>
Note i : Number of registers to be restored.
</td><td>
<table>
<tr><th>Addressing mode</th><th>Cycles</th></tr>
<tr><td>STK</td><td>3 × i_1 + 13</td></tr>
</table>
Note: i_1 = number of registers to be restored.
</td></tr>
<tr><td>P4-171
ROL,
P4-174
ROR</td><td>
<table>
<tr><th>Syntax</th></tr>
<tr><td>ROL/ROR A
ROL/ROR B
ROL/ROR A, dd
ROL/ROR A, dd, X
ROL/ROR A, mmll
ROL/ROR A, mmll, X</td></tr>
</table>
</td><td>
<table>
<tr><th>Syntax</th></tr>
<tr><td>ROL/ROR A
ROL/ROR B
ROL/ROR dd
ROL/ROR dd, X
ROL/ROR mmll
ROL/ROR mmll, X</td></tr>
</table>
</td></tr>
<tr><td>P4-189
SEP
Operation</td><td>PS_L (bit n) ← 1 (n = 0 to 7. Multiple bits can be specified.)</td><td>PS_L (bit n) ← 1 (n = 0, 1, 3 to 7. Multiple bits can be specified.)</td></tr>
<tr><td>P4-189
SEP
Description</td><td>Sets the specified flags ···· be specified.
● This instruction is ····</td><td>Sets the specified flags ···· be specified.
Do not use this instruction to specify bit 2 of PS_L (flag I). (When setting flag I to "1," be sure to use the SEI instruction.)
● This instruction is ····</td></tr>
<tr><td>P4-189
SEP
Status flags</td><td>
<table>
<tr><td>IPL</td><td>N</td><td>V</td><td>m</td><td>x</td><td>D</td><td>I</td><td>Z</td><td>C</td></tr>
<tr><td>—</td><td>N</td><td>V</td><td>m</td><td>x</td><td>D</td><td>I</td><td>Z</td><td>C</td></tr>
</table>
</td><td>
<table>
<tr><td>IPL</td><td>N</td><td>V</td><td>m</td><td>x</td><td>D</td><td>I</td><td>Z</td><td>C</td></tr>
<tr><td>—</td><td>N</td><td>V</td><td>m</td><td>x</td><td>D</td><td>I (Note)</td><td>Z</td><td>C</td></tr>
</table>
Note: Do not use this instruction to specify flag I.
</td></tr>
</table>

Corrections and Supplementary Explanation for “7900 Series Software Manual” (REV.C) NO.6

Page	Error	Correction
P4-202 SUBMD **Description example**	SUBMB MEM32, #IMM32 ; MEM32 ← ····	SUBMD MEM32, #IMM32 ; MEM32 ← ····
P4-229 XAB **Description example**	CLM x XAB ; A ⇆ B SEM x XAB ; A_L ⇆ B_L	CLM XAB ; A ⇆ B SEM XAB ; A_L ⇆ B_L
P5-10 BSR Function	(S) ← PC PC ← PC + 2 + REL (–1024 to +1023)	(S) ← PC + 2 PC ← PC + 2 + REL (–1024 to +1023)
P5-14 CMPD Function	E – IMM32	E – M32
P5-16 EXTSD Function	E ← E_L(= A) (Extension sign) (Bit 15 of A = 0) b15 b0 b15 b0 00000000 | 0 E_H(B) E_L(A) (Bit 15 of A = 1) b15 b0 b15 b0 11111111 | 1 E_H(B) E_L(A)	E ← E_L(= A) (Extension sign) (Bit 15 of A = 0) b15 b0 b15 b0 0000_{16} | 0 E_H(B) E_L(A) (Bit 15 of A = 1) b15 b0 b15 b0 $FFFF_{16}$ | 1 E_H(B) E_L(A)
P5-16 EXTZD Function	E ← E_L(= A) (Extension zero) b15 b8 b7 b0 00000000 | E_H(B) E_H(A)	E ← E_L(= A) (Extension zero) b15 b0 b15 b0 0000_{16} | E_H(B) E_H(A)
☆ P5-28 PLP	PLP	PLP **(Note 22)**
☆ P5-30 PUL	PUL **(Note 18)**	PUL **(Notes 18 and 22)**

Corrections and Supplementary Explanation for “7900 Series Software Manual” (REV.C) NO.7

<table>
<tr><th>Page</th><th>Error</th><th>Correction</th></tr>
<tr><td>P5-31
PUL
STK</td><td>
<table>
<tr><td colspan="4">Addressing Modes</td></tr>
<tr><td></td><td colspan="3">STK</td></tr>
<tr><td></td><td>op</td><td>n</td><td>#</td></tr>
<tr><td></td><td>67</td><td>13
+
3 i</td><td>2</td></tr>
</table>
</td><td>
<table>
<tr><td colspan="4">Addressing Modes</td></tr>
<tr><td></td><td colspan="3">STK</td></tr>
<tr><td></td><td>op</td><td>n</td><td>#</td></tr>
<tr><td></td><td>67</td><td>13
+
3 i_1</td><td>2</td></tr>
</table>
</td></tr>
<tr><td>P5-36
SEP</td><td>
<table>
<tr><td>Symbol</td><td>Function</td></tr>
<tr><td>SEP</td><td>PS_L (bit n) ← 1
(n = 0 to 7. Multiple bits can be specified.)</td></tr>
</table>
</td><td>
<table>
<tr><td>Symbol</td><td>Function</td></tr>
<tr><td>SEP</td><td>PS_L (bit n) ← 1
(n = 0, 1, 3 to 7. Multiple bits can be specified.)</td></tr>
</table>
</td></tr>
<tr><td>P5-37
SEP</td><td>
<table>
<tr><td colspan="12">Processor Status register</td></tr>
<tr><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="3">IPL</td><td>N</td><td>V</td><td>m</td><td>x</td><td>D</td><td>I</td><td>Z</td><td>C</td></tr>
<tr><td>•</td><td>•</td><td>•</td><td colspan="8">Specified flag becomes “1.”</td></tr>
</table>
</td><td>
<table>
<tr><td colspan="12">Processor Status register</td></tr>
<tr><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="3">IPL</td><td>N</td><td>V</td><td>m</td><td>x</td><td>D</td><td>I</td><td>Z</td><td>C</td></tr>
<tr><td>•</td><td>•</td><td>•</td><td colspan="8">Specified flag becomes “1” (Note 21).</td></tr>
</table>
</td></tr>
<tr><td>P5-43
Note 18</td><td>Note 18. Letter “i” indicates the number of registers to be restored.</td><td>Note 18. Letter “i_1” indicates the number of registers to be restored.</td></tr>
<tr><td>☆
P5-43
Below
Note 20</td><td></td><td>Note 21. Do not use the SEP instruction to specify flag I. (When setting flag I to “1”, be sure to use the SEI instruction.)
Note 22. Be sure to keep flag I = “1” when executing the PLP or PUL instruction. Also, be sure to use the SEI instruction when setting flag I to “1.”</td></tr>
</table>