Microcontroller Technical Information

<table>
<tr><td rowspan="3">RA78K0
78K0 Assembler Package

Usage Restrictions</td><td>Document No.</td><td colspan="2">ZBG-CD-07-0064</td><td>1/2</td></tr>
<tr><td>Date issued</td><td colspan="3">September 18, 2007</td></tr>
<tr><td rowspan="2">Issued by</td><td colspan="3" rowspan="2">Development Tool Solution Group
Multipurpose Microcomputer Systems Division
Microcomputer Operations Unit
NEC Electronics Corporation</td></tr>
<tr><td rowspan="5">Related documents
RA78K0 Structured Assembly Language V3.80: U17197EJ1V0 (1st)
RA78K0 Ver. 3.80 Language: U17198EJ1V0 (1st)
RA78K0 Ver. 3.80 Operation: U17199EJ1V0 (1st)
PM+ Ver. 6.30: U18416EJ1V0 (1st)
78K0 Assembler Package RA78K0 Ver. 4.00 Operating Precautions: ZUD-CD-07-0102</td></tr>
<tr><td rowspan="4">Notification classification</td><td>√</td><td colspan="2">Usage restriction</td></tr>
<tr><td></td><td colspan="2">Upgrade</td></tr>
<tr><td></td><td colspan="2">Document modification</td></tr>
<tr><td></td><td colspan="2">Other notification</td></tr>
</table>

1. Affected product

   RA78K0 V4.00

2. New issue

   A new restriction (No. 43) has been added. See the attachment for details.

   No. 43 An E3410 error is output if there are multiple files whose primary name length is nine characters or longer and the first eight characters are the same.

3. Workaround

   The following workaround is available for this restriction. See the attachment for details.

   No. 43 Disable outputting of assembler module files, by using the compiler.

4. Modification schedule

   Product in which the restriction (No. 43) is corrected is scheduled for release as follows.

   RA78K0 V4.01: Planned for release in October 2007.

   * Note that this schedule is subject to change without notice. For the detailed release schedule of modified products, contact an NEC Electronics sales representative.

5. List of restrictions

   A list of restrictions in the RA78K0, including the revision history and detailed information, is described on the attachment.

6. Document revision history

78K0 Assembler Package RA78K0 - Usage Restrictions

Document Number	Date Issued	Description
SBG-TT-0004-E	October 12, 2001	Newly created.
SBG-DT-03-0308	December 17, 2003	Addition of restrictions (No. 28 to No. 34) Correction of erroneous description on corrected version (No. 8)
SBG-DT-04-0111	March 12, 2004	Addition of restrictions (No. 35 to No. 37)
ZBG-CD-07-0039	June 28, 2007	Addition of restrictions (No. 38 to No. 42)
ZBG-CD-07-0064	September 18, 2007	Addition of restriction (No. 43)

List of Restrictions in RA78K0

1. Product History

No.	Bugs and Changes/Additions to Specifications	Version					
		3.50	3.51	3.60	3.70	3.80	4.00
1	An error occurs if a control statement is crossed in a structured assembly language description.	×	×	×	×	×	×
2	An SFR read/write is not checked in the structured assembler.	○	○	○	○	○	○
3	An error is output for a correct absolute name for a register in the structured assembly language.	○	○	○	○	○	○
4	An error is not output for an incorrect absolute name for a register in the structured assembly language.	○	○	○	○	○	○
5	The assembler performs illegal processing if the label receiving the effect of optimization is described in the *saddr* part when an EQU definition is performed for a bit symbol with the value *saddr.bit*.	×	×	×	×	×	×
6	The number of characters that can be described on one line is 256 or less.	○	○	○	○	○	○
7	An include file name with a 2-byte character is not recognized. (Japanese version only)	○	○	○	○	○	○
8	The assembler does not operate correctly when "^Z" is not placed at the end of an include file. (Windows version only)	○	○	○	○	○	○
9	Concatenate (&) will not be linked if the macro quasi directive IRP is nested.	×	×	×	×	×	×
10	The library file included with the compiler cannot be manipulated by the librarian.	○	○	○	○	○	○
11	The dependence of include files when the structured assembler is started via the Project Manager	×	×	×	×	×	×
12	The assembler is always started when the structured assembler is started via the Project Manager.	○	○	○	○	○	○
13	The debug information output option cannot be set in the structured assembler even if “Debug” is selected from the [Option] menu in the Project Manager.	○	○	○	○	○	○
14	The include path specification option (-I) of the structured assembler cannot be made valid even if it is specified via the Project Manager.	○	○	○	○	○	○
15	The dependence of include files when the assembler is started via the Project Manager	×	×	×	×	×	×
16	The debug information output option cannot be set in the assembler even if “Debug” is selected from the [Option] menu in the Project Manager.	○	○	○	○	○	○
17	The parameter file specification option (-F) of the linker cannot be specified via the Project Manager.	○	○	○	○	○	○
18	The help specification option (--) of the assembler cannot be specified via the Project Manager.	○	○	○	○	○	○

×: Applicable, ○: Not applicable, –: Not relevant

No.	Bugs and Changes/Additions to Specifications	Version					
		3.50	3.51	3.60	3.70	3.80	4.00
19	The 128th and subsequent characters are ignored in a command line.	○	○	○	○	○	○
20	Only the source files in the project directory can be specified when using the Project Manager.	○	○	○	○	○	○
21	A device other than the μPD7800xx cannot be specified as the target device when the assembler is installed using *dosinst.bat*.	–	–	–	–	–	–
22	The commands of the assembler and other tools cannot be activated when only the assembler package is installed using setup31.exe.	–	–	–	–	–	–
23	File name in the object created from the structured assembly language source is invalid.	○	○	○	○	○	○
24	An illegal processing error is output by Windows when the object module file no-output option (-NO) is specified in the assembler.	○	○	○	○	○	○
25	An error occurs when the output file name specification option (-O) is specified in the structured assembler or assembler via the Project Manager.	○	○	○	○	○	○
26	An error occurs when a project file created by ID78K0-NS (V2.xx), ID78K0 (V2.xx) or SM78K0 (V2.xx), which is started solely, is opened via the Project Manager.	×	×	×	×	×	×
27	An error occurs in the linker or object converter if the assembler code EXTERN appears first when C source and assembler source exist together.	○	○	○	○	○	○
28	An error will be output from the command prompt if there is no device file when the structured assembler is started from the command prompt.	×	○	○	○	○	○
29	A multi-byte character comment in a macro definition in an assembler source is not displayed in the macro expansion in the assemble list.	×	○	○	○	○	○
30	Even if a numeric value is specified for a symbol using the -D option, the macro can use it only as a truth value.	×	○	○	○	○	○
31	An invalid code may be output as a result of calculating the number of codes after the BR directive in the last segment of the assembler source is optimized.	×	○	○	○	○	○
32	A segment for which “MERGE AT” is specified by a directive is not prioritized for allocation.	×	○	○	○	○	○
33	The debugger or simulator outputs an error if the ancillary symbol table output from the compiler exists in the PUBLIC/EXTRN symbol.	×	○	○	○	○	○
34	When the segment name consists of eight characters, the linker may output an invalid segment name in the error message.	×	○	○	○	○	○
35	Allocation of segments whose size is 0 may be invalid.	×	×	×	○	○	○
36	Name definition in the assembler results in an A106 error in the object converter.	×	×	×	○	○	○

×: Applicable, ○: Not applicable, –: Not relevant

No.	Bugs and Changes/Additions to Specifications	Version					
		3.50	3.51	3.60	3.70	3.80	4.00
37	An A402 error is output in the linker if the source file name consists of 15 or more characters.	×	×	×	×	×	○
38	An F4106 error is output if –Dsymbol option is specified in the RA78K0.	×	×	×	×	×	○
39	An F6105 error is output in the list converter if the definition of a function name that has been referenced externally from a C source is a PUBLIC symbol in a structured assembler source or an assembler source	×	×	×	×	×	○
40	An F6106 error is output in the list converter if coding of the segment name is omitted in the ORG quasi directive.	×	×	×	×	×	○
41	Only the first one item in an absolute assembler list is output if multiple C sources are registered and the assembler source output option is specified.	×	×	×	×	×	○
42	The size specified in the Object Complement field in the Object Converter Options dialog box in PM+ becomes invalid.	○	○	○	○	×	○
43	An E3410 error is output if there are multiple files whose primary name length is nine characters or longer and the first eight characters are the same.	–	–	–	–	–	×

×: Applicable, ○: Not applicable, –: Not relevant

2. Restriction Details

No. 1 An error occurs if a control statement is crossed in a structured assembly language description.

[Description]

If a control statement is divided or crossed by code from *#ifdef* to *#endif*, an error occurs when *#ifdef* is true.

Example:

```
switch(mode)
#ifdef   stsw        ← Range from #ifdef to #else or #endif
  case 1:            ← Range from case to next case, default or ends
         break
#endif
  default:
         break
ends
```

[Workaround]

There is no error in the case of nesting. The source should be rewritten so that the ranges of the control statements do not cross.

Example:

```
#ifdef  stsw         ← Range from #ifdef to #else or #endif
  switch(mode)
    case 1:          ← Range from case to next case, default or ends
       break
    default:
       break
  ends
#else
  switch(mode)
    default:
       break
  ends
#endif
```

[Correction]

Regard this issue as a usage restriction.

No. 2 An SFR read/write is not checked in the structured assembler.

[Description]

When an SFR is specified for the register, an error may occur at assembly because an SFR read/write is not performed.

[Workaround]

Confirm the SFR read/write before use.

[Correction]

This issue has been corrected in V3.30.

No. 3 An error is output for a correct absolute name for a register in the structured assembly language.

[Description]

When a correct absolute name is described for a register, an error is output even though it should be processed normally.

```
AX = PR1  PR1 = AX    AX <-> PR1
AX = PR2  PR2 = AX    AX <-> PR2
AX = PR3  PR3 = AX    AX <-> PR3
```

[Workaround]

Describe the name with the function name.

```
AX = BC   BC = AX     AX <-> BC
AX = DE   DE = AX     AX <-> DE
AX = HL   HL = AX     AX <-> HL
```

[Correction]

This issue has been corrected in V3.30.

No. 4 An error is not output for an incorrect absolute name for a register in the structured assembly language.

[Description]

When an incorrect absolute name is described for a register, the assembler terminates normally even though it should output an error.

```
A = R1
R1 = A
A <-> R1
```

[Workaround]

There is no workaround.

[Correction]

This issue has been corrected in V3.30.

No. 5 The assembler performs illegal processing if the label receiving the effect of optimization is described in the *saddr* part when an EQU definition is performed for a bit symbol with the value *saddr.bit*.

[Description]

The assembler performs illegal processing if the label receiving the effect of optimization is described in the *saddr* part when an EQU definition is performed for a bit symbol with the value *saddr.bit*.

[Phenomenon]

The assembler performs illegal processing in the following cases.

- When *saddr.bit* is 0FD20H, path 1 of a label is outside the area, and path 2 is inside the area, an error is output in path 1 for the EQU definition line, but not output in path 2. At this time, the object is created but it is incorrect.

- When *saddr* is 0FF1FH, path 1 of a label is inside the area, and path 2 is outside the area, no error is output in path 1 for the EQU definition line, while an error is output in path 2. The assembly error “F410 Phase error” will be output for a label that is defined after this EQU symbol has been referenced.

  When this label is referenced, the object becomes incorrect.

[Workaround]

There is no workaround.

[Correction]

Regard this issue as a usage restriction.

No. 6 The number of characters that can be described on one line is 256 or less.

[Description]

If the number of characters on one line exceeds 256, the error “W701 Too long source line” error is and the characters exceeding 256 are assembled as the next line.

[Workaround]

Keep the number of characters on one line to 256 or less (including return and line feed characters).

[Correction]

This issue has been corrected in V3.50.

No. 7 An include file name with a 2-byte character is not recognized. (Japanese version only)

[Description]

If an include file name includes a 2-byte character, the file may not be recognized correctly.

[Workaround]

Do not use 2-byte characters for include file names.

[Correction]

This issue has been corrected in V3.50.

No. 8 The assembler does not operate correctly when "^Z" is not placed at the end of an include file. (Windows version only)

[Description]

The absence of "^Z" at the end of an include file may cause an error or the output of invalid debug information.

[Workaround]

Enter a line containing only ^Z at the end of the include file.

[Correction]

This issue has been corrected in V3.50.

No. 9 Concatenate (&) will not be linked if the macro quasi directive IRP is nested.

[Description]

Nesting IRP that includes the character string concatenation symbol "&” will render the macro expansion results invalid since the parameter is not converted.

Example:

```
IRP      ZZZ, <1,2,3>
         IRP   XXX, <4,5,6>
               LABEL&ZZZ&XXX:      ; Incorrect
         ENDM
ENDM
```

[Workaround]

Do not use “&” when an IRP is nested.

[Correction]

Regard this issue as a usage restriction.

No. 10 The library file included with the compiler cannot be manipulated by the librarian.

[Description]

When the library file included with the compiler is manipulated by the librarian, the error message “F401 Illegal library file - different target chip” will be output and manipulation will be stopped.

[Workaround]

There is no workaround. Manipulation is possible if the device file for the μPD78014 is installed.

[Correction]

This issue has been corrected in V3.50.

No. 11 The dependence of include files when the structured assembler is started via the Project Manager

[Description]

While checking the dependence of include files described in structured assembler language when creating a make file in the Project Manager, deletion of comments and character string processing are performed but conditions such as *#ifdef* are ignored.

[Phenomenon]

Lines *#ifdef*, *#else* and *#endif* are ignored in the following example. As a result, the existence of the file name specified by *#include* is checked regardless of whether the file is referenced. An error will occur if the file specified by *#include* does not exist when executing a build.

Example:

```
#ifdef SYM
#include "func1.inc"
#else
#include "func2.inc"
#endif
```

[Workaround]

There is no workaround.

[Correction]

Regard this issue as a usage restriction.

No. 12 The assembler is always started when the structured assembler is started via the Project Manager.

[Description]

Starting the structured assembler solely via the Project Manager is not possible. The assembler is always started simultaneously.

[Workaround]

There is no workaround.

[Correction]

This issue has been corrected in V3.50.

No. 13 The debug information output option cannot be set in the structured assembler even if “Debug” is selected from the [Option] menu in the Project Manager.

[Description]

Specification of the debug information output option set in the structured assembler is not synchronized with the selection of “Debug” from the [Option] menu in the Project Manager.

[Workaround]

Specify the -GS option in the [Structured Assembler Options] dialog box in the Project Manager to enable the debug information output option.

[Correction]

This issue has been corrected in V3.30.

No. 14 The include path specification option (-I) of the structured assembler cannot be made valid even if it is specified via the Project Manager.

[Description]

When the include path specification option -I of the structured assembler is specified via the Project Manager, it may be interpreted as an assembler option.

[Workaround]

Place into the current directory the include file written in the structured assembly language, or write the include file with the directory name when including it into the source file.

[Correction]

This issue has been corrected in V3.30.

No. 15 The dependence of include files when the assembler is started via the Project Manager

[Description]

While checking the dependence of include files described in assembler language when creating a make file in the Project Manager, deletion of comments and character string processing are performed but conditions such as *$if* and *$_if* are ignored.

[Phenomenon]

Lines of *#if*, *#else* and *#endif* are ignored in the following example. As a result, the existence of the file name specified by *#include* is checked regardless of whether the file is referenced or not. An error will occur if the file specified by *#include* does not exist when executing a build.

Example:

```
#if (SYM)
#include(func1.h)
#else
#include(func2.h)
#endif
```

[Workaround]

There is no workaround.

[Correction]

Regard this issue as a usage restriction.

No. 16 The debug information output option cannot be set in the assembler even if “Debug” is selected from the [Option] menu in the Project Manager.

[Description]

Specification of the debug information output option set in the assembler is not synchronized with the selection of “Debug” from the [Option] menu in the Project Manager.

[Workaround]

Specify the -G and -GA options in the [Assembler Options] dialog box in the Project Manager to enable the debug information output option.

[Correction]

This issue has been corrected in V3.30.

No. 17 The parameter file specification option (-F) of the linker cannot be specified via the Project Manager.

[Description]

Because the Project Manager automatically creates the -F option when a make file is automatically created, if the -F option is specified in the assembler, the linker outputs the error message “A107 Too many parameters ‘-fxxxx.xxx’”.

[Workaround]

There is no workaround. Do not specify the -F option.

[Correction]

This issue has been corrected in V3.50.

No. 18 The help specification option (--) of the assembler cannot be specified via the Project Manager.

[Description]

When the -- option is specified, the message “Press RETURN to continue …” will be displayed, but the key operation is not recognized in the PRJTMAKE window. As a result, the program cannot return from the help display.

[Workaround]

Do not specify the -- option. Use the Help button in the Option Setting dialog box when referencing help.

[Correction]

This issue has been corrected in V3.30.

No. 19 The 128th and subsequent characters are ignored in a command line.

[Description]

Commands activated via the Project Manager, such as assembler commands, are commands activated in the DOS prompt. When a command is activated via the Project Manager, the maximum number of characters that can be written on one line is therefore 127, in the same way as commands activated from the command line. The 128th and subsequent characters will be ignored.

[Workaround]

Describe the option with 127 characters or less, making allowances for commands and options that are automatically output.

[Correction]

This issue has been corrected in V3.30.

No. 20 Only the source files in the project directory can be specified when using the Project Manager.

[Description]

Only the source files in the project directory can be registered.

[Workaround]

Place the source files in the project directory.

[Correction]

This issue has been corrected in V3.30.

No. 21 A device other than the μPD7800xx cannot be specified as the target device when the assembler is installed using *dosinst.bat*.

[Description]

When the assembler is installed using *dosinst.bat*, an error occurs if a device other than the μPD7800xx is specified as the target device because *ra78k0.is2* is not installed.

[Workaround]

Implement either of the following workarounds.

- Install the assembler using the installer of Windows

  Activate Windows installer *setup31.exe* or *setup95.exe* and install the assembler.
- Install the assembler by executing the EXPAND command in the DOS prompt

  Execute the EXPAND command as shown below after installing the assembler using *dosinst.bat*.

  ```
  a:\expand ra78k0.i2_ c:\nectools\bin\ra78k0.is2 >nul
  ```

  (a: is the floppy disk drive and c: is the drive in which the assembler is installed.)

[Correction]

This issue has been corrected in V3.30.

No. 22 The commands of the assembler and other tools cannot be activated when only the assembler package is installed using *setup31.exe*.

[Description]

When only the assembler package is installed using *setup31.exe*, the commands of the assembler and other tools cannot be activated on Windows 3.1 because *windpmi.386* is not installed.

[Workaround]

Install the Project Manager as well as the assembler package.

[Correction]

This issue has been corrected in V3.30.

No. 23 File name in the object created from the structured assembly language source is invalid.

[Description]

The file name used for debug information, which is generated by the structured assembler and output to the assembler source, includes a directory. The assembler, however, does not assume that the file names include directory names. As a result, the assembler creates an invalid file name and outputs it to the object.

[Workaround]

There is no workaround.

[Correction]

This issue has been corrected in V3.30.

No. 24 An illegal processing error is output by Windows when the object module file no-output option (-NO) is specified in the assembler.

[Description]

The error "This program has performed an illegal operation and will be shut down." is output by Windows when the object module file no-output option (-NO) is specified in the assembler during debug information generation processing, and Windows is terminated abnormally.

[Workaround]

There is no workaround. Do not specify the -NO option.

[Correction]

This issue has been corrected in V3.50.

No. 25 An error occurs when the output file name specification option (-O) is specified in the structured assembler or assembler via the Project Manager.

[Description]

When a file name is changed by specifying the output file name specification option (-O), an error occurs because the change of the file name is not reflected in the command that is activated using the changed file name as an input file.

[Workaround]

There is no workaround. Do not specify the -O option.

[Correction]

This issue has been corrected in V3.50.

No. 26 An error occurs when a project file created by ID78K0-NS (V2.xx), ID78K0 (V2.xx) or SM78K0 (V2.xx), which is started solely, is opened via the Project Manager.

[Description]

The error message “F150: Failed to read the project file.” is output if an attempt is made to read a project file created by ID78K0-NS (V2.xx), ID78K0 (V2.xx) or SM78K0 (V2.xx), which is started solely (a file that is created by tools other than the Project Manager).

[Workaround]

Create a new project file in the Project Manager that has the same name as the project file created by ID78K0-NS (V2.xx), ID78K0 (V2.xx) or SM78K0 (V2.xx), which is started solely. After that, select [Select Debugger] from the [Option] menu in the Project Manager and specify the debugger (ID or SM) used.

[Correction]

Regard this issue as a usage restriction.

No. 27 An error occurs in the linker or object converter if the assembler code EXTERN appears first when C source and assembler source exist together.

[Description]

An error may occur in the linker (error A402) or object converter (error A106) if the assembler code EXTERN appears first while a C source and an assembler source exist together, and multiple symbol definitions and the modules that reference the definitions exist.

[Workaround]

Specify to the linker and link modules generated from the C source before modules generated from the assembler source. In addition, register the C source in the Project Manager first. When using a startup routine, link the startup routine first.

[Correction]

This issue has been corrected in V3.50.

No. 28 An error will be output from the command prompt if there is no device file when the structured assembler is started from the command prompt.

[Description]

An error will be output from the command prompt or no response will be made from the program if there is no device file when the structured assembler is started from the command prompt.

[Workaround]

Make sure that the device file is installed. In addition, if the structured assembler or device file is not installed in the default folder, specify the path for the device file using the -Y option.

[Correction]

This issue has been corrected in V3.51.

No. 29 A multi-byte character comment in a macro definition in an assembler source is not displayed in the macro expansion in the assemble list.

[Description]

If a comment including a multi-byte character is written in a macro definition, only the multi-byte character comment cannot be displayed in macro expansion in the assemble list file created after assembly.

Example:

sample.asm

```
NAMEsample
MAC1MACRO
        NOP     ; comment1 Multi-byte character  Single-byte Japanese syllabary
                  ENDM

 CSEG
 SAMPLE1:
                  MAC1

                  END
```

sample.prn

```
          Assemble list

ALNO  STNO  ADRS OBJECT M I  SOURCE STATEMENT

1     1                                NAME    SAMPLE
2     2                  M   MAC1      MACRO
3     3                  M             NOP     ; comment1 Multi-byte character
                                               Single-byte Japanese syllabary
4     4                  M             ENDM
5     5
6     6     ----                       CSEG
7     7     0000             SAMPLE1:
8     8                                MAC1
      9     0000   00    #1            NOP     ; comment1  Single-byte
                                               Japanese syllabary
9     10
10    11                               END
```

[Workaround]

There is no workaround.

[Correction]

This issue has been corrected in V3.51.

No. 30 Even if a numeric value is specified for a symbol using the -D option, the macro can use it only as a truth value.

[Description]

A symbol value specified using the -D option in the assembler source, instead of describing *$set*, *$reset* or EQU, does not become valid.

Example 1: (2) may be valid even if “other than *-Dsyma=0*” is specified

Normally, error A109 should be output if *$set(syma)* or *$reset(syma)* is written when -D is specified, but coding of *$set(syma)* or *$reset(syma)* becomes valid and the -D is ignored.

```
$if (syma)
  ; (1) When “-Dsyma” or “other than -Dsyma=0” is specified
$else
  ; (2) When “-Dsyma=0” is specified
$endif
```

Example 2: The -D option is ignored and F303 or F407 will be output.

Normally, error A109 should be output if an EQU is written when -D is specified, but EQU becomes valid because -D is ignored.

```
$_if ( symb = 0FFH )
  ; (3) When “-Dsymb=0FFH” is specified
$_elseif ( symb = 01H )
  ; (4) When “-Dsymb” or “-Dsymb=1” is specified
$else
  ; (5) When a value other than (3) or (4) is specified for symb
$endif
```

[Workaround]

Write *$set(syma)*, *$reset(syma)* or EQU without specifying the -D option.

[Correction]

This issue has been corrected in V3.51.

No. 31 An invalid code may be output as a result of calculating the number of codes after the BR directive in the last segment of the assembler source is optimized.

[Description]

An invalid code may be output as a result of calculating the number of codes after the BR directive in the last segment of the assembler source is optimized.

Though information in the assemble list file (*.prn) and link list file (*.map) is correct, information in the relocatable object file (*.rel) is invalid. Consequently, information in the load module file (*.lmf) and HEX-format object module file (*.hex) also becomes invalid.

[Workaround]

Implement either of the following workarounds.

- Use the BR instruction instead of the BR quasi directive because its instruction length is clear.
- Insert NOPs for the amount of insufficient code at the end of the segment in which codes are lacking.

[Correction]

This issue has been corrected in V3.51.

No. 32 A segment for which “MERGE AT” is specified by a directive is not prioritized for allocation.

[Description]

A segment for which “MERGE AT” is specified by a directive must be prioritized for allocation to the address specified by AT, but is not. Consequently, a segment without AT specification is allocated first, which may cause the error F206 or F304.

[Workaround]

Move the already allocated segment to the area where the segment for which “MERGE AT” is specified is to be allocated.

[Correction]

This issue has been corrected in V3.51.

No. 33 The debugger or simulator outputs an error if the ancillary symbol table name output from the compiler exists in the operand field of the PUBLIC/EXTRN quasi directive.

[Description]

Because of a linker restriction, the debugger or simulator outputs the error message “b019(A): Cannot seek file” if there is an ancillary symbol table name output from the compiler exists in the operand field of the PUBLIC/EXTRN quasi directive.

[Workaround]

There is no workaround.

[Correction]

This issue has been corrected in V3.51.

No. 34 When the segment name consists of eight characters, the linker may output an invalid segment name in the error message.

[Description]

When the segment name consists of eight characters, the linker may output an invalid segment name in the error message.

[Workaround]

There is no workaround.

[Correction]

This issue has been corrected in V3.51.

No. 35 Allocation of segments whose size is 0 may be invalid.

[Description]

Allocation of segments may be invalid when the following three conditions are satisfied.

(1) A segment whose size is 0 exists.

(2) The segment of (1) is not addressed.

(3) A segment that is addressed to a segment other than (1) exists.

A segment whose address is not output to the link list in the ascending address order is an incorrectly-allocated segment.

[Workaround]

Address the incorrectly allocated segment using the directive file.

[Correction]

This issue has been corrected in V3.70.

No. 36 Name definition in the assembler results in an A106 error in the object converter.

[Description]

Symbol information may be invalid when all the conditions shown below are satisfied, which results in an A106 error in the object converter. In addition, the error message “Ab019 : Reading of file went wrong.” will be output when the load module file is downloaded by the debugger.

(1) The name is defined in the structured assembler source or assembler source

(2) The name of (1) is externally referenced as a function name by the C source.

(3) The file of (1) is linked after the C source of (2).

[Workaround]

Link the C source to a location below where the structured assembler source or assembler source is linked.

[Correction]

This issue has been corrected in V3.70.

No. 37 An A402 error is output in the linker if the source file name consists of 15 or more characters.

[Description]

An A402 error is output in the linker when the following two conditions are satisfied.

(1) The source file name consists of 15 or more characters.

(2) Debug information is not output by the assembler.

The output of debug information is not specified in any of the following cases.

- “Release Build” is selected in PM or PM plus.
- The “-nga -ng” option is specified in the assembler.

[Workaround]

Implement either of the following workarounds.

(1) Change the name of the source file so that it includes less than 15 characters.

(2) Select “Debug Build” in PM or PM plus, or specify the “-ga -g” option (specified by default) in the assembler.

Debug information output by the assembler does not affect the generated HEX file.

[Correction]

This issue will be corrected in V4.00.

No. 38 An F4106 error is output if –Dsymbol option is specified in the RA78K0.

[Description]

If a *symbol* in the –Dsymbol option is specified in the RA78K0 for an assembler source file in which debug information is appended via the CC78K0, invalid symbol information (symbol name that cannot be referenced) is output and the OC78K0 outputs an F4106 error for the symbol table.

[Workaround]

Do not specify the “symbol definition” option (assembler option) when specifying the assembler source module output option in the CC78K0. Specify the “symbol definition” option for an assembler source file by using the [Special Compiler Options...] menu.

[Correction]

This issue will be corrected in V4.00.

No. 39 An F6105 error is output in the list converter if the definition of a function name that has been referenced externally from a C source is a PUBLIC symbol in a structured assembler source or an assembler source.

[Description]

An F6105 error is output in the list converter if the definition of a function name that has been referenced externally from a C source is a PUBLIC symbol in a structured assembler source or an assembler source.

[Workaround]

Do not start the list converter.

[Correction]

This issue will be corrected in V4.00.

No. 40 An F6106 error is output in the list converter if coding of the segment name is omitted in the ORG quasi directive.

[Description]

An F6106 error is output in the list converter if coding of the segment name is omitted in the ORG quasi directive.

Example:

```
-------------------- Assembler source -----------------------------
                 ORG     1000H
                 MOV             A, SADR2
                 MOVW            AX, SADR3
```

[Workaround]

Describe the segment name for the ORG quasi directive.

Example:

```
-------------------- Assembler source -----------------------------
         MAIN1   ORG     1000H
                 MOV             A, SADR2
                 MOVW            AX, SADR3
```

[Correction]

This issue will be corrected in V4.00.

No. 41 Only the first one item in an absolute assembler list is output if multiple C sources are registered and the assembler source output option is specified.

[Description]

Only the first one item in an absolute assembler list is output if multiple C sources are registered and the assembler source output option is specified.

[Workaround]

Create an absolute assembler list by using the LCNV78K0 activated via the command prompt.

[Correction]

This issue will be corrected in V4.00.

No. 42 The size specified in the Object Complement field in the Object Converter Options dialog box in PM+ becomes invalid.

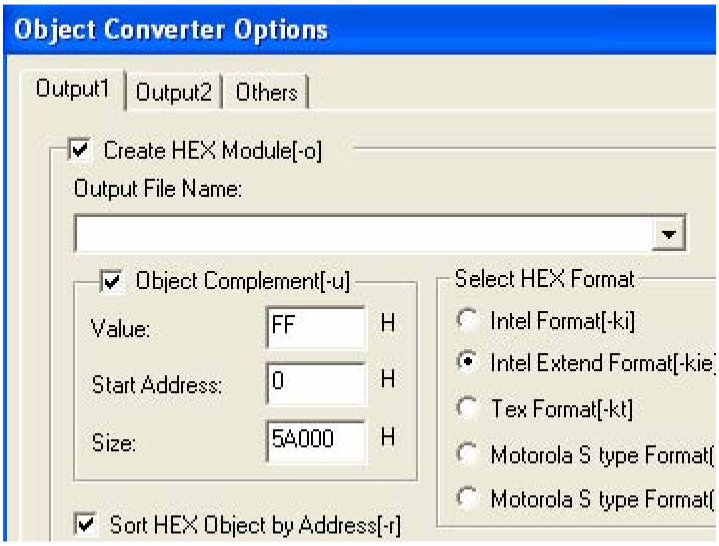
[Description]

The size specified in the Object Complement field in the Object Converter Options dialog box in PM+ becomes invalid under the following conditions.

(1) The target device has a memory bank.

(2) Either the start address or the address area size is 58000H or more.

Example:



☑ Create HEX Module[-o]

Output File Name:

☑ Object Complement[-u]

Value: FF H

Start Address: 0 H

Size: 5A000 H

Select HEX Format

○ Intel Format[-ki]

● Intel Extend Format[-kie

○ Tex Format[-kt]

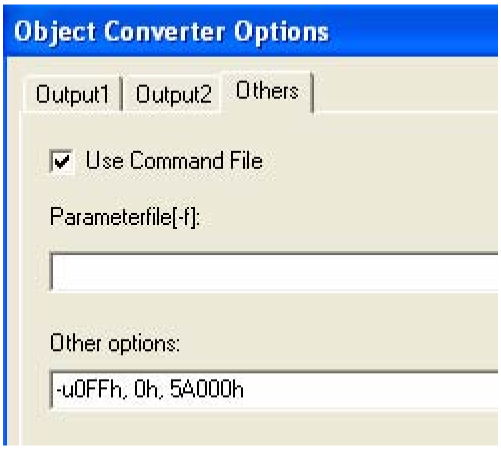
○ Motorola S type Format(

○ Motorola S type Format(

☑ Sort HEX Object by Address[-r]

[Workaround]

Set "-u0FFh, 0h, 5A000h" in the Other options text box on the [Others] tab in the Object Converter Options dialog box.



☑ Use Command File

Parameterfile[-f]:

Other options:

-u0FFh, 0h, 5A000h

[Correction]

This issue will be corrected in V4.00.

No. 43 An E3410 error is output if there are multiple files whose primary name length is nine characters or longer and the first eight characters are the same.

[Description]

An E3410 error is output if there are multiple files whose primary name length is nine characters or longer and the first eight characters are the same.

[Workaround]

If the file is a C source file, disable outputting of assembler module files by specifying a C compiler option.

[Correction]

This issue will be corrected in V4.01.

3. Cautions

None.