CUSTOMER NOTIFICATION

SUD-DT-03-0193-E

May 15, 2003

Koji Nishibayashi, Senior System Integrator
Microcomputer Group
2nd Solutions Division
Solutions Operations Unit
NEC Electronics Corporation

CP (K), O

Library Source File for CC78K0
CC78K0-L V3.50
Operating Precautions

PC-9800 series (Windows) Based
IBM PC/AT compatible machines (Windows) Based
HP9000 series 700 (HP-UX) Based
SPARCstation Family (SunOS, Solaris) Based

Be sure to read this document before using the product

1. PRODUCT OVERVIEW

The CC78K0-L is an assembler source file for the runtime library and standard library included in the C compiler CC78K0.
The batch files for updating the source are also included.
Use the CC78K0-L in combination with C compiler CC78K0 V3.50 or later and assembler package RA78K0 V3.60 or later.

1.1 Host Machine and Supply Medium

Host Machine	OS	Version	Supply Medium
PC-9800 series	Japanese Windows	98/Me/2000/NT4.0 Note	CD-ROM
IBM PC/AT compatible machines	Japanese Windows	98/Me/2000/NT4.0 Note	
	English Windows	98/Me/2000/NT4.0 Note	
HP9000 Series 700	HP-UX	Rel. 10.20 or later	
SPARCstation Family	SunOS	Rel. 4.1.4 or later	
	Solaris	Rel. 2.5.1 or later	

Note Japanese Windows and English Windows are independent products.

To assemble this library source file, use a batch file and run the program from the DOS prompt (Windows 98 and Windows Me) or command prompt (Windows NT, Windows 2000 and Windows XP) or on an EWS (HP-UX, SunOS, and Solaris).

1.2 Handling of Library Source File

The CC78K0-L is an assembler source file for the runtime library and standard library included in the C compiler CC78K0.
The purpose of the CC78K0-L is to make the contents of the runtime and standard libraries available **as references** for users of the CC78K0; it is not intended for modification by users.
Therefore, modified and updated libraries will not be supported.
If it is necessary to update the startup routine or some of the library functions, be sure to use the batch file described in **3.2 Types of Batch File**. Also, use **mklib.bat** and **mkflib.bat** for newly creating a library.

2. INSTALLATION AND FILE CONFIGURATION

2.1 Installing in PC-9800 Series or IBM PC/AT Compatible Machines

Execute the self-extracting file. Change the current directory to the one in which the CC78K0-L is to be installed, as necessary.

2.2 Installing in HP9000 Series 700 or SPARCstation Family

Mount the CD-ROM and execute the cp command to copy the files from the CD-ROM.
See the EWS user's manual for how to mount the CD-ROM because it varies depending on the environment.

2.3 File Configuration After Installation

The file configuration after installation is as follows.

Directory	File	Description
bat	mkstup.bat[Note]	Batch file for assembling startup routine
	reprom.bat[Note]	Batch file for updating ROMization routine
	*.bat[Note]	Batch file for updating library I/O library when processing is changed
	*.sub	Sub file referenced at batch file execution
fsrc	*.asm	Source file for library supporting floating-point operations
inc	*.inc	Assembler header file
src	*.asm	Source file such as startup routine, ROMization routine, library, etc.

Note This is a shell file (*.sh) in the HP9000 Series 700 and SPARCstation Family.

The C compiler CC78K0 includes source files such as the startup routine and ROMization routine, include file, batch file for assembly, etc.
When the C compiler CC78K0 is installed, some of the CC78K0-L components are installed in C:\NECTools32\SRC\CC78K0 (PC-9800 Series and IBM PC/AT compatible machines) or /nectools/src/cc78k0 (HP9000 Series 700 and SPARCstation Family).
Therefore, by specifying the same directory as that in which this library source file is to be installed, the contents are installed by overwriting the existing contents in the directory.

3. BATCH FILES

The CC78K0-L includes the batch files used for updating the startup routine supplied with the C compiler CC78K0 and some standard library functions.
In the HP9000 Series 700 and SPARCstation Family, shell files (*.sh) are supplied instead of batch files. The file names are the batch file names with the extension .bat replaced by .sh.

3.1 How to Use Batch Files

Before executing a batch file

The environment in which assembler package RA78K0 V3.50 can operate is necessary for execution of a batch file. Specify the directory in which the RA78K0 execution format file is stored using the environment variable PATH before executing the batch file.

Execution of batch file

Change the current directory to the BAT directory and execute each batch file for assembly under the BAT directory. At this time, parameters must be specified.

Refer to **3.2 Types of Batch Files** for details of batch files, and **3.3 Parameters for Batch Files** for details of parameters.

Some batch files execute assembly assuming that the library exists in the LIB directory at the same level as the BAT directory. Refer to **Note 2** under **3.2 Types of Batch Files** for details of the applicable batch file.

3.2 Types of Batch Files

The list of batch files is shown below.

No.	Batch File	Description
1	mkstup.bat	Updates the startup routine (cstart*.asm). Perform assembly using this batch file when the startup routine is updated.
2	reprom.bat	Updates the ROMization terminal routine (rom.asm). Perform assembly using this batch file when rom.asm is updated.
3	repgetc.bat	Updates the *getchar* function. The SFR P0 is set as an input port by default. To change the port, change the EQU definition value of PORT in getchar.asm and update the library using this batch file.
4	repputc.bat	Updates the *putchar* function. The SFR P0 is set as an output port by default. To change the port, change the EQU definition value of PORT in putchar.asm and update the library using this batch file.
5	repputcs.bat	Updates the *putchar* function so as to support the SM78K0. To confirm the output of the *putchar* function in the SM78K0, update the library using this batch file.
6	repselo.bat	Saves/restores the compiler reserved area (_@KREGxx) during *setjmp/longjmp* function save/restore processing. (Under the default setting, save/restore processing is not performed.) Update the library using this batch file when specifying the -QR and -QR2 options.
7	repselon.bat	Disables save/restore processing of the compiler reserved area (_@KREGxx) during *setjmp/longjmp* function save/restore processing. (Under the default setting, save/restore processing is not performed.) Update the library using this batch file when not specifying the -QR and -QR2 options.
8	repbank.bat	Updates the bank function call routine (bankcall.asm). The SFR P0 is set as a bank switching control port by default. To change the port, change the EQU definition value of PORT in bankcall.asm and update the library using this batch file.
9	repvect.bat	Updates the setting processing (vect*.asm) of the address value of the branch table for the interrupt function allocated in the flash memory area when the flash memory self-writing function of the device is used. The #pragma ext_table 0x4000 declaration is set by default. When the address value of #pragma ext_table is changed to another one, change the EQU value of ITBLTOP in INC\vect.inc to the value specified by #pragma ext_table and update the library using this batch file.
10	mklib.bat	Updates the standard library and runtime library.
11	mkflib.bat	Updates the floating-point library.

Notes 1. When No.1, 10, 11 are executed, a LIB directory is created under the same level as the BAT directory and files after assembly will be created under that directory.

2. Assembly from No.2 to 9 is executed assuming that the library exists in the LIB directory at the same level as the BAT directory. If the library does not exist, therefore, an error will be output. In such a case, use the library under the LIB78K0 directory in the C compiler CC78K0, or create the library using **mklib.bat** and **mkflib.bat** included in the CC78K0-L. (The libraries included in the C compiler CC78K0 support all devices in the 78K0 Series.)

3.3 Parameters for Batch Files

The list of the batch file parameters is shown below. Specification of parameters is not case-sensitive.

No.	Classification	Parameter	Description
1	Chip type (chiptype)	(E.g.) 0138	Target device type (e.g. uPD780138)
2	Memory model (model)	NORMAL	Normal model
		STATIC	Static model[Note 1, 2]
3	Area (area)	BOOT	Ordinary area and boot area[Note 3]
		FLASH	Flash area[Note 3]
4	Use of multiplication/ division instruction (muldiv)	USE	Used (specify when using other than the μPD78002, 78002Y Subseries)
		NOUSE	Not used (specify when using the μPD78002, 78002Y Subseries)
5	Function interface (func_interface)	NEW	Function interface in CC78K0 V3.00 or later
		OLD	Function interface[Note 1] in CC78K0 V2.11
6	Pascal function interface (pascal)	NOTPASCAL	Ordinary function interface (when pascal function is not used)
		PASCAL	Pascal function interface (when pascal function is used)[Note 2]

Notes 1. Because the old function interface is not provided in the static model, do not specify STATIC and OLD at the same time.

2. Because the static model does not support pascal functions, do not specify STATIC when PASCAL is specified.

3. When the flash memory self-writing function is used.

3.4 Example of Batch File Usage

The parameter format of each batch file and examples of how to use each batch file in PC-9800 and IBM PC/AT compatible machines are shown below.
Execute “/bin/sh *.sh parameter” when using the EWS version HP9000 Series 700 or SPARCstation Family.

(1) For updating startup routine

Format:	mkstup	Chip type
Example:	mkstup	0138
	(EWS: /bin/sh mkstup.sh 0138)	

(2) For updating ROMization routine

Format:	reprom	Chip type	Use of multiplication/division instruction
Example:	reprom	0138	use

(3) For updating **getchar** function

Format:	repgetc	Chip type	Use of multiplication/division instruction
Example:	repgetc	0138	use

(4) For updating **putchar** function

Format:	repputc	Chip type	Use of multiplication/division instruction
Example:	repputc	0138	use

(5) For updating **putchar** function (supports simulator)

Format:	repputcs	Chip type	Use of multiplication/division instruction
Example:	repputcs	0138	use

(6) For updating **setjmp/longjmp** function (with restore/save processing)

Format:	repselo	Chip type	Use of multiplication/division instruction
Example:	repselo	0138	use

(7) For updating **setjmp/longjmp** function (without restore/save processing)

Format:	repselon	Chip type	Use of multiplication/division instruction
Example:	repselon	0138	use

(8) For updating bank function call processing routine

Format:	repbank	Chip type	Use of multiplication/division instruction
Example:	repbank	0138	use

(9) For updating branch table address for interrupt function

Format:	repvect	Chip type	Use of multiplication/division instruction
Example:	repvect	0138	use

(10) For creating library

Format:	mklib	Chip type	Memory model	Area	Use of multiplication/division instruction	Function I/F	Pascal function I/F
Example:	mklib	0138	normal	boot	use	new	notpascal

(11) For creating library supporting floating points

Format:	mkflib	Chip type	Area	Use of multiplication/division instruction
Example:	mkflib	0138	boot	use