General Notice

When using this document, keep the following in mind:

1. This document is confidential. By accepting this document you acknowledge that you are bound by the terms set forth in the non-disclosure and confidentiality agreement signed separately and /in the possession of SEGA. If you have not signed such a non-disclosure agreement, please contact SEGA immediately and return this document to SEGA.

2. This document may include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new versions of the document. SEGA may make improvements and/or changes in the product(s) and/or the program(s) described in this document at any time.

3. No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without SEGA'S written permission. Request for copies of this document and for technical information about SEGA products must be made to your authorized SEGA Technical Services representative.

4. No license is granted by implication or otherwise under any patents, copyrights, trademarks, or other intellectual property rights of SEGA Enterprises, Ltd., SEGA of America, Inc., or any third party.

5. Software, circuitry, and other examples described herein are meant merely to indicate the characteristics and performance of SEGA's products. SEGA assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples describe herein.

6. It is possible that this document may contain reference to, or information about, SEGA products (development hardware/software) or services that are not provided in countries other than Japan. Such references/information must not be construed to mean that SEGA intends to provide such SEGA products or services in countries other than Japan. Any reference of a SEGA licensed product/program in this document is not intended to state or simply that you can use only SEGA's licensed products/programs. Any functionally equivalent hardware/software can be used instead.

7. SEGA will not be held responsible for any damage to the user that may result from accidents or any other reasons during operation of the user's equipment, or programs according to this document.

NOTE: A reader's comment/correction form is provided with this document. Please address comments to :

SEGA of America, Inc., Developer Technical Support (att. Evelyn Merritt)
150 Shoreline Drive, Redwood City, CA 94065

SEGA may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

SEGA OF AMERICA, INC.
Consumer Products Division

SATURN
Software Library

Release 3.01
Supplemental Disk

Doc. # ST-208-110194

READER CORRECTION/COMMENT SHEET

Keep us updated!

If you should come across any incorrect or outdated information while reading through the attached document, or come up with any questions or comments, please let us know so that we can make the required changes in subsequent revisions. Simply fill out all information below and return this form to the Developer Technical Support Manager at the address below. Please make more copies of this form if more space is needed. Thank you.

General Information:

Your Name _____ **Phone** _____

Document number ST-208-110194 **Date** _____

Document name SATURN Software Library, rel. 3.01

Corrections:

Chpt.	pg. #	Correction

Questions/comments: _____

Where to send your corrections:

Fax: (415) 802-1717
Attn: Evelyn Merritt,
Developer Technical Support

Mail: SEGA OF AMERICA
Attn: Evelyn Merritt,
Developer Technical Support
150 Shoreline Dr.
Redwood City, CA 94065

==========

SATURN Software Library (Release 3.01 Supplemental Disk)

1. Supplemental Explanation

This disk (Rel. 3.1) contains libraries and tools which have been upgraded from Software Library Rel. 3 and debugged. Since this disk is configured the same as the Rel. 3 directory structure, it updates and adds libraries and tools when it is installed after Rel. 3 has been installed (unwound).

Note: In this instance the sound driver has been placed in the SATURN\SEGADRV directory.

2. Library Upgrade/Debug History

All versions of Rel. 3 were provided in Ver 1.01, but the versions of Rel 3.1 which are provided here are different for each library. Subject matter applicable to each library will be discussed next.

2.1 SPRITE Library (SEGA_SPR.LIB ver 1.02)

Ver 1.01->Ver 1.02 Changes

- Correction of allocBlock Function (private function) defect.

2.2 Scroll Library (SEGA_SCL.LIB ver 1.03)

Ver 1.01->Ver1.02 Changes

- Correction of SCL_SetAutoCol Mix function defect.
- Correction of SCL_VblankEnd function defect.

Ver 1.02->Ver 1.03 Changes

- Correction of SCL_SetConfig function defect (error in rotation scroll plane setting.) Correction of defect whereby garbage comes out in last line during x-axis revolution and y-axis revolution.

2.3 Sound Library (SEGA_SND.LIB ver 1.03)

Ver1.01->Ver_1.02 Changes

- Correction of SND_Init function memory zero clear part defect.
- Correction of SND_SetCdDaLev function parameter processing defect.
- Correction of PCM PAN setting defect.
- Deletion of SND_StartPcm function PCM start offset setting.

Ver 1.02->Ver 1.03 Changes

- Internal specification change due to DMA library change (Ver 1.02 and later).

2.4 File System Library (SEGA_GFS.LIB, SEGADGFS.LIB ver1.15)

Version upgrade from ver 1.01 to Ver 1.10

- Changes and cautions regarding how to use SCU-DMA (Transfer Mode GFS_TMODE_SCU).
- Additions to CD-DA file processing functions.
- File attribute changes.
- (GFS_GetFileInfo function output files are to conform to CD-ROM XA standard).
- GFS_Init function and GFS_LoadDir function function value change.
- Error code additions (GFS_ERR_BFUL, GFS_ERR_FATAL).
- Additions to tray open condition recognition methods.
- Changes in transfer mode depending on transfer destination address.

· Debug file related changes
  Commonality of GFMC_base definitions and changes in file identifiers.

(Changes from Ver 1.10 to Ver 1.11)
· Changes in GFS_MwCdRead usage methods.
· Access completion check (when GFS_NwCdRead is called) specification changes.

(Corrections from Ver 1.11 to Ver 1.13)
· Correction of defect whereby read errors occur when the CD buffer is FULL during play.

(Corrections from Ver 1.13 to Ver 1.15)
· Correction of unwanted time out error defect.

2.5 Stream Library (SEGA_STM.LIB) Ver 1.13

Version upgrade from Ver 1.10 to Ver 1.11

· STM_Init function specification changes.
· Additions to STM_OpenFid function and STM OpenFrange function arguments.
· Additions to STM_SetTrBuf function arguments.
· STM_SetLoop function specification changes (loop play / no play).
· Changes in specifications for when error codes are passed to error function second arguments.
· Changes in operation during loop play.
· Changes in specifications for transfer area settings.
· Even debugging is possible with sega_stm.lib.

(Changes from Ver 1.10 to Ver 1.11)
· Correction of defect in operation whereby fetch functions return (-1) or (0)
· Correction of defect whereby a back stream is opened immediately before the end position of the play range, and drive play mode FAD no longer changes.

(Changes from Ver 1.11 to Ver 1.12)
· Correction of defect whereby there is movement to a loop stream if there is a buffer full in mid-play range.

(Changes from Ver 1.12 to Ver 1.13)
· Correction of defect whereby file access denial occurs if STM_SetExecGrp(NULL) is executed.

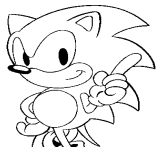
2.6 DMA Library (SEGA_DMA.LIB) Ver 1.03

Changes from Ver 1.01 to Ver 1.02

· Correction of defect whereby debug programs are mixed in with sega_dma.lib and debug information is added to them.
· Correction of defect whereby an interrupt status register which cannot be used by the hardware is used.

Changes from Ver 1.02 to Ver 1.03

· DMA parameter mask item contents described in the manual were wrong.
· Correction of DMA_ScuResult function defect.
· Re-registration of vectors saved during DMA end interrupt processing for each channel has been added.
· Correction of DMA parameter mask constant value defect
  (Low level SCU DMA functions and low level CPU DMA functions).
· Deletion of DMA_ScuAIIStop function.
· Deletion of DMA_ScuStop function.
· Change in specifications allowing DMA end interrupt mask registers which correspond to the channels used when low level SCU DMA is used.



2.7 Peripheral Libraries (SEGA_PER.LIB) Ver 1.02

Changes from Ver 1.01 to Ver 1.02

- Correction of defect in X and Y mouse movement amount data obtained.
- Correction of mouse digital device information defect.

2.8 V-BLANK Sample Programs

Changes

A change has been made whereby the SCL_VblankEnd function in the UsrVblankEnd function is performed at the top. (When pad data acquisition processing is heavy, it is possible that instances may arise of frame changes not being issued)

[Addendum]

Please be sure to refer to all documentation in <SATURN\SEGALIB\MAN> for details about version upgrades.

3. Sound Driver Version

Sound drivers were provided in Release 3 under the file name of SDDRVS.TSK (Ver 1.11) in <SATURN\SEGASMP\SND, but in this release they are provided in the <SATURN\SEGADRV) directory.

3.1 Sound Driver Version

Sound drivers are now provided in Version 1.24. In this version, the generation format (compression program) in the Convert Standard MIDI File of the SoundSimulator (supplementary) and the SoundDriver (thawing program) have been changed in order to reduce the sequence play load from Ver 1.2x.

For that reason, although sequence play load has been lightened, compatibility with the Ver 1.1x sequence format has been lost. Run the Convert Standard MIDI File in SoundSimulator Ver 1.24 or later or M6CNV 2 BIN OUT, and rewrite the Sequence Bank in Make Sequence Bank.

[Supplememt]

SoundSimulator Ver 1.24 now includes the M6CNV 2 BIN OUT tool in the SEGA sound tool (Macintosh Version) upgrade.

3.2 Sound driver Upgrade/Bug Fix (Debug) History

Changes from Ver 1.11 to Ver 1.12

- Noise during the PCM stream play LOOP has been corrected.
- Irregularities during PCM stream play stereo play have been corrected.
- Irregularities in PCM stream play PAN control have been corrected.

Changes from Ver 1.12 to Ver 1.13

- Irregularities in PCM Play Address upgrading during PCM stream play have been corrected.
- Correction of defect whereby operation continues even though there are no bank changes in sequence data. (In this and subsequent versions, no sound is generated if bank changes are not correctly included in sequence data.)

Changes from Ver 1.13 to Ver 1.14

- The fault whereby the PCM Play Address was not properly upgraded during multiple channel simultaneous play has been improved.

Details of Version Upgrade from Ver 1.14 to Ver 1.24

- Sequence data format change. (Sequence data must be converted.)
- Support for tempo change in the loop during sequence play.
- Improvement of time lag during tempo track play in sequence play.

[Addendum]

For details, please be sure to refer to MANSDRV.DOC in <SATURN\SEGADRV\MAN)

4. CD Tool Upgrade

4.1 CD Development Tools

Be sure to use the CD tools provided in this and later versions. For details about these, refer to the Virtual CD System User's Manual Ver1.00. (Release 3.)

4.2 VCDEMU.EXE (Ver 1.72)

Virtual CD Emulator

- Corrections from Ver 1.5x to Ver 1.70
  Implementation of multi-index and scan play.
  Scan play valid in real time emulation mode only.
- Improvements from Ver 1.70 to Ver 1.71
  Correction of defect during CD tray OPEN/CLOSE.
- Improvements from Ver 1.71 to Ver 1.72
  Correction of defect whereby the wrong file is obtained during DOS file retrieval in direct DOS mode. Correction of defect whereby row numbers are displayed as a minus in the upper right part of the screen. Correction of defect whereby the Lsn value is displayed as a minus during display of REL information sub-menus.

[Restrictions]

File interleave in direct DOS file access is not supported in this version

[Cautionary Items]

In order to operate VCDEMU.EXE Ver 1.7x and later versions, virtual CD board ROM must be replaced by Ver 3.2.

4.3 VCDPRE.EXE and VCDBUILD.EXE (Ver 3.04)

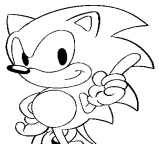
Corrections of the Disk configuration information file creation tool and CD image file creation tool.

Corrections from Ver 2.11 to Ver 2.16

- Corrective measures that without fail put directory code at the top of the sectors when directory files straddle multiple sectors because the number of files is very large.
- Designed to suspend processing when there is more than one file of the same name.
- Corrective measures were implemented for faults during directory configuration.
- I/O buffer was expanded to 32K bytes and speed increases of 2-6 times were implemented.

Corrections from Ver 2.16 to Ver 3.02

- This is a DOS extender (DOS/4G) version, and operation was confirmed up to a file number limit of approximately 10,000 files (in machines loaded with 8M bytes of memory).
- An error in the directory code position was corrected.
- An audio data TOC position slippage fault in the CDDA track was handled.



Corrections from Ver 3.02 to Ver 3.03

· A defect in the directory path table was improved.

Corrections from Ver 3.03 to Ver 3.04

· A fault whereby file positions became incorrect when directories were defined after track designation was handled.

[Cautionary Item]

In Ver 1.xx and Ver 3.xx (including Ver 2.XX) script file entry methods were changed. Persons who have been using Ver 1.xx up to now should change their script files. Please see the examples of script entry in 4 - 8.

4.4 VCDUTL.EXE (Ver 1.00)

From this version on, a tool for partial upgrade of image files is provided.

4.5 VCDMKTOC.EXE (Ver 1.22)

Tool for creating TOC information files for write-once CDs.

4.6 SWAP.EXE (Ver 1.00)

From this version on, a program for Endian conversion of CDDA data will be provided.

4.7 SEGACDW.EXE (Ver 1.00 Rel 1.00)

CD writer command. (There is no upgrade.)

4.8 Script Description Examples (EXSAMPLE.SCR, JVC.SCR, BTSMPFS.SCR)

These are examples of new script files for VCDPRE Ver2.xx and Ver 3.xx. Refer to them when changing (EXSAMPLE.SCR) scripts, etc. The script files provided in the software libraries (JVC.SCR and BTSMPFS.SCR) have been changed to the new script files.

[Addendum]

Substitution of EPROM (Ver 3.2) for virtual CD I/F board.
The defect whereby CD-DA data on virtual CDs cannot be played by multiplayers has been improved.

****

- **readme.doc classification : Development explanation files for each library**
- **File symbol name : MANGFS.DOC**
- **Relevant library symbol name : gfs**
- **Relevant library name : File system**
- **Version : 1.15**
- **Creator : M. S.**
- **Creation date : 10-19-1994**
- **Other messages : none**

****

1. Supplemental Explanation

1.1 DOS File Parameters

"1" is the initial value for DOS file fetch parameters. Because of processing constraints, only one sector can be transferred at a time, thus fetch parameters are invalid if set to any value except 1.

2. Changes from V1.10 to V1.1

2.1 Items RequiringAttention when using SCU-DMA

When the transfer mode is GFS_TMODE_SCU, the file system libraries use the DMA library to activate SCU-DMA. The level used is 0. Thus in GFS_Init, SCU-DMA level 0 interrupt is enabled by INT_ChgMsk. When using transfer mode GFS_TMODE_SCU, refer to the DMA library documentation.

3. Changes from V1.01 to V1.10

3.1 CD Pre-read Processing Changes

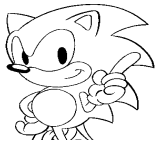
Change in how GFS_NwCdRead is used

After GFS_NwCdRead is called, pre-read from the CD becomes effective even if GFS_NwExecOne is not called. When pre-read from the CD is used, either the designated number of sectors are all transferred to the host or the operation could not be completed until access is stopped. (GFS_NwIsComplete does not return true.)

Access Completion Check [Important]

When GFS_NwCdRead is called, either data transfer ends or GFS_NwExecOne does not return completion. Do not wait for GFS_NwExecOne to return completion without calling GFS_NwFread. As shown in the example on Page 18 of the Release 2 Manual, caution must be exercised, since an infinite loop results if you wait for completion of NwExecOne immediately after GFS_ExecOne.

If GFS_NwCdRead is not called, completion of GFS-NwExecOne can be checked by the same procedure as in the previous version. When pre-reading from the CD, also perform a completion check by using GFS_NwGetStat to monitor the transfer complete byte at the same time.



The following shows an example of this:

```
#define      RD_UNIT 10
GfsHn  gfs;
Sint32  fid,  stat,  nbyte;
Uint32  buf[RD_UNIT*2048/4];

gfs = GFS_Open(fid);
GFS_NwCdRead (gfs,  100);
for  (I = 0;  I < 100/RD_UNIT;  ++I)  {
   GFS_NwFread (gfs,  RD_UNIT,  buf,  RD_UNIT*2048);
   while (GFS_NwExecOne (gfs)  != GFS_SVR_COMPLETED  {
      GFS_NwGetStat (gfs,  &stat,  &nbyte);
      /* Check whether the number of bytes designated by GFS_NwFread
      has been read */
      if (nbyte >= RD_UNIT*2048)  {
          break;
            }
            user ();            /* application processing */
      }
}
```

3.2 Addition of CD-DA File Processing Function

A CD-DA file processing function has been added. A CD file read is played on the music track which is designated by that file. However, for sound output SCSP must be set by the application.

File Control

The only control which the file system exercises over the CD-DA files is that of play and play range. The play mode is a default value (no repeat, move pickup).

How to Play

The following are methods of playing CD-DA files:

(1) GFS_Load
(2) GFS_Fread
(3) GFS_NwFread
(4) GFS_NwCdRead

In Methods (1) and (2) there is no return until play is ended. In methods (3) and (4) a server function must be called. When CD-DA play ends, the server function returns GFS_SVR_COMPLETED.

GFS_Seek is used to play from mid-disk. CD-DA files and CD ROM files cannot be simultaneously accessed.

Parameters for File Operation

Fetch mode, transfer mode, read parameters and fetch parameters cannot be changed for CD-DA files. If the following functions are called for CD-DA files, an error is returned.

GFS_SetGmode	GFS_SetTmode
GFS_SetReadPara	GFS_SetTransPara

3.3 File Attributes

In order to make file attribute values output by GFS_GetFileInfo, conform to the CD-ROM XA standard. The following changes have been made:

GFS_ATR_DIR	0x80
GFS_ATR_CDDA	0x40
GFS_ATR_INTLV	0x20
GFS_ATR_FORM2	0x10
GFS_ATR_FORM1	0x80
GFS_ATR_END_TBL	0x01

GFS_ATR_CDDA was added to provide an additional CD-DA file processing function. Its bit when it is a CD-DA file is 1. The names and meanings of other constants remain unchanged.

3.4 GFS_Init and GFS_LoadDir Functional Values

If NULL is set to the argument directory information control structure pointer, individual units of directory information held by the CD block file system are returned as function values.

3.5 Error Code Additions

The following error codes have been added.

GFS_ERR_BFUL

This error code occurs when the CD buffer becomes full during resident mode (GFS_GMODE_RESIDENT) file reads. Access order and the like should be adjusted so that a buffer full condition does not occur during resident mode file access.

GFS_ERR_FATAL

This error code is a notification that the CD drive is in a FATAL condition. If the file system detects this condition, it stops CD play (seeks home position) and tries to effect recovery from the error condition. If this error code is detected, retry processing.

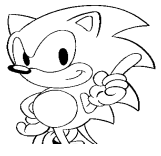
3.6 Recognition of Tray Open Condition

When the CD block interrupt factor register (HIRQREQ) DCHG bit (bit 5) is 1, this is also handled as an open tray condition. During development, confirm that the CD status is not open and perform processing to clear the DCHG bit. Since in the BOOT ROM clears the DCHG flag in the product version, the application must not clear it.

3.7 Transfer Mode Changes According to Transfer Destination Address

Even if transfer mode is set to GFS_TMODE_SCU, when the transfer destination address is contained in the following spaces, software transfer is forcefully performed by the CPU.

```
WORKRAM-L space          00200000H - 002FFFFFH
A-BUS space              02000000H - 058FFFFFH
```

3.8 Debug File Related

GFMC_base

The variable GFMC_base, which sets the top address of SIMM files, is defined in both sega_gfs.lib and segadgfs.lib. The GFMC_base in sega_gfs.lib exists solely for the sake of compatibility with segadgfs.lib. It has no effect on file system operation.

File Identifiers

The function which automatically added "." and ".." (current directory and parent directory) when CD files were not being used has been deleted. Accordingly, file identifiers are the same as when booted.

4. Bug Correction Information

4.1 Corrections from V1.11 to V1.13

The bug when there was a read error (GFS_ERR_CDRD) if the CD buffer became full during play has been corrected.

4.2 Corrections from V1.13 to V1.15

The bug when there was an unwanted time out error (GFS_ERR_TMOUT) was corrected.

**********end of file**********

- **Document Classification : 3rdSTEP (08-22-94) -> 3rdSTEP (10-12-1994) Changes**
- **Relevant library symbol name : dma**
- **Relevant library name : DMA**
- **Version : 1.03**
- **Creator : N. T.**
- **Creation date : 10-12-1994**
- **Other messages : none**

****

//

Cautionary Items when moving to a changed version

//

! ! IMPORTANT ! !

Be sure to recompile user programs which include header files, as the header files will have changed.

//

History of Changes

//

- **09-23-1994 Ver 1.02**
- **10-12-1994 Ver 1.03**

//

Details of Changes

//

Corrective Action

Corrective action for defects in 3rdSTEP has been taken as follows. The following defects were all "SCU" DMA.

1. Correction of defect whereby debug programs were mixed in and debug information was added to the sega_dma.lib.

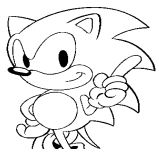
<Nature of Defect>

A debug program was mistakenly mixed into the sega_dma.lib. This debug program was writing in the neighborhood of address 6060000. Additionally, since /deb was designated to compile options, debug information was added, and the sega_dma.lib became larger than necessary.

<Corrective Action>

(1) Compiling was redone
(2) dma.bat dma.mk was properly corrected

2. Correction of defect whereby an interrupt status register which cannot be used by the hardware is used.



<Nature of Defect>
An interrupt status register which could not be used by the hardware was being used inside the library. (Because of the timing of the writes to the interrupt status register, there were times when interrupt signals could not be received. This is forbidden from a hardware standpoint.)

<Corrective Action>
Measures were taken so that the interrupt status register would not be used in the library, and the use of interrupt processing was substituted in its place. Because of this, the following specification changes have taken place in the DMA library. Since DMA end interrupt processing is used in the DMA SCU library, when DMA SCU is executed in V-BLANK IN interrupt processing and the like and in interrupt processing with a higher order of priority than DMA, it waits for the end of DMA until the end of interrupt processing with the high order of priority. The following are two ways of preventing this wait:

(1) Do SCU DMA in the main process
(2) Change the order of interrupt priority

The recommended method of interrupt priority order per (2) above is discussed separately.

<Specification Changes>
(1) SCU DMA high level
The following function specifications have been added:

Function Specification	SCU DMA initialization	DMA_ScuInit

Syntax	void DMA_ScuInit(void)
Input	none
Output	none
Function value	none
Action	Initializes SCU DMA. Must be performed before execution of DMA_ScuCopyMem () and DMA_ScuResult ()

!! Caution!!
DMA mode 0 transfer end interrupt processing is used by SCU DMA. Therefore, do not change the interrupt mask register of the DMA mode 0 transfer end interrupt.

(2) SCU DMA low level
The following function specifications and data specifications have been added.

Function Specification	Get DMA Status	DMA_ScuGetStatus

Syntax	void DMA_ScuGetStatus (DmaScuStatus *status, Uint32 ch)
Input	ch : Input channel
Output	status : status pointer
Function value	none
Action	Gets status of the designated DMA channel

!! Caution!!
There was a similar function in 2ndSTEP. However, it differs with this version as shown on the next page.

DmaScuStatus structure members and dxbk (DMA interruption flag) . dxwt (DMA wait flag) have been deleted.

Data Specification	Status	DMAScuStatus

Status is structured as shown below.

```
struct  {
    Uint32 dxmv;
}
```

The constants which can be used by each of these members are shown below.

dxmv /* DMA operation flag */

Constant	Explanation
DMA_SCU_MV	In operation
DMA_SCU_NO_MV	Not in operation

//

Corrective Action

Corrective measures have been taken for Ver 1.02 (09-30-1994) as shown below.

- **Common**

1. A DMA parameter mask item description in the manual was wrong.

<Defect>
The manual wrongly stated, "Bits which designate a DMA mask constant do not make settings to registers."

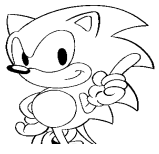
<Corrective Action>
The text should be corrected to read, "Bits which do not designate a DMA mask constant do not make settings to registers."

- **High Level SCU DMA**

1. Even when DMA_ScuResult () is executed before SMA_ScuMemCopy, a correct value is not returned.

<Defect>
Even when DMA_ScuResult () is executed before SMA_ScuMemCopy, a correct value is not returned.

<Corrective Action>
Make corrections to return a correct value.



- **Low Level SCU DMA**

1. The DMA end interrupt vectors are not returned to the origin.

<Defect>
When DMA starts, DMA end interrupt vectors are saved. However, after DMA ends, the saved vectors are not re-registered.

<CorrectiveAction>
Vectors saved during end interrupt processing for each channel are re-registered.

2. DMA Parameter Masks cannot be correctly designated.

<Defect>
Settings to DXR (read address register) were made unconditionally without relation to the settings of DMA parameter mask constants.

<CorrectiveAction>
Mask constants have been properly changed.

3. DMA_ScuAIIStop existed in the library even though it could not be used by the hardware.

<Defect>
The DMA forced end register used by DMA_ScuAllStop existed in the library even though its use is prohibited because of hardware limitations.

<CorrectiveAction>
This function was deleted.

4. DMA_ScuStop existed in the library even though it could not be used by the hardware.

<Defect>
DMA_ScuStop existed even though it had no hardware functions.

<CorrectiveAction>
This function was deleted.

- **Low Level CPU DMA**
  DMA parameter mask cannot be properly designated

<Defect>
Settings were made unconditionally to PR (priority mode) without regard to settings of DMA common parameter mask constants. In addition, settings were made unconditionally to SAR (DMA source address) without regard to settings of DMA parameter mask constants.

<CorrectiveAction>
Mask constants were properly changed.

Additions to Manual

Low Level SCU DMA
DMA end interrupt mask registers which correspond to the channels in use when low level SCU DMA is used must be permitted.

Correction of Sample Program

It has been changed so that it will operate in his version.

SMPDMA10.C **SMPDMA12.C**

********** end of file **********

****

- **Document Classification : 3rdSTEP (08-22-94) -> 3rdSTEP (10-12-1994) Changes**
- **Relevant library symbol name : snd**
- **Relevant library name : Sound l / F**
- **Version : 1.03**
- **Creator : N. T.**
- **Creation date : 10-12-1994**
- **Other messages : none**

****

//

Cautionary Items when moving to a changed version.

//

! ! IMPORTANT ! !

Be sure to recompile user programs which include header files, as the header files will have changed.

//

History of Changes

//

- **09-23-1994 Ver 1.02**
- **10-12-1994 Ver 1.03**

//

Details of Changes Ver 1.02

//

Corrective Action

The following measures were taken to correct defects in 3rdStep.

1. Corrective action for defect in SND_Init () memory zero clear part.
   The part which was mistakenly using memcpy was changed to memset.

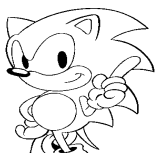
2. Corrective action for SND_SetCdDaLev () parameter processing defect.
   There was a defect whereby in the manual parameter (0 - 7) settings CD-DA sound was not output. This was corrected by designating manual parameters so that sound can be output.

3. Corrective action for PCM PAN setting defect
   There was a defect whereby the PAN setting in SND_StartPcm () could not swing to the right. This is a sound driver defect. Sound driver Ver 1.12 should be used. (The 08-22-94 sound driver is Ver 1.11)

Specification Changes

The following specification changes have been made starting with 3rdSTEP.

1. Deletion of SND_StartPcm () PCM start offset setting.
   The PCM start offset setting function has been deleted.



Other

A CD-DA and PCM stream simultaneous play sample program has been added. This is SMPSND5. SMPSND4 has been restructured.

//

Details of Changes Ver 1.03

//

Specification Changes

(Ver 1.10 [08-22-1994 -> Ver1.02[09-30-1994]) The internal specifications of the sound l/F library have been changed. The external specifications have also been partially changed. Details of these changes are shown below.

Internal Specification Changes

DMA_ScuInit (), which is a SCU DMA high level initialization function in SND_Init(), has been added. Since DMA_ScuInit () cannot be executed in interrupt processing, SND_Init () should be made so that it does not execute in interrupt processing.

External Specification Changes

- SND_Init should not be executed in interrupt processing. (For details, see internal specification changes.)
- When interrupt priority is the default, SND_MoveData () should not used in interrupt processing. In order to use it in interrupt processing, interrupt priority should be changed so that DMA end interrupt can be used in interrupt processing. Use the system library to change interrupt priority. An example is shown in the sound sample program (SMPSND4).

Additions to the Manual

Access to Sound Memory

Frequent access of the sound memory is prohibited. If there is frequent access of the sound memory by the host, the sound driver ceases to operate because it cannot see that memory.

(Example)

```
SND_StartSeq (1,0,0,0):
while(SND_SEQ_STAT_MODE(status) == SND_MD_STOP {
     SND_GetSeqStat(&status, 0);
}
```

In the above example, there is the possibility of a permanent loop. The reason for this is that, because the host frequently accesses the status memory, the sound driver cannot write a STOP sequence to the status memory even though the sequence has ended.

Corrections to Sample Programs

SMPSND3.C

This sample will be deleted. The reason for this is that the interrupt status register used in this sample can no longer be used as a result of SCU hardware limitations.

SMPSND4.C

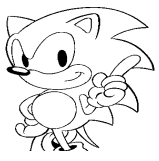
Interrupt priority has been changed, and SND_MoveData () was changed so that it can be used in sound request interrupt processing.

SMPSND5.C

- CDC_Init () Parameter change
  Changes were made because of CDC library limitations. The parameters were made as follows: CDC_Init (0x00, 0x00, 0x05, oxof);

- Interrupt priority was changed, and SND_MoveData () was changed so that it can be used in sound request interrupt processing.

********** end of file**********



****

- **File symbol name : manstm.doc**
- **Relevant library symbol name : stm**
- **Relevant library name : Stream System Library**
- **Version : 1.13**
- **Creator : N. T.**
- **Creation date : 10-07-1994**
- **Other messages : none**

****

Cautionary Items for Stream System Libraries Ver 1.10 and Later

1. Function Specification Changes

Functions in versions preceding Ver 1.01 should be replaced as follows:

(a) STM_Init ()

```
Uint8 work[STM_WORK_SIZE(12, 24);
        .
        .
stm_Init (12, 24, work);
```

(b) STM_OpenFid(grp, fid. &key), STM_OpenFrange(grp, &frange, &key)

```
STM_OpenFid(grp, fid, &key, STM_LOOP_READ);
STM_Open Frange(grp, &frange, &key, STM_LOOP_READ);
```

(c) STM_SetTrBuf(stm, buf, BUF_SIZE)

```
STM_SetTrBuf(stm, buf, BUF_SIZE, STM_UNIT_WORD);
```

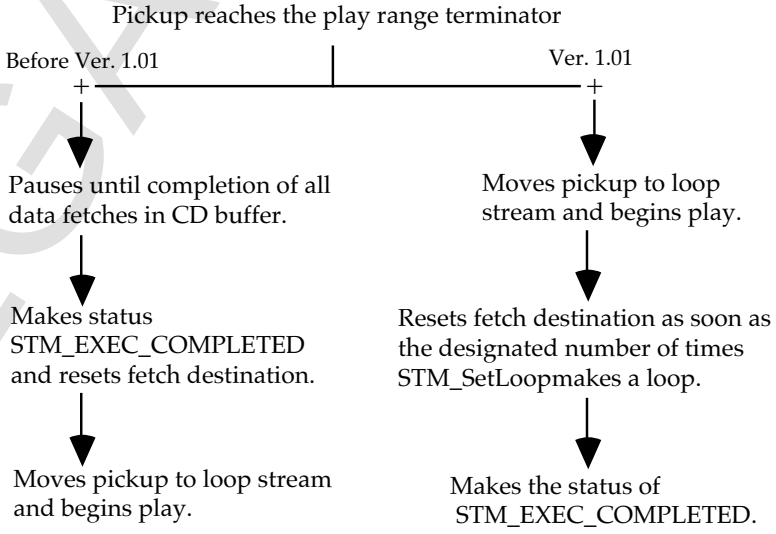
(d) STM_SetLoop(grp, stm)

When a loop is played STM_SetLoop(grp, stm, STM_LOOP_ENDLESS),
When a loop is not played STM_SetLoop(grp, stm, 1);

(e) Error code is passed to error function second argument

2. Operation During Loop Play

When pickup reaches the play range terminator, operation differs as shown below:



The program on the left which was written for Ver 1.01 can be made to operate in the same way regardless of the previous changes by inserting STM_SetLoop(grp, STM_LOOP_DFL, 1) as shown on the right.

```
grp = STM_OpenGrp ();
stm = STM_OpenFid(grp, FID1, &key);
STM_SetTrBuf(stm, buf, BUFSIZE1);
STM_SetExecGrp(grp);
for (i = 0; i < LOOPMAX; ) {
      stat = STM_ExecServer ();
      if (stat == STM_EXEC_COMPLETED) {
            i++
         }
        user ();
     }
```

```
grp = STM_OpenGrp ();
stm = STM_OpenFid(grp, FID1, &key,
            STM_LOOP_READ);
STM_SetTrBuf(stm, buf, BUFSIZE1,
            STM_UNIT_WORD);
STM_SetLoop(grp, STM_LOOP_DFL, 1);
STM_SetExecGrp(grp);
for (I = 0; I < LOOPMAX; ) {
     stat = STM_ExecServer ();
     if (stat == STM_EXEC_COMPLETED) {
          i++;
       STM_SetLoop(grp, STM_LOOP_DFL, 1);
     }
     user ();
  }
```

3. Prior to Ver 1.01, when transfer area setting → transfer function registration → transfer function cancellation was done, the next transfer destination was the initially registered transfer area, but in Ver 1.10 transfer areas must be reset. Moreover, the maximum number of transfer sectors and the transfer mode revert to the default status.

4. In order to use the stream system, the file system and CD communication interface must be linked.

5. All files in CD-ROM, virtual CD, SIMM, SCSI can be handled by sega_stm.lib. However, linked file system libraries must be properly used.

6. There is a change in sega_stm.h. Source files which include sega_stm.h should all be recompiled.

7. Cautionary items
(a) If the stream is closed immediately after the close fetch function of the stream which registers the fetch function has returned (-1), it is continuously called until the fetch function returns a value of 0 or greater. When the fetch has ended, a value of 0 or above should be returned.

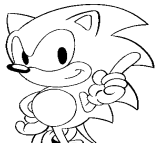
8. Bug corrections in Ver 1.10 and later
The following corrections are being made in the file system and settings. File system Version 1.11 should be used.

(Ver 1.11)
- Correction of the defect where fetch functions sometimes return (-1) or (0).
- Correction of the phenomenon whereby a back stream is opened immediately before the end position of the play range, and drive play mode FAD no longer changes

(Ver 1.12)
- Correction of the phenomenon whereby if the buffer becomes full in mid-play range, the system moves to a loop stream.

(Ver 1.13)
- Correction of the phenomenon whereby file access denial because of a runaway/infinite loop/file system occurs if STM_SetExecGrp(NULL) is executed.

********** end of file **********



****

- **File symbol name : MANSCL.DOC**
- **Relevant library symbol name : stl**
- **Relevant library name : VDP2 Library**
- **Version : 1.03**
- **Creator : K. M.**
- **Creation date : 10-25-1994**
- **Other messages : none**

****

Differences from 2ndStep Libraries (Ver 0.90)

Basically, it is compatible at the source code level. By recompiling them, applications written in 2ndStep can be operated as they are. However, caution is required in the following cases:

- Table initialization functions are not used.
  [Corrective action] Initialize regarding the newly added table item.
- When memcopy() is used:
  [Corrective action] Change the function name to SCL_Memcpyw (). If the register, color RAM and the like are not in an area in which a transfer cannot be made, it can be replaced with the C language function memcpy ().

- **Five Newly Added Functions**
  - SCL_SetRotateCenterDis ()
    Sets center point when making revolving scroll surfaces revolve on the screen.
  - SCL_SetCoefficientData ()
    Sets enlargement and reduction data to the revolve parameter coefficient table.
  - SCL_DisableBlur ()
    Disables shading operations.
  - SCL_DisableLineCol ()
    Disables line color screen.
  - SCL_AboreAutoVe ()
    Forces end of automatic VE.

- **Revolving Screen ZAxis Direction Movement Specification Change**
  When the angle of revolution was 0, Z coefficient designation was effective.
  (When values are changed at a zero angle of revolution, they do not differ from the apparent enlargement and reduction.)

- **Revolve Parameter Coefficient Table Can Be Set to Color RAM**
  One item has been added to the VRAM configuration data structure. (Total of three items added.)

- **The Revolve Parameter Coefficient Table Set Position Can Be Designated in 0x400 Units.**
  Two items have been added to the VRAM configuration data structure. (Total of three items added.)

- **Correction of Fault in One-Word Pattern Name Data Specifications in Scrolls**
  In the 2ndStep version, there was a fault whereby character pattern data in the last part of VRAM could not be positioned. One item has been added to the scroll configuration data structure.

- **Library Source Files have been Subdivided to Make Execution Type Files Smaller.**

- **The Sin(90) and Cos(0) Trigonometric Functions which were used Internally have been Corrected to 1.0.**

Differences from 3rdStep Libraries (Ver 1.00)

- SCL_SetAutoColMix () bugs have been corrected.
- SCL_VblankEnd ()bugs have been corrected.

Differences from 3rdStep Libraries (Ver 1.02)

- SCL_SetConfig () bugs have been corrected.
  There was a mistake in the revolving scroll surface setting. (Screen over-processing and plane size.)
- Bug whereby garbage appears on last line during X-axis revolution and Y-axis revolution.

Library Restructuring

In 3rdStep, the VDP2 library register buffer to facilitate customizing of library function strength, speed and the like is described in the manual. A simple explanation of how to add functions is given.

- Screen display ON / OFF (designation of ON / OFF for the entire screen)

[Screen ON)

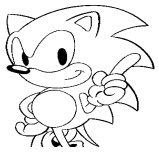
```
void disp_on ()
{
       Scl_s_reg.tvmode |= 0x8000;
       SclProcess = 1;
       SCL_DisplayFrame () : /* when frame change is not auto */
}
```

[Screen OFF]

```
void disp_off ()
{
       Scl_s_reg.tvmode &= 0x7fff;
       SclProcess = 1;
       SCL_DisplayFrame () : /* when frame change is not auto */
}
```

- When change of only the X coordinates of a normal scroll screen (0) without using other functions is desired.
  [can be used anywhere]
  Scl_n_reg_.n ()_move_x = ???; (Scroll coordinate designation)
  [In V-Blank In interrupt]
  (*(Uint16 *) (0x25f80070)) = Scl_n_reg.n0_move_x;

********** end of file **********



****

- **Document Classification : All Library Development Explanation Files**
- **File symbol name : manspr.doc**
- **Relevant library symbol name : spr**
- **Relevant library name : Sprite**
- **Version : 1.02**
- **Creator : H. E.**
- **Creation date : 10-4-1994**
- **Other messages : none**

****

1. Additional Explanations

1.1 Sprite isplay BasicProcessing Library

None

1.2 Sprite Display Expand Processing Library

- If #define in spr_2c. c shown below is changed and compiled, commands to VRAM can be transferred and characters can be transferred using SCU-DMA.

  #define USE_DMA_LEVEL n
  n = 0 : DMA is not used (default)
  1 : Only command transfer can be done by DMA
  2 : Only character transfer can be done by DMA
  3 : Command and character transfer can both be done by DMA.

  When command transfer is done by DMA, once commands have been written to the work area, configured and transferred, the number of commands at that time is defined below. In S model DMA transfer one-time transfer size is limited to 2K bytes or less, and the number of commands is therefore 80.
  #define CMD_BUF_MAX 80

- Addition of Work Area Size Table

Work Area Size Table	SprAreaSize

```
struct SprAreaSize {
    Uint16      commandMax:         /* Number of possible command area registrations     */
    Uint16      gourTblMax:         /* Number of possible Gouraud table registrations    */
    Uint16      lookupTblMax:       /* Number of possible color lookup table registrations*/
    Uint16      charMax:            /* Number of possible character registrations        */
    Uint16      drawPrtyBlkMax:     /* Number of drawing priority table entries         */
    Uint16      useCommandCount:    /* Number of currently registered commands          */
    Uint16      charRemainBlkCount: /* Total number of open blocks in character area    */
    Uint16      charNullBlkMaxSize  /* Character open block maximum area block size     */
};
```

- Addition of SPR_2GetAreaSizeInfo Routine

Work Area Size Table	SPR_2GetAreaSizeInfo

[Syntax]	void SPR_2GetAreaSizeInfo (SprAreaSize *areaSizeInfo);
[Input]	areaSizeInfo : work area size notification area address
[Output]	areaSizeInfo : Notification of current work area size
[Function value]	none
[Function]	Returns size and use status of each work area set during initialization

- The following bug which was in the allocBlock routine was corrected (Ver 1. 01 bug.) When the requested number of blocks and the size of the open blocks is the same, allocation cannot properly be done.

1.3 Sprite 3D Display Library

- The following items have been changed so that 2ndStep 3D data can be used in 3rdStep. They must be added.
  (1) The following items will be corrected and added to the SprObject3D table.
  - Uint16 **shdldxTbl = NULL
    When the shading index table is not used, it is NULL.
  - Fixed32 surfNormK = MTH_FIXED(1, 0)
    Calculated compensation value of surface normal vector

  (2) The following items will be added to the SprCluster table.
  - SprInbetInf *inbetInf = NULL
    3D object indirect connect polygon information
  - void *transStart = NULL
    User call back routine before the start of coordinate conversion
  - void *transEnd = NULL
    User call back routine after the end of coordinate conversion
  - void *context = NULL
    User Context Area

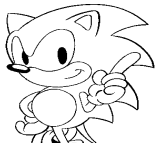
- Be sure to note that in 3rdSTEP the SPR_3SetView () parameters are being changed. "pivotViewPoint" is gone and in its place ViewCoordPoint is being added. Changes should be made as follows:
  2nd) SPR_3SetView (0, &viewPoint, &viewPoint, &viewAngle, ROT_SEQ_ZYX);
  3rd) SPR_3SetView (0, &viewPoint, &viewAngle, ROT_SEQ_ZYX, 0);

- Restrictions
  DSP is not to be used, as writeback from DSP to work RAM does not operate normally. Therefore, even if SPR_3USE_DOUBLE_BUF is designated by a 3D SPRITE work area definition macro, execution speed does not increase. In order to use DSP, refer to MANMTH.DOC and correct the MTH library.

2. Plans for Expanded Functions and Performance

None

********** end of file **********



****

- **Document Classification : 3rdSTEP (08-22-94) -> 3rdSTEP (10-12-1994) Changes**
- **Relevant library symbol name : PER**
- **Relevant library name : System/Peripheral**
- **Version : 1.02**
- **Creator : N. T.**
- **Creation date : 09-30-1994**
- **Other messages : none**

****

//

♦ **History of Changes**

//

• **09-23-1994 Ver 1.02**

//

♦ **Details of Changes for Ver 1.02**

//

Corrective Action

Corrective action for defects in 3rdSTEP have been taken as follows.

1. Correction of defect whereby X and Y mouse movement amount data obtained was wrong.

<Nature of Defect>
Mouse X and Y movement amount data minus values which had been obtained were wrongly being converted to positive values.

<Corrective Action>
Matched to manual specifications. (Adjusted so that data from -128 to 127 can be obtained.)

2. Correction of mouse digital device information defect.

<Nature of Defect>
Because of the defect in (1.) above, mouse digital device information was wrong.

<Corrective Action>
Same as (1.) above.

********** end of file **********

****

- **readme.doc classification : CD development tool explanation file**
- **File symbol name : MANBINCD.DOC**
- **Relevant library symbol name : CD development tool**
- **Relevant library name : File system**
- **Version :**
- **Creator :**
- **Creation date : 10-25-1994**
- **Other messages : none**

****

1. CD Development Tools

CD development tools must always be used from this version forward. These tools should be installed in the directories which comprise software library <SATURN\SEGABIN\BINCD>.

2. VCDEMU.EXE (Ver 1.72)

2.1 Corrections from Ver 1.5x to Ver 1.70

Multi-index and scan play have been implemented. (Note that scan play is only effective in real time emulation mode.)

2.2 Corrections from Ver 1.70 to Ver 1.71

Correction of a CD tray OPEN/CLOSE defect.

2.3 Corrections from Ver 1.71 to Ver 1.72

Correction of a defect whereby during DOS file retrieval in direct DOS mode the wrong files were obtained. Correction of the defect whereby row numbers on the lower right of the screen were displayed as minus numbers. Correction of the defect whereby during the REL Information submenu display, the value of Lsn was displayed as a minus.

2.4 Restrictions

The current version does not support file interleave in direct DOS file access.

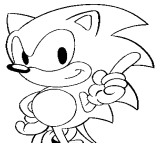
2.5 Cautionary Items

In order to operate VCDEMU.EXE Versions 1.7x and later, virtual CD board ROM must be replaced by Ver 3.2.

3. VCDPRE.EXE and VCDBUILD.EXE (Ver 3.03)

3.1 Changes from Ver 2.11 to Ver 2.16

- Steps have been taken to ensure that directory code is always at the top of a sector when the number of files is large and directory files straddle several sectors.
- Action has been taken to stop processing when the same file names coincide.
- Corrective action has been taken for faults in directory configuration.
- Speed increases of from 2 to 6 times have been built in.
- When there are files with the same name, process will be interrupted.



3.2 Changes from Ver 2.16 to Ver 3.02

- This is to become a DOS extender (DOS/4G), and operation has been confirmed up to a limit of about 10,000 files. (in machines loaded with 8 MB of memory).
- An error in directory record position was corrected.
- A fault whereby the CDDA track audio data TOC position was slipping was handled.

3.3 Changes from Ver 3.02 to Ver 3.03

- A defect in the directory path table was improved.

3.4 Changes from Ver 3.03 to Ver 3.04

- A defect whereby file position was wrong when directories were defined after track designation was handled.

3.5 Cautionary Items

In Ver 1.xx and Ver 3.xx, script file description methods are different. Persons who up to now have been using Ver 1.xx should change their script files. Please see the examples of script description.

4. VCDUTL.EXE (Ver 1.00)

Tool for partial update of image files.

5. VCDMKTOC.EXE (Ver 1.23)

TOC information file generation tool for write-once CDs.
(An upgraded version is being prepared to handle VCDPRE.EXE and VCDBUILD.EXE bugs and support upgrades.)

6. SWAP.EXE (Ver 1.00)

CDDA data Endian conversion program.

7. SEGACDW.EXE (Ver 1.00 Rel 1.00)

CD writer command.

8. Script Description Example (EXSAMPLE.SCR, JVC.SCR, BTSMPFS.SCR)

This is an example of the new script file for VCDPRE.EXE Ver 2.XX. (EXSAMPLE.SCR) Please refer to it when changing scripts. The script files (JVC.SCR and BTSMPFS.SCR) provided in the software libraries have been changed.

Other: Replacement of EPROM (with Ver 3.2) for Virtual CD I/F board

This has improved the defect whereby CD-DA data on virtual CDs could not be played on multiplayers.

********** end of file **********

****

- **readme.doc classification : Sound driver explanation file**
- **File symbol name : MANSDDRV.DOC**
- **Relevant library symbol name : SDDRV**
- **Relevant library name : Sound drive**
- **Version : 1.14 / 1.24**
- **Creator :**
- **Creation date : 10-21-1994**
- **Other messages : none**

****

1. Details about Sound Drivers

SDDRVS.TSK	Sound driver for use in the actual product.
SDDRV.TSK	Sound driver for sound simulator use.
SYSTBL.TSK	System information table for sound simulator. (The system information table for use in the actual product is contained in SSDRVS.TSK.)

Be sure to use SDDRVS.TSK as the sound driver incorporated in games. (Be sure to read the SATURN Sound Driver System Interface Manual regarding execution methods.)

2. Sound Driver Versions

Sound drivers are provided with this version (1.14) and with Ver 1.24. In order to reduce the sequence play load in versions later than 2.x, the generation format (compression program) in the Convert Standard MIDI File of the Sound Simulator (supplement) and SoundDriver (thaw program) have been changed.

For this reason, while sequence play load has been lightened, there is no longer compatibility with Ver 1, 1x sequence format. Make a Convert Standard MIDI file with SoundSimulator Ver 1.24 or later or with M6CNV 2 BIN OUT and rewrite the Sequence Bank with Make Sequence Bank.

[Supplementary]
SoundSimulator Ver 1.24 is SEGA-supplied sound tool version upgrade (Macintosh edition), and the M6CNV 2 BIN OUT tool is included.

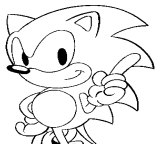
3. History of Sound Driver Changes

3.1 Changes from Ver 1.11 to Ver 1.12

- Noise during PCM stream play loops has been corrected.
- Trouble during PCM stream play stereo play has been corrected.
- PCM stream play PAN control trouble has been corrected.

3 .2 Changes from Ver 1.12 to Ver 1.13

Trouble in PCM Play Address upgrade during PCM stream play has been corrected.



3.3 Changes from Ver 1.13 to Ver 1.14

A fault whereby the PCM Play Address was not properly upgraded during multiple channel simultaneous play has been improved.

3.4 Changes from Ver 1. 14 to Ver 1.24 (version upgrade)

- In order to reduce the sequence play load, the generation format (compression program) in the Convert Standard MIDI File of the Sound Simulator (supplement) and SoundDriver (thaw program) have been changed. (For Ver 1. 1x and later versions, sequence data conversion is required.
- Tempo changes in the loop during sequence play have been addressed.
- Tempo track play time lag during sequence play has been improved.

4. Cautionary Items

Bank changes must be set in the sequence data. While it was clearly explained in the manual that original bank changes should be set, in Ver 1.12 and earlier versions sound was produced even though there were no bank changes. From Ver 1.13 on, however, checks reveal that if bank changes are not set, no sound is produced.

********** end of file **********