

# **SuperPlay-Library**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> SuperPlay-Library	
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>
WRITTEN BY		February 12, 2023
<i>SIGNATURE</i>		

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>SuperPlay-Library</b>	<b>1</b>
1.1	SuperPlay Library Documentation . . . . .	1
1.2	Copyright . . . . .	2
1.3	Disclaimer . . . . .	2
1.4	Distribution . . . . .	2
1.5	Abilities, Purposes and much more . . . . .	3
1.6	Installation and Configuration . . . . .	3
1.7	Overview of currently available SPObjects . . . . .	3
1.8	Software supporting SuperPlay.library . . . . .	4
1.9	Thanks, greetings, credits and the rest . . . . .	4
1.10	How to contact the author . . . . .	4
1.11	The future of the SuperPlay.library . . . . .	5
1.12	History . . . . .	5
1.13	Credits . . . . .	8
1.14	Requirements for the SuperPlay.library Package . . . . .	9
1.15	ST.spobject . . . . .	10
1.16	MED.spobject . . . . .	10
1.17	JAM.spobject . . . . .	11
1.18	8SVX.spobject . . . . .	12
1.19	SPO.spobject . . . . .	12
1.20	VOC.spobject . . . . .	14
1.21	WAV.spobject . . . . .	14
1.22	SoundDT.spobject . . . . .	15

---

# Chapter 1

## SuperPlay-Library

### 1.1 SuperPlay Library Documentation

superplay.library V4.4

- FREeware -

© 1994-95 by Andreas R. Kleinert. All rights reserved.

This program was written under OS V3.1 and is therefore fully compatible. It needs OS V2.04+.

Release Date : 26.08.1995

Copyrights

Disclaimer

Distribution

Requirements

How to install and configure it

Short: Purpose and Abilities

Supporting Software

SPObject Descriptions/History

Possible future enhancements ...

History

Thanks and Greetings

Credits

How to contact the author

-

---

— //  
Only \X/ Amiga makes it possible!

Die CHAOS-Theorie:

"Dabei geht es zum Beispiel darum, den verdammten Schmetterling zu finden, dessen Flügelschläge die vielen Stürme in letzter Zeit verursacht haben."  
(Terry Pratchett in "Total verhext")

Dedicated to all people, who actually READ docfiles ;-)

## 1.2 Copyright

The program superplay.library V4.4 and its documentation files are (C)opyright 1994-95 by Andreas R. Kleinert. All rights reserved.

Usage of the "superplay.library" for non-commercial programs (Freeware, Shareware), is allowed without any restrictions. With commercial programs an explicite license is required.

Some of the mentioned names or products above may be copyrighted by companies or trademarks of companies.

Remark: Drawericons have been taken from Fred Fish's AmigaLibDisks.

## 1.3 Disclaimer

The author takes no responsibility for any results of the use of this program.

This software is provided "AS IS" and there is no warranty of any kind, so that you use this software at your own risk.

## 1.4 Distribution

The program superplay.library V4.4 is freely distributable (FREWARE). You may copy it, if the copyright notice is left intact and all of its parts are included in the distribution.

This program must not be included in commercial packages or commercial program collections without my written permission.

This program must not be sold in any way, but it is allowed to take a nominal fee including the costs for copying.

This program may be put on public domain disks or included in public domain disk libraries.

---

Special permission hereby goes to Fred Fish's AmigaLib-Disks and the german series (in alphabetical order) :  
 AmigaSzene, BerndSPD, FRANZ, GPD, SaarAG, TAIFUN and TIME.  
 Also CD-ROMs by Fred Fish, AmiNet and ADX (Amiga Demos and Tools).

This program may also be distributed via electronic mail and may be put into mailboxes as long as the redistribution conditions are respected in all points.

By using or distributing this program you automatically agree to all of the above conditions and terms.

## 1.5 Abilities, Purposes and much more

The "superplay.library" is used in many of my programs and it consists of many functions, which make playing, saving and converting of various sound formats (SoundTracker, MED, IFF-8SVX, VOC, ...) much easier. External Player-Libraries (SPOjects) allow highest possible flexibility and easy to do expansions.

Just send me your own SPOjects if you want to have them added to the main distribution archive or onto the developer disk.

## 1.6 Installation and Configuration

Just copy the "superplay.library" into your "LIBS:"-directory. Copy all the SPOjects to "LIBS:spobjects/".

## 1.7 Overview of currently available SPOjects

Available SPOjects (as far as included) :

SPOject	Formats	Type	Write
	8SVX		
	IFF-8SVX (unencoded)	Sample	Sample
	JAM		
	JamCracker V1.00+	Module	-
	MED		
	MMD0	Module	-
	SPO		
	SPO V1.00	Sample	Sample
	ST		
	Sound/ProTracker	Module	-

VOC	PC VOC V1.10	Sample	Sample
WAV	RIFF-WAVE (Windows)	Sample	Sample
SoundDT	OS3 Sound Datatypes	Sample	-

## 1.8 Software supporting SuperPlay.library

The following programs are currently supporting "superplay.library":

- SuperPlay           by Andreas R. Kleinert
- SimplePlay         by Andreas R. Kleinert
- DosXS               by Frank Dittmar

## 1.9 Thanks, greetings, credits and the rest

in alphabetical order :)

Alex     Carbin           (TuC / PowerBrei Red.),  
 Fred    Fish             (AmigaLibDisks),  
 Gerd    Frank            (BerndsPD),  
 Stefan Grad             (GPD),  
 Stefan Kremer          (TuC / PowerBrei Red.),  
 Alex    Lange            (Time PD-Serie),  
 H.P.    Lattka           (Franz PD-Serie),  
 Andreas Manewaldt      (Taifun PD-Serie),  
 Michael Petrikowski    (Amiga Szene),

and all I forgot !

## 1.10 How to contact the author

You may reach me the following way.  
 Send bug-reports, money or whatever to:

Andreas R. Kleinert,  
 Grube Hohe Grethe 23,  
 D-57074 Siegen,  
 Germany.

Will change in late 1995,  
 but snail mail will be routed.

Phone: +49-271-331859 (weekdays after 18.00h)  
 +49-271-332147 (weekdays after 18.00h)

When calling via phone you may leave a message,

if I'm not available. - Andy

E-Mail: Fido Andreas Kleinert 2:2457/435.10  
 Usenet/InterNet Andreas\_Kleinert@SuperPlay.ftn.sub.org  
 Andreaskleinert@n2susx.sauerland.de

If nothing else works, try one of these Fido-InterNet gateways:

Andreas\_Kleinert@p10.f435.n2457.z2.fido.sub.org (in Germany)  
 Andreas\_Kleinert@p10.f435.n2457.z2.fidonet.org (USA or other)

When reporting any bugs, please don't forget to include a detailed description of the bug and tell me, if it is reproduceable or not. Also mention the version number of SuperPlay.library (and e.g. SuperPlay) which caused the bug and describe your system configuration (Amiga model CDTV/CD-32/600 (HD)/500 (+)/1000/1500/1200 (HD)/1300/2000/2500 (UX)/3000 (+/T)/4000 (/030/040/060) (T)/DraCo, Kickstart/OS Version, RAM, HardDisk, GfxCard, any special Hardware/Software). Since their recently have been some problems: please don't forget to mention, whether your machine is accelerated in any way, or not (68020/.../060, 68881/882, PMMU).

## 1.11 The future of the SuperPlay.library

The Future (or: The Undiscovered Zone ;-)  
 =====

Well, ideas cannot be planned and creativity is not able to be calculated ;-)

So I can only express, what I'm thinking about the future of the library: There are many things, which I want to implement, on the other hand there are things, which are just necessary to be implemented sooner or later. So much work has to be done (still).

I'll try to do this, but I hope that there's enough support from the Amiga programmers all over the world (via Inter/AmiNet, Fido, etc.) to help me to continue doing so.

This does not only mean financial support, but also additional programming efforts like third-party improvements on the library (why not also write your own SPOjects ?)

- Andy

## 1.12 History

V4.4 (26.08.1994) :

-----

- recompiled main library with SAS/C V6.55
- recompiled all supplied SPOjects with SAS/C V6.55
- all modules now do save A1 within assembler routines
- completely rewrote documentation:



- moved english docs to a new guide files, which also includes docs for the single SPObjects
- removed german docs - replaced with "LiesMich.doc"
- removed redundant information
- added note about distribution and commercial licensing
- added note about Frank Dittmar's DosXS, which is allowed to use superplay.library since a long time already
- overall reworked documentation
- updated SuperPlay to V4.4
- completely replaced Installer-Script
- and so on

#### V4.3 (13.12.1994) :

- 
- added JamCracker support (JAM.spobject)
  - added OS3 Sound DataTypes support (SoundDT.spobject). Needs at least superplay.library V4.3, since some additional work has to be done.
  - fixed bug in ST sample extraction routine (was only possible with XPK packed modules)
  - fixed internal bug inside all SPObjects.

#### V4.1/4.2 (02/09.12.1994) :

- 
- file recognition did not work with V4.1 (UNUSEABLE)
  - adjusted some version descriptions in the Reference-Docs
  - added new SPL\_GetFileType() function, like in superview.library
  - SPObject-Docfiles had been included twice in different directories (old and new docs). Removed.  
(-> Gerd Frank)

#### V3.1 (21.07.1994) :

- 
- This is not really a bug-fix, but more an additional feature : increased Version to 3, because it was not save in V2.1 to pass a NULL value for SuperWrite()'s "source\_handle" parameter, although this is supported by the V2 SPObjects (now). If you plan to add this feature to your software, request at least V3+ !

#### V2.1 (20.07.1994) :

- 
- introduced "SampleLists" and added two new functions to superplay-Library and Version 2 SPObjects for getting and setting of the Lists. Added two Example-Programs which demonstrate usage of SampleLists.
  - re-arranged directory structure of most sub-dirs
  - all NON-DISK loading and saving, e.g. from/into Clipboard DID NOT WORK. Specifications had to be changed slightly - in a compatible way - to re-arrange READING to work (see Reference-Docs, SPL\_InitHandleAsClip-Function). For WRITING only superplay-Library had to be changed. (This bug-fix has not yet been tested, since SuperPlay does not support Clipboard-Access, but it has been done the same way as in superview.library, so it should work ...)
  - ST.spobject and MED.spobject now allow extraction of Samples
-

- of Modules
- (see example-programs "SampleListDemo" and "SampleListPlay")
- fixed bad bug in ST.spobject's file-recognition (see there)

V1.6 (02.07.1994) :

-----

- new version of SuperPlay (GUI improved)
- changes to ST.spobject and WAV.spobject
- added ".fd"-files to "Programmers/fd", which might be useful for Basic programmers (".bmap" generation) or for generating Link-Libraries
- last not least : I'm now reachable via EMail (Fido : 2:2457/245.10)

V1.5 (17.05.1994) :

-----

- use of SAS/C V6.51
- strongly improved ST.spobject

V1.4 (30.04.1994) :

-----

- now WAV.spobject is really DEBUGGED and WORKS, which it actually was not and did not before ;<
- also improved the other SPOjects a lot

V1.3 (30.3.1994) :

-----

- added WAV.spobject to the distribution
- the new ST.spobject is now freely available as one of the supplied Example-SPOjects
- improved Example-Programs : e.g. removed dependency on ak\_gen0.library
- removed some garbage from the header-files
- replaced SAS/C's sprintf()-routine with own, RawDoFmt()-based one. This increases codesize a little bit, because sprintf() wasn't actually used yet. This might come in the future.
- now using Utility-Library for integer operations (activated SAS/C's UTILLIB option) : this increases 68000-codesize a little bit, but increases speed on 68020+ systems (should also decrease codesize, but does not ...)

V1.2 (27.2.1994) :

-----

- recompilation : use of SAS/C V6.5.
- Parameter-BugFix in SPO\_SetWriteSubType() : pragmas put it into D1 instead of A2
- fixed small bug in the memory management routines

V1.1 :

-----

- First Version. FREeware.

```
*****
* Revision history entries below V1.8 have been deleted          *
*                               to save diskspace !              *
*****
```

## 1.13 Credits

ST.spobject, JAM.spobject, SPO.spobject

-----  
The "powerpacker.library" V33+ is (C)opyright by Nico François.  
The Library is FREEWARE and is needed by ST.spobject, to be able  
to read and play also PP20-packed Modules.

The "xpkmaster.library" V2+ is (C)opyright by its authors  
(c/o Urban Dominik Müller).  
The Library is freely distributable and is needed several SPObjects  
to be able to read and play XPK-(also PP-)packed Modules.

ST.spobject

-----  
Upto V1.2 the main Player Source-Code in Assembler had been taken from  
the Public Domain Library "STReplay" by Oliver Wagner, which can be  
found on AmigaLib-Disk #258.  
Oliver said in the Docs to STReplay, that he did not obtain any rights  
to the Source, so I used it for my own purposes.  
Nevertheless I did not just "include" the Source, but made many - more  
or less useful and important - changes to it.

Since V1.3 the new ProTracker V2.2a Player Routine is used.  
I got it from Gerd Frank (thanks Gerd !) and modified it for my own  
purposes : Some changes had to be made to assemble it with the SAS/C  
ASM V6.55 and to make it living in the multitasking better than before.  
Finally, I had to change and add some more code to create an interface  
to the C-SourceCode of the SPObject's main routines.

Here's an excerpt from the SourceCode as far as concerning the original  
authors (no special copyright notice was found, so it is perhaps  
Public Domain) :

- \* ProTracker2.2a replay routine by Crayon/Noxious. Improved and modified
- \* by Teeme of Fist! Unlimited in 1992. Share and enjoy! :)
- \* Rewritten for Devpac (slightly..) by CJ.

JAM.spobject

-----  
The main Player Source-Code in Assembler was taken from a Public  
Domain Assembler Source-Code, which has originally been written  
by M. Gemmel and re-written by "Arcade of CADCAM".  
Some changes had to be made to assemble it with the SAS/C ASM V6.55  
and to make it living in the multitasking better than before.  
Finally, I had to change and add some more code to create an interface  
to the C-SourceCode of the SPObject's main routines.

Here's an excerpt from the SourceCode as far as concerning the original  
authors (no special copyright notice was found, so it is perhaps  
Public Domain) :

- \*
- \* JamCracker V1.0 Replay routine, written by M. Gemmel

---

```

*
[...]
```

\* This is not a demonstration source, showing how to write a song-play routine. This source is merely ment for inclusion in other sources, or as assembly module for linkage with other programs.

\* This source was modified for the MasterSeka V1.53 assembler and can be easily modified for other assemblers. (Not true, it's a lot of work ← - Arcade)

\* The song should be inserted at 'mysong' near the bottom of this source, in chip memory.

\* Modified for MasterSeka 1.53 by Arcade of CADCAM

\* Note from Arcade: I saved all of you who want to use the replayer with MasterSeka a lot of work. So leave this text here, I want the credits.

## 1.14 Requirements for the SuperPlay.library Package

Generally, you need at least an 68000 Amiga, running with OS 2.04+. Better performance results require better Software/Hardware.

\textdegree{} For usage of the IFF-based SPOjects, like 8SVX, you should take care to have iffparse.library V37+ on your bootdisk.

\textdegree{} For DataType support with OS 3.x+ you need datatypes.library V39 ← + to be present. If it is not present, datatype-support is disabled, but you might not need it, anyway.

\textdegree{} SPOjects with Packer support - like ST, JAM and SPO - usually do require the following:

Packer	needed Libraries
XPk	xpkmaster.library V2+, sppecific xpk-compressor libraries
PP20	a) xpkmaster.library V2+, powerpacker.library V33+ b) <u>only</u> powerpacker.library V33+
<None>	No special libraries are needed, but they will simply pass through unpacked data modules.

\textdegree{} MED.spobject does require med.library V1+ (see Credits)

Additionally there are some more requirements, which do not necessarily depend on the OS or the Hardware:

\textdegree{} Some SPOjects additionally require different Hardware/Software configurations, but usually this is stated within their own documentation.

Turbo versions would be installed by the Installer-Script automatically, if it makes sense (CPU auto-detection).

## 1.15 ST.spobject

© 1994-95 by Andreas R. Kleinert.

FREEWARE. All rights reserved.

Version : 2.3  
Release Date : 26.08.1995

### Description

~~~~~

ST.spobject is an external Library-Module for the superplay.library V1+.

It supports reading of ST Modules (SoundTracker V2.4+, ProTracker V2.2a, NoiseTracker and compatible Modules).

### Remarks

~~~~~

- As with other Players, the filename of the Module has to begin or end with the prefix/extension "MOD" ("MOD.module" or "module.MOD")
- ST.spobject at first tries to use xpkmaster.library. If this one is not available, it tries powerpacker.library. If none of them is available, it simply uses dos.library : So, if you read a packed module, although you did not install any decompression libraries, please note the following : After loading the data is checked, so that modules which could not be decompressed are rejected without any further notice.

### History

~~~~~

V2.3 (26.8.1995) :

- completely recompiled with SAS/C V6.55
- now saves A1 within assembler routines

V2.2 (11.12.1994) :

- extraction of samples only worked with modules, which had been XPK-packed. Now also PP20 or unpacked modules can be handled.
- fixed internal bug

\*\*\*\*\*  
\* Older revision history entries have been deleted for space reasons \*  
\*\*\*\*\*

## 1.16 MED.spobject

© 1994-95 by Andreas R. Kleinert.

FREEWARE. All rights reserved.

Version : 2.3  
Release Date : 26.08.1995

## Description

~~~~~

MED.spobject is an external Library-Module for the superplay.library V1+.

It supports reading of MED Files.

## History :

~~~~~

V2.3 (26.8.1995) :

- completely recompiled with SAS/C V6.55
- now saves A1 within assembler routines

V2.2 (11.12.1994) :

- fixed internal bug

```
*****
* Older revision history entries have been deleted for space reasons *
*****
```

## 1.17 JAM.spobject

© 1994-95 by Andreas R. Kleinert.

FREEWARE. All rights reserved.

Version : 2.2

Release Date : 26.08.1995

## Description

~~~~~

JAM.spobject is an external Library-Module for the superplay.library V1+.

It supports reading of JAM Modules (JamCracker 1.00+).

## Remarks

~~~~~

- As with other Players, the filename of the Module has to begin or end with the prefix/extension "JAM" ("JAM.module" or "module.JAM")
- JAM.spobject at first tries to use xpkmaster.library. If this one is not available, it tries powerpacker.library. If none of them is available, it simply uses dos.library : So, if you read a packed module, although you did not install any decompression libraries, please note the following : After loading the data is checked, so that modules which could not be decompressed are rejected without any further notice.

## History

~~~~~

V2.2 (26.8.1995) :

- completely recompiled with SAS/C V6.55
- now saves A1 within assembler routines

V2.1 (11.12.1994) :

- first version

## 1.18 8SVX.spobject

© 1994-95 by Andreas R. Kleinert.

FREEMWARE. All rights reserved.

Version : 2.4  
Release Date : 26.08.1995

### Description

~~~~~

8SVX.spobject is an external Library-Module for the superplay.library V1+.

It supports reading and writing of unencoded IFF-8SVX Sample-Files.

### Features / Handicaps

~~~~~

- once a sample has been loaded, it will be repeated until an explicit stop command arrives
- an arriving stop command waits (asynchronous) until the current repeat loop ends
- there's no possibility to continue a once stopped Sample

### History :

~~~~~

V2.4 (26.8.1995) :

- completely recompiled with SAS/C V6.55
- now saves A1 within assembler routines

V2.3 (11.12.1994) :

- fixed internal bug.

\*\*\*\*\*  
\* Older revision history entries have been deleted for space reasons \*  
\*\*\*\*\*

## 1.19 SPO.spobject

© 1994-95 by Andreas R. Kleinert.

FREEMWARE. All rights reserved.

---

Version : 2.3  
Release Date : 26.08.1995

#### Description

~~~~~

SPO.spobject is an external Library-Module for the superplay.library V1+.

It supports reading and writing of SPO Sample-Files.

#### File Format Information

~~~~~

Sound Samples in the "SPO Sample File Format" consist of two Files :  
The Header-File (<Sample>.spo) and the Data-File (<Sample>).  
The Header-File contains information about the Raw-Data in the  
Data-File. For loading and saving the Sample it makes no difference,  
which FileName - Header-File or Data-File - is given.  
This is managed by the SPOobject.

Because recognition of the Types of packed Files is only possible  
by reading the whole File into a buffer and then check it, we use  
two different Files for both : Sample Information and Sample-  
The Data-File CAN be packed, the Header-File MUST NOT be packed.

The SPO.object does NOT save packed data, but it is possible, to pack  
the Data File with ANY XPK-Packer you like, because the SPO.spobject  
uses the xpk.library for decoding.

This allows efficient packing and unpacking with PowerPacker and all  
other XPK-Packers.

Remember, that the original Data written by the SPO.spobject is NOT packed  
and therefore needs a lot of Diskspace. So pack it immediately after it  
has been written.

The Data saved into the <Sample> File is written almost in the same  
way like in an IFF-8SVX. The only difference is : Future versions might  
contain 12- or 16-Bit Sample Data.

#### Features / Handicaps

~~~~~

- once a sample has been loaded, it will be repeated until an explicit  
stop command arrives
- an arriving stop command waits (asynchronous) until the current repeat  
loop ends
- there's no possibility to continue a once stopped Sample

#### History :

~~~~~

V2.3 (26.8.1995) :

- completely recompiled with SAS/C V6.55
- now saves A1 within assembler routines

V2.2 (21.07.1994) :

---



- added possibility to save all Samples from all multiple-SampleList and to accept NULL parameters for the Source-SPObjct with SPO\_Write(), so that custom applications may save their self-created SampleLists by simply setting them via SPO\_SetSampleList(). The old way behaviour is still supported (as also handling of V1 SPObjcts).
- will correctly write Samples with more Bits than 8, if it is confronted with them

```
*****
* Older revision history entries have been deleted for space reasons      *
*****
```

## 1.20 VOC.spobject

© 1994-95 by Andreas R. Kleinert.

FREEWARE. All rights reserved.

Version : 2.4  
Release Date : 26.08.1995

### Description

~~~~~

VOC.spobject is an external Library-Module for the superplay.library V1+.

It supports reading and writing of PC VOC-Sample-Files (V1.10).

### Features / Handicaps

~~~~~

- Currently only VOC-Version V1.10 is supported.
- All Blocks except Block 1 and the Header are ignored, e.g. Stereo-Sample-Data.

### History :

~~~~~

V2.4 (26.8.1995) :

- completely recompiled with SAS/C V6.55
- now saves A1 within assembler routines

V2.3 (11.12.1994) :

- fixed internal bug

```
*****
* Older revision history entries have been deleted for space reasons      *
*****
```

## 1.21 WAV.spobject

© 1994-95 by Andreas R. Kleinert.

FREEWARE. All rights reserved.

Version : 2.4  
Release Date : 26.08.1995

Description

~~~~~

WAV.spobject is an external Library-Module for the superplay.library V1+.

It supports reading and writing of PC RIFF-WAVE (WAV) Sample-Files,  
playing is supported as far as the Amiga is able to (8 Bit only).

History :

~~~~~

V2.4 (26.8.1995) :

- completely recompiled with SAS/C V6.55
- now saves A1 within assembler routines

V2.3 (11.12.1994) :

- fixed internal bug

\*\*\*\*\*  
\* Older revision history entries have been deleted for space reasons \*  
\*\*\*\*\*

## 1.22 SoundDT.spobject

© 1994-95 by Andreas R. Kleinert.

FREEWARE. All rights reserved.

Version : 2.2  
Release Date : 26.08.1995

Description

~~~~~

SoundDT.spobject is an external Library-Module for the superplay.library V1+.

It supports reading and writing of Sound Datatypes.

Features / Handicaps

~~~~~

- SoundDT.spobject will cause the system crashing, if datatype.library is not already present in the system-lists (ramlib error: WILL guru). If you're using superplay.library V4.3+ this will be ensured by the library, otherwise you've to do this by yourself.
- once a sample has been loaded, it will be repeated until an explicit stop command arrives
- an arriving stop command waits (asynchronous) until the current repeat

- loop ends
- there's no possibility to continue a once stopped Sample

History :

~~~~~

V2.2 (26.8.1995) :

- completely recompiled with SAS/C V6.55
- now saves A1 within assembler routines

V2.1 (11.12.1994) :

- first version
-