

**dbplayer**

**COLLABORATORS**

	<i>TITLE :</i> dbplayer		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		August 7, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>dbplayer</b>	<b>1</b>
1.1	dbplayer.guide . . . . .	1
1.2	What's this? . . . . .	1
1.3	Requirements . . . . .	2
1.4	Author . . . . .	2
1.5	Thans must go to... . . . . .	2
1.6	To do... . . . . .	2
1.7	Known and Unknown . . . . .	3
1.8	How to use this library . . . . .	3
1.9	dbmplayer.library/DBM_StartModule . . . . .	3
1.10	dbmplayer.library/DBM_StopModule . . . . .	4
1.11	dbmplayer.library/DBM_SetPosition . . . . .	5
1.12	dbmplayer.library/DBM_SetVolume . . . . .	5
1.13	dbmplayer.library/DBM_CheckPosition . . . . .	6
1.14	dbmplayer.library/DBM_Get7Command . . . . .	6
1.15	dbmplayer.library/DBM_GetModuleAttrA . . . . .	7
1.16	The story so far... . . . . .	8

---

# Chapter 1

## dbplayer

### 1.1 dbplayer.guide

Digi Booster replayer library guide, v. 2.0

Written in 1998 by Sebastian Jedruszkiewicz

What's this?  
Introduction

Requirements  
What do you need to use this library...

Developer info  
And how to use it.

Author  
That's me ;-)

Thanks  
Do I have to explain...

Bugs  
Known and Unknown...

History  
The story so far...

Future  
To do...

### 1.2 What's this?

In short, this is shared library that contains functions

---

for playing Digi Booster Modules.

Digi Booster is copyrighted 1997 by Tomasz & Waldemar Piasta.

This piece of software is free. You dont have topay, if you want to use it. Just credit me somewhere...

### 1.3 Requirements

- AHI device version at least 4  
AHI can be found on any aminet site:  
(dev/misc/ahi.lha)  
And is copyrighted 1994-1997 by Martin Blom
- libraries that are used by AHI (for sure it is asl.library and iffparse. ←  
library)
- at least MC68020 Processor

### 1.4 Author

This library was created by:

Sebastian Jedruszkiewicz  
ul. Gdanska 10  
74-125 Chojna  
POLAND

You can reach me also via e-mail at:  
bjsebo@delta.ii.tuniv.szczecin.pl

### 1.5 Thans must go to...

- The autors of "example.library".On their sources i have based this Library.The source for example.library can be found on Aminet:  
dev/c/CLib37X.lha
- Bartlomiej Pater for first testing and some sugestions...

### 1.6 To do...

If you think that there is something missing in this ←  
library  
feel free to contact me. Snail mail and email addresses can be found  
here

Personally i think this library has everything it needs.

If you can make includes for other compilers than SASC,you are welcome.  
Mail

me

---

if you need some additional infos for creating such a stuff.

## 1.7 Known and Unknown

If you find any bugs in this library, don't hesitate to contact me  
.

## 1.8 How to use this library

This archive contains all includes needed to use dbplayer. ↔  
library.

These includes are created for SASC compiler.

In lib/ directory you will find source and linkable library for SASC compiler, handling autoopening and autoclosing dbplayer.library.

In bin/ directory you can find test program, which i am using during developing process.

List of Available functions:

```
DBM_StartModule()  
DBM_StopModule()  
DBM_SetPosition()  
DBM_SetVolume()  
DBM_CheckPosition()  
DBM_Get7Command()  
DBM_GetModuleAttrA()
```

## 1.9 dbmplayer.library/DBM\_StartModule

NAME

DBM\_StartModule -- Play a DigiBooster module.

SYNOPSIS

```
error = DBM_StartModule( module, size, AudioModeID, AudioFrequency, Flags)  
D0                A0      D0      D1                d2          d3
```

```
ULONG DBM_StartModule( APTR, ULONG, ULONG, ULONG, ULONG);
```

---

## FUNCTION

Initialize the replayer and start playing the given module.

## INPUTS

module - pointer to a DigiBooster module.  
 After player starts playing module  
     you can free memory allocated for module  
 size - size of the module in bytes  
 AudioModeID - AHI Audio mode  
 -1 can be also passed as this parameter. Then  
     AHI AslModeRequester will be prompt to select  
     AudioModeID and AudioFrequency.  
 In oldest versions this feature was available when  
 0 was given as AudioModeID. Now it is -1  
 due to incompatibility with AHI\_DEFAULT\_ID (which  
 is also defined as 0).  
     Values that was selected from this asl requester are then  
     stored in:  
     DBPlayerBase->dbpb\_AudioModeID;  
     DBPlayerBase->dbpb\_AudioFrequency;  
     And are ready for you to use.

These values are also stored there if there was specified valid  
 an AudioModeID and AudioFrequency.

AudioFrequency - Audio Frequency  
     This value is not important if 0 or -1 was passed as  
     AudioModeID.

Flags - Only one flag is supported right now:

DBF\_AUTOBOOST - turns AutoBoost On.

## RESULT

error - if the replayer started successfully, this will be 0.  
 Otherwise an error as described in <dbplayer/dbplayer.h>

## NOTES

OF course you cannot play two module at the same time.  
 If you call DBM\_StartModule(), and there is some already module  
 played error DBM\_ALREADY\_PLAYING is returned.

## SEE ALSO

DBM\_StopModule()

## 1.10 dbmplayer.library/DBM\_StopModule

## NAME

DBM\_StopModule -- stop the replayer.

## SYNOPSIS

DBM\_StopModule( )

```
void DBM_StopModule( void );
```

**FUNCTION**

If the replayer was started successfully with `DBM_StartModule()`, this function turns the replayer off and deallocates all resources.

**NOTES**

You can call this function even if there is no module played. Simply Nothing happens.

**SEE ALSO**

`DBM_StartModule()`

## 1.11 dbmplayer.library/DBM\_SetPosition

**NAME**

`DBM_SetPosition -- jump to given position in module`

**SYNOPSIS**

```
DBM_SetPosition( SongPos , PattPos)  
D0             D1
```

```
VOID DBM_SetPosition( UWORD ,UWORD );
```

**FUNCTION**

After this functions has been called, replayer will jump to Song Pattern given by "SongPos" and to position in this pattern given by "PattPos"

**NOTES**

You should call

```
DBM_StartModule()  
before  
this function.
```

**SEE ALSO**

`DBM_CheckPosition()`

## 1.12 dbmplayer.library/DBM\_SetVolume

**NAME**

`DBM_SetVolume -- Set new master volume.`

**SYNOPSIS**

---



```
DBM_SetVolume( Volume )
D0
```

```
VOID DBM_SetVolume( WORD );
```

#### FUNCTION

Set volume for all channels to given value.  
Valid values are from range 0 - 64. If given value is less than 0 volume will be set to zero. If it is greater than 64 volume will be set to 64.

#### NOTES

This function can be called even if replayer is not running.

SEE ALSO

## 1.13 dbmplayer.library/DBM\_CheckPosition

#### NAME

```
DBM_CheckPosition -- check if replayer is at given position
```

#### SYNOPSIS

```
DBM_CheckPosition( SongPos, PattPos)
D0
```

```
BOOL DBM_CheckPosition( WORD , WORD);
d0      d1
```

#### FUNCTION

Check if the replayer is at given position in song.

#### RESULT

Function returns TRUE if replayer plays pattern (SongPos) that is equal or greater than given value, and position in this pattern is greater or equal than given value(PattPos). Otherwise it returns FALSE.

#### NOTES

You should be very carefull when calling this routine.

You cannot for example do something like this:

```
while(!DBM_CheckPosition(10,3));
```

if there are only 5 patterns in module, it can simply be an infinity loop.

SEE ALSO

```
DBM_SetPosition()
```

## 1.14 dbmplayer.library/DBM\_Get7Command

## NAME

DBM\_Get7Command -- Get a byte that comes after last 7xx command.

## SYNOPSIS

```
value = DBM_Get7Command()
D0
```

```
LONG DBM_Get7Command(VOID);
```

## FUNCTION

This functions returns value that was used in last 7xx command. You can use this function just like the old E8x command from protracker.

if replayer has found 7xx command in module, xx is stored in DBPlayerBase->dbpb\_Last7Command.

This value is not changed until next 7xx command appears.

7xx command is not used by DigiBooster (I hope so...).

## INPUTS

NONE

## RESULT

value - the xx after last 7xx command.

## NOTES

Function returns LONG, but xx can be a value from 0 - 255.

## SEE ALSO

DBM\_CheckPosition()

## 1.15 dbmplayer.library/DBM\_GetModuleAttrA

## NAME

DBM\_GetModuleAttrA -- Obtain some info about actually replayed module.

## SYNOPSIS

```
DBM_GetModuleAttrA(Tags)
```

```
VOID DBM_GetModuleAttrA(struct TagItem *);
a0
```

## FUNCTION

This function will give you some info about currently played module.

Currently, one or more of this tags can be used with this function:

DBMATTR\_InstNum

Number of instruments in module.ti\_Data should  
contains pointer to ULONG,where this number will be stored.

DBMATTR\_PattNum

Number of patterns in module.ti\_Data should  
contains pointer to ULONG,where this number will be stored.

DBMATTR\_ChanNum

Number of channels that module is using.ti\_Data should  
contains pointer to ULONG,where this number will be stored.

DBMATTR\_ModName

Module name.ti\_Data should contains pointer to  
STRPTR ,where address of name string will be stored.

DBMATTR\_InstNames

Instruments names.ti\_Data should be pointer STRPTR \*.  
Function will return pointer to table,where are stored  
pointers to each instrument name.

INPUTS

Pointer to filled TagItem structure

RESULT

NOTES

It may not work,with some modules,dunno why.

SEE ALSO

## 1.16 The story so far...

History:

v2.0 - replayer was rewritten to gain some more  
speed.

- DBM\_StartModule() was rewritten (requester  
can be now opened with -1 given as  
audiomodeID. This enables possibility  
of using AHI\_DEFAULT\_ID.

v1.3 - new function DBM\_GetModuleAttrA() was added  
See docs for more info.

- function DBM\_WaitPosition was renamed to  
DBM\_CheckPositon.  
- assembler includes added

v1.2 (not publicly released)

- some speedup made in player code  
- added missing #if statement in  
includes for ppc  
- Added includes for BlitzBasic.Those files  
was made by Scott Beardwood  
(scott@online.u-net.com)  
Thanks a lot.

---

- new function DBM\_Get7Command() was added  
See docs for more details.
- Added includes for assembler.

v1.1 - not publicly released

v1.0 - first public release