

Player60A

Jarno Paananen

COLLABORATORS

	<i>TITLE :</i> Player60A		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Jarno Paananen	January 8, 2025	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Player60A	1
1.1	The Player 6.0A	1
1.2	The Player® 6.0A: Foreword	1
1.3	The Player® 6.0A: Distribution and copyrights	2
1.4	The Player® 6.0A: Requires	3
1.5	The Player® 6.0A: Features	3
1.6	The Player® 6.0A: P60con, the Converter	5
1.7	The Player® 6.0A: P60con, the Converter, Converting Protracker-modules	5
1.8	The Player® 6.0A: P60con, the Converter, Playing modules	5
1.9	The Player® 6.0A: P60con, the Converter, Converting back to Protracker	6
1.10	The Player® 6.0A: P60con, the Converter, Sample packer-selector	6
1.11	The Player® 6.0A: P60con, the Converter, Preferences	6
1.12	The Player® 6.0A: P60, the Player	7
1.13	The Player® 6.0A: Using The Player in your own programs	7
1.14	The Player® 6.0A: Using The Player in your own programs, VBlank	8
1.15	The Player® 6.0A: Using The Player in your own programs, Non-lev6	8
1.16	The Player® 6.0A: Using The Player in your own programs, CIA	9
1.17	The Player® 6.0A: Using The Player in your own programs, 68020	9
1.18	The Player® 6.0A: Using The Player in your own programs, Player6.i	10
1.19	The Player® 6.0A: The Usecode-system	10
1.20	The Player® 6.0A: Bugs?	11
1.21	The Player® 6.0A: Hellos	11
1.22	The Player® 6.0A: Contact address	11

Chapter 1

Player60A

1.1 The Player 6.0A

```
The Player ® 6.0A 600.2 / 22.01.94
Copyright © 1992-94 Jarno Paananen
Guru / Sahara Surfers
A Sahara Surfers Product 1994
```

1. Foreword
2. Distribution and copyrights
3. Requirements
4. Features
5. P60con, the Converter
6. P60, the Player
7. Using The Player in your own programs
8. The Usecode-system
9. Bugs?
10. Hellos
11. Contact address

1.2 The Player® 6.0A: Foreword

1. Foreword:

This little utility was first started just for fun about two years ago and was meant just to my own use, but when I noticed that there weren't a GOOD Protracker-compatible replay available and after I saw a few tries to fix this (ProPacker, Prorunner, etc.) I decided to give other people a chance to play their Protracker-tunes FAST!

When designing the moduleformat, I also included packing and this 6.0 packing method is the most efficient I've seen for this purpose.

Afterwards features like the two delta-formats have been introduced to keep this player on the edge of development.

I also made a converter to make it easier to use PT-modules. It uses Reqttools and Powerpacker libraries (both by Nico François) so it can also load Powerpacked modules. The converter is made as easy as possible to the user.

1.3 The Player® 6.0A: Distribution and copyrights

2. Distribution and copyrights:

This program and all other stuff coming in this package are FULLY copyrighted by Jarno Paananen / Guru of Sahara Surfers.

With the exception of Reqttools and Powerpacker libraries © by Nico François (You have made a big favour to all programmers on the Amiga!)

BUT!:

You're allowed to make additions or modifications or what so ever to fit the playroutine to your own needs. That's why the sources are here. But, please: credits for the author and a copy to myself, if possible, so I can add your new features to the future releases!

Contents of this package:

- P60Con - The converter
- P60Con.info - Da icon
- P60 - The player
- P60.info - Another icon
- P50to60 - Converter for old modules
- P50to60.info - An icon perhaps?
- Player60A.guide - Amigaguide format docs, this file
- Player60A.guide.info - More same stuff
- Player60A.doc - ASCII format docs
- Player60A.doc.info - Do we have to take all these?
- Player60A.changes - History of changes, bugfixes and everything
- 600.2.G - The main thing, Replay source
- 600.2_devpac2.asm - Same for DevPac2
- 600.2_devpac3.asm - And for DevPac3
- 600.2.bin - Binary version
- CIA_Example.G - An example of how to use the CIA-version
- Player6.i - Include file
- Vibtab - 1kb vibratortable
- Periods - Periodtable with finetunes
- Periods.nft - Same with no finetunes

The normal disclaimer:

The author cannot be held liable for the suitability or accuracy of this manual and/or the program it describes. Any damage directly or indirectly caused by the use or misuse of this manual and/or the program it describes is the sole responsibility of the user her/him self.

A few words about distribution:

This program is freeware. You can distribute it as long as ALL files are included and not more than a nominal fee for copying is asked. This program can NOT be used for commercial purposes without written permission from the author and a contribution for development costs (like Coca-Cola :-). If you have suggestions or remarks about this program, or if you find any bugs, please let me know.

THIS MEANS: IF YOU ARE GOING TO USE THIS IN COMMERCIAL SOFTWARE YOU SHOULD AT LEAST ASK ME FIRST AND I WOULD BE GLAD TO RECEIVE SOMETHING AS A COMPENSATION FOR THE TROUBLE THIS THING HAS CAUSED ME, BECAUSE MAKING THIS HASN'T BEEN JUST FUN AND SUNSHINE...

This was meant especially to software houses using this without my permission, like Team 17: From where the hell did you get that P41 you used in Super Frog?!? It was never released due to P50. I just gave it to some friends for beta testing... If you want to clear your conscience, contact me.

To everyone: If you're going to use this player in your commercial software, you should know, that the one released is never the newest version and is always somewhat cut down. If you contact (and pay...) me, you get misc stuff like sound effect engines with it, if you want. And tidy conscience...

1.4 The Player® 6.0A: Requires

3. Requirements:

- An Amiga computer (If we get some GUS-cards, PC-version is also coming)
- Reqtools.library and Powerpacker.library
- about 30 kB of memory for the converter, and if converting, $2 * (\text{number of patterns} * 1024 + 1084)$ bytes for buffers
- Works on 512kB machines, more memory recommended, at least for converting.
- Kickstart 1.2 or greater.
- Kickstart 2.0-style colors recommended (looks better)

1.5 The Player® 6.0A: Features

4. Features:

- a very efficient method to pack the pattern data and play it in realtime
 - cut-down header compared to the original one (4 bytes info, 6 bytes/sample, 8 bytes/pattern, 1 byte/position plus one byte for stopmark)
 - removes all names, unused patterns, samples, data after loops, empty data after unlooped samples, commands with no use and some smaller things
 - Delta format samples! Samples have the same quality, but packing efficiency is better with for example Imploder or Powerpacker. Look below for some stats!
 - 4-bit delta packing! Packs samples in ratio 1:2 with very little loss of quality. Try it!
-

- nearly impossible to rip without finding the call to the init routine, except if the optional 'P60A'-sign is used
- works on every Amiga, under KS1.2-3.1, from WB or CLI/Shell, detaches itself from CLI, supports WB icons, etc...
- loads files packed with Powerpacker, FImp or XPK (all references to Powerpacked files also apply to FImp and XPK files)

Should support modules with 100 patterns (At least in PT2.x from Noxious) IF patterns are packed under 64kB in PASS 1... (pointers are only words...)

Commands implemented at the moment:

- NoiseTracker 2.0 commands (0-6,A-F)
- Tremolo (7), Sample offset (9), Fine slides (E1/E2), Set finetune (E5), Pattern loop (E6), Retrigger note (E9), Fine volume slides (EA/EB), Note cut (EC), Note delay (ED), Pattern delay (EE), Invert loop (EF)
- finetune and tempo
- NOTE!:Patternbreak (D) does NOT support break to a specified line, linenumber is just ignored! This is due to my packing system.

Some statistics:

	Original	P60A	NP3.0	NP2.02	ProPacker 1.0
Tune:					
(alcoholic score)	121024		90118	93576*	104560* 108028
enigma year bkh-nhp	249550		205984	214148*	227770* 226326
knulla kuk !!!	160218		121834	128940*	138218* 147302
menuloder	23390	16992	17588	18292	20414
scrambled mind	163738		131420	132736	140614 148742

* = lacks some features (finetune, commands etc.)

Obvious?

The new Prorunner (with packing) didn't make it to this comparison, because I couldn't get it anywhere, but it's packing is nearly similar to the Player 3.0 ie. poor.

Delta vs. normal samples:

Tune: (((the war))) by Audiomonster / Melon Design

Original P60 tune: 261338

Normal with FImp: 185118
Delta with FImp: 168936
16182

Normal with PP: 185012
Delta with PP: 162812
22200

Make your own decisions...

1.6 The Player® 6.0A: P60con, the Converter

5. P60con, the Converter:

Note!: Do NOT merge hunks in this file, because it uses two SEPERATE code hunks for detaching. If you do, it crashes straight away...

Shell usage: P60con [-q][filename]

-q Be quiet. The only way to stop playing is to give CTRL-C-signal to the DETACHED piece of code. Use TaskX or similar.
filename to play. No checkings made (can't!), so be sure it IS a P60-module!

WB usage is normal. Double clicking on a modules icon or click on module, shift and click on P60con to play it.

1. Converting ProTracker-modules
2. Playing modules
3. Converting back to Protracker
4. Sample packer-selector
5. Preferences

1.7 The Player® 6.0A: P60con, the Converter, Converting Protracker-modules

1. Converting Protracker-modules:

First a normal Reqtools filerequester asks you to select one or more (supports batch-processing) modules. If you select multiple modules, the directory for the modules to be saved is asked. Then loading with the Powerpacker library. If you chose one module, the name and path for the save are asked. As the packing starts, an info requester with some bars and statistics pops up. If you are converting only one module and have sample packing on, Sample packer-selector appears. When the job is done, save and some statistics about the efficiency are shown. If you have converted only one module, also the Usecode is here.

1.8 The Player® 6.0A: P60con, the Converter, Playing modules

2. Playing modules:

First the converter asks for a module and then loads it using the Powerpacker library so it can also be packed with the Powerpacker :) DOES NOT SUPPORT FILES WITH SAMPLES IN OTHER FILE! If it finds P60 from the memory, it informs it to play the module. If not, a requester is shown with the name of the file and a stop gadget below. The play routine is the system friendly tempo one, allocates channels with priority 127, a CIA-timer and returns if it couldn't get something. Should work with every system friendly program.

1.9 The Player® 6.0A: P60con, the Converter, Converting back to Protracker

3. Converting back to Protracker:

There is in fact the good old P50-ripper with support for 4-bit deltas.

Usage is easy, just load P60A-module and save it as Protracker-module. Be sure it IS a P60A-module, otherwise I can't guarantee the result...

1.10 The Player® 6.0A: P60con, the Converter, Sample packer-selector

4. Sample packer-selector:

When you have sample packing-option on, a requester appears when you convert only one module. It is used to select which samples are to be packed with 4-bit delta packing and has the following functions:

Original -Plays the original sample. Stop with right mouse button!
Packed -Plays the sample after packing and unpacking it. This sounds same as the final packed sample. Stop with right mouse button!
Pack -Marks the sample to be packed.
Pack rest -Marks this and the rest of the samples to be packed.
Don't pack -Marks the sample NOT to be packed.
Don't pack rest -Marks this and the rest of the samples NOT to be packed.

Use of keyboard shortcuts is highly recommended!

1.11 The Player® 6.0A: P60con, the Converter, Preferences

5. Preferences:

Default loading directory: The directory converter first gives you, when you load Protracker modules. 127 chars.

Default saving directory: The directory converter first gives you, when you save ONE P60 module.

Default batch directory: The directory converter first gives you, when you save multiple P60 modules.

Prefices: These are used to mask files in filerequesters and put to the beginning of the filename, when saving. Can be up to 7 chars long.

Two files: If you want samples and rest of the tune to be saved in separate files.

P60A sign: If you want the P60A sign to be inserted in the beginning of the module.

No samples: If you're converting multiple modules with same samples, this disables the saving of

samples.

Tempo: Whether the player should use tempo or not.

Icon: If you want the icon to be saved with the module.

Delta: If you want to use delta format samples.

Sample packing: If you want to pack samples. Overrides delta settings.

1.12 The Player® 6.0A: P60, the Player

6. P60, the Player:

This program was under work (and still is...) for quite a lot time. The idea came from Simply / Parallax, due to the fact that Multiplayer can't play P50-modules without GMOD-header and that makes modules about 6kb larger... So I had to get working. Two days before Assembly'93 I rewrote this program completely in a big hurry, so please forgive me those bugs that will appear. Present program is the same, but plays P60-modules.

The interface is now quite neat, small is beautiful? Under KS3.0 those proportional gadgets are quite nice. Functions should be familiar to all who have used some player program (Multiplayer, Smartplay etc.). The only special feature is that if you try to load a Protracker module and you have P60con in memory waiting in its mainmenu, P60 gives this module to it to be packed. P60con saves it to the ram-disk and informs P60 to load it. After loading P60 deletes the module from the ram-disk.

1.13 The Player® 6.0A: Using The Player in your own programs

7. Using The Player in your own programs:

NOTE: The instructions for the binaryfile-versions expect you to have loaded the file to a label called "Player"

The versions are in the same file and these topics describe the various possible versions to compile the source. Also the binary versions have their instructions here.

Completely PC-relative!

1. Normal VBlank-version
 2. Non-lev6-version (too slow... Contact me if interested)
 3. CIA-version with tempo
 4. 68020-versions
 5. Player6.i
-

1.14 The Player® 6.0A: Using The Player in your own programs, VBlank

1. Normal VBlank version:

Options:

```
CIA = 0
system = 0
lev6 = 1
```

Exec = 1 if ExecBase is valid and 0 if not

This should be easy for all you who've used the ugly Protracker routine or one of the millions of others.

Call "P60_Init" with the address to the module in A0 and the address to the samples in A1 or if they are in the same file, clear A1. If you have a module with samples packed, put address to the sample buffer in A2. Uses all registers, so preserve them if you're going to use them afterwards.

TO CHECK IF MODULE HAS PACKED SAMPLES:

Check bit 6 (\$40) of the offset 3 from the beginning (excluding optional sign). If it is set, the module has packed samples. Size of the required buffer is then ←
at
offset 4.

To play the music, call P60_Music every frame (with NTSCs who want to play the tune in PAL-speed or vice versa, use the CIA-version).

To end the music and turn the DMAs off, call P60_End.

With the binary file, P60_Init is Player+P60_InitOffset, P60_Music is Player+P60_MusicOffset and P60_End is Player+P60_EndOffset.

The source-version can be compiled with at least Asm-One 1.09 by T.F.A. (the Best assembler around!) but it should compile with nearly any assembler.

The maximum rastertime taken is under 6 lines on a normal 68000 Amiga. Fast? Invert loop can take it beyond that, though. Everything has its costs.

NOTE: Works perfectly on AGA- and turbomachines, due to the VBR-check. Am I the lonely one to check the VBR before banging the interrupt-vectors?

1.15 The Player® 6.0A: Using The Player in your own programs, Non-lev6

2. Non-lev6 version:

Options:

```
CIA = 0
system = 0
lev6 = 0
```

This routine is basically the same as the lev6-version, but you have to call P60_Setloop about 7 raster lines after P60_Music. This system is done to

enable the use of those 7 lines. So if have a routine, which fits to these 7 lines, do it while waiting...

This is not too much slower compared to the lev6-version, but I'd still advise you to use it if possible. It makes your life a hell lot easier...

1.16 The Player® 6.0A: Using The Player in your own programs, CIA

3. CIA version with tempo:

Options:

```
CIA = 1
system = 0/1
lev6 = 1
```

Exec = 1 if ExecBase is valid and 0 if not

The first tempoplayer I know, which uses only the lev6-int and does NOT use those fucking processor loops. As fast as the normal lev6-version!

Otherwise normal init, but some info for the setting of CIA is needed in D0:

- 0 - Autodetect, if PAL/NTSC, uses ExecBase->VBlankFrequency.
- 1 - Force PAL. If you've destroyed the ExecBase, detect the timing yourself and use these force modes.
- 2 - Force NTSC

No further actions needed. Returns non-zero in D0, if it couldn't allocate something. The player starts playing right away, so do NOT call P60_Music.

To end call the normal P60_End.

To enable using of tempo, set P60_UseTempo to some non-zero value.

If system is set to 1, allocations for timer and channels are included. Use if possible.

NOTE: If you want to use this in your own program that needs synchronizing to the beam, do one of these:

1. Do your routine in VBlank-interrupt.
2. If you do NOT have to wait for a specific line, wait for VBlank bit in INTREQR (\$1e) and clear it afterwards.
3. If you have to wait for some specified line, disable Copper-interrupt from INTENA (\$9a), set Copper-interrupt bit in INTREQ (\$9c) with Copper at the specified line and wait for the bit in INTREQR (\$1e) in our own routine. Clear it afterwards.

See the examples supplied.

Otherwise this WILL steal some frames from you time to time...

1.17 The Player® 6.0A: Using The Player in your own programs, 68020

4. 68020-versions:

Options:

```
opt020 = 1
```

These are a bit optimized versions for the 020+ machines. The difference to the 000/010-versions is only a slight one and because those versions take only a half of a rasterline on a 25mhz A4000, I think these are nearly useless. But if you really want to make a 020+ program or AGA-only stuff, this is for you...

Can be compiled at least with Asm-One 1.15 by T.F.A.

Interface is similar to the 000/010-versions.

1.18 The Player® 6.0A: Using The Player in your own programs, Player6.i

5. Player6.i:

This is the include file of all the structures needed, if you're going to compile the source. Also if you are making some additions to this player (like quadrascopes, etc. (these kind of things are maybe coming, I already have quite a bunch of 'em linked to this player)) you can look for the information there. First I was going to release this only as a binary version, but after thinking a few more moments, I decided to make this include file and release the source, just to make supporting and developing of this product possible.

1.19 The Player® 6.0A: The Usecode-system

8. The Usecode-system:

This system was created for fun on a sunny afternoon. I got a crazy idea of putting a code of which features a module needs to the beginning of the source and let conditional compiling take care of the rest.

The main idea is to reduce the lenght of this massive player to the real needs of the module. So only the commands the module needs are saved to the player. Also if the module doesn't use finetune, a smaller periodtable is used. And finally if the module doesn't use vibrato or tremolo, the 1kb-long vibratotable is left out.

What you have to do is:

Take down the Usecode shown in the requester after the converter has packed your module (if only one module is being processed) and put this code to the beginning of the source into the place for it. Compile and you have a customised version for this module only. Usecode -1 leaves all features ie. compiles the full version.

for example:

```
use = $9f0f
```

1.20 The Player® 6.0A: Bugs?

9. Bugs?:

Some of you might still remember (in horror...) the previous versions of this marvelous :) player. Those 4.x-versions were all more or less beta-versions and that showed... (previous versions weren't even released, thank god..)

This version has gone through quite a hell of beta-testing, by myself, my Good friends Janne Juhola (Simply/Parallax), Steffan Schumacher (Metal/Parallax), Kari-Pekka Koljonen (Moonstone/Hippopotamus Design), WDO/Carillon (never remember your name...Mikko?) and some others.

1.21 The Player® 6.0A: Hellos

10. Hellos:

A bit more hellos than in the converter to:

To all who make those nice multiformat-player-programs (Multiplayer, Smartplay, etc.): If you want some information about this format to be included in your program, or anything else, just let me know!

To Turbo and Marley of Infect (authors of Exotic ripper): You did it! I just ask how?!? You say it's deep search and it surely is. "Unfortunately" 2.17 didn't catch these P60-modules, although the format is nearly the same. Anyway, Fab prog!

And hellos especially to:

- Janne Oksanen / Stratos of Sahara Surfers -
The demo will be a shock!

- Kari-Pekka Koljonen / Moonstone of Hippopotamus Design -
Great sendings (when coming...). Happy with this?

Should write something new...

1.22 The Player® 6.0A: Contact address

11. Contact address:

I'm improving this utility for my own use (they made me release this!), but send me some \$\$\$ (Finnish marks, please) or two disks with a return envelope WITH stamps and I'll send you the newest version (Mention which version you have, so I won't send the same version!) Also for licences for commercial use!

TO: (NO swapping!)
J.Paananen
Puskalantie 6
37120 Nokia

Finland

Or by phone: +358-31-3422147 / Jarno