Pro2Fast

Jail

COLLABORATORS

	TITLE : Pro2Fast		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Jail	March 28, 2025	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

Pro2Fast

1.1 Root

Welcome to the Amigaguide®d world of Pro2Fast:

Please select one of the following topics:

Disclaimer	Legal Information
Requirements	What you need
Introduction	Introduction to Pro2Fast
How to Install	How to install Pro2Fast
Pro2Fast - How ?	How Pro2Fast works
CLI	Tooltypes and CLI arguments
How to use	How to use Pro2Fast
Future	The Future ..
Thanks to	I want to thank my producer, and mom & dad...
History	The history of Pro2Fast v1.1
Contact	How to reach the author

1.2 disclaimer

```
Copyright information:

We won’t give you any guaranties that this program functions at all ;)
Everything you do with it is at your OWN risk!

If you or your company  want to use Pro2Fast on  a coverdisk or something
like that, we would like to know about it, and get a copy of the product.
```

Although this program is FREEWARE, you are welcome to send anything, like money, ski masks, postcards, cars, burgers (preferably McChicken!?!) Kinder Mælkesnitte etc.

1.3 requirements

Hard'n'soft-ware requirements:

You need at least kickstart v2.04 to use this program.

One general library is necessary to run the Pro2Fast with all functions. Pro2Fast cannot start without the Reqtools.library (v38+) (included).

If you want Pro2Fast to unpack your modules, you should have the Xpk package installed. This can be downloaded from aminet.

1.4 introduction

Introduction:

The idea to Pro2Fast© originally erupted from a sick brain belonging to Jakob Langgaard (me :) . Even if you aren't a cand.amiga you know the most popular/widespread amiga format for music - Protracker modules. And you don't have to be a cand.pc (yik!) to know that most pc music formats offers upto 32 soundchannels at the same time, while the protracker format only gives you 4 (yes four!) sound channels.

As the first step in the direction of more sound channels for the Amiga, Jarno Paananen wrote PS3M which astounded us pretty much :) .. It was able to play all those nifty pc music formats on your very own Amiga.

Upto now, no one has been able to develop a music program that would allow anyone to compose those xx-channel modules on the Amiga.
Hopefully the forthcoming SoundStudioPro from the makers of OctaMED Pro will cure this problem. But until then we have developed this little utility to help YOU!

With Pro2Fast© it is possible to join upto 8 protracker modules/songs into one Fast/Taketracker module. That is, you can make your own 32 channel modules.

Features:

- upto 32 channels Taketracker, or 8 channels Fasttracker I modules.

- Can use both Protracker modules AND songs.

- automatic xpk unpacking.

- Each module can be choosed to either loop or stop at the end.

- For each module you can choose which channels to place in the left or right speaker, or to leave out completely.

1.5 installation

Installation:

You have to install Pro2Fast manually, either by shell, wb or any filemanager like dopus. Here is a short description of all the files in the package:

Pro2Fast - The main program (i.e. Pro2Fast WON'T work without it ;)

Pro2Fast.guide - This documentation (perhaps you've already guessed that)

Libs/reqtools.library - Needed library for Pro2Fast. It makes all the nice (file)requesters

1.6 works

What Pro2Fast does:

The short version: The source modules is scanned and the one with the greatest song length is used to set the songlength in the destination module.
All samples used in module 1 is used in the dest module, then all samples NOT used in module 1 but used in module 2 is used and so on ..

The pattern data from all the source modules is placed along each other. If you have chosed a module to loop at end, it will only do so if its song length is smaller than the largest song length. E.g. if one of the modules is a drumpattern, you can choose is to loop all the way trough the whole destination module.

The Long Explanation:

1. All the specified modules is scanned to find the longest module. This one determines the length of the destination module.

2. The samples which is used in the destination module is determined using the following pattern:

   - First all the samples used in module number 1 (max. 31) is placed in the destination module.
   - Then the second module is searched to see if some of the 31 samples not used by module 1 is used by module number 2.
     If that is the case, these samples is placed in the destination module too.
   - This procedure continues until there aren't more modules or until all samples are used.

```
  3. Then the pattern data from all the source modules is placed along each
     other in the destination module.
     How the  patterns is  placed depends on  which channels you  have told
     Pro2fast to use from the  source modules. Perhaps one module only uses
     2 channels, both which  you want to place in the  left channel  in the
     destination  module. To do  this  you  have  to  set the  cyclegadgets
     corresponding to the channels (1 and 2), to L (Left channel).
     To leave the other two channels out, set their  cycle to O (for Off)

  4. At last the  destination module is  saved. Depending on  the number of
     channels used, it  can be either a Fasttracker I (max 8 channels) or a
     Taketracker (max 32 channels) module.
```

1.7 cli

```
  CLI Arguments:

The CLI argument  line of Pro2Fast looks very  akward and  messy. Hopefully
the explanation below will help clear this up.

C=CALCULATE/S,
M1=MODULE1/K,C11/K,C12/K,C13/K,C14/K,L1=LOOP1/S,
M2=MODULE2/K,C21/K,C22/K,C23/K,C24/K,L2=LOOP2/S,
M3=MODULE3/K,C31/K,C32/K,C33/K,C34/K,L3=LOOP3/S,
M4=MODULE4/K,C41/K,C42/K,C43/K,C44/K,L4=LOOP4/S,
M5=MODULE5/K,C51/K,C52/K,C53/K,C54/K,L5=LOOP5/S,
M6=MODULE6/K,C61/K,C62/K,C63/K,C64/K,L6=LOOP6/S,
M7=MODULE7/K,C71/K,C72/K,C73/K,C74/K,L7=LOOP7/S,
M8=MODULE8/K,C81/K,C82/K,C83/K,C84/K,L8=LOOP8/S,
DM=DESTMODULE/K:

C=CALCULATE : This is a switch which tells  Pro2Fast only to calculate some
              info of the  destination module. Please note  that this means
              that the destination module won't be saved.
              An example output could be:

              8 111428 20 25

              8 - indicates the number of used channels.

              111428 - is the size of the destination module.

              20 - the number of different patterns used.

              25 - the songlength.


M1=MODULE1/K - M8=MODULE3 : This is the filename,including full path, to the
                            module number 1-8.

Cxy/K : All the Cxy arguments  is ordered this way: x is the  number of the
        module (1-8) and y is the number of the single channel (1-4).
        The keyword after Cxy can be one of three choices:

        r - place it in the right channel
        l - place it in the left channel
        o - turn off the channel

        Example:
        --------

        C11=r c12=l c13=l c14=r

        c11=r : this places channel 1 from module 1 in the right channel in
                the destination module.

        c12=l : this places  channel 2 from module 1 in the left channel in
                the destination module.

        c23=l : this places  channel 3 from module 2 in the left channel in
                the destination module.

        c84=o : this turns off  channel 4 from  module 8 in the destination
                module.

        .. and so on!


Lz=LOOPz/S : This toggles wether you want  the z'th module  to loop when it
             ends. If  you don't  specify  this  switch argument, the  z'th
             module will simply stop when it ends.

             Example:
             --------

             LOOP1 : this will cause module number 1 to loop at end.


DM=DESTMODULE/K : This is the filename (including path) of  the destination
                  module.
```

1.8 usage

```
  How to use Pro2Fast:

  The usage of Pro2Fast is  fairly simple (?) and pretty straigth forward -
  as long as you know what you're doing ;)

  ___________________________________________________________
 |                                                           |
 |  \/ This is module 1          o @|L @|R @|R @|L @|End |
 |                                                           |
 |                               o @|L @|R @|R @|L @|End |
 |                                                           |
 |                               o @|L @|R @|R @|L @|End |
 |                                                           |
 |                               o @|L @|R @|R @|L @|End |
 |                                                           |
 |                               o @|L @|R @|R @|L @|End |
 |                                                           |
 |                               o @|L @|R @|R @|L @|End |
 |                                                           |
 |                               o @|L @|R @|R @|L @|End |
 |                                                           |
 |                               o @|L @|R @|R @|L @|End |
 |                                                           |
 |  Destination:                                          o |
 |___________________________________________________________|
 |        Convert                    Calculate              |
  -----------------------------------------------------------
```

The menus Calculate, Convert and Quit correspond to the gadgets with the same names (Quit is done by pressing the close gadget in the top left corner).
The About menu item pops up the about picture which also appears every time Pro2fast starts.

1.9 checkmarks

To include a module you have to check the checkmark to the left of the module name you want to use.

1.10 modules

This is the modules you want Pro2fast to include in your destination module.

For more info see: How Pro2Fast works!

1.11 disk

This opens a filerequester where you can select the name of the module to include.

1.12 channels

Every cycle tells Pro2Fast where to place the channel number 1-4 from the source module. They can be placed in the (L)eft or the (R)ight channel of the destination module. Alternatively you can choose that some channels from the source modules should be left (O)ut.

For more info see: How Pro2Fast works!

1.13 loop

This determines wether you want the specific module to start over when it ends or not.

For more info see: How Pro2Fast works!

1.14 destination

This is the path+filename under which you want Pro2Fast to save the destination module.

For more info see: How Pro2Fast works!

1.15 disk2

This opens a filerequester where you can select the name of the module to save. (The destination module)

1.16 convert

This joins/compiles the source modules and saves the outcome as the destination module.

For more info see: How Pro2Fast works!

1.17 calculate

This calculates some information on the destination module. The information is printed in the info bar at the top of the window.

1.18 future

Future:

The future belongs to you my friend .. and may the force be with you :)

Since this Pro2Fast is freeware we have decided to release the source code to the public. This gives you the possibility to enhance this program yourself. We do not plan to continue to develop Pro2Fast, but we still have some ideas for future improvements.

- Pattern optimizer (removes redundant patterns)

- Sample name editor

- Automatic export to a music player (APlayer), via ARexx

- and perhaps more .. this is all up to you!

If you decide to make some enhancements to Pro2Fast, and release it as a new version. We would be thankful to recieve a copy of the program. (perhaps by e-mail)

Please remember: Pro2Fast was originally developed by Thomas Neumann, so be kind to remember to credit him.

1.19 thanks

Thanks to:

Nico François for the fabulous Reqtools library. + Magnus Holmgren for continuing the development of reqtools.

Urban Dominik Mueller & Bryan Ford for the even more fabulous Xpk master library.

Michael Sinz for Enforcer

Commodore for Mungwall and Segtracker

Amiga for being the best computer EVER (I.... outside!)

And to Amiga Technologies/Escom for showing a lot of interest in keeping developing the Amiga (We are looking forward to the PPC :)

1.20 history

```
v1.0 - Initial release

v1.1 - Fixed a simple bug in the sample start calculation (used the wrong
       variable (oops! :) This caused the destination module to crash a bit
       (a lot ;)
```

1.21 contact

Send bugreports and suggestions to

Coder:

Thomas Neumann (Tax)
Kongensgade 78
3550 Slangerup

```
Denmark

Send anything to

Designer, Alphatester & Guide writer:
-------------------------------------

Jakob Langgaard (Jail)
Krebsen 101
3650 Ølstykke
Denmark

E-mail: jail@diku.dk
```