PICSim

Ioannes Petroglou

COLLABORATORS

	TITLE : PICSim		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Ioannes Petroglou	February 12, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

PICSim

1.1 PICSim Manual

```
                                PICSim
              Version 1.3

What’s new?
                            A Pic Simulator
          for Amiga computers

 © Copyright 1996-1997 Ioannes Petroglou

            18.05.1997
```

1.2 What’s new?

third release:

-listview font select
-cpu part about 30% faster
-PICAsm loader
-Watchdog support
-source overwork

this is the fourth release:

-added symbol manager
-added watched registers window
-added conditionl register break
-...?

1.3 Overview

PICSim is a versatile software simulator for Microchip PIC16C5X and 16C84 series microcontrollers. It allows the user to simulate their code on Amiga to verify its proper operation.

PICSim uses the listfile or the objectfile that is produced by various assemblers.

PICSim requires an Amiga or compatible computer (OS3.0 or above) with at least some bytes of free memory to run.

1.4 Registration

PICSim is shareware. To ease your decision whether to pay, the unregistered version has one limitation:

- PIC size is limited to 64 bytes

The shareware fee is DM 40,-. Registered users will receive a personalized Program with the missing function and support for the assembler.

How do you become a registered user?

The simple way is to fill in the file "OrderForm", to print it out, sign it (in this order, if possible), put it in an envelope together with the registration fee and send it to me . I won't accept order forms which are not signed.

PLEASE DON'T SEND CHEQUES! I would prefer if you send me an International Postal Money Order.

snail mail:

Dirk Düsterberg
Jahnstr. 9
31860 Emmerthal
Germany

or Bank Draft:

Name	:	Joannis Petroglu
Institut	:	STADTSPARKASSE BAD PYRMONT
BLZ	:	254 513 45
Konto-Nr.	:	10346708

I will send you the program per EMAIL, if you didn't want this or if there is no EMAIL adress I will sent the program on disk to your postal address. The shipment on disk may take 4..5 weeks, please be patient! Shipping per Disk costs 10 DM extra.

1.5 Support

The official PICSim homepage in the WWW has always the latest version and other information related to PICSim:

http://linux.rz.fh-hannover.de/~duesterb

1.6 Installation

just copy the files in a directory on your harddisk

1.7 notes

no notes

1.8 Using

PICSim determines the device type being simulated (e.g., PIC16C54, 55, 56, 57, 84) by a variety of methods. The recommended method is to include the "device" directive in source code (see assembler instructions for details). The simulator locates the "device" directive in the list file and sets the device type accordingly.

Another method to select the device type is to select it in the pulldown menu.

It my be necessary to load the listfile again if you switch from a 5x type to the 16c84.

You may scroll through the source code displayed in the listview using the up and down arrow keys, you can jump with the shift key. The Current position the programcounter address, and breakpoints are marked different. When you press the up or down arrow keys, you can scroll up or down through the code. The programcounter is marked white when you scroll off of the current program counter line. To set a breakpoint in your code, you may scroll through the code until the desired line is highlighted at the center, then press the space bar. This will set a breakpoint at that line and denote this by highlighting it in blue. There are no limits to the number of breakpoints that you can set.

To alter a register during simulation, you can move the mouse to the register. Then press the left mouse button to increment the register contents or the right mouse button to decrement it. The upper and lower nibbles of a register can be incremented or decremented separately for those that are displayed in hex. For those that are displayed in binary, you may alter each bit

separately. Some of the registers are not allowed to change. You may not change the indirect address register (f0). The indirect address is not physically implemented in the processor and altering it would have no effect. Also, if the selected device is a PIC54 or 55 you may not change the upper two bits of the program counter, since these devices do not implement these bits. For the PIC56, you are not allowed to alter the upper bit of the program counter.
You can also press the shift key and click on the desired register, now you can give in the value with the keyboard.

The menus

Project

Controls

Device

Reset

Tools

Preferences
SPACE.

Toggle the breakpoint at the highlighted line. To insert a breakpoint at a specific line, use the cursor keys to scroll the program display up or down until the desired line is displayed in the center highlighted bar, then press the space bar. The line will then turn blue to indicate that the breakpoint is set. Pressing space again on the same line will clear the breakpoint.

BACK STEP. Pressing the left arrow key steps backward one line in your program. The entire state of the microcontroller is stored in a 1000 element history buffer to allow the user to step back through the code up to 1000 steps. The right arrow key steps forward in the history buffer.
You can jump to end of history with "e" and to start with the "s" button.

HELP. On pressing the HELP key the listview jumps to the actully Program counter.

1.9 System requirements

Requirements:

- The Amiga must have at least a 68020 processor. PICSim will run on every Amiga 1200/3000/4000, but not on stock Amiga 500/2000.
- 2MB of free memory are necessary to use over 1000 history buffers
- PICSim runs under AmigaOS 3.0 and higher

1.10 Questions and answers

Question:
Where is the assembler ?

Answer:

Assembler
Question:
What to do when PICSim ignores the Pic type ?

Answer:
Choose the correct Pic type in the device menu and load the listfile again.

1.11 Assembler

PICAsm has an own guide, please look there.

1.12 Copyright

The programs "PICSim", "PicProgger", "PicASM" may be freely distributed as long as they remain unchanged (archiving and packing are allowed).

No profit must be made by distributing PICSim, especially the price of a disk containing PICSim may not exceed US$ 5,- (or equivalent amounts in other currencies).

Please feel free to distribute PICSim over bulletin board systems and networks and ↩
as
part of shareware/freeware CD-ROMs. All rights for commercial use remain at the

author
.

The Program that registered users will receive, must only be installed one one computer and in no case passed on to others. Offences will result in penal prosecution by me. With your signature on the order form, you accept these conditions.

The program is presented to the users as it is, without any warranty of any kind, be it expressed or implicit. Anyone using this program agrees to incur the risk of using it for himself. In no way can the author be made responsible for any damage directly or indirectly caused by the use or misuse of the program.

The user interface of the program was designed with GadToolsBox
© Copyright 1991-1993 Jaba Development.

Parts of the program are © Copyright 1992-1993 Jaba Development.

"Amiga" and "Commodore" are registered trademarks.

Names of other hardware and software items mentioned in this manual and in program texts are in most cases registered trade marks of the respective companies and not marked as such. So the lack of such a note may not be used as an indication that these names are free.

1.13 Bug reports

If you find a bug or a misfeature in PICSim, or have an idea how to make some things better, then please drop me a note so I'll be able to improve PICSim in the future. My address can be found
here
.

Important for a bug report is the following information:

- Version of PICSim
- Used AmigaOS version (e.g. 3.1, 3.0 etc.)
- Used Listfile, objectfile
- Installed hardware, if of interest for the problem
- Information about installed startup programs on the Amiga
- Detailed description what program produces the bug and how it can be reproduced

But first please look
here

```
                 if there's a solution to your problem.
```

1.14 Credits

- Stephen Marsden with his EPIC1.2.lha where the IBMKEY25 example came from

1.15 The author

```
                There was no Pic Simulator on Amiga.
So I had to do it. :-)

My address is:



Mail:
  petroglo@unixserv.rz.fh-hannover.de



WWW:
  http://linux.rz.fh-hannover.de/~duesterb/

Questions, criticism, suggestions and
                bug reports
                 are always welcome.
```

1.16 History

```
04.02.97  - first release,  v1.0
15.02.97  - second release, v1.1
06.05.97  - third release,  v1.2
```

1.17 Project

```
LOAD  >>  PICAsm    this loads a PICAsm listfile

    Parallax  this loads a Parallax listfile

    MPASM   this loads a Microchip listfile

    INHX8M    this loads and disassembles a Intelhex file
```

```
Save as Source      Save the Hexobjectfile as source code
        Labels are the word addresses from Pic, it works
        only on Pics with one Page correctly, because every
        jump or call determines in the actually Page.
```

```
About       Program Information
```

```
Quit        Quit the Program. A pop-up prompt will allow you to confirm
        the quit command.
```

1.18 Controls

```
Run       F1. This causes the simulator to execute the code
        until a breakpoint is reached or any key is pressed.
        The screen is not updated until execution is stopped.
        The execution starts at the marked programcounter line.
```

```
Go        Pressing F2 will start the simulator running, and will
        not stop until a key is pressed or a breakpoint is reached.
        The screen is updated and the changes are highlighted after
        each line is executed. The execution starts at the marked
        programcounter line.
```

```
Run to PC+1     F3. execute the code without screen update until the next
        programcounter is reached. Usefull with loops and calls.
```

```
Single Step     This causes one line to be executed.  The changes in the registers
        are highlighted after the line is executed. If you scrolled through
        the code the current list adress is taken for the programcounter.
        Pressing F4 causes executing single lines repeatly.
```

1.19 Device

```
16C54       18pin Pic Type, 2 Ports (PortA=4bit, PortB=8bit), 512 * 12bit


16C55       28pin Pic Type, 3 Ports (PortA=4bit, PortB and PortC=8bit), 512 *12bit


16C56       18pin Pic Type, 2 Ports (PortA=4bit, PortB=8bit), 1024 * 12bit


16C57       28pin Pic Type, 3 Ports (PortA=4bit, PortB and PortC=8bit), 2048 *12 ↩
   bit


16C84       18pin Pic Type, 2 Ports (PortA=4bit, PortB=8bit), 1024 * 14bit
        Program and Data EEProm
```

1.20 Reset

```
Pic       This simulates a hardware reset.


Time        This resets the Time counter


Cycles        This resets the cycle counter


File Register       This resets the file register


XTal        setting up the XTal


Watchdog        resets the Watchdogtimer


Breakpoints       clear all set breakpoints
```

1.21 Tools

Convert converts ascii, binary, hex and dezimal

Stack Window shows on 16C84 all 8 Stacks

EEProm Window shows the EEProm Window on 16C84

Symbol Manager manages the symbols
(conditional break, register watching)

Watch Window shows selected symbols with their values

Random FR fills the fileregister with random values

1.22 Preferences

Screen >> Select Mode requester to choose screen mode

Use Public uses the public screen

Palette Palette Preferences

Select Font selects Font for the Listview

Settings switch and adjust several functions

Save Saves the Preferences, PicType and Window positions