TilEm2 USER MANUAL

DUPONCHELLE Thibault - MOODY Benjamin

March 21, 2012

Contents

Chapter 1

Introduction

1.1 What's TilEm2?

TilEm2 is a TI calculator emulator. It emulates all the Z80 calculators (73, 76.fr, 81, 82, 82stats, 82stats.fr, 83, 83+, 83+ SE, 84+, 84+ SE, 85, and 86) and all known ROM/OS versions.
TilEm2 is completely free, and designed for Linux (but available for Windows). We put a lot of work in this software to offer to the community the best possible product.
TilEm2 also provides a full featured debugger with disassembler, breakpoints, memory view and more.

1.2 Some history

Some of you probably already know TilEm because a first version was released around 2000/2001 by Julien Solignac (the maintained by Benjamin Moody since 2004).
This first version was working fine but there were some issues, skins were too small and bad resolution and a lot of feature were missing.
Anyway, this software was pretty good (especially because the core emulation was very good).
We decided to rewrite this emulator from scratch, keeping the philosophy of TilEm but improving all the rest.
A new core has been developped by Benjamin Moody, and I started to work on the GTK user interface (later he helped me for this task).

We are proud to release our work for beta testing !

1.3 Features

TilEm2 has basically all the TilEm old features plus a lot of new things :

- Linking : Send and receive var (use libticalcs2).
- Screenshot.
- Animated screenshot.
- Grayscale.
- Save states.
- Use TiEmu skin file format (easy to do your own skin).
- And more...

1.4 Skins

You can use TilEm2 without skin (just uncheck the "Use skin" checkbox into the Preferences menu) but skins are more user friendly :)
We have made some officials and free to use skins (thank you to our contributors).
You can do your own skins using skinedit. If you want, you can send us the skin file, maybe it could become "official".

What do you need to do your own skin?
Just take a picture of your calc using your smartphone by example.
Scale it keeping proportion to have around 900 pixels high.
Then start skinedit, create a new skin, open the picture and set the key positions.
It takes less than 20/30 minutes I think.
Then you can test it with TilEm2.That's all!
Here are the current skins available by default :

Figure 1.1: The skins

Chapter 2

Installation

2.1 Generalities

Before installing TilEm2, you should know that no ROM is included in this software.
In order to use TilEm2, you must use your own rom (use TILP to get it).
TilEm2 provides an installer msi for windows and script autoconf for Linux.
There's not a lot of dependancies so you should really have no problem to install it.

2.2 Dependancies

TilEm2 uses the following libraries :

- GTK+ 2.6 or higher (but 3.x not supported yet).
- libticalcs2.

You can find libticalcs2 on ticalc (from Romain Lievins).

2.3 Install from sources

Download the sources of TilEm2 on sourceforge.net.
Or eventually :
Dowload the source from the trunk like this :

```
svn co https://tilem.svn.sourceforge.net/svnroot/tilem\newline\newline
```

:

Then install gtk+ (e.g. for debian : sudo apt-get install libgtk2.0-dev).
Then install libticalc2.

After that, simply use the configure script and the well know Linux install :

```
./configure
```

:

If you have no errors, so dependancies are checked and it's ok.
The Makefile have been generated so type:

```
make
```

:

Then to copy the icons, configuration files and tilem2 binary type:

```
sudo make install
```

:

Usually, icons will be copied into /usr/share/tilem2/
Keybindings and configuration file will be installed into $HOME/.config/tilem2

Then you can launch TilEm2 with the command :

```
tilem2 -r /path/to/rom
```

:

Or simply :

```
tilem2
```

:

2.4 First use

If you do not specify explicitely a rom on the command line, the first launch will ask you which rom you want to use.
TilEm2 will open a file chooser dialog.

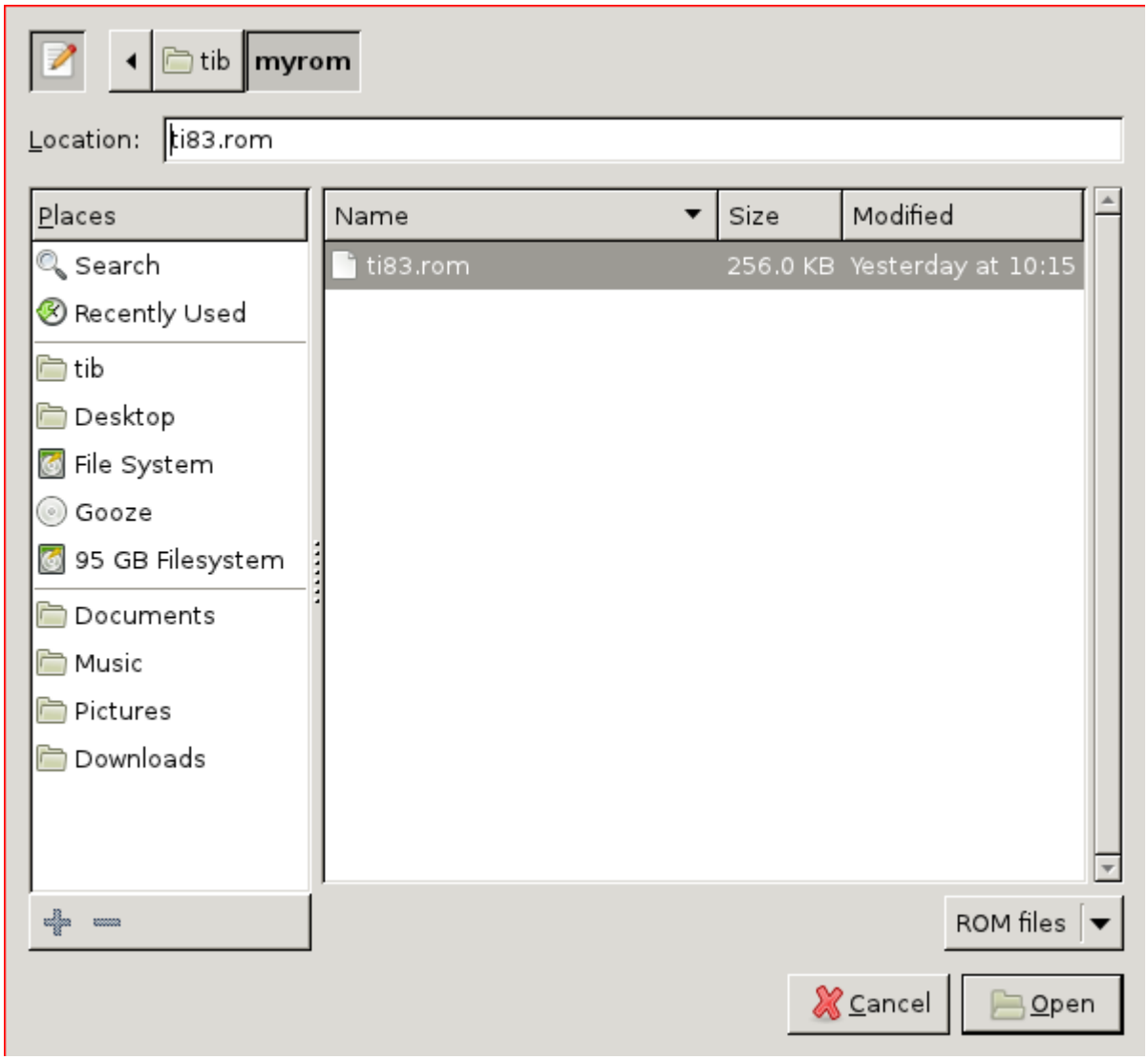

Figure 2.1: Starting TilEm2 for the first time

As soon you press "Ok", TilEm2 will try to guess the model of this rom (and check if it's a correct rom).
When TilEm2 has a doubt, he will ask you for the model but will display only the possible candidates (not all the z80 calc).
Anyway, if you launch a rom, you usually know what's model it is because as I've already said : you should have the calculator of the rom you're trying to emulate...

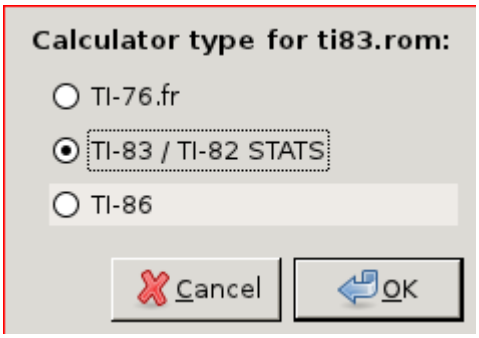

Figure 2.2: Choose the model

After that, TilEm2 start (but calc is off you need to press on).

Figure 2.3: It works !

The last used rom is automatically used for the next launch of TilEm2. You can save the current state of the calculator by using "Save Calculator". If you cancel the rom chooser dialog, TilEm2 automatically shutdown.

Chapter 3

Utilisation

3.1 Menu

As TilEm1, the menu is a popup menu (right click).
All you want to do need to use this menu.

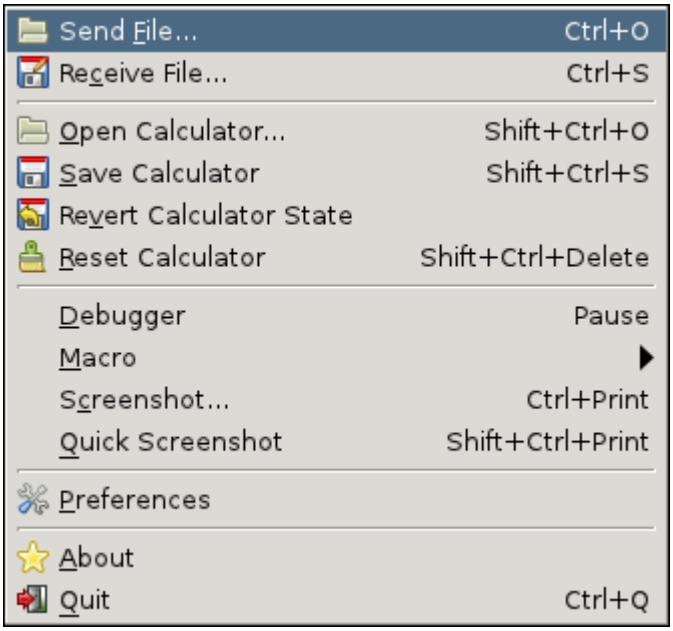

Figure 3.1: The right click menu

As you can see, there's all you need, no more, no less:

- Send File... : Load a file from your computer to TilEm2.
- Receive File : Launch a menu where you can store a variable from TilEm2 to your computer.
- Open Calculator... : Load a ROM.
- Save Calculator : Save the current state of the calculator (in a separate sav file)
- Revert Calculator : Revert the state of the calculator.
- Reset Calculator : Reset the calc of course.
- Debugger : Open the debugger window.
- Macro : Record, play, open or save a macro (a kind of script to do some actions automatically).
- Screenshot : Open the screenshot menu (static and animated screenshot).
- Quick Screenshot : Grab a screenshot and save it without prompting (that's why it's "quick").
- Preferences : Open the preference window.
- About : Open the about dialog (informations on the authors and more)
- Quit : Close TilEm2 properly

3.2 Send File...

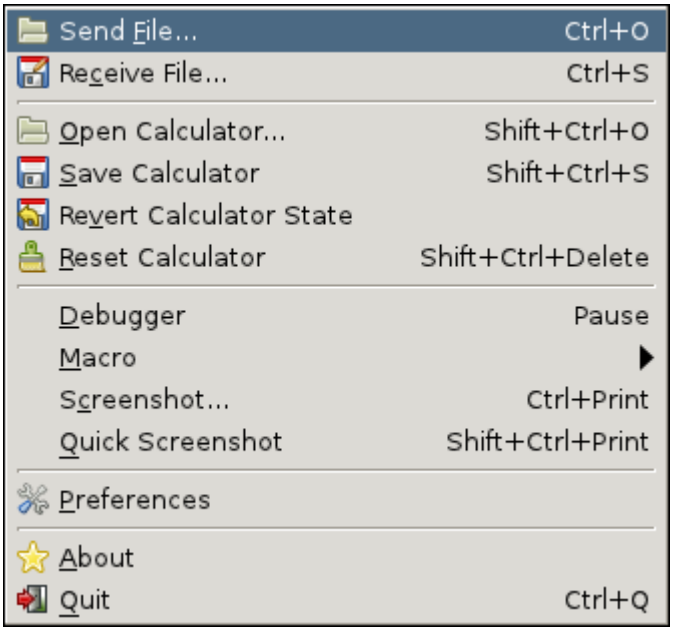

Figure 3.2: The "Send File..." menu entry

This is one very important feature, because emulators are usually used to try some programs before really transferring it to real calc.

When you click on this menu entry, a file chooser dialog is opened and let you choose a file.

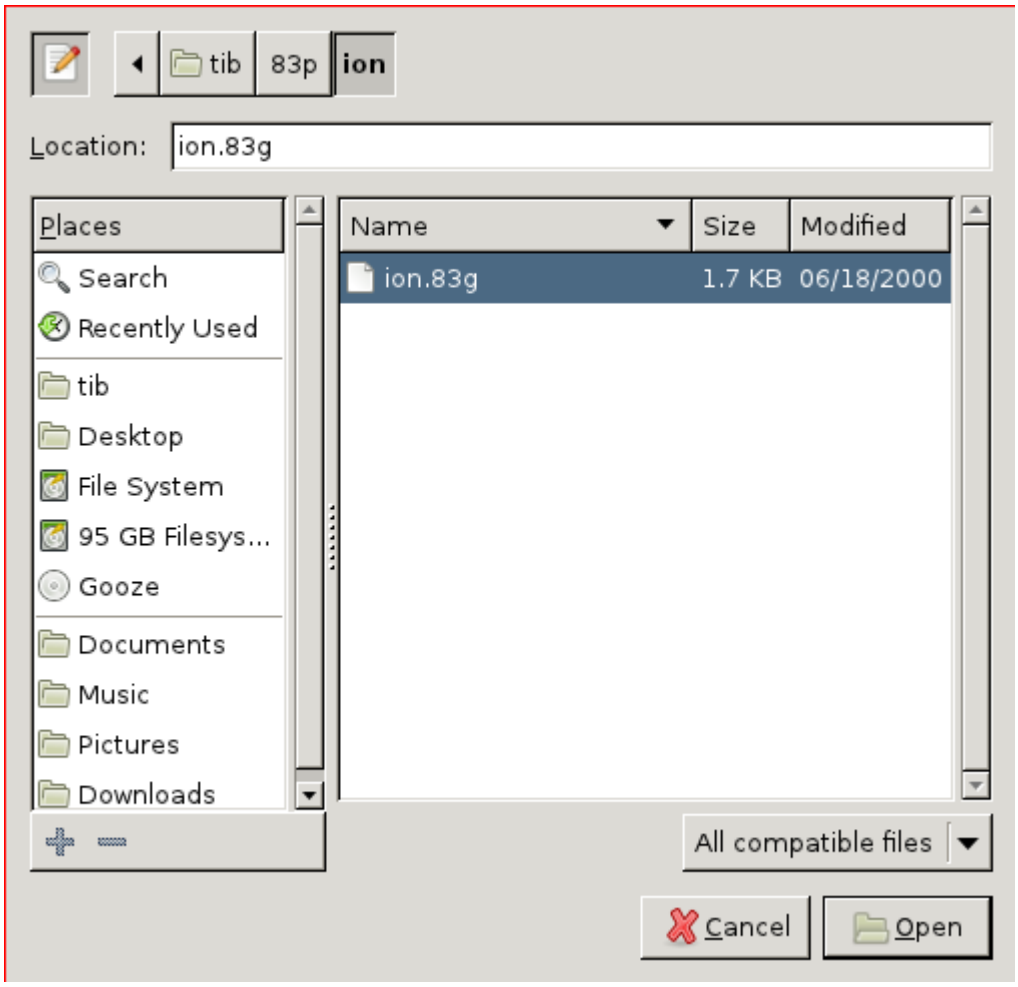

Figure 3.3: The "Send File..." file chooser dialog

A lot of people don't know which file extension is associated with the emulated model...
To help them, some patterns are used to do the selection.
When you let "All compatible files", TilEm2 do the job for you, but you can choose "All files" if you know what you're doing (a file with an incorrect extension by example).

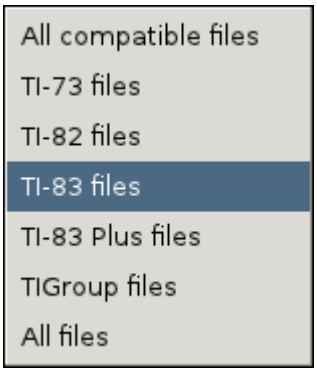

Figure 3.4: The "Send File..." patterns

It could take some time to load a variable so a current progress bar is printed while loading to know what's happening.

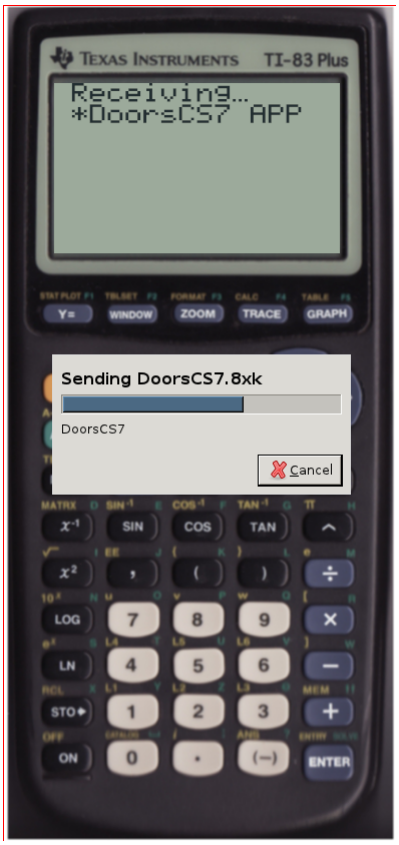

Figure 3.5: The "Senf File..." progress bar update

3.3 Receive File...

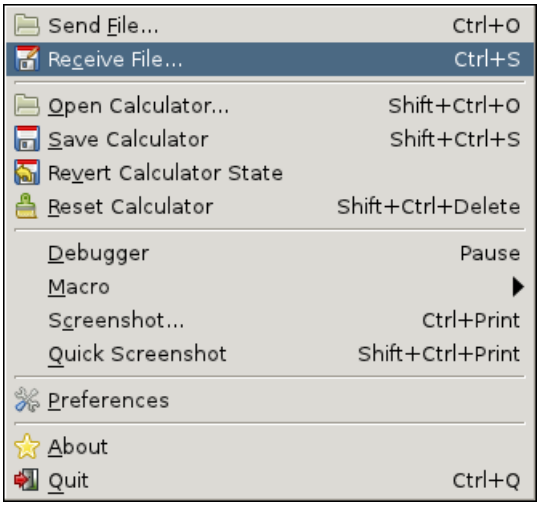

Figure 3.6: The "Receive File..." menu entry

When you click on "Receive File..." menu entry, TilEm2 firstly get the vars then prints it into a listview.

Figure 3.7: The "Receive File..." get the variables

After the first launch, refresh is made only on request !
If you click "Receive File..." then close the window, then create a program and click "Receive File..." you will not see your program.
The variable list let you choose the stuff you want to backup.

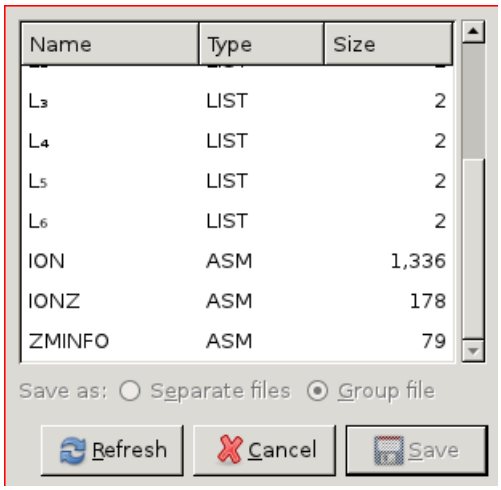

Figure 3.8: The "Receive File..." window

If more than one variable is selected, you can choose between two modes of backup : "Separate files" or "Group file".
If you choose separate, each file is saved as if you have saved one by one.

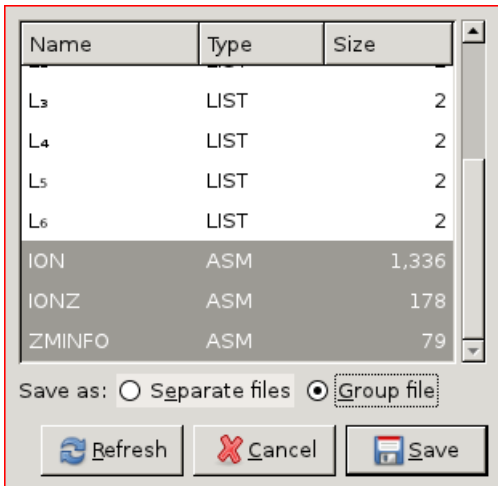

Figure 3.9: The two modes of backup (multiple files only)

If you save grouped, a group file will be created on disk.
When you finally click on "Save" button, a file save dialog is opened.
Choose a directory and a name and click "Save".

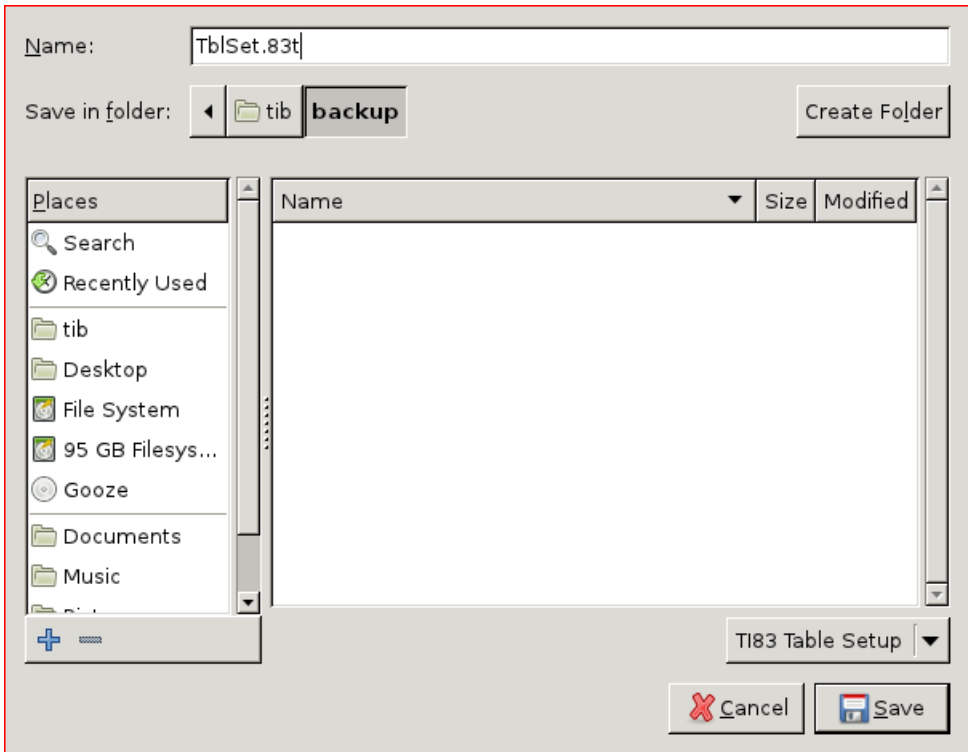

Figure 3.10: The "Receive File..." file save

3.4 Open Calculator...

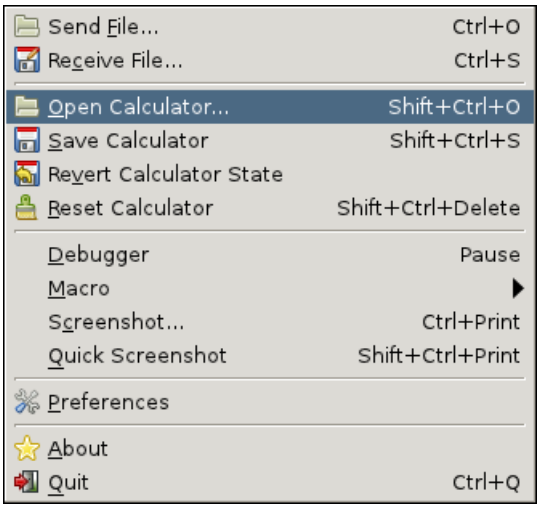

Figure 3.11: The "Open Calculator..." menu entry

When you click on "Open Calculator...", a file chooser dialog pop up and let you choose a rom to load.
So in fact even if you already emulates a calculator, you can switch to another just by opening a new rom file.
Another way to do that is to quit TilEm2 and restart it using another rom file (option -r).

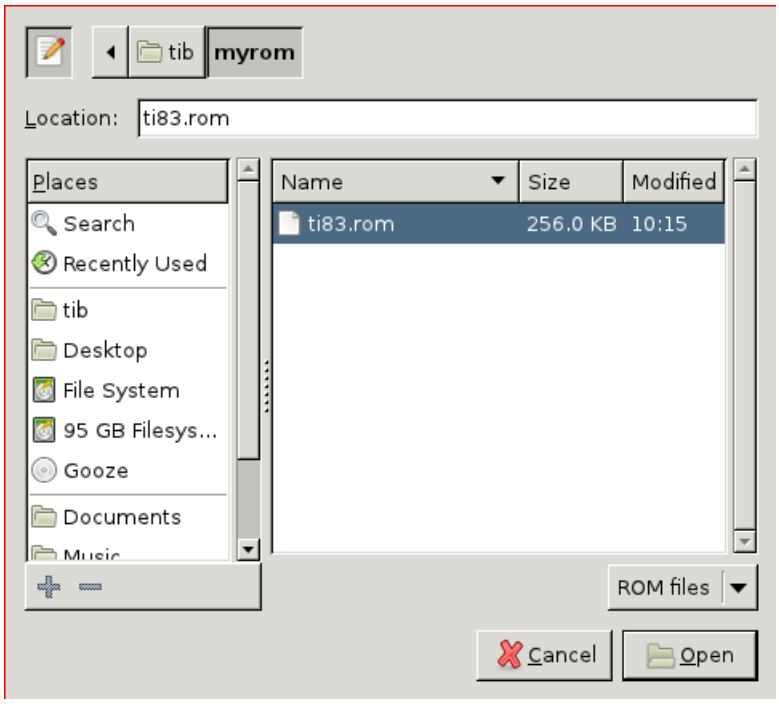

Figure 3.12: The "Open Calculator..." file chooser

Rom files usually finish by .rom as extension but you can use "All files" pattern if you have a rom with a odd extension.

Figure 3.13: The file chooser patterns

3.5 Save Calculator...

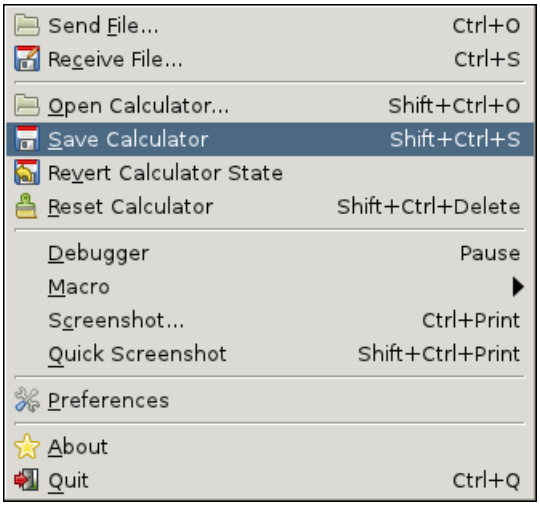

Figure 3.14: The "Save Calculator" menu entry

This option just save the current state of the calculator in a .sav file.
The file is created in the same directory as the rom file and with the same name.

3.6 Revert Calculator State

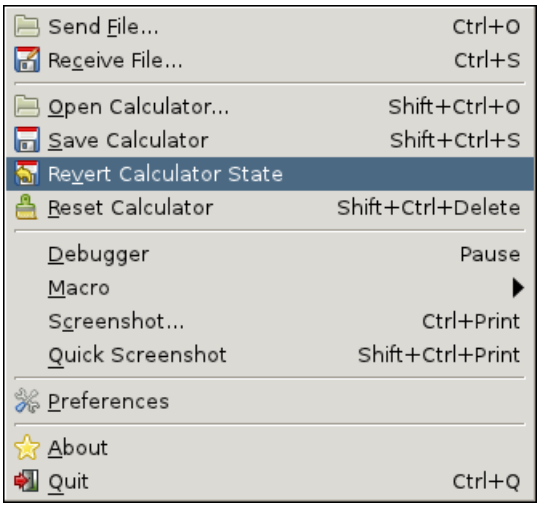

Figure 3.15: The "Revert Calculator State" menu entry

No surprise, this option just revert the calculator state (if possible).

3.7 Reset Calculator

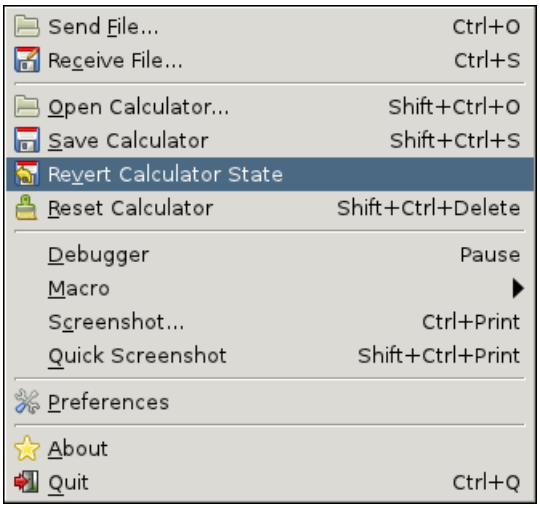

Figure 3.16: The "Reset Calculator" menu entry

Guess what does this option :)

3.8 Debugger

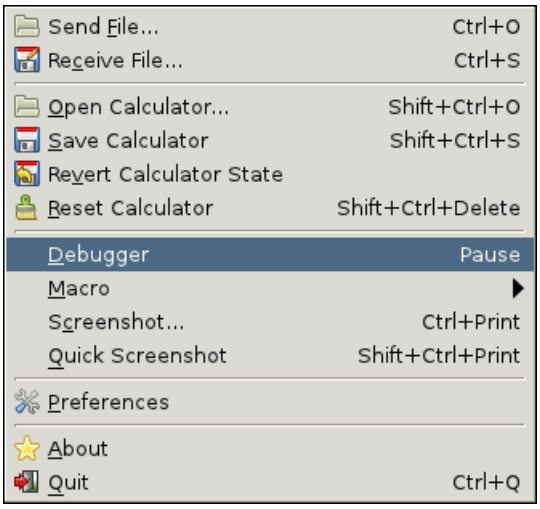

Figure 3.17: The "Debugger" menu entry

When you click on this option, the debugger window will appear.

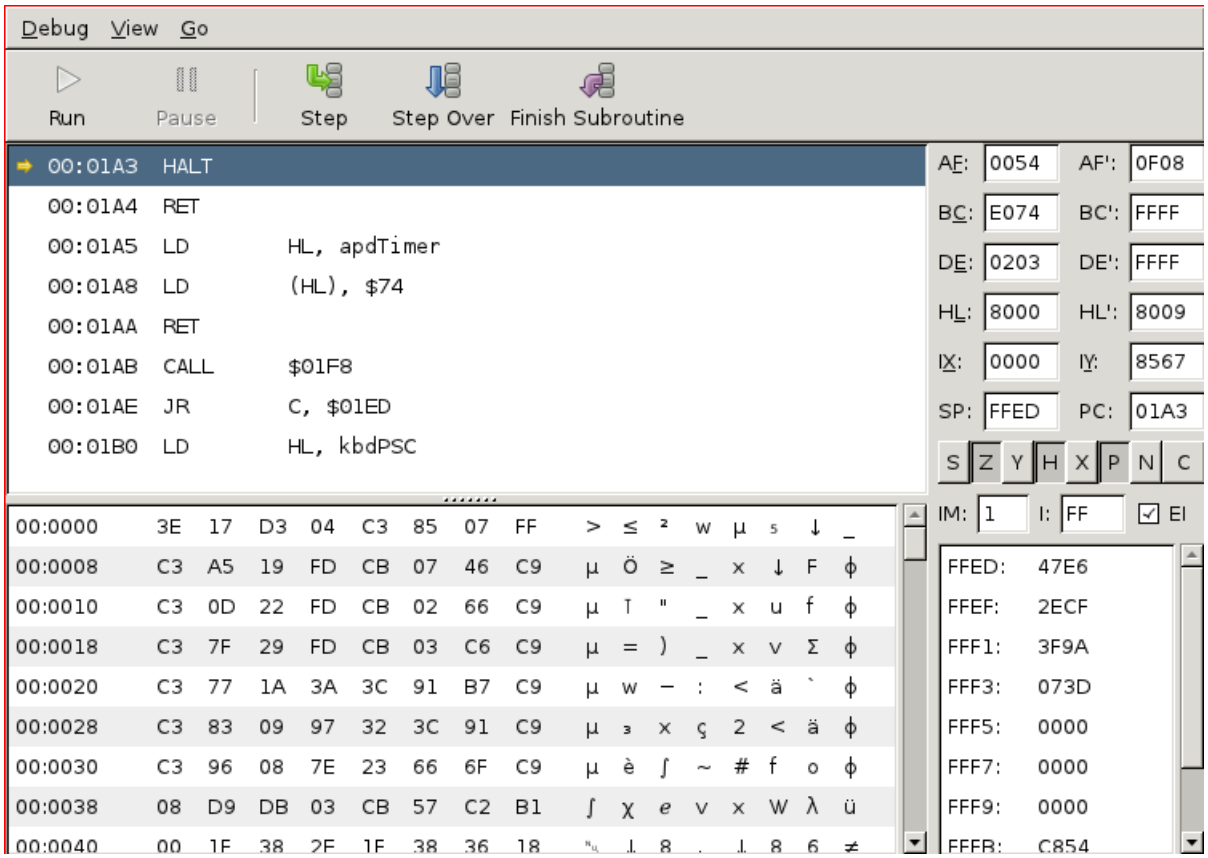

Figure 3.18: A nice and powerful debugger

There's a lot of things to say about debugger.
When you launch it, calculator is automatically paused.
As you can see there are 5 big buttons : Run, Pause, Step, Step Over, Finish Subroutine.
Step just execute one instruction.
As you can see, all the instructions are not the same length, that's why it doesn't step one byte per one byte.
Step over do the same job than step but do not follow call.
Finish subroutine just do basically the same job but stop after a ret.

Now just see what's the differents view of the debugger dialog.
There's a big frame for disassembly view.
In this frame, you can see the adress and the disassembly instruction.
On right click, you can do some useful actions : Breakpoint here, Go to adress, go to PC.

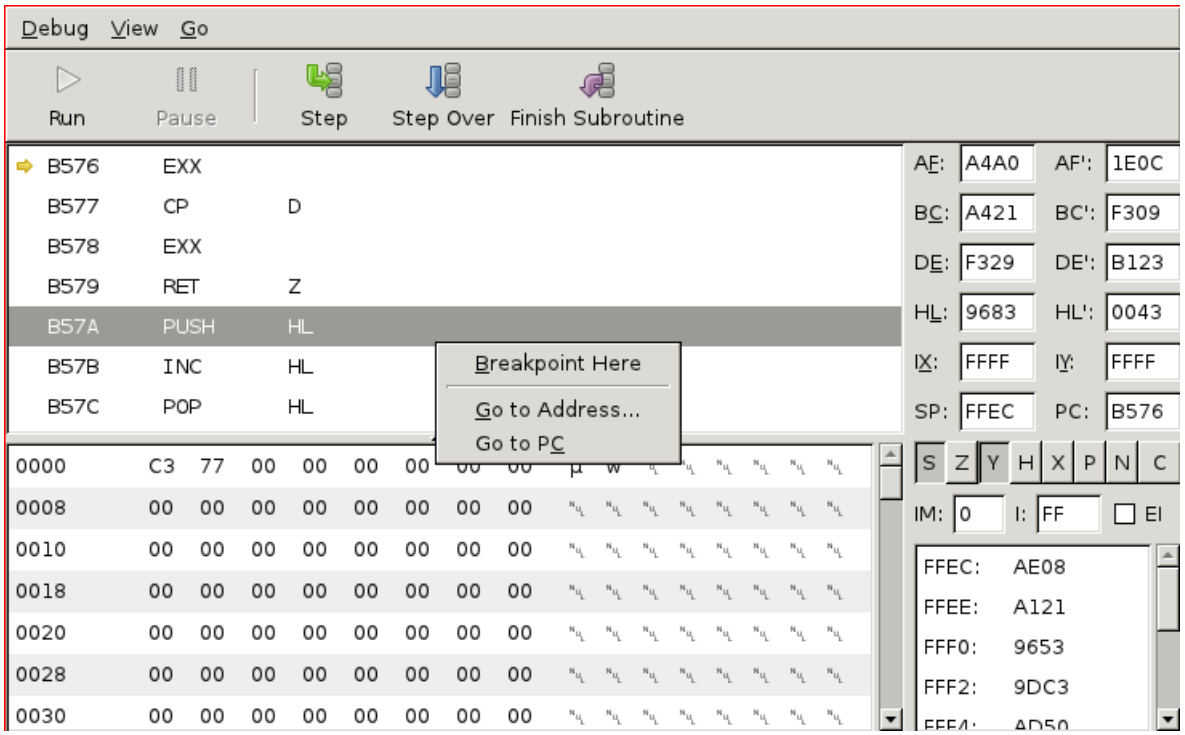

Figure 3.19: The right click menu on disasm view

There's 2 kind of adress notation for this view : Logical and Absolute.
You can switch it into the "View" menu.

Figure 3.20: Switch between logical and absolute adresses

The second big frame is the memory view.
For this view you can switch the adresses representation if you want.
In this view you can see what your calculator contains.
You can also edit the memory and change some values by your own.

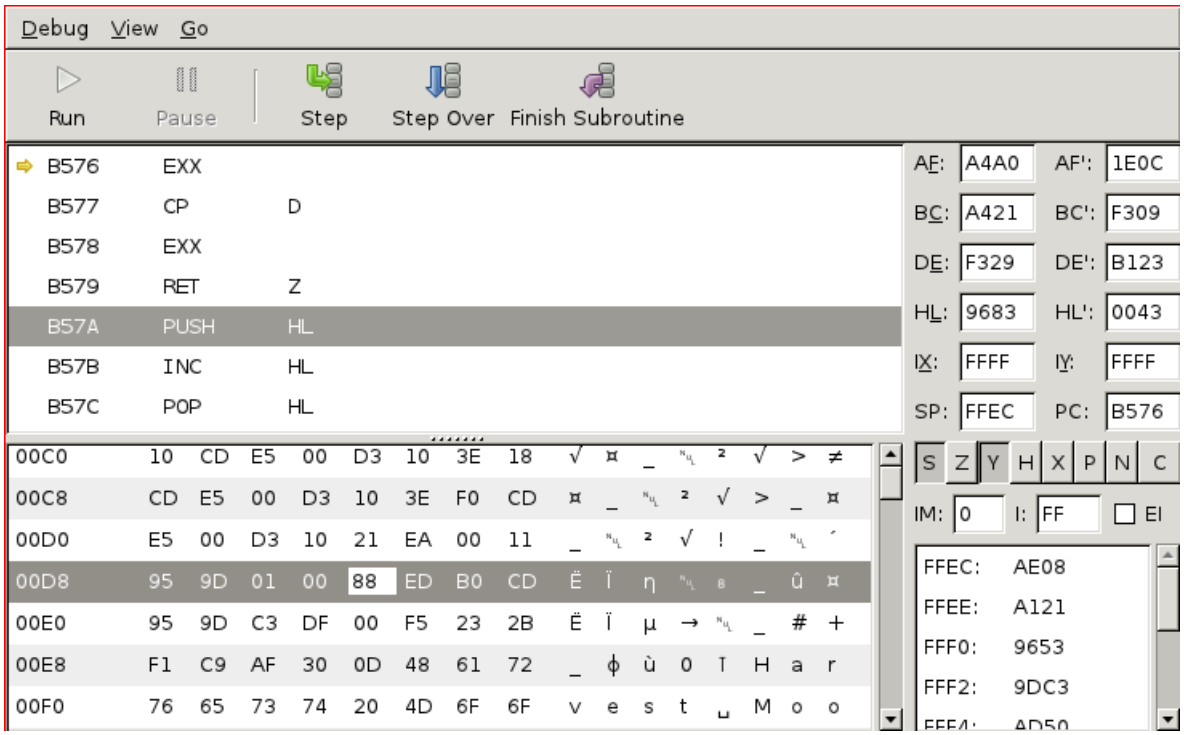

Figure 3.21: Edit the memory

A third view represents the registers.
You can edit them too.
Below registers there is a bunch of toggle button to represent the flags (you can change it).
Then Interruption Mode IM, I, and Enable Interrupt (checkbox).
The finally the stack.

At the top of the debugger window, you can see a menu "Debug".

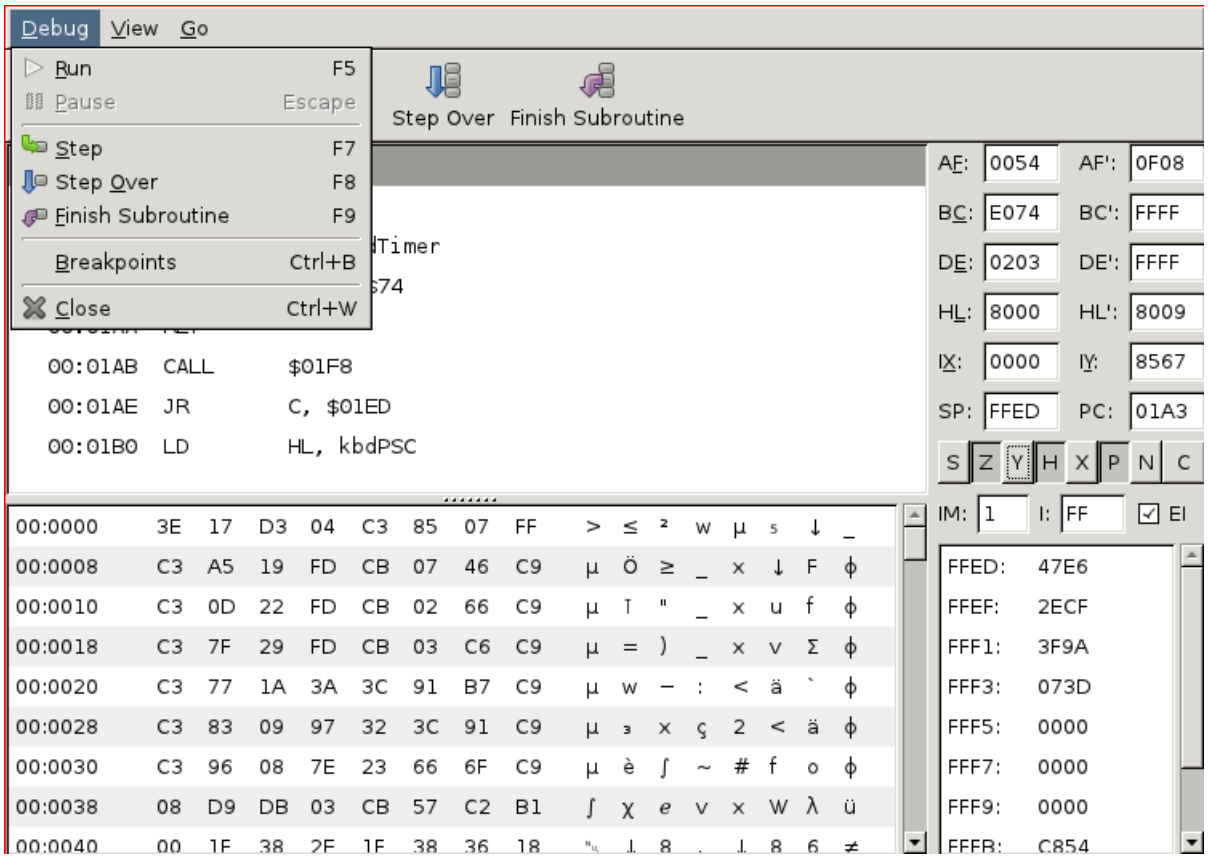

Figure 3.22: The "Debug" menu entry

The options are the same than buttons but there is a big news : breakpoints. Breakpoints are a big part of the life of a assembly developpers.
This option opens the Breakpoint menu when you can "Add", "Remove", "Edit", "Clear" or some special action like "Break on invalid instructions" or "Break on undocumented instructions".

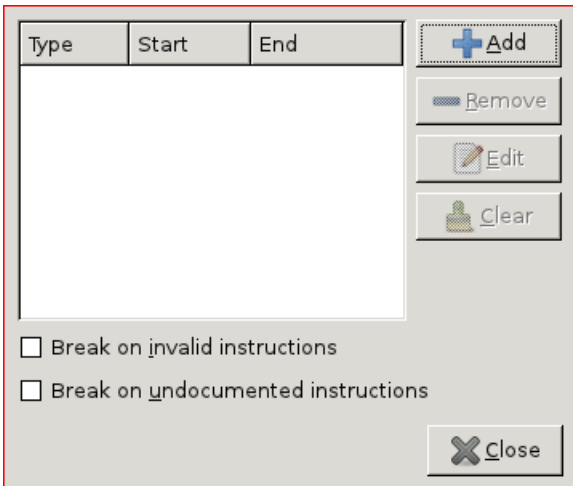

Figure 3.23: The "Breakpoints" menu

The easiest way to add a breakpoint is to right click on the disasm view and click on "Breakpoint here" but you can also set a breakpoint using its adress (logical or absolute).

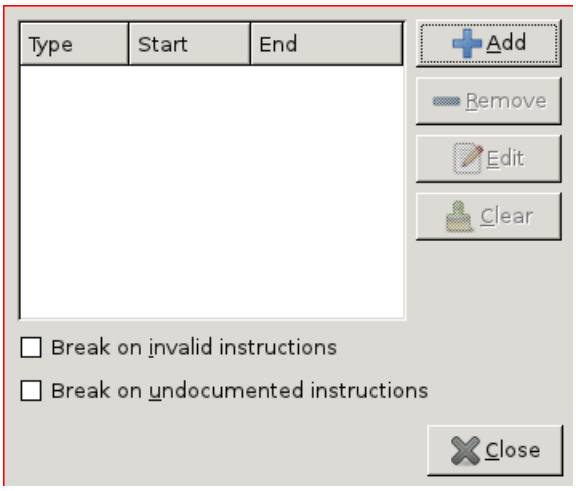

Figure 3.24: Adding a breakpoint

The "Go" menu basically provide and easy way to navigate into the disasm view and th e stack.

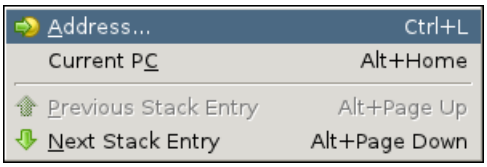

Figure 3.25: The "Go" menu entry

Now I must talk about a nice tool called Keypad (into View menu).

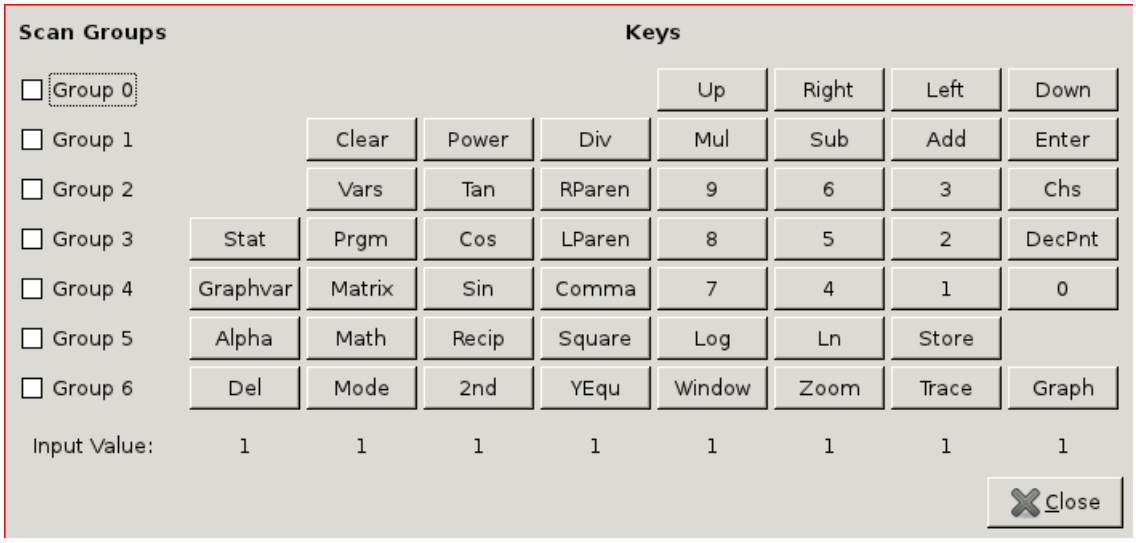

Figure 3.26: The keypad

3.9 Macro

Macros are an easy way to simulates key press, file loading, reset automatically.
It means that you could record a macro then click on some keys, then stop.
If you play it, tilem will press the same keys for you.
Have you never think too lazy to press always "2nd catalog asm(" each time you want to test your new asm production.
Simply use a macro to load and launch your program automatically !

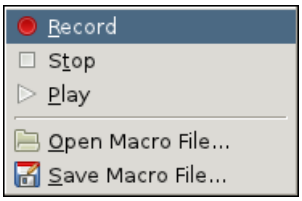

Figure 3.27: The "Macro" submenu

About the options, you can play an already loaded macro or a macro you just have recorded.
You can also open a macro and save the current macro (which one you just have recorded).

3.10 Screenshot...

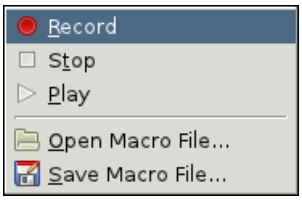

Figure 3.28: The "Screenshot..." submenu

By clicking on this option, you launch a screenshot dialog.

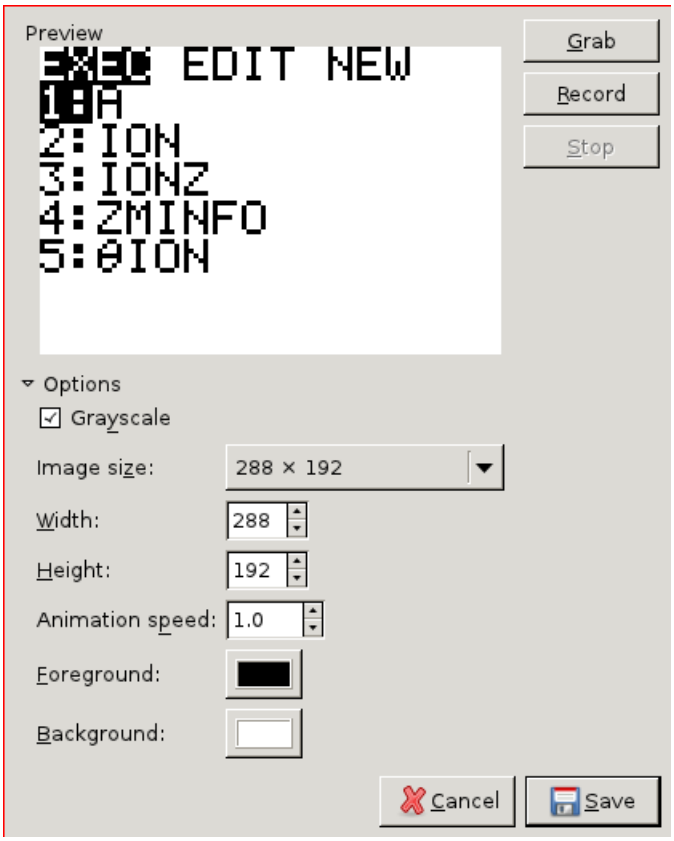

Figure 3.29: The screenshot dialog

You can grab static screenshot (multiple format) or animated screenshot (will be saved as gif).
As you can see TilEm2 has a lot of screenshot configuration.
So you can change the size, change the foreground and background colors.
Use or not grayscale.

3.11 Preferences

This is where you can set the skin (or disable using it) and some important other stuff.

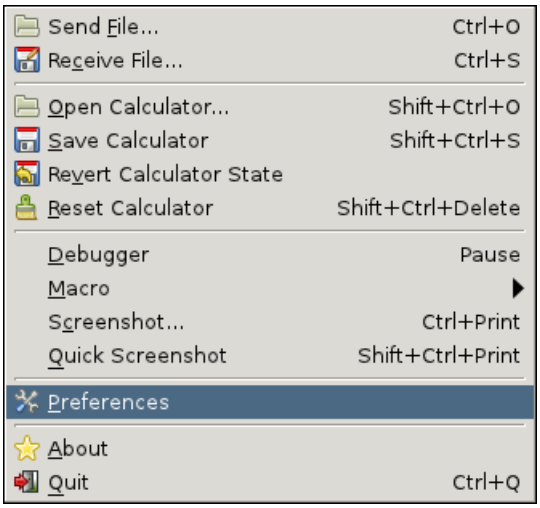

Figure 3.30: The "Preferences" menu entry

You can limit speed or not.
Emulate grayscale (if you don't know just let it checked by default).
Use smooth scrolling.

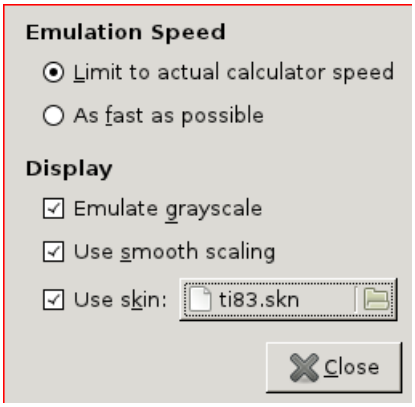

Figure 3.31: The preferences dialog

And an important user friendly feature...
Set skin !
When you click on the button, a file choose will popup and lt you choose the skin.

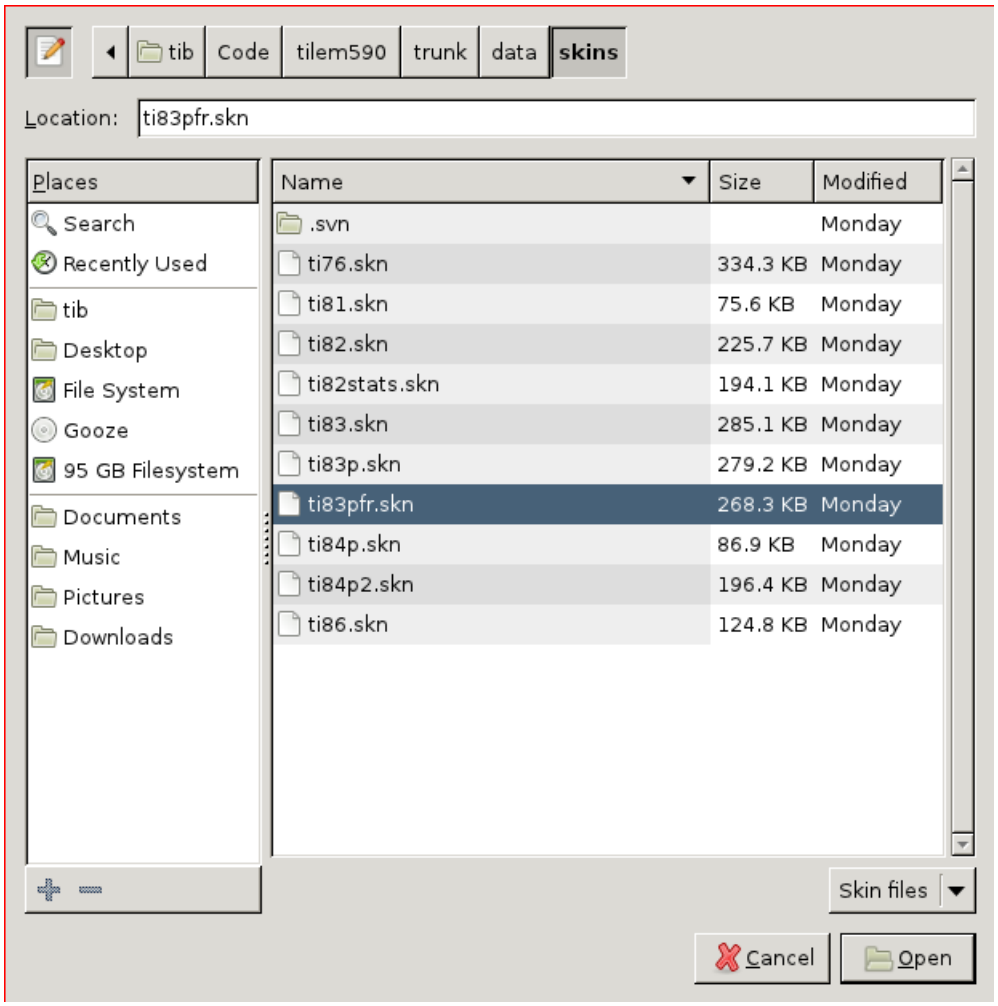

Figure 3.32: The skin file chooser

3.12 About

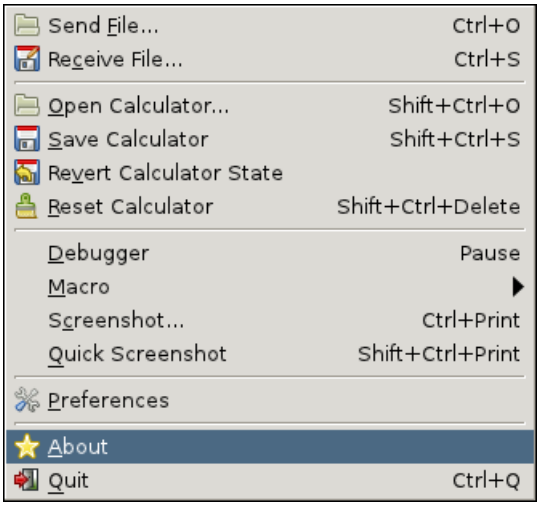

Figure 3.33: The "About" menu entry

No more than an about dialog :)

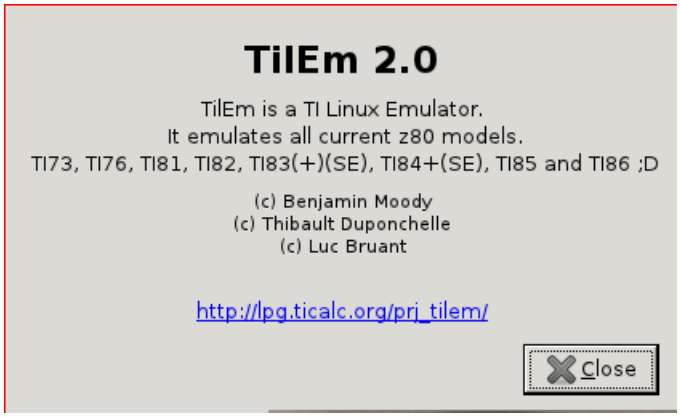

Figure 3.34: The about dialog

3.13 Quit

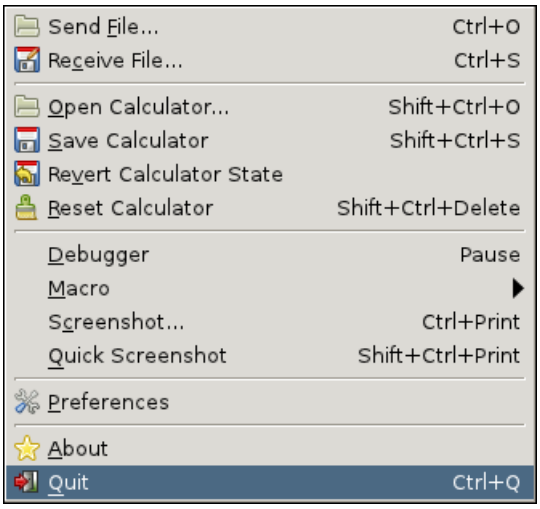

Figure 3.35: The "Quit" menu entry

Bye Bye ;)

Chapter 4

Command line usage

4.1 Basics

TilEm2 is basically made for Linux and command line is our first love :)
Here the options at launch.

```
Usage:
  tilem2 [OPTION...] FILE

Help Options:
  -h, --help                Show help options
  --help-all                Show all help options
  --help-gtk                Show GTK+ Options

Application Options:
  -r, --rom=FILE            The rom file to run
  -k, --skin=FILE           The skin file to use
  -m, --model=NAME          The model to use
  -f, --file=FILE           The file to load
  -s, --state-file=FILE     The state-file to use
  -l, --without-skin        Start in skinless mode
  --reset                   Reset the calc at startup
  --get-var=FILE            Get a var at startup
  -p, --play-macro=FILE     Run this macro at startup
  -d, --debug               Launch debugger
  --normal-speed            Run at normal speed
  --full-speed              Run at maximum speed
  --display=DISPLAY         X display to use
```

:

You should usually use something like :
tilem2 -r /path/to/my/rom

But as you can see you can specify the skin with -k.
If you usually use more than one model, you can try -m and it will load the rom associated with this model (if you already start a rom from this model).
You can specify a different save state (by default it uses the one which is called

as the rom file).
You can start skinless.
You can load a file and even launch a macro at startup (in this case loading a file is done before macro playing).
You can reset too, get a var (if possible) and launch debugger.
Some options could be set at startup as normal speed or full speed (as fast as possible).
Other options are not TilEm2 options (–display by example).
Something is missing?

4.2 Examples

First starting using configuration file (need to have already started one time before).

```
tilem2
```

:

You can specify the rom to use:

```
tilem2 -r /path/to/my/rom.rom
```

:

You can choose a skin at startup

```
tilem2 -r /path/to/my/rom.rom -k /path/to/my/skin
```

:

Choose a save save state:

```
tilem2 -r /path/to/my/rom.rom -k /path/to/my/skin.skn -s /path/to/my/
    savestate.sav
```

:

Or just starting a model without giving a rom file:

```
tilem2 -m ti83
```

:

Starting skinless :

```
tilem2 -m ti83 -l
```

:

Reset the calc at startup :

```
tilem2 -m ti82 --reset
```

:

Trying to get a var at startup:

```
tilem2 -m ti82 --get-var=ION
```

:

Starting in full speed mode:

```
tilem2 -r $rom --full-speed
```

:

Or in normal speed:

```
tilem2 -r $rom --normal-speed
```

:

Play a macro at startup:

```
tilem2 -r $rom -f /path/to/my/best/program -p /path/to/my/macro
```

:

Warning, in this situation, load + macro play, the load is done before the macro playing.

You can also launch debugger (and pause the calc by extension) :

```
tilem2 -r $rom -d
```

:

Chapter 5

Configuration files

5.1 General configuration

You should not edit this file because you don't need it.
If something wrong, by your fault or a bug in TilEm2, just get a new configuration file from the TilEm2 website and replace the wrong config.ini file.
It's usually copied into /.config/tilem2/config.ini

5.2 Keybindings

The keybindings are defined in a keybindings.ini file (usually copied into /.config/tilem2/keybindings.ini).
There's currently no tool to interactively edit this file, but you can edit it by hand if you do this carefully.
This file uses inheritance, it means that a part of the keybindings are common to all models, but you can rewrite them for each model.
So firstly, TilEm2 parses common, then if he find another keybindings for the same keypress, he rewrite it.
There's one subsection per model.