SPASM-ng is a z80 assembler with extra features to support development for TI calculators.

⚖ GPL-2.0 license

☆ 66 stars ⑂ 20 forks

☆ Star	◉ Watch ▾

<> Code	⊙ Issues 24	⑃ Pull requests 1	▶ Actions	▦ Projects	📖 Wiki	⚠ Security

⑂ master ▾   ···

| 🧑 calc84maniac Fix segfault in built-in buf() macro ··· | ✓ on Jul 3 | 🕑 220 |

View code

☰ README.md

spasm-ng

`build` `passing` 🔧 `build` `passing` `coverage` `46%` `coverity` `passed` `irc channel` `#spasm-ng on EFNet`

SPASM-ng is a z80/eZ80 assembler with extra features to support development for TI calculators.

Downloads

Releases with pre-built binaries may be found here:

[Downloads](#)

Requirements

On Windows, you will need the following to build SPASM-ng:

- Visual Studio with C++ support

Visual Studio 2010 or newer is recommended, but it may be possible to downgrade. See http://stackoverflow.com/a/16196505 for more info.

Note that if you're using Visual Studio 2010, SP1 or newer is required. Without the service pack, the build will fail.

Visual Studio Express editions can be used for building. This include Express for Desktop, Visual C++ Express, etc.

If you want to build with COM support, however, you will need the ATL include files. You can get these by either:

- using Visual Studio Community (2015+, free w/restrictions);
- using Visual Studio Professional;
- installing Windows SDKs that contain the ATL include files, and reconfiguring the project accordingly.

On Linux, Mac OS X, and other *nix, you will need the following to build SPASM-ng:

- C/C++ Compiler (GCC, Clang, etc.)
- GMP Library (found at https://gmplib.org/)
- OpenSSL Library (found at https://www.openssl.org/)

Note that if you are using Linux, your distribution will likely have pre-built packages (and development packages) available to install. You should use these first before attempting to install the above libraries yourself.

Suggested packages for Ubuntu/Debian:

- build-essential
- libssl-dev
- zlib1g-dev
- libgmp-dev
- checkinstall (optional, only if you want to build Debian packages - .deb files)

If building without app signing support (see *Building* in this document), GMP and OpenSSL are not required.

To run the tests, you will also need Python 3.1+. Versions below Python 3.1 will not work!

Building

On Windows, simply build with the included Visual Studio project file.

If you wish to build with COM support, you must:

- define `SPASM_NG_ENABLE_COM` globally within the Visual Studio project;
- ensure that ATL include files are accessible.

On Linux, Mac OS X, and other *nix, simply build by running:

```
# Git users: run this to update the version
# ONLY RUN THIS IF YOU ARE USING GIT
make version

# Now build!
make

# Optional: install SPASM-ng (you may need sudo/su/root):
make install
```

To disable app signing support in your binary (thus eliminating the dependencies on OpenSSL and GMP), define NO_APPSIGN when invoking `make`:

```
make NO_APPSIGN=1
```

For Debian systems, you can run the following to create a Debian package:

```
# You should "make clean" before running this!
make clean

# Create a Debian package (builds opt target):
make debian
```

Other Makefile commands:

```
# You should "make clean" before running any of this!
make clean

# Build with -static (static linking, bigger binary due to
# including libraries):
make static

# Build with -O3 optimization:
make opt
```

```
# Build with both -static and -O3 optimization:
make opt-static

# Create tarball of spasm-ng binary (builds opt-static target):
make tar
```

Cross-compiling can be enabled by simply setting the cross-compiling prefix, e.g.

```
make CROSS_COMPILE=i686-w64-mingw32-
```

Documentation

The program accepts standard Z80 assembly and eZ80 assembly.

SPASM command line help shown below:

```
SPASM-ng Z80 Assembler by Spencer Putt and Don Straney


spasm [options] <input file> <output file>

Options:
-E = Assemble eZ80 code
-T = Generate code listing
-C = Code counter mode
-L = Symbol table mode
-S = Stats mode
-O = Don't write to output file
-I [directory] = Add include directory
-A = Labels are cAse-sensitive
-D<name>[=value] = Create a define 'name' [with 'value']
-N = Don't use colors for messages
-V <Expression> = Pipe expression directly into assembly
```

Added features for eZ80 code include:

- The .ASSUME ADL={0 or 1} directive for specifying 16-bit or 24-bit code (24-bit is the default).
- The .DL and .LONG directives for including 24-bit data.
- Mode-change suffixes available for all instructions, as seen in the eZ80 manual.

Issues/Bugs

Report issues/bugs to the issue tracker, found here:

https://github.com/alberthdev/spasm-ng/issues

Copyright/License

SPASM-ng was originally from the SPASM project, and was forked to fix a few bugs. It was originally written by Spencer Putt and Don Straney, with additional development by Chris Shappell and James Montelongo.

License:

```
SPASM-ng - a z80 assembler with extra features to support dev for TI calcs!
Copyright (C) 2015 Spencer Putt and Don Straney
Copyright (C) 2015 Chris Shappell and James Montelongo
Copyright (C) 2015 Albert Huang (fixes to SPASM)
Copyright (C) 2015 Brendan Fletcher (eZ80 support)

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
```

Releases

🏷 **3** tags

Packages

No packages published

Contributors 7

Languages

- **C++** 42.8%
- **Assembly** 40.2%
- **C** 14.8%
- **Python** 1.2%
- **POV-Ray SDL** 0.5%
- **Makefile** 0.3%
- **Other** 0.2%