Z88DK - The Development Kit for Z80 Computers

z88dk passing

Z88DK is a collection of software development tools that targets the 8080 and z80 family of machines. It allows development of programs in C, assembly language or any mixture of the two. What makes z88dk unique is its ease of use, built-in support for many z80 machines and its extensive set of assembly language library subroutines implementing the C standard and extensions.

INSTALLATION

There are several ways to install z88dk.

1. Use the Most Recent Official Release currently 2.2 dated 23 July 2022. Follow these installation instructions.
2. Get the Nightly Build. Every night we build complete binary packages for windows and osx and generate source packages for everyone else. The same installation instructions apply.

Using a nightly build means you can keep up with bugfixes and new features rather than having to wait an entire year for a release to occur.

3. Use the Snap package on Linux
4. Use GitHub and build it yourself. The z88dk repository uses git submodules, these are not automatically downloaded by git by default so you will have to either adjust your clone line, or retrieve them manually. To clone with submodules use `git clone --recursive https://github.com/z88dk/z88dk.git`. To add the submodules to an already existing clone use `git submodule update --init --recursive`. To build, the following instructions should be followed.

CMAKE

CMake can be used to build Z88DK projects (a toolchain is provided). Please refer to CMake wiki.

THE TOOLS

Many tools have a `z88dk-` prefix to distinguish them from tools from other packages that may be installed with the same name. The documentation generally omits the prefix when referring to them.

- **ZCC** is the toolchain's front end. zcc can generate an output binary out of any set of input source files.
- **SCCZ80** is z88dk's native c compiler. sccz80 is derived from small c but has seen much development to the point that it is nearly c90 compliant.
- **ZSDCC** is z88dk's customization of the sdcc compiler. Our patch makes sdcc compatible with the z88dk toolchain, gives it access to z88dk's extensive assembly language libraries and ready-made crts, addresses code generation bugs where present and improves on sdcc's generated code.
- **Z88DK-Z80ASM** (not to be confused with several external projects called z80asm) is a fully featured assembler / linker / librarian implementing sections.
- **Z88DK-Z80NM** is z80asm's companion archiver. It can provide a listing of functions or data encoded in an object or library file.
- **Z88DK-APPMAKE** processes the raw binaries generated by the toolkit into a form suitable for specific target machines. For example, it can generate intel hex files, tapes, ROMs, etc.
- **Z88DK-TICKS** is a command line z80 emulator that can be used to time execution speed of code fragments. Ticks includes a debugger and disassembler in addition to supporting some of the ZX Next Z80N extension instructions.
- **Z88DK-GDB** provides the debugger interface from ticks and connects to a gdbserver to permit line-by-line debugging of software in emulators or on real hardware.
- **Z88DK-DIS** is a command line disassembler for Z80, Z180, Z80N and Rabbit 2000/3000. It can additionally read map files generated by z80asm to provide a more symbolic output.

- **Z88DK-LIB** is an installer for third party libraries. It manages installation, removal and listing of available libraries.
- **Z88DK-ZX0** and **Z88DK-ZX7** are PC side data compression tools with companion decompression functions in the z80 library.
- **Z88DK-ZX0** and **Z88DK-ZX7** are PC-side decompressor counterparties to the z88dk-zx0 and z88dk-zx7.

These tools are not normally directly invoked by the user:

- **M4** acts as z88dk's macro preprocessor and can optionally process files ahead of the c preprocessor or assembler.
- **Z88DK-UCPP** is the c preprocessor invoked for sccz80.
- **ZSDCPP** is the c preprocessor invoked for zsdcc.
- **Z88DK-ZPRAGMA** is used by the toolchain to process pragmas embedded in c source.
- **Z88DK-COPT** is a regular expression engine that is used as peephole optimizer for sccz80 and as a post-processing tool for both sccz80 and zsdcc.

BENCHMARKS

The assembly language libraries supplied by z88dk give it performance advantages over other z80 compilers. For details please look at the Benchmarks section in the Wiki.

- **Dhrystone 2.1** Dhrystone was a common synthetic benchmark for measuring the integer performance of compilers in the 1980s until more modern benchmarks replaced it. It attempts to simulate typical programs by executing a set of statements statistically determined from common programs.
- **Pi** Mainly measures 32-bit integer performance.
- **Sieve of Eratosthenes** Popular benchmark for small machine compilers because just about everything is able to compile it. As a benchmark it doesn't reveal much more than loop overhead.
- **Whetstone 1.2** Whetstone is a common synthetic floating point benchmark.
- **Program Size** Program size has great importance for small machines. A collection of test programs were compiled for the common cp/m target and resulting binary sizes were compared.

USING Z88DK

Unfortunately, like a lot of open source projects, we could use a lot of help with the documentation.

Some things to know:

- There are two c libraries in z88dk. These are referred to as the classic c library and the new c library.
- There are two c compiler in z88dk. Projects built using the classic library can mix object files generated by the both compilers. Projects built with newlib must use only one of the compilers.
- Thankfully there is only one assembler so we only need to deal with 2*2 combinations :)

When you form a compile line you must decide which compiler you will use and which c library you will link against. You will make that decision based on which targets you want to compile for and what features you need.

The classic c library is z88dk's original c library and it has crts that allow generation of programs for 80+ different z80 machines. The level of support for each is historically determined by user interest. Documentation begins here and example programs can be found in z88dk/examples with compile lines most often appearing at the top of .c files.

The new c library is z88dk's rewrite aiming for a large subset of C11 conformance. It directly supports eleven targets currently (cpm, hbios, rc2014, scz180, sega master system, yaz180, z180, z80, zx spectrum, and zx spectrum next) but the z80 target (aka embedded target) can also be used to compile programs for any z80 machine. Documentation begins here and example programs can be found in z88dk/libsrc/_DEVELOPMENT/EXAMPLES with compile lines most often appearing at the top of .c files.

QUICK LINKS

Z88DK Home Page Includes a link to the nightly builds where you can get an up-to-date package.

Install Instructions

Bug Reporting (old bugs in the forum)

Introduction to Compiling Using the Classic C Library Examples in z88dk/examples

Introduction to Compiling Using the New C Library Examples in z88dk/libsrc/_DEVELOPMENT/EXAMPLES

Using z88dk with the rc2014 target, covers cpm, hbios, and rc2014 subtypes.

Using z88dk with zx, covers the zx target, and by extension the zxn target.

Releases 4

Version 2.2 Latest
on Jul 23, 2022

+ 3 releases

Packages

No packages published

Contributors 54

+ 43 contributors

Languages

● **C** 43.3% ● **Assembly** 42.8% ● **M4** 6.1% ● **Roff** 2.5% ● **Perl** 1.9% ● **HTML** 1.8% ● **Other** 1.6%