

## how to compile stdio C program for ZX Spectrum using Z88DK and using the old compiler sccz80??



Z88DK is a Z80 C cross compiler supplied with an assembler/linker and a set of libraries implementing the C standard library for a number of different z80 based machines. It was designed to allow developers to easily program for systems that utilize the Z80 processor, like the ZX Spectrum. Here's a basic walkthrough on how to compile a C program using Z88DK and sccz80, which is the older compiler included in Z88DK.

1. First, you should have your C program ready. For this example, let's consider you have a basic "Hello, World!" program saved as **hello.c**. This program should include **stdio.h** for input/output functions.

c

Copy code

```
#include <stdio.h> int main() { printf("Hello, World!"); return 0; }
```

2. Make sure that the Z88DK is correctly installed and the binary directory is in your system's PATH. If you installed Z88DK in **/usr/local/bin**, for example, you can add it to the PATH in a Unix-like system with:

bash

Copy code

```
export PATH=$PATH:/usr/local/bin/z88dk/bin
```

3. Compile your program using the **zcc** command, which is the front-end to the Z88DK compilation suite. Specify the target machine as **zx**, indicate the use of the **sccz80** compiler and provide the input and output file names. Here's how you can do this:

bash

Copy code

```
zcc +zx -vn -clib=new -O3 -startup=1 -o  
hello.bin hello.c
```

In this command:

- +zx**: Sets the target machine as the ZX Spectrum.
- vn**: Increases verbosity to help catch any errors or warnings during compilation.

**-c lib=new**: Chooses the new C library, which includes better support for the C standard library.

**-O3**: Sets the optimization level to 3 (sccz80 only supports **-O0**, **-O1**, **-O2** but for compatibility with zsdcc, **-O3** is also accepted). Note that high optimization levels might make debugging harder.

**-startup=1**: Sets the type of startup (0=basic, 1=tap, 2=plus3dos, 3=tap+plus3dos). Choose according to your needs.

**-o hello.bin**: Names the output binary file as **hello.bin**.

**hello.c**: Specifies the C source file to compile.

4. After running this command, you will have **hello.bin** file. To convert it to a **.tap** file that can be loaded on a ZX Spectrum or an emulator, use the **appmake** utility:

```
bash  Copy code  
  
appmake +zx -b hello.bin -o hello.tap --org  
32768 --blockname main
```

In this command:

**+zx**: Sets the target machine as the ZX Spectrum.

**-b hello.bin**: Specifies the input binary file.

**-o hello.tap**: Names the output tape file as **hello.tap**.

**--org 32768**: Sets the origin (starting address) of the binary code. The value **32768** is a typical starting address for ZX Spectrum programs.

**--blockname main**: Names the tape block as **main**.

Now you should have a **hello.tap** file that can be loaded onto your ZX Spectrum or into an emulator.

Please note that not all of the C standard library is fully supported for the ZX Spectrum, particularly when using the older **sccz80** compiler. You may need to find or create alternative functions for any that are missing or don't work as expected. Always check the Z88DK documentation and

---