



The Sinclair QL is a personal computer launched by Sinclair Research in 1984. It uses a Motorola 68008 CPU and runs an operating system called QDOS. To compile C code for the Sinclair QL using a modern GCC-based toolchain like qdos-gcc, you would need to follow these steps:

1. Obtain a suitable GCC toolchain: Start by getting a GCC toolchain that supports the Motorola 68000 family of processors, including the 68008 used in the Sinclair QL. You may need to compile this toolchain yourself or find a precompiled version for your operating system.
2. Configure GCC for the target architecture: Configure the GCC toolchain to generate code for the Motorola 68008 processor, which may involve specifying a target architecture or processor flag when compiling your C code. For example, you might need to use the `-mcpu=68008` flag.
3. Add support for QDOS: To interface with the QDOS operating system, you will need to include appropriate header files and libraries that define QDOS system calls, data structures, and other necessary components. These files may be available as part of a Sinclair QL development package or from a third-party source.
4. Write C code: Create your C source files, making sure to use the appropriate QDOS-specific functions and data structures provided by the headers and libraries you included.
5. Compile your code: Use the GCC toolchain to compile your C source files, specifying the target architecture, QDOS headers and libraries, and any other necessary flags. For example:

Copy code

```
qdos-gcc -mcpu=68008 -o my_program my_program.c
```

6. Transfer the compiled binary to the Sinclair QL: Once you have compiled your program, you'll need to transfer the binary to your Sinclair QL. This may involve converting the

binary to a format compatible with the QL (e.g., a QDOS executable file) and then using an emulator, an SD card interface, or another method to transfer the file to your QL.

7. Run your program on the Sinclair QL: With the binary transferred to your Sinclair QL, you should be able to run your compiled C program using the QDOS operating system.

Please note that the specific details and commands may vary based on the exact GCC toolchain and libraries you are using. Make sure to consult the documentation for the toolchain and libraries you are using for more specific information.



well okay, but which `#define` is issued to identify compilation for QL by compiler ??



Apologies for misunderstanding your question earlier. When you compile C code for a specific platform or architecture, the compiler typically provides predefined macros to help identify the target system. These predefined macros can be used in the C code to enable or disable specific features or code sections depending on the target platform.

For the Sinclair QL and QDOS, there doesn't seem to be a widely accepted or standardized predefined macro. However, you can define your own macro to identify the compilation for the Sinclair QL and QDOS. For example, you can use:

c

Copy code

```
#define QDOS
```

or

c

Copy code

```
#define SINCLAIR_QL
```

Then, in your C code, you can use `#ifdef` directives to conditionally compile code based on the presence of the predefined macro:

c

Copy code

```
#ifdef QDOS // Code specific to Sinclair QL and QDOS #endif
```

or

```
c
```

 Copy code

```
#ifdef SINCLAIR_QL // Code specific to Sinclair QL and QDOS #endif
```

If you want to automatically include this macro during compilation, you can pass it as a command line option to the GCC compiler:

 Copy code

```
qdos-gcc -mcpu=68008 -DQDOS -o my_program my_program.c
```

or

 Copy code

```
qdos-gcc -mcpu=68008 -DSINCLAIR_QL -o my_program my_program.c
```

Keep in mind that the macro name you choose should be descriptive and unique enough to avoid potential conflicts with other macros.



well, I completely replaced detection of desktops by explicit define DESKTOP ... better

---