ECROS Technology

www.ecrostech.com

Flash Loader for ZiLOG Z8 Encore!®

Comparison with ZiLOG Flash Loaders

Document Revision 1.0

for Flash Loader Version 1.10

December 9, 2003

Table of Contents

Introduction

ECROS Technology produces a Flash Loader for the ZiLOG Z8 Encore! It allows the firmware of a Z8 Encore!-based product to be changed in the field by end-users and / or field service personnel. Existing firmware can be erased and new firmware loaded through one of the serial ports (UARTs) of the microcontroller.

ZiLOG also provides capabilities to program the Flash memory of the Z8 Encore! This document compares the ECROS Technology Flash Loader with ZiLOG's offerings.

ZDS II Built-In Flash Loader

Fairly obviously, the ZiLOG Developer's Studio tool set has built into it the ability to load programs into Flash. This uses the Z8 Encore! on-chip debugger. Normally, it is used as part of the IDE by clicking the Download Code button or making menu selections. It can also be used via the command line interface as described in ZiLOG Application Note AN016601.

This means of loading Flash is perfectly adapted for a development environment and is also usable in a manufacturing environment. It is not well suited to the field-update of firmware because of the need to expose the DEBUG signal and use ZDS II.

Direct Programming

The Z8 Encore! Flash can be programmed "directly" by bypassing the internal Flash Controller and applying external signals to the device. This is described in ZiLOG Application Note AN011702.

This means of loading Flash is perfectly adapted for manufacturing environment. It is not at all suited to the field-update of firmware for obvious reasons.

Application Note AN011802

In this Application Note, entitled *Flash Loader for Z8 Encore!™*, ZiLOG describe a Flash Loader utility that resides in Flash along with the application program and can erase and reprogram the application in response to commands and uploaded Intel Hex files from a UART.

This utility is, at first glance, suitable for the field-update of Z8 Encore! firmware as well as in a manufacturing setting. However, the Flash Loader takes up about 13 Kbytes of the microcontroller's Flash memory and has other more obscure problems. ZiLOG would probably not recommend using this utility at this point in time (see AN016401, below).

Application Note AN016401

ZiLOG published a new Application Note describing a Flash Loader in late November or early December of 2003. The application note is entitled *Boot Loader for Z8 Encore!® MCUs*, but it nevertheless describes a Flash Loader. There is considerable overlap in functionality between this ZiLOG "boot loader" and the ECROS Technology Flash Loader. The remainder of this document concentrates on comparing these two utilities to help you decide which to use.

Cost

The ZiLOG boot loader is free and source code is provided. This is a clear advantage as long as it does what you need or something very close to it. ECROS Technology charges for the commercial use of its Flash Loader. If your requirements are not met by the boot loader and it will require some modification, you should compare the likely cost of those modifications to the cost of the Flash Loader. Even if the Flash Loader will also need to be changed, bear in mind that it is written in C and is therefore easier to change and that ECROS Technology is available to make the changes for you.

Size

The ZiLOG boot loader occupies only a single page of Flash memory, less than 512 bytes. In part, this is achieved by the use of assembly code at the expense of the ability to understand, verify and modify its functionality. However, as with all software, there is a tradeoff between feature set and size. The ECROS Technology Flash Loader takes up three Flash pages (1½ Kbytes). If you have the additional 1 Kbyte of Flash to spare, you may benefit from the Flash Loader's more complete feature set, as explained in the paragraphs below.

Robustness

It is not know whether the ZiLOG boot loader has been deployed in the field. The ECROS Technology Flash Loader has been in use in six countries without incident. If you want to make it possible for the firmware of your product to be updated in the field, you must evaluate the possible cost to you of supporting this feature and addressing any problems that your users experience. A Flash Loader that has seen field use may be a better choice than one that has not.

Ease of Evaluation

The ECROS Technology Flash Loader is delivered ready to configure and install. It comes with a sample application program and a suite of Intel Hex files with which you can verify its operation over a range of both valid and invalid inputs. For some reason, the instructions for installing the ZiLOG boot loader include modification of the project settings. No sample application is supplied, so once you have the boot loader installed you have to devise your own tests.

Basic Capability

The ZiLOG boot loader has one basic capability. If you activate it, it erases the existing application firmware and uploads and programs a perfectly constructed Intel Hex file as the new application. If you activate it by mistake or there are any errors in your new firmware file, you are left with a product that has no application in it. This may not matter, for example if you make it difficult to activate the flash programming mode by mistake and provide a hosted front-end to validate the firmware file before activation. Or, you could modify the boot loader to make it more safe, but see Cost and Size above.

The ECROS Technology Flash Loader requires an explicit command to erase the current application. As well as uploading a complete new firmware file, it can merge Intel Hex files into an existing application, compare files against the current contents of Flash and, perhaps most important, tell you if your file will load into Flash without problems.

Speed

The ZiLOG application note states that the boot loader operates at a UART data rate of 9,600 bit/s if a special linker directive is used to shorten the length of records in the Intel Hex file below the default. If you don't use this directive, you are cautioned to change the program to use 2,400 bit/s. At 9,600 bit/s, ZiLOG report that an 18 Kbyte Intel Hex file took 90 seconds up be programmed.

The ECROS Technology Flash Loader has been tested at up to 115,200 bit/s, although 38,400 bit/s would be a more sensible choice to keep well clear of the upper limit of the serial protocol speed range. At 115,200 bit/s, a 60 Kbyte Intel Hex file was programmed in 6 seconds, making the Flash Loader 50 times faster than the boot loader. To be fair, part of this difference in speed is due to ZiLOG's use of HyperTerminal, but unless the boot loader is tested with TeraTerm Pro, it cannot be assumed to work with faster terminal emulators. There are no restrictions on the Intel Hex file record length with the Flash Loader. ZDS II produces compatible files when left at its default settings.

Error Checking

It has already been mentioned that the ZiLOG boot loader cannot pre-test a file for errors as the ECROS Technology Flash Loader can. In addition, the only error checking in the Intel Hex file that is performed is the record checksum. This will catch a large class of errors, but provides little in the way of diagnostic information to help you figure out what's wrong. The Flash Loader directly detects and reports non-hexadecimal digits in the file and a bad record type as well as the checksum. Both utilities claim to check that addresses in the input file do not overlap the utility itself (but see Test Results, below), but only the Flash Loader explicitly reports this to the user.

Other Features

The ECROS Technology Flash Loader includes space for a product serial number which can be programmed as part of the manufacturing process and is announced in the Flash Loader banner message.

The ZiLOG boot loader allows the user to chose a starting address for the application program. The ECROS Technology Flash Loader does not. It does not seem a very useful feature, but the main reason this was omitted is that at the time the Flash Loader was designed, ZiLOG's ZDS II made it very difficult to use any starting address other than 0038_{16}. Ironically, the boot loader cannot use that starting address, which makes things awkward for anyone who has chosen to stay with early versions of ZDS II. Space has been set aside in the Flash Loader design to store a user-selected start address and this is planned for a future release in response to user demand.

The ZiLOG boot loader stores a flag to confirm that the upload of an Intel Hex file was completed without error. The ECROS Technology Flash Loader does not, but issues an error report at the UART. If the error report is ignored and the user resets the system or quits the Flash Loader, any partially loaded application will be executed. The additional security of an “upload complete” flag is a good idea. Again, space was set aside in the Flash Loader for this (at Version 1.10) but the feature was not included in that release.

Test Results

The ZiLOG boot loader (from AN0164-SC02.zip) was tested using the general procedure

applied to the ECROS Technology Flash Loader. Additional tests were devised for restrictions and features that are not present in the Flash Loader. HyperTerminal was used in place of TeraTerm Pro as the terminal emulator. This section presents the results. Note that **PASS** means either correctly loaded the application or correctly reported that the load failed. **FAIL** includes reporting success when the load failed.

Correctly Constructed Intel Hex File – PASS. Starting addresses of 0040_{16} and 1000_{16} were tried. The delay before the appearance of the 'R' for ready (while Flash is erased) is a little unnerving.

Bad Starting Address (0038_{16}) – FAIL. The boot loader reports success ('S') but the application is not properly loaded and does not run.

Program Text at Bad Address (0038_{16}) – FAIL. The boot loader reports success ('S') but the application is not properly loaded and does not run.

Long Records in Intel Hex File (ZDS II default) – PASS, BUT ... Using the ZiLOG recommended terminal emulator (HyperTerminal) at 9,600 bit/s there were no problems. However, when using TeraTerm Pro at the same bit rate, the boot loader reported success ('S') but the application was not properly loaded and did not run. Therefore, it is essential to use short records or rely on HyperTerminal to slow down the data. If this is not done, the boot loader fails, i.e. appears to have loaded the file when it has not.

Bad Starting Address ($FE00_{16}$) – PASS. The boot loader reports an error ('E'). As has been pointed out, you can't use the boot loader to pre-test the file, so when this happens you have a system with no firmware in it until you can get a replacement firmware file.

Lower Case in Intel Hex File – PASS.

Comments in Intel Hex File – FAIL. If a colon appears in a comment, the boot loader attempts to interpret what follows as a valid record. Results are unpredictable. An error may be indicated but on other files the boot loader hung.

Checksum Error – PASS.

Non-Hex Characters in Intel Hex File – FAIL. Non-hexadecimal characters are not detected and instead map to some valid digit (for example, G masquerades as 0). This is not a serious problem as the error is likely to be caught by the checksum.

Line Endings – PASS. The boot loader is not sensitive to how line endings are encoded.

Conclusion

The boot loader described in ZiLOG Application Note AN016401 is appropriate where you can be absolutely sure that the user has a perfectly correct Intel Hex file at the ready, will not invoke the flash loading mode by mistake and needs no features other than the total replacement of the application firmware. It is free of charge and source is provided.

The ECROS Technology Flash Loader is appropriate when its extra features justify the price and larger footprint in memory. It performs much more checking of the input file than the boot loader and can check a new firmware file *before* the existing application is erased. It can also compare files to the current contents of Flash, operates at higher bit rates without restrictions as to the record length and includes a product serial number. No situations have been found in which the Flash Loader reports success after a failed upload, which has been seen to happen with the ZiLOG boot loader.