Integrated Device Technology, Inc.

IDT/SIM™ 5.1 SOURCE CODE

TECHNICAL NOTE TN-20

By Upendra Kulkarni

This Technical Note offers a quick overview of the source code environment of IDT/SIM™ (System Integration Manager) (version 5.1).

IDT offers a number of RISC evaluation boards each with a variety of unique features. Consequently, the IDT/SIM on each board has some features which are uniquely tailored for that specific board and some features which are common to all boards. The source code for IDT/SIM for all boards is maintained in a single directory tree structure.

Source code for IDT/SIM (version 5.1) is expected to be used by individuals who have designed boards using a member of IDT's RISController™ family and are in the process of modifying IDT/SIM to achieve compatibility with their boards. The capabilities of IDT/SIM are described in its data sheet and user's manual.

A good number of source files are common to all SIMs; there is absolutely nothing specific to a particular board in these files. There are other files which are common but have parts of code in them which are unique to specific boards - a feature implemented using "#if defined()" or "#ifdef" conditional compilation directives. There is a third variety of files which bear the same name but exist in different directories; this indicates that the files contain code which performs similar tasks for different target boards, but the implementations are so different that conditional compiling would lead to confusion instead of ease of understanding. Finally there are files which are entirely specific only to one particular board. These files have no conditional compile statements, no equivalents in any other subdirectory, and are called for compilation and linking only for one specific SIM for one specific board.

Evaluation boards currently supported are 79S385™, 79RS381™, 79S341™, and 79S460™. Specific "Makefiles" for each board are provided.

From the top-most level of directories, there are 3 main directories - COMMON, SIM3000, SIM4000.
COMMON directory has two subdirectories:

- header - contains common header (#include) files used by all SIMs for all evaluation boards.
- c_asm - contains "C" and "assembler" files which are common to all versions of SIM for all evaluation boards. Most of these files use conditional compiling for different boards.

SIM3000 directory contains source code specific to boards designed with R3000 derivatives in mind. Currently, these boards include 79S385, 79S381, and 79S341. There are a number of subdirectories containing "Makefile"s specific for each evaluation board and possibly different tool-chains. The directory names are suggestive of which tool-chain or which evaluation board the Makefile in that directory supports.

For example, a directory name "_RS385C50" suggests that there is a Makefile in this directory which will create a SIM for the 79S385 board and will use IDT/C™ 5.1 tool-chain for compiling, etc. Directory names are appropriately abbreviated for DOS. In addition to the directories for Makefiles, there is also a "header" directory containing header (#include) files related to R3000-derivative based boards.

After making changes to the source code, the user needs to go into the directory appropriate for the intended target board and tool-chain, and simply run "make" (or "gmake" in case of DOS). All of the object files, and s-record files are built in the same chosen directory. The name(s) of the final product file(s) can be obtained by studying the Makefile(s). Typically, for a board using four ROMs (79S385, 79S381) the file names of the final s-record files are idtmonb0, idtmonb1, idtmonb2, and idtmonb3. For the 79S341 board, the final s-records are in file "idtmon.prm". Each Makefile also creates a version of code which can be run out of RAM on the target board. (the RAM-version). The RAM-version allows the user to debug or test modifications to SIM without actually having to program a new set of ROMs every time a change is made to the source code. The board may contain older version of SIM in its ROM, and the user downloads the newly created RAM-version into the RAM using the "load" command as if the RAM-version were a user application program. Issuing a "go" after the download is completed invokes the new RAM-version SIM.

The SIM4000 directory is similar to the SIM3000 directory; the only difference is that the code pertains to R4000 derivatives. Currently the 79S460 board is supported. The Makefile for this board can be found in the directory "IDTELF64".

Following is a list of global symbols which are used extensively in the source files to achieve conditional compiling for a specific CPU or a specific evaluation board. Please review these symbols in the context of the files you are likely to modify. Conforming to these conditional compiling rules is critical to a successful port of the SIM code to a new board design. Additional symbols can be defined in the Makefiles with "-D" switches and can be used to uniquely identify and support specific features of specific boards in future. Although new global symbols can also be defined in the source files, it is highly recommended that they be defined in the Makefiles to facilitate easy access to software developers other than the creator of the symbols.

CPU_R4000: to identify code specific to R4000 and its derivatives.

CPU_R3000: to identify code specific to R3000 and its derivatives.

The IDT logo is a registered trademark and IDT/SIM, RISController, 79S385, 79S381, 79S341, 79S460, and IDT/C are trademarks of Integrated Device Technology, Inc.

©1996 Integrated Device Technology

R381: to identify code specific to 79RS381 board.
RS341: to identify code specific to 79S341 board.
P4000: to identify code specific to 79S460 board.
INET: to indicate code to be executed only if ethernet support is available on the target board.
PROM: to indicate that networking code is running out of PROM.
IDTSIM: to indicate modifications to industry standard ethernet drivers for IDT-SIM compatibility.
KERNEL: related to ethernet drivers.
XDS: IDT/C 4.1.1 compatibility-specific code modifications. Obsolete.