Preface

This book describes the MIPS R4000 and R4400 family of RISC microprocessors (also referred to in this book as *processor*).

Overview of the Contents

Chapter 1 is a discussion (including the historical context) of RISC development in general, and the R4000 microprocessor in particular.

Chapter 2 is an overview of the CPU instruction set.

Chapter 3 describes the operation of the R4000 instruction execution pipeline, including the basic operation of the pipeline and interruptions that are caused by interlocks and exceptions.

Chapter 4 describes the memory management system including address mapping and address spaces, virtual memory, the translation lookaside buffer (TLB), and the System Control Processor (CP0).

Chapter 5 describes the exception processing resources of R4000 processor. It includes an overview of the CPU exception handling process and describes the format and use of each CPU exception handling register.

Chapter 6 describes the Floating-Point Unit (FPU), a coprocessor for the CPU that extends the CPU instruction set to perform floating-point arithmetic operations. This chapter lists the FPU registers and instructions.

Chapter 7 describes the FPU exception processing.

Chapter 8 describes the signals that pass between the R4000 processor and other components in a system. The signals discussed include the System interface, the Clock/Control interface, the Secondary Cache interface, the Interrupt interface, the Initialization interface, and the JTAG interface.

Chapter 9 describes in more detail the Initialization interface, which includes the boot modes for the processor, as well as system resets.

Chapter 10 describes the clocks used in the R4000 processor, as well as the processor status reporting mechanism.

Chapter 11 discusses cache memory, including the operation of the primary and secondary caches, and cache coherency in a multiprocessor system.

Chapter 12 describes the System interface, which allows the processor access to external resources such as memory and input/output (I/O). It also allows an external agent access to the internal resources of the processor, such as the secondary cache.

Chapter 13 describes the Secondary Cache interface, including read and write cycle timing. This chapter also discusses the interface buses and signals.

Chapter 14 describes the Joint Test Action Group (JTAG) interface. The JTAG boundary scan mechanism tests the interconnections between the R4000 processor, the printed circuit board to which it is mounted, and other components on the board.

Chapter 15 describes the single nonmaskable processor interrupt, along with the six hardware and two software processor interrupts.

Chapter 16 describes the error checking and correcting (ECC) mechanisms of the R4000 processor.

Appendix A describes the R4000 CPU instructions, in both 32- and 64-bit modes. The instruction list is given in alphabetical order.

Appendix B describes the R4000 FPU instructions, listed alphabetically.

Appendix C describes sub-block ordering, a nonsequential method of retrieving data.

Appendix D describes the output buffer and the Δi/Δt control mechanism.

Appendix E describes the passive components that make up the phase-locked loop (PLL).

Appendix F describes Coprocessor 0 hazards.

Appendix G describes the R4000 pinout.

A Note on Style

A brief note on some of the stylistic conventions used in this book: bits, fields, and registers of interest from a software perspective are italicized (such as *Config* register); signal names of more importance from a hardware point of view are rendered in bold (such as **Reset***).

A range of bits uses a colon as a separator; for instance, **(15:0)** represents the 16-bit range that runs from bit 0, inclusive, through bit 15. (In some places an ellipsis may used in place of the colon for visibility: **(15...0)**.)

Preface to the Second Edition

Changes From the First Edition

The second edition of this book incorporates certain low-level changes and technical additions, but retains a substantive identity with the original version.

Changes from the first edition are indicated by left-margin vertical rules.

Getting MIPS Documents On-Line

MIPS documents (including an electronic version of the errata) are available on-line, through the file transport protocol (FTP). To retrieve them, follow the steps below. The text you are to type is shown in **`Courier Bold`** font; the computer's responses are in shown in `Courier Regular` font.

1. First, place yourself in the directory on your system within which you want to store the retrieved files. Do this by typing:

   `cd <directory_you_want_file_to_be_in>`

2. Access the MIPS document server, **sgigate**, through FTP by typing:

   `ftp sgigate.sgi.com`

3. The server tells you when you are connected for FTP by responding:

   `Connected to sgigate.sgi.com.`

4. Next (after some announcements) the server asks you to log in by requesting a name and then a password.

   `Name (sgigate.sgi.com:<login_name>):`

5. Login by typing **anonymous** for your name and your **electronic mail address** for your password.

   ```
   Name (sgigate.sgi.com:<login_name>): anonymous
   331 Guest login ok, type your name as
        password.
   Password: your_email_address
   ```

6. The system indicates you have successfully logged in by supplying an FTP prompt:

   `ftp>`

7. Go to the pub/doc directory by typing:

   `ftp> cd pub/doc`

8. You can take a look at the contents of the **doc** directory by listing them:

   `ftp> ls`

9. You will find several R4000-related subdirectories, such as R4200, R4400, and R4600. When you find the subdirectory you want, **cd** into that subdirectory and retrieve the file you want by typing:

   `get <filename>`

   This copies the file from **sgigate** back to your system.

10. When you have retrieved the files you want, exit from **ftp** by typing:

    `ftp> quit`

11. If the file was encoded for transmission, you must decode it, after retrieval, by typing:

    `uudecode <filename>`

12. If the file was compressed for transmission, you must uncompress it, after retrieval, by typing:

    `uncompress <filename>`

13. If you tarred the file, type:

    `tar xvof <filename>`