supercomputer: *Any machine still on the drawing board.*

Stan Kelly-Bootle, *The Devil's DP Dictionary*, 1981

Historical Perspective and Further Reading

This section discusses the history of the first pipelined processors, the earliest superscalars, and the development of out-of-order and speculative techniques, as well as important developments in the accompanying compiler technology.

It is generally agreed that one of the first general-purpose pipelined computers was Stretch, the IBM 7030 (Figure 4.16.1). Stretch followed the IBM 704 and had a goal of being 100 times faster than the 704. The goals were a "stretch" of the state of the art at that time—hence the nickname. The plan was to obtain a factor of 1.6 from overlapping fetch, decode, and execute by using a four-stage pipeline. Apparently, the rest was to come from much more hardware and faster logic. Stretch was also a training ground for both the architects of the IBM 360, Gerrit Blaauw and Fred Brooks, Jr., and the architect of the IBM RS/6000, John Cocke.

FIGURE 4.16.1 The Stretch computer, one of the first pipelined computers.

Control Data Corporation (CDC) delivered what is considered to be the first supercomputer, the CDC 6600, in 1964 (Figure 4.16.2). The core instructions of Cray's subsequent computers have many similarities to those of the original CDC 6600. The CDC 6600 was unique in many ways. The interaction between pipelining and instruction set design was understood, and the instruction set was kept simple to promote pipelining. The CDC 6600 also used an advanced packaging technology. James Thornton's book [1970] provides an excellent description of the entire computer, from technology to architecture, and includes a foreword by Seymour Cray. (Unfortunately, this book is currently out of print.) Jim Smith, then working at CDC, developed the original 2-bit branch prediction scheme and explored several techniques for enhancing instruction issue. Cray, Thornton, and Smith have each won the ACM Eckert-Mauchly Award (in 1989, 1994, and 1999, respectively).

The IBM 360/91 introduced many new concepts, including dynamic detection of memory hazards, generalized forwarding, and reservation stations (Figure 4.16.3). The approach is normally named *Tomasulo's algorithm*, after an engineer who worked on the project. The team that created the 360/91 was led by Michael Flynn, who was given the 1992 ACM Eckert-Mauchly Award, in part for his contributions to the IBM 360/91; in 1997, the same award went to Robert Tomasulo for his pioneering work on out-of-order processing.

The internal organization of the 360/91 shares many features with the Pentium III and Pentium 4, as well as with several other microprocessors. One major

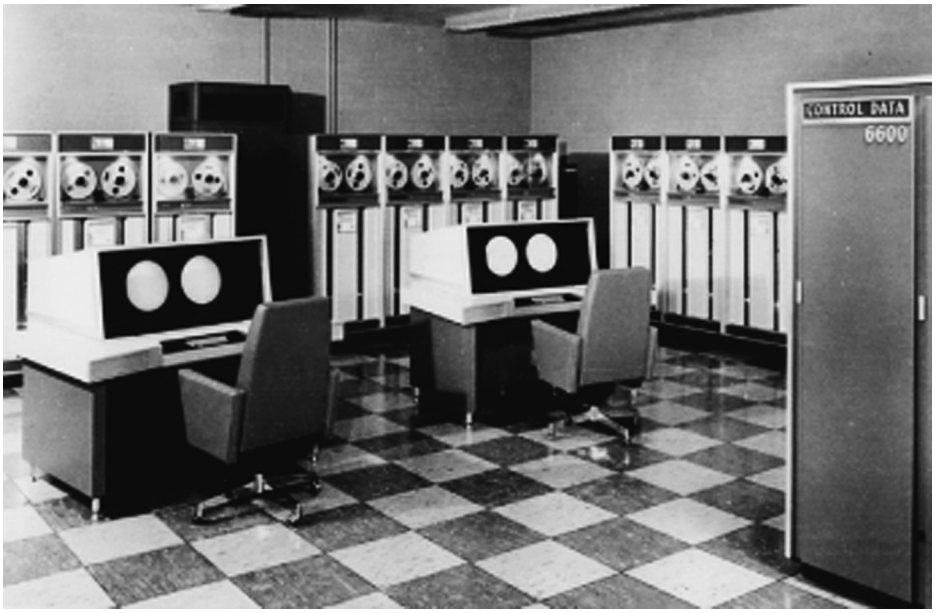

FIGURE 4.16.2 The CDC 6600, the first supercomputer.

FIGURE 4.16.3 The IBM 360/91 pushed the state of the art in pipelined execution when it was unveiled in 1966.

difference was that there was no branch prediction in the 360/91 and hence no speculation. Another major difference was that there was no commit unit, so once the instructions finished execution, they updated the registers. Out-of-order instruction commit led to *imprecise interrupts*, which proved to be unpopular and led to the commit units in dynamically scheduled pipelined processors since that time. Although the 360/91 was not a success, its key ideas were resurrected later and exist in some form in the majority of microprocessors of the last decade.

Improving Pipelining Effectiveness and Adding Multiple Issue

The RISC processors refined the notion of compiler-scheduled pipelines in the early 1980s. The concepts of delayed branches and delayed loads—common in microprogramming—were extended into the high-level architecture. In fact, the Stanford processor that led to the commercial MIPS architecture was called "Microprocessor without Interlocked Pipelined Stages" because it was up to the assembler or compiler to avoid data hazards.

In addition to its contribution to the development of the RISC concepts, IBM did pioneering work on multiple issue. In the 1960s, a project called ACS was underway. It included multiple-instruction issue concepts and the notion of integrated compiler and architecture design, but it never reached product stage. The earliest proposal for a superscalar processor that dynamically makes issue decisions was

by John Cocke; he described the key ideas in several talks in the mid-1980s and, with Tilak Agarwala, coined the name *superscalar*. This original design was a two-issue machine named Cheetah, which was followed by a more widely discussed four-issue machine named America. The IBM Power-1 architecture, used in the RS/6000 line, is based on these ideas, and the PowerPC is a variation of the Power-1 architecture. Cocke won the Turing Award, the highest award in computer science and engineering, for his architecture work.

Static multiple issue, as exemplified by the *long instruction word* (LIW) or sometimes *very long instruction word* (VLIW) approaches, appeared in real designs before the superscalar approach. In fact, the earliest multiple-issue machines were special-purpose attached processors designed for scientific applications. Culler Scientific and Floating Point Systems were two of the most prominent manufacturers of such computers. Another inspiration for the use of multiple operations per instruction came from those working on microcode compilers. Such inspiration led to a research project at Yale led by Josh Fisher, who coined the term VLIW. Cydrome and Multiflow were two early companies involved in building mini-supercomputers using processors with multiple-issue capability. These processors, built with bit-slice and multiple-chip gate array implementations, arrived on the market at the same time as the first RISC microprocessors. Despite some promising performance on high-end scientific codes, the much better cost/performance of the microprocessor-based computers doomed the first generation of VLIW computers. Bob Rau and Josh Fisher won the Eckert-Mauchly Award in 2002 and 2003, respectively, for their contributions to the development of multiple processors and software techniques to exploit ILP.

The very beginning of the 1990s saw the first superscalar processors using static scheduling and no speculation, including versions of the MIPS and PowerPC architectures. The early 1990s also saw important research at a number of universities, including Wisconsin, Stanford, Illinois, and Michigan, focused on techniques for exploiting additional ILP through multiple issue with and without speculation. These research insights were used to build dynamically scheduled, speculative processors, including the Motorola 88110, MIPS R10000, DEC Alpha 21264, PowerPC 603, and the Intel Pentium Pro, Pentium III, and Pentium 4.

In 2001, Intel introduced the IA-64 architecture and its first implementation, Itanium. Itanium represented a return to a more compiler-intensive approach that they called EPIC. EPIC represented a considerable enhancement over the early VLIW architectures, removing many of their drawbacks. It has had modest sales. In 2013, the IA-64 architecture is used only in low-volume, high-end servers and is outnumbered by x86 processors by more than 100:1.

Compiler Technology for Exploiting ILP

Successful development of processors to exploit ILP has depended on progress in compiler technology. The concept of loop-unrolling was understood early, and a number of companies and researchers—including Floating Point Systems, Cray, and

the Stan ford MIPS project—developed compilers that made use of loop-unrolling and pipeline scheduling to improve instruction throughput. A special purpose processor called WARP, designed at Carnegie Mellon University, inspired the development of software pipelining, an approach that symbolically unrolls loops.

To exploit higher levels of ILP, more aggressive compiler technology was needed. The VLIW project at Yale developed the concept of trace scheduling that Multiflow implemented in their compilers. Trace scheduling relies on aggressive loop unrolling and path prediction to compile favored execution traces efficiently. The Cydrome designers created early versions of predication and support for software pipelining. Hwu at Illinois worked on extended versions of loop-unrolling, called *superblocks*, and techniques for compiling with predication. The concepts from Multiflow, Cydrome, and the research group at Illinois served as the architectural and compiler basis for the IA-64 architecture.

Further Reading

Bhandarkar, D. and D. W. Clark [1991]. "Performance from architecture: Comparing a RISC and a CISC with similar hardware organizations," *Proc. Fourth Conf. on Architectural Support for Programming Languages and Operating Systems,* IEEE/ACM (April), Palo Alto, CA, 310–19.

A quantitative comparison of RISC and CISC written by scholars who argued for CISCs as well as built them; they conclude that MIPS is between 2 and 4 times faster than a VAX built with similar technology, with a mean of 2.7.

Fisher, J. A. and B. R. Rau [1993]. *Journal of Supercomputing* (January), Kluwer.

This entire issue is devoted to the topic of exploiting ILP. It contains papers on both the architecture and software and is a wonderful source for further references.

Hennessy, J. L. and D. A. Patterson [2001]. *Computer Architecture: A Quantitative Approach*, fourth edition, Morgan Kaufmann, San Francisco.

Chapter 2 and Appendix A go into considerably more detail about pipelined processors (almost 200 pages), including superscalar processors and VLIW processors. Appendix G describes Itanium.

Jouppi, N. P. and D. W. Wall [1989]. "Available instruction-level parallelism for superscalar and superpipelined processors," *Proc. Third Conf. on Architectural Support for Programming Languages and Operating Systems,* IEEE/ACM (April), Boston, 272–82.

A comparison of deeply pipelined (also called superpipelined) and superscalar systems.

Kogge, P. M. [1981]. *The Architecture of Pipelined Computers,* McGraw-Hill, New York.

A formal text on pipelined control, with emphasis on underlying principles.

Russell, R. M. [1978]. "The CRAY-1 computer system," *Comm. of the ACM* 21:1 (January), 63–72.

A short summary of a classic computer that uses vectors of operations to remove pipeline stalls.

Smith, A. and J. Lee [1984]. "Branch prediction strategies and branch target buffer design," *Computer* 17:1 (January), 6–22.

An early survey on branch prediction.

Smith, J. E. and A. R. Plezkun [1988]. "Implementing precise interrupts in pipelined processors," *IEEE Trans. on Computers* 37:5 (May), 562–73.

Covers the difficulties in interrupting pipelined computers.

Thornton, J. E. [1970]. *Design of a Computer. The Control Data 6600,* Glenview, IL: Scott, Foresman.

A classic book describing a classic computer, considered the first supercomputer.