intel

Development Toolset

SPECIFICS

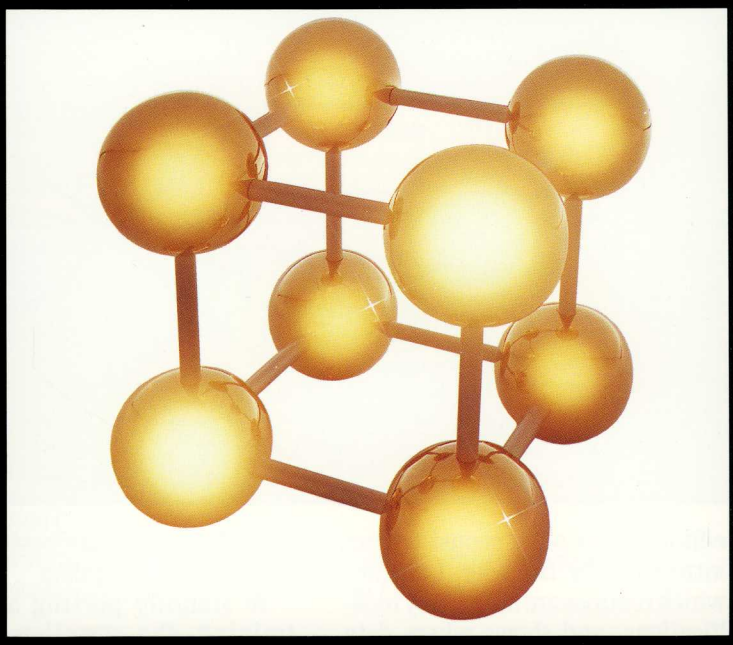
COMPREHENSIVE TOOLS FOR PERFORMANCE AND PRODUCTIVITY

Parallel supercomputers, such as Intel's iPSC®/860, offer unmatched performance today and even higher performance tomorrow. But to realize that performance, new applications must be written and older ones restructured to take advantage of the underlying parallel architecture. To help programmers with that task, Intel's iPSC/860 offers a comprehensive family of high-performance cross-development tools. Intel's parallel development toolset includes state-of-the-art CASE tools for parallel programming, graphical performance analysis tools to take the guesswork out of performance tuning, and sophisticated optimizing compilers that exploit the power of the iPSC/860's pipelined superscalar node architecture to produce top run-time performance. In addition, the development environment offers a variety of industry-standard UNIX, C and FORTRAN tools. The iPSC/860 software development toolset supports the entire development process (see Figure 1).

To promote cost-effective application development, Intel offers cross-development tools for UNIX workstations, with Ethernet and TCP/IP access to the iPSC/860. This arrangement simplifies user training, enhances programmer productivity and frees iPSC/860 resources for user applications. Supported environments include the Sun/3 and Sun/4 workstations and the iPSC/860's 80386-based System Resources Manager (SRM).

PARALLEL CASE TOOLS SIMPLIFY RESTRUCTURING

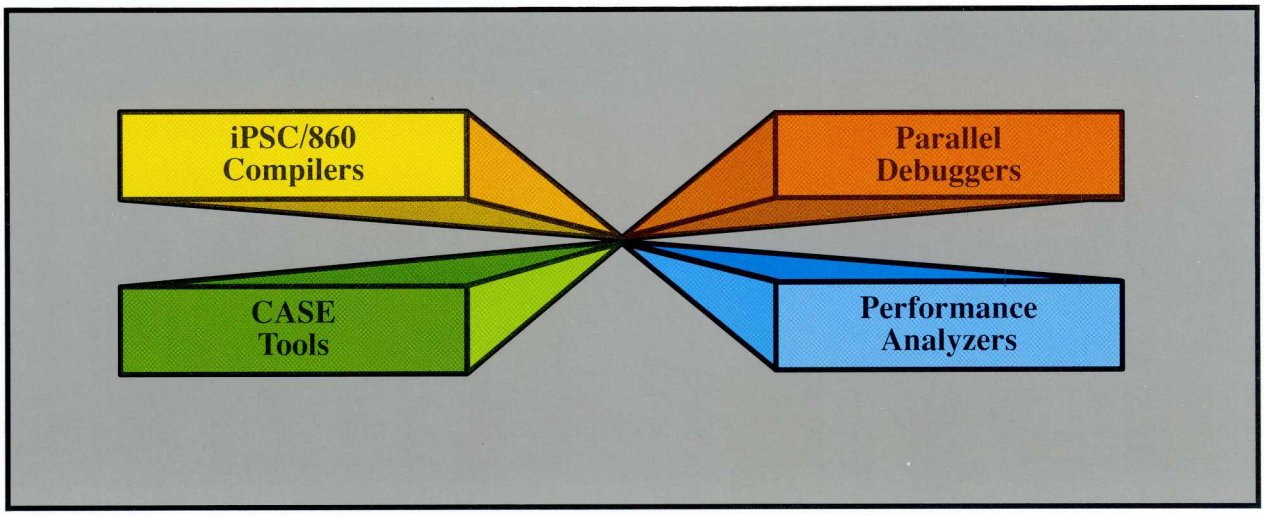
Many applications for the iPSC/860 begin as large programs written for sequential computers. Restructuring these applications can require code changes in many sections of the program—code changes that may involve tracing variable usage and control flow through many lines of code and often imposing a significant clerical burden for the programmer. The iPSC/860's parallel CASE tools automate these clerical functions, freeing programmers to concentrate on high-level analysis and parallelization strategies.



Intel's innovative technology has made the supercomputer a practical and affordable tool.

FORGE® gives FORTRAN programmers a powerful analytical tool for managing and understanding program content and structure. FORGE parses FORTRAN code to produce a database for the application showing all variable usage and data dependencies. Global data flow analysis provides lists of all variables, constants, subroutines, parameters and COMMON blocks, enabling programmers to manage a large application as a single entity. FORGE also includes scalar and array section analysis and a DO loop analyzer. Using FORGE's intuitive, graphical interface, programmers can trace variable and subroutine usage over thousands of lines of code and quickly see where changes are needed. After the code is written or adapted, FORGE provides a program maintenance facility for supporting large applications.

CAST™—the Concurrent Applications Structuring Tool—is an enhancement to FORGE that offers user-directed restructuring and parallelization as well as the program management and analysis features of FORGE. CAST speeds parallel code adaptation and development by allowing users to direct the parallel strategy interactively. Using CAST's intelligent editor, programmers can manipulate data structures and create parallel FORTRAN structures from existing sequential program structures. The



editor tracks any inconsistencies introduced by the user, indicates which routines are affected by modifications, and shows where data must be supplied to or by a new routine. The new structure representation is then transformed via a code generator, producing source code that can be compiled by the iPSC/860 FORTRAN compiler.

OPTIMIZING COMPILERS PROMOTE SPEED

For the critical step of compilation, the iPSC/860 C and FORTRAN cross-compilers offer high run-time execution speed and fast compilations. Both compilers use sophisticated optimization techniques to ensure that applications fully exploit the superscalar architecture of the i860 microprocessor. Among supported optimizations are:

- Vectorization
- Procedure In-lining
- Cache Utilization Strategies
- Software Pipelining
- i860 Dual Instruction Mode
- i860 Dual Operation Mode
- Global Optimization
- Local Optimization
- Loop Optimizations

The compilers' internal vectorizer can exploit more vector processing opportunities than vector preprocessors. When combined with the software pipeliner, the internal vectorizer can further utilize the pipelined i860 floating point units and also optimize cache management. As a result, the iPSC/860 optimizing compilers achieve run-time speeds that are up to three times faster than are possible with conventional compilers.

To simplify porting and user training, the compilers meet a broad range of industry standards. iPSC/860 C conforms to the proposed ANSI C standard (X3JII), and iPSC/860 FORTRAN is a true superset of FORTRAN77 that includes VAX/VMS 4.0 and IBM/VS FORTRAN extensions. The compilers support IEEE-compliant floating-point run-time libraries, as well as faster non-IEEE floating-point libraries. The compilers also provide language-compatible libraries for iPSC/860 message-passing and file system services.

UNIX DEVELOPMENT TOOLS ASSIST IN CODE MANAGEMENT

Included with the cross compilers are UNIX-style tools including enhanced versions of the i860 assembler and linker, to process the compiler output. For object code management, the environment provides a librarian and the UNIX tools nm, size, strip and dump.

PARALLEL DEBUGGER GETS IT ALL WORKING

Once the object code is ready, programmers can use the iPSC/860 Interactive Parallel Debugger, a source-level debugger, to get their applications running smoothly. Along with the conventional debugger features, the Interactive Parallel Debugger includes special support for debugging message-passing communications and for debugging the synchronization of multiple independent processes.

Figure 1: *Intel's advanced Development Toolset includes a comprehensive set of software productivity tools.*

PERFORMANCE ANALYSIS TOOLS HELP THE TUNE-UP

For the performance-tuning phase, Intel provides a multiprocessor version of the UNIX prof profiling tool and a sophisticated, graphically-oriented Parallel Performance Analysis Tool (PAT). Prof profiles each node's CPU usage, making it easy to identify opportunities for performance tuning.

The Parallel Performance Analysis Tool produces CPU usage histograms, event tracing of multiple processes and internode communications tracing, helping programmers spot opportunities for greater parallel processing efficiency. PAT's graphical interface uses timelines, graphs and bar charts to give programmers greater insight into their applications. The performance tools support both C and FORTRAN.

iPSC/860 SIMULATOR PROVIDES FLEXIBILITY

In the early stages of code development, or for training purposes, programmers may find it useful to run iPSC/860 codes on a local UNIX workstation or system. For example, a programmer might want to develop parallel code even when an iPSC/860 system is unavailable. Or, the supercomputer might be too heavily utilized to support fast turnaround for test runs or classroom examples. To meet these needs, the iPSC/860 software environment includes a simulator that emulates the message-passing and application-level environment of the iPSC/860 on a standard UNIX host. The simulator provides the functionality of the target system, but at a reduced performance level sufficient for quick-running jobs. The iPSC/860 's development toolset provides a powerful and efficient environment that allows programmers to work productively—and to produce applications that reap the performance rewards offered by Intel's state-of-the-art parallel supercomputing architecture.

Specifications

iPSC/860 FORTRAN

FORTRAN-77 as defined by *American National Standard Programming Language FORTRAN, ANSI x3.9-1978.* There are no deviations from this standard. Supported extensions:
IBM/VS FORTRAN extensions
VAX/VMS FORTRAN extensions except for:
REAL*16 OPTION ISAM Files %DESCR VOLATILE
RECORDTYPE in OPEN 128-bit constants or variables

Conformance Tests for FORTRAN:

NTIS Suite	Whetstones	Linpack
NAS kernels	SPICE (of SPEC suite)	NBS FORTRAN Suite
Livermore Loops	Elefunt	PERFECT Club

iPSC/860 C

Conforms to *Draft, Proposed American National Standard for Information Systems—Programming Language C.*

Conformance Tests for C:

Plum-Hall C Validation Suite	gnu cc
Paranoia	Perennial Test Suite

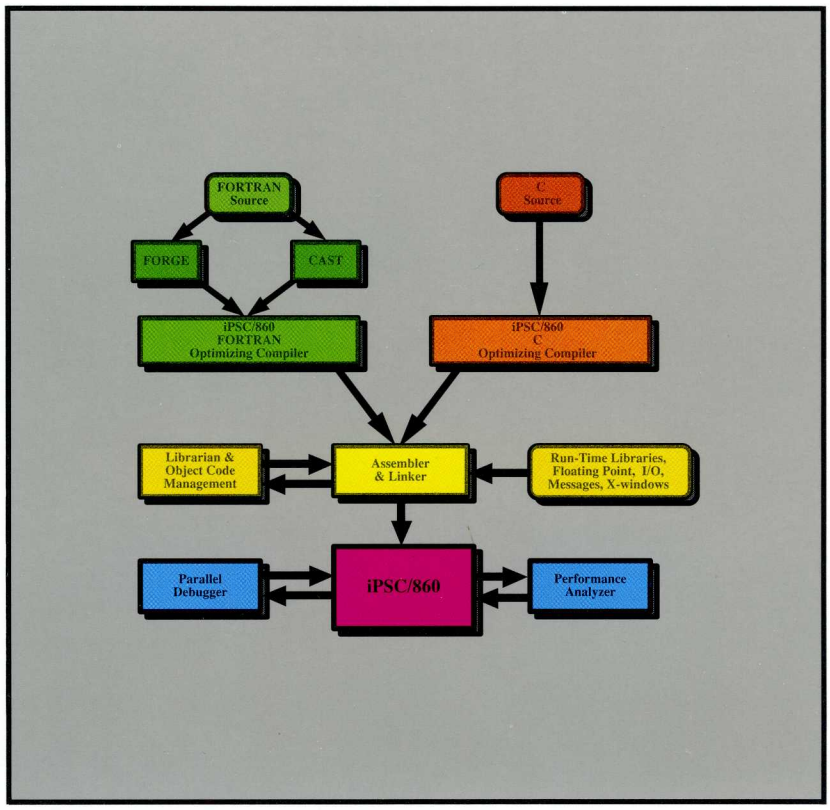


Figure 2: *iPSC/860 program development is cost-effective and efficient.*

Intel Corporation
Supercomputer Systems Division
15201 NW Greenbrier Pkwy
Beaverton, OR 97006
(503) 629-7600