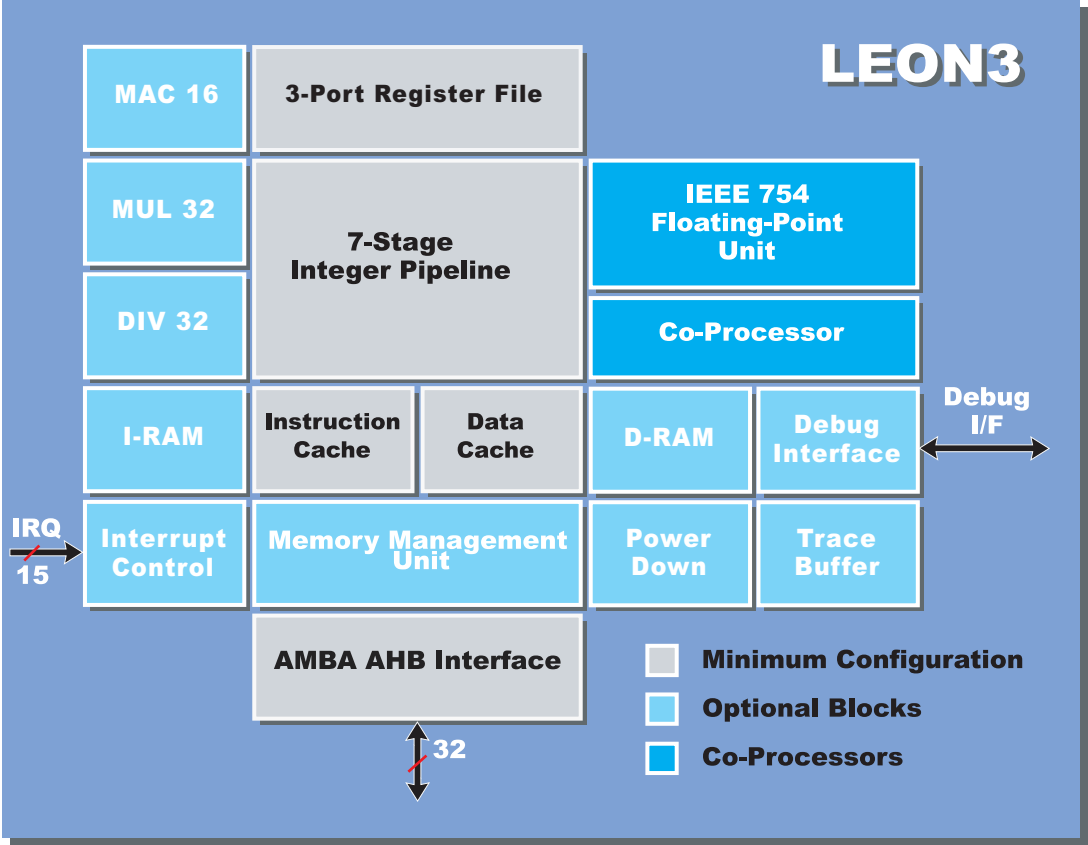


## LEON3 32-BIT PROCESSOR CORE

The LEON3 processor core is a synthesizable VHDL model of a 32-bit processor compliant with the SPARC V8 architecture. The core is highly configurable and particularly suitable for system-on-a-chip (SOC) designs. The configurability allows designers to optimize the processor for performance, power consumption, I/O throughput, silicon area and cost. The core is interfaced using the AMBA 2.0 AHB bus and supports the IP core plug&play method provided in the Gaisler Research IP library (GRLIB). The processor can be efficiently implemented on FPGA and ASIC technologies and uses standard synchronous memory cells for caches and register file. The LEON3 core is also available in a fault-tolerant version immune to single event upsets (SEU) for use in space and other high-reliability applications. To promote the SPARC architecture and simplify early evaluation and prototyping, the processor and the associated IP library are provided in full source code under the GNU GPL license for evaluation, research and educational purposes. A low cost commercial license is available for commercial applications.



The LEON3 processor is highly configurable, allowing the designers to select and configure functional blocks to optimize the processor for a specific application.

## LEON3 architecture and features

LEON3 is a 32-bit processor based on the SPARC V8 architecture. It implements an advanced 7-stage pipeline with separate instruction and data cache buses (Harvard architecture). The processor supports the full SPARC V8 instruction set, including the MUL, MAC and DIV instructions. An optional IEEE-754 floating-point unit (FPU) provides support for both single- and double-precision floating-point operations. The cache system supports multi-set caches with up to 4 ways per cache, 256 kbyte per way and a choice of LRU, LRR or random replacement policy.

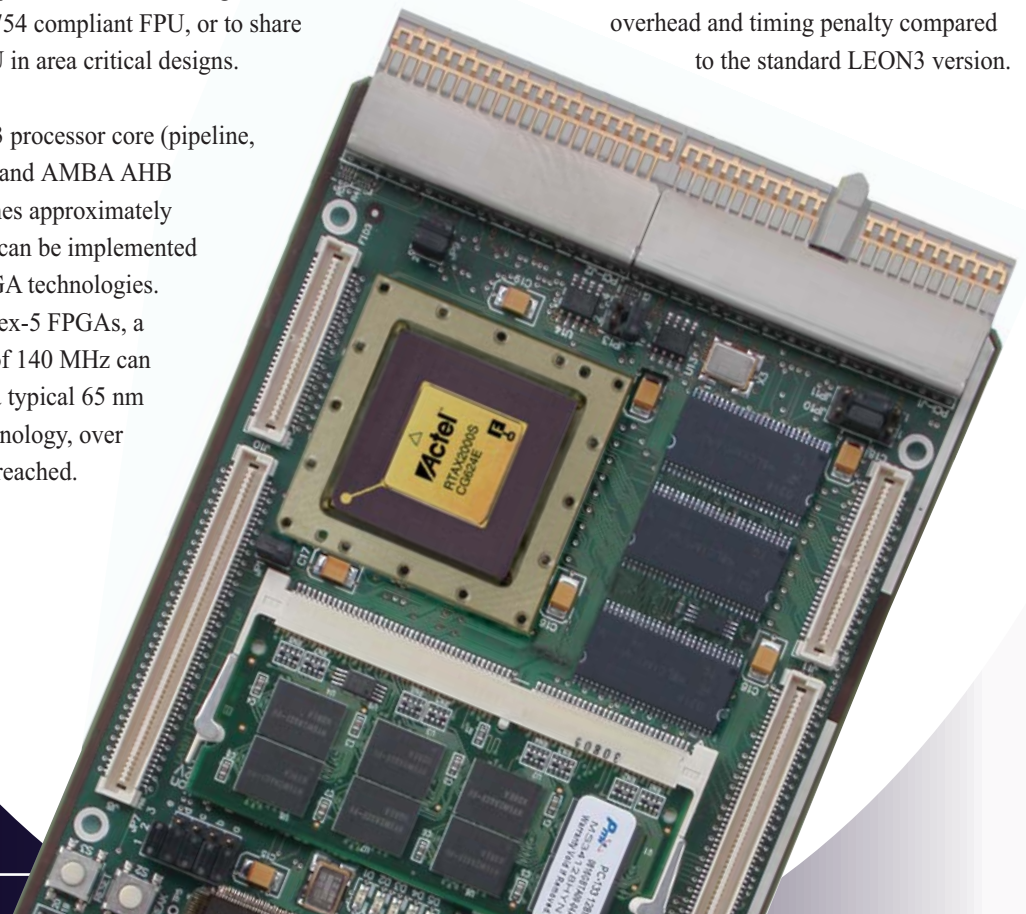
LEON3 can be utilized in asymmetric multiprocessing (AMP) or synchronous multiprocessing (SMP) configurations. Up to 16 processor cores can be implemented in a system, with a typical four-processor system capable of delivering up to 1600 Dhrystone MIPS of performance. The processor provides hardware support for cache coherency, processor enumeration and SMP interrupt steering. A unique debug interface allows non-intrusive hardware debugging of both single- and multiple-processor systems, and provides access to all on-chip registers and memory. Trace buffers for both instructions and AMBA bus traffic are provided. An AMBA round-robin arbiter provides fair bus utilization for the processors. Each processor can be configured to use a separate IEEE-754 compliant FPU, or to share one common FPU in area critical designs.

The basic LEON3 processor core (pipeline, cache controllers and AMBA AHB interface) consumes approximately 25,000 gates and can be implemented on ASIC and FPGA technologies. Using Xilinx Virtex-5 FPGAs, a clock frequency of 140 MHz can be achieved. On a typical 65 nm standard-cell technology, over 650 MHz can be reached.

## LEON3 Fault Tolerant version for aerospace, military and critical applications

LEON3 is the successor of the ERC32 and LEON2 processors developed for the European Space Agency (ESA). The ERC32 and LEON2 devices are currently used in European and international space programmes. The experience from the previous processor developments has been used to design the Fault Tolerant version of LEON3 (LEON3FT). The LEON3FT core is used in SOC developments initiated by ESA and has been incorporated amongst others in the UT699RH device from Aeroflex and the RTAX FPGA devices from Actel.

LEON3FT is fault tolerant and Single Event Upset (SEU) immune by design, protecting all on-chip memories and registers. For space applications this is a mandatory requirement due to the severe radiation environment. As the manufacturing technology is scaled towards finer geometries the probability for SEUs is increasing. With present technology (130 -180 nm) this phenomena must be addressed at high altitudes (commercial and military aircraft). For newer technologies (45-65 nm) these effects will occur on ground level and need to be handled in critical applications such as medical equipment and transportation. The fault tolerance of LEON3FT has been implemented with a minimum of gate count overhead and timing penalty compared to the standard LEON3 version.



## GRLIB IP library improves SOC design by reducing development time and cost

To achieve optimum performance and minimum cost for a SOC design, it is mandatory to reuse existing IP cores and at the same time be able to configure these cores for the specific application. The overall concept of the GRLIB IP library is to provide a standardized and vendor-independent infrastructure for IP cores.

Integrating third-party IP cores from different suppliers can require significant adaptation and harmonization of both functional and logistical (tool-oriented) interfaces. The GRLIB IP library enhances the development of SOC devices by providing reusable IP cores with common functional and logistical interfaces.

A key aspect to IP core integration is to ensure vendor independence at both core and tool level. The GRLIB IP library is designed to be easily portable to different CAD tools and target technologies. It does not depend on any vendor specific interface or technology that need to be licensed or procured. It is designed to allow contributions or extensions from other parties.

GRLIB is designed to be "bus-centric", i.e. it is assumed that most of the IP cores are connected through an on-chip bus (AMBA 2.0 AHB or APB buses). Plug&Play information from the IP cores connected to the AMBA buses is available to simplify software development and debugging.

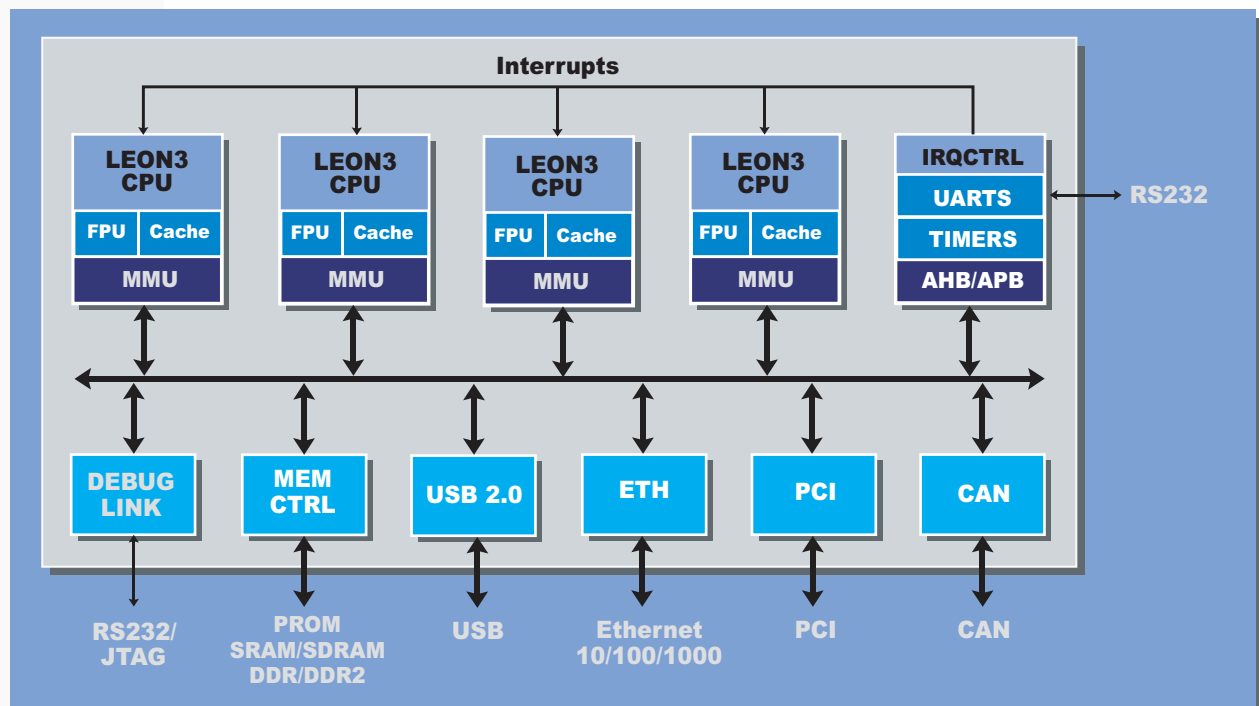
## Software environment based on standard development tools and operating systems

The software development environment for LEON3 consists of a range of popular open-source and commercial tools and embedded operating systems. The operating systems range from very small footprint task handlers to full featured Real-Time Operating Systems (RTOS).

The VxWorks SPARC port supports LEON3 and provides drivers for standard on-chip peripherals. The port supports both non-MMU and MMU systems allowing users to program fast and secure applications. Along with the graphical Eclipse based workbench comes the extensive VxWorks documentation.

The ThreadX SPARC port supports LEON3 and its standard on-chip peripherals. ThreadX is an easy to learn and understand advanced pico-kernel real-time operating system designed specifically for deeply embedded applications. ThreadX has a rich set of system services for memory allocation and threading.

Nucleus is a real time operating system developed by Mentor Graphics that offers a rich set of features in a scalable and configurable manner. The SPARC port includes BSP and drivers for all standard LEON3 on-chip peripherals.



Example system-on-a-chip design with multiple LEON3 processor cores and other IP cores from GRLIB

Linux support for MMU based LEON3 systems is furnished through a special version of the SnapGear Embedded Linux distribution, which is provided as a full source package containing kernel, libraries and application code for rapid development of embedded SPARC systems. In addition, support for non-MMU LEON3 systems is provided through  $\mu$ CLinux.

The RTEMS SPARC port supports LEON3 and its standard on-chip peripherals. RTEMS is a real-time kernel that has standard interfaces such as pthreads, POSIX and driver models making it is easy to port applications and drivers between operating systems.

The eCos real-time operating system SPARC port supports LEON3 and its standard on-chip peripherals. The highly configurable nature of eCos allows the operating system to be customized to precise application requirements, delivering the best possible run-time performance and an optimized footprint.

The above operating systems and kernels, except for VxWorks, Nucleus and ThreadX, are based on a common open-source GNU-based cross-compilation system which includes the GNU C/C++ cross-compiler, assembler, linker, Newlib embedded C-library, boot-PROM builder, Eclipse based IDE, etc. It also provides a simple bare-C run-time system with tasking and interrupt support, and an additional pthreads library.

## Full LEON3 simulator TSIM/GRSIM, including modules for GRLIB cores

TSIM and GRSIM are SPARC architecture simulators capable of emulating LEON3 based processor systems. TSIM can be run in stand-alone mode or connected through a network socket to the GNU gdb debugger. In stand-alone mode, a variety of debugging commands are available to allow manipulation of memory contents and registers, breakpoint / watchpoint insertion and performance measurement. Connected to gdb, TSIM acts as a remote target and supports all gdb debug requests. The communication between gdb and TSIM is performed using the gdb extended remote protocol. Any third-party debugger supporting this protocol can be used, e.g. DDD.

### Features

- Cycle-true emulation of LEON3 processors
- Superior performance: up to +25 MIPS
- Accelerated processor standby mode
- Connection to GNU debugger (gdb)
- 64-bit time for unlimited simulation periods
- Loadable modules for user-defined I/O devices
- Instruction trace buffer
- Stack backtrace with symbolic information
- Non-intrusive execution time profiling
- Unlimited number of breakpoints and watchpoints
- Save and restore complete simulator state
- Linkable library
- Multiple processor support (GRSIM)

The screenshot shows the DDD (Data Display Debugger) interface. The main window displays a C program with the following code:

```

int i;
fixed = 0.0;
floated = 0.0;
printf ("Starting \n");
/* rewrite (output); */
printf (" Perm");
timer = Getclock ();
Perm ();
xtimes[1] = Getclock () - timer;
fixed = fixed + permbase * xtimes[1];
floated = floated + permbase * xtimes[1];
printf (" Towers");
timer = Getclock ();
Towers ();
xtimes[2] = Getclock () - timer;
fixed = fixed + towersbase * xtimes[2];
floated = floated + towersbase * xtimes[2];
printf (" Queens");
timer = Getclock ();
Queens ();
xtimes[3] = Getclock () - timer;
fixed = fixed + queensbase * xtimes[3];
floated = floated + queensbase * xtimes[3];
printf (" Intrm");
timer = Getclock ();

```

The interface also shows a memory dump, registers, a backtrace, and performance statistics. The backtrace shows the following stack frames:

```

#4 0x2b000008 in ?? ()
#3 0x0000000c in ?? ()
#2 0x40001020 in _hardreset ()
#1 0x40001254 in zerobss ()
#0 0x40000800 in text_start ()

```

The performance statistics are as follows:

```

Program exited normally.
tsim> per

Cycles      : 32841271
Instructions: 19106376
Overall CPI : 1.72

CPU performance (400.0 Mhz) : 232.71 MDPS (228.21 MIPS, 4.50 MFLOPS)
Cache hit rate              : 95.2 % (99.4 / 75.0)
AHB bandwidth utilisation  : 16.3 % ( 2.4 / 13.9)
Simulated time              : 82.10 ms
Processor utilisation        : 100.00 %
Real-time performance       : 3.96 %
Simulator performance       : 9205.60 KIPS
Used time (sys + user)      : 2.08 s

```

The TSIM simulator can be controlled for example from DDD via the gdb extended remote protocol

## Extensive debug monitor, GRMON for quick hardware and software validation

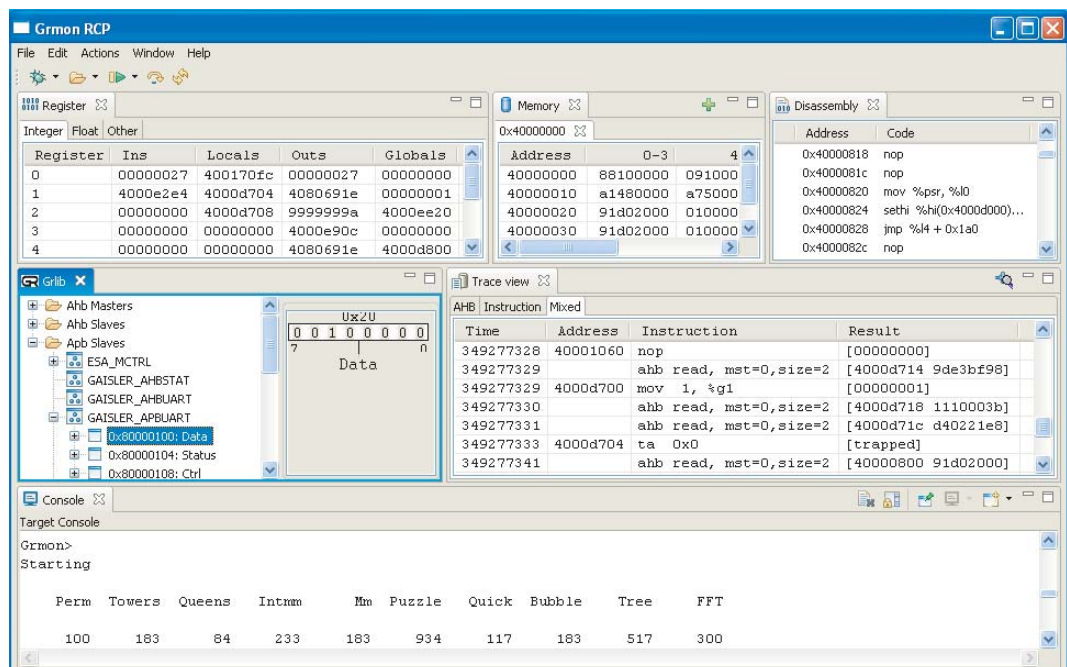
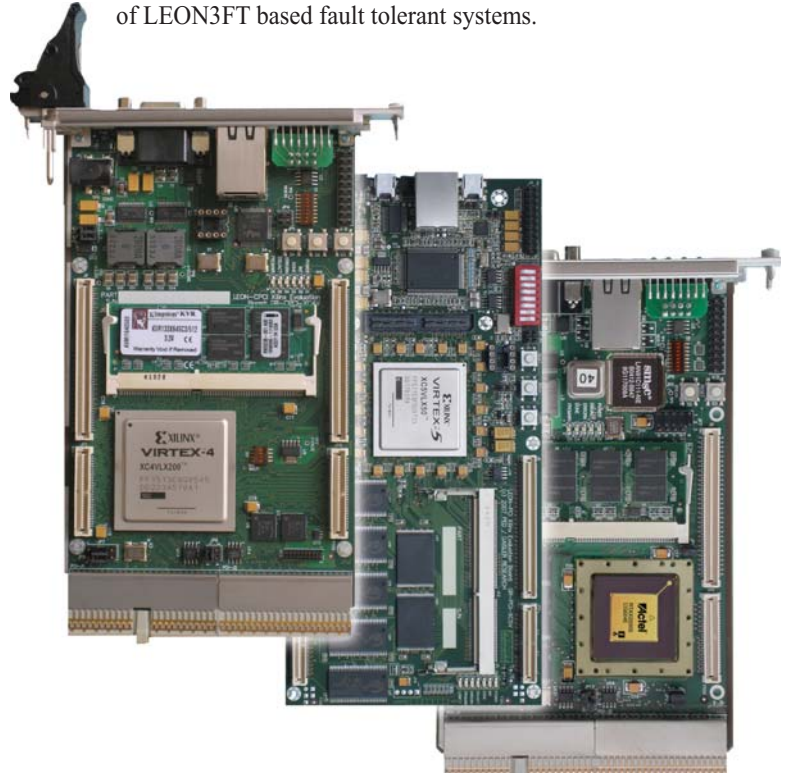
GRMON is a debug monitor for the LEON3 Debug Support Unit (DSU), providing a non-intrusive debug environment on real target hardware. The DSU can be controlled through any AMBA AHB master and GRMON supports communication through a large number of interfaces. It can operate in stand-alone mode or attached to the GDU debugger (gdb). In stand-alone mode, LEON3 applications can be loaded and debugged using a command line interface or via a graphical user interface. Numerous commands are available to examine data, insert breakpoints and advance execution. When attached to gdb, GRMON acts as a gateway and translates the gdb extended remote protocol to debug commands on the target. The GrmonRCP graphical user interface for GRMON is based on the Eclipse Rich Client Platform (RCP). It provides a comprehensive and customizable interface to all GRMON functions.

### Features

- Read/write access to all registers and memory
- Downloading and execution of LEON3 applications
- Disassembler and trace buffer management
- Breakpoint and watchpoint management
- Command line mode or graphical user interface
- Remote connection to GNU debugger (gdb)
- Auto-probing and initialization of LEON3 peripherals and memory settings
- Supported debug interfaces: PCI, USB, Ethernet, JTAG, UART, SpaceWire with RMAP

## LEON3 boards for fast prototyping and evaluation

Several FPGA boards are available to support early development and fast prototyping of LEON3 systems. The boards are provided with volatile and non-volatile memory, as well as serial, USB and Ethernet interfaces. This makes them ideal for implementing LEON3 designs. The boards are based on Xilinx or Actel FPGA devices and are capable of operating stand-alone or attached to PCI / CPCI backplanes. The boards support the implementation of LEON3FT based fault tolerant systems.



## Applications

### Ground-based applications (LEON3)

- Automotive
- Multimedia (MP3, DVD players etc.)
- GPS receivers
- Set-top boxes (satellite receivers etc.)
- General SOC platforms
- Wireless and mobile phone applications

The LEON3 processor IP core is highly configurable and is suitable for system-on-a-chip (SOC) designs. The processor combines high performance and an advanced architecture with low gate count and low power consumption. Implementing the SPARC V8 architecture (IEEE-1754), the LEON3 processor offers a truly open and well-supported instruction set.

### Aerospace applications (LEON3FT)

- Avionics Equipment
- Data Handling Systems
- Attitude and Orbit Control Systems
- Instrumentation Control
- Payload Control
- General-purpose radiation-hardened processor

The LEON3FT processor implements unique fault-tolerance features that allow it to function correctly also in the severe space environment.

The processor includes on-chip error-detection and error-correction logic to detect and remove any soft error caused by cosmic radiation.

Technology	MHz	Area
TSMC 65 nm ASIC (std-cell)	650 MHz	25 kgates
UMC 130 nm ASIC (std-cell)	400 MHz	25 kgates
Virtex-5 FPGA	140 MHz	3,500 LUT
Virtex-4 FPGA	125 MHz	3,500 LUT
Altera Stratix-3 FPGA	150 MHz	3,500 cells
Altera Stratix-2 FPGA	125 MHz	3,500 cells
Actel ProASIC3 FPGA	40 MHz	6,500 cells
Actel AX1000 FPGA	33 MHz	4,000 cells
Actel RTAX2000S FPGA	25 MHz	4,000 cells

## Feature and performance summary

### The LEON3 processor has the following features:

- SPARC V8 integer unit with 7-stage pipeline
- Hardware multiply, divide and MAC units
- Interface to high-performance IEEE-754 FPU
- Interface to custom co-processors
- Separate instruction and data cache
- Multi-way caches: 1-4 ways, 1-256 kbytes/way, random, LRR or LRU replacement policy
- Data cache snooping
- On-chip zero-waitstate scratch pad memory
- SPARC V8 Reference Memory Management Unit with hardware table walk
- Power-down mode
- Debug support unit and instruction trace buffer
- AMBA 2.0 AHB and APB on-chip buses

### The GRLIB IP library contains the following IP cores:

- AMBA AHB arbiter/multiplexer with plug&play support
- AMBA AHB/ APB bridge
- Bi- and Uni-directional AMBA AHB/AHB bridges
- 8-, 16-, and 32-bit PROM and SRAM controller
- 32-bit PC133 SDRAM controller (mobile support)
- 16-, 32- & 64-bit DDR controller (mobile support)
- 16-, 32- & 64-bit DDR2 controller
- 32-bit SSRAM controller
- UART, timers, interrupt controller and GPIO port
- 32-bit Initiator / Target PCI interface (FIFO/DMA)
- PCI trace buffer
- 10/100/1000 Mbit Ethernet MAC
- USB 2.0 Host Controller / Device Controller
- CAN 2.0B Bus Controller
- SpaceWire with RMAP
- I2C Master / Slave controller
- SPI Master / Slave controller
- VGA controller with DMA
- PS2 controller
- IDE/ATA controller with DMA
- Mil-Std-1553B BC/RT/MT
- 128-bit and 256-bit AES Encryption/Decryption
- Elliptic Curve Cryptography (ECC)
- Combined ADC / DAC interface
- Pipelined single- and double-precision IEEE-754 FPU
- Technology-independent memory, PLL and pad wrappers

## CONTACT INFORMATION

Gaisler Research AB  
Första Långgatan 19  
413 27 Gothenburg  
Sweden

Tel: +46 31 7758650  
Fax: +46 31 421407

Sales contact:  
sales@gaisler.com  
www.gaisler.com