

32 bit RISC Processor Core IP

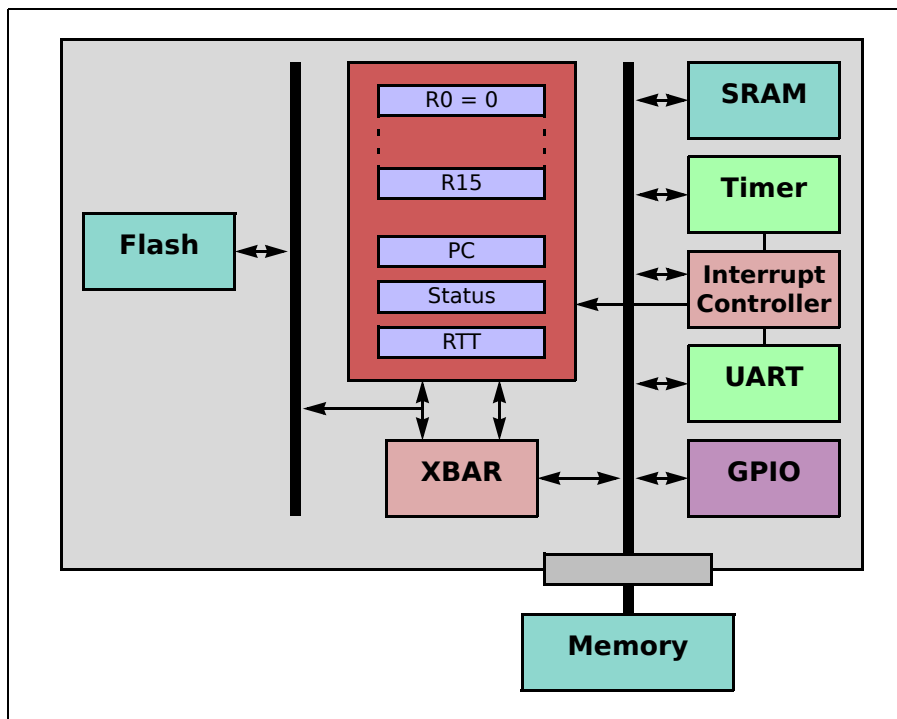
The APS2 processor is a fully 32 bit processor targeted specifically at embedded systems.

The APS2 core is a small (7k gates, including register file) high performance (250 MIPS) processor and has a very low power consumption (18µW/MHz). It is available as a fully verified soft macro for ASIC implementers and equally for FPGA users, for example the low cost Xilinx FPGAs (such as the Spartan3 XC3S200) and low cost Actel FPGAs (such as the flash APA300). Since the same IP is used in the FPGA as would be used in the ASIC flow, design migration from FPGA to ASIC is straightforward.

APPLICATIONS

The APS2 is suitable for all embedded control applications, including:

- Industrial Control Systems
- Automotive Applications
- Signal Processing
- Portable Devices



Features

ARCHITECTURE

- RISC
- Fully 32 bit
- 4 GBytes of addressable memory
- Most instructions are single cycle, including load/store
- Latency hiding through out-of-order instruction completion
- Sixteen 32 bit registers
- Single length 32 bit instructions give a very small core

PROGRAMMING

- GNU GCC C and C++ available
- GDB source code debugger available
- Debugging Stub and Monitor available
- Configurable with and without Multiply and Barrel Shifter
- Embeddable 'C' Library available
- Easy to use co-processor interface

AVAILABILITY

- Available for FPGA and ASIC
- Soft core
- Source code
- Complete GNU GCC tool-suite and compiler available for free

ARCHITECTURE

The APS2 processor is a modern RISC processors with a load/store architecture. Sixteen 32 bit general purpose registers are available alongside an optimised 32 bit ALU.

A 32 bit uniform address space permits embedded applications to be developed in high level languages such as C or C++ without any constraints. The core has a Harvard architecture, but a cross bar switch is available to provide a single bus interface. A variety of off-the-shelf memory interfaces are available.

The instruction set features 32 bit instructions, which gives a very small core with flexible instructions.

Out-of-order instruction completion hides memory latency.

Fully vectored interrupts allow the low latency required in real time applications. A configurable, programable priority interrupt controller is available to manage interrupts.

A patented co-processor interface enables the easy extension of the instruction set to further optimise the processor for the specific application domain, for example increasing the speed of signal processing algorithms.

A barrel shifter and a multiplier co-processor are available, these are supported in GNU GCC.

PROGRAMMING TOOLS

The APS2 has been designed from the ground up to be programmed in high level languages, notably C and C++.

This processor brings ease of programming combined with power to the embedded system, and can be programmed entirely in C and C++

using the industry standard GNU GCC tool chain.

The entire GCC tool chain has been adapted for the APS2 architecture, giving embedded programmers the possibility of using the industry standard GNU C/C++ Compiler suite and debugger. A stub is provided allowing the use of the GNU debugger via a serial port on any PC. These tools along with a minimal embeddable 'C' library implementation are freely available with their source code at no charge.

DELIVERABLES

The core is available for ASIC as synthesizable HDL (either VHDL or Verilog) and for FPGA implementation as a netlist.

- HDL RTL source code or post synthesis EDIF netlist
- Example mini-system design
- Synthesis scripts
- Constraints file
- Ported GNU GCC C/C++ compiler, compiled for both Windows and Linux. Plus source code.
- Complete tool chain, including GNU GDB debugger, linker, source code and library maintenance tools.
- C library, optimised for embedded applications.

APS3 — APS2 COMPARED

	APS2	APS3
Optimised for small core size	✓	
Optimised for compact code		✓
32 bit long instructions	✓	
16 & 32 bit long instructions		✓

IMPLEMENTATION DETAILS

The APS2 core is a technology independent design and can be implemented on a number of devices and processes. The table below gives typical implementation results.

Family	Device	Performance			
Actel	ProASIC060...1000	2754	Tiles	20	MHz
Xilinx	Virtex 4	778	Slices	110	MHz
ASIC 0.13µm	TSMC 0.13G — Artisan Library	0.07	mm ²	330	MHz