

CAST

PIP7-TDMI

ARM7 Native Bus Pre-Integrated IP

The PIP7-TDMI provides the essential IP cores and infrastructure software needed for systems using the native TDMI bus of the ARM7 family of microprocessors. Ready for software development out of the box but also easy to customize and extend, it serves as a basic platform for the rapid development of a variety of system-on-chip (SOC) applications.

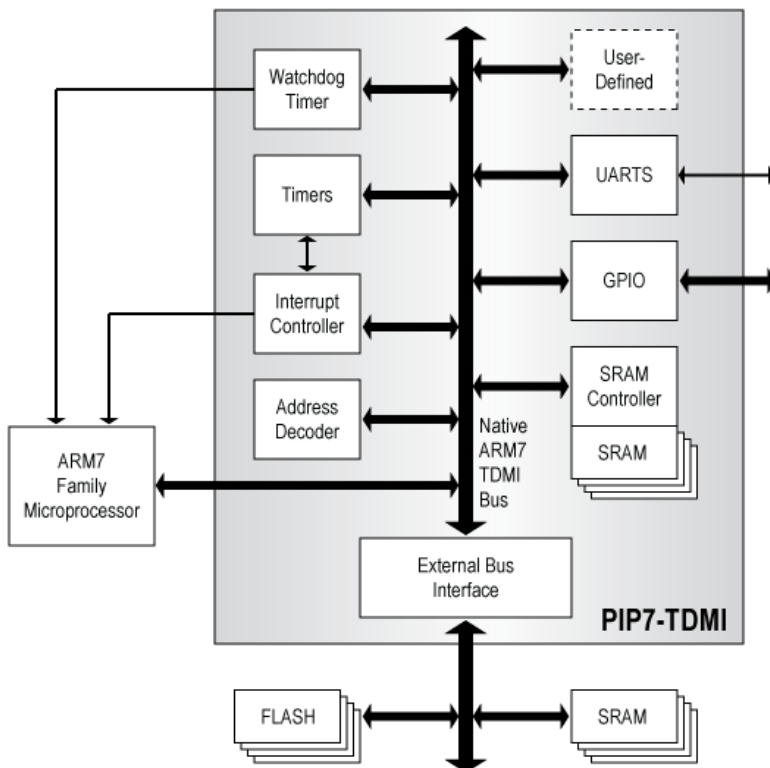
With a single-cycle internal bus, the platform is well suited to low-power, low-gate count, high-performance SOC designs. The internal memory also has single-cycle access, allowing for the fastest possible interrupt service routine handling and context switching. Designed for the ARM7TDMI bus, the platform is easily adapted to other microprocessor buses.

The platform includes synthesizable HDL cores for various timers, controllers, interface functions, communications functions, and an internal SRAM block. (FPGA netlist versions are also available.) Generous standard deliverables include device drivers, boot code, and support for an embedded real-time operating system (RTOS). The included SOC test and validation suite features a bus functional model of the ARM7.

Applications

The platform is suitable for small microcontrollers and mixed-signal controllers for a variety of applications, including factory automation, automotive systems, hand-held devices, motor controls, and intelligent toys.

Block Diagram



Features

- Integrated IP cores and software subsystem provides basic infrastructure for many SoC applications
- Platform saves significant time over acquiring and integrating separate elements
- Works with low-power, 32-bit ARM7 processor family
- Fast, single-cycle internal bus and single-cycle internal memory access
- Supports ARMTDMI™ system bus (adaptable to others)
- Easily add custom logic or additional IP cores to tailor or expand the system
- Runs RTOS out of the box; immediately begin software development and test
- Supports ARM® Developer Suite (ADS) tools, and code | lab® Debug software tool from ATI

Complete infrastructure includes essential hardware and software

- IP cores (Verilog source):
 - Two Timers
 - Extendable Address Decoder
 - Internal SRAM
 - Scalable Interrupt Controller
 - Watchdog Timer
 - Two 16550 UARTs with FIFOs
 - General Purpose I/O (GPIO)
 - External Bus Interface
- Plug-in architecture for user-defined custom IP blocks
- Software device drivers and boot code
- Support for real-time operating system (RTOS) such as ATI's Nucleus Plus
- Comprehensive SOC test and validation suite, including:
 - ARM7 bus functional model with support for interrupts and subroutines
 - Sophisticated HDL Testbench with external models and interfaces
 - Simulations scripts, vectors, expected results, and comparison utilities
- Complete user documentation

Functional Description

The PIP7-TDMI is a completely integrated and tested platform, including the bus system, memory system, and peripherals. The small, simple native bus architecture has no inherent wait states and uses single-cycle transactions. Users can readily add their own custom logic or other IP cores. The included cores are as follows.

Watchdog Timer

Issues warning alerts in the event of software failures. Each warning generates an interrupt to the Interrupt Controller and a reset to the system. The Watchdog Timer is a 16-bit down counter with a selectable prescaler, watchdog reset, warning interrupt, and reset controller. The selectable rescale values are 1, 16 and 256.

Timers

The 16-bit counter/timers are necessary for any RTOS needing a timebase and scheduling. They are fully programmable and include selectable prescale values of 1, 16, and 256. The prescaler extends the Timer's range at the expense of precision. Two modes of operation provide a free running value and also periodic interrupts.

Interrupt Controller

Manages processor attention requests for the RTOS. Fully scalable to support from one to 32 interrupt sources, and provides a programmable register used when generating an interrupt under software control.

UARTs

Two 16450/16550 compatible Universal Asynchronous Receiver/Transmitters. Each contains a baud rate generator that can be configured for a wide range of baud rates depending on the system clock frequency and the programmable divisor. Includes 16-byte internal FIFOs for both receive and transmit modes.

GPIO

Configurable, General Purpose I/O module with a scalable set of up to 32 I/O lines. Each line can be configured independently of the others, with any combination of inputs and outputs or as an interrupt source, detecting level- or edge-triggered interrupts. Useful for a wide variety of applications where simple I/O control is needed.

SRAM Controller and SRAM

The Internal SRAM Controller provides a method of communicating with an integrated Synchronous Static Random Access Memory (SSRAM). The SSRAM array comes in byte, half-word (double byte), and word (four bytes) widths and various depths. The default configuration is two kilowords, where each word is 32 bits wide (2K x 32). The memory interface allows word, half-word, or byte wide addressing.

External Bus Interface (EBI)

A configurable module interfacing the ARM7TDMI Native Bus to up to four external devices. The devices may be external SRAM, Flash, or memory-mapped peripherals. The proper number of read and write wait states and the memory size are programmable to allow proper communication. The EBI is flexible enough to work with 32-bit, 16-bit, and 8-bit external devices using word, half-word, and byte addressing.

Support

The platform as delivered is warranted against defects for ninety days from purchase. Thirty days of phone and email technical support are included, starting with the first interaction. Additional maintenance and support options are available.

Verification

The platform has been verified through extensive simulation and rigorous code coverage measurements. It is in use for several customer applications, and is a part of the SwiftTrax™ co-development systems sold by Avnet Design Services.

Deliverables

The platform includes everything required for successful implementation:

- HDL RTL source code for the included cores (post-synthesis EDIF netlists for FPGAs are also available)
- Essential software, including boot code and device drivers
- An ARM7 Bus Functional Model
- Sophisticated HDL Testbench, including external models and interfaces
- Simulation script, vectors, expected results, and comparison utility
- Synthesis scripts (place and route scripts for FPGAs)
- Comprehensive user documentation, including detailed specifications and a system integration guide