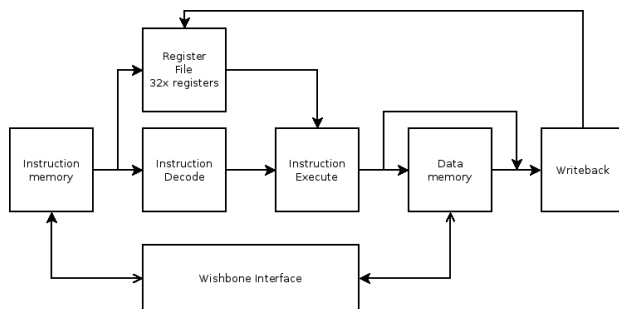


## Architecture



## Features

- Supports the complete 32-bit RISC-V base integer ISA (RV32I) version 2.0
- Supports machine mode from the RISC-V Privileged Architecture version 1.10
- Includes a hardware timer with microsecond resolution and compare interrupt
- Up to 8 IRQ inputs that can be individually enabled
- Classic 5-stage RISC pipeline
- Wishbone interface
- Optional instruction cache
- Automatic test suite

## Interface

The processor includes a wishbone interface conforming to the B4 revision of the wishbone specification.

Interface type	Master
Address port width	32 bits
Data port width	32 bits
Data port granularity	8 bits
Maximum operand size	32 bits
Endianness	Little
Sequence of data transfer	In-order

## Specifications

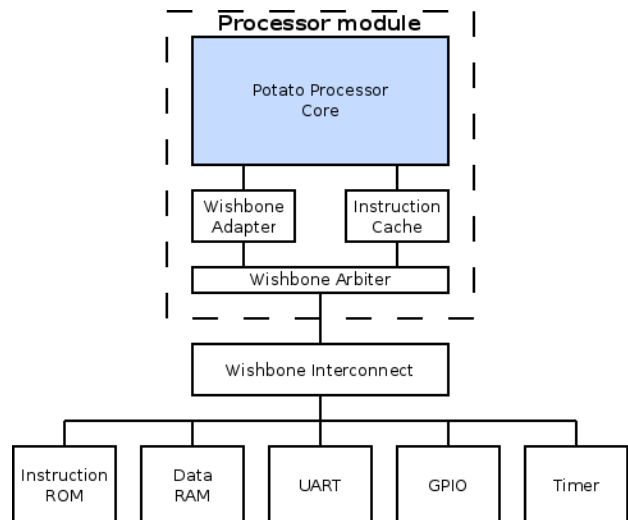


<http://riscv.org/specifications/>



<http://opencores.org/opencores,wishbone>

## Example Application



## Signals

The processor is provided by a VHDL module named `pp_potato`. All signals are active high and the following signals are provided:

Name	Width	Description
<code>clk</code>	1	Processor clock
<code>timer_clk</code>	1	10 MHz timer clock
<code>reset</code>	1	Reset signal
<code>irq</code>	8	IRQ inputs
<code>wb_adr_out</code>	32	Wishbone address
<code>wb_sel_out</code>	4	Wishbone byte select
<code>wb_cyc_out</code>	1	Wishbone cycle
<code>wb_stb_out</code>	1	Wishbone strobe
<code>wb_we_out</code>	1	Wishbone write enable
<code>wb_dat_out</code>	32	Wishbone data output
<code>wb_dat_in</code>	32	Wishbone data input
<code>wb_ack_in</code>	1	Wishbone acknowledge

An additional output, `test_context_out` is used to provide feedback to testbenches when running automated tests. This port should be left unconnected.

## Tools and Utilities

Tools for writing applications for the RISC-V architecture are available from the RISC-V project, at:

<https://github.com/riscv/riscv-gnu-toolchain>