

A Practical Platform-Level Interrupt Controller Implementation

Dr. Richard Herveille
7th RISC-V Workshop



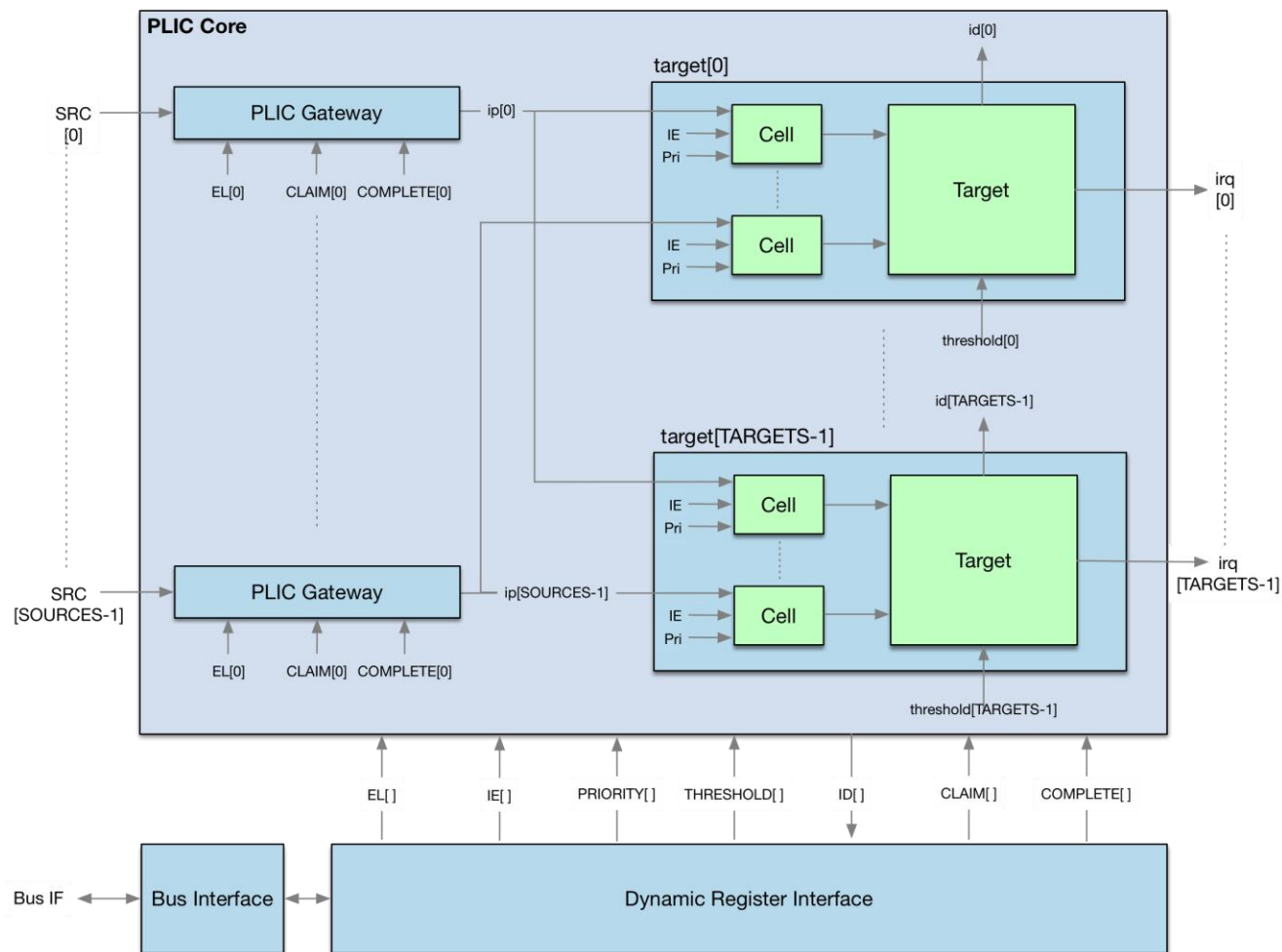
About Roa Logic

- Privately held and financed consultancy firm
- Specialized in custom IP and FPGA migrations
- Incorporated in 2014
- Strong industry basis
 - Founded by one of the original OpenCores members
 - Engaged in FPGA migrations since 2003.
 - Largely academic team

PLIC Design Goals

- Easy integration with external bus interfaces
- Support user defined number of Interrupt Sources and Targets
- Enabling and disabling of individual interrupt sources per target
- Full Priority Level and Priority Threshold support
- Low latency handling of queued interrupt requests
- Programmable depth queue of pending interrupts
- Ease of Management

Design Hierarchy



Parameters

SOURCES:

Number of Sources

TARGETS:

Number of Targets

PRIORITIES:

Number of Priority Levels

MAX_PENDING_COUNT:

Max number of Pending Interrupts (per source)

HAS_THRESHOLD:

Priority Threshold Enable

HAS_CONFIG_REG:

Config Register Enable

Design Management Challenge

- Design Flexibility → Management Complexity
 - Potentially hundreds and even thousands of registers in Memory Mapped Management Interface
 - Interface Complexity discourages design changes
 - Documentation is time-consuming and error-prone
- Management Interface Design Goals
 - Keep memory map as small as possible
 - Maintain an intuitive, logical arrangement of registers
 - Avoid need for manual maintenance of interface code

Design Management Solution

- Create Memory Map Dynamically
 - Easily adapt to wide range of parameters
 - Simplify potential future Memory Map updates
 - Automate practical register arrangement
 - Automate documentation of memory map
- Reduce Total Number of Registers
 - Pack PRIORITY, on nibble boundaries
 - Leverage ‘non-idempotent’ nature of Claim / Complete
 - Re-Use Read ID register as “Interrupt Claim”
 - Re-Use Write ID register as “Interrupt Complete”

Solution Examples

- Management Interface
 - Auto-Generated
 - Optimised
 - Documented
- Small Example
 - Simple Embedded System
 - 7 sources, 2 Targets
 - 16 Priority Levels

RISC-V Platform Level Interrupt Controller

```
- Configuration Report -----
Sources | Targets | Priority-lvl | Threshold? | Event-Cnt
  7     |  2     |    16      |     NO     |    8
- Register Map -----

Address  Function                Mapping
0x0000  Configuration        TARGETS,SOURCES
0x0004  Configuration        15'h0,TH,PRIORITIES
0x0008  Edge/Level         25'h0,EL[6:0]
0x000c  Interrupt Priority  4'h0,P[6][3:0],
                                P[5][3:0],P[4][3:0],
                                P[3][3:0],P[2][3:0],
                                P[1][3:0],P[0][3:0]

0x0010  Interrupt Enable   25'h0, IE[0][6:0]
0x0014  Interrupt Enable   25'h0, IE[1][6:0]
0x0018  ID                 29'h0, ID[0][2:0]
0x001c  ID                 29'h0, ID[1][2:0]

- End Configuration Report -----
```

Solution Examples

- Management Interface

- Auto-Generated
- Optimised
- Documented

- Larger Example

- Server/Control System
- 64 sources, 4 Targets
- 15 Priority Levels

RISC-V Platform Level Interrupt Controller

- Configuration Report -----

Sources	Targets	Priority-lvl	Threshold?	Event-Cnt
64	4	15	YES	8

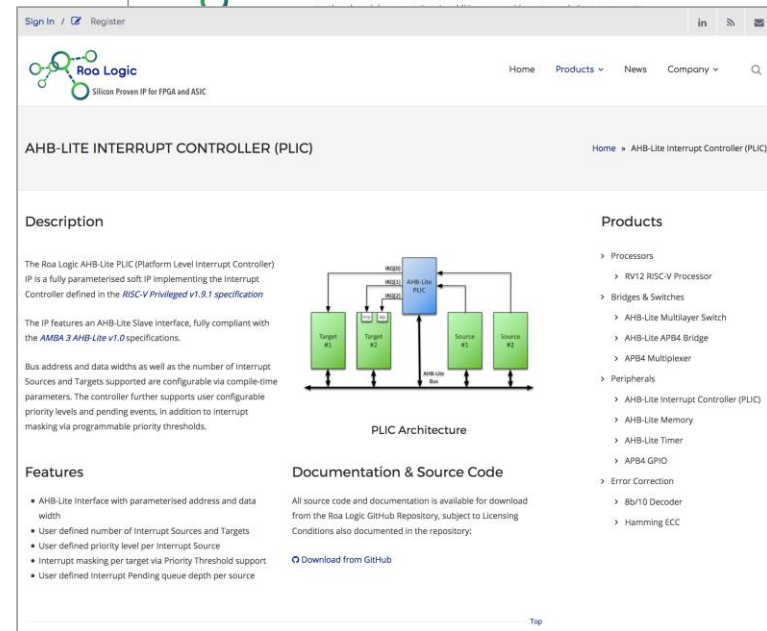
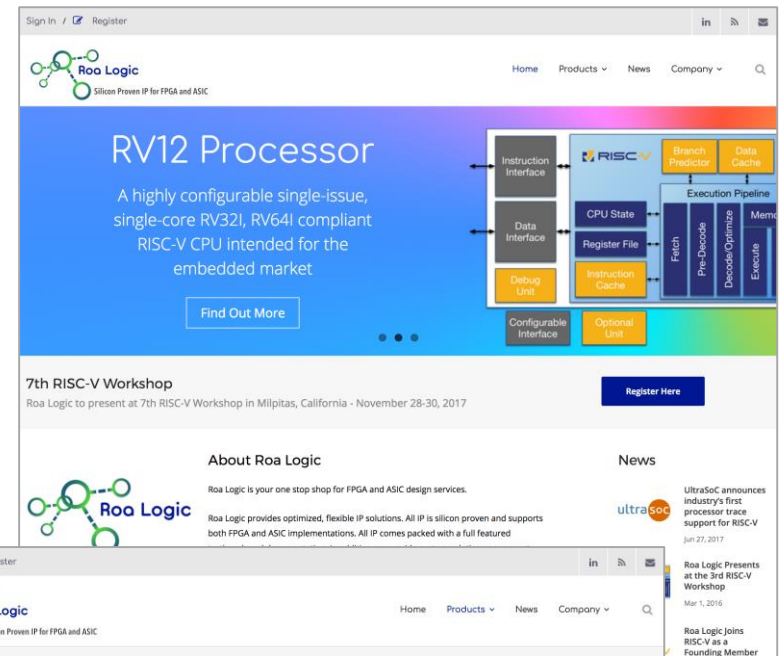
- Register Map -----

Address	Function	Mapping
0x0000	Configuration	15'h0, TH, PRIORITES, TARGETS, SOURCES
0x0008	Edge/Level	EL[63:0]
0x0010	Interrupt Priority	P[15][3:0], P[14][3:0], P[13][3:0], P[12][3:0], P[11][3:0], P[10][3:0], P[9][3:0], P[8][3:0], P[7][3:0], P[6][3:0], P[5][3:0], P[4][3:0], P[3][3:0], P[2][3:0], P[1][3:0], P[0][3:0]
0x0018	Interrupt Priority	P[31][3:0], P[30][3:0], P[29][3:0], P[28][3:0], P[27][3:0], P[26][3:0], P[25][3:0], P[24][3:0], P[23][3:0], P[22][3:0], P[21][3:0], P[20][3:0], P[19][3:0], P[18][3:0], P[17][3:0], .
0x0028	Interrupt Priority	P[63][3:0], P[62][3:0], P[61][3:0], P[60][3:0], P[59][3:0], P[58][3:0], P[57][3:0], P[56][3:0], P[55][3:0], P[54][3:0], P[53][3:0], P[52][3:0], P[51][3:0], P[50][3:0], P[49][3:0], P[48][3:0]
0x0030	Interrupt Enable	IE[0][63:0]
0x0038	Interrupt Enable	IE[1][63:0]
0x0040	Interrupt Enable	IE[2][63:0]
0x0048	Interrupt Enable	IE[3][63:0]
0x0050	Priority Threshold	60'h0, Th[0][3:0]
0x0058	Priority Threshold	60'h0, Th[1][3:0]
0x0060	Priority Threshold	60'h0, Th[2][3:0]
0x0068	Priority Threshold	60'h0, Th[3][3:0]
0x0070	ID	57'h0, ID[0][6:0]
0x0078	ID	57'h0, ID[1][6:0]
0x0080	ID	57'h0, ID[2][6:0]
0x0088	ID	57'h0, ID[3][6:0]

- End Configuration Report -----

More Information

- Web Site:
 - <https://roalogic.com>



More Information

- Web Site:
 - <https://roalogic.com>
- Source code and Full Documentation on GitHub
 - Released under BSD-Style License
 - <https://roalogic.github.io>

The screenshot displays the GitHub repository page for the 'AHB-Lite PLIC' project by Roa Logic. The page is divided into several sections:

- Overview:** Describes the IP as a Fully Parameterized & Programmable Platform Level Interrupt Controller (PLIC) for RISC-V based Processor Systems. It mentions that the IP is silicon proven and supports both FPGA and ASIC implementations. It also notes that most IPs are available for direct download for free, with commercial applications requiring contact.
- Projects:** Lists the current available GitHub repositories.
- Product Brief:** Provides a detailed description of the IP, including its compliance with the RISC-V Privileged v1.9.1 specification and its adherence to the AMBA 3 AHB-Lite v1.0 specifications. It also mentions configurable bus address and data widths, as well as interrupt sources and targets.
- Diagram:** A block diagram of the AHB-Lite PLIC IP. It shows a central blue box labeled 'AHB-Lite PLIC' with multiple input and output ports. Inputs include HCLK, HRESETn, HSEL, HTRANS, HADDR, HWDATA, HRDATA, HWRITE, HSIZE, HBURST, HPROT, HREADYOUT, HREADY, and HRESP. Outputs include SRC and IRQ.
- Download Links:** Buttons for 'Download ZIP', 'Download TAR', and 'View On GitHub' are provided.
- License Terms:** The license is listed as 'BSD License'.
- Hosted on GitHub Pages:** A note indicating that the project is hosted on GitHub Pages.

The End

