Soft IP-Cores: ARM/RISC-V Processors

OBI NNAMDI ELIJAH

***Abstract*—A soft-core processor is a type of processor that can be customized for a specific application and synthesized for use in an ASIC or FPGA. It is usually represented as a high-level language model or a netlist format model and is generally portable, but not optimized for a specific process technology. The main advantages of a soft-core processor are its functionality, reusability, and conformance to industry standards. This paper surveys soft-core processors that are commonly used in embedded systems, with a focus on ARM and RISC-V processors, and includes brief discussions of implementation examples for these two types of processors.**

***Index Terms*—VHDL - Very High Speed Integrated Circuit Hardware Descriptive Language, IP - Intellectual Property, SIP - Semiconductor Intellectual Property, ISA - Instruction Set Architecture**

I. INTRODUCTION

In the early years of the semiconductor industry, IC (integrated circuit) suppliers like Fairchild, Intel, TI, and Motorola developed proprietary SIP (intellectual property cores) for their internal use, including data and circuit design expertise, process knowledge, packaging test equipment, and other items. These companies protected their SIP through patents, trade secrets, and other legal means.

Over the last decade, the productivity of IC (integrated circuit) design has not kept up with Moore's Law, leading to a "design gap" despite the growth of the electronic design automation (EDA) industry. In response to this gap, IC suppliers have developed reusable SIP (intellectual property cores) with more complex functionality to increase design productivity. The foundry model, in which fabless IC companies use customer-owned tooling (COT) and block-based design methodologies to leverage cost advantages and allow for design mobility, has also contributed to the development of SIP as a standalone entity that can be licensed to third parties. ARM Holdings is a notable example of this trend, offering a RISC processor architecture in both soft (synthesizable) and hard versions for use in ASIC design.

Embedded systems designs have become increasingly complex, making it impractical and expensive for designers to create every hardware component from scratch. To address this issue, the use of pre-designed and pre-tested intellectual property (IP) cores has become a popular alternative. Soft-core processors are microprocessors that can be customized for specific applications and synthesized for use in ASICs or FPGAs, giving designers a high level of flexibility. They are described using an HDL and can be synthesized for various technologies.

Soft-core processors offer several advantages for designers of embedded systems. They are flexible and can be customized for specific applications, are technology independent and can be synthesized for different technologies, and have a higher level of abstraction that makes them easier to understand. These features make them more durable and useful compared to circuit or logic level descriptions of processors.

In using Soft cores we have to interact with the specified Instruction set architectures (ISAs), and these are standards that specify how a processor should function and interact with its own assembly language. They include information about instructions, registers, memory access, arithmetic, and data buses, among other things, and allow for the creation of processors that will run machine code consistently according to the standard. While the functions of a processor are defined in the ISA, the hardware and circuit implementation is left to the designer. ISAs often have many extensions and variations to suit different design requirements, such as support for multiplication, floating point numbers, or different data and instruction widths, or for different types of systems such as embedded systems, personal computers, or supercomputers.

A. PAPER ORGANISATION

This paper is organised as follows: In Section 2, after a review of what is an IP Core, I endeavoured to provide additional comparison as regards to the types of IP Cores we have and the differences between them. In Section 3, I discussed about the ARM Soft IP-Core with several examples currently in use in the industries. In Section 4, I looked at what RISC-V soft IP-cores are all about, citing different industry use-cases as well as implementation and board examples and lastly, in Section 5 conclusions were drawn.

II. IP CORES

IP cores, or intellectual property cores, are a longstanding aspect of the semiconductor industry. There are two main types of IP cores: soft and hard. The form and type of an SIP product can affect its price, availability, and support needs. The value of an SIP product can also vary depending on how it is used by the customer.

A. HARD IP CORES

Hard IP cores are digital logic designs that are implemented in a specific integrated circuit (IC) technology and cannot be easily reprogrammed. They are often used to implement complex functions or features that require high performance or are critical to the operation of a system. Hard IP cores are typically developed by specialized IC design companies or in-house design teams at large semiconductor companies. They are usually optimized for a specific IC technology and manufacturing process, which allows them to achieve higher

performance and lower power consumption than soft IP cores. One key advantage of hard IP cores is that they are optimized for a specific IC technology, which allows them to achieve higher performance and lower power consumption than soft IP cores. This makes them a good choice for applications that require high performance or low power consumption. Another advantage of hard IP cores is that they are fixed in a specific IC technology, which can make them more reliable and easier to manufacture than soft IP cores. This can be particularly important for applications that require high levels of reliability or that need to be manufactured in large quantities. Overall, hard IP cores offer a high-performance and reliable way to implement complex functions or features in a digital system, but they are generally more expensive to develop and license than soft IP cores, and they are not as flexible or reusable.

Hard SIP, or hard intellectual property cores, are optimized for a specific foundry process and are usually offered in GDSII format with accompanying EDA views or models. They may also be offered in ”bit-stream” format if they were derived from a ”hardened” soft SIP for a particular FPGA device. Hard SIP often has a data sheet with information on its power, speed, and area, similar to a finished discrete IC. Examples of hard SIP include processors, standard cells, memories, phase locked loops (PLLs), I/Os, and analog blocks. Hard SIP is generally not portable to other foundry processes, although there are libraries of standard cells, I/Os, and memory that are generally not optimized for any specific foundry. Some hard SIP suppliers conduct extensive silicon validation on test chips to ensure high quality and yield.

B. SOFT IP CORES

Soft IP cores are digital logic designs that can be implemented in programmable logic devices such as field-programmable gate arrays (FPGAs) or used as a component in the design of a system on a chip (SoC). They are usually offered in a high-level hardware description language (HDL) such as Verilog or VHDL, or sometimes in netlist format. Soft IP cores are typically used to implement a specific function or feature in a digital system, such as a processor, memory controller, or peripheral interface. They are often designed to be flexible and customizable, allowing them to be easily integrated into a variety of different systems and technologies.

One key advantage of soft IP cores is that they are portable and can be easily moved from one manufacturing process to another. This makes them a good choice for designs that need to be implemented in different technologies or that may need to be adapted to different process technologies over time.

Another advantage of soft IP cores is that they can be easily modified or customized to meet the specific requirements of a system. This can be done by modifying the high-level hardware description language (HDL) code that defines the core, or by using customization options provided by the supplier of the core.

Overall, soft IP cores offer a flexible and reusable way to implement specific functions or features in a digital system, allowing designers to focus on the overall system architecture and functionality rather than on the implementation of individual components.

In summary, Soft IP cores are generally portable and can be easily moved from one manufacturing process to another, but they are not optimized for a specific process technology, so their power, performance, and area are not known until they are implemented in a specific process. The value of a soft IP core lies in its functionality, re-usability, and potential conformance to industry standards. Some suppliers of soft IP cores may also provide data for implementing the core in an FPGA for validation or in a finished product.

C. DIFFERENCES BETWEEN HARD IP-CORES AND SOFT IP-CORES

A soft IP core is a digital logic design that can be implemented in a programmable logic device, such as a field-programmable gate array (FPGA). A hard IP core, on the other hand, is a digital logic design that is implemented in a specific integrated circuit (IC) technology and cannot be easily reprogrammed.

There are several key differences between soft and hard IP cores:

1) Flexibility: : Soft IP cores are more flexible than hard IP cores because they can be implemented in a variety of programmable logic devices and can be easily modified or customized. Hard IP cores, on the other hand, are fixed in a specific IC technology and cannot be easily modified.

2) Re-usability: : Soft IP cores can be reused in multiple designs and across different technologies, whereas hard IP cores are tied to a specific IC technology and cannot be easily reused. Performance: Hard IP cores can typically achieve higher performance than soft IP cores because they are optimized for a specific IC technology and are not limited by the performance of the programmable logic device.

3) Cost: : Hard IP cores can be more expensive to develop and license than soft IP cores because they require more specialized design and manufacturing processes. However, the cost of using a hard IP core may be lower in the long run because it can be used in multiple designs without incurring additional licensing costs.

In general, the choice between a soft IP core and a hard IP core depends on the specific requirements of the system and the trade-offs between flexibility, performance, and cost.

III. ARM SOFT IP-CORES

ARM is a leading provider of microprocessor technology and intellectual property (IP). ARM offers a range of soft IP cores that can be used as components in the design of a system on a chip (SoC) or as processors in larger systems. ARM soft IP cores are based on the ARM instruction set architecture (ISA), which is widely used in a variety of applications, including mobile phones, tablets, and other embedded systems. ARM offers a range of processors that are optimized for different applications, including high-performance processors for applications that require high performance and low-power processors for applications that need to operate for long periods of time on a single battery charge.

ARM soft IP cores are usually offered in a high-level hardware description language (HDL) such as Verilog or VHDL, or

sometimes in netlist format. They are designed to be flexible and customizable, allowing them to be easily integrated into a variety of different systems and technologies. ARM also offers a range of tools and support services to help designers integrate ARM soft IP cores into their systems, including software development kits (SDKs), hardware development kits (HDKs), and technical support. ARM processors are based on RISC architecture, with some improvements made based on current embedded systems requirements. They have two instruction sets, the general 32-bit ARM instruction set and the 16-bit Thumb instruction set, and feature 3-address data processing instructions, conditional execution, load and store multiple register instructions, a general shift operation and ALU operation in a single instruction, a 16-bit compressed representation of the instruction set in the Thumb architecture, instruction pipelining, and a high clock rate with single-cycle execution.

A. ARM SOFT IP-CORES USE CASE AND APPLICATION

ARM offers a wide range of soft IP cores that are optimized for different applications and target markets. Some examples of ARM soft IP cores include:

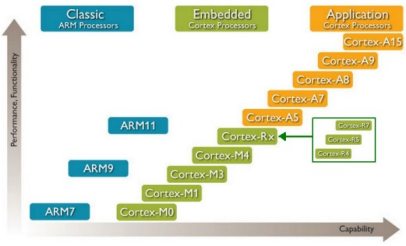

Fig. 1. ARM Classification

1) Cortex-A processors: These are high-performance processors that are optimized for applications that require high performance, such as smartphones, tablets, and other portable devices. Cortex-A processors are available in a range of configurations, including single-core and multi-core designs.

2) Cortex-M processors: These are low-power processors that are optimized for applications that need to operate for long periods of time on a single battery charge, such as wearable devices and Internet of Things (IoT) devices. Cortex-M processors are available in a range of configurations, including single-core and multi-core designs.

3) Cortex-R processors: These are real-time processors that are optimized for applications that require fast interrupt handling and deterministic performance, such as automotive and industrial systems. Cortex-R processors are available in a range of configurations, including single-core and multi-core designs.

4) Cortex-D processors: These are digital signal processor (DSP) cores that are optimized for applications that require high-performance signal processing, such as audio and video processing. Cortex-D processors are available in a range of configurations, including single-core and multi-core designs.

5) Mali Graphics Processors: These are graphics processing units (GPUs) that are optimized for applications that require high-performance graphics rendering, such as gaming and multimedia applications. Mali graphics processors are available in a range of configurations, including single-core and multi-core designs. These are just a few examples of the many ARM soft IP cores that are available. ARM also offers a range of other IP products, including memory controllers, peripheral interfaces, and other subsystems.

The cost of an ARM soft IP license will depend on the specific core you are using, as well as the intended use and scale of your project. ARM typically offers a range of licensing options for its soft IP cores, including per-core, per-project, and perpetual licenses, and the cost will vary based on the specific terms of the license you choose. The cost of an ARM soft IP license will include the licensing fee for the core, as well as any additional tools or software required for integration. You may also need to pay for engineering services or additional hardware components to help with the integration process.

In general, the cost of an ARM soft IP license can range from a few hundred dollars to several thousand dollars, depending on the complexity and scale of the core and the specific terms of the license. It is generally advisable to carefully evaluate the costs and benefits of using an ARM soft IP core before making a decision, and to consider any potential long-term costs such as maintenance and support fees. It may also be helpful to speak with ARM or a qualified technical expert to get a better understanding of the costs and requirements involved.

Based on the cost implication we will be looking at a much more viable open source alternative that is currently being backed by major industrial players in different fields. In the next section we will look more into

IV. RISC-V SOFT IP-CORES

RISC-V is an architecture for processors that was developed at the University of California, Berkeley, by Yunsup Lee, Krste Asanović, David A. Patterson, and Andrew Waterman. It was publicly announced in 2014 and is the fifth major RISC architectural design project at UC Berkeley after four earlier failed attempt, hence the name RISC-V (V represents the Roman numeral five). RISC-V is an open-source instruction set architecture (ISA) for processors, which means that it defines a set of instructions that a processor can execute. RISC-V processors can be implemented in hardware or as software, known as a soft IP core. A RISC-V soft IP core is a digital logic design that can be used to create a processor that is compatible with the RISC-V ISA. It can be implemented in a programmable logic device, such as an FPGA, or in an ASIC (application-specific integrated circuit). RISC-V soft IP cores are often used in embedded systems and other applications where a processor with a specific set of features is required. According to the book, “Modern Computer Architecture and Organization” by Jim Ledin, *The RISC-V project began as a clean sheet with these major goals:*

- *Design a RISC instruction set architecture (ISA) suitable for use in a wide variety of applications, spanning the*

spectrum from micro-power embedded devices to high-performance cloud server multiprocessors.
- *Provide an ISA that is free to use by anyone, for any application. This contrasts with the ISAs of almost all other commercially available processors, which are the carefully guarded intellectual property of the company that designed them.*
- *Incorporate lessons learned from previous decades of processor design, avoiding wrong turns and suboptimal features that other architectures must retain in newer generations to maintain compatibility with previous, sometimes ancient in technological terms, generations.*
- *Provide a small but complete base ISA suitable for use in embedded devices. The base ISA is the minimal set of capabilities any RISC-V processor must implement. The base RISC-V is a 32-bit processor architecture with 31 general-purpose registers. All instructions are 32 bits long. The base ISA supports integer addition and subtraction, but does not include integer multiplication and division. This is to avoid forcing minimal processor implementations to include the fairly expensive multiplication and division hardware for applications that do not require those operations.*
- *Provide optional ISA extensions to support floating-point mathematics, atomic memory operations, and multiplication and division.*
- *Provide additional ISA extensions to support privileged execution modes, similar to the x86, x64, and ARM privileged implementations.*
- *Support a compressed instruction set, implementing 16-bit versions of many 32-bit instructions. In processors implementing this extension, 16-bit instructions may be freely interspersed with 32-bit instructions.*
- *Provide optional ISA extensions to support 64-bit, and even 128-bit, processor word sizes using paged virtual memory on single- and multi-core processors, and in multiprocessing configurations.*

[1] RISC-V processors are already available at competitive prices and are expected to increase in market share due to the sophistication of their design and the fact that their instruction set architecture is free to use. RISC-V Linux distributions, which include software development tools, are also available for building and running applications on RISC-V-based devices.

A. IMPLEMENTATION AND DIFFERENT EXAMPLES THAT EXIST

B. RISC-V SOFT IP-CORES USE-CASE AND APPLICATION

A RISC-V soft IP core is a digital logic design that implements the RISC-V instruction set architecture (ISA) in hardware. It can be used as a component in the design of a system on a chip (SoC) or as a processor in a larger system. There are several ways to implement a RISC-V soft IP core, depending on the specific requirements of the system and the resources available. Some common approaches include:

- Using a high-level hardware description language (HDL) like Verilog or VHDL to describe the behavior of the processor at a high level, and then synthesizing the design into a gate-level implementation using a synthesis tool.
- Using a pre-designed RISC-V processor core that is available as a soft IP core from a third-party vendor. These cores may come with a range of customization options, such as the number of instructions per clock cycle and the size of the register file.
- Using a pre-designed RISC-V processor core as a starting point, and then modifying it to meet the specific requirements of the system. This may involve adding or removing instructions or changing the micro-architecture of the processor.

Regardless of the approach chosen, implementing a RISC-V soft IP core typically involves the following steps:

- Defining the instruction set and micro-architecture of the processor: This includes specifying the registers, memory hierarchy, and pipeline stages.
- Designing the data-path and control logic for the processor: This involves designing the circuits that execute the instructions and control the flow of data through the processor.
- Integrating the processor into the overall system design: This may involve interfacing the processor with other components in the system, such as memory and peripherals.
- Verifying the correct operation of the processor using simulation or other techniques.
- Synthesizing the design into a gate-level implementation and integrating it into the final chip layout.

There are several RISC-V soft IP cores available from various vendors and open-source projects. Some examples include:

1) Rocket: An open-source RISC-V processor developed by UC Berkeley that is widely used for research and education. Rocket is available in various configurations, including a single-core design and a multi-core design.

2) SiFive Core IP: A commercial RISC-V processor developed by SiFive, a leading provider of RISC-V IP. SiFive offers a range of Core IP products, including high-performance processors and microcontrollers.

3) PULPino: An open-source RISC-V processor developed by the PULP (Parallel Ultra-Low Power) project at ETH Zurich. PULPino is optimized for low-power and energy-efficient applications.

4) OpenRISC: An open-source RISC processor developed by the OpenCores community. OpenRISC is designed to be flexible and customizable, and it is available in various configurations.

5) AndesCore: A commercial RISC-V processor developed by Andes Technology, a leading provider of embedded processor IP. AndesCore is available in various configurations and is optimized for a range of applications.

These are just a few examples of the many RISC-V soft IP cores that are available. There are many other options available, depending on the specific requirements of your system.

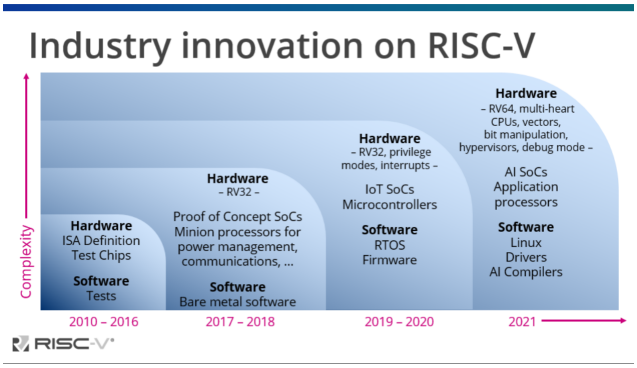

Fig. 2. Industry Innovation on RISC-V

C. RISC-V SOFT IP-CORES USE CASES AND APPLICATION

RISC-V soft IP cores are digital logic designs that can be used to create processors that are compatible with the RISC-V instruction set architecture (ISA). They can be implemented in programmable logic devices, such as FPGAs, or in ASICs (application-specific integrated circuits) which are often used in embedded systems and other applications where a processor with a specific set of features is required. Some common use cases for RISC-V soft IP cores include:

1) Internet of Things (IoT) devices: RISC-V soft IP cores can be used to create processors for IoT devices, which often have low power requirements and need to be able to run on small batteries.

2) Industrial control systems: RISC-V soft IP cores can be used to create processors for industrial control systems, which often require real-time processing and low power consumption.

3) Consumer electronics: RISC-V soft IP cores can be used to create processors for a wide range of consumer electronics, including smartphones, tablets, and smart TVs.

4) Networking equipment: RISC-V soft IP cores can be used to create processors for networking equipment, such as routers and switches.

5) Automotive systems: RISC-V soft IP cores can be used to create processors for automotive systems, including infotainment systems, advanced driver assistance systems (ADAS), and autonomous vehicles.

RISC-V soft IP cores are also used in a variety of other applications, including data centers, storage systems, and scientific computing.

V. CONCLUSION

In conclusion, soft IP cores offer a number of benefits and limitations that should be carefully considered when deciding whether to use them in a project. Soft IP cores can be a cost-effective and flexible way to implement a processor that is compatible with a particular instruction set architecture (ISA), and they are commonly used in embedded systems and other applications where a specific set of features is required. However, they may not offer the same level of performance as a hardware processor. There are various types of soft IP cores available, including processor cores, memory cores, and connectivity cores, and the form and type of the core can impact its price, availability, and ease of integration. As the use of soft IP cores continues to grow and evolve, it is likely that they will have a significant impact on the industry and the way that processors are designed and implemented.

REFERENCES

[1] Modern Computer Architecture and Organization by Jim Ledin
[2] A. Oleksiak, S. Cieślak, K. Marcinek and W. A. Pleskacz, "Design and Verification Environment for RISC-V Processor Cores," 2019 MIXDES - 26th International Conference "Mixed Design of Integrated Circuits and Systems", 2019, pp. 206-209, doi: 10.23919/MIXDES.2019.8787108.
[3] M. Eckert, J. Haase and B. Klauer, "Unifying Timer and Interrupt Management for an ARM-RISC-V-Heterogeneous Multi-Core," IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society, 2020, pp. 2302-2309, doi: 10.1109/IECON43393.2020.9255193.
[4] D.K. Dennis et al., "Single cycle RISC-V micro architecture processor and its FPGA prototype", 2017 7th International Symposium on Embedded Computing and System Design (ISED), pp. 1-5, 2017.
[5] D. K. Dennis et al., "Single cycle RISC-V micro architecture processor and its FPGA prototype," 2017 7th International Symposium on Embedded Computing and System Design (ISED), 2017, pp. 1-5, doi: 10.1109/ISED.2017.8303926.
[6] https://riscv.org/
[7] A. Singh, N. Franklin, N. Gaur and P. Bhulania, "Design and Implementation of a 32-bit ISA RISC-V Processor Core using Virtex-7 and Virtex- UltraScale," 2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA), 2020, pp. 126-130, doi: 10.1109/ICCCA49541.2020.9250850.
[8] J. -Y. Lai, C. -A. Chen, S. -L. Chen and C. -Y. Su, "Implement 32-bit RISC-V Architecture Processor using Verilog HDL," 2021 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), 2021, pp. 1-2, doi: 10.1109/ISPACS51563.2021.9651130.
[9] L. Poli, S. Saha, X. Zhai and K. D. Mcdonald-Maier, "Design and Implementation of a RISC V Processor on FPGA," 2021 17th International Conference on Mobility, Sensing and Networking (MSN), 2021, pp. 161-166, doi: 10.1109/MSN53354.2021.00037.