Watermarking Strategies for RNS-based system Intellectual Property Protection

Encarnación Castillo, Luis Parrilla, Antonio García, Antonio Lloris
Department of Electronics and Computer Technology
University of Granada, Spain
{ecastillo, lparrilla, agarcia, lloris}@ditec.ugr.es

Uwe Meyer-Baese
Department of Electrical and Computer Engineering
FAMU-FSU College of Engineering
Tallahassee, Florida, USA
umb@eng.fsu.edu

Abstract—This paper present a new procedure for Intellectual Property Protection (IPP) of circuits based on the Residue Number System (RNS). The aim is to protect the author rights of reusable IP cores by means of an electronic digital signature that uniquely identifies both the original design and the design recipient. The signature embedding stores the digital signature in non-used positions of look-up tables of RNS-based designs. This embedding does not increase the area of the system. A procedure for signature extraction is also included, so it is possible to detect the ownership right without interfering the normal operation of the system. This signature extraction requires some extra hardware, basically additional logic and some multiplexers. As an example, a 160-bit signature is introduced into a FIR filter. The presented IPP design examples are implemented over FPL devices and cell-based ASICs with negligible performance penalties.

I. INTRODUCTION

Silicon Technology has progressed to the point of making chips with millions of transistors commercially available. This allows more features and capabilities in reprogrammable technology, as well as promises new levels of integration into a single chip. Design methodologies and CAD tools struggle to keep pace with the advances of integrated circuit technology that are leading toward higher performance, greater densities, and increasing system complexity. Thus, reuse-based design methodologies allow to assemble complex system using smaller components, reducing system design complexity. These strategies based in the use of reusable modules, commonly referred to as Intellectual Property (IP) cores [1-2], are enabling the optimization of company resources due to the reduced development time and costs. With the availability of new FPGA architectures designed for system level integration and FPGA design tools that are compatible with ASIC design methodologies, it is now possible to employ design reuse methodology for ASICs and FPGAs. For design teams, this design reuse methodology adds a new level of freedom: it is possible to eliminate costs and improve time-to-market without becoming FPGA experts. Reusable IP is essential for constructing bug-free multimillion-gate designs in a reasonable amount of time. Without reuse, the electronics industry would struggle to keep pace with the challenge of delivering better, faster and cheaper devices that consumers expect. However this design methodology has some risks, one of them being the violation of Intellectual Property rights of those reusable modules. Because IP cores are designed to be modular and integrated with other components it is possible for a third party to sell an IP block as their own, only requiring to understand the interface and function and without even knowing the internal architecture or implementation. This causes the need for Intellectual Property Protection (IPP) techniques that allow the safe exchange of designs and the protection of ownership rights. It is in this context where watermarking techniques come to get these goals. The usual watermarking-based procedures for IPP in media and hardware support [3] consist of embedding a watermark into an IP block, in such way that it is difficult to change or to remove this watermark. The watermark is an invisible identification code that is permanently embedded as an integral part within a design. There should be no perceptible difference between the watermarked and original IP core, and the watermark should be difficult to remove or alter without damaging the host IP core.

This paper presents a new watermarking solution to protect RNS-based IP cores. We propose the introduction of a digital signature that allows identifying the author and the yielded recipient rights for RNS-based designs. This protection scheme embeds a digital signature at HDL design level that is preserved through synthesis, placement and routing processes. The scheme enables also simple signature extraction at the physical layout level. The new technique provides high invulnerability while requiring low area and negligible timing overhead. The IPP design examples are based on a complex but general DSP IP structure, a reconfigurable FIR filter [4].

II. OVERVIEW OF WATERMARKING

Throughout history, watermarking and data hiding have been widely used for copyright protection. Watermarking is the direct embedding of additional information into the original content. Perceptible marks of ownership or authenticity have been around for centuries in the form of...
stamps, seals, signatures or classical watermarks. Recently, a candidate solution for the copyright protection problem of digital media has emerged, this is digital watermarking.

A. Digital watermarking

Digital watermarking has focused on multimedia content (image, audio, video, text data), becoming a very active research area over the last several years. Digital watermarking includes many different techniques which are usually designed for specific applications and may not be applicable in other contexts. One of the basic requirements for digital watermarking is to maintain the quality of the original data, this not being distorted when a watermark is embedded into it. However, there are more requirements that may be required for specific applications to achieve maximum protection of intellectual property with watermarked media. Some of the fundamental requirements for a watermarking-based IP core protection technique are maintenance of functional correctness, transparency to existing design flows, minimal overhead cost, invisibility, proportional component protection, enforceability and persistence.

In digital watermarking there are two basic processes, watermark embedding and watermark extraction. The watermark can be information about ownership, user identity or description of the original data. The watermark can be perceptible or imperceptible in the watermarked content, depending on the applications. For IP cores protection the watermark has to be nearly invisible to human and machine inspection and not interfere with the design functionality.

B. IP Cores watermarking techniques

In the literature, several techniques have been developed for IP core watermarking. A typical FPGA design flow includes a number of design stages or levels including behavioral hardware description language (HDL), synthesis to register transfer language (RTL), technology mapping and physical layout involving place and route. At each stage, a number of choices are available to the designer: optimize design for area and/or speed, allow register duplication or not, and so on. Watermarking techniques can be applied to any of these levels and will propagate to later stages.

Some authors have proposed to use design constraints to implant watermarks [1]. In [5] the watermark is stored in some of the unused configurable control logic blocks (CLB) of the FPGA. This last watermarking technique, in combination with “tiling” algorithms, is used in [3, 6] where little extra effort is required to generate multiple physical designs. In [7] a method for embedding a watermark by modifying the number of bends used to route a design is proposed. All of these techniques, as well as the proposed in [2, 8-9], embed watermarks at the physical level. Physical level should not be an exclusive domain for IPP, since in that case only the solution to the physical design is protected.

Although some researches have investigated into the physical design level, few interests have been directed to behavioral design level [10-12]. The advantage of making use of watermarking techniques at high-level design processes resides on the difficulty to remove the mark, since the signature is embedded in preliminary stages and it is dragged through the whole design flow. In addition, the watermark could be embedded as a functional part of the design. The main drawback could be related to the constraints introduced in the design process. In the following section the new procedure to achieve IPP in reusable modules at HDL design level, oriented to RNS-based hardware to be implemented on FPL devices, is revised and fundamental improvements are introduced.

III. RESIDUE NUMBER SYSTEM

Residue Number System (RNS) [13-14] is an efficient alternative to the classical 2’s complement (2C) arithmetic system for digital signal processing (DSP) applications demanding high speed and high precision. The system is divided in a number of smaller replicas with no arithmetic dependencies between them. These subsystems, called channels, operate in parallel processing residues at high speed by means of specialized RNS arithmetic units. Computer arithmeticians have long held that the RNS offers the best speed-area advantage and it has shown important benefits for the implementation of modern DSP systems using cell-based integrated circuit (CBIC) technologies [15] and field-programmable logic (FPL) devices [16].

A. RNS arithmetic

In the RNS, numbers are represented in terms of a relatively prime basis set (moduli set) $P=\{m_1, \ldots, m_l\}$. Any number $X \in \mathbb{Z}_P = \{0, \ldots, M-1\}$, where $M = \prod_{i=1}^{l} m_i$ has a unique RNS representation $X = \{X_1, \ldots, X_l\}$, where $X_i = X \mod (m_i)$. As well as the 2C system, RNS arithmetic is exact as long as the final result is bounded within the system’s dynamic range $M$. Mapping from the RNS back to the integer domain is defined by the Chinese Remainder Theorem (CRT) [13-14]. RNS arithmetic is defined by pair-wise modular operations:

$$Z = X \pm Y \leftrightarrow \{X_1 \pm Y_1, \ldots, X_l \pm Y_l\}$$

$$Z = X \times Y \leftrightarrow \{X_1 \times Y_1, \ldots, X_l \times Y_l\}$$

(1)

where $<Q>$ denotes $Q \mod (m_i)$. The RNS differs from traditional weighted numbering systems in that the RNS arithmetic is a carry-free and can operate at a constant speed over a wide range of word widths (precision). It is the ability of the RNS to do arithmetic within independent small word length channels what has raised the interests of system
designers. This property has mainly been exploited for performance enhancement with respect to traditional arithmetic systems, while this representation system has been shown particularly well-suited for FPL implementation [17-19]. Because of the RNS particularities, RNS-based applications on FPL devices have traditionally made extensive use of the variety of look-up tables available in programmable technologies.

B. RNS-to-binary converters

In RNS systems conversion blocks are required to interconnect the circuit with standard binary systems. The most referenced RNS-to-binary conversion methods are based on the Chinese Remainder Theorem (CRT) and the mixed-radix conversion (MRC) [13-14]. An efficient algorithm scheme, in terms of area and throughput, called ε-CRT algorithm [20], can be used. An ε-CRT algorithm based converter passes a minimum latency and allows an output scaling. The ε-CRT algorithm is defined by:

\[ y(n) = \left( \sum_{i=0}^{n} \left( y \left( x_i \right) \cdot s_i \right) \bmod m_i \right) \bmod \left( M/v \right) \]  

(3)

where \( s_i = M/m_i \) and \( \left( x_i^{-1} \right) \bmod m_i = 1 \). The key point in this algorithm consists on choosing \( V \), any real number, so that \( MV \) is a power of 2. Thus, the modulo \( MV \) is implemented using conventional adders. The \( n \) terms \( y(n) \cdot x_i^{-1} \bmod m_i \) \( V \) are precomputed and store in \( n \) LUTs. So this converter only requires look-up tables and conventional binary adders.

C. Galois Field index arithmetic

Index arithmetic [14] is an efficient means for designing high performance, reduced complexity DSP systems. It is based on the mathematical properties associated with Galois fields, denoted \( \text{GF}(p) \), with \( p \) being a prime. All the non-zero elements in a Galois field can be generated by exponentiating a primitive element, denoted \( g \). This property can be exploited for multiplication in \( \text{GF}(m) \) through the use of the well-known isomorphism between the multiplicative group \( Q = \{ 1, 2, ..., m-1 \} \), with multiplication modulo \( m \), and the additive group \( I = \{ 0, 1, ..., m-2 \} \), with addition modulo \( m-1 \). The mapping is given by:

\[ q = \Phi_1(l) = g^l \bmod m, \quad l = 1, 2, ..., L \]  

(4)

where \( q \in Q, i.e. l \) and multiplication is based on:

\[ q_1q_2 \bmod m = g^{l_1+l_2} \bmod m \]  

(5)

Thus, multiplication of two operands, namely \( q_1 \) and \( q_2 \), can be performed by adding exponents in a modular sense. The exponents, or indexes \( l_1 \) and \( l_2 \), are precomputed and stored in a LUT. Index addition can be performed with a modulo \( m-1 \) adder [21], and the inverse index transformation can be performed again using a LUT.

IV. New Watermarking Technique

The proposed watermarking technique hosts the bits of a digital signature through some of the non used cells included in designs, with this signature embedding being performed at the high-level design description stage. This watermarking scheme includes a method to extract the signature of the circuit at the physical layout level. The signature extraction allows to detect the ownership rights of the IP provider and does not interfere with the correct functioning of the circuit. Thus, two objectives are achieved simultaneously: the identification of the author and the provision of a procedure for claiming author rights.

This watermarking technique can be applied to any type of system, but in this paper the methods are oriented to RNS-based systems. In these systems there are different possibilities of look-up tables with unused positions, as shown Fig. 1. We are going to distinguish between input converter, data processing block and output converter look-up tables:

- **Input converter LUTs.** Input binary-RNS conversion block is required to interconnect the RNS-based circuit with standard binary systems. When the internal modular processing engine is intended to work using index arithmetic the input converter includes binary-to-RNS and RNS-to-index or direct index conversions. The look-up tables used in direct index transformation included in the input converter are not completely full, as pointed in Fig. 1, and could be used for embedding the digital signature.

- **Data processing block LUTs.** The data processing is achieved by channels which operate in parallel, processing residues by means of specialized RNS arithmetic units. When index arithmetic is used in the design, index-based Galois field conversion look-up tables used in data processing are not completely full either, also pointed in Fig. 1 for each

![Fig. 1. Basic structure of an RNS system showing LUTs with unused positions.](image-url)
RNS channel, so their unused memory positions may also be used for signature hosting. An advantage of using these look-up tables resides on the fact that they are not a part of the input or output converters, but are part of the main data processing block. Thus, such an IP core containing the embedded signature may be used with another IP cores to form a more complex RNS-based system with global input and output conversion, while retaining IPP capabilities for every block.

- **Output converter LUTs.** The $\varepsilon$-CRT algorithm can be used for the RNS-binary conversion. The look-up tables used in $\varepsilon$-CRT algorithm based output converter include unused positions, which may be exploited for embedding the signature bits, as pointed in Fig. 1. In addition, as these tables are usually part of the system output stage, penalties in the design implementation should not be produced.

In [11] a watermark generating circuit is designed to generate the watermark and embedding it into a soft IP core (behavioral description); this watermark generating circuit is not included as a part of the IP design. The fact that in our technique the signature is embedded in tables that are part of the design, with the signature bits being interlaced with the data required by the circuit to operate, makes extremely difficult the possibility of an attacker finding and removing these signature bits. Problems related with the methods based on the use of tables or logic elements not in use are solved [3, 5-6]. The novelty in this watermarking scheme is that the signature embedding does not require extra hardware and after the chip has been packed, it is still easy to detect the ownership rights of the IP provider, since this IPP scheme includes a method for extracting the signature of the circuit at the physical layout level, different to the signature embedding level. The potential disadvantages are related to possible penalties in circuit performance and area, mainly generated by the signature extraction hardware.

V. WATERMARKING APPROACH

As we said, the IPP procedure consists of two processes: signature embedding and signature extraction. These processes are detailed in this section.

A. Signature embedding

The signature identifying the design (text, image, etc.) is selected and stored in a public domain document. This signature is converted into a bit-stream or digital signature using a cryptographic hash function. The final step of this signature preparation involves the partitioning of the digital signature in blocks. The block length is equal to the memory position width where the signature bits will be embedded. The signature embedding process stores the signature bit blocks into empty positions of the design look-up tables; these positions are called Signature Memory Positions (SMP). There is not a fixed algorithm for spreading the bits through the circuit, but it will depend on each particular RNS design. Thus, this extra effort in the embedding process makes harder to change a few bits of the embedded signature to overcome IP protection. In opposition to other watermarking techniques [3, 6, 11], this signature embedding does not increase the area of the system, neither degrades performance.

B. Signature extraction

The signature extraction requires some extra hardware, basically additional logic, as well as some multiplexers for the overhead routing. The additional logic task is to detect a data input sequence previously selected, called Signature Extraction Sequence (SES). This additional logic will depend on the options to generate the SES. In our IPP technique the additional logic has been realized in two different manners:

- The SES is manually selected, so in this case the additional logic has been realized via a finite state machine (FSM).
- In order to improve SES invulnerability, it is generated using a linear-feedback shift register (LFSR) [22], so a corresponding LFSR, an FSM and some additional hardware are required for the signature extraction, as shown in Fig. 2.

The FSM, in combination with the LFSR when required, activates the signature extraction process every time the SES is detected. The FSM addresses successively each SMP and routes its content to the circuit output. It must be noted again that these SMPs are part of the same LUTs used during normal operation by the RNS DSP hardware. Thus, the binary-to-RNS and RNS-to-binary converters, as well as the rest of the RNS hardware, keep their normal operation working but, during a few clock cycles, the system output consists of different digital signature bit blocks. The digital signature is obtained grouping these signature bit blocks properly.

![Fig. 2. Additional logic for the signature extraction based on a LFSR.](image-url)
Signature validation consists of applying the hash function to the public domain document containing the owner signature who claims his rights. The digital signature obtained is compared with the one extracted from the IP block. The coincidence of both would demonstrate the rights of the reusable module author.

VI. IPP DESIGN EXAMPLES

An 8-tap reconfigurable FIR filter [23] was chosen as study case for signature embedding. This FIR core works externally with 2C data and includes the option to introduce filter coefficients through input ports. Thus, this FIR filter includes 2C-to-index conversion and eight index-based Galois field parallel RNS channels, including index-to-RNS conversion tables. Efficient block decomposition 2C-to-RNS [4, 15], followed by RNS-to-index conversion, and ε-CRT-based [20] RNS-to-2C conversions are used to optimize area and performance, while Galois field arithmetic allows an efficient implementation of filter coefficient multiplications that are combined with a proper addition tree for output generation.

There are three different options for this FIR filter for embedding the bit signature, although only ε-CRT and index-to-RNS LUTs were considered for the current examples. We distinguish between three different IPP designs that are based on the choices for the signature embedding and the options to the additional logic for the signature extraction process.

1) FSM IPP design: this option has a finite state machine for activating the signature extraction process and the signature bits are embedded into empty memory positions of ε-CRT tables.

2) LFSR IPP design: this has the circuit in Fig. 2 for activating the signature extraction process, because of the SES was generated using an LFSR, and the signature bits are also embedded into ε-CRT tables.

3) Embedded LFSR IPP design: it is basically equivalent to the previous LFSR IPP design but the signature bits are embedded within index-to-RNS conversion tables used for each filter tap.

VII. RESULTS

The proposed IPP schemes have been evaluated in terms of area and timing overhead, as well as resistance to attacks. The secure hash function SHA1 was used to convert the selected signature into a 160-bit signature stream.

A. Synthesis results

To evaluate the area and timing overhead of approach, conventional RNS-based FIR filter core data and IPP design examples were synthesized from their VHDL description for Xilinx Virtex2 devices, using ISE 6.2i tools, as well as for a custom IC process. For this last one, circuits have been synthesized too from their VHDL description and optimized for speed and size using Synopsys synthesis tools. Table 1 compares the design before and after the application of the watermarking approach. Analysis of these results shows that throughput penalization is negligible, always less than 1%, while the area increase is never higher than 6.4 % for all the study cases. It must also be noted that our method requires no extra hardware for signature embedding. The area increase, generated by the additional hardware in the signature extraction process, would be less noticeable for more complex applications, since this increase is a fixed quantity for the signature length.

B. Resistance against attacks

A way to evaluate the strength of a given watermark scheme is to assess its resistance to attacks. In watermarking terminology, an attack is any processing that may impair detection of the watermark. There are some general ways of attacking our watermarking scheme, one of them being the detection of the signature in order to know the signature memory positions. In this case, the attacker should know that it is necessary to apply a specific input sequence in order to activate the signature extraction process. Fig. 4 shows the probability of the SES being applied just by chance to the input circuit. This has been calculated considering 16-bit input data, that is the input data length for the FIR core used as example. In FIR-FSM IPP design, the SES chosen was a 6-bit SES, being the study odds equal to 1.3×10⁻²⁹. To increase the IPP invulnerability, a 100-bit SES was chosen to activate signature extraction. For these design examples, FIR-LFSR and FIR-LFSR embedded, the probability to activate the signature extraction by chance is just above 1.0×10⁻³⁰⁰.

VIII. CONCLUSIONS

The proposed watermarking technique creates a protection for RNS-based IP cores, by inserting a digital signature. This protection is achieved at the HDL design level and it embeds a digital signature in empty look-up table positions used in RNS-based designs. In this work, ε-CRT LUTs and index-based Galois look-up tables were

<table>
<thead>
<tr>
<th>TABLE I</th>
<th>Summary of synthesis results</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SLICEs F (MHz) Area F (MHz)</td>
</tr>
<tr>
<td>Non-IPP designs</td>
<td>43688 50.0</td>
</tr>
<tr>
<td>FIR-filter</td>
<td>1642 151.6</td>
</tr>
<tr>
<td>IPP designs</td>
<td>Area up Speed down Area up Speed down</td>
</tr>
<tr>
<td>FIR-FSM</td>
<td>2.4% &lt;1% 1.7% &lt;1%</td>
</tr>
<tr>
<td>FIR-LFSR</td>
<td>3.2% &lt;1% 3.5% &lt;1%</td>
</tr>
<tr>
<td>FIR-LFSR embedded</td>
<td>6.4% &lt;1% 3.9% &lt;1%</td>
</tr>
</tbody>
</table>
proposed for signature hosting. The embedded signature is carried through synthesis, placement and routing processes allowing efficiently enabling IPP for RNS-based IP cores. Efficient signature extraction scheme is also developed by means of a finite state machine. In order to increase resistance against attacks, this scheme is improved generating the SES by means of LFSR usage and including this LFSR into the signature extraction required hardware. This improved signature extraction process allows longer SES with minimal area overhead, making more difficult signature detection by intruders. A 160-bit signature was embedded in an RNS-based FIR filter, with design examples being implemented on Xilinx devices as well as a custom IC process. Synthesis results show that the proposed IPP scheme results in negligible degradation of system performance and very low area penalties for a variety of embedded signature and signature extraction hardware options.

ACKNOWLEDGMENT

The authors wish to acknowledge financial support from the Dirección General de Investigación (Spain) under project TIC2002-02227. CAD tools and supporting material were provided by Xilinx Inc. and Synopsys Inc. (ID 10806) through their University Programs.

REFERENCES


![Figure 3. Chance to find the SES](image-url)