Residue Number System Arithmetic based on Integrated Nanophotonics

Jiaxin Peng, Sun Shuai, Prof. Volker Sorger, Prof. Tarek El-Ghazawi, Advisor

Department of Electrical and Computer Engineering, The George Washington University

Introduction

Residue number system (RNS) decomposes a large number into smaller ones by utilizing the residues of a set of moduli. In the field of digital computer arithmetic, RNS takes advantages in decomposing the larger integers into a set of smaller integers in calculation independently (without carry propagation) and in parallel. Fundamentally, adapting photonics into the RNS arithmetic for signal process could benefit from:

- the fast execution time which is given by the photon's time-of-flight through the structure;
- the nature of light that a photon always has to propagate with a momentum which in other words, an operation could be computed while switching;
- the wavelength division multiplexing (WDM) capable of RNS computing units achieve highly instruction level parallelism with broadband nanophotonic devices.

In this study, we proposed a hybrid photonic-plasmonic (HPP) RNS adder with all-to-all sparse directional (ASD) structure, based on cascaded HPP 2×2 switches forming a crossbar with broad spectrum operating bandwidth [1].

Residue Number System

Residue number system uses remainders of different moduli to describe a given number [2]. It is a method to record a number, similar to decimal. However, it is a mixed radix number system, which means each digit has its own radix.

Residue Number

- A number ***X*** is represented by its *residue*, or remainder, remainder obtained by dividing it by a *modulus* ***M***
- If X=96, M=11, then $|96|_{11} = 96 \bmod 11 = 8$

Residue Number System

- A number ***X*** is represented by its *residue*, or remainder, remainder obtained by dividing it by a set of *moduli* $\boldsymbol{M_i}$
- If X = 96, Y= 32, M_i = {11,19,23}, then
  $|96|_{[11,19,23]} = \{8,1,4\}_{[11,19,23]}$
  $|32|_{[11,19,23]} = \{10,13,9\}_{[11,19,23]}$
- The set of moduli $\{M_1, M_2, \ldots, M_n\}$ should be relative prime
- There are $\prod M_i$ states in total. Normally we will use it to represent number 0 to $\prod M_i - 1$

RNS Arithmetic

- Example: X+Y = 96 + 32 = 128

$$\begin{array}{l} \{8,\ \ 1,\ \ 4\} \\ \underline{+\{10,\ 13,\ 9\}} \\ \{18,\ 14,\ 13\} \\ =\{7,\ \ 1,\ \ 13\}_{[11,19,23]} \end{array}$$

$$|128|_{[11,19,23]} = \left\{ \begin{array}{l} 128 \bmod 11 \\ 128 \bmod 19 \\ 128 \bmod 23 \end{array} \right\} = \{7, 1, 13\}_{[11, 19, 23]}$$

- No carry and independent
- Good for high-performance computing
- Easy to compute in parallel
- Applicable in addition, subtraction and multiplication

Methods

2×2 Hybrid Photonic-Plasmonic Switch

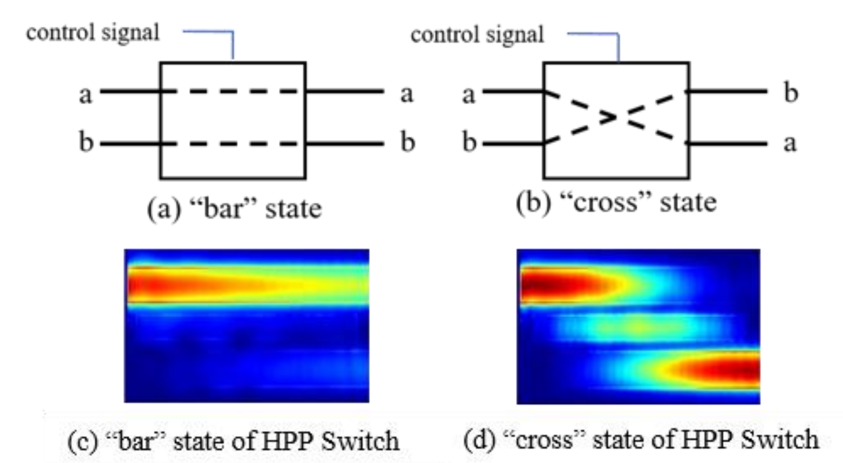

Figure 1. Two States of a 2×2 Switch. (a) and (b): the conceptual schematic of two states in a 2×2 switch. (c) and (d): the top view of the FDTD simulation results of two states in our HPP switch with two silicon waveguides (up and down) as buses and a switching island covered by indium tin oxide in between to achieve signal switching [3].

All-to-all sparse directional Modulo-5 Adder/Multiplier

The first RNS adder design uses mesh grid, utilizing M(M-1) 2×2 switch [4]. Each operation uses (M-1) switch at one time, which wastes resources. Here we proposed a ASD RNS adder and multiplier with less switch.

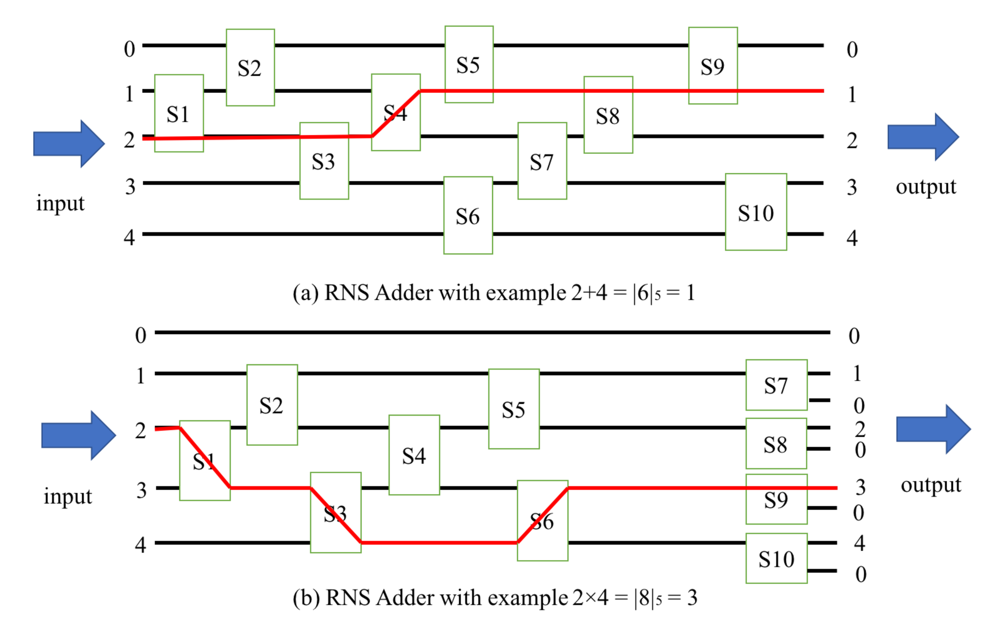

Figure 2. Modulo-5 RNS Adder (a) and Multiplier (b) with Examples.

- "+4": states for S_1 to S_{10} are "BCBCBBCBBC"
- "×4": states for S_1 to S_{10} are "CCCCCCBBBB"
- B/C represents bar/cross state

Lumerical FDTD and Interconnect

To evaluate our HPP switch, a single switch is implemented in Lumerical FDTD. Two states of HPP switch are shown as Figure 1 (c) and (d). To evaluate our adder and multiplier design, Lumerical Interconnect is used to measure the overall result.

Lumerical is a simulation software that focuses on optical side. FDTD concentrates on the device design while the Interconnect provides the connection with different optical components.

Results

RNS Modulo-5 Adder/Multiplier Performance

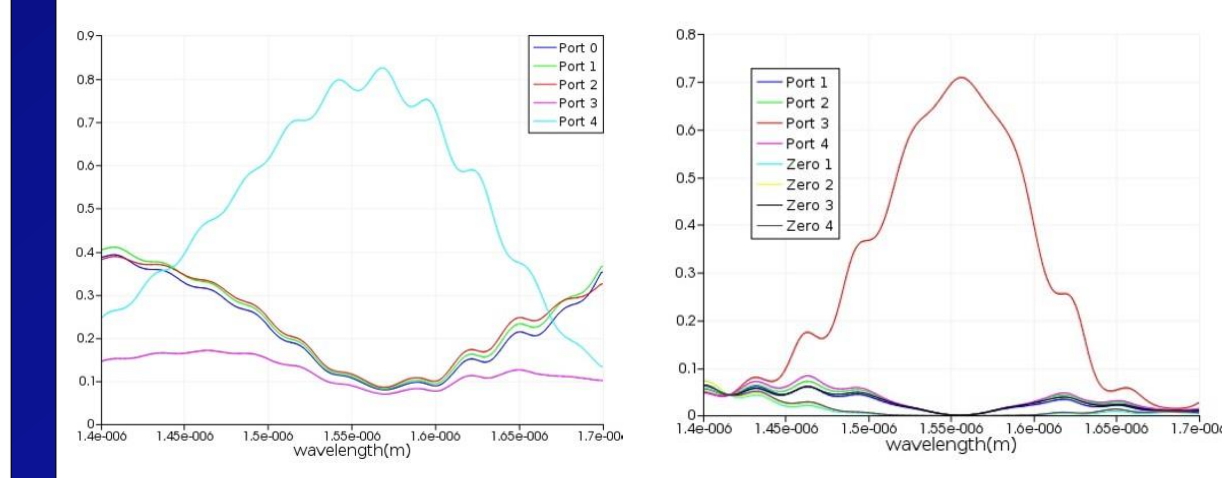

Figure 3. Performance of operation $|2+2|_5 = 4$ (left), and operation $|2×4|_5 = 3$ (right).

Modulo-M RNS Adder Performance

- Table 1 shows the components requirement of architectural design of Mesh and ASD RNS models with modulo-M system
- Figure 4 shows the speed, energy, area, and speed-energy-area product (SEAP) of both design
- Optical component including micro ring resonator (MMR) [5], Mach Zehnder interferometer (MZI) [6], All-Optical switch (AOS) [7], and hybrid photonic-plasmonic (HPP) ITO switch

Parameter	Mesh RNS Model [4]	ASD RNS Model
# of optical component	M(M-1)	$(M-1)^2/2+2$
# of control circuit	M	$(M-1)^2/2+2$
Logic circuit	-	$(M-1)^2/2+2$ MUX $(M-1)^2/2+2$ NAND gates $(M-1)^2/2+2$ AND gates

Table 1. Comparison of mesh RNS model and ASD RNS model in architecture design.

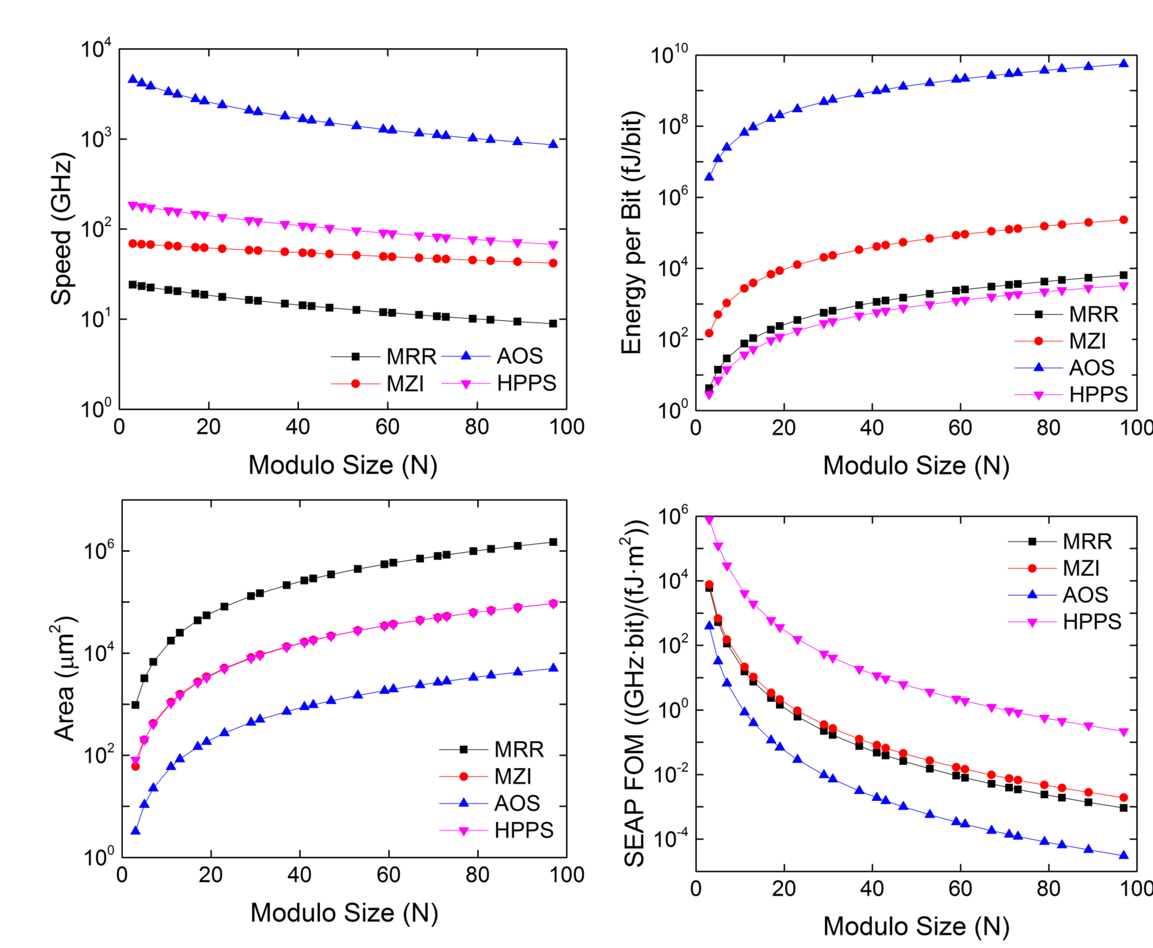

Figure 4. Comparison of different type of optical switch in two designs

Application

Our proposed HPP device has a further feature that it is WDM capable. Here, we implemented a new design for RNS adder and multiplier, with additional ring resonators and photo-detectors at the end of outputs (Figure 5). It allows multiple operations to perform simultaneously, increasing the system efficiency. Several bunches of light could be identified to corresponding operations.

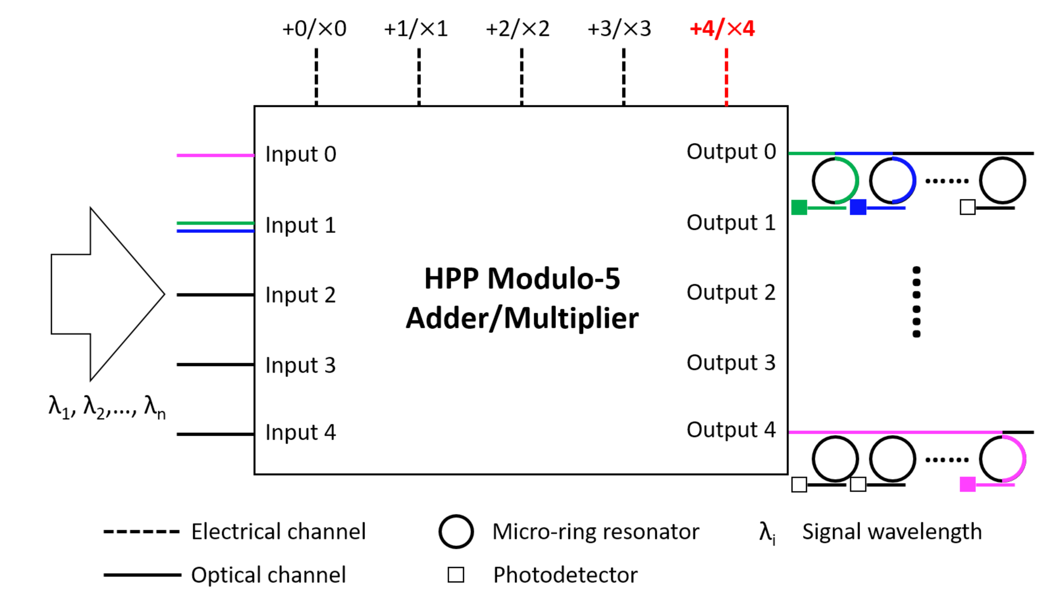

Figure 5. WDM Modulo-5 RNS Schematic

With additional micro-ring and photon-detectors, multiple operations could be computed simultaneously. Every time the change of switches states has a response time. Therefore, our design is ideal for convolutional neural network, which has millions of multiplication-accumulation computation (MAC). Only one time set up allows multiple calculation, which increase the efficiency dramatically.

Conclusion

Here we show a photonic residue number system (RNS) adder and multiplier based on an all-to-all, non-blocking, sparse directional crossbar. The RNS arithmetic is synergistically implemented by spatial routing of light using nanophotonic 2×2 switching building blocks, thus enabling a highly parallel compute engine. This one-shot programmable photonic processor utilized a extremely short execution time, only limited by the picosecond short time-of-flight through the 10's of micrometer compact optical router.

References

1. J. Peng, S. Sun, V. Narayana, V. J. Sorger, and T. El-Ghazawi, "Residue Number System Arithmetic based on Integrated Nanophotonics", https://arxiv.org/abs/1712.00049 (2017)
2. H. L. Garner, "The residue number system," IRE Transactions on Elec- tronic Computers pp. 140–147 (1959).
3. S. Sun,V. Narayana, I. Sarpkaya, J. Crandall, R.A.Soref, H.Dalir, T.ElGhazawi, and V. J. Sorger, "Hybrid photonic-plasmonic non-blocking broadband 5× 5 router for optical networks", IEEE Photonics Journal (2017).
4. A. Tai, I. Cindrich, J. R. Fienup, and C. Aleksoff, "Optical residue arithmetic computer with programmable computation modules," Applied optics 18, 2812–2823 (1979).
5. Baba, S. Akiyama, M. Imai, N. Hirayama, H. Takahashi, Y. Noguchi, T. Horikawa, and T. Usuki, "50-gb/s ring-resonator-based silicon modu- lator," Optics express 21, 11869–11876 (2013).
6. C. Haffner, W. Heni, Y. Fedoryshyn, J. Niegemann, A. Melikyan, D. El- der, B. Baeuerle, Y. Salamin, A. Josten, U. Koch et al., "All-plasmonic mach–zehnder modulator enabling optical high-speed communication at the microscale," Nature Photonics 9, 525–528 (2015).
7. C. Min, P. Wang, C. Chen, Y. Deng, Y. Lu, H. Ming, T. Ning, Y. Zhou, and G. Yang, "All-optical switching in subwavelength metallic grating structure containing nonlinear optical materials," Optics letters 33, 869– 871 (2008). residue number system arithmetic based on integrated nanophotonics