# HOST 2026 Tutorial:
# Part I
# Lightweight Cryptography and Mathematics Overview
# May 4, 2026

*Arman Allahverdi and *^Vincent John Mooney III

^Associate Professor, *School of Electrical and Computer Engineering

^Adjunct Associate Professor, School of Computer Science

**Georgia Institute of Technology**

**Atlanta, Georgia**

Georgia Tech

# Acknowledgement


- This work has been partially supported by the U.S. Department of Energy (DoE) Office of Cybersecurity, Energy Security, and Emergency Response (CESER) under Cybersecurity for Energy Delivery Systems (CEDS) Agreement Number #DE-CR0000055 to the Georgia Tech Research Corporation: GRIDLOGIC: Hardware/Software Codesign for Deep Grid Visibility and Security

# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

Georgia Tech

# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

Georgia Tech

# Introduction

- Arman Allahverdi
  - B.S. in Electrical Engineering from the University of New Mexico, Albuquerque, New Mexico, 2021
  - M.S. in Electrical and Computer Engineering from The Georgia Institute of Technology, Atlanta, Georgia, 2024
  - 4th-year Ph.D. student in Electrical and Computer Engineering (Hardware-Software Codesign for Security) at The Georgia Institute of Technology

# Introduction

- Vincent John Mooney III
  - B.S. in Electrical Engineering, Yale University, 1991
  - B.S. in Computer Science, Yale University, 1991
  - M.S. In Electrical Engineering, Stanford University, 1994
  - M.A. in Philosophy, 1997
  - Ph.D. in Electrical Engineering, 1998
  - Associate Professor, School of Electrical and Computer Engineering, The Georgia Institute of Technology
  - Adjunct Associate Professor, School of Computer Science, The Georgia Institute of Technology

# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

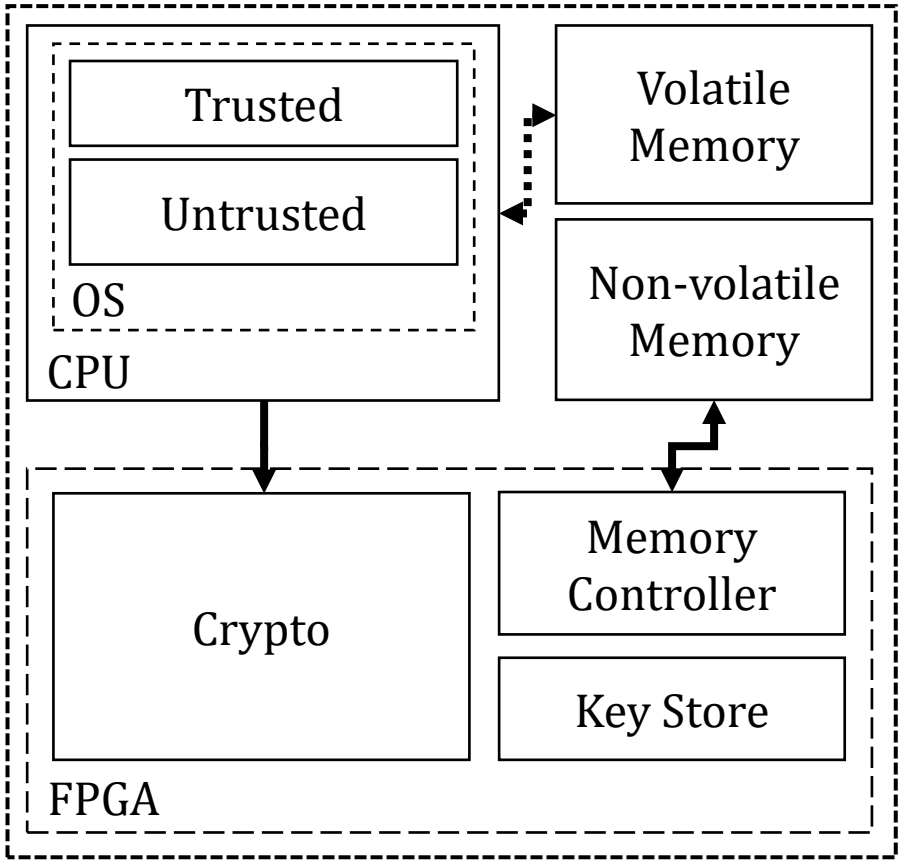
Georgia Tech

# Research Overview

- With the advent of battery powered and ultra-small devices requiring security, **lightweight cryptography has become an area of research interest**, for example with the proposal of hardware-oriented cryptographic schemes in the NIST Lightweight Cryptography Competition (2018-2023)
- Nonlinear Feedback Shift Registers (NLFSRs) are a valuable building block for lightweight cryptographic hardware, as they generate periodic nonlinear state sequences and can be efficiently implemented in hardware
  - NLFSRs do not automatically scale with register size, and so as a result there is no efficient method of building an NLFSR of an arbitrary size

# Research Impact

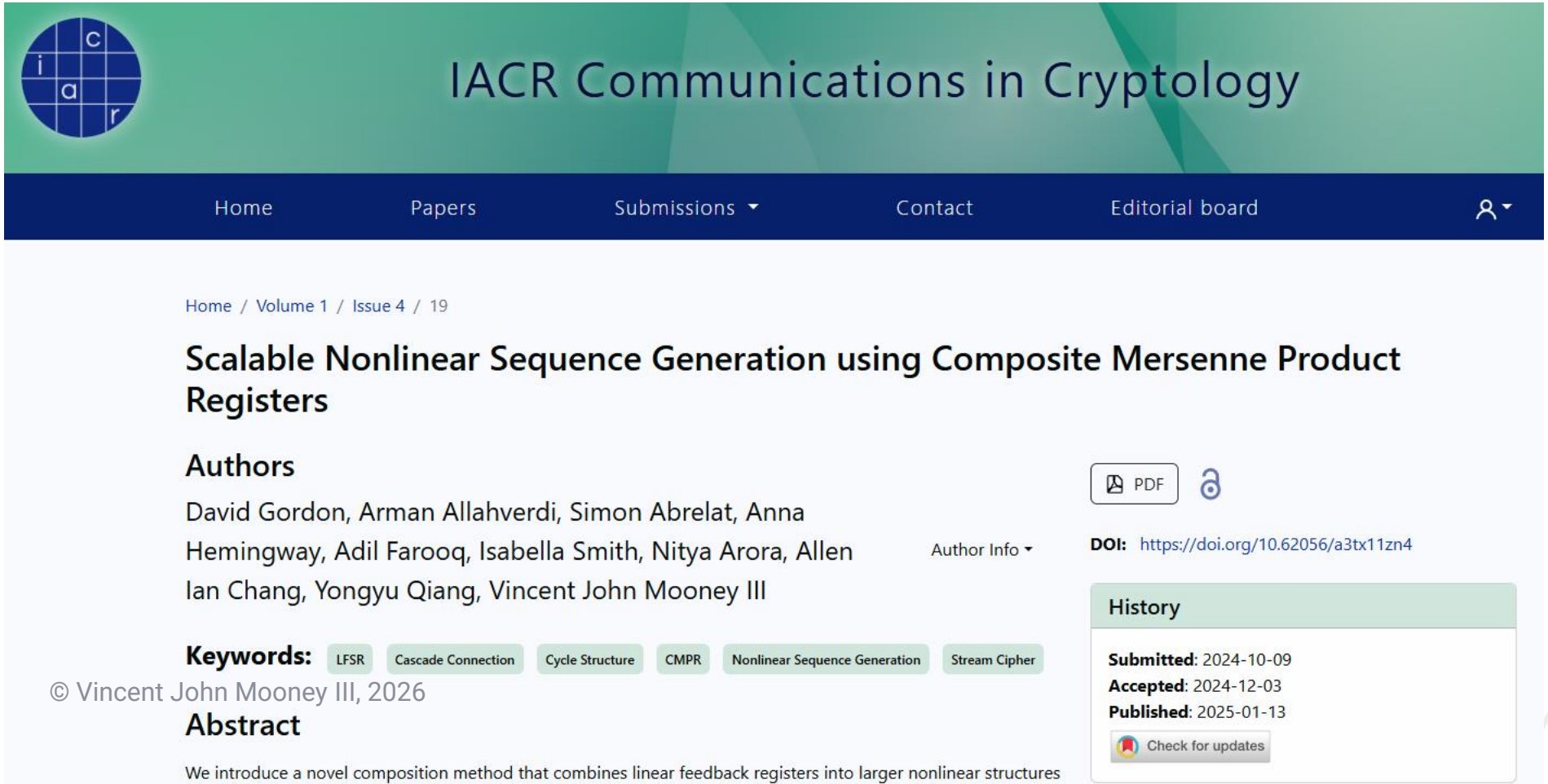
- We have discovered a new, scalable class of nonlinear feedback register that can outperform the prior state-of-the-art in lightweight cryptography
  - Scalable to large values (e.g., 256, 1024 and larger) based on Mersenne primes
  - Mathematical proof of exponential expected period
  - Hardware-efficient due to register-based construction
  - Exponential number of hardware constructions allow for the key to be synthesized into the register structure, thus producing very low area in reconfigurable logic compared to other lightweight options




- In summary, CMPRs provide the first nonlinear feedback structure scalable to thousands of bits of size with provable exponential state and a similarly exponential number of distinct CMPR structures

# IACR Communications in Cryptology Publication

[1] D. Gordon, A. Allahverdi, S. Abrelat, A. Hemingway, A. Farooq, I. Smith, N. Arora, A. Chang, Y. Qiang and V. Mooney, "Scalable Nonlinear Sequence Generation using Composite Mersenne Product Registers," IACR Communications in Cryptology, 1(4), pp. 1-77, January 2025, https://doi.org/10.62056/a3tx11zn4.



IACR Communications in Cryptology

Home Papers Submissions Contact Editorial board

Home / Volume 1 / Issue 4 / 19

**Scalable Nonlinear Sequence Generation using Composite Mersenne Product Registers**

**Authors**

David Gordon, Arman Allahverdi, Simon Abrelat, Anna Hemingway, Adil Farooq, Isabella Smith, Nitya Arora, Allen Ian Chang, Yongyu Qiang, Vincent John Mooney III

Author Info

PDF

**DOI:** https://doi.org/10.62056/a3tx11zn4

**Keywords:** LFSR | Cascade Connection | Cycle Structure | CMPR | Nonlinear Sequence Generation | Stream Cipher

**History**

**Submitted**: 2024-10-09
**Accepted**: 2024-12-03
**Published**: 2025-01-13

Check for updates

**Abstract**

We introduce a novel composition method that combines linear feedback registers into larger nonlinear structures





Georgia Tech

# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

Georgia Tech

# Lightweight Cryptography

- **NIST Lightweight Cryptography Workshops:** 2015-2023
  - **Goal:** Get feedback on industry need, relevant applications, and security requirements
- **NIST Lightweight Cryptography Competition (LWC):** 2018-2023 [3]
  - **Goal:** Source secure, hardware-efficient symmetric-key algorithms from the cryptography community
  - **Call for Algorithms:** 2018
  - **Scope of Algorithms:** encryption, authentication, hashing
  - Candidate algorithms selected in several phases
  - Evaluation process concerned with (i) hardware + software efficiency, (ii) security analysis, and (iii) overall flexibility and additional features (e.g., different modes of operation)
  - Ascon [4] selected as winner out of 10 finalists

# Lightweight Cryptography

- **NIST Lightweight Cryptography Competition:** 2018-2023 [3]
  - Candidate algorithms required to support authenticated encryption with associated data (AEAD)
  - Minimum key size of 128 bits required
    - At least $\approx 2^{128}$ security against key recovery attacks
  - Candidate algorithms proposing a hash function were required to have at least $2^{112}$ security against a classical computer

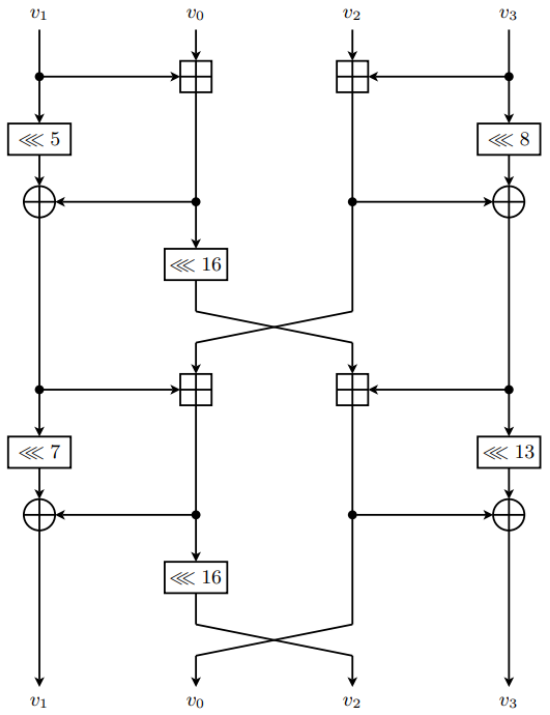
| AEAD-only | AEAD + Hashing |
|---|---|
| Grain-128AEADv2 | Ascon |
| GIFT-COFB | PHOTON-Beetle |
| ISAP | SPARKLE |
| TinyJAMBU | Xoodyak |
| Elephant | Romulus |

**Cryptographic Capabilities of the 10 NIST LWC Finalists**

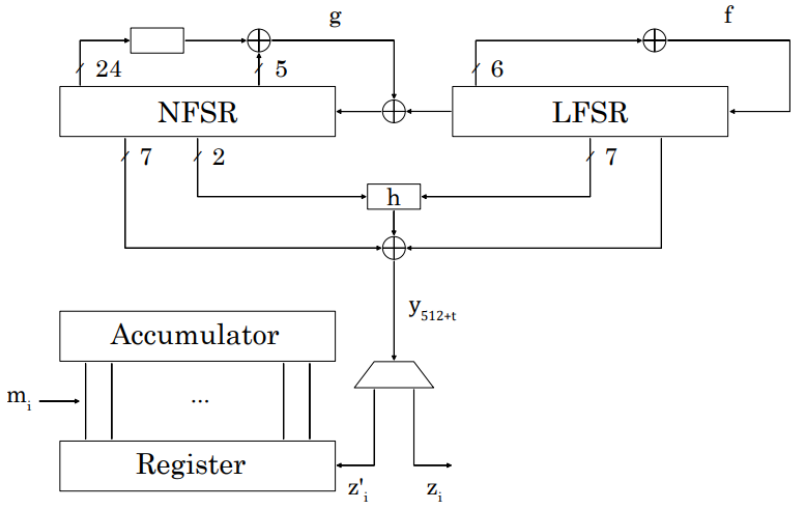
 

# Lightweight Cryptography

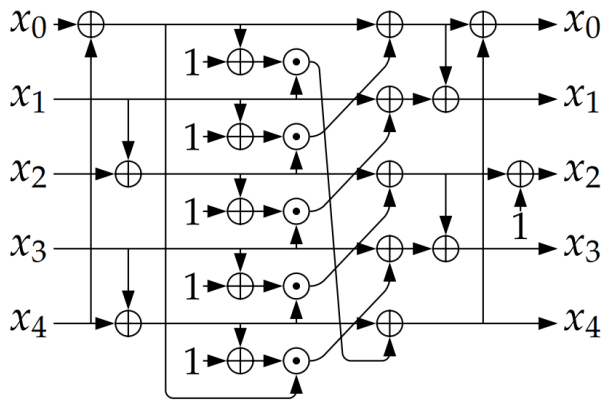
- Different hardware paradigms for lightweight cryptosystem design:
  - ARX (Add-Rotate-XOR)
  - Register-based (LFSRs, NLFSRs, and this work)
  - S-Boxes implemented as Boolean circuits (ex: Ascon)




Chaskey [5] (ARX)




Grain-128AEADv2 [6] (Register-Based)




Ascon [4] S-Box





Georgia Tech

# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

Georgia Tech

# Terminology

- **Mersenne Exponent:** An integer $n$ such that $2^n - 1$ is prime
- **Mersenne Prime:** A prime number in the form $2^n - 1$
- **Feedback Register:** A form of register where the next state, $A[t+1]$, is governed by a function $f$ of the current state $A[t]$ (e.g., $A[t+1] = f(A[t])$
- **Feedback Shift Register:** A form of feedback register where the output of each state bit $a_i$ is connected to the input of the next bit $a_{i+1}$
- **Linear Feedback Shift Register (LFSR):** A feedback shift register where successive states are determined by a linear function of the current state
- **Nonlinear Feedback Shift Register (NLFSR):** A feedback shift register where successive states are determined by a nonlinear function of the current state
- **Feedback Polynomial:** A polynomial that determines the feedback taps of a feedback register
- **Primitive Polynomial:** An irreducible (unfactorable) polynomial of degree $n$ over a finite field $GF(p)$ whose roots are primitive elements of the extension field $GF(p^m)$

# Terminology (continued)

- **Product Register:** A feedback register that performs Galois Field Multiplication of the current state $A[t]$ and an update polynomial $U$, modulo a feedback polynomial $P$
- **Mersenne Product Register (MPR):** A product register of size $n$, where $n$ is a Mersenne exponent
- **Composite Mersenne Product Register (CMPR):** A feedback register formed by several MPRs combined through nonlinear chaining functions
- **Stream Cipher:** An encryption scheme that encrypts data by performing an XOR with a pseudorandom bitstream
- **Keystream:** A pseudorandom bitstream generated from a secret key and a public initialization vector (IV)
- **Chaining Density:** The proportion of the bits of a CMPR that receive chaining functions
- **Initialization Round:** A single clocking of a feedback register (application of the next state function to the feedback register)
- **Z-transform:** A mathematical transformation of a sequence in discrete time into the frequency domain, i.e., a complex-valued representation of the sequence [13]

# Mersenne Examples

- 13 is a Mersenne exponent because $2^{13} - 1 = 8191$ which is a prime number
- 13 is not a Mersenne prime because there is no integer $n$ such that $2^n - 1 = 13$
- 11 is a prime number that is neither a Mersenne exponent nor a Mersenne prime
- 7 is both a Mersenne exponent and a Mersenne prime
- The first 20 Mersenne exponents are 2, 3, 5, 7, 13, 17, 19, 31, 61, 89, 107, 127, 521, 607, 1279, 2203, 2281, 3217, 4253 and 9689
- 8191 is a Mersenne prime (with Mersenne exponent 13) but is not a Mersenne exponent
- All Mersenne exponents are prime numbers

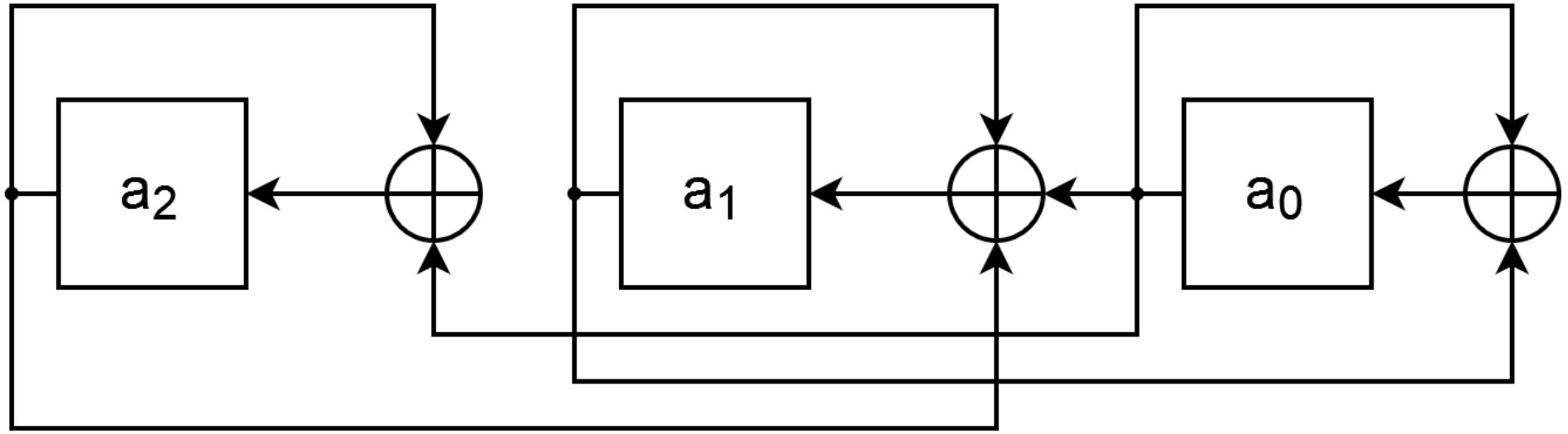
# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

Georgia Tech

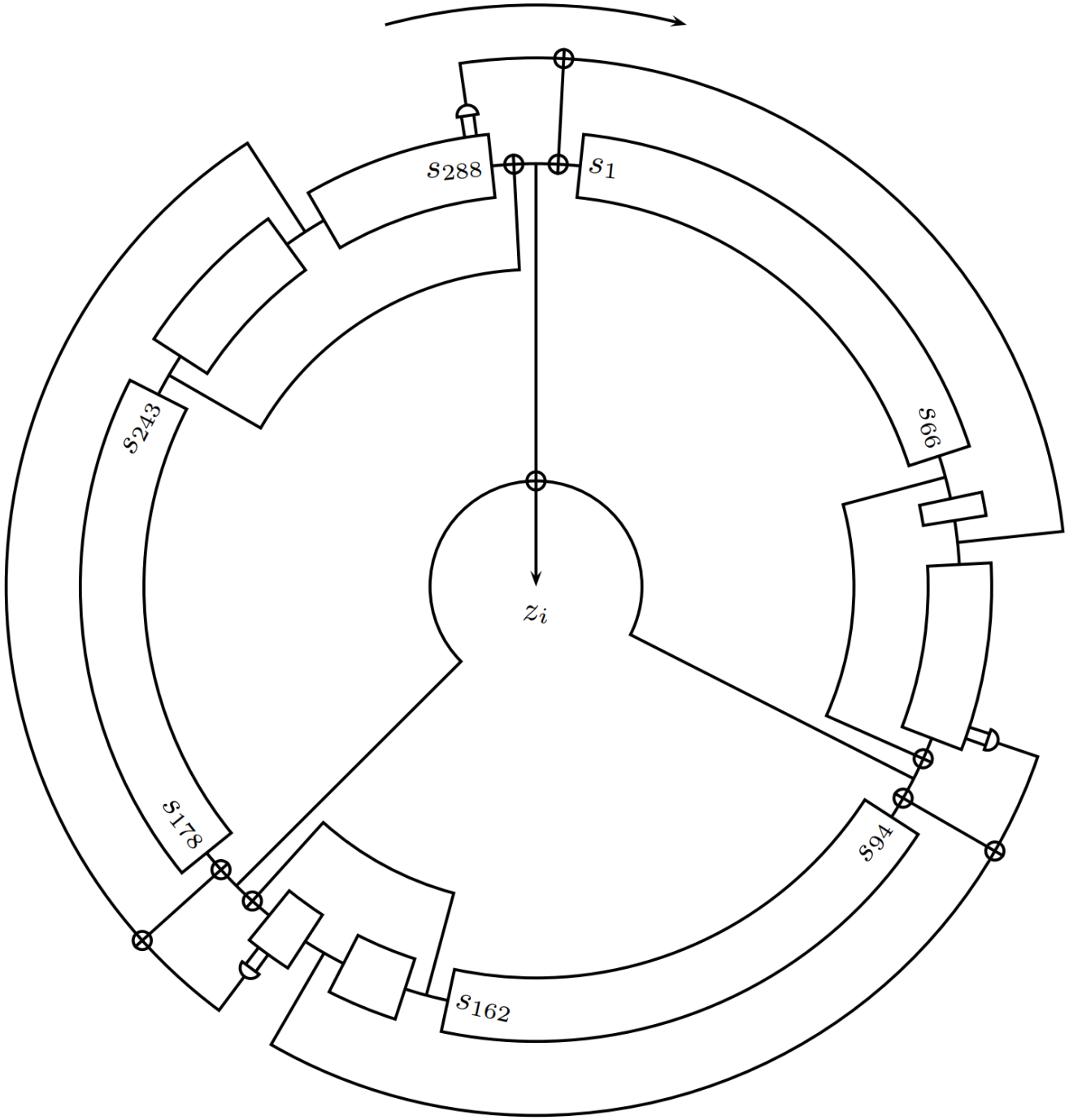
# Feedback Register Definition and Example

- **Feedback Register:** A form of register where the next state, $A[t+1]$, is governed by a function $f$ of the current state $A[t]$ (e.g., $A[t+1] = f(A[t])$)



# Feedback Shift Register Definition and Example

- **Feedback Shift Register:** A form of feedback register where the output of each state bit $a_i$ is connected to the input of the next bit $a_{i+1}$
- **Nonlinear Feedback Shift Register (NLFSR):** A feedback shift register where successive states are determined by a nonlinear function of the current state



# Linear Feedback Shift Register Definition and Examples

- **Linear Feedback Shift Register (LFSR):** A feedback shift register where successive states are determined by a linear function of the current state
- **Feedback Polynomial:** A polynomial that determines the feedback taps of a feedback register

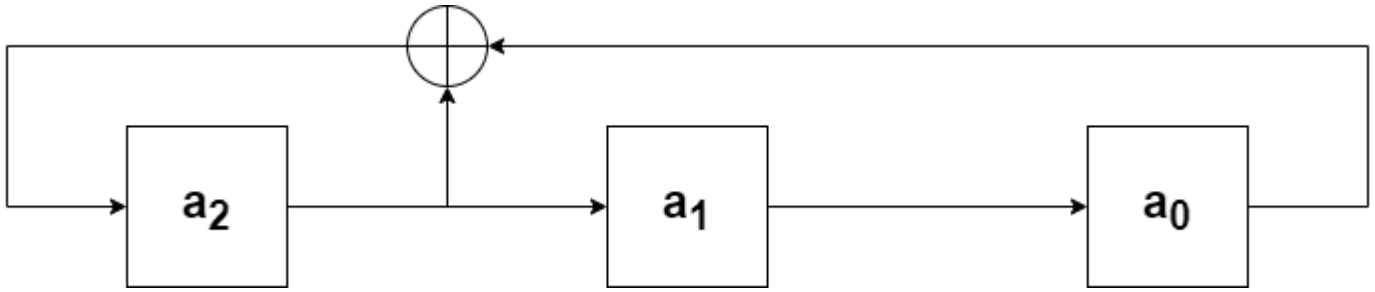



**Figure 1(a): LFSR in Fibonnaci Configuration [1]**

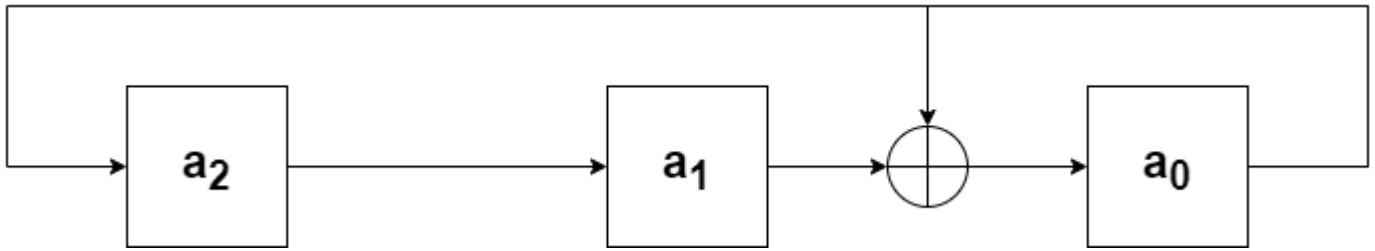
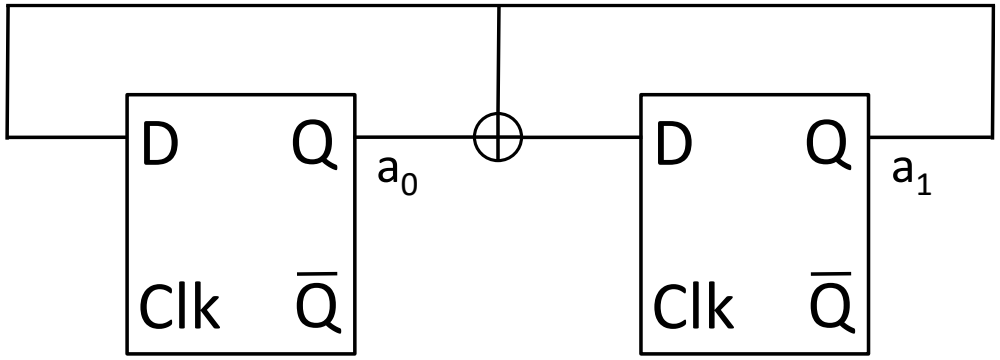


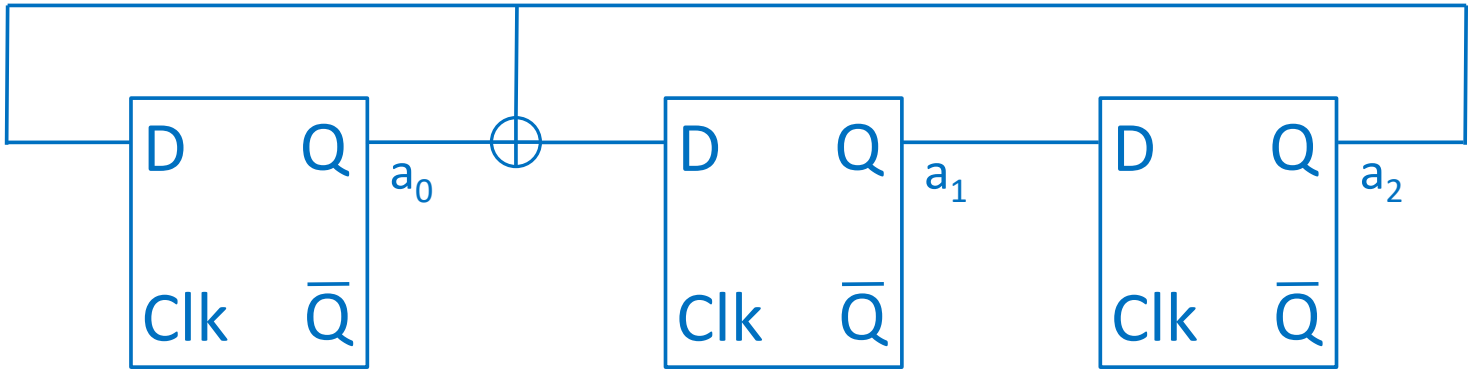

**Figure 1(b): LFSR in Galois Configuration [1]**

# Linear Feedback Shift Register Examples

- **Primitive Polynomial:** An irreducible (unfactorable) polynomial of degree $n$ over a finite field $GF(p)$ whose roots are primitive elements of the extension field $GF(p^m)$
- LFSR, P=$x^2$+x+1, two cycles: {01,11,10} and {00}




- LFSR, P= $x^3$+x+1, two cycles: {000} and {001,010,100,011,110,111,101}



# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

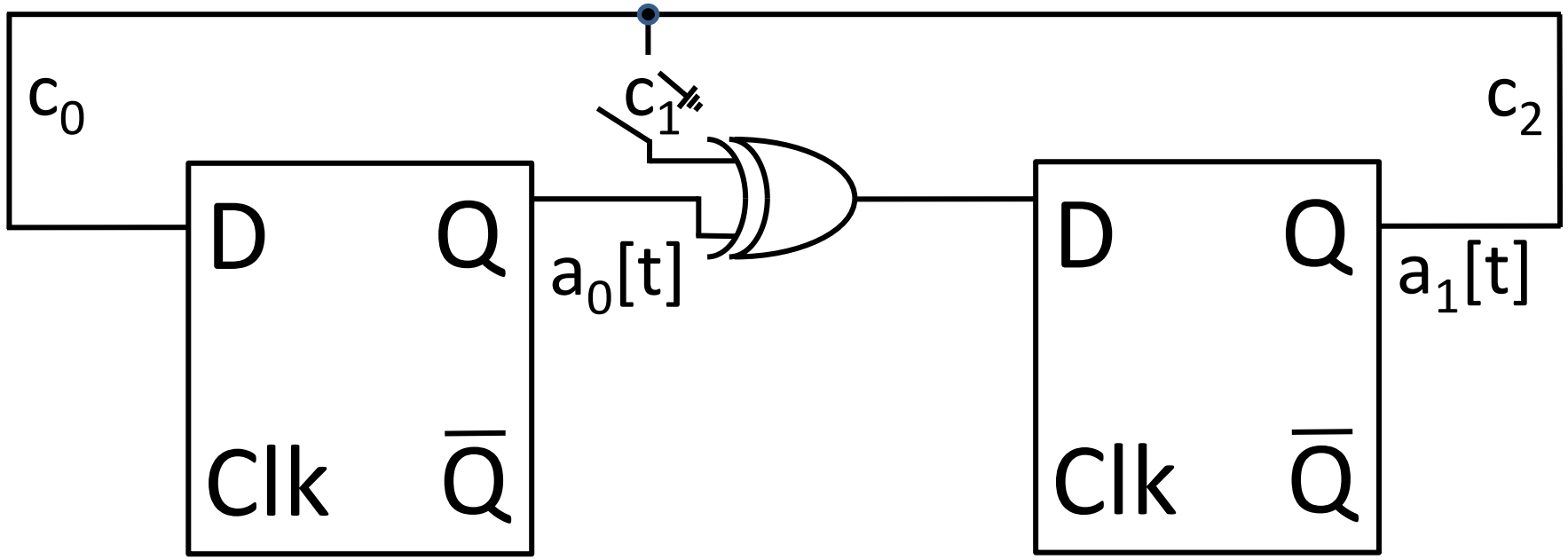
Georgia Tech

# Polynomial Representation

- Galois tried to find the roots of the quintic equation $a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0 = 0$ using the coefficients for a general formula, similar to $ax^2 + bx + c$ where the quadratic formula is expressed in terms of *a*, *b* and *c*
- Can view the bits in a feedback shift register as coefficients in a polynomial equation where $x^5$, $x^4$, $x^3$, $x^2$, $x^1$, $x^0$, etc., are placeholders (i.e., not evaluated or substituted for with numbers)
- Multiplication by x, modulus the characteristic polynomial, calculates the next state
  - $f(x)[t] = \sum_{i=0}^{n-1} a_i[t]x^i$
  - $P(x) = \sum_{i=0}^{n} c_i x^i$
  - $f(x)[t+1] = xf(x)[t] \bmod P(x)$

# Example Computation in GF($2^n$)

- Can be quickly implemented in hardware with feedback shift registers




- $f(x)[t] = a_1[t]x + a_0[t]$ , $P(x) = c_2x^2 + c_1x + c_0$ , $f(x)[t+1] = xf(x)[t] \bmod P(x)$

# Feedback Registers, Field Theory and Periodicity

- A feedback register is a register $A$ such that on each clock cycle (state update), the register state $A[t]$ is updated according to some function $f$, such that:
  - $\boldsymbol{A[t+1] = f(A[t])}$
- In the prior work, feedback registers show up in many different forms:
  - **LFSRs:** $f$ is a linear function of $A[t]$, **can be used as a component in lightweight cryptography provided nonlinear components are also present**
    - **Two Possible Configurations:** Galois ("Internal-XOR"), Fibonacci ("External-XOR")
    - This presentation focuses on the Galois configuration
  - **NLFSRs:** $f$ is a nonlinear function of $A[t]$, **can be used as a standalone component in lightweight cryptography**
  - **De Bruijn Sequences, T-Functions, Cross-Joined Pairs [1]:** Derived from LFSR and NLFSR theory, imposing special restrictions on $f$
- LFSRs and NLFSRs are parametrized by their feedback polynomials $P(x)$
- For a Galois LFSR, the state updates according to:
  - $\boldsymbol{A[t+1] = xA[t] \bmod P(x)}$

# Background: Feedback Registers and Periodicity

- Generally, an $n$-bit feedback register can hold $2^n$ possible unique states
- The **period** of a feedback register refers to the number of unique states the register assumes before returning to its starting state, but it is not always possible to reach a period of $2^n$
- Most LFSRs and NLFSRs have a maximum attainable period of $2^n - 1$ for an $n$-bit register size
  - Maximum/full-period LFSRs can be constructed by ensuring $P(x)$ is a primitive polynomial
  - Full-period NLFSRs exist for small register sizes [7]
  - **No scalable methodology or mathematical process for constructing full-period NLFSRs of arbitrary size, with limited exceptions:**
    - For example, cross-joined pairs [8] can be used to construct larger full-period NLFSRs; however, such NLFSRs have limitations: at most half of the state updates are nonlinear, and the feedback polynomial must depend only on linearly-updated state variables

# Background: Feedback Register-Based LWC

| Design | Capabilities | Components | Security |
|---|---|---|---|
| TRIVIUM [9] | Encryption | NLFSR | $2^{80}$ (Key) |
| Espresso [10] | Encryption | NLFSR (Cross-Joined) | $2^{128}$ (Key) |
| Grain-128AEADv2 [6] | AEAD | LFSR + NLFSR | $2^{128}$ (Key)<br>$2^{64}$ (Tag) |
| Quark [11] | Hashing | LFSR + NLFSRs | Varies<br>$2^{64}$ up to $2^{112}$<br>(Collision Resistance) |

- Cryptographic primitives based on LFSRs and NLFSRs have lightweight hardware implementations
- Limitations:
  - NLFSRs require many clock cycles for the internal state to appear randomized
  - Scaling NLFSRs to support different security levels is not straightforward

# Terminology (cont'd)

- **Product Register:** A feedback register that performs Galois Field Multiplication of the current state $A[t]$ and an update polynomial $U(x)$, modulo a feedback polynomial $P(x)$:
  - $A[t+1] = U(x)A[t] \; mod \; P(x)$

256-bit Product Register

$$U(x) = x^2$$

$$P(x) = x^{256} + x^{213} + x^{133} + x^{115} + x^{48} + x^{12} + 1$$

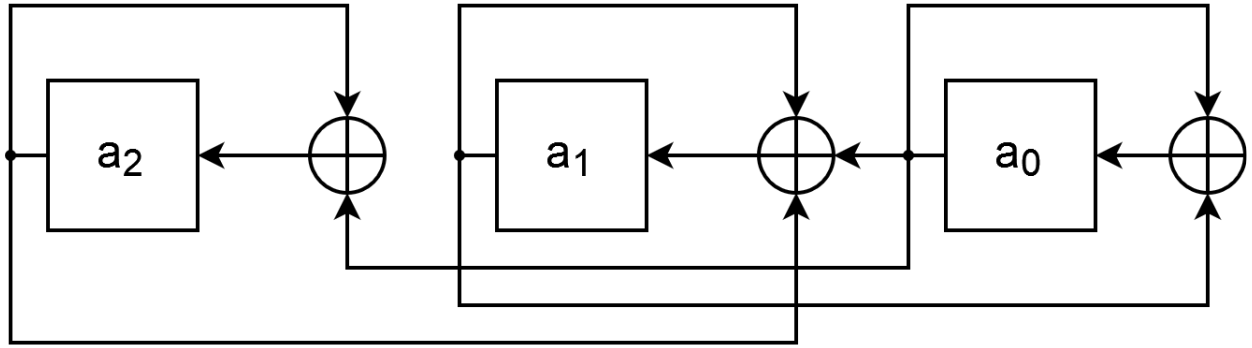
**An Example of a 256-bit PR with a Primitive Feedback Polynomial**

**State Update:**

$$A[t+1] = x^2\left(a_{255}x^{255} + \cdots + a_o\right) \; mod \; P(x)$$

# Terminology (cont'd)

- **Mersenne Product Register (MPR):** A product register of size $n$, where $n$ is a Mersenne exponent




- **Composite Mersenne Product Register (CMPR):** A feedback register formed by several MPRs combined through nonlinear chaining functions

# Terminology (cont'd)

- **Stream Cipher:** An encryption scheme that encrypts data by performing an XOR with a pseudorandom bitstream
- **Keystream:** A pseudorandom bitstream generated from a secret key and a public initialization vector (IV)
- **Chaining Density:** The proportion of the bits of a CMPR that receive chaining functions
- **Initialization Round:** A single clocking of a feedback register (application of the next state function to the feedback register)
- **Z-transform:** A mathematical transformation of a sequence in discrete time into the frequency domain, i.e., a complex-valued representation of the sequence [13]

# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

Georgia Tech

# Definitions

- **Definition 1:** A discrete-time finite-state dynamical system is describe by the pair $(S, f)$ where $S$ is a finite set of states for the system and $f: S \to S$ is a function which describes how the system evolves at each time step; we further denote by  the state of the system at time $t$, with the initial state of the system being at time $t = 0$

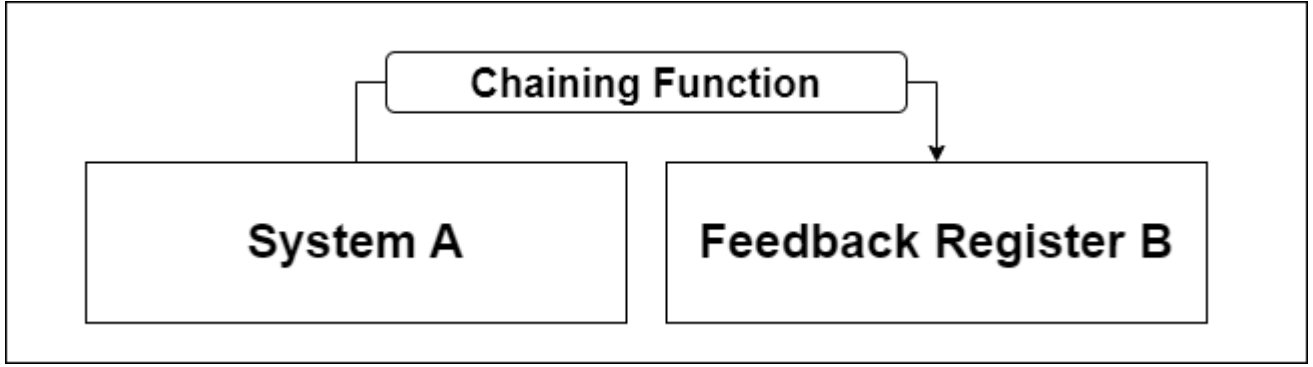



Figure 2(a) from [1]

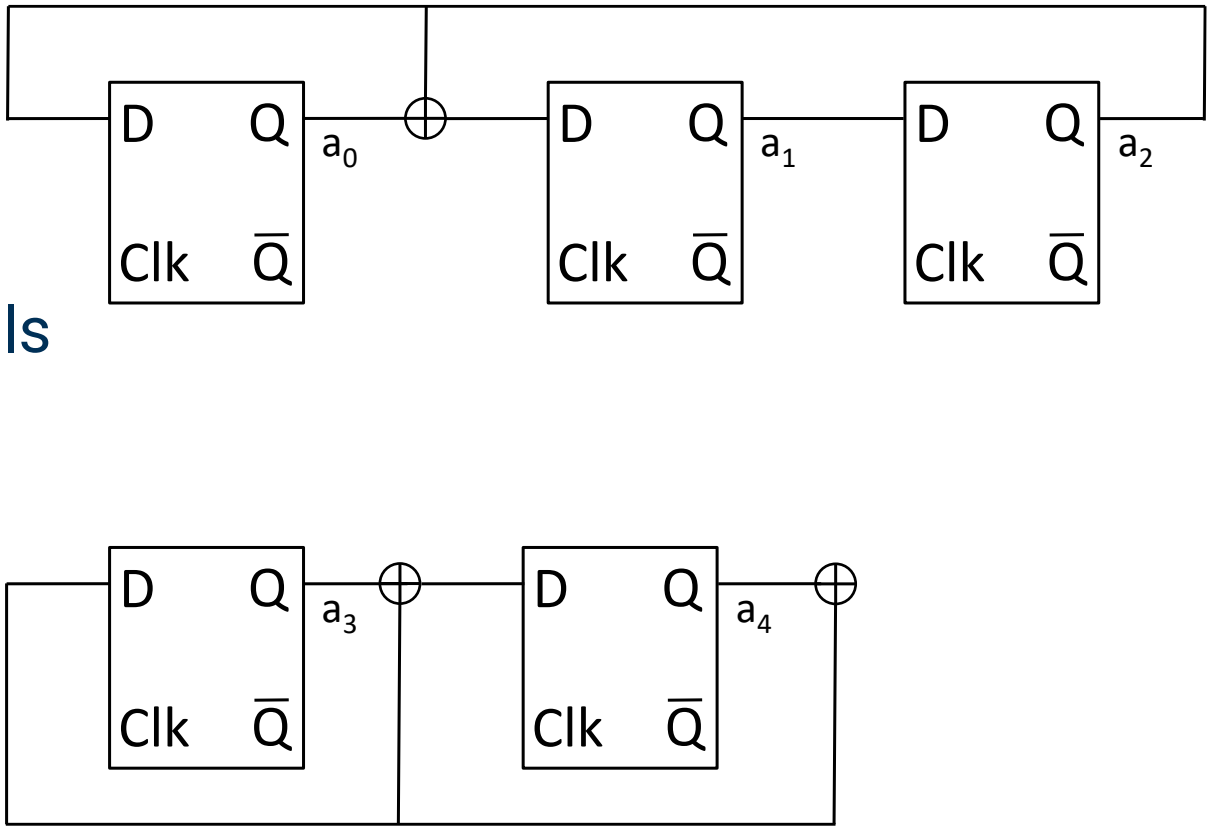
***THEOREM 1 (Chaining Period Theorem)*** *As shown in Figure 2(a), let A=(S,f) be a nonsingular system, and let B be an n-bit register with linear feedback represented by the matrix* $U \epsilon \mathbb{GL}(n,2)$*. Let the characteristic polynomial of U be irreducible, and let B have period m. Let* $C: S \rightarrow \mathbb{F}_2^n$ *be some chaining function from A to B, and let R denote the composite system formed by A and B via chaining function C. Then the cycle structure of R can be determined from the cycles of A and B as follows:*

*For any cycle of length p in A:*

1. *If* $m \mid p$ *and* $f_p = 1_{G_U}$ *then there are* $2^n$ *cycles of length p*
2. *If* $m \mid p$ *and* $f_p \neq 1_{G_U}$ *then there are* $2^{n-1}$ *cycles of length 2p*
3. *If* $m \nmid p$ *then there are* $\frac{2^n-1}{m}\gcd(m,p)$ *cycles of length* $\mathrm{lcm}(m,p)$ *and one cycle of length p*

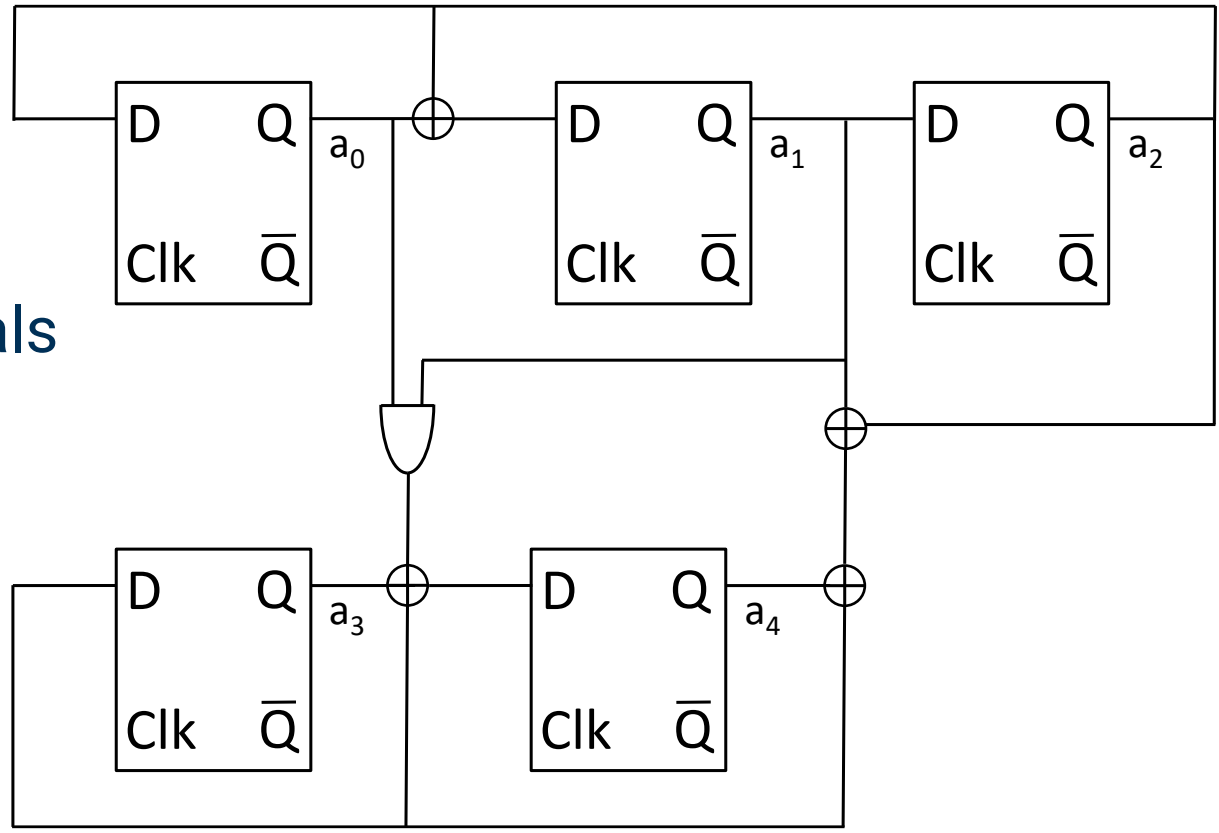
# Example 1

- Shown is a 3-bit LFSR and a 2-bit LFSR
- These two LFSRs with primitive polynomials were shown earlier
- The 2-bit LFSR is relabeled $a_3$, $a_4$

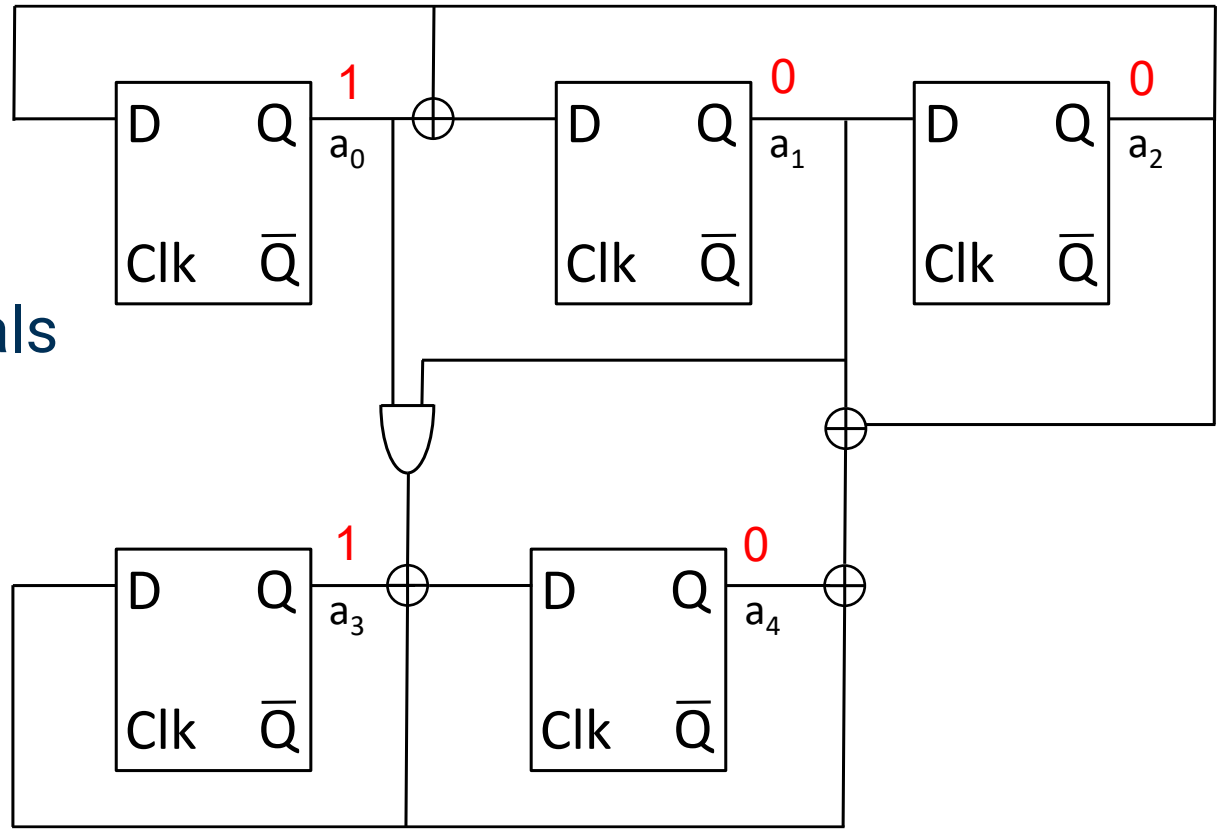


# Example 1

- Shown is a 3-bit LFSR and a 2-bit LFSR
- These two LFSRs with primitive polynomials were shown earlier
- The 2-bit LFSR is relabeled $a_4$, $a_5$
- The chaining function is $a_1$ XOR $a_2$ and $a_0$ AND $a_1$
- The next state functions for the 2-bit LFSR are now $a_4 = a_3$ XOR ($a_0$ AND $a_1$) and $a_3 = a_4$ XOR $a_1$ XOR $a_2$

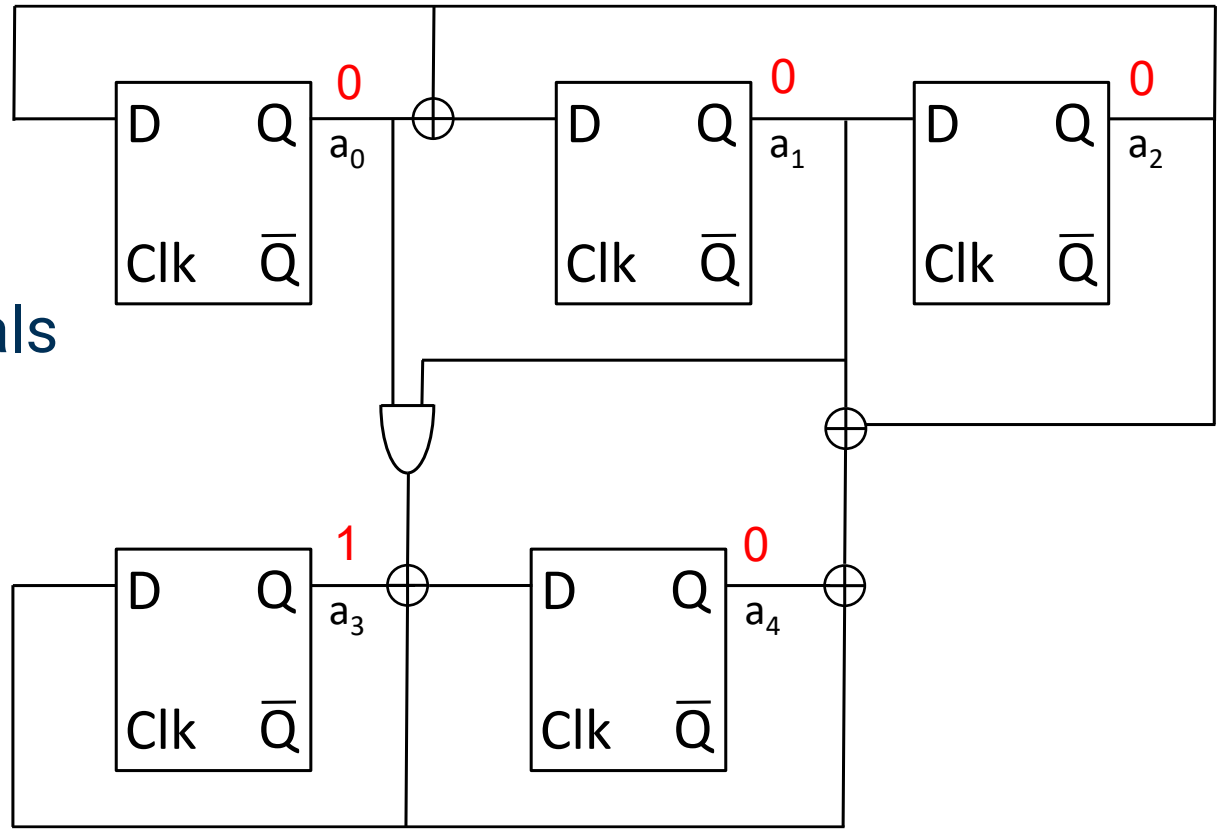


# Example 1



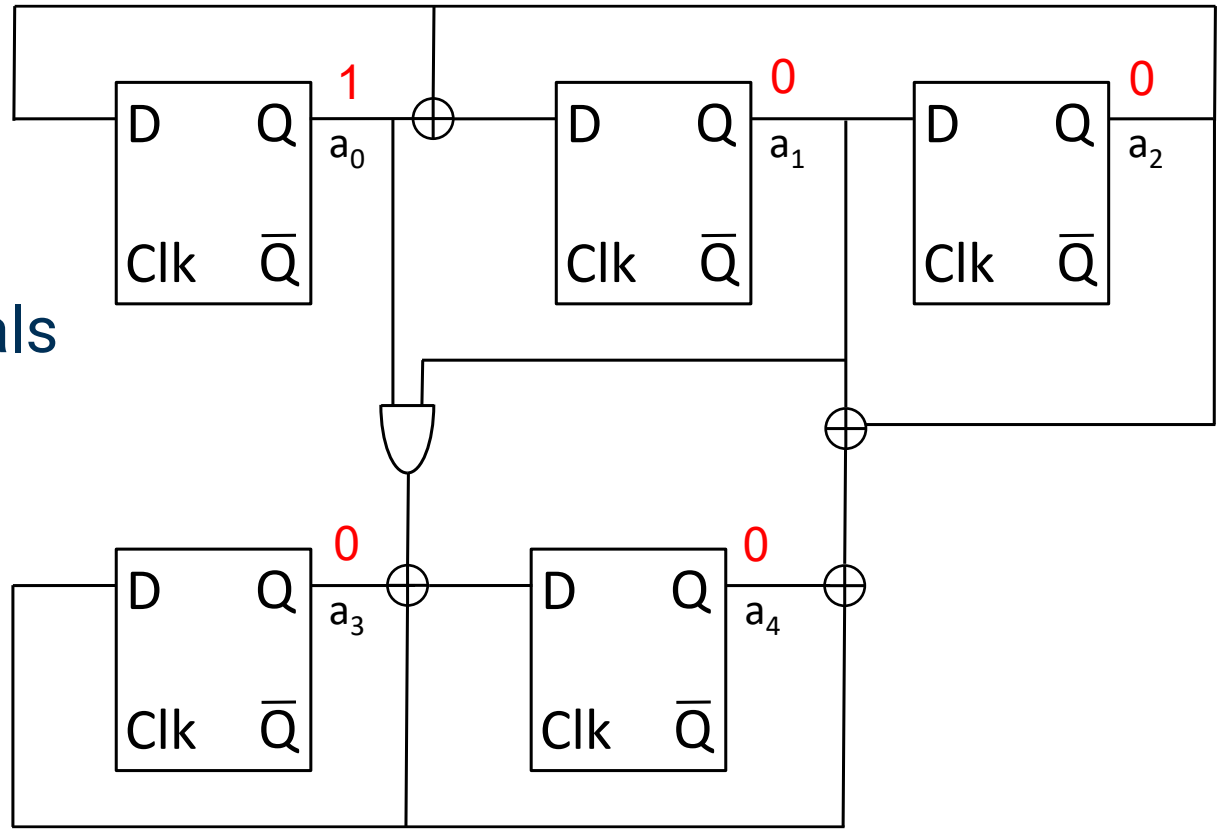

- Shown is a 3-bit LFSR and a 2-bit LFSR
- These two LFSRs with primitive polynomials were shown earlier
- The 2-bit LFSR is relabeled $a_4$, $a_5$
- The chaining function is $a_1$ XOR $a_2$ and $a_0$ AND $a_1$
- The next state functions for the 2-bit LFSR are now $a_4 = a_3$ XOR ($a_0$ AND $a_1$) and $a_3 = a_4$ XOR $a_1$ XOR $a_2$
- Consider starting state $a_2,a_1,a_0 = 001$ and $a_4,a_3 = 01$; the result is a cycle of size 21

# Example 1



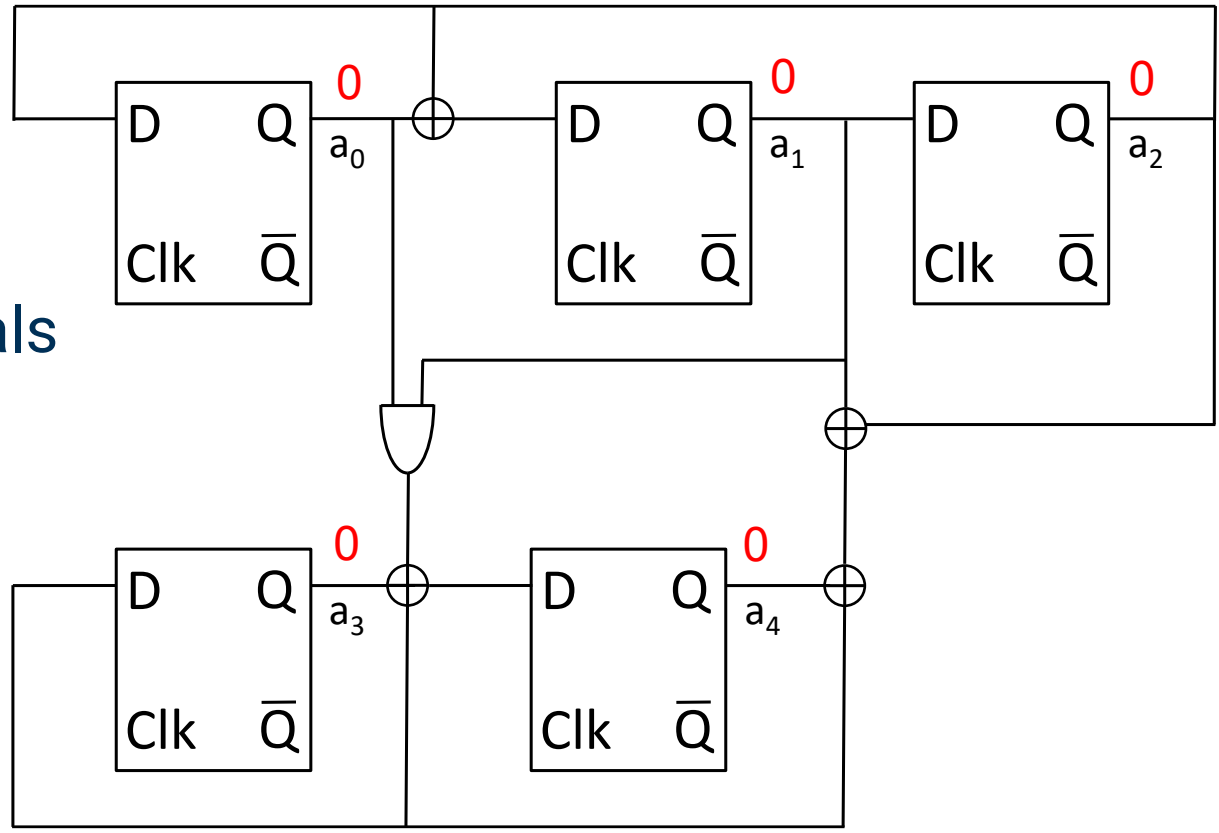

- Shown is a 3-bit LFSR and a 2-bit LFSR
- These two LFSRs with primitive polynomials were shown earlier
- The 2-bit LFSR is relabeled $a_4$, $a_5$
- The chaining function is $a_1$ XOR $a_2$ and $a_0$ AND $a_1$
- The next state functions for the 2-bit LFSR are now $a_4 = a_3$ XOR ($a_0$ AND $a_1$) and $a_3 = a_4$ XOR $a_1$ XOR $a_2$
- Consider starting state $a_2,a_1,a_0 = 001$ and $a_4,a_3 = 01$; the result is a cycle of size 21
- Consider starting state $a_2,a_1,a_0 = 000$ and $a_4,a_3 = 01$; the result is a cycle of size 3

# Example 1




- Shown is a 3-bit LFSR and a 2-bit LFSR
- These two LFSRs with primitive polynomials were shown earlier
- The 2-bit LFSR is relabeled $a_4$, $a_5$
- The chaining function is $a_1$ XOR $a_2$ and $a_0$ AND $a_1$
- The next state functions for the 2-bit LFSR are now $a_4 = a_3$ XOR ($a_0$ AND $a_1$) and $a_3 = a_4$ XOR $a_1$ XOR $a_2$
- Consider starting state $a_2,a_1,a_0$ = 001 and $a_4,a_3$ = 01; the result is a cycle of size 21
- Consider starting state $a_2,a_1,a_0$ = 000 and $a_4,a_3$ = 01; the result is a cycle of size 3
- Consider starting state $a_2,a_1,a_0$ = 001 and $a_4,a_3$ = 00; the result is a cycle of size 7

# Example 1




- Shown is a 3-bit LFSR and a 2-bit LFSR
- These two LFSRs with primitive polynomials were shown earlier
- The 2-bit LFSR is relabeled $a_4$, $a_5$
- The chaining function is $a_1$ XOR $a_2$ and $a_0$ AND $a_1$
- The next state functions for the 2-bit LFSR are now $a_4 = a_3$ XOR ($a_0$ AND $a_1$) and $a_3 = a_4$ XOR $a_1$ XOR $a_2$
- Consider starting state $a_2,a_1,a_0$ = 001 and $a_4,a_3$ = 01; the result is a cycle of size 21
- Consider starting state $a_2,a_1,a_0$ = 000 and $a_4,a_3$ = 01; the result is a cycle of size 3
- Consider starting state $a_2,a_1,a_0$ = 001 and $a_4,a_3$ = 00; the result is a cycle of size 7
- Consider starting state $a_2,a_1,a_0$ = 000 and $a_4,a_3$ = 00; result cycle size 1

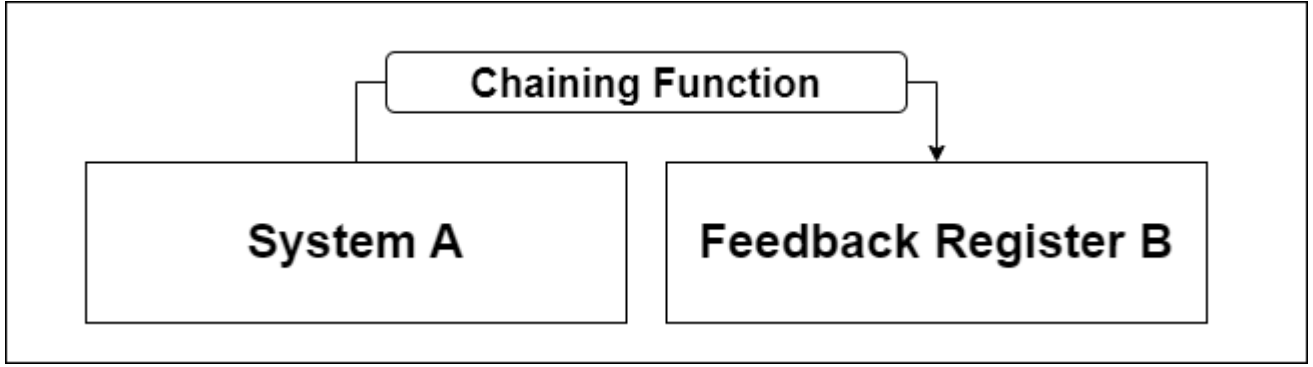



Figure 2(a) from [1]

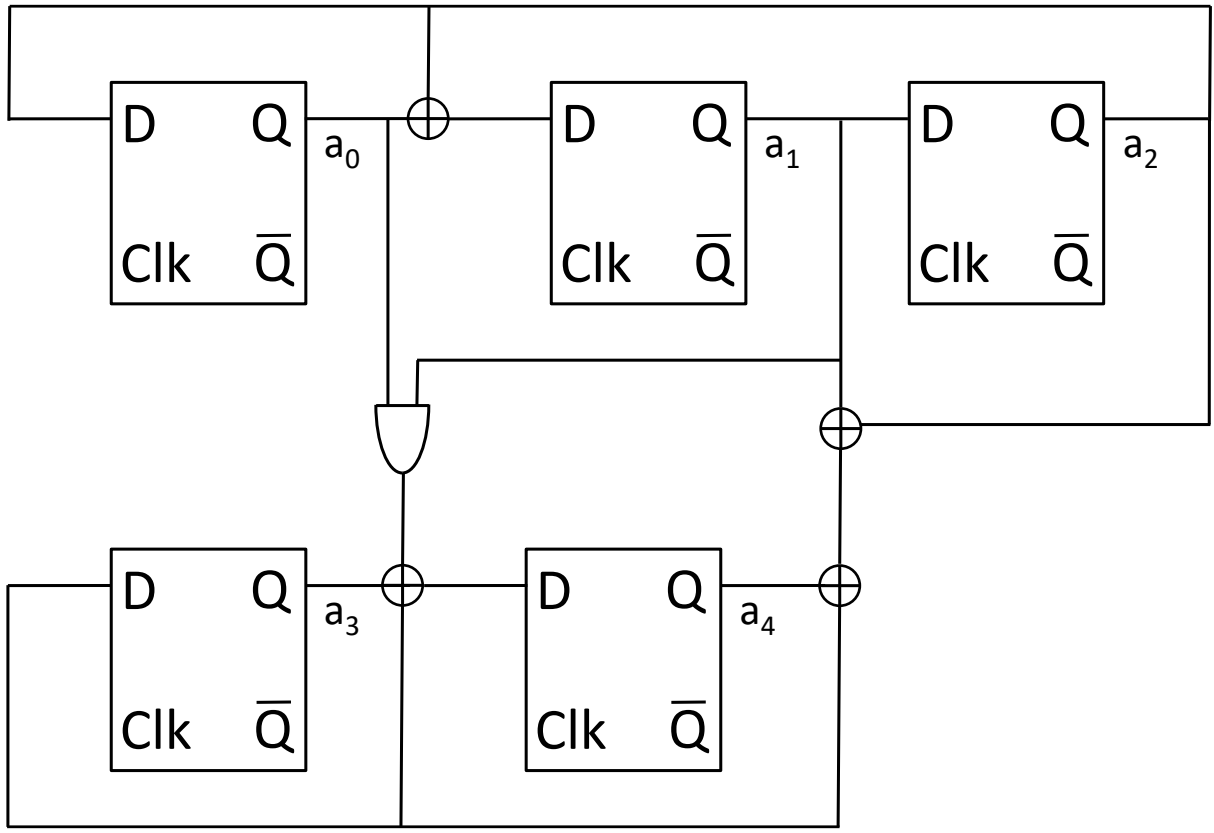
***THEOREM 1 (Chaining Period Theorem)*** *As shown in Figure 2(a), let A=(S,f) be a nonsingular system, and let B be an n-bit register with linear feedback represented by the matrix* $U \epsilon \mathbb{GL}(n,2)$*. Let the characteristic polynomial of U be irreducible, and let B have period m. Let* $C: S \to \mathbb{F}_2^n$ *be some chaining function from A to B, and let R denote the composite system formed by A and B via chaining function C. Then the cycle structure of R can be determined from the cycles of A and B as follows:*

*For any cycle of length p in A:*

1. *If* $m \mid p$ *and* $f_p = 1_{G_U}$ *then there are* $2^n$ *cycles of length p*
2. *If* $m \mid p$ *and* $f_p \neq 1_{G_U}$ *then there are* $2^{n-1}$ *cycles of length 2p*
3. *If* $m \nmid p$ *then there are* $\frac{2^n-1}{m}\gcd(m,p)$ *cycles of length* $\mathrm{lcm}(m,p)$ *and one cycle of length* $p$

# Example 1 (continued)

- The 3-bit LFSR has a cycle of size $p = 7$
- The 2-bit LFSR has a cycle of size $m = 3$
- Clearly, $m$ does not evenly divide $p$
- Therefore, according to the Chaining Period Theorem, corresponding to the cycle of size 7 in the 3-bit LFSR, the overall 5-bit register has a cycle of size 21 and a cycle of size 7




- More formally, $\mathrm{lcm}(m,p) = 21$, and $\frac{2^n-1}{m}\gcd(m,p) = \frac{2^2-1}{3}\gcd(3,7) = 1$, so for the cycle of size $p = 7$ in the 3-bit LFSR the overall 5-bit register has cycles of size $\mathrm{lcm}(m,p) = 21$ and $p = 7$
- Similarly, for the cycle of size 1 (in other words, $p = 1$) in the 3-bit LFSR, the overall 5-bit register has cycles of size $\mathrm{lcm}(m,1) = 3$ and $p = 1$

# Chaining Period Theorem Works for Nested Chaining

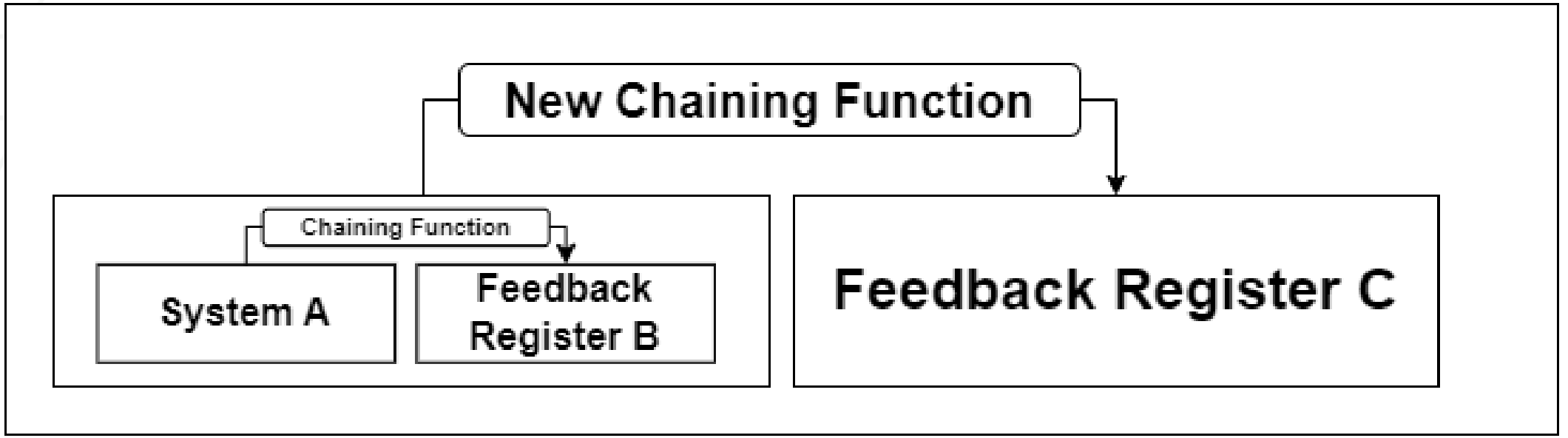



**Figure 2(b) from [1]**

# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

Georgia Tech

# Product Registers

- The Product Register (PR) [1] is a generalization of the Galois LFSR, allowing for the internal state to be permuted in a general way rather than restricting to a shift
- An $n$-bit Product Register is parametrized by:
  - A feedback polynomial $P(x)$, where $\deg\big(P(x)\big) = n$
  - An update polynomial $U(x)$, where $\deg(U(x)) \leq n - 1$
- The state of a PR updates according to:
  - $\boldsymbol{A[t+1] = U(x)A[t] \bmod P(x)}$
- From the above, it is evident that $U(x) = 0, 1$ result in degenerate behavior:
  - $U(x) = 0$ causes a PR to remain stuck in the all-zeros state
  - $U(x) = 1$ causes a PR to remain stuck in its initial state
- Thus, we omit $U(x) = 0, 1$ from the set of "valid" update polynomials for a PR
  - => An $n$-bit PR has $2^n - 2$ possible valid update polynomials
- We consider Galois LFSRs as a subset of PRs, namely the special case where $U(x) = x$

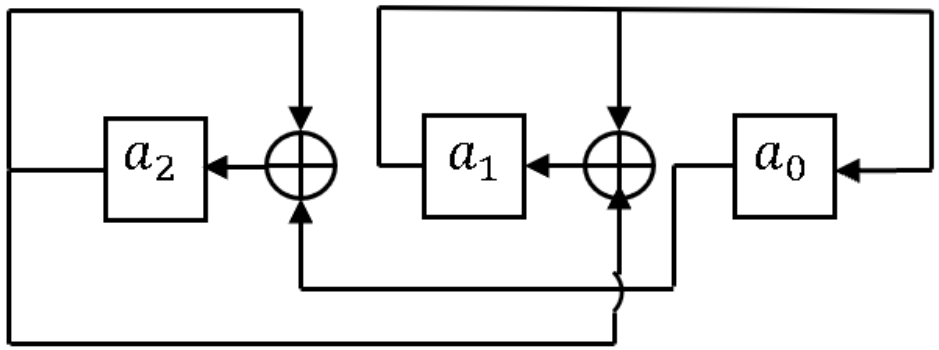
# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

Georgia Tech

# Mersenne Product Registers

- A Mersenne Product Register (MPR) is a Product Register whose size is a Mersenne exponent
  - Ex: 2-bit PR, 3-bit PR, 5-bit PR are all MPRs
  - Larger MPRs: 61-, 89-, 127-bit
  - After $n = 127$, gaps between Mersenne exponents become much larger
- From Definition 20 of [1], an $n$-bit MPR achieves period $2^n - 1$ provided $P(x)$ is primitive and $U(x) \neq 0, 1$
  - For an MPR, $\deg(P(x))$ is prime; thus, any irreducible $P(x)$ is also primitive [12]




**A 3-bit MPR with $P(x) = x^3 + x + 1$ and $U(x) = x^2$**

# Mersenne Product Registers

- The set of possible states of an MPR always remains the same
- Changing $U(x)$ alters the order in which an MPR traverses its possible states when the MPR is seeded to the same initial state
- Example from [1]:
  - 3-bit MPR
  - Fixed primitive $\mathrm{P(x)} = \mathrm{x}^3 + \mathrm{x} + 1$ => MPR has full period of $2^3 - 1 = 7$
  - Fixed initial state: $001$
  - Vary $\mathrm{U(x)}$

| Update | $t=0$ | $t=1$ | $t=2$ | $t=3$ | $t=4$ | $t=5$ | $t=6$ |
|---|---|---|---|---|---|---|---|
| $x$ | 001 | 010 | 100 | 101 | 111 | 011 | 110 |
| $x+1$ | 001 | 011 | 101 | 010 | 110 | 111 | 100 |
| $x^2$ | 001 | 100 | 111 | 110 | 010 | 101 | 011 |
| $x^2+1$ | 001 | 101 | 110 | 100 | 011 | 010 | 111 |
| $x^2+x$ | 001 | 110 | 011 | 111 | 101 | 100 | 010 |
| $x^2+x+1$ | 001 | 111 | 010 | 011 | 100 | 110 | 101 |

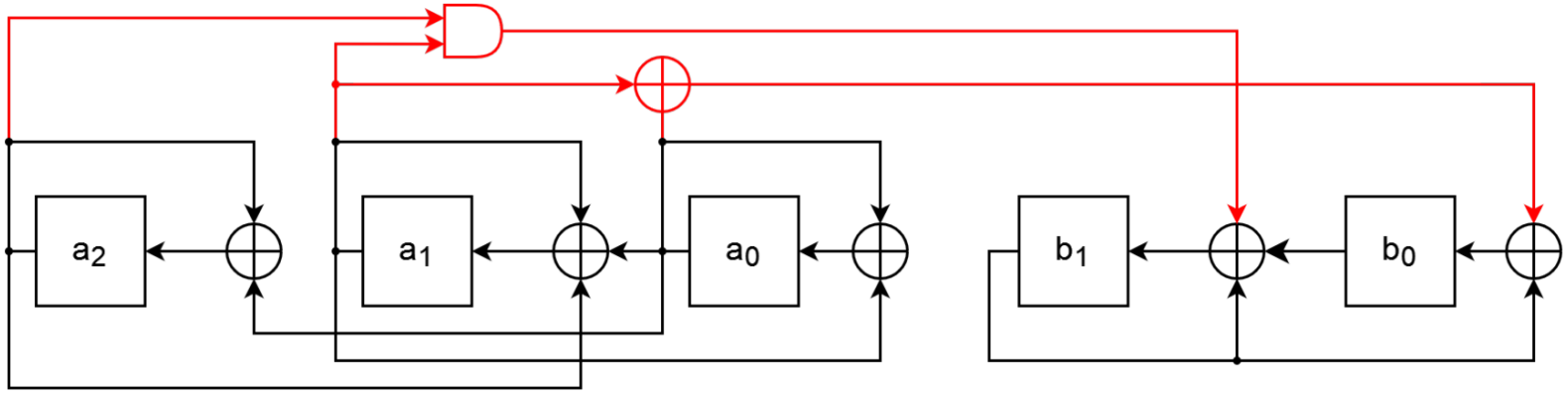
# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

Georgia Tech

# Composite Mersenne Product Registers

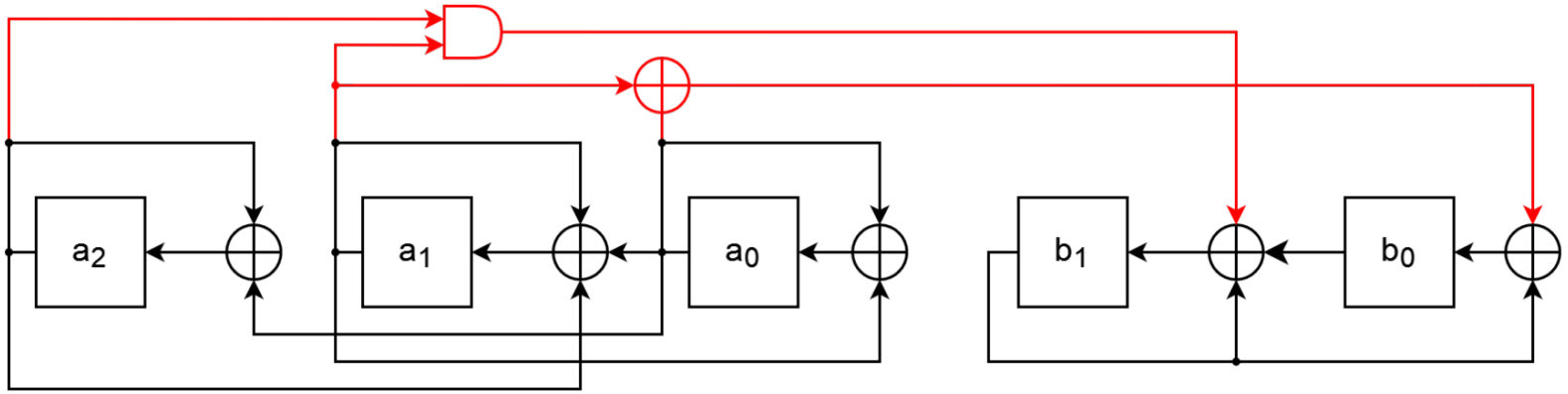
- On their own, MPRs are linear, meaning they cannot be used as a standalone cryptographic primitive
- It is possible to connect MPRs to form a larger nonlinear structure called a Composite Mersenne Product Register (CMPR)
- MPRs can be connected using **chaining functions**, or sets of Boolean functions that allow the state of an MPR to affect another MPR
  - The resulting structure is a CMPR
- CMPRs can be made nonlinear by using multiplicative terms (e.g., AND gates) in the chaining functions
- **Cryptographic Analogy:** MPRs provide a linear diffusion mechanism, whereas chaining functions provide nonlinear confusion




**A 5-bit CMPR formed by chaining a 3-bit MPR into a 2-bit MPR (Chaining function shown in red)**

# Composite Mersenne Product Registers

- In the 5-bit CMPR below, denote the state of the 3-bit MPR by $\mathrm{A}_1[\mathrm{t}]$, the state of the 2-bit MPR by $\mathrm{A}_2[\mathrm{t}]$, and the chaining function by $\mathrm{C}$
- Similarly, denote the update polynomial and primitive polynomial of the 2-bit MPR by $\mathrm{U}_2(\mathrm{x})$ and $\mathrm{P}_2(\mathrm{x})$, respectively
- Then, the state update of the 2-bit MPR becomes:
  - $\boldsymbol{A_2[t+1] = [U_2(x)A_2[t] \oplus C] \bmod P_2(x)}$
- In the example below, $C = a_2a_1 \oplus a_1 \oplus a_0$




**A 5-bit CMPR formed by chaining a 3-bit MPR into a 2-bit MPR (Chaining function shown in red)**

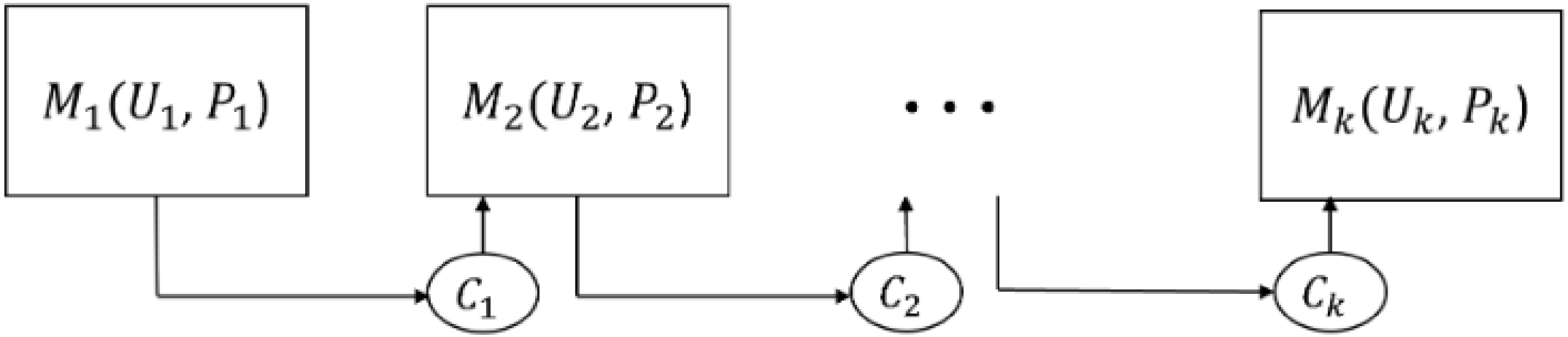
# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

Georgia Tech

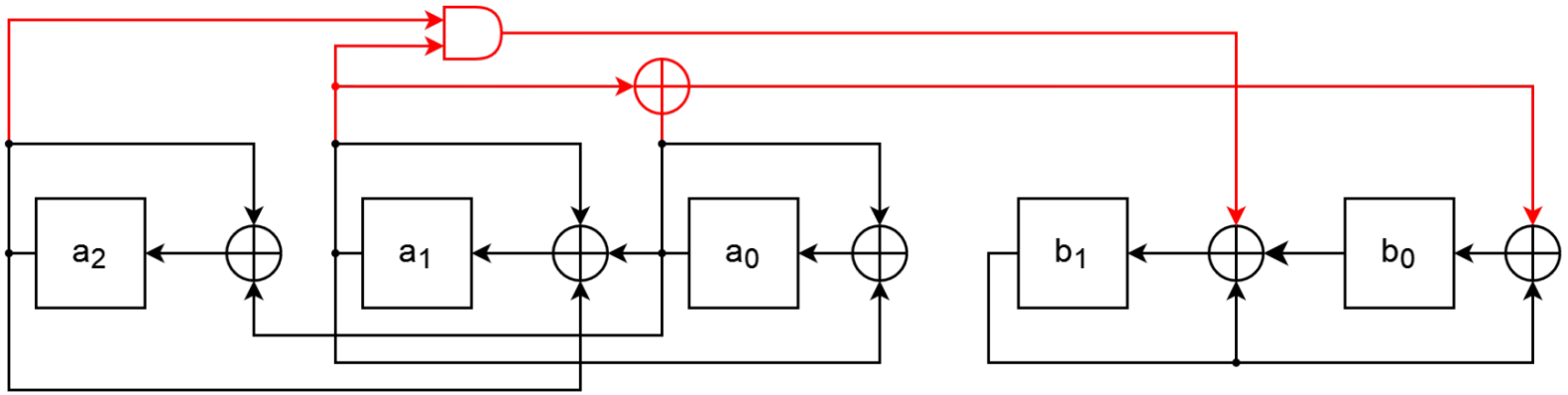
***THEOREM 2*** *Let R be a CMPR composed of k MPRs, denoted* $\{M_1, M_2, \ldots, M_k\}$*, with sizes* $S = \{m_1, m_2, \ldots, m_k\}$ *where each* $m_i$ *is a unique Mersenne exponent. Then R has a unique cycle of states for each subset* $S = \subseteq \{m_1, m_2, \ldots, m_k\}$*, with that cycle having a length equal to* $\prod_{m_i \epsilon S}(2^{m_i} - 1)$.

Proof by induction available in [1].



# CMPR Example for Theorem 2

- In the 5-bit CMPR below, the 3-bit MPR can be considered $M_1$ in Theorem 2 with size $m_1 = 3$
- Similarly, the 2-bit MPR can be considered $M_2$ in Theorem 2 with size $m_2 = 2$
- Then, the 5-bit CMPR has a cycle of size 21
  - $\prod_{m_i \epsilon S}(2^{m_i} - 1) = (2^3 - 1)(2^2 - 1) = 21$




**A 5-bit CMPR formed by chaining a 3-bit MPR into a 2-bit MPR (Chaining function shown in red)**

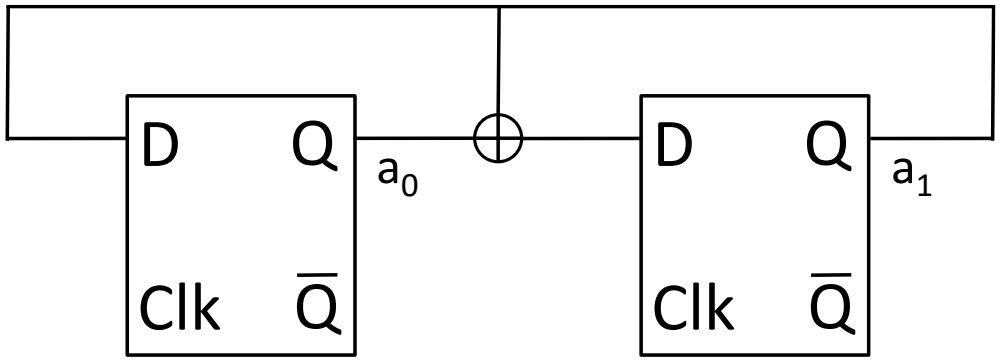




Georgia Tech

**DEFINITION 23** The Expected Period Ratio (EPR) of a CMPR of size *n* is given by $\frac{\mathbb{E}(X)}{2^n}$, where $\mathbb{E}(X)$ is the expected value of the period when the CMPR is initialized to a random state.

**DEFINITION 23** The Expected Period Ratio (EPR) of a CMPR of size *n* is given by $\frac{\mathbb{E}(X)}{2^n}$, where $\mathbb{E}(X)$ is the expected value of the period when the CMPR is initialized to a random state.




$$EPR = \frac{\mathbb{E}(X)}{2^n} = \frac{1}{2^2}\left(\frac{3}{4}3 + \frac{1}{4}\right) = \frac{10}{16} = 0.625$$

***THEOREM 3*** *Let C be a CMPR composed of k MPRs, denoted* $\{M_1, M_2, \dots, M_k\}$*, with sizes* $S = \{m_1, m_2, \dots, m_k\}$ *where each* $m_i$ *is a unique Mersenne exponent. Let n be the total number of bits in register C. The total number of states in C is then* $2^n = \prod_{m_i \epsilon S} 2^{m_i}$*. Let the initial state of C be chosen uniformly at random, and let the random variable* $X$ *denote the length of the cycle before this initial state repeats (the period of the register, started from this state). Then,*

$$\frac{\mathbb{E}(X)}{2^n} = \prod_{m_i \epsilon S} \left(1 - \frac{\left(2^{(m_i+1)} - 2\right)}{2^{2m_i}}\right)$$

Proof by induction available in [1].

# Some Examples of Theorem 3

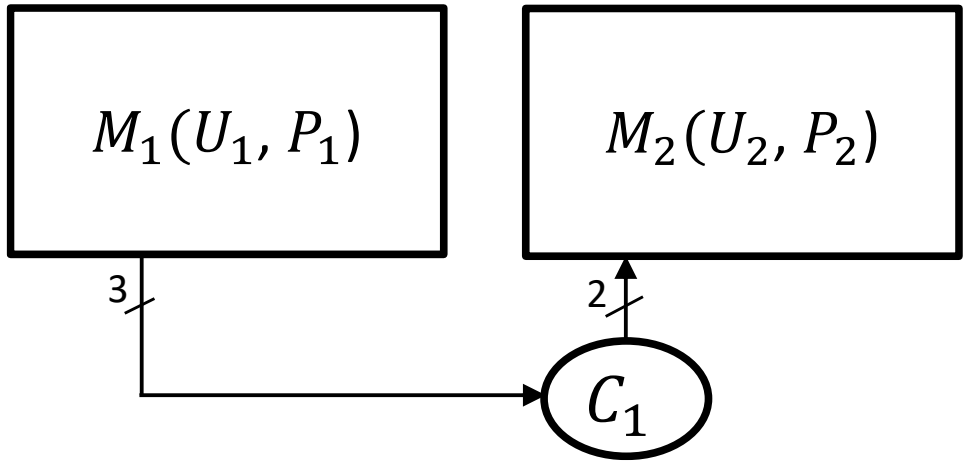
- MPR $M_1$ with size $m_1$

$$M_1(U_1, P_1)$$

$$\mathbb{E}(X) = \left(1 - \frac{\left(2^{(m_i+1)} - 2\right)}{2^{2m_i}}\right) = \frac{2^{2m_i} - 2^{(m_i+1)} + 2}{2^{2m_i}} = \frac{(2^{m_i} - 1)^2 + 1}{2^{2m_i}}$$

For $m_1 = 3$, $\frac{(2^{m_i}-1)^2+1}{2^{2m_i}} = \frac{(2^3-1)^2+1}{2^6} = \frac{50}{64} = 0.78125$

# Some Examples of Theorem 3 (continued)



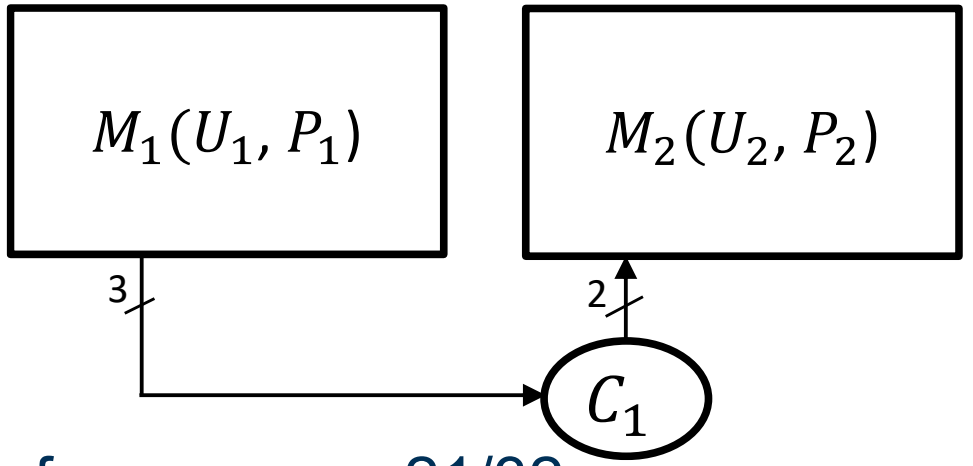

- CMPR: $M_1$ with size $m_1$=3 and $M_2$ with size $m_2$=2

For $m_2$ = 2, $\frac{(2^{m_i}-1)^2+1}{2^{2m_i}} = \frac{(2^2-1)^2+1}{2^4} = \frac{10}{16} = 0.625$

For $m_1$ = 3, $\frac{(2^{m_i}-1)^2+1}{2^{2m_i}} = \frac{(2^3-1)^2+1}{2^6} = \frac{50}{64} = 0.78125$

For this CMPR, $\mathrm{EPR} = (0.625 * 0.78125) = 0.48828125$

# Some Examples of Theorem 3 (continued)




- CMPR: $M_1$ with size $m_1$=3 and $M_2$ with size $m_2$=2

This CMPR has a cycle of size 21 with probably of occurrence 21/32
A second cycle of size 7 occurs with probably 7/32
A third cycle of size 3 occurs with probability 3/32
Finally, a cycle of size 1 exist with probability 1/32

$$\mathbb{E}(X) = 21\frac{21}{32} + 7\frac{7}{32} + 3\frac{3}{32} + \frac{1}{32} = \frac{500}{32}$$

$$\Rightarrow \frac{\mathbb{E}(X)}{32} = \frac{500}{1024} = 0.48828125$$

# Expected Period and Ratio Bounds

## Table 3: Lower EPR Bound

| Smallest MPR Size | Approximate EPR Lower Bound |
|---|---|
| 2 | 0.4514 |
| 3 | 0.7223 |
| 5 | 0.9246 |
| 7 | 0.9842 |
| 13 | 0.9997 |

## Table 4: Example Constructions

| CMPR Size | Mersenne Exponents | Expected Period | Expected Period Ratio (EPR) |
|---|---|---|---|
| 32 | 19, 13 | $4.2939 \times 10^{9}$ | 0.99975 |
| 64 | 61, 3 | $1.4412 \times 10^{19}$ | 0.78125 |
| 128 | 61, 31, 19, 17 | $3.4028 \times 10^{38}$ | 0.99998 |
| 256 | 127, 61, 31, 17, 13, 7 | $1.1397 \times 10^{77}$ | 0.98424 |

# Composite Mersenne Product Registers

- For a CMPR formed by MPRs $\{M_1, \ldots, M_k\}$, the size of the CMPR is given by $\sum_{i=1}^{k} \text{len}(M_i)$
- CMPRs also have a mathematically-proven **expected period** [1], which is the number of unique states the CMPR will, on average, cycle through when initialized to a uniformly-chosen initial state
- In this presentation, we assume the following conventions are followed when constructing a CMPR:
  - Each Mersenne exponent is used once
  - Chaining functions connect from larger MPRs to smaller MPRs
  - Feedback polynomials for all MPRs are irreducible (primitive) and $U(x) \neq 0, 1$
- Then, from Theorem 3 of [1], an $n$-bit CMPR has an expected period $\geq \mathbf{0.45 * 2^n \approx 2^{n-1}}$
- More generally, the expected period of an $n$-bit CMPR constructed from a set of Mersenne exponents $S$ is given by:
  - $\mathbf{[\sum_{m_i \in S}(1 - \frac{2^{m_i+1}-2}{2^{2m_i}})] * 2^n}$ **[1]**

- The expected period is independent of the chaining functions used to construct a CMPR
  - Nonetheless, it is advantageous to use **balanced Boolean equations** as chaining functions to ensure that the CMPR state evolution is not biased
  - In other words, the truth tables for each Boolean equation used as a chaining function should consist of equal numbers of 0's and 1's

***THEOREM 4*** *Let C(A[t]) be any periodic sequence of n-bit vectors in $\mathbb{F}_2^n$, and let its Z-transform be denoted C(z). Let B be an n-bit linear feedback register, with n prime and its characteristic polynomial primitive, and which updates according to Equation 1, with C(A[t]) as input on each cycle. Then the Z-transform of B[t] is that*

$$B(z) = (zI \oplus U)^{-1}(C(z) \oplus zB[0]) = \frac{1}{\chi_u(z)}\mathrm{Adj}(zI \oplus U)(C(z) \oplus zB[0]))$$

*where $\chi_u(z)$ is the characteristic polynomial of $U$, the matrix by which $B$ updates.*

# Theorem 4 Provides the Following

- Analysis of linear complexity of CMPRs using root expressions
- The ability to bound the linear complexity of CMPRs without the need to generate a long sequence
  - Instead, the $B(z)$ representation of Theorem 4 can be used to analyze how nonlinear expressions propagate through a CMPR
- Enables block-by-block propagation of root expressions, allowing CMPRs to be evaluated compositionally rather than through brute-force sequence generation

# Outline

- Introduction
- Research Overview
- Lightweight Cryptography
- Background
  - Terminology
  - Feedback Registers
  - Finite Field Theory
- Chaining Period Theorem
- Product Registers
  - Mersenne Product Registers
  - Composite Mersenne Product Registers (CMPRs)
  - CMPR Theorems
- References

Georgia Tech

# References


[1] D. Gordon, A. Allahverdi, S. Abrelat, A. Hemingway, A. Farooq, I. Smith, N. Arora, A. Chang, Y. Qiang, and V. Mooney, “Scalable nonlinear sequence generation using Composite Mersenne Product Registers,” *IACR Commun. Cryptol.*, vol. 1, no. 4, Jan. 2025, doi: 10.62056/a3tx11zn4.

[2] A. Allahverdi and V. Mooney, “A hardware-efficient AEAD stream Cipher based on a hybrid nonlinear feedback register structure,” *2025 IEEE International Conference on Cyber Security and Resilience (CSR),* Chania, Crete, Greece, 2025, pp. 1016-1023, doi: 10.1109/CSR64739.2025.11130096.

[3] I. T. L. Computer Security Division, “Lightweight Cryptography | CSRC,” *CSRC | NIST*, Jan. 03, 2017. https://csrc.nist.gov/projects/lightweight-cryptography

[4] C. Dobraunig, M. Eichlseder, F. Mendel, and M. Schläffer, “Submission to NIST,” 2019. Available: https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/round-2/spec-doc-rnd2/ascon-spec-round2.pdf

[5] N. Mouha, B. Mennink, A. Van Herrewege, D. Watanabe, B. Preneel, and I. Verbauwhede, “Chaskey: An Efficient MAC Algorithm for 32-bit Microcontrollers.” Available: https://eprint.iacr.org/2014/386.pdf

[6] M. Hell *et al.*, “Grain-128AEADv2 -A lightweight AEAD stream cipher Cover sheet Corresponding submitter: Backup point of contact.” Available: https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/finalist-round/updated-spec-doc/grain-128aead-spec-final.pdf

[7] E. Dubrova, “A list of maximum period NLFSRs,” *IACR Cryptology ePrint Archive*, 2012. [Online]. Available: https://eprint.iacr.org/2012/166.pdf

[8] E. Dubrova, “A scalable method for constructing Galois NLFSRs with period $2^n - 1$ using cross-join pairs,” *IEEE Transactions on Information Theory*, vol. 1, no. 59, pp. 703–709, 2013

[9] C. De Canniere, “Trivium: A stream cipher construction inspired by block cipher design principles,” in *Lecture Notes in Computer Science*, vol. 4377, A. Biryukov, Ed. Berlin, Germany: Springer, 2006, pp. 171–186, doi: 10.1007/11836810 13.

[10] E. Dubrova and M. Hell, “Espresso: A stream cipher for 5G wireless communication systems,” *Cryptogr. Commun.*, vol. 9, no. 2, pp. 273–289, Dec. 2015, doi: 10.1007/s12095-015-0173-2

# References

[11] Aumasson, JP., Henzen, L., Meier, W., Naya-Plasencia, M. (2010). QUARK: A Lightweight Hash. In: Mangard, S., Standaert, FX. (eds) Cryptographic Hardware and Embedded Systems, CHES 2010. CHES 2010. Lecture Notes in Computer Science, vol 6225. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-15031-9_1.

[12] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, Public-Key Parameters. Boca Raton, FL: *CRC Press*, 1997, pp. 154–160. [Online]. Available: https://doi.org/10.1201/9780429466335.

[13] Roman Kuc, *Introduction to Digital Signal Processing*. McGraw-Hill, New York, NY, 1988. ISBN: 0-07-035570-3.

[14] Joseph J. Rotman, *An Introduction to the Theory of Groups*. Springer-Verlag, New York, NY, 1995. ISBN: 0-387-94285-8.