Quantum Leap:
An Exploration of the Development of Computing and Cryptography

Jake Lyon

The College of Wooster

Advised by Dr. Dan Palmer

2026

Did you use a password today?

Odds are — yes.

Caesar Cipher

~50 BC
One of the first
encryption methods

Enigma Machine

WWII
15 trillion possible
combinations

RSA Today

1977–present
2048-bit+ keys
Internet standard

Three Phases of Computation & Cryptography

01

Circuitry

Shift Cipher

Physical digital logic using ICs, breadboards, and binary signals

02

Arduino

RSA Encryption

Hardware + software interaction on a microcontroller

03

Quantum

Breaking RSA

Shor's algorithm and period-finding with quantum resources

Shift Cipher in Hardware

How it works

1 **Input**

User types a number (0–25) on a 4×4 keypad

Encode

EEPROM lookup converts to 5-bit binary representation

Encrypt

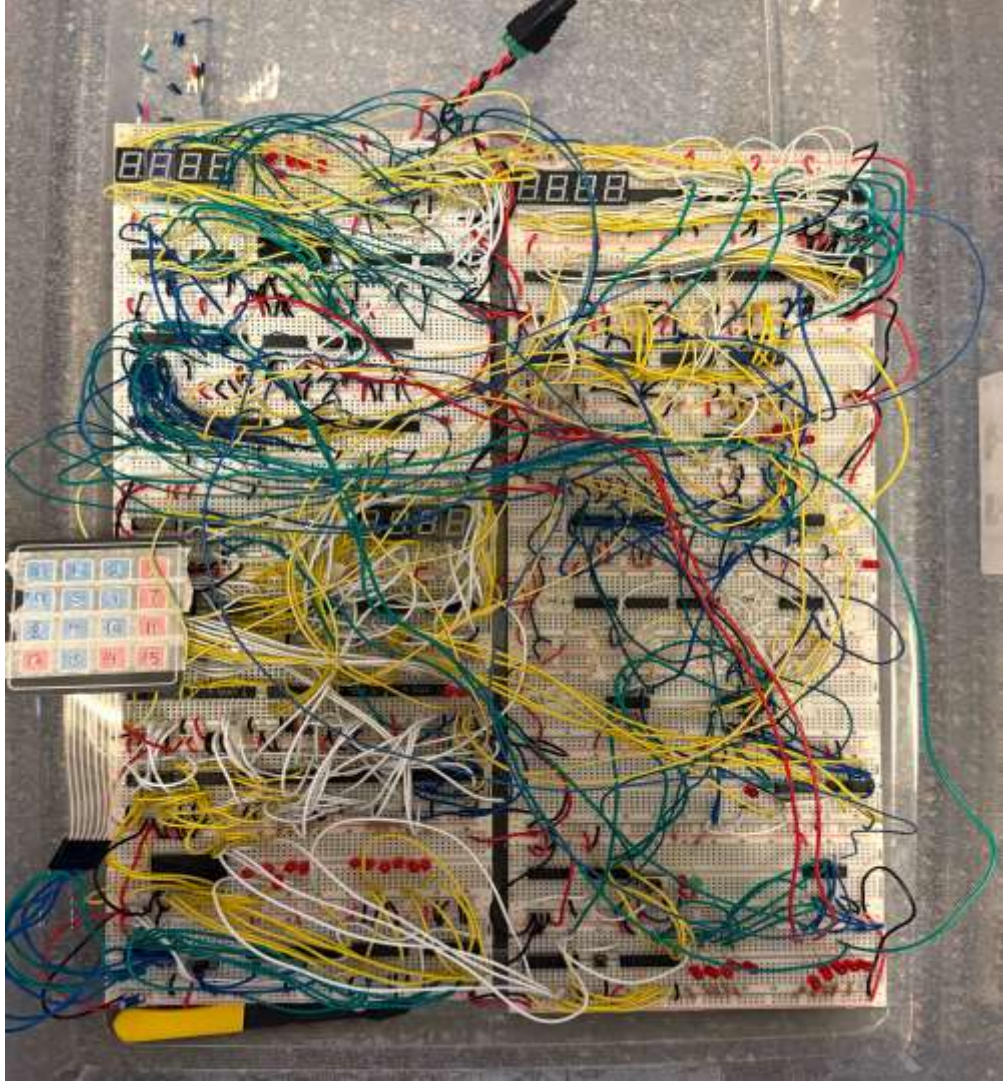
Binary adders compute $C = (P + K) \bmod 26$

4 **Output**

Seven-segment displays show original → encrypted → decrypted

Key Components

- 15+ Integrated Circuits (ICs)
- SN74LS83 4-bit binary adders
- LS374 Octal flip-flops (storage)
- EEPROMs as lookup tables
- 4×4 keypad for input
- Seven-segment displays
- Two's complement for decryption



RSA Encryption on a Microcontroller

Change in Strength 25 possible shift-cipher keys → 2,147,483,648 possible RSA keys (32-bit implementation)

RSA Key Generation

Choose primes:	`p = 283,  q = 307`
Compute n & ϕ(n):	`n = 86,881  ϕ(n) = 86,292`
Select e:	`e = 7  (coprime with ϕ(n))`
Compute d:	`d = 24,655  (modular inverse)`
Keys:	`Public (7, 86881) · Private (24655, 86881)`

Arduino UNO R3

- ATMega328p microcontroller
- 32 KB flash + 2 KB SRAM
- 64-bit unsigned integers for RSA
- Modular exponentiation by squaring
- LCD displays encryption in real-time
- 4-character block processing

Breaking RSA with Shor's Algorithm

Shor's Algorithm — The Key Steps

Classical	Choose random r < N, coprime to N
Quantum	Find period p of $f(x) = r^x \bmod N$
Classical	Compute gcd(r^(p/2)±1, N)
Classical	Extract prime factors p and q
Classical	Calculate ϕ(n), then private key d

Live Experiment

n = 203 (p=29, q=7)

Platform: IBM Quantum

Max factorable: ~7–8 bit

Encrypted message:

```
172085...197085
```

Recovered message:

```
"iamhanding
jake feedback
iffinecanbefini"
```

What Comes After RSA?

Code-Based

McEliece (1978)

Based on error-correcting codes. Decoding a linear code without the private key is NP-hard.

Lattice-Based

CRYSTALS-Kyber, Dilithium

Finding the shortest vector in a high-dimensional lattice is hard even for quantum computers.

Multivariate

MQ-based systems

Solving systems of multivariate quadratic equations is NP-hard in general.

2022 NIST Post-Quantum Standards

CRYSTALS-Kyber	**CRYSTALS-Dilithium**	**Falcon**	**SPHINCS+**

CONCLUSION

Circuit → Shift Cipher

Arduino → RSA

Quantum → Post-Quantum

Key Takeaways

- Computing and encryption are always evolving
- RSA's security rests on factorization hardness, which quantum computers will eventually break.
- Shor's algorithm demonstrates the pipeline even at small scale
- Post-quantum algorithms (lattice-based) are being standardized now
- Innovation in security is always happening; the next standard is already being built.

Acknowledgments

Thank you to:

My friends and family

My advisor Dr. Palmer

The Henry J. Copeland Fund

And of course, my amazing Fiancé Ryann!