Primary parameters of hardware implementations for secret-key block ciphers

Latency

Encryption/decryption

Time to encrypt/decrypt a single block of data

Encryption/decryption

Number of bits encrypted/decrypted in a unit of time

Throughput

Block_size · Number_of_blocks_processed_simultaneously

Latency

Throughput =

Dependence of the encryption time on latency and throughput

Message size

Latency

\( \frac{\text{Message size} - \text{Block size}}{\text{Throughput}} \)

Time

Encryption time
Primary factor in choosing the encryption/decryption unit architecture

Symmetric-key cipher mode of operation:

1. Non-feedback cipher modes
   ECB, counter mode

2. Feedback cipher modes
   CBC, CFB, OFB

Non-feedback Counter Mode - CTR

\[
C_i = M_i \oplus \text{AES}(IV+i) \quad \text{for} \quad i=0..N
\]

Feedback cipher modes - CBC

\[
\begin{align*}
& C_i = \text{AES}(M_i \oplus IV) \\
& C_i = \text{AES}(M_i \oplus C_{i-1}) \quad \text{for} \quad i=2..N
\end{align*}
\]
Feedback cipher modes
CBC, CFB, OFB

Typical Flow Diagram of a Secret-Key Block Cipher

Basic iterative architecture
### Basic architecture: Timing

<table>
<thead>
<tr>
<th>CLK</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUT</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#rounds · clock_period

### Architectures suitable for feedback modes

Architecture with feedback modes:

- **Register:**
- **One round:**
- **Combinational logic:**
  - Round 1
  - Round 2
  - Round K

### Partial Loop Unrolling

- **K rounds**
- **Combinational logic:**
  - Round 1
  - Round 2
  - Round K

- **Multiplexer**
- **Register**
**Loop Unrolling: Timing**

<table>
<thead>
<tr>
<th>CLK</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN</td>
<td>C1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUT</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#rounds/k - extended_clock_period

**Loop Unrolling: Speed vs. Area**

speed = speed_{basic} \frac{1 + \tau}{1 + \tau / k}

\tau << 1

**Decreasing area by resource sharing**

Before:

- D0
- D1
- F
- D0'
- D1'

After:

- D0
- D1
- multiplexer
- F
- D0'
- D1'
- register
- register
Throughput

Resource sharing: Speed vs. Area

- basic architecture
- resource sharing

Non-Feedback Cipher Modes
ECB, counter

Comparison for non-feedback cipher modes, e.g.
Counter Mode - CTR

\[ C_i = M_i \oplus AES(IV+i) \] for \( i = 0..N \)
Traditional methodology

$k$-stage Outer-Round Pipelining

WPI: Partial Pipelining, PP-$k$. NSA: Pipelined Architecture

UC Berkeley: Unrolled Pipeline

Outer-Round Pipelining: Timing

<table>
<thead>
<tr>
<th>CLK</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

# rounds - clock_period
Outer-Round Pipelining: Speed vs. Area

- **Outer-round pipelining**
- **Non-feedback modes**
- Basic architecture
- **Area**
- **Speed**

Full outer-round pipelining

- **#rounds registers**
- Round 1
  - = one pipeline stage
- Round 2
  - = one pipeline stage
- Round #rounds
  - = one pipeline stage

Total # of pipeline stages = #rounds

New methodology

- a)
  - Register
  - One round, no pipelining
  - One round, no pipelining

- b)
  - Register
  - One round, 4 pipeline stages

- c)
  - Register
  - One round, 4 pipeline stages

- d)
  - Register
  - One round, 4 pipeline stages
  - Round 1
    - = 4 pipeline stages
  - Round 2
    - = 4 pipeline stages
  - Round #rounds
    - = 4 pipeline stages

- c)
  - Register
  - One round, 4 pipeline stages

- d)
  - Register
  - One round, 4 pipeline stages
  - Round 1
    - = 4 pipeline stages
  - Round 2
    - = 4 pipeline stages
  - Round #rounds
    - = 4 pipeline stages
Inner-Round Pipelining

one round

Inner-Round Pipelining: Timing

CLK

IN

OUT

# rounds \cdot (k \cdot \text{reduced}\_\text{clock}\_\text{period})

Inner-Round Pipelining: Speed vs. Area

speed

area

inner-round pipelining non-feedback modes

basic architecture

inner-round pipelining feedback modes

k=5

k=4

k=3

k=2

k=2

k=3

k=4

k=5
Comparison of the traditional and new design methodologies

Throughput

- inner-round pipelining
- mixed inner and outer-round pipelining
- basic architecture
- outer-round pipelining

Area

Latency vs. area dependence for the new design methodology

Latency

- inner-round pipelining
- mixed inner and outer-round pipelining
- basic architecture
- outer-round pipelining

Area

Full mixed inner- and outer-round pipelining

- $k$ registers
- round 1 = $k$ pipeline stages
- round 2 = $k$ pipeline stages
- round $#rounds$ = $k$ pipeline stages

Total # of pipeline stages = $#rounds \times k$
Critical path: Time

<table>
<thead>
<tr>
<th>Cipher</th>
<th>Time [ns]</th>
<th>0</th>
<th>20</th>
<th>40</th>
<th>60</th>
<th>80</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serpent</td>
<td>regular round</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>implementation round</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rijndael</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Twofish</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RC6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mars</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Inner-Round Pipelining

<table>
<thead>
<tr>
<th>Cipher</th>
<th>Time [ns]</th>
<th>0</th>
<th>20</th>
<th>40</th>
<th>60</th>
<th>80</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serpent</td>
<td>regular round</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>implementation round</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rijndael</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Twofish</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RC6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mars</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Full Mixed Inner and Outer-Round Pipelining

<table>
<thead>
<tr>
<th>Cipher 1</th>
<th>Cipher 2</th>
<th>Time [ns]</th>
<th>0</th>
<th>20</th>
<th>40</th>
<th>60</th>
<th>80</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Speed = \frac{128 \text{ bits}}{\text{target_clock_period}}
Conclusions for non-feedback cipher modes
ECB, counter

- All ciphers can achieve approximately the same speed.
  Area should be the primary criteria of comparison.
- Architecture with inner round pipelining combined with full outer round pipelining is the fastest

Performance of alternative architectures:
in non-feedback cipher modes (ECB, counter)

Performance of alternative architectures:
in feedback cipher modes (CBC, CFB, OFB)
Encryption in Communication Protocols

Modes of operation: CBC

RFC 2405

CBC: Implementation Issues: Encryption
Parallel processing of data (1)

Sequential processing of data

Parallel processing of different security associations

Parallel processing of data (2)

Parallel processing of packets belonging to the same security association

Encryption in CBC: multiple IVs required for the same SA
Decryption in CBC: no problems

Parallel processing of data (3)

Parallel processing of blocks belonging to the same packet

Encryption in CBC: not feasible
Decryption in CBC: no problems
Secret-key ciphers

Interface