1. Preliminaries

\[ X = m_i \cdot q_i + x_i, \quad 0 \leq x_i \leq (m_i - 1) \]

\[ x_i = X \text{ modulo } m_i = X \mod m_i = |X|_{m_i} \]

<table>
<thead>
<tr>
<th>[X]</th>
<th>[x_2]</th>
<th>[x_1]</th>
</tr>
</thead>
<tbody>
<tr>
<td>-4</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>-3</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>-2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>-1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>-1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>-3</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

**TABLE 11.1** The representation of numbers in the \((m_2, m_1) = (3, 2)\) residue system.
For example, $X = 5$

\[ x_2 = \left\lfloor \frac{5}{3} \right\rfloor = 2, \text{ since } 5 = 3 \cdot 1 + 2, \]

\[ x_1 = \left\lfloor \frac{5}{2} \right\rfloor = 1, \text{ since } 5 = 2 \cdot 2 + 1. \]

The number of elements in the useful range is

\[ M = \operatorname{l.c.m}(m_1, m_2, \ldots, m_N) \]

\[ M = \prod_{i=1}^{N} m_i \]

if moduli are pairwise relatively prime.

Drawbacks:
- Comparison between two numbers is not simple, since the residue system is not weighted.
- The sign of a number is not apparent from its residue representation.
2. ARITHMETIC OPERATIONS

Addition:
\[ |X + Y|_m = |X|_m + |Y|_m = |x_i + y_i|_m \]

In general,
\[ \sum_{j=1}^{k} X_{j} = \sum_{j=1}^{k} |X_{j}|_m = |x_{i} + y_{i}|_m \]

Multiplication:
\[ |XY|_m = |X|_m \cdot |Y|_m = |x_i \cdot y_i|_m \]

In general,
\[ \prod_{j=1}^{k} X_{j} = \prod_{j=1}^{k} |X_{j}|_m = |x_{i} \cdot y_{i}|_m \]
2. ARITHMETIC OPERATIONS

Example 11.2

\( X=1, \ Y=2 \) \quad \left( m_2, m_1 \right) = (3, 2).

\[ X+Y = (1, 1) + (2, 0) = (0, 1) \] representing 3.

\[
\begin{align*}
|x_2 + y_2|_{m_2} &= |1 + 2|_3 = 0 \\
|x_1 + y_1|_{m_1} &= |1 + 0|_2 = 1
\end{align*}
\]

\( X \cdot Y = (1, 1) \cdot (2, 0) = (2, 0) \) representing 2.

\[
\begin{align*}
|x_2 \cdot y_2|_{m_2} &= |1 \cdot 2|_3 = 2 \\
|x_1 \cdot y_1|_{m_1} &= |1 \cdot 0|_2 = 0
\end{align*}
\]
2. ARITHMETIC OPERATIONS

Subtraction:

Additive inverse of $c$ modulo $m_i$

$$\left| -c \right|_{m_i} = \left| m_i - c \right|_{m_i}, \text{ since } \left| m_i \right|_{m_i} = 0.$$ 

$$\left| X - Y \right|_{m_i} = \left| X \right|_{m_i} - \left| Y \right|_{m_i} = \left| x_i - y_i \right|_{m_i} = \left| x_i + \left| -y_i \right|_{m_i} \right|_{m_i}$$

For example, $X = 5 = (2, 1)$ and $Y = 3 = (0, 1)$

$$\left| x_2 - y_2 \right|_{m_2} = \left| 2 - 0 \right|_{3} = \left| 2 + 0 \right|_{3} = 2$$

$$\left| x_1 - y_1 \right|_{m_1} = \left| 1 - 1 \right|_{2} = \left| 1 + 1 \right|_{2} = 0$$

$$\Rightarrow X - Y = (2, 0) \text{ representing } 2.$$
2. ARITHMETIC OPERATIONS

Example 11.3

\((m_4, m_3, m_2, m_1) = (7, 5, 3, 2)\).

\[ M = \text{l.c.m}(m_1, \ldots, m_4) = \prod_{i=1}^{4} m_i = 210. \]

\[
\begin{array}{c}
(7 \, 5 \, 3 \, 2) \\
3 \\
4 + \\
7 \\
\hline
(7 \, 5 \, 3 \, 2) \\
(3 \, 3 \, 0 \, 1) \\
(4 \, 4 \, 1 \, 0) \\
(0 \, 2 \, 1 \, 1) \\
\hline
(7 \, 5 \, 3 \, 2) \\
(3 \, 3 \, 0 \, 1) \\
(4 \, 4 \, 1 \, 0) \\
12 \\
\hline
(7 \, 5 \, 3 \, 2) \\
(3 \, 1 \, 2 \, 0) \\
(0 \, 2 \, 1 \, 1) \\
\hline
(3 \, 3 \, 0 \, 1) \\
\end{array}
\]

\[ (3,3,0,1) = 3 = |213|_{210} \]

Overflow is difficult to identify.
2.1 Multiplicative Inverse

The multiplicative inverse does not always exist

<table>
<thead>
<tr>
<th>m = 5</th>
<th>m = 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>$\frac{1}{c} \mod m$</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

The unique inverse exists iff $g.c.d(c, m) = 1$ and $|c|_m \neq 0$.

If $m$ is a prime number, the inverse exists for $1 \leq c \leq m - 1$
3. ASSOCIATED MIXED-RADIX SYSTEM

- Associated mixed-radix number system for magnitude comparison, sign detection, and overflow detection.
- A weighted number system.

\[ X = a_N \cdot (m_{N-1} \cdot m_{N-2} \cdots m_1) + \cdots + a_3 \cdot (m_2 \cdot m_1) + a_2 \cdot m_1 + a_1 \]

\[ 0 \leq a_i < m_i; \quad i = 1, 2, \cdots, N. \]

Example: \((m_4, m_3, m_2, m_1) = (7, 5, 3, 2)\)

\[ X = 30 \cdot a_4 + 6 \cdot a_3 + 2 \cdot a_2 + a_1 \]

\[ 43 = (1, 2, 0, 1) > (1, 1, 0, 1) = 37 \]
3. ASSOCIATED MIXED-RADIX SYSTEM

\[ a_1 = X \mod m_1 = x_1 \]
\[ a_2 = (X - a_1) \left\lfloor \frac{1}{m_1} \right\rfloor \mod m_2 \]
\[ a_3 = \left( (X - a_1) \left\lfloor \frac{1}{m_1} \right\rfloor - a_2 \right) \left\lfloor \frac{1}{m_2} \right\rfloor \mod m_3 \]
\[ \vdots \]

The above calculations come from

\[ Y_{i+1} = (Y_i - a_i) \left\lfloor \frac{1}{m_i} \right\rfloor \]  with  \[ Y_1 = X \]

\[ a_i = Y_i \mod m_i \]
Example 11.5

\[ X = (x_4, x_3, x_2, x_1), \text{ where } (m_4, m_3, m_2, m_1) = (7, 5, 3, 2) \]

\[ a_1 = X \mod 2 = x_1 \]
\[ a_2 = (X - a_1) \mod 3, \]
\[ a_3 = ((X - a_1) - a_2) \mod 5, \]
\[ a_4 = (((X - a_1) - a_2) - a_3) \mod 7. \]
3. ASSOCIATED MIXED-RADIX SYSTEM

Example 11.5   For $X = 43 = (1,3,1,1)$

$Y_1 = (1,3,1,1)$ and therefore, $a_1 = Y_1 \mod 2 = x_1 = 1$.

$Y_1 - a_1 = (0,2,0,-), |\frac{1}{2}| = (4,3,2,-)$

$\rightarrow Y_2 = Y_1 \mid \frac{1}{2} \mid = (0,1,0,-), a_2 = Y_2 \mod 3 = 0$.

$Y_2 - a_2 = (0,1,-,-), \mid \frac{1}{3} \mid = (5,2,-,-)$

$\rightarrow Y_3 = Y_2 \mid \frac{1}{3} \mid = (0,2,-,-), a_3 = Y_3 \mod 5 = 2$.

$Y_3 - a_3 = (5,-,-,-), \mid \frac{1}{5} \mid = (3,-,-,-)$

$\rightarrow Y_4 = Y_3 \mid \frac{1}{5} \mid = (1,-,-,-), a_4 = Y_4 \mod 7 = 1$. 
3. ASSOCIATED MIXED-RADIX SYSTEM

- The mixed-radix number system is useful for overflow detection.
  
  1. Add a redundant modulus $m_{N+1}$.
  2. Convert $(x_{N+1}, x_N, ..., x_1)$ to the associated mixed number system.
  3. If $a_{N+1} \neq 0$, an overflow has occurred.
Chinese Remainder Theorem (CRT)

\[
\left| X \right|_M = \left| \sum_{j=1}^{N} \hat{m}_j \left| \frac{x_j}{\hat{m}_j} \right|_{m_j} \right|_M
\]

where \( \hat{m}_j = \frac{M}{m_j} \), \( M = \prod_{j=1}^{N} m_j \)

and all the values of \( m_j \) are pairwise relatively prime.
Example 11.6

\[(m_3, m_2, m_1) = (7, 3, 2) \rightarrow M = 42\]

\[X = (x_3, x_2, x_1) = (0, 2, 1)\]

\[\hat{m}_1 = \frac{M}{m_1} = \frac{42}{2} = 21; \quad \hat{m}_2 = \frac{42}{3} = 14; \quad \hat{m}_3 = \frac{42}{7} = 6\]

\[|X|_{42} = |36x_3 + 28x_2 + 21x_1|_{42} = |77|_{42} = 35.\]
Alternate form:

\[ \left| X \right|_M = \left| A_3 x_3 + A_2 x_2 + A_1 x_1 \right|_M \]

where \( A_3 \) is the value of (1,0,0),
\( A_2 \) is the value of (0,1,0) and
\( A_1 \) is the value of (0,0,1).
4.1 Binary to the Residue System

Given \( X = \sum_{j=0}^{n} x_j 2^j \) with \( x_j \in \{0, 1\} \),

\[
|X|_m = \left| \sum_{j=0}^{n} x_j 2^j \right|_m
\]

where \( |2^j|_m \) can be precalculated and stored in a table.

Example 11.7

\( X = 1101101_2 \), To find \(|1101101|_3\)

\[
\begin{align*}
    |2^0|_3 &= 1, \\
    |2^1|_3 &= 2, \\
    |2^2|_3 &= 1, \\
    |2^3|_3 &= 2, \\
    &\cdots
\end{align*}
\]

\(|1101101|_3 = |1+2+2+1+1|_3 = 1.\)
5. Selecting the Moduli

• A large number of small moduli is desirable, since the execution time is determined by the largest modulus.

• Residues are normally coded in some binary code.
  1. Efficient binary representation to minimize the total number of bits.
  2. Convenient binary coding to simplify the execution of arithmetic operations

• The smallest number of bits to represent the residue digit for $m_i$ is $\lceil \log_2 m_i \rceil$

• We prefer to select an $m_i$ that equals $2^k$ or is close to it; that is $(2^k - 1)$.

• Not all terms of $(2^l - 1)$, since $(2^k - 1) = (2^{k/2} - 1) (2^{k/2} + 1)$. 
5. Selecting the Moduli

Example 11.8
4 moduli: \(32 = 2^5, 31 = (2^5-1), 15 = (2^4-1), 7 = (2^3-1)\).

Total number of bits = 5+5+4+3 = 17 bits

\[M = 32 \cdot 31 \cdot 15 \cdot 7 > 2^{16}\]

The 4 moduli yield a very efficient coding.

Example 11.9
For \(m = (2^l-1)\), the additive inverse of \(c\) is \(m - c = (2^l-1) - c\), one's complement of \(c\).
Suppose \(m = 7\), and we subtract 4 from 6.

\[
\begin{array}{cccc}
110 & + & 011 & \text{one's complement of } 100=4_{10} \\
& & & \text{End-around carry} \\
1 & 001 & & 010
\end{array}
\]
5. Selecting the Moduli

- For moduli different from $2^k$ or $(2^k - 1)$, look-up tables must be used.

<table>
<thead>
<tr>
<th>+</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>×</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

**TABLE 11.2** Modulo 5 addition and multiplication tables.
In most cases, the conventional binary coding is used for the digits.

We may select a different coding shown below, where the pairs 1 and 4, and 2 and 3 are additive inverse and also one’s complements.

<table>
<thead>
<tr>
<th>Digit</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary Code</td>
<td>000 or 111</td>
<td>001</td>
<td>010</td>
<td>101</td>
<td>110</td>
</tr>
</tbody>
</table>

**TABLE 11.3** Alternate binary coding for residues modulo 5.
6. **ERROR DETECTION AND CORRECTION**

6.1 Error Codes for Conventional Number Systems

An error code is preserved under arithmetic operation $*$ if for $X$ and $Y$, and the corresponding encoded entities $X'$ and $Y'$, there is an operation $\otimes$ satisfying

$$(X' \otimes Y')' = (X * Y)'$$

The simplest non-separate code is the **AN-code**, which is formed by multiplying the operand by a constant $A$.

- $X'$ is $A \cdot X$
- $\otimes$ and $*$ are identical.
- The error is easily detectable when $X'$ is not a multiple of $A$. 
The simplest separate code is the residue code and the inverse residue code, where a separate check symbol $C(X)$ is attached to every operand $X$.

- For the residue code, $C(X) = X \mod A$, where $A$ is the check modulus.
- For the inverse residue code, $C(X) = A - (X \mod A)$.
- For both codes,

$$C(X) \otimes C(Y) = C(X \times Y).$$

- The above holds for multiplication, addition, and subtraction.
- For division, $X - S = Q \cdot D$, therefore

$$\left| X \right|_A - \left| S \right|_A = \left| Q \right|_A \cdot \left| D \right|_A.$$
In the error detection block, the residue modulo of the $X+Y$ input is calculated and compared to the other input.

**FIGURE 11.1** An adder with a separate residue check.
6. ERROR DETECTION AND CORRECTION

- $A = (2^a - 1)$, with $a$ being an integer, simplifies the calculation of the remainder when dividing by $A$.
- The calculation of the remainder when dividing by $(2^a - 1)$ is simple, because

$$|z_i r^i|_{r-1} = |z_i|_{r-1}, \quad r = 2^a.$$

- Allows the use of modulo $(2^a - 1)$ summation of the groups of $a$ bits.

Example: $X = 1101101_2$, To find $|1101101|_3$

$$|1101101|_3 = |1+3+2+1|_3 = 1.$$
Example 11.11

Calculate the remainder by dividing \( X = 11_{110_{101_{011}}} \) by \( A = 7 \).

\[
\begin{array}{cc}
11 & z_3 \\
+ & 110 \\
\hline
1 & 001 \\
+ & 1 \\
\hline
010 & \text{end-around carry} \\
+ & 101 \\
\hline
111 & z_1 \\
+ & 011 \\
\hline
1 & 010 \\
+ & 1 \\
\hline
& 011 \\
+ & 011 \\
\hline
\end{array}
\]

\(|8|_7 = 1\)
• To include signed operands, the code should be complementable with respect to $R$, where $R$ is $2^n$ or $2^n - 1$.

• The original operand is complementable with respect to $M$, where $M$ is $2^m$ or $2^m - 1$, but with $m < n$.

• For the $AN$ code, $R - AX = A(M - X)$, $\therefore R = AM$.

• If $A$ is odd, $R = 2^n$ is excluded, which makes $A$ being a factor of $2^n - 1$. 
Example 11.12

For \( n = 4 \), \( R \) is \( 2^n - 1 = 15 \), and is divisible by \( A = 3 \).

\[ X = 0110, \quad 3X = 010010 \]

-> Its 1’s complement 101101 is divisible by 3,
    but the 2’s complement 101110 is not divisible by 3.

If \( n = 5 \), \( R = 31 \), which is not divisible by 3.

\[ X = 00110, \quad 3X = 0010010, \]

-> Its 1’s complement 1101101 is not divisible by 3.
6. ERROR DETECTION AND CORRECTION

- For the residue code with check modulus $A$, 
  \[ A - C(X) = (R - X) \mod A \]

- $R$ must be an integer multiple of $A$, allowing only 1’s complement arithmetic.

- $(2^n - X) \mod A = (2^n - 1 - X + 1) \mod A$
  \[ = (2^n - 1 - X) \mod A + 1 \mod A \]

- When performing 2’s complement, we need to add a correction term $|1|_A$.

- In 2’s complement arithmetic, a carryout is discarded. To compensate for this, we need to subtract $|2^n|_A = |1|_A$. 
Example 11.13
For the residue code with $A = 7$ and $n = 6$, $R$ is $2^6 = 64$, and $R - 1 = 63$ is divisible by 7.

$10_{10} = 001010_2$ has the residue 3 modulo 7.
2’s complement of $001010_2 = 110110$
The complement $|3|_7 = |4|_7$
-> adding $|1|_7$ yields 5, which is $110110 \mod 7$.

$$
\begin{align*}
110110 &= X \\
001101 &= Y \\
\hline
1000011 &= X + Y \\
\end{align*}
$$

110110 = X
+ 001101 = Y
= 1000011
1 000011

1 011
\hline
1 011
\hline

1 end-around carry
100
- 1 correction term
011
6. Error Detection and Correction

Error Correction by using 2 or more residue checks.

- The simplest case is the biresidue code consisting of two residue checks, $A_1 = 2^a - 1$ and $A_2 = 2^b - 1$, where $n = \text{l.c.m} (a, b)$ is the number of bits in the operand.
  - Any single-bit error can be corrected.

The residue system is inherently more fault-tolerant.

- A fault in a residue digit does not result in errors in other digits.
  - Fault isolation property.
A set of redundant moduli allows the detection of errors and the identification of faulty residue digit circuits.

- Assume \( L \) redundant moduli
- \( M_T = \prod_{i=1}^{N+L} m_i \), \( M = \prod_{i=1}^{N} m_i \)
- \([0, M - 1] : \text{legitimate range}\)
- \([M, M_T - 1] : \text{illegitimate range}\)
- A single error always moves the operand from the legitimate range to the illegitimate range