Microprocessor and Computer Architecture

Dr. Trần Thị Anh Xuân

Department of Instrumentation and Industrial Informatics

School of Electrical Engineering

Documents

- MCS-51 Microcontroller Family Users Manual – Intel 1994

Requirements for students

- Experiments/Practice
- Evaluation (EE3480E):
  - Midterm Exam: Writing (can be combined with MCQ) or/combine with Project
  - Final Exam: Writing or Oral exam
- Students need to participate fully in lectures (online and offline)

Syllabus

- Chapter 1: Computer Architecture
- Chapter 2: MCS-51 Microcontroller
- Chapter 3: Assembler language
- Chapter 4: Digital Input/Output
- Chapter 5: Basic peripheral connection

1.1. Numbering and Coding Systems

- In this subjects:

Number Systems	**Base (Radix)**	**Used Symbols**
Decimal	10	0,1,2,..,9
Binary	2	0,1
Octal	8	0,1,2,..,7
Hexa-Decimal	16	0,1,2,,9, A, B,…,F

1.2. Number Systems and Conversion

- **Conversion of other Base-R to Decimal**

$$
\begin{aligned}
N &= (a_4 a_3 a_2 a_1 a_0 . a_{-1} a_{-2} a_{-3})_R \\
&= a_4 \times R^4 + a_3 \times R^3 + a_2 \times R^2 + a_1 \times R^1 + a_0 \times R^0 \\
&\qquad + a_{-1} \times R^{-1} + a_{-2} \times R^{-2} + a_{-3} \times R^{-3}
\end{aligned}
$$

1.2. Number Systems and Conversion

- Conversion of other Base-R to Decimal

$$N = (a_4a_3a_2a_1a_0.a_{-1}a_{-2}a_{-3})_R$$
$$= a_4 \times R^4 + a_3 \times R^3 + a_2 \times R^2 + a_1 \times R^1 + a_0 \times R^0$$
$$+ a_{-1} \times R^{-1} + a_{-2} \times R^{-2} + a_{-3} \times R^{-3}$$

- Example:

$$\begin{aligned} 1011.101_2 &= (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) \\ &\quad + (1 \times 2^{-1}) + (0 \times 2^{-2}) + (1 \times 2^{-3}) \\ &= 8 + 0 + 2 + 1 + 0.5 + 0 + 0.125 \\ &= 11.625_{10} \end{aligned}$$

1.2. Number Systems and Conversion

- **Conversion of other Base-R to Decimal**

$$N = (a_4 a_3 a_2 a_1 a_0 . a_{-1} a_{-2} a_{-3})_R$$
$$= a_4 \times R^4 + a_3 \times R^3 + a_2 \times R^2 + a_1 \times R^1 + a_0 \times R^0$$
$$+ a_{-1} \times R^{-1} + a_{-2} \times R^{-2} + a_{-3} \times R^{-3}$$

- Example:
  - $365{,}25_{10} = 3 \times 10^2 + 6 \times 10^1 + 5 \times 10^0 + 2 \times 10^{-1} + 5 \times 10^{-2}$
  - $11011.01_2 = 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2}$
    $= (27.25)_{10}$
  - $123.35_8 = 1 \times 8^2 + 2 \times 8^1 + 3 \times 8^0 + 3 \times 8^{-1} + 5 \times 8^{-2} = (85.453125)_{10}$
  - $26.15_{16} = 2 \times 16^1 + 6 \times 16^0 + 1 \times 16^{-1} + 5 \times 16^{-2} = (38.08203125)_{10}$

1.2. Number Systems and Conversion

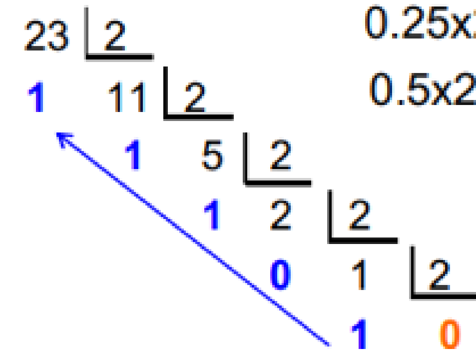
- Conversion of Decimal to other Base-R
  - Integer part:
    - Repeated division by Base-R
  - Fraction part:
    - Repeated multiplication with Base-R

1.2. Number Systems and Conversion

- **Conversion of Decimal to other Base-R**
  - Integer part:
    - Repeated division by Base-R
  - Fraction part:
    - Repeated multiplication with Base-R

- Example: $\mathbf{(23.25)_{10} = ?_2}$

Integer part



Fraction part

$0.25 \times 2 = 0.5$

$0.5 \times 2 = 1.0$

$(27.25)_{10} = (10111.01)_2$

1.2. Number Systems and Conversion

- **Conversion of Hexa-decimal to Binary**

$$
\begin{array}{llll}
9F2_{16} = & 9 & F & 2 \\
 & \downarrow & \downarrow & \downarrow \\
 & = 1\ 0\ 0\ 1 & 1\ 1\ 1\ 1 & 0\ 0\ 1\ 0 \\
 & = 100111110010_2 & &
\end{array}
$$

- **Conversion of Octal to Binary**

$705_8 = ?_2$

$$
\begin{array}{ccc}
7 & 0 & 5 \\
\downarrow & \downarrow & \downarrow \\
111 & 000 & 101
\end{array}
\Rightarrow 705_8 = 111000101_2
$$

1.2. Number Systems and Conversion

- Conversion of Binary to Octal, Hexa-decimal
  - Example:

$$(11010111110.0111)_2 = ?_8$$

$$(\underbrace{011}_{3}\underbrace{010}_{2}\underbrace{111}_{7}\underbrace{110}_{6}.\underbrace{011}_{3}\underbrace{100}_{4})_2 = (3276.34)_8$$

1.2. Number Systems and Conversion

- **Conversion of Binary to Octal, Hexa-decimal**
  - Example:

$$(11010111110.0011)_2 = (6BE.3)_{16}$$

$$(\underbrace{0110}_{6}\underbrace{1011}_{B}\underbrace{1110}_{E}.\underbrace{0011}_{3})_2 = (6BE.3)_{16}$$

1.2. Number Systems and Conversion

- Conversion of Binary to Octal, Hexa-decimal

Octal	Decimal	Binary
0	0	000
1	1	001
2	2	010
3	3	011
4	4	100
5	5	101
6	6	110
7	7	111

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

1.2. Number Systems and Conversion

- Practice: Given Base-R number. Convert into other Base-R?

Decimal	Binary	Octal	Hexa-decimal
33			
	1110101		
		703	
			1AF

1.3. Base-R number Arithmetic

- **Binary Addition**
  - Rule:

$$0+0=0$$
$$0+1=1$$
$$1+0=1$$
$$1+1=0 \quad \text{and carry 1 to the next column}$$

  - Example:

$$
\begin{array}{rcr}
 & & 1111 \longleftarrow \text{carries} \\
13_{10} & = & 1101 \\
11_{10} & = & \underline{1011} \\
 & & 11000 = 24_{10}
\end{array}
$$

1.3. Base-R number Arithmetic

- **Binary Subtraction**
  - Rule:

$$
\begin{aligned}
0-0&=0\\
0-1&=1 \quad \text{and return borrow 1 to the next column}\\
1-0&=1\\
1-1&=0
\end{aligned}
$$

  - Example:

$$
\begin{array}{rcccc}
C = & 1 & 1 & 0 & \\
 & 1 & 1 & 0 & 1 \\
- & 0 & 1 & 1 & 1 \\
\hline
 & 0 & 1 & 1 & 0
\end{array}
\qquad\qquad
\begin{array}{rr}
 & 13_{10} \\
- & 7_{10} \\
\hline
 & 6_{10}
\end{array}
$$

1.3. Base-R number Arithmetic

- **Binary Multiplication**
  - Rule:

$$0 \times 0 = 0$$
$$0 \times 1 = 0$$
$$1 \times 0 = 0$$
$$1 \times 1 = 1$$

  - Example:

$$\begin{array}{rrl}
 & 1101 & \text{multiplicand} \\
\times & 1011 & \text{multiplier} \\
\hline
 & 1101 & \\
 & 1101 & \\
+ & 0000 & \\
 & 1101 & \\
\hline
 & 10001111 = 143_{10} &
\end{array}$$

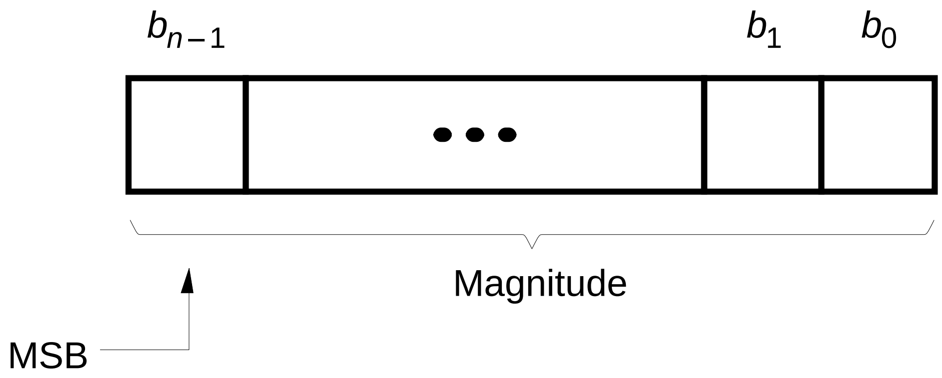
1.3. Base-R number Arithmetic

- Binary Division

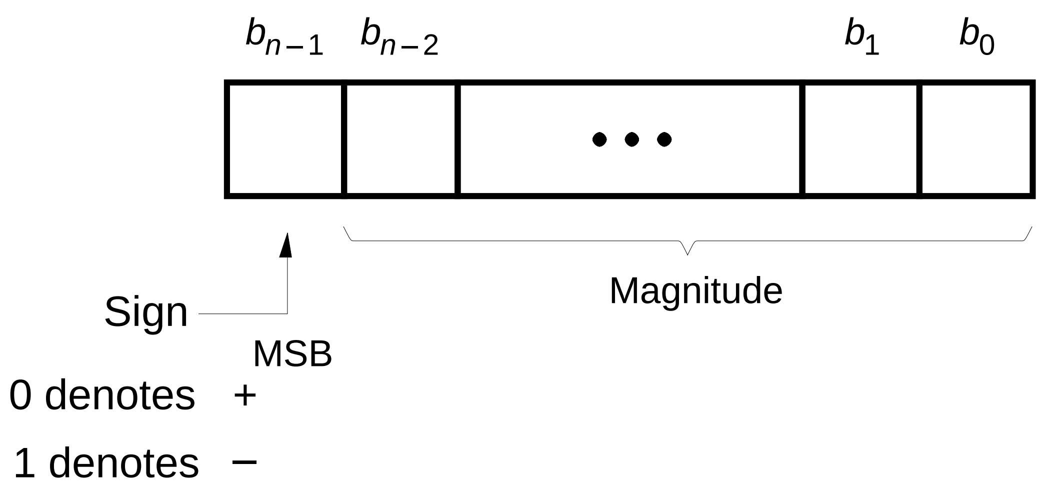
```
     1 0 1 1  |  1 0
              |-------
  -  1 0      |  1 0 1
  ------      |
       0 1    |
    -  0 0    |
    ------    |
         1 1  |
      -  1 0  |
      ------  |
         0 1  |
```

1.4. Negative Numbers

- Unsigned and Signed Number



(a) Unsigned number



(b) Signed number

1.4. Negative Numbers

- **Method 1: Signed-Magnitude Representation**
  - Rule:

$$\pm\ N = (\ s,\ a_{n-1}\ \ldots\ldots a_1\ a_0)$$

Signed magnitude

$$s = 0 \quad \text{if } N \geq 0$$

$$s = 1 \quad \text{if } N \leq 0$$

  - Range for n bits:

$$-(2^{n-1}-1) \text{ through } +(2^{n-1}-1)$$

  - Example: 5 bit signed-magnitude Binary

$$N = +13_{10} = 01101$$

$$N = -13_{10} = 11101$$

1.4. Negative Numbers

- **Method 2: Two's-Complement Representation**
  - Binary Number : $N = a_{n-1}a_{n-2}\ldots a_1a_0$
  - 1's complement : $B^{(1)}_N : a_k \rightarrow a'_k$ $\begin{cases} 1 \rightarrow 0 \\ 0 \rightarrow 1 \end{cases}$
  - Example:

$$N_2 = 1\ 0\ 1\ 1\ 0\ 1\ 0\ 0 \longrightarrow B^{(1)}_N = ?$$

$$\Downarrow$$

$$B^{(1)}_N = 0\ 1\ 0\ 0\ 1\ 0\ 1\ 1$$

  - 2's complement : 1^{st} way: $B^{(2)}_N = B^{(1)}_N + 1$

$$\begin{array}{rr} N = & 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1 \\ B^{(1)}_N = & 1\ 0\ 0\ 1\ 1\ 0\ 1\ 0 \\ + & 1 \\ \hline B^{(2)}_N = & 1\ 0\ 0\ 1\ 1\ 0\ 1\ 1 \end{array}$$

1.4. Negative Numbers

- Method 2: Two's-Complement Representation
  - Binary Number : $N = a_{n-1}a_{n-2}\ldots a_1a_0$
  - 2's complement : 2^{nd} way

$$N = 0\ 1\ 1\ 0\ 0\ 1\ 0\ 1$$

$$B^{(2)}_N = 1\ 0\ 0\ 1\ 1\ 0\ 1\ 1$$

$$N = 1\ 1\ 0\ 1\ 0\ 1\ 0\ 0$$

$$B^{(2)}_N = 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0$$

1.4. Negative Numbers

- Method 2: Two's-Complement Representation
  - Positive number : the same Method 1
  - Negative number : = $B^{(2)}_N$
  - Range for n bits:

$$-2^{n-1} \text{ through } +(2^{n-1}-1)$$

1.5. BCD code

- Binary-Coded-Decimal (BCD) Code:
  - *Each* digit of a decimal number is represented by its binary equivalent
  - Since a decimal digit from 0 to 9 → ***four bits*** are required to code ***each digit*** (0000 through 1001 are used)
- Example:

8	7	4	(decimal)
↓	↓	↓	
1000	0111	0100	(BCD)

9	4	3	(decimal)
↓	↓	↓	
1001	0100	0011	(BCD)

1.5. BCD code

- Depending on the store way of BCD code in digital system
  - Unpacked-BCD Code:
    - Each decimal digit is encoded into one byte,
  - Packed-BCD Code:
    - Two decimal digits are encoded into a single byte,
  - Example:

Decimal	:	9	1
Unpacked BCD	:	0000 1001	0000 0001
Decimal	:	9	1
Packed BCD	:	1001	0001

1.5. BCD code

- BCD Subtraction: the same way with Binary Subtraction
- BCD Addition: need adjust the result

1.6. ASCII

- ASCII (American Standard Code for Information Interchange)

		$b_6b_5b_4$ (column)							
$b_3b_2b_1b_0$	Row (hex)	000 0	001 1	010 2	011 3	100 4	101 5	110 6	111 7
0000	0	NUL	DLE	SP	0	@	P	`	p
0001	1	SOH	DC1	!	1	A	Q	a	q
0010	2	STX	DC2	"	2	B	R	b	r
0011	3	ETX	DC3	#	3	C	S	c	s
0100	4	EOT	DC4	$	4	D	T	d	t
0101	5	ENQ	NAK	%	5	E	U	e	u
0110	6	ACK	SYN	&	6	F	V	f	v
0111	7	BEL	ETB	'	7	G	W	g	w
1000	8	BS	CAN	(	8	H	X	h	x
1001	9	HT	EM	)	9	I	Y	i	y
1010	A	LF	SUB	*	:	J	Z	j	z
1011	B	VT	ESC	+	;	K	[	k	{
1100	C	FF	FS	,	<	L	\	l	|
1101	D	CR	GS	-	=	M	]	m	}
1110	E	SO	RS	.	>	N	^	n	~
1111	F	SI	US	/	?	O	_	o	DEL