Data representation

College of Saint Benedict & Saint John's University

5		8		0		3		6	
50000	+	8000	+	0	+	30	+	6	

5		8		0		3		6	
50000	+	8000	+	0	+	30	+	6	
5 × 10000	+	8 × 1000	+	0 × 100	+	3 × 10	+	6 × 1	

5		8		0		3		6
50000	+	8000	+	0	+	30	+	6
5 × 10000	+	8 × 1000	+	0 × 100	+	3 × 10	+	6 × 1
5 × 10 ⁴	+	8 × 10 ³	+	0 × 10 ²	+	3 × 10 ¹	+	6 × 10 ⁰

1	0	1	1	0
1×2^4	$+ 0 \times 2^{3}$	$+ 1 \times 2^{2}$	$+ 1 \times 2^{1}$	$+ 0 \times 2^{0}$

1	0	1	1	0
1×2^4	$+ 0 \times 2^{3}$	$+ 1 \times 2^{2}$	$+ 1 \times 2^{1}$	$+ 0 \times 2^{0}$
1 × 16	+ 0 × 8	+ 1 × 4	+ 1 × 2	+ 0 × 1

1	0		1		1		0
1 × 2 ⁴	+ 0×	: 2 ³ +	1×2^2	+	1 × 2 ¹	+	0 × 2 ⁰
1 × 16	+ 0×	. 8 +	1 × 4	+	1 × 2	+	0 × 1
16	+ 0	+	4	+	2	+	0

unsigned addition

$$0 + 0 = 0$$
 $0 + 1 = 1$
 $1 + 0 = 1$
 $1 + 1 = 10$

unsigned addition

$$0 + 0 = 0$$

$$0 + 1 = 1$$

$$1 + 0 = 1$$

$$1 + 1 = 10$$

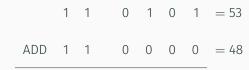
$$0 0 0 1 0 1 = 5$$

$$C \leftarrow 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad = 10$$

 $0 \quad 1 \quad 0 \quad 1 = 5$

ADD

more unsigned addition



more unsigned addition

signed addition

signed addition

method of complements

- technique used to subtract one number from another using only addition of positive numbers
- represent negative numbers as two's complement of their positive counterparts

method of complements

- technique used to subtract one number from another using only addition of positive numbers
- represent negative numbers as two's complement of their positive counterparts

two's complement

- find one's complement
- · add one



NOT	0	0	0	1	0	1
	1	1	1	0	1	0

NOT	0	0	0	1	0	1	
	1	1	1	0	1	0	
	0	0	0	1	0	1	
ADD	1	1	1	0	1	0	

NOT	0	0	0	1	0	1
	1	1	1	0	1	0
	0	0	0	1	0	1
ADD	1	1	1	0	1	0
C = 0	1	1	1	1	1	1

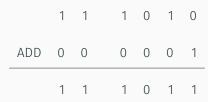
two's complement

NOT	0	0	0	1	0	1			
	1	1	1	0	1	0			
	0	0	0	1	0	1			
ADD	1	1	1	0	1	0			
C = 0	1	1	1	1	1	1	-		
ADD	0	0	0	0	0	1	_		

two's complement



two's complement cont'd



cpu bits

	0	1
N	otherwise	result is negative
Z	otherwise	result is all zeros
٧	otherwise	signed integer overflow occurred
С	otherwise	unsigned integer overflow occurred

register transfer language

operation	RTL symbol
AND	^
OR	V
XOR	•
NOT	_
Implies	\rightarrow
Transfer	←
Bit index	()
Informal description	{ }
Sequential separator	;
Concurrent separator	,

register transfer language

operation	RTL symbol
AND	^
OR	V
XOR	\oplus
NOT	一
Implies	\rightarrow
Transfer	←
Bit index	()
Informal description	{ }
Sequential separator	,
Concurrent separator	,

$$c \leftarrow a \oplus b$$
; $N \leftarrow c < 0, Z \leftarrow c = 0$

another example

	0	0	0	1	0	1
ADD	1	1	1	0	1	1
N ← 0	0	0	0	0	0	0
$Z \leftarrow 1$						
$V \leftarrow ?$						
$C \leftarrow 1$						

another example

arithmetic shift

arithmetic shift left (asl)

```
\begin{split} \text{C} \leftarrow \text{r}\langle 0 \rangle, \ \text{r}\langle 0..4 \rangle \leftarrow \langle 1..5 \rangle, \ \text{r}\langle 5 \rangle \leftarrow 0; \\ \text{N} \leftarrow \text{r} < 0, \ \text{Z} \leftarrow \text{r} = 0, \ \text{V} \leftarrow \{\text{overflow}\} \end{split}
```

arithmetic shift right (asr)

?

arithmetic shift

arithmetic shift left (asl)

$$\begin{split} C \leftarrow r\langle 0 \rangle, \ r\langle 0..4 \rangle \leftarrow \langle 1..5 \rangle, \ r\langle 5 \rangle \leftarrow 0; \\ N \leftarrow r < 0, \ Z \leftarrow r = 0, \ V \leftarrow \{\text{overflow}\} \end{split}$$

arithmetic shift right (asr)

$$C \leftarrow r\langle 5 \rangle$$
, $r\langle 1...5 \rangle \leftarrow \langle 0...4 \rangle$;
 $Z \leftarrow r = 0$

hexadecimal

unicode

Hello world.

¡Hola!, Grüß Gott, Hyvää päivää, Tere õhtust, Bonġu Cześć!, Dobrý den 你好, 早晨, こんにちは

unicode

Hello world.

¡Hola!, Grüß Gott, Hyvää päivää, Tere õhtust, Bonġu Cześć!, Dobrý den 你好, 早晨, こんにちは

 $\rm https://www.paypal.com$

unicode

Hello world.

¡Hola!, Grüß Gott, Hyvää päivää, Tere õhtust, Bonġu Cześć!, Dobrý den 你好, 早晨, こんにちは

https://www.paypal.com

floating-point

IEEE 754

single precision 1.8.23 — excess 127 / 126 double precision 1.11.52 — excess 1023 / 1022 special values

	exponent	significand
zero	all zeros	all zeros
denormalized	all zeros	non-zero
inifinity	all ones	all zeros
not a number (NaN)	all ones	non-zero

operations that result in NaN

- The divisions 0/0 and $\pm \infty / \pm \infty$
- The multiplications 0×± ∞ and ± ∞ ×0
- The additions $\infty + (-\infty), (-\infty) + \infty$ and equivalent subtractions

floating-point

IEEE 754

single precision 1.8.23 — excess 127 / 126 double precision 1.11.52 — excess 1023 / 1022 special values

	exponent	significand
zero	all zeros	all zeros
denormalized	all zeros	non-zero
inifinity	all ones	all zeros
not a number (NaN)	all ones	non-zero

operations that result in NaN

- The divisions 0/0 and $\pm \infty / \pm \infty$
- The multiplications 0×± ∞ and ± ∞ ×0
- The additions $\infty + (-\infty), (-\infty) + \infty$ and equivalent subtractions



except where otherwise noted, this worked is licensed under creative commons attribution-sharealike 4.0 international license