- **Decimal, Hexadecimal and Binary Numbers**
  - Binary numbers are a code, and represent what the programmer intends for the code
  - Convert binary and hex numbers to unsigned decimal
  - Convert unsigned decimal to hex
  - Unsigned number line and wheel
  - Signed number line and wheel
  - Binary, Hex, Signed and Unsigned Decimal
  - Signed number representation --- 2's Complement form
  - Using the 1's complement table to find 2's complements of hex numbers
  - Overflow and Carry
  - Addition and subtraction of binary and hexadecimal numbers
  - The Condition Code Register (CCR): N, Z, V and C bits

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

What does a number represent?

Binary numbers are a code, and represent what the programmer intends for the code.

0x72 Some possible codes:
’r’ (ASCII)
INC MEM (hh ll) (HC12 instruction)
2.26V (Input from A/D converter)
114_{10} (Unsigned number)
114_{10} (Signed number)
Set temperature in room to 69 F
Set cruise control speed to 120 mph

Binary to Unsigned Decimal:

Convert Binary to Unsigned Decimal

1111011_2

$1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$

$1 \times 64 + 1 \times 32 + 1 \times 16 + 1 \times 8 + 0 \times 4 + 1 \times 2 + 1 \times 1$

123_{10}

Hex to Unsigned Decimal

Convert Hex to Unsigned Decimal

82D6 16

$8 \times 16^3 + 2 \times 16^2 + 13 \times 16^1 + 6 \times 16^0$

$8 \times 4096 + 2 \times 256 + 13 \times 16 + 6 \times 1$

33494_{10}

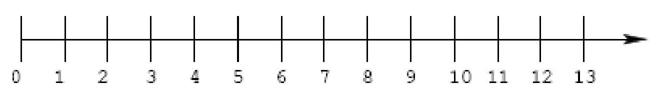
Unsigned Decimal to Hex

Convert Unsigned Decimal to Hex

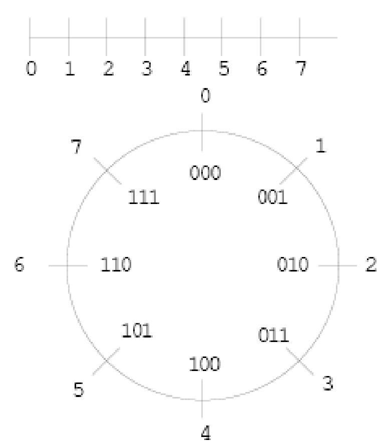
Division	Q	R	
		Decimal	Hex
721/16	45	1	1
45/16	2	13	D
2/16	0	2	2

$721_{10} = 2D1_{16}$

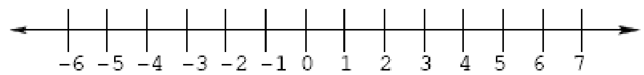
Unsigned Number Line: Numbers go from 0 to ∞



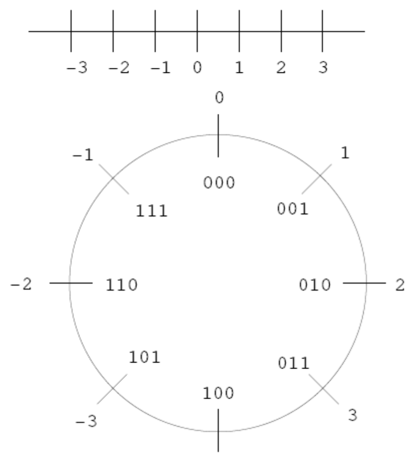
Unsigned Number Wheel: Numbers go from 0 to 2N − 1



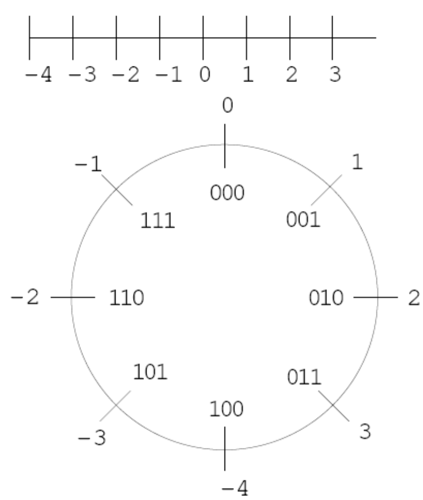
Signed Number Line: Numbers go from $-\infty$ to ∞



Number Wheel: What to do about 100_2

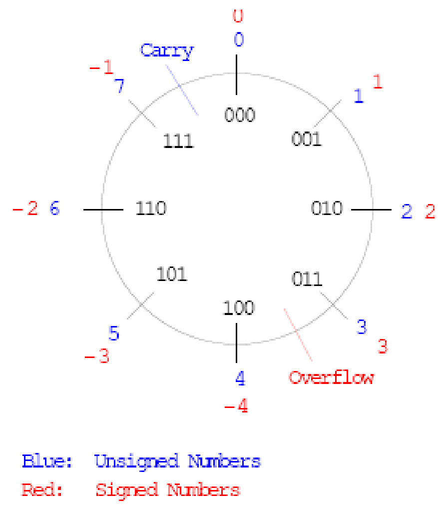


Number Wheel: Numbers go from $-2^{(N-1)}$ to $2^{(N-1)} - 1$



Number Wheel: Carry and Overflow

- Carry applies to unsigned numbers — when adding or subtracting, result is incorrect.
- Overflow applies to signed numbers—when adding or subtracting, result is incorrect.



Binary, Hex and Decimal (Signed & Unsigned) Numbers (4-bit representation)

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

Signed Number Representation in 2’s Complement Form:

If most significant bit is 0 (most significant hex digit 0−7), number is positive. Get decimal equivalent by converting number to decimal, and using + sign. Example for 8−bit number: $3A_{16} \rightarrow +(3 \times 16^1 + 10 \times 16^0)_{10}$ $+(3 \times 16 + 10 \times 1)_{10}$ $+58_{10}$
If most significant bit is 1 (most significant hex digit 8−F), number is negative. Get decimal equivalent by taking 2’s complement of number, converting to decimal, and using − sign. Example for 8−bit number: $A3_{16} \rightarrow -(5D)_{16}$ $-(5 \times 16^1 + 13 \times 16^0)_{10}$ $-(5 \times 16 + 13 \times 1)_{10}$ -93_{10}

One’s Complement Table Makes It Simple To Find 2’s Complements

0	F
1	E
2	D
3	C
4	B
5	A
6	9
7	8

To take two’s complement, add one to one’s complement.
Take two’s complement of D0C3 :
2F3C + 1 = 2F3D

- Overflow and Carry assume you have a fixed word size
- A carry is generated when you add two unsigned numbers together, and the result is too large to fit in the fixed word size.

• A carry is generated when you subtract two unsigned numbers, and the result should be negative.
• An overflow is generated when you add or subtract two signed numbers, and the fixed-length answer has the wrong sign.
Addition and Subtraction of Binary and Hexadecimal Numbers

1) Limit number of digits to specified word size.

4−bit word:

```
  1101
+ 1011
1 1000
```

Keep only 4 bits in answer

2) Does not matter if numbers are signed or unsigned − mechanics the same
Do the operation, then determine if carry and/or overflow bits are set.

4−bit word:

```
  1101 Neg
+ 1011 Neg
1 1000 Neg
```

Carry is set, overflow is clear

Condition Code Register (CCR) Gives Information On Result Of Last Operation

S	X	H	I	N	Z	V	C

Condition Code Register − 8 FFs
C − Carry : 1 −> last operation generated a carry
V − Overflow : 1 −> last operation generated an overflow
Z − Zero : 1 −> result zero, 0 −> result not zero
N − Negative : most significant bit of result
I − Interrupt mask
H − Half carry
X − Interrupt mask
S − Stop disable

Note: Not all HC12 instructions change CCR bits. A bit in the CCR is the result of the last executed instruction which affects that bit. For example, consider the following instruction sequence:

```
aba             ; Add B to A
staa $0900      ; Store A in address $0900
```

The ABA instruction will change the H, N, Z, V and C bits of the CCR. The STAA instruction will change the N and Z bit, and clear the V bit. After the two instructions, the H and C bits will reflect the result of the ABA instruction; the N and Z bits will reflect the result of the STAA instruction (was the number stored negative or zero), and the V bit will be 0.

Overflow occurs only under certain addition and subtraction operations.

- If you add a positive and a negative number, on overflow never occurs.
- If you subtract two positive numbers, an overflow never occurs.
- If you subtract two negative numbers, and overflow never occurs.