

Datoraritmetik

Dagens föreläsning:

Lärobok, kapitel 6

Arbetsbok, kapitel 5

Ur innehållet:

- hur man adderar och subtraherar tal i det binära talsystemet
- hur man kan koda om negativa binära tal, genom s k 2-komplementering, så att tecknet blir en del av talet
- hur addition och subtraktion kan utföras när talen kodats på 2-komplementsform

Binär addition – ”papper och penna metod”

Exempel: $(5)_{10} + (23)_{10} = ?$

$(5)_{10}$		$\begin{array}{r} 111 \\ \hline 101 \end{array}$	minnessiffror
$(23)_{10}$	+	$\begin{array}{r} 10111 \\ \hline \end{array}$	augend addend
$(28)_{10}$		11100	summa

Binär multiplikation – ”papper och penna metod”

Exempel: $(25)_{10} \times (11)_{10} = ?$

$(25)_{10}$		11001	multiplikand
$(11)_{10}$	×	1011	multiplikator
		11001	
		11001	
		00000	
	+	11001	
$(275)_{10}$		100010011	produkt

Binär subtraktion – ”papper och penna metod”

Exempel: $(110)_{10} - (43)_{10} = ?$

$(110)_{10}$		$\begin{array}{r} 1010 \\ \hline 1101110 \end{array}$	minnessiffror
$(43)_{10}$	-	$\begin{array}{r} 101011 \\ \hline \end{array}$	minuend subtrahend
$(67)_{10}$		1000011	skillnad

Binär division – ”papper och penna metod”

Exempel: $(33)_{10} : (6)_{10} = ?$

divisor	110	0101	kvot
		100001	dividend
	-	000	
		1000	
	-	110	
		0100	
	-	000	
		1001	
	-	110	
		011	rest

$(100001)_2 : (110)_2 =$
 $(0101)_2 + (011)_2 : (110)_2$

Addition av BCD-tal

Exempel: Utför additionen 5+7 där talen är kodade på NBCD-form.

Decimal siffra	NBCD-kodord
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	-
11	-
12	-
13	-
14	-
15	-

$$\begin{array}{cccc} & 1 & 1 & 1 \\ & 0 & 1 & 0 & 1 \\ + & 0 & 1 & 1 & 1 \\ \hline 1 & 1 & 0 & 0 & \end{array}$$

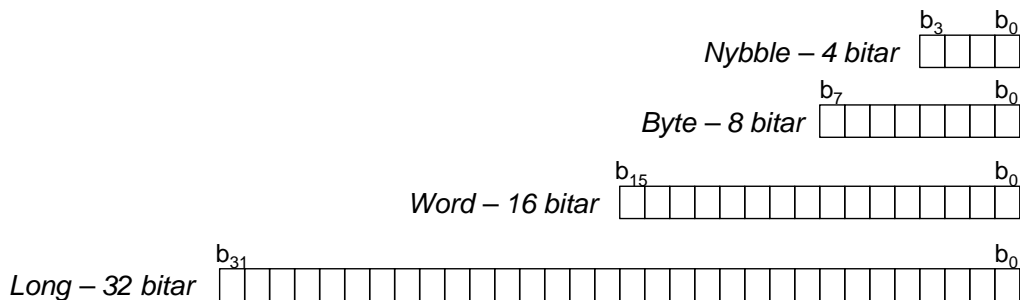
← Resultatet > 9 (ej NBCD-kod), vi tvingas därför decimaljustera, dvs, addera 6 till resultatet

$$\begin{array}{cccc} + & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \end{array}$$

← decimaljustering

Resultatet alltså $(0001\ 0010)_{NBCD} = (12)_{10}$.

Vanliga binära ordlängder



Talområden vid NBC

Antal bitar	Minsta tal	Största tal
4 (Nybble)	$(0000)_2 = (0)_{10}$	$(1111)_2 = (15)_{10}$
8 (Byte)	$(00000000)_2 = (0)_{10}$	$(11111111)_2 = (255)_{10}$
16 (Word)	$(0000000000000000)_2 = (0)_{10}$	$(1111111111111111)_2 = (65535)_{10}$
32 (Long)	$(00000000000000000000000000000000)_2 = (0)_{10}$	$(11111111111111111111111111111111)_2 = (4294967295)_{10}$

8-bitars addition

$$\begin{array}{r}
 5 \quad \quad \quad \overset{1}{00000101} \\
 + 6 \quad + \quad 00000110 \\
 \hline
 11 \quad \quad 00001011
 \end{array}$$

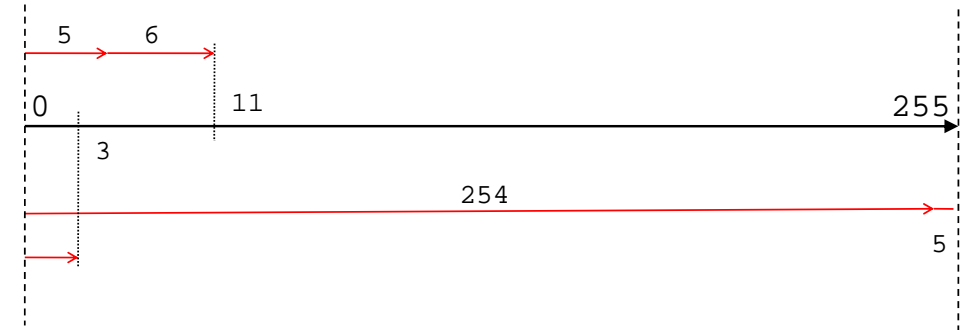
8 bitar ger talområdet $0..2^8-1 = 0..255$

$$\begin{array}{r}
 254 \quad \overset{1}{11111110} \\
 + 5 \quad + \quad 00000101 \\
 \hline
 259 \quad 100000011 \\
 \quad \quad 00000011
 \end{array}$$

Spill ! ("Overflow")

Minnessiffran, genererad från additionen av de mest signifikanta bitarna är en **spillindikator**. Vi kallar den "Carry"

Geometrisk tolkning - tallinje



8-bitars subtraktion

$$\begin{array}{r}
 254 \quad \quad \quad \overset{1}{11111110} \\
 - 5 \quad \quad - \quad 00000101 \\
 \hline
 249 \quad \quad 11111001
 \end{array}$$

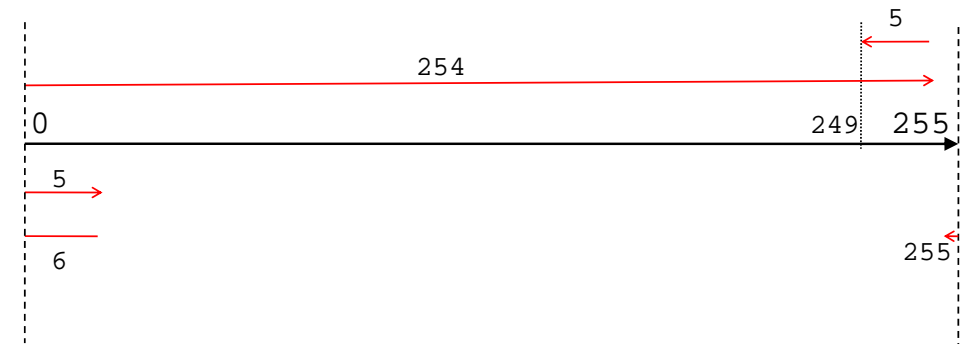
För att kunna utföra subtraktionen tvingas vi låna av en tänkt siffra med vikt 2^8 .

$$\begin{array}{r}
 5 \quad \quad \quad \overset{\oplus}{00000101} \\
 - 6 \quad \quad - \quad 00000110 \\
 \hline
 255 \quad \quad 11111111
 \end{array}$$

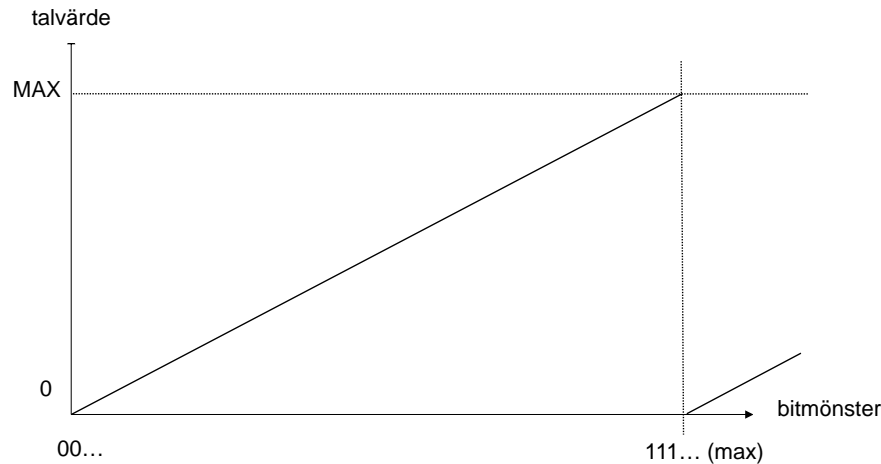
Spill ! ("Underflow")

Den tänkta "lånebiten" kallar vi "Borrow", en **spillindikator**.

Geometrisk tolkning - tallinje

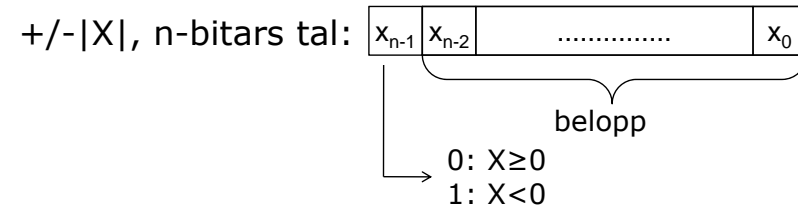


Grafisk representation av NBC tal

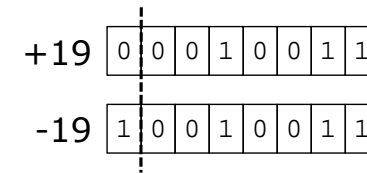


Spillfenomenet är oberoende av varje ändlig ordlängd....

Tal med tecken, Tecken/beloppsform



Exempel: 8-bitars tal +/- 19:



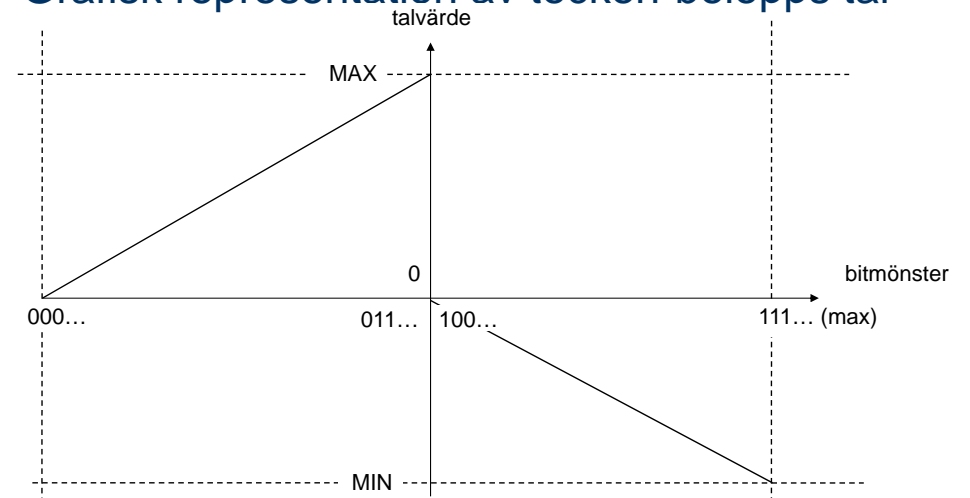
Talvärden vid tecken-belopp form

Tolkning av talvärdet N för ett n-bitars tal:

$$N = (1 - 2x_{n-1}) \sum_{i=0}^{n-2} x_i 2^i$$

n=4 $x_3x_2x_1x_0$	N
0111	7
0110	6
0101	5
0100	4
0011	3
0010	2
0001	1
0000	0
1111	-7
1110	-6
1101	-5
1100	-4
1011	-3
1010	-2
1001	-1
1000	-0

Grafisk representation av tecken-belopps tal



Den assymetriska avbildningen av talvärdet från bitmönstret antyder att aritmetiska operationer kan bli komplicerade...

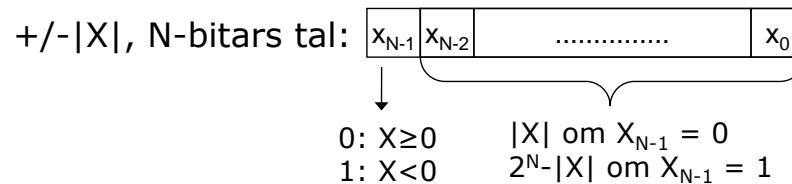
Tecken/belopsform – räkneregler för addition

Relation A och B, om:	Utförs A+B som:
$A, B \geq 0$	$ A + B $
$A \geq 0, B < 0, A > B $	$ A - B $
$A \geq 0, B < 0, A < B $	$-(B - A)$
$A < 0, B \geq 0, A > B $	$ B - A $
$A < 0, B \geq 0, A < B $	$-(A - B)$
$A, B < 0$	$-(A + B)$

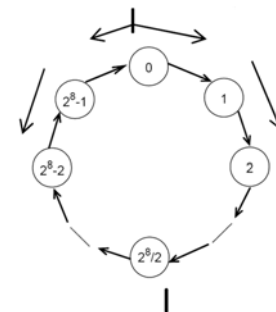
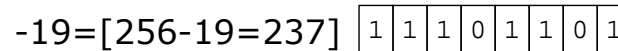
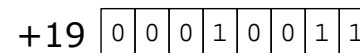
En addition kan resultera i en subtraktion. Dessutom tillkommer teckenöverläggning för resultatet.

Av tabellen framgår att ett kombinatoriskt nät för addition av tecken/belops-tal blir komplicerat.

Tal med tecken - Tvåkomplementsform



Exempel: 8-bitars tal ± 19 :



Tvåkomplementsform - Metod för teckenbyte

$X + Y = 2^n \Rightarrow Y$ är 2-komplementet till X (n-bitars tal)

För 8-bitars tal således:

$$\begin{aligned}
 Y = X_{2k} &= 2^8 - X = \\
 &= (2^8 - 1) - X + 1 \\
 &= \underbrace{11111111 - X}_{1\text{-komplement}} + 1
 \end{aligned}$$

Detta kallas 1-komplement (X_{1k}).
Bitvis invertering

Exempel: Bestäm maskintalet på 8 bitars tvåkomplementsform för decimala talet -50

Vi utgår enklast från $X=50$ (och söker X_{2k})

$(50)_{10} = X = 00110010$

$X_{1k} = 11001101$

$+ \quad \quad \quad 1$

$-(50)_{10} = X_{2k} = 11001110$

Tvåkomplementsform - addition

Relation A och B, om:	Utförs A+B som:
$A, B \geq 0$	$A+B$
$A \geq 0, B < 0,$	$A+B_{2k} = [A+(2^N-B)] \pmod{2^N} = A-B = A + (-B)$
$A < 0, B \geq 0,$	$A_{2k}+B = [(2^N-A)+B] \pmod{2^N} = -A+B = B + (-A)$
$A, B < 0$	$A_{2k}+B_{2k} = [(2^N-A)+(2^N-B)] \pmod{2^N} = -A-B = -(A+B)$

Dvs. Oavsett vilka tecken de ingående talen har så fungerar rättfram binär addition.

Tvåkomplementsform - subtraktion

Vi inser också att en subtraktion kan utföras med hjälp av en adderare ty

$$A-B = A+(-B) \text{ och } -B=B_{2k}= B_{1k}+1$$

Exempel: $6-5=6+(-5)= 00000110+ 11111010+ 1$

$$\begin{array}{r}
 (5)_{10} = \\
 (00000101)_2 \\
 \text{Dvs 1-komplement:} \\
 (11111010)_{1k} \\
 \hline
 \begin{array}{r}
 11111111 \\
 1 \\
 00000110 \\
 + 11111010 \\
 \hline
 00000001
 \end{array}
 \end{array}$$

Exempel: 4-bitars addition av $(0010)_2$ och $(0011)_2$

Bitmönstren tolkade som tal utan tecken

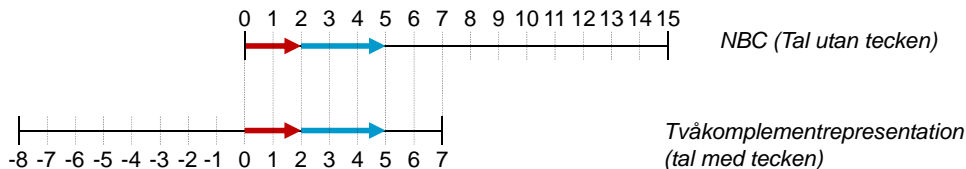
Bitmönster

Bitmönstren tolkade som tal med tecken

$$\begin{array}{r}
 2 \\
 + 3 \\
 \hline
 = 5
 \end{array}$$

$$\begin{array}{r}
 1 \\
 0010 \\
 + 0011 \\
 \hline
 = 0101
 \end{array}$$

$$\begin{array}{r}
 2 \\
 + 3 \\
 \hline
 = 5
 \end{array}$$



Exempel: 4-bitars addition av $(0110)_2$ och $(0101)_2$

Bitmönstren tolkade som tal utan tecken

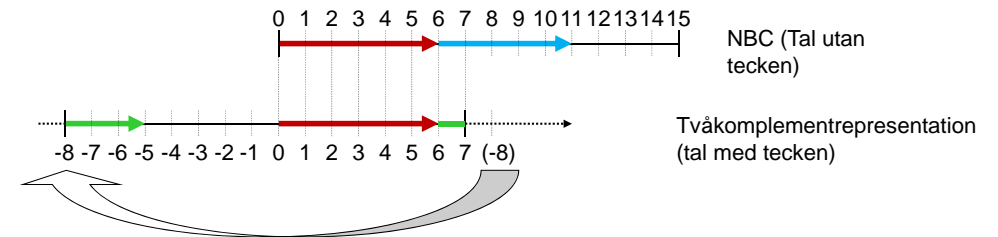
Bitmönster

Bitmönstren tolkade som tal med tecken

$$\begin{array}{r}
 6 \\
 + 5 \\
 \hline
 = 11
 \end{array}$$

$$\begin{array}{r}
 1 \\
 0110 \\
 + 0101 \\
 \hline
 = 1011
 \end{array}$$

$$\begin{array}{r}
 6 \\
 + 5 \\
 \hline
 = -5
 \end{array}$$



Exempel: 4-bitars addition av $(0101)_2$ och $(1101)_2$

Bitmönstren tolkade som tal utan tecken

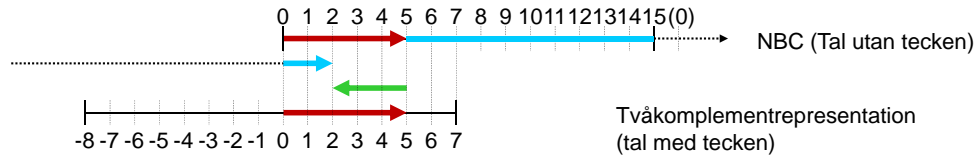
$$\begin{array}{r} 5 \\ +13 \\ \hline = 2 \end{array}$$

Bitmönster

$$\begin{array}{r} \underline{11} \ \underline{1} \\ 0101 \\ +1101 \\ \hline =0010 \end{array}$$

Bitmönstren tolkade som tal med tecken

$$\begin{array}{r} 5 \\ + -3 \\ \hline = 2 \end{array}$$



Exempel: 4-bitars addition av $(1000)_2$ och $(1100)_2$

Bitmönstren tolkade som tal utan tecken

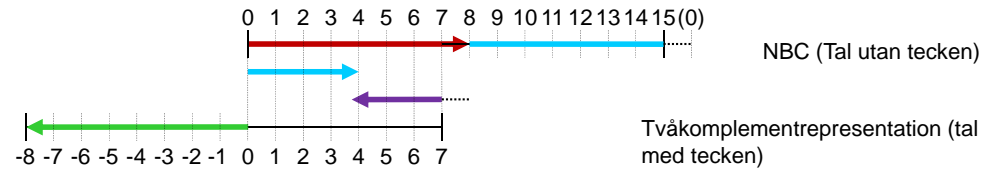
$$\begin{array}{r} 8 \\ + 12 \\ \hline = 4 \end{array}$$

Bitmönster

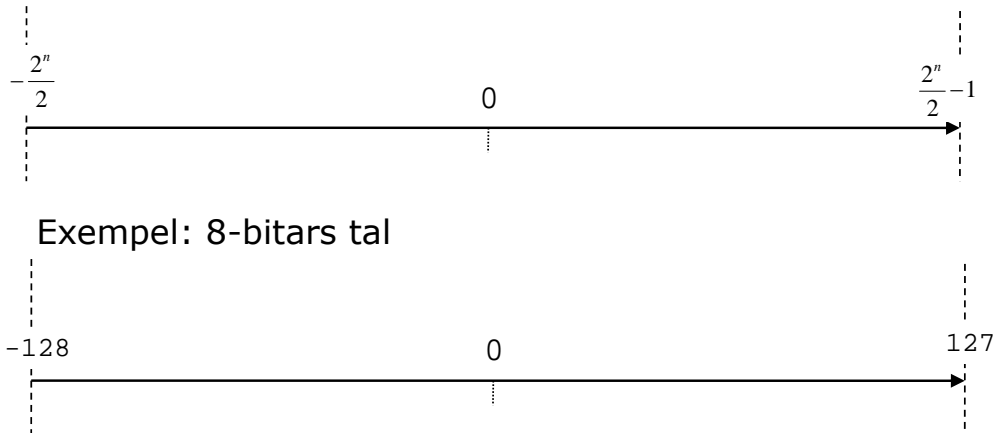
$$\begin{array}{r} \underline{1} \\ 1000 \\ + 1100 \\ \hline = 0100 \end{array}$$

Bitmönstren tolkade som tal med tecken

$$\begin{array}{r} -8 \\ + -4 \\ \hline = 4 \end{array}$$



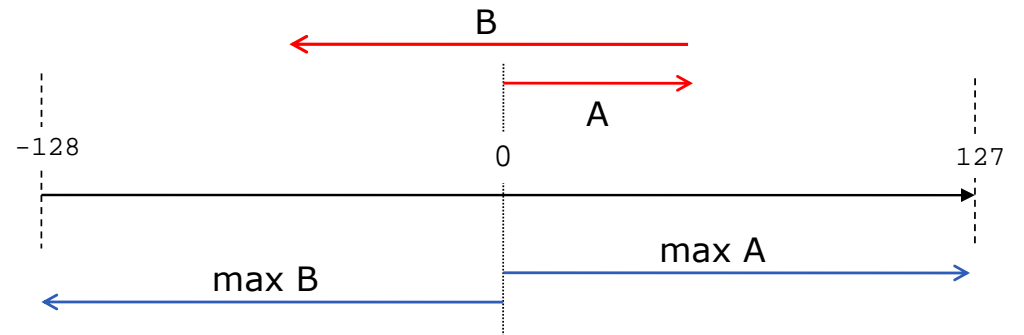
Tvåkomplementsform - talområde



Exempel: 8-bitars tal

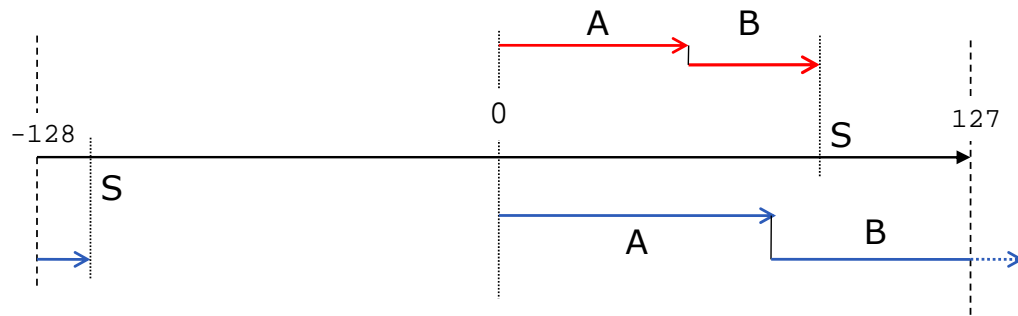
Tvåkomplementsform - spillindikatorer

$A + B$ där $A \geq 0$ och $B < 0$



Slutsats: Om A och B har olika tecken vid addition kan 2-komplementspill inte uppträda

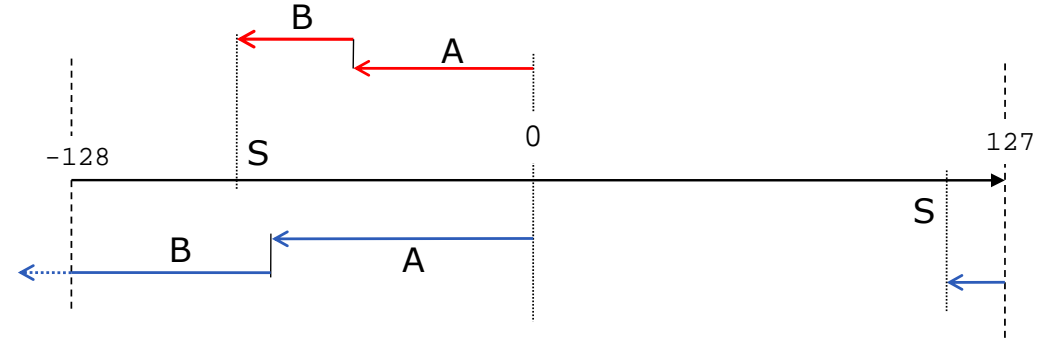
$A + B = S$, där $A \geq 0$ och $B \geq 0$



Slutsats:
Om A och B har samma tecken vid addition **kan** 2-komplementspill uppträda.
Vi kan konstatera spill genom en teckenöverläggning, dvs:

$$spill = (A \geq 0) (B \geq 0) (S < 0)$$

$A + B = S$, där $A < 0$ och $B < 0$



I detta fall kan vi skriva spillvillkoret:

$$spill = (A < 0) (B < 0) (S \geq 0)$$

Komplementformer och moduloaritmetik

Generell definition:

$$X + X_{\beta k} = \beta^n \Rightarrow X_{\beta k} = (\beta^n - 1) - X + 1$$

Modulo(β^n)-aritmetik, talintervall $0.. \beta^n - 1$

Exempel: 2-komplement, 8-bitars tal

$$X + X_{2k} = 2^8 \Rightarrow X_{2k} = (2^8 - 1) - X + 1$$

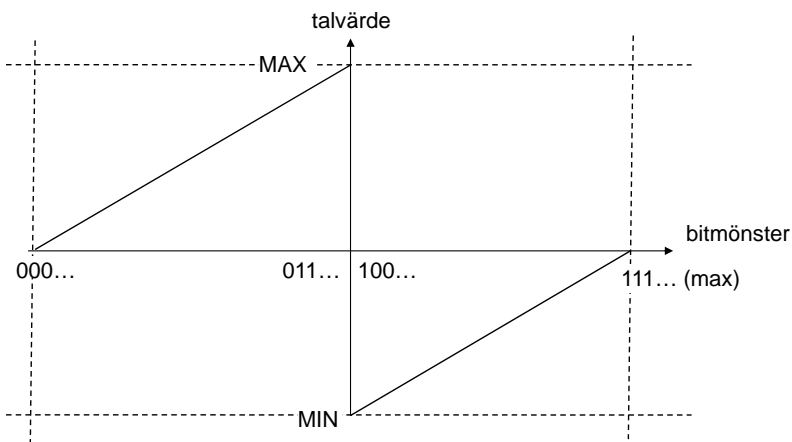
Modulo(256)-aritmetik, talintervall $0.. 255$

Talvärde vid tvåkomplementsform

$$N = -2^{n-1} x_{n-1} + \sum_{i=0}^{n-2} x_i 2^i$$

n=4 $x_3x_2x_1x_0$	N
0111	7
0110	6
0101	5
0100	4
0011	3
0010	2
0001	1
0000	0
1111	-1
1110	-2
1101	-3
1100	-4
1011	-5
1010	-6
1001	-7
1000	-8

Grafisk representation av tvåkomplementsform



Talvärdets avbildning är kontinuerlig bortsett från punkten med talvärdena MAX, MIN dvs, där talet (definitionsmsigt) byter tecken

Vi sammanfattar reglerna för flaggsättning vid addition av två n -bitars tal,
 $R = X + Y$,
 där index 0 betecknar den minst signifikanta biten och följaktligen index $n-1$ betecknar den mest signifikanta biten:

$$C = X_{n-1} \cdot Y_{n-1} + C_{n-1} \cdot (X_{n-1} + Y_{n-1})$$

$$V = (R_{n-1} \cdot \overline{X_{n-1}} \cdot \overline{Y_{n-1}}) + (\overline{R_{n-1}} \cdot X_{n-1} \cdot Y_{n-1})$$

$C =$	c_n	c_{n-1}	c_{n-2}	\dots	c_2	c_1	
	X_{n-1}	X_{n-2}	\dots	X_2	X_1	X_0	
+	Y_{n-1}	Y_{n-2}	\dots	Y_2	Y_1	Y_0	
	R_{n-1}	R_{n-2}	\dots	R_2	R_1	R_0	

Vi sammanfattar

Tvåkomplementsform är lämplig representation för binära negativa heltal.

Subtraktion utförs som addition av tvåkomplement

Spillindikator vid addition av naturliga tal $[0..N]$

"Carry" = c_n

Spillindikator vid subtraktion av naturliga tal $[0..N]$

"Borrow" (= c_n' då operationen utförs som addition)

Spillindikator vid addition/subtraktion av n -bitars heltal $[-M..N]$ med tvåkomplementsrepresentation:

"Overflow" = $s_{n-1}' x_{n-1} y_{n-1} + s_{n-1} x_{n-1}' y_{n-1}'$

Exempelsamling uppgift 4.8 e

Utför följande subtraktion ($R=X-Y$) av 8-bitars tal givna på tvåkomplementsform. Subtraktionerna ska utföras som addition av 2-komplement, dvs. $R = X + (2 \sim Y)$. Ange X, Y och R på decimal form, ange dessutom hur flaggorna N, V och Z påverkas av operationerna.

$X = 01000000 \quad Y = 10111111$

Vi löser detta på tavlan...