

Baza te Infomatikes

Leksioni 2

Elisa Reçi

Universiteti Luigj Gurakuqi
Fakulteti i Shkencave te Natyres
Departamenti i Matematikes dhe Informatikes
SHKODER

Veprime me Bit-et

- Veprime aritmetike mbi bit-et
 - Mbledhja, zbritja, shumezimi, pjestimi
- Instruksionet e manipulimit te biteve
 - Instruksionet logjike:
 - AND, OR, XOR, NOT
 - Instruksionet SHIFT :
 - SHL , SHR
 - Instruksionet Rotate
 - Rotate

Mbledhja e Bit-eve

- Mbledhja e bit-eve i permbahet ketyre rregullave:
 - $0+0 = 0$
 - $0+1 = 1$
 - $1+1 = 10$
 - $1+1+1=11$

- Prsh :

11 (ne mend)

1010

+ 1111

11001

Mbledhja e Bit-eve

$$\begin{array}{r} 111 \\ + 110 \\ \hline ? \end{array}$$

$$\begin{array}{r} 101 \\ + 111 \\ \hline ? \end{array}$$

$$\begin{array}{r} 111 \\ + 111 \\ \hline ? \end{array}$$

Zbritja e Bit-eve

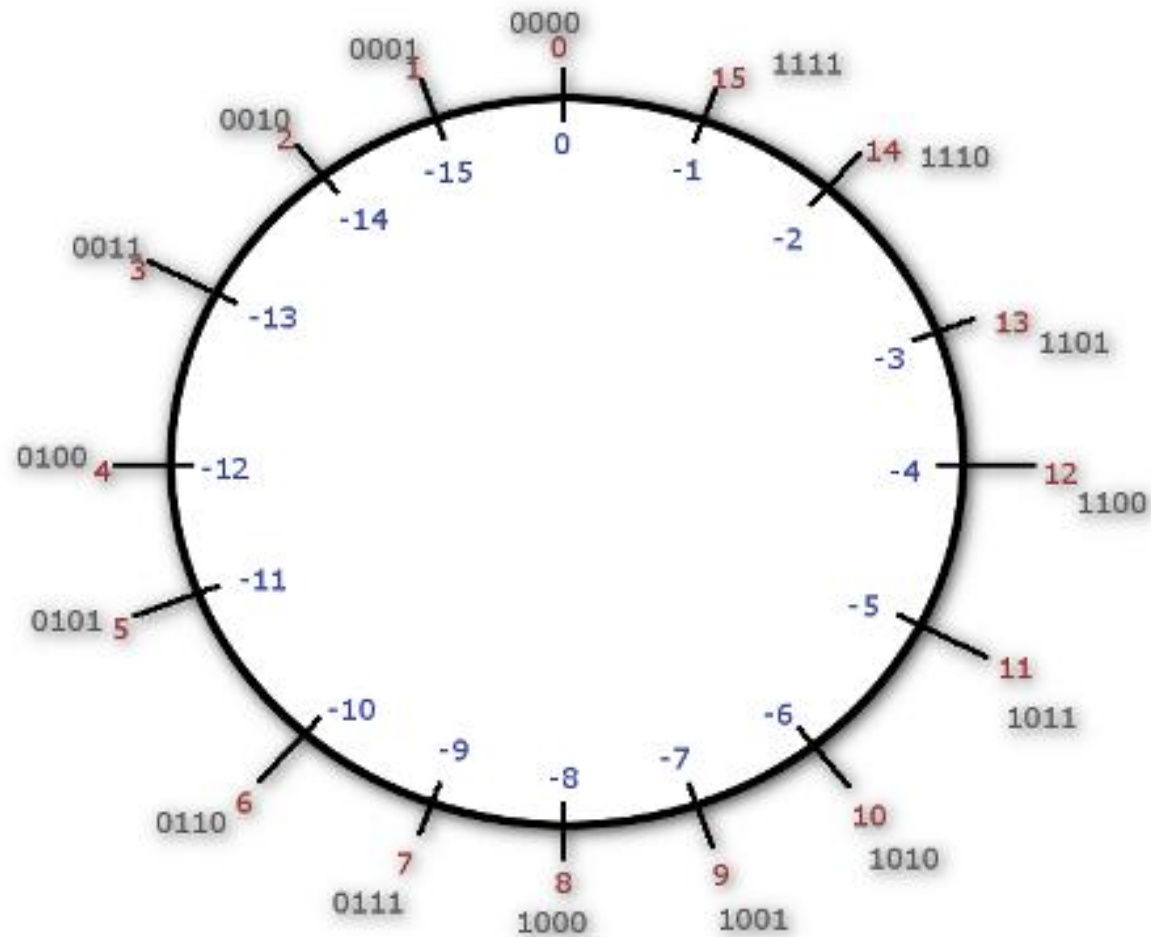
- Rregullat per zbritje jane :
 - $0 - 0 = 0$
 - $1 - 0 = 1$
 - $1 - 1 = 0$
 - $10 - 1 = 1$

$$\begin{array}{r} 10011 \\ - 1011 \\ \hline 1000 \end{array}$$

Zbritja e Bit-eve

- Praktike me e mire ne rastin kur nr l digit eshte l njejte ath rregullat e meposhtme ndihmojne:
 1. Konverto digit ne vleren e 2 (komplement 1- qe nenkupton 0 behet 1 dhe 1 behet 0)
 2. Shto vleren 1 komplementit 1
 3. Bej shumen e vleres se pare me te dyten
 4. Heq vleren e fundit te mbajtur mend gjate shumes
- **KUJDES-** nqs kemi nr fillon me 1 ath ai konsiderohet nr negativ prsh $1101 = -3$, kurse kur fillon me 0 ath eshte nr pozitiv prsh $0101 = 5$

Zbritja e Bit-eve



http://www.morkalork.com/mork/article/71/How_to_work_with_binary_subtraction.htm#.UIBp6G9mloQ

Zbritja e Bit-eve

- $0101 - 0011 = ?$
- **Rregulla 1** : 1st komplement i $0011 = 1100$
- **Rregulla 2** : $1100 + 0001 = 1101 \Rightarrow -3$
- **Rregulla 3** : $0101 + 1101 = 10010$
- **Rregulla 4** : ~~10010~~
- $5-3 \Rightarrow 5 + (-3) \Rightarrow 0101 + (1101) = 0010 = 2$

Zbritja e Bit-eve

$$\begin{array}{r} 11001 \\ - 1011 \\ \hline ? \end{array}$$

$$\begin{array}{r} 10001 \\ - 1100 \\ \hline ? \end{array}$$

$$\begin{array}{r} 11011 \\ - 1110 \\ \hline ? \end{array}$$

Prodhimi i bit-eve

- Prodhimi i bit-eve ka keto rregulla:

- $1 * 1 = 1$

- $1 * 0 = 0$

- $0 * 1 = 0$

$$\begin{array}{r} 101 \\ * 11 \\ \hline 101 \\ + 1010 \\ \hline 1111 \end{array}$$

Prodhimi i bit-eve

$$\begin{array}{r} 111 \\ * 110 \\ \hline ? \end{array}$$

$$\begin{array}{r} 101 \\ * 10 \\ \hline ? \end{array}$$

$$\begin{array}{r} 111 \\ * 11 \\ \hline ? \end{array}$$

Pjestimi i bit-eve

- Pjestime me bite ka te njejtën strategji si pjesimi i nr dhjetor.
- Prsh

$$\begin{array}{r} 10110 / 11 = 111 \\ - \quad 11 \downarrow \\ \hline 101 \\ - \quad 11 \downarrow \\ \hline 100 \\ - \quad 11 \\ \hline 001 \end{array}$$

Pjestimi i bit-eve

- $1010101/111 = ?$
- $101001/101 = ?$
- $001101/01 = ?$

- Instruksionet logjike

Logjik- AND

Bit 1	Bit 2	AND
0	0	0
0	1	0
1	0	0
1	1	1

Logjik- OR

Bit 1	Bit 2	OR
0	0	0
0	1	1
1	0	1
1	1	1

Logjik- XOR

Bit 1	Bit 2	XOR
0	0	0
0	1	1
1	0	1
1	1	0

Logjik- NOT

Bit	NOT
0	1
1	0

Veprime Logjike

1101100
AND 0101101

?

000100111
OR 110110001

?

1011011
XOR 1001010

?

NOT 0110101

?

Instruktionet logjike-Sintaksa

- AND Destinacion, Burim
- OR Destinacion, Burim
- XOR Destinacion, Burim
- NOT Destinacion

- Rezultati i veprimit ruhet ne Destinacion i cili duhet te jete nje regjister ose nje qelize kujtese
- Burimi mund te jete nje konstante, regjister ose qelize kujtese
- Destinacioni dhe burimi nuk mund te jene te dy qeliza kujtese
- Destinacioni dhe burimi duhet te jene te dy ne te njejten madhesi

MASK-a

- Modifikimi i biteve behet ne operandin Destinacion.
- Per kete ndertohet nje model biti Burim i cili njihet me emrin MASKE.
- Bitet e maskes zgjidhen ne menyre te tille qe bitet e dhena modifikohen ne menyren e duhur kur ekzekutohet nje forme e instruksionit :

Instruksion-logjik Destinacion , Mask

Mask-a

- Bitet e maskes zgjidhen duke u bazuar ne proprietite e AND, OR, XOR:
- Nese X perfaqeson nje bit (0 ose 1) ath:

$$\bullet X \text{ AND } 0 = 0 \quad X \text{ OR } 0 = X \quad X \text{ XOR } 0 = X$$

$$\bullet X \text{ AND } 1 = X \quad X \text{ OR } 1 = 1 \quad X \text{ XOR } 1 = \bar{X}$$

Mask-a AND

- Instruksioni AND mund te perdoret per te fshire bitet specifike te destinacionit duke lene te pandryshuar bitet e tjera.
- Nje maske bitesh me 0 fshin bitet korresponduese te destinacionit;
- Nje maske bitesh me 1 ruan bitet korresponduese te Destinacionit

	XXXXXXXX	Destinacion
AND	<u>00101011</u>	Mask
	00X0X0XX	

Mask-OR

- Instruksioni OR mund te perdoret per te vendosur bite specifike te Destinacionit duke lene te pandryshuara te tjerat.
- Nje maske bitesh me 1 vendos bitet korresponduese te Destinacionit
- Nje maske bitesh me 0 ruan bitet korresponduese te Destinacionit

	XXXXXXXX	Destinacion
OR	11101001	Maska
	<hr/>	
	111X1XX1	

Mask-XOR

- Instruksioni XOR mund te perdoret per te konvertuar bitet destinacion duke lene te pandryshuara te tjerat.
- Nje maske bitesh me 1 konverton bitet korresponduese te destinacionit
- Nje maske bitesh me 0 ruan bitet korresponduese te destinacionit.

	XXXXXXXX	Destinacion
XOR	<u>10000101</u>	Mask
	$\bar{X}X\bar{X}X\bar{X}X\bar{X}$	

Ushtrime....

XXXXXXXX
AND 10101101

?

XXXXXXXX
OR 10001111

?

XXXXXXXX
XOR 01111100

?

Ushtrime

- Te gjendet maska dhe veprimi logjik per :

$$\begin{array}{r} \text{XXXXXXXXXX} \\ ? \quad ? \\ \hline \overline{\text{XXXXXXXXXX}} \end{array}$$

$$\begin{array}{r} \text{XXXXXXXXXX} \\ ? \quad ? \\ \hline \text{XX0000XX} \end{array}$$

Instruktionet SHIFT

- Instrukcioni shift perdoret per te zhvendosur majta ose djathtas bitet qe ndodhen ne nje rregjister.
- => duhet me pare te diskutojme per shenjen e biteve (sign bit).
- Sign bit gjendet ne pozicionin me te majte (MSB-most significant bit)...
 - Tregon nqs nr eshte pozitiv apo negativ
 - Nqs vlera eshte 0 =>nr pozitiv, perndryshe negativ

Unsigned Right shift (“>>>”)

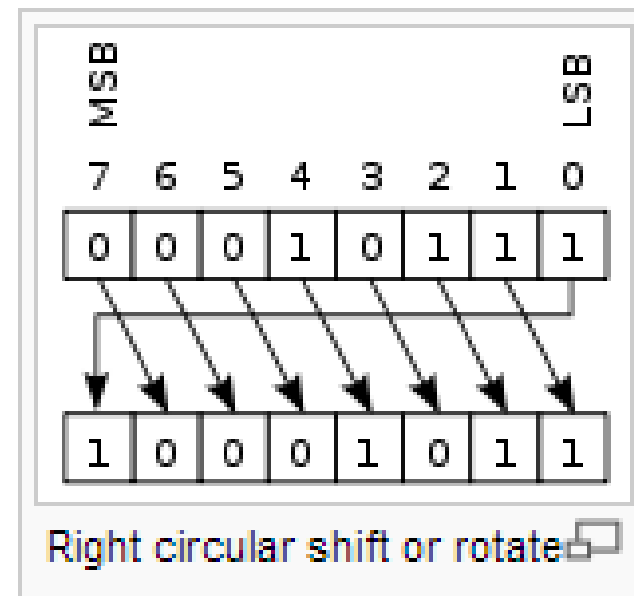
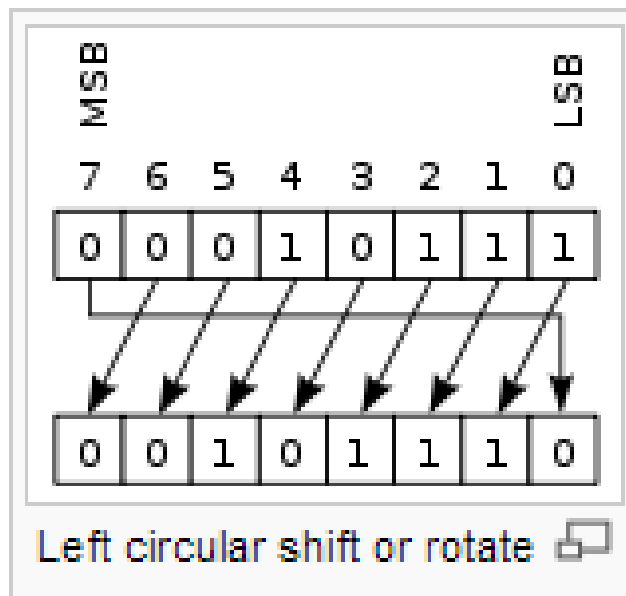
- Funkcionon si signed right shift.
- Ndryshon vetem ne faktin qe shton GJITHMON vetem 0 ne pozicionet bosh te leftmost.
 - => vepron gjithmon duke dhene nr pozitiv.
- Prsh 1110 edhe pse biti fillon me 1 qe tregon nr negativ.... Nqs bejme $1110 \ggg 2 = > 0011$
- Pra dy zerot e para te 0011 duhet te ishin zevendesuar me 11 nqs do kishin perdorur signed right shift

Ushtrime....

- $11001 \gg 1$?
- $0101001 \ll 3$?
- $1100011 \gg 2$?
- $0011101 \ll 1$?
- $1110101 \gg \gg 2$
- $11101010 \gg \gg 1$

InstrukSIONET ROTATE

- Bitet rrotullohen ne nje form te tille qe duket sikur fundet e anes se majte apo te djathte te regjistrit bashkohen



Instruksjonene ROTATE

- Rotullo me 2 bit djathtas 0010011 =>
1100100
- Rotullo me 1 bit majtas 100111 =>
001111

Ushtrime...

- Rrotullo majtas me 3 bit 10011101 ?
- Rrotullo majtas me 1 bit 01101 ?
- Rrotullo majtas me 2 bit 10110101 ?

- Rrotullo djathtas me 2 bit 0010010 ?
- Rrotullo djathtas me 1 bit 11011 ?
- Rrotullo djathtas me 3 bit 001110011 ?

