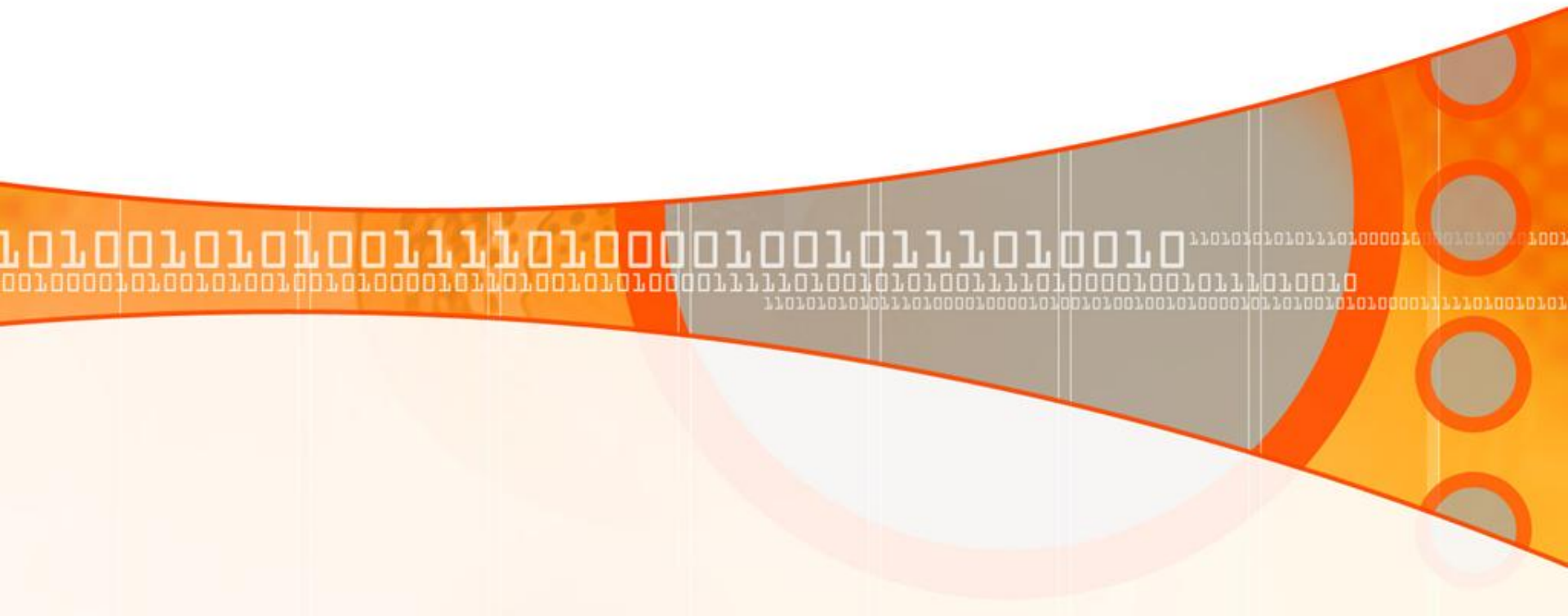


Организация на компютъра



Всички лекции от курса са подготвени по книгите на доц. д.н. Димитър Тянев

- [1]. <http://tyanev.com/> - On-line книги – ОРГАНИЗАЦИЯ НА КОМПЮТЪРА – книга 1
- [2]. <http://tyanev.com/> - On-line книги – ОРГАНИЗАЦИЯ НА КОМПЮТЪРА – упражнения книга 2;
- [3]. Димитър Тянев, ОРГАНИЗАЦИЯ НА КОМПЮТЪРА, том първи (ISBN 978-954-20-0412-7), Варна 2008г.
- [4]. Димитър Тянев, ОРГАНИЗАЦИЯ НА КОМПЮТЪРА – упражнения, ISBN 978-954-20-0258-0, Варна 2007г.

ЛИТЕРАТУРА

ОСНОВНА

- Димитър С. Тянев, Организация на компютъра, Изд. ТУ – Варна, 2008;
- <http://tyanев.com/> - On-line книги – ОРГАНИЗАЦИЯ НА КОМПЮТЪРА – книга [1]
- William Stallings, Computer Organization and Architecture, Seventh Edition, ISBN 0-13-185644-8



ЛИТЕРАТУРА

ДОПЪЛНИТЕЛНА

- Kai Hwang, Zhiwei Xu, Scalable Parallel Computing, McGraw Hill, 1998.
- Michael Flynn, Computer Architecture (Pipelined and Parallel Processor Design), Jones and Bartlett Publishers, 1995.
- Harry Jordan, Gita Alaghband, Fundamentals of Parallel Processing, Prentice Hall, 2003.
- Пламенка Боровска, Компютърни системи, Изд. "Сиела", 2005

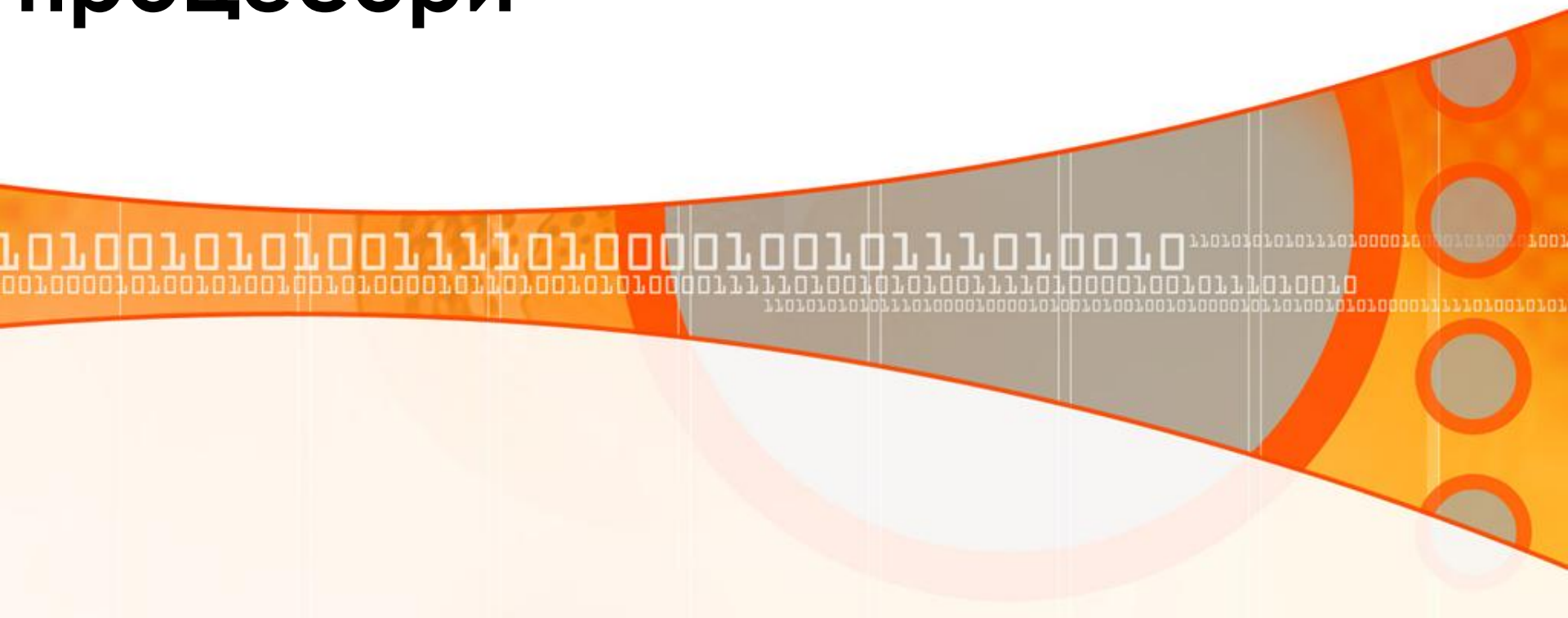


ЛИТЕРАТУРА

WEB SITES:

- High performance clusters www.topclusters.org
- Info centre for computer clusters www.buyya.com/cluster/
- Supercomputers Cray www.cray.com
- Supercomputers www.top500.org
- Advanced Clustering Technologies, Inc.
www.advancedclustering.com
- Encyclopedia Article About MPI www.thefreedictionary.com
- IEEE Task Force on Cluster Computing www.clustercomp.org
- IEEE Technical Committee on Scalable Computing
www.ieeetfcc.org
- Infiniband Trade Association
<http://www.infinibandta.org/home>

Лекция 1: Видове данни и тяхното представяне в цифровите процесори

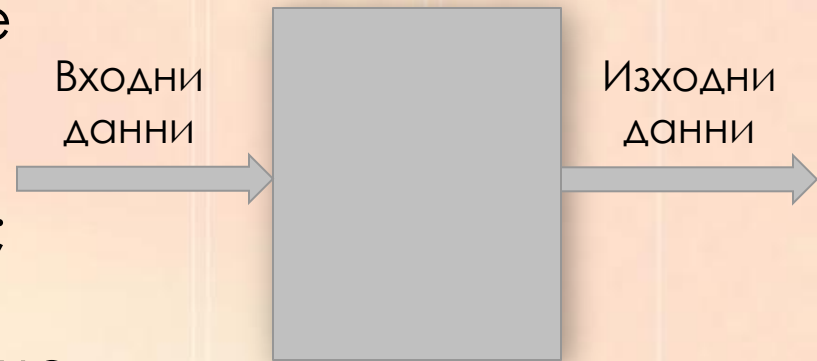


Съдържание

- Основни понятия и теоретични положения.
- Представяне на трите вида данни: логически, символни и числови.
- Организация на разрядната мрежа. Форми и формати – стандарти. Точност на представянето.
- Операция събиране.
- Машинни кодове за изобразяване на числата.
- Теореме за събиране на числа, изобразени в машинни кодове.
- Логически функции за откриване на препълването.
- Двоично-десетични машинни кодове.

Цифрова изчислителна машина

- Как ще представяме данните;
- Как ще ги въвеждаме и извеждаме;
- Как и кой ще преработва данните вместо нас;
- Как ще организираме автоматичното ѝ функциониране;
- Какви възможности тя ще ни предостави и какви ограничения ще ни наложи;
- Какви нейни параметри и характеристики ще ни интересуват;
- Как ще комуникираме с нея;
- Каква е теоретическата и техническата база за практическото ѝ реализиране.



Видове данни

- Данните, които обработват цифровите машини, се делят на три вида:
 - **Логически,**
 - **Символни,**
 - **Числови.**



Видове данни

- Човек използва различни **системи** за представяне на данните:
 - ✓ говорима;
 - ✓ писмена;
 - ✓ жестомимична и др.



Видове данни

- **Единственото** средство за представяне на данните в електронните изчислителни машини са **устойчивите** във времето природни явления.
- Две устойчиви състояния
- Елементите, които могат да съществуват дълготрайно (устойчиво) в такива състояния, се наричат **двузначни**.
- Устойчивите състояния се означават със символите **0** и **1**.
- Тези два символа с еднакъв успех се използват за означаване както на логически, така и на числови данни. Елементите, които ги реализират (разбирайте техническите средства), се наричат **логически** и в същото време **цифрови**.



Видове данни

- Двузначната система не може да се приложи буквално за представяне на данните от входно-изходния поток на интересувашщото ни техническо средство (изчислителна машина), а само в условията на еднозначен **двоичен код**.
- КОД** - изразно средство, съпоставимо еднозначно в права и обратна посока на съответен първичен елемент.

дешифриране (декодиране)



шифриране (кодиране)

- Кодът на даден първичен елемент, се формира в логическа схема, наречена **шифратор** (кодер). Възстановяването на първичния елемент по зададената му кодова комбинация се постига от логическа схема, наречена **дешифратор** (декодер).

Видове данни – логически данни

- **Логическите** данни представляват логическите константи "лъжа" и "истина".
- Кодират с двоичните цифри **0**="лъжа", **1**="истина".
- Изчислителните устройства оперират с логическите данни като изпълняват известните логически операции:
 - Логическо отрицание
НЕ; (NOT)
 - Логическо събиране (дизюнкция)
ИЛИ; (OR)
 - Логическо умножение (конюнкция)
И; (AND)
 - Логическа неравнозначност
(EOR; XOR)
 - Логическа равнозначност



Видове данни – логически данни

- Примери за използване на логически операции:

Пример 1 - маскиране:

$$Z = X \cap Y$$

$$\begin{array}{r} X = 00\mathbf{0011}01 \\ \cap \\ Y = \mathbf{00111100} \quad - \text{(маска)} \\ \hline Z = \mathbf{00001100} \end{array}$$

Пример 2 – логическа неравнозначност:

$$Z = X \oplus Y$$

$$\begin{array}{r} X = 0000\mathbf{11}01 \\ \oplus \\ Y = 00\mathbf{1111}00 \\ \hline Z = 00\mathbf{1100}01 \end{array}$$

Видове данни – СИМВОЛНИ ДАННИ

- За представяне на множеството **СИМВОЛИ** се построяват **КОДОВИ ТАБЛИЦИ**.
- **Кодови таблици:**
- **ISO-7** (КОИ-7) – 7-битов код, който стандартизира символната азбука на системи за телеобработка, предаване на данни по съобщителни канали и за нанасяне на данни върху перфолента.
- **ISO-8** (КОИ-8) – 8-битов код, който стандартизира символната азбука на изчислителните системи за връзка с периферни устройства.
- **EBCDIC** (ДКОИ) – 8-битов код, стандартизиран от IBM като код за използване в изчислителните системи.
- **ASCII** (*American Standard Code for Information Interchange*) – 6, 7 или 8-битов код, стандартизиран най-общо като код за обмен на информация.

Видове данни – СИМВОЛНИ ДАННИ

- Стандартизираното множество от символи, чийто обем е 256 символа, се дели най-общо на две части – подмножество **графични** символи и подмножество **управляващи** (служебни) символи.



ВИДОВЕ ДАННИ – СИМВОЛНИ ДАННИ

- ASCII кодова таблица – първа част

Decimal code	Hex-code	Symbol
000	00	NUL
001	01	SOH
002	02	STX
...
010	0A	LF
...
013	0D	CR
...
032	20	space
033	21	!
...
043	2B	+
...
048	30	0
049	31	1
...
057	39	9

Decimal code	Hex-code	Symbol
058	3A	:
...
064	40	@
065	41	A
066	42	B
...
090	5A	Z
091	5B	[
092	5C	\
093	5D]
...
096	60	'
097	61	a
098	62	b
...
122	7A	z
123	7B	{

Видове данни – СИМВОЛНИ ДАНИИ

- ASCII кодова таблица – втора част

Decimal code	Hex-code	Symbol
128	80	A
129	81	Б
130	82	B
...
158	9E	Ю
159	9F	Я
160	A0	a
161	A1	б
...
190	BE	ю
191	BF	я
192	C0	Ł
193	C1	ł
...
203	CB	ƒ
204	CC	ƒ
205	CD	=

Decimal code	Hex-code	Symbol
213	D5	№
214	D6	§
...
224	E0	α
225	E1	β
226	E2	Γ
...
234	EA	Ω
235	EB	δ
236	EC	¥
237	ED	Φ
238	EE	ε
...
241	F1	±
...
238	EE	ε
255	FF	blank

Видове данни – СИМВОЛНИ ДАННИ

- Последователностите от символи се наричат *символни низове* или просто *низове*.

Пример: Кодовата последователност или символният низ:

87 A4 B0 A0 A2 A5 A9 20 21

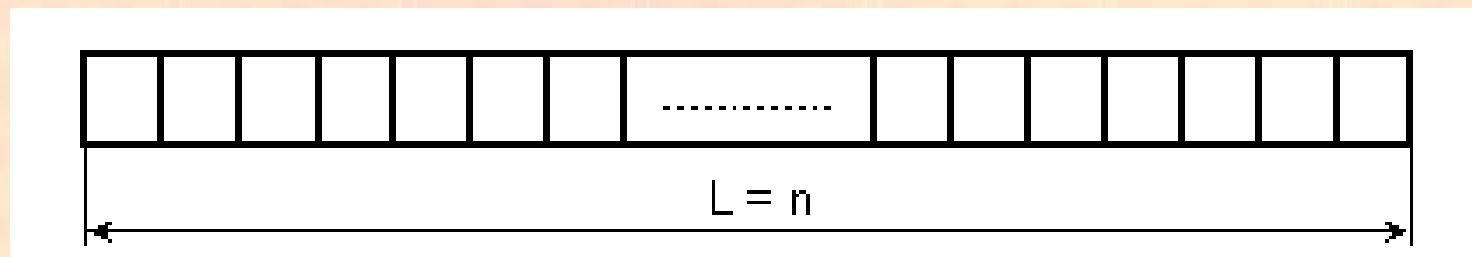
означава
съобщението:
(**Здравей !**)

З д р а в е й !



Видове данни – Числови данни

- Разрядна мрежа (PM)



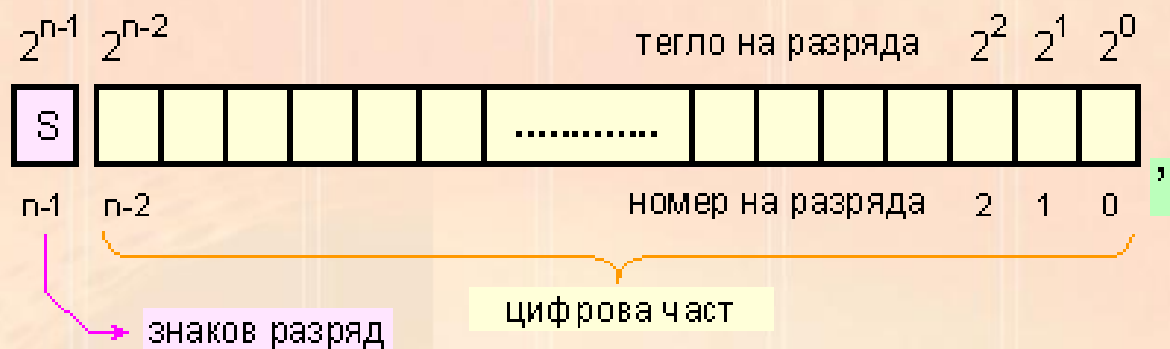
Дължината на PM = броя на разрядите.

Онази част от разрядната мрежа, в която може да се помести **една цифра** на числото, се определя като **един разряд**.

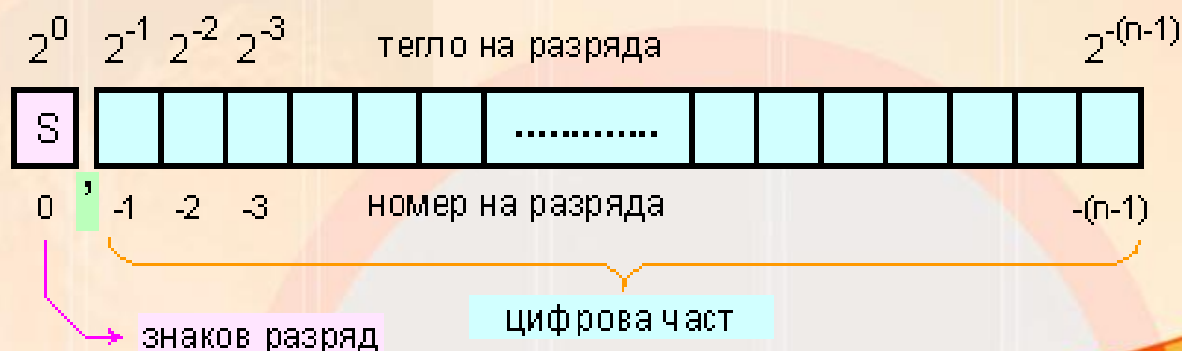
Видове данни – Числови данни

- Разрядната мрежа в изчислителните устройства най-често е така организирана, че положението на запетаята на числата спрямо нея е *фиксирано и неизменно* – ОТДЯСНО ИЛИ ОТЛЯВО.

РМ с ДФЗ



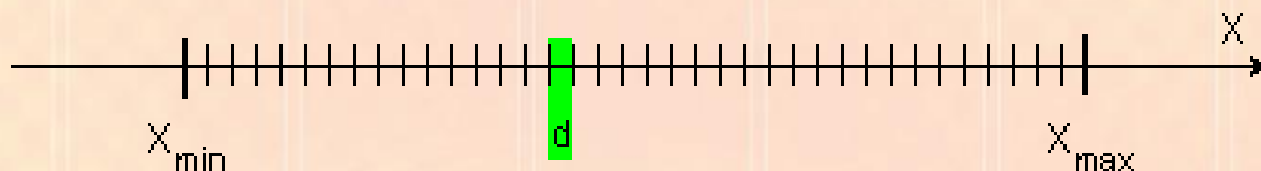
РМ с ЛФЗ



- Число, представено в разрядна мрежа с показаната по-горе структура, е във **форма с фиксирана запетая**.

Видове данни – Числови данни

- Характеристики на разрядната мрежа:
 - Най-голямо и най-малко представими числа;
 - Диапазон на представимите числа: $[X_{\min} \div X_{\max}]$



- В общия случай възможностите на дадена разрядна мрежа с дължина n q -ични разряда са ограничени от числата $X_{\min} = 000 \dots 0 = 0$ и $X_{\max} = rrr \dots r = q^n - 1$ и диапазонът на представимите числа за нея се определя така: $[0 \div q^n - 1]$.
 r е цифрата на най-голямото елементарно количество в съответната бройна система – например 1 за двоичната, 9 за десетичната или F за шестнадесетичната.

Видове данни – Числови данни

- Характеристики на разрядната мрежа:
 - За двоични числа без знак, представяни в разрядната мрежа с дължина $L=n[b]$, диапазонът се ограничава от числата $(000...0)$ и $(111...1)$, т.е. $[0 \div 2^n - 1]$. Този диапазон е в областта на целите числа, тъй като, когато те са без знак, най-често се интерпретират като представени с дясно фиксирана запетая.

Пример за 32-битова разрядна мрежа:

$$X_{\max} = 2^{32} - 1 = 4294967295$$

Видове данни – Числови данни

- Характеристики на разрядната мрежа:

- За числа със знак

- ✓ При дясно фиксирана запетая:

$$X_{\min} = -rrr\dots r, = -(q^{n-1} - 1) \quad X_{\max} = +rrr\dots r, = +(q^{n-1} - 1)$$

- ✓ За двоична бройна система:

$$X_{\min} = -111\dots 1, = -(2^{n-1} - 1) \quad X_{\max} = +111\dots 1, = +(2^{n-1} - 1)$$

- ✓ Модулът на най-голямото двоично число, възможно за цифровата част на разрядната мрежа с дясно фиксирана запетая е

$$(2^{n-1} - 1) = 2147483647$$

Видове данни – Числови данни

- Характеристики на разрядната мрежа:

- За числа със знак

- ✓ При ляво фиксирана запетая:

$$X_{\min} = -0,rrr\dots r = -(1 - q^{-(n-1)}) \quad X_{\max} = +0,rrr\dots r = +(1 - q^{-(n-1)})$$

- ✓ За двоична бройна система:

$$[-(1 - 2^{n-1}) \div +(1 - 2^{n-1})].$$

Видове данни – Числови данни

- Характеристики на разрядната мрежа:
 - Дискрет (равномерна стъпка) на разрядната мрежа – константа – най-малкото по модул представимо число, различно от нула.
 - ✓ При дясно фиксирана запетая $d = 2^0 = 1$.
 - ✓ При ляво фиксирана запетая $d = 2^{-(n-1)}$.
- т. е. дискретът и в двата случая по същество е теглото на най-младшия разряд в разрядната мрежа.
- ✓ В общия случай:

$$d = \begin{cases} q^0 = 1, & \text{за РМ с ДФЗ;} \\ q^{-(n-1)}, & \text{за РМ с ЛФЗ.} \end{cases}$$

Видове данни – Числови данни

- **Характеристики** на разрядната мрежа:
 - Представимите числа са разпределени *равномерно* в диапазона със стъпка равна на дискрета. Вътре в този диапазон има безкрайно много други числа, които не могат да бъдат представени с определения брой разряди. Те се наричат *непредставими*, а в случай на нужда могат да бъдат заместени от най-близкото до тях представимо число. Пример: 5,786 може да се представи с числото 6, ако изискванията за точност се удовлетворяват.

Видове данни – Числови данни

- **Характеристики** на разрядната мрежа:
 - Всички числа, които са разположени върху числовата ос във от диапазона на представимите числа – вляво или вдясно от него, са също *непредставими* числа. Те *не могат* да бъдат заместени от никое представимо число, тъй като изискванията за точност не позволяват това. За тези числа обикновено се казва, че **препълват** отляво или отдясно разрядната мрежа.



Видове данни – Числови данни

- Характеристики на разрядната мрежа:

➤ *Точност* на представимите числа

Разрядната мрежа, поради своята крайност, отрязва младшите значещи цифри на числото, попадащи вдясно от нея. Отхвърлената част от числото представлява всъщност *абсолютната грешка*, с която се плаща, за да стане едно число представимо.

Абсолютната грешка клони към теглото на най-младшата позиция $2^0 = q^0 = 1$ в РМ с ДФЗ и $2^{-(n-1)}$, съответно $q^{-(n-1)}$ в РМ с ЛФЗ.

Видове данни – Числови данни

- Характеристики на разрядната мрежа:
 - *Точност* на представимите числа. Обобщение:
Абсолютната грешка на двоични (q-ични) разрядни мрежи с фиксирана запетая се изменя в диапазона

$$\Delta_{\text{ДФЗ}} = [0 \div 1) ;$$

$$\Delta_{\text{ДФЗ}} = [0 \div 2^{-(n-1)}) , \text{ или } [0 \div q^{-(n-1)}) .$$

Видове данни – Числови данни

- Характеристики на разрядната мрежа:
- С цел да повишат универсалността на своите процесори, повечето производители поддържат няколко вътрешносхемни формати на числовите данни. Най-често стандартите определят за числата с фиксирана запетая следните формати:
- Байт - 8[b] ; *Byte Integer* (BI) ;
- Дума - 16[b] ; *Word Integer* (WI) ;
- Двойна дума - 32[b] ; *Double Word Integer* (DI) ;
- Дълга дума - 64[b] ; *Long Integer* (LI) .

ВИДОВЕ ДАННИ – ЧИСЛОВИ ДАННИ

- Числа с плаваща запетая - **мултипликативен вид** на числата

Числото се представя като произведение от значещата му част и нейния мащаб.

Пример:

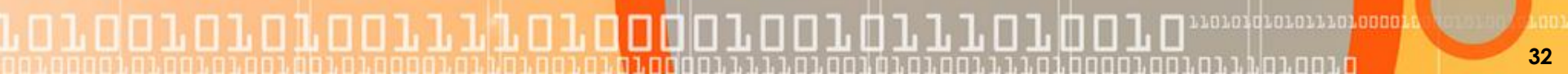
$$X1 = 0,00000000000035060 \quad \text{или} \quad X2 = -1207400000000000,0$$

$$X1 = 0,00000000000035060 = 3506,0 \cdot 10^{-14} ;$$

$$X2 = -1207400000000000,0 = -12074,0 \cdot 10^{+10} .$$

3506 или 12074 е значеща част, а числата 10^{-14} или 10^{+10} – *мащабен коефициент* или просто *мащаб*.

Мащабът е число, кратно на основата на бройната система (в примера $q=10$).



ВИДОВЕ ДАННИ – ЧИСЛОВИ ДАННИ

- **Числа с плаваща запетая - мултипликативен вид** числата. Числото се представя като произведение от значещата му част и нейния мащаб.

Пример:

$$X1 = 0,00000000000035060 \quad \text{или} \quad X2 = -1207400000000000,$$

$$X1 = 0,00000000000035060 = 3506,0 \cdot 10^{-14} ;$$

$$X2 = -1207400000000000,0 = -12074,0 \cdot 10^{+10} .$$

3506 или 12074 е значеща част, а числата 10^{-14} или 10^{+10} – мащабен коефициент или просто мащаб.

За мащаб - число, кратно на основата на бройната система (в примера $q=10$).

Така не се променят цифрите в значещата част на числото! Единствено при този вид на мащабния коефициент, при преминаване от естествен към мултипликативен вид на числото, се съхраняват цифрите на значещата му част.



Видове данни – Числови данни

- Числа с плаваща запетая - **мултипликативен вид** числата. Представянето на числата в мултипликативен вид се нарича още *представяне във **форма с плаваща запетая*** (ПЗ).
- Мащабът е степенна функция, чийто степенен показател има характер на *цяло число със знак*.
- Степенният показател се нарича **порядък** и показва на колко позиции е изместен разделителят (запетаята) в значещата част (наричана **мантиса**), спрямо позицията му (й) в естествения вид на числото.
- Знакът на порядъка показва посоката на числовата ос, в която следва да се премести разделителят (запетаята), за да се възвърне естественият вид на числото.

Видове данни – Числови данни

- Числа с плаваща запетая

Представяне в разрядната мрежа на:

знак на числото; **мантиса на числото**; знак на порядъка; порядък.

Мантисата се представя като **нормализирано** число **с ляво фиксирана** запетая.

Ако полето на мантисата има дължина m разряда и ако то е организирано с ляво фиксирана запетая, тогава мантисата, представена в него, е нормализирана, ако удовлетворява неравенството:

$$q^{-1} \leq |M_x| < 1$$

Най-голямата нормализирана мантиса е числото:

$$+ (1 - q^{-(m-1)})$$



ВИДОВЕ ДАННИ – ЧИСЛОВИ ДАННИ

- Числа с плаваща запетая

Представяне в разрядната мрежа на:

знак на числото; мантиса на числото; знак на порядъка; **порядък**.

Порядъкът се представя като **цяло** число **със знак, съдържано фиксирана** запетая.

Максималният порядък, който може да бъде представен в k разряда, е $+(q^{k-1} - 1)$.

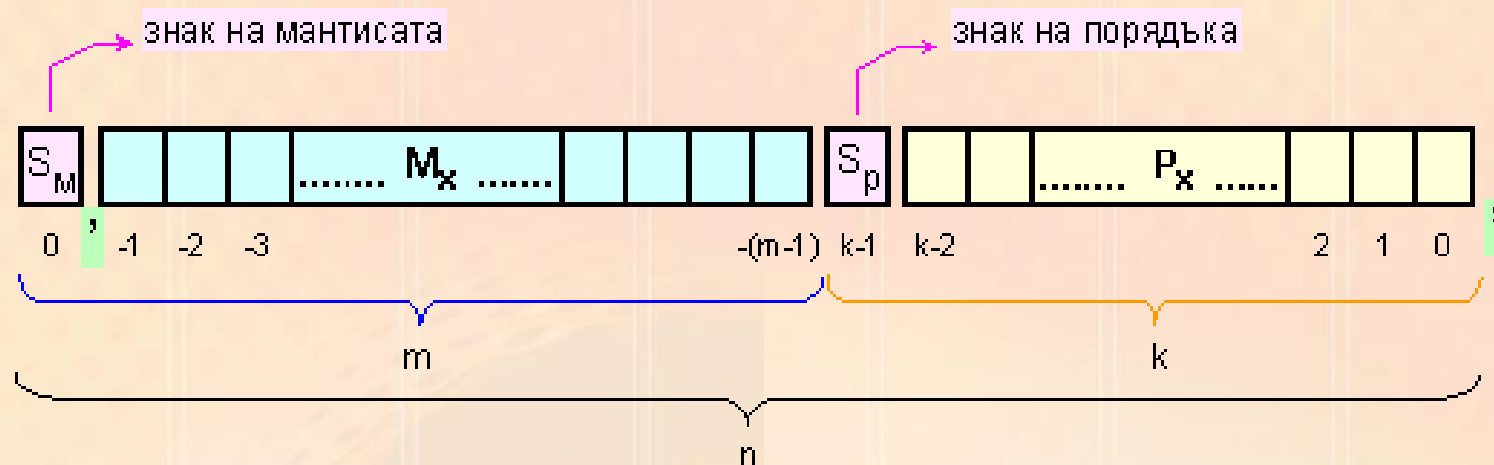
Максималното представимо число във форма с плаваща запетая, изобразено в бройна система с основа q , е:

$$X_{\max} = +(1 - q^{-(m-1)}) \cdot q^{+(q^{k-1} - 1)}.$$

ВИДОВЕ ДАННИ – ЧИСЛОВИ ДАННИ

- Числа с плаваща запетая

- Структура на разрядна мрежа с плаваща запетая



- Диапазон на представимите числа:

$$\text{от } -(1 - q^{-(m-1)}) \cdot q^{+(q^{k-1} - 1)} \text{ до } +(1 - q^{-(m-1)}) \cdot q^{+(q^{k-1} - 1)}$$

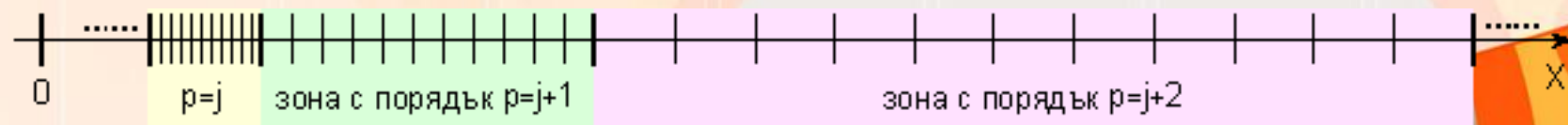
ВИДОВЕ ДАННИ – ЧИСЛОВИ ДАННИ

- Числа с плаваща запетая

Ако искаме да разширим диапазона на представимите числа, следва да удължим полето за представяне на порядъка. Това може да стане само за сметка на полето на мантисата.

Мультипликативният вид на записа на числата води до тяхното *неравномерно разпределение върху числовата ос*.

Диапазонът на представимите числа се разделя на толкова зони, наричани *бинади* за двоичната бройна система (или *декади* за десетична система), колкото са различните стойности на порядъка.



ВИДОВЕ ДАННИ – ЧИСЛОВИ ДАННИ

- Числа с плаваща запетая
- Порядъкът се представя неявно чрез друга величина, наречена *характеристика*, чиито стойности са числа без знак. Превръщането на порядъка в число без знак се постига чрез изместване на неговия диапазон в положителната посока на числовата ос.

Новополучената величина се нарича **изместен порядък** или **характеристика** и се определя така:

$$H = p + D,$$

където с D е означена константа, която се нарича **изместване**.

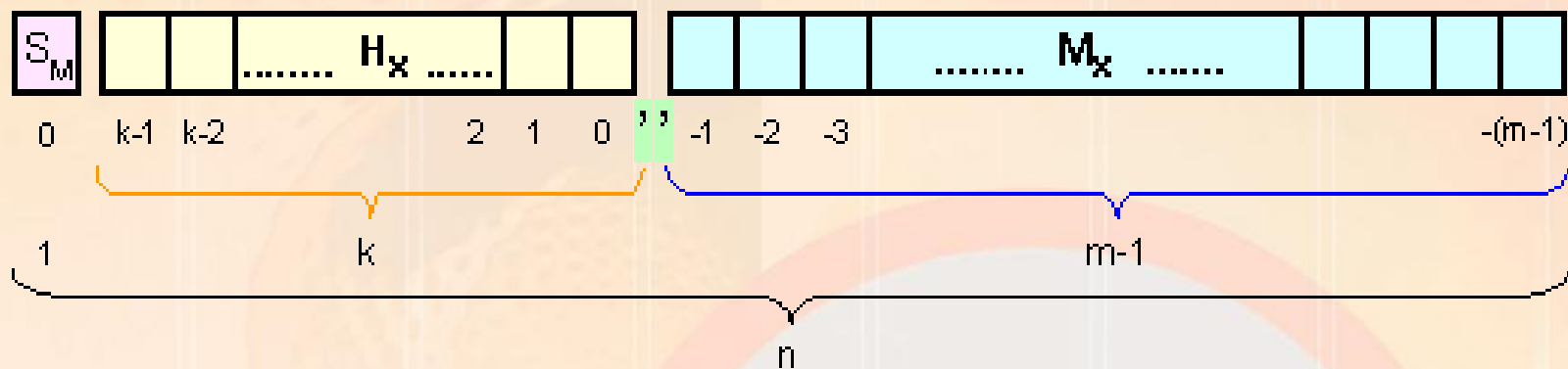
ВИДОВЕ ДАННИ – ЧИСЛОВИ ДАННИ

- Числа с плаваща запетая
- Изместването D се дефинира като най-малкото число, с помощта на което се постига неравенството $H \geq 0$, за всяко възможно p . Очевидно е, че стойността на изместването D зависи от конструктивните параметри на разрядната мрежа и по-точно от дължината на полето на порядъка, откъдето следва, че неговата стойност е равна на числото, представляващо теглото на знаковия разряд в полето на порядъка, т.е. $D = 2^{k-1}$



Видове данни – Числови данни

- Числа с плаваща запетая
- Премахвайки знака на порядъка, структурните елементи на числото, които следва да се представят в разрядната мрежа, остават три. Тяхното подреждане в разрядната мрежа най-често е по следния начин:



ВИДОВЕ ДАННИ – ЧИСЛОВИ ДАННИ

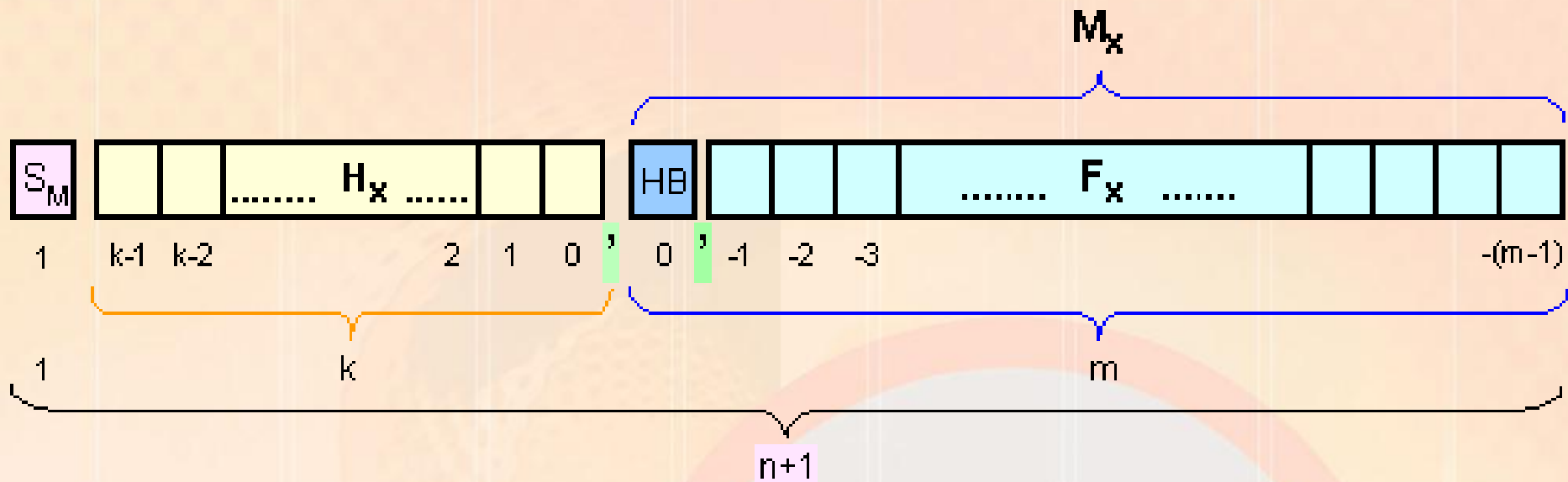
- Числа с плаваща запетая
- **"скрит бит" на мантисата** (NB – *Hidden Bit*).
- Скрытият бит на мантисата присъства само по време на изчисленията. При запомняне на числата в паметта скрытият бит не се съхранява.
- Едно число във форма с плаваща запетая, с нормализарана мантиса, с изместен порядък и с техника на скрытия бит, се определя така:

$$X = \pm 1, F_x \cdot 2^{\pm[H_x - (D-1)]}$$



Видове данни – числови данни

- Числа с плаваща запетая
- Структура на РМ с ПЗ, с изместен порядък и скрит бит

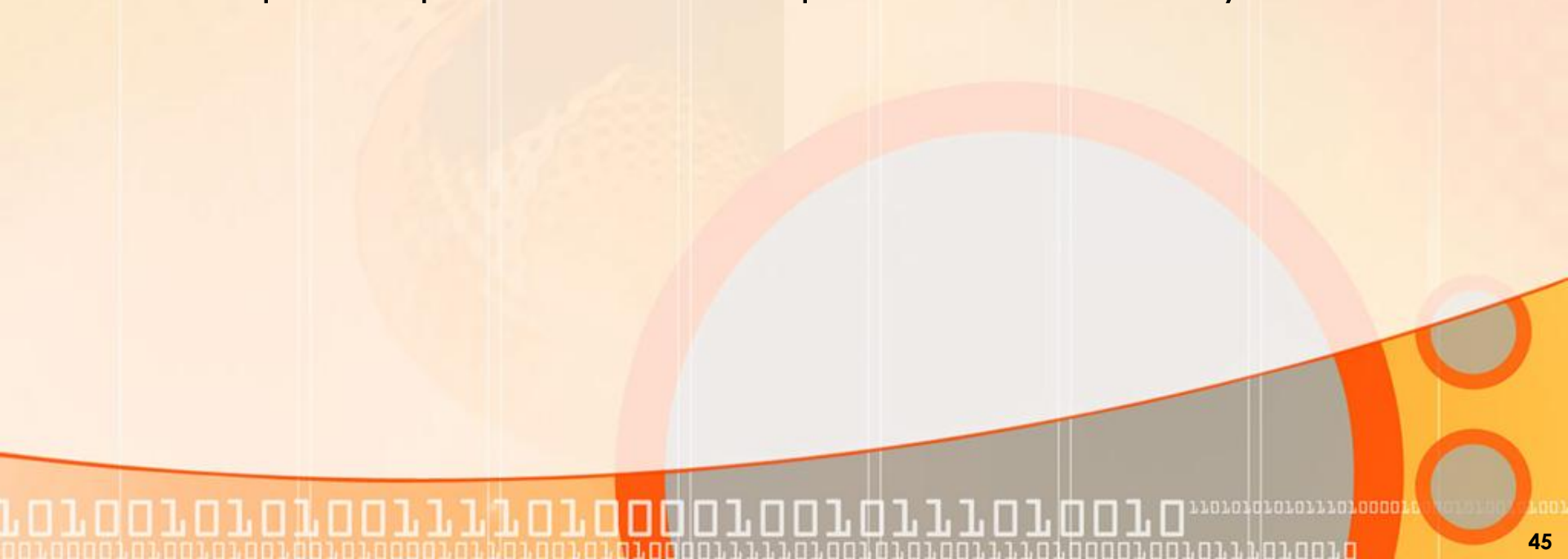


ВИДОВЕ ДАННИ – ЧИСЛОВИ ДАННИ

- Числа с плаваща запетая
- Структурата на разрядната мрежа е стандартизирана от американската организация IEEE (*Institution of Electrical and Electronic Engineers*). Стандартът е известен в литературата под номер 754 (IEEE 754 и по-новия 854) и се спазва от всички производители на компютърна техника. Главното предназначение на стандарта е да осигури преносимостта на програмите.
- Стандартът определя два формата за двоични числа – формат "единична точност" (SP – *Single Precision*) с дължина 32[b] и "двойна точност" (DP – *Double Precision*) с дължина 64[b]. Във формат SP са определени 8[b] за представяне на характеристиката, а във формат DP – 11[b].

ВИДОВЕ ДАННИ – ЧИСЛОВИ ДАННИ

- Числа с плаваща запетая
- Стандартът предоставя допълнителна свобода на производителите като определя в допълнение на основните формати още два, неречени "разширени" – (SEP – *Single Extended Precision*) и (DEP – *Double Extended Precision*).
- Дължините на полетата за тези разширени формати не са фиксирани, но са ограничени отдолу.



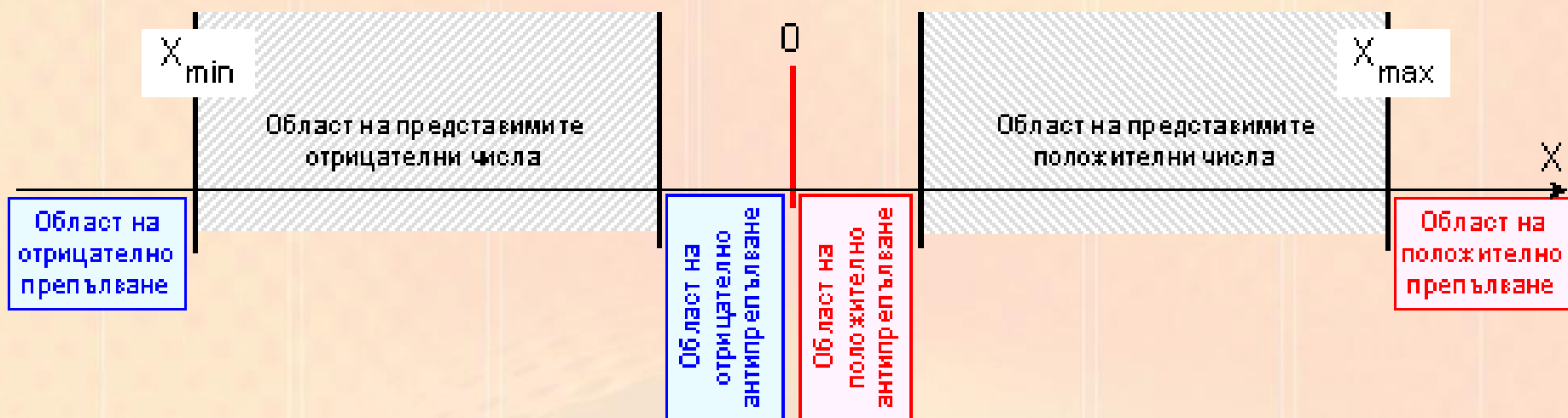
ВИДОВЕ ДАННИ – ЧИСЛОВИ ДАННИ

- Числа с плаваща запетая

	Single	Single Extended	Double	Double Extended
Дължина на разрядната мрежа	32	≥ 43	64	≥ 79
Дължина на полето H	8	≥ 11	11	≥ 15
Дължина на полето F	23	≥ 31	52	≥ 63
Стойност на изместването D	127	неопределена	1023	неопределена
Максимален порядък	127	≥ 1023	1023	≥ 16383
Минимален порядък	-126	≤ -1022	-1022	≤ -16382
Основен диапазон на числата	$10^{-38} - 10^{+38}$	неопределена	$10^{-308} - 10^{+308}$	неопределена

Видове данни – Числови данни

- Числа с плаваща запетая



- Област на отрицателно антипрепълване:
от $-0,5 \cdot 2^{[0-(D-1)]}$ до $0 = 0,0 \cdot 2^{[0-(D-1)]}$;
- Област на положително антипрепълване:
от $0 = 0,0 \cdot 2^{[0-(D-1)]}$ до $0 = 0,0 \cdot 2^{[0-(D-1)]}$.

Представяне на числа със знак

- Знаците на числата се кодират така:

“+” - с **0** и “-” с **1**

- **Прав код** на числата:

- За числа с дясно фиксирана запетая:

$$[X]_{\text{ПК}} = \begin{cases} X, & \text{ако } X \geq 0; \\ 2^{n-1} + |X|, & \text{ако } X \leq 0. \end{cases}$$

- За числа с ляво фиксирана запетая:

$$[X]_{\text{ПК}} = \begin{cases} X, & \text{ако } X \geq 0; \\ 1 + |X|, & \text{ако } X \leq 0. \end{cases}$$

- Така правият код на едно число (като съдържание на разрядната мрежа) ще представлява двоична комбинация от вида:

$$[X]_{\text{ПК}} = \begin{cases} 0 \text{ } 00 \dots 0 \text{ } xxx \dots xx, & \text{ако } X \geq 0; \\ 1 \text{ } 00 \dots 0 \text{ } xxx \dots xx, & \text{ако } X \leq 0. \end{cases}$$



Представяне на числа със знак

- **Прав код** на числата:

Пример:

+73 – **0**1001001

- 73 - **1**1001001



Представяне на числа със знак

- **Обратен код** на числата

$$[X]_{\text{ок}} = \begin{cases} X, & \text{ако } X \geq 0; \\ Q - |X|, & \text{ако } X \leq 0. \end{cases}$$

$$Q = q^n - d$$

q – основа на бройната система; d – единица на разрядната мрежа (за двоична бройна система $d = 1$)

- Числото нула има двойно представяне в обратен код:

$$[+0]_{\text{ок}} = 0\ 000\ \dots 00$$

$$[-0]_{\text{ок}} = 1\ 111\ \dots 11$$

- Може да се запише още, че:

$$[X]_{\text{ок}} = \begin{cases} 0xxx\dots xx, & \text{ако } X \geq 0; \\ 1\overline{xxx}\dots\overline{xx}, & \text{ако } X \leq 0. \end{cases}$$

Представяне на числа със знак

- Обратен код на числата

+ 73 – **0**1001001

- 73 - **1**0110110



Представяне на числа със знак

- **Допълнителен код** на числата:
- За (двоични) числа с дясно фиксирана запетая:

$$[X]_{DK} = \begin{cases} X, & \text{ако } X \geq 0; \\ 2^n - |X|, & \text{ако } X < 0. \end{cases}$$

- За числа с ляво фиксирана запетая:

$$[X]_{DK} = \begin{cases} X, & \text{ако } X \geq 0; \\ 2 - |X|, & \text{ако } X < 0. \end{cases}$$

Представяне на числа със знак

- **Допълнителен код** на числата (двоичните)
 - Диапазон на представимите числа:
 $-(2^{n-1}) \div + (2^{n-1} - 1)$
 - Числото нула съответства на комбинация нула “000...00”;
 - В допълнителен код старшите незначещи цифри на числото, с които се допълва цифровата част на разрядната мрежа, **съвпадат** с цифрата на знака.
 - Знакът на числото, получено в резултат на аритметична операция, се получава **автоматично**. Резултатът също е число в допълнителен код.

Представяне на числа със знак

- **Допълнителен код** на числата (двоичните)

+73 – **0**1001001

- 73 - **1**0110111



Представяне на числа със знак

- Връзка между обратен и допълнителен код

$$[X]_{\text{ДК}} = \begin{cases} [X]_{\text{ПК}} & \text{ако } X \geq 0; \\ [X]_{\text{ОК}} + 1 & \text{ако } X < 0. \end{cases}$$

- Други зависимости:

$$[[[X]_{\text{ОК}}]_{\text{ОК}}]_{\text{ОК}} = [X]_{\text{ОК}}; \quad [[[[X]_{\text{ОК}}]_{\text{ОК}}]_{\text{ОК}}]_{\text{ОК}} = [X]_{\text{ПК}}.$$

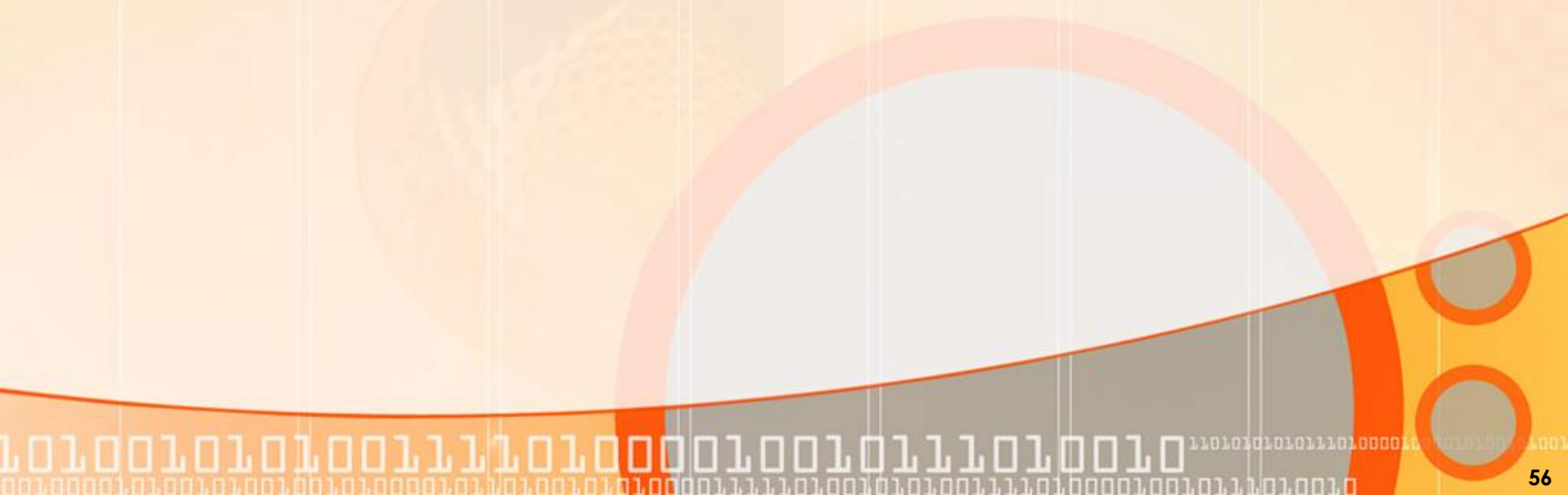


Представяне на числа със знак

- **Модифицирани кодове** на числата

Знаковите разряди са **два**.

Анализът на знаковите разряди на резултата позволява лесно да се определи наличието на пренос или препълване при извършването на аритметични операции.



Основи на аритметиката на двоични числа

- Операция **събиране**

Две многоразрядни числа без знак се събират ($X+Y=Z$) по определение, поразрядно, започвайки от младшите цифри на операндите X , Y , последователно към старшите, въз основа на полиномния им вид.

$$\begin{aligned} & x_{n-1} \cdot q^{n-1} + x_{n-2} \cdot q^{n-2} + \dots + x_3 \cdot q^3 + x_2 \cdot q^2 + x_1 \cdot q^1 + x_0 \cdot q^0 + \\ & + y_{n-1} \cdot q^{n-1} + y_{n-2} \cdot q^{n-2} + \dots + y_3 \cdot q^3 + y_2 \cdot q^2 + y_1 \cdot q^1 + y_0 \cdot q^0 = \sum_{i=0}^{n-1} (x_i + y_i) \cdot q^i. \end{aligned}$$

Основи на аритметиката на двоични числа

- Операция **събиране**

Поразрядното сумиране се представя със следната система уравнения:

$$x_i + y_i + p_{i-1} = \Sigma ;$$

$$p_i = \begin{cases} 0, & \text{ако } \Sigma < q; \\ 1, & \text{ако } \Sigma \geq q; \end{cases}$$

$$z_i = \Sigma - qp_i \quad i = \overline{0, (n-1)}.$$

n е дължината на разрядната мрежа.



Основи на аритметиката на двоични числа

- Операция **събиране**

- **Пренос и препълване** при сумиране:

$$\begin{array}{r} x_{n-1}x_{n-2} \dots x_i \dots x_3x_2x_1x_0, \\ + \quad y_{n-1}y_{n-2} \dots y_i \dots y_3y_2y_1y_0, \\ \hline 1 \quad z_{n-1}z_{n-2} \dots z_i \dots z_3z_2z_1z_0, \end{array}$$

Резултатът при препълване се удължава отляво, но само с една позиция, в която стои цифрата 1 (независимо каква е основата q на използваната бройна система).

- При събиране на числа **без знак**

$$V \equiv p_{n-1} = C$$

Основи на аритметиката на двоични числа

- Операция **събиране**
 - **Пренос и препълване** при събиране на числа **със знак**
 - ✓ При събиране в **допълнителен код**

$$V_1 = (x_{n-1} \wedge y_{n-1} \wedge \overline{z_{n-1}}) \vee (\overline{x_{n-1}} \wedge \overline{y_{n-1}} \wedge z_{n-1}) \quad \text{ИЛИ} \quad V_2 = p_{n-1} \oplus p_{n-2}$$

$$\begin{array}{r} \phantom{[-12]_{\text{ДК}} = } \phantom{\leftarrow \text{преноси}} \\ \phantom{[-12]_{\text{ДК}} = } 1 \phantom{\leftarrow \text{преноси}} \\ [-12]_{\text{ДК}} = 0100 \\ \phantom{[-12]_{\text{ДК}} = } + \\ [-13]_{\text{ДК}} = 0011 \\ \hline \phantom{[-12]_{\text{ДК}} = } 0111 \end{array}$$

$$\begin{array}{r} \phantom{[12]_{\text{ДК}} = } \phantom{\leftarrow \text{преноси}} \\ \phantom{[12]_{\text{ДК}} = } 0 \phantom{\leftarrow \text{преноси}} \\ [12]_{\text{ДК}} = 1100 \\ \phantom{[12]_{\text{ДК}} = } + \\ [13]_{\text{ДК}} = 1101 \\ \hline \phantom{[12]_{\text{ДК}} = } 1 001 \end{array}$$

Основи на аритметиката на двоични числа

- Операция **събиране**
 - **Пренос и препълване** при събиране на числа **със знак**
 - ✓ При събиране в **модифициран допълнителен код**

$$V_3 = z_n \oplus z_{n-1}$$

$$\begin{array}{r} \text{1 10} \quad \leftarrow \text{преноси} \\ [-12]_{\text{мдк}} = 11 \ 0100 \\ + \\ [-13]_{\text{мдк}} = 11 \ 0011 \\ \hline 10 \ 0111 \end{array}$$

$$\begin{array}{r} 0 \ 01 \ 1 \quad \leftarrow \text{преноси} \\ [12]_{\text{мдк}} = 00 \ 1100 \\ + \\ [13]_{\text{мдк}} = 00 \ 1101 \\ \hline 01 \ 1001 \end{array}$$

Основи на аритметиката на двоични числа

- Операция **изваждане**

Свежда се до операцията $Z = X + (-Y)$

- Операция **умножение**

($X \cdot Y = Z$) се определя като Y -кратно събиране на множимото X или като X -кратно събиране на множителя Y (по силата на комутативния закон), т.е.

$$Z = X + X + \dots + X = \sum_0^Y X; \quad Z = Y + Y + \dots + Y = \sum_0^X Y.$$

Основи на аритметиката на двоични и числа

- Операция **деление** ($X/Y = Z$) определя частното Z като число, което показва колко пъти делителят Y се съдържа в делимото X , ако $Y \neq 0$.

Частното може да се намери, като се преброят възможните последователни изваждания на делителя от делимото или на делителя от получаваните разлики, наричани *частични остатъци*. В зависимост от стойността на последния частичен остатък w , частното се представя така:

$$X = \begin{cases} Y \cdot Z + 0, & \text{ако } w = 0, \text{ точно деление;} \\ Y \cdot Z + \frac{w}{Y}, & \text{ако } w \neq 0, \text{ неточно деление.} \end{cases}$$

Двоично кодирани десетични числа

- BCD-числа (BCD – *Binary Coded Decimal*).
- Представянето на десетичните символи чрез техните двоични кодови комбинации, е своеобразна имитация на десетичните числа с машинни средства.

Пример:

268,47 → 0010 0110 1000 , 0100 0111

- Двоично-десетичните числа се представят най-често във форма с фиксирана запетая, т.е. като последователност (низ) от двоични кодове на десетичните цифри на числото.

Двоично кодирани десетични числа

- Формати на BCD-числата:
- **пакетиран** – две BCD – цифри в един байт

Пример:

$$1001\ 0111_{\text{BCD}} = 97_{10}$$

- **непакетиран** – една BCD – цифра в един байт

Пример:

$$00001001\ 00000111_{\text{BCD}} = 97_{10}$$

Двоично кодирани десетични числа

Съставеният за целите на двоично-десетичните изчисления код на десетичните цифри е желателно да притежава следните пет свойства, наричани *изисквания на Рутисхаузер*:

1. Еднозначност. Съответствието между кодовата комбинация и десетичната цифра трябва да бъде еднозначно. Ако това е изпълнено, става възможно да се оперира с двоично-десетично кодирани числа, при това декодирането е ефективно.



Двоично кодирани десетични числа

2. Монотонност. Според това свойство, на цифри, означаващи по-голямо елементарно количество, се полага код, който може да се интерпретира като по-голямо двоично число (или противоположното). Такова съотношение способства за по-лесно реализиране на операция сравнение.

3. Четност. На четни десетични цифри трябва да съответстват четни двоични комбинации (или противоположното). Свойството четност спомага за извършване на правилно десетично закръгляне и спомага за по-голяма простота при изпълнение на операция деление.



Двоично кодирани десетични числа

4. Допълняемост. Това свойство е в сила, ако във всяка сума от две десетични цифри, която е равна на максималната цифра, цифрите са кодирани с такива тетради, че едната може да бъде получена от другата чрез *поразрядна инверсия*. Това свойство способствува синтеза на допълнителен и обратен код на числата със знак.

5. Тегловна значимост. Според това свойство елементарното количество, означено с дадена десетична цифра, трябва да се получава от двоичния код като сума от теглата на позициите, в които има значеща цифра.

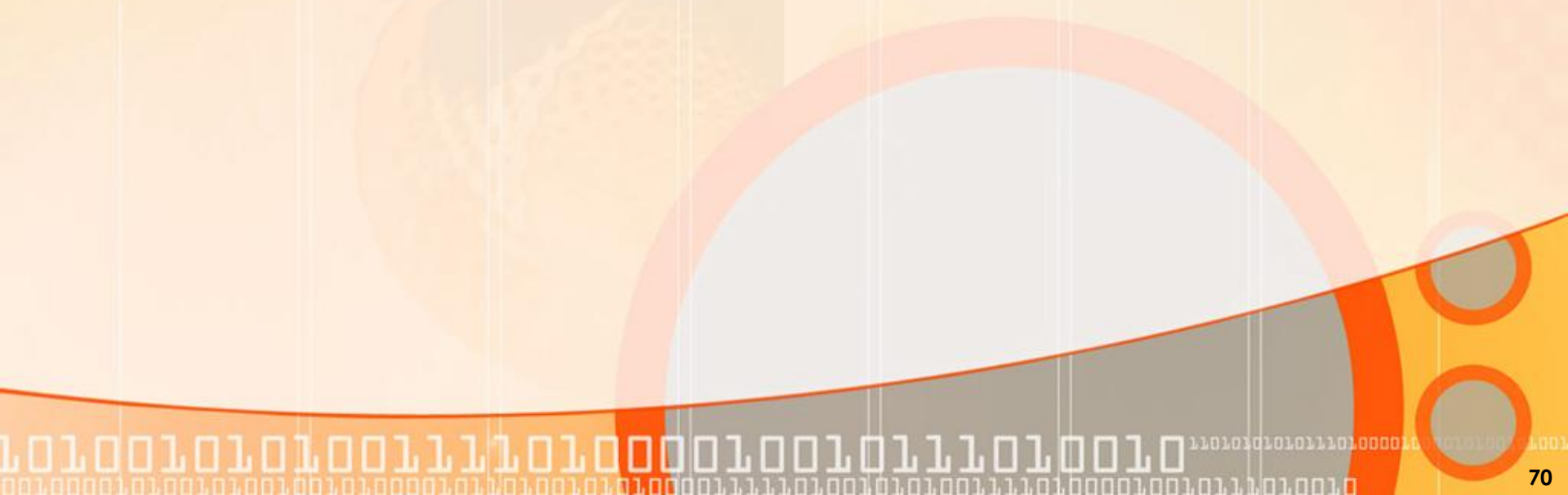
Двоично кодирани десетични числа

Ако теглата на позициите са положителни, то може да се покаже, че единственият код, който притежава безусловно свойството еднозначност, е кодът на *прякото* заместване (8421).

Двоично кодирани десетични числа

Кодове с излишък относно кода на прякото заместване (8421)

Формулата на тяхното получаване е сумата $(8421)+C$, където C е константа, формираща всяка кодова комбинация след означената операция.



Двоично кодирани десетични числа

Кодове с излишък C относно кода на прякото заместване (8421)

Всички кодове с излишък равен на 7 и по-голям от 7 ($C \geq 7$), не притежават свойството монотонност. Тези, които са с излишък по-малък от 7 ($C < 7$), притежават свойствата еднозначност, четност и монотонност, но само един от тях (**кодът с излишък три, $C=3$**) притежава свойството допълняемост.

Свойствата на кодовете с излишък са тясно свързани с тези на кодовете, имащи отрицателни тегла.



Двоично кодирани десетични числа

BCD 8421	BCD 8421+3	Десе- тични цифри
0000	0011	0
0001	0100	1
0010	0101	2
0011	0110	3
0100	0111	4
0101	1000	5
0110	1001	6
0111	1010	7
1000	1011	8
1001	1100	9

Забранени
комбинации

BCD 8421	BCD 8421+3
1010	0000
1011	0001
1100	0010
1101	1101
1110	1110
1111	1111

Основи на аритметиката на двоично кодирани десетични числа

- Операция **събиране**

- **Код 8421**

- **Корекция** на резултата:

$$Z_i = Z_i + 6_{(10)}, \quad \text{ако } 10_{(10)} \leq Z_i \leq 15_{(10)}$$

или ако **е** възникнал
междутетраден пренос

- **Код 8421⁺³**

- **Корекция** на резултата:

$Z_i = Z_i + 3_{(10)},$ ако **не е** възникнал
междутетраден пренос

$Z_i = Z_i - 3_{(10)},$ ако **е** възникнал
междутетраден пренос

• **Код 8421**

$$\begin{array}{r} r \text{ dddd} \dots \text{ddd} \\ + \quad 0 \text{ 6666} \dots \text{666} \\ \hline r \text{ d'd'd'} \dots \text{d'd'd'} \end{array}$$

$$[X]_{\text{ДК}} = [X]_{\text{ОК}} + 1.$$

Машинни кодове на двоично кодирани десетични числа

- Код 8421⁺³

$$[X]_{OK} = r \, \bar{d} \bar{d} \bar{d} \dots \bar{d} \quad , \quad r = q - 1 = 9;$$

$$\overline{(d_i)}_{(10)} = \left(\overline{b_3} \overline{b_2} \overline{b_1} \overline{b_0} \right)_{(2/10)}$$

$$S = \begin{cases} 0, & \text{за знак (+)}; \\ r, & \text{за знак (-)}. \end{cases}$$

$$[X]_{\text{ДК}} = [X]_{\text{ОК}} + 1.$$

Литература:

- [1]. <http://tyanev.com/> - On-line книги – ОРГАНИЗАЦИЯ НА КОМПЮТЪРА – книга [1] – **Глава 1. ПРЕДСТАВЯНЕ НА ДАННИТЕ;**
- [2]. <http://tyanev.com/> - On-line книги – ОРГАНИЗАЦИЯ НА КОМПЮТЪРА – упражнения книга [2];
- [3]. Димитър Тянев, ОРГАНИЗАЦИЯ НА КОМПЮТЪРА, том първи (ISBN 978-954-20-0412-7), Варна 2008г.
- [4]. Димитър Тянев, ОРГАНИЗАЦИЯ НА КОМПЮТЪРА - упражнения, ISBN 978-954-20-0258-0, Варна 2007г.

