

Binary Coded Decimal (BCD)

BCD е начин за изразяване на всяка десетична цифра в двоична бройна система. Всяка десетична цифра, започваща от 0 до 9, има свой собствен BCD код, който е 4-битов беззнаков двоичен еквивалент на цифрата. С помощта на тези индивидуални BCD кодове можем да запишем BCD стойността на всяко голямо десетично число.

- Например, десетичното число 36 може да се представи като "0011 0110"
- Цифровите часовници на много обществени места са пример за използване в ежедневието: цифрите за час, минута и секунда са седемсегментни дисплеи, които работят с двоично кодирани десетични числа (BCD).
- BCD кодът осигурява отличен интерфейс за двоични системи, например за клавиатурни входи и цифрови показания.

Въпреки че BCD е двоично представяне на десетичните числа, то не е еквивалентно на двоичното представяне на десетично число като int в компютрите.

- Представянето на десетичното число 13 като int-тип в компютъра има двоичния вид "1101", докато съответният BCD е "00010011".

Пример: Колко бита са необходими за кодиране на числата от 0 до 99 като int и като BCD

- За представяне като int са необходими 7 бита, защото броят на числата от 0 до 99 са 100. Тъй-като $100 = 2^{6.64}$, т.е. необходими са поне 7 бита за представяне на всички стойности в оказания диапазон.
- За представяне на числата от 0 до 99 като BCD-числа са необходими 8 бита, защото числата са най-много двуцифрени (по 4 бита за всяка цифра).

Пакетирани и непакетирани BCD числа

BCD кодът на една десетична цифра е група от четири двоични бита.

- В случая на непакетирани BCD числа всяка група от 4 двоични бита се съхранява в регистър на цифрова система.
 - Ако обаче размерът на регистъра е 8 бита или повече, тогава всяко съхраняване на BCD число губи много места в паметта.
 - В случай на 8-битов регистър долните 4 бита съхраняват BCD числото, а горните 4 бита - нулите.
- За да се спестят места в паметта, BCD числата могат да се съхраняват в пакетирана форма.
 - В случай на опаковано BCD число, 8-битов регистър може да съхранява две BCD цифри, една в горните 4 бита и друга в долните 4 бита.
 - Повечето компютърни процесори съхраняват BCD в опакована форма.

„8421“ BCD код

Най-популярният BCD код е кодът „8421“. Обозначението „8421“ показва двоичните тегла на четирите бита, използвани в BCD (2^3 , 2^2 , 2^1 , 2^0). Въпреки че съществуват различни видове BCD кодове, поради популярността на код 8421, като цяло BCD се отнася за код 8421, освен ако не е посочено друго.

Decimal digit	8421	2421	Excess-3
0	0000	0000	0011
1	0001	0001	0100
2	0010	0010	0101
3	0011	0011	0110
4	0100	0100	0111
5	0101	1011	1000
6	0110	1100	1001
7	0111	1101	1010
8	1000	1110	1011
9	1001	1111	1100
Unused code words / Invalid codes			
	1010	0101	0000
	1011	0110	0001
	1100	0111	0010
	1101	1000	1101
	1110	1001	1110
www.vlsifacts.com	1111	1010	1111

Кодовете 2421 и Excess-3 са самодопълващи се кодове. Това означава, че кодовата дума за допълване на всяка цифра до 9 може да се получи чрез допълване на отделните битове на кодовата дума на цифрата.

- Например десетичната цифра 2 в Excess-3 е "0101". Ако допълним "0101", ще се получи "1010", което е кодът на Excess-3 на десетичната цифра 7 (вижте таблицата по-горе). Тъй като 7 е допълнението на 2 към 9, това показва, че кодът Excess-3 е самодопълващ се.

Правило за преобразуване на десетично число в BCD-формат

За да превърнете едно десетично число в BCD, всяка цифра от десетичното число с се заменя с нейния 4-битов двоичен еквивалент.

Пример:

- Цяло число: $9673_{10} = (1001011001110011)_{BCD}$
- Реално число: $96.73_{10} = (10010110.01110011)_{BCD}$

За да преобразуваме BCD число в десетично число е необходимо отдясно-наляво да се групират битовете в групи по 4 бита, след което всяка група се заменя всяка група с еквивалентната десетична цифра (цифрата, имаща дадения 4-битов код).

Пример:

- $(100001010111)_{BCD} = 85710$
- 011110101000 – това не е BCD-код, защото трите групи от битове са **0111**, **1010** и **1000**. В случая втората група (1010) е недопустима като код за BCD-цифра и затова битовата поредица е невалидно BCD-число.

Сумиране на две BCD-числа

1. Сумиране на две числа

- Стъпка 1: Съберете двете BCD числа, като използвате правилата за двоично събиране, т.е. с преноси от предходен бит към следващия.
- Стъпка 2: Ако 4-битовата сума е равна или по-малка от 9 (1001_2), това е валидно BCD число.
- Стъпка 3: Ако 4-битовата сума е по-голяма от 9 (1001_2) или ако се генерира пренос от 4-я бит към следващата битова група, това е невалиден резултат.
 - В този случай се добавя 6 (0110_2) към 4-битовата сума - така се пропускат шестте невалидни кодови думи BCD, т.е. връщане към коректен 8421 BCD код.
 - Ако при добавянето на 6 се получи пренос, просто добавете преноса към следващата 4-битова група.

Пример:

<table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: right;">← Пренос</td> </tr> <tr> <td style="text-align: center;">0000</td> <td style="text-align: center;">0010</td> <td style="text-align: center;">0101</td> <td style="text-align: center;">0101</td> <td style="text-align: right;">+255</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">2</td> <td style="text-align: center;">5</td> <td style="text-align: center;">5</td> <td></td> </tr> <tr> <td style="text-align: center;">+</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">0000</td> <td style="text-align: center;">0000</td> <td style="text-align: center;">0110</td> <td style="text-align: center;">0011</td> <td style="text-align: right;">+63</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td style="text-align: center;">6</td> <td style="text-align: center;">3</td> <td></td> </tr> <tr> <td colspan="4" style="border-top: 1px solid black;"></td> <td></td> </tr> <tr> <td style="text-align: center;">0000</td> <td style="text-align: center;">0011</td> <td style="text-align: center;">0001</td> <td style="text-align: center;">1000</td> <td style="text-align: right;">+318</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">3</td> <td style="text-align: center;">1</td> <td style="text-align: center;">8</td> <td></td> </tr> </table>	0	1	0	0	← Пренос	0000	0010	0101	0101	+255	0	2	5	5		+					0000	0000	0110	0011	+63	0	0	6	3							0000	0011	0001	1000	+318	0	3	1	8		<table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">0111</td> <td style="text-align: center;">0101</td> <td style="text-align: right;">+75</td> </tr> <tr> <td style="text-align: center;">7</td> <td style="text-align: center;">5</td> <td></td> </tr> <tr> <td style="text-align: center;">+</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">0011</td> <td style="text-align: center;">0101</td> <td style="text-align: right;">+35</td> </tr> <tr> <td style="text-align: center;">3</td> <td style="text-align: center;">5</td> <td></td> </tr> <tr> <td colspan="3" style="border-top: 1px solid black;"></td> </tr> <tr> <td style="text-align: center;">1010</td> <td style="text-align: center;">1010</td> <td style="text-align: right;">Невалидно</td> </tr> <tr> <td style="text-align: center;">10</td> <td style="text-align: center;">10</td> <td></td> </tr> <tr> <td style="text-align: center;">+</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">0110</td> <td style="text-align: center;">0110</td> <td></td> </tr> <tr> <td style="text-align: center;">6</td> <td style="text-align: center;">6</td> <td></td> </tr> <tr> <td colspan="3" style="border-top: 1px solid black;"></td> </tr> <tr> <td style="text-align: center;">0001</td> <td style="text-align: center;">0001</td> <td style="text-align: center;">0000</td> <td style="text-align: right;">+110</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td></td> </tr> </table>	0111	0101	+75	7	5		+			0011	0101	+35	3	5					1010	1010	Невалидно	10	10		+			0110	0110		6	6					0001	0001	0000	+110	1	1	0	
0	1	0	0	← Пренос																																																																																						
0000	0010	0101	0101	+255																																																																																						
0	2	5	5																																																																																							
+																																																																																										
0000	0000	0110	0011	+63																																																																																						
0	0	6	3																																																																																							
0000	0011	0001	1000	+318																																																																																						
0	3	1	8																																																																																							
0111	0101	+75																																																																																								
7	5																																																																																									
+																																																																																										
0011	0101	+35																																																																																								
3	5																																																																																									
1010	1010	Невалидно																																																																																								
10	10																																																																																									
+																																																																																										
0110	0110																																																																																									
6	6																																																																																									
0001	0001	0000	+110																																																																																							
1	1	0																																																																																								

2. Изваждане на две числа

- Разглежда се като сумиране на едно положително и едно отрицателно число
 - $21 - 10 = 21 + (-10) = 11$
- Стъпка 1: Положителното число се преобразува в BCD код.
- Стъпка 2: Отрицателното число се преобразува в допълнителен код до 10:
 - 2.1: От числото 9999 се изважда отрицателното число. Това е обратния код.
 - 2.2: Към получения резултат се добавя 1. Това е допълнителния код до 10.

- Стъпка 3: Събират се двете BCD числа от стъпка 1 и стъпка 2 по правилото за събиране на положителни числа.
- Стъпка 4: Полученият резултат е в допълнителен код.
 - Резултатът е положително число ако най-лявата BCD цифра е между 0 и 4. В противен случай е отрицателно число и се преминава към стъпка 5.
- Стъпка 5: При отрицателен резултат се налага обратно преобразуване, т.е. от 9999 се изважда резултатът и се добавя 1.

Пример:

- Да се изчисли: $255 + (-63) = +192$
- Стъпка 1: $255_{10} \rightarrow (001001010101)_{BCD}$
- Стъпка 2.1: $9999_{BCD} - 0063_{BCD} = 9936_{BCD}$
- Стъпка 2.2: $9936_{BCD} + 0001_{BCD} = 9937_{BCD} = (1001100100110111)_{BCD}$
- Стъпка 3:

	1	1	0	1	1	← Пренос
	0000	0010	0101	0101		
					+255	
+		2	5	5		
	1001	1001	0011	0111		
					-63	
	9	9	3	7		
	1	0000	0001	1001	0010	+192
			1	9	2	
↑						
	Пренос					
	(не се използва)					

- Стъпка 4: най-лявата BCD-цифра е 0, т.е. полученото число е положително и не се налага обратна корекция.

Пример:

- Да се изчисли: $1 + (-2) = -1$
- Стъпка 1: $1_{10} \rightarrow (00000001)_{BCD}$
- Стъпка 2.1: $99_{BCD} - 02_{BCD} = 97_{BCD}$
- Стъпка 2.2: $97_{BCD} + 01_{BCD} = 98_{BCD} = (10011000)_{BCD}$
- Стъпка 3: $(00000001)_{BCD} + (10011000)_{BCD} = 99_{BCD}$
- Стъпка 4: най-лявата BCD-цифра е 9, т.е. полученото число е отрицателно и се налага обратна корекция.
- Стъпка 5.1: $99_{BCD} - 99_{BCD} = 00_{BCD}$
- Стъпка 5.2: $00_{BCD} + 01_{BCD} = 01_{BCD}$
- Следователно резултатът е -01_{BCD}