ANALELE UNIVERSITĂȚII
"EFTIMIE MURGU" REȘIȚA
ANUL XXII, NR. 1, 2015, ISSN 1453 - 7397

Silviu Draghici, Eugen Raduca, Cristian Rudolf

Designing a Transcode of B.C.D. in S.B.C. and Study its Function Using Software Electronic Workbench

The transcoder we have set out to synthetize and to simulate in this paper is important due to the fact that it brings into discussion and also presents the family of weighted and unweights binary decimal codes, presented in binary sequences of 4 bits each. Some of these are used even today with great success in the technique of data transmission, especially in recognition and access keys to certain dedicated programs and soft wares. The binary symmetric code is among the first binary decimal weighted codes, which has been the basis for the creation and development of other such codes. Therefore we have set out to design and simulate such a transcoder.

***Keywords**: transcoder, binary symmetric code, binary decimal code, electronic workbench*

1. Introductory notions and a brief history if the symmetric binary code (S.B.C.)

The combinational logic circuit (C.L.C.'s) that we intend to synthesize and simulate herein is part of the code or transcoders converters. We achieve a transcoder that converts binary code decimal (BCD classic, that code 8-4-2-1) in SBC (Also known as code 2-4-2-1). We synthesize this type of transcoded code 2-4-2-1 and 8-4-2-1 in code because the code SBC (2-4-2-1) has a special historical significance. He first car behind the numerical calculation is interesting to know the students how evolved historically codes. Still present a brief history of well-known codes which are today widely used in computer systems. Some of these were derived from code 2-4-2-1.

Code and 2-4-2-1 is called Aiken code after the name of the imagined and the first computers used automatically. It's Professor Howard Aiken of Harvard University, who in collaboration with I.B.M. (International Business Machines) conducted in the years 1939 to 1944 an automatic calculators, called MARK I, which is the precursor of the first electronic computers that have emerged in the coming years.

The Aiken code sequences have the first five decimals same expression as in the code 8-4-2-1. Further, the corresponding binary sequence is obtained from the figure 5 to figure 4 0 changing to 1 and 1 to 0. The same applies for the obtaining of 6 of the sequence of 3, 7 2, 8 1 and 9 of the 0. It is a representation of "mirror". This means in mathematical terms, that each of a complement of nine decimal digits is expressed by a sequence resulting complementing to one decimal bits of sequence in question.

The codes enjoys this property is called self-complementary. Such a code has advantages in performing arithmetical operations and simplifies the block diagram of the arithmetic of digital computers.

Code 4-2-2-1 has the same properties as the code 2-4-2-1 (Aiken code) shown above: 2 share positions using two distinct fourfold and tetrad representing decimal numbers whose sum is equal to 9 complement each other; first five decimals have the first position 0, and last January 5.

The 5-4-2-1 code is characteristic that between 5 and 9 figures differ from figures 0 ÷ 4 only the first bit.

Code 7-4-2-1 using the weights 7, 4, 2, 1; under these conditions it is observed that the number 7 can be represented in two ways (0111 or 1000) and to remove the ambiguity introduced an additional restriction, namely: use of all possible combinations that which corresponds to the maximum number of significant bits.

The codes given so far are weighted codes.

The Excess Code 3 is obtained from the code 8-4-2-1, by adding every nibble of number 3 (0011) in binary. It follows a code of self-complementary property and which eliminated a code combination 0000. But it is not weighted.

The Gray code and it bears the name of the imagined and characterized in that the transition from one to the next decimal place is by modifying a single binary rank fourfold. He is an unweight code. Gray code sequences can be derived from the code of 8-4-2-1, using the following relationships:

$$b_1 = a_1 \oplus a_2, b_2 = a_2 \oplus a_8, b_3 = a_4 \oplus a_8, b_4 = a_8 \quad (1)$$

where b1, b2, b3 and b4 are positions Gray sequence written from right to left, and a1, a2, a8 a4 and the positions of the code are written in the order 8-4-2-1 weights.

Code "2 of 5" is a code for pseudo weighted as decimal numbers 1, 2, ..., 9 bits of sequence can associate with 74,210 shares; Instead associated with the zero sequence no longer complies with this rule weights. The main feature of the code "2 of 5" lies in the fact that all of the decimal digits associated binary sequences have the same number of significant bits (1), which is by 2 (out of the 32 binary combinations that can be formed by 5 bits, ten satisfy this condition), which determined the award name. This property provides a criterion for detecting errors or otherwise creates the possibility of controlling transmission of information encoded in this way.

In the B.C.D. codes (8-4-2-1), each place is represented, as mentioned, a binary code with four or five bits (when we have negative numbers). Codes with four / five bits of each decimal are simply concatenated (number of groups in connection). For example, the number 2743 (base 10) will have correspondent 0010 0100 0111 0011 8-4-2-1 code (B.C.D.).

For a better understanding of the significance of these codes presented, we present comparative table below:

Table 1.

No. in decimal	Binary- decimal codes							
	Weighted codes					Unweight codes		
	8-4-2-1	2-4-2-1	4-2-2-1	5-4-2-1	7-4-2-1	Excess 3	Gray	"2 of 5" (74210)
0	0000	0000	0000	0000	0000	0011	0000	00011
1	0001	0001	0001	0001	0001	0100	0001	00101
2	0010	0010	0010	0010	0010	0101	0011	00110
3	0011	0011	0011	0011	0011	0110	0010	01001
4	0100	0100	0110	0100	0100	0111	0110	01010
5	0101	1011	1001	1000	0101	1000	0111	01100
6	0110	1100	1100	1001	0110	1001	0101	10001
7	0111	1101	1101	1010	0111	1010	0100	10010
8	1000	1110	1110	1011	1001	1011	1100	10100
9	1001	1111	1111	1100	1010	1100	1101	11000

Theoretically, we can say that all the codes shown in Table 1 belong, somehow, the extended family of B.C.D. codes

2. Designing the transcoder from B.C.D. classic (8-4-2-1) to S.B.C. (2-4-2-1)

Observing the Table 1 we actually represent the four Boolean logic functions of output DS, CS, BS, and AS that characterizes the S.B.C., depending on the variables D, C, B and A characterizing B.C.D. input classic. With D and DS have noted M.S.B. and L.S.B. sites A and AS's.

Also in Table 1 the truth, we note that:

$$A_S = A \tag{2}$$

For other Boolean functions D_S, C_S, B_S we draw diagrams and the Karnaugh-Veitch 4 binary variable, taking account of Table 1, so as to succeed the Boolean logic to determine the minimum form. We will present how to minimize them in the following figures:

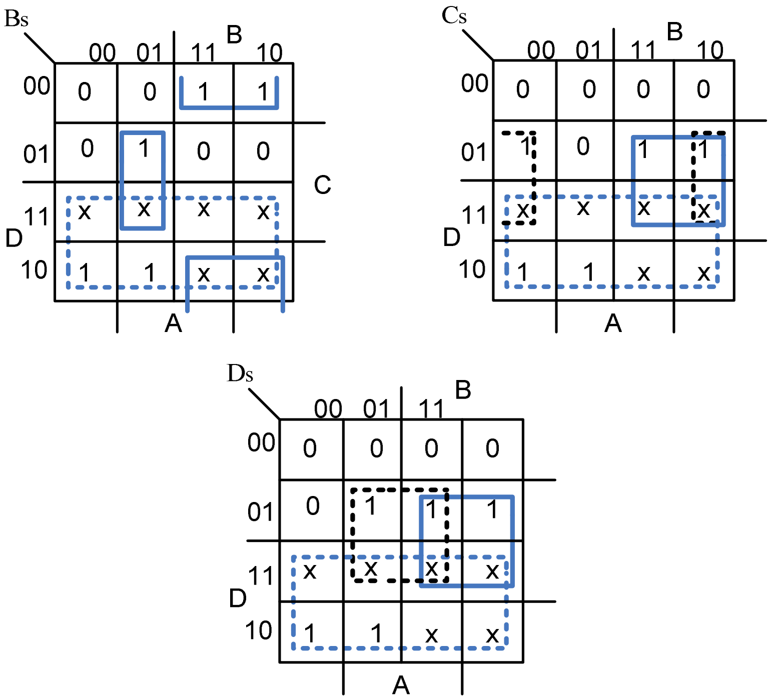

Figure 1: Minimising D_S, C_S and B_S.

From figure 1 we can clearly observe that:

$$B_S = D + \overline{C} \cdot B + C \cdot A \cdot \overline{B} \quad (3)$$

$$C_S = D + B \cdot C + C \cdot \overline{A} \quad (4)$$

$$D_S = D + C \cdot B + A \cdot C \quad (5)$$

Using relations (2), (3), (4) and (5) synthesizing a flowchart illustrating trans-coder 8-4-2-1 / 2-4-2-1, as follows:

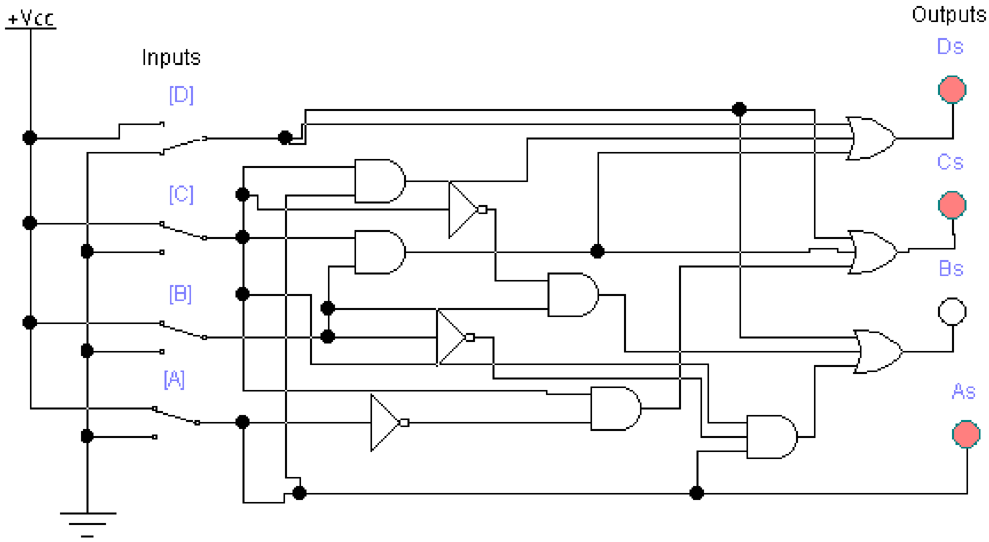

Figure 2. The logic scheme of the transcoder realised in the Electronic Workbench program

3. Conclusions

The + VCC symbol libraries Electronic Workbench TTL logic-level supply means; can do well flowchart of the transcoder respectively in CMOS technology, in which case the existing symbol libraries .ewb programming environment is + VDD. The outputs of the transcoder 8-4-2-1 / 2-4-2-1 I used to highlight 1L and 0L TTL logic levels LED indicator lights (1L logical level) or off (logic level 0L).

I represented only a pair of combination equivalent code (corresponding figure 7) to the two codes (binary decimal classic, symmetrical binary respectively) of space saving. But it is a suggestive presentation complementary with mathematical symmetry to 9. It is easy to notice that the outputs of the transcoder presented here are complementary to two (i.e. 1101). The same happens with all other values greater than 4. The operation flowchart of Figure 2 is accurately verified the values presented in Table 1. As such, the transcoder obtained and simulated here is correct.

References

[1]. Răduca E., Răduca M., Ungureanu-Anghel D., *Circuite digitale,* Editura Eftimie Murgu, Reşiţa, 2010.

[2]. Chioncel C., *Prelucrarea numerică a semnalelor,* Editura Eftimie Murgu, Reşiţa, 2009.
[3]. Iancu V., Protocsil C., Gillich N., Haţiegan C., Gillich G-R., *The Influence of the Number of Finite Elements upon the Accuracy of the Results Obtained Using Discrete Models,* Analele Universităţii "Eftimie Murgu", Fascicula de Inginerie, Anul XX, Nr. 3, Reşiţa, 2013.
[4]. https://www.youtube.com/watch?v=d5rNk30xU544 existing at 07.06.2015
[5]. https://numangift.files.wordpress.com/2013/03/chap2.pdff existing at 07.06.2015

Addresses:

- PhD. Cand. Eng. Silviu Drăghici, "Eftimie Murgu" University of Reşiţa, Piaţa Traian Vuia, nr. 1-4, 320085, Reşiţa, s.draghici@uem.ro,
- Prof. PhD. Eng. Eugen Raduca, "Eftimie Murgu" University of Reşiţa, Piaţa Traian Vuia, nr. 1-4, 320085, Reşiţa, e.raduca@uem.ro
- Lect. PhD. Cristian Rudolf, "Eftimie Murgu" University of Reşiţa, Piaţa Traian Vuia, nr. 1-4, 320085, Reşiţa, c.rudolf@uem.ro