Formation of optimal Boolean functions for analog-digital conversion

S. Kushch, *Senior Member, IEEE*, J.M. Corchado, *Member, IEEE*

Abstract — **This paper describes a situation in which the function obtained from an analog- to-digital conversion is considered a partially-certain function. In this case it is possible to neglect the least significant bits and therefore redefine them arbitrarily. A realization of proposed method simplifies the further processing of the received signal, and saves hardware consumption at designing and usage of hardware and software systems that include analog-to-digital converters.**

Keywords — **Cognate-implementation, analog-to-digital converter, Boolean functions, partially-certain functions, Boolean functions redefine.**

I. Introduction

THE extensive development of the idea of "Smart Cities" and "Smart Things" requires increased computing power and reducing the overall dimensions of chips for compact wearable devices and other mobile electronics. At the same time, increasing capacity entails increasing power consumption, which can be a problem especially in devices that use batteries for compact power supply (for example - Smartphones, Smart Watches, unmanned vehicles, GPS trackers, etcetera). Developers are also faced with another problem - the miniaturization of elements in modern electronic devices. Each time another step is taken to reduce the technological size of modern chips to a minimum, it is met with growing difficulties, this is due to the current technological restrictions on production possibilities. There is a number of fundamental obstacles that impede further reduction in the size of individual elements that are found in integrated circuits:

- the greater the amount of elements on a chip per unit volume, the more heat needs to be dissipated from the chip, this greatly increases the power consumption of cooling systems which can be even more than that of the electronic device itself;

- electrodynamic constraints caused by capacitance and inductance inertia in the circuit. This hinders the rapid change of a voltage and a current during transition from one state to another (for example, logic operation keys in the microprocessor or dynamic memory cells);

S. Kushch, is with the BISITE research group of the University of Salamanca, Edificio I+D+i, Calle Espejo s/n, 37007, Salamanca, Spain (e-mail: kushch@usal.es).
J. M. Corchado is with the University of Salamanca, Edificio I+D+i, Calle Espejo s/n, 37007, Salamanca, Spain (e-mail: corchado@usal.es).

- while reducing the size of the object to the atomic scale, an atomic and electronic discreteness is visible in the transport phenomena, in the interaction of elementary particles, etc. All these problems lead us to look for new ways of improving the microelectronic structures of computers, which are becoming increasingly important to manufacturers. A one of these ways is an optimisation of using different forms representation of Boolean functions for forming logical circuits at the stage of logical design and for processing of signals which can be presented as Boolean functions.

II. Problem formulation

It is well known that the form of the same BF will differ if written in different RF. It should be noted that the arguments in these forms are present, both directly and in inverted form. This means that for the hardware implementation of these functions it is necessary to increase the number of input contacts twice (for direct and inverse signals).

In addition to the above mentioned, there are also forms of presenting Boolean (logical) functions that use signal in only one form - direct or inverse, such as, Reed-Muller AF (polynomials Reed-Muller) [1], [9], [10], Algebraic RF [7] and others.

For example:

- the classical RF - in the disjunctive normal form

$$f(x_i) = \sum_{i=0}^{2^n-1} c_i \tilde{x}_1 \tilde{x}_2 ... \tilde{x}_n \,, \tag{1}$$

Here $c_i = \{0,1\}$ - are coefficients of $\tilde{x}_1 \tilde{x}_2 ... \tilde{x}_n$ - full or non-full conjunctions which have up to n variables in direct or in inverse form;

- the Algebraic form representation of BF (ARF)
- in the form of algebraic polynomials.

$$f(x_i) = \sum_{i=0}^{2^n-1} c_i S_i \,, \tag{2}$$

Here: S_i are a special piecewise-constant basis functions [2];

c_i - coefficients of S series.

Reed-Muller FR (RMFR) – in the form of sum mod2 of the some S functions.

978-1-5386-3073-0/17/$31.00 ©2017 European Union

$$f(x_i)=\sum_{i=0}^{2^n-1}\oplus c_i S_i, \bmod 2, \qquad (3)$$

$c_i=\{0,1\}$.

Consider in detail the latter two forms of presentation.

Example 1:

X2	X1	f
0	0	0
0	1	1
1	0	1
1	0	1

Logical function of 2 arguments $f_{14}=x_1\vee x_2$ has number 14 and its Q- vector has the form $Y^{(Q)}_{14,2}=[0111]$, also Q-series has form:

$$f_{14}=0\cdot q_{02}+1\cdot q_{12}+1\cdot q_{22}+1\cdot q_{32}$$

For CFR – the vector of coefficients of Q-series with q-functions of the same *n-th* order or, equivalently, a truth table column with logical function values which were written from top to bottom as a line. This variant is referred to as the Q-vector of the initial function.

For AFR – the vector of the coefficients of the S-series or, equivalently a result of canonic F-transformation.

For example:

$$f_{14}\sim \Phi_{14}=X_1+X_2-X_1X_2=0\cdot S_{02}+1\cdot S_{12}+1\cdot S_{22}-1\cdot S_{32},$$

The S-vector of the logical function has this form –

$$Y^{(S)}_{14,2}=[011-1].$$

For the RMRF – the vector of the coefficients of the G-series or equivalently, the result of representing a logical function as the Zhegalkin polynomials [11]-[13]. The conjunctions, which are included in the polynomial, are designated as the corresponding S-functions, which are summed mod2.

For the logical functions:

$$f_{14}=x_1\oplus x_2\oplus x_1x_2=0\cdot S_{02}\oplus 1\cdot S_{12}\oplus 1\cdot S_{22}\oplus 1\cdot S_{32_{\mathrm{mod}2}}$$

The G-vector of the logical function has this form

$$Y^{(G)}_{14,2}=[0111].$$

In general, all the representation forms BF (CRF, AFR, RMFR) can be treated uniformly as series consisting of ordinary conjunctions or conjunctions in the form of Q- or S-functions (that is, Φ-images of these conjunctions). Members of series are summed with weight coefficients, and the summation can be:

- logical (CFR):
- usual (AFR);
- by mod2 (RMFR).

With this approach, for example, the task of transforming the CFR to the AFT of the given *i-th* logical function from *n* arguments can be written in the analytical form, as follows:

$$\Upsilon^{(S)}_{in}=QS(\Upsilon^{(Q)}_{in}) \ . \qquad (4)$$

Here: $\Upsilon^{(S)}_{in}$ - the desired vector of the *i-th* of logical function in AFR, i.e. in the form of an S-series;

$\Upsilon^{(Q)}_{in}$ - a given vector of the *i-th* logical function in the CRF in the form of a Q-series;

QS - the transition operator from CRF to ARF, i.e. transformation Q-series to S-series.

Regardless of the RF the dimensions of Q-, S-, and G-vectors used are usually 2^n and the set of all logical functions of n arguments L(n) is a linear vector space of dimension 2^n.

The transition from one form of logical function to another can be interpreted as a change of basis, it is equivalent to transforming the coordinates of a logical function. This is because the coefficients of the Q-, S-, and G-series can be regarded as the coordinates of the logical function in the corresponding basis.

The multivariate optimization of Boolean (logical) functions is another promising research area today. This situation arises, for example, when a partially defined Boolean function is received, it is a result of specific processes in progress being predicted while independent expert hardware systems are built. In this case, there is a possibility of choice among several acceptable variants with similar parameters.

Cognate-representation of Boolean functions (BF) was proposed in [2]-[4], [8] as a generalization of classical one-valued realization of combinational circuits (CC), which are information cores of finite state machine (FSM). Classical implementation of CC on n-binary-inputs and m-binary-outputs lie in the formation of the m-BF, each of which implements a single BF of n arguments (5).

$$\begin{aligned} y_1&=f_1(x_1,x_2\ldots,x_n)=f_1(X_n)\\ y_2&=f_2(x_1,x_2\ldots,x_n)=f_2(X_n)\\ &\cdots\\ y_m&=f_m(x_1,x_2\ldots,x_n)=f_m(X_n)\end{aligned} \qquad (5)$$

Here: x_i - the input signals CC, y_i - the output functions of CC, f_i - the logic functions for the *i*-d output of CC.

Cognate-implementation of BF differs from the considered classic form in that it allows for the application of the so-called "Cognate" BFs along with polynomial BFs. This is what system (5) looks like, if Cognate-implementation is used:

$$\begin{cases} Z_1=f_1[X^{(n)}]\vee F_{11}[X^{(n)}]\vee\cdots\vee F_{1p_1}[X^{(n)}];\\ \quad\vdots\\ Z_m=f_m[X^{(n)}]\vee F_{m1}[X^{(n)}]\vee\cdots\vee F_{mp_m}[X^{(n)}];\end{cases} \qquad (6)$$

Here:

$X^{(n)}$ is the vector's arguments of dimension n, scilicet is the vector of discrete signals at the inputs of the CC;

$f_1[X^{(n)}],\ldots,f_m[X^{(n)}]$ are BFs which are set by the truth table or otherwise scilicet BF for CC, which realizes the nominal operation mode of CC;

$F_{11}[X^{(n)}],\ldots,F_{1p_1}[X^{(n)}]$ is the range of implementation variants for the initial function $f_1[X^{(n)}]$;

$F_{m1}[X^{(n)}],\ldots,F_{mp_1}[X^{(n)}]$ is the range of implementation variants for the initial function $f_m[X^{(n)}]$.

Further on we will omit references to the visual dependency on $X^{(n)}$ all components of (2), and the system (1) will be recorded as:

$$\begin{cases} Z_1 = f_1 \vee F_{11} \vee \cdots \vee F_{1p_1}; \\ \vdots \\ Z_m = f_m \vee F_{m1} \vee \cdots \vee F_{mp_m}. \end{cases} \quad (7)$$

Formula (7) shows that in the proposed system, each BF from m BFs is given by a truth table or otherwise, can be implemented in general as a related (Cognate) ρ_i option.

In [5] it was proved that any BF can be implemented in Cognate-form, i.e. any BF has a set of relative functions F_{ij}, the power of which depends on the conditions of the application of Cognate-implementation.

III. MAIN RESULTS

The aim of this paper is a quantitative assessment of the effectiveness of Cognate-implementation compared to classical unambiguous implementation in the form of (1), provided the existence of several permissible (close) variants of Boolean functions.

Clearly, the effectiveness of Cognate-implementation depends on the conditions of formation of the set of relative BFs. The variety of options for such shaping does not allow to describe the set of options in full volume. So in the paper three options are analyzed using the Cognate-implementation of BFs and three options of shaping the set of relative BFs:

1. Cognate-implementation of BF in blocks that process the digital information and, in addition, have elements of identification and correction errors in one, two or three arbitrary discharges of output signal;
2. Cognate-implementation of BF in which a set of relative BFs consists of various alternative forms of BF representation compared to the nominal classical form;
3. Cognate-implementation of BF in which digital information is processed after analog-to-digital conversion of analog input information.

The choice of indicated options is determined by strict formation rules of BFs relative sets, which, with the help of Extended Data Mining technology [6], will conduct a computational experiment aimed at establishing the statistical average of the effectiveness of Cognate-implementation.

Further on, we will study one of the above mentioned situations – saving hardware expenditures in the analog-to-digital conversion.

Usually the number of bits in typical ADC of the middle class are 12; that's why a mistake in one lower order bit is 2^{-12} =2,4*10^{-4}, in two bits - 2^{-11}=4.9*10^{-4}, in three bits – 2^{-10}=9,8*10^{-4}.

We will then study in detail one of the cases mentioned above - namely, the processing optimization of BF using the Cognate-FR after the analog-to-digital conversion. Subsequent processing of the functions is possible due to software or hardware via high speed - PLA (e.g. from the Achronix Semiconductor company).

Hence, it is reasonable to assume that nominal function fi (3) is partly determined by BF with three uncertain lower order digits, in typical engineering implementation tasks he precision of analog input data does not exceed ± 1%. This allows for each nominal, partially determined BF, after an optimal extension of definition, to make appropriate optimally-determined BFs, for example in one of the known forms of presentation – the Classical, the Reed-Muller or the Algebraic form of presentation [7]. Fig.1 shows the scheme of the process of analog-to-digital conversion with the additional blocks of processing and minimization.

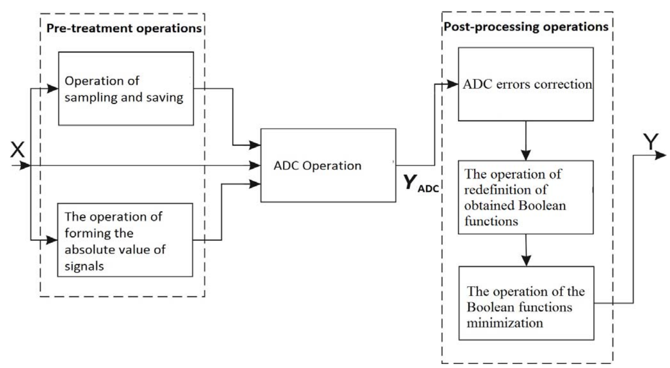

Fig. 1. The proposed process analog-to-digital conversion.

The quantitative comparison of parameters of Boolean functions in classical FR and optimal BF after the definition was extended by indicators - S_S, S_{ad}, S_L. The results of calculations are shown in Table. 1-3, where:

- S_{ad} - amount of summands in the Boolean function record which determines the number of inputs of the sub matrices of the PLA2, that is in the part of the PLA, where the disjunctions are formed;

- S_S – Overall area of submatrix of conjunctions formed in PLA1;

- S_L – the classic indicator – amount of letters in a minimized disjunction-normal form of BF [7].

TABLE 1: A comparison of the summary indicators for k=1

Amount of argu-ments	*Amount of BF in L(n)*	*A comparison of the summary indicators, k=1*					
		Ss precise	Ss approx.	S_L precise	S_L approx.	S_{ad} precise	S_{ad} approx.
2	16	80	28	29	16	20	14
3	256	3540	1548	1218	756	590	486
4	65536	2167176	1206936	766860	550650	270897	248066

TABLE 2: A comparison of the summary indicators for k=2

Amount of argu-ments	*Amount of BF in L(n)*	*A comparison of the summary indicators, k=2*					
		Ss precise	Ss approx.	S_L precise	S_L approx.	S_{ad} precise	S_{ad} approx.
2	16	80	16	29	8	20	16
3	256	3540	1332	1218	624	590	428
4	65536	2167176	1101424	766860	492908	270897	232344

TABLE 3: A comparison of the summary indicators for k=3

Amount of argu-ments	*Amount of BF in L(n)*	*A comparison of the summary indicators, k=1*					
		Ss precise	Ss approx.	S_L precise	S_L approx.	S_{ad} precise	S_{ad} approx.
2	16	80	-	29	-	20	-
3	256	3540	1128	1218	512	590	376
4	65536	2167176	995264	766860	437528	270897	215288

IV. Conclusion

In this paper, we examined cases where the definition of a function was extended for one, two and three least significant bits, for functions at n = 2, 3, 4 (variables) for all sets of functions (Table I-Table III). These results show that using of the proposed method for choosing the optimal form of Boolean functions representation can be used at the stage of logical design to construct the optimal form of a logical element. In addition, when processing noise-contaminated digital signals, which was obtained by analog-to-digital conversion, these signals can be considered as partially defined Boolean functions, which will allow recovering their shape using the most optimal form of function recording for a subsequent processing of them. This line of research requires further researching and will be covered in subsequent publications.

References

[1] B. Harking, “Efficient algorithm for canonical Reed-Muller expansions of Boolean functions,” *IEE Proceedings E-Computers and Digital Techniques,* vol. 137, no. 5, pp. 366-370, Sep. 1990.

[2] Yu. Kochkarev, S. Kushch, “A Cognate-realization of logic functions on the basis of their representation in isomorphic form,” *Electronic Modeling*. vol. 34, no. 4, pp. 119-123, Aug. 2012.

[3] Yu. Kochkarev, S. Kushch, E. Panasko, “Analysis of promising parts of logic functions in the problem of logical design of digital blocks,” *Bulletin of Cherkasy state technological university*, vol. 4, pp. 54-56, Dec. 2011.

[4] Yu. Kochkarev, S. Kushch, “Technology of Cognate-implementation of logical functions,” *Bulletin of Cherkasy state technological university*, vol. 3. pp. 35-38, Sep. 2011.

[5] Yu. Kochkarev, S. Kushch, “Justification of a Cognate-form of representation and realization of logical functions,” *Electronic Modeling*, vol. 33, no. 6. pp. 73-80, Des. 2010.

[6] Yu. Kochkarev, V. Busko, N. Kucherova, “The Investigation of the structure of a complete set of logical functions based on EDM technology,” *Bulletin of Cherkasy state technological university*, vol. 1-2, pp. 139-140, Apr. 2007

[7] Yu. Kochkarev, N. Kazarinova, N. Panteleeva, S. Shakun, *Catalog-reference “Classical and alternative minimal forms of logical functions,”* Yu. Kochkarev ed., Cherkassy, Ukraine: ChIM, 1999, pp.16-28.

[8] S. Kushch, “Forming of the optimal Boolean functions in the analog and digital conversion using cognate-realization,” *Scientific and Technical Journal “Priborostroenie”,* no. 10, pp. 55-59, Oct. 2014.

[9] K. Saluja, E. Ong, “Minimization of Reed-Muller canonic expansion”,*IEEE Trans.Comput*, vol. 28, no. 7, pp.535-537, Jul. 1979

[10] S. Swamy, "On Generalized Reed-Muller Expansions," *IEEE Transactions on Computers*, vol. C-21, no. 9, pp. 1008-1009, Sept. 1972. doi: 10.1109/TC.1972.5009078

[11] S. Marshall, “The Theory of Representations for Boolean Algebras”, *Transactions of the American Mathematical Society*, vol. 40, no. 1, pp. 37–111, Jan. 1936. doi:10.2307/1989664.

[12] D. Vladimirov, *Boolean algebras*, [in Russian], Moscow, USSR: Nauka, 1969, p. 320.

[13] I. Zhegalkin, “On the Technique of Calculating Propositions in Symbolic Logic”, *Matematicheskii Sbornik,* vol. 1, no. 34, pp. 9–28, 1927.