International Journal of Innovative Research in Information Security (IJIRIS) ISSN: 2349-7017(O)
Issue 2, Volume 5 (May 2015) ISSN: 2349-7009(P)
www.ijiris.com

Fully Efficient Encoding Technique Using SOBS method for DSRC Applications

Banarjun S M[1] Mr. Pradeep M N
M.tech VLSI Design, Dept. Of ECE, *Asst. Professor, Dept. Of ECE,*
Dayananda Sagar College of Engineering

Abstract— The automotive industry is aggressively emerging towards the direction of advanced active safety. The automotive industry is aiming for the development of Dedicated Short Range Communication (DSRC) technology, for the purpose in automobile-to-automobile and automobile-to-roadside communication. To transmit the vehicular information and broadcast vehicle position, DSRC communication technology acts as bridge. This work is devoted to improve the encoding techniques for WAVE/DSRC applications. The DSRC standards generally adopt FM0 and Manchester encoding to achieve dc balance and enhancing signal reliability. Nevertheless, the diversity of coding between FM0 and Manchester codes limits the potential to design a fully utilize hardware architecture for both. The Similarity oriented Boolean simplification (SOBS) technique is proposed here so that the Hardware utilization rate (HUR) will reach to 100% for both encoding techniques. The proposed architecture will have less delay, area as compared to existing architecture. In this paper the architecture analyzed to reduce the number of components. Using both encodings the area, delay, and power is reduced in DSRC. The power consumption is 22.13µW for both encoding, and area is 193µm^2. The encoding technique in this work supports the DSRC standards used by various organizations of America, Japan and Europe.

Keywords— DSRC (Dedicated Short Range Communication), FM0, Manchester, VLSI

I. INTRODUCTION

The DSRC (Dedicated Short Range Communication) is a protocol for one or two way medium range communication. The DSRC can be broadly classified into two: automobile-to-automobile and automobile-to roadside. In automobile-to-automobile, the DSRC enables the message sending and broadcasting among automobile. The automobile-to-roadside focuses on the intelligent transportation system, such as electronic toll collection (ETC).The DSRC architecture having the transceiver shown in Fig. A. The transceiver having the baseband processing, RF front end and microprocessor. The microprocessor is used to send the instruction to the baseband processing and RF front end. The RF front end is used to transmit and receive the wireless signals through the antenna. The baseband processing is responsible for modulation, error correction, encoding and synchronization. The transmitted signal consists of the arbitrary binary sequence; it is very difficult to obtain the dc-balance. The (SOBS) similarity oriented Boolean simplification having the two methods: area compact retiming and balance operation sharing. The area compact retiming used to reduce the transistor counts. The balance logic operation sharing is used to combine the FM0 and Manchester encoding.

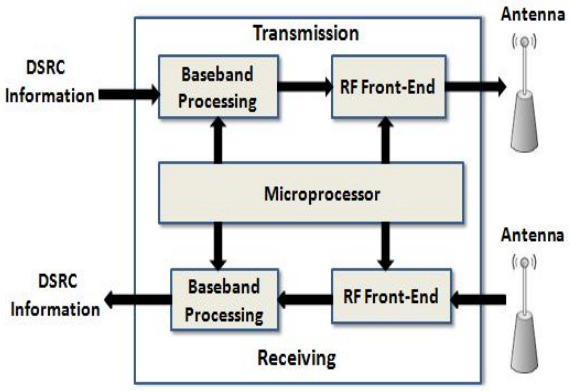


Fig. A. DSRC transeciver architecture

II. CODING PRINCIPLES OF FM0 AND MANCHESTER CODE

A. Fm0 encoding

The FM0 having the following three rules.

1) If X = logic-0, the FM0 code has the transition between the A and B.
2) If X = logic-1, there is no transition is allowed between the A and B.
3) The transition is allocated in each FM0 code.

The wave form is given below the following diagram Fig. B.

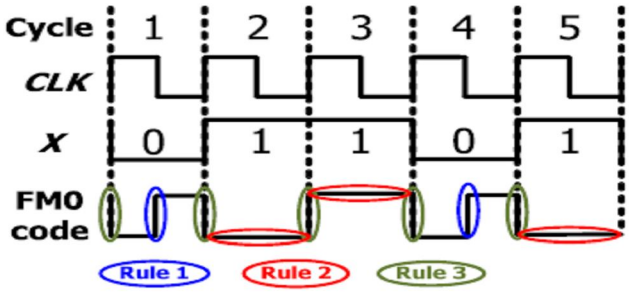


Fig. B. Illustration of FM0 coding for specified input

B. Manchester encoding

The Manchester encoding is derived with the XOR operation for using the CLOCK and X. The clock always has a transition within the one cycle. The waveform is given in the Fig. C.

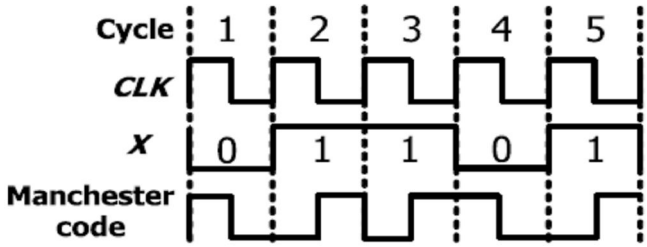


Fig. C. Illustration of Manchester coding for specified input

III. STATE CODE PRINCIPLE FOR FM0/MANCHESTER

The Manchester encoding is an XOR operation only. The FM0 code derived with the FSM principle. The FSM of FM0 code classified into four states. The four states as shown in the below in Fig. D.

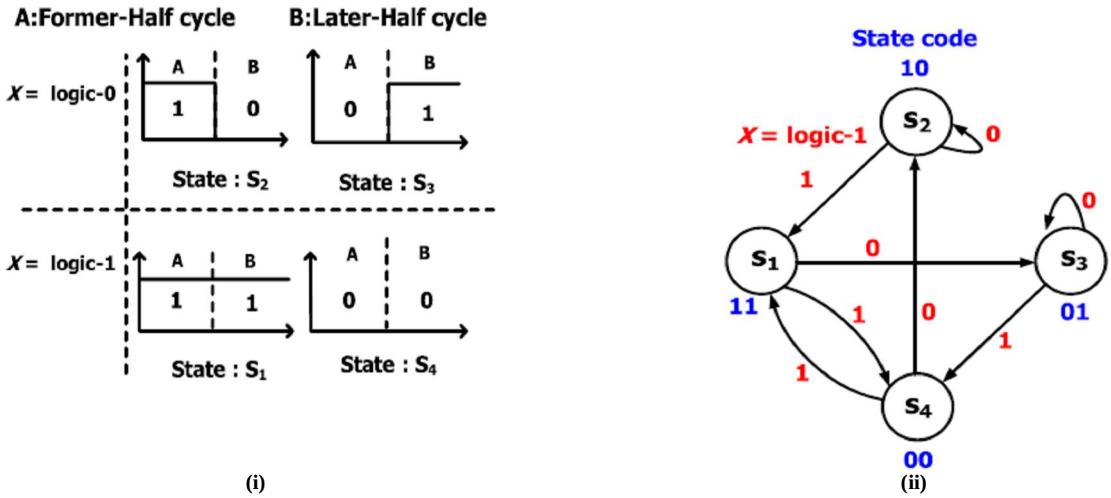


Fig. D. Illustration of FSM of FM0.
(i) States definition (ii) State diagram

Suppose initial state is S1, and its state code is 11 for A and B respectively.

- If X = logic 0, the state transition must follow both rules for FM0 1 and 3. The only remaining next state that satisfies both rules for the X of logic 0 is S3. If X = logic 1, then the transition must follow both rules for FM0 2 and 3. The only remaining next state that satisfies both rules for X of logic 1 is S4. Thus, the transition of each state can be completely designed. The FSM of FM0 results in transition table of each state A (t) and B (t). The previous states are denoted as A (t-1) and B (t-1), respectively.

- A (t) = B (t-1)
  B (t) = X⊕ B (t-1)

With A (t) and B (t), the expression for FM0 code is given as

CLK A (t) + ~ CLK B (t)

TABLE II

STATE TRANSITION TABLE OF FM0

Previous State		Current State			
A(t-1)	B(t-1)	A(t)		B(t)	
		X=0	X=1	X=0	X=1
1	1	0	0	1	0
1	0	1	1	0	1
0	1	0	0	1	0
0	0	1	1	0	1

IV. HARDWARE ARCHITECTURE OF FM0 AND MANCHESTER CODE

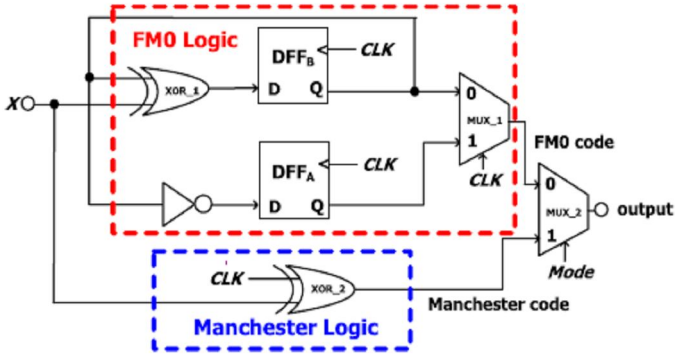


Fig. E. Hardware architecture of FM0 and Manchester encoding

The existing hardware architecture is as shown in Fig. F. The top part is the FM0 and the below part is Manchester code. In FM0, the DFFA and DFFB are used to store the state code of FM0 code and also mux_1 and inverter is use in FM0 code. When mode = 0, FM0 code is selected and if mode = 1, then Manchester code will be selected. The HUR (Hardware Utilization Rate) is given as,

$$HUR = \frac{Active\ Components}{Total\ Components} \times 100\%$$

where the active components are the components that work in both FM0 and Manchester code. The total components are the components that are present in whole circuit. The HUR rate for existing VLSI architecture is given below. For both encoding in the above architecture the total components is 7. For FM0 code the active components is 6 and in Manchester code the active component is 2. In both encoding the transistor count is 98 without the use of SOBS technique. Hence the calculations are done as shown in Table II.

TABLE II

HUR OF FM0 AND MANCHESTER ENCODINGS

Coding	Active Components (transistor count) / Total Components (transistor count)	HUR
FM0	6 (86) / 7 (98)	85.7%
Manchester	2 (26) / 7 (98)	28.6%
Average	4 (56) / 7 (98)	57.1%

V. FM0 AND MANCHESTER ENCODER USING SOBS TECHNIQUE

The SOBS is categorized into two parts area compact retiming and balance operation sharing.

A. *area compact retiming*

For FM0 the state code of the each state is stored into DFFA and DFFB .the transition of the state code is only depends on the previous state of B(t-1) instead of the both A(t-1) and B(t-1).

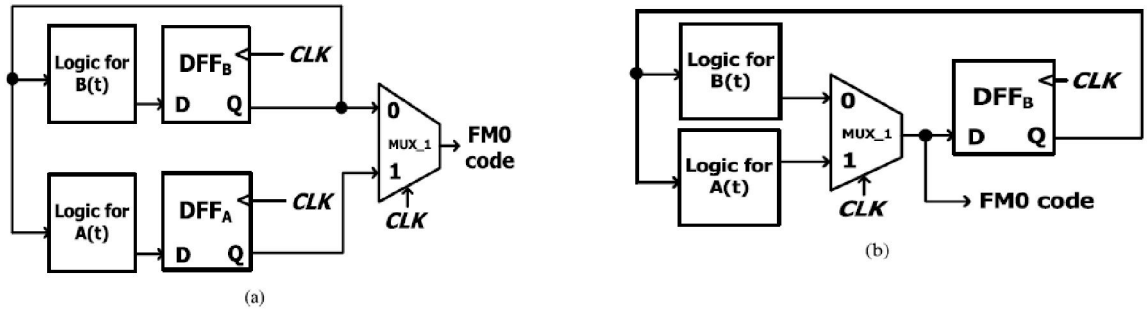


Fig. F. Illustration of area efficient retiming on FM0 encoding architecture.
(a) FM0 encoding without area efficient retiming. (b) FM0 encoding with area efficient retiming.

The previous state is denoted as the A (t-1) and then the B (t-1) and then the current state is denoted as the A (t) and then the B (t). Thus, the FM0 encoding just needs a single 1-bit flip-flop to store the previous value B (t−1). If the DFFA is removed, a non synchronization between A (t) and B (t) causes the logic error of FM0 code. To avoid this logic-error, the DFFB is relocated right after the MUX−1, where the DFFB is assumed to be positive-edge triggered flip flop. At each cycle, the FM0 code, comprising A and B, is derived from the logic of A (t) and the logic of B (t), respectively. The FM0 code is alternatively switched between A (t) and B (t) through the MUX−1 by the control signal of the CLK. In the Q of DFFB is directly updated from the logic of B (t) with 1-cycle latency. When the CLK is logic-0, the B (t) is passed through MUX−1 to the D of DFFB. Then, the upcoming positive-edge of CLK updates it to the Q of DFFB. The B (t) is passed through MUX−1 to the D of DFFB. Then, the upcoming positive-edge of CLK updates it to the Q of DFFB. The timing diagram for the Q of DFFB is consistent whether the DFFB is relocated or not. The transistor count of the FM0 encoding architecture without area-compact retiming is 72, and that with area-compact retiming is 50. The area-compact retiming technique reduces 22 transistors.

B. Balance operation sharing

The Manchester encoding is derived using the XOR operation. The equation of the XOR gate is given below.

$$X \sim CLK + \sim X\ CLK$$

The concept of balance logic-operation sharing is to integrate the X into A(t) and X into B(t). The FM0 and Manchester logics have a common point of the multiplexer like logic with the selection of the CLK. The A (t) can be derived from an inverter of B (t − 1), and X is obtained by an inverter of X. The logic for A (t)/~X can share the same inverter, and then a multiplexer is placed before the inverter to switch he operands of B (t − 1) and X. The Mode indicates either FM0 or Manchester encoding is selected. The similar method can be also applied to the logic for B (t)/X.

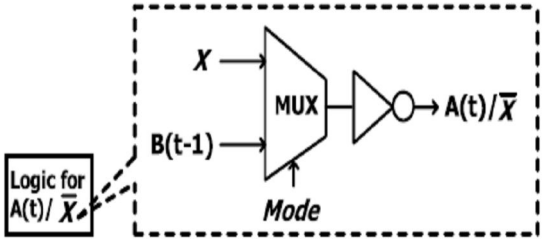


Fig. G. Balance logic simplification for A (t) and ~X

Nevertheless, this architecture results a drawback that the XOR is only used for FM0 encoding, and is not shared with Manchester encoding. Therefore, the HUR of this architecture is limited. The X can be also interpreted as the X⊕0, and thereby the XOR operation can be shared with Manchester and FM0 encodings, where the multiplexer irresponsible to switch the operands of B (t−1) and logic-0. This architecture shares the XOR for both B (t) and X, and there by increases the HUR. When the FM0 code is selected, the CLR is disabled, and the B (t −1) can be derived from DFFB .Hence, the multiplexer can be totally saved, and its function can be completely integrated into the relocated DFF. The logic for A (t)/~X includes the MUX−2 and an inverter. Instead, the logic for B (t)/X just incorporates a XOR gate. In the logic for A (t)/~X, the computation time of MUX−2is almost identical to that of XOR in the logic for B (t)/X. However, the logic for A (t)/~X further incorporates an inverter in the series of MUX−2.

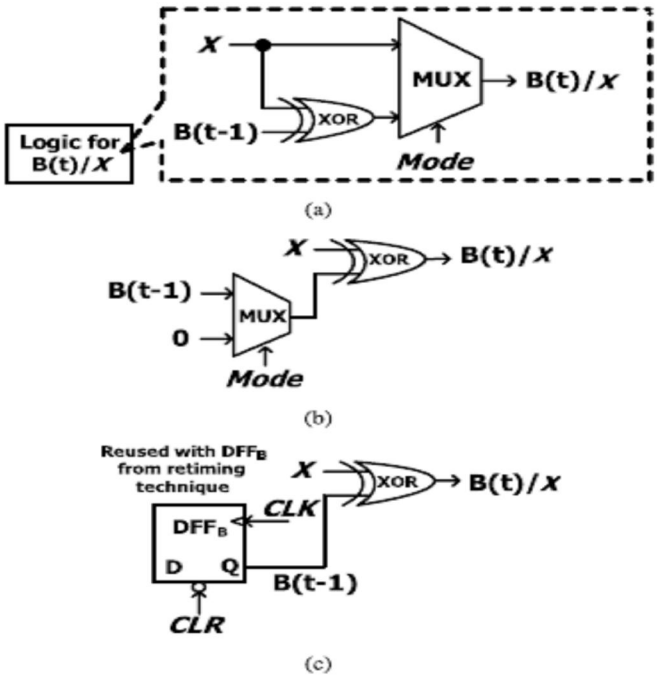


Fig. H. Balance logic simplification of B (t) and X. (a) without XOR sharing. (b)with XOR sharing (c) sharing the reused flip-flop

This unbalance computation time between A (t)/~X and B (t)/X results in the glitch to MUX−1, possibly causing the logic fault on coding. To alleviate this unbalance computation time, the architecture of the balance computation time between A (t)/~X and B (t)/X The XOR in the logic for B (t) /X is translated into the XNOR with an inverter, and then this inverter is shared with that of the logic for A (t)/~X. This shared inverter is relocated backward to the output of MUX−1. Thus, the logic computation time between A (t)/~X and B (t)/X is more balance to each other.

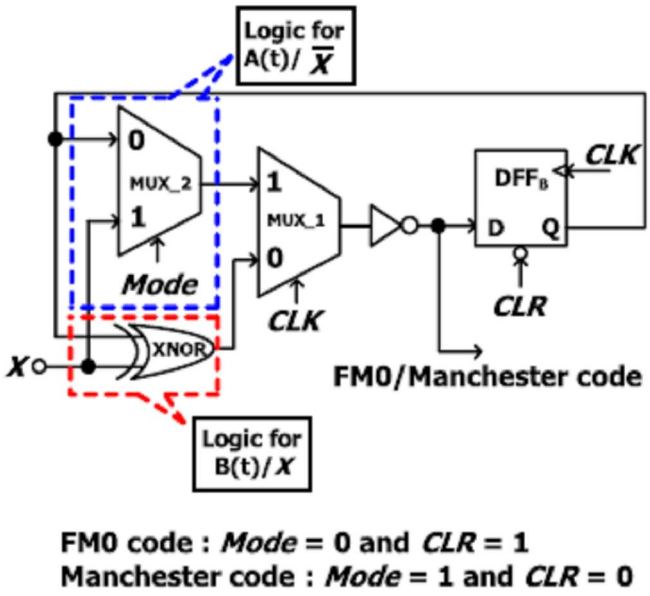


Fig. I. VLSI architecture for FM0/Manchester encoding

TABLE III

PERFORMANCE ANALYSIS OF SOBS METHOD

Coding	Active Components (transistor count) / Total Components (transistor count)	HUR
FM0	5 (44) / 5 (44)	100%
Manchester	5 (44) / 5 (44)	100%
Average	5 (44) / 5 (44)	100%

Proposed VLSI architecture

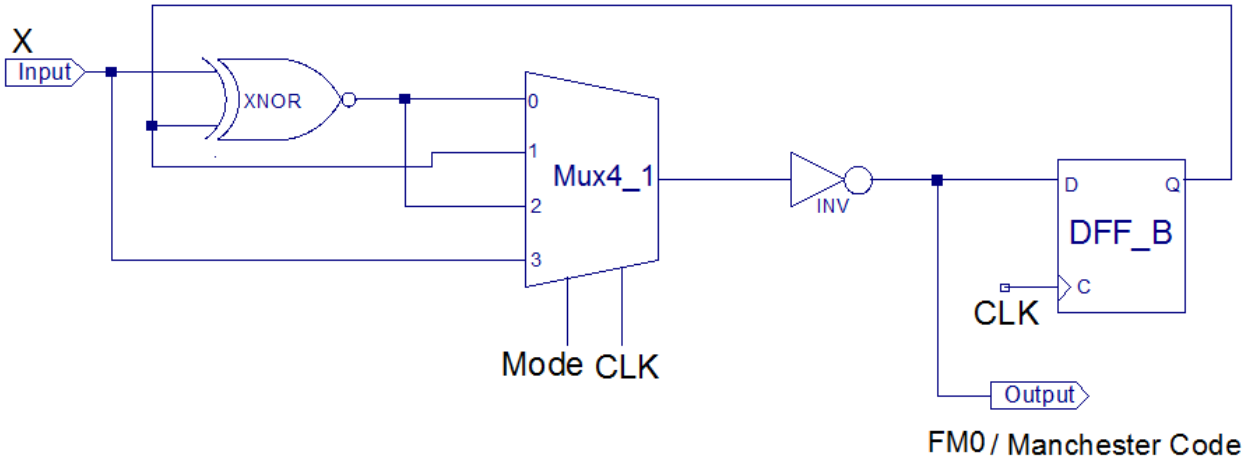


Fig. I. Proposed VLSI architecture for FM0/Manchester encoding

TABLE IV

PERFORMANCE ANALYSIS OF PROPOSED METHOD

Coding	Active Components (transistor count) / Total Components (transistor count)	HUR
FM0	4/ 4	100%
Manchester	4 / 4	100%
Average	4 / 4	100%

The proposed VLSI architecture contains only 4 hardware components which is fully reusable and efficient compared to the existing architecture. The Mux4_1 is used to replace the two 2_1 mux in the existing VLSI architecture designed using SOBS method. The HUR for the proposed is 100%.

VI. CONCLUSIONS

The coding-diversity between FM0 and Manchester encodings causes the limitation on hardware utilization of VLSI architecture design. A limitation analysis on hardware utilization of FM0 and Manchester encodings is discussed in detail. In this paper, the fully reused VLSI architecture using SOBS technique for both FM0 and Manchester encodings is proposed. The SOBS technique eliminates the limitation on hardware utilization by two core techniques: area compact retiming and balance operation sharing. The area-compact retiming relocates the hardware resource to reduce the transistors. The balance logic-operation sharing efficiently combines FM0 and Manchester encodings with the identical logic components. This paper is realized in 180nm technology with outstanding device efficiency. The power consumption is 22.13µW for Manchester encoding and FM0 encoding.

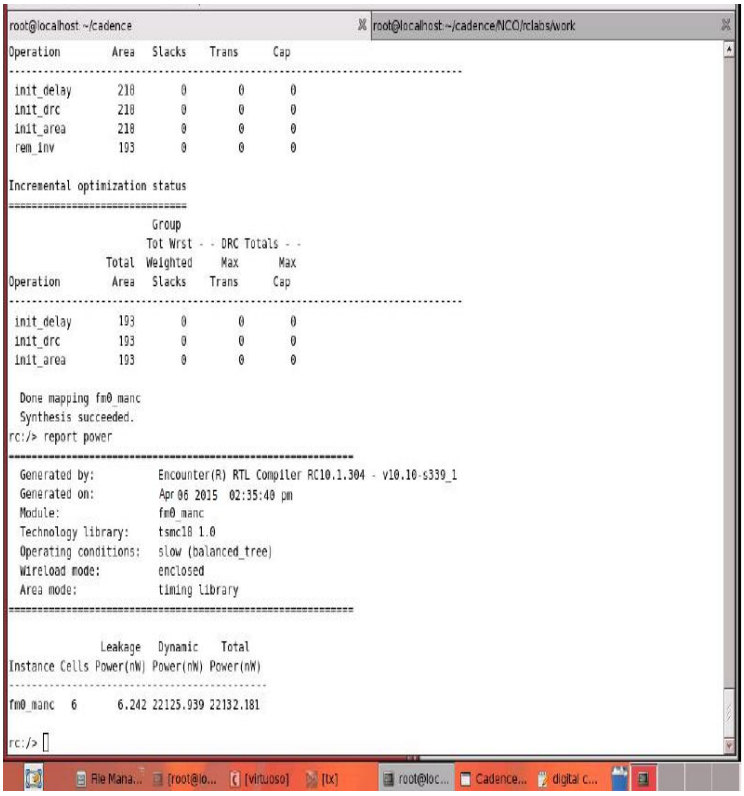
TABLE V

PERFORMANCE ANALYSIS OF PROPOSED ARCHITECTURE

	2014	This work
Realization	Xilinx FPGA Spartan 3	Xilinx FPGA Spartan 3
Supply voltage	3.3 V	3.3 V
Coding methods	Manchester FM0	Manchester FM0
Operation frequency	562 MHz	562 MHz
Power consumption	78mW	52 mW
HUR	100%	100%
Delay	5.92ns	5.78ns
FPGA resource usage	Slice : 2 Flip-flop : 1 4 input LUTs : 4 Bonded IOBs : 10	Slice : 1 Flip-flop : 1 4 input LUTs : 1 Bonded IOBs : 6

POWER REPORT

Technology	180nm
Power Consumption	22.13µW
Area	193µm^2
HUR	100%



REFERENCES

[1] Fully Reused VLSI Architecture of FM0/Manchester Encoding Using SOLS Technique for DSRC Applications, Yu-Hsuan Lee, *Member, IEEE*, and Cheng-Wei Pan, 1063-8210, 2014.

[2] F. Ahmed-Zaid, F. Bai, S. Bai, C. Basnayake, B. Bellur, S. Brovold, *et al.*, "Vehicle safety communications—Applications (VSC-A) final report," U.S. Dept. Trans., Nat. Highway Traffic Safety Admin., Washington, DC, USA, Rep. DOT HS 810 591, Sep. 2011.

[3] J. B. Kenney, "Dedicated short-range communications (DSRC) Standards in the United States," *Proc. IEEE*, vol. 99, no. 7, pp. 1162– 1182, Jul, 2011.

[4] J. Daniel, V. Taliwal, A. Meier, W. Holfelder, and R. Herrtwich, "Design of 5.9 GHz DSRC-based vehicular safety communication," *IEEE Wireless Commun. Mag.*, vol. 13, no. 5, pp. 36–43, Oct. 2006.

[5] P. Benabes, A. Gauthier, and J. Oksman, "A Manchester code Generator running at 1 GHz," in *Proc. IEEE, Int. Conf. Electron., Circuits Syst.*, vol. 3. Dec. 2003, pp. 1156–1159.

[6] A. Karagounis, A. Polyzos, B. Kotsos, and N. Assimakis, "A 90nm Manchester code generator with CMOS switches running at 2.4 GHz and 5 GHz," in *Proc. 16th Int. Conf. Syst., Signals Image Process.*, Jun. 2009, pp. 1–4.

