IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 6, Issue 6, Ver. III (Nov. - Dec. 2016), PP 12-20
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Implementation of FM0 and Manchester Encoder for Efficient Hardware Utilization

CH. Yagnasri[1], M. Kiran[2], K. Jamal[3]
[1]*Dept. of ECE, GRIET/JNTUH, India*
[2]*Dept. of ECE, GRIET/JNTUH, India*
[3]*Dept. of ECE, GRIET/JNTUH, India*

Abstract: *The automotive industry is working to develop the Dedicated Short-Range Communication (DSRC) technology. It is an emerging technique and key enabling technology to push the intelligent transportation system with safety applications into our daily life. The DSRC standards generally make use of FM0 and Manchester codes to have good dc-balance, enhancing the signal reliability. The distinguishable coding-diversity between the FM0 and Manchester codes causes a serious limitation to design fully reused VLSI architecture for both the codes. This paper proposes Similarity-Oriented Logic Simplification (SOLS) technique to overcome the design limitation. This SOLS technique is used to improve the hardware utilization rate from 57.14% to 100% for both FM0 and Manchester encoders. The design is implemented in the cadence virtuoso tool with 180nm CMOS technology. The power consumption is 196.9 μW for FM0 encoding and 197.08 μW for Manchester encoding. The maximum operating frequency is 900 MHz for FM0 and 2 GHz for Manchester encoding. This paper is also implemented in the Xilinx tool. The work done in this paper is used to develop a fully reused VLSI architecture, and it also reduces the power consumption when compared with the previous existing works.*

Keywords: *Dedicated Short Range Communication (DSRC), FM0, Manchester, SOLS, VLSI.*

I. Introduction

The Vehicle Safety Communication-Applications (VSC-A) [2] is Dedicated Short-Range Communication (DSRC) based. DSRC is a protocol for one-way or two-way medium range communication especially for the transportation systems. The DSRC is briefly classified into two categories. They are vehicle-to-vehicle and vehicle-to-roadside communication. In vehicle-to-vehicle, this DSRC protocol enables the message sending and broadcasting among vehicles for safety issues and public information announcement [3], [4]. The safety issues include intersection warning, blind-spot, inter-cars distance, and collision alarm. The second category vehicle to-roadside focuses on the transportation service, such as Electronic Toll Collection (ETC) system. With ETC the toll-collecting system is electrically accomplished with the contactless IC card platform. Moreover, the ETC can also be extended to the payment for parking service, and gas refueling. Thus, the DSRC system plays an important role in the modern automobile industry. The system architecture of DSRC transceiver is shown in Fig.1. The upper and bottom parts as shown in diagram are dedicated for transmission and receiving respectively. This transceiver is classified into three basic modules. They are microprocessor, baseband processing, and RF front-end. The microprocessor interprets instructions from media access control to schedule the tasks of baseband processing and RF front-end. The baseband processing is accountable and important for modulation, error correction, clock synchronization and encoding. The RF front-end is used for receiving and transmitting the wireless signal through the antenna.

The DSRC standards have been established by the several organizations in different countries. The Europe, America and Japan countries follow these DSRC standards. Different data rate, carrier frequency, modulation and encoding techniques are followed by different organizations. The data rate targets at 500 kb/s, 4Mb/s, and 27Mb/s with a carrier frequency of 5.8 and 5.9 GHz. The modulation method incorporates Amplitude Shift Keying, Phase Shift Keying, and Orthogonal Frequency Division Multiplexing modulation. Generally, the waveform of the transmitted signal is expected to have zero-mean for robustness concern, and this is also referred to as dc-balance. The transmitted signal consists of binary sequence, for which it is difficult to obtain dc-balance. The purpose of FM0 code and Manchester code is to provide the transmitted signal with good dc-balance.

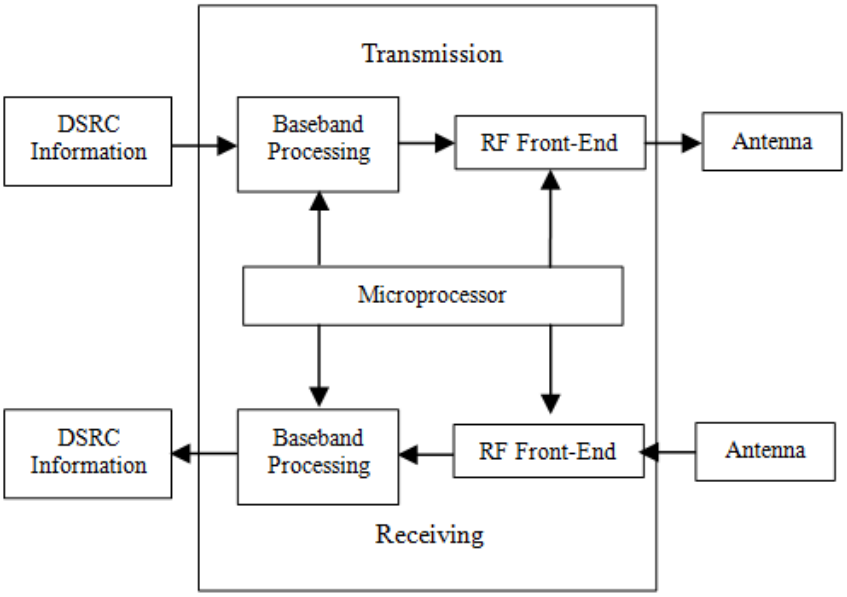

Fig. 1 System architecture of DSRC transceiver

Both the FM0 and Manchester codes are widely adopted in encoding part for downlink. The architectures of FM0 and Manchester encoders are reviewed as follows.

1.1 FM0 and Manchester Encoder Architectures Review.

The literature [5] describes about the VLSI architecture of Manchester encoder for optical communication. This design makes use of the CMOS inverter and the gated inverter as the switch and constructs Manchester encoder. It is realized in 0.35-µm CMOS technology and its operating frequency is 1 GHz. The literature [6] replaces the architecture of switch in [5] by the nmos device. It is implemented by 90-nm CMOS technology, and the maximum operating frequency is as high as 5 GHz. The literature [7] proposes a high speed VLSI architecture reused with Manchester and Miller encoders for Radio Frequency Identification (RFID) applications. This design is realized in 0.35-µm CMOS technology and its maximum operating frequency is 200 MHz. The literature [8] also proposes the architecture of Manchester encoding for Ultrahigh Frequency (UHF) RFID tag emulator. This hardware architecture is developed by using Finite State Machine (FSM) of Manchester code, realized into Field Programmable Gate Arrays (FPGA) prototyping system. The maximum operating frequency of this design is 256 MHz. The similar design methodology is applied further to individually develop Miller and FM0 encoders for UHF RFID Tag emulator [9]. Its maximum operating frequency is about 192 MHz. Furthermore, literature [10] combines Frequency Shift Keying (FSK) modulation and demodulation with the Manchester codec in hardware realization.

1.2 Organization

The organization of this paper is done as follows. Section 2 describes the coding principles of FM0 and Manchester encoder. Section 3 gives a limitation analysis on the Hardware Utilization Rate (HUR) of FM0 and Manchester encoders. This section depicts the difficulty to design fully reused hardware architecture for Manchester and FM0 encoders. The proposed architecture design which uses SOLS technique is reported in Section 4. The two core methods of SOLS technique are area-compact retiming and balance logic-operation sharing will be described in this section. The experiment results and performance evaluation are presented in Section 5. This section focuses on the performance evaluation of the design of Manchester and FM0 encoders. Finally, Section 6 deals with the conclusion.

II. Coding Rules of FM0 And Manchester Code

In the following discussion, the clock signal and the input data signal are abbreviated as CLK and X respectively. With these parameters, the coding rules of FM0 and Manchester codes are discussed and are shown as follows.

2.1. FM0 Encoding

As shown in Fig. 2, for each X, the FM0 code consists of two parts: one is for the former-half cycle of CLK, A, and the other one is for later-half cycle of CLK, B. The coding principles for FM0 are listed and is shown below with the following three rules.

1) If X has the logic-0 value, then FM0 code must have a transition between A and B.
2) If X is logic-1, no transition is allowed between A and B.
3) There must exist a transition between each FM0 code no matter what the X is.

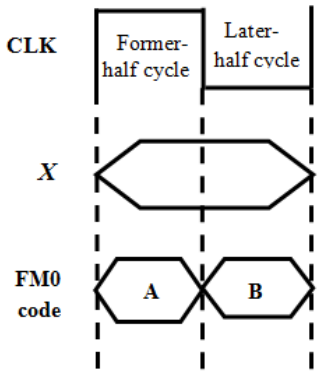

Fig. 2 Structure of FM0 codeword

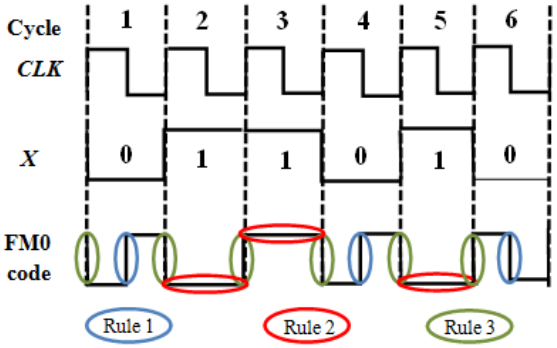

Fig. 3 Example of FM0 waveform

The above Fig. 3 shows an example of FM0 coding. At cycle 1, X has logic-0 hence, a transition occurs in its FM0 code, according to rule 1. For clearness, this transition is initially set from the value of logic-0 to logic-1. According to rule 3, a transition is allocated among each FM0 code, and therefore the logic-1 is changed to logic-0 at the beginning of cycle 2. Then, according to rule 2, this logic level continues to hold its value without any transition in the entire cycle 2 for the X of logic-1. Thus, the FM0 code for each cycle can be derived with these three rules that are mentioned.

2.2 Manchester Encoding

The example of Manchester coding is shown in Fig. 4. The Manchester code can be derived from

$$X \oplus CLK \qquad \text{---- (1)}$$

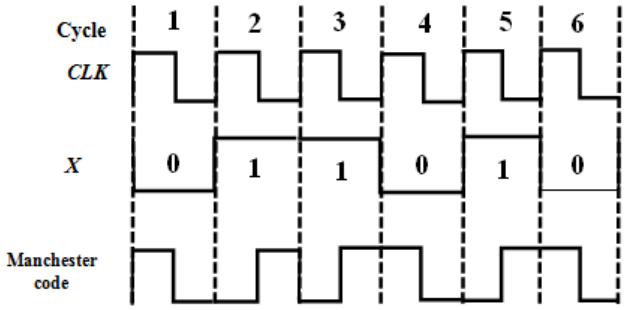

Fig. 4 Example of Manchester coding

Thus the Manchester encoding can be realized with an XOR operation of CLK and X. The clock signal always has a transition within one cycle, no matter what the X is.

III. Limitation Analysis On Hardware Utilization Of Fm0 Encoder And Manchester Encoder

To make a study on the hardware utilization of FM0 encoder and Manchester encoder, the hardware architecture of both are conducted first. As mentioned earlier, hardware architecture of Manchester encoding can be realized with a XOR operation. However, the realization of hardware architecture of FM0 is not so simple like the Manchester encoder.

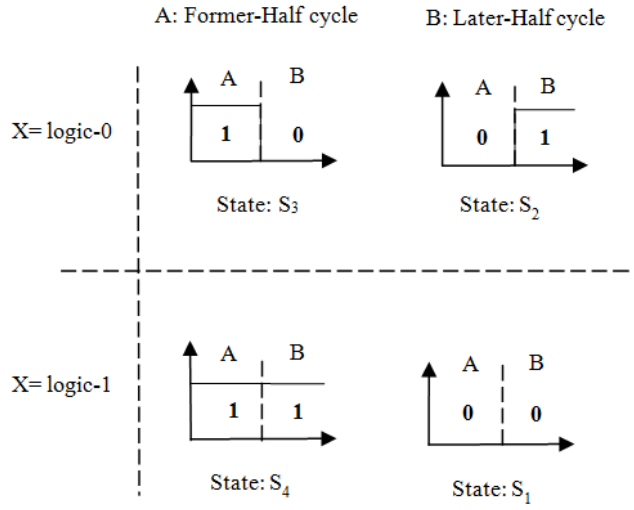

Fig. 5 States definition for FSM

To construct the hardware architecture of FM0 encoder we should know the FSM of FM0 first. As shown in Fig. 5, the FSM of FM0 code is classified into four states. A state code is assigned to each state, and each state code consists of A and B, as shown in Fig 2. According to the coding rules of FM0, the FSM of FM0 encoder is shown in Fig. 6. Suppose the initial state is S_1, let say its state code is 00 for A and B, respectively. If the value of X is logic-0, the state-transition must follow both the rules 1 and 3. The next state that can satisfy both rules for X, which is logic-0, is S_3. If the value of X is logic-1, the state-transition must follow both the rules 2 and 3. The next state that can satisfy both rules for X, which is logic-1, is S_4. Thus the state-transition of each and every state can be completely constructed. A(t) and B(t) represent the discrete-time state codes of current-state at the time instant t. Their previous-states are denoted by A(t-1) and B(t-1), respectively.

Here the Boolean functions of A(t) and B(t) are given as

$$A(t) = \overline{B(t-1)} \qquad \text{---- (2)}$$

$$B(t) = X \oplus B(t-1) \qquad \text{---- (3)}$$

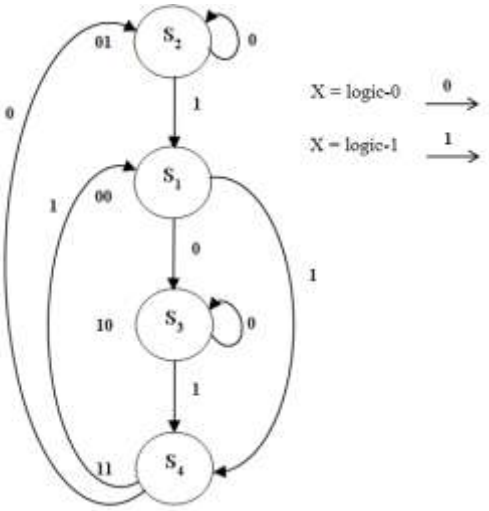

Fig. 6 FSM of FM0

With the both above A(t) and B(t) equations, the Boolean function of FM0 code is given by

$$CLK\ A(t) + \overline{CLK}\ B(t) \qquad \text{---- (4)}$$

With equation (1) and (4), the hardware architecture of FM0 and Manchester encoders is shown in Fig. 7.

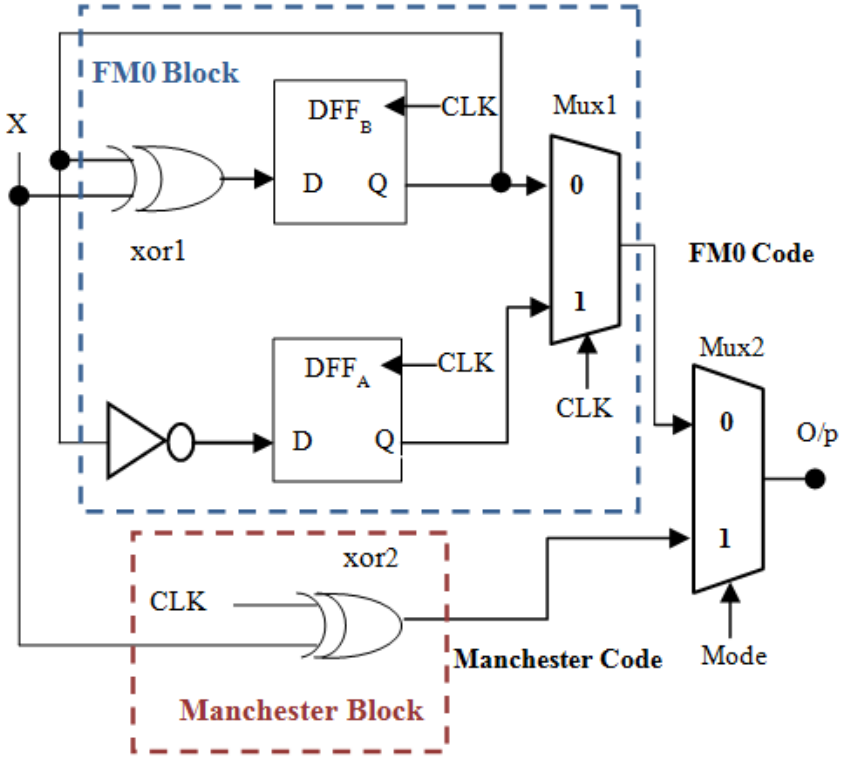

Fig. 7 Hardware architecture of FM0 and Manchester encoding

The upper part is the hardware architecture of FM0 encoder, while the lower part is the hardware architecture of Manchester encoder. As listed in equation (1), the Manchester encoding is as simple as a XOR operation of X with CLK. Nevertheless, the FM0 encoding depends not only on the input data signal X but also on the previous-state of the FM0 code. The DFF_A and DFF_B store the state code of FM0 code. The Mux1 is used to switch between A(t) and B(t) through the selection of CLK signal. Both A(t) and B(t) are realized by equation (2) and (3), respectively. The determination that which coding technique is adopted depends on the Mode

selection signal of the Mux2, where the Mode = 0 is for FM0 code, and the Mode = 1 is for Manchester code. To understand the concept of the hardware utilization, the Hardware Utilization Rate (HUR) is defined as

$$\text{HUR} = \frac{\text{Active Components}}{\text{Total Components}} \times 100\% \quad \text{---- (5)}$$

The component is defined as the hardware to perform a specific logic function such as NOT, OR, AND, and flip-flop. The active component means the component that work for FM0 or Manchester encoding. The total components represent the number of components in the whole hardware architecture no matter which encoding technique is adopted. The HUR of FM0 and Manchester encoders is listed in Table 1.

Table 1 HUR of FM0 and Manchester encoding

Coding	Active components (transistor count) / Total components (transistor count)	HUR
FM0	6 (86) / 7 (98)	85.71%
Manchester	2 (26) / 7 (98)	28.57%
Average	4 (56) / 7 (98)	57.14%

For both encoding techniques, the total components are 7; including Mux2 to indicate which coding technique is activated. For FM0 encoding, the number of active components is 6, and the HUR of this encoding is 85.71%. For Manchester encoding, the number of active components is 2, comprising of Xor2 and Mux2, and its HUR is low and it is 28.57%. On average, this hardware architecture has a poor HUR of 57.14%, and almost half of the total components are wasted. The transistor count for the hardware architecture without the SOLS technique is 98, where for FM0 encoding the transistor count is 86 and for Manchester coding is 26. On average, only 56 transistors can be reused, so the hardware utilization is not so efficient.

IV. VLSI Architecture Design of FM0 And Manchester Encoder Using SOLS Technique

The purpose of SOLS technique is to design a fully reused VLSI architecture of FM0 and Manchester encodings. The SOLS technique is classified into two parts. One is area-compact retiming and the other is balance logic-operation sharing.

4.1 Area-Compact Retiming

The FM0 logic in Fig.7 after area-compact retiming is shown in Fig.8. The logic for A(t) and the logic for B(t) are the Boolean functions to derive A(t) and B(t), here the X is omitted for a simple representation. For FM0, the state code of each state is stored into DFF_A and DFF_B. According to (2) and (3), the transition of state code only depends on B(t-1) instead of both A(t-1) and B(t-1). Thus, the FM0 encoding method just requires only a single 1-bit flip-flop to store the B(t-1). If the DFF_A is directly removed, non-synchronization between A(t) and B(t) causes the logic-fault in FM0 code. To avoid this logic-fault, the DFF_B is relocated to the right and it is placed after the Mux1, as shown in Fig. 8, where DFF_B is assumed to be positive-edge triggered.

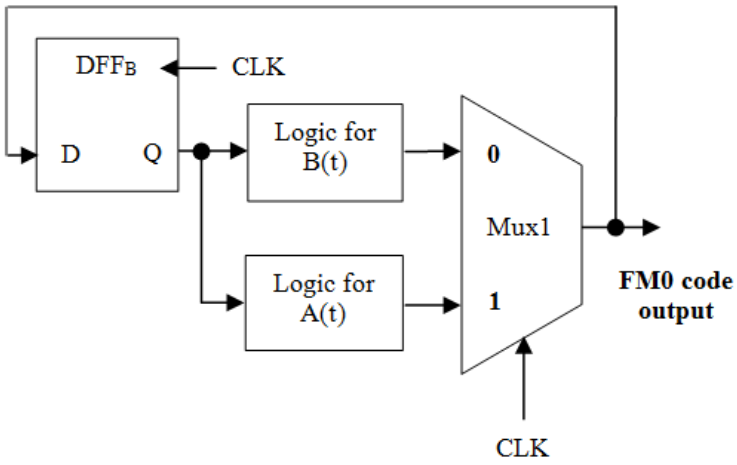

Fig. 8 FM0 encoding with area-compact retiming

The FM0 code is alternatively switched between logic for A(t) and logic for B(t) through the CLK, which is control signal of the Mux1. The transistor count of the FM0 encoding architecture realized with logic-family of static CMOS, without area-compact retiming is 72, and that of with area-compact retiming is 50. Thus the area-compact retiming technique implemented on FM0 encoder reduces 22 transistors.

4.2 Balance Logic-Operation Sharing

The Manchester encoding can be derived from Boolean function, X ⊕ CLK, and it is also equivalent to

$$X \oplus CLK = X\,\overline{CLK} + \bar{X}\,CLK \qquad \text{---- (6)}$$

The equation (6) can be realized by the multiplexer. It is quite similar to the Boolean function of FM0 encoding in (4). By comparing the equations (4) and (6) the FM0 and Manchester logics have a common point of resource i.e., multiplexer logic with CLK as the selection signal. The concept of balance logic-operation sharing is to integrate the $\bar{X}$ into A(t) and X into B(t), respectively. The A(t) can be derived by inverting the B(t-1), and $\bar{X}$ is obtained by inverting X. The logic for $A(t)/\bar{X}$ can share the same inverter, and then a multiplexer is placed before the inverter which is used to switch between the operands of B(t-1) and X. The mode signal indicates that either FM0 or Manchester is adopted. The similar concept can also be applied to the logic for$B(t)/X$. Nevertheless, this architecture exhibits a drawback that the XOR logic component is only dedicated for FM0 encoding, and is not shared with Manchester encoding. As a result, the HUR of this architecture is certainly limited. The X can also be interpreted as X xor 0, and thereby XOR operation can be shared with both codings. In the logic for B(t)/X, multiplexer is responsible for switching between B(t-1) and logic-0. This architecture shares XOR for both B(t) and X and thereby increases the HUR. Furthermore, multiplexer can be integrated into the relocated DFF_B.

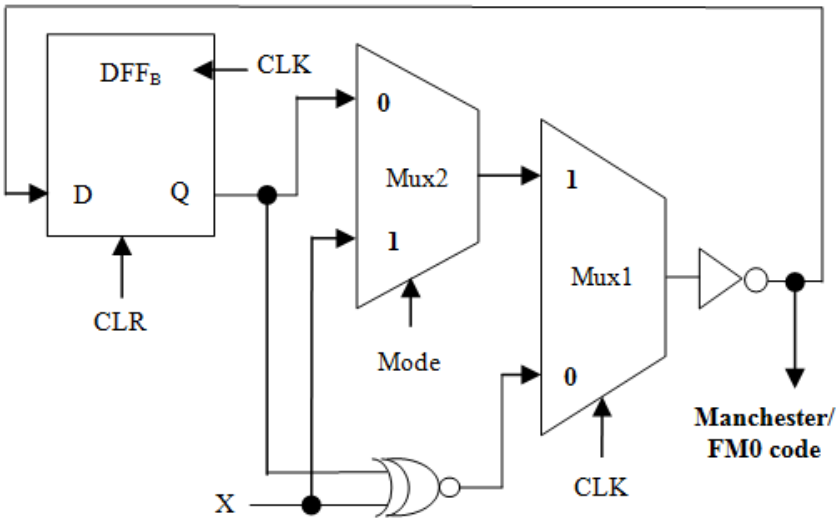

Fig. 9 Architecture of FM0 and Manchester encoding using SOLS technique

The proposed hardware architecture of FM0 and Manchester encoders using SOLS technique is shown in Fig. 9. The CLR is the clear signal for resetting the content of DFF_B to logic-0. The output state of DFF_B can be set to zero by activating the CLR signal for Manchester encoding. When the FM0 coding method is adopted, the CLR signal is disabled, and the B(t-1) can be derived from DFF_B.

The adoption of either FM0 coding or Manchester coding depends on Mode and CLR signals. In addition, the CLR further has another individual function of a hardware initialization. If the CLR signal is simply derived from the Mode signal without assigning an individual CLR control signal, this leads to a discrepancy between the coding mode selection and the hardware initialization. To avoid this discrepancy, both Mode and CLR signals are assumed to be separately assigned to this design. Whether FM0 or Manchester coding method is adopted, no logic component of the proposed architecture is wasted. Each and every component is active in both FM0 and Manchester encoding techniques. Hence, the HUR of the proposed hardware architecture is greatly improved.

V. Results

Tool: Xilinx ISE

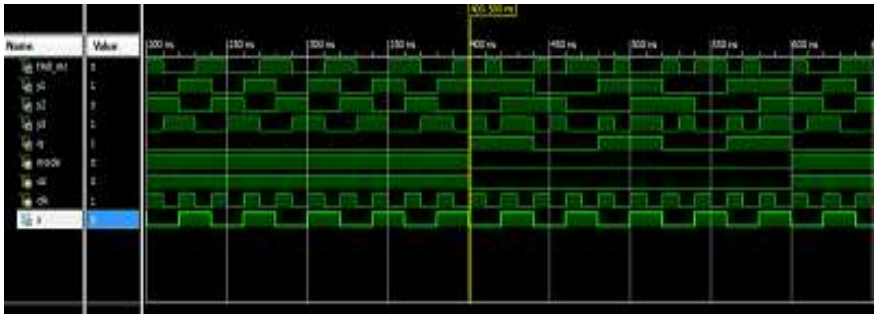

Fig. 10 Output waveform

The result of the proposed work implemented in Xilinx ISE is shown in Fig. 10. Here data signal X, clock signal CLK, clear signal CLR and mode are the input signals. The output is either FM0 or Manchester code depending on the mode selection signal.

Tool: Cadence Virtuoso

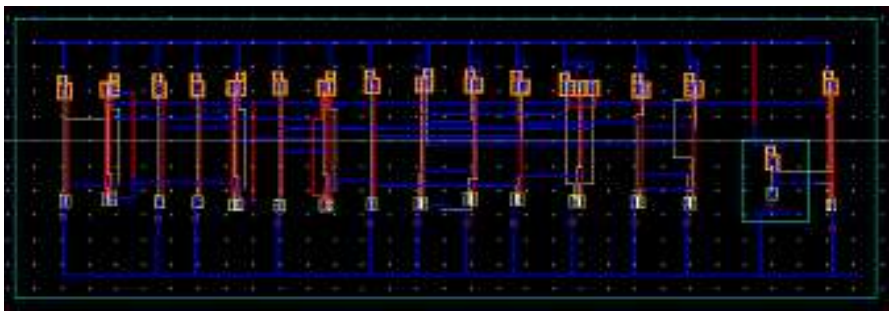

Fig. 11 Layout view

The performance evaluation of the designed architecture is shown in below figure.

Coding	Active components (transistor count) / Total components (transistor count)	HUR
FM0	5 (52) / 5 (52)	100%
Manchester	5 (52) / 5 (52)	100%
Average	5 (52) / 5 (52)	100%

Fig. 12 Performance evaluation with SOLS technique

VI. Conclusion

The coding-diversity between FM0 and Manchester encodings causes the limitation on hardware utilization of the architecture design. The limitation on hardware utilization of FM0 encoding and Manchester encoding is discussed in detail. In this paper, SOLS technique based reused VLSI architecture for both FM0 and Manchester encoding is proposed. The SOLS technique eliminates the limitation on hardware utilization by two core methods namely area-compact retiming and balance logic-operation sharing. The area-compact retiming relocates the hardware resources to reduce 22 transistors. The balance logic-operation sharing combines FM0

and Manchester encodings with the identical logic components. In overall total 46 transistors are reduced in this design. This paper is realized in the cadence virtuoso tool with 180-nm CMOS technology with outstanding device efficiency. The maximum operation frequency is 900 MHz and 2 GHz for FM0 and Manchester encodings, respectively. The power consumption is 197.08 μW at 2 GHz for Manchester encoding and 196.9 μW at 900 MHz for FM0 encoding. The encoding capability of this paper can fully support the DSRC standards of Europe, America and Japan. The work done in this paper is used to develop a fully reused VLSI architecture, and it also reduces the power consumption when compared with the previous existing works.

References

[1]. Yu-Hsuan Lee, Cheng-Wei Pan, "Fully Reused VLSI Architecture of FM0/Manchester Encoding Using SOLS Technique for DSRC Applications," IEEE VLSI systems, vol. 23, no. 1, pp. 18-29, Jan. 2015.

[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.

[7]. Y.-C. Hung, M.-M. Kuo, C.-K. Tung, and S.-H. Shieh, "High-speed CMOS chip design for Manchester and Miller encoder," in Proc. Intell. Inf. Hiding Multimedia Signal Process., Sep. 2009, pp. 538–541.

[8]. M. A. Khan, M. Sharma, and P. R. Brahmanandha, "FSM based Manchester encoder for UHF RFID tag emulator," in Proc. Int. Conf. Comput., Commun. Netw., Dec. 2008, pp. 1–6.

[9]. M. A. Khan, M. Sharma, and P. R. Brahmanandha, "FSM based FM0 and Miller encoder for UHF RFID tag emulator," in Proc. IEEE Adv. Comput. Conf., Mar. 2009, pp. 1317–1322.

[10]. J.-H. Deng, F.-C. Hsiao, and Y.-H. Lin, "Top down design of joint MODEM and CODEC detection schemes for DSRC coded-FSK systems over high mobility fading channels," in Proc. Adv. Commun. Technol. Jan. 2013, pp. 98–103.

[11]. I.-M. Liu, T.-H. Liu, H. Zhou, and A. Aziz, "Simultaneous PTL buffer insertion and sizing for minimizing Elmore delay," in Proc. Int. Workshop Logic Synth., May 1998, pp. 162–168.

[12]. H. Zhou and A. Aziz, "Buffer minimization in pass transistor logic," IEEE Trans. Comput. Aided Des. Integr. Circuits Syst., vol. 20, no. 5, pp. 693–697, May 2001.

[13]. N. H. E. Weste and K. Eshraghian, Principles of CMOS VLSI Design: A Systems Perspective, 2nd ed., Upper Saddle River, NJ, USA: Pearson Educ. Ltd., 1993, pp. 98–103.

[14]. Sung-Mo Kang, Yusuf Leblebici, CMOS Digital Integrated Circuits, 2nd ed., McGraw-Hill Education, 2003.