International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-3, January 2020

Design of Completion Detectors in Asynchronous Communication System

Norhuzaimin Julai , Shamsiah Suhaili , Yonis M Yonis Buswig

Abstract. In digital design, there are two types of design, synchronous design and asynchronous design. In synchronous design, global clock is one of the main system that consume a lot of power. The power in synchronous design is consumed by clock even if there is no data processing take place. The asynchronous design that depends on data is clockless and as far as the power is concerned, asynchronous design does not consume much power compared with synchronous design and this really make asynchronus design the preffered choice for low power consumption. Besides having low power consumption, there are many advantages of aynchronous design compared with synchronous design. This paper proposed new dual rail completion detector (CD), 3-6 CD, 2-7 CD and 1-4 CD for on-chip communication that are used widely in an asynchronous communication system. The design of CD is based on the principle of sum adder. The circuit is designed by using Altera Quartus II CAD tools, synthesis and implementation process is executed to check the syntax error of the design. The design proved to be successful by using asynchronous on-chip communication in the simulation.

Keywords:Asynchronous, converter, completion detector, Quartus II

I. INTRODUCTION

In digital design, there are two types of design, synchronous design and asynchronous design. In synchronous design, global clock is one of the main system that consume a lot of power and therefore need to be removed for better power efficiency [1] [2] [3]. The power in synchronous design is consumed by clock even if there is no data processing take place. The asynchronous design that depends on data is clockless and as far as the power is concerned, asynchronous design does not consume much power compared with synchronous design and this really make asynchronous design the preferred choice for low power consumption. Besides having low power consumption, there are many advantages of asynchronous design compared with synchronous design. [4], [5].

(i) Absence of clock skew as the arrival time of the clock signal is depended on the data.

(ii) Better than worst case performance because for asynchronous design, the average case performance is the most likely case due to the data-dependent data flow and functional unit that exhibit data-dependent delay.

(iii) Automatic adaption to physical properties since in asynchronous design, it is depended on the data as a clock, the above factors are automatically adjusted and hence the designer need not to worry the functionality of the circuit even under the worst case scenario.

(iv) Reduced electromagnetic interference as the activities of the circuit is very much independent from one to another. Despite all the advantages of asynchronous design over synchro-nous design, some of the disadvantages include

(v) Deadlock which is common situation in asynchronous design that the system faces due to the incorrect circuit design, token mismatch and also by arbitration.

(vi) The arrival of the data is detected by a completion detection (CD). Since the asynchronous design is data dependent and detected by CD, the design of CD is a trivial task and in some cases, it is a complex task.

One of the components in the above system is particularly important to substitute the clock system is the completion detector (CD). The dual rail CD is the simplest circuit which is basically an XOR gate to detect 1-bit of dual rail code. However, for other codes such as 3-6 code and 2-7 code, the CD circuits are quite complex. The design of CD to detect the arrival of data is an important and non-trivial problem [6, 7]. Other than dual rail CD, author in [8] proposed a method called as Delay-Insensitive Minterm Synthesis (DIMS) to tackle the problem. By using this method, for m to n code, the code is broken into smaller codes and concatenated. Part of the code is divided into control group and body of the code.

II. METHODOLOGY

The purpose of the communication system is converting single rail to dual rail and back to single rail data as shown by Figure 1. Dual rail encoding is necessary for long on-chip interconnect since it is delay insensitive which refers to the data is transmitted correctly regardless of the delay in the interconnecting wires. It uses two wires to represent 1 bit of information. The intermediate code in the communication is 3-6 code. Even though the dual rail encoding is suitable for long on-chip interconnect communication, it suffers the lowest capacity (bits/wire) compared with other code. Hence, this makes it more costly. Table 1 compares some of the codes with information regarding the number of wires, transitions, capacities and the complexities.

Revised Manuscript Received on January 30, 2020.
***** Correspondence Author
Norhuzaimin Julai*, Faculty of Engineering, Universiti Malaysia Sarawak(UNIMAS), 94300 Kota Samarahan, Sarawak, Malaysia,
Shamsiah Suhaili, Faculty of Engineering, Universiti Malaysia Sarawak(UNIMAS), 94300 Kota Samarahan, Sarawak, Malaysia,
Yonis M.Yonis Buswig, Faculty of Engineering, Universiti Malaysia Sarawak(UNIMAS), 94300 Kota Samarahan, Sarawak, Malaysia,

From Table 1, it is obvious that the 3-6 code has the highest capacity compared with other codes. Besides, the 3-6 code has double wire capacity compared with dual rail. The same goes with the number of transitions. The numbers of transitions are related to the dynamic power dissipated by circuit.

Compared with 3-6 codes, the numbers of transitions for dual rail is twice and 2-7 code has the lowest numbers of transitions. However, the disadvantage of 3-6 code is the complexity. The complexity refers to the number of standard gates and C-elements (C) to build completion detector (CD). Obviously, the CD for 3-6 code requires the most number of gates and C-elements.

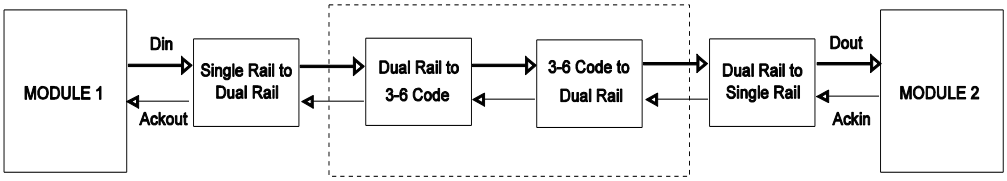

Figure 1: Asynchronous on-chip communication

Table 1: Code comparison in terms of wires, transition and capacity [9]

Codes	Wires	Transitions (Data + Ack)	Capacities (bit/wire)	Complexities
Dual rail	12	4x4=16	4/12=0.3	4+3C
1-4	10	4x2=8	4/10=0.4	6+1C
2-7	8	6x1=6	4/8=0.5	6+3C
3-6	7	8x1=8	4/7=0.6	10+3C

Due to the number of wires, numbers of transitions and the capacities, it is desirable to convert dual rail encoding to 3-6 codes for long on-chip communication as discussed above.

Figure 2(a) and (b) show the dual rail to 3-6 converter and 3-6 to dual rail converter respectively

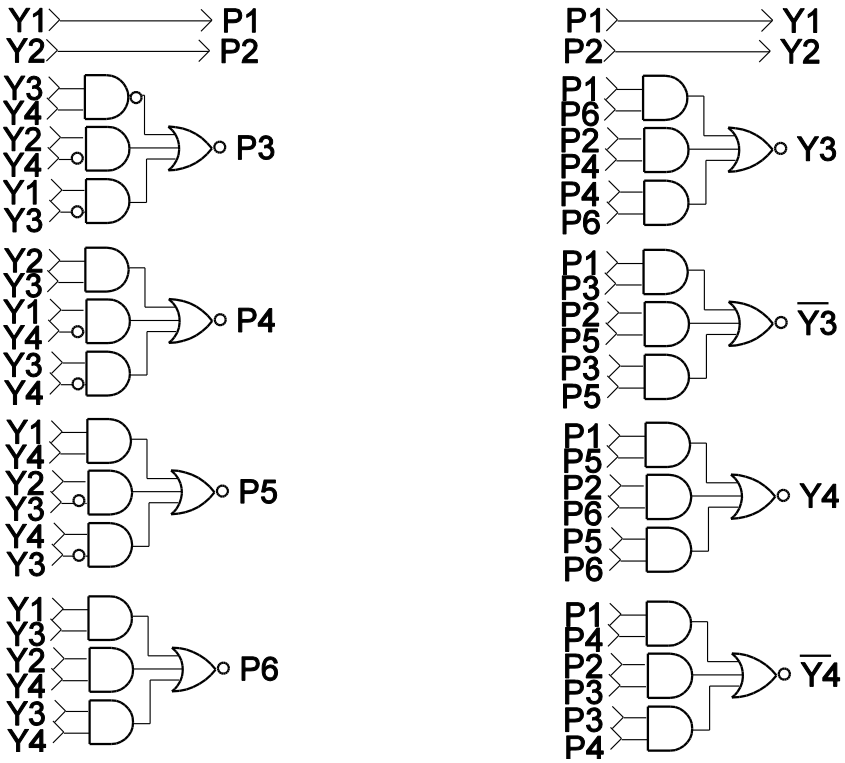

Figure 2

(a) : Dual rail to 3-6 converter **(b) : 3-6 to dual rail converter**

The conversion of the dual rail to 3-6 code back to dual rail code is shown by Figure 3 with the description in Table 2. Each of the pipelines is enabled from subsequent pipeline stage. The enable signal is obtained from the completion detector (CD) of the subsequent pipeline stage before it is inverted into second port of C-element of the previous pipeline. The on-chip communication consists of three pipelines.

The first register pipeline buffers the input of dual rail data and the CD detects the presence of dual rail data at the output of the first pipelines. This provides acknowledge signal to the preceding pipeline. The second pipeline outputs the 3-6 code from dual rail to 3-6 converter and the CD detects the presence of 3-6 acknowledge signal. The acknowledge signal is used to enable and disable the first pipeline. The third pipeline outputs the dual rail data from 3-6 to dual rail converter and allowing the acknowledge signal from CD that detects the presence of dual rail data.

The CD signal is used to enable and disable the second pipeline. It is desired to build the latch from pipeline by using EDCD [10] latch and the effect of using EDCD latch in the event of SEU hit to one of latch in the pipeline is discussed below

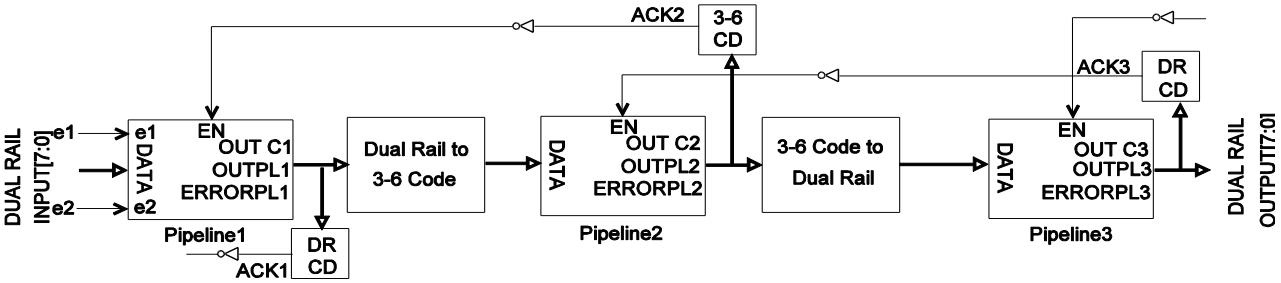

Figure 3: Dual rail to 3-6 and back to dual rail conversion

Table 2: Interface of dual rail to 3-6 and back to dual rail conversion

Name	Direction	Description
ACK1/ACK2/ACK3	I	Acknowledge signal for the input data
Data[7:0]	I	Input of dual rail data
e1/e2	I	Input of injected error
OUTC1/OUTC2/OUTC3	O	Output of C-element for Pipeline
OUTPL1/OUTPL2/OUTPL3	O	Output of the Pipeline
ERRORPL1/ERRORPL2/ERRORPL3	O	Error detection for Pipeline
A[7:0]/B[7:0]	I	Input of adder
3-6 CD	O	3-6 code completion detector
DR CD	O	Dual rail code completion detector
EN	I	Enable port for pipeline

Table 3 provides the 3-6 code and the corresponding dual rail code. It consists of 16 symbols according to the equation Encoding capacity =6!/[(6-3)!*3!]

The proposed 3-6 CD is shown by Figure 4. The proposed CD is based on the number of '1' bit presence in the code. The codes are arranged systematically according to the control group and body. For example, the 3-6 code is broken into 2-4 code and dual rail code. Similarly, for 2-7 code, it is broken to 1-3 code and 1-4 code. The half adder is proposed to be used in designing CD. The code is divided into three of two-bit. The XOR gate detects '01' or '10' bits. The AND gate detects '11' bit. Suppose the 3-6 code is represented by X5X4X3X2X1X0, with X5 is the most significant bit (MSB) and X0 is the least significant bit (LSB). From Table 3, it is observed that the code can be broken into three parts as shown by Table 4.

Table 3: 3-6 code and the corresponding dual rail code

3-6 code	Dual Rail Code
111000	10101010
101100	01101010
110100	10011010
011100	01011010
101010	10100110
001110	01100110
110010	10010110
010110	01010110
110001	10101001
101001	01101001
010101	10011001
001101	01011001
100011	10100101
001011	01100101
010011	10010101
000111	01010101

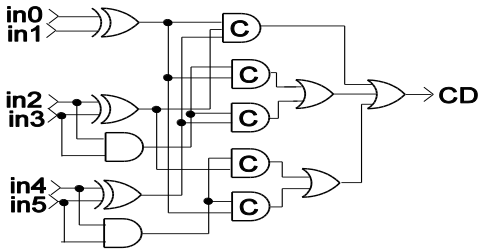

Figure 4 : Proposed 3-6 code detector

Table 4: Component of 3-6 Code		
X_5X_4	X_3X_2	X_1X_0
00	00	00
01	01	01
10	10	10
11	11	Not present

For X_5X_4 and X_3X_2 the XOR and AND gates are used to detect '01/10' and '11' bits respectively. For X_1X_0 bit '11' is not present and therefore only XOR gate is necessary to detect '01/10' bit. There are five probabilities that 3-6 code are detected and is shown by Table 5. For the case (a), one three-input AND gate is needed to receive inputs from three two-input XOR gate. For case (b), (c), (d) and (e) four two-input of AND gates are needed. In order to ensure delay insensitive (DI) design, the AND gates are replaced with C-elements. The full valid set of code is obtained by ORing the product of code. Figure 5 and Figure 6 show the proposed 2-7 CD and 2-4 CD derived with similar method as before

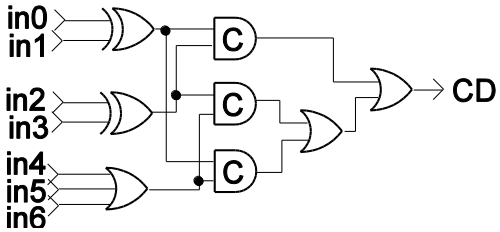

Figure 5: Proposed 2-7 code detector

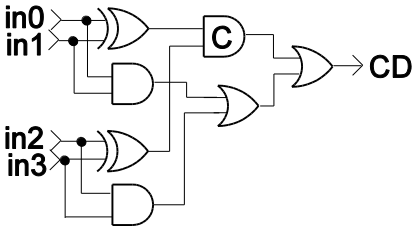

Figure 6: Proposed 1-4 code detector

Table 5: Probabilities of 3-6 Code			
	X_5X_4	X_3X_2	X_1X_0
(a)	01 10	01 10	01 10
(b)	11 11	01 10	00 00
(c)	11 11	00 00	01 10
(d)	00 00	11 11	01 10
(e)	01 10	11 11	00 00

III. RESULT AND DISCUSSION

The system was designed by using Altera Quartus II CAD tools in which the and implementation process is executed to check the syntax error of the design.

When the design is successful, the simulation process will take place.

The waveform simulation is shown by Figure 7. Error is injected to Pipeline1. The four-bit dual rail input starts as 'A9' (10101001) and alternate with spacer. At time T1, no error is injected and the dual rail CD detects the presence of the first valid symbol. An acknowledge signal (*ACK1*) is sent to the preceding pipeline. The symbol is converted to 3-6 code by converter and a symbol '31' appears at pipeline2. The 3-6 code is detected by CD and send an acknowledge signal (*ACK2*) to Pipeline1 causing Pipeline1 to reset.

A valid 3-6 code is converted to dual rail symbol and a symbol "A9" appears at the output of pipeline3. A valid dual rail symbol is detected at the output of pipeline3 and send an acknowledge signal (*ACK3*) causing Pipeline2 to reset. At time T2, an error is injected at the Pipeline1 and causing the MSB of data to change from 1-0. A new symbol appears at Pipeline1 as "29" (0010 1001) instead of "A9" (1010 1001). This data is not recognized by CD as a valid data and causing no acknowledge signal (*ACK1*) is sent as shown by (a). The corrupted signal propagates to the 3-6 converter and invalid signal "39" (0101 1001) appears at the output of Pipeline2. No acknowledge signal (*ACK2*) is sent by 3-6 CD. The signal propagates to Pipeline3 and a corrupted symbol "29" (0010 1001) appears at the output of Pipeline3 as shown by (b) with no acknowledge signal (*ACK3*) is sent as for the previous case

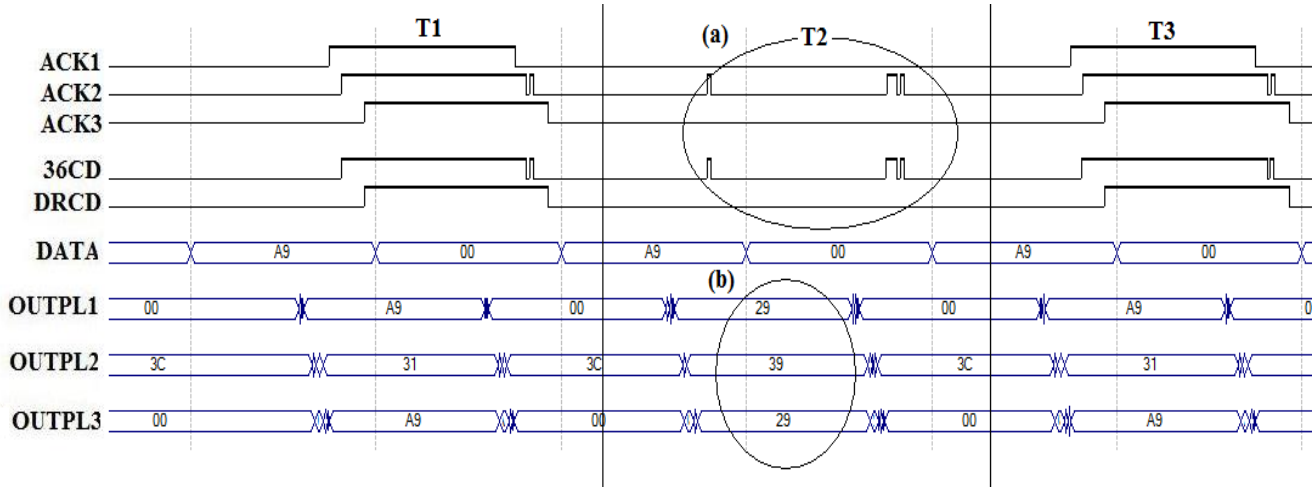

Figure 7: Waveform simulation for fault free condition and error injection on dual rail latch that does not have error correction capability

The error detection and correction for dual rail (EDCD) [10] latches are used to demonstrate functionality of the latches in the event of SEU hit on the pipeline causing the data to be corrupted. The waveform simulation is shown by Figure 8. The four-bit dual rail input starts as 'A9' (10101001) and alternate with spacer as in the previous case. At time T1, no error is injected and the dual rail CD detects the presence of the first valid symbol. An acknowledge signal (*ACK1*) is sent to the preceding pipeline. Error signal at Pipeline1 (*ERRORPL1*) is '0' indicates no error is detected at Pipeline1. The symbol is converted to 3-6 code by converter and a symbol '31' (0011 0001) appears at pipeline2. Error signal at Pipeline2 (*ERRORPL2*) is '0' indicates no error is detected at Pipeline2. The 3-6 code is detected by CD and send an acknowledge signal (*ACK2*) to Pipeline1 causing Pipeline1 to reset. A valid 3-6 code is converted to dual rail symbol and a symbol "A9" (1010 1001) appear at the output of Pipeline3. Error signal at Pipeline3 (*ERRORPL3*) is '0' indicates no error is detected at Pipeline3.

A valid dual rail symbol is detected at the output of Pipeline3 and send an acknowledge (*ACK3*) signal causing Pipeline2 to reset. At time T2, an error is injected (a) at the Pipeline1 and causing the second MSB of data to change from 0-1.

A new symbol appears at output of C-element (*OUTC1*) as "E9" (1110 1001) instead of "A9" (1010 1001) (e). This is an example of 1-1 error. At this instance, error signal at Pipeline1 is produced (*ERRORPL1*) to indicate that error is detected on Pipeline1 (c).

As a result of error is detected, the MUX is activated to replace the corrupted values with the true value. The true value of "A9" (1010 1001) appears at *OUTPL1* (g). With the correction mechanism in place, the dual rail CD is able to detect the presence of valid dual rail and acknowledge signal (*ACK1*) is sent to the preceding pipeline. The corrected dual rail values are converted to 3-6 code and a symbol of "31" (0011 0001) appears the output of pipeline2 (g).

The 3-6 CD recognize it as a valid data and sent an acknowledge signal (*ACK2*) to Pipeline1 and causing it to reset. No error signal is produced (*ERRORPL2*) at Pipeline2 to indicate that no error is detected on Pipeline2. The data is converted back to dual rail and a symbol of "A9" (1010 1001) appears at the output of Pipeline3 (g). Similarly no error is produced (*ERRORPL3*) at Pipeline3 to indicate that no error is detected on Pipeline3.At time T3 an error is injected (b) at the Pipeline1 and causing the MSB of data to change from 1-0. A new symbol appears at output of C-element (*OUTC1*) as "29" (0010 1001) instead of "A9" (1010 1001) (f). This is an example of 0-0 error. At this instance, error signal at pipeline1 is produced (*ERRORPL1*) to indicate that error is detected on Pipeline1 (d). As a result of error is detected, the MUX is activated to replace the corrupted values with the true value. The data proceed to converters, and Pipeline2 and Ppipeline3 with the same protocols as described above (h).

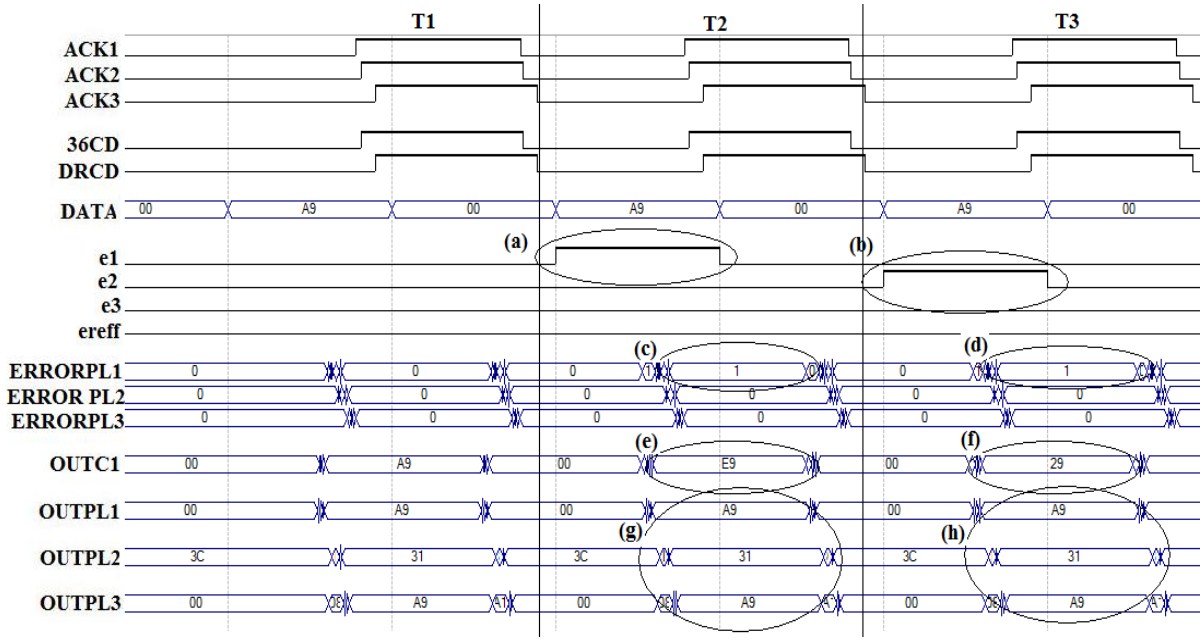

Figure 8: Waveform simulation for fault free condition and error injection on EDCD latch

CONCLUSION

This paper has presented the design of completion detector which is one of the most important component in the asynchronous communication system. For this purpose, Altera Quartus II software is used to obtain the waveform of the simulation. The design is based on the number of '1' s presence in the design. We obtained the output of on-chip communication by using EDCD latch which has been proven to counter SEU error. This paper proposed new dual rail completion detector (CD), 3-6 CD, 2-7 CD and 1-4 CD for on-chip communication that are used widely in an asynchronous communication system

ACKNOWLEDGMENT

This work is supported by OSAKA Gas Foundation Of International Cultural Exchange grant

REFERENCES

1. Yebin Shi, Steve. B. Furber, Jim Garside and Luis A. Plana, "Fault Tolerant Delay Insensitive Inter-Chip communication," IEEE Symposium on Asynchronous Circuits and Systems, 2009, pp. 77-84
2. J. Bainbridge, and S. B. Furber, "CHAIN: A delay-insensitive chip area Interconnect," IEEE Micro., 2002, 22, (5), pp. 16–23
3. A. Lines, "Asynchronous interconnect for synchronous SoC design," IEEE Micro, 2004, 24, (1), pp. 32–41.
4. S. Hauck, "Asynchronous design methodologies: An Overview", Proceeding of the IEEE, Vol 8, No1, pp 69-93, Jan 1995
5. Kimberly D Emerson, "Asynchronous Design-an Interesting Alternative", 10th International Conference on VLSI Design, pp 318-321, Jan 1997
6. Akella, V., Vaidya, N. H., Redinbo and G. R, *Limitations of VLSI Implementations of Delay-Insensitive Codes*, Proc.26th Int. Symp. On Fault-Tolerant Computing, Sendai, June 1996, pp. 208-217.
7. Piestrak, S.J, *Membership Test Logic for Delay-Insensitive Codes*, Proc Async '98, San Diego, California, April 1998, pp194-204
8. Bainbridge, W.J. and S.B. Furber, *Delay Insensitive System-on-chip Interconnect using 1-of-4 data encoding*, Seventh International Symposium on Asynchronous Circuits and Systems , ASYNC, 2001
9. Yebin Shi, Fault-Tolerant Delay-Insensitive Communication, Thesis submitted to the Faculty of Engineering and Physical Science, University of Manchester, 2010
10. N Julai, *Soft Error Mitigation on Dual Rail Latch*, Applied Mechanics and Material, April 2016, Vol 833 ppv119-1

AUTHORS PROFIL

Dr. Norhuzaimin Julai received B.Sc(Electrical and Computer Engineering) degree from Ohio State University USA in 1999 and M.Sc. degree in Advanced Electronics Engineering from Warwick University UK in 2004. In 2015, he was awarded his Ph.D. degree in Electrical & Electronic Engineering from Newcastle University. He is with the Department of Electrical and Electronics Engineering, Faculty of Engineering, Universiti Malaysia Sarawak. His research area of interest is in the soft error in integrated circuit (IC) He is a Member with Board of Engineers Malaysia (BEM) and Chartered Engineer from Energy Institute UK

Shamsiah binti Suhaili received the BEng. (Hons) and MSc. degrees in Electrical & Electronic Engineering from Universiti Sains Malaysia (USM) in 2001 and 2005, respectively. She now with Faculty of Engineering, Universiti Malaysia Sarawak. Her research interests include digital design, cryptography and application of FPGA design.

Dr. Yonis.M.Yonis Buswig was born in Benghazi, Libya, in 1984. He received his B.Eng. degree from Omar Al-Mukhtar University, Libya, in 2008. The M.Sc. degree in electrical and electronics engineering from Tun Hussein Onn University of Malaysia (UTHM), Johor, Malaysia, in 2011. In 2015, he was awarded his Ph.D. degree from Department of Power Engineering, Faculty of Electrical Engineering, Tun Hussein Onn University of Malaysia (UTHM). Currently, He is a Lecturer in the Electrical and Electronic Engineering Department, Faculty of Engineering, Universiti Malaysia Sarawak. Yonis has authored and co-authored a number of well-recognized journals and conference papers and has been a regular reviewer for IEEE, Wiley and other journals. His current research interests include the area of Power Electronics, Renewable energy technology, and Motor Drives Control. He is a Member in Board of Engineers Malaysia (BEM).