DESIGN OF EFFICIENT GOLAY ENCODER FOR DEEP SPACE MISSIONS

[1]Kiran AB, [2]Sandesh Kumar BV
[1]PG Student, [2]Asst. Professor,
[1],Dept. Of Electronics and Communication Engineering
[1]SJB Institute Of Technology, Bengaluru, India

Abstract*— *Linear feedback register (LFSR) based CRC and hardware architecture based on CRC were the two approaches we had for generation of binary golay code (23,12) . in this paper lets go on with new and different approach for the same to overcome the disadvantages of binary golay code (23,12). This new design has been verified to have efficient hardware to generate extended golay code(24,12) with high speed, lesser area , low latency, low power architecture.

Index Terms*— *Binary golay code (23, 12), Linear feedback shift register (LFSR), Cyclic redundancy check (CRC) and Extended Golay code(24,12)

I. INTRODUCTION

The reliable transmission of digital data over noisy interrupted communication channel lies in error detection and its correction which is the vital part of information theory. A message transmitted from a source can get distorted or corrupted in a wireless communication due to noise. So here the error is caused by noise its only a detection whereas its correction includes complete restoration of original data which was transmitted. The remnant bits are called check bits which should be joined together at the time of transmission. For detection and error correction we use FEC codes which is of two types that is Block codes and convolution codes.

Various types are block codes are available. Binary golay code , the known superior block code(23,12,7) represented as G23 which was proposed by M.J.E Golay in 1949 . it's a linear error correcting code and it's a part of non trivial perfect codes. Incorporation of partly bit to binary golay codes results the rate 1/2 with self dual extended golay code (24, 12, 28) that is G 24.this G24 told to have numerous applications such as in ultrasound imaging , coded excitation imaging for laser, voyager missions of JPL NASA. Block code notation (n,k,d) is described by m=length, message with length as k, minimum hamming distance between two code words will be d. C is code word which detects and corrects an error upto d¬1 and (d¬1)/2 respectively.

Architecture of this paper is as explained below. Accord to literature survey Section 2 and section 3 deals with proposed architecture for G23, G24 and its simulation results. Section 4 results with conclusions.

II. LITERATURE SURVEY

If 24 bit codeword is used in place of 12 bit in a extended golay code it can detect 7 error and corrects 3 errors of it. that is if, G=[I,B] is a 12*24 matrix,where I,B are the 12*12 identity matrix and matrix over GF. C is a linear code with G generation matrix is known as extended golay code (G24). The specification of this extended golay code G24 with generation matrix G and B include

- The 12*12 matrix B follows BT = B, BBT = I and B2 = I.
- One more generator matrix for G24 is [B,I].
- G24 is self-dual three error correcting code.
- Parity check matrix for G24 is a 24x12 matrix. H = [B,I]T or [I,B]T

Puncturing extended golay codeG24 gives binary golay code. If B'=12*11 matrix which is obtained by deleting the last column of 12*12 matrixB. Binary golay code G23 is obtained by any linear code of 12*23 generation matrix G=[I,B']. it is available in many encoding ways. Though [3] [5] deals with different algorithms, its not a efficient hardware implementation algorithm. The hardware implementation is done based on LFSR[6] [9] methods but or not suitable as of high latency. Compared to previous years encoding architectures[10] ¬–[19] the recent is [20] proposed hardware implementation is based on CRC. It briefs out two vital encoding methods they are

1) *LFSR Based Golay Encoder*

Figire.1 demonstrates an encoder design for (23, 12) golay code with generator polynomial.

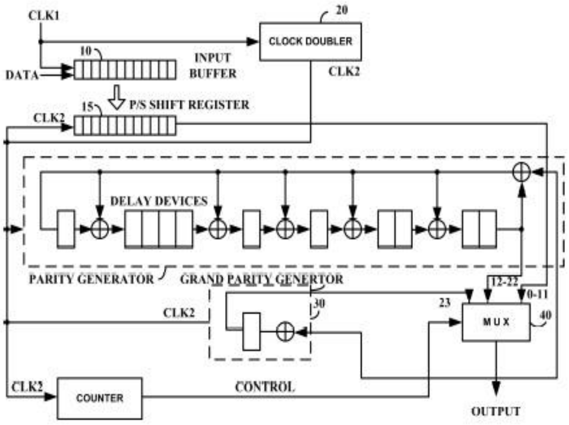


Fig.1. LFSR based golay encoder

g(x) = x11 + x9 + x7 + x6 + x5 + x1 + 1(AE3h)

The encoder comprises of a polynomial division enlist, (n−k)11 lag stages and (n) 23 codeword stages. The postpone stages are utilized for deferring the data bits.So, that data bits enter the codeword organizes in correct synchronization.

- Our data word i(x) = [ik−1, ik−2, i2, i1, i0]. The MSB ik−1i.e., i11 is the primary bit to go into the reg taken after by i10, i9, ...i2, i1, i0.
- After (k) 12 moves the data bits have entered the enroll and the rest of contain the rest of of i(x)/g(x). Yet, for deliberate codeword we require (xn−k.i(x))/g(x). This is accomplished by moving the reg an extra (n − k) 11 times. Thusly after (n−k+k) 23 moves, the reg stages contain the checkbits.
- Further (n−k) 11 movements are required to move the checkbits into the codeword stages. Extra two changes are utilized to redirect the bit stream when required.

2) Existing Architecture for Binary Golay Code Generation

Satyabrata sarangi and Swapna banerjee [20] proposed a design appeared in Fig. 2.

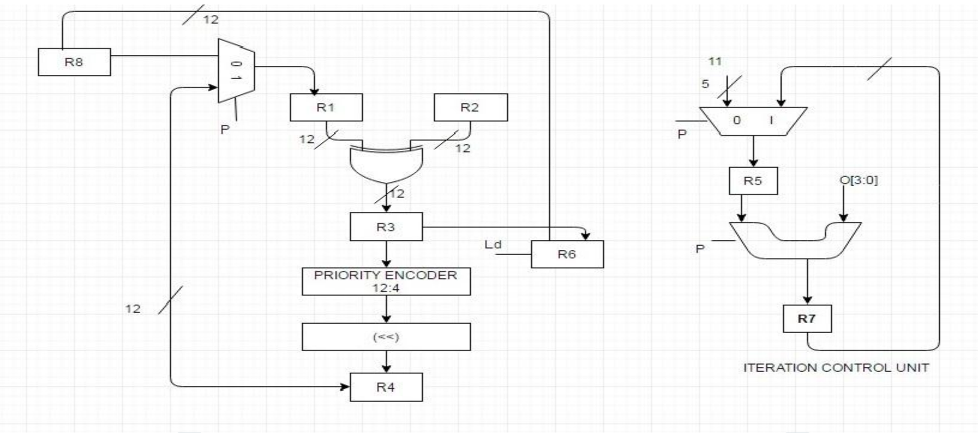
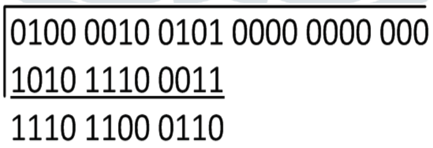


Fig.2. Existing architecture for binary Golay code

Initially the 12 bit message is put away in R8 and the characterstic polynomial AE3h in R2. The data of registers R1 and R2 are binary XORed and put away in enroll R3. A 12:4 priority encoder is utilized to discover the quantity of zeros present in the enroll R3 before 1. A cyclic shifter is utilized to left move the substance of enlist R3 by priority encoder yield times and the outcome is put away in R4. A 2x1 multiplexer is utilized to choose the underlying info or R4, relies on upon control flag P. Which is bit insightful OR of priority encoder yield o[3:0].Cycle control unit comprises a 2x1 multiplexer, used to select 11(no of zeros added to the info) or enroll R7. The other contribution to the subtractor is priority encoder yield. After last cycle, the yield of the subtractor will be zero. At the point when the control flag Ld winds up noticeably zero, R6 [22:11] is stacked with the data of R8 and R3 [10:0] individually. the control flag Ld = (!(R7[4])&(|R7)), where |(R7) speaks to bit astute OR operation. At long last, the data of R6 represents the encoded golay codeword. There are a few weaknesses in this design. They are

1) At the point when the information MSB is 0:

The current engineering does not bolster the message whose MSB is 0. Give us a chance to view the message as encoded is 425h. The priority encoder output is zero as there are no leading zeroes in the output. In this illustration, the control flag P is zero, accordingly the 2x1 multiplexer chooses input again rather than transitional outcome. This procedure proceeds without getting any yield. Find the example below in fig3.

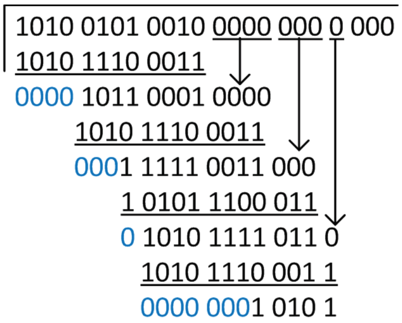


```
|0100 0010 0101 0000 0000 000
|1010 1110 0011
 1110 1100 0110
```

Fig.3: At the point when the information MSB is 0

2) When the priority encoder yield is more prominent than the required number of movements:

The long division handle proceeds until all the 11 annexed zeros are utilized. Give us a chance to consider the message to be encoded is A52h. consider the example below



```
|1010 0101 0010 0000 000 0 000
|1010 1110 0011
 0000 1011 0001 0000
      1010 1110 0011
      0001 1111 0011 000
         1 0101 1100 011
         0 1010 1111 011 0
           1010 1110 001 1
           0000 0001 010 1
```

Fig.4:the priority encoder yield is more prominent than the required number of movements

The leftover outcome gotten in the keep going phase of long division process is 0000 0001 0101(R4). As priority encoder yield is 7, there will be seven round movements on the lingering result. Ld winds up noticeably zero and the check bits produced will be 000 0110 0011. This isn’t right, when priority encoder yield is more noteworthy than required number of movements, the check bits ought to be created from R4 not at all like R3. As there are just three zeros remaining on right half of the message , just 3 round movements on middle of the road result (0000 1010 1000). The MSB is disregarded and the resultant check bits are 000 1010 1000.

3) The substance of the enroll R6 refreshes after generating binary golay code

As Ld ends up noticeably zero, the esteem put away in R6 gives the encoded extended golay code. Be that as it may, with the current design, the division procedure proceeds as Ld will be zero for next a few emphasess. With this, the data of the required R6 updates and brings about getting incorrectly encoded codeword. Recreation results are appeared in Fig. 5

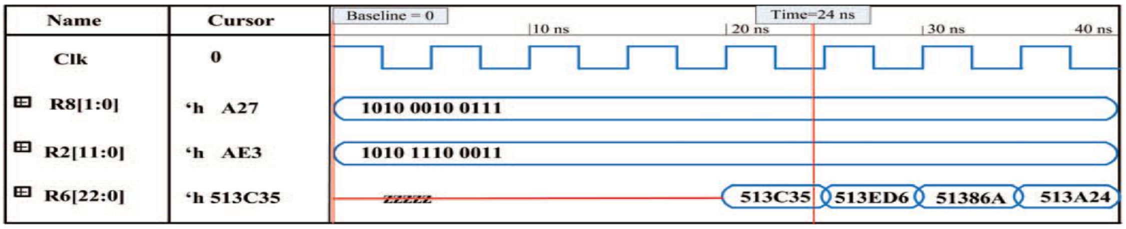


Fig.5.Simulation results with existing architecture

III. IMPLEMENTED ARCHITECTURE FOR BINARY GOLAY CODE AND EXTENDED GOLAY GENERATION

1) Proposed Architecture for Binary Golay Code(G23)

To correct the problems with the existing architecture a new architecture has been implemented in this paper shown in Fig. 6

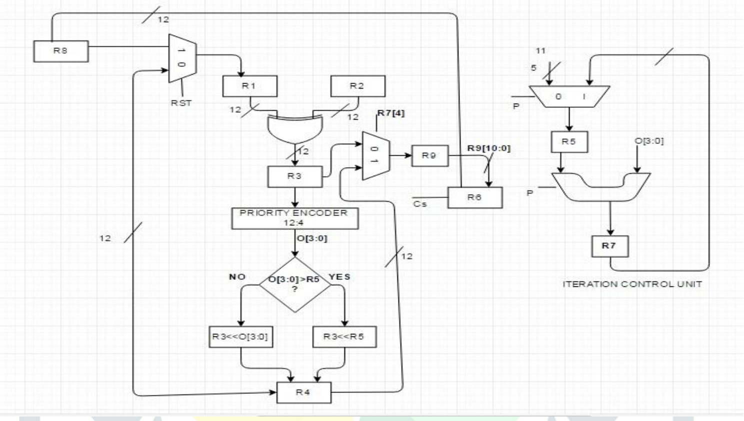


Fig.6: Proposed Architecture for Binary Golay Code(G23)

R8 and R2 registers are saved with the value of input message and characteristic polynomial separately. The data of the register R4 and register R8 are selected by using a 2*1 mux. At first the control signal "Rst" will be High and all through the polynomial division process it will be Low. XOR transformation occurs for modulo 2 subtraction, In each progression of long division process.

The quantity of 0's available before initial 1 in the Register R3 is found by using a 12:4 priority encoder . To find whether the priority encoder output is greater than the value stored in register R5 a condition block is used. On the off chance that the condition fulfills then the remaining outcome will be circularly left moved by R5 times else priority encoder yield times and the outcome is put away in register R4. As Rst is low, the data of R1 refreshes with data of R4. This procedure proceeds upto last cycle. The another control signal for 2:1 mux is R7[4],it will be High when the priority encoder yield is more prominent than required number of movements i.e, R5 (When higher number is subtracted from lower number, the resultant yields sign bit will be high which demonstrates a negative number.

Iteration control unit comprises of a 2:1 mux and a subtractor. At first reset will be high, with which R5 contains 11 and throughout the long division R5 gets refreshed with the data of R7 as reset is low. the control signal of the subtractor is P and another input is output of priority encoder. The Cs = ((Ld == 0)&(R7[4] == 1)&(R5[4] == 0)) controls the stacking of R8 and R9[10:0] into enroll R6, where Ld is (!(R7[4])&(|R7)). R6 is stacked when control flag is high, which implies end of the division procedure. At the point when Ld =0, R7[4]=1 and R5[4]=0, R8 substance are moved into R6[22:11] and R9[10:0] into R6[10:0].R6 contains the encoded golay code.

Give us a chance to view the message as encoded is 127h. Registers R8 127h and and R2 contain characterstic polynomial AE3h separately. Check bits by utilizing long division process is appeared in Fig.7

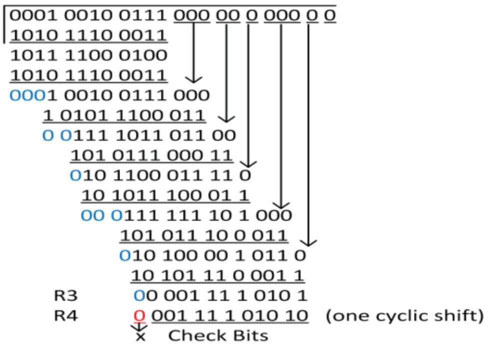

```
0001 0010 0111 000 00 0 000 0 0
1010 1110 0011
1011 1100 0100
1010 1110 0011
0001 0010 0111 000
   1 0101 1100 011
   0 0111 1011 011 00
      101 0111 000 11
      010 1100 011 11 0
       10 1011 100 01 1
       00 0111 111 10 1 000
           101 011 10 0 011
           010 100 00 1 011 0
            10 101 11 0 001 1
R3          00 001 11 1 010 1
R4          0 001 11 1 010 10   (one cyclic shift)
            x    Check Bits
```

Fig.7 :Example of check bits generation with proposed architecture

- At first reset will be High, with which registers R1 has 127h and R5 has 11(01011).
- At first phase of log division prepare, XOR operation happens. The outcome does not contain any driving number of zeros. Thus, priority encoder yield will be zero.
- The data in R5 is grater than the output of priority encoder. The condition block comes up short and the remaining outcome is circularly left shifted by priority encoder yield times (zero).
- In cycle control hinder, as the subtractor yield is positive (Which implies sign bit is high). R4 contains the circularly left moved transitional outcome. As reset is low throught the procedure the data of R1 gets refreshed with R4 and this procedure proceeds.
- In the Final phase of division process, the priority encoder yield is 4 which is more than the data of the R5 (1), with which subtractor yield will be negative. As condition fulfilled, the residuary result is circularly left moved by R5 times.
- As sign bit is high,the intermediate result of register R4 is stored into the register R9.
- As Ld ends up plainly zero, R7[4]=1 and R5[4]=0, the enroll R6[22:11] is refreshed with R8 data and R6[10:0] with R9[10:0]. Which speaks to the finish of the long division and the enlist R6 contains the encoded golay codeword(G23).

2) Proposed Architecture for Extended Golay Code(G24)

The extended golay architecture is shown in Fig. 8, where an extra bit is appended to make it a extended golay code word.

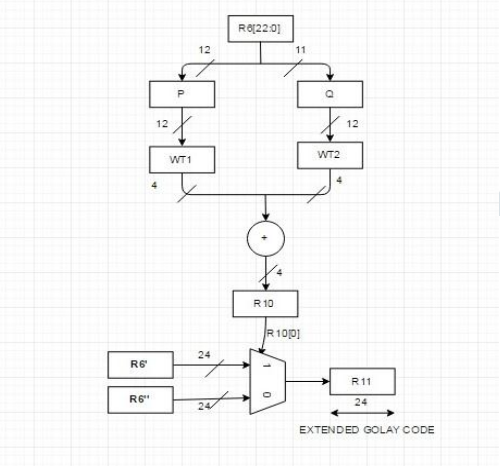


Fig.8. Proposed architecture for extended Golay code

The 23 bit binary golay code which is generated from the implemented architecture is stored in the register R6. Register P is saved with the MSB 12 bit data of reg R6, with an appended Zero reg Q has the LSB of reg R6 (As we are concerned just with number of 1s, 0 is added to make it as 12 bits enlist). The weight of P and Q is calculated by wt1 and wt2 individually. The register R10 will have the weight of binary golay code thai is the added value of output of wt1 and wt2. R6' is annexed with 1, wheras R6'' is affixed 0. The LSB of R10 is the control signal for the 2*1 mux.the 24 bit extended golay code (G24) is stored in Reg R11.

IV. EXPERIMENTAL RESULTS

Both the implemented architecture has been designed and verified using Cadence RTL compiler. The amalgamation comes about acquired from Cadence (R) RTL compiler RC10.1.304−V 10.10−S339 1 utilizing tsmc90.0 innovation show that the aggregate dynamic power utilized by both encoder designs is 188.701 µW,total memory use is 76952K and aggregate cell territory is 5861 µm2. Timing slack is 994607 ps. Recreation comes about with the implemented model are appeared in Fig. 9. R6 and R11 gives the 23 bit double golay codeword and 24 bit extended golay codeword individually.

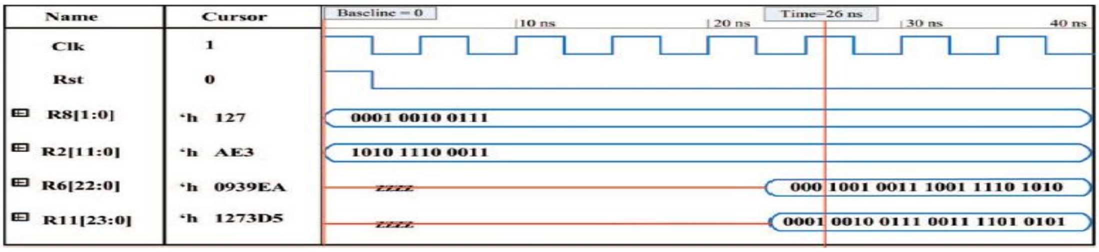


Fig.9. Simulation results with proposed architecture

Table I translates that the proposed engineering sidesteps the three detriments related with the current equipment design. The proposed design obviously outflanks the current equipment engineering.

Characterstics	Existing[20]	Implemented
1. When The Input MSB Is 0 Not Possible Possible	Not Feasible	Feasible
2. When The Priority Encoder Output Is Greater Than Required Number Of Shifts	Not Feasible	Feasible
3. Correct Binary Golay Code Generation	Not Feasible	Feasible

Table II portrays that the proposed encoder engineering cell range and dynamic power are somewhat higher than the current design yet it defeats every one of the disservices related with the current design. Most extreme and least fanout of the proposed encoder engineering are 30 and 0 separately.

Characteristics	Existing[20]	Implemented
1. Dynamic Power(μw)	160.82	188.7
2. Total number of cells	214	333
3. Total Cell Area($\mu m2$)	4001.66	5861.12

Table III speaks to that the proposed encoder design has an idleness of maximal 12 clock cycles like [20]. Also, the timing instrument utilized for the proposed design is framework clock dissimilar to [6].

Reference	Latency	Clocking Mechanism
[6]	23	System Clock + Clock Doubler
[20]	12(Maximum)	System Clock
implemented	12(Maximum)	System Clock

V. CONCLUSION

The defects with the existing architecture of Binary golay code and the extended golay code have been overcome by the implemented architecture. The structure has been composed and verified utilizing cadence tool. The implemented encoder has a lag of 12 clock cycles most extreme and uses the system clock mechanism. These equipment models for encoder can be a conceivable decision for some applications like ultrasonography, ultrasound imaging and in profound space missions.

VI. ACKNOWLEDGMENT

I am extremely grateful to Dr. Nataraj K R , Head of the Department of Electronics and Communication Engineering, for his co-operation and encouragement. I express my deepest gratitude and sincere thanks to Mr SANDESH KUMAR BV, Assistant professor for the valuable guidance throughout my Project.

REFERENCES

[1] Shu Lin, Daniel J. Costello,"Error Control Coding", Second Edition, ISBN 978-81-317-3440-7, 2011.

[2] M.J.E. Golay, "Notes on digital coding," Proc. IRE, volume 37, p.657, Jun.1949., 2000

[3] X.H.Peng and P.G.Farrell,"On construction of the (24, 12, 8) Golay codes," IEEE Trans on Information Theory, vol 52, no 8, pp. 3669-3675. Aug. 2006.

[4] B. Honary and G. Markarian, "New simple encoder and trellis decoder for Golay codes," Electron Lett, Vol.29, no.25, pp. 2170-2171, Dec.1993.

[5] B. K. Classon, "Method, system, apparatus, and phone for error control of Golay encoded data signals," U.S.Patent 6 199 189, Mar.6, 2001.

[6] M.I. Weng and L.N, Lee, "Weighted erasure codec for the (24, 12) extended Golay code," U.S.Patent 4 397 022, Aug.2, 1983.

[7] M. Spachmann, "Automatic generation of parallel CRC circuits,"IEEE.Des.Test Computers, Vol. 18, no.3, pp. 108-114, May 2001.

[8] G. Campobello, G. Patane and M. Russo, "Parallel CRC Realization,"IEEE Trans.Comput, Vol. 52, no. 10, pp. 1312-1319, Oct 2013.

[9] R. Nair, G. Ryan, and F. Farzaneh, "A symbol based algorith for hardware implementation of cyclic redundancy check(CRC)," In Proc. VHDL Int. Users Forum, oct. 1997, pp. 82-87.

[10] Pedro Reviriego, Liyi Xiao, Shanshan Liu and Juan Antonio Maestro "An Efficient Single and Double-Adjacent Error Correcting Parallel Decoder for the (24,12) Extended Golay Code" IEEE Transactions on VLSI systems, vol. 24, no. 4, April 2016.

[11] Shin-Yuan Su, and Pai-Chi Li "Photoacoustic Signal Generation with Golay Coded Excitation", IEEE International Ultrasonics Symposium Proceedings,pp. 2151-2154, 2010.

[12] Satyabrata Sarangi, Pradyut Kumar Sanki, Praful P. Pai, Swapna Banerjee "Comparative Analysis of Golay Code based Excitation and Coherent Averaging for Non-Invasive Glucose Monitoring System", International Symposium on Computer-Based Medical Systems, . IEEE , vol., no.,pp.485-486,2014.

[13] David Romero-Laorden, Carlos Julian Mart, Oscar Martinez-Graullera, Montserrat Parrilla-Romero, n-Arguedas "Application of Golay codes to improve SNR in coarray based synthetic aperture imaging systems" SAM Signal Processing Workshop, IEEE, vol., no., pp.325-328,2012.

[14] Aamir Hussain and Mohammad Bilal Malik "Bi-carrier Golay Code Based Ranging With Improved Range Resolution",ISVC, IEEE, PP.1-4, 2010.

[15] Xiaochun Wang, Jianjun Ji, Sheng Zhou, Yanqun Wang "Ophthalmological Ultrasound Biometer Using Golay-Coded Pulse Excitation", International Conference on BioMedical Engineering and Informatics, IEEE,pp.76-80,Oct. 2014.

[16] Sheng Zhou, Qingsheng Ye, Jun YangXiaochun Wang, Jianjun Ji, Yanqun Wang "Medical High-frequency Ultrasound Imaging using Golay-coded Pulse Excitation", International Conference on BioMedical Engineering and Informatics, IEEE, pp.71-75,Oct.2012.

[17] Mohd Saiful Dzulkefly Zan, Ahmad Ashrif A. Bakar and Tsuneo Horiguchi "Improvement of Signal-to-Noise-Ratio by Combining Walsh and Golay Codes in Modulating the Pump Light of Phase-Shift Pulse BOTDA Fiber", International Conference on Sensing Technology Sensor,IEEE,pp.269-273,2015.

[18] Fanxin Zeng, Zhenyu Zhang, Xiaoping Zeng and Guixin Xuan "16- QAM Golay Complementary Sequence Sets with Arbitrary Lengths", IEEE Communication letters, VOL. 17, NO. 6, JUNE 2013.

[19] Yi Hua Chen, Jue Hsuan Hsiao, Pang-Fu Liu, Jheng-Shyuan He "Golay (20, 8) C Code Simulation and Implementation in DSP Chip", IEEE, pp.4990-4994,Apr.2011.

[20] Satyabrata Sarangi and Swapna Banerjee "Efficient Hardware Implementation of Encoder and Decoder for Golay Code", IEEE Transactions on VLSI, vol. 23, no. 9, September 2015.