International Journal of Information and Communication Technology Research

http://www.esjournals.org

FPGA Based Digital System for Detection of Dicrotic Notch in the Carotid Pulse Signal

Sheikh Md. Rabiul Islam, Shumit Saha, Md. Imran Khan
Department of Electronics & Communication Engineering
Khulna University of Engineering & Technology, Khulna-9203, Bangladesh

ABSTRACT

Heart disease is one of the leading causes of death in human beings. A survey shows every year about 900,000 peoples die due to heart diseases worldwide. Carotid pulse is a pressure signal recorded over the carotid artery as it passes near the surface of the body at the neck. An abnormal carotid pulse called the dicrotic pulse occurs when patients suffer from sepsis, hypovolemic shock, cardiac tamponade, aortic stenosis. Dicrotic notch usually denotes a very low stroke volume, particularly in patients with dilated cardiomyopathy. Lehner and Rangayyan proposed a methodology to detect this dicrotic notch in the carotid pulse signal. This method used the least-squares estimate of the second derivative because a first-derivative operation would give an almost-constant output for the downward slope. In this paper, a digital system is designed to detect the dicrotic notch in the carotid pulse using Verilog Hardware Description Language. Hardware description languages such as Verilog differ from software programming languages because they include ways of describing the propagation of time and signal dependencies (sensitivity). Verilog modules that conform to a synthesizable coding-style, known as RTL (register transfer level), can be physically realized by synthesis software. Synthesis-software algorithmically transforms the Verilog source code into a netlist, a logically-equivalent description consisting only of elementary logic primitives (AND, OR, NOT, flip-flops, etc.) that are available in a specific FPGA or VLSI technology. Further manipulations to the netlist ultimately lead to a circuit fabrication blueprint. Here architecture of a digital system for detection of dicrotic notch in the carotid pulse signal has been proposed by using Verilog HDL based XILINX FPGA board. By this system patients can check his carotid pulse immediately after he feel sick without getting himself admitted in hospital which can save many lives.

Keywords: *Carotid Pulse, Dicrotic Notch, Verilog HDL*

1. INTRODUCTION

The carotid pulse is a pressure signal recorded over the carotid artery as it passes near the surface of the body at the neck. It provides a pulse signal indicating the variations in arterial blood pressure and volume with each heart beat. The dicrotic pulse is an abnormal carotid pulse found in conjunction with certain conditions characterized by low cardiac output. [1]-[4]. (Figure 1)

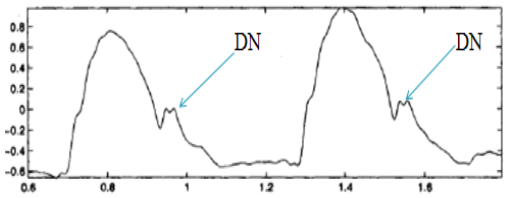

Figure 1: Dicrotic notch in the Carotid Pulse.

This notch tends to occur in patients with sepsis, severe heart failure, hypovolemic shock, cardiac tamponadeaortic, aortic stenosis & aortic insufficiency**.** Sepsis is a severe illness in which bacteria overwhelm the bloodstream. People whose immune systems (the body's defense against infections) are not functioning well because of an illness (such as diabetes or AIDS) or because of medical treatments (such as chemotherapy for cancer or steroids for a number of medical conditions) that weaken the immune system are more prone to develop sepsis. Hypovolemic shock is a state of decreased blood volume; more specifically, decrease in volume of blood plasma. Hypovolemic shock refers to a medical or surgical condition in which rapid fluid loss results in multiple organ failure due to inadequate circulating volume and subsequent inadequate perfusion. Cardiac tamponade is the pressure on the heart muscle which occurs when the pericardial space fills up with fluid faster than the pericardial sac can stretch. If the amount of fluid increases slowly (such as in hypothyroidism) the pericardial sac can expand to contain a liter or more of fluid prior to tamponade occurring. Aortic stenosis is that in which the valve fails to open fully, thereby obstructing blood flow out from the heart. With aortic valve stenosis, the valve cannot open as wide as normal. Because the valve does not open as wide, the heart must work harder to pump blood through the valve. Aortic insufficiency, also called aortic regurgitation, is that in which the aortic valve is incompetent and blood flows passively back to the heart in the wrong direction [5]-[8],[21].

This paper focuses on the design [10], [13]-[16], [19]-[20] of a portable, low power, and low cost device which detects this dicrotic notch in the carotid pulse. This device would be easy to operate, easy to transport and would be used to monitor admitted patients in these areas; patients who unfortunately can't afford the luxury of accommodation in the few well equipped hospitals that exist in their locale.

Xilinx ISE [11],[12] is a software tool produced by Xilinx for synthesis and analysis of HDL designs, which enables the developer to synthesize designs, examine RTL diagrams, simulate a design's reaction to different stimuli, and configure the target device with the programmer. The building block were designed, tested and evaluated using the ISE tool

available from Xilinx & VeriloggerPro6.5 (TestBencher pro)[12].

This paper is structured as follows. Section II describes the Lehner and Rangayyan proposed a method for detection of the dicrotic notch algorithm which is modify. Section III describes about the proposed architecture of the system. In Section IV describes the block of each part of this model and try to analyze the simulation and synthesized result. Finally Section VI for conclusion.

2. BACKGROUND

Lehner and Rangayyan [9], [17]-[18] proposed a method for detection of the dicrotic notch that used the least-squares estimate of the second derivative *p(n)* of the carotid pulse signal y(n) defined as

$$p(n) = 2y(n-2) - y(n-1) - 2y(n) - y(n+1) + 2y(n+2)$$

Observe that this expression is noncausal; it may be made causal by adding a delay of two samples.
The second derivative was used due to the fact that the dicrotic notch appears as a short wave riding on the downward slope of the carotid pulse signal. A first-derivative operation would give an almost-constant output for the downward slope. The second-derivative operation removes the effect of the downward slope and enhances the notch itself. The result was squared and smoothed to obtain the feature signal s(n).

$$s(n) = \sum_{k=1}^{M} P^2 (n - k + 1) w(k)$$

Where, $w(k) = (M - k + 1)$ is a linear weighting function, and window width, $M = 16$ for sampling rate, $fs = 256\ Hz$. The method yields two peaks for each period of the carotid pulse signal. The first peak in the result represents the onset of the carotid upstroke. The second peak that appears in the result within a cardiac cycle is due to the dicrotic notch.

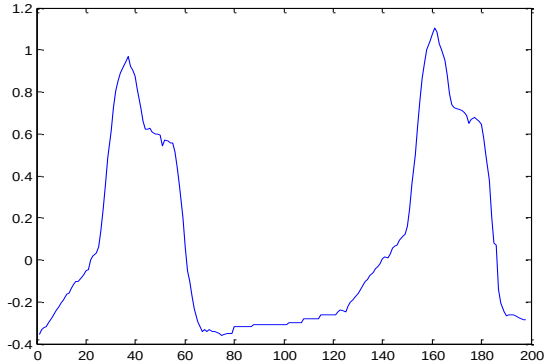

Figure 2: Carotid Pulse Signals.

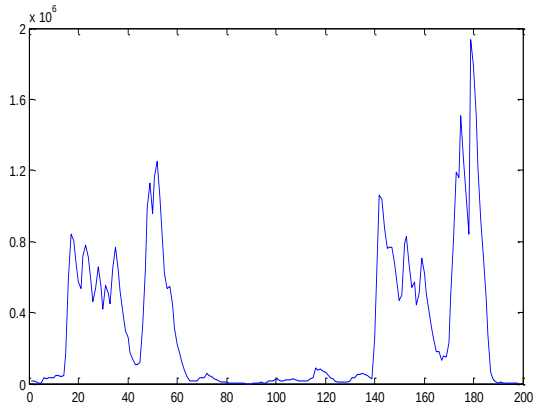

Figure 3: Feature signal, s(n) signal.

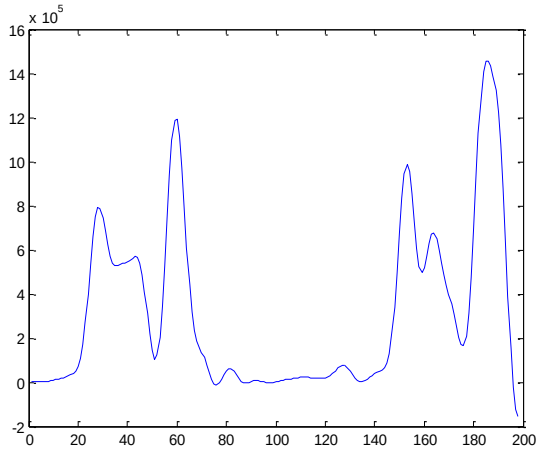

Figure 4: S(n) signal after filtering by 16 order Butterworth filter.

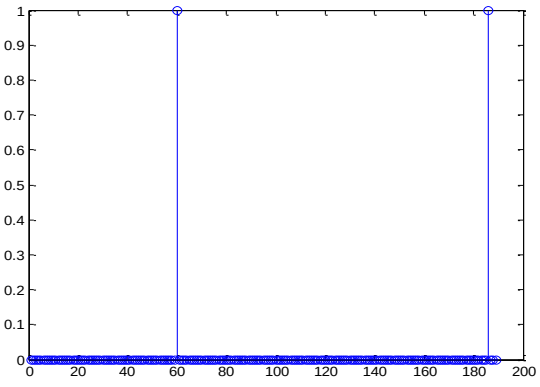

Figure 5:Detection of position dicrotic notch in carotid pulse using peak searching algorithm.

Figure 2 to 5 the modified Lehner & Rangayyan method is coded in MATLAB. Figure.2 shows a carotid pulse signal and in Figure.3 shows the s(n) output signal. This s(n) signal is then filtered by a 16 order Butterworth filter. This Butterworth filter is used for smoothing the signal. Here, 16 order means the window size is 16. The widow width needs to be chosen in accordance with the characteristics of the signal on hand as well as the low pass filter and sampling rate used. After that a peak searching algorithm is run to find the dicrotic notch peak. This peak searching algorithm is developed to find the peak of

the dicrotic notch. Here, at first a threshold value is needed to be set. This value can be set by seeing the data stream. In figure 5 the detection of the notch is seen. The peak searching algorithm for our system is given below:

i. Set threshold value.
ii. Scan first 70 samples whose are greater than the threshold.
If s(n)>=threshold then s(n)=1
Else s(n)=0.
iii. If two samples s(n1) and s(n2) are equal to unity then
If n1>n2 then s(n1)=1 and s(n2)=0
Else s (n2) =1 and s(n1)=0.
iv. Repeat step 2 and 3 for next 70 samples.
v. Repeat step 4 until whole s(n) is scanned completely.

3. PROPOSED ARCHITECTURE

In this section, block diagram of our proposed architecture of digital system has been represented based on data bus, control bus and including internal connection.

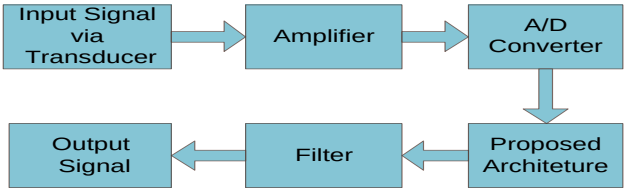

Figure.6 Block diagram of the digital system.

Figure.6 shows the block diagram of the digital system. System consists of transducer for sensing and converting physiological signal to electrical signal. Here, piezoelectric transducer can be used for acquisition of pulses. Amplifier is needed because carotid pulse is low voltage signal. For digital system design A/D converter is must. Processor provides feature signal S(n) from carotid pulse. Dicrotic notch is found using peak searching algorithm. In order to avoid floating point calculation complexity processor is designed such a way that the value of y(n) is 1000 times than original data.

In proposed architecture as shown in Figure.7 the data is inputted from input module and outputted on output module. The data's from RAM, Register A, Register B And ALU are bidirectional. Figure.8 every block of the system is connected with the control unit of the system with internal connection. All control signals is one way. The address bus of the proposed architecture is shown in Figure.9.

4. DESCRIPTION OF DIFFERENT BLOCKS WITH SIMULATION RESULTS & SYNTHESIS

The proposed design of the digital system has different blocks which described below:

A. ALU: It is an arithmetic logic unit. Figure.10 shows the block diagram of ALU. Bit<4:0> means 5 bits bit1, bit2, bit3, bit4, bit5 were come from control ROM. Those bits were represent the values of the number by which the ALU multiply with the values come from the registers. Subblocks are *Add, sub, mul, shift_left, shift_r* the control signals for specific applications. *Data_in1<12:0>, data_in2<12:0>* are input data come from Register A & Register B respectively. *Data_out<12:0>* is connected with Register B.

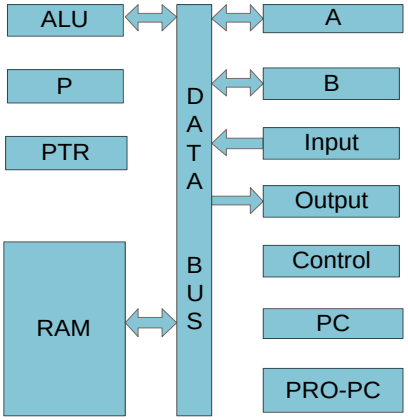

Figure 7: Block diagram of the system including data bus.

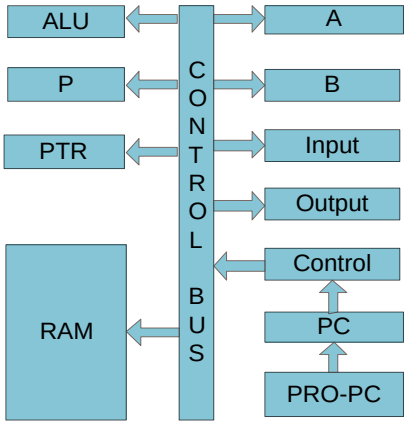

Figure.8 Block diagram of the system including control bus and internal structure.

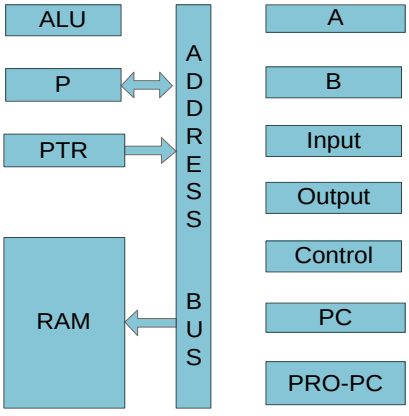

Figure 9: Block diagram of the system including address bus.

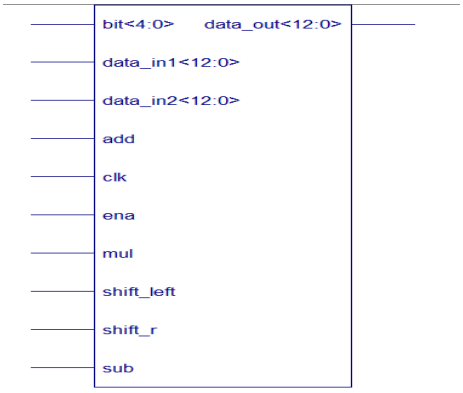

Figure 10: Block Diagram of ALU.

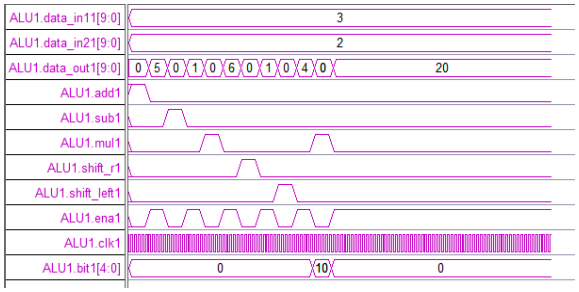

Figure 11: Timing diagram of ALU.

Figure.11 shows the simulation result of ALU. In this process have been used two values 2 and 3 as input sources. The value of 2 and 3 is added when the add control signal is enabled and got the output value 5. Again functions for subtraction of 3 and 2 are subtracted and the output value is 1, and then multiplied and the output is 6. The date from data_in2 which is 2 is right shifted and the value is 1, and the value 2 is left shifted and the value is 4. Here, assign value 10 (hexadecimal) is the control bit and it is multiplied by 2 and the value is 20.

B. Control Rom: It is ROM of digital system. It has input from the Program Counter and has 42 bit output which is called *control bus*. The control bus structure is shown in Table.1. The block diagram of the control ROM is in Figure.12.

Table 1: Control Bus from Control ROM to different Module

Module Name	Control Bus Name (Bus number)
input	Ena(1),out(2)
RAM	Read(3), write(4), clear(5)
Ptr	Read(6), increase(7), clear(8), bit1(9), bit2(10) , bit3(11)
A	Write(12), ena(13) ,ena_alu(14)
B	Write(15), ena(16) ,ena_alu(18), clr(17)
ALU	Add(19),sub(20),mul(21),shift_left(22), shift_right(23), ena(24), bit1(25), bit2(26),bit3(27),bit4(28),bit5(29)
P	Ena(30), ptr_bit1(31), ptr_bit2(32) , ptr_bit3(33), p_bit1(34), p_bit2(35) , p_bit3(36), p_bit4(37),p_bit5(38), clr(42), out(39)
Out	Write(40),out(41)

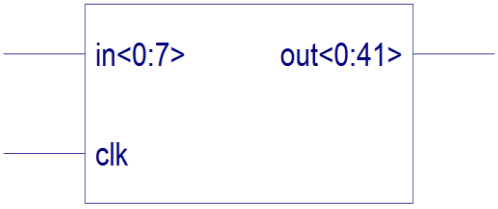

Figure.12 Block Diagram of Control Unit

C. Input: This is a register which receives and stores the input samples of carotid pulse signal that is Y (n). Figure 13 shows the block diagram of Input module. The time diagram for input block as example shown in Figure 14 where the *ena* control signal is on then the data out goes to 0. When the *out* control signal is enabled, data from input goes to output.

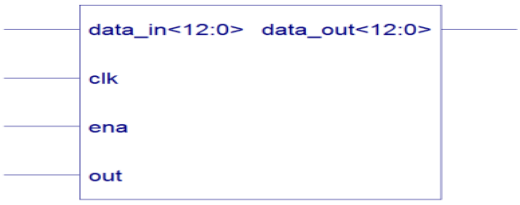

Figure 13: Block Diagram of Input Block

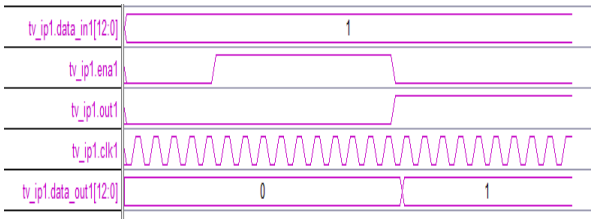

Figure 14 Timing Diagram of Input Block

D. Output: It is also a register and sends the data through the output bus as shown in Figure.15. The timing diagram of output is shown in Figure.16.

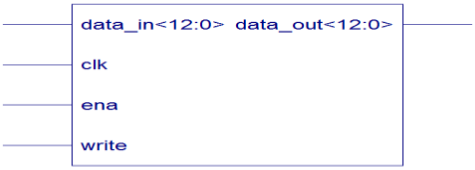

Figure 15: Block Diagram of Output Block

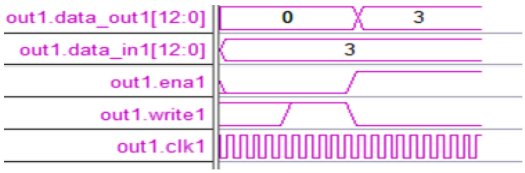

Figure 16: Timing Diagram of Output Block

E. *RAM:* It is a memory bank which stores the input value $Y(n)$,the values of $P^2(n)$ and output value $S(n)$ according to the DSP equation. Here,$P^2(n)$ is stored instead of $P^2(n)$ because in solving $S(n)$, $P(n)$ is not needed. All the addresses of RAM are filled with values like the Table.2 Initially.

Table 2: Initial Structure of RAM

Add.	Z	Add.	Z	Add.	Z
00001	Values of Y(n)	01001	Values of P^2(n-13)	10001	Values of P^2(n-5)
00010	Values of Y(n-4)	01010	Values of P^2(n-12)	10010	Values of P^2(n-4)
00011	Values of Y(n-3)	01011	Values of P^2(n-11)	10011	Values of P^2(n-3)
00100	Values of Y(n-2)	01100	Values of P^2(n-10)	10100	Values of P^2(n-2)
00101	Values of Y(n-1)	01101	Values of P^2(n-9)	10101	Values of P^2(n-1)
00110	Values of P^2(n)	01110	Values of P^2(n-8)	10110	Values of S(n)
00111	Values of P^2 (n-15)	01111	Values of P^2(n-7)		
01000	Values of P^2 (n-14)	10000	Values of P^2(n-6)		

Executions process of this block Y(n) comes from the input source and then Y(n-4) will become Y(n-5) but in equation range from 0 to 4, so Y(n-5) will not needed for the equation and we replace the new Y(n) in the address 00100 as shown in Table.3.

Table.3 New values are Executing in the RAM

Add. Initial	Z	Add. 1 entry	Z	Add. 2 entry	Z
00001	Values of Y(n)	00001	Values of Y(n-1)	1	Values of Y(n-2)
00010	Values of Y(n-4)	00010	Values of Y(n)	00010	Values of Y(n-1)
00011	Values of Y(n-3)	00011	Values of Y(n-4)	00011	Values of Y(n)
00100	Values of Y(n-2)	00100	Values of Y(n-3)	00100	Values of Y(n-4)
00101	Values of Y(n-1)	00101	Values of Y(n-2)	00101	Values of Y(n-3)

The above process is repeated for P^2(n) values. Figure.17 shows the block diagram of RAM. Here, *read & write* are the two control signal for reading the values from RAM and writing a new value in the RAM respectively.

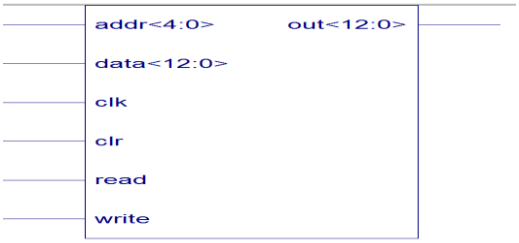

Figure 17: Block diagram of RAM

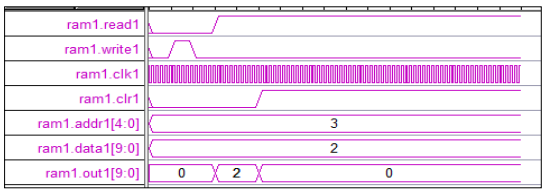

Figure 18: Timing Diagram of RAM

In Figure.18 we have investigated that the data value 2 is saved in memory location no 3 in RAM and when it will need for further process the data can be read from this location. This process is repeated for next execution.

F. Register A and B: These are the two register which is needed to process the data. Figure.19 shows the block diagram of register B.

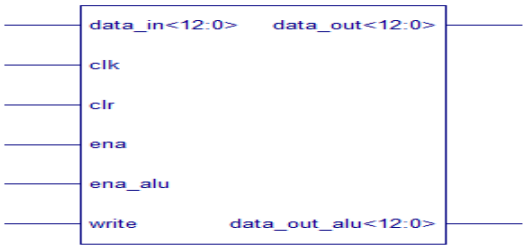

Figure 19: Block Diagram of Register B

G. *Pointer* (Ptr): It is a module which stores the address values of $Y(n)$, $P^2(n)$ and $S(n)$. Figure.20 shows the block diagram of Ptr. Table 4 shown the Ptr has 3 bit inputs which indicates whether it is $Y(n)$ or other. Ptr generates a 5 bit address for corresponding input bits.

Table.4: Generated Address via Pointer (Ptr)

Bit	Z	Address
011	Y(n)	00000
001	P^2(n)	00101
010	S(n)	10101

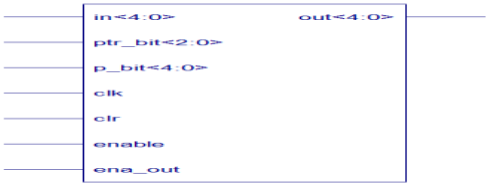

Figure 20: Block Diagram of Ptr Block

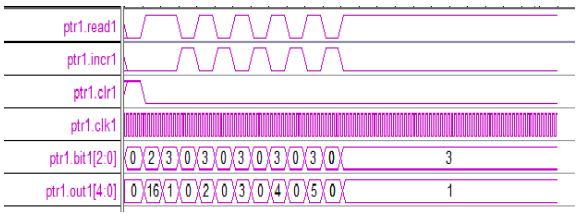

Figure 21: Timing Diagram of *Ptr* Block

In Figure.21 shown when the increment pin is active that increment the present state address of (n) . Initially the output is 1. And when, the increment pin is on then the value goes to 2 and so on. As we know, the value doesn't exceed 6 so that after 5 increments it goes back to 1.

H. Pointer (P): It is a module which generate the address of Y(n-1),Y(n-2) and P^2(n-1) and so on, where the address of Y(n) or P^2(n) is comes from (pointer) *Ptr*. Figure.22 shows the block diagram of P.

Figure 22: Block Diagram of P Block

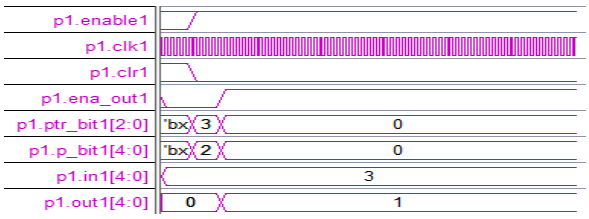

Figure 23: Timing Diagram of P Block

From Figure.23 shown for *ptr_bit* 3, *p_bit* is 2 and *input* bit is given 3 so that *p_out* bit is 1. Which means that $Y(n)$ is delayed by one. So now *P* points to the position of Y (n-1).

I. Pro-Program counter (Pro_PC): It is a one kind of clock pulse generator and counter. It generates clock pulse for driving the *Program counter (PC)* from the main clock pulse. In this design it generates a pulse with one positive edge against four main clock pulses. Figure.24 shows the block diagram of *Pro_PC*.

Figure 24: Block diagram of Pro_PC

J. Program counter (PC): It is a counter which counts the positive edge of its input pulses which is come from Pro_Pc as shown Table.5 after execution of this block. This device drives the Control ROM. Figure.25 shows the block diagram of PC.

Table 5: Execution of the PC Block

Input of Pro_PC (main clk)	Output of Pro_PC /input of PC(binary)	Output of PC (binary)
0	0	0
1	01	0
0	01	0
1	10	0
0	10	0
1	11	01
0	11	01
1	0	01
0	0	01
1	01	01
0	01	01
1	10	01
0	10	01
1	11	10
0	11	10
So on…	So on…	So on…

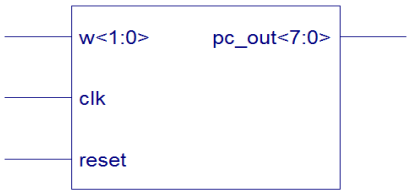

Figure 25: Block Diagram of PC Block

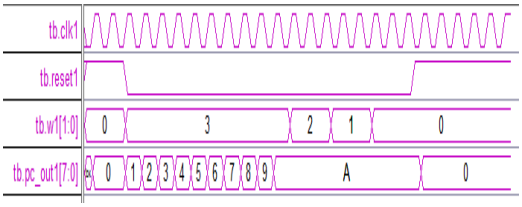

Figure 26: Timing Diagram of PC Block

In Figure.26 shown that for various values of *w* and it gives several outputs. When the *w* is 3 it counts clock simultaneously. When the value of *w* is 1 or 2 the output is fixed so that, we see that output of PC is changed only when *w* is goes to 3 as shown in details in Table.5.The final top diagram of the system is shown in Figure.27.

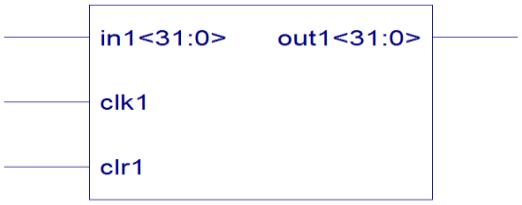

Figure 27: Top Diagram of the System

Table.6 shows the real time sample data and its corresponding output. The first input value is -0.355. This is multiplied by 1000 and the output is -0.355*1000= -355. It is seen from the Table.6 that the s(n) is 8065600(7b1240 in Hexadecimal) for input value -355. For -0.33, the Verilog input data is -0.33*1000 = -330. The S(n) value is 9049900(8A172C in Hexadecimal) for input value -330. In Figure.28, the output is shown for first five input values.

Table 6: Real Time Data Analysis

Time (µs)	Real Data	Input Data	Feature Signal s(n)	Hexadecimal output
0	-0.355	-355	8065600	7b1240
4	-0.33	-330	9049900	8A172C
8	-0.325	-325	10886375	A61CE7
12	-0.32	-320	17977150	1124F3E
16	-0.3	-300	16739525	FF6CC5
20	-0.275	-275	16500200	FBC5E8
24	-0.255	-255	17329975	1086F37
28	-0.240	-240	22144925	151E79D
32	-0.225	-225	21129346	1426882
36	-0.205	-205	19310550	126A7D6
So on	So on	So on	So on	So on

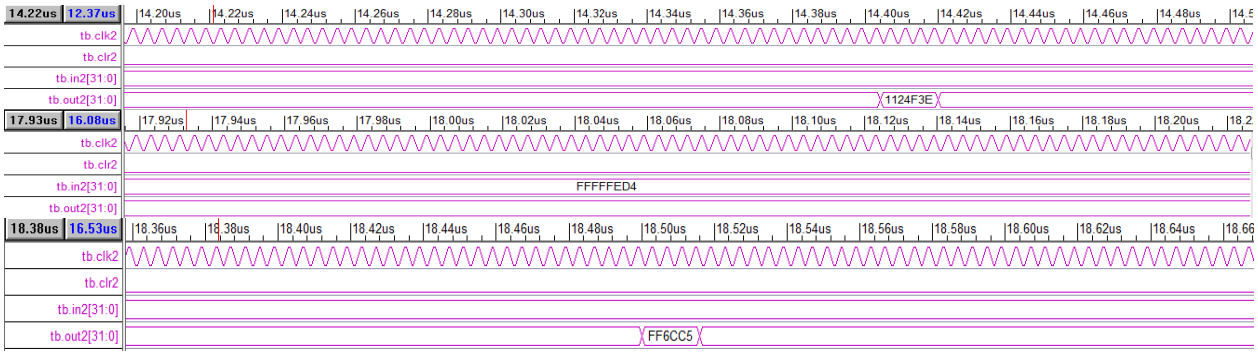

Figure 28: Timing Diagrams of Real Time Sample Data

In Figure.28, we see that Hexadecimal value 7b1240 is obtained at 2.10 µs, 8A172C is obtained at 6.22 µs, A61CE7 is obtained at 10.32 µs, 1124F3E is obtained at 14.40 µs, and FF6CC5 is obtained at 18.50 µs.

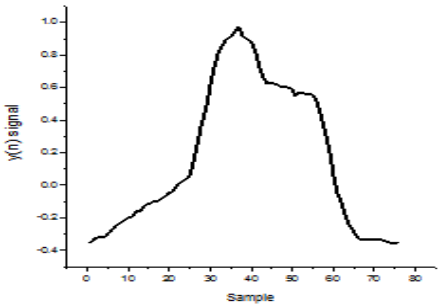

Figure 29: Real Time Input Data

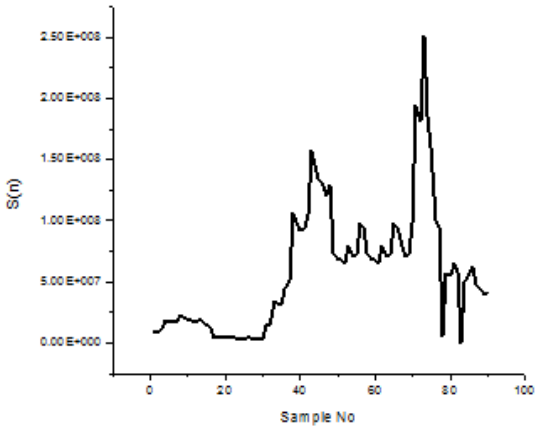

Figure 30: Real Time Output Data

Now, from the table.6 all the input values and output s(n) signal are plotted. And Figure.29 & Figure.30 are obtained. And in Figure.30, it is seen that the notch is being detected according to input signal.

5. CONCLUSION

The architecture for detecting the dicrotic notch in the carotid pulse signal has been presented in this paper. Using Verilog HDL in designing the system, not only the restraints in analogy circuit can be relaxed but also better speed, lower cost, and higher flexibility can be obtained than any other design technique. This designed system can be implemented by the manufacturer, according to the circuit description of Verilog coding. It could be a very much suitable device for the patients as this system will implement as a simple chip. Again the implementation of the system on FPGAs is better because FPGAs can give enhanced speed. By this system many heart disease like sepsis, severe heart failure, hypovolemic shock, cardiac tamponadeaortic, aortic stenosis & aortic insufficiency can be detected at a very low cost. In future we will develop it in hardware based design.

REFERENCES

[1]. Tavel ME., "Clinical Phonocardiography and External Pulse Recording", *Year Book Medical, Chicago, IL, 3rd edition*, 1978

[2]. http://en.wikipedia.org/wiki/Dicrotic_pulse

[3]. Ewy GA., Rios JC. and Marcus FI., "The Dicrotic Arterial Pulse", *CIRCULATION Journal of The American Heart Association*

[4]. Smith D.,Craige E., **"**Mechanism of the dicrotic pulse"**,** *From the Department of Medicine of the University of North Carolina*, Chapel Hill, North Carolina, USA

[5]. http://en.wikipedia.org/wiki/Cardiac_tamponade

[6]. http://www.ncbi.nlm.nih.gov/pubmedhealth/PMH0001687/

[7]. http://ourmedicine.blogspot.com/2009/10/180-dicrotic-pulse.html

[8]. http://en.wikipedia.org/wiki/Hypovolemia

[9]. Rangayyan RM., "Biomedical Signal Analysis - A Case-study Approach", *IEEE and Wiley*, New York, NY, 2002

[10]. Palnitkar S.; "Verilog HDL.", *SUNSOFT PRESS*, 1996

[11]. XILINX, Internet site address: http://www.xilinx.com

[12]. TestBencher pro Internet site address: http://www.syncad.com

[13]. Perry D., "VHDL. 3rd Edition", *Tata Mc. Graw Hill Publications*, 2001

[14]. Bhaskar J.,"VHDL primer.,3rd Edition", *Pearson Education Asia Publications*, 2000

[15]. Chen WK.,. "Logic Design.",*CRC Press*, 2000

[16]. Wakerly JF., "Digital Design & Practice., 3rd edition", *Pearson Education*, Asia

[17]. Udupa KJ., Murthy ISN. and Rangayyan RM.; "Epoch analysis of ECG and PCG signals", *ibid.(2)*, pp. 8-57 to 8-60, 1977

[18]. Burrus CS.; "Digital Filters Structures described by Distributed Arithmetic", *IEEE Transactions on Circuits and Systems*, vol. CAS-24, page: 12, December 1977

[19]. Mano M.; "Digital Logic Design, 3rd Edition", *Tata McGRAW HILL*

[20]. Malvino AP, Brown JA, "Digital Computer Electronics, 2nd Edition", *Tata McGRAW HILL*, New Delhi, 2007

[21]. Smith D. and Craige E., "Mechanism of the dicrotic pulse", *British Heart Journal*, 1986 December; 56(6): 531–53.

AUTHOR BIOGRAPHIES

Sheikh Md. Rabiul Islam received the B.Sc.in Engg. (ECE) from Khulna University, Khulna, Bangladesh in December 2003, and M.Sc. in Telecommunication Engineering from the University of Trento, Italy, in October 2009. He joined as a Lecturer in the department of Electronics and Communication Engineering of Khulna University of Engineering & Technology, Khulna, in 2004, where he is currently an Assistant Professor in the same department in the effect of 2008. He has published fifteen Journal and six international conferences. His research interests include Numerical analysis, VLSI, wireless communications, signal & image processing, and biomedical engineering.

Shumit Saha received a B.Sc. Engg. (ECE) degree from Department of Electronics and Communication Engineering at Khulna University of Engineering & Technology, Khulna_9203, Bangladesh. He joined as a Lecturer in the department of Electronics and Communication Engineering of Khulna University of Engineering & Technology, Khulna, in 2012. He has published one Journal and one international conference. His research interests include VLSI Design, biomedical signal processing and biomedical engineering.

Md. Imran Khan received a B.Sc. Engg. (ECE)of department of Electronics and Communication Engineering at Khulna University of Engineering & Technology, Khulna_9203, Bangladesh.