Efficient combination of Electronics Switching System and VLSI technology

K.P.Rane, S.V.Patil and A.M.Patil

***Abstract*-- Speed and size are the important factors while designing the electronic system. It's Speed of operation and flexibility to modify, measures the performance of the system operation. Microprocessor/microcontroller (MPMC) system can handle sequential operations with high flexibility and use of Complex Programmable Devices (CPLD) can handle concurrent operations with high speed in small size area. So combined features of both can make high performance system. We applied this new approach to design high performance Hybrid Telephone Switching System (HTSS) using combination of stored program control (SPC) electronic switching system and VLSI technology. Call handling operations are handled by concurrent operations using CPLD and complicated and sequential operations like services handling are handled by MPMC. VHDL codes are designed and tested for different call handling operations. The test benches are designed to act as MPMC for the testing of services like Do Not Disturb (DND), Call Forwarding (CF), Outgoing Bar (OGB) and STD Bar (STDB) facilities. Call handling speed of available system designed with MPMC and proposed system designed with CPLD is compared logically.**

***Keywords*-- CPLD, FSM, VLSI, VHDL and State Diagram**

I. INTRODUCTION

CONSIDER a problem of design of digital circuit. K-map is drawn from the truth table in this design. It is then minimized and finally equation obtained is implemented with logic gates. This will become dedicated circuit for the given problem but development time is less. If there is slight difference in problem definition then we have to do such thing repeatedly. It becomes very time consuming process. In above, problem has to change discrete component every time. Then it demands for flexible system i.e. using microprocessor and micro-controllers. In micro-processor/micro-controller (MPMC) system, if there are slight changes in problem definition, can change the software according to it. Hence flexibility can be obtained [1]. No need to change every time the hardware. Here development time is more but the system modification is done easily. This system may not be dedicated.

K.P.Rane is with Electronics and Telecommunication Department, J.T.Mahajan College of Engineering, Faizpur, District Jalgaon, Maharashtra (India). (e-mail: kantiprane@rediffmail.com).

S.V.Patil is with Electronics and Telecommunication Department, J.T.Mahajan College of Engineering, Faizpur, District Jalgaon, Maharashtra (India). (e-mail: svpatil67@yahoo.co.in).

A.M.Patil is with Electronics and Telecommunication Department, J.T.Mahajan College of Engineering, Faizpur, District Jalgaon, Maharashtra (India). (e-mail: a_m_patil21@gmail.com).

These are some advantages and disadvantages of the above two systems. Taking into account advantage of discrete system that is less development time [2] and advantage of microcontroller/microprocessor that is easily modify and more flexible, we combine these two advantages and propose VLSI technique using VHDL (Very High Speed IC, Hardware description Language).

Here attempt is made to reduce the size of hardware required and to increase the speed of system by using the VLSI technique. Using this VLSI design we can reduce all digital ICs in single VLSI IC**.** We considered here the design of Telephone Switching System (TSS) using VLSI technology. We introduced High performance HTSS using combined stored program control (SPC) and VLSI technology for optimization of size and speed.

Because of use of sequential operations of MPMC for available TSS [1], over all speed of system becomes less and due to that call processing time, traffic handling capacity, switching capacity decreases. Improvement of all those parameters is very great problem while designing large system. So it is required to think about those parameters. One solution is to design high speed switching matrix and another solution is to design a controlling system that works very fast. Here we tried to solve the problem with second solution by using CPLD. Here we synthesize, simulate and implement the VHDL codes for 8x4 TSS (8 extension (telephone) lines and 4 junction (trunk) lines). Here 11 to 18 are proposed numbers for eight extension lines.

In traditional TSS systems, it is not possible to increase the extension or junction lines because of limitation of processor or controlling system. If it is tried to design the system with higher processors then it become cost effective and complicated. Solution for this problem is to implement the controlling system for eight lines in a CPLD and make provision to co-control or cascade to the other CPLD of another eight lines so that we can have as many as possible lines. It is also possible because CPLDs available are not so costly.

Size of processor system or controlling circuit also increases if we are using higher processors to solve problem discussed earlier. But if we are using CPLDs then it is possible to implement maximum digital circuit into it and size of CPLD is respectively small with more IO pins. So it is better to choose CPLD for our system. There are different properties or optimization terms of CPLD that can be controlled by Xilinx Synthesis Tools for optimization in speed and size [3]**.**

II. HYBRID SWITCHING SYSTEM

Switching system development focuses on readily available, cheaper and reliable technology [4]. Service-rich telecommunication environment is proposed by Ronnie Lee Bennett and ask for switching system with high speed broadband service requirements. Today's switching systems are moving towards the technique which is having capability of multimedia telecommunication services [4]. So we introduced hybrid switching system to optimize speed, cast and service oriented operations. Requirements of service oriented switching system are 1) they should be expandable and 2) Enhanced Services should be added quickly [5]. Available SPC switching systems are real time systems. Each telephone set is considered as input/output port and telephone processes are executed in real time with non stop operations [6].

We proposed here to separate total digital TTS (processor card) circuit around processor into two parts. One is service-handling circuit (SHC) and another is call-handling circuit (CHC) as shown in Fig.2.1. It is great important to increase the call handling speed of system to improve the system performance. Use of SHC and CHC helps to improve speed because CHC is designed to work in concurrent manner to handle individual calls of subscribers and SHC helps CHC while any service/facility is to be provided. SHC is having all features of SPC with removed call handling operations. Generally each line card includes driver circuit of eight lines. Presently many processing functions are incorporated on the line cards and those perform many switching functions by themselves [7]. CHC performs all those functions. Size of circuit is most important factor while designing any circuit so we reduced CHC in to single IC [3], therefore reduced processor card can be placed in any corner of line cards. The attempt is made to design CHC to work with eight lines so that we can place it on each line card. Junction card includes driver circuit for trunk lines and Switching Matrix establishes the connection between calling line and called line or between calling line and junction line depending on the signals on sw and swx.

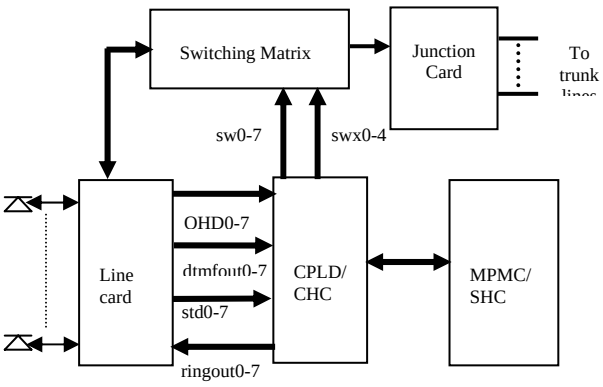

Fig.2.1. Block schematic of TSS

Service Status Buffer (SSB) is designed in CPLD so that it can be accessed as fast as possible for call handling and as shown in Fig.2.2. Subscriber is when asking for some facility, it is handover to SHC and it modifies the contents of SSB. While any number is dialed, it checks the self status and status of dialed number both in SSB. It finds any flag set of those statuses registers and then interrupts to MPMC, which handles the call to provide the proper service to subscriber. When flag is not found set after the number is dialed, the call is handled by CHC. Quickly addition of new services is also possible by assigning new buffer for it with incorporating Software Patch [6] in SHC.

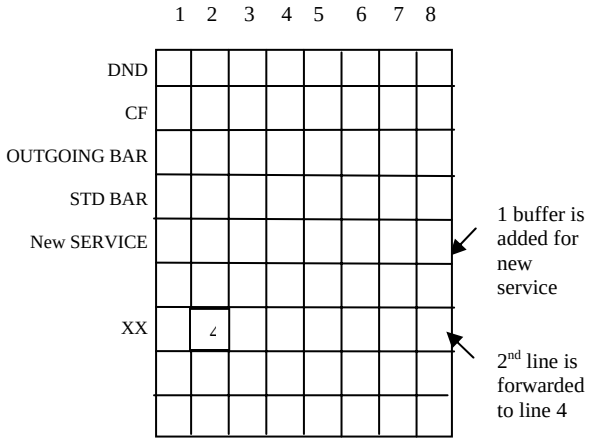

Fig.2.2. Service Status Buffer

MPMC when got service interrupt, set or execute the service and provides the corresponding tones to called subscriber by making use of CHC to handle the call. Therefore complicated and sequential operations are handled by MPMC in parallel with the operation of CHC and don't interrupts to the regular operation of CHC.

We tested call-handling operation of the system by designing CHC using VHDL. We designed test benches for testing the operation of MPMC for handling the different services (i.e. setting and executing the different services). The results are tested and shown in Appendix.1.

III. FINITE STATE MACHINE (FSM)

FSM design is the technique for the design of clocked sequential circuits which performs some repetitive actions. State diagrams and state tables are the best representations of FSM. State diagram is the graphical representation of FSM. It shows the actual flow of data and flow path from inputs to outputs. According to the problem definition, state diagrams can be easily designed.

Most telephone systems are modeled as FSM [5]. So we considered the same and designed finite state machine (FSM) for implementing the call handling operation of a line in CHC using VHDL. We proposed FSM because of easily implementation into VHDL with sophisticated tools available in Xilinx software for it. Katsumi Maruyama had proposed concurrent object oriented modeling for simultaneous call handling using MPMC [5]. But we proposed Structural type of Modeling [2] for implementing the replication of FSM for all lines in CHC which provides the concurrent operations for simultaneous calls from all lines. This FSM consists of 10

main states**.** Each having particular operation and there are so many conditional state switching in it. We divide the FSM in to two parts called process 1 and process 2.

Functions of Process 1 (p1) are

1) Detecting the condition of line and accordingly sending tones.
2) Detecting DTMF tones from subscriber and if it is of any other line (remote line) then sending the ring signal to ring generator of remote line.
3) Sending signal to switching matrix for making proper connection.
4) Detecting DTMF tones from subscriber if it is for asking any facility and according modifying the corresponding buffers with the use of SHC.

Functions of Process 2 (p2) are

1) Checking on/off hook condition of called line.
2) Generating ringout signal for ring generation on called line.

Table 3.1. Provides information for identifier codes used in FSM.

Steps for process p1 are as follows and corresponding state diagram is shown in fig.8.1.

Steps for P1…

1) When the telephone line is on hook, the state st becomes 0000 state.

TABLE 3.1
IDENTIFIER CODES USED IN FSM

Identifier Codes	**Description**
11 – 18 dial	Extension numbers
0 dial	Junction lines are four and selected by 0 dial only and changes by auto select technique.
811x	x is any between 1 to 8. 81 is code for **Call forwarding** facility and 1x is the number of line where the call has to be forwarded from the line from which 811x is dialed.
82	Code for **do not disturb** facility.
83pppp1x	83 is code for **Outgoing bar** facility. pppp is password, which is fix here (8383) and 1x is the number of line whose facility has to be activated or deactivated. This facility can be given from any extension to any one.
84pppp1x	84 is code for **std bar** facility. pppp is password, which is fix here (8484) and 1x is the number of line whose facility has to be activated or deactivated. This facility can be given from any extension to any one.
000	No tone
001	Dial tone
010	Call processing tone
011	Engage tone
100	Ring back tone
101	Activation tone

2) When line goes to off hook, state st goes to 0001 state.
3) When 1st digit is dialed on telephone, the dialed number is stored in dtmf1 and d1 buffer. Sub-state a1 becomes 001. Before it, it was 000. If a1 is 001 and q is 1 then it resets the buffer d1, dtmf1 i.e. q=1 => reset the buffer. Else if a1 is 001 and q is 0 then dialed number is stored in d1and dtmf1 buffers i.e. q=0 => ready to use buffer. Else if a1 is 001 and m is 0 then d1 and d2 resets and m is made 1 i.e. m=1 => reset the buffers d1, d2 in previous state.
4) If 1st dial is 1, state st becomes 0010**.** Else if 1st dial is 8 then state st becomes 1000 Else if 1st dial is 0 then state a1 becomes 010, after that it will check for junction line status through a1 state 011 and if line is not engage then a1 state becomes 100 and then st becomes 0100. Else if 1st dial is 0 and out going bar facility is activated then state st becomes 0000. Else if all lines are engage, engage tone is send to subscriber and goes to st state 0000.
5) If 2nd digit is dialed then it is stored in dtmf2 and d2 buffer.
6) If 2nd dial is valid number between 1 and 8 and previous state a1 was 001 then st becomes 0010. In this state selector of extension line is selected to remote or dialed line and blank tone is send (voice connection).
7) If 2nd dial is number between 1 and 4 and previous state st was 1000 then corresponding signal a to d becomes inverted to activates the corresponding process with acceptance tone. It resets d1 and d2 buffers and interrupts to MPMC.
8) If 2nd dial is invalid number then st becomes 0000.
9) If 2nd dial is 0 and previous state st was 0100 and std bar facility is activated then st becomes 0000.
10) If st state is 0010 then 2nd dial is stored in dtmf2 and d2 and acceptance tone is generated and it check conditions (1) If no self dial and no do not disturb facility and call forward facility is activated then called party status (ohdi) becomes 2nd dialed number. (2) If do not disturb facility is activated then ohdi becomes 0000. (3) If call forward facility is activated then called number status (ohdi) is taken from buffer xx(which was stored while activating CF facility).
11) After satisfaction of any previous condition, st becomes 0010 and checks dialed number's on hook status.
12) If remote line (dialed line) is on hook then st becomes 1011, ring back tone goes to calling line and called party status ohdi is sent to ring generator.
13) Before completion of all 16 rings, if dialed number line becomes off hook then st becomes 1100 and tone generated is blank and selector switch is switched to called line by signal sw.
14) After completion of all 16 rings, if a dialed number line remains on hook then st becomes 1100 and tone is engage tone to calling line.
15) As ohdi gives valid signal to ring generator, process p2 generates ring and return to previous state.

Steps for process p2 are as follows and corresponding state diagram is shown in fig. 8.3.

Steps for process P2…

1) P2 gets the event at every clock pulse of square wave rt and scans every lines so that any one line wants to send ring to remote line or not.
2) If one want to send the ring then remote statue rts is stored to the dummy buffer rts1 or if called line (remote line) is activated by call forward facility then rts1is loaded from xx buffer.(xx buffer stores the call forwarded status of all lines).
3) Then it checks on hook condition of remote line.
4) If previous state was 1011 (1011 state=>ring status comes to ring generator and called line is on hook condition) then ring tone is send to called line with ring back to calling line

and events to p2 process for next step. Else if previous state was 1011 (1011 state=>ring status comes to ring generator and remote is on hook condition) and all 16 rings are not over with called line goes to off hook then ring count resets and events to p2 process. Else if previous state is1100 (1100 state=>Before ring remote is off hook) and rings are not over or started then reset count, rings stops and events to p2 process.

IV. CALL PROCESSING TIME

Transition of states is given in Fig.4.1.

T is the clock period. From Fig.4.1 we find, Total delay of MPMC for one transition =22 T and Total delay of CPLD for one transition =4 T.

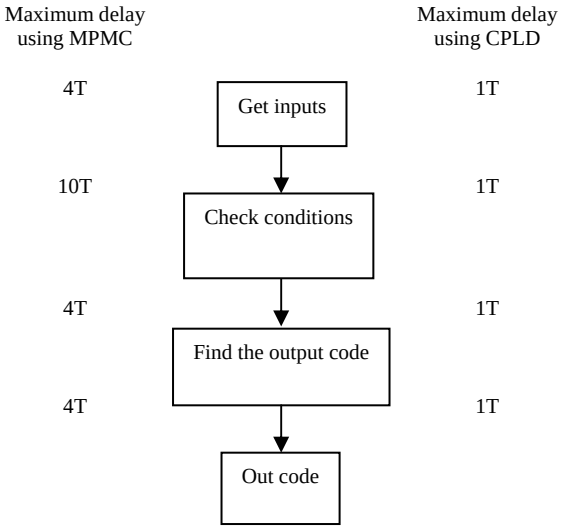

Fig.4.1. State transition steps

Minimum clock frequency of XC95 series CPLD is 100MHz (0.01 µS clock pulse) [8] and maximum clock frequency of MPMC used for most TSS is 50 MHz (0.02 µS clock pulse) [9].

So for transition, MPMC requires 0.44 µS delay.

And CPLD requires 0.04 µS delay.

Call handling operation from on hook condition to ring generation, call connection and disconnection requires 13 transitions. So MPMC requires 0.44 x 13 = 5.72 µS for a call handling and CPLD requires 0.04 x 13 = 0.52 µS for it.

In full traffic condition, MPMC has to go through sequential operations so 8 x 5.75 = 46 µS are required for handling simultaneous calls of all eight lines. CPLD requires only 0.52 µS for handling all calls in full traffic condition as all call handlings are implemented as concurrent processes in CPLD. So call handling speed increased 88 times in full traffic condition due to use of CPLD in HTSS. Racing problem is one of the most important problems in SPC/MPMC which is caused mainly by speech path response delay/call handling delay [1] and can be reduced by our proposed HTSS.

V. EXPERIMENT AND RESULTS

We designed VHDL codes for CHC, which works with MPMC for getting combined features of both CPLD and MPMC into HTSS. We designed two types of VHDL test benches for the testing of VHDL codes of our system. 1) Testbench1: Test bench for testing call handling using CHC 2) Testbench2: Test bench for the action of MPMC.

Fig.8.4 and Fig.8.5 shows the wave forms results from Testbench1 which indicates the events of on hook condition of a calling line, dialing, ring to called line and call establishment between calling and called line and call disconnect etc.

Fig.8.6 to Fig.8.9 shows the wave forms results from Testbench2. Here in HTSS, we proposed MPMC with CHC circuit for service handling operations. But we design VHDL test bench for simulation purpose, which provides the sequential operations that can be provided by MPMC instead of its actual use. Testbench2 tests services like DND, CF, OGB and STDB.

Activation/Deactivation of services is done with dialing the corresponding code in Table.3.1. Services when activated by the subscriber, testbench2 (MPMC) sets the corresponding flag in SSB and sends the activation tone through CHC to subscriber. If any one goes through the activated service, testbench2 (MPMC) executes the service by knowing the status of SSB and then it handover the action to CHC for further call processing. Different services execution steps designed in testbench2 are given as

1) DND service execution- when DND activated number is dialed by any other number, calling line got engage tone even the called line is on hook.

2) CF service execution- when CF activated number is dialed by any other number, call is forwarded to line number 1x.

3) OGB service execution- when OGB activated line dials 0, it got engage tone.

4) STDB service execution- when STDB activated line dials 0, it is connected to external/junction line and when it dials again 0, it got engage tone.

Waveforms shown in Fig.8.4 to Fig.8.9, indicates the process of activation/deactivation and execution for DND, CF, OGB and STDB facilities respectively.

VI. SUMMARY AND CONCLUSIONS

Introduction of this paper gives the idea about flexible and dedicated systems. It introduces advantages of CPLD using VLSI design for the implementation of flexible, high speed, low size and easily expandable/cascading TSS. Proposed HTSS combines SHC and CHC to give high speed call handling and enhanced services/facilities to subscribers. SHC provides all services of SPC TSS and design of SSB allow quick addition of new services to subscribers. CHC handles simultaneous call establishment. FSM is introduced for the design of CHC for a line and concurrent operations are obtained by using Structural Type of Modeling of FSM for eight lines CHC. Logical calculation of call handling delay proves the improvement of speed of HTSS up to 88 times than the speed of MPMC based TSS.

VII. APPENDIX.1

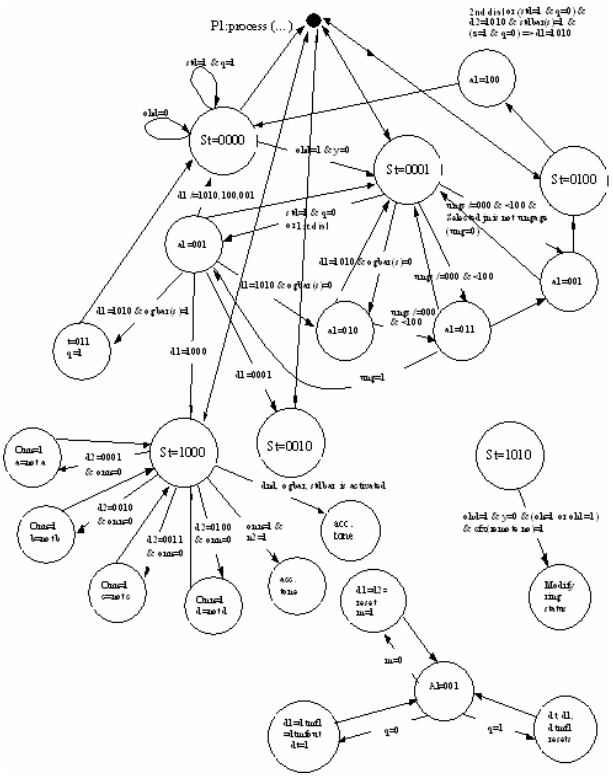

Fig.8.1. process p1 of FSM

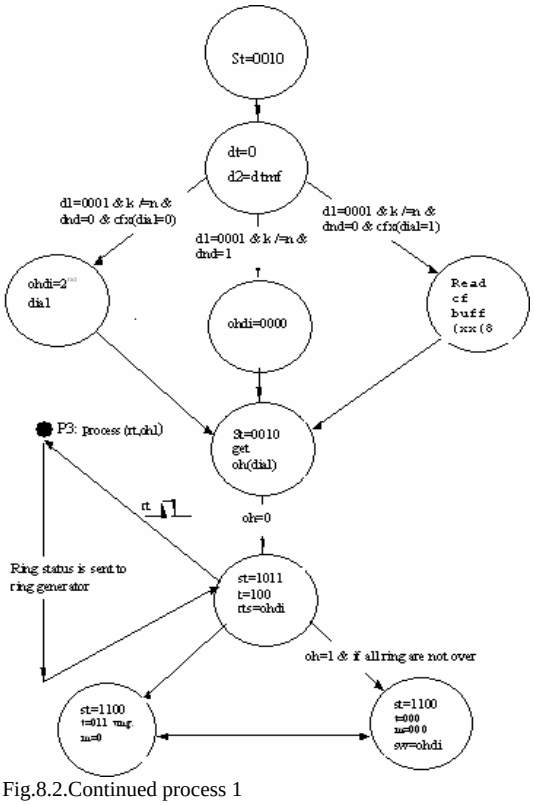

Fig.8.2.Continued process 1

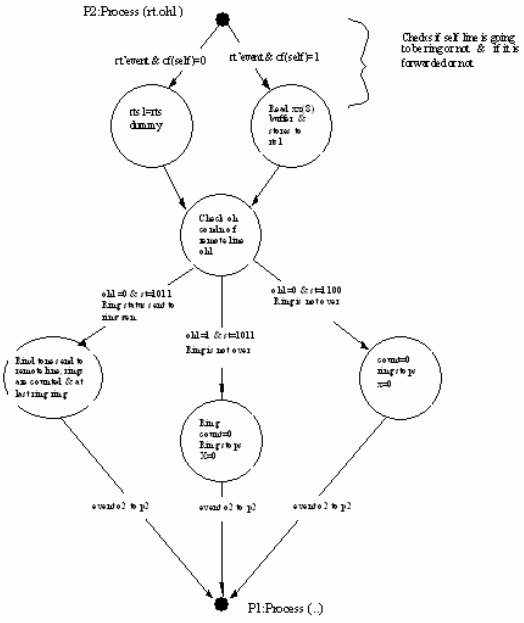

Fig.8.3. process p2 of FSM

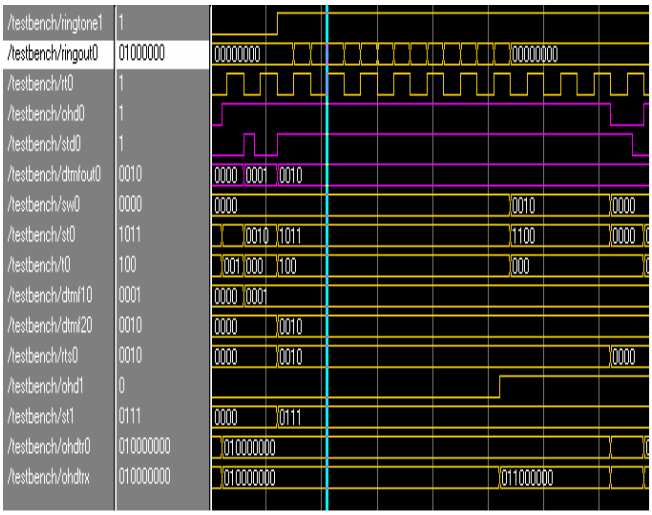

Fig.8.4. Simulation results of call from line 1 to line 2

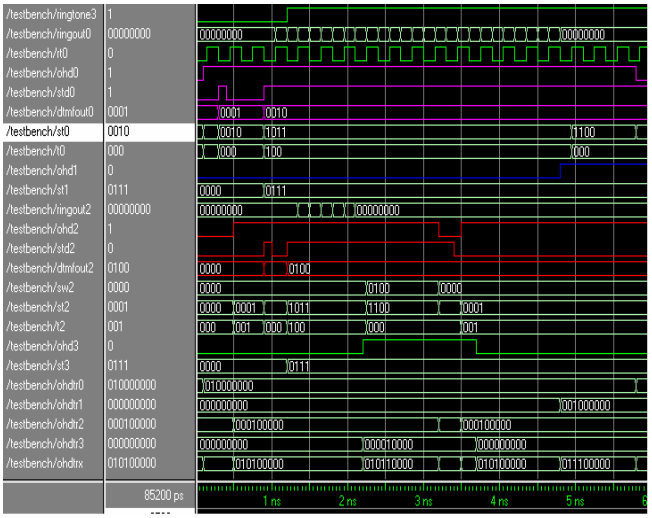

Fig.8.5. Simulation results of simultaneous call from line 1 to line 2 and line 3 to line 4

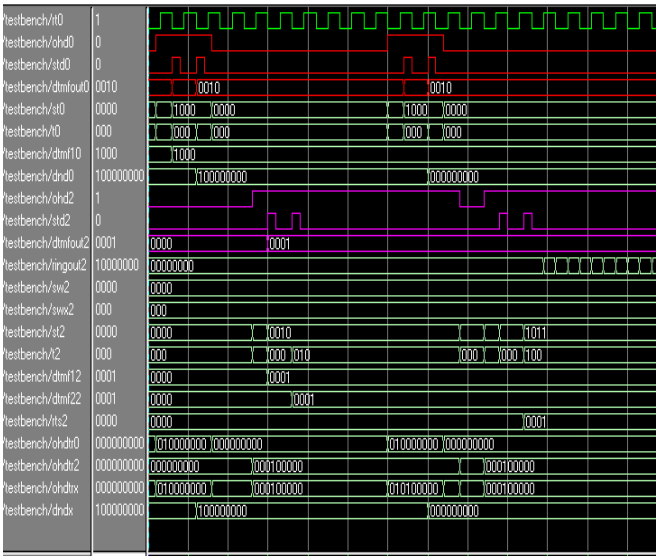

Fig.8.6. Simulation results of set and reset DND by line 1 and executed by call from line 2 to line 1

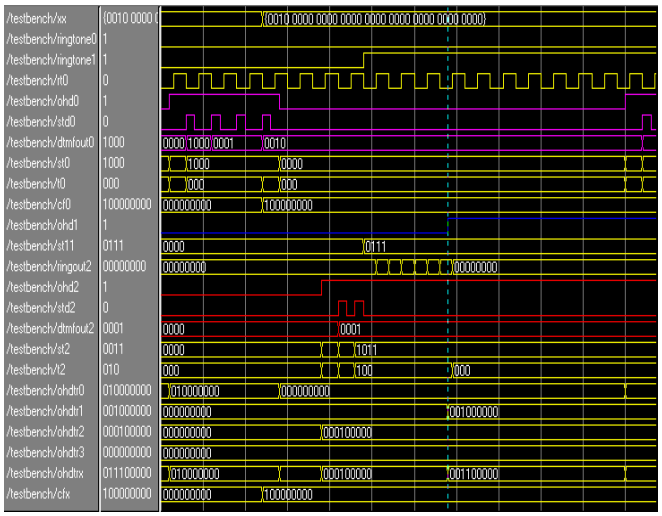

Fig.8.7. Simulation results of set CF from line 1 to line 2 and executed by call from line 3 to line 1, then call forwarded to line 2.

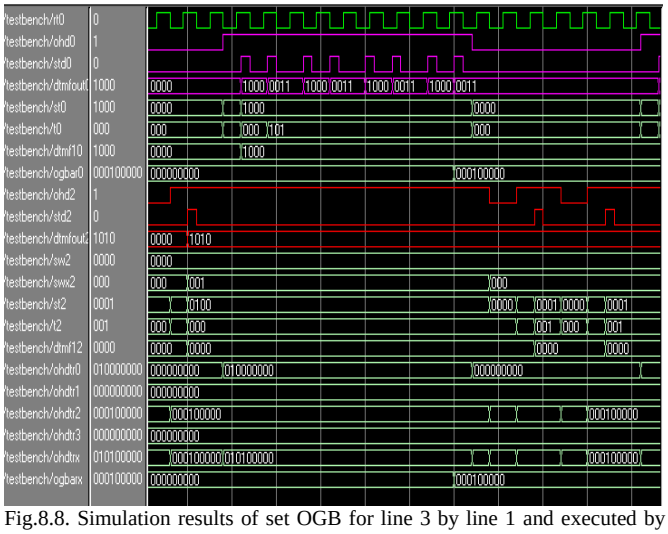

Fig.8.8. Simulation results of set OGB for line 3 by line 1 and executed by dialing 0 on that line 3

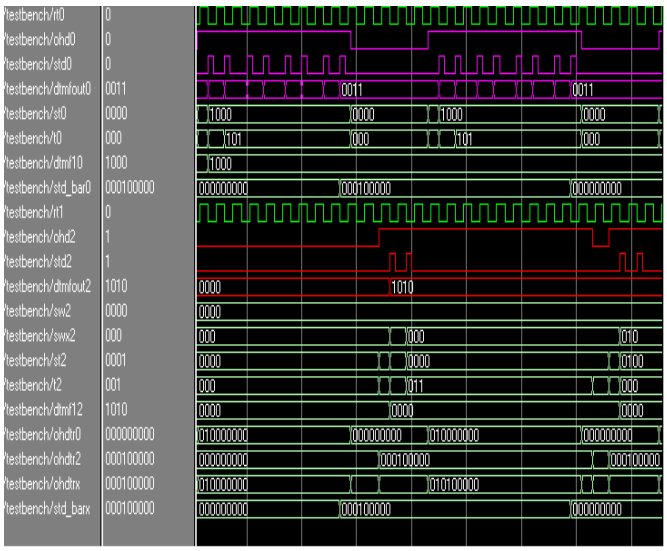

Fig.8.9. Simulation results of set and reset STDB for line 3 by line 1 and executed by dialing 00 on that line 3

VIII. REFERENCES

[1] Keiji Okada, Haruhisa Ichikawa, Toru Miyake and Norio Sato, "Microprocessor Application to Telephone Terminal Switching System Architecture", IEEE Transactions on Communications, Vol. COM-27 No. 7, PP. 1089-1095, July 1979.

[2] A Pariani, F. Natali and W. Rossi, "Innovative Developments in Dedicated Circuits for PABX Applications", IEEE Conference on Private Switching System and Networks, Pub No. 288, PP. 39-44, June 1988.

[3] "Xilinx Manual", *http://www.xilinx.com,* visited on March 2007.

[4] Ronnie Lee Bennett and George E. Policello I1, "Switching Systems in the 21st Century", IEEE Communications Magazine, Vol. 31 No. 30, PP. 24-28, March 1993.

[5] Katsumi Maruyama, Nobuyuki Watanabe, Keiich Koyanagi, Toshihiro Kai, Shuji Tomita, "A Concurrent Object-Oriented Switching Program in Chill", IEEE Communications Magazine, PP. 60-68, January 1991.

[6] Syed Riffat Ali, "Software Patching in the SPC Environment and its Impact on Switching System Reliability", IEEE Journal on Selected Areas Communications, Vol. 9, No. 4, PP. 626-631, May 1991.

[7] Syed Riffat Ali, "Implementation of Firmware on SPC Switching Systems", IEEE Journal on Selected Areas Communications, Vol. 6, No. 8, PP. 1401-1404, 1988.

[8] "Semiconductor Complex Limited (Data sheets of XC 95 series CPLD)", *http://sclindia.com*, visited on March 2007.

[9] "Digital Telecommunication Switching Exchange, Particularly for Private Systems (PABX)", *http://www.wipro.in* , visited on April 2007.

IX. BIOGRAPHIES

K.P.Rane was born in Jalgaon in the Maharashtra (India) on May 23, 1973. He graduated from J.T.M. COE Faizpur under North Maharashtra University, Jalgaon and obtained M.E. from WCE Sangali under Shivaji University, Kolhapur.

His employment experience included J.T.M. COE Faizpur. His special field of interest included Image Mosaic and Processing and VLSI Design.

S.V.Patil was born in Chinawal in the Maharashtra (India) on January 19, 1967. He graduated from S.S.G.M.COE Shegaon under Amarawati University, Amarawati and completed M.E. from MNREC Allahabad under Allahabad University Allahabad.

His employment experience included J.T.M. COE Faizpur. His special field of interest included VLSI Design.

A.M.Patil was born in Jalgaon in the Maharashtra (India) on September 14, 1965. He graduated from S.S.G.M.COE Shegaon under Amarawati University, Amarawati and completed M.E. from COE Badnera under Amarawati University, Amarawati.

His employment experience included J.T.M. COE Faizpur. His special field of interest included Image Processing.