An Efficient Low Power VLSI Architecture for Viterbi Decoder using Null Convention Logic

T.Kalavathidevi
Senior Lecturer, Department of EEE
Kongu Engineering College, Perundurai,Erode.

V.AnishKumar
PG, Scholar,Applied electronics
Kongu Engineering College, Erode

P.Sakthivel,
Associate Professor, Department of EEE
Velalar College of Engineering, Perundurai,Erode

ABSTRACT
In 3G mobile terminals the Viterbi decoder consumes approximately one third of the power consumption of a base band mobile transceiver. Viterbi decoders employed in digital wireless communications are complex and dissipate large power. In this paper, to reduce the power consumption, and to increase the speed, an asynchronous technique that is delay insensitive null convention logic (NCL) for Viterbi decoder using dual rail signal is proposed. NCL reduces the dynamic power consumption in terms of reducing the switching activity and also it reduces the glitch power significantly, thereby achieving the lower power. The Viterbi decoder consists of branch metric unit, add compare and select unit and the survivor path memory unit. It is designed in circuit level using null convention logic and simulated using tanner tool in 1.25µm technology 3v vdd and a frequency of 2GHz. The simulation results shows the power consumption of the Viterbi decoder using NCL is 36.14mw, delay of the Viterbi decoder is 6.150ns and the number of transistors required to design the Viterbi decoder is 2058. The Viterbi decoder designed using the null convention logic provides 26% of lower power consumption comparing with that of the CMOS logic.

Keywords: Viterbi, Null conventional logic, tanner, low power, high speed

1. INTRODUCTION
The Viterbi decoding algorithm, proposed in 1967 by Viterbi, is a decoding process for convolutional codes in memory-less noise. The algorithm can be applied to a host of problems encountered in the design of communication systems. The Viterbi Algorithm (VA) finds the most-likely state transition sequence in a state diagram, given a sequence of symbols. The Viterbi algorithm is used to find the most likely noiseless finite-state sequence, given a sequence of finite-state signals that are corrupted by noise.

In order to reduce the power consumption, and increase the speed, an asynchronous technique that is Delay Insensitive Null Convention Logic (NCL) for Viterbi Decoder (VD) and its Encoder using Dual rail signal [5] is proposed in this paper. In VLSI design the major cause for the power dissipation is the dynamic power dissipation about 80 to 90 percent of total power dissipation. NCL reduces the dynamic power consumption in terms of reducing the switching activity and also it reduces the Glitch power significantly, thereby achieving the lower power. The basic Viterbi algorithm [12] was applied in digital communication systems, speech and character recognition. It focused on the operations and the practical memory requirement to implement the Viterbi algorithm in real-time. Based on the data generated and decoded [10] from the zero Hamming distance path, unnecessary computations in the Viterbi decoder was avoided. Speed and power were not considered. Low-power bit-serial Viterbi decoder chip [1] for the code rate r = 1/ 3 and the constraint length K = 9 (256 states) was discussed. The add-compare-select (ACS) module was based on the bit-serial arithmetic and implemented with the pass transistor logic circuit. The Scarce state transition (SST) scheme [14] employed a simple pre decoder followed by a pre encoder to reduce the transitions of the Viterbi decoder.

2. PROPOSED METHODOLOGY
2.1 Null Convention Logic
Null Convention Logic (NCL) provides an asynchronous design methodology [5] employing dual-rail signals, to incorporate data and control information into one mixed path. NCL is a symbolically complete logic which expresses process completely in terms of the logic itself and inherently and conveniently expresses synchronous digital circuits. A dual-rail signal, D, consists of two wires, D0 and D1, which may assume any value from the set {DATA0, DATA1, NULL}. The DATA0 state (D0 = 1, D1 = 0) corresponds to a Boolean logic 0, the DATA1 state (D0 = 0, D1 = 1) corresponds to a Boolean logic 1, and the NULL state (D0 = 0, D1 = 0) corresponds to the empty set meaning that the value of D is not yet available. The two rails are mutually exclusive, so that both rails can never be asserted simultaneously; this state is defined as an illegal state. NCL uses two completeness criteria to achieve its delay-insensitive behavior: symbolic completeness of expression and completeness of input. Threshold gates with hysteresis are used to build NCL circuits. These gates are constructed directly at the transistor level, independent of basic Boolean gates.

2.2 Threshold gates with hysteresis
NCL gates have both set and hold equations, where the set equation determines when the gate will become asserted and the hold equation determines when the gate will remain asserted once it has been asserted. The general equation for an NCL gate with output Z is: Z = set + (Z- • hold), where Z is the previous output value and Z is the new value. Take the TH23 gate for example as shown in Fig. 1. The set equation is AB + AC + BC and the hold equation is A + B + C; therefore the gate is asserted when at least 2 inputs are asserted and it then remains asserted until all inputs are deasserted.

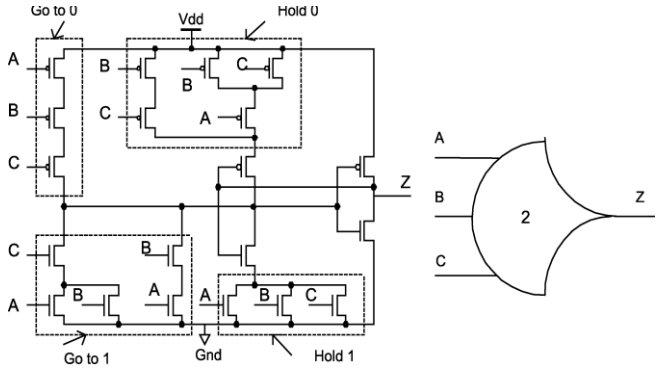

Figure 1. Schematic and symbol of TH23

3. DESIGN OF VITERBI DECODER USING NCL

The Viterbi decoder consists of the Branch Metric Unit, Add compare and Select Unit, and the Survivor Path Memory Unit. This chapter explains the internal blocks of the all the three units and how that blocks are designed using the Null Convention Logic.

3.1 Branch Metric Unit

The Branch Metric Unit (BMU) consists of two input Exor gate and three bit counter. The branch metric computation block compares the received code symbol with the expected code symbol and counts the number of differing bits. If the received sequence and expected sequence are different then the output of the exor gate becomes high and the number of 1's counted using the counter. The block diagram of BMU using NCL is shown in Figure 2. The NCL EXOR gate has two dual rail inputs are X (X^0, X^1) and Y (Y^0, Y^1) and a single dual rail output is Z (Z^0, Z^1). 3-bit counter is designed by cascading the T-FF and the output of the one flip flop is given as clock input for the next flip flop. Further the T input for all the flip flops are tied to HIGH input. The preset and clear input is used to make the counter working as asynchronous counter. Initially the counter output q3 ($q3^0$, $q3^1$) q2 ($q2^0$, $q2^1$) q1 ($q1^0$, $q1^1$) is = "10 10 10" when the preset='1' and clear='1'. Now if the first clock is applied then the counter starts counting the clock cycle as "001". Here the Exor gate output is the clock input of 3 bit counter. For the next clock cycle the output of the counter q3 ($q3^0$, $q3^1$) q2 ($q2^0$, $q2^1$) q1 ($q1^0$, $q1^1$) is "01 01 10". Similarly the counter will count all the clock cycles and it will go to"000" when reaches '111".

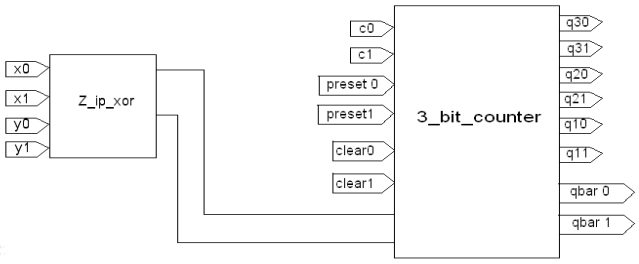

Figure 2.Block Diagram of BMU Using NCL

3.2 Add Compare and Select Unit

The Add Compare Select Unit (ACSU) which adds the BMs to the corresponding Path Metrics(PM), compares the new PMs, and then stores the selected PMs in the Path Metric Memory (PMM); at the same time, the ACSU stores the associated survivor path decisions in the Survivor Memory Unit (SMU). The PM of the survivor path of each state is updated and stored back into the PMM. Each butterfly wing is usually implemented by a module called ACS module. The output from the BMU and PM are processed as input to the two adder units.The two dual rail 3 bit inputs of the NCL adder units area a3 ($a3^0$, $a3^1$), a2 ($a2^0$, $a2^1$), a1 ($a1^0$,$a1^1$) and b3 ($b3^0$,$b3^1$), b2 ($b2^0$,$b2^1$), b1 ($b1^0$, $b1^1$). The output of the adder units are sum bits that is s3 ($s3^0$, $s3^1$), s2 ($s2^0$, $s2^1$), s1 ($s1^0$, $s1^1$) and carry bit that is cout ($cout^0$, $cout^1$). The comparator unit has two dual rail four bit inputs that can be obtained from the output of two adder units. The inputs are a3($a3^0$,$a3^1$), a2($a2^0$,$a2^1$), a1($a1^0$,$a1^1$), a0($a0^0$,$a0^1$) and b3($b3^0$,$b3^1$), b3($b2^0$,$b2^1$), b1($b1^0$,$b1^1$), b0($b0^0$,$b0^1$). The outputs are LT(LT^0,LT^1), EQ(EQ^0,EQ^1) and GT(GT^0,GT^1). When the value of a is less than the value of b then the LT output is high that is LT^0=0 and LT^1=1. When the value of a is greater than the value of b then the GT output is high that is GT^0=0 and GT^1=1. When the values of a and b are equal then the EQ output is high that is EQ^0=0 and EQ^1=1.The selector unit consists of four 2:1 multiplexor. The dual rail select input is s (s^0, s^1) which is from the LT output of the comparator. The dual rail two 4 bit inputs are from the output of adder units and the four bit outputs are f4($f4^0$,$f4^1$), f3($f3^0$,$f3^1$), f2($f2^0$,$f2^1$) and f1($f1^0$,$f1^1$).

3.3 Survivor Memory Unit

The Survivor memory unit is designed by using the serial in serial out shift register and the length of the shift register depends on the length of the convolution encoder. The Survivor memory unit consists of four 4 bit serial in serial out shift register as in figure 3. The two dual rail inputs of the shift register are clk (clk0, clk1) and d (d0, d1). When the clock pulse is positive then the data present in the d is transferred to the output of flip flop and for each positive clock cycle the path metric value shifted right from one register to another.

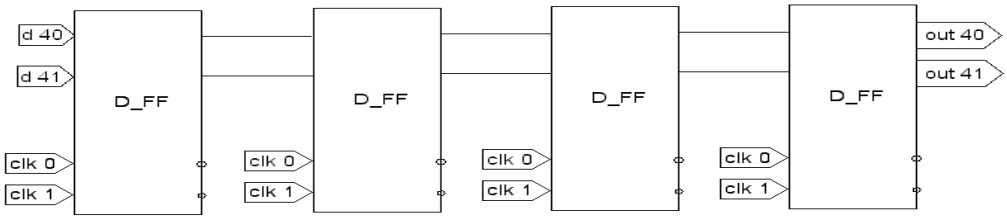

Figure 3. Single Stage of the Survivor Memory Unit

3.4 Integrated design of Viterbi Decoder

The block diagram of Viterbi decoder with NCL and CMOS Logic are designed in Tanner tool is shown in Figure 4. It consists of Branch metric unit, ACS unit and Survivor memory unit. Here two Branch metric units are used since two possible changes from one state to another. This BMU calculates the branch metric between the Expected sequence and the received sequence. Then adder unit adds branch metric with the previous path metric and comparator compares the two paths and select the less path using selector and survivor memory unit stores the path metric value and its corresponding states. The 2:1 multiplexor and 2 bit shift register used to get the decoded output.

3.5 Existing Method using CMOS logic

The Viterbi Decoder using CMOS circuits is designed by integrating all the units like BMU, ACSU and SMU. Initially assumed PM ”00” and the first of the 16-message bit VD “11” are the inputs of BMU. The first BMU calculates the differing bit as ‘2’ and the next BMU is calculated as ‘0’. The output of the BMU is added with the previous PM and the obtained output is the new PM for the next branch in the VD. The ACSU selects the minimum path as “0”. The register of the SMU stores the required decision path as “0”.

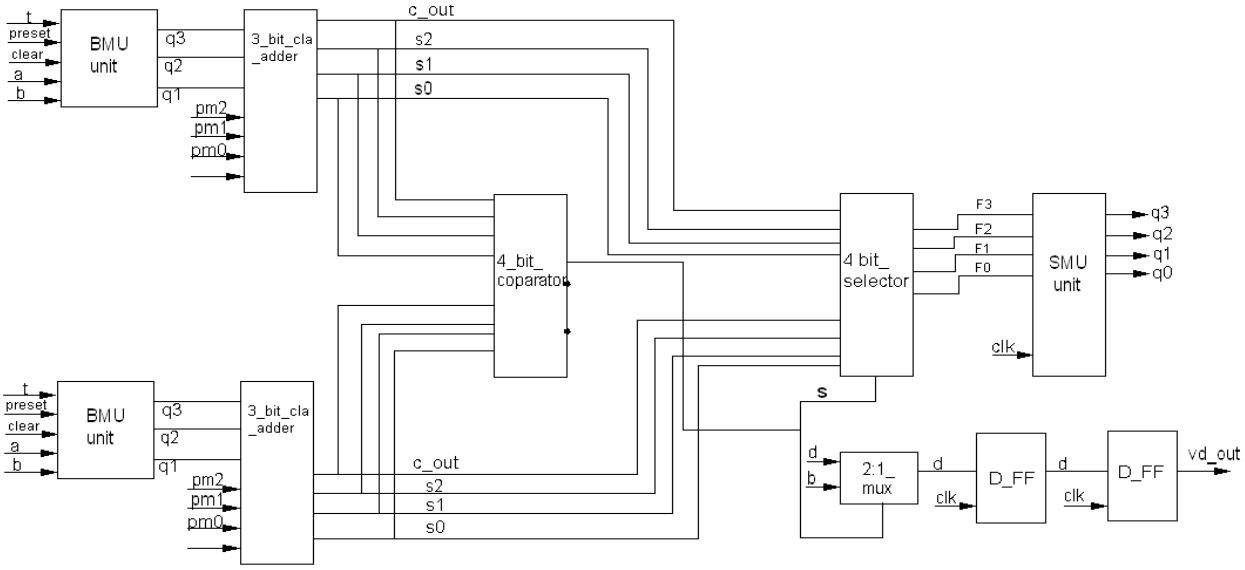

Figure 4. Internal circuit diagram of Viterbi decoder

4. EXPERIMENTAL RESULTS AND DISCUSSIONS

The viterbi decoder is designed using the Null Convention Logic and the simulation results are verified using TANNER TOOL (SCH and TSPICE) in the 1.25µm technology, 3V Vdd and at a frequency of 2GHz.The output waveform of the Viterbi decoder using NCL is shown in figure 6. The dual rail inputs of the Viterbi decoder are the received sequence and the expected sequence. The dual rail output of the Viterbi decoder is VD_out^0 and VD_out^1. The block diagram of Viterbi decoder uses two branch metric units since each state have two branches in the trellis. Here the received sequence is a=c=”11 01 11” and the Expected sequence for the first branch metric is b=”00 10 01” and expected sequence for the second branch metric is d=”11 01 10” and the decoded output sequence is VD out=”11 01 10”.

For the Branch metric unit the two dual rail inputs are $X(X^0, X^1)$ and $Y(Y^0, Y^1)$. The dual rail value $Z(Z^0, Z^1)$ be the output of the Exor gate that is the Hamming distance between the expected sequence and the received sequence that can be counted using the counter. The output of the branch metric units are $Q3(Q3^0, Q3^1)$, $Q2(Q2^0, Q2^1)$ and $Q1(Q1^0,Q1^1)$ which denotes the branch metric value. For the Addcompare select unit the two dual rail inputs of half adder are $X (X^0, X^1)$ and $Y (Y^0, Y^1)$. The two dual rail outputs of half adder are sum (sum^0, sum^1) and carry c (c^0, c^1). When $X^0=1$, $X^0=0$ and $Y^0=0$, $Y^1=1$ then $sum^0=0$, $sum^1=1$ and $c^0=1$, $c^1=0$. The four bit dual rail inputs of the four bit comparator are a4 $(a4^0, a4^1)$, a3 $(a3^0, a3^1)$, a2 $(a2^0, a2^1)$ a1 $(a1^0, a1^1)$ and b4 $(b4^0, b4^1)$, b3 $(b3^0, b3^1)$, b2 $(b2^0, b2^1)$, b1 $(b1^0, b1^1)$. The three dual rail outputs of four bit comparator are LT (LT^0, LT^1), GT (GT^0, GT^1) and EQ (EQ^0, EQ^1). When the inputs of four comparator are as follows a4a3a2a1=”1100” and b4b3b2b1=”0101” then the output becomes LT=0, GT=1 and EQ=0. When the inputs of four comparator are as follows a4a3a2a1=”0110” and b4b3b2b1=”1101” then the output becomes LT=1, GT=0 and EQ=0. When the inputs of four comparator are as follows a4a3a2a1=”1101” and b4b3b2b1=”1101” then the output becomes LT=0, GT=0 and EQ=1. The less than output of the comparator is used to select the minimum path. The dual rail inputs of the selector unit are a4 $(a4^0, a4^1)$, a3 $(a3^0, a3^1)$, a2 $(a2^0, a2^1)$,a1 $(a1^0, a1^1)$ and b4 $(b4^0, b4^1)$, b3 $(b3^0, b3^1)$, b2 $(b2^0, b2^1)$, b1 $(b1^0, b1^1)$. The dual rail select inputs are s^0 and s^1 . the dual rail

outputs of the selector unit are F4 ($F4^0$, $F4^1$), F3 ($F3^0$, $F3^1$), F2 ($F2^0$, $F2^1$) and F1 ($F1^0$, $F1^1$). When the signal value of the select input is boolean logic '1' then the value present in the a input is tranferred to the output. When the select input is boolean logic '0' then the b input value is transferred to the output F. For want of space all the results could not be shown .

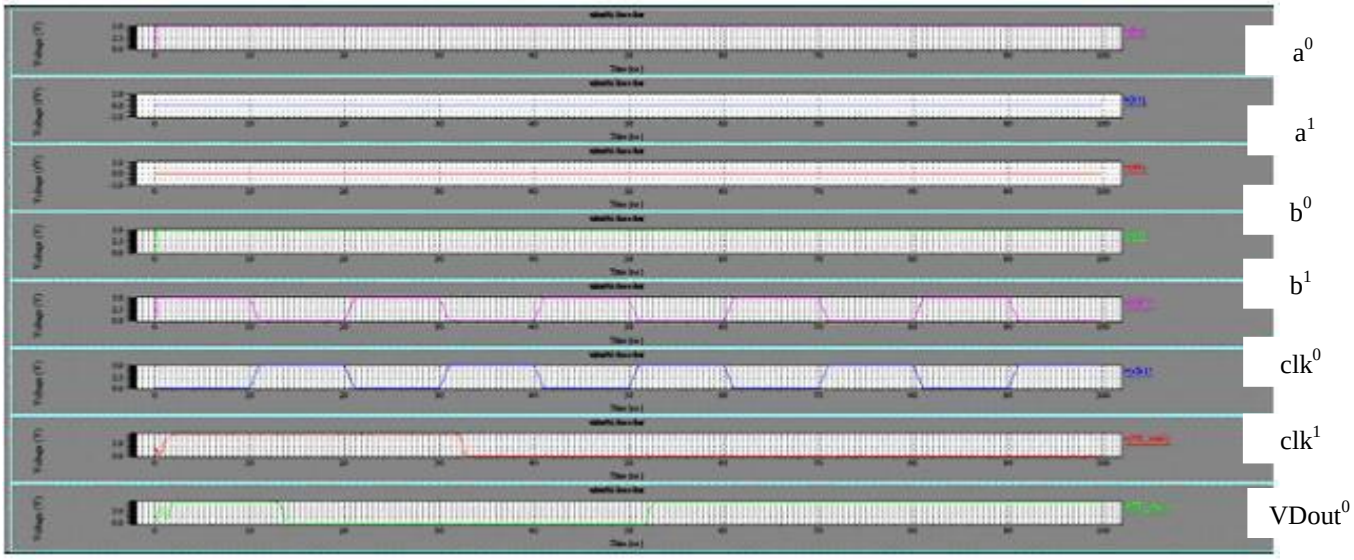

Figure5. Output waveform of Viterbi Decoder Using NCL

4.1 Comparison of performance

The Viterbi decoder is designed in circuit level using Null Convention Logic and simulated using Tanner tool in 1.25µm technology 3V Vdd and a frequency of 2GHz. The simulation results shows the power consumption of the Viterbi Decoder using NCL is 36.14mW, delay of the Viterbi decoder is 6.150ns and the number of transistors required to design the Viterbi Decoder is 2058. The Viterbi Decoder designed using the Null Convention Logic provides 26% of Lower power consumption comparing with that of the CMOS Logic. From theliterature survey though many works are designed in related to the viterbi decoder,for comparison purpose the same design is also implemented in CMOS logic. Hence the results prove that the proposed NCL logic has low power,high speed and low area.

Table1.Performance comparison

S.No.	Performance comparison		
	Viterbi decoder	***NCL***	***CMOS***
1	Transistor Count	2058	3632
2	Power Consumption	36.14mW	48.56mW
3	Delay	6.150ns	10.650ns

5. CONCLUSION

The Viterbi decoder is designed in circuit level using Null convention Logic and simulated the design using TANNER tool in the 1.25um technology, 3V Vdd and at a frequency of 2GHz. The power consumption of the Viterbi Decoder using NCL is 36.14mW. The simulated result is compared with the Viterbi decoder designed using CMOS Logic. Simulation results proved that the power consumed by the Viterbi Decoder is 26% less and Speed is increased by 4.13% compared to CMOS logic based Viterbi Decoder. Obtained results indicate that Null Conventional Logic consumes less power and area compare to the other MOS Logic styles.

6. REFERENCES

[1] Chien-Ching Lin, Yen-Hsu Shih, Hsie-Chia Chang, and Chen-Yi Lee, 2005. Design of a Power-Reduction Viterbi Decoder for WLAN Applications, IEEE Transactions on Circuits and System-I: regular papers, 52(6), 321-328G.

[2] Irfan Habib, Ozgun Paker, and Sergei Sawitzki,2009, Design Space Exploration of Hard-Decision Viterbi Decoding:

[3] ann S. Yuanand Weidong Kuang, 2004, Teaching Asynchronous Design in Digital Integrated Circuits, IEEE transactions on education,47(3),397-404

[4] Injin He, Zhongfeng Wang, Zhiqiang Cui, and Li Li, 2009, Towards an Optimal Trade-off of Viterbi Decoder Design, IEEE conferecne,3030-3033

[5] Joshi M.V., Gosavi S., Jegadeesan V., Basu A., Jaiswal S., Al-Assadi W.K. and Smith S.C. 2007, NCL Implementation of Dual-Rail 2s Complement 8×8 Booth2 Multiplier using Static and Semi-Static Primitives, IEEE region 5 Technical Conference, April 20-21, Fayetteville,59- 64.

[6] Jun Jin Kong, Keshhab K Parhi., 2004 Low-Latency Architectures for High-Throughput Rate Viterbi Decoder, IEEE Transactions on VLSI System, 12(6), 642-651.

[7] Meilana Siswanto1, Masuri Othman, Edmond Zahedi,2006 VLSI Implementation of 1/2 Viterbi Decoder for IEEE P802.15-3a UWB Communication, IEEE ICSE2006 Proc., Kuala Lumpur, Malaysia,666 – 670.

[8] Qing Li, Xuan-zhong Li, Han-hong Jiang and Wen-hao He 2008, A High-Speed Viterbi Decoder, Fourth International Conference on Natural Computation IEEE.,p.p. 313-316.

[9] Yao Gang, Ahmet T., Erdogan, and Tughrul Arslan, 2006, An Efficient Pre- Traceback Architecture for the Viterbi Decoder Targeting Wireless Communication Applications,IEEE Transactions on Circuits and Systems-I: regular papers, 53(9),423-432

[10] Yun-Ching Tang, Do-Chen Hu, Weiyi Wei, Wen-Chung Lin and Hongchin Lin , 2009. A Memory-Efficient Architecture for Low Latency Viterbi Decoders, IEEE.335-338

[11] Ajay Dholakia, 1994. Introduction to Convolutional Codes with Applications**.** Kluwer Academic Publishers.

[12] G. Forney, 1973. The Viterbi Algorithm, Proceedings of the IEEE, 61(3),268-278.

[13] Dalia A., El-Dib and Elmasry M.I. 2004. Modified Register-Exchange Viterbi Decoder for Low-Power Wireless Communications, IEEE Transactions on Circuits and Systems I, ,51(2), 371- 378.

[14] Lang L, Tsui C.Y and Cheng R.S.1997. Low power soft output Viterbi decoder scheme for turbo code decoding, IEEE Conference-Paper, ISCAS '97,New York, USA, 24, 1369-1372.