

# Advanced Telecom Computing Architecture (ATCA)

Shri Ram KV, Prabhu Raj R, Subashvi V, Venkateshwaran K.  
VIT, Vellore, Andhra Pradesh.

**Abstract-**Advanced Telecom Computing Architecture (ATCA) overcomes the difficulties that are found in existing Telecom Architecture. Most importantly the problems like 1.Lack of Standardization 2.Lack of plug-and-play amongst vendor equipment 3.Proprietary hardware design, same lessons being learnt by vendors to do their own hardware design and 4. Lacks of interoperability among equipment of different vendors are averted when ATCA is in place. During the late 1990s and early 2000s the focus of network OEMs was the delivery of high-performance technology-leading systems to meet the demands for ever greater telecom bandwidths. Chassis (Framework) and backplane design were key differentiators for market-leading manufacturers. CompactPCI cannot meet the needs of the telecom industry for high-availability platforms with a wide range of system capacities from Gbit/s to Tbit/s. This article will deal about the architecture and benefits of using ATCA.

## I INTRODUCTION

The telecommunications industry has fallen on difficult times over the last few years, but faces an encouraging future with the advent of Advanced Telecommunications Computing Architecture (ATCA). It is the First open industry specification for carrier grade equipment incorporating high-speed switched fabric technology capable of switching and processing 2.5 Terabits Per Sec in a single shelf. AdvancedTCA is targeted to requirements for the next generation of "carrier grade" communications equipment. This series of specifications incorporates the latest trends in high speed interconnect technologies, next generation processors, and improved Reliability, Availability and Serviceability (RAS).

Why do we call Advanced? The reason is the platform supports a variety of I/O sources and heterogeneous processing endpoints, thereby reducing integration costs, improving efficiency, and minimizing risks in design of next-generation applications. Also ATCA Promises to bring Plug and Play to carrier grade systems. (By defining high availability, chassis based platform that can scale from gigabits to terabits per sec). In Section 2, we describe the background and related details of ATCA. In Section 3, we explain the architecture of ATCA along with mezzanine cards. In Section 4, we explain the ATCA form factor. In section 5 we describe about the component of ATCA Shelf. In section 6 we deal about platform software. Section 7 and 8 deals about key design objectives and key benefits of ATCA. Section 9 talks about the applications. Section 9 talks about the status and improvements of ATCA and finally section 10 briefs on conclusion followed by references.

## II BACKGROUND & RELATED WORK

In September of 2001, the PICMG (PCI Industrial Computer Manufacturers Group) formed a subcommittee to define a next generation compute platform targeted at high throughput highly reliable applications. This subcommittee was

composed of 105 companies representing users and manufacturers of industrial and telecommunications compute equipment. This group had the goal of creating, reviewing and releasing a new specification by the end of 2002. The specification development was a good example of blitzkrieg engineering in which multiple teams focus on individual parts of the problem with the hope that in the end all the groups will converge with a usable solution. A core group of individuals was responsible for monitoring the progress of the teams and to watch for potential incompatibilities and roadblocks. The end result of this 12 month effort is Advanced Telecom Computing Architecture. This article is intended to provide some insight into the motivation behind the specification and an overview of the specification itself.

## III ARCHITECTURAL DETAIL OF ATCA

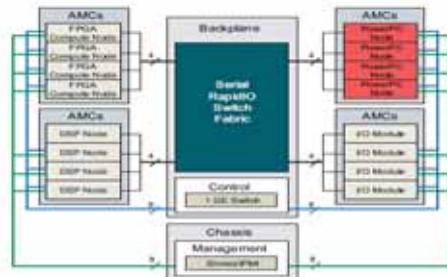


Fig.1, Architecture of ATCA

The architecture shown here makes use of AMC (Advanced Mezzanine Card). An Advanced Mezzanine Card (AdvancedMC or AMC) is a daughterboard that can be attached onto an ATCA Carrier Card or plugged directly into a TCA cabinet. AMCs provide a cost effective way to upgrade ATCA or μTCA based equipment as carriers seek to add more functions or channels to meet changing requirements.

AMC modules augment a baseline ATCA platform by extending it with individual hot swappable modules, providing OEMs with a versatile platform with reduced impact of component failures, and enabling service providers to scale, upgrade, provision, and repair live systems with minimal disruption to their network. The following schematics depict how a shelf and ATCA carrier card will look like.

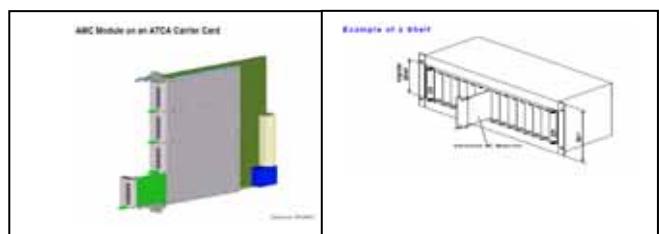


Fig.2, Shelf and ATCA- an External view

This Platform has many advantages that accelerate application development activities. The variety of heterogeneous AMCs allows developers to customize applications with options to plug in a wide array of processing elements. AMCs can be combined with carrier blades that provide RapidIO chip-to-chip and across-the-chassis connectivity, enabling seamless scaling from a single-sector system to, for example, multisector, multi-antenna, multicarrier base-station implementations.

#### IV. ADVANCEDTCA FORM FACTOR

ATCA form factor, with the front of the chassis on the left and the rear of the chassis on the right. Zone 1 supports power and control, and Zone 2 supports the base and fabric interfaces for data transport. Power to each slot is +/-48V with up to 200W per slot. The base interface supports triple-speed Ethernet, and the fabric interface is application specific. Both interfaces are optional

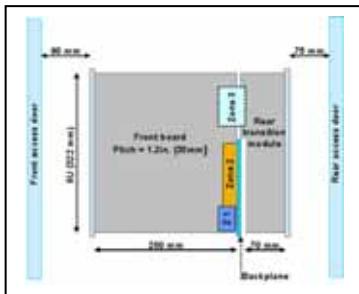


Fig.3, ATCA Form Factor

The ATCA backplane supports two to 16 slots. Further flexibility is added using the rear transition module (RTM) and mezzanine cards. The RTM sits behind the front boards and can be used for additional functionality or connectivity. The front board is connected to the rear transition module via Zone 3, but the base specification details no interfaces or connectors for Zone 3. Up to four fixed mezzanine cards can be plugged into a carrier front board, allowing ATCA to support smaller cards such as PCI and CompactPCI or interface-specific modules. Future ATCA systems will support hot-swappable advanced mezzanine cards (AMCs).

#### V. COMPONENTS OF ATCA SHELF

The ATCA platform has a number of elements that can be categorized based on their function within an integrated system:

1. Carrier blades
2. Switch blades
3. FPGA compute blades
4. Control-plane AMC modules
5. Data-plane AMC modules

These elements can be flexibly connected, combined, and integrated into high-performance systems that fit the size, complexity, and cost constraints of your unique application needs.

#### VI. PLATFORM SOFTWARE

The platform architecture can be diagrammatically represented by following way.

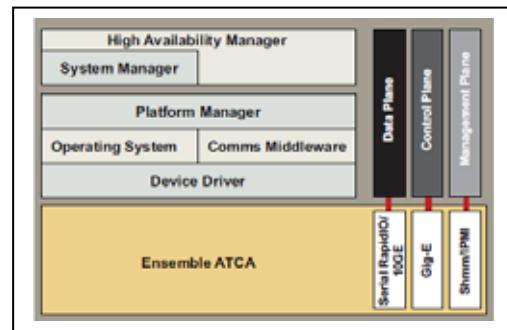


Fig.4, Platform Architecture of ATCA

The software identifies management, control, and data planes for discrete functionality:

1. In the management plane, the Shmm / IPMI controls and monitors temperature, current, and voltage, and interfaces with the Platform Manager and High Availability Manager software.
2. The data plane, using serial Rapid IO or 10 Gigabit Ethernet, supports up to 10 Gbps throughput and 112 ns of latency per switch (RapidIO).
3. Te control plane, using 1 Gigabit Ethernet, supports up to 1 Gbps bandwidth and latency in the 1 μs range.

Within the Platform Manager level, the operating system, the communications middleware, and the device driver for RapidIO are managed. This software brings up all hardware and software, including hardware component identification. The System Manager level handles run-time deployment and configuration, including initialization and monitoring, defining, generating and managing alarms, and defining and managing faults. The High Availability Manager software provides the hooks to enable the high availability of the application.

#### VII. KEY BENEFITS OF ATCA

**Increased flexibility:** ATCA has been designed to deliver significant flexibility through a choice of processors and interfaces, both internal and external. The same chassis and management blades can be used across a wide range of systems, thus reducing inventory, while each system can be optimized for the specific application with different line and switch blades.

##### *Reduced cost of ownership:*

Reduced COO is key in the current telecom environment, where maximizing return on investment is driving all business decisions. The use of ATCA significantly reduces the cost of R&D required to introduce a totally new system. The lower investment and the manufacturing savings from using a single chassis across multiple markets and customers will together lower support costs while driving down the total cost of ownership.

### VIII. KEY DESIGN OBJECTIVES OF ATCA

Key objectives in developing ATCA as a common platform were that it should have:

#### *High availability:*

- a. A. ATCA systems must be able to support five-nines availability (99.999%).
- A. To achieve this, dual redundant components are required throughout, with field-replaceable units and unified system management to support in-service upgrade and repair.

D.

#### *Scalability:*

- A. The system needs to be scaleable with both compute and I/O headroom.
- B. The internal fabric interfaces should be scaleable from Gbit/s to Tbit/s.

E.

#### *Flexibility:*

ATCA systems need to support multiple switch architectures and application-specific interfaces.

#### *Shelf Management:*

ATCA systems must include shelf management.

- A. This monitors and controls the ATCA boards and other field-replaceable units.
- B. Shelf management also controls the power, cooling, and interconnects across the system.
- C. This function is particularly important during failover to back up components and during the in-service installation of new components.

### IX. APPLICATIONS OF ATCA.

**Server:** For server applications such as media gateways and server farms, the ATCA system may include CPU blades, system managers, and base/Fiber-Channel switching cards.

**Networking:** For networking and telecom applications, the system would include networking blades, system managers, and base/Ethernet switching cards. In the short term, the majority of systems deployed will be in server and wireless networking applications. The modular system is more than just a hardware platform, however, and needs an operating system and application software. Many of these functions are now available as open-source applications

### X. IMPROVEMENTS AND STATUS

The ATCA base specification was approved in December 2002, and over 28 members exhibited ATCA products at Supercomm 2004. The PICMG has more than 600 members, and many now offer ATCA components. Several semiconductor companies, including Freescale, Intel, and NPU vendor EZchip Technologies, have already adopted ATCA as a platform for technology demonstrators and silicon evaluation. As other companies follow suit, ATCA systems will be appearing in development labs across the world.

### XI. CONCLUSION

ATCA is specifically developed for the telecommunications equipment market in order to address the

needs of a constantly evolving network infrastructure. This open-standards architecture brings many benefits to OEMs and Carriers:

- A. Promotes flexibility and innovation by providing OEMs and system integrators with an ecosystem of modular COTS components to choose from.
- B. Reduces vendor dependency and single vendor lock-in problems by providing a broader supplier ecosystem
- C. Accelerates time to market for OEMs
- D. Eases the challenge of building and upgrading infrastructure

### XII REFERENCES

- [1]. <http://www.narrabri.atnf.csiro.au/observing/pointing/>
- [2]. <http://www.compactpci-systems.com/news/Technology+Partnerships/8941>
- [3]. <http://www.intel.com/technology/atca/>
- [4]. [http://www.lightreading.com/topics.asp?node\\_id=1002](http://www.lightreading.com/topics.asp?node_id=1002)
- [5]. <http://www.freescale.com/webapp/sps/site/homepage.jsp?nodeId=02VS01>
- [6]. 833

# FPGA Based Hardware Co-Simulation of an Area & Power Efficient FIR Filter for Wireless Communication Systems

Lecturer-Rajesh Kumar, A.P-Dr. Swapna Devi, Prof. & Head- Dr.S. S. Pattnaik  
rajeshmehra@yahoo.com, NITTTR, Chandigarh

**Abstract-** In this paper FPGA based hardware co-simulation of an area and power efficient FIR filter for wireless communication systems is presented. The implementation is based on distributed arithmetic (DA) which substitutes multiply-and-accumulate operations with look up table (LUT) accesses. Parallel Distributed arithmetic (PDA) look up table approach is used to implement an FIR Filter taking optimal advantage of the look up table structure of FPGA using VHDL. The proposed design is hardware co-simulated using System Generator10.1, synthesized with Xilinx ISE 10.1 software, and implemented on Virtex-4 based xc4vlx25-10ff668 target device. Results show that the proposed design operates at 17.5 MHz throughput and consumes 0.468W power with considerable reduction in required resources to implement the design as compared to Coregen and add-shift based design styles. Due to this reduction in required resources the proposed design can also be implemented on Spartan-3 FPGA device to provide cost effective solution for DSP and wireless communication applications.

**Key Words:** FPGA, PDA, Simulation, Add/Shift, VHDL

## I. INTRODUCTION

Today's consumer electronics such as cellular phones and other multi-media and wireless devices often require digital signal processing (DSP) algorithms for several crucial operations[1]. Due to a growing demand for such complex DSP applications, high performance, low-cost Soc implementations of DSP algorithms are receiving increased attention among researchers and design engineers. There is a constant requirement for efficient use of FPGA resources [2] where occupying less hardware for a given system that can yield significant cost-related benefits:

- (i) Reduced power consumption;
- (ii) Area for additional application functionality;
- (iii) Potential to use a smaller, cheaper FPGA.

Finite impulse response (FIR) digital filters are common DSP functions and are widely used in multiple applications like telecommunications, wireless/satellite communications, video and audio processing, biomedical signal processing and many others. On one hand, high development costs and time-to-market factors associated with ASICs can be prohibitive for certain applications while, on the other hand, programmable DSP processors can be unable to meet desired performance due to their sequential-execution architecture [3]. In this context, reconfigurable FPGAs offer a very attractive solution that balance high flexibility, time-to-market, cost and performance. Therefore, in this paper, an important DSP function i.e. FIR filter is implemented on Virtex-4 FPGA. The impulse response of an FIR filter may be expressed as:

$$Y = \sum_{k=1}^K C_k x_k \quad (1.1)$$

where  $C_1, C_2, \dots, C_K$  are fixed coefficients and the  $x_1, x_2, \dots, x_K$  are the input data words. A typical digital implementation will require  $K$  multiply-and-accumulate (MAC) operations, which are expensive to compute in hardware due to logic complexity, area usage, and throughput [4]. Alternatively, the MAC operations may be replaced by a series of look-up-table (LUT) accesses and summations. Such an implementation of the filter is known as distributed arithmetic (DA).

## II. DISTRIBUTED ARITHMETIC

DISTRIBUTED ARITHMETIC (DA) is an efficient method for computing inner products when one of the input vectors is fixed. It uses look-up tables and accumulators instead of multipliers for computing inner products and has been widely used in many DSP applications such as DFT, DCT, convolution, and digital filters [4]. The example of direct DA inner-product generation is shown in equation 1 where  $x_k$  is a 2's-complement binary number scaled such that  $|x_k| < 1$ . We may express each  $x_k$  as

$$x_k = -b_{k0} + \sum_{n=1}^{N-1} b_{kn} 2^{-n} \quad (2.1)$$

where the  $b_{kn}$  are the bits, 0 or 1,  $b_{k0}$  is the sign bit. Now combining equation 1.1 and 2.1 in order to express  $y$  in terms of the bits of  $x_k$  then we get

$$Y = \sum_{k=1}^K C_k [-b_{k0} + \sum_{n=1}^{N-1} b_{kn} 2^{-n}] \quad (2.2)$$

The above equation 2.2 is the conventional form of expressing the inner product. Interchanging the order of the summations, gives us:

$$Y = \sum_{n=1}^{N-1} [\sum_{k=1}^K C_k b_{kn}] 2^{-n} + \sum_{k=1}^K c_k (-b_{k0}) \quad (2.3)$$

The above equation 2.3 shows a DA computation where the bracketed term is given by

$$\sum_{k=1}^K C_k b_{kn} \quad (2.4)$$

Each  $b_{kn}$  can have values of 0 and 1 so equation 2.4 can have  $2^K$  possible values. Rather than computing these values on line, we may pre-compute the values and store them in a ROM. The input data can be used to directly address the memory and the result. After  $N$  such cycles, the memory contains the result,  $y$ . As an example, let us consider  $K = 4$ ,  $C_1 = 0.45$ ,  $C_2 = -0.65$ ,  $C_3 = 0.15$ , and  $C_4 = 0.55$ . The memory must contain all

possible combinations ( $2^4 = 16$  values) and their negatives in order to accommodate the term

$$\sum_{k=1}^K C_k b_{kn} \quad (2.5)$$

which occurs at the sign-bit time. As a consequence,  $2 \times 2K$  word ROM is needed. Figure 1 shows the simple structure that can be used to compute these equations.. The S, signal is the sign-bit timing signal. The term  $x_k$  may be written as

$$x_k = \frac{1}{2}[x_k - (-x_k)] \quad (2.6)$$

and in 2's-complement notation the negative of  $x_k$  may be written as

$$-x_k = -\bar{b}_{k0} + \sum_{n=1}^{N-1} \bar{b}_{kn} 2^{-n} + 2^{-(N-1)} \quad (2.7)$$

where the over score symbol indicates the complement of a bit. By substituting equation 2.1 & 2.7 into equation 2.6, we get

$$x_k = \frac{1}{2}[-(\bar{b}_{k0} - \bar{b}_{k0}) + \sum_{n=1}^{N-1} (\bar{b}_{kn} - \bar{b}_{kn}) 2^{-n} - 2^{-(N-1)}] \quad (2.8)$$

In order to simplify the notation later, it is convenient to define the new variables as

$$a_{kn} = \bar{b}_{kn} - \bar{b}_{kn} \quad \text{for } n \neq 0 \quad (2.9)$$

and

$$a_{k0} = \bar{b}_{k0} - \bar{b}_{k0} \quad (2.10)$$

where the possible values of the  $a_{kn}$ , including  $n=0$ , are  $\pm 1$ . Then equation 2.8 may be written as

$$x_k = \frac{1}{2} \left[ \sum_{n=0}^{N-1} a_{kn} 2^{-n} - 2^{-(N-1)} \right] \quad (2.11)$$

By substituting the value of  $x_k$  from equation 2.11 into equation 1.1, we obtain

$$Y = \frac{1}{2} \sum_{k=1}^K C_k \left[ \sum_{n=0}^{N-1} a_{kn} 2^{-n} - 2^{-(N-1)} \right] \quad (2.12)$$

$$Y = \sum_{n=0}^{N-1} Q(b_n) 2^{-n} + 2^{-(N-1)} Q(0)$$

where

$$Q(b_n) = \sum_{k=1}^K \frac{C_k}{2} a_{kn} \quad \text{and} \quad Q(0) = \sum_{k=1}^K \frac{C_k}{2} \quad (2.13)$$

It may be seen that  $Q(b_n)$  has only  $2^{(K-1)}$  possible amplitude values with a sign that is given by the instantaneous combination of bits. The computation of  $y$  is obtained by using a  $2^{(K-1)}$  word memory, a one-word initial condition register for  $Q(0)$ , and a single parallel adder subtractor with the necessary control-logic gates.

### III. CIRCUIT DESCRIPTION

The basic LUT-DA scheme on an FPGA would consist of three main components as shown in figure1. These are input registers, 4-input LUT unit and shifter/accumulator unit.

**Input Registers:** To reduce the consumption of logic elements, RAM resources are used to implement the shift registers [1].



Figure1. LUT based DA implementation of 4-tap filter  
LUT Unit: To implement 4-input and 3-input LUT unit, an LUT table is used, which represent all the possible sum combinations of filter coefficients  
Figure1.

**Shifter and Accumulator Unit:** It consists of an accumulator and a shifter.

### IV. PROPOSED WORK

In DA implementation as the filter size  $K$  increases, the memory requirements grow exponentially as  $2K$ . This problem is solved in this paper by breaking up the filter into smaller base DA filtering units that require less memory sizes and, less area. If the  $K$  tap filter is divided into  $m$  units of  $k$  tap base units ( $K = m \times k$ ), then the total memory requirement would be  $m \times 2^k$  memory words. The total number of clock cycles required for this implementation is  $B + [\log_2(m)]$ ; the additional second term is the number of clock cycles required to implement an adder tree to calculate the sums of the units. Thus the decrease in throughput of this implementation is marginal. For instance, in this proposed design  $K = 41$ , instead of  $2^{41}$  in a full LUT implementation, we have chosen 12 partitions with  $k = 4$  for  $m = 5$  and  $k = 3$  for  $m = 7$  which would only require 136 memory words. In this proposed work a 41-tap low pass filter has been designed. The first step in design flow is to develop an optimized VHDL code using distributed Arithmetic Algorithm and implement it using black box of System generator to develop proposed model of design. Figure 2 shows the developed model of proposed design using various Simulink and System Generator blocks.

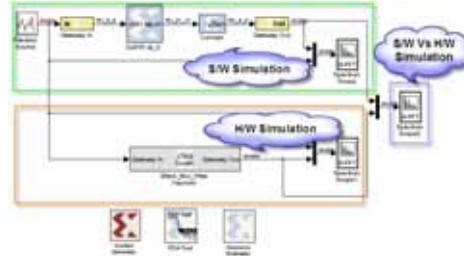


Figure2. Model for Hardware Co-simulation

The part of model enclosed in green boundary shows the software based simulation whose output can be seen in figure 3, part of model enclosed in orange boundary shows hardware based simulation whose output can be seen in figure 4 and spectrum scope in blue boundary shows the comparison between software and hardware based simulation whose output is shown in figure 5..

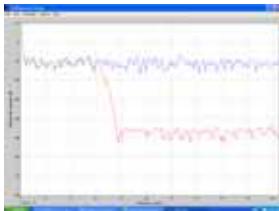


Figure3. Software Based Simulation

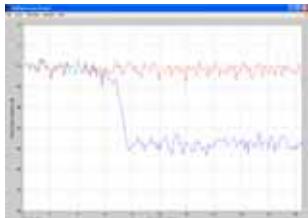


Figure4. Hardware Based Simulation



Figure5. S/W & H/W Based Simulation Comparison

The output wave form with green color in figure 5 means complete matching of software based simulation with hardware based simulation without errors.

## V. RESULTS

The proposed design is implemented on Virtex-4 based xc4vlx25-10ff668 target FPGA. Table 1 shows the comparison of proposed PDA design with the published add-shift and coregen based PDA [7] implemented on Virtex-4 device. It can be seen from the table that the throughput and performance of the proposed design are 17.50 MHz and 210 Msps respectively which are almost equal to other compared designs.

Design Style	Slices	LUTs	FFs	Throughput (MHz)	Performance (Msps)
Add-Shift	2154	1719	4161	18.58	223
Coregen PDA	2475	3642	4748	18.50	222
Proposed PDA	1840	3467	2985	17.50	210

Table 1 Virtex-4 Based Comparison PDA, Coregen & Add/Shift

Figure 6 shows the comparison of area utilization between add/Shift, PDA (coregen) and proposed PDA (PPDA) for 41 tap filter designs. It can be observed that the PPDA uses considerably less amount of resources on the target device as compared to other compared designs. Due to this reduction in required resources the proposed design can be implemented on Spartan-3 FPGA as shown in table2.

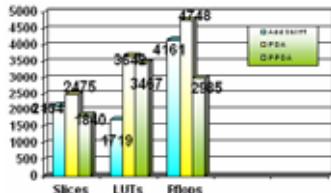


Figure6. Area Comparison of Add Shift and PDA with PPDA

Table2. Spartan-3 based implementation

Design Style	Slices (R/A)*	LUTs (R/A)*	FFs (R/A)*	Throughput (MHz)	Performance (Msps)
Proposed PDA	1840/1920	3467/3840	2985/3840	8.77	105.22
MACC Parallel	2046/1920	3193/3840	1323/3840	-	-
Add-Shift	2154/1920	1719/3840	4161/3840	-	-
Coregen PDA	2475/1920	3642/3840	4748/3840	-	-

(R/A)\*: Resources required / Resources available on target FPGA

Table 3 shows that the proposed design consumes total power of 0.468W at 31.3 degrees C junction temperature.

Table3. Power Consumption

Name	Value	Used	Total Available	Utilization (%)
Clocks	0.04479 (W)	1	21504	16.2
Logic	0.00000 (W)	5477	-	-
Registers	0.00000 (W)	8238	-	-
I/Os	0.00000 (W)	27	450	6.0
DCMs	0.00000 (W)	0	8	0.0
<b>Total Quiescent Power</b>	<b>0.42292 (W)</b>			
<b>Total Dynamic Power</b>	<b>0.04479 (W)</b>			
<b>Total Power</b>	<b>0.46772 (W)</b>			
Junction Temp	31.3 (degrees C)			

## VI. CONCLUSIONS

In this paper, a Parallel Distributed Arithmetic algorithm for high performance reconfigurable FIR filter is presented to enhance the area & power efficiency. The proposed design is taking optimal advantage of look up table structure of target FPGA. The throughput and performance of the proposed design are 17.50 MHz and 210 Msps respectively with considerable amount of reduction in used resources. Due to this reduction in required resources the proposed design can be implemented on Spartan-3 FPGA device to provide cost effective solution for DSP and wireless communication applications.

## VII. REFERENCES

- [1]. D.J. Allred, H. Yoo, V. Krishnan, W. Huang, and D. Anderson, "A Novel High Performance Distributed Arithmetic Adaptive Filter Implementation on an FPGA", in Proc. IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP '04), Vol. 5, pp. 161-164, 2004
- [2]. K.N. Macpherson and R.W. Stewart "Area efficient FIR filters for high speed FPGA Implementation", IEE Proc.-Vis. Image Signal Process., Vol. 153, No. 6, Page711-720, December 2006.
- [3]. Patrick Longa and Ali Miri "Area-Efficient FIR Filter Design on FPGAs using Distributed Arithmetic", pp248-252 IEEE International Symposium on Signal Processing and Information Technology,2006.
- [4]. S. A. White, "Applications of distributed arithmetic to digital signal processing: A tutorial review," IEEE ASSP Magazine, vol. 6, pp. 4-19, July 1989.
- [5]. Prithviraj Banerjee, Malay Haldar, David Zaretsky, Robert Anderson, "Overview of a compiler for synthesizing Matlab programs onto FPGAs",IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Page 312-324 Vol. 12, No. 3, March 2004.
- [6]. Matthew Ownby and Dr. Wagdy H. Mahmoud "A Design methodology for implementing DSP with Xilinx System Generator for Matlab", Page 404-408 IEEE 2002.
- [7]. Shahnam Mirzaei, Anup Hosangadi, Ryan Kastner "FPGA Implementation of High Speed FIR Filters Using Add and Shift Method", pp 308-313 in IEEE International conference on Computer Design, ICCD, 2006.
- [8]. Heejong Yoo and David V. Anderson "Hardware-Efficient Distributed Arithmetic Architecture for High-Order Digital Filters", ppV125-128 in Proc. IEEE , ICASSP 2005.
- [9]. Steve Zack, Suhel Dhanani" DSP Co-Processing in FPGAs Embedding High Performance, Low-Cost DSP Functions" WP212 (v1.0) March 18, 2004.

# Chip Architecture for Data Sorting using Recursive Algorithm

Megha Agarwal, Indra Gupta

Department of Electrical Engineering, Indian Institute of Technology, Roorkee, Uttarakhand-247667, India

**Abstract –** This paper suggests a way to implement recursive algorithm on hardware with an example of sorting of numeric data. Every recursive call/return needs a mechanism to store/restore parameters, local variables and return addresses respectively. Also a control sequence is needed to control the flow of execution as in case of recursive call and recursive return. The number of states required for the execution of a recursion in hardware can be reduced compared with software. This paper describes all the details that are required to implement recursive algorithm in hardware. For implementation all the entities are designed using VHDL and are synthesized, configured on Spartan-2 XC2S200-5PQ208.

**Index Terms -** Binary search tree, Field programmable gate arrays (FPGA), Recursive Algorithms, Very high-speed integrated circuits hardware description language (VHDL)

## I. INTRODUCTION

Recursion is an approach to write algorithms for repetitive tasks. In recursion the whole problem is decomposed into smaller sub-problems which are exactly similar to the original problem. Recursion can be efficiently used in optimization problems, non-linear data structures algorithms like binary trees, searching in dictionary, recursive filter, data compression etc. Many examples demonstrate advantages of recursion [1]. However, this technique is not always appropriate, particularly when a clear efficient iterative solution exists [2]. This is primarily due to the large amount of states that are accumulated during deep recursive calls because at each level of recursive call the current values of the parameters, local variables, return addresses etc of the subprogram are pushed on the stack until the subprogram is completed and these allocation records are popped out when the subprogram is reactivated [1]. This also consumes time to execute. But this execution time part can be taken care of by implementing the developed recursive algorithm in hardware e.g. FPGA.

If any high-level language which admits recursion is used then computer keeps the track of all the values of the parameters, local variables and return addresses. But if a high-level languages which does not admits recursion is used then a recursive procedure must be translated into a non-recursive procedure at the programmer hand. For example VHDL does not support for recursion. So, the algorithm is required to be converted into non-recursive algorithm. By combining the activation of a recursive subsequence of operations with the execution of the operations that are required by the respective algorithm, the recursion can be implemented in hardware much more efficiently [3]. The same event takes place when any recursive sub-sequence is being terminated, i.e. when control has to be returned to the point which is just after the last

recursive call and an operation of the executing algorithm that follows the last recursive call has to be activated.

The results obtained for some known methods for implementing recursive calls in hardware, have shown FPGA circuits to be significantly faster than software programs executing on general-purpose computers.

Recursion shows better performance in case of non-linear data structures problems [4]. For example a binary search tree can be constructed and used for sorting various types of data. In order to build such tree for a given set of values, the appropriate place for each incoming node in the current tree should be find out. Afterwards in order to sort the data, a special technique using forward and backtracking propagation steps that are exactly the same for each node is required. Thus, a recursive procedure is very efficient in this area. Other application to implement recursive algorithm in hardware may be in the field of lossless data compression such as Huffman coding, Dictionary search tree, recursive filter etc [2].

The design of data sorting are coded with VHDL [5], synthesized and configured onto Spartan-2 XC2S200-5Q208 FPGA, from Xilinx family [6]. The concept behind sorting is discussed in section 2 and results of synthesis and simulation are discussed in section 3. Here Modelsim 6.0d is used as simulation tool and ISE 8.1i project navigator is used as synthesis tool.

## II. IMPLEMENTATION OF SORTING ALGORITHM

To develop the sorting algorithm firstly a binary search tree should be constructed and after that inorder traversal of that binary search tree provides sorted data. Both of these functions can be effectively illustrated by recursive procedure. The concept used for the design is to divide the algorithm to a discrete number of modules (labeled  $z_i$ , i.e.  $z_1, z_2$  etc), each of which will have a number of discrete states (labeled  $a_i$ , i.e.  $a_1, a_2$  etc). Two separate stacks are used one to store the current module executed (the  $M_{stack}$ ), one to store the current state of the current module (the  $FSM_{stack}$ ). So, this needs support of control and execution mechanism. Control Unit is to transfer the control among the different modules and states so it needs to store and restore data from the different stacks [7].

Fig. 1 demonstrates the function of Recursive hierarchical finite state machine (RHFSM) which works as the Control Unit. This unit is having two stacks and one combinational circuit (CC). Functions of  $FSM_{stack}$  and  $M_{stack}$  are as discussed before. A combinational circuit is connected to the stacks and operates on the inputs (some of which are receives from the stacks) and produces the appropriate outputs [3]. It is

worth noting that both the stacks (M\_stack and FSM\_stack) use the same stack pointer.

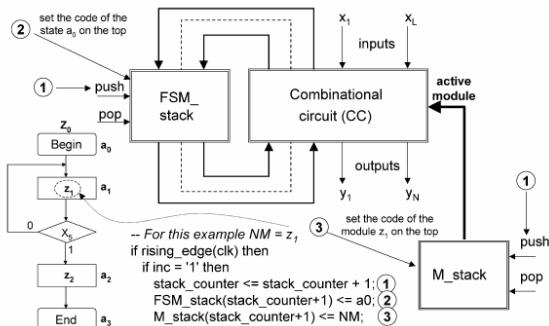


Fig. 1. Control Unit (RHFSM) demonstrating a new module z1 invocation

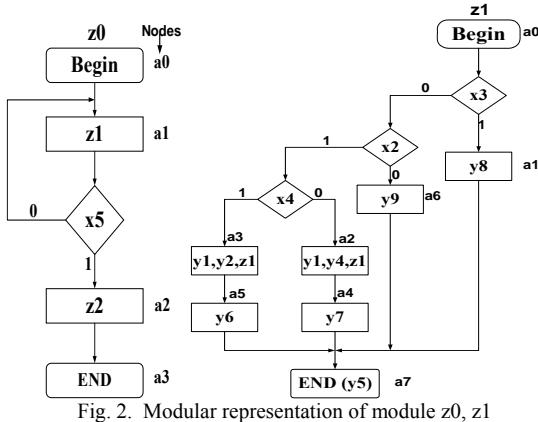


Fig. 2. Modular representation of module z0, z1

Complete problem of sorting is decomposed into three modules z0, z1, z2. In Fig. 2 module z0 is the main module and z1 is the module to construct binary search tree [8]. Module z0 is invoking the module z1 in the state a1.

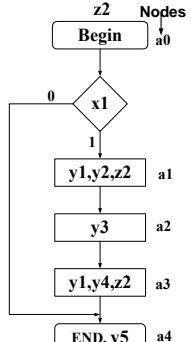


Fig. 3. Modular representation of module z2

Fig. 3 shows module z2. z2 accepts the constructed binary search tree and gives the sorted data. z2 is invoked by the a2 state of module z0 (Fig. 2). Module z1 and z2 are recursive because these contain self reference. Every module has its states (ai), its input (xi) and output (yi). Each module begins with state a0, which is the Begin state, and ends with the End state, which is labeled according to the module's number of states (in module z0 it's a3).

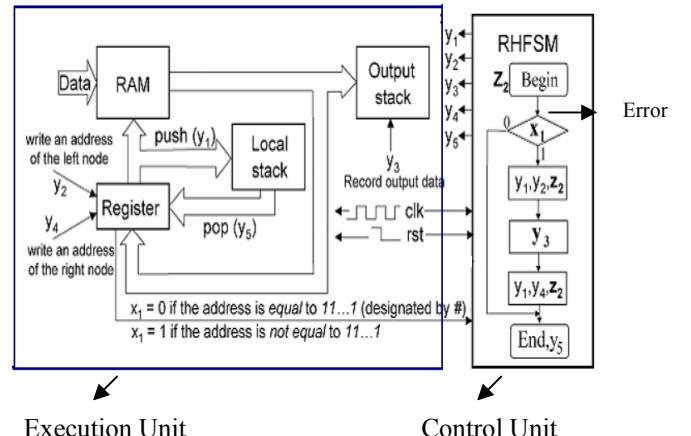


Fig. 4. A composition on Control Unit and Execution Unit

Fig. 4 gives the complete circuitry of implementation of sorting algorithm on hardware. In Fig. 4 RHFSM works as the Control Unit and the rest circuitry works as Execution Unit [9]. The function of the Fig. 4 is as follows; at some state the module produces the outputs ( $y_1, y_2, \dots$ ) that are inputs to the Execution Unit. At every decision point, the module's inputs ( $x_i$ ) are the outputs of the Execution Unit. Every time a new module is called from another module (i.e. module z2 from module z0) then the common stack pointer is incremented by one, the new module is saved at the M\_stack, the new Begin state is also saved at the FSM\_stack and the new module starts execution [10]. Every time a module state is changed, the new state is stored at the FSM\_stack, overwriting the previous state stored at the same location. Using this concept, the recursive algorithm can be easily described in hardware, as it is easy for the circuit to determine new states, modules and recursive calls using the stacks.

### III. RESULTS AND DISCUSSION

In this paper sorting algorithm is considered for the hardware implementation of recursive algorithms. Two different data sizes namely 12 and 6 are considered and the results are analyzed. Simulation results for the above two cases which verifies the functionality of complete design are given in Fig. 5 and Fig. 6 respectively. For this purpose Modelsim 6.0d is used as simulation tool. Design is also verified by the test bench. After simulation for synthesis and evaluation of design ISE 8.1.i project navigator synthesis tool is used.

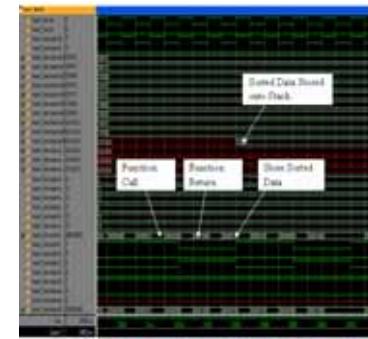


Fig. 5. Simulation results with 12 data sets

Fig. 5 is the simulation waveform obtained when the 12 dataset are given for data sorting. The waveforms show sorted output data and discrete steps for translation from recursive to non-recursive.

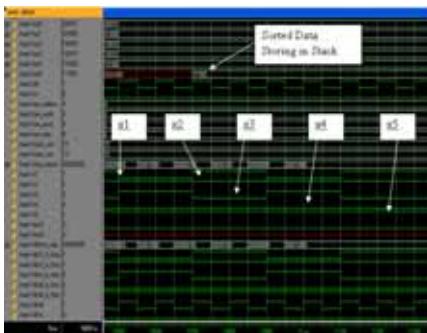


Fig. 6. Simulation results with 6 data sets

Fig. 6 shows the simulation waveform with 6 data sets. Waveforms showing change in outputs  $x_1, x_2, \dots, x_5$  of Execution Unit and storing of sorted data by Execution Unit.

After performing simulation on the data sets of length of 6 several times, the average simulation time is 10300 ns, similarly with the data set of length 12 the average simulation time is 23040 ns. For both the cases clock cycle of 100 ns is used. Hence here it can be observed that simulation time is approx doubled if the length of data set is doubled. Test bench of the main entity also gives the same result. So, the design is also verified by the test bench. Whenever the data is changed (for same number of data) the simulation time is also varied. Because the number of steps taken to get the sorted data varies with the change in the nature of data and number of steps affects the total number of clock cycles taken.

After the simulation the design is synthesized for Spartan-2 XC2S00-5PQ208 of Xilinx family using ISE8.1i. Synthesis process generates a flat netlist of HDL with the synthesis report and device utilization summary. Fig. 7 shows the RTL view of design entity.

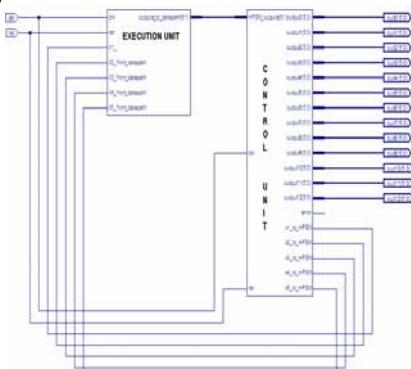


Fig. 7. Top level schematic diagram of Sorting design

In Fig. 7 clk, rst are the inputs port and out0, out1, ..., out11 are the output ports to get the sorted output data. Error signal sets to show stack overflow. All other signals are the intermediate signals. Both the units work on the same clock and resets simultaneously.

The synthesis report for the design is generated in ISE8.1i. The synthesis report gives the details of hardware resources used and also timing analysis. Here synthesis report is partially presented. The design can operate with minimum period of 17.841ns (Maximum Frequency: 56.051MHz).

Fig. 8 shows the device utilization report of data sorting design with 12 data. This report gives the information about total available hardware resources of the target device and used resources of the target device by the design. Total memory usage in this case is 110788 kilobytes.

Device Utilization Summary			
Logic Utilization	Used	Available	Utilization
Total Number Slice Registers	442	4,704	9%
Number used as Flip Flops	425		
Number used as Latches	17		
Number of 4 input LUTs	763	4,704	16%
Logic Distribution			
Number of occupied Slices	525	2,352	22%
Number of Slices containing only related logic	525	525	100%
Number of Slices containing unrelated logic	0	525	0%
Total Number 4 input LUTs	826	4,704	17%
Number used as logic	763		
Number used as a route-thru	63		
Number of bonded IOBs	79	140	56%
IOB Flip Flops	78		
Number of GCLKs	1	4	25%
Number of GCLKIOBs	1	4	25%
Total equivalent gate count for design	9,785		
Additional JTAG gate count for IOBs	3,840		

Fig. 8. Device utilization with 12 data sets

Fig. 8 shows that the numbers of slices occupied are 22% of total available slices and total gate count for the design is 9,785. Fig. 9 shows the device utilization for complete data sorting design with data size equal to 6. Total memory usage in this case is 108740 kilobytes.

Device Utilization Summary			
Logic Utilization	Used	Available	Utilization
Total Number Slice Registers	345	4,704	7%
Number used as Flip Flops	328		
Number used as Latches	17		
Number of 4 input LUTs	557	4,704	12%
Logic Distribution			
Number of occupied Slices	424	2,352	18%
Number of Slices containing only related logic	424	424	100%
Number of Slices containing unrelated logic	0	424	0%
Total Number 4 input LUTs	660	4,704	14%
Number used as logic	557		
Number used as a route-thru	63		
Number of bonded IOBs	37	140	26%
IOB Flip Flops	36		
Number of GCLKs	1	4	25%
Number of GCLKIOBs	1	4	25%
Total equivalent gate count for design	7,332		
Additional JTAG gate count for IOBs	1,824		

Fig. 9. Device utilization with 6 data sets

Fig. 9 shows that the total numbers of slices occupied by the design are 18% of total available number of slices and total gate count for the design is 7,332. This shows that when the data is increased by 100% then there is only 33.45% increment in total gate count. Similarly occupied slices are incremented by 22.22%.

In both the cases hardware resources used are less than the available hardware resources on the Spartan-2 so, synthesized designs are configured on Spartan-2 FPGA.

The total hardware resources should neither very much less nor more. It should be optimum. Because if we are configuring only a single design onto the FPGA and the design is coded into VHDL such as it consumes very less hardware resources then the rest of the available hardware resources of target device are wasted. But in case if multiple design entries are configured on the same FPGA and each design it taking more hardware resources then it will not be possible to

configure all the designs on FPGA. Further also there is a memory and time trade off. So all these constraint must be taken into consideration while designing any entity.

#### IV. CONCLUSIONS

Recursive algorithms permit complex solutions to be specified in relatively simple manner. Although recursion takes more time to get executed but it can be reduced by implementing recursion on hardware. FPGAs do not use operating system; minimize reliability concerns with true parallel executions and deterministic hardware dedicated to every task.

This paper covers a way to implement recursive algorithm on hardware with an example of sorting of numeric data. The complete unit is been designed in VHDL and verified by the testbench in Modelsim 6.0d and is implemented in an FPGA of the Spartan-2E family with ISE 8.0i project navigator. It is also verified that when the data is increased by 100% then there is only 33.45% increment in total gate count. Similarly occupied slices are incremented by 22.22%. Because there is time-memory tradeoff so, the model should be designed such as the hardware resources used by the design should be optimum.

All the modules developed for the discrete steps for the translation of recursive procedure to non-recursive procedures are reusable for different sizes of the same problem. Further the design can be made dynamic and accurate timing analysis can be performed by logic analyzer. Further work can be done in many more application involving recursive algorithms such as data compression and optimization problems.

#### V. REFERENCES

- [1]. Seymour Lipschutz “Theory and problems of data structures” Tata McGraw-Hill Edition 2002.
- [2]. Valery Sklyarov, Ioulia Skliarova, Bruno Pimentel, “FPGA- Based implementation and comparison of recursive and iterative algorithms” Proceedings of the 15th International Conference on Field-Programmable Logic and Applications - FPL2005, Finland, August 2005, Pages: 235-240.
- [3]. Valery Sklyarov, “FPGA-based implementation of recursive algorithm”, Microprocessors and Microsystems Vol.28, 2004, Pages: 197-211.
- [4]. Spyridon Ninos, Apostolos Dollas, “Modelling recursion data structures for FPGA-based implementation”, International Conference on Field-Programmable Logic and Applications, FPL’2008, Sept 2008.
- [5]. Charles H Roth, Jr., “Digital system design using VHDL”, Third reprint, PWS publishing company, 2001
- [6]. <http://www.xilinx.com> (Last viewed on 27/6/09)
- [7]. Valery Sklyarov, ”Hardware implementation of hierarchical FSMs”, ACM International Conference Proceeding Series; Vol. 92, Proceedings of the 4th international symposium on Information and communication technologies ,Cape Town South Africa, 2005, Pages: 148 – 153.
- [8]. Sklyarov, “Hierarchical finite-state machines and their use for digital control”, IEEE Transactions on VLSI Systems, Volume 7, Issue 2, June 1999, Pages: 222 – 228.
- [9]. V. Sklyarov, I. Skliarova, "Reconfigurable Hierarchical Finite State Machines", Proceedings of the 3rd International Conference on Autonomous Robots and Agents - ICARA'2006, Palmerston North, New Zealand, December 2006, Pages:.. 599-604.
- [10]. Valery Sklyarov, Ioulia Skliarova, Antonio B. Ferrari, “Hierarchical Specification and Implementation of Combinatorial Algorithms based on RHS Model” (Last viewed on 27/6/09) [online] Available: <http://www.ieeta.pt/~ioulia/Papers/2001/p13005.pdf>

# Automatic Protection Switching

Shriram K V M.Tech, Vivek C, Subashri V B.Tech<sup>1</sup>, Venkateshwaran. K<sup>2</sup>

VIT University, Vellore, TamilNadu, INDIA, shriramkv@rocketmail.com

<sup>1</sup>ASTRA University, Tanjore, TamilNadu, INDIA, <sup>2</sup>Kings College of engineering, Tanjore TamilNadu, INDIA,

**Abstract-** The aim of the paper is to discuss in detail about the FAULT MANAGEMENT and Ways of handling it. The network should survive even in the case of failures. One of the ways to get the network survivability ensured is to use APS (Automatic Protection Switching). The basic idea in APS is to keep a channel reserved (it can be dedicated or shared) having the same capacity. This paper will deal about the types of APS methods available for SONET/SDH.

## I. INTRODUCTION

In Network management there are three things that are mandatory. It is to Detect, Isolate and Correct the problems and issues in the telecommunication network. Other than this the network should be capable of acting on detection of error detection notifications. It should also carry out the diagnostic testing measures. The very important aspect in network is to maintain the error logs. The paper organized as follows: Chapter.2 explains about what is APS which is followed by types of APS in Chapter 3 and 4. Chapter 5 talks about BLSR followed by UPSR at chapter 6. Finally the paper is concluded with having references at the end.

## II. APS – DEFINITION AND DESCRIPTION

APS makes the network restoration possible in the event of network faults. The APS system will have working links and backup links in between the SONET NEs (Network Elements). If there are any problems detected the service will be automatically switched to backup link which will make the network to work fine. We can characterize different APS techniques based on the properties as

- a. Based on topology the network is built. i.e., it can be Linear or Ring.
- b. If the protection channel is shared with all working channels that require protection in case of the failure.
- c. How the network works after the problem is fixed (i.e., repaired). That is will it be capable to switch back to normal working channel or still even after restoration it will remain using the backup channel. This has to be looked into.
- d. The main focus is if the switching is unidirectional or bi directional.

The following figure depicts how APS works. Here the problem is detected in the working channel and now it has been changed to backup link.

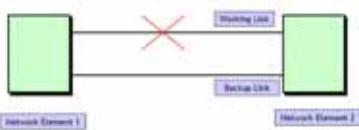


Figure.1 An Example of the APS System.

## III. TYPES OF APS

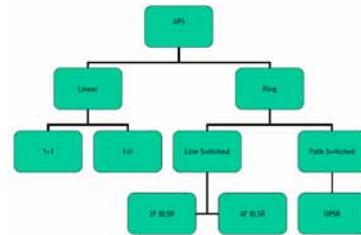


Figure.2 Types of APS

The above Figure shows the classification and types of APS available. They are discussed in detail in the following sections.

- a. **Linear APS:** point to point connections can be protected with this method. It is a classical and much traditional way of APS. This will provide protection at the line layer. This method terminates the entire SONET payload at each end of fiber span. The linear APS can be further classified into 1+1 and 1:N. These will be discussed in the next few chapters.
- b. **Ring APS:** This method is used for protecting rings (a group of nodes, which are interconnected to form a closed loop. Fiber cables will serve as links). Again Ring APS is further classified into Path switched rings and Line switched rings.
- i. **Line Switched Rings:** This method is providing protection at line layer. This enables protection of all of the STS payloads carried in OC-N Signal. (That is if a switch occurs all the SPEs are switched at the same time.)
- ii. **Path Switched Rings:** This method provides protection at the path layer. If a switch is occurring, only the affected STS pay load will be switched

Traffic will survive a single failure, either fiber or node on both the types of rings. When node failure has occurred Traffic sourced from or destined to will be lost, but all other ring traffic will survive.

## III. LINEAR APS 1+1

This is an Architecture in which the head-end signal is continuously bridged to working and protection equipment so that the same payloads are transmitted identically to the tail-end working and protection equipment. At the tail end, the working and protection OC-N signals are independently monitored. The receiving equipment chooses either the working or the protection signal as the one from which to select the traffic. Because of the continuous head-end bridge, the 1+1 architecture does not allow an unprotected extra traffic channel to be provided.

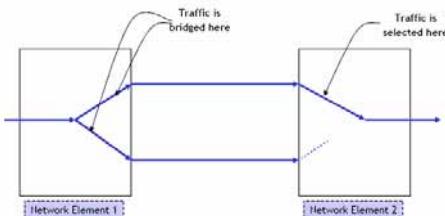


Figure. 3 Linear APS 1+1

A 1+1 system also uses, as a default, nonrevertive switching.

In nonrevertive switching, a switch to the protection line is maintained even after the working line has recovered from the failure that caused the switch or the manual switch command is cleared. In revertive switching, the traffic is switched back on the working line when the working line has recovered from the failure or the manual command is cleared.

By default system using the 1+1 architecture operates, in a unidirectional mode. In this mode, the switching is complete when a channel in the failed direction is switched to the protection line. In a bi-directional mode, a channel is switched to the protection line in both directions

#### IV. LINEAR APS 1:N

This is an architecture in which any of the n working channels can be bridged to a single protection line. Traffic bridged to both working & protect only when an APS event occurs. Till then, the protect line is free. Permissible values of n are from 1 to 14. Because the head end is switchable, the protection line can be used to carry an extra traffic channel when no APS is active. In a 1:n system, all switching is revertive. 1:1 is a special case of 1:N where N = 1

#### V. BLSR – AN INTRODUCTION

BLSR for SONET and MS-SPRING for SDH are (approximately) equivalent architectures. BLSR is abbreviation for Bi-directional Line Switched Ring and MS-SPRING is abbreviation for Multiplexer Section-Shared Protection Rings. Half of the available ring bandwidth is used for working traffic. The remaining half is dedicated for shared protection use. Both directions switch simultaneously. BLSR allows greater flexibility than UPSR. Different signals can use the same ring channel around different parts of the ring. There are many standards that support these concepts:

1. SONET (BLSR) - GR-1230
2. SDH (MS-SPRING) – G.841

There are two variants available in the form of 2fiber and 4fiber.

##### a. 2FR – BLSR:

Here in this method each node is connected to its neighbour by 2 fibers (one Tx, one Rx). There are no dedicated protection fibers. Both the fibers are used to carry working and protection traffic, in opposite directions. Half of the ring bandwidth (Channels 1...N/2 in an OC-N ring) is used for

working traffic. The remaining half (channels N/2+1...N) is dedicated for shared protection use. A fiber cut is always protected by a ring switch. The following figure is an depiction for 2FR-BLSR.

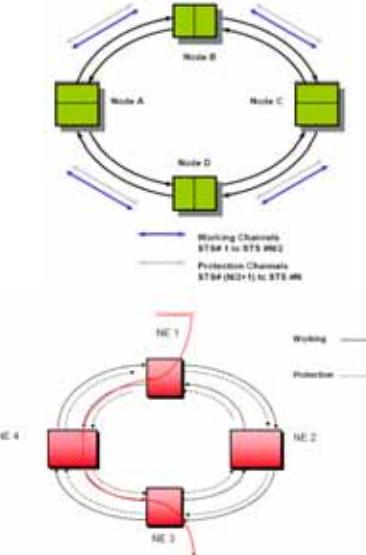


Figure. 4 2F BLSR Scheme

Following configuration diagram is showing the case where Fiber break is faced and how the remedial action is taken.

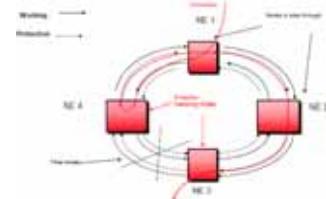


Figure. 5 Fiber breakage scenarios

##### b. 4FR – BLSR:

This is a much better schema where two dedicated fibers for Working and two dedicated fibers for Protection are being deployed. Thus, each node is connected to its neighbour by 4 Fibers in this case. There are two types of switching mechanisms available here.

1. Span Switch: This will provide the protection for a fiber cut on the working fiber.
2. Ring Switch: This will provide protection for both the protection and working fibers in the same span.

This scenario is diagrammatically shown below:

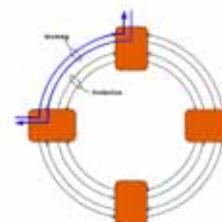


Figure. 6 4F BLSR Scheme

4FR – BLSR utilizes full bandwidth of the working fiber.

UPSR Schematic is shown as follows:

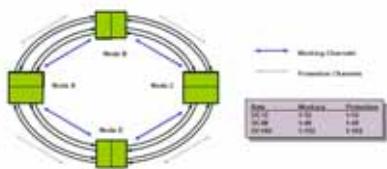


Figure. 7 4F BLSR Scheme

A typical 4FR – BLSR connection is shown as follows

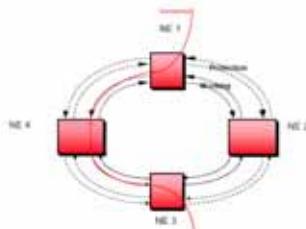


Figure. 8 4F BLSR – Typical connection

Span switch will occur when there is a working fiber breakage on a section. The same scenario can be diagrammatically depicted as follows:

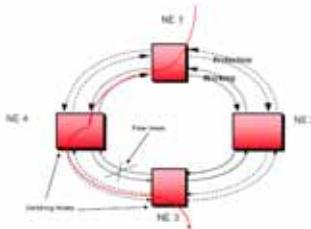


Figure. 9 4F BLSR – Span Switch

Ring switch will occur when both the working fiber and protection fiber breaks on a section. This scenario is represented pictorially as follows:

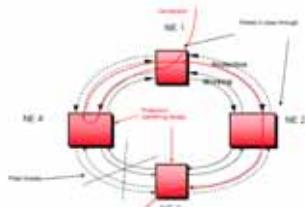


Figure. 10 4F BLSR – Ring Switch

## VI. UPSR – AN INTRODUCTION

Unidirectional Path switching rings is abbreviated as UPSR. SDH equivalent for UPSR is SNCP (Sub Network Connection Protection). The Traffic is transmitted simultaneously in both the directions Clockwise and Anti-clockwise. The destination selects one of the two paths based on the quality of the received signal. The main advantages of UPSR are

1. Simple and low cost
2. Switching is done on a per-path basis
3. Protection Switch happens only at the tail end node

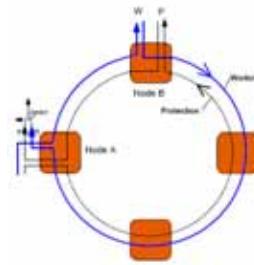


Figure. 11 UPSR Schematic

Disadvantage: Spatial reuse of fiber capacity. A single connection uses up bandwidth in the whole ring. This is used primarily in lower-speed point-to-multipoint configuration.

The following are the standards for UPSR

1. SONET (UPSR) - GR-1400
2. SDH (SNCP) – G.841

## VII. CONCLUSION

Thus the APS schemes are providing more secured way of maintaining the connectivity even at crisis times. APS makes the network restoration possible in the event of network faults. The APS system thus by using working links and backup links in between the SONET NEs (Network Elements), at the time of problems, switching to backup link happens which will make the network to work fine

## VIII. REFERENCES

- [1]. <http://www.extremenetworks.com>
- [2]. <http://www.tek.com>
- [3]. <http://www.bitpipe.com>

# Area and Speed Optimization of FIR Filter

Tarandip Singh, Indu Saini

Department of Electronics and Communication Engineering, singhtarandip@yahoo.co.in  
Dr. B. R. Ambedkar National Institute of Technology, Jalandhar, India.

**Abstract-** In this paper a low complexity and high speed digital Finite Impulse Response FIR filter has been implemented using primitive operators on FPGA. The filter has been implemented with different optimization techniques so as to reduce the complexity and increase the speed. The multipliers are realized using canonical sign digit (CSD) coefficients with the help of primitive operators such as adder, shifter and subtracters. There is reduction in the area and also increase in the speed when modified reduced adder graph algorithm is used to implement the multipliers

**Keywords-** Primitive operator FIR filter, Canonical Signed Digit (CSD), Multiplierless filter.

## I. INTRODUCTION

Digital filtering is the most important function in the digital signal processing. There are various applications in the digital signal processing such as telecommunication, digital products, and application in the information security which require high sampling speed and there is also demand for the reduction in the area for the devices like mobile phones, digital PDA's, digital cameras etc.. The digital filter is the key component in all these applications. Finite Impulse Response (FIR) filters are the most important part of the digital signal processing systems due to their linear phase response, stability, less severe to the quantization errors and roundoff noise. Digital finite impulse response filtering introduces one of many computationally demanding signal processing tasks. The reduction in the complexity of the primitive operator FIR filter [1] can be achieved by various techniques. The most common one is the use of the canonical signed digit CSD. The other technique is the reduced adder graph algorithm. This paper presents the use of these two techniques for the reduction in the complexity of the FIR filters and also to increase the speed of the filter.

## II. DIGITAL FIR FILTER

In Digital FIR filter the present output is result of past inputs and present inputs as given in the mathematical form in (1), where  $X(n)$  is the input to the system and  $Y(n)$  is the output of the system. The  $H(i)$  are the coefficients of the filter.

$$Y(n) = \sum_{i=0}^{i=n} H(i)X(n-i) \quad (1)$$

The Fig. 1 shows the 4-tap direct form FIR filter. In the figure D represents the delay; the triangle represents the multiplier and + represents the adder.

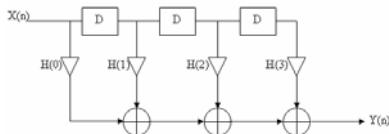


Figure 1. Figure 1 Direct form FIR filter

In the direct form there are various limitations like speed of the filter, to increase the speed extra register are needed which increases the hardware. So to overcome this limitation the transposed form of the FIR filter is preferred as shown in the Fig. 2.

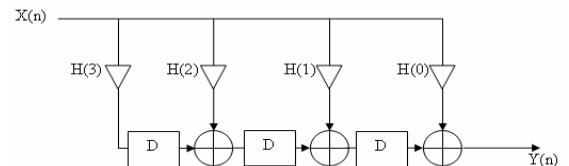


Figure 2. Transposed form FIR filter

In this structure the critical path gets reduced from  $T_M + 3T_A$  to  $T_M + T_A$ , where  $T_M$  represents time for multiplication and  $T_A$  represents the time for addition. This shows that the speed of the filter increases with this structure [2].

## III. COMPLEXITY REDUCTION USING DIFFERENT REPRESENTATION FOR COEFFICIENTS

### Normal binary representation

In this technique the number is represented using two bits either 0 or 1. In this each bit position carries some weight proportional to its position. Least Significant Bit has the weight associated with it as  $2^0$  and the weight is in increasing order towards the most significant bit, like  $2^1, 2^2, 2^3$  and so on. So in this the 7 will be represented like as  $7 = 2^2 + 2^1 + 2^0$ .

### CSD representation

In this type of representation the coefficients have the canonical property which means that no two non-zero digits are adjacent. The CSD uses ternary representation like (1, 0, -1) are used to represent a coefficient [3]. This technique helps in the reduction in the no. of non-zero terms in the coefficients, which helps in the reduction in the no of adder used to implement that particular coefficient.

For example the number 93 in normal binary format is represented as 1011101. In this case we are having five non-zero terms. When we have to implement it with the primitive operators in the FIR filter it will require four adders. Whereas in case of the CSD representation it will be represented as '10100101' here 1 represents -1, and this coefficient can be easily implemented with the help of two subtracters and an adder. This example shows that how the CSD representation helps in the reduction in the hardware of the primitive operator FIR filter.

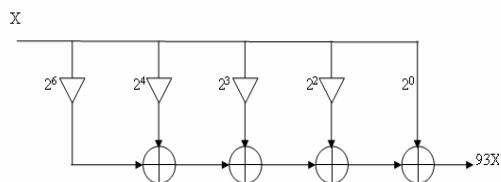


Figure 3. Representation of 93 using normal binary system

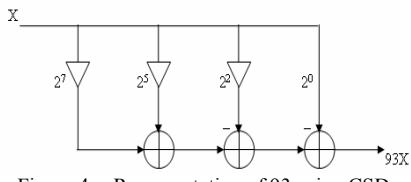


Figure 4. Representation of 93 using CSD

#### IV. COMPLEXITY REDUCTION USING MODIFIED REDUCED ADDER GRAPH ALGORITHM

In this algorithm [4] the main steps are given as

- Step 1: In the first step the sign of each coefficient has been removed as it can be realized using a subtracter.
  - Step2: The coefficients which are having power of two are removed as they can be implemented with hardwired shifts.
  - Step3: In this step all the coefficients which are having cost 1 are realized.
  - Step4: Now the cost 1 coefficients are used to implement higher cost coefficients.
- As an example if the set 7, 14 has to be implemented then first 7 will be realized and then 14 can be obtained by multiplying the 7 with 2. This helps in reduction of adders as compared to CSD representation. As in case of CSD the no. of adder will be two where as in case of reduced adder graph algorithm this set can be realized using only one adder.

#### V. DESIGN EXAMPLE AND ITS IMPLEMENTATION

The set of coefficients used to implement has been taken from [5] and given as (1, 5, 10, 16, -6, 56, -56, 76, -71, 122). The arbitrary set is not chosen because the result from the arbitrary data is inappropriate and not accurate. The set of coefficients first is implemented using normal binary representation with transposed form structure of the primitive operator digital FIR filter and then it is implemented using the Canonical Signed Digit (CSD) representation, in the last it is implemented using the reduced adder graph technique.

For the comparison of all architecture each architecture has been designed and verified with the help Xilinx ISE design tools and the simulation is done using the Modelsim simulator the results are verified using the synthesis report generated through the XST synthesis tool from Xilinx and are given in the table 1. the synthesis are carried out using the Xilinx Spartan 3E XC3S500E device. The primitive operator digital FIR filter has been coded in the Very High Speed Integrated Circuit Hardware Description Language (VHDL).

#### VI. CONCLUSION

In this paper the FIR filter using primitive operator has been implemented using different optimization techniques and from the table 1 it is cleat that with the help of the reduced

adder graph technique the complexity of the filter gets reduced by a large amount and also the speed of the structure is improved than by using only CSD and normal binary representation.

TABLE I COMPARISON OF VARIOUS TYPES OF FILTER STRUCTURES

Filter Structure	No. of adders	No. of slices	Speed (MHz)
Transposed with normal binary representation	14	154	119.046
Transposed with CSD representation	10	147	155.265
Transposed with modified reduced adder graph	6	133	237.093

#### VII. REFERENCES

- D.R Bull and D.H Horrocks, "Primitive operator digital filters," *IEE Proceedings-G*, vol. 138, 1103, pp.401-412,1991.
- K K Parhi, *VLSI Digital Signal Processing systems*, Wiley-Interscience Publications, John Wiley and Sons, 1999.
- Algirdas Avizienis, "Signed-digit number representations for fast parallel arithmetic," *IRE Transactions on Electronic Computers*, Vol. EC- 10, pp. 389-400, 1961.
- F. Xu, C.-H. Chang and C.-C. Jong, "Modified reduced adder graph algorithm for multiplierless FIR filters," *IEEE ELECTRONICS LETTERS*, 17th March 2005, vol. 41, no. 6.
- D.H. Horrocks and Y.Wongsuwan, "Reduced Complexity Primitive Operator FIR filters for low power dissipation," Proc *ECCTD '99*, Stresa, Italy, pp.273-276, 1999.

# Design and Development of Image Acquisition System

Sukesha- Lecturer, Dept. of ECE, UIET (PU), Sector-25, Chandigarh, India, er\_sukesha@yahoo.com

**Abstract-** In this work, a system has been developed to acquire triangular, circular and square two dimensional shapes. A CCD camera is used to take image of test shape. This image is stored in memory after digitizing it through analog to digital converter. This stored shape is then fed to a processor via memory. Shape is recognized using software written in processor and message is displayed on the LCD whether the acquired shape is triangular, circular or square. Shapes are recognized based upon the compactness value. In this way this system can be used to acquire and recognize three shapes.

## I. INTRODUCTION

Image acquisition is a process to acquire image using web CAM or CCD camera and store it in nonvolatile memory for storage. Most of the digital cameras are having this feature. Many times acquired image of particular shape and size needs to be further processed in order to recognize it. This task can be done using special hardware. Digital camera alone can not do this task of shape recognition. So a system is designed and developed to acquire image as well as recognize two dimensional shapes.

Shape acquisition and recognition can be used in different applications like machine vision, vehicle class recognition, assembling of PCB, grading of horticulture produce and other such applications. In case of machine vision machine parts are checked for their proper shape and size. Acquisition systems that are already available are mostly interfaced to computer instead of dedicated processor. In those systems images are acquired using digital camera or web CAM and a frame grabber card is installed on CPU of PC [1]. Frame grabber card send image to PC and there image is processed. In some systems very complicated feature extraction techniques have been used, which increase hardware and software complexities[2]. In some systems image is first enhanced with image enhancement techniques. Principal objective of enhancement techniques is to process an image so that the result is more suitable than the original image for a specific application[3]. In this system instead of sending image to PC image is acquired and then processed in embedded processor and result of shape recognition is displayed on LCD display. In other words an image acquisition card is developed which also recognizes the shape of the acquired image.

## II. SHAPE ACQUISITION SYSTEM

### 1.1 Hardware test set-up

Block diagram of Shape Acquisition system is shown in figure1

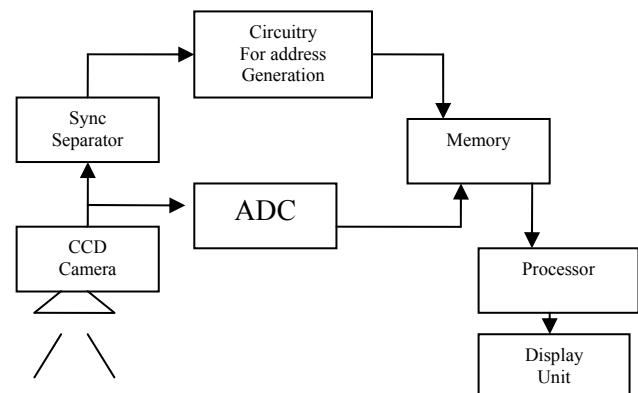


Fig 1: Block diagram of system

Test set-up consists of:

- (1) CCD Camera
- (2) Sync Separator
- (3) 9-bit counters
- (4) SRAM (static RAM)
- (5) ADC
- (6) Processor

CCD camera captures the image and sends it to Sync separator and ADC as composite video signal. Sync separator sends different sync signals to address generation circuitry viz 9-bit counters, which generates addresses for SRAM. ADC digitizes the data and stores it in given address in SRAM. Processor further retrieves this data for processing. Description and working of each component is as follows:

#### 2.1.1 CCD Camera:

Charged coupled device (CCD) camera is used to scan black and white two dimensional shapes. In this system CCD sensor with 512 X 312 lines is used. Where 312 is no. of scan lines and 512 is number of dots in a line. Scanned image in the form of composite video signal is sent to ADC and to the circuitry for addressing memory, which includes sync separator.

#### 2.1.2 Sync separator:

Output of CCD camera is also fed to sync separator. Composite video signal from CCD camera consists of three type of information: picture details, horizontal sync signal and vertical sync signal[4]. Both vertical and horizontal sync signal is used for generating address. The LM1881 video sync separator extracts timing information with amplitude from 0.5V to 2V peak to peak. Sync separator, separates the composite video signal into three parts viz composite sync signal, horizontal sync signal and vertical sync signal. Both vertical and horizontal sync signal are used for generating addresses. In Interlaced format all the odd lines of each frame are transmitted first, followed by the even lines. The group of odd lines is called the odd field, and the group of even lines is called the even field [5]. Since each frame consists of two fields, the video signal transmits 50 fields per second. Each

field starts with a complex series of vertical sync pulses within each line, the analog voltage corresponds to the grayscale of the image.

#### 2.1.3 9-bit counters:

As picture of 512\*312 pixels is used, therefore 9-bit counters are also used for generating addresses.  $2^9 = 512$  and  $2^9 \approx 312$

The sync details are fed to 9-bit counters and these two 9-bit counters generate 18 bit signal as address lines. 74HC4040B is ripple binary counter [6]. One counter is pixel counter another one is line counter as shown in fig 2. State of counter advances one count on negative transition of each input pulse. A high level on reset line resets counter to its all zero state. Pixel counter resets at the end of horizontal line and generates address of each pixel or dot. Clock to pixel counter is 10 MHz. Reset signal to pixel counter is composite sync. Because pixel counter should reset at the end of each horizontal line as well as at the start of vertical sync signal. Line counter counts at the rate (clock speed) of composite sync and resets at start of vertical sync. It generates address of each new line.

#### 2.1.3 Analog to digital converter

Speed of analog to digital converter here is 10 MHz [7][8]. Number of dots in a scanned line are 512, and picture signal duration in composite video signal is 52  $\mu$ s. So 52  $\mu$ s/512 is nearly equal to 0.1  $\mu$ s, or 1/0.1  $\mu$ s = 10 MHZ. Dot clock is also 10MHZ. ADC digitize the whole data but only picture detail is stored in SRAM. Write operation of SRAM is controlled by processor

#### 2.1.4 SRAM

According to number of address lines and data, memory being used is of size 256KB. Memory is used to store picture details. This data is used for processing. Where 256 mean there are 18 address lines and byte shows that data lines are 8. The CMOS BQ4014 is nonvolatile 2097152 bit static RAM organized as 262144 words by 8 bits.

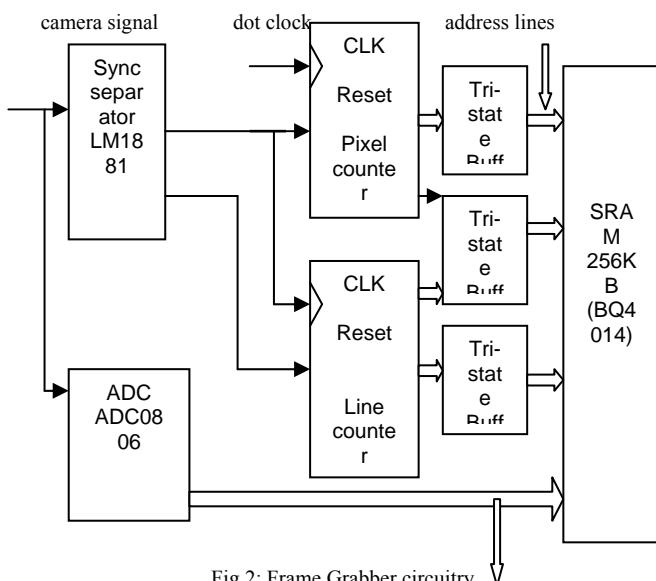


Fig 2: Frame Grabber circuitry

#### 2.1.5 Processor

AVR RISC processor AT90S8515 is used here [9][10]. Processing involves comparing the compactness values of different acquired shapes. Compactness here is defined as ratio of perimeter<sup>2</sup> and area and its value is always different for different shapes [11]. Area and perimeter of shapes are calculated by tracing the digitized data stored in SRAM. Perimeter is calculated by tracing the boundary of 1s using edge finding algorithm[12]. Already stored database is compared with acquired data. According to this comparison matching result is displayed. Shapes that are used for recognition are: square, triangular, and circular. Processor checks whether image is of any of these types. If yes it gives results according to that. Other than this enabling Tri-state buffers, setting direction of Tri-state buffers, ADC, SRAM, writing into SRAM are controlled by processor. As shown in fig 3 Address generated by counters is sent to SRAM only when Tri-state buffers are enabled by processor. An active low signal is sent from processor to enable the buffers for 20 m sec (time for one field). During this time buffer that is connected to ADC is also enabled. But write enable signal to SRAM is only sent when picture signal is present. A circuit is designed to control write operation for SRAM. When digitized data of one field is stored in SRAM, then these Tri-state buffers are disabled and buffers that are connected to processor are enabled. Data is retrieved from processor for further processing.

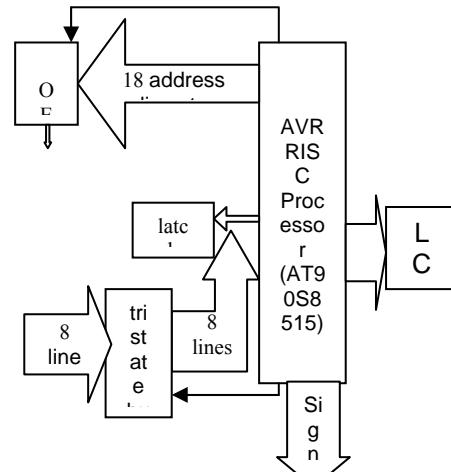


Fig 3: Interfacing Processor to Other Components

#### 2.1.6 LCD Display

LCD display is used in this system. A message is displayed on LCD system that whether acquired image belong to the particular class. Result of processing is displayed on LCD display[13]. Standard LCD requires three control lines as well as either 4 or 8 input output lines for data bus.

## 2.2 PCB Design

Hardware for this system is designed using powerful EDA (Electronic design automation) tool ORCAD. The circuit diagram entry is carried out in software capture 9.1 and software PCB layout plus 9.1. The schematics and PCB layout prints are shown below.

### 2.2.1 Crystal oscillator

Crystal oscillator of 10MHZ is designed, because ADC requires 10MHZ and clock input to line counter and pixel counter is also 10MHZ. crystal of 10 MHZ is used here. CMOS inverter is used in designing the circuit, because of its fast switching and low delay. Schematic designed in ORCAD is shown in fig 4.

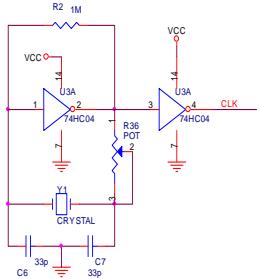


Fig 4: Schematic of master clock for ADC.

### 2.2.2 Schematic of ADC

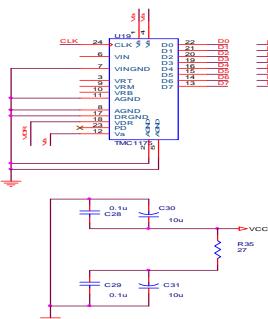


Fig 5: Schematic for ADC

Circuit design entry for ADC (ADC08L060) carried out in capture9.1 is shown in fig 5. Input signal that is fed to ADC should be in range from +3V to 0V. In this circuit it is 1.3V. Input amplifier LMH6702 connected at input of ADC, where gain is adjusted to 2.

### 2.2.3 Schematic for Sync Separator

Circuit design for sync separator LM1881 and counters carried out in capture9.1 is shown in fig 6 and 7. Supply requirement of sync separator is +5V. Input to sync separator is camera signal of 1V. Output of sync separator is composite sync signal, vertical sync signal and burst/back porch signal. Resistor on pin 6 allows LM1881 to be adjusted for source signals with line scan frequencies differing from 15.734 kHz.

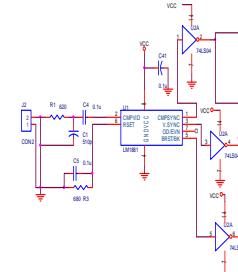


Fig 6: Schematic For Sync separator (Left half, Right half is in fig 7)

### 2.2.4 9-bit counters

9-bit counters are connected to sync separator through inverters as shown in fig 6 and 7. these 9 bit counters are used to generate 18 bit address for SRAM memory. Actually 74LS4040 is 12 bit counter. Only 9 bits are used, remaining 3 are left open.

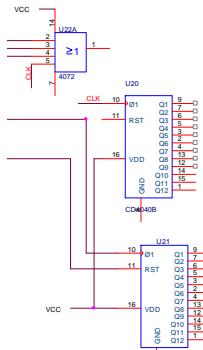


Fig 7: 9-bit counters (remaining right half of fig 6)

### 2.2.5 Schematic for Processor

Circuit design for AVR RISC processor carried out in capture9.1 is shown in fig 8. In this system AVR processor operates on +5V. Clock input to AVR is 8MHz. Filtering circuit at VCC pin is connected to remove any spike in supply. Reset circuit is connected at pin no. 9 and ALE pulses appear at pin no. 30. This processor is tested by observing pulses on ALE pin with high frequency probe. Here AVR is connected to tristate buffer at port A, port C and 1<sup>st</sup> and 2<sup>nd</sup> pin of port B. Some pins of port B (PB4, PB5, PB6, PB7) and some pins of port D (PD3, PD4, PD5) are connected to LCD display.

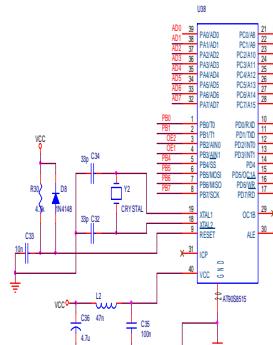


Fig 8: Schematic of AVR RISC Processor

### III. RESULTS

The developed system is used to acquire and recognize three shapes triangular, circular and square. If shape other than three shape is acquired it is declared as other shape.

Only one Field out of two frames is acquired for 20 m sec, which is the duration of each one field. So processor enables the Tri-state buffers for only 20mSec and only one field is stored in SRAM in digitized or binary form.

Data stored in SRAM is actually shape data in binary form. Image stored is actually black and white image. Black shade is stored as 1 and white shade is stored as 0.

Area of shape is calculated by calculating total number of 1s and perimeter by scanning boundary of 1s by ‘Edge finding algorithm’. Compactness is calculated by perimeter square over area. Shape is recognized as follows:

- (a) If compactness lies between 15-18 then it is square.
- (b) If compactness lies between 18-25 then it is a Triangle.
- (c) If compactness lies between 12-14 then it is circle.
- (d) If compactness lies between 25-48 less than 12 and greater than 25 then system declare it as other shape.

Different shapes are acquired with image acquisition card and then recognized and results are displayed on LCD.

### IV. CONCLUSIONS

It is possible to Acquire and recognize two-dimensional shapes using the presented system. System basically acquires and digitizes the shape and calculates compactness of test shape.

### V. REFERENCES.

- [1]. W. A. Steer, Video Digitizer, 1999 [www.techmind.org](http://www.techmind.org)
- [2]. Review of image and video Indexing Techniques by F Idris and S. Panchanathan, Journal of Visual communication and Image Representation, Vol 8, no.2, pp 144-166, 1997.
- [3]. Vision Systems by Gary A. Mintchell, Senior Editor Control Engg, 1998.
- [4]. GULATI R.R., Monochrome and Color TV, Wiley Eastern Limited, third print 1986.
- [5]. Sync separator datasheet from [www.ni.com](http://www.ni.com)
- [6]. Counter description from [www.texasinstruments.com](http://www.texasinstruments.com)
- [7]. ADC description [www.analogdevices.com](http://www.analogdevices.com)
- [8]. Analog to digital converter description from [www.texasinstruments.com](http://www.texasinstruments.com)
- [9]. AVR microcontroller's architecture details. [www.atmel.com](http://www.atmel.com)
- [10]. AVR microcontroller's user manual. (From Atmel )
- [11]. Rafael C. Gonzalez & Richard E. Woods, “Digital Image Processing”, Pearson education ASIA, Seventh Indian Print,2001.
- [12]. Jain A.K., “fundamental of Digital Image Processing” PHI, seventh printing July 2001.
- [13]. LCD datasheet [www.sharpmicroelectronics.com](http://www.sharpmicroelectronics.com)

# Unimodal and Multimodal Biometric

Reecha Sharma- Lecturer UCoE Punjabi University Patiala  
[richa\\_gemini@yahoo.com](mailto:richa_gemini@yahoo.com)

**Abstract – Biometrics is an automatic identification or verification of an individual (or a claimed identity) by using some physical or behavioral characteristics associated with the person.** By using biometrics it is possible to establish an identity based on 'who he is', rather than by 'what he possess', like an ID card or 'what he remember', like a password. Therefore, biometric systems use fingerprints, hand geometry, iris, retina, face, vasculature patterns, signature, gait, palm print, voiceprint or DNA print to determine a person's identity. The purpose of this paper is (a) to introduce the fundamentals of unimodal biometric technology (b) limitations of unimodal biometric such as noisy data, intra-class variations, restricted degrees of freedom, non-universality, spoof attacks and unacceptable error rates (c) the introduction to multimodal biometric in which multiple sources of biometric information are consolidated. We also present several examples of multimodal systems that have been described in the literature.

**Keywords:** Biometrics, identification, unimodal biometrics, multimodal biometrics, recognition.

## I.INTRODUCTION

A reliable identity management system is urgently needed in order to combat the epidemic growth in identity theft and to meet the increased security requirements in a variety of applications ranging from international border crossings to securing information in databases. Establishing the identity of a person is a critical task in any identity management system. Surrogate representations of identity such as passwords and ID cards are not sufficient for reliable identity determination because they can be easily misplaced, shared or stolen. Biometric recognition is the science of establishing the identity of a person using his/her anatomical and behavioral traits. Commonly used biometric traits include face, fingerprint, hand geometry, iris, retina, handwritten signatures and voice (see Figure 1)[20]. Biometric traits have a number of desirable properties with respect to their use as an authentication token, viz., reliability, convenience, universality etc. These characteristics have led to the widespread deployment of biometric authentication systems. But there are still some issues concerning the security of biometric recognition systems that need to be addressed in order to ensure the integrity and public acceptance of these systems.

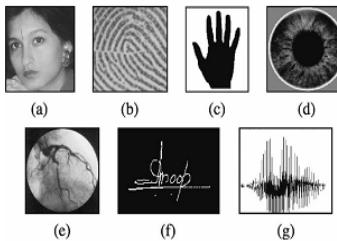


Fig.1. Examples of biometric characteristics. (a) Face (b) Fingerprint (c) Hand geometry (d) Iris (e) Retina (f) Signature (g) Voice. From D. Maltoni, D. Maio, A. K.Jain, S. Prabhakar, Handbook of Fingerprint Recognition (NewYork: Springer-

Verlag, 2003), Fig. 1.2, p. 8. Copyright by Springer-Verlag. Reprinted with permission.

There are five major components in a generic biometric authentication system, namely, sensor, feature extractor, template database, matcher and decision module (see Figure 2) [21]. Sensor is the interface between the user and the authentication system and its function is to scan the biometric trait of the user. Feature extraction module processes the scanned biometric data to extract the salient information that is useful in distinguishing between different users. In some cases, the feature extractor is preceded by a quality assessment module which determines whether the scanned biometric trait is of sufficient quality for further processing. During enrollment, the extracted feature set is stored in a database as a template ( $X_T$ ) indexed by the user's identity information. Since the template database could be geographically distributed and contain millions of records, maintaining its security is not a trivial task. The matcher module is usually an executable program which accepts two biometric feature sets  $X_T$  and  $X_Q$  (from template and query, respectively) as inputs and outputs a match score (S) indicating the similarity between the two sets. Finally, the decision module makes the identity decision and initiates a response to the query. Due to the rapid growth in sensing and computing technologies, biometric systems have become affordable and are easily embedded in a variety of consumer devices (e.g., mobile phones,), making this technology vulnerable to the malicious designs of terrorists and criminals. To avert any potential security crisis, vulnerabilities of the biometric system must be identified and addressed systematically. A number of studies have analyzed potential security breaches in a biometric system and proposed methods to counter those breaches [1]–[5]. Our goal here is to broadly categorize the various factors that cause biometric system failure and identify the effects of such failures.

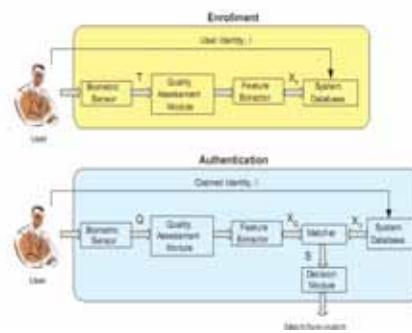


Fig.2.Enrollment and recognition stages in a biometric system. Here, T represents the biometric sample obtained during enrollment, Q is the query biometric sample obtained during recognition,  $X_T$  and  $X_Q$  are the template and query feature sets, respectively and S represents the match score

## II. BIOMETRICS

A number of biometric characteristics have been in use in various applications (see Fig. 1)[20]. Each biometric has its strengths and weaknesses, and the choice depends on the application. No single biometric is expected to effectively meet all the requirements (e.g., accuracy, practicality, cost) of all the applications (e.g., Digital rights management DRM, access control, welfare distribution). In other words, no biometric is “optimal.” The Table 1 Comparison of Various Biometric Technologies Based on the Perception of the Authors. High, Medium, and Low are Denoted by H, M, and L, Respectively match between a specific biometric and an application is determined depending upon the requirements of the application and the properties of the biometric characteristic. A brief comparison of some of the biometric identifiers based on seven factors is provided in Table 1[7]. Universality (do all people have it?), distinctiveness (can people be distinguished based on an identifier?), permanence (how permanent is the identifier?), and collectability (how well can the identifier be captured and quantified?) are properties of biometric identifiers. Performance (speed and accuracy), acceptability (willingness of people to use), and circumvention (foolproof) are attributes of biometric systems [9]. Use of many other biometric characteristics such as retina, infrared images of face and body parts, gait, odor, ear, and DNA in commercial authentication systems is also being investigated [7].

Biometric identifier	Universality	Distinctiveness	Permanence	Collectability	Performance	Acceptability	Circumvention
DNA	H	H	H	L	H	L	L
Ear	M	M	H	M	M	H	M
Face	H	L	M	H	L	H	H
Facial thermogram	H	H	L	H	M	H	L
Fingerprint	M	H	H	M	H	M	M
Gait	M	L	L	H	L	H	M
Hand geometry	M	M	M	H	M	M	M
Hand vein	M	M	M	M	M	M	L
Iris	H	H	H	M	H	L	L
Keystroke	L	L	L	M	L	M	M
Odor	H	H	H	L	L	M	L
Palmprint	M	H	H	M	H	M	M
Retina	H	H	M	L	H	L	L
Signature	L	L	L	H	L	H	H
Voice	M	L	L	M	L	H	H

Table 1 Comparison of Various Biometric Technologies Based on the Perception of the Authors. High, Medium, and Low are Denoted by H, M, and L, Respectivelyconsequently, biometrics is not only an important pattern recognition research problem but is also an enabling technology that will make our society safer, reduce fraud and lead to user convenience (user friendly man-machine interface) by broadly providing the following three functionalities: (a) Positive Identification (“Is this person who he claims to be?”). (b) Large Scale Identification (“Is this person in the database?”). (c) Screening (“Is this a wanted person?”). Here we categorize the

fundamental barriers in biometrics into four main categories: (i) accuracy, (ii) scale, (iii) security, and (iv) privacy [ 10].

## III. UNIMODAL BIOMETRICS

Establishing the identity of a person is becoming critical in our vastly interconnected society. Questions like “Is she really who she claims to be?”, “Is this person authorized to use this facility?” or “Is he in the watchlist posted by the government?” are routinely being posed in a variety of scenarios ranging from issuing a driver’s license to gaining entry into a country. The need for reliable user authentication techniques has increased in the wake of heightened concerns about security and rapid advancements in networking, communication and mobility. Biometrics, described as the science of recognizing an individual based on her physiological or behavioral traits, is beginning to gain acceptance as a legitimate method for determining an individual’s identity. Biometric systems have now been deployed in various commercial, civilian and forensic applications as a means of establishing identity. These systems rely on the evidence of fingerprints, hand geometry, iris, retina, face, hand vein, facial thermogram, signature, voice, etc. to either validate or determine an identity [6]. Most biometric systems deployed in real-world applications are unimodal, i.e., they rely on the evidence of a single source of information for authentication (e.g., single fingerprint or face). These systems have to contend with a variety of problems such as:

- (a) Noise in sensed data: A fingerprint image with a scar, or a voice sample altered by cold are examples of noisy data. Noisy data could also result from defective or improperly maintained sensors (e.g., accumulation of dirt on a fingerprint sensor) or unfavorable ambient conditions (e.g., poor illumination of a user’s face in a face recognition system).
- (b) Intra-class variations: These variations are typically caused by a user who is incorrectly interacting with the sensor (e.g., incorrect facial pose), or when the characteristics of a sensor are modified during authentication (e.g., optical versus solid-state fingerprint sensors).
- (c) Inter-class similarities: In a biometric system comprising of a large number of users, there may be inter-class similarities (overlap) in the feature space of multiple users. Golfarelli et al.[7] state that the number of distinguishable patterns in two of the most commonly used representations of hand geometry and face are only of the order of 105 and 103, respectively.
- (d) Non-universality: The biometric system may not be able to acquire meaningful biometric data from a subset of users. A fingerprints biometric system, for example, may extract incorrect minutiae features from the fingerprints of certain individuals, due to the poor quality of the ridges. [ 20]
- (e) Spoof attacks: This type of attack is especially relevant when behavioral traits such as signature or voice are used. However, physical traits such as fingerprints are also susceptible to spoof attacks. [21]

Some of the limitations imposed by unimodal biometric systems can be overcome by including multiple sources of information for establishing identity [8]. Such systems, known as multimodal biometric systems, are expected to be more

reliable due to the presence of multiple, (fairly) independent pieces of evidence [9]. These systems are able to meet the stringent performance requirements imposed by various applications. They address the problem of non-universality, since multiple traits ensure sufficient population coverage. They also deter spoofing since it would be difficult for an impostor to spoof multiple biometric traits of a genuine user simultaneously. Furthermore, they can facilitate a challenge response type of mechanism by requesting the user to present a random subset of biometric traits thereby ensuring that a ‘live’ user is indeed present at the point of data acquisition.

#### IV. APPLICATIONS OF BIOMETRIC SYSTEMS

The applications of biometrics can be divided into the following three main groups.

- Commercial applications such as computer network login, electronic data security, ecommerce, Internet access, ATM, credit card, physical access control, cellular phone, PDA, medical records management, and distance learning[19].
- Government applications such as national ID card, correctional facility, driver’s license, social security, welfare disbursement, border control, and passport control.
- Forensic applications such as corpse identification, criminal investigation, terrorist identification, parenthood determination, and missing children.

#### V. MULTIMODAL BIOMETRIC SYSTEMS

Some of the limitations imposed by unimodal biometric systems can be overcome by using multiple biometric modalities (such as face and fingerprint of a person or multiple fingers of a person). Such systems, known as multimodal biometric systems [23], are expected to be more reliable due to the presence of multiple, independent pieces of evidence [25]. These systems are also able to meet the stringent performance requirements imposed by various applications [24]. Multimodal biometric systems address the problem of nonuniversality, since multiple traits ensure sufficient population coverage. Further, multimodal biometric systems provide antispoofting measures by making it difficult for an intruder to simultaneously spoof the multiple biometric traits of a legitimate user. By asking the user to present a random subset of biometric traits (e.g., right index and right middle fingers, in that order), the system ensures that a “live” user is indeed present at the point of data acquisition. Thus, a challenge-response type of authentication can be facilitated using multimodal biometric systems.

Multimodal biometric systems can be designed to operate in one of the following five scenarios (see Fig. 3[21].

- 1) Multiple sensors: the information obtained from different sensors for the same biometric are combined. For example, optical, solid-state, and ultrasound based sensors are available to capture fingerprints.
- 2) Multiple biometrics: multiple biometric characteristics such as fingerprint and face are combined. These systems will necessarily contain more than one sensor with each sensor sensing a different biometric characteristic. In a verification

system, the multiple biometrics are typically used to improve system accuracy, while in an identification system the matching speed can also be improved with a proper combination scheme (e.g., face matching which is typically fast but not very accurate can be used for retrieving the top matches and then fingerprint matching which is slower but more accurate can be used for making the final identification decision).

- 3) Multiple units of the same biometric: fingerprints from two or more fingers of a person may be combined, or one image each of the two irises of a person may be combined.

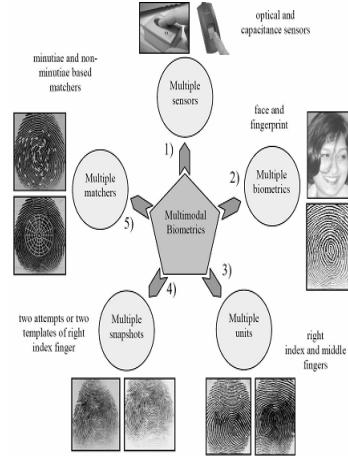


Fig. 3. Various scenarios in a multimodal biometric system.

- 4) Multiple snapshots of the same biometric: more than one instance of the same biometric is used for the enrollment and/or recognition. For example, multiple impressions of the same finger, multiple samples of the voice, or multiple images of the face may be combined.

- 5) Multiple representations and matching algorithms for the same biometric: this involves combining different approaches to feature extraction and matching of the biometric characteristic. This could be used in two cases. First, verification or an identification system can use such a combination scheme to make a recognition decision. Second, an identification system may use such a combination scheme for indexing.

#### VI. EXAMPLES OF MULTIMODAL BIOMETRIC SYSTEMS

Multimodal biometric systems have received much attention in recent literature. Brunelli et al. [14] describe a multimodal biometric system that uses the face and voice traits of an individual for identification. Their system combines the matching scores of five different matchers operating on the voice and face features to generate a single matching score that is used for identification. Bigun et al. developed a statistical framework based on Bayesian statistics to integrate information presented by the speech (text-dependent) and face data of a user [15]. Hong et al. combined face and fingerprints for person identification [26]. Their system consolidates multiple cues by associating different confidence measures with the individual biometric matchers and achieved a significant improvement in retrieval time as well as

identification accuracy. Kumar et al. combined hand geometry and palmprint biometrics in a verification system [18]. A commercial product called Bio ID [16] uses voice, lip motion, and face features of a user to verify identity. Jain and Ross improved the performance of a multimodal biometric system by learning user-specific parameters [19]

## VII. SUMMARY

Reliable personal recognition is critical to many business processes. Biometrics refers to automatic recognition of an individual based on her behavioral and/or physiological characteristics. The conventional knowledge based and token-based methods do not really provide positive personal recognition because they rely on surrogate representations of the person's identity (e.g., exclusive knowledge or possession). It is thus obvious that any system assuring reliable personal recognition must necessarily involve a biometric component. This is not, however, to state that biometrics alone can deliver reliable personal recognition component. In fact, a sound system design will often entail incorporation of many biometric and non biometric components (building blocks) to provide reliable personal recognition. Biometric-based systems also have some limitations that may have adverse implications for the security of a system. While some of the limitations of biometrics can be overcome with the is expected to result in a better improvement in performance than a combination of correlated modalities (e.g., different impressions of the same finger or different fingerprint matchers) Further, a combination of uncorrelated modalities can significantly reduce the failure to enroll rate as well as provide more security against "spoofing." On the other hand, such a combination requires the users to provide multiple identity cues which may cause inconvenience. Additionally, the cost of the system increases because of the use of multiple sensors (e.g. when combining fingerprints and face). The convenience and cost factors remain the biggest barriers in the use of such multimodal biometrics systems in civilian applications. We anticipate that high security applications, large-scale identification systems, and negative identification applications will increasingly use multimodal biometric systems, while small-scale low-cost commercial applications will probably continue striving to improve unimodal biometric systems.

## VIII. REFERENCES

- [1]. C. Roberts, "Biometric Attack Vectors and Defences," Computers and Security, vol. 26, no. 1, pp. 14–25, February 2007.
- [2]. M1.4 Ad Hoc Group on Biometric in E-Authentication, "Study Report on Biometrics in E-Authentication," International Committee for Information Technology Standards (INCITS), Technical Report INCITS M1/07-0185rev, March 2007.
- [3]. K. Jain, A. Ross, and S. Pankanti, "Biometrics: A Tool for Information Security," IEEE Transactions on Information Forensics and Security, vol. 1, no. 2, pp. 125–143, June 2006.
- [4]. Buhar and P. Hartel, "The State of the Art in Abuse of Biometrics," Centre for Telematics and Information Technology, University of Twente, Technical Report TR-CTIT-05-41, December 2005.
- [5]. K. Jain, A. Ross, and U. Uludag, "Biometric Template Security: Challenges and Solutions," in Proceedings of European Signal Processing Conference (EUSIPCO), Antalya, Turkey, September 2005.
- [6]. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," IEEE Trans. on Circuits and Systems for Video Technology, vol. 14, pp. 4–20, Jan 2004.
- [7]. Anil K. Jain, Fellow, IEEE, Arun Ross, Member, IEEE, and Salil Prabhakar, Member, IEEE "An Introduction to Biometric Recognition" IEEE Transactions On Circuits And Systems For Video Technology, Vol. 14, No. 1, January 2004
- [8]. Ross and A. K. Jain, "Information fusion in biometrics," Pattern Recognition Letters, vol. 24, pp. 2115–2125, Sep 2003.
- [9]. L. I. Kuncheva, C. J. Whitaker, C. A. Shipp, and R. P. W. Duin, "Is independence good for combining classifiers?," in Proc. of Int'l Conf. on Pattern Recognition (ICPR), vol. 2, (Barcelona, Spain), pp. 168–171,2000.
- [10]. Anil K. Jain, Sharath Pankanti, Salil Prabhakar, Lin Hong, and Arun Ross "Biometrics: A Grand Challenge "Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04) 1051-4651/04 \$ 20.00 IEEE
- [11]. L. Hong, A. K. Jain, and S. Pankanti, "Can multibiometrics improve performance?," in Proc. AutoID'99, Summit, NJ, Oct. 1999, pp. 59–64.
- [12]. L. Hong and A. K. Jain, "Integrating faces and fingerprints for personal identification," IEEE Trans. Pattern Anal. Machine Intell., vol. 20, pp. 1295–1307, Dec. 1998.
- [13]. L. I. Kuncheva, C. J. Whitaker, C. A. Shipp, and R. P.W. Duin, "Is independence good for combining classifiers?," in Proc. Int. Conf. Pattern Recognition (ICPR), vol. 2, Barcelona, Spain, 2001, pp. 168–171.
- [14]. R. Brunelli and D. Falavigna, "Person identification using multiple cues," IEEE Trans. Pattern Anal. Machine Intell., vol. 12, pp. 955–966,Oct. 1995
- [15]. E. S. Bigun, J. Bigun, B. Duc, and S. Fischer, "Expert conciliation for multimodal person authentication systems using bayesian statistics," in Proc. Int. Conf. Audio and Video-Based Biometric Person Authentication (AVBPA), Crans-Montana, Switzerland, Mar. 1997, pp. 291–300.
- [16]. R. W. Frischholz and U. Dieckmann, "BioID: A multimodal biometric identification system," IEEE Comput., vol. 33, pp. 64–68, 2000.
- [17]. A.K. Jain and A.Ross, "Learning user-specific parameters in a multibiometric system," in Proc. Int. Conf. Image Processing (ICIP), Rochester,NY, Sept. 2002, pp. 57–60.
- [18]. Kumar, D. C. Wong, H. C. Shen, and A. K. Jain, "Personal verification using palmprint and hand geometry biometric," presented at the 4th Int. Conf.Audio- andVideo-based Biometric Person Authentication,Guildford, U.K., June 9–11, 2003.
- [19]. A.K. Jain andA.Ross, "Learning user-specific parameters in amultibiometric system," in Proc. Int. Conf. Image Processing (ICIP), Rochester, NY, Sept. 2002, pp. 57–60.
- [20]. Umut Uludag, Sharath Pankanti, Salil Prabhakar and Anil k. Jain, "Biometric Cryptosystems: Issues and Challenges" PROCEEDINGS OF THE IEEE, VOL. 92, NO. 6, JUNE 2004
- [21]. Anil K. Jain, Karthik Nandakumar and Abhishek Nagar "biometric Template Security" Eurasp journal on Advances in signal processing, special issues on biometrics, January 2008
- [22]. Anil K. Jain, Fellow, IEEE, Arun Ross, Member, IEEE, and Salil Prabhakar, "An Introduction to Biometric Recognition" Ieee Transactions On Circuits And Systems For Video Technology, Vol. 14, No. 1, January 2004
- [23]. E. d.Os.H. Jongebloed,A. Stijnsiger, and L. Boves, "Speaker verification as a user-friendly access for the visually impaired," in Proc. Eur. Conf. Speech Technology, Budapest, Hungary, 1999, pp. 1263–1266.
- [24]. W. R. Harrison, Suspect Documents, Their Scientific Examination. Chicago, IL: Nelson-Hall, 1981.
- [25]. T. Matsumoto, H. Matsumoto, K. Yamada, and S. Hoshino, "Impact of artificial gummy fingers on fingerprint systems," Proc. SPIE, vol. 4677, pp. 275–289, Feb. 2002.
- [26]. L. Hong, A. K. Jain, and S. Pankanti, "Can multibiometrics improve performance ?," in Proc. AutoID'99, Summit, NJ, Oct. 1999, pp. 59–64.
- [27]. L. Hong and A. K. Jain, "Integrating faces and fingerprints for personal identification," IEEE Trans. Pattern Anal. Machine Intell., vol. 20, pp.1295–1307, Dec. 1998.
- [28]. L. I. Kuncheva, C. J.Whitaker, C. A. Shipp, and R. P.W. Duin, "Is independence good for combining classifiers?," in Proc. Int. Conf. Pattern Recognition (ICPR), vol. 2, Barcelona, Spain, 2001, pp. 168–171.

# Challenging Methods of Access Control for Grid Computing

Ramandeep Kaur, Rajdavinder Singh Boparai , Students- M.Tech , Punjabi University, Patiala

**Abstract-** In this paper, we have discussed various virtual organizations comprising of several independent autonomous domains. It also defines the relationship between resources and users that is more ad hoc. The users are need not to be in same security domain, therefore, the traditional identity-based access control models are not effective, and access decisions need to be made attributes based, role-based, agent-based and policy based, semantic based etc. i.e. users are identified by their characteristics or attributes rather than predefined identities. Also, in a Grid system, autonomous domains have their own security policies, so the access control mechanism needs to be flexible to support different kind of policies.

**Keywords:** Access Control, Grid Computing, Globus Toolkit, Authorization Framework

## I. INTRODUCTION

Grid computing can mean different things to different individuals. It can be presented as an analogy to power grids where users (or electrical appliances) get access to electricity through wall sockets with no care or consideration for where or how the electricity is actually generated. In this view of grid computing, computing becomes pervasive and individual users (or client applications) gain access to computing resources (processors, storage, data, applications, and so on) as needed with little or no knowledge of where those resources are located or what the underlying technologies, hardware, operating system, and so on are.

Grid computing is the application of several computers to a single problem at the same time - usually to a scientific or technical problem that requires a great number of computer processing cycles or access to large amounts of data.

## II.ACCESS CONTROL

Access control is the traditional center of gravity of computer security. It is where security engineering meets computer science. Its function is to control which principals(persons, processes, machines, . . .) have access to which resources in the system— which files they can read, which programs they can execute, how they share data with other principals, and so on.

Assurance that each user or computer that uses the service is permitted to do what he or she asks for. The process of authorization is often used as a synonym for access control, but it also includes granting the access or rights to perform some actions based on access rights. Access control is the ability to permit or deny the use of a particular resource by a particular entity. Access control mechanisms can be used in managing physical resources (such as a movie theater, to which only ticket holders should be admitted), logical resources (a bank

account, with a limited number of people authorized to make a withdrawal), or digital resources (for example, a private text document on a computer, which only certain users should be able to read).

Access control as an important protection mechanism in computer security is evolving with the development of applications. Since the early 1970s, several access control models have appeared, including Discretionary Access Control (DAC), Mandatory Access Control (MAC), and Role Based Access Control (RBAC). These models are considered identity-based access control models, where subjects and objects are identified by unique names and access control is based on the identity of the subject, either directly or through roles assigned to the subject. DAC, MAC and RBAC are effective for closed and relatively unchangeable distributed systems that deal only with a set of known users who access a set of known services.

Discretionary Access Control (DAC) was designed for multi-user databases and systems with a few previously known users. Changes were rare and all resources were under control of a single entity. Access control is based on the identity of the requestor and the access rules stating what requestors are (or are not) allowed.

Mandatory Access Control (MAC) had its origins in military environments where the number of users can be quite large, but with a static, linearly hierarchic classification of these users. This model is based on the definition of a series of security levels and the assignment of levels to resources and users. MAC policies control access based on mandated regulations determined by a central authority.

Role-based Access Control (RBAC) is inspired from the business world. The development of RBAC coincides with the advent of corporate intranets. Corporations are usually hierarchically structured and access permissions depend on the position of the user in the hierarchy. The roles are collections of entities and access rights grouped together based on different tasks they perform in the system environment. The main idiosyncrasy of role based access control is that the mechanisms are built on three predefined concepts: “user”, “role” and “group”. The definition of roles and the grouping of users can facilitate management, especially in corporate information systems, because roles and groups fit naturally in the organizational structures of the companies. However, when applied to some new and more general access control scenarios, these concepts are somewhat artificial. A more general access control approach is needed in these new environments. There are various other approaches for access control in grid computing, which are capable of fulfilling other requirements of specific domain. These approaches are: Attribute based access control, Agent based access control, Policy based access control framework, Semantic access control, and Semantic-aware access control for grid system.

### III. RELATED WORK

Bo Lang et al. [1] proposed an attribute based access control for grid computing. There are usually a large number of users, the users are changeable. The traditional access control models that are identity based are closed and inflexible. The Attribute Based Access Control (ABAC) model, which makes decisions relying on, attributes of requestors, resources, and environment, is scalable and flexible and thus is more suitable for distributed, open systems. But no ABAC model meets the special authorization requirements of Grid computing as different domains have their own policies. For this purpose, ABMAC (Attribute Based Multipolicy Access Control) model is presented based on ABAC and the authentication requirements of grid system.

Ernesto Damiani et al. [2] presents new paradigms for access control in open environments. In this paper, the emerging trends in the access control field are presented to address the new needs of today's systems. In particular, two new access control paradigms have been discussed. These are attribute-based access control and Semantics-aware access control. In this paper, they have surveyed the current state and future trends in the access control area and we have seen that they are a crucial part of tomorrow's communication infrastructure supporting mobile computing systems.

Lin Jie et al. [3] implements agent-based access control security in grid computing environment. This paper presents a framework of access control security in grid computing environment. Considering the dynamic characteristic and multi-management domains in grid computing, a model of combining inner-domain and cross-domain access control mechanism is designed which is based on the intelligent agent and basic network access control technology. It's a try on setting a unified Grid security strategy.

Jin Wu et al. [4] develop a policy-based access control framework for grid computing. It is important to provide a uniform access and management mechanism couple with fine-grain usage policies for enforcing authorization. In this paper, work on enabling fine-grain access control is described for resource usage and management. A prototype as well as the policy mark-up language is described that is designed to express fine-grain security policies. A fine-grain resource access control is implemented as an extension to Globus Toolkit (GT2).

In order to protect the information and resource from abuse, Wang Xiaopeng et al. [5] introduce new approach for access control in grid computing. In this paper, a new approach is presented to authorize and administrate Grid access requests, in which the requests are obliged to negotiate with a policy enforcement system in order to gain access to the target Grid resources. This new method is called semantic access control as it will exploit semantic web technology, and use machine reasoning about the messages and policies at a semantic level. The security elements' attributes are semantically described by XML documents in special formats. Based on these formalized policy languages and attributes specification documents, machine reasoning is easily performed to make the access permit decision.

Xiyuan Chen et al. [6] proposed the Semantic-Aware Access Control model for Grid Application based on RBAC (Role Based Access Control) for the Grid Application. The Semantic-aware access control model extended the RBAC model by supplying semantic specification for each element in the access control model for the sake of better describing the relationship of each element and security information.

### IV. CONCLUSION

We have studied various access control approaches for grid system. In open environment, such as internet, the decision to grant access to resource is often based on the characteristics of the requester rather than the identity. Attribute based access control, helps in making access decisions based on attributes of requestors, resources. In the Future Work, on the basis of characteristics rather than predefined identities, by making use of policies and authorization requirements of grid system, implementation of access control method for grid can be done.

### V. REFERENCES

- [1]. Bo Lang, Ian Foster, Frank Siebenlist, Rachana Ananthakrishnan, Tim Freeman, "Attribute Based Access Control for Grid Computing", In 5th Annual PKI R&D Workshop, April 2006.
- [2]. E. Damiani, S. De Capitani di Vimercati, and P. Samarati, "New Paradigms for Access Control in Open Environments", In Proc. 5th IEEE International Symposium on Signal Processing and Information, Athens, Greece, 18-21 December, 2005.
- [3]. Lin Jie, Wang Cong, Guo Yanhui, "Agent-based access control security in grid computing environment", 19-22 March 2005, pp:159-162.
- [4]. Jin Wu, Chokchai Box Leangsuksun, Vishal Rampure, "Policy-based Access Control Framework for Grid Computing", 6-19 May 2006, pp: 391 - 394 .
- [5]. Wang Xiaopeng, Luo Junzhou, Song Aibo, Ma Teng, "Semantic Access Control in Grid Computing", Parallel and Distributed Systems, 2005. Proceedings. 11th International Conference, 20-22 July 2005, pp: 661-667.
- [6]. Xiyuan Chen, Yang OUYang, Miaoliang Zhu, Yan He, "Semantic-Aware Access Control for Grid Application", Young Computer Scientists, 2008. ICYCS 2008. The 9th International Conference, Hunan, China, 18-21 Nov. 2008, pp: 971-975.

# Fragile Watermarking for Image Authentication in DCT domain

Shilpy Ghai- Student M Tech, Sukhjeet Kaur Ranade- Senior Lecturer  
Department of Computer Science, Punjabi University, Patiala, Punjab

**Abstract-** The tremendous growth in computer and network technology has led to new era of digital multimedia. It has raised some issues like intellectual properties and data authentication. Digital watermarks can provide a solution to these issues. The major applications of digital image watermarking are copyright protection and image authentication. The later one is provided by fragile watermarking. A fragile watermark is a mark that is readily altered or destroyed when the host image is modified.

This paper presents a novel DCT-based watermarking method for image authentication. In this method, a watermark in the form of a visually meaningful binary pattern is embedded for the purpose of tamper detection.

**Keywords:** Fragile watermarking, image authentication, tamper detection, DCT domain.

## I. INTRODUCTION

Over the past few years, there has been rapid development in computer networks and more specifically, World Wide Web. Digital multimedia has brought many advantages like easy creation, edition and distribution of image content but the ease of copying and editing also facilitates unauthorized use as well, such as illegal copy and modification of this content. So the demand of security is getting higher in these days due to easy reproduction of digitally created multimedia data.

Digital watermarks have been proposed as a way to tackle this tough issue. Digital image watermarking is an emerging technique to embed secret information (i.e. watermark) into the image for copyright protection and authentication. A digital watermark is a pattern of bits inserted into a digital image, audio or video file that may be perceptually visible or invisible depending upon the requirements of the user.

There are several types of watermarking like robust, fragile and semi fragile watermarking. Robust watermarks are those which resist the attacks that attempt to remove or destroy the marks. Robust watermarks provide copyright protection. But fragile marks act in an opposite way. These marks get altered when the host image is modified. And semi fragile watermarking schemes are sensitive only to malicious content modification but not to normal signal processing such as compression.

## II. FRAGILE WATERMARKING SYSTEM

A fragile watermark is a mark that is readily altered or destroyed when the host image is modified through a linear or non linear transformation. A fragile mark is designed to detect slight changes to the watermarked image with high probability. Fragile watermarks allow the determination of the exact pixel locations where the image has been modified. The insertion of watermark is perceptually invisible under normal human observation. A fragile marking system detects any tampering in

a marked image. The process of embedding and detecting a fragile mark is similar to that of any watermarking system.

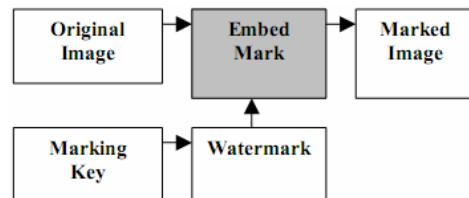


Figure 1: Watermark Embedding

The marking key is used to generate the watermark and is typically an identifier assigned to the owner of image. The original image is kept secret. When a user receives an image, they use the detector to evaluate the authenticity of the received image. The detector locates and characterizes alterations made to a marked image.

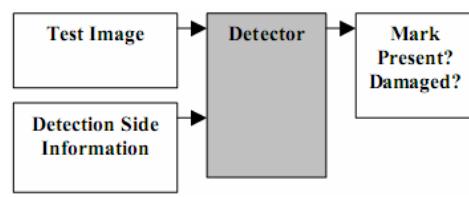


Figure 2: Watermark Detection

The main application area of fragile watermarking is in image authentication. This method is useful to the area where content is so important that it needs to be verified for it being edited, damaged or altered such as medical images. Kallel et al. [1] proposed different methods to verify integrity and authenticity of medical images. Image authentication systems have also applicability in law, commerce, defense and journalism. A combination of digital signature and fragile watermark is used to provide security to important documents in e-governance [2].

Fragile watermarks are prone to various attacks. One type of attack is blind modification of a marked image that is, arbitrarily changing the image assuming no mark is present. Variations of this attack include cropping and localized replacement such as substituting one person's face with another. A texture-based temper detection scheme [3] is sensitive to this type of attack. Another type of attack is that in which the attacker modifies the image without affecting that area where the watermark is present. The attacker may also remove the watermark completely which is being used for authentication purpose. For this purpose, an attacker may attempt to add random noise to the image using techniques designed to destroy the image.

### III. RELATED WORK

We now survey some fragile marking systems described in literature, we can classify these techniques as ones which work directly in spatial domain or in transform domain.

In spatial domain fragile watermarking schemes, the mark is embedded by directly modifying the pixels values. One of the first techniques used for detection of image tampering was based on inserting check-sums into the least significant bit (LSB) of image data [4].

Van Schyndel et al. [5] modify the LSB of pixels by adding extended m-sequences to rows of pixels. The phase of the sequence carries the watermark information. A simple cross-correlation is used to test for the presence of the watermark. As with any LSB technique, this method will provide a low degree of security and will not be robust to image operations with low-pass character. Their significant disadvantage was the inability to lossy compress the image without damaging the mark.

Wolfgang and Delp [6] extended van Schyndel's work and improved the localization properties and robustness. They use m-sequences of -1's and 1's arranged into  $8 \times 8$  blocks and add them to corresponding image blocks. Their technique is moderately robust with respect to linear and nonlinear filtering and small noise adding. Since the watermark is inserted in the LSB plane, it can be easily removed.

The security of the technique depends on the difficulty of inferring the look-up tables. The search space for the table entries can be drastically reduced if knowledge of the bi-level watermark image is known. A modification (position-dependent lookup tables) is proposed to dramatically increase the search space.

But these schemes are not directly suitable for some applications where transformation is needed to compress the images. So some transform domain schemes have been proposed which have the benefit that mark is embedded in compressed representation. Watermark is embedded in the transform domain e.g., DCT, DFT, wavelet by modifying the coefficients of global or block transform. The properties of a transform can be used to characterize how an image has been damaged or altered.

Wu and Liu [7] describe a technique based on a modified JPEG encoder. The watermark is inserted by changing the quantized DCT coefficients before entropy coding. A special lookup table of binary values is used to embed the mark.

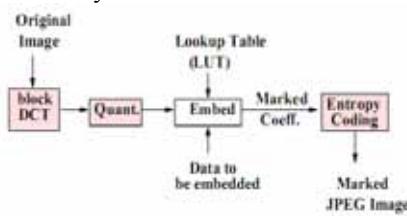


Figure 3: Block Diagram of Embedding Process

Li [8] describes a scheme to implicitly watermark all the coefficients. By involving the un-watermarked non-zero coefficients in watermarking process of DCT coefficients and registering the zero coefficients with watermark, the proposed scheme minimizes the distortion due to watermarking

embedding while providing the capability of authenticating all the coefficients including the zero ones. The idea of watermarking DCT coefficients according to secret sum extracted from dependence neighborhood puts up resistance to cropping, cover up, vector quantization and transplantation attacks. The original image is not required during watermark extraction process. No cryptography and hashing are needed in it.

Owing to recent advances in network and multimedia techniques, digital images can be easily transmitted over the internet. A web based image authentication method based in digital watermarking is described in [9] which provides more control to image owners and conveniences to clients. A client server model is used in this scheme. Server holds watermark detection method internally and client can access to the server using internet to verify the image.

He et al. [10] describes a fragile water-marking scheme for secure image authentication. A scrambling encryption scheme is introduced to extend security and increase the capacity of fragile watermarking algorithm. The proposed algorithm possesses tamper localization properties and tamper discrimination which is used to detect whether the modification made to image is on the contents or the embedded watermark or both.

Wu et al. [11] introduces a scheme which points out the tampered positions in the image and restores the image. DCT is applied to take certain frequency coefficients from the image as characteristic values which are embedded into least significant bits of image pixels, used to provide proof of image integrity. If the image is tampered, the characteristic values that are affected changed accordingly and then detected. The proposed recovery process restores the image using characteristic values of original image.

### IV. PROPOSED EMBEDDING ALGORITHM

Consider an  $H \times W$  original image  $O$  to be watermarked. The watermark  $W$  to be embedded is of size  $X \times Y$  and consists of a visually meaningful binary pattern (e.g. a logo). The image is first divided into  $B$  blocks each of size  $N \times N$ . The DCT is applied to each block  $B_i$  to produce a corresponding block  $C_i$  of DCT coefficients. A user key  $k$  is used to generate a random coefficient-selection vector  $S_i$  which denotes the index of the coefficient to be watermarked in block  $C_i$ .

Let  $w_i$  be the watermark bit to be embedded in coefficient  $C_{s_i}$  of block  $C_i$ . The watermarked coefficient

$$c'_{s_i} = \begin{cases} c_{s_i} & , \text{if } Q(c_{s_i}) = w_i \\ c_{s_i} + \Delta & , \text{if } Q(c_{s_i}) \neq w_i \text{ and } c_{s_i} \leq 0 \\ c_{s_i} - \Delta & , \text{if } Q(c_{s_i}) \neq w_i \text{ and } c_{s_i} > 0 \end{cases}$$

where  $\Delta$  is called the quantization parameter, and  $Q(c)$  is a coefficient-binary mapping function given by:

$$Q(c) = \begin{cases} 0 & , \text{if } \left\lfloor \frac{c}{\Delta} \right\rfloor \text{ is even} \\ 1 & , \text{if } \left\lfloor \frac{c}{\Delta} \right\rfloor \text{ is odd} \end{cases}$$

These equations indicate that the embedding process is based on shifting the selected coefficient, to have mapped value in coefficient-binary mapping function that is identical to the watermark bit  $w_i$

#### V. EXTRACTION ALGORITHM

In the extraction process, the received, and possibly tampered with, image is first divided into  $B$  blocks each of size  $N \times N$ . The DCT is applied to each block  $B$  to produce a corresponding block of DCT coefficients. The same user key  $k$  that was used at the transmitter is available also at the receiver. This vector is used to locate the watermarked coefficient in each block. The extracted watermark bit is obtained as follows:

$$w'_i = Q(c''_{s_i})$$

#### VI. PERFORMANCE MEASURES

In order to assess the effect of the method on the objective quality of the watermarked image, the PSNR measure is calculated.

$$PSNR = 10 \log_{10} \left[ \frac{255^2}{\frac{1}{H \times W} \sum_{y=1}^H \sum_{x=1}^W (o(x, y) - o'(x, y))^2} \right]$$

In order to assess the extent of tampering, a tamper assessment function (TAF) is calculated.

$$TAF = \frac{1}{B} \sum_{i=1}^B w_i \oplus w'_i$$

Lower TAF values indicate less tampering. Normally, a predefined threshold  $\tau$  is selected and compared with the TAF to determine the authentication state (flag) of the received image. When the TAF is smaller than the threshold  $\tau$ , the modification to the image is considered as legitimate and negligible. However, when the TAF is greater than the threshold  $\tau$ , the modification to the image is considered as Illegitimate.

#### VII. RESULTS AND DISCUSSIONS

The proposed method is tested using a number of standard 8-bits grayscale  $512 \times 512$  images and watermark images of size  $256 \times 256$ . We are applying different attacks on the images and calculating values for PSNR, TAF and Correlation.

Attacks	PSNR	TAF	Correlation
Without attack	49.66	0.517	0.968
Median Filtering	47.96	0.820	0.935
Brightening by +1	47.35	0.943	0.697
Contrast Stretching	48.35	0.617	0.918

#### VIII. CONCLUSION

A fragile marking system is useful in a variety of image authentication applications. A novel DCT-based watermarking method for image authentication has been developed in this

paper. In this method a watermark in the form of a visually meaningful binary pattern is embedded for the purpose of tamper detection. The method was tested under different attacks and was found to provide good detection performance while maintaining the quality of the watermarked image. Depending on the type of attack, the method may provide exact authentication, selective authentication, or localization.

#### IX. REFERENCES

- [1] I.F.kallel, M.kallel and M.S.Bouhlel, "A secure fragile watermarking algorithm for medical image authentication in the DCT domain," ICTTA 2006, vol.1, pp. 2024-2029.
- [2] Liu, "The application of fragile watermark in E-governance," ICMECG 2008, pp. 136-139.
- [3] Y. Liu, W. Gao, H. Yao and S. Liu, "A texture-based tamper detection scheme by fragile watermark," ISCAS 2004, vol. 2, pp. 177-180.
- [4] S. Walton, "Information authentication for a slippery new age," Dr. Dobbs Journal, vol. 20, Apr. 1995, pp. 18-26.
- [5] R. van Schyndel, A. Tirkel, and C. Osborne, "A digital watermark," Proc. of the IEEE Int. Conf. on Image Processing, vol. 2, Nov. 1994, pp. 86-90.
- [6] R. Wolfgang and E. Delp, "A watermark for digital images," Proc. of the IEEE Int. Conf. on Image Processing, vol. 3, 1996, pp. 219-222.
- [7] M. Wu and B. Liu, "Watermarking for image authentication," in Proc. IEEE ICIP, 1998, vol. 2, pp. 437-441.
- [8] C.-T. Li , "Digital fragile watermarking scheme for authentication of JPEG images," IEEE proc. Vision, image and signal processing, vol. 151, 2004, pp. 460-466.
- [9] Y. Lim, C. Xu and D. D. Feng, "Web based image authentication using invisible fragile watermark," ICACM 2001, vol. 11, pp.31-34.
- [10] H.He, J.Zhang and H.-M. Tai, "A secure fragile watermarking scheme for image authentication," ICCIAS 2006, vol. 2, pp. 1180-1185.
- [11] H.-C. Wu, C. -C. Chang and T.-W. Yu, "A DCT-based recoverable image authentication technique," JCIS 2006 proc., Oct. 2006.

# Implementation of Robust Watermarking in DCT Domain

Priyanka Jarial- Student M.Tech, Sr. Lecturer-Sukhjeet Kaur Ranade  
Department of Computer Sciences, Punjabi University, Patiala

**Abstract-** In this paper the discrete cosine transform (DCT domain) watermarking technique is being analyzed for the image authentication and copyright protection. It covers the research given by various authors in order to examine the robustness of the watermarking system against the attacks like common signal processing operations and geometric distortions. The various watermark embedding and extraction techniques are used by different. The DCT is applied upon the image by using the block of 8×8 pixel.

**Keywords-** Digital Watermarking, Robustness, Image authentication, Copyright protection, DCT domain.

## I.INTRODUCTION

WATERMARKING is the process that embeds data called watermark. A watermark, a tag, or label in a multimedia object that can be detected or extracted later on to make assertions about the object. The multimedia object may be an image, audio, video, or text and the insertion can be done in spatial domain, discrete cosine-transformed (DCT), or wavelet-transformed. Watermarks and watermarking techniques can be divided into various categories. Depending upon the Human perception watermarking can be a visible watermarking or invisible watermarking. Watermarking undergoes various attacks<sup>[1]</sup> It has also been concluded that the frequency-domain methods are much more robust than the spatial-domain techniques. Moreover, in terms of computational efficiency the frequency domain watermarking schemes gave better results than the spatial domain scheme. Watermarks of varying degree of visibility are added to present media as a guarantee of authenticity, ownership, source, and copyright protection. There are two parts to building a strong watermark: the watermark structure and the insertion strategy. In order for a watermark to be robust and secure, these two components must be designed correctly. We provide two key insights that make our watermark both robust and secure: We state that the watermark should be placed explicitly in the perceptually most significant components of the data.

## II.REVISED PAPERS

R.B.Wolfgang and E.J. Delp<sup>[2,3]</sup> has described an invisible watermarking technique in spatial domain. Further I. Pitas uses<sup>[4]</sup> an approach that allows information to be embedded in order to resist various attacks. Van Schyndel et.al<sup>[5]</sup> scheme of sequences was later enhanced by Cox. et.al<sup>[6]</sup> to improve the embedding strength of the mark.

S.P.Mohanty<sup>[7]</sup> represented the visible watermarking based technique for embedding and extracting watermark. He presented his paper for invisible robust watermarking for embedding and extracting the watermark<sup>[8]</sup>. In this paper the compound creation of the watermark in the most significant

region of the host image facilitates the homogeneous fusion of watermark.

Gengming Zhu and Nong sang gave their research algorithm based upon DCT block<sup>[9]</sup> that uses video frequency watermark proposal based on discrete cosine transform (DCT) domain DC component (DC). He concluded that it is much more suitable to embed watermark with DC component rather than AC component (AC) due to its larger perceptual capacity and resistance to common signaling operations.

Hsiang cheh Huang,Jeng-Shyang Pan, Chi-Ming Chu<sup>[10]</sup> designed the genetic algorithm system in order to obtain the survivability, acceptance and the capacity of the watermarking algorithm. He listed various requirements of the Robust watermarking algorithm and described the algorithm based upon conceptual optimization Based upon the existing metrics--survivability, acceptance and the capacity they took the number of embedded bits into account. As this additional feature proved to evaluate the applicability of robust watermarking while implementation. Peak Signal-to-Noise Ratio (PSNR) is used to represent the measure for imperceptibility. Bit-Correct Rate (BCR) is used as the measure of robustness, whereas the number of bits embedded into the original image denotes the Watermark capacity.

Hye-Joo Lee,Ji-Hwan park and Yuliang Zheng<sup>[11]</sup> proposes new method that uses the JPEG quality level as a parameter.Hence with the use of this quality level they embedded watermark into an image. They also added new feature that the watermark can be extracted even when the image is compressed using JPEG. As a result of this, we obtain a watermarking method that is robust against the JPEG compression.

Shelby Pereira, Sviatoslav Voloshynovskiy and Thierry Pun<sup>[12]</sup> has also described the effective channel coding strategies which can be used in conjunction with linear programming optimization techniques for the embedding of robust perceptually adaptive DCT domain watermarks. They proposed a coding strategy based on the magnitude of a DCT coefficient, which uses turbo codes for effective error correction, and finally the incorporation of JPEG quantization tables during embedding.

For channel coding Shelby Pereira, Sviatoslav Voloshynovskiy and Thierry Pun<sup>[13]</sup> he proposes using the magnitude of the coefficients. To encode a 1 we will increase the magnitude of a coefficient and to encode a0 we will decrease the magnitude. At decoding a threshold T will be chosen against which the magnitudes of coefficients will be compared. The Magnitude coding strategy is summarized as follows, In table, ci is the selected DCT.

sign( $c_i$ )	bit	Coding
+	0	decrease $c_i$ (set $L$ to stop at 0)
-	0	increase $c_i$ (set $U$ to stop at 0)
+	1	increase $c_i$
-	1	decrease $c_i$

Sangoh Jeong<sup>1</sup> Kihyun Hong<sup>2</sup> Chee Sun Won<sup>[14]</sup> gave the dual watermarking technique against JPEG compression that is robust enough to resist the attacks. Their proposed algorithm was advantageous as it was robust enough against broader attacks such as compression etc. The different watermarking techniques used under DCT domain to measure the robustness of the watermark-- *Image independent DCT domain embedding* and *Image-dependent DCT domain*. In the former domain, the DCT coefficients of the original image are ordered by a zigzag scan as in JPEG. Then, the coefficients from the (L+1)<sup>th</sup> to the (L+M)<sup>th</sup> are taken, where L is the first position of the ordered DCT coefficients and M is the number of DCT coefficients to be watermarked. Watermark data n is embedded into the chosen DCT coefficients whereas in later domain the DCT coefficients are embedded for each block<sup>[15]</sup>.

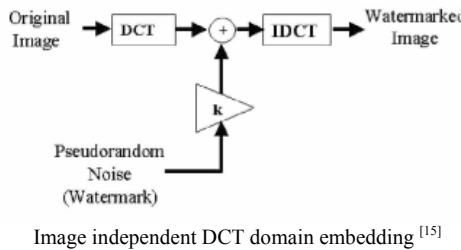


Image independent DCT domain embedding<sup>[15]</sup>

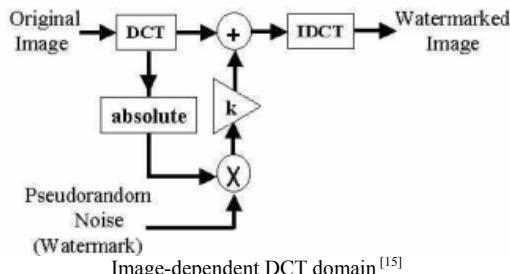
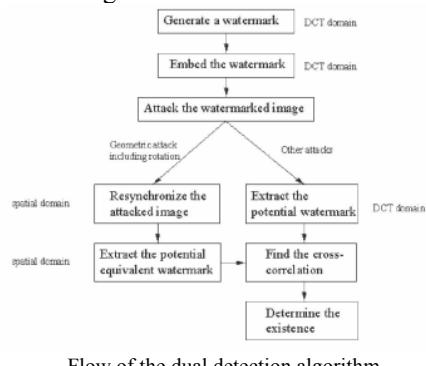


Image-dependent DCT domain<sup>[15]</sup>

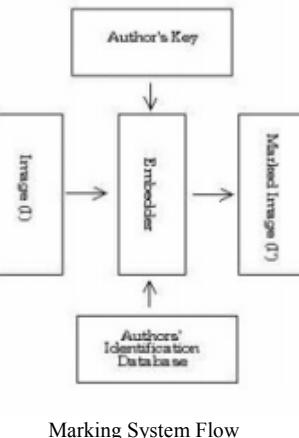
The proposed algorithm given by Gerardo Pineda Betancourt, Ayman Hagag, Mohamed Ghoneim, Takashi Yahagi, and Jianming Lu<sup>[15]</sup> is as follows:



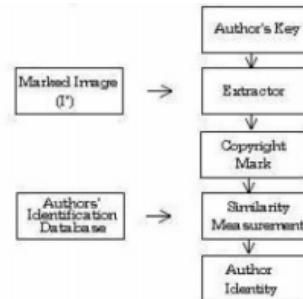
Flow of the dual detection algorithm

C. M. Kung, T. K. Truong, Fellow, J. H. Jeng, Senior Member<sup>[16]</sup> gave basic idea to protect the information in the frequency domain by changing the magnitude of some of the DCT coefficients. To achieve the robustness property the quantization algorithm of JPEG is used. The proposed scheme, is that the information will be hidden in the middle frequency region which will be hence selected according to the values of the table Q(u,v).

Raja.S.Omari and Ahmed-Al Jaber<sup>[17]</sup> proposed watermarking algorithm that uses DCT coefficients for embedding and hence in order to increase the robustness the high frequency coefficients were selected for embedding. They proposed watermarking algorithm to insert and extract the watermark using DCT domain. After giving the watermarking scheme, they listed various features that are required for building the copyright protection system. The proposed copyright protection system must contain--*The Marking System, the Mark Detector system and The Authors identification database*. It is shown as follows:



Marking System Flow



Mark Detector Systems workflow

The marking system is the embedder itself who embeds the input parameters like input image, identification mark etc.

The Mark Detector is the extractor who decides the authorization by comparing the extracted mark and the selected mark from the database. The author's identification marks are uniquely identified by comparing the extracted mark with the database of the properties of authors like identification number, fingerprints, and images and so on.

### III.EMBEDDING ALGORITHM

Convert message M into stream of bits and save in b.  
Compute the decimal equivalent of each m-bit string. Starting from first bit until the end and save in Num.

- While i < NumSize DO
- I\_DCT<sub>i</sub>; Find the DCT coefficients of the block (I<sub>i</sub>).
- h: Summation of I\_DCT values (a predefined set of high frequency) (exclude the DC term).
- S: h mod (2<sup>m</sup>).
- Diff: Num<sub>i</sub> – S.
- Find the two values ToAdd and ToSub according the following map:
- If Diff ∈ [- (2<sup>m</sup>) + 1, floor((-(2<sup>m</sup>) + 1) / 2)] Then ToAdd = -(2<sup>m</sup>) + Diff.
- If Diff ∈ [floor(-(2<sup>m</sup>) + 1) / 2, -1] Then ToSub = -1 × Diff.
- If Diff ∈ [0, floor((2<sup>m</sup>) - 1) / 2] Then ToAdd = Diff.
- If Diff ∈ [(2<sup>m</sup>) / 2, (2<sup>m</sup>) - 1] Then ToSub = (2<sup>m</sup>) - Diff.
- Either ToAdd or ToSub should be mapped (Only One of them).

### IV. EXTRACTION ALGORITHM

- Loop n / m times on the Watermarked Image I'.
- I\_DCT<sub>i</sub>; Find the DCT coefficients of the block (I<sub>i</sub>).
- h: Summation of I\_DCT<sub>i</sub>'s values (predefined set of high frequency).
- S: h mod 2<sup>m</sup>.
- Let S<sub>b</sub> be the binary representation of the decimal value S.
- Concatenate S<sub>b</sub> to M (which will lastly contain binary representation of the entire message).
- Convert the output as the desired form such as an image.

### V. CONCLUSION

In this paper, we have proposed various techniques to make the watermark robust against the malicious attacks. As we know that in daily routines the images are compressed and decompressed regularly, so in order to resist the various proposed methods has been studied to overcome this flaw. It has been concluded that in order to make the watermark robust The DCT domain is considered to be the best domain for implementation over the spatial domain.

### VI. REFERENCES

- [1]. Fabien A.P. ,Petit colas., Ross J., Anderson., Markus G. Kuhn, "Attacks on Copyright Marking Systems", In proceedings of second international workshop of information hiding,p-218-238 ,April14-17,1998.
- [2]. R. G. Wolfgang and E. J. Delp, "A Watermark for Digital Images", Proc. IEEE Intl. Conf. on Image Processing, ICIP-96, Vol.3, pp.219-222.
- [3]. R. B. Wolfgang and E. J. Delp, "A Watermarking Technique for Digital Imagery: Further Studies", Proc. Intl. Conf. on Imaging Sciences, Systems and Tech., Los Vegas, June 30-Jul 3, 1997,<http://dynamo.ecn.purdue.edu/ac/delp-pub.html>
- [4]. N. Nikolaidis and I. Pitas, "Robust Image Watermarking in Spatial Domain", Signal Procesing,Vol.66, No.3, pp 385-403.
- [5]. Van Schyndel, R.G., Tirkel, A.Z., Osborne, C.F." A digital watermark". In: Proc. IEEE Int. Conference on Image Processing, Austin, Texas, USA (1994) 86-89
- [6]. Cox, I. J., Kilian, J., Leighton, T. and Shamoon, T., "Secure Spread Spectrum Watermarking for Multimedia". NEC Research Institute, Technical Report 95-10, Princeton, NJ, October 1995
- [7]. Mohanty S P., "Digital Watermarking: A Tutorial Review", Mohanty 99 digital watermarking,Technical Report,1999.

- [8]. Mohanty S P, Bhargava Bharat K," Invisible Watermarking Based on Creation and Robust Insertion-Extraction of Image Adaptive Watermarks", Volume 5, Article No. 12, Nov 2008, ISSN 1551-6857.
- [9]. Zhu Gengming, Sang Nong," Watermarking Algorithm Research and Implementation Based on DCT Block ",Proc. of world Academy of Science, Engineering and Technology, ISSN 2070-3740,Volume 35,Nov 2008.
- [10]. Huang cheh Hsiang, Shyang Jeng, Ming Chu Chi,"Optimized Copyright Protection Systems with Genetic-Based Robust Watermarking",Vol 13, Issue 4(October 2008) Special Issue on Bio-Inspired Information Hiding,pp 333-343,2008,ISSN 1432-7643.
- [11]. Chi-Ming Chu A, Hsiang-Cheh Huang, Jeng-Shyang Pan, " An Adaptive Implementation for DCT-Based Robust Watermarking with Genetic Algorithm",p-19,2008,ISSN: 978-0-7695-3161-8.
- [12]. Lee Joo Hye-, park Hwan Ji, Zheng Yuliang,"Digital Watermarking Robust Against JPEG Compression", Information Security, Volume No. 1729, 1999.
- [13]. Pereira Shelby, Voloshynovskiy Sviatoslav,Pun Thierry,"Effective channel coding for DCT watermark", Image Processing, 2000. Proceedings. 2000 International Conference onVolume3,Issue2000,Page(s):671-673vol.3, Digital Object Identifier 10.1109/ICIP.2000.89954.
- [14]. Sangoh Jeong, Kihyun Hong, Chee Sun Won," Dual Detection of Watermarks Embedded in the DCT Domain", ELMAR, 2005. 47th International Symposium June 2005,pp 103-106,ISBN 953-7044-01-4.
- [15]. Betancourt Gerardo Pineda, Haggag Ayman, Ghoneim Mohamed, Yahagi Takashi, Jianming Lu , "Robust Watermarking in the DCT Domain Using Dual detection",Industrial Electronics, 2006 IEEE International Symposium ,2006 july,Vol 1,pp 579-584.
- [16]. Kung CM, Truong.TK., Fellow, Jeng J H, Senior Member , "A Robust Watermarking and Image Authentication Technique", Proceedings. IEEE 37th Annual 2003 International Carnahan Conference,Oct 2003,pp 400-404,ISBN 0-7803-7882-2.
- [17]. Omari Raja.S,Jaber Ahmed- Al , " Robust watermarking algorithm for copyright protection",2005,p 90,ISBN : 0-7803-8735-X.

# Design & Simulation of 12 bit Pipelined Analog to Digital Converter with improved Cascading Technique using 0.35μ TSMC Technology

Neeru Agarwal <sup>1</sup>, S. C. Bose, Anil Saini <sup>2</sup>, Rahul Tripathi <sup>3</sup>, Neeraj Agarwal <sup>4</sup>

<sup>1</sup>Electronics Engineering Department, Amity School of Engineering & Technology, Amity University, Noida

<sup>2</sup> CEERI (CSIR) Pilani, Rajasthan, <sup>3</sup>Semiconductor Laboratory, Mohali, Chandigarh

<sup>4</sup>Institute of Technology & Management, Meerut<sup>4</sup>. E: mail [eng.neeru@gmail.com](mailto:eng.neeru@gmail.com)

**Abstract-** This Proposed work presents a new architecture with reduced circuit components and complete circuit realization of its individual circuit blocks. This improved architecture applies an indigenous gain stage for multiply by two function particularly used for the subtraction and amplification block. When proposed technique was implemented the circuit was greatly improved and the resolution of the ADC reached 12-bit. The circuit was processed in 0.35μ standard CMOS process technology, and the advantage of the circuit structure has been verified. In this block we are successfully cascading the 12 bit without any degradation with the use of high gain buffer. This Proposed A/D architecture, analyzed its characteristic, discusses main building block circuit structure, and presents implementation and circuit simulation results. This presentation focuses on highlighting the circuit techniques employed in recent low-voltage ADCs while identifying their advantages and disadvantages.

**Key words:** CMOS, residue voltage, buffer, Pipelined ADC

## I. INTRODUCTION

Today's emergent technology for the improvement of performances of electronics information systems, analog/digital converter (ADC) must improve its performance in speed and precision greatly. We can say Analog-to-digital converters (ADCs) are key design blocks in modern microelectronic communication systems. The fast advancement of CMOS fabrication technology requires more and more signal processing functions which are implemented for lower cost, lower power consumption and higher yield. This has recently generated a great demand for low-power, low-voltage ADCs. It can be say that Data converters (ADCs and DACs) ultimately limit the performance of today's communication systems and consumer products. Therefore, high-speed, high-resolution, and power-aware converters are required [2-3].

By combining the merits of the successive approximation and sigma delta ADCs, this proposed Pipelined ADC architecture is fast has a high resolution and only requires die size. There is one complete conversion per clock cycle. If chip space is available additional stages can be added for increased resolution. In this proposed work of 12 bit Pipelined ADC represents the optimal trade-off among speed and power Consumption with low to medium-resolution requirements. This pipelined ADC is relatively easy to implement in a monolithic form and also simpler to compensate for error sources for high resolution [2-5].

The pipelined A/D converter to be described is a serial converter made of 1-bit A/D converters as illustrated in Fig. 1.

The pipelined in general is a sampled-data system and has a high throughput rate with relatively little circuitry due to the concurrent operation of each stage. We can say that the concept of pipelining, popular in digital circuits and microprocessors, consists of performing a number of operations serially in order to obtain a higher data throughput [2].

We know Data converters are mixed-signal circuits containing both analog and digital circuits. Hence, ADCs are no exception to the problem described above. When operated by a low-voltage supply these circuits suffer from reduced signal swings that limit their dynamic range. These limitations are produced by the constant transistor threshold voltage (VTH) that results in lower gate overdrive (VGS - VTH) as the supply voltage becomes lower. Moreover, unlike digital circuits, analog circuits do not benefit with lower power dissipation as the supply voltage is reduced. Now a days it becomes very challenging to design such circuits since the LSB approach noise levels. The urgent need for solutions to the problem of low-voltage circuit operation therefore becomes obvious.

Section II describes the proposed architecture and details the CMOS implementation of its building blocks. Experimental results are presented in Section III, and Section IV concludes this paper.

## II. SECTION

### DESIGN PROCEDURE OF PROPOSED CASCADED 12 BIT PIPELINED ADC

This work presents a new architecture with reduced circuit components and completes circuit realization of its individual circuit blocks. The beauty of this architecture is that it is designed without using digital to analog converter (DAC), here we are successfully converting the analog value in to corresponding digital output. There may be less power dissipation because this Pipelined ADC needs only one additional operational amplifier and a comparator compared to traditional one Pipeline A/D architecture.

After the input signal has been sampled compared it to Vref/2. If  $V_{in} > V_{ref}/2$  (comparator output is 1),  $V_{in}$  is subtracted from the held signal and pass the result to the amplifier. If  $V_{in} < V_{ref}/2$  (comparator output is 0), then pass the original input signal to the amplifier. The output of each stage in the converter is referred to as the residue [2].

The number of comparators in the two-step flash is doubled for an additional bit of resolution while four

operations were carried out, namely coarse A/D conversion, D/A conversion, subtraction, and fine A/D conversion

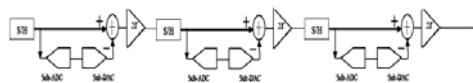


Fig.1 Block diagram of Pipelined Analog to Digital Converter

Pipelining in ADCs makes use of analog preprocessing in order to execute all these operations concurrently. A reference voltage  $V_{ref}$  is subtracted from the doubled input, and the residue voltage is compared with zero. If positive, the bit is ONE and the residue is sampled by the following stage. However, if negative, the bit is ZERO and the reference voltage is added back to the residue by switching back to the ground before it is sampled by the following stage. This Proposed 12 bit Pipeline ADC needs passing only 12 operational amplifier, 12 comparator compared to traditional one architecture to save more area and power consumption. It increases output data from 12-bit to 16-bit stage. For this specification of buffer is as shown below. Therefore, the sampled analog voltage is pipelined to determine digital bits sequentially starting from the most significant bit (MSB) to the least significant bit (LSB).

### III. DESIGN OF BUFFER

#### AC Analysis of buffer

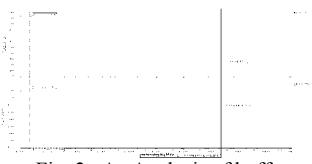


Fig. 2 Ac Analysis of buffer

#### DC analysis of buffer for ICMR



Fig. 3 Dc analysis of buffer for ICMR

#### Transient analysis of Buffer

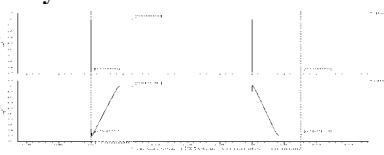


Fig. 4 settling time response of buffer

#### Transient analysis of buffer for Slew rate

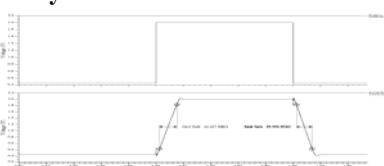


Fig. 5 Slew rate response of buffer

#### IV. MEASURED RESULTS

Open Loop gain: 83.95 db, Gain bandwidth: 23.92 MHz

Phase Margin: 53 degree, SR+ = 22.22 V/ $\mu$ s, SR- = 24.44V/ $\mu$ s, Output Swing: 0.15 to 3.2V, Power dissipation: 9.13mW  
**Conversion Output of 2 bit Cascaded Pipeline ADC**

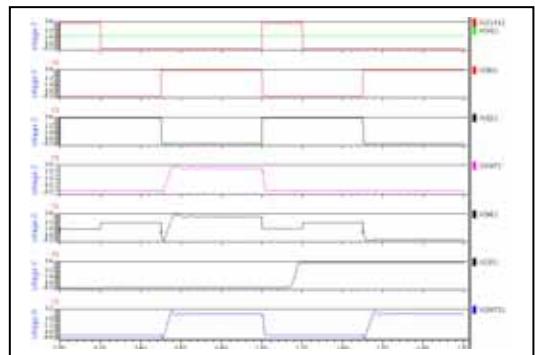


Fig.6 Output response of 2 bit cascaded Pipeline ADC without using switch

Fig.6 shows the output response without using Switch after Conversion of 1 bit. Hence when we Cascade, the Capacitor C1 and C2 comes into parallel this is default, to overcome this we have introduced a Switch. Fig.7 shows the Output response of 2 bit cascaded Pipelined ADC when we have introduced a buffer after that output characteristic response has improved.

#### Output response of 2 bit cascaded ADC with buffer

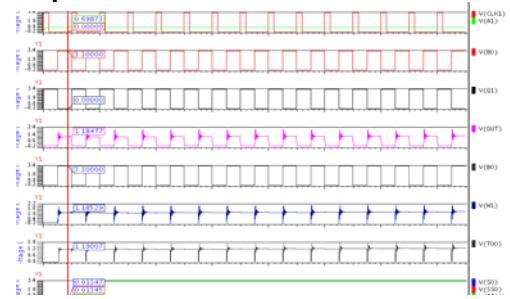


Fig.7 Output response of 2 bit cascaded ADC with buffer

## Output response for 12 bit Cascaded Pipelined A/D Converter with input sine wave

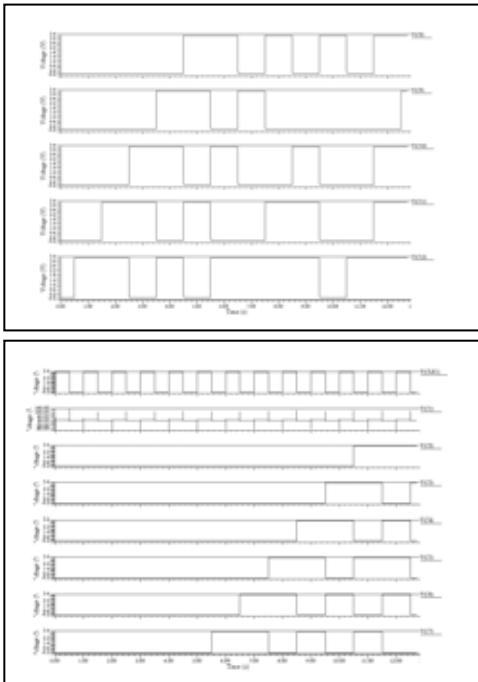


Fig.8 12 bit digital Output response for input sine wave

## Residue response of 12 bit Cascaded input sine wave

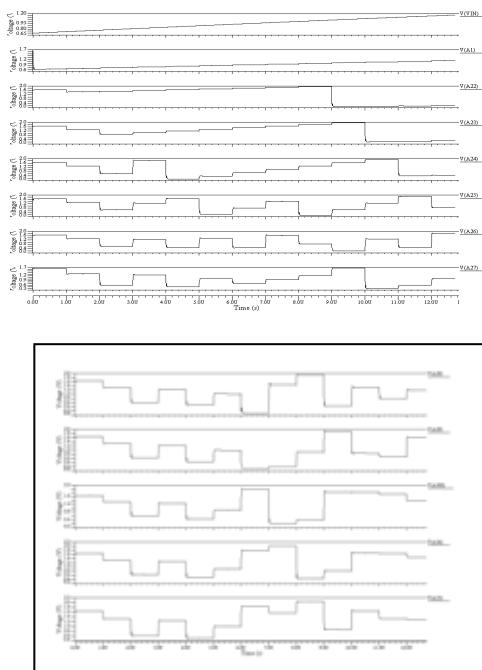


Fig.9 12 bit ADC residue response for input sine wave

## V. CONCLUSION

In this paper 12-bit cascaded Pipelined ADC structure and its circuit implementation has been discussed. When above technique was implemented the circuit was greatly improved and the resolution of the ADC reached 12-bit. The circuit was processed in 0.35μm standard CMOS process technology, and

the advantage of the circuit structure has been verified. In this work we are cascading 12 bit pipelined architecture with the use of high gain buffer. When we are not using this technique there was degradation. The use of gain stage cascading for multiply by two functions makes it less power consumption architecture compare to other A/D converters. It has been successfully cascaded to implement the complete 12 bit Pipelined A/D architecture.

## VI. REFERENCES

- [1]. P. E. Allen and D. R. Hollberg, "CMOS Analog Circuit Design", Oxford University Press, second ed., 2002".
- [2]. R. J. Baker, H. W. Li, and D.E. Boyce", "CMOS Circuit Design, Layout, and Simulation". Institute of Electrical and Electronic Engineer, Inc. 1998".
- [3]. D.W.Cline and P.R. Gray, IEEE JSSC, 31, p. 94(1996).
- [4]. Karanicolas, H.S. Lee, IEEE JSSC, 28, p.1207 (1993)
- [5]. Razavi, Design of Analog "CMOS Integrated Circuits". McGraw -Hill, first edition".
- [6]. R.Van de Plassch, "CMOS Integrated Analog-to-Digital and Digital-to-Analog Converters. Kluwer Academic Publishers, second ed., 2003".
- [7]. L. Coban and P. E. Allen, "A 1.5v, 1mW audio ΔΣ modulator with 98 dB dynamic range," Proceedings of the International Solid-State Conference, pp. 50–51, 1999".
- [8]. K. L. Lee, "Low Distortion Switched Capacitor Filters", Electron. Res. Lab. Memo M86/12, University of California, Berkeley, 1986".
- [9]. D. Johns and K. Martin, "Analog Integrated Circuit Design. John Wiley & Sons, Inc. 1997".
- [10]. Texas Instruments, Data Converter Selection Guide, 2004.
- [11]. P. R. Gray and R. G. Meyer, "Analysis and Design of Analog Integrated Circuits." John Wiley & Sons", Inc., third edition ed., 1993".
- [12]. B. Razavi and B. A. Wooley "Design techniques for high-speed, high-resolution comparators," IEEE Journal of Solid-State Circuits, vol. 27, no. 12, pp. 1916–1926, 1992".

# Computer Vision in an Embedded System

Vishal S.Vora- Pg Student, PG Student-Yagnesh N. Makawana, PG Student-A. M. Kothari, Prof.- Mr.A. C. Suthar  
Deptt of E&C, CCET, Wadhwani, Gujrat, India. [vsvora@aits.edu.in](mailto:vsvora@aits.edu.in)

**Abstract-** In this paper describes how the embedded systems community with an insight into computer vision challenges and techniques, basis on algorithmic and hardware specific considerations when “outsourcing” computation to a DSP, FPGA, or onto an embedded platform, and provides guidelines for how to best improve the runtime performance of computer vision applications. Because of Embedded computers and computer vision systems are two of the largest growths markets in the industry “Systems on a chip” is more popular because of the power, cost and space-saving combination of previously external chips onto the same die, including analog-to-digital converters, bus interfaces, I/O controllers, and even analog components. By using these integrated processing capabilities, computer vision has become a viable and preferable solution to many legacy and novel problem settings in industry, military, and consumer markets.

**Keywords—Digital Signal Processing, FPGA, System On Chip, Embedded Systems.**

## I. INTRODUCTION

Basically an embedded system is a set of digital and possibly analog circuitry that has a precisely defined purpose. It has one or multiple processors that serve a dedicated purpose for one particular system, optimized for a particular function to achieve shorter startup times, higher processing speed, greater reliability, lower cost, lower power consumption and/or some other property that a general-purpose computing system does not fulfill. The software is often called firmware, emphasizing its inseparability from the hardware. In an embedded system core part is microprocessor, usually surrounded by memory chips, input/output controllers, co-processors, analog components, and any imaginable electronic component.

Embedded system are particularly well suited to process streams of data at high speeds with relatively small software programs, such as video processing in television sets and DVD players, real-time control tasks in cars and airplanes, and cell phone signal processing, but also in microwave ovens, digital parking meters and pace makers. For vision, A *smart camera* is a combination of an imaging device and a computational unit whose combined output is a processed video stream, such as the location and identity of a person. However, integration and miniaturization make it increasingly feasible to develop smart cameras with much smaller form factors, down to single-chip devices that integrate processing power right on the imaging chip.

## II. HARDWARE

For general-purpose to system-specific chips various hardware are useful, some of hardware describe here with its main characteristics.

### A. Digital Signal Processor

A digital signal processor is similar to a general-purpose processor in many aspects. It has fixed logic, that is, the connections between logic gates can not be changed.

It provides a fixed instruction set (ISA) to the programmer, and it expects a program in this ISA which it will then execute in a sequential manner. Most DSP ISAs exhibit a similar structure as GPP ISAs, complete with arithmetic and logic instructions, memory access, registers, control flow. DSP's instruction set is optimized for matrix operations, particularly multiplication and accumulation traditionally in fixed-point arithmetic, but increasing also for double-precision floating point arithmetic. DSPs exhibit deep pipelining and thus expect a very linear program flow without frequent conditional jumps. They provide for SIMD instructions, assuming a large amount of data that has to be processed by the same, relatively simple, mathematical program. Many DSPs are VLIW architectures, the types of instructions that are allowed together within one VLIW will be executed in parallel depends on the function units that can operate in parallel.

DSP programs are relatively small programs with few branch and control instructions, as opposed to entire operating systems running on general purpose CPUs. Since DSPs usually execute small programs on huge amounts or endless streams of data, these two pieces of information are stored in separate memory blocks, often accessible through separate buses, many DSPs provide on-chip ROM for program storage, and a small but efficient RAM hierarchy for data storage. An embedded system also includes a separate non-volatile memory chip such as an EEPROM or flash memory.

DSPs lack just a few operations, mostly operating-specific instructions, but they can perform them faster, they need less power, they dissipate less heat, they have short start-up times, they can operate in a larger temperature range, and they are less expensive because the chip contains only the necessary components.

### B. Field Programmable Gate Array

A Field Programmable Gate Array, or FPGA, is a semiconductor in which the actual logic can be modified to the application builder's needs. The chip is a relatively inexpensive, off-the-shelf device that can be programmed in the “field” and not the semiconductor fabrication. It is important to note the difference in software programming and logic programming, or logic design as it is usually called: a software program always needs to run on some microcontroller with an appropriate instruction set architecture (ISA), whereas a logic program *is* the microcontroller.

Some of the modern FPGAs integrate platform-or hard multi-purpose processors on the logic such as a PowerPC, ARM, or DSP architecture. Other common hard and soft modules include multipliers, interface logic, and memory blocks. Some more complex FPGAs can modify and reprogram their general-purpose logic blocks even on the fly, that is, while another part of the chip keeps running.

The logic design determines the FPGA's functionality; there are three types of FGPAs: antifuse, SRAM, and FLASH. Antifuse chips are not re-programmable. FLASH (EPROM) is also non-volatile, meaning that the logic design stays on the chip through power cycles. It can be erased and re-programmed many times. SRAM programming on the other hand is volatile – it has to be programmed at power on.

### C. Application Specific Integrated Circuits

An Application-Specific Integrated Circuit (ASIC) is a chip that is designed and optimized for one particular application. The logic is customized to include only those components that are necessary to perform its task. Even though modules are reused from ASIC to ASIC just like FPGA modules, a large amount of design and implementation work goes into every ASIC. Their long production cycle, their immensely high one-time cost, and their limited benefits. Contributing to the high cost, they need to be resrun if the design changes just slightly, costing months and usually hundreds of thousands of dollars.

### D. System on Chip

A System on Chip (SoC) contains all essential components of an embedded system on a single chip. Sometimes only refers to the digital components and analog components. Most SoCs have a GPP such as an ARM, MIPS, PowerPC, or an x86-based core at their heart, supplemented by a DSP. Standard microcontroller components (busses, clocks, memory), typical integrated components like, Ethernet MAC, PCMCIA USB 1.0 and 2.0 controller, Bluetooth, RS232 UART, IrDA, IEEE 1394 (FireWire) controller, display interface, flash memory interfaces ADCs and DACs Systems on a chip are of particular interest to highly-integrated devices like mobile phones, portable DVD and mp3 players, set-top-boxes and cable modems. Many SoCs have dedicated circuitry for video processing, which usually means hardware support for decoding from (and sometimes for encoding into) the various video formats, including MPEG2, MPEG4, and H.263.

### E. Smart Camera Chip and Boards

Processors, SoCs, and chip sets that directly support higher-level computer vision tasks come in various flavors, from CMOS image capture chips that include one or multiple small processors to frame grabber PCI boards with multiple full-scale CPUs.

## III. SOFTWARE AND PROGRAMMING TOOLS

Software tools that is necessary or helpful to implement those programs and to transfer them to the embedded system.

### A. Language and Operating Systems for DSP

DSPs are usually programmed in C at first, followed by machine code optimization for critical parts. A DSP rarely just executes one tiny program on an endless stream of rather uniform data, but instead has to perform some general tasks occasionally. Thus, it is usually controlled by a DSP

operating system. A large number of companies offer an even larger number of them, frequently classified as a real-time OS. Linux is a common choice due to its flexibility, particularly on Systems-on-Chips. Following are some of your choices for operating for DSPs and/or SoCs. One of the main characteristics of these OSs is their small footprint, typically only one to tens of MB.

### B. FPGA Logic Design

Determining the function of each of the logic elements on a Field-Programmable Gate Array as well as their interplay and connectivity is called the logic design or digital design.

1) Design Flow: The primary block of an FPGA is called a logic element, which usually functions as a lookup-table and contains a fancy flip-flop and a few logic gates. The FPGA needs to know how to arrange these. The digital design needs to be mapped to logic elements, this is called technology mapping. The netlist is the actual file that gets mapped and routed to a specific FPGA.

The (logic) synthesis translates design into a so-called netlist. This is usually a heavily optimizing synthesis that not only optimizes the HDL with traditional compiler means, but also makes use of hard macros, chip specifics etc. The netlist is usually in the Electronic Design

Interchange Format, or EDIF. This entire process is also called FPGA technology mapping.

Next, the netlist needs to be written to the FPGA. This process is called place-and-route. First, the netlist's component references (registers, logic, macros) are assigned to FPGA locations. This is obviously a manufacturer-dependent process as FPGA layouts vary considerably. The result is a FPGA bitmap, which is sent to the FPGA via a number of different means, the most popular being JTAG.

2) Hardware Description Language: A Hardware Description Languages, or HDL [1], describes the behavior of a chip. Many companies offer a product that combines all necessary and a host of extra tools for FPGA programming into one software package. This is referred to as electronic design automation.

One essential component that EDA tools provide are hardware libraries or hardware macros, containing the building blocks for common functionalities like memory, signals, multipliers, and all the way up to entire DSP and GPP soft cores, which are to be created out of generic FPGA logic elements.

3) JTAG and Boundary Scan: The interface to generate tests and access test results on printed circuit boards was standardized by the Joint Test Action Group and has since itself become known as JTAG. The type of testing that JTAG was intended to perform is actually called boundary scan. It works by accessing the input and output lines of a chip or circuit board and storing their values in a boundary scan register. To facilitate this, a JTAG-compliant chip or board has additional logic around a chip's actual core. This logic observes input values to the core and writes them serially to a dedicated test access port, or TAP, where they can be compared to the input signals that were applied to the chip's

physical connectors. The TAP is a 5-pin serial connection through which all communication with the boundary logic is sequenced to or from the debugging facility, usually a PC. JTAG is frequently used as the method of choice to route the actual program (the netlist(bitmap) to an FPGA. While a convenient and wide-spread access method, JTAG is not suited for high-bandwidth debugging.

#### IV. ALGORITHMS

We present some of the most important real-time algorithms from different fields that embedded vision relies on: data analysis, digital signal and image processing, low-level computer vision, and machine learning.

##### F. Tensor Product

For a very important class of algorithms there is a theoretical tool that can be of use in comparing different algorithms as they map on different processor architectures [2]. This tool applies to digital filtering and transforms with a highly recursive structure. Important examples are [3]:

- 1) Linear convolution, e.g., FIR filtering, correlation, and projections in PCA (Principal Component Analysis)
- 2) Discrete and Fast Fourier Transform
- 3) Walsh-Hadamard Transform
- 4) Discrete Hartley Transform
- 5) Discrete Cosine Transform
- 6) Strassen Matrix Multiplication

To investigate these algorithms, the required computation is first represented using matrix notation. This algorithm efficiently implemented on a SIMD architecture. This method does not take into account non-deterministic effects of cache memory and super scalar issue parallelism. Same formulas can be used to determine many mathematically algorithm which are suitable for a particular processor architecture.

##### G. Sorting Algorithm

A common task in data analysis is sorting of data in numerical order. The fastest sorting algorithm is Sir C. A. R. Hoare's Quicksort algorithm [4]. A slightly slower J. W. J. Williams' Heapsort algorithm [5] but in-place sorting and has better worst-case asymptotics. In Table 1 shows some more information on asymptotic complexity.

##### H. Golden Search Algorithm

It is a simple but very effective algorithm to find a maximum of a function. Basic assumption is that getting the value of the function is expensive because it involves extensive measurements or calculations [6]. By using some application-specific method It is a one, and only one, maximum in the interval given to the algorithm.

##### I. Kalman filtering

A Kalman filter [7] is an estimator used to estimate states of dynamic systems from noisy measurements. This method is attractive for the recursive solution and suitable for real-time implementation on a digital computer. The fact

that the solution is recursive means that at each time instant updating the previous estimate forms the estimate using the latest measurement. If there was no recursive solution, we would have to calculate the current estimate using all measurements. Other, more complicated formulations of Kalman filter allow colored noise, others can account for unknown system model parameters, and some can deal with nonlinearities. In recent years, a more general framework called particle filtering has been used to extend applicability of Kalman's approach to non-Gaussian stochastic processes.

##### J. Fast Fourier Transform

Fast Fourier Transform (FFT) is a common name for a number of  $O(n\log n)$  algorithms used to evaluate the Discrete Fourier Transform (DFT) of  $n$  data points. Several ways the FFT can be used to speed up seemingly unrelated operations, such as convolution, correlation, and morphological operations [8]. Some of the other fast algorithms for convolution that are non-FFT based like fast convolution and correlation, windowing and data transfer, fast morphing.

##### K. PCA Object Detection and Recognition

Principal Component Analysis (PCA), also known as eigenpictures or eigenfaces [9, 10], is a popular computer vision technique for object detection and classification, especially in the context of faces. The most common PCA classification scheme is based on the reconstruction error. If PCA is implemented using a set of orthonormal eigenvectors, this implies use of the floating-point precision. On a TI DM642 processor, this technique has reduced the execution time of PCA classification by a factor of seventeen. The execution time is further reduced when utilizing the SIMD operations available on the chip. These operations utilize 32-bit multipliers and adders to do four simultaneous 8-bit operations. To estimate the numerical error introduced by fixed-point representation the quantization error as uniform noise with variance  $12222XmBe = \sigma$  (1) where  $B+1$  is the number of bits in the representation. As a result of going from a 32-bit floating-point representation to an 8-bit fixed-point representation, the SNR change due to quantization error is  $\Delta(\text{SNR}) = 20\log_2 B0 - B1 \approx -144\text{dB}$ , where  $B0 = 8\text{bit}$  and  $B1 = 32\text{bit}$ . By itself this may seem like a large degradation, but we are still left with around 48 dB of SNR. Object detection is a related but different problem of finding candidate images to be used in recognition. PCA can be used for detection, when the entire scene is scanned by sliding the PCA classifier in search of objects such as faces. In this case, projections turn into correlations. As we discussed earlier in this section, fast correlation algorithms exist, similar to fast convolution algorithms, based on FFT or other approaches.

##### L. Suitability of Vision Algorithm

Some image processing and computer vision methods are better suited to execution on a DSP or FPGA than others, for different reasons. The theoretical, potential speedup of common algorithms to be placed in a DSP or FPGA, and try to generalize and look for the reasons. In general, these are

properties that make an algorithm a good candidate to be executed in special hardware:

- 1) Streams/sequential data access, as opposed to random access
- 2) Multiple, largely independent streams of data
- 3) High data rates
- 4) Data packet size is fixed, or at least bounded by a small number
- 5) Stream computations can be broken up into pipeline stages, i.e. the same (possibly parameterized) computations independent of prior stage's output
- 6) Fixed-precision values (integer or fix-point fractions)
- 7) Analyzable at instruction and module level, that is, little between-stream communication necessary

Consider what really constitutes digital video data. Video data is dense 2D data with each element of a constant size. Most frequently, it is streamed along a communication bus in a line-by-line manner, or interlaced in the case of NTSC or PAL video sources. Traditionally, data is buffered and processed whenever the program is the proper instruction. But ideally suited to processing streamed data are “online algorithms” which process data as they arrive. This online or data-driven execution is advantageous for low latency and low buffering requirements. However, it might result in unnecessary computations as it is not usually known a-priori what data are needed in the end. A demand-driven approach avoids this, but incurs higher latency due to its reverse dependency chain.

Computer vision methods commonly entail linear algebra, particularly for non-image data such as feature vectors for classification. Thus, matrix operations are common, including matrix multiplication, calculating the inverse and transpose, Gaussian elimination, residual minimization for linear systems, singular value decomposition, and eigenvector and eigen value Calculations.

## V. CONCLUSION

Computer vision on embedded platforms and building smart cameras is an exciting field with tremendous growth prospects. In this paper on hardware and software that are most often found in embedded systems, with a focus on architectures and tools that are built for computer vision applications. With a particularly difficult issue real-time algorithms for common computer vision operations. These are great times to witness exceedingly rapid progress in a field that had in the past had difficulties living up to overly great expectations. Yet now computer vision has made inroads in many applications, from games to surveillance, from smart airbag deployment to automatic parking controllers for automobiles. Computer vision has the opportunity to sense beyond the confines of a computer, and embedded systems have the power to make the computer itself mobile. In combination, these technologies harbor possibilities that can only be imagined as the first steps are taken into this exciting field!

## VI. REFERENCES

- [1]. J. Bhasker. *A VHDL Primer*. Prentice Hall, 3 edition, 1999.
- [2]. J. Granata et al. The tensor product: A mathematical programming language for FFTs and other fast DSP operations. *IEEE Signal Proc Mag*, pp 40–48, 1992.
- [3]. J. Granata et al. Recursive fast algorithm and the role of the tensor product. *IEEE Trans Signal Proc*, pp 2921–2930, 1992.
- [4]. C A R Hoare. Quicksort. *Computer Journal*, pp 10–15, 1962.
- [5]. J W J Williams. Algorithm 232 (heapsort). *Comm ACM*, pp 347–348, 1964.
- [6]. EKPChongandSHZak. *An Introduction to Optimization*. Wiley, 2001.
- [7]. R E Kalman. A new approach to linear filtering and prediction problems. *ASME J of Basic Engineering*, pp 34–45, 1960.
- [8]. J W Cooley. How the FFT gained acceptance. *IEEE Signal Proc Mag*, pp 10–13, 1992.
- [9]. L Sirovich and M Kirby. Low-dimensional procedure for the characterization of human faces. *J Opt Soc Am A*, pp 586–591, 1987.
- [10]. M Turk and A Pentland, Face recognition using eigenfaces. *Proc CVPR*, 1991.

# A Computational Model for Cell Survival and Fabrication of Transistor

Shruti Jain<sup>1</sup>, Pradeep.K. Naik<sup>2</sup>, C.C. Tripathi<sup>3</sup>

<sup>1</sup>Department of Electronics and Communication Engineering

<sup>2</sup>Department of Biotechnology & Bioinformatics

Jaypee University of Information Technology, Solan-173215, India

<sup>3</sup>University Institute of Engineering Technology, Kurukshetra University, Kurukshetra-132119 India

E-mail<sup>1</sup>: [shrutijain@ieee.org](mailto:shrutijain@ieee.org), [jain.shruti15@gmail.com](mailto:jain.shruti15@gmail.com)

**Abstract-**A well-structured and controlled design methodology, along with a supporting hierarchical design system, has been developed to optimally support the development effort on several programs requiring gate array and semi custom Very Large Scale Integration (VLSI) design. In this paper, we will present an application of VLSI in System Biology. This work examines signaling networks that control the survival decision treated with combinations of three primary signals the pro death cytokine, tumor necrosis factor- $\alpha$  (TNF), and the pro survival growth factors, epidermal growth factor (EGF) and insulin. We have made model by taking three inputs, than made the truth table, Boolean equation and than implement the equation using transistors. Next, we have discussed the various steps for the fabrication of transistor.

**Keywords-** Truth table, Karnaugh Map, Boolean equation, Fabrication

## I. INTRODUCTION

The complexity of VLSIs being designed and used today makes the manual approach to design impractical. Design automation is the order of the day. With the rapid technological developments in the last two decades, the status of VLSI technology is characterized by the following

- A steady increase in the size and hence the functionality of the ICs.
- A steady reduction in feature size and hence increase in the speed of operation as well as gate or transistor density.
- A steady improvement in the predictability of circuit behavior.
- A steady increase in the variety and size of software tools for VLSI design.

The above developments have resulted in a proliferation of approaches to VLSI design. We briefly describe the procedure of automated design flow. The aim is more to bring out the role of a Hardware Description Language (HDL) in the design process. An abstraction based model is the basis of "the automated design.

### 1.1 Abstraction Model

The model divides the whole design cycle into various domains (shown in Figure 1). With such an abstraction through a division process, the design is carried out in different layers. The designer at one layer can function without bothering about the layers above or below. The thick horizontal lines separating the layers in the figure signify the compartmentalization. As an example, let us consider design at the gate level. The circuit to be designed would be described in terms of truth tables and

state tables. With these as available inputs, he has to express them as Boolean logic equations and realize them in terms of gates and flip-flops. In turn, these form the inputs to the layer immediately below. Compartmentalization of the approach to design in the manner described here is the essence of abstraction; it is the basis for development and use of CAD tools in VLSI design at various levels.

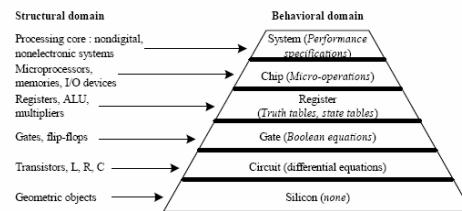


Figure 1: Design domains and Level of Abstraction

The design methods at different levels use the respective aids such as Boolean equations, truth tables, state transition table, etc. But the aids play only a small role in the process. To complete a design, one may have to switch from one tool to another, raising the issues of tool compatibility and learning new environments.

## II. SYSTEM

Very Large Scale Integration (VLSI) is the field, which involves packing more and more logic devices into smaller and smaller areas. Our aim is to use VLSI in System Biology. Computational systems biology addresses questions fundamental to our understanding of life, yet progress here will lead to practical innovations in medicine, drug discovery and engineering. Substantial progresses over the past three decades in biochemistry, molecular biology and cell physiology, coupled with emerging high throughput techniques for detecting protein-protein interaction, have ushered in a new era in signal transduction research. Cell signaling pathways interact with one another to form networks. Such networks are complex in their organization and exhibit emergent properties such as bistability and ultrasensitivity [1]. To understand complex biological systems requires the integration of experimental and computational research - in other words systems biology approach. Computational biology, through pragmatic modeling and theoretical exploration, provides a powerful foundation from which to address critical scientific questions head-on. Studies of signaling pathways have traditionally focused on delineating immediate upstream and

down stream interactions, and then organizing these interactions into linear cascades that relay and regulate information from cell surface receptors to cellular effectors such as metabolic enzymes, channels or transcription factors. This work examines signaling networks that control the survival decision treated with combinations of three primary signals [2, 3] the pro death cytokine, *tumor necrosis factor-a* (TNF) [4, 5], and the pro survival growth factors, *epidermal growth factor* (EGF) [6, 7] and insulin [8, 9, 10]. The system output is typically a phenotypic readout (death or survival); however, it can also be determined by measuring “early” signals that perfectly correlate with the death/ survival output. Examples of such early signals include phosphatidylserine exposure, membrane permeability, nuclear fragmentation and caspase substrate cleavage. We have implemented the signaling system heading by three input signals such as TNF, EGF and insulin. The block diagram of the signaling system that was modeled is shown in Figure 2.

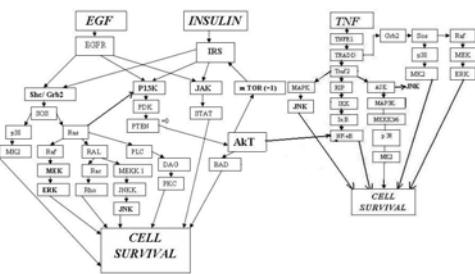


Figure 2 : Model for Cell Survival.

Above we had studied relating how TNF, EGF and Insulin works, its pathways and explain each possible path for that. Based on pathways we had made truth tables for every possible path for cell survival. Figure 3 shows one of the pathway's truth table. In output, '1' signifies cell survival and '0' signifies cell death.

p 38	MK2	Output
0	0	0
0	1	0
1	0	0
1	1	1

Figure 3 : Truth table for p38 and MK2.

Than we realize the truth tables by Karnaugh Map (K-Map) and get the Boolean expression for its individual possible paths. Figure 4 shows the K map for the truth table shown in Figure 3.

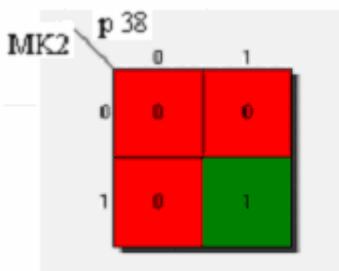


Figure 4 : Karnaugh map.

By solving the K map we get Boolean expression as

## Cell Survival: p38.MK2

The above equation shows the ANDing of two inputs. AND means if both input are present then it will give '1' otherwise '0'. The equation can implemented by using diodes, bipolar junction transistors (BJT), Complementary metal oxide semiconductor field effect transistor (CMOS) or Integrated circuits (IC).

The circuit diagram for the above equation using BJT is shown in Figure 5.

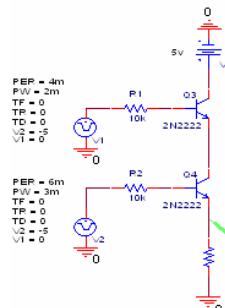


Figure 5 : Realization of equation using transistors.

Now our aim is to fabricate a transistor. In the next section, we will discuss the steps for the fabrication of a transistor.

### III. STEPS FOR FABRICATION OF A TRANSISTOR

#### 3.1 Mask Making

For fabrication of any semiconductor device the first essential step is to design the mask. For this we use Auto-CAD (computer aided design), which has many options and tools. After designing the mask using the software the printed format is placed under the camera reduction system to get the reduced pattern on photographic negative films, which are then used as a soft mask for lithography process. We have successfully designed and fabricated a camera reduction system using SLR camera, light source and other camera mounting assemblies. The in-house built camera reduction system has reduction ratio of 10 to 13.6X using calibrated rotational objective lenses mounting assembly of the camera. With this reduction camera system, we have successfully designed and fabricated mask of feature size as small as 30 micron using 1200 DPI laser printer. The feature size-limiting factor is due to the printer resolution used for taking the hard copy of AUTOCAD design microstructure.

Further fine geometries mask can be prepared using high-resolution laser/ line printer used for slide making in printing media industry. Prepared soft masks (high resolution photo films) are then used for preparing the hard mask on chrome coated glass plates using photolithographic technique, developed indigenously.

#### 3.2 CLEANING

The next step in fabrication is cleaning of the silicon wafer at the following specification p-type, resistivity 5 to 10  $\Omega$ , orientation  $<100>$ , thickness 40  $\mu\text{m}$ . Initial cleaning is necessary so that no dust particle is left on the wafer surface otherwise actual results can't be obtained. In order to obtain the

correct results we carefully do the wafer cleaning whenever required. Wafer cleaning can be done by following procedure:-  
 1. Sulphuric acid and hydrogen peroxide (3:1 i.e. 3 portions of sulphuric acid and 1 portion of hydrogen peroxide) treatment for 10 minutes.

2. Hot water treatment for 10 minutes.

3. Methanol treatment so that wafers get dries easily.

Note that wafer should not be placed in open air as it can react with air to form oxide layer and also impurities stick to the wafer. So after cleaning wafers are to be put in methanol till it is not taken for some process.

### 3.3 OXIDATION

Oxidation is a process during which a chemical reaction between the oxidants and silicon atoms takes place and this produces a layer of silicon oxide on the wafer surface. While starting oxidation process first step is cleaning the oxidation furnace tube. This tube is cleaned in chromic acid [11, 12]. After this furnace is switched ON. The temperature in PID controller is set to 1050°C so as to get same as working temperature inside the furnace. Now fill the water bubbler upto 75% with distilled water. Then attach the bubbler to the furnace and also switch on the bubbler to raise its temperature upto 100°C so that seam formation takes place. Then pass nitrogen gas into the furnace at a pressure of 1 liter per second (approx) for half an hour to clean the furnace. After the desired temperature is obtained the cleaned wafers are loaded in the furnace for oxidation (5 minutes on the mouth of furnace and then inserted slowly towards the center of the furnace). After half an hour the supply of nitrogen is stopped and then oxygen at a pressure of 0.5 liter per minute for 30 minutes. This dry oxidation is done to grow good quality oxide film as base layer. After this oxygen is supplied to furnace after passing it through bubbler (as shown in Figure 6) to continue the oxidation for desired time and the procedure is known as wet oxidation.

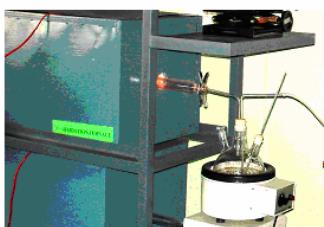


Figure 6: Apparatus showing the set up for dry/wet/dry oxidation.

After the required time again dry oxidation is done for half an hour before to stop the oxidation process. Now the temperature of the furnace is reduced at the rate of 10°C/ minute using PID controller setting upto 600°C. Now the furnace may be switched off to cool down to room temperature. The time of oxidation is decided according to the oxide layer thickness requirement.

### 3.4 LITHOGRAPHY

Lithography is the important step in fabrication of any semiconductor device. This lithography [11] is done to transfer the desired images on the wafer. Out of various lithography

techniques. Photolithography is done frequently because of its simplicity and cost effectiveness [12]  
 Various steps of photolithography are:

#### 3.4.1 Photo resist application (spinning)

The first step in photolithography is the deposition of photo resist layer on the oxidized wafer. Generally two types of photo resist are used:

1. Negative photo resist and
2. Positive photo resist.

The photo resist application is done on spin coater (shown in Figure 7). Spin coater is having a vacuum chuck to hold the wafer during spinning. The speed of rotation of wafer holding chuck and time depends upon the thickness requirement of photo resist layer.



Figure 7: Spin coater

#### 3.4.2 Pre-bake

After forming a uniform layer of photo resist on the wafer the coated wafers is placed in the oven at about 90° C for about 15-30 minutes to drive off solvents in the photo resist and to harden it into a semisolid film. Also adherence to the wafer is improved by pre-baking.

#### 3.4.3 Alignment and exposure

After baking the wafer it is ready for the exposure to U.V. radiations. For doing the exposure, mask aligner as shown in Figure 8 is used. This mask aligner is having a U.V. light source, a mask holder, wafer holding plate, an alignment system to align the mask to the wafer and a microscope to check the alignment mask pattern to the wafer. After this U.V. lamp is switched on to start exposure process so as to transfer the image pattern of mask to PR coated wafer. The exposure is carried out for a period of 1-2 min depending upon the PR film thickness.

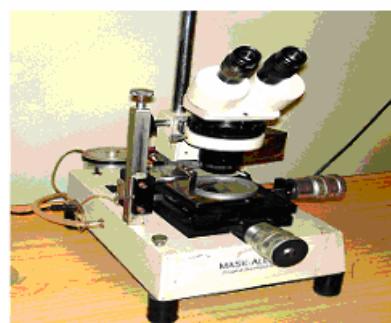


Figure 8 : Mask aligner

### 3.4.4 Developing

After exposure, the wafer is dipped in the developing solution to remove the unwanted photo resist from the wafer. Developer solutions are different for both the photo resists.

#### FOR POSITIVE PHOTO RESIST

- 10-12 tinches(tablets) of potassium hydroxide (KOH) in 100ml water ( $H_2O$ ) is mixed & then filtered.
- 200ml of D.I. water.

#### FOR NEGATIVE PHOTORESIST

Any of the following combinations can be used:

- Xylene followed by n-butyl acetate followed by acetone.
- Xylene followed by mixture of xylene and isopropanol (1:2) followed by acetone.
- Trichloroethylene followed by n-butyl acetate followed by acetone.
- Trichloroethylene followed by mixture of xylene and isopropanol followed by acetone.
- Trichloroethylene followed by mixture of trichloroethylene and isopropanol followed by acetone.

Here we can opt for any combination but the first one is standardized for developing purpose. Initially, wafer is dipped in Xylene for 75 seconds. Then they are dipped in n- butyl acetate followed by acetone dip 10 seconds in each.

### 3.4.5 Post bake

After development and rinsing, the wafers are usually given a post bake in an oven at a temperature of about 120°C for about 30-60 minutes to toughen further the remaining resist on the wafer. This is to make it adhere better to the wafer and to make it more resistant to the hydrofluoric acid (HF) solution used for etching of the silicon dioxide.

### 3.4.6 Oxide etching

The remaining resist is hardened and acts as a convenient mask through which the oxide layer can be etched away to expose areas of semiconductor underneath. These exposed areas are ready for impurity diffusion.

For etching of oxide, the wafers are immersed in or sprayed with hydro fluoride acid solution. This solution is usually a diluted solution of typically 10:1,  $H_2O$ : HF, or more often a 10:1,  $NH_4F$  (ammonium fluoride): HF solution. We use 40% HF to form buffer solution. Buffer solution is prepared using HF (40%) +  $NH_4F$  (40%) in ratio 1 to 6 by volume. The  $NH_4F$  is prepared by mixing 100 gm of  $NH_4F$  + 250 ml of DI water.

The duration of oxide etching should be carefully controlled so that all of the oxide present only in the photo resist window is removed. If etching is excessively prolonged, it will result in more undercutting underneath the photo resist and widening of the oxide opening beyond what is desired. Once the oxide is removed from window then etched surface becomes hydrophobic (water does not stick to the surface)

### 3.4.7 Photo resist stripping

Following oxide etching, the remaining resist is finally removed or stripped off with a mixture of sulphuric acid and hydrogen peroxide (3 to 1 ratio) and with the help of abrasion process. Finally step of washing and drying completes the required window in the oxide layer.

Negative photo resist are more difficult to remove. Positive photo resist can be easily removed in organic solvents like such as acetone.

### 3.5 CAVITY / DIAPHRAGM FORMATION

#### 3.5.1 Anisotropic Etching

After the desired oxide etching, the silicon wafer is etched in KOH solution (is effective only on <100> p-type silicon wafer) so as to form cavity. The resultant cavity is a V- shaped groove, or cut pits with tapered walls into silicon depending upon the window opening & etching time. Figure 9 shows the resultant cavity in the wafer.

Both oxide and nitride etch slowly in KOH. Oxide can be used as an etch mask for short periods in the KOH etch bath (i.e., for shallow grooves and pits). For long periods, nitride is a better etch mask as it etches more slowly in the KOH. The wafer now etched in KOH (30% at 70°C) so that we are able to get a deep cavity as shown in Figure 9. This depends upon the etching time & its oxide window through which etching is taking place.



Figure 9: KOH Etching

#### 3.5.2 Isotropic Etching

Silicon diaphragm (111) can also be made using isotropic etching method. In this method, initially thick wafers are thinned down using solution HF:  $HNO_3$ :  $CH_3COOH$  (3:5:3) solution (etch rate 15 micron/minute) at room temperature. This is fast etchant of silicon. KOH etching is relatively slow etching process therefore initial fast etching is required to thin down the wafer if the initial wafer thickness is very high. After this fast etching, when left out wafer thickness is of the order of 100 micron, slow etching is done to control the etch rate. In this, the left out wafer was etched in the mixture of HF+ $HNO_3$ + $CH_3COOH$  in ratio 1:3:5(etch rate 1-2 micron/ minute). By this method the fabricated diaphragm have relatively poor uniformity because of exothermic reaction. Using this method, a diaphragm as thin as 20 micron can be made by controlling the process parameter.

### 3.6 METALLIZATION

In the process of metallization aluminum contacts are taken on the p-layer by depositing the aluminum on the surface of the wafers & glass substrate and the unwanted aluminum is removed by using photolithography.

In metallization, after removal of oxide layer wafers are loaded in the metallization coating machine. The metal to be deposited is also loaded in the boat machine and the door of coating unit is closed to create the vacuum inside the chamber. When the required vacuum is achieved, the metal is deposited on both the wafer & glass by passing current through the aluminum so that it evaporates and gets deposited on the work pieces. The thickness of aluminum film is 0.4 to 0.5 $\mu$ m. Then by using photolithography process one metallic plate of condenser is defined on the glass & rest Al is etched by dipping in the solution of aluminum enchanter ( $H_3PO_4$ :  $CH_3COOH$ :  $H_2O$ ) as 16:4:2.

#### IV. CONCLUSION

We had successfully made computational model for cell survival using three inputs such as TNF, EGF and insulin. With that model we had made truth table, Boolean expression and logical circuit for each possible pathway. We then simulate the results of each path and then combine all the results and get result of TNF, EGF and Insulin for its survival using Bipolar junction transistor. Later we have discussed the various steps for fabrication of transistor. Our future work to implement the various steps for the fabrication of transistor practically.

#### V. REFERENCES

- [1] Gaudet Suzanne, Janes Kevin A., Albeck John G., Pace Emily A., Lauffenburger Douglas A, and Sorger Peter K. July 18, 2005 *A compendium of signals and responses triggered by prodeath and prosurvival cytokines* Manuscript M500158-MCP200.
- [2] Janes Kevin A, Albeck John G, Gaudet Suzanne, Sorger Peter K, Lauffenburger Douglas A, Yaffe Michael B. Dec.9, 2005 *A systems model of signaling identifies a molecular basis set for cytokine-induced apoptosis*; Science 310, 1646-1653.
- [3] Weixin Zhou 2006 *Stat3 in EGF receptor mediated fibroblast and human prostate cancer cell migration, invasion and apoptosis*, PhD thesis, University of Pittsburgh.
- [4] Brockhaus M, Schoenfeld HJ, Schlaeger EJ, Hunziker W, Lesslauer W, and Loetscher H (1990) Identification of two types of tumor necrosis factor receptors on human cell lines by monoclonal antibodies. Proc Natl Acad Sci USA 87, 3127-3131.
- [5] Thoma B, Grell M, Pfizenmaier K, and Scheurich P (1990) Identification of a 60-kD tumor necrosis factor (TNF) receptor as the major signal transducing component in TNF responses. J Exp Med 172, 1019-23.
- [6] Libermann TA , Razon TA., Bartal AD, Yarden Y., Schlessinger J and Soreq H 1984 *Expression of epidermal growth factor receptors in human brain tumors* Cancer Res. 44,753-760.
- [7] Normanno N, De Luca A, Bianco C, Strizzi L, Mancino M, Maiello MR,, Carotenuto A, De Feo G, Caponiqro F, Salomon DS. 2006 *Epidermal growth factor receptor (EGFR) signaling in cancer* Gene 366, 2–16.
- [8] Lizcano J. M. Alessi D. R. 2002 *The insulin signalling pathway*. Curr Biol. 12, 236-238.
- [9] Morris F. White 1997 *The insulin signaling system and the IRS proteins* Diabetologia 40, S2-S17
- [10] Morris F. White 2003 *Insulin Signaling in Health and Disease* Science 302, 1710–1711
- [11] Botkar, K.R. *Integrated circuits*
- [12] Gandhi Sourabh K, VLSI Fabrication Principles.

# Deployable Outdoor Surveillance System Capable of Video Transmission

Manish vaish, Jaskirat singh, Harsimran singh and Rabindranath Bera

Sikkim Manipal Institute of Technology, Sikkim manipal University

Majitar, rangpo, East Sikkim, 737132, email: manishvaish\_2007@rediffmail.com

**Abstract-The system that we are proposing is a digital one, where we are re-utilizing the Concepts of digital communication techniques like WCDMA and Bluetooth for ranging and imaging. The bandwidth used is of 5 MHz Any man sized object can be detected in the radius of 2 Kilometers. Modulation techniques used here are:-**

- DSSS (Direct Sequence Spread Spectrum)
  - Carrier frequency stepping (Somewhat like FH-SS)
- The DSSS is used in 3-G mobile technology, whereas FHSS has its use in Bluetooth technology, which we implement here for imaging Purpose using 80-ary GFSK signaling scheme. At the receiver after the RF demodulation, 1D-IFFT of the signal is done which gives us a peak, telling us the size and distance of the object.

## I. INTRODUCTION

Surveillance is the process of monitoring the behavior of people, objects or processes within systems .systems for security or social control. The all-seeing "eye in the sky" is a general icon of surveillance. Surveillance can be a useful tool for law enforcement and security. The word surveillance is commonly used to describe observation from a distance by means of electronic equipment or other technological means i.e. Close circuit camera. It is an important tool for the police, intelligence agencies and other public authorities,

### A. Some examples of surveillance systems:

- A mobile surveillance may be made on foot or by vehicle. It is conducted when persons being observed move.
- During Loose surveillance, subjects need not be kept under constant observation. It is used when a general impression of the subject's habits and associates are required.
- In close surveillance subjects are kept under constant observation continuously,

Directed surveillance is a type of covert surveillance where police, intelligence agencies and other public authorities follow an individual in public and record their movements.

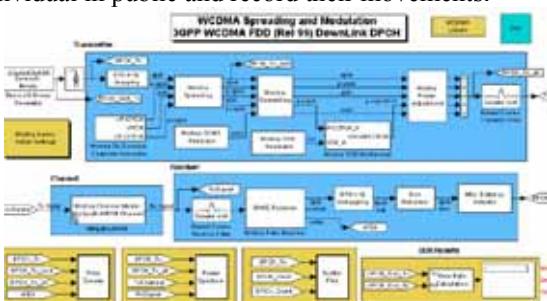


Fig.1 The complete simulink model of 3G WCDMA model

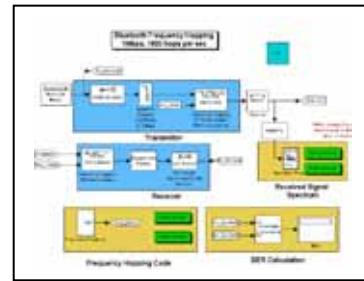


Fig.2 The complete simulink model of BLUETOOTH

Our proposed system will be digital. Here the concepts of WCDMA and Bluetooth (refer to the Fig.1 and 2) are used to do ranging and imaging of the object whose surveillance is being done. Basically the cell area in the case of WCDMA is about 2 Kilometers so this system will be able to do the surveillance of any man sized object which is in the radius of 2 Kilometers from the system. Now obviously the bandwidth of the system will be 5 MHz as in the case of WCDMA. But one thing is going to be sure that the whole tracking process will be secure. Now why this whole process is going to be secure? There is only one reason to this; it is because of one of the three modulation techniques that the WCDMA is using, i.e DSSS (Direct Sequence Spread Spectrum). Now it can be said as kind of secondary modulation technique because another modulation will take place after that, and that is where the concept of Bluetooth comes into existence, this concept is Carrier Frequency Stepping shown in Fig. 2

### B. Advantage of digital system

Video surveillance is the most common form of surveillance used in today's world and capable of serving our desired purpose here. But it has a major disadvantage for which its option has been discarded by us. Video surveillance is incapable of serving during night, fog, rain, etc. Although we do have Hi-Tech cameras which can persist the above conditions, but then it wont be cheap any longer. So we are proposing a cheap and effective surveillance by the use of Radar sensing. Analog signals are continuous and may get affected by noise thereby taking any shape. We can say that after the noise gets added to the signal, the analog signal can take any random waveform from the set of infinite number of distinct wave forms, with certainly very nil probability if we consider every event of selecting one waveform to be equally likely, i.e., The probability of occurrence of every event is equal. Once this analog waveform gets disturbed or distorted then it would be very difficult to obtain the original waveform, because of the condition imposed by the Continuity of the signal, but in getting the original signal back we have to

compromise with something and i.e., System simplicity. So now the system will be complex, it means that the whole system will consist of large number of subsystems which will make it complex and heavy weight. This phenomenon of getting of original signal from the distorted, noisy signal is called regeneration. So we can say that regeneration is very difficult in analog systems. Now let us consider the case of digital systems, they are very different from that of analog signals and systems. When an analog signal gets polluted or distorted by noise then we can say that it has taken any random waveform from the set of infinite distinct waveforms. But if we see digital signals clearly we can say that at the basic level without taking into account the concept of symbols, a digital signal consists of only finite set of waveforms that a signal can take and this finite set consists of 0 and 1, i.e now if noise gets added to the digital signal then it would take either of the waveforms 0 or 1 with the probability 0.5 considering that the event of selecting a bit will be equally likely, so now the detector simplicity will not be compromised, the detector will now be much less complex in comparison to the case of analog signals. But what advantage this thing is bringing to the systems? Well as we have seen the digital ICs, now what happens in the case of these ICs, they are having certain range of lower and higher threshold values and also some range of indeterminate values which denote some voltage levels. If the output voltage given by the IC lies in the lower threshold range then it will represent '0' and if that voltage lies in the higher range of threshold values then it will consider it '1'. The same happens with the case of digital signals and systems, when any digital signal likes 0 or 1 which will represent certain Value of voltage waveform after the pulse modulation if gets modulated by noise and appends to '1' or '0' then its obvious that the noise must be that much so that it can reduce or increase the voltage levels of the pulse modulated waveform to the respective higher or lower threshold values range of the detector so that it will treat them according to these voltage levels.

## II. SYSTEM DESCRIPTION

As we can see in the typical block diagram of the simple Digital communication system Fig.3 and modified block diagram of digital communication system Fig. 4 that there are various blocks which form the whole system, where some systems are optional and some are essential depending on the application in which we are going to use the concept of digital communication. We have also shown the modified block diagram of the digital communication system in which many blocks are omitted according to the application and these blocks are described one by one in the sequence starting from the information source.



Fig.3 Typical block diagram of digital communication

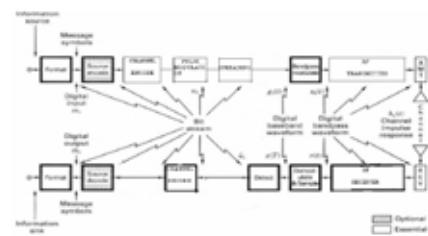


Fig.4 the modified block diagram of digital communication

### 1 Information source:

The information source can be said as any source of information that has been produced by the user sitting at the room from where he is operating the system. Now the information source can produce information in any of the three following forms:

- (a) Digital form which consists of stream of '0's and '1's
- (b) Textual form which denotes any form of textual message
- (c) Analog form which denotes any form of analog signals.

### 2. Formatting or Source coding:

After the information is generated by the information source, then it goes through either formatting or source coding. The basic function of formatting or source encoding is to bring any form of the signal (either digital, textual or analog) into the form of bit streams. This block consists of many subsystems depending upon the type of signal being used. If the information produced by the information source will be analog in nature then the this block will consist of sampling, quantization and encoding, where as if the information produced by the information source will be textual in nature then it will only consist of encoding and if the information produced by the source is digital in nature then it will fully bypass the formatting function and what we get after formatting or source coding are called PCM bits

### 3. Channel coding:

There are various techniques for channel coding. Basically the function of channel coding is to do the error correction or to reduce the probability of error. For achieving this it can use M-ary signaling. And after that it will apply some structured sequences to decrease the probability of error. Some of them which will help in bringing redundancy in the information and hence decrease the probability of error. We have used channel coding because at the RF after transmission and after reflection from the target if the signal will undergo noise, so obviously at the baseband level the probability of error will increase that's why we have used channel coding in our system. Since we are going to make one system which will work on the same mode of communication of WCDMA i.e full duplex communication as shown below in Fig. 5 so we are having only one full duplex channel for that and hence we won't be requiring the multiplexing block because here it will be only one forward channel for downlink and one reverse for uplink.

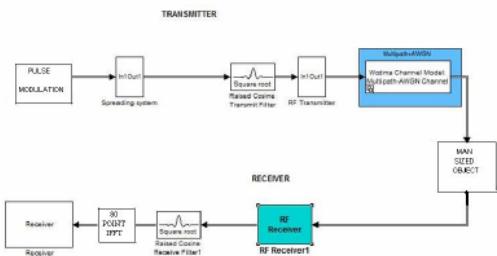


Fig.5 Essential block required at the WCDMA for doing spreading operation and for the whole model

#### 4. Pulse modulation (Baseband signaling):

There are many techniques used for baseband signaling. Baseband signaling is mainly a process of giving the bit stream of abstract '0's and '1's a certain kind of electrical waveform, otherwise how they could be send through some dedicated physical link like coaxial cable or some other physical link. We do baseband signaling also to take care of the aspect that pulse overlapping shouldn't be there otherwise there will be chance of inter-symbol interference. To avoid this there is basic criteria for the information at the bit level and that basic criteria is that the pulses should be nyquist pulses to avoid inter-symbol interference. The above pulse modulation techniques are done for binary symbols but all these modulation techniques are used for M-ary symbols, which we are going to use. The M-ary pulse modulation technique will consist of three types of modulation techniques and some are:

- Pulse amplitude modulation (PAM)
- Pulse position modulation (PPM).
- Pulse code modulation (PCM)

#### 5. Spreading:

Now our PCM bits are finally converted into the baseband signal with the help of certain baseband electrical waveforms, where we have many choices to make about the type of electrical waveforms we are going to use. Now comes the major block which will provide us with our purpose i.e spreading. Spreading is the process of encoding a baseband signal with some Psuedonoise sequence (PN codes or PN sequences) which will help in spreading the bandwidth of the signal, thereby indirectly lowering its PSD level much below. This is main concept of WCDMA on which our system will be working. There are two types of codes which are used for this spreading operation i.e orthogonal codes and PN codes. Here we are Using PN codes. In our system we can also use the most commonly used code in the case of DSSS(Direct Sequence Spread spectrum) in many cordless phones, they are using 12 bit Barker code as a spreading code for achieving spreading of the baseband signal using bipolar signaling scheme. So we can also use 12 bit Barker code for our purpose of spreading the bandwidth of the signal. The spreading process can be explained as follows Fig.5 so this will lead to the decrease in the PSD level of the signal due to which the signal will remain very less prone to noise and interference, but indirectly it is providing us with one more advantage that if we want to employ more than one system at somewhat nearer distances

then it is possible because due to spreading of the signal wouldn't undergo the interference, and as similar to the case of WCDMA. Here we get the spread signal by doing a kind of logical operation on our baseband signal with 12 bit Barker Code which is randomly generated for every symbol period, where each bit of a Barker code is called a chip.

Note: Here we won't be using the process of scrambling because in WCDMA scrambling is used to make sure that every base station in the communication process should consist of some different code so that the UE (user Equipment) will be able to detect that from which base station it is receiving a signal, but here we are having only one base station (our system).

#### 6. Bandpass modulation (Intermediate frequency):

After the signal gets spreader we will convert it into Analog form with the help of various digital to analog modulation techniques like: Amplitude shift keying, Frequency shift keying or Phase shift keying or any combination of these. Well now we have to do the carrier stepping at 80 points so for achieving that we will use the 80-ary GFSK signaling scheme(frequency hopping block in Bluetooth) to hop the carrier frequency at 80 points(GFSK modulator in Fig.2). From this we will achieve the band pass modulation. After the signal gets modulated by the band pass. This is achieved with the help of GFSK which is used in the case of Bluetooth. So we will use here 80- ary GFSK signaling scheme here and accordingly we will calculate as –

$M$  (in the case of  $M$ -array) = 80 and so  $2^k \geq 80$ . Hence we will reach a conclusion that to establish our system at the root level we will use the symbols which Consist of 7 bits/symbol, where  $k=7$ . So we can say that in the sense of Bluetooth we are hopping the carrier. Frequency at 80 different points, which is called Carrier Frequency Stepping. Video transmission requires a considerable amount of bandwidth and the bandwidth we are using is of 5 MHz, which is centered about some carrier frequency (in GHz). Hence it is efficient in sending video transmission. We have to change this carrier frequency at 80 different levels; in other words, the bandwidth of 5 MHz is swept over 80 different points. Hence:  $X / 5 = 80$  thus giving  $X = 400$  MHz of band. Thus we would be sweeping bandwidth of 5 MHz over the band of 400 MHz. So both of our purposes will be solved by this i.e. ranging as well as imaging. Well ranging is solved by the concept of WCDMA adding also the advantage of secure communication, but how this frequency hopping concept of Bluetooth is giving us the advantage of doing imaging.

#### 7. RF transmitter (Radio frequency):

Since now we have achieved the signal at the IF(intermediate frequency), i.e in the range of about 70-80 MHz, and now to make it suitable for transmitting wirelessly on the air we have to increase its power, so we have to do the frequency translation by translating its carrier frequency to a very large value.

### III. RESULT AND DISCUSSION

Modifications at the receiver after demodulation, also exposed in fig.5 now as we have said earlier that frequency is

hopping at 80 points in our system, so at the receiver after the rf demodulation 1d-ifft of the signal is to be taken and that operation of ifft will give us the peak which will tell us the size and distance of the object in the range of the system. So we can say indirectly that we are doing 1d ranging and imaging of the object under surveillance these are the steps which are required at the transmitter. Now for making all these things to solve our purpose, we have to do certain modifications at the receiver. The receiver will detect that a man sized object is there or not by doing the operation of autocorrelation just after receiving the signal. Then after the rf demodulation, when we get the signal at the intermediate frequency, we take the 80 point ifft of that signal which will give us some impulses at different times. The longest impulse and its nearby impulses will tell us the size and range of the object whose surveillance is being done. Since we have bandwidth of 5mhz so video transmission can also be done within this bandwidth. The rest of the blocks in the receiver will consist of just the inverse of the blocks in the transmitter. Simulink result of 1 d images refer to fig.1and 2 as shown below fig.6 distance of the object =  $c(\text{speed of light}) \times \text{distance of impulse from origin}$  in the graph obtained from IFFT.

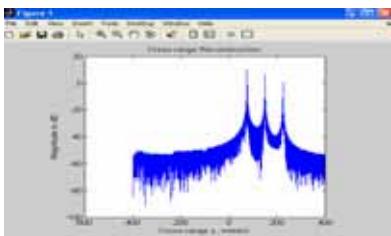


Fig.6 1D image through simulink

2 D cross range imaging simulink model are given in Fig. 7 and simulink model for range imaging is given in Fig. 8. In Fig.9 shows the 2D images of 3 objects.

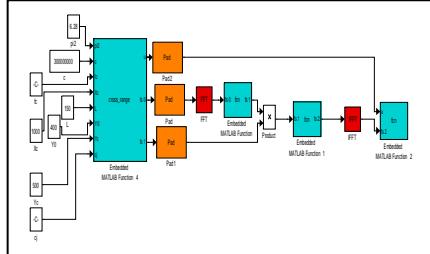


Fig.7 Simulink model for Cross-Range Imaging

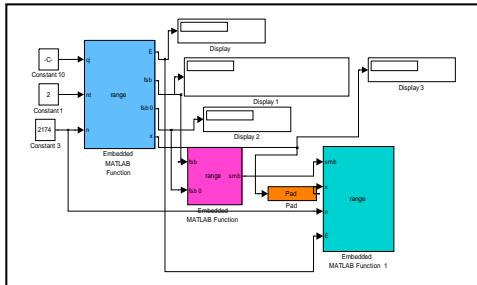


Fig.8 Model for Range Imaging

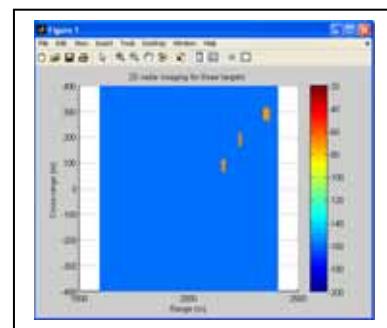


Fig.9 2D radar image for three targets

#### IV .CONCLUSION

We give the concept of digital surveillance system based on 3-G (WCDMA) for ranging and imaging used modulation technique frequency hopping spread spectrum at 80 point. It gives us 1D image of object from the help of IFFT .From the formula we easily found the distance of objet from the source. Simulink model of 1D and cross range imaging simulink model of 2D and image result are easily found.

#### V. ACKNOWLEDGMENT

I am very thankful to all the lab members of Electronic and Communication. I wish to thanks all my classmates who helped me directly or indirectly.

#### VI. REFERENCES

- [1]. Behrouz. A Forouzan "Data Communication and Networking" Tata Mc Graw Hill Publication 4<sup>th</sup> Edision 2006.
- [2]. Simon Haykin "Analog And Digital Communication" John Wiley And Sons Pvt.Ltd." 2007
- [3]. Bernard Sklar "Digital Communications Application" Prentice Hall" 2<sup>nd</sup> Edition
- [4]. MATLAB 7.4

# Audio Watermarking Algorithm Using DCT and Random Carrier Selection

Gursharanjeet Singh, Kiran Ahuja  
DAV Institute of Engineering and Technology, Jalandhar, Punjab, India.

**Abstract-** A novel audio watermarking algorithm is proposed in this paper for audio copyright protection. This algorithm embeds the watermark data into original audio signal using DCT and choosing random carrier. By choosing random carriers instead of fixed, protects the watermark from the common attacks like compression, re-quantization, echo effect, cropping and low pass filtering with high PSNR. In addition a key is also provided to enhance the protection of the watermark.

**Keywords:** DCT, Random carrier, Audio watermarking.

## I. INTRODUCTION

The growth of high speed computer networks has explored means of new business, scientific, entertainment, and social opportunities. Ironically, the cause for the growth is also of the apprehension use of digital formatted data. Digital media offer several distinct advantages over analog media, such as high quality, easy editing, high fidelity copying. The ease by which digital information can be duplicated and distributed has led to the need for effective copyright protection tools. It is done by hiding data (information) within digital audio, images and video files. The ways of such data hiding is digital signature, copyright label or digital watermark that completely characterizes the person who applies it and, therefore, marks it as being his intellectual property. Digital Watermarking is the process that embeds data called a watermark into a multimedia object such that watermark can be detected or extracted later to make an assertion about the copyright of the object. The object may be an image or audio or video or text only.

Audio files are more popular and can be easily downloaded from the internet, so piracy also becomes easy which is disastrous for the music industry. It can be restricted by audio watermarking which is either blind or non blind for audio files. If the detection of the digital watermark can be done without the original file, such techniques are called blind. Here, the source document is scanned and the watermark information is extracted. On the other hand, non blind techniques use the original file to extract the watermark [1] [2] by simple comparison and correlation procedures. However, it turns out that blind techniques are more insecure than non blind methods [3]. The watermark might contain additional information including the identity of the purchaser of a particular copy of the material.

A number of digital watermarking techniques exist for embedding information securely in an audio file depending on the domain in which the watermark is embedded. These domains may be time domain [1] [4], wavlet domain [5] [6] [7], cepstrum domain [8] [9], temporal domain [10] and spread spectrum [11] [12]. All these techniques uses other supporting methods to enhance the security and robustness of the embedded watermark. Some of these are analysis by synthesis

echo [13], support vector recognition [3], mean quantization in cepstrum domain [7], watermarking based on cloud's model [14].

For a digital watermark to be effective and practical, it should exhibit the following characteristics [15]:

- 1) *Imperceptibility*. The watermark should be invisible in a watermarked image/video or inaudible in watermarked digital music. Embedding this extra data must not degrade human perception about the object. Evaluation of imperceptibility is usually based on an objective measure of quality, called peak signal-to-noise ratio (PSNR) or a subjective test with specified procedures.
- 2) *Security*. The watermarking procedure should rely on secret keys to ensure security, so that pirates cannot detect or remove watermarks by statistical analysis from a set of images or multimedia files. An unauthorized user, who may even know the exact watermarking algorithm, cannot detect the presence of hidden data, unless he/she has access to the secret keys that control this data embedding procedure.
- 3) *Robustness*. The embedded watermarks should not be removed or eliminated by unauthorized distributors using common processing techniques, including compression, filtering, cropping, quantization and others.
- 4) *Adjustability*. The algorithm should be tunable to various degrees of robustness, quality, or embedding capacities to be suitable for diverse applications.
- 5) *Real-time processing*. Watermarks should be rapidly embedded into the host signals without much delay.

In this paper a semi blind technique is used to embed the watermark in the audio sample which is presented in part 2.1. Semi blind is due to the fact that a secret key is also embedded in the audio file with the watermark. During extraction process if the secret key will be provided then only the watermark can be extracted and it is presented in part 2.2. Experimental results and several attacks are performed to check for the robustness of the watermark in part 3.

## II. PROPOSED WATERMARKING ALGORITHM

The incoming audio bit stream is sampled and framing is done. The matrix of 64 x 64, 4096 samples, is framed on which DCT is performed. DCT converts its components into its value domains. These values are exploited with the random carrier selection to embed the watermark randomly so that after different attacks on the typical frequency sub bands or parts of audio file deleted after cropping, the watermark cannot be destroyed.

The DCT can be performed as [16] [17]:

$$X_k = \sum_{n=0}^{N-1} x_n \cos \left[ \frac{\pi}{N} \left( n + \frac{1}{2} \right) k \right] \quad k = 0, \dots, N-1. \quad (1)$$

' $x_n$ ' is te audio signal at 'n',  $X_k$  is the DCT of original audio signal  $x_n$ , N is the number of samples and (n,k) are the rows and columns of the matrix of the frame where ( $n=k=0, 1, \dots, 63$ ). There are two steps of audio watermarkin which are watermark embedding and watermark extraction.

### 2.1 Watermark embedding

Watermark embedding is the process of inserting copyright mark or bits into the audio file. The steps of watermark embedding are:

- Step 1: The watermark and the key are combined to make sequence key  $W_k$ , which is embedded in the audio file.
- Step 2: The audio sample file 'A' is segmented into frames of  $64 \times 64$  and the DCT is performed as given in equation 1. The different frequency components are choosen to embed the watermark.
- Step 3: Within each frame, choose the random carriers where the watermark bits are to be embedded. Let the random carriers selected are  $C_1, C_2, C_3, \dots, C_n$ . The value of carriers are changed at every frame by changing minimum and maximum ranges of carriers to be selected. Let the minimum value is  $C_{\min}$  and the maximum value is  $C_{\max}$ . Thus the watermark carrier,  $C_w$  must satisfy the following condition

$$C_{\min} < C_w < C_{\max} \quad (2)$$

Step 4: Repeat the step 3 untill all the bits embedded. For enhancing the robustness, the watermark is embedded M times where

$$M = \lceil \text{Length}(A) / (\text{Frame size} \times \text{Length}(W_k)) \rceil \quad (3)$$

Where 'A' is the audio sample file,  $W_k$  is the sequence key and frame size is  $64 \times 64$  i.e., 4096.

Step 5: Place all the watermarked bits and perform the inverse DCT [16] [17] as shown in equation 4.

$$X_k = \frac{1}{2}x_0 + \sum_{n=1}^{N-1} x_n \cos \left[ \frac{\pi}{N} n \left( k + \frac{1}{2} \right) \right] \quad k = 0, \dots, N - 1. \quad (4)$$

The flow chart of the watermark embedding is shown in fig.1

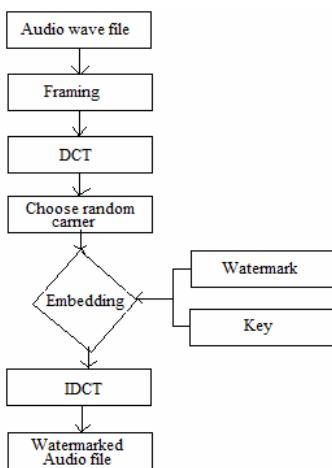


Fig.1: Flow chart of watermark embedding process.

### 2.2 Watermark extraction

The extraction process is simple and performed semi blindly as the key is required for the extraction process to be done. Here the key used a text file having 20 ASCII characters. The steps of extraction process are:

Step1: The audio file is segmented and framed as done in the embedding process. Then DCT can be performed as shown in equation 1.

Step 2: For each frame the carriers are selected where the data is embedded and the key is matched. The carriers must satisfy the condition in equation 2, if satisfy, extract the bits.

Step 3: Repeat the step 2 until watermark extracted, completely.

Step 4: If the key extracted, matched completely with the key applied then the extraction process is complete otherwise take the next sample where the watermark is repeatedly embedded.

### III. RESULTS AND DISCUSSION

The performance test and the robust test are illustrated for the proposed watermarking algorithm, and the proposed watermark detection results are compared with that of scheme [18] against various attacks. The algorithm in [18] focuses on compression attacks and low pass filtering. The peak signal to noise ratio (PSNR) is used to evaluate the algorithm which can be calculated as [19]:

$$\text{PSNR} = 10 \log_{10} \left( \frac{R^2}{\text{MSE}} \right) \quad (5)$$

Where

$$\text{MSE} = \sum_{n,k} \frac{[X_1(n,k) - X_2(n,k)]^2}{n * k} \quad (6)$$

$X_1$  and  $X_2$  are the original audio sample and watermarked audio sample. 'R' is 255 as data type used is 8-bit unsigned number representation. If the data type is double precision floating type then 'R' will be 1.

We firstly test effectiveness of algorithm without attack. The test sample is a 25 seconds wave format (44.1 KHz, mono, 16 bits/sample). The watermark is extracted and the value comes out to be PSNR as 61.75dB. The waveform of original and watermarked file is shown in fig.2 (a) and fig.2 (b) respectively.

In order to test effectiveness of the proposed algorithm, a series of following attacks tests are performed and the waveforms after attacks are shown in fig.2 (c), (d), (e), (f) and (g).

1). Requantization attack: When audio signal is quantized from 16 bit to 8 bit and again back to 16 bits, the PSNR of watermarked audio file becomes 61.3dB.

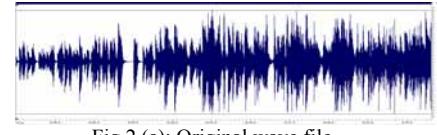


Fig.2 (a): Original wave file



Fig.2 (b): Watermarked wave file

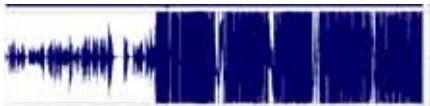


Fig.2 (c): After low pass filtering attack on watermarked file



Fig.2 (d): Requantized of watermarked attack on watermarked file



Fig.2 (e): After compressing attack on watermarked file

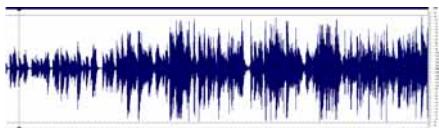


Fig.2 (f) Cropping (10%) of watermarked attack on watermarked file



Fig.2 (g) Echo effect of attack on watermarked file

- 2). Compression attack: Watermarked audio signal is compressed up to 35% and again converted back. The compression is done by converting wave file to flac audio file and again converting back to wave file, the PSNR of watermarked audio file becomes 59.6 dB.
- 3) Cropping attack: Watermarked audio file is cut 10% (or 20% or 30%) from start (or middle or behind) the PSNR of watermarked audio file becomes 61.65 dB.
- 4) Low pass filtering (LPF) attack: By adopting a low pass filtering with cut-off frequencies 22.05 KHz, the PSNR of watermarked audio file becomes 56.66dB.
- 5) Echo attack: Echo effect is added in the watermarked file with 300ms delay and 40% depth, the PSNR of watermarked audio file becomes 61.1dB.

The results in terms of PSNR of watermarked audio file with and without attacks above mentioned are shown in following Table 1.

Table 1: PSNR (dB) Performance of watermarked audio file.

Parameters	Proposed algorithm for audio watermarking	Bingwei algorithm for audio watermarking (18)	PSNR (dB)	PSNR (dB)

Without Attacks		
Watermarked file	61.71	57.59
With Attacks		
Requantization 16 bit-8 bit-16 bit	61.31	Not tested
Compression (wav-flac-wav)	59.6	-
Cropping (10% at middle)	61.65	Not tested
LPF (cut-off at 22.05 KHz)	56.66	48.34
Echo (Delay 300ms and depth 50%)	61.1	Not tested

#### IV. CONCLUSIONS

A novel audio watermarking algorithm is proposed in this paper based on embedding and extraction of watermarks which are performed by using randomly selected carriers. Subjective quality evaluation of proposed algorithm is proved by high PSNR with respect to existing algorithm. Test results proved high robustness of the proposed algorithm by showing high PSNR after the attacks i.e Requantization (61.3 dB), Compression (59.6 dB), Cropping (61.65 dB), LPF (56.66 dB), Echo (61.1 dB)) which is comparable to the other state-of-the-art audio watermarking algorithms (61.75 dB). However, these results are preliminary because this proposed algorithm performs well on lossless compression but still to be improved with lossy compression in future.

#### V. REFERENCES

- [1] Charfeddine Maha, Elarbi Maher and Ben Amar Chokri, "A blind audio watermarking scheme based on Neural Network and Psychoacoustic Model with Error correcting code in Wavelet Domain", *ISCCSP 2008*, Malta, 12-14 March 2008, pp.1138-1143.
- [2] Zhi Li, Qibin Sun and Yong Lian, "Design and Analysis of a Scalable Watermarking Scheme for the Scalable Audio Coder", *IEEE Transactions On Signal Processing*, Vol. 54, No. 8, August 2006, pp.3064-3077.
- [3] Xiangyang Wang, Wei Qi and Panpan Niu, "A New Adaptive Digital Audio Watermarking Based on Support Vector Regression", *IEEE Transactions On Audio, Speech, And Language Processing*, Vol. 15, No. 8, November 2007, pp. 2270-2277.
- [4] Paraskevi Bassia, Ioannis Pita, and Nikos Nikolaidis, "Robust Audio Watermarking in the Time Domain", *IEEE Transactions On Multimedia*, Vol. 3, No. 2, June 2001, pp.-232-241.
- [5] Lin Kezheng, Fan Bo and Yang Wei, "Robust Audio Watermarking Scheme Based on Wavelet Transforming Stable Feature", *2008 International Conference on Computational Intelligence and Security*, pp.-325-329.
- [6] Nedeljko Cvejic and Tapio Seppanen, "Robust Audio Watermarking in Wavelet Domain Using Frequency Hopping and Patchwork Method", *Proceedings of the 3rd International Symposium on Image and Signal Processing and Analysis (2003)*, pp.251-255.
- [7] B. Charmchamras, S. Kaengin, S. Airphaiboon and M. Sangworasil, "Audio Watermarking Technique using Binary Image in Wavelet Domain", *ICICS-2007*.
- [8] Sang-Kwang Lee and Yo-Sung Ho, "Digital Audio Watermarking In The Cepstrum Domain", *IEEE Transactions on Consumer Electronics*, Vol. 46, No. 3, August 2000, pp.744-750.
- [9] Vivekananda Bhat K, Indranil Sengupta, and Abhijit Das, "Audio Watermarking Based on Mean Quantization in Cepstrum Domain", *ADCOM 2008*, pp.73-77.

- [10] Aweke Negash Lemma, Javier Aprea, Werner Oomen, and Leon van de Kerkhof, “A Temporal Domain Audio Watermarking Technique”, *IEEE Transactions On Signal Processing*, Vol. 51, No. 4, April 2003, pp.1088-1097.
- [11] Shahrzad Esmaili, Sridhar Krishnan and Kaamran Raahemifar, “A Novel Spread Spectrum Audio Watermarking Scheme Based On Time-Frequency Characteristics”, *CCECE-2003, Montreal*, pp.1963-1966
- [12] Darko Kirovski and Henrique S. Malvar, “Spread-Spectrum Watermarking of Audio Signals”, *IEEE Transactions On Signal Processing*, Vol. 51, No. 4, April 2003, pp.1020-1033.
- [13] Wen-Chih Wu and Oscal Chen, “An Analysis-by-Synthesis Echo Watermarking Method”, *2004 IEEE International Conference on Multimedia and Expo (ICME)*, pp.1935-1938.
- [14] Wang Rang-ding and Xiong Yi-qun, “An Audio Aggregate Watermark Based on Cloud Model”, *ICSP2008 Proceedings*, pp.2225-2228
- [15] Wen-Nung Lie and Li-Chun Chang, “Robust and High-Quality Time-Domain Audio Watermarking Based on Low-Frequency Amplitude Modification”, *IEEE Transactions On Multimedia*, Vol. 8, No. 1, February 2006, pp.46-59.
- [16] Jain, A.K. *Fundamentals of Digital Image Processing*, Englewood Cliffs, NJ: Prentice-Hall, 1989.
- [17] Pennebaker, W.B., and J.L. Mitchell. *JPEG Still Image Data Compression Standard*, New York, NY: Van Nostrand Reinhold, 1993. Chapter 4.
- [18] Bingwei Chen<sup>1</sup>, Jiying Zhao<sup>1</sup>, Dali Wang<sup>2</sup>, “An Adaptive Watermarking Algorithm for MP3 Compressed Audio Signals”, *I<sup>2</sup>MTC 2008 - IEEE International Instrumentation and Measurement Technology Conference, Victoria, Vancouver Island, British Columbia, Canada, May 12-15, 2008*.
- [19] Shijun Xiang, *Member, IEEE*, Hyoung Joong Kim, *Member, IEEE*, and Jiwu Huang, *Senior Member, IEEE*, “Invariant Image Watermarking Based on Statistical Features in the Low-Frequency Domain”, *IEEE Transactions On Circuits And Systems For Video Technology*, Vol. 18, No. 6, June 2008, pp. 777-790.

# Comparison of CPWM and SPWM Based Speed Control of AC Servomotor

Kariyappa B. S- Asst. Professor, Hariprasad S. A- Asst. Professor , Dr. M. Uttara Kumari - Professor  
ECE Department, R V College of Engineering, Bangalore-59, India, [kariyappabs@yahoo.com](mailto:kariyappabs@yahoo.com)

**Abstract-** This paper presents comparison of CPWM and SPWM based Speed control of AC Servo motor using FPGA controller. First the basic principle of generating CPWM and SPWM gate signals are explained and then FPGA based controller is explained in the system overview. The comparison study is carried out by considering various parameters like hardware resources, speed, complexity and harmonic analysis by designing the controller using Conventional PWM and Sinusoidal PWM techniques.

The conventional PWM technique without using triangular wave as a carrier signal is less complex compared to SPWM for controlling the speed of AC servo motor. It is easier to implement and as well it takes less hardware resources. The gate count utilized in CPWM is less than 20 times as that of SPWM.

**Keywords:-** Conventional Pulse Width Modulation (CPWM), Sinusoidal Pulse Width Modulation (SPWM), Field Programmable Gate Array (FPGA), Total Harmonic Distortion (THD).

## I. INTRODUCTION

The technology growth in the field of digital electronics has undergone a significant evolution over the last few years, which has exposed to innovative openings for the implementation of increasingly complex control systems in industrial applications. The emergence of FPGA has drawn much attention due to its compact design style, hard wired logic, computation capability, simplicity, programmability and higher density compared to other controllers [7] [8].

Pulse Width Modulation is widely used in power electronics as a controller in power conversion and motion control. Among different types of modulating modes the CPWM, SPWM and Space Vector PWM are common strategies in adjustable speed drive system [1].

Pulse Width Modulation [3] of a power source involves the modulation of its duty cycle to control the amount of power sent to a load. The closed loop regulated pulse width modulated inverters have extensive applications in many types of AC power conditioning systems such as uninterruptible power supply, programmable AC source and automatic voltage regulators. The PWM inverter plays an important role in converting DC voltage to AC voltage, the performance of an AC power conditioning system is highly dependent on closed loop control of the PWM inverter.

## II. PRINCIPLE OF GENERATING CPWM AND SPWM GATE SIGNALS

The CPWM gate signals are generated by dividing the clock to get 50 Hz signal. In the on time period of 50 Hz, two gate signals are generated as shown in figure1. This will control the output voltage in one direction and in off time period of clock signal other two gate signals are generated which will control the output voltage in the other direction.

Here the available square wave is used as reference wave and carrier wave instead of triangular wave as a carrier signal.

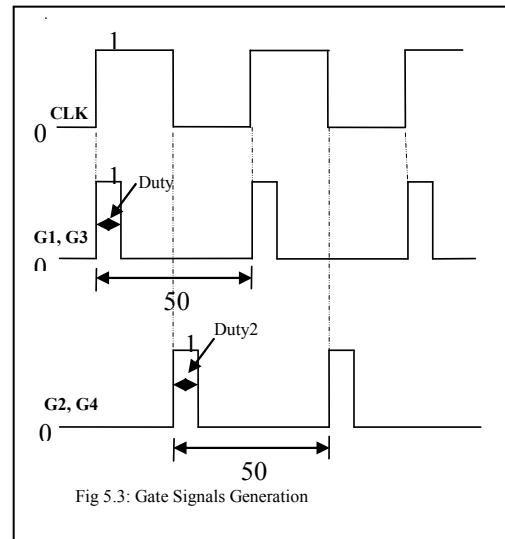


Fig 5.3: Gate Signals Generation

The SPWM gate signals are obtained by generating 50 Hz sinusoidal signal and one triangular wave of higher frequency called carrier signal. The required four SPWM gate signals to drive inverter bridge circuit is generated as follows: Sample values of first half cycle of sine wave is compared with the sample values of carrier signal [2] and a high is maintained in the gate signals 1 and 3, if reference sample value is higher than carrier sample value otherwise maintained low as shown in figure2. Similar procedure is carried out to generate SPWM gate signals 3 and 4, by comparing sample values of next half cycle with the sample value of carrier signal [6]. The FPGA based controller is designed for CPWM and SPWM by writing program in VHDL. The code is downloaded into XILINX SPARTAN 3 XC3S400 FPGA Board.

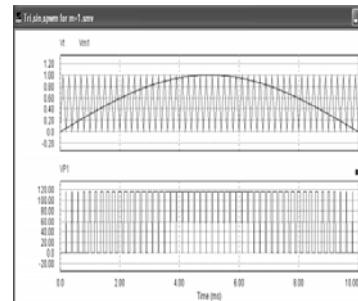


Fig 2: SPWM for First Half Cycle

## III. SYSTEM OVERVIEW

The block diagram of the FPGA Based Controller is as shown in figure3. It mainly consists of Keyboard, LCD, FPGA controller, PWM Inverter, Motor and Feedback system.

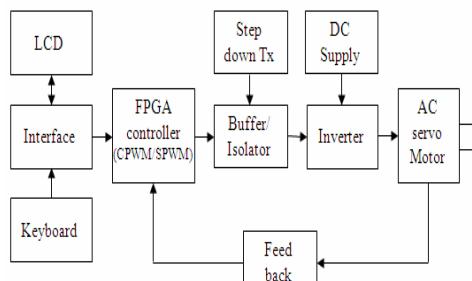


Fig 3: Block Diagram of FPGA Based Controller

The principle of operation can be explained as follows: AC servo motor is connected to the output of inverter. The speed of the motor is proportional to the width of signals connected to the inverter gate inputs. These gate signals are generated by FPGA controller based on the set speed and motor current running speed [5]. The required speed is entered through keyboard and the same is displayed using LCD. Through the IR sensors and FPGA controller the current running speed is calculated and is displayed on LCD, this speed is compared with the set speed and in accordance with the error PWM signals are generated by FPGA controller. Buffer/Isolator increases the input signal voltage to the required level. It also provides electrical isolation between FPGA controller and PWM inverter. The four signals from the FPGA controller are applied to gate inputs of PWM inverter through buffer. The AC voltage is proportional to the width of the PWM gate signals. The speed of the motor is proportional to AC voltage at the output which in turn depends on duty cycle at the PWM gate input. In CPWM the gate signals to the inverter are square wave signals and the duty cycle varies as set speed varies. In SPWM the gate signals to the inverter are sinusoidal modulated signals and modulation index varies as set speed varies.

#### IV. COMPARISON OF CPWM AND SPWM

The comparison is carried out based on the resources utilized, percentage error and THDs of CPWM and SPWM. The slice registers, LUTs and Gate counts utilized in CPWM are very much less than that of SPWM technique. Figure4 shows a plot of percentage of logic utilization v/s hardware resources of CPWM and SPWM.

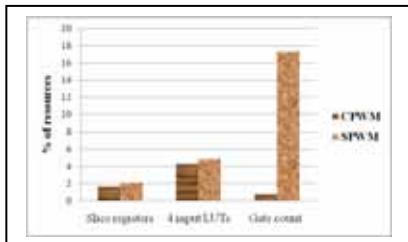


Fig 4: Percentage of Resources Utilized in CPWM and SPWM

The graph of Percentage Error v/s Set speed in RPM for CPWM and SPWM is shown in figure5. Though the percentage error is little more in CPWM, the error is less at middle order speeds. The percentage error decreases as set speed increases and it is less at middle order speeds and slightly increases at higher speeds due to inertia of the motor.

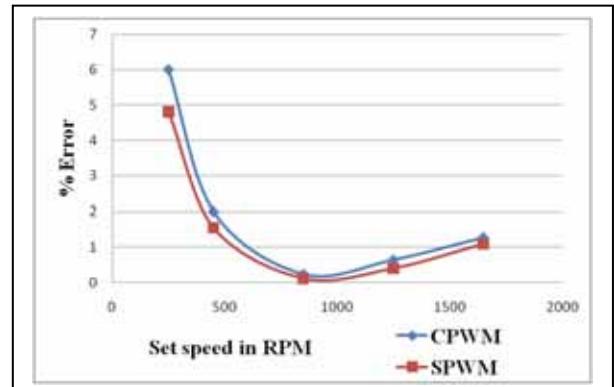


Fig 5: Percentage Error v/s Set Speed for CPWM and SPWM

The THD decreases as duty cycle/modulation index increases due to reduction in the harmonics. In SPWM, the THD is little less due to shifting of harmonics at higher frequencies compared to CPWM. Figure6 shows the plot of THD v/s different duty cycle/modulation index of CPWM and SPWM.

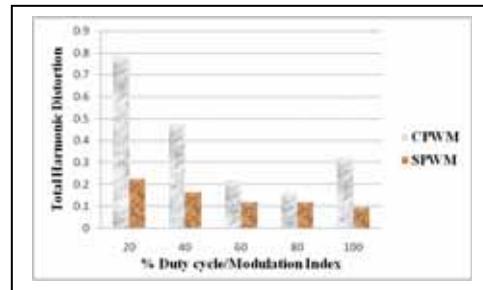


Fig 5.39: THD in CPWM and SPWM

Table 1: Comparison of CPWM and SPWM

S. No.	Parameter	CPWM	SPWM
1	Method of varying speed	By varying width of square wave	By varying modulation index
2	Hardware resources	Less	More
3	Design complexity	Simple	Complex
4	THD	Little more	Less
5	Percentage Error	Medium	Medium

## V. CONCLUSIONS

The method of varying speed, hardware resources, design complexity, THD and percentage error are compared with CPWM and SPWM techniques as shown in the table1.

The CPWM technique takes less hardware resources, reduces the hardware and software complexities and also helps in designing the controller in a simpler way compared to SPWM. The Gate counts utilized in CPWM are 20 times less than that of SPWM technique. The harmonics can be reduced by using suitable filter.

## VI. REFERENCES

- [1]. Bai Hua,Zhao Zhengming, Meng shuo,Liu Jianzheng, Sun xiaoying, "Comparison of Three PWM strategies—SPWM,SVPWM and one cycle control" IEEE 0-7803-7885-7/03 , 2003.
- [2]. Md Isa M. N, M.I. Ahmad, Sohiful A.Z. Murad and M. K. Md Arshad, "FPGA Based SPWM Bridge Inverter" American Journal of Applied Sciences, 584-586, 2007 ISSN 1546-9239, Science Publications, 2007.
- [3]. Caurentiu Dimitriu, Mihai Iucuonu, C Aghion, Ovidiu Ursaru "Control with microcontroller for PWM single phase inverter": IEEE 0-7803-7979-9/03 © 2003.
- [4]. Romli M. S. N, Z. Idris, A. Saparon and M. K. Hamzah, "An Area-Efficient Sinusoidal Pulse Width Modulation (SPWM) Technique for Single Phase Matrix Converter (SPMC)," 978-1-4244-1718-6, 2008, IEEE.
- [5]. Kariyappa B. S, Hariprasad S. A and Dr. R Nagaraj "Programmable speed controller of AC Servomotor using FPGA," International Journal of Applied Engineering Research (IJAER), ISSN 1087-1090, Vol. 3, No. 12, December 2008, India.
- [6]. Kay soon low "A DSP-based Single-Phase AC Power source" IEEE trans on industrial electronics vol-46,no.-5,OCT-1999.
- [7]. A. Fratta, G.Griffero and S. Nieddu, "Comparative Analysis among DSP and FPGA-based Control Capabilities in PWM Power Converters", the 30th Annual Conference of the IEEE Industrial Electronics Society, Nov. 2004.
- [8]. E. Monmasson, Y.A. Chapuis, "Contributions of FPGAs to the Control of Electrical Systems, a Review" IES News letter, Vol. 49, no.4, 2002.
- [9]. Heng Deng, Ramesh Oruganti and Dipti Srinivasan "PWM Methods to Handle Time Delay in Digital Control of a UPS Inverter," 1540-7985, IEEE Power Electronics Letters, Vol. 3, No. 1, March 2005.
- [10]. Da Costa J. P, H. T. Csmara and E. G. Carati, "A Microprocessor Based Prototype for Electrical Machines Control Using PWM Modulation," 0-7803-7912-8/03, 2003, IEEE.

# A VCO Based Power-Clock Supply Generator for Quasi-Adiabatic Circuits

<sup>1</sup>Prasad D Khandekar- A.P, <sup>2</sup>Dr. Mrs. Shaila Subbaraman- Prof. Dean Academics, <sup>3</sup>Achint Sharma-Student

<sup>1</sup>Member IEEE, Asst Professor-E&TC, Vishwakarma Institute of information Technology, Pune

<sup>2</sup>Professor and Dean Academics, Walchand College of Engineering, Sangli.

<sup>3</sup>International Institute of Information Technology, Pune

**Abstract-** Demands for low power electronics have motivated designers to explore new approaches to VLSI circuits. The classical approaches of reducing energy dissipation in conventional CMOS circuits include reducing the supply voltages, node capacitances, and switching frequencies. Energy-recovery circuitry, on the other hand, is a new promising approach to the design of VLSI circuits with very low energy dissipation. The supply voltage in adiabatic circuits in addition to providing the power to the circuit behaves as the clock of the circuit and for this reason is called power clock. One of the main concerns in the adiabatic logic circuits is the power clock generation. The design of an efficient power clock generator is a challenging problem in this field. This paper discusses the design and simulation results of a VCO based power-clock supply generation for quasi-adiabatic circuits. The analysis is carried out in cadence design environment using 180nm technology using cell based design approach.

## I. INTRODUCTION

Increasing demand to improve the portable system performance has fueled the necessity of low-power design techniques. Longer battery operational life has become a major design goal in low-power VLSI system. Adiabatic switching technique based on energy-recovery principle was proposed to reduce power dissipation in digital circuit [1-8]. These circuits use power-clock voltage in the form of a ramp or sinusoidal signal. The power-clock supply charges the load capacitors adiabatically during the time it is ramping up and allows the load capacitor energy to recycle back when it is ramping down. Adiabatic circuits are of two types: full adiabatic circuits with no non-adiabatic loss and quasi-adiabatic circuits with some non-adiabatic loss. The full adiabatic circuits like Split-charge Recovery Logic (SCRL) circuits and nMOS reversible recovery logic (nRERL) circuits use the technique of reversible logic to eliminate the non-adiabatic loss [2][7]. But the complexity of the full adiabatic circuit is high because of the forward and reverse logic paths. Full adiabatic circuits also suffer from the drawback of multi-phase clock generation and hence considerable amount of energy is wasted in power-clock generation [8].

Quasi-adiabatic circuit is less complex and some of the quasi-adiabatic circuits can be operated on a single-phase power-clock supply. Hence, it turns out to be the best practical energy-efficient design approach at low frequencies. We have already published the results of 2:1 MUX benchmark circuit using this approach [9-12] and have confirmed that these circuits consume considerably less energy compared to conventional CMOS circuits.

The supply voltage in adiabatic circuits in addition to providing the power to the circuit behaves as the clock of the

circuit and for this reason is called power clock. One of the main concerns in the adiabatic logic circuits is the power clock generation. In these circuits the supply voltage is desired to be a ramping voltage. Although, it can be approximated by a sinusoidal voltage that can easily be generated using resonant circuits. Some adiabatic logic families need multiphase power clocks for their cascade. The design of an efficient power clock generator is a challenging problem in this field.

In conventional dynamic digital circuits, power and clock lines are separate. The power is supplied through a DC supply voltage and the clock is generated by a separate circuit and is usually a square waveform. In adiabatic circuits, power and clock lines are mixed into a single power clock line which has both the functions of powering and timing the circuit. A DC to AC converter named power clock generator is needed for the generation of the power clock signal. The energy consumed by the power -clock generator is a dominant factor in the total energy consumed by the adiabatic system. Thus it may reduce the energy savings obtained from the energy recovery property of the adiabatic logic.

Inefficient power clock generations have become an obstacle to the adiabatic module integration into a VLSI system and hence, power clock generators with high conversion efficiency are strongly desired. A Colpitts oscillator based on LC resonant circuit is suitable for a power clock generator. The LC product determines the oscillation frequency of the power clock generator. The voltage controlled oscillator circuits, is suitable for the supply clock generator. In this paper, we present the design a VCO based power-clock supply generation and its simulation results. We have selected CMOS based VCO [13].

## II.DESIGN OF CELLS REQUIRED FOR VCO

The VCO comprises of four main sub-blocks or cells:

- V-I converter
- Damping factor controller
- Current controlled ring oscillator (CCO)
- Voltage level shifter clock

### A. V-I Converter and Damping Factor Controller

The V-I converter was modelled as a non-ideal voltage controlled current source. In the actual circuit, a transistor based current source would perform as the V-I converter. The effect of the following non-ideologies were considered:

- Threshold voltage of MOS transistors (  $V_{th}$  )
  - Output conductance of the current source (  $G$  )
- By changing the gain of the current source ( $G_m$ ) and the above non-linearity factors, one can model the behaviour of the V-I converter with good precision.

Transistors M1, M4 and M5 as shown in fig. 2, form a PMOS regulated cascade V-I converter that sources drive current to the CCO; compensation capacitor C1 stabilizes the regulated loop and suppresses the injection of high frequency supply noise into the CCO. The layout view of the V-I converter circuit is shown in fig. 2. Owing to the regulated cascade, the small signal output resistance of the CCO driving current source is very high. Hence, the driving current is nearly independent of the supply voltage for a given input voltage  $V_{control}$ , and excellent power supply noise rejection characteristics are achieved.

The DFC (damping factor controller) block consists of two parallel voltage controlled current sources and two sets of complementary switches. In terms of behavioural modelling, we can say that it is a mixed signal block, which has both digital inputs and analog terminals. Fig.1 shows the block diagram of this block.  $V_{control}$  is the same input to the VI converter block. The digital input signals are UP and DOWN and their complements divert and add current to the input of the CCO and act as a damping factor controller in the system. When all four input signals  $u$ ,  $ub$ ,  $d$ , and  $db$  are static (no pulse generated), the left branch current flows to ground through the drain current source while the right branch current adds to the main CCO driving current. When pulses are applied to the differential pairs due to a phase error at the phase detector inputs, both branch currents flow either to ground or to the CCO, depending on the polarity of the phase error. The amount of current that is subtracted from or added to the CCO driving current is proportional to the magnitude of the phase error. The loop stability increases by adding the damping factor controller block to the loop

The damping factor control circuit is biased using PMOS branches driven by  $V_{control}$  as shown in fig 2. When all four input signals  $u$ ,  $ub$ ,  $d$  and  $db$  are static (no pulse generated), the left branch current flows to ground through the diode-connected NMOS transistor while the right branch current adds to the main CCO driving current. When pulses are applied to the differential pairs .both branch currents flow either to ground or to the CCO, depending on the polarity of the phase error. The amount of the current that is subtracted from or added to the CCO driving current is proportional to the magnitude of the phase error and to the static current level, which depends on the operating frequency. The loop stability and equivalent damping factor increases with the magnitude of the dc bias currents applied to the damping factor control circuit.

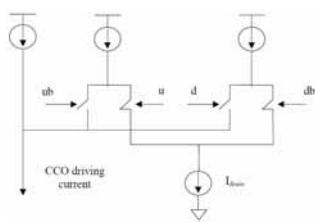


Fig. 1. Block Schematic of DFC

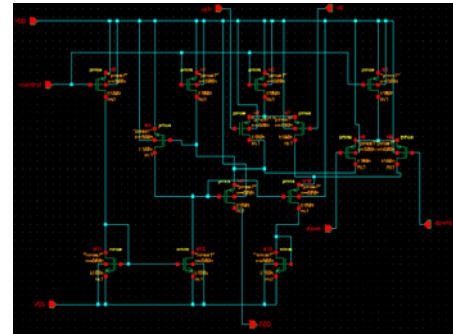


Fig. 2. Schematic of V-I Converter and DFC

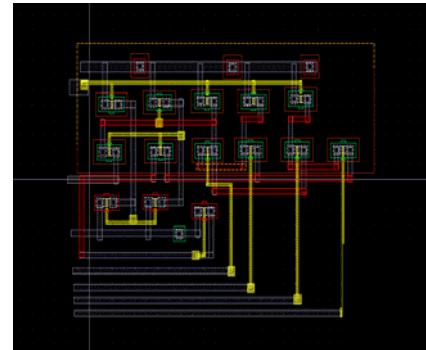


Fig. 3. Layout of V-I Converter and DFC

### B Current Controlled Oscillator

The current controlled oscillator (CCO) used in this design is a balanced five stage, single-ended ring oscillator shown in Fig. 4. In order to model this CCO one can either use the structural model shown or develop a behavioural model based on the electrical model of the CCO. The problem with structural model approach is simulation time and convergence of the oscillator response. On the other hand, this modelling technique does not have enough accuracy and we can not model input resistance and threshold voltage effect of the oscillator.

The second method of modelling is based on behavioural modelling, in this method the *current-frequency* characteristic of the oscillator is modelled using a linear or non-linear equation. The input impedance of the oscillator is modelled using a simple resistor. In order to increase the accuracy of the modelling, a second order equation for *I-F* characteristic was used. The coefficients of this equation can be parameters of the model. The VCO gain is chosen based on a trade off between operating frequency range and loop bandwidth. The CCO used in the VCO is a balanced five stage, single ended ring oscillator. The circuit schematic is shown in fig. 5. and its layout is shown in fig. 6.

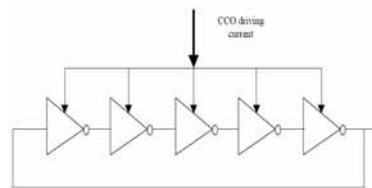


Fig. 4. Block Schematic of CCO

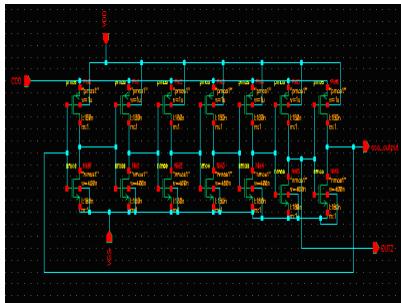


Fig. 5. Circuit Schematic of CCO

### C Voltage Level Shifter

The level shifter (LS) block changes the level of the oscillator as well as buffers the output of the oscillator. The circuit schematic is shown in fig.7. Due to the voltage drops across M1 and M5, the CCO operates from about  $V_{DD}/2$  to zero. Hence, a level-shifting buffer circuit follows the CCO to provide a rail-to- rail output signal. The layout view of the voltage level shifter is shown in fig. 8.. Since the voltage level shifter is a buffer too, it should be able to provide enough current for charging the output load while maintaining sharp edges required for triggering the counter.

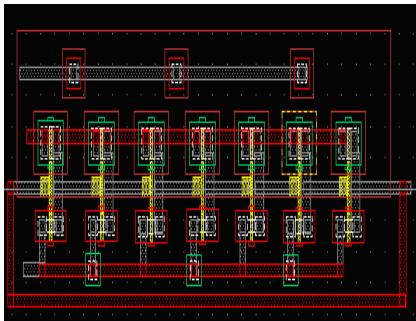


Fig. 6. Layout of CCO

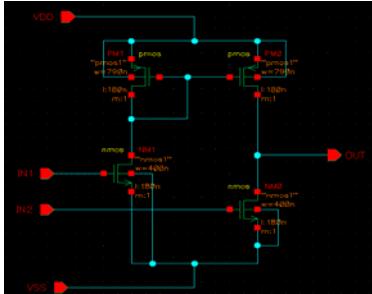


Fig. 7. Circuit Schematic of Level Shifter

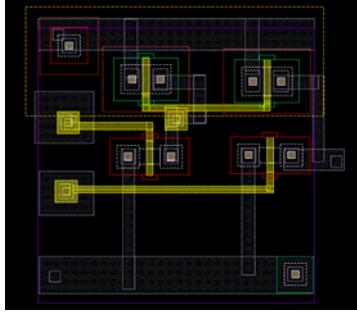


Fig. 8. Layout of Level Shifter

### III. COMPLETE DESIGN OF VCO

The cells designed for VCO were tested for their functionality in cadence design environment using 180nm technology and found that they are generating desired signal levels. Library symbols for all these cells were created. A complete circuit of VCO is then implemented in virtuoso and tested using cell based design approach. Fig. 9 shows the cell based design of VCO and its output waveforms are depicted in fig. 10.

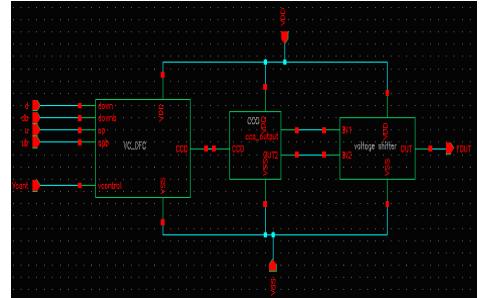


Fig. 9. Cell Based Design of VCO

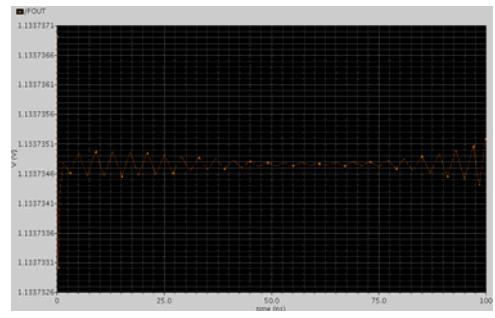


Fig. 10. Output of VCO

The circuit was tested for transient analysis of 100ns. The necessary inputs were applied as stimuli were applied in the virtuoso environment. All the transistors used have length equal to 180nm and width equal to 2μm available from gpd़k (general process design kit) 180 libraries.

The output waveform is not an ideal output of VCO. The voltage level is degraded to 1.2V instead of 1.8V but it can be restored by minimizing the node capacitances. Since the parasitic effects were not taken into account the interconnect capacitance has greatly affected the performance of the circuit function.

### IV. CONCLUSION

The satisfactory working of individual cells and the degraded output problem of the VCO clearly indicate that larger the circuit larger is parasitic effect. We are sure that these problems will be solved if the design is carried out in the environment which supports the RC extraction and allows back annotation of these RC values on the circuit schematic. Then the effects of parasitic are deterministic and the layout of the VCO can be adjusted accordingly to minimize the effects of parasitic.

#### V. ACKNOWLEDGMENT

Authors thank Prof Manish Patil and Dr D Nagchoudhuri for their valuable technical inputs. Authors also thank the office bearers of International Institute of Information Technology, Pune and Department of Electronics Engineering of Shivaji University for their constant support and resource sharing.

#### VI. REFERENCES

- [1] W. C. Athas, L.J. Svensson, J. G. Koller, N. Tzartzanis, and E. Y.-C. Chou, "Low-power digital systems based on adiabatic- switching principles," *IEEE Trans. VLSI Systems*, vol. 2, no. 4, pp 398-407, Dec. 1994.
- [2] S. G. Younis and T. F. Knight, "Practical implementation of charge recovery asymptotically zero power CMOS," in *Proc. 1993 Symp. on Integrated Syst.*, MIT Press, 1993, pp 234-250.
- [3] Y. Moon and D.-K. Jeong, "An efficient charge recovery logic circuit," *IEEE J. Solid-State Circuits*, vol. 31, no. 4, pp. 514-522, Apr. 1996.
- [4] V. G. Oklobdzija, D. Maksimovic, and F. Lin, " Pass-transistor adiabatic logic using single power-clock supply," *IEEE Trans. Circuits Syst. II Analog and Digital Signal Processing*, vol. 44, no. 10, pp 842-846, Oct. 1997.
- [5] F. Liu and K. T. Lau, "Pass-transistor adiabatic logic with NMOS pull-down configuration," *Electronics Letters*, vol. 34, no. 8, pp739-741, Apr. 1998.
- [6] D. Maksimovic, V. G. Oklobdzija, B. Nikolic, and K. W. Current, "Clocked cmos adiabatic logic with integrated single-phase power-clock supply," *IEEE Trans. VLSI Systems*, vol. 8, no. 4, pp 460-463, Aug. 2000.
- [7] J. Lim, D.-G. Kim, and S.-I. Chae, "nMOS reversible recovery logic for ultra-low-energy applications," *IEEE J. Solid-State Circuits*, vol. 35, no. 6, pp. 865-875, Jun. 2000.
- [8] P. Patra, "Approaches to design of circuits for low-power computation," Ph.D. dissertation, Univ. Texas at Austin, Aug. 1995.
- [9] P. D. Khandekar, and S. Subbaraman, "Low power 2:1 MUX for barrel shifter," *IEEE Int. Conf. ICETET 08*, Nagpur, India, IEEE xplore 978-0-7695-3267-7/08 © 2008 IEEE, pp 404-407, 16-18 July 2008.
- [10] P. D. Khandekar, and S. Subbaraman, "Optimal Conditions for Ultra Low Power Digital Circuits", *Journal of Active and Passive Electronic Device, USA*, Accepted for publishing ( paper identifier no.RC081)
- [11] P D Khandekar , and S Subbaraman, "Achieving Sub-Adiabatic Energy Dissipation by Varying  $V_{BS}$ ", *International Conference ECTI-CON 09*, Pattaya, Thailand, 6-8 May 2009, (978-1-4244-3388-9/09 © 2009 IEEE, pp600-603)
- [12] P D Khandekar , S Subbaraman, and R S Talware, "Ultra-Low Power Quasi-Adiabatic Inverter", *International Journal of Computational Intelligence Research & Applications presented in ICVCom'09*, SAINTGITS COE, Kottayam, Kerala, 16-18 April 2009, Vol.3,No.1, Jan-Jun 2009, pp 11-15.
- [13] Prithvi Shylenra, "A CMOS Based VCO and Frequency Divider for 5 GHZ Applications", M.S. Thesis, University of Texas.

# A new TISO Voltage-mode Biquad Filter using Single CCCCTA

<sup>1</sup> Jitendra Mohan, <sup>2</sup>Sudhanshu Maheshwari, <sup>3</sup>Sajai Vir Singh, <sup>4</sup>Durg Singh Chauhan

<sup>1,3</sup>Jaypee University of Information Technology, Waknaghat, Solan-173215 (India)

<sup>2</sup>Z. H. College of Engineering and Technology, Aligarh Muslim University, Aligarh-202002 (India)

<sup>4</sup>Institute of Technology, Banaras Hindu University, Varanasi-221005 (India)

jitendramv2000@gmail.com, Sudhnashu\_maheshwari@rediffmail.com, sajajvir@rediffmail.com, pdschauhan@gmail.com

**Abstract-** This paper presents a three-input single output (TISO) voltage mode universal biquad filter using single current controlled current conveyor trans-conductance amplifier (CCCTA). The proposed filter employs only single CCCTA, two capacitors and one permanently grounded resistor. The proposed filter realizes all the standard filter functions i.e. low pass(LP), band pass(BP) and high pass(HP), band reject(BR) and all-pass(AP) filters in the voltage form ,through appropriate selection of the input voltage signals. The circuit does not require inverting-type input voltage signals and double input voltage signals to realize any response in the design. The filter enjoys attractive features, such as orthogonal tunability of pole frequency and quality factor, low sensitivity performance and low power consumption of .119 mW. The circuit enjoys an current control of pole frequency and quality factor. The validity of proposed filter is verified through PSPICE simulations.

**Keywords**— CCCTA, filter, voltage mode.

## I. INTRODUCTION

Filters are the key elements in the signal processing. They are classified into two major categories: analog and digital filters. Analog filters are cheaper, faster and larger dynamic range in both amplitude and frequency as compared to digital filters. Such filters are widely used in many applications such as noise reduction, videos signal enhancement, graphic equalizers in hi-fi systems etc. voltage mode and current mode analog filters are two major categories of analog filter circuits. Voltage mode analog filter can be further divided into multi-input single output (MISO) type and single input multi-output (SIMO) type voltage filter. Recently there is a considerable attention towards designing MISO type voltage mode universal filter in voltage signal processing because of following attractive features : (i) realization of different filter functions with same topology depending upon the ports used. (ii) reduced number of active and passive components, as compared to single input and single output universal filters (iii) versatility and simplicity to the design and bring cost reduction to integrated circuit manufacturer. Over last two decades current -mode active elements have become very useful for analog filtering application due to their wider band width, larger dynamic range, less power consumption, low supply voltage and simple circuitry[1-3]. several such current mode elements and its applications have evolved namely current conveyors (CCII) , current feedback opamp, fully differential current conveyor (FDCCII), differential voltage current conveyor (DVCC), current controlled current conveyors (CCCI), ,current controlled current conveyor trans-conductance amplifier (CCCTA) and many more [4-15]. CCCTA is relatively new active element [15] and has received considerable attention as current mode active element, because its trans-conductance and

parasitic resistance can be adjusted electronically, which is suitable for analog circuit design. The flexibility of the device to operate in both current and voltage mode allows for a variety of circuit design. As far as the topic of this paper is concerned, the MISO type voltage mode filters based on a single active element are of interest. Such voltage-mode filters using different active elements have been reported in the literature [10-15]. The circuits presented in ref. [10-13] require more than three passive components and they suffer from lack of electronic tenability too. The circuit reported in ref. [14] consists of three passive components ( two capacitors and one resistor ) and single CCCTA as active element and realizes all the standard filter functions ( LP, HP, BP, BR and AP) with out requirement of any matching conditions and enjoy the feature of orthogonal and electronic tunability of filter parameters but it suffers from the following disadvantages : ( i ) requirement of both normal and inverted input voltage for all pass function realization and hence would require one or more active device (CC or op-amp) to implement with only one type of signal. (ii) resistor is not permanently grounded. So fabrication of floating resistor requires more chip area which is not suitable from fabrication point of view. The circuit reported in ref. [15] uses only two capacitor and single CCCTA and also realizes all the standard filter functions (LP, HP, BP, BR and AP) but suffers from the following disadvantages: (i) requirement of double voltage signal to obtain an all-pass response (ii) use of both normal and inverted input voltage for BR and LP function realization and (iii) use of one capacitor at port X which limits the use of filter in high frequency range. (iv) pole frequency ( $\omega_0$ ) and Quality factor (Q) are not orthogonal tunable. keeping these points in consideration, a new three-inputs single output (TISO) voltage mode universal biquad filter using single CCCTA. It uses single CCCTA, two capacitors and one permanently grounded resistor. The filter circuit provides all the five standard filter functions i.e. low pass(LP), band pass(BP) and high pass(HP), band stop(BR) and all-pass(AP) filters in the voltage form ,through appropriate selection of the input voltage signals. The circuit does not require inverting-type input voltage signals and double input voltage signals to realize any response in the design. The filter enjoys attractive features, such as orthogonal tunability of pole frequency and quality factor, low sensitivity performance and low power consumptions. The circuit also enjoys electronic current control of pole frequency and quality factor. The validity of proposed filter is verified through PSPICE simulations.

The CCCTA, shown in fig.1 is described by the following relationships.  $R_x$  and  $g_m$  are the parasitic resistance at x terminal and transconductance of CCCTA.

$$I_Y = 0, V_X = V_Y + I_X R_X, I_Z = I_X, I_O = -g_m V_Z \quad (1)$$

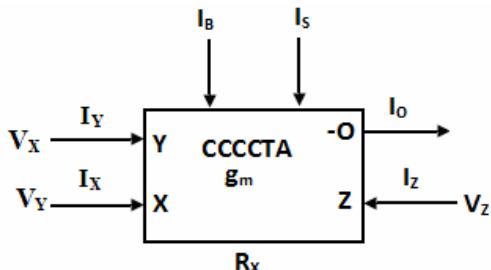


Fig.1 CCCCTA Symbol

For a CMOS CCCCTA, the  $R_X$ ,  $g_m$  can be expressed to be

$$R_X = \frac{1}{\sqrt{8\beta_n I_B}} \quad \text{and} \quad g_m = \sqrt{\beta_n I_s} \quad (2)$$

$$\text{Where } \beta_n = \mu_n C_{OX} \frac{W}{L} \quad (3)$$

where  $\mu_n$ ,  $C_{OX}$  and  $W/L$  are the electron mobility, gate oxide capacitance per unit area and transistor aspect ratio, respectively.  $I_B$  and  $I_S$  are the biasing currents of CCCCTA. The schematic symbol of CCCCTA is illustrated in Fig.1.

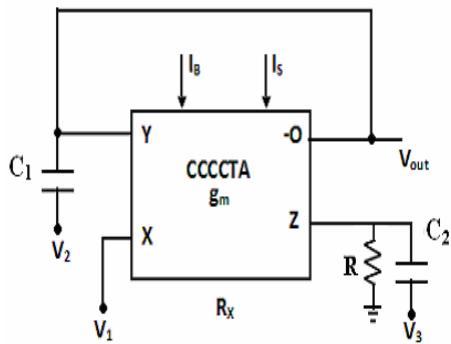


Fig. 2 . Proposed TISO Voltage-mode Biquad Filter

## II. PROPOSED TISO VOLTAGE-MODE BIQUAD FILTER

The proposed voltage-mode biquadratic filter is shown in Fig.2. It uses only single CCCCTA ,two capacitors and single grounded resistor. By routine analysis of the circuit in fig.2, the output voltage  $V_O$  can be obtained as

$$V_O = \frac{V_2 s^2 R R_X C_1 C_2 + V_2 s R_X C_1 - V_3 s g_m R R_X C_2 + V_1 g_m R}{s^2 R R_X C_1 C_2 + s C_1 R_X + g_m R} \quad (4)$$

From equations (4) various filter responses in voltage form can be obtained through appropriate selection of input voltages.

(i) high pass response with  $V_1 = 0$ ,  $V_2 = V_3 = V_{in}$ ,  $C_1 = C_2$ , and  $R g_m = 1$

(ii) low pass response with  $V_1 = V_{in}$ ,  $V_2 = V_3 = 0$

(iii) inverted band pass response with  $V_3 = V_{in}$ ,  $V_2 = V_1 = 0$ ,  $C_1 = C_2$ , and  $R g_m = 1$

(iv) notch responses with  $V_1 = V_2 = V_3 = V_{in}$ ,  $C_1 = C_2$ , and  $R g_m = 1$

(v) all pass responses with  $V_1 = V_2 = V_3 = V_{in}$ ,  $C_1 = C_2$ , and  $R g_m = 2$

It may be noted that realization of all the responses do not require any inverting-type input voltage signals and double input voltage signals to realize all the responses in the design. The inverted BP, HP, Notch and AP realization require matching conditions and that are simple to satisfy through design, particularly in monolithic technologies, where inherently matched devices are realized. The filter parameters pole frequency ( $\omega_o$ ) and the quality factor (Q) for the proposed circuit in fig 1, can be expressed as

$$\omega_o = \left( \frac{g_m}{R_X C_1 C_2} \right)^{\frac{1}{2}} = \left( \frac{1}{C_1 C_2} \beta_n \right)^{\frac{1}{2}} (8 I_B I_S)^{\frac{1}{4}} \quad (5)$$

$$Q = R \left( \frac{C_2 g_m}{C_1 R_X} \right)^{\frac{1}{2}} = R \left( \frac{C_1 \beta_n}{C_2} \right)^{\frac{1}{2}} (8 I_B I_S)^{\frac{1}{4}} \quad (6)$$

From equations (5) and (6), it can be remarked that both the pole frequency and quality factor are orthogonally controllable by variation of R. In addition both parameters can be electronically controlled by biasing current  $I_B$  and  $I_S$ . The ideal active and passive sensitivities of the parameters  $\omega_o$ , and Q, can be expressed as given in equations (7) and (8).

$$S_{C_1, C_2}^{\omega_o} = -\frac{1}{2}, S_{I_B, I_S}^{\omega_o} = \frac{1}{4}, S_{\beta_n}^{\omega_o} = \frac{1}{2}, S_R^{\omega_o} = 0 \quad (7)$$

$$S_{C_1, \beta_n}^Q = \frac{1}{2}, S_{C_2}^Q = -\frac{1}{2}, S_{I_B, I_S}^Q = \frac{1}{4}, S_R^Q = 1 \quad (8)$$

From the above calculations, it can be seen that all sensitivities are constant and equal or smaller than 1 in magnitude. To see the effects of non idealities, the defining equations of the CCCCTA can be rewritten as the following

$$V_X = \beta V_Y + I_X R_X \quad (9)$$

$$I_Z = \alpha I_X \quad (10)$$

$$I_O = -\gamma g_m V_Z \quad (11)$$

Where  $\beta$ ,  $\alpha$  and  $\gamma$  are transferred error values deviated from one. In the case of non-ideal and reanalyzing the proposed TISO voltage filter in Fig. 2 , it yields voltage outputs as

$$V_{OUT} = \frac{V_2 s^2 R R_X C_1 C_2 + V_2 s R_X C_1 - V_3 s \gamma g_m R R_X C_2 + V_1 \gamma \alpha_1 g_m R}{s^2 R R_X C_1 C_2 + s C_1 R_X + \alpha \beta \gamma g_m R} \quad (12)$$

In this case, the  $\omega_o$  and Q are changed to

$$\omega_o = \left( \frac{\alpha \beta \gamma g_m}{R_X C_1 C_2} \right)^{\frac{1}{2}} \quad (13)$$

$$Q = R \left( \frac{\alpha \beta \gamma C_2 g_m}{C_1 R_X} \right)^{\frac{1}{2}} \quad (14)$$

The non-ideal sensitivities can be found as

$$S_{C_1, C_2, R_X}^{\omega_o} = -\frac{1}{2}, S_{\alpha_1, \beta, \gamma, g_m}^{\omega_o} = \frac{1}{2}, \quad (15)$$

$$S_{C_1, R_X}^Q = -\frac{1}{2}, S_{C_2, \alpha_1, \beta, \gamma, g_m}^Q = \frac{1}{2}, S_R^Q = -1 \quad (16)$$

From the above results, it can be observed that all the sensitivities due to non-ideal effects are equal or less than 1 in magnitude.

### III. SIMULATION RESULTS

P-spice simulations are carried out to demonstrate the feasibility of the proposed circuit in fig.2 The CCCCTA is realized [14] by CMOS implementation as shown in fig.3. The simulations use a level 3, 0.35  $\mu$ m MOSFET parameters from TSMC (the model parameters are given in the table 1). The dimensions of PMOS are determined as  $W=3\mu\text{m}$  and  $L=2\mu\text{m}$ . In NMOS transistors, the dimensions are  $W=3\mu\text{m}$  and  $L=4\mu\text{m}$ . Fig.4 Shows the simulated gain and phase responses of the HP, LP, BP, BR, and AP of the proposed circuit in Fig.2, designed with  $I_B=4\mu\text{A}$ ,  $I_S=47.5$ ,  $R=15\text{K}$  and  $C_1=C_2=10\text{pf}$ . The supply voltages are  $V_{DD} = -V_{SS}=1.5\text{V}$ . The simulated pole frequency is obtained as 1 MHz. The power dissipations of the proposed circuit for the design values is found as .197 mW that is a low value. The simulation results agree quite well with the theoretical analysis.

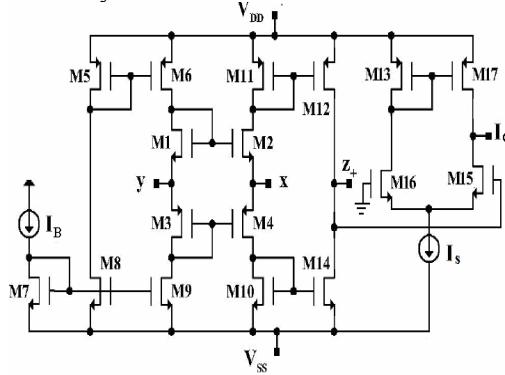


Fig.3 Implementation of CCCCTA using CMOS transistors

Table 1 TSMC SPICE parameters for level 3, 0.35  $\mu$ m CMOS process

Parameters	
NMOS	.MODEL MbreakN NMOS (LEVEL=3 TOX=7.9E-9 NSUB=1E17 GAMMA=0.5827871 PHI=0.7 VTO=0.5445549 DELTA=0 UO=436.256147 ETA=0 THETA=0.1749684 KP=2.055786E-4 VMAX=8.309444E4 KAPPA=0.2574081 RSH=0.0559398 NFS=1E12 TPG=1 XJ=3E-7 LD=3.162278E-11 WD=7.046724E-8 CGDO=2.82E-10 CGSO=2.82E-10 CGBO=1E-10 CJ=1E-3 PB=0.9758533 MJ=0.3448504 CJSW=3.777852E-10 MJSW=0.3508721)
PMOS	.MODEL MbreakP PMOS (LEVEL=3 TOX=7.9E-9 NSUB=1E17 GAMMA=0.4083894 PHI=0.7 VTO=-0.7140674 DELTA=0 UO=212.2319801 ETA=9.999762E-4 THETA=0.2020774 KP=6.733755E-5 VMAX=1.181551E5 KAPPA=1.5 RSH=30.0712458 NFS=1E12 TPG=-1 XJ=2E-7 LD=5.000001E-13 WD=1.249872E-7 CGDO=3.09E-10 CGSO=3.09E-10 CGBO=1E-10 CJ=1.419508E-3 PB=0.8152753 MJ=0.5 CJSW=4.813504E-10 MJSW=0.5)

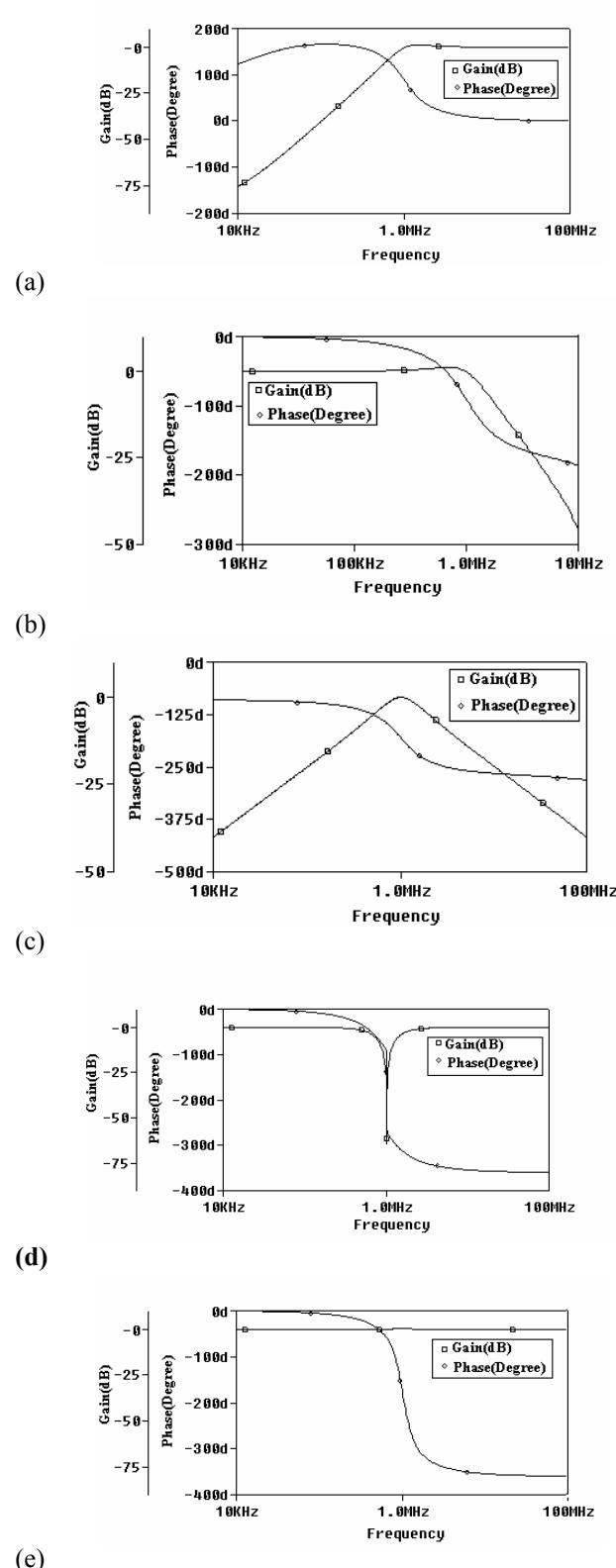


Fig.4 Gain and phase responses of the biquad filter (a) HP (b) LP (c) BP (d) BR (e) AP

Further simulations are carried out to verify the total harmonic distortion (THD). The circuit is verified by applying a sinusoidal voltage of varying frequency and constant

amplitude of 60mV. The THD is measured at the low-pass voltage output (when  $V_1=V_{in}$  and  $V_2=V_3=0$ ). The THD is found to vary from 0.0054% at 20khz to 3% at 180khz. The time domain response of low-pass output is shown in Fig.5. It is observed that 120mV peak to peak input voltage sinusoidal signal levels are possible with out significant distortions. Thus both THD analysis and time domain response of low pass output confirm the practical utility of the proposed circuit.

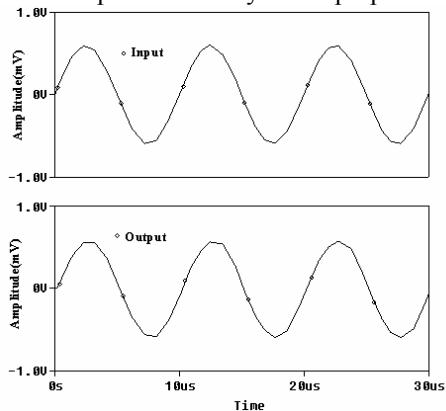


Fig.5 The time domain input and low pass output waveforms of the circuit in Fig. 2

#### IV. CONCLUSION

This paper presents a new TISO voltage-mode biquad filter using single CCCCTA. The proposed filter offers the following advantages :

- (i). Realizing of LP, HP, BP, BR, and AP responses in voltage form.
- (ii). The resistor being permanently grounded
- (iii). Low sensitivity figures and low power consumptions.
- (iv). The  $\omega_o$ , Q and  $\omega_o/Q$  are electronically tunable with bias currents of CCCCTA
- (v). Both  $\omega_o$  and Q are orthogonally tunable.
- (vi). Single active element.
- (vii) No requirement of inverting-type input voltage signals and double input voltage signals to realize any response in the design.
- (viii) Suitable for high frequency applications.

With above mentioned features it is very suitable to realize the proposed circuit in monolithic chip to use in battery powered, portable electronic equipments such as wireless communication system devices.

#### V. REFERENCES

- [1]. G. W. Roberts and A.S. Sedra, "All current-mode frequency selective circuits," Electronics Lett., vol .25, pp. 759-761,1989.
- [2]. C. Toumazou, F. J. Lidger and D. G. Haigh, " Analogue ic design: the current mode approach, London: Peter Peregrinus,1990.
- [3]. B. Wilson, "Recent developments in current mode circuits," IEE Proc. G, vol. 137, 1990, pp. 63-77.
- [4]. C. M Chang and S. H. Tu, "Universal voltage-mode filter with four inputs and one output using two CCII+s," Int'l J. Electronics, vol .86, 1999, pp.305-309.
- [5]. C. M. Chang and M. S. Lee, "Universal voltage-mode filter with three inputs and one output using three current conveyors and one voltage follower," Electronics Lett., vol .30, pp. 2112-2113 , 1994.
- [6]. J. W. Horng , C. G. Tsai and M. H. Lee, "Novel universal voltage-mode biquad filter with three inputs and one output using only two current conveyors," Int'l J.Electronics, vol.80, pp.543-546, 1996.

- [7]. R. Senani and S. S. Gupta, "Universal voltage mode/current mode biquad filter realized with current feedback opamps," Frequenz, vol.51, pp.203-208 , 1997.
- [8]. S. K. Paul, N. Pandey and S. B. Jain, "Realization of plus-type CCCIIs based voltage mode universal filter," International Symposium on Integrated Circuits(ISIC-2007), pp.119-122, 2007.
- [9]. S. Maheshwari, "High input impedance voltage mode first order all-pass sections," Int'l J. circuit Theory and Applications, vol.36, pp.511-522, 2008.
- [10]. A. U. Keskin, " Multifunction biquad using single CDBA", Electrical Engineering, vol.88, pp.353-356 , 2006
- [11]. C. M. Chang and H. P. Chen, "Single FDCCII-based tunable universal voltage-mode filter," Circuit Systems and Signal processing ,vol.24, 2005, pp.221-227.
- [12]. S. Maheshwari, " High performance voltage-mode multifunction filter with minimum component count," Wseas Transations on Electronics, issue 6, vol.5, pp.244-249, 2008.
- [13]. P. Kumar and K. Pal, " Universal biquadratic filter using a single current conveyor," J. of Active and Passive Electronic Devices, vol.3, pp.7-16, 2008.
- [14]. M. Siripruchyanun, P. Silapan and W. Jaikla, " Realization of CMOS current controlled current conveyor transconductance amplifier and its applications", J. of Active and Passive Electronic Devices, vol.4, pp. 35-53, 2009.
- [15]. M. Siripruchyanun and W. Jaikla, "Current controlled current conveyor transconductance amplifier (CCCCTA): a building block for analog signal processing," Electrical Engineering,vol.90, pp. 443-453, 2008.

# Web Mining Benefiting Societal Areas: A Review

Lecturer-Poonam Katyal<sup>1</sup>, Lecturer- Payal Gulati<sup>2</sup>

<sup>1</sup>CITM, Faridabad, Haryana, India, <sup>2</sup>YMCAIE, Faridabad, Haryana, India

**Abstract-** Web mining has been extended to use of data mining and other similar techniques to discover resources, patterns and knowledge from the web and web-related data. This paper focus on web mining that benefit societal areas by extracting new knowledge, providing support for decision making and empowering valuable management of societal issues. As per the critical review at the literature web mining research efforts lead to user (or group of users) satisfaction by providing accurate and relevant information retrieval, providing customized information, learning about user's demands so that services can target specific groups or even individual users; and by providing personalized services.

**Key Words-** Web Mining, E-Learning, E-Governance, E-Politics, E-Democracy

## I. INTRODUCTION

The WWW has become a huge, diverse, and dynamic information reservoir accessed by people with different backgrounds and interests. On the Web, access information is generally collected by web servers and recorded in the access logs. Web mining and user modeling are the techniques that make use of these access data, discover the surfer's browsing patterns, and improve the efficiency of web surfing. Web mining has been extended to denote the use of data mining and other similar techniques to discover resources, patterns and knowledge from the web and web-related data. Web Mining is categorized into three categories: Web structure mining [6, 11], Web Usage Mining [6, 13], Web Content mining [6]. Current web mining research [10] on E learning is based on web usage mining as the focus has been on how the student performs. E-Politics is based on web structure mining to identify political groups. It seems that the fields of E-services and web mining have recently met each other.

This paper is organized in the following way. Section 2 discusses about Web Mining and its categories. Web mining benefits in societal areas are presented in Section 3. And finally section 4 comprise of the conclusion.

## II. WEB MINING TAXONOMY

Web mining is divided into three mining categories according to the different sources of data analyzed.

- a) Web content mining focus on the discovery of knowledge from the content of web pages and therefore the target data consist of multivariate type of data contained in a web page as text, images, multimedia etc.
  - b) Web usage mining focus on the discovery of knowledge from user navigation data when visiting a website. The target data are requests from users recorded in special files stored in the website's servers called log files.
  - c) Web structure mining deals with the connectivity of websites and the extraction of knowledge from hyperlinks of the web.
- Figure. 1 shows the taxonomy of the Web mining.

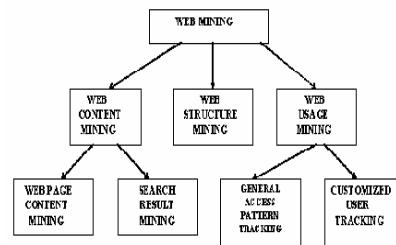


Figure 1: Taxonomy of Web

### 2.1 Web Content Mining

Web Content mining is concerned with the discovery of useful information from the real data in the web pages i.e. Extracting useful information from the contents of web documents. Content data corresponds to the collection of facts a web page was designed to convey to the users. This usually includes text, images, audio, video and hyperlinks. In the last several years, most content mining research focused on text classification and knowledge discovery in texts. Some research also concentrated on multimedia data mining, but this direction receives less attention than those on text or hypertext contents. Web content mining is related but different from data mining and text mining. It is related to data mining because many data mining techniques can be applied in Web content mining. It is related to text mining because much of the web contents are texts. However, it is also quite different from data mining because Web data are mainly semi-structured and/or unstructured, while data mining deals primarily with structured data. Web content mining is also different from text mining because of the semi-structure nature of the Web, while text mining focuses on unstructured texts. Web content mining thus requires creative applications of data mining and/or text mining techniques and also its own unique approaches.

Web content mining is more than selecting relevant documents on the web. Web content mining is related to information extraction and knowledge discovery from analyzing a collection of web documents. Related to web content mining is the effort for organizing the semi-structured web data into structured collection of resources leading to more efficient querying mechanisms and more efficient information collection or extraction. This effort is the main characteristic of the "Semantic Web", which is considered as them next web generation. Semantic Web is based on "ontologies", which are meta-data related to the web page content that make the site meaningful to search engines. Sebastian study may be used as a source for web content mining.

### 2.2 Web Structure Mining

Web structure mining is closely related to analyzing hyperlinks and link structure on the web for information retrieval and knowledge discovery. Web structure mining can

be used by search engines to rank the relevancy between websites classifying them according to their similarity and relationship between them [6]. Goggle search engine, for instance, is based on PageRank algorithm [7], which states that the relevance of a page increases with the number of hyperlinks to it from other pages, and in particular of other relevant pages. Web structure mining is also used for discovering community networks by extracting knowledge from similarity links. The term is closely related to “link analysis” research, which has been developed in various fields over the last decade such as computer science and mathematics for graph-theory, and social and communication sciences for social network analysis [8, 11, 10]. The method is based on building a graph out of a set of related data [4] and to apply social network theory [10] to discover similarities.

Recently Getoor and Diehl [9] introduce the term “link mining” to put special emphasis on the links as the main data for analysis and provide an extended survey on the work that is related to link mining. The structure of a typical web graph as shown in Figure 2, consists of web pages as nodes, and hyperlinks as edges connecting between two related pages

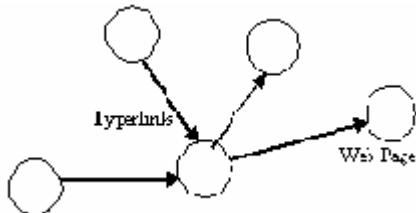


Figure2. Web graph structure

### 2.3 Web Usage Mining

While Web structure mining and Web content mining exploit the real or primary data on the WWW, Web usage mining works on the secondary data such as Web server access logs, proxy server logs, browser logs, user profiles, cookies, user queries, and bookmark data. Web usage-mining aims at utilizing data mining techniques to discover the usage patterns from these secondary data and better fulfill the needs of Web-based applications. Web usage mining research focuses on finding patterns of navigational behavior from users visiting a website. These patterns of navigational behavior can be valuable when searching answers to questions like: How efficient is our website in delivering information? How the users perceive the structure of the website? Can we predict user's next visit? Can we make our site meeting user needs? Can we increase user satisfaction? Can we targeting specific groups of users and make web content personalized to them? Answer to these questions may come from the analysis of the data from log files stored in web servers. Web usage mining has become a necessity task in order to provide web administrators with meaningful information about users and usage patterns for improving quality of web information and service performance. Figure. 3 show the architecture of Web Usage Mining.

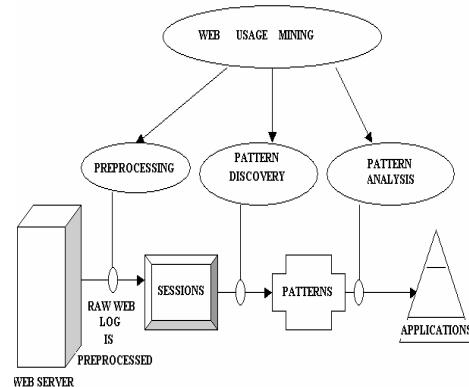


Figure3. Web Usage Mining Architecture

### III. WEB MINING IN SOCIETAL BENEFIT AREAS

Web mining may benefit those organizations that want to utilize the web as a knowledge base for supporting decision-making. Pattern discovery, analysis and interpretation of mined patterns may lead to better decisions for the organization and for the provided services. E-commerce and E-business were two fields empowered by web mining having lots of applications for increasing sales and doing intelligence business. Lots of web mining applications found in the literature describe the effectiveness of the application from the web administration point of view. The target in these applications is taking advantage of the mined knowledge from the users to increase the benefits for the organization. This approach focus on social beneficial areas from web mining, so our point of view is on web mining applications that can help users or group of users. An obvious societal benefit is that web mining research efforts lead to user (or group of users) satisfaction by providing accurate and relevant information retrieval; by providing customized information; by learning about user's demands so that services can target specific groups or even individual users; and by providing personalized services.

Investigation of societal beneficial areas divides web mining research into social beneficial areas: a) E-learning, where educational improvement is the benefit; b) E-Government, where improvement of government services to citizens is the benefit; c) E-politics and E-democracy, where community participation is the benefit d) Recommendation systems, where improvement of social services is the benefit; e) Digital Libraries, where improvement of productivity by diffusion of ideas is the benefit; f) Security and crime investigation, where public safety is the benefit. The first three areas are considered to be part of e-services. Following subsections shows how current web mining research is related to provide efficient e-services.

#### 3.1. E-Learning

Web mining can be used for enhancing the learning process in e-learning environments. Applications of web mining to e-learning are usually web-usage based applications. Bellaachia et. al. [5], introduce a framework, where they use logs to analyze the navigational behavior and the performance of e-learners so that to personalize the learning content of an adaptive learning environment in order to make the learner

reach his learning objective. Zaiane [13] studies the use of machine learning techniques and web usage mining to enhance web-based learning environments for the educator to better evaluate the learning process and for the learners to help them in their learning task. Students' web logs are investigated and analyzed in Cohen and Nachmias [12] in a web-supported instruction model.

### 3.2. E-Government

The process by organizations that interact with citizens for satisfying user (or group of users) preferences leads to better social services. The major characteristics of e-government systems are related to the use of technology to deliver services electronically focusing on citizens needs by providing adequate information and enhanced services to citizens to support political conduct of government. Empowered by web mining methods e-government systems may provide customized services to citizens resulting to user satisfaction, quality of services, support in citizens decision making, and finally leads to social benefits. Such social benefits much rely on the organization's willingness, knowledge and ability to move on the level of using web mining. The e-government dimension of an institution is usually implemented gradually.

### 3.3. E-Politics and E-Democracy

E-politics provides political information to the citizens improving political transparency and democracy, benefiting parties, candidates, citizens and the society. Election campaigners, parties, members of parliament and members of local governments on the web are part of e-politics. Despite the importance of e-politics in democracy there is limited web mining methods to meet citizen needs. As per reviewing the work done in this domain, link analysis has been used to estimate the size of political web graphs [2], to map political parties network on the web [1] and to investigate the U.S. political Blogosphere [3]. Political web linking is also studied by Foot et. al. [8] during the U.S. Congressional election campaign season on the web.

## IV. CONCLUSION

As per the critical review at the literature web mining research efforts lead to user's satisfaction, providing customized information, tracking about user's and thus providing personalized services. The E-Services areas of societal interest that have been included in this work are: E-learning, Governance, E-Politics and E-Democracy.

## V. REFERENCES

- [1]. Ackland, R., and Gibson, R. (2004). Mapping Political Party Networks on the WWW. *Australian Electronic Governance Conference*, Melbourne.
- [2]. Ackland, R. (2005). Estimating the Size of Political Web Graphs, revised paper presented to ISA Research Committee on Logic and Methodology Conference, Retrieved from [http://acsr.anu.edu.au/staff/ackland/papers/political\\_web\\_graphs.pdf](http://acsr.anu.edu.au/staff/ackland/papers/political_web_graphs.pdf) (15th, January, 2007).
- [3]. Ackland, R. (2005). Mapping the U.S. Political Blogosphere: Are Conservative Bloggers More Prominent? *paper presented to BlogTalk Downunder*, Sydney, January, 2007.
- [4]. Badia, A., and Kantardzik, M. (2005). Graph Building as a Mining Activity: Finding Links in the Small. *Proceedings of the 3rd International Workshop on Link Discovery*, ACM Press, 17-24.

- [5]. Bellaachia, A., Vommina, E., and Berrada, B. (2006). Minel: A Framework for Mining E-learning Logs, *Proceedings of the 5th IASTED International Conference on Web-based Education*, Puerto Vallarta, Mexico, 259-263.
- [6]. Kosala, R., and Blockeel, H., (2000). Web Mining Research: A Survey, *ACM* 2(1):1-15.
- [7]. Brin, S., and Page, L. (1998). The Anatomy of a Large- Scale Hypertextual Web Search Engine, *Proceedings of the 7th International World Wide Web Conference*, Elsevier Science, New York, 107-117.
- [8]. Foot, K., Schneider, S., Dougherty, M., Xenos, M., and Larsen, E. (2003). Analyzing Linking Practices: Candidate Sites in the 2002 U.S. Electoral Web Sphere. *Journal of Mediated Communication*, 8(4).
- [9]. Getoor, L. and Diehl, C.P. (2005). Link Mining: A Survey. *ACM SIGKDD Explorations Newsletter*, 7(2): 3-12.
- [10]. Wasserman, S. and Faust, K. (1994). *Social Network Analysis: Methods and Applications*, Cambridge University Press.
- [11]. Park, H.W. (2003). Hyperlink Network Analysis: A New Method for the Study of Social Structure on the Web, *Connections*, 25(1), 49-61.
- [12]. Cohen, A., and Nachmias, R. (2006). A Quantitative Cost Effectiveness Model for Web-Supported Academic Instruction. *The Internet and Higher Education*, 9(2):81-90.
- [13]. Zaiane, O.R. (2001). Web Usage Mining for a Better Web-Based Learning Environment. in *Proceedings of Conference on Advanced Technology for Education*, pp, Banff, Alberta, Canada, 60-64

# Voltage-mode Multi-function Filter with Minimum Component Count

\*Jitendra Mohan, \*Sajai Vir Singh, \*\*Sudhanshu Maheshwari, \*\*\*Durg Singh Chauhan

\*Dept. of Electronics and Communications, Jaypee University of Information Technology, Waknaghat, Solan-173215 (India)

\*\*Dept. of Electronics Eng., Z. H. College of Eng. and Tech., Aligarh Muslim University, Aligarh-202002 (India)

\*\*\*Department of Electrical Engineering, Institute of Technology, Banaras Hindu University, Varanasi-221005 (India)

[jitendramv2000@gmail.com](mailto:jitendramv2000@gmail.com), [Sudhnashu\\_maheshwari@rediffmail.com](mailto:Sudhnashu_maheshwari@rediffmail.com), [sajaivir@rediffmail.com](mailto:sajaivir@rediffmail.com), [pdschauhan@gmail.com](mailto:pdschauhan@gmail.com)

**Abstract-** This paper presents a new second order multifunction filter using one active element and four passive components. The proposed configuration realizes all standard filter functions i.e low pass(LP), high pass(HP), band pass(BP), band reject(BR) and all pass(AP) filters, through appropriate selection of the input voltage signals, with the advantage of minimum component count structure, and low active and passive sensitivity performance. PSPICE simulation results are given to verify the proposed circuit.

**Keywords**—Active filters, voltage-mode, current conveyor, multifunction filters.

## I. INTRODUCTION

Multi-function filters have received considerable attention continuously for their convenience on signal processing with a single circuit. Due to the capability to realize simultaneously more than one basic filter function with the same topology, continued researches have focused on realizing universal filters. Recently a number of voltage mode multifunction filter employing different types of active element such as current conveyor and its different variation have been reported in the literature [1-11]. The vast majority of the available circuit's uses excessive number of active and/or passive components [1-10], whereas, the circuit of [11] is a minimum component count work but realizes only three standard filter transfer functions. The circuits presented in [4-7, 9-10] require both normal type and inverting type of voltage input signals to realize all pass function, hence would require one more active device. The circuits presented in [5] require complex matching condition due to four inputs.

In this paper, a new configuration is proposed for realizing a voltage mode multifunction filter with minimum component count (four passive and one active). The proposed circuit is based on fully differential second generation current conveyor (FDCCII), an active element to improve the dynamic range in mixed mode application, where fully differential signal processing is required. The application of FDCCII in filters and oscillator design were demonstrated in [12-16]. PSPICE simulation results using TSMC 0.35μm CMOS parameters are given to validate the circuit.

## II. CIRCUIT DESCRIPTIONS

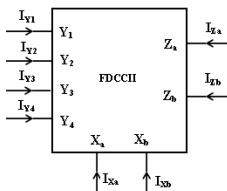


Figure 1. Circuit Symbol

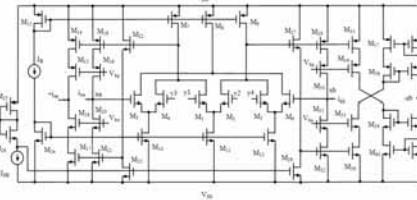


Figure 2. CMOS implementation of FDCCII

The FDCCII, whose electrical symbol is shown in Fig. 1, is a eight terminal network with terminal characteristics described by

$$\begin{bmatrix} i_{y1} \\ i_{y2} \\ i_{y3} \\ i_{y4} \\ V_{xa} \\ V_{xb} \\ i_{za} \\ i_{zb} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} V_{y1} \\ V_{y2} \\ V_{y3} \\ V_{y4} \\ i_{xa} \\ i_{xb} \\ V_{za} \\ V_{zb} \end{bmatrix} \quad (1)$$

The CMOS implementation of FDCCII is shown in Fig. 2 [12].

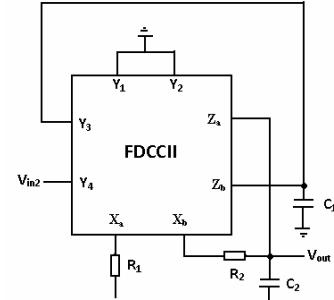


Figure 3. Proposed voltage-mode multifunction filter circuit.

The proposed voltage-mode multifunction filter configuration is shown in Fig.3. It is based on single FDCCII, two capacitors and two resistors. Routine analysis yields the voltage transfer function as

$$V_{\text{OUT}}(s) = \frac{s^2 C_1 C_2 R_1 R_2 V_{\text{in}3} + s C_1 R_1 V_{\text{in}2} + V_{\text{in}2} - s C_1 R_2 V_{\text{in}1}}{s^2 C_1 C_2 R_1 R_2 + s C_1 R_1 + 1} \quad (2)$$

From equation (2), one can realise various filters as follows.

- (i). **Low pass:** If  $V_{in3} = 0$  (grounded),  $V_{in1} = V_{in2} = V_{in}$  and  $R_1=R_2$ , a second order band pass filter can be obtained. The voltage transfer function is given by

$$\frac{V_{out}}{V_{in}} = \frac{1}{s^2 C_1 C_2 R_1 R_2 + s C_1 R_1 + 1} \quad (3)$$

- (ii). **High pass:** If  $V_{in2} = V_{in1} = 0$  (grounded),  $V_{in3}=V_{in}$ , a second order high pass filter can be obtained. The voltage transfer function is given by

$$\frac{V_{out}}{V_{in}} = \frac{s^2 C_1 C_2 R_1 R_2}{s^2 C_1 C_2 R_1 R_2 + s C_1 R_1 + 1} \quad (4)$$

- (iii). **Band pass:** If  $V_{in3} = V_{in2} = 0$  (grounded),  $V_{in1} = V_{in}$ , a second order band pass filter can be obtained. The voltage transfer function is given by

$$\frac{V_{out}}{V_{in}} = \frac{-s C_1 R_2}{s^2 C_1 C_2 R_1 R_2 + s C_1 R_1 + 1} \quad (5)$$

- (iv). **Band reject:** If  $V_{in3} = V_{in2} = V_{in1} = V_{in}$ , and  $R_1=R_2$ , a second order band reject filter can be obtained. The voltage transfer function is given by

$$\frac{V_{out}}{V_{in}} = \frac{s^2 C_1 C_2 R_1 R_2 + 1}{s^2 C_1 C_2 R_1 R_2 + s C_1 R_1 + 1} \quad (6)$$

- (v). **All pass:** If  $V_{in3} = V_{in2} = V_{in}$ ,  $V_{in1} = 2V_{in}$  and  $2R_1=R_2$ , a second order all pass filter can be obtained. The voltage transfer function is given by

$$\frac{V_{out}}{V_{in}} = \frac{s^2 C_1 C_2 R_1 R_2 - s C_1 R_1 + 1}{s^2 C_1 C_2 R_1 R_2 + s C_1 R_1 + 1} \quad (7)$$

Thus, the circuit is capable of realizing all filter functions. The circuit requires the minimum number of active and passive components. Our realisation is a purely single element and minimum passive components realization.

In all the cases, the resonance angular frequency ( $\omega_0$ ), bandwidth (BW) and the quality factor (Q) are given by:

$$\omega_0 = \left( \frac{1}{C_1 C_2 R_1 R_2} \right)^{\frac{1}{2}} \quad (8)$$

$$BW = \frac{\omega_0}{Q} = \frac{1}{C_2 R_2} \quad (9)$$

$$Q = \sqrt{\frac{R_2 C_2}{R_1 C_1}} \quad (10)$$

### III. NON-IDEAL ANALYSIS

To account for non ideal sources, two parameter  $\alpha$  and  $\beta$  are introduced where  $\alpha_i$  ( $i=1,2$ ) accounts for current tracking error and  $\beta_i$  ( $i=1,2,3,4,5,6$ ) accounts for voltage tracking error of the FDCCII. Incorporating the two sources of error onto ideal input-output matrix relationship of the modified FDCCII leads to:

$$\begin{bmatrix} V_{xa} \\ V_{xb} \\ I_{za} \\ I_{zb} \end{bmatrix} = \begin{bmatrix} 0 & 0 & \beta_1 & -\beta_2 & \beta_3 & 0 \\ 0 & 0 & -\beta_4 & \beta_5 & 0 & \beta_6 \\ \alpha_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\alpha_2 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} I_{xa} \\ I_{xb} \\ V_{y1} \\ V_{y2} \\ V_{y3} \\ V_{y4} \end{bmatrix} \quad (11)$$

from which the denominator of the non ideal voltage-mode multifunction filter is obtained as

$$D(s) = s^2 C_1 C_2 R_1 R_2 + s C_1 R_1 + \alpha_1 \alpha_2 \beta_1 \quad (12)$$

The non-ideal natural frequency ( $\omega_0$ ) and quality factor (Q) values are

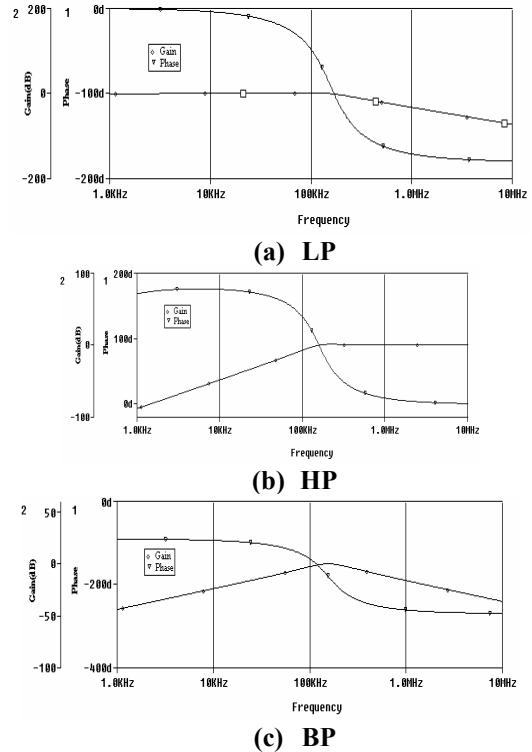
$$\omega_0 = \left( \frac{\alpha_1 \alpha_2 \beta_1}{C_1 C_2 R_1 R_2} \right)^{\frac{1}{2}} \quad (13)$$

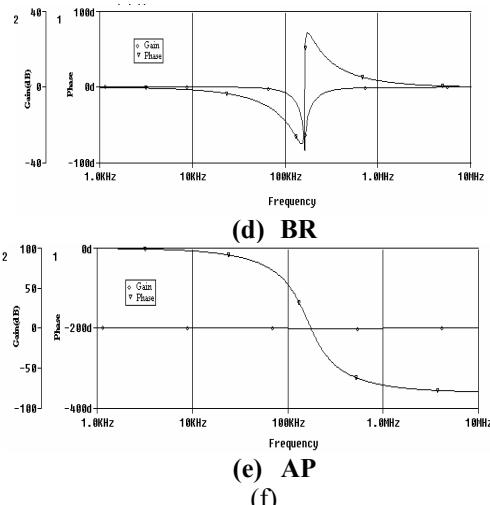
$$Q = \left( \frac{\alpha_1 \alpha_2 \beta_1 R_1 C_1}{R_2 C_2} \right)^{\frac{1}{2}} \quad (14)$$

This shows that  $\omega_0$  and Q for the ideal voltage-mode multifunction filter are affected by these two sources of error. Note that the value of  $\alpha_i$  and  $\beta_i$  are unity in the ideal case. The active sensitivity of  $\omega_0$  with respect to  $\alpha_1$ ,  $\alpha_2$ , and  $\beta_1$  is  $\frac{1}{2}$  and 0 with respect to  $\beta_2$ ,  $\beta_3$ ,  $\beta_4$ ,  $\beta_5$ ,  $\beta_6$ . The active sensitivity of Q with respect to  $\alpha_1$ ,  $\alpha_2$ ,  $\beta_1$  is  $\frac{1}{2}$ , and 0 with respect to  $\beta_2$ ,  $\beta_3$ ,  $\beta_4$ ,  $\beta_5$ ,  $\beta_6$ . The passive sensitivity of  $\omega_0$  and Q with respect to  $C_1$ ,  $C_2$ ,  $R_1$  and  $R_2$  is equal to  $\pm\frac{1}{2}$ . Thus the new circuits also possess good sensitivity performance.

### IV. SIMULATION RESULTS

The proposed voltage-mode multifunction filter has been simulated using PSPICE. The FDCCII was realized using CMOS implementation as shown in Fig. 2. and simulated using TSMC 0.35μm, Level 3 MOSFET parameters. The aspect ratio of the MOS transistors were chosen as in [12] and with the following DC biasing levels  $V_{dd}=5V$ ,  $V_{ss}=-5V$ ,  $V_{bp}=V_{bn}=0V$ ,  $I_B=7.0mA$ , and  $I_{SB}=6.0mA$ . The filter was designed with  $Q=1$  and cutoff frequency is 159.8 KHz by taking  $R_1=R_2=1\text{ K}\Omega$ , and  $C_1=C_2=1\text{ nF}$ .





**Figure 4.** Gain and phase responses of the Multifunction universal voltage mode filter (a) LP (b) HP (c) BP (d) BR (e) AP

Fig. 4 shows the simulated gain and phase responses of the voltage-mode multifunction filter outputs namely high-pass, band-pass, low-pass, band reject, and all-pass filters obtained from the circuit. The simulated pole frequency of 159 KHz is obtained as compared to ideal value 159.8 KHz. As can be seen there is a close agreement between theory and simulation.

## V. CONCLUSIONS

This paper presents a voltage-mode multifunction filter employing minimum number of component count i.e one active element and four passive components. Besides, the proposed circuit still offers the following advantages: (i) employ single active element, (ii) four passive components i.e two resistors and two capacitors, (iii) realization of all standard filter functions i.e low pass(LP), high pass(HP), band pass(BP), band reject(BR) and all pass(AP), and (iv) no need to employ inverting-type input signals. In Table 1, the main features of the proposed new circuit are compared with those of previous works. The circuit is verified through PSPICE simulation results.

Table 1. Performance parameters of recently reported voltage-mode filters

Circuits	Criteria			
	(i)	(ii)	(iii)	(iv)
The new circuit	Yes	Yes	Yes	Yes
Ref. 11 in 2008	Yes	Yes	No	Yes
Ref. 10 in 2008	No	No	Yes	No
Ref. 9 in 2006	No	Yes	No	No
Ref. 8 in 2003	Yes	No	No	Yes
Ref. 7 in 2001	No	Yes	Yes	No
Ref. 6 in 1996	No	No	Yes	No
Ref. 5 in 1999	No	No	Yes	No
Ref. 4 in 1996	No	No	Yes	No

## VI. REFERENCES

- [1]. Tomazou, and F.J. Lidgley, "Universal active filter using current conveyor," Electronics Letters, vol. 22, pp. 662 – 664, 1986.
- [2]. Higashimura, "Realization of voltage mode biquads using CCII's," Electronics Letters, vol. 27, pp. 1345- 1346, 1991.
- [3]. Higashimura, and Y. Fukui, "Universal filter using plus type CCII's," Electronics Letters, vol. 32, pp. 810-811,1996.
- [4]. C.M. Chang, and M.S. Lee, "Universal voltage mode filter with three inputs and one output using three current conveyors and one voltage followers," Electronics Letters, pp. 2112 – 2113,1994.
- [5]. C.M. Chang, and S.H. Tu, "Universal voltage mode filter with four inputs and one output using only two CCII+s," International Journal of Electronics, pp. 305 -309,1999.
- [6]. J.W. Horng, C.C. Tsai, and M.H. Lee, "Novel universal voltage mode biquad filter with three inputs and one output using only two current conveyors," International Journal of Electronics, pp. 543 – 546,1996.
- [7]. J.W. Horng, "High-input impedance voltage-mode universal biquadratic filter using three plus-type CCII's," International Journal of Electronics, vol. 8, pp. 465–475, 2004.
- [8]. R.K. Sharma, and R. Senani, "Multifunction CM/VM biquads realised with a single CFOA and grounded Capacitors," International Journal of Electronics and Communication (AEU), vol. 57, no.5, 301–308, 2003
- [9]. Kumar, and K. Pal, "High input impedance band pass, all pass and notch filters using two CCII's," HAIT Journal of Science and Engineering A, vol. 3, pp. 1-13, 2006.
- [10]. K. Pal, and M.J. Nigam, "A high input impedance multifunction filter using grounded capacitors." Journal of Active and Passive Electronic Devices, vol. 3, pp. 39-41, 2008.
- [11]. S. Maheshwari, "High performance voltage-mode multifunction filter with minimum component count," WSEAS Transactions on Electronics, vol. 5, pp. 244-249, 2008.
- [12]. A.A. El-Adway, A.M. Soliman, and H.O. Elwan, "A novel fully differential current conveyor and its application for analog VLSI," IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process, vol. 47, pp. 306-313, 2000.
- [13]. C.M. Chang, B.M. Al-Hashimi, C.I. Wang, and C.W. Hung, "Single fully differential current conveyor biquad filters," IEE Proc.-Circuits Devices Syst., vol. 150, pp. 394-398, 2003
- [14]. J.W. Horng, C.L. Hou, C.M. Chang, H.P. Chou, C.T. Lin, and Y. Wen, "Quadrature Oscillators with Grounded Capacitors and Resistors Using FDCCII's," ETRI Journal., vol. 28, pp. 486-494, 2006.
- [15]. S. Maheshwari, I.A Khan, and J. Mohan, "Grounded capacitor first order filters including canonical forms," Journal of Circuits Systems and Computers, vol. 15, pp. 289-300, 2006.
- [16]. J. Mohan, S. Maheshwari, and I.A. Khan, "Mixed-mode quadrature oscillators using single FDCCII," Journal of Active and Passive Electronic Devices., vol. 2, pp. 227-234 ,2007.

# Implementation of Real Time Scheduling in MATLAB

Mr. Vishal S.Vora- PG Studen, [tvsavora@aits.edu.in](mailto:tvsavora@aits.edu.in), Mr. Yagnesh N. Makawana- PG Student, [ynmakwana@aits.edu.in](mailto:ynmakwana@aits.edu.in)  
Mr. A. M. Kothari- PG Student, [amkothari@aits.edu.in](mailto:amkothari@aits.edu.in), Prof. A.C.Suthar- Prof. [acsuthar@yahoo.co.in](mailto:acsuthar@yahoo.co.in)  
Dept of E&C, CCET, Wadhwani

**Abstract-** This paper presents a Matlab based Scheduling toolbox TORSCHE (Time optimization of Resources, Scheduling). The toolbox offers a collection of data structures that allow the user to formalize various off-line and online scheduling problems. Algorithms are simply implemented as Matlab functions with fixed structure allowing users to implement new algorithms. A more complex problem can be formulated as an Integer Linear Programming problem or satisfiability of boolean expression problem. The toolbox is intended mainly as a research tool to handle control and scheduling co-design problems. Therefore, we provide interfaces to a real-time Matlab/Simulik based simulator TrueTime and a code generator allowing generating parallel code for FPGA.

## I. INTRODUCTION

### A. Tool Overview

TORSCHE (Time Optimization of Resources, SCHEduling) is a MATLAB-based toolbox including scheduling algorithms that are used for various applications such as high level synthesis of parallel algorithms or response time analysis of applications running under fixed-priority operating system. Using the toolbox, one can obtain an optimal code of computing intensive control applications running on specific hardware architectures. The tool can also be used to investigate application performance prior to its implementation. These values (e.g. the shortest achievable sampling period of the filter implemented on a given set of processors) can be used in the control system design process performed in Matlab/Simulik. The main contribution of the toolbox, which is built on well-known disciplines of the graph theory and operation research, is to make it easy to apply this type of reasoning to a wide range of problems. Many of them are combinatorial optimization problems, and as such they are Challenging from the theoretical point of view.

The toolbox offers a collection of Matlab routines that allow the user to formalize the scheduling problem, while considering appropriate configuration of resources (e.g. Field Programmable Gate Arrays (FPGA) based architecture [1] or micro controllers with real-time operating system [2]), task parameters (e.g. deadlines, release dates, preemption) and optimization criterion (e.g. makespan minimization, maximum lateness minimization, the task completion prior its deadline). The toolbox enables to solve these optimization and decision problems by their reformulation (e.g. to Integer Linear Programming (ILP) or satisfiability of boolean expression problem (SAT)) or to solve them directly while choosing appropriate scheduling algorithm. The input data of the problem instance are typically represented by a set of tasks, set of resources and optimization criterion. The output data of the optimization problems are typically

represented by a Gantt chart. The input data might be automatically generated from the problem description (e.g. equations of the filter algorithm) and output data, the schedule, may be used to automatically generate an implementation of embedded system (e.g. parallel code for dedicated processing units implemented on FPGA).

### B. Motivation

The toolbox is intended mainly as a research tool to handle control and scheduling co-design problems. The objective of these problems is to design a set of controllers and schedule them as real-time tasks, such that the overall control performance is optimized for given set of controlled systems and limited computational resources. In some cases, such optimization problem can be formulated analytically. Unfortunately, real applications are more complex and therefore design process cannot be fully automated. In such cases a simulation environment such as Matlab/Simulink presents excellent environment for rapid prototyping of new concepts, simulation and elaboration of design methodologies that are tailored to a specific class of applications and computational resources. For a given control algorithm and computational resources the toolbox makes it possible to derive such real-time parameters as sampling period and jitter. These real-time parameters are further used to derive the control performance (e.g. using TrueTime [3]) and to optimize the controller parameters or to choose another control algorithm and to repeat the design process.

### C. Related Work

The toolbox is mostly based on existing well-known scheduling algorithms. In part it contains our previous [4] and current research work [5]. It is very convenient platform to share ideas and tools among researchers and students. Several traditional off-line scheduling algorithms [6] and their extensions represent the basis of the toolbox. In additional, these algorithms can be simply used for scheduling of operations on specific hardware architectures, e.g. FPGAs with arithmetic modules [7]. On-line scheduling algorithms are based on proven approaches from real-time community [8],[9] and on the schedulability analysis for tasks with static and dynamic offsets [10]. In contrast to the MAST tool [11] built to support mainly timing analysis of realtime applications, TORSCHE is not as profound in this area, but covers also off-line scheduling algorithms and due to its implementation in Matlab it is suited to handle control and scheduling co-design problems. TORSCHE is focused on the schedule synthesis and

schedulability analysis. Therefore it is complementary to TrueTime, which is a Matlab/Simulink based simulator.

#### D. Outline

This paper is organized as follows: Section II presents the tool architecture and basic notation. Section III presents offline scheduling algorithm for the set of tasks with precedence constraints running on parallel identical processors. The problem of makespan minimization is solved via formulation to the satisfiability of boolean expressions problem (SAT). Section IV presents another off-line scheduling problem, cyclic scheduling aiming to find a periodic schedule with a minimum period. The next section describes on-line scheduling problems, that should be solved when the tasks are executed under real-time operating system based on the fixedpriority scheduler. In particular we show the schedulability analysis algorithm for tasks with offsets. All the algorithms are accompanied by illustrative examples including the use of the toolbox functions (for more details see the toolbox manual [12]). Section VI concludes the work.

## II. TOOL ARCHITECTURE AND BASIC NOTATION

TORSCHE is written in Matlab object oriented programming language and it is used in Matlab environment as a toolbox. Main objects are *Task*, *TaskSet* and *Problem*. Object *Task* is a data structure including all parameters of the task as processing time, release date, deadline etc. These objects can be grouped into a set of tasks with other related information as precedence constraints into a *TaskSet* object.

Object *Problem* is a small structure describing classification of deterministic scheduling problems in Graham and Błażewicz notation [6]. These objects are used as a basis that provide general functionality and make the toolbox easily extensible by other scheduling algorithms. In off-line scheduling problems, the task is given by the following parameters

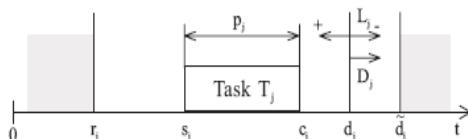


Fig. 1. Task parameters

*Processing time*,  $p_j$ , is time necessary for task execution (also called computation time).

*Release date*,  $r_j$ , is the moment at which a task becomes ready for execution (also called arrival time, ready time, request time).

*Deadline*,  $d_j$ , specifies a time limit by which the task has to be completed, otherwise the scheduling is assumed to fail.

*Due date*,  $d_j$ , specifies a time limit by which the task should be completed, otherwise the criterion function is charged by penalty.

*Weight* expresses the priority of the task with respect to other tasks (also called priority).

*Processor* specifies dedicated processors at which the task must be executed. Resulting schedule is represented by the following parameters:

- *Start time*,  $s_j$ , is the time when the execution of the task is started.
- *Completion time*,  $c_j$ , is the time when the execution of the task is finished.
- *Lateness*,  $L_j = c_j - d_j$ .
- *Tardiness*,  $D_j = \max\{c_j - d_j, 0\}$ .

The task is represented by the object data structure with the name *Task* in Matlab. This object is created by the command with the following syntax rule:

$t1 = \text{task}([\text{Name},] \text{ProcTime}, \text{ReleaseTime} \dots [\text{,Deadline},] \text{DueDate}, \text{Weight}, \text{Processor}])$

Command *task* is a constructor for object of type *Task* whose output is stored into a variable (in the syntax rule above it is variable  $t1$ ). Properties contained inside the square brackets are optional. The object *Problem* is used for classification of deterministic scheduling problems in Graham and Błażewicz notation.

This notation consists of three parts. The first part describes the processor environment, the second part describes the task characteristics of the scheduling problem as the precedence constraints, or the release time. The last part denotes an optimality criterion. An example of its usage is shown in the following code:

$\text{prob} = \text{problem('P|prec|Cmax')}$

Most of all algorithms use the following syntax:

$\text{tasksetWS} = \text{algorithmname}(\text{taskset}, \text{prob}, \text{processors}, \text{param})$

Where

- *tasksetWS* is the input *taskset* with an added schedule,
- *algorithmname* is the algorithm command name,
- *taskset* is the set of tasks to be scheduled,
- *prob* is the object of type *problem*,
- *processors* is the number of processors to be used,
- *param* denotes additional parameters, e.g. algorithm strategy etc.

## III. SCHEDULING ON PARALLEL IDENTICAL PROCESSORS

This section presents the SAT based approach to the scheduling problems. The main idea is to formulate a given scheduling problem in the form of CNF (conjunctive normal form) clauses (for more details see [13]). TORSCHE includes the SAT based algorithm for P|prec|Cmax problem, i.e. scheduling of tasks with precedence constraints on the set of parallel identical processors while minimizing the schedule makespan. is a function of Boolean variables in the form  $x_{ijk}$ . If task  $T_i$  is started at time unit  $j$  on the processor  $k$  then  $x_{ijk} = \text{true}$ , otherwise  $x_{ijk} = \text{false}$ . For each task  $T_i$ , where  $i = 1 \dots n$ , there are  $S \times R$  Boolean variables, where  $S$  denotes the maximum number of time units and  $R$  denotes the total number of processors.

The Boolean variables are constrained by the three following rules (modest adaptation of [14]): 1. For each task, exactly one of the  $S \times R$  variables has to be equal to 1. Therefore two clauses are generated for each task  $T_i$ . The first

guarantees having at most one variable equal to 1 (true): ( $\neg x_{11} \wedge \neg x_{21} \wedge \dots \wedge \neg x_{11} \wedge \neg x_{1R} \wedge \dots \wedge \neg x_{(S-1)R} \wedge \neg x_{SR}$ ).

The second guarantees having at least one variable equal to 1:  $(\exists i_1 \exists \dots \exists i_{(S-1)R} \exists x_{SR})$ . 2. If there is a precedence constraint such that  $T_u$  is the predecessor of  $T_v$ , then  $T_v$  cannot start before the execution of  $T_u$  is finished. Therefore,  $x_{ujk} \rightarrow ((\exists v_1 \dots \exists v_{(j+1)l} \exists x_{(j+1)l}) \dots \exists x_{(j+pu-1)l})$  for all possible combinations of processors  $k$  and  $l$ , where  $p_u$  denotes the processing time of task  $T_u$ . 3. At any time unit, there is at most one task executed on a given processor. For the couple of tasks with a precedence constraint this rule is ensured already by the clauses in the rule number 2. Otherwise the set of clauses is generated for each processor  $k$  and each time unit  $j$  for all couples  $T_u, T_v$  without precedence constraints in the following form:  $(x_{ujk} \rightarrow \neg x_{vjk}) \wedge (x_{ujk} \rightarrow \neg x_{v(j+1)k}) \wedge \dots \wedge (x_{ujk} \rightarrow \neg x_{v(j+pu-1)k})$ .

In the toolbox we use a *zChaff* [15] solver to decide whether the set of clauses is satisfiable. If it is, the schedule within S time units is feasible. An optimal schedule is found in iterative manner. First, the List Scheduling algorithm is used to find initial value of S. Then we iteratively decrement value of S by one and test feasibility of the solution. The iterative algorithm finishes when the solution is not feasible.

As an example we show a computation loop of a Jaumann wave digital filter [16]. Our goal is to minimize computation time of the filter loop, shown as directed acyclic graph in Fig. 2. Node in the graph represent the tasks and the edges represent precedence constraints. The nodes are labeled by the operation type and processing time  $p_i$ . We look for an optimal schedule on two parallel identical processors.

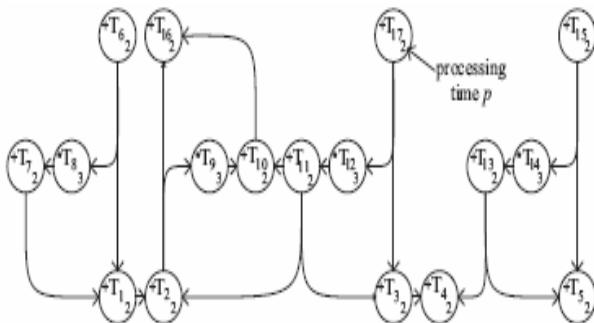


Fig. 2. Jaumann wave digital filter

Fig. 3 shows consecutive steps performed within the toolbox. First, we define the set of task with precedence Finally we plot the Gantt chart.

```

>> procTime = [2,2,2,2,2,2,2,3,3,2,2,3,2,3,2,2,2];
>> prec = sparse([
[6,7,1,11,11,17,3,13,13,15,8,6,2,9 ,11,12,17,14,15,2
,10],...[1,1,2,2 ,3 ,3 ,4,4 ,5 ,5 ,7,8,9,10,10,11,12,13,14,16,16],...
[1,1,1,1 ,1 ,1 ,1 ,1 ,1 ,1,1,1,1 ,1 ,1 ,1 ,1 ,1 ,1 ,1 ],...
17,17);
>> jaumann = taskset(procTime,prec);
>> jaumannSchedule
satsch(jaumann,problem('P|prec|Cmax'),2)
Set of 17 tasks
There are precedence constraints

```

```
There is schedule: SAT solver  
SUM solving time: 0.06s  
MAX solving time: 0.04s  
Number of iterations: 2  
>> plot(jaumannSchedule)
```

The *satsch* algorithm performed two iterations. In the first iteration 3633 clauses with 180 variables were solved as satisfiable for  $S = 19$  time units. In the second iteration 2610 clauses with 146 variables were solved with unsatisfiable result for  $S = 18$  time units. The optimal schedule is depicted in Fig. 4.

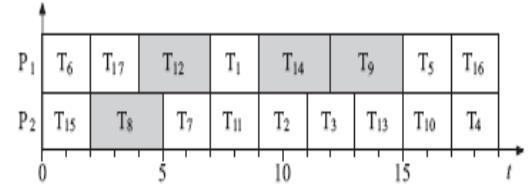


Fig. 4. The optimal schedule of Jaumann filter.

#### IV. CYCLIC SCHEDULING ON DEDICATED PROCESSORS

*Cyclic scheduling* deals with a set of operations (generic tasks) that have to be performed an infinite number of times [17]. This approach is also applicable if the number of loop repetitions is large enough. If execution of operations belonging to different iterations can interleave, the schedule is called *overlapped*. An overlapped schedule can be more effective especially if processors are pipelined hardware units or precedence delays are considered. The *periodic schedule* is a schedule of one iteration that is repeated with a fixed time interval called a *period* (also called *initiation interval*). The aim is then to find a periodic schedule with a minimum period [17]. As an example, we show a computation loop of a wave digital filter (WDF) [18] consisting of eight tasks. It is extended to five channels by assuming five clock cycles processing time of each task (i.e. single channels are shifted by one clock cycle). Fig. 5 shows the filter with corresponding processing times of operations executed using HSLA. The scheduling problem is to find a start time  $s_i(k)$  of every occurrence  $T_i$  [17]. arithmetic library on FPGA (input-output latency of ADD (MUL) unit is 9 (2) clock cycles, respectively [7]). Operations in a computation loop can be considered as a set of  $n$  generic tasks  $T = \{T_1, T_2, \dots, T_n\}$  to be performed  $K$  times where  $K$  is usually very large.

One execution of

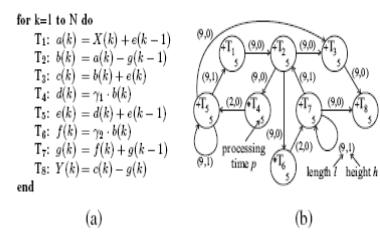


Fig. 5. (a) An example of a computation loop of wave digital filter (WDF).  
 (b) Corresponding data dependency graph  $G$  of WDF.

Data dependencies of this problem can be modeled by a directed graph G. Each task (node in G) is characterized by the processing time  $p_i$ . Edge  $e_{ij}$  from the node  $i$  to  $j$  is weighted by a couple of integer constants  $l_{ij}$  and  $h_{ij}$ . Length  $l_{ij}$  represents the minimal distance in clock cycles from the start time of the task  $T_i$  to the start time of  $T_j$  and is always greater than zero (corresponds to input-output latency in our example). On the other hand, the height  $h_{ij}$  specifies the shift of the iteration index (dependence distance) related to the data produced by  $T_i$  and read (consumed) by  $T_j$ . Assuming a *periodic schedule* with the *period* w (i.e. the constant repetition time of each task), each edge  $e_{ij}$  in graph G represents one precedence relation constraint  $s_j - s_i \geq l_{ij} - w \cdot h_{ij}$ , (1) where  $s_i$  denotes the start time of task  $T_i$  in the first iteration. Fig. 5(a) shows the data dependence graph of the computation loop shown in Fig. 5(b). When the number of processors m is restricted, the cyclic scheduling problem becomes NP-complete [17]. Unfortunately, in our case the number of processors is restricted and the processors are dedicated to execute specific operations.

In the toolbox we formulate the scheduling problem as a problem of Integer Linear Programming (ILP). For more detail about the scheduling algorithm see [12], [4]. The schedule of the WDF example, obtained as outlined in Fig. 7, is shown in Fig. 6. The toolbox is interconnected with Matlab/Simulink based simulator TrueTime [3] which facilitates cosimulation of realtime task execution and continuous plant dynamics. Arbitrary schedule can be directly transformed to a model used by TrueTime. This function allows to design complex control systems and simulate influence of external events on the schedule. Furthermore, the scheduling results can be used to generate parallel code in Handel C [19] for FPGA. It allows to design time critical algorithms especially for FPGA as is shown in WDF example in this section. The toolbox provides to designer full control over the scheduling algorithm.

```
>> load wdf
>> UnitProcTime=[5 5];
>> UnitLatency=[9 2];
>> G=cdfg2LHgraph(wdf,UnitProcTime,UnitLatency);
>> t=taskset(G);
>> prob=problem('m-DEDICATED');
>> schoptions=schoptionsset('ilpSolver','glpk',...
'cycSchMethod','integer','varElim',1);
>> taskset_sch=mdcycsch(t,prob,1,schoptions)
Set of 8 tasks
There are precedence constraints
There is schedule: MONOCYCSCH - ILP based algorithm
Tasks period: 31
Solving time: 0.094s
Number of iterations: 5
>> plot(taskset_sch)
```

Fig. 7. Solution of a cyclic scheduling problem in the toolbox.

## V. REAL-TIME SCHEDULABILITY ANALYSIS

Real-Time scheduling is usually used in Real-Time operating systems for scheduling a set of periodic

tasks. Simple scheduling algorithms such as *fixed-priority scheduling* are usually used since they need to be executed online. Given a system comprising of a set of real-time tasks and a scheduling algorithm, a verification algorithm can determine whether all the tasks in the system meet their real-time constraints deadlines). Besides basic *response-time analysis* for rate monotonic algorithm [9], TORSCHE contains a more advanced technique: *schedulability analysis for tasks with offsets*. This technique was firstly introduced by Tindell in [20], and later further formalized and enhanced by Palencia and Harbour in [10]. In both papers, authors designed exact algorithm for this NP-hard problem (determining response times of tasks in the system) as well as polynomial approximate analysis that finds upper bound to the task response times. Currently, TORSCHE contains only the exact algorithm. Note: This section uses notation different from the rest of this paper. The reason is that this notation is common in real-time community.

### A. Computational Model

The real-time system considered for analysis is composed of tasks executing in the same processor, but the analysis can be easily extended for multiprocessor systems. Tasks are grouped to *transactions*. Each transaction  $\Gamma_i$  is activated by a periodic sequence of external events with period  $T_i$ . The relative phasing between the different external events is arbitrary. Each task will be identified with two subscripts: the first one identifies the transaction to which it belongs and the second one the position that the task occupies within the tasks in its transactions, when they are ordered by increasing offsets. Task  $t_{ij}$  is activated (released) when a relative time—called the *offset*,  $\Phi_{ij}$ —elapses after the arrival of the external event. The offset can be static or dynamic. Dynamic offsets are represented by a value of jitter  $J_{ij}$ , which specifies the length of the interval in which a task can be activated.  $C_{ij}$  is the worst-case execution time. It is assumed that each task has its unique priority and all the tasks are scheduled using a preemptive fixed priority scheduler. If tasks synchronize for using shared resources in a mutually exclusive way they will be using a hard real-time synchronization protocol such as the priority ceiling protocol [9]. The *response time* of each task  $t_{ij}$  is defined as a difference between its completion time and the instant at which the associated external event arrived. The worst-case response time is denoted by  $R_{ij}$ . Each task may have associated global deadline  $D_{ij}$ , which is also relative to the arrival of external event. A system is said to be schedulable if for each task  $R_{ij} \leq D_{ij}$ .

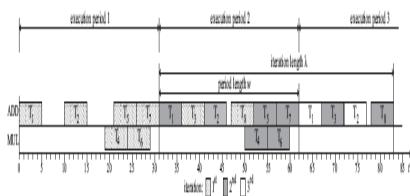


Fig. 6. The optimal schedule of SJDF benchmark on RSLA ( $n=3$ ). Transaction overlap is even significant due to operations pipelining that is not visible in this figure.

Fig. 8 shows an example of a real-time system. The horizontal axis represents time. Down-pointing arrows represent periodic external events, filled boxes represent task execution. Dashed lines below each transaction axis represent task jitter values. The system in this figure is composed of three transactions.

The first and the third ones group two tasks, the second one groups three tasks. No blocking is considered in this example and tasks have assigned priority corresponding to their subscript. The smaller the number the higher the priority. Note that task  $t_{12}$  has its offset greater than completion time of the previous task in the transaction and that tasks not initiating the transactions has non-zero jitter (dotted line). In this way, offsets and jitters can be used to represent self-suspending tasks (e.g. tasks calling sleep() OS service or waiting for data from a periphery). *B. Exact Response-Time Analysis Algorithm* The calculation of the exact worst-case response-time (WCRT) for any task in the system is NP-hard problem [21] and therefore the complexity of the exact algorithm is exponential to the size of the system. In [10] and [22] approximate methods that calculates an upper bound to the WCRT are developed. These polynomial time algorithms are based on simplification of the exact algorithm. TORSCHE currently implements only the exact algorithm.

To find the worst-case response time of a task under analysis ( $\tau_{ab}$ ), it is necessary to build the worst-case scenario for this task. Finding this scenario rests in finding such a combination of higher priority tasks having the highest contribution to  $\tau_{ab}$  response time. The time when this combination occurs is called the *critical instant* of task  $\tau_{ab}$ . In the case where all tasks are independent (without offsets), the critical instant occur at the time when all higher priority tasks are activated simultaneously with  $\tau_{ab}$ . This no longer holds for tasks with offsets, as it might be impossible for some sets of task to be activated at the same time.

The complexity of the algorithm rests in the fact that we have to explore all possible combinations of tasks from each transaction and find which combination produces the worstcase response time. Fig. 9 shows the scenario of the system from Fig. 8, for which the response-time of task  $\tau_{21}$  reaches its worstcase value. This situation happen when tasks  $\tau_{00}$ ,  $\tau_{12}$  and  $\tau_{21}$  are activated simultaneously after being delayed by their maximum jitter. The phasing of external events as well as activation times of all tasks in this scenario are depicted in the three top-level axes of the picture. The last axis shows the schedule as produced by the fixed-priority scheduler. The critical instant is at time 0 of this schedule and

the length of busy period  $L_{ab} = 150$ . For this scenario, (the worst-case) response-time  $R_{21} = L_{21} + \Phi_{21} + J_{21} = 200$ .

### C. Matlab Session Example

The model of a real-time system is entered into Matlab by creating appropriate *Task* and *Transaction* object instances. The response-time analysis can be performed by calling *taskoffsanal* function. Fig. 10 shows the steps needed to perform response-time analysis within TORSCHE. Computed response times are listed as *respTime* values. Other toolbox functions can be used to automatically draw figures like the ones in this section.

```
>> t00=task(10, 0);
>> t01=task(10,[10 15], 100);
>> G0=transaction([t00 t01], 100);
>> t10=task(25,0);
>> t11=task(10, [25 30]);
>> t12=task(20, [70 80], 100);
>> G1=transaction([t10 t11 t12], 130);
>> t20=task(30, 0);
>> t21=task(35, [30 50], 250);
>> G2=transaction([t20 t21], 300);
>> system =[G0 G1 G2]; setprio(system, 'rm');
>> taskoffsanal(system)
System:
Transaction: per=100.0
Task t00: prio=7 wcet=10.0 o=0 j=0 respTime=10.0
Task t01: prio=6 wcet=10.0 o=10.0 j=5.0 respTime=25.0
Transaction: per=130.0
Task t10: prio=5 wcet=25.0 o=0 j=0 respTime=45.0
Task t11: prio=4 wcet=10.0 o=25.0 j=5.0 respTime=60.0
Task t12: prio=3 wcet=20.0 o=70.0 j=10.0 respTime=120.0
Transaction: per=300.0
Task t20: prio=2 wcet=30.0 o=0 j=0 respTime=145.0
Task t21: prio=1 wcet=35.0 o=30.0 j=20.0 respTime=200.0
```

Fig. 10. Offset-based response-time analysis in a Matlab session

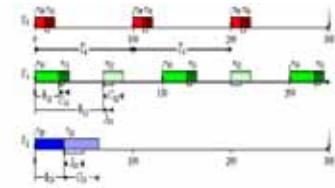


Fig. 8. Graphical representation of the real-time system.

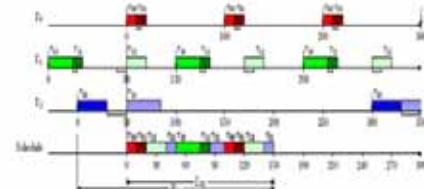


Fig. 9. The schedule that produce the worst-case response time for  $t_{21}$ .

## VI. CONCLUSIONS AND FUTURE WORK

This paper presents TORSCHE Scheduling Toolbox for Matlab for off-line and on-line scheduling. The toolbox includes scheduling algorithms, that are used for various applications as high level synthesis of parallel

algorithms or response time analysis of applications running under fixed priority operating system. The main objective of this project is to facilitate design of real-time applications mainly in control domain where the Matlab is frequently used. In this paper, we have shown the applicability on three examples. In the future work we will focus on interconnections to another designs tools and simulators and we will incorporate new scheduling algorithms. Actual version of the toolbox is freely available.

## VII. REFERENCES

- [1]. J. Kadlec, *FloatPipe1v35 Modules for Virtex and Virtex 2*, February 2004. <http://www.celoxica.com>.
- [2]. L. Waszniowski and Z. Hanz'alek, "Analysis of OSEK/VDX based automotive applications," in *IFAC Symposium on Advances in Automotic Control, Salerno*, Elsevier, April 2004.
- [3]. M. Andersson, D. Henriksson, and A. Cervin, *TrueTime 1.3 Reference Manual*. Lund University, Sweden, 2005. <http://www.control.lth.se/ldan/trutime/>.
- [4]. P. Šoūcha, Z. Pohl, and Z. Hanz'alek, "Scheduling of iterative algorithms on FPGA with pipelined arithmetic unit," in *10th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 2004)*, Toronto, Canada, 2004.
- [5]. A. Heřmánek, J. Schier, P. Šoūcha, and Z. Hanz'alek, "Optimization of finite interval CMA implementation for FPGA," in *In IEEE 2005 Workshop on Signal Processing Systems (SIPS'05)*, pp. 75–80, Piscataway:IEEE, 2005.
- [6]. J. Błažewicz, K. Ecker, G. Schmidt, and J. Weißlarz, *Scheduling Computer and Manufacturing Processes*. Springer, second ed., 2001.
- [7]. R. Matoušek, M. Tichý, A. Z. Pohl, J. Kadlec, and C. Softley, "Logarithmic number system and floating-point arithmetics on FPGA." Field-Programable Logic and Applications: Reconfigurable computing Is Going Mainstream. Lecture notes in Computer Science A 2438, Springer, Berlin, 2002.
- [8]. G. C. Buttazzo, *Hard Real-Time Computing Systems*. Kluwer Academic Publishers, second ed., 2005.
- [9]. J. W. S. Liu, *Real-Time Systems*. Upper Saddle River, NJ, USA: Prentice Hall, 2000.
- [10]. J. C. Palencia and M. G. Harbour, "Schedulability analysis for tasks with static and dynamic offsets," in *Proceedings of the 19th Real Time Systems Symposium*, pp. 26–37, IEEE Computer Society Press, December 1998.
- [11]. "MAST (Modeling and Analysis Suite for Real-Time Applications)." <http://mast.unican.es/>.
- [12]. M. Kutil, P. Šoūcha, M. Sojka, and Z. Hanz'alek, *TORSCHE:Scheduling Toolbox Manual*, February 2006. <http://rtme.felk.cvut.cz/scheduling-toolbox/>.
- [13]. Y. Crama and P. L. Hammer, "Boolean functions: Theory, algorithms and applications," 2006. <http://www.rogp.hec.ulg.ac.be/Crama/Publications/BookPage.html>.
- [14]. S. O. Memik and F. Fallah, "Accelerated SAT-based Scheduling of Control/Data Flow Graphs," in *ICCD*, pp. 395–400, IEEE Computer Society, 2002.
- [15]. M. W. Moskewicz, C. F. Madigan, Y. Zhao, L. Zhang, and S. Malik, "Chaff: Engineering an Efficient SAT Solver," in *Proceedings of the 38th Design Automation Conference (DAC'01)*, 2001.
- [16]. S. H. de Groot, S. Gerez, and O. Herrmann, "Range-chart-guided iterative data-flow graph scheduling," *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, vol. 39, pp. 351–364, 1992.
- [17]. C. Hanen and A. Munier, "A study of the cyclic scheduling problem on parallel processors," *Discrete Applied Mathematics*, vol. 57, pp. 167–192, February 1995.

# Associative Memory Neural Network Algorithm and its Implementation on Embedded System

Parul Goyal- Assistant Professor, Department of Electronics & Communication Engineering, Dehradun Institute of Technology Dehradun, Uttarakhand, India , Email parulgoya2007@yahoo.com

Dr. S.C. Gupta, Professor & Dean PG, Department of Electronics & Communication Engineering Dehradun Institute of Technology, Dehradun

**Abstract-**This paper describes Associative Memory Neural Network algorithms and its implementation on FPGA (Field Programmable Gates Arrays) and its applications in image pattern recognition systems. An associative memory is a content-addressable structure that maps specific input representations to specific output representations. It is a system that "associates" two patterns (X, Y) such that when one is encountered, the other can be recalled. In the design, learning and recognizing algorithms for the neural network are implemented by using VHDL Hardware Description Language. FPGA is used for implementation because they can reduce development time greatly, ease of fast reprogramming, low price, flexible architecture and permitting fast and non expensive implementation of the whole system. The architecture was evaluated as image recognizing system.

## I. INTRODUCTION

Associative memory (AM) models form a class of artificial neural networks characterized by their information storage and recall (error correcting) capabilities. They possess the features of distributive data storage and parallel information flow which cause them to be robust with respect to malfunctions of individual devices, as well as be computationally efficient. The origin of associative memories is derived in the correlation matrix memory proposed by Kohonen. The bidirectional associative memory (BAM) is basically the generalized Hopfield network model developed by Kosko Concepts, definitions, learning algorithms for associative memories and applications are studied.

In recent years, FPGA-based hardware systems have been used extensively for developing coprocessors, custom computing machines, and fast prototyping platforms. Due to their reconfigurable nature, FPGAs can implement many different functions at different times. Likewise, systems can be made scalable. Hardware realization of NNs is an interesting issue. There are many approaches to implement NNs. The FPGA is a very useful device to realize a specific digital electronic circuit in diverse industrial fields. Hikawa realizes an NN with on-chip BP learning using a field-programmable gate array (FPGA). Some hardware implementations for neural network used in different applications are reported.

The purpose of this work is to make a hardware implementation of an associative memory neural network using (FPGA) for image pattern recognizing applications.

## II. BIDIRECTIONAL ASSOCIATIVE MEMORY

In the design is used a bidirectional associative memory, whose structure is shown in Fig. 1. The diagram shows this

neural network is composed by two layers: the input layer X and the output layer Y, both have specific number of neurons.

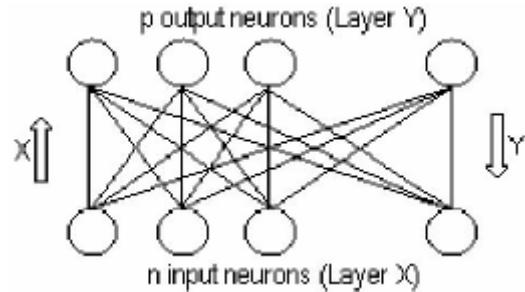


Figure 1 Bidirectional associative memory (BAM)

Each neuron of the layer X is connected with every neuron of the layer Y; there are not internal connections between neurons of the same layer. Vector patterns  $X_1, X_2, \dots, X_m$  are stored in a matrix memory. Input pattern X is presented to the memory by performing the multiplication  $XM$  and some subsequent non-lineal operation, such as thresholding, with resulting, output Y. Y is either accepted as the recollection or feedback into  $M^T$ , which produce  $X'$ , and so on. X stable memory will eventually produce a fixed output  $X_f$ .

In this design, the bidirectional associative memory is used as one-shot memory, X is presented to the correlation matrix M, Y is output, and the process is finished. Theoretically, the correlation matrix M is obtained by using the following equation:

$$M = \sum_i X_i^T Y_i \quad (1)$$

where  $X_i^T$  is the input pattern (input image in this work) and  $Y_i$  is the associated output vector. According to Kosko [4], it is better on average to use bipolar  $\{-1, 1\}$  coding than binary  $\{0, 1\}$  coding for hetero associative pairs  $(A_i, B_i)$ . In fact,  $X_i$  and  $Y_i$  are the bipolar version of  $A_i$  and  $B_i$ , respectively. The recognizing process is described through the following equation:

$$Y_i = \sum_j M_{ij} X_j \quad (2)$$

where M is the correlation matrix, X is the input pattern (input image) and Y is the output vector. Because the result in equation (2) is an integer quantity, a non-linear operation must be carried out in order to obtain the binary patterns of output vector. A step function is employed with threshold value 0:

$$\begin{aligned} B_i = 1 \text{ si } Y_i > 0 \\ B_i = 0 \text{ si } Y_i < 0 \end{aligned} \quad (3)$$

where B is the output binary pattern that identifies the image presented to the neural network.

### III. LEARNING ALGORITHM ARCHITECTURE

The neural network must learn all associations that will be stored. This algorithm is developed in six blocks (see Figure 2) and each block is designed using VHDL.

As Fig. 2 shows, it is necessary to store the patterns A and B before teaching the associative memory, where the pattern A corresponds to the input image that the net must learn and the pattern B is its associated output vector. Therefore, the image that will be analyzed is captured from a video camera and stored in a RAM memory; the pattern B is stored in a register.

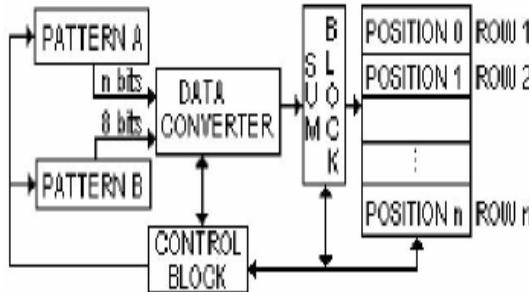


Figure 2 Learning algorithm block diagram

The acquired input image is a gray image with pixel values encoded between 0 and 255, which are converted in binary pixel. Finally, the obtained pattern A in the algorithm becomes a image matrix of 24x36 binary pixels and the pattern B becomes a vector of 8 binary elements, both are presented to the neural network and correspond to one association that it must learn. Each binary association pattern is transformed into bipolar pattern by using the following functions:

$$X = 2A - 1; Y = 2B - 1 \quad (4)$$

where A, B are the binary information, and X, Y are the bipolar information. This process is carried out by the "DATA CONVERTER" block shown below in Figure 2.

The "SUM" block produces the correlation matrix that contains all information of the previously learned associations and it is stored in a RAM memory with 32 bits wide, where each one of its elements uses four bits, one represents the sign of the element (the most significant bit - msb) and the others its absolute value (the three least significant bits). The "CONTROL" block synchronizes all the process in the learning stage through control signals that active each block in the system.

### IV. RECOGNIZING ALGORITHM ARCHITECTURE

When the neural network is learned, the system is ready to identify and to recognize the associations stored in the correlation matrix. The structure and block diagram of the recognizing algorithm is described in Figure 3, which is able to

make parallel processing of a whole row of the matrix M and sequential processing of each pixel of the input image. The vector b (b<sub>1</sub> ... b<sub>8</sub>) is the result of presenting the input image or pattern X to the matrix M by performing the multiplication XM and some subsequent non linear operation such as, thresholding. Findings of this process are obtained by "Multiplier 1 ... Multiplier 8" and "Adder 1... Adder8" blocks in Fig. 3. Similarly, the "CONTROL" block produces all signals to active each subsystem in order to generate the correct sequence of working and to obtain the pattern associated to the input image.

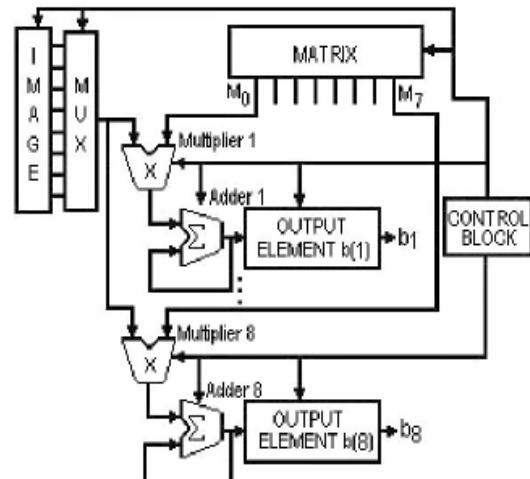


Figure 3 Recognizing Process block diagram

### V. FPGA IMPLEMENTATION

Number of Slices	1344 OUT OF 2352	57%
Number of Slice Flip Flops	871 OUT OF 4704	18%
Number of 4 Input LUTs	2421 OUT OF 4704	51%
Number of 4 Bonded IOBs	60 OUT OF 144	41%
Number of TBUTs	47 OUT OF 2352	1%
Number of BRAMs	9 OUT OF 14	64%
Number of GCLKs	1 OUT OF 4	25%
F (MHZ)	50	

Table 1 Hardware synthesis results - Resource utilization terms of FPGA area using Xilinx Spartan 2, XC2S200 device.

The system has been mapped to a Xilinx Spartan 2, XC2S200 device through Digilab 2 (D2) development board, and the hardware architectures were implemented by using VHDL in Xilinx ISE software which provides several tool for synthesizing the design, configuration techniques, performance analyses, including resource, speed, and power consumption. The resource utilization by the system for the FPGA, XC2S200 is shown in Table 1.

Table 2 Findings for VHDL design simulation.

The implementation of neural network VHDL model required 1344 Slices equivalent 57% of the resources on the FPGA device for recognizing image of 36x24 binary pixels. Therefore, 864 input neurons and 8 output neurons would fit on the FPGA.

## VI. SIMULATION RESULTS

The functional correctness of the whole system was verified including simulations and tests of each module. To evaluate the results of the neural network module implemented in VHDL, the algorithms are analyzed, synthesized and simulated with input matrix of 4x8 binary pixels, and associated output vector of 8 elements (input layer with 32 neurons and output layer with 8 neurons). Figure 4 presents nine associations that were used for learning the neural network.

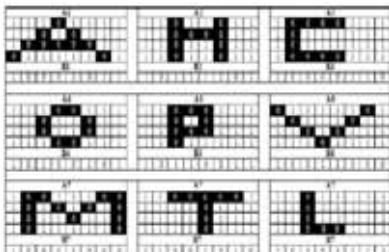


Figure 4 Learning Patterns- associations for VHDL design simulation.

Fists, associations ( $A_1, B_1$ ) and ( $A_2, B_2$ ) were employed to prove and simulated the design both theoretically and experimentally. The simulation time diagram for learning process is shown in Figure 5, in which the correlation matrix  $M$  is obtained and stored on the RAM memory. Here, three first memory addresses are read in order to verify their contents, which store the rows 1, 2 and 3 of the matrix  $M$ , respectively. Once the learning process is finished, the recognizing is proved. Figure 6 illustrates the time simulation diagram when the input image is presented to the neural network for recovering its associated vector. The input matrix (patter A) is introduced like a row vector, beginning from the first row until the last one. After several clock cycles is obtained the output pattern. Similarly, the number of associations for learning the neural network was increased using the patterns dataset shown below in Figure 4. The input patterns are also corrupted with noise by changing some pixels in the ideal input matrix in order to evaluate the performance of the associative memory for recognizing patterns with some distortions. Each image was modified by changing four pixel of its original matrix. After realizing many tests and simulations, the obtained results are summarized in table 2.

No. Learned Patterns	No. Recognized Patterns		No. Recognized Patterns with Noise	
	Quantity	%	Quantity	%
6	6	100	6	100
7	7	100	6	85.71
8	7	87.5	6	75.0
9	7	77.8	6	66.67
10	7	70	6	60

Performance for the associative memory recognizing process using input matrix of 4x8binary elements

For the FPGA implementation, the number of neurons is increased to 864 in the input layer due to the fact that input images have size of 24x36 binary pixels. Although the image pixels can possibly change by external noise in the acquisition process when the image is acquired from a camera, corrupted images by any distortion were also used to evaluate the performance of the system with distorted patterns. One of the images employed to prove the mapped FPGA design after its acquisition and storing process is illustrated in Fig. 7, which also shows the same image corrupted by a little distortion. Obtained results of performance for the system implemented on FPGA device can be seen in table 3.

No. Input images	No. Recognized Images		No. Recognized Images with Noise	
	Quantity	%	Quantity	%
6	6	100	6	100
7	6	85.71	5	71.42
8	6	75	5	62.5
9	6	66.67	5	55.5
10	6	50	5	50

Table 3 Findings of implemented FPGA design. Performance for the associative memory recognizing process using stored image of 24x36 binary pixels

The learning time of one association with input image to the neural network of 24x36 pixels binary and vector of 8 elements was 3821ns. The system can recover a pattern stored in 2592 FPGA clock cycles, it mean, 51.84 ps. All times are calculated with 50 MHz clock frequency.

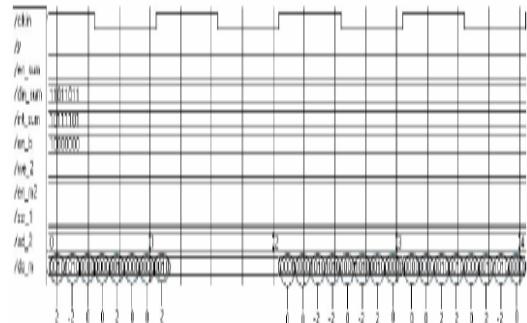
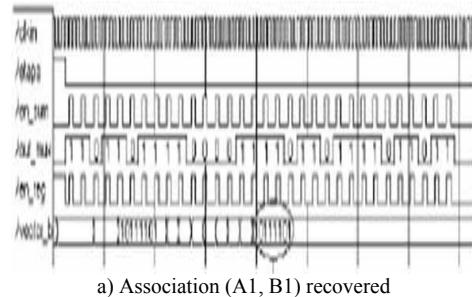
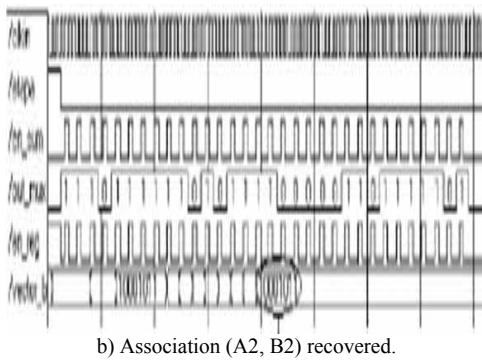


Figure 5 Learning stage time simulation diagram the "do\_m" signal shows the data contained for the memory that store the weight matrix and the "ad\_2" signal correspond to the memory address.





b) Association (A2, B2) recovered.

Figure 6 Associative memory timing diagram for recognizing stage The "out\_mux" signal is the pattern A presented as row vector, and the "vector\_b" signal is the pattern B recovered. The other signals are used for internal process of the VHDL design.

- [5] F. Daniel, G. Ramiro, F. Roberto, "Neuro FPGA Implementing Artificial Neural Networks on Programmable Logic Devices". IEEE Proceedings of the conference on Design, automation and test in Europe, vol. 3. Feb. 2004.
- [6] G. Wall, F. Iqbal, J. Isaacs, L. Xiuwen, S. Foo, "Real time texture classification using field programmable gate arrays". Applied Imagery Pattern Recognition Workshop, 2004, Proceedings, 33rd pp.130 - 135 Oct. 2004
- [7] S. M. Fakhraie, "Scalable closed-boundary analog neural networks," IEEE Trans. Neural Network., vol. 15, pp. 492-504, Mar. 2004. Digital Signal Processing A practical approach by Emmanuel C. Ifeachor & Barrie W. Jervis, 2004, Pearson Education
- [8] L. M. Reyneri, "Implementation issues of neuro-fuzzy hardware: going toward HW/SW codesign," IEEE Trans. Neural Network., vol. 14, pp. 176-194, Jan. 2003.
- [9] "A new digital pulse-mode neuron with adjustable activation functions" IEEE Trans. Neural Network, vol. 14, pp. 236-242, Jan. 2003.
- [10] Digital Image Processing Second Edition by Rafael C. Gonzalez & Richard E. Woods, 2002
- [11] Yingquan Wu and B. Stella, "An Efficient Learning Algorithm for Associative Memories," IEEE Trans. Syst., Neural Network, vol. 11, pp. 859-866, Sep. 2000.

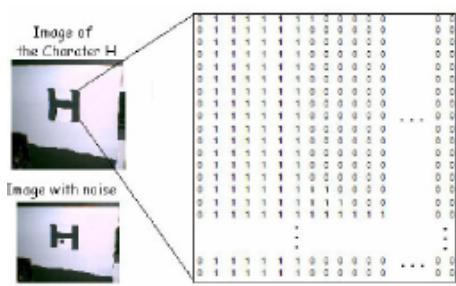


Figure 7 a) Acquired and stored image, and corrupted image with noise

## VII. CONCLUSIONS

Construction solution for implementation of associative memory neural network using FPGA is described. In every case, the performance of the implemented system is lower when the number of associations is increased. This work can be useful for applications that involve recognizing of small objects from hardware architectures as robotic applications. In this work, the developed hardware architecture is used as image recognizing system but it is not only limited to this applications, it mean, the design can be employed to process other type of signal.

Future scope is to extend the design for processing applications using bigger images as well as to implement others learning algorithms for improving the performance of the neural network.

## VIII. REFERENCES

- [1] Embedded Systems, Architecture, Programming and Design by Raj Kamal, 2008, Tata Mc Graw – Hill Publishing Company Limited
- [2] Embedded System Design, A unified Hardware/Software Introduction by Frank Vahid/Tony Givargis, 2007. Wiley India (P) Ltd
- [3] Fundamentals of Neural Networks Architecture, Algorithms, and Applications by Laurene Fausett, 2005, Pearson Education
- [4] Y. Maeda, M. Wakamura. "Simultaneous perturbation learning rule for recurrent neural networks and its FPGA implementation," IEEE Trans. Neural Network, vol. 16, Issue 6, pp. 1664- 1672. Nov. 2005.

# Application of DSP in Image Processing

Anu Suneja- Lecturer, Department of Computer Applications, Chitkara Institute of Engineering and Technology  
Rajpura, Punjab., Email id: [anusuneja3@gmail.com](mailto:anusuneja3@gmail.com)

**Abstract-** This paper describes an application of digital signal processors in the field of image processing. As Images are intensity signals reflected from the source object on which light falls, the operations that can be applied on digital signals can similarly be applied on images for their processing. Like in DSP, Signals are filtered, sampled and quantized, in Image processing before extracting information from images they have to be filtered sampled and digitized.

**KeyWords :-** Signal, Filtering, Sampling, Quantization.

## I INTRODUCTION

Very wide application areas of DSP have become available due to availability of high speed DSP processors. These processors have provided great storage capability and supporting tools. Image processing is one of vast application areas of DSP. Like most of signals measured as parameter over time, images are measured as parameter over space. Special characteristics of Image processing have made it as a separate subgroup within DSP. <sup>[1],[2]</sup>Image processing is concerned with manipulation of 2-D dataset using digital computer. Images are 2-D function  $f(x, y)$  where  $(x, y)$  are spatial co-ordinates and amplitude of  $f$  at point  $(x, y)$  is its intensity/gray level.

<sup>[3]</sup>Objective of image processing includes improvement of visual appearance of images for human, extracting features of image and take some decision on the basis of extracted information. To achieve this objective, DSP techniques can be used for processing of digital images.

The fundamental requirement of DSP is signal sampling and then its quantization. After quantization digitized signal is given as input to

DSP. Similarly we can represent an image in frequency or spatial domain and apply DSP techniques filtering, sampling and Quantization before processing of image.

## II DIGITAL SIGNAL FILTERING

<sup>[4]</sup>Digital signal processing allows the inexpensive construction of a wide variety of filters. The signal is sampled and an analog to digital converter turns the signal into a stream of numbers. A computer program running on a CPU or a specialized DSP (or less often running on a hardware implementation of the algorithm) calculates an output number stream. This output can be converted to a signal by passing it through a digital to analog converter.

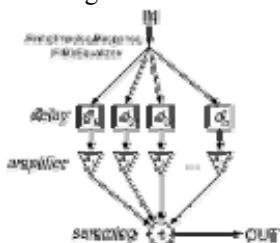


Fig 1: A finite impulse response filter

There are problems with noise introduced by the conversions, but these can be controlled and limited for many useful filters. Due to the sampling involved, the input signal must be of limited frequency content or aliasing will occur.

## III DIGITAL IMAGE FILTERING

<sup>[5]</sup>Digital images can be processed in a variety of ways. The most common one is called filtering and creates a new image as a result of processing the pixels of an existing image. Each pixel in the output image is computed as a function of one or several pixels in the original image, usually located near the location of the output pixel. If the function used does some kind of interpolation (eg. linear, cubic or gaussian), then the result will look smoother than the original, but care needs to be taken that the output values are not computed from too many input pixels, or the resulting image may get blurred. The most common purpose for this interpolation is antialiasing.

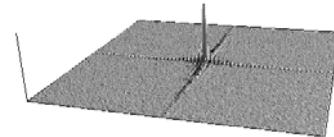


Fig 2: Blurred image (before filtering)

Applying a low pass filter in the frequency domain means zeroing all frequency components above a cutoff frequency. This is similar to what one would do in a 1 dimensional case except now the ideal filter is a cylindrical "can" instead of a rectangular pulse.

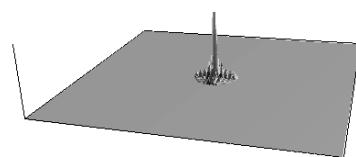


Fig 3: Filtered image

## IV DIGITAL SIGNAL SAMPLING

<sup>[6]</sup>Analog signals, in general, are continuous in time. In digital signal processing, we do not use the whole analog signal but replace it by its amplitudes taken at regular intervals. This is sampling. The problem is we must sample the signal so that the samples represent correctly the signal, i.e. from the samples we can reconstruct the original analog signal perfectly.

*Sampling of continuous-time signals:*

Sampling a continuous-time signal turns it into correspond discrete-time signal so that it can be processed on digital systems. Actually, the sampling is followed by two other operations, quantization and binary encoding. In reality, the analog-to-digital converters (abbreviated ADC or A/D) do all the three steps.

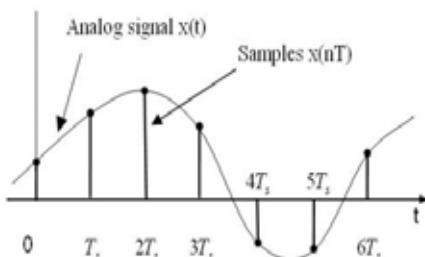


Figure 4: Sampling of signal at sampling interval (period) T

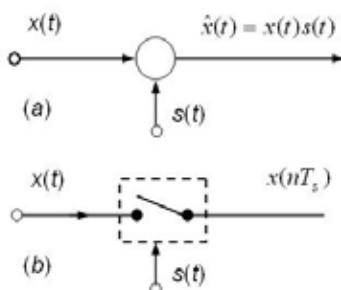


Figure 5: The principle of sampling (a)Multiplying; (b)Switching

The sampling signal  $s(t)$  is a regular sequence of narrow pulses  $\delta(t)$  of amplitude 1 (Figure 3) when multiplying  $s(t)$  with the signal  $x(t)$  we obtain the instantaneous values of  $x(t)$  which are the samples. An electric switch (Figure 2b) is a way to implement the sampling: When the contact closes in a short time, the signal passes; and when the contact opens, no output signal appears.

## V DIGITAL IMAGE SAMPLING

[7] Images can be considered as signals, in one or two dimensions. Images are spatial distributions of values of luminance or color, the latter being described in its RGB or HSB components. In order to be processed by numerical computing devices, have to be reduced to a sequence of discrete samples, and each sample must be represented using a finite number of bits. The first operation is called sampling, and the second operation is called quantization of the domain of real numbers.

### Sampling of 2-D images:

Let us assume we have a continuous distribution, on a plane, of values of luminance or, more simply stated, an image. In order to process it using a computer we have to reduce it to a sequence of numbers by means of sampling. There are several ways to sample an image, or read its values of luminance at discrete points. The simplest way is to use a regular grid, with spatial steps  $X$  e  $Y$ . Similarly to what we did for sounds, we define the spatial sampling rates  $F_X = X$  and  $F_Y = Y$ . For two-dimensional signals, or images, sampling can be described by three facts and a theorem.

- Fact 1: The Fourier Transform of a discrete-space signal is a function (called spectrum) of two continuous variables  $\omega_X$  and  $\omega_Y$ , and it is periodic in two dimensions with periods  $2\pi$ . Given a couple of values  $\omega_X$  and  $\omega_Y$ , the

Fourier transform gives back a complex number that can be interpreted as magnitude and phase (translation in space) of the sinusoidal component at such spatial frequencies.

- Fact 2: Sampling the continuous-space signal  $s(x,y)$  with the regular grid of steps  $X, Y$ , gives a discrete-space signal  $s(m,n) = s(mX,nY)$ , which is a function of the discrete variables  $m$  and  $n$ .
- Fact 3: Sampling a continuous-space signal with spatial frequencies  $F_X$  and  $F_Y$  gives a discrete-space signal whose spectrum is the periodic replication along the grid of steps  $F_X$  and  $F_Y$  of the original signal spectrum. The Fourier variables  $\omega_X$  and  $\omega_Y$  correspond to the frequencies (in cycles per meter) represented by the variables  $f_X = \omega_X / 2\pi Y$  and  $f_Y = \omega_Y / 2\pi X$

The Figure 6 shows an example of spectrum of a two-dimensional sampled signal. There, the continuous-space signal had all and only the frequency components included in the central hexagon. The hexagonal shape of the spectral support (region of non-null spectral energy) is merely illustrative. The replicas of the original spectrum are often called spectral images.

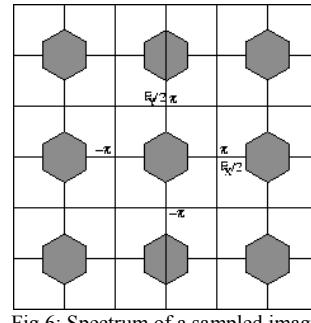


Fig 6: Spectrum of a sampled image

## VI DIGITAL SIGNAL QUANTIZATION

[8] In digital signal processing, quantization is the process of approximating ("mapping") a continuous range of values (or a very large set of possible discrete values) by a relatively small ("finite") set of ("values which can still take on continuous range") discrete symbols or integer values. For example, rounding a real number in the interval [0,100] to an integer

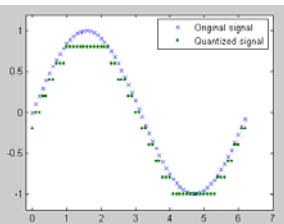


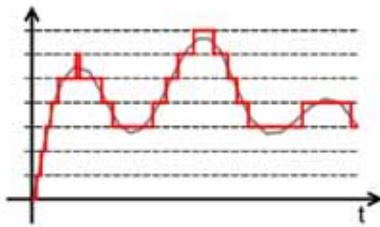
Fig 7: Quantization of Images

In other words, quantization can be described as a mapping that represents a finite continuous interval  $I = [a,b]$  of the range of a continuous valued signal, with a single number  $c$ , which is also on that interval. For example, rounding to the nearest integer (rounding  $\frac{1}{2}$  up) replaces the interval  $[c - .5, c + .5)$  with the number  $c$ , for integer  $c$ . After that quantization we produce a finite set of values which can be encoded by say binary techniques.

In signal processing, quantization refers to approximating the output by one of a discrete and finite set of values, while replacing the input by a discrete set is discretization, and is done by sampling: the resulting sampled signal is called a discrete signal (discrete time), and need not be quantized (it can have continuous values). To produce a digital signal (discrete time and discrete values), one both samples (discrete time) and quantizes the resulting sample values (discrete values).

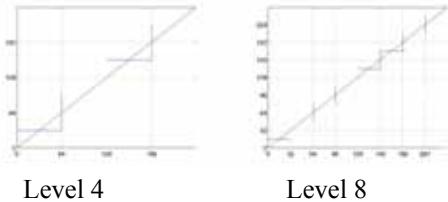
## VII DIGITAL IMAGE QUANTIZATION

[9] Quantization corresponds to a discretization of the intensity values. That is, of the co-domain of the function.



After sampling and quantization, we get  $f : [1; \dots; N]$   
 $[1; \dots; M]$  !

$$[0; \dots; L].$$



Quantization corresponds to a transformation  $Q(f)$ . Typically, 256 levels (8 bits/pixel) suffices to represent the intensity.

For color images, 256 levels are usually used for each color intensity. Quantization, involved in image processing, is a lossy compression technique achieved by compressing a range of values to a single quantum value. When the number of discrete symbols in a given stream is reduced, the stream becomes more compressible. For example, reducing the number of colors required to represent a digital image makes it possible to reduce its file size. Specific applications include DCT data quantization in JPEG and DWT data quantization in JPEG 2000.

## VIII COLOR QUANTIZATION

Color quantization reduces the number of colors used in an image; this is important for displaying images on devices that support a limited number of colors and for efficiently compressing certain kinds of images. Most bitmap editors and many operating systems have built-in support for color

quantization. Popular modern color quantization algorithms include the nearest color algorithm (for fixed palettes), the median cut algorithm, and an algorithm based on octrees. It is common to combine color quantization with dithering to create an impression of a larger number of colors and eliminate banding artifacts.

## IX FREQUENCY QUANTIZATION FOR IMAGE COMPRESSION

The human eye is fairly good at seeing small differences in brightness over a relatively large area, but not so good at distinguishing the exact strength of a high frequency brightness variation. This fact allows one to get away with a greatly reduced amount of information in the high frequency components. This is done by simply dividing each component in the frequency domain by a constant for that component, and then rounding to the nearest integer. This is the main lossy operation in the whole process. As a result of this, it is typically the case that many of the higher frequency components are rounded to zero, and many of the rest become small positive or negative numbers.

## X CONCLUSION AND FUTURE SCOPE

This paper ends up with the block diagram of the general complete DSP system.

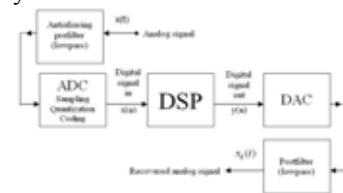


Figure 8: Block diagram of general [10] complete DSP system

The digital signal output  $y(n)$  from the DSP unit is converted by the digital-to-analog converter (DAC or D/A) back to a coarse analog signal which is then lowpass filtered in the postfilter. Similar steps can be followed for processing of images. Also like embedded Digital Signal processors, embedded systems for processing images can be a good area of research work.

## XI REFERENCES:

- [1]. C.Gonzalez Rafel, "Digital Image Processing", Pearson education, 2005, pages 23-25.
- [2]. Jain, Anil.K. , "Fundamentals of digital Image Processing", PHI, 2005, pages 4-6.
- [3]. B.Chanda,D. Dutta Majumder, "Digital Image Processing and Analysis", PHI publications, 2005, pages 80-115 .
- [4]. <http://www.adinstruments.com/solutions/attachments/pr-signalfilters-05a.pdf> Date accessed 12<sup>th</sup> April,2009.
- [5]. JC.Gonzalez Rafel,"Digital Image Processing", Pearson education, 2005, pages 138-149.
- [6]. <http://www.jhu.edu/signals/sampling/> date accessed 8<sup>th</sup> June, 2009.
- [7]. [http://www.ph.tn.tudelft.nl/Courses/FIP/noframes/fip--4.html/](http://www.ph.tn.tudelft.nl/Courses/FIP/noframes/fip--4.html) Date accessed 10<sup>th</sup> April,2009.
- [8]. <http://cnx.org/content/m13045/latest/> Date Accessed 8<sup>th</sup> June, 2009.
- [9]. C.Gonzalez Rafel,"Digital Image Processing", Pearson education, 2005, pages 74-76.
- [10]. <http://www.vocw.edu.vn/content/m10777/latest/> Date accessed 12<sup>th</sup> April, 2009.

# A New Linear CMOS Transconductor

Rumi Rastogi

Inderprastha Engineering College, 63, Flyover Road, Surya Nagar, Sahibabad, Ghaziabad

**Abstract-** The idea behind using transconductor circuits in different analog and mixed signal applications is that these circuits are very simple and consume less area as compared to operational amplifier based active circuits and are therefore are the most popular choice for integrated circuit design. They also have a good frequency response and much wider bandwidth as the basic stage of a transconductance amplifier does not contain any internal nodes and high impedance nodes, where parasitic capacitances can result in long time constants that limit the bandwidth. This paper proposes a linear transconductor based on MRC. An alternative form of the transconductor has also been proposed by changing the inputs and control voltages. Different applications of the proposed transconductor have also been tested and verified.

## I INTRODUCTION

A new linear CMOS transconductor employing a second generation current conveyer (CCII) and a MOS resistive circuit (MRC) has been proposed. The proposed transconductor is supposed to offer a wide linearity range as the MRC cancels all even and odd nonlinearities. An alternative form of the transconductor has also been proposed by changing the inputs and control voltages. Various applications of the proposed transconductor for example a four quadrant analog multiplier, a differential integrator and a second order band-pass have also been tested which verify the workability of the structure.

## II CIRCUIT DESCRIPTION

The proposed transconductor circuit is shown in Fig. 1 which employs a MOS resistive circuit (MRC) [1], and a negative impedance converter (NIC) [2] realized with a second generation current conveyor (CCII). It may be pointed out that although the MRC circuit has been extensively used earlier in realizing fully differential MOSFET-C integrators and filters in conjunction with differential inputs complimentary output type op-amps and current conveyors, along with two matched capacitors ([3], [4]), its use in realizing a transconductor /OTA has never been recognized in literature earlier. Here we show how the MRC in conjunction with a CCII can be used to implement a single output linear transconductor with wide input voltage range which can be easily extended to multiple current-mode outputs.

The second generation current conveyor (CCII) is defined as a three port active building block characterized by the terminal equations  $i_y = 0$ ,  $v_x = v_y$ , and  $i_z = i_x$ . With terminals y and z shorted, the resulting z-port is characterized by the transmission matrix  $[T] = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$  and thus, constitutes an NIC

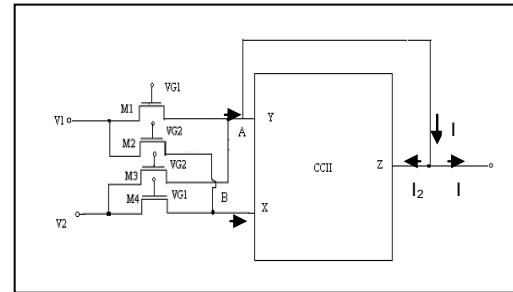


Fig. 1 The proposed Transconductor Circuit

Assuming the MOS transistors of the MRC to have same W/L ratios and operating in triode region, the output current  $I_0$  of the transconductor is, therefore, given by

$$I_0 = I_1 - I_2 = (I_{DM1} + I_{DM3}) - (I_{DM2} + I_{DM4}) \quad (1)$$

$$I_0 = 2K[(V_{G1} - V_{G2})(V_1 - V_2)] \quad (2)$$

Where  $V_{G1}$  and  $V_{G2}$  are the control voltages applied at the gates of the MRC,  $V_1$  and  $V_2$  are the inputs,  $V_T$  is the threshold voltage of the MOS transistors and  $K = \mu_n C_{ox} W/2L$ . The value of transconductance  $G_m$  realized by the circuit is therefore, given by

$$G_m = 2K(V_{G1} - V_{G2}) \quad (3)$$

## III SIMULATION RESULTS

The proposed transconductor circuit has been simulated in SPICE using 1.2 micron CMOS process parameters. The aspect ratios of the transistors of MRC are all equal with  $W = 4\mu m$  and  $L = 20\mu m$ .

The DC transfer characteristics of the proposed transconductor is shown in Fig. 2 which has been obtained with  $V_{G1} = 5V$  and  $V_{G2} = 3.5V$ . It is seen that, as expected, the circuit offers a linear range of input voltage around  $\pm 4$ Volts for the DC bias supply voltage of the CCII+ taken as  $\pm 5$ Volts.

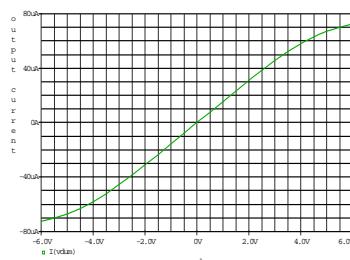


Fig. 2 DC transfer characteristics of the transconductor. (Output current in microamperes versus input voltage in volts)

The frequency response of the transconductor for different values of  $G_m$  obtained from simulation is shown in Fig. 3. From the frequency response, it is seen that in all cases, transconductance gain remains nearly constant till about 100 MHz.

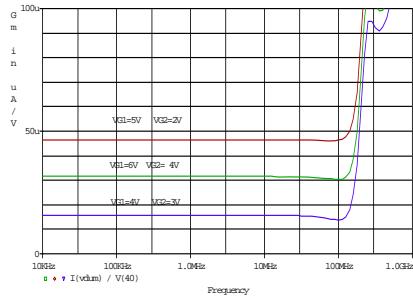


Fig. 3  $G_m$  versus frequency for different values of  $V_{G1}$  and  $V_{G2}$

#### IV ALTERNATIVE FORM OF THE TRANSCONDUCTOR

The transconductor circuit proposed in Fig. 1 can also be used with the inputs applied at the gates and the control voltage ( $V_1$  and  $V_2$ ) at the drain terminals of MRC. In this form, (Fig. 4) the circuit has the attractive feature of offering ideally infinite impedance. It is found that in this mode also the circuit can be used with either differential input or single ended input. However, appropriate DC level shift needs to be applied along with the signal input to ensure MOSFETs operating in triode region.

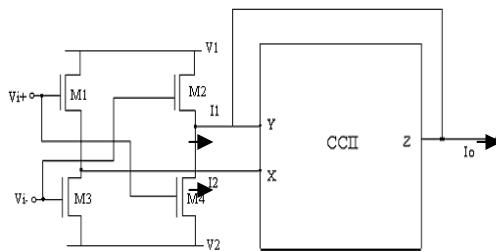


Fig. 4 Transconductor with inputs applied at the gates

In this case, transconductance ( $G_m$ ) of the transconductor is varied by applying different control voltages at the drain terminals of the MOSFETs. The  $G_m$  versus frequency plot for this version of the transconductor is shown in Fig. 5.

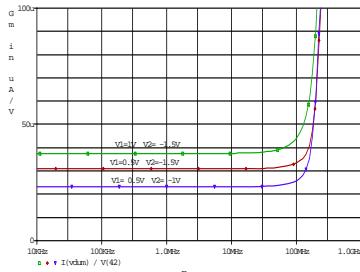


Fig. 5  $G_m$  versus frequency curve for different values of control voltages applied at the drain

#### V THE PROPOSED TRANSCONDUCTOR AS MOTA

The proposed transconductor can be easily extended as a multiple output transconductance amplifier (MOTA) by connecting a multiple output current conveyer (MOCC) at the output stage as shown in Fig. 6. This circuit can have  $m$  identical current outputs each providing  $I_{Zj}^+ = +I_0 = G_m V_{in}$ ;  $j = 1 - m$  and  $n$  identical current outputs each providing  $I_{Zk}^- = -I_0 = G_m V_{in}$ ;  $k = 1 - n$ .

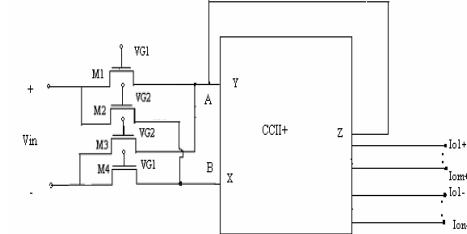


Fig. 6 Conceptual diagram of the MOTA and its implementation using the proposed transconductor

#### VI APPLICATIONS OF THE PROPOSED TRANSCONDUCTOR

A number of applications using the proposed transconductor, both versions (with inputs at the gate as well as with inputs at the drain) have been tested for validating its utility in various fields.

##### A. Four Quadrant-Analog Multiplier

A four quadrant analog multiplier circuit performing real time multiplication of two bipolar signals has been designed using the proposed transconductor. Two signals  $V_x$  and  $V_y$  are being multiplied to produce an output voltage  $V_o = kV_x V_y$ ,  $k$  being a scale factor. In Fig. 7, the voltages  $V_x$  and  $V_y$  are time varying signals. From (3) we can obtain the output current as

$$I_o = I_1 - I_2 = 2KV_x V_y \quad (4)$$

with the voltage  $V_y$  applied at the inputs and  $V_x$  applied to the gates of MRC.

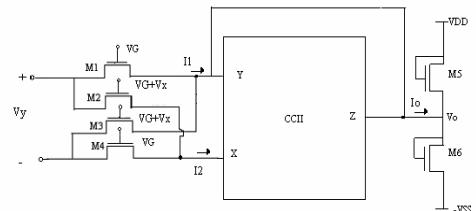


Fig. 7 Fully-integrable CMOS analog multiplier

The output voltage of the multiplier at the output of the trans-resistance (constituted by  $M_5$  and  $M_6$ ) is given by

$$V_o = kV_x V_y \quad (5)$$

where

$$k = \frac{K}{k_5(V_{DD} - V_T)} \quad (6)$$

The circuit, thus, implements a Four-quadrant analog multiplier.

In order to verify the above theoretical analysis, an analog multiplier has been designed using the proposed configuration of Fig. 7 and simulated in PSPICE using 1.2  $\mu\text{m}$  CMOS process parameters. The MOS trans-resistance is designed using  $(W/L)_5 = (W/L)_6 = 5\mu/50\mu$ . The DC power supply is  $V_{DD} = -V_{SS} = 5\text{V}$ . Fig. 8 illustrates the simulation result in time domain. A 10 kHz voltage signal,  $V_y$ , with 4V amplitude is multiplied by 1kHz signal,  $V_x$  of the same amplitude.

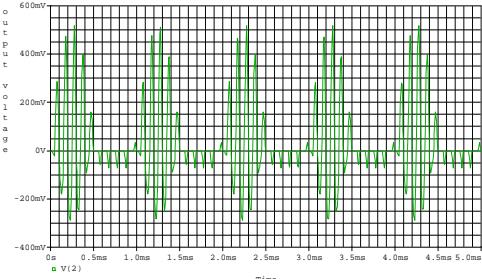


Fig. 8 Trace showing multiplication of input signals  $V_x$  and  $V_y$  at the output of the multiplier in time domain

### B. Differential Integrator

An integrator can be implemented using the proposed transconductor just by terminating the output current of the circuit into a grounded capacitor as shown in Fig. 9. The output voltage of this integrator is obtained in terms of the basic MOSFET parameters as

$$V_{out} = \frac{1}{RC} \int (V_1 - V_2) dt \quad (7)$$

where

$$R = \frac{1}{\mu C_{ox} \frac{W}{L} (V_{G1} - V_{G2})} \quad (8)$$

$V_1 - V_2$  is the differential input voltage and  $V_{G1}, V_{G2}$  are the gate-control voltages. The integrator can be inverting or noninverting depending on  $V_{G1} < V_{G2}$  or  $V_{G1} > V_{G2}$ .

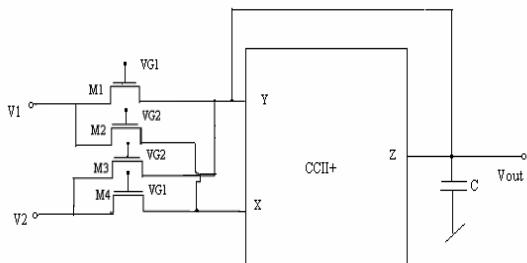


Fig. 9 An integrator using the proposed transconductor

**Simulation results of the Integrator:** The SPICE simulation results for this integrator have been shown in Fig. 10. The time domain analysis has been performed by applying a square wave

input (300mV  $\text{p-p}$ ) with  $C = 0.001\mu\text{F}$ . The simulation results confirm the workability of the circuit as an integrator.

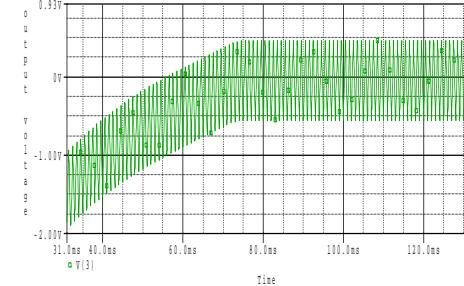


Fig. 10 Transient response of the integrator obtained with a pulse input

### C. An Active Second Order Band-Pass Filter

Using the inverting and noninverting integrators to realize a lossless simulated grounded inductance in a parallel LC tank circuit, in conjunction with a series resistor judiciously simulated by only one transconductor, we have obtained a simple second order bandpass filter shown in Fig. 11 which employs only three transconductors along with both grounded capacitors, as preferred for IC implementation. The transfer function of this bandpass filter is found to be

$$T(s) = \frac{V_2(s)}{V_1(s)} = \frac{s \left( \frac{G}{C} \right)}{s^2 + s \left( \frac{G}{C} \right) + \frac{G_1 G_2}{C C_0}} \quad (9)$$

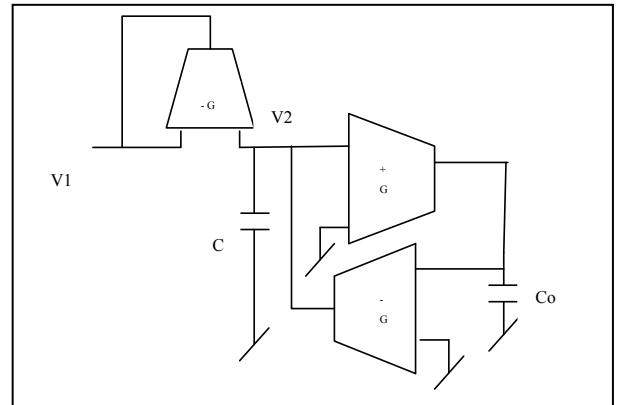


Fig. 11 An active second-order Band-Pass Filter

**Simulation Results of the Band-pass Filter:** The second-order bandpass filter has been simulated to conform its workability. With  $G = 0.14 \mu\text{A/V}$ ,  $G_1 = 30 \mu\text{A/V}$ ,  $G_2 = 27.6 \mu\text{A/V}$ ,  $C_0 = 0.1\text{nF}$  and  $C = 0.21\text{pF}$ , the circuit realized a BP filter with center frequency,  $f_0 = 1\text{MHz}$ , bandwidth = 0.65MHz and Gain = 1. The tunability property of the filter has been checked by varying the bias control voltage of the OTA configured as resistor ( $G$ ). The resulting variation in the BW with respect to the control voltage is shown in Fig. 12. By varying  $G$  for 0.14 $\mu\text{A/V}$ , 0.07 $\mu\text{A/V}$  and 0.02 $\mu\text{A/V}$  the bandwidths obtained were 0.66 MHz, 0.34 MHz and 0.1 MHz respectively for the same center frequency of 1MHz.

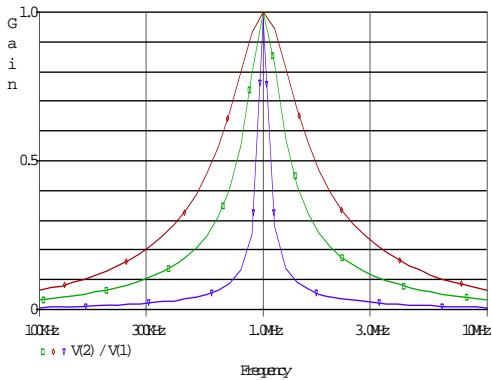


Fig. 12 Frequency response of the band-pass filter showing tunable bandwidth

## VII ACKNOWLEDGEMENT

The author wants to thank Dr. A.K.Singh for his continuous support and comments on the manuscript.

## VIII. REFERENCES

- [1]. Czarnul Z., ‘Novel MOS resistive circuit for synthesis of fully integrated continuous-time filters’, IEEE Transactions on Circuits and Systems, vol. CAS-33, No. 7, pp. 718-720, July 1986
- [2]. Liu S.-I., TSAO, H.-W, Wu, J. and Lin, T. -K., ‘MOSFET capacitor filters using unity gain CMOS current conveyors’, Electronic Letters, vol. 26, no.18, pp.1430-1431, 1990.
- [3]. Elwan, H.O. and Soliman, A.M. ‘A novel CMOS Current Conveyor realization with an electronically tunable current mode Filter suitable for VLSI,’ IEEE Trans on Circuits and Systems –II: Analog and Digital Signal Processing, vol.43, no.9, pp.663-670, 1996
- [4]. Ryan, P.J. and Haigh, D.G. ‘Novel fully differential MOS transconductor for integrated continuous time filter,’ Electronics Letters, vol.23, pp742-743, 1987

# FANSys: A Software Tool to Design Artificial Neural Network System

Bijeta Oberoi\*, Savita Wadhawan\*\*

Email:oberoi.bijeta@gmail.com, \*JMIT, Radaur, Haryana, \*\*Institute of Science & Technology, Klawad, Haryana  
Email:oberoi.bijeta@gmail.com, \*\*savitawadhawan@gmail.com

**Abstract-** This paper includes a short description of the newly developed software tool named FANSys. FANSys is a Feedforward Artificial Neural network backpropagation System design tool which implements the ANNs using Levenberg Marquardt algorithm. Multilayer perception is perhaps the most popular network architecture in use today. This paper aims to provide an intuitive explanation of learning process of artificial multilayer neural networks using Levenberg Marquardt algorithm. LM algorithm is the most widely used optimization algorithm and is considered to be the most efficient second order algorithm for training feedforward neural network. It is a variation of feedforward Backpropagation algorithm. The algorithm is experimented with different types of linear and nonlinear problems and ANN based control system rapid NiCa battery charger.

## I. INTRODUCTION

ANN's, the branch of artificial intelligence, date back the 1940's, when McCulloch and Pitts first developed the first neural model. Since, then the wide interest in ANN's both among researchers and in area of various applications, has resulted in more powerful networks, better training algorithms and improved hardware. Because of their abstraction from the brain, ANN's are good at solving problems that human are good at solving but which computers are not. Pattern recognition and classification are examples of problems that are well suited for ANN application [1].

ANN's are a biologically motivated form of parallel computation where simple but highly interconnected processing elements known as neurons perform weighted aggregation of their inputs followed by a activation function. Figure 1 shows the structure of artificial neuron.

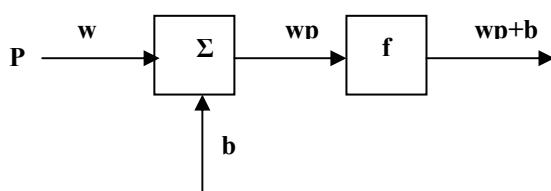


Fig 1: Structure of artificial neuron

Here 'p' is the input to the neuron, w is the weights required, b is the bias value, f is the transfer function which is actually the deciding factor for the output we get.

This paper is organized as follows: In the second section, description of training multilayer network structures is given. The next section focuses on the Levenberg Marquardt algorithm. Section four gives the implementation details of FANSys. Case study of NiCa battery charger is discussed in

fifth section. Finally, conclusion and future scope constitute the last part of the paper.

## II. TRAINING MULTILAYER PERCEPTRON (MLP)

A Feed forward network allows signals to move from input to output nodes only (figure 2). There is no feedback from output to input or hidden nodes or lateral connections among the same layer whereas a feedback network allows signals to travel in both directions by introducing loops in the network.

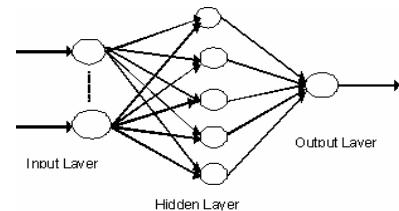


Fig 2: Feedforward network

An MLP distinguishes itself by presence of one or more hidden layers with computation nodes called hidden neurons whose function is to intervene between the external input and the network output in a useful manner [3]. There may be one or more hidden layers.

Neural networks have the capability of transforming inputs into the desired output changes; this is called neural network Training or Learning. These changes are generally produced by the sequentially applying input values to networks while adjusting network weights. This is similar to learning in biological system that involves adjustment to the synaptic connections that exists between neurons. During the learning process the network weights converge to values such that each input vector produces the desired output vector.

## III. THE LEVENBERG MARQUARDT ALGORITHM

The Backpropagation (Werbos,1974) (Rumelhart, McClelland,1986) is one of the best and most used algorithms developed so far to train feed forward multilayer neural network using supervised training. In Backpropagation, the gradient vector of the error surface is calculated. This vector points along the line of steepest descent from the current point so we know that if we move along it a "short" distance we will decrease the error. A sequence of such moves will eventually find a minimum of some sort. Although the best known example of a neural network training algorithm is Back Propagation which has been a significant milestone in neural network research area of interest, it has been known as algorithm with a very poor convergence rate and slow speed.

The slow speed has been attributed to the presence of large flat plateaus on the surface of the error function while the quality of convergence is dependable on the presence of several local minima on this same surface. These problems became more evident as the complexity of the network and the size of the application increases [4].

Modern second order algorithms such as Newton's method, Conjugate gradients and the Levenberg Marquardt (LM) are substantially faster as compared to Error Backpropagation Algorithm (EBP). Among the mentioned methods the LM algorithm is widely accepted as the most efficient method as it gives a good compromise between the speed of Newton algorithm and the stability of the steepest descent method, and consequently it constitutes a good transition of these methods [2].

While Backpropagation with gradient descent technique is a steepest descent algorithm, the Levenberg Marquardt (LM) algorithm is an approximation to Newton's method. Gradient based training algorithms are not efficient due to the fact that the gradient vanishes at the solution, Hessian based algorithms used as reported in [5], allow the network to learn more subtle features of a complicated mapping. The training process converges quickly as the solution is approached, because the Hessian does not vanish at solution. The LM algorithm is basically a Hessian based for nonlinear least squares optimization. For neural network training, the objective function is the error function of the type of "sum squared error" where the individual errors of output units on each case are squared and summed together and is given as

$$e(x) = \frac{1}{2} \sum_{k=0}^{p-1} \sum_{l=0}^{n_0-1} (d_{kl} - a_{kl})^2 \quad (1)$$

where  $a_{kl}$  is the actual output at the output neuron 1 for the input  $k$  and  $d_{kl}$  is the desired output at the output neuron 1 for the input  $k$ .  $p$  is the total number of training patterns and  $n_0$  represents the total number of neurons in the output layer of the network.  $x$  represents the weights and biases of the network [6].

If a function unction  $V(x)$  is to be minimized with respect to the parameter vector  $x$ , then Newton's method would be:

$$\Delta x = -[\nabla^2 V(x)]^{-1} \nabla V(x) \quad (2)$$

where  $\nabla^2 V(x)$  is the Hessian matrix and  $\nabla V(x)$  is the gradient. If  $V(x)$  reads:

$$V(x) = \sum_{i=1}^N e_i^2(x) \quad (3)$$

then it can be shown that:  $\nabla V(x) = J^T(x)e(x)$

(4)

$$\nabla^2 V(x) = J^T(x)J(x) + S(x) \quad (5)$$

where  $J(x)$  is the Jacobian matrix and

$$S(x) = \sum_{i=1}^N e_i \nabla^2 e_i(x) \quad (6)$$

For the Gauss-Newton method it is assumed that  $S(x) \approx 0$ , and equation (2) becomes:

$$\Delta x = [J^T(x)J(x)]^{-1} J^T(x)e(x) \quad (7)$$

The Levenberg-Marquardt modification to the Gauss-Newton method is

$$\Delta x = [J^T(x)J(x) + \mu I]^{-1} J^T(x)e(x) \quad (8)$$

The parameter  $\mu$  is multiplied by some factor ( $\beta$ ) whenever a step would result in an increased  $V(x)$ . When a step reduces  $V(x)$ ,  $\mu$  is divided by  $\beta$ . When the scalar  $\mu$  is very large the Levenberg-Marquardt algorithm approximates the steepest descent method. However, when  $\mu$  is small, it is the same as the Gauss-Newton method. Since the Gauss-Newton method converges faster and more accurately towards an error minimum, the goal is to shift towards the Gauss-Newton method as quickly as possible. The value of  $\mu$  is decreased after each step unless the change in error is positive; i.e. the error increases [3].

The steps involved in training a neural network using the Levenberg-Marquardt algorithm are as follows:

1. Present all inputs to the network and compute the corresponding network outputs and errors. Compute the mean square error over all inputs as in equation 1.
2. Compute the Jacobian matrix,  $J(x)$  where  $x$  represents the weights and biases of the network. Order of this matrix is  $p \times n$  where  $p$  is the number of training set,  $n$  is the number of weight and bias in network.  $J(x)$

$$J(x) = \begin{bmatrix} \frac{\partial v_1(x)}{\partial x_1} & \frac{\partial v_1(x)}{\partial x_2} & \dots & \frac{\partial v_1(x)}{\partial x_n} \\ \frac{\partial v_2(x)}{\partial x_1} & \frac{\partial v_2(x)}{\partial x_2} & \dots & \frac{\partial v_2(x)}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial v_N(x)}{\partial x_1} & \frac{\partial v_N(x)}{\partial x_2} & \dots & \frac{\partial v_N(x)}{\partial x_n} \end{bmatrix}$$

is given as

3. Solve the Levenberg-Marquardt weight update equation to obtain  $\Delta x$ . (Refer to equation 8)
4. Recompute the error using  $x + \Delta x$ . If this new error is smaller than that computed in step 1, then reduce the training parameter  $\mu$  by  $\mu^-$ , let  $x = x + \Delta x$ , and go back to step 1. If the error is not reduced, then increase  $\mu$  by  $\mu^+$  and go back to step 3.  $\mu^+$  and  $\mu^-$  are predefined values set by the user. Typically  $\mu^+$  is set to 10 and  $\mu^-$  is set to 0.1.
5. The algorithm is assumed to have converged when the norm of the gradient is less than some predetermined value, or when the error has been reduced to some error goal.

The key drawback of the LBMP algorithm is the storage requirement. The algorithm must store the approximate Hessian matrix,  $J^T J$ . This is an  $n \times n$  matrix, where  $n$  is the number of parameters (weights and biases) in the network. When the number of parameters is very large, it may be impractical to use the Levenberg Marquardt algorithm.

#### IV. IMPLEMENTATION

A Feedforward backpropagation Artificial Neural network System design (FANSys) is a design automation tool for the artificial neural network based systems. It is a GUI based tool which interacts with the user to obtain user specifications and the observed input output data. Figure 3 shows the welcome screen of FANSys. In order to train a neural network, two data files have been prepared; the input values are listed followed by the desired output values in two separate files. The training parameters can be selected from the user interface. The user enters the number of inputs and outputs, the maximum acceptable error and number of iterations as shown in figure 4. Next as shown in figure 5 and figure 6 respectively, the type of activation functions for the hidden and output layers are chosen by the user himself depending upon the specifications of the problem. After the user specifications has been provided the tool explores various network structures to train and obtain an appropriate network structure for a given application. Feedforward Levenberg Marquardt algorithm has been used for training the network as found in figure 7. Random initialization of weights between the layers is done. Computation of weight update is done using Jacobian matrix and Hessian matrix w.r.t the error being calculated. Once the performance criterion is met, the FANSys propose a structure in graphical form; it then also implements the ANN in “C” language giving the error, number of iterations and appropriate weights.



Fig 3: FANSys Screen Snapshot\_1



Fig 4: FANSys Screen Snapshot\_2



Fig 5: FANSys Screen Snapshot\_3

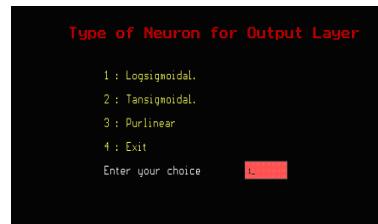


Fig 6: FANSys Screen Snapshot\_4

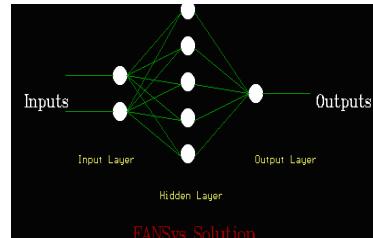


Fig 7: FANSys Screen Snapshot\_5

FANSys selects single or two hidden layers network structure depending upon the value of flag, then trains the network and if the required performance is not achieved, the network structure is changed by adding neurons in the hidden layer. The software optimizes the number of hidden layer neurons depending upon the number of iterations and mean square error.

Several simulations were run to test the architecture and algorithm. The performance of the system is validated using a validating data set and also with the help of MATLAB NNTOOL software. FANSys has been applied to different types of linear and non linear problems.

#### V. CASE STUDY

Nickel Cadmium battery charger, which has been rendered intelligent by the application of Artificial Neural networks [7] continuously monitors the battery and as the input varies, pumps varying current into the battery accordingly. Since, any battery is most affected by the increase in temperature as the charging continues; we have taken temperature and temperature gradient as the two input factors. As soon as any change occurs, it manipulates the output current being pumped accordingly. The data for the batteries charger has been obtained with an objective to charge the NiCa batteries as fast as possible. From the data we select some data values and divide them into a training data set and a validating data set (Table 4.1). The training data set is used to train the neural network. Figure 8 and figure 9 shows the graphs between the desired output and obtained output for training data set and the validating data set.

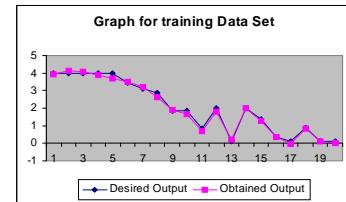


Fig 8: Graph for training data set

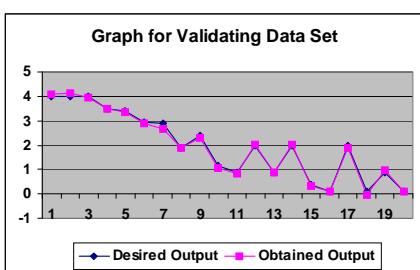


Fig 9: Graph for validating data set

## VI. CONCLUSION & FUTURE SCOPE

In this paper a software design automation tool; the FANSys has been presented that can be very useful in design and implementation of an ANN based system. The FANSys is capable of automatically implementing an ANN in ‘C’. The design tool slashes the design and implementation time considerably. The tool is extremely useful to the designers who are not very comfortable with ANN design techniques but like to test their concepts as an ANN. Further, design of ANN based Nickel Cadmium based battery charger was simulated and found to be very satisfactory. In future, modifications to LM algorithm can be made by changing the performance index and calculating gradient information.

## VII. REFERENCES

- [1]. Laurene Fausett," Fundamentals of Neural Network, Architectures, Algorithm and Applications", Prentice Hall International 1994.
- [2]. Bogdan M. Wilamowski, Serdar Iplikci , Okyay Kaynak and M. Önder Efe "An Algorithm for Fast Convergence in Training Neural Networks".
- [3]. Ozgur Kisi, "Multi-layer perceptrons with Levenberg- Marquardt training algorithm for suspended sediment concentration prediction and estimation"
- [4]. Arthur Salvetti and Bogdan M. Wilamowski," Introducing Stochastic processes within the Backpropagation Algorithm for improved convergence"
- [5]. E.P.A. Jr and T.J.Bartolac," Parallel Neural Network Training" in Proc AAAI Spring Symposium on Innovative Applications of Massive Parallelism, Stanford University, March 1993.
- [6]. N. N. R. Ranga Suri, Dipti Deodhare .. Nagabhushan, "Parallel Levenberg-Marquardt-based Neural Network Training on Linux Clusters - A Case Study".
- [7]. Shakti Kumar,Neetika Ohri and Savita Wadhawan, "Ann Based Design of A Rapid Battery Charger".

Training data set				
Data point	Input1(Temp)	Input2(Tem p_Gradient)	Desired output	Obtained output
1	0	0.1	4	4.01
2	0	1	4	4.02
3	10	0.2	4	4.00
4	20	0	4	3.85
5	20	1	4	3.25
6	24	0.4	3.47	3.20
7	24	0.6	3.09	3.10
8	28	0.4	2.89	2.50
9	28	0.7	1.87	1.70
10	28	1	1.87	1.60
11	32	1	0.86	0.71
12	35	0	2	1.71
13	35	1	0.1	0.2
14	40	0	2	2.00
15	40	0.4	1.37	1.20
16	40	0.6	0.357	0.30
17	40	1	0.1	-0.02
18	45	0.5	0.871	0.81
19	45	0.7	0.1	0.10
20	50	1	0.1	0.04

Validating data set				
Data point	Input1(Temp)	Input2(Tem p_Gradient)	Desired output	Obtained output
1	0	0.4	4	4.01
2	10	0.5	4	4.05
3	20	0.3	4	3.7
4	24	0.4	3.47	3.40
5	24	0.5	3.39	3.20
6	24	0.9	2.93	2.80
7	28	0.4	2.89	2.511
8	28	0.7	1.87	1.801
9	32	0.2	2.4	2.11
10	32	0.6	1.13	1.00
11	32	0.7	0.86	0.812
12	35	0.2	2	2.00
13	35	0.5	0.871	0.91
14	40	0.2	2	2.00
15	40	0.6	0.357	0.311
16	40	0.7	0.1	0.05
17	45	0.3	2	1.810
18	45	0.9	0.1	-0.027
19	50	0.5	0.871	0.912
20	50	0.8	0.1	0.100

Table 4.1 Data points used as training and validating data set

# Embedded Systems Security Issues and Implementation by using Cryptographic Algorithms.

Manu Dev- Lecturer, Sarita Soi- Lecturer

Department of Computer Science and Applications, Swami Devi Dal institute of Engg. & Tech., Barwala (Panchkula), Haryana  
email: [manu5\\_dev@yahoo.co.in](mailto:manu5_dev@yahoo.co.in)

Chander Kant- Lecturer

Department of Computer Science and Applications, Kurukshetra University, Kurukshetra, Haryana.

**Abstract-** from cars to cell phones, video equipment to MP3 players, and dishwashers to home thermostats— embedded computers increasingly permeate our lives. But security for these systems is an open question and could prove a more difficult long-term problem than security does today for desktop and enterprise computing. It is widely recognized that data security will play a central role in the design of future IT systems. Many of those IT applications will be realized as embedded systems which rely heavily on security mechanisms. Examples include security for wireless phones, wireless computing, pay-TV, and copy protection schemes for audio/video consumer products and digital cinemas. Note that a large share of those embedded applications will be wireless, which makes the communication channel especially vulnerable. Today you can buy Internet-enabled home appliances and security systems, and some hospitals use wireless IP networks for patient care equipment. Cars will inevitably have indirect Internet connections—via a firewall or two—to safety-critical control systems. There have already been proposals for using wireless roadside transmitters to send real-time speed limit changes to engine control computers. There is even a proposal for passenger jets to use IP for their primary flight controls, just a few firewalls away from passengers cruising the Web. Internet connections expose applications to intrusions and malicious attacks. Unfortunately, security techniques developed for enterprise and desktop computing might not satisfy embedded application requirements. In this paper we highlight the security issues, challenges and implementation of security methods in embedded systems in terms of cryptography techniques. All modern security protocols use symmetric-key and public-key algorithms. In this paper we highlights the contribution surveys several important cryptographic concepts and their relevance to embedded system applications. We give an overview of the previous work in the area of embedded systems and cryptography.

## I. INTRODUCTION

It is widely recognized that data security will play a central role in the design of future IT systems. Until a few years ago, the PC had been the major driver of the digital economy. Recently, however, there has been a shift towards IT applications realized as embedded systems. Many of those applications rely heavily on security mechanisms, including security for wireless phones, faxes, wireless computing, pay-TV, and copy protection schemes for audio/video consumer products and digital cinemas. Note that a large share of those embedded applications will be wireless, which makes the communication channel especially vulnerable and the need for security even more obvious. This merging of communications and computation functionality requires data processing in real time, and embedded systems have shown to be good solutions

for many applications. Examples of such applications are cellular phones, faxes, pagers, and Internet solutions such as modems, multi-service network solutions that allow the implementation of IP telephony, Digital Subscriber Line (DSL) technologies, and some electronic commerce devices, to name just a few. Since many of these applications will need security functionality in the future, this paper discusses cryptographic algorithms and their implementation on embedded systems. In the next section we are going to discuss the challenges in context of security in embedded systems.

## II. SECURITY ISSUES IN EMBEDDING SYSTEMS

### II. A. COST SENSITIVITY

Embedded systems are often highly cost sensitive—even five cents can make a big difference when building millions of units per year. For this reason, most CPUs manufactured worldwide use 4- and 8-bit processors, which have limited room for security overhead. Many 8-bit microcontrollers, for example, can't store a big cryptographic key. This can make best practices from the enterprise world too expensive to be practical in embedded applications. Cutting corners on security to reduce hardware costs can give a competitor a market advantage for price-sensitive products. And if there is no quantitative measure of security based before a product is deployed who is to say how much to spend on it.

### B. Interactive matters

Many embedded systems interact with the real world. A security breach thus can result in physical side effects, including property damage, personal injury, and even death. Backing out financial transactions can repair some enterprise security breaches, but reversing a car crash isn't possible. Unlike transaction-oriented enterprise computing, embedded systems often perform periodic computations to run control loops with real-time deadlines. Speeds can easily reach 20 loops per second even for mundane tasks. When a delay of only a fraction of a second can cause a loss of controlloop stability, systems become vulnerable to attacks designed to disrupt system timing. Embedded systems often have no real system administrator. Who's the sysadmin for an Internet-connected washing machine? Who will ensure that only strong passwords are used? How is a security update handled? What if an attacker takes over the washing machine and uses it as a platform to launch distributed denial-of service (DoS) attacks against a government agency?

### *1) C. Energy constraints*

Embedded systems often have significant energy constraints, and many are battery powered. Some embedded systems can get a fresh battery charge daily, but others must last months or years on a single battery. By seeking to drain the battery, an attacker can cause system failure even when breaking into the system is impossible. This vulnerability is critical, for example, in battery-powered devices that use power-hungry wireless communication.

## III. CRYPTOGRAPHY AND ITS IMPLEMENTATION ISSUES

The explosive growth of digital communications also brings additional security challenges. Millions of electronic transactions are completed each day, and the rapid growth of eCommerce has made security a vital issue for many consumers. In the future, valuable business opportunities will be realized over the Internet and megabytes of sensitive data will be transferred and moved over insecure communication channels around the world. Thus, it is imperative for the success of modern businesses that all these transactions be realized in a secure manner. Specially, unauthorized access to information must be prevented, privacy must be protected, and the authenticity of electronic documents must be established. Cryptography, or the art and science of keeping messages secure [26], allows us to solve these problems. We believe that cryptographic engines realized on embedded systems are a promising option for protecting eCommerce systems.

The implementation of cryptographic systems presents several requirements and challenges. First, the performance of the algorithms is often crucial. One needs encryption algorithms to run at the transmission rates of the communication links. Slow running cryptographic algorithms translate into consumer dissatisfaction and inconvenience. On the other hand, fast running encryption might mean high product costs since traditionally, higher speeds were achieved through custom hardware devices. In addition to performance requirements, guaranteeing security is a formidable challenge. An encryption algorithm running on a general-purpose computer has only limited physical security, as the secure storage of keys in memory is difficult on most operating systems. On the other hand, hardware encryption devices can be securely encapsulated to prevent attackers from tampering with the system. Thus, custom hardware is the platform of choice for security protocol designers. Hardware solutions, however, come with the well-known drawback of reduced flexibility and potentially high costs. These drawbacks are especially prominent in security applications which are designed using new security protocol paradigms. Many of the new security protocols decouple the choice of cryptographic algorithm from the design of the protocol. Users of the protocol negotiate on the choice of algorithm to use for a particular secure session. The new devices to support these applications, then, must not only support a single cryptographic algorithm and protocol, but also must be "algorithm agile," that is, able to select from a variety of algorithms. For example, IPSec (the security standard for the Internet) allows to choose out of a list of different symmetric as well asymmetric ciphers. Some of the

symmetric-key algorithms are: DES, 3DES, Blowfish, CAST, IDEA, RC4, RC6, and so on. Thus, software-based systems would seem to be a better fit because of their flexibility. However, the security engineer is faced with a difficult choice. Should he/she choose in favor of performance and security, and pay the price of inflexibility and higher costs? Or should he/she favor flexibility instead? Fortunately, many embedded processors combine the flexibility of software on general-purpose computers with the near-hardware speed and better physical security than general-purpose computers. Embedded processors are already an integral part of many communications devices and their importance will continue to increase. If we combine this with their flexibility to be programmed and their ability to perform arithmetic operations at moderate speeds, it is easy to see that they are a very promising platform to implement cryptographic algorithms. This paper focuses on the basics of cryptography and the implementation of cryptographic applications on embedded systems. In Section 2, we introduce the general theory and concepts of symmetric-key and public-key cryptography as well as the operations which are most commonly performed. We will show that public-key operations are very computationally intensive and therefore require platforms which have strong arithmetic. In Section 3, a survey of previous cryptographic implementations on embedded systems is presented, as well as some of the characteristics of the proposed algorithms. We give an overview of implementations of symmetric-key and public-key algorithms. Finally, we end this contribution with some conclusions.

## IV. CRYPTOGRAPHY: PUBLIC-KEY AND SYMMETRIC-KEY ALGORITHMS

### WHAT WE CAN DO WITH CRYPTOGRAPHY

Cryptography involves the study of mathematical techniques that allow the practitioner to achieve or provide the following objectives or services [22, 27]:

Confidentiality is a service used to keep the content of information accessible to only those authorized to have it. This service includes both protection of all user data transmitted between two points over a period of time as well as protection of traffic flow from analysis. Integrity is a service that requires that computer system assets and transmitted information be capable of modification only by authorized users. Modification includes writing, changing, changing the status, deleting, creating, and the delaying or replaying of transmitted messages. It is important to point out that integrity relates to active attacks and therefore, it is concerned with detection rather than prevention. Moreover, integrity can be provided with or without recovery, the first option being the more attractive alternative. Authentication is a service that is concerned with assuring that the origin of a message is correctly identified. That is, information delivered over a channel should be authenticated as to the origin, date of origin, data content, time sent, etc. For these reasons this service is subdivided into two major classes: entity authentication and data origin authentication. Notice that the second class of authentication implicitly provides data integrity.

*Non-repudiation* is a service which prevents both the sender and the receiver of a transmission from denying previous commitments or actions. These security services are provided by using cryptographic algorithms.

There are two major classes of algorithms in cryptography: Private-key or Symmetric-key algorithms and Public-key algorithms. The next two sections will describe them in detail.

#### A. Symmetric-Key Algorithms

Private-key or Symmetric-key algorithms are algorithms where the encryption and decryption key is the same, or where the decryption key can easily be calculated from the encryption key and vice versa. The main function of these algorithms, which are also called secret-key algorithms, is encryption of data, often at high speeds. Private-key algorithms require the sender and the receiver to agree on the key prior to the communication taking place. The security of private-key algorithms rests in the key; divulging the key means that anyone can encrypt and decrypt messages. Therefore, as long as the communication needs to remain secret, the key must



remain secret. There are two types of symmetric-key algorithms which are commonly distinguished: block ciphers and stream ciphers [26]. Block ciphers are encryption schemes in which the message is broken into strings (called blocks) of fixed length and encrypted one block at a time. Examples include the Data Encryption Standard (DES) [11], the International Encryption Standard (IDEA) [19, 21], and the Advanced Encryption Standard (AES) [29]. Note that, due to its short block size and key length, DES expired as a US standard in 1998, and that the National Institute of Standards (NIST) selected Rijndael algorithm as the AES in October 2000. AES has a minimum block size of 128 bits and the ability to support keys of 128, 192 and 256 bits in length. Stream ciphers operate on a single bit of plaintext at a time. In some sense, they are block ciphers having block length equal to one. They are useful because the encryption transformation can change for each symbol of the message being encrypted. In particular, they are useful in situations where transmission errors are highly probable because they do not have error propagation. In addition, they can be used when the data must be processed one symbol at a time because of lack of equipment memory or limited buffering. It is important to point out that the trend in modern symmetric-key cipher design has been to optimize the algorithms for efficient software implementation in modern processors. This is evident if one looks at the performance of the AES on different platforms. The internal AES operations can be broken down into 8-bit operations, which is important because many cryptographic applications run on smart cards. Furthermore, one can combine certain steps to get a suitable performance in the case of 32-bit platforms. As a final remark, notice that one of the major issues with symmetric-key systems is the need to find an efficient method to agree on and exchange the secret keys securely [22].

This is known as the key distribution problem. In 1977, Diffie and Hellman [9] proposed a new concept that would revolutionize cryptography as it was known at the time. This new concept was called public-key cryptography.

#### B. Public-Key Algorithms

Public-key (PK) cryptography is based on the idea of separating the key used to encrypt a message from the one used to decrypt it. Anyone that wants to send a message to party A can encrypt that message using A's public key but only A can decrypt the message using her private key. In implementing a public-key cryptosystem, it is understood that A's private key should be kept secret at all times. Furthermore, even though A's public key is publicly available to everyone, including A's adversaries, it is impossible for anyone, except A, to derive the private key (or at least to do so in any reasonable amount of time). In general, one can divide practical public-key algorithms into three families:

Algorithms based on the integer factorization problem: given a positive integer n, find its prime factorization. RSA [25], the most widely used public-key encryption algorithm, is based on the difficulty of solving this problem.

-Algorithms based on the discrete logarithm problem: given  $\alpha$  and  $\beta$  and x such that  $\beta = \alpha^x \pmod p$ . The Diffie-Hellman key exchange protocol is based on this problem as well as many other protocols, including the Digital Signature Algorithm (DSA).

Algorithms based on Elliptic Curves. Elliptic curve cryptosystems are the most recent family of practical public-key algorithms, but are rapidly gaining acceptance. Due to their reduced processing needs, elliptic curves are especially attractive for embedded applications.

Despite the differences between these mathematical problems, all three algorithm families have something in common: they all perform complex operations on very large numbers, typically 1024-2048 bits in length for the RSA and discrete logarithm systems or 160-256 bits in length for the elliptic curve systems. Since elliptic curves are somewhat less computationally intensive than the other two algorithm families, they seem especially attractive for embedded applications. The most common operation performed in public-key schemes is modular exponentiation, i.e., the operation  $x^e \pmod n$ . Performing such an exponentiation with, e.g., 1024-bit long operands is extremely computationally intensive. Interestingly enough, modular exponentiation with long numbers requires arithmetic which is very similar to that performed in signal processing applications [18], namely integer multiplication. Public-key cryptosystems solve in a very elegant way the key distribution problem of symmetric-key schemes. However, PK systems have a major disadvantage when compared to private-key schemes. As stated above, public-key algorithms are very arithmetic intensive and if not properly implemented or if the underlying processor has a poor integer arithmetic performance this can lead to a poor system performance. Even when properly implemented, all PK schemes proposed to date are several orders of magnitude slower than the best known private-key schemes. Hence, in practice, cryptographic systems are a mixture of symmetric-key

and public-key cryptosystems. Usually, a public-key algorithm is chosen for key establishment and authentication through digital signatures, and then a symmetric-key algorithm is chosen to encrypt the communications and the data transfer, achieving in this way high throughput rates.

## V. EMBEDDED SYSTEMS AND CRYPTOGRAPHY

The field of efficient algorithms for the implementation of cryptographic schemes is a very active one (for an overview on current techniques see [22, Chapter 14]). However, essentially all cryptographic research is being conducted independent of hardware platforms, and little research focuses on algorithm optimization for specific processors. In the following, we will review previous implementations of symmetric-key and public-key algorithms on embedded systems. We will also summarize two of the fastest software implementations of PK schemes on general purpose computers. This will give the reader an idea as to the kind of speeds that are to be expected on general purpose machines, and which speeds can be expected in embedded system applications.

### A. Symmetric-key Algorithms on DSPs

In [31], the authors investigated how well high-end DSPs are suited for the implementation of the final five AES candidate algorithms. In particular, the implementations are on a 200 MHz TMS320C6201 which performs up to 1600 million instructions per second (MIPS) and provides thirty two 32-bit registers and eight independent functional units. In what follows, we briefly describe the way [31] chose to code the Rijndael algorithm because there are several implementation options. In [7], the authors of Rijndael proposed a way of combining the different steps of the round transformation into a single set of table lookups. Thus, the implementation in [31] uses 4 tables with 256 4-byte word entries. In addition to the optimizations described above, a second version of code in which data blocks can be processed in parallel was implemented. With parallel processing, the encryption and the decryption functions can operate on more than one block at a time using the same key. This allows better utilization of the DSP's functional units which leads to better performance. With parallel processing, however, the speedups may only be exploited in modes of operations which do not require feedback of the encrypted data, such as Electronic Code-Book (ECB) or Counter Mode. When operating in feedback modes such as Ciphertext Feedback mode, the ciphertext of one block must be available before the next block can be encrypted. The authors in [31] noticed that the Rijndael code can be optimized by the tools very efficiently. Thus, no performance advantage is obtained by parallel processing, which results in the same speed for single-block and multi-block modes. Table 1 summarizes the performance of Rijndael on the TMS320C6201.

Table 1. Performance results for the Rijndael algorithm on the TMS320C6201 [31]

	DSP multi-block mode @ 200MHz cycles	DSP single-block mode @ 200MHz cycles	Pentium-Pro @ 200MHz Mbit/sec	DSP multi-block mode/Pentium
Rijndael encryption	228	112.3	228	112.3
decryption	269	95.2	269	95.2

All the timings are obtained from a C implementation using the compiler version 4.0 alpha

running at the DSP's maximum speed of 20 MHz. Reference [10] describes the implementation of a cryptographic library designed for the Motorola DSP56000 which was clocked at 20 MHz. The authors focused on the integration of modular reduction and multi-precision multiplication according to Montgomery's method [18, 23]. This RSA implementation achieved a data rate of 11.6 Kbits/s for a 512-bit exponentiation using the Chinese Remainder Theorem (CRT) and 4.6 Kbits/s without using it.

The authors in [15] described an ECDSA implementation over GF(p) on the M16C, a 16-bit 10 MHz microcomputer. Reference [15] proposes the use of a field of prime characteristic  $p = e^{2^c} + 1$ , where  $e$  is an integer within the machine word size and  $c$  is a multiple of the machine word size. This choice of field allows to implement multiplication in GF(p) in a small amount of memory. Notice that [15] uses a randomly generated curve with the a coefficient of the elliptic curve equal to  $p-3$ . This reduces the number of operations needed for an EC point doubling. They also modify the point addition algorithm in [24] to reduce the number of temporary variables from 4 to 2. This contribution uses a 31-entrytable of precomputed points to generate an ECDSA signature in 150 msec. On the other hand, scalar multiplication of a random point takes 480 msec and ECDSA verification 630 msec. The whole implementation occupied 4 Kbyte of code/data space. In [16], two new methods for implementing public-key cryptography algorithms on the 200 MHz TI TMS320C6201 DSP are proposed. The first method is a modified implementation of the Montgomery variant known as the Finely Integrated Operand Scanning (FIOS) algorithm [18] suitable for pipelining. The second approach suggests a method for reducing the number of multiplications and additions used to compute  $2^m P$ , for  $P$  a point on the elliptic curve and  $m$  some integer. The final code implemented RSA and DSA combined with the k-ary method for exponentiation, and ECDSA combined with the improved method for multiple point doublings, sliding window exponentiation, and signed binary exponent recoding. The total instruction code was 41.1 Kbytes. They achieved 11.7 msec for a 1024-bit RSA signature using the CRT (1.2 msec for verification assuming a 17-bit exponent) and 1.67 msec for a 192-bit ECDSA signature over GF(p) (6.28 msec for verification and 4.64 msec for general point multiplication). Recently, two papers have introduced fast implementations on 8-bit processors over Optimal Extension Fields (OEFs), originally introduced in [1]. Reference [5] reports on an ECC implementation over the field GF( $p^m$ ) with  $p = 2^{16}-165$ ,  $m = 10$ , and  $f(x) = x^{10}-2$  is the irreducible polynomial. The authors use the column major multiplication method for field multiplication and squaring, for the specific case in which  $f(x)$  is a binomial. They achieve better performance than when using Karatsuba multiplication because in this processor additions and multiplications take the same number of cycles. Modular reduction is done through repeated use of the division step instruction. For inversion, they use the variant of the Itoh and Tsujii Algorithm [17] proposed in [2]. For EC arithmetic they combine the mixed coordinate system methods of [6] and [20]. These combined methods allow them

### B. Public-Key Algorithms On Embedded Systems

In [3], the Barret modular reduction method is introduced. The author implemented RSA on the TI TMS32010 DSP. A 512-bit RSA exponentiation took on the average 2.6 seconds

to achieve 122 msec for a 160-bit point multiplication on the CalmRISC with MAC2424 math coprocessor running at 20 MHz. The second paper [32] describes a smart card implementation over the field  $GF(2^8 \cdot 17)^{17}$  without the use of a coprocessor. Reference [32] focuses on the implementation of ECC on the 8051 family of micro-controllers, popular in smart cards. The authors compare three types of fields: binary fields  $GF(2^k)$ , composite fields  $GF((2^n)^m)$ , and OEFs. Based on multiplication timings, the authors conclude that OEFs are particularly well suited for this architecture. A key idea of this contribution is to allow each of the 16 most significant coefficients resulting from a polynomial multiplication to accumulate over three 8-bit words instead of reducing modulo  $p = 2^8 \cdot 17$  after each 8-bit by 8-bit multiplication. Fast field multiplication allows the implementation to have relatively fast inversion operations following the method proposed in [2]. This, in turn, allows for the use of affine coordinates for point representation. Finally, the authors combine the methods above with a table of 9 precomputed points to achieve 1.95 sec for a 134-bit fixed point multiplication and 8.37 sec for a general point multiplication using the binary method of exponentiation. We end this section by summarizing the contributions in [13] and [30]. In [13], an ECC implementation over prime fields on the 16-bit TI MSP430x33x family of low-cost microcontrollers is described. The authors in [13] show that it is possible to implement EC cryptosystems in highly constrained embedded systems and still obtain acceptable performance at low cost. They modified the EC point addition and doubling formulae to reduce the number of intermediate variables while at the same time allowing for flexibility. In addition, [13] use Generalized-Mersenne primes to implement the arithmetic in the underlying field, taking advantage of the special form of the moduli to minimize the number of recompilations needed to implement the underlying arithmetic. These ideas are combined to achieve an EC scalar point multiplication in 3.4 seconds without any stored/precomputed values and the processor clocked at 1 MHz. The authors in [30] implemented EC over binary fields on a Motorola Dragonball CPU which is used on the popular Palm Personal Digital Assistants (PDAs). The Dragonball offers 16-bit and 32-bit operations and runs at 16 MHz. Using Koblitz curves over  $GF(2^{163})$ , [30] shows that it is possible to perform an ECDSA signature generation operation in less than 0.9 sec. while a verification operation requires less than 2.4 sec. The authors point out that Koblitz curves over fields  $GF(2^{163})$  provide about the same level of security as RSA with a 1024-bit length, while at the same time providing acceptable performance which is not possible to achieve by using RSA-based systems since the integer multiplier in the Dragonball processor is very slow.

## VI.CONCLUSIONS

We have introduced the basic concepts, characteristics, and goals of various cryptographic algorithms. We have shown how embedded systems are essential parts of most communications systems and how this makes them especially attractive as a potential platform to implement cryptographic algorithms. Furthermore, although a challenging task, previous

implementations of arithmetic intensive cryptographic algorithms seem to indicate that they can achieve acceptable performance on embedded processors and constrained platforms. Thus, it is our view that designing and implementing efficient cryptographic algorithms on embedded systems will continue to be an active research area.

## VII.REFERENCES

- [1]. D. V. Bailey and C. Paar. Optimal Extension Fields for Fast Arithmetic in Public-Key Algorithms. In H. Krawczyk, editor, Advances in Cryptology | CRYPTO '98, volume LNCS 1462, pages 472-485, Berlin, Germany, 1998. Springer-Verlag.
- [2]. D. V. Bailey and C. Paar. Efficient arithmetic in finite field extensions with application in elliptic curve cryptography. Journal of Cryptology, 14(3):153-176, 2001.
- [3]. P. Barrett. Implementing the Rivest Shamir and Adleman Public Key Encryption Algorithm on a Standard Digital Signal Processor. In A. M. Odlyzko, editor, Advances in Cryptology | CRYPTO '86, volume LNCS 263, pages 311-323, Berlin, Germany, August 1986. Springer-Verlag.
- [4]. M. Brown, D. Hankerson, J. Lopez, and A. Menezes. Software Implementation of the NIST Elliptic Curves Over Prime Fields. In D. Naccache, editor, Topics in Cryptology | CT-RSA 2001, volume LNCS 2020, pages 250-265, Berlin, April 2001. Springer-Verlag.
- [5]. Jae Wook Chung, Sang Gyoo Sim, and Pil Joong Lee. Fast Implementation of Elliptic Curve Defined over  $GF(p^m)$  on CalmRISC with MAC2424 Coprocessor. In Cetin K. Koc and Christof Paar, editors, Workshop on Cryptographic Hardware and Embedded Systems | CHES 2000, pages 57-70, Berlin, 2000. Springer-Verlag.
- [6]. Henry Cohen, Atsuko Miyaji, and Takatoshi Ono. Efficient Elliptic Curve Exponentiation Using Mixed Coordinates. In Kazuo Ohta and Dingyi Pei, editors, Advances in Cryptology | ASIACRYPT'98, volume LNCS 1514, pages 51-65, Berlin, 1998. Springer-Verlag.
- [7]. J. Daemen and V. Rijmen. AES Proposal: Rijndael. In First Advanced Encryption Standard (AES) Conference, Ventura, California, USA, 1998.
- [8]. E. DeWin, S. Mister, B. Preneel, and M. Wiener. On the Performance of Signature Schemes Based on Elliptic Curves. In J. P. Buhler, editor, Algorithmic Number Theory: Third International Symposium (ANTS 3), volume LNCS 1423, pages 252-266. Springer-Verlag, June 21-25 1998.
- [9]. W. Diffie and M. E. Hellman. New directions in cryptography. IEEE Transactions on Information Theory, IT-22:644-654, 1976.
- [10]. S. R. Dusse and B. S. Kaliski. A Cryptographic Library for the Motorola DSP56000. In I. B. Damgård, editor, Advances in Cryptology | EUROCRYPT '90, volume LNCS 473, pages 230-244, Berlin, Germany, May 1990. Springer-Verlag.
- [11]. Federal Information Processing Standards, National Bureau of Standards, U.S. Department of Commerce. NIST FIPS PUB 46, Data Encryption Standard, January 15, 1977.
- [12]. Gladman. AES Algorithm Efficiency. World Wide Web, 2002. Available at [http://fp.gladman.plus.com/cryptography\\_technology/aes/index.htm](http://fp.gladman.plus.com/cryptography_technology/aes/index.htm).
- [13]. J. Guajardo, R. Bluemel, U. Krieger, and C. Paar. Efficient Implementation of Elliptic Curve Cryptosystems on the TI MSP430x33x Family of Microcontrollers. In K. Kim, editor, Fourth International Workshop on Practice and Theory in Public Key Cryptography - PKC 2001, volume LNCS 1992, pages 365-382, Berlin, February 13-15 2001. Springer-Verlag.
- [14]. Hankerson, J. Lopez Hernandez, and A. Menezes. Software Implementation of Elliptic Curve Cryptography Over Binary Fields. In C. Koc and C. Paar, editors, Workshop on Cryptographic Hardware and Embedded Systems | CHES 2000, volume LNCS, Berlin, 2000. Springer-Verlag.
- [15]. Toshio Hasegawa, Junko Nakajima, and Mitsuru Matsui. A Practical Implementation of Elliptic Curve Cryptosystems over  $GF(p)$  on a 16-bit Microcomputer. In Hideki Imai and Yuliang Zheng, editors, First International Workshop on Practice and Theory in Public Key Cryptography | PKC'98, volume LNCS 1431, pages 182-194, Berlin, 1998. Springer-Verlag.
- [16]. K. Itoh, M. Takenaka, N. Torii, S. Temma, and Y. Kurihara. Fast Implementation of Public-Key Cryptography on a DSP TMS320C6201. In Cetin K. Koc and Christof Paar, editors, Workshop on Cryptographic

- Hardware and Embedded Systems | CHES'99, volume LNCS 1717, pages 61-72, Berlin, Germany, August 1999. Springer-Verlag.
- [17]. T. Itoh and S. Tsujii. A fast algorithm for computing multiplicative inverses in  $GF(2^m)$  using normal bases. *Information and Computation*, 78:171-177, 1988.
- [18]. K. Koc, T. Acar, and B. Kaliski, Jr. Analyzing and Comparing Montgomery Multiplication Algorithms. *IEEE Micro*, pages 26-33, June 1996.
- [19]. X. Lai and J. L. Massey. Markov Ciphers and Differential Cryptanalysis. In D. W. Davies, editor, *Advances in Cryptology | EUROCRYPT '91*, volume LNCS 547, pages 17-38, Berlin, Germany, 1991. Springer-Verlag.
- [20]. Chae Hoon Lim and Hyo Sun Hwang. Fast Implementation of Elliptic Curve Arithmetic in  $GF(pn)$ . In Hideki Imai and Yuliang Zheng, editors, *Third International Workshop on Practice and Theory in Public Key Cryptography | PKC 2000*, volume LNCS 1751, pages 405-421, Berlin, 2000. Springer-Verlag.
- [21]. J. L. Massey and X. Lai. Device for converting a digital block and the use thereof. European Patent, Patent Number 482154, April 29, 1992.
- [22]. J. Menezes, P. C. van Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, Boca Raton, Florida, USA, 1997.
- [23]. P. L. Montgomery. Modular multiplication without trial division. *Mathematics of Computation*, 44(170):519-521, April 1985.
- [24]. IEEE P1363 Standard Specifications for Public Key Cryptography, November 1999. Last Preliminary Draft.
- [25]. R. L. Rivest, A. Shamir, and L. Adleman. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM*, 21(2):120-126, February 1978.
- [26]. Schneier. *Applied Cryptography*. John Wiley & Sons Inc., New York, New York, USA, 2nd edition, 1996.
- [27]. W. Stallings. *Cryptography and Network Security*. Prentice Hall, Upper Saddle River, New Jersey, USA, second edition, 1999.
- [28]. U.S. Department of Commerce/National Institute of Standard and Technology. FIPS 186-2, Digital Signature Standard (DSS), February 2000. Available at <http://csrc.nist.gov/encryption>.
- [29]. U.S. Department of Commerce/National Institute of Standard and Technology. FIPS PUB 197, Specification for the Advanced Encryption Standard (AES), November 2001. Available at <http://csrc.nist.gov/encryption/aes>.
- [30]. Weimerskirch, C. Paar, and S. Chang Shantz. Elliptic Curve Cryptography on a Palm OS Device. In V. Varadharajan and Y. Mu, editors, *The 6th Australasian Conference on Information Security and Privacy | ACISP 2001*, volume LNCS 2119, pages 502-513, Berlin, 2001. Springer-Verlag.
- [31]. T. Wollinger, M. Wang, J. Guajardo, and C. Paar. How well are high-end DSPs suited for the AES algorithms? In *The Third Advanced Encryption Standard Candidate Conference*, pages 94-105, New York, New York, USA, April 13-14 2000. National Institute of Standards and Technology.
- [32]. A. Woodbury, D. V. Bailey, and C. Paar. Elliptic curve cryptography on smart cards without coprocessors. In *IFIP CARDIS 2000, Fourth Smart Card Research and Advanced Application Conference*, Bristol, UK, September 20-2 2000. Kluwer.

# Understanding Programmable Logic Controllers (PLCs) and Programmable Automation Controllers (PACs) in Industrial Automation.

K.R.Patel, A.B.Patel, M.L. Institute of Diploma Studies, Bhandu.

[Kirti3183@yahoo.co.in](mailto:Kirti3183@yahoo.co.in), [amit\\_b Patel82@yahoo.com](mailto:amit_b Patel82@yahoo.com)

**Abstract-** Programmable Logic Controllers (PLC) and Programmable Automation Controllers (PAC) continue to evolve as new technologies are added to their capabilities. Today they are the brain vast majority of Automation, Process and special machines. This paper introduces PLC, PAC and its bidirectional correspondences. This paper also will highlight Analog I/O, its types and functions; Digital I/O and its types and functions; In parallel this paper also introduces various PROTOCOLS and its supporting agents for communication.

In this paper we have tried to project the differentiation between PLC and PAC. To sum up we would like to explain some selection criteria for implementing this system practically.

**Index Terms—** Analog I/P, Digital I/O, PAC, PLC, Protocols.

## I. INTRODUCTION OF PLC

Programmable Logic Controllers (PLCs) are digital devices that are used to control the state of output ports based on the state of input ports. They greatly simplify and increase the capability of relay logic control systems.



Fig : - 1 Allen Bradley PLC

## II. PRINCIPLES OF MACHINE CONTROL

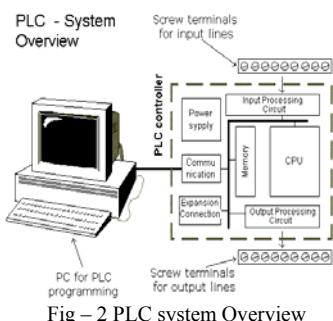


Fig – 2 PLC system Overview

The controller consists of a built-in power supply, central processing unit (CPU), inputs, which you wire to input devices (such as pushbuttons, proximity sensors, limit switches), and

outputs, which you wire to output devices (such as motor starters, solid-state relays, and indicator lights).

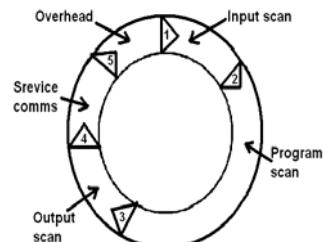


FIG - 3 operating cycle

With the logic program entered into the controller, placing the controller in the Run mode initiates an operating cycle. The controller's operating cycle consists of a series of operations performed sequentially and repeatedly, unless altered by your program logic.

1. Input scan – the time required for the controller to scan and read all input data; typically accomplished within microseconds.
2. Program scan – the time required for the processor to execute the instructions in the program. The program scan time varies depending on the instructions used and each instruction's status during the scan time.

*Subroutine and interrupt instructions within your logic program may cause deviations in the way the operating cycle is sequenced.*

3. Output scan – the time required for the controller to scan and write all output data; typically accomplished within milliseconds.
4. Service communications – the part of the operating cycle in which communication takes place with other devices, such as an HHP or personal computer.
5. Housekeeping and overhead – time spent on memory management and updating timers and internal registers.

You enter a logic program into the controller using a programming device. The logic program is based on your electrical relay print diagrams. It contains instructions that direct control of your application

## III. INTRODUCTION OF PAC

Automation manufacturers have responded to the modern industrial application's increased scope of requirements with industrial control devices that blend the advantages of PLC-style deterministic machine or process control with the flexible configuration and enterprise integration strengths of PC-based systems. Such a device has been termed a programmable automation controller, or PAC. While the idea of combining

PLC and PC-based technologies for industrial control has been attempted previously, it has usually only been done through the “add-on” type of approach described earlier, where additional middleware, processors, or both are used in conjunction with one or more PLCs. A PAC, however, has the broader capabilities needed built into its design. For example, to perform advanced functions like counting, latching, PID loop control, and data acquisition and delivery, a typical PLC-based control system requires additional, and often expensive, processing hardware. A PAC has these capabilities built in. A PAC is notable for its modular design and construction, as well as the use of open architectures to provide expandability and interconnection with other devices and business systems. In particular, PACs are marked both by efficient processing and I/O scanning, and by the variety of ways in which they can integrate with enterprise business systems.

#### IV. APPLYING THE PAC TO A MODERN INDUSTRIAL APPLICATION

Let's look more closely at how a PAC is applied to a modern industrial application, using the factory application illustrated in Fig – 4

##### *Single Platform Operating in Multiple Domains*

The single PAC shown in the example is operating in multiple domains to monitor and manage a production line, a chemical process, a test bench, and shipping activities. To do so, the PAC must simultaneously manage analog values such as temperatures and pressures; digital on/off states for valves, switches, and indicators; and serial data from inventory tracking and test equipment. At the same time, the PAC is exchanging data with an OLE for Process Control (OPC) server, an operator interface, and a SQL (Structured Query Language) database. Simultaneously handling these tasks without need for additional processors, gateways, or middleware is a hallmark of a PAC.

##### *Support for Standard Communication Protocols*

In the factory example , the PAC, operator and office workstations, testing equipment, production line and process sensors and actuators, and barcode reader are connected to a standard 10/100 Mbps Ethernet network installed throughout the facility. In some instances, devices without built-in Ethernet connectivity, such as temperature sensors, are connected to I/O modules on an intermediate Ethernet-enabled I/O unit, which in turn communicates with the PAC. Using this Ethernet network, the PAC communicates with remote racks of I/O modules to read/write analog, digital, and serial signals. The network also links the PAC with an OPC server, an operator interface, and a SQL database. A wireless segment is part of the network, so the PAC can also communicate with mobile assets like the forklift and temporary operator workstation. The PAC can control, monitor, and exchange data with this wide variety of devices and systems because it uses the same standard network technologies and protocols that they use. This example includes wired and wireless Ethernet networks, Internet Protocol (IP) network transport, OPC, and SQL. In another control situation, common application-level protocols such as Modbus®, SNMP (Simple Network

Management Protocol), and PPP (point-to-point protocol) over a modem could be required. The PAC has the ability to meet these diverse communication requirements.

##### *Exchange Data with Enterprise Systems*

In the factory example, the PAC exchanges manufacturing, production, and inventory data with an enterprise SQL database. This database in turn shares data with several key business systems, including an enterprise resource planning (ERP) system, operational equipment effectiveness (OEE) system, and supply chain management (SCM) system. Because data from the factory floor is constantly and automatically updated by the PAC, timely and valuable information is continually available for all business systems.

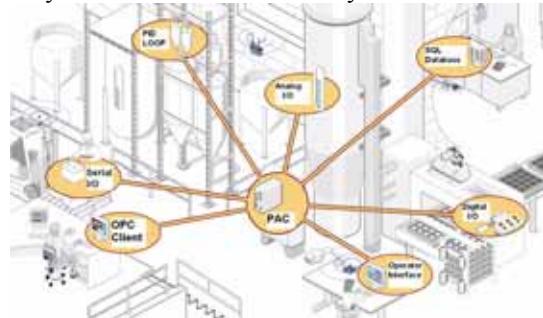


Fig:- 4 This modern industrial application encompasses multiple tasks requiring I/O point monitoring and control, data exchange via OPC, and integration of factory data with enterprise systems.

#### V. ANALOG INPUT

Analog signals are like volume controls, with a range of values between zero and full-scale. These are typically interpreted as integer values (counts) by the PLC, with various ranges of accuracy depending on the device and the number of bits available to store the data. As PLCs typically use 16-bit signed binary processors, the integer values are limited between -32,768 and +32,767. Pressure, temperature, flow, and weight are often represented by analog signals. Analog signals can use voltage or current with a magnitude proportional to the value of the process signal. For example, an analog 4-20 mA or 0 - 10 V input would be converted into an integer value of 0 – 32767.

RTD : Output in Ohms (Temperature)

Thermocouples : Output in mV (Temperature)

Pressure Transmitters : 4-20mA, 0-10 V .....

Flow Transmitter : 4-20mA, 0-10 V .....

Level Transmitter : 4-20mA, 0-10 V .....

Conductivity meter : 4-20mA, 0-10 V .....

Density meter : 4-20mA, 0-10 V .....

pH transmitter : 4-20mA, 0-10 V .....

Current inputs are less sensitive to electrical noise (i.e. from welders or electric motor starts) than voltage inputs.

#### VI. DIGITAL I/O

Digital or discrete signals behave as binary switches, yielding simply an On or Off signal (1 or 0, True or False, respectively). Push buttons, limit switches, and photoelectric sensors are examples of devices providing a discrete signal. Discrete signals are sent using either voltage or current,

where a specific range is designated as *On* and another as *Off*. For example, a PLC might use 24 V DC I/O, with values above 22 V DC representing *On*, values below 2VDC representing *Off*, and intermediate values undefined. Initially, PLCs had only discrete I/O.

## VII. COMMUNICATION PROTOCOLS

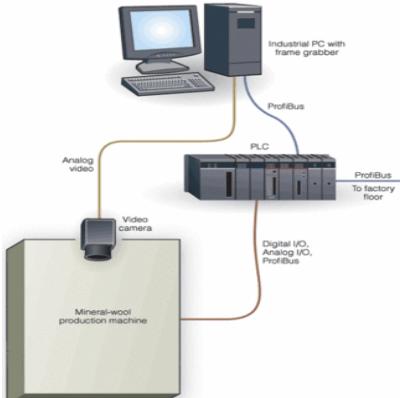


Figure 5 Communication Interface

PLCs have built in communications ports usually

- RS232
- RS485
- Ethernet
- Modbus
- BACnet or DF1
- Profibus - by PROFIBUS International.
- ControlNet - an implementation of CIP, originally by Allen-Bradley
- DeviceNet - an implementation of CIP, originally by Allen-Bradley
  - EtherNet/IP - IP stands for "Industrial Protocol". An implementation of CIP, originally created by Rockwell Automation
  - Modbus RTU or ASCII
  - Modbus-NET - Modbus for Networks
  - Modbus/TCP
  - Modbus Plus

Most modern PLCs can communicate over a network to some other system, such as a computer running a SCADA (Supervisory Control and Data Acquisition) system or web browser.

PLCs used in larger I/O systems may have peer-to-peer (P2P) communication between processors. This allows separate parts of a complex process to have individual control while allowing the subsystems to co-ordinate over the communication link. These communication links are also often used for HMI devices such as keypads or PC-type workstations. Some of today's PLCs can communicate over a wide range of media including RS-485, Coaxial, and even Ethernet.

## OPC

*OLE for Process Control (OPC)* which stands for Object Linking and Embedding (OLE) for Process Control. The **OPC Specification** was based on the OLE, COM, and DCOM technologies developed by Microsoft for the Microsoft Windows operating system family. The specification defined a standard set of objects, interfaces and methods for use in process control and manufacturing automation applications to facilitate interoperability.

OPC was designed to bridge Windows based software applications and process control hardware. Standard defines consistent method of accessing field data from plant floor devices. This method remains the same regardless of the type and source of data.

OPC servers provide a method for many different software packages to access data from a process control device, such as a PLC or PAC. Traditionally, any time a package needed access to data from a device, a custom interface, or driver, had to be written. The purpose of OPC is to define a common interface that is written once and then reused by any business, SCADA, HMI, or custom software packages.

Once an OPC server is written for a particular device, it can be reused by any application that is able to act as an OPC client. OPC servers use Microsoft's OLE technology (also known as the Component Object Model, or COM) to communicate with clients. COM technology permits a standard for real-time information exchange between software applications and process hardware to be defined.

## VIII. DIFFERENTIATION BETWEEN PAC & PLC

A programmable automation controller (PAC) is a compact controller that combines the features and capabilities of a PC-based control system with that of a typical programmable logic controller (PLC). Therefore, the PAC features not only reliability as a PLC but flexibility and power as a PC at the same time. PACs are most often used in industrial settings for process control, data acquisition, remote equipment monitoring, machine vision, and motion control. Additionally, because they function and communicate over popular network interface protocols like TCP/IP, OLE for process control (OPC) and SMTP, PACs are able to transfer data from the machines they control to other machines and components in a networked control system or to application software and databases.

PLC (programmable logic controller) a PLC is dedicated standalone microcontroller that is hardened against the harsh industrial operational environments. (vibration, electrical noise). It has some limited computational power compared to a PC based systems.

PAC (programmable automation controller) - is like a combination of a PLC with a "PC based" automation system. The PAC has a smarter brain than the PLC. There is an increased amount of computational and communication power that you would expect to find in a PC. But unlike a regular PC, is it hardened against the harsh industrial operational environment.

## IX. SELECTION CRITERIA FOR IMPLEMENTING THIS SYSTEM PRACTICALLY

We implemented this system on one small – scale textile industry. A Project focused on during raw wet saree material having 72 meters length (approx) with the help of a boiler. Here the boiler converted water into hot- dry gas, at exceeding the limit of given temp. A heater was used to increase the temperature the moment water was transformed into hot dry gas valve was opened, for free flow of gas on that wet saree material passing across its pathway to make it dry.

Two sensors were used (which we call ‘proximity’) for precise calculation of meter of dry saree material. We used limit switch to see how much wet material is still left to be dried. The time when the limit switch indicated, that now no material had been left to be dried, we switched off the system & get the whole material dried.

### **Selections of system component are as follow**

**PLC:** Allen Bradley Micrologix 1000 (Analog/Digital)

**PROGRAMMING SOFTWARE:** RSLOGIX 500

**SCADA:** Intouch 9.0 (Wonderware) Demo

**COMMUNICATION PROTOCOL:** RS232 (9 PIN)

**COMMUNICATION DRIVER:** RSLINX CLASSIC

**ANALOG INPUT:**

**TEMPERATURE :** THERMO COUPLE

**DIGITAL INPUT:**

**SENSOR:** NPN PROXIMITY (2 Pcs)

: LIMIT SWITCH (1 Pcs.)

**DIGITAL OUTPUT:**

**RELAY:** 24 V DC ( 2 Pcs.)

**VALVE:** 24 V DC

## X. REFERENCES

- [1]. www.ab.com
- [2]. www.automation.com
- [3]. PLC Architecture & Application by Giles Michael
- [4]. PLC hardware, software and Application by G. L. Batten
- [5]. www.wikipedia.com

# Synthetic Biology: Towards Digital Circuits

Shruti Jain, Department of Electronics and Communication Engineering

Pradeep.K. Naik, Department of Biotechnology & Bioinformatics

Jaypee University of Information Technology, Solan-173215, India, E-mail<sup>1</sup>: [juit.shruti@gmail.com](mailto:juit.shruti@gmail.com)

**Abstract :** Today most VLSI circuits are built in silicon using CMOS transistors. Developments in design automation and process fabrication have resulted in the progressive increase of the number of transistors per chip and decrease in the size of the transistors. Thus, engineers are looking at alternate technologies such as nano-devices and bio-circuits for next-generation circuits. Our eventual goal is the design and simulation of complete systems integrating bio-circuits and VLSI technology appropriately. Bio-circuits are circuits developed *in vivo* or *in vitro*, using DNA and proteins. A biological process such as glycolysis or bioluminescence can be viewed as a genetic regulatory circuit, a complex set of bio-chemical reactions regulating the behavior of genes, operon, DNA, RNA, and proteins. Similar to voltage in an electrical circuit, a genetic regulatory circuit produces an output protein in response to an input stimulus. This paper is intended to pave the pathway for electrical engineers to start exploring the field of bio-circuits. Biological systems can create complex structures from very simple systems. To do this, there must be a method to differentiate different regions where identical systems create different structures, such as the abdomen and the head of a fruit fly. This paper highlights an emerging field known as synthetic biology that envisions integrating designed circuits into living organisms in order to instruct them to make logical decisions based on the prevailing intracellular and extra cellular conditions and produce a reliable behavior. The networks in a system can be dissected into small regulatory gene circuit modules. Synthetic biology attempts to construct and assemble such modules gradually, plug the modules together and modify them, in order to generate a desired behavior. Using biological circuit, we can produce a new concentration gradient that has twice the frequency. If 1 is represented by a concentration of the chemical within the threshold, and a 0 is represented by a concentration outside the threshold, then we can represent any two digit binary number. Thus, we can differentiate separate regions at certain distances away from a point source

**Keywords :** Biocircuits, VLSI, Digital Circuits, AND, OR, NOT

## I. INTRODUCTION

Biological systems have the immense capability to generate complex structures from very simple systems[1, 2, 3]. With simple rules and few inputs, a biological system can grow from a single cell to a multicellular organism in a relatively short amount of time. Biological systems [4, 5], however, have been accurately synthesizing nano scale machines for millions of years. Logic gates are the basic building blocks in electronic circuits [6, 7] that perform logical operations. These have input and output signals in the form of 0's and 1's; '0' signifies the absence of signal while '1' signifies its presence. Similar to the electronic logic gates, cellular components can serve as logic gates. A typical biological circuit [8, 9, 10] consists of i) a coding region, ii) its promoter, iii) RNA polymerase and the iv) regulatory proteins with their v) DNA binding elements, and

vi) small signaling molecules that interact with the regulatory proteins. Messenger RNA [11, 12] or their translation products can serve as input and output signals to the logic gates formed by genes with which these gene products interact. The concentration of the gene product determines the strength of the signal. High concentration indicates the presence of signal (=1) whereas low concentration indicates its absence (=0).

The focus of these early studies was on using digital models as a way of understanding events in the living cell. The Boolean approximation was a way of avoiding an unwieldy analysis of a complex chemical web. To follow all those molecular interactions in complete detail would have required tracking the concentrations of innumerable molecular species, measuring the rates of chemical reactions, and solving hundreds of coupled differential equations. Either pretending that every gene is on or off reduced the problem to a simpler digital abstraction. Silicon circuits perform complex operations using a handful of simple components known as logic gates. Genetic-circuit engineers are now building the same devices inside cells.

## II. THE LOGIC OF LIFE

A digital technology usually starts with Boolean logic gates—devices that operate on signals with two possible values, such as *true* and *false*, 1 and 0. There are three basic logic gates: **AND**, **OR**, **NOT**.

### 2.1 An AND GATE :

An **AND** gate has two or more inputs and one output; the output is *true* only if all the inputs are *true*. The symbol, switching circuit and truth table for AND gate shown in Fig. 1

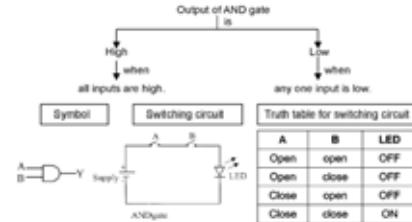


Fig.1 Switching Circuit and Truth table for AND gate

### 2.2 An OR GATE :

An **OR** gate is similar except that the output is *true* if any of the inputs are true. The symbol, switching circuit and truth table for OR gate is shown in Fig.2

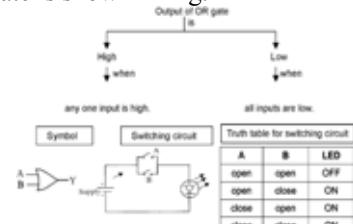


Fig.2 Switching Circuit and Truth table for OR gate

### 2.3 NOT GATE :

The simplest of all gates is the *NOT* gate, which takes a single input signal and produces the opposite value as output: *true* becomes *false*, and *false* becomes *true*. The symbol, switching circuit and truth table for NOT gate is shown in Fig.3

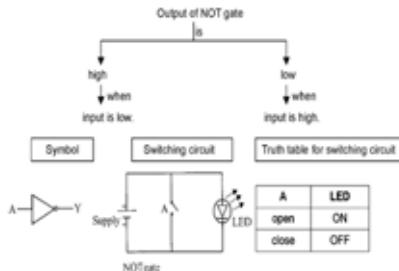


Fig. 3 Switching Circuit and Truth table for NOT gate

The archetypal example of genetic regulation in bacteria is the *lac* operon of *E. coli*, first studied in the 1950s. The operon [13, 14, 15, 16] is a set of genes and regulatory sequences involved in the metabolism of certain complex sugars, including lactose. The bacterium's preferred nutrient is the simpler sugar glucose, but when glucose is scarce, the cell can make do by living on lactose. The enzymes for digesting lactose are manufactured in quantity only when they are needed—specifically when lactose is present and glucose is absent.

As in the expression of any genes, synthesis of the *lac* enzymes is a two-stage process. First the DNA is transcribed into messenger RNA by the enzyme RNA polymerase; then the messenger RNA is translated into protein by ribosome. The process is controlled at the transcription stage [14, 15, 16]. Before the genes can be transcribed, RNA polymerase must bind to the DNA at a special site called a promoter, which is just "upstream" of the genes; then the polymerase must travel along one strand of the double helix, reading off the sequence of nucleotide bases and assembling a complementary strand of messenger RNA. One mechanism of control prevents transcription by physically blocking the progress of the RNA polymerase molecule. The blocking is done by the *lac* repressor protein, which binds to the DNA downstream of the promoter region and stands in the way of the polymerase. When lactose enters the bacterial cell, the *lac* operon is released from this restraint. A metabolite of lactose binds to the *lac* repressor, changing the protein's shape and thereby causing it to loosen its grip on the DNA. As the repressor protein drifts away, the polymerase is free to march along the strand and transcribe the operon.

The repressor system is only half of the *lac* control strategy. Even in the presence of lactose, the *lac* enzymes are synthesized only in trace amounts if glucose is also available in the cell. The reason, it turns out, is that the *lac* promoter site is a feeble one, which does a poor job of attracting and holding RNA polymerase. To work effectively, the promoter requires an auxiliary molecule called an activator protein, which clamps onto the DNA and makes it more receptive. Glucose causes the activator to fall away from the DNA just as lactose causes the

repressor to let go—but the ultimate effect is the opposite. Without the activator, the *lac* operon lies dormant. All these tangled interactions of activators and repressors can be simplified by viewing the control elements of the operon as a logic gate. The inputs to the gate are the concentrations of lactose and glucose in the cell's environment. The output of the gate is the production rate of the three *lac* enzymes. The gate computes the logical function: (lactose AND (NOT glucose)).

### III. RESULTS AND DISCUSSIONS

#### 3.1 AND GATE

Let there are two inputs (A and B) in Fig 3. For this, we apply to pulse input V1 and V2. There are two diodes for two input gate. V1 voltage is for D1 and V2 voltage is diode D2. First case when both inputs A and B are 0 then diodes D1 and D2 both are forward biased, both will conduct and current flows from V<sub>CC</sub> to ground. No current flows to the output. Hence, output is 0. When input is (0 1), then D1 is forward bias and D2 is reverse bias, D2 does not conduct while D1 conducts. Current flows from V<sub>CC</sub> to ground but does not go to the output. Hence, output is logic 0. When input is (1 0), then D1 is reverse bias and D2 is forward bias, D1 does not conduct while D2 conducts. Current flows from V<sub>CC</sub> to ground but does not go to the output. Hence, output is logic 0. When input is (1 1), then D1 and D2 are reverse bias, both will not conduct. Whole V<sub>CC</sub> appears of the output. Hence output is logic 1 (=V<sub>CC</sub>). Inputs and Outputs are shown in Fig. 5, and Fig.6

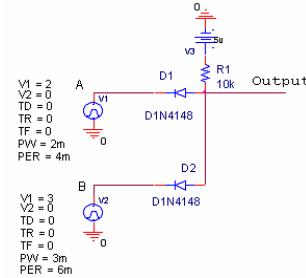


Fig. 4: Circuit for AND gate

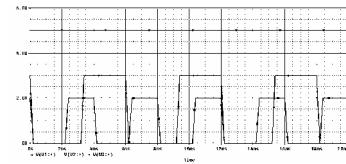


Fig. 5 : Input given to the AND circuit

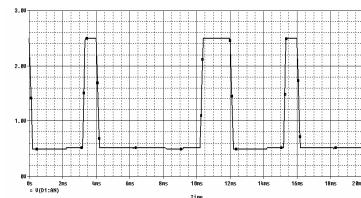


Fig. 6 : Output for AND gate

#### 3.2 OR GATE

Let there are two inputs (A and B) shown in Fig. 7. For this, we apply to pulse input V1 and V2. There are two diodes for two input gate. V1 voltage is for D1 and V2 voltage is diode D2

diode D2. First case when both inputs A and B are 0 then diodes D1 and D2 both are reverse biased, so does not conduct. No current flows through R. No voltage drop across R. Hence, output is 0. When input is (0 1), then D1 is reverse bias and D2 is forward bias, D1 does not conduct while D2 conducts. Current flows from B to R through D2. Hence output is logic 1 ( $=V_{CC}$ ). When input is (1 0), then D1 is forward bias and D2 is reverse bias, D2 does not conduct while D1 conducts. Current flows from A to R through D1. Hence output is logic 1 ( $=V_{CC}$ ). When input is (1 1), then D1 and D2 are forward bias, both will conduct. Current flows through A and B to R through D1 and D2 respectively. Hence output is logic 1 ( $=V_{CC}$ ). Inputs and Outputs are shown in Fig. 8, and Fig.9

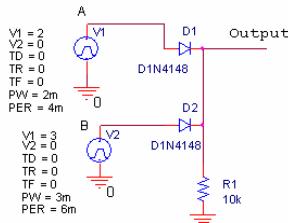


Fig. 7 : Circuit for OR gate

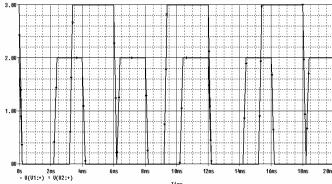


Fig. 8 Input given to the OR circuit

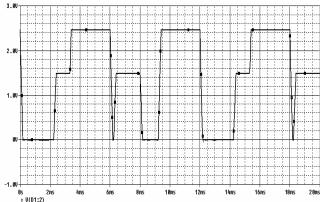


Fig. 9 Output for OR gate

### 3.3 NOT GATE

When Base Emitter junction of Q is reverse bias shown in Fig.10 , whole VCC appears across R, output is 1. When Base Emitter junction of Q is forward bias, current flows from  $V_{CC}$  to ground through Q. Hence, the output is 1. Inputs and Outputs are shown in Fig. 11

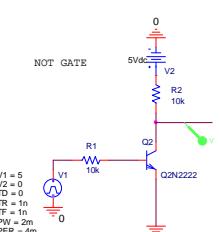


Fig.10 : Circuit for NOT gate

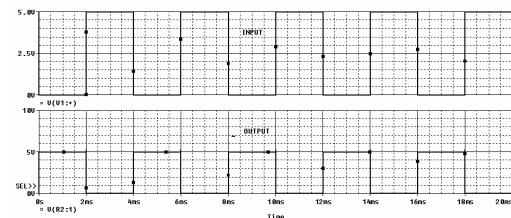


Fig. 11 : Input and Output for the corresponding circuit

### IV. CONCLUSION

We had successfully designed the basic gates in PSPICE. Next is to use all the basic gates to get the circuit which tells whether there is cell survival or cell death. Synthetic biology attempts to construct and assemble such modules gradually, plug the modules together and modify them, in order to generate a desired behavior. We can engineer bio-circuits to meet design specifications, using genetic engineering. This approach is similar to design space exploration in traditional VLSI, but takes into account biological knowledge obtained through experiments. We also provide insight into the robustness of bio-circuits in the presence of noise.

### V. REFERENCES

- [1] Campbell, M. L. (2002). Cell modeling. M.S. Thesis, Air Force Institute of Technology, WPAFB.
- [2] Belta, C., Schug, J., Thao, D., Kumar,V., Pappas,G. J., Rubin, H.,& Dunlap, P. (2001). Stability and reachability analysis of a hybrid model of luminescence in the marine bacterium *Vibrio fischeri*. 40<sup>th</sup> IEEE Conference on Decision and Control, 1(4), 869–874.
- [3] Weiss, R. (2001). Cellular computation and communications using engineered genetic regulatory networks. PhD Thesis, MIT, September 2001.
- [4] Weiss, R., Homsy, E., Knight, F. (1999). Toward *in vivo* digital circuits. DIMACS Workshop on Evolution as Computation, January 1999.
- [5] Ptashne, M. (1986). A genetic switch: gene control and phage [lambda]. Cambridge, MA: Cell Press, Blackwell Scientific Publications.
- [6] Hasty, J., McMillen, D., & Collins, J. J. (2002). Engineered gene circuits. *Nature*, 420(6912), 224–230.
- [7] Atsumi, S., & Little, J. W. (2006). A synthetic phage {lambda} regulatory circuit. *PNAS*, 103(50), 19045–19050.
- [8] Guet, C., Elowitz, M. B., Hsing, W., & Leibler, S. (2002). Combinatorial synthesis of genetic networks. *Science*, 296(5572), 1466–1470.
- [9] Elowitz, M. B., & Leibler, S. (2000). A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767), 335–338.
- [10] Bhalla, U. S., & Iyengar, R. (1999). Emergent properties of networks of biological signaling pathways. *Science*, 283(5400), 339–340.
- [11] Li, G., Rosenthal, C., & Rabitz, H. (2001). High dimensional model representations. *Journal of Physical Chemistry*, 105(33), 7765–7777.
- [12] R Weiss, S Basu, *Device Physics of Cellular Logic Gates*, First workshop on non-silicon computing, Boston, MA, 2002.
- [13] R Weiss, S Basu, S Hooshangi, A Kalmbach, D Karig, R Mehreja and I Netravali, Genetic circuit building blocks for cellular computation, communications, and signal processing, *Natural Computing*, pp.47–84, 2003.
- [14] T S Gardner, CR Cantor and J J Collins, Construction of genetic toggle switch in *Escherichia coli*, *Nature*, Vol.403, pp. 339–342, 2000.
- [15] MB Elowitz, S Leibler, A synthetic oscillatory network of transcription regulators, *Nature*, Vol.403, pp. 335–338, 2000.
- [16] Krishnan, Rajesh. et al., “Circuit development using biological components: Principles, models and experimental feasibility,” *Analog Integr Circ Sig Process* (2008) 56: 153-161.

# To Improve the Performance of Outsource Testing

Parul Gupta<sup>1</sup> Tripti Sharma

Ideal institute of Technology, Ghaziabad (UPTU, Lucknow), parul.bmiet@gmail.com<sup>1</sup>, tr\_sharma27@yahoo.co.in<sup>2</sup>

**Abstract-** As organizations expand globally, demands on IT departments increase exponentially. To meet these demands, many companies outsource IT services offshore. The traditional focus has been on cost savings due to the increased availability of lower cost skill sets in global locations. Companies often transition offshore aggressively to capture cost savings quickly in response to declining margins. Outsourcing firms offer other advantages and benefits that may have been previously overlooked, including cost savings. This paper provides an overview of the flaws of global outsourcing and how we can overcome those flaws. The common flaw in global sourcing programs are: excessive focus on costs is commonly undermining global sourcing relationships and wasting important opportunities for companies to improve quality of service, increase productivity and shorten time to market.

**Keywords:** Outsourcing, global IT outsourcing, off shoring, cost saving, quality of service.

## I. INTRODUCTION

Outsourced testing is on the verge of a major boom. Quality is expensive for organizations to invest in, and outsourcing is making inroads in several discrete areas of testing. Companies should look to outsource testing as a quality gain, not just a cost savings. Higher application quality, greater flexibility in staffing, and, yes, reduced staffing and tool carry costs are the primary benefits provided by testing outsourcing. However, organizations must realize that outsourcing can result in the loss of customer private information, increased collaboration time between development and testing, and may require process changes to core development activities.

In this paper we recommend following Forrester's testing outsourcing decision path to determine what testing to outsource, focusing on certain core competencies when choosing an outsourcer. IT's continuing push towards cost reduction has led to the burgeoning of the outsourcing industry.

## II. STRATEGIES FOR GLOBAL IT OUTSOURCING.

Cost savings, however, are a one-time return. Companies are now looking beyond the one-time, cost-based decision to move offshore and are broadening the scope of their global sourcing strategy. It is common wisdom that if an activity is not a core competency of an individual, team or company, you either build up your skill set or have an expert third party perform the work for you.

Let us take an example of laundry. Separate the dirty clothes by color and sort out the delicate. Don't forget to empty the pockets, if you don't want to launder money. The actual act of washing is simple. Load the pile of clothes into the washing machine, choose your cycle, and start. When it comes to ironing though, I am clueless. I will spend half an hour pressing a shirt and it will still be full of wrinkles. No wonder I hate it. I therefore started buying easy-iron or non-iron shirts. While the latter don't quite live up to their name, it did make the job somewhat easier. But even then some

creases remained and I still have a number of "legacy" garments in my closet. The logical conclusion was therefore to outsource this activity to the dry cleaners. Similarly, testing is an essential part of software development, but it is not the core competency of most companies. As of 2008, the (U.S.) average for defect removal efficiency is about 85%, meaning that 15% of total defects are found after software applications are delivered to customers. Best in class organizations have defect removal efficiencies that top 95 percent [1]. "Wrinkled" software though is becoming less and less acceptable in an ever more demanding and competitive market, considering that it costs at least ten times more to fix a bug once the product is shipped than to fix it in the development process, damages to the company image aside.

So, why not outsource software testing?

### 2.1 Benefits of Outsource software testing

Outsourced Testing is a surprisingly wide-ranging subject area, as there are a multitude of testing tasks that can be carried out by a third party, from test strategy development to penetration testing. While acquiring testing services from outside contractors isn't unusual, its importance has increased considerably, as have the importance and maturity of testing itself. This is shown by the fact that the test outsourcing market is no longer driven solely by independent testing companies. All the major outsourcing companies now offer testing services as well. Through outsourced testing, the following main benefits can be achieved:

#### 2.1.1. Speed up testing:

Test execution, located at the end of the software development life cycle can be a bottleneck, especially if development is already late. To make the most of the time available, a thorough test preparation and readily available testing staff are needed. These resources can be procured by an outsourcing provider. Also, in some cases testing can be sped up by taking advantage of the time zone differences, testing at night what has been developed or debugged during the day.

#### 2.1.2. Improve quality:

The need to adhere to a formal testing process and to provide a proper test basis in the form of well-documented and stable requirements will result in finding more bugs, therefore improving the quality of the end product. In addition, some of the bugs will be found earlier, when analyzing the requirements or reviewing the test cases.

#### 2.1.3. Reduce testing costs:

The main driver for outsourcing is often cost reduction, by moving operations to low-cost countries. Also in testing,

costly IT and business resources can be freed from repetitive and time-consuming test activities as, for example, regression testing. The improved software quality can also lead to cost savings.

Trends in the IT industry for the past few years demonstrate that companies are seeking to reap the benefits of outsourcing IT application development. Countless examples exist of corporations outsourcing their software development needs, call centers, data centers, hardware purchases, system support, help desks, etc. Testing is another area of IT that is rapidly being outsourced. Companies are outsourcing test case executions, test script automation, and test case development tasks to offshore based companies, independent contractors, niche QA companies, and system integrators. Outsourcing approaches vary widely. Some companies outsource manual testing needs, while other companies outsource testing tasks. Regardless of the approach, outsourcing a company's testing needs can militate strongly in a company's favor by lowering costs while delivering reliable testing results. The following sections explore the benefits of outsourcing testing.

#### *2.1.4. Plug-In for Temporary Assignments:*

Some companies experience demand for testing services that exceeds the capability of the existing testing team. Even when the company has a large testing team, it may not have the bandwidth or expertise to take on ad hoc testing tasks (i.e. a capacity test for measuring the response times for a new GUI, hardware component, or LAN). Employing an outsourcing firm to handle surges or increases in demand for ad-hoc testing tasks provides a practical solution for test teams that cannot support such efforts.

#### *2.1.5. Automation:*

Many companies regard test tools, or test automation, as a foreign and esoteric subject. Even companies that have invested hundreds of thousands of dollars in test tools struggle because they don't have the properly trained in-house resources to implement these test tools. Sometimes the test tools are not even suitable for their intended environment (i.e. a test tool for a CRM web based system may not support a Mainframe environment). Another common problem with purchasing test tools is resistance to change that compels many companies to conduct their regression and functional tests by hand because the testing team is resistant to the tools.

In contrast to these problems with test tools, outsourcing firms own licenses to a variety of test tools from different vendors and have testers who are savvy in test automation. Outsourcing firms understand what can be automated and how it should be automated. Automation of business processes and test cases is critical to providing consistent and repeatable test results; many outsourcing firms are capable of providing this service.

#### *2.1.6. Minimize Costs*

Hiring full-time testers involves providing them with company benefits and training which is a costly and time-consuming venture. Furthermore, recruiting costs for qualified full-time testers are additional. Companies looking to cut costs should seriously look into outsourcing as a cheaper alternative. The cost of employing a tester from an outsourcing firm with specific testing experience in a given industry or with a particular test tool is far cheaper than the cost of hiring in-house testers.

Outsourcing firms lower testing costs by offering testers and testing solutions at a percentage of the cost of hiring full-time testers. Furthermore, outsourcing firms have libraries and repositories of automated tests that can be leveraged or recycled for other testing needs, which also lower the costs of various test automation tasks.

### **2.2. What to outsource?**

Outsourcing resource- and time-consuming test activities, not only allows for greater capacity, scalability and flexibility, but also has the additional benefit of an independent validation. By that I mean testing that is not carried out by the development team itself (typically unit & integration testing), but by a separate test team. But greater and/or cheaper capacity isn't the only reason for outsourced testing and a strong case can be made for outsourcing testing tasks that require specialist know-how, for which you don't have the skills or only need on a one-off basis [5]. When deciding on what parts of testing to outsource, you will have to look at it from different angles, mainly at the dimensions of test levels, test types and test activities:

- **Test Levels:** Low-level testing (Unit & Integration test) is generally carried out by the developers themselves. If development is outsourced, these activities are outsourced with it. System testing on the other hand can be performed by an independent test team and is therefore an excellent candidate for test outsourcing. Finally, Acceptance testing requires business know how (for user acceptance) and a production-like test environment. It is therefore difficult to outsource.
- **Test Types:** Generally speaking, all types of testing, both functional and non-functional, can be outsourced. By its nature, regression test is a good candidate for cost saving, because it involves regular repetition and test automation. Know-how intense test types like load & performance, usability and security are best outsourced to a specialist organisation, on a case-by-case basis.
- **Test Activities:** Defining what activities within the test process should be outsourced (e.g., test planning, specification, execution and reporting) requires a strategic decision on how much control and knowledge is given away. It can range from test execution only, to performance of the whole process.

Not only do you have to decide on what parts of testing to outsource, but also which projects are most suited for it. If it is the first release of a totally new application, it may not be ideal to outsource its testing. However, if the project is an

enhancement version of a stable product, with existing test cases, it would be a very good candidate. Possible selection criteria include: technical complexity in-house know-how maturity of requirements and test cases availability of test data .

A typical example of such an analysis is shown in graph (a) .

Highly suitable	Sometimes suitable	Not suitable
<b>Test Level</b>		
System Testing	Unit and Integration Testing (to developers)	Acceptance Testing
<b>Test Types</b>		
Regression, Functionality Localisation, Compatibility	Load and Performance, Usability, Security	
<b>Test Activities</b>		
Test Execution, Test Reporting, Test Automation	Test Planning, Test Analysis & Design (requires good business knowhow)	Test Management

Graph (a)

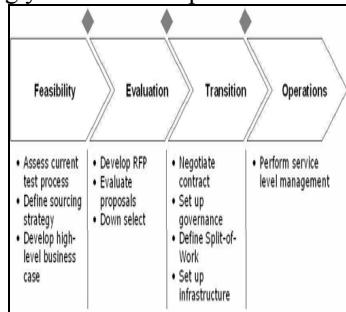
### 2.3 How to outsource?

Outsourcing is never a simple task and many such efforts will fail because they haven't been managed thoroughly. Unrealistic expectations, the selection of the wrong partner and communication problems are common reasons for failure. It is therefore imperative to use a well-thought out and planned procedure, that makes use of best practices and actively addresses the risks involved.

In the first phase, graph (b), you have to look at the feasibility of your outsourcing effort. Beside the decision on what to outsource (as discussed previously), this includes the assessment of the maturity of your test process. Outsourcing is seldom a solution for internal process problems. In the contrary, the problem is outsourced and stays unresolved or even becomes accentuated. Identify and address these deficiencies first. Outsourcing may still be possible but might involve additional costs [5].

This leads us to the calculation of a rough business case. Consider different outsourcing options, from bringing in external testing experts to setting up an offshore development center, to find the most cost-effective alternative.

If you decide that it is all worthwhile, you can start the search for a suitable outsourcing partner by developing a request for proposal and inviting offers. It is important to look not only for testing know how, but also for product and technology knowledge. This will result in a steeper learning curve, allowing you to become productive sooner.



Graph (b) showing the different phases of the procedure of How to outsource.

The selection of the provider will then kick off the transition phase. It is all about paper work and getting to know each other. While the lawyers do the contractual work, start defining the governance and split of work, as well as setting up the infrastructure. Start with a small number of projects, to gain experience. All going well, the transition phase will lead you to steady state operation, typically within 6 to 9 months.

A key ingredient for success in day-to-day operations is an effective service level management. Identify the key performance indicators, define the service levels to be achieved and describe how these will be measured (what, who, when/how often) in a service level agreement. Besides the usual metrics such as on-time delivery, estimation accuracy and personnel retention, test specific metrics include: relative test effort,test coverage ,percentage of test automation, remaining defects

### 2.4 Important aspects for successful outsourcing:

I would recommend that you consider the following points when deciding whether to outsource a project or application for testing.

#### ➤ The nature of the project

Firstly, consider the nature of the project or application that you are considering outsourcing.If the project is an enhancement version of a stable product, with existing test scripts that could be used by the outsourcing partner, it would be ideal to outsource.However, if it is the first release of a totally new application, it may not be ideal to outsource the testing. For example, if there are major problems with the application, testing is not going to solve them, and if your testing partner encounters a large number of serious errors the test coverage will probably be lessened.

Lastly, carefully consider what you expect to achieve by outsourcing. In all likelihood, outsourcing will never (and probably should never) completely do away with the necessity to have in-house expertise, so you should consider the loss of knowledge and hands-on experience that your test team will lose.

#### ➤ What testing do you want them to do?

It is also important to consider what types of testing that you actually require to be performed. For example, if stress testing is important, an outsourced testing firm may be more experienced and suited towards performing this. Or, you may wish to perform a certain aspect of the testing that is very complex and that you believe you would not sufficiently benefit from by outsourcing.

#### ➤ The Practicabilities

Who is going to be fixing the bugs? If your company is going to, the process and procedures by which the error reports are going to be exchanged is critical. If your error management system cannot be remotely updated [i.e. by the outsourced partner reporting the bugs directly] then I would suggest either not outsourcing at all, or else implementing an error management tool which can cater for remote logging, and can be under your direct control. It is important that you retain control (or at least possession) of the bug database for future reference.

Secondly, how are you going to hand over new builds of the s/w? If you have to burn CD's, unless there is a quick way of transferring the new builds over to the outsourced testing partner, valuable testing time may be lost.

### III. FLAWS IN OUTSOURCING

Not many outsourcing collaborators realize that the seeds for outsourcing failure is often sown at the time of due diligence. Often spending very little time on due diligence in the contract negotiation process in a bid to 'hurry up' the negotiations leads to this monumental pitfall. Five areas where outsourcers should apply more thought before taking the plunge:

- **Unrealized cost savings:**

Most businesses push work overseas in the hope of cutting labor costs. An application maintenance worker in India, for example, earns about \$25 per hour, compared with \$87 per hour in the U.S., according to Gartner. But businesses make a mistake by looking at salaries alone. Hidden expenses for things like infrastructure, communications, travel and cultural training take a bite out of the wage differential. What's more, planning and start-up costs are high, so offshore deals lasting less than a year may not pay off at all, and savings from longer-term deals will emerge slowly.

- **Loss of productivity:**

Staff at an offshore service center probably won't be as productive as internal staff back at home, at least not initially. Gartner offers several reasons: Staff turnover can be high in competitive offshoring markets such as Bangalore, India, which also means programmers there may be new and inexperienced. And service centers overseas struggle with ambiguities in the work they are assigned and shifting directives. Sending jobs overseas can also lower morale at home, creating a drag on output.

- **Poor commitment and communications:**

Senior executives often drift out of the picture once a deal is signed. But they need to stay engaged to keep morale high and strategy on track. And good communications among all parties is paramount. Projects, goals and expectations have to be defined clearly and in detail. On the home front, managers need to explain clearly why work has been sent overseas and what benefits are expected.

- **Cultural differences:**

Communication styles and attitudes toward authority vary from region to region and those differences can cause problems. In some cultures, questioning authority is considered disrespectful, so a team may push ahead with a given plan even if they see a better approach. Offshore should get expert advice about a local culture, provide cultural training and even arrange exchanges among staff on both sides.

- **Lack of offshore expertise and readiness:**

Some organizations make the leap before they are ready. Offshore need to get everything in place internally and secure the support of key stakeholders in the company before launching a project. They should also figure out the risks and how to mitigate them.

Here are the reasons why due diligence is critical. Outsourcing is akin to acquisition. IT Outsourcing (ITO), for example, essentially involves the acquisition of the client's IT department (or portion of it) by the vendor. The vendor, selected by the client as the result of a competition, acquires in some cases the physical assets, such as computers, peripherals, networks, etc., assumes the responsibility for third party licenses and procurement functions, and usually hires the employees from the client. This is exactly what happens during an acquisition. The only difference is that the client enters into a time bound contract with the vendor to provide back the services previously delivered by the client's IT department. As with an acquisition, outsourcing requires sound due diligence and the process used should not be any different than those used in an acquisition of company.

Due diligence is done to confirm the baseline initially set by the client during the RFP phase and subsequent negotiations. Some key aspects include:

1. *Financial verification:* Baseline costs for each element of service to be provided by the vendor. Usually these costs refer to the initial or baseline year of the contract.
2. *Procurement verification:* Third party software licenses and associated costs. This includes both initial license fees and subsequent maintenance costs.
3. *Infrastructure verification:* Asset inventory and associated costs, including computers, peripherals and networks.
4. *Human Resources:* Employee costs, including benefit plans, salary scales, union involvement, and organizational structures.
5. *Processes and systems:* Processes and methodologies used by the client organization.

There are other aspects that are equally important such as culture, governance models and so on. It is important to note that both the vendor (seller) and the client (buyer) in any outsourcing relationship need to do due diligence on the other. The process, if performed effectively, sets the stage for the long term relationship between the client and vendor and is vitally important to the overall success of the outsourcing contract. Clients are to approach the due diligence process with an open mind and disclose all relevant information to enable the vendor to accurately assess the business. At the same time vendors need to present themselves as credible partners often establishing the ability to match up the capabilities of IT delivery and improve it. Due diligence is the first step in cementing an open and mutually trustworthy relationship that will benefit both parties. Don't underestimate the time and effort necessary to set-up and transition. It will probably take you longer than you expected. Adapt your business case accordingly and ensure you have strong management commitment. It will also take some time before you can fully harvest the benefits of your outsourcing effort, so keep looking for improvement potential and conduct periodic reviews with the provider's management team to review progress as defined in the service level agreement. The setup of the test infrastructure is another cost and success factor for outsourced testing. Setting up a full-blown test environment at the provider's site is an ideal solution, but it is often difficult to replicate your environment, especially with regard to interfaces

to peripheral systems and third-party products. In any case, a reliable and secure communication link is needed. This raises data privacy issues about giving access to your test environment and sharing test data. Only grant access on a need to know basis and use anonymous, or even better, synthetic test data, if you want to avoid surprises. Remember though, that this is not primarily about technology, it is all about people. Communication problems, aggravated by geographical distance, cultural differences and different languages, are a major reason of failure. Engaging an onsite coordinator will facilitate communication with the offshore team. Well-defined processes for the creation and handover of test deliverables as well as defect-reporting will help minimize the potential for friction in daily work. To help avoid misunderstandings it is important to define quality gates at the interfaces between you, the customer, and the provider - in both directions. Quality gates define what has to be delivered and in what quality. They usually involve one or more reviews. A detailed and understandable requirements specification is the key to each test (outsourcing) project. Make sure that the test team has understood what it is testing against. Later on, review of the test plan and test cases will help to establish if the testing effort is on track. As the extent of the outsourcing grows, however, so too will the challenge of managing relationships with them. But if companies fail to do this effectively, projects could be jeopardised. The danger is that weak outsourcing partners can stall entire projects. So, how do you provide for this situation? Apart from the contractual elements to formalise the relationship, you must naturally consider how you can actually verify their progress. Additionally, you need also to be sure that the work is being performed to a high level of quality.

### *3.1. Pay attention to dependencies and quality of current test artifacts.*

A successful outsourced automation requires that project dependencies are well understood by all stakeholders. Access of application from offshore, access to testers/developers/SMEs on test cases, Access to reviewers and code acceptance people are few dependencies that one needs to track as part of project [6]. If automation is on the basis of existing manual test cases- make sure that these are detailed enough and available in a form that can be sent across to vendor offshore team.

### **3.2. OFFSHORE OUTSOURCE IMPROVEMENT**

Many organizations have offshore their software testing with varying degrees of success. Some have worked well. Many others have struggled and it is not uncommon to find organizations that have onshored their offshore resources or are simply having difficulty realizing the value from these outsource deals that the business case outlined. We have provided our offshore testing improvement service to a number of organization across different sectors to help them improve their results and migrate back offshore their software testing activities.

### IV. CONCLUSION

What is the quintessence? Outsourced testing can speed up testing, improve quality and reduce costs. But, no big surprise, it won't fall into your lap – it requires a well-managed effort before, during and after the transition. Firstly, define the channels of communication. Appoint a person from your company, to act as the first level of contact between your company and the outsourced partner, who will deal directly with an agreed specified person on their side (ideally the test manager for the project). These two people are the first-level team for resolving problems as they arise, and additionally your representative also provides and archives all information passed to the outsourcing partner. Secondly, agree escalation procedures, and specify to whom the problem is to be escalated to - on each side. Set out milestones – but beware when setting milestones, that YOU are the ones who state whether the outsourced testing company has reached the milestone or not – for example, if one milestone is the production of the test plan, if you are not happy with the test plan then you must be in a position to state that they have NOT reached the milestone. Otherwise you could end up in a position whereby they gave you a test plan that you considered inadequate, yet they are looking for payment for reaching the milestone. To cater for this, and other concerns, it is vitally important to specify formal review points – stages at which progress can be measured, and where issues can be highlighted.

### V. REFERENCES

- [1.] "Outsource Testing For Quality Gains, Not Just Cost Savings" by Uttam Narsu on July 12, 2004 in Forrester magazine.
- [2.] ISTQB Standard Glossary of Terms used in Software Testing <http://www.istqb.org/>
- [3.] "Outsourced Testing–Friend or Foe?" Published: Oct 08 2008 By martinig via methodsandtools.com.
- [4.] "The effects of global outsourcing strategies on participants attitudes and organizational effectiveness" In International Journal of Manpower Year: 2000 , Volume: 21 by Dean Elmuthi and Yunus Kathawala
- [5.] "Making the decision to outsource human resources" In Journal - Personnel Review in Year: 2009, Volume: 38 by Jean Woodall, William Scott-Jackson.
- [6.] Chapter 16" Moving People or Jobs? A New Perspective on Immigration and International Outsourcing" in Book Series: Frontiers of Economics and Globalization Year: 2008, Volume: 4 Page: 317 – 327. Chapter URL: [http://www.emeraldinsight.com/10.1016/S1574-8715\(08\)04016-5](http://www.emeraldinsight.com/10.1016/S1574-8715(08)04016-5).
- [7.] "The Benefits of Outsourced Testing", <http://members.tripod.com/bazman/index.html>.
- [8.] "Primer on Outsourced Software Testing "By: Thinksoft , published on sep 02,2005.

# Comparison of Wavelet CDF 9/7 and Lifting Wavelet Transform in Image Watermarking

Er.Navjeet Sidhu- Student, M.Tech, Er.Kamaldeep Kaur- Lecturer  
BBSBEC, Fatehgarh Sahib, Punjab, Email-ID: - navjeet\_sdh@yahoo.co.in, kamal\_deep2k3@yahoo.com

**Abstract -** Nowadays, digital documents can be distributed via the World Wide Web to a large number of people in a cost-efficient way. The increasing importance of digital media brings new challenges, as it is now straightforward to duplicate and even manipulate multimedia content. There is a strong need for security services in order to keep the distribution of digital multimedia work both profitable for the document owner and reliable for the customer. Watermarking technology plays an important role in securing the business as it allows placing an imperceptible mark in the multimedia data to identify the legitimate owner; track authorized users via fingerprinting or detects malicious tampering of the document.

This paper studies the use of image transforms like wavelet transforms in watermarking. The central idea is to develop algorithms using MATLAB for using wavelet cdf 9/7 and lifting wavelet transforms in watermarking. Our focus is on invisible watermarks.

DWT (discrete wavelet transforms) considered for this work are wavelet cdf 9/7 and lifting wavelet transform. The results of the above mentioned transforms are compared. Analysis of results is also done.

**Keywords:** Watermarking, PSNR, CDF 9/7,LIFTING WAVELET TRANSFORM

## I. INTRODUCTION

Digital watermarking is the process of embedding information into a digital signal. In visible watermarking, the information is visible in the picture or video. Typically the information is a text or logo, which identifies the owner of the media. In invisible watermarking, information is added as digital data to audio, picture or video, but it cannot be perceived as such. In digital watermarking a low energy signal is embedded in another signal. The low energy signal is called watermark and it depicts some metadata, like security or rights information about the main signal. Digital watermarking includes a number of techniques that are used to imperceptibly convey information by embedding it into the cover data. [1]. An important application of invisible watermarking is to copyright protection systems, which are intended to prevent or deter unauthorized copying of digital media. According to the Cox *et.al's* work [2], the embedded watermark should be robust to common signal processing, common geometric distortions and forgery. In robust watermarking applications, the extracted algorithm should be able to correctly produce the watermark, even if the modifications were strong.

## II. TRANSFORMS USED

### 1. WAVELET CDF 9/7

Lifting scheme [3] represents wavelets transform as a sequence of predict and update steps. Let  $X=[X(1), X(2), \dots, X(2N)]$  be an array of length  $2N$ . The lifting scheme begins

with the "polyphase decomposition," splitting  $X$  into two sub bands, each of length  $N$ :

$$X_o = [X(1), X(3), X(5), \dots, X(2N-1)], \quad X_e = [X(2), X(4), X(6), \dots, X(2N)].$$

Since  $X_o$  and  $X_e$  can be merged to recover  $X$ , no information has been lost.

Next, the scheme performs lifting steps on the sub bands

$$X_o \text{ and } X_e. \text{ Let } p \text{ be a filter, then}$$

$$X'_e = X_e + p^* X_o$$

is called a "prediction" step, where  $*$  denotes convolution. Similarly,  $X'_o = X_o + u^* X_e$  is called an "update" step. Notice that  $X_e$  can always be recovered from  $X'_e$  with

$$X_e' = X_e - p^* X_o.$$

This simple relationship between a forward step and an inverse step is the key to the lifting scheme: any sequence of prediction and update steps can be "undone" to recover  $X_o$  and  $X_e$ .

CDF 9/7 is an especially effective biorthogonal wavelet, used by the FBI for fingerprinting compression and selected for the JPEG 2000.

Any FIR wavelet transform can be factored into a sequence of lifting steps [4]. For the CDF 9/7 wavelet, the lifting scheme decomposition used in cdf 9/7 is

$$X_o = [X(1), X(3), X(5), \dots, X(2N-1)]$$

$$X_e = [X(2), X(4), X(6), \dots, X(2N)]$$

$$X_e^1(n) = X_e(n) + \square(X_o(n+1) + X_o(n))$$

$$X_o^1(n) = X_o(n) + \square(X_e^1(n) + X_e^1(n-1))$$

$$X_e^2(n) = X_e^1(n) + \square(X_o^1(n+1) + X_o^1(n))$$

$$X_o^2(n) = X_o^1(n) + \square(X_e^2(n) + X_e^2(n-1))$$

The sub bands are then normalized with  $X_o^3 = \square X_o^2$  and  $X_e^3 = \square^{-1} X_e^2$ . For a multi-level decomposition, the algorithm above is repeated with  $X = X_o^3$ .

The numbers  $\square, \square, \square, \square, \square$  are irrational values

approximately equal to

$$\square \approx -1.58613432,$$

$$\square \approx -0.05298011854,$$

$$\square \approx 0.8829110762,$$

$$\square \approx 0.4435068522,$$

$$\square \approx 1.149604398.$$

The inverse CDF 9/7 transform is done by performing the lifting steps in the reverse order and with  $\square, \square, \square, \square$  negated. What if  $X$  has odd length  $2N-1$ ? The trick is to extrapolate one extra element  $X(2N)=x$  in such a way that transforming the augmented  $X$  has  $X_e^3(N)=0$ . This zero element can then be thrown away without losing information. The result is decomposition with  $N$  elements in  $X_o^3$  and  $N-1$  elements in  $X_e^3$  for a total of  $2N-1$  elements; the decomposition is nonredundant. To invert an odd-length transform, append the zero element  $X_e^3(N)=0$  and proceed with the usual even-length inverse transform.

## 2. Lifting Wavelet Transform

This is a multi-level discrete two-dimension wavelet transform based on lifting method. Currently, wavelift only support two kind of wavelets i.e. cdf 9/7 and spline 5/3 (also called LeGall 5/3).

Lifting scheme (LS) is an alternative approach to the filter bank structure for computing DWT scheme. Lifting is more flexible and may be applied to more general problems is commonly used in image processing to refer to each set of samples which are the output of the same 2-D filter. In 1-D linear processing both concepts are interchangeable.

The lifting scheme formally introduced in by W. Sweldens is a well-known method to create biorthogonal wavelet filters from other ones. The scheme comprises the following parts:

- (a) Input data  $x_0$ .
- (b) Polyphase decomposition (or lazy wavelet transform, LWT) of  $x_0$  into two subsignals. The term polyphase comes from digital filter theory where it is used to describe the splitting of a sequence of samples into several subsequences, which can be processed in parallel. The subsequences can be seen as phase-shifted versions of each other, hence the name [5].
  - An approximation signal  $x$  formed by the even samples of  $x_0$ .
  - A detail signal  $y$  formed by the odd samples of  $x_0$ .
- (c) Lifting steps:
  - Prediction P (or dual) lifting step that predicts the detail signal samples using the approximation samples  $x$ ,  
 $y'[n] = y[n] - P(x[n]).$
  - Update U (or primal) lifting step that updates the approximation signal with the detail samples  $y_0$ ,  
 $x'[n] = x[n] + U(y'[n])$

The update phase follows the predict phase. [6] The original value of the odd elements has been overwritten by the difference between the odd element and its even "predictor". So in calculating an average the update phase must operate on the differences that are stored in the odd elements.

(d) Output data: the transform coefficients  $x'$  and  $y'$ .

Possibly, there are scaling factors  $K_1$  and  $K_2$  at the end of each in order to normalize the transform coefficients  $x'$  and  $y'$ , respectively.

The prediction and update operators may be a linear combination of  $x$  and  $y$ , respectively, or any nonlinear operation, since by construction the LS is always reversible. The lifting scheme is constantly under development and is investigated by many. Recent additions are the lifting scheme in a redundant setting in order to improve the translation invariance [5] and adaptive prediction schemes for integer lifting.

### III. WORK DONE

There are many new recently added filters, which operate in the wavelet domain. Many of them have not been applied in the watermarking schemes. Effort is being done to implement some of these in the watermarking technique. Comparison is being done between two wavelet transforms i.e. wavelet cdf 9/7 and lifting wavelet transform.

Three different images were taken for the analysis. The images are named as B1, B2 and B3. Image quality of the watermarked

image was observed by finding out the PSNR, WPSNR. At the detection stage, the good or bad recovery of the message from the watermarked image was decided by the quality of the message (through visualization). On the basis of performance parameters (PSNR and WPSNR) the two discrete wavelet transforms were compared.

First of all, considering only one image, for different levels of this image, PSNR and WPSNR were observed. This was done for the cdf 9/7 transform. In the same way, the above-mentioned parameters were found for the lifting transform. For both of these transforms values of the parameters (PSNR and WPSNR) were observed at different levels and the best-suited decomposition level was decided. Then at that particular level, the work was continued further.

CDF 9/7 and lifting wavelet transforms are compared with each other. This was done for all the three images considered in this work. The values are compiled in a tabular form to reach a conclusion. The results of only one image are shown in this paper to avoid unnecessary lengthy essay.

#### PSNR

*The phrase peak signal-to-noise ratio, often abbreviated PSNR, is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation.*

The only rigorously defined metric is PSNR. The main reason for this is that no good rigorously defined metrics have been proposed that take the effect of the Human Visual System (HVS) into account. PSNR is provided only to give us a rough approximation of the quality of the watermark.

PSNR does not take aspects of the HVS into effect so images with higher PSNR's may not necessarily look better than those with a low PSNR. This will prove particularly true in the case of the DCT and DWT domain techniques.

#### Weighted PSNR (WPSNR)

A simple approach to adapt the classical PSNR for watermarking application consist in the introduction of different weights for the perceptually different regions oppositely to the PSNR where all regions are treated with the same weight. WPSNR uses an additional parameter called the Noise Visibility Function (NVF) that is a texture masking function as a penalization factor. NVF uses a Gaussian model to estimate how much texture exists in any area of an image.

### IV. RESULTS

In this paper, wavelet cdf 9/7 and lifting wavelet transform have been analyzed and compared. PSNR and WPSNR were observed for different images and their results were compared. In this work, the hidden data to be embedded in the image was taken to be the 'signatures'. This hidden data is the watermark. The image where the watermark is to be embedded is called the host image. Matlab have been used for finding out the results. Three images B1, B2 and B3 were taken for the analysis purpose. The message to be embedded is shown in figure 2. The message is in the form of a specimen signature.



Fig 1: Original image B1

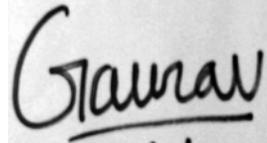


Fig 2: Embedded Message

Firstly, considering only image B1, for different decomposition levels of this image, PSNR and WPSNR were calculated for the two discrete wavelet transforms considered for this work.. It was found that after decomposition level 2, PSNR became constant. As it can be seen from the table, level 2 was preferred since it showed a higher PSNR and WPSNR as compared to level 1. This is shown in table 1.

TABLE 1: CDF 9/7 TRANSFORM AT DIFFERENT LEVELS

LEVEL	PSNR (in db)	WPSNR (in db)
1	51.1075	-11.4590
2	67.6034	0.6142
3	67.6034	0.6142

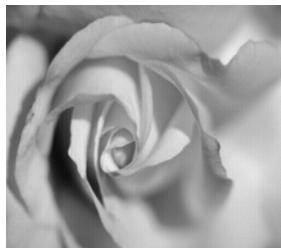


Fig 3: Watermarked image for CDF 9/7 at level 2

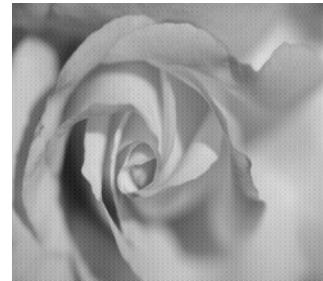
The parameter values for the above image in fig 3 are shown in the table below.

TABLE 2: PARAMETER VALUES FOR WAVELET CDF 9/7 TRANSFORM

	B1 (512*512)	B2 (512*512)	B3 (512*512)
PSNR (in db)	67.6034	65.7312	65.6861
WPSNR (in db)	0.6142	-0.2209	-0.2734

Table 3 : Lifting Wavelet Transform At Different Levels

LEVEL	PSNR (in db)	WPSNR (in db)
1	50.9767	-11.5287
2	48.0675	-10.4103
3	48.0675	-10.4103



a) Fig 4: Watermarked image for lifting wavelet transform at level 2

Table 4: Values For Lifting Wavelet Transform

	B1 (512*512)	B2 (512*512)	B3 (512*512)
PSNR (In db)	48.0675	47.5232	47.4930
WPSNR (In db)	-10.4103	-10.6718	-10.6890

The watermarked image for lifting wavelet transform is shown above and values are compiled in table 4. Comparing cdf 9/7 and lifting wavelet transform (at level 2) very good PSNR was observed for cdf 9/7.

For image B1, cdf 9/7 showed a PSNR value of 67.6034 db whereas lifting wavelet transform showed 48.0675 db. Similarly for WPSNR, cdf 9/7 showed a value of 0.6142 db, whereas lifting wavelet transform showed a value of -10.4103 db. Thus cdf 9/7 proved to be a much better choice than lifting wavelet transform.



Fig 5: Recovered message

The recovered message showed a PSNR value of 4.1022 db and WPSNR value of -39.8112db.

## V. CONCLUSION

It has been observed from this work that if there is a requirement of quality of watermarked image, then wavelet CDF 9/7 is a better choice than lifting wavelet transform.

## VI. REFERENCES

- [1] Ingemar J. Cox, Joe Kilian, Tom Leighton, and Talal G. Shamoon. Secure spread spectrum watermarking for multimedia. In *Proceedings of the IEEE International Conference on Image Processing, ICIP'97*, volume 6, pages 1673-1687, Santa Barbara, California,USA, October 1997.
- [2] I.Cox, J.Kilian, F.Leighton and T. Shamoon, “ Secure Spread Spectrum watermarking for Multimedia”, *IEEE Transactions on Image Processing*, vol .6, no.12, pp. 1673-1678, Dec. 1997.
- [3] Algazi, V. R. and DeWitte, J. T., “Theoretical Performances of Entropy Coded DPCM”, *IEEE Trans. Comm.* vol. 30, no. 5, pp. 1088-1095, May 1982.
- [4] Anil K. Jain, “Fundamental of Digital Image Processing”, *Prentice-Hall Inc*, 1989.
- [5] C.Valens The Math forum@Drexel Fast lifting wavelet transform 1999-2004 (USA): 1995.
- [6] Robert D.Kaplan Wavelet lifting scheme Vintage press,1996.

# Design and Implementation of Cheque Receiver System

Manish Sharma, Raj Kumar and Parmender Singh  
 Gurgaon Institute of Technology and Management, Gurgaon, India.  
[raj\\_indya2000@yahoo.com](mailto:raj_indya2000@yahoo.com), [manishsharma1978@yahoo.com](mailto:manishsharma1978@yahoo.com)

**Abstract-** The Cheque Receiver system is aiming at the future of Banking System. Data of a Cheque is obtained after scanning, is further used for processing to verify signature and amount validation. This paper discusses an advanced and new banking system information using Microcontroller with supporting Modules. This paper is in further reference to Design and Implementation of cheque receiver System where HITACHI's H8S/2148 Microcontroller is used. Microcontroller produces the required operating and control signals required to operate the driver circuit for stepper motor and sensor module. These generated signals enables stepper motor and cheque inserted is accepted inside the system. Sensor starts scanning the cheque leaf and data is converted from analog to digital form with the help of A/D converter within the microcontroller. The converted raw data is sent to HyperTerminal and this raw data is converted to .jpg Image. Microcontroller is so programmed that it generates the required driving and control signals and frequency required to operate driver circuit and timer module.

**Key Words:** Cheque, Microcontroller, Scanner, stepper motor, A/D Converter.

## I. INTRODUCTION

Cheque Receiving System is the advancement of simple cheque receiving system where in the earlier system only Magnetic Ink Character Recognition was possible but including this the present system it is possible to scan the cheque and the scanned image is further processed recognizing the important features such as signature, date and amount validation. Once the date and signature, date and amount has been validated through RBI , RBI can then send acknowledgement and the cheque can be cleared off within no time. With the proper design modules such as GLCD interface for display,stepper motor for movement of cheque on scanner and scanner combination along with the Microcontroller for data processing and generating different control signals for the system to be implemented. Higher end microcontroller can be used for design purpose so that the internal memory of the microcontroller can be used .High frequency operating microcontroller can be used so that the working can be made much faster as it is one of the requirement nowadays.

## II. DESCRIPTION OF SYSTEM

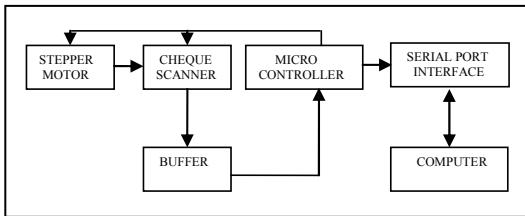


Fig.1 Block Diagram of Cheque Receiving System

In the above system Microcontroller H8S/2148 is the heart of the system. The objective is that to get the image of the cheque and then this image (BMP) can be further used for various image processing steps so that the parameters such as date and amount can be checked for validation. Cheque is inserted in the slot where scanner module is mounted on the Stepper Motor setup. As the motor rotates the cheque passes through the scanner and the analog data is obtained. Once the analog data is obtained, it is given to the buffer 74LVC244A IC which is a high-performance, low-power, low voltage, Si gate CMOS device, superior to most advanced CMOS compatible TTL families. Inputs can be driven from either 3.3V or 5V devices. In 3-State operation, outputs can handle 5V. These features allow the use of these devices as translators in a mixed 3.3V/5V environment. O/P from the buffer is connected to the microcontroller so that the analog data is converted to digital form and then data is sent to PC through serial port interface channel. Data is stored i.e. displayed on HyperTerminal and data is stored. The stored is then converted to BMP format and it is used for various image processing steps so that date & amount can be checked for validation.

## III. DESCRIPTION OF DESIGN

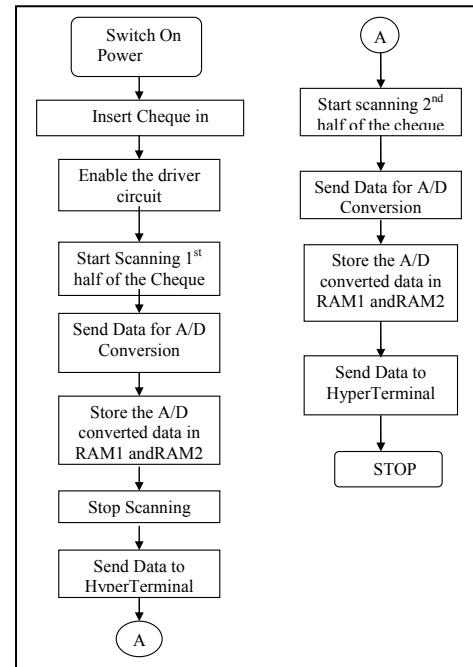


Fig.2 Flow Chart of Cheque Receiving System

Flow chart of the complete system operation is shown above. When the system is switched on microcontroller

produces the operating and control signals required to operate the driver circuit and sensor module. This signal enables stepper motor and inserted cheque is accepted inside the system. Sensor starts scanning the cheque leaf and data is sent to the a/d converter module. 10 bit resolution a/d converted output is sent to external RAM. Once the half scanning is over data is read from the RAM to the hyper terminal. When transfer is over controller sends the control signal to the stepper motor and scanning of next half of the cheque is started. Microcontroller is programmed to generate the required driving and control signals and frequency required to operate driver circuit and timer module is generated by programming the microcontroller. Serial input pulse signal required to generate the output data from the scanner is accurately calculated and microcontroller is programmed to generate the accurate serial signal.

#### IV. DESIGN PARAMETER CALCULATION

##### A. Stepper Motor Driver

The cheque which is to be scanned is inserted in the slot for scanning. The main design steps can be listed as below:

###### a. Half step rotation of motor

Scanning speed of the cheque by the scanner is 0.4ms/line with effective scanning width 104 mm. Therefore perfect rotation of motor and scanning speed has to be synchronized. Therefore rotation of motor per step is very important otherwise data will be lost. Half step rotation of the motor is chosen.

###### b. Enabling Signal for the Driver (clock signal)

Board operating frequency = 20MHz

Selected frequency  $F_s = 20\text{MHz} / 1024$  (selected by setting CKS=3 & ICKS=0)

$$= 19.5 \text{ KHz}$$

So time  $T_s = 1 / 19.5 \text{ KHz}$   
 $= 0.051 \text{ msec}$

$T_s = 51.2 \text{ micro sec}$

Register count set for smooth stepper motor running is

Register A= 50

Register B= 25

Having the relation, full count = clock in time / operating time

Full count = 50

Operating time = 51.2 micro sec

Clock in time = full count \* operating time  
 $= 50 * 51.2 \text{ micro sec}$   
 $= 2.56 \text{ ms}$

So clock in frequency =  $1 / 2.56 \text{ msec}$   
 $= 0.390625 \text{ KHz}$   
 $= 390 \text{ Hz}$

So clock in frequency for the smooth running of the stepper motor is 390Hz

###### c. Operating frequency for sensor head

Board operating frequency = 20 MHz

Clock in frequency selected =  $20 \text{ MHz} / 4$

$F_s = 5 \text{ MHz}$

So time =  $1 / 5 \text{ MHz}$   
 $= 0.2 \text{ micro sec}$

Operating frequency for the sensor ranges from 0.2 MHz to 1 MHz

Selected operating frequency  $F_o = 0.5 \text{ MHz}$

So time  $T_o = 1 / 0.5 \text{ MHz}$

$T_o = 2 \text{ micro sec}$

1 count is equivalent to 0.2 micro sec

So full count = 2 micro sec

$$= 10 \text{ counts}$$

Register A = 10

Register B = 5

0.5 MHz frequency is selected for scanner head.

##### B. Serial In Clock

Selection of Serial In clock for scanner is very important for proper data output.

Time taken to scan one line = 0.4 msec

Overall scanning speed = 16 sec

Board operating frequency = 20 MHz

Clock in frequency =  $20 \text{ MHz} / 1024$

$$F_s = 19.5 \text{ KHz}$$

So time =  $1 / 19.5 \text{ KHz}$

$$= 0.051 \text{ msec}$$

$$T_s = 51.2 \text{ micro sec}$$

Total lines scanned in the cheque N = 1632 lines

Time required to scan one line =  $16 / 1632$

$$= 0.0098 \text{ sec}$$

$$= 9.8 \text{ msec}$$

Serial in time to scan one line is 9.8 msec

So 1 count is equivalent to 51.2 micro sec

Full count = 9.8 msec / 51.2 micro sec

Full count = 192

So register settings are given as register A = 192

Register B = 96

Clock in frequency =  $1 / \text{clock in time}$

$$= 1 / 9.8 \text{ msec}$$

$$= 102 \text{ Hz}$$

Clock in frequency = 102 Hz

##### C. Cheque Data

The total data obtained from the scanning of the cheque is calculated as below:

Sensor resolution used to scan the cheque = 8 pixels / mm

Overall breadth of the cheque = 92 mm

Overall length of the cheque = 92 mm

Analog output will be got pixel by pixel at the output pin of the sensor head

So number of pixels in one line =  $92 * 8 \text{ pixels}$   
 $= 736 \text{ pixels / line}$

Overall data output from one line =  $736 * 2 \text{ bytes}$   
 $= 1472 \text{ bytes}$

So data output from one line = 1.472 Kbytes

Length of the cheque = 204mm

Line to line separation = 0.125mm

So number of lines =  $204 \text{ mm} / 0.125 \text{ mm}$   
 $= 1632 \text{ lines}$

Overall lines scanned = 1632 lines

Hence overall data = 1 line data \* total lines  
 = 1.472Kbytes \* 1632  
 = 2402.304 Kbytes

Data obtained from the output of the sensor = 2.4 MB

#### D. A/D Conversion

The A/D converter operates by successive approximations with 10-bit resolution. It has two operating modes: single mode and scan mode.

Single Mode (SCAN = 0)

Single mode is selected when A/D conversion is to be performed on a single channel only. A/D conversion is started when the ADST bit is set to 1 by software, or by external trigger input. The ADST bit remains set to 1 during A/D conversion, and is automatically cleared to 0 when conversion ends.

On completion of conversion, the ADF flag is set to 1. If the ADIE bit is set to 1 at this time, an

ADI interrupt request is generated. The ADF flag is cleared by writing 0 after reading ADCSR.

When the operating mode or analog input channel must be changed during analog conversion, to prevent incorrect operation, first clear the ADST bit to 0 in ADCSR to halt A/D conversion. After making the necessary changes, set the ADST bit to 1 to start A/D conversion again. The ADST bit can be set at the same time as the operating mode or input channel is changed.

Typical operations when channel 1 (AN1) is selected in single mode are described next.

1. Single mode is selected (SCAN = 0), input channel AN1 is selected (CH1 = 0, CH0 = 1), the A/D interrupt is enabled (ADIE = 1), and A/D conversion is started (ADST = 1).
2. When A/D conversion is completed; the result is transferred to ADDRDB. At the same time the ADF flag is set to 1, the ADST bit is cleared to 0, and the A/D converter becomes idle.
3. Since ADF = 1 and ADIE = 1, an ADI interrupt is requested.
4. The A/D interrupt handling routine starts.
5. The routine reads ADCSR, and then writes 0 to the ADF flag.
6. The routine reads and processes the conversion result (ADDRB).
7. Execution of the A/D interrupts handling routine ends. After that, if the ADST bit is set to 1, A/D conversion starts again and steps 2 to 7 are repeated.

#### V. DATA TO HYPER TERMINAL

Once the A/D conversion is over the data is stored in RAM from where it is sent to the HyperTerminal. The stored data is converted to BMP format which may be used for further image processing.

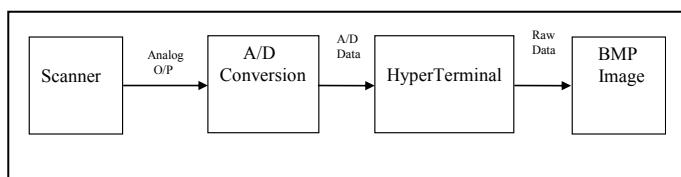


Fig. 3 Conversion of Raw data to BMP Image

#### VI. RESULTS AND CONCLUSION

The Microcontroller ‘Renesas’ used in this Cheque Receiver System is found to be far superior when compared to the other Microcontrollers available in the market. The Microcontroller has huge on – chip program memory, having 10,000 write cycles. The in-circuit system-programming (ISP) feature of this controller is very useful, since the controller is not disturbed in the circuit during up gradation of software. It supports on – chip PS/2 keyboard controller, which enables direct connection of a PS/2 keyboard. The Microcontroller supports MAC instructions, which increases the computational speed. The Microcontroller supervisory circuit is designed such that, the back-up power is activated as soon as the external power fails. Thus memory corruption is avoided. The software is developed keeping in mind all the requirements of the end users, thus making this Cheque Receiver System very powerful.

#### a. Few Snapshots



The above two figures shows the snapshot display of sending the data and scanning the data.

#### VII. ACKNOWLEDGEMENTS

The authors are very thankful to Dr. NC Parasana Kumar, Director, Gurgaon Institute of Technology and Management, Gurgaon for his constant encouragement, guidance and support.

#### VIII. REFERENCES

- [1]. "L297/L297D Stepper Motor Controllers." Data Sheet. SGS Thompson Micro electronics, August 1996.
- [2]. "L298 Dual Full-Bridge Drive." Data Sheet. SGS Thompson Microelectronics, July 1999.
- [3]. IA2004-ME32A, "Image Sensor Heads for Narrow-Width Scanners Reference Manual", ROHM Limited.
- [4]. H8S/2148 Series, "H8S/2144 Series Hardware Manual", Renesas Co. Ltd.
- [5]. Ertugrul, N. "Position Estimation and Performance Prediction for Permanent-Magnet Motor Drives." Ph.D. thesis, University of Newcastle, 1993.
- [6]. Ertugrul, N. "The Speed Control of Slip-Ring Induction Motor from Rotor Circuit and the Design of a Static Starting Circuit." M.Sc., Istanbul Technical University, Institute of Science and Technology, 1989.
- [7]. [http://www.image-sensors.com/\\_files/s5530\\_DataSheet.pdf](http://www.image-sensors.com/_files/s5530_DataSheet.pdf)

# SoC (SYSTEM-ON-CHIP) Power Consumption Refinement and Analysis

Parmender Singh, Dr Raj Kumar and Manish Sharma

Department of Electronics and Communication Engineering

Gurgaon Institute of Technology & Management, Gurgaon, Haryana, India

[parmender1979@yahoo.com](mailto:parmender1979@yahoo.com)

**Abstract-** As integrated circuits have become more and more complex, the ability to make post-fabrication changes will become more and more attractive. This ability can be realized using programmable logic cores. Currently, such cores are available from vendors in the form of a “hard” layout. An alternative approach is to use a “soft”, or synthesizable programmable logic core that can be synthesized using standard library cells. One such emerging technique is the System-on-a-Chip (SoC) design methodology. In this methodology, pre-designed and pre-verified blocks, often called cores or intellectual property (IP.), are obtained from internal sources or third-parties, and combined onto a single chip. As SoC design enters into mainstream usage, the ability to make post-fabrication changes will become more and more attractive. This ability can be realized using programmable logic cores. These cores are like any other IP in the SoC design methodology, except that their function can be changed after fabrication. This paper outlines ways in which programmable logic cores can simplify SoC design, and describes some of the challenges that must be overcome if the use of programmable logic cores is to become a mainstream design technique.

## I. INTRODUCTION

The drivers for the technology improvements are the applications that are very heterogeneous in nature. They range from high-performance multicenter clusters to portable appliances for wireless communication and embedded applications. The applications complexity will further increase which sets ever-increasing demands on the systems on which they are executed Video and audio applications in personal portable communication devices Entertainment applications in game, music, and video players Computer networking applications with high data rate requirements

Television production and broadcasting applications to process (e.g. compress, encrypt, decrypt, and decompress) high-definition video in real-time

In order to fulfill the requirements for performance, size, energy consumption, and reliability, the complete system must be more often implemented in a single chip, also called System-on-Chip (SoC). Leading-edge systems-on-chip (SoC) being designed today could reach 20 Million gates and 0.5 to 1 GHz operating frequency. In order to implement such systems, designers are increasingly relying on reuse of intellectual property (IP) blocks. Since IP blocks are pre-designed and pre-verified, the designer can concentrate on the complete system without having to worry about the correctness or performance of the individual components.

## II. PROGRAMMABLE LOGIC IP CORES IN SOC DESIGN

No matter how seamless the SoC design flow is made, and no matter how careful a SoC designer is, there will always be

some chips that are designed, manufactured, and then deemed unsuitable. This may be due to design errors not detected by simulation or it may be due to a change in requirements. This problem is not unique to chips designed using the SoC methodology. However, the SoC methodology provides an elegant solution to the problem: one or more programmable logic cores can be incorporated into the SoC. The programmable logic core is a flexible logic fabric that can be customized to implement any digital circuit after fabrication. Before fabrication, the designer embeds a programmable fabric (consisting of many uncommitted gates and programmable interconnects between the gates). After the fabrication, the designer can then program these gates and the connections between them.

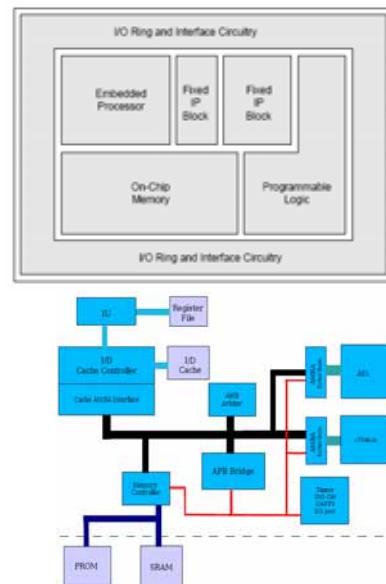


Fig-2 SoC Block Diagram

## III.THERMAL MANAGEMENT SYSTEM

Increases in circuit density and clock speed in modern VLSI designs have brought thermal issues into the spotlight of high-speed integrated circuit design. Local overheating in one spot of a high-density circuit, such as CPUs and high-speed mixed-signal circuits can cause a whole system to crash due to resulting clock synchronization problems, parameter mismatches or other coefficient changes due to the uneven heat-up on a single chip

## IV.ARCHITECTURE OF THE SYSTEM

The architecture of the thermal management circuitry is divided into two portions: the thermal management circuit

blocks and the system integration blocks. The former represent the designed thermal management system, and the latter represent the interface to the target system. The designed thermal management system could be applied to different SoC designs. The block diagram of the dynamic thermal management circuit is shown in figure given below. The thermal management circuit blocks are the white boxes with shadows; the gray boxes represent the system integration blocks

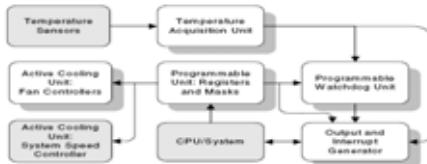


Fig-3 Energy Management for Soc Design

One of the biggest problems in complicated and high-performance SoC design is management of energy and/or power consumption. Dynamic power consumption is the major factor of energy consumption in the current CMOS digital circuits. The dynamic power consumption is affected by supply voltage, load capacitance and switching activity. The approach to control supply voltage, load capacitance and switching activity dynamically and statically in system architecture and algorithm design levels have been designed. In the future CMOS technology, leakage power consumption becomes dominant, because the threshold voltages are scaled as the transistor size shrinks. The techniques for reducing leakage power in system architecture design are being summarized. The contents include the followings:

- (1) Power and energy consumptions in SoC design,
- (2) Tradeoff between energy and performance,
- (3) Techniques for reducing dynamic power consumption.

#### V. POWER AND ENERGY CONSUMPTIONS IN SOC

The energy consumption of a system, E, can be defined as the summation of both spatial and temporal power consumption of circuits.

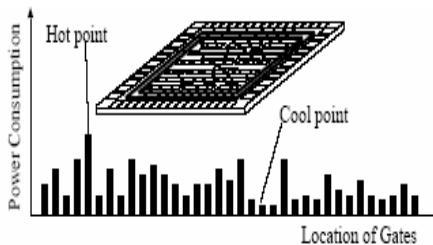


Fig-4a Power Dissipation vs. Energy Dissipation

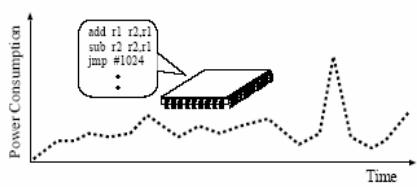


Fig-4b Power Dissipation vs. Energy Dissipation

$$P = P_{dynamic} + P_{leak} = \sum_{g \in G} SA(g) \cdot CL(g) \cdot V_{DD}(g)^2 + P_{leak}(g)$$

$$E = \int_0^t P dt$$

P: Power consumption of the target system

$P_{dynamic}$ : Dynamic power consumption of the target system

$P_{leak}$ : Leakage power consumption of the target system

SA(g): switching activity of gate g (expected number of 0->1 transitions per second)

CL(g): load capacitance of g

$V_{DD}(g)$ : operation voltage of g

t: Execution time of an application program

We treat the energy consumption, E, as an objective function to be optimized, because the energy consumption is close related to the heat and reliability of chips, battery life time of portable devices, and the number of nuclear and gas turbine power stations required. The main approach is detecting a spatial and temporal hot spot and reducing the power consumption of the spot. Since the power consumption, P, dynamically changes according to the behavior of the software running on a chip and a location of the logic gate on the chip as shown in figures above, both the software and the hardware should be taken into account for reducing the energy consumption of a SoC chip. As one can see from equations given above, we can reduce the energy consumption of the SoC chip by lowering SA(g), CL(g),  $V_{DD}(g)$ ,  $P_{leak}(g)$  and t. However, lowering these parameters sometimes causes an increase of the execution time, a degradation of computational quality, system reliability and design flexibility. The key point of the energy reduction in SoC design is considering design tradeoffs among energy consumption, performance, computational quality, system reliability and design flexibility. The goal is minimizing the energy consumption under the constraint of performance, computational quality, system reliability and/or design flexibility. There is a third source of power consumption, short-circuit power, which results from a short-circuit current-path between the power supply and ground during switching. Short-circuit power is projected to be constant around 10% of total power consumption.

#### A. Techniques for Lowering Operating Voltage

Since energy dissipation is quadratically proportional to supply voltage lowering the VDD has a strong impact on the energy reduction. However, the following drawbacks should be taken into account;

1. Loss of compatibility to external voltage standards,
2. Performance degradation, and
3. Reliability issues (very low voltage).

The following three ways for lowering the operating voltage with out sacrificing the performance of the system.

1. Parallelize tasks so that the performance does not degrade even in a low voltage operation. We refer this approach as static voltage scaling.
2. Use the maximum available supply voltage for gates on a critical-path and use a lower supply voltage for the other gates. We refer this approach as multiple voltage assignment.

3. Lower the clock frequency and operating voltage when the maximum performance is not needed. We refer this approach as dynamic voltage scaling.

### B . Techniques for Reducing Switching Activity

Lowering the switching activity is a very promising way of decreasing the power consumption. There are numerous researches on this issue. In this section, we introduce system level approaches for reducing the switching activity. System level switching activity reduction can be categorized as follows:

- Turn off unused HW modules.
- Adjust datapath, the bit width of buses and operational units in a system

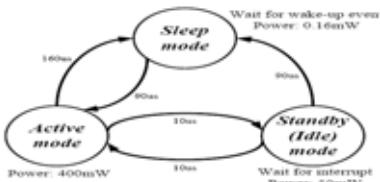


Fig-5a Example Dynamic Power Management

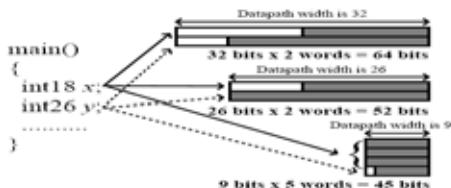


Fig-5b Example Dynamic Power Management

## VI CONCLUSION

System on Chip (SoC) describes an evolving paradigm for the timely design of integrated circuits (ICs) that contain tens to hundreds of millions of transistors implementing a large variety of different functions. Programmable logic adds another dimension that designers must come to grips with before the full potential of these cores can be realized. The innovative temperature offset monitoring provides a mechanism for system-on-chip designs to monitor the temperature offset across the system and enhance stability. With proper handling of this information, the system not only prevents failure but also enhances performance by controlling each subcomponent's operation speed with feedback from thermal information. With minimum overhead in chip area and system resources, this design provides intricate control and optimal thermal management on chip, upon which a complete dynamic thermal management system for modern computer designs can be implemented.

## VII REFERENCES

- [1]. C. Matsumoto, "LSI Logic ASICs to add Programmable Logic Cores", E.E. Times, August 29, 1999.
- [2]. S. Ohr, "ADI Taps Systolix Processor Array," E.E. Times, April 21, 2000.
- [3]. "Lucent Introduces ORCA Series 4 FPGA," Programmable Logic News and Views, pages 7-11.
- [4]. Hardware-Software Co-Design of Embedded Systems: the POLIS Approach. Boston, MA: Kluwer Academic Publishers, 1997.
- [5]. Altera homepage, Sept. 2006, <http://www.altera.com>.
- [6]. T. Arpinen, P. Kukkala, E. Salminen, M. Hännikäinen, and T. D. Hämäläinen, "Configurable multiprocessor platform with RTOS for distributed execution of UML 2.0 designed applications," in Proceedings of the Design, Automation and Test in Europe, Mar. 2006, pp. 1324–1329.
- [7]. Augé, F. Pérot, F. Donnet, and P. Gomez, "Platform-based design from parallel C specifications," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 24, no. 12, pp. 1811–1826, Dec. 2005.
- [8]. Baghdadi, N.-E. Zergainoh, W. O. Cesario, and A. A. Jerraya, "Combining a performance estimation methodology with a hardware/software codesign flow supporting multiprocessor systems," IEEE Transactions on Software Engineering, vol. 28, no. 9, pp. 822–831, Sept. 2002.
- [9]. F. Balarin, Y. Watanabe, H. Hsieh, L. Lavagno, and C. Passerone, "Metropolis: an integrated electronic system design environment," IEEE Computer, vol. 36, no. 4, pp. 45–52, Apr. 2003.
- [10]. Benveniste, P. Caspi, S. A. Edwards, N. Halbwachs, P. Le Guernic, and R. de Simone, "The synchronous languages 12 years later," Proceedings of the IEEE, vol. 91, no. 1, pp. 64–83, Jan. 2003.
- [11]. M. Berekovic, S. Flagel, H.-J. Stolberg, L. Fribe, S. Moch, M. B. Kulaczewski and P. Pirsch, "HiBRID-SoC: a multi-core architecture for image and video applications," in Proceedings of the International Conference on Image Processing, vol. 2, Sept. 2003, pp. 101–104.

# Wavelet Based Image Coding Techniques: A Study

Er. Rashima Mahajan<sup>a</sup> Lecturer and Er. Gurpadam Singh, <sup>b</sup> Asst. Prof.

<sup>a</sup>Department of Electronics & Communication Engineering, ACE, Sohna, Gurgaon, INDIA

<sup>b</sup>Department of Electronics & Communication Engineering, BCET, Gurdaspur, INDIA

rashimamahajan@gmail.com , gurpadam@yahoo.com

**Abstract-** Image compression is the application of data compression on digital images. Image sample values are represented with bits. Compression is only possible if some of these bits are redundant. The primary goal of compression is to reduce redundancy of an image data, leaving the informational content in order to store or transmit data in an efficient form. This leads an image to be represented using a lower number of bits per pixel, without losing the ability to reconstruct the image. Compression of images saves storage capacity, channel bandwidth and transmission time. The degree to which an image may be compressed depends upon the number of bits required to represent an image with an allowable level of distortion. This paper first presents a review on wavelet based coding algorithms EZW, SPIHT, EBCOT and JPEG2000, to encode and compress an image data and then the simulative comparison of SPIHT and JPEG2000 by looking on the limitations and possibilities of each.

## I. INTRODUCTION

A digital image is a rectangular array of picture elements, arranged in 'm' rows and 'n' columns. The expression  $m \times n$  is called the resolution of an image and the elements are called pixels. Due to the increasing traffic caused by multimedia information and digitized representation of images; image compression has become a necessity. Image coding and compression techniques; convert the images into the form image that require low memory storage space, smaller bandwidth for transmission, high PSNR with acceptable image quality. In digital image data, there is statistical redundancy which should be reduced. Statistical image redundancy implies inter-pixel redundancy which comes from the fact that adjacent pixels of an image are not independent of each other. They are correlated to each other in space domain. Thus to achieve compression, a less correlated representation of the image has to be obtained by reducing the redundant information. Basically, an image compression scheme consists of a decorrelator, followed by a quantizer and an entropy encoding stage. The purpose of decorrelator is to remove the spatial redundancy eg. DCTs, DWTs. The quantizer introduces a distortion to allow a decrement in the entropy rate to be achieved. Once a signal has been decorrelated, it is necessary to find a compact representation of its coefficients. An entropy coding algorithm is then used to map such coefficients into codewords in such a way that the average codeword length is minimized.

This paper is organized as follows: **Section II** describes the performance measures for image compression. **Section III** deals with the scheme of Wavelet based coding and **Section IV** covers different coding techniques. **Section V** provides the simulation results with SPIHT and JPEG and the paper concludes with **Section VI**.

## II. PERFORMANCE MEASURES FOR IMAGE COMPRESSION

The different compression algorithms can be compared based on certain performance measures. To evaluate the performance of the image compression algorithms, the metrics taken into consideration are: *Amount of compression (Compression ratio)* and *Quality of compression (PSNR)*:

- Compression Ratio is defined as the ratio between the uncompressed size and the compressed size of an image. An image possessing higher compression ratio requires less storage space and less time to transmit.

$$\text{Compression ratio} = \frac{\text{Uncompressed size}}{\text{Compressed size}}$$

- Distortion is quantified by a parameter called Mean Square Error (MSE). MSE refers to the average value of the square of the error between the original signal and the reconstruction. The important parameter that indicates the quality of the reconstruction is the peak signal-to-noise ratio (PSNR). PSNR is defined as the ratio of square of the peak value of the signal to the mean square error, expressed in decibels. Thus, PSNR is most commonly used as a measure of quality of reconstruction in image compression. It is most easily defined via the mean squared error (MSE) which for two  $m \times n$  monochrome images I and K where I is the reconstructed image of K is defined as:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \|I(i, j) - K(i, j)\|^2$$

Where MSE = Mean squared error

$m \times n$  = size of a monochrome image

K (i, j) = Input image

I (I, j) = Reconstructed image

The PSNR (db) is defined as:

$$PSNR = 10 \log_{10} \left( \frac{MAX_I^2}{MSE} \right) = 20 \log_{10} \left( \frac{MAX_I}{\sqrt{MSE}} \right)$$

Where PSNR = Peak signal to noise ratio in decibels

MAX<sub>I</sub> = is the maximum pixel value of the image,

eg. the pixels are represented using 8 bits per sample, and then this is 255 ( $2^8 - 1$ ). Generally, if samples are represented with Q bits per pixel (bpp), max possible value of MAX<sub>I</sub> is  $2^Q - 1$ . Increasing PSNR represents increasing fidelity of compression. A lower value for MSE means lesser error, and as seen from the inverse relation between the MSE and PSNR, this translates to a high value of PSNR. Logically, a higher value of PSNR is good because it means that the ratio of Signal to Noise is higher. Here, the 'signal' is the original image, and the 'noise' is the error in reconstruction. So, a compression scheme having a lower MSE (and a high PSNR), can be recognized as a better one

### III. WAVELET CODING

Wavelet-based coding is more robust under transmission and decoding errors. It avoids blocking artefacts [1] and also provides progressive transmission of images. Discrete wavelet transform (DWT) has emerged as a popular technique for image coding applications. DWT [2] has high decorrelation and energy compaction efficiency. Wavelet coding utilizes the concept of overlapping basis functions. This overlapping nature of the wavelet transform removes blocking artifacts. Wavelet-based coding [3] provides noticeable improvements in picture quality at higher compression ratios. The wavelet transform is a powerful tool for analysis and compression of signals. It uses two functions: scaling and wavelet functions, for the transform. The wavelet function is equivalent to a high pass filter and produces the high frequency components (details) of the signal at its output [1]. The scaling function is equivalent to a low pass filter and passes the low frequency components (approximations) of the signal. The wavelet coefficients are retained and represent the details in the signal. The scaling coefficients are decomposed further using another set of low pass and high pass filters. This sort of decomposition is called pyramidal decomposition.

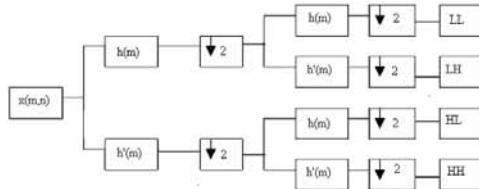


Figure 1: Sub band decomposition of an image.

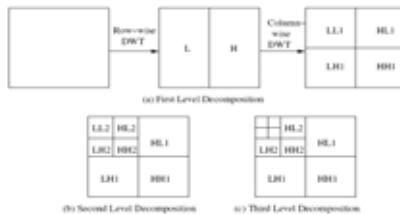


Figure 2: 2-D Wavelet Decomposition

A 2D DWT of an image is obtained by using Low pass & high pass filters successively shown in Fig 1. When a signal passes through these filters, it is split into two bands. The low pass filter, which corresponds to an averaging operation, extracts the coarse information of the signal. The high pass filter, which corresponds to a differencing operation, extracts the detail information of the signal. The output of the filtering operations is then decimated by two. A two-dimensional transform can be accomplished by performing two separate one-dimensional transforms. First, the image is filtered along the x-dimension using low pass and high pass analysis filters and decimated by two. Low pass filtered coefficients are stored on the left part of the matrix and high pass filtered on the right. Because of decimation, the total size of the transformed image is same as the original image. Then, it is followed by filtering

the sub-image along the y-dimension and decimated by two. Finally, the image has been split into four bands denoted by LL, HL, LH, and HH, after one level of decomposition. The LL band is again subject to the same procedure. This process of filtering the image is called pyramidal decomposition of image [4]. This is depicted in Fig. 2. The reconstruction of the image can be carried out by reversing the above procedure and it is repeated until the image is fully reconstructed.

A group of transforms coefficients resulting from the same sequence of low pass and high pass filtering operations, both horizontally and vertically are called sub bands. Thus, the DWT [4] separates an image into a lower resolution approximation image (LL) as well as horizontal (HL), vertical (LH) and diagonal (HH) detail components.

The number of decompositions performed on original image to obtain sub bands is called sub band decomposition level. The total number of sub bands for a given 'K' level decomposition is  $3K+1$ . Figure 2 shows the number of sub bands and resolution levels for K=1, K=2 and K=3 respectively.

### IV. CODING TECHNIQUES

A variety of novel and sophisticated wavelet-based image coding schemes have been developed. These include *EZW*, *EBCOT*, *SPIHT*, *JPEG 2000*, *SPIHT using 2D dual tree DWT*.

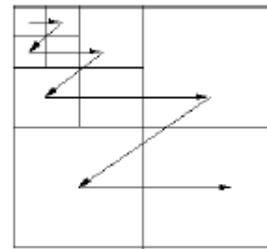


Figure 3: EZW scanning order [5].

#### A. EZW (Embedded Zero tree Wavelet)

The Embedded Zero tree Wavelet, developed by *Shapiro* [5] marked the beginning of a new era of wavelet coding. The two important features of the EZW coding are significance map coding and successive approximation quantization. This algorithm using hierarchical nature of the wavelet transform as it forms a tree structure, does not code the location of significant coefficients but instead codes the location of zeros. The EZW encoder is based on progressive encoding also known as embedded coding to compress an image into a bit stream with increasing accuracy. With an embedded bit stream, the reception of code bits can be stopped at any point and the image can be decompressed and reconstructed. This technique was extremely fast in execution as compared to the complex *block coding* techniques and produced an embedded bit stream. A zerotree is a quad-tree of which all nodes are equal to or smaller than the root. Here, encoding of the wavelet coefficients is done in decreasing order, in several passes. For every pass a threshold is chosen against which all the wavelet coefficients are measured. If a wavelet coefficient is larger than the threshold it is encoded and removed from the image, if it is smaller it is left for the next pass. When all the wavelet

coefficients have been visited the threshold is lowered and the image is scanned again to add more detail to the already encoded image. This process is repeated until all the wavelet coefficients have been encoded completely. EZW encoding uses a predefined scan order as shown in Figure 3, to encode the position of the wavelet coefficients so that the decoder will be able to reconstruct the encoded signal.

EZW encoding does not really compress anything; it only reorders wavelet coefficients in such a way that they can be compressed very efficiently. An EZW encoder should therefore always be followed by an arithmetic encoder. The next scheme, called SPIHT, is an improved form of EZW which achieves better compression and performance than EZW.

#### B. SPIHT (Set Partitioning in Hierarchical Tree)

Said and Pearlman [6] further enhanced the performance of EZW by presenting a more efficient and faster implementation called set partitioning in hierarchical tree. SPIHT achieved better performance than the EZW without having to use the arithmetic encoder and so the algorithm was computationally more efficient. The SPIHT uses a more efficient subset partitioning scheme. It is also a progressive transmission coder that produces embedded bit-streams. This means that the bit-stream can be truncated at any instant, and is then guaranteed

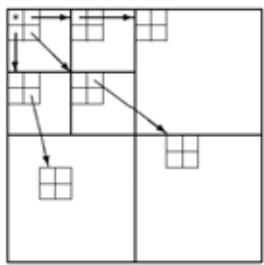


Figure 4: Spatial – orientation Tree [6]

to yield the best possible reconstruction. This algorithm is capable of lossless as well as lossy compression. It works on the principle of spatial relationship among the wavelet coefficients at different levels and frequency sub-bands in the pyramid structure of wavelet decomposition. This pyramid structure is commonly known as spatial orientation tree as shown in Figure 4. If a given coefficient at location is significant in magnitude then some of its descendants will also probably be significant in magnitude. It employs an iterative partitioning of sets of transform coefficients, in which the tested set is divided when the maximum magnitude within it exceeds a certain threshold. When the set passes the test and is hence divided, it is said to be significant. Otherwise it is said to be insignificant. Insignificant sets are repeatedly tested at successively lowered thresholds until isolated significant pixels are identified. This procedure sorts sets and coefficients by the level of their threshold of significance. Since the binary outcomes of these tests are put into the bit stream as a '1' or '0', the binary decisions are taken using binary search algorithm and these binary decisions can optionally be arithmetically coded. The produced bit stream is SNR scalable only.

#### C. EBCOT (Embedded Block Coding With Optimized uncation)

The drawback of EZW and SPIHT algorithms is that they are not ‘resolution scalable’. EBCOT proposed by Taubman [7] is both ‘SNR scalable’ and ‘resolution scalable’. This algorithm is block based i.e. it encodes blocks independently (Embedded block based coding). Due to this, a particular region of interest (ROI) can be decoded separately without decoding the full image. Also the embedded bit stream includes the information about the number of bits to be decoded, to give the optimal reconstructed quality at a given bit rate. Apart from providing additional functionality, it gives better performance than SPIHT.

The EBCOT algorithm [7] uses a wavelet transform to generate the subband coefficients which are then quantized and coded. Each subband is then partitioned into relatively small blocks of samples and generates a separate highly scalable bit-stream to represent each code-block. This bit-stream is composed of a collection of quality layers and unwanted layers are discarded to obtain SNR scalability. The subbands of an image obtained by applying DWT are organized into increasing resolution levels and this makes EBCOT resolution

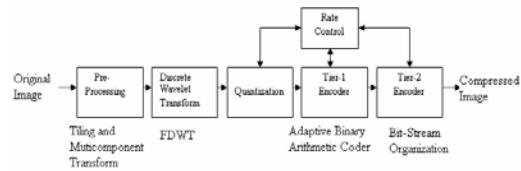


Figure 5: Codec structure of the JPEG2000 encoder [8].

scalable. The lowest resolution level consists of the single LL subband. Each successive resolution level contains the additional subbands, which are required to reconstruct the image with twice the horizontal and vertical resolution.

#### D. JPEG2000

JPEG2000 [8] standard is based on EBCOT [7]. The performance of EBCOT decreases as the number of layers increases because the overhead associated with identifying the contributions of each code-block to each layer grows. JPEG2000 [8] is a new still image compression standard developed by the Joint Photographic Experts Group supporting both lossy and lossless encoding. JPEG2000 picture has much better image quality even with low bit rates. JPEG2000 uses wavelet transform + bit plane coding + Arithmetic entropy coding as shown in Figure 5. The main feature of this standard is Region-of-interest (ROI) coding.

An image is divided into tiles and each tile is transformed using discrete wavelet transform [10]. The wavelet coefficients obtained thus are quantized. An arithmetic coder and the bit-stream coder both comprise the entropy coder. The arithmetic coder removes the redundancy in the data by assigning short code-words to the more probable events and longer code-words to the less probable ones. Tier 1, here provides a collection of bit streams by generating one independent bit stream for each code-block. Tier 2 multiplexes the bit streams for inclusion in the code stream in a succession of packets. These packets, along with additional headers, form the final JPEG2000 code-

stream. JPEG2000 provides better resolution, SNR, error-resilience, arbitrarily shaped region of interest [9], lossy and lossless coding, etc., all in a unified algorithm.

## V. SIMULATION RESULTS

This paper summarizes the simulation results of coding the ‘Cameraman’ image (256x256) using SPIHT and JPEG2000 algorithms. The results are obtained in terms of Compression ratio and PSNR values as presented in Table 1. We performed the simulation in MATLAB. The data in the Table 1 reveals that the best results in terms of compression ratio are obtained when SPIHT coding is used for image for 3<sup>rd</sup> level of decomposition and in terms of PSNR are obtained when JPEG2000 is used again for the 3<sup>rd</sup> level of decomposition. Thus, the best choice on the basis of compression ratio is SPIHT and on the basis of PSNR is JPEG2000.



Figure 6: Cameraman ‘test’ image

## VI. CONCLUSION

From the above results and discussion we conclude that, although SPIHT provides comparatively higher compression ratio but the image quality in this case reduces as PSNR value decreases significantly here with increase in compression ratio. The improved image quality can be obtained using JPEG2000 as it offers higher values of PSNR as compared to SPIHT with accepted values of compression ratio. Increasing PSNR represents increasing fidelity of compression. Generally, when the PSNR is 40 dB or larger, then the two images are virtually indistinguishable by human observers. Compression ratio also improves with higher values of Quantization but quality of the reconstructed image degrades *i.e.* the performance of the coding algorithm degrades in terms of PSNR value.

## VII. REFERENCES

1. G.Piella, H.J.A.M.Heijmans, An Adaptive Update Lifting Scheme with Perfect Reconstruction, IEEE Transactions on Signal Processing, Vol. 50, No. 7, pp. 1620-1630, July, 2002.
2. Anotonini M, Barlaud M, Mathieu P, et al, “Image coding using wavelet transform”, IEEE Trans on Image Processing, vol. 1, pp. 205-220, 1992.
3. Nirendra K.C. and W.A.C. Fernando, “Effects of DWT Resolutions in Reduction of Ringing Artifacts in JPEG- 2000”,
4. Telecommunication program, Asian Institute of Technology, Oct 2001.
- Strang, G. and Nguyen, T. Wavelets and Filter Banks, Wellesley-Cambridge Press, Wellesley, MA, 1996, <http://www-math.mit.edu/~gs/books/wfb.html>.
5. Shapiro J M, “Embedded image coding using zerotrees of wavelet coefficients”, IEEE Trans on signal processing, ol. 41, pp.3445-3462, 1993.
6. A. Said and W.A. Pearlman, A new, fast, and efficient image codec based on set partitioning in hierarchical trees, IEEE Trans Circuits Syst Video Technol, pp. 243-250 (1996).

7. Huakai Zhang, Jason Fritts , “EBCOT coprocessing architecture for JPEG2000” , *Proc. of SPIE Applications of digital image processing XXIV*, San Diego, California,U.S.A, August 2001 vol. 4471, pp.276-283.
8. M. W. Marcellin, M. Gormish, A. Bilgin, M. Boliek, “An Overview of JPEG 2000,” *Proc. IEEE Data Compression Conference*, Snowbird, Utah, March 2000.
9. D. Santa Cruz and T. Ebrahimi: “An Analytical Study of the JPEG2000 Functionalities”, to be presented at *IEEE Int. Conf. Image Processing*, Vancouver, Canada, Sep. 2000.
10. Shaorong Chang and Lawrence Carin, “A Modified SPIHT Algorithm for Image Coding With a Joint MSE and Classification Distortion Measure”, *IEEE TRANSACTIONS ON IMAGE PROCESSING*, VOL. 15, NO. 3, MARCH 2006.

# Electronically Controllable Current-mode Biquad Active-C Filter using CCCCTAs

<sup>1</sup> Jitendra Mohan, <sup>2</sup>Sudhanshu Maheshwari, <sup>3</sup>Sajai Vir Singh, <sup>4</sup>Durg Singh Chauhan

<sup>1,3</sup>Jaypee University of Information Technology, Waknaghat, Solan-173215 (India)

<sup>2</sup>Z. H. College of Engineering and Technology, Aligarh Muslim University, Aligarh-202002 (India)

<sup>4</sup>Institute of Technology, Banaras Hindu University, Varanasi-221005 (India)

jitendramv2000@gmail.com, Sudhanshu\_maheshwari@rediffmail.com, sajajvir@rediffmail.com, pdschauhan@gmail.com

**Abstract-** This paper presents an electronically tunable current-mode universal biquad filter using current controlled current conveyor trans-conductance amplifiers (CCCCTAs) and grounded capacitors. The proposed filter employs only three CCCCTAs, three grounded capacitors. The proposed filter realizes low pass(LP), band pass(BP) and high pass(HP) responses simultaneously. The filter can also realize regular notch and all pass(AP) responses with interconnection of relevant output currents .The low pass notch (LPN) and high pass notch (HPN) responses can also be realize with out passive component matching conditions. The circuit enjoys an independent current control of pole frequency and bandwidth as well as quality factor and bandwidth. Both the active and passive sensitivities are no more than unity. The validity of proposed filter is verified through PSPICE simulations.

**Keywords**— CCCCTA, filter, current mode.

## I. INTRODUCTION

In analog signal processing, continuous time (CT) filters play an important role for realizing frequency selective circuits. In analog circuit design, current mode approach has gained considerable attention. This stems from its inherent advantages such as wider bandwidth, larger dynamic range, less power consumption, simple circuitry [1]. Second generation current conveyors (CCIIs) have been found very useful in filtering applications. The applications and advantages in the realization of various active filter transfer function using current conveyors have received considerable attention [2-7]. However, CII-based filters do not offer electronic adjustment properties. The second generation current controlled conveyor (CCCII) introduced by Fabre at [8] can be extended to the electronically adjustable domain for different applications In recent past, there has been greater emphasis on design of current mode current controlled universal active filters[9-20] using CCCIIs. The circuit reported in [9-11] uses three CCCIIs and two grounded capacitors whereas [12] uses two CCCIIs and two capacitors but one of the capacitor is floating which is the disadvantage from the IC fabrication point of view. Moreover, either one or two of the outputs[9-12] are available on the passive components. Hence one or two additional current conveyor(s) will be required to implement all the standard universal filter functions (LP, HP, BP, Notch and AP). The circuit proposed in refs. [13,14,15,16,17,18] enjoy high impedance outputs and can realize LP, HP, BP, Notch and AP responses by connecting appropriate output currents without any passive component matching conditions and consists of either four CCCIIs[13,14,15] or three CCCIIs[16,17,18], with two grounded capacitors. However, all these circuits [13-18] use either dual outputs or multi outputs type of CCCIIs (both plus and minus type of outputs) which increase the hardware of the

circuit. The circuit in [19] has orthogonal tuning capability of the characteristic parameters  $\omega_0$  and Q, grounded capacitors and high impedance outputs. However, it uses too many active components (five CCCIIs) and passive components (three capacitors) and capacitor value matching for the notch responses. The circuit in [20] uses three multi outputs CCCIIs, two grounded capacitors and one grounded resistor. Use of resistor in this circuit requires large space area which is the disadvantage from the IC fabrication point of view. The circuit in [21] involves three CCCIIs and two capacitors, and it can provide high impedance outputs. However, the characteristic parameters ( $\omega_0$  and Q) can not be orthogonally adjusted.

Recently, a new current mode active building block, which is called as a current controlled current conveyor trans-conductance amplifier (CCCCTA), has been proposed [22] which is the modified version of CCTA. This device can be operated in both current and voltage modes, providing flexibility. In addition, it can offer several advantages such as high slew rate, high speed, wider bandwidth and simpler implementation. Moreover in the CCCCTA one can control the parasitic resistance at X ( $R_X$ ) port by input bias current. It is suited for realization of electronically tunable filters design [23].

From the above study it is clear that there exist no current mode filter based on CCCIIs or CCCCTAs which can realize low pass and high pass notch responses along with all standard transfer functions (LP,HP,BP, regular notch and AP) with out passive component matching conditions. keeping this point in consideration, a new electronically tunable current mode universal biquad filter using CCCCTA is proposed. In this paper a new electronically tunable current mode universal biquad filter using CCCCTA is proposed which uses two CCCCTAs and three grounded capacitors. The filter circuit realizes low pass, band pass and high -pass responses simultaneously. The filter can also realize regular notch and all pass responses with interconnection of relevant output currents .The low pass and high pass notch responses can also be realize without passive component matching conditions. It is clear from sensitivity analysis that biquad filter has very low sensitivities with respect to circuit active and passive components. Additionally the circuit enjoys an independent current control of parameters  $\omega_0$  and  $\omega_0/Q$ , and Q and  $\omega_0/Q$ . Moreover, the low pass and band pass gain can be independently tuned by external biasing current of active elements without disturbing the centre frequency, quality factor and bandwidth. The performances of proposed circuit are illustrated by PSPICE simulations.

## II. BASIC CONCEPT OF CCCCTA

The CCCCTA, shown in fig.1 is described by the following relationships, where  $\alpha$ ,  $\beta$ , and  $\gamma$  are transferred error values deviated from one.  $R_x$  and  $g_m$  are the parasitic resistance at x terminal and transconductance of CCCCTA.

$$I_Y = 0, V_X = \beta V_Y + I_X R_X, I_Z = \alpha I_X,$$

$$I_O = -\gamma g_m V_Z \quad (1)$$

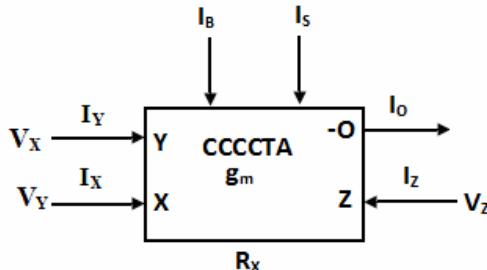


Fig.1 CCCCTA Symbol

For a bipolar CCCCTA, the  $R_x$  and  $g_m$  can be expressed to be

$$R_X = \frac{V_T}{2I_B} \text{ and } g_m = \frac{I_S}{2V_T} \quad (2)$$

where  $I_B$  and  $I_S$  are the bias currents and  $V_T$  is the thermal voltage of the CCCCTA

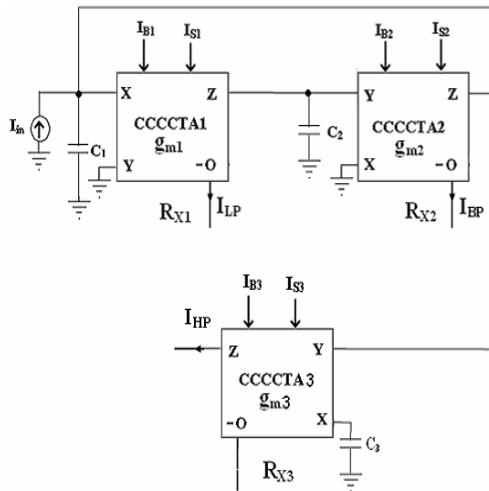


Fig.2 An electronically tunable current mode universal biquad filter based on the CCCCTA and grounded capacitors

## III. PROPOSED CURRENT-MODE UNIVERSAL FILTER

The proposed current mode universal filter is shown in Fig. 2. It is based on three CCCCTA and three grounded capacitors. By taking  $I_{B3}$  too high so that  $R_{X3}$  is very small and negligible where  $R_{X3}$  is input resistance at terminal X of CCCCTA3. The transfer functions of the proposed circuit  $T_{LP}(s)$ ,  $T_{BP}(s)$  and  $T_{HP}(s)$  for the current outputs ( $I_{LP}(s)$ ,  $I_{BP}(s)$  and  $I_{HP}(s)$ ) can then be given by

$$T_{LP}(s) = \frac{I_{LP}(s)}{I_{in}(s)} = \frac{\gamma_1 \alpha_1 g_{m1} R_{X2}}{s^2 C_1 C_2 R_{X1} R_{X2} + s C_2 R_{X2} + \beta_2 \alpha_1 \alpha_2} \quad (3)$$

$$T_{BP}(s) = \frac{I_{BP}(s)}{I_{in}(s)} = \frac{-\gamma_2 s g_{m2} R_{X1} R_{X2} C_2}{s^2 C_1 C_2 R_{X1} R_{X2} + s C_2 R_{X2} + \beta_2 \alpha_1 \alpha_2} \quad (4)$$

$$T_{HP}(s) = \frac{I_{HP}(s)}{I_{in}(s)} = \frac{\alpha_3 \beta_3 s^2 C_2 C_3 R_{X1} R_{X2}}{s^2 C_1 C_2 R_{X1} R_{X2} + s C_2 R_{X2} + \beta_2 \alpha_1 \alpha_2} \quad (5)$$

The pole frequency ( $\omega_o$ ), the quality factor (Q) and Bandwidth(BW)  $\omega_o/Q$  of each filter response can be expressed as

$$\omega_o = \left( \frac{\beta_2 \alpha_1 \alpha_2}{C_1 C_2 R_{X1} R_{X2}} \right)^{\frac{1}{2}}, Q = \left( \frac{\beta_2 \alpha_1 \alpha_2 C_1 R_{X1}}{C_2 R_{X2}} \right)^{\frac{1}{2}} \quad (6)$$

$$\text{and } BW = \frac{\omega_o}{Q} = \frac{1}{C_1 R_{X1}} \quad (7)$$

For the ideal case, the  $\omega_o$  and Q are changed to

$$\omega_o = \left( \frac{1}{C_1 C_2 R_{X1} R_{X2}} \right)^{\frac{1}{2}}, Q = \left( \frac{C_1 R_{X1}}{C_2 R_{X2}} \right)^{\frac{1}{2}} \quad (8)$$

$$BW = \frac{\omega_o}{Q} = \frac{1}{C_1 R_{X1}} \quad (9)$$

Substituting intrinsic resistances as depicted in (2), it yields

$$\omega_o = \frac{2}{V_T} \left( \frac{I_{B1} I_{B2}}{C_1 C_2} \right)^{\frac{1}{2}}, Q = \left( \frac{C_1 I_{B2}}{C_2 I_{B1}} \right)^{\frac{1}{2}} \quad (10)$$

From (10), by maintaining the ratio  $I_{B1}$  and  $I_{B2}$  to be constant, it can be remarked that the pole frequency can be adjusted by  $I_{B1}$  and  $I_{B2}$  without affecting the quality factor. In addition, bandwidth (BW) of the system can be expressed by

$$BW = \frac{\omega_o}{Q} = \frac{2I_{B1}}{C_1 V_T} \quad (11)$$

We found that the bandwidth can be linearly controlled by  $I_{B1}$ . From Eq. (10) and (11), we can see that parameter  $\omega_o$  can be controlled electronically by adjusting bias current  $I_{B2}$  with out disturbing parameter  $\omega_o/Q$ . Furthermore, parameter Q can also be controlled by adjusting the bias current  $I_{B2}$  with out disturbing parameter  $\omega_o/Q$ . The gains of the low pass, high pass and band pass can be expressed as

$$G_{LP} = \frac{I_{S1}}{4I_{B2}}, G_{BP} = \frac{I_{S2}}{4I_{B1}}, G_{HP} = \frac{C_3}{C_1} \quad (12)$$

From Eq.(12), it can be seen that, the low pass and band pass gain can be independently tuned by biasing current ( $I_{S1}$ ) of CCCCTA1 and biasing current ( $I_{S2}$ ) of CCCCTA2 respectively, without disturbing the pole frequency, quality factor and bandwidth. It can be seen that the filter circuit can realize the low pass, band pass and high pass transfer function at current outputs of  $I_{LP}(s)$ ,  $I_{BP}(s)$  and  $I_{HP}(s)$ , respectively. The notch transfer function  $T_{Notch}(s)$  can be easily obtained from the currents  $I_{Notch}(s) = I_{LP}(s) + I_{HP}(s)$ . It is clear from Eq.(13) that one can obtain regular notch filter for  $I_{S1}=4I_{B2}$  note that since zero and pole frequency can take different values, one can also obtain low pass notch and high pass notch filters for  $I_{S1}>4I_{B2}$  and  $I_{S1}<4I_{B2}$  respectively. Also, all pass transfer function can be obtained from the currents  $I_{AP}(s)=I_{LP}(s) + I_{BP}(s) + I_{HP}(s)$ , by keeping  $g_{m1}R_{X2}=1$ ,  $g_{m2}R_{X1}=1$  and  $C_1=C_3$ . Thus, five different

circuit transfer functions can be realized by choosing the suitable current output branches. Moreover, capacitor  $C_3$  and input resistance  $R_{X3}$  at terminal X of CCCCTA3 result in a dominant pole in the high pass response, which restricts the frequency range of the filter.

$$T_{Notch}(s) = \frac{I_{Notch}(s)}{I_{in}(s)} = \frac{s^2 C_2 C_3 R_{X1} R_{X2} + g_{m1} R_{X2}}{s^2 C_1 C_2 R_{X1} R_{X2} + s C_2 R_{X2} + 1} \quad (13)$$

#### IV. SENSITIVITY ANALYSIS

The ideal sensitivities of the pole frequency, the quality factor and band width with respect to active and passive components can be found as

$$S_{C_1, C_2}^{\omega_o} = -\frac{1}{2}, S_{I_{B1}, I_{B2}}^{\omega_o} = \frac{1}{2}, S_{V_T}^{\omega_o} = -1 \quad (14)$$

$$S_{I_{B1}, C_2}^Q = -\frac{1}{2}, S_{C_1, I_{B2}}^Q = \frac{1}{2}, S_{V_T}^Q = 0 \quad (15)$$

$$S_{C_1, V_T}^{BW} = -1, S_{I_{B1}}^{BW} = 1, S_{C_2, I_{B2}}^{BW} = 0 \quad (16)$$

From the above calculations, it can be seen that all sensitivities are constant and equal or smaller than 1 in magnitude. The non-ideal sensitivities can be found as

$$S_{C_1, C_2, R_{X1}, R_{X2}}^{\omega_o} = -\frac{1}{2}, S_{\alpha_1, \alpha_2, \beta_2}^{\omega_o} = \frac{1}{2}, S_{\gamma_1, \gamma_2, \gamma_3, \beta_1, \beta_3, \alpha_3}^{\omega_o} = 0 \quad (17)$$

$$S_{R_{X2}, C_1}^Q = -\frac{1}{2}, S_{R_{X1}, C_2, \alpha_1, \alpha_2, \beta_2}^Q = \frac{1}{2}, S_{\gamma_1, \gamma_2, \gamma_3, \beta_1, \beta_3, \alpha_3}^Q = 0 \quad (18)$$

From the above results, it can be observed that all the sensitivities due to non-ideal effects are equal and less than one in magnitude.

#### V. SIMULATION RESULTS

To verify the theoretical analysis, PSPICE simulation has been used to confirm the proposed an electronically tunable current mode universal biquad filter based on the CCCCTA of Fig.2. In simulation, the CCCCTA is realized using BJT implementation [22] as shown in Fig. 3 with the transistor model of PR100N (PNP) and NP100N (NPN) of the bipolar arrays ALA400 from AT&T [24].

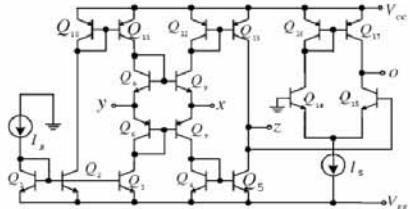


Fig.3 Internal Topology of CCCCTA

To obtain  $f_o = \omega_o / 2\pi = 1.49\text{MHz}$  at  $Q=1$ , the active and passive components are chosen as  $I_{B1} = I_{B2} = 45\mu\text{A}$ ,  $I_{B3} = 900\mu\text{A}$ ,  $I_{S1} = I_{S2} = 175\mu\text{A}$ ,  $I_{S3} = 800\mu\text{A}$  and  $C_1 = C_2 = C_3 = 370\text{pf}$ . Fig.4 Shows the simulated gain responses of the LP, HP and BP of the proposed circuit in fig.2. Fig5 shows the gain and phase response of AP. Fig.6 shows the gain response of regular notch

The supply voltages are  $V_{DD} = -V_{SS} = 1.85\text{V}$ . The simulated pole frequency is obtained as 1.35 MHz. Fig.7 and fig.8 show the gain responses of high pass notch(HPN) and low pass notch(LPN) respectively. Low pass notch is obtained with  $I_{B1} = 15\mu\text{A}$ ,  $I_{B2} = 90\mu\text{A}$ ,  $I_{B3} = 265\mu\text{A}$ ,  $I_{S1} = 550\mu\text{A}$ ,  $I_{S2} = 175\mu\text{A}$ ,  $I_{S3} = 800\mu\text{A}$ ,  $C_1 = C_2 = C_3 = 370\text{pf}$ . High pass notch is obtained with  $I_{B1} = 45\mu\text{A}$ ,  $I_{B2} = 150\mu\text{A}$ ,  $I_{B3} = 620\mu\text{A}$ ,  $I_{S1} = 300\mu\text{A}$ ,  $I_{S2} = 175\mu\text{A}$ ,  $I_{S3} = 800\mu\text{A}$ ,  $C_1 = C_2 = C_3 = 370\text{pf}$ . The simulation results agree quite well with the theoretical analysis. The difference in the high frequency region of the high pass response stems primarily from the nonzero value of the the  $R_{X3}$  resistance. Next, the frequency tuning aspect of the circuit is verified for a constant  $Q=1$  value for the band pass response. The bias currents  $I_{B1}$  and  $I_{B2}$  are varied simultaneously, by keeping its ratio to be constant. The pole frequency variation , for  $Q=1$ , is shown in fig.9. The frequency is found to vary as 220Khz, 800 KHz and 1.5 Mhz for three values of  $I_{B1}=I_{B2}=6\mu\text{A}$  ,  $25\mu\text{A}$  and  $50\mu\text{A}$  respectively. Further simulations are carried out to verify the total harmonic distortion (THD). The circuit is verified by applying a sinusoidal current of varying frequency and amplitude  $40\mu\text{A}$ .The THD are measured at the  $I_{LP}$  output. The THD are found to be less than 3% while frequency is varied from 150Khz to 800Khz. Thus THD analysis of low pass output confirm the practical utility of the proposed circuit.

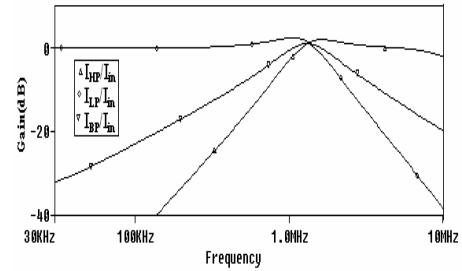


Fig.4 Gain responses of LP,HP and BP with  $C_1=C_2=C_3=370\text{pf}$ , of the circuit in Fig. 2

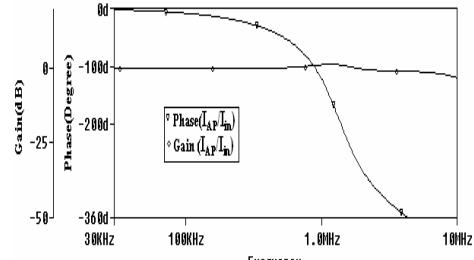


Fig.5 Gain and phase response of AP with  $C_1=C_2=C_3=370\text{pf}$ , of the circuit in Fig. 2

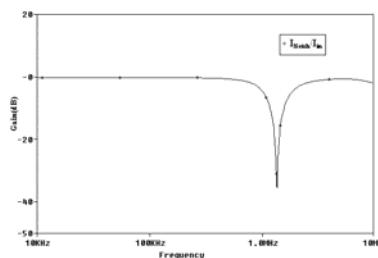
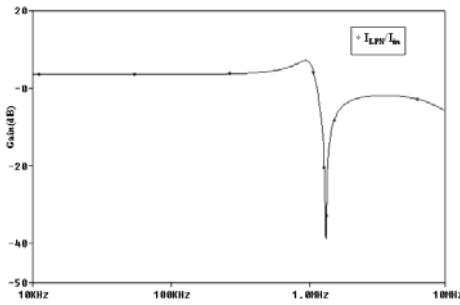
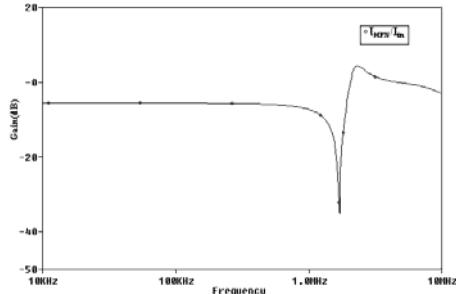


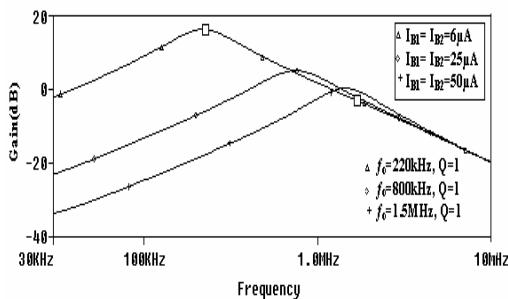
Fig.6 Gain response of regular notch with  $C_1=C_2=C_3=370\text{pf}$ , of the circuit in Fig. 2



**Fig.7** Gain response of the low pass notch with  $C_1=C_2=C_3=370\text{pf}$ , of the circuit in Fig. 2



**Fig.8** Gain response of the high pass notch with  $C_1=C_2=C_3=370\text{pf}$ , of the circuit in Fig. 2



**Fig.9** Band pass responses for different values of  $I_{B1}=I_{B2}$  of the circuit in Fig. 2

## VI. CONCLUSION

An electronically tunable current-mode universal biquad filter using only three current controlled current conveyor transconductance amplifiers (CCCCTAs) and three grounded capacitors is proposed. The proposed filter offers the following advantages

- Realizing low pass, high pass, band pass, notch and all pass simultaneously.
- Low pass and high pass notch responses can also be obtained without passive component matching conditions.
- All the capacitor being permanently grounded
- Low sensitivity figures
- The  $\omega_o$ ,  $Q$  and  $\omega_o/Q$  are electronically tunable with bias currents of CCCCTAs
- Both  $\omega_o$  and  $\omega_o/Q$ , and  $Q$  and  $\omega_o/Q$  are orthogonally tunable.
- Low pass and band pass gain can be independently tuned by biasing current of CCCCTA

## VII REFERENCES

- G. W. Roberts and A.S. Sedra, "All current-mode frequency selective circuits," Electronics Lett., vol. 25, pp. 759-761, 1989.
- B. Wilson, "Recent developments in current mode circuits," IEE Proc. G, vol. 137, pp. 63-77, 1990.
- C.M Chang, "Novel universal current-mode filter with single input and three outputs using only five current conveyors," Electronics Lett., vol .29, pp.2005-2007, 1993.
- C.M Chang, "Universal active current-mode filter with single input and three outputs using cciiis," Electronics Lett., vol.29, pp. 1932-1933, 1993.
- S. Özoguz, A. Toker, and O. Çiçekoğlu, "New current-mode universal filters using only four (CCII+)-s," Microelectronics J., vol. 30, pp. 255-258, 1999.
- A. M. Soliman, "New current-mode filters using current conveyors," Int'l J. Electronics and Communications (AEÜ), vol. 51, pp. 275-278, 1997.
- R. Senani, "New current-mode biquad filter," Int'l J. Electronics, vol. 73, 1992, pp. 735-742.
- Fabre, O. Saaid, F. Wiest, and C. Boucheron, "High frequency application based on a new current controlled conveyor," IEEE Tran. on Circuit and System.-I: Fundamental Theory and Applications, vol. 43, pp. 82-91, 1996.
- W. Kiranon, J. Kesorn, and N. Kamprasert, "Electronically tunable multi-function translinear-c filter and oscillators," Electronics Lett., vol.33, pp.573-574, 1997.
- M. T. Abuelmaatti and N. A. Tassadug, "New current-mode current controlled filters using current-controlled conveyor," Int'l J. Electronics, vol.85, pp.483-488, 1998.
- I. A. Khan and M. H. Zaidi, "Multifunction translinear-c current mode filter," Int'l J. Electronics, vol.87, pp.1047-1051, 2000.
- M. Sagbas and K. Fidaboylu, "Electronically tunable current-mode second order universal filter using minimum elements," Electronics Lett., vol.40, pp.2-4, 2004.
- T. Katoh , T. Tsukutani , Y. Sumi and Y. Fukui, "Electronically Tunable Current-Mode Universal Filter Employing CCIIIs and Grounded Capacitors," ISPACS, 2006, pp.107-110.
- S. Maheshwari and I. A. Khan, "High Performance Versatile Translinear-C Universal Filter," J. of Active and Passive Electronic Devices, vol.1, 2005, pp.41-51.
- C. Wang, H. Liu and Y. Zhao, "A new current-mode current-controlled universal filter based on CCCII(+)," Circuits System Signal Process, vol.27, pp.673-682, 2008.
- H. P. Chen and P. L. Chu, "Versatile universal electronically tunable current-mode filter using CCIIIs," IEICE Electronics Express, vol.6, pp.122-128, 2009.
- T. Tsukutani, Y. Sumi, S. Iwanari and Y. Fukui, "Novel current-mode biquad using mo-cciiis and grounded capacitors," Proceeding of 2005 International Symposium on Intelligent Signal Processing and Communication Systems, pp.433-436, 2005.
- Pandey, S. K. Paul, A. Bhattacharyya and S. B. Jain, "A Novel Current Controlled Current Mode Universal Filter :SITO Approach," IEICE Electronics Express, vol.2, pp.451-457, 2005.
- S. Minaei and S. Türköz, "Current-mode electronically tunable universal filter using only plus-type current controlled conveyors and grounded capacitors," ETRI Journal, vol. 26, pp.292-296, 2004.
- R. Senani, V. K. Singh, A. K. Singh, and D. R. Bhaskar, "Novel electronically controllable current mode universal biquad filter," IEICE electronics express, vol. 1, pp.410-415, 2004.
- S. Maheshwari and I. A. Khan, "Novel cascadable current-mode translinear-c universal filter," Active Passive Electronics Component, vol.27, pp. 215-218, 2004.
- M. Siripruchyanun and W. Jaikla, "Current controlled current conveyor transconductance amplifier (CCCCTA): a building block for analog signal processing," Proceeding of ISCIT, Sydney, Australia, pp.1072-1075, 2007.
- M. Siripruchyanun, M. Phattanasak and W. Jaikla, "Current controlled current conveyor transconductance amplifier (CCCCTA): a building block for analog signal processing," 30th Electrical Engineering Conference (EECON-30), pp. 897-900, 2007.
- D. R. Frey, "log-domain filtering: an approach to current-mode filtering", IEE Proceedings-G: Circuits, Devices and Systems, vol.140, pp.406-416, 1993.

# High Output Impedance CM-APS with Minimum Component Count

<sup>1</sup> Jitendra Mohan, <sup>2</sup>Sudhanshu Maheshwari, <sup>3</sup>Sajai Vir Singh, <sup>4</sup>Durg Singh Chauhan

<sup>1,3</sup>Jaypee University of Information Technology, Waknaghat, Solan-173215 (India)

<sup>2</sup>Z. H. College of Engineering and Technology, Aligarh Muslim University, Aligarh-202002 (India)

<sup>4</sup>Institute of Technology, Banaras Hindu University, Varanasi-221005 (India)

jitendramv2000@gmail.com, Sudhnashu\_maheshwari@rediffmail.com, sajajvir@rediffmail.com

jitendramv2000@gmail.com, Sudhnashu\_maheshwari@rediffmail.com, sajajvir@rediffmail.com, pdschauhan@gmail.com

**Abstract-** This paper presents a new current mode all pass section with minimum component count. The circuit employs one active element and two grounded passive element, ideal for IC implementation. The new circuit is ideal for current-mode cascading by possessing high output impedance. The theory is validated through PSPICE simulation using TSMC 0.35μm CMOS process parameters.

**Keywords**—Active filters, all-pass filter, current-mode, current conveyor.

## I. INTRODUCTION

Current-mode circuits have become quite popular for their potential advantages over the voltage-mode counterparts [1-3]. Current-mode filters with high output impedance offer easy cascading and are quite desirable to realizing higher order filters as well.

First order all-pass filters are important analog signal processing blocks with applications in the realization of band pass filters and oscillators [ 4-5]. In the literature, several current-mode first order all-pass sections employing different types of current conveyors have been reported [6-15]. Most of the available work uses floating capacitor or necessitate additional current follower for sensing output current at high impedance [9]. Though the circuits described in [10] fall in the separate category of tunable, resistorless realizations, but enjoys high output impedance. Other more relevant works with high output impedance use matching condition by way of employing three passive components, all of which are not in grounded form [11]. A very recent work uses four passive components (not all grounded) with matching conditions but enjoys high output impedance [12]. The circuits reported in [14] require input current insertion at different nodes for realizing all-pass filters but enjoy high output impedance with minimum grounded passive components. In the same year, another circuit presented in [15] uses two active element and three passive elements but enjoy low input impedance and high output impedance with grounded components feature.

This paper presents a new first order current-mode all pass section (CM-APS), with high output impedance by employing one multiple output fully differential second generation current conveyor (MO-FDCCII) and two passive components, both in grounded form, which are suitable for IC implementation [16]. The proposed circuit is verified through computer simulations by PSPICE, the industry standard tool.

## II. CIRCUIT DESCRIPTIONS

The matrix input-output relationship of the FDCCII is

$$\begin{bmatrix} V_{X+} \\ V_{X-} \\ I_{Za1+} \\ I_{Za2+} \\ I_{Zb1-} \\ -I_{Zb2-} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} I_{X+} \\ I_{X-} \\ V_{Y1} \\ V_{Y2} \\ V_{Y3} \\ V_{Y4} \end{bmatrix} \quad (1)$$

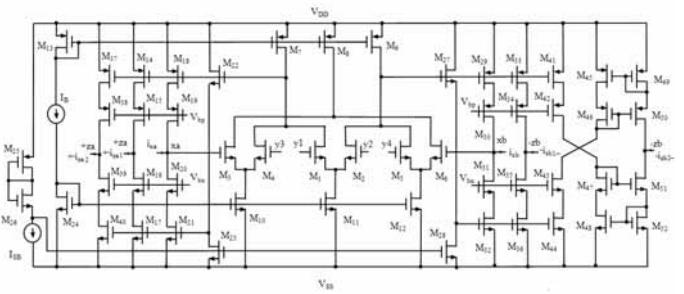


Figure 1. CMOS implementation of MO-FDCCII

CMOS implementation of MO-FDCCII [17-18] is shown in Fig.1. The proposed first order current-mode all pass section (CM-APS), is shown in Fig. 2. It is based on single MO-FDCCII and two grounded passive components i.e one capacitor and one resistor. Circuit analysis using (1) yields the following current-mode filter transfer functions:

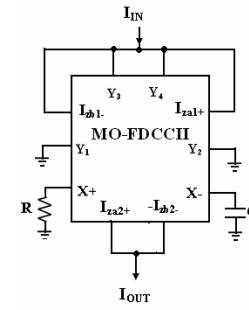


Figure 2. CM-APS

$$\frac{I_{OUT}}{I_{IN}} = -\frac{s - 1/RC}{s + 1/RC} \quad (2)$$

Equation (2) is the standard first-order all-pass transfer function. The transfer function yields a unity gain at all frequencies and a frequency dependent phase function ( $\Phi$ ) with a value  $\Phi = -2 \tan^{-1}(\omega RC)$ .

The salient feature of the proposed circuit are single active element, use of two grounded passive components and high output impedance, that enable easy cascading to successive current inputs in signal processing.

### III. NON-IDEAL ANALYSIS

To account for non ideal sources, two parameter  $\alpha$  and  $\beta$  are introduced where  $\alpha_{a1}, \alpha_{a2}, \alpha_{b1}, \alpha_{b2}$  accounts for current tracking error and  $\beta_i (i=1,2,3,4,5,6)$  accounts for voltage tracking error of the MO-FDCCII. Incorporating the two sources of error onto ideal input-output matrix relationship of the modified MO-FDCCII leads to:

$$\begin{bmatrix} V_{X+} \\ V_{X-} \\ I_{Za1+} \\ I_{Za2+} \\ I_{Zb1-} \\ -I_{Zb2-} \end{bmatrix} = \begin{bmatrix} 0 & 0 & \beta_1 & -\beta_2 & \beta_3 & 0 \\ 0 & 0 & -\beta_4 & \beta_5 & 0 & \beta_6 \\ \alpha_{a1} & 0 & 0 & 0 & 0 & 0 \\ \alpha_{a2} & 0 & 0 & 0 & 0 & 0 \\ 0 & \alpha_{b1} & 0 & 0 & 0 & 0 \\ 0 & \alpha_{b2} & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} I_{X+} \\ I_{X-} \\ V_{Y1} \\ V_{Y2} \\ V_{Y3} \\ V_{Y4} \end{bmatrix} \quad (3)$$

The CM-APS of Fig. 2 is analysed using (3) and the non-ideal current transfer function is found as

$$\frac{I_{OUT}}{I_{IN}} = -\frac{\alpha_{b2}}{\alpha_{b1}} \left( \frac{s - \beta_3 \alpha_{a2} / (\beta_6 \alpha_{b2} RC)}{s + \beta_3 \alpha_{a1} / (\beta_6 \alpha_{b1} RC)} \right) \quad (4)$$

Thus, the pole frequency ( $\omega_0$ ) and gain (H) of the CM-APS of Fig. 2 can be expressed as

$$\omega_0 = \frac{\beta_3 \alpha_{a1}}{\beta_6 \alpha_{b1} RC}, \quad H = \frac{\alpha_{b2}}{\alpha_{b1}} \quad (5)$$

From equation (5), the pole frequency ( $\omega_0$ ) and gain (H) sensitivities can be expressed as

$$S_{\beta_6, \alpha_{b1}}^{\omega_0} = -1; \quad S_{\beta_3, \alpha_{a1}}^{\omega_0} = 1; \quad S_{\alpha_{b1}}^H = -1; \quad S_{\alpha_{b2}}^H = 1 \quad (6)$$

Thus, from equation (6), the pole frequency ( $\omega_0$ ) and gain (H) sensitivities to the active and passive components are within unity in magnitude. Thus, CM-APS circuit enjoy attractive active and passive sensitivity performance.

### IV. SIMULATION RESULTS

The proposed CM-APS has been simulated using PSPICE. The MO-FDCCII was realized using CMOS implementation as shown in Fig. 1. and simulated using TSMC 0.35 $\mu$ m, Level 3 MOSFET parameters . The aspect ratio of the MOS transistors were shown as in Table 1 and with the following DC biasing levels  $V_{dd}=3V$ ,  $V_{ss}=-3V$ ,  $V_{bp}=V_{bn}=0V$ , and  $I_B=I_{SB}=1.2mA$ . The circuit was designed with  $C=1 nF$ , and  $R=1 k\Omega$ . The designed pole frequency was 159.4 KHz. The phase is found to vary with frequency from 0 to -180° with a value of -90° at the pole frequency as shown in Fig. 3, and the pole frequency was found to be 160.4 KHz, which is approximately equal to the designed value. The circuit was next used as a phase shifter introducing 90° shift to a sinusoidal voltage input of 1 mA peak at 160.4 KHz was applied. The input and 90° phase shifted output waveforms (given in Fig. 4) which verify circuit as a phase shifter. The THD variation at the output for varying signal amplitude at 160.4 KHz was also studied and the results

shown in Fig. 5. The THD for a wide signal amplitude (few  $\mu$ A- 1000 $\mu$ A) variation is found within 2.5% at 160.4 KHz. The Fourier spectrum of the output signal, showing a high selectivity for the applied signal frequency (160.4 KHz) is also shown in Fig. 6.

Table 1. Transistor aspect ratios for the circuit shown in Fig. 1

Transistors	W( $\mu$ m)	L( $\mu$ m)
M1-M6	60	4.8
M7-M9, M13	480	4.8
M10-M12, M24	120	4.8
M14,M15,M18,M19,M25,M29,M30,M33,M 34,M37,M38,M41,M42,M45,M46,M49,M50	240	2.4
M16,M17,M20,M21,M26,M31,M32,M35,M 36,M39,M40,M43,M44,M47,M48,M51,M52	60	2.4
M22,M23,M27,M28	4.8	4.8

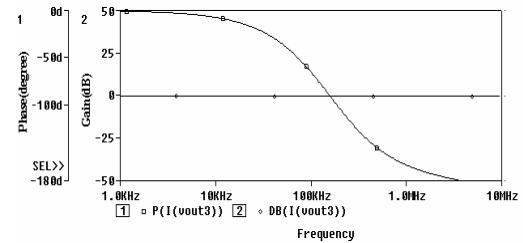


Figure 3. Simulated gain and phase response for CM-APS

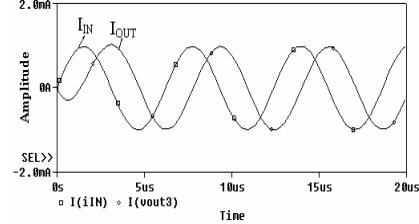


Figure 4. Time-domain waveforms of the CM-APS for input frequency 160.4 KHz

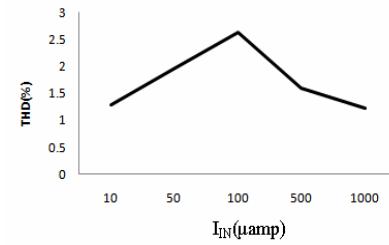


Figure 5. THD variation at output with signal amplitude at 160.4 KHz

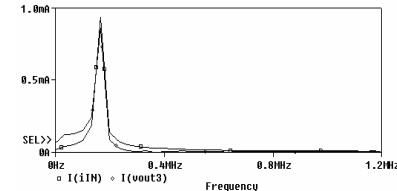


Figure 6. Fourier spectrum of output signal of 160.4 KHz

## V. CONCLUSIONS

This paper presents a new current mode all-pass filter, employing one MO-FDCCII and two passive components. The salient features of the proposed circuit are high output impedance, single active element and use of minimum as well as grounded components. The proposed circuit is verified through PSPICE simulation using TSMC 0.35μm CMOS process parameter.

## VI REFERENCES

- [1]. G.W. Roberts, and A.S. Sedra, "All current-mode frequency selective circuits," Electronic Letter, vol. 25, pp. 759–761, 1989.
- [2]. C. Tomazou, F.J. Lidgey, and D.G. Haigh, "Analogue IC design: the current-mode approach", IEE, UK, 1998.
- [3]. B. Wilson, "Recent developments in current conveyors and current-mode circuits," IEE Proc. G Circuits, Devices System, vol. 132, pp. 63–76, 1990.
- [4]. D.T. Comer, D.J. Comer, and J.R. Gonzalez, "A high frequency integrable band-pass filter configuration," IEEE Trans. Circuits Syst. II, vol. 44, pp. 856–860, 1997.
- [5]. S.J.G. Gift, "The applications of all-pass filters in the design of multiphase sinusoidal systems," Microelectronics Journal, vol. 31, pp. 9–13, 2000.
- [6]. Fabre, and J.P. Longuemard, "High performance current processing all-pass filters," International Journal of Electronics, vol. 66, pp. 619–632, 1989.
- [7]. M. Higashimura, "Current mode all-pass filter using FTFN with grounded capacitors," Electronic Letter, vol. 27, pp. 1182–1183, 1991.
- [8]. A.M. Soliman, "Theorems relating to port interchange in current-mode CCII circuits," International Journal of Electronics, vol. 82, pp. 584–604, 1997.
- [9]. S. Maheshwari, and I.A. Khan, "Novel first order all-pass sections using a single CCIII," International Journal of Electronics, vol. 88, pp. 773–778, 2001.
- [10]. S. Maheshwari, "A new current-mode current controlled all-pass section," Journal of Circuits Systems Computer, vol. 16, pp. 181–189, 2007.
- [11]. S. Minaei, and M.A. Ibrahim, "General configuration for realizing current-mode first order all-pass filter using DVCC," International Journal of Electronics, vol. 92, pp. 347–356, 2005.
- [12]. J.W. Horng, C.-L. Hou, C.M. Chang, W.Y. Chung, H.L. Liu, and Lin C.T., "High output impedance current mode first order all-pass networks with four grounded passive components and two CCIIIs," International Journal of Electronics, vol. 93, pp. 613–621, 2006.
- [13]. Toker, O. Ozoguz, O. Cicekoglu, and C. Acar, "Current mode all-pass filter using CDBA and a new high Q bandpass filter configuration," IEEE Trans. Circuits Syst. II, vol. 47, pp. 949–954, 2000.
- [14]. S. Maheshwari, "Novel cascadable current-mode first order all-pass sections," International Journal of Electronics, vol. 94, pp. 995-1003, 2007.
- [15]. B. Metin, K.Pal and O. Cicekoglu, "All-pass filter for rich cascability options easy IC implementation and tunability", International Journal of Electronics, 2007, 94, pp.1037-1045.
- [16]. M. Bhushan, and R. W. Newcomb, "Grounding of capacitors in integrated circuits," Electronic Letter, vol. 3, pp. 148–149, 1967.
- [17]. A. el-Adway, A. M. Soliman and H. O. Elwan, "A novel fully differential currenty conveyor and its application for analog VLSI," IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process, vol. 47, pp. 306-313, 2000.
- [18]. J.W. Horng, C.L. Hou, C.M. Chang, H.P. Chou, C.T. Lin, and Y. Wen, "Quadrature Oscillators with Grounded Capacitors and Resistors Using FDCCIIIs," ETRI Journal., vol. 28, pp. 486-494, 2006.

# Design Modeling and Simulation of RF MEMS Shunt and Series Switches

Chirag Ahuja, Navneet Tohan and Akhil Sharma

**Abstract—**This paper is focused on the analytical design, mechanical and electromechanical simulations for different structures of the RF MEMS based shunt switch and modeling for different configurations of the MEMS based series switch, which operate at microwave frequencies. The structures for MEMS based cantilever and fixed beam type series and shunt switches respectively, are first characterized using a full wave analysis based on finite element method aiming to extract the S-parameters of the switches for different geometrical configurations. From the S-parameter data base, a scalable lumped circuit model is extracted to allow validation of the switch model through the available microwave circuit simulator. The simulated results are compared with published measured data as validation of our model.

**Index Terms**—RF, MEMS switch, electromechanical, finite element analysis

## I. INTRODUCTION

### A. About The Technology: MEMS

With the ever increasing demand for smaller and more reliable systems, attention has been focused for some time on exploiting the advances in silicon processing to fabricate some of the commonly used components on silicon. The concept of fabricating micro scale devices that can function like the ones made using conventional technology has been of interest for some time in the semiconductor technology. It is only in the recent past that this has gained momentum due to the strides made in silicon fabrication technology. Micro electro-mechanical systems are most popularly known as MEMS [1]. MEMS devices are designed and fabricated by techniques similar to those of very large-scale integration, and can be manufactured by traditional batch-processing methods [8]. MEMS based devices are at present finding a huge range of applications. These include accelerometers, pressure sensors and optical switches. Recently the work has been done in the field of RF MEMS switches that are being used to switch power between the transmitter and receiver or in antenna arrays to form configurable array of antennas.

### B. RF MEMS

For the first time in recent history, a technology is emerging that promises to enable both new paradigms in RF circuits and systems topologies and architectures as well as unprecedented levels of performance and economy [7]. RFMEMS is widely believed to be such a technology. The evolving nature of MEMS has evolved RF and millimeter-wave MEMS from the sensors based technology. RF MEMS [2] use the same fabrication techniques and have become an area of significant interest. The most critical issue in the development of RF MEMS technology is the fabrication and release of suspended structure. The main building block of any RF application is the RF MEMS switch as it is extensively

used in almost every RF MEMS component like phase shifter, switching circuits, varactor, tunable filter etc.

### C. RF MEMS Switches

RF MEMS switches are the specific micromechanical switches that are designed to operate at RF-to-millimeter wave frequencies (0.1 to 100 GHz). The drive for MEMS switches for RF applications has been mainly due to the highly linear characteristics of the switch over a wide range of frequencies. The main advantages of MEMS based switches are mainly low loss, high isolation, and low distortion of signal, low power consumption and size reduction [9]. The high performance is due to the very low capacitance and contact resistance, which can be achieved using RF MEMS as compared to GaAs PIN diodes or FETs. RF MEMS switches have superior insertion loss and isolation performance compared to MESFET and p-i-n diode switches [12]. It is possible to build RF MEMS switches with a figure-of-merit cut-off frequency of 30-80 THz, which is about 100 times better than GaAs transistors.

MEMS switches are devices that use mechanical movement to achieve a short circuit or an open circuit in the RF transmission line. The forces required for the mechanical movement can be obtained using electrostatic, magneto-static, piezoelectric, or thermal designs. To date, only electrostatic type switches have been demonstrated at 0.1 – 100 GHz with high reliability.

There are two basic switches used in RF to mm. wave design: - the shunt switch and the series switch.

- The **shunt switch** is placed in shunt between the t – line and ground and depending on the applied bias voltage (Figure 1); it either leaves the t- line undisturbed or connects it to ground. Therefore the ideal shunt switch results in zero insertion loss when no bias is applied (up state) and infinite isolation when bias is applied (down state position). Shunt capacitive switches are more suited for higher frequencies (5 – 100 GHz) [7]. The switch length is 300 $\mu$ m and the anchors are 40 $\mu$ m from the CPW ground plane edge. Silicon Substrate thickness=270 $\mu$ m, CPW length = 800 $\mu$ m, Gap/Width/Gap = 60/100/60  $\mu$ m, Thickness of the CPW line =1.5 $\mu$ m, Thin dielectric ( $\text{SiN}_4$ ) = 0.1  $\mu$ m, Air gap = 2 $\mu$ m Beam material =Gold (Au), Beam width = 80 $\mu$ m, Beam length = 300 $\mu$ m, Beam Thickness =1 $\mu$ m. A well-designed shunt capacitive switches switch results in:
- Low insertion loss (- 0.04 to 0.1 dB at 5 – 50 GHz) in the upstate position.
- Acceptable isolation more than – 20 dB at 10 – 50 GHz in the down state position.

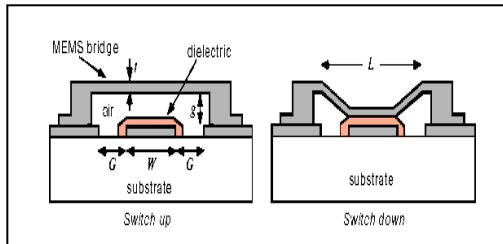


Fig. 1. RF MEMS switch up and down state

- The **ideal series switch** results in an open circuit in the t – line when no bias is applied (up state position) (Fig2) and it results in a short circuit in the t – line when a bias voltage is applied (down state position). Ideal series switch have infinite isolation in the up – state position and have zero insertion loss in the down state position. MEMS series switches are used extensively used for 0.1 – 40 GHz applications [6]. They offer high isolation at RF frequencies, around – 50 dB to – 60 dB at 1 GHz and MEMS series switches are used extensively used for 0.1 – 40 GHz applications. They offer high isolation at RF frequencies, around – 50 dB to – 60 dB at 1 GHz and rising to – 20 to – 30 dB at 20 GHz. In the down state position, they result in very low insertion loss, around – 0.1 to 40 GHz

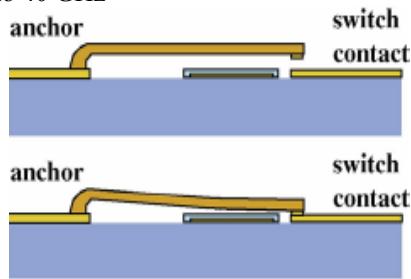


Fig. 2. RF MEMS series switch upstate and downstate

## II. DESIGN PARAMETERS

### a. Insertion Loss

Insertion loss [3] is a measure of how much loss is being introduced into the system due to the switch. Introduction of the additional circuitry to the system leads to losses. Factors contributing to loss in the switch are contact resistance and resistance of the waveguide. Contact resistance is the function of the contact force that is generated when the switch is turned ON and when in turn is the function of the actuation voltage, stiffness of the beam and contact spacing. The lower the insertion loss the better it is. The insertion loss is calculated as **20 log (V<sub>out</sub> / V<sub>in</sub>)**. Typical insertion losses that have been reported are in the range of -0.1dB at 40 GHz.

### b. Power Handling

The power handling capacity of the switch is mainly determined by the properties of transmission line. We can compute the dimensions of the transmission line that can handle required amount of the power.

### c. Isolation

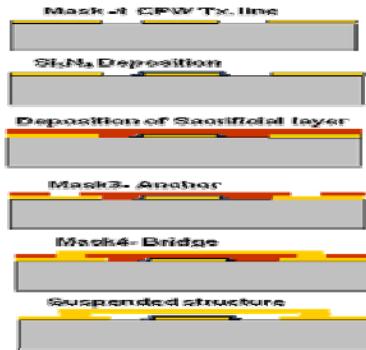
Isolation [3] determines how well the output is isolated from the input higher isolation translates to less coupling between the output and the input ports. Isolation is dependent on the spacing between the input ports and the output ports or the OFF state capacitance of the switch which can be improved by moving the contact arm as far away from the waveguide as possible when the switch is OFF. Typical values of isolation come out to be -15dB.

### d. Actuation Voltage

Actuation voltage [4] is applied to the actuation electrodes of the switch to bring about a change in the state of the switch. Typical actuation voltages of switches range 20-80 volts. This is an important factor in the design of the switch as this alone decides the compatibility of the switch with other circuitry on the die. In some cases, though the actuation voltages are high voltage scaling circuits is used to make the switch compatible with the interface circuitry. The main factors influencing the magnitude of the actuation voltage are electrode spacing, area of the electrode and dielectric material between the two electrodes. The formula has been derived later in the section. Actuation voltage in this switch comes out to be 20V.

## III. PROCESSING AND FABRICATION

Process compatibility has become a major issue as more and more MEMS technology is being integrated to the systems. We have used gold as a metal in our switch for our convenience. Gold is the material used for coplanar wave guide (CPW) and the lower electrode of the switch [10]. Gold is deposited via electroplating and patterned to form CPW line dielectric/insulator between the lower and the upper electrode (Fig 5). A thick layer of PECVD oxide is deposited (Fig 6). This layer is then patterned to form the anchors of the structure of the membrane (Fig 7). Then again metallization is carried out for the anchors and the membrane so that gold contacts are made (Fig 8). Patterning is done for the membrane and the structure is removed by removing the sacrificial oxide layer that has been deposited. The released structure is shown in Figure 9.



## IV. OPERATION OF THE SWITCH

When a voltage is applied between a fixed – fixed or cantilever beam [3] and the pull down electrode, an electrostatic force is induced on the beam. This is well known electrostatic force which exists on the plates of a capacitor under an applied voltage. In order to approximate this force the beam over the

pull down electrode is modeled as a parallel plate capacitor. Although the actual capacitance is about 20 – 40% larger due to fringing fields. The model provides a good understanding of how electrostatic actuation works. Given that the width of the beam is ‘w’ and the width of the pull down electrode is ‘W’ ( $A = w \times W$ ), the parallel plate capacitance is

$$C = \epsilon_0 A / g = (\epsilon_0 w W) / g \quad (1)$$

Where ‘g’ in equation (1), is the height of the beam above the electrode. The electrostatic force is applied to the beam is found by considering the power delivered to a time dependent capacitance and is given by

$$F_e = \frac{1}{2} V_p^2 dC(g) / dg = -\frac{1}{2} (\epsilon_0 w W V_p^2) / g^2 \quad (2)$$

Where  $V_p$  is the voltage applied between the beam and the electrode. Notice that the force is independent of the voltage polarity. Equation (2) neglects the effects of dielectric layer between the bridge and the pull down electrode.

The electrostatic force is approximated as being evenly distributed across the section of the beam above the electrode. Therefore the appropriate spring constant must be associated with the distance moved under the location of the applied force. But the spring constant is associated with the displacement at the center of the beam and not under the location of the applied force. Equating the applied electrostatic force with the mechanical restoring force due to stiffness of the beam ( $F = kx$ ), we find

$$-\frac{1}{2} (\epsilon_0 w W V_p^2) / g^2 = k (g_0 - g) \quad (3)$$

Where  $g_0$  is zero bias bridge height. Solving this eqn. for the voltage results in

$$V_p = [(2k / \epsilon_0 w W) g^2 (g_0 - g)]^{1/2} \quad (4)$$

The plot of the beam height vs. applied voltage shows that the position of the beam becomes unstable at  $(2/3 g_0)$ , which is due to the positive feedback in the electrostatic actuation. At  $(2/3 g_0)$  the increase in the electrostatic force is greater than the increase in the restoring force, resulting in (a) the beam position become unstable and (b) collapse of the beam to the down state position. By taking the derivative of eqn. (4) with respect to the beam height and setting it to zero, the height at which the instability occurs is found to be exactly  $2/3$  the zero bias beam height. Substituting this value back into the pull down is found to be

$$V = [(8k g_0^3 / 27\epsilon_0 w W)]^{1/2} \quad (5)$$

It should be noted that equation (5) shows a dependence on the beam width,  $w$  the pull down voltage is independent of the beam width since the spring constant  $k$  varies linearly with  $w$ . Fig 10 shows the graphical variation between the pull down voltage and displacement.

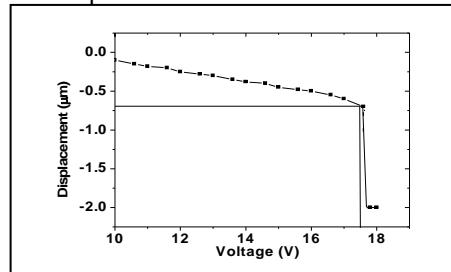


Fig. 10. The membrane starts bending towards the dielectric or the lower electrode when the pull down voltage is increased and at voltage of around 17 Volts it results into the down state of the switch.

#### Actuation of the shunt switch

It consists of a thin metal membrane “bridge” suspended over the center conductor of a coplanar waveguide and fixed at both ends to the ground conductors of the CPW line. A 1000–2000-Å dielectric layer is used to dc isolate the switch from the CPW center conductor. When the switch (membrane) is up, the switch presents a small shunt capacitance to ground. When the switch is pulled down to the center conductor, the shunt capacitance increases by a factor of 20–100. The actuation mechanism is shown in Figure 11

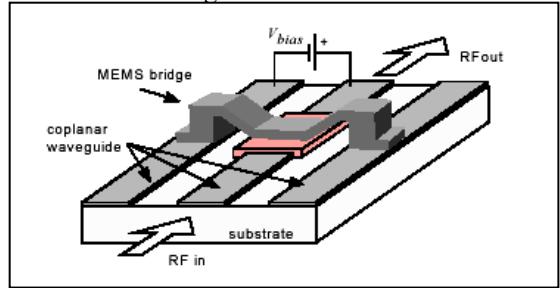


Fig. 11. Actuation mechanism in RFMEMS shunt switch.

#### A. Actuation of the series switch

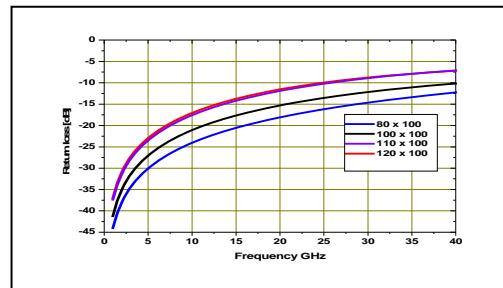
In this switch the contacting surface is usually at the end of the cantilever beam that is supported from a single side with a control electrode under the beam. This control electrode is called as actuation electrode and is needed to mechanically actuate the switch. A gap (open circuit) is created in the microwave t-line when the switch is in the upstate position resulting in high isolation.

By applying a voltage to the control electrode the beam can be pulled down to complete the connection between two conductors. When the bias voltage is removed, the MEMS switch returns back to its original position due to the internal restoring forces of the cantilever.

## V. SIMULATIONS

#### A. Introduction

FEA (Finite element analysis) consists of a computer model of a material or design that is loaded and analyzed for specific results. Mathematically, the structure to be analyzed is subdivided into mesh of finite sized elements of simple shape. This electromagnetic analysis is carried out by simulating [S]-parameters of the switch using full-wave analysis HFSS software. The device has been simulated for the return loss in the up-state position and the isolation in the down-state position of the switch using HFSS. The plots of the return loss and isolation of the shunt switch are given in Fig 12 & Fig 13.



The isolation at upstate is found to be -23dB at 20 GHz and the insertion loss is found to be -0.02dB at 20 GHz. The respective curves are shown in Figure 14 & Figure 15.

Fig. 12. Return loss of the shunt

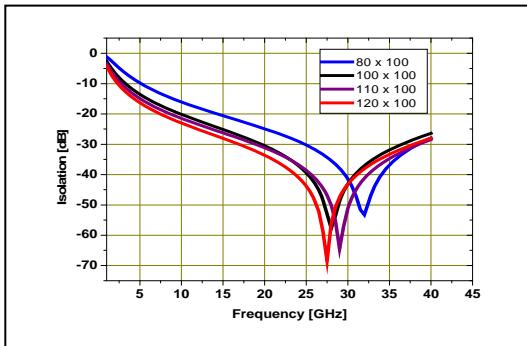


Fig. 13. Isolation of the shunt switch

#### B. Values of electrical parameters

Values of upstate capacitance, downstate capacitance and inductance have been calculated using formulas in equation (6) involving the simulated S-parameters values [11].

$$|S_{11}| < -10 \text{ dB} \text{ or } wC_u Z_o \ll 2$$

$$S_{11}^2 = \frac{w^2 C_u^2 Z_o^2}{4}$$

$$|S_{11}|^2 = \begin{cases} \frac{4}{w^2 C_u^2 Z_o^2} & \text{for } f \ll f_0 \\ \frac{4R_s^2}{Z_o^2} & \text{for } f = f_0 \\ \frac{4w^2 C_u^2}{Z_o^2} & \text{for } f \gg f_0 \end{cases}$$

$$f_0 = \frac{1}{2\pi\sqrt{LC_u}}$$

(6)

#### C. Simulation of the series switch

The series switch simulations are carried out for two configurations: Inline and Broad line configurations. The various physical parameters during simulation are kept to be as follows.

- Dimensions of the substrate: 500 X 220 X 100  $\mu\text{m}$ , Gap b/w the signal line: 60  $\mu\text{m}$
- Signal line width: 100 tapered to 60  $\mu\text{m}$
- Bridge length: 80  $\mu\text{m}$
- Bridge width: 30  $\mu\text{m}$
- Height of the bridge from the signal line: 3  $\mu\text{m}$
- Overlapping area: 20 \* 20  $\mu\text{m}^2$ .

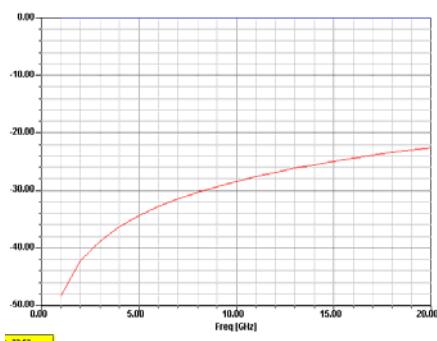


Fig. 14. Isolation of the series switch

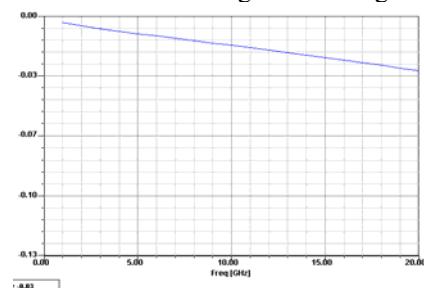


Fig. 15. Insertion loss of the series switch

## VI. CONCLUSION

The RF MEMS series switch has been designed for maximum isolation and minimum insertion loss. The simulation has been carried out for varying geometries of the switch and the effect of beam widths and length has been studied on the design parameters like insertion loss, isolation. Various problems of stiction and planarization have been encountered but steps are being carried out to remove these. So, the fabrication of the shunt switch is near completion. The mechanical analysis of the series switch and its fabrication is under investigation and will be covered in the near future.

## VII. ACKNOWLEDGMENT

The authors would like to extend his heartfelt thanks to his instructor **Dr. Renu Sharma Scientist 'D'**, for guiding him all through and providing him with all the possible resources regarding RF MEMS technology and for helping him to get acquainted with the software available and the latest development in the field of MEMS. The author is also grateful to all the members of the group of **silicon MEMS division SSPL, Delhi** for their co-operation and guidance.

## VIII. REFERENCES

- [1]. Foundation of MEMS by Chang Liu, Pearson International Publications.
- [2]. Theory, design and technology of RF MEMS, G. M. Rebeiz and J. B. Muldavin
- [3]. F. D. Flavii and R. Cocciali, "Combined mechanical and electrical analysis of microelectro mechanical switch for RF applications," presented at European Microwave Conference Germany.
- [4]. J. B. Muldavin and G. M. Rebeiz, "30 GHz Tuned Mems Switched," *IEEE MTT Symposium*, 1998.
- [5]. N.V.Lakamraju, Stephen M Philips "Bistable RF MEMS with low actuation voltage," proceedings of SPIE MEMS. VOL.54
- [6]. S.M. Sze, *Semiconductor Sensors*, John Wiley and Sons, INC 1994
- [7]. Hector J. De Los Santos, "RFMEMS Circuit Design", British Library Cataloguing in Publication Data, USA, 2002.
- [8]. Elliot R. Brown, "RF-MEMS Switches for Reconfigurable Integrated Circuits", *IEEE Trans. On Microwave Theory and Techniques*, Vol. 46, no.11, November 1998
- [9]. V. K. Varadan, Kalarickaparambil Joseph Vinoy, K. A. Jos, "RF MEMS and their applications", Wiley Publications, 2004.
- [10]. Nava Setter, "Electroceramic-based MEMS", Springer Publications, USA, 1999
- [11]. Jan G. Korvink, Oliver Paul, "MEMS- A practical guide to design, analysis and applications", Printed-Hall, 2005.

# Embedded System Architecture Design Based on Real-Time Emulation

Lecturers- Shailza kamal, Amritpal kaur, Navdeep choudhary  
Baba Hira Singh Bhattal Inst. Of Engg & Techology, Lehragagam, Sunam 148031  
sk\_rssb@yahoo.com, nancydhillon\_2007@yahoo.co.in, navdeepchoudhary@gmail.com

**Abstract-** This paper presents a new approach to the design of embedded systems. Due to restrictions that state-of the art methodologies contain for hardware/software partitioning, we have developed an emulation based method using the facilities of reconfigurable hardware components, like Field Programmable Gate Arrays (FPGA). We create a embedded system real time application for specific hardware part which interact with environment and on other side specific software runs on microcontroller ,aim is to define how embedded system architecture works function of one system with different system using RT emulator.

**Keywords :** System Emulation, Prototyping of Real-Time Systems, The Role of FPGAs in System Prototyping,

## I. INTRODUCTION

An embedded system is a special-purpose computer system designed to perform one or a few dedicated functions,<sup>[1]</sup> often with real-time computing constraints. It is usually *embedded* as part of a complete device including hardware and mechanical parts. In contrast, a general-purpose computer, such as a personal computer, can do many different tasks depending on programming. Embedded systems control many of the common devices in use today. Since the embedded system is dedicated to specific tasks, design engineers can optimize it, reducing the size and cost of the product, or increasing the reliability and performance. Some embedded systems are mass-produced, benefiting from Physically, embedded systems range from portable devices such as digital watches and MP4 players, to large stationary installations like traffic lights, factory controllers, or the systems controlling nuclear power plants. Complexity varies from low, with a single *microcontroller* chip, to very high with multiple units, peripherals and networks mounted inside a large chassis or enclosure. In general, "embedded system" is not an exactly defined term, as many systems have some element of programmability. For example, Handheld computers share some elements with embedded systems — such as the operating systems and microprocessors which power them — but are not truly embedded systems, because they allow different applications to be loaded and peripherals to be connected.

### Examples of embedded systems

PC Engines' ALIX.1C Mini-ITX embedded board with an x86 AMD Geode LX 800 together with Compact Flash, miniPCI and PCI slots, 22-pin IDE interface, audio, USB and 256MB RAM.

An embedded RouterBoard 112 with U.FL-RSMA pigtail and R52 miniPCI Wi-Fi card widely used by wireless Internet service providers (WISPs) in the Czech Republic. Embedded

systems span all aspects of modern life and there are many examples of their use. Telecommunications systems employ numerous embedded systems from telephone switches for the network to mobile phones at the end-user. Computer networking uses dedicated routers and network bridge to route data.

Consumer electronics include personal digital assistants (PDAs), mp3 players, mobile phones, videogame consoles, digital cameras, DVD players, GPS receivers, and printers. Many household appliances, such as microwave ovens, washing machines and dishwashers, are including embedded systems to provide flexibility, efficiency and features. Advanced HVAC systems use networked thermostats to more accurately and efficiently control temperature that can change by time of day and season. Home automation uses wired- and wireless-networking that can be used to control lights, climate, security, audio/visual, surveillance, etc., all of which use embedded devices for sensing and controlling.

Transportation systems from flight to automobiles increasingly use embedded systems. New airplanes contain advanced avionics such as *inertial guidance systems* and *GPS* receivers that also have considerable safety requirements. Various electric motors — *brushless DC motors*, *induction motors* and *DC motors* — are using electric/electronic *motor controllers*. *Automobiles*, *electric vehicles*, and *hybrid vehicles* are increasingly using embedded systems to maximize efficiency and reduce pollution. Other automotive safety systems such as *anti-lock braking system* (ABS), *Electronic Stability Control* (ESC/ESP), *traction control* (TCS) and automatic *four-wheel drive*. *Medical equipment* is continuing to advance with more embedded systems for *vital signs monitoring*, *electronic stethoscopes* for amplifying sounds, and various *medical imaging* (PET, SPECT, CT, MRI) for non-invasive internal inspections.

In addition to commonly described embedded systems based on small computers, a new class of miniature wireless devices called *motes* are quickly gaining popularity as the field of wireless sensor networking rises. Wireless sensor networking, *WSN*, makes use of miniaturization made possible by advanced IC design to couple full wireless subsystems to sophisticated sensor, enabling people and companies to measure a myriad of things in the physical world and act on this information through IT monitoring and control systems. These motes are completely self contained, and will typically run off a battery source for many years before the batteries need to be changed or charged.

## II. EMBEDDED SYSTEM EMULATION

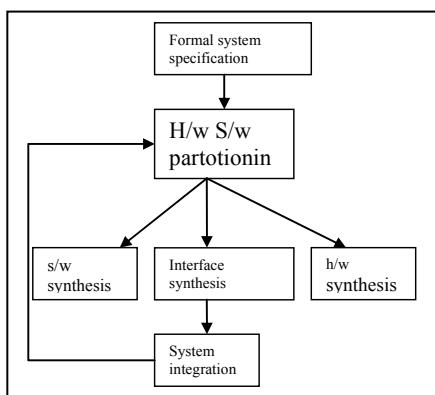
Most of today's existing technical applications are controlled by so-called embedded systems<sup>1</sup>. Many different

application areas which demands their own specific embedded system architecture exist. Therefore, a common definition of embedded systems cannot find wide acceptance.[2] In this domain, an embedded system architecture consists of an application-specific hardware part, which interacts with the environment. At the same time, an application specific software part runs on a microcontroller. In the last few years, rapid progress in microelectronic technology has reduced component costs, while simultaneously increasing the complexity of microcontrollers and application specific hardware. Nevertheless, developers of embedded systems have to design low cost, high performance systems and reduce the timeto- market to a minimum. The most important taste a specification must complete is the partitioning of the system into 2 parts. The first part is the software which runs on a microcontroller. Powerful on-chip features, like data and instruction caches, programmable bus interfaces and higher clock frequencies, speed up performance significantly and simplify system design. These hardware fundamentals allow Real-time Operating Systems (RTOS) to be implemented, which leads to the rapid increase of total system performance and functional complexity. Nevertheless, if fast reaction times must be guaranteed, the software overhead due to task switching becomes a limiting performance factor and application-specific hardware must be implemented. This can be done by developing ASICs. Due to the decreasing life cycles of many high-end electronic products, there is a gap between the enormous development costs and limited reuse of an ASIC. In the last few years, so-called IP Core components became more and more popular. They offer the possibility of reusing hardware components in the same way as software libraries. In order to create such IP-Core components, the system designer uses Field Programmable Gate Arrays instead of ASICs. The designer still must partition the system design into a hardware specific part and a microcontroller based part.

**2 State of the Art** Basically two major design methodologies for embedded systems exist.

**2.1 Hardware First Approach** The most commonly applied methodology in industry is based on a sequential design flow. This design-oriented approach is shown

Figure 1:

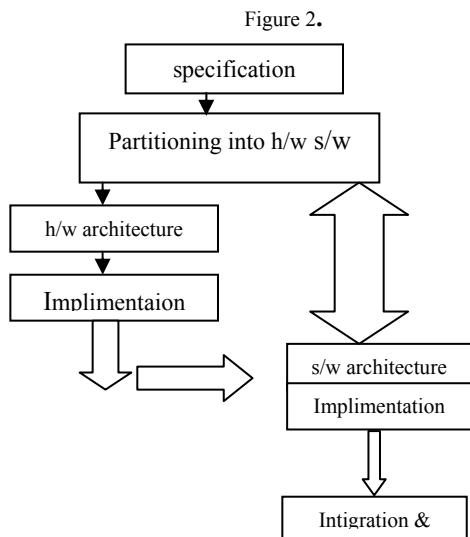


**Design-Oriented Sequential Design Flow** The first step (milestone 1) of this approach is the specification of the embedded system, regarding functionality, power consumption,

costs, etc. After completing this specification, a step called „partitioning“ follows. The design will be separated into two parts: • A hardware part, that deals with the functionality implemented in hardware add-on components like ASICs or IP cores. • A software part, that deals with code running on a microcontroller, running alone or together with an real-time-operating system (RTOS) The second step is mostly based on the experience and intuition of the system designer. After completing this step, the complete hardware architecture will be designed and implemented (milestones 3 and 4). After the target hardware is available, the software partitioning can be implemented. The last step of this *sequential* methodology is the testing of the complete system, that means the evaluation of the behavior of all the hardware and software components. Unfortunately developers can only verify the correctness of their hardware/software partitioning in this late development phase. If there are any uncorrectable errors, the design flow must restart from the beginning, which can result in enormous costs. For this reason, developers often use „well-known“ components rather than new available circuits. They want to reduce the risk of design faults and to reuse existing know-how. This is especially important for the design of systems consisting of few, but highly complex components. Another disadvantage of this approach is that it is not pos- 1 This work was supported in part with funds from the Deutsche Forschungsgemeinschaft under reference number 3221040 within the priority program “Design and Design Methodology of Embedded Systems”. sible to start software development before the design and test of the hardware architecture has finished. Software developers have to wait until a bug-free hardware architecture is available. This time (and cost) intensive delay is graphically displayed in Figure 1 between milestone two and four. Once again, the disadvantages of this methodology are: complete redesign in case of design faults, reduced degrees of freedom in selection of components (due to reuse of knowledge and experiences) and time delays. Nonetheless, the hardwarefirst approach is still a valuable approach to system design with low or medium complexity, because the initial step of partitioning is less time-consuming than in other approaches. For high-end embedded systems new methods are needed to recognize errors during an early phase of the design process.

## 2.2 Hardware / Software Co-Design

The first step in this approach focuses on a formal specification of a system design as shown in Figure 2. This specification does not focus on concrete hardware or software architectures, like special microcontrollers or IP-cores. Using several of the methods from mathematics and computer sciences, like petri-nets, data flow graphs, state machines and parallel programming languages; this methodology tries to build a complete description of the system’s behavior. The result is a decomposition of the system’s functional behavior, it takes the form of a set of components which implements parts of the global functionality. Due to the use of formal description methods, it is possible to find different alternatives to the implementation of these components.



### Hardware / Software Co-Design

The next step is a process called hardware/software partitioning. The functional components found in step one can be implemented either in hardware or in software. The goal of the partitioning process is an evaluation of these hardware/software alternatives. Depending on the properties of the functional parts, like time complexity of algorithms, the partitioning process tries to find the best of these alternatives. This evaluation process is based on different conditions, such as metric functions like complexity or the costs of implementation. After a set of best alternatives is found, the next step is the implementation of the components. In Figure 2, these implementations are shown as hardware synthesis, software synthesis and interface synthesis. Hardware components can be implemented in languages like VHDL, software is coded using programming languages like Java, C or C++. The last step is system integration. System integration puts all hardware and software components together and evaluates if this composition complies with the system specification, done in step one. If not, the hardware/software partitioning process starts again. An essential goal of today's research is to find and optimize algorithms for the evaluation of a partitioning. Using these algorithms, it is theoretically possible to implement hardware / software co-design as an automated process. Due to the algorithm-based concept of hardware/software co-design there are many advantages to this approach. The system design can be verified and modified at an early stage in the design flow process. Nevertheless, there are some basic restrictions which apply to the use of this methodology:

- **Insufficient knowledge:** As described in this section, hardware/ software codesign is based on the formal description of the system and a decomposition of its functionality. In order to commit to real applications, the system developer has to use *available* components, like IP-cores. Using this approach, it is necessary to describe the behavior and the attributes of these components completely. Due to the blackbox nature of IP-cores, this is not possible in all cases.

- **Degrees of freedom:** Another of the building blocks of hardware/software codesign is the unrestricted substitution of hardware components by software components and vice versa. For real applications, there are only a few degrees of freedom in regards to the microcontroller, but for ASIC or IP-core components, there is a much greater degree of freedom. This is due to the fact that there are many more IPcores than microcontrollers which can be used for dedicated applications, available. Due to the limitations that have been mentioned, the hardware-software co-design approach is not suitable for some design projects, like very complex systems used in automotive, aeroplane or space technologies.

### III. EMULATION BASED METHODOLOGY

Analyzing the hardware-first approach we have documented major advantage to this method. Developers using this design method focus on developing a prototype as soon as possible. This strategy complies with the major time-to-market constraints of today's high tech industry. To reduce the risk of design faults and cost intensive redesigns, system designers often use well known components instead of newly available technologies. Our design methodology tries to benefit from the advantages of rapid system design, without the disadvantages of the restrictions described in the previous section. The methodology can be described as a two-stage process:

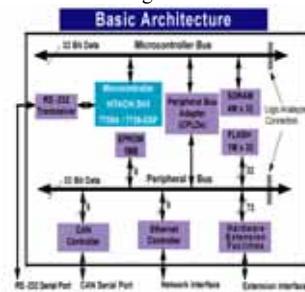
- **Stage One - System design by evaluation**

The basic goal of this stage is the evaluation of components that can be used in the system design. In contrast to the classical hardwarefirst approach, this procedure is not restricted to known or already used hardware or software components. All potentially available components will be analyzed using criteria like functionality, technological complexity, or testability. The source of the criteria used can be data sheets, manuals, etc. The result of this stage is a set of components for potential use, together with a ranking of them.

- **Stage Two: Validation by Emulation:**

Although stage one is based on functional and non-functional criteria, the knowledge and experience of the system designer still exerts a large influence on decisions. In order to avoid fatal design errors, stage two validates the decisions made in stage one. The basic methodology for this validation is system emulation. In contrast to other approaches like computer simulation, emulation can check „serious“ problems, like real time behavior. It is highly essential to verify the criteria used in stage one, for example, the correctness of data sheet specifications.

Figure 3.



Emulation Based Methodology Figure 3 gives an more detailed overview of our methodology. After the specification of the system design, the developer makes an initial hardware/software partitioning. The outcome is a set of hardware and software IP-Cores, the potential candidates that can be used to construct the system. The candidates can be selected from a library or another data base of information. After these introductory steps, the first stage of our methodology follows. The evaluation and selection process focuses on a set of criteria, like testability. The output is a set of components which satisfy such special criteria in the best possible manner. Refer to [6] for a detailed description of this process. After establishing the criteria, the already described „validation“ stage follows. Only if a component passes this „test phase“, it will be used in the final system design.

### *3.1 Stage One: Decision-making Criteria and Ranking*

The previous chapter gave a short overview of the principles of our approach. The evaluation stage which was described is based on a process that puts together a ranking for components by focussing on special criteria. This chapter will explain how to define these criterias and which ranking will be used for selecting or throwing out components. The most important component of an embedded system is the microcontroller. That is why there are only a few types of controllers available, but the choice of the microcontroller determines basics like the system bus, power supply voltages, etc. The first stage of our emulation-based design approach is aware of such choices, as Figure 4 shows. The features of the microcontroller, especially performance determine what will be implemented as software. A system which is equipped with a high performance microprocessor can implement time-consuming functions, like MPEG-decoding software. If the microcontroller fails to complete this task, additional hardware must be added. Due to the high costs of ASIC design, the only possibility is to select the right components from a pool of available chips or IP-Cores.

Embedded systems pose unique debugging challenges. With neither terminal nor display (in most cases), there's no natural way to probe these devices, to extract the behavioral information needed to find what's wrong. This magazine is filled with ads from vendors selling a wide variety of debuggers. They let us connect an external computer to the system being debugged to enable single stepping, breakpoints, and all of the debug resources enjoyed by programmers of desktop computers.

An in-circuit emulator (ICE) is one of the oldest embedded debugging tools, and is still unmatched in power and capability. It is the only tool that substitutes its own internal processor for the one in your target system. Using one of a number of hardware tricks, the emulator can monitor everything that goes on in this on-board CPU, giving you complete visibility into the target code's operation. In a sense, the emulator is a bridge between your target and your workstation, giving you an interactive terminal peering deeply into the target and a rich set of debugging resources.

Until just a few years ago, most emulators physically replaced the target processor. Users extracted the CPU from its socket, plugging the emulator's cable in instead. Today, we're usually faced with a soldered-in surface-mounted CPU, own CPU. In other cases, the emulator vendor making connection strategies more difficult. Some emulators come with an adapter that clips over the surface-mount processor, tri-stating the device's core, and replacing it with the emulator's provides adapters that can be soldered in place of the target CPU. As chip sizes and lead pitches shrink, the range of connection approaches expands.

### IV. TARGET ACCESS

An emulator's most fundamental resource is target access: the ability to examine and change the contents of registers, memory, and I/O. However, since the ICE replaces the CPU, it generally does not need working hardware to provide this capability. This makes the ICE, by far, the best tool for troubleshooting new or defective systems. For example, you can repeatedly access a single byte of RAM or ROM, creating a known and consistent stimulus to the system that is easy to track using an oscilloscope.

Breakpoints are another important debugging resource. They give you the ability to stop your program at precise locations or conditions (like "stop just before executing line 51"). Emulators also use breakpoints to implement single stepping, since the processor's single-step mode, if any, isn't particularly useful for stepping through C code. There's an important distinction between the two types of breakpoints used by different sorts of debuggers. Software breakpoints work by replacing the destination instruction by a software interrupt, or trap, instruction. Clearly, it's impossible to debug code in ROM with software breakpoints. Emulators generally also offer some number of hardware breakpoints, which use the unit's internal magic to compare the break condition against the execution stream. Hardware breakpoints work in RAM or ROM/flash, or even unused regions of the processor's address spaces.

Complex breakpoints let us ask deeper questions of the tool. A typical condition might be: "Break if the program writes 0x1234 to variable buffer, but only if function get\_data() was called first." Some software-only debuggers (like the one included with Visual C++) offer similar power, but interpret the program at a snail's pace while watching for the trigger condition. Emulators implement complex breakpoints in hardware and, therefore, impose (in general) no performance penalty. ROM and, to some extent, flash add to debugging difficulties. During a typical debug session we might want to recompile and download code many times an hour. You can't do that with ROM. An ICE's emulation memory is high-speed RAM, located inside of the emulator itself, that maps logically in place of your system's ROM. With that in place, you can download firmware changes at will.

Many ICES have programmable guard conditions for accesses to both the emulation and target memory. Thus, it's easy to break when, say, the code wanders off and tries to write to program space, or attempts any sort of access to unused addresses.

Nothing prevents you from mapping emulation memory in place of your entire address space, so you can actually debug much of the code with no working hardware. Why wait for the designers to finish? They'll likely be late anyway. Operate the emulator in stand-alone mode (without the target) and start debugging code long before engineering delivers prototypes. Real-time trace is one of the most important emulator features, and practically unique to this class of debugging tool. Trace captures a snapshot of your executing code to a very large memory array, called the trace buffer, at full speed. It saves thousands to hundreds of thousands of machine cycles, displaying the addresses, the instructions, and transferred data. The emulator and its supporting software translates raw machine cycles to assembly code or even C/C++ statements, drawing on your source files and the link map for assistance. Trace is always accompanied by sophisticated triggering mechanisms. It's easy to start and stop trace collection based on what the program does. An example might be to capture every instance of the execution of an infrequent interrupt service routine. You'll see everything the ISR does, with no impact on the real time performance of the code.

Generally, emulators use no target resources. They don't eat your stack space, memory, or affect the code's execution speed. This "non-intrusive" aspect is critical for dealing with real-time systems.

#### V. PRACTICAL REALITIES

Be aware, though, that emulators face challenges that could change the nature of their market and the tools themselves. As processors shrink, it gets awfully hard to connect anything to those whisker-thin package leads. ICE vendors offer all sorts of adapter options, some of which work better than others.

Skyrocketing CPU speeds also create profound difficulties. At 100MHz, each machine cycle lasts a mere 10ns; even an 18-inch cable between your target and the ICE starts to act as a complex electrical circuit rather than a simple wire. One solution is to shrink the emulator, putting all or most of the unit nearer the target socket. As speeds increase, though, even this option faces tough electrical problems.

Skim through the ads in Embedded Systems Programming and you'll find a wide range of emulators for 8- and 16-bit processors, the arena where speeds are more tractable. Few emulators exist, though, for higher-end processors, due to the immense cost of providing fast emulation memory and a reliable yet speedy connection.

Oddly, one of the biggest user complaints about emulators is their complexity of use. Too many developers never use any but the most basic ICE features. Sophisticated triggering and breakpoint capabilities invariably require rather complex setup steps. Figure on reading the manual and experimenting a bit. Such up-front time will pay off later in the project. Time spent in learning tools always gets the job done faster.

#### VI. CONCLUSION

We started with the introduction of state-of-the-art methodologies for designing embedded systems, focussing on hardware- software partitioning. We have shown the basic

restrictions of these classical approaches. Our solution to overcome these restriction is a new design methodology, which consists of two stages:

- preselection of available components
- validation by emulation

The major advantages of our methodology is a parallel design flow for hardware and software, rapid prototyping and the avoidance of dangerous design risks. We have developed an emulation system called SPYDER to use our approach with real system designs. The methodology and the SPYDER tool set are successfully applied in industrial OEM development projects. Our future work will focus on the internet integration of our emulation environment. The basic goal of our research activities is a world wide distributed development.

#### VII. REFERENCE

- [1]. Russell, D. ,IDE: The Interpreter. *Intelligent Tutoring Systems: Lessons Learned*, Psotka, J., L. Massey, and S. Mutter, eds., Lawrence Erlbaum Associates, Hillsdale, 1988.
- [2]. Carbonell, J. R. AI in CAI: An artificial intelligence approach to computer assisted instruction. IEEE Transactions on Man Machine Systems. Issue No. 11, 1970.
- [3]. Rich, E., Artificial intelligence. New York: McGraw Hill., 1983.
- [4]. Sleeman, D. H. and Brown, J. S. Intelligent tutoring systems. New York: Academic Press, 1982
- [5]. Woolf, B. AI in Education. *Encyclopedia of Artificial Intelligence*, Shapiro, S., ed., John Wiley & Sons, Inc., New York, 1992, pp. 434-444.
- [6]. Greiner R., Learning by Understanding Analogue, Artificial Intelligence35, 81-125.
- [7]. Michalski, R.S., Theory and Methodology of inductive Learning, Artificial Intelligenc, 1983.

# Signal Processing for Reliable Communcition

\*Rakesh Khanna, \*\*Rajni Bala, \*Shubhla puri  
 \*GGS College of Modern Technology, Kharar, \*\*B.B.S.B.E.C, Fatehgarh Sahib

**Abstract:** This paper discuss the importance of filtration for improving the efficency, flexibility in a modal devolved to transfer the packets from one terminal to another. In this paper I have discussed the Butterworth filter IIR type low pass because they have infinite impulse response i.e impulse response is computed at infinite number of samples and it is a feed back system and they are more susceptible to round off noise associated with finite precision arithmetic quantization error and coefficient in accuracies. The experimental results are shown with the help of graphs at different values of order and cut off frequencies.

## I. INTRODUCTION

Filtering is a process by which frequency spectrum of a signal can be modified, reshaped or manipulated to achieve some desired objectives. These objectives can be listed as under

1. To eliminate noise which may be contained in a signal?
2. To remove signal distortion which may be due to imperfection transmission channel?
3. To separate two or more distinct signals which may be purposely mixed for maximizing channel utilization?
4. To resolve signals into their frequency components.
5. To demodulate the signals which were modulated at the transmitter end.
6. To convert digital signals into analog signals.
7. To limit the bandwidth of signals.

## II.TYPES OF FILTERS

There are two types of filters.

- a. Analog Filters: Analog Filters may be defined as a system in which both the input and output are continuous time signals
- b. Digital Filters: Digital filter may be defined as a system in which both the input and the output are discrete-time signals.

## III. TYPES OF ANALOG FILTERS

- a. Low Pass Analog Filter
- b. High Pass Analog Filter
- c. Band pass Analog Filter
- d. Band stop Analog Filter
- e. All Pass Analog Filter

### Types of Digital Filters

- a. Infinite Impulse Response Filter (IIR)
- b. Finite Impulse Response Filter (FIR)

## IV. BUTTERWORTH FILTERS

Butterworth is name of the Engineer which invented the filter. Butterworth approximation is low pass filter approximation it is one of the type of electronic filter design it is designed to have frequency response which is as flat as mathematically possible in pass band. Another name for that is maximally flat magnitude filters.

The frequency response of butter worth filter is maximally flat (no ripple) in the pass band and rolls off towards zero in stop band. They have a monotonically changing magnitude function with W. The Butterworth is the only filter that maintains this same shape for higher order whereas other vertices of filters like Bessel, chebyshev, elliptic have different shapes at higher orders.

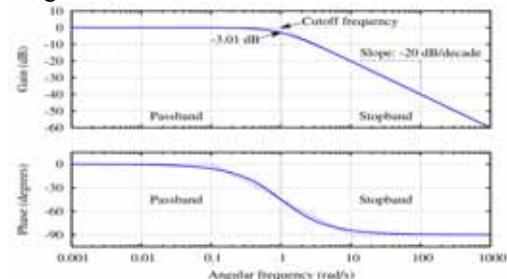


Figure 1.2 - bode plot of a first order Butterworth low pass filter

For a first order filter, the response rolls off at -6 db per octave (-20 db per decade) all first order filter regardless of name, have the same normalized frequency response for a second order butter worth filter, the response decrease at -12 db per octave, a third order at -18 db & so on.

Compared with chebyshev Type I / Type II filter or an elliptic filter, the Butterworth filter has a slower roll off & these will require a higher order to implement a particular stop band specification. How ever Butterworth filter will have a more linear phase response in the pass band than the chebyshev Type I / Type II and elliptic filter.

A Simple example:

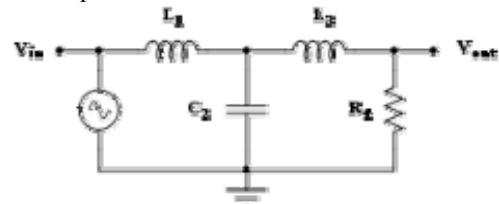


Figure 1.3 - A third order low pass filter (Cauer topology).

The filter becomes a Butterworth filter with cutoff frequency  $\omega_c=1$  when (for example)  $C_2=4/3$  farad,  $R_4=1$  ohm,  $L_1=3/2$  henry and  $L_3=1/2$  henry.

Taking the impedance of the capacitor C to be  $\frac{1}{j\omega}$  & impedance of the inductor L to be  $j\omega S$  where  $S=a+j\omega$  is the complex frequency, the circuit equation yields the transfer function for this device.

$$H(s) = \frac{V_o(s)}{V_i(s)} = \frac{1}{1 + 2s + 2s^2 + s^3}$$

The magnitude of frequency response (gain) g(w) is given by

$$G^2(w) = |H(jw)|^2 = \frac{1}{1+w^6}$$

& the phase is given by

$$\phi(w) = \text{avg}(H(jw))$$

The group delay is defined as derivative of the phase w.r.t angular frequency & is a measure of distortion in the signal introduced by phase differences for different frequencies. The gain & delay for this filter are plotted as under.

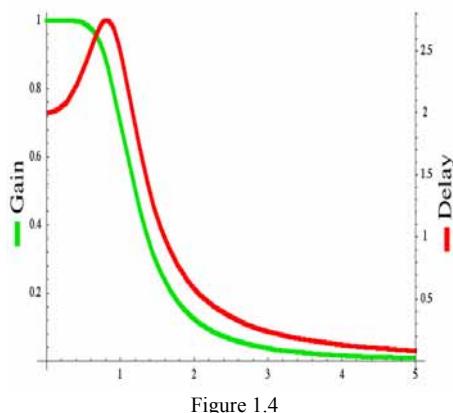


Figure 1.4

It can be seen that there are no ripples in the gain curve in either the pass band or the stop band.

#### V. PROBLEM

There are many methods to upgrade the performance of the system, like multiplexing, spread spectrum techniques and so on, but all these techniques show their merits if and only if the suitable filtering techniques are used. In digital signal processing all the systems are realized by using there block diagram and transfer function. In our work we take both types of techniques in to account. I.e. FIR they are used when the samples are taken of finite length, if the impulse response approaches to infinity then IIR filters are used. Using MATLAB algorithm is written which should have the defined attributes.

The model should have the following attributes:

- Efficiency -- are the constructs efficiently designed?
- Flexibility – is the system flexible for the above model.

Once the model is designed then it will evaluate and compared with the previous developed algorithms/models/systems. To evaluate the system performance, simulations are not only performed for the bit error probability but also for the packet error rate (PER), in which the packet is defined as the number of transmitted data bits in one frame unit. Packet error rate is the percentage in packet received to the packets sent and bit error probability is

the ratio of probability of number of bits, elements, characters incorrectly received to sent during a time interval.

#### VI. CONCLUSION

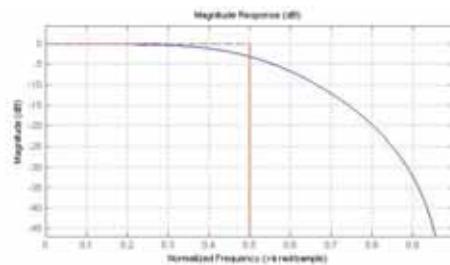
The process of filtration improves the attributes in a modal developed to transfer the data bits from one terminal to another. Experimental results show that for a Butterworth/IIR filter(low pass)

- a) Ideal response should have sharp cut off.
- b) it passes all frequencies below cut off frequency and attenuate all higher frequencies.
- c) Keeping the same order increase in cut off frequency approach towards ideal response.

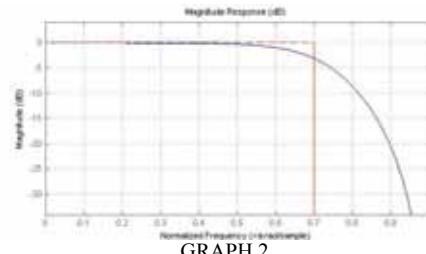
#### VII. EXPERIMENT RESULT

Description: Butter Worth / IIR Filter Type: Low Pass

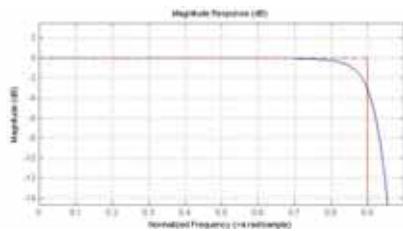
Order	Cut Off Frequency	Graph No
02	0.5	1
02	0.7	2
02	0.9	3
05	0.5	4
05	0.7	5
05	0.9	6
10	0.5	7
10	0.7	8
10	0.9	9
16	0.5	10
16	0.7	11
16	0.9	12
20	0.5	13
20	0.7	14
20	0.9	15



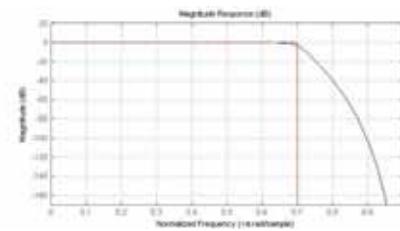
GRAPH 1



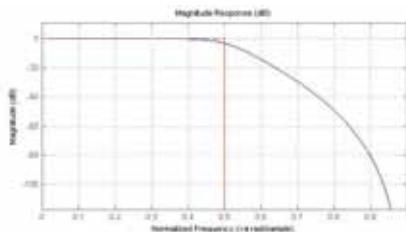
GRAPH 2



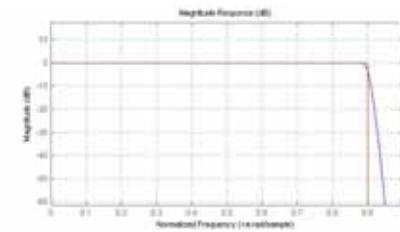
GRAPH 3



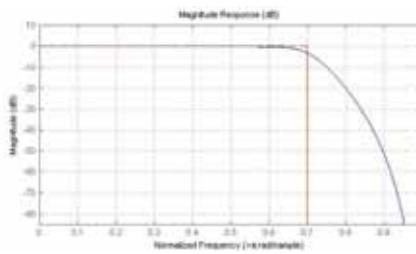
GRAPH 8



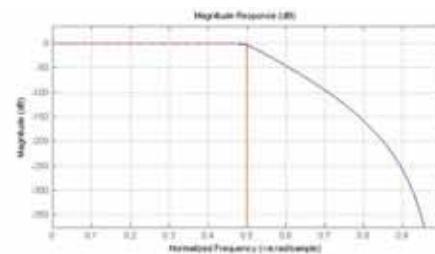
GRAPH 4



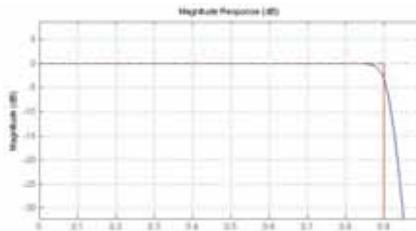
GRAPH 9



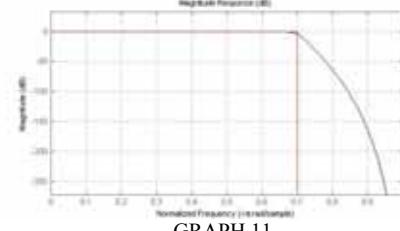
GRAPH 5



GRAPH 10



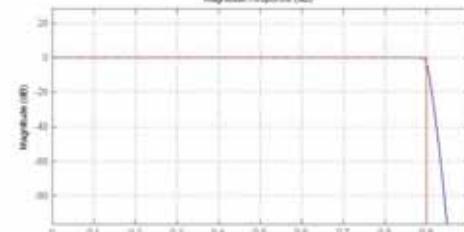
GRAPH 6



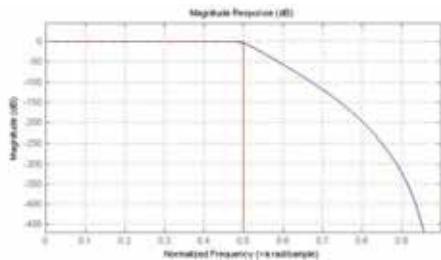
GRAPH 11



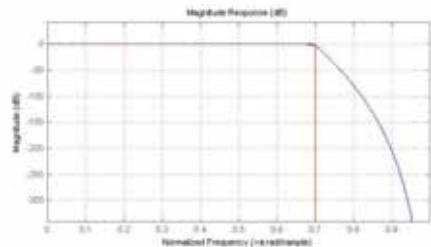
GRAPH 7



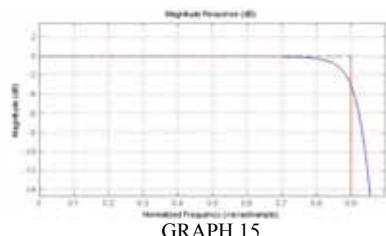
GRAPH 12



GRAPH 13



GRAPH 14



### VIII. REFERENCES

- [1]. Butterworth filter – wikipedia, the free encyclopedia.
- [2]. Kocneneman and mark moonen “a relation between sub band and frequency domain adaptive filtering” IEEE transactions on filters, vol 45, 1977, PP 247.
- [3]. J.M. Deacon and J. Highway “SAW. Filters: some case histories” IEEE transactions on filters, vol 127, 2 april 1980, PP 107.
- [4]. Redondo Beach “EFFECTS OF FINITE WORD LENGTH ON FIR FILTERS FOR MTD PROCESSING” IEEE transactions on filters, vol 130, 6 Oct 1983, PP 573.
- [5]. I.K. Proudler and P.J.W. Rayner “New design procedure for digital filters based on a finite-state machine implementation” IEEE transactions on filters, vol 132, 7 dec 1985, PP 581.
- [6]. Adamo and L.F. Lind “Intersymbol interference and timing jitter performance of realisable data transmission filters” IEEE transactions on filters, vol 133, 1 feb 1986, PP 21.
- [7]. Peter Strobach “FAST RECURSIVE EIGENSPACE ADAPTIVE FILTERS” IEEE transactions on filters, vol 135, 5 Jan 1992, PP 1416.
- [8]. Electronics and communication engineering journal February 1993, PP 532.
- [9]. S. C. Dorigas “GENERALIZED GRADIENT ADAPTIVE STEP SIZES FOR STOCHASTIC GRADIENT ADAPTIVE FILTERS” IEEE transactions on adaptive filters, vol 43, 2 Feb. 1995, PP 1396.

# Post-Processing Method for Reducing Blocking Artifacts using a Deblocking Filter

<sup>1</sup>Meera Thapar Khanna, <sup>2</sup>Jagroop Singh Sidhu- A.P

<sup>1</sup>Dept. of Computer Science and Engineering Punjab Technical Univ. Jalandhar, Punjab

<sup>2</sup> Dept. of Electronics and Communication Engineering DAVIET Jalandhar, Punjab

**Abstract-**A major drawback of block discrete cosine transform-based compressed image is the appearance of visible discontinuities along block boundaries in low bit-rate compressed images, which are commonly referred as blocking artifacts. In this paper, post-processing method for removing these discontinuities are proposed. In this work, a Deblocking algorithm is proposed based on three filtering modes in terms of the activity across block boundaries. The method works in DCT domain. According to the simulation results, the proposed method gives better performance in terms of PSNR. Experiments shows that the proposed method gives excellent results compared with other approaches.

**Keywords:** Blocking artifacts; Deblocking filter; Post-processing; Perceptual quality; PSNR

## I. INTRODUCTION

Blocking artifact is one of the most annoying defects in DCT-based (Discrete Cosine Transform) image compression standards (e.g. JPEG, MPEG) [1, 2], especially when image quality is measured at low bit-rates. This phenomenon is characterized by visually noticeable changes in pixel values across block boundaries. The degradation is a result of a coarse quantization of the DCT coefficients of each image block without taking the inter-block correlations into account. Since blocking artifacts significantly degrade the visual quality of the reconstructed images, it is desirable to be able to monitor and control the visibility of blocking effects in DCT-coded images.

In B-DCT the quantization noise is highly correlated with the characteristics of the original signals, so that different areas of the coded image suffer from distinctly different impairments [3]. In particular, the artifacts create two kinds of visual distortions (i) blurring of sharp edges and changes in the texture patterns, (ii) formation of false edges at interblock boundaries. The first kind of distortion is generally due to near elimination or improper truncation of the high- and mid-frequency DCT coefficients and is efficiently reduced by the proposed AC distribution-based restoration. The other kind is due to severe reductions in the low-frequency DCT coefficients (especially in the DC coefficient) and is tackled with the proposed adaptive spatial filtering. Hence, the two stages are acting complementarily for the removal of blocking artifacts. There are many techniques for the distribution-based estimation of the DCT coefficients. Most of them are based on the fact that there should be a bias in the reconstructed DCT coefficients.

The post-processing method, described in this paper, estimates the local characteristics of the decoded image. It is used to identify the high and low activity regions of the

decoded image. The filtering process is then based on the region classification.

The rest of the paper is organized as follows: Section 2 contains a review and discussion of various techniques that have been proposed in the past for the removal of blocking artifacts. Section 3 describes the model of blocking artifacts. Section 4 presents in detail the blocking artifact reduction algorithm by the Deblocking filtering procedure.

## II. BACKGROUND

Many approaches have been proposed in the literature aiming to alleviate the blocking artifacts in the B-DCT image coding technique. At the encoding end, different transform schemes have suggested, such as the interleaved block transforms [4], the combined transform [5], and so on. However, none of these transform schemes conform to the existing image-/video-coding standards such as JPEG or MPEG, it is difficult or impossible to integrate them with the existing standards.

The other strategy is via post-processing techniques at the decoder side. It appears to be the most practical solution. It does not require changes to existing standards, and with the rapid increase of available computing power more sophisticated methods can be implemented. The blocking-effect is a major obstacle for using BTC to achieve very low bit-rate compression.

Various post-processing techniques have been suggested for the reduction of blocking artifacts, but they often introduce excessive blurring, ringing and in many cases they produce poor deblocking results at certain areas of the image.

The MPEG4 standard [6] offers a deblocking algorithm, which operates in two modes: dc offset mode for low activity blocks and default mode. Block activity is determined according to the amount of changes in the pixels near the block boundaries. All modes apply a one-dimensional (1-D) filter in a separable way. The default mode filter uses the DCT coefficients of the pixels being processed and the dc offset mode uses a Gaussian filter. However, this is not exactly a “pure” post processing method since every quantization factor from each macro block has to be fed into the algorithm.

Another class of postprocessors using iterative image recovery methods based on the theory of projections onto convex sets (POCS) are proposed in [7, 8, 9]. In the POCS based method, closed convex constraint sets are first defined that represents all of the available data on the original uncoded image. Then alternating projections onto these convex Combined Frequency and Spatial Domain Algorithm for the Removal of Blocking Artifacts 603 sets are iteratively

computed to recover the original image from the coded image. POCS are effective in eliminating blocking artifacts but less practical for real-time applications, since the iterative procedure increases the computational complexity.

In [10], Zeng proposed a simple DCT-domain method for blocking effect reduction; applying a zero masking to the DCT coefficients of some shifted image blocks. However, the loss of edge information caused by the zero masking schemes can be noticed.

Most post-processing methods of removing deblocking artifacts result in the filtration of images in the spatial domain [11]-[16] or the transformed domains [17]-[18] and wavelet domain [19]-[21]. Often some constraints reflecting the properties of unprocessed images are imposed on the filtering result [13], [14], [19]. Since the filters commonly have some lowpass properties, this deblocking procedure is actually a smoothing operation. The major challenge is how to effectively smooth out the blocking artifacts without blurring image details.

### III. MODEL OF BLOCKING ARTIFACTS

From the earlier discussion, it has been cleared that the blocking effect normally occurs at the 8×8 block boundary, and is visualized as a false edge when the compression ratio is high [22]. It can be efficiently suppressed by a smoothing procedure. According to the characteristics of a local region to the blocking area, the blocking effect becomes more visible. In designing a deblocking filter, observations of the reconstructed

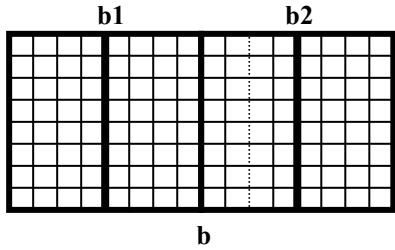


Fig. 1. Graphical representation of b1, b2 and b

image are useful in formulating the appropriate characteristics of a filtering procedure [23]. Let b1 and b2 be the neighboring blocks that are horizontally adjacent to each other. Fig. 1 shows intermediate block b that includes the right part of b1 and left part of b2

### IV. PROPOSED MEASUREMENT SYSTEM

If any blocking artifact is introduced between b1 and b2, the pixel values in b will be abruptly changed. By modeling the abrupt change in b, we can measure the blocking artifacts. The proposed filter attempts to remove blockiness from an image degraded by quantization noise, by observing the characteristics of each region.

#### A. An overview of proposed deblocking algorithm

The proposed algorithm is based on three separate modes, smooth, intermediate, and complex. It appropriately classifies the local characteristics of images according to the

above requirements. The extent of the blocking artifacts clarifies the type of filtering appropriate for each region. Based on the observation, strong filtering is applied to the flat area of block boundary, whereas weak filtering is to be applied to preserve the details in areas of high spatial or temporal activity. An intermediate mode is designed to solve the problem of a too simplistic decision, and either excessive blurring or inadequate removal of the blocking effect. Figure 2 and 3 present a detailed flowchart of the proposed deblocking algorithm.

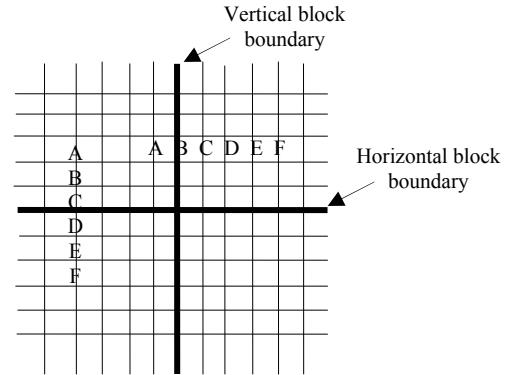


Fig. 2. Position of filtered pixels

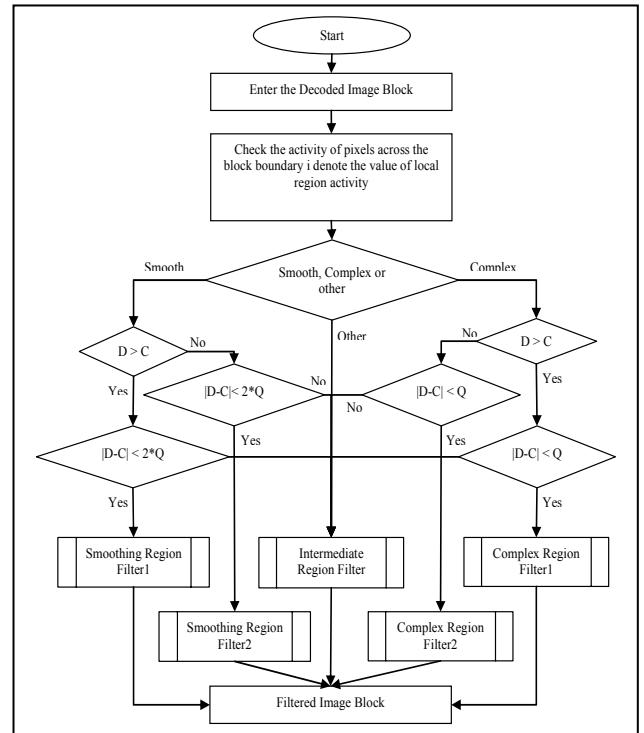


Fig. 3. Flowchart of the proposed deblocking algorithm *Mode Detection*

The key idea behind smoothing the blocked image is to reduce the extent of blockiness without blurring the image. The strength of the blocking effect, is measured by analyzing the pixels between two adjacent blocks. Smoothening is done by considering three neighboring pixels on either side of pixels containing the block edge. As depicted in Fig. 4, X is a 1-D

array of pixel values across a block boundary edge. The following method measures activity.

$$I = \sum_{j=1}^5 \phi(x_j - x_{j+1}) \quad (1)$$

Where

$$\phi(\Delta) = \begin{cases} 0, |\Delta| \leq S \\ 1, \text{otherwise} \end{cases} \quad (2)$$

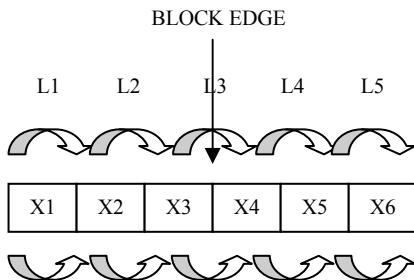


Fig. 4. Mode Detection

The threshold  $S$  is set to a low value so that the function  $\phi(\cdot)$  will return zero to represent an insignificant difference between neighboring pixels. After the five values are determined, then their sum can be seen as a suitable measure of activity. The low value of  $I$  indicate a smooth region, whereas a high value indicates a region with edge detail. The detection is performed on the basis of the value of  $I$ . The value of  $I$  is compared to two thresholds,  $T_1$  and  $T_2$ , to determine the appropriate filtering mode. That is, the pixel being deblocked depends upon the detection criteria. When  $I < T_1$  smooth mode filter is applied and when  $I > T_2$  then complex mode filter is applied. When  $I$  is between  $T_1$  and  $T_2$ , intermediate filtering is used to improve the PSNR and visual quality.  $S$  is set to a minimal value so that  $I$  may reflect the flatness of the local image across a block boundary.

#### B. Deblocking smooth regions

In this mode, the block boundary is between two adjacent smooth 8x8 blocks. The region of pixels near the block boundary, at which the deblocking filter has updated the pixel values, must be accurately determined.

First, the offset is determined from the difference between two pixels across the block boundary. It affects the strength of the blocking effect. So filtering based on offset eliminates the blocking effects in smooth regions. A total of six pixels are updated across the boundary. To prevent real edges, filtering is not performed when the offset is larger than a certain value  $2Q$ . Here  $Q$  is the quality parameter. The value of these updated pixels must be adjusted again within the grayscale range, from 0-255.

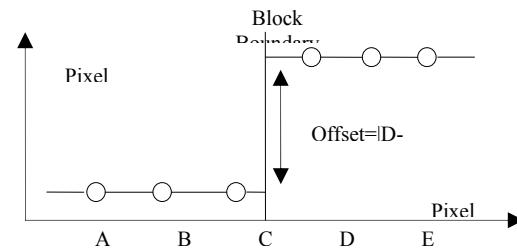


Fig. 5. 1-D illustration of the proposed filtering for smooth region

The steps of the algorithm across a vertical edge (Fig.5) are as follows:

1. Evaluate offset =  $|D-C|$  //difference between two boundary pixels.
2. if  $D > C$  goto step 3 else goto step 6
3. if offset  $< 2*Q$  goto step 4 else goto intermediate filter.
4. Update A, B and C using the offset  
 $a = A + \text{offset}/6$   
 $b = B + \text{offset}/4$   
 $c = C + \text{offset}/2$   
 Update D, E and F by means of the offset  
 $d = D - \text{offset}/2$   
 $e = E - \text{offset}/4$   
 $f = F - \text{offset}/6$   
 if offset  $< 2*Q$  goto step 7 else goto intermediate filter.  
 Update A, B and C using the offset  
 $a = A - \text{offset}/6$   
 $b = B - \text{offset}/4$   
 $c = C - \text{offset}/2$   
 Update D, E and F by means of the offset  
 $d = D + \text{offset}/2$   
 $e = E + \text{offset}/4$   
 $f = F + \text{offset}/6$   
 Adjust these updated pixels value within 0-255

Similar algorithm can be applied to the horizontal edges for reducing blocking artifacts.

#### C. Deblocking complex regions

If the region is of high activity then strong filtering is not appropriate because it over-blurs the true edges of the image. Also increases the computational burden. In this region, filtering is applied only to two pixels from both the boundary regions. So it requires simple control mechanism and hence reduces the computational complexity. The steps of the algorithm across a vertical edge are as follows:

1. Evaluate offset =  $|D-C|$  //difference between two boundary pixels.
2. if  $D > C$  goto step 3 else goto step 6
3. if offset  $< Q$  goto step 4 else goto intermediate filter.
4. Update B and C using the offset  
 $b = B + \text{offset}/6$   
 $c = C + \text{offset}/2$   
 Update D and E by means of the offset  
 $d = D - \text{offset}/2$   
 $e = E - \text{offset}/6$

6. if offset<Q goto step 7 else goto intermediate filter.
7. Update B and C using the offset  
 $b = B - \text{offset}/6$   
 $c = C - \text{offset}/2$
8. Update D and E by means of the offset  
 $d = D + \text{offset}/2$   
 $e = E + \text{offset}/6$
9. Adjust these updated pixels value within 0-255  
Similar algorithm can be applied to the horizontal edges for reducing blocking artifacts.

#### D. Deblocking intermediate regions

A  $3 \times 3$  low pass filter is presented as an intermediate mode filtering. The filtering for an intermediate region balances the strong filtering in a smooth region with the weak filtering in a complex region. It also reduces the computational complexity because only shifting is applied as compared to the division in other filtering methods. The filter specifications (Fig. 6) are as follows:

$$S_1 = \sum_{\substack{i=1 \\ i \neq 5}}^9 \alpha_i p_i \quad (3)$$

$$\alpha_i = \begin{cases} 1 & \text{if } |p_5 - p_i| < Th \\ 0 & \text{else} \end{cases} \quad (4)$$

$$Th = 33 - Q/3 \quad (5)$$

$$S_2 = \sum_{\substack{i=1 \\ i \neq 5}}^9 \alpha_i \quad (6)$$

$$p'_5 = (p_5 + S_1)/(1 + S_2) \quad (7)$$

Applying this low pass filter to the pixels on the either side of the block boundary, C and D, reduces the blocking effect with minimal loss of image content. This filter is adaptive in two ways. First, only the pixels near the boundary are selected in the filtering window and their gray value is within a specified range around the gray value of the pixel to be filtered. Secondly, the threshold Th (0-30) is adjusted according to the JPEG quality parameter Q (1-100) of the block.

#### V. RESULTS AND CONCLUSION

PSNR - Peak Signal to Noise Ratio is used to measure the blocking artifacts in the image. PSNR is basically a logarithmic scale of the mean squared difference between two sets of values (pixel values, in this case). It is used as a general measure of image quality, but it does not specifically measure blocking

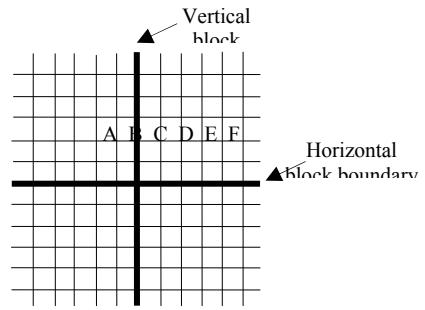


Fig. 6. Pixels of the filtering for intermediate region.

artifacts. In observed literature, PSNR is used as a source-dependant artifact measure, requiring the original, uncompressed image to compare with. PSNR is defined as:  
 $\text{PSNR} = 20 \log_{10} (255/\sqrt{\text{MSE}})$

where

$$\text{MSE} = \sum (X_i - Y_i)/n$$

where X and Y are the original and deblocked images respectively; i = 0 to n and n is the number of pixels in the image.

It is easily seen that this blockiness measure is not actually aware of the artifacts it is measuring - it is simply a gauge of how different the corresponding (that is, the same position) pixel values are between two images. So PSNR is an acceptable measure, and hence the primary measure used to compare the proposed method. However, two images with completely different levels of perceived blockiness may have almost identical PSNR values.

In this experiment, the proposed algorithm depends on some predefined parameters. For the mode to be selected, the activity between block boundaries must be measured. The threshold S is set to 2 to correspond to the smooth region appropriate for intenseive filtering. After the activity across block boundary is measured, two thresholds, T1 and T2 determine the appropriate deblocking mode. T1 is set to 2 and T2 is set to 3.

In order to evaluate the performance of proposed technique six different  $512 \times 512$  images are coded at different bit rates using the JPEG standard. Fig. 7-9 show the results of six test images which are Lena, Pentagon, Peppers, Elaine, Bridge, Baboon at 0.25bpp respectively.



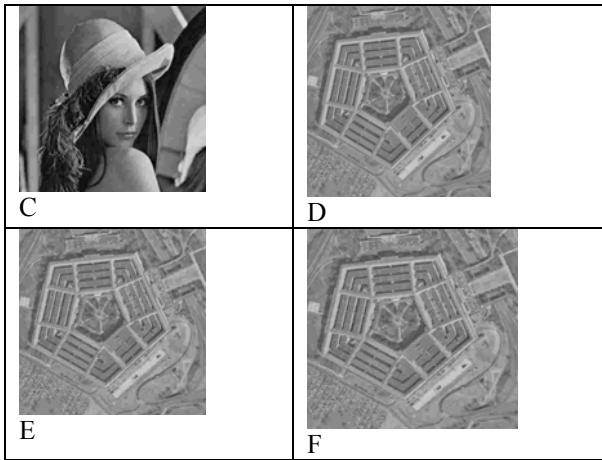


Fig.7. bit rate 0.25bpp (a) Lena original test image. (b) Compression by JPEG, PSNR= 35.4275dB. (c) Filtered image PSNR=35.7632dB (d) Pentagon original test image. (e) Compression by JPEG, PSNR=31.9495dB (f) Filtered image PSNR=32.1051Db

The mode classification algorithm enables better control of the image quality. The result of the proposed algorithm demonstrates not only removing the blocking artifacts but also removing the remaining edges near the real edges. According to Figs. 7-9, the proposed scheme provides better perception quality than other methods.

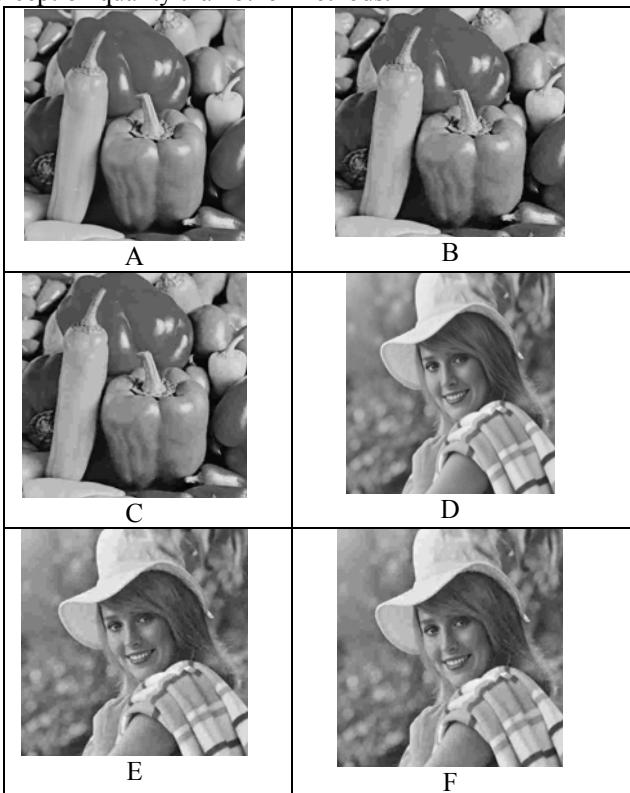


Fig.8. bit rate 0.25bpp (a) Peppers original test image. (b) Compression by JPEG, PSNR= 34.6165dB. (c) Filtered image PSNR=34.9133dB (d) Elaine original test image. (e) Compression by JPEG, PSNR=34.5714dB (f) Filtered image PSNR=34.8590Db

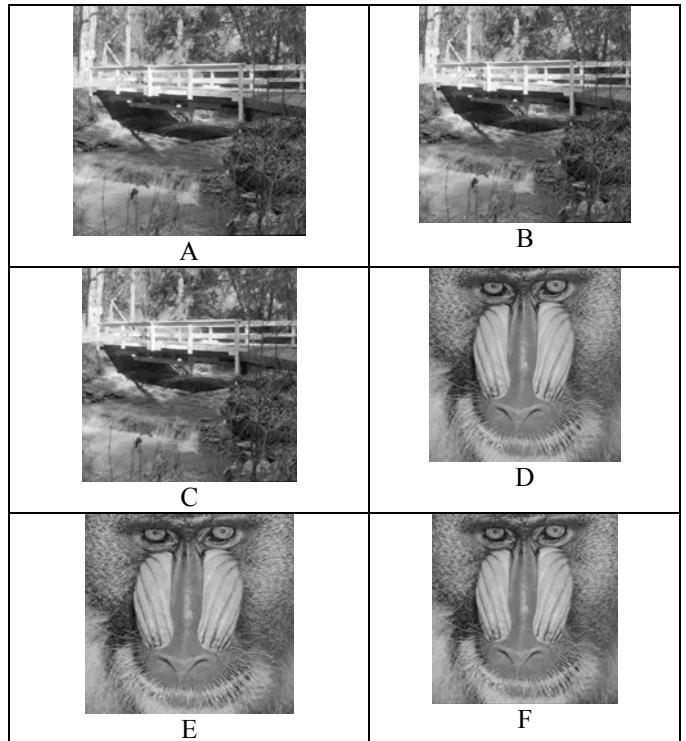


Fig.9 bit rate 0.25bpp (a) Bridge original test image. (b) Compression by JPEG, PSNR= 30.9228dB. (c) Filtered image PSNR=31.0046dB (d) Baboon original test image. (e) Compression by JPEG, PSNR=30.3605dB (f) Filtered image PSNR=30.4126Db

Resulting MSE and PSNR values at different bit rates are listed in Table 1 and Table 2 respectively. At the same bit rates, the proposed algorithm gives better performance in terms of PSNR and MSE as compared to JPEG compressed images. The proposed filter can improve the PSNR value by 0.4 dB over other filters.

Table 1  
The MSE values in comparison with JPEG and Proposed algorithm at different bit rate

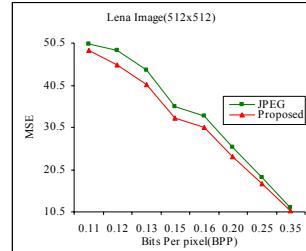
Image	Bit rate	JPEG	Proposed
Lena	0.11	50.2962	48.8787
	0.12	48.7651	45.4267
	0.13	44.0466	40.6497
	0.15	35.4108	32.6457
	0.16	33.1491	30.4677
	0.20	25.9318	23.7004
	0.25	18.6352	17.249
	0.35	11.4535	10.7158
Pentagon	0.11	65.7216	64.5414
	0.12	63.6492	61.3621
	0.13	57.7637	55.6363
	0.15	53.5961	51.5548
	0.16	51.2855	49.2261
	0.20	46.5899	44.9178
	0.25	41.5079	40.0472
Peppers	0.35	33.8937	32.9109
	0.11	50.8628	49.6995
	0.12	49.0536	45.6506

	0.13	45.0688	41.6985		0.16	32.6005	32.9373
	0.15	39.0461	36.1728		0.20	33.5726	33.9344
	0.16	35.7298	33.0639		0.25	34.6165	34.9133
	0.20	28.5638	26.2806		0.35	36.1326	36.3864
	0.25	22.4609	20.9772	Elaine	0.11	30.9561	31.0431
	0.35	15.8425	14.943		0.12	31.2756	31.6414
Elaine	0.11	52.1756	51.1417		0.13	31.6243	31.9868
	0.12	48.4747	44.5597		0.15	32.3192	32.6831
	0.13	44.7353	41.1532		0.16	32.5818	32.9642
	0.15	38.1203	35.0568		0.20	33.5013	33.8808
	0.16	35.884	32.8598		0.25	34.5714	34.8590
	0.20	29.0369	26.6072		0.35	36.0523	36.2978
	0.25	22.6957	21.2411	Bridge	0.11	29.5033	29.5572
	0.35	16.1381	15.251		0.12	29.5519	29.6551
Bridge	0.11	72.9034	72.0039		0.13	29.7578	29.8536
	0.12	72.0924	70.4002		0.15	30.0136	30.1010
	0.13	68.7548	67.2542		0.16	30.1528	30.2594
	0.15	64.8219	63.53		0.20	30.4962	30.5867
	0.16	62.772	61.2553		0.25	30.9228	31.0046
	0.20	58.0046	56.8085		0.35	31.6782	31.7418
	0.25	52.5776	51.5966	Baboon	0.11	29.4114	29.4470
	0.35	44.1831	43.541		0.12	29.3966	29.4697
Baboon	0.11	74.4634	73.8543		0.13	29.4282	29.5049
	0.12	74.7176	73.4701		0.15	29.6738	29.7573
	0.13	74.1751	72.8775		0.16	29.7692	29.8455
	0.15	70.0966	68.763		0.20	30.0554	30.1285
	0.16	68.5737	67.3802		0.25	30.3605	30.4126
	0.20	64.2005	63.1295		0.35	30.9023	30.9324
	0.25	59.8456	59.132				
	0.35	52.826	52.4617				

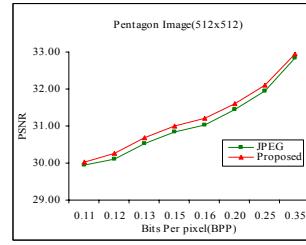
Table 2

The PSNR values in comparison with JPEG and Proposed algorithm at different bit rate

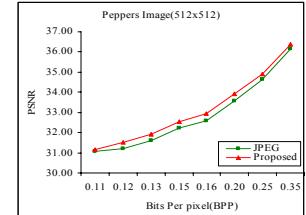
Image	Bit rate	JPEG	Proposed
Lena	0.11	31.1154	31.2396
	0.12	31.2497	31.5577
	0.13	31.6917	32.0402
	0.15	32.6394	32.9926
	0.16	32.9261	33.2924
	0.20	33.9925	34.3832
	0.25	35.4275	35.7632
	0.35	37.5414	37.8306
Pentagon	0.11	29.9537	30.0324
	0.12	30.0929	30.2518
	0.13	30.5143	30.6772
	0.15	30.8395	31.0081
	0.16	31.0309	31.2089
	0.20	31.4479	31.6066
	0.25	31.9495	32.1051
	0.35	32.8296	32.9574
Peppers	0.11	31.0668	31.1673
	0.12	31.2241	31.5363
	0.13	31.5920	31.9296
	0.15	32.2150	32.5470



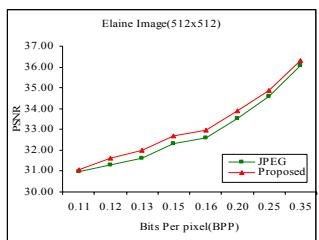
(a)



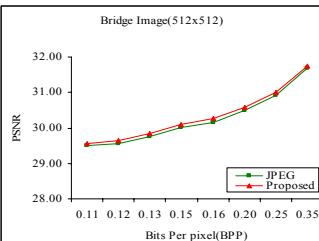
B



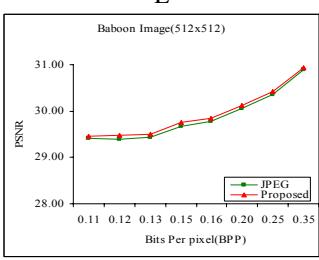
C



D



E



F

Fig.11. PSNR versus bit rate (bits per pixel) for the JPEG coded images (a) Lena image (512x512) (b) Pentagon image (512x512) (c) Peppers image (512x512) (d) Elaine image (512x512) (e) Bridge image (512x512) (f) Baboon image (512x512)

Fig. 10-11 are the graphical representation of bits per pixel versus MSE and PSNR respectively both in JPEG and proposed images. Graphical representation shows the improvements in PSNR and MSE.

## VI. CONCLUSION

This paper proposed a post-processing algorithm for reducing blocking artifact in transformed coded images. The proposed algorithm is based on the 1-D filtering of block boundaries. Results show that using three filtering modes effective deblocking is achieved. The proposed technique provides better image quality across a wide variety of images. To demonstrate the performance of the proposed algorithm PSNR measure has been used. It is found that there is a significant improvement in the perceptual quality of the JPEG compressed images after removal of blocking artifact by the proposed method. Due to its low computational cost, the technique can be integrated into real-time image/video applications as a method for online quality monitoring and control.

## VII REFERENCES

- [1] ISO/IEC JTC11/SC29/WG11, “Generic Coding of Moving Pictures and Associated Audio Information: Video”, ISO/IEC 13818-2.
- [2] Draft ITU-T Recommendation and Final Draft International Standard of joint Video Specification (ITU-T Rec. H. 264/ISO/IEC 14 496-10 AVC), March 2003.
- [3] H. A. Peterson, A. J. Ahumada, and A. B. Watson, “The visibility of DCT quantization noise” in J. Marreale, ed., Digest of Technical Papers, Society for Information Display, Playa del Rey, CA, 1993, vol.24, pp. 942-945.
- [4] H. Reeve, J. Lim, Reduction of blocking effect in image coding, Proc. ICASSP 83(1983) 1212-1215.
- [5] B. Ramamurthy, A. Gershoff, Nonlinear space-variant post processing of block coded images, IEEE Trans. Account. Speech Signal Process. ASSP-34 (1986) 1258-1268.
- [6] “Mpeg4 video verification model version 18.0,” ISO/IEC 14496-2.
- [7] Y. Yang, N. P. Galatsanos, and A. K. Katsaggelos, “Projection based spatially adaptive reconstruction of block-transform compressed images,” IEEE Trans. Image Processing, vol. 4, no. 7, pp. 896–908, 1995.
- [8] H. Paek, R.-C. Kim, and S.-U. Lee, “On the POCS-based post processing technique to reduce the blocking artifacts in transform coded images,” IEEE Trans. Circuits and Systems for Video Technology, vol. 8, no. 3, pp. 358–367, 1998.
- [9] H. Paek, R.-C. Kim, and S.-U. Lee, “A DCT-based spatially adaptive post processing technique to reduce the blocking artifacts in transform coded images,” IEEE Trans. Circuits and Systems for Video Technology, vol. 10, no. 1, pp. 36–41, 2000.
- [10] B. Zeng, Reduction of blocking effect in DCT-coded images using zero masking techniques, Signal Image Process. 79 (1999) 205-211.
- [11] MPEG4 Video Group, MPEG4 video verification model version 18.0, JTC1/SC29/WG11 N3908, 2002.
- [12] H.263 Video Group, Video Coding for Low Bit Rate Communication ITU-T Rec. H.263, 1998.
- [13] A. Zakhori, Iterative procedure for reduction of blocking effects in transform image coding, IEEE Trans. Circuits Syst. Video Technol., vol.2, no. 1, pp. 91-95, Jan. 1992.
- [14] Y. Y. Yang, P. Galatsanos, and A. K. Katsaggelos, Regularized reconstruction to reduce blocking artifacts of block discrete cosine transform compressed images. IEEE Trans. Circuits Syst. Video Technol., vol.3, no. 6, pp. 421-432, Aug. 1993.
- [15] R. Castagno, S. Marsi, and G. Rampony, A simple algorithm for the reduction of blocking effects in images and implementation, IEEE Trans. Consum. Electron., vol.44, no. 3, pp. 1062-1070, 1998.
- [16] X. Gan, A. W. C. Liew, and H. Yan, Blocking artifact reduction in compressed images based on edge-adaptive quadrangles meshes, J. Vis. Commun. Image Represent., vol.14, no. 4, pp. 492-507, 2003.
- [17] Y. Luo and R.K. Ward. Removing the Blocking Artifacts of Blocked-Based DCT Compressed Images, IEEE TIP, 12(7): 838–842, July 2003.
- [18] T. Chen, H. R. Wu, and B. Qui, Adaptive postfiltering of transform coefficients for the Reduction of blocking artifacts, IEEE Trans. Circuits Syst. Video Technol., vol.11, no. 5, pp. 594-602, May. 2001.
- [19] A. W. C. Liew and H. Yan, blocking artifacts suppression in block coded images using overcomplete wavelet representation, IEEE Trans. Circuits Syst. Video Technol., vol.14, no. 4, pp. 450-461, Apr. 2004.
- [20] T. C. Hsung, D. P. –K. Lun, and W. C. Siu, A deblocking technique for block-transform images using wavelet transform modules maxima, IEEE Trans. Image Process., vol.7, no. 10, pp. 1488-1496, Oct. 1998.
- [21] Z. Xiong, M. T. Orchard, and Y. Q. Zhang, A deblocking algorithm for JPEG compressed images using overcomplete wavelet representations, IEEE Trans. Circuits Syst. Video Technol., vol.7, no. 2, pp. 433-437, Feb. 1997.
- [22] Merhay, N., Bhaskaran, V. “Fast Algorithms for DCT-domain Image Down-Sampling and for Inverse Motion Compensation”, IEEE Trans. Circuits Syst. Video Technol., vol.7, pp. 468-476 (1997).
- [23] Liu, S.Z., Bovik, A.C. “Efficient DCT-domain Blind Measurement and Reduction of Blocking Artifacts”, IEEE Trans. Circuits Syst. Video Technol., 12(12), 1139-1149 (2002).

# Algorithm of Back Propagation Network Implementation in VHDL

Amit Goyal, Prof. J.P.S Raina\*

Swami Devi Dyal Institute of Engineering & Technology, Barwala (INDIA).

\*Baba Banda Singh Bahadur Engineering College, Fatehgarh Sahib (INDIA)

amitgoyal1979@yahoo.co.in, \*jps.raina@bbsbec.ac.in

**Abstract-**A neural network is a powerful data-modeling tool that is able to capture and represent complex input/output relationships. The motivation for the development of neural network technology stemmed from the desire to develop an artificial system that could perform "intelligent" tasks similar to those performed by the human brain. An Artificial neural network (ANN), also called a simulated neural network (SNN) or commonly just neural network (NN) is an interconnected group of artificial neurons that uses a mathematical or computational model for information processing. In most cases ANN is an adaptive system that changes its structure based on internal or external information that flows through the network. Minsky and Papert (1969) showed that there are many simple problems such as the exclusive-or problem which linear neural networks can not solve. Note that term "solve" means learn the desired associative links. Argument is that if such networks can not solve such simple problems how they could solve complex problems in vision, language, and motor control.

## I. INTRODUCTION

The highlighted area of the paper is the implementation of the back propagation algorithm of neural network in VHDL and formulation of individual modules of the Back Propagation algorithm for efficient implementation in hardware that creates a flexible, fast method and high degree of parallelism for implementing the algorithm. An ANN can create its own organization or representation of the information it receives during learning time. ANN computations may be carried out in parallel, and special hardware devices are being designed and manufactured which take advantage of this capability. Partial destruction of a network leads to the corresponding degradation of performance. However, some network capabilities may be retained even with major network damage [1].

Use of back Propagation Neural Network solution:

— A large amount of input/output data is available, but you're not sure how to relate it to the output.

— The problem appears to have overwhelming complexity, but there is clearly a solution.

— It is easy to create a number of examples of the correct behavior.

— The solution to the problem may change over time, within the bounds of the given input and output parameters (i.e., today  $2+2=4$ , but in the future we may find that  $2+2=3.8$ ).

— Outputs can be "fuzzy", or non-numeric.

Hardware description languages are especially useful to gain more control of parallel processes as well as to circumvent some of the idiosyncrasies of the higher level programming languages.

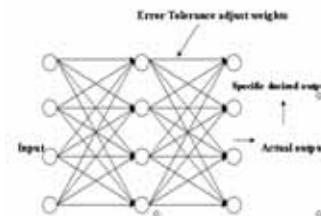


Fig 1: Back Propagation Network

VHDL is a programming language that has been designed and optimized for describing the behavior of digital systems. VHDL has many features appropriate for describing the behavior of electronic components ranging from simple logic gates to complete microprocessors and custom chips. One of the most important applications of VHDL is to capture the performance specification for a circuit, in the form of what is commonly referred to as a test bench.

## II. OBJECTIVE

In this paper I have proposed an expandable on-chip back-propagation (BP) learning neural network. The network has four neurons and 16 synapses. Large-scale neural networks with arbitrary layers and discretionary neurons per layer can be constructed by combining a certain number of such unit networks. A novel neuron circuit with programmable parameters, which generates not only the sigmoid function but also its derivative, is proposed. The back-propagation (BP) algorithm often provides a practical approach to a wide range of problems. There have been many examples of implementations of general-purpose neural networks with on-chip BP learning.

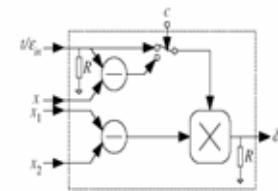


Fig 2: Block diagram of error generator

The block diagram of the error generator unit is shown in the above figure. It provides a weight error signal  $\delta$ . The control signal  $c$  decides whether the corresponding neuron is an output neuron.  $t/\epsilon_{in}$  is a twofold port. If  $c=1$ , a target value  $t$  is imported to the  $t/\epsilon_{in}$  port and  $\delta$  is obtained by multiplying  $(t-x)$  by  $(x_1-x_2)$ ; If  $c=0$ , a neuron error value  $\epsilon_{in}$  is imported to the  $t/\epsilon_{in}$  port and  $d$  is achieved by multiplying  $\epsilon_{in}$  by  $(x_1-x_2)$ . So no additional

output chips are needed to construct a whole on-chip learning system.

In broad sense the objectives of thesis are:

– Exploration of a supervised learning algorithm for artificial neural networks i.e. the

Error Back propagation learning algorithm for a layered feed forward network.

– Formulation of individual modules of the Back Propagation algorithm for efficient implementation in hardware.

– Implementation of the Back Propagation learning algorithm in VHDL. VHDL implementation creates a flexible, fast method and high degree of parallelism for implementing the algorithm.

– Analysis of the simulation results of Back Propagation algorithm.

## 2.2 Description

The back propagation algorithm is an involved mathematical tool; however, execution of the training equations is based on iterative processes, and thus is easily implementable on a computer [2].

– Weight changes for hidden to output weights just like Widrow-Hoff learning rule.

– Weight changes for input to hidden weights just like Widrow-Hoff learning rule but error signal is obtained by "back-propagating" error from the output units.

## 2.3 Problem Statement

Minsky and Papert (1969) showed that there are many simple problems such as the exclusive-or problem which linear neural networks can not solve. Note that term "solve" means learn the desired associative links. Argument is that if such networks can not solve such simple problems how they could solve complex problems in vision, language, and motor control. Solutions to this problem were as follows:

- Select appropriate "recoding" scheme which transforms inputs.
- Perceptron Learning Rule -- Requires that you correctly "guess" an acceptable input to hidden unit mapping.
- Back-propagation learning rule -- Learn both sets of weights simultaneously.

## 2.4 Background

The study of the human brain is thousands of years old.

With the advent of modern electronics, it was only natural to try to harness this thinking process. [1], [3].

One particular structure, the feed forward, back-propagation network, is by far and away the most popular. Most of the other neural network structures represent models for "thinking" that are still being evolved in the laboratories. Yet, all of these networks are simply tools and as such the only real demand they make is that they require the network architect to learn how to use them. [4]

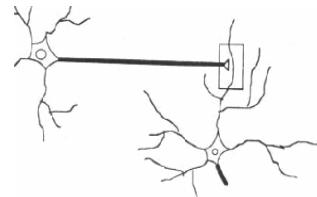


Fig 3: The Synapse

## III. APPROACH

There are many existing theoretical approaches to strategy - designed strategy, strategy as revolution etc and yet few examples of organizations applying these well defined models to secure competitive advantage in an environment of constant change. Proper utility of technology and time will increase the general output as given in the below theory [5].

### 3.1 Theory

To make a neural network that performs some specific task, we must choose how the units are connected to one another, and we must set the weights on the connections appropriately. The connections determine whether it is possible for one unit to influence another. The weights specify the strength of the influence. A three-layer network can be taught to perform a particular task by using the following procedure:

- The network is presented with training examples, which consist of a pattern of activities for the input units together with the desired pattern of activities for the output units.
- It is determined how closely the actual output of the network matches the desired output.
- The weight of each connection can be changed so that the network produces a approximation of the desired output.

### 3.2 Evolutionary approach

Take a collection of training patterns for a node, some of which cause it to fire (the 1-taught set of patterns) and others, which prevent it from doing so (the 0-taught set). Then the patterns not in the collection cause the node to fire if, on comparison, they have more input elements in common with the 'nearest' pattern in the 1-taught set than with the 'nearest' pattern in the 0-taught set. If there is a tie, then the pattern remains in the undefined state.

For example, a 3-input neuron is taught to output 1 when the input ( $X_1, X_2$  and  $X_3$ ) is 111 or 101 and to output 0 when the input is 000 or 001. Then, before applying the firing rule, the truth table is:

X1:	0	0	0	0	1	1	1	1
X2:	0	0	1	1	0	0	1	1
X3:	0	1	0	1	0	1	0	1
OUT:	0	0	0/1	0/1	0/1	1	0/1	1

Truth Table 1: before applying the firing rule

As an example of the way the firing rule is applied, take the pattern 010. It differs from 000 in 1 element, from 001 in 2 elements, from 101 in 3 elements and from 111 in 2 elements.

Therefore, the 'nearest' pattern is 000, which belongs, in the 0-taught set. Thus the firing rule requires that the neuron should not fire when the input is 001. On the other hand, 011 is equally distant from two taught patterns that have different outputs and thus the output stays undefined (0/1).

By applying the firing in every column the following truth table is obtained:

X1:		0	0	0	0	1	1	1
X2:		0	0	1	1	0	0	1

Truth Table 2: after applying the firing rule

### 3.3 Methodology / Planning of Work

Extensive literature survey will be conducted to study the literature available in the field of Back Propagation Algorithm in VHDL Implementation, characterization and simulation. Generally the behavior of an ANN (Artificial Neural Network) depends on both the weights and the input-output function (transfer function) that is specified for the units. his function typically falls into one of three categories:

- o Linear (or ramp): The output activity is proportional to the total weighted output.
- o Threshold: The output is set at one of two levels, depending on whether the total input is greater than or less than some threshold value.
- o Sigmoid: The output varies continuously but not linearly as the input changes.

Sigmoid units bear a greater resemblance to real neurons than do linear or threshold units, but all three must be considered rough approximations.



Step function	Sign funt.	Sigmoid funt.
Step(x) = 1, if x >= threshold x < threshold	Sign(x) = if x >= 0 Sign(x) = -1 If x < 0	+1 Sigmoid(x) = 1 / (1+ex)

Fig 4: Transfer Functions

## IV. SIMULATION RESULTS

Before starting the back propagation learning process, we need the following:

- The set of training patterns, input, and target
- A value for the learning rate
- A criterion that terminates the algorithm
- A methodology for updating weights
- The nonlinearity function (usually the sigmoid)
- Initial weight values (typically small random values)

Then the simulation process is implemented. Simulation will allow investigation of models which are highly intractable from a mathematical point of view. Simulation allows checking whether certain approximations made in the mathematical

model for the sake of mathematical tractability [6]. Then iterations will be performed and the results of the implementation of the back propagation algorithm may be presented in the form of MODELSIM SE 5.5c.

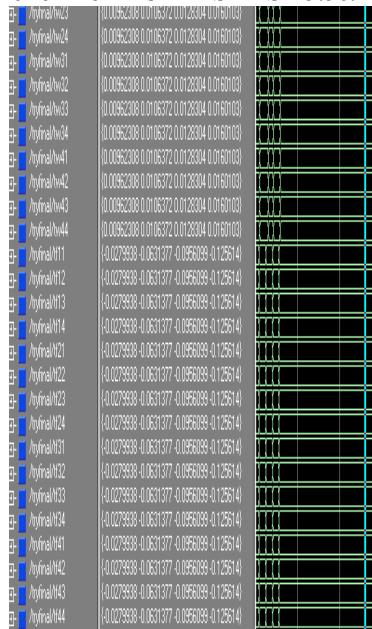


Fig 5: Simulation results for the final entity

## V. CONCLUSION

This paper describes the VHDL implementation of a supervised learning algorithm for artificial neural networks. The algorithm is the Error Back propagation learning algorithm for a layered feed forward network and this algorithm has many successful applications for training multilayer neural networks. VHDL implementation creates a flexible, fast method and high degree of parallelism for implementing the algorithm. Then a final entity having structural modeling has been formulated in which all the entities are port mapped. The results constitute simulation of VHDL codes of different modules in MODELSIM SE 5.5c. The simulation of the structural model shows that the neural network is learning and the output of the second layer is approaching the target.

## VI. REFERENCES

- [1]. Christos Stergiou and Dimitrios Siganos, "Neural Networks", Computer Science
- [2]. Deptt. University of U.K., Journal, Vol. 4, 1996.
- [3]. Andrew Blais and David Mertz, "An Introduction to Neural Networks – Pattern
- [4]. Learning with Back Propagation Algorithm", Gnosis Software, Inc., July 2001.
- [5]. Jordan B.Pollack, "Connectionism: Past, Present and Future", Computer and
- [6]. Information Science Department, The Ohio State University, 1998.
- [7]. Harry B.Burke, "Evaluating Artificial Neural Networks for Medical applications", New York Medical College, Deptt. Of Medicine, Valhalla, IEEE, 1997.
- [8]. Tom Baker and Dan Hammerstrom, "Characterization of Artificial Neural Network Algorithms", Dept. of Computer Science and Engineering, Oregon Graduate Center, IEEE, 1989.
- [9]. Christopher Cantrell, Dr. Larry Wurtz, "A parallel Bus Architecture for Ann's",
- [10]. University of Albana, Deptt. OF electrical Engg, Tuscaloosa, IEEE, 1993.

# Energy Harvesting using Piezoelectric Materials

Anil Kumar Goyal<sup>1</sup>Harmeek Singh Hans<sup>2</sup>

<sup>1</sup>Lecturer, Department of ECE, Chitkara University, Barotiwala

<sup>2</sup>Student, Department of Mechanical Engineering, UIET, Panjab University, [er.hshans@gmail.com](mailto:er.hshans@gmail.com)

**Abstract-** Piezoelectricity is the ability of some materials (notably crystals and certain ceramics) to generate an electric potential. Primarily used for sensor and transducer functions, these materials are now seen as potential source of mass scale electricity production. The study explains the fundamentals of generation of electric potential due to imposing of external applied mechanical stress in piezoelectric crystals and a review of technology being used for electricity production.

## I. INTRODUCTION

A piezoelectric substance is one that produces an electric charge when a mechanical stress is applied. Conversely, a mechanical deformation (the substance shrinks or expands) is produced when an electric field is applied. This effect is formed in crystals that have no centre of symmetry. The first demonstration of the direct piezoelectric effect was in 1880 by the brothers Pierre Curie and Jacques Curie. They combined their knowledge of Pyroelectricity with their understanding of the underlying crystal structures that gave rise to Pyroelectricity to predict crystal behavior, and demonstrated the effect using crystals of tourmaline, quartz, topaz, canesugar, and Rochelle salt (sodium potassium tartrate tetrahydrate). Quartz and Rochelle salt exhibited the most piezoelectricity. The origin of the piezoeffect was, in general, clear from the very beginning. The displacement of ions from their equilibrium positions caused by a mechanical stress in crystals that lack a centre of symmetry result in generation of an electric moment, i.e., in electric polarization.

The Curies, however, did not predict the converse piezoelectric effect. The converse effect was mathematically deduced from fundamental thermodynamic principles by Gabriel Lippmann in 1881. The Curies immediately confirmed the existence of the converse effect, and went on to obtain quantitative proof of the complete reversibility of electro-elasto-mechanical deformations in piezoelectric crystals.

## II. MATERIALS

Natural crystal - Berlinit (ALPO<sub>4</sub>), cane sugar, quartz, Rochelle salt, Topaz and Tourmaline-group minerals. Dry bone exhibits some piezoelectric properties. Studies of Fukada et al. showed that these are not due to the apatite crystals, which are centrosymmetric, thus non-piezoelectric, but due to collagen. Collagen exhibits the polar uniaxial orientation of molecular dipoles in its structure and can be considered as bioelectret, a sort of dielectric material exhibiting quasipermanent space charge and dipolar charge. Potentials are thought to occur when a number of collagen molecules are stressed in the same way displacing significant numbers of the charge carriers from the inside to the surface of the specimen.

Man made ceramics – Barium titanate, lead titanate, lead zirconate titanate, potassium niobate, lithium niobate, lithium tantalate , sodium tungstate etc.

## III. MECHANISM

In a piezoelectric crystal, the positive and negative electrical charges are separated, but symmetrically distributed, so that the crystal overall is electrically neutral. Each of these sides forms an electric dipole and dipoles near each other tend to be aligned in regions called Weiss domains. The domains are usually randomly oriented, but can be aligned during *poling* (not the same as magnetic poling), a process by which a strong electric field is applied across the material, usually at elevated temperatures. When a mechanical stress is applied, this symmetry is disturbed, and the charge asymmetry generates a voltage across the material. For example, a 1 cm<sup>3</sup> cube of quartz with 2 kN (500lbf) of correctly applied force can produce a voltage of 12,500 V. Piezoelectric materials also show the opposite effect, called converse piezoelectric effect, where the application of an electrical field creates mechanical deformation in the crystal.

Piezoelectricity is the combined effect of the electrical behavior of the material:

$$D = \epsilon E$$

(where D is the electric charge density displacement,  $\epsilon$  is the permittivity and E is the electric field strength.) and Hooke's law:

$$S = \frac{1}{\epsilon} T$$

The above two equations may be combined as coupled equations, of which the strain charge form is –

$$\{S\} = [s^E] \{T\} + [d^t] \{E\}$$

$$\{D\} = [d] \{T\} + [\epsilon^T] \{E\}$$

where [d] is the matrix for the direct piezoelectric effect and [d<sup>t</sup>] is the matrix for converse piezoelectric effect. The superscript E indicates a zero, or constant, electric field; the superscript T indicates a zero, or constant, stress field; and the superscript t stands for transposition of a matrix.

## IV. APPLICATIONS

### 4.1 Sensors

To detect sound, e.g. piezoelectric microphones (sound waves bend the piezoelectric material, creating a changing voltage) and piezoelectric pickups for electrically amplified guitars. A piezo sensor attached to the body of an instrument is known as a contact microphone.

Piezoelectric elements are also used in the generation of sonar waves.Piezoelectric microbalances are used as very sensitive chemical and biological sensors.

Piezoelectric transducers are used in electronic drum pads to detect the impact of the drummer's sticks.

### 4.2 Transducers

As very high voltages correspond to only tiny changes in the width of the crystal, this width can be changed with better-than-micrometre precision, making piezo crystals the most important tool for positioning objects with extreme accuracy.

Loudspeaker: Voltages are converted to mechanical movement of a piezoelectric polymer film.

Piezoelectric motor: piezoelectric elements apply a directional force to an axle, causing it to rotate. Due to the extremely small distances involved, the piezo motor is viewed as a high-precision replacement for the stepper motor.

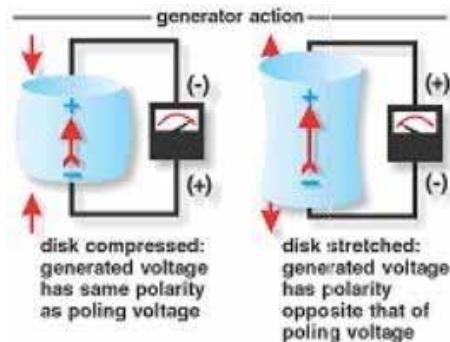
Piezoelectric elements can be used in laser mirror alignment, where their ability to move a large mass (the mirror mount) over microscopic distances is exploited to electronically align some laser mirrors. By precisely controlling the distance between mirrors, the laser electronics can accurately maintain optical conditions inside the laser cavity to optimize the beam output.

A related application is the acousto-optic modulator, a device that vibrates a mirror to give the light reflected off it a Doppler shift. This is useful for fine-tuning a laser's frequency.

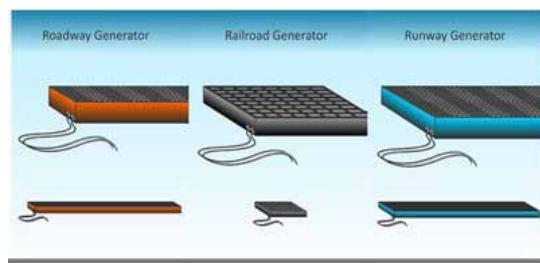
Atomic force microscopes and scanning tunneling microscopes employ converse piezoelectricity to keep the sensing needle close to the probe.

#### 4.3 Electric generation and Review of IPEG™

The piezoelectric ceramic element converts the mechanical energy of compression and tension into electrical energy which can be stored. Techniques used to make multilayer capacitors have been used to construct multilayer piezoelectric generators



An Israeli company Innowattech has developed unique breed of piezoelectric materials which are being used as an alternative for electrical energy generation. The system harvests mechanical energy imparted to roadways, railways and runways from passing vehicles and trains and converts it into green electricity. The system does not harm the efficiency of vehicles and trains. The piezoelectric generators are embedded between the superstructured layers, and usually covered with an asphalt topping. The size of sheets carrying these generators can vary from few centimeters to large surfaces per need.



#### Advantages of IPEG

- Energy is harvested that will otherwise go to waste.
- Harvesting potential is an average of 250 kWh per hour from a 1 km of a highway, one way, one lane.
- This much power is sufficient to power 500 homes.
- The building cost is much less than that of solar system with higher returns and efficiency.

#### V. REFERENCES

- [1] Walter Guyton Cady "Piezoelectricity"
- [2] Sergey V. Bogdanov "The origin of piezoelectric effect in pyroelectric crystals."
- [3] J. Bert "Vibration control using piezoelectric Transducers"
- [4] Innowattech "Energy Harvesting Systems"
- [5] Carmen Galassi "Piezoelectric materials"

# Generalization of Artificial Intelligence

<sup>1</sup>Ravita Chahar, Lecturer, <sup>2</sup>Komal Hooda- Lecturer, <sup>3</sup>Ashok Nain- Sr. SEO executive

<sup>1</sup>CIET, Rajpura, Dist. Patiala-140401, Punjab, <sup>2</sup>Vaish college of engg., Rohtak, <sup>3</sup>Shoogloo online marketing Pvt. Ltd, Delhi  
Er.ashoknain@gmail.com, Ravita.chahar@chitkara.edu.in, Komal.hooda@gmail.com

**Abstract-**We propose a developmental evaluation procedure for artificial intelligence that is based on two assumptions: that the Turing Test provides a sufficient subjective measure of artificial intelligence, and that any system capable of passing the Turing Test will necessarily incorporate behavioristic learning Techniques.

## I. INTRODUCTION

In 1950 Alan Turing considered the question "Can machines think?" Turing's answer to this question was to define the meaning of the term 'think' in terms of a conversational scenario, whereby if an interrogator cannot reliably distinguish between a machine and a human based solely on their conversational ability, then the machine could be said to be thinking. Originally called the imitation game, this procedure is nowadays referred to as the Turing Test. In this paper we shall demonstrate that the Turing Test is a sufficient evaluation criterion for artificial intelligence provided that the expectation level of the interrogator is set appropriately. We propose to achieve this by complementing the Turing Test with objective developmental evaluation. we begin with a definition of artificial intelligence, we continue with a discussion of the theory and methods which we believe are an essential prerequisite for the emergence of artificial intelligence and we conclude with our proposed evaluation procedure.

## II. THE TURING TEST

The Turing Test is an appealing measure of artificial intelligence because, as Turing himself writes, it : has the advantage of drawing a fairly sharp line between the physical and the Intellectual capacities of a man. The Loebner Contest, held annually since 1991, is an instantiation of the Turing Test. It bears out to our introductory remark that the Turing Test has been largely ignored by the field. In a recent thorough review of conversational systems, Hasida and Den emphasize the absurdity of performance in the Loebner Contest . They assert that since the Turing Test requires that systems "talk like people", and since no system currently meets this requirement, the ad-hoc techniques which the Loebner Contest subsequently encourages make little contribution to the advancement of dialog technology. Although we agree wholeheartedly that the Loebner Contest has failed to contribute to the advancement of artificial intelligence, we do believe that the Turing Test is appropriate evaluation criteria, and therefore our approach equates artificial intelligence with conversational skills. We further believe that engaging in domain-unrestricted conversation is the most critical evidence of intelligence.

### 2.1. Turing's Child Machine

Turing concluded his classic paper by theorizing on the design of a computer program, which would be capable of passing the Turing Test. He correctly anticipated the limitations of simulating adult level conversation, and proposed that : instead of trying to produce a program to simulate the

adult mind, why not rather try to produce one which simulates the child's? If this were then subjected to an appropriate course of education one would obtain the adult brain. Turing regarded language as an acquired skill, and recognized the importance of avoiding the hard wiring of the computer program wherever possible. He viewed language learning in a behavioristic light, and believed that the language channel, narrow though it may be, is sufficient to transmit the information, which the child machine requires in order to acquire language.

### 2.2. The Traditional Approach

Contrary to Turing's prediction that at about the turn of the millennium computer programs will participate in the Turing Test so effectively that an average interrogator will have no more than a seventy percent chance of making the right identification after five minutes of questioning, no true conversational systems have yet been produced, and none has passed an unrestricted Turing Test. This may be due in part to the fact that Turing's idea of the child machine has remained unexplored-the traditional approach to conversational system design has been to equate language with knowledge, and to hard-wire rules for the generation of conversations. This approach has failed to produce anything more sophisticated than domain-restricted dialog systems which lack the kind of flexibility, openness and capacity to learn that are the very essence of human intelligence. It is our thesis that true conversational abilities are more easily obtainable via the currently neglected behavioristic approach.

## III. VERBAL BEHAVIOR

Behaviorism focuses on the observable and measurable aspects of behavior. Behaviorists search for observable environmental conditions, known as stimuli, that co-occur with and predict the appearance of specific behavior, known as responses. This is not to say that behaviorists deny the existence of internal mechanisms; they do recognize that studying the physiological basis is necessary for a better understanding of behavior. They favour a functional rather than a structural approach, focusing on the function of language, the stimuli that evoke verbal behavior and the consequences of language performance. We believe this to be the right approach for the generation of artificial intelligence. The processes of forming such associations are known as classical conditioning and operant conditioning.

### 3.1. Classical Conditioning

Classical conditioning accounts for the associations formed between arbitrary verbal stimuli and internal responses or reflexive behavior. In classical conditioning, for example, the word "milk" is learned when the infant's mother says "milk" before or after feeding, and this word becomes associated with the primary stimulus (the milk itself) to eventually elicit a response similar to the response to the milk.

Words stimulate each other and classical conditioning accounts for the interrelationship of words and word meanings.

### 3.2. Operant Conditioning

Operant conditioning is used to account for changes in voluntary, non-reflexive behavior that arise due to environmental consequences contingent upon that behavior. Operant conditioning is used to account for the productive side of language acquisition. Imitation is another important factor in language acquisition because it allows the laborious shaping of each and every verbal response to be avoided. The learning principle of reinforcement is therefore taken to play a major role in the process of language acquisition, and is the one we believe should be used in creating artificial intelligence.

## IV. THE DEVELOPMENTAL MODEL

We maintain that a behavioristic developmental approach could yield breakthrough results in the creation of artificial intelligence. Programs can be granted the capacity to imitate, to extract implicit rules and to learn from experience, and can be instilled with a drive to constantly improve their performance. Human language acquisition milestones are both quantifiable and descriptive, and any system that aims to be conversational can be evaluated as to its analogical human chronological age. Such systems could therefore be assigned an age or a maturity level beside their binary Turing Test assessment of 'intelligent' or 'not intelligent'.

## V. LANGUAGE MODELING

We are interested in programming a computer to acquire and use language in a way analogous to the behavioristic theory of child language acquisition. In fact, we believe that fairly general information processing mechanisms may aid the acquisition of language by allowing a simple language model, such as a low {order Markov model, to bootstrap itself with higher-level structure.

### 5.1 Markov Modeling

Claude Shannon, the father of Information Theory, was generating quasi-English text using Markov models in the late 1940's. Such models are able to predict which words are likely to follow a given finite context of words, and this prediction is based on a statistical analysis of observed text. Using Markov models as part of a computational language acquisition system allows us to minimize the number of assumptions we make about the language itself, and to eradicate language-specific hard-wiring of rules and knowledge. Information theoretic measures may be applied to Markov models to yield analogous behavior, and more sophisticated techniques can model the case where long-distance dependencies exist between the stimulus and the response.

### 5.2. Finding Higher-Level Structure

Information theoretic measures may be applied to the predictions made by a Markov model in order to find sequences of symbols and classes of symbols, which constitute higher-level structure. Markov model inferred from English text can easily segment the text into words, while a word-level Markov model inferred from English text may be used to 'discover' syntactic categories. Although each level of the hierarchy is formed in a purely bottom-up fashion from the

data supplied to it by the level below, the fact that each model provides a top-down view with respect to the models below it allows a feedback process to be applied. It is our belief that combining this approach with positive and negative reinforcement is a sensible way of realizing Turing's vision of a child machine.

## VI. EVALUATION PROCEDURE

Our proposal is to measure the performance of conversational systems via both subjective methods and objective developmental metrics.

### 6.1. Objective Developmental Metrics

The ability to converse is complex, continuous and incremental in nature, and thus we propose to complement our subjective impression of intelligence with objective incremental metrics. Examples of such metrics, which increase quantitatively with age, are:

- Vocabulary size: The number of different words spoken.
- *Mean length of utterance*: The mean number of word morphemes spoken per utterance.
- *Response types*: The ability to provide an appropriate sentence form with relevant content in a given conversational context, and the variety of forms used.
- *Degree of syntactic complexity*: For example, the ability to use embedding to make connections between sentences, and to convey ideas.

### 6.2. The Subjective Component

We do not claim that objective evaluation should take precedence over subjective evaluation; We believe that the subjective evaluation of artificial intelligence is best performed within the framework of the Turing Test. Using objective metrics to evaluate maturity level will help set up the right expectation level to enable a valid subjective judgement to be made. Given that subjective impression is at the heart of the perception of intelligence, the constant feedback from the subjective evaluation to the objective one will eventually contribute to an optimal evaluation system for perceiving intelligence.

## VII. CONCLUSION

We submit that a developmental approach is a prerequisite to the emergence of intelligent lingual behavior and to the assessment thereof. This approach will help establish standards that are in line with Turing's understanding of intelligence, and will enable evaluation across systems. We predict that the current paradigm shift in understanding the concepts of AI and natural language will result in the development of groundbreaking technologies, which will pass the Turing Test within the next ten years.

## VIII. REFERENCES

- [1]. A.M. Turing, "Computing machinery and intelligence," in Collected works of A.M. Turing: Mechanical Intelligence, D.C. Ince, Ed., chapter 5, pp. 133{160. Elsevier Science Publishers, 1992.
- [2]. Stuart M. Shieber, "Lessons from a restricted Turing test," Available at the Computation and Language e-print server as cmp-lg/9404002., 1994.
- [3]. K. Hasida and Y. Den, "A synthetic evaluation of dialogue systems," in Machine Conversations, Yorick Wilks, Ed. Kluwer Academic Publishers, 1999.
- [4]. J.B. Gleason, The Development of Language, Charles E. Merrill Publishing Company, 1985.
- [5]. Goren, G. Tucker, and G.M. Ginsberg, "Language dysfunction in schizophrenia," European Journal of Disorders of Communication, vol. 31, no. 2, pp. 467{482, 1996.

# Network on Chip Architectures using VLSI

Student- Abhishek Acharya, Student-Sukhbir Singh Kinha, Student-Vikram Singh  
Department of Electronics & Communication Engineering, NITTTR, Chandigarh, [acharya\\_g@rediffmail.com](mailto:acharya_g@rediffmail.com)

**Abstract-** The objective of this paper is to explore the networking community to the concept of network-on-chip (NoC), an emerging field of study within the VLSI realm, in which networking principles play a significant role, and new network architectures are in and. Networking researchers will find new challenges in exploring solutions to familiar problems such as network design, routing, and quality-of-service, in unfamiliar settings under new constraints. It can be defined as the *layered-stack approach* to the design of the on-chip inter-core communication networks. In a NoC system, modules such as processor cores, memories and specialized IP blocks exchange data using a network as a "public transportation" sub-system for the information traffic. An NoC is constructed from multiple point-to-point data links interconnected by switches, such that messages can be relayed from any source module to any destination module over several links, by making routing decisions at the switches. An NoC is similar to a modern telecommunications network, using digital bit-packet switching over multiplexed links. Although packet-switching is sometimes claimed as necessity for a NoC, there are several NoC proposals utilizing circuit-switching techniques. This definition based on routers is usually interpreted so that a single shared bus, a single crossbar switch or a point-to-point network are not NoCs but practically all other topologies are. This is somewhat confusing since all above mentioned are networks (they enable communication between two or more devices) but they are not considered as network-on-chips. Note that some articles erroneously use NoC as a synonym for mesh topology although NoC paradigm does not dictate the topology. Likewise, the regularity of topology is sometimes considered as a requirement which is, obviously, not the case in research concentrating on "application-specific NoC topology synthesis".

The wires in the links of the NoC are shared by many signals. A high level of *parallelism* is achieved, because all links in the NoC can operate simultaneously on different data packets. Therefore, as the complexity of integrated systems keeps growing, an NoC provides enhanced performance (such as throughput) and *scalability* in comparison with previous communication architectures (e.g., dedicated point-to-point signal wires, shared buses, or segmented buses with bridges). Of course, the algorithms must be designed in such a way that they offer large parallelism and can hence utilize the potential of NoC.

Network on Chip links can reduce the complexity of designing wires for predictable speed, power, noise, reliability, etc., thanks to their regular, well controlled structure. From a system design viewpoint, with the advent of multi-core processor systems, a network is a natural architectural choice.

## I. INTRODUCTION

As VLSI technology becomes smaller, and the number of modules on a chip multiplies, on-chip communication solutions are evolving in order to support the new inter-module communication demands. Traditional solutions, which were based on a combination of shared-buses and dedicated module to- module wires, have hit their scalability limit, and are no longer adequate for sub-micron technologies. Current chip designs incorporate more complex multilayered and segmented

interconnection buses. More recently, chip architects have begun employing on-chip network-like solutions. We believe that the considerations that have driven data communication from shared buses to packet-switching networks (spatial reuse, multi-hop routing, flow and congestion control, and standard interfaces for design reuse, etc.) will inevitably drive VLSI designers to use these principles in on-chip interconnects. In other words, we can expect the future chip design to incorporate a full-fledged network-on-a-chip (NoC), consisting of a collection of links and routers and a new set of protocols that govern their operation.

Several trends have forced evolutions of systems architectures, in turn driving evolutions of required busses. These trends are:

- Application convergence: The mixing of various traffic types in the same SoC design (Video, Communication, Computing and etc.). These traffic types, although very different in nature, for example from the Quality of Service point of view, must now share resources that were assumed to be "private" and handcrafted to the particular traffic in previous designs.
- Moore's law is driving the integration of many IP Blocks in a single chip. This is an enabler to application convergence, but also allows entirely new approaches (parallel processing on a chip using many small processors) or simply allows SoCs to process more data streams (such as communication channels)
- Consequences of silicon process evolutions between generations: Gates cost relatively less than wires, both from an area and performance perspective, than a few years ago.
- Time-To-Market pressures are driving most designs to make heavy use of synthesizable RTL rather than manual layout, in turn restricting the choice of available implementation solutions to fit a bus architecture into a design flow.

These trends have driven of the evolution of many new bus architectures. These include the introduction of split and retry techniques, removal of tri-state buffers and multi-phase-clocks, introduction of pipelining, and various attempts to define standard communication sockets.

However, history has shown that there are conflicting tradeoffs between compatibility requirements, driven by IP blocks reuse strategies, and the introduction of the necessary bus evolutions driven by technology changes : In many cases, introducing new features has required many changes in the bus implementation, but more importantly in the bus interfaces, with major impacts on IP reusability and new IP design.

Busses do not decouple the activities generally classified as transaction, transport and physical layer behaviors. This is the key reason they cannot adapt to changes in the system architecture or take advantage of the rapid advances in silicon process technology.

Consequently, changes to bus physical implementation can have serious ripple effects upon the implementations of higher-level bus behaviors. Replacing tri-state techniques with multiplexers has had little effect upon the transaction levels. Conversely, the introduction of flexible pipelining to ease timing closure has massive effects on all bus architectures up through the transaction level.

Similarly, system architecture changes may require new transaction types or transaction characteristics. Recently, such new transaction types as exclusive accesses have been introduced near simultaneously within OCP2.0 and AMBA AXI socket standards. Out-of-order response capability is another example. Unfortunately, such evolutions typically impact the intended bus architectures down to the physical layer, if only by addition of new wires or op-codes. Thus, the bus implementation must be redesigned.

As a consequence, bus architectures cannot closely follow process evolution, nor system architecture evolution. The bus architects must always make compromises between the various driving forces, and resist change as much as possible.

In the data communications space, LANs & WANs have successfully dealt with similar problems by employing a layered architecture. By relying on the OSI model, upper and lower layer protocols have independently evolved in response to advancing transmission technology and transaction level services. The decoupling of communication layers using the OSI model has successfully driven commercial network architectures, and enabled networks to follow very closely both physical layer evolutions (from the Ethernet multi-master coaxial cable to twisted pairs, ADSL, fiber optics, wireless..) and transaction level evolutions (TCP, UDP, streaming voice/video data). This has produced incredible flexibility at the application level (web browsing, peer-to-peer, secure web commerce, instant messaging, etc.), while maintaining upward compatibility (old-style 10Mb/s or even 1Mb/s Ethernet devices are still commonly connected to LANs).

Following the same trends, networks have started to replace busses in much smaller systems: PCI-Express is a network-on-a board, replacing the PCI board-level bus. Replacement of SoC busses by NoCs will follow the same path, when the economics prove that the NoC either:

- Reduces SoC manufacturing cost
- Increases SoC performance
- Reduces SoC time to market and/or NRE
- Reduces SoC time to volume
- Reduces SoC design risk

In each case, if all other criteria are equal or better NoC will replace SoC busses.

This paper describes how NoC architecture affects these economic criteria, focusing on performance and manufacturing cost comparisons with traditional style busses. The other criteria mostly depend on the maturity of tools supporting the NoC architecture and will be addressed separately.

## II.NOC ARCHITECTURE

The advanced Network-on-Chip developed by Arteris employs system-level network techniques to solve on chip traffic transport and management challenges. As discussed in

the previous section and shown in Figure 1, synchronous bus limitations lead to system segmentation and tiered or layered bus architectures.

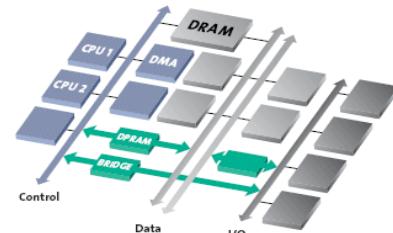


Figure 1: Traditional synchronous bus

Contrast this with the Arteris approach illustrated in Figure 2. The NoC is a homogeneous, scalable switch fabric network. This switch fabric forms the core of the NoC technology and transports multi-purpose data packets within complex, IP-laden SoCs. Key characteristics of this architecture are:

- Layered and scalable architecture
- Flexible and user-defined network topology.
- Point-to-point connections and a Globally Asynchronous Locally Synchronous (GALS) implementation decouple the IP blocks

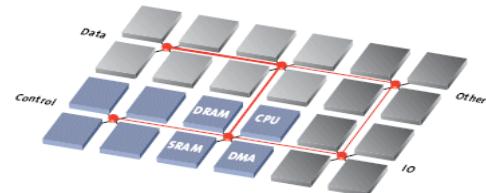


Figure 2: Arteris switch fabric network

## III. NOC LAYERS

IP blocks communicate over the NoC using a three layered communication scheme (Figure 3), referred to as the Transaction, Transport, and Physical layers

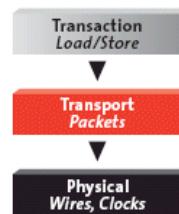


Figure 3 : Arteris NoC layers

The Transaction layer defines the communication primitives available to interconnected IP blocks. Special NoC Interface Units (NIUs), located at the NoC periphery, provide transaction-layer services to IP blocks with which they are paired. This is analogous, in data communications networks, to Network Interface Cards that source/sink information to the LAN/WAN media. The transaction layer defines how information is exchanged between NIUs to implement a particular transaction. For example, a NoC transaction is typically made of a request from a master NIU to a slave NIU, and a response from the slave to the master. However, the transaction layer leaves the implementation details of the exchange to the transport and physical layer. NIUs that bridge

the NoC to an external protocol (such as AHB) translate transactions between the two protocols, tracking transaction state on both sides. For compatibility with existing bus protocols, Arteris NoC implements traditional address-based Load/ Store transactions, with their usual variants including incrementing, streaming, wrapping bursts, and so forth. It also implements special transactions that allow sideband communication between IP Blocks.

The Transport layer defines rules that apply as packets are routed through the switch fabric. Very little of the information contained within the packet is needed to actually transport the packet. The packet format is very flexible and easily accommodates changes at transaction level without impacting transport level. For example, packets can include byte enables, parity information, or user information depending on the actual application requirements, without altering packet transport, nor physical transport.

A single NoC typically utilizes a fixed packet format that matches the complete set of application requirements. However, multiple NoCs using different packet formats can be bridged together using translation units.

The Transport Layer may be optimized to application needs. For example, wormhole packet handling decreases latency and storage but might lead to lower system performance when crossing local throughput boundaries, while store-and forward handling has the opposite characteristics. The Arteris architecture allows optimizations to be made locally. Wormhole routing is typically used within synchronous domains in order to minimize latency, but some amount of store-and forward is used when crossing clock domains.

The Physical layer defines how packets are physically transmitted over an interface, much like Ethernet defines 10Mb/s, 1Gb/s, etc. physical interfaces. As explained above, protocol layering allows multiple physical interface types to coexist without compromising the upper layers. Thus, NoC links between switches can be optimized with respect to bandwidth, cost, data integrity, and even off-chip capabilities, without impacting the transport and transaction layers. In addition, Arteris has defined a special physical interface that allows independent hardening of physical cores, and then connection of those cores together, regardless of each core clock speed and physical distance within the cores (within reasonable limits guaranteeing signal integrity). This enables true hierarchical physical design practices.

A summary of the mapping of the protocol layers into NoC design units is illustrated by the following figure

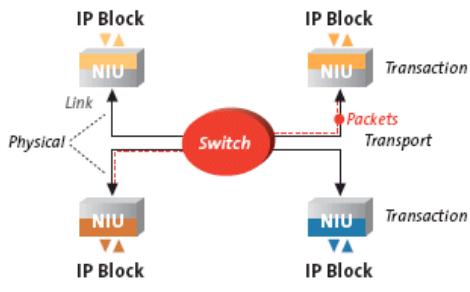


Figure 4: NoC Layer mapping summary

#### IV. NOC LAYERED APPROACH BENEFITS

A summary of the benefits of this layered approach are:

- Separate optimizations of transaction and physical layers. The transaction layer is mostly influenced by application requirements, while the physical layer is mostly influenced by Silicon process characteristics. Thus the layered architecture enables independent optimization on both sides. A typical physical optimization used within NoC is the transport of various types of cells (header and data) over shared wires, thereby minimizing the number of wires and gates.
- Scalability. Since the switch fabric deals only with packet transport, it can handle an unlimited number of simultaneous outstanding transactions (e.g., requests awaiting responses). Conversely, NIUs deal with transactions, their outstanding transaction capacity must fit the performance requirements of the IP Block or subsystems that they service. However, this is a local performance adjustment in each NIU that has no influence on the setup and performance of the switch fabric.
- Aggregate throughput. Throughput can be increased on a particular path by choosing the appropriate physical transport, up to even allocating several physical links for a logical path. Because the switch fabric does not store transaction state, aggregate throughput simply scales with the operating frequency, number and width of switches and links between them, or more generally with the switch fabric topology.
- Quality of Service. Transport rules allow traffic with specific real-time requirements to be isolated from best-effort traffic. It also allows large data packets to be interrupted by higher priority packets transparently to the transaction layer.
- Timing convergence. Transaction and Transport layers have no notion of a clock: the clocking scheme is an implementation choice of the physical layer. Arteris first implementation uses a GALS approach: NoC units are implemented in traditional synchronous design style (a unit being for example a switch or an NIU), sets of units can either share a common clock or have independent clocks. In the latter case, special links between clock domains provide clock resynchronization at the physical layer, without impacting transport or transaction layers. This approach enables the NoC to span a SoC containing many IP Blocks or groups of blocks with completely independent clock domains, reducing the timing convergence constraints during back-end physical design steps.
- Easier verification. Layering fits naturally into a divide-and-conquer design & verification strategy. For example, major portions of the verification effort need only concern itself with transport level rules since most switch fabric behavior may be verified independent of transaction states. Complex, state-rich verification problems are simplified to the verification of single NIUs; the layered protocol ensures interoperability between the NIUs and transport units.
- Customizability. User-specific information can be easily added to packets and transported between NIUs. Custom-designed NoC units may make use of such information, for example “firewalls” can be designed that make use of predefined information to shield specific targets from unauthorized transactions. In this case, and many others, such

application-specific design would only interact with the transport level and not even require the custom module designer to understand the transaction level.

#### V. NETWORK ON CHIP: PITFALLS

In spite of the obvious advantages, a layered strategy to on-chip communication must not model itself too closely on data communications networks.

In data communication networks the transport medium (i.e., optical fiber) is much more costly than the transmitter and receiver hardware and often employs “wave pipelining” (i.e. multiple symbols on the same wire in the case of fiber optics or controlled impedance wires). Inside the SoC the relative cost and performance of wires and gates is different and wave pipelining is too difficult to control. As a consequence, NoCs will not, at least for the foreseeable future, serialize data over single wires, but find an optimal trade-off between clock rate (100MHz to 1GHz) and number of data wires (16, 32, 64...) for a given throughput.

Further illustrating the contrast, data communications networks tend to be focused on meeting bandwidth related quality of service requirements, while SoC applications also focus on latency constraints.

Moreover, a direct on-chip implementation of traditional network architectures would lead to significant area and latency overheads. For example, the packet dropping and retry mechanisms that are part of TCP/IP flow control require significant data storage and complex software control. The resulting latency would be prohibitive for most SoCs.

Designing a NoC architecture that excels in all domains compared to busses requires a constant focus on appropriate trade-offs.

#### VI. COMPARISON WITH TRADITIONAL BUSSES

In this section we will use an example to quantify some advantages of the NoC approach over traditional busses. The challenge is that comparisons depend strongly on the actual SoC requirements. We will first describe an example we hope is general enough that we may apply the results more broadly to a class of SoCs.

The “design example” is comprised of 72 IP Blocks, 36 masters and 36 slaves (the ratio between slaves and masters does not really matter, but the slaves usually define the upper limit of system throughput) The total number of IP Blocks implies a hierarchical interconnect scheme; we assume that the IP Blocks are divided in 9 clusters of 8 IP Blocks each.

Within each cluster, IP blocks are locally connected using a local bus or a switch, and the local busses or switches are themselves connected together at the SoC level.

With a regular, hierarchical floor plan, the two architectures look somewhat like Figure 5

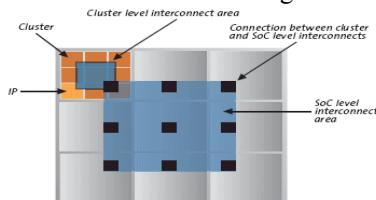


Figure 5 : Hierarchical floor plan of generic design

The SoC is assumed to be 9mm square, clusters are 3mm square and the IP blocks are each about 1mm square. Let us also assume a 90nm process technology and associated standard cell library where an unloaded gate delay is 60pS, and DFF traversal time (setup + hold) is 0.3nS. Based on electrical simulations, we also can estimate that a properly buffered wire running all along the 9mm of the design would have a propagation delay of at least 2nS. According to the chosen structure it then takes approximately 220pS for a wire transition to propagate across an IP block, 660pS across a cluster.

In the bus case, cluster-level busses connect to 4 master IP Blocks, 4 slave IP Blocks, and the SoC level bus, which adds a master and slave port to each cluster-level bus. Thus, each cluster-level bus has 5 master and 5 slave ports, and the SoC-level bus has 9 master and 9 slave ports. The length of wire necessary to connect the 9 ports of the top-level bus is at least the half-perimeter of the SoC-level interconnect area, approximately between 2 and 4 cluster sides (i.e., between 6 and 12 mm) depending on the actual position of the connection ports to the cluster busses.

Similarly in the NoC case, two 5x5 (5 inputs, 5 outputs) switches are required in each cluster, one to handle requests between the cluster IP Blocks and the SoC-level switch, and another identical one managing responses. The SoC-level switches are 9x9. However since the NoC uses point-to-point connections, the maximum length wires between the center of the SoC, where the 9x9 switch resides, and the ports to the cluster-level switches, is at worst only half of the equivalent bus length, i.e. 1 to 2 cluster sides or between 3 and 6 mm. Actual SoC designs differ from this generic example, but using it to elaborate comparison numbers and correlating these to commonly reported numbers on actual SoC designs provides valuable insight about the superior fundamentals of NoC.

#### VII. MAXIMUM NOC FREQUENCY

In the NoC case, point-to-point links and GALS techniques greatly simplify the timing convergence problem at the SoC level. Synchronous clock domains typically need only span individual clusters.

Arteris has demonstrated a mesochronous link technique operable at 800MHz for an unlimited link length, at the expense of some latency which will be accounted for later in this analysis. Thus only the switches and clusters limit the maximum frequency of our generic design.

Within the synchronous clusters, point-to-point transport does not exceed 2mm. Arteris has taken care to optimize the framing signals accompanying the packets in order to reduce to 3 gates or less the decision logic to latch the data after its transport. Thus transport time is no more than  $2*2/9+3*0.06=0.6$ ns. Within a cluster, skew is more easily controlled than at SoC level and is typically about .3ns. Taking into account the DFF, we compute a maximum operating frequency of  $1/(0.6+0.3+0.3)=800$ MHz. But in fact, this estimate is rather pessimistic, because within a synchronous cluster the switch pipeline stages tend to be distributed (this may be enforced by the physical design synthesis tools) such that there should never be cluster-level wires spanning 2mm.

Experiments using a standard physical synthesis tool flow show that proper pipelining of the switches enables NoC operating frequencies of 800Mhz for 3x3mm clusters. The very simple packet transport and carefully devised packet framing signals of Arteris NoC architecture enable such pipelining (most pipelining stages are optional in order to save latency cycles in the case that high operating frequencies are not required).

### VIII. PEAK THROUGHPUT ESTIMATION

For the remainder of this analysis we assume frequencies of 250MHz for the bus-based architecture, and 750MHz for the NoC-based. This relationship scales. For example, a set of implementations employing limited pipelining might run at 166MHz vs. 500 MHz.

Assuming all busses are 4-byte data wide, the aggregate throughput of the entire SoC (9 clusters) is  $250*4*9 = 9\text{GB/s}$ , assuming one transfer at the same time per cluster.

The NoC approach uses crossbar switches with 4-byte links. Aggregate peak throughput is limited by the masters or slaves send/receive data. Here however we must take into account two factors:

- Request and response networks are separate, and in the best case responses to LOAD transactions flow at the same time as WRITE data, leading to a potential 2x increase (some busses also have separate data channels for read and write and this 2x factor then disappears).
- The NoC uses packetized data. Packet headers share the same wires as the data payload, so the number of wires per link is less than 40. The relative overhead of packet headers compared to transaction payload depends on the average payload size and transaction type. If we assume an average payload size of 16 bytes the packetization overhead is much less than the payload itself: as a worst case we assume 50% payload efficiency. If all 36 initiators issue a transaction simultaneously, the peak throughput is :  $750*4*2*50%*36 > 100 \text{ GB/s}$ . The NoC has a potential 10x throughput advantage over the bus-based approach. The actual ratio may be lower if multi-layered busses are used at the cluster level. Because multi-layers are similar to crossbars, the added complexity could limit the target frequency.

### IX. MINIMUM LATENCY

Latency is a difficult comparison criterion, because it depends on many application-specific factors: are we interested in minimum latency on a few critical paths or statistical latency over the entire set of dataflow? The overall system-level SoC performance usually depends only on a few latency-sensitive data flows (typically, processor cache refills) while for most other data flows only achievable bandwidth will matter. But even for the latter dataflow, latency does matter in the sense that high average latencies require intermediate storage buffers to maintain throughput, potentially leading to area overhead. Let us first analyze minimum latency for our architectures. We assume that all slave IP blocks have a typical latency of 2 clock cycles @250MHz, i.e. 5nS. This translates into 6 clock cycles @750MHz (for comparison fairness we assume that IP Blocks run at the same speed as in the bus case).

### X. SYSTEM THROUGHPUT

Assuming that the inevitable retry or busy cycles limit the inter-cluster bus to 50% efficiency it can handle  $250*4*50\% = 0.5\text{GB/s}$ . This is the system bottleneck and limits overall traffic to 2.5GB/s, each of the 9 clusters having a local traffic of  $2500/9 = 277\text{MB/s}$ , far from the potential peak. Inter-cluster peak traffic could be increased, typically by making the bus wider. Doubling inter-cluster bus width would increase the total average traffic up to 5GB/s but at the expense of area, congestion, and inter-cluster latency. Similarly, a lower ratio of inter-cluster traffic, for example 10% instead of 20%, also leads to 5GB/s total system throughput. For reasonable traffic patterns the achievable system throughput is thus limited to much lower sustainable rates than theoretical peak throughput, because the backbone performance does not scale with traffic complexity requirements.

Within the NoC architecture the inter-cluster crossbar switches are less limiting to system-level traffic. Assuming

- 20% inter-cluster traffic,
- 4-byte wide links
- 50% efficiency resulting from packetization and conflicts between packets targeting the same cluster
- separate request and response paths

The achievable system throughput is  $(750*4*9*2*50\%)/20\% = 130\text{GB/s}$ . This is higher than the peak throughput that the initiators and targets can handle, clearly illustrating the intrinsic scalability of the hierarchical NoC approach.

### XI. AREA AND POWER COMPARISON

Traditional busses have been perceived as very area efficient because of their shared nature. As we already discussed, this shared nature drives both operation frequency and system performance scalability down. Some techniques have been introduced in recent busses to fix these issues:

- Pipelining added to sustain bus frequencies: with busses having typically more than 100 wires, each pipeline stage costs at least 1Kgates. Moreover, to reach the highest frequencies one or several pipeline stages are needed at each initiator and target interface (for example: one at each initiator for arbitration and one before issuing data to the bus, one at each target for address decode and issuing data to the target, and similar retiming on the response path). For our cluster-level bus, this leads to  $2*100*4*2*10 = 16\text{ K gates}$ , for the inter-cluster bus  $2*100*9*2*10 = 36\text{K gates}$ , totaling 180K gates just for pipelining. Gate count increases further if the data bus size exceeds 32 bits.
- FIFOs inserted to deal with arbitration latency: Even worse, to sustain throughput as latency grows, buffers must be inserted in the bridges between the inter-cluster and cluster-level busses. When a transaction waits to be granted arbitration to the inter-cluster bus, pushing it into this buffer frees the originating cluster-level bus, allowing it to be used by a cluster-level transaction. Without such buffers, inter-cluster congestion dramatically impacts cluster-level performance. For these buffers to be efficient, they should contain the average number of outstanding transactions derived from average

latencies. In our case each inter-cluster bus initiator requires 10% of its bandwidth, i.e. one 16 byte (4 cycles bus occupation) transaction every 40 cycles, while we have seen an average latency also on the order of 40 cycles, peaking at more than 100. Thus, to limit blocking of cluster-level busses, each inter-cluster bus initiator should typically be able to store two stores and two load 4-byte transactions with their addresses, e.g.  $2 \times 4 \times 100 \times 10 = 8\text{K}$  gates per initiator, 72K gates total buffering.

Pipelining and buffering add up to 250K gates. Adding bus MUX, arbiters, address decoders, and all the state information necessary to track the transaction retries within each bus, the total gate count for a system throughput of less than 10GB/s is higher than 400K gates. The NoC implementation uses two 4x5 32-bit wide switches in each cluster. Including three levels of pipelining, this amounts to about 8k gates. Because arbitration latency is much smaller than for busses, intermediate buffers are not needed in the switch fabric. The two inter-cluster switches are approximately 30K gates each, for a total of  $9 \times 8 \times 2 + 2 \times 30 = 210\text{K}$  gates. Thus for a smaller gate count, the NoC is able to handle an order of magnitude more aggregate traffic - up to 100GB/s.

## XII. POWER MANAGEMENT

The modular, point-to-point NoC approach enables several power management techniques that are difficult to implement with traditional busses:

- The GALS paradigm allows subsystems (potentially as small as a single IP block) to always be clocked at the lowest frequency compatible with the application requirement.
- The NoC can also be partitioned into sub-networks that can be independently powered-off when the application does not require them, reducing static power consumption. Quantification of power consumption improvements due to these techniques is too closely tied to the application to be estimated on our generic example.

## XIII. COMPARISON TO CROSSBARS

In the previous section the NoC was compared and contrasted with traditional bus structures. We pointed out that system level throughput and latency may be improved with bus based architectures by employing pipelined crossbars or multilayer busses. However, because traditional crossbars still mix transaction, transport and physical layers in a way similar to traditional busses, they present only partial solutions. They continue to suffer the following:

**Scalability:** To route responses, a traditional crossbar must either store some information about each outstanding transaction, or add such information (typically, a return port number) to each request before it reaches the target and rely on the target to send it back. This can severely limit the number of outstanding transactions and inhibit one's ability to cascade crossbars. Conversely, Arteris' switches do not store transaction state, and packet routing information is assigned and managed by the NIUs and is invisible to the IP blocks. This results in a scalable switch fabric able to support an unlimited number of outstanding transactions.

- **IP block reusability:** Traditional crossbars handle a single given protocol and do not allow mixing IP blocks with different protocol flavors, data widths or clock rates. Conversely, the Arteris transaction layer supports mixing IP blocks designed to major socket and bus standards (such as AHB, OCP, AXI), while packet-based transport allows mixing data widths and clock rates.

- **Maximum frequency, wire congestion and area:** Crossbars do not isolate transaction handling from transport. Crossbar control logic is complex, data paths are heavily loaded and very wide (address, data read, data write, response...), and SoC-level timing convergence is difficult to achieve. These factors limit the maximum operating frequency. Conversely, within the NoC the packetization step leads to fewer data path wires and simpler transport logic. Together with a Globally Asynchronous Locally Synchronous implementation, the result is a smaller and less congested switch fabric running at higher frequency.

Common crossbars also lack additional services that Arteris NoC offers and which are outside of the scope of this whitepaper, such as error logging, runtime reprogrammable features, and so forth.

## XIV. SUMMARY AND CONCLUSION

Criteria	Bus	NoC
Max Frequency	250 MHz	> 750 MHz
Peak Throughput	9 GB/s (more if wider bus)	100 GB/s
Cluster min latency	6 Cycles @250MHz	6 Cycles @250MHz
Inter-cluster min latency	14-18 Cycles @250MHz	12 Cycles @250MHz
System Throughput	5 GB/s (more if wider bus)	100 GB/s
Average arbitration latency	42 Cycles @250MHz	2 Cycles @250MHz
Gate count	400K	210K
Dynamic Power	Smaller for NoC, see discussion in 3.5.2	
Static Power	Smaller for NoC (proportional to gate count)	

## XV. REFERENCES

- [1]. R. Ho, K. Mai, and M. Horowitz. Managing wire scaling: A circuit perspective. In IEEE Interconnect Technology Conference, June 2003.
- [2]. IBM. The coreconnect bus architecture, 1999. [3] S. Kumar, A. Jantsch, J.-P. Soininen, M. Forsell, M. Millberg, J. Oberg, K. Tiensyrija, and A. Hemani. A network on chip architecture and design methodology. In ISVLSI, 2002.
- [3]. N. Magen, A. Kolodny, U. Weiser, and N. Shamir. Interconnect-power dissipation in a microprocessor. In SLIP'04, Feb. 2004.
- [4]. Morgenshtern, I. Cidon, A. Kolodny, and R. Ginosar. Comparative analysis of serial vs. parallel links in NoC. In International Symposium on System-on-Chip, pages 185–188, Nov. 2004.
- [5]. Morgenshtern, I. Cidon, A. Kolodny, and R. Ginosar. Low-leakage repeaters for NoC
- [6]. Interconnects. In ISCAS, pages 600–603, May 2005.
- [7]. L. Peterson, S. Shenker, and J. Turner. Overcoming the Internet impasse through virtualization. In HotNets III, Nov. 2004.
- [8]. Radulescu and K. Goossens. Communication services for networks on chip, pages 193–213. Marcel Dekker, 2004.
- [9]. V. Raghunathany, M. B. Srivastavay, and R. K. Gupta. A survey of techniques for energy efficient on-chip communication. In DAC, pages 900–905, 2003.
- [10]. E. Rijkem, K. Goossens, and P. Wielage. A router architecture for networks on silicon. In Progress 2001, 2nd Workshop on Embedded Systems, Veldhoven, the Netherlands, Oct. 2001.

# Dataflow Graph Generation in LCC Compiler: A Compiler for Embedded Systems

Piyush Agnihotri, Phunstog Toldhan, Manish kr. Yadav, Kumar Sambhav Pandey  
Dept. Of Computer Science, NIT hamirpur, [piyush.cse26@nitham.ac.in](mailto:piyush.cse26@nitham.ac.in)

**Abstract—** In order to convert High Level Language (HLL) into hardware, Dataflow Graph (DFG) is a fundamental element to be used. We propose in this paper a method for generating dataflow graphs in lcc compiler. Dataflow graph is powerful intermediate representation. Which can be executed on to hardware of embedded system based on Dataflow model of computation? we also applied Optimizations, both classical and new, transform the graph through graph rewriting rules prior to code generation

## I. INTRODUCTION

In the sense of a compiler being a person who compiles, then the term compiler has been known since the 1300's. Our more usual notion of a compiler, a software tool that translates a program from one form to another form, has existed for little over half a century. For a definition of what a compiler is, we refer to Aho *et al* [1]: A compiler is a program that reads a program written in one language, the source language, and translates it into an equivalent program in another language, the target language. Early compilers were simple machines that did little more than macro expansion or direct translation; these exist today as assemblers, translating assembly language (*e.g.*, .add r3, r1, r2.) into machine code (.0xE0813002. in ARM code). Over time, the capabilities of compilers have grown to match the size of programs being written. However, Proebsting [2] suggests that while processors may be getting faster at the rate originally proposed by Moore [3], compilers are not keeping pace with them, and indeed seem to be an order of magnitude behind. When we say *not* keeping pace, we mean that, where processors have been doubling in capability every eighteen months or so, the same doubling of capability in compilers seems to take around eighteen years! Which then leads to the question of what we mean by the *capability* of a compiler? Specifically, it is a measure of the power of the compiler to analyze the source program, and translate it into a target program that has the same meaning (does the same thing) but does it in fewer processor clock cycles (is faster) or in fewer target instructions (is smaller) than a naive compiler. Improving the power of an optimizing compiler has many attractions: Increase performance without changing the system. Ideally, we would like to see an improvement in the performance of a system just by changing the compiler for a better one, without upgrading the processor or adding more memory, both of which incur some cost either in the hardware itself, or indirectly through, for example, higher power consumption[4].

**More features at zero cost** We would like to add more features (*i.e.*, software) to an embedded program. But this extra software will require more memory to store it. If we can reduce the target code size by upgrading our compiler, we can squeeze more functionality into the same space as was used before[5].

**Good programmers know their worth** The continual drive for more software, sooner, drives the need for more programmers to design and implement the software. But the number of *good* programmers who are able to produce fast or compact code is limited, leading technology companies to employ average-grade programmers and rely on compilers to bridge (or at the very least, reduce) this ability gap[5].

**Same code, smaller/faster code** One mainstay of software engineering is *code reuse*, for two good reasons. Firstly, it takes time to develop and test code, so re-using existing components that have proven reliable reduces the time necessary for modular testing. Secondly, the time-to-market pressures mean there just is not the time to start from scratch on every project, so reusing software components can help to reduce the development time, and also reduce the development risk. The problem with this approach is that the reused code may not achieve the desired time or space requirements of the project. So it becomes the compiler's task to transform the code into a form that meets the requirements[5].

## II. THE COMPILER

In order to reduce the complexity of designing and building computers, nearly all of these are made to execute relatively simple commands (but do so very quickly). A program for a computer must be built by combining these very simple commands into a program in what is called Machine language. Since this is a tedious and error-prone process most programming is, instead, done using a high-level programming language. This language can be very different from the machine language that the computer can execute, so some means of bridging the gap is required. This is where the compiler comes in. A compiler translates (or compiles) a program written in a high level programming language that is suitable for human programmers into the low-level machine language that is required by computers. During this process, the compiler will also attempt to spot and report obvious programmer mistakes using a high-level language for programming has a large impact on how fast programs can be developed. The main reasons for this are:

- Compared to machine language, the notation used by programming languages is closer to the way humans think about problems.
- The compiler can spot some obvious programming mistakes.
- Programs written in a high-level language tend to be shorter than equivalent programs written in machine language[1,7].

#### A. The phases of a compiler

Since writing a compiler is a nontrivial task, it is a good idea to structure the work. A typical way of doing this is to split the compilation into several phases with well-defined interfaces. Conceptually, these phases operate in sequence (though in practice, they are often interleaved), each phase (except the first) taking the output from the previous phase as its input. It is common to let each phase be handled by a separate module. Some of these modules are written by hand, while others may be generated from specifications. Often, some of the modules can be shared between several compilers. A common division into phases is described below. In some compilers, the ordering of phases may differ slightly, some phases may be combined or split into several phases or some extra phases may be inserted between those mentioned below.

##### 1) Lexical analysis

This is the initial part of reading and analyzing the program text: The text is read and divided into tokens, each of which corresponds to a symbol in the programming language, e.g., a variable name, keyword or number.

##### 2) Syntax analysis

This phase takes the list of tokens produced by the lexical analysis and arranges these in a tree-structure (called the syntax tree) that reflects the structure of the program. This phase is often called parsing.

##### 3) Type checking

This phase analyses the syntax tree to determine if the program violates certain consistency requirements, e.g., if a variable is used but not declared or if it is used in a context that doesn't make sense given the type of the variable, such as trying to use a Boolean value as a function pointer.

##### 4) Intermediate code generation

The program is translated to a simple machine-independent intermediate language.

##### 5) Register allocation

The symbolic variable names used in the intermediate code are translated to numbers, each of which corresponds to a register in the target machine code.

##### 6) Machine code generation

The intermediate language is translated to assembly language (a textual representation of machine code) for specific machine architecture.

##### 7) Assembly and linking

The assembly-language code is translated into binary representation and addresses of variables, functions, etc., are determined [1].

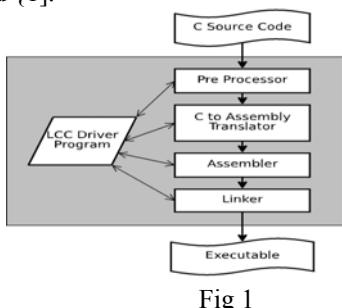


Fig 1

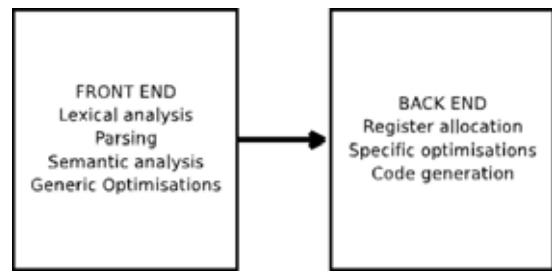


Fig 2

### III. CHOICE OF COMPILER

When developing a compiler for a new architecture it makes sense to develop as little new, and thus untested, code as possible. Using an existing compiler, especially one that is designed to be portable, makes a great deal of sense. There are many C compilers available, but I was only able to find only two that fulfill the following requirements:

- Designed to be portable
- Well documented with freely available source code. In order to modify the compiler, source code needs to be available.
- Supports the C language.

These two are GCC and LCC

#### A. GCC Vs LCC

GCC originally stood for the GNU1 C Compiler it now stands for the GNU Compiler Collection. It is explicitly targeted at fixed register array architectures, but is freely available and well optimized. It is thoroughly, if not very clearly, documented [7]. lcc does not seem to be an acronym, just a name, and is designed to be a fast and highly portable C compiler. Its intermediate form is much more amenable to machines. It is documented clearly and in detail in the book [8] and subsequent paper covering the updated interface [9]. 1GNU is a recursive acronym for GNU's Not UNIX. It is the umbrella name for the Free Software Foundation's operating system and tools.

- Constant folding. Dead code elimination. Common sub-expression elimination.
- Loop optimizations. Instruction scheduling. Delayed branch scheduling.

A point by point comparison of lcc and gcc is shown . The long list of positives in the GCC column suggests that using GCC would be the best option for a compiler platform f, at least from the user's point of view. However the two negatives suggest that this porting GCC to a dataflow graphs would be difficult, if not impossible. After all, a working and tested version of lcc is a very useful tool, whereas a non-working version of GCC is useless. This leads to the question: Could a working, efficient version of GCC be produced in the time available.

#### B. LCC

lcc is a portable C compiler with a very well defined and documented interface between the front and back ends. The intermediate representation for lcc consists of lists of trees known as forests, each tree representing an expression. The nodes represent arithmetic expressions, parameter passing, and flow control in a machine independent way. The front end of

lcc does lexical analysis, parsing, type checking and some optimizations. The back end is responsible for register allocation and code generation. The standard register allocator attempts to put as many eligible variables in registers as possible, without any global analysis. Initially all such variables are represented by local address nodes that are replaced with virtual register nodes. The register allocator then determines which of these can be stored in a real register and which have to be returned to memory. The stack allocator uses a similar overall approach, although the algorithm used to do the register allocation is completely different and does include global analysis. Previous attempts at register allocation for stack machines have been implemented as post-processors to a compiler rather than as part of the compiler itself. The compiler simply allocates all local variables to memory and the post-processor then attempts to eliminate memory accesses. The problem with this approach is that the post-processor needs to be informed which variables can be removed from memory without changing the program semantics, since some of the variables in memory could be aliased, that is, they could be referenced indirectly by pointer. Moving these variables to registers would change the program semantics.

#### 1) Initial Attempt

There has been some effort in generating dataflow graphs in lcc these are in form value state dependence graphs program dependence graphs. but that there are issues relating to the implementation of analyses and transformations.

Flow information, which was readily available within the compiler, has to be extracted from the assembly code. It is extremely difficult, if not impossible; to extract flow control from computed jumps such as the C ‘switch’ statement. Implementing the register allocator within the compiler not only solves the above problems, it also allows some optimizations which work directly on the intermediate code trees [8]. Finally, the code-generator always walked the intermediate code tree bottom up and left to right. This can cause *swap* instructions to be inserted where the architecture expects the left hand side of an expression to be on top of the stack. By modifying the trees directly these back to front trees can be reversed.

#### IV. THE IMPROVED VERSION

The standard distribution of lcc comes with a number of backends for the Intel x86 and various RISC architectures. These share a large amount of common code. Each backend has a machine specific part of about 1000 lines of tree-matching rules and supporting C code, while the register allocator and code-generator-generator are shared. In order to port lcc to a stack architecture a new register allocator was required, but otherwise as much code as possible is reused. Apart from the code-generator-generator, which is reused, the backend is new code. The machine descriptions developed for the initial implementation were reused.

A. The Register Allocation Phase Since the register allocation algorithm which was to be used was unknown when the architecture of the back-end was being designed. The optimizer architecture was designed to allow new optimizers to be added later. The flow-graph is built by the back-end and then handed

to each optimization phase in turn. The optimized flow-graph is then passed to the code generator for assembly language output.

B. Producing the Flow-graph as the front-end generates forests representing sections of source code these are added to the flow-graph. No novel techniques are used here. When the flow-graph is complete, usage and liveness information is calculated for use in later stages.

#### C. Optimization

lcc already supports modification and annotation of the intermediate code forests, all that is required here is an interface for an optimizer which takes a flow-graph and returns the updated version of that flow-graph. The optimization phase takes a list of optimizations and applies each one in turn. Ensuring that the optimizations are applied in the correct order is up to the programmer providing the list.

Tree Flipping some binary operators, assignment and division for example, may require a reversal of the tree to prevent the insertion of extra *swap* instructions. All stack machine descriptions include a structure describing which operators need this reversal performed and which do not[8].

#### D. Infrastructure

Built upon these classic computer science data structures are the application specific data structures: the flow-graph, basic blocks, liveness information and means of representing the e-, p-, l- and x-stacks. These follow a similar pattern to the fundamental data structures, but with more complex interfaces and imperfect data hiding. For example the ‘Block’ data-structure holds information about a basic block. This includes the depths of the stack at its start and end and which variables are defined or used in that block. It also provides functions for inserting code at the start or end of the block, as well as the ability to join blocks[8].

#### E. LCC Tree

The intermediate form in lcc, that is, the data structures that are passed from the front end to the back end, are forests of trees each representing a statement or expression. For example the expression  $y = a[x]+4$ , where  $a$  is an array of ints, is represented by the tree in Figure 3.4. Trees also represent flow control and procedure calls. For example the C code in Figure 3 is represented by the forest[8].

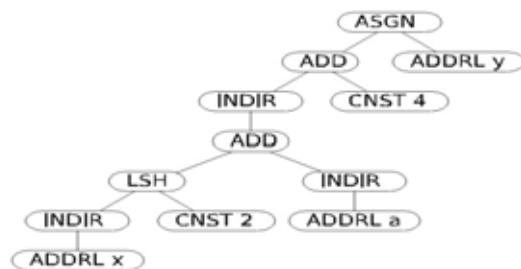


Fig 3

#### V. PROPOSED SOLUTION

*Our proposed solution consists of the following passes*

During compilation LCC performs some simple transformations (strength reduction, constant folding, etc), some of which are mandated by the language standard (e.g.,

constant folding). For each statement, LCC generates an abstract syntax tree (AST), recursively descending into source-level statement blocks. DFG nodes and edges are generated directly from the AST with the aid of the symbol tables.

In 2<sup>nd</sup> pass transformation of abstract syntax tree in to program dependence graph

It consist of fallowing stages

1. Identifying data and control dependencies by the methodology used in[10] and creates conventional block structure.

2 Content of each block transformed into dataflow graph which show data dependencies

#### A. DFG refinement

*DFG Refinement* is the first process in the optimizing pass (*the 3rd pass*) of the compiler. The objective of *DFG Refinement* is to detect and delete redundant nodes in *DFG* produced by the previous pass. *DFG Refinement* can be classified into three sub transformations: *Basic -Refinement (BR)*, *Identical-Refinement (IR)*, and synchronization deletion refinement (*SDR*), which are conducted as follows

##### Step 1.

Traverse the *DFG* from the start(-node) and apply to each node, in order, the rule of *BR* and *IR*. Repeat this until no further refinement is possible.

##### Step 2.

Traverse the *DFG* from the start and apply to each node, in order, the rules of *SDR*. Repeat this until no further refinement is possible. Go back to Step 1 if any refinement has taken place.

The main function of *BR* is either to combine some number of individual nodes in to a single compound node, or to parallelize nodes for reduction of the critical path (see Fig.4), whereas the main function of *IR* is either to delete records or to move them outside loops

#### B. Flow control

In order to prevent exhaustion of hardware resources, timing control: for data token production and consumption must be done. As nodes in *DFG* can be classified into producer-nodes(pr), operation-nodes(op), and consumer nodes(cn), where data produced by gs are bounded so that they are absorbed by cns, achieved in the following manner: first, the value for the Unit-Flow (the upper-boundary on the amount of data for one production action) is determined on the basis of the amount of hardware resource available. Next, for each pr, the en having the largest rank-value among all corresponding cns, i.e. the last node to be executed among corresponding cns, is chosen, and a count (a token counter) is inserted before it. The role of the count is to send a reactivation (feed-back) token back to the corresponding pr after a Unit-Flow of tokens has Since detailed flow analyses are required for optimizing the value of the Unit-Flow, this value has been left un-optimized in the current version of compiler, and it is assumed that individual users will specify an optimum value.

#### C. Loop unfolding

While loop unfolding is the conventional way of speeding up loop executions on dataflow architectures, it can lead to the exhaustion of hardware resources due to its overwhelming

parallelism. For the control of parallelism in loop unfolding, the control of the two parameters, i.e. the unfolding degree k , and the initiation interval for each iteration d, is important, The objective of PG loop unfolding is to perform graph reconstructions for each loop structure in *DFG* in order to achieve proper control on these two parameters, For the control of k, a technique derived from early ImPP assembly programs is employed, where a sync Is used to synchronize the delivery of boolean tokens to the switches with the termination of every iteration, while k tokens are preloaded on the input link to the sync from the sync-tree, which produces the termination signal taken from every iteration The actual value for k is adjusted to  $L/d$  provided that  $2 < k < 5$  under the consideration of matching memory capacity of ImPP, where  $L$  is the average execution time for each iteration. The effect is that, the number of iterations which proceed in parallel is limited to a constant value  $k$ , i.e. a so called &-bounded loop execution[11] is achieved. The control of d is accomplished by adjusting the value of  $d$  , which initially depends on individual iteration content( e.g. the update latency of an inductive variable), to an adequate value in order to reduce both execution time for the loop and consumption of hardware resources (the matching memory).

## VI. CONCLUSION

Research to generate dataflow graph as intermediate form then execute it on hardware has put forward various possibilities mainly with the flexibility and capacity of the reconfigurable architectures. A Dataflow graph (*DFG*) is a fundamental element in this process. Dataflow graph for dataflow architecture in lcc can be generated in order to produce a high level of parallelism.

## VII REFERENCES

- [1]. A.V. Aho , R sethi , Monica s. lam J.D. Ulman compiler: principle practice and tools: pearson education 2006
- [2]. [http://research.microsoft.com/\\_toddpro/papers/law.htm](http://research.microsoft.com/_toddpro/papers/law.htm), May 1999.
- [3]. MOORE, G. E. Cramming more components onto integrated circuits. *Electronics* 38, 8 (April 1965).
- [4]. Ali, F. M. and Das, A. S.Azme, Hardware-software co-synthesis of hard real- time systems with reconfigurable FPGAs, *ELSEVIER - Computer and Electrical Engineering*(2004), volume=30,pg 471-489
- [5]. Arvind, Dataflow: Passing the token, *ISCA Keynote* (2005)
- [6]. Bailey. Proposed mechanisms for super-pipelined instruction-issue for ILP stack machines. In *DSD*, pages 121–129. IEEE Computer Society, 2004.
- [7]. <http://gcc.gnu.org/>
- [8]. W. Fraser and D. R. Hanson. *A Retargetable C Compiler: Design and Implementation*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1995.
- [9]. W. Fraser and D. R. Hanson. The lcc 4.x code-generation interface. Technical Report MSR-TR-2001-64, Microsoft Research (MSR), July 2001
- [10]. Ferrante, K. J. Oteasteina and J. D. Warren: The Program Dependence Graph and Its Use in Optimization,
- [11]. ACM Trans. on Programming Language and Syrtrmr, Vo1.9, No.3,July 1987, pp.319-349.
- [12]. Acvind, R. Nikhil : Executing a Program on the MIT Tnggd-Tahn Dataflow Architecture, *IEEE %an#*. On Computerr, Vo1.39, No.3, 199Qrpp.900-318

# Recognition of Handwritten Text by Neocognitron Based System

Sr. Lect- Er. Ramandeep Singh, Lect.-Er. Harpreet Singh, Lect.-Er. Shiwani Aggarwal, LCET Katani-kalan

**Abstract-** The architecture of the “neocognitron” neural network in the task of searching for structure

ral units in a gray scale image of an integrated circuit is considered. Neural networks have been used to recognize handwritten characters such as typed, hand written, etc. But their performance, i.e., the recognition rate, depends on a number of factors which may include the network architecture, feature selection, network parameter setting, learning strategy, learning sample selection, test pattern preprocessing, etc. These factors are important for engineers in designing a network for a particular application problem, but unfortunately there is a lack of systematic way to guide their decision-making regarding the selection of these parameters. This paper presents a neural network model neocognitron for robust visual pattern recognition. The comparative outcomes of recognition have shown the advantages of neocognitron neural network approach

**Keywords:** Neural network model; Multi-layered network; Neocognitron

## I. INTRODUCTION

The neocognitron is a hierarchical multilayered neural network that has been used for handwritten character recognition and other pattern recognition tasks. The neural network model is used for neocognitron for robust visual pattern recognition. It acquires the ability to recognize robustly visual patterns through learning. The hypothesized a hierarchical structure in the visual cortex: simple cells → complex cells → lower-order hypercomplex cells → higher-order hypercomplex cells. They also suggested that the relation between simple and complex cells resembles that between lower- and higher-order hypercomplex cells. Although physiologists do not use recently the classification of lower- and higher-order Hypercomplex cells, hierarchical repetition of similar anatomical and functional architectures in the visual system still seems to be plausible from various physiological experiments.

The architecture of the neocognitron was initially suggested by these physiological findings. The neocognitron consists of layers of S-cells, which resemble simple cells, and layers of C-cells, which resemble complex cells. These layers of S- and C-cells are arranged alternately in a hierarchical manner. In other words, a number of modules, each of which consists of an S- and a C-cell layer, are connected in a cascade in the network. S-cells are feature-extracting cells, whose input connections are variable and are Modified through learning. C-cells, whose input connections are fixed and unmodified, exhibit an approximate invariance to the position of the stimuli presented within their receptive fields. The local features are extracted by S-cells, and these features deformation, such as local shifts, are tolerated by C-cells. Local features in the input are integrated gradually and classifying in the higher layers.

The C-cells in the highest stage work as recognition cells, which indicates the result of the pattern recognition. After learning, the neocognitron can recognize input patterns robustly, with little effect from deformation, change in size, or shift in position.

## II. ARCHITECTURE OF THE NETWORK

### 2.1. Outline of the network

Fig. 1 shows the architecture of the proposed network. As can be seen from the figure, the network has 4 stages of S- and C-cell layers:  $U_0 \rightarrow U_G \rightarrow U_{S1} \rightarrow U_{C1} \rightarrow U_{S2} \rightarrow U_{C2} \rightarrow U_{S3} \rightarrow U_{C3} \rightarrow U_{S4} \rightarrow U_{C4}$ .

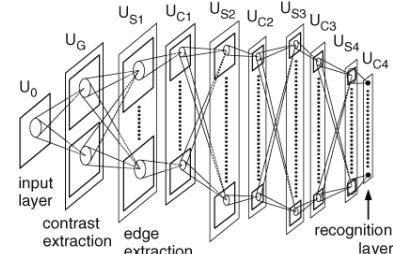


Fig. 1. The architecture of the proposed neocognitron.

The lowest stage is the input layer consisting of two-dimensional array of cells, which correspond to photoreceptors of the retina. There are retinotopically ordered connections between cells of adjoining layers. Each cell receives input connections that lead from cells situated in a limited area on the preceding layer. Layers of "S-cells" and "C-cells" are arranged alternately in the hierarchical network. (In the network shown in Fig.1, a contrast-extracting layer is inserted between the input layer and the S-cell layer of the first stage).

#### 2.1.1. S-cells

S-cells work as feature-extracting cells. They resemble simple cells of the primary visual cortex in their response. Their input connections are variable and are modified through learning. After having finished learning, each S-cell come to respond selectively to a particular feature presented in its receptive field. The features extracted by S-cells are determined during the learning process. Generally speaking, local features, such as edges or lines in particular orientations, are extracted in lower stages. More global features, such as parts of learning patterns, are extracted in higher stages.

#### 2.1.2 C-cells

C-cells which resembles complex cells in the visual cortex, are inserted in the network to allow for positional errors in the features of the stimulus. The input connections of C-cells, which come from S-cells of the preceding layer, are fixed and invariable. Each C-cell receives excitatory input connections from a group of S-cells that extract the

same feature, but from slightly different positions. The C-cell responds if at least one of these S-cells yields an output. Even if the stimulus feature shifts in position and another S-cell comes to respond instead of the first one, the same C-cell keeps responding. Thus, the C-cell's response is less sensitive to shift in position of the input pattern. We can also express that C-cells make a blurring operation, because the response of a layer of S-cells is spatially blurred in the response of the succeeding layer of C-cells.

Each layer of S-cells or C-cells is divided into sub-layers, called "cell-planes", according to the features to which the cells responds. The cells in each cell-plane are arranged in a two-dimensional array. A cell-plane is a group of cells that are arranged retinotopically and share the same set of input connections. In other words, the connections to a cell-plane have a translational symmetry. As a result, all the cells in a cell-plane have receptive fields of an identical characteristic, but the locations of the receptive fields differ from cell to cell. The modification of variable connections during the learning progresses also under the restriction of shared connections

#### 2.1.3 Input layer cells

The stimulus pattern is presented to the input layer (photoreceptor layer)  $U_0$ . The input connections to a single cell of layer  $U_G$  are designed in such a way that their total sum is equal to zero. This means that the dc component of spatial frequency of the input pattern is eliminated in the contrast-extracting layer UGA layer of contrast-extracting cells ( $U_G$ ), which correspond to retinal ganglion cells follows layer  $U_0$ . The contrast-extracting layer  $U_G$  consists of two cell-planes: one cell-plane consisting of cells with concentric on-center receptive fields, and one cell-plane consisting of cells with off-center receptive fields. The former cells extract positive contrast in brightness, whereas the latter extract negative contrast from the images presented to the input layer.

The output of layer  $U_G$  is sent to the S-cell layer of the first stage ( $U_{S1}$ ). The S-cells of layer  $U_{S1}$  correspond to simple cells in the primary visual cortex. They have been trained using supervised learning to extract edge components of various orientations from the input image. The present model has four stages of S- and C-cell layers. The output of layer  $U_{S1}$  is fed to layer  $U_{C1}$ , where a blurred version of the response of layer  $U_{S1}$  is generated. The density of the cells in each cell-plane is reduced between layers  $U_{S1}$  and  $U_{C1}$ . The S-cells of the intermediate stages ( $U_{S2}$  and  $U_{S3}$ ) are self-organized using un supervised competitive learning similar to the method used by the conventional neocognitron. Layer  $U_{C4}$ , which is the highest stage of the network, is the recognition layer, whose response shows the final result of pattern recognition by the network.

Layer  $U_{S1}$ , namely, the S-cell layer of the first stage, is an edge-extracting layer. It has 16 cell-planes, each of which consists of edge-extracting cells of a particular Preferred orientation. The threshold of the S-cells, which determines the selectivity in extracting features, is set low enough to accept edges of slightly different orientations. The S-cells of this layer have been trained using supervised learning. To

train a cell-plane, the "teacher" presents a training pattern, namely a straight edge of a particular orientation, to the input layer of the network. The teacher then points out the location of the feature, which, in this particular case, can be an arbitrary point on the edge. The cell whose receptive field center coincides with the location of the feature takes the place of the seed cell of the cell-plane, and the process of reinforcement occurs automatically. It should be noted here that the process of supervised learning is identical to that of the unsupervised learning except the process of choosing seed cells. The speed of reinforcement of variable input connections of a cell is set so large that the training of a seed cell (and hence the cell-plane) is completed by only a single presentation of each training pattern. The optimal value of threshold can be determined as follows. A low threshold reduces the orientation selectivity of the S-cells and increases the tolerance for rotation of edges to be extracted. Computer simulation shows that a lower threshold usually produces a greater robustness against deformation of the input patterns. If the threshold becomes too low, however, S-cells of this layer come to yield spurious outputs, responding to features other than desired edges. Hence it can be concluded that the optimal value of the threshold is the lower limit of the value that does not generate spurious responses from the cells. The optimal threshold value, however, changes depending whether an inhibitory surround is introduced in the connections to the C-cells or not. The inhibitory surround produces an effect like a lateral inhibition, and small spurious responses generated in layer  $U_{S1}$  can be suppressed in layer  $U_{C1}$ . Hence the threshold can be lowered down to the value by which no spurious responses are observed, not in  $U_{S1}$ , but in  $U_{C1}$ .

#### 2.1.4 intermediate layers

The S-cells of intermediate stages ( $U_{S2}$  and  $U_{S3}$ ) are self-organized using unsupervised competitive learning similar to the method used in the conventional neocognitron. Seedcells are determined by a kind of winner-take-all process. Every time a training pattern is presented to the input layer, each S-cell competes with the other cells in its vicinity, which is called the competition area and has the shape of a hypercolumn. If and only if the output of the cell is larger than any other cells in the competition area, the cell is selected as the seedcell. Because of the shared connections within each cell-plane, all cells in the cell-plane come to have the same set of input connections as the seedcell. Line-extracting S-cells are generated (or, self-organized) together with cells extracting other features in the second stage ( $U_{S2}$ ). The cells in this stage extract features using information of edges that are extracted in the preceding stage. The neural networks' ability to recognize patterns robustly is induced by the selectivity of feature-extracting cells, which is controlled by the threshold of the cells. Fukushima [3] had proposed the use of higher threshold values for feature-extracting cells in the learning phase than in the recognition phase, when un supervised learning with a winner-take-all process is used to train neural networks. This method of dual threshold is used for the learning of layers  $U_{S2}$  and  $U_{S3}$ .

### 2.1.5 Learning method for the highest stage

S-cells of the highest stage ( $U_{S4}$ ) are trained using a supervised competitive learning. The learning rule resembles the competitive learning used to train  $U_{S2}$  and  $U_{S3}$ , but the class names of the training patterns are also utilized for the learning. When the network learns varieties of deformed training patterns through competitive learning, more than one cell-plane for one class is usually generated in  $U_{S4}$ . Therefore, when each cell-plane first learns a training pattern, the class name of the training pattern is assigned to the cell-plane. Thus, each cell-plane of  $U_{S4}$  has a label indicating one of the 10 digits. Every time a training pattern is presented, competition occurs among all S-cells in the layer. (In other words, the competition area for layer  $U_{S4}$  is large enough to cover all cells of the layer.) If the winner of the competition has the same label as the training pattern, the winner becomes the seedcell and learns the training pattern in the same way as the seedcells of the lower stages. If the winner has a wrong label (or if all S-cells are silent), however, a new cell-plane is generated and is put a label of the class name of the training pattern. During the recognition phase, the label of the maximum-output S-cell of  $U_{S4}$  determines the final result of recognition. We can also express this process of recognition as follows. Recognition layer  $U_{C4}$  has 10 C-cells corresponding to the 10 digits to be recognized. Every time a new cell-plane is generated in layer  $U_{S4}$  in the learning phase, excitatory connections are created from all S-cells of the cell-plane to the C-cell of that class name. Competition among S-cells occur also in the recognition phase, and only one maximum output S-cell within the whole layer  $U_{S4}$  can transmit its output to  $U_{C4}$ .

### III. PRINCIPLES OF DEFORMATION-RESISTANT RECOGNITION

In the whole network, with its alternate layers of S-cells and C-cells, the process of feature-extraction by S-cells and toleration of positional shift by C-cells is repeated. During this process, local features extracted in lower stages are gradually integrated into more global features, as illustrated in Fig.2

Since small amounts of positional errors of local features are absorbed by the blurring operation by C-cells, an S-cell in a higher stage comes to respond robustly to a specific feature even if the feature is slightly deformed or shifted.

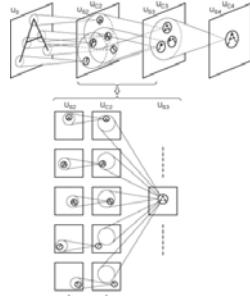


Figure 2: The process of pattern recognition in the neocognitron. The lower half of the figure is an enlarged illustration of a part of the network.

Fig.3 illustrates this situation. Let an S-cell in an intermediate stage of the network have already been trained to extract a global feature consisting of three local features of a training pattern 'A' as illustrated in Fig. 3(a). The cell tolerates a positional error of each local feature if the deviation falls within the dotted circle. Hence, the S-cell responds to any of the deformed patterns shown in Fig. 3(b). The toleration of positional errors should not be too large at this stage. If large errors are tolerated at any one step, the network may come to respond erroneously, such as by recognizing a stimulus like Fig. 3(c) as an 'A' pattern.

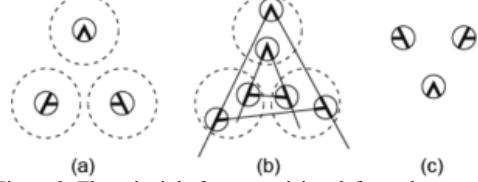


Figure 3: The principle for recognizing deformed patterns.

Thus, tolerating positional error a little at a time at each stage, rather than all in one step, plays an important role in endowing the network with the ability to recognize even distorted patterns.

The C-cells in the highest stage work as recognition cells, which indicate the result of the pattern recognition. Each C-cell of the recognition layer at the highest stage integrates all the information of the input pattern, and responds only to one specific pattern. Since errors in the relative position of local features are tolerated in the process of extracting and integrating features, the same C-cell responds in the recognition layer at the highest stage, even if the input pattern is deformed, changed in size, or shifted in position. In other words, after having finished learning, the neocognitron can recognize input patterns robustly, with little effect from deformation, change in size, or shift in position.

### IV. SELF-ORGANIZATION OF THE NETWORK

The neocognitron can be trained to recognize patterns through learning. Only S-cells in the network have their input connections modified through learning. Various training methods, including unsupervised learning and supervised learning, have been proposed so far. This section introduces a process of unsupervised learning.

In the case of unsupervised learning, the self-organization of the network is performed using two principles. The first principle is a kind of winner-take-all rule: among the cells situated in a certain small area, which is called a hypercolumn, only the one responding most strongly becomes the winner. The winner has its input connections strengthened. The amount of strengthening of each input connection to the winner is proportional to the intensity of the response of the cell from which the relevant connection leads. To be more specific, an S-cell receives variable excitatory connections from a group of C-cells of the preceding stage as illustrated in Fig.4. Each S-cell is accompanied with an inhibitory cell, called a V-cell.

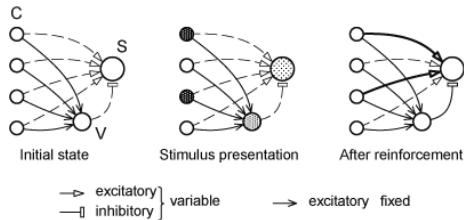


Figure 4: Connections converging to an S-cell in the learning phase.

The S-cell also receives a variable inhibitory connection from the V-cell. The V-cell receives fixed excitatory connections from the same group of C-cells as does the S-cell, and always responds with the average intensity of the output of the C-cells.

The initial strength of the variable connections is very weak and nearly zero. Suppose the S-cell responds most strongly among the S-cells in its vicinity when a training stimulus is presented. According to the winner-take-all rule described above, variable connections leading from activated C-cells are strengthened. The variable excitatory connections to the S-cell grow into a template that exactly matches the spatial distribution of the response of the cells in the preceding layer. The inhibitory variable connection from the V-cell is also strengthened at the same time to the average strength of the excitatory connections.

After the learning, the S-cell acquires the ability to extract a feature of the stimulus presented during the learning period. Through the excitatory connections, the S-cell receives signals indicating the existence of the relevant feature to be extracted. If an irrelevant feature is presented, the inhibitory signal from the V-cell becomes stronger than the direct excitatory signals from the C-cells, and the response of the S-cell is suppressed. Once an S-cell is thus selected and has learned to respond to a feature, the cell usually loses its responsiveness to other features. When a different feature is presented, a different cell usually yields the maximum output and learns the second feature. Thus, a "division of labor" among the cells occurs automatically.

The second principle for the learning is introduced in order that the connections being strengthened always preserving translational symmetry, or the condition of shared connections. The maximum-output cell not only grows by itself, but also controls the growth of neighboring cells, working, so to speak, like a seed in crystal growth. To be more specific, all of the other S-cells in the cell-plane, from which the "seed cell" is selected, follow the seed cell, and have their input connections strengthened by having the same spatial distribution as those of the seed cell.

#### V. RECOGNITION RATE

This part is to test the behavior of the proposed network by computer simulation using handwritten digits (free writing) from randomly sampled database. For S-cells of layers  $U_{S2}$  and  $U_{S3}$ , the method of dual thresholds is used for the learning and recognition phases. Each training pattern of the training set was presented once for the learning of layers  $U_{S2}$  and  $U_{S3}$ . For the learning of layer  $U_{S4}$  at the highest stage, the same training set was presented repeatedly until all

the patterns in the training set were recognized correctly. Although the required number of repetition changes depending on the training set, it usually is not so large. In the particular case shown below, it was only 4.

We searched the optimal thresholds that produce the best recognition rate. Since there are a large number of combinations in the threshold values of four layers, a complete search for all combinations has not been finished yet. The recognition rate varies depending on the number of training patterns. When we used 3000 patterns (300 patterns for each digit) for the learning, for example, the recognition rate was 98.6% for a blind test sample (3000 patterns), and 100% for the training set.

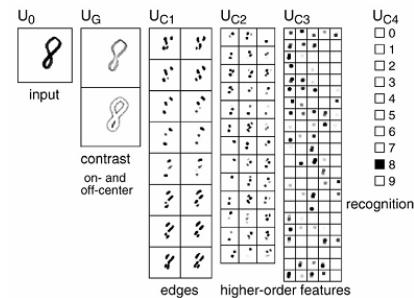


Fig. 5 An example of the response of the neocognitron. The input pattern is recognized correctly as '8'.

Fig. 5 shows a typical response of the network that has finished the learning. The responses of layers  $U_0$ ,  $U_G$  and layers of C-cells of all stages are displayed in series from left to right. The rightmost layer,  $U_{C4}$ , is the recognition layer, whose response shows the final result of recognition. The responses of S-cell layers are omitted from the figure but can be estimated from the responses of C-cell layers: a blurred version of the response of an S-cell layer appears in the corresponding C-cell layer, although it is slightly modified by the inhibitory surround in the connections.

#### VI. ACKNOWLEDGEMENT

To improve the recognition rate of the handwritten text, several modifications have been applied in Artificial Neural network. These modifications allowed the removal of accessory circuits appended in the previous techniques, resulting in an improvement of recognition rate as well as simplification of the network architecture. Some people claim that a neocognitron is a complex network, but it is not correct. The mathematical operation between adjacent cell-planes can be interpreted as a kind of two-dimensional filtering operation because of shared connections. If we count the number of processes performed in the network, assuming that one filtering operation corresponds to one process, the neocognitron is a very simple network compared to other artificial neural networks. The required number of repeated presentation of a training set is much smaller for the neocognitron than for the network trained by back propagation. Although the phenomenon of overlearning (or overtraining) has not been observed in the simulation shown in this paper, the possibility cannot be completely excluded. If some patterns in a database set had wrong class names, the learning of the highest stage might be affected by

the erroneous data. A serious overlearning, however, does not seem to occur in the intermediate stages, if the thresholds for the learning phase are properly chosen. Since the selectivity (or the size of the tolerance area) of a cell is determined by a fixed threshold value and does not change, all training patterns within the tolerance area of the winner cell contribute to the learning of the cell, and not for the generation of new cell-planes. In other words, an excessive presentation of training patterns does not necessarily induce the generation of new cell-planes.

Although a repeated presentation of similar training patterns to a cell increases the values of the connections to the cell, the behavior of the cell almost stops changing after having finished some degrees of learning. Their optimal values vary depending on the characteristics of the pattern set to be recognized. We need to search optimal thresholds by experiments. To find out a good method for determining these thresholds is a problem left to be solved in the future.

## VII. REFERENCES

- [1]. C. Neubauer, Evaluation of convolutional neural networks for visual recognition, IEEE Trans. Neural Networks 9 (4) (1998) 685–696.
- [2]. D. Shi, C. Dong, D.S. Yeung, Neocognitron's parameter tuning by genetic algorithms, Internat. J. Neural Systems 9 (6) (1999) 495–509.
- [3]. K. Fukushima: "Restoring partly occluded patterns: a neural network model", *Neural Networks*, **18**[1], pp. 33-43 (2005).
- [4]. K. Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4): 93-202, 1980.
- [5]. K. Fukushima, S. Miyake, and T. Ito. Neocognitron: a neural network model for a mechanism of visual pattern recognition. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC - 13 (Nb.3) pp. 826 - 834, September/October 1983.
- [6]. M.C.M. ElaliFe, E.T. Rolls, S.M. Stringer, Invariant recognition of feature combinations in the visual system, *Biol. Cybernet.* 86 (2002) 59–71.
- [7]. S. Sato, J. Kuroiwa, H. Aso, S. Miyake, Recognition of rotated patterns using a neocognitron,in: L.C. Jain, B. Lazzerini (Eds.), *Knowledge Based Intelligent Techniques in Character Recognition*, CRC Press, Boca Raton, FL, 1999, pp. 49–64.

# Using Neural Networks to Forecast Stock Market Prices

Ms. Richa setiya, N.C.C.E., ISRANA, INDIA, Er. Swati Arora- Lecturer, Indus Institute of Engg. & Tech., Jind  
[richasetiya@rediffmail.com](mailto:richasetiya@rediffmail.com), [er.swati19@gmail.com](mailto:er.swati19@gmail.com), [er.swati@yahoo.co.in](mailto:er.swati@yahoo.co.in)

**Abstract-** A neural network is a computer program or hardwired machine that is designed to learn in a manner similar to the human brain. More specifically, a neural network is a massively parallel distributed processor that has a natural propensity for storing experiential knowledge and making it available for use. It resembles the brain in two respects: Knowledge is acquired by the network through a learning process and interneuron connection strengths known as synaptic weights are used to store the knowledge.

This paper describes survey on the application of neural networks in forecasting stock market prices. With their ability to discover patterns in nonlinear and chaotic systems, neural networks offer the ability to predict market directions more accurately than current techniques. Common market analysis techniques such as technical analysis, fundamental analysis, and regression will be discussed and compared with neural network performance. However, no one technique or combination of techniques has been successful enough to consistently "beat the market". With the development of neural networks, researchers and investors are hoping that the market mysteries can be unraveled .Also, the Efficient Market Hypothesis (EMH) will be presented and contrasted with chaos theory and neural networks. Paper predicts stock market prices to attain financial gain .Neural networks are used to predict stock market prices because they are able to learn nonlinear mappings between inputs and outputs hence a good forecasting tool.

**Keywords**-Neural networks, Forecasting, EMH.

## I. INTRODUCTION

From the beginning of time it has been man's common goal to make his life easier. The prevailing notion in society is that wealth brings comfort and luxury, so it is not surprising that there has been so much work done on ways to predict the markets. Various technical, fundamental, and statistical indicators have been proposed and used with varying results. However, no one technique or combination of techniques has been successful enough to consistently "beat the market". With the development of neural networks, researchers and investors are hoping that the market mysteries can be unraveled.

## II. WHAT IS A NEURAL NETWORK?

A neural network is a computational technique that benefits from techniques similar to ones employed in the human brain. It is designed to mimic the ability of the human brain to process data and information and comprehend patterns. It imitates the structure and operations of the three dimensional lattice of network among brain cells (nodes or neurons and hence the term "neural") the neural network is composed of many simple processing elements or neurons operating in parallel whose function is determined by network structure, connection strengths, and the processing performed at computing elements or nodes .The network's strength, however, is in its ability to comprehend and discern subtle patterns in a large number of variables at a time without being stifled by detail. It can also carry out multiple operations

simultaneously. Not only can it identify patterns in a few variables, it also can detect correlations in hundreds of variables. It is this feature of the network that is particularly suitable in analyzing relationships between a large numbers of market variables. The networks can learn from experience. They can cope with "fuzzy" patterns – patterns that are difficult to reduce to precise rules. They can also be retrained and thus can adapt to changing market behavior.

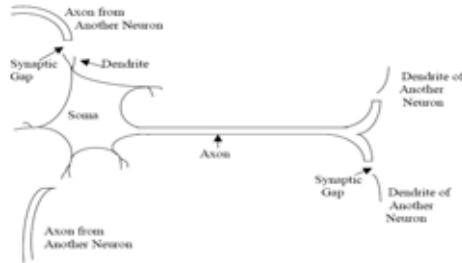


Figure I. Biological Neuron

## III. MOTIVATION

There are several motivations for trying to predict stock market prices. The most basic of these is financial gain. Any system that can consistently pick winners and losers in the dynamic market place would make the owner of the system very wealthy. Thus, many individuals including researchers, investment professionals, and average investors are continually looking for this superior system which will yield them high returns. There is a second motivation in the research and financial communities. It has been proposed in the Efficient Market Hypothesis (EMH) that markets are efficient in that opportunities for profit are discovered so quickly that they cease to be opportunities. The EMH effectively states that no system can continually beat the market because if this system becomes public, everyone will use it, thus negating its potential gain. Neural networks are used to predict stock market prices because they are able to learn nonlinear mappings between inputs and outputs. Contrary to the EMH, several researchers claim the stock market and other complex systems exhibit chaos. Chaos is a nonlinear deterministic process which only appears random because it can not be easily expressed. With the neural networks' ability to learn nonlinear, chaotic systems, it may be possible to outperform traditional analysis and other computer-based methods. In addition to stock market prediction, neural networks have been trained to perform a variety of financial related tasks. There are experimental and commercial systems used for tracking commodity markets and futures, foreign exchange trading, financial planning, company stability, and bankruptcy prediction. Banks use neural networks to scan credit and loan applications to estimate bankruptcy probabilities, while money managers can use neural networks to plan and construct profitable portfolios in real-time.

#### IV. ANALYTICAL METHODS

Before the age of computers, people traded stocks and commodities primarily on intuition. As the level of investing and trading grew, people searched for tools and methods that would increase their gains while minimizing their risk. Statistics, technical analysis, fundamental analysis, and linear regression are all used to attempt to predict and benefit from the market's direction. None of these techniques has proven to be the consistently correct prediction tool that is desired, and many analysts argue about the usefulness of many of the approaches.

##### *Technical Analysis*

The idea behind technical analysis is that share prices move in trends dictated by the constantly changing attitudes of investors in response to different forces. Using price, volume, and open interest statistics, the technical analyst uses charts to predict future stock movements. Technical analysis rests on the assumption that history repeats itself and that future market direction can be determined by examining past prices. Thus, technical analysis is controversial and contradicts the Efficient Market Hypothesis. However, it is used by approximately 90% of the major stock traders [3]. Despite its widespread use, technical analysis is criticized because it is highly subjective. Different individuals can interpret charts in different manners. Price charts are used to detect trends. Trends are assumed to be based on supply and demand issues which often have cyclical or noticeable patterns. Although technical analysis may yield insights into the market, its highly subjective nature and inherent time delay does not make it ideal for the fast, dynamic trading markets of today. ‘

##### *Fundamental Analysis*

Fundamental analysis involves the in-depth analysis of a company's performance and profitability to determine its share price. By studying the overall economic conditions, the company's competition, and other factors, it is possible to determine expected returns and the intrinsic value of shares. This type of analysis assumes that a share's current (and future) price depends on its intrinsic value and anticipated return on investment. As new information is released pertaining to the company's status, the expected return on the company's shares will change, which affects the stock price. The advantages of fundamental analysis are its systematic approach and its ability to predict changes before they show up on the charts. Companies are compared with one another, and their growth prospects are related to the current economic environment. This allows the investor to become more familiar with the company. However, fundamental analysis is a superior method for long-term stability and growth. Basically, fundamental analysis assumes investors are 90% logical, examining their investments in detail, whereas technical analysis assumes investors are 90% psychological, reacting to changes in the market environment in predictable ways.

##### *Traditional Time Series Forecasting*

Time series forecasting analyzes past data and projects estimates of future data values. Basically, this method attempts to model a nonlinear function by a recurrence relation derived from past values. The recurrence relation can then be used to

predict new values in the time series, which hopefully will be good approximations of the actual values. There are two basic types of time series forecasting: univariate and multivariate. Univariate models, like Box-Jenkins, contain only one variable in the recurrence equation. Box-Jenkins is a complicated process of fitting data to appropriate model parameters. The equations used in the model contain past values of moving averages and prices. Box-Jenkins is good for short-term forecasting but requires a lot of data, and it is a complicated process to determine the appropriate model equations and parameters. Multivariate models are univariate models expanded to "discover causal factors that affect the behavior of the data." [3] As the name suggests, these models contain more than one variable in their equations. Regression analysis is a multivariate model which has been frequently compared with neural networks. Overall, time series forecasting provides reasonable accuracy over short periods of time, but the accuracy of time series forecasting diminishes sharply as the length of prediction increases.

##### *The Efficient Market Hypothesis*

The Efficient Market Hypothesis (EMH) states that at any time, the price of a share fully captures all known information about the share. Since all known information is used optimally by market participants, price variations are random, as new information occurs randomly. Thus, share prices perform a "random walk", and it is not possible for an investor to beat the market. If stock market crashes, such as the market crash in October 1987, contradict the EMH because they are not based on randomly occurring information, but arise in times of overwhelming investor fear. The EMH is important because it contradicts all other forms of analysis. If it is impossible to beat the market, then technical, fundamental, or time series analysis should lead to no better performance than random guessing.

##### *Chaos Theory*

A relatively new approach to modeling nonlinear dynamic systems like the stock market is chaos theory. Chaos theory analyzes a process under the assumption that part of the process is deterministic and part of the process is random. Chaos is a nonlinear process which appears to be random. Various theoretical tests have been developed to test if a system is chaotic (has chaos in its time series). Chaos theory is an attempt to show that order does exist in apparent randomness. By implying that the stock market is chaotic and not simply random, chaos theory contradicts the EMH. In essence, a chaotic system is a combination of a deterministic and a random process. The deterministic process can be characterized using regression fitting, while the random process can be characterized by statistical parameters of a distribution function. Thus, using only deterministic or statistical techniques will not fully capture the nature of a chaotic system. A neural networks ability to capture both deterministic and random features makes it ideal for modeling chaotic systems.

##### *Comparing The Various Models*

In the wide variety of different modeling techniques presented so far, every technique has its own set of supporters

and detractors and vastly differing benefits and shortcomings. The common goal in all the methods is predicting future market movements from past information. The assumptions made by each method dictate its performance and its application to the markets. The EMH assumes that fully disseminated information results in an unpredictable random market. Thus, no analysis technique can consistently beat the market as others will use it, and its gains will be nullified. If an investor does not believe in the EMH, the other models offer a variety of possibilities. Technical analysis assumes history repeats itself and noticeable patterns can be discerned in investor behavior by examining charts. Fundamental analysis helps the long-term investor measure intrinsic value of shares and their future direction by assuming investors make rational investment decisions. Statistical and regression techniques attempt to formulate past behavior in recurrent equations to predict future values. Finally, chaos theory states that the apparent randomness of the market is just nonlinear dynamics too complex to be fully understood. So what model is the right one? There is no right model. Each model has its own benefits and shortcomings. I feel that the market is a chaotic system. It may be predictable at times, while at other times it appears totally random. The reason for this is that human beings are neither totally predictable nor totally random. Although it is nearly impossible to determine a person's reaction to information or situations, there are always some basic trends in behavior as well as some random elements. The market is a collection of millions of people acting in a chaotic manner. It is as impossible to predict the behavior of a million people as it is to predict the behavior of one person. Investors are neither mostly psychological as predicted by technical analysis, nor logical as predicted by fundamental analysis. In conclusion, these methods work best when employed together. The major benefit of using a neural network then is for the network to learn how to use these methods in combination effectively, and hopefully learn how the market behaves as a factor of our collective consciousness.

#### V.APPLICATION OF NEURAL NETWORKS TO MARKET PREDICTION OVERVIEW

The ability of neural networks to discover nonlinear relationships in input data makes them ideal for modeling nonlinear dynamic systems such as the stock market. Various neural network configurations have been developed to model the stock market. Commonly, these systems are created in order to determine the validity of the EMH or to compare them with statistical methods such as regression. Often these networks use raw data and derived data from technical and fundamental analysis discussed previously. This section will overview the use of neural networks in financial markets including a discussion of their inputs, outputs, and network organization. For example, many networks prune redundant nodes to enhance their performance. The networks are examined in three main areas:

1. Network environment and training data
2. Network organization
3. Network performance

#### *Training A Neural Network*

A neural network must be trained on some input data. The two major problems in implementing this training discussed in the following sections are:

1. Defining the set of input to be used (the learning environment)
2. Deciding on an algorithm to train the network

#### *A) The Learning Environment*

One of the most important factors in constructing a neural network is deciding on what the network will learn. The goal of most of these networks is to decide when to buy or sell securities based on previous market indicators. The challenge is determining which indicators and input data will be used, and gathering enough training data to train the system appropriately. The input data may be raw data on volume, price, or daily change, but it may also include derived data such as technical indicators (moving average, trend-line indicators, etc.) or fundamental indicators (intrinsic share value, economic environment, etc.). Normalizing data is a common feature in all systems as neural networks generally use input data in the range [0, 1] or [-1, 1]. Scaling some input data may be a difficult task especially if the data is not numeric in nature.

#### *B) Network Training*

Training a network involves presenting input patterns in a way so that the system minimizes its error and improves its performance. The training algorithm may vary depending on the network architecture, but the most common training algorithm used when designing financial neural networks is the backpropagation algorithm. The most common network architecture for financial neural networks is a multilayer feedforward network trained using backpropagation. Backpropagation is the process of backpropagating errors through the system from the output layer towards the input layer during training. Backpropagation is necessary because hidden units have no training target value that can be used, so they must be trained based on errors from previous layers. The output layer is the only layer which has a target value for which to compare. As the errors are backpropagated through the nodes, the connection weights are changed. Training occurs until the errors in the weights are sufficiently small to be accepted. It is interesting to note that the type of activation function (sigmoid function) used in the neural network nodes can be a factor on what data is being learned.

#### *Network Organizations*

The most common network architecture used is the backpropagation network. However, stock market prediction networks have also been implemented using genetic algorithms, recurrent networks, and modular networks. Backpropagation networks are the most commonly used network because they offer good generalization abilities and are relatively straightforward to implement. Although it may be difficult to determine the optimal network configuration and network parameters, these networks offer very good performance when trained appropriately. Genetic algorithms are especially useful where the input dimensionality is large. They allowed the network developers to automate network configuration without relying on heuristics or trial-and-error.

*Recurrent network* architectures are the second most commonly implemented architecture. The motivation behind using recurrence is that pricing patterns may repeat in time. A *self-organizing* system was also developed to predict stock prices. The self-organizing network was designed to construct a nonlinear chaotic model of stock prices from volume and price data. Features in the data were automatically extracted and classified by the system. The benefit in using a self-organizing neural network is it reduces the number of features (hidden nodes) required for pattern classification, and the network organization is developed automatically during training. Interesting *hybrid network architecture* was developed, which combined a neural network with an expert system. The neural network was used to predict future stock prices and generate trading signals. The expert system used its management rules and formulated trading techniques to validate the neural network output. If the output violated known principles, the expert system could veto the neural network output, but would not generate an output of its own. This architecture has potential because it combines the nonlinear prediction of neural networks with the rule-based knowledge of expert systems. Thus, the combination of the two systems offers superior knowledge and performance. There is no one correct network organization. Each network architecture has its own benefits and drawbacks. Backpropagation networks are common because they offer good performance, but are often difficult to train and configure. Recurrent networks offer some benefits over backpropagation networks because their "memory feature" can be used to extract time dependencies in the data, and thus enhance prediction. More complicated models may be useful to reduce error or network configuration problems, but are often more complex to train and analyze.

#### Network Performances

A network's performance is often measured on how well the system predicts market direction. Ideally, the system should predict market direction better than current methods with less error. Some neural networks have been trained to test the EMH. If a neural network can outperform the market consistently or predict its direction with reasonable accuracy, the validity of the EMH is questionable. Other neural networks were developed to outperform current statistical and regression techniques. Many of the first neural networks used for predicting stock prices were to validate the EMH. The EMH, in its weakest form, states that a stock's direction cannot be determined from its past price. Several contradictory studies were done to determine the validity of this statement. The JSE-system demonstrated superior performance and was able to predict market direction, so it refuted the EMH. A backpropagation network which only used past share prices as input had some predictive ability, which refutes the weak form of the EMH. In contrast, an earlier study on IBM stock movement [11] did not find evidence against the EMH. However, the network used was a single-layer feedforward network, which does not have a lot of generalization power. Overall, neural networks are able to partially predict share prices, thus refuting the EMH. Neural networks have also been compared to statistical and regression techniques. The ultimate

goal is for neural networks to outperform the market or index averages. Returns from various networks (i.e. Backpropagation network, single layer network etc.) are recorded and compared during test. A large performance increase resulted by adding "windowing". Windowing is the process of remembering previous inputs of the time series and using them as inputs to calculate the current prediction. Recurrent networks returned 50% using windows, and cascade networks returned 51%. Thus, the use of recurrence and remembering past inputs appears to be useful in forecasting the stock market.

#### VI. FUTURE WORK

Using neural networks to forecast stock market prices will be a continuing area of research as researchers and investors strive to outperform the market, with the ultimate goal of bettering their returns. Network pruning and training optimization are two very important research topics which impact the implementation of financial neural networks. Financial neural networks must be trained to learn the data and generalize, while being prevented from overtraining and memorizing the data. The major research thrust in this area should be determining better network architectures. The commonly used backpropagation network offers good performance, but this performance could be improved by using recurrence or reusing past inputs and outputs. The architecture combining neural networks and expert systems shows potential. Neural networks appear to be the best modeling method currently available as they capture nonlinearities in the system without human intervention. Continued work on improving neural network performance may lead to more insights in the chaotic nature of the systems they model.

#### VII. CONCLUSION

This paper surveyed the application of neural networks to financial systems. It demonstrated how neural networks have been used to test the Efficient Market Hypothesis and how they outperform statistical and regression techniques in forecasting share prices. Although neural networks are not perfect in their prediction, they outperform all other methods and provide hope that one day we can more fully understand dynamic, chaotic systems such as the stock market.

#### VIII. REFERENCES

- [1]. Dirk Emma Baestaens and Willem Max van den Bergh. Tracking the Amsterdam stock index using neural networks. In *Neural Networks in the Capital Markets*, chapter 10, pages 149–162. John Wiley and Sons, 2000.
- [2]. K. Bergerson and D. Wunsch. A commodity trading model based on a neural network-expert system hybrid. In *Neural Networks in Finance and Investing*, chapter 23, pages 403–410. Probus Publishing Company, 2003.
- [3]. Robert J. Van Eyden. *The Application of Neural Networks in the Forecasting of Share Prices*. Financeand Technology Publishing, 1999.
- [4]. K. Kamijo and T. Tanigawa. Stock price pattern recognition: A recurrent neural network approach. In *Neural Networks in Finance and Investing*, chapter 21, pages 357–370. Probus Publishing Company, 1998.
- [5]. T. Kimoto, K. Asakawa, M. Yoda, and M. Takeoka. Stock market prediction system with modular neural networks. In *Proceedings of the International Joint Conference on Neural Networks*, volume 1, pages 1–6, 1990.
- [6]. C. Klimasauskas. Applying neural networks. In *Neural Networks in Finance and Investing*, chapter 3, pages 47–72. Probus Publishing Company, 1993.

# Swarm Intelligence

Gurdip Kaur, Jatinder Pal Singh, Computer Science and Engineering Department, CIET, Rajpura,  
[ergurdip85@gmail.com](mailto:ergurdip85@gmail.com) [rasamrit@hotmail.com](mailto:rasamrit@hotmail.com)

**Abstract-Swarm Intelligence is an AI technique that focuses on studying the collective behavior of a decentralized, self-organized system made up by a population of simple agents interacting locally with each other and with the environment. Although there is typically no centralized control dictating the behavior of the agents, local interactions among the agents often cause an “intelligent” global pattern to emerge. Examples of systems like this can be found abundant in nature, including ant colonies, bird flocking, animal herding, honey bees, bacterial growth, fish schooling and many more. The “swarm-like” algorithm, such as Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO), has already been applied successfully to solve real-world optimization problems in engineering and telecommunication. Research in swarm intelligence can be classified according to different criteria: - Natural vs. Artificial and Scientific vs. Engineering. Natural/Scientific research includes Foraging Behavior of Ants. Deneubourg in 1990 has shown that this behavior can be explained via a simple probabilistic model in which each ant decides where to go by taking random decisions based on the intensity of pheromone perceived on the ground, the pheromone being deposited by the ants while moving from the nest to the food source and back. Artificial/Scientific research includes Clustering by a Swarm of Robots. Deneubourg in 1991 proposed a model that ants pick up and drop items with probabilities that depend on information on corpse density which is locally available to the ants. Natural/Engineering work is undergoing promising research. Artificial/Engineering includes Swarm-based Data Analysis.**

**Keywords-** Swarm, collective behavior, decentralized, self-organized.

## I. INTRODUCTION

The kind of aggregated behavior shown by the colonies of ants, flocks of birds, herds of animals, honey bees is called Swarm Intelligence and the groups are called swarms. A swarm has been defined as a set of (mobile) agents which are liable to communicate directly or indirectly (by acting on their local environment) with each other, and which collectively carry out a distributed problem solving. And Technically, Swarm Intelligence is a type of Artificial Intelligence based on the collective behavior of decentralized and self organized systems that consist of number of agents or boids interacting with one another and with the environment as shown in Fig. 1. The concept was introduced by Gerardo Beni and Jing Wang in 1989. The agents follow very simple rules, and although there is no centralized control depicting how the individual agents behave, local and a certain degree random, interactions between such agents lead to the emergence of “intelligent” global behavior, unknown to the agents [1].



Figure 1: Swarm Intelligence in Honey Bees

### Properties of a Swarm Intelligence System:

- Composed of many individuals
- Relatively homogeneous individuals (agents are identical)
- The interactions among the individuals are based on simple behavioral rules that exploit only local information that the individuals exchange directly or via the environment (stigmergy)
- The overall behavior of the system results from the interactions of individuals with each other and with their environment, that is, the group behavior self-organizes.

The characterizing property of a swarm intelligence system is its ability to act in a coordinated way without the presence of a coordinator. Many examples observed in nature of swarms that perform some collective behavior without any individual controlling the group, or being aware of the overall group behavior are ant colonies, bird flocking, animal herding, honey bees, bacterial growth, fish schooling etc. Some human artifacts also fall into the domain of swarm intelligence, such as some multi-robot systems, and also certain computer programs that are written to tackle optimization and data analysis problems [3].

For example, for a bird to participate in the flock, it only adjusts its movements to coordinate with the movements of its flock-mates especially with the neighboring birds. A bird in a swarm simply tries to remain close to its neighbors but avoids any collisions.

### Reasons to form Swarm

- Defense against predators
  - ❖ Enhanced predator detection
  - ❖ Minimizing chance of capture
- Enhanced foraging success
- Better chances to find a mate
- Decreased energy consumption
- Ease in finding food, building nest, feeding the brood.

### Characteristics of Swarm Intelligence

- **No Central Control:** - The swarms do not have any leader to command them. They simply remain close to their neighbors without knowing that they are working collaboratively.
- **Simple rules:** - Each agent in a swarm follows the simple rule that is they remain closer and do not collide with their neighbors.
- Self-Organization
- Defense
- Ease to find food, mate and build a nest
- Enhanced task performance
- Efficient

### Boid Rules

#### Rule 1: Collision Avoidance

The boids or agents try to avoid any collision with the neighbors and maintain a certain distance as shown in fig. 2.

#### Rule 2: Velocity Matching

Each boid matches the velocity with the neighboring boids in order to be in a group as shown in fig. 3.

#### Rule 3: Flock Centering

Each boid remains close to the neighboring boid and keeps in mind that the first two rules are obeyed. Fig. 4 represents flock centering [8].

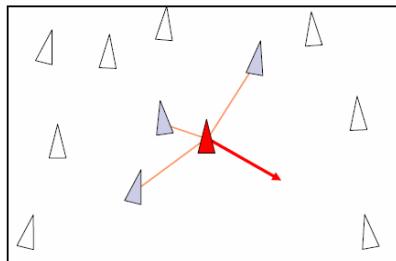


Figure 2: Avoidance of collision with neighboring boids

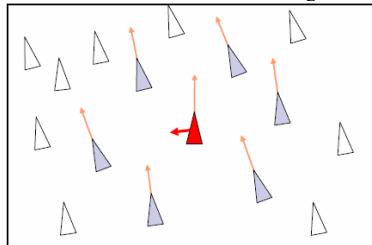


Figure 3: Velocity matching in a swarm

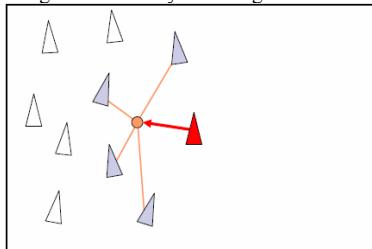


Figure 4: Flock Centering

### Types of Behaviors involved in Swarm Intelligence

- Emergent Behavior
- Stigmergy (Indirect communication and coordination, by local modification and sensing of the environment).



Figure 5: Natural Examples of Swarms [7]

*Natural Examples of Swarm Intelligence*  
Examples of Swarm Intelligence existing in nature are as shown in Fig. 5

- Ant colonies
- Bird Flocking
- Animal Herding
- Honey Bees
- Bacterial growth
- Fish Schooling [1].

## II. EXAMPLE ALGORITHMS

### Ant Colony optimization (ACO)

Ant Colony Optimization is a class of optimization algorithms modeled on the actions of an ant colony. ACO methods are useful in the problems that need to find paths to goals. Artificial ‘ants’ – simulation agents – locate optimal solutions by moving through a parameter space representing all possible solutions. Real ants lay down pheromones directing each other to resources while exploring their environment. The simulated ‘ants’ similarly record their positions and quality of their solutions, so that in later simulation iterations more ants locate better solutions. One variation on this approach is Bee’s algorithm, which is more analogous to the foraging patterns of the honey bee [3, 6].

### Particle Swarm optimization (PSO)

Particle Swarm optimization (PSO) is a global optimization algorithm dealing with problems in which a best solution can be represented as a point or surface in an n-dimensional space. Hypothesis are plotted in this space and seeded with an initial velocity, as well as a communication channel between the particles. Particles then move through the solution space, and are evaluated according to some fitness criterion after each time step. Over time particles are accelerated towards those particles within their communication grouping which have better fitness value. The main advantage of such an approach over other global minimization strategies such as simulated annealing is that the large numbers of

members that make up the particle swarm make the technique impressively resilient to the problem of local minima [3].

#### *Stochastic Diffusion Search (SDS)*

Stochastic Diffusion Search (SDS) is an agent-based probabilistic global search and optimization technique best suited to problems where the objective function can be decomposed into multiple independent partial-functions. Each agent maintains a hypothesis which is iteratively tested by evaluating a randomly selected partially objective function parameterized by the agent's current hypothesis. In the standard version of SDS such partial function evaluations are binary, resulting in each agent becoming active or inactive. Information on hypothesis is diffused across the population via inter-agent communication. Unlike the stigmergic communication used in ACO, in SDS agents communicate hypothesis via a one-to-one communication strategy analogous to the tandem running procedure observed in some species of ant. A positive feedback mechanism ensures that, over time, a population of agents stabilizes around the global-best solution. SDS is both an efficient and robust search and optimization algorithm, which has been extensively mathematically observed [3].

### III. CLASIFICATION OF RESEARCH IN SWARM INTELLIGENCE

Research in swarm intelligence can be classified as:

- **Natural vs. Artificial:** - It is customary to divide swarm intelligence research into two areas according to the nature of the systems under analysis. We speak therefore of *natural* swarm intelligence research, where biological systems are studied; and of *artificial* swarm intelligence, where human artifacts are studied [3].
- **Scientific vs. Engineering:** - An alternative and somehow more informative classification of swarm intelligence research can be given based on the goals that are pursued: we can identify a *scientific* and an *engineering* stream. The goal of the scientific stream is to model swarm intelligence systems and to single out and understand the mechanisms that allow a system as a whole to behave in a coordinated way as a result of local individual-individual and individual-environment interactions. On the other hand, the goal of the engineering stream is to exploit the understanding developed by the scientific stream in order to design systems that are able to solve problems of practical relevance [3].

#### *Natural/Scientific: Foraging Behavior of Ants*

In a now classic experiment done in 1990, Deneubourg and his group showed that, when given the choice between two paths of different length joining the nest to a food source, a colony of ants has a high probability to collectively choose the shorter one. Deneubourg has shown that this behavior can be explained via a simple probabilistic model in which each ant decides where to go by taking random decisions based on the intensity of pheromone perceived on the ground, the pheromone being deposited by the ants while moving from the nest to the food source and back [3].

#### *Artificial/Scientific: Clustering by a Swarm of Robots*

Several ant species cluster corpses to form cemeteries. Deneubourg et al. (1991) were among the first to propose a distributed probabilistic model to explain this clustering behavior. In their model, ants pick up and drop items with probabilities that depend on information on corpse density which is locally available to the ants. Beckers et al. (1994) have programmed a group of robots to implement a similar clustering behavior demonstrating in this way one of the first swarm intelligence scientific oriented studies in which artificial agents were used [3].

#### *Natural/Engineering: Exploitation of collective behaviors of animal societies*

A possible development of swarm intelligence is the controlled exploitation of the collective behavior of animal societies. No example is available in this area of swarm intelligence although some promising research is currently in progress: For example, in the *Leurre* project, small insect-like robots are used as lures to influence the behavior of a group of cockroaches. The technology developed within this project could be applied to various domains including agriculture and cattle breeding [3].

#### *Artificial/Engineering: Swarm-based Data Analysis*

Engineers have used the models of the clustering behavior of ants as an inspiration for designing data mining algorithms. A



Figure 6: Clustering behavior of Ants

Seminal work in this direction was undertaken by Lumer and Faieta in 1994. They defined an artificial environment in which artificial ants pick up and drop data items with probabilities that are governed by the similarities of other data items already present in their neighborhood. The same algorithm has also been used for solving combinatorial optimization problems reformulated as clustering problems (Bonabeau et al. 1999) [3].

### IV. APPLICATIONS OF SWARM INTELLIGENCE

A few examples of scientific and engineering Swarm Intelligence are:

#### *Clustering behavior of Ants*

Ants build cemeteries by collecting dead bodies into a single place in the nest. They also organize the spatial disposition of larvae into clusters with younger, smaller larvae in cluster center and the older ones into its periphery. This clustering behavior has motivated a number of scientific studies. Scientists have built simple probabilistic

models of these behaviors. The basic models state that an unloaded ant has probability to pick up a corpse or a larva that is inversely proportional to their locally perceived density, while the probability that a loaded ant has to drop the carried item is proportional to the local density of similar items. This model has been validated against experimental data obtained with real ants. This is an example of natural/scientific swarm Intelligence System [2, 3, 6] as shown in fig. 6.

#### *Flocking and Schooling in Birds and Fish*

Flocking and schooling are the examples of highly coordinated group behaviors exhibited by large group of birds and fish. Scientists have shown that these elegant swarm-level behaviors can be understood as the result of a self-organized process where no leader is in charge and each individual bases



Figure 7: A Swarm of Robots collaborates to pass an obstacle [9].

its movement decisions solely on locally available information: the distance, perceived speed, direction of movement of neighbors. These studies have inspired a number of computer simulations that are now used in computer graphics industry for the realistic reproduction of flocking in movies and computer games. These are examples respectively of natural/scientific and artificial/engineering swarm intelligence systems [3].

#### *Building Nests*

Wasps build nests with a highly complex internal structure that is well beyond the cognitive capabilities of a single wasp. Termites build nests whose dimensions (they can reach many meters of diameter and height) are enormous when compared to a single individual, which can measure as little as a few millimeters. Scientists have been studying the coordination mechanisms that allow the construction of these structures and have proposed probabilistic models exploiting stigmergic communication to explain the insect's behavior. Some of these models have been implemented in computer programs. This is an example of natural/scientific swarm intelligence system [3].

#### *Swarm based Network Management*

The algorithms used in mobile ad-hoc networks and internet like networks are another example of successful artificial/engineering swarm intelligence system [3, 5].

#### *Cooperative Behavior in Swarms of Robots*

There are a number of swarm behaviors observed in natural systems that have inspired innovative ways of solving problems by using swarms of robots. This is what is called swarm robotics. In other words, swarm robotics is the application of swarm intelligence principles to the control of swarms of robots. An example of artificial/engineering swarm intelligence system is the collective transport of an item too heavy for a single robot as shown in Fig. 7, a behavior also often observed in ant colonies [2, 3, and 6].

#### *Ant Colony Optimization*

In ant colony optimization (ACO), a set of software agents called "artificial ants" search for good solutions to a given optimization problem transformed into the problem of finding the minimum cost path on a weighted graph. The artificial ants incrementally build solutions by moving on the graph. Examples are the application to routing in communication networks.

#### *Particle Swarm Optimization*

In particle swarm optimization (PSO), a set of software agents called particles search for good solutions to a given continuous optimization problem. Each particle is a solution of the considered problem and uses its own experience and the experience of neighbor particles to choose how to move in the search space. PSO has been applied to many different problems and is another example of successful artificial/engineering swarm intelligence system.

## V. CONCLUSION

Swarm Intelligence has seen an increased interest in the previous years because of the increasing applications and optimization techniques used to find the best path by using the algorithms. There is still a scope of finding the solutions of the unsolved mysteries. Swarm Robotics is an interesting application to work upon and with the invention of Einstein Robot, the area is open for exploration. In case of stampedes, the algorithms provided in the swarm intelligence area can be helpful to find the best path to proceed further. And Swarm Robotics may prove helpful in achieving India's mission to reach Mars by 2025. On the whole Swarm Intelligence started from the natural observations and is proceeding towards the advancements like swarm robotics.

## VI. REFERENCES

- [1]. Swarm Intelligence, Wikipedia, free Encyclopedia
- [2]. From Swarm Intelligence to Swarm Robotics, Gerardo Beni. <http://www.swarm-robotics.org/SAB04/presentations/beni-review.pdf>.
- [3]. [http://www.scholarpedia.org/wiki/index.php?title=Swarm\\_intelligence&action=cite&rev=40214](http://www.scholarpedia.org/wiki/index.php?title=Swarm_intelligence&action=cite&rev=40214)
- [4]. E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial System*. Oxford University Press, New York, 2009.
- [5]. G. Di Caro and M. Dorigo. AntNet: Distributed stigmergetic control for communications networks. *Journal of Artificial Intelligence Research*, 9:317-365, 1998
- [6]. M. Dorigo and T. Stützle. *Ant Colony Optimization*. MIT Press, Cambridge, MA, 2004.
- [7]. [www.youtube.com/Video\\_3.flv](http://www.youtube.com/Video_3.flv)
- [8]. Jonas Pfeil, Swarm Intelligence Slides
- [9]. Swarm-bots,marco Dorigo, 2005

# Design of Low Power FIR Filter Coefficients Using Genetic Algorithm (Optimization)

Shaveta Goyal, Prof. J.P.S Raina\*

Swami Devi Dyal Institute of Engineering & Technology, Barwala (INDIA), Baba Banda Singh Bahadur Engineering College, Fatehgarh Sahib (INDIA), ershavetagoyal@yahoo.co.in, \*jps.raina@bbsbec.ac.in

**Abstract-**With the explosive growth of wireless communication system and portable devices, the power reduction has become a major problem. Many of the communication system today utilize digital signal processors (DSP) to resolve the transmitted information. Finite

impulse response (FIR) filters have been and continued to be important building blocks in many digital processing systems (DSP). Hamming distance is a measure of switching activity corresponding to the number of energy consuming transition in multiplier and accumulate (MAC) of filter while implementing on digital signal processors (DSP). The hamming distance between consecutive coefficient values and the number of signal toggling in opposite directions thus forms the measure of bus power dissipation. Genetic algorithms can implemented as a computer simulation in which a population of abstract representations (called chromosomes or the genotype or the genome) of candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem evolves toward better solutions.

In this paper the hamming distance of fir filter is minimized by minimizing the switching activity using “Genetic Algorithms” optimization technique to reduce the power dissipation and to increase the battery life of portable multimedia devices.

## I. INTRODUCTION

A major problem associated with increases in the processing power and the sophistication of signal processing algorithms is the increasing levels of power dissipation.

## II.LOW POWER DESIGN METHODOLOGY

An optimization that could be done at this level is driven by voltage scaling. It is necessary to scale supply voltage for a quadratic improvement in energy per transition. Unfortunately, we pay a speed penalty for a V<sub>dd</sub> reduction with delays increasing, as V<sub>dd</sub> approaches the threshold voltage of the devices. The simple first order relationship between V<sub>dd</sub> and gate delay, t<sub>d</sub> for a CMOS gate is given in (1.1),

$$t_d \propto \frac{1}{(V_{dd} - V)^2} \quad (1.1)$$

The objective is to reduce power consumption while keeping the throughput of the overall system fixed.

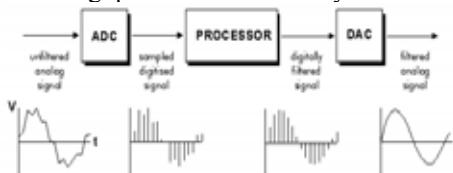


Fig 1: implementation of filter on digital signal processor

An optimization or a mathematical programming problem can be stated as follows:

$$\text{Find } x = [x_1 x_2 x_3 \dots x_n]$$

Subjected to the constraints

$$G_j(x) \leq 0, \quad j = 1, 2, \dots, m$$

$$H_j(x) = 0, \quad j = 1, 2, \dots, p$$

Where x is an n-dimensional design vector,

F(x) is termed the objective function and g<sub>j</sub>(x) and h<sub>j</sub>(x) are known as inequality and equality constraints, respectively.

### 2.1 objective

Finite impulse response (fir) filter is implemented as a series of multiply and accumulate operations on a programmable digital signal processor (dsp). The multiply and accumulate (mac) unit of a digital signal processor experiences high switching activity due to signal transitions which results in higher power dissipation. Hamming distance forms a measure of the switching activity during implementation of the filter. The objective of the paper is to minimize the hamming distance and reduce the signal toggle by using optimization technique, genetic algorithm (ga), so that its power dissipation is reduced while its implementation on a digital signal processor.

The purpose of the optimization is to choose the best one of many acceptable designs available. Thus a criterion has to be chosen for comparing the different alternative acceptable design and for selecting the one. The criterion, with respect to which the design is optimized, when expressed as a function of the design variables, is known as objective function. If f<sub>1</sub>(x) and f<sub>2</sub>(x) denote two objective functions, a new objective Function for optimization is constructed as

$$F(x) = a_1 f_1(x) + a_2 f_2(x)$$

Where f(x) is a new objective function, a<sub>1</sub> and a<sub>2</sub> are constants whose values indicate the relative importance of one objective function relative to the other.

### 2.2 description

Genetic algorithm is an emerging optimization algorithm for signal processing and considered a powerful optimizer in away areas. The ga has been demonstrated a powerful method for these multi objective problems, enabling to obtain the pareto optimal set instead of single solution. Genetic Algorithms (gas) were invented by john holland and developed by him and his students and colleagues.

### 2.3 Search Space

The space of all feasible solutions (the set of solutions among which the desired solution resides) is called search space (also state space). Each point in the search space represents one possible solution. Each possible solution can be "marked" by its value (or fitness) for

the problem. With GA we look for the best solution among a number of possible solutions - represented by one point in the search space.

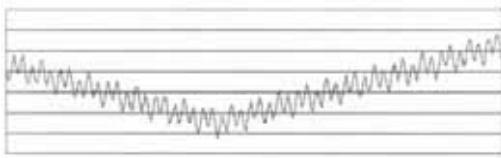


Fig 2: G.A. Search Space

#### 2.4 Working Principle

To illustrate the working principle of GA consider a unconstrained optimization problem

Maximize  $f(X)$

$$X_i^L \leq X_i \leq X_i^U \quad \text{for } i = 1, 2, \dots, N$$

If  $f(X)$ , for  $f(X) > 0$  is to be minimized, then the objective function is written as maximize

$$\frac{1}{1+f(X)}$$

#### III. ENCODING

Since genetic algorithms search directly in the solution space, it needs a way to encode solutions in a way that can be manipulated by the genetic algorithm.

##### Binary Encoding

In binary encoding, every chromosome is a string of bits - 0 or 1.

Chromosome A	101100101100101011100101
Chromosome B	111111000011000011111

Table1: Chromosomes with Binary Encoding

##### Permutation Encoding

In permutation encoding, every chromosome is a string of numbers that represent a position in a sequence.

Chromosome A	1 5 3 2 6 4 7 9 8
Chromosome B	8 5 6 7 2 3 1 4 9

Table2: Permutation Encoding

#### 3.1 Rank Selection

Rank selection ranks the population first and then every chromosome receives fitness value determined by this ranking. The worst case will have the fitness 1, the second worst 2 etc. and the best will have fitness N (number of chromosomes in population).

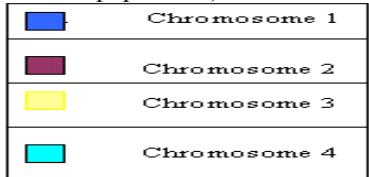


Fig 3: Situation before Ranking

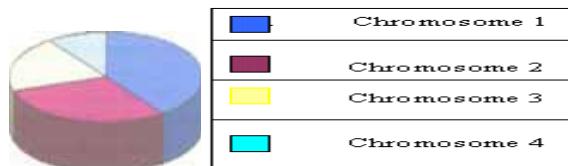


Fig4: Situations After Ranking (Graph Of Fitness)

#### 3.2 Hamming Distance Minim. Algorithm Problem Definition

The Hamming distance minimization problem using Steepest Descent approach stated as follows For a Given N-tap FIR filter with coefficient  $A_i$ ,  $i = 0, N-1$  that satisfy the filter response in terms of pass band ripples, stop band attenuation and linear phase, find a new set of coefficient  $A_i$ ,  $i = 0, N-1$  such that the total Hamming distance between successive coefficients is minimized while still satisfied the desired filter characteristics in terms of pass band ripple and stop band attenuation.

##### Coefficient Scaling

The first phase of the algorithm involves uniformly scaling the coefficient so as to reduce the total Hamming distance between successive coefficients. For N-tap filter with N coefficients

( $A_i$ ,  $i = 0, N-1$ ), the output  $Y(n)$  is given by equation.

$$N-1$$

$$Y(n) = \sum_{i=0}^{N-1} (A_i * X_{n-i})$$

Genetic Algorithms are successfully used for the design of FIR filters. The problem is formulated as error minimization between the Ideal frequency response and the desired frequency response as per the design specification in terms of pass band ripple, stop band attenuation and linear phase. Here one more objective added is the Hamming distance between the successive values of the designed filter should be minimum than the ideal filter coefficients. As the Hamming Distance is the measure of the signal switching activity it should also be minimized to reduce the power dissipation in the multipliers while implementing the FIR filtering operation on digital signal processors. So the problem is multi objective optimization problem and it is solved by using weighted sum approach, converting the problem into single objective by assigning the appropriate weights. The problem is then solved using Genetic Algorithms.

#### IV. SOLUTION METHODOLOGY

The intent of work is to optimize the coefficients of FIR filter, to minimize the Hamming distance and satisfying the desired filter characteristics in terms of pass band ripple and stop band attenuation.

The multi objective problem of minimizing the Hamming distance and mean square error is converted into a scalar problem by constructing a weighted sum of the objectives to generate Pareto optimal solution. The Pareto optimal solutions for different simulated weight combination are generated considering both the objectives simultaneously. To simulate weight combination, weights  $w_i$ ,  $i = 1, 2, \dots, L$  are varied from

0.1 to 1.0 in steps 0.1, so that their sum is 1.0. The weighting coefficients  $w_1$  and  $w_2$  are used to select of error and Hamming distance. The weighted objective function is written as

$$F = w_1 f_M + w_2 f_H$$

Here  $f_M$  and  $f_H$  are the fitness functions for Mean square error and Hamming distance. The scalar optimization mentioned above is solved using Genetic Algorithm the random number population is generated and the chromosomes are selected based upon the maximum fitness of the fitness function using the roulette wheel selection. The Genetic operator's crossover and mutation are applied; uniform crossover is applied at the defined in a mating pool to produce the new generation which are maximally fit.

#### 4.2 Methodology / Planning of Work

The Hamming Distance minimization problem is formulated as a local search problem, where the optimum coefficient values are searched in their neighborhood. This is done by using an iterative improvement process. During the each iteration one or more coefficients are suitably modified so as to reduce the total Hamming distance while still satisfying the desired filter characteristics. The optimization process continues till no further reduction is possible.

The coefficient optimization is done in two phases:

In the first phase, all the coefficients are scaled uniformly. The advantage of such an approach is that it does not affect the filter characteristics in terms of pass band ripples and stop band attenuation and phase response. The sealing results in the same gain /attenuation ratio.

In the second phase of optimization one coefficient is perturbed in the each iteration. In case of requirement to retain the linear phase characteristics, the coefficients are perturbed in pairs ( $A_i$  and  $A_{n-1-i}$ ) so as to preserve coefficients symmetry. The selection of coefficient for perturbation and the amount of perturbation has the direct impact on overall optimization quality. Various strategies can be adopted for coefficient perturbation. The strategies adopted here include ‘Genetic Algorithms’. The Genetic Algorithms are the evolutionary algorithm which generates the random numbers and selects the best fit value according to the fitness function and search the whole space to find the global value.

#### 4.3 Simulation Results

*Output of the ‘c’ programme for calculating hamming distance*

Enter the no. Of parameters: 8

ENTER THE VALUES OF PARAMETERS IN FRACTION.....

.1234

.2345

.3456

.4567

.5678

.6789

.7890

.8901

The values of parameters are:

0.123400 0.234500 0.345600 0.456700 0.567800 0.678900

0.789000 0.890100

Truncated values are following:

0.123400  
0.234500  
0.345600  
0.456700  
0.567800  
0.678900  
0.789000  
0.890100

Binary equivalent of entered coefficients  
BINARY EQUIVALENT OF 0.123400 IS 111110/10 TO POWER 9  
BINARY EQUIVALENT OF 0.234500 IS 1111000/10 TO POWER 9  
BINARY EQUIVALENT OF 0.345600 IS 10110000/10 TO POWER 9  
BINARY EQUIVALENT OF 0.456700 IS 11101000/10 TO POWER 9  
BINARY EQUIVALENT OF 0.567800 IS 100100010/10 TO POWER 9  
BINARY EQUIVALENT OF 0.678900 IS 101011010/10 TO POWER 9  
BINARY EQUIVALENT OF 0.789000 IS 110010010/10 TO POWER 9  
BINARY EQUIVALENT OF 0.890100 IS 111000110/10 TO POWER 9  
EXOR OF 0.123400 AND 0.234500 IS 1000110/10 TO POWER 9  
EXOR =70\N  
NUMBER OF ONES =3

Similarly other calculations for Number of Ones can be done. Thus

Total Hamming Distance among the Filter Coefficients = 24.

**Output of matlab designing code for windows:**

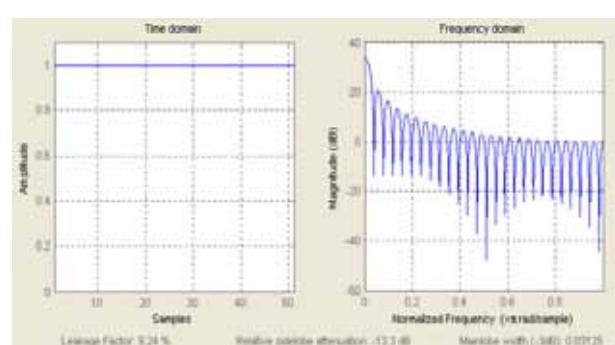


Fig 5: Rectangular Window in Matlab

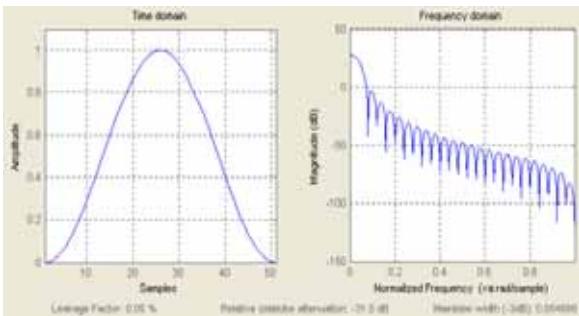


Fig 6: Hamming Window in Matlab

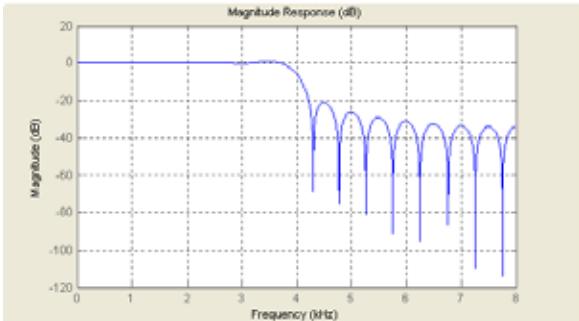


Fig7: Low Pass Filter Normal Response

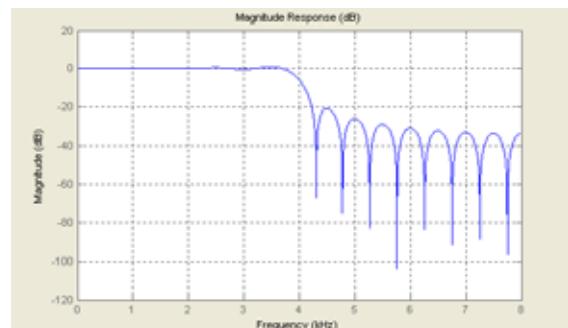


Fig8: Response with Genetic Algorithm

## V. REFERENCES

- [1]. Michel. R. Lightner and Stephen. W. Director, “*Multicriterion Optimization for the Design of Electronic Circuits*”, IEEE Transactions on Circuits and Systems, Vol. CAS-28, No. 3, March 1981. pp. 169-179.
- [2]. Christakis Charalambous, “*A New Approach to Multicriterion Optimization Problem and Its application to the Design of 1-D Digital Filters*”, IEEE Transactions on Circuits and Systems, Vol. 36, No. 6, June 1989. pp.773 -784.
- [3]. G. Wade, A. Roberts and G. Williams, “*Multiplier-less Filter Design using a Genetic Algorithm*”, IEE proceedings on Vis. Image Signal processing, Vol. 44, No.3, June 1994. pp 175.
- [4]. Darren N. Pearson, Keshab K. Parhi, “*Low Power digital Filter Architectures*”, IEEE Symposium on Circuit and Systems, vol. 1, 28 April to 31 May 1995. pp. 231-234.
- [5]. William L. Fréking and Keshab Parhi, “*Low Power Residue Digital Filter using Residue Arithmetic*”, IEEE 31st Asilomar Conference on Signal, Systems and Computers Vol.1 Nov.2-5, 1998. pp. 739-743.

# Immunoinformatics: The Roll of Computers in Medicine

Dr. Vidushi Bharadwaj Department of Applied Science, Haryana Engineering College, Jagadhri  
dr.vidushibharadwaj@gmail.com.

With the rapid growth in business size, today's business orient. Computers in medicine are a medium of international communication of the revolutionary advances being made in the application of the computer, to the field of bioscience and medicine. The awareness about using computers in medicine has recently increased all over the world. Hope this will help starting changing the culture, promoting the awareness about the importance of implementing information technology (IT) in medicine. One of the useful application of health care data is to understand the pattern of diseases like human immunodeficiency virus or HIV, which is now most deadly disease among all over world. The overall intention of AIDMED is to address the need which medical personal have for easy access to wide variety of information and data sources available now, or soon in computerized form.

**Keywords:** AiDMED, Noninfectious, Health care,Psoriasis

## I. INTRODUCTION

With the advent of science ,the Computers in medicines are becoming a medium of international communication of the revolutionary discoveries being made in the application of the Computer to the fields of bioscience and medicine .The awareness about using computers in medicines ,has recently increased Globally ,helping researchers to diagnose a patient correctly and curing accordingly Regarding the role of Computers in the medicines ,following are few basic concerned terms

- 1 **Digital Library (D.Lib)** -The field of Digital library has always been poorly defined a discipline of amorphous borders and crossroads however its role is becoming very important for medical teaching learning and health care development.
- 2 **Webopedia -(e-encyclopaedia)-** is an online dictionary and search engine for computer and internet technology.
- 3 **Evidence Based Medicine(EBM)-** provides the information regarding type of providing care to the patient and how to provide it.
- 4 **Artificial Intelligence(AI)** - is the study of ideas which enables computers to perform the things that make people seem intelligent. Artificial intelligence in medicine (AIM) is AI specialized to medical applications e.g. a large database collection of clinical histories of patients.
5. **Medical Informatics** : is that scientific field that deals with the storage, retrieval and optimal use of information and data in medicine. Also called healthcare informatics or bio-medical informatics. The final objective of bio-medical informatics is the coalescing of data, knowledge and the tool necessary to apply that data and knowledge in the decision making process, at the time and place that a decision needs to be made.

6. **Natural Language Processing(NLP)** : is a range of computer techniques for analyzing and representing naturally occurring text at one or more levels of linguistic analysis for the purpose of achieving human like language processing for knowledge intensive application for medicine, it is termed as Medical Language Processing (MLP).
7. **State of the Art** : generally is the highest level of development, very up-to- date, as of a device technique or specific field achieve that a particular time .Computer Aided Learning(CAL),Computer Aided Detection(CAD),Computer Aided Surgery(CAS)etc. are few examples relating to the State of the Art Computer Based Medicine.
8. **AIDMED** (Assistant for interacting with multimedia,medical databases): The overall intention of AIDMED is to address the need which medical personnel have for easy access to wide variety of information and data sources available in computerized form.

**T-cell:**A new paradigm of vaccine design is now emerging, following essential discoveries in immunology and the development of bioinformatics tools for T-cell epitope prediction from primary protein sequences. One rationale for this new paradigm is that following exposure to a pathogen, epitope-specific memory T-cell clones are established. These clones respond rapidly and efficiently upon any subsequent infection, elaborating cytokines, killing infected host cells, and marshalling humoral and cellular defences against the pathogen. The most efficient immune response to some pathogens is derived from a number of different T cells that respond to an ensemble of pathogen-derived short peptides called epitopes.Whether an immune response is directed against a single immunodominant epitope or against many epitopes, the generation of a protective immune response does not require the development of T-cell memory to every possible peptide in the entire pathogen. T-cell response to the ensemble of epitopes, not the whole pathogen, is the source from which a protective immune response is derived.

## II. CLINICAL IMMUNOLOGY

Clinical immunology is the study of diseases caused by disorders of the immune system (failure, aberrant action, and malignant growth of the cellular elements of the system). It also involves diseases of other systems, where immune reactions play a part in the pathology and clinical features. The diseases caused by disorders of the immune system fall into two broad categories: immunodeficiency, in which parts of the immune system fail to provide an adequate response (examples include chronic granulomatous disease), and autoimmunity, in which the immune system attacks its own

host's body (examples include systemic lupus erythematosus, rheumatoid arthritis, Hashimoto's disease and myasthenia gravis). Other immune system disorders include different hypersensitivities, in which the system responds inappropriately to harmless compounds (asthma and other allergies) or responds too intensely.

The most well-known disease that affects the immune system itself is AIDS, caused by HIV. AIDS is an immunodeficiency characterized by the lack of CD4+ ("helper") T cells and macrophages, which are destroyed by HIV.

The study for the ways, to prevent transplant rejection, in which the immune system attempts to destroy allografts or xenografts falls under clinical immunology .

### III. HISTOLOGICAL EXAMINATION OF THE IMMUNE SYSTEM

Even before the concept of immunity was developed, the organs were characterized earlier, that would later proved to be part of the immune system. The key primary lymphoid organs of the immune system are thymus and bone marrow, and secondary lymphatic tissues such as spleen, tonsils, lymph vessels, lymph nodes, and skin. When health conditions warrant, immune system organs including the thymus, spleen, portions of bone marrow, lymph nodes and secondary lymphatic tissues can be surgically excised for examination while patients are still alive.

Many components of the immune system are actually cellular in nature and not associated with any specific organ but rather are embedded or circulating in various tissues located throughout the body.

### IV. IMMUNOTHERAPY

The use of immune system components to treat a disease or disorder is known as immunotherapy. Immunotherapy is most commonly used in the context of the treatment of cancers together with chemotherapy (drugs) and radiotherapy (radiation). However, immunotherapy is also often used in the immunosuppressed (such as HIV patients) and people suffering from other immune deficiencies or autoimmune diseases.

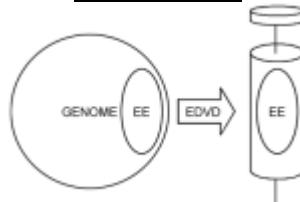
### V. DIAGNOSTIC IMMUNOLOGY

The specificity of the bond between antibody and antigen has made it an excellent tool in the detection of substances in a variety of diagnostic techniques. Antibodies specific for a desired antigen can be conjugated with a radiolabel, fluorescent label, or color-forming enzyme and are used as a "probe" to detect it.

**Epitope-driven vaccine design (EDVD):** An epitope ensemble (EE) is derived, using immuno-informatics, from a genome and inserted into a vaccine vehicle (epitope ensemble: set of CTL and Th epitopes necessary to induce a protective immune response). As has been observed for many vaccines, a protective immune response is not dependent on immunization with epitopes representing the entire pathogen. Instead, a set of epitopes, or epitope

ensemble, is sufficient to generate enough T memory cells to contain or eradicate the pathogen upon exposure. The epitope ensemble may contain T helper epitopes, cytotoxic T-cell epitopes, and B epitopes, or any combination of the three subsets. The observation that immunization with an epitope ensemble is sufficient for protection against pathogens has led to the development of sub-unit vaccines (such as hepatitis B vaccine, based on hepatitis B surface antigen). More recently, 'epitope-driven' vaccines are being developed, following the same reasons.

FIGURE OF EDVD



If an individual is previously exposed to a language, upon hearing just a few words of that language he/she will usually recognize, for example, that French or English is being spoken. Complete mastery of the language is not required for this recognition. Using this analogy to describe epitopes, one could say that they are pathogen-specific 'words' that alert the immune system to the presence of a pathogen. It is now possible to envisage the design of vaccines based on an ensemble of epitopes (a string of words, a few sentences, a paragraph, or a chapter) derived from the genome of a pathogen, using tools that have been developed in the field of immuno-informatics.

The two main components of immuno-informatics:(i) epitope mapping algorithms; and (ii) *in vitro* assays that confirm epitopes. Following a decade of discoveries related to MHC-T-cell interactions, tools that permit the scanning of protein sequences for T-cell epitopes have been developed and refined by a number of teams. A selection of these tools, including EpiMatrix (in a site limited to HIV or tuberculosis sequences), are performing well.

The field of immuno-informatics has also benefited from the development of a number of sensitive T-cell assays, such as the ELI spot assay, tetramers, and intracellular cytokine staining, that permit the measurement of vanishingly small numbers of epitope-specific T cells. Because of these tools, the breadth of T-cell responses *in vitro* can be determined more accurately. Achieving the same level of accuracy (detection of one epitope-specific T cell in among 5000 others) was simply not possible using older methodologies, such as T-cell proliferation and chromium release assays, due to the difficulty of separating T-cell responses from background noise. The union of marriage of epitope-mapping informatics tools with new sensitive *in vitro* screening methods are termed computational immunology, or immuno-informatics.

Computer science is giving scientists new ways to look at the virus that causes AIDS, perspectives that may help efforts to develop an effective vaccine and other medicines. Human immunodeficiency virus ( HIV) causes AIDS, a disease that weakens a person's immune system, leaving them vulnerable

to infections and other diseases that would rarely threaten a healthy person. A lot of underlying Computer science language is used to describe cell processes ,and then, the mathematics that is used to analyze programs, can also be applied to analyze cell activities because there is an underlying mathematical relationship

Many wellknown computer related firms and institutions are taking initiatives for computer based diagnosis of a patient and the remedial measures,Microsoft is the best example in the regard.The Microsoft has looked for ways to track how HIV mutates to evade the human immune system. Microsoft has recently released software code for the four tools which are helping researchers for analyzing the genetic make up of HIV, the genome is basically digital, it can described as a string and analyzed as astring. Microsoft researcher, are looking at ways to apply computer science to computational biology . Microsoft has also sought to apply machine-learning techniques, including technology used in spam and antivirus filters, to AIDS research. The goal is to find genetic patterns in HIV that can be used to "train" the human immune system to fight the virus. The AIDS virus has evolved the ability to decoy the immune system, have it actually work against itself, based on the analysis, that some of those decoys actually made it into the vaccine, so it was actually weakening the immune system.

## VI.CONCLUSION

With the increasing growth of the science & technology, the information technology( I T) in particular, the day to day life has become highly dependent on the fast processing things and naturally the medical field, also got affected too much from the technology of the computers. The depth of analyzing and diagnosing the cause of illness has enabled the computer to be a front runner. The various applications of AIM have been proved to be very useful. Establishing EBM information and resource center to help the promotion and use of EBM is becoming important,meeting the international standard of good health care.Also AI&Web based information have proved to be very useful and intelligent using them in all the fields of human life including health care , With the time to come computer based medical technology shall completely dominate the area of medicines.

## VII. REFERENCE

- [1]. van der Most RG, Sette A, Oseroff C *et al.* Analysis of cytotoxic T cell responses to dominant and subdominant epitopes during acute and chronic lymphocytic choriomeningitis virus infection. *J. Immunol.* 1996; 157: 5543–54. | PubMed | ChemPort |
- [2]. Gillespie GM, Wills MR, Appay V *et al.* Functional heterogeneity and high frequencies of cytomegalovirus-specific CD8(+) T lymphocytes in healthy seropositive donors. *J. Virol.* 2000; 74: 8140–50. | Article | PubMed | ISI | ChemPort |
- [3]. Gianfrani C, Oseroff C, Sidney J, Chesnut RW, Sette A. Human memory CTL response specific for influenza A virus is broad and multispecific. *Hum. Immunol.* 2000; 61: 438–52. | Article | PubMed | ChemPort |
- [4]. Harrer T, Harrer E, Kalams SA *et al.* Cytotoxic T lymphocytes in asymptomatic long-term nonprogressing HIV-1 infection. Breadth and specificity of the response and relation to *in vivo* viral quasispecies in a person with prolonged infection and low viral load. *J. Immunol.* 1996; 156: 2616–23. | PubMed | ISI | ChemPort |
- [5]. MICROSOFT TAKES COMPUTER SCIENCE INTO FIGHT AGAINST HIV,THURSDAY,NOVEMBER 06,2008,9:50 PM PST
- [8]. Leighton J, Sette A, Sidney J *et al.* Comparison of structural requirements for interaction of the same peptide with I-Ek and I-Ed molecules in the activation of MHC class II-restricted T-cells. *J. Immunol.* 1991; 147: 198–204. | PubMed | ChemPort |
- [9]. Lipford GB, Hoffman M, Wagner H, Heeg K. Primary *in vivo* responses to ovalbumin. Probing the predictive value of the K<sup>b</sup> binding motif. *J. Immunol.* 1993; 150: 1212–22. | PubMed | ChemPort |
- [10]. Parker KC, Bednarek MA, Coligan JE. Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side chains. *J. Immunol.* 1994; 152: 163–75. | PubMed | ISI | ChemPort |
- [11]. Meister GE, Roberts CGP, Berzofsky JA, De Groot AS. Two novel T cell epitope prediction algorithms based on MHC-binding motifs; comparison of predicted and published epitopes from *Mycobacterium tuberculosis* and HIV protein sequences. *Vaccine* 1995; 13: 581–91. | Article | PubMed | ISI | ChemPort |
- [12]. Brusik V, Rudy G, Harrison LC. Prediction of MHC binding peptides using artificial neural networks. In: RJ Stonier, XS Yu (eds). Complex Systems, Mechanisms of Adaption. Amsterdam. IOSPress, 1994; 253–60.
- [13]. Rosenfeld R, Zheng Q, Delisi C. Flexible docking of peptides to class I major-histocompatibility-complex receptors. *Genet. Anal.* 1995; 12: 1–21. | PubMed | ChemPort |
- [14]. Altuvia Y, Schueler O, Margalit H. Ranking potential binding peptides to MHC molecules by a computational threading approach. *J. Mol. Biol.* 1995; 249: 244–50. | Article | PubMed | ChemPort |
- [15]. Hammer J, Bono E, Gallazzi F, Belunis C, Nagy Z, Sinigaglia F. Precise prediction of major histocompatibility complex class II-peptide interaction based on peptide side chain scanning. *J. Exp. Med.* 1994; 180: 2353–8. | Article | PubMed | ISI | ChemPort |
- [16]. Sette A, Sidney J, Oseroff C *et al.* HLA DR4w4-binding motifs illustrate the biochemical basis of degeneracy and specificity in peptide-DR interactions. *J. Immunol.* 1993; 151: 3163–70. | PubMed | ChemPort |
- [17]. Goldsby RA, Kindt TK, Osborne BA and Kuby J (2003) **Immunology**, 5th Edition, W.H. Freeman and Company, New York, New York, ISBN 0-7167-4947-5
- [18]. Jaspan Heather, S.D Lawn; et al. "The maturing immune system: implications for development and testing HIV-1 vaccines for children and adolescents" AIDS21 Mar. 2006, Vol 20 p.p 483-494.
- [19]. Sizonenko PC, Paunier L. Hormonal changes in puberty III: Correlation of plasma dehydroepiandrosterone, testosterone, FSH, and LH with stages of puberty and bone age in normal boys and girls and in patients with Addison's disease or hypogonadism or with premature or late adrenarche. *J Clin Endocrinol Metab* 1975; 41:894–904.
- [20]. Verthelyi D. Sex hormones as immunomodulators in health and disease. *Int Immunopharmacol* 2001; 1:983–993.
- [21]. Stimson WH. Oestrogen and human T lymphocytes: presence of specific receptors in the T-suppressor/cytotoxic subset. *Scand J Immunol* 1998; 28:345–350.
- [22]. Benten WPM, Stephan C, Wunderlich F. B cells express intracellular but not surface receptors for testosterone and estradiol. *Steroids* 2002; 67:647–654.
- [23]. Beagley K, Gockel CM. Regulation of innate and adaptive immunity by the female sex hormones oestradiol and progesterone. *FEMS Immunol Med Microbiol* 2003; 38:13–22.

# Finite Element Modeling of MEMS Diaphragms of Capacitive Sensors

Lovely Chawla\*, Kapil Sachdeva\*\*

\*Assistant professor Indus Institute of Engineering and Technology Kinana, Jind

\*\* Lecturer Jind Institute of Engineering and Technology, Jind

**Abstract-** For designing and fabricating sensors and actuators, diaphragms play an important role. The diaphragm taken is rectangular whose ratio of length to width is changed, such higher the ratio of length vs. width with clamped edges. The understanding of deflection and stress of diaphragms on application of pressure is atmost important parameter for designing of capacitive pressure transducer. Capacitive pressure sensors is taken due to its robust structure and less sensitivity to enviornmental effects. The analysis of the stress distribution and deflection of diaphragms is done using Finite element analysis software ANSYS version 9.

## I. INTRODUCTION

MEMS stands for Micro-electromechanical systems, a manufacturing technology that enables the development of electromechanical systems using batch fabrication techniques similar to those used in integrated circuit (IC) design. MEMS integrate mechanical elements, sensors, actuators and electronics on a silicon substrate using a process technology called micro fabrication. Capacitive pressures sensor [1] fabricated through MEMS technology are mechanically similar to traditional sensors with the exception that they are in micrometer scale. The material taken for diaphragm of capacitive sensor is silicon with elastic property. The comparison is done between different ratio of length to width ratio of rectangular diaphragm and maximum deflection and stress is calculated using ANSYS. It is very important to calculate the relationship between deflections, stress on application of pressure.

## II. THEORY OF DEFLECTION OF RECTANGULAR DIAPHRAGM OF CAPACITIVE TRANSDUCER

According to the theory of rectangular plate bending, the deflection of rectangular plate may fall in small deflection or large deflection regime. A *small-deflection* case is defined as a displacement small compared with the plate thickness [22]. When the deflection is no longer small in comparison with the thickness of the plate but still small as compared with other dimensions, then this case is referred to as large deflection of plates. The structure of a capacitive transducer is as given in

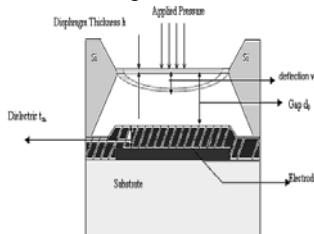


Fig.1. Capacitive pressure sensor

Fig. 1.The important parameter of the capacitive pressure sensor is its sensitivity and the capacitance-pressure

characteristics. These parameters are in turn influenced by the relation between deflection of the diaphragm and applied mechanical pressure. The deflection characteristics of the diaphragm are determined by the structural parameters such as area and thickness and the material properties such as Young's modulus (E), Poisson's ratio (v), residual stress etc. The diaphragm may exhibit plate characteristics or membrane characteristics depending on whether it is stress free deflection or stress dominated deflection. Therefore, to model, the proper choice of the partial differential equation is necessary.

### A.FOR LARGE DEFLECTION OF RECTANGULAR PLATE

The governing differential equation is [2]

$$\Delta w(x,y) = \frac{P}{D} + \frac{h}{D} \left[ \frac{\partial^2 F \partial^2 w}{\partial y^2 \partial x^2} + \frac{\partial^2 F \partial^2 w}{\partial x^2 \partial y^2} - 2 \frac{\partial^2 F}{\partial x \partial y} \frac{\partial^2 w}{\partial x \partial y} \right]$$

$$D \text{ is flexural rigidity} = \frac{Eh^3}{12(1-v^2)}$$

w is deflection of the plate mid-surface, P is the normal load per unit area,E is the Young's modulus of the diaphragm material and h is the thickness of the diaphragm and v is the Poisson's ratio

## III.SIMULATION

### A. Modeling Of Elastic Diaphragm

Analysis model is drawn through menu and progam files using Ansys [3] software version 9. The model input includes the diaphragm shape, size, location and material properties .After the diaphragm model displays on the screen, common pre-processor menus are used to define the simulation arameters.

### B. Diaphragm size

The diaphragm model size or the ratio of length to width was changed in the Notepad and save as another filename. Repeat the procedure 1, different size models can be simulated.

### C.Material Properties And Designing Parameters

Material properties [4] play an important role for the sensors and specify the respective operation how to operate.

TABLE I

Material used	Youngs modulus	Poissions ratio	Pressure
Silicon	130GPA	0.3	12psi

## IV. SIMULATION RESULTS AND ANALYSIS

The different ratio of length to width of rectangular diaphragm is taken such that the ratio of length is taken higher then width and on that basis results are compared .The effect of

stress ,strain and deflection is determined by applying the same pressure on different parameters of rectangular diaphragm. On the basis of simulation the maximum stress or strain is found at the middle of the longer side and minimum stress and strain is found at the center.

The maximum deformation value can be used to select the cavity depth. The bottom of cavity is designed to protect the diaphragm from fracture.Hence it is found that as the ratio of length to width is increased and the same pressure is applied the deformation is reduced or we can say deflection is reduced.

TABLE II  
DIAPHGRM SHAPE SIMULATION RESULT COMPARED

Length/width	1.75:1 $437.5 \times 10^6 : 250 \times 10^{-6}$	2.25:1 $562.5 \times 10^6 : 250 \times 10^{-6}$
SMX	$.365 \times 10^{-3}$	$.326 \times 10^{-3}$
DMX	$.365 \times 10^{-3}$	$.326 \times 10^{-3}$

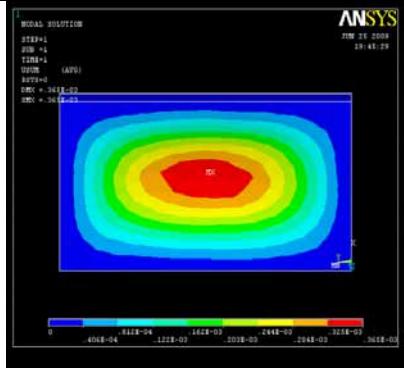


Fig.2. Stress and deflection distribution for 1.75:1

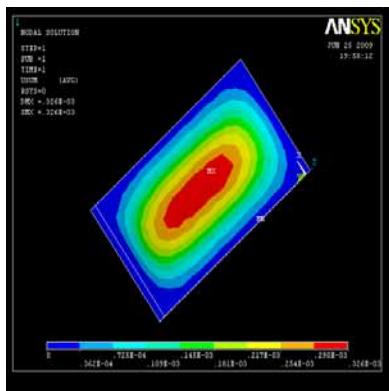


Fig.3. Stress and deflection distribution for 2.25:1

## VI. REFERENCES

- [1]. W. H. Ko, "Solid state capacitive transducers," Sensors and Actuator, vol.10, pp. 303-320, Feb.1986.
- [2]. S.Timoshenko, Theory of plates and shells,NewYork,Second edition,McGraw-Hill,1959
- [3]. <http://www.mece.ualberta.ca/Tutorials/ansy>
- [4]. R. Balasubramanyam, Material science and engineering, Seventh Edition, John Wiley and Son's, 2008

V. CONCLUSIONS  
Finite element analysis is used to model the large deflection rectangular diaphragms with clamped edges and different length to width ratio.Ansys version 9 is being successfully used to found the results. A complete simulation

# Investigations of the Phonemes in the Calls of Little Owls using MFCC'S

Randhir Singh<sup>1</sup>, Parveen Lehana<sup>2</sup> and Gurdadam Singh<sup>3</sup>

<sup>1</sup> ECE Dept., Sri Sai College of Engineering and Technology, Badhani (Pathankot), Punjab, India

<sup>2</sup> Department of Physics and Electronics, University of Jammu, Jammu, India

<sup>3</sup> ECE Dept., BCET, Gurdaspur, Punjab, India

er\_randhir81@rediffmail.com, lehana@iitb.ac.in, gurdadam@yahoo.com

**Abstract**—Birds have developed excellent capabilities of using vision and hearing senses. They can resolve the sound 10 times better than us. This allows more information to be communicated in lesser time. Bird's sounds can be divided into four categories: chip notes, calls, songs, and composite sounds. The vocal apparatus in birds consists of an oral cavity, larynx, trachea, syrinx, and lung system. The vocal tract above the larynx is very short and effectively terminates at lateral edges of an open beak. The bird larynx consists of muscular folds whose aperture can be regulated to prevent food particles from entering the respiratory system during ingestion. During vocalization the acoustic tube formed by vocal tract above and trachea below can be constricted at larynx by the action of laryngeal muscles. Trachea is terminated below by a special sound producing organ called as syrinx. It is clear that basic structure of sound production mechanism of birds is similar to that of human beings. As the human beings is composed of small units called phonemes, the bird's calls can also be assumed to be made of small units. The objective of this paper is to investigate the number of phonemes present in the little owl's calls by using MFCC's. Scope of this research is limited to only the sounds of little owl because of their special nature to communicate during noiseless environment. This may modify their adaptation with respect to phonemes.

**Index Terms**— Bird's sounds, bird's calls, owl's calls, call's types, chip notes, mel frequency cepstral coefficients

## I. INTRODUCTION

BIRDS are excellent in using vision and hearing capabilities for detecting prey. A barn owl can hunt mice by sound alone; woodpeckers can hear insects moving under the bark of a tree; pigeons can detect the infrasonic waves (under 16 vibrations per second) that come just prior to an earthquake. But it is the auditory stimuli of other birds that is often most important. Auditory signals help birds detect and locate danger, territory, food and shelter. Birds hear a greater range of sounds than humans. The birds are able to hear and produce sounds within a few days of birth. They have developed sense of time resolution, which is about 10 times better than ours. Several separate notes in sequence may sound to us like one long note. Because of their time resolution ability they hear the note separated into the smaller segments. This allows more information to be communicated.

Bird's sounds can be divided into four categories: chip notes, calls, songs, and composite sounds. Chip notes are short and high-pitched notes given by species such as warblers and sparrows. These are used to announce a food source or stay in contact with other birds of the same species. Calls are composed of either a single emphatic note or a series of notes

and may be divided into 10 main categories [1] as shown in Table 1.

The analysis of calls is very difficult because many birds use very similar calls in different contexts to convey different messages. Most birds seem to have between 5 and 15 distinct calls.

TABLE I  
BIRD'S CALLS

S.N.	Type	S.N.	Type
1	General alarm calls	6	Flight calls
2	Specialized alarm calls	7	Nest calls
3	Distress calls	8	Flock calls
4	Aggressive calls	9	Feeding calls
5	Territorial defense calls	10	Pleasure calls

Songs are the most complex sounds of birds. Singing is a mechanism used by birds to communicate their emotions to other living beings, mostly other companion birds. These are generally used to attract a mate. Not all birds sing. Singing is limited to Passeriformes and perching birds, whose population is nearly half of the total birds in the world. Song birds came into the knowledge around 60 million years ago. Birds of some species are born with the ability to sing their unique song. Eastern phoebes, for example, can sing their raspy two-noted fee-bee, even if they have never heard another phoebe sing.

Some times the bird's sounds are complex combinations of many basic sounds. It is not necessary for all sounds to be vocal. For example, Woodpeckers rap and several species have specialized feathers and behavior designed to send an audible message. These types of sounds may be grouped in fourth category that is composite sounds.

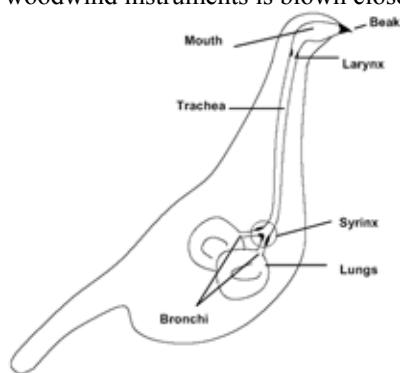
Three mechanisms are used by the birds to learn the production and recognition of the sounds. Some birds are born with the inherent capabilities. Most young birds learn sounds from their parents. Some of birds learn these skills from other adult males of their kind. By the following spring, all the males need to perform the song well if they hope to attract a mate. The chipping sparrow learns its simple series of musical chips from another male chipping sparrow nearby. Some male birds, such as the song sparrow, learn a repertoire of two or three songs, which they sing over and over. Birds of the same species may have different accents depending on their geographical location. The vocal apparatus in birds consists of

an oral cavity, larynx, trachea, syrinx, and lung system. The vocal tract above the larynx is very short and effectively terminates at lateral edges of an open beak. The bird larynx consists of muscular folds whose aperture can be regulated to prevent food particles from entering the respiratory system during ingestion. During vocalization the acoustic tube formed by vocal tract above and trachea below can be constricted at larynx by the action of laryngeal muscles. Trachea is terminated below by a special sound producing organ called as syrinx. It is clear that basic structure of sound production mechanism of birds is similar to that of human beings.

Klatt [2] has shown that mynah can imitate human speech very effectively using its tongue. Other birds are not so efficient, but they can produce limited number of phonemes by modulating the behavior of the syrinx. The objective of this paper is to investigate the number of phonemes present in the crow's calls. It is very difficult to collect and analyze the sounds of all birds; hence, scope of this research is limited to only the sounds of crows because of their wide spread population obtained from different sources. The detailed explanation of sound producing mechanism in birds is explained in the following section. The methodology and results & conclusions are presented in the subsequent sections.

## II. MECHANISM OF SOUND PRODUCTION

Birds do not have a larynx like human beings. Instead they have an organ called a syrinx (Fig 1) [3]. Syrinx may be located at the junction of the two primary bronchi and the trachea or entirely in the trachea or in the bronchi. Syrinx resembles human vocal cords in function, but it is very different in form. Also the vocal tract, whose main parts are trachea, larynx, mouth and beak, interacts to the sound of birds [4]. When a bird is singing, airflow from lungs makes syringeal medial tympaniform membrane (MTM) in each bronchi to vibrate through the Bernoulli effect [5]. The membrane vibrates nonlinearly opposite to the cartilage wall. Voice and motion of the membrane is controlled by a symmetrical pair of muscles surrounding the syrinx. Membranes can vibrate independently to each other with different fundamental frequencies and modes. Membranes are pressure controlled like a reed in woodwind instruments, but membranes are blown open while the reed in the woodwind instruments is blown closed.



**Fig. 1.** Sound producing mechanism in birds.

In contrast to the MTM theory recent studies with endoscopic imaging have shown that MTM would not be the

main source of sound [6]. Goller suggests that sound is produced by two soft tissues, medial and lateral labia (ML and LL in Fig 2), similar to human vocal folds. Sound is produced by airflow passing through two vibrating tissues. Further evidence to this comes from a study where MTM's were surgically removed [7]. After removal birds were able to phonate and sing almost normally. Small changes in song structure however were found, which indicates that MTM's have a function in sound production. However it is possible that birds may be able to compensate the loss of MTM. Also, because of large diversity in structure of avian syrinx and also in sounds, it is possible that the MTM theory is correct for some species. For example Goller and Larsen limited their study only to cardinals (*Cardinalis cardinalis*) and zebra finches (*Taeniopygia guttata*). In contrast in [8] ring doves (*Streptopelia risoria*) were studied as evidence for the MTM theory. Furthermore in [9] it was found that the main source of sound in pigeons and doves is the tympaniform membrane. However this membrane is located in the trachea and not in the bronchi.

The anatomy of the syrinx and the avian vocal tract vary considerably among different orders of birds and sometimes even in different families within the same order. Syrinx may act as a double instrument for generating two different notes at the same time, or even sing a duet with itself. For example thrushes use this mechanism; they are even capable of singing a rising note with one side and a falling note with the other. It is this sort of ability that allows some birds to sing as many as 30 separate notes per second.

Structure of bird song has large diversity. Typical song may contain components which are pure sinusoidal, harmonic, nonharmonic, broadband and noisy in structure [10]. Sound is often modulated in amplitude or frequency or even both together (coupled modulation) [11]. Frequency range is relatively small, typically fundamental frequency lies between 3 and 5 kHz. A well-established way to divide song into four hierarchical levels is: elements or notes, syllables, phrases, and song [12]. Elements can be regarded as elementary building units in bird song [13] whereas phrases and songs often contain individual and regional variation. Duration of one syllable ranges from few to few hundred milliseconds.

## III. METHODOLOGY

### *Characteristics of little owls*

The Little Owl is a bird which is resident in much of the temperate and warmer parts of Europe, Asia east to Korea, and North Africa. Latin name of little owl is *Athene noctua*. This small owl was introduced to the UK in the 19th century. Little owls have a wingspan of 54-58cm and are 21-23cm long. The males weigh an average of 170g and the female's average 174g. Their back and wings is a deep grey-brown, spotted with white. The underside is white with broad, broken streaks of grey-brown. The face is marked by dark areas around the yellow eyes that give a frowning look. Extremely acute eyesight, sees well night or day. Eyes fixed in sockets, can only look straight ahead, turns head to see surroundings, head can swivel 270°. It will bob its head up and down when alarmed.

They feed on a wide variety of prey - mostly small mammals, such as mice, voles, shrews, even small rabbits, as well as insects, earthworms, snails, slugs and small fish. They can be found all year round, during the day. It hunts at night and dawn. The environment at this time is generally quite and this may modify the adaptation of owls to communicate using the whole available spectrum. This is a sedentary species which is found in open country such as mixed farmland and parkland. Little owls also occupy woodland, fields, coastal areas and semi-desert areas. They nest in tree holes, pollarded willows, walls of old buildings, rabbit burrows and cliff holes. The female lays 3-5 eggs in early May and incubates the eggs for 29 days. Only the male feeds the chicks at first, but later the female helps. After 26 days, the chicks leave the nest. If living in an area with a large amount of human activity, Little Owls may grow used to man and will remain on their perch, often in full view, while humans are around. It can be seen in the daylight, usually perching on a tree branch, telegraph pole or rock. Little owls are not considered to be at threat, and there is a population of 9,000 pairs in the UK [14] - [17].

#### Recording material

The main call of a Little Owl is a ringing *kiew, kiew*, repeated every few seconds. The second is a rapidly repeated, yelping *wherrow*. Little owls use a variety of chattering notes at the nest and in particular during the breeding season a loud *hoo-oo* note. We have taken five calls of little owls from [18], [19] of *Athene noctua* or Civetta. These calls were recorded at Poderone near Magliano in Toscana (Latitude: 42°36' N, Longitude: 11°17' E). The detail of these calls is given below.

CALL 1: One male sings (not seen) perched somewhere in a cultivated field with scattered olive trees (Date: 01 Feb'04, Time: 6.00 p.m., Altitude: +130 m).

CALL 2: One bird (not seen) is perched somewhere in a cultivated field with scattered olive trees. It seems to answer to the male hoots of call 1. Might be a female answering to her mate (Date: 01 Feb'04, Time: 6.00 p.m., Altitude: +130 m).

CALL 3: One bird (not seen) is perched on a pine tree over a camping area, which is full of tourists; zone with cultivated fields and farms. A slightly different "kew" call, maybe less excited compare with Call 2 (Date: 23 Aug'04, Time: 10.30 p.m., Altitude: +17 m).

CALL 4: One bird (not seen) is perched on a pine tree near the border of Patanella pinewood. A third different example of "kew" call. Maybe the bird is slightly anxious, because sound recoder quite close to its perch (Date: 24 July'05, Time: 0.30 a.m., Altitude: 0 m).

CALL 5: One bird (not seen) is perched on an oak tree in a zone with cultivated fields, vineyards and woods. A plaintive cry, not unlike Call 3, but less bisyllabic (Date: 4 July'07, Time: 11.30 p.m., Altitude: +130 m).

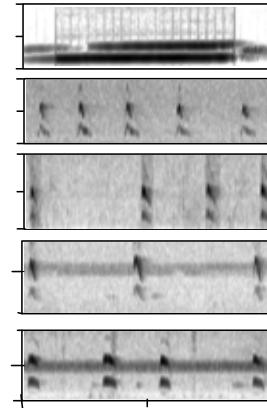
#### Signal Processing

The five calls were analyzed individually and collectively for obtaining 21 order mel frequency cepstral coefficients (MFCC's) using a hamming window of 10 ms with 1 ms shifting. The calls were analyzed at sampling frequencies of 8 kHz and 44.1 kHz. The MFCC's of each individual call were used to get the centroids of the calls. The centroids were

obtained by using vector quantization. Total of 2, 4, 8, 16, 32, and 64 centroids were used for estimating the number of phonemes. For calculating the MFCC's the windowed signal was used to obtain 1024 point FFT. The process for this is shown in Fig 3. After estimating the energy  $E_m$  in each critical band in the spectrum by using a triangular function [20], MFCC's were calculated as

$$c_m = A \sum_{m=0}^{M-1} \cos(j \frac{\pi}{M}(m+0.5)) \log_{10} E_m$$

The factor A was taken as 100 [21]. The order of MFCC's,  $M$ , is taken as 21. The Euclidean distances among the centroids were found and plotted as 3-D plots using multiple colours. Each colour represented a specific range of distances. The 3-D plots were analyzed at different viewing angles to estimate the total major shades of the colours. The number of shades indicated the number of phonemes in the crow's calls.

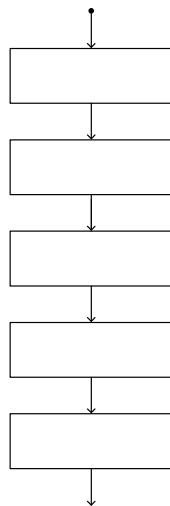


**Fig. 2.** Spectrograms of the five calls described in Section II-B.

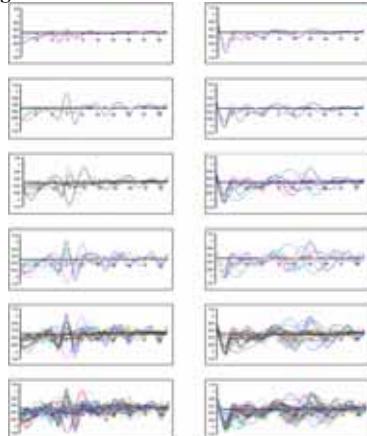
#### IV. RESULTS AND CONCLUSIONS

The spectrograms of some segments of the calls used for these investigations are shown in Fig 2. The analysis of the spectrograms show that in each call, there are some finite variations of duration, silence, formants, randomness of spectral structure, and amplitudes. It is difficult to say whether each call is composed of a single phoneme or multiple phonemes. From the analysis of the centroids of individual calls, it was observed that there were considerable differences in the single centroids and multiple centroids. It tells us that each call may be composed of more than one phoneme. As the number of centroids are increased from 8 to 32, the plots become very confusing after 16 centroids. Many centroids strats coming close to each other. For deriving a solid conclusion, the centroids for all the five calls were analyzed.

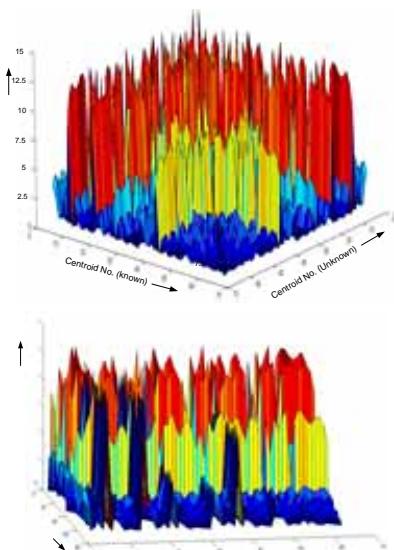
The plots in Fig 4 show the centroids estimated for the five calls of little owls, described in Section II. The first column is for sampling frequency of 8 kHz and second column is for 44.1 kHz. It can be observed that there is a definite effect of sampling frequency on the centroids. Hence, for further analysis, only sampling of 44.1 kHz was taken. The higher one was chosen because of the relatively higher time/frequency resolution of birds for understanding the calls.



**Fig. 3.** Estimation of MFCC's from bird's calls.



**Fig. 4.** Comparison of MFCC's of all the five calls of little owls collectively analyzed. The first column is for sampling frequency of 8 kHz and the second column is for 44.1 kHz. The numbers of centroids for these plots are 2, 4, 8, 16, 32, and 64.



**Fig. 5.** Three dimension plot of the distances between different centroids of all the five calls of little owls. Each plot is taken for different viewing angle for

observing the total number of phonemes (basic sound elements). The total number of centroids are taken as 64. The sampling frequency is fixed at 44.1 kHz. (View 1 and View 2).

Three dimension plots of the distances between 64 centroids are shown in Fig 5. Each plot is taken for different viewing angle for observing the total number of phonemes. It can be concluded from these plots that the total numbers of colors in all the plots viewing at different angles fall in the range of 9 to 10. It means, the total number of phonemes for little owls may be around 9. It should be noted that total number of messages communicated by the little owls may be more as the silence duration also plays a major role in deciding the meaning of the calls. The number and types of phonemes may be slightly different for little owls belonging to different habitats.

## V. REFERENCES

- [1]. [http://www.dnr.state.mn.us/young\\_naturalists/birdsong/](http://www.dnr.state.mn.us/young_naturalists/birdsong/)
- [2]. Dennis H. Klatt and Raymond A. Stefanski, "How does a mynah bird intimate human speech," J. Acoust. Soc. Am., Vol. 55, No. 4, 1974, 822-832.
- [3]. J. McLelland. Larynx and trachea, chapter 2, pp. 69–103, 1989.
- [4]. S. Nowicki, 'Vocal tract resonances in oscine bird sound production: Evidence from birdsongs in a helium atmosphere', Nature, 325 (6099), 53–55, 1987.
- [5]. N. H. Fletcher, Acoustics Systems in Biology, Oxford U.P., New York, 1992.
- [6]. F. Goller and O. N. Larsen, "A new mechanism of sound generation in songbirds," in Proc. National Academy of Sciences, Vol. 94, pp. 14787–14791, 1997.
- [7]. F. Goller and O. N. Larsen, "New perspectives on mechanism of sound generation in songbirds", J. comp. Physiol. A 188, 841–850, 2002.
- [8]. S. Gaunt, S. L. L. Gaunt, and R. M. Casey, "Syringeal mechanics reassessed: Evidence from streptopelia", Auk 99, 474–494, 1982.
- [9]. F. Goller and O. N. Larsen, "In situ biomechanism of the syrinx and sound generation in pigeons", J. exp. Biol 200, 2165–2176, 1997.
- [10]. S. Nowicki. Bird acoustics, in M. J. Crocker, ed., 'Encyclopedia of Acoustics', John Wiley & Sons, chapter 150, pp. 1813–1817, 1997.
- [11]. J. H. Brackenbury. Functions of the syrinx and the control of sound production, chapter 4, pp. 193–220, 1989.
- [12]. K. Catchpole and P. J. B. Slater. Bird Song: Biological Themes and Variations, Cambridge University Press, Cambridge, UK, 1995.
- [13]. S. E. Anderson, A. S. Dave, and D. Margoliash, 'Template-based automatic recognition of birdsong syllables from continuous recordings', J. Acoust. Soc. Am. 100(2), 1209–1219, 1996.
- [14]. [http://en.wikipedia.org/wiki/Little\\_Owl](http://en.wikipedia.org/wiki/Little_Owl)
- [15]. <http://www.rspb.org.uk/wildlife/birdguide/>
- [16]. <http://www.birding.in/birds/Strigiformes/Strigidae/>
- [17]. <http://www.thelittleowlnyc.com/>
- [18]. <http://www.birdsongs.it/songs/songspectro.asp>
- [19]. [http://www.birdsongs.it/songs/athene\\_noctua/](http://www.birdsongs.it/songs/athene_noctua/)
- [20]. J. Picone, "Signal modeling techniques in speech recognition", in Proc. of the IEEE, 1993, 81(9): 1215-1247.
- [21]. O. Cappe, J. Laroche, and E. Moulines, "Regularized estimation of cepstrum envelope from discrete frequency points", in Proc. EuroSpeech, 1995.

Bird's Calls

Windowing

IFFT<sup>2</sup>

Logarithm

Mel Filter Bank

DCT

MFCC

# Image Compression Using the Discrete Cosine Transform

Vikrant Vij\*, Rajesh Mehra\*\*, Sumeet Prashar\*\*\*

\* Student, ME ECE, NITTTR Chandigarh, INDIA, \*\* Sr. Lecturer, ECE Department, NITTTR, Chandigarh, INDIA  
 \*\*\* Student, ME ECE, NITTTR Chandigarh, INDIA

**Abstract-** Uncompressed multimedia (graphics, audio and video) data requires considerable storage capacity and transmission bandwidth. Despite rapid progress in mass-storage density, processor speeds, and digital communication system performance, demand for data storage capacity and data-transmission bandwidth continues to outstrip the capabilities of available technologies. The recent growth of data intensive multimedia-based web applications have not only sustained the need for more efficient ways to encode signals and images but have made compression of such signals central to storage and communication technology. For still image compression, the 'Joint Photographic Experts Group' or JPEG standard has been established by ISO (International Standards Organization) and IEC (International Electro-Technical Commission). All these are based on block based discrete cosine transform.

## I. INTRODUCTION

A discrete cosine transform (DCT) expresses a sequence of finitely many data points in terms of a sum of cosine functions oscillating at different frequencies. The DCT has the property that, for a typical image, most of the visually significant information about the image is concentrated in just a few coefficients of the DCT. For this reason, the DCT is often used in image compression applications. The rapid growth of digital imaging applications, including desktop publishing, multimedia, teleconferencing, and high-definition television (HDTV) has increased the need for effective and standardized image compression techniques. Among the emerging standards are JPEG, for compression of still images [Wallace 1991]; MPEG, for compression of motion video [Puri 1992]; and CCITT H.261 (also known as Px64), for compression of video telephony and teleconferencing.

All three of these standards employ a basic technique known as the discrete cosine transform (DCT). A DCT is a Fourier-related transform similar to the discrete Fourier transform (DFT), but using only real numbers. The DCT works by separating image in to parts of different frequencies. During process called Quantization, the less important frequencies are discarded. Only the most important frequency parts are used to retrieve the image during de-compression process. Its performance is compared with that of a class of orthogonal transforms and is found to compare closely to that of the Karhunen-Loeve transform, which is known to be optimal. The performances of the Karhunen-Loeve and discrete cosine transforms are also found to compare closely with respect to the rate-distortion criterion.

## II. DEFINITION

The discrete cosine transform is a linear, invertible function  $F: \mathbf{R}^N \rightarrow \mathbf{R}^N$  (where  $\mathbf{R}$  denotes the set of real

numbers), or equivalently an invertible  $N \times N$  square matrix DCT-I

$$X_k = \frac{1}{2}(x_0 + (-1)^k x_{N-1}) + \sum_{n=1}^{N-2} x_n \cos\left[\frac{\pi}{N-1}nk\right] \quad k=0, \dots, N-1.$$

The DCT-I is exactly equivalent (up to an overall scale factor of 2), to a DFT of  $2N - 2$  real numbers with even symmetry. Thus, the first transform coefficient is the average value of the sample sequence 300. The DCT-I is not defined for  $N$  less than 2. (All other DCT types are defined for any positive  $N$ .) Thus, the DCT-I corresponds to the boundary conditions:  $x_n$  is even around  $n=0$  and even around  $n=N-1$ ; similarly for  $X_k$ . The one-dimensional DCT is useful in processing one-dimensional signals such as speech waveforms. For analysis of two-dimensional (2D) signals such as images, we need a 2D version of the DCT.

## DCT-II

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n + \frac{1}{2}\right)k\right] \quad k=0, \dots, N-1.$$

The DCT-II is probably the most commonly used form, and is often simply referred to as "the DCT". This transform is exactly equivalent (up to an overall scale factor of 2) to a DFT of  $4N$  real inputs of even symmetry where the even-indexed elements are zero. That is, it is half of the DFT of the  $4N$  inputs  $y_n$ , where  $y_{2n} = 0$ ,  $y_{2n+1} = x_n$  for  $0 \leq n < N$ , and  $y_{4N-n} = y_n$  for  $0 < n < 2N$ . The 2D DCT can be computed by applying 1D transforms separately to the rows and columns, we say that the 2D DCT is *separable* in the two dimensions.

## III. THE COMPRESSION PROCESS

Efficacy of a transformation scheme can be directly gauged by its ability to pack input data into as few coefficients as possible. This allows the quantizer to discard coefficients with relatively small amplitudes without introducing visual distortion in the reconstructed image. DCT exhibits excellent energy compaction for highly correlated images.

General overview of JPEG Process:

1. Firstly image is broken in blocks of Pixels.
2. DCT is applied to each block.
3. Each block is compressed through quantization.
4. The array of blocks that constitute space is stored in reduced amount of space.
5. When desired, Image is reconstructed through decompression using Inverse Discrete cosine Transform.

## IV. DCT QUANTIZATION

During quantization each pixel of the transformed digital image is mapped to a discrete number. Each integer in the range of numbers used in the mapping symbolizes a color. After the image is DCT transformed, it is divided into 8x8

blocks. These blocks are then encoded individually. The blocks on the top left corner would be encoded with more bits to keep the important information or energies. As we move away from the upper left hand corner, the blocks are encoded with fewer and fewer bits. Eventually hitting the bottom right corner, the blocks are encoded with few if any bits l. This is the usual DCT quantization method.

## V. CODING

In this step , Quantized matrix is converted by an encoder in to stream of binary date ( 01 , ,1,0...). After Quantization, it is common that some of the coefficients reduce to zero. JPEG takes advantage of this by sequencing quantized coefficients in Zigzag Pattern as shown in figure 1. The advantage lies in consolidation of large number of zeroes that compresses very well. The sequence shown in figure continues for entire block.

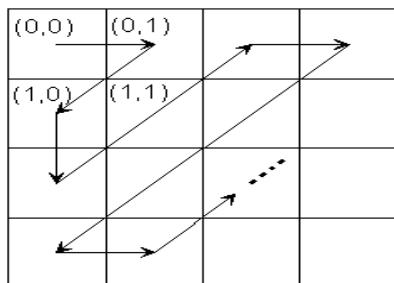


Figure 1.

## VI. DE-COMPRESSION

De-compression is accomplished by decoding the bit stream representing quantized matrix.. Each element in this matrix is multiplied with corresponding element of Original Quantization matrix. IDCT is applied to the resultant matrix which is rounded to nearest integer. The JPEG encoding does not fix the precision needed for the output compressed image. On the contrary, the JPEG standard (as well as the derived MPEG standards) have very strict precision requirements for the decoding, including all parts of the decoding process (variable length decoding, inverse DCT, dequantization, renormalization of outputs); the output from the reference algorithm must not exceed:

- a maximum 1 bit of difference for each pixel component
- low mean square error over each 8×8-pixel block
- very low mean error over each 8×8-pixel block
- very low mean square error over the whole image

## VII. COMPARISON

In this section we are comparing images obtained at various steps in DCT compression process. Figure 2 is the Actual image.



Figure 2: Original Image

Than on applying DCT, the resulting image is shown in Figure 3.

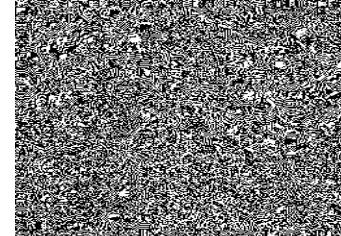


Figure 3 : DCT applied to Original

Each element in each block of image is quantized with quantization matrix of quality level 50. At this point many of elements become zeroed out and the image takes much less space to store.

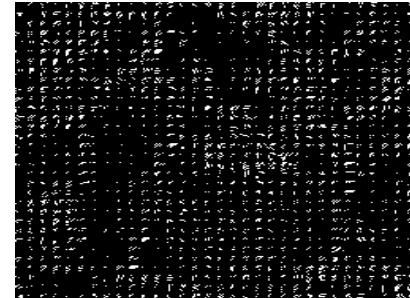


Figure 4 : Quantized DCT

Image can now be decompressed using inverse discrete cosine transform. At such a quality level there is no visible loss in Quality of Image. At lowest Quality level, the quality goes down but the compression does not increase very much.



Figure 5: Quality 50-84 % zeroes.

## VIII. CONCLUSION

DCT exploits inter pixel redundancies to render excellent de correlation for most natural images, The DCT packs energy in the low frequency regions. Therefore, some of the high frequency content can be discarded without significant quality degradation. Such a (course) quantization scheme causes further reduction in the entropy (or average number of bits per pixel). Lastly, it is concluded that successive frames in a video transmission exhibit high temporal correlation (mutual information). This correlation can be employed to improve coding efficiency

## IX. REFERENCES

- [1]. Ahumada, A. J., Jr. and A. B. Watson. A visual detection model for DCT coefficient quantization. Technology 2003. 2: 404-415, 1993SC..
- [2]. G. Strang , “The Discrete Cosine Transform SIAM review , Volume 41, Number 1, pp 135 ,
- [3]. G. Aggarwal and D. D.Gajski, “Exploring DCT Implementations,” UC Irvine, Technical Report ICS-TR-98-10, March 1998

# Real-Time Control of an Electronic Knee Joint

Neelesh Kumar, Dinesh Pankaj, Davinder Pal Singh, Sanjeev Soni, Amod Kumar, B S Sohi\*  
 Central Scientific Instruments Organization (CSIO) Chandigarh, \*Director UIET, Chandigarh

**Abstract-** The process of human walking seems to very simple and natural. In reality this repeated course of action involves several concealed processes. To mimic near natural gait with a prostheses require a control strategy in real-time. In this paper a robust real-time control strategy for an electric knee joint is proposed. The intelligent prosthesis provides stance control ranging from minimal resistance to a lock like situation. The control employed uses the Analog Devices ADuC842 analogue microcontroller and other sensors. This control strategy is tested successfully on an indigenously developed electronic knee joint.

**Keywords:** real-time control, gait, above knee, prosthetic, amputee.

## I INTRODUCTION

In a normal walking cycle, repetitive events occur. The repetitive pattern can be divided into two distinct events: 1) foot strike and 2) toe-off. When in a walking cycle, both legs contribute to four different events: 1) foot strike, 2) opposite toe-off, 3) opposite foot-strike, and 4) toe-off [1]. Since the events occur in a similar sequence and are independent of time, the gait cycle can be described in terms of percentage, rather than time, thus allowing normalization of the data for multiple subjects. The initial foot strike occurs at 0%, and occurs again at 100% (0-100%). Each stride represents one gait cycle and is divided into two periods (main phases): *stance* and *swing*. Stance is the period when the foot is in contact with the support surface and constitutes 62% of the gate cycle. The remaining 38% of the gait cycle constitutes the swing period that is initiated as the toe leaves the ground. As most of the part is in stance phase better control can be achieved if resistance offered in stance phase can be adjusted according to the walk phase. Resistive knees consist of hydraulic or pneumatic cylinders to provide variable resistance. Therefore, amputee would be able to have different walking speed. The essential part of this prosthesis is the knee joint actuator based on control of basic flexion and extension functions of the knee joint with the motor [2]. In a microcontroller based prosthetic knee joint, the controller changes the knee impedance (damping and/or stiffness) based on sensory information. This resistive torque for the knee joint can be provided by hydraulic or pneumatic cylinder. The hydraulic/pneumatic damper with variable impedance comprises a double acting cylinder where two sides of the piston are connected through a valve [3]. The commands determine the position of a valve that controls the flow of oil from one chamber to the other. One stepper motor is used to adjust the position of a needle valve (orifice) which controls the flow rate between two sides of the piston [4]. The stepper motor is controlled by a microcontroller based on the sensory information according to the swing speed of the prosthetic leg. Figure -1 shows the developed electronic knee joint.



Figure-1: artificial knee developed

The purpose of our research is to automatically adjust the position of the valve to attain maximum possible control in stance phase.

## II METHODOLOGY

In above knee prosthesis generally mechanism shown in figure-2 is implemented. A hydraulic or pneumatic cylinder at knee joint is used in this arrangement to adjust the stance and swing phase resistance. Sensors for sensing angle, heel strike and toe-off are included in the prosthetic unit. A microcontroller processes information and control the knee resistance. Usually a needle valve is used to adjust the resistance in knee joint. Opening of the valve decides the required resistance. This control valve is generally driven by a precise stepper motor. Valve adjusts the flow of fluid or air in cylinder which in turn builds the required resistance.

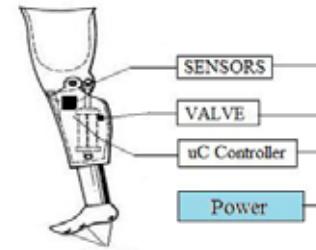


Figure-2: Schematic diagram of the prosthetic knee.

In complete system the critical part is the algorithm which processes information from sensors and then operates the valve. Although hardware is also an essential part, but we restrict our discussion on control strategy only.

Control of electronic knee joint consists of different modes. Each mode represents a different phase of walk. There are various modes of operation in our control strategy these modes are as:

- A. Calibration mode
- B. Training Mode

- C. Run Mode
- D. Backup Mode(*Safe mode*)
- E. Fail Safe Mode

- a) *Calibration Mode*: This mode is for initial sensor calibrations and other electro-mechanical synchronizations. e.g if an angle sensor is replaced by a new one its initial zero setting is done in this mode. Similar settings for other sensors are available in this mode. This mode is of much use for the prostheses engineer or service engineer than to the user.
- b) *Training Mode*: Before the electronic knee actually works or starts assisting the amputee it requires some training. For training the knee it should be bring into training mode. In training mode the knee is fitted to the patient and training mode is initiated. The person in then asked to walk 10-15 steps in slow motion. The sensors on knee monitors the range of angle and other sensor inputs for this walking pattern and stores the result in memory and mark it as slow mode. Similarly training for normal and fast mode is also performed. The outcome of these training sessions is stores as normal and fast mode respectively. This process of training is one time process and it is required only when a prosthetic is fitted to a new amputee. This training mode makes the knee adaptive to the user and thus reduces mental stress on amputee's mind.

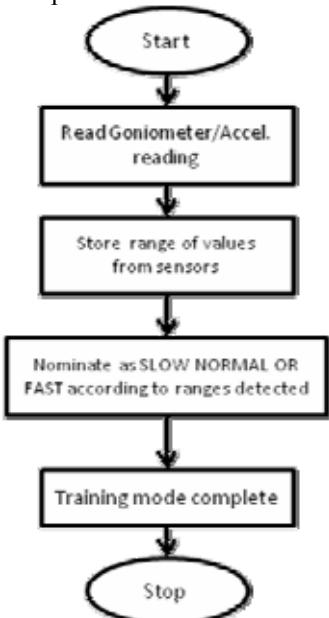


Figure- 3: Training Mode flow chart

- c) *Run Mode*: An artificial knee when powered ON and the amputee starts walking the knee operate in RUN mode. Flow chart in figure-4 shows the flowchart for RUN mode. This mode actually consists of three modes named as:

- a. Slow Mode
- b. Normal Mode
- c. Fast Mode

In slow mode of operation the knee detects a particular range of sensor inputs and adjusts the position of the valve to offer damping required in slow walking to the amputee.



Figure-4: Run mode flow chart

In case the microcontroller detects that the input from sensors provide a different range that particular mode is initialized (Normal or Fast).

*Backup Mode*: No doubt the electronics and mechanics of the prosthetics are designed and developed for operation in rough conditions (dust, heat & moisture) but there should be some way out for worst cases. Backup mode of the knee is to handle such conditions. Figure-5 shows flowchart for backup mode. When any sensory input goes down the knee unit shifts its control to backup mode. In backup mode the prosthetic knee operates as in case for slow mode operation but the difference is that no sensing is done whether the person is walking slow normal or fast. This mode reduces the chance for falling due to failure of sensors in pure mechanical hardware.

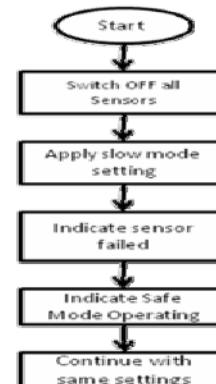


Figure-5: backup mode flow chart

- e) *Failsafe Mode*: a failsafe mode is also incorporated in control if complete electronics fails the mechanical hardware operates and fixed resistance is offered to the amputee.

### **III CONCLUSION**

The present paper describes approach in which developed control strategy was implemented and knee control by microcontroller were analysed.

It is observed that the control strategy works successfully in simulated conditions. If system is out of order because the any sensor failures, the device acts as a prosthesis equipped with a mechanical knee with fixed resistance. Therefore, the subject is not immobilized. He only loses the advantage of sensors and other electronics in the assistive device. Future research will improve the size, the weight, adaptability and the intelligence of the system.

### **IV REFERENCES**

- [1]. Michael A Whittle, Gait Analysis an introduction. Heidi Harrison Elsevier 2007.
- [2]. Winter A., Biomechanics and motor control of human movement, WILEY 1990.
- [3]. Akin O. Kapti, M. Sait Yucenur, Design and control of an active artificial knee joint, Mechanism and Machine Theory 41 (2006) 1477–1485.
- [4]. Neelesh Kumar, Davinder Pal Singh, dinesh Pankaj, “Algorithm For Control Of Prosthetic Knee Joint”IACC-09, Thaper Patiala.

# Real Time Audio Transmission over Internet

\*Sr. Lecturer- Er. Supriya Kinger \*\*Lecturer-Er. Jyoti Snehi  
Chitkara Institute Of engineering & Technology, Rajpura, Punjab, India

**Abstract-** This research identifies the problems encountered in transmitting voice over the Internet and proposes approaches to solve these problems. The current Internet is not very suitable for transmitting real-time data because its underlying protocols and switches are only engineered to transmit non-real time data. The problems posed by voice over the Internet can be studied by conducting a series of experiments. The problems caused by high loss, large delay, and jitter seriously affect the transmission quality. A good design thus needs to combine different components that solve these problems together. Silence removal and compression is used to reduce bandwidth usage.

Jitter buffers are used to smooth the burstiness in the received stream caused by the network. To conceal loss, we investigate existing methods and propose two new nonredundant reconstruction algorithms based on a simple but effective average reconstruction scheme. One is to apply adaptive filtering on top of average reconstruction in order to explore the signal trend and to obtain better estimations. Effects of important filter parameters, such as filter length and adaptation step size, are studied. Another new method, called transformation-based reconstruction method, is developed to handle low reconstruction quality caused by some rapidly changing parts of signals. Its basic idea is to let the sender transform the input voice stream, based on the reconstruction method used at the receiver, before sending the data packets. Consequently, the receiver is able to recover much better from losses of packets than it would without any knowledge of what the signals should be.

This method can improve reconstruction quality without significant computational cost and can be easily extended to different interleaving factors and different interpolation-based reconstruction algorithms.

## I. INTRODUCTION

Voice over the Internet is the process of transmitting voice information which is traditionally transmitted via public switched telephone network (PSTN) over the Internet, a packet switched network. The basic steps involve converting analog voice signals to digital format, the compression/ translation of the signals into Internet Protocol (IP) packets for transmission over the Internet, and the reverse process at the receiving end. The integration of voice and data transmissions over the Internet offers an opportunity for significant communication cost savings if reliable, high-quality voice service similar to PSTN [1] can be achieved. Although the benefits of real-time voice transmissions over the Internet are obvious, and there have already been many commercial products for Internet telephony in the market, only a small fraction of users who have tried these applications are

willing to adopt and actively use the technology [2]. The most important reason is that the level of speech quality (such as end-to-end delay, jitter, clarity, and continuity), achieved is not comparable to the quality of PSTN [3]. These considerations motivate us to investigate the problem for transmitting real-time voice over the Internet.

## II. PROBLEM STATEMENT

The quality of most current Internet real-time voice transmission systems is not satisfactory because of the current Internet's delivery and scheduling mechanisms. The Internet has been traditionally designed to support text-based non-real-time data communications, but not real-time voice transmissions, such as Internet phone. These real-time applications have quite different characteristics as outlined here.

The first significant characteristic of real-time applications is their high delay sensitivity. Given strict end-to-end delay and interframe delay requirements for real-time transmissions, packets delayed over a certain time limit are considered lost [4] and cannot be retransmitted by the sender [5].

The current Internet does not support real-time transmissions because it has no special delivery mechanism to differentiate between real-time data and non-real-time data. Hence, all real-time data frames are treated the same way as non-real-time frames and will be dropped or delayed with equal chance under heavy load and congestion. The current Internet also may have large delay variations and loss. Measurements carried out by others [6] and us have shown that the loss rate of packets to some destinations can be as high as 50%.

The second significant characteristic is that most real-time applications do not require data to be 100% precise, unlike services provided by Transmission Control Protocol (TCP), which ensure that all data packets are sent correctly and reliably all the time. This characteristic is very useful because the receiver can tolerate a certain level of loss or distortion of data without significant degradation in performance [7].

The above two characteristics define the potential problems that should be considered in order to develop a high-quality real-time voice transmission system. Reliability and predictability are the two major problems [8]. Reliability ensures the reliable delivery of voice packets so that loss is concealed from users, whereas predictability avoids the delivery of voice packets with excessive delay and to maintain a certain playback rate. These requirements can be measured by quality-of-service (QoS) measures. The key QoS measurements include end-to-end delay, jitter, and loss. End-to-end delays and jitters are key measurements of predictability, whereas loss is the key measurement of reliability. Obviously, reliability is a more urgent problem to be handled because without reliability, predictability is hard to achieve. The purpose of this paper is to examine the necessary components of a real-time voice transmission system by studying Internet voice-traffic behaviors and by designing new reconstruction methods to conceal loss and improve transmission quality.

### III. RELATED WORK

In the current literature, there is no good way to reduce end-to-end delay except by keeping the processing and buffering delay at both ends low, because network delay is not controllable [9]. The use of jitter buffers [10] is universally accepted for controlling jitters, although they will introduce extra delay.

For concealing loss, algorithms can be classified into two major categories: receiver based, and sender and receiver based. The first class of nonredundant reconstruction methods for concealing loss involves the receiver only. In these methods, only one copy of each voice packet is sent, and the receiver is responsible to recover the lost packets. A common strategy to compensate the loss of a packet is to replay the last packet received during the interval when the lost packet is supposed to be played back. If the length of a bursty loss is short, then this scheme can give reasonable playback quality. Another strategy is to replace the lost packets using a segment of silence or a segment of white noise. These two simple strategies can fill the gap between noncontiguous speech frames received at the receiver and work well when the occurrence of lost frames is infrequent and the frame size is small. However, they do not work well when the length of a bursty loss is large [11]. A third strategy reconstructs the missing data based on data received already.

The second class of reconstruction methods for concealing loss involves both the sender and the receiver. The sender first processes the input data streams in such a way that the receiver can reconstruct the missing data better. Based on the different ways of processing the input data, these schemes can further be split into two subclasses: one that adds redundant control information and one that does not.

There are several methods for the sender to add redundancy to data streams. The first method [12] sends redundant information at the expense of increased bandwidth. Its general idea is to replicate and send the  $i^{\text{th}}$  packet along with the  $(i+1)^{\text{st}}$  packet so that when a packet is lost, the receiver still has another packet as a backup. Obviously, the network bandwidth is doubled, leading to higher delays and congestions. Another method used is based on forward error correction (FEC) [13, 14] that protects every  $n$  packets by inserting a redundant packet containing an error correcting code. This approach increases the bandwidth by  $1/n$  and may increase the latency by  $n$  times, since in the worst case,  $n$  packets have to be received before the lost packet can be reconstructed. A third method is adopted in the MICE project [11] that utilizes a kind of sequence loss protection. The basic idea is to add extra data in the  $i^{\text{th}}$  packet that contains redundant information for the  $(i-1)^{\text{st}}$  or  $(i-2)^{\text{nd}}$  packet. Although this process reduces the latency as compared to the FEC method, it still requires more network bandwidth.

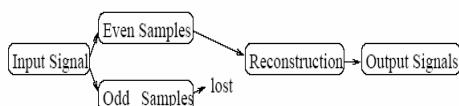
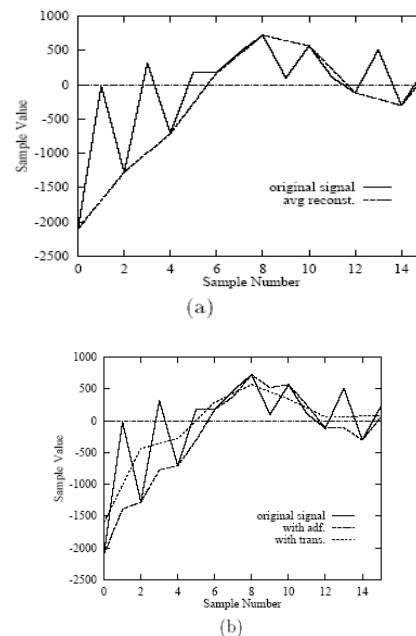


Figure 1: Reconstruction in two-way interleaving.

There are also quite a few algorithms that do not add any redundancy to the data sent. Instead, they utilize the inherent redundancy of the source input stream.

A typical method based on interleaving transmits interleaved voice samples in one packet and reconstructs approximately lost samples using their surviving neighbors contained in other packets. One simple but effective method [15] is to group all the odd samples into one packet and all the even samples into another, and then send them independently. We call the two packets with the corresponding even and odd samples an interleaving pair. When one of the packets is lost, the receiver can easily use the average of samples in the other packet to reconstruct the lost samples (see Figure 1.1). Simple averaging works well for voice signals because most samples are related to their neighboring samples. It is easy, it is fast, and it does not require redundant information to be sent. However, simple averaging may fail when signals are rapidly changing or when signals are totally independent. In the later case, averaging amounts to adding noise to fill the missing gaps.



**Fig 2.** Comparison of reconstruction quality among simple averaging, averaging with adaptive filtering, and averaging based on transformation, assuming that only the even samples are available. The original voice samples are in solid lines. (a) The odd samples were reconstructed by taking the average of its two adjacent even samples (reconstruction by averaging) leading to SNR of 1.69 dB.

(b) The data that were reconstructed using averaging and adaptive \_ltering are in dashed lines with SNR of 3:18 dB. The performance given by the dotted line was obtained by \_rst transforming the even samples before they were sent and by reconstructing the odd samples using simple averaging at the receiver (SNR of 4:03 dB).

In short, a combined sender and receiver reconstruction technique generally improves reconstruction quality significantly over a receiver-only reconstruction technique. Trade-offs must be made to balance between the amount of redundancy and the amount of increased network traffic.

#### IV. OUR APPROACHE TO CONCEAL LOSS

To overcome the shortcomings of reconstruction using simple averaging, we investigate in this research two new approaches to reconstruct lost packets built on top of interleaving and simple averaging. The performance measure for reconstruction quality is the signal-to-noise ratio

$$SNR = 10 \log_{10} \frac{\sum(s^2)}{\sum(s - \hat{s})^2}$$

where  $s$  is the original signal and  $\hat{s}$  is the reconstructed signal of  $s$ .

One shortcoming of reconstruction using averaging is that it does not track changes in the original signals. To address this problem, the first approach applies adaptive filtering after reconstruction using averaging at the receiver in order to track the voice signals and to give better reconstruction quality. The algorithm works as follows.

The sender

- Interleaves the original voice stream into two substreams,
- Packetizes the substreams separately, and
- Sends the packets to the receiver.

At the receiver, there are two cases to be considered. First, when the receiver receives both packets in an interleaving pair, the receiver

- De-interleaves,
  - Trains the adaptive filter by assuming one packet is lost, and
  - Feeds the de-interleaved packets to the sound card.
- Second, when the receiver receives only one packet in an interleaving pair, the receiver Reconstructs using average reconstruction,
- Passes the reconstructed stream through the adaptive filter, and
  - Feeds the output of the adaptive filter to the sound card.

Because adaptive filtering needs time to adapt, it may not perform well when the signals are changing rapidly. This motivates us to develop a new transformation-based reconstruction algorithm that transforms the original signals at the sender before sending them out. The transformations are done according to the reconstruction method used at the receiver in order to enable better reconstruction quality.

The algorithm works as follows.

- The sender transforms the voice stream according to the reconstruction method used at the receiver,
- Interleaves the transformed stream,
- Packetizes each substream, and
- Sends the packets to the receiver.

There are two possibilities at the receiver. First, the receiver receives both packets in an interleaving pair. The receiver then

- De-interleaves the stream, and
- Feeds the de-interleaved packets to the sound card.

Second, the receiver receives only one packet in an interleaving pair. The receiver then

- Reconstructs using simple averaging, and
- Feeds the reconstructed stream to the sound card.

To illustrate the basic ideas of the two reconstruction methods, consider a simple two-way interleaved data stream based on a typical segment of voice data with 16 samples. Assuming that the odd samples were lost at the receiver and that the eight even samples were used to reconstruct the missing samples, Figure 2(a) plots the reconstructed stream in which a missing odd sample is computed as the average of its two adjacent even samples. In contrast, Figure 2(b) shows the reconstructed streams of our proposed reconstruction methods. In the method based on averaging and adaptive filtering, the missing samples are first reconstructed using averaging, and the stream is then passed through an adaptive filter. After adaptive filtering, the reconstructed stream is a better approximation of the original one because adaptive filtering can track the shape of a waveform.

However, adaptive filtering is not able to adapt to rapid changes that span only a few samples in the original signals. In contrast, the dotted line in Figure 2(b) shows the data reconstructed by our transformation-based method. The even samples were first transformed at the sender, and the receiver uses the transformed even samples to reconstruct the missing odd samples. It is obvious that the reconstructed stream based on the transformed samples gives a better approximation to the original stream.

Therefore, the reconstructed samples based on averaging are more accurate.

#### V. CONCLUSION

This paper investigates the necessary components of a high-quality real-time voice transmission system, and presents new reconstruction algorithms that can achieve better reconstruction quality. Reliable transmissions of real-time voice streams over the Internet are difficult because no underlying protocol supports transfers with given QoS requirements. TCP is not realistic to deal with delay-sensitive multimedia data. By using UDP, a connectionless, best-effort transport-layer protocol, applications need to handle out-of-order packets, loss, long delay, and large jitter that are inherent in the resource-limited Internet.

Voice traffic has unique characteristics. It is regular, has relatively smaller packet size and low bandwidth requirement as compared to video traffic. Experiments show that voice-traffic behavior is time varying and destination dependent. Also, a voice stream may encounter large jitters and high loss, while the length of consecutive packet losses is rather small (usually 1-2). These observations make interleaving and reconstruction both practical and attractive. It is practical because interleaving will not introduce too much delay, usually only one more packet delay at each end. It is attractive because it does not need to send redundant information to overcome loss.

In this paper, we have proposed new schemes for handling loss via interleaving and reconstruction.. Existing reconstruction methods that recover missing signals based on the average of adjacent samples are quite simple and generally work well. However, they do not take advantage of any signal-specific information and have difficulty in dealing with rapidly changing voice signals. Based on averaging and adaptive

filtering, the missing samples are then reconstructed using averaging passed through an adaptive filter. After adaptive filtering, the reconstructed stream is a better approximation of the original one because adaptive filtering can track the shape of a waveform.

## VI. REFERENCES

- [1]. L. Sweet, "Toss your dimes-Internet video phones have arrived," ZD Internet Magazine,August1997, <http://www.zdnet.com/products/content/zdim/0208/zdim0010.html>.
- [2]. M. S. Shuster, Diffusion of network innovation: implications for adoption of Internet services," M.S. thesis, Massachusetts Institute of Technology, Cambridge, MA, June 1998.
- [3]. S. Bass, "Internet phones take on Ma Bell," PC World, June 1997, <http://www.pcworld.com/software/internet/www/articles/jun97/1506p165.html?SRC=mag>.
- [4]. R. Ramjee, J. Kurose, D. Towsley, and H. Schulzrinne, "Adaptive playout mechanisms for packetized audio applications in wide-area networks," in Proceedings of the 13th Annual Joint Conference of the IEEE Computer and Communications Societies on Networking for Global Communication, vol. 2, 1994, pp. 680-688.
- [5]. J. C. Bolot and A. V. Garcia, "Control mechanisms for packet audio in the Internet," in Proceedings of IEEE INFOCOM, April 1996, pp. 232-239.
- [6]. Z. G. Chen, "Coding and transmission of digital video on the Internet," Ph.D. dissertation, University of Illinois, Urbana-Champaign, IL, 1997.
- [7]. B. Dempsey, T. Strayer, and A. Weaver, "Adaptive error control for multimedia data transfers," in Proceedings of International Workshop on Advanced Communications and Applications for High Speed Networks, March 1992, pp. 279-289.
- [8]. G. Held, Voice Over Data Networks. New York: McGraw-Hill, 1998.
- [9]. T. J. Kostas, M. S. Borella, I. Sidhu, G. M. Schuster, J. Grabiec, and J. Mahler, "Real-time voice over packet-switched networks," IEEE Network, vol. 12, pp. 18-27, January-February 1998.
- [10]. B. J. Dempsey and Y. Zhang, "Destination buffering for low-bandwidth audio transmission using redundancy-based error control," in Proceedings of 21st IEEE Local Computer Networks Conference, October 1996, pp. 345-355.
- [11]. V. Hardman, M. A. Sasse, M. Handley, and A. Watson, "Reliable audio for use over the Internet," in International Networking Conference, June 1995, pp. 171-178.
- [12]. Telogy Networks, "Voice over packet tutorial," November 1997, <http://www.webproforum.com/telogy/full.html>.
- [13]. N. Shacham and P. McKenney, "Packet recovery in high-speed networks using coding and buffer management," in Proceedings of IEEE INFOCOM, May 1990, pp. 124-131.
- [14]. J. C. Bolot and P. Hoschka, "Adaptive error control for packet video in the Internet," in International Conference on Image Processing, vol. 1, September 1996, pp. 25-28.
- [15]. N. S. Jayant and S. W. Christensen, "Effects of packet losses in waveform coded speech and improvements due to odd-even sample-interpolation procedure," IEEE Transactions on Communications, vol. 29, pp. 101-110, February 1981.

# Application of Filter Bank in Software Defined Radio (SDR)

Gagandeep Kaur <sup>1</sup> Vikesh Raj <sup>2</sup>

<sup>1</sup>. Lecturer, YCOE, Talwandi Sabo, Guru Kanshi Campus Punjabi University, Patiala

<sup>2</sup>. Executive (R&D), MIRC Electronics (ONIDA), Mumbai

**Abstract** - In the past, radio systems were designed to communicate using one or two wave forms. As a result, two groups of people with different types of traditional radio were not able to communicate due to incompatibility. The need to communicate with people using different types of equipment can only be solved by using Software Defined Radio (SDR).

Software defined Radio (SDR) is a wireless device that works with any communication system, be it a cellular phone, a pager, a WiFi transceiver an AM or FM radio, a satellite communications etc. In both hardware and software DSP techniques are used to design a real time system. Various parameters like compression, noise cancellation, detection and estimation can be done.

**Index Terms** - Software Defined Radio (SDR), Hardware Functional Block of SDR, Software Functional Block of SDR, DSP Real Time Software Technology, Filter bank, ADC, DAC.

## I. INTRODUCTION

The future wireless environment is expected to consist of multiple radio access standards that provide users different level of mobility and bandwidth. The superior re-configurability and re-programmability of Software Defined Radio (SDR) had made itself become the most promising technology to realize such a flexible radio system.

Software Defined Radio technology is also attractive for future mobile communication because of re-configurable and multimode operation capabilities. The re-configurable feature is useful for enhancing function of equipment without replacing hardware. Multimode operation is essential for future wireless terminal because a number of wireless communication standards will still co-exist.

A Software Defined Radio can be considered as an open architecture that creates a communication platform by connecting modularize and standardized flexible hardware building blocks. Software load define task and interconnects between the blocks and gives an identity to the system. SDR Forum describes SDRs as “radio that provide software control of a variety of modulation techniques wide and narrow band operation, communication security functions and wave form requirement of current and evolving standards over a broad frequency range”. Software Defined Radio concept promises the man solution for supporting a multitude of wireless communication services in a single infracture design.

## II. FUNCTIONAL BLOCK OF SDR

The main blocks of a true Software Defined Radio are namely [1]

Intelligent Antenna

Programmable RF Module

High Performance DAC and ADC

Digital Signal Processing Techniques

The Interconnect Technology

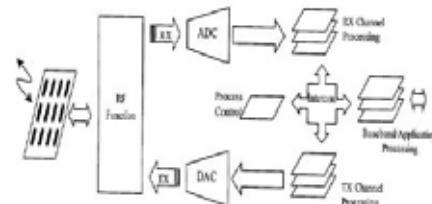


Fig1: SDR Decomposition

## III. ROLE OF ADC IN SDR

Sampling rate of Analog to digital converters (ADCs) is the main parameter to decide the performance of digital Signal processing system working on analog input . ADCs are the main part of all signal processing and communication systems, sometime they are most critical component because they decide the overall presentation of the system. Thus performance, limitation and realizations are important parameters of ADCs to be decided by the designer. There are several existing analog-to-digital conversion techniques, which can be categorized as flash converters, subranging converters, successive-approximation converters, integrating converters, and oversampling sigma-delta converters.

Due to outstanding linearity Oversampling Converters have become a popular technique for data conversion. The main limitation of the Oversampling Converters is the speed, as they are much slower than their Nyquist-rate counterparts. Thus we can use this technique when we require low speed and high linearity. Better bandwidth cab be obtained by the higher order modulators and lower sampling rate but the design of anti-aliasing filter becomes very complex, and hence it losses the key feature of Oversampling Converters i.e. simple design of anti-aliasing filter.

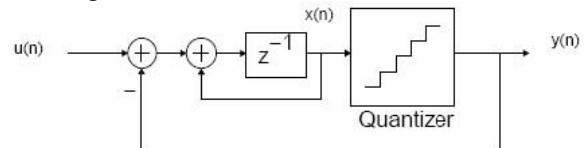


Fig 2: One bit oversampling Converter

In another approach, one can achieve the required sampling rate not by performing more oversampling but by increasing the number of modulators [Z]. The idea of increasing the number of modulator lead to the idea of time-interleaving which is one of the oldest scheme that is used for high sampling rate analog-to-digital conversion

Offset sensitivity, gain mismatch as well as aperture errors between the interleaved channels determine the performance of the time-interleaved ADCs. There are several techniques that can be used to overcome the above mentioned problem such as hybrid filter banks, pipelining, and digital filter banks. These techniques increase the bandwidth and resolution of the system. One approach that has been proposed in the literature in case of Hybrid Filter Banks is to design the analog filter bank from a digital prototype. Considering the magnitude responses are not sufficient in order to control the aliasing, and, hence, the use of the bilinear transformation is not sufficient. Prototyping the digital filter bank into analog filter bank has a limitation i.e. the order of analog filter bank tends to become higher. For example, when using higher-order transformations, the resulting filter order of the analog filters becomes the product of the digital prototype and the order of the transformation. The first challenge in constructing an HFB is to design the analog and digital filters in the filter bank to provide adequate channel separation and accurate reconstruction of the converted signal. As with conventional filter banks, the HFB itself can introduce gain and phase *distortion* and *aliasing* error into the system, even if the subband conversion is distortionless. Proper design of the filters will minimize the errors introduced by the HFB while still providing good filter characteristics (i.e., sharp cutoff and large stopband attenuation). By first designing the analog filters the complexity of these can be minimized, given that the frequency selective requirements due to the dynamic performance requirements of the HFB ADC are met. Then, with a fixed analysis filter bank, the digital filters can be designed in order to meet the requirements on the distortion and aliasing. By this way the analysis and synthesis filter banks will have different complexity.

#### IV.OVERSAMPLING

In signal processing , oversampling is the process of sampling a signal with a sampling frequency significantly higher than twice the bandwidth or highest frequency of the signal being sampled. An oversampled signal is said to be oversampled by a factor of  $\beta$ , defined as

$$\beta = f_s/2B$$

where

$f_s$  is the sampling frequency

$b$  is the bandwidth of the signal

Oversampling in general can bring two advantages. First, the specification of the anti-aliasing and the reconstruction filters is reduced from the Nyquist specification.

#### V.TIME INTERLEAVING

Time-interleave represents one of the possible approaches that can be used to implement ADC using the concept of multirate signal processing. By using  $M$  interconnected modulators working in parallel with each running at the same clock, the effective sampling rate becomes  $M$  times the clock rate of each modulator. In other words, one can achieve the required sampling rate not by performing more oversampling but by increasing the number of modulators. Hence the required resolution is obtained without utilizing a faster and

more costly fabrication processor or using a higher order modulator.

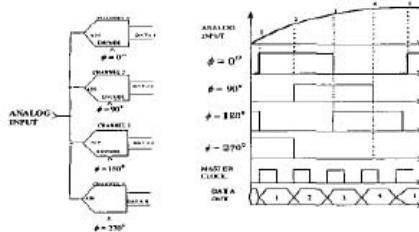


Fig. 3 . Time-Interleaving

#### VI.Hybrid Filter Bank

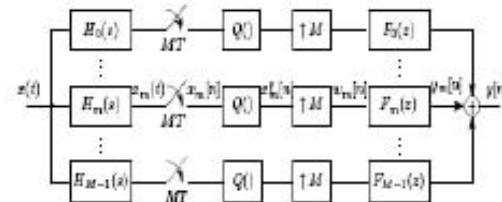


Fig. 4. General structure of an ADC using HFB with maximally-decimated architecture

Fig. 4 shows an example of an ADC using HFB.

The input signal  $x(t)$  is split into  $M$  sub-band signals  $x_m(t)$ ,  $m = 0, \dots, M-1$  via the  $M$  continuous-time analysis filters  $H_0(s), H_1(s), \dots, H_{M-1}(s)$ . Each sub-band signal is down-sampled at  $1/MT$ , and quantized. Then, the digitized signals are upsampled by  $M$  and the sampled version of the input signal is reconstructed via the discrete-time synthesis filters  $F_0(z), F_1(z), \dots, F_{M-1}(z)$ . The goal of the synthesis stage of the HFB is to reconstruct the original signal. Neglecting the effects of quantization, the synthesis stage of the HFB would have firstly to eliminate the aliasing terms due to down-sampling and secondly to reconstruct the original signal by compensating the distortion effects due to spectral aliasing with the sampling rate compressor and spectral imaging with the sample rate expander.

##### a. Frequency-domain analysis

The nonlinear part due to quantizers are hereby neglected. The Fourier transform of the output signal can be written:

$$Y(e^{j\omega}) = \sum_{m=0}^{M-1} F_m(e^{j\omega}) X_m(e^{j\omega M}), \quad (1)$$

with  $\omega = \Omega T$  and:

$$X_m(e^{j\omega}) = \frac{1}{MT} \sum_{k=-\infty}^{+\infty} H_m \left( \frac{j\omega}{T} - \frac{2\pi j k}{MT} \right) X \left( \frac{j\omega}{T} - \frac{2\pi j k}{MT} \right). \quad (2)$$

Noting  $H_m(j\Omega)$  the  $2/T$  periodic extension of the analysis filter  $H_m(j\Omega)$  and  $X(j\Omega)$  the  $2/T$  periodic extension of the input signal  $X(j\Omega)$ , (1) can be rewritten as follows:

$$Y(e^{j\omega}) = \sum_{m=0}^{M-1} T_m(e^{j\omega}) \tilde{X} \left( \frac{j\omega}{T} - \frac{2\pi j m}{MT} \right), \quad (3)$$

Where

$$T_m(e^{j\omega}) = \frac{1}{MT} \sum_{k=0}^{M-1} \tilde{H}_k \left( \frac{j\omega}{T} - \frac{2\pi jm}{MT} \right) F_k(e^{j\omega}). \quad (4)$$

$T_0(e^{j\omega})$  stands for the distortion function and  $T_m(e^{j\omega})$ ,  $m = 1, \dots, M-1$  are the  $(M-1)$  terms of the aliasing function.

### b. Perfect reconstruction in band limited case

To make possible a perfect reconstruction for all input signals, the input signal  $x(t)$  should be band-limited to  $]l\pi/T, (l+2)\pi/T[$ ,  $l \in \mathbb{Z}$  [5]. Therefore, if the input signal spectrum is null outside the frequency interval  $]-\pi/T, \pi/T[$  for example, it is necessary to extend  $H_m(j\Omega)$  considering only its  $\pm\pi/T$  duration. It is clear that

there are only three replica of  $eX_m(j\Omega)$  which take part in  $Y(e^{j\omega})$ ,  $|\omega| < \pi$  through  $X'm(j\Omega)$ . These are  $X_m(j\Omega)$  itself and its replicas  $X_m(j\Omega \pm 2\pi/T)$ . Therefore, for  $|\omega| < \pi$  and  $k = 0, \dots, M-1$ :

$$T_m(e^{j\omega}) = \frac{1}{MT} \sum_{k=0}^{M-1} H_k^s \left( \frac{j\omega}{T} - \frac{2\pi jm}{MT} \right) F_k(e^{j\omega}), \quad (5)$$

Where

$$\begin{aligned} H_k^s \left( \frac{j\omega}{T} \right) &= H_k \left( \frac{j\omega}{T} \right) \\ &+ H_k \left( \frac{j\omega}{T} - \frac{2\pi j}{T} \right) + H_k \left( \frac{j\omega}{T} + \frac{2\pi j}{T} \right) \end{aligned} \quad (6)$$

For the case of perfect reconstruction for all input signals, the distortion function  $T_0(e^{j\omega})$  should be a pure delay all over the band and aliasing  $T_m(e^{j\omega})$  ( $m = 1, \dots, M-1$ ) are undesirable terms. There will be  $M$  equations, which every equation is defined throughout a period of  $\omega$ , for example  $\pm\pi$ :

$$\begin{aligned} T_0(e^{j\omega}) &= ce^{-j\omega\tau} & |\omega| < \pi \\ T_m(e^{j\omega}) &= 0 & m = 1, \dots, M-1 \quad |\omega| > \pi, \end{aligned} \quad (7)$$

where  $\tau \in \mathbb{Z}^+$  is the filter bank's delay,  $c \in \mathbb{R}$  is a scale factor.

### C. Design method

Even if the input signal is band limited, analog filters cannot be band limited and the HFB design procedure leads to discontinuities of the synthesis frequency responses. Therefore, (7) cannot be exactly satisfied. Synthesis methods aim at finding an HFB that minimizes the distortion and aliasing. Several methods may be found in the literature. We chose the global frequency domain least square solving method since it gives the best results in terms of distortion and aliasing. The input signal is assumed band limited to  $\pm\pi/T$ . Starting with the knowledge of frequency responses of analog filters (for the sake of analog feasibility), the synthesis filter bank includes a set of  $M$   $L$ -coefficients FIR filters to design.

Perfect reconstruction conditions (7) are then written for each of the  $N$  frequency points  $\omega_n$  equally distributed in  $\pi < \omega_n < \pi$ .

Noting  $H$  the  $MN \times MN$  matrix of the frequency response of

the analysis filters calculated at the selected frequency values and  $F$  the  $MN \times 1$  matrix of the associated frequency response of the synthesis filters, (7) gives:

$$HF = t, \quad (8)$$

With

$$t = c [e^{-j\omega_1\tau} \dots e^{-j\omega_N\tau} 0_{(M-1)N}]^T, \quad (9)$$

where  $0_{(M-1)N}$  is the  $1 \times (M-1)N$  raw vector filled with zeros. A delay  $\tau$  equal to half the FIR filter length ( $\tau = L/2$ ) was used in the simulations

If  $f$  represents the  $ML \times 1$  FIR coefficients matrix of the synthesis filter bank, (8) can be written:

$$(HD)f = t. \quad (10)$$

where  $D$  stands for the  $MN \times ML$  matrix of the Discrete Fourier Transform coefficients. If  $N > L$ , the linear system (10) is over determined. However, a least square solution can be found [9]:

$$\begin{pmatrix} \text{Re}\{(HD)f\} \\ \text{Im}\{(HD)f\} \end{pmatrix} = \begin{pmatrix} \text{Re}\{t\} \\ \text{Im}\{t\} \end{pmatrix} \quad (11)$$

Where  $\text{Re}\{A\}$  and  $\text{Im}\{A\}$  denote the real and imaginary parts of matrix  $A$ .

In order to dramatically increase the filter bank performances and reach acceptable levels of aliasing, chooses to slightly rise the sample frequency against the band of interest. An oversampling ratio of  $\eta\%$  is used.

### VII.FLOW CHART OF THE ALGORITHM

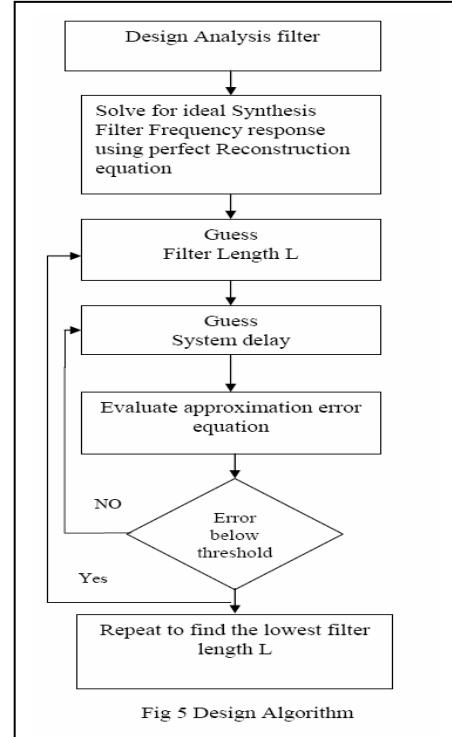


Fig 5 Design Algorithm

### VIII. CONCLUSION

As future A/D converters will have to deal with wide bandwidths, which is not likely to be possible with today's solutions. HFB A/D converters appear to be appropriate, especially because they allow wider bandwidth. A small oversampling allows reaching great performances through the band of interest.

### IX. REFERENCES

- [1]. D.S. Dawoud, S.E. Phakathi "Advanced Filter Bank based ADC for Software Defined Radio Applications". IEEE Africon 2004
- [2]. Scott R. Velazquez, Truong Q. Nguyen " Design of Hybrid Filter Banks for Analog/Digital Conversion" IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL.46, NO.4, APRIL 1998.
- [3]. Daniel Poulton," Anti-aliasing Filter in Hybrid Filter Banks".
- [4]. Lelandais-Perrault, D.Poulton, and J.Oksman,"Synthesis of hybrid filter banks for a/d conversion with implementation constraints-direct Approach", Proceedings of IEEE Midwest Symposium for Circuits and Systems, December 2003.
- [5]. P. Lowenborg, "Analysis and asymmetric filter banks with application to analog-to-digital conversion", Ph.D. thesis, Institute of Technology - Linkopings universitet, may 2001.
- [6]. P. Lowenborg, H. Johansson, and L. Wanhammar, "A two-channel hybrid analog and iir filter bank approximating perfect magnitude reconstruction", Proceedings of IEEE Nordic Signal Processing Symposium, vol. vol. 1, June 2000.
- [7]. Shafiq M. Jam& D. FuPaul J. Hunt, and Stephen H. Lewis, "A IO-b 120-Msamplefs Time-Interleaved Analog-to-Digital Converter with Digital Background Calibration. " IEEE 1, Solid-State Circuits, vol. 37, no. 12, Dec. 2002
- [8]. Ramin Khojai-Poorfard and David A. Johns, "Time-Interleaved Oversampling Converters: Theory and Practice." IEEE Trans. Circuits Svst. 11, vol. 44, no. 1 . 8, Aug. 1997
- [9]. Technical Description, "Advanced Filter Bank (AFB) Analog-to-digital converter Technical description, " v-COT Technologies, <http://www.vcoro.com/analog-interbank.htm>
- [10]. P. Lowenborg, H. Johansson, and L. Wanhammar, "A two-channel hybrid analog and IIR digital filter bank approximating perfect magnitude reconstruction, " in Proc. IEEE Nordic Signal Processing Symp, Kolmarden, Norrkoping, Sweden, June, 2000

# cMote: Development of a 8051-Based Mote and Porting it with TinyOS 2

Rajvir Singh\*, Prof. R.S. Uppal\*\*, Prof. Archana Mantri\*\*\*

\*Senior Lecturer, Chitkara Institute of Engineering and Technology, Rajpura

\*\*HOD, ECE, BBSBEC, Fatehgarh Sahib (Punjab)

\*\*\*Director Academics, Chitkara Institute of Engineering and Technology, Rajpura

[rajvir.singh@chitkara.edu.in](mailto:rajvir.singh@chitkara.edu.in), [rsuppal@gmail.com](mailto:rsuppal@gmail.com), [archana.mantri@chitkara.edu.in](mailto:archana.mantri@chitkara.edu.in)

**Abstract—** A ‘Mote’ is a hardware platform that consists of a radio transceiver, microcontroller, one or more sensors and an energy source. Many of the educational institutes in India do not have access to the commercially available motes, as they are costly and need to be imported. The motive behind this initiative is to develop a simple and cost-effective hardware platform for educational research in the field of wireless sensor networks. The hardware design of the mote is based on nRF24E1, a VLSI chip manufactured by Nordic Semiconductor. The mote will have an on-board temperature sensor and LED’s for debugging and testing purposes. The software platform used for the mote is TinyOS 2, an open-source operating system especially designed for wireless embedded sensor networks. TinyOS 2 currently supports 8051 based platforms through its new initiative of TinyOS8051 workgroup. A new hardware platform with the name ‘cMote’ was created to test the porting of our mote with TinyOS 2.

**Keywords-** Wireless Sensor Networks, 8051-based Hardware Platform, TinyOS8051 workgroup.

## I. INTRODUCTION

A Wireless Sensor Network (WSN) consists of a large number of cooperating small-scale nodes capable of limited computation, wireless communication, and sensing capability [1]. Recent advances in micro-electro-mechanical systems (MEMS) technology, wireless communications, and digital electronics have enabled the development of low-cost, low-power, multifunctional sensor nodes that are small in size and communicate untethered over short distances [2]. Due to their small size these wireless nodes are also referred to as ‘motes’. The sensor units in the motes can perform in situ sensing of any desired parameter (like temperature, pressure humidity, movement etc.) in an area covered by the network. The architecture of a typical wireless sensor network is shown in Figure 1.

## II. MOTIVATION AND OBJECTIVES

A number of commercial-off-the-shelf motes are available in the market nowadays. However, most of them are not available in India and need to be imported. The cost of procuring the motes in sufficient quantity is also prohibitive for an experimental work. To design and use your own motes thus appears to be an indigenous and cost-effective way to carry out educational research in the field of WSN’s. The use of the familiar 8051 family for the new mote would further help in keeping the hardware design simple and easy to implement.

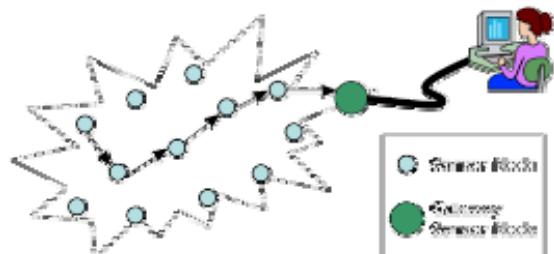


Fig. 1 A typical Wireless Sensor Network

The 8051 family has been thoroughly tested, relatively cheap and has a reasonable small footprint - which makes it ideal for embedded solutions and sensor networks [3]. We have selected TinyOS [4] as the software platform for our system. It supports the 8051 architecture through its new initiative of TinyOS8051 workgroup [5]. The use of TinyOS also ensures the basic compatibility at the programming level with a variety of other motes that are currently available [6]. The objectives of this work are summarized below as-

- i. Design of a low-cost, 8051-based hardware platform for wireless sensor networks.
- ii. To build a working prototype that can be improved upon to implement most of the features of a commercial mote.
- iii. Porting of the Mote with TinyOS 2

## III. DESIGN CONSIDERATIONS AND GUIDELINES

The guidelines followed for designing this hardware platform are given below:

- The design should be based on one of the 8051 family of microcontrollers supported by TinyOS8051 workgroup.
- The programming interface should be kept simple and easy to implement.
- Provision of expansion connectors for connecting other I/O devices like LCD.
- Provision of LED’s for debugging the mote.
- Integration of a temperature sensor on the mote.
- Availability of free or open-source development tools.

## IV. DESIGN AND IMPLEMENTATION STEPS

The design steps given below are the important milestones that were identified on the roadmap prepared for carrying out this work.

- a) Choice of microcontroller
- b) Architecture of the mote
- c) Prototype design

d) Porting and testing of hardware platform with TinyOS

#### A. Choice of Microcontroller

The design guidelines for the mote clearly specify the use of 8051 family of microcontroller supported by the TinyOS8051 workgroup. Presently the TinyOS8051 workgroup supports only three microcontrollers [5] listed below:

- CC2430 from Texas Instruments
- nRF24E1 from Nordic Semiconductor
- C8051F340 from SiLabs

We have chosen nRF24E1 for our mote in this work. The reasons for choice are as follows-

- It can be programmed easily through a simple SPI interface [7] using a universal programmer
- It has 9 analog channels to connect up to 9 sensors
- Combines 8051 core and a radio transceiver on the same chip, thus making our design more compact

The other two devices were not used for the following reasons-

- a) The programming interface of CC2430 requires implementation of an external programming device [8].
- b) C8051F340 from SiLabs also has a nontrivial programming interface. In addition to that it needs a separate radio transceiver chip [9].

#### B. Architecture Of The Mote

The architecture diagram of the mote based on the nRF24E1 chip is shown below in fig. 2. A microcontroller is a central component of the mote. The whole functionality of the mote is built around it. Another important component of the mote is the radio which enables the mote to communicate with other nodes in a wireless sensor network. The architecture of our mote is built around nRF24E1 which has an embedded 8051 compatible microcontroller along with nRF2401 2.4GHz radio transceiver on a single chip. This will make our mote compact and small in size. The programming interface of the mote is provided by 25AA320, a 32K serial EEPROM which is connected to nRF24E1 using SPI interface. The mote also has a precision temperature sensor DS18S20 embedded on it. Three LED's are provided on the mote to

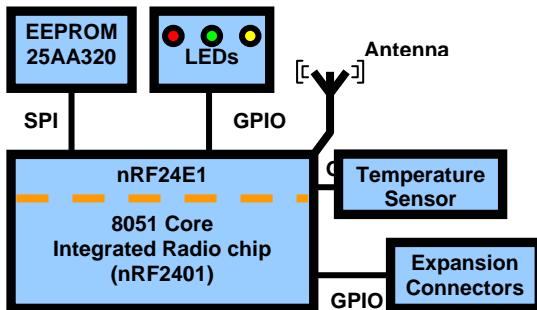


Fig. 2      Architecture of the Mote

test and debug TinyOS applications on the mote. Expansion connectors are provided on the mote to connect any other peripheral devices or sensors to the mote.

#### C. Prototype Design

The prototype design consists of two main modules – *Processor-Radio* module and *I/O* module. The processor-radio module is a separate entity which consists of a nRF24E1 and its associated circuitry housed as a single unit. I/O module consists of various I/O devices along with the interfacing circuitry.

##### 1. Design of Processor-Radio module

A reference design recommended by the Nordic Semiconductor [10] was used. However it was found that the reference PCB design cannot be used directly for our mote. The following changes were made to the design:

- The reference design uses a SMD EEPROM with SO8 size. In order to use a universal programmer to program the EEPROM, we had to replace it with the PDIP package.
- The provision for expansion connectors for the I/O pins and 9 analog inputs was made in the design.
- The footprint of the crystal was changed according to the procured crystal size.
- Provision for connecting a single-ended 50 ohm antenna was made.
- A power-pin connector along with a voltage regulator IC was incorporated in the design.

The fabrication of the PCB was ordered online at URL [www.PCBPower.com](http://www.PCBPower.com).

##### 2. Design of Input/Output module

The reference design we have used for the processor-radio module does not include any I/O devices. So we have designed an Input/Output module which has LED's and a temperature sensor mounted on it. This was done to test the interfacing of LED's and temperature sensor before making them a part of our processor-radio module. The I/O module can be connected to the processor-radio module by using a connector cable.

##### i. Interfacing LED's with the Mote

Three LED's (Red, Green and Yellow) have been connected to the mote using general purpose I/O pins of nRF24E1. To maintain compatibility with the TinyOS code we have kept the wiring same as the nRF24E1\_EVBOARD platform [11] developed as a part of TinyOS8051 workgroup initiative.

##### ii. Interfacing Temperature Sensor with the Mote

The temperature sensor DS18S20 is a high-precision digital thermometer with 9-bit resolution [12]. It is interfaced to processor-radio module using a single-wire serial interface. The wiring of DS18S20 is done with I/O pin P1.6 of nRF24E1. The mounting of sensor is also done on I/O module and the connections to processor-radio module are done using a connector cable.

#### E. Porting and Testing of Hardware Platform with TinyOS

We need to test the working of our hardware platform with TinyOS. We have named our platform as ‘cMote’, with which

it will be identified by TinyOS. The steps for porting a new hardware platform are given in [13]. After creating our new platform for TinyOS with name ‘cMote’, we now proceed to test our platform by using a BlinkNoTimerTask application. Figure 3 shows the screenshot of the successful compilation of the application for our platform ‘cMote’.

## V.RESULTS

In this work we have described design and development of a hardware platform named ‘cMote’ for use in wireless sensor networks. The results achieved from this work are discussed as under:

- We have been successful in achieving our twin objectives of keeping our design simple and low-cost. The total cost of our prototype-mote comes out to be Rs 2500/- only. This includes the BOM cost (Rs 1645/-) for our prototype design and the PCB fabrication cost (Rs 850/-) of the Processor-radio module and I/O module. In comparison the price for procuring a commercial mote would be anything from Rs 10,000/- to 30,000/- per mote. This cost factor would be significantly enhanced when we consider that to deploy a wireless sensor network even in a single building we would be requiring many such motes.

## VII. REFERENCES

- K. Romer, O. Kasten and F. Mattern. Middleware Challenges for Wireless Sensor Networks. Mobile Computing and Communications Review, Volume 6, Number 2.
- I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci. “Wireless Sensor Networks: A Survey”. Computer Networks Journal, 38(4):393-422, 2002
- A.E. Peterson, S. Jensen and M. Leopold. Towards TinyOS for 8051. Technology Enhancement Proposal (TEP) – 121.
- J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, “System Architecture Directions for Networked Sensors”, *Architectural Support for Programming Languages and Operating Systems 2000*, pages 93-104.
- <http://www.tinyos8051wg.net/platforms>
- Kling, R. “Intel Mote: An Enhanced Sensor Network Node”. Proceedings of the International Workshop on Advanced Sensors, Keio, Japan, November 2003.
- NRF24E1 Datasheet. [www.keil.com/dd/docs/datasheets/\\_nordic/nrf24e1.pdf](http://www.keil.com/dd/docs/datasheets/_nordic/nrf24e1.pdf)
- CC2430 Datasheet. <http://focus.ti.com/lit/ds/symlink/cc2430.pdf>
- C8051F340 Datasheet. [http://www.keil.com/dd/docs/\\_datasheets/silabs/c8051f34x.pdf](http://www.keil.com/dd/docs/_datasheets/silabs/c8051f34x.pdf)
- <http://www.nordicsemi.com>
- <http://www.tinyos8051wg.net/nRF>
- DS18S20 Datasheet. <http://datasheets.maxim-ic.com/en/ds/DS18S20.pdf>
- <http://www.tinyos.net/tinyos-2.x/doc/>

```

OBJECT TO HEX FILE CONVERTER OH51 V2.6
COPYRIGHT KEIL ELEKTRONIK GmbH 1991 - 2001

GENERATING INTEL HEX FILE: app.hex

OBJECT TO HEX CONVERSION COMPLETED.
compiled BlinkNoTimerTaskAppC to a CMote binary

MODULE INFORMATION: STATIC OVERLAYABLE
Code size of app.o
MODULE INFORMATION: STATIC OVERLAYABLE
CODE SIZE : 727
CONSTANT SIZE : 4 12
XDATA SIZE : 12
PDATA SIZE :
DATA SIZE :
IDATA SIZE :
BIT SIZE :
Total sizes
Program Size: data=9.0 xdata=10 code=786

```

Fig. 3 TinyOS Application compiled for cMote platform

- We have also successfully ported our 8051-based mote with ‘TinyOS 2’, a popular embedded networked operating system used for developing sensor network applications.

## VI. CONCLUSION

It is concluded that for continuous growth and evolution of an emerging technology the educational and research institutes must have cheap, easy and ready access to the hardware and software platforms. This work has presented development of ‘cMote’ – a 8051-based hardware platform for wireless sensor networks. The porting of ‘cMote’ was done with TinyOS, which supports the 8051 family through its new initiative of TinyOS8051 workgroup. The combination of hardware and software platform as described in this paper will be cost-effective and would contribute towards promoting educational research and experimentation in the field of Wireless Sensor Networks.

# A Survey of Techniques for Software Project Effort Estimation

Rupandeep Kaur -M.Tech student, Punjab Engg College, Chandigarh, India

S. S..Sehra- lecturer, Dept. of Computer Science & Engg, Guru Nanak Dev Engg, College, Ludhiana, Punjab, India e-mail  
S. K.Sehra- Lecturer Dept. of Computer Science & Engg. Guru Nanak Dev Engg, College, Ludhiana, Punjab, India e-mail e-rupandeep.kaur@gmail.com, sukhjitsehra@gmail.com sumeetksehra@gmail.com

**Abstract**— Software Effort estimation is the process of predicting the most realistic use of effort required to develop or maintain software based on incomplete, uncertain and/or noisy input. Estimation of the effort to be spent in a software project is still a complex problem. Improving the estimation techniques available to project managers can facilitate more effective control of time and budgets in software development. In this paper, different methods of effort estimation are investigated.

**Keywords**—Effort Estimation, Fuzzy Logic, Genetic Programming, Linear Regression, MMRE, Neural Networks.

## I. INTRODUCTION

In recent years, software has become the most expensive component of computer system projects. The bulk of the cost of software development is due to the human effort. Accurate software development effort estimates are critical to both developers and customers. Accurate estimation of software project effort at an early stage in the development process is a significant challenge for the software engineering community [17]. Estimates can be used for generating request for proposals, contract negotiations, scheduling, monitoring and control.

Software effort prediction models fall into two main categories: algorithmic and non-algorithmic. Even after several decades of research into software management, the question of how to predict effort with sufficient reliability is still unsolved. Effort Estimation carries inherent risk and this risk can lead to uncertainty. Consequently, there is an ongoing, high level activity in this research field in order to build, to evaluate, and to recommend prediction techniques. A large number of different predictive models have been investigated over the last years. They range from mathematical functions: regression analysis [24] and COCOMO [6] to machine learning models like estimation by analogy [21], clustering techniques [29], and artificial neural networks [2].

The remainder of this paper can be described as follows: Next section contains a description of the methods used for Effort estimation. In Section III results of methods applied on data sets are discussed. The paper ends with conclusions and future directions for the modeling of the software effort estimation.

## II. TECHNIQUES USED IN LITERATURE

### A. Effort estimation by Analogy

Software effort estimation by analogy [21, 30] is mainly a data-driven method. It compares the project under consideration (target project) with similar historical projects through their common attributes, and determines the effort of the target project as a function of the known efforts from

similar historical projects. A decision-centric process model can be implemented by generalizing the existing Effort estimation by analogy. The given data set for EBA is represented as a data base called *Raw Historical Data* that is composed of objects and attributes describing the objects using <Attribute, Value> pairs [13].

“How to compare and then choose an appropriate EBA method for a certain type of data sets” is a very critical question. A method, called COSEEKMO, regarding the selection of best practices of mathematical model based on software effort estimation is proposed in [28]. EBA can be used for effort estimation for objects at levels of project, feature, or requirement, given corresponding historical data sets.

### B. Linear Regression Technique

This method attempts at finding linear relationship between one or more predictor parameters and a dependent variable, minimizing the mean square of the error across the range of observations in the data set [12]. Some researchers have tried building simple local models, using this type of approach. Ordinary least squares regression is the most common modeling technique applied to software estimation [16].

Regression modeling is one of the most widely used statistical modeling technique for fitting a response (dependent) variable as a function of predictor (independent) variables[1]. The resulting prediction systems take the form:

$$y_{ext} = \beta_0 + \beta_1 X_1 + \dots + \beta_n X_n \\ \dots \quad (1)$$

Where  $y_{ext}$  is the estimated value and  $X_1$  to  $X_n$ , are independent variables e.g. project size (in source code lines), that the estimator has found to significantly contribute to the prediction of effort. A disadvantage with this technique is its vulnerability to extreme outlier values although robust regression techniques, that are less sensitivity to such problems, have been successfully used [16]

An approach based on Regression towards Mean is proposed and applied on several data sets. The results indicate that it improves the estimation accuracy. Surprisingly, current analogy based effort estimation include adjustments related to extreme analogues and inaccurate estimation models [18]. Another potential problem is the impact of co-linearity - the tendency of independent variables to be strongly correlated with one another - upon the stability of a regression type prediction system.

### C. Neural Networks

Neural networks are nets of processing elements that are

able to learn the mapping existent between input and output data. The neuron computes a weighted sum of its inputs and generates an output if the sum exceeds a certain threshold. This output then becomes an excitatory (positive) or inhibitory (negative) input to other neurons in the network. The process continues until one or more outputs are generated. [23] reports the use of neural networks for predicting software reliability, including experiments with both feed forward and Jordan networks with a cascade correlation learning algorithm.

The Neural Network is initialized with random weights and gradually learns the relationships implicit in a training data set by adjusting its weights when presented to these data. The network generates effort by propagating the initial inputs through subsequent layers of processing elements to the final output layer. Each neuron in the network computes a non-linear function of its inputs and passes the resultant value along its output [2]. The favored activation function is Sigmoid Function given as:

$$f(x) = \frac{1}{1 + e^{-x}}$$

....(2)

Among the several available training algorithms the error back propagation is the most used by software metrics researchers. The drawback of this method lies in the fact that the analyst can't manipulate the net once the learning phase has finished [10]. Neural Network's limitations in several aspects prevent it from being widely adopted in effort estimation. It is a 'black box' approach and therefore it is difficult to understand what is going on internally within a neural network. Hence, justification of the prediction rationale is tough. Neural network is known of its ability in tackling classification problem. Contrarily, in effort estimation what is needed is generalization capability. At the same time, there is little guideline in the construction of neural network topologies [2].

In recent years, various new methods have been proposed based on neural networks. One of the methods is the use of Wavelet Neural Network (WNN) to forecast the software development effort. The effectiveness of the WNN variants is compared with other techniques such as multiple linear regression in terms of the error measure which is mean magnitude relative error (*MMRE*) obtained on Canadian financial (CF) dataset and IBM data processing services (IBMDPS) dataset. Based on the experiments conducted, it is observed that the WNN outperformed all the other techniques [14].

#### D. Fuzzy Logic

The development of software has always been characterized by parameters that possess certain level of fuzziness. Study showed that fuzzy logic model has a place in software effort estimation [20]. The application of fuzzy logic is able to overcome some of the problems which are inherent in existing effort estimation techniques [4]. Fuzzy logic is not only useful for effort prediction, but that it is essential in order to improve the quality of current estimating models [26]. Fuzzy logic enables linguistic representation of the input and output of a model to tolerate imprecision [1]. It is particularly suitable for effort estimation as many software attributes are measured

on nominal or ordinal scale type which is a particular case of linguistic values [13].

A method is proposed as a Fuzzy Neural Network (FNN) approach for embedding artificial neural network into fuzzy inference processes in order to derive the software effort estimates. Artificial neural network is utilized to determine the significant fuzzy rules in fuzzy inference processes. The results showed that applying FNN for software effort estimates resulted in slightly smaller mean magnitude of relative error (MMRE) and probability of a project having a relative error of less than or equal to 0.25 (Pred(0.25)) as compared with the results obtained by just using artificial neural network and the original model[27].

Another proposal is the use of subset selection algorithm based on fuzzy logic for analogy software effort estimation models. Validation using two established datasets (ISBSG, Desharnais) shows that using fuzzy features subset selection algorithm in analogy software effort estimation contribute to significant results [19]. Another proposal based on same logic is by [11] who propose a hybrid system with fuzzy logic and estimation by analogy referred as Fuzzy Analogy. COCOMO'81 is used as dataset. The use of fuzzy set supports continuous belongingness (membership) of elements to a given concept (such as small software project) [32] thus alleviating a dichotomy problem (yes/no) [31] that caused similar projects having different estimated efforts. Fuzzy logic also improves the interpretability of the model allowing the user to view, evaluate, criticize and adapt the model.

#### E. Genetic Programming

Genetic programming is one of the evolutionary methods for effort estimation. Evolutionary computation techniques are characterized by the fact that the solution is achieved by means of a cycle of generations of candidate solutions that are pruned by the criteria 'survival of the fittest' [30]. A comparison is suggested by [7] based on the well-known Desharnais data set of 81 software projects derived from a Canadian software house in the late 1980s. It shows that GP can offer some significant improvements in accuracy and has the potential to be a valid additional tool for software effort estimation.

Genetic Programming is a nonparametric method since it does not make any assumption about the distribution of the data, and derives the equations according only to fitted values. One limitation of canonical Genetic Programming is the requirement of closure [15] which mean that functions should be well defined. To overcome this problem, a new evolutionary computational method is introduced based on genetic programming, known as Grammar Guided genetic Programming. GGGP because of its flexibility has potential to model other software applications also [25].

### III.RESULTS OF TECHNIQUES USED IN LITERATURE

Different error measurements have been used by various researchers, but the main measure for model accuracy is the Mean Magnitude of Relative Error (*MMRE*) [8].

$$MMRE = \frac{\left( \sum_{i=1}^n i \left| \frac{M_{ext} - M_{act}}{M_{act}} \right| * 100 \right)}{n} \quad .... (3)$$

Where  $M_{ext}$  is the estimated effort and  $M_{act}$  is the actual effort. The other measure is prediction level  $pred(p)$ , which is defined by:

$$pred(p) = \frac{k}{N} \quad \dots \dots (4)$$

Where  $N$  is the total number of observations and  $k$  is the number of observations having value of  $MMRE$  less than or equal to  $p$ . The common value for  $p$  is 0.25 in literature.

#### a) Effort estimation by analogy

Although in this two basic datasets have been used but the Desharnais has been partitioned into sets according to the development environment. Albrecht contains 24 projects especially IBM DP service projects [3] and Desharnais has 77 Canadian software house commercial projects [9].

TABLE I  
MMRE RESULTS FOR DIFFERENT DATASETS USED IN LITERATURE

Dataset	MMRE value
Albercht	62%
Desharnais	64%
Desharnais-1	37%
Desharnais-2	29%
Desharnais-3	26%

From the results shown in the above table-I, it is clear that  $MMRE$  value is smaller, so it can succeed even where no statistical relationships can be found. It does not require calibration or for that matter recalibration [5]. Lastly, it is a more intuitive method so it is easier to understand the reasoning behind a particular prediction. But it is not clear about the effect of old datapoints.

#### b) Linear Regression

The results are obtained with a set of measures taken from COCOMO dataset [6] as shown in table II. The dataset used in this work is COCOMO a public available data set consisting of a total of 63 projects. . The effort is represented by the variable EsforcolT (the amount of man-hour for the software integration and test phase).

TABLE II  
ESTIMATE DVALUES USING LINEAR REGRESSION [6]

. Esforyolt	Estimation value
6,753	26,026
3705	673,847
59,3	93,678
120,05	202,332
1,12	21,885
12,93	48,577
10,58	111,104
375,24	969,057
3,5	26,846

The value of  $MMRE$  for this data set is 4.62. These results indicate that stepwise regression's show strong relationship with the actual development. Linear Regression is not very successful at estimation of very large efforts that are out of all

proportions to their sizes, as well as one with a small effort for its size. An attempt to solve these outliers could influence negatively in the accuracy.

#### c) Neural Networks

The results are obtained with a set of measures taken from COCOMO dataset [6] as shown in table III.

TABLE III  
ESTIMATE D VALUES FOR NEURAL NETWORKS [6]

. Esforyolt	Estimation Value
6,753	26,5586
3705	859,7053
59,3	52,37
120,05	165,4667
1,12	25,5485
12,93	33,0107
10,58	63,0066
375,24	638,452
3,5	26,7646

The value of  $MMRE$  for this data set is 4.2. From the results, it is obvious that neural networks are adaptable and can be tailored to the data at a particular site. Although ANN has demonstrated significant advantages in certain circumstances, it does not replace regression

#### d) Fuzzy logic

The ten programs suggested were used to obtain the test data. Seventy-one modules distributed into ten programs resulted from this task [33].Eighteen of them were at least reused once and twenty-eight were new. Forty-one were selected, the five remaining were considered outliers. The values obtained for this data set are as follows:

TABLE IV  
MMRE AND PRED% ESTIMATED VALUE S FOR FUZZY LOGIC [33]

Dataset of 41 modules	MMRE	Pred(20%)
	0.1057	0.9268

Result showed that the value of  $MMRE$  applying fuzzy logic was slightly higher than Regression.

#### e) Genetic Programming

The data for 423 projects was collected and the projects are drawn from the public ISBSG data repository. From the results shown in the table, it is very much clear that Genetic programming performs better than Regression in all respects. This method can minimize the value of  $MMRE$  to a large extent. This method performs much better on test data rather than training data.

TABLE V  
RESULTS OF ESTIMATE DVALUE S USING FUZZY LOGIC [25]

Data	Mean/Standard	MMRE	Pred(25%)
Training Data	Mean	2.67	0.19
	Standard	0.81	0.62
Test Data	Mean	1.91	0.21
	Standard	0.67	0.02

#### IV.CONCLUSION

In an absolute sense, none of the models perform particularly well at estimating software development effort, particularly along the *MMRE* dimension. But in a relative sense ANN approach is competitive with traditional models. Again as a comparative analysis, genetic programming can be used to fit complex functions and can be easily interpreted. So the research is on the way to combine different techniques for calculating the best estimate.

#### V. REFERENCES

- [1]. Agustín Gutiérrez T., Cornelio Yáñez M.and Jérôme Leboeuf Pasquier, "Software Development Effort Estimation Using Fuzzy Logic: A Case Study", Proceedings of the Sixth Mexican International Conference on Computer Science (ENC'05), 2005.
- [2]. A. Idri, T. M. Khoshgoftaar, A. Abran, "Can neural networks be easily interpreted in software cost estimation?", IEEE Trans. Software Engineering, Vol. 2, 2002, pp. 1162 – 1167.
- [3]. Albrecht, A.J. and J.R. Gaffney, "Software function, source lines of code, and development effort prediction: a software science validation", IEEE Trans. on Softw. Eng., 9(6), 1983, pp 639-648.
- [4]. A. R. Gray, S. G. MacDonell, "Applications of Fuzzy Logic to Software Metric Models for Development Effort Estimation", Fuzzy Information Processing Society 1997 NAFIPS' 97, Annual Meeting of the North American, September 21 – 24, 1997, pp. 394 – 399.
- [5]. Barbara Kitchenham, Martin Shepperd and Chris Schofield, "Effort Estimation by Analogy", Proceedings of ICSE-18, pp 170-178.
- [6]. Boehm B. W., Software Engineering Economics, Prentice-Hall, Englewood Cliffs, NJ, 1981.
- [7]. C. J. Burgess, M. Lefley, "Can Genetic Programming improve Software Effort Estimation? A Comparative Evaluation", Machine Learning Applications In Software Engineering: Series on Software Engineering and Knowledge Engineering, pp. 95–105, May 2005
- [8]. Conte, S., H. Dunsomore, and V.Y. Shen, Software Engineering Metrics and Models. Benjamin Cummings: Menlo Park, CA, 1986.
- [9]. Desharnais, J. M., Analyse statistique de la organisation. productivitie des projects de development en infomatique apartir de la techniques des points de function de fonction. Masters thesis. Universite du Quebec, Montreal, 1988
- [10]. Finnie, G. R., G.E. Wittig and J-M. Desharnais, "A Comparison of Software Effort Estimation Techniques Using Function Points with Neural Networks, Case- Based Reasoning and Regression Models", Journal of Systems and Software, Vol. 39, pp. 281-289, 1997.
- [11]. Idri, A., Abran, A., Khoshgoftaar, T. Fuzzy Analogy: a New Approach for Software Cost Estimation. International Workshop on Software Measurement (IWSM'01), Montréal, Québec, Canada, August 28- 29, 2001
- [12]. Iris Fabiana de Barcelos Tronto, Jose Demisio Simoes da Silva, Nilson Sant'Anna," Comparison of Artificial Neural Network and Regression Models in Software Effort Estimation."Proceedings of International Joint Conference on Neural Networks, Orlando, Florida, USA, pp 771-776, 2007
- [13]. Jingzhou Li and Guenther Ruhe, "Decision Support Analysis for Software Effort Estimation by Analogy", International Conference on Software Engineering archive Proceedings of the Third International Workshop on Predictor Models in Software Engineering, pp 65-106, 2007.
- [14]. K. Vinay Kumar, V. Ravi, Mahil Carr and N. Raj Kiran, "Software development cost estimation using wavelet neural networks", Journal of Systems and Software ,Volume 81, pp 1853-1867, 2008
- [15]. Koza R. John, "Genetic programming: on the programming of computers by natural selection", IT Press, Cambridge, USA, 1992.
- [16]. L.C. Briand, I. Wieczorek, "Software resource estimation," Encyclopedia of Software engineering, 2002, vol. P-Z, no. 2, pp. 1160-1196.
- [17]. Martin Shepperd and Chris Schofield, "Estimating Software Project Effort Using Analogies", IEEE Transactions On Software Engineering, Vol. 23, No. 12, November 1997,pp 736-743
- [18]. M. Jørgensen, D. I. K. Sjøberg, and U. Indahl, "Software Effort Estimation by Analogy and Regression Toward the Mean", Journal of Systems and Software, 68(3), pp. 253-262
- [19]. Mohammad Azzeh, Daniel Neagu and Peter Cowling, "Improving analogy software effort estimation using fuzzy feature subset selection algorithm", Proceedings of the 4th international workshop on Predictor models in software engineering, pp 71-78 , 2008
- [20]. Moon Ting Su1, Teck Chaw Ling, Keat Keong Phang, Chee Sun Liew, Peck Yen Man, "Enhanced Software Development Effort And Cost Estimation Using Fuzzy Logic Model", Malaysian Journal of Computer Science, Vol. 20(2), pp 199-207, 2007.
- [21]. M. Shepperd, C. Schofield, "Estimating Software Project Effort Using Analogies", IEEE Transactions on Software Engineering, Vol. 23, No. 12, 1997, pp 736-743.
- [22]. Musfek, P., Pedrycz, W., Succi, G., Reformat, M., "Software Cost Estimation with Fuzzy Models.", Applied Computing Review, Vol. 8, No. 2, 2000, pp. 24-29..
- [23]. N. Karunanithi, D. Whitley, and Y. K. Malaiya, "Using Neural Networks in Reliability Prediction", IEEE Software, 1992, vol. 9, no.4, pp. 53-59.
- [24]. Sentas, P., Angelis, L., Stamelos, I. and Bleris, G., Software productivity and effort prediction with ordinal regression, Journal Information and Software Technology, 47, pp. 17-29, 2005.
- [25]. Shan, Y., McKay, R. I., Lokan, C. J., Essam, D. L., "Software project effort estimation using genetic programming ", IEEE 2002 International Conference on Communications, Circuits and Systems and West Sino Expositions", pp. 1108 - 1112
- [26]. S. Kumar, B. A. Krishna, and P. S. Satsangi, Fuzzy systems and neural networks in software engineering project management, Journal of Applied Intelligence, no. 4, pp. 31-52, 1994.
- [27]. Sun-Jen Huang and Nan-Hsing Chiu, "Applying fuzzy neural network to estimate software development effort", journal of Applied Intelligence, 2007.
- [28]. T. Menzies, Z. H. Chen, J. Hihn, and K. Lum, "Selecting Best Practices for Effort Estimation", IEEE Transactions on Software Engineering, Vol. 32, No. 11, 2006, pp. 1-13.
- [29]. T. Mukhopadhyay, S. Vicinanza, and M.J. Prietula, "Examining the Feasibility of a Casebased Reasoning Model for Software Effort Estimation", MIS Quarterly, Vol. 16, No. 2, pp 155-171, 1992.
- [30]. Urkola Leire , Dolado J. Javier , Fernandez Luis and Otero M. Carmen , "Software Effort Estimation: the Elusive Goal in Project Management", International Conference on Enterprise Information Systems, pp. 412- 418, 2002.
- [31]. X. Huang, J. Ren and L.F. Capretz, "A Neuro-Fuzzy Tool for Software Estimation", Proceedings of the 20th IEEE International Conference on Software Maintenance, pp. 520 2004
- [32]. Xu, Z. and Khoshgoftaar, T. M., "Identification of fuzzy models of software cost estimation", 2003
- [33]. Zhong, S., Khoshgoftaar, T. M., and Seliya, N., "Analysing Software Measurement Data with Clustering Techniques", IEEE Intelligent Systems, pp. 20-27, 2004.

**Mrs. Rupandeep Kaur** is currently persuing her M.Tech from Punjab Engineering College, Chandigarh. She received her Becholer in Computer Sci Engg from Guru Nanak Dev Engg College, Ludhiana in 2003

**Sukhjit Singh Sehra** is currently working as Lecturer in the Deptt. of Computer Science Engg. at Guru Nanak Dev Engg. College, Ludhiana, India. He has received his B.Tech. from Punjab Technical University and M.Tech. in Computer Science Engg. from Panjab Agricultural University, Ludhiana in 2001 and 2006 respectively. His areas of interests are soft computing, Fuzzy controllers embedded system.

**Sumeet Kaur Sehra** is currently working as Lecturer in the Deptt. of Computer Science Engg. at Guru Nanak Dev Engg. College, Ludhiana, India. She has received her M.Tech. in Computer Science Engg. in 2006 from Punjab Agriculture University, Ludhiana. Her active areas of interests are Soft Computing, Bioinformatics and Artificial Intelligence.

# Line Tracing Robot: A Multi Sensor and Closed Loop Instrumentation Design

Akash<sup>1</sup>, Bibek Kabi<sup>2</sup>, Mr. S.Karthick<sup>3</sup>

<sup>1</sup>Comp. Science & Engineering, <sup>2</sup>Electrical & Electronics Engineering, <sup>3</sup>Lecturer, Comp Science& Engg.  
SRM University, Chennai – 603203, <sup>1</sup>s.akash279@gmail.com, <sup>2</sup>bibek.kabi@yahoo.in, <sup>3</sup>shivkaar@hotmail.com

**Abstract-** A line follower is an autonomous robot which traces a white line on a black surface. Line Tracing Robot is most commonly used system in industries for many applications. In this paper we are proposing a low cost line tracing system implemented with a light chassis, DC Motors, infrared Proximity sensors and manually developed controller board. Controller board contains microcontroller, motor driver and MAX232 circuit. Software side we are controlling the robot using C program using Serial Port Communication. Motion control is based on the differential drive mechanism. Suitable upgrades to the base design have also been implemented at the end of this paper.

**Index terms** - Controller board, microcontroller, motor driver, MAX232, Serial Port.

## I. INTRODUCTION

Line Follower Robot is a system which traces white line on a black surface, or a magnetic field with the help of embedded magnets. In this paper a line tracer has been presented which will trace a white line on a black surface or vice versa. We have made use of sensors to achieve this objective. Sensors work with **analog signals**. They are converted to **digital signals** by the **microcontroller** (It has an in-built ADC) and the digital input is used to drive the motors. The motors work according to the sensor input. For eg: if a sensor on the left gives a high signal it indicates that the line tracer has gone out of path and it must be brought back to the path. Hence appropriate motor signal is given to the motors. The motors have been interfaced using a **motor driver**. The microcontroller has been **programmed** accordingly to make a perfect coordination between the sensor input and the motor output. The programmer circuit which is usually available in the markets has been designed manually. We have made use of **Serial Port Communication** technique to burn the program to the microcontroller.

As a programmer we had the opportunity to teach a robot how to respond to **human-like stimuli of color** and form an **effective closed loop system**.

## II. MICROCONTROLLER PHILIPS P89V51RD2

It belongs to the 8051 family of microcontrollers.

Some advantages of using P89V51RD2 are –

- It has a vast 64 KB programmable flash memory and 1024 bytes of RAM.
- Its most important feature is it has got X2 mode which means the user can use 12 clocks per machine cycle or 6, so by that we can access instructions quickly.
- Easily Available
- Cheap compared to other microcontrollers.
- Easy process of burning

- It can withstand temperatures between the range 0<sup>0</sup> C to 70<sup>0</sup> C.
- It also has a built in Analog to Digital converter.

## 2.2 Basic Algorithm

We have made use of three sensors. The IR sensors will reflect light when they come across a white surface. Let us keep the assumption that when the sensor is on the line it will give a digital output of 1. At the times when it is not on the line it is giving an output of 0.

When the middle sensor is on the white line it will return a 1(high) signal, and the left and right sensor give low (0) value.

L	M	R
0	1	0

When the robot goes a little out of the line, one of the left or right sensors will come on the white line along with the middle sensor.

In case of a right turn –

The middle sensor will return a 1(high), the left sensor will return a 0(high) and the right sensor will return a 1(low).

L	M	R
1	1	0

In case of a left turn –

The middle sensor will return a 1(high), the left sensor will return a 1(high) and the right sensor will return a 0(low) signal.

L	M	R
0	1	1

## 2.3 Sensor Details

Sensors are made of a transmitter and a photo diode. But in order to make it effective an OP-AMP is used. An **OP-AMP** which is usually used in large industries in complex circuits, also finds its application in sensors.

SENSOR SCHEMATIC:-

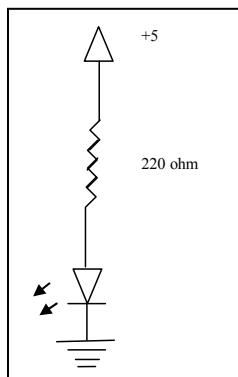


Figure 1.1 IR TRANSMITTER PART

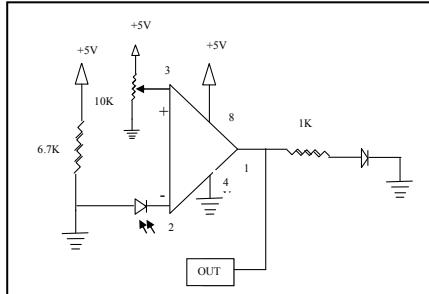


Figure 1.2 IR RECEIVER PART

The line follower sensor contains two parts transmitter and receiver. Considering the transmitter part it consists of an IR led and a resistor (220 ohm) which works on 5V supply, current ranging from 7.5mA to 22.7mA passes through the resistor which limits the current and enable the IR led to transmit IR rays. These rays falling on the receiver or photodiode generates voltage difference across its leads. This voltage difference enable the led to glow by which we come to know that line has been detected. But this voltage difference after undergoing a drop in the depletion region is not sufficient to glow the led so we require an operational amplifier to amplify the signal.

Considering the receiver part the 6.7k resistor is connected to the cathode part of the photodiode .when light falls on the photodiode the voltage difference created on the cathode side undergo some loss in the depletion layer and out coming voltage is the input signal to the inverting input of the op-amp and the potentiometer (10k) is connected to the non-inverting input of the op-amp

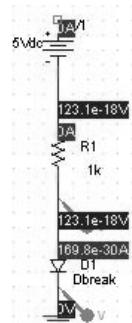


Figure 1.3 IR Receiver with current & voltage values

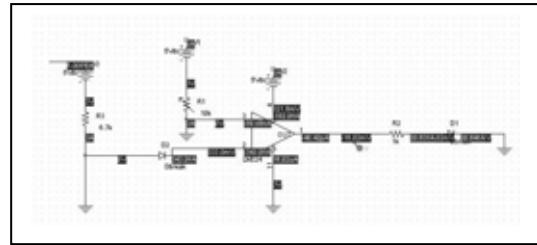


Figure 1.4 IR Transmitter with current & voltage values

As per the circuit supposing at the cathode side of the photodiode 2.5V pass through and it drops to 1.7 V across the anode this 1.7 V is applied to the inverting input. In the op-amp there are two inputs and one output. The inputs are inverting and non-inverting. The inverting input is denoted by minus sign (-) as per the name suggests the input signal is not in phase with the output signal. The other one is non-inverting denoted by (+) sign whose input signal is in phase with the output. if and only if the non-inverting input voltage is more than the inverting input voltage, so if 1.7V is being applied to the inverting input then we should adjust the potentiometer to apply voltage more than 1.7V so that it amplifies.

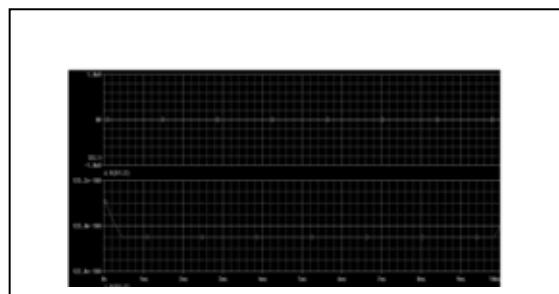


Figure 1.5 Graph of Voltage vs. Time for Receiver



Figure 1.6 Graph of Voltage vs. Time for Transmitter  
MOTOR CONTROL

Differential drive- this is the name given to the motion of any robot, considering a two wheel drive robot we can move the robot in the forward direction by both the wheels moving forward .in the similar manner the vice versa case for backward direction, for left turn we stop the supply for left wheel and give supply to the right wheel and the same vice versa case for right turn.

For a robot to turn, it turns about a point which lies on the axis of the line joining the wheels this point is ICC (instantaneous centre of curvature) .Let the distance between ICC and midpoint of the line joining the wheels be R (say). By varying the velocities of the two wheels, we can vary the trajectories that the robot takes.

Because the rate of rotation  $\omega$  about the ICC must be the same for both wheels, we can write the following equations:

$$\omega(R + l/2) = V_r \dots\dots\dots(1)$$

$$\omega(R - l/2) = V_l \dots\dots\dots(2)$$

At any instance in time we can solve for R and  $\omega$ :

$$R = \frac{l}{2} \frac{V_l + V_r}{V_r - V_l} ; \quad \omega = \frac{V_r - V_l}{l} \dots\dots\dots(3)$$

Definition of parameters:

$l$ : the distance between the centers of the two wheels  
 $V_r, V_l$ : the right and left wheel velocities along the ground respectively.

R: The signed distance from the ICC to the midpoint between the wheels.

There are three interesting cases with these kinds of drives.

1. If  $V_l = V_r$ , then we have forward linear motion in a straight line. R becomes infinite, and there is effectively no rotation -  $\omega$  is zero.

2. If  $V_l = -V_r$ , then  $R = 0$ , and we have rotation about the midpoint of the wheel axis - we rotate in place.

3. If  $V_l = 0$ , then we have rotation about the left wheel. In this case  $R = l$ . same is true if

$V_r = 0$ , but in this case it rotates about right wheel.

Note that a differential drive robot cannot move in the direction along the axis - this is a singularity.

Differential drive vehicles are very sensitive to slight changes in velocity in each of the wheels. Small errors in the relative velocities between the wheels can affect the robot trajectory. They are also very sensitive to small variations in the ground plane, and may need extra wheels (castor wheels) for support.

## 2.5 Circuit Working

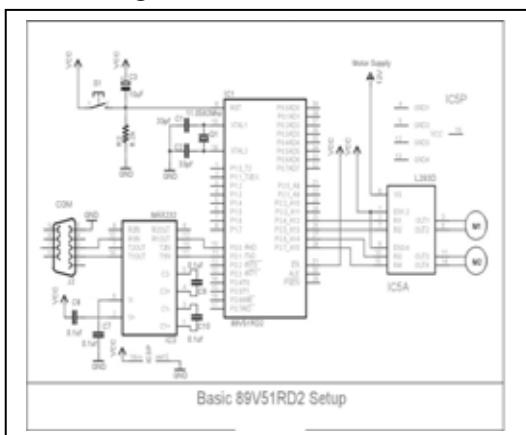


Figure 2 Circuit Diagram

The 18 and the 19 pin these are the XTAL pins. The p89v51rd2 microcontroller has an on chip oscillator but in order to run it we require an external clock which is made by two 30 Pico farad capacitors and a quartz crystal ,the type of crystal decides the speed. As p89v51rd2 is a microcontroller

which can work at frequency ranging from (0-40 MHz) so we have used 11.0592 MHz crystal

## 2.6 Time Period Calculation

We take the example of the reset switch which we have connected to 9 and 31 pins with a 10 micro farad capacitor in between it and the other end of the capacitor grounded .the purpose of reset switch is that it terminates the entire program and reset the microcontroller .for this to be done it requires two machine cycles for the reset switch to work the signal should be high to it for a period of two machine cycles ,the time in which the capacitor gets charged.

Time period to Access an INSTRUCTION –

For p89V51RD2

Clocks per machine cycle=12,

Crystal used =11.0592 MHZ ,

Time = 1/11.0592=90.42 ns,

Time for 12 clocks=90.42NS\*12=1.085μs

So we can make a simple conclusion that for reset switch it takes (1.085\*2)  $\mu$ s to become high. Machine cycle calculation for various instructions will be different, in this paper we are concentrating on machine cycle for accessing an instruction (like giving high pulse to reset pin).

## 2.7 Program Flow

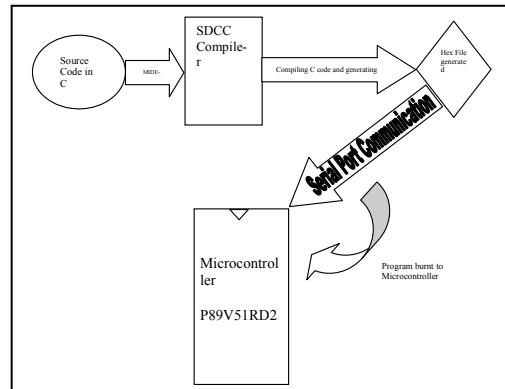


Figure 3 Program Flow

- The program for the line tracer is initially written in C language.
- It was compiled using SDCC (Small Device C Compiler).
- The interfacing was done by MIDE-51 which is quite popular and a freely available tool.
- The HEX file thus generated was written to the controller using Serial Port as medium of communication.
- The Baud Rate and COM port number were determined by a tool called FLASH MAGIC which burns the Hex file to the controller thus completing the process of Burning.

## 2.8 Benefits Of Serial Port Mode

1. Fewer number of cables were required as compared to parallel port mode. Moving the robot around with the cables was easier.
2. Mode of burning was easier. Defining the port value was not required.
3. Constructing the MAX232 circuit for serial port support was quite easy.

### III. OVERALL WORKING

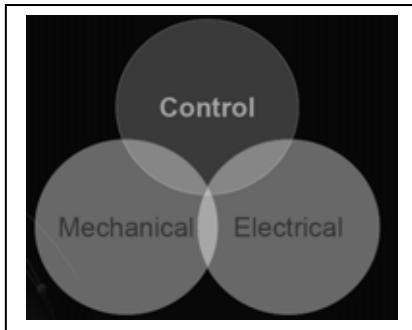


Figure 4 Venn diagram Representation

The Venn diagram just gives a representation of how the various aspects involved in the line follower are connected. We just want to convey the message that all components used in the robot are interrelated. For example, if the mechanical part like motors fails, then the overall robot will not work. Similarly coordination between these three aspects mentioned above causes a line follower to function perfectly.

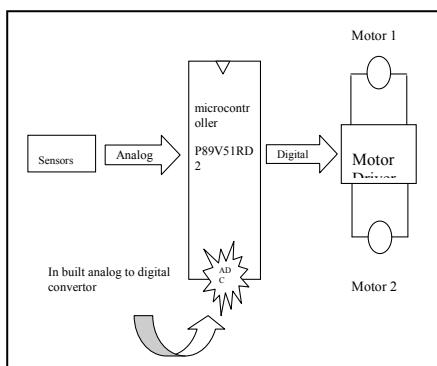


Figure 5 Block Diagram

Block Diagram shows overall working. How a signal travelling from the sensors in analog form and going through the various components ultimately reaches the motors attached to the wheels is being shown in the figure.

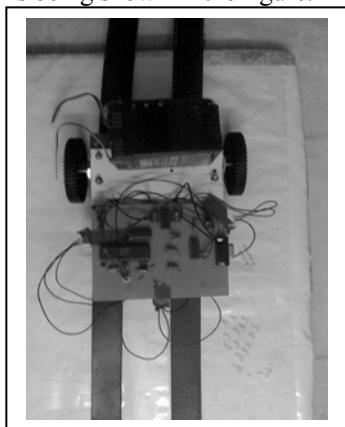


Figure 6 The autonomous microcontroller robot in the Testing phases

### IV. FUTURE ENHANCEMENTS

- General improvements like using a low dropout voltage regulator, lighter chassis etc.
- Using a quad comparator for better sensitivity of sensors like LM324, LM329.
- Using a wheel encoders with motors which gives better synchronization.
- Using Pulse Width Modulation technique.
- **Increasing the range of a sensor.** To fulfill this purpose we need to send huge amount of instantaneous currents through our infrared led for a short duration of time and allowing it to cool for a longer duration of time. This is a delicate task, as we need to send pulses of IR instead of constant IR emission. The duty cycle of the pulses turning the LED ON and OFF have to be calculated with precision, so that the average current flowing into the LED never exceeds the LED's maximum DC current (or 10mA as a standard safe value). The duty cycle is the ratio between the ON duration of the pulse and the total period. A low duty cycle will enable us to inject in the LED high instantaneous currents while shutting it OFF for enough time to cool down from the previous cycle.

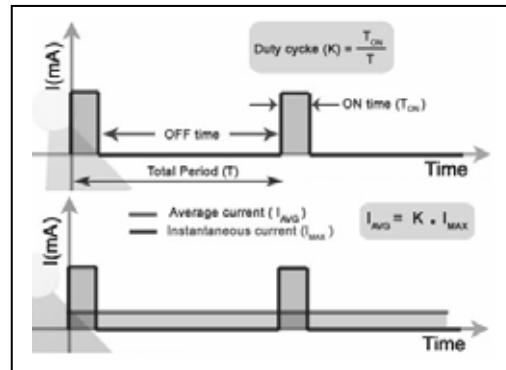


Figure 7 Duty Cycle Graphs

The 2 graphs shows the meaning of the duty cycle, and the mathematical relations between the ON time, the Total period, and the average current.

In the second graph, the average current in blue is exaggerated to be visible, but real calculations would yield a much smaller average current.

Having a look on the graph , the low duty cycle pulses will be produced. For this purpose we have an LM555 timer IC which helps in producing the pulse.

- To prevent the false triggering of the IR sensor from ambient light, the IR LED is placed at the top of the PCB and the photodiode or the receiver is placed just below it below the PCB. By this we conclude that the photodiode wont receive all other sources of light because all ambient sources come from the top. The distance between the OP-AMP and the photodiode should be 12mm to 35mm, thus facilitating a smaller circuit with higher precision.

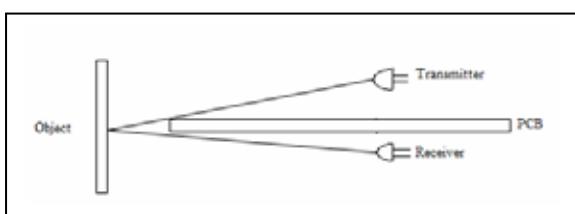


Figure 8 Component Positioning

- Improvements can be made that IR LEDs can be made modulated so that they won't be susceptible to ambient lighting and reflectivity of the objects .we make this by making the IR LED flash light at a particular frequency. Making same for the receiver making it to work at the same specific frequency, so indirectly transmitter and the receiver are nothing but modulator and demodulator. For example we generate 40 kHz frequency pulses with the help of NAND gates oscillator without using continuous beam this pulse is fed to the emitter and the it sends the same pulse which after sensing the object is made to pass through a band pass filter which allows only the 40 kHz pulse to reach the receiver filtering all other light ambient light , halogen ,sun thus preventing it from false triggering without using a band pass filter we can use same 40 kHz IR receiver module.

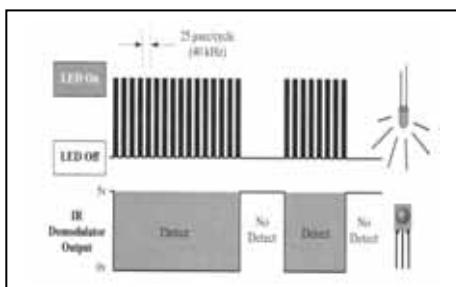


Figure 9 Modulations and Demodulation

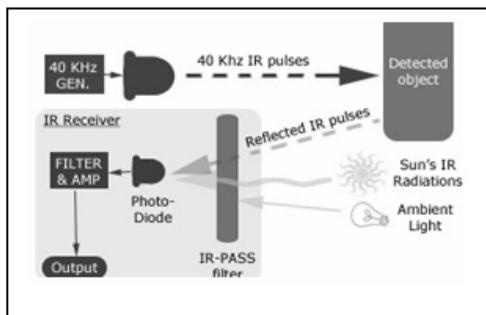


Figure 10 Transmitting, Filtering & Receiving

## V. REFERENCES

- [1]. "Robots Enhance Engineering Education". David J. Mehrl, Micheal E. Parten, Darrell L. Vines. 0-7803-4086-8 0/1997 IEEE
- [2]. "Fuzzy Logic Controlled Miniature LEGO Robot for Undergraduate Training System". N. Z. Azlan1, F. Zainudin2, H. M. Yusuf3, S. F. Toha4, S. Z. S. Yusoff5, N. H. Osman6. 1-4244-0737-0/07/\$20.00 c\_2007 IEEE
- [3]. "A Project-based Laboratory for Learning Embedded System Designs with Support from the Industry". Chyi-Shyong Lee, Juing-Huei Su, Kuo-En Lin, Jia-Hao Chang and Gu-Hong Lin. 978-1-4244-1970-8/08/\$25.00 ©2008 IEEE 38th ASEE/IEEE Frontiers in Education Conference
- [4]. "A SISO STRATEGY TO CONTROL AN AGV" R. Corteletti', P. R. Barros2 and A.M.N. Lima2 0-7803-9484-4/05/\$20.00 ©2005 IEEE
- [5]. CS W4733 NOTES - Differential Drive Robots. compiled from Dudek and Jenkin, *Computational Principles of Mobile Robotics*.
- [6]. "Evolving a Vision-Based Line-Following Robot Controller", Jean-Francois Dupuis and Marc Parizeau. Proceedings of the 3rd Canadian Conference on Computer and Robot Vision (CRV'06)0-7695-2542-3/06 \$20.00 © 2006 IEEE

# K-L Partitioning Algorithm Implementation in MATLAB

Lecturer- S. Supreet Singh, Prof.- S.S. Gill, Lecturer -Er. J.P.S. Raina  
 Baba Banda Singh bahadur Engineering College, Fatehgarh Sahib, Sirhind Punjab,  
 #Guru Nanak Dev Engineering College, Ludhiana, Punjab, e-mail: [supreet.e@gmail.com](mailto:supreet.e@gmail.com)

**Abstract—** Circuit partitioning is the one of the fundamental problems in VLSI design. It appears in several stages in VLSI design, such as logic design and physical design. Circuit partitioning is generally formulated as the graph partitioning problem. For this problem, a heuristic proposed by Kernighan and Lin is the most well-known and widely used one in practical applications. However, due to recent advances of semiconductor technologies, a VLSI chip contains millions of transistors, and hence the size of the problem of circuit partitioning also becomes very large. Good partitioning techniques can positively influence the performance and cost of a VLSI product.

The main objective to Partition a circuit into parts is that every component is within a prescribed range and the # of connections among the components is minimized.

The K-L (Kernighan-Lin) algorithm was first suggested in 1970 for bisecting graphs in relation to VLSI layout. It is an iterative algorithm. Starting from a load balanced initial bisection, it first calculates for each vertex the gain in the reduction of edge-cut that may result if that vertex is moved from one partition of the graph to the other. At the each inner iteration, it moves the unlocked vertex which has the highest gain, from the partition in surplus (that is, the partition with more vertices) to the partition in deficit. This vertex is then locked and the gains updated. The procedure is repeated even if the highest gain may be negative, until all of the vertices are locked.

MATLAB software is used for programming the User Interface for placing the components, making the net connections and defining the initial partition. The Program then applies K-L Algorithm to the problem by first, finding the Interfacing Matrix, (also called Adjacency Matrix) and then, calculating the gain at each node. The final partitions are made such that the number of cuts across the partition is minimized and program then calculates the final cut-size.

## I. INTRODUCTION

Partitioning is a technique to divide a circuit or system into a collection of smaller parts (components). It is on the one hand a design task to break a large system into pieces to be implemented on separate interacting components and on the other hand it serves as an algorithmic method to solve difficult and complex combinatorial optimization problems as in logic or layout synthesis.

The VLSI designs have increased to systems of hundreds and millions of transistors. The complexity of the circuit has become so high that it is very difficult to design and simulate the whole system without decomposing it into sets of smaller subsystems. This divide and conquer strategy relies on partitioning to manipulate the whole system into hierarchical tree structure.

## II. PARTITIONING – AN REQUISITE IN VLSI

Widely accepted powerful high-level synthesis tools allow the designers to automatically generate huge systems. Synthesis and simulation tools often cannot cope with the

complexity of the entire system under development. Thus, the present state of design technology often requires a partitioning of the system.

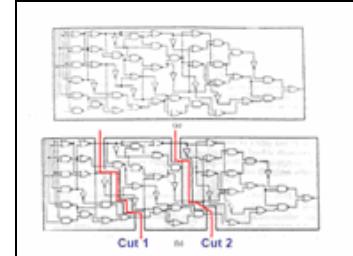


Figure 1: Circuit Partitioning

Typically, the partitioning problem involves the following:

- Decomposition of a complex system into smaller subsystems.
- Each subsystem to be designed independently.
- Decomposition scheme to minimize the Interconnections between the subsystems.
- Decomposition to be carried out hierarchically until each subsystem is of manageable size.

## III. K-L PARTITIONING ALGORITHM

The K-L (Kernighan-Lin) algorithm was first suggested in 1970 for bisecting graphs in relation to VLSI layout. It is an iterative algorithm. Starting from a load balanced initial bisection, it first calculates for each vertex the gain in the reduction of edge-cut that may result if that vertex is moved from one partition of the graph to the other. At the each inner iteration, it moves the unlocked vertex which has the highest gain, from the partition in surplus (that is, the partition with more vertices) to the partition in deficit. This vertex is then locked and the gains updated. The procedure is repeated even if the highest gain may be negative, until all of the vertices are locked. The last few moves that had negative gains are then undone and the bisection is reverted to the one with the smallest edge-cut so far in this iteration. This completes the outer one iteration of the K-L algorithm and the iterative procedure is restarted. Should an outer iteration fail to result in any reductions in the edge-cut or load imbalance, the algorithm is terminated. The initial bisection is generated randomly and for large graphs, the final result is very dependent on the initial choice. The K-L algorithm is a local optimization algorithm. The different steps involved are:

- An iterative, 2-way, balanced partitioning (bi-sectioning) heuristic.
- Till the cut size keeps decreasing
- Vertex pairs which give the largest decrease or smallest increase in cut size are exchanged.

- These vertices are then locked (and thus are prohibited from participating in any further exchanges).
- This process continues until all the vertices are locked.
- Find the set with the largest partial sum for swapping.
- Unlock all vertices.

It is assumed that each edge has a unit weight.

#### IV. IMPLEMENTATION INTERFACE ALGORITHM

- Step 1:** Place The Components (Nodes) On The Workplace.  
**Step 2:** Make The Interconnections (Nets) Between The Components Placed In Step 1.  
**Step 3:** Draw The Partitioning Line.  
**Step 4:** Press Enter To Apply The K-L Algorithm On The Circuit.  
**Step 5:** The Program Calculates The Initial Cut-Size.  
**Step 6:** The Program Makes Resulting Partitions (Partition A & Partition B) such That The Partitions Have The Least Interconnections Or Cut Size.  
**Step 7:** The Program Calculates The Final Cut-Size.

#### V. MATLAB CODING FOR KL ALGORITHM

##### IMPLEMENTATION

```

close all;
clear all;

axis ([0 30 0 30])
hold on
grid on;
% initially, the list of points is empty.
xy = [ ];
n = 0;
% Loop, picking up the points.
disp ('Left mouse button picks points.')
disp ('Right mouse button picks last point.')
but = 1;
xi = 0;
yi = 0;
title ('Place the components over the Grid and Press Esc,
when finished.. ', 'FontSize', 18,...
'FontName', 'Times New Roman','Margin',5);
while but == 1
    [xi, yi, but] = ginput (1);
    xi = round (xi);
    yi = round (yi);
    if (but ==1)
        plot (xi,yi,'ro','LineWidth',2)
        n = n+1;
        text (xi+0.2,yi+0.2,num2str(n));
        xy (:,n) = [xi;yi];
    end
end

% Initialize the Interfacing Matrix to Zero
for i = 1:1:n
    for j = 1:1:n
        interfacing_mat (i, j) = 0;
    end
end

```

```

title ({'Placing Components finished ... Now Do
Interconnections ';'Press Esc When Finished'},
'FontSize', 18,...,
'FontName', 'Times New Roman','Margin', 5);
refresh;

% This is the main loop for interconnections
while (1)
    [xi, yi, but] = ginput (1);
% Initial Point for Interconnection line
    if (but == 1)
        xi = round(xi);
        yi = round(yi);
        for i = 1:n
            if ( xy(1,i) == xi && xy(2,i) == yi)
                xyvector (1, 1) = xi;
                xyvector (2, 1) = yi;
                pt_variable =1;
                break;
            end
        end
    else
        break;
    end

% xyvector stores the initial and final point of the line to be
drawn
% xyvector = | intial_x_point  final_x_point |
%           | intial_y_point  final_y_point |
%
[xi, yi, but] = ginput (1);
% Final Point for Interconnection line
if (but == 1)
    xi = round (xi);
    yi = round (yi);
    for j = 1 : n
        if( xy(1,j) == xi && xy(2,j) == yi)
            xyvector (1,2) = xi;
            xyvector (2,2) = yi ;
            break;
        end
    end
else
    break;
end

line (xyvector (1, :), xyvector(2,:));
interfacing_mat (i, j) =1;
interfacing_mat (j, i) =1;
end %end of while

% Draw the partitioning line
title ('Draw the partitioning line ...', 'FontSize', 18,...
'FontName', 'Times New Roman','Margin',5);
while (1)
    [xi, yi, but] = ginput (1);
    if (but == 1)
        xi = round (xi);
        yi = round (yi);

```

```

xyvector (1, 1) = xi;
xyvector (2, 1) = yi;
break;
else
    continue
end
end

% xyvector stores the initial and final point of the line to be
drawn
% xyvector = | intial_x_point  final_x_point |
%           | intial_y_point  final_y_point |

while (1)
    [xi, yi, but] = ginput (1);

    if (but == 1)
        xi = round (xi);
        yi = round (yi);
        xyvector (1,2) = xi;
        xyvector (2,2) = yi ;
        break;
    else
        continue;
    end
end

plot (xyvector (1, :), xyvector (2, :), 'r') ;
% finding the equation of the line
% Slope of the line

m = (xyvector (2, 2) - xyvector (2, 1)) / (xyvector (1, 2) -
    xyvector (1, 1));
c = xyvector (2, 1) - (m * xyvector (1, 1));

X = [0 (-1*c)/m]
Y = [c 0]
plot (X,Y, 'r');

% xy Matrix xy (1, i) x-vector and xy (2, i) y-vector
% Partition A and Partition B
B = 0;
A = 0;
for i = 1:n
    Q = xy (2, i) - m*xy (1, i) - c;
    if Q > 0
        A = [A i];
    else
        B = [B i];
    end
end
%Q = y - mx -c;
for i = 2: length (A)
    A1 (i-1) = A (i);
end
A = A1
for i = 2:length (B)
    B1 (i-1) = B (i);
end

```

```

B = B1

interfacing_mat
nodes =n;
title ('Press Any key to Continue ...', 'FontSize', 18,... 
    'FontName', 'Times New Roman','Margin', 5);

initialcutsize =0;
for i = 1: length (A)
    for j = 1: length(B)
        if interfacing_mat (A(i),B(j))==1
            initialcutsize = initialcutsize +1;
        end
    end
end
if (Initial cut size = ' );
    disp (initialcutsize);

xlabel ([‘Initial Cut Size = ‘ num2str (initialcutsize)],'FontSize',
18,... 
    'FontName', 'Times New Roman','Margin', 5);

pause

% K-L Algorithm is going to be implemented
%*****
c0 = interfacing_mat;
parA = A;
parB =B;
disp ('c0 = ');
disp (c0);
%*****


count=0;
tmp=1;
if n==size(c0,1), % c0 is a square, symmetric n x n matrix
    error ('invalid connectivity matrix');
end

iter = 1;
done = 0;
temp = 1;
x = 0;
idxA = parA;
idxB = parB;
while temp>0
    c=c0 ([parA parB], [parA parB]);
    % permute c0 to fit initial partition
    disp ('c = ');
    disp (c);
    n=size (c0, 1);
    if rem(n,2)~=0
        n1 = floor (n/2);
        n2 = n1+1;
    else
        n1=n/2; n2=n1;
    end

```

```

while ~done, % while not yet done,
disp (['*** Iteration ' int2str(iter) ' ***']);
if iter==1,
    disp ('intcost = ');
    intcost = [sum(c(1:n1,1:n1)) sum(c(n1+1:n,n1+1:n))]

% compute external cost, 1 x n
extcost =[sum(c(1:n1,n1+1:n)) sum(c(n1+1:n,1:n1))]

% compute D values, 1 x n
Dvalue=extcost-intcost;
% first n1 is partition A, next n2 is partition B
    disp (['Initial partition cost = ' int2str (sum (extcost
(1:n1)))]);

end % otherwise, the D values will be updated later.

gmat =
Dvalue(1:n1)*ones(1,n2)+ones(n1,1)*Dvalue(n1+1:n)...
-2*c(1:n1,n1+1:n);
[mt mp,id1]= max (gmat);
% find max g of each column of gmat matrix
[g(iter),id2]=max (mtmp);
% g(iter) is max g of gmat matrix
ida=id1(id2); idb=n1+id2;
% c matrix indices of two exchange nodes
% that yield maximum g (gain) in reducing cut set cost.
disp ('The g matrix is:')
disp (gmat);
% now convert into node indices for the two selected nodes
bidx (iter) = idxB (id2); % node index for node b
aidx (iter) = idxA (ida); % node index for node a
disp (['iter = ' int2str (iter) ', nodes to be exchanged: ' ...
    int2str (aidx (iter)) ', and ' int2str(bidx (iter)) ' Max.
    g = ' num2str (g (iter))]);
x=x + g (iter);

if size(c,1)==2, done=1; % stop the next iteration, done
else
    iter = iter+1; % move to the next iteration
    idxA = setdiff (idxA, aidx); % A-{a}
    idxB = setdiff (idxB, bidx); % B-{b}

% next, remove the ida, idb rows and columns of the c
matrix
idv = [setdiff ([1:n],[ida idb]) [ida idb]];
% permute the indices of c
c1 = c (idv,idv);
% move ida, idb rows and columns to the last two rows
Dvalue=Dvalue (idv (1:n-2));
% move the Dvalue of a and b to last two entries
Dvalue (1:n1-1)=Dvalue (1:n1-1)+2*c1 (n-1, 1:n1-1)-
2*c1 (n, 1:n1-1);
Dvalue(n1:n-2)=Dvalue(n1:n-2)+2*c1(n,n1:n-2)-2*c1(n-
1,n1:n-2);
% Dvalue is reduced length by 2
disp ('Updated D values are:');
disp (Dvalue)
c= c1 (1: n-2, 1: n-2);

```

```

n = size(c, 1); n1=n/2; n2=n1; % update c matrix indices
disp ('Press any key to continue to the next iteration ...')
title ('Applying the KL Algorithm...Press any key to go
for Next Iteration.. ','...')
'FontSize',18, 'FontName', 'Times New
Roman','Margin',5);
end % if not terminate, update D'
end % of while loop

[tmp,K]=max (sum(triu(toeplitz(g))));
disp ('Decision ...');
disp (['Exchange first ' int2str(K) ' pairs of nodes']);
disp ('Final partition = ');
disp (['Partition A: ' int2str(union (setdiff(parA, aidx
(1:K)),bidx(1:K)))]);
disp (['Partition B: ' int2str(union (setdiff(parB, bidx
(1:K)),aidx(1:K)))]);
count =count+1;
iter = 1; % iteration count
done = 0; % Boolean condition on whether iteration is done
parA = ( union(setdiff (parA, aidx(1:K)),bidx(1:K)));
parB = ( union(setdiff (parB, bidx(1:K)),aidx(1:K)));
idxA = parA;
idxB = parB;
bidx = 0;
aidx = 0;
temp = x;
end

disp ('See The final Figure for the Resulting Partition ...');
title ('Partitioning Finished.. Press Any key to see the
Resulting Partition...', 'FontSize', 18, ...
'FontName', 'Times New Roman','Margin',5);
pause;

for i =1: length (parA)
    temp_x = xy (1, parA (i));
    temp_y = xy (2, parA (i));
    plot (temp_x, temp_y, 'ks','LineWidth', 2,
    'MarkerEdgeColor','k',...
    'MarkerFaceColor','g','MarkerSize', 10)
end
for i =1: length(parB)
    temp_x = xy (1, parB (i));
    temp_y = xy (2, parB (i));
    plot (temp_x, temp_y,
    'ks','LineWidth',2,'MarkerEdgeColor','k',...
    'MarkerFaceColor', 'r', 'MarkerSize', 10)
end

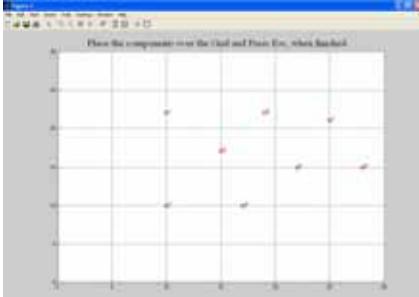
finalcutsize =0;
for i = 1: length (parA)
    for j = 1:length (parB)
        if interfacing_mat (parA (i), parB (j)) ==1
            finalcutsize = finalcutsize +1;
        end
    end
disp ('final cut size = ');
disp (finalcutsize);

```

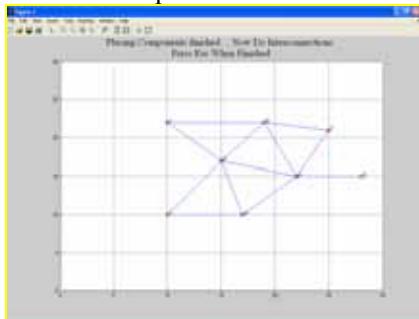
```
xlabel ([ 'Final Cut Size = ' num2str
    (finalcutsize)],'FontSize', 18,...
    'FontName', 'Times New Roman','Margin',5);
```

## VI. OUTPUT- FINAL RESULTS

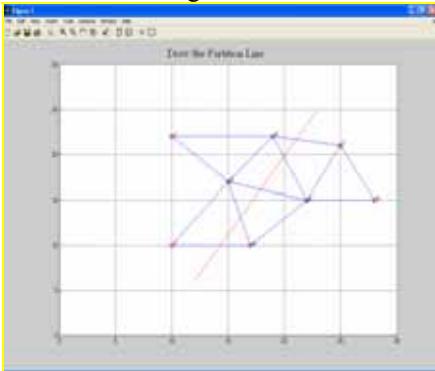
Step 1: Placing the Components (Nodes) Over the Workplace.



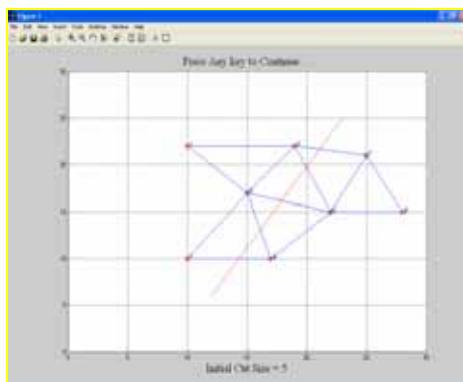
Step 2: Making The Interconnections (Nets) Between The Components Placed In Step 1.



Step 3: Draw The Partitioning Line.

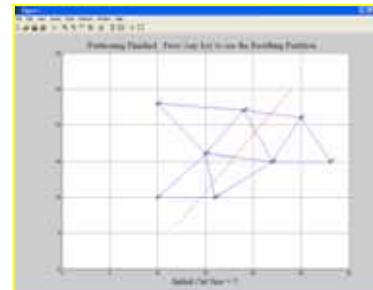


Step 4: Press Enter To Apply The K-L Algorithm On The Circuit.

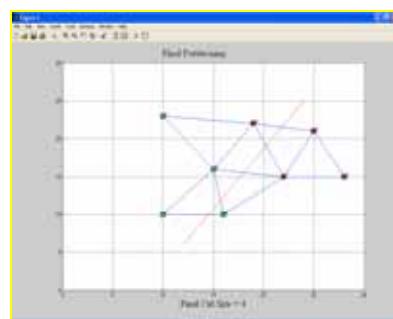


Step 5: The Program Calculates The Initial Cut-Size. Initial Cut Size (For the Circuit Example) = 5

Step 6: The Program Makes Resulting Partitions (Partition A & Partition B) Such That The Partitions Have The Least Interconnections Or Cut Size.



Step 7: The Program Calculates The Final Cut-Size.



## VII. REFERENCES

- [1]. Kernighan and Lin, "An efficient heuristic procedure for partitioning graphs," *The Bell System Technical Journal*, vol. 49, no. 2, Feb. 1970.
- [2]. Tutorial on VLSI Partitioning By:SAO-JIE CHEN(a) and CHUNG-KUAN CHENG(b) , (a) Dept. of Electrical Engineering, National Taiwan University, Taipei, Taiwan 10764; (b) Dept. of Computer Science and Engineering, University of California, "San Diego, La Jolla, CA 92093-0114 (Received March 1999," In finalform 10 February 2000)
- [3]. Bernhard M. Ries, Konrad Doll, and Frank M. Johannes. Partitioning every large circuit using analytical placement techniques. In *Design Automation Conference (DAC)*, pages 646-651. ACM/IEEE, 1994.B. Smith, "An approach to graphs of linear forms (Unpublished work style)," unpublished.
- [4]. Bernhard M. Ries, Heiko A. Giselbrecht, and Bemd Wurth. A new k-way partitioning approach for multiple types of FPGAs. In *Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 3 1 3 3 18. IFIP/ACM/IEEE, 1995.
- [5]. Ren-Song Tsay and Ernest Kuh. A unified approach to partitioning and placement. *Transactions on Circuits and Systems*, 38:521-533, 1991.
- [6]. Hirendu Vaishnav and Massoud Pedram. Delay optimal partitioning targeting low power VLSI circuits. In *International Conference on Computer Aided Design (ICCAD)*, pages 638- 643. IEEE IACM, 1995.
- [7]. By Jens Lienig, Springer Verlag, Berlin Heidelberg Introductory Lectures in VLSI Physical Design Algorithms New York, 2006.
- [8]. VLSI Physical System Design and Automation – Theory and Practical Sadiq M Sait , Habib Youssef.
- [9]. By Moses Charikar, Konstantin Makarychev, Yury Makarychev Symposium on discrete Algorithms Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm Miami, Florida 2006 Pg: 51 – 60.
- [10]. By:Yao-Ping Chen, Ting-Chi Wang and D. F. Wong "A Graph Partitioning Problem for Multiple-Chip Design" Department of Computer Sciences University of Texas at Austin, Texas 78712

# New Ways of Improving Life using Artificial Intelligence: Fuzzy Logic

Ms. Pooja Dhiman, Lecturer \*Ms. Arti Dhiman, Student M.Tech  
CIET, Rajpura, \*JMIT, Radaur. E-mail: [poojadhiman23@gmail.com](mailto:poojadhiman23@gmail.com)

**Abstract-** Fuzzy Logic is a problem-solving control system methodology that lends itself to implementation in systems ranging from simple, small, embedded micro-controllers to large, networked, multi-channel PC or workstation-based data acquisition and control systems. It can be implemented in hardware, software, or a combination of both. FL provides a simple way to arrive at a definite conclusion based upon vague, ambiguous, imprecise, noisy, or missing input information.

Fuzzy logic has rapidly become one of the most successful of today's technologies for developing sophisticated control systems. The reason for which is very simple. Fuzzy logic addresses such applications perfectly as it resembles human decision making with an ability to generate precise solutions from certain or approximate information. It fills an important gap in engineering design methods left vacant by purely mathematical approaches , and purely logic-based approaches in system design. While other approaches require accurate equations to model real-world behaviors, fuzzy design can accommodate the ambiguities of real-world human language and logic. It provides both an intuitive method for describing systems in human terms and automates the conversion of those system specifications into effective models.

## I. WHAT DOES IT OFFER?

The first applications of fuzzy theory were primly industrial, such as process control for cement kilns. However, as the technology was further embraced, fuzzy logic was used in more useful applications. In 1987, the first fuzzy logic-controlled subway was opened in Sendai in northern Japan. Here, fuzzy-logic controllers make subway journeys more comfortable with smooth braking and acceleration. Best of all, all the driver has to do is push the start button! Fuzzy logic was also put to work in elevators to reduce waiting time. Since then, the applications of Fuzzy Logic technology have virtually exploded, affecting things we use everyday. Take for example, the *fuzzy washing machine* . A load of clothes in it and press start, and the machine begins to churn, automatically choosing the best cycle. The fuzzy microwave, Place chili, potatoes, or etc in a *fuzzy microwave* and push single button, and it cooks for the right time at the proper temperature. The *fuzzy car*, manuvers itself by following simple verbal instructions from its driver. It can even stop itself when there is an obstacle immediately ahead using sensors. But, practically the most exciting thing about it, is the simplicity involved in operating it.

## II. WHAT DO YOU MEAN BY FUZZY??!!

Before illustrating the mechanisms which make fuzzy logic machines work, it is important to realize what fuzzy logic actually is. Fuzzy logic is a superset of conventional (Boolean) logic that has been extended to handle the concept of partial truth- truth values between "completely true" and

"completely false". As its name suggests, it is the logic underlying modes of reasoning which are approximate rather than exact. The importance of fuzzy logic derives from the fact that most modes of human reasoning and especially common sense reasoning are approximate in nature. The essential characteristics of fuzzy logic as founded by Zadeh Lotfi are as follows.

- In fuzzy logic, exact reasoning is viewed as a limiting case of approximate reasoning.
- In fuzzy logic everything is a matter of degree.
- Any logical system can be fuzzified
- In fuzzy logic, knowledge is interpreted as a collection of elastic or, equivalently , fuzzy constraint on a collection of variables
- Inference is viewed as a process of propagation of elastic constraints.

The third statement hence, define Boolean logic as a subset of Fuzzy logic.

## III. FUZZY SETS

Fuzzy Set Theory was formalized by Professor Lofti Zadeh at the University of California in 1965. What Zadeh proposed is very much a paradigm shift that first gained acceptance in the Far East and its successful application has ensured its adoption around the world.

A paradigm is a set of rules and regulations which defines boundaries and tells us what to do to be successful in solving problems within these boundaries. For example the use of transistors instead of vacuum tubes is a paradigm shift - likewise the development of Fuzzy Set Theory from conventional bivalent set theory is a paradigm shift. Bivalent Set Theory can be somewhat limiting if we wish to describe a 'humanistic' problem mathematically. For example, Fig 1 below illustrates bivalent sets to characterize the temperature of a room.



The most obvious limiting feature of bivalent sets that can be seen clearly from the diagram is that they are mutually exclusive - it is not possible to have membership of more than one set ( opinion would widely vary as to whether 50 degrees Fahrenheit is 'cold' or 'cool' hence the expert knowledge we

need to define our system is mathematically at odds with the humanistic world). Clearly, it is not accurate to define a transition from a quantity such as 'warm' to 'hot' by the application of one degree Fahrenheit of heat. In the real world a smooth (unnoticeable) drift from warm to hot would occur.

This natural phenomenon can be described more accurately by Fuzzy Set Theory. Fig.2 below shows how fuzzy sets quantifying the same information can describe this natural drift.

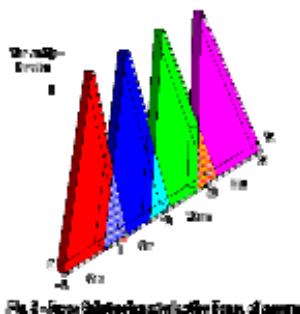


Fig.2 - Fuzzy Sets Quantifying the Same Information

The whole concept can be illustrated with this example. Let's talk about people and "youthness". In this case the set S (the universe of discourse) is the set of people. A fuzzy subset YOUNG is also defined, which answers the question "to what degree is person x young?" To each person in the universe of discourse, we have to assign a degree of membership in the fuzzy subset YOUNG. The easiest way to do this is with a membership function based on the person's age.

$$\text{young}(x) = \begin{cases} 1, & \text{if } \text{age}(x) \leq 20, \\ (30-\text{age}(x))/10, & \text{if } 20 < \text{age}(x) \leq 30, \\ 0, & \text{if } \text{age}(x) > 30 \end{cases}$$

A graph of this looks like:



Given this definition, here are some example values:

Person Age degree of youth

Johan	10	1.00
Edwin	21	0.90
Parthiban	25	0.50
Arosha	26	0.40
Chin Wei	28	0.20
Rajkumar	83	0.00

So given this definition, we'd say that the degree of truth of the statement "Parthiban is YOUNG" is 0.50.

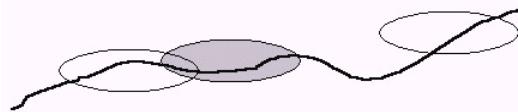
Note: Membership functions almost never have as simple a shape as age(x). They will at least tend to be triangles pointing up, and they can be much more complex than that. Furthermore, membership functions so far is discussed as if they always are based on a single criterion, but this isn't

always the case, although it is the most common case. One could, for example, want to have the membership function for YOUNG depend on both a person's age and their height (Arosha's short for his age). This is perfectly legitimate, and occasionally used in practice. It's referred to as a two-dimensional membership function. It's also possible to have even more criteria, or to have the membership function depend on elements from two completely different universes of discourse.

#### IV. FUZZY RULES

Human beings make decisions based on rules. Although, we may not be aware of it, all the decisions we make are all based on computer like if-then statements. If the weather is fine, then we may decide to go out. If the forecast says the weather will be bad today, but fine tomorrow, then we make a decision not to go today, and postpone it till tomorrow. Rules associate ideas and relate one event to another. Fuzzy machines, which always tend to mimic the behavior of man, work the same way. However, the decision and the means of choosing that decision are replaced by fuzzy sets and the rules are replaced by fuzzy rules. Fuzzy rules also operate using a series of if-then statements. For instance, if X then A, if y then b, where A and B are all sets of X and Y. Fuzzy rules define fuzzy patches, which is the key idea in fuzzy logic.

A machine is made smarter using a concept designed by Bart Kosko called the Fuzzy Approximation Theorem(FAT). The FAT theorem generally states a finite number of patches can cover a curve as seen in the figure below. If the patches are large, then the rules are sloppy. If the patches are small then the rules are fine.



#### V. FUZZY PATCHES

In a fuzzy system this simply means that all our rules can be seen as patches and the input and output of the machine can be associated together using these patches. Graphically, if the rule patches shrink, our fuzzy subset triangles gets narrower. Simple enough? Yes, because even novices can build control systems that beat the best math models of control theory. Naturally, it is *math-free* system.

#### VII. FUZZY TRAFFIC LIGHT CONTROLLER



This part of the paper describes the design procedures of a real life application of fuzzy logic: A Smart Traffic Light Controller. The controller is suppose to change the cycle

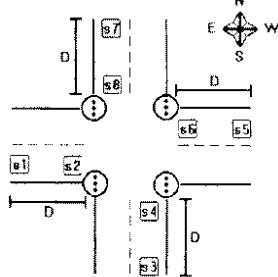
time depending upon the densities of cars behind green and red lights and the current cycle time.

## VI. BACKGROUND

In a conventional traffic light controller, the lights change at constant cycle time, which is clearly not the optimal solution. It would be more feasible to pass more cars at the green interval if there are fewer cars waiting behind the red lights. Obviously, a mathematical model for this decision is enormously difficult to find. However, with fuzzy logic, it is relatively much easier.

## VII. FUZZY DESIGN

First, eight incremental sensors are put in specific positions as seen in the diagram below.



The first sensor behind each traffic light counts the number cars coming to the intersection and the second counts the cars passing the traffic lights. The amount of cars between the traffic lights is determined by the difference of the reading of the two sensors. For example, the number of cars behind traffic light North is  $s_7 - s_8$ .

The distance D, chosen to be 200ft., is used to determine the maximum density of cars allowed to wait in a very crowded situation. This is done by adding the number of cars between two paths and dividing it by the total distance. For instance, the number of cars between the East and West street is  $(s_1 - s_2) + (s_5 - s_6)/400$ . Next comes the fuzzy decision process which uses the three step mentioned above(fuzzification, rule evaluation and defuzzification).

### Step 1

As before, firstly the inputs and outputs of the design has to be determined. Assuming red light is shown to both North and South streets and distance D is constant, the inputs of the model consist of :

- 1) Cycle Time
- 2) Cars behind red light
- 3) Cars behind green light

The cars behind the light is the maximum number of cars in the two directions. The corresponding output parameter is the probability of change of the current cycle time. Once this is done, the input and output parameters are divided into overlapping member functions, each function corresponding to different levels. For inputs one and two the levels and their corresponding ranges are zero(0,1), low(0,7), medium(4,11), high(7,18), and chaos(14,20). For input 3 , the levels are very short(0,14), short(0,34), medium(14,60), long(33,88), very long(65,100), limit(85,100). The levels of

output are no(0), probably no(0.25), maybe(0.5), probably yes (0.75), and yes(1.0). Note: For the output, one value (singleton position) is associated to each level instead of a range of values. The corresponding graphs for each of these membership functions are drawn in the similar way above.

### Step 2

The rules, as before are formulated using a series of if-then statements, combined with AND/OR operators. Ex: if cycle time is medium AND Cars Behind Red is low AND Cars Behind Green is medium, then change is Probably Not. With three inputs, each having 5,5, and 6 membership functions, there are a combination of 150 rules. However using the minimum or maximum criterion some rules are combined to a total of 86.

### Step 3

This process, also mentioned above converts the fuzzy set output to real crisp value. The method used for this system is *center of gravity*:

$$\text{Crisp Output} = \{\text{Sum}(\text{Membership Degree} * \text{Singleton Position})\} / (\text{Membership degree})$$

For example, if the output membership degree, after rule evaluation are:

Change Probability Yes=0, Change Probability Probably Yes=0.6, Change Probability Maybe=0.9, Change Probability Probably No= 0.3, Change Probability No=0.1 then the crisp value will be: Crisp Output=(0.1\*0.00)+(0.3\*0.25)+(0.9\*0.50)+(0.6\*0.75)+(0\*1.00)/0.1+0.3+0.9+0.6+0 =0.51

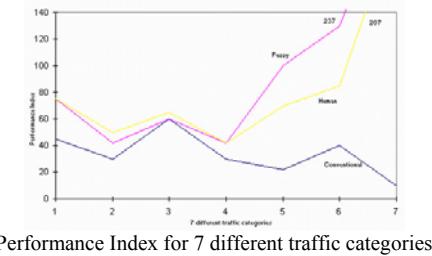
*Is Fuzzy Controller better?*

*Testing of the controller*

The fuzzy controller has been tested under seven different kinds of traffic conditions from very heavy traffic to very lean traffic. 35 random chosen car densities were grouped according to different periods of the day representing those traffic conditions.

## VIII. PERFORMANCE EVALUATION

The performance of the controller was compared with that of a conventional criteria used for comparison were number of cars allowed to pass at one time and average waiting time. A performance index which maximizes the traffic flow and reduces the average waiting time was developed. A means of calculating the average waiting time was also developed, however, a detailed calculation of this evaluation is beyond the scope of this article. All three traffic controller types were compared and can be summarized with the following graph of performance index in all seven traffic categories.



Performance Index for 7 different traffic categories

## IX. CONCLUSION

The fuzzy controller passed through 31% more cars, with an average waiting time shorter by 5% than the theoretical minimum of the conventional controller. The performance also measured 72% higher. This was expected. However, in comparison with a human expert the fuzzy controller passed through 14% more cars with 14% shorter waiting time and 36% higher performance index. In conclusion, as Man gets hungry in finding new ways of improving our way of life, new, smarter machines must be created. Fuzzy logic provides a simple and efficient way to meet these demands and the future of it is limitless.

## X. REFERENCES

- [1]. Zadeh "The birth and evolution of fuzzy logic" Journal Soft. Vol.2, No.1, 1990
- [2]. F. Kawaguchi, M. Miyakoshi, "A fuzzy rule interpolation technique based on bi-spline in multiple input systems" 2000, the ninth IEEE international conference on fuzzy systems, Vol.1, PP: 488-492.
- [3]. Van Schyndel, R. G., Tirkel, A. Z., Osborne C. F.; "A Digital Watermark"; Proc of ICIP 1994, Vol 2; pp. 688-90
- [4]. [www.wikipedia.org/fuzzylogics.html](http://www.wikipedia.org/fuzzylogics.html)
- [5]. [http://www.ics.uci.edu/\\_mlearn/MLRepository](http://www.ics.uci.edu/_mlearn/MLRepository)
- [6]. <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
- [7]. [http://www.doc.ic.ac.uk/~nd/surprise\\_96/journal/vol4/sbaa/report.html](http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/sbaa/report.html)
- [8]. [http://www.sciam.com/askexpert\\_question.cfm?articleID=000E9C72-536D-1C72-9EB7809EC588F2D7&catID=3](http://www.sciam.com/askexpert_question.cfm?articleID=000E9C72-536D-1C72-9EB7809EC588F2D7&catID=3)
- [9]. [www.doc.ic.ac.uk/~nd/surprise\\_96/journal/.../article2.html -](http://www.doc.ic.ac.uk/~nd/surprise_96/journal/.../article2.html)
- [10]. Daniel Mcneil and Paul Freiberger " Fuzzy Logic"
- [11]. <http://www.ortech-engr.com/fuzzy/reservoir.html>
- [12]. <http://www.quadralay.com/www/Fuzzy/FAQ/FAQ00.html>
- [13]. <http://www.fll.uni.linz.ac.at/pdhome.html>
- [14]. <http://soft.amcac.ac.jp/index-e.html>
- [15]. <http://www.abo.fi/~rfuller/nfs.html>

# Microcontroller Interfacing with Various Peripheral Electronic Devices

\*Senior Lecturer-Tejinder Singh, \*\*PG Scholar-Supriya Saxena  
 \*CIET, Rajpura, \*\*Thapar University, Patiala, India

**Abstract—** Circumstances that we find ourselves in today in the field of micro controllers had their beginnings in the development of technology of integrated circuits. This development has made it possible to store hundreds of thousands of transistors into one chip. That was a prerequisite for production of microprocessors, and adding external peripherals such as memory, input-output lines, timers and other made the first computers. Further increasing of the volume of the package resulted in creation of integrated circuits. These integrated circuits contained both processor and peripherals. That is how the first chip containing a microcomputer, or what would later be known as a micro controller came about.

## I. INTRODUCTION

In transforming an idea into a ready-made product, Frederico Faggin from company Intel in only 9 months had succeeded in making a product from its first conception. INTEL obtained the rights to sell this integral block in 1971. During that year, there appeared on the market a microprocessor called 4004. That was the first 4-bit microprocessor with the speed of 6 000 operations per second. Not long after that, American company CTC requested from INTEL and Texas Instruments to make an 8-bit microprocessor for use in terminals. Even though CTC gave up this idea in the end, Intel and Texas Instruments kept working on the microprocessor and in April of 1972, first 8-bit microprocessor appeared on the market under a name 8008. It was able to address 16Kb of memory, and it had 45 instructions and the speed of 300 000 operations per second. That microprocessor was the predecessor of all today's microprocessors. Intel kept their developments up in April of 1974, and they put on the market the 8-bit processor under a name 8080 which was able to address 64Kb of memory, and which had 75 instructions, and the price began at \$360.

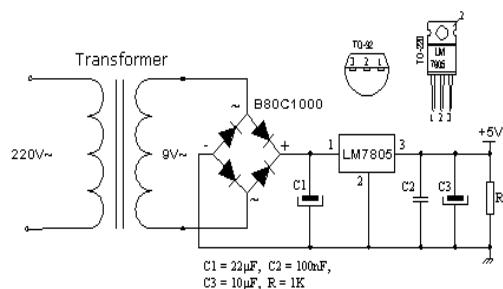
In another American company Motorola, they realized quickly what was happening, so they put out on the market an 8-bit microprocessor 6800. Chief constructor was Chuck Peddle, and along with the processor itself, Motorola was the first company to make other peripherals such as 6820 and 6850. At that time many companies recognized greater importance of microprocessors and began their own developments. Chuck Peddle left Motorola to join MOS Technology and kept working intensively on developing microprocessors. At the WESCON exhibit in United States in 1975, a critical event took place in the history of microprocessors. The MOS Technology announced it was marketing microprocessors 6501 and 6502 at \$25 each, which buyers could purchase immediately. This was so sensational that many thought it was some kind of a scam, considering that competitors were selling 8080 and 6800 at \$179 each. As an answer to its competitor, both Intel and Motorola lowered their prices on the first day of the exhibit down to \$69.95 per microprocessor. Motorola

quickly brought suit against MOS Technology and Chuck Peddle for copying the protected 6800. MOS Technology stopped making 6501, but kept producing 6502. The 6502 were an 8-bit microprocessor with 56 instructions and a capability of directly addressing 64Kb of memory. Due to low cost, 6502 becomes very popular, so it was installed into computers such as: KIM-1, Apple I, Apple II, Atari, Comodore, Acorn, Orin, Galeb, Orao, Ultra, and many others. Soon appeared several makers of 6502 (Rockwell, Sznertek, GTE, NCR, Ricoh, and Commodore takes over MOS Technology), which was at the time of its prosperity sold at a rate of 15 million processors a year. In 1976, Intel came up with an improved version of 8-bit microprocessor named 8085. However, Z80 was so much better that Intel soon lost the battle. Although a few more processors appeared on the market (6809, 2650, SC/MP etc.), everything was actually already decided. There weren't any more great improvements to make manufacturers convert to something new, so 6502 and Z80 along with 6800 remained as main representatives of the 8-bit microprocessors of that time.

Now we see how to connect the micro controller with other peripheral components or devices when developing your own micro controller system. Each example contains detailed description of hardware with electrical outline and comments on the program

## II Supplying the micro controller

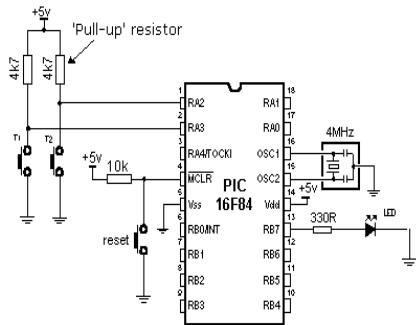
Generally speaking, the correct voltage supply is of utmost importance for the proper functioning of the micro controller system. It can easily be compared to a man breathing in the air. It is more likely that a man who is breathing in fresh air will live longer than a man who's living in a polluted environment. For a proper function of any micro controller, it is necessary to provide a stable source of supply, a sure reset when you turn it on and an oscillator. According to technical specifications by the manufacturer of PIC micro controller, supply voltage should move between 2.0V to 6.0V in all versions. The simplest solution to the source of supply is using the voltage stabilizer LM7805 which gives stable +5V on its output. One such source is shown in the picture below.



In order to function properly, or in order to have stable 5V at the output (pin 3), input voltage on pin 1 of LM7805 should be between 7V through 24V. Depending on current consumption of device we will use the appropriate type of voltage stabilizer LM7805. There are several versions of LM7805. For current consumption of up to 1A we should use the version in TO-220 case with the capability of additional cooling. If the total consumption is 50mA, we can use 78L05 (stabilizer version in small TO - 92 packaging for current of up to 100mA).

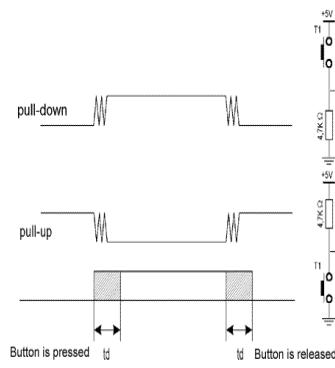
### III Push Buttons

Buttons are mechanical devices used to execute a break or make connection between two points. They come in different sizes and with different purposes. Buttons that are used here are also called "dip-buttons". They are soldered directly onto a printed board and are common in electronics. They have four pins (two for each contact) which give them mechanical stability.



#### A. Example of connecting buttons to micro controller pins

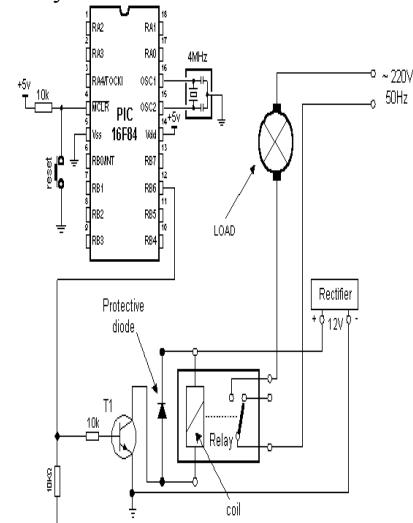
Button function is simple. When we push a button, two contacts are joined together and connection is made. Still, it isn't all that simple. The problem lies in the nature of voltage as an electrical dimension, and in the imperfection of mechanical contacts. That is to say, before contact is made or cut off, there is a short time period when vibration (oscillation) can occur as a result of unevenness of mechanical contacts, or as a result of the different speed in pushing a button (this depends on person who pushes the button). The term given to this phenomenon is called Switch (Contact) debounce. If this is overlooked when program is written, an error can occur, or the program can produce more than one output pulse for a single button push. In order to avoid this, we can introduce a small delay when we detect the closing of a contact. This will ensure that the push of a button is interpreted as a single pulse. The debounce delay is produced in software and the length of the delay depends on the button, and the purpose of the button. The problem can be partially solved by adding a capacitor across the button, but a well-designed program is a much-better answer. The program can be adjusted until false detection is completely eliminated. Image below shows what actually happens when button is pushed.



As buttons are very common element in electronics, it would be smart to have a macro for detecting the button is pushed. Macro will be called *button*. *Button* has several parameters that deserve additional explanation.

### IV Micro controller connected with Relay

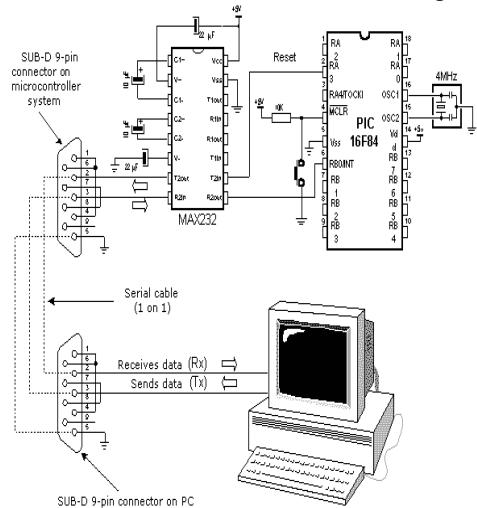
The relay is an electromechanical device, which transforms an electrical signal into mechanical movement. It consists of a coil of insulated wire on a metal core, and a metal armature with one or more contacts. When a supply voltage was delivered to the coil, current would flow and a magnetic field would be produced that moves the armature to close one set of contacts and/or open another set. When power is removed from the relay, the magnetic flux in the coil collapses and produces a fairly high voltage in the opposite direction. This voltage can damage the driver transistor and thus a reverse-biased diode is connected across the coil to "short-out" the spike when it occurs. Since micro controller cannot provide sufficient supply for a relay coil (approx. 100+mA is required; micro controller pin can provide up to 25mA), a transistor is used for adjustment purposes, its collector circuit containing the relay coil. When a logical one is delivered to transistor base, transistor activates the relay, which then, using its contacts, connects other elements in the circuit. Purpose of the resistor at the transistor base is to keep a logical zero on base to prevent the relay from activating by mistake. This ensures that only a clean logical one on RA3 activates the relay.



### B Connecting a relay to the micro controller via transistor

In order to connect a micro controller to a serial port on a PC computer, we need to adjust the level of the signals so communicating can take place. The signal level on a PC is -10V for logic zero, and +10V for logic one. Since the signal level on the micro controller is +5V for logic one, and 0V for logic zero, we need an intermediary stage that will convert the levels. One chip specially designed for this task is MAX232. This chip receives signals from -10 to +10V and converts them into 0 and 5V.

The circuit for this interface is shown in the diagram below:



Connecting a micro controller to a PC via a MAX232 line interface chip

### V CONCLUSION

Despite of its great applications in the field of Electronics, microcontrollers have been widely used these days. Many derivative microcontrollers have since been developed that are based on--and compatible with these microcontrollers. Thus, the ability of interfacing and programming is an important skill for anyone who plans to develop products that will take advantage of microcontrollers.

The various topics of the this paper will explain How to Interface different circuits to microcontroller like Switch, Relay, push buttons etc. An example of serial interfacing has also been discussed. The topics are targeted at people who are attempting to learn microcontroller hardware, interfacing and assembly language programming.

### VI. REFERENCES

- [1]. Microcontroller Interfacing. By Angel Custodio, & Ramon Pallàs-Areny.
- [2]. SanDisk CompactFlash & Motorola 8bit microcontroller interface design reference.
- [3]. Microcontroller Interfacing by Powell.
- [4]. Architecture of proficiency in microcontroller interfacing by Ellipse Librairie.

# Artificial Intelligence for Optimization

Shikha Gupta and Supriya

Lecturers in EE department, Chitkara Institute of Engineering and technology, Rajpura

**Abstract**— Artificial intelligence started as a field whose goal was to replicate human level intelligence in a machine. The use of Artificial Intelligence methods is becoming increasingly common in every field of engineering. It includes basically three branches namely Fuzzy logic, Genetic Algorithm and Artificial Neural Network. In this paper GA and its application on Optimal power flow is discussed. The genetic algorithm technique was applied to the calculation of the fuel cost and optimal value of fuel cost is calculated.

**Keywords**-Artificial intelligence, genetic algorithm, generation, fuel cost.

## I. INTRODUCTION

**Genetic Algorithm:** It is proposed by Holland and is known to be an efficient search and optimization mechanism which incorporates the rules of natural selection. The development of parallel computers and microprocessors have made this algorithm one of the real interest and the birth of a new field of research, experimentation and application known as evolutionary computation. GA has a capability of parallelism and is used for solving stochastic optimization problems. GA was originally proposed by Holland and then reformulated and customized by many other scientists.

## II. HISTORY OF GENETIC ALGORITHM

Genetic Algorithms were invented to mimic some of the processes observed in natural evolution. Many people, biologists included, are astonished that life at the level of complexity that we observe could have evolved in the relatively short time suggested by the fossil record. The idea with GA is to use this power of evolution to solve optimization problems. The father of the original Genetic Algorithm was John Holland who invented it in the early 1970's and thereafter he and his students contribute much to the development of this field. Holland research was not focused on optimization and domain specific practical problem but was on the concept of adaptation as seen in nature. so we are saying that the genetic algorithm is related with the nature, so there is some analogy between them and this can be described as

Genetic Algorithm	Nature
Optimization problem	Environment
Feasible solution	Individuals living in that environment
A set of feasible solution	Population of organism
Fitness function	Individual degree of adaptation
Operators used for results	Selection, recombination, mutation in nature

## III. VARIOUS OPERATORS USED IN GENETIC ALGORITHM

In GA basically three types of operators are used to find the optimal solution and these are

1. selection operator
2. crossover operator
3. Mutation operator.

### A. Selection operator

Initially from the initial population many individual solutions are randomly generated. The population size depends on the nature of the problem, but we take the optimal and feasible size of solutions. Traditionally, the population is generated randomly, covering the entire range of possible solutions. Firstly the selection operator is applied. During each successive generation, a proportion of the existing population is selected and from this a new generation is generated. Individual solutions are selected through a *fitness-based* process, where the fittest population is selected. There are many ways in which the selection operator is applied it includes roulette wheel selection and tournament selection, rank based selection etc.

### B. Crossover operator

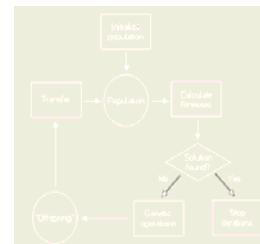
After selection crossover operator is applied. In this the crossover between the two best generated population is done and the best population is selected for the further evaluation. Cross over can be done by various methods. Like matrix crossover, uniform crossover, multidimensional crossover, etc.

### C. Mutation operator

This is also known as termination operator and is the last operator which operated over the best suited population.. Common terminating conditions are:

- A solution is found that satisfies minimum criteria
- Fixed number of generations reached
- Allocated budget (computation time/money) reached
- The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results
- Manual inspection
- Combinations of the above

The following above procedure can be explained with the help of a diagram



## IV. OPTIMAL POWER FLOW BASED ON GENETIC ALGORITHM

Although different classical methods can be used for solving the optimal power flow, but all these methods suffers from the drawback and are not able to provide the efficient results. So now a days artificial intelligent tech are used to solve Optimal power flow. Genetic algorithms offer a new and powerful approach to the optimization problems and make the performance of computers better at relatively low

costs. These algorithms have recently found extensive applications in solving global optimization searching problems. Genetic algorithms (GAs) are parallel and global search techniques that emulate natural genetic operators. The GA is more likely to converge toward the global solution because it, simultaneously, evaluates many points in the parameter space.

- 1) Linear programming method and non linear programming method are not suitable for constraints problem.
- 2) In Newton method the inequality constraints are added as quadratic penalty terms to the problem objective and multiplied by appropriate penalty multiplier. Newton method suffers from the difficulty in handling inequality constraints
- 3) These are not able to provide the optimal solution and usually getting stuck at a local optimum
- 4) All these methods are based on the assumption of continuity and differentiability of the objective function, which is not true in a practical system
- 5) All these methods cannot be applied with discrete variables which are transformer taps.

#### A. Problem formulation

The main objective of the optimal power flow is to minimize the objective function and can be given as:

$$\text{Minimize } F(x) \quad (\text{the objective function})$$

subject to :

$$\begin{aligned} h_i(x) = 0, i = 1, 2, \dots, n & \quad (\text{equality constraints}) \\ g_j(x) = 0, j = 1, 2, \dots, m & \quad (\text{inequality constraints}) \end{aligned}$$

here

$x$ = vector of control variables.

In this paper the objective is to minimize the fuel cost and fuel cost of a generator is given by:

$$F(x) = \sum_{i=1}^{ng} (a_i + b_i P_{gi} + c_i P_{gi}^2)$$

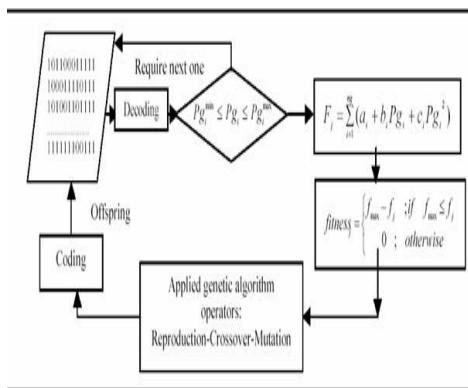
Where

$ng$  is the number of generation including the slack bus.

$P_{gi}$  is the generated active power at bus  $i$ .

$a_i$ ,  $b_i$  and  $c_i$  are the unit costs curve for  $i^{th}$  generator

#### B. Flow chart of GA based OPF



#### V. ADVANTAGES OF GA

A GA has a number of advantages.

1. It can quickly scan a vast solution set.
2. Bad proposals do not effect the end solution negatively as they are simply discarded.
3. The inductive nature of the GA means that it doesn't have to know any rules of the problem - it works by its own internal rules. This is very useful for complex or loosely defined problems

#### VI. CONCLUSIONS

The main purpose of the optimal power flow is to find the optimized solution. In the optimized solution the optimal value of the objective function is find out and it can be done by various methods. But due to the shortcomings of the classical methods those method s are not used and artificial intelligent tech are used, which have the advantage over the previous one methods.

#### VII. REFERENCES

- [1]. www.wikipedia.com.
- [2]. Kalyanmoy Deb, "Multi Objective Optimization using Evolutionary Algorithm," John Wiley & Sons, Ltd, 2001.
- [3]. S.Rajasekaran, G.A Vijayalakshmi pai, "Neural Networks, Fuzzy Logic, and Genetic Algorithm Synthesis and Application", Prentice -Hall of India private Ltd. New delhi, 2008.
- [4]. J.A Momoh , " A generalized quadratic based model for optimal power flow", in proc. of IEEE ,pp. 261-267,1989 B. J. Rabi, T. Parithimarkalaignan and R. Arumugam, "Harmonic elimination of inverters using blind signal separation", Proc. International Conf. on Solid-State and Integrated Circuits Technology, ICSICT 2004, pp 1625-1628.
- [5]. Tarek Bouktir, Linda Slimani, M.Belkacem, " A genetic algorithm for solving Optimal power flow problem"

# FPGA Based Modeling and Simulation of RLC Transient Circuit

Mrs S.L Shimi<sup>\*1</sup>, Mrs Anu Singla<sup>#2</sup>, \*Sr lecturer, CIET,Rajpura,Punjab, #Assistant professor, CIET,Rajpura,Punjab

**Abstract-** Electrical power circuits are difficult to be directly implemented in FPGA. Thus such circuits should be first modeled, normalized and then implemented using FPGA. The paper discusses the basic results from modeling and simulation of Field Programmable Gate Array (FPGA) based RLC transient circuit.

**Key words-** FPGA, LE

## I. INTRODUCTION

Any system can be represented by a mathematical model. Dynamic systems are represented by differential equations or difference equations. In online simulation tools, the solution is carried out in a non-real time manner. Here the actual time involved in the calculation of the variables is more. On the other hand in Real-Time Simulation, the results are produced almost instantly. This is possible if the system model is implemented by an electronic circuit. But analog circuits are less flexible in simulating complex systems. Real-time simulation can be done with digital circuits or microprocessors. But conventional microprocessors and Digital Signal Processors (DSPs) suffer from an inherent bottleneck in performing calculations. They process the data in a sequential manner. So, there is a limit to the minimum simulation step time that is possible for a fixed clock frequency. This limitation can be overcome by paralleling processors. An FPGA is a suitable platform for implementing such systems. The basic advantage of an FPGA is that it can be programmed to process data in parallel. Thus the implementation of system equations on an FPGA results in very short execution time. The system model is realized as a combination of sequential and combinational logic elements. This digital circuit is then programmed in to the FPGA[2].

## II. FIELD PROGRAMMABLE GATE ARRAY (FPGA) BASED DIGITAL PLATFORM

FPGA has Logic Elements (LEs) as their building blocks. Each LE contains hardware resources such as gates, flipflops, decoder, counter etc., The hardware resources available in these LEs are wired every time to realize required logic. Number of logic elements available indicates logic density of the FPGA. The hardware resources available in each LE varies according to the manufacturer. In FPGA the hardware is configurable which was not possible with DSP or Microcontroller. FPGA provides resources for the user in a more abstract level. So with these resources any specific hardware such as DSP/Microcontroller can be configured according to the requirement.

## III. HARDWARE REQUIREMENTS FOR FPGA BASED DIGITAL PLATFORM

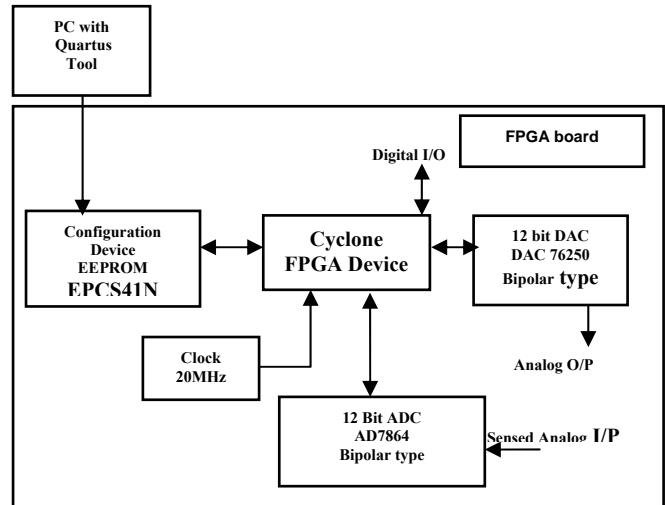
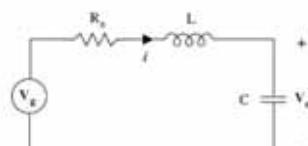


Fig. 1.1 Block diagram of the developed digital platform

The block diagram of the developed digital platform is shown in Fig. 1.1. This digital platform consists of cyclone FPGA, configuration device (EEPROM) and other interfacing devices such as Analog to Digital Converter (ADC), Digital to Analog Converter (DAC) and digital I/Os which are dedicated I/O pins of the FPGA device. ADC and DAC are also interfaced using dedicated I/O pins of the FPGA device. For control of power electronic systems, analog signals such as voltages and currents must be sensed. Hence ADC is required. Sometimes there may be need to send control output in analog form so DAC is required. DAC can also be used to view some digital signals in analog form for debugging purposes. Digital I/Os can be used for control output such as gating pulses to power converter, and also it can be used to interface devices such as ADC, DAC, LCD display, keyboard etc. These are the minimum requirements of the FPGA based digital platform for control of power electronic systems.

## IV. SIMULATING AN RLC CIRCUIT



$$V_g = 100 \text{ V}; R = 10 \Omega; L = 20 \text{ mH}; C = 4 \mu\text{H}$$

Fig 1.2

An electrical series RLC circuit is shown in Fig 1.2. A transient current and voltages are established in the circuit

when the switch is suddenly closed. Equations that describe the transient behavior of the circuit are as given below.

Equations (1.1 & 1.2) are a pair of first order linear differential equations that can be solved using any of the numerical methods technique. The equations are first normalized with the help of arbitrary values  $V_b$ ,  $R_b$ .

$$\frac{V_g}{V_b} = \frac{R}{R_b} \frac{i}{t_b} + \frac{L}{R_b} \frac{d\frac{i}{t_b}}{dt} + \frac{v_c}{V_b} \quad \dots \quad 1.3$$

where  $i_b = \frac{v_b}{R_b}$

With the following abbreviations,

a nondimensional equation results

$$\begin{bmatrix} \tau LR \frac{di^*}{dt} \\ \tau CR \frac{dv_c^*}{dt} \end{bmatrix} = \begin{bmatrix} -R^* & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} i^* \\ v_c^* \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} V_g^* - 1.8$$

These first order linear differential equations can be solved using any numerical methods[1].

## V IMPLEMENTATION

Parameters : The Table. 1.1 gives the base values for voltage, current and the values of other quantities.

Table 1.1  
Parameters

Parameters	
Input Voltage (V <sub>a</sub> )	100V
R <sub>a</sub> , L, C	10Ω, 20mH, 4μF
Voltage (V <sub>b</sub> )	100V
Current (I <sub>b</sub> )	10A
R <sup>*</sup>	V <sup>*</sup> /I <sup>*</sup> = 1
□LR	L/R <sub>b</sub> = 2e <sup>-3</sup>
□CR	CR <sub>b</sub> = 40e <sup>-6</sup>
Step time (ΔT)	25.6μ

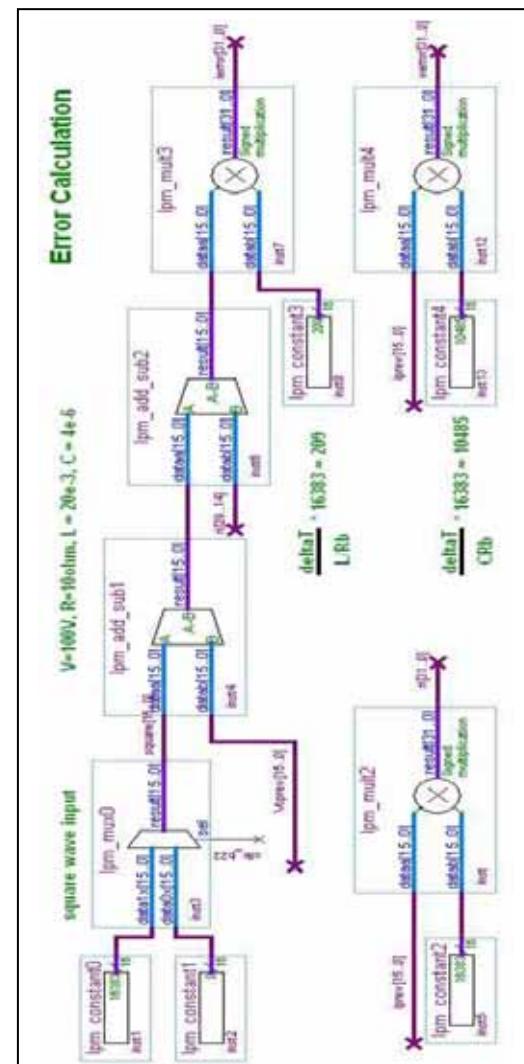
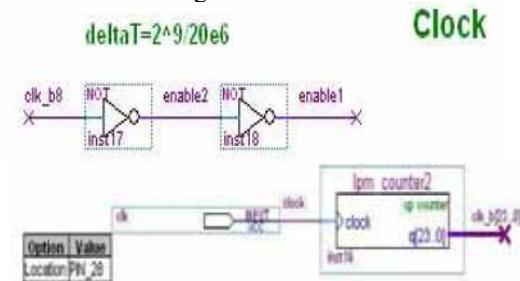
Table 1.2  
PU Values

PC Values		
pu value	Equivalent digital Value	Equivalent decimal value
2 pu	$7FFF_H$	$32767_d$
1 pu	$3FFF_H$	$16383_d$
0 pu	$000_H$	$0_d$
-1 pu(16bit)	$C000_H$	$49152_d$
-2 pu(16bit)	$8000_H$	$32768_d$

**PU System Followed :** The Table 1.2 shows the digital equivalent for the pu values. The bit length of the digital word is not limited in an FPGA. In order to incorporate the signed arithmetic, the digital equivalent for a negative pu value is chosen as the 1's compliment of digital equivalent of its corresponding positive pu value.

## VI. FPGA DESIGN FILES

FPGA Design files of RLC Circuit are placed in 'fpga program files' folder. Fig. 1.3 and fig 1.4 are the FPGA design files to implements the equations given in (1.8). The output waveforms are shown in Fig. 1.5



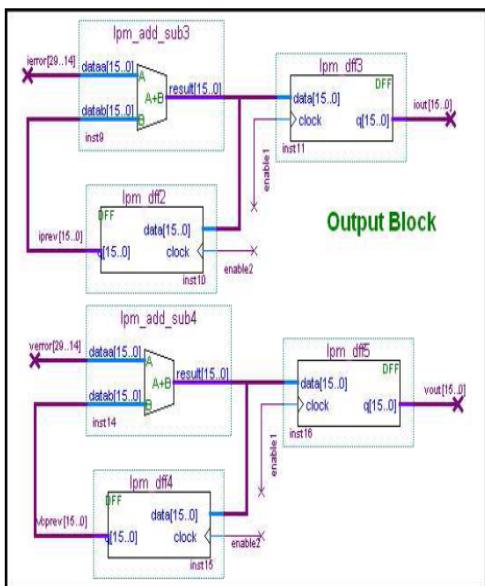


Fig 1.4 FPGA Design File for Euler's Integration



Fig 1.5: RLC Circuit (Eulers Integration).

- (1) Input Voltage
- (2) Inductor Current
- (3) Capacitor Voltage

## VII. CONCLUSION

In this paper a simple RLC circuit has been implemented in FPGA based controller.

## VIII. REFERENCES

Books:-

- [1]. Steven C. Chapra, Raymond P. Canale(1990), Numerical Methods for Engineers,2nd edition.
- [2]. S. Venugopal and G. Narayanan (2005)"Design of FPGA Based Digital Platform for Control of Power Electronics Systems", Proceedings of 2nd National Power Electronics Conference, December 22-24, pp. 409-413.
- [3]. V.T Ranganathan, 'Course Notes on Electric Drives,"Department of Electrical Engg., IISc, Bangalore.Website:- <http://www.altera.com/products/software/quartus-ii/web-edition/qts-we-index.html>

# Review of Challenges of Modeling VLSI Interconnects in the DSM Era

MANINDER KAUR- Lecturer, RIEIT Railmajra, [Kmaninder23@yahoo.com](mailto:Kmaninder23@yahoo.com)

**Abstract-** The parasitic due to interconnects are becoming a limiting factor in determining circuit performance, As VLSI technology shrinks to deep sub-micron (DSM) Geometries. This paper reviews initially the reasons behind the replacement of aluminium by copper. With rise in technology, copper also faces various challenges.

An accurate modeling of interconnect parasitic resistance (R), capacitance (C) and inductance (L) is thus essential in determining various chip interconnect related issues such as delay, cross-talk, IR drop, power dissipation etc. Later in this paper practical methods of accurately estimating R, C, and L of a given circuit layout that maximizes the accuracy while minimizing the time and resources that such accuracy demands is discussed.

**Keywords:** Interconnect parasitic extraction, interconnect modeling.

## I. INTRODUCTION

The feature size of integrated circuits has been aggressively reduced in the pursuit of improved speed, power, silicon area and cost characteristics. Semiconductor technologies with feature sizes of several tens of nanometers are currently in development. As per International Technology Roadmap for Semiconductors (ITRS), the future nanometer scale circuits will contain more than a billion transistors and operate at clock speeds well over 10GHz[1].

Distributing robust and reliable power and ground; clock; data and address; and other control signals through interconnects in such a high-speed, highcomplexity environment, is a challenging task. The performance of a high-speed chip is highly dependent on the interconnects, which connect different macro cells within a VLSI chip[1].With ever increasing circuit density, operating speed, and high level of integration in deep sub-micron (DSM) chip designs, an accurate and proper interconnect modeling is a must to assure the performance and functionality of multi-million transistor VLSI circuits.

These days, a VLSI chip is characterized according to the parameters like parasitic capacitance, inductance and resistance of interconnects. Leaving the few flaws like scaling, skin effects, electron surface scattering and grain boundary scattering these interconnects are studied under the constraints of increasing size, power consumption, crosstalk, electromagnetic interference and delay.

## II. COPPER AS SUITABLE INTERCONNECT

For several semiconductor technology generations, aluminium was used as the on-chip interconnect metal and silicon-di-oxide ( $\text{SiO}_2$ ) as the inter-level and intra-level insulator. With rapid scaling of feature size to deep submicron levels, the signal delay caused by the interconnect became increasingly significant compared to the delay caused by the gate and thus affecting the circuit's reliability. As per ITRS predictions, for nanometer size gate lengths the

interconnections will decide the communication speed of a VLSI chip. The interconnect delay is mostly affected by resistive and capacitive parasitics. For decreasing the resistive part of the RC delay, various alternatives to aluminium were considered in early 90's.

Copper with close to half the resistivity (1.7mW.cm) compared to Al/0.5% Cu alloys (3.0mW.cm) and with electromigration of the order of ten times better appeared to be most appropriate material for VLSI interconnect. Although copper also creates deep levels in the silicon band gap, but several materials that can act as diffusion barrier for copper were found. To prevent copper from diffusing into transistors, it must be encapsulated in a barrier film, usually a derivative of tantalum or titanium. Adding to its merit, copper has a higher melting point (1357 K) than aluminum (933 K), which gives copper the advantage over aluminum in electromigration and stress migration as well. The typical VLSI application temperature range (373 K) is about 40% of the aluminum melting point and 27.4% of the copper melting point. This suggests that mass transport (copper diffusion) in copper is generally slower than that in aluminum at room temperature. Today, copper is widely used on chip interconnect for advanced integrated circuits.

## III. CHALLENGES FOR COPPER

In early 2000, it was realized that even copper is not able to fulfill the demands of high-speed interconnects. With the increase in the integration density of the CMOS and higher clock frequency, the requirement of lower resistance and higher bandwidth is the major concern in interconnect design. To accommodate more interconnects in a chip the cross-sectional dimensions are being reduced rapidly resulting in dimensions reaching in the order of the mean free path of electrons (~40 nm in Cu at room temperature). Furthermore, the resistivity of copper interconnects is increasing rapidly under the effects of enhanced grain and surface scattering; larger interconnect length; and higher frequency operation. Due to the decreasing thermal conductivity of low-k dielectrics and increasing current density in concise dimension interconnect, the rising Cu resistivity also poses a reliability concern due to Joule heating. The increased heating stimulates electromigration induced hillocks and voids. One key constraint in the conventional scaling of silicon VLSI is the high interconnect-related power dissipation per unit area. Thus researchers require serious refinement in copper interconnect technology because copper interconnect is limited by skin effect, dispersion, signal degradation, power dissipation, and electromagnetic interference at higher frequency. While working at high frequencies certain problems persist like crosstalk, skin effect, signal degradation and crosstalk induced propagation delay. Electrons transmitted through metal wires have an information

carrying capacity limited by the resistance and capacitance of the cable and the terminating electronic circuits.

#### IV. FUTURE INTERCONNECTS

To overcome such problems, VLSI designers gear up with certain methods and materials which will be of eminent use in upcoming days, promising ones are Carbon Nanotubes (CNTs) and Optical interconnects. Optical interconnects with their lowered cost due to advancement in technology is becoming intense researched area. The on-chip optical interconnection has several advantages such as no interference at interconnection crossings, no crosstalk between high-density signal lines, low cost interconnect architecture, reduced power loss, reduction in number of I/O pins by use of wavelength division multiplexing (WDM). Optical interconnect is expected to be used for clock distribution in order to achieve high-speed processing.

Carbon nanotubes exhibit a ballistic flow of electrons with electron mean-free paths of several micrometers, and are capable of conducting very large current densities[1]. Moreover, they posses larger lifetime compared to copper interconnect. It is believed that carbon nanotube shall outperform copper interconnects in near future.

#### V. INTERCONNECT MODELING

An interconnection can be described by means of its three electrical parameters, resistance (R), capacitance (C) and inductance (L). The values of these parameters depend on the physical and geometric description of the wire. However, faster on-chip rise times, longer wire lengths, use of low resistance Cu interconnects and low-K dielectric insulators have necessitated the modeling of wire inductive (L) effects. Wide wires are frequently encountered in global and semi-global interconnects in upper metal layers. These wires are low resistive lines that can exhibit significant inductive effects. Due to presence of these inductive effects, the new generation VLSI designers have been forced to model the interconnects as distributed RLC transmission lines [2-3]. These RLC transmission line when running parallel to each other have capacitive and inductive coupling, which makes the design of interconnects even more important in terms of crosstalk. Modeling interconnects as distributed RLC transmission line, has posed many challenges in terms of accurately determining the signal propagation delay; power dissipation through an interconnect; crosstalk between co-planar interconnects and interconnects on different planes due to capacitive and inductive coupling; and optimal repeater insertion.

Selection of the RLC model in terms of both simplicity and accuracy must be made in relation to values of wire parameters, driver parameters and the frequency bandwidth of the signal that the wire must transmit [4]. Historically interconnect was modeled as a single lumped capacitance using the well known parallel plate capacitance formula. With the continuous scaling of technology the wire cross sectional area decreased, while at the same time wire length increased due to increase in the die area, resulting in a wire resistance that becomes significant. This leads to the development of the RC delay model first as a lumped RC circuit (Figure 1i) and then as a distributed RC

model (many sections of lumped RC) to improve accuracy. With faster on-chip rise time, higher clock frequencies and use of Cu wire as interconnect has necessitated the use of RLC models as a distributed network (Figure 1). The computation of R, L and C matrices of a multiport, multiconductor VLSI interconnects involves solution of quasi-static electro-magnetic system of equations (Maxwell Equations) [5]

$$\text{curl } \mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t} \quad \text{div } \mathbf{D} = \rho \quad (i)$$

#### Ampere theorem

$$\text{curl } \mathbf{E} = - \frac{\partial \mathbf{B}}{\partial t} \quad \text{div } \mathbf{B} = 0 \quad (ii)$$

#### Faraday Law: Constitutive equations

$$\mathbf{B} = \mu \mathbf{H}, \quad \mathbf{D} = \epsilon \mathbf{E}, \quad \mathbf{J} = \sigma \mathbf{E} \quad (iii)$$

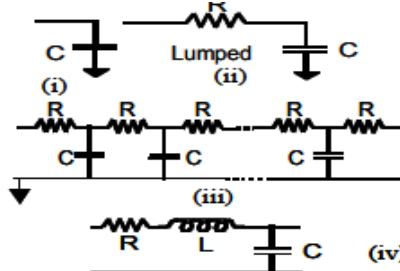


Figure 1: Interconnect delay model. Lumped (i) capacitor model, (ii) RC model. (iii) Distributed RC model, and (iv) RLC model.

To be solved with initial and boundary conditions, where symbols have their usual meaning. Choice of numerical techniques to solve the partial differential equations such as finite difference (FD), finite element (FE), boundary element method (BEM) etc. leads to different methods of discretization of the domains by elements of simple shapes (meshes) [5]. In all these methods differential equations are converted into algebraic equations which in turn are solved by different methods such as iterative, direct or multigrid methods to calculate potential and fields. Interpolation or integration then leads to interconnect parameter of interest such as R, L, and C. Though each of these methods gives similar results in a dense layout system, they often give different results in a sparse layout system depending upon the boundary conditions used and problem set-up. Also each method has its own advantages and disadvantages.

Although numerical methods discussed above provide an accurate electromagnetic field solution of a complex geometry, they are too compute intensive and cannot be used at the chip level extraction of wire RLC. In fact these methods are not practicable for circuits containing larger than few tens of transistors. At the chip level, an estimation of wire RLC requires an approach that is very efficient and has reasonable accuracy to say within 10% (of the field solver value) for each net in a chip containing millions of nets. Different modeling

approaches of computing RLC at the chip level has resulted in different types of tools [6] namely (1) Rule based that used Boolean operations, (2) pattern matching using look-up table and (3) context based that looks at each conductor with 3D surrounding. All these approaches use field solvers to precalculate R and C for different structures. With table look-up, the memory requirement grows very quickly due to an increase in the number of parameters required for each configuration. Sophisticated interpolations are used to reduce the amount of data that needs to be stored. Analytical models execute quickly providing faster extraction, but needs sophisticated model development, particularly for C calculation, because capacitance is complex function of layout parameters [7]. For R calculation, it is the analytical modeling approach that is more common, while for L and C both approaches are used. The software tool that reads the layout geometry and calculates the corresponding RLC of the wires is known as Layout to Parasitic Extractor (LPE), or simply parasitic extractor.

### 5.1 Resistance Extraction

Computation of R is the simplest of the three interconnect parameters because R is a function of the conductor geometry and conductivity only, being independent of adjoining conductors. To calculate R, interconnections are approximated by 2D homogeneous regions (metals, polysilicon, and diffusion area) that are connected to each other along surfaces representing vias between these regions. Resistance of each wire is then obtained simply by multiplying sheet resistance,  $r_s$ , by ratio of the length to width of the wire. Due to interconnect being multilayer, sheet resistance of a wire, particularly Cu wires, now becomes line width and pattern dependent. The lower the width, higher the resistance. New processes require slotting of the wide wires and as such one needs to use 2D as solver to precalculate impact of the slots on wire resistance. As signal frequency increases the penetration depth of the electromagnetic field in the conductor decreases thereby increasing conductor resistance. As clock frequency exceeds 1 GHz, impact of skin effect must be considered. However, modeling R with skin effect is not straight forward [8].

### 5.2 Capacitance Extraction

The capacitance estimation of a wire is more complex compared to the resistance estimation. This is because capacitance between two wires is affected by the proximity of the other wires that could be above, below, or adjacent to the wire under consideration. Historically, the wire was modeled as a parallel plate capacitance ( $C = \epsilon_0 W / H$ ) where  $W$  and  $H$  are length and width, respectively, of the wire and  $H$  is height of the wire from the substrate such that  $W \ll H$  and wire thickness  $T$  is negligible. This, the so called 1D formula, became inadequate with the scaling of the technology when lateral capacitance became significant part of the total line capacitance. In the 2D modeling approach all capacitive effects (area plus lateral) are modeled as long routing lines with per unit length values. This 2D model though more accurate compared to the 1D, still gives significant error while calculating capacitance of two crossing lines which is a 3D

problem. To address the 3D effects the so called 2.5D model is proposed which in effect combines two orthogonal 2D models [7]. However, for DSM technologies a true 3D model is needed, where capacitance of a 3D pattern is calculated. 3D model is not a trivial extension of a 2D model.

Given the description of the process cross-section, the first step is to generate capacitance data using 3D field solvers for tens of thousands of structures. This data is generated once for each technology as a function of line width,  $W$ , and spacing,  $S$ . The resulting data is then fitted into physics based empirical models using non-linear optimization method. The model parameter coefficients are then stored. In some other approaches the data is stored as a look-up table (also called a pattern library). Layout geometry is fractured first into stripes (window) and then into elemental area with polygons. Capacitance is then calculated for each vertical profile containing these polygons using stored model equations or look-up table.

### 5.3 Inductance Extraction

Inductance of a wire is much more complicated to model compared to resistance or capacitance because inductance is defined only for a loop of a wire. In a VLSI chip, calculation of wire inductance will therefore require knowledge of the current return path(s). However, often the return path is not easily identified from the layout as it is not necessarily through the silicon substrate. Calculation is further complicated by the fact that not all the return currents follow DC paths as some are in the form of displacement current through the interconnect capacitances. Furthermore, unlike capacitive coupling, inductive coupling is much stronger (has long range effects). As such, localized windowing (nearest neighbor approximation), commonly used for capacitance calculation, is not valid for inductance calculation as it leads to unstable interconnect models [9]. On a chip it is not clear which conductor forms the loop. The concept of partial inductance is introduced which allows algebra to take care of determining the loops [11]. Each partial inductance assumes current return at infinity so that “infinity return” paths cancel out when we do the subtraction. Based on this approach one can easily calculate self and mutual inductance of two parallel lines of length  $l$ , width  $w$ , and thickness  $t$ , separated by distance  $d$ .

While wire capacitance is important for any length of a wire, the wire inductance is significant only for a certain ranges of wire length. Extracting wire inductance at the chip level for all nets is not practicable due to large amount of data generated (unknown return path). As such, inductance calculation is restricted to clock lines and buses only. Furthermore today's commonly used RC delay calculators do not efficiently handle inductance and work is in progress [10]. For these reasons inductance often is reduced by process design using metal plates or intersdigitated shields.

A typical parasitic extractor often takes 24-48 hours (single CPU) to just extract R and C. To use this huge parasitic database for downstream tools such as timing analysis, signal integrity etc, the data needs to be reduced. The most commonly used model order reduction technique is the asymptotic

waveform evaluation (AWE) with varying order moment matching (Elmore delay being first order matching).

These RLC transmission line when running parallel to each other have capacitive and inductive coupling, which makes the design of interconnects even more important in terms of crosstalk. These RLC transmission line when running parallel to each other have capacitive and inductive coupling, which makes the design of interconnects even more important in terms of crosstalk. In a modern interconnect design, the interconnect in an adjacent metal layers are kept orthogonal to each other. This is done to reduce crosstalk as far as possible. But with growing interconnect density and reduced chip size, even the non-adjacent interconnects exhibit significant coupling effects. These coupling effects are significantly dependent on length of interconnects, distance between them, transition time of the input and the pattern of input. On-chip inductance induced noise to signal ratio is increasing because of the increase in switching speed; decrease in separation between interconnects and decrease in noise margins of devices. The impact of this noise, such as oscillation, overshoots and undershoots, on chip's performance is thus of concern in design. The effect of a crosstalk induced overshoot and undershoot generated at a noise-site can propagate false switching and create a logic error. The false switching occurs when the magnitude of overshoot or undershoot is beyond the threshold of the gate. The peak overshoot and undershoot generated at a noise-site can also wear out the thin gate oxide layer resulting in permanent failure of the chip. This problem will be significant as the feature size of transistor reduces with advancement of technology. Extraction of exact values of capacitance and inductance induced noise for an interconnect is a challenging task. For an on-chip interconnect, the different behavior of capacitive and inductive noise must be taken into consideration. Since electrostatic interaction between wires is very short range, consideration of only nearest neighbors provides sufficient accuracy for capacitive coupled noise. Unlike an electric field, a magnetic field has a long range interaction. Therefore in inductive noise extraction not only nearest neighbors but also many distant wires must be considered. As a consequence, defining current loops or finding return paths becomes a major challenge in inductive noise modeling. Since magnetic fields have a much longer spatial range compared to that of electric fields, in practical high-performance ICs containing several layers of densely packed interconnects, the inductive noise are sensitive to even distant variations in the interconnect topology [12]. Secondly, uncertainties in the termination of the neighboring wires can significantly affect the signal return path and return current distributions and therefore the effective inductive noise. Also, accurate estimation of the effective inductive noise estimation requires details of the 3-D interconnect geometry and layout, technology etc., and the current distributions and switching activities of the wires, which are difficult to predict. Moreover, at high frequencies the line inductance parameters are also dependent on the frequency of operation. These are the added complexities for the designers involved in analyzing the behavior of the interconnects. Without involving in the

complexities of a high-performance chip, this paper shows the prominent factors such as edge rate, length and pattern of inputs affecting the noise.

## VI. CONCLUSION

Accurate modeling and characterization of interconnect is essential as it strongly impacts circuit performance. Different modeling approaches for estimating and extracting interconnect parameters - resistance, capacitance and inductance - have been reviewed. Optical interconnects and carbon nanotubes will overcome the regime of copper in near future. Although optical and carbon nanotubes can be safely predicted as future interconnects, but these technologies are still amateur as compared to well established fabrication technique in copper.

## VII. REFERENCES

- [1]. Brajesh Kumar Kaushik,"Exploring Future VLSI Interconnects"
- [2]. A. Deutsch, G.V. Kopcsay, P. Restle, G. Katopis, W.D. Becker, H. Smith, P.W. Coteus, C.W. Surovic, B.J. Rubin, R.P. Dunne, T.Gallo, K.A. Jenkins, L.M. Terman, R.H. Dennard, G.A. Sai-Halasz, and D.R. Knebel, In: Proc. 47-th Electronic Components TechnologyConf., p. 704 (1997).
- [3]. Y.I. Ismail, E.G. Friedman, and J.L. Neves, IEEE Trans. VLSI Syst. 7,442 (1999).
- [4]. C.K. Cheng, J. Lillis, S. Lin and N. Chang, "Interconnect Analysis and Synthesis", John Wiley & Sons, 2000.
- [5]. M.N.O. Sadiku, "Numerical techniques in Electromagnetics", CRC press, 1992.
- [6]. Y.L. LeCoz and R.B. Iverson, Solid-State Electron. 35, 1005, 1992
- [7]. W.H. Kao, C.Y. Lo, M. Basel and R. Singh, Proc. IEEE, 89,729 (2001).
- [8]. B. Krauter and S. Malhotra, Proc. 35th DAC, 303, 1998.
- [9]. M. Beattie and R. Pileggi, Proc. 36th DAC, 915 (1999).
- [10]. Y. Ismail, and E. Friedman, "On chip inductance in High speed integrated circuits", Kluwer Academic Publishers, 2001.
- [11]. A.E. Ruehli, IBM J. Res. Dev., 16, 470 (1972).
- [12]. J.M. Rabaey, Digital Integrated Circuits, A Design Perspective (Prentice-Hall, Englewood Cliffs, NJ, 1996).
- [13]. Narain D. Arora, "Challenges of Modeling VLSI Interconnects in the DSM Era" international conference of modeling , 2002
- [14]. Agarwal, D. (2002), "Optical interconnects to silicon chips using short pulses", *Phd. Thesis submitted at Stanford University*.
- [15]. Banerjee, K. and Srivastava, N. (2006), "Are Carbon Nanotubes the Future of VLSI Interconnections?" *IEEE-DAC*, San Francisco, California.

# An Approach for Improving the Performance of Search Engine using Rank Aggregation through Focused Web Crawling

Neeraj Mangla , Dept of Computer Engineering Vikas Kumar, M. Tech  
M. M. Engineering College, Mullana-133203, Haryana, INDIA, [erneerajynr@yahoo.com](mailto:erneerajynr@yahoo.com)

**Abstract-** The voluminous amounts of web documents, crawl the web have posed huge challenges to the web search engines making their results less relevant to the users. It is an important requirement for search engines to provide users with the relevant results for their queries in the first page without redundancy. In many ways, the instruction learned from the Internet take over directly to intranets, but others do not concern. In this thesis, we study the problem of Internet search. Our approach focuses on the use of rank aggregation through focused web crawling, and allows us to study the effects of different heuristics on ranking of search results [4]. The documents having similarity scores greater than a threshold value are considered as near duplicates. The detection in reduced memory for repositories, improved search engine quality.

## I. INTRODUCTION

The programs that navigate the web and retrieve pages to construct a confined repository of the segment of the web that they visit. Earlier, these programs were known by different names such as wanderers, robots, spiders, fish, and worms, words in accordance with the web descriptions [6]. Generic and Focused crawling are the two main types of crawling. Focused crawling, enhanced quality and diversity of the query results and identification can be facilitated by determining the near duplicate web pages [6, 7, and 12]. In Internet search, many of the most important techniques that guide to quality successfully. Because they use the reflection of shared services in the way people present information on the web.

Generic crawlers [2] differ from focused crawlers [5] in a way that the former crawl documents and links of diverse topics, whereas the latter limits the number of pages with the aid of some prior obtained specialized knowledge. Storage of web pages are built by the web crawlers so as to present input for systems that index, mine, and analyze pages (for instance, the search engines) [13].

The continuation of near duplicate data is an issue that accompanies the drastic development of the Internet and the growing need to incorporate heterogeneous data. Even though the near duplicate data are not bit wise identical, they bear a striking similarity. Web search engines face huge problems due to the duplicate and near duplicate web pages [17]. These pages either increase the index storage space or slow down or increase the serving costs thereby irritating the users. Thus the algorithms for detecting such pages are inevitable [11]. Web crawling issues such as freshness and efficient resource usage have been addressed previously [14].

## II. RANKING: AN EFFICIENT SEARCHING APPROACH

Rank aggregation algorithms, take as input multiple ranked lists from the different heuristics and produces an ordering of the pages, aimed at minimizing the number of “upsets” with respect to the ordering, produced by individual heuristics ranking [14]. Rank aggregation allows us to easily add and remove heuristics, which makes it especially compatible for our experimental purposes [4].

We argue that this architecture is also well suitable for Internet search in general. When designing a general purpose Internet search tool look forward to the architecture to be used in a variety of environments—corporations, small businesses, governmental agencies, academic institutions, etc. It is virtually impossible to claim an understanding of the characteristics of Internet in each of these scenarios [10].

## III. RELATED WORK

The broad method of rank aggregation was applied to the problem of Meta search in [2]. In [14] the authors present the problem of combining different ranking functions in the context of a Bayesian probabilistic model for information retrieval. The well-known theorem of Arrow [4] shows that there can be no election algorithm that simultaneously satisfies five apparently desirable properties for elections.

Andrei Z. Broder et al. [1] have developed an efficient way to determine the syntactic similarity of files and have applied it to every document on the World Wide Web. Using their mechanism, they have built a clustering of all the documents that are syntactically similar [1]. Possible applications include a "Lost and Found" service, filtering the results of Web searches, updating widely distributed web-pages, and identifying violations of intellectual property rights.

Jack G. Conrad et al. [16] have determined the extent and the types of duplication existing in large textual collections. Their research is divided into three parts. Initially they started with a study of the distribution of duplicate types in two broad-ranging news collections consisting of approximately 50 million documents [17]. Then they examined the utility of document signatures in addressing identical or nearly identical duplicate documents and their sensitivity to collection updates. Finally, they have investigated a flexible method of characterizing and comparing documents in order to permit the identification of non-identical duplicates [6].

Donald Metzler et al. [8] have explored mechanisms for measuring the intermediate kinds of similarity, focusing on the task of identifying where a particular piece of information originated.

#### IV.ARCHITECTURE OF FOCUSED WEB CRAWLER

In this section we explain our research prototype for Internet search. The search system has six identifiable components, namely a focused crawler, a course home page, an information extraction engine, an inverted index engine, a query runtime system, and an information retrieval engine.

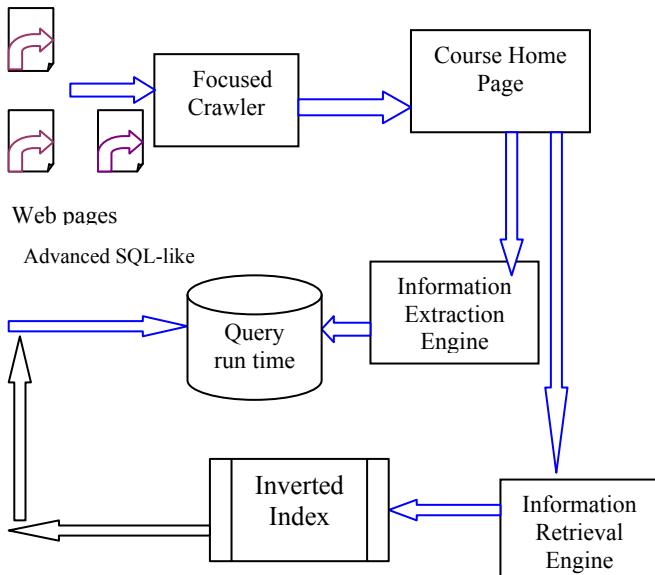


Figure 1: Architecture of Focused Web Crawler

#### *Focused Crawling*

In this study, we first implemented the baseline focused crawler described in [7]. In particular, the crawler involves a document classifier which is trained Web pages with example for each topic in the system. The target topic is given by the user as one or more of these training topics. During the crawling, the crawler maintains a priority queue of URLs to be visited. A URL score is computed according to the classifier score indicating the relevancy of the target topic to the page from which the URL is extracted [9].

#### *Searching and Querying*

The system provides two basic methods to retrieve relevant information to a user query.

- Inverted index based keyword searching: An inverted index of terms included in the Web pages is used to answer the keyword based searches [15]. The Information Retrieval (IR) engine assigns term weights by using the vector space model weighting scheme (see [15] for details). The comparisons between the pages and user queries are computed by the cosine measure [12].
- SQL-like advanced querying: In this case, the users can specify search values for one or more of the fields that are populated during the information extraction stage. The final query is constructed as a typical SQL query and sent to the underlying database system.

#### V. CHALLENGES

- Due to huge size and explosive growth of the Web [10], it becomes more and more difficult for search engines to provide effective services to end-users.

- Browsing many documents to find the relevant ones is time-consuming and tedious.
- The indexable Web has more than 11.5 billion pages. Even Goggle, the largest search engine, has only 76.16% coverage [13]. About 7 million new pages go online each day.
- To address the above problems, domain-specific search engines were introduced, which keep their Web collections for one or several related domains.
- Most existing focused crawlers use local search algorithms to decide the order in which the target URLs are visited, which can result in low-quality domain-specific collections [9].

#### VI. PROPOSED MODEL

All crawling modules share the data structures needed for the interaction with the simulator. The simulation tool maintains a list of unvisited URLs called the frontier [3]. This is initialized with the seed URLs specified at the configuration file. Besides the frontier, the simulator contains a queue [8]. It is filled by the scheduling algorithm with the first k URLs of the frontier, where k is the size of the queue mentioned above, once the scheduling algorithm has been applied to the frontier.

Each crawling loop involves picking the next URL from the queue, fetching the page corresponding to the URL from the local database that simulates the Web and determining whether the page is relevant or not. If the page is not in the database, the simulation tool can fetch this page from the real Web and store it into the local repository. If the page is relevant, the outgoing links [9] of this page are extracted and added to the frontier, as long as they are not already in it.

The crawling process stops once a certain end condition is fulfilled, usually when a certain number of pages have been crawled or when the simulator is ready to crawl another page and the frontier is empty. If the queue is empty, the scheduling algorithm is applied and fills the queue with the first k URLs of the frontier, as long as the frontier contains k URLs. If the frontier doesn't contain k URLs, the queue is filled with all the URLs of the frontier [9]. If the queue is empty, the scheduling algorithm is applied and fills the queue with the first k URLs of the frontier, as long as the frontier contains k URLs. If the frontier doesn't contain k URLs, the queue is filled with all the URLs of the frontier [9].

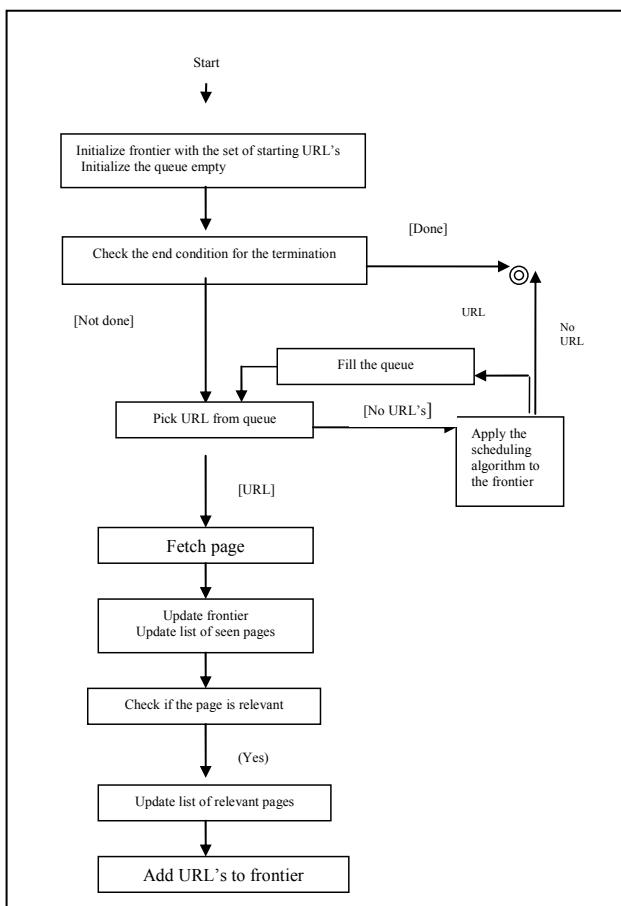


Figure2: Flow of web crawling indexer simulation tools

## VII. REFERENCES

- [1]. Broder, A. Z., Glassman, S. C., Manasse, M. S. and Zweig, G., "Syntactic clustering of the web", Computer Networks, vol. 29, no. 8- 13, pp.1157–1166 1997.
- [2]. Javed, A. Aslam and Mark, Montague. Models for metasearch. In *Proc. 24th SIGIR*, pages 276–284, 2001.
- [3]. Bar-Yossef, Z., Keidar, I., Schonfeld, U., "Do not crawl in the dust: different URLs with similar text", Proceedings of the 16th international conference on World Wide Web, pp: 111 – 120 2007.
- [4]. Dwork, Cynthia, Kumar, Ravi, Naor, Moni, and Sivakumar, D. "Rank aggregation methods for the web". In *Proc. 10th WWW*, pages 613–622, 2001.
- [5]. Castillo, C., "Effective web crawling", SIGIR Forum, ACM Press, Volume 39, Number 1, N, pp.55-56 2005.
- [6]. Manku, Gurmeet S., Jain, Arvind, Sarma, Anish D., "Detecting near-duplicates for web crawling," Proceedings of the 16th international conference on World Wide Web, pp: 141 – 150 2007.
- [7]. Pandey, S.; Olston, C., "User-centric Web crawling", Proceedings of the 14th international conference on World Wide Web, pp: 401 – 411 2005.
- [8]. Metzler, D., Bernstein, Y., and Bruce Croft, W., "Similarity Measures for Tracking Information Flow", Proceedings of the fourteenth international conference on Information and knowledge management, CIKM'05, October 31.November 5, Bremen, Germany 2005.
- [9]. Thijs, Westerveld, Wessel, Kraaij, and Djoerd, Hiemstra. "Retrieving web pages using content links", URLs and anchors. In *Proc. 10th TREC*, pages 663–672, 2001.
- [10]. Agichtein, E., Brill, E., and Dumais, S. T., "Improving Web Search Ranking by Incorporating User Behavior Information", in the Proceedings of SIGIR, 2006.
- [11]. Agichtein, E., Bril,I E., Dumais, S., and . Ragno, R, "Learning User Interaction Models for Predicting Web Search Result Preferences", in the Proceedings of SIGIR, 2006.
- [12]. Al Halabi, W., Kubat, Tapia, M., M. "Search Engine Personalization Tool Using Linear Vector Algorithm" the proceedings of the 4th Saudi Technical Conference. Dec 2006.
- [13]. Beccetti, L., Castillo, C., Donato, D., Yates, Baeza, R. -, Leonardi, S. "Link Analysis for Web Spam Detection", ACM Trans. Web, 2 (1), pp. 1-42, 2008.
- [14]. Anyanwu, K., Maduko, A., and Sheth, A.P.: "SemRank: Ranking Complex Relationship Search Results on the Semantic Web", Proceedings of the 14th International World Wide Web Conference, ACM Press, May 2005.
- [15]. Milne, D. N., Witten, I. H., and Nichols, D. M. "A Knowledge-Based Search Engine Powered by Wikipedia". In CIKM: Proceedings of the sixteenth ACM conference on Information and knowledge management, pages 445–454, New York, NY, USA, ACM. 2007.
- [16]. Conrad, J. G., Guo, X. S., and Schriber, C. P., "Online duplicate document detection: signature reliability in a dynamic retrieval environment", Proceedings of the twelfth international conference on Information and knowledge management, New Orleans, LA, USA, pp. 443 – 452 2003.
- [17]. Narayana, V.A., Premchand, P., Govardhan, Dr. A., "A Novel and Efficient Approach For Near Duplicate Page Detection in Web Crawling" IEEE International Advance Computing Conference (IACC 2009) Patiala, India, 6–7March 2009.

# Led Matrix Display with Labview Code

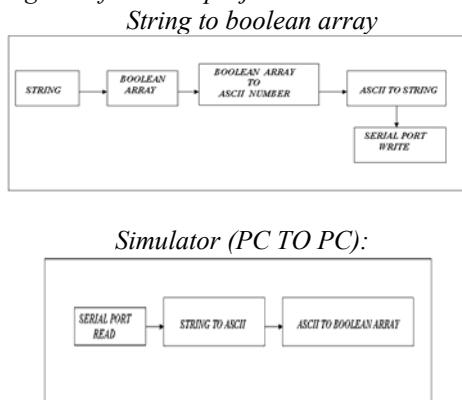
Sanjeev Jain- Student M.TECH (ECE), Vaish College of Engineering, Rohtak  
 Yashudeep Jain- Student B.TECH (ECE), UIET Panjab University, Chandigarh  
 Email:jainmtechb4u@gmail.com,yashuenator@gmail.com

**Abstract-** As computer based measurement and automation become a more integral part of industrial research and development. This paper will present the integration of a LabVIEW measurement system with RS232 controllable hardware. This paper also demonstrate a unique application to make an interface of labview with P89V51RD2 microcontroller via RS232 protocol to display the given message on the hardware board of leds. The string would be given on the lab view editor and it will display on the led board which is already interfaced with microcontroller.

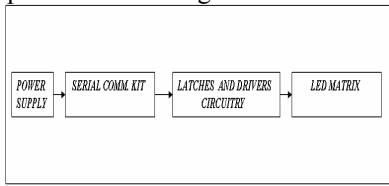
## I. INTRODUCTON

Virtual instrumentation is breaking down the barriers of developing and maintaining instrumentation systems that challenge the world of test, measurement, and industrial automation. By leveraging off the latest computing technology, virtual instrumentation delivers innovative, scalable solutions that incorporate many different I/O options and maximizes code reuse-- saving you time and money.

### 1 Block Diagram of labview project:



### Hardware implementation using P89V51RD2 Microcontroller :



## II. USER INTERFACE EDITOR

Early schematic editors were text-based and difficult to use, but modern editors have a graphical interface that allows designers to drag and drop electronic components onto the screen and make connections between them. Sections of circuitry can be saved and imported later, saving many hours of work. Templates of common circuits can also be imported from third party libraries as shown below in fig1.

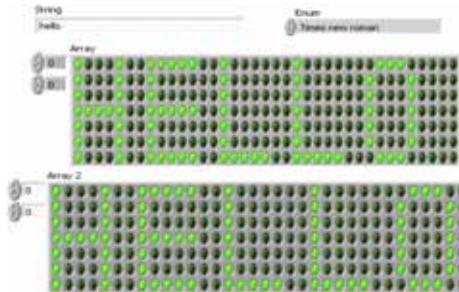


Fig.1. User interface editor and VI of led board displaying the given string "hello"

## III. LOGIC LAYER AND PROTOCOL RS232:

In RS-232, user data is sent as a time-series of bits. Both synchronous and asynchronous transmissions are supported by the standard. In addition to the data circuits, the standard defines a number of control circuits used to manage the connection between the DTE and DCE. Each data or control circuit only operates in one direction, that is, signaling from a DTE to the attached DCE or the reverse. Since transmit data and receive data are separate circuits, the interface can operate in a full duplex manner, supporting concurrent data flow in both directions. The standard does not define character framing within the data stream, or character encoding.

### 1) Voltage levels

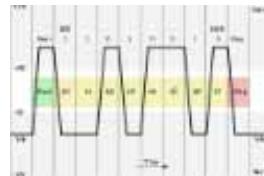


Fig 2. Diagrammatic oscilloscope trace of voltage levels for an uppercase ASCII "K" character (0x4b) with 1 start bit, 8 data bits, 1 stop bit

The RS-232 standard defines the voltage levels that correspond to logical one and logical zero levels for the data transmission and the control signal lines as shown in fig2. Valid signals are plus or minus 3 to 15 volts - the range near zero volts is not a valid RS-232 level. The standard specifies a maximum open-circuit voltage of 25 volts: signal levels of  $\pm 5$  V,  $\pm 10$  V,  $\pm 12$  V, and  $\pm 15$  V are all commonly seen depending on the power supplies available within a device. RS-232 drivers and receivers must be able to withstand indefinite short circuit to ground or to any voltage level up to  $\pm 25$  volts. The slew rate, or how fast the signal changes between levels, is also controlled.

For data transmission lines (Tx, Rx and their secondary channel equivalents) logic one is defined as a negative voltage, the signal condition is called marking, and has the functional significance. Logic zero is positive and the

signal condition is termed spacing. Control signals are logically inverted with respect to what one would see on the data transmission lines. When one of these signals is active, the voltage on the line will be between +3 to +15 volts. The inactive state for these signals would be the opposite voltage condition, between -3 and -15 volts. Examples of control lines would include request to send (RTS), clear to send (CTS), data terminal ready (DTR), and data set ready (DSR).

Because the voltage levels are higher than logic levels typically used by integrated circuits, special intervening driver circuits are required to translate logic levels. These also protect the device's internal circuitry from short circuits or transients that may appear on the RS-232 interface, and provide sufficient current to comply with the slew rate requirements for data transmission.

Because both ends of the RS-232 circuit depend on the ground pin being zero volts, problems will occur when connecting machinery and computers where the voltage between the ground pin on one end, and the ground pin on the other is not zero. This may also cause a hazardous ground loop.

Unused interface signals terminated to ground will have an undefined logic state. Where it is necessary to permanently set a control signal to a defined state, it must be connected to a voltage source that asserts the logic 1 or logic 0 level. Some devices provide test voltages on their interface connectors for this purpose.

## 2) Connectors

RS-232 devices may be classified as Data Terminal Equipment (DTE) or Data Communications Equipment (DCE); this defines at each device which wires will be sending and receiving each signal as shown in Fig 3 . The standard recommended but did not make mandatory the D-subminiature 25 pin connector. In general and according to the standard, terminals and computers have male connectors with DTE pin functions, and modems have female connectors with DCE pin functions. Other devices may have any combination of connector gender and pin definitions. Many terminals were manufactured with female terminals but were sold with a cable with male connectors at each end; the terminal with its cable satisfied the recommendations in the standard.

Presence of a 25 pin D-sub connector does not necessarily indicate an RS-232-C compliant interface. For example, on the original IBM PC, a male D-sub was an RS-232-C DTE port (with a non-standard current loop interface on reserved pins), but the female D-sub connector was used for a parallel Centronics printer port. Some personal computers put non-standard voltages or signals on some pins of their serial ports.

The standard specifies 20 different signal connections. Since most devices use only a few signals, smaller connectors can often be used.

string  
7F08 0808 7F00 007F 4949 4949 0000 7F40 4040 4000 007F 4040 4040 0000 3E41 4141 3E00  
0000 0000 0000 7F08 0808 7F00 007F 4949 4949 0000 7F40 4040 4000 007F 4040 4040  
0000 3E41 4141 3E00 0000 0000 7F08 0808 7F00 007F 4949 4949 0000 7F40 4040 4040

VISA resource name  
ASRL2

Fig 3. String given in user editor is written on Serial port of PC

For example, the 9 pin DE-9 connector was used by most IBM-compatible PCs since the IBM PC AT, and has been standardized as TIA-574. More recently, modular connectors have been used. Most common are 8P8C connectors. Standard EIA/TIA 561 specifies a pin assignment, but the "Yost Serial Device Wiring Standard" invented by Dave Yost (and popularized by the Unix System Administration Handbook) is common on Unix computers and newer devices from Cisco Systems. Many devices don't use either of these standards. 10P10C connectors can be found on some devices as well. Digital Equipment Corporation defined their own DECconnect connection system which was based on the Modified Modular Jack connector. This is a 6 pin modular jack where the key is offset from the center position. As with the Yost standard, DECconnect uses a symmetrical pin layout which enables the direct connection between two DTEs. Another common connector is the DH10 header connector common on motherboards and add-in cards which is usually converted via a cable to the more standard 9 pin DE-9 connector (and frequently mounted on a free slot plate or other part of the housing).

## IV. SIMULATOR ON OTHER PC

A computer simulation (or "sim") is an attempt to model a real-life or hypothetical situation on a computer so that it can be studied to see how the system works. By changing variables, predictions may be made about the behaviour of the system.<sup>[1]</sup>

Computer simulation has become a useful part of modeling many natural systems in physics, chemistry and biology<sup>[4]</sup>, and human systems in economics and social science (the computational sociology) as well as in engineering to gain insight into the operation of those systems. A good example of the usefulness of using computers to simulate can be found in the field of network traffic simulation. In such simulations, the model behaviour will change each simulation according to the set of initial parameters assumed for the environment.

Traditionally, the formal modeling of systems has been via a mathematical model, which attempts to find analytical solutions enabling the prediction of the behaviour of the system from a set of parameters and initial conditions. Computer simulation is often used as an adjunct to, or substitution for, modeling systems for which simple closed form analytic solutions are not possible. There are many different types of computer simulation, the common feature they all share is the attempt to generate a sample of

representative scenarios for a model in which a complete enumeration of all possible states would be prohibitive or impossible.

Several software packages exist for running computer-based simulation modeling (e.g. Monte Carlo simulation, stochastic modeling, multimethod modeling AnyLogic) that makes the modeling almost effortless.

Modern usage of the term "computer simulation" may encompass virtually any computer-based representation.

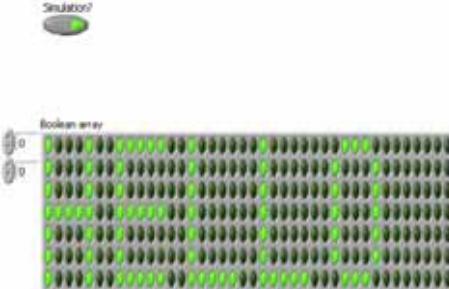


Fig 4.Simulation result of given string in User interface editor in labview.

## V. HADWARE IMPLEMENTATION USING P89V51RD2 MICROCONTROLLER

Since VI writes data on serial port ,it is based on RS232 standards .This output is to be first converted to TTL levels .We do it with help of MAX232 IC .It converts RS232 level signals to TTL signals. A set of receiving and transmitting pair of pins are connected to respective pins of microcontroller,which programmed to recive data at predefined baud rate compatible with computer . Computer sends data in ascii code which convereted to values used by led matrix to display characters.One port of controller is used to send data to latch and another to control latch enable .Latches are enabled and disabled according to need. After latches data is given to LED matrix via drivers and resistors to limit currents and voltages. IN leds matrix anode is given supply while cathode is controlled by data as shown in fig 5.

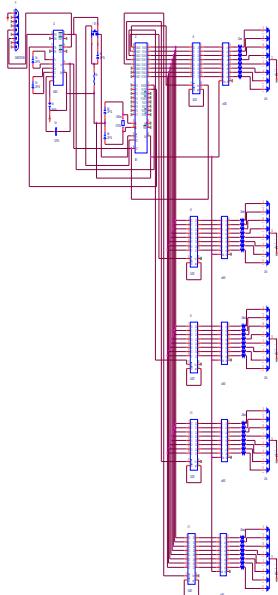


Fig 5.Schematic diagram of hardware using P89V51RD2 controller and 74LS373 with ULN2803 to display on 5x7 Led matrix.

## VI. REFERENCES

- [1] National Instruments. National Instruments™ LabVIEW™: Database Connectivity Toolset User Manual. Austin, Texas, May 2001.
- [2] Bishop, Robert. National Instruments Learning with LabVIEW 7 Express. Upper Saddle River, NJ: Pearson Education, Inc., 2004.
- [3] LabVIEW™ Database Connectivity Toolset v1.0
- [4] Craig, J.J. Introduction to Robotics: Mechanics and Control. 3<sup>rd</sup> Edition. Upper Saddle River, NJ: Pearson Prentice Hall, 2005.
- [5] Hurst, Jeffrey W., and Mortimer, James W. Laboratory Robotics: A Guide to Planning, Programming, and Applications. New York, N.Y.: VCH Publishers, Inc. 1987.
- [6] Niku, Saeed B. Introduction to Robotics: Analysis, Systems, Applications. Upper Saddle River, NY: Prentice Hall, 2001.
- [7] B. Mihura, *LabVIEW for Data Acquisition*, Upper Saddle River, New Jersey, 2001.
- [8] National Instruments Corporation, *Using LabVIEW with TCP/IP and UDP*, 2000.

# Beyond the CMOS Technology: Single-Electron Transistors

Sanjeev Jain- Student M.TECH (ECE), Vaish College of Engineering, Rohtak  
 Payal Verma- Student M.TECH(ECE), Vaish College of Engineering, Rohtak  
 Email:jainmtechb4u@gmail.com,payal\_11232003@yahoo.co.in

**Abstract-**Current microelectronics technology relies on continued shrinkage of transistor features. At a certain point in the miniaturization drive, conventional transistors will behave differently with different current conduction mechanisms. This paper sheds a light on how electrical Conduction takes place in a nanoscale transistor. Single-electron transistors (SET's) are often discussed as elements of nanometer scale electronic circuits because they can be made very small and they can detect the motion of individual electrons. However, SET's have low voltage gain, high output impedances, and are sensitive to random background charges. This makes it unlikely that single-electron transistors would ever replace field-effect transistors (FET's) in applications where large voltage gain or low output impedance is necessary. The most promising applications for SET's are charge-sensing applications such as the readout of few electron memories, the readout of charge-coupled devices, and precision charge measurements in metrology.

## I. INTRODUCTON

Continued miniaturization of metal-oxide-semiconductor field-effect transistor (MOSFET) technology is one of the driving forces behind the electronics industry. The semiconductor industry association has predicted that by the year 2016, the gate length will be reduced to as small as 13 nm [1]. At such small geometry, random dopant distribution in the substrate and charge impurity in the oxide are likely to cause the channel connecting the source and the drain to be irregular. This spatial variation in electrical characteristic result in regions of high and low electrical conductivity. Low conductivity region forms a barrier to charge transport. Early experiments have shown the presence of tunnel barriers along narrow conducting channel [2]. If two closely-spaced tunnel barriers are present along the channel, the confined region between the barriers can be considered as a dot or a reservoir of charges, provided the channel is populated with charges. The principle of electrical conduction of nanometrescale transistors differs from that of micrometre-scale transistors which are the building block of the current complementary MOS circuits.

In micro-scale transistors, charge carriers are transported via the inversion layer along the channel. But in a nano-scale transistor, where a single dot is placed between source and drain, the transfer of individual electrons through the dot is possible via a nearby gate electrode; the structure is now known as a single-electron transistor [3].

## II. DEVICE STRUCTURE

The structure investigated is one of gated double-barrier systems in which the source and drain regions are separated by two tunnel barriers with a small region of no larger than 100 nm, usually called a dot, in between. A nearby electrode (gate)

controls the electrostatic potential of the dot, see Fig.1. The source, drain and dot are doped heavily, beyond the metal-insulator transition. However, conduction between the source and drain is impeded by the presence of the two tunnel barriers. Each barrier is characterized by a tunnel resistance  $R_T$  and a junction capacitance  $C_J$ . When  $R_T$  is small, charge transport is due to the typical drift process.

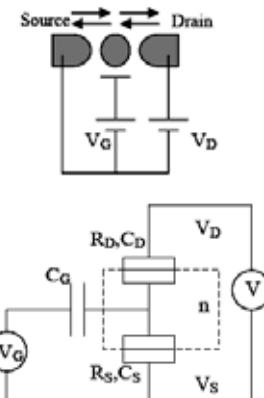


Fig. 1. The single-electron transistor and its equivalent circuit diagram. The dashed box encloses the island with a excess/deficit electrons.

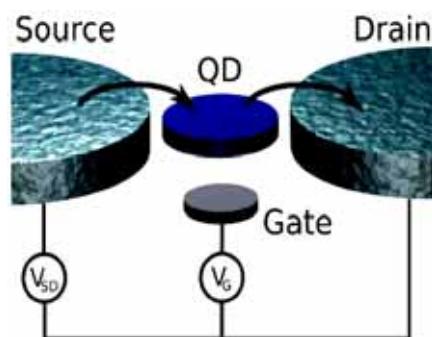


Fig2:Schematic diagram of SET

However, when  $R_T$  is larger than the quantum resistance  $R_k (= h/e^2 \sim 25 \text{ k}\Omega)$ , charge transport across the barrier is possible via tunnelling mechanism. Since only discrete number of charges can be transported, a single negative charge (electron) or positive charge (hole) can be transferred from source to drain, and vice versa.

## III. TYPES OF SINGLE ELECTRON TRANSISTORS

Single-electron transistors can be made using metals, semiconductors, carbon nanotubes, or single molecules. Aluminum SET's made with Al/AlOx/Al tunnel junctions are

the SET's that have been used most often in applications. This kind of SET is used in metrology to measure currents, capacitance, and charge. [8] They are used in astronomical measurements [9] and they have been used to make primary thermometers. [10] However, many fundamental single-electron measurements have been made using GaAs heterostructures. The island of this kind of SET is often called a quantum dot. Quantum dots have been very important in contributing to our understanding of single-electron effects because it is possible to have just one or a few conduction electrons on a quantum dot. The quantum states that the electrons occupy are similar to electron states in an atom and quantum dots are therefore sometimes called artificial atoms. The energy necessary to add an electron to a quantum dot depends not just on the electrostatic energy of Eq. 2 but also on the quantum confinement energy and the magnetic energy associated with the spin of the electron states. By measuring the current that flows thorough a quantum dot as a function of the gate voltage, magnetic field, and temperature allows one understand the quantum states of the dot in quite some detail. [11]

The SET's described so far are all relatively large and have to be measured at low temperature, typically below 100 mK. For higher temperature operation, the SET's have to be made smaller. Ono et al. [23] used a technique called pattern dependent oxidation (PADOX) to make small silicon SET's. These SET's had junction capacitances of about 1 aF and a charging energy of 20 meV. The silicon SET's have the distinction of being the smallest SET's that have been incorporated into circuits involving more than one transistor. Specifically, Ono et al. constructed an inverter that operated at 27 K. Postma et al. [24] made a SET that operates at room temperature by using an AFM to buckle a metallic carbon nanotube in two places. The tube buckles much the same way as a drinking straw buckles when it is bent too far. Using this technique, a 25 nm section of the nanotube between the buckles was used as the island of the SET and a conducting substrate was used as the gate. The total capacitance achievable in this case is also about 1 aF. Pashkin et al. [2] used e-beam lithography to fabricate a SET with an aluminum island that had a diameter of only 2 nm. This SET had junction capacitances of 0.7 aF, a charging energy of 115 meV, and operated at room temperature. SET's have also been made by placing just a single molecule between closely spaced electrodes. Park et al. [5] built a SET by placing a C60 molecule between electrodes spaced 1.4 nm apart. The total capacitance of the C60 molecule in this configuration was about 0.3 aF. Individual molecules containing a Co ion bonded to polypyridyl ligands were also placed between electrodes only 1-2 nm apart to fabricate a SET. [26] In similar work, Liang et al. [17] placed a single divanadium molecule between closely spaced electrodes to make a SET.

In the last two experiments, the Kondo effect was observed as well as the Coulomb blockade. The charging energy in the molecular devices was above 100 meV.

One of the conclusions that can be drawn from this review of SET devices is that small SET's can be made out of variety of materials. Single electron transistors with a total capacitance of about 1 aF were made with aluminum, silicon, carbon nanotubes and individual molecules. It seems unlikely that SET's with capacitances smaller than the capacitances of the molecular devices can be made. This sets a lower limit on the smallest capacitances that can be achieved at about 0.1 aF. Achieving small capacitances such as this has been a goal of many groups working on SET's. However, while some of the device characteristics improve as a SET is made smaller, some of the device characteristics get worse as SET's are made smaller. For some applications, the single molecule SET's are too small to be useful. As SET's are made smaller, there is an increase in the operating temperature, the operating frequency, and the device packing density. These are desirable consequences of the shrinking of SET devices. The undesirable consequences of the shrinking of SET's are that the electric fields increase, the current densities increase, the operating voltage increases, the energy dissipated per switching event increases, and the power dissipated per unit area increases, the voltage gain decreases, the charge gain decreases, and the number of Coulomb oscillations that can be observed decrease.

#### IV. APPLICATIONS

Because of its small size, low energy consumption and high sensitivity, SET has found many applications in many areas. What's most exciting is the potential to fabricate them in large scale and use them as common units in modern computer and electronic industry.

##### **Single electron memory:**

Scientists have long been endeavored to enhance the capacity of memory devices. If single electron memory can be realized, the memory capacity is possible to reach its utmost limit. SET can be used as memory cell since the state of Coulomb island can be changed by the existence of one electron. Chou and Chan [5] first pointed out the possibility of using SET as memories in which information is stored as the presence or absence of a single electron on the cluster. They fabricated a SET by embedding one or several nano Si powder in a thin insulating layer of SiO<sub>2</sub>, then arranging the source and drain as well as gate around this Coulomb island. The read/write time of Chan's structure is about 20ns, lifetime is more than 109 cycles, and retention time (during which the electron trapped in the island will not leak out) can be several days to several weeks.

These parameters would satisfy the standards of computer industry, so SET can be developed to be a candidate of basic computer units. If a SET stands for one bit, then an array of 4~7 SETs will be substantial to memorize different states. The properties of the memory unit composed of SETs are far more advantageous than that of CMOS. But the disadvantage is the practical difficulty in fabrication. When the time comes for the large scale integration of SETs to form logic gates, the full advantages of single electron memory will show. This is the threshold of quantum computing.

#### *High sensitivity electrometr:*

The most advanced practical application currently for SETs is probably the extremely precise solid-state electrometers (a device used to measure charge). The SET electrometer is operated by capacitively coupling the external charge source to be measured to the gate.

Changes in the SET source-drain current are then measured. Since the amplification coefficient is very big, this device can be used to measure very small change of current. Experiments showed that if there is a charge change of  $e/2$  on the gate, the current through the Coulomb island is about  $10^9 e/sec$ . This sensitivity is many orders of magnitude better than common electrometers made by MOSFET. SETs have already been used in metrological applications as well as a tool for imaging localized individual changes in semiconductors. Recent demonstration of single photon detection [6] and rf operation [7] of SETs make them exciting for new applications ranging from astronomy to quantum computer read-out circuitry.

The SET electrometer is in principle not limited to the detection of charge sites on a surface, but can also be applicable to a wide range of sensitive chemical signal transduction events as well. For example, the gate can be made coupling with some molecules, thus can measure other chemical properties during the process.

However, as Lewis K M etc. pointed out [8], SETs electrometer must be designed with care. If the device under test has a large capacitance, it is not advantageous to use SETs as an electrometer. Since for a typical SET,  $C_{SET} < 1mF$ , the suppression factor becomes unacceptable when the macroscopic device has a capacitance in the pF or nF range.

Therefore, SET amplifiers are not currently used for measuring real macroscopic devices. Other low-capacitance electrometers such as a recently proposed quantum point contact electrometer also suffer from a similar capacitance mismatch problem. But it is believed that if the capacitance mismatch can be solved efficiently, SETs may find many new ultra low-noise analog applications.

#### **Microwave detection**

If a SET is attacked black body radiation, the photon-aided tunneling will affect the charge transfer of the system. Experiments show that the electric character of the system will be changed even by a tiny amount of radiation. The sensitivity of this equipment is about 100 times higher than the current best thermal radiation detector.

### V. MAIN PROBLEMS FACING THE APPLICATION OF SET

#### *a) Integration of SETs in a large scale*

As has been mention above, to use SETs at room temperature, large quantities of monodispersed nanoparticles less than 10nm in diameter must be synthesized. It is very hard to fabricate large quantities of SETs by traditional optical lithography and semiconducting process. Chemical self-assembly has the potential to solve this problem.

This method adopts the metal-organic precursors and deposits nano clusters on the substrate. But the position cannot be decisively control and it's not a mature technology.

The large quantity integration of SETs depends greatly on the development of semiconducting industry of nanotech.

#### *b) Linking SETs with the outside environment*

Methods must be developed for connecting the individual structures into patterns which function as logic circuits, and these circuits must be arranged into larger 2D patterns.

There are two ideas. One is by doping, that is, integrating SET as well as related equipments with the existed MOSFET, this is attractive because it can increase the integrating density. The other is to give up linking by wire, instead utilizing the static electronic force between the basic clusters to form a circuit linked by clusters, which is called quantum cellular automata (QCA). Many have tried to use carbon nanotube[12] as leads between a serial of insulating nanoclusters. The state of "0" and "1" can be given by polarization direction affected by applied voltage. Complex analog circuits can be made by QCA. The advantage of QCA is its fast information transfer velocity between cells (almost near optic velocity) via electrostatic interactions only, no wire is needed between arrays and the size of each cell can be as small as 2.5nm, this made them very suitable for high density memory and the next generation quantum computer.

Fig. 4 Nanoparticle-insulator structures proposed in the wireless computing schemes of Korotkov (top) and Lent (bottom). The circles represent quantum dots, the lines are insulating spacers.[5]

### VI. CONCLUSION

Single-electron transistors are the most sensitive charge-measuring devices presently available. They have become an important tool in the field of fundamental measurements. The fact that most SET's only perform at low temperature is not seen as a disadvantage for fundamental measurements because these measurements are often performed at low temperature anyway to reduce noise. In fact, very low temperature operation (less than 100 mK) is seen as an advantage because many semiconducting devices don't work in this temperature range. However for mass-market applications, room temperature operation is necessary. The SET's that operate at room temperature have the problems of low gain, high output impedance, and background charges. No room temperature SET logic or memory scheme is now widely accepted as being practical. The most promising room temperature applications for SET's are in charge sensing circuits where the problems of low gain, high output impedance, and background charges can be solved by integrating SET's with field-effect transistors.

### VII. REFERENCES

- [1]. Lithographically-defined gate length. See <http://public.itrs.net/>
- [2]. J. H. F. Scott-Thomas, S. B. Field, M. A. Kastner, H.I. Smith, and D. A. Antoniadis, "Conductance oscillation periodic in the density of a one-dimensional electron gas", Phys. Rev. Lett. 62, 583 (1989).
- [3]. M. A. Kastner, "The single-electron transistor", Rev. Mod. Phys. 64, 849 (1992).
- [4]. D. V. Averin and K. K. Likharev, in Mesoscopic Phenomena in Solids, edited by B. L. Altshuler, P. ALee, and R. A. Webb (Elsevier Science, North Holland, 1991), p. 173.

- [5]. G.-L. Ingold and Yu. V. Nazarov, "Charge tunnelling rates in ultrasmall junctions", in Single Charge Tunnelling – Coulomb Blockade Phenomena in Nanostructures, NATO ASI Series B, Edited by H. Grabert and M. H. Devoret (Plenum, New York, 1992).
- [6]. M. Amman, R. Wilkins, E. Ben-Jacob, P. D. Maker, and R. C. Jaklevic, "Analytic solution for the current-voltage characteristic of two mesoscopic tunnel junctions coupled in series", Phys. Rev. B 43, 1146(1991). 1222 IEEE ICIT'02, Bangkok, THAILAND
- [7]. For an overview of single-electron devices and their applications, see: K. K. Likharev, Proceedings of the IEEE 87 p.606 (1999).
- [8]. Yu. A. Pashkin, Y. Nakamura, and J.S. Tsai, Appl. Phys. Lett. 76 2256 (2000).
- [9]. Yukinori Ono, Yasuo Takahashi, Kenji Yamazaki, Masao Nagase, Hideo Namatsu, Kenji Kurihara, and Katsumi Murase, Appl. Phys. Lett. 76 p. 3121 (2000).
- [10]. H. W. Ch. Postma, T.F. Teepen, Z. Yao, M. Grifoni, C. Dekker, Science, 293 p. 76 (2001).
- [11]. H. Park, J. Park, A. K. L. Lim, E. H. Anderson, A. P. Alivisatos and P. L. McEuen, Nature 407 p. 57 (2000).
- [12]. Jiwoong Park, Abhay N. Pasupathy, Jonas I. Goldsmith, Connie Chang, Yuval Yaish, Jason R. Petta, Marie Rinkoski, James P. Sethna, Héctor D. Abruna, Paul L. McEuen, and Daniel C. Ralph, Nature 417 p. 722 (2002).
- [13]. Wenjie Liang, M. P. Shores, M. Bockrath, J. R. Long, Hongkun Park, Nature 417 p. 725 (2002).
- [14]. N. M. Zimmerman and M. W. Keller, Electrical Metrology with Single Electrons", to be published in IOP J. Phys.
- [15]. T. Stevenson, A. Aassime, P. Delsing, R. Schoelkopf, K. Segall, C. Stahle, IEEE Transactions on Applied Superconductivity, Vol. 11. No. 1, pp. 692-695, March 2001.
- [16]. K. Gloos, R. S. Poikolainen, and J. P. Pekola, Appl. Phys. Lett. 77, 2915 (2000).
- [17]. L. P. Kouwenhoven, D. G. Austing, S. Tarucha, Reports on Progress in Physics 64pp. 701-736 (2001).
- [18]. C. P. Heij and P. Hadley, Review of Scientific Instruments 73 pp. 491-492 (2002).
- [19]. J. R. Tucker, J. Appl. Phys. 72 4399(1992).
- [20]. G. Lientschnig, I. Weymann, and P. Hadley, submitted to Jap. J. Appl. Phys.
- [21]. For an overview of single-electron devices and their applications, see: K. K. Likharev, Proceedings of the IEEE 87 p. 606 (1999).
- [22]. Yu. A. Pashkin, Y. Nakamura, and J.S. Tsai, Appl. Phys. Lett. 76 2256 (2000).
- [23]. Yukinori Ono, Yasuo Takahashi, Kenji Yamazaki, Masao Nagase, Hideo Namatsu, Kenji Kurihara, and Katsumi Murase, Appl. Phys. Lett. 76 p. 3121 (2000).
- [24]. H. W. Ch. Postma, T.F. Teepen, Z. Yao, M. Grifoni, C. Dekker, Science, 293 p. 76 (2001).
- [25]. H. Park, J. Park, A. K. L. Lim, E. H. Anderson, A. P. Alivisatos and P. L. McEuen, Nature 407 p. 57 (2000).
- [26]. Jiwoong Park, Abhay N. Pasupathy, Jonas I. Goldsmith, Connie Chang, Yuval Yaish, Jason R. Petta, Marie Rinkoski, James P. Sethna, Héctor D. Abruna, Paul L. McEuen, and Daniel C. Ralph, Nature 417p. 722 (2002).
- [27]. Wenjie Liang, M. P. Shores, M. Bockrath, J. R. Long, Hongkun Park, Nature 417 p. 725 (2002).
- [28]. N. M. Zimmerman and M. W. Keller, "Electrical Metrology with Single Electrons", to be published in IOP J. Phys.
- [29]. T. Stevenson, A. Aassime, P. Delsing, R. Schoelkopf, K. Segall, C. Stahle, IEEE Transactions on Applied Superconductivity, Vol. 11. No. 1, pp. 692-695, March 2001.
- [30]. K. Gloos, R. S. Poikolainen, and J. P. Pekola, Appl. Phys. Lett. 77, 2915 (2000).
- [31]. L. P. Kouwenhoven, D. G. Austing, S. Tarucha, Reports on Progress in Physics 64 pp. 701-736 (2001).
- [32]. C. P. Heij and P. Hadley, Review of Scientific Instruments 73 pp. 491-492 (2002).
- [33]. J. R. Tucker, J. Appl. Phys. 72 4399 (1992).
- [34]. G. Lientschnig, I. Weymann, and P. Hadley submitted to Jap. J. Appl. Phys.

# Development of Microcontroller Based Missile Launcher for Anti-Terrorism Applications

C Ghanshyam, Naveen Kumar and Gaurav Puri  
CSIO, Sector-30, Chandigarh, (Council of Scientific and Industrial Research, New Delhi)

**Abstract-**The terrorist threat against free world is serious and enduring. The daunting task lies in the creation of an arsenal of counterterrorism technologies that are practicable and affordable. The paper aims at developing a system capable of electrical and environmental monitoring of a Global System for Mobile (GSM) based missile system. This paper discusses the method of launching the missile through a GSM based mobile system. This system basically consists of a cell phone, dual tone multi frequency decoder, AT89C51 microcontroller and an array of relays. This paper deliberates on the design aspects, practical issues involved in the development and some simulated results in MATLAB.

**Key Words:** GSM, Microcontroller.

## I. INTRODUCTION

Over the last quarter of a century increasing attention has had to be paid for development of advanced weapon based technology for counterterrorism. Fortunately significant progress has been made in applicability of remote sensing in these fields and but there is a strongly perceived need for improved data communication between two stations, i.e. one which are in particular area, in which all the missiles are connected and 2<sup>nd</sup> one , which controls all the functioning of the circuit through locations by mobile. In this work, a call is made with a Dual Tone Multi Frequency Tone (DTMF) compatible phone to the mobile phone attached to the circuitry; in the course of the call, if any button is pressed control corresponding to the button pressed is heard at the other end of the call, which is DTMF. The tone is received with the help of phone stacked on the circuitry. The received tone is processed by the 89C51 microcontroller with the help of DTMF decoder MT8870, the decoder decodes the DTMF tone in to its equivalent binary digit and this binary number is sent to the microcontroller, the microcontroller is pre-programmed to take a decision for any given input and output its decision to relay circuitry in order to launch the missile. DTMF signaling is used for telephone signaling over the line in the voice frequency band to the call switching center. The version of DTMF used for telephone dialing is known as touch tone. DTMF assigns a specific frequency (consisting of two separate tones) to each key so that it can easily be identified by the electronic circuit. The signal generated by the DTMF encoder is the direct algebraic summation, in real time of the amplitudes of two sine (cosine) waves of different frequencies, i.e., pressing 5 will send a tone made by adding 1336 Hz and 770 Hz to the other end of the mobile. Use of a mobile phone for control provides the advantage of robust control, working range as large as the coverage area of the service provider, no interference with other controllers and up to twelve controls. Each DTMF tone having a unique frequency, so using 12 keys with their proper permutation and combination, large number of missiles can be controlled and simultaneously incorporating security features.

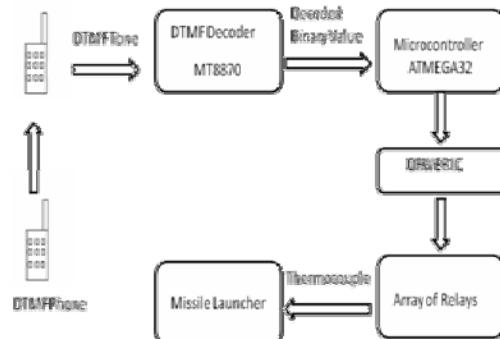


Fig. 1 Schematic diagram of mobile operated Anti-Missile Launcher

## II. System description

The system consists of transmitter, receiver, control and timing section as shown by schematic Diagram in Fig.1.

### A. Transmitter Section

We use any DTMF compatible phone to transmit the signal.

### Receiver Section

This section consists of 8051(89C51) microcontroller, MT8870 DTMF decoder and auto answer compatible cellular phone. The signal coming from the cell phone attached to circuitry is fed to decoder with the help of headphone. The wires of headphone are insulated with a coating and need to be removed by burning with matchstick.

### DTMF Decoder (MT8870D)

The MT8870D is a complete DTMF receiver integrating both the band split filter and digital decoder functions. The filter section works by applying the DTMF signal to the inputs of two sixth-order switched capacitor band pass filters, the bandwidths of which correspond to the low and high group frequencies. Each filter output is followed by a single order switched capacitor filter section which smooths the signals prior to limiting. Separation of the low-group and high group tones is achieved by applying limiting. Limiting is performed by high-gain comparators which are provided with hysteresis to prevent detection of unwanted low-level signals. The outputs of the comparators provide full rail logic swings at the frequencies of the incoming DTMF signals. Following the filter section is a decoder employing digital counting techniques to determine the frequencies of the incoming tones and to verify that they correspond to standard DTMF frequencies.

### i. AT(89C51) Microcontroller

The AT89C51 is a low-power, high-performance CMOS 8-bit microcomputer with 4 KB of Flash Programmable and Erasable Read Only Memory (PEROM)[10,11]. The device is manufactured using Atmel's high density nonvolatile memory technology and is compatible with the industry standard MCS-51Ô instruction set and pin out. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer.

### iii. IC 7805 Voltage Regulator

The KA78XX/KA78XXA [10] series of three-terminal positive regulator are available in the TO-220/D-PAK package and with several fixed output voltages, making them useful in a wide range of applications.7805 in the circuit provides an output of 5V from an input power supply of 12 V.

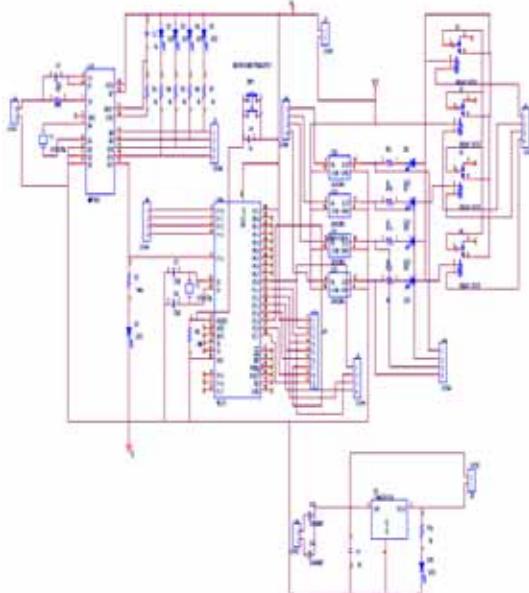


Fig. 2 Circuit Diagram of the Hardware

### III. Circuit Description

Fig 2 shows the circuit diagram with important components as DTMF decoder, 8051 microcontroller, ULN 2803 and relays. When the input signal is given at pin1 (IN+) and pin2 (IN-) of MT8870 DTMF decoder, the decoder decodes it into its binary equivalent which is displayed by LEDs. Table 1 shows the DTMF data output table of MT8870. Q1 through Q4 outputs of the DTMF decoder are connected to port1 of AT89C51 microcontroller. Output from port2 of microcontroller is fed to the array of relays through ULN2803, which magnetizes and activates the detonator to launch the missile. The activated relays are indicated by corresponding LEDs.

### IV. SOFTWARE DESCRIPTION

The software is written in “C” language and compiled using Keil MuVision. The source code is converted into hex code by the compiler and is burned into the 89C51 microcontroller. MATLAB R2008a is used for simulation of DTMF tone and spectral analysis of DTMF tone.

### V. V. Dtmf tone (dual tone muti frequency)

A DTMF tone consists of sum of two sinusoidal waveforms superimposed together from a set of seven standardized frequencies [6]. These standardized frequencies consist of two mutually exclusive frequency groups: the low frequency group and the high frequency group [5, 6]. These frequencies were chosen to prevent any harmonics from being incorrectly detected by the receiver as some other DTMF frequency. The keypad of a push-button telephone set is arranged into a  $3 \times 4$  matrix that selects the appropriate pair of frequencies to be transmitted[7]; four low frequency tones ( $< 1$  kHz) are assigned to rows, while three high frequency tones ( $> 1$  kHz) [8] are assigned to columns as shown in Table 1[9]. This allows a touch tone keypad to have up to twelve unique DTMF tones [6].

Table1. Touch tone keypad corresponding to DTMF tone frequencies.

Upper Band ( $> 1$ kHz)				
	1209 Hz	1336 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
Lower Band ( $< 1$ kHz)	4	5	6	B
770 Hz	7	8	9	C
852 Hz	*	0	#	D
941Hz *				

For example, in order to generate the DTMF tone for "1", you mix a pure 697 Hz signal with a pure 1209 Hz signal, like so:

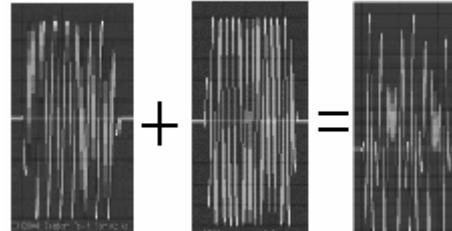


Fig. 3- 697Hz + 1209 Hz = DTMF Tone '1'

Two pure sinusoids combine to form DTMF Tone '1' as shown if Fig.3 above.

### VI. SIMULATIONS ON DTMF, TONE RESULTS AND DISCUSSION

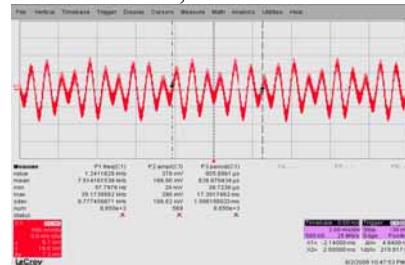


Fig. 4- shows DTMF Plot for Tone '1' LeCroy Oscilloscope



Fig. 5

The above Fig.5 shows simulation[4] DTMF decoding and analysis done in MATLAB shows '1' and '4' has same high frequency group which can be clearly seen from the spectrogram while low frequency changes in the spectrogram.

## VII. SPECTRAL ANALYSIS OF DTMF TONE

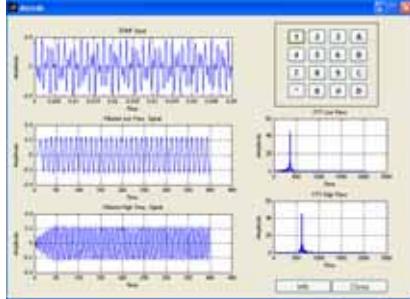


Fig. 6

The DTMF decoder simulink [2] model decodes the time domain signal into individual sinusoidal components as shown in Fig.6.

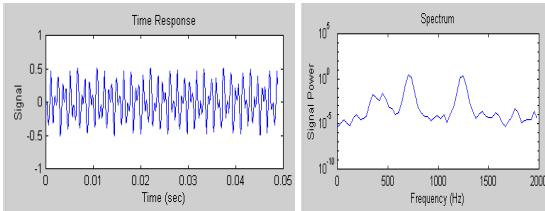


Fig. 7

The above figure 7 shows that DTMF Tone '1' is made up of sum of two sinusoids 1336Hz and 770Hz.

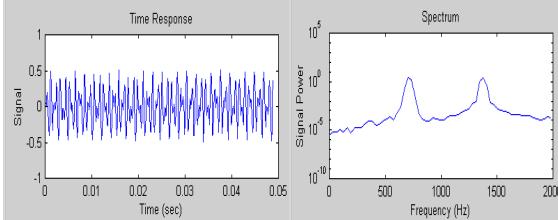


Fig. 8

The above figure8 shows that DTMF Tone '2' is made up of sum of two sinusoids 1336Hz and 697Hz.

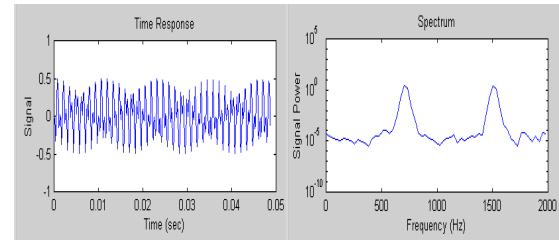


Fig. 9

The above figure 9 shows that DTMF Tone '3' is made up of sum of two sinusoids 1477Hz and 697Hz.

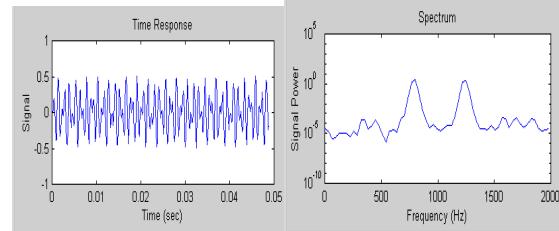


Fig. 10

The above figure 10 shows that DTMF Tone '4' is made up of sum of two sinusoids 1209Hz and 770Hz.

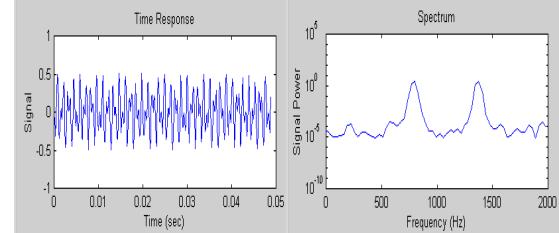


Fig. 11

The above figure 11 shows that DTMF Tone '5' is made up of sum of two sinusoids 1336Hz and 852Hz.

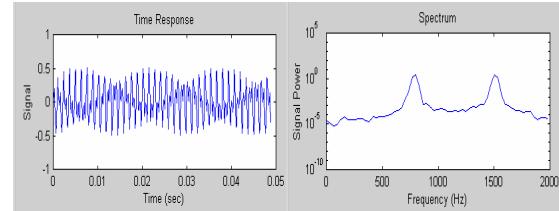


Fig. 12

The above figure12 shows that DTMF Tone '6' is made up of sum of two sinusoids 1477Hz and 770Hz.

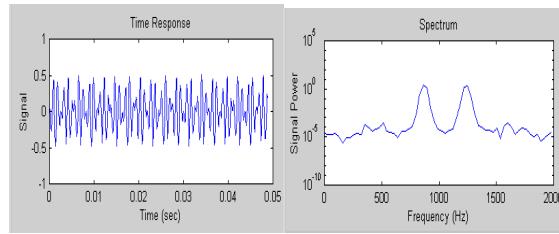


Fig. 13

The above figure13 shows that DTMF Tone '7' is made up of sum of two sinusoids 1209Hz and 852Hz.

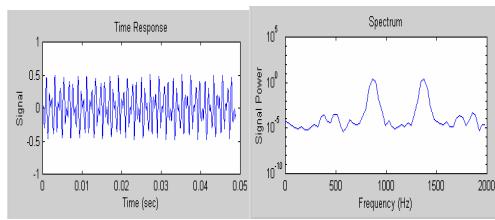


Fig. 14

The above figure 14 shows that DTMF Tone ‘8’ is made up of sum of two sinusoids 1336Hz and 852Hz.

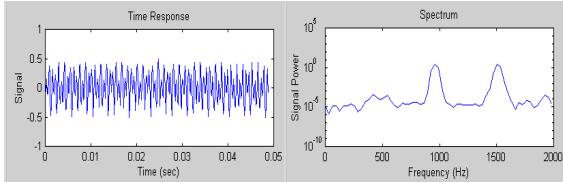


Fig. 15

The above figure 15 shows that DTMF Tone ‘9’ is made up of sum of two sinusoids 1477Hz and 852Hz.

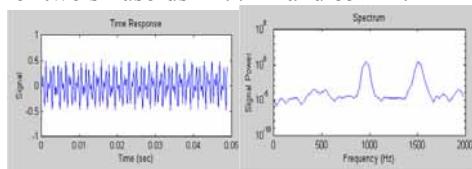


Fig. 16

The above figure 16 shows that DTMF Tone ‘#’ is made up of sum of two sinusoids 1477Hz and 941Hz.

### VIII. CONCLUSION

We have described a novel GSM based Anti Missile Communication system that is able to launch a missile from any location of the world using DTMF compatible handset. The main focus of this article is on the development and convenient control of a tele-operated device. The problem of incorporating security features can be handled by modifying the code.

### IX REFERENCES

- [1]. Stuart McGarry, Simulink model of “DTMF Generator and receiver”.
- [2]. Randolph Sequera, Simulink model of “DTMF Decoder”
- [3]. [www.mathworks.com](http://www.mathworks.com), “DTMF Generator and Receiver”
- [4]. Rahul Garg, Simulink Model of “DTMF Filtering and Noise Simulator”
- [5]. M. D. Felder, J. C. Mason, and B. L. Evans, “Efficient Dual-tone Multifrequency Detection using the Nonuniform Discrete Fourier Transform”, IEEE Signal Processing Letters, Vol. 5, No. 7, July 98, pp.
- [6]. Miloš Trajković and Dušan Radović, performance Analysis of the DTMF Detector based on the Goertzel’s Algorithm”, in Proc. 14<sup>th</sup> Telecommunications Forum, TELFOR 2006, Serbia Belgrade, November 2006.
- [7]. A. M. Shatnawi, A. I. Abu-El-Haija, and A. M. Elabdalla, “A Digital Receiver for Dual Tone Multifrequency (DTMF) Signals”, in Proc. Technology Conference, Ottawa, Canada, May 1997, pp. 997-1002.
- [8]. M. J. Park, S. J. Lee, and D. H. Yoon, “Signal Detection and Analysis of DTMF Receiver with Quick Fourier Transform”, in Proc. 30th Annual Conference of IEEE Industrial Electronics Society, IECON 2004, Vol. 3, November 2004, pp. 2058-2064.
- [9]. R. A. Valenzuela, “Efficient DSP based detection of DTMF Tones”, in Proc. Global Telecommunications Conference, GLOBECOM 1990, Vol. 3, December 1990, pp. 1717-1721.
- [10]. [www.datasheetcatalogue.com](http://www.datasheetcatalogue.com)
- [11]. “The 8051 Microcontroller and Embedded Systems”, Muhammad Ali Mazidi, Janice Mazidi.

# Design and Construction of Programmable Medium Voltage Pulse Generator for the Preservation of Liquid/ Semi-Liquid Food Items

C Ghanshyam, Saurab Saini, Nilotpal, K Khanikar, Vijay Kumar Verma and Garima Bajwa  
 CSIO, Sec-30, Chandigarh, (Council of Scientific and Industrial Research, New Delhi)

**Abstract-** Technological advances in food processing industry have evolved various instruments, ranging from simple to complex equipments. These are either processing equipment or preserving equipment. Preservation of food item calls for new techniques; which involve various thermal and non-thermal methods. Pulsed Electric Field (PEF), which is a non-thermal method offers freshness, flavor and nutritional value and is used to improve the shelf-life of liquid food. In this technique, a short pulse is given for a short duration for effective inactivation of microbes. This paper elaborates a method to generate medium voltage pulses using power MOSFET and switching circuit. This microcontroller based pulse generator circuit controls and varies the amplitude and pulse width of the generated pulse. Medium voltage pulses up to 100V and 20 ms have been generated, tested and verified for inactivation of microorganism. A user interface has been provided for defined settings and monitoring the current parameters. Further, Data Acquisition System is used to acquire the control variable for amplitude and pulse width variation and to make this system PC-based/ online. Virtual Instrumentation could be used for data acquisition and control purpose.

**Key Words-** *Data Acquisition, Food Processing, Embedded System, Virtual Instrumentation.*

## I. INTRODUCTION

Preserving food items yields new methods and techniques in food processing industry. In past drying, salting, smoking, pickling, canning, fermenting etc. were tried to increase the shelf life of foods. Technological advances evolved various new thermal and non-thermal methods, like; cooling, freezing, and application of electric field. In all these methods, one thing is common and that is in-activation of micro-organism to preserve food. One latest and advanced method is to apply electric field in the form short duration voltage pulse to inactivate the microorganism. PEF produces products with slightly different properties than conventional pasteurization treatments. Most enzymes are not affected by PEF; the fact that the maximum temperature reached is lower than in thermal pasteurization. Additionally, some of the flavors associated with the raw material are not destroyed by PEF. The lack of heat treatment makes PEF somewhat comparable to irradiation as a treatment. When exposed to high electrical field pulses, cell membranes develop pores either by enlargement of existing pores or by creation of new ones. These pores may be permanent or temporary, depending on the condition of treatment. The pores increase membrane permeability, allowing loss of cell contents or intrusion of surrounding media, either of which can cause cell death. Thus, electroporation of cell membranes caused by PEF helps to increase the shelf life of liquid foods at comparatively low

treatment temperatures, thereby, preserving freshness, flavor and texture etc.

Food processing methods used in food industries try to create as many hurdles to bacterial growth and survival as possible. Bacterial growth increases rapidly as soon as the containers containing these foods are opened and this is the most favorable to bacterial growth. In this paper, it is proposed to extend the life expectancy of liquid food and commercial/processed juices available in the market using electrical voltage pulses. High voltage, short duration pulsed electric field application for processing liquid/ semi-liquid food has the advantage that the heating effect is significantly less compared to conventional “thermal” processes and more desirable than chemical processes. A range of electric field values have been proposed from a few kV/cm to tens of kV/cm for this application.

Medium voltage pulses are selected because of following reasons:

- A high voltage pulse introduces some drawbacks in the food itself. In order to produce high magnitude electric fields, large, costly power supplies are required. Also, high magnitude electric fields cause a significant increase in temperature, having an adverse affect on the flavor of the food.

- Hence, alternative are used, like using medium voltage pulses of very short duration (nano-second pulses).

Pulse Generator is the heart of such type of food preserving system. In this paper design of a Medium voltage pulse generator has been discussed with design aspects, problems and possible issues that come across with this research work.

## II. FUNDAMENTALS OF MICROORGANISM IN-ACTIVATION

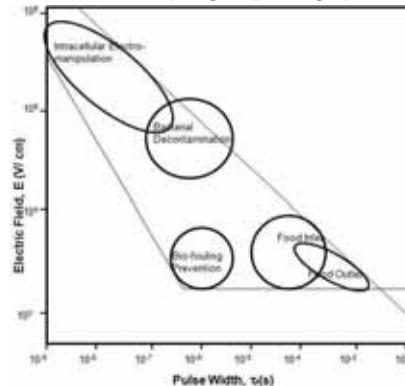


Fig. 1 Range of electric field and pulse width

Fig. 1 shows the optimum pulse width and electric field range

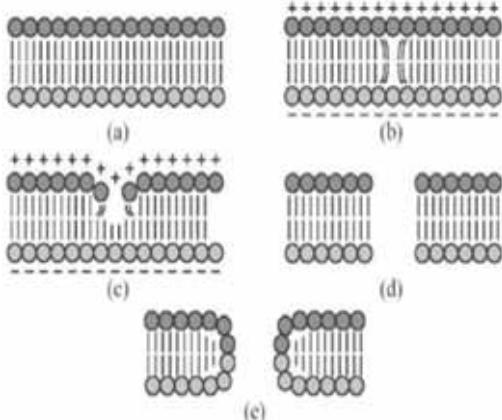


Fig. 2 Process of pore formation (a) normal cell membrane, (b) a cell excited short electrical pulse resulting in irregular molecular structure (c) the membrane being melted (d) the cell with a temporary hydrophobic pore and (e) the cell with a membrane restructuring

### III. DESIGN OF MEDIUM VOLTAGE PULSE GENERATOR

The Medium Voltage Pulse generator can be considered as an assembly of following three parts:

- a. Power Supply
- b. Switching/ Timer circuit
- c. Medium voltage pulse generator circuit consisting of Power MOSFET which is connected to the load

One regulated power supply has been designed with all desired voltage levels, like +5V, ±15V, +100V using various voltage regulators with appropriate ratings and standards.

Timer/ Switching circuit has been designed using ATMEL's microcontroller AT89C51 which generates short duration pulses but these pulses don't have enough current capability to drive the further circuitry.

Generation of Short duration pulses of the order of few  $\mu$ s to nS requires fast power MOSFET to switch medium voltage pulses. A high current driver was used to switch this MOSFET and this driver also has to be very fast to meet the requirement. The CMOS/TTL input of the MOSFET driver is connected to the pulse generation circuit output. The MOSFET's source is connected to ground and the drain is connected to the negative side of the load. A fast switching diode is placed across the load to reduce ringing on the load.

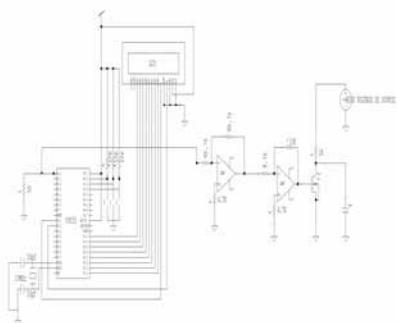


Fig.3 Circuit Diagram of Pulse Generator

### V. DESCRIPTION

The Circuit Diagram of pulse generator is shown in Fig.3. In this, the pulses are drawn from microcontroller.

The two op-amps are used in inverting mode which together act as the driver circuit. The first one is taking its input from the output pin P1.0 of the microcontroller. It is having unity gain. The output of the first inverter is connected to the input of the second inverter. It is having a gain of 2.3. The first inverter circuit is inverting the output voltage signal available at the output pin P1.0. The function of the second inverter circuit is to convert the first inverter circuit output to more than 4 volt (Threshold voltage of the power MOSFET) when the output of P1.0 is high. The driver circuit is used to switch MOSFET which control the high voltage. The MOSFET (IRFZ44N), used in this circuit is able to survive a drain to source breakdown voltage of 55 V and it has a drain current of 49 A (maximum). In addition, the MOSFET has an on static drain source on resistance 17.5mΩ (maximum). The MOSFET will rapidly charge and discharge when the square wave pulse from the driver circuit become to the gate of the MOSFET. It can provide a pulse width of a few micro-second or a continuous dc supply. In addition, the high voltage supply current will pass through a 10Ω resistor which acts as a current limit to protect the MOSFET by damping the voltage during the turn on time. When the output of the second inverter is more than 4 volt, the high voltage supply is switched on and used to charge a 4.7 μF capacitor. The gate of MOSFET should not exceed ±20V which keeps the gate safe between ranges of ±20V. MOSFET is protected by two zener diodes (PHC12). When the MOSFET is in turn off state, the capacitor (4.7 μF) will be charged and store an energy/ high voltage. When the MOSFET is switched on, the capacitor discharges through the low impedance path provided by the MOSFET.

### VI. RESULTS AND DISCUSSION

The resulting circuit is having two modes. One mode is used for continuous pulse generation and the other mode is used for generation of single shot pulse which can be manually triggered. The pulse period and duty cycle are taken as user inputs through three keys which are interfaced with the microcontroller. These input parameters are displayed on an LCD also interfaced with the microcontroller. Various snapshots of the working circuit are shown in Fig.4.

The circuit can produce pulses of minimum pulse width of 40 usec and maximum pulse width of 20msec. The maximum pulse amplitude that can be generated is 50V which is limited by the selection of the power MOSFET (IRFZ44N).

The manually triggered mode can generate pulses of minimum pulse width 50us and maximum amplitude same as stated above. The varying duty cycle and the time period of pulses are shown in Fig.5, Fig.6 and Fig.7.



Fig. 4-Snapshots of the working circuit

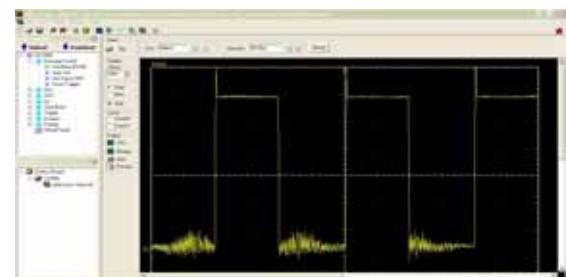


Fig. 7-Snapshot showing pulses of 50% duty cycle and 200 $\mu$ sec period at P1.0

## VII. CONCLUSION

This paper elaborates a method to generate medium voltage pulses using power MOSFET and switching circuit. This microcontroller based pulse generator circuit controls and varies the amplitude and pulse width of the generated pulse. Medium voltage pulses up to 100V and 20 ms have been generated, tested and verified for inactivation of microorganism. A user interface has been provided for defined settings and monitoring the current parameters. It has been observed that, diodes and zener diodes, the use of Triac as the optical isolator between the microcontroller and the operation amplifier circuitry can shield better the microcontroller from the rest of the circuit. This will address both the back current and the back voltage problems by providing a high isolation voltage between the input and the output pins.

An improvised version of the circuit calls for prevention of the back emf and current from damaging the microcontroller. For this, the optical isolator Triac can be used. ; an IR diode and IR detector pair with a high isolation voltage (nearly 5000 volts or more). This also allows only one way communication from the microcontroller to the driving circuit and no back voltage or current is allowed to be propagated to the microcontroller.

## VIII. REFERENCES

- [1]. Design and Construction of a Programmable system for Biological Applications ;Rodamporn S., Beeby S. , Harris N.R. ,Brown A.D. and Chad J.E. ; Proceddiings of Thai BME.
- [2]. A Compact High Voltage Nanosecond Pulse Generator; Drew Campbell, Jason Harper, Vinodhkumar Natham, Funian Xiao, and Raji Sundararajan ; Proc. ESA Annual Meeting on Electrostatics 2008, Paper H3.
- [3]. PIC in practice, A project based approach; D. W. Smith.
- [4]. How to use Intelligent LCDs , Julyan Ilett
- [5]. 8051 Microcontroller and Embedded systems, Muhammed Ali Mazidi, Janice Gillispie Mazidi
- [6]. [www.datasheetcatalogue.com](http://www.datasheetcatalogue.com)

Fig. 5- Snapshot showing pulses of 70% duty cycle and 400 $\mu$ sec period at P1.0

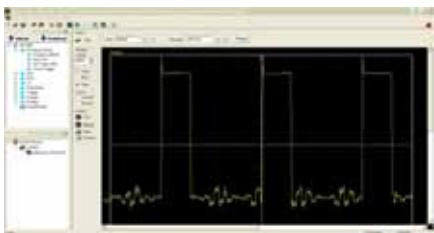


Fig. 6- Snapshot showing pulses of 30% duty cycle and 400 $\mu$ sec period at P1.0

# A Literature Study of Software Testing and the Automated Aspects thereof

Surjeet Singh- Lecturer in Computer Science, G.M.N. (P.G.) College, Ambala Cantt.

Dr. Rakesh Kumar- Reader, Deptt. of Computer Sci. & Application, K.U.K.

**Abstract-** Software testing is the process of executing a program with the intent of finding errors in the code. It is the process of exercising or evaluating a system or system module by manual or automatic means to validate that it satisfies specified requirements or to identify differences between expected & actual outcome. As computer technology advances, systems are getting larger and accomplish more tasks. As a result the job of testing has become ever more imperative as part of the system design and implementation. This paper introduces automatic test data generation. Automatic testing significantly reduces the effort of individual tests. This implies that performing the same test becomes cheaper, or one can do more tests within the same budget and time.

## I. INTRODUCTION

Software testing is any activity aimed at evaluating an attribute or capability of a program or system as well as determining that it meets its required consequences [8]. Although crucial to software quality and widely deployed by programmers and testers, software testing still remains an art, due to imperfect understanding of the ethics of software. The difficulty in software testing stems from the complexity of software. We cannot completely test a program with reasonable complexity. Testing is more than just debugging. The purpose of testing can be quality assurance, verification and validation, or reliability estimation. Testing can be used as a generic metric as well. Correctness testing and reliability testing are two foremost areas of testing. Software testing is a trade-off between budget, time and quality. Software testing is labor-intensive, and therefore expensive, yet heavily used technique to control quality. It is a part of almost every software project. The testing phase of typical projects takes up to 50% of the total project effort, and hence contributes significantly to the project costs. Studies also show that maintenance can consume up to 80% of the cost for the entire software lifecycle, and much of that cost is devoted to testing. Any change in the software can potentially influence the result of a test. For this reason tests have to be repeated often. This is error-prone, boring, time consuming, and expensive.

To assist in program testing various kinds of techniques have been developed. For the testing phase, a black box approach to testing is supported by generating test cases that cover the program's expected functionality [4]. Test data can be automatically generated to support a white box testing [5, 6, 10]. Test data generation in program testing is the process of identifying a set of test data which satisfies given testing criterion. Most of the existing test data generators [1, 2, 3, 9, 11] use symbolic evaluation to derive test data. However, in practical programs this technique frequently requires complex algebraic manipulations, especially in the presence of arrays. In this paper I also present an alternative approach of test data

generation which is based on actual execution of the program under test. Testing is the most common way to increase confidence in the correctness and reliability of software. Rapidly changing software and computing environments present many challenges for successful and efficient testing in practice. Past research in testing of evolving software has resulted in techniques that attempt to automate or partially automate the process. Although few of these techniques have been successfully transferred to practice, existing techniques show promise for use in industry. By combining program analysis, machine learning, and visualization techniques, we can expect significant improvement in the process of testing evolving software that will provide reduction in cost and improvement in quality [7].

## II. BASIC CONCEPTS

Figure 2 explains a typical test data generation system, which consists of program analyzer, path selector and test data generator. The source code is run through the program analyzer, which produces the essential data used by the path selector and the test data generator. The path selector inspects the program data in order to find the paths that lead to high code coverage. The paths are then given as arguments to the test data generator which derives input values that exercise the given paths. The generator may provide the selector with feedback such as information concerning infeasible paths.

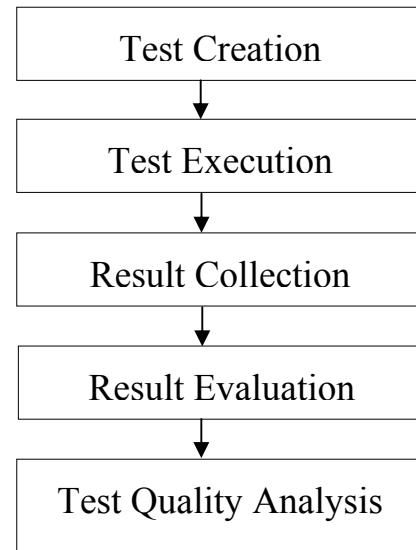


Figure1: The Testing Process.

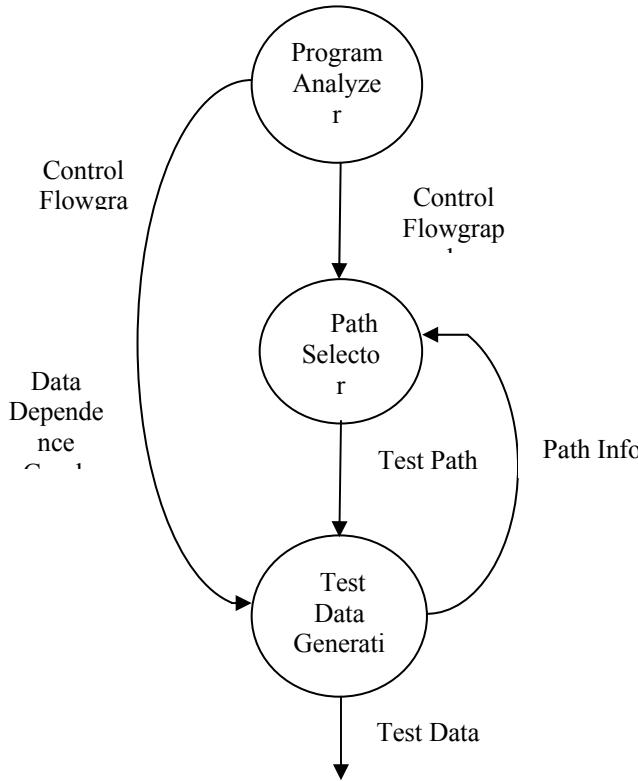


Figure2. Architecture of a Test Data Generator System.

A program P could be considered as a function,  $P: S_i \rightarrow S_o$ , where  $S_i$  is the set of all possible inputs and  $S_o$  is the set of all possible outputs. More formally  $S_i$  is the set of all vectors  $x = (d_1, d_2, d_3, \dots, d_n)$  such that  $d_i \in D_{xi}$  where  $D_{xi}$  is the domain of input variable  $x_i$ . An input variable  $x$  of P is a variable that either appears as an input parameter of P or in an input statement of P, e.g. `read(x)`. Execution of P for a certain input  $x$  is denoted by  $P(x)$ . A control flowgraph of a program P is a directed graph  $G = (N, E, s, e)$  consisting of a set of nodes N and a set of edges  $E = \{(n, m) \mid n, m \in N\}$  connecting the nodes. In each flow graph there is one entry node s and one exit node e.

#### An Automatic Test Data Generation: Problem in Automatic Test Data Generator System

A test data generator system consists of three parts: a program analyzer, a path selector and a test data generator. Let a program P and the unspecific path u, generate input  $x \in S_i$ , so that  $x$  traverses the path u. This means that we can assume to have a program analyzer and a path selector such as in figure 2. The program analyzer provides all information concerning the program's data-dependence graphs, control flow graph etc. In turn the path selector identifies paths for which the test data generator will derive input values. Depending upon the type of generator system paths could either be specific or unspecific. The goal is to find input values that will traverse the paths received from the path selector. This is achieved in two steps. First find the path predicate for the path. Second, solve the path predicate in terms of input variables. The solution will then be a system of (in) equalities describing how input data should be

formed in order to traverse the path. Having such a system we can apply a variety of search methods to come up with a solution.

#### Symbolic execution and Dynamic Execution

There are two types of executions that is symbolic execution and actual execution i.e. the generation occurs either statically or dynamically. Executing a program symbolically means that instead of using actual values variable substitution is used. For instance, let x and y be input variables.

$$x = x + y;$$

$$y = x - y;$$

$$z = x * y;$$

Then the z in the above code will contain  $x * x - y * y$ . The problem with this technique is that it requires ample of computer resources.

#### Random Test Data Generation

There are many approaches to select test cases. One simple approach is called Random Testing (RT), which randomly selects test cases/sequences of events from the input domain (that is, the set of all possible inputs) [12, 13]. The advantages of RT include its low cost, ability to generate numerous test cases automatically, and the generation of test cases in the absence of the software specification and source code. Apart from these, RT brings "randomness" into the testing process. Such randomness can best reflect the chaos of system operational environment; as a result, RT can detect certain failures unable to be revealed by deterministic approaches. All these advantages make RT irreplaceable and so popularly used in industry for revealing software failures [14, 15, 16, 17, 18, 19, 20, 21, 22, 23]. This approach may yield a large number of event sequences that are not legal & hence not executable, wasting valuable resources. Moreover, the test designer has no control over choice of event sequences and they may not have acceptable test coverage. Random testing selects arbitrarily test data from the input domain & then these test data are applied to the program under test. The automatic production of random test data, drawn from uniform distribution, should be the default method by which other systems should be judged [24]. The random generation of tests identifies members of the sub domains arbitrarily with a homogeneous probability which is related to the cardinality of the sub domains. Under these circumstances, the chances of testing a function, whose sub domain has a low cardinality with regard to the domain as a whole, is much reduced. A random number generator generates the test data with no use of feedback from previous tests. The tests are passed to the procedure under test, in the hope that all branches will be traversed [25].

#### Anti Random Testing

In Anti random testing the test cases should be selected to have maximum distance from each other. Parameters & interesting values of the test object are encoded using a binary vector such that each interesting value of every parameter is represented by one or more binary values. Test cases are then selected such that a new test case resides on maximum Hamming distance from the already selected test cases. Anti random testing can be used to select a subset of all possible test cases, while ensuring that they are as far apart as possible.

Moreover it has proved useful in a series of empirical evaluations. Unfortunately, this method basically requires enumeration of the input space & computation of each input vector when used on an arbitrary set of existing test data. This prevents scale-up to large test sets and/or long input vectors.

#### **Adaptive Random Testing:**

Recently, Adaptive Random Testing (ART) is an improvement over Random Testing (RT). It has been introduced to improve the fault detection effectiveness of RT for the situations where failure-causing inputs (that is, program inputs that reveal failures) are clustered together [26, 27]. Such situations do occur frequently in real life programs as reported in [28, 29, 30]. When failure-causing inputs are concentrated in regions (Known as the failure regions [30]), intuitively speaking, and keeping test cases apart shall enhance the effectiveness of RT. Therefore, ART does not just randomly generate but also evenly spreads test cases or it generates fewer duplicate test cases. Studies [26, 31, 32, 33, 34, 35] shows that ART can be very effective in detecting failures when there exist continuous failure regions inside the input domain as compared to RT. Since ART is as simple as RT and preserves certain degree of randomness, ART could be an effective replacement of RT. Random testing is the simplest method of generation techniques. Consider the following piece of code written in C language:

```
void equal(int x, int y)
{
if (a==b)
printf("1");
else
printf("2");
}
```

The probability of exercising the printf("1") statement is 1/n, where n is the maximum integer, since in order to execute this statement variables a and b must be equal. We can easily imagine that generating even more complex structure than integer will give us even worse probability.

#### **Goal-Oriented Test Data Generation**

This approach is much stronger than random generation. In this method the goal is important than the path. Since this method traverse any path to reach to the goal state so it is hard to predict the coverage given a set of goals. Assertion-oriented testing uses the approach of goal-oriented generation. Certain conditions, called assertion are inserted into the actual code and when that particular condition is executed it is supposed to hold, otherwise there is an error either in the actual code or in the assertion code. For example in the following module:

```
void division(int a, int b)
{
Int c;
C=(a+b)*(a-b);
assert(a!=b)
printf("%d",1/c)
}
```

The code is inserted with checking that a!=b. So before executing the printf statement the variable a and b must not be

equal. The goal of assertion-oriented generation is then to find any path to an assertion that does not hold.

#### **III. CONCLUSION**

In this paper the different techniques used to improve the quality of the software through testing have been discussed. The automation process of testing can improve the quality and reduced the risk and time consumed. Hence the cost of developing the quality software can be reduced.

#### **IV. ANNOTATED REFERENCES**

- [1]. Bicevskis J. et al. (1979) "SMOTL-A system to construct samples for data processing program debugging," IEEE Trans. Sofrware Engineering., vol. SE-5, no. 1, pp.60-66.
- [2]. Boyer R., Elspas B., & Levitt K. (1975) *SELECT-A formal system for testing and debugging programs by symbolic execution,*" SIGPLAN Notices, vol. 10, no. 6, pp. 234-245.
- [3]. Clarke L. (1976) *A system to generate test data and symbolically execute Programs.* IEEE Trans. Software Engineering, vol. SE-2, no. 3, pp. 215-222.
- [4]. Cohen, D., M. et al. (1997) *An approach to testing based on combinatorial design.* IEEE Transactions on Software Engineering,23(7) pp437-444.
- [5]. Ferguson, R. & Korel, B. (1996) *The chaining approach for software test data generation.* ACM Transactions on Software Engineering and Methodology, 5(1):pp63-86.
- [6]. Gallagher, M., J., & Narasimhan, V., L. (1997) *A test data generation suite for ada software systems.* IEEE Transaction on Software Engineering, 23(8): pp473-484.
- [7]. Harrold Jean Mary (2008) *Testing Evolving Software: Current Practice and Future Promise.* ISEC'08, Hyderabad, India ACM 978-1-59593-917-3/08/0002. Pp19-22.
- [9]. Hetzel, William C., *The Complete Guide to Software Testing*, 2nd ed. Publication info: Wellesley, Mass. : QED Information Sciences, 1988. ISBN: 0894352423.Physical description: ix, 280 p. : ill.
- [10]. Howden W. (1977) Symbolic testing and the DISSECT symbolic evaluation system. IEEE Trans. Sofrware Eng., vol. SE-4, no. 4, pp. 266-278.
- [12]. Korel, B., Wedde, H., & Ferguson R. (1991) *Automated test data generation for distributed software.* In Proc. COMPSAC'91, pp 680-685.
- [13]. Ramamoorthy C. (1976) S. Ho, and W. Chen, "On the automated generation of program test data," *IEEE Trans. Software Eng.,* vol. SE-2, no. 4, pp. 293-300, Dec. 1976.
- [14]. Hamlet, R.(2002) *Random testing.* In J. Marciniaik, editor, *Encyclopedia of Software Engineering.* John Wiley & Sons, second edition.
- [15]. Myers, G., J. (1979). *The Art of Software Testing.* Wiley, New York, second edition.
- [16]. Cobb, R. & Mills, H. D., (1990) *Engineering software under statistical quality control.* IEEE Software, 7(6): Pp.45-54.
- [17]. Dab'oczi T., et al (2003). *Automatic testing of graphical user interfaces.* In Proceedings of the 20th IEEE Instrumentation and Measurement Technology Conference 2003 (IMTC '03), pages 441–445, Vail, CO, USA.
- [18]. Forrester, J., E. & Miller, B., P.(2000). *An empirical study of the robustness of Windows NT applications using random testing.* In Proceedings of the 4th USENIX Windows Systems Symposium, pp59–68, Seattle.
- [19]. Miller B., P., Fredriksen L., & So. B. (1990). *An empirical study of the reliability of UNIX utilities.* Communications of the ACM, 33(12):pp32–44.
- [20]. Miller B., P. et al (1995). *Fuzz revisited: A re-examination of the reliability of UNIX utilities and services.* Technical Report CS-TR-1995-1268, University of Wisconsin.
- [21]. Miller. E., Website testing. <http://www.soft.com/eValid/Technology/White.Papers/website.testing.html>, Software Research, Inc., 2005.
- [22]. [20] Nyman, N. *In defense of monkey testing: Random testing can find bugs, even in well engineered software.* <http://www.softtest.org/sigs/material/nnyman2.htm>, Microsoft Corporation.

# Offset Minimization in a High Speed Low Power Sense Amplifier for CMOS SRAM Memory

Anand Kumar<sup>1</sup>, Manjit Kaur<sup>2</sup>, Gurmohan Singh<sup>3</sup>, <sup>1</sup>Lecturer, College of Engineering (COER), Roorkee  
<sup>2</sup> Senior Research Fellow, C-DAC Mohali, <sup>3</sup> Junior Telecom Officer (Telecom projects), BSNL Chandigarh  
manjit\_k4@yahoo.com, anand.vlsi07@gmail.com

**Abstract-** The offset voltage is the differential voltage developed between the bit-lines of a memory cell. Offset voltage should be low for lower power consumption and higher speed. The offset in sense amplifier is due to transistor mismatch in the identical matched transistor pair. This mismatch exists due to process related variations such as random dopant number fluctuations, interface-state density fluctuations etc. The transistor mismatch is expected to get worse with technology scaling due to the demanding requirements on process tolerance. Therefore, it becomes very important to develop the techniques to minimize the external manifestation of mismatch on circuit performance. The design of a high-speed low power sense-amplifier circuit for large CMOS SRAM memory is a big challenge. The large memories usually has long bit-lines which results in a large capacitive load to the memory cells, thus causing extra signal delay. The importance of transistor mismatch in the sense amplifier circuit for SRAM application has been very well recognized. Several techniques have been proposed to minimize the offset in sense amplifier. But, these techniques increase the sense amplifier circuit complexity, which is not desirable. In this paper, we investigated the effect of transistor mismatch on latch type sense amplifier and used a technique to minimize the offset by varying the dimensions of transistors used in memory cell and sense amplifier and thus reducing power consumption and sensing delay. We designed a 1Kb memory block, pre-charge circuitry and a block of sense amplifier connected with the bit-lines pairs of memory block. The design is simulated for optimized values of offset voltage, power consumption and access time of memory.

## I. INTRODUCTION

One of the major issues in the design of CMOS SRAMs is the access time or speed of read operation. To lower the access time or to increase the speed of read operation, it is necessary to take care of the read speed both in the memory cell-level design and in the design of a sense amplifier. Sense amplifiers are one of the most important building blocks in the organization of CMOS memories. The performance of sense amplifier strongly influences both memory access time and overall power consumption of the memory. A large sized memory has large bit-line parasitic capacitances. These large parasitic capacitances slow down voltage sensing and makes bit-line voltage swings energy-consuming, which result in slower and more power hungry memories. The need for high density memories, higher speed, and lower power dissipation imposes following trade-offs in the design of sense amplifier:

- 1) Increase in number of cells per bit-line increases the bit-line parasitic capacitance.
- 2) Increasing cell area to integrate more memory on a single chip reduces the current that is driving the heavily loaded bit-line. This causes smaller voltage swing on the bit-line.
- 3) Decreased power supply voltage lead to smaller noise margin that affects the sense amplifier reliability.

In this research work, the sense amplifier has been designed for read access time less than 12 ns, supply voltage range from 1.8 to 3.3 V, rise time of SAEN signal range from 100 to 400 ps, offset voltage range from 45 to 80 mV and power consumption less than 160 mW.

## II. IMPORTANCE OF SENSE AMPLIFIER IN SRAM

The sense amplifier is a very important circuit to regenerate the bit-line signals in a memory design. The sense amplifier is usually used to receive long interconnection signal with large RC delay and large capacitive load signal. The complexity of the differential logic circuit can be enhanced by combining the sense amplifier with differential logic networks to reduce the delay time. The design of a high-speed low power sense-amplifier circuit for large CMOS is a big challenge.

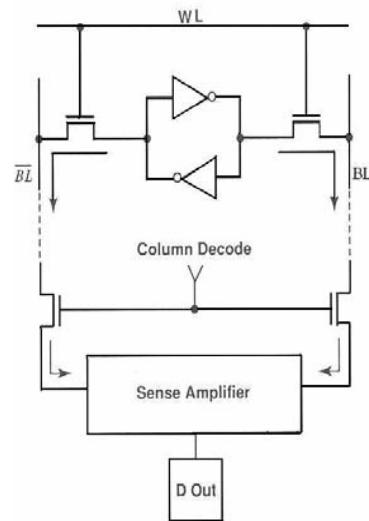


Fig. 1: Typical use of a Sense Amplifier

Sensing and amplifying the data signal which transmits through memory cell to bit-lines are the most important capability for a sense amplifier. To sense the data accurately and faster, it is getting very difficult due to scaling down of power supply levels. Since large numbers of memory cells are associated with bit-lines, the sensing delay becomes one of the bottlenecks of memory reading access time. Fig. 1 demonstrates a typical use of a sense amplifier.

## III. WORKING OF SENSE AMPLIFIER

The term "sensing" means the detection and determination of the data content of a selected memory cell. The sensing may be "nondestructive," when the data content of the selected

memory cell is unchanged e.g. in SRAMs, ROMs, PROMs, etc. and "destructive," when the data content of the selected memory cell may be altered e.g. in DRAMs by the sense operation. Sensing is performed in a sense circuit.

The full-complementary positive feedback sense amplifier shown in Fig. 2 improves the performance of the simple positive feedback amplifier by using an active load circuit comprising of CMOS transistors MP4, MP5 and MP6 in positive feedback configuration. In reality, transistor pairs MP4-MP5 and MN1 -MN2 cannot be completely matched despite carefully symmetrical design. Usually the asymmetry between the p-channel MP4 and MP5 is more substantial than that between the n-channel MN1 and MN2, because most of the CMOS processes focus more to optimize n-channel device characteristics.

To avoid a large initial offset resulting from the added effects of imbalances in the NAND p-channel device pair, source devices MN3 and MP6 are not turned on simultaneously, but first the n-channel and later the p-channel complex is activated by impulses  $\Phi_S$  and  $\Phi_L$  respectively. The delayed activation of transistors MP4-MP5-MP6 by clock  $\Phi_L$  results that until the time MP6 is turned on, device triad MN1-MN2-MN3 operates alone.

When the sense signal on the bit-line is large enough, e.g., when the drain-source voltage of either MN1 or MN2 reaches the saturation voltage  $V_{DSAT}$ , clock  $\Phi_L$  activates triad MP4-MP5-MP6. The activated feedback in MP4-MP5-MP6 introduces a pair of time dependent load resistances  $r L 1(t) = r d4(t) + 2r d6(t)$  and  $r L 2(t) = r d5(t) + 2rd6(t)$ . Here,  $r d(t)$  is the time dependent drain-source resistance, and indices 4, 5 and 6 represent devices MP4, MP5 and MP6. The resistances of these devices may be considered as time invariant parameters during the activation of MP6  $t_{SAT}$ , so that  $rL = r L1 = r L2$  may be used.

In the transient analysis, the differential signal development time  $t_d$  during the presence of impulse  $\Phi_S$  until the appearance of clock  $\Phi_L$  is determined by the switching time of the n-channel triad  $t_{dN}$ , and thereafter  $t_d$  is dominated by the transient time of the p-channel triad  $t_{dp}$  (Fig. 3). With this, the sense-signal development time in the full-complementary positive feedback differential voltage sense amplifier  $t_d$  may be approached as

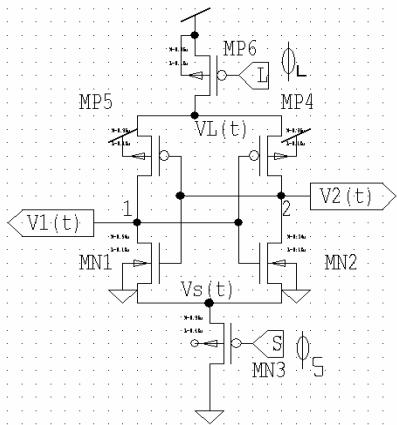


Fig 2 : Sense Amplifier

$$t_d = t_{dN} + t_{dp} = \tau_{dN} \ln \frac{V_{DSAT}}{2 \Delta V_o} + \tau_{dp} \ln \frac{0.9 (V_{DD} - V_{PR})}{V_{DSAT}}$$

Where

$$\tau_{dN} \approx \frac{C_B + C_{GSN} + 4C_{GDN}}{\beta_N [V_{PR} - v_s(0) - V_{TN}(V_{BG})]},$$

$$\tau_{dp} \approx \frac{C_B + C_{GSP} + 4C_{GDP}}{\beta_P [v_L(0) - V_{PR} - V_{TP}(V_{BG})]}$$

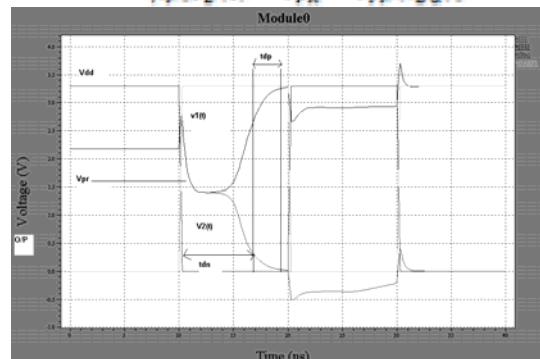


Fig 3: Output Voltage of Sense Amplifier

indices N and P designate n- and p-channel devices,  $V_{DSAT}$  is the saturation voltage,  $\Delta V_o$  is the amplitude of the initial voltage difference generated by the accessed memory cell on nodes 1 and 2,  $V_{PR}$  is the precharge voltage,  $C_B$  is the bit-line capacitance,  $C_{GS}$  and  $C_{GD}$  are the gate source and gate-drain capacitances, and  $\beta$  is the individual gain factor for devices MN1, MN2, MP4 and MP5,  $V_s(0)$  and  $V_L(0)$  are the initial potentials on the drains of device MN3 and MP6,  $V_T$  is the threshold voltage and  $V_{BG}$  is the backgate bias.

The equation of  $t_d$  demonstrate that in a full-complementary positive feedback differential sense amplifier quicker operation can be obtained by increasing the gain factors  $\beta_N$  and  $\beta_P$ , by decreasing the parasitic gate source capacitance  $C_{GS}$  and gate-drain capacitance  $C_{GD}$  of the N- channel and P-channel latch devices MN1, MN2, MP4 and MP5, and by decreasing the bit-line capacitance  $C_{BL}$ . Additionally, reductions in the fall time of  $V_s(t)$  and in the rise time of  $V_r(t)$  also shorten  $t_d$ .

#### IV. OFFSET IN SENSE AMPLIFIER

Low-power SRAMs have become a critical component of many VLSI chips. This is especially true for embedded memories like on-chip caches. The key to low-power operation in the SRAM is to reduce the signal swings on the high capacitance bit-lines. This minimum required signal swing is limited by the offset in the sense amplifier. The higher the offset, the higher is the power consumption and the sense delay. This brings us to the typical trade-off between memory yield and power-delay product.

##### A. Effect of Offset on SRAM Performance

The access time of the memory in presence of offset depends on two factors:-

- 1) First Delay Factor- It is the time required to develop the required differential input at the sense amplifier.

2) Second Delay Factor- It is the delay due to sense amplifier itself.

B. *Delay factors dependence on  $C_{BL}$  and rise time of SAEN*

- 1) The first delay factor is approximately  $(\Delta V_{BL} C_{BL}) / I_{MEMCELL}$  where  $\Delta V_{BL}$  is difference in bit-lines voltage or offset voltage of sense amplifier.  $C_{BL}$  is directly proportional to the number of rows in memory array.
- 2) The second delay factor is independent of  $C_{BL}$  but depends upon the rise time of SAEN.

C. *Reduction in overall delay and power dissipation by increasing rise time of SAEN signal*

- 1) By decreasing the size of MN2 in the sense amplifier and increasing the rise time of SAEN signal, the second factor increases but first factor decreases. For large  $C_{BL}$ 's the decrease in first delay factor can outweigh the increase in second delay factor, resulting in lower total delay.
- 2) The power dissipation in bit-lines is  $(\Delta V_{BL} \cdot C_{BL} \cdot V_{dd})$  which decrease with decrease in  $\Delta V_{BL}$ . Thus low offset or low  $\Delta V_{BL}$  results in low power.

## V. DESIGNING OF SENSE AMPLIFIER AND MEMORY BLOCK

This work consists of designing and simulation of 1Kb memory block, pre-charge circuitry and a block of sense amplifiers connected with the bit-lines pairs of memory block. The target was to improve the power consumption and access time of memory. A input decoupled latch type sense amplifier architecture is used to have low power and high speed. The fig. 4 shows the schematic of the circuit drawn in Schematic Editor (S-Edit) tool of Tanner. The fig. 5 shows the complete 1Kb memory block, pre-charge circuitry and a block of sense amplifiers connected with the bit-lines pairs of memory block. The aspect ratio or W/L ratio of transistors is the main factor to achieve the desired objectives for analog designers. We varied the dimensions of the transistor to achieve the optimum values of offset voltage of the bit-lines. Another very important parameter is power consumption during read operation. This work shows improved power consumption results by optimizing the value of W/L ratio of SRAM cell transistor and sense amplifier.

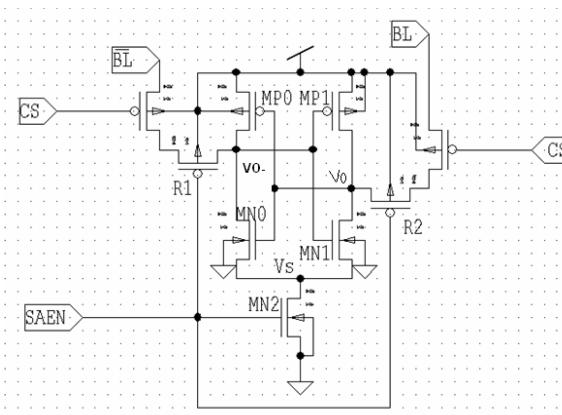


Fig 4: Input Decoupled Latch Type Sense Amplifier

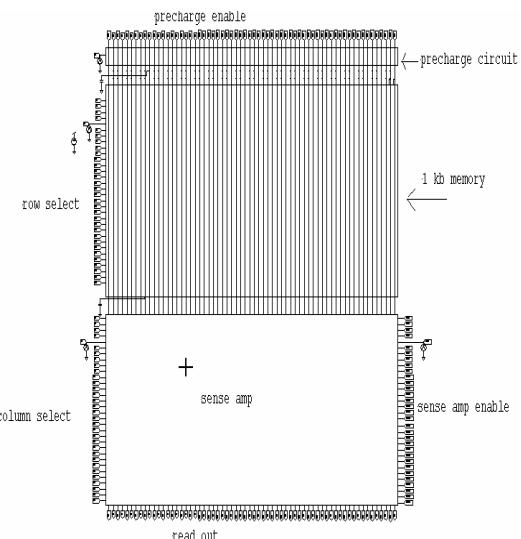


Fig. 5: Block Diagram in Tanner S-Edit Tool

## V. SIMULATION RESULTS

The following simulation waveforms are observed for input decoupled latch type sense amplifier and system schematic shown in fig. 4 and fig. 5. All the results shown below are for reading „1“ from SRAM memory cell. We have used  $0.18\mu m$  standard CMOS technology parameters for simulation purposes. Offset voltage of sense amplifier and power consumption during read operation was found decreasing with decreasing the size of transistor MN2. However, the delay is also kept within tolerable limit with power consumption. Simulated result is shown below in fig. 6. The table I shows simulated results for access time, offset voltage and power consumption. As the width of transistor MN2 decreases 2nd delay factor increases, but it results in lower values of offset voltage and power consumption.

Table I  
Variation in Power Consumption and Delay

Width (W) Transistor MN2( $\mu m$ )	2 <sup>nd</sup> Delay fact (ns)	Offset voltas (mV)	Power Consumptid ( mW)
0.54	3.3	82.53	15.7
0.30	3.8	75.66	14.9
0.18	4.4	61.90	13.8

A. *Output voltage of Sense Amplifier*

The figure 6 shows the output voltage of Sense Amplifier for reading a '1' from memory cell. The Output voltage shows the logic '1' i.e.  $V_{dd}$ . Access time of memory is shown in figure below by an arrow. It comes out to be 3 ns.

Pre-charge signal and sense amplifier enable signal are also shown in figure

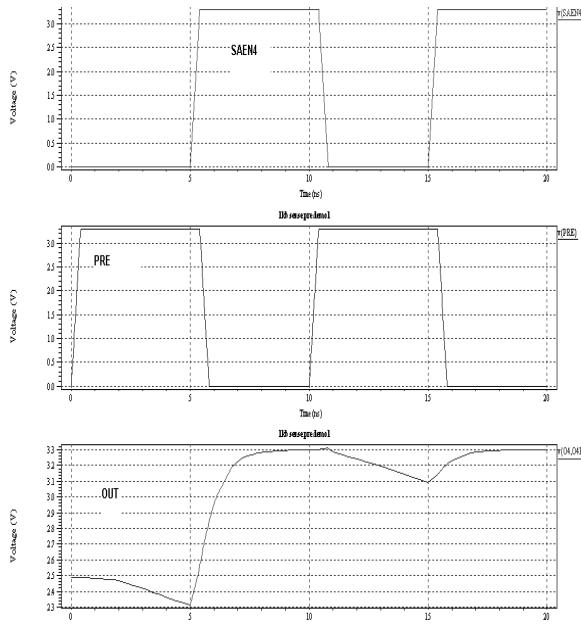


Fig. 6: Output Voltage of Sense Amplifier

#### B. Power Consumption Results

$V_3$  from time 1ns to 20ns

Average power consumed > 13.20 mW

Max power 27.22 mW at time 1.61ns

Min power 0.785 mW at time 15ns

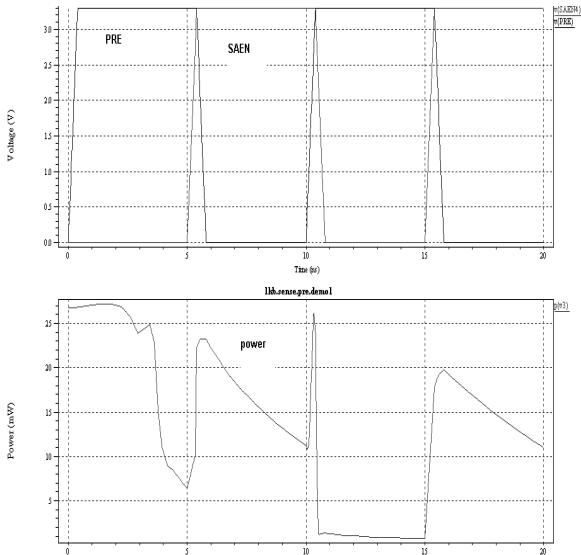


Fig. 7: Simulation waveform showing power consumption

#### C. Bit-line Voltage of SRAM Memory Cell

The figure 8 shows the variation in bit-line voltage. Bit-lines are pre-charged to 1.65 V during PRE signal and during read operation , these voltages change and difference of the bit-lines is amplified by Sense Amplifier.

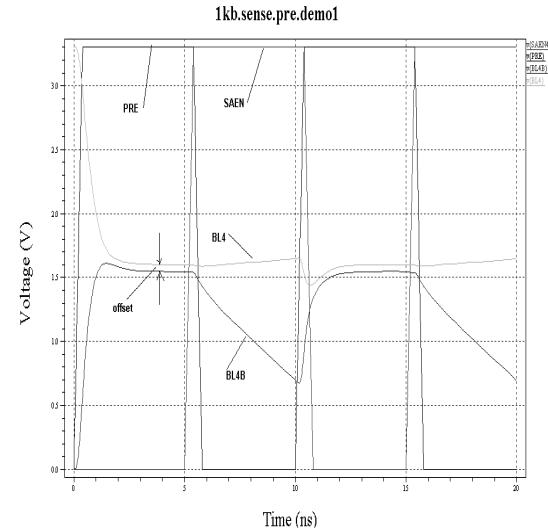


Fig. 8: Bit-lines Voltage of Memory Cell

#### VI. REFERENCES

- [1]. H. Mahmoodi, S. Mukhopadhyay, and K. Roy, "Estimation of delay variations due to random-dopant fluctuations in nanoscale CMOS circuits," IEEE J. Solid-State Circuits, vol. 40, pp. 1787-1796, Sept. 2005.
- [2]. J. Bhavnagarwala, X. Tang, and J. D. Meindl, "The impact of intrinsic device fluctuations on CMOS SRAM cell stability" IEEE J. Solid-State Circuits, vol. 36, pp. 658-665, Apr. 2001.
- [3]. Agarwal, B. Paul, S. Mukhopadhyay, and K. Roy, "Process variation in embedded memories: Failure analysis and variation aware architecture", IEEE J. Solid-State Circuits, vol. 40, pp. 1804 - 1813, 2005.
- [4]. Kang, Sung-Mo and Leblebici, Yusuf, "CMOS Digital Integrated Circuits – Analysis and Design", McGraw-Hill International Editions, Boston, 2<sup>nd</sup> Edition, 1999.
- [5]. Adel S. Sedra and Kenneth C. Smith, "Microelectronics Circuits" Oxford University Press International Edition, New York, 5th Edition 2006.
- [6]. Ardalan,S.; Chen, D.; Sachdev, M.; Kennings, A.; "Current mode sense amplifier" Circuits and Systems, 2005. 48th Midwest Symposium Vol. 1, 7-10 Aug. 2005 Page(s):17–20.
- [7]. Tegze P. Haraszti, Microcirc Associates "CMOS Memory Circuits", Kluwer Academic publishers New York, Boston, Dordrecht, London, Moscow, Pages 238-239.
- [8]. Hwang-Cherng Chow; Shu-Hsien Chang; "High performance sense amplifier circuit for low power SRAM application", " Circuits and System, 2004,ISCAS'04, Proceedings of the 2004 International Symposium" Vol.2,23-26 May 2004,Page(s):II-741-4 Vol.2

# Overview of Microelectro-Mechanical Systems and Fabrication Technologies

\*Ms. Jayshri Shelke, \*\*Mr. S.M. Salodkar

\*Faculty, C.C.E.T, Chandigarh, \*\*Faculty, PEC University of Technology, India, jayshri\_shelke@rediffmail.com  
Chandigarh, India sndpchd@yahoo.co.in

**Abstract-**This paper gives the basic idea of microelectromechanical systems and the different design processes required for the fabrication of MEMS devices. New design tools and automation strategies are required to create robust, cost-effective, and manufacturable micro machined devices and systems. Some of the design automation includes mixed technology simulation, material property prediction in the micron-size, integrated modeling environment, and synthesis of device geometries and process flows. Advancement in these areas will the way to full-scale maturity of the MEMS field.

## I. INTRODUCTION

Since the invention of integrated circuits, batch fabrication techniques developed for the microelectronic industry have been used to create micromechanical structures on silicon substrates. A few early examples include the resonant gate transistor [1], silicon diaphragms for pressure sensing [2], and accelerometers [3]. In recent years, the use of planar technologies to develop commercial MEMS devices has become more and more sophisticated, because of the increased demands for micro sensors and micro actuators with improved performance-to-cost ratio, better reliability, and new functionality over conventional counterparts. The commercial markets based on MEMS technology include the automotive airbag accelerometers [4], pressure sensors [5], and thermal ink-jet print heads [6]. In addition, there are number of emerging research areas that take advantage of the new functionality enabled by MEMS. A few examples include the study of fluid dynamics in the micron-size regime [7], tribology [8], miniaturized chemical analysis systems [9], and biomedical research [10]. New design tools and automation strategies are needed in order to provide robust, cost-effective, and manufacturable MEMS products. In particular, the characteristics and performance of various fabrication techniques must be simulated together with material property prediction. The electromechanical coupling in certain MEMS devices demands a self-consistent modeling tool. Micro-fluidic devices may require modeling of high-viscous flows and low-pressure damping. Finally, the ability to synthesize process flows and device geometries from function definition will represent the ultimate design automation of micromechanical systems.

## II. COMPONENTS OF MEMS

MEMS integrate sensors, electronics, and actuators on a common platform using micro fabrication technology. Sometimes passive structures are also found on MEMS. This unique combination of various devices makes it possible to develop miniature system with huge capabilities. The various component of MEMS are shown in fig 1

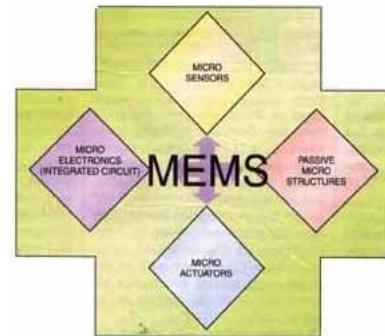


Fig: 1 Various components of MEMS

## III. MEMS FUNCTIONING

Fig 2 shows the functioning of MEMS. The sensor collects information from the environment and provides it to the circuit. The electronics circuit processes this information and gives control signals to the actuator, which then manipulates the environment for the desired purpose. In other word MEMS work just like us. They sense, think, and then work according to the situations. In this way, MEMS bring a revolution in technology by making systems more intelligent. The electronics component is fabricated using IC processing. the mechanical components are fabricated using micromachining. The integration of micro electromechanical systems with electronics makes it possible to realize complete system on chip with a smaller size and hence a higher performance.

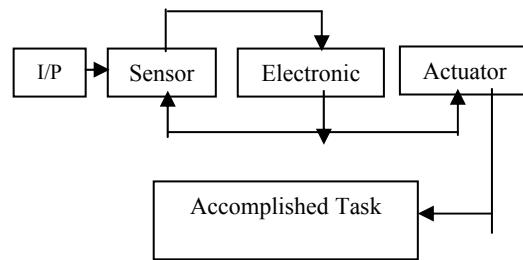


Fig 2 MEMS Functioning

## IV. MEMS TECHNOLOGIES

The most common fabrication techniques for microelectromechanical systems include three distinct categories: Bulk Micromachining, Surface Micromachining, and high aspect-ratio lithography and plating (LIGA, a German acronym, for Lithographie, Galvanoformung, Abformung means Lithography, Electroforming and Moulding) [11]. The first two i.e. Bulk Micromachining and Surface Micromachining are sometimes mixed to create microstructures with specific functions.

**Bulk Micromachining:** Figures 3, outlines the steps of a typical Bulk Micromachining process, in which anisotropic wet chemical etching is used to create structures from the silicon

substrate. The structures are created by depositing masking layers of silicon dioxide, silicon nitride, or metals (Au, Ti etc) and patterning this using lithography. Initial MEMS products like Pressure sensor, accelerometers, ink-jet printer heads etc. are made using Si bulk micromachining.

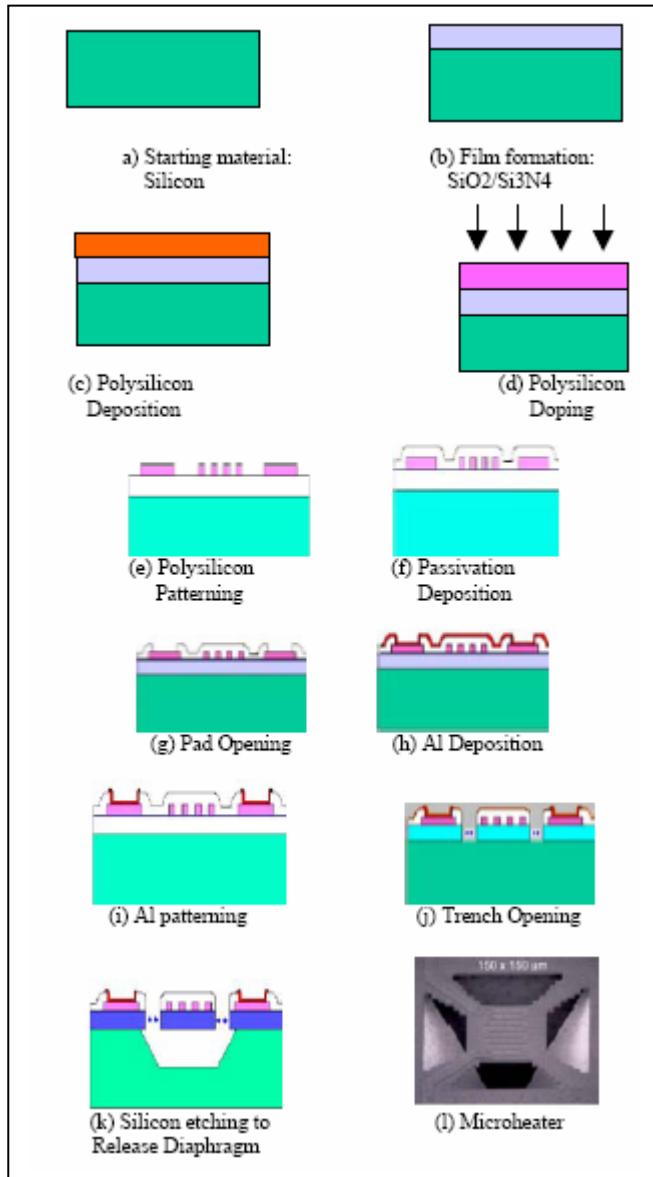


Fig: 3.fabrication steps for Microheater using Bulk Micromachining

**Surface micromachining:** Steps are illustrated in (Fig: 4) this technique is a more advanced technique that offers flexibility to make novel structures on the surface of silicon wafer. In this variant MEMS technology thin films are sequentially deposited and patterned on top of the substrate, using lithographic and etching techniques. Electrical, structural and sacrificial are the main three layers employed in Surface Micromachining. Electrical layers conduct signals to and form the MEMS structure. Structural layer form the mechanical body of the MEMS and the sacrificial layers serves the purpose of releasing (Making deflectable or movable) the structural

layers. Sacrificial layers are completely removed by etching in the final micromachining step. Structures like Cantilever, Micromirror, Gears, and Micromotor etc. are made using surface micromachining.

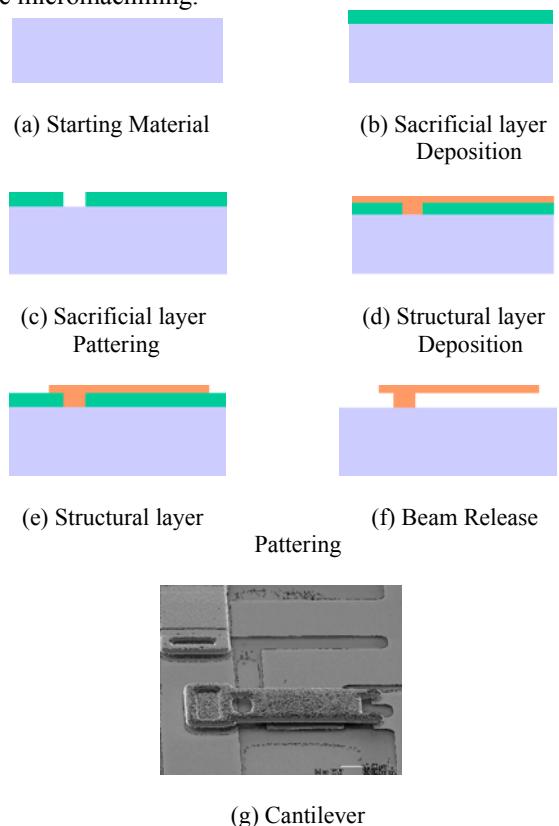


Fig: 4. Fabrication steps for Cantilever using Surface Micromachining

**LIGA:** The LIGA technique is used to form micron sized structures. Here, mould of the pattern required is fabricated on a 10-1000 $\mu\text{m}$  thick polymer (such as polymethyl methacralate) using x-ray lithography. Then the mould is filled by electroplating followed by removal of resist to achieve the final structures. This technique could not become much popular because of many process difficulties like electroplating and x-ray lithography. Three dimensional structures are possible using this technique.

Deep Reactive Ion Etching is the alternative technique, applicable only to silicon. (Fig. 5). These three techniques are compared and summarized in Table I [12].

Recently, Deep Reactive Ion Etching (DRIE) has become popular. It employs plasma etching to generate high ratio (depth/width) structures in silicon. The requirement of more sophisticated and peculiar structures has generated interest in dry etching techniques. Micromachining with help of dry etching techniques, offers better control on the etch profile. DRIE is one such technique that is finding use in more sophisticated structures like micro fluidics, bio-MEMS, and gyroscope. In this technique, high density plasma sources are used to etch vertically deep trenches in wafers at high etch rates. Inductively coupled plasma sources are commonly

employed to generate plasmas of high densities. DRIE offers the flexibility of choosing the etch profile, type of mask, and etch rates.

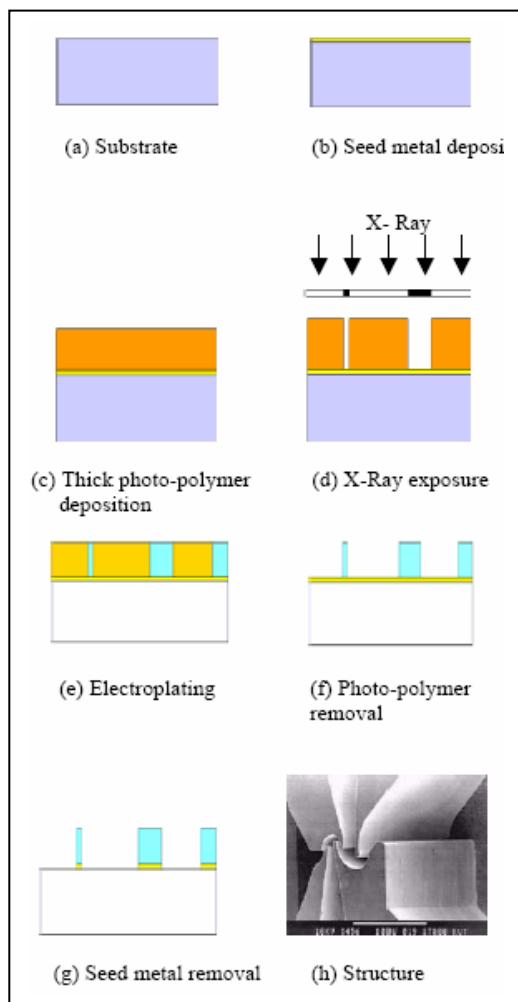


Fig: 5. LIGA Process

Table 1. MEMS Technology Comparison

Capability	Bulk	Surface	LIGA
Maximum structural thickness	Wafer(s) Thickness	5 μm	500 μm
Planar geometry	Rectangular	Unrestricted	Unrestricted
Surface and edge definitions	Excellent	Mostly Adequate	Very Good
Material properties	Very well controlled	Mostly Adequate	Well controlled
Integration with electronics	Possible	Demonstrated	Difficult
Capital investment & costs	Low	Moderate	High
Published knowledge	Very high	High	Moderate

#### IV APPLICATIONS OF MEMS

The first application of MEMS was a silicon based strain gauge, which was commercialized around 1958. Silicon based pressure sensor and resonant gate field effect transistor (FET) were developed in the 1960s. In the 1970s, accelerometer and

nozzle for inkjet printer were developed. These two devices hold a major market share of silicon based MEMS devices. Later on, many new MEMS technologies like surface micromachining and LIGA were developed during 1980s and 1990s. The late 1990s witnessed development of MEMS applications in different fields like optics, radio frequency, and life science.

Now MEMS have penetrated almost all walks of our life.

**Consumer:** Typical consumer appliances that incorporate MEMS are washing machine, refrigerators, air-conditioners, lighting, toys, and safety systems. MEMS add automation and intelligence to these products.

**Communications:** Most wireless systems rely upon radio frequency (RF) devices like capacitors, inductors, antennae, and tuning components, which are very bulky and cover a large space in any system. MEMS offer seamless miniaturization of antennae, switches, frequency-selection mechanisms, microphones, etc, making the RF systems more capable and versatile.

In light based communication systems like optical fibers and lasers, the micro-optical-electromechanical systems (MOEMS) technology is used to achieve miniaturization. MOEMS devices include micro lenses, deflectors, polarizer, filters, optical benches, fiber couplers, cross connectors, modulators, multiplexers, attenuators, equalizers, light switches, etc.

**Computers:** The largest market for MEMS is computer peripherals and data storage systems. Among these, major parts using MEMS are read/write heads of hard disks and floppy disks. Inkjet printer nozzles and thermal and piezoelectric print heads are also made using MEMS technology. MEMS based sensor and actuator can make computer peripherals much smarter by providing better human and environment interface.

**Life sciences:** MEMS used in various disciplines of life sciences are referred to as Bio-MEMS. After computers and information technology, life science is the next biggest market for MEMS. Bio-MEMS find applications in medical science, forensics, pharmaceuticals, food and drink industry, and environment sensing. In future, new applications in fermentation control, agriculture, and anti-bio weapons may also be developed.

**Automobile:** Using MEMS technology, different mechanical structures like gears, micro motors, cranks, anchors, cantilevers, diaphragms, etc can be fabricated with great precision. These structures can be employed to sense a variety of parameters in automobiles.

Accelerometers find the biggest use in automobiles, mainly in airbag safety system to detect the collision impact and inflate the airbags to protect the passengers. Conventional accelerometers are now being replaced with MEMS counterparts the cost 10 to 20 times less.

MEMS based pressure sensors facilitate checking of the tyre pressure on digital readout at the vehicle's dashboards.

**Aerospace and military:** various types of inertial MEMS are used in spacecraft, airplanes, satellites, missiles, and so on. MEMS devices like micro thrusters, micro-turbines, micro-satellites, and micro engines gave a new horizon to the

exploration of the space. Using these systems, sophisticated space vehicles can be made at reduced manufacturing and launching costs. In military domain, night-vision devices, micro-cameras, and micro-spacing systems are all based on MEMS.

#### V. REFERENCES

- [1]. H. C. Nathanson, W. E. Newell, R. A. Wickstrom, and J. R. Davis, Jr., "The resonant gate transistor," IEEE Trans. Electron Devices, vol. ED-14, pp. 117–133, Mar. 1967.
- [2]. Samaun, K. D. Wise, and J. B. Angell, "An IC piezoresistive pressure sensor for biomedical instrumentation," IEEE Trans. Biomed. Eng., vol. BME-20, pp. 101–109, Mar. 1973.
- [3]. M. Roylance and J. B. Angell, "A batch-fabricated silicon accelerometer," IEEE Trans. Electron Devices, vol. ED-26, pp. 1911–1917, Dec. 1979.
- [4]. Spangler and C. J. Kemp, "A smart automotive accelerometer with on-chip airbag deployment circuits," Tech. Dig., Solid-State Sensor and Actuator Workshop, Hilton Head Island, SC, pp. 211–214, June 3–6, 1996.
- [5]. Ajluni, "Pressure sensors strive to stay on top," Electronic Design, pp. 67–74, Oct. 3, 1994.
- [6]. C. Beatty, "A chronology of thermal ink-jet structures," Tech. Dig., Solid-State Sensor and Actuator Workshop, Hilton Head Island, SC, pp. 200–204, June 3–6, 1996.
- [7]. J. C. Shih, C-M Ho, J. Liu, and Y.-C. Tai, "Monatomic and polyatomic gas flow through uniform microchannels," Proc., 1996 ASME Int. Mech. Eng. Congress and Exposition, vol. DSC-Vol. 59, pp. 197–203, Nov. 17–22, 1996.
- [8]. J. F. Burger, G-J. Burger, T. S. J. Lammerink, S. Imai, and J. J. J. Fluitman, "Miniaturized friction force measuring system for tribological research on magnetic storage devices," Proc., IEEE Int. Workshop on Micro Electro Mech. Syst., San Diego, CA, pp. 99–104, Feb. 11–15, 1996.
- [9]. J. R. Webster, D. K. Jones, and C. H. Mastrangelo, "Monolithic capillary gelectrophoresis stage with on-chip detector," Proc., IEEE Int. Workshop on Micro Electro Mech. Syst., San Diego, CA, pp. 491–496, Feb. 11–15, 1996.
- [10]. Q. Bai and K. D. Wise, "A high-yield process for three-dimensional microelectrode arrays," Tech. Dig., Solid-State Sensor and Actuator Workshop, Hilton Head Island, SC, pp. 262–265, June 3–6, 1996.
- [11]. H. Mastrangelo and W. C. Tang, "Chapter 2: Sensor technology," Semiconductor Sensors, (S. M. Sze, Ed.), New York: John Wiley & Sons, Inc., 1994.
- [12]. William C. Tang, "Overview of Microelectromechanical systems and design processes," Design Automation Conference, 06/97

# FPGA Implementation of Adaptive Filter using LMS Algorithm

Ms. Neha Mittal\*, Ms Preeti Sharma\*, Ms.Priyanka Mehta\*, Er. Balwinder Singh\*\*

\*Chitkara Institute of Engg. & Tech, Rajpura, \*\* Centre For Development of Advanced Computing, Mohali

**Abstract -** The paper describes hardware implementation of Adaptive Filter using modified LMS algorithm. Filtering data in real-time requires dedicated hardware to meet demanding time requirements. If the statistics of the signal are not known, then adaptive filtering algorithms can be implemented to estimate the signals statistics iteratively. The Least Mean Square (LMS) adaptive filter is a simple well behaved algorithm which is commonly used in applications where a system has to adapt to its environment. Modern field programmable gate arrays (FPGAs) include the resources needed to design efficient filtering structures. Reconfigurable hardware devices offer both the flexibility of computer software, and the ability to construct custom high performance computing circuits. An approach to the implementation of digital filter algorithms based on reconfigurable platform i.e. field programmable gate arrays (FPGAs) is presented. Adaptive Filter's study and practical implementation is carried out by using VHDL language.

## I. INTRODUCTION

On systems that perform real-time processing of data, performance is often limited by the processing capability of the system [1]. Therefore, evaluation of different architectures to determine the most efficient architecture is an important task. The purpose of this paper is to explore the use of Field Programmable Gate Arrays (FPGAs) offer. Specifically, it investigates their use in efficiently implementing adaptive filtering applications. Different architectures for the filter are compared. These are compared to training algorithms implemented in the FPGA fabric only, to determine the optimal system architecture.

## II. ADAPTIVE FILTER OVERVIEW

Adaptive filters are digital filters capable of self adjustment. These filters can change in accordance to their input signals. An adaptive filter is used in applications that require differing filter characteristics in response to variable signal conditions. Adaptive filters are typically used when noise occurs in the same band as the signal, or when the noise band is unknown or varies over time. The adaptive filter requires two inputs: the signal and a noise or reference input. An adaptive filter has the ability to update its coefficients. New coefficients are sent to the filter from a coefficient generator. The coefficient generator is an adaptive algorithm that modifies the coefficients in response to an incoming signal. In most applications the goal of the coefficient generator is to match the filter coefficients to the noise so the adaptive filter can subtract the noise out from the signal. Since the noise signal changes the coefficients must vary to match it, hence the name adaptive filters. The digital filter is typically a special type of finite impulse response (FIR) filter, but it can be an infinite impulse response (IIR) or other type of filter. Adaptive filters have uses in a number of applications including noise cancellation, linear

prediction, adaptive signal enhancement, and adaptive control.

## III. GENERAL DESCRIPTION OF FPGA-BASED SIGNAL PROCESSING

Most digital signal processing done today uses a specialized microprocessor, called a digital signal processor, capable of very high speed multiplication. This traditional method of signal processing is bandwidth limited. There are a fixed number of operations that the processor can perform on a sample before the next sample arrives. This limits either the applications that can be performed on a signal or it limits the maximum frequency signal that the application can handle. This limitation stems from the sequential nature of processors. DSPs using a single core can only perform one operation on one piece of data at a time. They cannot perform operations in parallel. For example, in a 64 tap filter they can only calculate the value of one tap at a time, while the other 63 taps wait. Nor can they perform pipelined applications. In an application calling for a signal to be filtered and then correlated, the processor must first filter, then stop filtering, then correlate, then stop correlating, then filter, etc. If the applications could be pipelined, a filtered sample could be correlated while a new sample is simultaneously filtered. Digital Signal Processor manufacturers have tried to get around this problem by cramming additional processors on a chip. This helps, but it is still true that in a digital signal processor most of your application is idle most of the time.

FPGA-based digital signal processing is based on hardware logic and does not suffer from any of the software based processor performance problems. FPGAs allow applications to run in parallel so that a 128 tap filter can run as fast as a 10 tap filter. Applications can also be pipelined in an FPGA, so that filtering, correlation, and many other applications can all run simultaneously. In an FPGA, most of your application is working most of the time. An FPGA can offer 10 to 1000 times the performance of the most advanced digital signal processor at similar or even lower costs.

## IV. ADAPTIVE FILTER DESIGN

This Design is based on a 12 bit data, 12 bit coefficient, full precision, block adaptive filter design. It can be modified to accommodate different data and coefficient sizes, as well as lesser precision. The applications note covers how to modify the design including the trade-offs involved. The filter is engineered for use in the XC4000E and XC4000EX families. The synchronous RAM and carry logic in these families make this design possible.

There are a large number of designs that could fit this application. This design has a good balance between performance and density (gate count). This design can sustain a 15.5 MHz, 12 bit sample rate with an unlimited number of

filter taps. Modified versions of this design can provide higher throughput at the expense of consuming more resources, or modified versions can provide better resource efficiency at a lower performance.

Figure 1 is an overview of the entire adaptive filter. There are four basic components to the filter: the Table Generator, the Data Framer, the Filter Tap, and the Adder Tree.

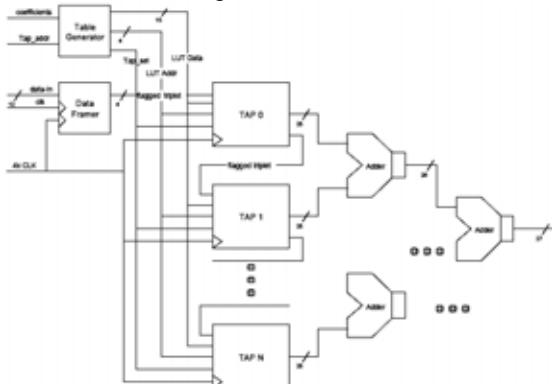


Figure 1 Block Diagram of the Block Adaptive Filter

#### V. ADAPTIVE FILTERING PROBLEM

The goal of any filter is to extract useful information from noisy data. Whereas a normal fixed filter is designed in advance with knowledge of the statistics of both the signal and the unwanted noise, the adaptive filter continuously adjusts to a changing environment through the use of recursive algorithms. This is useful when either the statistics of the signals are not known beforehand or change with time.

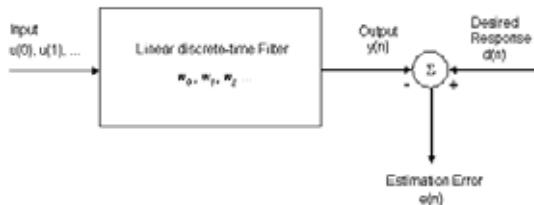


Figure 2 Block diagram for the adaptive filter problem.

The discrete adaptive filter (see figure 2) accepts an input  $u(n)$  and produces an output  $y(n)$  by a convolution with the filter's weights,  $w(k)$ . A desired reference signal,  $d(n)$ , is compared to the output to obtain an estimation error  $e(n)$ . This error signal is used to incrementally adjust the filter's weights for the next time instant. Several algorithms exist for the weight adjustment, such as the *Least-Mean-Square* (LMS) and the *Recursive -Least-Squares* (RLS) algorithms. The choice of training algorithm is dependent upon needed convergence time and the computational complexity available, as statistics of the operating environment.

#### VI. ADAPTIVE ALGORITHM

There are numerous methods for the performing weight update of an adaptive filter. There is the Wiener filter, which is the optimum linear filter in the terms of mean squared error, and several algorithms that attempt to approximate it, such as the method of steepest descent. There is also least mean square algorithm, developed by Widrow and Hoff originally for use in artificial neural networks. Finally, there are other techniques

such as the recursive least squares algorithm and the Kalman filter. The choice of algorithm is highly dependent on the signals of interest and the operating environment, as well as the convergence time required and computation power available. We have used LMS algorithm in our design

The least-mean-square (LMS) algorithm is similar to the method of steepest-descent in that it adapts the weights by iteratively approaching the MSE minimum. The error at the output of the filter can be expressed as

$$e_n = d_n - \mathbf{w}_n^T \mathbf{u}_n,$$

This is simply the desired output minus the actual filter output. Using this definition for the error an approximation of the gradient is found by

$$\hat{\nabla} = -2e_n \mathbf{u}_n.$$

Substituting this expression for the gradient into the weight update equation from the method of steepest-descent gives

$$\mathbf{w}_{n+1} = \mathbf{w}_n + 2\mu \cdot e_n \mathbf{u}_n,$$

This is the Widrow-Hoff LMS algorithm. As with the steepest-descent algorithm, it can be shown to converge [2] for values of  $\mu$  less than the reciprocal of  $\lambda_{\max}$ , but  $\lambda_{\max}$  may be time-varying, and to avoid computing it another criterion can be used. This is

$$0 < \mu < \frac{2}{MS_{\max}},$$

where  $M$  is the number of filter taps and  $S_{\max}$  is the maximum value of the power spectral density of the tap inputs  $u$ .

The relatively good performance of the LMS algorithm given its simplicity has caused it to be the most widely implemented in practice. For an  $N$ -tap filter, the number of operations has been reduced to  $2*N$  multiplications and  $N$  additions per coefficient update. This is suitable for real-time applications, and is the reason for the popularity of the LMS algorithm.

#### VII. TRAINING ALGORITHM MODIFICATION

The training algorithms for the adaptive filter need some minor modifications in order to converge for a fixed-point implementation. Specifically, the learning rate  $\mu$  and all other constants should be multiplied by the scale factor. First,  $\mu$  is adjusted

$$\hat{\mu} = \frac{1}{\mu} \cdot \text{scale}.$$

The weight update equation then becomes:

$$\hat{\mathbf{w}}(n+1) = \hat{\mathbf{w}}(n) + \frac{\mathbf{u}(n)e^*(n)}{\hat{\mu}}.$$

This describes the changes made for the direct form FIR filter, and further changes may be needed depending on the filter architecture at hand.

### VIII. LOADABLE COEFFICIENT FILTER TAPS

The heart of any digital filter is the filter tap. This is where the multiplications take place and is therefore the main bottleneck in implementation. Many different schemes for fast multiplication in FPGAs have been devised, such as distributed arithmetic, serial-parallel multiplication, and Wallace trees [3], to name a few. Some, such as the distributed arithmetic technique, are optimized for situations where one of the multiplicands is to remain a constant value, and are referred to as constant coefficient multipliers (KCM) [4]. Though this is true for standard digital filters, it is not the case for an adaptive filter whose coefficients are updated with each discrete time sample. Consequently, an efficient digital adaptive filter demands taps with a fast variable coefficient multiplier (VCM).

A VCM can however obtain some of the benefits of a KCM by essentially being designed as a KCM that can reconfigure itself. In this case it is known as a dynamic constant coefficient multiplier (DKCM) and is a middle-way between KCMs and VCMs [4]. A DKCM offers the speed of a KCM and the reconfiguration of a DCM although utilizes more logic than either. This is a necessary price to pay however, for an adaptive filter.

### IX. TAP IMPLEMENTATION RESULTS

Of the DKCM architectures described, several were chosen and coded in VHDL to test their performance. Namely, the serial-parallel, partial products multiplication, and embedded multiplier are compared to ordinary CLB based multiplication inferred by the synthesis tool. All were designed for 12-bit inputs and 24-bit outputs. The synthesis results relevant to the number of slices flip-flops, 4 input LUTs, BRAMs, and embedded multipliers instantiated is offered. A comparison of the speed in Megahertz and resources used in terms of configurable logic blocks for the different implementations is presented in figure 3.

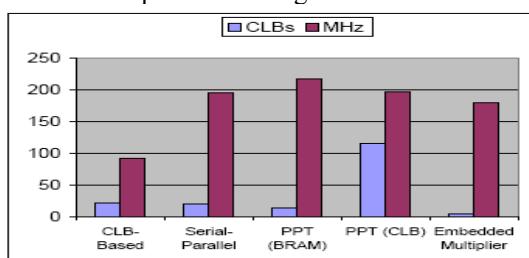


Figure 3 CLB Resources and Speed of Selected Tap Implementations

Since the filter is adaptive and updates its coefficients at regular intervals, the time required to configure the tap for a new coefficient is important. The reconfiguration times for the various multipliers are listed in table 1.

Table 1 Reconfiguration Time and Speed for Different Multipliers

Architecture	Reconfiguration Time (clks)	Speed (MHz)
CLB-Based	1	93.075
Embedded Multiplier	1	179.988
Serial-Parallel	1	196.425
Partial Product (CLB)	16	197.902
Partial Product (BRAM)	16	217.96

### X. CONCLUSIONS

Until an appropriate fixed-point structure is found for the Recursive Least-Squares algorithm, the Least Mean-Square algorithm was found to be the most efficient training algorithm for FPGA based adaptive filters. The issue of whether to train in hardware or software is based on bandwidth needed and power specifications, and is dependent on the complete system being designed.

### XI. REFERENCES

- [1]. K.A. Vinger, J. Torresen, "Implementing evolution of FIR-filters efficiently in an FPGA." Proceeding,. NASA/DoD Conference on Evolvable Hardware, 9-11 July 2003. Pages: 26 – 29
- [2]. Sudhakar Yalamanchili, Introductory VHDL, From Simulation to Synthesis, Prentice Hall, 2001.
- [3]. Douglas L. Jones,"Learning Characteristics of Transpose-Form LMS Adaptive Filters",IEEE 1057-7130/92,1992.
- [4]. S. Haykin, Adaptive Filter Theory, Prentice Hall, Upper Saddle River, NJ, 2002.
- [5]. Xilinx Inc., "Block Adaptive Filter," Application Note XAPP 055, Xilinx, San Jose, CA, January 9, 1997.
- [6]. S. S. Godbole1, P. M. Palsodkar2 and V.P. Raut3,"FPGA Implementation of Adaptive LMS Filter "SPIT-IEEE Colloquium and International Conference, Mumbai, India,1992.

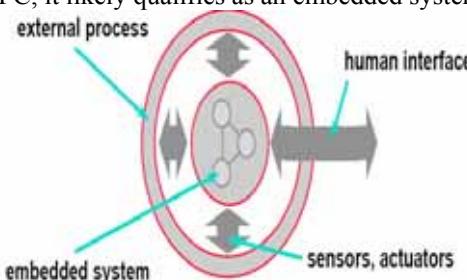
# Embedded System Design using Programmable Logic Controllers

\*Poonam Jindal, \*\*Isha Verma, \*\*\*Aarti Bansal  
 \*Sr.Lecturer (ECE), \*\*Sr.Lecturer (ECE), \*\*\*A.P(E.C.E)  
 CIET, Rajpura, [poonam.jindal@chitkara.edu.in](mailto:poonam.jindal@chitkara.edu.in), +91-9417004922

**Abstract -** Programmable logic controllers (PLCs) are complex cyber-physical systems which are widely used in industry. The Programmable Logic Controller (PLC) in general meets today's automation requirements. The automation of an industrial process is invariably faced with the decision of the optimal resources for implementation. It is important to find a way of selecting the most efficient programmable logic controller that is suitable to facilitate a particular control process. Variables such as the number of inputs or outputs, program memory size, data memory size, auxiliary memory, timers and counters have to be given the requisite consideration. The method undertaken considers the use of ladder diagrams and XML in the process of modeling PLC characteristics which are organized in form of a database. Criteria for PLC selection are used to provide a list of PLCs or embedded controllers that best fits the control process envisioned. Programmable Logic Controllers are at the forefront of manufacturing automation. Many factories use Programmable Logic Controllers to cut production costs and/or increase quality. This paper presents an extensive survey of trends in embedded processor use with an emphasis on automation in industries using embedded system design.

## I.INTRODUCTION

Embedded systems can be regarded today as some of the most lively research and industrial targets. An embedded system is simply a device that contains a computer to provide a very fixed set of functions. Many consumer electronics and office automation products are excellent examples like cell phones, business phones, PBX, television sets; most radios sets, fax machines, printers. Embedded systems can be part of a larger embedded system. An example of this would be a voice mail system that uses a hard drive to store messages. In fact, if a processor is not used in a general purpose computer such as a PC, it likely qualifies as an embedded system.



A microcontroller is a microprocessor with a lot of extras features onboard the same chip as the processor. These added features usually expedite the design process for embedded systems, and thus most microcontrollers are found in embedded applications. Developing an embedded system product has certain key advantages. Developing an embedded system is a labor intensive task, and thus expensive. Once the

product is finished, the cost of manufacture is usually very low, and the cost of development can be spread over the production volume. This dependency on volume to defray the development costs means it is rarely appropriate for on-off type products. Exceptions would be products like satellites, rockets, robotics, essentially any complex control situation that merits the expense to gain the unique benefits.

Embedded system also takes time to develop. Some parts of the development process are not easily expedited regardless of the size of the development effort. Taking short cuts in the development process will likely lead to extended overall development times since they usually add problems later in the project when commitment to a particular direction is significant.

The most profound advantage of an embedded system is the ability to closely tailor the product to the design objectives. The embedded system is developed to a specific set of requirements derived from the intended application. An embedded system will easily surpass any off the shelf product with respect to meeting requirements, just by nature of the design process. The product was design for the application.

## II.CLASSIFICATION OF EMBEDDED SYSTEMS

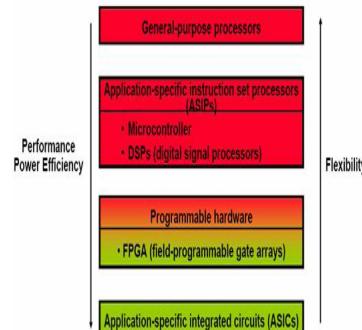


Figure 1: Classification of embedded systems

## III. HOW DOES A PLC COMPARE TO AN EMBEDDED SYSTEM

A PLC (Programmable Logic Controller) is one of the main devices used in industry to implement: monitoring, logic, control, or other events/functions impossible (or complicated) to be done mechanically. With respect to embedded systems, a PLC is in fact an embedded system running a program to provide the various functions PLCs typically provide. Depending on the particular model of PLC, software running on the embedded system likely includes the following functions:

- Interpret a high level command language such as ladder logic and carry out the meaning of those commands.
- Provide an environment to facilitate the programming in a high level language such as ladder logic.
- Provide appropriate communication facilities to allow the PLC to easily communicate with other devices.
- Automatic fault recovery.
- Automatic program execution on startup

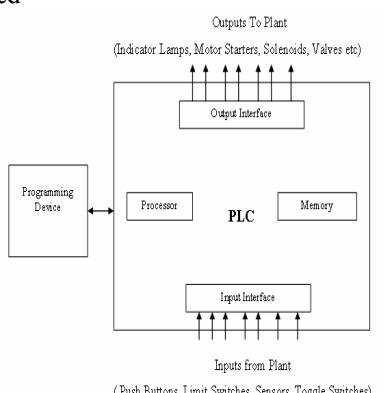
The list of features varies considerably with the make and model of PLC. PLCs are generally packaged to with a basic set of features, which can be further enhanced via expansion modules. A Programmable Logic Controller is a solid-state device, designed to operate in noisy industrial environments and to perform all the logic functions previously achieved using Electro-mechanical relays, drum-switches, mechanical timers and counters.

#### Some major specifications of PLC

- The program can be entered on the factory floor using a programming device.
- Designed for an electrically noisy environment no extra filtering is required.
- Smaller in size.
- Fast in speed.
- More reliable than hardwired systems.
- Modification of program is easy.
- Timers, counters and sequencers are all implemented using software programs.
- No external physical device exists.

#### IV PLC ARCHITECTURE

The architecture of a general PLC is shown in Fig. 2. The main parts of a PLC are its processor, power supply, and input/output (I/O) modules. In a micro PLC, all three main parts are enclosed in a single unit. For larger PLCs, these three parts are separately purchased and combined to form a PLC. The programming device, often a personal computer, connects directly to the processor through a serial port or remotely through a local area network. Depending on the manufacturer, the local area network interface may be built into the processor, or may be a separate module. Many of the PLC local area networks are proprietary to one manufacturer. However, interfaces to standard networks, such as Ethernet, have recently been introduced



#### V. THE CHALLENGE OF EMBEDDED SYSTEM

The explosive growth of electronics in the automotive industry, especially the growth of embedded system software, changes the dynamics of automotive design and presents significant challenges. These new challenges escalate when coupled with the demands of a highly competitive industry. But this new functionality must meet stringent quality standards, at competitive costs. The cost/ quality dynamic is very difficult given the increased complexity, the interactions between them and the software proliferation within them. The trend of increasing automotive electronic content is the direct result of many new features that will greatly increase both safety and comfort but that will require more sophisticated embedded software component.

#### VI CONCLUSION

Embedded system design requires a wide range of talents, from parallel program design to power distribution design. Embedded systems are common now and are bound to become increasingly common as microprocessors become fast and cheap enough to replace custom logic and as customers demand more features. The software side of embedded system design resembles hardware design in that is driven by tight performance and size (memory) constraints.

Tools exist to aid embedded system design, but much work remains to be done to develop standard methodologies akin to the FPGA/ASIC/custom system of design methodologies and the tools to support those methodologies.

Programmable logic controllers and their unique language, ladder logic, are the workhorses of factory automation. Higher-level languages, such as sequential function charts and function blocks, ease the programming task for large systems. However, ladder logic remains the dominant language at present. Any engineer working in a manufacturing environment will at least encounter PLCs and ladder logic, if not use them on a regular basis.

Future work will concentrate on the implementation of the presented method and its application to industrial systems.

#### VII REFERENCES

- [1]. Joseph Sifakis, CNRS/Verimag, FR “Embedded Systems Design – Scientific Challenges and Work Direction”, June 2009
- [2]. Wayne Wolf, Ernest Frey” Embedded System Design” Princeton University, June 1992.
- [3]. Gerrit Muller” Research Agenda for Embedded Systems” 21st February 2008
- [4]. gaudisite.nl/index.html, 1999.
- [5]. programmable logic controller.(2009) in computer desktop encyclopedia
- [6]. M desousa, A. Carvalho MatPLC - the Truly Open Automation Controller”, Proceedings of the 28th Annual Conference of IEEE Industrial Electronics, pp. 2278- 2283, 2002
- [7]. Mader, A.: classification of PLC model andapplications’. Discrete event systems-analysis and control, Proc. WODES2000, 2000, USA
- [8]. International electrotechnical commission, technical committee No.65. programmable controllers IEC 611313 second edition, November 1998.

# Prototype Filter Designs for Cosine Modulated Filter Banks

Neela. R. Rayavarapu, and Neelam Rup Prakash, Member, IEEE  
 Email: neela.rayavarapu@chitkara.edu.in

**Abstract-** The paper discusses several simple design methods that are available for the design of prototype filters for the pseudo-qmf filter bank. As opposed to the earlier methods that were based on the method of nonlinear optimization, several methods not involving nonlinear optimization have been designed. An attempt has been made to obtain a comparative study of some of these methods, in respect of computation cost, computation time, and number of coefficients to be optimized

**Keywords** prototype filter, pseudo-qmf filterbank, nonlinear optimization, computational costs.

## I. INTRODUCTION

Prior to the development of perfect reconstruction filter banks, researchers developed techniques for the design of approximate reconstruction systems called pseudo QMF filter banks. In the case of these systems the analysis/synthesis filters are chosen so that there is no aliasing between adjacent bands, assuming the stopband attenuation of given analysis filters in all the nonadjacent bands is infinite. This kind of approximate reconstruction is acceptable in several practical application such as audio applications.

In wideband audio applications stopband attenuation in nonadjacent channels is required to be greater than -100dB. Relaxing the perfect reconstruction condition it is possible to obtain high stopband attenuation required for such applications. In these filter banks the aliasing is cancelled approximately and the distortion function is approximately a delay. These systems are called pseudo-qmf filter banks. Over the years several techniques have been developed for the design of near-perfect reconstruction filter banks. When the analysis and synthesis filters of the M-channel pseudo-qmf filter bank are cosine-modulated versions of the prototype filter we have a cosine-modulated filter bank[1]. The advantage of the cosine modulated filter bank is that the cost of the analysis filter bank is the cost of one filter plus modulation overheads. Also during the design phase we are required to optimize the coefficients of the prototype filter only. The impulse responses of the analysis filters  $H_k(z)$  and the synthesis filters  $F_k(z)$  can be given as follows.

$$h_k(n) = 2h_0(n) \cos\left(\frac{\pi}{M}(k + 0.5)(n - \frac{N}{2}) + \theta_k\right)$$

$h_0(n)$  is the impulse response of the prototype filter,  $N$  is filter order and  $\theta_k = (-1)^k \cdot \pi/4$ .

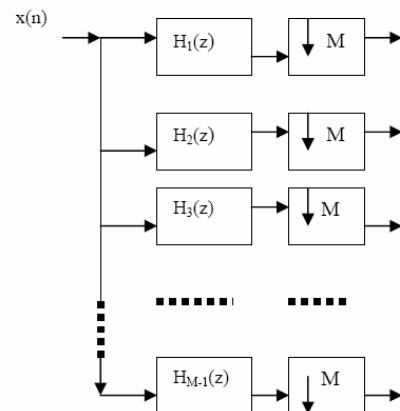


Fig. 1. M-Channel Analysis Filter Bank

$$f_k(n) = 2h_0(n) \cos\left(\frac{\pi}{M}(k + 0.5)(n - \frac{N}{2}) - \theta_k\right)$$

$$0 \leq k \leq M-1$$

The prototype filter  $H_0(z)$  can be designed to have linear phase but the analysis and synthesis filters may not have linear phase. Also since all the analysis and synthesis filters are derived from only one prototype filter the only design freedom that is available to us is the design of this prototype filter.

For approximate reconstruction the following two conditions are required to be satisfied as nearly as possible.

$$|H(\omega)|^2 + |H(\omega-\pi)|^2 = 1 \quad \text{for } 0 < \omega < \pi/M \quad (1)$$

$$|H(\omega)| = 0 \quad \text{for } \omega > \pi/M \quad (2)$$

If equation (1) is satisfied exactly then the amplitude distortion produced by the filter bank is zero. There will be no aliasing between non adjacent subbands if equation(2) is satisfied. While traditional methods of design of prototype filters were based on nonlinear optimization later researchers suggested simple methods of design not involving complex nonlinear optimisation. These methods have some limitations like loss of design flexibility, since parameters such as transition bandwidth cannot be chosen independently. But the ease of computation makes them very attractive. All the methods discussed here are based on coefficient optimization. In this paper we review some of these methods that come up with efficient and computationally simple methods for the design of the prototype filter.

## II. Method 1: THE IFIR FILTER approach

Say for example that it has been determined that the order of the prototype filter required to meet the desired specifications is  $N$ . Now say that instead of trying to meet the

given specifications we meet the two fold stretched specifications. The stretched filter will have a transition band that is twice that of the given filter and hence the order will be  $N/2$ . This means therefore our computations namely additions and multiplications are reduced by a factor of 2. In general FIR filter  $H(z)$  is designed in the frequency domain as a cascade of two FIR sections. The first section generates a sparse set of impulse response values with every  $L$ th sample being non-zero. The other section performs the function of "image-suppression".

Consider a filter  $G(z)$  having an impulse response sequence  $g(n)$ . Inserting the  $L-1$  zeroes between the original samples of  $g(n)$  we obtain the upsampled sequence  $g'(n)$ .

$$g'(n) = \begin{cases} g(n/L) & \text{for } n=iL, i=0,1,2,3 \dots \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

$$G'(z) = G(z^L). \quad (4)$$

$G(z^L)$  is the  $L$  times stretched version of  $G(z)$ . The model filter has a passband and transition band that is  $L$  times that of the desired filter. The frequency response of  $G(z^L)$  is periodic with period  $2\pi/L$ . Any of the passbands in the interval 0 to  $\pi$  may be selected as the desired one. The image suppressor filter  $I(z)$  is therefore used to attenuate the unwanted replicas of the desired passband.

Assume that we want the prototype filter to have a gain of 1 in the passband and of zero in the stopband with a deviation of  $\delta_p$  in the passband and  $\delta_s$ , in the stopband. The passband of the overall response has ripples that are larger than that of  $G(z^L)$  and  $I(z)$ . To meet the desired specifications we take the peak passband ripple to be  $\delta_p/2$  for the model filter and for the image-suppressor, and the stopband ripple to be not greater than  $\delta_s$ . The length of the model filter and the image suppressor required to meet the desired response is calculated using the formula

$$N = (-20 \log \sqrt{\delta_p \delta_s} - 13) / (14.36 * \Delta f) + 1 \quad (5)$$

$$\Delta f = (\omega_p - \omega_s) / 2\pi \text{ denotes transition bandwidth.}$$

Since the transition width is  $L$  times that of a conventional FIR filter the order of the model filter is approximately  $1/L$  times that of the conventional FIR filter. In practice the model filter will have order a little greater than that because it is required to meet more stringent passband requirements. For the IFIR filter required to meet the same set of specifications as a equivalent FIR filter, the IFIR filter requires approximately  $1/L$  times the number of multipliers and adders as the FIR, neglecting of course the image-suppressor. Coefficient sensitiveness and the output round-off noise properties being dependent on the order of the filter these effects will be considerably reduced in the case of the IFIR filter. Now coming to the choice of the stretch factor  $L$ . In [2] the stretch factor  $L$  is chosen as  $M/2$ . As is well known the number of multipliers required for the implementation of the FIR filter is dependent on the filter order  $N$ , which is in turn dependent on the transition width for a chosen values of  $\delta_p$  and  $\delta_s$ . Since the transition width is directly proportional to the stretch factor,  $L$ , we may conclude that the order of the model filter and therefore the number of multipliers for the implementation of the IFIR filter will depend on the value of  $L$ . Mehrnia and

Wilson[3], determined that the value of  $L$  that will yield minimum number of multipliers is given by expression

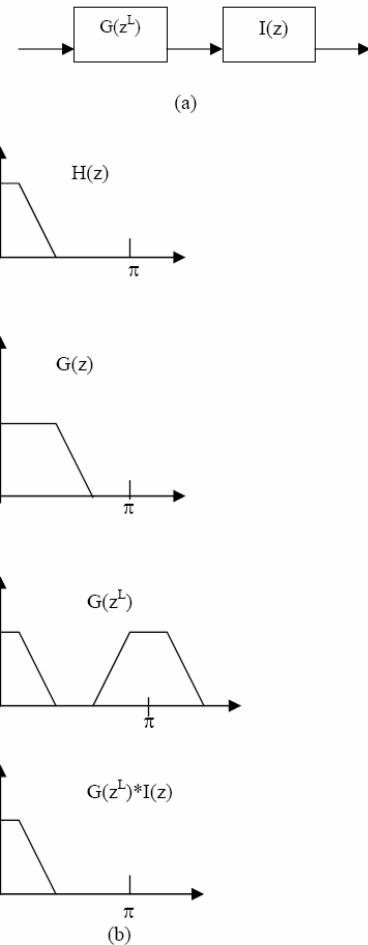


Fig.1(a) Implementation of the IFIR filter. (b) Response of IFIR filter for  $L=2$  at various stages.

$$L_{opt} = 2\pi / (\omega_p + \omega_s + \sqrt{2\pi(\omega_s - \omega_p)}) \quad (6)$$

The optimum choice of  $L$  for minimum multipliers is the value of  $L_{opt}$  rounded off to nearest integer. The design and optimization process for the filter coefficients of the model filter and the image suppression filters may be any of the the procedures that have been reviewed here. We have followed suggested in [4]. For  $M=8$  the order of the model filter was obtained as 32 and that of the interpolator filter was obtained to be 41. Maximum peak ripple in the overlapping passbands was estimated to be  $4 \times 10^{-4}$ .

### III. METHOD 2. KAISER WINDOW APPROACH

In [4] the prototype filters are designed using the Kaiser window approach. The optimization process for the filter coefficients is reduced to a single parameter to be optimized, which in this case is the cutoff frequency. Let  $H(z)$  be the transfer function of the prototype filter. The objective function is chosen to be

$$\Phi_{new} = \max |g(2Mn)| \quad (7)$$

$$\text{Where } G(e^{j\omega}) = |H(e^{j\omega})|^2$$

The objective function which is a convex function of the parameter being optimized , the cutoff frequency, is then minimized by varying this parameter.

For the case of M = 8 the prototype filter had a stopband attenuation of 100dB , The order of the prototype filter was determined to be 116.

This method is conceptually simple and easy to implement. Also using the Kaiser window it is possible to have control over the stop band attenuation.

#### IV METHOD 3.

##### DESIGN USING THE PARKS-MCCELLAN ALGORITHM

In the method suggested by Creusere and Mitra[5] the filter is designed as an equiripple filter using the Parks-McClellan algorithm to satisfy (1) and (2) . Here the pass band edge varied to minimize the cost function which is the maximum value of the ripple in the overlapping pass bands. The pass band error is weighted more heavily than the stop band error when using the Parks-McClellan algorithm. Filter length was obtained to be 128 for M=8, and 516 for M=32. The stop band attenuation was obtained to be more than 100 dB. The overlapped pass band ripple in the 8 band filter bank was obtained to be .11dB .

#### V METHOD 4: WINDOW METHOD.

The method suggested Fernando Cruz-Roldan et al, in [6]involves the design of the prototype filter by using any window. Then the 6dB cut off frequency is modified in order to obtain the 3dB cut off frequency at  $\pi/2M$ . The optimization procedure involves the adjustment of the 6db cutoff frequency to find the best impulse response sequence  $h[n]$  that yields thesmallest

$$\Phi = \left\| H(e^{j\pi/2M}) \right\| - 1/\sqrt{2}$$

The objective function therefore is the deviation between the actual value of the magnitude at the 3 db frequency  $\pi/2M$  and the ideal value which is .707.

The order of the filter that is required for number of channels M = 8,16,32 and 64 is obtained and tabulated as in Table1. This will give us the number of filter coefficients that need to be optimized. For example for M=32 the prototype filter has a length of 439 and has stopband attenuation of about 100 dB. The maximum aliasing error was obtained as  $2.611 \times 10^{-7}$ .

Table 1. Number of coefficients to be optimized

	No. of channels			
	8	16	32	64
Method 1	32	46	62	79
Method 2.	128	255	511	1021
Method 3.	116	233	466	932
Method 4.	108	215	430	860

Table 2. Comparison of Computation for M=8

Quantity Compared	Method 2	IFIR Method		
		G(z)	I(z)	Total
Filter Order	128	32	41	73
Multiplications	64	16	21	37
Additions	128	32	41	73

Tables 1 and 2 show the amount of saving obtained as a result of substitution of FIR approach with the IFIR filter.

#### VII CONCLUSION

We have reviewed in this paper designs of prototype filters that are based on coefficient optimization. Comparison of the methods is based on the number of coefficients that are required to be optimized and hence the cost of computation involved. The IFIR approach has been found to yield filters of much smaller order than the FIR filter used in the other three methods. In the case of Method2 the order is the largest. Also the deviation from zero was found to be comparable in all the methods. In respect of savings in computational time and cost only the IFIR method has been found to obtain significant savings when compared to Method 2. Since in the other 2 methods the order of the filters is in the same range as in Method 2 the factors will not be significantly different.

#### VIII. REFERENCES.

- [1]. P.P Vaidyanathan, Multirate Systems and Filter Banks. Englewood Cliffs, NJ: Prentice- Hall,1993
- [2]. Zijing Zhang and Licheng Jiao, "A Simple Method for Designing Pseudo QMF Banks". ICCT2000
- [3]. Alireza Mehrnia and Alan N Wilson,Jr. " An Optimal IFIR filter design," ISCAS 2004,pp. 133-136.
- [4]. Yuan-pei Lin and P.P. Vaidyanathan, " A Kaiser window approach for the design of prototype filters of cosine modulated filterbanks". IEEE Signal Processing Letters, vol5 pp.132-134, June 1998.
- [5]. C.D. Creusere and S.K. Mitra, "A Simple method for designing high quality prototype filters for M-band pseudo QMF banks," IEEE Trans. Signal Processing, vol 43, pp.1005-1007, April 1994
- [6]. Fernando Cruz-Roldan, Pedro Amo-lopez, " An efficient and simple method for designing prototype filters for cosine-modulated pseudo-qmf banks," IEEE Signal Processing letters, Vol.9, January 2002, pp. 29-31.

# Acoustic ECHO Cancellation using LMS

Amit Munjal-Assistant Prof., Electronics and Communication Engg, CIET Rajpura, Punjab, India  
amit.munjal@chitkara.edu.in

**Abstract-** Acoustic echo occurs when an audio signal is reverberated in a real environment, resulting in the original intended signal plus attenuated, time delayed images of this signal. Acoustic echo cancellation is a common occurrence in today's telecommunication systems. It occurs when an audio source and sink operate in full duplex mode; an example of this is a hands-free loudspeaker telephone. In this situation the received signal is output through the telephone loudspeaker (audio source), this audio signal is then reverberated through the physical environment and picked up by the system's microphone (audio sink). The effect is the return to the distant user of time delayed and attenuated images of their original speech signal. The signal interference caused by acoustic echo is distracting to both users and causes a reduction in the quality of the communication. This paper focuses on the use of RLS algorithm to reduce this unwanted echo, thus increasing communication quality.

## I.INTRODUCTION

Acoustic echo occurs when an audio signal is reverberated in a real environment, resulting in the original intended signal plus attenuated, time-delayed images of this signal. This paper will focus on the occurrence of acoustic echo in telecommunication systems. Such a system consists of coupled acoustic input and output devices, both of which are active concurrently. An example of this is a hands-free telephony system. In this scenario the system has both an active loudspeaker and microphone input operating simultaneously. The system then acts as both a receiver and transmitter in full duplex mode. When a signal is received by the system, it is output through the loudspeaker into an acoustic environment. This signal is reverberated within the environment and returned to the system via the microphone input. These reverberated signals contain time-delayed images of the original signal, which are then returned to the original sender (Figure 1.1,  $a_k$  is the attenuation,  $t_k$  is time delay). The occurrence of acoustic echo in speech transmission causes signal interference and reduced quality of communication. The method used to cancel the echo signal is known as adaptive filtering.

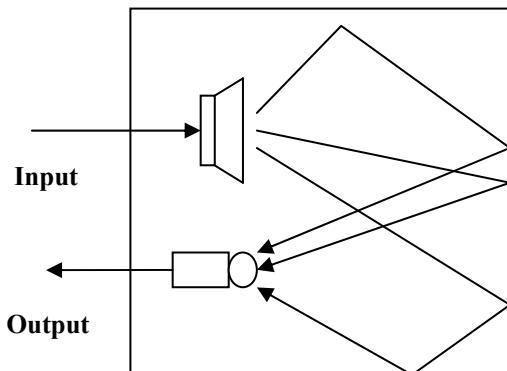


Figure 1.1: Origins of acoustic echo.

Adaptive filters are dynamic filters, which iteratively alter their characteristics in order to achieve an optimal desired output. An adaptive filter algorithmically alters its parameters in order to minimize a function of the difference between the desired output  $d(n)$  its actual output  $y(n)$ . This function is known as the cost function of the adaptive algorithm. Figure 1.2 shows a block diagram of the adaptive echo cancellation system implemented throughout this paper. Here the filter  $H(n)$  represents the impulse response of the acoustic environment,  $W(n)$  represents the adaptive filter used to cancel the echo signal. The adaptive filter aims to equate its output  $y(n)$  to the desired output  $d(n)$  (the signal reverberated within the acoustic environment). At each iteration the error signal,  $e(n) = d(n) - y(n)$  is fed back into the filter, where the filter characteristics are altered accordingly.[1]

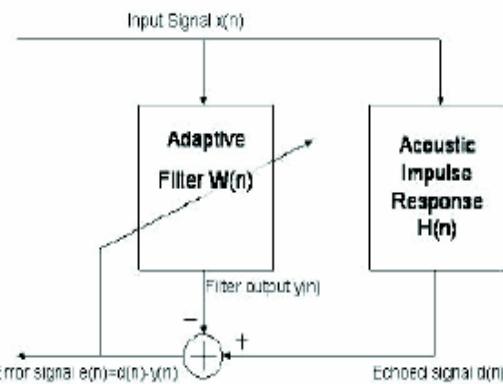


Figure 1.2: Block diagram of an adaptive echo cancellation system.

The aim of an adaptive filter is to calculate the difference between the desired signal and the adaptive filter output,  $e(n)$ . This error signal is fed back into the adaptive filter and its coefficients are changed algorithmically in order to minimize a function of this difference, known as the cost function. In the case of acoustic echo cancellation, the optimal output of the adaptive filter is equal in value to the unwanted echoed signal. When the adaptive filter output is equal to desired signal the error signal goes to zero. In this situation the echoed signal would be completely cancelled and the far user would not hear any of their original speech returned to them. [4]

## II.LEAST MEAN SQUARES (LMS) ALGORITHM.

The Least Mean Square (LMS) algorithm was first developed by Widrow and Hoff in 1959. The LMS algorithm is an important member of stochastic gradient algorithms. It utilizes the gradient vector of the filter tap weights to converge on the optimal wiener solution. It is well known and widely used due to its computational simplicity. It is this simplicity that has made it the benchmark against which all other adaptive

filtering algorithms The LMS algorithm is a linear adaptive filtering algorithm which consists of two basic processes

1. A filtering process which involves computing the output of a linear filter in response to an input signal and generates an estimation error by comparing this output with a desired response.
2. An adaptive process which involves the automatic adjustment of the parameters of the filter in accordance with the estimation error.

With each iteration of the LMS algorithm, the filter tap weights of the adaptive filter are updated according to the following formula [5,32].

$$w(n+1) = w(n) + 2\mu e(n)x(n)$$

Here  $x(n)$  is the input vector of time delayed input values,  $x(n) = [x(n) \ x(n-1) \ x(n-2) \ \dots \ x(n-N+1)]^T$ . The vector  $w(n) = [w_0(n) \ w_1(n) \ w_2(n) \ \dots \ w_{N-1}(n)]^T$  represents the coefficients of the adaptive FIR filter tap weight vector at time n. The parameter  $\mu$  is known as the step size parameter and is a small positive constant. This step size parameter controls the influence of the updating factor. Selection of a suitable value for  $\mu$  is imperative to the performance of the LMS algorithm, if the value is too small the time the adaptive filter takes to converge on the optimal solution will be too long; if  $\mu$  is too large the adaptive filter becomes unstable and its output diverges.

## 2.2 Derivation of the LMS algorithm.

The derivation of the LMS algorithm builds upon the theory of the wiener solution for the optimal filter tap weights,  $w_0$ , as outlined in the previous section [23,24]. It also depends on the steepest descent algorithm as stated in equation 3.8, this is a formula which updates the filter coefficients using the current tap weight vector and the current gradient of the cost function with respect to the filter tap weight coefficient vector,  $\nabla \xi(n)$ .

$$w(n+1) = w(n) - \mu \nabla \xi(n)$$

$$\text{where } \xi(n) = E[e^2(n)]$$

As the negative gradient vector points in the direction of steepest descent for the N dimensional quadratic cost function, each recursion shifts the value of the filter coefficients closer toward their optimum value, which corresponds to the minimum achievable value of the cost function,  $\xi(n)$  [5,32].

The LMS algorithm is a random process implementation of the steepest descent algorithm, from equation 3.8. Here the expectation for the error signal is not known so the instantaneous value is used as an estimate. The steepest descent algorithm then becomes equation 3.9.

$$w(n+1) = w(n) - \mu \nabla \xi(n)$$

$$\text{where } \xi(n) = e^2(n)$$

The gradient of the cost function,  $\nabla \xi(n)$ , can alternatively be expressed in the following form.

$$\begin{aligned} \nabla \xi(n) &= \nabla(e^2(n)) \\ &= \frac{\partial e^2(n)}{\partial w} \end{aligned}$$

$$\begin{aligned} &= 2e(n) \frac{\partial e(n)}{\partial w} \\ &= 2e(n) \frac{\partial(d(n) - y(n))}{\partial w} \\ &= -2e(n) \frac{\partial e w^T(n) - x(n)}{\partial w} \\ &= -2e(n)x(n) \end{aligned}$$

Substituting this into the steepest descent algorithm of equation 3.8, we arrive at the recursion for the LMS adaptive algorithm.

$$w(n+1) = w(n) + 2\mu e(n)x(n)$$

## 2.3 Implementation of the LMS algorithm.

Each iteration of the LMS algorithm requires 3 distinct steps in this order:

1. The output of the FIR filter,  $y(n)$  is calculated using equation 3.12.

$$y(n) = \sum_{i=0}^{N-1} w(n)x(n-i) = w^T(n)x(n)$$

2. The value of the error estimation is calculated using equation 3.13.

$$e(n) = d(n) - y(n)$$

3. The tap weights of the FIR vector are updated in preparation for the next iteration, by following equation.

$$w(n+1) = w(n) + 2\mu e(n)x(n)$$

The main reason for the LMS algorithms popularity in adaptive filtering is its computational simplicity, making it easier to implement than all other commonly used adaptive algorithms. For each iteration the LMS algorithm requires  $2N$  additions and  $2N+1$  multiplications ( $N$  for calculating the output,  $y(n)$ , one for  $2\mu e(n)$  and an additional  $N$  for the scalar by vector multiplication) [5,32]

## III.CONCLUSION

The success of the echo cancellation can be determined by the ratio of the desired signal and the error signal. The average attenuation for this simulation of the LMS algorithm is -25.2144 dB. This is the simplest to implement and is stable when the step size parameter is selected appropriately. This requires prior knowledge of the input signal which is not feasible for the echo cancellation system. Thus we need to select other adaptive algorithms which should provide higher attenuation and as well as Higher ratio of desired signal to error signal. Some of the algorithms include NLMS, VSNLMS, VSLMS and RLS etc.

## IV.REFERENCES

- [1]. Haykin, Simon. 1991, Adaptive Filter Theory. 2nd Edition. Prentice-Hall Inc., New Jersey.
- [2]. Farhang-Boroujeny, B. 1999, Adaptive Filters, Theory and Applications. John Wiley and Sons, New York.
- [3]. Diniz, Paulo S. R. 1997, Adaptive Filtering, Algorithms and Practical Implementation. Kluwer Academic Publishers, Boston.
- [4]. Bellanger, Maurice G., 2001. "Adaptive Digital Filters", 2nd edition. Marcel Dekker Inc., New York.
- [5]. Chassaing, Rulph. 2002, "DSP applications using C" John Wiley and Sons, New York.

# Content Based Image Retrieval and Resizing of Image Using Discrete Cosine Transform

P.K. Deshmane- Student M.E.II, Sinhgad College of Engineering, Pune

Prof. S.R. Ganorkar- Professor, Sinhgad College of Engineering, Pune

**Abstract -**Retrieval of a query image from a large database of images is an important task in the area of computer vision and image processing. A number of good search engines are available today for retrieving the image, but there are not many fast tools to retrieve intensity and color images. Thus there is continued need to develop efficient algorithms in image mining and content based image retrieval and resizing. In content based image retrieval and resizing system (CBIR & R) and resizing systems, the images are searched and retrieved based on the visual content of the images. In the first part of CBIR & R system, the images from the image database are processed offline. The features from each image in the image database are extracted to form the metadata information of the image, in order to describe the image using its visual content features. Next these features are used to index the image, and they are stored into the metadata database along with the images. In the second part, the retrieval process is depicted. The query image is analyzed to extract the visual features, and these features are used to retrieve the similar images from the image database. Rather than directly comparing two images, similarity of the visual features of the query image is measured with the features of each image stored in the metadata database as their signatures. The retrieval systems returns the most matching image and resize the retrieved image to half of double of its original size.

## I.INTRODUCTION

Retrieval of a query image from a large database of images is an important task in the area of computer vision and image processing. The advent of large multimedia collection and digital libraries has led to an important requirement for development of search tools for indexing and retrieving information from them. Many image attributes such as color, shape are having direct correlation with semantics embedded in the image. Image retrieval using similarity measures is an elegant technique used in content-based image retrieval (CBIR). To a very large extent, the low-level image features such as color, texture, and shape are widely used for CBIR. While attempting the task of image retrieval, we identify the mutual correspondence between two images in a set of database images using similarity relations. The content-based query system processes a query image and assigns this unknown image to the closest possible image available in the database.

## II. RELATED WORK

An image may have one or more major regions. For image identification and retrieval, we need to segment the regions from the background before we can accurately describe image. The architecture for a possible content-based image retrieval system is shown in Figure 1. The CBIR systems architecture is essentially divided into two parts. In the first part, the images from the image database are processed offline. The features from each image in the image database are extracted to form the metadata information of the image, in order to describe the

image using its visual content features. Next these features are used to index the image, and they are stored into the metadata database along with the images. In the second part, the retrieval process is depicted. The query image is analyzed to extract the visual features, and these features are used to retrieve the similar images from the image database. Rather than directly comparing two images, similarity of the visual features of the query image is measured with the features of each image stored in the metadata database as their signatures. Often the similarity of two images is measured by computing the distance between the feature vectors of the two images. The retrieval systems return the first  $k$  images, whose distance from the query features have been used to index images for content-based image retrieval systems. Most popular among them are color; image is below some defined threshold. Several image

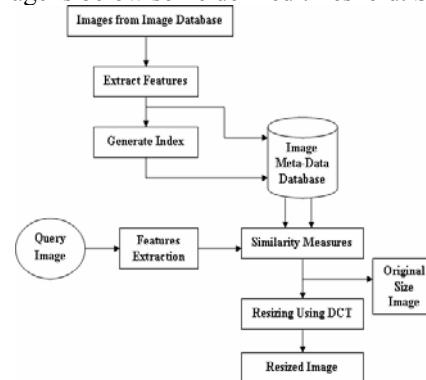


Fig1: Architecture of CBIR & R System

texture, shape, image topology, color layout, region of interest, etc..

### A. Image Region Extraction

The pixels corresponding to different regions are normally clustered spatially and have a certain shape. Each pixel of the image can be represented as a point in 3-D color space. Commonly used color spaces for image retrieval include RGB, Munsell, CIE L\*a\*b\*, CIEL\*u\*v\*, HSV (or HSL HSB), and the opponent color space. It is Difficult to determine which color space is the best for tackling the problemWe select the RGB color space, RGB color space is the most used color space for computer graphics. Note that R, G, and B stand here for intensities of the Red, Green, & Blue guns in a CRT, not for primaries as meant in the CIE RGB space. It is an additive color space: red, green, and blue light are combined to create other colors.

Clustering is a fundamental approach in pattern recognition. The color clustering algorithm that we developed is described as follows:

- (1)Transform the image to size 64\*64

- (2) Obtain the RGB components of an image
- (3) Find all color clusters
  - (i) Compute the color distance of each pixel from the existing color clusters. If no color clusters exist, then set the first pixel as a new cluster. The color distance is given by:
  - (ii) If the minimum color distance is less than the pre-set threshold, then a match is found. Otherwise, a new color cluster is generated, and set the unmatched pixel as the new cluster.
  - (iii) For each match, the R, G, B values and the population of the cluster are updated. The new representative color of the cluster is the weighted average of the original cluster and the color of the current pixel.
- (3) Compute the population of every cluster. The clusters with a population of less than a threshold are discarded.
- (4) For each pixel, compute the color distance to different clusters. Assign the pixel to the cluster to which the color distance is minimum. We consider each cluster as an image layer and each pixel is assigned to one image layer. Fig.3 shows the image layers of the flower image

If the layer's color belongs to a background color, we discard the whole layer. Due to diversities of flowers, different illumination conditions, and noise introduced in acquiring the image, we have to consider the following three situations:

- 1) There is no flower region remained. We will bring back the largest cluster and label it as a main object region. Normally, object region should dominate the image and hence it is quite safe to assume that the largest cluster is an object region.
- 2) Some background regions are kept as object regions. In general, there are no object regions spread on the narrow peripheral zone. The clusters locating in the narrow peripheral area will be considered as background regions and removed.
- 3) There are small noise blocks in the object region. Those small blocks in a object layer with size of small than a pre-set threshold (1/10 of size of the largest flower region in the layer) will be removed. To extract the shape features, the contour of a object region is extracted based on the segmentation. Fig.5 shows the flowchart of our flower image segmentation approach based on clustering and domain knowledge.

### B. Shape Features

Shape is one of the most important features characterizing an object. Many investigations in shape representation such as chain codes, centroid-contour distance (CCD) curve, and medial axis transform (MAT), Wavelet descriptions, moment invariants, and deformable templates, had been carried out.

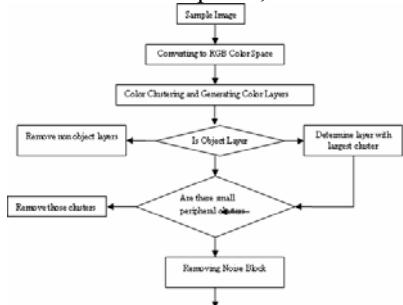


Fig2: Flowchart for color cluster analysis

All of these features perform well and have advantages for some applications. An important criterion for a good shape representation is that the representation has to be invariant to rotation, scaling, and translation. In this paper, we use two shape features, CCD and angle code histogram (ACH).

### C. Centroid-Contour Distance (CCD)

Centroid-contour distance (CCD) can reflect the global character of a shape, but the CCD curve is neither scaling nor rotation invariant. Actually, the number of contour points is dependent on the object size. Consequently, the number of CCD curve sample points and the amplitude of CCD samples will change if the scale of the object changes. The key for a similarity measure with CCD curves to be rotation invariant is to locate fixed starting point of CCD curves. In order to solve this problem, we set the farthest point from the centroid as the start point for each data sample in the database. In retrieving image with a query image, we select several farthest points from the centroid possible start points. The difference between two CCD curves is computed when a possible start point of an enquiry image is aligned with the start point of the database image. The smallest difference between two CCD curves among all possible start points is used to measure the dissimilarity of two contours.

### D. Angle Code Histogram (ACH)

It was observed that the CCD curve cannot characterize local properties of a contour effectively. However, local properties are very important for the identification of flower shapes. proposed an angle code method for shape characterization. In their approach, each closed contour is represented by a sequence of line segments with two successive line segments forming an angle. The angles at contour points on each closed contour were computed and the resulting sequence of successive angles was used to characterize the contour. The retrieval process was performed by matching the angle code string. However, flower images are quite different from artificially generated graphics that have ideal lines or arcs. Following the idea of the angle code, we computed the angle for each contour point based on two approximate lines coming to and leaving the point. If the distributions of the angle codes of two closed contours are close, they will have similar local features. We propose to use an angle code histogram (ACH) to characterize the local properties of a image. If the distributions of the angle codes of two contours are similar, they will have similar local properties. The difference between two angle code histograms is defined as:

$$d = \sum_{j=1}^m |h_j^{(1)} - h_j^{(2)}|$$

Where  $m$  is the number of bins in which the angle code histogram is partitioned.

### E. Multidimensional Indexing

Multidimensional indexing is an important component of content-based image retrieval. In the information retrieval community, the indexing mechanism is concerned with the process to assign terms to a document so that the document can be retrieved based on these terms. The indexing in content-based image retrieval similar to the notion adopted in the

information retrieval. The primary concern of indexing is to assign a suitable description to the data in order to detect the information content of the data. As we explained in the previous sections, the descriptors of the multimedia data are extracted based on certain features or feature vectors of the data. These content descriptors are then organized into a suitable access structure for retrieval.

#### *F. Image Retrieval*

In CBIR, the dimensionality of feature vectors is normally very high. Before indexing, it is very important to reduce the dimensionality of the feature vectors. The most popular approach to reduce high dimensionality is application of the principal component analysis, based on singular decomposition of the feature matrices. The theory behind singular value decomposition of a matrix and generation of principal components for reduction of high dimensionality has been discussed later.. The technique has also been elaborate with regard to text mining. This can be applied to both text and image data types in order to reduce the high dimensionality of the feature vectors and hence simplify the access structure for indexing the multimedia data. After dimensionality reduction, it is very essential to select an appropriate multidimensional indexing data structure and algorithm to index the feature vectors. There have been some limited efforts in this direction. Multimedia database indexing particularly suitable for data mining applications remains a challenge. So exploration of new efficient indexing schemes and their data structures will continue to be a challenge for the future. After indexing of images in the image database, it is important to use a proper similarity measure for their retrieval from the database. Similarity measures based on statistical analysis have been dominant in CBIR. Distance measures such as Euclidean distance and similar techniques have been used for similarity measures. Distance of histograms and histogram intersection methods have also been used for this purpose, particularly with color features. Another aspect of indexing and searching is to have minimum disk latency while retrieving similar objects. Chang et al. proposed a clustering technique to cluster similar data on disk to achieve this goal, and they applied a hashing technique to index the clusters. In spite of lots of development in this area, finding new and improved similarity measures still remains a topic of interest in computer science, statistics, and applied mathematics.

#### *G. Image Resizing*

Scalability of an image representation is required in various applications, such as transmission, storage, retrieval, and display of digital images. One could directly resize the image in the spatial domain using various interpolation techniques. But for efficient storage, images are usually represented in the transform domain as compressed data. It is thus of interest to develop resizing algorithms directly in the compressed stream. As discrete cosine transform (DCT)-based JPEG standard is widely used for image compression, a number of approaches have been advanced to resize the images in the DCT space. In this work, we propose a modification to Dugad and Ahuja algorithm. In our approach, during doubling of the images, it is not necessary to go back to the spatial

domain, if the objective is to get the final result in the spatial domain. It may be noted, however, that Dugad and Ahuja also suggested similar operation. In our work, we have further used a 16\*16 DCT transform for obtaining the coefficients in the transformed space for the up sampled image. It should be noted that Dugad and Ahuja also presented an efficient implementation of their algorithm. In this case, they have used direct matrix multiplication and addition for converting a block (or a set of blocks) of DCT coefficients to a set of blocks (or a block) of DCT coefficients in the resulting image. A similar computation scheme has also been developed for our approaches.

For halving an image in its compressed form (the DCT based JPEG standard), in the first step, the image reduced in size in the spatial domain, is obtained. This is carried out by considering the 4\*4 lower frequency-terms and applying a 4-point inverse DCT (IDCT) on them. Hence, from 8\*8 blocks, one gets 4\*4 blocks in the spatial domain. In the next stage, this image (in the spatial domain) is once again compressed by 8\*8 block DCT encoding (JPEG standard). The algorithm is described below.

For doubling the images, first the DCT encoded image is transformed to its spatial domain. Then for each 4\*4 block, the DCT coefficients are computed applying a 4-point DCT. These 4\*4 DCT coefficients are directly used as the low-frequency components of 8\*8 blocks, which are subsequently converted to an 8\*8 block in spatial domain by applying an 8-point IDCT.

### III. EXPERIMENTAL RESULTS

This approach has been evaluated on a flower image as flower images are most colorful & shapeup.

We used (1) the color feature (2) shape features to analyze flower images, out of which color cluster analysis for the separating the different colors in the image give the successful results. The original image and the different image layers are as shown in figure 3. X-Y coordinates of the all the pixels from same color layer are obtained and placed in their respective positions, in this way the number of images of different color layers are obtained as shown in figure.

### IV. CONCLUSION

In this paper, we first present an effective method to segment object regions from images based on color clustering and domain knowledge. The Centroid-Contour Distance (CCD) and the Angle Code Histogram (ACH) of the contour. Experimental results on some flower images showed that our approach performs well in terms of color cluster analysis of different objects. Also, this project provides an effective method for the resizing of retrieved image to half of double of its original size.

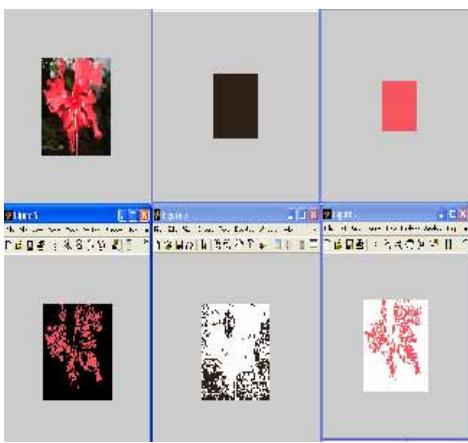


Fig 3: Original image and Different image layers

#### V. REFERENCES

- [1]. S. Mitra and T. Acharya. Data Mining: Multimedia Soft Computing and Bioinformatics. Wiley, Hoboken, N J, 2003.
- [2]. Y. P. Tan, "Content-based Multimedia Analysis and Retrieval," in Information Technology: Principles and Applications, Ed. A. K. Ray and T. Acharya, 233-259, Prentice Hall India, New Delhi, 2004.
- [3]. R. Dugad and N. Ahuja, "A fast scheme for image size change in the compressed domain," IEEE Trans. Circuits Syst. Video Technol., vol. 11, pp. 461–474, Apr. 2001.
- [5]. T. Deselaers. Features for image retrieval. Diploma thesis, Lehrstuhl für Informatik VI, RWTH Aachen University, Aachen, Germany, Dec. 2003.
- [7]. Flickner, M., Sawhney, H., Niblack, W., Ashley, J., et al: Query by image and video content: The QBIC system. IEEE Computer, 28 (Sept. 1995) 23-32
- [9]. Gupta, A., Jain, R.: Visual information retrieval. Comm. Assoc. Comp. Mach., 40 (May 1997) 70-79
- [10]. Pentland, A., Picard, R., Sclaro, S.: Photo book: Content-based manipulation of image databases. Int. J. Comp. Vis., 18 (1996) 233-54
- [11]. Smith, J. R., Chang, S.-F: Single color extraction and image query. In Proc. IEEE Int. Conf. on Image Processing (1995) 528-531
- [12]. Lipson, P., Grimson, E., Sinha, P.: Conguration based scene classification and image indexing. In Proc. IEEE Comp. Soc. Conf. Comp. Vis. and Patt. Rec., (1997) 1007-1013
- [13]. J. Mukherjee and S. K. Mitra, "Image resizing in the compressed Domainusing subband DCT," IEEE Trans. Circuits Syst. Video chnol. vol. 12, no. 7, pp. 620– 627, Jul. 2002.