International Journal of Electronics Engineering Research.
ISSN 0975-6450 Volume 9, Number 4 (2017) pp. 575-586

http://www.ripublication.com

An Effective Reconstruction of Replica Memory Design Optimization for Embedded System

Ravi Khatwal
Senior Researcher, Department of computer science
Mohan Lal Sukhadia University, Udaipur, India.

Manoj Kumar Jain
Professor, Department of computer science
Mohan Lal Sukhadia University, Udaipur, India.

Abstract

Application specific custom memory architecture requires implementing the behavior of system performance in various aspects. Application specific replica cell designed, analyzed with lower cost and delay which is used in ASIP (application specific instruction set processor) and High level synthesis environment. By the help of replica cell design operational computing data storage frequency is analyzed according to operational optimal degree or shortest data transition path is effective contribution in storage optimization section. We have also analyzed column circuitry replica design architecture used in low power embedded devices. This reconstructive replica cell architecture is improving the operational computing frequency with low power consumption. Replica implemented memory cell design reduces the delay time or cost which is highly required for ASIP and High level synthesis environment.

Keywords: HLS Design; Interpolated memory design; Application Specific architecture; Embedded system; ASIP architecture; Replica cell architecture.

1. INTRODUCTION

Embedded processor and memory evolution mechanism are required for an increasing the number of features at lower power and integration design into a single chip. The Embedded system challenges are implemented with reduction of power consumption and integration of heterogeneous systems into the single chip to reduce area, power

and delay [Fig. 1]. In HLS design section memory design synthesis implemented with the help of hardware implementation techniques [Fig. 2]. After synthesis the register-transfer-level (RTL) code implements the same behavior in targeted hardware environments. Currently Multi-core and/or multi-threaded system requires high speed bus-bandwidth architectures used for achieving high performance with low power consumption. An efficient embedded system implements computational task according to our hardware (memory and processor) availability which can reduce the challenges like in terms of scheduling time, low power, temperature, scalability, design complexity, efficiency, flexibility and heterogeneity.

Various techniques are available which can uses for implementing the memory design according to their operational optimal complexity and effective memory design analyzed for HLS and ASIP area. An effective replica memory system is used to reduce the operational complexity with the help of internal hardware implementation mechanism [Fig. 2]. Lots of memory implementation techniques are available which can implement the storage and operational computing according to requirement of global market demand. A. Punitha and M. Joseph [1] proposed a method of effective temperature aware scheduling mechanism with incremental binding for all IC design. During this method, temperature awareness is incorporated in high level and physical level synthesis design. O. Esko, P. Jaaskelainen, P. Huerta, C. S.D L. Lama, J. Takala and J.I. Martinez [2] Proposed and evaluated an application specific processor design architecture that uses a multi issue exposed data path processor architecture template design. M. Kudlur, K. Fan, M. Chu, R. Ravindran, N. Clark, S. Mahlke [3] proposed foresighted latency aware scheduling heuristic [FLASH] mechanism that performs look ahead across scheduling steps to estimate the effects of a potential binding and reduces the compile time and implements the data path design. B. Gorjiara and D. Gajski [4] analyzed optimization techniques such as software transformations, operation chaining, Data path pipelining, controller pipelining, and functional unit customization to improve the quality of the ASIP design. B.H. Calhoun and A.P. Chandrakasan [5] proposed low-voltage operation of traditional six–transistor (6T) SRAM. K. Dhanumjaya, M. Sudha, M.N. Giri Prasad, K. Padmaraju [6] presented the dynamic column based power supply of 8T SRAM cell. N-curve metric have analyzed for verifying the read stability and write ability.

G.K. Chen, D. Sylvester, D. Blauw and T. Mudge [7] compared the 6T and 8T bit cell behavior. T.T.H. Kim and Z.H. Kang [8] analyzed the impacts of NBTI and PBTI on SRAM Vmin and presented a design solution for mitigating the impact of NBTI and PBTI on SRAM Vmin. N.K. Shukla, R.K. Singh and M. Pattanaik [9] presented a novel structure of the SRAM Bit-Cell, called as P4-SRAM Bit-Cell structure. The proposed bit-cell utilizes the Gated-VDD technique for transistor stacking in the PP-SRAM along with the full-supply body biasing to reduce the active, standby, and dynamic power in the memory. S. Mukhopadhyay, H. Mahmoodi and K. Roy [10] analyzed and modeled failure probabilities (access-time failure, read/write failure, and hold failure) of synchronous random-access memory (SRAM) cells due to process-parameter variations. A.K. Dadoria, A.S.Yadav, C.M. Roy [11] analyzed the comparison of the different type of SRAM topology. S. Vijayyan and P. Kodavanti

[12] proposed a novel method that reduces power consumption of SRAM cell, standard 6T (6 transistors) are used by which the leakage current is drastically reduced and low power is achieved. H. Yamauchi [13] analyzed the comparisons of area scaling trends of various SRAM and discussed the margin-assist solutions for VT variability of SRAM.

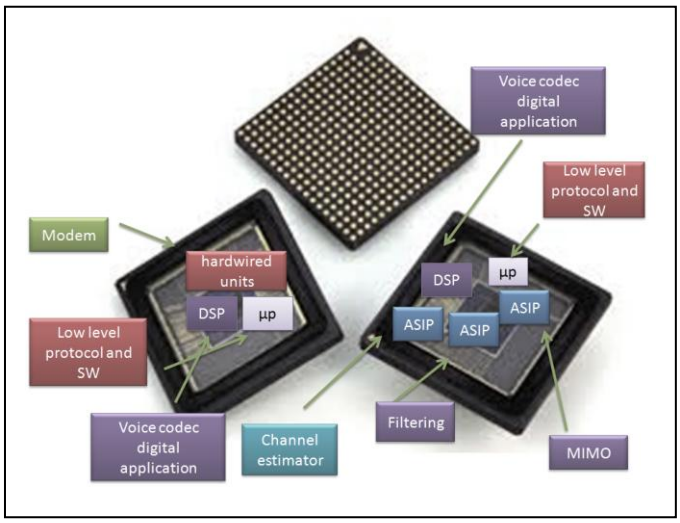

Fig. 1. Application Specific Embedded System

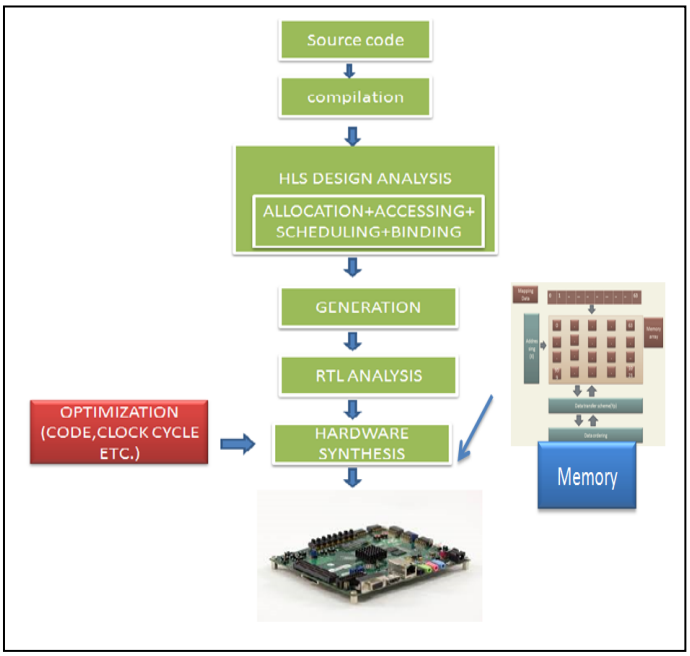

Fig. 2. HLS Design Analysis

2. MEMORY DESIGN IMPLEMENTATION BY REPLICA CELL ARCHITECTURE

The Replica has basically reorganized memory cell architecture designed with lower cost and delay. The replica cell design implements memory internal cell architecture with lower cost, delay and it's reorganized with bus bandwidth implementation mechanism. Data transitions speeds have been implementing with the help of replica cell design [Fig. 3 and Fig. 4]. Replica cell design also implements the storage mechanism due to re-organized memory array and it may be implemented with reduction of LUT's cell design in xup-5 FPGA platform. The replica circuit design is implemented with partitioned in, such as a forest of trees and each tree is separately mapped into circuits of K-inputs LUTs. When these circuits are assembled to form the circuits implementing the network, the replication logic may reduce the total many LUTs in the final circuit. Suppose that three LUTS is required to implement the network as the output of LUTs. Replica cell circuit design is required only two LUTs to implement the network and also reduces delay time. Replica cell combination implemented with the help of xup-5 FPGA board and analyzed cell combination in the form of x1 to x6 cell combination with respect to replica design [Fig. 6].

When the original network contains many nodes, the replicated reorganized memory used a greedy local optimization techniques applied at every node. If the destination tree of the first node is the sources or destination tree of a different node, there can integration among the replication logic at the two nodes [Fig. 5]. The overall success of the replication optimization depends on the order in that it applies to the nodes [Fig. 3] means we have reorganized memory cell to replica design pattern and then analyzed its performance according to data transition or storage design pattern with the help of xup-5 FPGA [Fig. 4 and Fig. 5]. The replica cell is implemented in the SRAM memory area and Replica cell is replaced with SRAM. The internal cell of SRAM half and full cell section replaced by replica cell. After simulation we get lower data transition time or cell delay or replica cell reduces the cache miss conditions or decoder requirements conditions then we get better results required for high performing computing system [Fig. 6].

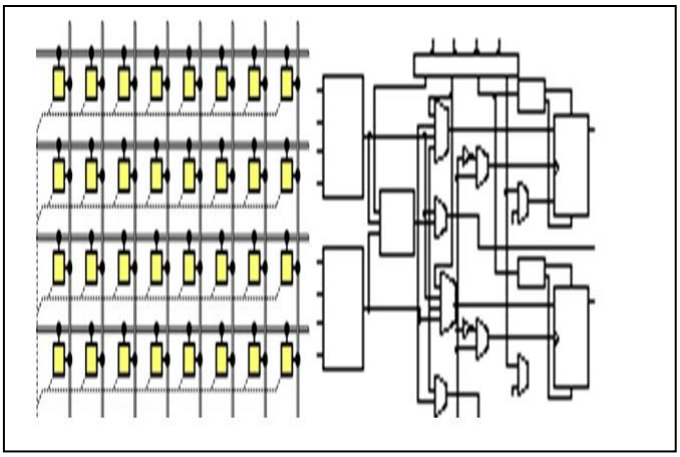

Fig. 3. Hardware for Data storage in memory cell

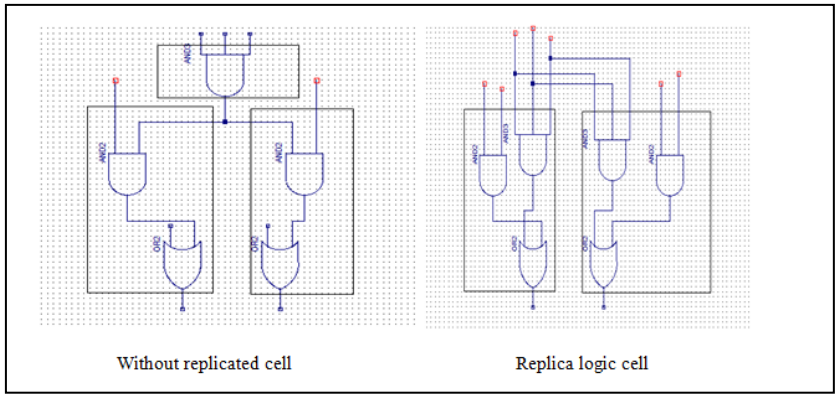

Fig. 4. FPGA based replica cell design

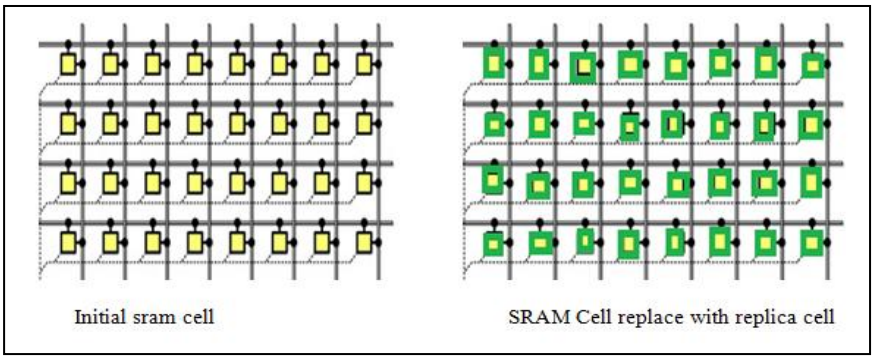

Fig. 5. Replica Cell hardware implementation with SRAM cell

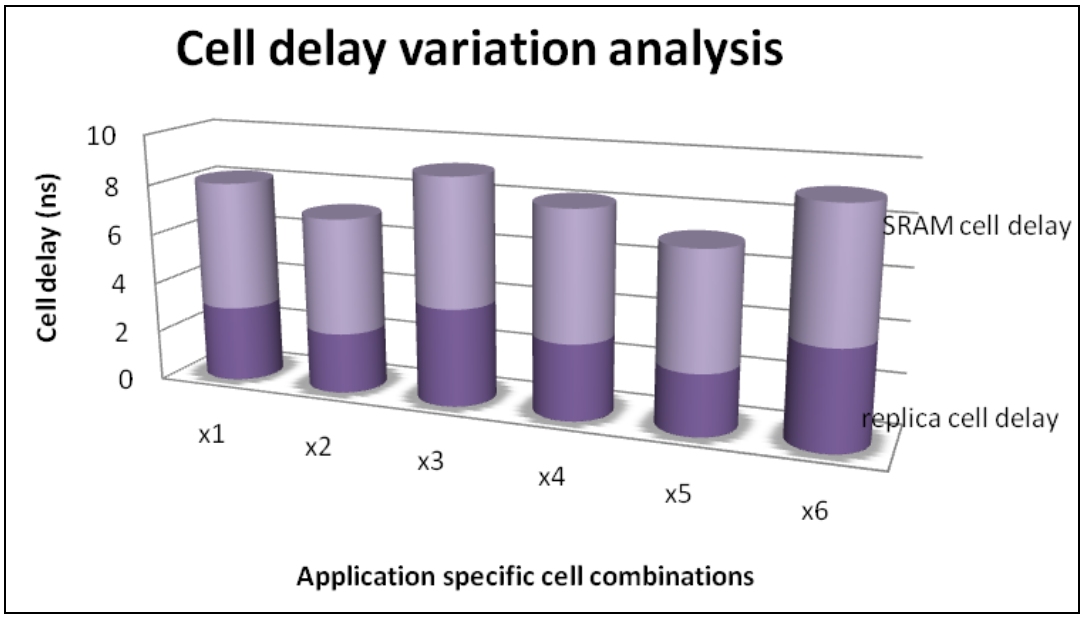

Fig. 6. Lower delay time of replica cell with comparative to SRAM cell

3. APPLICATION SPECIFIC REPLICA MEMORY OPTIMIZATION

By the help of reference replica cell memory storage area analyzed with lower and high operational probability condition. Replica based memory is used matrix blocks which in turn further divided into sub-blocks or sub-matrixes. Replica cell used Relocating collector which moves reachable block objects and eliminates previous memory block fragments. We have analyzed various application benchmarks (Tri-angularization, FIR filter, etc.) and their Directed Acyclic Graphs (DAG) and control flow graph also depicts how the program control is being passed among the memory blocks [Fig. 7 and Fig. 8]. Each application has various Operational conditions (multiplication, addition, subtraction, and dividend) and operational conditions are implemented with the help of operational class merging according to application conditional dependency or priority based modules. Each application operations have different priorities according to application data dependency so we have analyzed same class operational by the help of unification design and operational computing analyze according to reference replica cell storage mechanism. Suppose that if the equivalence operational classes of a1 and b1 have same operational class nodes, union make the same operation node be the representative of the merged equivalence; otherwise, the union makes one's or the other of the original representatives be the new representative. If s and t are both interior nodes for a binary type constructor means same op node merges their equivalence op classes on speculation [Fig. 9 and Fig. 10]. By merging it may be decrease the many equivalence classes. In application sections have p and q has same equivalence classes that merge the operation and have same probability that reduces the memory space area and reduces the computation time [Fig. 10 and Fig. 11]. In unification mechanism match two address units based expressions Source address (node1) and Destination address (node2) can be identified by substituting expressions for the variables in S and D [Fig. 10]. In ASIP system Lower the CPI (cycle per instruction) have more advantage when scheduling approaches or effective storage mechanism apply then we get effective system performance [Fig. 11].

```
//
//  Set up the edge array.
//
  edge = new int[m*n];

  for ( j = 0; j < triangle_num; j++ )
  {
    edge[0+(j                )*m] = triangle_node[0+j*3];
    edge[1+(j                )*m] = triangle_node[1+j*3];
    edge[0+(j+  triangle_num)*m] = triangle_node[1+j*3];
    edge[1+(j+  triangle_num)*m] = triangle_node[2+j*3];
    edge[0+(j+2*triangle_num)*m] = triangle_node[2+j*3];
    edge[1+(j+2*triangle_num)*m] = triangle_node[0+j*3];
  }
//
//  In each column, force the smaller entry to appear first.
//
  for ( j = 0; j < n; j++ )
  {
    e1 = i4_min ( edge[0+j*m], edge[1+j*m] );
    e2 = i4_max ( edge[0+j*m], edge[1+j*m] );
    edge[0+j*m] = e1;
    edge[1+j*m] = e2;
  }
//
//  Ascending sort the column array.
//
  i4col_sort_a ( m, n, edge );
//
//  Records which appear twice are internal edges and can be ignored.
//
  node_boundary = new bool[node_num];
```

Fig. 7. Tri-angular application section

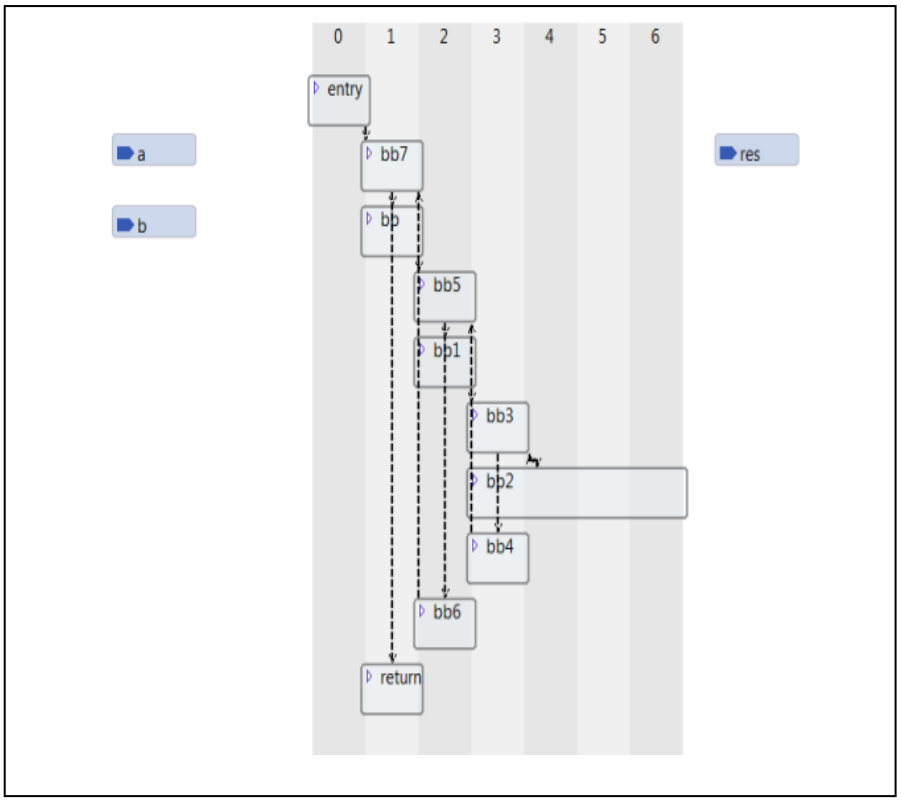

Fig. 8. CFG design of tri-angularization application

Memory Reference design (Replica design) is implemented with reusability design. After all miss collectors have already analyzed every memory block reference within the reachable sub-block objects and updating them to point to the new locations. The relocation based replica design improves a program temporal and spatial locality which collecting memory misses data and updating them for new data arrives and also provide space for data storage. We have associate reference replica cell with SRAM or replacing them with replica combination. We have analyzed shortest path based node design by the help of replica design. We have analyzed the shortest path degree according to reducing memory data transition time or cell delay which is required for ASIP and HLS system and shortest path span requires for implementing the power or time consumption for fast computing [Fig. 14]. By the help of Xilinx [14], we have optimized the Application specific simulation and integrating the replica cell design. We have analyzed various benchmarks such as filter, neural network etc. By the help of effective replicated cell design combination integration we gets lower cell delay time which is highly required for low power consumption embedded systems and HLS system [Fig. 15].

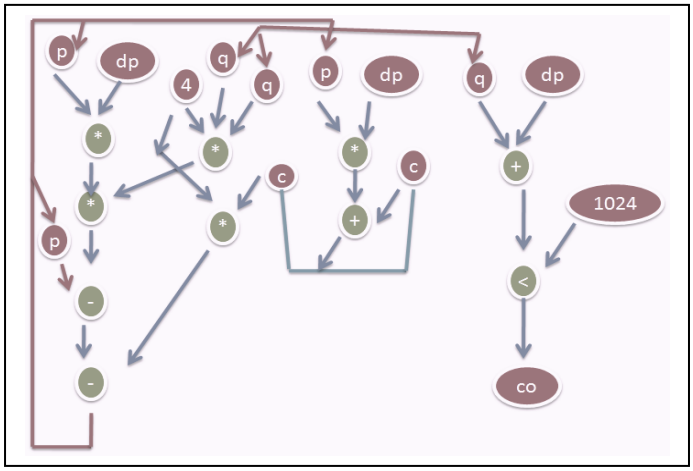

Fig. 9. Operational analysis of 1KB size of tri-angularization matrix based application section

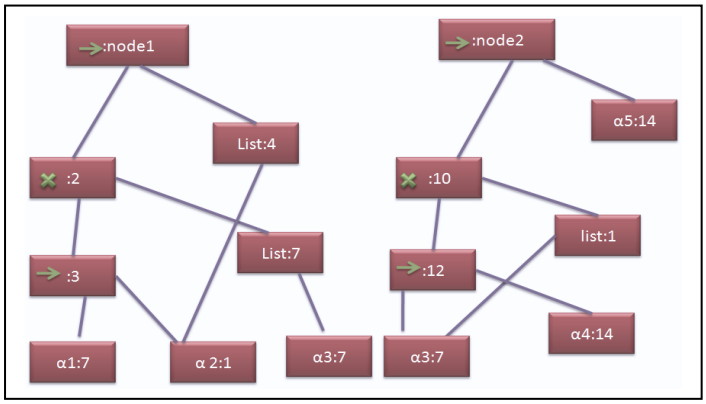

Fig. 10. Mapping based Application specific operational behavior analysis

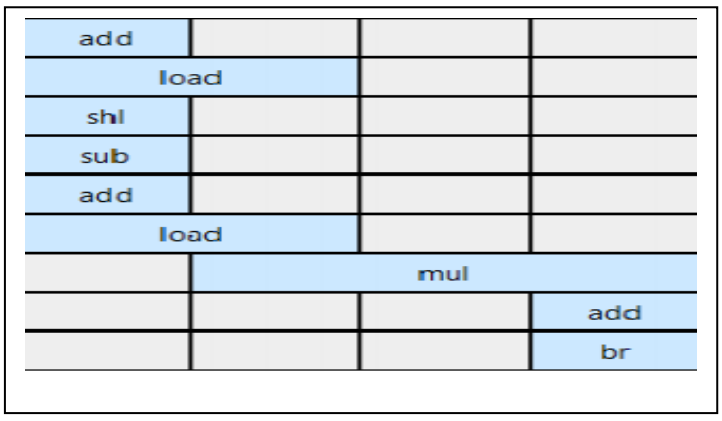

Fig. 11. Application specific operational analysis in memory area

4. INTERPOLATION DEGREE BASED DATA STORAGE OPTIMIZATION AND APPLICATION SPECIFIC SIMULATION

The interpolation based memory architecture is implemented with the programming of NxN array size of replica memory system [Fig. 12]. Memory data is mapped to a specific position with the help of data ordering or data transfer schemes and memory data is placed in a fixed location by the help of column circuitry implemented replica cell design mechanism [Fig. 13].Memory blocks array is arranged with x and y coordinates. In interpolation based schemes h provides a space that used for x position as addressing and yp used as a column circuitry design. In these scheme data properly shifted according to first and second differences degree order [Fig. 12].

```
array size n block
ax= is an array containing values of x(x0,x1,.....xn)
ay= is an array containing values of y(y0,y1,.....yn)
diff is difference table
h=space beetween the values of x
implementaed through intrepolation mechanism
h=ax[i]-ax[0]
we have calculating first order differences
for(i=0;i<=n-1;i++)
diff[i][1]=ay[i+1]-ay[i];
we can calculated high order diffrences
for(j=2;j<=order;j++)
for(i=0;i<n-1;i++)
diff[i][j]=diff[i+1][j-1]-diff[i][j-1];
/*  we have find x0
i=0;
while(!(ax[i]>x))i++;
/*now  ax[i] is x0 & ay[i]is y0
i-;
p=x-ax[i]/h;
yp=ay[i];
/* interpolated  design
for(k=1;k<=order;k++)
{
nr*=p-k+1;
dr*=k;
yp+=(nr/dr)*diff[i][k];
}
```

Fig. 12. Interpolation implemented Application specific node based shortest path analysis

We have analyzed shortest path based interpolation implemented optimal degree by the help of replica design [Fig. 5] and there are three paths analyzed for source (node1) to destination (node6) first section path are node 1 to node 6, second section path are node1 to node 4, node 5 to node 4 and node 4 to node 6 and third section path are node1 to node 5, node 5 to node 3 node 5 to node 3, node 3 to node 4, node 4 to node 6. Nodes 1 to node 6 have lower ordered degree path time span because of it's

made up of replica cell combinations and interpolations based design [Fig. 14]. Our main target in this paper is to the reducing memory data transition time or cell delay which is required for ASIP and HLS system so we have targeting degree based shortest path for source to destination path with least waste of time efforts for fast computing. By the help of Xilinx [14], we have optimized the Application specific simulation and integrating the replica cell design then we get final simulation result in the form of cycle time, latency, pipeline and iterations, etc [Fig. 15]. This effective replicated cell combination reduces the delay time or cell cost due to it's made up of high bus bandwidth combination. This cell combination is implemented with SRAM cell so delay time play dominant role for recently develop low power embedded devices and HLS system [Fig. 15].

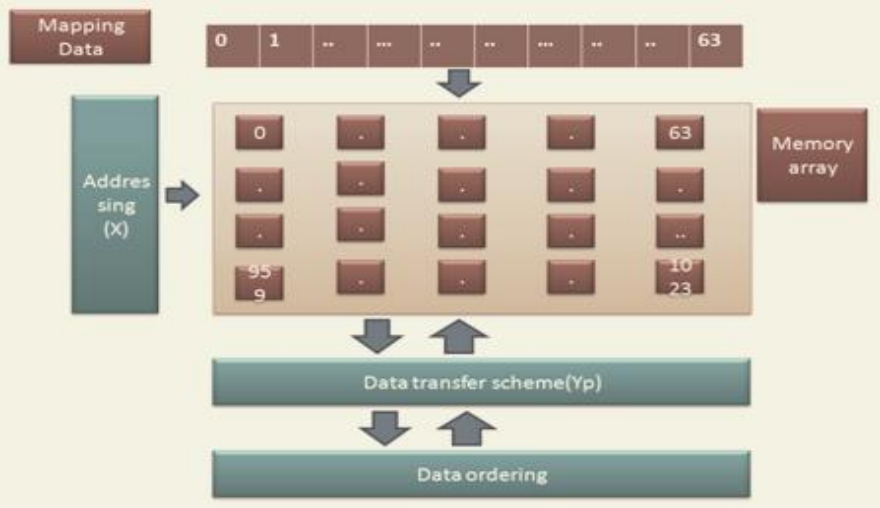

Fig. 13. Memory storage architecture

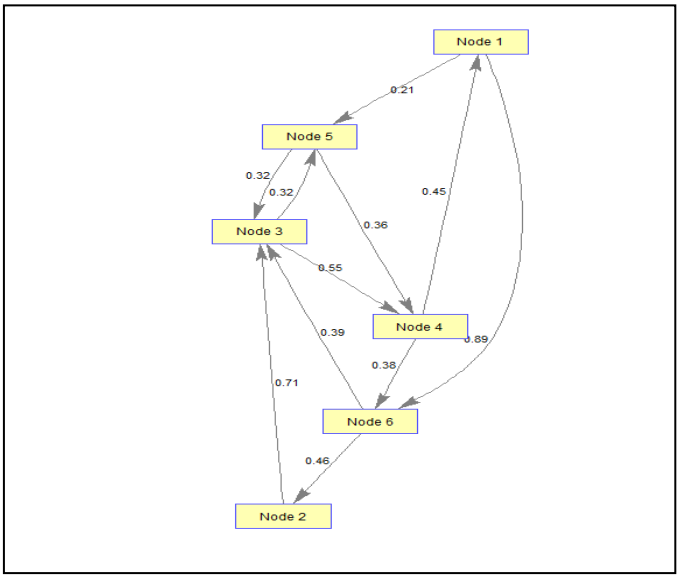

Fig. 14. Shortest degree path analysis according to replica cell combinations

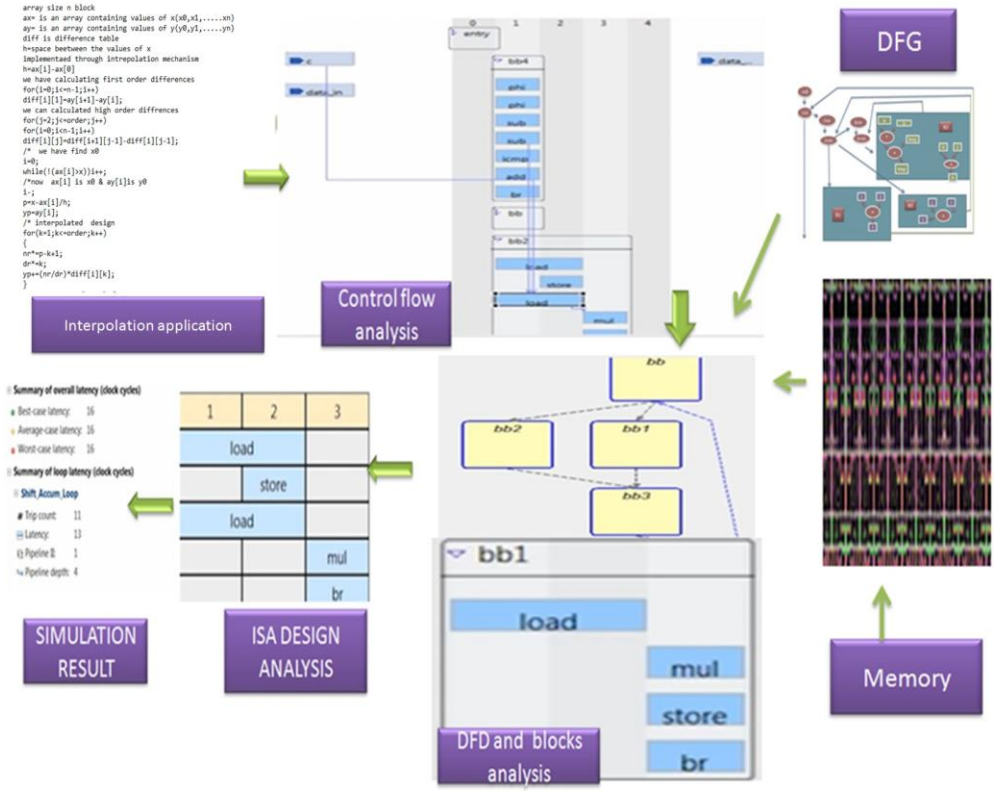

Fig. 15. Application specific simulation and memory Integration for HLS

5. CONCLUSION

High level synthesis and ASIP design technology required high speed and low power consumption memory design which play dominant role in performance improvement so an effective reference replica based memory architecture have been implemented to reduces the cost and delay. By the help of reference replica cell storage area and delay implemented for various benchmarks such as neural network or images optimization. The reconstructive replica cell architecture improves with column circuitry design and point to point data storage mechanism by the help of effective bus bandwidth implemented mechanism. The Replica cell design architecture is implemented in the area of sense amplification section of SRAM cell and its combination implements memory storage and system performance which is utilized for ASIP and HLS area.

REFERENCES

[1] A. Punitha and M. Joseph, "A temperature aware scheduling with incremental binding and floor planning for HLS". Middle East J. Scientific Research. Vol. 22, pp. 822-828, 2014.

[2] O. Esko, P. Jaaskelainen, P. Huerta, C. S.D. L. Lama, J. Takala and J.I. Martinez, "Customized exposed Data path softcore design flow with compiler support". Int. proc. of ICFPLA. (2010) pp. 217-222,2010.

[3] M. Kudlur, K. Fan, M. Chu, R. Ravindran, N.C lark and S. Mahlke, "FLASH: Foresighted latency aware scheduling heuristic for processor with customized data path".*CG*- 2004, pp. 201-212,2004.

[4] B. Gorjiara and D. Gajski, "Custom processor design using NISC: A case study on DCT algorithm", *Int. Workshop on embedded system for real time multi-media*. pp. 55-60,2015.

[5] B.H. Calhoun and A.P. Chandrakasan, "A 256-kb 65-nm sub-threshold SRAM design for ultra-low-voltage operation", *IEEE J. of solid-state circuits*, pp.680-688, vol. 42,2007.

[6] K. Dhanumjaya, M. Sudha, M.N. Giri Prasad and K. Padmaraju, "Cell stability analysis of conventional 6t dynamic 8t SRAM cell in 45 nm technology", Int. J. of VLSI design & Communication Systems. pp.41-51,vol.3, 2012.

[7] G.K. Chen, D. Sylvester, D. Blauw and T. Mudge, "Yield-Driven Near-Threshold SRAM Design", IEEE Trans. Very Large Scale Integration (VLSI) Systems, pp.1590-1598, vol.18,2010.

[8] T.T.H. Kim and Z. H. Kan, "Impact Analysis of NBTI/PBTI on SRAM Vmin and Design Techniques for Improved SRAM Vmin", J. of Semiconductor Technology and Science, pp.87-97,vol.13,2013.

[9] N. K. Shukla, R. K. Singh and M. Pattanaik, "Design and Analysis of a Novel Low-Power SRAM Bit-Cell Structure at Deep-Sub-Micron CMOS Technology for Mobile Multimedia Applications", (IJACSA) Int. J. of Advanced Computer Science and Applications, pp.43-49,vol.2,2011.

[10] S. Mukhopadhyay, H. Mahmoodi and K. Roy, "Modeling of Failure Probability and Statistical Design of SRAM Array for Yield Enhancement in Nanoscaled CMOS", IEEE trans. Computer-aided design of integrated circuits and systems, pp.1859-1880,vol.24,2005.

[11] A. K. Dadoria, A.S.Yadav and C.M. Roy. "Comparative Analysis of Variable N-T SRAM Cells". Int. J. of Advanced Research in Computer Science and Software Engineering, pp.612-619, vol.3, 2013.

[12] S. Vijayan and P. Kodavanti, "Novel Design and Implementation Of 8 X 8 SRAM Cell Array for Low Power Applications", Int. J. of Engineering Research & Technology (IJERT). pp. 1355-1359, vol.3, 2014.

[13] H. Yamauchi, "A Scaling Trend of Variation-Tolerant SRAM Circuit Design in Deeper Nanometer Era", J. of Semiconductor Technology and Science, pp. 37-50, vol.9, 2009.

[14] Tool available from http//www.xilinx.com.