IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 5, Issue 4, Ver. II (Jul - Aug. 2015), PP 01-13
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Parallel Test Scheduling of 3D Stacked SoCs with Temperature and Time Constraints

Indira Rawat[1], M.K. Gupta[2], Virendra Singh[3]
[1](*Department of Electrical Engineering, Govt. Engineering College, Ajmer, Rajasthan , India)*
[2](*Deptt. Of Electronics and Communication Engg.,Maulana Azad National Institute of Technology,Bhopal, India)*
[3](*Deptt. Of Electrical Engineering,Indian Institute of Technology,Mumbai, India)*

Abstract : *Today's VLSI circuits are very compact and complex designs. As the advancements made are very fast with cut throat competitions from various manufacturers, they are likely to have more defects and faults. This requires a proper testing process to be adopted for all products. Testing is a process which has to be done on all pieces of products. At the same time it requires a low cost, highly efficient method to be adopted. Fault coverage should also be maximized for ensuring fast and efficient work. The technology is also undergoing fast transitions. System on Chip is a design paradigm which involves integration of entire system onto a single chip. It can be a RAM, DRAM, CPU, UDL, analog , digital, A/D or D/A converters needed for any particular requirement. In this paper we have worked on test scheduling of 3D SoCs with thermal and time constraints. The circuits used have been built using benchmark SoC circuits. They have been piled on top of each other to build two, three and four stack circuits. The method has been compared with the sequential method of testing. The method proposed in this work shows good thermal response and elimination of hotspots.*

Keywords*: 3D SoCs, sequential testing, Thermal awareness, RHDF, HHDF, VHDF.*

I. Introduction

Testing is a very important aspect to be successfully accomplished before the marketing of any product. To shorten the testing time, concurrent testing of many cores of the chip is considered. The concurrent testing is not a simple process as it gives rise to a number of complications. The steep rise in temperature is observed which can prove fatal for the chips. Very fast switching activity is observed during switching which can increase the level of temperatures to an extent leading to localized increase of temperature at spots called hotspots. The chips gets permanently damaged if hotspots are formed. Till recently, power level reduction during testing used to be the accepted approach during testing [1, 2, 3] but this approach proved to be insufficient because other factors like package, cooling method and layout must be considered. Many good works have been reported in this area. Test scheduling is also required to be done in a manner so that testing time is minimized considering many constraints. Power constrained test scheduling [4] by W-D Tseng in 2006 is one approach. He has presented a method to integrate the management of power consumption to augment the parallelism of the testing activities to reduce testing time. Wu, et.al [5] in 2008 presented an optimization technique for minimizing the test time for core based 3D SoCs under constraints on the number of TSVs and the test access mechanisms(TAM) bit width. 3D SoCs is an attractive technology due to its potential benefits. This involves the vertical stacking of different ICs leading to a 3D structure. In 2009, Jiang et al [6] gave 3D test access mechanisms by taking pre-bond test times into account to optimize total test times. In 2010, Marinissen [7] highlighted the challenges with respect to design to test infrastructure required for wafer level and package test required for 3D SICs. Since 3D technology has thermal issues of much concern , cooling methods have also been a topic of research. Cooling by various methods has also been proposed by many researchers.

II. Brief Background

Test scheduling has been addressed for over 30 years and was developed with different constraints. Test resource conflicts were the original concern when it was first developed [8, 9, 10]. Test resource conflicts are mainly caused by sharing of test resources like test pattern generator, response compressor and the paths. The deeply embedded cores in an SoC or multi core system may be tested using external tester (ATE) or built-in-self-test. The test scheduling problem with resource conflicts is NP-complete. Some have adopted the graph theory problems to solve the resource problems. Power constrained test scheduling has been addressed by many [11, 12, 15] as the power densities are also increasing substantially. The test buses or the test access mechanisms (TAMs) may be shared for cores. Various TAM optimization techniques have been reported [13, 14]. Power consumption in test mode are more important to consider as it is increasing rapidly in today's chips.

Power constrained or power aware type test scheduling takes into account both power consumption of the chip as well as resource conflicts. A set of tests cannot be scheduled together if the sum of their power consumption exceeds the power consumption limit of the chip. These problems have been solved by graph based algorithms. Chou et al [11,12] constructed test compatibility graph with power information. Rectangular packing approach is popular approach which has been adopted by for solving power constrained problem [14]. Here Iyengar et.al discussed precedence based power constrained test scheduling and formulated it into a mixed –integer linear programming (MILP) .Test scheduling using simulated annealing , genetic and ant colony optimization have also been developed. Test access mechanism optimization (TAM) [13,14] have also been used.

The consequence of high power densities is the increase in temperature which adversely affects the device reliability and performance. An increase in 10-15˚C in temperature decreases the life of the device by a factor of two. There is a corresponding increase in gate delays and deterioration of the circuit performance. Leakage power also increases with temperature. Timing errors are more likely to occur in overheated system . Rosinger et al in [15] addressed this problem and proposed a method for generating thermal safe test schedules. They also proposed a thermo-resistive model for computation of thermal profile of the chip. Liu et. Al [16] proposed a method to spread heat evenly in the chip and reduce hotspots. Z. He [17] proposed a thermal aware test scheduling scheme by way of test set partitioning and interleaving and employed a constraint logic programming (CLP) to generate thermal aware test schedules. The thermal aware test scheduling which is the problem dealt with in this paper has become a very challenging job to be accomplished. Developing a simple test model for the purpose is also a requirement. The popular model used is the RC model which is a well known linear model. This is the model which has been used in the HotSpot tool [18].

III. Problem Formulation

- There are N no. of floorplans of the circuits, with each floorplan having X_i (i = 1 to N) cores in respective floorplan.
- Given are the parameters like placement of cores in stack, area of cores and test length of all cores.
- It is required to find a test schedule such that the temperature of the chip should not rise above thermal limit and there is no hotspot formation during testing.
- Time required for testing should be minimized.
- Test these cores schedule wise on Hotspot and record the temperature of cores after testing.
- Compare the temperature and time of testing with the sequential test scheme to assess the superiority of the scheduling scheme.

Select standard benchmark circuits [19] for working upon in the problem formulation. Temperatures of all cores after completion of each schedule and after complete testing are to be recorded. Temperature profile of all cores to examine impact on whole chip in the form of svg images is to be recorded.

In this paper, we investigate how to schedule the tests for a 3D stacked SoC built using the benchmark circuits. Two, three and four stacked structures of benchmark circuits have been built. The circuits considered are d695 with 10 cores, d281 with 8 cores, f2126 with 4 cores and 2f2126 with 8 cores. The simulation model consists of the parameters as shown in Table. 1.

Table. 1 Simulation Model Parameters

Chip thickness	0.00015 m
Heat spreader thickness	0.001 m
Heat spreader size	0.03 m
Heat sink thickness	0.0069 m
Heat sink size	0.06 m
Ambient temperature	300 K
Silicon thermal conductivity	100.0 W/m.K
Silicon specific heat	1.76 e6 J/m^3-K
Temperature threshold	355K
Thermal interface material (TIM) thickness	2.0 e^-05
TIM thermal conductivity	4.0W/m.K
Hotspot calling interval	10 K cycles at 3 GHz

The size of the chip is 4mm x 4mm. The size of all the dies are taken to be the same. Since we are concerned with the heat spread and consequent temperature rise, we estimate the vertical and horizontal adjacencies.

IV. The Methodology

In this work we consider the heat dissipation of cores in all directions i.e. in planar as well as in vertical direction. When the core is tested, it gets heated up and transmits the heat generated through conduction in the adjoining cores thereby increasing the temperature of the cores which are not even tested. We are working on the problem with the specific requirement that the cores under testing should not be adjacent i.e. we are adopting the adjacency exclusion scheme while selecting cores for testing such that the cores get enough surrounding cores to dispense with their heat generated. Since the core with maximum test length is expected to get heated more, therefore we target to test these cores first so that during the whole test duration these cores get time to cool down thereby keeping the total test time minimal. We know that the capability of heat dissipation of each core in the stack varies based on its positioning in the SoC. A core which is embedded in the centre is likely to get cooled at slower rate than the core which is positioned at the corner or closer to any edge of the layer. This basic principle of relative heat dissipation capability in a planar direction gives an idea of calculating the Horizontal Heat Dissipation Function (HHDF) of the cores. This parameter depends on the position of the core in the layer and is determined on the basis of distance of the core from the edges of the layer in all four directions viz. -x, +x direction in the x axis and -y, +y direction in the y axis. It is well known that the heat dissipation takes place in exponential pattern and the resistance of the substrate plays a key role in the time constant of the heat dissipation. The resistance of the substrate is in turn directly proportional to the length in the heat flow path. Therefore, all the four distances of core from edges in the plane will add up to contribute in determining core's heat dissipation capability, more the added up values, more time the core is going to take in cooling down in horizontal direction as compared to the core in the same plane which is closer to the edge. This scheme is better illustrated in Figure. 1 where a standard benchmark circuit d 281 is shown with a reference of core 5 of its circuit with its distance from all edges.

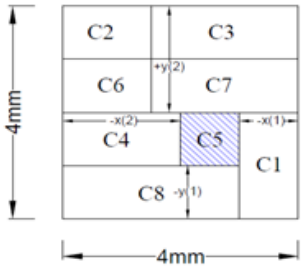

Figure 1. Core 5 in D 281 circuit

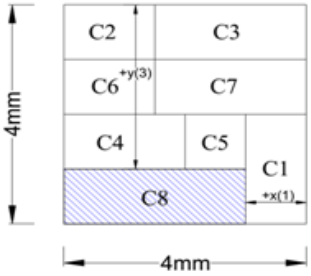

Figure 2. Core 8 in D 281 circuit

It can be easily visualized that the relative heat dissipation capability of core 5 is less than that of core 8 (Figure. 2) as there are 4 components of distances adding up in core 5 whereas there are only 2 components of distances affecting heat dissipation in core 8. In addition to HHDF, one more important component which effects heat dissipation capability of the core is its vertical placement in the stack. 2 cores in the same position in their respective planes will have different heat dissipation capability depending on their relative distances from the heat sink.

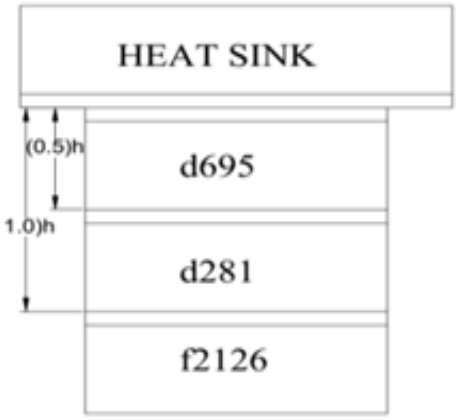

Figure 3 . Vertical stacking of 3 layers in a SoC

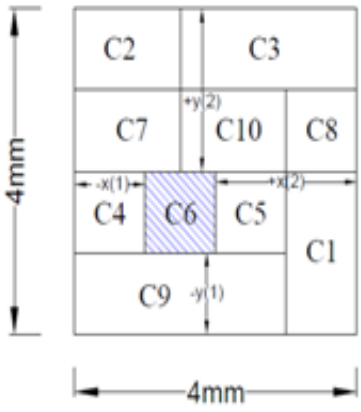

Figure 4. Core 6 in D 695 circuit

This heat dissipation capability of the whole plane is termed as Vertical Heat Dissipation Function (VHDF) and is calculated in the same way as that of HHDF, i.e. based on its positioning in the stack and its

distance from the heat sink. It is explicit from Figure 3 of 3 layer stacked circuit comprising of 3 benchmark circuits viz. d 281, d695 and f2126 that the VHDF of the layer d281 is more than the VHDF of layer d695 because of the proximity of later to Heat Sink. The VHDF of any layer is calculated based on its distance from Heat Sink. The distance of d281 is taken as 1 whereas that of d695 is taken as 0.5. Accordingly, any core in these two layers, with similar positioning in their respective layer and thereby having same HHDF, will have different VHDF based on their positioning in the stack. The Relative Heat Dissipation Factor (RHDF) of any core is combination of both VHDF and HHDF and collectively they determine the relative heat dissipation capability of all cores. In Figure 4, core 6 is positioned in the same place in d695 circuit as that of core 5 in d281, and will therefore have same HHDF value but owing to proximity of d695 to the Heat Sink, the combination of HHDF and VHDF of core 5 of d281 will be more than that of core 6 of d695.

In addition to the positioning of cores in the Stack, one more factor which plays a key role in the heat dissipation capacity of any core is the Test Length of the core, i.e. clock cycles applied to cores during their testing. The test length of any core determines the heat generated in the core on account of its testing; more the test time, more the heat generated and therefore more time to cool down. based on the test length, we calculate the P_Trace value of the cores. P_Trace value of any core is the averaging of 400:1 test cycles at 1.2 GHz test frequency. In our case, the test frequency applied is 3 GHz and thus P_trace is calculated as averaging of 1000:1 test cycles. This P_Trace value is also combined with the combined value of HHDF and VHDF to get RHDF of the cores. Table 2 shows the various parameters values of d 695 and d 281 circuits where, HHDF, VHDF, Test length, P_Trace and RHDF of all cores are displayed.

Table 2. Core details of 2 layer Stacked Circuit

Layer	Core No.	HHDF	VHDF	HHDF*VHDF	Test length	P_Trace	RHDF
d695	C2,1	1.3231	0.1353	0.1790	12	0.012	0.0021
	C2,2	1.3868	0.1353	0.1876	73	0.073	0.0137
	C2,3	1.2299	0.1353	0.1664	2507	2.507	0.4172
	C2,4	1.6909	0.1353	0.2288	5829	5.829	1.3336
	C2,5	1.9488	0.1353	0.2637	5105	5.105	1.3460
	C2,6	1.9488	0.1353	0.2637	9869	9.869	2.6022
	C2,7	1.6447	0.1353	0.2225	3359	3.359	0.7475
	C2,8	1.6909	0.1353	0.2288	4605	4.605	1.0535
	C2,9	1.0844	0.1353	0.1467	714	0.714	0.1048
	C2,10	1.4878	0.1353	0.2013	3863	3.863	0.7776
d281	C0,1	1.323	0.3679	0.4867	282	0.282	0.1373
	C0,2	1.3868	0.3679	0.5102	2204	2.204	1.1245
	C0,3	1.2299	0.3679	0.4525	2144	2.144	0.9701
	C0,4	1.5809	0.3679	0.5816	1734	1.734	1.0085
	C0,5	1.9488	0.3679	0.7170	2624	2.624	1.8813
	C0,6	1.6447	0.3679	0.6051	1010	1.01	0.6111
	C0,7	1.4878	0.3679	0.5474	4028	4.028	2.2048
	C0,8	1.0844	0.3679	0.3990	3248	3.248	1.2958

From the above Table, we will now prepare a sorted List of cores in descending order of their RHDF. The sorted 2 layer Stack list is shown in Table 3 where the core with highest RHDF is at the top. This indicates that this core will take maximum time to cool.

Table 3. Sorted 2 Layer Stacked Structure

Layer	Core No.	HHDF	VHDF	HHDF*VHDF	Test length	P_Trace	RHDF
d695	C2,6	1.9488	0.1353	0.2637	9869	9.869	2.6022
d281	C0,7	1.4878	0.3679	0.5474	4028	4.028	2.2048
d281	C0,5	1.9488	0.3679	0.7170	2624	2.624	1.8813
d695	C2,5	1.9488	0.1353	0.2637	5105	5.105	1.3460
d695	C2,4	1.6909	0.1353	0.2288	5829	5.829	1.3336
d281	C0,8	1.0844	0.3679	0.3990	3248	3.248	1.2958
d281	C0,2	1.3868	0.3679	0.5102	2204	2.204	1.1245
d695	C2,8	1.6909	0.1353	0.2288	4605	4.605	1.0535
d281	C0,4	1.5809	0.3679	0.5816	1734	1.734	1.0085
d281	C0,3	1.2299	0.3679	0.4525	2144	2.144	0.9701
d695	C2,10	1.4878	0.1353	0.2013	3863	3.863	0.7776
d695	C2,7	1.6447	0.1353	0.2225	3359	3.359	0.7475
d281	C0,6	1.6447	0.3679	0.6051	1010	1.01	0.6111
d695	C2,3	1.2299	0.1353	0.1664	2507	2.507	0.4172
d281	C0,1	1.323	0.3679	0.4867	282	0.282	0.1373
d695	C2,9	1.0844	0.1353	0.1467	714	0.714	0.1048
d695	C2,2	1.3868	0.1353	0.1876	73	0.073	0.0137
d695	C2,1	1.3231	0.1353	0.1790	12	0.012	0.0021

Our requirement is to minimize temperature rise of cores during testing of multiple cores in a particular test schedule in such a way that the cores selected are not adjacent to each other, horizontally and vertically, and secondly maximum numbers of cores get selected to minimize the number of test Schedules. Adjacent cores are excluded so that cores get enough space in vicinity to dispense with their generated heat and no Hotspot gets formed on account of localized high temperature in any core as it would damage the chip. Therefore while selecting cores from this sorted list, Adjacency Exclusion Principle needs to be applied as per which, the cores which are selected in any schedule are not adjacent to each other in planar as well in vertical position. In this way, the cores selected in any schedule are spread out uniformly on the stack and therefore does not cause abrupt and excessive temperature rise at any spot.

V. Algorithm

The above requirement is implemented using an algorithm. The proposed algorithm is presented here in a simplified flow chart which is shown in Figure. 7. The proposed algorithm implements this scheme.

The requirements of the algorithm are as follows:

- All cores of given SoC are to be tested but the temperature needs to be checked.
- Test Scheduling has to be developed for testing of all cores.
- Test Schedule should have cores which are widely spread on the SoC.
- No. of test schedules should be minimal to keep test time low.
- Proposed scheme should be better than the conventional method of sequential testing viz. one floorplan at a time for testing in respect of temperature rise and the test time.

In the algorithm, we use the benchmark circuits for the testing. The benchmark circuits are taken from [19]. The circuits which we use are d695, d281, f2126 and 2f2126. The core layout in these benchmark circuits is shown in Figure. 1 (d281), Figure. 4 (d695), Figure. 5 (f2126) and Figure. 6 (2f2126). The dimension of all layers are 4mm x 4mm as per the requirements of Hotspot, the tool on which the Test Schedules so generated are tested.

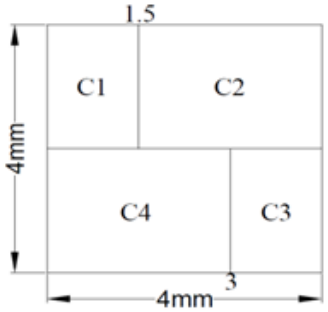

Figure. 5 f2126 circuit.

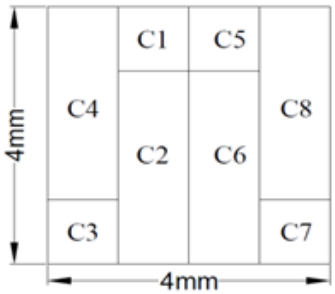

Figure. 6 2f2126 circuit.

As stated above, all these benchmark circuits have the same dimensions. The thickness of each layer is 0.00015 m. In between two layers, exist layer of Thermal Interface Material (TIM) which is widely spread along the surface area of layer i.e. it also has the same cross sectional area as of layer i.e. 4mm x4mm. The thickness of TIM is 0.00002 m. The Heat Sink is placed above the whole stacked setup with a Heat Spreader (HS) interfaced between TIM and Heat Sink. The dimensions of Heat Sink are 0.06m x0.06 m and the thickness of Heat Sink is 0.0069m. The cores in the layers are different and they have different test length.

The implementation of this algorithm has been done on 2, 3 and 4 layered stack as shown in Figure 8, 3 and 9 respectively. In 2 layered stack we have considered circuits d695 and d281 where d 695 is closest to Heat Sink. In 3 layered stack, we have considered d695 (closest to Heat Sink), d281 and f2126 (farthest from Heat Sink). In 4 layered stack, we have considered d695 (closest to Heat Sink), d281, f2126 and 2f2126 (farthest from Heat Sink). In these diagrams between two layers exist Thermal Insulating Material (TIM) due to which when we show the results, the circuits are represented by alternate numbers viz. in 2 stacked structures, the d281, farthest from the Heat Sink gets numbered layer 0 while d695 gets numbered 2 while testing on HotSpot. Therefore the results will also be depicted accordingly.

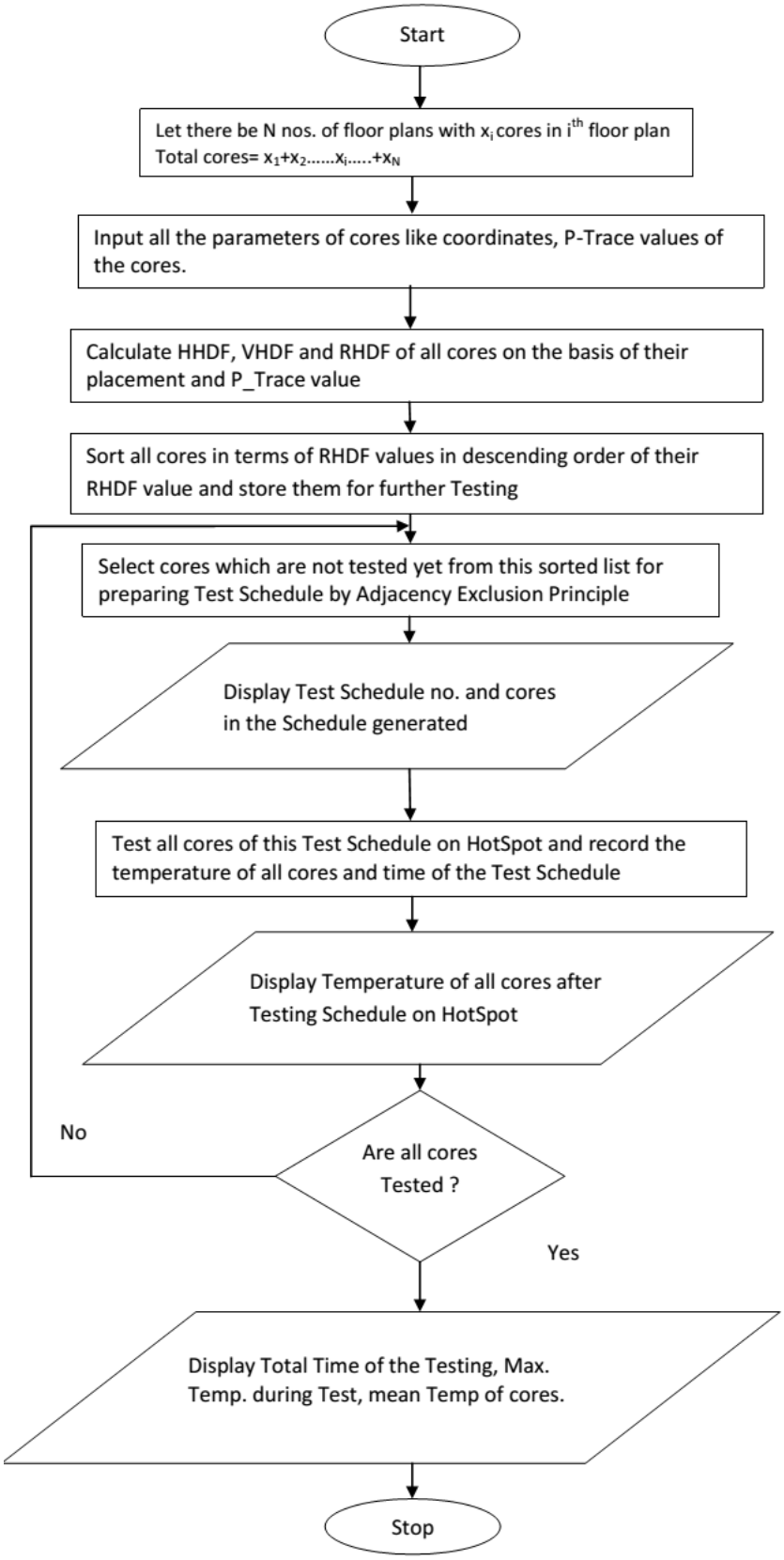

Figure. 7 Algorithm Flowchart

In 3 layered stack, the numbering is - f2126 (farthest from Heat Sink) gets numbered layer 0, d281 gets numbered 2 and d695 gets numbered 4. Similarly, in 4 layered stack, 2f2126, farthest from Heat Sink gets numbered layer 0, f2126 gets 2, d281 gets 4 and d695, closest to Heat Sink is numbered 6. The results after testing will appear with this protocol only.

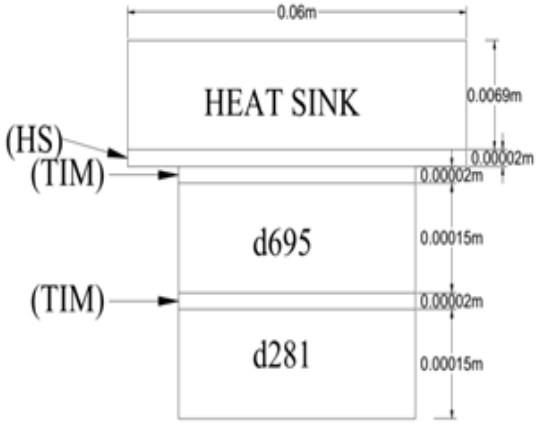

Figure 8 . Vertical stacking of 2 layers in a SoC

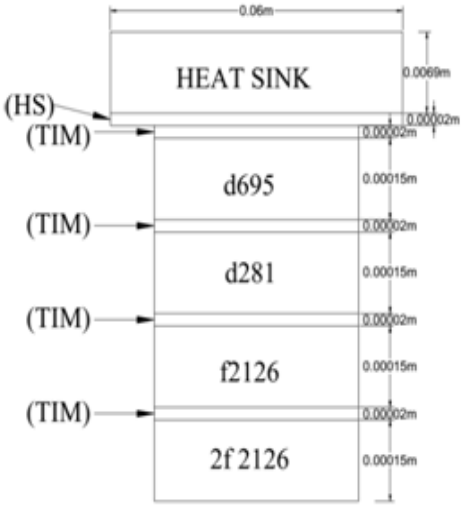

Figure 9 . Vertical stacking of 4 layers in a SoC

VI. Results

This algorithm when implemented on 2, 3 and 4 layered benchmark circuits as discussed above, give sorted arrays based on RHDF values of cores for designing the test Schedules and then the Test Schedules are generated based on Adjacency Exclusion Principle of Algorithm. The sorted list for 2 stacked structures is already shown in Table 2 and for 3 stacked structures in Table 4 below.

Table 4. Sorted 3 Layer Stacked Structure

Layer	Core No.	HHDF	VHDF	HHDF*VHDF	Test Length	P_Trace	RHDF
f2126	C0,2	1.1199	0.5134	0.5750	7190	7.19	4.1339
d695	C4,6	1.9488	0.1353	0.2637	9869	9.869	2.6022
d281	C2,7	1.4878	0.3679	0.5474	4028	4.028	2.2048
f2126	C0,1	1.2768	0.5134	0.6555	3014	3.014	1.9757
d281	C2,5	1.9488	0.3679	0.7170	2624	2.624	1.8813
d695	C4,5	1.9488	0.1353	0.2637	5105	5.105	1.3460
d695	C4,4	1.6909	0.1353	0.2288	5829	5.829	1.3336
d281	C2,8	1.0844	0.3679	0.3990	3248	3.248	1.2958
d281	C2,2	1.3868	0.3679	0.5102	2204	2.204	1.1245
d695	C4,8	1.6909	0.1353	0.2288	4605	4.605	1.0535
d281	C2,4	1.5809	0.3679	0.5816	1734	1.734	1.0085
d281	C2,3	1.2299	0.3679	0.4525	2144	2.144	0.9701
d695	C4,10	1.4878	0.1353	0.2013	3863	3.863	0.7776
d695	C4,7	1.6447	0.1353	0.2225	3359	3.359	0.7475
f2126	C0,3	1.323	0.5134	0.6792	905	0.905	0.6147
d281	C2,6	1.6447	0.3679	0.6051	1010	1.01	0.6111
f2126	C0,4	0.9744	0.5134	0.5003	905	0.905	0.4527
d695	C4,3	1.2299	0.1353	0.1664	2507	2.507	0.4172
d281	C2,1	1.323	0.3679	0.4867	282	0.282	0.1373
d695	C4,9	1.0844	0.1353	0.1467	714	0.714	0.1048
d695	C4,2	1.3868	0.1353	0.1876	73	0.073	0.0137
d695	C4,1	1.3231	0.1353	0.1790	12	0.012	0.0021

This table clearly indicates that how the cores are sorted on the basis of their RHDF values. In a 3 stacked structure, core no. 2 of circuit f2126 has the highest RHDF in the whole circuit. It is due to the combined weight age of HHDF*VHDF and the P_Trace value. Core no. 1 of the same layer i.e. of f2126 has a higher value of HHDF*VHDF but due to the higer P_Trace value of the former, combined RHDF of core 2 of f2126 is greater than the core 1 of f2126. It implies that core 2 of f2126 will be heated the most and also will take maximum time to dispense with the heat generated.

Similarly, the sorted 4 layered list of cores is generated which is shown below in Table 5.

Table 5. Sorted 4 Layer Stacked Structure

Layer	Core No.	HHDF	VHDF	HHDF*VHDF	Test Length	P_Trace	RHDF
2f 2126	C0,2	1.4878	0.6065	0.9024	7190	7.19	6.4879
2f 2126	C0,6	1.4878	0.6065	0.9024	7190	7.19	6.4879
f2126	C2,2	1.1199	0.5134	0.5750	7190	7.19	4.1339
2f 2126	C0,1	1.6447	0.6065	0.9975	3014	3.014	3.0065
2f 2126	C0,5	1.6447	0.6065	0.9975	3014	3.014	3.0065
d695	C6,6	1.9488	0.1353	0.2637	9869	9.869	2.6022
d281	C4,7	1.4878	0.3679	0.5474	4028	4.028	2.2048
f2126	C2,1	1.2768	0.5134	0.6555	3014	3.014	1.9757
d281	C4,5	1.9488	0.3679	0.7170	2624	2.624	1.8813
d695	C6,5	1.9488	0.1353	0.2637	5105	5.105	1.3460
d695	C6,4	1.6909	0.1353	0.2288	5829	5.829	1.3336
d281	C4,8	1.0844	0.3679	0.3990	3248	3.248	1.2958
d281	C4,2	1.3868	0.3679	0.5102	2204	2.204	1.1245
d695	C6,8	1.6909	0.1353	0.2288	4605	4.605	1.0535
d281	C4,4	1.5809	0.3679	0.5816	1734	1.734	1.0085
d281	C4,3	1.2299	0.3679	0.4525	2144	2.144	0.9701
2f 2126	C0,3	1.433	0.6065	0.8691	905	0.905	0.7865
2f 2126	C0,7	1.433	0.6065	0.8691	905	0.905	0.7865
d695	C6,10	1.4878	0.1353	0.2013	3863	3.863	0.7776
d695	C6,7	1.6447	0.1353	0.2225	3359	3.359	0.7475
f2126	C2,3	1.323	0.5134	0.6792	905	0.905	0.6147
d281	C4,6	1.6447	0.3679	0.6051	1010	1.01	0.6111
2f 2126	C0,4	1.0844	0.6065	0.6577	905	0.905	0.5952
2f 2126	C0,8	1.0844	0.6065	0.6577	905	0.905	0.5952
f2126	C2,4	0.9744	0.5134	0.5003	905	0.905	0.4527
d695	C6,3	1.2299	0.1353	0.1664	2507	2.507	0.4172
d281	C4,1	1.323	0.3679	0.4867	282	0.282	0.1373
d695	C6,9	1.0844	0.1353	0.1467	714	0.714	0.1048
d695	C6,2	1.3868	0.1353	0.1876	73	0.073	0.0137
d695	C6,1	1.3231	0.1353	0.1790	12	0.012	0.0021

The various Test Schedules so generated are shown in Tables 6, 7 and 8 respectively for 2 Layers, 3 Layers and 4 Layers.

Table 6. Schedules for 2 Layered Stacks

Schedule	Cores selected (Layer – Core No)
Sch 1	2-3, 2-6; 0-2, 0-7, 0-8:
Sch 2	2-2, 2-4, 2-8; 0-3, 0-5, 0-6:
Sch 3	2-5, 2-7; 0-1, 0-4:
Sch 4 & Sch 5	2-9, 2-10; 2-1:

Table 7. Schedules for 3 Layered Stacks

Schedule	Cores selected (Layer – Core No)
Sch 1	4-6, 4-8; 2-2, 2-5, 0-2:
Sch 2	4-3, 4-4, 4-5; 2-7, 2-8; 0-1, 0-3:
Sch 3	4-2, 4-9, 4-10; 2-1, 2-3, 2-4:
Sch 4 & Sch 5	4-1, 4-7; 2-6; 0-4:

Table 8. Schedules for 4 Layered Stacks

Schedule	Cores selected (Layer – Core No)
Sch 1	6-6,6-8; 4-2, 4-5; 2-2; 0-2, 0-7:
Sch 2	6-3, 6-4, 6-5; 4-7, 4-8; 2-3; 0-1, 0-3, 0-6:
Sch 3	6-2, 6-9, 6-10; 4-1, 4-3, 4-4; 2-1; 0-5:
Sch 4	6-1, 6-7; 0-4, 0-8:
Sch5	4-6; 2-4:

Figures 10, 11 and 12 show the cores selected in various Schedules and depict the position of cores so selected in each Schedule which are widely spread in the structure.

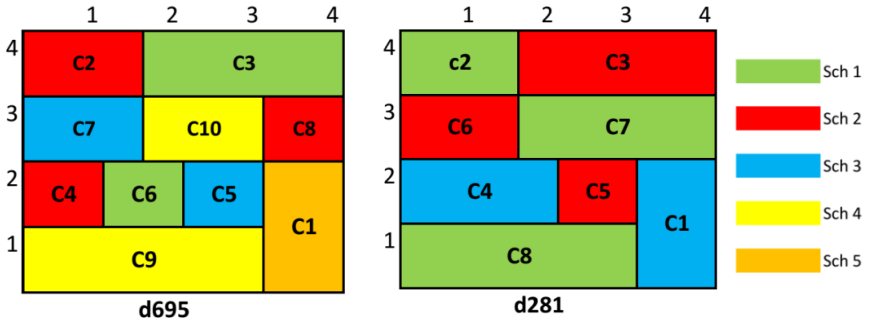

Figure. 10 Pictorial view of Schedules Generated in 2 layered Stacked Structures

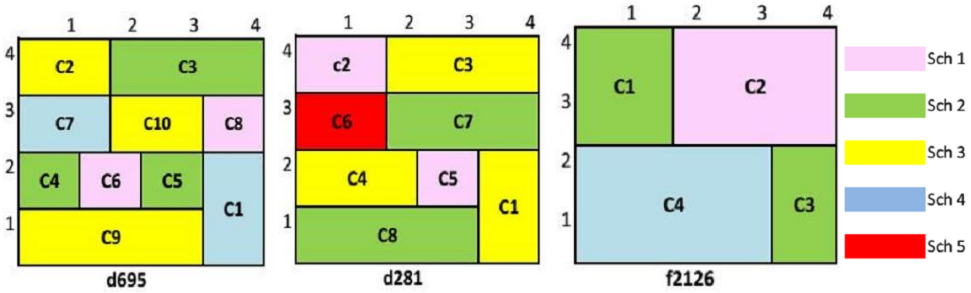

Figure. 11 Pictorial view of Schedules Generated in 3 layered Stacked Structures

These figures represent the cores of various layers of the circuit selected for testing based on the proposed Scheduling Algorithm in a simplified pictorial view where all the layers are of equal size and the placement of cores is appearing on them. Cores selected in a particular Test Schedule in any SoC are shown with a particular color. It can be inferred after going through these pictures that in any Test Schedule, the cores which are selected not only avoid sharing any of the sides with each other but also avoid overlapping each other in vertical direction. This methodology of Adjacency Exclusion is the back bone of this proposed Scheduling Algorithm.

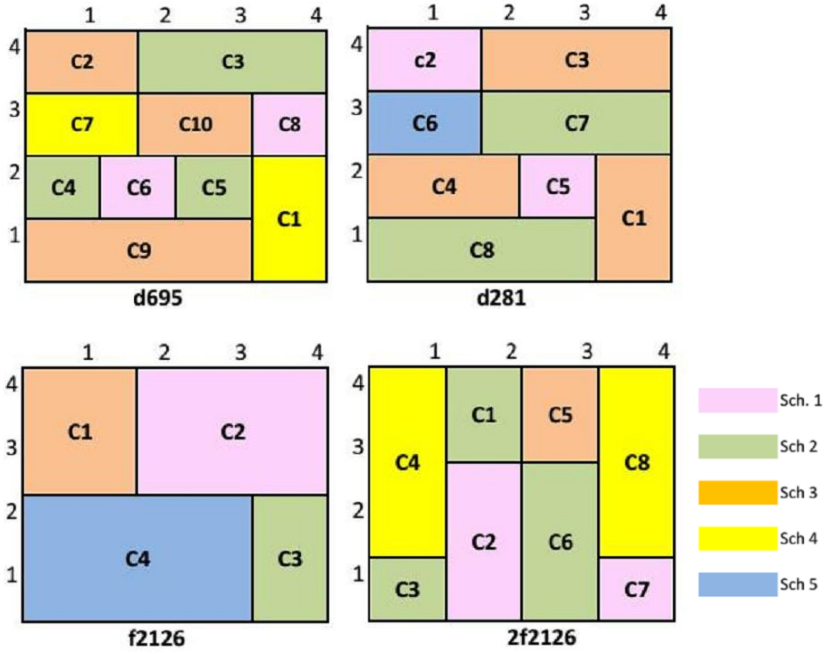

Figure. 12 Pictorial view of Schedules Generated in 4 layered Stacked Structures

VII. Schedule Application

The test application in the form of power trace was done in HotSpot tool. The Hotspot tool is an accurate and fast model based on an equivalent circuit of thermal resistances and capacitances that correspond to micro architecture blocks and essential aspects of the thermal package. Validation of this model has been performed using finite element simulation. The chips today are typically packaged with the die placed on a spreader plate, made of aluminium, copper, or some other highly conductive material, which is in turn placed against a heat sink of aluminium or copper. This is the configuration modelled by HotSpot. We have prepared our stacks similarly, consisting of stacks with interface material in between, heat spreader and heat sink. HotSpot dynamically generates the RC circuit when provided with an input consisting of the blocks' layout and their areas. It is also provided with a power input values (these are the values for the current sources) over any time step and the present temperature of each block. It then generates the temperatures at the centre of each block. We provided Hotspot with the inputs details of our stacks, viz. floorplan, power trace files, area and initial temperatures.

The temperature rise of all cores after application of the schedules was observed as shown in graphs in Figure. 13, Figure. 14 and Figure. 15. Figure. 13 shows the results of scheduling as per the proposed method for 2 Layer stack. The maximum temperature rise i.e.346.7 K is observed for core 6 of d695 which has the highest P_Trace value. In 3 Layers, the maximum temperature rise i.e.367 K is observed for core 6 of layer 4 followed by core 2 of layer 0 (364.5 K) as both have very high value of P_Trace and thereby fall in first schedule itself.

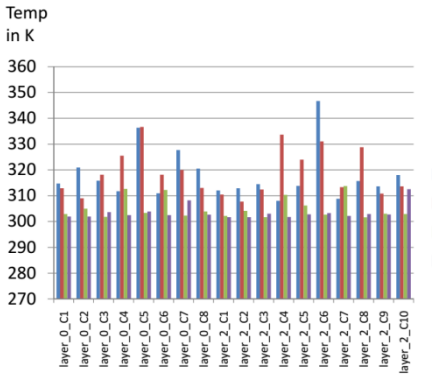

Figure. 13 Results 2 Layers Scheduling Testing

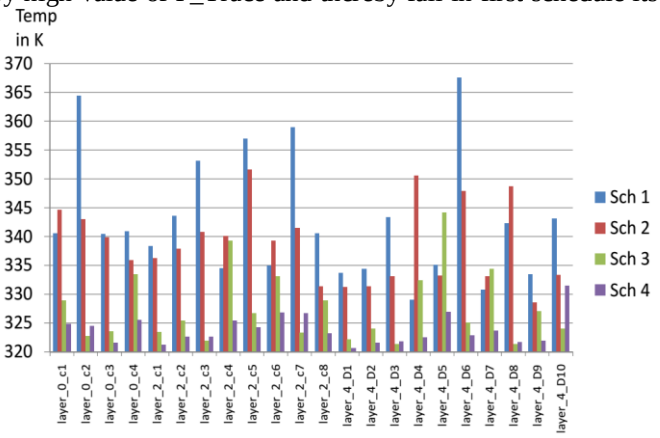

Figure. 14 Results 3 Layers Scheduling Testing

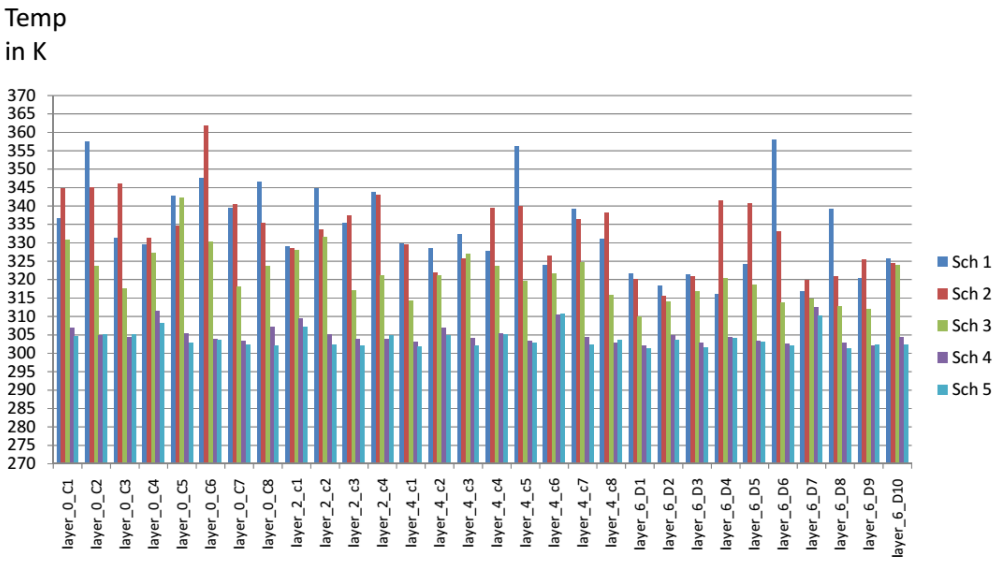

Figure. 15 Results 4 Layers Scheduling Testing

Figure. 15 shows the scheduling results for a 4 Layer stack where maximum temperature observed is 362 K for core 6 of topmost layer.

The results so obtained are of proposed Scheduling Algorithm. In order to assess the suitability of this Algorithm, we compare the results with the Sequential Testing where the cores are tested sequentially i.e. one layer after other to ensure concurrent testing of cores. The results after sequential testing are displayed in Figure

16, 17 and 18 where Figure 16 is for 2 layer Sequential, Figure 17 is for 3 layer Sequential and Figure 18 is for 4 layer Sequential testing. Comparison of the results of sequential and scheduling testing of 2, 3 and 4 layers is also shown in details in tables 9, 10 and 11 where the parameters are compared exhaustively to prove the superiority of proposed scheduling Algorithm over conventional sequential testing.

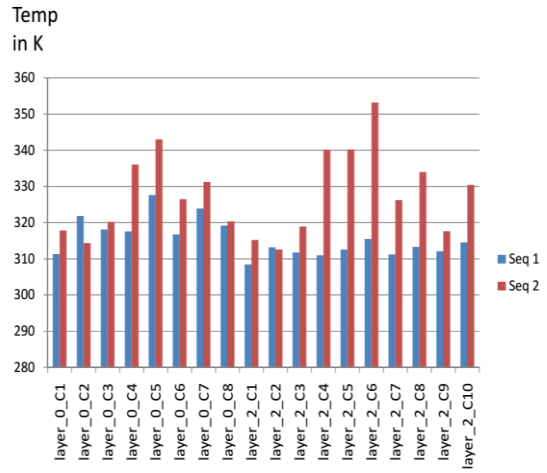

Figure. 16 Results 2 Layers Sequential Testing

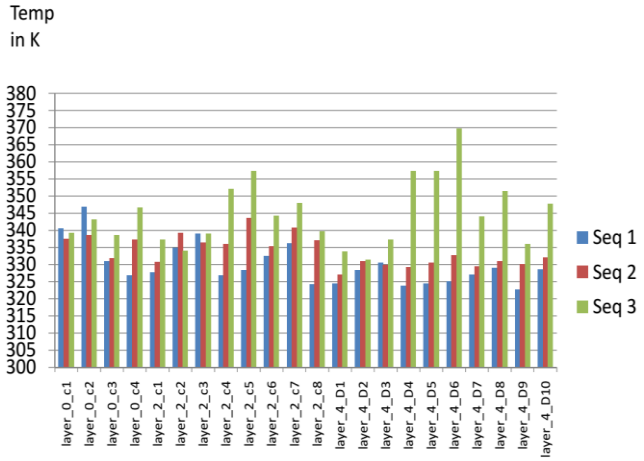

Figure. 17 Results 3 Layers Sequential Testing

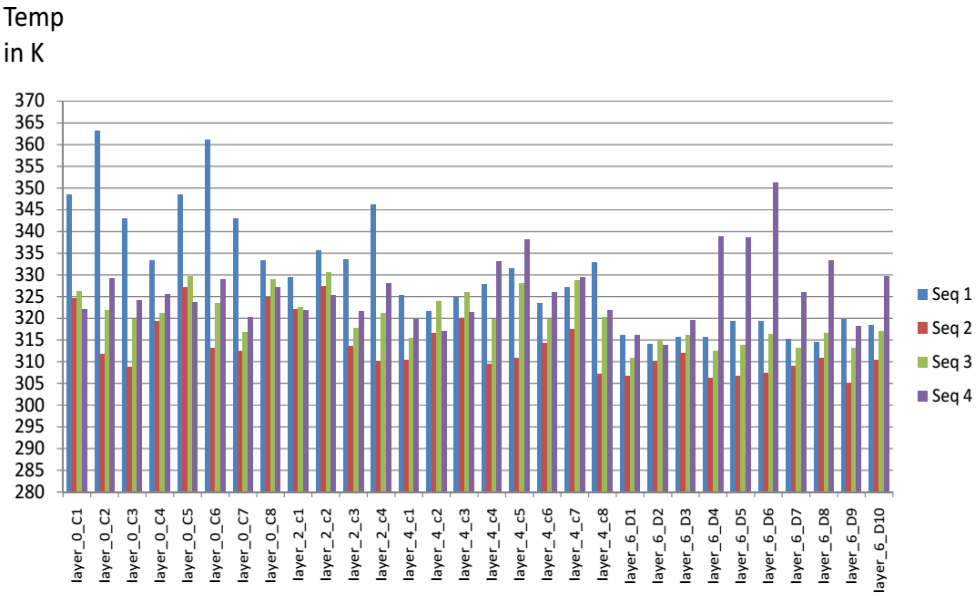

Figure. 18 Results 4 Layers Sequential Testing

Table 9. Results comparison of 2 Layers

Sequential				**Scheduling**			
Test	**Core**	**Max Temp.K**	**Time (ms)**	**Test**	**Core**	**Max Temp.K**	**Time (ms)**
Seq 1	C 0,5	327.62	32.56	Sch 1	C 2,6	346.71	18.5
Seq 2	C 2,6	353.23	25.21	Sch 2	C 0,5	336.65	13.3
				Sch 3	C 2,7	313.76	12.32
				Sch 4	C 2,10	312.53	12.74
Total Time			57.77	Total Time			56.86
Test Max Temp = 353.23 K				Test Max Temp = 346.71 K			
Layer		Avg. Temp K		Layer		Avg. Temp K	
D281		322.87		D281		311.99	
D695		334.36		D695		310.84	
% time improvement of Scheduling testing over Sequential testing = 2%							

Table 10. Results comparison of 3 Layers

Sequential				Scheduling			
Test	**Core**	**Max Temp.K**	**Time (ms)**	**Test**	**Core**	**Max Temp.K**	**Time (ms)**
Seq 1	C 0,2	346.81	23.9	Sch 1	C 4,6	367.51	23.9
Seq 2	C 2,5	343.61	17.01	Sch 2	C 2,5	351.6	13.4
Seq 3	C 4,6	369.66	32.8	Sch 3	C 4,5	344.21	17.01
				Sch 4	C 4,10	331.51	12.8
Total Time			73.7	Total Time			67.16
Test Max Temp = 369.66 K				Test Max Temp = 367.51 K			
Layer		Avg. Temp K		Layer		Avg. Temp K	
F2126		338.21		F2126		334.68	
D281		337.54		D281		334.2	
D695		334.45		D695		331.87	
% time improvement of Scheduling testing over Sequential testing = 9%							

Table 11. Results comparison of 4 Layers

Sequential				Scheduling			
Test	**Core**	**Max Temp.K**	**Time (ms)**	**Test**	**Core**	**Max Temp.K**	**Time (ms)**
Seq 1	C 0,6	363.05	23.9	Sch 1	C 6.6	357.92	32.8
Seq 2	C 4,2	327.35	23.9	Sch 2	C 0,6	361.76	23.9
Seq 3	C 4,2	330.59	13.4	Sch 3	C 0,5	342.04	12.8
Seq 4	C 0,6	351.18	32.8	Sch 4	C 6,7	312.56	11.1
				Sch 5	C 4,6	310.65	3.36
Total Time			94.06	Total Time			84.2
Test Max Temp = 363.05 K				Test Max Temp = 361.79 K			
Layer		Avg. Temp K		Layer		Avg. Temp K	
2F2126		328.24		2F2126		324.15	
F2126		325.40		F2126		321.62	
D281		322.17		D281		319.19	
D695		317.06		D695		315.14	
% time improvement of Scheduling testing over Sequential testing = 10.5%							

It is very much clear from the above tabulated results that the performance of proposed Test Scheduling for parallel testing of cores in SoC is much better than that of conventional Sequential testing in terms of Highest Temperature achieved during the complete test, Highest Temperature of individual test, mean temperature of individual layer and mean temperature of complete chip. It is observed that there is an improvement in the time of testing also in the proposed schedule and the testing time performance improves with the increase in the number of layers. It can therefore be concluded that the proposed Parallel Test Scheduling Algorithm is better in all respects as compared to the Sequential testing of the cores

VIII. Conclusion And Future Works

The paper has outlined an efficient method of test scheduling of 3D SoCs. Using the method a marked reduction in temperature rise of cores under test is observed. This method also leads to reduction of hotspot formation which can permanently damage the chips. In future work, we intend to extend this algorithm to test more than 4 layers stacks to further ascertain its utility. We also intend to introduce partitioning of test schedules during this parallel test scheduling which is expected to reduce the temperature rise during test and will also result in reduced test time. We are in the process of developing a smart algorithm for this. We also intend to make use of some cooling methods like TSVs and liquid cooling to further keep a check on temperature rise.

References

Proceedings Papers:

[1]. V. Iyengar et.al, " Test Access mechanism Optimization, Test Scheduling and tester Data Volume reductionfor System-on-Chip," IEEE Trans. On Computers,Vol.52, no. 12 pp 1619-1632,Dec.2003.

[2]. Y.Huang et.al, "Optimal CoreWrapper Width Selection and SoC Test Schedulingbased on 3-D Bin Packing Algorithm," Proc. Of IEEE International Test Conference (ITC) pp. 74-82,2002.

[3]. S.Samii et. Al, "Cycle Accurate Test Power Modeling and its Application to SoC Test Scheduling," Proc. IEEE ITC pp 1-10, 2006.

[4]. W. D. Tseng, "Power –Oriented test Scheduling for SoCs," International Journal of Computer Science and Network Security, vol.6, no.11 November 2006.
[5]. X. Wu et.al, " Test- Access Mechanism Optimization for Core-based Three Dimensional SoCs," in Proc ICCD, 2008, pp 212-218.
[6]. Jiang et.al, "Test Architecture Design and Optimization for Three Dimensional SoCs," . In Proc. DATE 2009,pp 220-225.
[7]. E. J. Marinissen, " Testing TSV based Three-Dimensional Stacked ICs," in Proc. DATE 2010, pp. 1689-1694.
[8]. C.R. Kime and K.K. Saluja, "Test scheduling in testable VLSI circuits. Intl. Symposium on Fault Tolerant Computers," pages 406–412, 1982.
[9]. H. Krawczyk and M. Kubale, " An Approximation Algorithm for Diagnostic Test scheduling in Multicomputer Systems," IEEE Trans. on Computers, C-34(9):869–872, 1985.
[10]. M.S. Abadir and M.A. Breuer, " Constructing optimal test schedules for VLSI circuits having built-in test hardware," Intl. Symposium on Fault Tolerant Computers, pages 165–171, 1985.
[11]. R.M. Chou, K.K. Saluja, and V.D. Agrawal, " Power constraint scheduling of tests. In Intl. Conference on VLSI Design," pages 271–274, 1994.
[12]. R.M. Chou, K.K. Saluja, and V.D. Agrawal, " Scheduling tests for VLSI systems Under Power Constraints," IEEE Trans. on Very Large Scale Integration Systems, 5(2):175–185, 1997.
[13]. V. Iyengar, K. Chakrabarty, and E.J. Marinissen, " Wrapper/TAM cooptimization, constraint-driven test scheduling, and tester data volume reduction for SOCs," In Design Automation Conference, pages 685–690, 2002.
[14]. V. Iyengar, K. Chakrabarty, and E.J. Marinissen, " On using rectangle packing for SOC wrapper/TAM co-optimization," In VLSI Test Symposium, pages 253–258, 2002.
[15]. P.M. Rosinger, B.M. Al-Hashimi, and N. Nicolici, " Power constrained test scheduling using power profile manipulation,"In Intl. Symposium on Circuits and Systems, pages 251–254, 2001.
[16]. C. Liu and V. Iyengar, " Test scheduling with thermal optimization for network-on-chip systems using variable-rate on-chip clocking," In Design, Automation and Test in Europe Conference, pages 652–657, 2006.
[17]. Z. He, Z. Peng, P. Eles, P. Rosinger, and B.M. Al-Hashimi, "ThermalAware SoC Test Scheduling with Test Set Partitioning and Interleaving," In Intl. Symposium on Defect and Fault Tolerance in VLSI Systems, pages 477–485, 2006.
[18]. K. Skadron et.al , "Temperature Aware Microarchitecture", in Proc. ISCA-30 pages 2-13, June 2003.
[19]. Erik J. Marinisen, V. Iyenger and K. Chakrabarty, "A set of benchmarks for modular Testing of SoCs", ITC, 2002.