An Integrated Temperature-Cycling Acceleration and Test Technique for 3D Stacked ICs

Nima Aghaee, Zebo Peng, and Petru Eles
Embedded Systems Laboratory (ESLAB), Linkoping University, Sweden
{nima.aghaee, zebo.peng, petru.eles}@liu.se

***Abstract*—In a modern 3D IC, electrical connections between vertically stacked dies are made using through silicon vias. Through silicon vias are subject to undesirable early-life effects such as protrusion as well as void formation and growth. These effects result in opens, resistive opens, and stress induced carrier mobility reduction, and consequently circuit failures. Operating the ICs under extreme temperature cycling can effectively accelerate such early-life failures and make them detectable at the manufacturing test process. An integrated temperature-cycling acceleration and test technique is introduced in this paper that integrates a temperature-cycling acceleration procedure with pre-, mid-, and post-bond tests for 3D ICs. Moreover, it reduces the need for costly temperature chamber based temperature-cycling acceleration procedures. All these result in a reduction in the overall test costs. The proposed method is a schedule-based solution that creates the required temperature cycling effect along with performing the tests. Experimental results demonstrate its efficiency.**

I. Introduction

It is well-known that large and frequent temperature changes create fatigue and wearout in materials. This phenomenon is called temperature cycling. Temperature cycling affects ICs by causing various symptoms including solder joint fatigue, fracture in bond wires, and deformation of the dies [1]. In addition to these undesirable effects, 3D stacked ICs (3D-SIC) suffer from defects related to Through Silicon Vias (TSV). TSV protrusion and void formation in TSV are two of such defect mechanisms. These mechanisms are accelerated by temperature cycling [2]–[4]. Defects such as resistive opens and stress induced carrier mobility reduction are also caused by temperature cycling.

Temperature cycling accelerates some of the defect mechanisms, as pointed out above. Therefore, operating the dies under intensive temperature cycling can effectively accelerate such failures so that they are detected in the subsequent test, before the 3D-SIC is shipped to the customers. This procedure is called temperature-cycling acceleration [5], [6]. An example for the effect of temperature cycling is the protrusion of TSV out of the die. Immediately after TSV fabrication, there is normally no protrusion and the TSV has the same size as the die. When temperature-cycling acceleration starts, the TSV length increases with the number of cycles [2], [3]. After a certain number of cycles, the TSV length approaches a saturation level. Continuing the temperature cycling has almost no effect on the TSV length, afterwards. This process is affected by electrical current that flows in TSV [2], [3]. Therefore, operating the IC under extreme temperature cycling helps to speed up the process.

The standard procedure for temperature-cycling acceleration is based on one or multiple temperature chambers [6]. Although this procedure is affordable for 2D ICs, it is likely to be expensive for 3D-SICs. This is due to TSV-related defects and a manufacturing process that includes multiple bonding stages. Usually, in 3D-SIC manufacturing process, pre-, mid-, or post-bond tests are introduced in order to avoid: (1) wasting a good die that is bonded to a bad die or stack, (2) wasting bonding effort spend for bonding bad dies or stacks, and (3) wasting packaging effort spent for packaging a bad stack. Depending on the cost distribution, temperature-cycling acceleration could be beneficial at one or multiple test stages. Integrating the temperature-cycling acceleration with the normal tests that are performed at different stages and eliminating the need for temperature chambers will reduce the overall manufacturing costs.

The test power density is very large for modern core-based system-on-chips, including 3D-SICs, especially since the tests are mostly scan-based [7], [8]. High power densities will lead to very high temperatures, in particular for 3D-SICs, and therefore should be taken into account when planning the tests [9], [10]. On the other hand, this otherwise undesirable effect is utilized in this paper to create temperature-cycling acceleration. Temperature-cycling acceleration is achieved by switching between tests and cooling intervals. A cooling interval is the time interval that no stimuli are applied to a core and, therefore, the core's temperature decreases. Some cooling intervals are expected to be present in the original test schedule for thermal safety reasons. If required, even more intensive temperature-cycling acceleration is achieved by introducing cooling intervals and heating sequences into the process. A heating sequence consists of stimuli that generate large switching activities in a core and, therefore, increases the core's temperature rapidly. The interaction of cooling intervals and heating sequences will create the required temperature-cycling acceleration effect, as presented in this paper.

II. Related Works

Traditionally, temperature-cycling acceleration is performed using one or multiple temperature chambers followed by a final test [5], [6]. This approach will be in many cases too expensive to be performed at pre-, mid-, and post-bond stages for 3D-SICs. The downside of this traditional approach includes: (1) costs for running the chambers and (2) time and equipment required for handling the dies/stacks between test equipment and chambers. In order to avoid costs indicated in (1) and (2), in current practice, some or even all of the temperature-cycling acceleration operations are avoided. This increases the temperature-cycling related early-life failures in the final products.

Several works that are not directly related to temperature cycling but are, in methodology, similar to our proposed technique are briefly reviewed here. A burn-in technique is proposed in [11] to enforce specific temperature gradients on the IC. This results in an effective burn-in process for gradient-dependent early-life failures. A test technique is proposed in [12] to perform tests when specific temperature gradients are enforced on the IC. This helps to detect gradient-dependent defects that are usually related to signal delay and clock jitter.

A linear programming approach is used in [9] to generate thermally-safe test schedules for 3D-SICs. A temperature-based test partitioning technique is introduced in [13] in order to generate fast and thermally-safe test schedules for 3D-SICs. A thermal-aware test scheduling approach is introduced in [10] for stacked multi-chip modules, which tries to achieve a vertical uniform temperature distribution throughout the 3D IC during the test.

Two different methods for detecting temperature-dependent defects are introduced in [14] and [15]. These methods guarantee

978-1-4799-7792-5/15/$31.00 ©2015 IEEE

that the cores' temperatures are kept within the specified range when the corresponding tests are applied. They focus on the temperature of the individual cores that are under test and the temperatures of other cores are neglected.

Speeding up the test by carefully planning safety margins that counteract negative effects of process variation is addressed in [16]. The test temperatures are kept sufficiently low by introducing cooling intervals into the test schedule. The cooling intervals are carefully planned using temperature simulations. A fast temperature simulation technique is suggested in [16].

These existing methods for controlling the chips' temperatures try to respect a global upper temperature limit to prevent overheating or to respect upper and lower bounds for cores in order to target temperature-dependent or gradient-dependent defects. In all above cases, modules' temperatures are considered independent of their cycling effects. To our knowledge, there is no existing method for controlled temperature-cycling acceleration without utilizing temperature chambers. This paper is the first to present a technique to rapidly achieve the required amount of temperature cycling without a temperature chamber, and integrated with application of the normal tests.

III. Problem Formulation

Assume that there are M modules in an IC. The modules are on one or multiple dies. Tests applied to a particular module can be started and stopped independently. The modules could be cores with core wrappers in a core-based design. The extension of this scenario to 3D-SIC is proposed as the IEEE P1838 standard [17]. Test stimuli are, therefore, transferred through a Test Access Mechanism (TAM) to the targeted module. It is assumed that the TAM only affords W (a positive integer number) modules to be accessed simultaneously.

As discussed before, along with pre-, mid-, or, post-bond tests, temperature-cycling acceleration might be beneficial. In this case, there will be tests that target cycling-dependent defects in addition to other tests. In this paper, tests that target cycling-dependent defects are called cycling tests and the other tests are called normal tests. Normal tests are scheduled along with heating and cooling intervals in order to generate the required amount of temperature cycling. Then the cycling tests can be performed. The effect of temperature cycling can be described based on the Amount of Temperature Cycling induced fatigue (denoted by ATC_m for module m). Based on a modified Coffin-Manson or Paris Law model [1], [18], ATC is estimated as:

$$ATC_m \cong \frac{N_m}{k_0} \times \left(\frac{\Delta\theta_m}{k_1}\right)^{\gamma} \quad (1)$$

Considering module m, in this equation N_m is the number of temperature cycles and $\Delta\theta_m$ is the amplitude of temperature changes during cycles. k_0, k_1, and γ are constant that are obtained analytically or empirically by reliability analysts. γ is a constant that depends on the failure mechanisms and materials involved in the process. Its value is usually between one and nine ($1 \leq \gamma \leq 9$). The comprehensive explanation and the details of these constants can be found in [1] and [18].

Equation 1 is easy to use when the temperature fluctuates in a uniform periodic manner similar to Fig. 1a. In this case five cycles of amplitude $\Delta\theta$ are easily identified. In the general case, for example when the IC is under test, the temperature changes with time are irregular, similar to Fig. 1b. In this case it is impossible to easily identify cycles and their amplitudes. In such a situation, the amount of temperature cycling induced fatigue, ATC, is calculated using the widely used Rainflow-counting algorithm [19]. The required amount of temperature cycling is denoted by ATC_m^R. The current amount of temperature cycling generated by the proposed integrated test up, to time t, is denoted by $ATC_m(t)$. For a certain test schedule, the temperature evolution curves are obtained using temperature simulations. Then a fast version of Rainflow-counting algorithm, introduced in [18], calculates $ATC_m(t)$. Assuming that for $t < \hat{t}_m$, $ATC_m(t) < ATC_m^R$, only normal tests can be applied before time $\hat{t}_m$. The cycling tests can be applied only after the required amount of cycling related stress (ATC_m^R) has been achieved. Therefore, after time $\hat{t}_m$, cycling tests can be applied too. The Test Application Time (TAT_m) marks the moment that testing module m is complete. TAT_m consists of the time spent before and after $\hat{t}_m$. The goal is to generate a schedule with minimal overall test application time (TAT). TAT is defined as $\max_m\{TAT_m\}$.

In addition to the dynamic power generated by switching activities, there will be a stray power dissipation by the module. The stray power could not be independently controlled with available test controls. It consists of the leakage power in addition to the clock networks' power. Its exact value depends on the module's current temperature. In this paper, stray power is taken into account during temperature simulations.

High power test stimuli and heating sequences will increase the modules temperatures. The temperature can increase so much that it induces unrealistic failures and harms the device. In order to avoid damages caused by overheating, the modules' temperatures must be kept below the overheating temperature ($\theta^{overheating}$). The overheating temperature is equal to the temperature limit minus a safety margin to ensure trouble free operation.

The problem is summarized as follows. The inputs to the proposed method include the IC's thermal model, the IC's electrical model (e.g., specification of the TAM and power-related specifications), the switching activities of the tests and heating sequences, the ambient temperature ($\theta^{ambient}$), the cycling and normal test sets, and the required amount of temperature cycling, ATC_m^R. The objective is to minimize the test application time. The output is the corresponding schedule that guides the application of the tests, heating sequences, and cooling intervals to the modules so that they perform all the tests, given the required ATC.

IV. Motivational Example

A. An IC with Two Modules

As an example, consider an IC with two modules. Assume that $\theta^{overheating} = 150$°C and $\theta^{ambient} = 30$°C. The required amount of temperature cycling is ATC_0^R and ATC_1^R for modules m_0 and m_1, respectively. The TAM can only support one module to be tested at a time (W=1). A three-phase approach consisting of the following three phases is discussed here. Phase 1: normal tests are scheduled. A thermal aware scheduling of tests based on the method proposed in [20] is used. The corresponding temperature curves are shown in Fig. 2 (green for m_0 and blue for m_1). Phase 1 starts at time 0 and end at t^0.

Phase 2 starts by evaluating the achieved ATC in phase 1. This value is less than required in this example. Therefore, phase 2

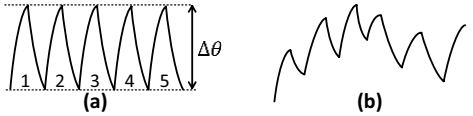

Figure 1. Temperature patterns. (a) Uniform periodic. (b) Irregular

starts to create additional temperature cycling. This is done by applying the heating sequences and cooling intervals. Corresponding temperature cycles can be seen in Fig. 2 from t^0 to $\hat{t}$. As mentioned earlier $\hat{t}_m$ marks the point that the required ATC is achieved for module m. Phase 2 ends when all required ATCs for all modules are met. This point is marked with $\hat{t}$ that is defined as $\max_m\{\hat{t}_m\}$. After this, phase 3 starts by applying the cycling tests. Phase 3 ends when all the tests are complete. This point is marked with TAT.

As discussed before, a small TAT is desirable. Test application time from 0 to t^0 and from $\hat{t}$ to TAT are already minimized by the given third-party test scheduling algorithm. The only optimization opportunity in this three-phase approach is to speed up phase 2. This means that a large ATC should be achieved in a short time. Therefore, $ATC_m(t)/t$ should be maximized. Here we assume a uniform periodic temperature profile. Moreover, for this motivational example we assuming that in equation 1, $k_0 = 1$ and $k_1 = 1$. The ATC rate (denoted by ρ_m for module m) is defined as

$$\rho_m = \frac{ATC_m(t)}{t} = \frac{N_m(t)}{t} \times (\Delta\theta_m)^\gamma. \quad (2)$$

Frequency (i. e., the number of cycles per time unit) of temperature changes depends on the physics of the system and the amplitude of temperature changes, $\Delta\theta$. It is possible to achieve a high frequency if $\Delta\theta$ is small. A larger $\Delta\theta$, on the other hand, may increase the ATC, only if it dominates the resulting reduction in the frequency.

B. An IC with One Module

In order to clarify the tradeoff between the frequency and the amplitude of the temperature cycling, the physics of the system should be captured in the ATC rate equation (equation 2). In the following this is done for a simple IC with only one module. Assume that the IC's thermal behavior could be modeled with only one thermal element. This element has a heat capacity equal to C. Thermal resistance between the element and the ambient is R. Assume that the heating sequence generates a power equal to P and the power during a cooling interval is zero. Assume that the temperature varies between $\mu - \sigma$ and $\mu + \sigma$. Both μ and σ are positive real numbers.

The period of a temperature cycle is denoted by T. This period consists of a rise time denoted by T_r plus a fall time denoted by T_f. T_r is the time taken for the temperature to increase from $\mu - \sigma$ to $\mu + \sigma$. T_f is the time taken for the temperature to decrease from $\mu + \sigma$ to $\mu - \sigma$. These values are calculated as follows. First, the system's differential equation is solved in the time domain [16].

$$\theta^t = \theta^0 \cdot \exp\left(-\frac{t}{R\cdot C}\right) + P \cdot R \cdot \left(1 - \exp\left(-\frac{t}{R\cdot C}\right)\right). \quad (3)$$

Let us denote $R \cdot C$ by RC and $P \cdot R$ by PR. For heating:

$$(\mu + \sigma) = (\mu - \sigma)\exp\left(-\frac{t}{RC}\right) + PR\left(1 - \exp\left(-\frac{t}{RC}\right)\right) \quad (4)$$

Then

$$T_r = RC \times \ln\left(\frac{\mu-\sigma-PR}{\mu+\sigma-PR}\right). \quad (5a)$$

Similarly for cooling, T_f can be calculated:

$$T_f = RC \times \ln\left(\frac{\mu+\sigma}{\mu-\sigma}\right). \quad (5b)$$

The period, T, is calculated as follows:

$$T = T_r + T_f = RC \times \ln\left(\frac{(\mu-\sigma-PR)(\mu+\sigma)}{(\mu+\sigma-PR)(\mu-\sigma)}\right). \quad (6)$$

Now, the ATC rate (equation 2) could be re-written incorporating the physics of the system:

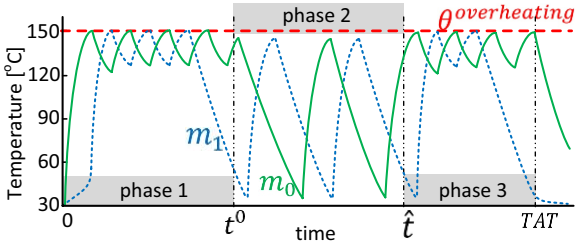

Figure 2. Temperature curves for the three-phase approach.

$$\rho_m = \frac{(2\sigma)^\gamma}{RC\times\ln\left(\frac{(\mu-\sigma-PR)(\mu+\sigma)}{(\mu+\sigma-PR)(\mu-\sigma)}\right)} = \frac{(2\sigma)^\gamma}{RC\times\ln(\xi)}. \quad (7)$$

Let us first focus on the optimal value for μ, assuming that σ is constant. In this case optimality happens when the denominator in equation 7 is minimized. Considering a realistic situation, this is equivalent to finding the minimum for

$$\xi = \frac{(\mu-\sigma-PR)(\mu+\sigma)}{(\mu+\sigma-PR)(\mu-\sigma)}. \quad (8)$$

Following the classical approach:

$$\frac{d}{d\mu}\xi = 0 \rightarrow \frac{(2\mu-PR)\left((\mu-\sigma-PR)(\mu+\sigma)-(\mu+\sigma-PR)(\mu-\sigma)\right)}{\left((\mu+\sigma-PR)(\mu-\sigma)\right)^2} = 0 \quad (9)$$

The valid solution is $\mu = PR/2$. Here for the sake of simplicity, the ambient temperature was not included in the equations. Since the temperature model is a Linear Time-Invariant (LTI) system [16], the ambient temperature can be added later on. Assume that power and resistance values are so that $PR = 120℃$. This means that considering the ambient temperature (30℃), the IC's temperature will reach to 150℃ if no control is applied. Consequently, the optimal value for μ is $\frac{120℃}{2} + 30℃ = 90℃$.

The resulted equations for finding the optimal value for σ do not have a simple analytical form. Therefore, a numerical method is utilized. The ATC rate, ρ, is plotted in Fig. 3 versus σ for $\mu = 90℃$. It is assumed that $\gamma = 4$ and $RC = 50$ µs. The ATC rate is maximal at $\sigma_{max} = 55.6℃$. For values of σ less than σ_{max} the ATC rate increases by increase in σ. This is due to the increase in $(\Delta\theta_m)^\gamma$ dominating the decrease in $N_m(t)/t$ in equation 2. For larger σ values the ATC rate decreases by increase in σ. This is due to the increase in $(\Delta\theta_m)^\gamma$ being dominated by the decrease in $N_m(t)/t$. In other words, a very large temperature cycle takes too much time to complete.

V. Three-Phase Approach

Section IV.B presents a technique to find the best temperature interval, and if the high temperature level $(\mu + \sigma)$ is under the overheating temperature, it is fine. However, often the overheating temperature is relatively low compared with $\mu + \sigma$. For the example in section IV.B, $\mu + \sigma$ is equal to 145.6℃ while the overheating temperature might be at 120℃. There are some other complications, as well. In practice there are a number of modules, instead of one. In order to accurately simulate the temperatures, the number of thermal elements is larger than the number of modules. Their temperatures depend on each other due to heat transfer among thermal elements. Furthermore, the power values

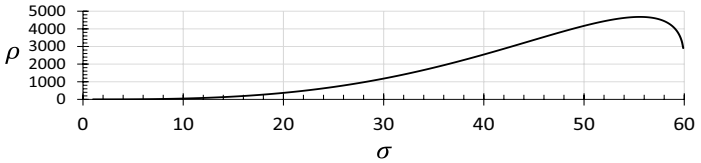

Figure 3. ATC rate, ρ, versus σ for three-phase approach.

fluctuate with time. Besides, power values include the stray powers that depend on the temperature due to the temperature dependent leakage currents. Additionally, the modules may not be able to receive their heating sequences at desired times due to the TAM limitation. A new approach that is capable of taking all these circumstances into account is, therefore, introduced in this section.

As discussed in Section IV.A, in phase 1 and 3 the tests are scheduled using a third-party thermal-aware approach. Therefore, there is nothing we can do in these phases to reduce the TAT. Phase 2, on the other hand, can be designed to minimize the TAT. This was demonstrated using a small example in section IV.B. Assume that in phase 2 the temperature of module m is targeted to cycles between a low temperature level denoted by θ_m^L and a high temperature level denoted by θ_m^H ($\theta_m^L < \theta_m^H$). In comparison with the example in section IV.B, θ_m^L and θ_m^H have roles similar to that of $\mu - \sigma$ and $\mu + \sigma$, respectively.

The heating sequences are powerful enough to raise the module's temperature to θ_m^H and then a cooling interval is immediately applied. The high temperature level should always be lower than the overheating temperature ($\theta_m^H < \theta^{overheating}$) to avoid any damage. Natural cooling will reduces the module's temperature to θ_m^L and then the heating sequences should be applied again. But, the TAM might not be available at the moment. Consequently, the temperature may fall below θ_m^L from time to time.

An on-the-fly approach is used to determine the heating sequences schedule in phase 2, based on temperature simulations. The temperatures that are obtained by simulation are then compared with θ_m^L and θ_m^H in order to determine the schedule. Heating sequences for different modules will then compete for access to TAM. The priority is decided based on the following equation.

$$\pi_m = (\theta_m^L - \theta_m) \times \frac{ATC_m^R}{\epsilon + ATC_m} \qquad (10)$$

The priority is higher if the module's current temperature is much below θ_m^L. Note that the priorities are calculated only for modules that need heating, therefore $\theta_m < \theta_m^L$. This is because if a module gets really cold, it takes too much time to warm it up again. A module that has a large amount of temperature cycling left to fill has a higher priority. This is indicated by $\frac{ATC_m^R}{\epsilon + ATC_m}$. Such a module is likely to need a relatively long time to achieve its required ATC. It is likely that at the later stages of phase 2 this module remains alone. This implies that the interleaving opportunities are reduced. Consequently TAM utilization may decrease and TAT may increase. A small value, ϵ, is added to the denominator in order to prevent numerical problems when ATC is zero (e.g., at the very beginning of phase 2, if there have not been any normal test). Both θ_m and ATC_m depend on time and are shortened forms of $\theta_m(t)$ and $ATC_m(t)$, respectively, at time t.

The TAT for the schedules generated by this on-the-fly approach depends on θ_m^L and θ_m^H. These temperature levels could assume a range of values provided that $\theta^{ambient} \leq \theta_m^L < \theta_m^H < \theta^{overheating}$. The combination of these temperature levels among different modules affects the TAT. The proper values for these temperature levels will be found in an external optimization loop. In the inner scheduling loop, the temperature levels defined by the outer optimization loop are used to determine the schedule. The outer optimization loop consists of a Particle Swarm Optimization (PSO) algorithm. PSO is a well-known iterative population-based optimization metaheuristic. For each alternative solution in the PSO's population on-the-fly scheduling is performed to compute the cost function (i.e., TAT). A canonical form of PSO [21] is used in this paper in a straightforward manner.

VI. Integrated Approach

The test application time could be shortened if normal tests (phase 1) are integrated into the temperature-cycling acceleration process (phase 2). For example a test can be utilized to heat a module instead of a heating sequence. It could be that the test is not sufficiently powerful to raise the modules' temperature as high as desired for a large temperature cycle (θ_m^H). In this situation, a heating sequence is required to rapidly raise the temperature. The other extreme is when a low power test is used when the temperature is supposed to drop. It may happen that the temperature decreases but not as low as desired for a large temperature cycle (θ_m^L). In this case, a cooling interval can be introduced to achieve a desirably low temperature.

Assume that a test is being utilized for heating as shown in the upper part of Fig. 4a. Assume that the test power for the current time interval is denoted by P_m^{test}. This power, initially, increases the temperature rapidly. Assume that this level of power is applied for a long time. In this case a steady state temperature equal to θ_m^{SS} will eventually be reached. As the current temperature approaches θ_m^{SS}, the rate of increase becomes small. The derivative of the temperature (i.e., rate of increase) is shown in the lower part of Fig. 4a. When this rate reduces below a certain threshold ($threshold^H$ in Fig. 4a), it is time to switch to the heating sequence. This quickly increases the temperature to the desired θ_m^H. Heating sequence (shown as the red curve in Fig. 4a) introduces a temperature increase rate much larger than that of the utilized test. All these mean that it is better to save the rest of the test for a time that the initial temperature is lower and the test can offer a high temperature increase rate.

In order to obtain the rate of temperature change (green curves in Fig. 4), the thermal behavior of the IC should be studied. The thermal behavior can be described as [16]:

$$\boldsymbol{A} \times \frac{d}{dt}\boldsymbol{\Theta} + \boldsymbol{B} \times \boldsymbol{\Theta} = \boldsymbol{P}^{test}. \qquad (11)$$

All the characteristics of the thermal model are captured in two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. $\boldsymbol{\Theta}$ is the temperature vector and $\boldsymbol{P}^{test}$ is the power. $\boldsymbol{\Theta}$ and $\boldsymbol{P}^{test}$ consist of θ_ms and P_m^{test}s, respectively, put together in a vector format. The rate of temperature change is $\frac{d}{dt}\boldsymbol{\Theta}$. Therefore the condition on increase rate is:

$$\frac{d}{dt}\boldsymbol{\Theta} < threshold^H . \qquad (12a)$$

This rate can be described based on the current temperature and following power values using equation 11:

$$\boldsymbol{A}^{-1} \times (\boldsymbol{P}^{test} - \boldsymbol{B} \times \boldsymbol{\Theta}) < threshold^H . \qquad (12b)$$

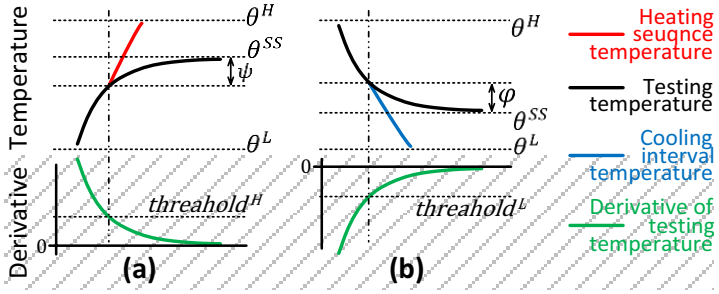

Figure 4. Thresholds in the integrated approach: (a) Heating and (b) Cooling.

This could be re-written to have the condition expressed for the current temperature:

$$\boldsymbol{\Theta} > (\boldsymbol{B}^{-1} \times \boldsymbol{P}^{test}) - (\boldsymbol{B}^{-1} \times \boldsymbol{A} \times threshold^{H}) . \quad (12c)$$

Renaming $(\boldsymbol{B}^{-1} \times \boldsymbol{A} \times threshold^{H})$ to ψ results in:

$$\boldsymbol{\Theta} > (\boldsymbol{B}^{-1} \times \boldsymbol{P}^{test}) - \boldsymbol{\psi} . \quad (13a)$$

Similarly, for the situation that the temperature must decrease, the proper condition for switching from a test to a cooling interval is:

$$\boldsymbol{\Theta} < (\boldsymbol{B}^{-1} \times \boldsymbol{P}^{test}) + \boldsymbol{\varphi} . \quad (13b)$$

Switching to the cooling interval when indicated by the above equation speeds up the temperature decrease towards θ_m^L. This way, the normal tests are utilized in an efficient way during temperature-cycling acceleration process so that the overall test application time is reduced.

In equation 13a and 13b, the term $(\boldsymbol{B}^{-1} \times \boldsymbol{P}^{test})$ is actually a vector consisting of the steady state temperatures. In the steady state situation, $\boldsymbol{\Theta}$ is constant (no variation in time). Therefore, the temperature derivatives in equation 11 are zero ($\frac{d}{dt}\boldsymbol{\Theta} = \boldsymbol{0}$). Consequently, steady state temperatures can be described as

$$\boldsymbol{\Theta}^{SS} = \boldsymbol{B}^{-1} \times \boldsymbol{P}^{test} . \quad (14)$$

Equation 13a and 13b could, therefore, be re-written as

$$\theta_m > \theta_m^{SS} - \psi_m \text{ and} \quad (15a)$$

$$\theta_m < \theta_m^{SS} + \varphi_m . \quad (15b)$$

Therefore, the scheduling heuristic does not need to compute the derivatives of the upcoming tests' temperatures. Instead, it is sufficient to compute the steady state temperatures corresponding to the test sequences, in an initialization stage, before anything else starts. As mentioned before, scheduling is performed offline based on temperature simulations. During the scheduling, current temperatures are compared with the steady state temperatures of the upcoming tests. Whenever equation 15 indicates, a new switching event is introduced in the schedule. For example in Fig. 4b as soon as the temperature falls below $(\theta^{SS} + \varphi)$ the test is interrupted and a cooling interval is introduced. This moment of time is identical to the moment that the magnitude of the temperature derivative becomes smaller than the magnitude indicated by $threshold^{L}$ in the lower part of Fig. 4b. The variables ψ_m and φ_m, are to be optimized along with θ_m^H and θ_m^L to achieve a short TAT. These variables are optimized using a canonical form of particle swarm optimization in a straight-forward manner [21].

Note that the heat transfer among thermal elements is taken into account by using matrix $\boldsymbol{B}$ to obtain the steady state temperatures. The steady state temperature, θ_m^{SS}, is calculated for a short part of the tests that immediately follows. The length of this short part of the test is denoted by λ. Previously, P_m^{test} was introduced as the average power of the test. In fact P_m^{test} is computed for the upcoming test cycles within the time interval λ. The proper value of λ is determined based on the dynamics of the system.

Consider a λ that corresponds to $100x$ $(0 < x < 1)$ percent of the final response (i.e., steady state temperature) to a step input. Assuming a constant power, the temperature equation in the time-domain is [16]

$$\boldsymbol{\theta}^t = \boldsymbol{\alpha}(t) \times \boldsymbol{\theta}^0 + \boldsymbol{\beta}(t) \times \boldsymbol{P}^{test} . \quad (16)$$

Where

$$\boldsymbol{\alpha}(t) = \exp(-\boldsymbol{A}^{-1} \times \boldsymbol{B} \times t) \text{ and} \quad (17a)$$

$$\boldsymbol{\beta}(t) = (\boldsymbol{I} - \boldsymbol{\alpha}(t)) \times \boldsymbol{B}^{-1} \quad (17b)$$

$\boldsymbol{I}$ is the identity matrix. We assume that the step response starts from the initial temperature equal to zero ($\boldsymbol{\theta}^0 = \boldsymbol{0}$). Replacing $\boldsymbol{\theta}^t$ with the $100x$ percent of the final temperature results in

$$x \times \boldsymbol{\theta}^{SS} = \boldsymbol{\beta}(t) \times \boldsymbol{P}^{test} . \quad (18)$$

Replacing $\boldsymbol{\theta}^{SS}$ from equation 14 and $\boldsymbol{\beta}$ from equation 17b in the above equation results in

$$x \times \boldsymbol{I} \times \boldsymbol{B}^{-1} \times \boldsymbol{P}^{test} = (\boldsymbol{I} - \boldsymbol{\alpha}(t)) \times \boldsymbol{B}^{-1} \times \boldsymbol{P}^{test} . \quad (19)$$

Here we are going to replace a scalar time, t, with a matrix of time, $\boldsymbol{\Lambda}$. Besides, we assume that the equivalence of the sides in the above equation is achieved by satisfying the following equation (equation 20). These assumptions work for estimating the value of λ [22].

$$x \times \boldsymbol{I} = \boldsymbol{I} - \boldsymbol{\alpha}(\boldsymbol{\Lambda}) . \quad (20)$$

Replacing $\boldsymbol{\alpha}$ from equation 17a results in

$$\exp(-\boldsymbol{A}^{-1} \times \boldsymbol{B} \times \boldsymbol{\Lambda}) = (1 - x) \times \boldsymbol{I} . \quad (21)$$

And finally

$$\boldsymbol{\Lambda} = \boldsymbol{B}^{-1} \times \boldsymbol{A} \times \ln(1/(1 - x)) . \quad (22)$$

$(\boldsymbol{B}^{-1} \times \boldsymbol{A})$ is the time constants matrix [22]. $\boldsymbol{\Lambda}$ is the matrix that contains the values of λ's. A diagonal element, $\lambda_{m,m}$, represents the proper length for averaging the upcoming test powers for module m. A λ value obtained this way is not unnecessarily short. On the other hand, the use of such $\lambda_{m,m}$ values prevents the temperature changes that are larger than $x \times \theta_m^{SS}$ from going unnoticed. This percentage, x, is only used in the process of estimating the steady state temperatures for the upcoming tests. Note that the temperature simulations are always performed based on the original power sequence (not the average of upcoming tests' power). Therefore, the value of x will not affect them.

Normal tests, heating sequences, and cycling tests may compete for access to TAM. The priority for module m is assessed based on the following criterion.

$$\pi_m = (\theta_m^L - \theta_m) \times \frac{ATC_m^R}{\epsilon + ATC_m} \times r_m \quad (23)$$

Similar to equation 10, the priority is higher if the module is colder or if the remaining ATC is larger. In addition to these factors, the modules' priority is higher if the module's current amount of remaining tests (denoted by r_m) is larger. Both normal and cycling tests are taken into account. The motivation for having r_m is similar to the motivation for having the remaining ATC. In case of normal or cycling tests, "*stop cooling temperature*" introduced in [20] is used instead of θ_m^L. In case of cycling tests, $\frac{ATC_m^R}{\epsilon + ATC_m}$ is replaced with one since the demanded amount of temperature cycling is already met.

VII. Experimental Results

Experiments have been performed to demonstrate that the proposed technique can efficiently achieve desired temperature-cycling accelerations. Moreover, it is demonstrated that the proposed integrated approach offers a small TAT and, therefore, outperforms the three-phase approach. However, if the normal or cycling test schedules provided by a third-party have to be used, three-phase approach must be chosen.

The proposed techniques are evaluated on a set of 24 experimental ICs as detailed in Table I. Column 1 is the IC's number. These ICs have one to four stacked dies (column 2). The ICs with one layer (number 1 to 6) correspond to dies at the pre-bond test stage. The ICs with more than one layer represent a mid-bond or a post-bond test stage. Each die accommodates 2, 12, 20, 30, 42, and 49 modules resulting in two to 196 modules per IC as shown in column 3. The integrated approach achieves shorter

TAT compared with the three-phase approach for all of the experimental ICs. The percentage changes are reported in column 4.

Compared with the three-phase approach, the integrated approach is more complicated for one decision-point[1] in the schedule. On the other hand, the integrated approach produces shorter schedules which mean less decision-points. The CPU times are reported in columns 5 and 6. The integrated approach is not particularly slower than the three-phase approach, as shown in column 7. This means that in average, shorter schedules (smaller TAT) compensate for the more complicated and time-consuming decision-points. Note that if the normal or cycling test schedules are provided by a third-party, the actual CPU times for the three-phase approach will be smaller than the values reported in column 5. In this case, shorter CPU time can be considered as an advantage for the three-phase approach.

CPU times, in general, grow with the number of modules and layers as shown in Fig. 5. The growth rate is acceptably low and the scheduling process for the largest IC (number 24) takes less than 25 minutes to complete.

Another set of experiments are performed in order to evaluate the accuracy of λ values estimated using equation 22. The accurate value for λ is obtained based on high quality temperature simulations. The average error is found to be around five percent. Besides, for 95 percent of the samples, the error is smaller than 14 percent.

VIII. Conclusions

Temperature-cycling acceleration is used in order to detect the cycling-dependent early-life failures. These failures are not considered as a major issue for conventional 2D ICs. Therefore, cycling acceleration is recommended when a high degree of reliability is crucial. However, recent studies suggest that the cycling-dependent early-life failures can be a major issue for 3D stacked ICs. The existing cycling acceleration procedures are very costly since they are usually performed using temperature chambers. In this paper we propose an inexpensive technique to apply the normal tests, heating sequences, and cooling intervals, in an integrated manner in order to achieve the required temperature cycling effect in a short time. This integrated approach can be used in pre-, mid-, and post-bond test stages.

TABLE I. Experimental Results

	IC specifications		Percentage change in TAT	CPU time [sec]		Percentage change in CPU time
	Number of layers	*Number of modules*		*Three-phase Approach*	*Integrated Approach*	
1	1	2	–29.14	1	1	0
2	1	12	–24.35	1	1	0
3	1	20	–22.01	3	2	–33.33
4	1	30	–19.04	6	4	–33.33
5	1	42	–16.29	9	7	–22.22
6	1	49	–21.66	11	8	–27.27
7	2	4	–6.95	1	1	0
8	2	24	–8.19	9	11	+22.22
9	2	40	–13.39	25	25	0
10	2	60	–9.77	50	52	+4
11	2	84	–5.97	96	100	+4.17
12	2	98	–3.89	135	139	+2.96
13	3	6	–8.78	2	1	–50
14	3	36	–12.88	35	35	0
15	3	60	–12.56	99	101	+2.02
16	3	90	–12.52	242	235	–2.89
17	3	126	–13.33	452	475	+5.09
18	3	147	–9.70	569	597	+4.92
19	4	8	–7.91	3	5	+66.67
20	4	48	–5.96	94	100	+6.38
21	4	80	–6.60	241	263	+9.13
22	4	120	–9.73	496	537	+8.27
23	4	168	–5.74	1078	1183	+9.74
24	4	196	–7.69	1333	1455	+9.15
	Average		–12.25			–0.60

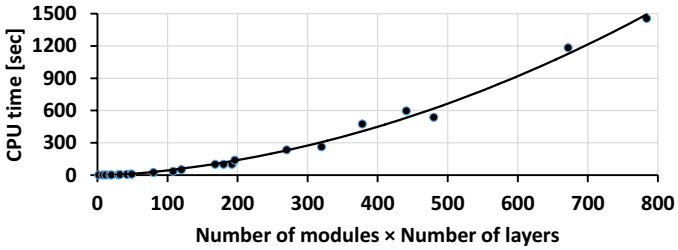

Figure 5. CPU time growth.

IX. References

[1] "Failure mechanisms and models for semiconductor devices," http://www.jedec.org/standards-documents/docs/jep-122e.

[2] P. Kumar, I. Dutta, and M. S. Bakir, "Interfacial effects during thermal cycling of Cu-filled through-silicon vias," *J. Electron. Mater.*, vol. 41, no. 2, pp. 322–335, Feb. 2012.

[3] D. Zhang, K. Hummler, L. Smith, and J. J.-Q. Lu, "Backside TSV protrusion induced by thermal shock and thermal cycling," *Electronic Components and Technology Conference (ECTC), 2013*.

[4] C. Okoro, J. W. Lau, F. Golshany, K. Hummler, and Y. S. Obeng, "A detailed failure analysis examination of the effect of thermal cycling on Cu TSV reliability," *IEEE Trans. Electron Devices*, vol. 61, no. 1, pp. 15–22, Jan. 2014.

[5] "Temperature cycling (MIL-STD-883; METHOD 1010)," *DLA Land and Maritime MilSpecs & Drawings*, Jun-2004. http://www.landandmaritime.dla.mil/programs/milspec/ListDocs.aspx?BasicDoc=MIL-STD-883.

[6] "Temperature cycling," JEDEC Standard, Mar. 2009. http://www.jedec.org/standards-documents/docs/jesd-22-a104d

[7] Y. Bonhomme, P. Girard, C. Landrault, and S. Pravossoudovitch, "Test power: a big issue in large SOC designs," *IEEE International Workshop on Electronic Design, Test and Applications (DELTA), 2002.*

[8] Y. Zorian, "A distributed BIST control scheme for complex VLSI devices," *VTS 1993.*

[1]A decision-point is a point that a module's state (testing/heating/cooling) may change.

[9] S. K. Millican and K. K. Saluja, "Linear programming formulations for thermal-aware test scheduling of 3D-stacked integrated circuits," *ATS 2012*.

[10] N. S. Vinay, I. Rawaty, E. Larsson, M. S. Gaurx, and V. Singh, "Thermal aware test scheduling for stacked multi-chip-modules," *East-West Design & Test Symposium (EWDTS), 2010*.

[11] N. Aghaee, Z. Peng, and P. Eles, "An efficient temperature-gradient based burn-in technique for 3D stacked ICs," *DATE 2014*.

[12] N. Aghaee, Z. Peng, and P. Eles, "Temperature-gradient based test scheduling for 3D stacked ICs," *IEEE International Conference on Electronics, Circuits, and Systems (ICECS), 2013*.

[13] S. K. Millican and K. K. Saluja, "A test partitioning technique for scheduling tests for thermally constrained 3D integrated circuits," *VLSI Design 2014*.

[14] Z. He, Z. Peng, and P. Eles, "Multi-temperature testing for core-based system-on-chip," *DATE 2010*.

[15] C. Yao, K. K. Saluja, and P. Ramanathan, "Temperature dependent test scheduling for multi-core system-on-chip," *ATS 2011*.

[16] N. Aghaee, Z. Peng, and P. Eles, "Process-variation and temperature aware SoC test scheduling technique," *J. Electron. Test.*, vol. 29, no. 4, pp. 499–520, Aug. 2013.

[17] "IEEE SA-P1838-standard for test access architecture for three-dimensional stacked integrated circuits." http://standards.ieee.org/develop/project/1838.html.

[18] M. Musallam and C. M. Johnson, "An efficient implementation of the Rainflow counting algorithm for life consumption estimation," *IEEE Trans. Reliab.*, vol. 61, no. 4, pp. 978–986, Dec. 2012.

[19] M. Matsuishi and T. Endo, "Fatigue of metals subjected to varying stress," *Jpn. Soc. Mech. Eng. Fukuoka Jpn.*, pp. 37–40, 1968.

[20] Z. He, Z. Peng, and P. Eles, "Simulation-driven thermal-safe test time minimization for System-on-Chip," *ATS 2008*.

[21] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intell.*, vol. 1, no. 1, pp. 33–57, Jun. 2007.

[22] T.-M. Lin and C. A. Mead, "Signal delay in general RC networks," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 3, no. 4, pp. 331–349, Oct. 1984.