

Degradation Prediction of Electronic Packages using Machine Learning

Alexandru Prisacaru^{1 3}, Ernesto Oquielis Guerrero¹, Przemyslaw Jakub Gromala¹, Bongtae Han², Guo Qi Zhang³

¹Robert Bosch GmbH, Reliability Modeling and System Optimization (AE/ECU3), Reutlingen, 72703, Germany

²Mechanical Engineering Department, University of Maryland College Park, MD 20742, USA

³Microelectronics Department, Delft University of Technology, Delft, 2600, Netherlands

Abstract

The piezoresistive silicon based stress sensor has the potential to detect a precursor for the Prognostics and Health Management (PHM) implementation in automotive electronics. One solution to enforce reliability in automotive electronics is the use of Machine Learning (ML). One or more physical parameters are being monitored, and algorithms are used to illustrate the health state and predict remaining useful life based on the current and past health information. Piezo-resistive stress sensors are employed to measure the internal stresses of electronic packages, an Acquisition Unit (AU) to read out sensor data and a Raspberry Pi as PHM server to perform evaluation. Accelerated tests in air thermal chamber are performed to get time series data of the stress sensor signals, with which we can know better about how delamination develops inside the package. In this study stress measurements are performed in several electronic packages during the delamination. The delamination is detected by the stress sensor due to the continuous change of the stiffness and the local boundary conditions causing the stresses to change. Moreover, the stress change in multiple cells can give more information regarding the delamination such as the location and its state. Data preprocessing methods to remove outliers and filter raw measurement results, and feature extraction methods to capture only meaningful information by reducing the data are chosen and applied to raw data. A logical assumption is made regarding the data behavior and delamination state, based on data analytics and with Scanning Acoustic Microscope (SAM) images confirmed the delaminated area. FEM simulation are used to provide a qualitatively physical explanation of the stress change due to the delamination. A prognostic model using neural network is trained to estimate the degradation grade. Back Propagation Neural Networks are chosen to provide a fast and quick training for the mechanical stress data.

1. Introduction

There are several definitions of Reliability [1], [2]. In engineering [2], it is set as the "ability of a system or component to perform its required functions under stated conditions for a specified period of time". These conditions refer to conditions like mechanical, thermal, electrical specifications. In this definition, a system [3]

is "a combination of interacting elements (components) organized to achieve one or more stated purposes".

Reliability prediction methods date 70 years into the past [4]. In that time, the concept of Reliability has been extensively used on design, operation and maintenance tasks. On the design stage of a product the concept of Design for Reliability (DfR) [5] appears, in which a model of the product is developed. This model is used by the manufacturer to define the working conditions and lifetime of the product that it will guarantee. A more reliable component (or system) works longer hours on tougher conditions on the operation stage; saving with this maintenance, repairs and replacement costs.

There are two main approaches in reliability: A statistic-based approach (like the Weibull model [6] or the fault tree analysis [7] that considers the reliability of each component to estimate the reliability of a system) or a physics-of-failure-based approach [5] (like the use of Finite Element Method (FEM) software to simulate loading conditions and ensure that the component will perform at its design capacity).

Both models does not predict the *real* status of a singular component. As neither monitors the real time state of the component. Their assessments are of statistical relevance.

It is because of this that, for example, other techniques as redundancy are usually used to ensure zero Downtime (the period in which a system is unavailable) on relevant components. But this solution requires having another component identical in function as the original as backup and this solution is not cost-effective.

With this new notion and the foreseeable advent of widespread complex consumer grade systems as autonomous vehicles, it becomes evident that a new paradigm must be set to overcome the limitations of current reliability models.

A. Automotive Electronics

A recent report by PwC [8] predicts that by 2020 the % of total car cost used on automotive electronics will be 35%, and it will climb to 50% by 2030. The main reasons for this predicted increase are:

- An increment in electronic systems quantity on the vehicle required to keep and monitor the vehicles efficiency required by regulation laws in the EU and the US.

- An increase in functionalities added by the manufacturers to improve the user experience. New electronic systems (that includes new sensors, control units, communication network and power supply) are added to give the car new functions as assisted parking, vehicle cruise speed control, infotainment, emergency braking, etc).
- The push towards automotive driving adds new systems like radar, lidar, gps location and steering control that was not required on traditional autos.

The automotive industry is a particular as consumer demands on its final product are higher than on other products. Car manufacturers must ensure that their electronic products last longer and ensure their functionality.

With the industry moving forward towards offering consumer-grade autonomous driving vehicles, there has never been higher requirements to automotive electronics in terms of power efficiency, size and weight and especially in Reliability.

B. Machine Learning

ML should not be confused with Artificial Intelligence (AI). AI is the concept of a machine exhibiting intelligence similar to humans or animals¹ AI is divided into Weak and Strong AI. The former is an AI that can perform a narrowly defined set of tasks or just one task. The latter is an AI that is capable of applying intelligence to a problem and even showing consciousness. This means that the machine exhibiting Strong AI is not constrained to just one set of problem, as its knowledge and intelligence generalizes to all sets of tasks.

Strong AIs do not exist up to date. All current AIs are Weak as they specialize into solving a set of tasks tightly constrained and work using curated data sets², differing from the human learning experience.

For mentioning a set of examples:

- Google's search algorithm
- Google's image recognition algorithm
- A predictive keyboard algorithm
- An email spam filter
- AlphaGo³ [9]

ML is a subset of artificial intelligence (AI) that creates systems to learn and predict outcomes without manually programming a computer and is also known as predictive analytics or statistical learning" [10]. It is a set of algorithms and techniques focused to learn from data. Here, data is an organized collection of measures and/or classes; and learning means the ability to get information from

data that would generalize to other sets of data. This last is what differences ML techniques from other statistical tools. They focus on the generalization aspect of the data analysis and not only in creating a model that works with the data at hand. A ML model is *general*, it is valid for new data points the model has never been exposed to.

ML is not a recent research field. One of the first steps in this direction is considered to be taken by Thomas Bayes [11] with "An Essay towards solving a Problem in the Doctrine of Chances" on 1763, where he stated the bases for Bayesian Statistics, a statistical theory in which the evidence about the real state of the world is expressed in terms of degree of belief. Scattered advances were performed on the early 18th century as the prolific mathematician Adrien-Marie Legendre publishes the Least Squares method on 1805 and Pierre-Simon Laplace formalizes what is known today as Baye's Theorem on 1812. No more significant progress would be done until mid 19th century.

The context of the mid-19th century contemplates the beginning of the Digital Revolution⁴ that was triggered by invention of the transistor on 1947 by John Bardeen, Walter Brattain, and William Shockley at AT&T's Bell Labs in the US. This invention will allow the miniaturization of contemporary computers, which relayed on vacuum tubes and solving two of its biggest problems: Its size and its power consumption. In a few years a computer passed from using a whole room's space to fitting in a suitcase, then to fit on a desk, to be the size of a notebook and so on. The miniaturization process mostly followed a linear pattern defined on 1965 by Gordon Moore⁵ Current computers have transistors in its architecture whose size is of 7nm.

The miniaturization of computers carried an exponential growth in computational power that allowed ML models to be applicable, and in the later decades to become mainstream with open source projects as Scikit-learn, TensorFlow and PyTorch.

The Digital Revolution gave another vital asset required for applying ML algorithms: Data. As DOMO's reports on Data usage details [12], [13] 2.5 quintillion of bytes were generated on every minute of the year 2017. During this same year, 90% of all the Data humankind has ever produced was generated. It is estimated that on year 2020 a new 1.7MB will be generated every second *per person living*.

It was because of this joint context of easily available computational power, and colossal quantities of Data that drove the Machine Learning explosion at the beginning of 21st century. Soon, ML models outmatched humans in

¹This is what is called natural intelligence.

²It is considered best practice to preprocess the data before feeding it to a learning algorithm to expedite the learning process. In occasions it is mandatory to preprocess the data as learning would be impossible otherwise.

³AlphaGo is an AI developed by Google's DeepMind to play the board game Go. Its the first AI to beat a Go's professional player without the use of handicaps on a 19x19 board.

⁴Also called as the Third Industrial Revolution.

⁵Moore proposed in 1965 that the number of transistors in a micro-processor doubles each year, he would later correct it to defend that said doubling will happen every two years. For over 50 years this law has prevailed.

tasks as song recognition [14], handwriting recognition⁶, face recognition [15], etc.

In this paper stress measurements during reliability test are recorded by an AU capable to send the data to a centralized data unit through a WiFi connection. Data stored in the Raspberry Pi is then sent to a server where data is processed. The data is transformed by removing the outliers, filtering, labeling and scaling. A ML regression technique is then applied to create a degradation model by feeding a complete till full delamination data from a single test vehicle. The model is then used to assess the degradation state of other test vehicles.

2. Experiment

Accelerated reliability testing are used to stress the Test Vehicles (TV). In this study air thermal shock chamber was used, which consists in two separate chambers with constant temperature, one at 150° and the other one at -40° as shown in Figure 1. The basket containing the TVs is moving up and down between chambers. The transition time between chambers is relatively small making the experiment suitable to accelerate the degradation of the electronic packages.

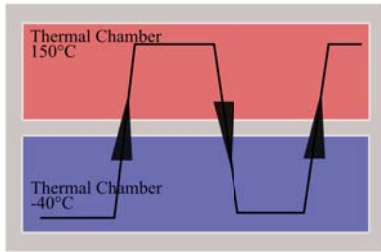


Figure 1: Thermal Chamber procedure description.

The dwelling time was predetermined to provide a condition where all components reach the uniform distribution at target temperatures.

A. Test Vehicle

Thin Quad Flat Packages (TQFP) 100x100 pins with encapsulated piezoresistive silicon based stress sensor are mounted on a PCB. As shown in Figure 2 the functional die is replaced by the stress sensor die.

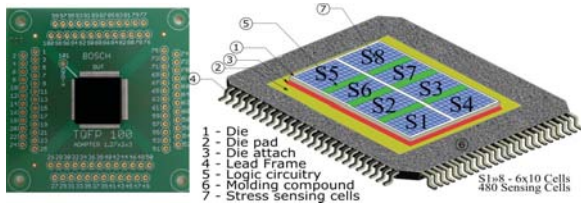


Figure 2: TQFP Mounted on a PCB Test Vehicle.

The package silicon die consists of 8 sensors, with 60 stress measuring cells each, having a total of 480 stress sensing cells. The packages were specially designed to have low adhesion strength between the leadframe and

⁶Software that solve this task are called OCR, that stands for Optical Character Recognition.

the molding compound. One of the way to do this was to use oxidized leadframes in the packaging process. Also, two molding compounds were used for packaging MC1 and MC2. In total six TVs were tested by performing two separate experiments. Two TV are of type 1 molding compound *MC1_1*, *MC1_2* and another four of type 2 molding compound *MC2_1*...*MC2_4*, respectively. In this paper only *MC1_1* sample is used to show the stress difference distribution and the SAM images in case of delamination.

B. Stress Evaluation

In TQFP stress sensors are encapsulated as a regular die to record the mechanical stresses during reliability tests. In this paper TQFP contains 8 sensors with each 6 by 10 stress sensing cells. To evaluate the stresses the following formulas have been used.

The relationship between the measured currents and the stresses are :

$$D(\sigma) = \sigma_{xx} - \sigma_{yy} = \frac{1}{\pi_{44}^p} \frac{I_{OUT} - I_{IN}}{I_{OUT} + I_{IN}} \quad (1)$$

$$\sigma_{xy} = \frac{1}{\pi_{11}^n - \pi_{12}^n} \frac{I_{OUT} - I_{IN}}{I_{OUT} + I_{IN}} \quad (2)$$

where $\pi_{11}, \pi_{12}, \pi_{44}$ are the piezoresistive coefficients of silicon; and I_{IN} , I_{OUT} are the currents measured at the input and output of the sensor, respectively. More details can be found in Ref. [16], [17]

After the experiments were performed, data was processed and only one measurement point per cycle is extracted at the dwell time. Then the stress values at -40° are extracted from the values at 150° as follows:

$$D(\sigma)_{ij}^{rel} = (\sigma_{xx} - \sigma_{yy})^{-40^\circ C} - (\sigma_{xx} - \sigma_{yy})^{150^\circ C} \quad (3)$$

,where $i = 1, \dots, n$ is the number of measurement points, $j = 1, \dots, 480$ is the number of sensing cells and $D(\sigma)_{ij}$ are the relative stress difference.

C. Acquisition Unit

A dedicated acquisition unit was used to power, steer and acquire data from the stress sensor. Additional improvements has been made to the AU to facilitate efficient experiments. In terms of dimension, and weight, the former AU is consisted by three board, it is 110 x 66 x 45 mm in size. The new AU has only one board, its size is 90 x 71 x 20 mm. Shown in Figure 3.

In terms of speed, the former acquisition unit consumes several minutes to collect data from 480 cells (8 sensor x 60 cells/sensor), at 12 bit accuracy. The new acquisition unit has two ADCs, ADS1115 and ADS1015, by changing I2C slave address value in I2C library, one can choose between two ADC. When ADS1115 chosen, we can get

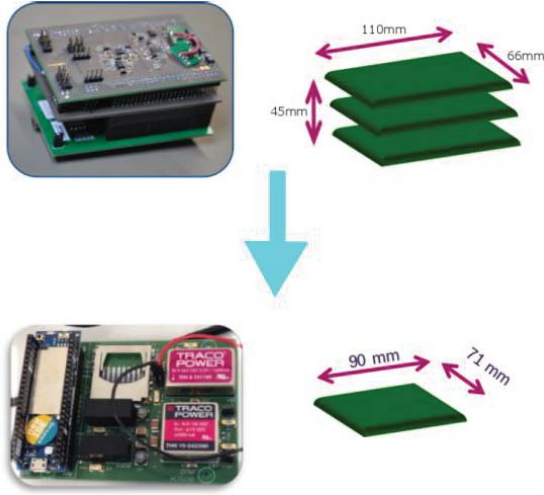


Figure 3: Acquisition unit downsize.

16 bit accuracy, but it requires more time to go through all 480 cells; When choose ADS1015, we still have 12 bit accuracy data, but faster measurements. As for other functions, the new one has built-in WIFI module that facilitates the connection with Raspberry Pi, and the wireless data transmission through WIFI is realized; Pi was also added, because stress value calculation method, stress prediction algorithms based on Neural Network (NN), are all coded in Python and can run on Raspberry Pi. That is to say, Pi is a microcomputer which can afford all the functions in this task, and it is of course lighter, cheaper, more convenient, and mobile than a PC. Meanwhile, real time data processing becomes possible, because WIFI connection can realize real time data transmission between Pi and Arduino.

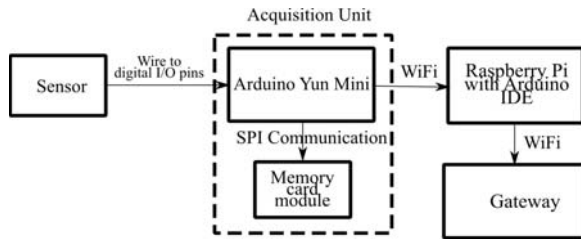


Figure 4: Data flow.

By realizing wireless WIFI connection between Arduino and Pi (see Figure 4), we can imagine that Pi can communicate with multiple Arduinos at same time. Pi has the ability to generate its own network as a hot spot, and Arduinos can have access to the hot spot, thus build the grid that multiple Arduino transmit data to a common central process Pi. So we can realize the in-situ stress monitoring on multiple sensors and AU by one Pi.

D. Delamination validation through SAM images

SAM image of the sample was recorded at 0 cycles, 1170 Cycles and at the end. This can help in correlating between the stress difference values and delamination. In Figure 5 SAM images of the TQFP package targeting

the interface between copper pad/molding compound, die attach/copper pad and die/die attach are shown. One image was performed at the beginning of the test, showing no initial delamination. An intermediate picture at 1170 Cycles, where initial delamination is detected, was performed. The third picture was performed after 2500 temperature cycles and more than 80% of delamination is found.

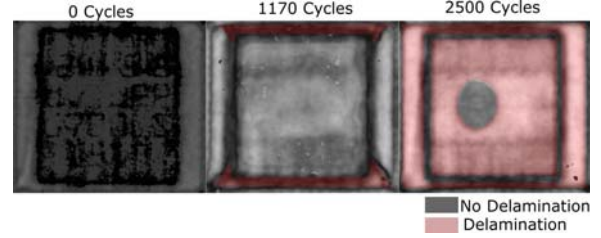


Figure 5: MC1_1 TV SAM image before and after the reliability test. The delamination area is shown in red color.

E. Data from Thermal Cycling Data

The in-plane mechanical stresses were recorded during the entire test for all the 480 measuring cells. The AUs were placed outside the chamber with a wire connection with the samples inside. For visualization purposes average relative stress changes are calculated by averaging over 480 cells as follows:

$$D(\sigma)_j^{average} = \sum_{i=1}^{480} D(\sigma)_{ij}^{rel} \quad (4)$$

The influence of delamination in the die attach over the stress difference on top of the die is shown in Figure 6. An average value of stress per cycle is depicted by using equation 4. The first observation is that the stress values completely change after 900 Cycle reaching a maximum change at 1300 Cycle. This behavior due to the delamination is confirmed by the SAM image taken at 1170 cycle. At this stage it is fair to assume that the sensor is able to detect the structural damage given by the delamination. For a better understanding of the stress signal a FEM model is constructed.

The geometry and the loading conditions are the same as in the experiment. First a component level TQFP is used to establish the agreement between experiments and simulation. This is also a reference for how the stress distribution looks like in the healthy samples.

The values of relative stress along the x and y direction on top of the die and the stress difference calculation for the TQFP component alone from the simulation are shown in Figure 7. This can give us a better understanding of what the sensor can measure, which is the stress difference.

The modelling predictions are compared with the experimental data in Figure 8. The results show good agreement. The deviations are attributed to the uncertainties of the stress sensor, geometry imperfections (variations) and

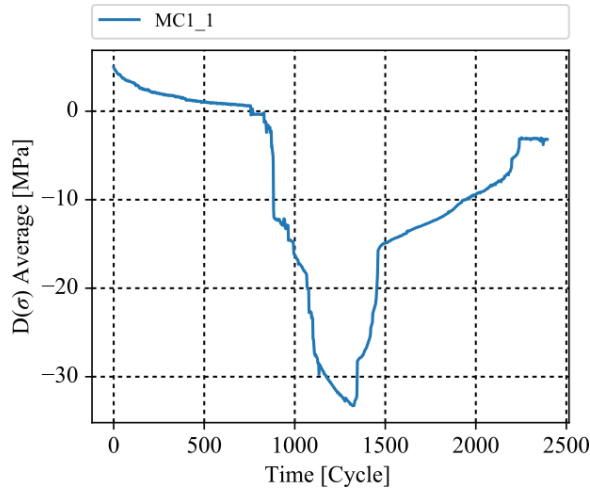


Figure 6: Normal in-plane relative stress difference average along all 480 Cells for MC1_1 Sample.

the material properties used in the simulation. It is worth mentioning that the minimum and maximum values of stress difference are located near the edges. This can give us later an indication of where the delamination is located.

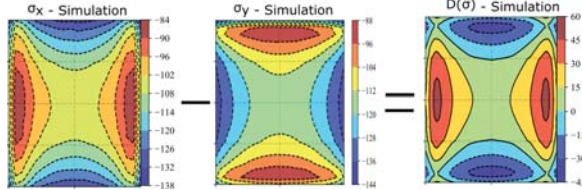


Figure 7: TQFP component level mechanical stress at the top of the die. Loading condition used in the simulation is an environmental temperature of -40° and 150° . The stress values at -40° are extracted from 150° stress values.

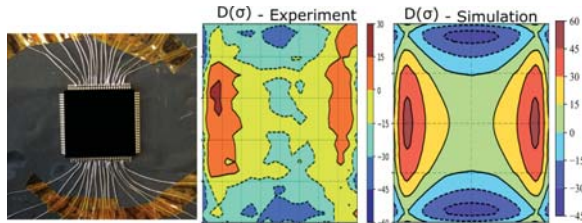


Figure 8: TQFP component level mechanical stress at the top of the die experiment vs. simulation. Loading condition used in the simulation and experiment is an environmental temperature of -40° and 150° . The stress values at -40° are extracted from 150° stress values.

One way to get the stress values along x and y direction when the delamination is present, is depicted in Figure 9. This visualization has the purpose to show the link between delamination, the stresses along x and y direction and the stress difference. Also, this is an efficient indication that the stress difference values are able to capture the delamination.

The same amount of contact area shown in the SAM image (Figure 5) is used for modeling the delamination in the simulation. In the area where delamination is

considered, the interface mesh node is divided in two mesh node with no connection. In the other areas node to node connectivity is maintained.

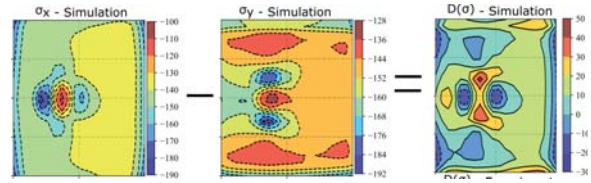


Figure 9: Simulation mechanical stress at the top of the die of the delaminated MC1_1 TV. Loading condition used in the simulation is an environmental temperature of -40° and 150° . The stress values at -40° are extracted from 150° stress values.

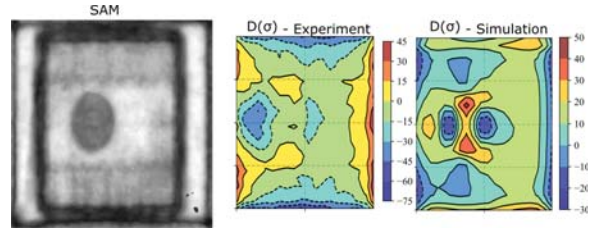


Figure 10: Simulation and experiment mechanical stress at the top of the die of the delaminated MC1_1 TV. Loading condition used in the simulation is an environmental temperature of -40° and 150° . The stress values at -40° are extracted from 150° stress values.

Figure 10 shows the contour plot of the relative stress difference from both simulation and experiment with the designated delamination area shown in the SAM image. Even in this case the agreement is quite good between the experiment and the simulation in the undelaminated area, despite the limitation of the method used in case of simulating the delamination area. Both plots show similar stress distribution where the undelaminated area is present. The distribution near the undelaminated area is similar as in the case of Figure 8. Overlay pictures between the stress difference and SAM image of both simulation and experiment are shown in Figure 11. The top/bottom and left/right stress distribution of maximum and minimum values are exactly on top of the edge of the undelaminated area.

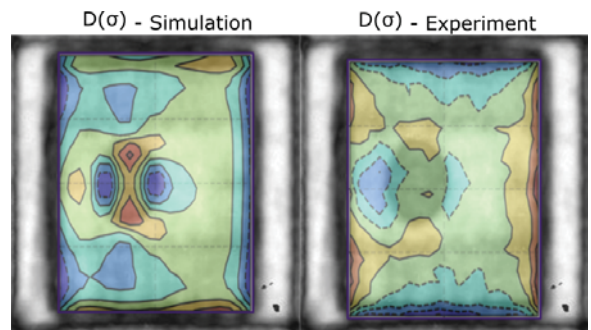


Figure 11: MC1_1 TV overlay picture of stress difference on top of SAM image. Both simulation and experiment gives a very good indication where there is still contact underneath the die.

3. Degradation prediction

The amount of cells provides a very good resolution of the stress difference distribution over the surface of the silicon die. However, to predict the delamination by just looking at the stress difference is not trivial. Machine learning is used to cope with this problem. In this paper, as an initial step a model is trained with the assumption that the delamination happens linearly. The advantage of this method is that it can provide a fast prediction degradation of the TQFP. Relative stress difference average of 6 samples during experiments are shown in Figure 12. In case of delamination all of them shows a similar stress difference trend. The MC1_1 TV is used to train the machine learning model, due to the nature of the data. It contains stress behavior from the undelaminated state, until fully delaminated state. We consider a fully delaminated state, when the delamination does not further propagate.

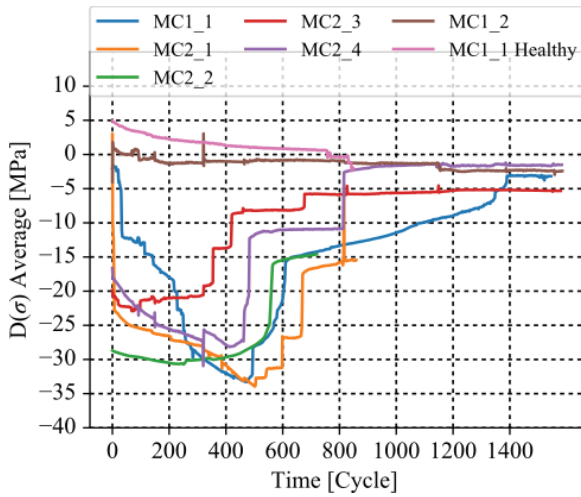


Figure 12: Normal in-plane stress difference average along all 480 Cells for all samples.

A. Back Propagation Artificial Neural Network

Nowadays there is a set of techniques that shines for their performance called Artificial Neural Networks (ANN). A simple representation of a Feed-Forward Artificial Neural Network can be seen on Figure 13. By themselves ANN are not a method, but a tool to enable the collaboration between methods towards a common goal. An ANN is a collection of Artificial Neurons, therefore it is necessary to review what is it and what it does a single Artificial Neuron (AN).

ANs have been inspired in biology in the way it keeps a biological structure aiming to replicate a biological function⁷ On Figure 14 it can be seen a comparison

⁷This is a common practice in Engineering: In the same fashion the wing of a plane preserves its general structure, replicating a specific function of a biological wing. ANs keep the structure of biological neuron and aims to loosely replicate the biological function of a biological neuron.

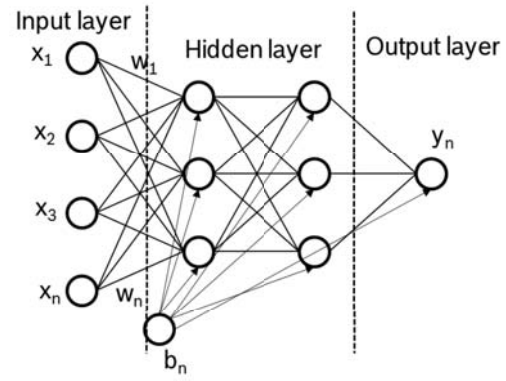
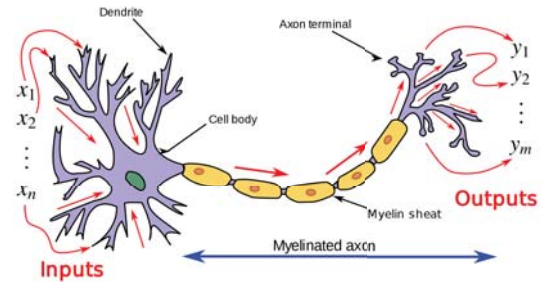
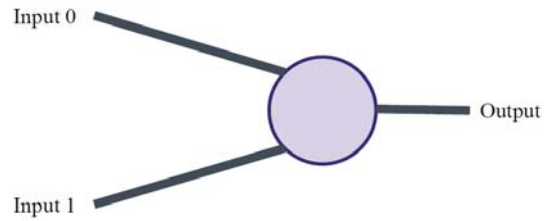


Figure 13: Visual representation of an Artificial Neural Network.



(a) Biological neuron [18].



(b) Artificial Neuron.

Figure 14: Comparison between biological and artificial neurons.

between a biological neuron (14a) and an artificial neuron (14b).

The most basic model that can be used on an AN is a Perceptron. This linear model was invented by Frank Rosenblatt in 1958 [19].

Neural Networks consist of the following components:

- An input layer, x
- An output layer, y
- A set of weights and biases between each layer, W and b
- A choice of activation function for each hidden layer, σ

Having a set of data points organized in the matrix $\mathbf{X}$ that is formed by individual data points vectors $\mathbf{x}$. The model comprises a mathematical function as in equation

5.

$$y(x) = f(w \times x) + b \quad (5)$$

This means, if we know weight matrix w and bias vector b , under the chosen activation function f , we can predict unknown output value y , by known input vector x . Input vector x is always known, so how to get weight matrix w , and the bias vector b becomes the key problem, which is exactly the purpose of ANN training purpose.

The following visualization of the working of a perceptron will be performed with the help of Figure 17.

A perceptron can have an arbitrary number of inputs and a unique output.

On Figure 17 (a) a perceptron is represented as a circle with two inputs and one output⁸.

- The inputs will be the values of the features of a data point. For example, data point [2, 7] will be assigned as in Figure 17 (b).
- Each dendrite of the perceptron has an assigned "Weight" as shown on Figure 17 (c).
- The inputs are multiplied by the weight of its dendrite as in Figure 17 (d).
- The summation of the values coming from the dendrites is passed through an activation function and its result given as output as in Figure 17 (e).

Most used activation functions are step, sign, sigmoid⁹, tanh¹⁰ and ReLU¹¹:

$$step(z) = \begin{cases} 1 & \text{if } z \geq 0; \\ 0 & \text{if } z < 0. \end{cases} \quad (6)$$

$$sign(z) = \begin{cases} 1 & \text{if } z \geq 0; \\ -1 & \text{if } z < 0. \end{cases} \quad (7)$$

$$sigmoid(z) = \frac{1}{1 + e^{-z}} \quad (8)$$

$$tanh(z) = \frac{e^z - 1}{e^z + 1} \quad (9)$$

$$ReLU(z) = \max(0, z) \quad (10)$$

$$identity(z) = z \quad (11)$$

The selection of the activation function depends on what type of behavior the perceptron is designed to abstract from the data that will feed it. Like this, a perceptron with a sign activation function would be used for linear classification tasks, a perceptron with an identity activation function would be used for linear regression and a

⁸The resemblance of a Perceptron to a biological neuron can be noticed as the artificial dendrites mirrors biological ones and the output an axon.

⁹Also called logistic function.

¹⁰Hyperbolic tangent function.

¹¹Rectified linear unit.

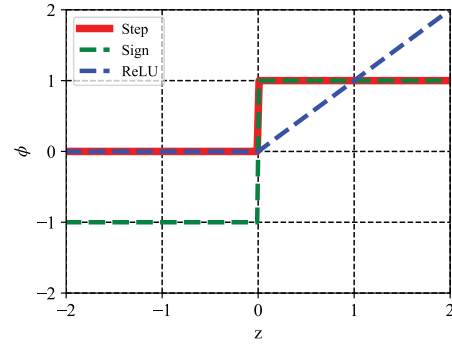


Figure 15: z vs ϕ plot for Step, Sign and ReLU functions.

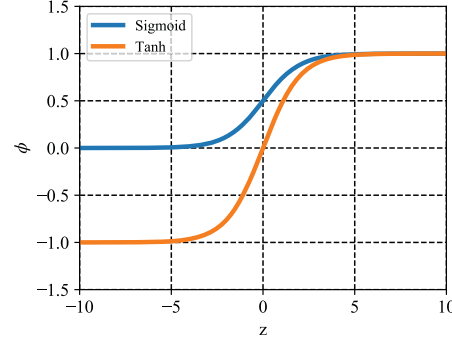


Figure 16: z vs ϕ plot for Sigmoid and Tanh functions.

perceptron with a sigmoid activation function would be used for logistic regression.

Backpropagation (BP) is a method used in artificial neural networks to calculate the error contribution of each neuron after a batch is processed. This is used by an enveloping optimization algorithm to adjust the weight of each neuron, completing the learning process for that case. We predict the input part of training data by the current neural network, this is usually called forward propagation. And we get a set of output data, which has differences with the output part of the training data, this is error. In optimization algorithm, this error is named

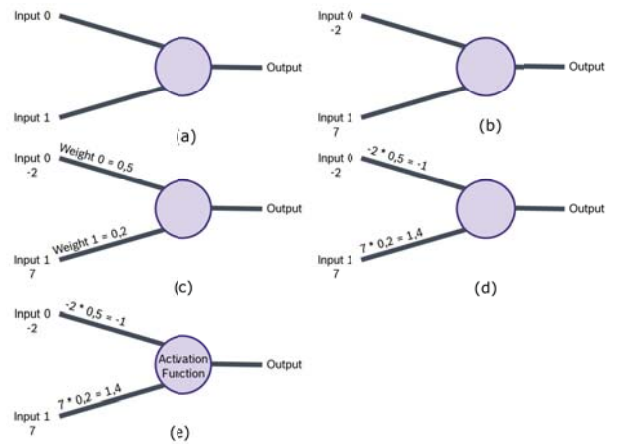


Figure 17: Example of perceptron learning algorithm.

loss function, our goal is to optimize the parameters of the network to make the loss function reach its minimum point. Which means the neural network fit the training data and can represent the real model. Technically it calculates the gradient of the loss function to reach the optimization goal. It is commonly used in the gradient descent optimization algorithm. It is also called backward propagation of errors, because the error is calculated at the output and distributed back through the network layers to adjust all the parameters inside the network. This BP method has limitations, because it is gradient based optimization method, so BP is not guaranteed to find the global minimum of the loss function, maybe just a local minimum. This can be solved by making some improvements, such as:

- add momentum factor to make learning rate adaptive
- training more times, then we may get the global minimum at larger probability
- combining other optimization algorithms into BP, for example Particle Swarm Optimization, Genetic algorithm, etc.

Learning rate can be regarded as improvement step, represent how far we take the next step towards the negative gradient direction. If this value is big, we only need several steps (iterations) to approach the minimum point, which means faster convergence and time saving. There are many rule-of-thumb methods for determining an acceptable number of neurons to use in the hidden layers, such as the following: the number of hidden neurons should be between the size of the input layer and the size of the output layer. Momentum factor it can be regarded as an adjustment of the learning rate, to make the step length no longer fixed, thus can realize large steps at beginning to make loss function drop fast, and shrink the step when approaching the minimum.

Table 1: Chosen parameters of BP NN

Parameter	Value
Number of hidden neurons	100
Learning Rate	0.001
Iterations	2200
Momentum	0.31

For our application the input data contains the relative stress difference from all the 480 cells during 2400 temperature cycles. The output data contains one vector with values between 100 and 0 based on the assumption that the delamination degradation happens linearly and it start when the stresses are degrading. In Figure 18 the details of such vector is shown, but also the testing data.

The model is trained based on minimizing the loss function and the optimized neural network parameters are shown in Table 1.

The stress difference data from the other 5 samples are tested with the neural network model and the predictions

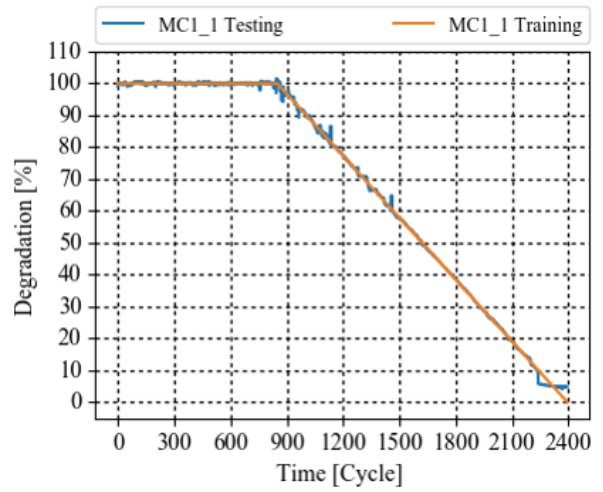


Figure 18: Training output and testing data.

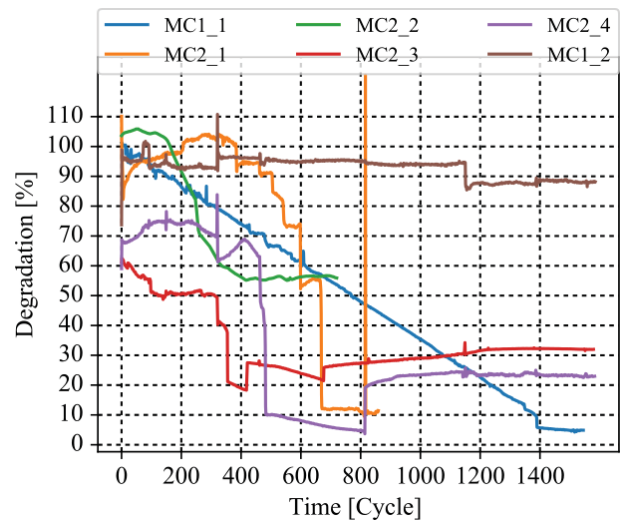


Figure 19: Degradation Prediction of the BP NN Model .

are depicted in Figure 19. This percentage degradation model is a fast tool to evaluate the package delamination status. Some of the packages indicates an apriori delamination for examples *MC2_3*, *MC2_4* and others are showing that the delamination is yet to happen, for example *MC2_2*.

Further work is required to improve the accuracy of the neural network model, by feeding more data for training and also using more efficient neural networks methods.

4. Conclusions

In this paper, a degradation model based on in-plane stress measurements is proposed. Mechanical stresses are able to capture structural change in the packages including delamination. The BP NN model shows promising results in a fast and automated way to estimate the delamination inside the package. This non-intrusive method can be used in testing new package designs, which can lead in

better and fast design. Also, has the potential to be used in Prognostics and Health Management for automotive electronics. Future work should be focused in acquiring more testing data for different designs and implementing more efficient ML methods.

5. Acknowledgements

Authors would like to show the gratitude to Silber Christian, Broermann Katrin from EIM4 department and to STMicroelectronics for supporting this study.

References

- [1] Seyedmohsen Hosseini, Kash Barker, and Jose E. Ramirez-Marquez. A review of definitions and measures of system resilience. *Reliability Engineering & System Safety*, 145:47 – 61, 2016.
- [2] 24765-2017 - ISO/IEC/IEEE International Standard - Systems and software engineering—Vocabulary - IEEE Standard.
- [3] Claus Ballegaard Nielsen, Peter Gorm Larsen, John Fitzgerald, Jim Woodcock, and Jan Peleska. Systems of systems engineering: Basic concepts, model-based techniques, and research directions. *ACM Comput. Surv.*, 48(2):18:1–18:41, September 2015.
- [4] W. Denson. The history of reliability prediction. 47(3):SP321–SP328.
- [5] Dana Crowe and Alec Feinberg, editors. *Design for Reliability*, volume 7 of *Electronics Handbook Series*. CRC Press, April 2001. DOI: 10.1201/9781420040845.
- [6] M. Xie and C.D. Lai. Reliability analysis using an additive weibull model with bathtub-shaped failure rate function. *Reliability Engineering & System Safety*, 52(1):87 – 93, 1996.
- [7] Hideo Tanaka, L. T. Fan, F. S. Lai, and K. Toguchi. Fault-tree analysis by fuzzy probability. *R-32(5)*:453–457.
- [8] Spotlight on automotive, pwc semiconductor report. Accessed: 2019-02-04.
- [9] DeepMind. Alphago, 2010. [Online; accessed 6th of January, 2019].
- [10] Andreas C. Mueller and Sarah Guido. *Introduction to machine learning with Python: a guide for data scientists*. O'Reilly Media, Inc, first edition edition. OCLC: ocn895728667.
- [11] T. Bayes. An essay towards solving a problem in the doctrine of chances. *Phil. Trans. of the Royal Soc. of London*, 53:370–418, 1763.
- [12] DOMO. Data never sleeps 5.0, 2018. [Online; accessed 6th of January, 2019].
- [13] DOMO. Data never sleeps 6.0, 2019. [Online; accessed 6th of January, 2019].
- [14] Avery Wang. An industrial strength audio search algorithm. In *ISMIR*, 2003.
- [15] Yaniv Taigman, Ming Yang, Marc’Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [16] Alexandru Prisacaru, Alicja Palczynska, Przemyslaw Jakub Gromala, Bongtae Han, and Guo Qi Zhang. Condition monitoring algorithm for piezoresistive silicon-based stress sensor data obtained from electronic control units. *2017 IEEE 67th Electronic Components and Technology Conference (ECTC)*, pages 1119–1127, 2017.
- [17] Alicja Palczynska, Alexandru Prisacaru, Przemyslaw Jakub Gromala, Bongtae Han, Dirk Mayer, and Tobias Melz. Towards prognostics and health monitoring: The potential of fault detection by piezoresistive silicon stress sensor. In *2016 17th International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE)*, pages 1–8. IEEE.
- [18] Prof. Loc Vu-Quoc. Neuron3, 2018. [Online; accessed 6th of January, 2019].
- [19] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, pages 65–386, 1958.