

D.K. MARUKHYAN

THE PHYSICAL DESIGN METHOD FOR INTEGRATED CIRCUITS USING ARTIFICIAL INTELLIGENCE

The technological advancement of integrated circuits (ICs) accompanied by an unprecedented reduction in component sizes to a few nanometers and an increase in transistor counts to over 100 billion has introduced new challenges related to circuit reliability. In these conditions, the interaction between components, or "crosstalk," becomes a key issue, arising from parasitic capacitances and inductances between adjacent wires. Crosstalk can lead to serious failures, including signal timing parameter violations, logical errors, and increased power consumption.

To address this problem, a novel approach utilizing artificial intelligence (AI) and machine learning (ML) algorithms was developed to predict and mitigate crosstalk effects. The application of this method achieved a crosstalk reduction of nearly 18%, with setup timing improved by 17% and hold timing improved by almost 20%. These gains were realized with acceptable trade-offs, including a 3% increase in area, a 5.5% increase in power consumption, and a 17% increase in CTS wire length, which resulted in only a 0.8% increase in the total wire length of the circuit.

Keywords: crosstalk, artificial intelligence, integrated circuits, modeling, design method, machine learning.

Introduction. Crosstalk arises from the formation of parasitic capacitances and inductances between closely spaced wires in a circuit. When a signal passing through one wire (the "aggressor") changes, it affects the neighboring wire (the "victim") through the electromagnetic field [1]. The effect caused by the mutual capacitance (C_m) is described by formula (1) [1-3], and its impedance is described by formula (2) [1,3,4]:

$$I_{\text{crosstalk},C_m} = C_m * \frac{dV_{\text{agg}}}{dt}, \quad (1)$$

$$X_{C_m} = \frac{1}{2\pi f C_m}. \quad (2)$$

As seen from the formulas, as the frequency (f) increases and the distance (d) between the wires decreases (which increases C_m according to formula (3)), the capacitive reactance drops sharply, leading to stronger interaction [2, 4]:

$$C_m = \frac{S}{d}. \quad (3)$$

The consequences of crosstalk are varied and can critically affect circuit operation, leading to timing violations, logical errors, increased power consumption, and accelerated circuit aging. These problems demonstrate the urgent need to develop effective methods for detecting and mitigating crosstalk. This paper proposes an innovative approach that uses AI and ML tools to not only quickly predict critical points of crosstalk but also to offer solutions for their mitigation.

The existing crosstalk mitigation techniques and their limitations. Over years, various methods have been developed to combat crosstalk at different stages of IC design, but none fully meet modern technological requirements.

Bus encoding reduces simultaneous transitions on adjacent wires but increases area, power, and delay while being effective only in specific cases. Differential signaling cancels common-mode noise but doubles wire count and power usage. Repeater insertion restores the signal quality and reduces overlap but adds area, power, and delay. Shielding lowers interference but requires extra layout space and careful design. Routing blockage increases wire spacing but consumes routing resources and complicates the design.

A method proposed for data collection and preparation for crosstalk prediction. Unlike traditional approaches that require multiple analyses, the proposed method predicts crosstalk using minimal yet highly informative data. It aggregates physical and logical design information to build a feature-rich database for the ML model.

As illustrated in Fig. 1, the process begins by extracting physical metal parameters from the GDSII file followed by the extraction of signal properties from the .nlib file [5]. Signal integrity analysis is then performed to calculate coupling coefficients, which measure how changes in one signal affect another. Finally, all extracted data is consolidated into a unified database containing key physical and logical features. This comprehensive database serves as the foundation for training and applying the ML-based crosstalk prediction model.

Fig. 1 shows a flowchart illustrating the data aggregation process. Inputs from GDSII, NLIB, and auxiliary info files are processed to extract the physical metal characteristics and logical signal properties. These are combined to create a final feature database for the model.

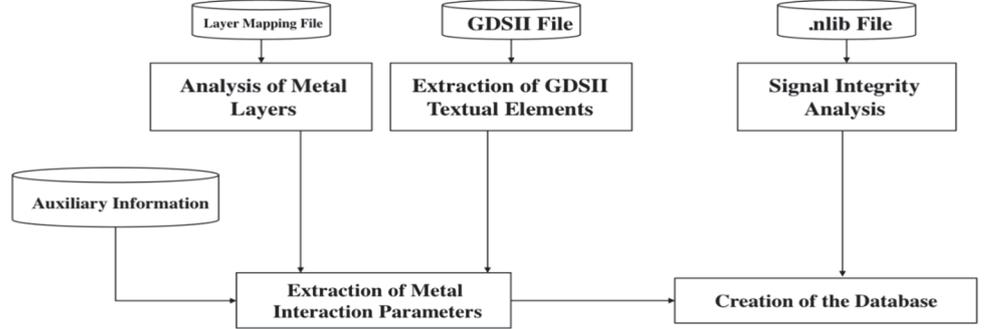


Fig. 1. A flowchart illustrating the data aggregation process

A method proposed for architecture and training of the machine learning model. After data collection, the most critical part of the method is implemented: the creation and training of the machine learning model:

- Final data preparation. The database undergoes final processing, including encoding categorical data into numerical values, removing duplicate and constant data, and performing data normalization (min-max scaling) to bring all feature values into the $[0, 1]$ range. This is done using the following formula:

$$X_{normalized} = \frac{x - x_{min}}{x_{max} - x_{min}}. \quad (4)$$

- Selection of ML Model Architecture. A Branched Artificial Neural Network (BANN) architecture was chosen.

- Activation Functions. Since all input data is normalized to the $[0, 1]$ range, corresponding activation functions were chosen. A variant of the ReLU function, described by formula (5), is used in the hidden layers:

$$f(x) = \begin{cases} 0, & \text{if } x \leq 0, \\ x, & \text{if } x > 0. \end{cases} \quad (5)$$

- The Sigmoid function described by formula (6), is used in the output layer, as its output is always in the $(0, 1)$ range, which is ideal for the normalized target value:

$$f(x) = \frac{1}{1 + e^{-x}}. \quad (6)$$

- Model Training and Application. The model was trained using the TensorFlow library and its Keras API.

Fig. 2 shows a diagram of the Branched Artificial Neural Network architecture. An input layer feeds into two separate parallel branches of hidden layers. The

outputs of these branches are then concatenated and passed through a final layer to produce the predicted crosstalk value.

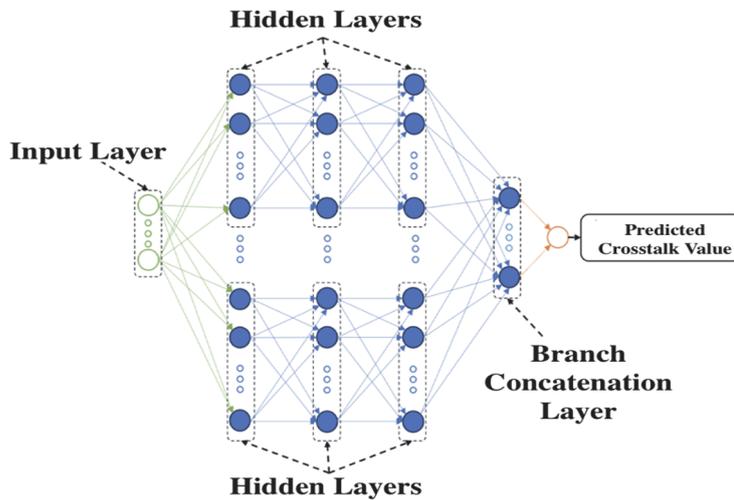


Fig.2. A diagram of the Branched Artificial Neural Network architecture

Fig. 3 shows a flowchart of the machine learning model workflow. It starts with the formatted database, proceeds to data preprocessing and normalization, then to the construction and training of the ML model followed by testing, evaluation, and finally, saving the trained model.

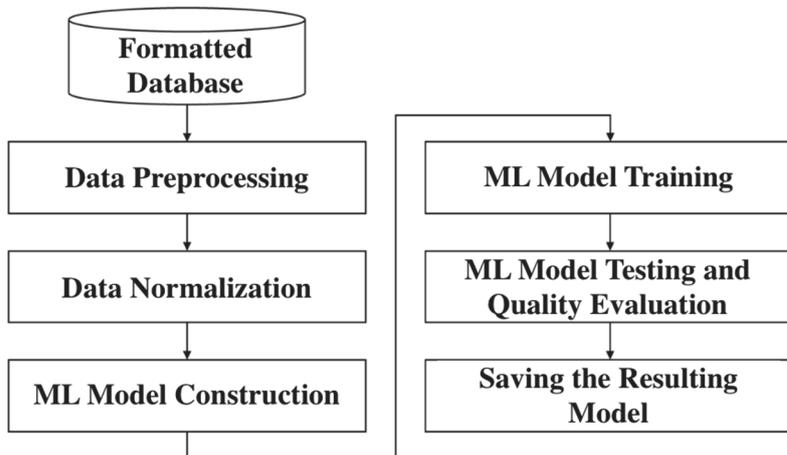


Fig. 3. A flowchart of the machine learning model workflow

The performance evaluation of the artificial intelligence model. Experimental studies were conducted to verify the effectiveness and applicability of the proposed ML model:

- Experimental environment. Circuits composed of cells from the ASAP7 educational library for the 14nm technology process were used. A complex circuit with 22,826 standard cells and an input clock frequency of 2 GHz was used for training and evaluation. The traditional SIA method in Fusion Compiler took 8227.7 seconds for this circuit, which served as the baseline for comparison.

- Evaluation metrics. The Mean Squared Error (MSE) described by formula (7), was chosen as the loss function to measure the accuracy of the model:

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2. \quad (7)$$

- Results and analysis. The model was trained for approximately 48-53 epochs, achieving a prediction accuracy of about 91%. Most importantly, the analysis time was significantly reduced. The proposed ML model completed the prediction in 5216.4 seconds. This represents a time saving of 36.6%. While this time saving comes at the cost of a minor (~10%) loss in accuracy, this level of precision is often sufficient for identifying critical areas in the early stages of design.

A method proposed for crosstalk mitigation for the clock tree. After developing a predictive model successfully, the next step is to mitigate the effects of crosstalk. The proposed approach involves making targeted changes to the most sensitive and aggressive part of the circuit: the clock tree (CTS). The CTS is targeted because its wires carry the highest frequency signals in the circuit, making them the "main" aggressors, and they are highly sensitive to timing deviations that can cause critical setup and hold violations. An improved combined method of routing blockages and shielding is proposed.

1. Target pair identification: Using the ML model's predictions, all metal pairs where at least one wire belongs to the CTS are identified.

2. Condition check: For each pair, the distance between wires is compared to the minimum allowable distance (d_{min}) defined by the design rules for that metal layer.

3. Decision making:

- If the distance is less than $3 * d_{min}$, the wires are considered too close. In this case, routing blockages are applied, instructing the design tool to re-route the wires to increase the spacing between them.

- If the distance is greater than or equal to $3 * d_{min}$, there is sufficient space between the wires. Instead of increasing the distance further, the shielding method is applied by inserting a protective wire connected to ground between the aggressor and victim.

Evaluation of results. Applying the proposed mitigation method to a test circuit yielded the results summarized in Tables 1 and 2.

Table 1

Timing and Crosstalk Violation Improvements

Parameter	Initial IC	With Shielding	With Blockage	With the Proposed Method
Components with Setup Violation	940	885	901	854
Total Setup Violation (ps)	-1128.3	-1031.2	-1052.4	-928.6
Components with Hold Violation	605	571	579	538
Total Hold Violation (ps)	-415.2	-370.1	-386.8	-329.7
Components with Crosstalk	482	369	395	307
Crosstalk Violation (ps)	-11.03	-10.11	-9.96	-8.88

Table 2

Physical Design and Power Costs

Parameter	Initial IC	With Shielding	With Blockage	With the Proposed Method
Area (μm^2)	9150.2	9150.2	9514.8	9431.54
CTS Wire Length (μm)	7790.4	7790.4	9895.1	9110.8
Total Wire Length (μm)	86470.3	86835.1	87640.7	87190.6
Total Power (μW)	1742.1	1742.1	1889.7	1837.5
Leakage Power (μW)	92.4	92.4	93.5	92.7

- Improvements:

- Crosstalk reduction: 18.1%.
- Total Negative Slack (Setup) improvement: 17.7%.
- Total Negative Slack (Hold) improvement: 20.5%.

- Costs:

- Area increase: 3.07%.
- Power consumption increase: 5.5%.
- CTS wire length increase: 17.1%, which contributed to only a 0.83% increase in the total wire length of the circuit.

Conclusion. This work presents a comprehensive solution for managing crosstalk in IC design. It demonstrates that the application of artificial intelligence can not only automate and accelerate complex analyses but also offer more intelligent and targeted solutions than the traditional approaches are. The developed methods provide designers with a powerful tool to create higher-performance, more reliable, and energy-efficient integrated circuits, addressing the contemporary demands of the technology industry.

REFERENCES

1. **Jerse T.** Introduction to Crosstalk // 2017 IEEE International Symposium on Electromagnetic Compatibility & Signal/Power Integrity (EMCSI). - Washington, DC, USA, 2017. - P. 1-45, doi: 10.1109/ISEMC.2017.8078014.
2. **Hall S., Hall G.** High-Speed Digital System Design: A Handbook of Interconnect Theory and Design Practices. - 2000. - 327 p.
3. **Electronics Tutorials.** AC Inductors, Inductive Reactance. - 2021. - Retrieved from <https://www.electronics-tutorials.ws/inductor/ac-inductors.html>
4. **Electronics Tutorials.** Capacitive Reactance. - 2021. - Retrieved from https://www.electronics-tutorials.ws/filter/filter_1.html
5. **Artwork.** Examples of GDSII Files with Different Extents. - 2011. - Retrieved from https://www.artwork.com/gdsii/gds_extents/examples.html

National Polytechnic University of Armenia. The material is received on 08.10.2025.

Ղ.Կ. ՄԱՐՈՒԽՅԱՆ

ԻՆՏԵԳՐԱԼ ՄԻԿՐՍԱԿՆԵՐԻ ԱՐՇԵՍՏԱԿԱՆ ԲԱՆԱԿԱՆՈՒԹՅԱՄԲ ՖԻԶԻԿԱԿԱՆ ՆԱԽԱԳԾՄԱՆ ՄԵԹՈՂ

Վերջին տասնամյակներում ինտեգրված սխեմաների տեխնոլոգիական զարգացումը, որն ուղեկցվում է տարրերի չափերի կտրուկ նվազումով մինչև մի քանի նանոմետր և տրանզիստորների քանակի ավելացմամբ՝ գերազանցելով 100 միլիարդը, առաջացրել է նոր խնդիրներ՝ կապված սխեմաների հուսալիության հետ: Այս պայմաններում տարրերի փոխազդեցությունը, հայտնի որպես «տարրերի միջև փոխազդեցություններ», դառնում է հիմնական խնդիր՝ առաջանալով հարևան հաղորդալարերի միջև պարազիտային ունակություններ և ինդուկտիվություններ: Տարրերի միջև փոխազդեցությունները կարող են առաջացնել լուրջ խափանումներ, ներառյալ ազդանշանների ժամանակային շեղումներ, տրամաբանական սխալներ և էներգիայի սպառման աճ:

Այս խնդրի լուծման համար մշակվել է նորարարական մոտեցում՝ կիրառելով արհեստական բանականության և մեքենայական ուսուցման ալգորիթմներ՝ կանխատեսելու և նվազեցնելու տարրերի միջև փոխազդեցությունները: Մեթոդի կիրառումն ապահովել է խանգարումների մոտավորապես 18% նվազում, ժամանակային setup պարամետրերի 17% և hold պարամետրերի գրեթե 20% բարելավում: Այս արդյունքները ձեռք են բերվել ընդունելի փոխզիջումներով՝ տարածքի 3% աճ, էներգիայի սպառման 5.5% ավելացում և CTS հաղորդալարերի երկարության 17% աճ, որը հանգեցրել է միայն հաղորդալարերի ընդհանուր երկարության 0.8% ավելացման:

Առանցքային բառեր. տարրերի միջև փոխազդեցություններ, արհեստական բանականություն, ինտեգրված սխեմաներ, մոդելավորում, նախագծման մեթոդ, մեքենայական ուսուցում:

Д.К. МАРУХЯН

МЕТОД ФИЗИЧЕСКОГО ПРОЕКТИРОВАНИЯ ИНТЕГРАЛЬНЫХ СХЕМ С ИСПОЛЬЗОВАНИЕМ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

За последние десятилетия технологическое развитие интегральных схем, сопровождаемое резким уменьшением размеров элементов до нескольких нанометров и увеличением числа транзисторов более чем до 100 миллиардов, привело к новым проблемам, связанным с надежностью схем. В таких условиях взаимодействие между элементами, известное как "перекрестные помехи", становится критической проблемой, возникающей из-за паразитных емкостей и индуктивностей между соседними проводниками. Перекрестные помехи могут вызывать серьезные сбои, включая нарушения временных параметров сигналов, логические ошибки и рост энергопотребления.

Для решения этой проблемы был разработан инновационный подход, использующий алгоритмы искусственного интеллекта и машинного обучения для прогнозирования и снижения воздействия перекрестных помех. Применение метода позволило снизить уровень помех почти на 18%, улучшить временные параметры setup на 17% и hold почти на 20%. Эти результаты достигнуты при приемлемых затратах: увеличение площади на 3%, рост энергопотребления на 5,5% и увеличение длины проводов CTS на 17%, что привело лишь к росту общей длины проводов на 0,8%.

Ключевые слова: перекрестные помехи, искусственный интеллект, интегральные схемы, моделирование, метод проектирования, машинное обучение.