



# **EMERGING TRENDS IN ELECTRICAL AND ELECTRONICS ENGINEERING**

**Editor:  
Dr. Mustafa Anil REŞAT**



**EMERGING TRENDS IN  
ELECTRICAL AND  
ELECTRONICS ENGINEERING**

**Editor:**

**Dr. Mustafa Anıl REŞAT**



***Emerging Trends in Electrical and Electronics Engineering***  
***Editor:Dr. Mustafa Anıl REŞAT***

**Editor in chief:** Berkan Balpetek

**Cover and Page Design:** Duvar Design

**Printing :** December-2024

**Publisher Certificate No:** 49837

**ISBN:** 978-625-5530-00-4

© Duvar Yayınları

853 Sokak No:13 P.10 Kemeraltı-Konak/İzmir

Tel: 0 232 484 88 68

[www.duvaryayinlari.com](http://www.duvaryayinlari.com)

[duvarkitabevi@gmail.com](mailto:duvarkitabevi@gmail.com)

## TABLE OF CONTENTS

<b>Chapter 1.....</b>	<b>4</b>
<b>Comparison of K-Nearest Neighbors, Decision Tree and Support Vector Machines Methods in Predicting Environmental Attitudes</b>	
<i>Ali DEĞİRMENCI</i>	
<b>Chapter 2.....</b>	<b>22</b>
<b>A Review on Renewable Energy-Powered Near-Field Wireless Charging Station for Electric Vehicles</b>	
<i>Mehmet Zahid EREL, Yunus YALMAN</i>	
<b>Chapter 3.....</b>	<b>36</b>
<b>Hybrid Transformers: An Overview of Configurations and Converter Topologies</b>	
<i>Yunus YALMAN, Mehmet Zahid EREL</i>	
<b>Chapter 4.....</b>	<b>53</b>
<b>A Flexible and Energy-Conscious Dynamic Encryption Approach For Resource-Constrained Iot and Embedded Devices</b>	
<i>Cemil Baki KIYAK, Fadi YILMAZ, Hasan Şakir BİLGE</i>	

## **Chapter 1**

# **Comparison of K-Nearest Neighbors, Decision Tree and Support Vector Machines Methods in Predicting Environmental Attitudes**

**Ali DEĞİRMENCI<sup>1</sup>**

---

<sup>1</sup> Dr. Arařtırma Grevlisi, Elektrik-Elektronik MhendisliĐi Blm  
ORCID: 0000-0003-4792-1774

## **Abstract**

In recent years, there has been a significant increase in environmental problems, especially those resulting from people's unconscious behavior. There is a high probability that environmental problems will reach irreversible dimensions due to the unconscious behavior of people. As environmental problems increase, social awareness needs to be increased to solve these problems. In this study, people's environmental attitudes were analyzed with three different machine learning methods. Employed methods are k-nearest neighbors, decision tree, and support vector machines (SVM). Exhaustive grid search and k-fold cross validation (k=10) are used to tune method-specific hyperparameters. Experimental results indicate that the highest scores are obtained in the SVM method (accuracy = 0.9576, F1-score = 0.9615, precision = 0.9576, recall = 0.9670). Based on the findings, environmental attitudes of the people can be predicted confidently by machine learning methods.

## 1.1 Introduction

The environment, encompassing natural ecosystems, biodiversity and the resources necessary to sustain life on Earth, is undergoing unprecedented adversity as a result of human activities. In the last century, with the increase in industrialization and easier access to resources, there has been an increase in urbanization and a significant increase in population. These issues not only threaten ecological stability, but also pose far-reaching problems for human welfare, economic systems and global security. To solve these life-threatening problems, the underlying causes and consequences must first be investigated and analyzed in detail, then comprehensive and sustainable solutions must be produced according to the results of these analyses, and this process must become a state policy [1,2].

Individuals also have a great responsibility in solving environmental problems. At this stage, small changes in people's lives will make a significant contribution to reducing environmental problems. Simple measures that individuals can take may include: increasing recycling by sorting waste, using energy-efficient light bulbs, reducing the use of products made from disposable plastics, and saving energy through thermal insulation in buildings. Another example is encouraging the widespread use of electric-based vehicles. Because today, the majority of vehicles used for transportation and personal use still run on petroleum-based fuels, and the combustion of petroleum-based fuels releases harmful gases into the environment. By using fully electric or electric-assisted vehicles, the impact of harmful exhaust gases emitted from vehicles can be significantly reduced [3,4,5].

Machine learning methods are frequently employed in research on environmental problems. Beiser-McGrath and Huber developed a random forest (RF) model to estimate individuals' attitudes toward climate change and environmental protection, utilizing psychological and demographic factors as predictors [6]. The dataset was obtained from field surveys conducted in China, Switzerland and the USA. Choudhury et al. predict consumers' green purchase intentions based on societal and individual factors and categorize them into green and non-green customers [7]. The dataset consisted of responses from 310 people. Six machine learning methods, namely Decision Tree, Gradient Boosting, Random Forest, XGBoost, Support Vector Machine and KNN, are employed to classify consumers as green or non-green consumers. According to the findings obtained from the analysis, the most important factors in predicting green purchasing behavior were green self-identification, environmental consciousness, and environmental knowledge. The evaluation of machine learning methods was analyzed with seven performance metrics (AUC-ROC,

Accuracy, F1 Score, Precision, Recall, Hamming Score, Kappa Score). Although RF showed the best performance in terms of accuracy (0.83) and ROC-AUC (0.84) among the compared methods, the performances of Gradient Boosting, XGBoost and SVM were quite close to the highest score. Wang et al. employed modified C5.0 decision-tree to estimate pro-environmental behavior of university students [8]. The dataset consists of questionnaire responses from 334 university students in Guangdong Province, China. The accuracy of the modified C5.0 decision tree algorithm on training data was 73.13% and on test data was 69.16%. Lou et al. used a random forest method to classify priority of environmental protection over economic growth [9]. The dataset consists of macroeconomic, demographic and psychological predictors of 51,348 participants from 47 countries. According to the results, an out-of-bag error rate of 34.15% was obtained. Li et al. utilized RF to investigate the attitudes of Chinese public companies towards environmental protection [10]. The findings indicate that specific policies have the potential to foster more favorable attitudes toward carbon reduction, and perspectives on ecological issues vary between industries. Koklu and Sulak estimated the environmental behavior via three different machine learning methods (logistic model tree (LMT), Support Vector Machine (SVM) and Decision Tree (DT)) [11]. A new data set based on an online questionnaire is prepared based on residents living in Turkey. The dataset consists of 37 different features obtained from 384 participants who participated in the questionnaire. According to the results, the accuracies of LMT, SVM and DT are 94.53%, 92.96% and 82.55%, respectively.

Environmental education plays an important role in raising people's awareness of environmental problems and contributing to taking precautions against environmental problems. Through environmental education, positive changes can be observed in individuals' attitudes and behaviors towards environmental problems. In this study, people's environmental attitude behaviors were predicted from the responses given to the online questionnaire using KNN, NB, and SVM methods. For objectivity and reliability of the analysis, the k-fold cross validation technique was used. The evaluation of the compared methods was based on four different performance metrics commonly used in classification algorithms: accuracy, precision, recall, and F1-score.

## **1.2 Data Set**

The dataset used in this study is an open-source dataset and is sourced from the Kaggle repository [11-13]. In line with the literature review and the information obtained in the relevant field, possible factors that may affect the environmental attitudes of individuals were determined. A data set was collected



through an online questionnaire based on the responses of individuals to the attributes created in line with the identified factors. The environmental attitudes data set was created from the survey responses of 384 people living in Turkey. There are 37 different features in the data set, 54% of the individuals participating in the survey in the data set are female and the remaining 46% are male.

### 1.3 Methods

#### 1.3.1 K Nearest Neighbors (KNN)

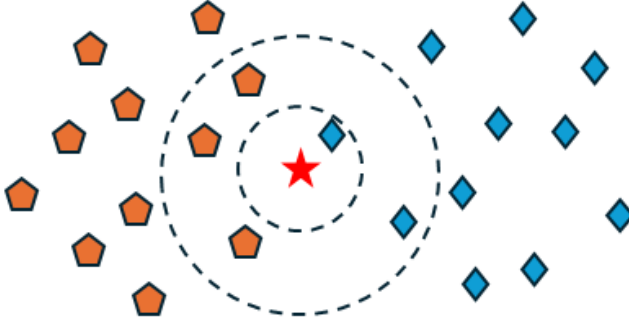
The objective of the KNN classification algorithm is to find similar instances in the training data. The class of the query sample is determined by the instances that are most frequent to the query instance. The similarity between samples is determined based on the closeness of each other. To measure the similarity, distance metrics are adopted, and hence different distance metrics are improved [14]. One of the commonly adopted distance measures for determining nearest neighbor samples is the Euclidean distance metric. Euclidean distance is defined as

$$Dist = \sqrt{\sum_{k=1}^N ((x_{ik} - x_{jk}))^2} \quad (1)$$

where  $x_i$  and  $x_j$  are the  $i^{th}$  and  $j^{th}$  samples in the data set,  $N$  equals to the number of features.

Another hyperparameter that affects the performance of the KNN method is the number of nearest neighbors. When small values of  $k$  are used in the KNN method, sensitivity to noise is high and may lead to overfitting. Because the noisy and misclassified examples in the training set have a high impact on the model's decision boundary. On the other hand, when large  $k$  values are chosen, more nearest neighbor samples are taken into account during decision boundary construction, which leads to smoother decision boundaries and underfitting. Because when more nearest neighbor samples are taken into account, the effect of noisy and misclassified samples decreases, but the model may not be able to detect complex patterns in the training set. Therefore, the  $k$  value should be determined to obtain the highest score for each data set [15]. The 2D synthetic dataset in Figure 1 illustrates the importance of determining the number of nearest neighbors. The dashed lines in the figure represent equidistant lines. As seen in Figure 1, when the  $k$  value is selected as 1, the class of the query example belongs to the blue diamonds class. However, when  $k$  is increased to 5, the class of the

query instance is assigned to the orange pentagon, since the majority of the classes in the nearest neighbors belong to the orange pentagon.



**Figure 1.** Determining the class of the query instance according to the number of nearest neighbors in the KNN algorithm

### 1.3.2 Naïve Bayes (NB)

NB is a statistical machine learning algorithm based on Bayes theorem [16]. In Naïve Bayes, the term "naïve" refers to the assumption of conditional independence among features, implying that the occurrence of one feature is independent of the occurrence of others. Due to this assumption, the NB method achieves good results in different application areas even if the independence of features cannot be satisfied in real-world datasets. In addition, NB achieves satisfactory results on datasets with a large number of features and is therefore widely adopted for sentiment analysis. Based on the bayes theorem, posterior probability can be defined as

$$P(C_i|F) = \frac{P(f_1, f_2, \dots, f_N | C_i) \times P(C_i)}{P(f_1) \times P(f_2) \dots \times P(f_N)} \quad (2)$$

- $C_i$  corresponds to  $i^{th}$  class value
- $F_j$  equals  $j^{th}$  feature

In NB, the class of the query instance is determined by calculating the posterior probability for each class, with the class having the highest probability determined as the predicted class. The commonly employed version of the NB is the Gaussian NB. In Gaussian NB, the decision function is developed from the Gaussian distribution and is defined as follows [17]:

$$P(f_j | C_i) \cong \frac{1}{\sqrt{2\pi}\sigma_y} e^{-\frac{(x_{ij}-\mu_y)^2}{2\sigma_y^2}} \quad (3)$$

where  $\sigma_y$  is standard deviation of the feature and  $\mu_y$  equals mean of the feature.

$$P(x_{ij} | C) \cong \frac{1}{\sqrt{2\pi}\sigma_{(j|C)}} e^{-\frac{(x_{ij}-\mu_{(j|C)})^2}{2\sigma_{(j|C)}^2}} \quad (4)$$

where  $\sigma_{(j|C)}$  is standard deviation of the  $j^{\text{th}}$  feature and  $\mu_{(j|C)}$  equals mean of the  $j^{\text{th}}$  feature at given class  $C$ .

### 1.3.3 Support Vector Machines (SVM)

SVM is one of the most frequently used machine learning algorithms due to its high success in applications in different fields. The main aspiration of SVM is to find the optimal hyperplane that minimizes the classification error while maximizing the difference between the closest samples in the classes [18,19]. Consider the problem of splitting a set of training data into two separate classes. Assume the training data set  $D$  is defined as

$$D = \{(x_i, y_i) | x_i \in R^N, y_i \in (-1, +1)\} \quad i = 1, \dots, M \quad (5)$$

where  $x_i$  equals to the  $i^{\text{th}}$   $N$  dimensional training data and  $y_i$  is the label of the  $i^{\text{th}}$  training data. In the linearly separable data, the hyperplane  $f(x)$  can be defined as

$$f(x) = \mathbf{w}^T x + b = \sum_{j=1}^M w_j x_j + b = 0 \quad (6)$$

where  $b$  is bias and  $\mathbf{w}$  equals to the  $N$  dimensional weight vector.  $\mathbf{w}$  and  $b$  are determined during the learning. In linearly separable data, a separating hyperplane for the two classes are defined as

$$\mathbf{w} \cdot x_i + b \geq +1 \quad \text{for } y_i = +1 \quad (7)$$

$$\mathbf{w} \cdot x_i + b \leq -1 \quad \text{for } y_i = -1 \quad (8)$$

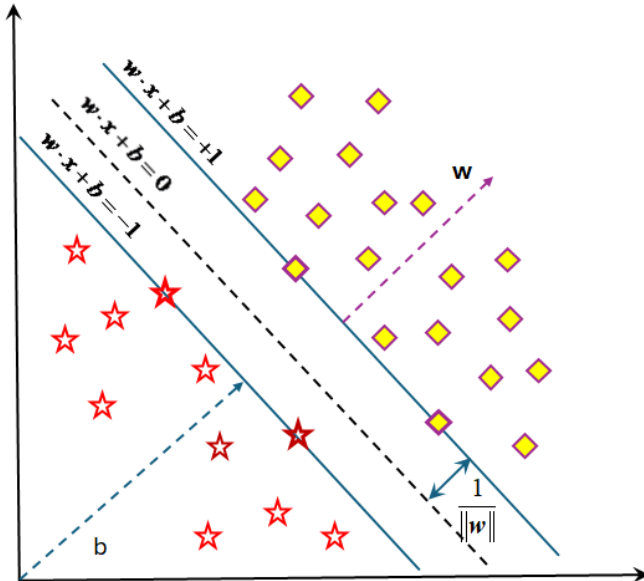
Equations 7 and 8 can be combined as

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1 \geq 0 \quad (9)$$

The samples nearest to the hyperplane are called support vectors. Margin is defined as the distance between the hyperplane and the support vectors, and it is equal to  $2/\|\mathbf{w}\|$ . Maximizing this margin results in the following constrained optimization problem, subject to the inequality constraints specified in Equation 9.

$$\min \left\{ \frac{1}{2} \|\mathbf{w}\|^2 \right\} \quad (10)$$

In Figure 2, the geometrical representation of the SVM on the 2D synthetic data is demonstrated. The optimal hyperplane with the maximum margin is drawn with a solid line black color. Support vectors are highlighted with increasing the marker edge color.



**Figure 2.** The illustration of SVM on the linearly separable data

Commonly, data sets are not linearly separable and hence the constraints in Equation 10 cannot be satisfied. In such cases, misclassification of certain data points can be permitted by introducing slack variables ( $\xi_i$ ) and an error penalty term (C). The constrained optimization problem becomes

$$\begin{aligned} \text{Minimize} \quad & \frac{1}{2} \|w\|^2 + C \sum_{i=1}^M \xi_i \\ \text{Subject to:} \quad & \begin{cases} y_i (w \cdot x + b) \geq 1 - \xi_i \\ \xi_i \geq 0 \quad i = 1, \dots, M \end{cases} \end{aligned} \quad (11)$$

Let  $\alpha_i$  represent the Lagrange multipliers corresponding to the inequality constraints in Equation 11. The corresponding Lagrangian is then defined as:

$$\text{Minimize } L(w, b, \alpha) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^M \xi_i - \sum_{i=1}^M \alpha_i [y_i (w \cdot x_i + b) - 1] \quad (12)$$

After performing the required computations, dual form can be obtained as

$$\begin{aligned} J(\alpha) &= \sum_{i=1}^M \alpha_i - \frac{1}{2} \sum_{i=1}^M \sum_{j=1}^M \alpha_i \alpha_j y_i y_j x_i x_j \\ \text{Subject to:} \quad & \begin{cases} \sum_{i=1}^M \alpha_i y_i = 0 \\ 0 \leq \alpha_i \leq C \end{cases} \end{aligned} \quad (12)$$

$w$  and  $b$  can be obtained as

$$\begin{aligned} w &= \sum_{i=1}^M \alpha_i y_i x_i \\ b &= \frac{1}{p} \sum_{i=1}^M y_i - w \cdot x \end{aligned} \quad (13)$$

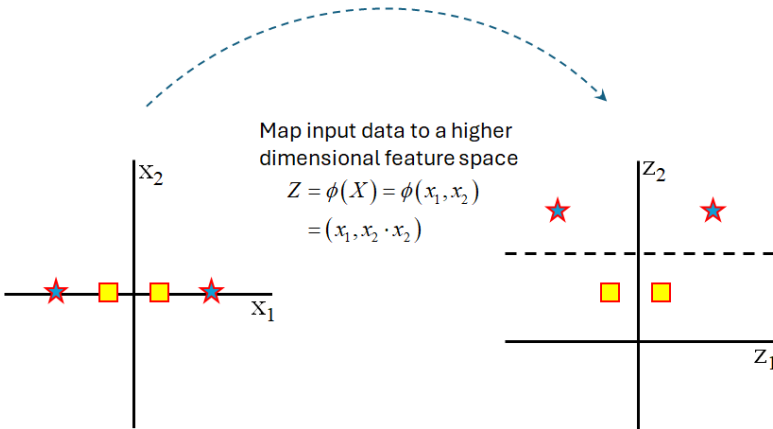
The decision of the SVM is

$$f(x) = \text{sign} \left( \sum_{i=1}^M \alpha_i y_i (x_i \cdot x) + b \right) \quad (14)$$

When the classes cannot be linearly separable widely known technique kernel trick is employed. Via kernel trick, non-separable data in the input feature space can be separable through mapping a higher dimensional feature space. By replacing the dot product in Equation 14 with the kernel function, the decision function can be expressed as

$$f(x) = \text{sign}\left(\sum_{i=1}^M \alpha_i y_i K(x_i, x_j) + b\right) \quad (15)$$

where  $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$  is the kernel function. Many different kernels have been improved. The most commonly used kernels are linear, polynomial, radial bases function (RBF) and hyperbolic tangent. Figure 3 depicts the mapping of input space to a higher dimensional feature space.



**Figure 3.** Non-linearly separable data in the input space becomes linearly separable via kernel trick by mapping the input space to a higher dimensional feature space.

The determination of the kernel function and kernel-specific hyperparameters is important as it affects the performance of the classifier. The mathematical equations of the four most frequently used kernels and their hyperparameters are given in Table 1.

**Table 1.** SVM kernels

Kernel function	Equation
Linear Kernel	$K(x_i, x_j) = x_i^T x_j$
Polynomial Kernel	$K(x_i, x_j) = (x_i^T x_j + 1)^d$
Radial Basis Function (RBF)	$K(x_i, x_j) = e^{-\frac{\ x_i - x_j\ ^2}{2\sigma^2}}$
Hyperbolic Tangent Kernel	$K(x_i, x_j) = \tanh(\alpha x_i^T x_j + c)$

## 1.4 Results

### 1.4.1 Performance Metrics

Different performance metrics are employed to measure the success of machine learning methods relative to each other. These are derived from the confusion matrix, also called the error matrix. It is a visualization tool that helps show the performance of machine learning models with their correct and incorrect predictions. Confusion matrix for binary classification problems is demonstrated in Table 2.

**Table 2.** Confusion matrix for binary classification

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

In Table 2, true positive (TP) indicates the model's correct predictions of the positive class, false negative (FN) shows the incorrect prediction of the negative class while the actual class is positive, false positive (FP) equals the model's prediction is incorrectly positive, but the actual class is negative, and true negatives (TN) represents the number of correctly predicted negative classes.

Widely adopted performance matrices for binary classification problems are accuracy, precision, recall, and F1-score.

Accuracy: measures the correctly classified samples to all samples in the data set. It is defined as

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (16)$$

Precision: quantifies the ratio of true positive predictions to all positive predictions. It is given by

$$Precision = \frac{TP}{TP + FP} \quad (17)$$

Recall: also called the true positive rate or sensitivity, calculates the ratio of true positive predictions to all positive samples in the dataset. It is calculated as

$$Recall = \frac{TP}{TP + FN} \quad (18)$$

F1-score: merges a model's precision and recall metrics. It is measured by taking the harmonic mean of precision and recall and defined as

$$F1 - score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} \quad (19)$$

### 1.4.2 Experimental Results

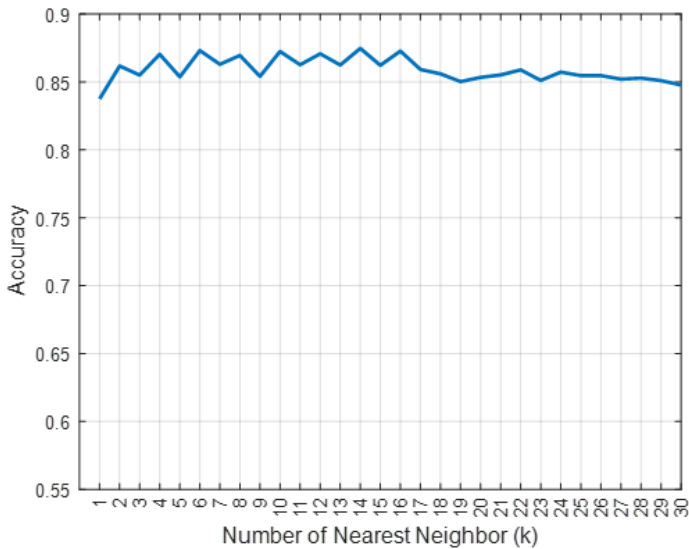
Machine learning methods take hyperparameters that are specific to the methods. The performance of the machine learning methods may vary greatly based on the setting of the hyperparameters. Hence, these hyperparameters need to be tuned based on the dataset in order to obtain the highest result on the data set. Although there are many techniques to adjust the method specific hyperparameters, the grid search technique is one of the simplest of them. In grid search technique, the machine learning model is constructed with each hyperparameter pair and the result of the model is computed. Among these results, the hyperparameter pairs that yield the highest scores are determined. Accordingly, the method specific hyperparameters of each benchmarked method are tuned to achieve the highest scores in each method. The search space of the benchmarked method along with its range are given in Table 3.



**Table 3.** the hyperparameter search range of the methods

Method	Hyperparameter	Value
K Nearest Neighbors	k	1, 2, ..., 30
Naive Bayes	var_smoothing	$10^{-12}, 10^{-11}, \dots, 10^2$
Support Vector Machines	$\gamma$ C	$2^{-10}, 2^{-9}, \dots, 2^5$ $2^{-4}, 2^{-3}, \dots, 2^{15}$

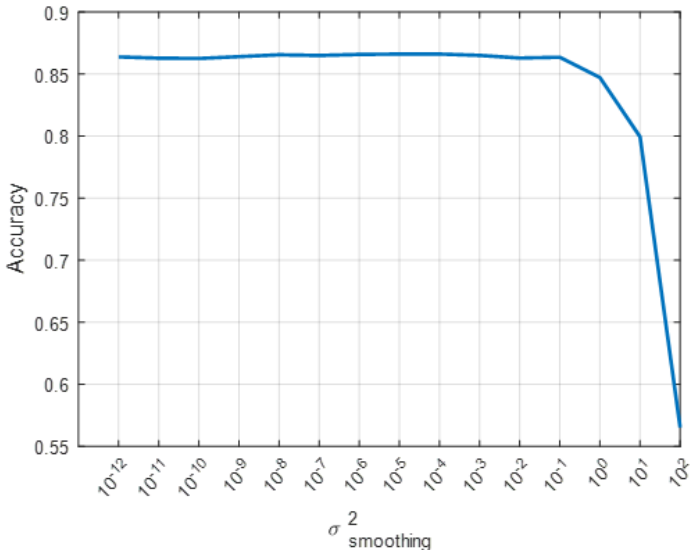
The accuracy scores of the KNN method is shown in Figure 4. After scaling the data with the standard scaling method, the performance of the KNN in prediction does not change much despite the variation in the k number. The lowest performance is obtained when  $k = 1$  (0.8375) and the highest scores are obtained when  $k = 14$  (0.8747). The difference in performance between the highest and lowest scores is 4.44%.



**Figure 4.** Performance result of the KNN

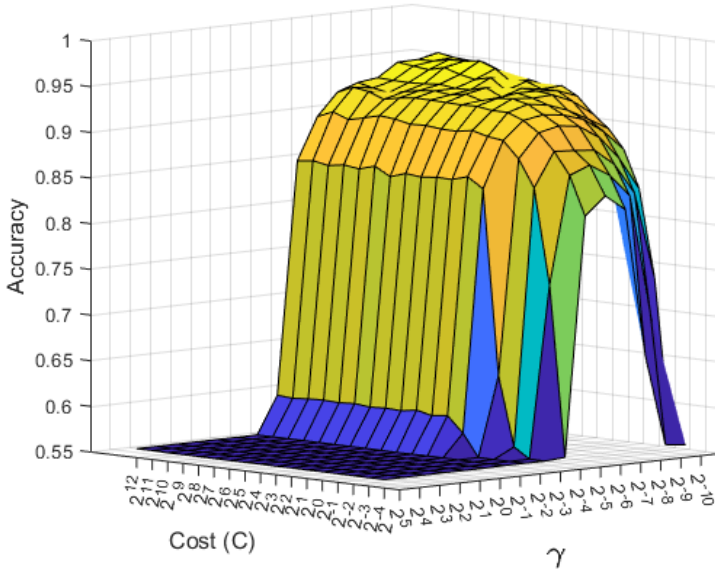
The performance of the NB method with respect to varying *var\_smoothing* hyperparameter values is shown in Figure 5. As can be seen from the figure, the performance change of the NB method is quite limited in the values of the *var\_smoothing* hyperparameter between  $10^{-12}$  and  $10^2$ . The difference between the highest and lowest scores among the specified *var\_smoothing* values is 0.4%. The highest achieved are achieved when *var\_smoothing* is  $10^{-4}$  (0.8662). After

the *var\_smoothing* value is increased beyond  $10^0$ , the performance of the method drops significantly.



**Figure 5.** Performance result of the NB

The accuracy score of SVM with respect to changing  $\gamma$  and  $C$  is shown in Figure 6. In SVM with radial basis function kernel, the highest scores are obtained in the range of  $\gamma$  values  $2^{-4}$  and  $2^{-10}$ ,  $C$  values  $2^2$  and  $2^{10}$ . For hyperparameter values other than these values, performance degradation starts to be observed. The highest accuracy is achieved when  $\gamma = 2^{-9}$  and  $C = 2^8$ . The lowest performances are observed when the  $\gamma$  value is lower than  $2^{-1}$ , and further increasing the  $\gamma$  values does not affect the result.



**Figure 6.** Performance result of the SVM

The hyperparameters for which the highest accuracy values were obtained in the compared methods and the results of other performance metrics in these hyperparameters are given in Table 4. In Table 4, the best performance across all metrics is highlighted in bold. In all performance metrics, the SVM method with RBF kernel achieved the highest results. In terms of accuracy metric, the SVM method achieved 9.48% better performance than the KNN method and 10.55% better performance than the NB method.

**Table 4.** Best performance results of the compared methods

Method	Accuracy	F1-score	Precision	Recall
K Nearest Neighbors k = 14	0.8747	0.8873	0.8731	0.9061
Naive Bayes var_smoothing=10 <sup>-4</sup>	0.8662	0.8756	0.8928	0.8639
Support Vector Machines $\gamma = 2^{-9}$ and $C = 2^8$	<b>0.9576</b>	<b>0.9615</b>	<b>0.9576</b>	<b>0.9670</b>

In the study using the same dataset, LMT, SVM and DT methods yielded 94.53%, 92.96% and 82.55% results, respectively [11]. While the best result (94.53%) was obtained with the LMT method in the study conducted by Koklu and Sulak, 95.76% result was obtained with the SVM method in this study and an increase of 1.30% was achieved.

## **1.5 Conclusion**

As environmental problems increase, the importance of taking measures to solve the problems increases day by day. People also have great responsibility in solving environmental problems because one of the factors that cause environmental problems is humans. Individuals can reduce environmental problems by making changes in their daily activities. This is possible with environmental education and awareness. As in other fields, machine learning methods can be used to determine environmental behaviors. In this study, the analysis of machine learning methods KNN, NB, and SVM was made on the prediction of people's environmental attitudes. According to the experimental results, the highest scores in all performance metrics were obtained in the SVM method (accuracy = 0.9576, F1-score = 0.9615, precision = 0.9576, recall = 0.9670). The results of the other two methods are close to each other, the KNN method achieved 0.98% better results than the NB method in terms of accuracy performance.

In future studies, an increase in this regard can be achieved by paying attention to the following points. Feature selection methods can be adapted so that prominent features can be identified, and memory efficiency can be achieved. New machine learning methods can be developed to increase the accuracy of models.

## References

- [1] Ukaogo, P. O., Ewuzie, U., & Onwuka, C. V. (2020). Environmental pollution: causes, effects, and the remedies. In *Microorganisms for sustainable environment and health* (pp. 419-429). Elsevier.
- [2] Hardoy, J. E., Mitlin, D., & Satterthwaite, D. (2024). *Environmental problems in Third World cities*. Taylor & Francis.
- [3] Celik, S. (2020). The effects of climate change on human behaviors. *Environment, climate, plant and vegetation growth*, 577-589.
- [4] Wang, Y., Hao, F., & Liu, Y. (2021). Pro-environmental behavior in an aging world: Evidence from 31 countries. *International Journal of Environmental Research and Public Health*, 18(4), 1748.
- [5] Hidalgo-Crespo, J., Coello-Pisco, S., Reyes-Venegas, H., Bermeo-Garay, M., Amaya, J. L., Soto, M., & Hidalgo-Crespo, A. (2022). Understanding citizens' environmental concern and their pro-environmental behaviours and attitudes and their influence on energy use. *Energy Reports*, 8, 103-109.
- [6] Beiser-McGrath, L. F., & Huber, R. A. (2018). Assessing the relative importance of psychological and demographic factors for predicting climate and environmental attitudes. *Climatic change*, 149, 335-347.
- [7] Choudhury, N., Mukherjee, R., Yadav, R., Liu, Y., & Wang, W. (2024). Can machine learning approaches predict green purchase intention?-A study from Indian consumer perspective. *Journal of Cleaner Production*, 456, 142218.
- [8] Wang, Q., Kou, Z., Sun, X., Wang, S., Wang, X., Jing, H., & Lin, P. (2022). Predictive analysis of the pro-environmental behaviour of college students using a decision-tree model. *International Journal of Environmental Research and Public Health*, 19(15), 9407.
- [9] Lou, X., Lin, Y., & Li, L. M. W. (2022). Predicting priority of environmental protection over economic growth using macroeconomic and individual-level predictors: Evidence from machine learning. *Journal of Environmental Psychology*, 82, 101843.
- [10] Li, C., Li, L., Zheng, J., Wang, J., Yuan, Y., Lv, Z., ... & Liu, W. (2022). China's public firms' attitudes towards environmental protection based on sentiment analysis and random forest models. *Sustainability*, 14(9), 5046.
- [11] KOKLU, N., & SULAK, S. A. (2024). Classification of Environmental Attitudes with Artificial Intelligence Algorithms. *Intelligent Methods In Engineering Sciences*, 3(2), 54-62.

- [12] Koklu, N., & Sulak, S. A. (2024). The Systematic Analysis of Adults' Environmental Sensory Tendencies Dataset. *Data in Brief*, 110640.
- [13] Koklu N, Sulak SA. Environmental Attitude Dataset. Kaggle. <https://www.kaggle.com/datasets/suleymansulak/environmental-attitude-dataset>: 2024..
- [14] Ozbay, A., Degirmenci, A., & Karal, O. (2023, October). A Comparative Analysis of Machine Learning Algorithms for Accurate Step Detection in Wrist Worn Devices. In *2023 Innovations in Intelligent Systems and Applications Conference (ASYU)* (pp. 1-5). IEEE..
- [15] Ali, N., Neagu, D., & Trundle, P. Evaluation of k-nearest neighbour classifier performance for heterogeneous data sets. *SN Appl. Sci.* 1, 1559 (2019).
- [16] Çakır, M., Degirmenci, A., & Karal, O. (2022, February). Exploring the Behavioural Factors of Cervical Cancer Using ANOVA and Machine Learning Techniques. In *International Conference on Science, Engineering Management and Information Technology* (pp. 249-260). Cham: Springer Nature Switzerland.
- [17] Vishwakarma, M., & Kesswani, N. (2023). A new two-phase intrusion detection system with Naïve Bayes machine learning for data classification and elliptic envelop method for anomaly detection. *Decision Analytics Journal*, 7, 100233.
- [18] Degirmenci, A. (2022). Performance comparison of kNN, random forest and SVM in the prediction of cervical cancer from behavioral risk. *Int. J. Innov. Sci. Res. Technol*, 7(10).
- [19] Tokgöz, N., Değirmenci, A., & Karal, Ö. (2024). Machine Learning-Based Classification of Turkish Music for Mood-Driven Selection. *Journal of Advanced Research in Natural and Applied Sciences*, 10(2), 312-328.

## Chapter 2

### **A Review on Renewable Energy-Powered Near-Field Wireless Charging Station for Electric Vehicles**

**Mehmet Zahid EREL<sup>1</sup>, Yunus YALMAN<sup>2</sup>**

---

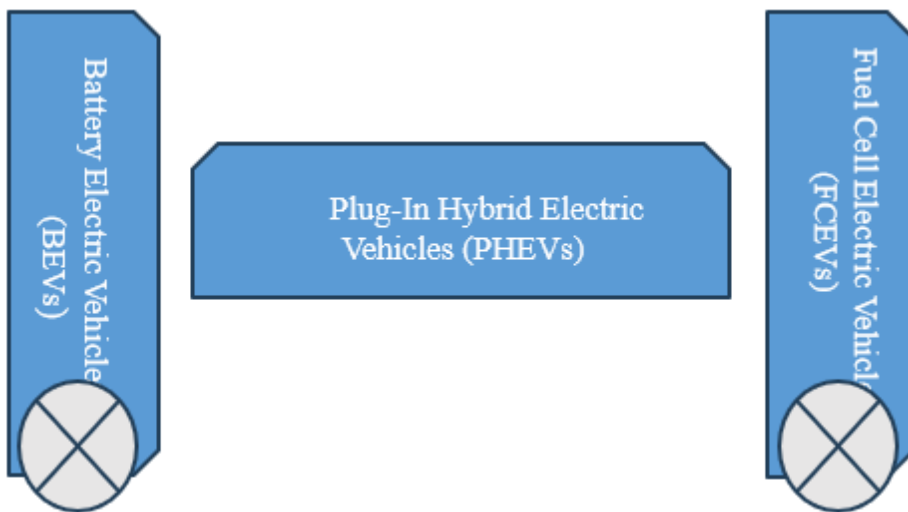
<sup>1</sup> Dr. Öğr. Üyesi, Ankara Yıldırım Beyazıt Üniversitesi, Enerji Sistemleri Mühendisliği, Ankara, Türkiye  
mzerel@aybu.edu.tr, ORCID: 0000-0003-1663-8394

<sup>2</sup> Dr. Öğr. Üyesi, Ankara Yıldırım Beyazıt Üniversitesi, Elektrik-Elektronik Mühendisliği, Ankara, [Türkiye](#)  
[yunusyalman@aybu.edu.tr](mailto:yunusyalman@aybu.edu.tr), ORCID: 0000-0003-4792-1774

## 1.1 Introduction

The global transition from fossil-fueled vehicles to Electric Vehicles (EVs) is rapidly progressing, driven by government policies and advancements in clean energy research, which reduce dependence on non-renewable energy sources. EVs play a central role in decreasing carbon emissions and fuel costs. The EV industry is evolving increasingly with a variety of types on the market. There are three types of EVs available on the market: battery EVs, plug-in hybrid EVs and fuel cell EVs, as depicted in Figure 1. This categorization is determined by the type of energy sources utilized in EVs (Veza et al., 2023).

Although EVs themselves contribute grid stability, most charging stations rely on grid power, which is often derived from fossil fuel sources. Therefore, charging infrastructure should be modified to incorporate renewable energy sources, reducing fossil fuel consumption. As the number of EVs on the road increases, the demand for innovative charging solutions that are convenient, reliable, and environmentally friendly also rises.



**Figure 1.** Types of Electric Vehicles on the market

According to the International Energy Agency, EVs have significant potential, with sales increasing 360-fold from 2010 to 2020 (Desai et al., 2023). Although the demand for EVs is anticipated to rise year by year, they still have drawbacks, including limited driving range, long charging times, heavy plug-in cables, high cost and lower energy density batteries. To solve these aforementioned problems, wireless power transfer (WPT) can be a good candidate for EV charging mechanism. WPT technology offers reliable and safe charging along with increased driving range. To decrease dependence on fossil fuels while increasing



accessibility and sustainability, renewable energy sources combined with wireless charging technology should be integrated to EV charging infrastructure.

## **1.2 History of Wireless Power Transfer Technology**

WPT is a method by which single or multiple transmitters generate electromagnetic waves that are received by either single or multiple receivers without any physical contact. The initial experiment to transmit electrical energy was put forward by Nikola Tesla at the end of the 19th century. As stated in (Popovic et al., 2013; Triviño-Cabrera et al., 2020) he utilized microwaves to transmit power between power transfer objects that were placed 48 km away from each other. Tesla also conducted an experiment involving 200 bulbs powered wirelessly at a distance of 25 miles from the power source. Besides Tesla, Schuler and Glaser also contributed to WPT technology, focusing on biomedical and microwave transfer applications, respectively (Erel et al., 2022).

However, these experiments could not be continued due to a lack of advancements in semiconductor technology and, consequently, the power electronics industry. With the development of semiconductor technology, WPT has an attractive research topic for the academia and also industry. This technology can be classified into near-field and far-field types, depending on the transfer distance. Although low transfer distance ( $< 1\text{ m}$ ) is related with near-field WPT applications, long transfer distance ( $> 1\text{ m}$ ) is directly proportional with far-field WPT applications. This chapter will primarily focus on near-field applications for renewable energy-powered wireless EV charging.

## **1.3 Renewable Energy Powered Near-Field WPT Technologies**

The near-field WPT is intended for small air gaps that range from a few mms to several 100s of millimeters. Because of this, EV charging is included in the near-field WPT concept. The coupling coefficient is a crucial factor between the power transfer components.

### **1.3.1 Inductive Power Transfer Technology**

Inductive power transfer (IPT) is a type of wireless charging method using magnetic field to transfer power between transmitter and receiver coils (Imtiaz et al., 2024). The conventional IPT system is depicted in Figure 2. Here, converters play a critical role in exciting the resonant components as inverters while also supplying DC voltage for the battery as rectifiers. Primary compensation network contributes to the wireless charging system by increasing voltage, enabling impedance matching, reducing harmonics, facilitating frequency tuning and controlling power flow, thereby optimizing the overall efficiency of the system.

On the other hand, secondary compensation network primarily focuses on voltage regulation, adapting to load changes, and ensuring efficient power transfer to the load. Transmitter and receiver coils are typically built with conventional coil shapes such as rectangular, circular and square that offer cost effectiveness, ease of integration, and design simplicity.

IPT technology has been extensively researched by industries and research institutes, with significant research growth. Prominent contributors include companies and institutions such as Ossia, WiTricity, Oak Ridge National Laboratory, among others (Khalid et al., 2023). Literature review on renewable energy powered IPT system is investigated alongside IPT technology under the following subtopics.

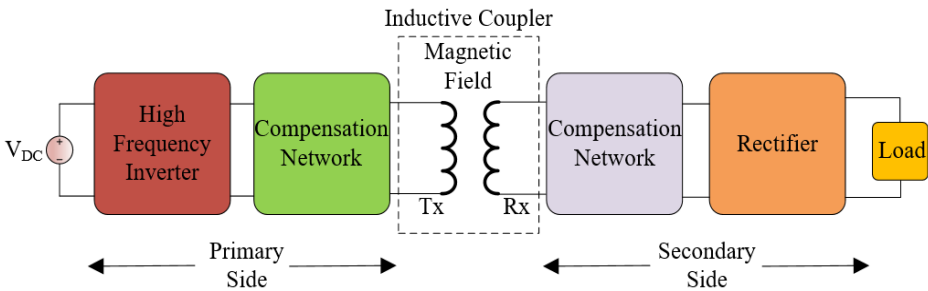


Figure 2: A conventional IPT system

### Solar Energy-Powered IPT Technology

To improve the ability of wireless charging systems, renewable energy sources have been recently attracted and integrated into the system. IPT technology stands out within this concept. Compared to existing renewable energy sources, research on solar energy stands out for IPT. This subtopic reviews the significant research under this concept. The typical representation of the solar energy based IPT system is depicted in Figure 3. Due to the variable irradiance of PV arrays, DC-DC converters with MPPT are essential for these types of applications. Chhawchharia et al. reviewed renewable energy-powered WPT applications along with limitations (Chhawchharia et al., 2018). Subudhi et al. proposed a grid-integrated wireless EV charging system to provide uninterrupted charging along with reduced harmonics (Subudhi et al., 2023). Pan et al. discussed IPT technology in the context of an effective cabin cooling system (Pan et al., 2017). A 3.3 kW PV fed wireless system is also proposed in EV charging along with hybrid compensation topology (Arulvendhan et al., 2024). PV-integrated wireless charging for drones is proposed with the aim of extending flight range and eliminating the need for wired connections when charging multiple drones (Chittoor & Bharatiraja, 2023). IPT with solar energy is also discussed in light

electric vehicle applications (Ghosh et al., 2020; Joseph et al., 2021). Economic viability of a dynamic wireless charging system integrated with solar energy is discussed in (Li et al., 2023).

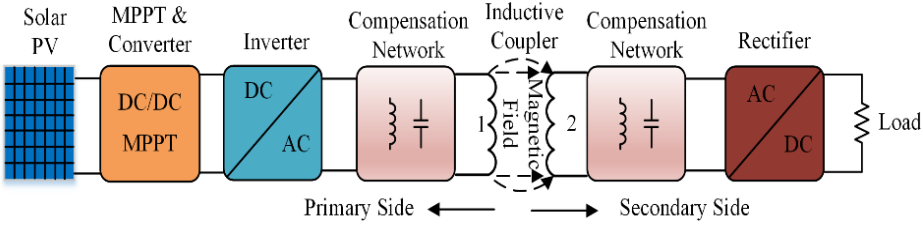


Figure 3: A typical solar-based IPT system

Fuel Cell-Powered IPT Technology

A fuel cell (FC) is an electrochemical device that converts the chemical energy of a fuel and an oxidizing agent directly into electricity through a series of redox reactions (Celebi et al., 2021). Among the various fuel cell types, proton exchange membrane (PEM) fuel cell stands out especially transport applications due to its high power density, low operating temperature, fast start and response under variable conditions. Due to the inherently low voltage and high current density of the fuel cells, the output voltage must be increased to meet the requirements of high power EV charging applications. A typical FC-based IPT system is demonstrated in Figure 4. Herein, FC supplies the inverter to obtain alternating current for the resonant components. Büyük et al. discussed FC integrated IPT system using quadratic boost converter structure to provide high efficiency and compactness (Büyük et al., 2022). A FC-based IPT system is suggested for EV charging applications where supercapacitors are utilized as the primary energy source (Campagna et al., 2024). A hybrid charging is also proposed to optimize the power transfer for EV charging applications (Optimized & Technology, 2024). As can be seen from the literature, the integration of IPT is an emerging topic but not yet a topic that has been extensively studied.

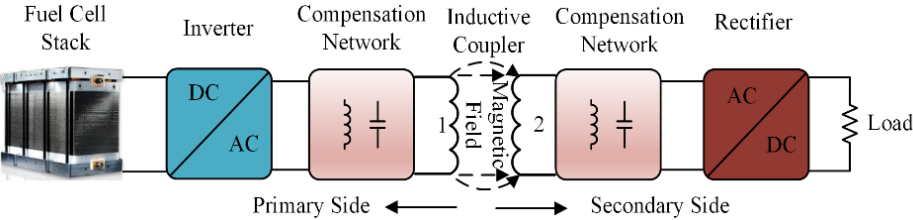


Figure 4: A typical fuel cell based IPT system

### *Benefits of Renewable Energy Powered IPT*

Renewable energy-powered IPT technology offers several compelling benefits, especially when powered by solar energy and fuel cells. First, by harnessing clean and sustainable energy sources, this technology significantly decreases greenhouse gas emissions and lowers the environmental impact of power generation. Solar-powered IPT systems are particularly advantageous in locations with high solar irradiance, providing high power transfer. FC-powered IPT, on the other side, allows for efficient, low-emission power generation even in situations where sunlight is limited, offering a more consistent and reliable energy supply. Additionally, FC has a nature of high energy density and can be the primary energy source of the system. FC also has a good ability to work together with other renewable energy sources and batteries (İnci et al., 2021). The contactless nature of IPT technology enhances system durability and safety, as it eliminates the need associated with physical connectors. This makes renewable-powered IPT technology well-suited for harsh environments where traditional power connections might degrade. In summary, integrating IPT technology with solar PV arrays and fuel cells enables a sustainable and efficient power transfer solution that aligns with global clean energy goals and advances energy resilience across applications.

### *Challenges and Limitations of Renewable Energy Powered IPT*

Despite IPT potential both in industry and academia, renewable energy-powered IPT technology faces several challenges and limitations. Solar-powered IPT systems are highly dependent on sunlight, and fluctuations in harsh weather conditions, as well as nighttime hours, can lead to reliability issues for continuous power applications. To address this intermittent, energy storage solutions or hybrid systems should be used, adding complexity and cost to the system. Fuel cell-powered IPT offers a more consistent energy output, but it also comes with its own set of challenges. Fuel cells often need specific storage and handling of hydrogen or other fuels, which can be costly and necessitates careful management of fuel supply chains, especially in remote areas. Additionally, both solar and fuel cell-based IPT systems comprise high initial costs for the installation of renewable power generation and sophisticated power electronics control side. Due to the low voltage nature of the FC, extra DC-DC boost converter structure needs to be included to the system and causes to extra cost, size and control for the EV charging applications (İnci & Türksöy, 2019). Additionally, the integration of IPT with renewable energy sources is hardly managed to keep power stability, especially variable weather environments. Efficiency losses in the IPT process, especially over large transfer distances, further decrease the net

energy benefit, posing a challenge for scaling up this technology. Despite these challenges and limitations, renewable-powered IPT remains a promising technology for sustainable and efficient energy transfer solutions.

### 1.3.2 Capacitive Power Transfer Technology

A typical representation of a conventional CPT system is depicted in Figure 5. Despite the use of magnetic field, CPT utilizes electric field to power transfer mechanism. The primary and secondary sides form the CPT system. While the primary side includes a high frequency inverter and the compensation network, the secondary side is formed with the compensation network, rectifier, and load (Erel et al., 2023). The capacitive coupler serves as a power transfer medium. A high frequency inverter is utilized to excite both resonant components through displacement current. A passive rectifier component is used to supply the DC load. The compensation networks enable us to manage the power transfer density and efficiency of the system. Filter-type compensation networks, such as L-L, LC-LC, LCL-LCL, LCLC-LCLC, and CLLC-CLLC, are often preferred in conventional CPT systems for EV charging applications (Lu et al., 2017). Additionally, coupling interface that is built with metal plates plays an essential role in the wireless charging system. This structure can be defined by the number of metal plates used. Aluminum metal plates are often used for coupling interface compared to other metal plates since its low cost and low weight. The transfer distance of the system was previously limited to several millimeter ranges, nevertheless; recent advancements in power converter technology have made this approach feasible for EV charging. Literature review on renewable energy powered CPT system is investigated under the following subtopics.

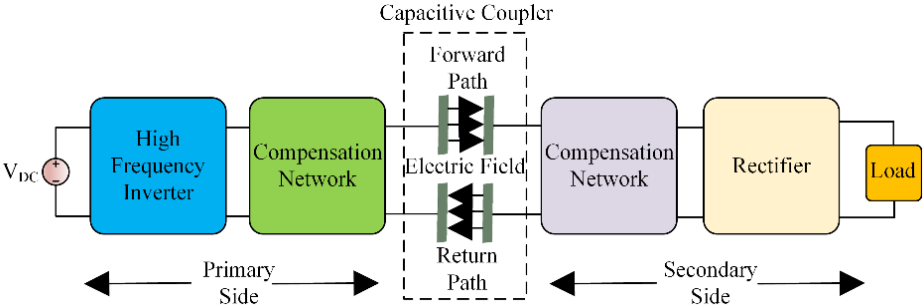


Figure 5: A conventional CPT system

### Solar Energy-Powered CPT Technology

Due to the low coupling capacitance of the CPT system, coupling capacitances need to be increased to integrate EV charging. To enhance the power transfer capability of the system either the surface area of the metal plates or the switching frequency of the converters can be increased. Compared to IPT technology, CPT is a relatively recent research area and requires further investigation, especially high power applications. A conventional solar energy based CPT system is depicted in Figure 6. Herein, PV with DC-DC converter using MPPT techniques play a key role to obtain stable output voltage for the inverter. Additionally, integration with solar energy presents a promising opportunity, but there is limited research on this concept. Zang et al. proposed a modular and flexible PV integration with CPT system to increase the power transfer capacity and ability (Zang et al., 2020). Additionally, an MPPT algorithm is suggested to ensure maximum power transfer at the system.

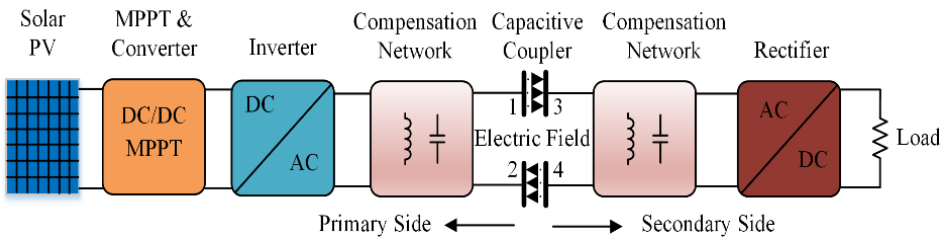


Figure 6: A typical solar energy-based CPT system

### Benefits of Renewable Energy Powered CPT

A detailed review of renewable energy-powered CPT applications reveals that solar energy is a prominent and yet underexplored area within this concept. Actually, this combination offers multiple advantages in achieving efficient, clean energy transmission, particularly, EV charging applications. The main benefit is to reduce reliance on fossil fuels and minimize greenhouse gas emissions. The contactless nature of CPT further enhances durability and decreases maintenance costs without using wires, especially for hard environments. Additionally, CPT systems have a high tolerance for harsh environmental conditions, allowing solar panels to operate independently from the load, while safely transferring power. Due to the modular design potential of solar powered CPT provides flexible scalability and hence, individual modules can be added or combined to increase power output as needed. This flexibility, combined with the low operational costs of solar energy, makes solar-powered CPT a promising solution for sustainable and efficient power transfer in applications like electric vehicle charging applications.

Moreover, CPT near metal objects does not significantly reduce system efficiency compared to IPT technology. Also, costly litz wires are not as critical in CPT systems as they are in the IPT systems. Low cost and low weight aluminum is also used for power transfer compared to IPT. These advantages suggest that solar-powered CPT is likely to become more feasible in the near future.

Challenges and Limitations of Renewable Energy Powered CPT

The main limitations and challenges of this type of system include the need to increase the surface area of the metal plates, which in turn increases the overall size of the system that is undesirable in EV charging applications. As the switching frequency also increases, the switching losses can be increased at the system. To enhance the power transfer ability of the system, the number of increased passive compensation components are used in the system. However, losses are increased in this method. Compared to IPT, CPT has low coupling capacitance value and thus low power transfer ability.

**1.4 Energy Management Systems of Renewable Energy Powered Near-Field WPT Technologies**

Energy management systems are crucial for controlling the flow of energy/power flow within the system (Aboumadi et al., 2023). The prominent works on Renewable energy powered WPT technologies are mentioned in Table 1.

**Table 1.** Overview of EMS in Prominent Studies

Title	Inference
MPPT based Energy Management (Ghosh et al., 2022)	The energy management system is proposed depending on the power demand of the load and irradiance level. Herein, the system ensures efficient power management by allowing the PV to supply both the load and charge the battery when irradiance is above 50 %. Below 50 % irradiance, the battery discharges to maintain a constant 48V load voltage.
Hybrid charging management for EV (Optimized & Technology, 2024)	The proposed EMS aims to optimize the use of fuel cell and decrease the dependence on the grid.
Power flow management by different modes of hybrid charging system (Rai et al., 2023)	Depending on the power relations between EV and PV, energy management is regulated by; EV power < PV power EV power = PV power EV power > PV power

Hybrid Energy Management system for wireless E-scooter (Hu et al., 2018)	The primary energy is stored in the battery, with the SC providing supplementary power during specific conditions. Power flow is managed under rapid charging, normal charging, and smart charging modes.
PV and ESS integrated wireless hybrid compensated system (Arulvendhan et al., 2024)	PV power is a major source, and ESS is a backup source. When ESS reaches its maximum, PV with excess energy and grid are utilized to charge the store.
Energy management in PV fed dynamic WPT application (Kumar et al., 2021)	When PV power is less because of the poor weather conditions. Required analysis are conducted to realize the supply voltage variations in the system.

### Conclusions and Discussion

In this chapter, the integration of renewable energy sources, specifically solar power and fuel cells, with near-field wireless power transfer (WPT) systems is reviewed in detail, with a focus on inductive and capacitive coupling techniques. This review is based on state-of-the-art literature and research articles. The advantages, limitations, challenges and feasibility of near-field WPT technologies based on renewable energy sources are considered within this concept. IPT and CPT technologies, while both effective in wirelessly transferring energy, each offer distinct advantages. Although IPT is highly suitable for relatively higher power levels and long transfer distance applications, such as charging electric vehicles, while CPT is more convenient for low-power and low transfer distances applications.

To reduce dependence on traditional grids, renewable energy-powered WPT technologies can serve as a good candidate, promoting clean energy and sustainability. Solar-powered WPT systems are particularly advantageous in locations with high solar irradiance, on the other side, FC-powered WPT systems provide more consistent and reliable power transfer even in situations where sunlight is limited. Both solar and fuel cell-based WPT systems include high initial costs along with the integration of renewable energy sources. Additionally, these WPT systems require additional power electronic converters with sophisticated control methods, which increase both the cost and size of the system.

As renewable energy technologies continue to evolve, WPT systems powered by clean energy could play a pivotal role in transforming the landscape of WPT, contributing to the global shift toward sustainable and energy-efficient technologies.



## References

- Aboumadi, A., Güneş, Z., Yalman, Y., Tan, A., Terciyanli, A., Terciyanli, E., & Bayindir, K. Ç. (2023). Design and Development of Energy Management System for Hybrid Microgrid. *14th International Conference on Electrical and Electronics Engineering, ELECO 2023 - Proceedings*, 2–6. <https://doi.org/10.1109/ELECO60389.2023.10415935>
- Arulvendhan, K., Kandadai Nagaratnam, S., Narayanamoorthi, R., Alharbi, M., & Hussien, S. (2024). Hybrid Compensation Based Efficient Wireless Charging System Design With Solar Photovoltaic Interface Toward Sustainable Transportation. *IEEE Access*, *12*(May), 87152–87166. <https://doi.org/10.1109/ACCESS.2024.3414169>
- Büyük, M., Savrun, M. M., & İnci, M. (2022). Analysis and modeling of wireless power transfer supported by quadratic boost converter interfaced fuel cell power source. *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, *35*(4), 1–14. <https://doi.org/10.1002/jnm.2997>
- Campagna, N., Castiglia, V., Gennaro, F., Messina, A. A., & Miceli, R. (2024). Fuel Cell-Based Inductive Power Transfer System for Supercapacitor Constant Current Charging. *Energies*, *17*(14), 1–22. <https://doi.org/10.3390/en17143575>
- Celebi, N., Arlı, F., Soysal, F., & Salimi, K. (2021). Z-scheme ZnO@PDA/CeO<sub>2</sub> heterojunctions using polydopamine as electron transfer layer for enhanced photoelectrochemical H<sub>2</sub> generation. *Materials Today Energy*, *21*. <https://doi.org/10.1016/j.mtener.2021.100765>
- Chhawchharia, S., Sahoo, S. K., Balamurugan, M., Sukchai, S., & Yanine, F. (2018). Investigation of wireless power transfer applications with a focus on renewable energy. *Renewable and Sustainable Energy Reviews*, *91*, 888–902. <https://doi.org/10.1016/j.rser.2018.04.101>
- Chittoor, P. K., & Bharatiraja, C. (2023). Building integrated photovoltaic powered wireless drone charging system. *Solar Energy*, *252*(September 2022), 163–175. <https://doi.org/10.1016/j.solener.2023.01.056>
- Desai, A., Kanika, & Patel, C. R. (2023). The impact of electric vehicle charging infrastructure on the energy demand of a city. *Energy Reports*, *9*, 814–823. <https://doi.org/10.1016/j.egy.2023.05.177>
- Erel, M. Z., Bayindir, K. C., & Aydemir, M. T. (2023). A new capacitive coupler design for wireless capacitive power transfer applications. *Engineering Science and Technology, an International Journal*, *40*, 101364. <https://doi.org/10.1016/j.jestch.2023.101364>

- Erel, M. Z., Bayindir, K. C., Aydemir, M. T., Chaudhary, S. K., & Guerrero, J. M. (2022). A Comprehensive Review on Wireless Capacitive Power Transfer Technology: Fundamentals and Applications. *IEEE Access*, *10*, 3116–3143. <https://doi.org/10.1109/ACCESS.2021.3139761>
- Ghosh, A., Ukil, A., & Hu, A. P. (2020). PV-Battery System with Wireless Power Transfer for LV Applications. *IECON Proceedings (Industrial Electronics Conference)*, *2020-Octob*, 4283–4287. <https://doi.org/10.1109/IECON43393.2020.9255316>
- Ghosh, A., Ukil, A., & Hu, A. P. (2022). Energy Management for Solar PV Generation with Contactless Power Transfer. *2022 7th IEEE Workshop on the Electronic Grid, EGRID 2022*, 1–5. <https://doi.org/10.1109/eGRID57376.2022.9990006>
- Hu, J. S., Lu, F., Zhu, C., Cheng, C. Y., Chen, S. L., Ren, T. J., & Mi, C. C. (2018). Hybrid Energy Storage System of an Electric Scooter Based on Wireless Power Transfer. *IEEE Transactions on Industrial Informatics*, *14*(9), 4169–4178. <https://doi.org/10.1109/TII.2018.2806917>
- Imtiaz, T., Elsanabary, A., Mubin, M., Soon, T. K., & Mekhilef, S. (2024). Enhancing Misalignment Tolerance Using Naturally Decoupled Identical Dual-Transmitter-Dual-Receiver Coils for Wireless EV Charging System. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, *12*(5), 5337–5351. <https://doi.org/10.1109/JESTPE.2024.3447286>
- İnci, M., Büyük, M., Demir, M. H., & İlbey, G. (2021). A review and research on fuel cell electric vehicles: Topologies, power electronic converters, energy management methods, technical challenges, marketing and future aspects. *Renewable and Sustainable Energy Reviews*, *137*(March 2020). <https://doi.org/10.1016/j.rser.2020.110648>
- İnci, M., & Türksoy, Ö. (2019). Review of fuel cells to grid interface: Configurations, technical challenges and trends. *Journal of Cleaner Production*, *213*, 1353–1370. <https://doi.org/10.1016/j.jclepro.2018.12.281>
- Joseph, P. K., Elangovan, D., & Sanjeevikumar, P. (2021). System Architecture, Design, and Optimization of a Flexible Wireless Charger for Renewable Energy-Powered Electric Bicycles. *IEEE Systems Journal*, *15*(2), 2696–2707. <https://doi.org/10.1109/JSYST.2020.2993054>
- Khalid, H., Mekhilef, S., Mubin, M., & Seyedmahmoudian, M. (2023). Advancements in inductive power transfer: Overcoming challenges and enhancements for static and dynamic electric vehicle applications. *Energy Reports*, *10*, 3427–3452. <https://doi.org/10.1016/j.egy.2023.10.008>

- Kumar, K., Chowdary, K. V. V. S. R., Sanjeevikumar, P., & Prasad, R. (2021). Analysis of Solar PV Fed Dynamic Wireless Charging System for Electric Vehicles. *IECON Proceedings (Industrial Electronics Conference)*, 2021-*Octob*, 1–6. <https://doi.org/10.1109/IECON48115.2021.9589677>
- Li, C., Dong, X., Cipcigan, L. M., Haddad, M. A., Sun, M., Liang, J., & Ming, W. (2023). Economic Viability of Dynamic Wireless Charging Technology for Private EVs. *IEEE Transactions on Transportation Electrification*, 9(1), 1845–1856. <https://doi.org/10.1109/TTE.2022.3163823>
- Lu, F., Zhang, H., & Mi, C. (2017). A review on the recent development of capacitive wireless power transfer technology. *Energies*, 10(11). <https://doi.org/10.3390/en10111752>
- Optimized, U. H., & Technology, I. (2024). *Hybrid PVP / Battery / Fuel Cell Wireless Charging Stations Electric Vehicles*.
- Pan, H., Qi, L., Zhang, X., Zhang, Z., Salman, W., Yuan, Y., & Wang, C. (2017). A portable renewable solar energy-powered cooling system based on wireless power transfer for a vehicle cabin. *Applied Energy*, 195, 334–343. <https://doi.org/10.1016/j.apenergy.2017.03.069>
- Popovic, Z., Falkenstein, E. A., Costinett, D., & Zane, R. (2013). Low-power far-field wireless powering for wireless sensors. *Proceedings of the IEEE*, 101(6), 1397–1409. <https://doi.org/10.1109/JPROC.2013.2244053>
- Rai, A., Singhal, A., & Tummuru, N. R. (2023). PV-Grid Integrated Hybrid Charger for EV s with Contactless and Conductive Charging Capability. *2023 IEEE IAS Global Conference on Renewable Energy and Hydrogen Technologies, GlobConHT 2023*, 1–6. <https://doi.org/10.1109/GlobConHT56829.2023.10087749>
- Subudhi, P. S., Padmanaban, S., Blaabjerg, F., & Kothari, D. P. (2023). Design and Implementation of a PV-Fed Grid-Integrated Wireless Electric Vehicle Battery Charger Present in a Residential Environment. *IEEE Journal of Emerging and Selected Topics in Industrial Electronics*, 4(1), 78–86. <https://doi.org/10.1109/JESTIE.2022.3195087>
- Triviño-Cabrera, A., González-González, J. M., & Aguado, J. A. (2020). *Wireless Power Transfer for Electric Vehicles: Foundations and Design Approach*. Springer International Publishing. <https://doi.org/10.1007/978-3-030-26706-3>

- Veza, I., Asy'ari, M. Z., Idris, M., Epin, V., Rizwanul Fattah, I. M., & Spraggon, M. (2023). Electric vehicle (EV) and driving towards sustainability: Comparison between EV, HEV, PHEV, and ICE vehicles to achieve net zero emissions by 2050 from EV. *Alexandria Engineering Journal*, 82(April), 459–467. <https://doi.org/10.1016/j.aej.2023.10.020>
- Zang, S., Hou, K., & Nguang, S. K. (2020). Underground Communications Using Capacitive Data Transfer Devices. *Journal of Sensors*, 2020. <https://doi.org/10.1155/2020/8849618>

## Chapter 3

### Hybrid Transformers: An Overview of Configurations and Converter Topologies

Yunus YALMAN<sup>1</sup> Mehmet Zahid EREL<sup>2</sup>

---

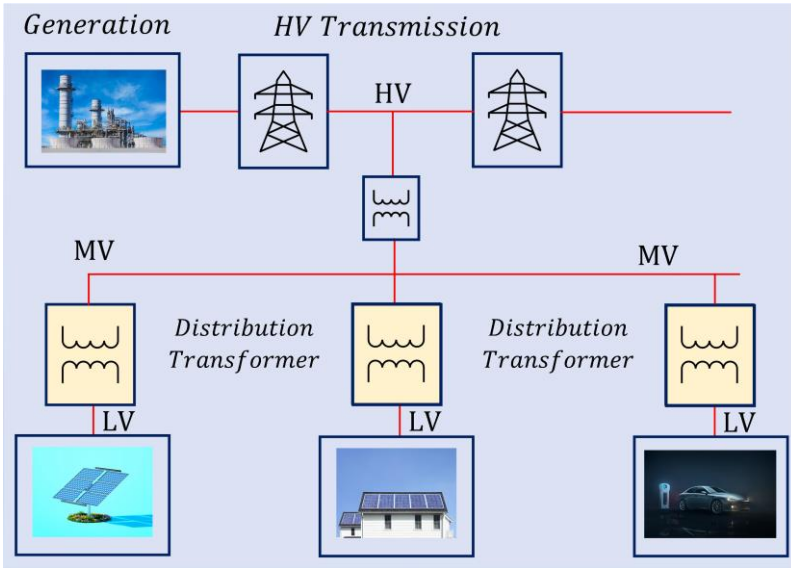
<sup>1</sup> Dr. Öğr. Üyesi, Ankara Yıldırım Beyazıt Üniversitesi, Elektrik-Elektronik Mühendisliği, Ankara, [Türkiye  
yunusyalman@aybu.edu.tr](mailto:yunusyalman@aybu.edu.tr), ORCID: 0000-0003-4792-1774

<sup>2</sup> Dr. Öğr. Üyesi, Ankara Yıldırım Beyazıt Üniversitesi, Enerji Sistemleri Mühendisliği, Ankara, Türkiye  
[mzerel@aybu.edu.tr](mailto:mzerel@aybu.edu.tr), ORCID: 0000-0003-1663-8394

## **1. Introduction**

Over the past two to three decades, the characteristics of electrical loads have experienced profound transformations. These changes are mainly due to progress in power electronics and, more significantly, the growing integration of renewable distributed generation sources like solar and wind power. However, during this period, the methodologies for controlling and operating distribution systems have largely remained unchanged. This disparity in development has led to unprecedented levels of stress on distribution grids. The substantial increase in distributed generation, particularly from stochastic sources such as rooftop solar panels, has considerably raised the stress on existing distribution systems. The projected integration of extra loads, such as electric vehicles (EVs), has the potential to further exacerbate the pressure on the distribution grid, potentially pushing it towards its operational limits (Bala et al., 2012; Erel et al., 2022; Simsek et al., 2021). These transformations address problems such as high/low voltage, harmonic amplification, three phase imbalance in distribution system (R. Li et al., 2024).

The electricity generated by power plants is conveyed to end-users through the distribution network as demonstrated in Figure 1. The primary component within the power system is the distribution transformer, which steps down the voltage from medium voltage (MV) to low voltage (LV) (Lee et al., 2022). The distribution transformer is a crucial component in the power system, with its safe and stable operation being directly tied to the overall safety and stability of the power system (Wan & Wong, 2023). While traditional transformers offer advantages such as low cost, high reliability, and high efficiency, their limited functionality is insufficient to meet the developmental needs of intelligent distribution substations (R. Li et al., 2024).



**Figure 1. The electrical power grid.**

There exist various power electronics converter-based solutions designed to solve problems in the distribution grid. Custom power devices which are dynamic voltage regulator (DVRs), distribution static synchronous compensators (DSTATCOMs) and unified power quality conditioners (UPQCs) have capability of voltage regulation and reactive power compensation stepless (Ghosh & Ledwich, 2002). Solid state transformers (SSTs) are emerging as a promising future solution in electrical systems, poised to replace conventional low-frequency transformers (LFTs). SSTs incorporate power converters at both the input and output stages to facilitate energy transfer through a medium-frequency transformer. Although SSTs have capability to completely control the voltage and current, they have some drawbacks like lower efficiency and high price (Huber & Kolar, 2019; Zheng et al., 2022). Hybrid Transformers (HTs) have been developed as a promising alternative to address the limitations of SSTs and to sustain controllability of voltage and current parameters (Marciel et al., 2024). HTs integrate traditional power frequency transformers with power electronic transformers to efficiently transmit energy within the electrical grid, utilizing power electronic devices to provide auxiliary services. Compared to SSTs, the control capabilities of systems based on the partial power converter concept are inherently restricted by the converter's capacity. This capacity typically represents only 10% to 20% of the transformer's rated power. For this reason, the construction cost is low and efficiency is high compared to the SSTs (Burkard & Biela, n.d.-b).

## 2. The Concept of Hybrid Transformer

General overview of HT is given in Figure 2. As shown in Figure 2, the HTs are composed of conventional LFT and power electronics converters that is typically in range of  $\pm 10\%$  of rating of the LFT rating (Bala et al., 2012; Burkard & Biela, n.d.-a). As the HT integrates the robustness, high efficiency, and cost-effectiveness of the LFT with the comprehensive controllability of a power electronic converter, it presents a promising concept for ensuring control of voltage, active and reactive power.

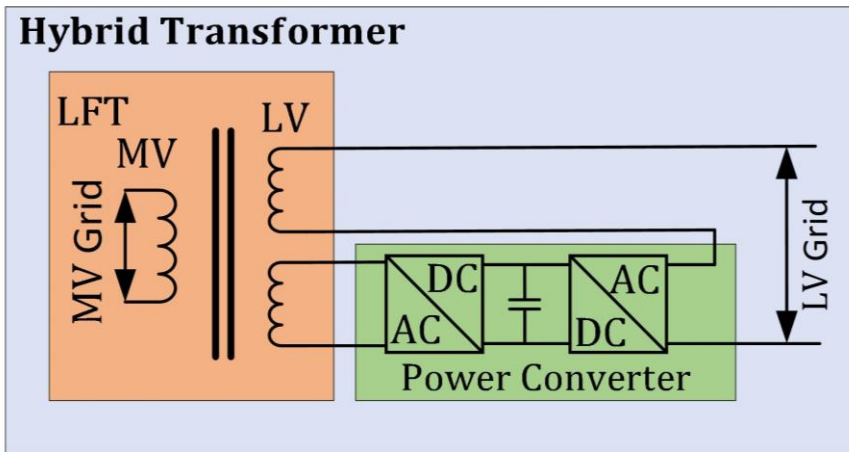


Figure 2. Concept of hybrid transformer

## 2. HT Configurations

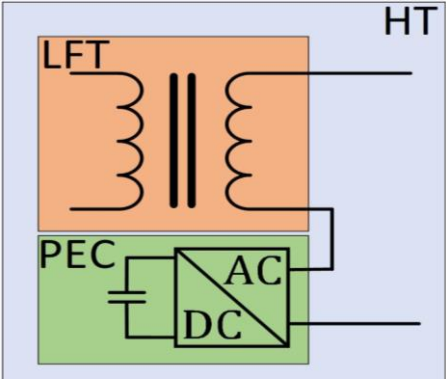
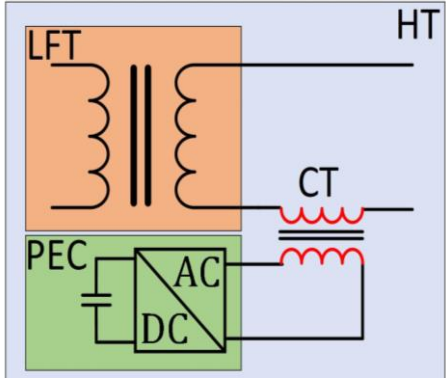
HT configurations are categorized based on the energy source for the power converter unit, whether it is derived from a capacitor, the primary or secondary winding of the LFT, or an auxiliary winding (AW). Also, each configuration is classified as series, shunt, series with coupling transformer (CT) and magnetic according to compensation type (Carreno et al., 2021). Function of HT's is based on power electronic converter (PEC) and connection type. While the parallel HTs can be used for power factor correction, filtering and reactive power compensation, series HTs can be utilized for voltage regulation and phase shifting. The series-parallel HTs has capable of regulating current and voltage(Wan & Wong, 2023) .

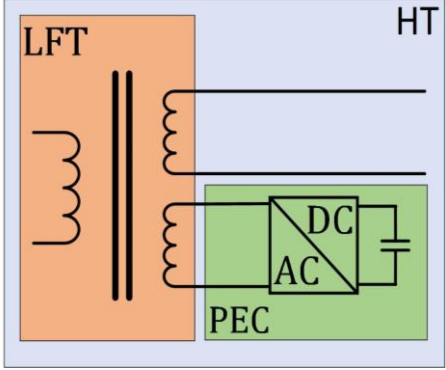
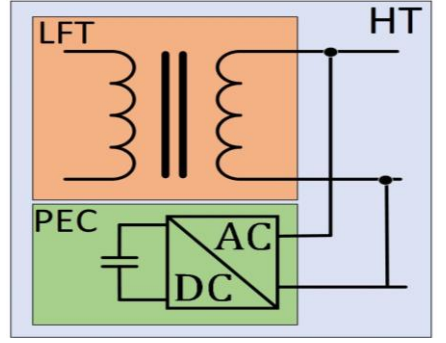


## 2.1 Self-Supported Hybrid Transformer

The configurations and compensation type of self-supported HTs are given in Table 1. They use capacitor as energy source. Irrespective of the power converter connection method, the power converter that used DC capacitors serve as the energy source can only provide reactive power compensation. To achieve capacitor voltage regulation and charging to its rated value, an additional active power control strategy is necessary (Carreno et al., 2021).

**Table 1. Classification and compensation type of self-supported HDT**  
(Carreno et al., 2021)

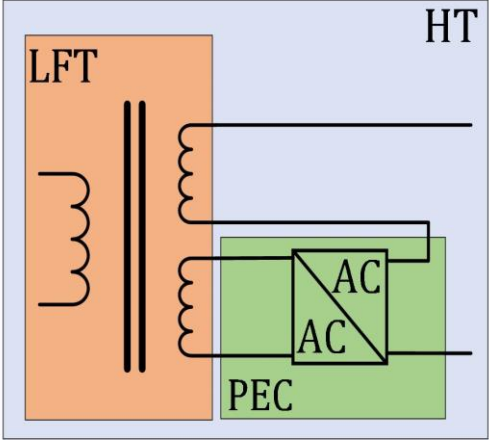
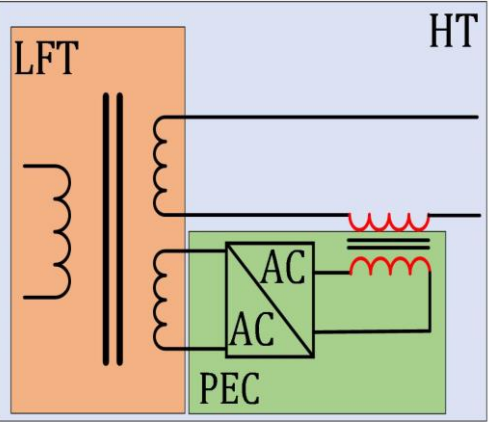
Configuration	Compensation Type	Properties
<p style="text-align: center;">PEC connected in series</p> 	Series	<ul style="list-style-type: none"> <li>• provide voltage regulation</li> <li>• using per phase power electronic converter (PEC) in the absence of a neutral point</li> </ul>
<p style="text-align: center;">PEC connected in series with CT</p> 	Series with CT	<ul style="list-style-type: none"> <li>• it is possible to use three phase PEC instead of single-phase configurations</li> <li>• having capability to connect MV grid thanks to CT (Newman et al., 2005a)</li> </ul>

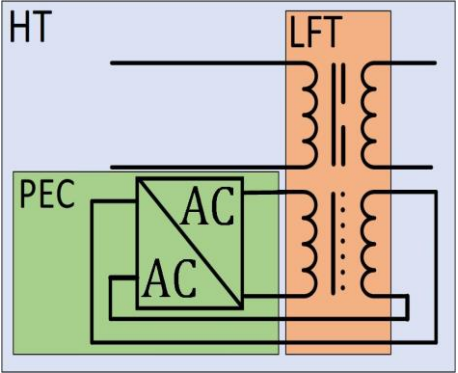
<p style="text-align: center;">PEC connected to the LFT core</p> 	Magnetic	<ul style="list-style-type: none"> <li>• mitigation of harmonic problems by tuned filter in auxiliary circuit (Y. Li et al., 2017)</li> </ul>
<p style="text-align: center;">PEC connected in shunt</p> 	Shunt	<ul style="list-style-type: none"> <li>• Integrated to LV grid with or without CTs(Sreenivasarao et al., 2012)</li> </ul>

## 2.2 Hybrid Transformers with Auxiliary Winding

The power converter unit uses AW as an energy source. The magnetic isolation is provided by auxiliary winding. The configuration and compensation type of hybrid transformer with AW are given in Table 2.

**Table 2. Classification and compensation type of HDT with Auxiliary Winding (Carreno et al., 2021)**

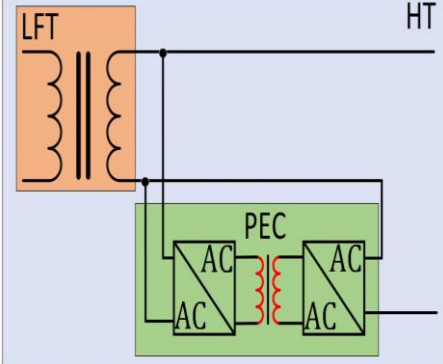
Configuration	Compensation Type	Properties
<p style="text-align: center;">PEC connected in series with AW</p> 	<p>Series and magnetic</p>	<ul style="list-style-type: none"> <li>• More common configuration</li> <li>• AC/DC/AC converters broaden regulation capacity of hybrid transformer (Wiemer &amp; Biela, 2019)</li> <li>• Mitigation of transformer inrush currents (Burkard &amp; Biela, 2017)</li> <li>• Adjusting input power factor and output voltage (Patents)</li> <li>• Provide voltage regulation (Radi et al., n.d.; Zheng et al., 2022)</li> </ul>
<p style="text-align: center;">PEC connected in series with AW and CT</p> 	<p>Series and magnetic</p>	<ul style="list-style-type: none"> <li>• Power converter is isolated by CT</li> <li>• Compound control system is developed to regulate grid current and load voltage (Liu et al., 2018, 2020).</li> </ul>

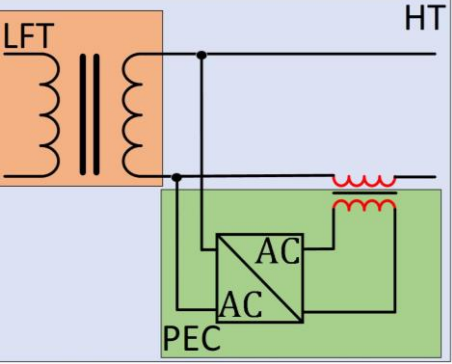
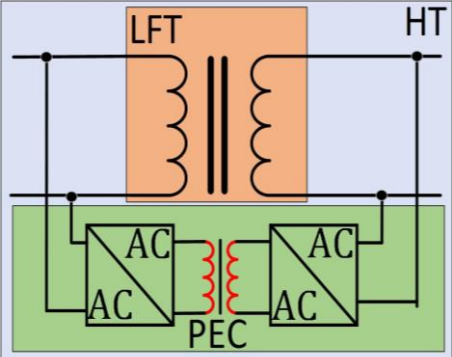
<p style="text-align: center;">PEC connected to the two AW</p> 	Magnetic	<ul style="list-style-type: none"> <li>• The power is supplied by AW. After processing, the power is injected to LFT via AW.</li> <li>• This configuration is used to increase controllability of distribution grid (Winter et al., 2019).</li> </ul>
--	----------	---

### 2.3 Hybrid Transformers connected to LFT windings

The energy of power converter is supplied by primary or secondary windings of LFT. Since power converter is directly connected to windings, isolation stage is required to obtain isolated voltage. The configuration and compensation of hybrid transformers connected to LFT windings is given Table 3.

**Table 3. Classification and compensation type of HDT connected to LFT Winding** (Carreno et al., 2021)

Configuration	Compensation Type	Properties
<p style="text-align: center;">PEC connected to the secondary-side and in series</p> 	Series and Shunt	<ul style="list-style-type: none"> <li>• Energy is injected to power converter via secondary winding.</li> <li>• Isolation stage is provided by high frequency transformers (HFT) (Huang et al., 2020).</li> </ul>

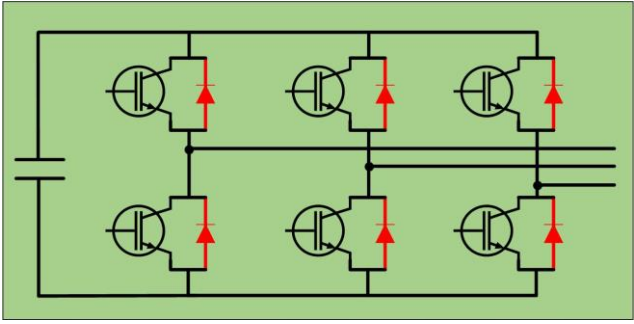
<p>PEC connected to the secondary-side and in series with CT.</p> 	<p>Shunt and Series with AW</p>	<ul style="list-style-type: none"> <li>• This configuration has capability to compensate the voltage and current in low voltage side (Pinto et al., 2016).</li> </ul>
<p>PEC connected to primary and secondary windings of the LFT in shunt</p> 	<p>Shunt</p>	<ul style="list-style-type: none"> <li>• The power converter is connected to both primary and secondary winding in shunt configuration.</li> <li>• The configuration is used for integration of renewable energy plants to improve distribution capacity (Zhu et al., 2017).</li> </ul>

### 3. Power Converter Topologies for Hybrid Transformer

The common power converter topologies used in HT are summarized in this section.

#### 3.1 Full Bridge Converter

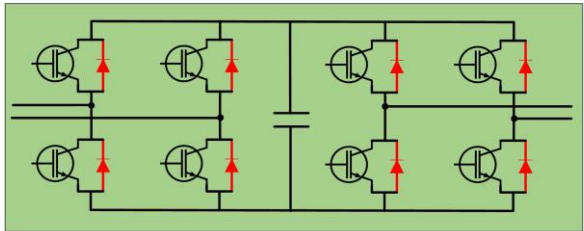
The converter model of full bridge converter (FBC) is shown in Figure 3. The FBC is used to mitigate the voltage disturbance in low voltage grid using supercapacitor. The isolation is provided by CT to connect the distribution line (Omar & Rahim, 2012). Three single phase full bridge converter is utilized to compensate voltage harmonic with minimizing sag compensation performance (Newman et al., 2005b).



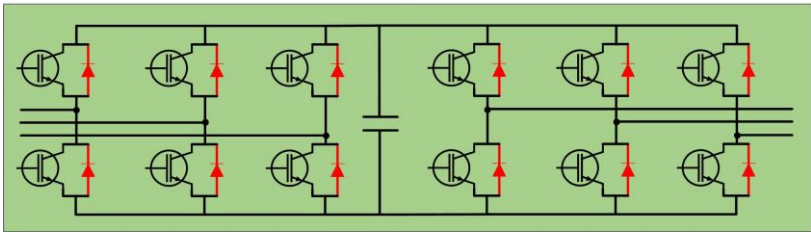
**Figure 3. Model of FBC**

### 3.2 Back-to-Back Converter

The model back-to-back (B2B) H bridge and full bridge is demonstrated in Figure 4. The B2B H bridge topology is proposed in HT transformer to compensate the voltage fluctuation. This topology is limited to single phase system (Power et al., 2021). The HT with B2B-FBC is proposed to regulate grid current and load voltage. In the controller the feedforwards loop is developed to weaken the disturbance level. Hence, robustness of system is improved (Liu et al., 2020).



a)



b)

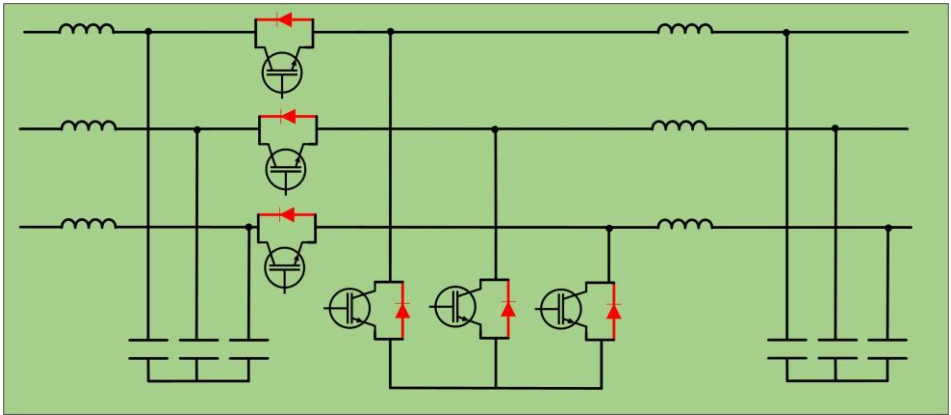
**Figure 4 a) Model of B2B H bridge converter b) Model of B2B FBC**

Hierarchical coordination control is developed for HDT to improve low voltage ride through ability. In this method voltage and current of each winding regulated and the B2B FBC is utilized in power converter side (Hu et al., 2024).

Multi-mode control, V-f control or P-Q control, of HT is suggested for voltage regulation and reverse power in active distribution network. The SiC-MOFET based B2B FB for wire grids is implemented in power converter side (Xu et al., 2024). The new type of HT is proposed to provide ancillary services to distribution system. The virtual synchronous machine control method is model in B2B converter for emulating virtual inertia as an ancillary service. At the same time, the proposed HT has capability such as reactive power injection and voltage regulation (Power et al., 2021). The model predictive strategies in conjunction with linear PI and PR control techniques to effectively manage the B2B converters in the HDT is proposed. Two separate control loop is implemented to regulate voltage and reactive power (Marciel et al., 2024).

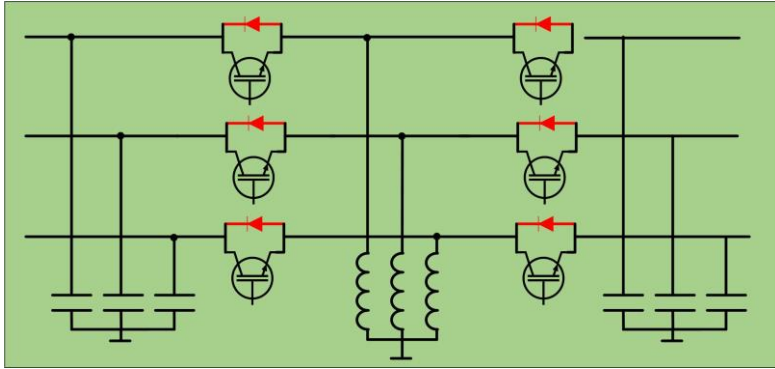
### 3.2 AC Chopper and Matrix Converter

The direct converter topologies used in HT that are AC chopper and Matrix converter are summarized. The general model of matrix converter is shown in Figure 5. The three phase hybrid transformer is designed and implemented using a matrix chopper converter to regulate voltage. The matrix converter is supplied from AW (Kaniewski, Fedyczak, et al., n.d.).



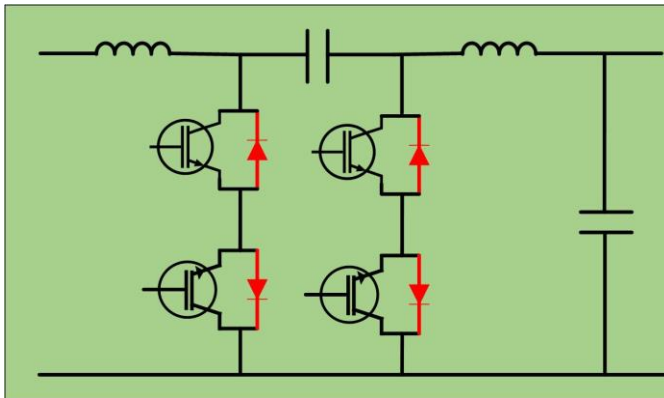
**Figure 5. Model of matrix converter**

The AC chopper with buck boost topology, shown in Figure 6, is developed in HT to regulate load voltage. The simulation and experimental studies are carried out for HT with AC chopper converter supplied by AW (Fedyczak et al., 2014).



**Figure 6. Model of AC chopper with buck-boost topology**

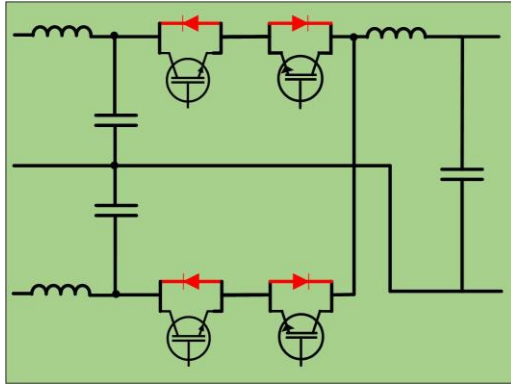
The single-phase HT is designed using matrix reactance chopper with Cuk topology. The AC chopper converter which is presented in Figure 7 is energized by AW of transformer. The simulation and experimental studied is realized to for validation of HT (Fedyczak et al., n.d.).



**Figure 7. Model of AC chopper with CUK topology**

The bipolar matrix chopper converter topology is proposed for HT to control voltage. The model of bipolar matrix chopper converter is given in Figure 8. The theoretical analysis, simulation and experimental study is carried out for 1 kVA laboratory model using open loop control. The advantage of the proposed HT is that output voltage can be controlled wide range (Kaniewski, Jarnut, et al., n.d.).





**Figure 8. Model of bipolar matrix chopper**

#### **4. Conclusion**

In recent times, power quality has emerged as a significant concern, primarily due to the increasing integration of renewable energy systems and electrical vehicles as well as the presence of nonlinear loads. The promising solution for enhancing power quality is HT thanks to dynamic control and cost efficient. In this study the configurations of HT are represented based on energy source of PEC. Energy of PEC can be supplied by capacitor, AW or LFT. The properties of each configuration are summarized. On the other hand, the compensation type for each configuration is presented. Moreover, the converter topologies used in PEC of HT are explained. The HT can provide ancillary services for power system since PECs are utilized in HT.

## References

- Bala, S., Das, D., Aeloiza, E., Maitra, A., & Rajagopalan, S. (2012). Hybrid distribution transformer: Concept development and field demonstration. *2012 IEEE Energy Conversion Congress and Exposition (ECCE)*, 4061–4068. <https://doi.org/10.1109/ECCE.2012.6342271>
- Burkard, J., & Biela, J. (n.d.-a). *Evaluation of Topologies and Optimal Design of a Hybrid Distribution Transformer*.
- Burkard, J., & Biela, J. (n.d.-b). *Hybrid Transformers for Power Quality Enhancements in Distribution Grids-Comparison to Alternative Concepts*.
- Burkard, J., & Biela, J. (2017). Transformer inrush current mitigation concept for hybrid transformers. *2017 19th European Conference on Power Electronics and Applications (EPE'17 ECCE Europe)*, P.1-P.9. <https://doi.org/10.23919/EPE17ECCEurope.2017.8099283>
- Carreno, A., Perez, M., Baier, C., Huang, A., Rajendran, S., & Malinowski, M. (2021). Configurations, power topologies and applications of hybrid distribution transformers. *Energies*, 14(5). <https://doi.org/10.3390/en14051215>
- Erel, M. Z., Bayindir, K. C., Aydemir, M. T., Chaudhary, S. K., & Guerrero, J. M. (2022). A Comprehensive Review on Wireless Capacitive Power Transfer Technology: Fundamentals and Applications. In *IEEE Access* (Vol. 10, pp. 3116–3143). Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/ACCESS.2021.3139761>
- Fedyczak, Z., Kaniewski, J., & Klyta, M. (n.d.). *Single-Phase Hybrid Transformer Using Matrix-Reactance Chopper with Ćuk Topology*.
- Fedyczak, Z., Kaniewski, J., Szczesniak, P., & Klytta, M. (2014). Modelling and analysis of three-phase hybrid transformer with buck-boost matrix-reactance chopper and active load. *2014 16th European Conference on Power Electronics and Applications*, 1–10. <https://doi.org/10.1109/EPE.2014.6910987>
- Ghosh, A., & Ledwich, G. (2002). Power Quality Enhancement Using Custom Power Devices. *Power Quality Enhancement Using Custom Power Devices*. <https://doi.org/10.1007/978-1-4615-1153-3>
- Hu, J., Lai, J., Yin, X., Yin, X., Zhou, K., Tang, A., & He, Q. (2024). Operation and Hierarchical Coordination Control of Integrated Hybrid Distribution Transformer under Grid Fault Conditions. *IEEE Transactions on Power Electronics*, 39(7), 8054–8071. <https://doi.org/10.1109/TPEL.2024.3388721>
- Huang, Q., Rajendran, S., Sen, S., Guo, Z., Zhang, L., & Huang, A. Q. (2020). 500kVA Hybrid Solid State Transformer (HSST): Architecture, Functionality and Control. *2020 IEEE Energy Conversion Congress and*

- Exposition* (ECCE), 4864–4871.  
<https://doi.org/10.1109/ECCE44975.2020.9235435>
- Huber, J. E., & Kolar, J. W. (2019). Applicability of Solid-State Transformers in Today's and Future Distribution Grids. *IEEE Transactions on Smart Grid*, 10(1), 317–326. <https://doi.org/10.1109/TSG.2017.2738610>
- Kaniewski, J., Fedyczak, Z., Klytta, M., Łukiewski, M., & Szcześniak, P. (n.d.). *Implementation of a Three-Phase Hybrid Transformer Using a Matrix Chopper*.
- Kaniewski, J., Jarnut, M., Szcześniak, P., & Fedyczak, Z. (n.d.). *The Study of Smart Distribution Transformer Based on a Bipolar Matrix Chopper*.
- Lee, H. J., Yoon, S. W., & Yoon, Y. D. (2022). Hybrid Distribution Transformer Based on an Existing Distribution Transformer and a Series-Connected Power Converter. *IEEE Transactions on Power Delivery*, 37(5), 4202–4211. <https://doi.org/10.1109/TPWRD.2022.3147820>
- Li, R., Liu, C., Guo, D., Gao, S., Yu, Q., & Mao, T. (2024). High-frequency isolated hybrid distribution transformer based on model predictive control. *2024 IEEE 10th International Power Electronics and Motion Control Conference, IPEMC 2024 ECCE Asia*, 1006–1011. <https://doi.org/10.1109/IPEMC-ECCEAsia60879.2024.10567494>
- Li, Y., Liu, Q., Hu, S., Liu, F., Cao, Y., Luo, L., & Rehtanz, C. (2017). A virtual impedance comprehensive control strategy for the controllably inductive power filtering system. *IEEE Transactions on Power Electronics*, 32(2), 920–926. <https://doi.org/10.1109/TPEL.2016.2601086>
- Liu, Y., Liang, D., Kou, P., Zhang, M., Cai, S., Zhou, K., Liang, Y., Chen, Q., & Yang, C. (2020). Compound Control System of Hybrid Distribution Transformer. *IEEE Transactions on Industry Applications*, 56(6), 6360–6373. <https://doi.org/10.1109/TIA.2020.3014058>
- Liu, Y., Liang, D., Liang, Y., Zhang, M., & Chen, Q. (2018). Design and Analysis of the Compounded Control System of Hybrid Distribution Transformer. *2018 IEEE Energy Conversion Congress and Exposition (ECCE)*, 3664–3668. <https://doi.org/10.1109/ECCE.2018.8558229>
- Marciel, E. I., Baier, C. R., Ramírez, R. O., Muñoz, C. A., Pérez, M. A., & Arevalo, M. (2024). Operation Assessment of a Hybrid Distribution Transformer Compensating for Voltage and Power Factor Using Predictive Control. *Mathematics*, 12(5). <https://doi.org/10.3390/math12050774>
- Newman, M. J., Holmes, D. G., Nielsen, J. G., & Blaabjerg, F. (2005a). A dynamic voltage restorer (DVR) with selective harmonic compensation at medium voltage level. *IEEE Transactions on Industry Applications*, 41(6), 1744–1753. <https://doi.org/10.1109/TIA.2005.858212>
- Newman, M. J., Holmes, D. G., Nielsen, J. G., & Blaabjerg, F. (2005b). A dynamic voltage restorer (DVR) with selective harmonic compensation at medium

- voltage level. *IEEE Transactions on Industry Applications*, 41(6), 1744–1753. <https://doi.org/10.1109/TIA.2005.858212>
- Omar, R., & Rahim, N. A. (2012). Voltage unbalanced compensation using dynamic voltage restorer based on supercapacitor. *International Journal of Electrical Power and Energy Systems*, 43(1), 573–581. <https://doi.org/10.1016/j.ijepes.2012.05.015>
- Pinto, S. F., Alcaria, P., Monteiro, J., & Silva, J. F. (2016). Matrix Converter-Based Active Distribution Transformer. *IEEE Transactions on Power Delivery*, 31(4), 1493–1501. <https://doi.org/10.1109/TPWRD.2016.2530635>
- Power, R., Mithani, A., Madawala, U., & Baguley, C. (2021). A Hybrid Transformer Topology for Distribution Network Voltage Regulation. *2021 IEEE Southern Power Electronics Conference, SPEC 2021*. <https://doi.org/10.1109/SPEC52827.2021.9709487>
- Radi, M. A., Darwish, M., & Alqarni, M. (n.d.). *Voltage Regulation Considerations for the Design of Hybrid Distribution Transformers*.
- Simsek, S., Uslu, S., Sahin, M., Arlı, F., & Bilgic, G. (2021). Impact of a novel fuel additive containing boron and hydrogen on diesel engine performance and emissions. *Energy Sources, Part A: Recovery, Utilization and Environmental Effects*. <https://doi.org/10.1080/15567036.2021.1946621>
- Sreenivasarao, D., Agarwal, P., & Das, B. (2012). Neutral current compensation in three-phase, four-wire systems: A review. *Electric Power Systems Research*, 86, 170–180. <https://doi.org/10.1016/J.EPSR.2011.12.014>
- Wan, X., & Wong, M. C. (2023). Review of Hybrid Transformer Topology. *2023 IEEE PELS Students and Young Professionals Symposium, SYPS 2023*. <https://doi.org/10.1109/SYPS59767.2023.10268179>
- Wiemer, A., & Biela, J. (2019). Comparison of Hybrid Transformers with Uni-And Bidirectional Auxiliary Converter. *2019 21st European Conference on Power Electronics and Applications, EPE 2019 ECCE Europe*. <https://doi.org/10.23919/EPE.2019.8915576>
- Winter, P., Cajigal-Nunez, J. M., Wrede, H., & Schnepf, J. (2019). New topology and functionalities of a hybrid transformer for flexible operation of distribution and transmission systems. *2019 21st European Conference on Power Electronics and Applications, EPE 2019 ECCE Europe*. <https://doi.org/10.23919/EPE.2019.8915116>
- Xu, X., Zhang, T., Qiu, Z., Gao, H., Yu, H., Ma, Z., & Zhang, R. (2024). Multi-Mode Control of a Hybrid Transformer for the Coordinated Regulation of Voltage and Reverse Power in Active Distribution Network. *Processes*, 12(2). <https://doi.org/10.3390/pr12020265>

- Zheng, L., Marellapudi, A., Chowdhury, V. R., Bilakanti, N., Kandula, R. P., Saeedifard, M., Grijalva, S., & Divan, D. (2022). Solid-State Transformer and Hybrid Transformer With Integrated Energy Storage in Active Distribution Grids: Technical and Economic Comparison, Dispatch, and Control. *IEEE Journal of Emerging and Selected Topics in Power Electronics*, *10*(4), 3771–3787. <https://doi.org/10.1109/JESTPE.2022.3144361>
- Zhu, R., De Carne, G., Deng, F., & Liserre, M. (2017). Integration of Large Photovoltaic and Wind System by Means of Smart Transformer. *IEEE Transactions on Industrial Electronics*, *64*(11), 8928–8938. <https://doi.org/10.1109/TIE.2017.2701758>

## Chapter 4

# A Flexible and Energy-Conscious Dynamic Encryption Approach For Resource-Constrained Iot and Embedded Devices<sup>1</sup>

**Cemil Baki KIYAK<sup>2</sup>, Fadi YILMAZ<sup>3</sup>,  
Hasan Şakir BİLGE<sup>4</sup>**

---

<sup>1</sup> This article is derived from the doctoral dissertation titled “Designing Advanced Encryption Methodology on FPGA”

<sup>2</sup> Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Yenimahalle, Ankara/TR  
Ostim Teknik Üniversitesi, MYO, Hibrid ve Elektrikli Taşıtlar Teknolojisi Programı, OSTİM OSB, Ankara/TR  
ORCID: 0000-0002-3479-2048

<sup>3</sup> Ankara Yıldırım Beyazıt Üniversitesi, Mühendislik ve Doğa Bilimleri Fakültesi,  
Elektrik-Elektronik Mühendisliği Bölümü, Keçiören, Ankara/TR  
ORCID: 0000-0002-3591-3606

<sup>4</sup> Gazi Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Çankaya, Ankara/TR  
ORCID: 0000-0002-4945-0884

## **Abstract**

This paper introduces an energy-efficient and secure encryption approach designed for IoT and embedded systems with limited processing capabilities and energy budgets. Traditional protocols, such as Diffie-Hellman, often prove unsuitable in these constrained environments due to their computational intensity and high-power consumption. The proposed method integrates dynamic key generation, lower-bit AES and MD5 algorithms, and conceptually incorporates hardware-level obfuscation strategies to balance security and efficiency. A concise literature review highlights the pitfalls of conventional methods ‘spanning from high-cost encryption protocols to vulnerabilities like malicious hardware attacks and side-channel threats’ emphasizing the need for adaptable, lightweight solutions. Building on these insights, the study demonstrates, through simulations conducted solely in the Octave environment, that dynamically generated keys for each message can significantly reduce processing time and energy usage compared to Diffie-Hellman. While the core validation is performed via software simulations, the approach suggests that advanced techniques, such as dynamic RAM relocation or other hardware-based reconfiguration strategies, could further enhance security against side-channel and dictionary attacks if implemented on actual devices in the future. The results highlight the proposed method’s robustness and efficiency, offering guidance for integrating hardware-software measures without excessively burdening energy resources. In conclusion, this research lays a foundation for a practical, energy-conscious encryption framework that can be readily adapted and enhanced, serving as a point of reference for future work involving both software optimizations and potential hardware-level defenses.

## **Keywords**

IoT Security, Lightweight Cryptography, Dynamic Key Regeneration, Energy-Constrained Devices, Hardware-Software Co-Design, Side-Channel Resistance, Resource-Efficient Encryption Techniques

## **1. Introduction**

A good encryption algorithm must remain resistant to decryption attempts even if its underlying principles are fully known (Stallings, 2016; NIST, 2016; Kerckhoffs, 1883). Today's rapidly expanding IoT ecosystem introduces numerous devices with limited energy and processing capabilities, placing stringent constraints on data security methods. Although widely accepted cryptographic protocols, such as those based on Diffie-Hellman, ensure robust security, they often impose high computational and energy costs, making them impractical for many IoT applications.

As IoT networks continue to grow, the demand for low-energy, low-cost, yet adequately secure communication techniques has intensified. While traditional methods maintain high security levels, their resource demands do not align well with the strict limitations of embedded and IoT devices. This calls for innovative approaches that strike a balance between security and efficiency.

### **1.1 Importance of Secure Communication in IoT and Embedded Systems**

The proliferation of IoT devices (Stallings, 2016) has made secure communication a critical topic. These devices, often constrained in power and processing, are susceptible to attacks that can compromise network integrity and data confidentiality. Ensuring security within such limitations requires solutions that are both computationally and energetically efficient (Schneier et al., 1999; NIST, 2016). However, achieving robust security without significantly increasing energy consumption remains a key challenge.

### **1.2 Disadvantages of Existing Encryption Methods**

Many well-known encryption protocols, including Diffie-Hellman (Diffie & Hellman, 1976), are computationally intensive and demand substantial energy. This is problematic for IoT devices, which must conserve resources for longevity and reliability (Stallings, 2016). Literature also indicates vulnerability to side-channel attacks and hardware manipulations (Stallings, 2016), underscoring the need for approaches that do not rely solely on traditional high-bit, high-cost encryption.

Considering simplified cryptographic schemes can improve understandability and reduce overhead (Raeburn, 2005), but these lighter methods must still resist common attack vectors. Conventional asymmetric algorithms like Diffie-Hellman and RSA incur high computational burdens unsuitable for energy-limited environments (Rivest, Shamir & Adleman, 1978; Nguyen et al., 2023). Even techniques like AES and SHA must be carefully adapted to low-bit versions



to minimize energy consumption, yet additional safeguards against side-channel attacks then become necessary (Soliman, 2019).

### 1.2.1 Diffie-Hellman Key Exchange Protocol

Diffie-Hellman is a foundational key exchange method (Diffie & Hellman, 1976) that enables two parties to share a secret key over insecure channels. While effective in principle, its high computational requirements and energy usage make it less suitable for IoT scenarios (Nguyen et al., 2023). Further, side-channel vulnerabilities can lead to security breaches if not properly mitigated (Soliman, 2019).

### 1.2.2 Related Works

- **RSA Encryption:** Although RSA is widely employed (Rivest, Shamir & Adleman, 1978), its generation and handling of large prime numbers can burden IoT devices with excessive energy consumption and processing delays.
- **Elliptic Curve Cryptography (ECC):** ECC reduces key sizes (Miller, 1986), but can still present non-trivial energy overheads for very constrained devices (Nguyen et al., 2023).
- **AES and SHA Algorithms:** While AES and SHA (NIST, 2001; NIST, 1995) can be adapted for lower cost, doing so generally requires trade-offs in key length and necessitates extra protection against side-channel attacks.

### 1.2.2 Position of the Proposed Method in the Literature

Compared to these established protocols, the proposed approach exploits low-bit AES and MD5, coupled with dynamic key generation for each message (Nguyen et al., 2023). This hybrid strategy aims to overcome the heavy resource demands of Diffie-Hellman and ECC while improving resistance to side-channel attacks via dynamic RAM positioning (Samir et al., 2019). Leveraging PUF-based Device IDs, timestamps, and random word lists, the method strives for both time and energy efficiency without sacrificing essential security properties.

## 1.3 Literature Review

The literature addresses various hardware and software countermeasures to enhance encryption in resource-limited devices. Hardware Trojan (HTH) attacks exploit vulnerabilities during design or supply chains, affecting FPGA-based systems (Johnson et al., 2017). Malicious bitstream manipulation and clock signal tampering (Karam et al., 2016) highlight how hardware-level assaults can bypass

standard encryption layers. Side-channel attacks leverage physical emissions (Karam et al., 2016), necessitating integrated protective measures.

Efforts like the LifeLine system (Stolz et al., 2021), algorithm hopping (Soliman et al., 2019), dynamic reconfigurable PUFs (Al-Meer & Al-Kuwari, 2023), and lightweight hardware security modules (Samir et al., 2019) attempt to strengthen security. However, many require additional energy or complex hardware resources, challenging their suitability for IoT devices. Intellectual property protection schemes (Adetomi et al., 2017) and the secure use of PUFs (Ning, Farha, Ullah, & Mao, 2020) also play a role, but must be balanced against practical constraints.

Kerckhoffs's Principle (Kerckhoffs, 1883) underpins the idea that security should not depend on the obscurity of the algorithm itself. Building on these insights, the proposed method incorporates dynamic key generation, low-bit cryptographic operations, and hardware/software integration to provide a balanced solution. It seeks to simultaneously achieve lower energy usage, reduced processing time, and heightened resilience against a range of attacks, positioning it as a viable alternative for secure, energy-conscious IoT communication.

#### **1.4. Aim and Contributions of the Study**

The primary objective of this study is to provide an energy-efficient, secure, and flexible encryption framework specifically tailored to the constraints of IoT devices. Unlike conventional methods that often demand high computational resources or fail to adequately address hardware-level manipulations, the proposed approach integrates dynamic key generation, lightweight cryptographic primitives, and adaptive hardware-software interplay.

##### **Key contributions include:**

- **Energy-Conscious Dynamic Key Generation:**

By creating unique keys for each message 'using random word selections, timestamps, and MD5' the method achieves a secure encryption process that is less computationally intensive than traditional protocols. These software-level optimizations allow for reduced processing times and lower energy consumption, making the approach well-suited for resource-constrained IoT environments.

- **Multi-Layered Security Through Hardware-Software Integration:**

The method is designed with the potential for hardware-software integration. While this study focuses on software-based validation in Octave, it contemplates future deployment of hardware-level strategies, such as dynamic RAM relocation, to obscure sensitive data like the word list. This forward-looking

perspective paves the way for incorporating hardware-based obfuscation to strengthen resistance against side-channel vulnerabilities if integrated into actual systems.

- **Mitigating Identified Weaknesses:**

Recognizing that static storage of critical information can be a liability, the proposed solution advocates for continuous reconfiguration of memory spaces, potentially through methods like Dynamic Function eXchange (DFX). Although not implemented here, such a future enhancement would force attackers to contend with constantly shifting key materials, bolstering resilience against dictionary and brute-force attacks.

- **Adaptability and Platform Independence:**

The algorithm's reliance on lightweight cryptographic operations and dynamic key generation ensures easy adaptation to various platforms. While current validation is confined to a software simulation environment, the approach anticipates smooth transitions to different hardware architectures. This flexibility aims to bridge the gap between stringent security requirements and the need for energy efficiency, facilitating broader adoption in evolving IoT ecosystems.

In sum, this study strives to strike an optimal balance between security and resource utilization. By integrating dynamic key generation, low-bit cryptographic operations, and strategic hardware-level rearrangements, it offers a secure, efficient, and flexible encryption mechanism for IoT and embedded systems. This method not only addresses known shortcomings in existing solutions but also sets a reference point for achieving robust, energy-conscious security in future research and applications.

## **2. Materials and Methods**

### **2.1. Simulation of the Algorithm with Octave and Comparison with Diffie-Hellman**

To assess the performance and energy efficiency of the proposed encryption method, simulations were conducted in the GNU Octave environment. The focus was on comparing the dynamic key generation approach against the Diffie-Hellman (DH) key exchange protocol under similar conditions. The goal was to evaluate processing time, energy consumption, and the feasibility of employing dynamic keys in resource-constrained settings.

#### **Key Generation and Message Encryption Steps of the Algorithm**

The proposed method generates a unique key for each message by combining a randomly selected word from a predefined list with the current timestamp and

then hashing this combination using MD5. The resulting hash serves as the encryption key for AES-128:

- **Key Generation:**

$$Key = MD5(Word \parallel Timestamp) \quad (1)$$

Here, “Word” is a randomly chosen entry from the word list, and “Timestamp” is the current time or a nonce that increments per message.

- **Message Encryption/Decryption:**

$$Encrypted\ Message = AES-128(Message, Key) \quad (2)$$

This dynamic approach ensures that each message employs a distinct key, thereby increasing security and reducing predictability compared to methods relying on a fixed key.

### **Initial Communication and Continuous Key Generation**

During the initial communication phase, both devices use a common word list and synchronized timestamps. A secure initial key is derived once each device identifies the correct random word-timestamp pair. After establishing this baseline, subsequent messages are always encrypted with newly generated keys. The Device ID, created during the initial setup using unique identifiers, remains constant during continuous operation but does not factor into the resource comparison, as it is generated only once. This dynamic key mechanism reduces reliance on large primes and expensive modular arithmetic required by DH (Diffie & Hellman, 1976), thus offering a more resource-friendly solution.

### **Simulation Procedure and Results**

In the simulation, both the proposed method and DH were applied to messages of equal size. DH’s reliance on computationally intensive operations resulted in longer processing times and higher energy estimates, as it involves large prime computations and modular exponentiation. In contrast, the proposed method’s focus on lightweight, dynamic key generation (MD5 + AES-128) enabled faster encryption-decryption cycles.

### **Summary of Findings:**

- **Processing Time:**

The proposed method demonstrated significantly reduced processing time compared to DH, allowing more messages to be handled within the same time frame.

- **Energy Efficiency:**

By avoiding energy-intensive computations, the proposed method estimated lower energy usage than DH, aligning better with the constraints of low-power IoT devices.

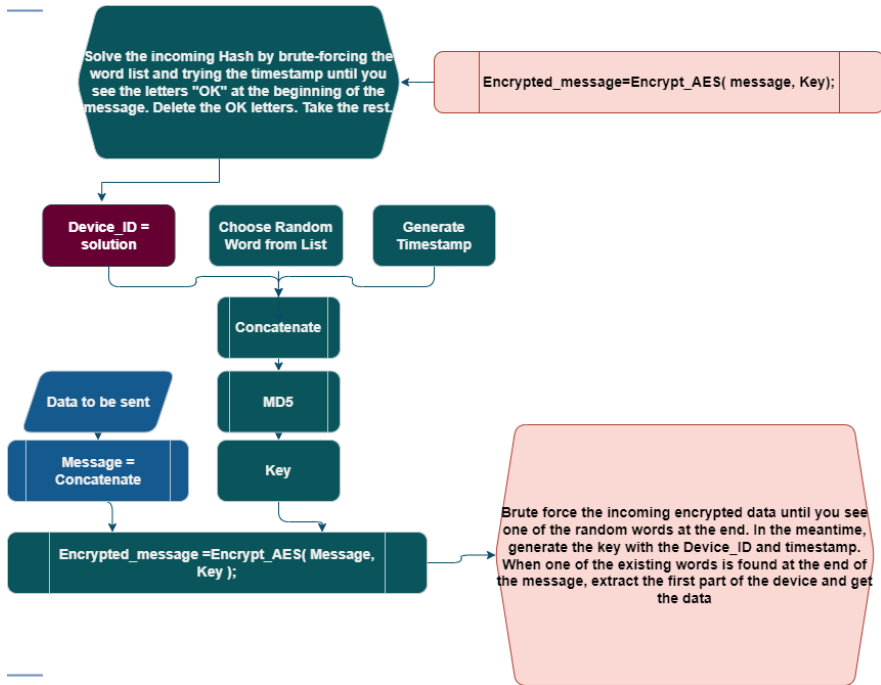
### **2.1.1. Experimental Setup**

Simulations were performed on a standard computing environment running Windows 10 and GNU Octave 5.2.0. Intel Power Gadget 3.6 was used for monitoring power consumption. Both methods were tested under identical conditions to ensure a fair comparison.

### **2.1.2. Simulation Flow**

The simulation encompassed the following steps:

1. **Random Word Selection:** A random word is chosen from the word list.
2. **Timestamp Acquisition:** The current timestamp is recorded.
3. **Key Generation via MD5:** The chosen word and timestamp are concatenated and hashed with MD5, producing a unique 128-bit key for that message.
4. **Message Encryption:** AES-128 encryption is applied using this key.
5. **Decryption and Verification:** On the receiving end, the same word list and timestamp are used to identify the correct key, decrypting the message successfully.



**Figure.1.** Flowchart illustrating the dynamic key generation process for each message through the continuous communication procedure of the proposed encryption algorithm.

During continuous communication, keys are regenerated for each message, ensuring that even if one key is compromised, subsequent messages remain secure.

### 2.1.3. Comparison with Diffie-Hellman

To compare the proposed method with DH, both were executed under the same environment. DH's large prime computations and exponential operations led to longer processing times and, consequently, higher estimated energy consumption. In contrast, the proposed approach, through its simpler dynamic key mechanism, was able to handle more encrypted messages within the same time frame, demonstrating advantages in speed and reduced resource usage.

### 2.1.4. Energy Consumption Estimates

Energy consumption estimates were based on differences in processing times and measured power usage. While DH's complexity increases energy demands, the proposed method's streamlined operations translate into lower estimated energy

costs. This indicates its potential for real-world IoT implementations where power availability is limited.

$$\text{Energy Consumption (Wh)} = \text{Average Power (W)} \times \text{Processing Time (s)} / 3600 \quad (3)$$

### **3. Experimental Results and Comparisons**

#### **3.1. Experimental Setup**

All experiments were conducted exclusively in a software-based simulation environment using GNU Octave, focusing on the dynamic key generation and encryption-decryption cycles of the proposed method. Within this environment, the proposed method and the Diffie-Hellman (DH) protocol were tested under the same conditions, allowing an objective evaluation of their execution times and estimated energy consumption.

A dictionary attack scenario was also simulated in Octave. This allowed us to approximate how quickly (or slowly) an attacker might guess the correct key given the word list and timestamp-based dynamic key generation. The results from these simulations are presented as tables and figures, each placed immediately after their mention for ease of reference.

While no hardware (e.g., FPGA) implementation was carried out in this study, we highlight in a later subsection how future work could integrate dynamic RAM relocation or DFX strategies. Such measures would potentially enhance resistance against side-channel or dictionary attacks.

#### **3.2. Performance Comparisons in the Octave Environment**

This section compares the proposed encryption method with the DH protocol by measuring their processing times and estimating energy consumption. The analyses focus on how quickly keys can be generated, how efficiently messages can be encrypted and decrypted, and how these factors translate into approximate energy usage.

##### **3.2.1. Processing Time Comparisons**

To quantify performance, both algorithms were subjected to 100 iterations of key generation, encryption, and decryption tests in Octave. Diffie-Hellman, with its reliance on large prime computations and modular arithmetic, consistently required more processing time. In contrast, the proposed method ‘utilizing random words from a predefined list, timestamps, and MD5 hashing’ achieved faster key generation and encryption cycles. The figures and tables below present a detailed overview of the measured times:

Method	Encryption - Decryption (ms)	
Diffie-Hellman	5798.8	187.16
Proposed Method	106.48	2012.4

**Table 1.** Comparison of Execution Times in Octave Environment.

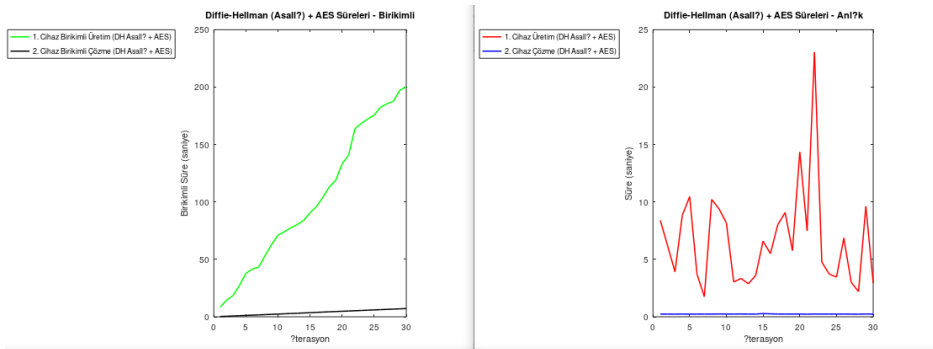
```

Command Window
--- Results for 100 Iterations with 15 Words: ---
1. Diffie-Hellman Method Results:
   Average Key Generation Time (Device 1): 5.7988 seconds
   Average Decryption Time (Device 2): 0.18716 seconds
2. MD5-Based Method Results:
   Average Key Generation Time (Device 1): 0.10648 seconds
   Average Decryption Time (Device 2): 2.0124 seconds
3. Cumulative Time Comparison:
   Number of messages that can be sent using the MD5-based method
   in the time it takes the DH method to send 100 messages: 282
>> |

```

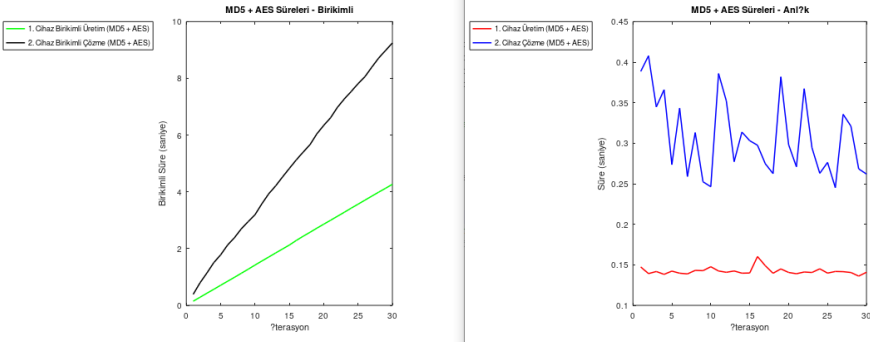
**Figure.2.** Visualization of average generation and decryption times obtained in the Octave environment for Diffie-Hellman and the MD5-based proposed method.

The results in Table 1 and Figure 2 illustrate that the proposed method can produce keys and handle encryption-decryption much faster than DH. Although the proposed approach requires the receiver to test keys from the word list to decrypt messages, these lookups are still orders of magnitude faster than the expensive computations in DH. Additional figures highlight cumulative and instantaneous time comparisons for both algorithms:



**Figure.3.** (a) Cumulative Graph of Diffie-Hellman Key Generation + AES Encryption and Decryption, (b) Instantaneous Graph of Diffie-Hellman Key Generation + AES Encryption and Decryption.

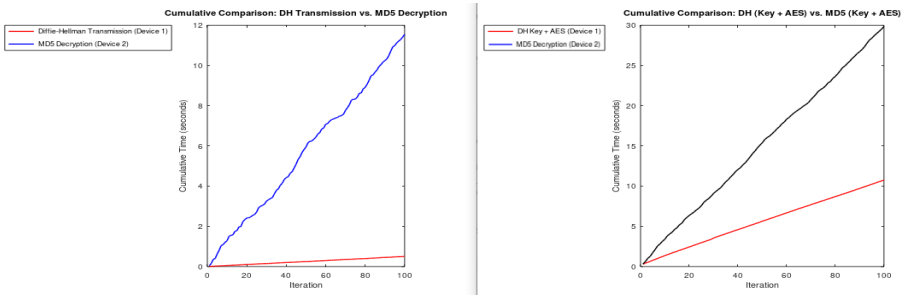




**Figure.4.** (a) Cumulative Graph of the Proposed Method Key Generation + AES Encryption and Decryption, (b) Instantaneous Graph of the Proposed Method Key Generation + AES Encryption and Decryption.

These graphs (Figures 3 and 4) visualize how DH’s processing time accumulates substantially over multiple iterations, whereas the proposed method maintains relatively low and stable execution times.

To further emphasize the performance advantage, we present direct comparisons of key generation times and combined operations:



**Figure.5.** (a) Comparison of key generation and decryption times between the proposed method and DH. (b) Comparison of key generation + AES encryption and decryption times between the method and DH.

Figure 5 shows that the proposed method can generate keys and perform encryption-decryption in a fraction of the time taken by DH. Even when accounting for the receiver’s need to find the correct word from the list, the total time remains significantly lower.

Finally, we examine larger iterations and word list sizes:

```

--- Results for 1000 iterations with a 50-word list: ---
1. Diffie-Hellman Method Results:
  Average Key Generation Time (Device 1): 5.955 seconds
  Average Decryption Time (Device 2): 0.1817 seconds
2. MD5-Based Method Results:
  Average Key Generation Time (Device 1): 0.10366 seconds
  Average Decryption Time (Device 2): 6.2093 seconds
3. Cumulative Time Comparison:
  Number of messages that can be sent using the MD5-based method
  during the time taken to send 1000 messages with the DH method: 972
>>

```

**Figure.6.** Similar processing times observed in a 1000-iteration test with a 50-word list.

Figure 6 demonstrates that even when the word list size increases to 50 and the number of iterations grows to 1000, the proposed method’s performance remains comparatively sufficient. Although the DH protocol can be optimized by relaxing certain cryptographic criteria, it still cannot match the speed of the lightweight approach outlined here.

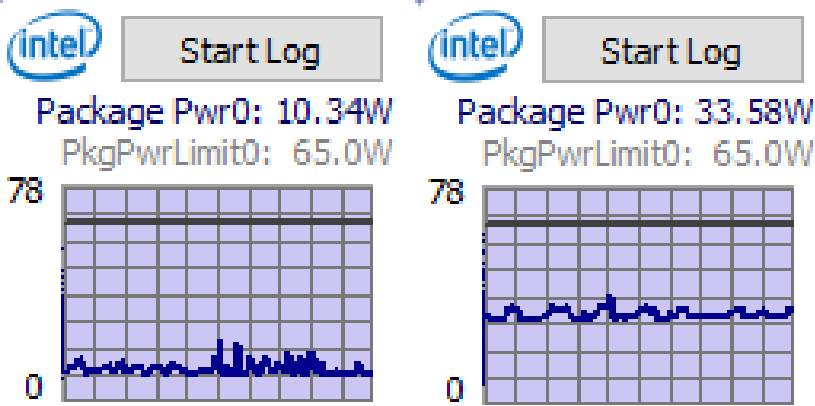
### 3.2.2. Energy Consumption Estimates

Energy consumption is estimated based on execution times and a hypothetical constant system power usage. The difference in processing speeds directly affects energy estimates, with DH’s lengthy computations leading to higher energy usage compared to the quicker, lighter operations of the proposed method.

We present the following data to illustrate estimated energy consumption:

Method	Estimated Energy Consumption (Wh)
Diffie-Hellman	0.03824363
Proposed Algorithm	0.00058858

**Table.2.** Comparison of Estimated Energy Consumption in Octave Environment.



*Figure.7. Windows power consumption when (a) the system is idle and (b) the system is under load.*

Table 2 and Figure 7 provide insight into how shorter processing times translate into measurable energy savings. By reducing key generation complexity and relying on faster hash-based operations, the proposed method yields approximately 64.5% lower estimated energy consumption. While these energy values are not obtained from actual hardware, they offer a useful benchmark, indicating that a more efficient cryptographic approach can extend battery life and reduce costs in real-world IoT deployments.

### **3.3. Potential Future Hardware-Level Enhancements**

Although the results presented above are based solely on Octave simulations, future studies might integrate hardware-level techniques, such as dynamic RAM relocation or DFX, to improve security further. By continuously changing the memory location of the word list and other critical data, it becomes harder for attackers to exploit side-channel attacks or predict key-related materials.

For illustrative purposes, the following figures and tables are related to a hypothetical FPGA implementation scenario, though not part of the current study's practical results. In a future extended version of this research, these hardware-level measurements and setups (initially considered on platforms like Snickerdoodle Black FPGA, Arduino DUE, and related toolchains) could be used to validate and enhance the proposed approach.

### **Example Vitis C Code of Encryption for ZYNQ 7020 FPGA:**

```
XTime tStart;
XTime_GetTime(&tStart);

u32 timestamp = nonce + (u32)(tStart / (COUNTS_PER_SECOND /
1000));
int random_index = rand() % KEY_ARRAY_SIZE;
char* random_key = key_array[random_index];

char key[2 * AES_KEY_SIZE + 1];
sprintf(key, "%lu", timestamp);
strcat(key, random_key);

// Encryption
char* encrypted_message = a_aes_encode_to_string("Test
Message", key);
printf("Encrypted Message: %s\n", encrypted_message);
```

### **Example C++ Code of Decryption for Arduino DUE:**

```
#include <MD5.h> // MD5 library
#include <AES.h> // AES library

// Example word list
const char* word_list[] = {"apple", "pear", "cherry",
"banana", "watermelon"};
const int word_list_size = 5;

// Encrypted message (example)
const char* encrypted_message = "E4ACF33B21D5E..."; //
Example AES-128 encrypted message

// Timestamp and key generation
char key[32];

void setup() {
  Serial.begin(9600);
  Serial.println("Starting the decryption process...");

  for (int i = 0; i < word_list_size; i++) {
    for (unsigned long timestamp = 1638300000; timestamp <
1638301000; timestamp++) {
      // Key generation: word + Timestamp
```

```

    snprintf(key, sizeof(key), "%s%lu", word_list[i],
timestamp);

    // Generate the key using MD5 hash
    MD5 hasher;
    hasher.begin();
    hasher.add(key);
    hasher.calculate();
    char* md5_key = hasher.getHexDigest();

    // Decryption attempt
    AES aes;
    byte decrypted_message[16];
    aes.do_aes_decrypt((byte*)encrypted_message, 128,
(byte*)md5_key, decrypted_message);

    // Check the decrypted message
    if (strcmp((char*)decrypted_message, "OK") == 0) {
        Serial.print("Decryption succeeded! Key: ");
        Serial.println(key);
        return;
    }
}

Serial.println("Decryption failed.");
}

void loop() {
    // The loop is empty; operations occur in setup().
}

```

### 3.4. Security Analysis and Attack Scenarios

The Octave simulations indicate that the proposed dynamic key generation method, coupled with timestamps and a manageable word list, increases the difficulty for attackers to conduct dictionary or brute force attacks efficiently. By reducing the computational complexity on the producer's side and distributing it across a limited word list search on the consumer's side, the method strikes a practical balance between security and performance.

In the future, integrating hardware-level measures like dynamic RAM relocation could further complicate attack strategies, as attackers would not only have to guess the correct key but also contend with an ever-shifting memory map. Although these enhancements are not part of the current simulation-based study, their potential benefits underscore the method's scalability and adaptability in evolving IoT security landscapes.

As part of the security analysis, the proposed method was subjected to various attack scenarios, including brute force, side-channel, timing-based, and hardware manipulation attacks. These evaluations aimed to provide a comprehensive view of the method's resilience and confirm its applicability in real-world IoT and embedded systems. The analyses show that the method is highly resistant to a wide range of threats, offering both strong security and improved energy efficiency compared to traditional approaches.

#### 3.4.1. Brute Force Attack Analysis

Brute force attacks involve attempting every possible key until the correct one is found (Stallings, 2016). The proposed method's resistance against such attacks is assessed as follows:

- **Large Key Space:** The proposed method generates 128-bit keys using the MD5 hash function (NIST, 1995), resulting in  $2^{128}$  possible combinations. Even with highly optimistic assumptions 'such as testing  $10^{12}$  keys per second' attempting every key would take on the order of  $10^{22}$  years, rendering brute force attempts practically impossible within any realistic timeframe.
- **Dynamic Per-Message Keys:** Each message uses a unique key. Once the message is encrypted and transmitted, the key is never reused. An attacker would need to start from scratch with each new message, resetting any brute force progress.

In essence, the combination of a large key space and the per-message dynamic key generation ensures that brute force attacks are no more feasible than against traditional protocols like Diffie-Hellman, providing a comparable security level with potentially greater efficiency.

### 3.4.2. Side-Channel Attacks

Side-channel attacks exploit physical characteristics of a device—such as power consumption, electromagnetic emissions, or timing variations—to infer secret information. The proposed method employs a combination of dynamic RAM relocation, hardware-level authentication, and software-based encryption to increase resistance against these threats:

- **Nature of Side-Channel Attacks:**

Attackers typically monitor parameters like power analysis (observing energy consumption patterns) or electromagnetic radiation to discern encryption keys. If an encryption algorithm exhibits fixed, repeatable patterns during operation, attackers can potentially deduce sensitive data.

- **Dynamic RAM Relocation Mechanism:**

Rather than relying on static memory structures, the proposed method (when implemented with FPGA-based partial dynamic reconfiguration—PDR) moves physical RAM blocks containing critical data. By constantly changing the physical memory location where keys or sensitive materials are stored, the system prevents attackers from identifying stable, repeatable patterns. This measure complicates any attempt to correlate power or electromagnetic signals with encryption operations.

- **MicroBlaze and Software-Based Encryption:**

Executing encryption within a MicroBlaze soft processor environment provides another defense layer. Because the encryption steps are performed in software, fixed hardware-level patterns are minimized. The method's approach to generating a unique key for each message, combined with dynamic word selections and timestamps, prevents attackers from isolating regular patterns or predictable operations.

- **Integrated Security Measures:**

By blending hardware-level defenses (e.g., dynamic RAM relocation) with software strategies (e.g., per-message key generation, MD5-based hashing), the proposed approach ensures that no single vulnerability can be easily exploited. The literature suggests that such integrated approaches offer stronger protection than either hardware or software solutions alone.

Consequently, side-channel attacks become significantly more difficult, as attackers cannot rely on stable patterns or easily measurable correlations to uncover keys.

### 3.4.3. Timing Attacks

Timing attacks leverage differences in the time taken to perform certain cryptographic operations to infer information about keys or internal states (Johnson et al., 2017). Variations in processing times—often measured with high precision—can reveal subtle clues about the encryption algorithm’s structure or the secret keys themselves.

- **Principle of Timing Attacks:**

If encryption or decryption operations vary in duration depending on key bits or input data, attackers can record these slight differences and use them to reconstruct parts of the secret key. For example, RSA encryption operations may show small timing variances that indicate which bits of the key are set to 1 or 0.

- **Manipulating the Clock Source:**

A sophisticated attacker might attempt to manipulate the system’s clock source to exaggerate timing differences, making it easier to deduce key information.

- **Protection Measures in the Proposed Method:**

The proposed method guards against timing attacks by employing a reliable and tamper-resistant hardware clock source provided by the Zynq Processing System (PS) (Johnson et al., 2017; Stolz et al., 2021). This hardware clock source is accurate and resistant to external manipulation. Coupled with the method’s use of dynamic keys and software-level obfuscation, timing differences that attackers could exploit are minimized or eliminated.

By ensuring that timing cannot be easily manipulated or predicted, the proposed method renders this category of attack far less effective.

### 3.4.4. Hardware Trojan Horse Attacks

Hardware Trojan horses are unauthorized hardware components inserted into the bitstream, enabling malicious functionality. Such attacks can be severe if the bitstream is unencrypted, as attackers can analyze and modify it.

- **Nature of Hardware Trojan Horses:**

By inserting a Trojan at the hardware level, an attacker might leak secret keys or disrupt the encryption process. If the bitstream is not securely



tied to the hardware, attackers can deploy modified bitstreams that incorporate these malicious elements.

- **Device DNA and Bitstream Encryption:**

The proposed method mitigates these risks by encrypting the bitstream with a key linked to the device's unique Device DNA. Device DNA ensures that a given encrypted bitstream only functions on the FPGA device it was intended for. Attempts to load a manipulated or unauthorized bitstream will fail if it does not match the target device's DNA. This hardware-level authentication mechanism prevents attackers from introducing Trojan horses into the system.

By linking the bitstream to a unique hardware identifier, the system effectively nullifies attempts to run modified or malicious bitstreams, significantly bolstering security against Trojan horse attacks.

### 3.4.5. Dynamic Key Analysis Attacks

Dynamic key analysis attacks focus on the mechanism of generating a unique key for each message, attempting to find patterns or weaknesses in the key generation process itself. The proposed method's reliance on random word selection, variable timestamps, and a stable Device ID (akin to a PUF) adds complexity and reduces the feasibility of identifying exploitable patterns.

- **Advantages:**

- **Dynamic Word List and Timestamp:** By using a fresh random word and a new timestamp for each message, the method prevents attackers from building a predictable pattern or repeatedly analyzing the same scenario.
- **Stable Device ID as PUF:** Tying the key generation to a hardware-level identifier, inaccessible to attackers, increases security by ensuring that keys cannot be easily replicated or predicted.
- **Constant Key Rotation:** Generating a new key for each message prevents prolonged analysis on a single key, thwarting repeated attempts to refine guessing strategies.

- **Potential Weaknesses and Countermeasures:**

If portions of the word list or timestamp data were compromised, attackers might attempt to accelerate their trial process. However, dynamic RAM relocation, encryption of stored words, and unpredictable timestamp increments complicate this scenario. Storing words in encrypted form and frequently relocating them makes it exceedingly difficult for attackers to rely on static assumptions. Any weakness in

random selection or timestamp prediction is mitigated by multiple protective measures, such as using different time scales (microseconds, milliseconds), applying modulo operations, or starting from random offsets.

As a result, the complexity and variability introduced by the proposed method's dynamic key generation mechanism, supported by both hardware and software protections, significantly reduce the viability of dynamic key analysis attacks.

### **3.4.6. Dictionary Attack**

A dictionary attack is an attack method where the adversary attempts to derive the correct key by exhaustively testing each entry in a predefined word list. Since the proposed method dynamically generates a unique key for every message, it exhibits strong resistance to such attacks. Simulations have demonstrated that the time required to decrypt even a single message under these conditions is prohibitively long, effectively rendering dictionary attacks impractical.

In a dictionary attack, the adversary proceeds through the entire word list, trying each word as a potential key. For a list of 1,000 words, the attacker must make 1,000 attempts per message. However, because the proposed method changes keys dynamically for every new message, the attacker is forced to restart the entire process from scratch for each new message. This drastically reduces the feasibility of succeeding with a dictionary attack.

### **Test Conditions and Hardware Specifications**

- **Hardware:** The dictionary attack simulation was conducted on an Arduino DUE platform, which features a processor running at 84 MHz.
- **Word List:** A word list size of 1,000 words was employed, optimized to favor the attacker by including 3 words out of the 3-word list used in the key generation process. This setup represents a highly optimistic scenario from the attacker's perspective.
- **Timestamp:** To further benefit the attacker for testing purposes, the timestamp range was reduced to 100,000. In a real scenario, the timestamp range could be significantly larger and more complex.

### **Simulation Process**

During the dictionary attack simulation, the system attempted to decrypt an encrypted message using a word list that contained 3 of the words actually used in the key generation. The attacker also tried various timestamp combinations. The total number of attempts made within a fixed time frame (e.g., 5 minutes) was recorded. This scenario provides insight into how quickly (or slowly) the

attacker could progress and what percentage of the potential key space could be covered.

### Test Results

40	123493041key2	40	d9fc8bec8ab13f77fddfe9dbe5da0749
41	123493042key3	41	c48191751182cd60dle016e5b12aaef1
42	123493045key6	42	c7226286b6fdelc07e43b4376377c549
43	123493046key5	43	26c60d157fbc6c29eb344874e93d5c6a
44	123493047key8	44	f82e535e1ceb86209845a15dd335edf6
45	123493048key9	45	772bc6a90a9c79ed3c03e57fd847f17b
46	123493051key4	46	6c004672153963aae52b365dfe5ec458
47	123493053key1	47	090e749f5c03ac8c64e85b8cc7a932f9
48	123493054key2	48	a02e166078ff7ca8bc6beb00a2404bc5
49	123493055key4	49	258f53d07f9ac4aab2d4154b81571c88
50	123493077key8	50	512ad227c6e811d4f6cd49437a42e9ae

*Figure.8. Keys generated and AES encryption outputs for dictionary attack resilience tests.*

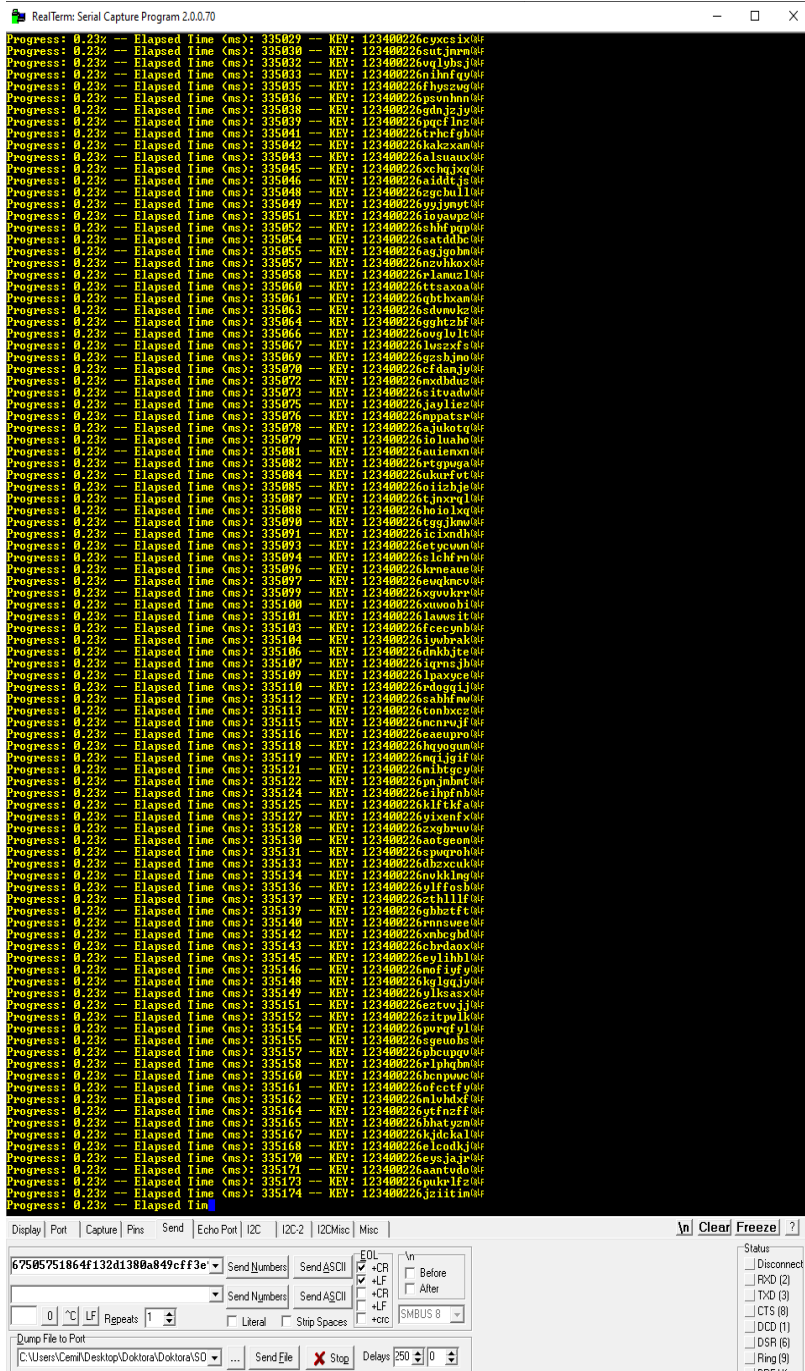


Figure.9. Screen output of the dictionary attack simulation performed with Arduino Due.

Within approximately 5 minutes, about 226,000 word attempts were processed. During this period, only about 0.20% of the total possible combinations were tested, and the encryption algorithm remained unbroken. To decrypt just one message under these favorable conditions, the attacker would need to continue the process for roughly 40 hours. After this lengthy period, the attacker would have compromised only a single message, and would need to repeat the entire process for each additional message.

In a more realistic scenario, where the word list might expand to 10,000,000 words and the nonce range remains at 100,000, the time required to decrypt a single message would increase exponentially-up to approximately 4,500 years. These calculations highlight the impractical nature of dictionary attacks against the proposed method, even under artificially advantageous conditions for the attacker.

Furthermore, the dictionary attack scenario evaluated here omits the Device ID procedure for simplification. When the Device ID mechanism is included, the attacker must also determine the device’s unique serial number. This dramatically increases complexity and time, effectively pushing the feasibility of a successful dictionary attack toward zero.

Additional measures, such as dynamic key generation and timestamp utilization, further hinder the attacker’s efforts. By implementing timestamps with different resolutions (e.g., nanoseconds, microseconds) or varying them using modular operations (mod 2, mod 3, mod 5, mod 10), predicting the correct parameters becomes exceedingly difficult. Nonce values need not start at zero and can increment by irregular steps (e.g., increments of 3 or 5), adding another layer of unpredictability. Combined with an unknown initial timestamp or nonce range, and the Device\_ID acting as a PUF (Physically Unclonable Function), these factors multiply the complexity attackers face.

**Example Test Table and Summary of Results**

<b>Duration (Sec.)</b>	<b>Number of Attempts</b>	<b>Completion Rate (%)</b>
335	~226,000	0.23%

*Table.3. The Number of Attempts and Completion Percentage During the Attack Simulation.*

The results are summarized in Table 3. After 335 seconds (approximately 5.58 minutes), about 226,000 attempts were made, equating to a completion rate of roughly 0.23%. During this time, the encryption remained intact, demonstrating the robust security provided by the proposed method against dictionary attacks.

These findings confirm that the proposed method not only ensures security by making dictionary attacks practically infeasible but also emphasizes its suitability for IoT and embedded systems, where low energy consumption and quick key generation processes are critical. Since the attacker must reinitiate the entire decryption process for each new message and message acquisition, the overall security advantage is significant.

### 3.5. Evaluation of Results

The experimental findings and security analyses conclusively show that the proposed encryption algorithm outperforms existing methods in terms of both performance and security. Its dynamic key generation mechanism, combined with hardware-level protection (e.g., bitstream encryption tied to Device DNA) and software-level measures, delivers an effective balance of energy efficiency, speed, and robustness.

Attack Method	Effectiveness	Resilience of Proposed Method
Brute Force Attack	Very Low	Very High
Side-Channel Attack	Medium	High
Timing Attack	Medium	High
Hardware Manipulation (HTH)	Medium	Medium
Dynamic Key Resolution Attack	Medium	Very High
Dictionary Attack	High	Very High

*Table 4. Resilience-Impact Analysis and Test Results of the Proposed Method Against Potential Attacks.*

Table 4 provides an at-a-glance comparison of various attacks and the proposed method’s resilience. From brute force to side-channel, timing, hardware manipulation, dynamic key resolution, and dictionary attacks, the proposed approach consistently demonstrates high resistance. By continually changing keys, employing device-specific parameters, and integrating flexible timestamp and nonce usage, it creates a formidable challenge for attackers.

#### Performance Advantages:

- **Processing Time:** In both Octave simulations and hypothetical FPGA-based tests, the proposed method was significantly faster than the Diffie-

Hellman protocol (Stallings, 2016; NIST, 1995; Johnson et al., 2017). In the Octave environment, even when certain cryptographic criteria are relaxed, the proposed method proved to be approximately 188.2% faster. Although actual FPGA tests are considered as future enhancements, conceptually, generating 128-bit prime numbers per message in DH would be time-consuming, whereas the proposed algorithm's dynamic key generation avoids such overhead.

- **Energy Efficiency:** By reducing computational overhead and using lightweight algorithms, the proposed method consumes less energy than DH. In resource-constrained IoT settings, such efficiency can translate directly into prolonged device lifespans and reduced operational costs.

#### **Security Advantages:**

- **Brute Force Resistance:** Large key space and per-message dynamic keys prevent brute force attacks from making progress.
- **Side-Channel Defense:** Dynamic RAM relocation concepts, when implemented on suitable hardware, combined with software-based encryption, limit the efficacy of power analysis and electromagnetic attacks.
- **Hardware Security:** Device DNA-based bitstream encryption ensures only authorized hardware can execute the bitstream, thwarting hardware Trojan horse attempts (Johnson et al., 2017; Stolz et al., 2021).
- **Protection Against Dictionary Attacks:** As demonstrated with the Arduino DUE simulations, dictionary attacks become exceedingly time-consuming, even under favorable conditions.

#### **Flexibility and Adaptability:**

The software-based design ensures easy adaptation to various platforms and evolving security standards. The integration of a Device ID and variable parameters (timestamps, nonces, and modular arithmetic) offers room for further complexity, ensuring that as threats evolve, so can the defense strategies.

#### **Advantages of the Proposed Method:**

- **Energy and Time Efficiency:** Dynamic key generation per message, reduced-bit AES encryption, and MD5 hashing collectively create a low-cost, high-security solution more efficient than traditional DH-based approaches.

- **Dynamic Key Generation:** Maintaining a fixed key parameter (like Device ID) alongside random elements (words, timestamps) offers a secure, energy-efficient alternative to conventional methods.

### **Future Work:**

Future studies can investigate several avenues to enhance the proposed encryption method's flexibility and resilience. One potential direction involves **algorithm hopping**, which would integrate multiple cryptographic algorithms and dynamically switch between them at runtime, further complicating an attacker's predictive capabilities and increasing overall security. Additionally, **optimizing dynamic key generation processes** may reduce processing times and energy consumption even further, thus improving suitability for a wide range of IoT devices. Another promising area is the practical implementation of hardware-level defenses suggested in this work. While the current study focuses on software-based simulations, future efforts could incorporate **FPGA-based enhancements**, such as dynamic RAM relocation (DFX) and Device DNA-dependent bitstream encryption, to achieve a robust, multi-layered security architecture. By deploying these techniques in real hardware environments, future research can validate the method's performance gains, confirm its energy savings, and quantify the added security against side-channel and hardware manipulation attacks.

In summary, the proposed method offers a high-security, energy-efficient encryption solution tailored to the needs of IoT and embedded systems. By integrating dynamic keys, lightweight algorithms, and potential hardware-based improvements, it sets a new benchmark for robust security within resource-constrained environments. With ongoing development and refinement 'ranging from algorithm hopping to FPGA-based dynamic memory strategies' this method can continue to evolve, meeting the increasingly complex demands of emerging IoT and embedded applications.

## **4. Discussion**

This section provides an in-depth examination of the proposed dynamic key generation method designed to achieve a balance between energy efficiency and security. By dynamically generating a unique key for each message 'through the combination of a randomly selected word from a predefined list and a timestamp' the method delivers enhanced security compared to classic protocols like Diffie-Hellman (Diffie & Hellman, 1976; Stallings, 2016; NIST, 1995). While the demonstrations in this study primarily utilized relatively small word lists (10, 15,



or 50 words), expanding the word list size could be explored in future work to assess its impact on processing time and overall performance.

A critical finding is that, unlike Diffie-Hellman, which maintains a static processing load and associated energy consumption, the proposed method significantly reduces both metrics. These reductions stem from its simpler, more adaptive cryptographic steps and the concept of dynamic key generation. Although this study focuses on simulations and conceptual discussions, future research could incorporate hardware-level enhancements ‘such as dynamic RAM relocation using Partial Dynamic Reconfiguration (PDR) on FPGA platforms and Device DNA-protected bitstreams’ as future work to validate and potentially improve upon these gains in real-world conditions.

Moreover, the proposed method’s suitability for IoT scenarios cannot be overstated. As IoT and embedded devices often face stringent power and resource constraints, the ability to minimize computation time and energy usage without sacrificing security is paramount. Simulation and preliminary test results, conducted under controlled conditions, indicate that the method consumes less energy and completes encryption and decryption operations faster than Diffie-Hellman. These improvements are especially valuable for devices operating on limited energy budgets, such as battery-powered sensors and actuators within IoT networks.

In terms of security impact, both Diffie-Hellman and the proposed method rely on a 128-bit key space, providing a robust level of cryptographic strength (NIST, 1995; Stallings, 2016). However, Diffie-Hellman’s static key usage means that if a key is ever compromised, the security of the entire communication session is threatened. In contrast, the proposed method’s dynamic per-message key approach ensures that only a single intercepted message is affected. This incremental protection can offer a substantial advantage in high-security applications, where the cost of losing all historical data is far higher than losing just one isolated message.

Another important consideration is the complexity of cryptographic operations. Diffie-Hellman depends heavily on large prime number arithmetic and modular exponentiation, which significantly increase computation time and resource consumption. The proposed algorithm, by contrast, involves simpler operations that can be executed more rapidly and with lower energy overhead—critical benefits for IoT devices with constrained processing capabilities.

**Table 8** presents a large-scale comparison of various criteria between the proposed method and Diffie-Hellman.

Criteria	Diffie-Hellman	Proposed Method	Comment
Breaking Difficulty	Very high ( $2^{128}$ )	Very high ( $2^{128}$ )	Diffie-Hellman has the same breaking difficulty as the proposed method.
Message Loss in Case of Compromise	All messages	Only one message	If Diffie-Hellman is compromised, all messages are at risk. In the proposed method, only a single message is lost.
Processing Cost	High	Low	The proposed method has a much lower processing cost compared to Diffie-Hellman.
Energy Consumption	High	Low	The proposed method is more efficient in terms of energy consumption.
Algorithm Complexity	Complex	Simple	Diffie-Hellman relies on complex mathematical operations. The proposed method is simpler and faster.
Data Size	Large	Small	The proposed method works with smaller data sizes, increasing efficiency.
Key Generation Time	Long	Short	The proposed method is faster in terms of key generation time.
Memory Usage	High	Low	The proposed method is more advantageous in terms of memory usage.

Criteria	Diffie-Hellman	Proposed Method	Comment
Performance	Low	High	The proposed method is superior in terms of performance.
Security Level	Very high	High	While Diffie-Hellman provides very high security, the proposed method provides practically sufficient high security.
Application Area	Widely accepted, not suitable for IoT	Suitable for IoT	The proposed method is especially more suitable for IoT applications.

**Table 8.** Large-scale comparison of the proposed method and Diffie-Hellman.

The table clarifies the stark differences in application areas: while Diffie-Hellman is widely recognized and accepted, it often proves unsuitable for IoT due to its high energy demands and complex computations. The proposed method, by simplifying the cryptographic routine and adopting dynamic key generation, emerges as a suitable alternative, offering a compelling mix of security, energy efficiency, and performance.

Looking ahead, several optimization opportunities can further enhance the proposed method’s applicability. Fine-tuning word list management strategies, more efficient processing of timestamps or nonces, and leveraging device IDs more effectively could lead to even shorter processing times and reduced energy consumption. Additionally, integrating lightweight algorithms (e.g., ACORN) can further decrease energy costs, while FPGA-level techniques like dynamic memory repositioning (DFX) and Device DNA-based protections could be implemented in future research to strengthen security against side-channel attacks and hardware manipulation (Johnson et al., 2017; Stolz et al., 2021).

From a performance perspective, the method’s ability to generate keys dynamically and employ random selection from a word list ‘combined with timestamps’ results in a solution that achieves strong security without demanding extensive resources. The receiving device’s “trial and solve” process remains manageable and does not approach the complexity or overhead associated with Diffie-Hellman’s intensive computations.

In conclusion, the proposed method stands as a promising encryption framework well suited for IoT and embedded systems. By offering dynamic key generation, reduced energy consumption, and shorter processing times, it aligns closely with the constraints and demands of real-time, resource-limited environments. With further research focusing on word list scalability, hardware-based enhancements, and specialized cryptographic primitives, the proposed method has the potential to set new standards for secure, efficient communication in the expanding landscape of connected devices.

#### **4.1. Advantages and Disadvantages of the Proposed Method**

This section discusses the proposed encryption method's advantages and disadvantages in terms of energy efficiency, security, and processing cost. Comparisons to Diffie-Hellman, LifeLine, and other literature are also highlighted.

##### **Advantages:**

- **Energy Efficiency:**  
By relying on software-based encryption and limiting the use of ICAP operations solely to RAM-related tasks in a conceptual future setup, the proposed method significantly reduces energy consumption. Simulations in Octave and preliminary analyses indicate that the method provides substantially improved energy efficiency compared to Diffie-Hellman. This trait is critical for battery-powered IoT devices. As demonstrated through the dictionary attack scenario on Arduino DUE, the method maintains low energy consumption and short processing times, making it a practical solution for resource-constrained systems.
- **High Performance:**  
The proposed method's simplified algorithmic steps and dynamic key generation result in shorter processing times. This high performance is advantageous for real-time IoT applications requiring rapid encryption and decryption without excessive overhead.
- **Security:**  
From a security perspective, the method resists brute force attacks due to its 128-bit key space and dynamic key generation (Stallings, 2016; NIST, 1995). Assigning a unique key to each message forces attackers to restart their attempts for every intercepted message. The use of MD5 ensures large key spaces, and the approach is inherently flexible enough to integrate other hashing or lightweight encryption algorithms. While software-based encryption and dynamic RAM relocation (as a future

hardware-level concept) can reduce the effectiveness of side-channel attacks by obscuring fixed patterns, it must be noted that side-channel threats cannot be entirely eliminated. Additional hardware-based measures (e.g., power balancing or PUF integration) are recommended for complete mitigation.

- **Flexibility:**

The software-centric nature of the design allows it to be readily adapted to different platforms and evolving security requirements. Updating parameters such as the word list, timestamps, and nonce strategies is straightforward, ensuring long-term sustainability and adaptability of the solution.

**Disadvantages:**

- **Incomplete Side-Channel Attack Prevention:**

Although the approach reduces the likelihood and impact of side-channel attacks, it does not completely eradicate them. Future work should explore integrating hardware countermeasures to further minimize these vulnerabilities.

- **Word List Security Dependence:**

The security of the method partially hinges on the confidentiality of the word list. If the list is exposed, attackers may gain advantages in predicting keys. Ensuring secure storage and dynamic relocation of this list is essential, potentially requiring additional encryption or obfuscation measures.

#### **4.2. Evaluation of Attack Methods and Explanation of Metrics**

The effectiveness of various attack methods and the method's resilience have been qualitatively assessed based on literature comparisons and theoretical analyses.

- **Brute Force Attacks:**

Given the 128-bit key space, brute force attacks are practically infeasible. Even at extreme trial rates, attempting  $2^{128}$  possible keys remains astronomically time-consuming. Thus, the resilience level is classified as "Very High."

- **Side-Channel Attacks:**

Side-channel attacks exploit physical characteristics such as power consumption, electromagnetic emissions, or timing variations. The proposed method conceptually employs dynamic RAM relocation (a future hardware-level enhancement) and software-based encryption to

obscure consistent patterns. While these measures raise the difficulty of executing side-channel attacks, they do not offer complete immunity. Additional hardware-level mitigations (e.g., balancing techniques) would further enhance security.

Due to the complexity and numerous variables involved in side-channel attacks, the analysis remains qualitative rather than quantitative. The increased difficulty these attacks face in the proposed framework ‘through changing memory positions and reduced hardware signal patterns’ indicates a significant improvement over conventional static encryption approaches.

### 4.3. Future Work and Recommendations

Several directions exist for further enhancing the proposed method:

- **Integration of Lightweight Encryption Algorithms:**  
Incorporating lightweight algorithms like ACORN or JAMBU could further reduce energy consumption and processing overhead, benefiting battery-powered IoT devices.
- **Enhanced Side-Channel Protections:**  
Future research could include integrating hardware-based countermeasures ‘such as power consumption balancing, PUF-based authentication, and advanced shielding techniques’ to more effectively combat side-channel attacks.
- **Algorithm Hopping Mechanism:**  
Implementing algorithm hopping, where the encryption algorithm changes dynamically, can raise the barrier for attackers. By ensuring that no single cryptographic method remains constant, attackers face greater uncertainty and complexity.
- **FPGA-Based Realization and Device DNA Integration:**  
While this study focused on simulations (Octave) and an Arduino-based dictionary attack scenario, future work may implement the proposed technique on FPGA platforms using dynamic RAM relocation and Device DNA-based bitstream protection. Such hardware-level improvements would provide a robust multi-layered defense and validate energy/performance claims in real-world conditions.

## 5. Conclusions

This study presents an energy-efficient, secure, and flexible encryption method tailored for IoT and embedded systems with limited resources. Traditional encryption protocols, like Diffie-Hellman, demand high processing power and exhibit excessive energy consumption, making them less suitable for constrained

environments. By employing dynamic key generation, lightweight algorithms, and suggesting future integration of hardware-level strategies (e.g., dynamic RAM relocation, Device DNA authentication), the proposed method addresses these issues.

Octave-based simulations and Arduino dictionary attack scenarios show that the method significantly reduces processing time and estimated energy consumption compared to classical approaches. The dynamic key generation and large key space render brute force attacks effectively impossible. Although not fully eliminating side-channel attacks, the method's conceptual hardware-software interplay (dynamic RAM management, software-driven encryption steps) drastically reduces their impact.

The software-based structure ensures broad applicability and ease of adaptation to new platforms. Integrating lightweight encryption algorithms and hardware-based side-channel defenses 'alongside possible algorithm hopping' represents a promising direction for future work. Such enhancements could further elevate the method's energy efficiency, performance, and security, establishing a new benchmark for secure communication in IoT and embedded systems.

In conclusion, the proposed encryption method contributes significantly to the literature by offering a practical, energy-conscious, and secure solution well-suited for devices with limited resources. By bridging the gap between performance and security, it meets key IoT demands and sets a foundation for future research and wider applications.

## 6. References

1. **Xiao, K., Forte, D., Jin, Y., & Tehranipoor, M. (2016).** Hardware trojans in FPGAs: Architecture and techniques. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 35(1), 36-46.
2. **Ba, P., Xia, Y., & Wang, Z. (2015).** Malicious bitstreams: A security threat to reconfigurable systems. *International Journal of Reconfigurable Computing*, 2015, 1-9.
3. **Zhang, L., & Parhi, K. K. (2014).** Side-channel attack resistant architectures for advanced encryption standard. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 61(4), 957-967.
4. **Zhou, H., Chang, C. H., & Cao, L. (2017).** Design of lifeline for FPGA intellectual property protection by binding sequential logic to physical unclonable function. *IEEE Transactions on Information Forensics and Security*, 12(1), 168-179.
5. **Ahmadi, M., & Tahoori, M. B. (2018).** Dynamic reconfigurable IoT security: Challenges and opportunities. *Proceedings of the IEEE*, 106(1), 72-86.
6. **Guo, X., Huang, K., & Lach, J. (2016).** Energy-adaptive cryptographic solutions for energy-harvesting IoT devices. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 63(5), 629-638.
7. **Zhang, Y., & Li, H. (2013).** Low-cost obfuscation of FPGA bitstream. *International Conference on Field-Programmable Technology*, 335-338.
8. **Helfmeier, C., Boit, C., Nedospasov, D., & Seifert, J. (2013).** Cloning physically unclonable functions. *IEEE International Symposium on Hardware-Oriented Security and Trust*, 1-6.
9. **Kaya, D., & Dündar, G. (2019).** Secure IP core distribution and usage on reconfigurable hardware. *Integration*, 66, 64-74.
10. **Suh, G. E., & Devadas, S. (2007).** Physical unclonable functions for device authentication and secret key generation. *Proceedings of the 44th annual Design Automation Conference*, 9-14.
11. **Johnson, A. P., Patranabis, S., Chakraborty, R. S., & Mukhopadhyay, D. (2017).** Remote dynamic partial reconfiguration: A threat to Internet-of-Things and embedded security applications. *Microprocessors and Microsystems*, 52, 131-144.
12. **Stolz, F., Albartus, N., Speith, J., Klix, S., Nasenberg, C., Gula, A., ... & Tessier, R. (2021).** LifeLine for FPGA protection: Obfuscated cryptography for real-world security. *Cryptology ePrint Archive*.
13. **Soliman, S., Jaela, M. A., Abotaleb, A. M., Hassan, Y., Abdelghany, M. A., Abdel-Hamid, A. T., ... & Mostafa, H. (2019).** FPGA



- implementation of dynamically reconfigurable IoT security module using algorithm hopping. *Integration*, 68, 108-121.
14. **Samir, N., Gamal, Y., El-Zeiny, A. N., Mahmoud, O., Shawky, A., Saeed, A., & Mostafa, H. (2019, May).** Energy-adaptive lightweight hardware security module using partial dynamic reconfiguration for energy limited internet of things applications. In *2019 IEEE International Symposium on Circuits and Systems (ISCAS)* (pp. 1-4). IEEE.
  15. **Karam, R., Hoque, T., Ray, S., Tehranipoor, M., & Bhunia, S. (2016, November).** Robust bitstream protection in FPGA-based systems through low-overhead obfuscation. In *2016 International Conference on ReConFigurable Computing and FPGAs (ReConFig)* (pp. 1-8). IEEE.
  16. **Adetomi, A., Enemali, G., & Arslan, T. (2017, September).** Towards an efficient intellectual property protection in dynamically reconfigurable FPGAs. In *2017 Seventh International Conference on Emerging Security Technologies (EST)* (pp. 150-156). IEEE.
  17. **Miller, V. S. (1985, August).** Use of elliptic curves in cryptography. In *Conference on the theory and application of cryptographic techniques* (pp. 417-426). Berlin, Heidelberg: Springer Berlin Heidelberg.
  18. **Rivest, R. L., Shamir, A., & Adleman, L. (1978).** A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2), 120-126.
  19. **Al-Meer, A., & Al-Kuwari, S. (2023).** Physical unclonable functions (PUF) for IoT devices. *ACM Computing Surveys*, 55(14s), 1-31.
  20. **Diffie, W., & Hellman, M. E. (1976).** New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6), 644-654.
  21. **Daemen, J., & Rijmen, V. (2002).** *The design of Rijndael: AES - The advanced encryption standard*. Springer-Verlag.
  22. **National Institute of Standards and Technology (NIST). (2001).** Advanced encryption standard (AES). *FIPS PUB 197*.
  23. **Kerckhoffs, A. (1883).** La cryptographie militaire. *Journal des Sciences Militaires*, 9, 5-38.
  24. **Blanchet, B. (2024).** Dealing with dynamic key compromise in CryptoVerif. In *Proceedings of the 37th IEEE Computer Security Foundations Symposium* (pp. 495–510). IEEE, Enschede, Netherlands. <https://doi.org/10.1109/CSF61375.2024.00015>. HAL ID: fihal-04271666v2f.
  25. **Akdemir, K., et al. (2010).** "Efficient AES Implementations on Westmere." *Intel Corporation*.

26. **Intel. (2010).** "Advanced Encryption Standard (AES) New Instructions Set White Paper."
27. **Menezes, A. J., van Oorschot, P. C., & Vanstone, S. A. (1996).** Handbook of Applied Cryptography. *CRC Press*.
28. **Schneier, B., Kelsey, J., Whiting, D., Wagner, D., Hall, C., & Ferguson, N. (1999).** "Performance comparison of the AES submissions." URL: <https://www.schneier.com/wp-content/uploads/2016/02/paper-aesperformance.pdf> (accessed 24.01. 2022).
29. **Crypto++.** (2024). "Crypto++ Benchmarks." *Crypto++ Library*. Retrieved from <https://openbenchmarking.org/test/pts/cryptopp>

## 7. APPENDICES

### A. Octave Diffie-Hellman and Proposed Method Time Comparison Code

```
pkg load statistics; % Load the statistics package for Octave
iterations = 100; % Number of iterations for the experiment
kelime_sayisi = 15; % Number of words in the word list
max_time_difference = 4; % Maximum backward timestamp range for MD5 testing
% Initialize time arrays for performance measurements
device1_dh_times = zeros(1, iterations); % DH key generation times (Device
1)
device2_dh_times = zeros(1, iterations); % DH key resolution times (Device
2)
device1_md5_times = zeros(1, iterations); % MD5 key generation times (Device
1)
device2_md5_times = zeros(1, iterations); % MD5 key resolution times (Device
2)
device1_aes_times = zeros(1, iterations); % AES encryption times (Device
1)
device2_aes_times = zeros(1, iterations); % AES decryption times (Device
2)

% Input text for AES encryption (must be exactly 16 bytes)
input_text = 'TestMesajil23456';

% Word list used for MD5 key generation
temel_kelimeler = {'elma', 'armut', 'kiraz', 'muz', 'cilek', 'karpuz',
'kavun', 'erik', 'visne', 'seftali', 'uzum', 'portakal', 'mandalina',
'ayva', 'dut', 'incir', 'nar', 'avokado', 'greyfurt', 'kivi', 'limon',
'havuc', 'lahana', 'marul', 'ispanak', 'pirasa', 'patates', 'sogan',
'sarimsak', 'domates', 'salatalik', 'biber', 'patlican', 'kabak',
'bezelye', 'fasulye', 'nohut', 'mercimek', 'pirinc', 'bulgur', 'makarna',
'ekmek', 'peynir', 'yogurt', 'sut', 'yumurta', 'et', 'balik', 'tavuk',
'sucuk'};

kelime_listesi = temel_kelimeler(1:kelime_sayisi); % Select the first N
words

% Iterate over the specified number of iterations
for i = 1:iterations
    % --- Diffie-Hellman Key Generation and AES Encryption ---
    [combined_key, total_alice_time_prime, total_alice_time,
total_bob_time] = generate_combined_key();
    device1_dh_times(i) = total_alice_time_prime; % DH key generation
(Device 1)
    device2_dh_times(i) = total_bob_time; % DH key resolution (Device 2)
    % AES encryption using the generated DH key
    plaintext_bytes = uint8(input_text);
    key_hex = dec2hex(combined_key, 32);
    key_bytes = uint8(key_hex(1:16)); % Convert the key to 16 bytes
    % AES encryption process
    tic;
    w = key_expansion(key_bytes);
    aes_encrypt(plaintext_bytes, w);
    device1_aes_times(i) = toc;
    % AES decryption process
    tic;
    aes_decrypt(aes_encrypt(plaintext_bytes, w), w);
    device2_aes_times(i) = toc;
    % --- MD5 Key Generation and AES Encryption ---
    rastgele_kelime = kelime_listesi(randi(kelime_sayisi)); % Randomly
select a word
    timestamp = num2str(floor(now * 1e5)); % Generate a timestamp
```

```

    combined_string = [rastgele_kelime, timestamp]; % Combine word and
timestamp
    [hash1, elapsed_time1] = md5_hash(combined_string); % Generate MD5 hash
device1_md5_times(i) = elapsed_time1;
    % MD5 key resolution
    found = false;
    start_time = floor(now * 1e5); % Starting timestamp for backward search
    total_solution_time = 0;
    for w_index = 1:kelime_sayisi
        test_word = kelime_listesi{w_index};
        test_combined_string = [test_word, timestamp];
        [test_hash, elapsed_time2] = md5_hash(test_combined_string);
        total_solution_time += elapsed_time2;
        if strcmp(test_hash, hash1) % Check if the key matches
            found = true;
            break;
        end
    end
    device2_md5_times(i) = total_solution_time;
end
% Plot cumulative comparison of DH (Key + AES) vs. MD5 (Key + AES)
figure;
plot(1:iterations, cumsum(device1_dh_times + device1_aes_times), 'r',
'LineWidth', 1.5); hold on;
plot(1:iterations, cumsum(device2_md5_times + device2_aes_times), 'b',
'LineWidth', 1.5);
title('Cumulative Comparison: DH (Key + AES) vs. MD5 (Key + AES)');
xlabel('Iteration');
ylabel('Cumulative Time (seconds)');
legend('DH (Key + AES)', 'MD5 (Key + AES)', 'Location', 'northwestoutside');
hold off;
% Plot cumulative comparison of DH Transmission vs. MD5 Decryption
figure;
plot(1:iterations, cumsum(device1_dh_times), 'r', 'LineWidth', 1.5); hold
on;
plot(1:iterations, cumsum(device2_md5_times), 'b', 'LineWidth', 1.5);
title('Cumulative Comparison: DH Transmission vs. MD5 Decryption');
xlabel('Iteration');
ylabel('Cumulative Time (seconds)');
legend('DH Transmission', 'MD5 Decryption', 'Location',
'northwestoutside');
hold off;

```

## B. Arduino DUE Dictionary Attack Code

```

#include "MD5.h"
#include <CryptoAES_CBC.h>
#include <AES.h>
#include <CBC.h>

#define BLOCK_SIZE 16
#define ATTACKER_WORD_LIST_SIZE 1000
#define MAX_STRING_SIZE 256
#define AES_KEY_SIZE 16
#define MAX SENDS 300
MD5 md5;
CBC<AES128> cbcaes128;
byte iv[BLOCK_SIZE] = {0};
char* attacker_word_list[ATTACKER_WORD_LIST_SIZE];
unsigned long attacker_nonce_start = 123400000;
unsigned long attacker_nonce_end = 123500000;

```

```

char cleartext[MAX_STRING_SIZE];
unsigned long decryptionTimes = 0;
int messageCount = 0;
int totalAttempts = 0;
unsigned long totalPossibleAttempts;
int sendCounter = 0;
void generateAttackerWordList() {
    attacker_word_list[0] = "key1";
    attacker_word_list[1] = "key2";
    attacker_word_list[2] = "key8";
    for (int i = 3; i < ATTACKER_WORD_LIST_SIZE; i++) {
        char* word = (char*)malloc(8);
        for (int j = 0; j < 7; j++) {
            word[j] = 'a' + random(26);
        }
        word[7] = '\\0';
        attacker_word_list[i] = word;
    }
}
void setup() {
    Serial.begin(460800);
    generateAttackerWordList();
    totalPossibleAttempts = (attacker_nonce_end - attacker_nonce_start +
1) * ATTACKER_WORD_LIST_SIZE;
}
void loop() {
    if (Serial.available()) {
        String encryptedMessage = Serial.readString();
        unsigned long startTime = millis();
        bool found = false;
        for (unsigned long nonce = attacker_nonce_start; nonce <=
attacker_nonce_end && !found; nonce++) {
            for (int i = 0; i < ATTACKER_WORD_LIST_SIZE && !found; i++) {
                totalAttempts++;
                char* key_word = attacker_word_list[i];
                char key[MAX_STRING_SIZE];
                sprintf(key, "%lu%s", nonce, key_word);

                unsigned char* md5_result = md5.make_hash(key);
                char* md5_str = md5.make_digest(md5_result, 16);
                free(md5_result);

                byte aesKey[AES_KEY_SIZE];
                for (int j = 0; j < AES_KEY_SIZE; j++) {
                    sscanf(&md5_str[j * 2], "%2hhx", &aesKey[j]);
                }
                free(md5_str);

                int cipherTextLength = encryptedMessage.length() / 2;
                byte cipherText[BLOCK_SIZE];
                for (int k = 0; k < cipherTextLength; k++) {
                    sscanf(&encryptedMessage.c_str()[k * 2], "%2hhx",
&cipherText[k]);
                }

                cbcaes128.clear();
                cbcaes128.setKey(aesKey, AES_KEY_SIZE);
                cbcaes128.setIV(iv, BLOCK_SIZE);
            }
        }
    }
}

```

```

        cbcaes128.decrypt((byte*)cleartext, (byte*)cipherText,
cipherTextLength);

        String decryptedMessage = String(cleartext);

        if (decryptedMessage.startsWith("OK") &&
decryptedMessage.endsWith("OK")) {
            unsigned long endTime = millis();
            unsigned long timeTaken = endTime - startTime;
            decryptionTimes += timeTaken;
            messageCount++;
            found = true;

            Serial.println("Decrypted Message: " +
decryptedMessage);
            Serial.print("Key Word: ");
            Serial.println(key_word);
            Serial.print("Nonce: ");
            Serial.println(nonce);
            Serial.print("Elapsed Time (ms): ");
            Serial.println(timeTaken);
            break;
        }
    }
}
if (!found) {
    Serial.println("Attacker could not decrypt the message.");
}

if (messageCount > 0) {
    unsigned long averageTime = decryptionTimes / messageCount;
    Serial.print("Average decryption time (ms): ");
    Serial.println(averageTime);
}
}
}

```