Implementation of A MEMS Resonator-based Digital to Frequency Converter Using Artificial Neural Networks

Xuecui Zou, Sally Ahmed, Hossein Fariborzi
King Abdullah University of Science and Technology (KAUST) Thuwal, Saudi Arabia

***Abstract*—This paper proposes a novel approach for micro-electromechanical resonator-based digital to frequency converter (DFC) design using artificial neural networks (ANN). The DFC is a key building block for multiple digital and interface units. We present the design of a 4-bit DFC device which consists of an in-plane clamped-clamped micro-beam resonator and 6 partial electrodes. The digital inputs, which are DC signals applied to the corner partial electrodes, modulate the beam resonance frequency using the electrostatic softening effect. The main challenge in the design is to find the air gap size between each input electrode and the beam to achieve the desired relationship between the digital input combinations and the corresponding resonance frequencies for a given application. We use a shallow, fully-connected feedforward neural network model to estimate the airgaps corresponding to the desired resonance frequency distribution, with less than 1% error. Two special cases are discussed for two applications: equal airgaps for implementing a full adder (FA), and weight-adjusted airgaps for implementing a 4-bit digital to analog converter (DAC). The training, validation, and testing datasets are extracted from finite-element-method (FEM) simulations, by obtaining resonance frequencies for the 16 input combinations for different airgap sets. The proposed method based on ANN model paves the way for a new design paradigm for MEMS resonator-based logic and opens new routes for designing more complex digital and interface circuits.**

Keywords—Digital-to-Frequency converter, micro-resonators, nonlinear regression, machine learning, neural networks.

I. Introduction

The interest in circuit implementation using micro-electromechanical systems (MEMS) technology has increased in recent years due to its inherent ultra-low power consumption compared to complementary-metal-oxide-semiconductor (CMOS) technology [1]. In particular, MEMS resonator-based logic gates [2], memory devices [3], computational units [4], and data converters [5, 6] were demonstrated recently, showing interesting aspects of resonators such as reprogrammability, reduced circuit complexity, durability, and low power consumption. To implement such circuits, modeling the MEMS resonator devices for functional verification and optimization is required. Some commercial software such as ANSYS®, HFSS®, CoventorWare®, and COMSOL Multiphysics® are generally used in modeling MEMS devices based on well-known modeling techniques such as finite element analysis (FEA), and boundary element analysis (BEA). Although these modeling techniques are very accurate, they are computationally expensive. For example: to optimize the design parameters of the resonator-based DAC presented in [5], numerous COMSOL simulations were done to optimize device parameters by trial and error, which is time-consuming and inefficient, especially for large circuits.

Since optimizing design parameters is essential, an inverse process that can estimate those parameters from the expected results shall reduce the design time significantly and improve design efficiency as well. Artificial neural networks (ANNs) offer a simple, yet powerful approach for the creation of the inverse model. Fully connected (FC) ANNs are used since they can easily learn the relationship between any two sets of data and generalize it [7]. They can predict the correct response to any set of inputs that the ANN has never seen before in a very short amount of time. Therefore, ANNs have been adapted in various fields for different applications. A neural network based on the radial basis function was used to compensate for temperature variations for inertial sensors [8]. By training a two-layer feed-forward back-propagation network, the artificial neural network in [9] could predict the electrical responses of a resonator for different driving parameters. The trained ANNs in [10] have already been utilized to model integrated circuits in the RF/microwave area and predict the device responses for arbitrary input datasets within the range of ANNs. An ANN-based model for a capacitive RF MEMS switch design was created to determine the geometrical parameters that achieve the desired electrical resonance frequency or the required actuation voltage, thereby avoiding the time-consuming optimizations [11].

The purpose of this paper is to develop an ANN model for the MEMS resonator device that has been used to implement logic gates, a full adder [4], and a digital to analog converter [5]. This model tries to obtain the required design parameters for given resonance frequencies. The device used to implement these circuits consists of a clamped-clamped micro-beam resonator and multiple side electrodes on each side of the beam as shown in Fig.1. The digital inputs, applied on corner side electrodes, tune the beam resonance frequency using the electrostatic softening effect. Depending on the sizes of the airgaps that separate the electrodes from the beam, different relationships between the digital input combinations and the resulting resonance frequencies are obtained. In other words, by manipulating the airgap sizes, we can change the circuit application of the resonator. Optimizing these airgaps, using commercial modeling softwares, to obtain the desired output is time-consuming. Therefore, in this paper, a fully-connected feed-forward ANN model is used to obtain the required air gap sizes for a 4-bit digital to frequency converter (DFC) which is a core block for implementation of a wide range of digital and interface circuits. The proposed model can respond instantaneously while retaining accuracy, thereby eliminating the need for tedious optimizations and numerous simulations.

The paper is organized as follows: Section II describes the device structure and operation of the resonator-based DFC. Section III describes the used ANN structure and procedures for the resonator-based DFC inverse modeling in detail. The modeling results and analysis are discussed in section IV.

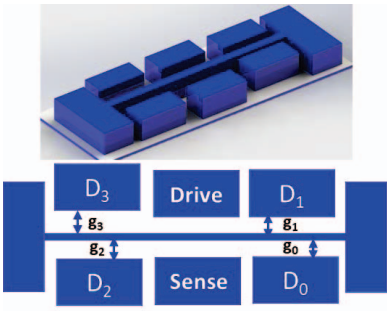


Fig.1. A 3D image and a top view of the MEMS resonator-based DFC with airgaps marked as g_0-g_3 for the digital inputs D_0-D_3.

II. MEMS RESONATOR-BASED DFC STRUCTURE AND OPERATION

A. MEMs Resonator-based DFC Device

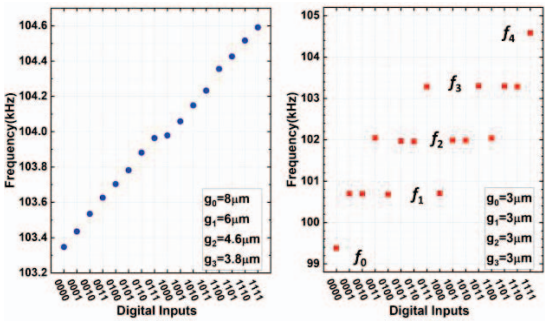
Figure 1 shows the 3D schematic and the top view of the MEMs resonator-based 4-bit digital-to-frequency converter (DFC). The demonstrated 4-bit DFC in [5] consists of a clamped-clamped micro-beam resonator biased with DC voltage V_{bias} and six partial electrodes. The two center electrodes are used as the drive and sense electrodes. The other four corner electrodes D_0-D_3 are used for applying the digital inputs. Each electrode D_n is separated from the beam by g_n as shown in Fig. 1. The digital inputs are DC voltages, where 0V represents the digital '0' input, and a higher DC voltage is used for the digital '1' input. The voltage difference between the beam and the corner electrodes modulates the resonance frequency of the resonator using electrostatic softening effects. The relationship between the digital input combinations and the beam resonance frequency is described by the following equation [5]:

$$\omega = \sqrt{\frac{k}{m} - \frac{\varepsilon}{m}\left[\begin{array}{l}\frac{A_{D0}(V_{bias}-V_{D0})^2}{(g_0+\delta(x))^3} + \frac{A_{D1}(V_{bias}-V_{D1})^2}{(g_1-\delta(x))^3} + \\ \frac{A_{se}(V_{bias}-V_{sense})^2}{(g+\delta(x))^3} + \frac{A_{Dr}(V_{bias}-V_{Drive})^2}{(g-\delta(x))^3} \\ + \frac{A_{D2}(V_{bias}-V_{D2})^2}{(g_2+\delta(x))^3} + \frac{A_{D3}(V_{bias}-V_{D3})^2}{(g_3-\delta(x))^3}\end{array}\right]} \quad (1),$$

where ω is the resonance frequency of the beam, m is the mass of the beam, A is the cross-sectional area between each electrode and the beam, k is the beam stiffness, δ is the static deflections of the beam and g is the gap size of the sense/drive electrodes. The relationship between the different input parameters and the resulting frequency is so complicated that only a numerical solution can be obtained at the expense of massive computing resources.

B. Two Applications Based on The DFC Device

The device, shown in Fig.1, can be used to implement two different circuits based on the airgap sizes. The first application is the digital-to-analog converter (DAC), in which the airgap sizes were adjusted to account for the weight of each bit. A linear relationship between the digital combinations and the resonance frequency was obtained as shown in Fig.2 (a) [5]. The second application is a full adder using the same structure with equal beam/electrode airgap sizes. Five different frequency levels were obtained, where the combinations that contain the same number of '1's resulted in the same beam resonance frequency as illustrated in Fig.2 (b). Details of the full adder operation and implementation can be found in [4].



(a) (b)

Fig.2. The relationship between different digital input combinations and the resonance frequencies of the beam for two different gap combinations: (a) weight-adjusted airgap sizes where g_0 is for the least significant bit and g_3 is for the most significant bit, and (b) equal airgap sizes.

Moving forward, developing a general model that can obtain the required airgap sizes for given resonance frequency combinations will reduce the design time and make it more efficient. The next section describes how ANN modeling is used to precisely predict the design parameters of the MEMS resonator-based DFC.

III. ANN MODELING FOR DESIGN PARAMETER PREDICTION

In this part, artificial neural network modeling is proposed and demonstrated by using machine learning techniques to solve the inverse design problem. To model the mathematical relationship between the frequency combinations and airgap sizes (which is a regression problem), a two-layer fully-connected feed-forward ANN is employed.

A. The Artificial Neural Network

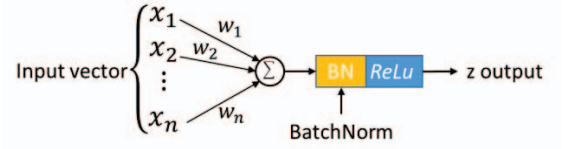


Fig.3 The mathematical flowchart of a single neuron used in the network. The input vector is multiplied by the weights, then processed by the normalization layer and ReLu nonlinearity to generate the output which is the input for the neurons in the subsequent layer.

Each neuron in the network, except for the output layer, has the same structure and function. Fig. 3 shows the structure of a single neuron used in the fully connected model. The input vector $X = [x_1, \ \dots \ x_n]^T$ is the normalized resonance frequency sequence of f_0-f_{15}, n=16. The inputs are multiplied by the weights vector $W \in \mathbb{R}^{16}$, then fed into the neuron to get the summation with bias b added. A batch normalization (BN) transform is applied to the product, then an activation function is applied to process the normalized values. The mathematical representation of the single neuron is described in the following equation:

$$z = f\big(BN(b + X^T W)\big) = f\big(BN(X^T W)\big) \quad (2)$$

Where W, b are learned parameters of the model, and ReLu is chosen as the activation function (AF) in the model to account for the nonlinear relationship between the resonance frequencies (inputs) and airgap values (outputs). Since the term ($b + X^T W$) is normalized, the bias b will be canceled by the mean subtraction. It is worth mentioning that batch normalization decreases the training time and makes it more

stable, while it limits the output range. To avoid the restrictions on output values, the neurons in the output layer directly produce the final output values from the summation, without the normalization process.

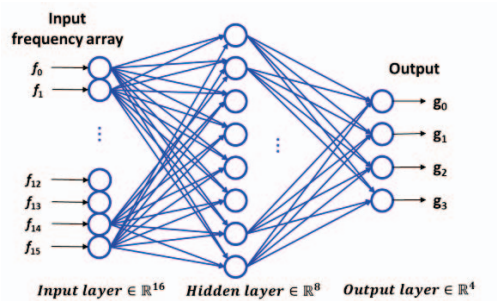


Fig.4 The architecture of the two-layer ANN. The frequency distribution is set as input and the gaps of the design parameters are outputs.

The network used to solve this regression problem is a multiple-to-multiple fully-connected neural network, shown in Fig. 4. The number of neurons in the input and output layers need to match the dimensions of the input and output vectors, which are 16, and 4 respectively. All the outputs of the neurons would be passed to all the neurons in the subsequent layer as inputs after being multiplied by the weights. Two hidden layers, at least, are required to properly describe the nonlinear relationship between the inputs and the outputs. The number of neurons in each hidden layer is optimized to be 8 as adding more neurons only consumes more time without a noticeable change in the output. Although the performance of the neural network is better with more neurons, excessive neurons present the potential overfitting situation and prolonged training time.

B. Dataset Preparation and Parameter Optimization

In order to study the relationship between the airgap sizes and the resonance frequency resulting from each input combination, all other design parameters such as the beam width, length, thickness, and electrode sizes were fixed and the only variables are the airgap sizes. The training, validation, and testing datasets are obtained using COMSOL® simulations by sweeping the gaps. For each set of airgaps (g_0, g_1, g_2, and g_3), the 16 different digital input combinations from '0000' to '1111' are applied and the corresponding 16 resonance frequencies are obtained. All gaps can take any value between 3 μm and 9 μm, where the minimum airgap value (3 μm) is restrained by the pull-in phenomena, while the maximum value is set to 9 μm as larger gaps will result in the same beam resonance frequency regardless of the applied input value. During supervised learning, the total dataset is split into three subsets: 1295 samples in the training dataset, 256 samples in the validation dataset, and 4 specific samples in the testing dataset. The training dataset is fed to train the model and the performance of the model is evaluated by the validation loss of the validation dataset.

The resonance frequencies of all samples are in a range of 99k-105 kHz which needs to be preprocessed to eliminate the significant frequency bias. In this study, we use min-max scaling to scale down the frequency values to be fixed between 0 and 1, then the datasets need to be shuffled to avoid the effect of data order before being fed into the neural network. The loss function used as the performance indicator of the ANN is the smooth mean absolute error function, also known as Huber loss [12]:

$$Loss(Y, Y^p) = \frac{1}{n}\begin{cases}\sum_i 0.5\left(y_i - y_i^p\right)^2, & if\ \left|y_i - y_i^p\right| < 1\\ \sum_i \left|y_i - y_i^p\right| - 0.5, & otherwise\end{cases} \quad (3),$$

where $Y = \{y_1, y_2, \ldots, y_n\}$, contains the given target values and $Y^p = \{y_1^p, y_2^p, \ldots, y_n^p\}$, is the predicted output of the model. The activation function is ReLu which improves computation effectively. The artificial neural network minimizes the loss function using the stochastic gradient descent (SGD) algorithm with back-propagation. The network connection weights and parameters of the batch norm layer are adjusted in the supervised training progress. All the parameters used in the ANN model are summarized and listed in table I.

TABLE I. THE SUMMARY LIST OF THE ANN PARAMETERS

Parameters/metrics	Value
Number of layers	2
Number of neurons for each layer	16, 8, 4
Batch Size	96
AF	ReLu
Optimizer	SGD (lr=5e-3, momentum=0.95)
Percentage of training data	83.28
Percentage of validation data	16.46

After the model's parameters are confirmed and the model is trained, three loss functions are used to evaluate the performance of the model fitting by the error of the validation dataset in Fig. 5. Since the validation dataset is unknown for the model, the error values of the validation data would reveal how well the model is trained. Overfitting refers to a model that is over-trained and has a bad performance when the validation dataset is applied, which is used to explain the poor generalization of a well-trained network.

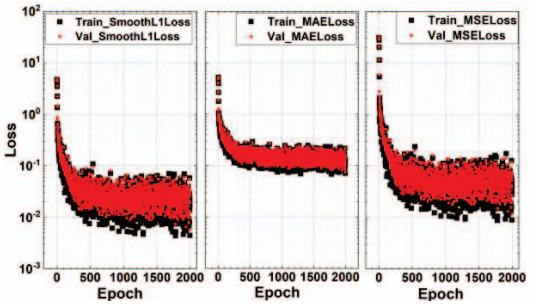


Fig.5 Visualization of (a) the smooth mean absolute error (SmoothL1L), (b) mean absolute error (MAE), and (c) mean square error (MSE) of the training dataset (red circles) and the validation dataset (black squares) over the training steps, where the epoch is limited to 2000.

In Fig. 5, the errors for the training dataset and the validation dataset are similar, which indicates that the trained model avoided the overfitting problem. Although the error values of the smoothL1Loss and the MSE fluctuate a bit more widely compared to the MAE, the MAE value of the trained model converges to 0.1 μm which is better than we projected since the resolution of the fabrication process used for micro-beam resonators is limited to 0.25 μm[13].

IV. Results and Discussion

As a well-trained model, the regression coefficient matrix *W* and thresholds of activation functions can be obtained after the training process. After the verification of the ANN model, it is tested with 4 different frequency sets. The model is fed with the target frequency distribution in the testing dataset to predict the gaps. Fig.6 visualizes the comparison between the predicted and the true values of the gaps for the 4 devices.

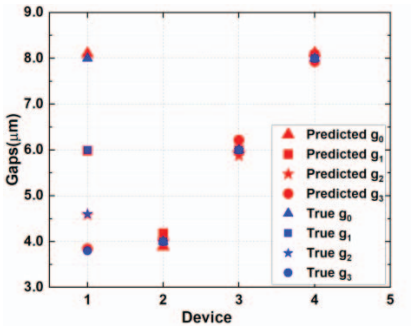


Fig.6 The visualization of the predicted values and the true values of the gaps designed for different devices. The gaps of device1 are designed for the resonator-based DAC in [5]. The gaps of device2, 3, and 4 are designed for the full adder device in which the gaps of all inputs are equal.

The performance of the trained model can be evaluated by the error functions MAE and MSE. The error of the model prediction is found to be 0.0755±0.0094 μm by calculating the two evaluation functions' values. Since the real gaps are not scaled, the error is smaller than 1%, which can be considered negligible compared to common fabrication errors and defects. The theoretically estimated results provided in [5] and the predicted results are compared in Table II using evaluation functions, which confirmed the error value of the predicted results is smaller than the estimated results.

It is very important to highlight the DFC design speedup using ANN modeling compared to conventional techniques. Although generating the dataset for the ANN model using COMSOL software is very time-consuming (a single run with one airgap set takes around 30 mins, and 32 days for 1551 runs), this is only done once. This could be almost the same time consumed to optimize each new design using COMSOL only. However, for the ANN model, this is only done once to generate the dataset. While the training process itself consumes only 30 mins, the final model can predict airgap values for new designs instantaneously without going through numerous simulations. Furthermore, the principal component analysis can be used to reduce the inputs' dimensions and accelerate the training speed. However, it will add noise to the results and decrease accuracy.

TABLE II. Comparison of the True Values, Theoretically-Estimated Results, and the ANN Predicted Values.

Gaps	True value[5]	Estimated in [5]	Predicted
g_0	8 μm	8 μm	7.969 μm
g_1	6 μm	6.349 μm	6.118 μm
g_2	4.6 μm	5.039 μm	4.679 μm
g_3	3.8 μm	4 μm	3.764 μm
Mean Absolute Error		0.247	0.066
Mean Square Error		0.0886	0.0056

V. Conclusion

A fully-connected feed-forward ANN is trained for the MEMS resonator-based DFC design and the parameters can be precisely predicted for different applications, which verifies the model's accuracy. It is worth noting that the proposed model can be made more generic to predict airgap values for different resonator dimensions and different frequency ranges. The negligible error in predicting design parameters, compared to normal fabrication errors, demonstrates that machine learning can significantly improve the efficiency of MEMS resonator-based device and circuit design and pave the way for the implementation of more complex multi-input/output digital circuits and arithmetic units.

Acknowledgment

The authors would like to thank Chenxin Xiong for her valuable insight into the machine learning models.

References

[1] D. J. Frank, "Power-constrained CMOS scaling limits," *IBM J. Res. Dev.,* vol. 46, no. 2.3, pp. 235-244, 2002.

[2] M. A. A. Hafiz, L. Kosuru, and M. I. Younis, "Microelectromechanical reprogrammable logic device," *Nature communications,* vol. 7, no. 1, pp. 1-9, 2016.

[3] M. A. Al Hafiz, S. Ilyas, S. Ahmed, M. I. Younis, and H. Fariborzi, "A microbeam resonator with partial electrodes for logic and memory elements," *IEEE Journal on Exploratory Solid-State Computational Devices and Circuits,* vol. 3, pp. 83-92, 2017.

[4] S. Ahmed, S. Ilyas, X. Zou, N. Jaber, M. I. Younis, and H. Fariborzi, "A compact adder and reprogrammable logic gate using micro-electromechanical resonators with partial electrodes," *IEEE Transactions on Circuits and Systems II: Express Briefs,* vol. 66, no. 12, pp. 2057-2061, 2019.

[5] S. Ahmed, X. Zou, N. Jaber, M. I. Younis, and H. Fariborzi, "A Low Power Micro-Electromechanical Resonator-Based Digital to Analog Converter," *Journal of Microelectromechanical Systems,* vol. 29, no. 3, pp. 320-328, 2020.

[6] X. Zou, S. Ahmed, S. M. S. Kazmi, P. d. Costa, M. I. Younis, and H. Fariborzi, "A Nanoelectromechanical Resonator-Based Flash Style Analog to Digital Converter," *2020 IEEE 33rd International Conference on Micro Electro Mechanical Systems (MEMS),* pp. 822-825, 2020.

[7] I. Goodfellow, Y. Bengio, and A. Courville, "Deep Learning, ser. The Adaptive Computation and Machine Learning Series," ed: Cambridge, MA: The MIT Press, 2016.

[8] G. Araghi, "Temperature compensation model of MEMS inertial sensors based on neural network," in *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, 2018: IEEE, pp. 301-309.

[9] Y. Lee, Y. Park, F. Niu, B. Bachman, K. C. Gupta, and D. Filipovic, "Artificial neural network modeling of RF MEMS resonators," *International Journal of RF and Microwave Computer-Aided Engineering: Co-sponsored by the Center for Advanced Manufacturing and Packaging of Microwave, Optical, and Digital Electronics (CAMPmode) at the University of Colorado at Boulder,* vol. 14, no. 4, pp. 302-316, 2004.

[10] G. L. Creech, B. J. Paul, C. D. Lesniak, T. J. Jenkins, and M. C. Calcatera, "Artificial neural networks for fast and accurate EM-CAD of microwave circuits," *IEEE Trans. Microwave Theory Tech.,* vol. 45, no. 5, pp. 794-802, 1997.

[11] Z. Marinković *et al.*, "ANN based inverse modeling of RF MEMS capacitive switches," in *2013 11th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services (TELSIKS)*, 2013, vol. 2: IEEE, pp. 366-369.

[12] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.

[13] A. Cowen, G. Hames, D. Monk, S. Wilcenski, and B. Hardy, "SOIMUMPs design handbook," *MEMSCAP Inc,* pp. 2002-2011, 2011.