# Class-E Power Amplifiers Incorporating Fingerprint Augmentation With Combinatorial Security Primitives for Machine-Learning-Based Authentication in 65 nm CMOS

Yuyi Shen, *Graduate Student Member, IEEE*, Jiachen Xu, *Graduate Student Member, IEEE*, Jinho Yi, *Graduate Student Member, IEEE*, Ethan Chen, and Vanessa Chen, *Member, IEEE*

***Abstract*— One means by which the security of Internet-of-Things (IoT)-enabled devices may be augmented is through radio-frequency fingerprinting-based authentication methods. As variability in CMOS processes increases with technology scaling, the hardware imperfections that form RF fingerprints can be controlled with small reconfigurable elements, enabling the feasibility of RF fingerprinting as a low overhead security measure for device authentication. To achieve rapid RF identification, we present an inherently secure RF power amplifier and a convolutional neural network-based machine learning classifier through an exploration of combinatorial randomness and self-aware detection mechanisms. By selecting different subsets of thinly sliced power amplifier elements, combinations of random process variations are exploited and updated to form a large search space of distinct RF fingerprints and improve fingerprint prominence. The rich features enabled by augmented device primitives are updated in a time-varying manner to strengthen built-in hardware security. Measurement results demonstrate the effectiveness of this approach at generating distinguishable RF fingerprints across a significant number of configurations.**

***Index Terms*— Hardware security, radio-frequency fingerprinting, integrated circuits, power amplifiers, combinatorial randomness, random intra-die variations, deep learning.**

## I. Introduction

THE increasing popularity of the Internet of Things (IoT) paradigm has yielded an explosive increase in the data production and diversity of the wireless ecosystem [1]. While this growth has the potential to increase the quality of life and enable business models to deliver higher levels of service and customer satisfaction, the advantages of the Internet of Things come coupled with major security concerns stemming from a mix of lack of security measures for cost-effective electronics and a strong incentive for attackers for whom successful attacks may yield disproportionate dividends [2].

Threats to IoT systems include malign data injection and node cloning at the physical layer, collision attacks and channel congestion attacks at the network layer, and malware at the application layer. Addressing these threats is complicated by the natural resource constraints of IoT nodes, which severely limit the usage of generic encryption algorithms for maintaining the confidentiality and force the usage of lightweight authentication processes [1].

Physical layer (PHY) security has been investigated [3]–[5] as one of the means of augmenting the security of IoT systems and can serve as an additional barrier against node cloning attacks. Supposing an attacker succeeds in collecting valid authentication credentials from a weak node in a network, they will also need to subvert PHY security to authenticate to the network. PHY security approaches can be grouped into three general subsets: channel state information-based authentication, signal watermarking-based authentication, and device-specific RF fingerprint authentication [3]. The first type relies on such channel-dependent information as received signal strength, angle-of-arrival, and channel state information to authenticate a device, and is highly unreliable in a dynamic environment. The second involves the embedding of a low-power authentication signal into the original transmission. Lastly, device-specific RF fingerprint authentication relies on the extraction of features imprinted on wireless signals by device-dependent nonidealities such as gain and phase imbalances between the in-phase (I) and quadrature (Q) paths of transmitters and power amplifier nonlinearity [4].

Device-specific RF fingerprinting has been shown to be an efficient way of distinguishing between signals from different transmitters, but often requires the usage of resource-intensive modules to extract the features produced by the aforementioned device-dependent nonidealities [3]. Past works have implemented this through the use of the fast Fourier transform (FFT) to compute the power spectra of signal turn-on transients [5], the generation of density-mapped images from signals' signal space representations [4], or the usage of the FFT with I/Q samples of entire packets [6].

Manuscript received September 30, 2021; revised December 14, 2021; accepted January 4, 2022. This work was supported by the National Science Foundation under Grant 1952907, Grant 1953801, and Grant 2028893. This article was recommended by Associate Editor B. Zhao. *(Corresponding author: Ethan Chen.)*

The authors are with the Electrical and Computer Engineering Department, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: yuyis1@andrew.cmu.edu; jxu3@andrew.cmu.edu; jinhoy@andrew.cmu.edu; ethanchen@cmu.edu; vanessachen@cmu.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSI.2022.3141336.

Digital Object Identifier 10.1109/TCSI.2022.3141336

However, the effectiveness of any RF fingerprinting scheme is ultimately contingent on the distinctiveness of individual devices' features. Although it is axiomatic that even distinct wireless devices with the same design will exhibit different RF fingerprints as a result of random variation in the manufacturing process, the actual extent to which two devices' RF fingerprints may be distinguished from one another is dependent on the device design and the specifics of the manufacturing process.

To enhance the effective user capacity of RF fingerprinting-based systems, there is a need to integrate configurability on the transmit-side of RF fingerprinting systems. Configurability adds dimensionality to the random variation that results in the RF fingerprint of a device, increasing the range across which RF fingerprints may occur and thus improving the user capacity of the system. Past works have implemented it by modifying the response of the wireless channel with a configurable reflector positioned in proximity to the transmit antenna [7], or by adding a functional block at the transmitter to modify I/Q samples prior to transmission [8].

There exist limited options for imprinting configurable RF fingerprints on a given signal with hardware. Deliberately introducing differential nonlinearity (DNL)/integral nonlinearity (INL) into the transmitter-side DACs in order to modulate the transmitter nonlinearity is difficult to implement without severely compromising system performance. The same is true of introducing mismatches between transmitter in-phase and quadrature signal paths and modulating mixer nonlinearity. Configurable RF filters for applying RF fingerprints through subtly shaping the power spectrum of the transmitted signal are possible but require significant design effort to avoid overly distorting the signal across configurations.

In this work, we propose an RF fingerprinting system with transmitter-side configurability based on a reconfigurable power amplifier (PA). By selecting subsets of thinly sliced PA elements, the PA's transfer functions are altered, yielding distinctive fingerprints that may be classified with conventional machine-learning (ML)-based techniques. Random process variations ensure inter-subset distinctiveness, while the cumulative parasitics of the PA slices maintain normal operation across all configurations. Testing is conducted with modulated input signals at 2.4 GHz and recordings made at baseband are passed through a machine-learning classifier for extracting and classifying the fingerprints.

The proposed approach for introducing RF fingerprint variability in order to enhance the effective user capacity of an RF-fingerprinting system is both integrable on-chip, unlike the meta-surface reflector proposed in [7], and does not overly impact the performance of the configurable block. Furthermore, a wide range of variability is achievable in exchange for moderate additional area consumption.

The remainder of this paper is organized as follows. Section II describes the system architecture and the impact of $V_{th}$ and $\beta$ variation on a number of attributes of the PA output signal and goes over the structure of the machine learning model used to classify said signal. Section III describes the measurement results and performance attained with the RF fingerprinting system in a wireless communication system. Section IV describes an FPGA implementation of the receiver-side machine learning classifier, and Section V concludes the paper.

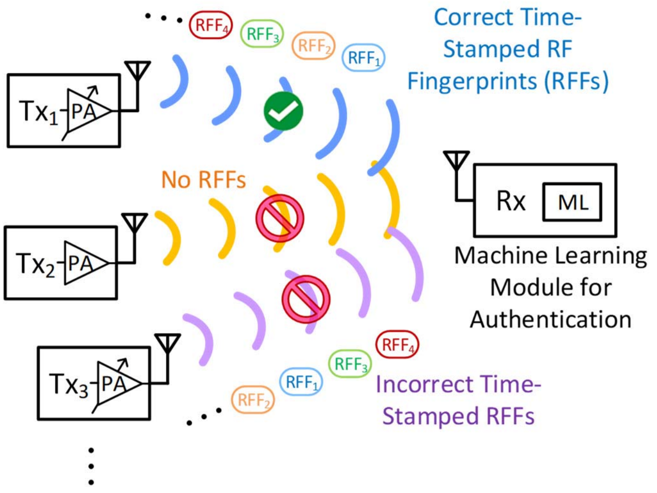

Correct Time-Stamped RF Fingerprints (RFFs)
RFF4
RFF3
RFF2
RFF1
Tx1
PA
No RFFs
Tx2
PA
Rx
ML
Machine Learning Module for Authentication
Tx3
PA
RFF4
RFF3
RFF1
RFF2
Incorrect Time-Stamped RFFs

Fig. 1. The overview of the presented system. Only the devices with correct time-varying RF fingerprints (RFFs) will be authenticated by the machine-learning classifier in the receiver.

## II. System Architecture

### A. System Overview

The overview of the presented system is shown in Fig. 1. Transmitters will be authenticated when they manifest the correct time-varying RF fingerprints (RFFs). The main transistor of the transmitter PA is divided into thin selectable slices, a subset of which is enabled to configure the PA fingerprint. The combinatorial random variations present in sub-micron CMOS processes introduce significant threshold voltage ($V_{th}$) and transconductance parameter ($\beta$) mismatch between the selectable slices [9], injecting distinct features into the PA output. The overall fingerprint embedded into the transmitted signal is a result of the composite impact of the variations presented within the enabled slices on the transfer function of the PA.

In order to filter out malicious attackers from the signals transmitted from local transmitters through time-varying RF signatures, a convolutional neural network (CNN) classifier is applied on the receive-side to extract signal features and reconstruct the original element-selection bitstream map to determine the differences between the time-stamped RFFs, as shown in Fig. 2.

In a system with $N$ reconfigurable transmitters, the receive-side ML classifier would need to be trained across all $M$ possible configurations of each transmitter for a total of $MN$ configurations. $M$, the number of configurations to be used for each transmitter, would be chosen to balance the number of possible RFF sequences each transmitter could support and the training overhead of the classifier. Information on the specific RFF sequence to expect from a transmitter would be exchanged during the handshake process in the form of a sequence of pointers to a look-up table of RFFs.

Although it is feasible for an attacker to apply blind PA modeling and predistortion to mimic the RFF of a transmitter

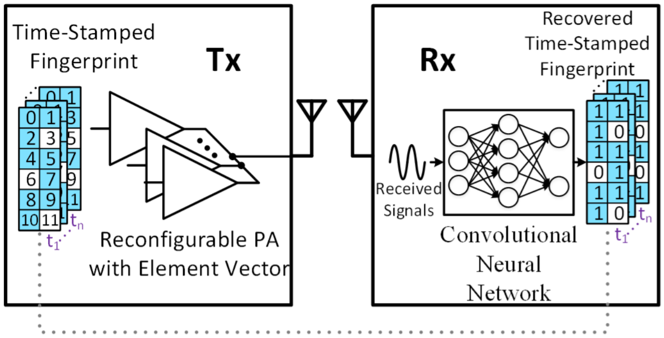


Fig. 2. The transmitter utilizes combinatorial security primitives to augment RF signatures for secure data transmission, while the convolutional neural network (CNN) module is exploited to perform device authentication in the receiver. The time-varying RFFs are synchronized with reconfigurable maps between the transmitters and receivers.

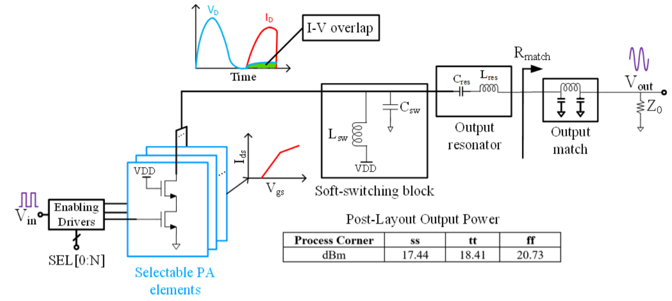
in a typical RF fingerprinting system using an approach similar to that of [10], the time-varying nature of the proposed system's RFFs would require an attacker to apply PA modeling to each observed RFF and perform predistortion using the correct sequence of fitted PA models, a task that requires significantly more resources on the part of the attacker. Furthermore, the configurability of the transmit-side RFFs permits changes to the transmitter RFF sequences that can force an attacker to re-train their PA models.

Because of the extreme popularity of the unlicensed 2.4 GHz ISM-band, especially for such IoT communications protocols as IEEE 802.15.4 and Bluetooth Low Energy (BLE) [11], the system design was targeted at this frequency band. A nonlinear high-efficiency Class E power amplifier topology was chosen for the transmit-side PA in recognition of the power constraints IoT systems are subject to.

## B. Reconfigurable PA Structure

To validate our approach for generating configurable RFFs, a single-ended 2.4 GHz Class E PA was designed in a 65 nm CMOS process. Configurability was implemented by slicing the main switching transistors into 12 selectable elements. PA elements are selected by enabling individual PA transistor gate drivers with a rudimentary scan chain. Choosing 9 elements out of the 12 to be enabled yields a large search space of 220 configurable subsets while avoiding a significant reduction in output power. With 9 transistor elements selected and a $V_{DD}$ of 1.2 V, the PA outputs 17.4 – 20.7 dBm of power across process corners in post-layout simulation after parasitic lumped and coupling capacitances are extracted, roughly 1.5 dB down from the output power resulting from enabling all 12 elements. The full schematic and range of output power across process corners are displayed in Fig. 3.

Configurability aside, cascoding with a bias voltage of $V_{DD}$ is applied to lower the voltage stress across the switching transistor drains so as to account for the high drain voltages characteristic of Class E operation. The switching transistors are placed across an off-tuned series LC tank that brings the drain voltage back down to zero at switching instants, and fundamental power is extracted through a series fundamental frequency resonator connected to the drain. Output power is set



| Process Corner | ss | tt | ff |
|---|---|---|---|
| dBm | 17.44 | 18.41 | 20.73 |

Fig. 3. The Class-E power amplifier is designed with 12 selectable sections that consist of the driver and the power-amplifier switch shared with the RF choke and the output matching network. Enabling 9 elements from the 12 selectable ones results in a search space of 220 configurable subsets to generate useful security primitives.

by matching the load to the corresponding resistance. On the input-side, inverter pre-drivers are sized and collocated with the selectable PA transistors to provide sharp switching to achieve better drain efficiency.

## C. PA Fingerprint Analysis

As stated in Pelgrom's seminal paper [9], the standard deviations of the $V_{TH}$ and $\beta$ mismatch between two MOSFETs of the same size is approximately proportional to the square root of the MOSFET gate area. Given that the variances of independent random variables sum, it is then possible to derive a "self-mismatch" for $V_{TH}$ and $\beta$:

$$\sigma_\beta = \frac{1}{\sqrt{2}} \frac{\beta A_\beta}{\sqrt{WL}} \tag{1}$$

$$\sigma_{VT} = \frac{1}{\sqrt{2}} \frac{A_{VT0}}{\sqrt{WL}} \tag{2}$$

Variations in these device parameters within both the main PA transistors and the inverter gate drivers ultimately superimpose themselves on top of the transmitted signal, yielding the overall PA fingerprint. Intuitively, $\beta$ variations within the PA transistors most impact the final height of the drain current waveform in the PA transistor on-state by setting the PA transistor on-resistance, while $V_{TH}$ variations in the PA transistors and $\beta$ variations in the inverter gate driver transistors primarily affect the timing of the PA switching action. To simplify this analysis, it is assumed that the inverter gate drivers see a trapezoidal input signal with negligible rise times, so that $V_{TH}$ variations in the inverter gate drivers can be mostly ignored outside of their impact on the gate driver transistor on-resistances. Because the reactive Class E soft-switching LC tank restores the drain voltage to zero at switching instances, it can also be assumed that the PA transistors directly cross between cut-off and triode.

*1) Fourier Analysis of PA Drain Current:* Modeling the inverter gate driver of an individual element as a pair of switched resistances connected to the PA transistor gate capacitance produces the gate voltage waveform depicted in Fig. 4. The associated drain current waveform is approximately exponential as a result of the series LR circuit formed by the PA transistor on-resistance $R_{ON}$ and $L_{SW}$, the inductor of the soft-switching block of the drain network. The effect of the

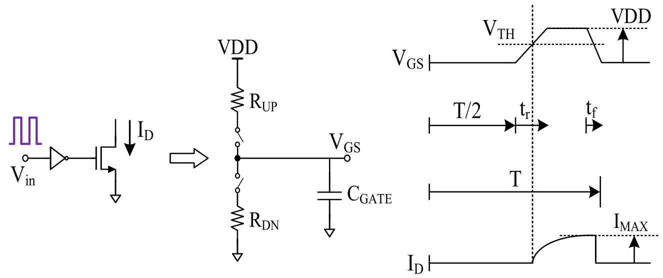


Fig. 4. The $V_{GS}$ waveform produced by the inverter gate driver at the PA transistor input, and the resultant exponential drain current waveform. The turn-on and turn-off times of the drain current are set by the rise and fall times produced by the inverter gate driver, which are impacted by variations in gate driver transistor $\beta$. $I_{MAX}$, the peak transistor current the drain current approaches in the on-state, is set by and impacted by variations in the PA transistor $\beta$.

drain capacitance can be ignored because of it being shunted by the comparatively greater conductance of the PA transistor. T denotes the period of the 2.4 GHz carrier frequency ($f_c$).

The rise and fall times ($t_r$, $t_f$) of the $V_{GS}$ waveform produced by the inverter gate driver are proportional to the time constant of the overall RC system, and can be written as:

$$t_r, t_f = \tau \times \ln(9) \tag{3}$$

$$\tau = R_{DN}C_{GATE}, \quad R_{UP}C_{GATE} \tag{4}$$

$$R_{DN}, R_{UP} = \frac{1}{V_{OV}\beta}, \quad V_{OV} = VDD - V_{TH} \tag{5}$$

The asymptote approached by the exponential drain current waveform, $I_{MAX}$, is simply set by the on-resistance of the PA transistor:

$$I_{MAX} = \frac{VDD}{R_{on}} \tag{6}$$

Of the variables mentioned here, note that $R_{DN}$, $R_{UP}$, and $I_{MAX}$ are random variables with distributions dependent on those of the $V_{TH}$, $\beta$ parameters associated with the inverter gate driver and PA transistors. Writing the exponential equation for the PA transistor on-state drain current and integrating over one period of the 2.4 GHz carrier frequency enables the derivation of the frequency content of the PA drain current over the duration of the on-state:

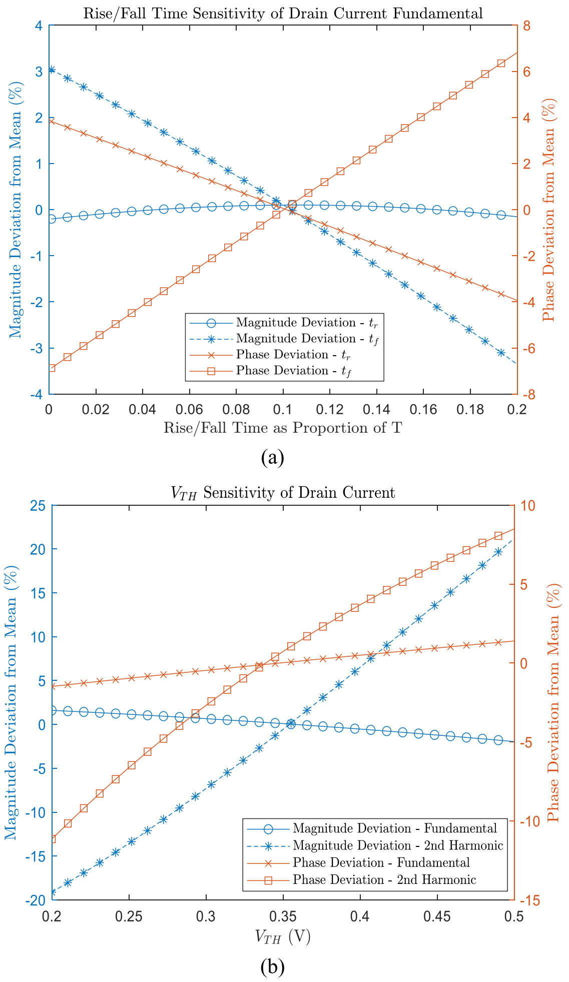
$$I_{D,ON}(\mathrm{t}) = I_{MAX}\left(1 - e^{-\frac{t}{\tau}}\right), \quad \tau = \frac{L_{SW}}{R_{on}} \tag{7}$$

$$I_{D,ON}(\omega) = I_{MAX}\left(\frac{1}{j\omega}(A - B) + \frac{1}{j\omega + \tau^{-1}}(C - D)\right) \tag{8}$$

$$A = \mathrm{e}^{\left(-j\omega\left(0.5T + \frac{t_r V_{TH}}{VDD}\right)\right)}, \quad B = \mathrm{e}^{\left(-j\omega\left(T - \frac{t_f V_{TH}}{VDD}\right)\right)} \tag{9}$$

$$C = e^{\left(-(j\omega+\tau^{-1})\left(T - \frac{t_f V_{TH}}{VDD}\right)\right)}, \quad D = e^{\left(-(j\omega+\tau^{-1})\left(0.5T + \frac{t_r V_{TH}}{VDD}\right)\right)} \tag{10}$$

The periodic occurrence of the on-state at 2.4 GHz permits the derivation of Fourier coefficients $C_{n,ID}$ for the full drain current waveform of the PA element, where n denotes the harmonic number:



(a)

(b)

Fig. 5. Empirical calculation is used to conduct a sensitivity analysis of drain current frequency components to changes in (a) rise and fall time of $V_{GS}$ and (b) PA transistor $V_{TH}$.

$$C_{n,ID} = f_c I_{D,ON}(n \times 2\pi f_c) \tag{11}$$

The overall drain current of the PA can then be determined by summing the coefficients $C_{n,ID}$ over the total number of selected elements.

*2) PA Drain Current Sensitivity Analysis:* The values of a single PA element's drain current waveform's Fourier coefficients corresponding to the fundamental and 2nd harmonic were evaluated across variations in $V_{GS}$ rise and fall time, and PA transistor $V_{TH}$ so as to assess the sensitivity of drain current properties to variations in $V_{TH}$ and $\beta$ within the inverter gate driver and the PA transistor. For this calculation, IMAX was set to one, while VDD and $\tau$ were set to typical values for a PA design in the 65 nm process. Afterwards, the percent deviations of said Fourier coefficients' magnitudes and angles from their mean values in the sweep were calculated and plotted in Fig. 5.

It can be seen from Fig. 5a that the drain current frequency content is comparatively more sensitive to variations in VGS fall time as opposed to rise time. This is consistent with the behavior associated with Class E operation – PA transistor

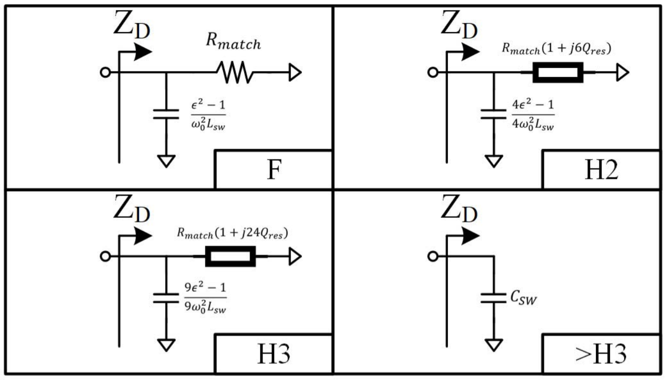

$Z_D$
$R_{match}$
$\frac{\epsilon^2 - 1}{\omega_0^2 L_{sw}}$
F
$Z_D$
$R_{match}(1 + j6Q_{res})$
$\frac{4\epsilon^2 - 1}{4\omega_0^2 L_{sw}}$
H2
$Z_D$
$R_{match}(1 + j24Q_{res})$
$\frac{9\epsilon^2 - 1}{9\omega_0^2 L_{sw}}$
H3
$Z_D$
$C_{SW}$
>H3

Fig. 6. Each selectable PA element sees the above equivalent drain impedances at the harmonics of the operating frequency. Beyond the 3rd harmonic, the drain capacitance $C_{SW}$ dominates the impedance seen by the switching elements and provides an approximate harmonic short.

drain voltage is brought down to zero at the transition to the on-state, lowering the sensitivity of the PA transistor current to marginal changes in the timing of the turn-on instant. Furthermore, all of the PA drain current is abruptly shunted through the drain capacitance at the turn-off instance, increasing the potential for changes in the timing of the turn-off instance to strongly impact the PA operation.

Fig. 5b indicates that the 2nd harmonic component of the drain current is significantly more sensitive to variations in PA transistor $V_{TH}$ than the fundamental. Because the magnitudes of the harmonics are an indicator of the overall power amplifier nonlinearity, this implies that random variations in the PA transistor $V_{TH}$ significantly impact the PA's overall nonlinearity, which has been shown to introduce distinguishable RFFs into the transmitted signal [6].

Overall, the frequency domain sensitivity analysis indicates that variations in $V_{TH}$ and $\beta$ within an individual selectable PA element can produce significant RFFs that superimpose upon the transmitted signal when the element is selected by the PA reconfiguration process.

*3) Drain Network-Induced Effects:* The summed drain current drawn by the selected PA elements is converted to the overall $V_{DS}$ seen across all elements by the harmonic terminations supplied by the common PA drain network. The approximate equivalent impedances seen at the common drain node are shown in Fig. 6 and are primarily composed of the equivalent capacitance of the parallel LC tank $L_{SW}||C_{SW}$ responsible for restoring the drain voltage towards zero at switching transitions and the impedance of the fundamental frequency resonator that couples the common drain node with the PA load. $Q_{res}$ is simply the quality factor of this coupling resonator at the fundamental that results from the equivalent load resistance $R_{MATCH}$, while the parameter $\varepsilon$ is a property of the drain voltage restoration tank that is defined as:

$$\epsilon = \frac{\omega_{fund}}{\omega_{res,LC}} \tag{12}$$

This parameter is typically set to some value greater than or equal to unity in consideration of the PA topology's power capacity curve, which approaches an asymptote of 0.102 as $\varepsilon$ approaches infinity and is fairly constant for $\varepsilon$ greater than 1.5 [12]. For values of $\varepsilon$ near unity, the effective impedance of the LC tank peaks near the fundamental and causes the equivalent impedance seen at the common drain node to be that of the coupling resonator.

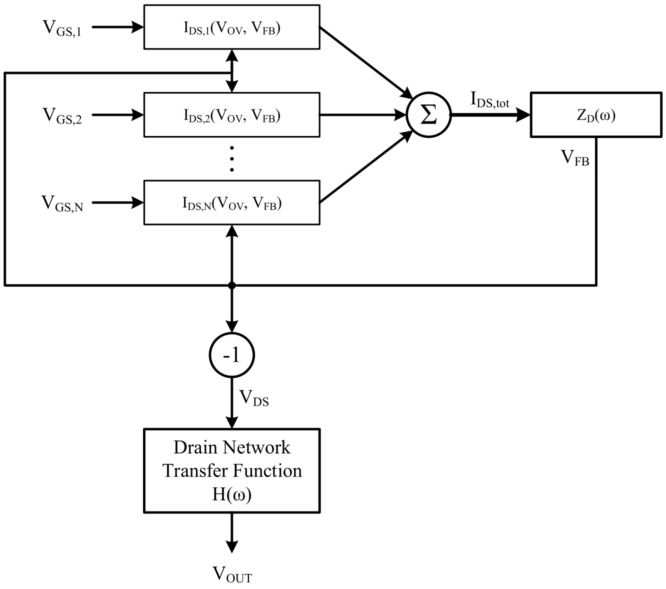

$V_{GS,1}$
$I_{DS,1}(V_{OV}, V_{FB})$
$V_{GS,2}$
$I_{DS,2}(V_{OV}, V_{FB})$
$V_{GS,N}$
$I_{DS,N}(V_{OV}, V_{FB})$
Σ
$I_{DS,tot}$
$Z_D(\omega)$
$V_{FB}$
-1
$V_{DS}$
Drain Network Transfer Function H(ω)
$V_{OUT}$

Fig. 7. The full PA system can be expressed as a feedback system with summed currents from selected PA elements, feedback from $V_{DS}$ to $I_{DS}$ through the equivalent impedance seen at the common PA drain $Z_D(\omega)$, and the final transformation from $V_{DS}$ to the PA output $V_{OUT}$.

The full PA system can be roughly modeled as a set of summed memory-less nonlinear current source blocks corresponding to the $I_{DS}$ curves of each selected PA element, whose summation feeds the PA drain network transfer function after being converted to $V_{DS}$ by the equivalent drain network impedance [13] as shown in Fig. 7. Feedback from $V_{DS}$ to each of the current source blocks is present as a result of the coupled relationship between $I_{DS}$, $V_{DS}$, and $Z_D$, the equivalent drain network impedance:

$$V_{DS} = -I_{DS} Z_D \tag{13}$$

$$I_{DS} = \mathrm{G_m} V_{ov} \left(1 - e^{\frac{-V_{DS}}{V_{knee}}}\right) = G_m V_{ov} \left(1 - e^{\frac{V_{FB}}{V_{knee}}}\right) \tag{14}$$

$$V_{ov} = V_{GS} - V_{TH} \tag{15}$$

Intuitively, the reactive nature of the drain network impedance causes the $Z_D$ block to exhibit a memory span of some time $M_D$ over which the output signal $V_{FB}$ depends on the input $I_{DS,tot}$. Because the memory span of the overall PA system must be at least as long as that of the feedback $Z_D$ block, we can take $M_D$ as a lower bound for the overall system memory. Plugging in typical values for 2.4 GHz PA drain network components and empirically determining the impulse response of the $Z_D$ block shows in Fig. 8 that $M_D$ can span well over several periods of the fundamental frequency for a typical design as a result of the relatively lightly damped resonant tanks present in the drain network.

Although the drain network is common to all PA elements and thus does not directly contribute to the variations that result in the PA's configurable RFF, the memory effect

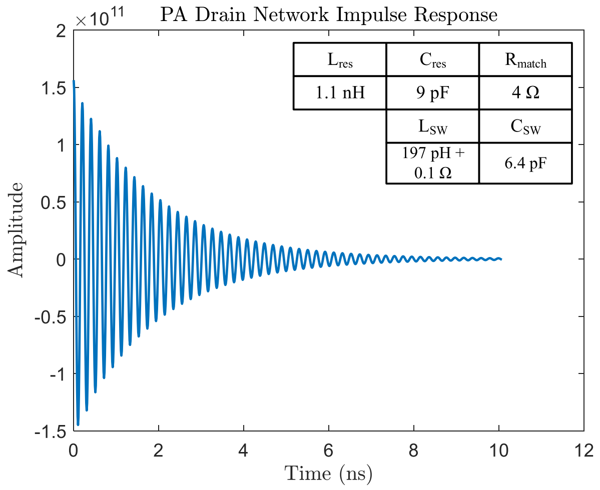


Fig. 8. Typical component values for a 2.4 GHz design were used to calculate the impulse response of the feedback block $Z_D(\omega)$. It is clear from the depicted trace that the memory of $Z_D$ easily spans over 10 periods of the fundamental frequency.

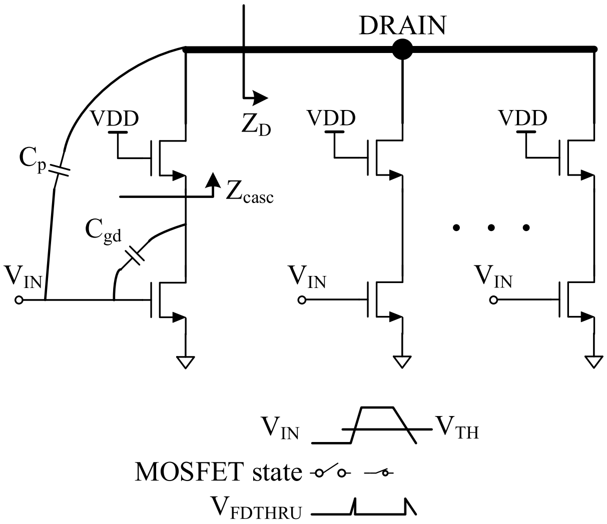


Fig. 9. Parasitic capacitive coupling between the input and transistor drain nodes results in mixing action of the input signal by the switching of the PA transistors. Finite rise and fall times at the PA transistor gates cause the waveform resulting at the internal cascode nodes and common drain node from the mixing of the capacitive feedthrough signal to appear as a series of small asymmetrical spikes.

produced by its lightly damped elements effectively smears the features generated by the variations in PA $V_{TH}$, $\beta$, $t_r$, and $t_f$ over several periods of the carrier, changing the "appearance" of the fingerprints over time.

*4) Input Signal Feedthrough Mixing:* In addition to the RFFs directly produced by the modulation of the PA nonlinearity through variations in transistor $\beta$, $V_{TH}$, and PA transistor gate driver rise/fall times, the feedthrough of the PA input signal through such parasitic capacitances as the inherent transistor gate-drain capacitors and electromagnetic coupling between gate and drain metal lines in the layout is mixed by the switching action of the PA transistor to yield an additional feature.

Intuitively, the cascoded transistor of a given PA element whose gate is driven with the finite rise and fall times $t_r$ and $t_f$ will turn on when the gate voltage increases beyond the transistor $V_{TH}$ at slope $V_{DD}/t_r$ and shut off when the gate voltage decreases below $V_{TH}$ at slope $V_{DD}/t_f$. When the transistor is on, it pulls both the common PA drain node and the internal cascode transistor nodes towards the ground. When it is off, however, the voltages at the common drain node and internal cascode nodes are free to move about in accordance with the energy stored in the common drain load network and any signals coupled in via parasitic capacitances. The result is that the periodically switching transistor acts upon the latter in the same manner of a basic current commutating mixer's current steering differential pair on an RF signal fed in through the tail transistor.

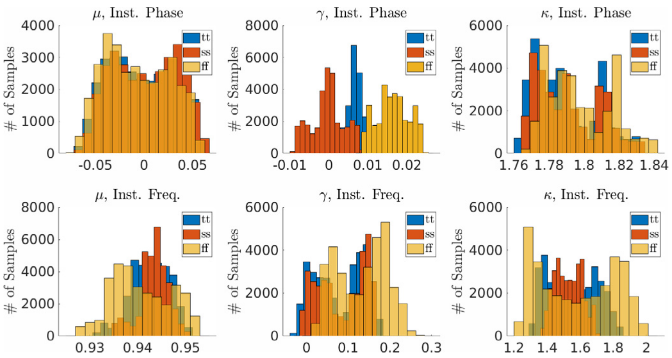
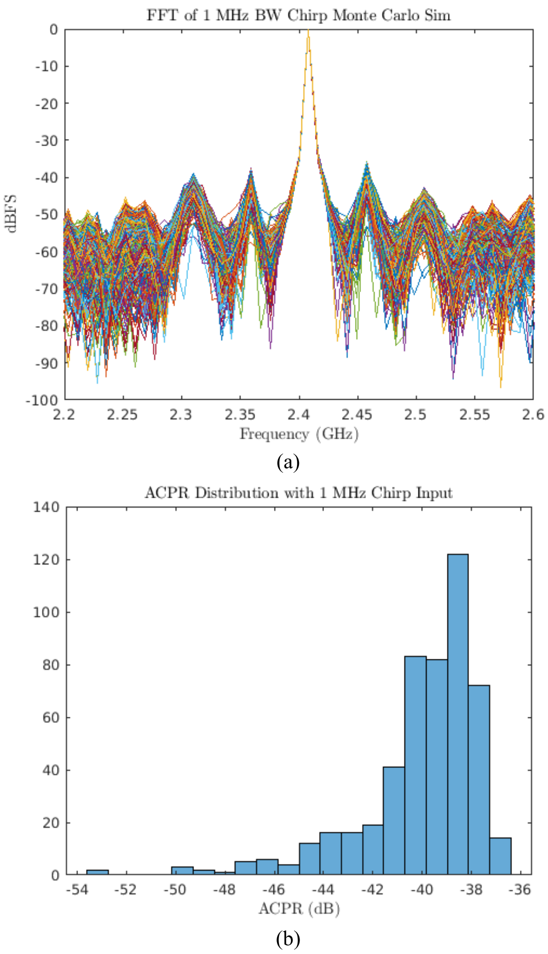


Fig. 10. Histograms were generated for the Monte Carlo simulations for the 12-slice Class E amplifier with 9 elements enabled. Each signal sample is normalized to its peak amplitude and the mean ($\mu$), skewness ($\gamma$), and kurtosis ($\kappa$) of the sample's instantaneous phase and frequency were calculated. The results show the full range of RF fingerprints supported by the configuration as induced by random mismatch.

Fig. 9 shows the shape of the ensuing time-domain waveforms produced at the internal cascode and common drain nodes as a result of parasitic capacitive coupling to the input signal. The sharp "spikes" of the mixed waveform indicate that it is rich in spectral content that is ultimately superimposed upon the output waveform. Because the slopes of the spike edges are inversely proportional to the rise and fall times of the PA element's $V_{GS}$ waveform while the heights are directly proportional to the PA element's $V_{TH}$, the nature of this spectral content is strongly dependent on random variations in input gate driver transistor $\beta$ and PA transistor $V_{TH}$.

*5) Monte Carlo Simulation for RFF Distribution:* To further verify the dependence of the PA RFF on $\beta$ and $V_{TH}$ variations that can be selected by enabling a subset of PA elements, Monte Carlo mismatch simulation was conducted using the 65 nm PDK following post-layout extraction, with 9 PA slices enabled out of 12. Each run used transient simulation to calculate the output waveform using a single-tone 2.4 GHz input signal. As an archetypal RFF, the Hilbert transform was used to calculate RF-distinct native attributes (RF-DNA) fingerprints from the simulation data [14]. Each signal sample for which an RFF was calculated was normalized to its peak amplitude and the mean, skewness, and kurtosis of the sample's instantaneous phase and frequency were calculated. The resultant histograms are displayed in Fig. 10 and show significant variation.

Similar reasoning to that which was used in the drain current Fourier analysis can be applied to intuitively reason that there also exists a dependence between the instantaneous



(a)

(b)

Fig. 11. Monte Carlo simulations were conducted for enabling 9 of 12 Class E PA elements with a 1 MHz bandwidth chirped input signal. The transient simulation results from the Monte Carlo runs were used to calculated (a) FFT and (b) ACPR metrics.

frequency/phase of the PA output signal and the $V_{th}$s of the selected PA transistor elements. We note that current only flows through the PA transistor block when the PA transistor input voltage exceeds the minimum Vth of the selected PA transistor elements. Thus, the location of the positive and negative edges of the PA transistor current waveform in time effectively depends on the lowest $V_{th}$ value among the selected PA transistor elements. The timing of these edges translates to small changes in the instantaneous phase of the PA output, permitting variations in PA transistor element $V_{th}$ values produced by PA reconfiguration to translate to these RF features.

*6) Monte Carlo Simulation With Chirped Signal:* To verify the presence of a wide distribution of RFFs with a modulated signal as input, a similar Monte Carlo simulation was conducted, but with a 1 MHz bandwidth chirp signal to emulate a bandlimited passband signal. A 4096-point FFT was taken across runs and is shown in Fig. 11a. There appears to be a wide variation of the FFT floor beyond the chirp bandwidth across FFT runs, indicating that a wide distribution of RFFs may be injected into the transmitted signal by the PA. This is further confirmed by calculating the ratio between the power of the chirped output signal and the average power of a 20 MHz-wide band 100 MHz below the 2.4 GHz center frequency, yielding an ACPR figure whose histogram is shown in Fig. 11b. The wide variation is reflective of the results of our sensitivity analysis of the PA drain current frequency content, which appeared to indicate that variations in the PA transistor $V_{TH}$ would yield distinguishable variations in the PA nonlinearity characteristic.

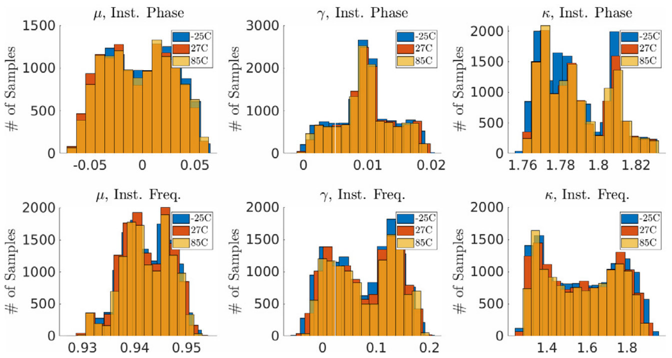


Fig. 12. Monte Carlo simulations for the 12-slice Class E amplifier with 9 elements enabled were performed at the tt corner across temperature corners to demonstrate the impact of temperature on RFF variability. Each signal sample is normalized to its peak amplitude and the mean ($\mu$), skewness ($\gamma$), and kurtosis ($\kappa$) of the sample's instantaneous phase and frequency were calculated in the same manner as in Fig. 10.

*7) Impact of Temperature and Aging on RFF Distribution:* Because of the dependence of transistor parameters such as $V_{TH}$ and $\beta$ on both temperature and aging and the relationship between said transistor parameters and the RFFs produced by the proposed PA's configurations, it can be deduced that temperature and aging both exhibit a significant impact on the set of RFFs associated with the possible PA configurations. Running a schematic-level Monte Carlo mismatch simulation with 9 slices enabled out of 12 at the *tt* corner across a wide temperature range and calculating RF-DNA fingerprints in the same manner as in Section II.C.5 demonstrates this, as the shapes of the resulting distributions of calculated RF-DNA fingerprints visibly change between temperature corners. This is illustrated in the histograms shown in Fig. 12.

Transistor aging can be expected to have a more pronounced effect on the set of RFFs associated with the individual PA configurations, as a result of its unevenly applied nature. Hot carrier injection (HCI), one of the primary mechanisms of device degradation, occurs through the injection of charge into gate dielectric as a result of energetic carriers present in the channel [15], and so ages the individual PA elements that are enabled during operation. Because of the time-varying nature of the transmitter RFFs, and thus the PA configurations in the proposed system, HCI slowly skews the most commonly occurring RFFs in the sequence of fingerprints away from their positions in the RX-side training data distribution, degrading classification accuracy over time.

Because both transistor aging and temperature changes directly skew the transistor parameters connected to the RFFs of the PA's configurations, the RX-side classifier in the proposed system would need to be re-trained for significant temperature changes or after a sufficient period of time for aging to become noticeable.

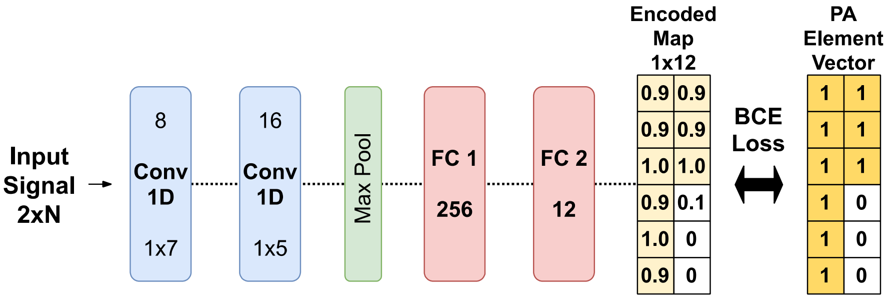


Fig. 13. A convolutional neural network (CNN) model is used for signal classification. The output of the CNN aims to reconstruct the PA element vector.

TABLE I
CNN ARCHITECTURE

| Layer | Output dimensions |
|---|---|
| Input | 2×N |
| 8 Ch 1×5 Conv, Stride = 2 | 8×N/2 |
| 16 Ch 1×5 Conv, Stride = 2 | 16×N/4 |
| Max Pool 1×5, Stride = 2 | 16×N/8 |
| FC | 256 |
| FC | 12 |

### D. Convolutional Neural Network

With excellent performance in computer vision and voice recognition, deep learning techniques have also been investigated for learning features from RF signals. In [16], O'Shea *et al.* suggested exploiting shift-invariant properties of convolutional networks to extract features directly from frames of time series data, without extensive pre-processing. References [8] and [16] demonstrated that deep learning techniques could classify various modulation types or transmitter features of raw RF signals with high accuracy. In this work, we designed a lightweight CNN targeted to be deployed in FPGA for RFF classification.

Table. I shows the overall layout of the CNN architecture that was chosen after examining several candidates for their accuracy and robustness over data with different RF receiver parameters. The model structure is visualized in Fig. 13. This model accepts an input with size (B, N, 2) representing (Batch-Size, number of input samples, I/Q channels of the signal). The 1D convolution with I/Q channels can capture the local temporal features between I/Q symbols. Our CNN model utilizes two 1D convolutional layers, with 8 filters of kernel size 7 and 16 filters of kernel size 5, respectively. After the last convolutional layer, the max-pooling layer with a kernel size of 2 and stride of 2 down samples and propagates the features with a minimal tradeoff in the classification performance, effectively reducing the number of operations and thus lowering the computing effort required. Because our CNN structure is shallow, introducing any skip connections or residual layers does not improve the accuracy of the model. The convolutional layers are followed by two fully-connected (FC) dense layers with a size of 256 in FC1 and 12 in FC2 (output layer) to generate a 12-element output vector. Instead of using one-hot encoding with an output size of 220 to classify the 220 possible combinatorial RFFs generated by the PA, we expected that 9 of the 12-element output vectors will be activated to reconstruct the positions of the selected elements in the PA in a multi-hot encoding style to classify up to 220 RFF classes. As shown in Fig. 13, an exact matching between the indices of the 9 largest elements in the output vector and the indices of the enabled PA element vectors indicates a successful classification. This enables the CNN to account for the possibility that many combinations of the PA elements emit non-prominent RFFs, by providing the ability to drop some combinations and reduce the number of RFF classes. Using this multi-hot encoding makes sure that the CNN's output size always stays constant at 12, and the model can readily adapt to most RFFs configured with different combinations of 12 elements. Also, using the output size of 12 instead of 220 largely reduces the computation cost and the memory requirement of the CNN output layer.

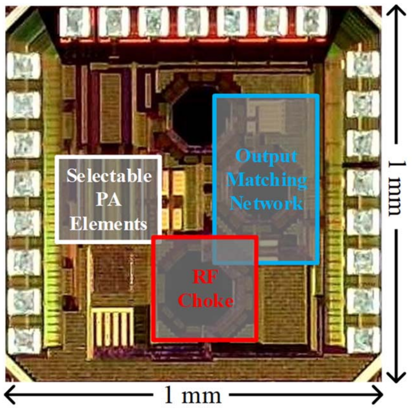
The output of the model is connected to the Sigmoid function as we use binary cross-entropy (BCE) loss function for training the model and we need to make sure the output is numerically stable to be compared with the original binary PA element map. Any other layers' output in the model are connected with the Rectified Linear Unit (ReLU) activation function. The Adam optimizer with the learning rate of 0.001 is used for training the model with batch size of 256. The train-validation-test dataset split is 6:2:2. For the dataset size of 160,600 with receiver ADC oversampling of 4 (input vector size $= 2 \times 160$), each epoch takes 1.2 seconds in average for the model to be trained. After training for 4000 epochs, the model with the lowest validation loss was selected to test the performance on the testing dataset.

## III. MEASUREMENT RESULTS

In order to demonstrate proof of concept, the 12-slice Class E PA was taped out and measured in a simulated wireless communications system using Bluetooth Low Energy advertising packets. The chip micrograph is shown in Fig. 14, with the PA occupying a core area of 0.27 mm$^2$.

### A. Data Collection Methodology

An AD9082-FCMA-EBZ evaluation board was used as both transmitter and receiver with the Class E PA. An image of the measurement setup is shown in Fig. 15. The onboard RF DAC was set to repeatedly transmit BLE advertising packets at the 2.4 GHz carrier frequency, while the ADC sampled the PA output at RF for down-conversion to baseband using a numerically controlled oscillator. Measurements were conducted both



| | |
|---|---|
| Technology | 65nm CMOS |
| Chip Area | 1 mm$^2$ |
| Core Area | 0.267 mm$^2$ |
| Total Selectable Set Number | 4095 |
| Set Number of Choosing 9 from 12 | 220 |

Fig. 14. A chip photo was taken of the power amplifier incorporating fingerprint augmentation with combinatorial randomness in 65nm.

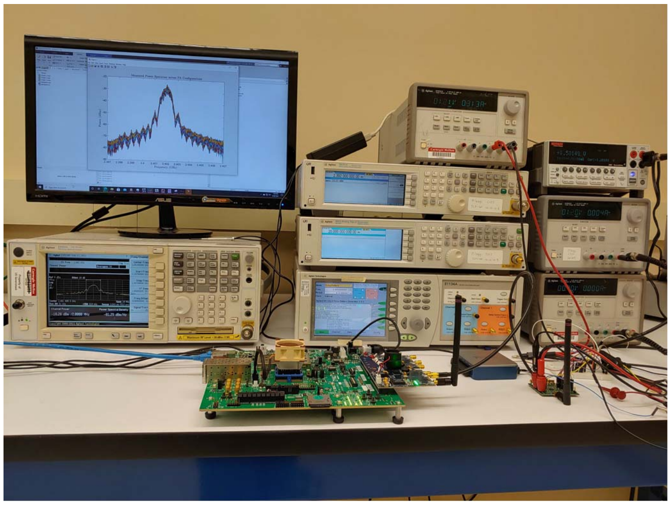


Fig. 15. The measurement setup used to transmit and receive BLE packets over the air with the configurable PA is shown here. In the foreground is the AD9082-FCMA-EBZ evaluation board (mounted on Xilinx ZCU102) and PA test PCB.

over the air (OTA) and with a cable. 730 preamble data are recorded from each of the 220 PA combinatorial configurations in order to have sufficient data points for training and testing the convolutional neural network model. Upon collection, the BLE packet preambles were isolated, making a total dataset size of 160,600, and fed into the convolutional neural network with the structure described in II. D for training, validation, and classification tasks. A split of 6:2:2 was used to partition the collected data between training, validation, and testing datasets, apportioning 146 data points per PA configuration for testing and validation to ensure accurate evaluation of model performance.

### *B. Dataset Overview*

Spectrum analyzer measurements of the PA output signals are displayed across the 220 PA configurations in Fig. 16. Mean power across the 1 MHz bandwidth of the signal varies between ±2 dB across all 220 configurations.

The resultant dataset was inspected with a dimensionality reduction technique known as t-Distributed Stochastic Neighbor Embedding (t-SNE) [17] to determine the presence of clusters in the dataset corresponding to the different PA configurations. t-SNE in particular is a variant of Stochastic Neighbor Embedding that aims to capture the local structure of high-dimensionality data while revealing global structures using an approach that seeks to alleviate what is known as the "crowding problem" and a number of performance issues associated with other methods [18].

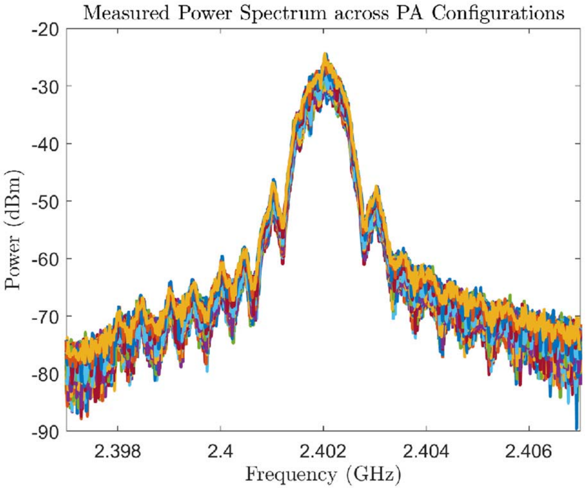


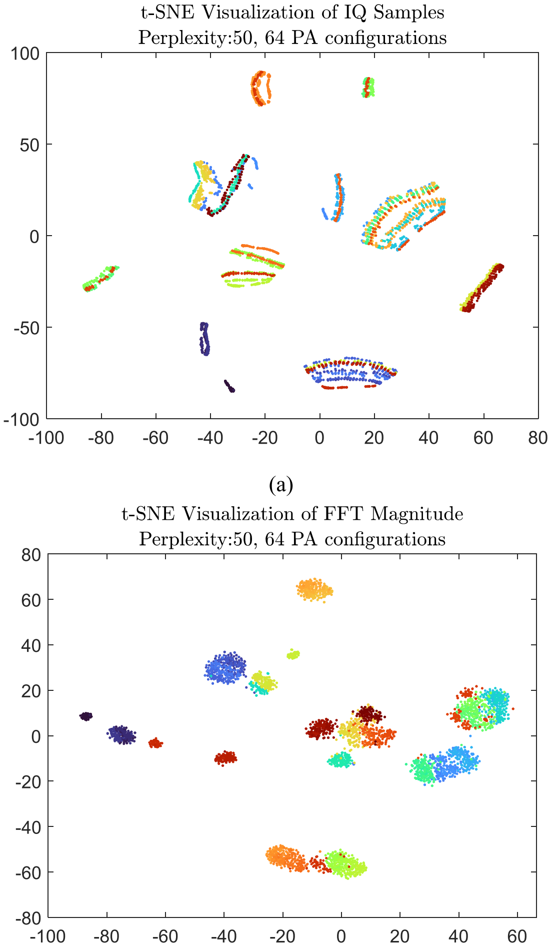
Fig. 16. Spectrum analyzer measurements were taken of the PA output signal while transmitting PHY1M BLE advertising packets at the 2.402 GHz channel.

To avoid needing to process the entirety of the dataset in this way, the CNN model was first used to process the data and generate a confusion matrix, from which the most distinguishable PA configurations could be determined. Then, 60 BLE preambles corresponding to each chosen PA configuration were randomly taken from the dataset for inspection. t-SNE was used to cluster these preambles across the 64 most distinguishable PA configurations as determined from the confusion matrix produced by the CNN model.

Naively concatenating the real and imaginary components of each preamble's IQ samples and performing t-SNE with a perplexity of 50 to prioritize the visualization of the dataset's global structure resulted in the plot shown in Fig. 17(a). Although visible clusters corresponding to different PA configurations are present, several groups of points are located close to one another in large "global" clusters, indicating that data collected from several PA configurations are difficult to distinguish from one another while using IQ samples to represent the data. Using the magnitude of the FFT of each data vector as input to the t-SNE routine and using Chebyshev distance as a metric yielded significantly better clustering results as shown in Fig. 17(b), with points in the low-dimensional representation further clumping together into dense groups. Although this result implies that IQ samples as taken in by the CNN model are not an optimal representation of the data, computing alternative data representations such as instantaneous frequency and short-time Fourier transforms [19] introduces additional overhead in the RX-side of the RF fingerprinting system.

### *C. Signal Classification*

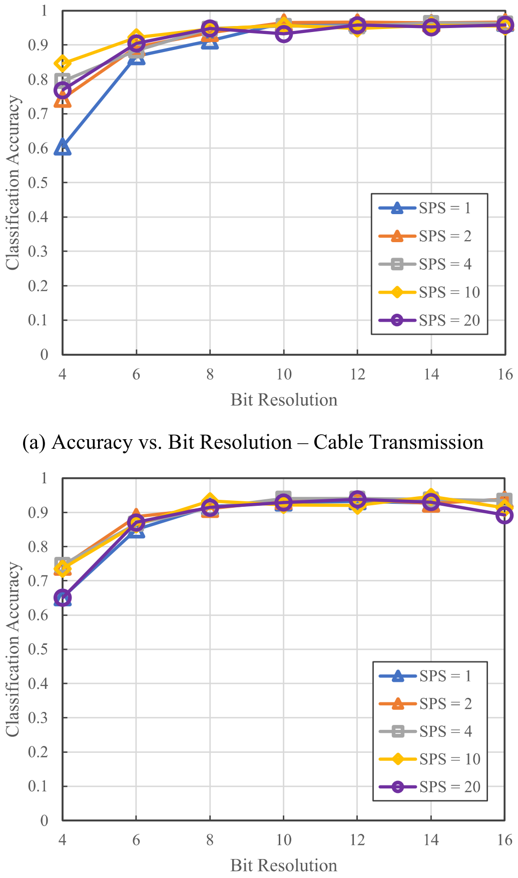
For the purpose of examining the robustness of the CNN to different signal processing environments, the model is trained



(a)

(b)

Fig. 17. t-SNE was used to visualize a subset of the dataset that was collected by transmitting through a cable in order to verify the presence of distinguishable RFFs associated with different PA configurations. In (a), the real and imaginary components of the IQ samples were directly fed into the t-SNE process, while in (b), the magnitude of the FFT of each BLE preamble was fed into the t-SNE. Each color corresponds to a PA configuration.



(a) Accuracy vs. Bit Resolution – Cable Transmission

(b) Accuracy vs. Bit Resolution - OTA

Fig. 18. Bit resolution and sampling rate were varied for both cable (a) and over the air (OTA) (b) data to determine their impact on CNN accuracy.

and tested across signals with different sample-per-symbol (SPS) rates and bit resolutions to understand the receiver ADC's speed (over-sampling) and quantization requirements. The collected dataset is quantized with 4, 6, 8, 10, 12, 14, and 16 bits and decimated to sampling rates of 1, 2, 4, 10, and 20 samples per symbol for a total of 35 different configurations to be trained and tested by the CNN model. A low-pass filter is applied prior to decimation to remove high-frequency components that could be aliased.

The classification accuracy results are shown in Fig. 18. Training with the cable transmission data was used to simulate a low noise condition, while OTA data was used to simulate environments with additional noise. Looking at the cable transmission results, the impact of bit-resolution becomes less apparent for bit-resolutions higher than 10 bits, beyond which the bit-resolution negligibly affected classification performance for the CNN model. Accuracy is not significantly impacted by sample rate down to twice the symbol rate for bit resolutions above 10-bits as well, below which sample rate exhibits an increasing impact on classifier accuracy. In Fig. 18(b), the classifier's performance degrades for SPS $=$ 20. We observed that this is because the lightweight CNN model overfits the training dataset when the sampling rate is high and the input sequence is too long, which could be addressed by using larger filters in the convolution layers that are more suitable for longer input sequences. Overall, the CNN could learn features more easily from the signals with more sampling points, but it comes with the trade-off of a linear increase in computational cost in both classification and signal processing. We chose the model with an oversampling rate of 4 samples per symbol (complex signal length $=$ 160) relative to the BLE symbol rate and 10-bit resolution as a reference, which achieves $>$94% accuracy on 220-class classification with both OTA and cable transmission to carry out further analysis. This sample rate is typical of recent RF fingerprinting works reported in CAS conferences, which range from 8 samples per symbol in the case of a work using BPSK modulation to 16 samples per symbol in a work covering Zigbee devices [20]–[22].

It is expected that among the 220 possible RFFs generated from the PA configurations, some RFFs are more prominent than others, therefore, showing better classification results. Following the previous test, the most prominent RFFs were identified using a validation dataset of both OTA and cable transmission data. We generated a confusion matrix, and by

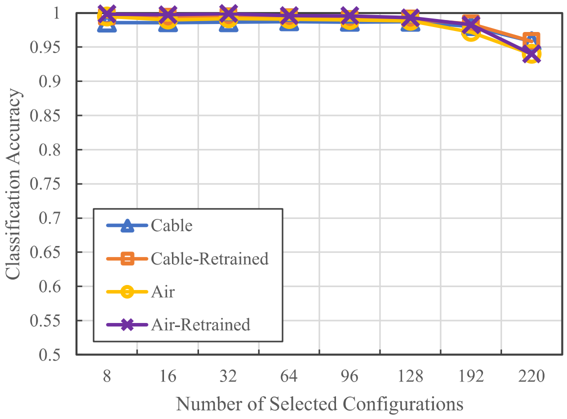


Fig. 19. CNN accuracy was determined for different numbers of selected PA configurations. The CNN performance after the transfer learning process (retraining) is included.

TABLE II

TRANSFER LEARNING SPECIFICATION

| Number of Fingerprints | Batch Size | Epoch |
|---|---|---|
| **8** | 64 | 500 |
| **16** | 64 | 500 |
| **32** | 64 | 500 |
| **64** | 128 | 800 |
| **96** | 128 | 800 |
| **128** | 128 | 800 |
| **192** | 256 | 1000 |

selecting the configurations with the highest sensitivity (true positive rate), sets of the N = 8, 16, 32, 64, 96, 128, and 192 most distinguishable PA configurations were composed for each transmission model, and the configurations with high false-positive rates are excluded in the dataset for this new experiment. The CNN was then tested on the dataset only with the best N configurations to show the classification accuracy on the most prominent RFFs. Furthermore, we performed a transfer learning procedure (re-training) for each configuration set to boost the model's classification accuracy on the selected group of RFFs. This retraining process still only utilizes the original training dataset that was being used to train the model. The difference is that the signals generated from the least-distinguishable combinations are excluded from the training dataset at this time. The results shown in Fig. 19 depict the promising performance of this method with and without the re-training process. After re-training, the CNN model successfully classifies 192 RFFs on the OTA signal with >98% accuracy and 128 RFFs on the OTA signal with >99% accuracy. The specification for the re-training process is shown in Table II. The batch size and number of epochs used for the retraining process are adjusted according to the desired number of fingerprints, which takes the size of the available dataset with different numbers of selected PA configurations (730 × N) into account.

A 2$^{\text{nd}}$ chip sample was used to investigate the impact of noise on system performance. The transmit power was adjusted using a set of attenuators to set the signal-to-noise ratio of the over-the-air measurement environment from 15 dB to 35 dB. The lower end of this SNR range was chosen in consideration of the minimum SNR specification for achieving the 0.1% bit error rate (BER) standard for BLE reported in the literature pertaining to BLE demodulators, which ranges from 14.4 to 15 dB for practical demodulators [23]–[24] and is 12 dB for an optimal demodulator [25]. As shown in Fig. 20, the accuracy for 220 classes falls from 98% at 35 dB SNR to 87% at 15 dB SNR and remains above 91% for the 192 most prominent RFF configurations. Although the classification accuracy is intuitively sensitive to noise level, it is sufficiently high at lower SNR to indicate the resilience of this approach for introducing RFF variability to channel noise.

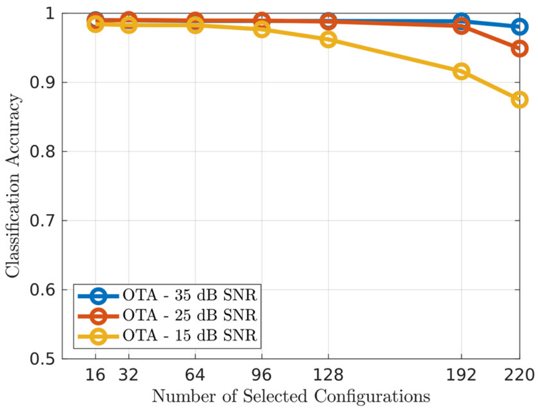


Fig. 20. CNN accuracy was determined for different numbers of selected PA configurations using a second chip sample across different signal-to-noise ratios.

## IV. FPGA IMPLEMENTATION

FPGAs have shown growing interests in deploying the deep learning inference engine [26], and more modern AI-optimized FPGAs are being manufactured to specifically support machine learning applications [27]. FPGAs have many advantages over general-purpose computing units like CPUs and GPUs for their task-specific parallel processing and energy efficiency. Compared to ASIC designs, FPGAs provide low engineering cost and hardware reconfigurability, which promise a long shelf life. In this work, to support and accelerate the device identification on the edge, we implemented an FPGA inference module for CNN with the architecture proposed in section II.D, which is visualized in Fig. 13. The CNN inference module is designed with Vivado High-Level-Synthesis (HLS) as an IP Block and deployed on Xilinx ZCU102.

There are four types of layers presented in the CNN during the inference: 1-D Conv layer, a fully-connected dense layer, max-pooling layer, and ReLU layer. In this work, these layers are implemented with an optimized memory access pattern and are pipelined to achieve high throughput.

Fig. 21 shows the data dimensions in the 1D Conv layer and how the operations are paralleled to accelerate the convolution. The input to the Conv layer has a length of $L_{in}$ with $C_{in}$ input channels. Each kernel set has a size of $K$ along with $C_{in}$ channels to convolve over input data's channels. The total number of kernel sets is equal to the number of output channels

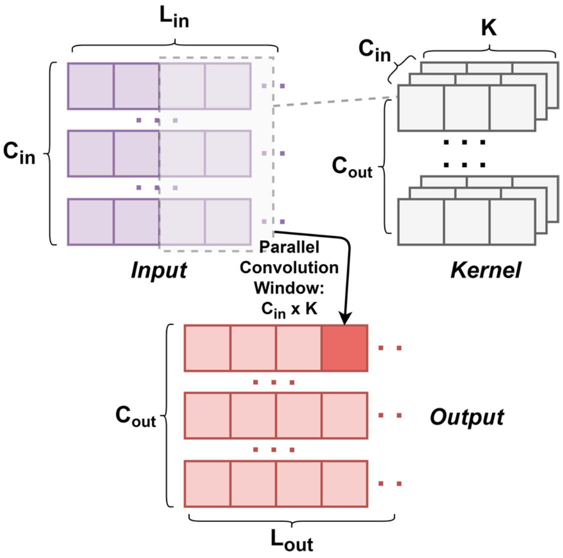


Fig. 21. Visualization of the 1D convolution operation and its dimensionalities. The Parallel Convolution Window has a size of Input Channel ($C_{in}$) × Kernel Size ($K$) to execute ($C_{in} \times K$) MAC operations in parallel to produce a single output. The total trip count for convolving the whole input is Output Channel ($C_{out}$) × Output Length ($L_{out}$) ÷ Stride ($S$), performing $\frac{C_{in}\times K\times C_{out}\times L_{out}}{S}$ operations.

in this layer, which is $C_{out}$. If the input is applied with the same-padding and the stride is $S$, the output length $L_{out}$ is equal to the input length $L_{in}$ divided by stride $S$, and $S = 2$ is how we implemented our convolutional layers. The total number of multiply-accumulate (MAC) operations in this layer is $\frac{L_{in}\times C_{in}\times K\times C_{out}}{S}$. To accelerate the convolutional operations, we use a Parallel Convolution Window to compute some of the MAC operations in parallel to reduce the latency of producing the output. The Parallel Convolution Window has a size of $C_{in} \times K$, which is the size of a single kernel set. With this window, $C_{out}$ kernel sets roll over the input data $L_{out} = \frac{L_{in}}{S}$ times to produce all output, which brings down the total trip count to $\frac{L_{in}\times C_{out}}{S}$. Combining this convolution acceleration method with the CNN model introduced in section III, each convolution layer has a trip count of 640 in this design.

Fully-connected (FC) layer is essentially a matrix multiplication between a 1-dimensional array and a 2-dimensional array. The number of MAC operations in this layer is equal to the input size $L_{in}\times$ output size $L_{out}$, which is also the size of the weights in the layer. The throughput of the FC layer can be improved by running multiple MAC operations in parallel. Fig. 22 shows two examples of matrix multiplication in the FC layer with different parallel factors. At each clock cycle, one input is loaded to be multiplied with the weights and accumulated into the output buffer. The operation on top of Fig.22 has a parallel factor of 4, and the operation on the bottom has a parallel factor of 2. They differ in the number of processing units (PE) for the MAC operation and the memory access pattern. With processing units running in parallel, the final trip count in the FC layer would become $\frac{L_{in}\times L_{out}}{ParallelFactor}$. The parallel factors for FC1 layer and FC2 layer are 128 and 6 respectively so that their trips counts do not exceed the trip count of the Conv layers to achieve efficient pipelining of the layers.

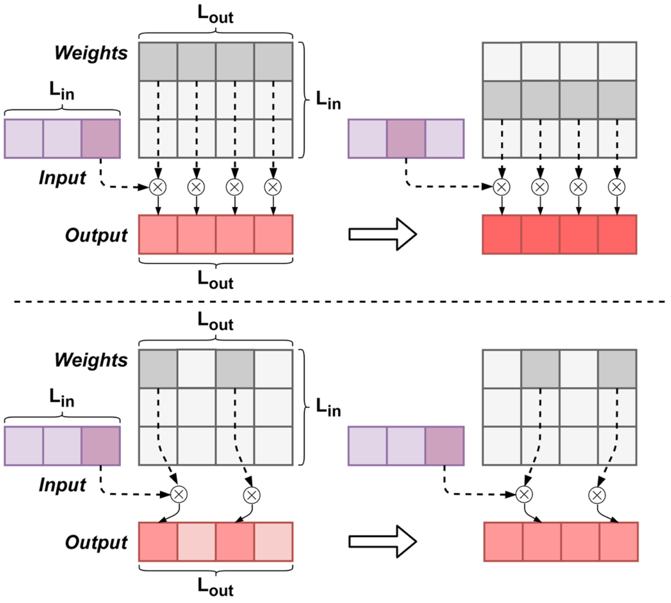


Fig. 22. Top: A dense layer matrix multiplication is shown where the input is multiplied with the weights in parallel and the results are accumulated into the output to achieve high throughput. Bottom: The same dense layer matrix multiplication is shown but with 2x less parallelism, 2x reduced PE usage and less array partitions.

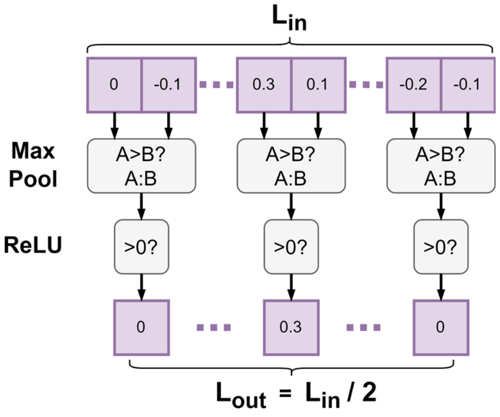


Fig. 23. The implementation of the max pooling layer and ReLU layer is shown. The max pooling layer has kernel size of 2 with stride of 2 to down sample the input, and ReLU simply clears negative input to zero.

Fig. 23 shows the operations for the max-pooling layer and the ReLU layer. The max-pooling layer in this model has a kernel size of 2 and stride of 2 to downsample the input size by a factor of 2. The ReLU layer's output array is initialized with zeros and any input larger than 0 will be passed into the output array.

The final complete CNN module has the same structure as shown in Fig. 13 with each layer developed in the methods discussed in this section pipelined. It achieves a throughput of 151,975 classifications per second, latency of 0.043ms with a dynamic power consumption of 0.46W running at 100MHz, and a max clock frequency of 143MHz. The FPGA implementation's performance and hardware resource utilization are collected with Vivado implementation report and summarized in Table III. We also compared the FPGA design with other works targeting RF transmitter classification tasks that are

TABLE III
FPGA IMPLEMENTATION PERFORMANCE AND COMPARISON WITH OTHER WORKS

| Resource Type | This work | OJCAS21 [28] | MILCOM19 [29] |
|---|---|---|---|
| Task | **220-class RFF classification** | **6-class RFF classification** | **6-class Modulation classification** |
| Classifier | Convolutional Neural Network | Bayesian Neural Network | Fully Connected Neural Network |
| Platform | Xilinx ZCU102 (XCZU9EG) | Xilinx ZCU102 | Xilinx XCZU9EG |
| LUT | 26,013 (9%) | 9,094 (3%) | 158,435 (58%) |
| BRAM Block | 72.5 (8%) | 49 (5%) | Unknown |
| FF | 25,347 (5%) | 7,176 (1%) | 16,222 (3%) |
| DSP | 215 (9%) | 89 (4%) | 210 (8%) |
| Data Type | 16-bit fixed-point | 16-bit fixed-point | 16-bit fixed-point |
| Dynamic Power | **0.46 W** | **0.19 W** | **0.50 W** |
| Throughput (classifications/ second) | 151,975 | 347,222 | 41,666 |

mostly related to our work. For multi-class classification tasks, more classes to be classified would require stronger and more complicated classifiers, therefore the hardware usage and power consumption would also scale up. The CNN FPGA module in this work provides a high throughput that may be far beyond the throughput requirement in the real-RFF identification environment, and it is also possible to have trade-offs between the clocking rate and power consumption to enable deploying the system on lower-power devices.

## V. CONCLUSION

In this work, an RF fingerprinting system with a configurable PA on the transmit-side is proposed for the purpose of enhancing user capacity. A Class-E PA with 220 configurations total is designed and used to imprint configurable RFFs on transmitted signals. We presented the measurement results with an over-the-air test to demonstrate that the RF-fingerprints generated by the combinatorial PA can be effectively classified by CNN with accuracy $>93\%$ for 220-RFF classification task and $>98\%$ for 128-RFF classification task. The CNN is developed on FPGA to achieve an inference throughput of more than 150,000 classifications per second.

## REFERENCES

[1] W. Iqbal, H. Abbas, M. Daneshmand, B. Rauf, and Y. A. Bangash, "An in-depth analysis of IoT security requirements, challenges, and their countermeasures via software-defined security," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 10250–10276, Oct. 2020, doi: 10.1109/JIOT.2020.2997651.

[2] N. Neshenko, E. Bou-Harb, J. Crichigno, G. Kaddoum, and N. Ghani, "Demystifying IoT security: An exhaustive survey on IoT vulnerabilities and a first empirical look on internet-scale IoT exploitations," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2702–2733, 3rd Quart., 2019, doi: 10.1109/COMST.2019.2910750.

[3] N. Zhang *et al.*, "Physical-layer authentication for Internet of Things via WFRFT-based Gaussian tag embedding," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 9001–9010, Sep. 2020, doi: 10.1109/JIOT.2020.3001597.

[4] K. Sa, D. Lang, C. Wang, and Y. Bai, "Specific emitter identification techniques for the Internet of Things," in *IEEE Access*, vol. 8, pp. 1644–1652, 2020, doi: 10.1109/ACCESS.2019.2962626.

[5] M. Köse, S. Taşcioğlu, and Z. Telatar, "RF fingerprinting of IoT devices based on transient energy spectrum," *IEEE Access*, vol. 7, pp. 18715–18726, 2019.

[6] S. S. Hanna and D. Cabric, "Deep learning based transmitter identification using power amplifier nonlinearity," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, 2019, pp. 674–680, doi: 10.1109/ICCNC.2019.8685569.

[7] S. Rajendran, Z. Sun, F. Lin, and K. Ren, "Injecting reliable radio frequency fingerprints using metasurface for the Internet of Things," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 1896–1911, 2021, doi: 10.1109/TIFS.2020.3045318.

[8] S. Mohanti, N. Soltani, K. Sankhe, D. Jaisinghani, M. Di Felice, and K. Chowdhury, "AirID: Injecting a custom RF fingerprint for enhanced UAV identification using deep learning," *Proc. IEEE Global Commun. Conf.*, Dec. 2020, pp. 1–6, doi: 10.1109/GLOBECOM42002.2020.9322561.

[9] M. J. M. Pelgrom, A. C. J. Duinmaijer, and A. P. G. Welbers, "Matching properties of MOS transistors," *IEEE J. Solid-State Circuits*, vol. 24, no. 5, pp. 1433–1439, Oct. 1989, doi: 10.1109/JSSC.1989.572629.

[10] M. Aziz, M. Vejdani Amiri, M. Helaoui, and F. M. Ghannouchi, "Statistics-based approach for blind post-compensation of modulator's imperfections and power amplifier nonlinearity," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 3, pp. 1063–1075, Mar. 2019, doi: 10.1109/TCSI.2018.2877940.

[11] R. Natarajan, P. Zand, and M. Nabi, "Analysis of coexistence between IEEE 802.15.4, BLE and IEEE 802.11 in the 2.4 GHz ISM band," in *Proc. 42nd Annu. Conf. Ind. Electron. Soc.*, Jun. 2016, pp. 6025–6032, doi: 10.1109/IECON.2016.7793984.

[12] K. C. Tsai, "CMOS power amplifiers for wireless communications," Ph.D. Dissertation, Dept. EECS, Univ. California, Berkeley, CA, USA, 2007.

[13] A. Zhu, J. C. Pedro, and T. R. Cunha, "Pruning the Volterra series for behavioral modeling of power amplifiers using physical knowledge," *IEEE Trans. Microw. Theory Techn.*, vol. 55, no. 5, pp. 813–821, May 2007, doi: 10.1109/TMTT.2007.895155.

[14] W. E. Cobb, E. W. Garcia, M. A. Temple, R. O. Baldwin, and Y. C. Kim, "Physical layer identification of embedded devices using RF-DNA fingerprinting," in *Proc. Mil. Commun. Conf.*, 2010, pp. 2168–2173, doi: 10.1109/MILCOM.2010.5680487.

[15] S. Pazos, F. Aguirre, F. Palumbo, and F. Silveira, "Reliability-aware design space exploration for fully integrated RF CMOS PA," *IEEE Trans. Device Mater. Rel.*, vol. 20, no. 1, pp. 33–41, Mar. 2020, doi: 10.1109/TDMR.2019.2957489.

[16] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 168–179, Feb. 2018, doi: 10.1109/JSTSP.2018.2797022.

[17] K. Y. Wong and F.-L. Chung, "Visualizing time series data with temporal matching based t-SNE," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2019, pp. 1–8, doi: 10.1109/IJCNN.2019.8851847.

[18] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[19] J. Wang, Z. Mo, H. Zhang, and Q. Miao, "A deep learning method for bearing fault diagnosis based on time-frequency image," *IEEE Access*, vol. 7, pp. 42373–42383, 2019, doi: 10.1109/ACCESS.2019.2907131.

[20] B. Li and E. Cetin, "Waveform domain deep learning approach for RF fingerprinting," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, 2021, pp. 1–5, doi: 10.1109/ISCAS51556.2021.9401486.

[21] T. Morehouse and R. Zhou, "RF device identification using CNN based PUF," in *Proc. IEEE 63rd Int. Midwest Symp. Circuits Syst. (MWSCAS)*, 2020, pp. 217–220, doi: 10.1109/MWSCAS48704.2020.9184695.

[22] H. Tamura, K. Yanagisawa, A. Shirane, and K. Okada, "Wireless devices identification with light-weight convolutional neural network operating on quadrant IQ transition image," in *Proc. 18th Int. New Circuits Syst. Conf. (NEWCAS)*, 2020, pp. 106–109, doi: 10.1109/NEWCAS49341.2020.9159777.

[23] M. S. Pereira, J. C. Vaz, C. A. Leme, J. T. de Sousa, and J. C. Freire, "An ultra-low power low-IF GFSK demodulator for Bluetooth-LE applications," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2015, pp. 1226–1229, doi: 10.1109/ISCAS.2015.7168861.

[24] M. S. Pereira, J. C. Vaz, C. A. Leme, J. T. de Sousa, and J. C. Freire, "A 170 all-digital GFSK demodulator with rejection of low SNR packets for bluetooth-LE," *IEEE Microw. Wireless Compon. Lett.*, vol. 26, no. 6, pp. 452–454, Jun. 2016, doi: 10.1109/LMWC.2016.2562639.

[25] A. Pipino, A. Liscidini, K. Wan, and A. Baschirotto, "Bluetooth low energy receiver system design," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Jun. 2015, pp. 465–468, doi: 10.1109/ISCAS.2015.7168671.

[26] J. Fowers *et al.*, "A configurable cloud-scale DNN processor for real-time AI," in *Proc. ACM/IEEE 45th Annu. Int. Symp. Comput. Archit. (ISCA)*, Los Angeles, CA, USA, Oct. 2018, pp. 1–14.

[27] A. Boutros *et al.*, "Beyond peak performance: Comparing the real performance of AI-optimized FPGAs and GPUs," in *Proc. Int. Conf. Field-Program. Technol. (ICFPT)*, Dec. 2020, pp. 1–10.

[28] J. Xu, Y. Shen, E. Chen, and V. Chen, "Bayesian neural networks for identification and classification of radio frequency transmitters using power amplifiers' nonlinearity signatures," *IEEE Open J. Circuits Syst.*, vol. 2, pp. 457–471, 2021, doi: 10.1109/OJCAS.2021.3089499.

[29] S. Soltani, Y. E. Sagduyu, R. Hasan, K. Davaslioglu, H. Deng, and T. Erpek, "Real-time and embedded deep learning on FPGA for RF signal classification," in *Proc. Mil. Commun. Conf. (MILCOM)*, 2019, pp. 1–6, doi: 10.1109/MILCOM47813.2019.9021098.

**Yuyi Shen** (Graduate Student Member, IEEE) received the B.S. degree in electrical and computer engineering from Carnegie Mellon University in 2020, where she is currently pursuing the Ph.D. degree. She held an internship position at Apple Inc., in 2020, and she is primarily interested in RFIC design with a focus on the application of RF circuits to security and device identification.

**Jiachen Xu** (Graduate Student Member, IEEE) received the B.S. degree in computer engineering from Purdue University in 2020. He is currently pursuing the Ph.D. degree with Carnegie Mellon University. His interests lie in brain-inspired machine-learning algorithms and embedded system design for wireless applications.

**Jinho Yi** (Graduate Student Member, IEEE) received the B.S. degree in computer engineering from Purdue University in 2021. He is currently pursuing the Ph.D. degree with Carnegie Mellon University, with an interest in the applications of machine learning algorithms to low power systems.

**Ethan Chen** is currently a Research Scientist with the Energy-Efficient Circuits and Systems Laboratory, Carnegie Mellon University. His research interests include neuromorphic computing, hardware security, and biomedical interfaces.

**Vanessa Chen** (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, USA, in 2013.

She was with Qualcomm, San Diego, CA, USA, working on energy-efficient data-acquisition systems for mobile devices. From 2010 to 2013, she was with Carnegie Mellon University, where she focused her research on self-healing systems and high-speed ADCs, and held a research internship position at the IBM Thomas J. Watson Research Center, Yorktown Heights, NY, USA, in 2012. She was an Assistant Professor with The Ohio State University, Columbus, OH, USA. She is currently an Assistant Professor of electrical and computer engineering with Carnegie Mellon University. Her research interests focus on data conversion interfaces for machine learning, RF/analog hardware security, ubiquitous sensing, and communication systems.

Dr. Chen was a recipient of the NSF CAREER Award in 2019, the Analog Devices Outstanding Student Designer Award in 2013, and the IBM Ph.D. Fellowship in 2012. She is also an Associate Editor of the IEEE OPEN JOURNAL OF CIRCUITS AND SYSTEMS and a Guest Editor of the *ACM Journal on Emerging Technologies in Computing Systems*. She is a Technical Program Committee Member of the IEEE Symposium on VLSI Circuits, the IEEE Custom Integrated Circuits Conference, the IEEE Asian Solid-State Circuits Conference, and the IEEE/ACM Design Automation Conference.