Journal of Signal Processing Systems
https://doi.org/10.1007/s11265-021-01722-7

Energy Efficient Deep Learning Inference Embedded on FPGA for Sleep Apnea Detection

Omiya Hassan[1] · Tanmoy Paul[1] · Maruf Hossain Shuvo[1] · Dilruba Parvin[1] · Rushil Thakker[1] · Mengrui Chen[1] · Abu Saleh Mohammad Mosa[2] · Syed Kamrul Islam[1]

Received: 12 March 2021 / Revised: 1 October 2021 / Accepted: 15 November 2021

Abstract

Sleep apnea is a type of disorder caused by the absence of breathing for a specific period of time coupled with a significant decrease in the blood oxygen saturation level. The monitoring process of sleep apnea is challenging due to the requirement of overnight expensive sleep study, hand-crafted feature extraction from breathing signals, and manual annotations by the sleep experts. Therefore, a low-cost, energy-efficient, portable, and automated biomedical system is necessary to improve early detection, frequent monitoring, and clinical decision-making. In this paper, a digital hardware design of a trained deep feedforward neural network (FNN) is implemented on a Field Programmable Gate-Array (FPGA) for the detection of sleep apnea. The model was trained and evaluated with hyperparameters obtained from a three-step optimization process which ensures compact design solution in low-power miniaturized CMOS circuits. A three-layer FNN trained with ADAM optimizer and mean square error (MSE) loss minimization shows an accuracy of around 88%. An application-specific deep learning inference module realized in FPGA hardware platform confirms a power consumption below 34 W which is 5 × lower than that of commercially available machine learning accelerators. The outcome of this research can be integrated into a system-on-a-chip (SoC) platform for developing a smart automated sleep apnea detection device.

Keyword Deep learning · Feedforward neural network · FPGA · Sleep apnea · ECG · Oxygen saturation · System-on-a-chip

1 Introduction

The prevalence of sleep-related breathing disorders such as sleep apnea, chronic obstructive pulmonary diseases (COPD), and asthma has recently gained significant attention due to their consequent adverse effects on health [1]. Sleep apnea (SA) syndrome is the recurrent occurrences of partial or complete cessation of respiration during sleep. More than five episodes of SA per sleep hour are usually regarded as pathological conditions. There are two main types of sleep apnea: obstructive sleep apnea (OSA) and central sleep apnea (CSA). In obstructive sleep apnea, the upper airway narrows down or closes due to muscle relaxation during sleep resulting in reduced airflow (hypopnea) or absence of airflow (apnea). The severity of the SA disorder is commonly assessed by the number of apnea and hypopnea events per hour of sleep called apnea–hypopnea index (AHI). In central sleep apnea, the brain fails to transmit stimulus to the breathing muscles due to neurological disorders that cause lack of breathing for a short period of time. The duration of minimal apnea events in adults is 10 s, and the typical event duration varies between 30 and 60 s [2]. The lack of sufficient air in the cardiovascular system results in the desaturation of oxygen in the blood that can contribute to high blood pressure, stroke, heart failure, irregular heartbeats, diabetes, depression, and headaches. Therefore, sleep apnea detection devices are necessary for proper diagnosis, monitoring, and clinical decision making to minimize these risks. Cardiorespiratory polysomnography (PSG) is considered to be the gold standard for diagnosis and monitoring of sleep apnea.

There are some disadvantages of sleep monitoring using PSG methods such as discomfort experienced by the patients

✉ Omiya Hassan
omiya.hassan@mail.missouri.edu; ohbk4@mail.missouri.edu

[1] Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, Missouri 65201, USA

[2] Department of Health Management and Informatics, University of Missouri, Columbia, Missouri 65211, USA

due to the attachment of wires and patches to the body and stressful overnight stay in the sleep laboratory. Other limitations include manual intervention and scoring requirements, high expense, slow speed of the process, and lack of accessibility to a large group of people. Therefore, cost-effective, portable, comfortable, miniaturized, and smart devices for diagnosis and follow-up monitoring of SA are imperative [3]. Portable sleep apnea testing devices have become increasingly accurate with better sensitivity and specificity [4, 5]. Various methods available for detection of apnea presented in recent literature include a pyroelectric sensor, chest belt with a piezoelectric sensor, single-lead electrocardiogram (ECG), blood oxygen saturation (SpO_2) level, microphone for tracheal sound monitoring, and snoring signals [5–8]. These devices are playing the first-line diagnostic methods for establishing the point-of-care diagnosis of sleep-disordered breathing.

In [4], electronic circuitry for sleep apnea detection focusing on the respiratory monitoring of neonatal infants has been presented. The device is based on a pyroelectric transducer that converts the heat generated due to breathing into an equivalent electrical charge. If no breathing signal arrives within 10 s, the device considers this as an apneic event and wirelessly transmits an impulse to a central coordinator which alerts the caregivers. Significant efforts have been made on the development of non-invasive, low-power respiration monitoring devices with wireless telemetry capability both for adults and neonatal infants [6, 8]. An apnea detection and monitoring system to restart breathing function of patients has been presented in [9] with a MEMS-based 3-axis accelerometer and a wristband. The acceleration sensor placed on the chest continuously monitors the movements of the diaphragm to detect apneic events during sleep. If an apneic event occurs, a closed-loop control system sends a signal to the wristband to trigger a vibration motor to disturb the patient until breathing is resumed. One underlying problem with these devices is their subject dependence. Different subjects have different breathing patterns and therefore there is a need for calibration to ensure accurate detection of apneic event for proper monitoring of breathing function in each patient.

With recent improvements in algorithms and automatic feature learning capability, deep learning is now actively being used for sleep apnea detection employing various physiological signals. An OSA detection algorithm with good noise immunity using a deep convolutional neural network (CNN) with single-lead ECG is presented in [10]. A comparative study has been presented in [11] to identify suitable machine learning (ML) algorithm and appropriate physiological signals for detection of sleep apnea in terms of a trade-off between computational resource and performance. Based on a comparison of deep learning (RNN and CNN) with conventional machine learning algorithms (Random Forests, Decision Tree, and Multi-Layer Perceptron), researchers have identified deep learning as the best performing algorithm and oxygen saturation level (SpO_2) as the most important physiological parameter for SA detection [11]. A similar conclusion has been drawn in [12] that a deep learning model outperforms classical machine learning methods for detection of CSA for data obtained through pressure-sensitive mat (PSM). In most of the existing deep learning-based solutions, the data acquired from complex PSG experiments is most often computation intensive. A SpO_2 based apnea monitoring device has been presented in [13] with separate sensing and processing units with a wireless interface. This device requires pre-processing, complex hand-crafted feature extraction, and a classical threshold-based classification typically performed in a microcomputer. Despite significant improvement in recent years, these devices still suffer from computational needs, power budget, extra circuitry requirements for wireless transmission, and a separate processing unit. Therefore, a machine learning-based portable hardware model in real-time is necessary for point-of-care diagnosis of SA events.

In this paper, we propose a data-driven machine learning-based smart electronic device trained with a large amount of data from subjects of different ages, gender, body pattern, and different physiological conditions. Machine learning-based digital hardware designs have already demonstrated high accuracy [14, 15] with low-power consumption and miniaturized size. These smart apnea detection devices offer important advantages over traditional instruments such as on-chip decision-making capability, diagnosis based on multiple parameters, and less or no calibration due to training from a large volume of data. Studies demonstrated that, among all the physiological signals, pulse oxygen saturation level (SpO_2) [16] and ECG [17] are the most relevant signals for sleep apnea detection. The proposed system presented in this paper takes SpO_2 and ECG signals as the input and provides the presence or absence of apnea as binary output. Most of the current medical devices for apnea detection are connectivity dependent due to the separate data acquisition and processing units, and therefore lack privacy and security. In future, the proposed deep learning inference module can be incorporated in an all-in-one embedded hardware platform thus ensuring low latency, low energy consumption, memory efficiency, and high processing speed.

The rest of the paper is organized as follows: a brief overview of the proposed system is presented in Sect. 2 followed by a discussion of the AI-enabled software-hardware co-simulation method in Sect. 3. Section 4 describes the model building, tuning, and optimization in software and Sect. 5 describes the design and implementation of the proposed model in hardware platform. Section 7.1 presents the simulated and the test results while the concluding remarks are summarized in Sect. 8.

2 System Overview

The proposed apnea detection system takes in two types of input: biopotential ECG data from chest strap and oxygen saturation (SpO_2) signal from pulse oximeters. The chest movement ECG data comes from a polyvinylidene fluoride (PVDF) based piezoelectric transducer attached to a medical chest strap and oxygen saturation data SpO_2 comes from a pulse oximeter attached to the patient's finger. Figure 1 illustrates the block diagram of the proposed system where the input to the deep learning inference block includes the chest movement and the SpO_2 data. Both signals are subsequently processed into digital form and then fed as the input to the decision-making block. The pre-trained deep learning inference module processes the input data based on the learned parameters and predicts the apneic occurrences. In future the system will include additional circuits which will receive the binary output of the FNN module (1: *sleep apnea*, 0: *absence of apnea/normal condition*) and initiate an alarm signal. This will facilitate the patients to resume breathing by waking them up or alerting caregivers present in the unit to respond upon detection of an apneic event. Wireless data transfer module will be integrated into the device that will aid in further study and diagnosis. This paper is focused on two major functions of the sleep apnea detection system: collection of data from biosensors for prediction of apneic occurrences with high accuracy and hardware implementation of the highly accurate deep learning model.

3 Software-Hardware Co-Simulation Process

The deep learning (DL) based decision-making signal processing block has been designed employing a software-hardware co-simulation process. This approach was used to translate the proposed DL model into equivalent digital logic blocks [18]. Figure 2 illustrates the software-hardware co-simulation process for application specific efficient machine learning hardware accelerator design.

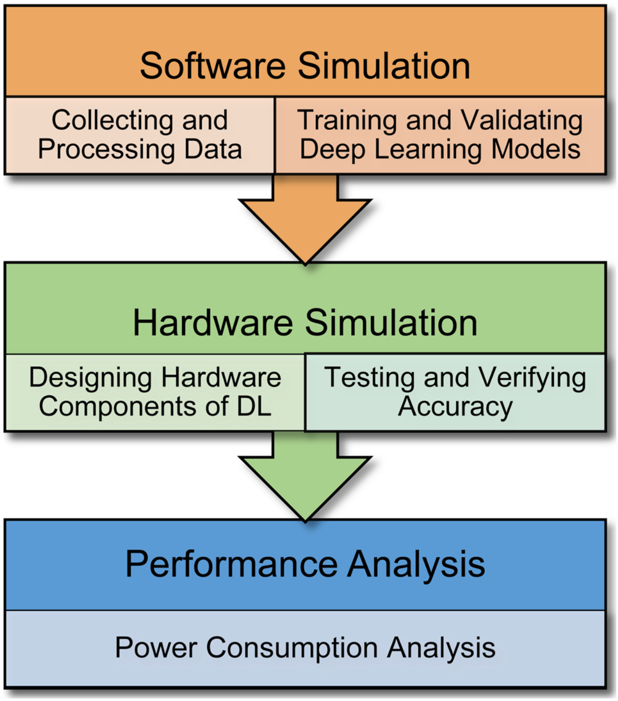
Software Simulation
Collecting and Processing Data
Training and Validating Deep Learning Models
Hardware Simulation
Designing Hardware Components of DL
Testing and Verifying Accuracy
Performance Analysis
Power Consumption Analysis

Figure 2 Flow chart of software-hardware co-simulation process [18].

As shown in Figure 2, the proposed hardware-software co-simulation process involves three steps: software simulation, hardware simulation and performance analysis. In the software simulation step data is being collected and processed for the targeted applications. The processed data is classified into training, validating and testing datasets which are fed into the machine learning model. After achieving optimal accuracy results and evaluating the model with various performance analyses, the learned parameters

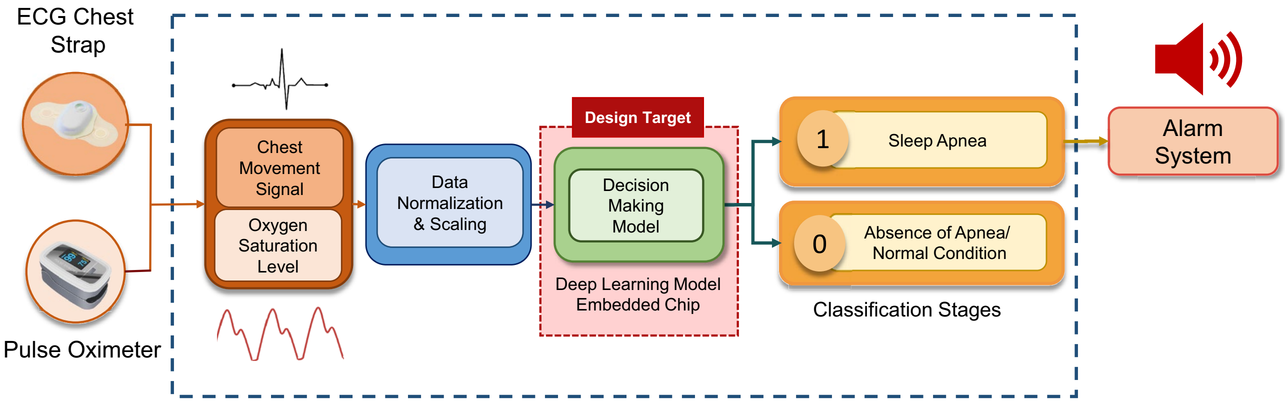

Figure 1 Block diagram of the overall system design with front-end biosensors.

(weights and biases) are extracted from the trained model and used for designing the network architecture for digital hardware implementation. In the hardware simulation step the extracted parameters and network architecture model are designed on reprogrammable FPGAs using ModelSim software which is then tested with the testing dataset for accurate prediction. In the performance analysis step, power consumption analysis is performed to analyze the energy efficiency of the hardware realization using Vivado HLx software.

4 Software Simulation: Deep Learning Model Building

First step in DL based hardware system is to build and tune the model in software to get optimum performance result with real-life clinical data. Therefore, data collection, pre-processing, and labelling are important steps before training the supervised DL model. Following the pre-processing and labelling of the data, the model hyperparameters need to be tuned in the training phase to achieve a compact model architecture that ensures an optimal accuracy in the evaluation phase.

The following sub-sections describe the data collection, processing, design and evaluation methods of the proposed system. Figure 3 illustrates the process of the DL model development with an open-source clinical data collected from sleep studies.

4.1 Data Collection and Signal Processing

4.1.1 Dataset and Pre-processing

In this study, the PhysioNet Apnea-ECG dataset was used [19, 20]. There are a total of 35 records in this dataset that includes both healthy subjects and moderate to severe sleep apnea patients. All subjects were recorded twice for two consecutive nights each time, with an interval of four weeks in-between. The lengths of each record vary between 7 to 10 h. Each of the recordings includes a digitized ECG signal sampled at 100 Hz and normal/apneic conditions annotated by sleep experts. Among the 35, only eight of these recordings have both ECG and blood oxygen saturation signals (SpO_2) where each recordings had nearly 7 to 10 h of data logging. Since research efforts already confirmed that ECG and SpO_2 are highly significant for sleep apnea prediction [16, 17], these eight recordings that include both the type of data were considered. In each recording, annotatin symbol ‘A’ or ‘N’ is placed at the start of every minute depending on whether an apnea event was in progress or not at the start of the 1-min reecording period. Since apnea annotations were provided for a 1-min interval, both the SpO_2 and ECG signals were segmented into 1-min long intervals. In this dataset, apnea annotation depends only on the presence or absence of apneic events at the beginning of the recording period. However, such an annotation scheme may be misleading. For example, an interval may be annotated as ‘N’, although the apneic event was persistent for most of the 1-min duration. Similarly, another interval may be annotated as ‘A’ despite the apneic event lasting only for a short period of time during the 1-min recording period. To overcome this problem, we only considered the initial 30 s of each of the 1-min segments and the rest were discarded. This ensures the removal of apneic events from the normal condition ‘N’ of 1-min recording segments. Similarly scheme was applied, if there were non-apneic sample points in an apnea ‘A’-annotated segment. The decision to consider only 30 s of each interval is justified as it is more than the required minimum duration (10 s) for an apneic event. This type of pre-processing scheme significantly minimizes the chance of false positives.

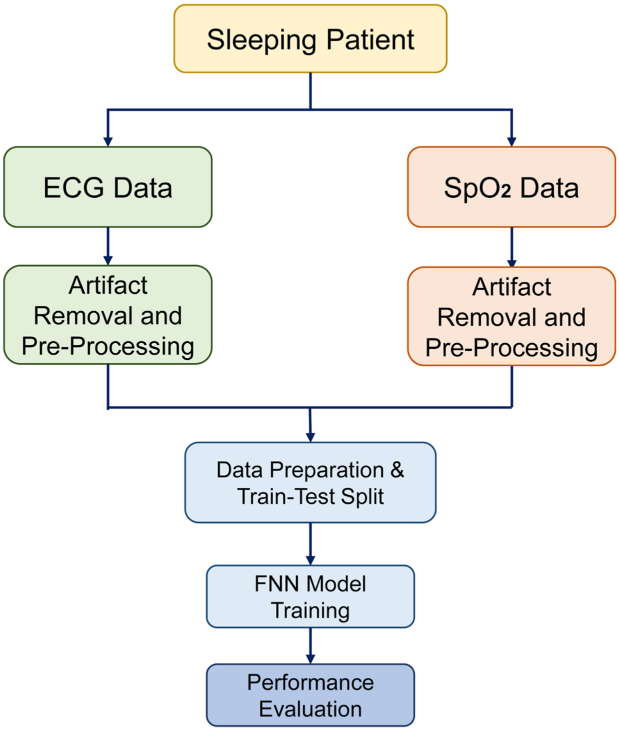
Sleeping Patient
ECG Data
SpO₂ Data
Artifact Removal and Pre-Processing
Artifact Removal and Pre-Processing
Data Preparation & Train-Test Split
FNN Model Training
Performance Evaluation

Figure 3 Flow chart of the software design process side: clinical data processing and FNN model evaluation.

4.1.2 SpO_2 Signal Processing

Any SpO_2 value less than 50% was marked as artifact as it is not physiologically possible. Moreover, all changes of SpO_2 values greater than 4% were marked as artifacts as well. Once those artifacts were removed, a simple moving average filter was used to resample the signal at 1 Hz. After resampling, there were 30 sample points generated from the first 30 s of every 1-min segment. If the first 30 s had more artifacts than the 10% of the total number of sample points before resampling then the entire 1-min recordning segment was discarded. It made sure that the intervals with major information loss, caused by the artifact removal, were not used in training of the DL model.

4.1.3 ECG Signal Processing

From the first 30 s of each 1-min ECG segment, the R-peaks were extracted using the machine-generated QRS annotations. Then the R-R interval was calculated which is simply the time interval between two successive R-peaks. A sliding window technique was implemented to remove the ectopic sample points from the R-R interval series. The window length was 5 and any R-R interval value larger than 20% of the average value within the window was marked as ectopic beats and was removed. Following the completion of the artifact removal process the entire 1-min recording interval was discarded if there are less than 30 sample points from the 30-s interval. If there were more than 30 R-R interval points, only the first 30 of those were considered to be consistent with the number of points in each input vector derived from the SpO_2 signal for training the FNN model. Following the processing of the SpO_2 and ECG signals, a total of 2530 sequences remain, out of which 1512 were normal and 2530 were apneic.

4.2 Feedforward Neural Network (FNN) Design

Processed ECG and SpO_2 data was used to develop an optimized feedforward neural network (FNN) using machine learning libraries such as Tensorflow, Keras for neural network training, NumPy for data reading and Matplotlib plotting library library for data visualization. Data normalization was executed using the MinMaxScaler() function from scikit-learn library which scales and translates each feature individually within the range of 0 and 1. The entire model was trained in Google Collaborative Platform. The parameters (weights and biases) of the best learned model that fitted the data with high accuracy were extracted for hardware implementation purposes. The FNN was chosen because it automatically learns the complex features and patterns of data by accumulating information from its previous layer and sends the updated information to its next layer in a forward propagation manner [21]. The network was trained with two input vectors: ECG signal and SpO_2 signal as mentioned in Sect. 2. For constructing a hardware friendly and easily deployable deep learning model, careful consideration and selection of data normalization, activation function, loss function and optimization techniques were performed over multiple training iterations. The resultant model consists of three hidden layers (8–6-4 units respectively) with two input variables and a one neuron output layer. Figure 4 illustrates the proposed model architecture with the designated activation function of each layer.

The forward propagation of the FNN model [22] for a single unit layer is shown in Figure 5. Here y_i is calculated by

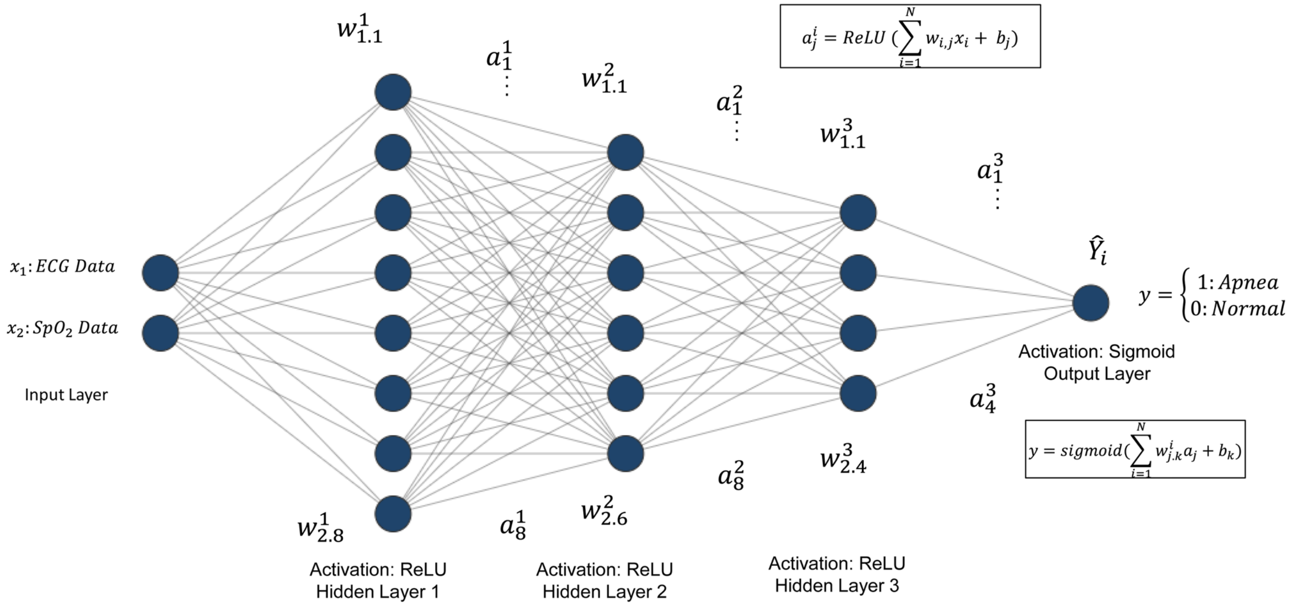

Figure 4 Proposed FNN model with two inputs, 3 hidden-layers (8–6-4 neurons), and one output neuron. Each layer is specified with their respective activation functions.

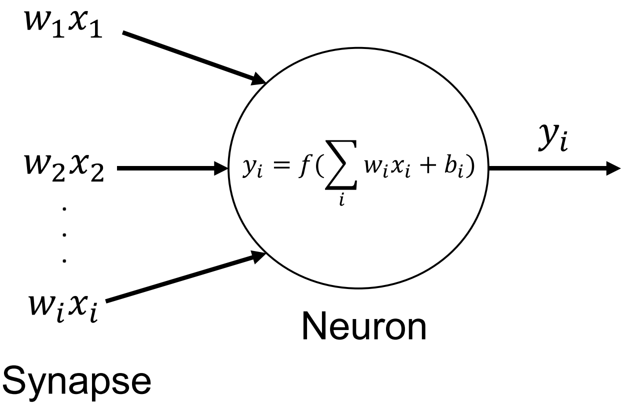

Figure 5 Forward propagation of a single unit FNN layer.

adding bias, b_i to the sum of products (w_i*x_i) and f is termed as the activation function where the resultant gets classified and fed into the next layer as input data. In all three of the hidden layers Rectified Linear Unit (ReLU) [23] was used and at the output layer a Sigmoid activation function was chosen for binary predictions [24].

$$ReLU\ (Z) = \begin{cases} 0, x < 0 \\ x, x \geq 0 \end{cases} \tag{1}$$

$$Sigmoid\ (Z) = \frac{1}{1 + e^{-z}} \tag{2}$$

ReLU is a piece-wise linear function which sends the input as output if the value is positive or zero and a forced zero for the negative input values. The sigmoid function at the output layer successfully categorizes the input values from the previous layer between 0 and 1. In the training phase, mean-squared-error (MSE) as in Eq. 3 was used as the loss function as MSE converges faster compared to other loss functions.

$$MSE\ (J) = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \widehat{Y}_i)^2 \tag{3}$$

where, MSE (J) is the mean squared error, n is the number of data points, Y_i are the observed values, and $\widehat{Y}_i$ are the predicted values. ADAM optimizer as shown in Eq. 4 has been used to calculate the moving squared gradient in order to avoid the exploding or vanishing of gradients of the trained weight [25].

$$w+ = (-learning\_rate.\ m)/((\sqrt{v.1e^{-8}})) \tag{4}$$

Where

$$m = \left(\beta_{1.m}\right) + \left(1 - \beta_1\right).\ dw$$

$$v = \left(\beta_1.v\right) + (1 - \beta_2)(dw)^2,\ \text{and}$$

dw is the gradient vector of the weight.

The standard training parameters (learning_rate: 0.001, β1: 0.9, and β2:0.999) of ADAM optimizer has been used to train and validate the model.

4.3 Performance Evaluation of the FNN

To evaluate the performance of the proposed FNN model, four widely accepted quality metrics were calculated: accuracy, precision, recall/sensitivity, and F-1 score.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

$$F - 1\ Score = 2\ (\frac{Precision\ \times\ Recall}{Precision\ +\ Recall}) \tag{8}$$

The performance of the FNN was also graphically demonstrated using the AUC-ROC (Area Under the Receiver Operating Characteristics) curve which is often used to characterize the performance of a binary classification problem. The ROC is the probability curve, whereas AUC illustrates the degree of separability. The higher the AUC, the better the model is capable of distinguishing between apnea and absence of apnea classes.

5 Hardware Simulation: FNN Design Process

The second step in the software-hardware co-simulation process is the design of FNN model in hardware. After obtaining optimal results with acceptable accuracy (over 80% [26]) the trained model was realized into digital hardware system design. All the corresponding learned parameters (weights and biases) were extracted from the network and embedded onto the FPGA system. In the software model, each of the weights and biases consisted of sign floating point values. When implementing arithmetic operations on the FPGAs with real numbers containing floating-point values, the calculated power consumption increases drastically. To reduce the power consumption rate of the system each of the parameters was converted into the corresponding integer value by multiplying it with 2^6 where fractional parts were discarded [27]. The shift to 6 bits keeps the data information consistent even without the floating-point library. The dimension of weights and biases of the system consists of

9 bits where the most significant bit (MSB) represents its sign value. The digital hardware system which was designed using re-programmable hardware is targeted to have a power consumption lower than existing DL hardware accelerators available in the market.

5.1 Activation Function Design

Choosing the proper activation function for different layers is one of the most important tasks to accurately classify the input data as well as to minimize the power consumption in hardware realization. This function enables the neural networks to learn complex patterns of the data that is used in training the model. The proposed deep learning model uses two activation functions: Rectified Linear Unit (ReLU) and Sigmoid function. Due to the clinical data having positive samples with no negative values, it was feasible to use these two functions as they best fitted the model.

A comparative power consumption study report between three commonly used activation functions was performed in Vivado HLx software using FPGA Artix-7 Nexys as shown in Figure 6. Even though Hyperbolic Tan consumes lower power than Sigmoid the model accuracy decreased significantly which is nearly 61.1%(with confidence interval of ± 1.41%). Equation 9 [26] shows the hardware logic calculation of Eq. 2. This represents the piece-wise linear function equation of 9-bit sigmoid activation function and ReLU is a simple conditional if-else function. These two equations were used in designing the activation function units of the hardware FNN model.

$$Sigmoid = \begin{cases} 128, & |x| \geq 512 \\ 2^{-5} * |x| + 107, & 256 \leq |x| < 512 \\ 2^{-3} * |x| + 80, & 128 \leq |x| < 256 \\ 2^{-2} * |x| + 64, & 0 \leq |x| < 128 \end{cases} \tag{9}$$

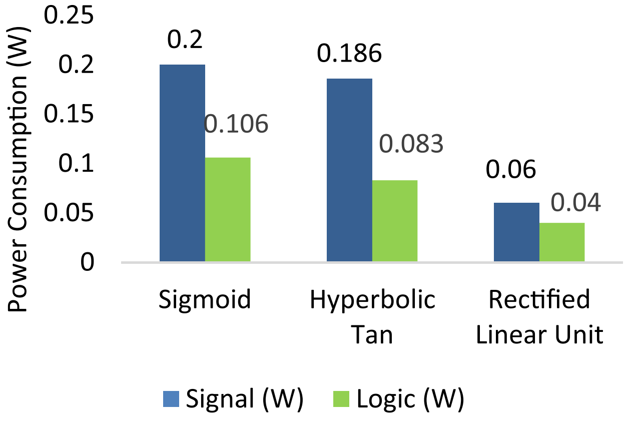

Figure 6 Comparative power consumption study between three activation functions in FPGA.

5.2 Shift Accumulate Method (SAC)

A typical digital hardware accelerator uses multiply-accumulate function (MAC) as its neuron units. According to Figure 5, the synapse of the fully connected neural network multiplies its weights to the corresponding output data from the previous layer and sends it to the neuron of the next layer to get summed and either classified (at the output layer) or passed to the next layer (at the hidden layer). As a result, many multipliers need to be designed when representing the synapse of the model. When designing energy efficient hardware model, the power consumption of the system increases at a high rate due to the usage of many multipliers. To avoid this, a shifter based low-power design technique (Shift-Accumulate, SAC) has been introduced in this work where shifters have been used instead of multipliers. During the training phase of the model in software, the weights and biases were constrained in order of magnitude of 2 while maintaining the consistency of their original values. Figure 7 illustrates the power consumption study between a 12-bit array multiplier and a 15-bit shifter with maximum amount of bit considered in this design.

According to Figure 7 the 12-bit multiplier consumes around 13 × times more dynmic power than that of the 15-bit shifter and requires significantly higher number of I/O ports which result in high junction temperature when executed on normal general purpose hardware such as Artix-7 FPGAs. The 12-bit shifters had fewer I/O ports and did not exceed the maximum temperature. All power consumption studies were executed using Vivado HLx software.

6 Experimental Results and Discussions

For the development of the model 70% of the data was used for training and validation and the rest 30% for testing. The model was trained for 1000 epochs with a tenfold cross validation on the training set having a mini-batch size of 10. The proposed

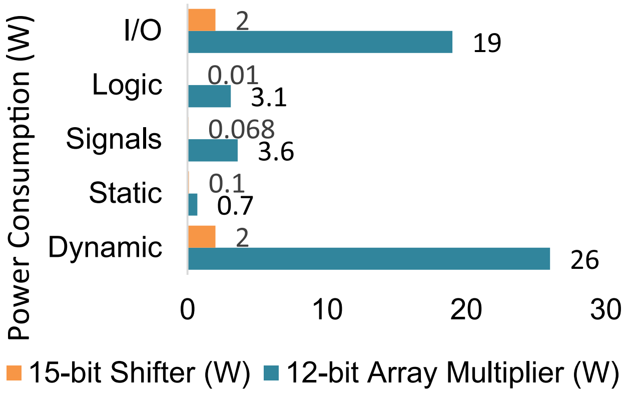

Figure 7 Comparative power consumption study between 12-bit multiplier and 15-bit shifter.

Table 1 Performance Evaluation Chart of the FNN.

Evaluation	Score (%)
Accuracy	88. 94 (Confidence Interval: ±2.17)
Precision	0: 88 1: 91
Sensitivity	0: 94 1: 82
F-1 Score	0: 91 1: 86

FNN model showed promising and acceptable results by detecting sleep apnea with an accuracy of nearly 88% whereas traditional PSG testing showed near 80% accuracy in [26].

6.1 FNN Evaluation Results

Table 1 presents the performance evaluation metrics by evaluating Eqs. 5 to 8. Figure 8 is the learning behavior with the training and testing set of the model and Figure 9 graphically illustrates the performance measurement results based on the AUC-ROC curve showing around 92% sensitivity. Table 2 showcases a comparative study between different machine learning models that used ApneaECG database and our proposed model showed the higher performance results compared to the others.

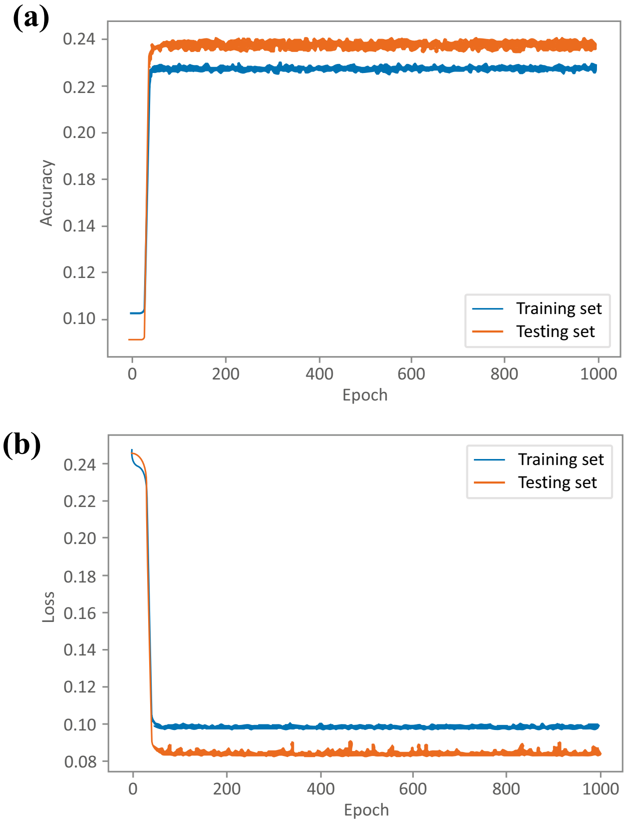

Figure 8 Training and testing performance learning curve. (**a**) learning curve for accuracy and (**b**) learning curve of model loss.

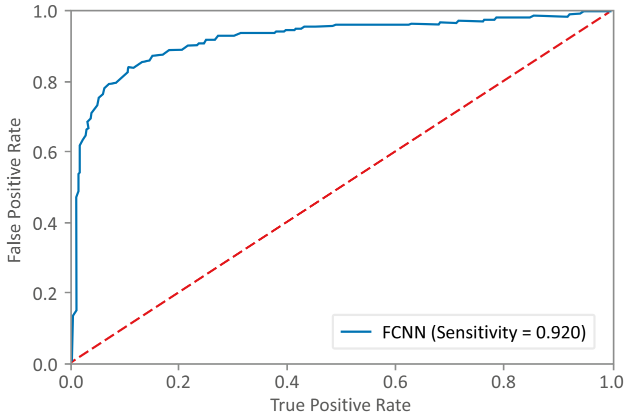

Figure 9 Performance measurement curve AUC-ROC.

6.2 Hardware Realization of FNN Inference

The trained deep learning model was translated into hardware system after optimal model accuracy and compact model architecture were attained. All model parameters were extracted and used in designing each logic block of the digital hardware system. Figure 10 shows the inference hardware architecture of the proposed FNN model (shown in Figure 4), on Artix-7 Nexys FPGA. Each hidden IP block consists of multiple shifters, adders, and activation functions.

The testbench of the system fed with normalized and processed clinical data is presented in Figure 11. According to Figure 11, the proposed hardware system successfully detects apneic and normal conditions by showing values of ‘1’ and ‘0’ respectively on the output registers.

6.3 Power Consumption of the FNN Hardware Design

The FNN inference digital hardware module has a power consumption of nearly 34 W. The power consumption report of the proposed model is presented in Table 3

Table 2 Comparative Study of the Performance of the Proposed FNN Model with other ML Models Reported in Literature.

Classifier	Accuracy (%)	Sensitivity (%)	F-1 Score	Reference
HMM-SVM	86.2	82.6	88.4	[28]
LS-SVM	84.7	84.7	84.7	[29]
Decision Fusion	83.8	88.9	88.4	[29]
TW-MLP	87.33	85.1	88.7	[30]
FNN	**88.94**	**94**	**91**	**Our Work**

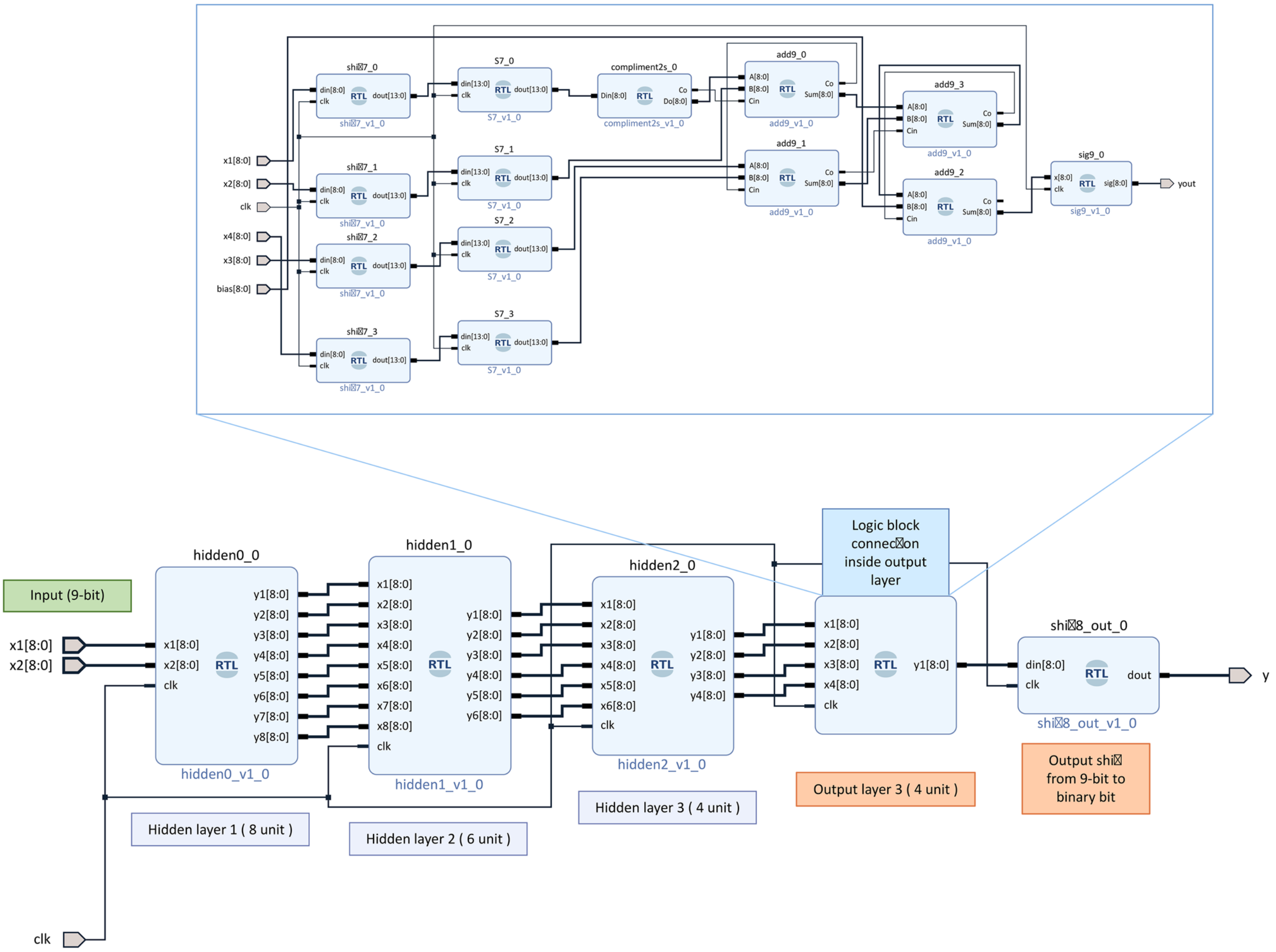

Figure 10 IP block diagram of three hidden layer (8–6–4) feedforward neural network inference digital hardware design on Artix-7 Nexys FGPA using Vivado HLx software.

which is typical for deep learning based hardware accelerators [31]. The proposed hardware model successfully detected each SA conditions within 12.24 ms with the testbench provided in Vivado HLx software. Table 4 shows a comparison of the power consumption of commercially available accelerators with that of the proposed deep learning embedded hardware system. This comparison data confirms that the proposed hardware architecture tested on a general purpose Artix-7 Nexys FPGA board significantly reduces the power consumption compared to the rest. The low-power consumption and the fast execution time of the proposed hardware-software co-design

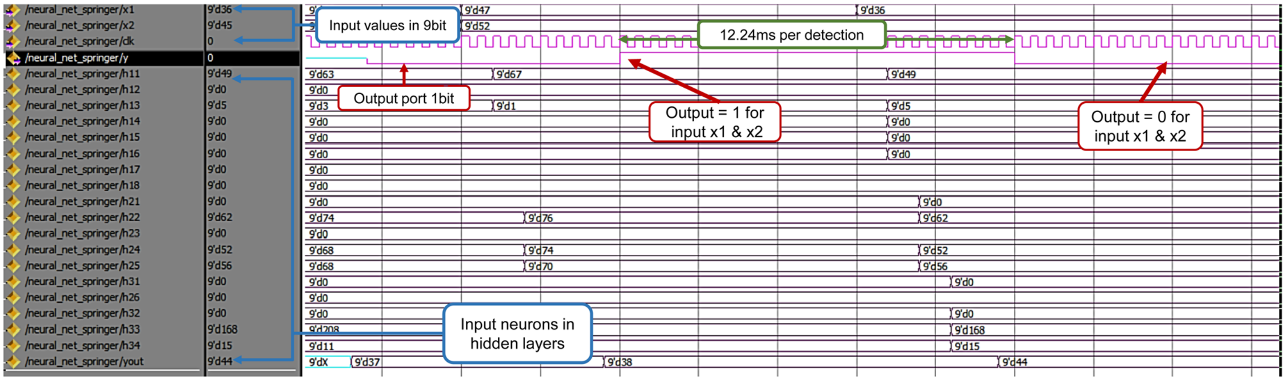
/neural_net_springer/x1	9'd36
/neural_net_springer/x2	9'd45
/neural_net_springer/clk	0
/neural_net_springer/y	0
/neural_net_springer/h11	9'd49
/neural_net_springer/h12	9'd0
/neural_net_springer/h13	9'd5
/neural_net_springer/h14	9'd0
/neural_net_springer/h15	9'd0
/neural_net_springer/h16	9'd0
/neural_net_springer/h17	9'd0
/neural_net_springer/h18	9'd0
/neural_net_springer/h21	9'd0
/neural_net_springer/h22	9'd62
/neural_net_springer/h23	9'd0
/neural_net_springer/h24	9'd52
/neural_net_springer/h25	9'd56
/neural_net_springer/h31	9'd0
/neural_net_springer/h26	9'd0
/neural_net_springer/h32	9'd0
/neural_net_springer/h33	9'd168
/neural_net_springer/h34	9'd15
/neural_net_springer/yout	9'd44

Figure 11 Testbench of the proposed FNN hardware resulting in “0” and “1” for normal and apneic conditions respectively when feeding in testing dataset.

Table 3 Power Consumption Report of Hardware Inpmelented FNN Model.

Artix-7 Nexys	Power Consumption (W)
Signal	14
Logic	18.5
I/O	0.82
Total Power	**34.37** **Dynamic: 33.3** **Static: 0.446**

Table 4 Power Consumption of Commercially Available Machine Learning Hardware.

ML Hardware	Power Consumption (W)
Google TPU v2 (float 16) [31]	200 est
Google TPU v3 (float 32) [31]	250 est
ML server: Tensoflow (i9-AVX2/i9-SSE4) [32]	205
Proposed Hardware Model	**34.37**

procedure is undoubtedly a promising step towards developing customized deep learning based decision support tools for numerous applications.

7 Conclusions

A deep learning inference based compact hardware design for sleep apnea detection has been presented. The overall prediction accuracy for unseen test data is about 88%. The proposed embedded system consumes less than 34 W of power. The high accuracy and low power consumption of the proposed system will facilitate its application in critical medical diagnosis such as wearable smart apnea monitoring system. The proposed non-invasive, wearable and portable smart biomedical device is a significant improvement over the existing technologies for monitoring and detection of sleep apnea. In future, the automated alarm system of the proposed system will cut down long working hours for both the caregivers in healthcare facilities and the sleep experts in sleep laboratories. The three-step design process introduced in this paper will also aid in developing cost-effective smart wearable biomedical devices for other applications.

References

1. Vanegas, E., Igual, R., & Plaza, I. (2020). Sensing systems for respiration monitoring: A technical systematic review. *Sensors, 20*(18), 5446. https://doi.org/10.3390/s20185446
2. Penzel, T., Schöbel, C., & Fietze, I. (2018). New technology to assess sleep apnea: wearables, smartphones, and accessories. *F1000 Research, 7*. https://doi.org/10.12688/2Ff1000research.13010.1
3. Gottlieb, D. J., & Punjabi, N. M. (2020). Diagnosis and management of obstructive sleep apnea: A review. *JAMA, 323*(14), 1389–1400. https://doi.org/10.1001/jama.2020.3514
4. Mahbub, I., Hasan, M. S., Pullano, S. A., Quaiyum, F., Stephens, C. P., Islam, S. K., Fiorillo, A. S., Gaylord, M. S., Lorch, V., & Beitel, N. (2015). A low power wireless apnea detection system based on pyroelectric sensor. *Topical Conference on Biomedical Wireless Technologies, Networks, and Sensing Systems*, 1–3. https://doi.org/10.1109/BIOWIRELESS.2015.7152130
5. Mendonça, F., Mostafa, S. S., Ravelo-García, A. G., Morgado-Dias, F., & Penzel, T. (2018). Devices for home detection of obstructive sleep apnea: A review. *Sleep medicine reviews, 41*, 149–160. https://doi.org/10.1016/j.smrv.2018.02.004
6. Mahbub, I., Pullano, S. A., Wang, H., Islam, S. K., Fiorillo, A. S., To, G., & Mahfouz, M. R. (2017). A low-power wireless piezoelectric sensor-based respiration monitoring system realized in CMOS process. *IEEE Sensors Journal, 17*(6), 1858–1864. https://doi.org/10.1109/JSEN.2017.2651073
7. Shamsir, S., Hesari, S. H., Islam, S. K., Mahbub, I., Pullano, S. A., & Fiorillo, A. S. (2018). Instrumentation of a pyroelectric transducer-based respiration monitoring system with wireless telemetry. In *2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 1–6. https://doi.org/10.1109/I2MTC.2018.8409796
8. Pullano, S. A., Mahbub, I., Bianco, M. G., Shamsir, S., Islam, S. K., Gaylord, M. S., Lorch, V., & Fiorillo, A. S. (2017). Medical devices for pediatric apnea monitoring and therapy: Past and new trends. *IEEE reviews in biomedical engineering, 10*, 199–212. https://doi.org/10.1109/RBME.2017.2757899
9. Shamsir, S., Hassan, O., & Islam, S. K. (2020). Smart infant-monitoring system with machine learning model to detect physiological activities and ambient conditions. In *2020 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 1–6. https://doi.org/10.1109/I2MTC43012.2020.9129295
10. Hassan, O., Shamsir, S., & Islam, S. K. (2020). Machine Learning Based Hardware Model for a Biomedical System for Prediction of Respiratory Failure. In *2020 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, 1–5. https://doi.org/10.1109/MeMeA49120.2020.9137291
11. Yüzer, A. H., Sümbül, H., & Polat, K. (2020). A novel wearable real-time sleep apnea detection system based on the acceleration sensor. *IRBM, 41*(1), 39–47. https://doi.org/10.1016/j.irbm.2019.10.007
12. Dey, D., Chaudhuri, S., & Munshi, S. (2018). Obstructive sleep apnoea detection using convolutional neural network based deep learning framework. *Biomedical engineering letters, 8*(1), 95–100. https://doi.org/10.1007/s13534-017-0055-y
13. Kristiansen, S., Nikolaidis, K., Plagemann, T., Goebel, V., Traaen, G. M., Øverland, B., Aakerøy, L., Hunt, T. E., Loennechen, J. P., Steinshamn, S. L., & Bendz, C. H. (2021). Machine Learning for Sleep Apnea Detection with Unattended Sleep Monitoring at Home. *ACM Transactions on Computing for Healthcare, 2*(2), 1–25. https://doi.org/10.1145/3433987
14. Azimi, H., Xi, P., Bouchard, M., Goubran, R., & Knoefel, F. (2020). Machine Learning-Based Automatic Detection of Central Sleep Apnea Events From a Pressure Sensitive Mat. *IEEE Access, 8*, 173428–173439. https://doi.org/10.1109/ACCESS.2020.3025808
15. Álvarez, D., Cerezo-Hernández, A., Crespo, A., Gutiérrez-Tobal, G. C., Vaquerizo-Villar, F., Barroso-García, V., Moreno, F., Arroyo, C. A., Ruiz, T., Hornero, R., & Del Campo, F. (2020). A machine learning-based test for adult sleep apnoea screening at

home using oximetry and airflow. *Scientific reports, 10*(1), 1–12. https://doi.org/10.1038/s41598-020-62223-4

16. Ye, G., Yin, H., Chen, T., Chen, H., Cui, L., & Zhang, X. (2021). FENet: A Frequency Extraction Network for Obstructive Sleep Apnea Detection. *IEEE Journal of Biomedical and Health Informatics*. https://doi.org/10.1109/JBHI.2021.3050113
17. Mendonça, F., Mostafa, S. S., Morgado-Dias, F., & Ravelo-García, A. G. (2020). An oximetry based wireless device for sleep apnea detection. *Sensors, 20*(3), 888. https://doi.org/10.3390/s20030888
18. Hassan, O., Parvin, D., & Islam, S. K. (2020). Machine Learning Model Based Digital Hardware System Design for Detection of Sleep Apnea Among Neonatal Infants. 607–610.
19. Penzel, T., Moody, G. B., Mark, R. G., Goldberger, A. L., & Peter, J. H. (2000). The apnea-ECG database. *In Computers in Cardiology, 27*, 255–258. https://doi.org/10.1109/CIC.2000.898505
20. Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., Mietus, J. E., Moody, G. B., Peng, C. K., & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation, 101*(23), 215–220. https://doi.org/10.1161/01.CIR.101.23.e215
21. LeCun, Y., Bengio, Y., & Hinton, G. (2015). *Deep learning. nature, 521*(7553), 436–444. https://doi.org/10.1038/nature14539
22. Nikonov, D. E., & Young, I. A. (2019). Benchmarking Physical Performance of Neural Inference Circuits. *arXiv preprint arXiv:1907.05748*. https://arxiv.org/abs/1907.05748v1
23. Agarap, A. F. (2018). Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*. https://arxiv.org/abs/1803.08375v2
24. Tisan, A., Oniga, S., Mic, D., & Buchman, A. (2009). Digital implementation of the sigmoid function for FPGA circuits. *ACTA Technica Napocensis, 50*(2), 15–20.
25. Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. https://arxiv.org/abs/1412.6980v9
26. Bhattacharjee, R., Benjafield, A., Blase, A., Dever, G., Celso, J., Nation, J., Good, R., & Malhotra, A. (2021). The accuracy of a portable sleep monitor to diagnose obstructive sleep apnea in adolescent patients. *Journal of Clinical Sleep Medicine,* jcsm-9202.
27. Tsmots, I., Skorokhoda, O., & Rabyk, V. (2019). Hardware implementation of sigmoid activation functions using FPGA. In *IEEE 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM),* 34–38. https://doi.org/10.1109/CADSM.2019.8779253
28. Song, C., Liu, K., Zhang, X., Chen, L., & Xian, X. (2015). An obstructive sleep apnea detection approach using a discriminative hidden Markov model from ECG signals. *IEEE Transactions on Biomedical Engineering, 63*(7), 1532–1542.
29. Varon, C., Caicedo, A., Testelmans, D., Buyse, B., & Van Huffel, S. (2015). A novel algorithm for the automatic detection of sleep apnea from single-lead ECG. *IEEE Transactions on Biomedical Engineering, 62*(9), 2269–2278.
30. Wang, T., Lu, C., & Shen, G., (2019). Detection of sleep apnea from single-lead ECG signal using a time window artificial neural network. *BioMed Research International, 2019*.
31. Jouppi, N. P., Young, C., Patil, N., Patterson, D., Agrawal, G., Bajwa, R., Bates, S., Bhatia, S., Boden, N., Borchers, A., & Boyle, R. (2017). In-datacenter performance analysis of a tensor processing unit. In *Proceedings of the 44th annual international symposium on computer architecture,* 1–12. https://doi.org/10.1145/3079856.3080246
32. Reuther, A., Michaleas, P., Jones, M., Gadepally, V., Samsi, S., Kepner, J. (2020). Survey of Machine Learning Accelerators. In *IEEE High Performance Extreme Computing Conference (HPEC),* 1–12. https://doi.org/10.1109/HPEC43674.2020.9286149

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.