An Efficient Embedded Hardware for High Accuracy Detection of Epileptic Seizures

Mushfiq U. Saleheen, Homa Alemzadeh, Ajay M. Cheriyan, Zbigniew Kalbarczyk, Ravishankar K. Iyer

{msalehe2, alemzad1, cheriya1, kalbarcz, rkiyer}@illinois.edu

Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, USA

Abstract—**This paper presents design, implementation and evaluation of an efficient embedded hardware for accurate automated detection of epileptic seizures. Three hardware configurations are proposed and evaluated in terms of accuracy of detection, utilization of hardware resources, and power consumption. The results show that a solution based on combination of the statistical function of variance (for feature extraction) and an artificial neural network (ANN) classifier allows to achieve high detection accuracy (99.18%) with moderate hardware footprint (around 44% of the FPGA resources). Furthermore, use of algorithmic and architectural optimization techniques (reduction in precision of the fixed-point number representation and reuse of hardware components) allows reducing hardware footprint by a factor of 4.4 and power consumption by a factor of 2.7 as compared with an un-optimized hardware configuration. High accuracy, real-time detection, simplicity, power efficiency and small hardware footprint make our approach a good candidate for embedded epileptic seizure detection implementation.**

Keywords- **Biomedical Devices, Biomedical Signal Processing, Epileptic Seizure Detection, Reconfigurable Hardware.**

I. INTRODUCTION

Epileptic seizures are characterized by recurrent atypical brain activities with unusual excessive electrical discharges [1]. Persons suffering from episodes of such abnormal brain functions are referred to as having epilepsy. The physical symptoms of seizure can range from none to unprovoked muscle contractions, numbness, visual anomaly, loss of memory and loss of consciousness. Such episodes can be of very short duration or can be continuous until intervention occurs. Epilepsy is a chronic neurological disorder affecting about 50 million people around the world [2]. One of the commonly used methods of diagnosing epilepsy is by measuring brain electrical activity with scanning electroencephalography or EEG signals. A patient suffering from seizure is often required admittance to a healthcare facility where his or her EEG is recorded over a prolonged period of time. Such records help in obtaining specific characteristics of the seizure and determining the precise location of the origin of seizures in the cases where surgery is needed to treat the seizure occurrences.

For the purpose of diagnosis, the EEG records are carefully inspected by an experienced health professional to detect the seizure's onset and characteristics. An automated detection method for epileptic seizure can be helpful as it can perform this tedious and lengthy inspection process without human intervention, and the findings can be used to aid the final diagnosis by a clinician. In the context of a healthcare facility, such automated detection mechanisms can save significant amount of doctors' time. It can also reduce the number of clinical visits that the patient has to make and the amount of time she has to stay in the hospital for such diagnosis procedures if it can potentially be performed at home. Also, when used on a patient in a home setting, this automated system can generate an alarm and a request for medical help when positive detections are made.

The general trend in developing embedded biomedical devices for health monitoring has focused on analysis of signals for visualization purposes or development of high accuracy detection algorithms using complex signal processing and statistical computations, without considering the real hardware implementation constraints. An important consideration in implementing wearable embedded devices is battery life. Generally, microprocessors, because they are less power-efficient than special low-power processors, are not capable of satisfying this requirement. On the other hand, accurate analysis of different biomedical signals can easily lead to a high degree of computational complexity and requires significant processing power that commercial off-the-shelf (COTS) microcontrollers (e.g. TI MSP430 family [3]) cannot support. Moreover, although the low-power DSP processors (e.g., TMS320 DSP family from TI [4] and SHARC floating-point DSPs from Analog Devices [5]) support high-performance and power efficiency for medical applications, they limit the degree of flexibility and customization often demanded for implementation of embedded health monitoring systems and still cannot achieve the power efficiency of custom implementations.

This paper presents design, implementation and evaluation of embedded low-power hardware for accurate automated detection of epileptic seizures. The specific contributions of this study are:

- Design and FPGA (field programmable gate array) based implementation of embedded hardware for the EEG signal processing and automated real-time detection of seizure events.

- Evaluation of the prototype hardware configurations ((i) sample entropy and ANN (artificial neural network),(ii) variance and predetermined threshold value, and (iii)

variance and ANN) in terms of: accuracy of detection and utilization of hardware resources. Our measurements indicate that the detection scheme which combines use of variance (for feature extraction) and ANN (for classification) achieves high detection accuracy (99.18%) with moderate hardware footprint (consumes about 44% of the FPGA resources).

- Evaluation of algorithmic and architectural optimization techniques to reduce hardware overhead and power consumption while preserving the high detection accuracy. The measurements (for the *variance and ANN* configuration) show that: (i) reduction in precision of the fix-point number representation and (ii) reuse of hardware components can significantly reduce the hardware footprint (by factor of 4.4) and power consumption (by factor of 2.7) as compared with an un-optimized hardware configuration.

II. RELATED WORK

There are two distinctive approaches taken for automated seizure detection: software- or hardware-based solutions.

For *software-based epileptic seizure detection*, the widespread availability of powerful computing platforms and sophisticated signal-processing/statistical analysis tools have led to a great deal of attention on non-linear neural network-based classifiers. [6] has shown the usefulness of multiple signal classification (MUSIC), autoregrssive (AR), and periodogram to compute power spectral density (PSD) of patient EEG signals. The PSD is fed to conventional logistic regression (LR) as well as multilayer perceptron neural network (MLPNN) classifiers where the MLPNN outperformed the LR method with both methods generating accuracies over 88%. Computing embedding dimension of EEG using Cao's method followed by probabilistic neural network (PNN) is employed in [7] with accuracy reaching 100%. The time and frequency multi-scale property of wavelet transformation (WT) is exploited in [8] as a variant called lifting-based discrete wavelet transform (LBDWT) along with LR and MLPNN; accuracy for both classifiers were around 90%. Patnik et al were able to reach accuracy near 99% using similar WT technique but employing a genetic algorithm to obtain the training set to be fed to the neural network along with harmonic weights as post-classification optimizations to boost the accuracy [9]. The Lyupunov exponent, a dynamical measure of the stability of a steady-state behavior, is investigated in [10] along with WT and PSD for feature extraction from the EEG time series. Used with an MLPNN, an accuracy of 96.3% was achieved. Methods based on entropy that quantify the randomness in a time series have also proved to be of considerable value for EEG signal analysis.

[11] investigated approximate entropy with the Elman neural network and PNN showing accuracy up to 100%. Spectral entropy was used in [12] with a variant of recurrent neural network to achieve accuracy of 99.6%. Recently, Kumar et al have compared wavelet entropy, sample entropy, and spectral entropy used with the recurrent Elman network for identifying seizures with accuracy up to 99.75% [13]. [14] presents a comparison of a variety of seizure detection approaches based on different kinds of signal processing and statistical methods.

Wearable *hardware embedded devices* capable of capturing and analyzing EEG signals and detecting epileptic seizure onsets have attracted attention in recent years. An ASIC implementation of seizure detection using root mean square, number of maxima and minima, line length, and non-linear energy is realized by Patel et al in [15]. Using low power optimization techniques such as data bit width reduction, down-sampling, and other hardware approximations, they have managed to keep the power consumption of the design to 10.8 μW with accuracy of 87.7%. [16] used a flexible threshold based event and inter-event-interval measurements to detect seizures that achieved an accuracy of about 80%. A programmable threshold, along with computation of rhythmicity is studied in [17], achieving a sensitivity of 95.3% and a selectivity of 88.9% while consuming less than 350 nW of power. Similar low power approaches are taken in [18] and [19].

As is evident from the discussion above, the software-based approaches utilize sophisticated and often computationally intensive algorithms for feature extraction as well as classification without regard to detection latency, hardware cost, or power consumption. For the hardware based detectors, the main focus tends to be on low power consumption to achieve extended battery life often at the cost of lower overall accuracy. The focus of our study is to provide a small footprint and low power hardware design and implementation for epileptic seizure detection algorithms while achieving a detection performance comparable to software-based approaches.

III. AUTOMATED SEIZURE DETECTION

The overall process in automated EEG-based seizure detection is shown in Figure 1. EEG signal processing starts with the collection of analog electrical waveforms obtained from electrodes placed on the scalp using the international 10-20 system [20]. The analog waveforms are sampled at a desired frequency using ADC, filtered for noise and artifacts (due to movement, power-line interference), and fed to the feature extraction stage.
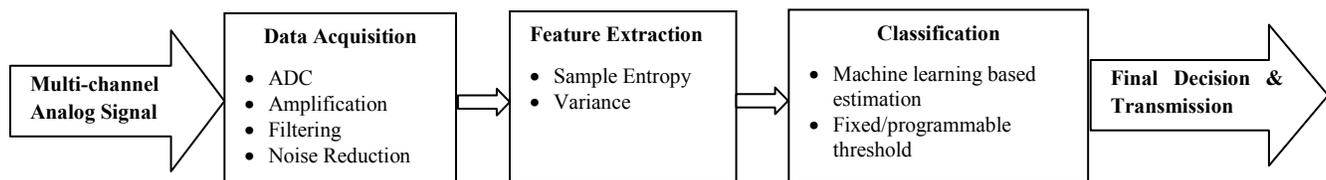
Figure 1. EEG based seizure detection flow.
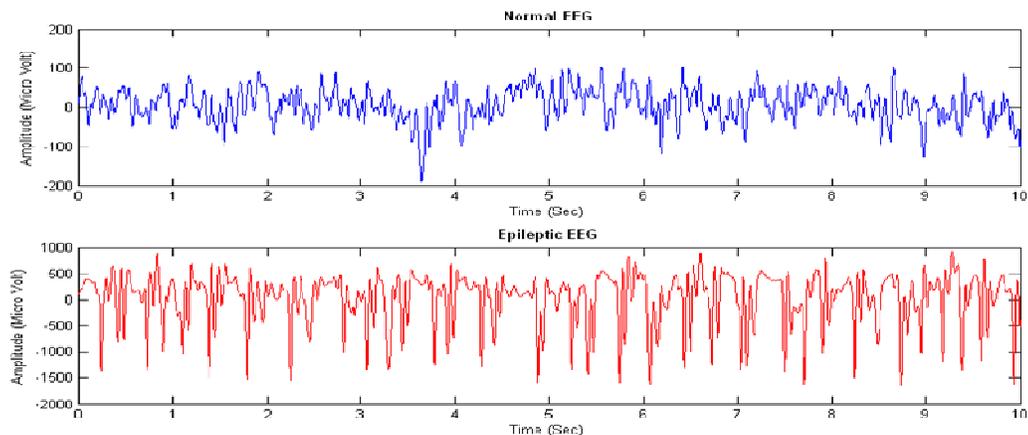
Figure 2. Normal EEG and epileptic seizure EEG.

The feature extraction stage analyzes the digitized signals using appropriate statistical and/or signal transformation functions to extract signal characteristics that are helpful in differentiating abnormal events from normal ones. In addition to highlighting the inherent characteristics of the signal, the feature extraction stage can also help minimize the signal space so that the computation load for the next stage is reduced. Finally, a classifier is employed which either uses a machine learning-based algorithm or a threshold value to determine the appropriate class of the analyzed signal. For a portable embedded device, additional steps such as displaying detection results, generating alarms for abnormal event detection, and securely transmitting the detection results to an offsite server/storage location may be employed.

For the purpose of this paper, a digital database of actual EEG signals collected by Department of Epileptology at the University of Bonn in Germany [21] is used to evaluate the seizure detection algorithms. The database contains five sets with each set consisting of 100 single channel EEG segments with the length of 23.6 seconds. The five sets (A-E) are: *Set A*-surface EEG recordings of five healthy relaxed individuals with eyes open, *Set B* – surface EEG of five healthy relaxed individuals with eyes closed (B), *Set C* – seizure free EEG of five patients from the hippocampal formation of the opposite hemisphere of the brain, *Set D* – seizure free EEG of five patients from the epileptogenic zone, and *Set E* – seizure EEG of five patients from different sites. After analog-to-digital conversion of the amplified signal, digital data were written at 173.61 Hz. For this paper, only sets A and E are used to differentiate normal EEG from epileptic seizure EEG.

IV.   EPILEPTIC SEIZURE DETECTION SCHEMES

In this section we introduce common techniques for feature extraction and classification of signals in an automated seizure detection scheme.

The main purpose of the feature extraction stage is to generate a minimal set of data that emphasizes the inherent characteristics of the base signal in a way that the feature set of a normal EEG becomes significantly different from the feature set of an epileptic seizure EEG. In order to provide transparent detection of abnormal events, it is also necessary

to complete detection computations within a fixed time interval and before the next batch of digitized signals arrive. Since the selected EEG database uses a sampling rate of 173.61 Hz, we chose our time window of computation to be 1 second consisting of 173 data points. Our detection schemes perform the necessary computations on these 173 data points to generate a binary decision of normal/abnormal events and then move on to the next batch of 173 data points.

Figure 2 shows examples of normal and epileptic seizure EEGs. Preliminary observations reveal that the seizure waveform has more regularity than the normal waveform in the form of large amplitude spikes. The long spikes seem to appear approximately 4 times every second. Such regularity, on the other hand, is not apparent in the normal waveform. This observation indicates that the 1 second waveform can be subdivided into 4 sections where each section can be used as the smallest unit of data upon which feature extraction analysis can be performed to differentiate the seizure data from normal data.

A. Sample Entropy for Feature Extraction

Sample Entropy (SampEn) is a measure of the randomness in a time series. Sample entropy is expressed by the negative natural logarithm of the probability that two subsets of a time series that match each other for a specific number of points or length will also match for the next point [22]. SampEn for a time series sample can be calculated as:

$$SampEn(m,r,N) = -\log\left(\sum_{i=1}^{N-m} A_i / \sum_{i=1}^{N-m} B_i \right) \qquad (1)$$

where m is the pattern length, r is the tolerance and N is the number of data points in the sample time series.

With values of m and r selected, the algorithm starts by scanning the N point sample time series for matches of the first pattern of length m and the first pattern of length $m+1$. Matches of length m and $m+1$ are stored in A and B respectively. The scanning and counting continues for all subsequent patterns of length m and $m+1$, and counts are always added to A and B respectively. Finally, the negative natural logarithm of the ratio of A and B are computed to obtain the SampEn value.
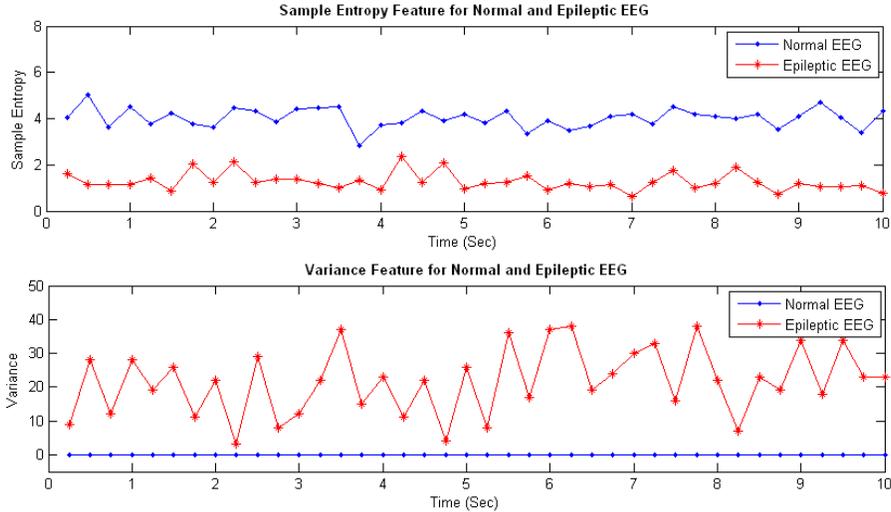
Figure 3. Sample Entropy and Variance Features extracted from normal and seizure EEG data of Figure 2.

As mentioned earlier, the amount of randomness is a characteristic that differentiates normal EEG from epileptic seizure EEG. As evident in Figure 2, normal EEG signal is more random than seizure EEG signal which has more predictable regular spikes. Such observation implies the applicability of SampEn as a measure of randomness to detect seizure waveforms from normal ones. For this paper, we chose *m = 1* as suggested in [22] and *r = 0.0001* (standard deviation of data set). The top graph in Figure 3 shows the SampEn plot of two data sets calculated using *N = 42*, where the SampEn values (3 to 5.5) for normal EEG are significantly larger than that of seizure EEG (0.5 to 2.5).

B. Variance for Feature Extraction

Variance is a measure of the variation of the data set from its mean. For a set of sample points $x_1, x_2, ..., x_N$ the variance of the set is calculated as:

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^{N} \left(x_i - \bar{x} \right)^2 \tag{2}$$

where $\bar{x}$ is the mean of the set. A high value of variance indicates that the set of data has wide deviation from the mean while a small variance indicates closely-spread data. In the case of EEG data, the epileptic seizure is characterized by high amplitude spikes which are not present in normal data. Hence, for a given time window, the spread, i.e. the variance of data samples in seizure data, should be significantly higher than that of normal data. This indicates that variance can be used as a feature extraction method to differentiate between seizure and normal EEG data. The result of computing variance for 10 seconds of normal and seizure data are shown in the bottom plot of Figure 3. In both cases, each 1-second window of data has been divided into 4 sections and then variance is computed for each section. Also, the computed variance values have been scaled down to minimize the number of digits needed for representation and hence reduce the complexity of implementation in hardware.

C. Artificial Neural Network for Classification

A multilayer perceptron artificial neural network (ANN) is a non-linear classifier that incorporates the biological model of connected neurons. It maps a set of inputs to a set of outputs using weights to minimize the difference errors. Theoretically, an ANN can map an arbitrary complex relationship with an acceptable error margin. Generally, a neural network consists of an input layer, one or more hidden layer and an output layer. Each layer consists of neurons whose exact number is determined by trial and error. For example, a 4-3-1 neural network [23] with 4 input neurons, 3 hidden layer neurons, and 1 output neuron is shown in Figure 4. Each neuron is connected to all the neurons in the previous layer with specific weights associated with each of the connections. These weights are determined based on the importance of the connections. The computation of a neuron output is performed in two stages. First, the weighted sum of the outputs of the connected neurons is calculated as multiply-add (MAC) operations. Next, an activation function is used to restrict the neuron output to a predefined range [24]. It relates the weighted sum result to a bounded set of values based on the specific function used to implement the relation. The activation function works as a source of non-linearity in the ANN structure.
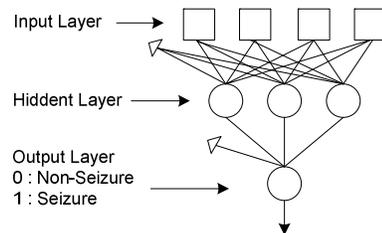
Figure 4. ANN with 4-3-1 configuration.

V. IMPLEMENTATION

While our final goal is an ASIC implementation of embedded seizure detection, for the purpose of prototyping a reconfigurable hardware, a field programmable gate array

(FPGA) platform is chosen for this work. The target FPGA is the 90nm Stratix II EP2S60F672C3 by Altera with 48,352 adaptive look-up tables (ALUTs) and the same number of dedicated logic registers. The MATLAB [25] software was used to perform the initial test implementation of the sample entropy [26] and variance. The results from MATLAB were then used with the multiple back propagation software tool [23] for ANN training. From the 200 data sets obtained from the EEG database, 120 (60 from set A and 60 from set E) were used for training and the remaining 80 were used for testing of ANN with the multiple back propagation software tool. Based on experiments, a 4-3-1 ANN with satisfactory low mean square error (~0.03) was chosen. For the actual hardware implementation, the MATLAB codes along with the ANN were coded in hardware description language. The weights obtained from the ANN software at the end of the training phase were used as constant weights for the testing phase in hardware. While the ANN software [23] had the options of using *sigmoid*, *tanh*, *Gaussian*, and *linear* functions as the activation function for neuron computation, in order to minimize required computation and resultant hardware cost we used the linear function of $F(x) = kx$ (where k is a constant) for activation of the hidden layer. However, for the single neuron on the output layer, we had to use the *non-linear sigmoid* function of $F(x) = \dfrac{1}{1 + e^{-kx}}$ for activation to achieve high detection accuracy. In both cases, x is the weighted sum of the inputs that are connected to a given neuron.

In order to maintain the real-time constraints, the entire detection computations on 1 second of sampled EEG data should be performed within 1 second. For example, data sampled within the ith second should be processed while data for the $(i+1)$th second is being buffered and the processing should complete before data for the $(i+2)$th second starts to arrive. The data processing configuration is shown in Figure 5. For processing the EEG data sampled at 173 Hz, each 1 second of data is buffered and divided into 4 windows, each containing 42 samples.
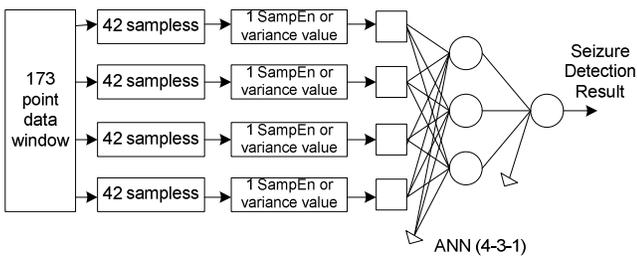
Figure 5. Processing configuration.

In the feature extraction phase, 1 SampEn or variance value is computed based on the 42 samples of each of the windows. The 4 feature values obtained are then applied to the 4-3-1 ANN to obtain the final binary detection result.

VI. EVALUATION RESULTS

In this section, we present the evaluation results for three different implementations which correspond to combinations of (i) sample entropy and ANN, (ii) variance and

predetermined threshold value, and (iii) variance and ANN. We also discuss and assess a number of design optimizations to reduce the hardware complexity and power consumption.

A. Hardware Utilization and Performance

The detection performance in terms of accuracy, sensitivity and specificity is shown in Table I. Both of the feature extraction algorithms (SampEn and variance), when accompanied with an ANN classifier, provide high detection accuracy of more than 99%. Although the differences are small, SampEn is superior to variance in terms of accuracy as well as specificity. On the other hand, using a predetermined threshold comparison for classification stage provides the best performance in terms of sensitivity but the overall accuracy is reduced.

The amount of ALUTs utilized for the detection processes based on ANN classification are shown in Table II. The SampEn along with an ANN classifier requires more resources than the target FPGA can provide. One reason for such large amount of logic usage is because of SampEn's inherently sequential nature of computation that requires a significant amount of sample history to be stored in the buffers. Also, the algorithm requires the computation of a nested loop which exhibits loop counter dependency and hence destroys any parallelism that can be extracted from it. On the other hand, the variance module requires a significantly smaller logic count for onboard implementation because of the absence of the loops and presence of rather simple operations.

Therefore, the third detection scheme, based on the variance accompanied with ANN, provides the best balance between detection accuracy and hardware usage.

TABLE I.   DETECTION PEFORMANCE

Measurement	SampEn + ANN (%)	Variance + Threshold (%)	Variance + ANN (%)
Overall Accuracy	99.73	98.52	99.18
Sensitivity	99.46	99.47	98.60
Specificity	100.00	97.61	99.78

TABLE II.   HARDWARE FOOTPRINT

SampEn + ANN			Variance + ANN		
Module	ALUT Usage	Percentage	Module	ALUT Usage	Percentage
SampEn	36964	76.44	Variance	2490	5.15
ANN	15432	31.91	ANN	18683	38.64
Total	52396	108.35	Total	21173	43.79

B. Optimizations

In this section we investigate optimization techniques in the implementation of variance plus ANN-based detection scheme to reduce complexity and power consumption.

Two major components of power consumption on reconfigurable hardware are static power and dynamic power. Static power is mostly technology-dependent and is consumed

regardless of the logic operations in the circuit. Dynamic power is the power consumed because of the switching activities when the logic circuit is operational. The following optimization techniques are primarily focused on improving the dynamic power consumption by reducing logic elements and signal activities in the design. It should be noted that, while the power consumption analysis performed here is aimed at FPGAs, the proportion of power consumption by different components of the detection scheme would be largely similar for an ASIC implementation.

1) Reduced Precision. Most of the computations for the ANN part of the design were initially implemented using 32-bit fixed-point numbers. The format of a fixed-point number is sign bit (s) + integer bits (i) + fraction/precision bits (p). As an optimization we considered lowering bit width representation without compromising the detection performance. Table III shows the impact of reduced precision on performance. $p = 5$ bits of precision along with $i = 9$ bits for integer part and $s = 1$ bit for sign needs a total word-length of 15 bits which provides the best acceptable accuracy results of 99.08% detection rate. Hence, all the fixed-point numbers were reduced from 32 bits to 15 bits representation. Arithmetic operations using fixed-point numbers were also scaled accordingly to maintain the same scaling factor. The sigmoid function in the ANN was also approximated as a look-up table to reduce computation in hardware.

TABLE III. PRECISION AND PERFORMANCE

Precision (bits)	Total Word-Length (bits)	Overall Accuracy (%)	Sensitivity (%)	Specificity (%)
1	11	47.45	100.00	48.75
2	12	84.02	68.04	100.00
3	13	92.23	86.45	100.00
4	14	96.47	93.49	99.88
5	15	99.08	99.24	98.92

2) Folding for Component Reuse. If a hardware process has relaxed time constraints for completing its tasks, which is often the case in biomedical domain applications such as automated seizure detection, then a folding technique can be employed to reuse redundant components over time instead of performing parallel computation in a single clock cycle. While folding reduces the logic area utilization by removing unnecessary parallel components, additional control logic to perform the time-multiplexing/scheduling and memory to store intermediate values must be employed. Since the ANN structure has parallel neurons in the hidden layer and the output layer, folding is used to break the parallelism with a scheduling logic, as shown in Figure 6. The top diagram is the parallel ANN from which the folded architecture, shown in the bottom of the figure, is obtained. The time period for the scheduler is 4 cycles where the first 3 cycles complete the hidden neuron computations and then the results are fed back for the final output computation in the last cycle. Appropriate delays are implemented in the form of storage registers shown

as *nD* in the figure. Similar folding has also been used for the input normalization phase of the ANN.
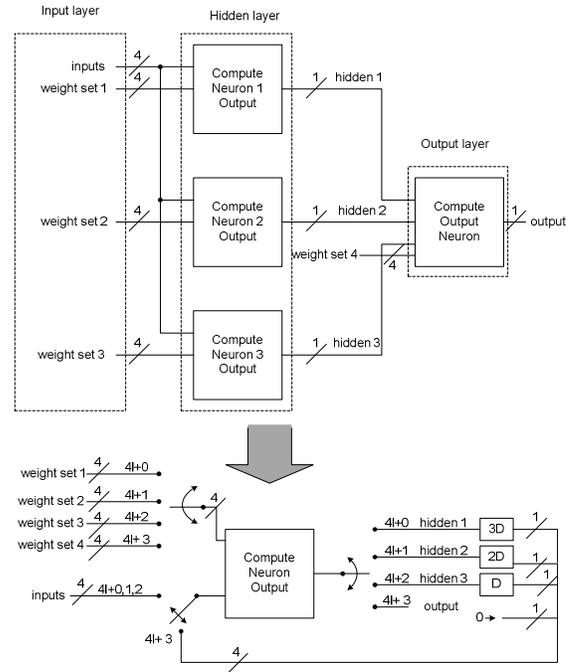
Figure 6. Folding for neuron computation.

3) Pipelining. Pipelining can effectively reduce the propagation delay in architecture by employing intermediate delays to break long paths. Although pipelining can result in increased use of registers to store intermediate values of the pipelined stages, this added footprint can be compensated by savings in power consumption as the computation paths are shortened in the pipelined architecture. The computation of a neuron in the ANN consists of repeated multiply-add (MAC) operations followed by the sigmoid function calculation, as shown in the top diagram of Figure 7. All these operations, which are computed in one clock cycle, can be broken down to multiple stages with intermediate storage registers to be pipelined as shown in the bottom diagram of Fugure 7. In this figure, *Mi, Ai, and regi* stand for multiplier, adder, and register respectively. The output of register 3 is used to compute the *sigmoid* function.
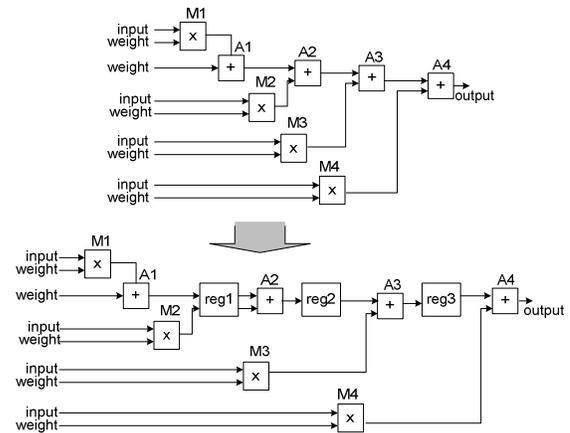
Figure 7. Pipelining the neuron computation.

The area and power consumptions of the original hardware implementation and the comparative savings for the different optimizations in those two factors are shown in Table IV. Reduced precision and folding are found to be effective in significantly reducing logic usage as well as dynamic power. Less precision occupies less logic and hence contributes to reduced dynamic power as there are fewer logic signals that toggle for a given time window. Similarly, compared to the reduced precision version, adding folding saved area by 28.21% by reusing the same logic, which also resulted in reduced signal switching by spreading the signal activities of the parallel logic over multiple cycles to give power saving by 43.16%. However, the power consumption improvement resulting from pipelining was masked by the comparatively larger increase in power because of increased signal activities for the intermediate storage elements.

TABLE IV.   AREA AND POWER CONSUMPTIONS

Architecture/ Optimization	Resource Usage (ALUTs + Registers)	Resource Usage Decrease (%)	Dynamic power (mW)	Power Saving (%)
Un-optimized	21472	--	4.3	--
Reduced precision	6828	68.20	2.85	33.72
Red. prec. + Folding	4902	77.17	1.62	62.33
Red. prec.+ Fold.+ Pipeline	5022	76.61	1.63	62.09

The trend in dynamic power consumption over different proposed optimizations is shown in Figure 8. A small increase in register count can be seen in ANN folding and pipelining because intermediate values are stored. This also resulted in the increased dynamic power in the ANN for the two optimizations, as can be seen in Figure 8. However, a balance between overall area and power usage is found in the cumulative optimizations of reduced precision and folding. For this stage of the hardware, about a 77.17% ((21472-4902)/21472*100) savings in total resource usage and a 62.33% ((4.3-1.62)/4.3*100) reduction in total dynamic power over the un-optimized implementation is achieved without compromising the performance of 99.08% overall accuracy.
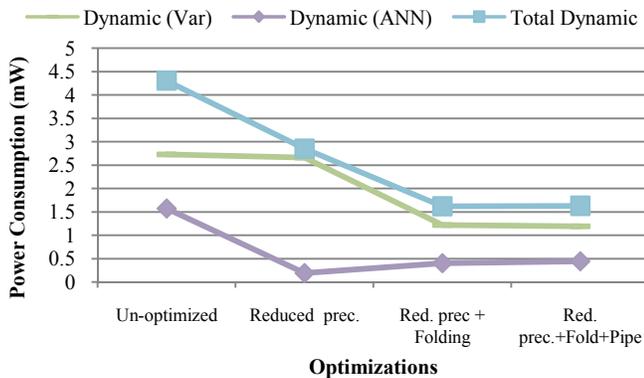
Figure 8. Power consumption vs optimizations.

VII.   CONCLUSIONS

We presented an efficient hardware implementation for real time automated seizure detection with high accuracy and low hardware overhead and power consumption. The experimental results show that by using a variance method for feature extraction accompanied with an artificial neural network (ANN) for classification we can achieve a high degree of accuracy (99.18%) while introducing a limited amount of hardware overhead. Also, by performing further optimization techniques, the hardware complexity and power consumption could be reduced as much as 77.17% and 62.33% respectively. This custom implementation can later be used in a more generic reconfigurable platform for biomedical processing.

REFERENCES

[1] R. Fisher, W. van Emde Boas, W. Blume, C. Elger, P. Genton, P. Lee, and J. Engel. "Epileptic seizures and epilepsy: definitions proposed by the international league against epilepsy (ilae) and the international bureau for epilepsy (ibe)," *Epilepsia*, vol. 46, pp. 470–472, 2005.

[2] Epilepsy. World Health Organization, [Online]. Available: http://www.who.int/mediacentre/factsheets/fs999/en/index.html.

[3] MSP430 Ultra-Low-Power Microcontrollers, Texas Instruments, [Online]. Available: http://focus.ti.com/lit/sg/slab034r/slab034r.pdf.

[4] TMS320C54x DSP Reference Set, Texas Instruments, [Online]. Available: http://focus.ti.com/lit/ug/spru131g/spru131g.pdf.

[5] SHARC Processor Architectural Overview, Analog Devices, [Online]. Available: http://www.analog.com/en/embedded-processing-dsp/sharc/processors/index.html.

[6] A. Alkan, E. Koklukaya, and A. Subasi, "Automatic seizure detection in EEG using logistic regression and artificial neural network," *Journal of Neuroscience Methods*, vol. 148, no. 2, pp. 167 – 176, 2005.

[7] Y. Yuan, Y. Li, D. Yu; D.P. Mandic, "Automated detection of epileptic seizure using artificial neural network," *Bioinformatics and Biomedical Engineering, 2008. ICBBE 2008. The 2nd International Conference on,*, pp.1959-1962, 16-18 May 2008

[8] A. Subasi and E. Erc¸elebi, "Classification of EEG signals using neural network and logistic regression," *Computer Methods and Programs in Biomedicine*, vol 78, pp. 87–99, May 2005.

[9] L.M. Patnaik and O.K. Manyam, "Epileptic EEG detection using neural networks and post-classification", *Computer Methods and Programs in Biomedicine*, vol 91, no. 2, pp. 100-109, August 2008.

[10] E. D. Ubeyli, "Statistics over features: EEG signals analysis," *Computers in Biology and Medicine*, vol 39, no. 8, pp. 733-741, August 2009.

[11] V. Srinivasan, C. Eswaran, and N. Sriraam, "Approximate entropy-based epileptic EEG detection using artificial neural networks," *Information Technology in Biomedicine, IEEE Transactions on* , vol.11, no.3, pp.288-295, May 2007.

[12] V. Srinivasan, C. Eswaran, and N. Sriraam, "Artificial Neural Network Based Epileptic Detection Using Time-Domain and Frequency-Domain Features," *Journal of Medical Systems*, vol 29, no. 6, December 2005.

[13] S.P. Kumar, N. Sriraam, P.G. Benakop, and B.C. Jinaga, "Entropies based detection of epileptic seizures with artificial neural network classifiers", *Expert Systems with Applications*, vol 37, no. 4, pp. 3284-3291, April 2010.

[14] Hamid R. Mohseni, A. Maghsoudi and Mohammad B. Shamsollahi, "Seizure Detection in EEG signals: A Comparison of Different Approaches" in Proceedings of 28th Annual IEEE International Conference on Engineering in Medicine and Biology Society, August-September 2006, pp: 6724-6727.

[15] K. Patel, C. Chua, S. Fau, and C.J. Bleakley, "Low power real-time seizure detection for ambulatory EEG," *Pervasive Computing Technologies for Healthcare, 2009. PervasiveHealth 2009. 3rd International Conference on* , pp.1-7, 1-3 April 2009.

[16] S. Raghunathan, M.P. Ward, K. Roy, and P.P. Irazoqui, "A low-power implantable event-based seizure detection algorithm," *Neural Engineering, 2009. NER '09. 4th International IEEE/EMBS Conference on* , pp.151-154, April 29 2009-May 2 2009.

[17] S. Raghunathan, S.K. Gupta, M.P. Ward, R.M. Worth, K. Roy, and P.P. Irazoqui, "The design and hardware implementation of a low-power real-time seizure detection algorithm," *Journal of Neural Engineering*, vol 6, no. 5, 2009.

[18] M.T. Salam, M. Sawan, A. Hamoui, and D. K. Nguyen, "Low-power CMOS-based epileptic seizure onset detector," *Circuits and Systems and TAISA Conference, 2009. NEWCAS-TAISA '09. Joint IEEE North-East Workshop on* , pp.1-4, June 28 2009-July 1 2009.

[19] N. Verma, A. Shoeb, J. Bohorquez, J. Dawson, J. Guttag, and A. P. Chandrakasan, "A micro-power EEG acquisition SoC with integrated feature extraction processor for a chronic seizure detection system," *Solid-State Circuits, IEEE Journal of* , vol.45, no.4, pp.804-816, April 2010.

[20] E. Niedermeyer, F.L.D. Silva, *Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*, Philadelphia: Lippincott Williams & Wilkins, 2004, page 140.

[21] R. G. Andrzejak, K. Lehnertz, F. Mormann, C. Rieke, P. David, and C. E. Elger, "Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state," Phys. Rev. E, Stat. Nonlinear SoftMatter Phys., vol. 64, pp. 061907-1–061907-8, 2001.

[22] J. Richman and J. R. Moorman, "Physiological time-series analysis using approximate entropy and sample entropy," American *Journal of Physiology - Heart and Circulatory Physiology*, vol. 278, pp. H2039-H2049, June 2000.

[23] N. Lopes, and B. Ribeiro, "An efficient gradient-based learning algorithm applied to neural networks with selective actuation neurons," *Neural, Parallel & Scientific Computations*, vol 11, no. 3, pp. 253–272, September 2003.

[24] D. Graupe, Principles of Artificial Neural Networks—Advanced Series on Circuits and Systems, Vol. 3, Singapore: World Scientific Publishing, 1997, pp. 24-26.

[25] MATLAB version 7.9. Natick, Massachusetts: The MathWorks Inc., 2009.

[26] D. K. Lake, J. R. Moorman, and C. Hanqing, Sample Entropy estimation using sampen, [Online]. Available: http://physionet.incor.usp.br/physiotools/sampen/