Implementation of an automatic gain control for audio signals in an application for environmental protection

Erick Salas  
Electronics Engineering School  
Tecnológico de Costa Rica  
E-Mail: ericksc@gmail.com

Pablo Alvarado  
Electronics Engineering School  
Tecnológico de Costa Rica  
E-Mail: palvarado@itcr.ac.cr

Abstract—The protection of forest areas is a complex task due to the limited financial support available, especially under consideration of the forest extensions to be guarded. Hunting, illegal logging and forest fires are activities that require immediate detection in order to enable corrective actions, but the number of forest rangers available are usually insufficient to provide the required surveillance. Acoustic events produced by chainsaws and guns provide a promising mean to detect the occurrence of those illegal activities, if their detection is integrated in the nodes of a Wireless Sensor Network (WSN) spread over the area to be guarded.

The present work concentrates on the acoustic detection component in a node of a WSN, particularly on a digital implementation of the automatic gain control (AGC) system in charge of keeping a signal with a relatively constant output power level in order to simplify the subsequent pattern recognition tasks. A digital implementation of the AGC on a Spartan 3E FPGA is proposed, which consumes less than 5% of the available hardware resources. Energy consumption and area are the factors minimized in the design to comply with the conditions imposed by the forest protection application.

Index Terms—automatic gain control, wireless sensor network, pattern recognition, FPGA, forest protection

I. INTRODUCTION

The environmental costs associated with the destruction of forests —including, among other things, the destruction of ecosystems, the impact on air quality, increased susceptibility to erosion, and the effects on the dioxide carbon cycle—are nowadays quantifiable. For instance, in Latin America payments for environmental services [1] and transferable carbon trading certificates [2] have been implemented. Illegal activities of hunting and logging on the one hand, and wildfires caused by humans or by natural causes, on the other, destroy forests and their associated ecosystems, which has considerable environmental, social and economical consequences.

One fundamental component in the mitigation of the above mentioned activities is the ability to detect an abnormal event at the precise site and moment it occurs. Only under consideration of this information decision makers are enabled to trigger counteracting actions.

In tropical forests the responsibility for surveillance of thousands of hectares falls upon a few park rangers, with highly limited resources, thus hindering an effective area protection.

Acoustic events produced by chainsaws and guns provide a promising mean to detect the occurrence of illegal logging and hunting, if the detection is integrated in the nodes of a Wireless Sensor Network (WSN) spread over the area to be guarded.

A WSN is comprised by a set nodes with limited computational power, equipped with sensing and wireless ad-hoc networking capabilities [3]. These sensor nodes are deployed over the region under study, where each node is responsible for capturing from its immediate surroundings data such as humidity, temperature, air pressure, irradiation, etc. This data is locally processed in the node and send directly or in a multihop approach to a sink-node, from which end-users receive and manipulate the captured data.

The acoustic detection of gunshots and chainsaw motors on a WSN node has to cope with the signal variability caused, on the one hand, by the diversity of types and conditions of the sound sources (e.g. pistols, rifles, revolvers), and on the other hand on the particular behavior of the forest as filter, which alters the acoustic signal from the source in unpredictable ways on its path to the node.

In [4] an architecture for the acoustic pattern recognition subsystem has been proposed, which signalizes to the processing unit of a WSN node the occurrence of a particular event. The final goal is to implement this subsystem in a low-power ASIC. This allows to set the rest of the node into a low-consumption state until otherwise signalized by the acoustic detector.

A microphone is used to capture the sound on the acquisition block. The signal average power is normalized by an automatic gain control module (AGC), before being subsequently decomposed into a set of spectral bands or channels by the filter bank. The power level of each channel is estimated for a fixed time interval. The output of all estimators is considered as an eight-dimensional vector output, which is reduced in dimensionality by a linear projection and transformed to a sequence of discrete symbols. Then, the symbol sequences are analyzed by three Hidden Markov Models (HMM) to estimate the probability for each sequence to correspond to...
A detailed discussion of the FPGA-based implementation of this recognition system is presented in [5].

Power level variations of the acoustic signals arriving to the node’s microphone are considerable during each day and during the seasons of the year due to the changes on the presence of rain, winds, animals and other factors. This level needs to be normalized by the AGC module to establish a common reference for the pattern recognition stages.

Figure 2 shows a block diagram of a theoretical feed-forward AGC system. The power or amplitude of the input signal is first computed, and compared to a desired constant reference to produce an error measure, from which a new gain value is estimated to scale the delayed input signal [6]. The present work simplifies this model to make it suitable for the WSN application, in its energy and area constraints.

Even though an analog implementation of the AGC is best suited for the final ASIC implementation, in the current prototype a digital version is required to be integrated in the FPGA-based system. In any case, the digital version of the power estimation block is also required for the Feature Extraction subsystem, where the energy levels of the outputs of a digital filter bank need to be computed.

This article is structured as follows. The next section reviews previous work. The elements of the proposed AGC structure are introduced in section III, followed by FPGA implementation details on section IV. Results obtained with the system are outlined in section V. Section VI summarizes the conclusions of this work.

II. PREVIOUS WORK

The broad application range for normalization of signal levels has resulted in an equally broad spectrum of publications, where a plethora of digital and analog AGC circuits have been proposed to optimize speed, accuracy, power consumption, efficiency, spectral shaping, etc. For instance, in [6] different structures of feed forward (FFAGC) and feed backward (FBAGC) AGC are introduced. FFAGC and FBAGC are systems with a power estimation block, an error detector, a delay module and an amplifier, but differ in whether they compute the gain adjustments from the input (as in our case) or from the adjusted output (building a feedback loop).

In [7] an AGC for an GNSS RF receiver is proposed, which includes a power detector, a special decoder, an ADC and a programmable gain amplifier which is digitally controlled in an AGC loop.

A mixed signal implementation with the digital component synthesized on an FPGA was introduced in [8]. That proposal included a peak-detector and a floating point data acquisition system in an electro-chemical measurement application.

With the forest monitoring application at hand, it is of interest to considerably reduce area and power consumption, while the found systems all focused more or less on accuracy.

III. DIGITAL AGC ARCHITECTURE

The Automatic Gain Control (AGC) proposed on this work is given by

\[ \hat{x}_{\text{nor}}[n] = \frac{x[n]}{\bar{x}[n] + k_{\text{nor}}} \]

where \( \hat{x}_{\text{nor}}[n] \) is the normalized sample, \( \bar{x}[n] \) is the input power estimation for the \( n \)-th sample of the input \( x[n] \). The positive constant \( k_{\text{nor}} \) avoids division by zero and controls the position of the inflexion point in the normalization mapping. A block diagram for this system is depicted in figure 3; the similarities with the theoretical feed forward AGC of figure 2 are clear.

On the AGC loop an approximation of the signal power level is estimated in such a way that multipliers are avoided.
For this task a non-linear signal envelope detector is used, given by
\[
\bar{x}[n] = \begin{cases} 
\bar{x}[n-1] + \Delta_a & \text{for } |x[n]| < \bar{x}[n-1] + \Delta_a \\
\bar{x}[n-1] - \Delta_d & \text{for } |x[n]| > \bar{x}[n-1] - \Delta_d \\
|x[n]| & \text{otherwise}
\end{cases}
\]

where the constants \(\Delta_a\) and \(\Delta_d\) limit how fast the computed envelope is allowed to ascend or descend, respectively. These restrictions on the speed of signal change is consistent with a low-pass behavior required in the signal normalization to enhance abrupt changes like the ones occurring in a gunshot event, but also suppress slow-speed changes like produce buy changes in the biological and environmental changes during the day. The low-pass behavior also renders the use of a delay module unnecessary.

The use of an constant addition schema (linear charge/discharge) over the common constant factors used in linear systems (exponential charge/discharge) was chosen to eliminate the use of multipliers. Furthermore, the addition of constants can be optimally implemented in reconfigurable hardware.

The constant \(k_{\text{nor}}\) is added before entering the fixed point division unit. The given \textit{norm} controls how many bits will be used to represent the integer part of the results, and hence
\[
\hat{x}_{\text{nor}}[n] = x[n] \left( \frac{2^{b-i}}{\bar{x}[n] + k_{\text{nor}}} \right)
\]

for \(a\) bit fixed point number representation with \(i\) bits on its integer part and \(b - i\) bits on the fractional part, and thus \textit{norm} = \(2^b - i\). It has to be noted that since all three terms (\textit{norm}, \(\bar{x}[n]\) and \(k_{\text{nor}}\)) are positive, the term enclosed in parenthesis in (1) is also positive. However, the signal \(x[n]\) is signed and so it is the normalized signal \(\hat{x}_{\text{nor}}[n]\).

The hardware area and speed for the division module is improved by using a look-up table (LUT) instead of the sequential circuitry necessary to perform a generic division.

The saturation module cuts off values out of range assigning them to the corresponding maximal or minimal representable values. It follows the equation
\[
x_{\text{nor}}[n] = \begin{cases} 
2^{b-1} - 1 & \text{for } \hat{x}_{\text{nor}}[n] > 2^{b-1} - 1 \\
-2^{b-1} & \text{for } \hat{x}_{\text{nor}}[n] < -2^{b-1} \\
\hat{x}_{\text{nor}}[n] & \text{otherwise}
\end{cases}
\]

where the maximal and minimal values are limited by the range representable with two-complement integer numbers with \(b\) bits.

IV. FPGA IMPLEMENTATION

Table I presents the hardware resources utilized by the implementation of the proposed AGC in an FPGA Spartan 3E from Xilinx.

<table>
<thead>
<tr>
<th>Module</th>
<th>Slice</th>
<th>Flip Flop</th>
<th>LUT</th>
<th>Mult.18×18</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average estimation</td>
<td>72</td>
<td>16</td>
<td>134</td>
<td>0</td>
</tr>
<tr>
<td>Divider</td>
<td>124</td>
<td>0</td>
<td>218</td>
<td>0</td>
</tr>
<tr>
<td>Standard Multiplier</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Saturation</td>
<td>15</td>
<td>0</td>
<td>26</td>
<td>0</td>
</tr>
<tr>
<td><strong>Total usage (%)</strong></td>
<td><strong>4</strong></td>
<td><strong>0.204</strong></td>
<td><strong>283</strong></td>
<td><strong>1</strong></td>
</tr>
</tbody>
</table>

The division module requires most slices within the AGC, for which a total of 4% of the available FPGA resources are required. Sixteen flip-flops are needed to store the previous average sample computation, which is 16-bit long. In total 4% of the available slices are used in the AGC. The use of a ROM-based division explains the large number of LUT employed for this module. The AGC requires 4% of all available LUT in the Spartan 3E. Only one standard multiplier is used to apply the computed gain factor to the signal, leaving other 19 multipliers free for the rest of the acoustic detector.

The implemented AGC uses a sampling frequency of 44.1 kHz with \(b = 16\) bits word length.

V. RESULTS

The proposed implementation of the AGC module has been empirically evaluated. For this task, the acquisition module of the whole recognition system (figure 1) is used, which includes an analog input for the acoustic signal and a ADC. An analog rectangular pulse is fed into the acquisition module, whose output depicted in figure 4a) is obtained after the usual anti-aliasing LP-filter followed by the ADC. The envelope follower used as power estimator produces the signal shown in figure 4b), where a \(\Delta_a = 16\) and a \(\Delta_d = 1\). This values allow a rapid increment of the output and a 16 times slower decrement rate. Both signals in figure 4 were captured with
an oscilloscope from the analog output of a DAC connected to the digital outputs of the submodules of the AGC.

An additional example to show the operation of the envelope follower is given in the figure 5. The differences in amplitude are caused by the voltage reference used in the DAC.

Figure 6a) shows the first 40 ms of a shotgun audio sample. Previous to the gun shot (the first 8 ms in the figure) background noise is present. Figure 6b) shows the output of the complete AGC system. On the first 8 ms it is clear that the AGC has amplified the background noise to a reference level. When the gun shot occurs, the amplification factor cannot abruptly change, since the growth of the envelop is limited by $\Delta a$. The result is an over-amplified signal that is saturated on the output from $t \approx 8.5$ ms to $t \approx 10$ ms. As the time goes by, the envelope follows the signal growing fast at first, what produces smaller amplification factors, and then start to decrease as the gun shot attenuates, producing a growing amplification factor. Note that on the last 10 ms the signal varies on the same range as the first samples corresponding to background noise.

VI. CONCLUSIONS

The proposed digital feed-forward AGC architecture reduces the use of standard multipliers by employing an envelope follower as power estimation, and by using a LUT-based division implementation. Around 5% of the available resources of the Spartan 3E FPGA were required to implement the complete AGC system, which leaves enough area to incorporate the rest of the shotgun and chainsaw pattern recognition system into the same device.

Even though an analog AGC component is better suited for the final ASIC implementation of the recognition system, the proposed digital version allows to integrate it into an FPGA-based prototype which is flexible enough to be integrated into field tests.

ACKNOWLEDGEMENT

This work has been part of the 540213601701 funded by ITCR and by CYTED’s D2ARS project, UNESCO code: 120325;330417;120314;120305.

We want to extend our gratitude to M.Sc. Néstor Hernández for his collaboration in this project.

REFERENCES