ABSTRACT
Design of a cost effective and accurate positioning of a wireless device is one of the challenging issues in Wireless Local Area Networks (WLANs). In this paper we present a novel positioning system by utilizing WLAN received signal strength measurements. The technique includes two parts; First, distance estimation using received signal strength indication (RSSI) using polynomial fitting method. Second, simplified and accurate geometric location algorithm (GLA) is proposed. The proposed technique is less complex and easy to implement. The technique not requires any extra hardware and offline training.

This location algorithm is designed and tested in MATLAB tool and described using Very high speed integrated circuit Hardware Description Language (VHDL), synthesized in XILINX ISE 10.1 and simulated in ISE simulator. Novelty of the HDL description is that, IEEE754 floating point representation is used, which increases the accuracy.

Keywords
RSSI, WLAN, GLA, Positioning, VHDL, IEEE-754

1. INTRODUCTION
Positioning of objects and humans in both indoor and outdoor environment has become a more challenging research topic now a day, due to the improvements in the wireless communication technology. Global positioning system (GPS) is the most suitable for outdoor environment [1], as it is efficient, low cost and provides scalable positioning service. However using GPS scheme the signals received from the satellites are not efficient to calculate the indoor positioning of the objects.

Various accepted distance estimation methods in use are time of arrival (TOA) [4], time difference of arrival (TDOA) [5], angle of arrival (AOA) [6], and received signal strength indicator (RSSI) [14]. The synchronization requirement between transmitter and receiving nodes in TOA and TDOA, direction of radio waves calculation in AOA requires extra hardware. However, RSSI based mathematical models are less complex.

In this paper, first, RSSI based polynomial fitting algorithm [3] is used to calculate the distance between the base station (BS) and the mobile station (MS) and, second, geometric location algorithm [2] is used to locate the object. Calculation of this algorithm is done at the MS (mobile-based schemes). MS is equipped with battery. Conventional algorithm uses lot of computations hence energy. But, this geometric location algorithm reduces the computations carried out at the MS to reduce the power consumption. And also it is showed that this algorithm is accurate [2]. The main objective of our approach is to improve positioning accuracy and simplified calculations at the MS in the positioning process.

In this paper, section II describes related work. Section III describes the methodology of proposed design which includes polynomial fitting algorithm and geometric location technique. In section IV how proposed design is designed in IEEE-754 floating point representation is given. Finally results and conclusions are given in section V and VI respectively.

2. RELATED WORK
Various positioning estimation methods have been proposed to find the location of the MS using various parameters such as time of arrival (TOA) [4], time difference of arrival (TDOA) [5], angle of arrival (AOA) [6] and received signal strength (RSS) [14].

RSSI based distance estimation methods are becoming more popular because of simple and low cost. Geometric location algorithms are simple and accurate. There are many propagation models to describe indoor positioning system [7]. Okumura model is applicable in large urban area but it respond slowly to the sudden changes in the terrain [8]. Hata model [9] uses graphical path loss data obtained by Okumura. It works well in the current cellular system but not suits indoor environment. Extension to the Hata model is COST231 model [7] : it is suitable for urban, suburban and rural area. But diffractions from rooftops and buildings are not considered by this model. Egli’s propagation model is simple and it assumes terrain with hill of approximately 50 feet height. But this is limited to flat surfaces [10]. In this paper we used polynomial fitting method to estimate the distance between AP and MS.

Positioning calculation can be done either at MS or BS. But positioning in BS involves high cost and low accuracy [11]. Hence, we go for MS based positioning. In this paper we used geometric location algorithm to estimate the position of MS because minimum instructions are used, these instructions are simple and can be easily executed at MS. Main motivation for this paper is low cost, simple and high accuracy.

3. METHODOLOGY
3.1 Design flow
Flow chart of the proposed localization algorithm is shown in Fig.1
Receive signal from each AP and record its signal strength (PR1, PR2, and PR3)

Convert RSSI from each AP to distance using polynomial fitting (d1, d2, d3)

Calculate location of MS using geometric positioning algorithm (x, y)

3.2 Geometric location algorithm (GLA)

GLA used here requires three APs. Consider that range circle of all three APs intersect at single point as shown in Fig.2

Where,
D, The distance between any two APs
A0, B0 and C0 are the orthogonal projections of the MS on (AP1, AP2), (AP2, AP3) and (AP3, AP1) respectively
d1, d2 and d3 are the distances between the MS and AP1, AP2 and AP3 respectively calculated by the RSSI received from each APs which is given in next section.

Some assumptions are made in the GLA [2]. First, All three APs are placed at each corner of the equilateral triangle. Hence, distances between any two APs are same. Second, coordinates of APs are initially known to MS. Third, Now MS can be placed anywhere inside the triangle.

Distance between AP1AP2, AP2AP3and AP3AP1 are calculated as below

\[ D_1 = \sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2} \]  
\[ D_2 = \sqrt{(X_2 - X_3)^2 + (Y_2 - Y_3)^2} \]  
\[ D_3 = \sqrt{(X_3 - X_1)^2 + (Y_3 - Y_1)^2} \]  

Distance between AP1 and orthogonal projection of MS on AP1AP2 is \( r_1 \) and it is given in [12] as,

\[ r_1 = \frac{D_1^2 + D_2^2 - d_1^2}{2D_1} \]  

And one intermediate factor \( q_1 \) is defined as

\[ q_1 = \frac{r_1}{D_1} = \frac{D_1^2 + D_2^2 - d_1^2}{2D_1^2} \]  

Using this \( q_1 \), coordinate of point A0 i.e., \((x_{12}, y_{12})\) are given by [12] as

\[ x_{12} = q_2X_2 + (1 - q_1)X_1 \]  
\[ y_{12} = q_2Y_2 + (1 - q_1)Y_1 \]  

Where \((X_1, Y_1)\) and \((X_2, Y_2)\) are coordinates of AP1 and AP2 respectively. Similarly \((x_{23}, y_{23})\) and \((x_{31}, y_{31})\) are calculated as shown in [12]

Now new virtual APs are placed at A0, B0 and C0. Distance between MS and these A0, B0 and C0 are calculated using Pythagoras formula as shown below

\[ d_1(\text{MS}, A_0) = \sqrt{d_1^2 - r_1^2} \]  
\[ d_2(\text{MS}, B_0) = \sqrt{d_2^2 - r_2^2} \]  
\[ d_3(\text{MS}, C_0) = \sqrt{d_3^2 - r_3^2} \]  

3.3 Algorithm

Step 1. Calculate \(d_1\), \(d_2\) and \(D_3\) using (1), (2) & (3)

Note: In first iteration \(D_1=D_2=D_3\)

Step 2. Calculate \(r_1\), \(r_2\) and \(r_3\) using (4). Using (5) calculate \(q_1\), \(q_2\) & \(q_3\)

Step 3. Using these \(q_1\), \(q_2\) & \(q_3\) and \((X_1, Y_1)\), \((X_2, Y_2)\) & \((X_3, Y_3)\) calculate \((x_{12}, y_{12})\), \((x_{23}, y_{23})\) & \((x_{31}, y_{31})\) as given in (6)

Step 4. Update \(d_1\), \(d_2\) and \(d_3\) using (7)

Step 5. Update \((x_{12}, y_{12})\), \((x_{23}, y_{23})\) & \((x_{31}, y_{31})\) as \((X_1, Y_1)\), \((X_2, Y_2)\) & \((X_3, Y_3)\)

Step 6. Repeat step 1 using the updated values obtained in step 4 and step 5 till \(x_{12}=x_{23}=x_{31}\) and \(y_{12}=y_{23}=y_{31}\)

Step 7. Calculate \((X, Y)\) using below formulas

\[ X = \frac{x_{12} + x_{23} + x_{31}}{3} \]  
\[ Y = \frac{y_{12} + y_{23} + y_{31}}{3} \]  

3.4 Polynomial fitting algorithm

RSSI values received from each AP are converted into distance using this algorithm. The signal propagation model which is most used is log-normal shadowing model [14]:

\[ P(d)_{dBm}=P(d_{0})_{dBm}-10\log\left(\frac{d}{d_0}\right)+X_0 \]

Where, \(P(d_0)\) is the path loss at distance of \(d_0\) meters. \(d_0\) value is assumed to be 1m, and \(n\) is the path loss exponent. \(X_0\) is the RSSI measurement noise. This model cannot characterize RSSI data and distance relationship [13]
Because of the different indoor environment, to fit various situations various indoor wireless signal transmission models are adopted, a set of some indoor data is measured in the paper and corresponding wireless transmission model is found which improve the precision of the following localization calculation.

Polynomial fitting can be defined as [3]

\[ d = f(P) = \alpha_0 + \alpha_1 P + \alpha_2 P^2 + \ldots + \alpha_n P^n + \omega \]  \hspace{1cm} (8)

Where, \( P \) is the received signal strength, \( d \) is the distance between the transmitter and receiver, \( \alpha = [\alpha_0, \alpha_1, \ldots, \alpha_n]^T \) is the fitting coefficient. \( n \) is the polynomial fitting order which is taken as 3 in our paper. Its calculation is given in [3].

The ith measurement is for \( i = 1 \ldots n \):

\[ d_i = f(P_i) = \alpha_0 + \alpha_1 P_i + \alpha_2 P_i^2 + \ldots + \alpha_n P_i^n + \omega_i \]  \hspace{1cm} (9)

Least square estimation of \( \hat{\alpha} \) with least errors is as follows

\[ \hat{\alpha} = (P^T P)^{-1} P^T D \]  \hspace{1cm} (10)

Where,

\[
P = \begin{bmatrix}
1 & P_1 & \cdots & P_1^n \\
1 & P_2 & \cdots & P_2^n \\
\vdots & \vdots & \ddots & \vdots \\
1 & P_n & \cdots & P_n^n 
\end{bmatrix},
\]

\[
D = [d_1, d_2, \ldots, d_n]^T.
\]

![Figure 3. Measured distance error versus distance](image)

Accuracy and error using this polynomial fitting algorithm is given in Fig.3 which is less than 0.5 m

4. VHDL DESIGN

The proposed design is described in VHDL language using IEEE-754 floating point representation. IEEE-754 representation is used to represent the floating point number in binary format. There are three formats

1) Half precision (16 bit)
2) Single precision (32 bit)
3) Double precision (64 bit)

In our design we used single precision format because of its high accuracy compared to half precision and less hardware is required than double precision. Sign bit, number of exponent bits, mantissa bits, precision are given in [16].

5. RESULTS

Experiments are carried out to take RSSI measurements using d-link Wi-Fi adapter and NetStumbler software [15].

Table 1 shows the results of the GLA using empirical values of \( d_1, d_2 \) and \( d_3 \). The error obtained using empirical values of \( d_1, d_2 \) and \( d_3 \) is 0.062 m.

<table>
<thead>
<tr>
<th>( X_{\text{actual}} ) (meter)</th>
<th>( Y_{\text{actual}} ) (meter)</th>
<th>( d_1_{\text{actual}} ) (meter)</th>
<th>( d_2_{\text{actual}} ) (meter)</th>
<th>( d_3_{\text{actual}} ) (meter)</th>
<th>( X_{\text{estimated}} ) (meter)</th>
<th>( Y_{\text{estimated}} ) (meter)</th>
<th>Error (meter)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>2.23</td>
<td>8.06</td>
<td>8.26</td>
<td>2.00</td>
<td>0.91</td>
<td>0.09</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3.61</td>
<td>7.28</td>
<td>6.99</td>
<td>3.00</td>
<td>1.92</td>
<td>0.08</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>5.00</td>
<td>6.70</td>
<td>5.78</td>
<td>4.01</td>
<td>2.94</td>
<td>0.07</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>5.65</td>
<td>7.21</td>
<td>4.80</td>
<td>3.99</td>
<td>3.95</td>
<td>0.06</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>7.07</td>
<td>7.07</td>
<td>3.70</td>
<td>5.00</td>
<td>4.96</td>
<td>0.04</td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>8.60</td>
<td>8.60</td>
<td>1.70</td>
<td>5.00</td>
<td>6.98</td>
<td>0.02</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>8.48</td>
<td>7.20</td>
<td>2.88</td>
<td>6.00</td>
<td>5.96</td>
<td>0.04</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>7.20</td>
<td>5.65</td>
<td>4.80</td>
<td>5.99</td>
<td>3.94</td>
<td>0.07</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>7.62</td>
<td>4.24</td>
<td>6.04</td>
<td>7.00</td>
<td>2.94</td>
<td>0.06</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>8.24</td>
<td>2.83</td>
<td>7.34</td>
<td>7.99</td>
<td>1.92</td>
<td>0.09</td>
</tr>
</tbody>
</table>

Mean 0.062
Table II. Comparison of MS coordinates, (X, Y) obtained by experimental values with actual values

<table>
<thead>
<tr>
<th>X&lt;sub&gt;actual&lt;/sub&gt; (meter)</th>
<th>Y&lt;sub&gt;actual&lt;/sub&gt; (meter)</th>
<th>Using MATLAB</th>
<th>Localization Error (LE1) (meter)</th>
<th>Using VHDL</th>
<th>Localization Error (LE2) (meter)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>X&lt;sub&gt;est&lt;/sub&gt; (meter)</td>
<td>Y&lt;sub&gt;est&lt;/sub&gt; (meter)</td>
<td></td>
<td>X&lt;sub&gt;est&lt;/sub&gt; (meter)</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2.16246998</td>
<td>1.19683866</td>
<td>0.2552</td>
<td>2.16246986</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>2.49641364</td>
<td>2.13423568</td>
<td>0.2512</td>
<td>2.48765873</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>4.25721245</td>
<td>3.35509385</td>
<td>0.4385</td>
<td>4.25720453</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>3.61956190</td>
<td>4.16909936</td>
<td>0.4163</td>
<td>3.61956095</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>4.99999999</td>
<td>5.52574095</td>
<td>0.5257</td>
<td>5.00000238</td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>5.00000000</td>
<td>6.38348530</td>
<td>0.6165</td>
<td>5.00000572</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>5.50543779</td>
<td>6.13979977</td>
<td>0.5139</td>
<td>5.50534030</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>6.38043809</td>
<td>4.16909936</td>
<td>0.4163</td>
<td>6.38044357</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>7.51400680</td>
<td>2.41908622</td>
<td>0.7757</td>
<td>7.51407194</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>8.04988319</td>
<td>1.62541659</td>
<td>0.3779</td>
<td>8.04988479</td>
</tr>
</tbody>
</table>

Mean = 0.4857                Mean = 0.4866

Table 2 shows the comparison of (X, Y) coordinates values i.e., actual coordinates, values obtained by MATLAB implementation and values obtained by VHDL implementation. It also gives the localization error (LE), which is the distance between actual coordinates and estimated coordinates. LE1 and LE2 are the localization errors with respect to actual coordinates and coordinates obtained by MATLAB and VHDL. These are given by (8) and (9) respectively

\[
\text{LE1} = \sqrt{(X_{\text{act}} - X_{\text{MAT}})^2 + (Y_{\text{act}} - Y_{\text{MAT}})^2}
\]

\[
\text{LE2} = \sqrt{(X_{\text{act}} - X_{\text{VHDL}})^2 + (Y_{\text{act}} - Y_{\text{VHDL}})^2}
\]

From the table we can observe that, LE1 is 0.4857m and LE2 is 0.4866m.

5.1 MatLab simulations

Simulations are carried out by taking the APs locations as follows: AP<sub>1</sub>,AP<sub>2</sub>, AP<sub>3</sub>

AP<sub>1</sub>: (X<sub>1</sub>,Y<sub>1</sub>) = (0, 0)
AP<sub>2</sub>: (X<sub>2</sub>,Y<sub>2</sub>) = (10, 5)
AP<sub>3</sub>: (X<sub>3</sub>,Y<sub>3</sub>) = (5, 8.6)

The above simulation is carried out for 8 iterations and then we find that results converge to the actual values. Variation of the X-coordinates, x<sub>12</sub>,x<sub>23</sub> and x<sub>31</sub>, Y-coordinates y<sub>12</sub>,y<sub>23</sub> and y<sub>31</sub> and average of these values i.e., (X, Y) in every iteration are given in Fig.4, Fig.5 and Fig.6 respectively.
5.2 VHDL results

VHDL Implementation of GLA using IEEE754 32-bit single precision floating point representation is given in this section.

The proposed design is synthesized at RTL level using XILINX ISE 10.1 software targeted to the device 5vlx110tff1738 with speed grade of -2. Top module is shown in Fig.7 and RTL schematic of GLA is shown in Fig.8.

Device utilization summary of the synthesis process is given in table 3

Table III. Device utilization summary

<table>
<thead>
<tr>
<th>Device Utilization Summary (estimated values)</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Logic Utilization</td>
<td>Used</td>
<td>Available</td>
<td>Utilization</td>
</tr>
<tr>
<td>Number of Slice LUTs</td>
<td>29421</td>
<td>69120</td>
<td>42%</td>
</tr>
<tr>
<td>Number of fully used LUT-FF pairs</td>
<td>0</td>
<td>29421</td>
<td>0%</td>
</tr>
<tr>
<td>Number of bonded IOBs</td>
<td>381</td>
<td>680</td>
<td>56%</td>
</tr>
<tr>
<td>Number of DSP48Es</td>
<td>54</td>
<td>64</td>
<td>84%</td>
</tr>
</tbody>
</table>

Fig.9 shows the VHDL simulation of our positioning technique for MS coordinate values (6, 4). Expected values of (X, Y) are (6.38043809, 4.16909936), and obtained values of (X, Y) using VHDL simulation are (6.38044357, 4.16910289). Error obtained is (0.00000548, 0.00000353). Simulations are carried out using ISE simulator.

6. CONCLUSION

Proposed location algorithm provides accurate WLAN positioning technique. Real time RSSI values are used to obtain the position of the MS from set of distance estimations to the access points. We have compared localization error through numerical simulations and real experiments. Accurate results are obtained. Comparisons of the MS coordinates obtained by MATLAB and VHDL with actual coordinates are presented.

The system is tested and validated in MATLAB and is described in VHDL language using XILINX ISE10.1 tool and is simulated using ISE simulator. IEEE-754 single precision floating point representation is used, which increases the accuracy of the positioning system. The technique could be extended to WLANs in other environments such as office, auditorium and industry.

7. REFERENCES


