

# RESISTIVE TOUCHSCREEN BASED CHARACTER RECOGNITION SYSTEM

Meer Suri<sup>1</sup>, Dr. Sukhwinder Singh<sup>2</sup>

<sup>1</sup>Undergraduate Student, PEC University of Technology, Chandigarh, (India)

<sup>2</sup>Supervisor, Assistant Professor, PEC University of Technology, Chandigarh, (India)

## ABSTRACT

The project uses a 4 wire resistive touchscreen to acquire an input character from the user which is then processed in LabVIEW to compare with the database of stored characters and to display the closest match. The touchscreen produces an analog output that is converted to digital using the built in 10 bit ADC of the ATmega16 microcontroller and then sent serially to the computer equipped with LabVIEW.

**Keywords:** Character Recognition, LabVIEW

## I. INTRODUCTION

Character or Handwriting recognition is used as an input method in mobile phones, PDAs, tablet PCs etc. The project aims to recognize a character drawn on the touchscreen which sent to a computer serially by an ATmega16 microcontroller. The digital version of the input produced by the microcontroller is processed by LabVIEW which compares it with the database of stored characters using statistics and then displays the closest match.

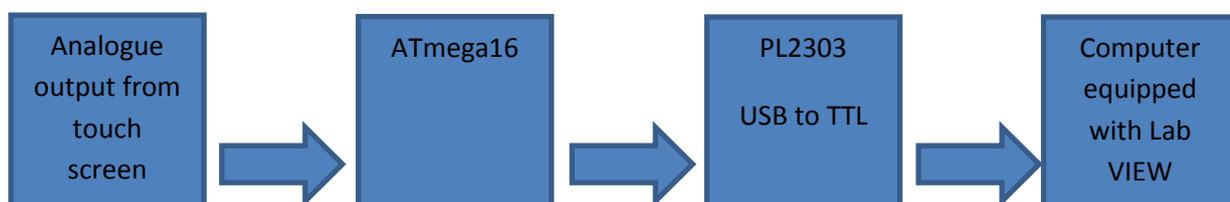
## II. RELATED WORKS

Character or Handwriting recognition software has been in use in mobile phones, PDAs, tablet PCs for a long time. The first handwriting recognition system in a PDA came in 1993 in the Apple MessagePad[1]. Since then several sophisticated algorithms have been developed that can recognize words at a time and produce accurate results. In contrast this project uses a very simple algorithm based on statistics to do the recognition.

## III. HARDWARE

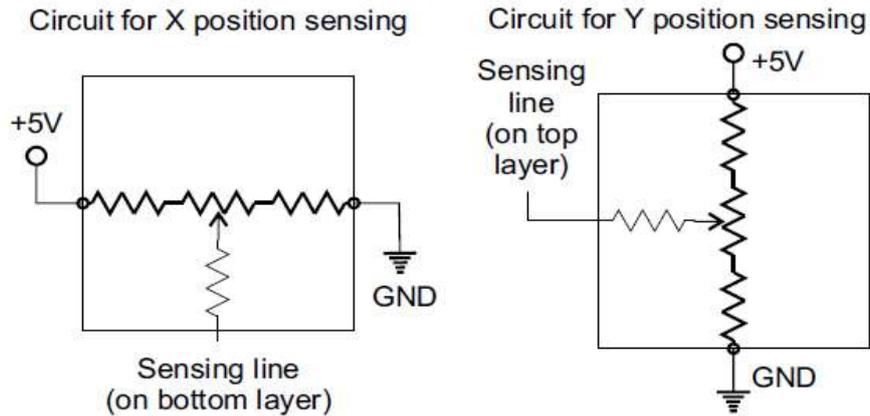
There are 4 components-

1. 4 wire analog resistive touchscreen
2. ATmega16 microcontroller
3. PL2303 USB to TTL converter
4. Computer equipped with LabVIEW



#### Resistive Touchscreen [2]

A 2- Dimensional sensing device that is constructed of 2 sheets of material separated slightly by spacers. A common construction is a sheet of glass providing a stable bottom layer and a sheet of Polyethylene (PET) as a flexible top layer. The 2 sheets are coated with a resistive substance, usually a metal compound called Indium Tin Oxide (ITO). When the PET film is pressed down, the two resistive surfaces meet. The position of a touch can be read by a touch screen microcontroller circuit.



ATmega16 microcontroller-

It has been used to acquire the input from the touchscreen and convert it to digital values using the built-in 10 bit Analog to Digital converter. It has been programmed to output digital coordinates of the touch in the format  $x_1, x_2, x_3, y_1, y_2, y_3$  using full 10 bit resolution. Each digit of the coordinates is transmitted in one byte. These coordinates are sent serially to the computer through the USB to TTL converter.

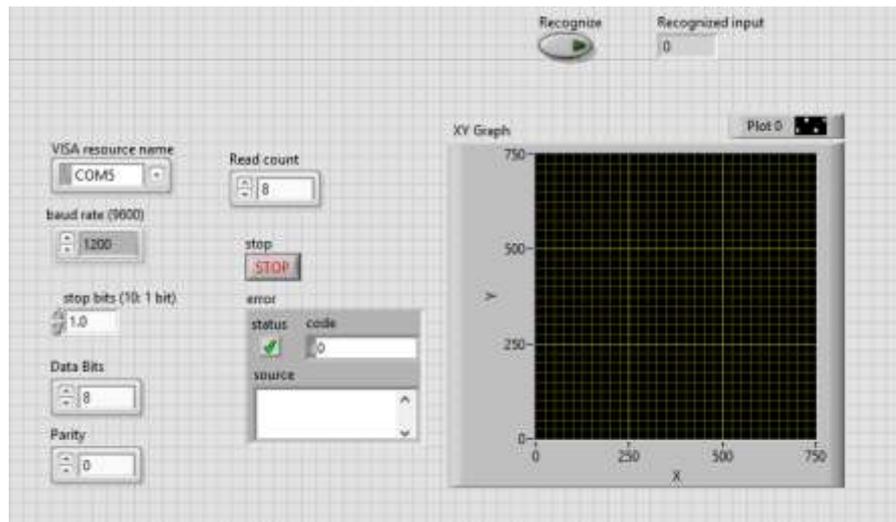
#### IV. LABVIEW

It stands for Laboratory Virtual Instrument Engineering Workbench. It was developed to allow scientists and engineers to control instruments through computers using an easy to understand graphical programming language. LabVIEW has been used to serially acquire data using VISA libraries and then to plot and compare the input character with the database of characters.

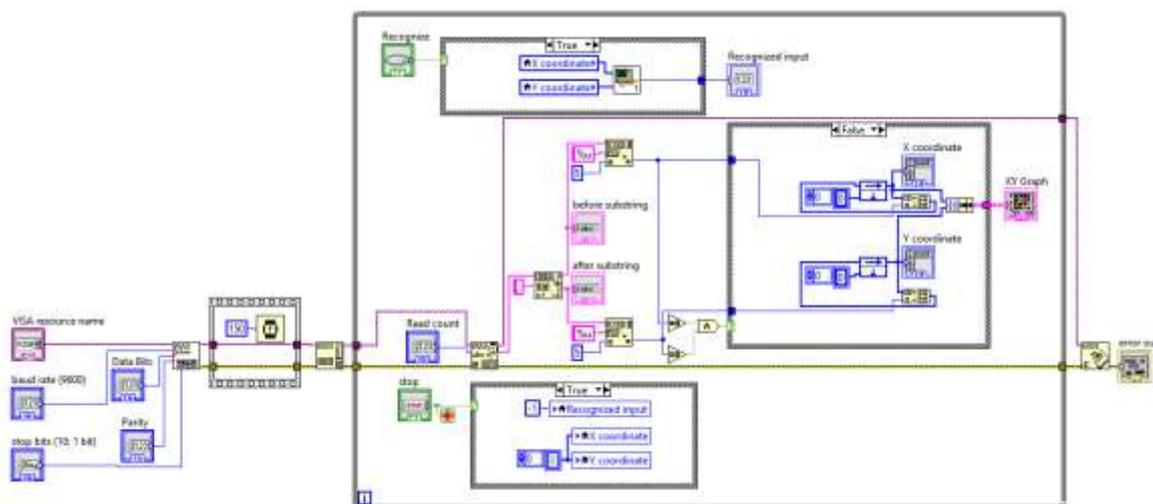
#### V. MAIN VI

The main VI configures the serial port using controls setting the appropriate data bits, stop bits, baud rate and parity (not used). After the port has been configured it uses the VISA Read function to read data in every iteration of the while loop. In every iteration it reads one set of coordinates, uses the Match Pattern function to separate the X and Y coordinates and stores them in separate arrays. These arrays are then clustered together and sent for plotting to the XY Graph. Each set of coordinates is 8 bytes, 6 for the numbers, 1 for the comma and 1 for a new line character at the end, so 8 bytes are read every iteration.

When the user has finished entering the character, he may press the Recognize button on the front panel to do the character recognition and display the recognized character.



**Figure 1 Front Panel of the Main VI**



**Figure 2 Block Diagram of the Main VI**

After the user presses the Recognize button, the Compare subVI is passed the X and Y arrays of the input character. It does the recognition and displays the output.

## VI. COMPARE SUBVI

The Compare subVI does the calculations for the recognition. It uses the Mean Absolute Deviation [3] function of Statistics for doing the comparisons. Mean Absolute Deviations of difference of input arrays and stored arrays calculated and compared. Mean Absolute Deviation of a data set tells us how far, on average, all values are from the mean value.

The mean absolute deviation of a set  $\{x_1, x_2, \dots, x_n\}$  is

$$\frac{1}{n} \sum_{i=1}^n |x_i - m(X)|.$$

$m(X)$  is the mean of the data set

It be zero for difference of 2 curves if they are identically drawn on the screen irrespective of their locations. It will be non-zero and will increase as the 2 curves being compared get more different from each other.

The subVI has access to the database of characters which consists of a text file for each character. Each text file contains the X and Y arrays of the character in the database, with which the input character's X and Y arrays will be compared. The subVI calculates the Mean Absolute Deviations of the difference of the input and stored characters and then compares these values iteratively in a Formula Node to find the character with least deviations compared to all others in the database. This is taken as the closest match and is displayed as the Recognized input on the Front Panel of the main VI

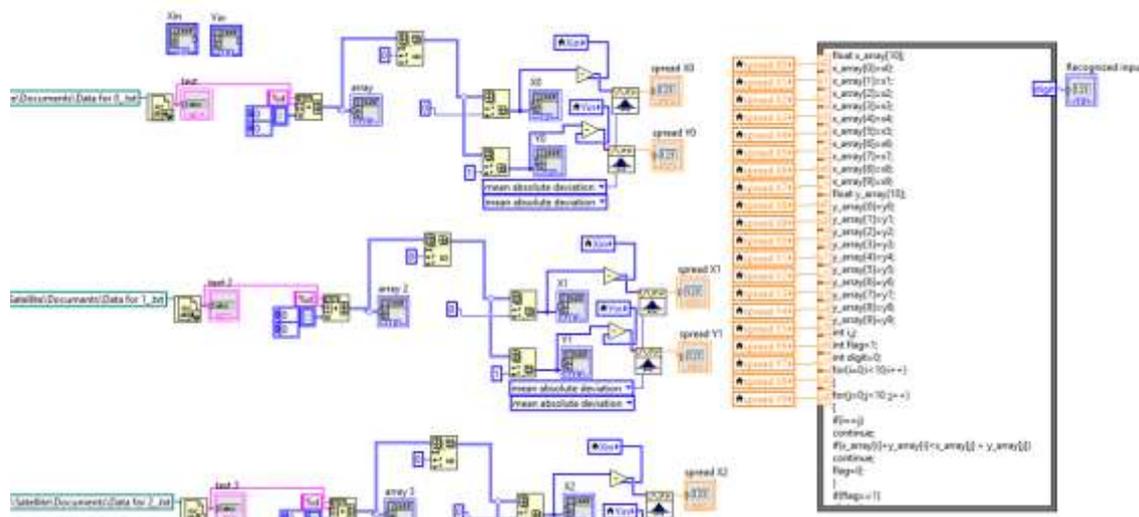


Figure 3 Block Diagram of the COMPARE subVI

## VI. CONCLUSION

The accuracy of the character recognition system depends on the following:

1. The shape of the character in comparison to shape stored in the database. We got accurate results when the input was similar to the database. It was not a big constraint as characters did not have to be exactly the same for accurate results.
2. The speed at which the character was drawn when it was stored in the database. The speed has more influence on the accuracy as even if the same shape is drawn, but at a significantly different speed from the database, the result will be inaccurate.
3. While drawing on the touch screen, some points can appear randomly on the XY graph. If too many of these points appear, the accuracy will be decreased and results may not be as expected. These points may be due to noise in the ADC.
4. The baud rate plays a very significant role in deciding the error and speed of the data transmission.

So, if the speed at which the character is drawn and its shape are similar to the character in the database, the results will be accurate most of the times.

## VII. FUTURE SCOPE

This system is accurate enough to differentiate between similar characters provided the speed at which the character is drawn and its shape are similar to the ones in the database, so it would be possible to include many more characters in the database without affecting accuracy.

A more advanced version could be made which would be able to recognize a word of writing at a time.

It could be used as pattern lock system for devices like mobile phones, PDAs, tablet PCs. A pattern to unlock the device would be stored in the database and if the input would match with that, the device would unlock.

## REFERENCES

- [1] <http://en.wikipedia.org/wiki/MessagePad>
- [2] <https://www.sparkfun.com/datasheets/LCD/HOW%20DOES%20IT%20WORK.pdf>
- [3] [http://en.wikipedia.org/wiki/Average\\_absolute\\_deviation](http://en.wikipedia.org/wiki/Average_absolute_deviation)