Make Your Own Sinhala Word Processor

By Dilum Bandara

Why do you pay thousands of Rupees when you can make your own Sinhala word processor? The truth is that most of PC users have some Sinhala fonts but do not have a word processing program. Stop searching the Character Map, now build your own one. If you have fair knowledge on Visual Basic (or any other language) and ASCII (American Standard Code for Information Interchange) with minimum effort you can make a simple Sinhala notepad.

This article concentrates only on how to receive certain character from the keyboard and interpret it to the appropriate Sinhala character, any instructions on how to save as RTF (Rich Text Format) file, how to copy certain text or change the font is not covered here.

How do Sinhala fonts work?

In Sinhala there are over 300 different characters so how can we represent all of these using ASCII (ASCII can represent only up to 256 characters) actually we can not do it. Therefore we have to use two sets of fonts (ex. aKandyNew and aKandyNewSupplement), eliminate rarely use fonts (most font creators do this) or use UNICODE characters.

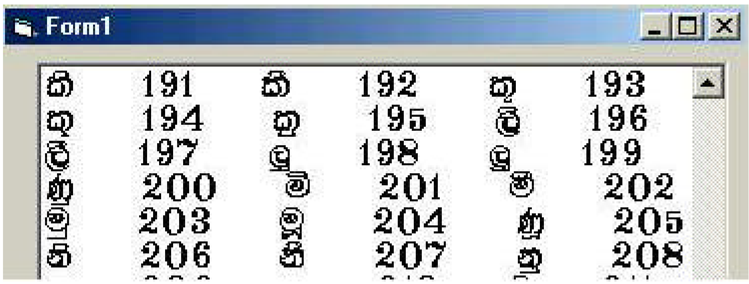
If you press a simple character which can be directly typed from the standard Win98 keyboard it'll be displayed as it is, Sinhala fonts also work on the same principle. But you may have seen that, most of the time when a modification character (like biaṣạ, mdmṣạ) is pressed it give a garbage output. But if you use a good word processor it won't happen. Word processor completely replaces the garbage output by single, better quality character which represents actual character we want.

Basic problem in all the Sinhala fonts is that they do not fallow any standard although they belong to the same keyboard layout or even created by same person. Therefore first we have to find out what are the available characters and its corresponding ASCII value. Fallowing coding illustrates how to do it.

Select a new project (standard exe) from VB and add a text box to it. Set its font name to "aKandyNew" and size to 14. You can find "aKandyNew" from the Web. Use this font for all the examples given here. Slight changes to the coding may required depending on your font. After you throw with the concepts, you can move to different fonts. Add the fallowing sub routing to it. The output would look like Fig1.

```
Private Sub Form_Load()
  Dim i As Integer
  Dim j As String
  For i = 35 To 255 'no need of starting at 0 try changing it to 0
    j = I   'cast to string
    Text1.Text = Text1.Text + Chr(i) + "   " 'convert ASCII to char
    Text1.Text = Text1.Text + j + "   "   'add space between char
  Next i
End Sub
```

Run the program and copy the output to Word or notepad, do some formatting and get a printout or you can direct output directly to the printer using `Visual Basic` **`print`** command.



```
   191      192      193
   194      195      196
   197      198      199
   200      201      202
   203      204      205
   206      207      208
```

Figure 1 – A sample output

What is a keyboard layout and how can I find it?

Layout is the arrangement of keys in a keyboard, depending on the order of keys layouts can be differ. In Sinhala many layouts are available, "Wijesekara keyboard" which is used in Typewriters is the most famous, efficient but more difficult to learn one, also you can use sound of a single character (A for W, B for n, L for , etc.) or use the sound of combination of characters (like most of us use in e-mails) or you can think of your own one. Decide what type of layout is to be implemented on your notepad. In this article I will introduce how to use the Wijesekara layout.

Here is a simple way to find out the layout of the font you select. In MS Word set font to the font you like and it size to 14 even 16 is better. Starting from the top left-hand corner of the keyboard, type the first character then press spacebar once, then type the second character again a space and repeat this way for all the characters row by row. Press the Shift key and repeat same procedure, the resulting document is your keyboard layout.

In the fallowing examples I am using the aKandyNew font, which is not designed for Wijesekara keyboard layout. But I'll show how it can be mapped for the Wijesekara one. Fig 2 illustrates a sample Wijesekara layout which is slightly different from the one you find in typewriters.

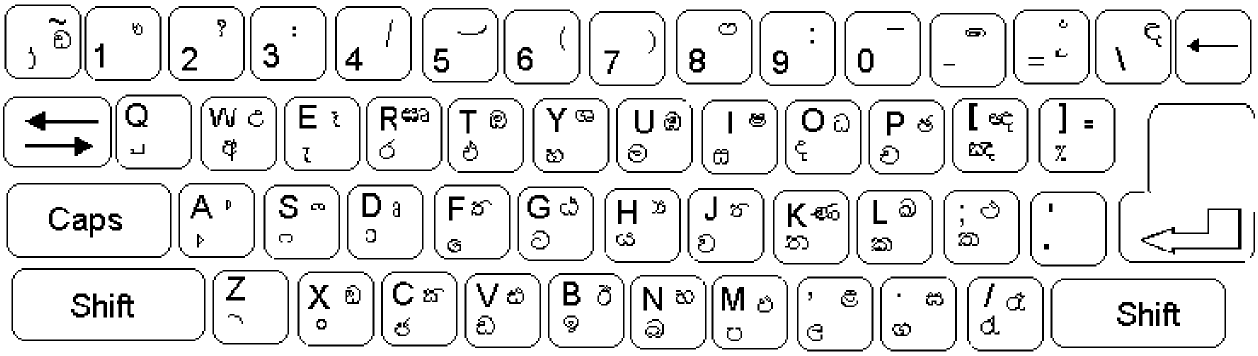


Figure 2 – The Wijesekara keyboard layout

This layout can found using a font which is specifically designed for Wijesekara layout.

Start a new VB project and select standard exe and design a simple word processor using either Text box or RichText box as the text area. Add all the functionalities that a notepad should have. I have used a RichText box control.

Double click on the Richtext box then you can see the `richtextbox1_change()` subroutine. Select KeyPress from the procedure drop down list. Fallowing subroutine can be seen. Add fallowing lines of codes to it.

If you compare the layout you get and the given one (fig2). In Wijesekara layout w is represented by letter **w** (ASCII 119) and in KandyNew font it is represented by letter **a** (97). W (W 87) in Wijesekara layout is represented by letter **u** (117) in KandyNew. So first we have to map all the normal characters. Similarly k (k 107) in Wijesekara is represented by n (110) in KandyNew. Line 1 to 14 in the fallowing coding illustrates how the interpretation is done.

```
1  Private Sub RichTextBox1_KeyPress(KeyAscii As Integer)
2  ‘KeyAscii – ASCII value of key pressed
3    Dim SelTex As String     'holds the recently selected text
4    Select Case KeyAscii     'select depending on the ASCII value
5      Case 119  ‘if the pressed key is w(ASCII 119) replace with 97
6        RichTextBox1.SelText= Chr(97) ’convert ASCII value to character
7      Case 87    ‘if the pressed key is W(ASCII 119) replace with 117
8        RichTextBox1.SelText = Chr(117)
9      Case 107    ‘if the pressed key is W(ASCII 107) replace with 110
10         RichTextBox1.SelText = Chr(110)
11    ‘continue for all the none modifying characters in the layout...
12     End Select ‘end selecting characters
13     KeyAscii = 0 ‘set the ASCII value to 0 otherwise last pressed
                  ‘character also appear
14 End Sub
```

Then how can we modify certain character? In modification we have to concern about two character, fist the last given modification character and then the previous character to be changed. We have to be sure that there is at least one character to be changed. If the cursor is not at the beginning of the text starting position is not null. When a character is typed starting position automatically increase by 1. Consider the fallowing examples.

Suppose we want type is First we type i (i 105) then we type s(biams,a ASCII 115) to change character i to is In kandyNew is is represented by a single character which has the ASCII value 226. But when we type i it is represented by ASCII values 115 (s) in KandyNew. So we have to change ASCII 115 instead of ASCII 105. Similarly in KandyNew ks is represented by a single character having ASCII value of 206. And k in Wijesekara keyboard is represented by ASCII 107 but 110 (n) in KandyNew. So remember that, before you use the characters mentioned above first you have to interpret them so they can be typed.

Let's say we need ¨. First we type , (, 44) then we type q(mdms,a ASCII 113) to change character , to ¨. In kandyNew ¨ is represented by a single character which has the ASCII value 198. But , is represented by ASCII 108 (l) in KandyNew so we have to concern only on ASCII 108. Similarly in KandyNew ¦ is represented by a single character having ASCII value of 218. Fallowing coding can perform the above tasks.

```
Private Sub RichTextBox1_KeyPress(KeyAscii As Integer)
'KeyAscii – ASCII value of key pressed
  Dim SelTex As String      'holds the selected text
  Dim SelTexVal As Byte     'holds the ASCII value of the selected text
  Select Case KeyAscii      'select depending on the ASCII value
    Case 115                'if pressed key is s (ASCII 115)
      If RichTextBox1.SelStart <> 0 Then 'any character to be change
 'reduce the start position by 1 so last character can be select
        RichTextBox1.SelStart = (RichTextBox1.SelStart - 1)
        RichTextBox1.SelLength = 1
        SelTexVal = Asc(RichTextBox1.SelText) 'character to ASCII
        If SelTexVal =  115 Then ' if the selected text is i
          RichTextBox1.SelText= Chr(226) 'ASCII value to character
        ElseIf SelTexVal =  110 Then ' if the selected text is k
          RichTextBox1.SelText= Chr(206) 'ASCII value to character
        End If
      End If
 'Continue for all the possible combinations of  s biams,a, . . .
    Case 113                'if pressed key is s (ASCII 115)
       If RichTextBox1.SelStart <> 0 Then 'any character to be change
        RichTextBox1.SelStart = (RichTextBox1.SelStart - 1)
        RichTextBox1.SelLength = 1
        SelTexVal = Asc(RichTextBox1.SelText) 'character to ASCII
        If SelTexVal =  108 Then ' if the selected text is i
          RichTextBox1.SelText= Chr(198) 'ASCII value to character
        ElseIf SelTexVal =  113 Then ' if the selected text is k
          RichTextBox1.SelText= Chr(218) 'ASCII value to character
        End If
      End If
 'Continue for all the possible combinations of  q mdms,a,  . . .
    End Select
  KeyAscii=0   'set the ASCII value to 0
End Sub
```

It can be tedious entering the possible character combinations, but to get better and correct functionality you have to do it, that is why people who know the concept are not interested in build programs. I think wasting two or three hours for coding this is worthwhile than searching the Character Map. Try to get the concept how it works and then apply it your self the way you want. This is a really good topic to do some research because we can implement the same concept using different kind of fonts and different keyboard layouts.

This is not the only way characters can be interpreted. You can use array of command buttons or use a toolbar icons. If you know how to handle Windows API functions you can make a program that can be used in all the WP (word processing) abd DTP (desk Top Publishing) packages in the windows environment.

I have selected the hard path to implement the "aKandyNew" font, and map it to the "wijesekara" layout, it can be easily done using a phonotic (using the sound of the character) layout because KandyNew font is specifically designed for it. If I do so there will be legal problems so try it your self. Full source code of my notepad is freely available at www.cse.mrt.ac.lk/~dilumba.