

(51) International Patent Classification:
G06F 3/023 (2006.01)(21) International Application Number:
PCT/CA2010/000219(22) International Filing Date:
18 February 2010 (18.02.2010)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/154,150 20 February 2009 (20.02.2009) US

(72) Inventor; and

(71) Applicant : MOHSEN, Mohamed, Madi [CA/CA]; 85
Littles Road, Toronto, Ontario M1B 5E2 (CA).(74) Agent: KHAN, Sheema; Miltons IP/p.i., Suite 700, 225
Metcalf Street, Ottawa, Ontario K2P 1P9 (CA).(81) Designated States (unless otherwise indicated, for every
kind of national protection available): AE, AG, AL, AM,
AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ,
CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO,
DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT,
HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP,KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD,
ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI,
NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD,
SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR,
TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.(84) Designated States (unless otherwise indicated, for every
kind of regional protection available): ARIPO (BW, GH,
GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM,
ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ,
TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE,
ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV,
MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, SM,
TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW,
ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))
- as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))

Published:

- with international search report (Art. 21(3))

(54) Title: KEYBOARD FOR LANGUAGES BASED ON THE ARABIC SCRIPT

Fig. 4a

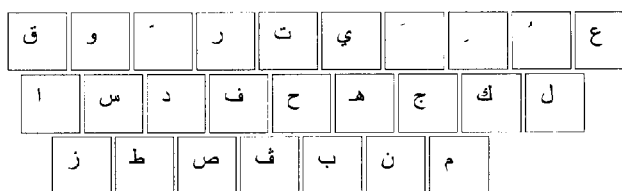


Fig. 4e



(57) Abstract: A keyboard layout for typing in any of the languages whose alphabet is based on Arabic script is designed for typists with any level of familiarity with a Latin alphabet keyboard layout. The keyboard layout reduces the user's learning curve by leveraging the user's familiarity with the Latin alphabet keyboard layout, and the user's knowledge of how alphabet characters of the language are hand-stroked and lexicographically ordered in the Arabic script-based language. For each Latin alphabet letter key, the layout assigns one or more script letters or diacritics based on the phonetic similarity between Latin alphabet characters and alphabet character of the language, sound and shape relationship among the characters, and the frequency distribution analysis of the Arabic script-based alphabet. The language is preferably Arabic, Farsi or Urdu.

KEYBOARD FOR LANGUAGES BASED ON THE ARABIC SCRIPT

FIELD OF THE INVENTION

5 The present invention relates to keyboards and methods of entering data into computerized systems for languages based on the Arabic script. More specifically, a keyboard layout design provides for each Latin alphabet key to be mapped onto one or more Arabic script-based alphabet and/or diacritical characters.

10

BACKGROUND OF THE INVENTION

For many computer users, keyboarding is a tedious process and it may take years to become comfortable with inputting text. When learning to type efficiently, users have to familiarize themselves with two particulars about the keyboard layout of interest: i) the location of every character-editing key on the keyboard, and ii) the proper hand and finger to use to signal or press a given key. With adequate keyboard typing tutoring, typists can comfortably type at rates of 50 words per minute or greater.

20 FIG. 1 shows the typical character printing keys available on most standard keyboards layouts, together with the ideal mapping of such keys to the eight character-typing fingers of a trained computer typist for maximizing touch-typing efficiency and typing throughput, together with the number of rows.

25 A standard Latin alphabet keyboard layout most typists encounter when learning to type is the English keyboard layout known as QWERTY, shown in FIG. 2, which is named after the arrangement of the first six left letter keys of the top alphabet row on this keyboard. Although many users strive to learn touch-typing, the hunt-and-peck typing style, which typically is slower than touch-typing is the style many typists adopt unless learning measures are taken at an early stage of using a keyboard. Regardless of the style, typists typically become comfortable with whatever layout they use to input text as time progresses, and a sense of where a particular key is located on a keyboard layout tends to become readily available through what is known as neuromuscular facilitation or muscle memory.

Due to the wide availability of the QWERTY keyboard layout (QKL), most multilingual computer typists who are trained on the QKL can capitalize on their knowledge when typing in another Latin alphabet language, given the similarity in keyboard layouts. As such, the learning curve remains relatively flat while the user
5 reaches personal typing speeds on the new keyboard layouts. For example, a typical QKL typist can readily start typing in other languages such as Danish, Dutch, Finnish, Swedish, Icelandic, Italian, Norwegian, Portuguese, and Spanish to name a few.

Furthermore, with minimal effort, the same typist can also type with little difficulty
10 using a similar keyboard layout known as QWERTZ, where only the positions of the letter keys Y and Z are interchanged. QWERTZ is widely used in Germany and much of Central Europe. The same knowledge transfer is applicable to other close variants of QWERTY such as AZERTY, used in France and Belgium, and QZERTY, common in Italy. Another common but less competitive keyboard layout is Dvorak, whose
15 keyboard layout is arranged such that faster typing is made easier to achieve.

For QWERTY users who need to type in non-Roman scripted alphabet languages such as Arabic, Armenian, Greek, Hebrew, Farsi (Persian), Urdu or Russian for example, most keyboard layouts are quite different from what a QKL or a Dvorak
20 keyboard layout user is accustomed to. Consequently the user must expend a significant amount of time to learn and memorize a different keyboard layout. In addition, the learning process is wrought with typing mistakes and frustration, as the user strives to learn an unfamiliar layout.

25 For example, although a QKL user typically memorizes most, or all, of the character key locations, the user is faced with the challenge of having to learn a fresh set of new key locations to type in Arabic-based script languages, such as Arabic, Farsi (Persian), Urdu, Pashto, etc. This occurs even though the majority of the letters of the Arabic-based script are phonetically similar to a corresponding letter in the Latin
30 alphabet.

For example, consider the case of Arabic where 48 characters, as shown in Table 1, are used to write Arabic script. Whereas a user of the QKL or a Dvorak keyboard layout needs to be familiar with only 26 Latin alphabet keys, a user of prior art Arabic

keyboard layouts is required to memorize at least 48 different key locations over two or more modes (i.e. Normal, Shift, and Option).

TABLE 1 - Characters of the Arabic Script

Number	Character	Number	Character	Number	Character
1	ـ	19	غ	37	ق
2	ب	20	ف	38	ك
3	ت	21	ق	39	م
4	ث	22	ك	40	ن
5	ج	23	ل	41	هـ
6	ح	24	م	42	و
7	خ	25	ن	43	ي
8	د	26	هـ	44	أ
9	ذ	27	و	45	إ
10	ر	28	ي	46	آ
11	ز	29	أ	47	ء
12	س	30	إ	48	ة
13	ش	31	آ		
14	ص	32	ء		
15	ض	33	ة		
16	ط	34	ؤ		
17	ظ	35	ى		
18	ع	36	ئ		

5

One of the most widely-used Arabic keyboard layouts, shown in FIG. 3, is the Arabic Windows™ Keyboard Layout (AWKL). As shown in Table 2, many Arabic letters have a phonetic counterpart in the Latin alphabet. Yet, there is little correlation between the location of Arabic letters on the AWKL and their phonetic counterparts on the QKL. For example, the letter T has the same sound as that of the letter ت in Arabic (Character 3 in Table 1); yet the letter T is located at key 18 of the QKL in FIG. 1, while the letter ت is found at key 32 of the AWKL as seen in FIG. 3. In fact, only the letter س in the AWKL maps onto its English equivalent in QKL, namely the letter S at key 28. Given such low phonetic correlation between keys of the QKL and those of AWKL, there is a steep learning curve for those familiar with the QKL. This leads to continuous typing errors, which in turn lead to many BACK-SPACE key hits, confusion and frustration as the typist tries to locate and type character keys on a fresh layout. As such, a considerable amount of time is required for the QKL user to become familiar with the AWKL.

20

This is further illustrated by the following example. To type the six-letter word BRAZIL using the QKL, the typist presses the keys 42, 17, 27, 38, 21, and 35 on the keyboard layout shown in Figures 1 and 2. This implies that a touch-typist would rely on their fingers' muscle-memory to have their fingers activated in this order: (4, 4), (4, 2), (1, 3), (1, 4), (6, 2), and (7, 3), where the tuple (f, r) indicates that finger f = 1, 2, ..., 8, presses a key at row r = 1, 2, 3, or 4. Now, to type the same word in Arabic based on the phonetic resemblance, the typist needs to type the following Unicode string: برازيل (Arabic scripting is strictly cursive, left to right, top to bottom). The distinct letters that make up this word are as follows: ب, ر, ا, ز, ي and ل. According to Table 2, all of the letters in this example have a one-to-one mapping to six English letter counterparts. However, a typist using the AWKL must press keys 30, 41, 32, 46, 29, and 31, as opposed to the above-mentioned sequence of the QKL. There is no match between the keys pressed on AWKL and those of QKL, even though each Arabic character has a direct phonetic match with a Latin character.

15

In addition, the AWKL user must activate the fingers' muscle-memory sequence {(4, 3), (4, 4), (5, 3), (7, 4), (3, 3) and (4, 3)}, compared with the QKL sequence {(4, 4), (4, 2), (1, 3), (1, 4), (6, 2), and (7, 3)}. As shown in this example, there is no match between sequence elements of AWKL and that of the QKL. The user is forced to relearn key locations that share the same vocal sound, and as a result, retrain the fingers' muscle-memory to access different keys on the keyboard to produce the same word. This simple example illustrates problems encountered with prior art keyboards when typing text that contain terms or phrases to be transliterated or transcribed into Arabic. Further examples include country and city names, person and/or character names, industrial and commercial product or company names, amongst others.

25

TABLE 2 - Phonetic pronunciation of letters of the Arabic alphabet

No	Letter	Name	Phonetic pronunciation in English
1	ا	<i>Alif</i>	a
2	ب	<i>Baa'</i>	b
3	ت	<i>Taa'</i>	t
4	ث	<i>Thaa'</i>	th (as in throw)
5	ج	<i>Jeem</i>	j
6	ح	<i>Haa'</i>	H (as in Mohammad)
7	خ	<i>Khaa'</i>	kh
8	د	<i>Daal</i>	d

9	ذ	<i>Thaal</i>	th (as in there)
10	ر	<i>Raa'</i>	r
11	ز	<i>Zein</i>	z
12	س	<i>Seen</i>	s
13	ش	<i>Sheen</i>	sh
14	ص	<i>Saad</i>	c (as in cyber)
15	ض	<i>Dhaad</i>	d (as in dawn ; the "d" is heavy)
16	ط	<i>Taa'</i>	t (as in tub)
17	ظ	<i>Zhaa'</i>	th (as in th yself)
18	ع	<i>Ain</i>	a (as in Abdu) or o (as in Omar)
19	غ	<i>Ghain</i>	gh (as in gh ost or the French sound of the letter r)
20	ف	<i>Faa'</i>	f
21	ق	<i>Qaaf</i>	q (as in Qura'an or quote)
22	ك	<i>Kaaf</i>	k
23	ل	<i>Laam</i>	l
24	م	<i>Meem</i>	m
25	ن	<i>Noon</i>	n
26	هـ	<i>Haa'</i>	h
27	و	<i>Waaw</i>	(as in work or win), oo (as in school) or ou (as in your)
28	ي	<i>Yaa'</i>	y (as in yin or York), ee (as in between) or ea (as in bean)

Another drawback of prior art keyboards is the complexity of keys required to produce the modified forms of certain Arabic characters. For example, an AWKL-user presses key 32 to type the Arabic letter **ا**. However, the letter **ا** has four modified forms: **أ**, **إ**, **آ** and **ء**, which are generated by AWKL keys SHIFT 32, SHIFT 19, SHIFT 43, and 39, respectively. While there is significant similarity between the letter **ا** and its four modified forms for the skilled worker, the AWKL requires the use of five dispersed key locations or access methods (i.e., through pressing the SHIFT modifier key first) for typing a similar Arabic letter. The same disadvantages of the AWKL are found for the letter **و** and its modified form **ؤ**, the letter **ى** and its modified forms **ي** and **ي**, and each diacritic and its modified forms.

The most frequently occurring ligature in Arabic, **لا**, is a two-letter string made from letters **ل** and **ا**. This ligature, located at key 42 on the AWKL, frequently occurs in the three other modified forms **لا**, **لا** and **لا**. However, these modified ligatures are located at keys SHIFT 31, SHIFT 18, and SHIFT 42 on the AWKL, which requires

the user to remember four different key locations or access methods to produce essentially the same ligature. These examples illustrated the inefficiency of the AWKL for typing modified forms. Such an approach not only over-clutters the keyboard layout, but also introduces a burden upon the typist who needs to understand
5 and memorize key locations to some degree to achieve satisfactory typing performance.

The AWKL suffers an other drawback by the location of frequently-used Arabic characters. For example, the letters د (pronounced like 'd' as in "dog") and ث (pronounced like 'th' as in "then") have both similar sound and shape. Yet, on the
10 AWKL, they are assigned to keys that lie at opposite ends of the keyboard (keys 1 and 25), and are far from the home row. . The same is true of the frequently-used diacritic symbol ˘ which is assigned to key 1. Once again, the user is required to leave the keyboard home row position in order to access frequently occurring letters or
15 symbols.

A further drawback of the AWKL keyboard is the re-arrangement of non-alphanumeric symbols such as parentheses (), brackets [], braces { }, inequality symbols < > and the forward-slash key / . These symbols are often used in software
20 development and mathematics where users have become accustomed to the location of each symbol, along with the ordering of elements within pairs. On the QKL, the aforementioned symbols are located at keys 10 and 11, 24 and 25, Shift 24 and Shift 25, 45 and 46, and 47, respectively. On the AWKL, these symbols are assigned to key locations 11 and 10, 30 and 29, 41 and 40, 24 and 25, and 35, respectively,
25 requiring the QKL user to relearn the position of these keys on the AWKL. A similar problem exists for punctuation keys.

Many of the drawbacks of the AWKL are also found in the keyboard disclosed in U.S. Patent No. 4,298,773 to Diab wherein the keyboard map is not based on any
30 similarity with QKL or any existing Latin keyboard layout.

Similarly, U.S. Patent No. 4,670,842 to Metwaly discloses an Arabic keyboard, wherein the arrangement of Arabic letters is based on the lexicographical order of the Arabic alphabet, rather than any phonetic connection between letters of the Latin

alphabet and those of the Arabic alphabet. The layout disclosed in Metwaly does not correspond to the QKL. As such, a typist on the Metwaly keyboard who is familiar with the QKL must learn an entirely unrelated Arabic script layout.

- 5 There have been prior attempts made to adapt the Arabic keyboard to QKL typists. U.S. Patent No. 5,416,898 to Opstad correlates many keys when compared to the simple AWKL layout. For example, the following Arabic letters:

م and ق, ر, ت, ا, س, د, ف, ج, ك, ل, ز, ص, ب, ن

10

- on the keyboard disclosed in Opstad are mapped onto the Latin keys N, B, C, Z, L, K, J, F, D, S, A, T, R, Q and M, respectively, of QKL. According to Table 2, this mapping maintains the phonetic correspondence between many Arabic characters and their Latin counterparts. The Arabic symbols !, ' , ' , ' , ' , ' and ' are also arranged on this layout to match the location of their Latin counterparts. The symbol keys are also left to match their Latin counterparts with one exception: elements of paired symbols (i.e., <, [], () and {}) are position-swapped (i.e., >[,)(and }{), leading to confusion and mistyping. Another drawback of Opstad's keyboard is that a user must memorize the location of 48 Arabic characters over three modes (Normal, Shift and Option). Similar to the AWKL layout, Opstad's keyboard also suffers from the difficulty of access to modified forms and common ligatures.
- 15
- 20

- Another attempt to adapt the Arabic letters to keys on the QKL is ISLAM-91 in Java Script, a free online Arabic keyboard accessible at <http://www1.architektur.tu-darmstadt.de/islam/ara/>. While allowing for some phonetic mapping of Latin characters onto the Arabic alphabet, this layout includes additional non-intuitive mapping of Arabic characters onto Latin keyboard characters. The user must spend additional time to learn and master the new keyboard. For example, the top row of the ISLAM-91 keyboard includes six Arabic letters, which requires the user to leave the home row in order to type such characters. This mapping is counterintuitive in that Arabic characters are mapped onto numbers. Furthermore, the assignment of the letters ظ ش ث ذ to the bottom row is counter-intuitive, as is the assignment of bracket characters [and] to the keys 'Q' and 'W', respectively. In addition, Arabic letters
- 25
- 30

with modified forms are somewhat randomly scattered across the keyboard. Finally, the ISLAM-91 keyboard does not make use of the Latin keys 'P' and 'V'.

Another attempt to fashion an Arabic script data input device is described in U.S. Patent No. 6,874,960 to Daoud which discloses a keyboard that comprises a plurality of touch areas that represent symbols such as letters of the Roman script, Greek script, Hebrew script, Arabic script, or Cyrillic script. The touch areas are arranged in groups that include at least two of the touch areas. Each of the groups includes touch areas that are arranged in a distinctive shape that incorporate one or more of the symbols or parts of the symbols. The user enters a symbol by touching a part of the distinctive shape that is recognized with the symbol. There is little or no correspondence between the touch areas and the QWERTY keyboard, nor does Daoud disclose a method of inputting diacritics necessary for the Arabic language.

U.S. Patent No. 6,799,914 to Eo discloses a layout of the Arabic alphabet for a 12-key keyboard found on such devices as mobile telephones and PDAs. The keyboard described by Eo is not suited to standard keyboard typing, and is not comparable to the QWERTY or similar keyboards.

There is a need for a standard keyboard layout for typing in a language whose alphabet is based on Arabic script, which is designed for typists who have little or some knowledge of, or experience with, a Latin alphabet keyboard layout. Such an Arabic alphabet-based keyboard layout should take advantage of the user's familiarity with the Latin alphabet keyboard layout, and the knowledge of how Arabic alphabet characters are hand-stroked and lexicographically ordered in order to reduce the time it takes the user to learn the Arabic alphabet-based layout. In addition, such a layout should be simple to use and intuitive, without requiring use of multiple modes.

SUMMARY OF THE INVENTION

The present invention addresses the drawbacks of the prior art by providing an intuitive keyboard layout for a language based on the Arabic script, based on a mapping of Latin alphabet characters to characters of said language. The Latin

alphabet characters form part of a Latin alphabet, including, but not limited to, QWERTY, QWERTZ, AZERTY and Dvorak. The keyboard layout of the present invention can also be used by those learning to type in an Arabic-based language for the first time. In addition, users of the present invention who, at first, are not familiar with Latin-based keyboards, can quickly learn one or more prior art Latin alphabet keyboards, based on knowledge and experience with the present invention. The invention reduces the user's learning curve by leveraging (i) the user's previous knowledge of Latin alphabet keyboard layouts, (ii) the lexicographic ordering of letters of alphabets based on the Arabic script; and (iii) the manner in which the letters and words are hand-stroked in languages based on the Arabic script.

The keyboard layout of the present invention further remedies the problems present in prior art layout designs as it exploits shape and sound similarities that are inherent in Arabic-based scripts when hand-stroking in the Arabic-based language. In one single mode (the Normal mode), a typical user of the invention who has familiarity of one or more prior art Latin keyboard layouts may be able to (i) type at a rate similar to that achieved on a Latin alphabet layout, (ii) intuitively remember the location of all letters, and (iii) extend his or her typing abilities to intuitively type any of the diacritics commonly used in languages based on the Arabic script.

20

The term 'keyboard' used herein refers to both physical and virtual keyboards.

The keyboard layout of the present invention assigns characters to keys based on shape, sound, or frequency analysis. In contrast to prior art keyboard layouts that assign one letter per key in any one single mode, or, assign multi-characters uniformly to a single key with little or no regard to the shape-similarity, sound-similarity, or the letter frequency analysis, the keyboard layout of the present invention uses a multi-character key, and assigns multiple Arabic-based script letters and diacritics to keys based on shape, sound, and letter frequency distribution analysis.

30

In one aspect of the present invention, there is provided a keyboard for inputting characters of an Arabic script-based language. The keyboard is based on a mapping of a Latin alphabet keyboard onto alphabet characters of that language, wherein the mapping comprises phonetic similarity between Latin alphabet members and alphabet

members of the language, shape similarity within alphabet members of the language, and lexicographic ordering of alphabet members within the language. The mapping further comprises frequency analysis of alphabet members of the language. 5.

Each alphabet character of the Latin alphabet keyboard is mapped onto 'n' characters of the language, with $n = 1$ to 5. The language may be chosen from Arabic, Farsi (Persian), Urdu, Pashto, Baloch, Malay, Balti, Brahui, Panjabi, Kashmiri, Sindhi, Uyghur, Kazakh, Kyrgyz, Azerbaijani and Kurdish. The language is preferably a widely-spoken language such as Arabic, Farsi (Persian) or Urdu. The Latin alphabet keyboard from which the invention is mapped is selected from the group consisting of QWERTY, AZERTY, QZERTY and Dvorak. Furthermore, the keyboard of the present invention can be used as part of or in association with devices such as, but not limited to, laptop computers, desktop computers, wireless phones, handheld computers, MP3 playing devices, interactive remote controls, two-way pagers, automobile PCs, navigational computers, data loggers, assistance technology devices, electronic games, and graphic pads. 15

In another aspect of the present invention, there is provided a keyboard for inputting characters of Arabic, Farsi, and/or Urdu. The keyboard is based on a mapping of a Latin alphabet keyboard onto one or more alphabet characters 13. The keyboard of claim 12, wherein for a Latin alphabet key associated with two or more Farsi characters, said Farsi characters are accessed sequentially by multiple presses of said Latin alphabet key, or by use of Function keys in association with said Latin alphabet key. of each language, wherein the mapping comprises phonetic similarity between Latin alphabet members and alphabet members of each language, shape similarity within alphabet members of a given language, and lexicographic ordering of alphabet members within a given language. The mapping further comprises frequency analysis of alphabet members of the language. Where a Latin alphabet key is associated with two or more Arabic, Farsi or Urdu characters, the characters may accessed sequentially by multiple presses of the Latin alphabet key, or by use of Function keys in association with the Latin alphabet key. The Latin alphabet keyboard from which the invention is mapped is selected from the group consisting of QWERTY, AZERTY, QZERTY and Dvorak. Furthermore, the keyboard of the present invention can be used as part of or in association with electronic devices such as, but not limited to, wireless phones, handheld computers, MP3 playing devices, interactive remote 20 25 30

controls, two-way pagers, automobile PCs, navigational computers, data loggers, assistance technology devices, electronic games, and graphic pads.

Various objects, features, aspects and advantages of the present invention will become
5 more apparent from the following detailed description of preferred embodiments of the invention, along with the accompanying drawings.

BRIEF DESCRIPTION OF FIGURES

10

FIG. 1 shows the 47 character-producing keys found on prior art Latin keyboard layouts, with labelled row numbers and various shading and texture to illustrate how such keys are mapped to the fingers of a touch-typist.

15 FIG. 2 shows a typical arrangement of character keys in a prior art QWERTY keyboard layout.

FIG. 3 shows a known keyboard layout for typing in Arabic, called the Arabic Windows™ Keyboard Layout (AWKL). The illustrated keys show characters in both
20 modes: the normal mode showing the bottom characters, and the shift mode showing the upper characters.

FIGS. 4a – 4e illustrate an Arabic keyboard layout of the present invention correlated to a QWERTY keyboard..

25

FIGS. 5a – 5e illustrate an Arabic keyboard layout of the present invention correlated to an AZERTY keyboard.

FIG. 6 shows a flowchart for text input using a keyboard of the present invention.

30

DETAILED DESCRIPTION

Before the present invention is described in further detail, it is to be understood that the invention is not limited to the particular embodiments described, as such may, of course, vary. It is also to be understood that the terminology used herein is for the purpose of describing particular embodiments only, and is not intended to be limiting, since the scope of the present invention will be limited only by the appended claims.

Unless defined otherwise, all technical and scientific terms used herein have the same meaning as commonly understood by one of ordinary skill in the art to which the present invention pertains. Although any methods and materials similar or equivalent to those described herein can also be used in practice or test of the present invention, a limited number of exemplary methods and materials are described herein.

Wherever ranges of values are referenced within this specification, sub-ranges therein are intended to be included within the scope of the invention unless otherwise indicated. Where characteristics are attributed to one or another variant of the invention, unless otherwise indicated, such characteristics are intended to apply to all other variants of the invention where such characteristics are appropriate or compatible with such other variants.

The following is given by way of illustration only and is not to be considered limitative of this invention. Many apparent variations are possible without departing from the spirit and scope thereof.

The present invention will be described, without loss of generality, with reference to the prior art Latin keyboard known as QKL. It should be obvious to those skilled in the art, however, that the invention is also applicable to any keyboard layout that hosts Latin alphabet letters. These layouts include, but are not limited to: QWERTY-like layouts (e.g. QWERTZ, AZERTY, etc.), and Dvorak or Dvorak-like layouts.

For the purposes of this invention, the discussion is limited to the most common character printing keys common on most keyboards regardless of the layout type.

The Arabic script languages include, but are not limited to: Arabic, Farsi (also known as Persian), Urdu, Pashto, Baloch, Malay, Balti, Brahui, Panjabi (in Pakistan), Kashmiri, Sindhi (in India and Pakistan), Uyghur (in China), Kazakh (in China), Kyrgyz (in China), Azerbaijani (in Iran) and Kurdish (in Iraq and Iran). While the detailed description focuses on a keyboard for Arabic, Urdu and Farsi, the present invention is applicable to the Arabic script languages listed above.

Embodiments of the present invention include, but are not limited to, a complete 1-to- n mapping from the keys on a QKL to all the Arabic letter characters shown in Table 1, where $n = 1, 2, 3, 4$ or 5 . That is, each key from the QKL is mapped to 1, 2, 3, 4 or 5 characters of the Arabic alphabet. For example, 'A' is mapped to five characters of the Arabic alphabet, 'Q' is mapped to one character, 'S' is mapped to two characters, and so on. It will be apparent to those skilled in the art in consideration of the invention, however, that similar mappings may be drawn from any keyboard layout to the Arabic characters.

Starting from the knowledge base of a QKL user, it may be assumed that the locations of the 26 standard Latin alphabet keys have been, to some degree, memorized even by the average typist and that a corresponding muscle memory exists. The keyboard layout of the present invention leverages this muscle memory of key locations of the QKL user by establishing a correlation between each letter in the Latin alphabet and one or more intuitively suitable counterparts from the Arabic letters or diacritic symbols. Such mapping reduces the learning curve of a QKL-familiar user. In establishing such correlations, the invention relies on factors such as: (i) direct sound pronunciation similarity, (ii) approximate sound pronunciation similarity, (iii) direct shape similarity, (iv) approximate shape similarity; (v) key location proximity, and (vi) frequency distribution analysis of Arabic letters. In cases where more than one candidate Arabic character can be assigned to a QWERTY key, the invention relies on a letter frequency distribution analysis of Arabic letters performed on more than 1,375,058 words (giving a total of 5,452,865 letters), This analysis is documented at <http://www.intellaren.com/web/software/alfa/alfa.html> , the results of which are shown in Table 3. The number column in Table 3 corresponds to the number column of Table 1.

The present invention also introduces two letters imported from the English alphabet: P, transcribed in Arabic as پ, and V, transcribed in Arabic as ف. These two letters are imported for purposes of accurate transcription and transliteration from Latin to Arabic.

5

TABLE 3 - Frequency of Arabic letters in descending order

Number	Letter	Frequency	Number	Letter	Frequency
1	ا	12.54	33	ة	1.38
23	ل	12.04	35	ى	1.26
25	ن	6.71	5	ج	1.21
24	م	6.61	30	إ	1.03
28	ي	6.38	14	ص	1.01
27	و	5.90	9	ذ	0.99
26	هـ	5.04	4	ث	0.85
2	ب	4.59	7	خ	0.79
10	ر	4.18	13	ش	0.73
18	ع	3.94	11	ز	0.52
29	أ	2.88	16	ط	0.49
20	ف	2.82	15	ض	0.45
21	ق	2.66	19	غ	0.33
3	ت	2.64	32	ء	0.32
8	د	2.61	36	ئ	0.29
12	س	2.43	17	ظ	0.18
22	ك	2.11	31	آ	0.17
6	ح	1.83	34	و	0.09

As shown in Table 1, the Arabic alphabet available on any keyboard layout designed to produce Arabic script is typically composed of the 48 characters (38 letters and 10 non-letter symbols). These are grouped as follows: (i) characters 1 to 28 are the 28 primary Arabic letters consisting of 25 consonants and three vowels; (ii) characters 29 to 36 are eight modified letters; (iii) characters 37 and 38 are two imported English letters for transcription purposes; (iv) characters 39 to 47 are the nine diacritic symbols used as an aid for correct pronunciation; and (v) character 48 is a letter-stretching symbol that enhances readability of the script.

Similarities in the shape and sound of Arabic letters form a significant aspect of the lexicographical ordering within the Arabic alphabet. Students of the Arabic language often exploit these similarities in order to learn and memorize the Arabic alphabet. In some instances, successive letters resemble each other in shape and/or sound, and vary only in the number of dots. Examples of successive letters that have shape similarity include the pair ب and ت (B and T) and the pair ر and ز (R and Z). Examples where successive letters share both sound and shape similarity include the pair ت and ث (T and TH as in “thank”), and the pair د and ذ (D and TH and in “then”). In each of these examples, the succeeding letter has one more dot than its predecessor. Such an observation is important in the development of the keyboard layout of the present invention.

Further, when hand-writing the aforementioned Arabic letters, the succeeding letter is usually written with one or more strokes than its predecessor, where a stroke is a single but continuous hand sketch or a mouse drag without any spatial discontinuity in the form of lifting the hand or releasing the mouse, respectively. The additional strokes are usually in the form of one or more dot(s) stroked above or below the letter. Therefore, the user who writes in Arabic knows that whereas a letter takes one stroke to draw, a similar succeeding letter would take one or more strokes to draw. Examples from Table 1 include: (i) letter 14, ص is written in one stroke, whereas the succeeding letter, ض, requires an additional stroke in the form of a dot; and (ii) letter 16, ط, can be written in one or two strokes, whereas the succeeding letter, ظ, requires an additional stroke in the form of a dot.

In the keyboard layout of the present invention, each Latin key is associated with one or more alphabetic or diacritic character from the Arabic set of letters and diacritics. Where the Latin key is associated with two or more Arabic characters, the second Arabic character is produced by pressing the corresponding Latin key twice within a given time interval (for example, between 150 - 750 milliseconds). Similarly, the third Arabic character is produced by pressing the corresponding Latin key a third time within the same or a similar allowed tolerance after the second press, and so on. Performing the same operations while depressing the SHIFT key can produce the results in reverse order. Alternatively, the function keys (e.g. F2, F3, F4 and F5) can

be used to simulate the number of times a Latin key should be pressed in order to produce the required Arabic character.

According to the phonetic pronunciation of the 28 Arabic letters listed in Table 2, there exists a direct one-to-one phonetic correspondence between 19 Latin and Arabic letters as follows:

- | | |
|----|-----------|
| | 1. A → ا |
| | 2. B → ب |
| 10 | 3. C → ص |
| | 4. D → د |
| | 5. F → ف |
| | 6. H → هـ |
| | 7. J → ج |
| 15 | 8. K → ك |
| | 9. L → ل |
| | 10. M → م |
| | 11. N → ن |
| | 12. Q → ق |
| 20 | 13. R → ر |
| | 14. S → س |
| | 15. T → ت |
| | 16. V → ث |
| | 17. W → و |
| 25 | 18. Y → ي |
| | 19. Z → ز |

It should be noted that while the letter V has no equivalent in the Arabic alphabet, the letter ث (which is a member of the Farsi (Persian) alphabet), is used for the purposes of accurately transcribing or transliterating Latin nouns containing the letter V.

Nineteen of the Latin letters can directly reference nineteen Arabic letters solely based on direct vocal sound similarity. Statistically, this means that 73% (19 of 26) of the Latin letters can already be used to directly reference 50% (19 of 38) of the Arabic characters listed in Table 1 (i.e. character numbers 1 to 38 of Table 1).

Next, nine of the above Arabic letters can be grouped with subsets of Arabic letters based on direct sound- and/or shape-similarity:

20. A → ا → آ, إ, إ and ء
 21. B → ب → پ
 22. C → ص → ض
 5 23. D → د → ذ
 24. K → ك → خ
 25. S → س → ش
 26. T → ت → ة and ث
 27. W → و → و
 10 28. Y → ي → ى and ئ

As such, fourteen other Arabic letters can be indirectly referenced from the above 19 Latin letters already known to the QKL user based on intuitive relationships amongst the Arabic letters themselves. This accounts for another 37% (14 of 38) of Arabic letters.

Therefore, in accordance with the present invention, 73% of the Latin alphabet corresponds to 87 % (33 of 38) of the Arabic letters.

The three primary Arabic vowels (ا, و and ي) are grouped with their respective several modified forms, based on similarity in both sound and shape.

The letter پ (pronounced as 'paa') is phonetically identical to the letter P. Although not part of the Arabic alphabet *per se*, it is a member of the Urdu and Farsi (Persian) alphabets which are derived from the Arabic alphabet. However, the letter پ is used in Arabic scripts today for the purposes of accurately transcribing Latin nouns containing the letter P. It should be noted that ب and پ are both shape- and sound-similar; shape-wise, the latter is rendered with two extra dots below. This justifies mapping پ to the second press of the letter key B.

The letter ض succeeds the letter ص in the Arabic alphabet. Since it requires only one more stroke than ص in the form of a dot, and is less frequent than the letter ص, it is assigned to the second press of letter key C.

The letter ذ is pronounced like 'th' (as in the word 'than'). The shape- and sound-similarity between ذ and the letter د is apparent. Furthermore, ذ is written with one

more stroke than د in the form of a dot than د , and it occurs 2.2 times less frequently than د . It is therefore intuitive to assign it to the second press of letter key D.

5 The letter خ (pronounced 'khaa') has no equivalent letter sound in the English language. However, it is most similar in sound to the letter ك . Since the letter خ is written with two strokes, and is less frequent than ك , it is intuitive to assign it to the second press of letter key K.

10 The letter ش is phonetically identical to the sound produced by 'SH'. It also succeeds the letter س in the Arabic alphabet. The shape- and sound-similarity between س and ش is apparent. Furthermore, ش has three dots on top whereas the letter س is dot-less, and ش occurs 2.7 times less frequently than س . It is therefore intuitive to assign ش to the second press of letter key S.

15 The character ة has the same vocal sound as that of the Latin letter T, but its use is restricted to finishing off feminine nouns only. The direct sound pronunciation similarity warrants rendering ة accessible through the second press of letter key T. Meanwhile, the letter ث is pronounced like 'th' (as in 'three'), and is similar in shape to the letter ت . Since the letter ث occurs less frequently than the letter ة (according to
20 Table 3), it is made accessible through three presses of the letter key T.

From the above groupings, it follows that the letter key 'A' is assigned five Arabic characters (ا , آ , إ , إ and ء); the letter key 'B' is assigned two Arabic characters (ب and پ), and so on. For example, to type the letter ش, a typist either i) presses S twice
25 within a specified time tolerance, ii) presses SHIFT S, where the SHIFT key causes reverse accessibility, or simply iii) presses F2 and then the letter S to obtain the character ش. Similarly, to generate the letter آ, the typist either i) presses A four times while respecting the time frame, ii) presses SHIFT 'A' twice while respecting the allowed time frame, or iii) presses F4 and then presses the letter key 'A'.

30

Where three or more Arabic characters are assigned to one Latin letter key (as in the case of A, T and Y), the letter frequency analysis described above provides a basis for the order of assignment of Arabic letters to a given Latin letter key. As an example,

according to the frequency analysis shown in Table 3, the letters ت , ة , ث occur at respective frequencies of 2.64%, 1.38% and 0.85%. All three letters are accessible through the key 'T'. Since ة occurs more frequently than ث , it can be obtained by pressing the key 'T' twice within a certain time interval, while ث can be obtained by pressing 'T' three times. Alternatively, the SHIFT key or the function keys F2 and F3 can be used in conjunction with the key 'T' to produce ة or ث .

There remains only five out of thirty-eight Arabic letters from Table 1 to be assigned, namely: ح, the pair ط and ظ, and the pair غ and ف.

10

The letter ح is very close in vocal sound to that generated by هـ, which corresponds to the letter H in English. When transcribing nouns containing the letter ح from Arabic to English, for example, it is typically transcribed and transliterated as "H". Since the letter G has yet to be assigned and is to the immediate left of H on the QWERTY keyboard layout, it is assigned to the letter ح. It should be noted that for Latin alphabet keyboard layouts other than QWERTY, the letter key 'G' may not be to the immediate left of the letter key 'H'. Nonetheless, due to the shape similarity between 'G' and ح, such keyboards can also maintain the same assignment. Alternatively, given the sound similarity between the letters ح and هـ, and the fact that ح is several times less frequent than هـ (see Table 3), it follows that ح can be accessed by pressing the letter 'H' key twice.

Next, we consider the lexicographical ordering of the letters ص , ض , ط and ظ, which are listed as having positions 14, 15, 16 and 17, respectively, in Table 1. Since the letter key 'C' is assigned to both ص and ض, it follows that the unassigned letter key 'X', which is to the immediate left of the letter key 'C' on the QWERTY keyboard, is assigned to ط. Furthermore, ظ requires one more stroke (in the form of a dot) than ط, its lexicographical predecessor. It follows naturally to assign ظ to the second press of letter key X. This assignment, like that of letter key 'G' to ح, uses key location proximity in conjunction with the letters in the Arabic alphabet. It should be noted that in most other Latin-based keyboard layouts (a notable exception being Dvorak), the letter key 'X' is always to the left of the letter key 'C'.

Attention is now directed to the last pair: ع and غ. When an Arabic word containing the letter ع is transcribed into English or another Latin-based language, it is typically represented by vowels such as A, O, I, U. Since the letter P is unused in the Arabic alphabet and is to the immediate right of a subset of the vowel letters O, I, and U on a QWERTY keyboard, it is assigned to the Arabic letter pair ع and غ. As discussed below, the vowel letters O, I and U are assigned to Arabic diacritic marks based on sound similarity between the diacritics and English vowels. Therefore, the remaining letters are assigned as follows:

- 29. G → ح
- 30. P → ع
- 31. X → ط

And in turn, the last two Arabic letters in the above list can reference the two dotted letters that resemble them in shape:

- 32. P → ع → غ
- 33. X → ط → ظ

For convenience and consistency reasons, optionally, the letter خ is accessible through the letter ح (key 29 above) as well as through the letter ك.

In addition to the three long vowels mentioned above, there are also short vowels represented by simple-shaped symbols that are written above or under an Arabic letter. These symbols, or diacritics, (in Arabic, harakaat - حركات or tashkeel - تشكيل) serve as aids to accurate pronunciation of words. Table 4 provides a list of diacritics, along with examples of their usages when compared to English, as an example of a Latin-based language.

As seen from Table 4, Arabic diacritics are best approximated on the QWERTY keyboard by corresponding English vowels. In the present embodiment, the letter O is assigned the *dammah* and diacritics similar to the vowel 'O'; namely, the *dammah tanween* , َ , and the *sukoon* , ْ . The diacritic *dammah tanween* is basically two *dammahs* and can therefore be written as َّ instead of َ , but the latter is preferred

because it is quicker when written by hand. The diacritic *sukoon* is seated on top of any alphabet letter to indicate that there is a no vowel sound associated with that letter. The shape-similarity between the letter O and the *sukoon* ' is obvious; it is therefore intuitive to assign the character ' to the next available press of O.

5

Next, the letter I is assigned the *kasrah* and the related diacritic *kasrah tanween*, which is basically two *kasrahs*. In order to maintain key location proximity amongst the diacritics, the letter key ‘U’ is designated to render the *fat-ha* and the related diacritics *fat-ha tanween* and the superscripted *alif*. The *fat-ha tanween* is basically two *fat-has*, while the superscripted *alif*, also known as *alif mamdoodah*, is placed above letters and plays the role of a long *alif* vowel; it is almost exclusively used in sacred Koranic script. It should be noted that whereas the ideal letter for modeling *fat-ha* and related diacritics is the letter A, this key is already used for modeling the long vowel and/or consonant forms of A, the letter ا and four associated modified forms. Therefore, the diacritics are generated as follows:

34. O \rightarrow _____
35. I \rightarrow _____
36. U \rightarrow _____

20

The three diacritics can in turn be used as pointers to several other diacritics that are shape- and/or sound-dependent, namely:

- 25
37. $O \rightarrow ' \rightarrow "$ and $'$
38. $I \rightarrow ' \rightarrow "$
39. $U \rightarrow ' \rightarrow "$ and $'$

The remaining characters to be assigned are the *shaddah*, which is usually used in combination with most of the above diacritics, and the extending character which is added to horizontally stretch the shape of certain letters for better readability. These two characters are assigned to the one remaining and unassigned vowel: E. This assignment is suitable for rendering the *shaddah* since the shape of the *shaddah* resembles the shape of an E that is rotated by 90 degrees; and it is suitable for the shape-extending character since E can be regarded as an acronym for the English

word *extension*, which is suggestive of the character. Therefore, the assignment is as follows:

40. E → َ and -

5

TABLE 4 - Diacritics of the Arabic language

Diacritic	Sound	Shape & Location	English example	Arabic equivalent	Arabic letters
Fat-hah = فَتْحَة	Short “a” vowel	َ, above the letter	Panama (the last “a” produces a long vowel)	پَنَامَا	ا م ن پ
Dhammah = ضَمَّة	Short “o” or “u” vowel	ُ, above the letter	Yugoslavia (the “u” produces a long vowel while the “o” produces a short one)	يُوغُسْلَاوِيَا	ا ي ف ا ل س ن غ و ي (the second letter shows a long vowel, while the third shows a short one)
Kasrah = كَسْرَة	Short “e” or “i” vowel	ِ, under the letter	Finland	فِنْلَنْدَا	ا د ن ل ن ف
Sukoon = سُكُون	No vowel sound permitted (no air during pronunciation)	ْ, above the letter	Armenia	أَرْمِينِيَا	ا ي ن ي م ر ا
Shaddah = شَدَّة	Doubling of consonants	ّ, above the letter	Jannah, Mohammad (note the doubled consonants)	مُحَمَّدٌ, جَنَّة	ج ن ه and د م ح م (note how the fat-ha is combined with the shaddah to control the vowel sound)
Tanween = تَنْوِين	The ending sounds of “an”, “on” or “in”, depending on whether the haraka of the last letter is a fatha, dammah, or kasra	َ, ُ, ِ, on the last letter of the word; it is simply a doubling of the last haraka	waladan, waladon, waladin (walad = a boy).	وَلَدَا, وَلَدُ, وَلَدٍ	د ل و (note that in the case of tanween of fat-ha, an ا is added but is never pronounced)
Hamzah* = هَمْزَة	A glottal stop	ء, above vowels, or at end of words	nash-aa (origin), ra-ouf (linient), bareia' (innocent), sama-a (sky)	رَوْوُفٌ, نَشَاءٌ, بَرِيءٌ, سَمَاءٌ	ه ا ش ن ف و ر ب ي ا ر ب ء ا م س
Maddah* = مَدَّة	Long vowel “a” following a hamzah	~ , only above the letter alif(ا)	Quraan (read as two syllabi: qur/aan)	قُرْآن	ن ا ر ق

* The hamzah and maddah interact with short vowels to modify pronunciation.

The above assignments are summarized in

- 5 which shows an English-Arabic Character Table (EACT) of the present invention. The keyboard layout corresponding to Table 5 is shown in the five planes illustrated in Figs. 4a – 4e. Each plane is a function of the number of key presses provided they are performed within a certain time tolerance. In particular, Fig. 4a illustrates the Arabic characters obtained after pressing a given key once; Fig. 4b refers to Arabic
- 10 characters obtained after pressing a given key twice; Fig. 4c refers to Arabic characters obtained after pressing a given key three times; Fig. 4d refers to Arabic characters obtained after pressing a given key four times; and Fig. 4e refers to Arabic characters obtained after pressing a given key five times. Blank keys in subsequent planes indicate that no further characters are mapped to those keys. Therefore,
- 15 pressing a key more than the number of characters assigned to it simply causes it to start from the beginning in a rotary fashion. Note that when the number of presses is more than one, the number of presses may also be simulated by pressing any of the function keys, such as F2 to F5, to simulate the corresponding number of presses prior to the character key. Last, pressing the SHIFT key reverses the order of generating the
- 20 characters. If the modifier keys SHIFT or CAPS-LOCK are active, the order of access of Arabic characters is reversed. For example, if successive presses within tolerance of Key “a” produce ا ا ا ا ا , then Key “A” produces ا ا ا ا ا . The two punctuation characters “?” and “.” correspond to their Arabic equivalent ؟ and . , respectively. All other keys on the keyboard keys not shown in Table 5 produce the same
- 25 characters shown on the typical QWERTY keyboard. The numbers 0, 1, 2, ... are sometimes used in Arabic as well, although these can be modified to generate the Arabic numerals ٠, ١, ٢, ٣, ٤, ٥, ٦, ٧, ٨, ٩ (which are respectively 0,1,2,3,4,5,6,7,8,9). The modification can occur through the use of the SHIFT or ALT modifier keys.

- 30 The following example illustrates how to type an Arabic sentence using the keyboard layout of the present invention. Repeated characters shown in brackets are accessed by pressing the association key within a certain time interval. For example, ‘(dd)’ means that the letter key ‘d’ is pressed twice, within a maximum time interval, to

produce ٣; '(aaa)' means the key 'a' is pressed three times within a maximum time interval to produce ا , and so on.

Arabic sentence: هذا الاختراع له علاقة بلوحات المفاتيح وطرق إدخال الحروف بالعربي

- 5 Keys used: h(dd)a al(aaa)(kk)trap lh plaq(tt) blwgat almfatyg wxrq (aaa)d(kk)al algrwf balprby. English Translation: The invention relates to keyboards and methods of entering letters in Arabic.

- Another embodiment of the present invention is illustrated in the five planes of Figs. 10 5a – 5e, where the 48 Arabic characters of Table 1 are mapped onto an AZERTY keyboard, based on the relationships between Arabic and Latin characters discussed above, and the frequency analysis summarized in Table 3. Each plane is a function of the number of key presses provided they are performed within a certain time tolerance. In particular, Fig 5a illustrates the Arabic characters obtained after pressing 15 a given key once; Fig. 5b refers to Arabic characters obtained after pressing a given key twice; Fig. 5c refers to Arabic characters obtained after pressing a given key three times; Fig. 5d refers to Arabic characters obtained after pressing a given key four times; and Fig. 5e refers to Arabic characters obtained after pressing a given key five times. Blank keys in subsequent planes indicate that no further characters are mapped 20 to those keys. Therefore, pressing a key more times than the number of characters assigned to it simply causes it to start from the beginning in a rotary fashion. An AZERTY layout has the following differences from a QWERTY layout: i) the positions of keys A and Q are swapped, ii) the positions of Z and W are swapped, and iii) Key M is located at the right end of the home-row (or, Row 3).

25

TABLE 5 - Mapping of Arabic characters to a QWERTY keyboard

Latin Character	Number of accessible Arabic Characters (NOAC)	Number of key presses and associated Arabic character				
		0	1	2	3	4
	NOAC					
a	5	ا	أ	إ	آ	آ
b	2	ب	پ			
c	2	ص	ض			
d	2	د	ذ			
e	2	-	-			
f	1	ف				
g	1	ح	خ			
h	1	هـ				
i	2	-	-			10
j	1	ج				
k	2	ك	خ			
l	1	ل				
m	1	م				
n	1	ن				
o	3	-	-	-		
p	2	ع	غ			15
q	1	ق				
r	1	ر				
s	2	س	ش			
t	3	ت	ة	ث		
u	3	-	-	-		
v	1	ف				
w	2	و	ؤ			
x	2	ط	ظ			20
y	3	ي	ى	ئ		
z	1	ز				

The methodology used to map Arabic characters onto a Latin alphabet keyboard may also be used to separately map Farsi (or Persian) and Urdu characters onto the keyboard, as illustrated respectively in Tables 6 and 7.

- 5 As seen in Table 6, the Farsi alphabet has four additional characters: δ , produced by two presses of the Latin character 'h'; چ , produced by two presses of 'j'; ک , produced by three presses of 'k'; and ژ , produced by two presses of 'z'. The two punctuation characters "?" and "." correspond to their Farsi equivalent. All other keys on the keyboard keys not shown in Table 6 produce the same characters shown on the typical
- 10 QWERTY keyboard.

- As shown in Table 7, the Urdu script has eight additional characters (compared to the Arabic script), four of which are identical to the aforementioned Farsi characters (i.e., δ , چ , ک , and ژ) and can thus be mapped like the four Farsi characters. In addition, the
- 15 Urdu script has four additional characters: ڈ , produced from two presses of 'd'; ڑ , produced from two presses of 'r'; ٹ , produced from three presses of 't'; and ے , produced from four presses of 'y'. The two punctuation characters "?" and "." correspond to their Urdu equivalent. All other keys on the keyboard keys not shown in Table 7 produce the same characters shown on the typical QWERTY keyboard.

20

In each of the Farsi and Urdu keyboards, The numbers 0, 1, 2, ... of the Latin character keyboard can be modified to generate the respective Farsi and Urdu numerals The modification can occur through the use of the SHIFT or ALT modifier keys.

25

TABLE 6 - Mapping of Farsi (Persian) characters to a QWERTY keyboard

Latin Character	Number of accessible Farsi Characters (NOFC)	Number of key presses and associated Farsi character				
		0	1	2	3	4
	NOFC					
a	5	ا	آ	إ	آ	ء
b	2	ب	پ			
c	2	ص	ض			
d	2	د	ذ			
e	2	ـ	-			
f	1	ف				
g	1	ح	خ			
h	2	هـ	هـ			
i	2	ـ	ـ			
j	2	ج	چ			
k	3	ك	خ	گ		
l	1	ل				
m	1	م				
n	1	ن				
o	3	و	و	و		
p	2	ع	غ			
q	1	ق				
r	1	ر				
s	2	س	ش			
t	3	ت	ة	ث		
u	3	ـ	ـ	ا		
v	1	ف				
w	2	و	و			
x	2	ط	ظ			
y	3	ي	ی	ئ		
z	2	ز	ژ			

TABLE 7 - Mapping of Urdu characters to a QWERTY keyboard

Latin Character	Number of accessible Urdu Characters (NOUC)	Number of key presses and associated Urdu character				
		0	1	2	3	4
a	5	ا	آ	إ	آ	ء
b	2	ب	پ			
c	2	ص	ض			
d	3	د	ذ	ڈ		
e	2	ـ	ـ			
f	1	ف				
g	1	ح	خ			
h	2	ھ	ہ			
i	2	ـ	ـ			
j	2	ج	چ			
k	3	ک	خ	گ		
l	1	ل				
m	1	م				
n	1	ن				
o	3	و	و	ہ		
p	2	ع	غ			
q	1	ق				
r	2	ر	ڑ			
s	2	س	ش			
t	4	ت	ہ	ٹ	ٹ	
u	3	ـ	ـ	ا		
v	1	ٹ				
w	2	و	و			
x	2	ط	ظ			
y	4	ی	ی	ئ	ے	
z	2	ز	ڑ			

A flowchart for text input of the present invention is shown in Fig. 6. Conventions within the flowchart are as follows: *italic* words indicate program variables & constants; words suffixed with () denote program functions; and underlined words indicate programming keywords.

5

The descriptions within the flowchart are as follows:

- *prevKey* → previous pressed key;
- *prevTime* → previous recorded system time;
- *TOL* → time tolerance, the program starts with a default 250 milliseconds, but
10 can be reset by the typist;
- *dispPosition* → the position at which the next character is displayed. If it refers to the position of a displayed character, it is over-written by the new character;
- *currTime* → current system time;
- 15 - *alc* → Arabic letter counter, used to determine which column to access the Arabic character from;
- *EACT*[*i*][*j*] → Access the *i*th row, *j*th column in English-Arabic Character Table (EACT, shown in Table 5);
- *tChar* → character retrieved from Table 5; and
- 20 - *display(k, p)* displays key *k* at position *p*.

The rendering of characters on an editor as based on user typing proceeds according to the following algorithm (each box is numbered with its corresponding step below) :

- [1] Perform one-time preliminary initializations when the editor application loads up
25 as follows:
- Set previous key (*prevKey*) to null, meaning no keys have so far been processed.
 - Set previous time (*prevTime*) to be the current time of the system. By
30 the time the user actually enters a key in the future, the time of such a press is recorded; as a result, the time obtained at this initialization stage is set to become “previous” time.

- Set tolerance (TOL) to a default time frame (e.g., 300 milliseconds). The user is allowed to modify such a value. Consecutive key presses within this TOL carry the logic that the user indeed wants to access the next character within the adjacency list (or row) shown in Table 5.
- 5 ○ Set the display position (dispPosition) to 0 at this time, which is the location where the next letter to be rendered is displayed. In most programming languages (Java, C++, etc.), counting of memory addresses and indexed elements starts from 0, rather than 1.
- 10 [2] get the key the user just pressed; store it in a variable name current key (currKey).
- [3] currKey is examined: If it is a function key between F2 and F5, proceed to step 9; otherwise, proceed to step 4.
- 15 [4] get the current time of the system and store it in variable currTime.
- [5] calculate the time difference (timeDiff) between the two recorded times shown in the equation.
- 20 [6] set prevTime to be currTime, this way, the next time a key is pressed, that time overwrites the current value stored in currTime, while the old time is stored in prevTime.
- [7] This step runs two tests: i) checks whether the currKey and the previous key are the same, and ii) checks whether the time difference is less than or equal to TOL of step 1. If either test fails, proceed to Step 8. It should be noted that for the first pass through the flowchart, the test must fail since prevKey is null and no key on the keyboard can generate it. This means that the Arabic letter column (alc) to choose a letter from will be reset to be the first column as shown in step 8. This column is in the EACT shown in Table 5, the column heading is 0. In total, there are five columns to choose from, labeled 0 to 4 under the alc, wherein “0” means first. If the test passes, proceed to step 11.
- 25 the same, and ii) checks whether the time difference is less than or equal to TOL of step 1. If either test fails, proceed to Step 8. It should be noted that for the first pass through the flowchart, the test must fail since prevKey is null and no key on the keyboard can generate it. This means that the Arabic letter column (alc) to choose a letter from will be reset to be the first column as shown in step 8. This column is in the EACT shown in Table 5, the column heading is 0. In total, there are five columns to choose from, labeled 0 to 4 under the alc, wherein “0” means first. If the test passes, proceed to step 11.
- 30 the EACT shown in Table 5, the column heading is 0. In total, there are five columns to choose from, labeled 0 to 4 under the alc, wherein “0” means first. If the test passes, proceed to step 11.

[8] Reset the Arabic Letter Column to the value 0. This means that the character will be chosen from the column with header value 0 in Table 5 (characters headers run from 0 to 4 and are highlighted in light gray).

- 5 [9] Being in this step means that a function key between F2 and F5, inclusive, has been pressed already, so the actual QWERTY letter key must be obtained and stored in currKey.

10 [10] The value of alc is updated accordingly to reflect the column that must be accessed in Table 5 to retrieve the corresponding Arabic character from. Here, we have arrived from step 3 followed by step 9. If the function key pressed is F2, this means that two presses are simulated, so the second column in the Arabic Letter Columns in Table 5 is accessed (Column with heading 1). Similarly, if F4 is pressed, alc is set to 3 to access the fourth column.

15

[11] This step is reached only if both tests in step 7 succeed. Reaching this step means a different key in the adjacency list is about to be retrieved instead of the one currently picked and displayed. Therefore, alc is incremented by one.

20 [12] This step precisely identifies the column from which to select the Arabic letter. Given the complexity of this step, a number of examples are provided. Note that the symbol “%” means modulus, also referred to as the mod operation in computing. Modulus is simply another word for remainder when a division operation is carried out. Therefore, the equation $(a \% b)$ reads $a \bmod b$. So, $7 \bmod 5$ is 2; $12 \% 5$ is 2; and
25 $3 \bmod 5$ is 3.

To type ة, the typist presses letter key t twice. The letter t has a numerical value which maps it to the correct row in EACT of Table 5. The number of Arabic letters accessible through t is 3 (see column NOAC). Therefore, the equation in step 12 is simplified as follows (because counting starts from 0, 1 means two presses):

30

$$\text{alc} = 1 \% \text{EACT}[\text{numerical value of t}][\text{number of accessible letters}]$$

$$\text{alc} = 1 \% 3$$

alc = 1, which is used to access the second column (remember, counting starts from 0)

Now, suppose the user press t four times, but there are only three corresponding Arabic characters, then the equation will still produce the correct Arabic letter after cycling all the way through. In other words, pressing t four times for the present invention is the same as pressing it once. This is illustrated as follows:

$$\text{alc} = 3 \% \text{ EACT}[\text{numerical value of t}][\text{number of accessible letters}]$$

$$\text{alc} = 3 \% 3$$

$$\text{alc} = 0, \text{ which is used to access the first column (the column with heading 0).}$$

10

[13] the display position (dispPosition) is updated. Basically, the last printed character is deleted in preparation for the new character to be printed in its place. This happens typically within a fraction of a second.

15 [14] Using the same approach as in step 6, prevKey is updated to hold this current key, so that when a new key is pressed in the next iteration, both old and new keys are correctly compared in step 7 cycled through.

[15] This step fetches the next character to be rendered from the adjacency table of Table 5. For example, $\text{EACT}[\text{t}][1] \rightarrow \text{ة}$. Since the value of alc is always correct and within range, the correct corresponding character is retrieved from EACT.

20

[16] the fetched character, tChar, is displayed at the designated position, disposition.

25 [17] display position (dispPosition) is updated in preparation for the next user action of inserting more letters from EACT.

In summary, the present invention provides a keyboard for typing alphabets that are based on the Arabic script. The invention fully exploits the following intuitive relationships between Latin and characters of the Arabic-based script: (i) direct phonetic similarity; (ii) approximate phonetic similarity; (iii) direct shape similarity; (iv) approximate shape similarity and (v) key location proximity.

30

In addition, the present invention makes use of only 26 keys of the Latin alphabet for all of the characters of the Arabic-based script, thereby facilitating the user's task. The minimal use of Latin keys is based on assigning a subset of Arabic-based script characters to a given Latin key. Members of a given subset are accessed in an order,
5 preferably determined by a frequency analysis of the individual members of the Arabic-based alphabet.

The keyboard of the present invention can be used as part of or in association with electronic devices such as, but not limited to, wireless phones, handheld computers,
10 MP3 playing devices, interactive remote controls, two-way pagers, automobile PCs, navigational computers, data loggers, assistance technology devices, electronic games, and graphic pads.

15 CONCLUSION

The foregoing has constituted a description of specific embodiments showing how the invention may be applied and put into use. These embodiments are only exemplary. The invention in its broadest, and more specific aspects, is further described and
20 defined in the claims which now follow.

These claims, and the language used therein, are to be understood in terms of the variants of the invention which have been described. They are not to be restricted to such variants, but are to be read as covering the full scope of the invention as is
25 implicit within the invention and the disclosure that has been provided herein.

What is claimed is:

1. A keyboard for inputting characters of an Arabic script-based language, said keyboard based on a mapping of a Latin alphabet keyboard onto characters of said language, wherein said mapping comprises:
 - phonetic similarity between Latin alphabet characters and alphabet character of said language;
 - shape similarity within alphabet characters of said language; and
 - lexicographic ordering of alphabet characters of said language.
2. The keyboard of claim 1 wherein said mapping further comprises frequency analysis of alphabet characters of said language.
3. The keyboard of claim 1 or 2, wherein said language is selected from the group consisting of Arabic, Farsi, Urdu, Pashto, Baloch, Malay, Balti, Brahui, Panjabi, Kashmiri, Sindhi, Uyghur, Kazakh, Kyrgyz, Azerbaijani and Kurdish.
4. The keyboard of claim 3, wherein said language is selected from the group consisting of Arabic, Farsi and Urdu.
5. The keyboard of any one of claims 1 to 4, wherein each alphabet character of the Latin alphabet keyboard is mapped onto 'n' characters of said language, with n = 1 to 5.
6. The keyboard of any one of claims 1 to 5, wherein said Latin alphabet keyboard is selected from the group consisting of QWERTY, AZERTY, QZERTY and Dvorak.
7. The keyboard of any one of claims 1 to 6, wherein said keyboard is adapted to activate a device selected from the group consisting of laptop computers, desktop

computers, wireless phones, handheld computers, MP3 playing devices, interactive remote controls, two-way pagers, automobile PCs, navigational computers, data loggers, assistance technology devices, electronic games, and graphic pads.

5

8. The keyboard of any one of claims 1 to 7, wherein said keyboard is virtual.

9. A keyboard for inputting characters of a language selected from Arabic, Farsi and Urdu, said keyboard based on a mapping of a Latin alphabet keyboard onto one or more of said characters, wherein said mapping is based on:

10

phonetic similarity between Latin alphabet characters and alphabet characters of said language;

shape similarity within alphabet characters of said language; and

15

lexicographic ordering of alphabet characters within said language.

10. The keyboard of claim 9, wherein said mapping further comprises a frequency analysis of alphabet characters of said language.

20 11. The keyboard of claim 9 or 10, wherein said language is Arabic and the mapping comprises:

Latin Character	Number of accessible Arabic Characters	Associated Arabic character(s)				
		0	1	2	3	4
	NOAC					
a	5	ا	آ	إ	أ	ء
b	2	ب	پ			5
c	2	ص	ض			
d	2	د	ذ			
e	2	-	-			
f	1	ف				
g	1	ح	خ			
h	1	هـ				
i	2	-	-			
j	1	ج				10
k	2	ك	خ			
l	1	ل				
m	1	م				
n	1	ن				
o	3	و	و	و		
p	2	ع	غ			
q	1	ق				15
r	1	ر				
s	2	س	ش			
t	3	ت	ة	ث		
u	3	-	-	ا		
v	1	ف				
w	2	و	و			
x	2	ط	ظ			
y	3	ي	ى	ئ		20
z	1	ز				

25 12. The keyboard of claim 11, wherein for a Latin alphabet key associated with two or more Arabic characters, said Arabic characters are accessed sequentially by multiple presses of said Latin alphabet key, or by use of Function keys in association with said Latin alphabet key.

30 13. The keyboard of claim 9 or 10, wherein said language is Farsi and the mapping comprises:

	Latin Character	Number of associated Farsi Characters	Associated Farsi character(s)				
			ا	آ	إ	آ	ء
5	a	5	ا	آ	إ	آ	ء
	b	2	ب	پ			
	c	2	ص	ض			
	d	2	د	ذ			
	e	2	ـ	ـ			
	f	1	ف				
	g	1	ح	خ			
10	h	2	هـ	هـ			
	i	2	ـ	ـ			
	j	2	ج	چ			
	k	3	ك	خ	گ		
	l	1	ل				
	m	1	م				
	n	1	ن				
15	o	3	و	ی	ہ		
	p	2	ع	غ			
	q	1	ق				
	r	1	ر				
	s	2	س	ش			
	t	3	ت	ة	ث		
	u	3	ـ	ـ	ا		
20	v	1	ف				
	w	2	و	و			
	x	2	ط	ظ			
	y	3	ي	ی	ئ		
	z	2	ز	ژ			

14. The keyboard of claim 13, wherein for a Latin alphabet key associated with two or more Farsi characters, said Farsi characters are accessed sequentially by multiple presses of said Latin alphabet key, or by use of Function keys in association with said Latin alphabet key.

15. The keyboard of claim 9 or 10, wherein said language is Urdu and the mapping comprises:

5

10

Latin Character	Number of associated Urdu Characters	Associated Urdu character(s)				
		0	1	2	3	4
	NOUC					
a	5	ا	آ	إ	آ	ء
b	2	ب	پ			
c	2	ص	ض			
d	3	د	ذ	ڈ		
e	2	ـ	-			
f	1	ف				
g	1	ح	خ			
h	2	ھ	ہ			
i	2	ـ	ـ			
j	2	ج	چ			
k	3	ک	خ	گ		
l	1	ل				
m	1	م				
n	1	ن				
o	3	و	و	ہ		
p	2	ع	غ			
q	1	ق				
r	2	ر	ڑ			
s	2	س	ش			
t	4	ت	ة	ٹ	ٹ	
u	3	ـ	ـ	ا		
v	1	ف				
w	2	و	و			
x	2	ط	ظ			
y	4	ي	ی	ئ	ے	
z	2	ز	ڑ			

16. The keyboard of claim 15, wherein for a Latin alphabet key associated with two or more Urdu characters, said Urdu characters are accessed sequentially by multiple presses of said Latin alphabet key, or by use of Function keys in association with said Latin alphabet key.

17. The keyboard of any one of claims 9 to 16, wherein Latin alphabet keyboard is selected from the group consisting of QWERTY, AZERTY, QZERTY and Dvorak.

18. The keyboard of any one of claims 9 to 17, wherein said keyboard is adapted to
activate a device selected from the group consisting of laptop computers, personal
computers, wireless phones, handheld computers, MP3 playing devices,
interactive remote controls, two-way pagers, automobile PCs, navigational
5 computers, data loggers, assistance technology devices, electronic games, and
graphic pads.

19. The keyboard of any one of claims 9 to 18, wherein said keyboard is virtual.

10

15

20

25

	Latin Character	Number of associated Urdu Characters	Associated Urdu character(s)				
			0	1	2	3	4
		NOUC					
5	a	5	ا	آ	إ	آ	ء
	b	2	ب	پ			
	c	2	ص	ض			
	d	3	د	ذ	ڈ		
	e	2	ـ	-			
	f	1	ف				
	g	1	ح	خ			
	h	2	ھ	ہ			
10	i	2	ـ	ـ			
	j	2	ج	چ			
	k	3	ک	خ	گ		
	l	1	ل				
	m	1	م				
	n	1	ن				
15	o	3	و	و	و		
	p	2	ع	غ			
	q	1	ق				
	r	2	ر	ڑ			
	s	2	س	ش			
	t	4	ت	ة	ث	ط	
	u	3	ـ	ـ	ا		
20	v	1	ف				
	w	2	و	ؤ			
	x	2	ط	ظ			
	y	4	ي	ی	ئ	ے	
	z	2	ز	ژ			

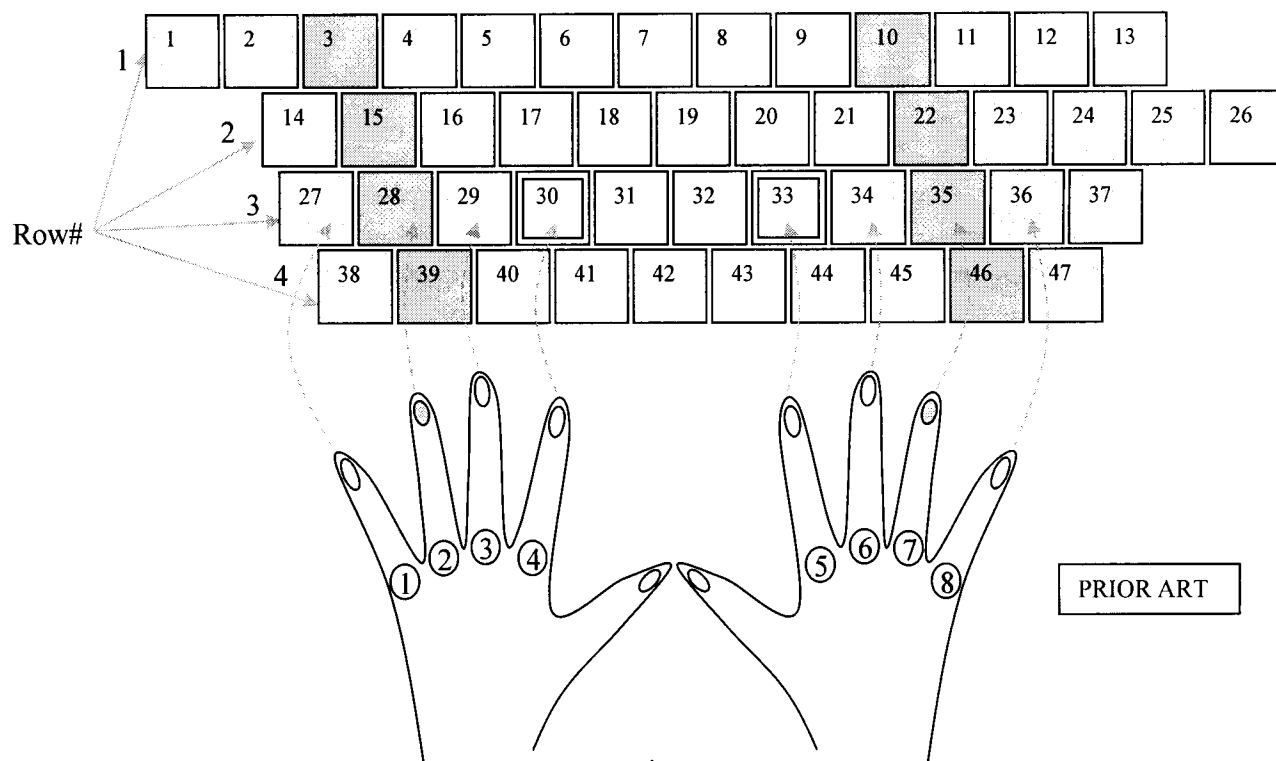


Fig. 1

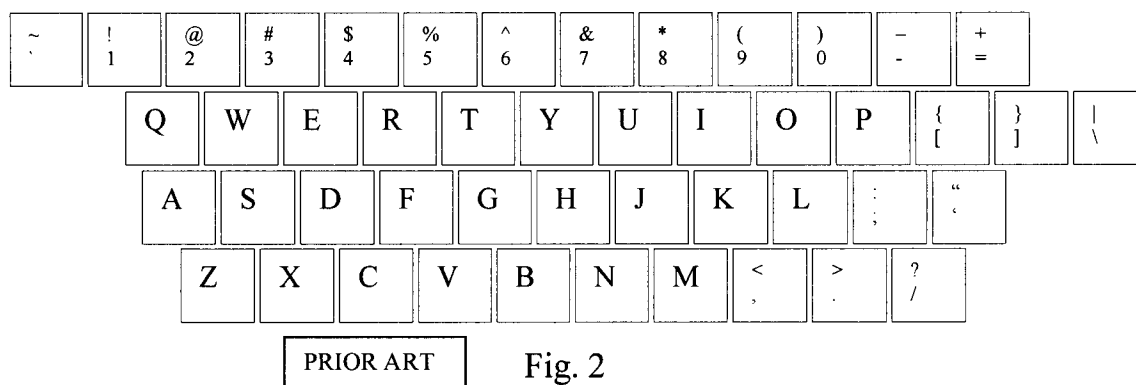


Fig. 2

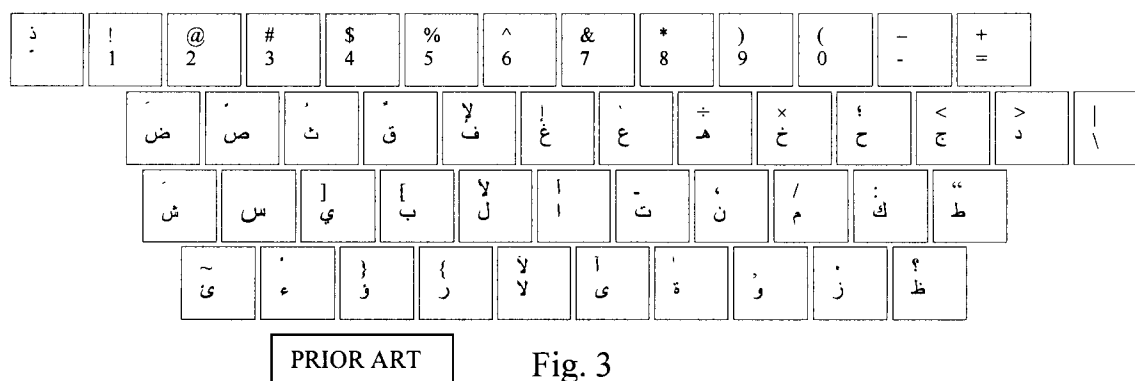


Fig. 3

Fig. 4a

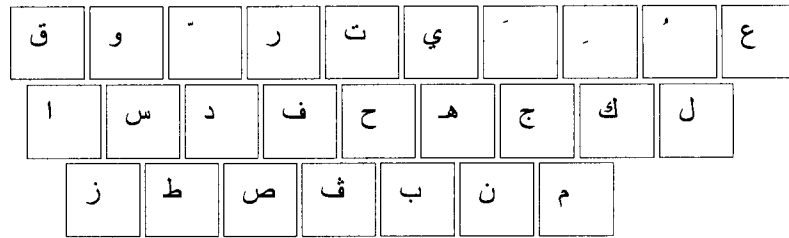


Fig. 4b

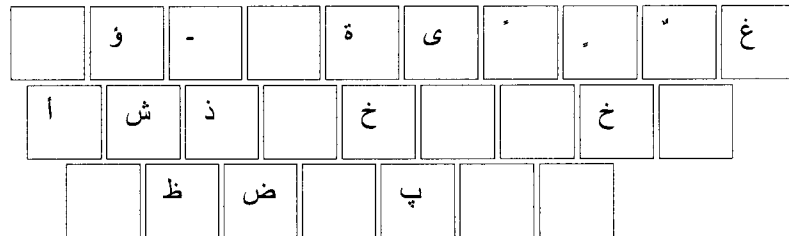


Fig. 4c

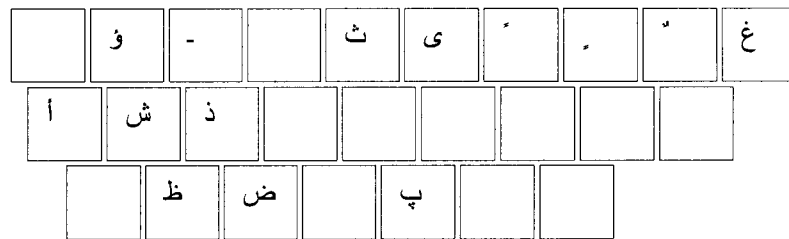


Fig. 4d

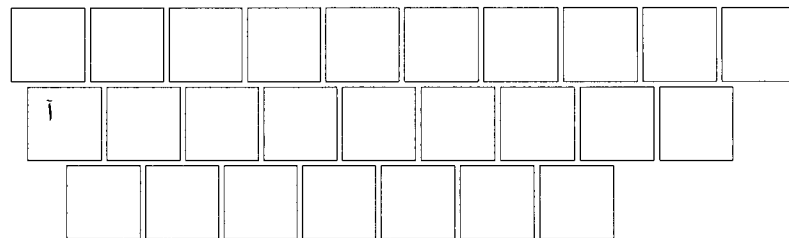
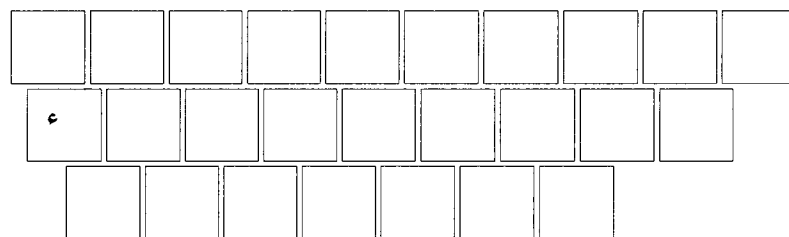


Fig. 4e



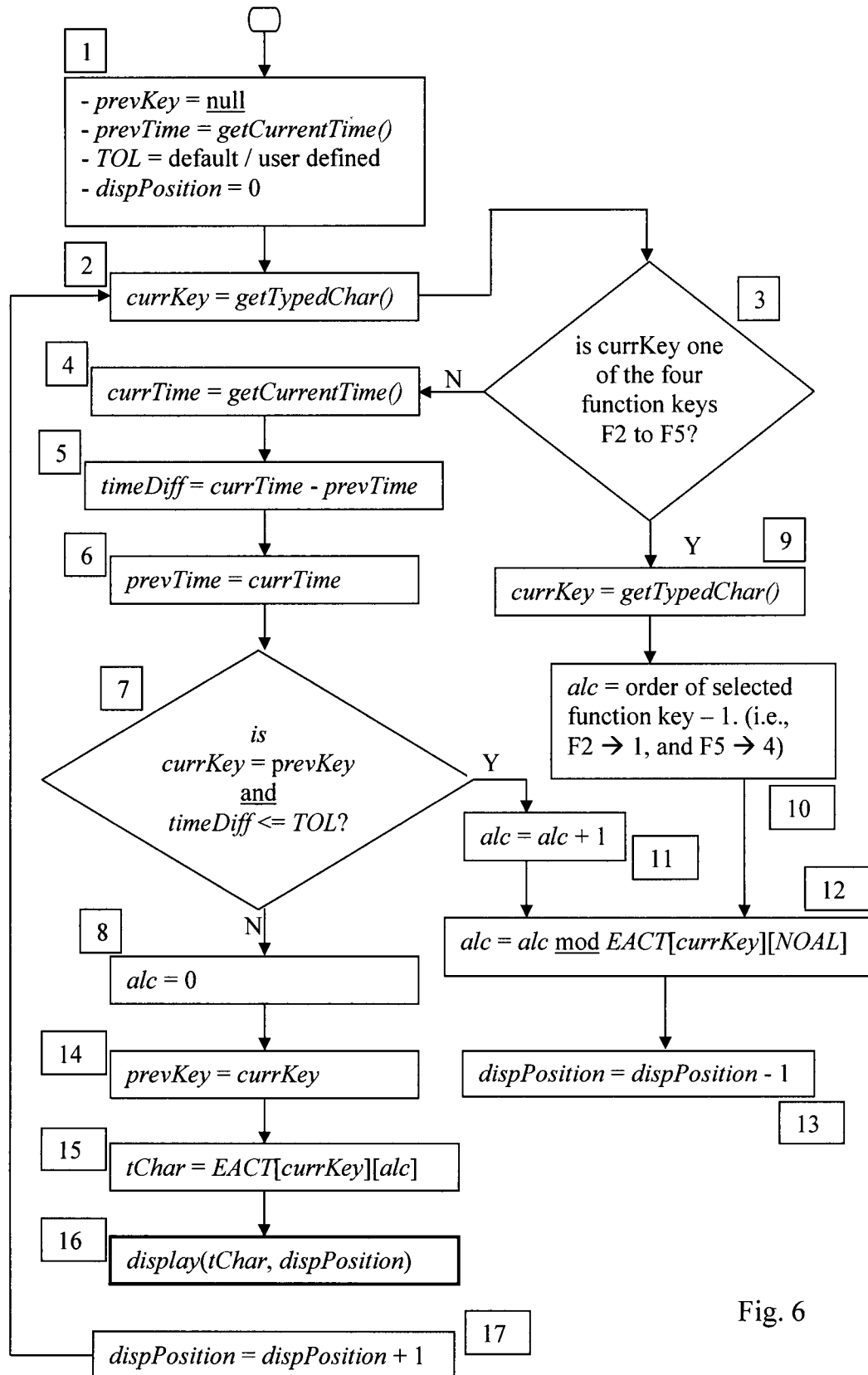


Fig. 6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CA2010/000219

A. CLASSIFICATION OF SUBJECT MATTER

IPC: **G06F 3/023** (2006.01)

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC: **ALL** (2006.01)

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic database(s) consulted during the international search (name of database(s) and, where practicable, search terms used)

TotalPatent & keywords:

Arabic keyboard/layout/script/alphabet/characters/letters/symbols; Latin/Roman alphabet; (re-)map/(re-)assign/re-arrange/convert/translate/adapt;
phonetic/sound correlation/similarity/resemblance/match/correspondence/related/equivalent;
shape similarity/resemblance/match; qwerty, azerty, qzerty, dvorak

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2008/077393 A1 (<i>Gao et al.</i>) - 27 March 2008 (27-03-2008) * [0032]; Figs. 4, 6 *	1, 9
A	US 5,416,898 (<i>Opstad et al.</i>) - 16 May 1995 (16-05-1995) * col. 7, ll. 6-13 *	1, 9
A	[document cited in the application] US 2007/222644 A1 (<i>Jung et al.</i>) - 27 September 2007 (27-09-2007) * abstract *	1, 9
A	US 4,507,734 (<i>Kaldas</i>) - 26 March 1985 (26-03-1985) * col. 4, ll. 29-30, 60-65; Fig. 3 *	
A	GB 2 007 413 A (<i>Diab</i>) - 16 May 1979 (16-05-1979) * Fig. 5 *	
A	US 4,527,919 (<i>Aoun</i>) - 9 July 1985 (09-07-1985)	

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be of particular relevance	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E" earlier application or patent but published on or after the international filing date	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&" document member of the same patent family
"O" document referring to an oral disclosure, use, exhibition or other means	
"P" document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search

13 May 2010 (13-05-2010)

Date of mailing of the international search report

21 May 2010 (21-05-2010)

Name and mailing address of the ISA/CA
Canadian Intellectual Property Office
Place du Portage I, C114 - 1st Floor, Box PCT
50 Victoria Street
Gatineau, Quebec K1A 0C9
Facsimile No.: 001-819-953-2476

Authorized officer

Cristian S. Popa (819) 997-2299

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CA2010/000219

Patent Document Cited in Search Report	Publication Date	Patent Family Member(s)	Publication Date
US2008077393A1	27-03-2008	US2008077386A1	27-03-2008
US5416898A	16-05-1995	FR2691819A1 FR2691819B1	03-12-1993 25-10-1996
US2007222644A1	27-09-2007	US2007222644A1 US7576662B2	27-09-2007 18-08-2009
US4507734A	26-03-1985	FR2490365A1 FR2490365B1	19-03-1982 21-11-1986
GB2007413A	16-05-1979	CA1121061A1 DE2847085A1 ES474730A1 GB2007413A GB2007413B US4145570A US4298773A	1982-03-3 1979-05-31 1979-11-01 1979-05-16 1982-04-28 1979-03-20 1981-11-03
US4527919A	09-07-1985	AT38079A AT380834B AU532224B2 AU4397479A BE873950A1 BR7900726A CA1123360A1 CH640781A5 CS7900688A2 CS241017B2 CY1187A DD141912A5 DE2902312A1 DE2902312C2 DK29479A EG13982A ES477504A1 FR2416117A1 FR2416117B1 FR2590843A1 GB2013577A GB2013577B HK36583A IN150815A1 IT7919574D0 IT1109760B JP55019591A MY18584A NL7900830A SE439898B SE439898C SE7901020A SE7901020L SG17983G	15-12-1985 10-07-1986 22-09-1983 16-08-1979 06-08-1979 04-09-1979 11-05-1982 31-01-1984 13-06-1985 13-03-1986 07-10-1983 28-05-1980 09-08-1979 14-11-1985 08-08-1979 31-03-1983 16-06-1980 31-08-1979 31-12-1987 05-06-1987 15-08-1979 08-12-1982 23-09-1983 25-12-1982 24-01-1979 23-12-1985 12-02-1980 31-12-1984 09-08-1979 08-07-1985 17-10-1985 08-08-1979 08-08-1979 11-01-1985