International Journal of Computer Applications (0975 – 8887)
Volume 77– No.7 September 2013

An Implementation of Dynamic Reconfigurable Touch Screen Keyboard

Ajeya Anand, Dhanush R Nair, Nithin Paul Cherian, Rajat Kumar, Rajesh Sola

Advanced Computing Training School (ACTS),
Center for Development of Advanced Computing (CDAC), Pune

ABSTRACT
In this paper, an implementation of Dynamic Reconfigurable Touch Screen Keyboard (DRTSK) using a touch-screen panel and display are integrated in a development board is presented. An ARM920T based development kit (FriendlyARM) is programmed and implemented as a touch screen reconfigurable keyboard device. The FriendlyARM is interfaced with PC. The communication is through standard interfaces. The reconfigurable touch screen keyboard implemented on the hardware, tested under different operating systems and the functionality is validated.

General Terms
Keyboard, ARM, User Interface, Input Device.

Keywords
Dynamic,Reconfigurable,Keyboard,Touch-screen, ARM.

1. INTRODUCTION
The paper deals with implementation of Dynamic Reconfigurable Touch Screen Keyboard [1]. Touch-screen interfaces are effective as generic pointing devices. The implementation involved a touch-screen panel and display integrated to a development board. An ARM [2] based development board is chosen to support graphical display. The touch-screen allows the user inputs to be detected with the help of sensor and proper ADC interrupt subroutines are activated as per the programming.

The development kit is interfaced with the PC. Communication is through standard serial port and USB. A 16/32 bit ARM920T RISC microprocessor based FriendlyARM[10] development kit is programmed in C to support Graphical Display and Touch response. The required keyboard design is obtained as an image; extracted using MATLAB. The required keyboard design display, mapping and reconfiguration functions of keys were done using Embedded C through Code Warrior IDE and DNW downloader. The development kit is interfaced with the PC where the desired keyboard outputs are obtained. The keyboard implementation is tested under Windows and Linux Operating Systems.

1. Motivation

The motivation for the development of a dynamic reconfigurable touch screen keyboard comes from the various necessities faced by normal keyboard users.

A non-English-user has to struggle with a standard English keyboard[8]. This is because of the unavailability of the desired keyboard layouts in the hardware. Though some attempts are seen in commercialising customised layouts and desired language keys, it does not inclusively satisfy all the intended users of a computer. This is a problem especially when a single computing resource is shared by people having multiple and different language preferences.

Some keyboard layouts such as Dvorak keyboards are scientifically proven to be ergonomically better than normal “QWERTY” layout, but unavailability of the hardware leads to limited usage of the proven method. A Touch-screen keyboard [3] with the facility to change keyboard layouts and inclusion of the desired fonts [4] would help out the user to cross the basic input hurdle to computation. Keyboard should be able to handle single shortcut keys such as single keys to enter email-ids and frequently visited websites and data.

More than the ergonomics of the typing, users prefer to have their own keyboard layout; this is driven by the fact that most handheld devices are oriented towards a single user mode of operation. The user thus wishes to customize the controls and the reconfigure-ability of the keyboard is important for the single end user’s preference of typing based on language and keyboard layouts.

The idea emerges from the motivation to build a keyboard which would have universal acceptance and ease of use. The project aimed to implement a few keys that are reconfigurable [5][7]; the output is obtained in a terminal application. More keys and key layouts are proposed to be included incrementally with function to switch between layouts and reconfigure.

2. SURVEY OF KEYBOARD VARIANTS
Touch-screen interfaces are quite popular as keyboard interfaces. Most prominent keyboard variants in the market are the Optimus Maximus, Virtual Keyboard Interfaces.

2.1 Virtual Keyboard Layouts
Virtual Keyboard layouts [6] provide an effective interface that has enhanced security features, by jumbling of alphabets it provides for safety from password theft. But, virtual keyboards are not generic. Mostly they are customised implementations for a specific task such as banking services, railway information kiosk or other similar service. A lack of dedicated hardware and portability are the major drawbacks from a common user’s perspective.

2.2 Virtual Projection Type Keyboards
As discussed in reference [1] the keyboards of this type are costly and need a surface for projection. Also, some colours of the projection surface may render such projection type keyboards a hassle to use.

2.3 Optimus Maximus

Optimus Maximus is a commercially available artistic keyboard with the flexibility to include wide range of inputs and configure it according to user's wish. Affordability is the main drawback with the keyboards of this type.

3. IMPLEMENTATION OF DRTSK

3.1 Hardware Used

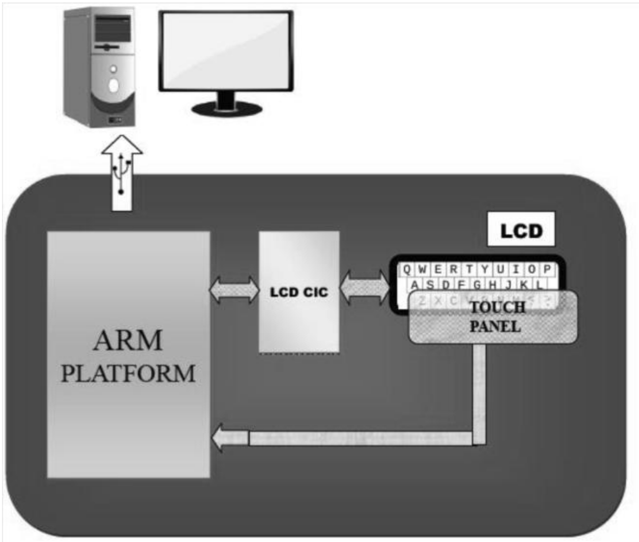

Fig 1: Schematic diagram of Hardware Interaction in DRTSK

The hardware requirements and their interaction were as depicted in Fig 1. The hardware referred is as follows.

1. *ARM platform*: ARM based development kit with standard interface for communication with PC.

2. *LCD CIC*: LCD Controller IC (SSD 1963)

3. *LCD*: TFT LCD (>5") chosen so as to display required keyboard design.

4. *TOUCH PANEL*: Suitable Touch Panel over LCD for touch interface.

An ARM based development kit The FriendlyARM -Micro 2440 is used. The FriendlyARM kit supports Graphical LCD and Touch Screen panel.

The FriendlyARM kit based on the Samsung SC2440 ARM920T based microprocessor has a 16/32 bit RISC microprocessor. The SAMSUNG's S3C2440A[9] 16/32 bit RISC microprocessor is designed to provide low power, simple and elegant, low cost, high performance solutions in small die size. It adopts the Advanced Microcontroller Bus Architecture (AMBA). The S3C2440A has the following features:-16/32 bit ARM 920T RISC processor ARM920T implements MMU, AMBA BUS, and Harvard architecture.

3.2 Software Used

Code Warrior ARM Development Suite 1.2. The Metrowerks Codewarrior IDE is used for compilation of the code and generating the binary file for the FriendlyARM kit.

Embedded C , the versatile language for perfect control on the hardware functionalities. *MATLAB R2008a,* for image extraction for desired keyboard design to be displayed. *DNW Download Tool* –Serial Console with serial/USB download. In Linux –*Minicom and Usbpush* combination helps to achieve the same functionality as DNW download tool under Windows.

3.3 Algorithm

The flowchart (Fig 2) depicts the implemented logic in a simple manner.

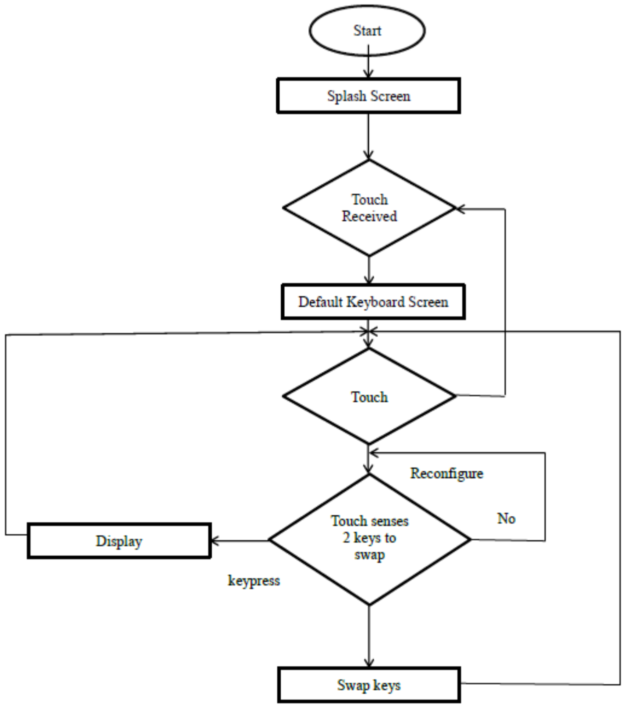

Fig 2: Flowchart depicting the sequence of operations

The implemented program starts with a default splash screen on switching on. A touch is sensed as hardware interrupt. The hardware interrupt is sensed by the in-built ADC, the call of interrupt requests for the default keyboard screen. This gives the primary keyboard layout to the user. The user can use this layout and get the desired characters typed. This is transmitted via the serial port to input the corresponding characters on PC screen. The user can reconfigure the keyboard layout using the reconfigure option provided in every screen. The desired keyboard layout can be obtained by the user by the swapping logic. The user has to click on any two keys and those keys get swapped. The desired configuration may also be obtained by multiple swapping.

The swapping operation is achieved through the implementation on code which maps the co-ordinate of the key and assigns it to the new desired key. Thus the user is able to get the desired layout.

3.4 Implementation and Testing

The code is compiled using CodeWarrior IDE to get binary output. This is flashed onto the kit. The keyboard layouts implemented are as follows.

Fig 3: Default Keyboard layout implementation

In Fig.3 the implementation of the default layout of the keyboard is shown. It is possible to display any image that represents an action such as backspace or a set of characters and map it to the display grid. Whereby, the flexibility to include any key is made possible. With this it is possible to include multiple language layouts also.

Fig 4: Image showing the second section of implementation in DRTSK

The second keyboard layout is shown in Fig. 4. Here, the keyboard layout has the options to clear the screen as indicated by the key "cls" on the display. The second layout enables the user to input some symbols and lower case Roman alphabets. The output has been tested and validated on Terminal with serial port support under Windows and also under Linux.

4. CONCLUSION

The Dynamic Touch Screen Keyboard (DRTSK) has been implemented. Reconfiguration key supports reconfiguration of keys to implement user defined layouts. As a hardware input device this is tested and found comfortable by most users. In fact, most users wanted to play around with the swapping operation giving us ideas to implement some touch sense based arcade games in the kit. With inclusion of more languages in the DRTSK it is possible to satisfy users with multiple and differing language preferences. Hence, the problem of shared computing resources but different language preference can be solved. The keyboard can be more interactive and help the user to get the desired input function for the end user.

5. ACKNOWLEDGMENTS

Thanks to Mr. Ashish Kuvelkar for constant motivation and continuous encouragement. Special thanks to Mr. Swapnil Belhe for encouragement and technical guidance and Ms. Bhawwna Aggarwal for all the support provided. Thanks to the internet forums and ARM resources for sharing knowledge and helping in timely manner.

6. REFERENCES

[1] S.R. Patil. Vichare Akanksha Subhash, Salvi Leena Avinash, Saumya Bhasin, Sparsh- A Reconfigurable Touch Screen Keyboard, Emerging Trends in Computer Science and Information Technology - 2012(ETCSIT2012) Proceedings published in International Journal of Computer Applications® (IJCA)

[2] A. N. Sloss, D. Symes, C. Wright, ARM System Developer's Guide:Designing and Optimizing System Software, Morgan Kaufmann Publishers, 2012.

[3] Seokhoon Ko, Seman Oh ,Dynamic reconfigurable screen keyboard generation method using probe key test. 01/2010; In proceeding of: Fifth IEEE International Conference on Digital Information Management, ICDIM 2010, July 5-8, 2010, Lakehead University, Thunder Bay, Canada

[4] Po, L.M. ; Wong, C.K. ; Ting, C.W. ; Ng, K.H. ; Wong, K.M. ; Cheung, C.H. ; Cheung, K.W, Dynamic candidate keypad for stroke-based Chinese input method on touchscreen devices Computers & Informatics (ISCI), 2011 IEEE Symposium on Publication Year: 2011 , Page(s): 147 – 152

[5] Usability differential in positions of software keyboard on smartphone Nakagawa, T. ; Uwano, H. Consumer Electronics (GCCE), 2012 IEEE 1st Global Conference on Publication Year: 2012 , Page(s): 304 – 308

[6] Sarcar, S. ; Ghosh, S. ; Saha, P.K. ; Samanta, D., Virtual keyboard design: State of the arts and research issues Students' Technology Symposium (TechSym), 2010 Publication Year: 2010 , Page(s): 289 – 299

[7] Son, Yunsik, Dynamic Touch-Screen UI Generation of Mobile Application Using Probe Key Test http://dx.doi.org/10.1007/978-3-642-24267-0_14 Springer Berlin Heidelberg 2011-01-01

[8] Usability differential in positions of software keyboard on smartphone Nakagawa, T. ; Uwano, H. Consumer Electronics (GCCE), 2012 IEEE 1st Global Conference on Digital Object Identifier: 10.1109/GCCE.2012.6379610 Publication Year: 2012 , Page(s): 304 – 308

[9] S3c2440a 32-Bit CMOS Microcontroller User's Manual Revision 1, Samsung Electronics, 2012

[10] Friendly ARM Internet Forum www.friendlyarm.net