Middle-East Journal of Scientific Research 20 (6): 738-741, 2014
ISSN 1990-9233

DOI: 10.5829/idosi.mejsr.2014.20.06.114155

Head Operated Human Computer Interface

Nirubama

Department of ECE,
Bharath university, Chennai-73, India

Abstract: We experimentally test the effectiveness and reliability of the accelerometer-based MEMS sensor as an interaction device for pointing and scrolling tasks on handheld devices. Easy Input is a fore head-controlled keyboard and mouse input device for paralyzed users. In addition it focuses on the invention of a head-operated computer mouse that employs mems sensors are placed to determine eye movement and to function as head operated computer mouse .pointing and scrolling are of the most extensively used tasks in almost all applications.

Key words: We experimentally test · Sensor as an interaction device · Computer mouse that employs

INTRODUCTION

Smart phones are now widely used for many of our everyday tasks. However, there are at least two reasons that make the interaction on those devices difficult compared to desktop interfaces: small screen size and limited computing power. Pointing and scrolling are of the most extensively used tasks in almost all computing applications [1].

The environment in which mobile devices are used is different from that in which desktop is used. some users cannot use both hands due to other reasons such as disability or accidents. The technique requires only eye movement We evaluate the effectiveness of using mems in the two tasks then a model for predicting their execution time is developed. From the last three years have the sensors become small, cheap and required sufficiently low power to realize this vision.

Overview

Block Diagram:

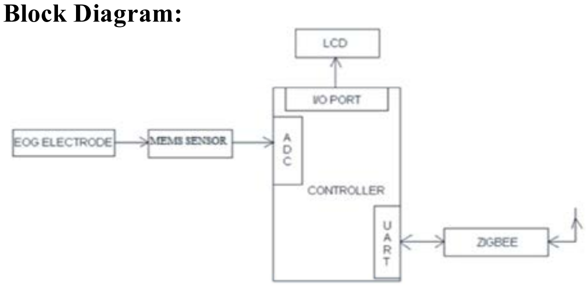

Fig. 1: Block diagram of Human side

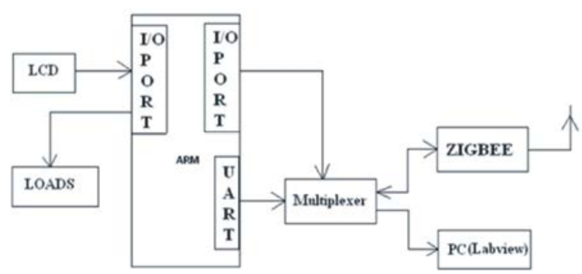

Fig 2 : Block diagram of computer side

Basic Operation: The main thesis which includes two steps one is human side and other one is computer side.

Human Side: In human side which contain mems sensors and electrodes, they placed on top and bottom of our eye. When we move our towards up and downward direction, our forehead will sense small change and gives the information to the system mainly it uses arm microcontroller and zigbee module .we use zigbee for transmitting and receiving the information [2].

System receives the information from the user as upward and downward direction. It scrolls the letters towards left and right direction. In this way scrolling operation is performed. When system receives normal direction as input from the user it points out that particular letter.in this way pointing operation is performed. Pointing and scrolling operations are performed in the system by using lab view software. In computer side also microcontroller and zigbee module is available.

Corresponding Author: Nirubama, ECE Department, Bharath university, Chennai-73, India.

It includes microcontroller ARM7LPC 2148, graphical LCD, mems sensors as hardware components and Keil software, Labview as software components.

Hardware Blocks

Power Distribution: In this we connect 230v ac input, after that we connect to bridge rectifier, it converts ac input to dc input and then we connect 12v,5v power supply to mems sensor,microcontroller unit, relay driver.

Graphical LCD: We use 16X5 LCD.LCD is connected to port P2 of the controller.JHD 16x2 A

Feautures:

- Operating voltage : 5 V.8 data lines D0-D7 (Pin 7 to Pin14) [3].
- 3 control pins **RS** (Pin4) – To send data on LCD RS is set. Clear the RS pin to send the commands to configure LCD. **WR** (Pin5) – To write data to LCD WR pin is cleared. And to read LCD screen, set WR. Generally it is not required, so this pin is grounded. **EN** (Pin6) – LCD will be active only when EN pin goes high to Low.
- 3 lines for supply GND (Pin1), VCC (Pin2), VEE (Pin3)

MEMS Sensor: A sensor which senses the orientation of itself is commonly called as mems sensor. These devices show the orientation variations in terms of voltage differences.Here we use LT1167 mems sensor.

Features:

- Single Gain Set Resistor: G = 1 to 10,000
- Gain Error: G = 10, 0.08% Max
- Input Offset Voltage: G = 10, 60mV Max
- Input Offset Voltage Drift: 0.3mV/°C Max
- Input Bias Current: 350pA Max
- Supply Current: 1.3mA Max
- Wide Supply Range: ±2.3V to ±18V

Characteristics:

- Absolute Maximum Ratings are those values beyond which the life of a device may be imparied.
- Does not include the effect of the external gain resistor RG. This parameter is not 100% tested.
- The LT1167AC/LT1167C are designed, characterized and expected to meet the industrial temperature limits, but are not tested at –40°C and 85°C. I-grade parts are guaranteed.
- This parameter is measured in a high speed automatic tester that does not measure the thermal effects with longer time constants.
- The magnitude of these thermal effects are dependent on the package used,heat sinking and air flow conditions [4].

Microcontroller Unit: Here we use LPC2103 Microcontroller

Features:

- 16-bit/32-bit ARM7TDMI-S microcontroller in tiny LQFP48 and HVQFN48 packages
- 2 kB/4 kB/8 kB of on-chip static RAM and 8 kB/16 kB/32 kB of on-chip flash program memory.
- 100 ms and programming of 256 bytes in 1 ms.
- The 10-bit ADC provides eight analog inputs, with conversion times as low as 2.44 ms per channel and dedicated result registers to minimize interrupt overhead.
- Two 32-bit timers/external event counters with combined seven capture and seven compare channels [5].

Its static function is to maintain some display information on the Graphical LCD. Its dynamic function is to control the information of the display at different angles. The dynamic function is controlled by the output level of the mems sensor. A microcontroller unit with ADC should be used or else a interface circuit is needed to perform the same functionality.

Labview Software

About Lab View: Lab VIEW is a graphical programming language that uses icons instead of lines of text to create applications. In contrast to text-based programming languages, where instructions determine program execution. Lab VIEW uses dataflow programming, where the flow of data determines execution. IN Lab VIEW, you build a user interface by using a set of tools and objects [6].

The user interface is known as the front panel. You then add code using graphical representations of functions to control the front panel objects. The block diagram contains this code. In some ways, the block diagram resembles a flowchart.

Using Lab VIEW, you can create 32-bit compiled applications that give you the fast execution speeds needed for custom data acquisition, test, measurement and control solutions. You also can create stand-alone executable and shared libraries, like DLLs, because Lab VIEW is a true 32-bit compiler.

Pointing and Scrolling Tasks: With the evolution of technology and as electronic devices increasingly become more compact, palmtop computers have become commodities. Most palmtop computers have small displays and awkward input devices, such as small buttons or tablets. To compensate for their limited data preparation ability, they are usually designed to support frequent data exchange with desktop computers (PC).

When a desktop PC adopts desktop metaphor, a companion palmtop computer should appear on the screen as an icon and it should move as the palmtop computer moves. The fundamental idea behind the Scroll Display is to make a palmtop computer into a mouse device that keeps and displays information on it and treats the palmtop computer itself as a scrolling device. This provides intuitive and direct manipulation of large objects.

Scrolling Task: Here we are controlling the mouse by using eye movement including fore head operation. The explanation for scrolling tasks is explained as follows. When we move our eye towards upward direction, forehead senses that by using mems sensor and gives that to system.

When system receives input from the user it scrolls the characters towards left direction by using lab view software.simalrly when we move our towards downward direction, then fore head senses that by using mems sensor and gives to system. When system receives down ward direction as input it scrolls the character towards right direction.

We use Zigbee module for transmitting and receiving between system and user. The front panel and block diagram view for scrolling tasks is described below.

Pointing Tasks: When we stay in normal state without any movement, by then we can point out that particular character. We can display that selected character in the text box. But we can`t stay in normal state for long time. For that we are giving some time period .The time period which we are using is in the order of milli seconds. After particular time period got over it again starts performing operation [7]. We are using Zigbee module for transmitting and receiving the information. The front panel view for the pointing tasks is described below.

Writing Operation: We can perform writing operation by using this project. We can select characters when we stay in normal state and displayed in the text box. We can write paragraphs that can be displayed in text box. Front panel for writing operation is described below. Not only controlling the mouse by using eye movement we can also perform writing operation

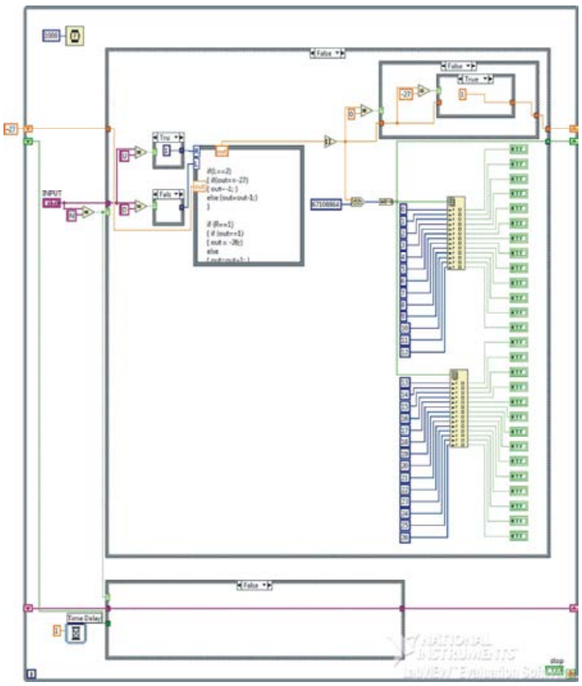

Fig. 3: Block diagram of pointing &scrolling tasks

Simulation Results

Pointing Task: When we stay in normal state without any movement, by then we can point out that particular character. We can display that selected character in the text box. But we can`t stay in normal state for long time. For that we are giving some time period. The time period which we are using is in the order of milli seconds.

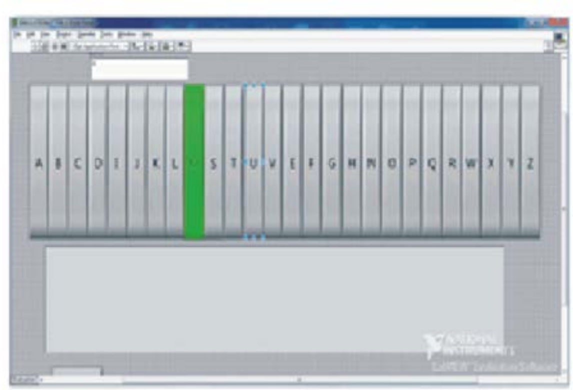

Fig. 4: Simulation result of pointing tasks

Scrolling Task: When we move our eye towards upward direction, forehead senses that by using mems sensor and gives that to system.

When system receives input from the user it scrolls the characters towards left direction by using lab view software. Similarly when we move our towards downward direction, then fore head senses that by using mems sensor and gives to system. When system receives down ward direction as input it scrolls the character towards right direction.

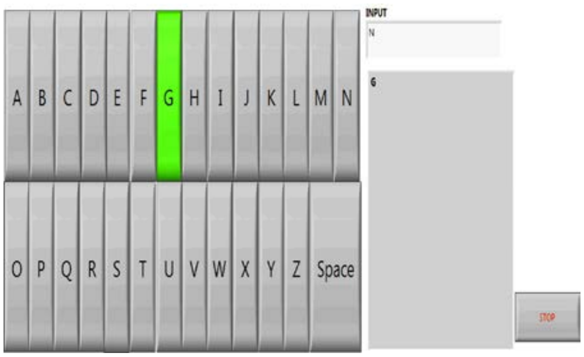

Fig. 5: Simulation result of scrolling task

Writing Operation: We can perform writing operation by using this project. We can select characters when we stay in normal state and displayed in the text box. We can write paragraphs that can be displayed in text box.

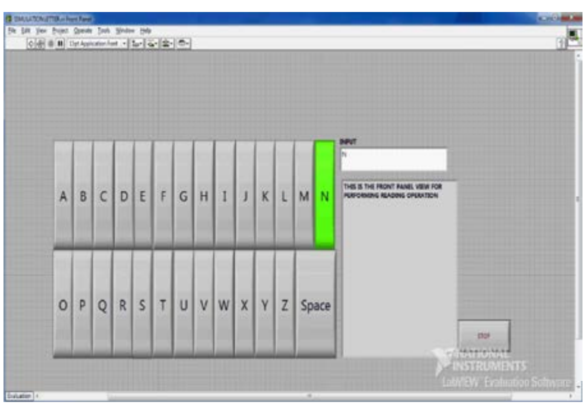

Fig. 6: Simulation result of writing operation

CONCLUSION

In this work we tested the effectiveness of pointing and scrolling tasks using Mems sensors on personal computers. The results indicate that pointing and scrolling can be effectively done by using sensors. This project is implemented according to the specifications by using ARM microcontroller. Advantage of this project is, very helpful for the paralyzed users.

Future Work: Till now using the LABVIEW software the simulation results are obtained for pointing and scrolling tasks. The working of MEMS sensors is identified. Hardware implementation of sensors interfacing with the microcontroller and then we have to give to the system as input. Finally we combine both hardware and software & we perform the operation.

REFERENCES

1. Rekimoto, J., 1996. Tilting operations for small screen interfaces. In Proceedings of the 9th Annual ACM Symposium on User interface Software and Technology(Seattle, Washington, United States, November, pp: 06-08.
2. Hinckley, K., J. Pierce, M. Sinclair E. Horvitz, 2000. Sensing techniques for mobile interaction. In Proceedings of the 13th Annual" ACM Symposium on User interface Software and Technology (San Diego, California, United States, pp: 06-08.
3. Crossan, A and R. Murray-Smith, 2004. Variability in wrist-tilt accelerometer based gesture interfaces. In: Mobile HCI, pp: 144-155.
4. Eslambolchilar, P. and R. Murray-Smith, 2004. Tilt-based Automatic Zooming and Scaling in mobile devices-a state space implementation. In S. A. Brewster and M. D. Dunlop, editors, Mobile Human-Computer Interaction-MobileHCI 2004: 6th International Symposium, volume 3160 of Lecture Notes in Computing Science, pp: 120-131, Glasgow, Scotland, September 2004. Springer-Verlag.
5. Partridge, K., S. Chatterjee, V. Sazawal, G. Borriello and R. Want, 2002. TiltType: accelerometer-supported text entry for very small devices. In Proceedings of the 15th Annual ACM Symposium on User interface Software and Technology Paris, France, pp: 27-30.
6. Quantitative analysis of scrolling techniques. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Changing Our World, Changing Ourselves (Minneapolis, Minnesota, USA, pp: 20-25.
7. Andersen, T.H., 2005. A simple movement time model for scrolling. In CHI '05 Extended Abstracts on Human Factors in Computing Systems.