ZIGBEE BASED SMART MENU

Anilkumar Joshi[1], Amit Kukadia[2], Kushal Gala[3], Bhavesh Korat[4]
Department of Electronics and Telecommunication
K.J. Somaiya Institute of Engineering and Information Technology, University of Mumbai
Mumbai, India

ABSTRACT: This project mainly focuses on designing a prototype for the restaurants which will add to its opulence and class. A graphical GUI is displayed on the screen which has the description of the items served by the restaurant. The order will be placed through the Touch-Screen on the table, which will then be received by the admin. The transmission of data is via a Zig-Bee module. Touch screen panel adds to the advantages like maximum accuracy, no training time (for the waiters) and a visual access to the food items displayed in the GUI. Furthermore Zig-Bee meets the needs of reliability, low cost, power saving and large network capacity wirelessly.

Keywords: Zig-Bee, Smart Menu, Raspberry Pi, Touch Screen, Java, Netbeans, XCTU, IPSPI40display

I. INTRODUCTION

The restaurants these days are customer-satisfaction oriented. A study showed that customers are more likely to spend more money and patronize a specific restaurant more often when they feel engaged -- if they're greeted warmly and treated with care and respect by the staff. This project Smart menu helps in adding to this feature by cutting the queues, enhancing the interaction and employee engagement, and also reducing the wait time once the order is placed. It allows guests to interact with menus, place orders, pay bills and perform a range of other self-serve functions.

II. REVIEW OF LITERATURE

The usual procedure used for food ordering in restaurants is a manual process. It involves the waiters noting down the menu from customers, transferring the orders to the kitchen, serving the menu, and finally preparing bills. This process even though looks simple, is prone to human errors while note making & delays involved. So the customers end-up with an unsatisfactory experience. New technologies and approaches are introduced to automate the conventional food ordering process. The systems are PDA (Personal digital assistants) based systems. In the PDA based system, the customers or waiters key in ordering process. When order making completes, these PDAs are to be collected by the waiters to be used by other customers. With wireless technology, the communication between the server and PDA is feasible.

III. PROPOSED MODEL

A tablet PC will be placed on the table. The menu items will be displayed along with their corresponding images and description, so the customer has the choice of selecting the item. Whenever a customer comes to a table, they can select their order with the help of IPSPI40 (4" TFT LCD with capacitive touch screen) provided. This IPS screen in tablet PC is interfaced with a Raspberry pi which is a credit card-sized mini CPU. As soon as the customer selects the item, it will be transmitted via Zig-bee and displayed on the LCD provided in the receiver section which is placed in kitchen.

IV. COMPONENTS

A. HARDWARE COMPONENTS

RASPBERRY PI B+ MODEL

The Raspberry Pi is a single-board computer developed in the UK by the Raspberry Pi Foundation. Raspberry Pi is based on the Broadcom BCM2835 system on a chip with 900 MHz ARM CORTEX A7 CPU with 512 MB RAM. Because it has an ARMv7 processor, it can run full range of ARM/Linux Distributions. It has 4 USB ports, improved power consumption, increased connectivity and 40 GPIO pins.

B. IPSPI40 DISPLAY

It is a 4.0" IPS TFT LCD with touch screen and optional five buttons. It is mainly used for graphical user interface that require good resolution (320x480, 65k colors, 15 frames per second). The IPS-PI40 connects directly to the GPIO connector of the Raspberry Pi. IPS-PI40 uses only SPI pins and one GPIO pin. Optionally additional pin can be used for backlight dimming. The rest of pins are free for other use. The IPS-PI40's driver makes it a primary display for Raspberry Pi which means that there is no need for any special libraries or software to control the LCD. Driver and display support DMA, SPI compression technology and up to 64 MHz SPI clock.

Main Features:

- Resistive touch screen and TSC2046 compatible controller.
- Linux frame buffer driver makes IPS-PI40 a primary display
- Can work parallel with HDMI display.
- Double display possible, independent displays.
- Powered directly from the Raspberry Pi, no additional power required.

C. ZIGBEE MODULES

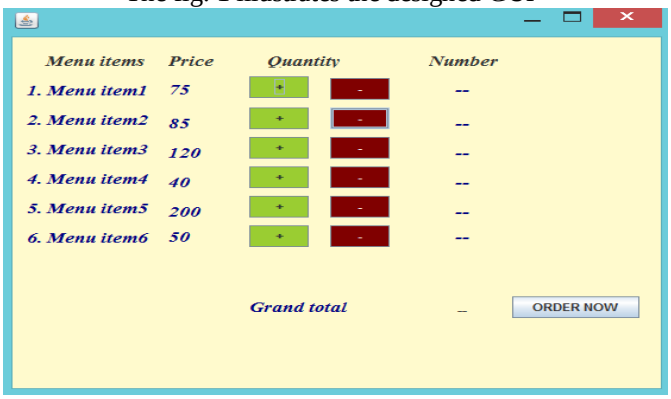
ZigBee is a low-cost, low-power, wireless mesh network standard. The low cost allows the technology to be widely deployed in wireless control and monitoring applications.

Low power-usage allows longer life with smaller batteries. Mesh networking provides high reliability and more extensive range. ZigBee has a self healing network. The technology is intended to be simpler and less expensive than other WPAN's such as Bluetooth. ZigBee chip vendors typically sell integrated radios and microcontrollers with between 60 KB and 256 KB flash memory. ZigBee operates in the industrial, scientific and medical (ISM) radio bands; 868 MHz in Europe, 915 MHz in the USA and Australia, and 2.4 GHz in most jurisdictions worldwide. Data transmission rates vary from 20 to 250 kilobits/second. The ZigBee network layer natively supports both star and tree typical networks, and generic mesh networks. Every network must have one coordinator device, tasked with its creation, the control of its parameters and basic maintenance. Within star networks, the coordinator must be the central node. Both trees and meshes allow the use of ZigBee routers to extend communication at the network.

V. SOFTWARE COMPONENTS

The operating system of Raspberry pi is Raspbian (Debian Wheezy) which is Linux based. Java being platform independent, GUI is made in java with the help of Netbeans IDE.

The fig. 1 illustrates the designed GUI



The next step in software build is making the two Zig-Bees communicate through the designed GUI. This is done with the help of sending data through the serial port of Raspberry pi. XCTU software is used for the purpose of configuring and testing RF products modems.

A. RASPBIAN

Raspbian is a free operating system based on Debian optimized for the Raspberry Pi hardware. An operating system is the set of basic programs and utilities that make your Raspberry Pi run. However, Raspbian provides more than a pure OS: it comes with over 35,000 packages, pre-compiled software bundled in a nice format for easy installation on your Raspberry Pi. However, Raspbian is still under active development with an emphasis on improving the stability and performance of as many Debian packages as possible.

B. XCTU

XCTU is a free multi-platform application designed to enable interaction with Digi RF modules through a simple-to-use graphical interface. XCTU includes all of the tools a developer needs to quickly get up and running with Zig-Bee. Unique features like graphical network view, which graphically represents the XBee network along with the signal strength of each connection, and the XBee API frame builder, which intuitively helps to build and interpret API frames for XBees being used in API mode, combine to make development on the XBee platform easier than ever.

C. JAVA COMMUNICATIONS API

The Java Communications 3.0 API is a Java extension that facilitates developing platform-independent communications applications for technologies such as Smart Cards, embedded systems, and point-of-sale devices, financial services devices, fax, modems, display terminals, and robotic equipment. The Java Communications API (also known as javax.comm) provides applications access to RS-232 hardware (serial ports) and limited access to IEEE-1284 (parallel ports), SPP mode.

VI. WORKING

A. TRANSMITTER SECTION

The transmitter block starts with the user who will browse through the menu items on the IPS screen. Touch feedback will be given by the users which will be processed via Raspberry pi and sent to the Zig-Bee transmitter.

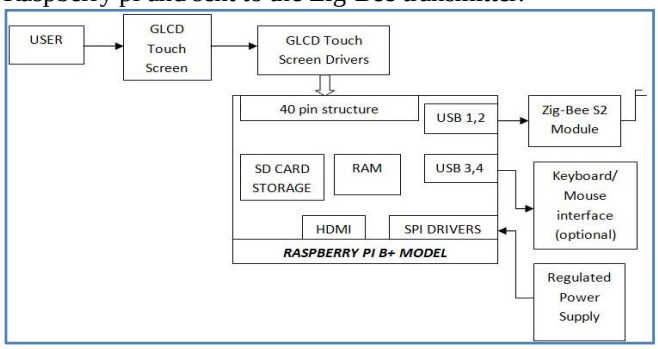


Fig 2 Block diagram of Transmitter section

B. RECEIVER SECTION

The receiver system may be either a PC or again a raspberry pi interfaced LCD screen which displays the order along with the table number from where the order is placed. The receiver system starts with the Zig-bee module that receives the data code bits which is then sent to the java application to display the order (conversion from binary to string data) on the admin PC.

Basic receiver block diagram is as shown:

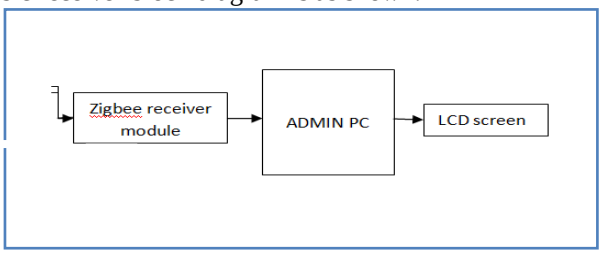


Fig 3 Block diagram of Receiver section

VII. CONCLUSION

Hence we have successfully implemented the SMART MENU using a Raspberry pi. Each module's presence has been reasoned out and placed. We are aiming for reduction in the cost of the project best possible with higher efficiency. We are further modifying the project for giving a feedback once the order is placed and smart payment options via various debit/credit cards. Also modifications can be done in setting up Smart Advertisement on the screen which can add to the economic benefits.

REFERENCES

[1] Wan-Ki Park, I. Han, KR Park, "ZigBee based dynamic control scheme for multiple legacy IR controllable digital consumer devices", IEEE Transactions on Consumer Electronics, vol.53, no.1, 2007, pp.172-177.
[2] Karlof, C., Wagner, D.: Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures, First IEEE International Workshop on Sensor Network Protocols and Applications, May 2003. Page 16.
[3] Dr. Robert B. Reese, A ZigBee-subset/IEEE 802.15.4 Multi-platform Protocol Stack website http://www.ece.msstate.edu/reese
[4] Broadcom Corporation. "BCM2835 ARM Peripherals" General purpose I/o (GPIO) page 90 February 2012 edition.
[5] Raspberry Pi Model B+ http://www.raspberrypi.org/products/model-b-plus
[6] Java Communications AI http://www.oracle.com/technetwork/java/index-jsp-141752.html

