



An Automated Cooking Robot Using Embedded C

Prakruthi K.S¹, Raksha S²., Saroja N Patgar³, Sindhu M.S⁴, Dr. R. Kanagavalli.⁵

Final year B.E Students (UG), Department of Information Science & Engineering, The Oxford College of Engineering,
Bangalore, India^{1,2,3,4}

Assistant Professor and HOD, Department of Information Science & Engineering, The Oxford College of Engineering,
Bangalore, India⁵

ABSTRACT: Electric household appliances are now an indispensable component of our life. Microwave ovens cook our meals, dishwashers clean the dishes and laundry machines wash and dry our clothes. However, most of these machines operate in a closed environment (box) and it significantly limits what they can do. We foresee that next generation electric household appliances (systems) should execute more advanced tasks in an open environment that is shared with users. We are developing a cooking system as an initial example of such household systems. Cooking in a closed environment consumes too much space for typical home kitchens. We therefore, designed a system that works in an open environment. When using the system, the user puts pre-processed cooking ingredients on the table and has small robots execute the cooking tasks using a pot on an induction heating (IH) cooker. When not using the system, the user can cook on the same table using the same cooker. Consideration to human factors is critically important for the successful deployment of such open household systems. They must be safe and able to adapt to dynamic changes of the environment because they share the space with the user. They also must provide an appropriate user interface for controlling complicated real-world tasks. We address these issues in the design of our cooking system.

KEYWORDS: Cooking robot, Internet of things, Android application, automation, intelligent systems, robotics, cooking systems, Arduino.

I. INTRODUCTION

We propose a cooking system that operates in an open environment. The system cooks a meal by pouring various ingredients into a boiling pot on an induction heating cooker and adjusts the heating strength according to the user's instructions. We then describe how the system incorporates robotic- and human-specific elements in a shared workspace so as to achieve a cooperative rudimentary cooking capability. First, we use small mobile robots instead of built-in arms to save space, improve flexibility and increase safety. Second, we use detachable visual markers to allow the user to easily configure the real-world environment. Third, we provide a graphical user interface to display detailed cooking instructions to the user. We hope insights obtained in this experiment will be useful for the design of other household systems in the future.

The Kitchen is an essential part of human life. Cooking food is considered as special skill. Cooking food in present day busy life is a tedious work. For a busy life schedule, solutions such as restaurants, packaged food, ready to eat food came into existence. These solutions turned out to high profit and medium scale industries. The quality of food and adulteration became major factors in restaurants, addition of chemicals for long lasting food in packaged and ready to eat food products may affect the health of human-life. And the taste in these foods are depended on the chief chef of the particular industry. In this paper, authors describes a mechanism to overcome the above-mentioned problem with a solution of an autonomous cooking system which cooks the desired dishes in the home kitchen which can be controlled from android application.

Present day kitchen automation system includes features like smart notifications, scheduled notifications, smart displays, prediction of dishes from the ingredients available [1]. But most of these methods are software oriented. Most of the commercially available devices, which is used to cook are specific to one dish and are costly for use in home and medium scale applications. The mechanism discussed in this paper integrates device that is capable of cooking multiple recipes and can be cooked on a single pan using up to nine or more ingredients with churning, cooking, frying mechanisms and smart weigh scale.



II. PROPOSED SYSTEM

We are creating a special table and placed it higher than the pot so that the robots can easily pour ingredients and seasonings. We also created special plates so that the robots can handle them and the user can place a visual marker on it. Inside the IH cooker we implanted a micro controller circuit and Bluetooth communication module for controlling the temperature level remotely. We carefully designed the environment so that the user can also cook without using the system. The user can use the table as a working space and put ingredients on the same plates. We are developing three types of customized small iot robots, one for transporting the ingredients on plates, one for transporting seasonings in bottles, and one for stirring the pot. The first robot grabs a plate using a single arm, moves to the pot, and tilts the plate to drop the ingredients into the pot. The second robot grabs a bottle by using a hand with two fingers, moves to the pot, and shakes the bottle to sprinkle the seasoning into the pot. The stirring robot stirs the pot at appropriate times. Before cooking begins, the user needs to select and attach a cooking utensil to the robot. We are developing a cooking system as an initial example of such household systems. Cooking in a closed environment consumes too much space for typical home kitchens. When not using the system, the user can cook on the same table using the same cooker. This robot can cook egg boil, rice, coffee, tea and vegetables item etc.

Advantages:

- It works really well
- It looks fantastic
- Helpful for elder people
- They make it easier/help people who are stressed

III. METHODOLOGY

In our work we use four modules, these modules are listed below.

➤ Chopping Vegetables:

Vegetable cutting dicing is a daily domestic task which takes up a lot of time and effort. So here we propose a motorized vegetable cutting machine that does this in a fully automatic manner. The vegetable cutter machine consists of a narrow feeder which is attached to a motorized blade cutter arrangement. The blade is contained in a circular transparent vessel to collect cut vegetables. The system consists of a narrow feed pipe used to insert vegetables. This feed pipe is fabricated to the cutting blade area. As soon as the vegetable is pushed through it rotating blade which is connected to a powerful motor through a custom designed coupling is continuously rotating as per the speed we set for it. The motor mount is fabricated with the lower collector vessel which is used to collect the diced vegetable pieces.

➤ Process Request:

The users can select the cook/chef as per their interest. The cook might specialize in preparing Indian or maybe something else. Under this section, the users can send the request to the desired cook as per their preferences.

➤ Cook Process:

When Robot Cook gets instruction from the user then Robot cook will be working automatically. Like lemon-rice -It takes around 20 minutes for the meal to be cooked and the entire procedure is completely unaided. The recipes can be pre-programmed into the bot and after loading the required ingredients; the bot functions independently.



➤ **Android Application:**

In this paper, the cooking robot is connected with the android app through the low power hc05 (Bluetooth) technology based on the technology of the internet of things. User are able to use android app to give the instructions to the Arduino microcontroller.

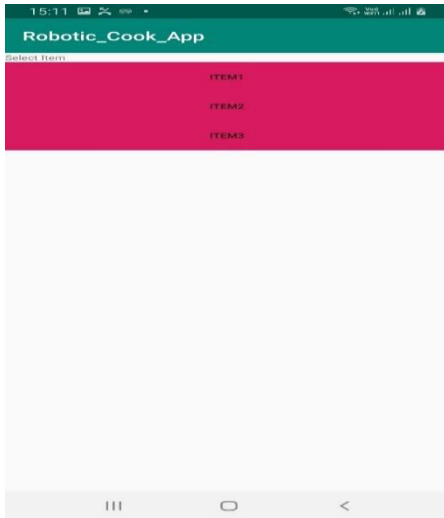


Fig1 application pg1

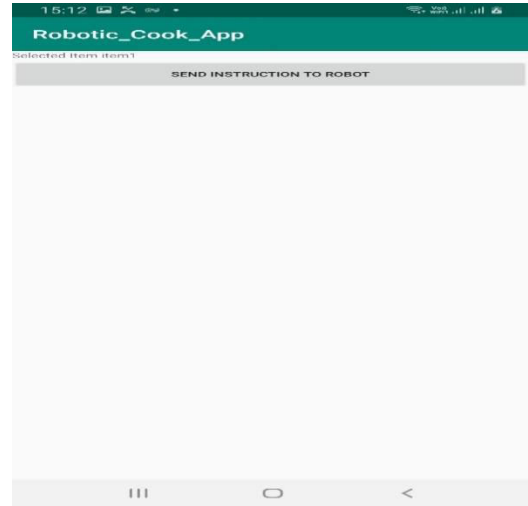


Fig 2 application pg2

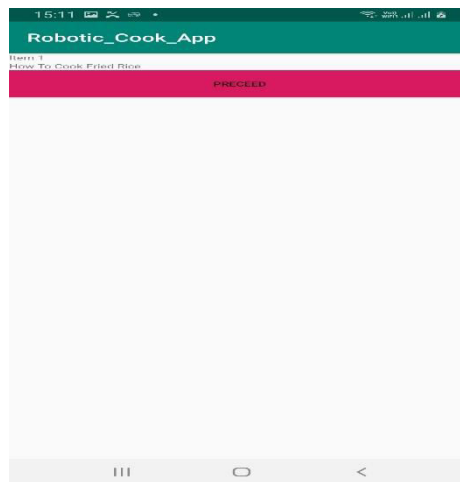


Fig 3 application pg 3

SYSTEM DESIGN AND MODULE DESCRIPTION

◆ **INTRODUCTION**

A good system design is to organise the program modules in such a way that are easy to develop and change. Structured design techniques help developers to deal with the size and complexity of programs. Analysts create instructions for the developers about how code should be written and how pieces of code should fit together to form a program.

◆ **SYSTEM ARCHITECTURE**

The architecture of a system describes its major components, their relationships (structures), and how they interact with each other. Software architecture and design includes several contributory factors such as Business strategy, quality attributes, human dynamics, design, and IT environment. We can segregate Software Architecture and Design into two distinct phases: Software Architecture and Software Design. In Architecture, non-functional decisions are cast and separated by the functional requirements. In Design, functional requirements are accomplished.

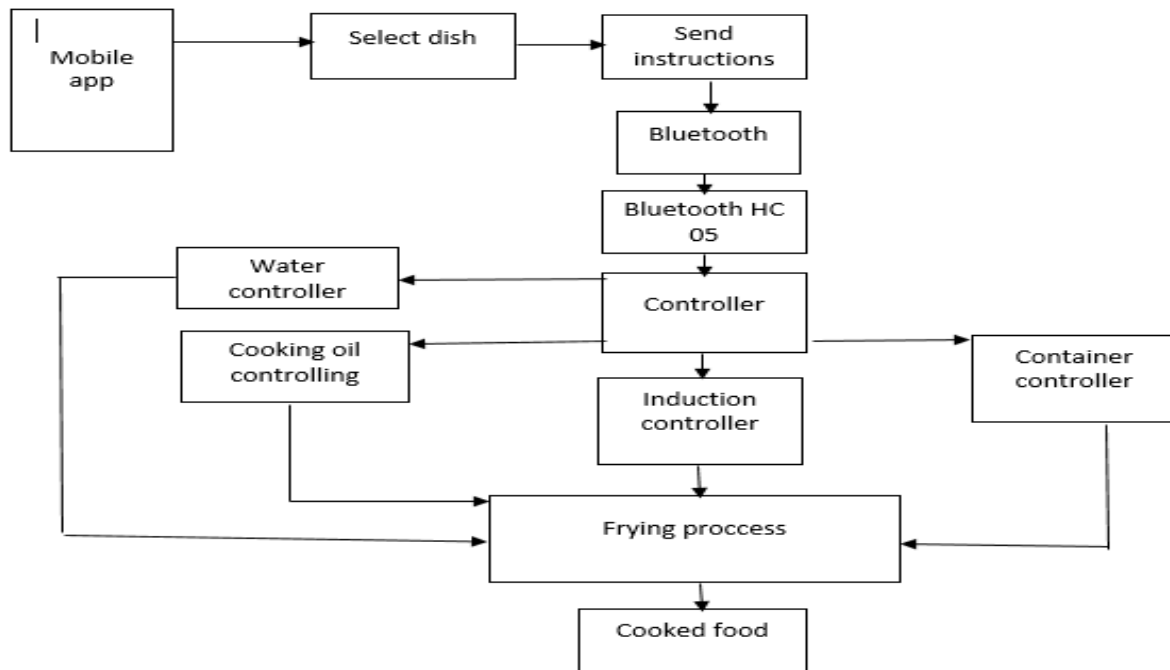


FIG 4. SYSTEM ARCHITECTURE

IV. DESIGN CONSIDERATION

◆ REQUIREMENT SPECIFICATION

A software requirements specification (SRS) is a document that captures complete description about how the system is expected to perform. It is usually signed off at the end of requirements engineering phase. Framework Requirement Specification (SRS) is a focal report, which outlines the foundation of the item headway handle. It records the necessities of a structure and in addition has a delineation of its noteworthy highlight. A SRS is basically an affiliation's seeing (in making) of a customer or potential client's edge work necessities and conditions at a particular point in time (for the most part) before any veritable design or change work. It's a two-way insurance approach that ensures that both the client and the affiliation understand exchange's necessities from that perspective at a given point in time.

◆ SOFTWARE REQUIREMENT

ANDROID DESCRIPTION

Android applications are written in the Java programming language. The Android SDK tools compile the code along with any data and resource files into an Android package, an archive file with an .apk suffix. All the code in a single .apk file is considered to be one application and is the file that Android-powered devices use to install the application. Once installed on a device, each Android application lives in its own security sandbox. The Android operating system, is a multi-user Linux system in which each application is a different user.

ARDUINO IDE

The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development boards.

The source code for the IDE is released under the GNU General Public License, version 2.

The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures.

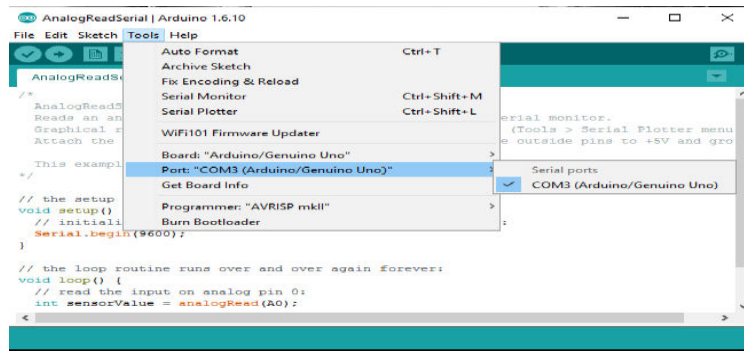


Fig 5.Arduino ide

BLUETOOTH HC05

HC05 module has an internal 3.3v regulator and that is why you can connect it to 5v voltage. But we strongly recommend 3.3V voltage, since the logic of HC05 serial communication pins is 3.3V. Supplying 5V to the module can cause damage to the module. In order to prevent the module from damages and make it work properly, you should use a resistance division circuit (5v to 3.3v) between Arduino TX pin and module RX pin.

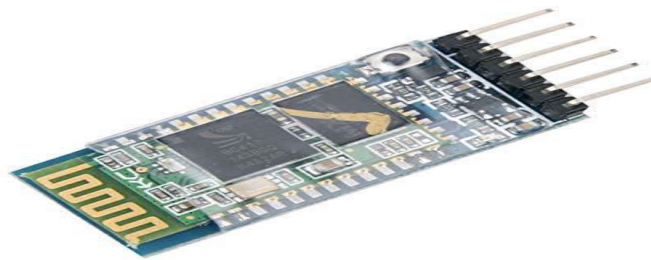


Fig 6. Bluetooth hc 05

ANDROID APPLICATION

Android applications are written in the Java programming language. The Android SDK tools compile the code along with any data and resource files into an Android package, an archive file with an .apk suffix. All the code in a single .apk file is considered to be one application and is the file that Android-powered devices use to install the application. Once installed on a device, each Android application lives in its own security sandbox: The Android operating system, is a multi-user Linux system in which each application is a different user. By default, the system assigns each application a unique user ID (the ID issued only by the system and is unknown to the application). The system sets permissions for all the files in an application so that only the user ID assigned to that application can access them. Each process has its own virtual machine (VM), so an application's code runs in isolation from other applications. By default, every application runs in its own Linux process. Android starts the process when any of the application's components need to be executed, then shuts down the process when it's no longer needed or when the system must recover memory for other applications.

V. RESULTS ANALYSIS

We proposed a cooking system in a shared cooking environment with the user, called Cooky. User can send the instruction to Arduino (smart robotic cooking frame) Smart robotic cooking frame can receive the instruction from mobile app. This robot can cook rice, coffee, tea and vegetables item etc. The user gives cooking instructions to the system such as timing for the ingredients adding, stirring a pot, and controlling an electric cooker. Also, we developed small robots for the actual cooking tasks. The robots perform their tasks according to the user's instructions. We tested the system and confirmed that it successfully cooked a meal with given instructions.



Fig .11(a)Final prototype

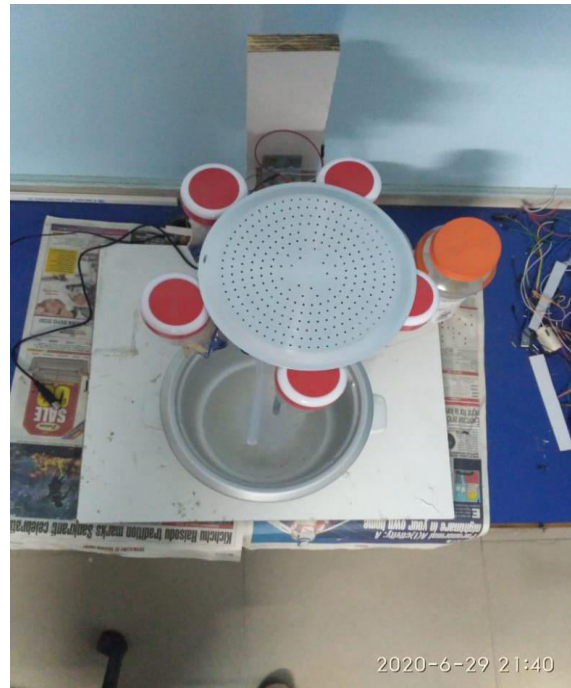


Fig .12(b) Final prototype

VI. FUTURE SCOPE OF THIS STUDY

- To reduce the man power
- Make cooking process automated.
- To maintain the test food quality
- Robotic cook for disable person, elder people, fast-food restaurant who doesn't know how to cook
- They make it easier/help people who are stressed
- Vegetable cutting dicing is a daily domestic task which takes up a lot of time and effort. So here we propose a motorized vegetable cutting machine that does this in a fully automatic manner.
- The users can select the cook/chef as per their interest. The cook might specialize in preparing Indian or maybe something else.
- smart robotic cook is able to manage automatically

VII. CONCLUSION

We proposed a cooking system in a shared cooking environment with the user, called Cooky. The system uses recipes, which include cooking instructions and visual markers; for simple cooking tasks. The user gives cooking instructions to the system such as timing for the ingredients adding, stirring a pot, and controlling an electric cooker. Also, we developed small robots for the actual cooking tasks. The robots perform their tasks according to the user's instructions. We tested the system and confirmed that it successfully cooked a meal with given instructions. The design result is including system design machines, physical machines design, also interface designed that will be embedded in the machine. Later, if this system implemented successfully, this machine will be very helpful in terms of shortening the cooking process. This machine is expected can work automatically so that users do not need to wait from start to finish in cooking process.



REFERENCES

1. Forlizzi, J. How robotic products become social products: an ethnographic study of cleaning in the home. In Proc. HRI2007. ACM (2007), 129-136.
2. Guo, C. and Sharlin, E. Exploring the use of tangible user interfaces for human-robot interaction: a comparative study. In Proc. CHI 2008. ACM (2008), 121-130.
3. Hamada, R., Okabe, J., Ide, I., Satoh, S., Sakai, S., and Tanaka, H. Cooking navi: assistant for daily cooking in kitchen. In Proc. MULTIMEDIA 2005. ACM (2005), 371-374.
4. Kato, F., Shiina, M., Tokizaki, T., Mitake, H., Aoki, T., and Hasegawa, S. Culinary art designer. In Proc. ACE 2008, vol. 352. ACM (2008), 398-398.
5. Kemp, C. C., Anderson, C. D., Nguyen, H., Trevor, A. J., and Xu, Z. A point-and-click interface for the real world: laser designation of objects for mobile manipulation. In Proc. HRI 2008, (2008), 241-248.
6. Nakauchi, Y., Fukuda, T., Noguchi, K., and Matsubara, T. Time Sequence Data Mining for Cooking Support Robot, In Proc. CIRA 2005, No.We-B3-4, 2005
7. Okada, K., Ogura, T., Haneda, A., Fujimoto, J., Gravot, F., and Inaba, M. Humanoid motion generation system on HRP2-JSK for daily life environment. In Proc. ICMA 2005, IEEE (2005), Vol. 4, 29, 1772-1777.
8. Sakamoto, D., Honda, K., Inami, M. and Igarashi, T. Sketch and run: a stroke-based interface for home robots. In Proc. CHI 2009, ACM (2009), 197-200