



IOT Embedded Module-Based Automatic Hot Milk Overflow Avoidance and Alerting System

Syed Munawwar¹, Bhuma Manoj Kumar², Shaik Nabi Rasool³, Koyalathota Santhosh Reddy⁴, Banala Venkata Prashanth⁵, Pasupula Tarun Kumar⁶

^{1,2,3,4,5,6}Department of Electronics and Communication, Santhiram Engineering College, India

ABSTRACT

The spilling of milk when boiling from the vessel is a common problem in most of the households. A simple liquid temperature sensor is employed to measure the temperature of the milk and given to an embedded microcontroller to actuate a servo motor (or DC motor) to turn off the knob of the stove. The information is sent to a server through an IoT module. For a given vessel used several interesting analytics can be made in the collected data.

Keywords: Sensors, IoT, Embedded, NodeMCU

1. INTRODUCTION

The overflowing of milk on the stove is a very common problem in Indian households. The most dangerous scenario that can happen because of this incident is the extinguishing of the stove fire that may later result in a fire accident. Apart from that it is also a waste of precious LPG gas fuel when the milk is allowed to overflow. Home and kitchen automation is a growing sector with a projected market value of 170 billion US dollars approximately by 2025. A simple way proposed here is the placement of an external non-contact heat sensor outside the milk boiling vessel to detect the temperature and actuate a servo motor that is fixed to the stove knob. The reason for using a non-contact sensor is to avoid any chemical reaction between a contact sensor with the milk at high temperature. The scope for developing a lot of such products is very simple: to choose an appropriate sensor to identify the required level of cooking and then consequently turn off the heat source by an actuator.

2. PROPOSED SYSTEM

In proposed system consist of LM35 sensor, a NodeMCU connected with ThingSpeak cloud platform. The LM35 is a temperature sensor outputs an analog signal which is proportional to instantaneous temperature which detects the temperature and transmits it to the NodeMCU ESP8266 which is a low power Wi-Fi solution, microcontroller. This NodeMCU is programmed to observe the sensor reading and further processing for buzzer activation. The entire program is compiled using Arduino IDE. Additionally, ThingSpeak, a cloud platform is enabled where the NodeMCU is connected with the cloud using a API. IFTTT, a communication service provider is linked with the ThingSpeak API so that an email is generated from IFTTT through the data it observed from the ThingSpeak and thus the email is communicated to the end user.

3. ARDUINO MICROCONTROLLER

The microcontroller used to implement this project and similar projects are Arduino boards. Presented below is an outline of Arduino family boards. [2]

3.1 Introduction

Arduino is an open-source hardware and software company, project and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices. It's hardware products are licensed under a CC-BY-SA license, while software is licensed under the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL), permitting the manufacture of Arduino boards and software distribution by anyone. Arduino boards are available commercially from the official website or through authorized distributors.[2]

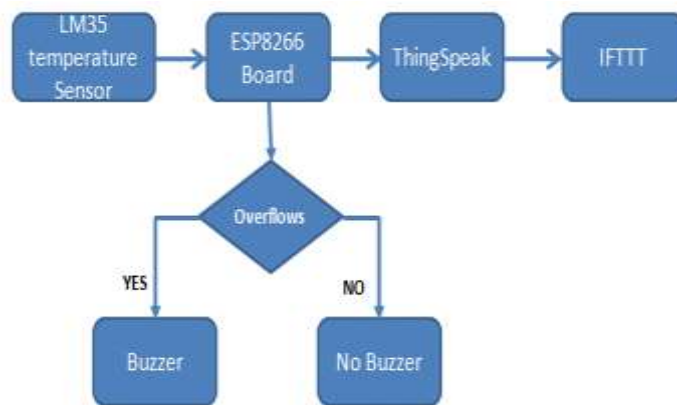


Figure 1. Block Diagram

Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards ('shields') or breadboards (for prototyping) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs. The microcontrollers can be programmed using the C and C++ programming languages, using a standard API which is also known as the Arduino language. In addition to using traditional compiler tool chains, the Arduino project provides an integrated development environment (IDE) and a command line tool (arduino-cli) developed in Go.[2]

The Arduino project began in 2005 as a tool for students at the Interaction Design Institute Ivrea in Ivrea, Italy, aiming to provide a low-cost and easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. Common examples of such devices intended for beginner hobbyists include simple robots, thermostats and motion detectors. The name Arduino comes from a bar in Ivrea, Italy, where some of the founders of the project used to meet. The bar was named after Arduin of Ivrea, who was the margrave of the March of Ivrea and King of Italy from 1002 to 1014"[2]

3.2 Hardware

Arduino is open-source hardware. The hardware reference designs are distributed under a Creative Commons Attribution Share-Alike 2.5 license and are available on the Arduino website. Layout and production files for some versions of the hardware are also available.[2]

Although the hardware and software designs are freely available under copy left licenses, the developers have requested the name Arduino to be exclusive to the official product and not be used for derived works without permission. The official policy document on use of the Arduino name emphasizes that the project is open to incorporating work by others into the official product. Several Arduino-compatible products commercially released have avoided the project name by using various names ending in -duino.[2]

An early Arduino board with an RS-232 serial interface (upper left) and an Atmel ATmega8 microcontroller chip (black, lower right); the 14 digital I/O pins are at the top, the 6 analog input pins at the lower right, and the power connector at the lower left.[2]

Most Arduino boards consist of an Atmel 8-bit AVR microcontroller (ATmega8, ATmega168, ATmega328, ATmega1280, or ATmega2560) with varying amounts of flash memory, pins, and features. The 32-bit Arduino Due, based on the Atmel SAM3X8E was introduced in 2012. The boards use single or double-row pins or female headers that facilitate connections for programming and incorporation into other circuits. These may connect with add-on modules termed shields. Multiple and possibly stacked shields may be individually addressable via an I²C serial bus. Most boards include a 5 V linear regulator and a 16 MHz crystal oscillator or ceramic resonator. Some designs, such as the Lily Pad, run at 8 MHz and dispense with the onboard voltage regulator due to specific form-factor restrictions.[2]

Arduino microcontrollers are pre-programmed with a boot loader that simplifies uploading of programs to the on-chip flash memory. The default boot loader of the Arduino Uno is the Optibootbootloader. Boards are loaded with program code via a serial connection to another computer. Some serial Arduino boards contain a level shifter circuit to convert between RS-232 logic levels and transistor-transistor logic (TTL) level signals. Current Arduino boards are programmed via Universal Serial Bus (USB), implemented using USB-to-serial adapter chips such as the FTDI FT232. Some boards, such as later-model Uno boards, substitute the FTDI chip with a separate AVR chip containing USB-to-serial firmware, which is reprogrammable via its own ICSP header. Other variants, such as the Arduino Mini and the unofficial Boarduino, use a detachable USB-to-serial adapter board or cable, Bluetooth or other methods. When used with traditional microcontroller tools, instead of the Arduino IDE, standard AVR in-system programming (ISP) programming is used. An official Arduino Uno R2 with descriptions of the I/O locations. The Arduino board exposes most of the microcontroller's I/O pins for use by other circuits. The Diecimila, Duemilanove, and current Uno provide 14 digital I/O pins, six of which can produce pulse-width modulated signals, and six analog inputs, which can also be used as six digital I/O pins. These pins are on the top of the board, via female 0.1-inch (2.54 mm) headers. Several plug-in application shields are also commercially available. The Arduino Nano, and Arduino-compatible Bare Bones Board and Boarduino boards may provide male header pins on the underside of the board that can plug into solderless breadboards.[2]

Many Arduino-compatible and Arduino-derived boards exist. Some are functionally equivalent to an Arduino and can be used interchangeably. Many enhance the basic Arduino by adding output drivers, often for use in school-level education, to simplify making buggies and small robots. Others are electrically equivalent, but change the form factor, sometimes retaining compatibility with shields, sometimes not. Some variants use different processors, of varying compatibility.[2]

3.3 Software

A program for Arduino hardware may be written in any programming language with compilers that produce binary machine code for the target processor. Atmel provides a development environment for their 8-bit AVR and 32-bit ARM Cortex-M based microcontrollers: AVR Studio (older) and Atmel Studio (newer).[2]

IDE:

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, and Linux) that is written in the Java programming language. It originated from the IDE for the languages Processing and Wiring. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. The source code for the IDE is released under the GNU General Public License, version 2.[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. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub `main()` into an executable cyclic executive program with the GNU tool chain, also included with the IDE distribution. The Arduino IDE employs the program `avrdude` to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.[2]

Pro IDE:

On October 18, 2019, Arduino Pro IDE (alpha preview) was released. The system still uses Arduino CLI (Command Line Interface), but improvements include a more professional development environment, auto completion support, and Git integration. The application frontend is based on the Eclipse Theia Open Source IDE. The main features available in the alpha release are:[2]

Sketch:

A sketch is a program written with the Arduino IDE.[64] Sketches are saved on the development computer as text files with the file extension `.ino`. Arduino Software (IDE) pre-1.0 saved sketches with the extension.[2]

A minimal Arduino C/C++ program consists of only two functions:

Setup ():

This function is called once when a sketch starts after power-up or reset. It is used to initialize variables, input and output pin modes, and other libraries needed in the sketch. It is analogous to the function `main ()`.

loop ():

After `setup ()` function exits (ends), the `loop()` function is executed repeatedly in the main program. It controls the board until the board is powered off or is reset. It is analogous to the function `while`.[2]

4. INDUSTRY 4.0

Wikipedia defines Industry 4.0 as thus: The Fourth Industrial Revolution (4IR or Industry 4.0) is the ongoing automation of traditional manufacturing and industrial practices, using modern smart technology. Large-scale machine-to-machine communication (M2M) and the internet of things (IoT) are integrated for increased automation, improved communication and self-monitoring, and production of smart machines that can analyze and diagnose issues without the need for human intervention.[3]

Automation under Industry 4.0 has a particular schema or pattern at its outset. Presented below is how automation in the mass production industry as well as consumer level products are built in today's technological era.

The schema presented in Figure 2 has a lot of other components involved but the generic outline of it stands justifiable for all kinds of automation today.

The software automaton of the conventional automation model, which is the status quo, was built by a human expert or a team of human experts till now. With the advent of machine learning technology, the software automaton was not fully directly designed by human experts. The human experts build the machine learning software and give the real world data set as training information. The machine learning software identifies the pattern between the input and the output parameters of the dataset in the form of a mathematical model. This mathematical model can be downloaded as a working software module to other electronic computing devices. This mathematical model is referred to as the 'trained machine learning module'. The software automaton of all the current digital embedded devices is a mathematical model that gives a numerical output for a numerical input based on arithmetic and logical conditions. This software automaton, as explained above can be either directly developed by a set of human experts by means of setting the boundary conditions themselves based on observation and requirement or can be downloaded as an executable module from machine learning training systems that are trained with relevant dataset. In whatever way the software automaton is developed, it can be loaded onto the relevant embedded computing module that can be used for either sensor based closed loop automation or open loop automation.

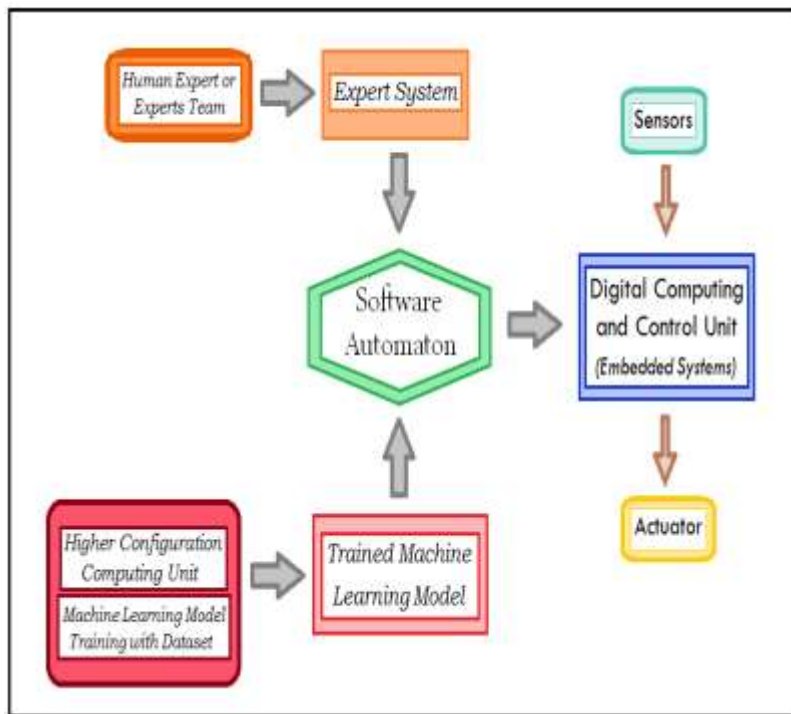


Figure 2. Schema of Automation

The technological components of Industry 4.0 includes IoT, augmented reality, virtual reality, cloud computing, 3D printing, big data analytics, networking, data security, human-machine interaction and others. IoT is a very effective way to collect real world data. Sensors integrated with data acquisition and transmission systems can be placed anywhere and the collected data can be pre-processed if required and used as datasets to train machine learning models.

Cloud computing is employed for optimized utilization of computing resources. There are many third party vendors like Google and Amazon which are very reliable in terms of data security and speed of computation. These services offer companies and organizations a cheap and reliable way to harness the power of artificial intelligence and machine learning.

Big data analytics is the set of technological components involved with collecting, collating and managing large quantities of data for analytics and decision making. When so much data is involved, especially with third party service providers, data security plays an important role.

One of the paramount concerns about Industry 4.0 is the unemployment it can create due to powerful automations. The field of human-machine interactions and co-working has been a very developing field now to mitigate the above mentioned problem.

5. WORKING

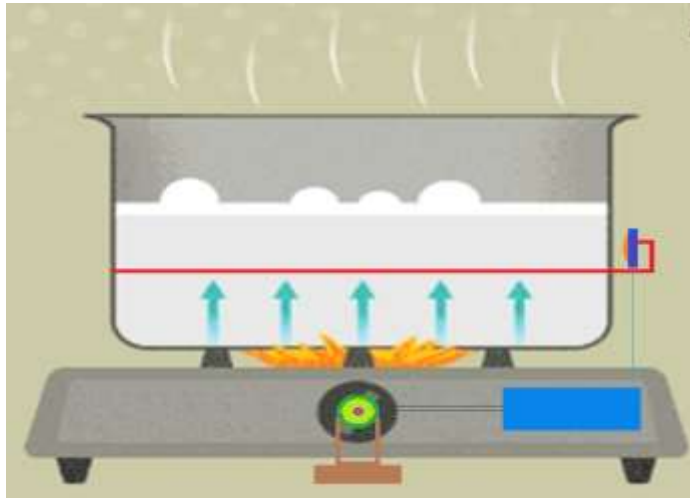


Figure 3. Schematic Diagram

As shown in the above schematic diagram (Figure 3), the non-contact temperature sensor is fixed outside the vessel with a 3D printed ring fitted onto it. The temperature sensor output is given to the microcontroller and the battery unit, which controls the servo motor that turns off the stove knob.



Figure 4. Milk Vessel 1



Figure 5. Milk Vessel 2

Presented in Figure 4 and 5 are two examples of common milk boiling vessels. It can be observed that the one that is taller has more space to have a 3D printed ring fitted on it.

6. RESULTS AND DISCUSSION

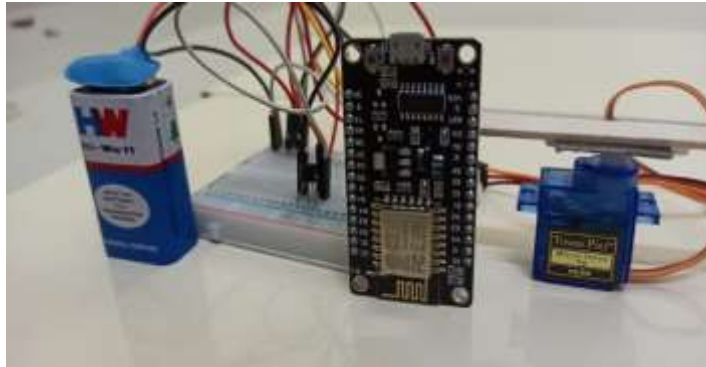


Figure 6. Arduino Setup

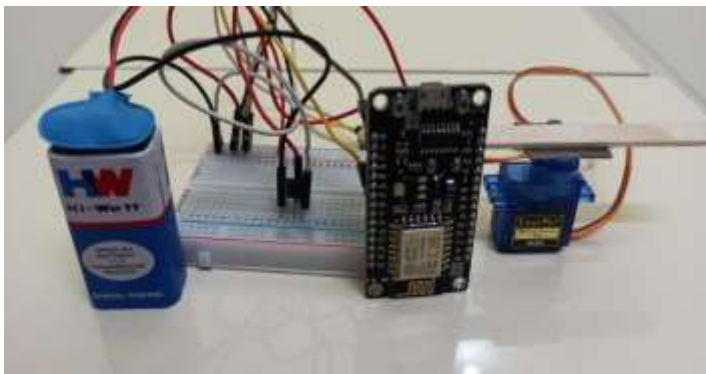


Figure 7. Output Image Top view



Figure 8. Status Output

As shown in the Figure 6 and 7, the circuit of the system consists of a temperature measuring sensor, a Node MCU and a servo motor. This servo motor is connected to the knob of the gas stove in the industrial grade product. The sensor detects the heat of the container. The sensor is placed outside in a non-contact manner by being mounted on a 3D printed ring that can be fastened to the vessel in the industrial grade product version. The IoT enabled Node MCU sends a message mentioning the exact time and date along with the temperature recorded to the registered smart device as shown in Figure 8. Most of these message generating services for IoT are open source platform services. When being implemented for industrial grade commercial products, then a paid service of the open source platform can be utilized for better options that include data analytics.

7. CONCLUSION AND FUTURE WORK

There are two areas in which this particular presented product can be improved: one in the sensor area where the boiling of the milk has to be identified and other is the actuation of the knob of the gas stove. In the area of the sensor, the external sensor is placed so that no plastic material comes in contact with the boiling milk allowing the possibility of any plastic impurities being mixed with the milk. That is the reason why the sensor is placed outside and the temperature of the vessel is being measured. Therefore, to develop a long needle metallic sensor that is chemically inert will be the perfect solution

for food products temperature sensing and subsequently automation. In the actuation, fixing the knob inside the stove so that whenever the automated system is actuated, then it can actuate. If the automated system is not activated, then it should be manually turnable. In this paper, the data of the boiling is connected and via IoT, it is updated to a server. This gives data of milk consumption from household to household. In this particular product, the data of when the temperature goes up is measured and transmitted to the open source server. A lot of data analytics and decision making can be done based on that.

References

- [1]. Priyanka R Chaudhari, Pooja V Kakade, S. I. Patel and Ahesanvarish Shaikh, "Effect of temperature on performance of qualitative tests for detection of common adulterants and preservatives in milk", *International Journal of Dairy Science*, 71(5):441-446, May 2019.
- [2]. <https://en.wikipedia.org/wiki/Arduino>
- [3]. https://en.wikipedia.org/wiki/Fourth_Industrial_Revolution