# Arduino Based Smart Parking System

**Mr. Kush Shah[1], Ms. Priya Chaudhari[2]**

[1]*Final year Student B.E., Department of Computer engineering, SVIT Vasad, Gujarat, India*
[2]*Final year Student B.E., Department of Computer engineering, SVIT Vasad, Gujarat, India*

---------------------------------------------------------------------***---------------------------------------------------------------------

**Abstract –** *"Creativity is inventing, experimenting, growing, taking risks, breaking rules, making mistakes and having fun." Quoted by Mary Lou Cook, encouraged us to think on such a topic which can make our lives easier by overcoming the most common problem faced by the whole world. Sometimes, it is very difficult to find a suitable parking place in parking lot. We have proposed a suitable solution to this problem. Arduino has already made a huge impact on learning. The widespread acceptance gave the open source hardware a new life, potentially challenging many industrial products and new interests in hardware prototyping and electronics. Recent transition has been made from 8 bit to 32 bits and it is expected that in future Arduino may be seen in form of a cheap practical computer. You can also presently make custom cellphones using open source boards like the Arduino. Looking at this actionable advancement towards Arduino, allured us to make systematic and organized parking system for vehicles, using Arduino Uno.*

***Key Words***: Ultrasonic Sensors, Arduino,

## 1. ARDUINO

Arduino is basically an open-source computer hardware/software platform for building digital devices and interactive objects that can sense and control the physical world around them. It's intended for artists, designers, hobbyists, and anyone interested in creating interactive objects or environments. It comes with an open supply hardware feature that permits users to develop their own kit. The software of the Arduino is well-suited to all kinds of operation systems like Linux, Windows, and Macintosh, etc. It also comes with open supply software system feature that permits tough software system developers to use the Arduino code to merge with the prevailing programing language libraries and may be extended and changed. For beginners, it is very simple to use and also cheap. It can be used to create such devices that can interact with the environment using sensors and actuators. Some common examples include robot, thermostats and motion detectors. This paper introduces the use of Arduino in one such area, that is, Smart Parking System, so that people can become aware of where free parking place is available and save time, while avoiding traffic congestion.

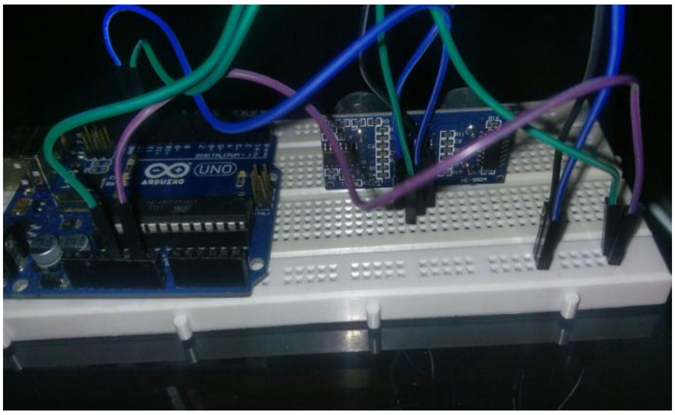


**Fig -1:** Arduino

### 1.1 Functions

- Digital read pin reads the digital value of the given pin.
- Digital write pin is used to write the digital value of the given pin.
- Pin mode pin is used to set the pin to I/O mode.
- Analog read pin reads and returns the value.
- Analog write pin writes the value of the pin.
- Serial begins pin sets the beginning of serial communication by setting the rate of bit.

## 2. Overview

This project's main purpose is to produce a real life solution to the car parking problem which the whole world is facing frequently. People usually roam around in the parking lots trying to find a suitable place to park in. To solve that problem we have created the automatic car parking system, using an open source hardware, programmable sensors and the use of computers to provide an interface to understand the digital output produced.

## 3. Components and Compatibility

The basic components used in our system are as follows:

1) Ultrasonic Sensors (x6)
The Arduino Ultrasonic Range Detection Sensor is used with Arduino in order to calculate distances from objects. It's an IC (Integrated Circuit) that's works by sending an ultrasound pulse of about 40KHz. And then it waits and listens for the

pulse to echo back, by calculating the time taken in microseconds.

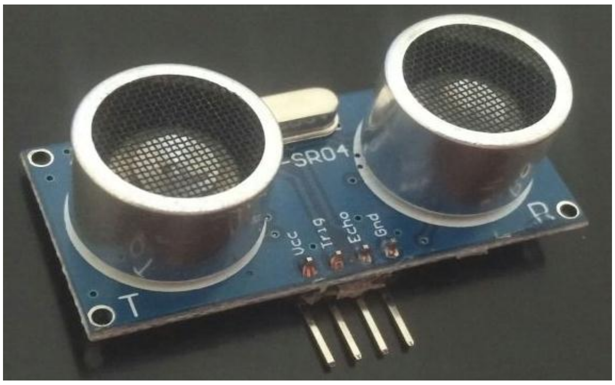


**Fig-2:** Ultrasonic Sensors

2) Arduino Uno (x1)
The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 6 analog inputs, 14 digital input/output pins (of which 6 can be used as PWM outputs), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; all we have to do is simply connect it to PC with a USB cable or power it with an AC-to-DC adapter or battery to get started.

**Fig-3:** Arduino Uno

3) Wires(x30)
Wires are nothing but single, cylindrical, flexible strands of metal. They are usually used to bear loads of electric or telecommunication signals.

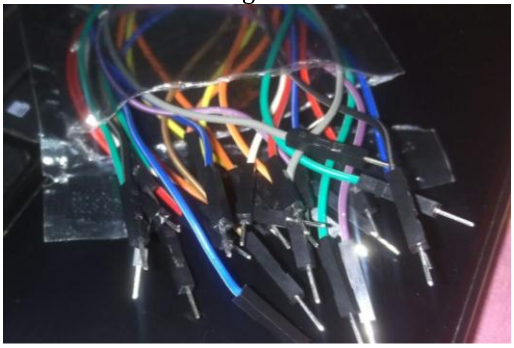


**Fig-4:** Wires

4) Breadboard(x1)
While building electric circuits, breadboards are the most fundamental pieces.

**Fig-5: Breadboard**

## 4. Hardware Connections

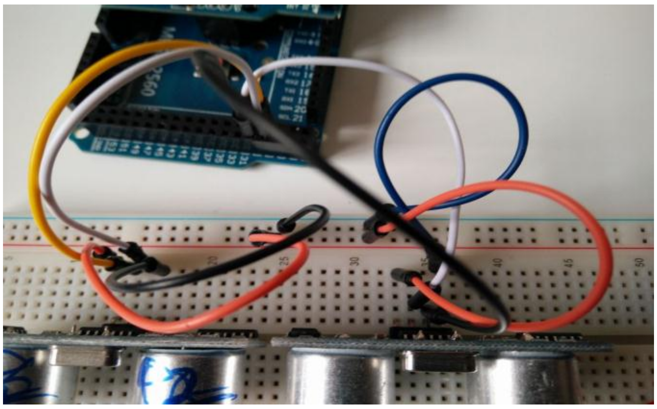
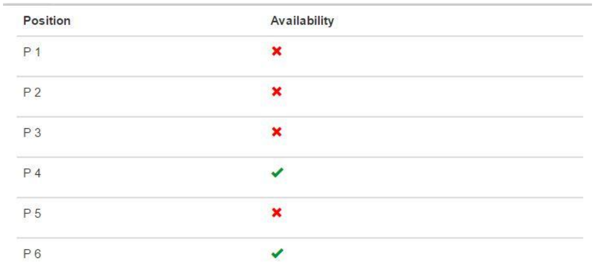


**Fig- 6:** Hardware connection

- Connect the Vcc pin to the positive rail on your breadboard.
- Connect the Gnd pin to the negative rail on your breadboard.
- Connect the Trig pin to any digital pin on the arduino.
- Connect the Echo pin to any digital pin on the arduino.
- Finally, connect the positive rail of the breadboard to 5V pin on the arduino and the negative rail of the breadboard to the Gnd pin on the arduino.

## 5. Software

To achieve the desired results, a programming language and interface was needed to devise a logic that'd make the Arduino board understand the requirements asked of it. Arduino comes with its own code editor, which accepts the C and C++ languages. Additionally, support for Java has also been provided by the use of modules that create a virtual runtime environment for the hardware to run. Also, the ultrasonic sensor needs to be programmed in order to function. Coding for that has also been done in the Arduino Web Editor, which employs its features of interrupts, pulses, time-outs, and signals. A question might arise that how to store database of availability of parking space? There is no need of storing it as it can be directly displayed on a temporary webpage. So for this we use HTML with little bit of CSS and then add the logic for displaying the parking sensor via PHP. The below picture shows how output will be displayed.



| Position | Availability |
|---|---|
| P 1 | ✗ |
| P 2 | ✗ |
| P 3 | ✗ |
| P 4 | ✓ |
| P 5 | ✗ |
| P 6 | ✓ |

**Fig-7:** How we get output of available parking spaces

## 6. Execution Plan

Step 1: Connections
Hardwiring the devices on Breadboard which includes the sensor and the Arduino Board.

Step 2: Software
Programming the sensor and the Arduino to function the way desired and then assigning values to the outlets of the Arduino and the sensor i.e. Vcc, ground, trigger, echo, etc.

Step 3: Data Connection
Pushing this data to the webpage using HTML as the language, CSS for the style, and PHP for the logic.

Step 4: Arduino connection
Connecting the arduino to the USB port of the PC, for data streaming.

Step 5: Final assembly of the design required for functioning.

Step 6: Test and execute.

## 7. CONCLUSIONS

Our project ensures to find free parking places for public. As soon as parking place is found to be empty it is detected using ultrasonic sensors which report it further. We achieved this by programming the sensors and Arduino. Pushing the data to webpage gives us tabular output which shows availability of parking places. The project aims at fast results so that anyone can easily find place for parking and save time in doing so. As Arduino is the latest technology, using it gives uniqueness to our project.

## REFERENCES

- The Arduino Platform, Coursera
- Arduino, Available at http://www.arduino.cc
- Yusuf Abdullahi Badamasi, The working principle of An Arduino, Electronics , Computer and Computation, 2014.