****

**PR201 Project Report**

**Mentor: Dr. VARUN BAJAJ**

**Harshit | Sahil | Nishchay | Shashi | Chandresh**

**RFID Based Vehicle Parking System**

**ABSTRACT**

Now a day’s many big institutes face the problem of parking. So to solve those various problems we have designed a system that would be very helpful. This is also seen that because of the vehicles of the outsiders the permanent employees and the people within the organization faces problems, so our parking system will be a boon for such organizations. This whole parking system is based on the RFID technology. At present there are many companies which provides the solutions for parking system of big organization and many big institutes but their cost is very high, but if we take this prototype on a practical level then as compared to the cost of the solutions provided by the big companies our costs will be very low.

**OBJECTIVE**

Developed in PR201:

A prototype of the parking system by using basic electronics components.

**Main Components used:**

* RFID
* Servo Motors
* Arduino

This will be a complete approach in overcoming most of the issues faced by the faculties and other employees of the institutes.

|  |  |  |
| --- | --- | --- |
| **Low Frequency (LF) RFID** | **High Frequency (HF) RFID** | **Ultrahigh Frequency (UHF) RFID** |
| Covers frequencies from 30 to 300 KHz | Ranges from 3 to 30 MHz | Ranges from 300 MHz to 3 GHz |
| Provides a short read range of 10cm | Read range is between 10cm and 1m | Read range of passive UHF system can be as long as 35m |
| Has slower read speed than the higher frequencies | Has moderate read speed | Has a very good read speed |
| Is least sensitive to radio wave interference | Experiences moderate sensitivity to radio wave interference | Is most sensitive to interference, but manufacturers have found ways to keep performance high even in difficult environments |

**Shortlisted RFID tags:**

* **OMNI ID DURA 1500 tag:** Omni-ID Dura 1500 is the durable and long range tag product offering long read ranges across all geographies. Designed with heavy industry in mind, the Omni-ID Dura 1500 features impact resistance and high temperature ratings, enabling it to be deployed in outdoor heavy industry environments anywhere in the world.
* **OMNI ID DURA 600 tag:** Omni-ID Dura 600 is a small form factor RFID tag, with impact resistance, and good on-metal performance. The combination of its size, flexible durable case design and foam adhesive makes it ideal for heavy industrial applications where curved or contoured assets are in use including pipes and valves
* **OMNI ID DURA 3000 tag:** Omni-ID Dura 3000 is a globally compliant, passive UHF tag with extreme read ranges of up to 35m, on, off or near metals and liquids. Omni-ID Dura 3000 features a high impact, waterproof, durable encasement. Omni-ID Dura 3000 is an excellent choice for tracking large assets in open storage environments, without the concern of a battery.

**WORKING METHODOLOGY**

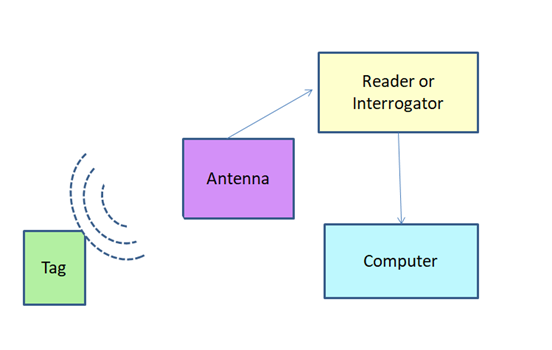
**1. Vehicle Identification through RFID**

The main part of the project depends on the identification of RFID tags incorporated on the parking slots. Radio Frequency Identification (RFID) is a generic term used to describe a system that transmits the identity of an object or person wirelessly using radio waves. An RFID reader of 13.5 MHz is used to read the UHF tags for the prototyping but only for a short range.

Radio-frequency identification (RFID) uses electromagnetic fields to automatically identify and track tags attached to objects.

**Structure of an RFID system**

The tags contain electronically stored information. Passive tags collect energy from a nearby RFID reader's interrogating radio waves.

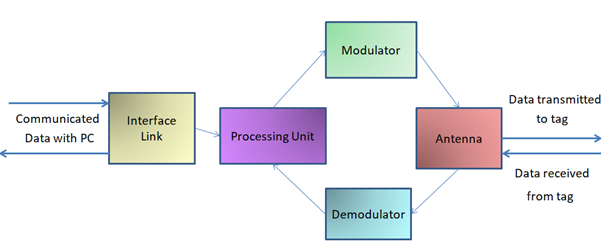


**Figure 3**

*Block diagram of RFID System*

**RFID tag**

The RFID component on the tags has two parts: a microchip that stores and processes information, and an antenna to receive and transmit a signal. The tag contains the specific serial number for one specific object.

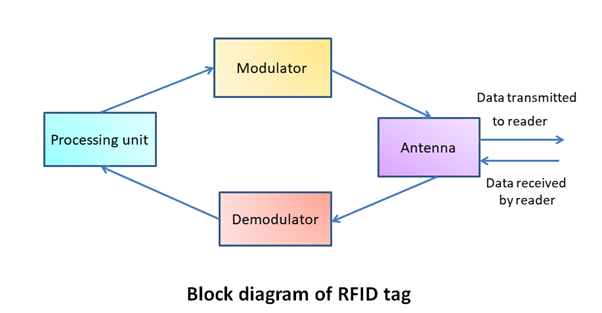


**Figure 4**

*Block diagram of an RFID Tag*

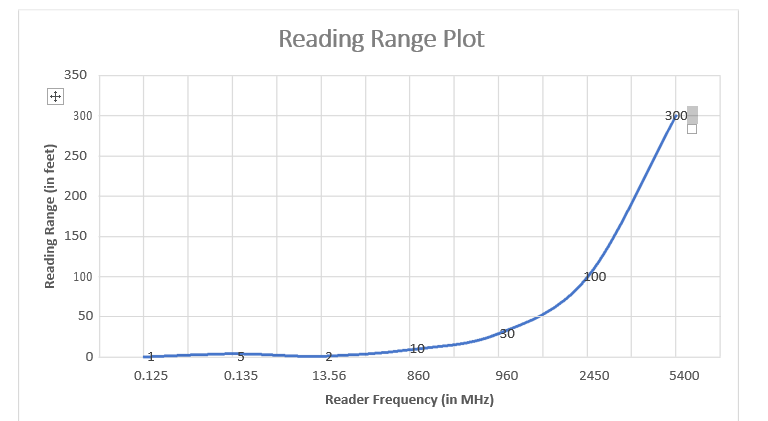
**RFID reader**

Depending on the frequency that is used and its performance, an RFID reader sends radio waves of between one centimeters and 30 meters or more. If a transponder enters this electromagnetic region, it detects the activating signal from the reader. The RFID reader decodes the data stored in the integrated circuit of the transponder (silicon chip), and communicates them, depending on the application, to a host system.



**RFID read range vs Frequency**

The read range of passive UHF systems can be as long as 12 m, and UHF RFID has a faster data transfer rate than LF or HF. UHF RFID is the most sensitive to interference, but many UHF product manufacturers have found ways of designing tags, antennas, and readers to keep performance high even in difficult environments. Passive UHF tags are easier and cheaper to manufacture than LF and HF tags.

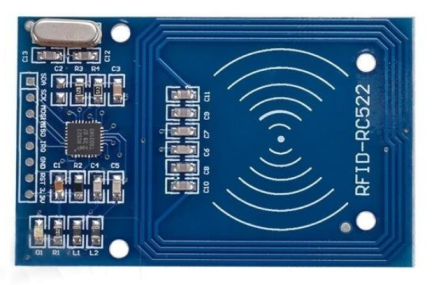


**Figure 6**

*Plot between the RFID reader frequency and its reading range*

**Modules for prototyping**

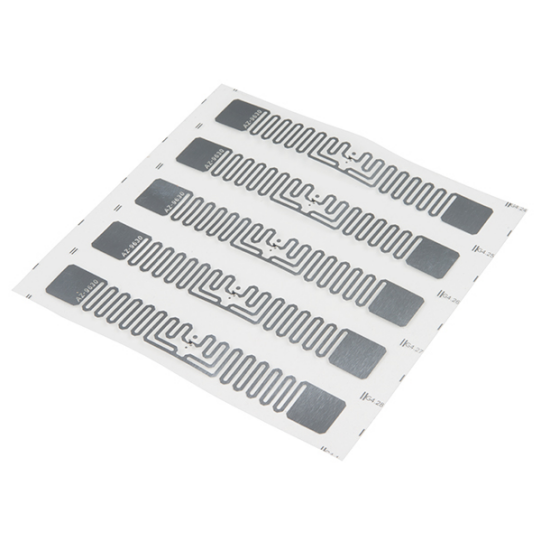
**1.RFID reader: MFRC522 RC522 RFID Reader Writer Mifare Sensor RF Module**

****

**Specifications:** Current :13-26mA / DC 3.3V Idle Current :10-13mA / DC 3.3V Sleep Current<80uA Peak Current<30mA Operating Frequency: 13.56MHz<30mA

MF RC522 is applied to 13.56MHz contactless communication highly integrated chip card reader. The module uses SPI to communicate with microcontrollers. The open-hardware community already has a lot of projects exploiting the RC522 – RFID Communication, using Arduino

1. **RFID tags: MFRC522 RC522 RFID Reader Writer Mifare Sensor RF Module**



* EPCglobal Gen2 and ISO/IEC 18000-6c
* 800 Bits of Memory
* 512 User Bits
* 64-Bit Unique TID (unalterable serial number)
* 32-Bit Access and 32-Bit Kill Passwords\* Width: 1cm
* Length: 7cm

These paper-thin EPCglobal Gen2 tags work with an RFID reader. The tags can read and be written to --- and also have a kill feature. Each tag comes with a TID (Truly Unique ID) that can't be changed, but there's plenty of memory for you to write and read from.

**3. Microcontroller: Arduino Mega 22**

****

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button.

* Microcontroller ATmega2560
* Operating Voltage 5V
* Input Voltage 7-12V
* Input Voltage (limit) 6-20V
* Digital I/O Pins 54 (of which 15 provide PWM output)
* Analog Input Pins 16
* DC Current per I/O Pin 20 mA
* DC Current for 3.3V Pin 50 mA
* Flash Memory 256 KB of which 8 KB used by bootloader
* SRAM 8 KB
* EEPROM 4 KB
* Clock Speed 16 MHz

1. **Servo Motor**

A servomotor is a [closed-loop](https://en.wikipedia.org/wiki/Closed-loop_controller" \o "Closed-loop controller) [servomechanism](https://en.wikipedia.org/wiki/Servomechanism" \o "Servomechanism) that uses position feedback to control its motion and final position. The input to its control is a signal (either analogue or digital) representing the position commanded for the output shaft.

The motor is paired with some type of [encoder](https://en.wikipedia.org/wiki/Encoder" \o "Encoder) to provide position and speed feedback. In the simplest case, only the position is measured. The measured position of the output is compared to the command position, the external input to the controller. If the output position differs from that required, an [error signal](https://en.wikipedia.org/wiki/Error_signal" \o "Error signal) is generated which then causes the motor to rotate in either direction, as needed to bring the output shaft to the appropriate position. As the positions approach, the error signal reduces to zero and the motor stops.

The very simplest servomotors use position-only sensing via a [potentiometer](https://en.wikipedia.org/wiki/Potentiometer" \o "Potentiometer) and [bang-bang control](https://en.wikipedia.org/wiki/Bang-bang_control" \o "Bang-bang control) of their motor; the motor always rotates at full speed (or is stopped). This type of servomotor is not widely used in industrial [motion control](https://en.wikipedia.org/wiki/Motion_control" \o "Motion control), but it forms the basis of the simple and cheap [servos](https://en.wikipedia.org/wiki/Servo_(radio_control)" \o "Servo (radio control)) used for [radio-controlled models](https://en.wikipedia.org/wiki/Radio-controlled_model" \o "Radio-controlled model).

More sophisticated servomotors use optical [rotary encoders](https://en.wikipedia.org/wiki/Rotary_encoder" \o "Rotary encoder) to measure the speed of the output shaft and a variable-speed drive to control the motor speed. Both of these enhancements, usually in combination with a [PID control](https://en.wikipedia.org/wiki/PID_controller" \o "PID controller) algorithm, allow the servomotor to be brought to its commanded position more quickly and more precisely, with less [overshooting](https://en.wikipedia.org/wiki/Overshoot_(signal)" \o "Overshoot (signal)).



**Working**

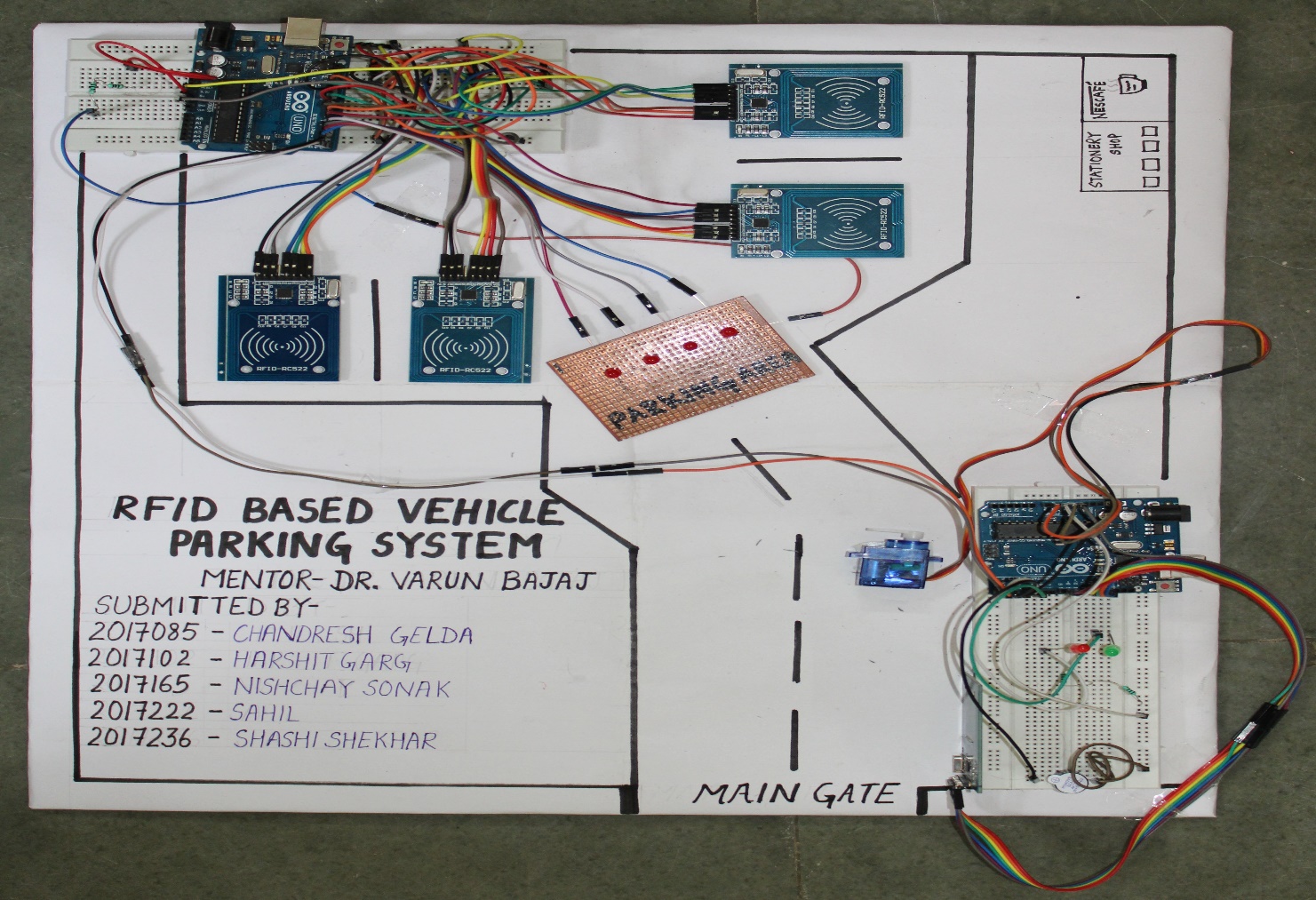
The RFID reader containing the antenna transmits the radio waves which is received by an electromagnetic loop in the tags. The passive tag (which we have used) uses the reader’s wave energy to relay its information back to the reader. The reader is interfaced with the microcontroller, i.e., Arduino (Atmega328p) via SPI communication with the Reader and thus receives the data stored in the tag. The stored data in the tags is the unique ID of every different tag. The unique ID defines the particular bus as stored in our database.

For this prototype three test cases are decided.

1. If car do not arrive in the parking zone then according to the code after the decided time the person will get the warning that he/she has to park the vehicle in the parking zone.
2. If the car arrived at the parking slot then it gets detected by the RFID reader and then we get the information about the spaces.
3. If the car again parks in the no parking zone and tries to get out before the warning time then he will get caught at the entrance gate with the help of RFID reader.

The most important part of this parking system is not only to generate penalty on the name of the persons who have not parked their vehicle in the parking slots but also to warn them that they have not parked their vehicle in the parking zone and to send them message that park the vehicle in the parking zone otherwise fine will be generated.

**Actual Image Of The Prototype**

****

1. **Further Improvement And Extension**

1. After the successful execution of the prototype, we have decided to take it to the institute level.

2. So for that the components list is given below and all the connections will be wireless as it would be difficult for us to do all the complex connections.

3. In the prototype, the system is not developed to send direct messages to the persons, so in the improvement we have decided to include this also in our model. So if the person coming to the institute will not park the vehicles in the parking zone then that person will get a warning message that park the car in the parking zone otherwise the fine will be generated.