Smart KEY

**Summery**

Smart key is Made to increase the safety for car key to Preventing sending signals by mistake or without the key owner's fingerprint



Objectives:

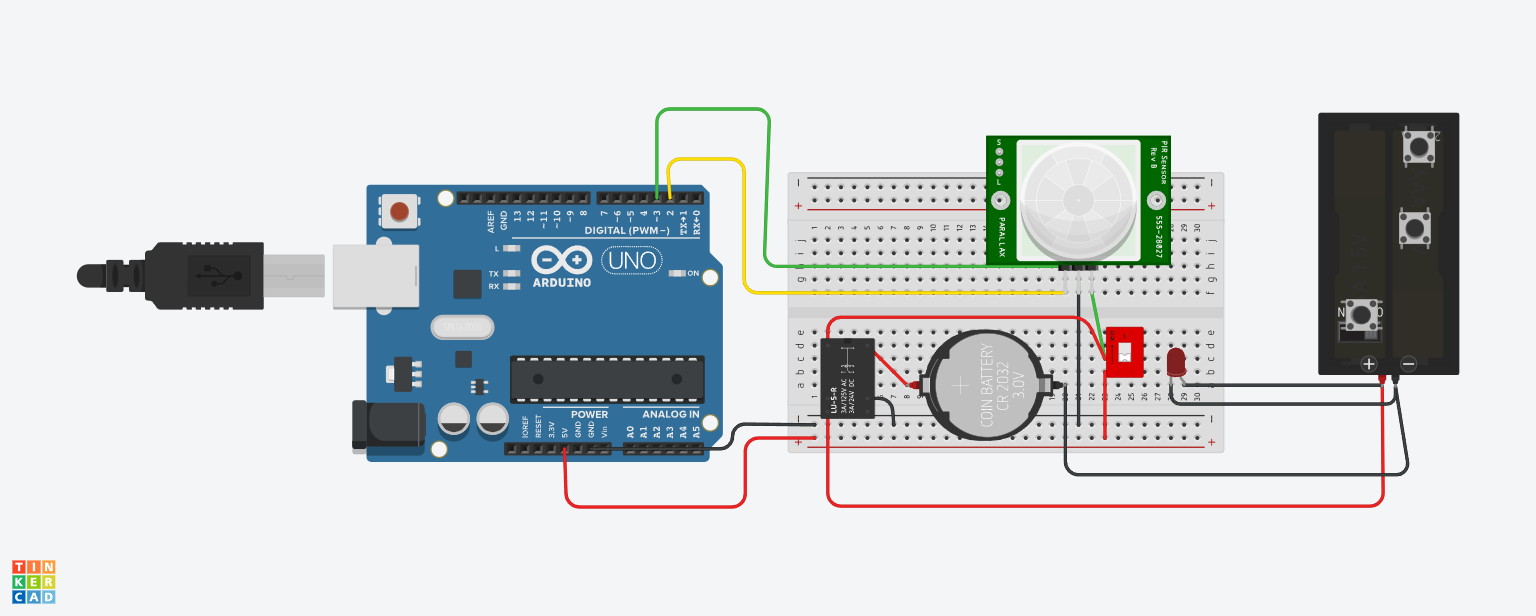
1. Increasing the safety of cars and the possibility of entry
2. Reduces theft and fraud

**Components:**

1. Main Switch
2. Controller board “ Arduino uno “
3. Fingerprint sensor
4. Relay model
5. Led to show case
6. Remote control

**How the circuit work:**

We have a Main Switch responsible about running the controller start to take action from fingerprint sensor until it matches with owner the controller send signal to relay to run the key for 3 minutes and after this the circuit will should down automatically

**Diagram:**

6- Remote control

3-Fingerprint sensor

2-Controller board “ Arduino uno “

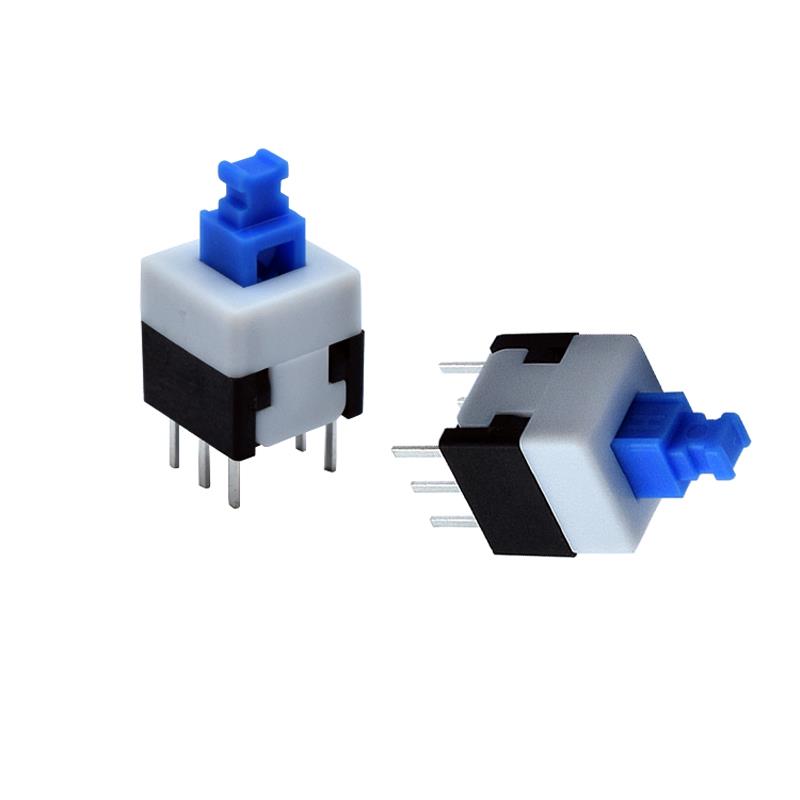
5-Signal led

4-Relay model

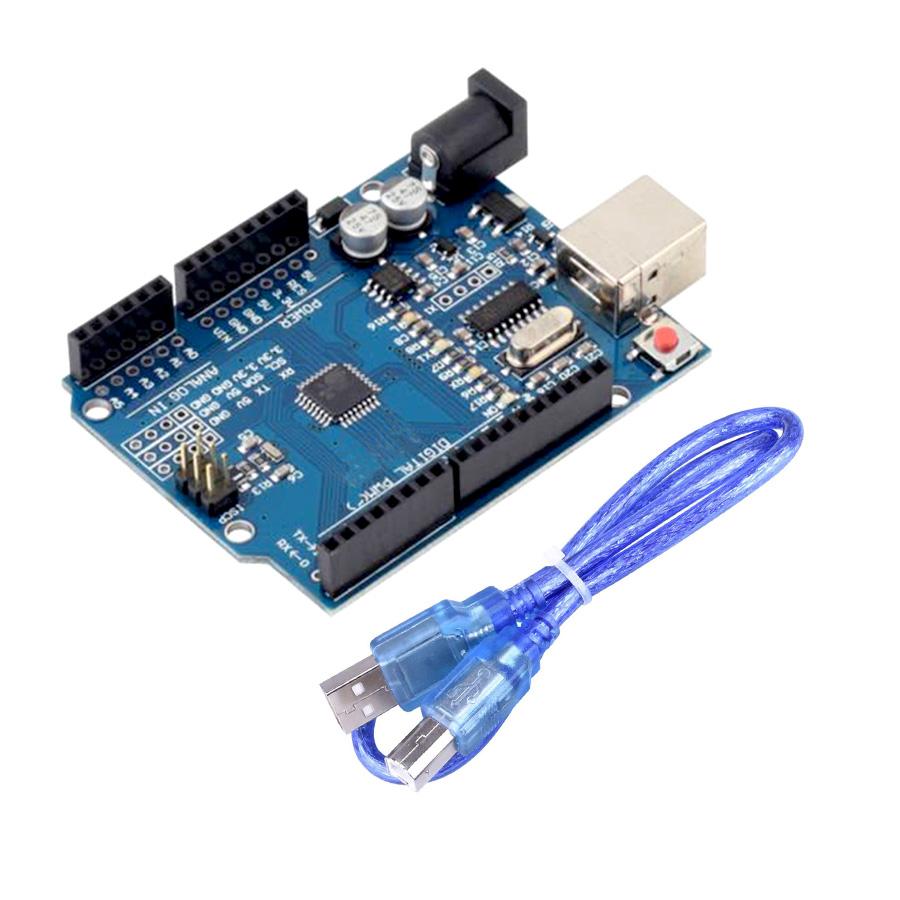
1-Main Switch

**Usage**

1. Main Switch  
   responsible about running the cruciate and connect the power from main source to controller and relay



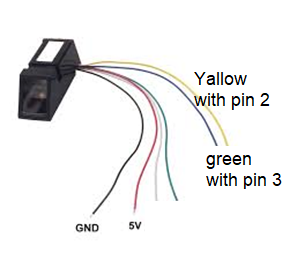
1. Controller board “ Arduino uno “   
   responsible about controlling for circuit it’s work with 5 v Supplying from USB and can programming with Arduino Ide using c++ language, Arduino start to check there is signal from fingerprint sensor then start to match the id of user with recorded owner the controller if it’s correct start to send signal for relay to close circuit and turn on the key



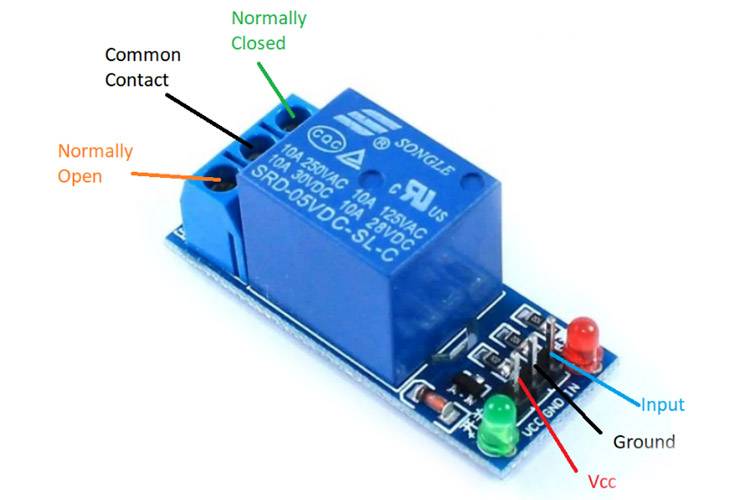
1. Fingerprint sensors and connections

he fingerprint sensor we are using is an Optical Type, there exists two more types of sensor like capacitive which can be found in smart phones and ultrasonic ones, which are yet in testing phase, and both these options are expensive, so we will focus on this optical type for this hobby electronics and similar projects.

The way this optical fingerprint sensor works is that it captures a photo of our finger ridges, and then it uses certain algorithm to match it with stored data and displays result of the same.



1. Relay Modul   
   is an electrical switch that is operated by an electromagnet. The electromagnet is activated by a separate low-power signal from a micro controller. When activated, the electromagnet pulls to either open or close for the key



1. Led to show case

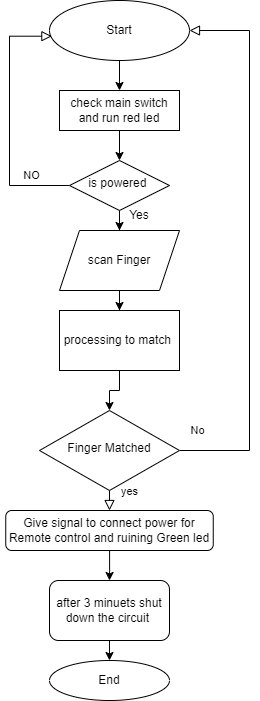
Red led : to show the power is connecting for all circuit

blue led: to show the signal for the key

1. Key “remote control “ for car

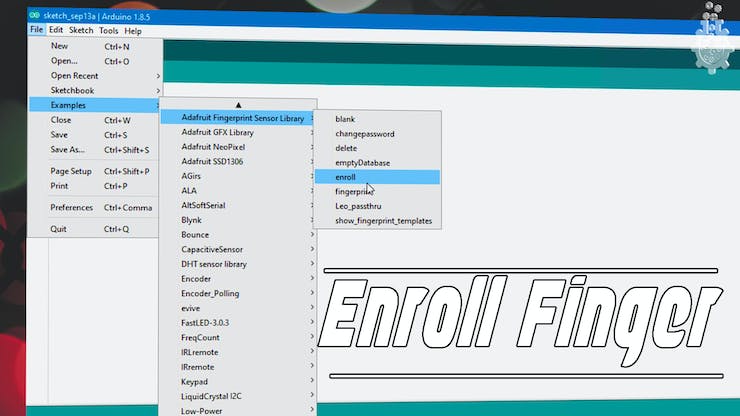
****

**Flowchart :**



Code Explanation:

1. First we need to upload Enroll code for the Finger Prints



first we need to enroll the fingers into the EPROM of our Sensor module, so we will follow these steps to enroll our fingerprints.

**select**the ***enroll***example.

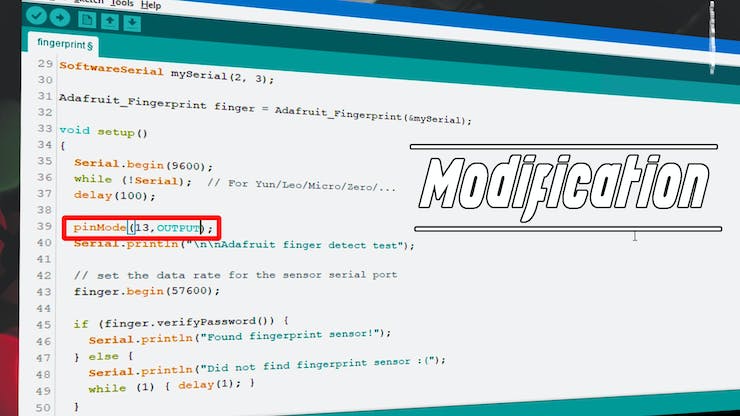
* **Open A**rduino **IDE**
* Under the **FILE**menu, and **Examples**, find the **Adafruit's fingerprint** library.
* **Upload**the enroll example.
* **Open** the **Serial Monitor**.
* **Select**the **baud rate** to be **9600**.
* **Enter**the **Finger Print ID number** when the serial monitor prompts to enter the fingerprint id number.
* **Place**the **finger**you need to enroll **on**the **sensor**.
* Place the finger again on the sensor once prompted by serial monitor.

1. Upload the Main Code

Since we enrolled the fingerprints in the last step, now we can test if this sensor works or may be check for match using this following code and steps.

* **Open**arduino **IDE**
* Under the **FILE**menu, and **Examples**, find the **adafruits finger print** library.
* **Select**the **fingerprint**example.
* **Upload** the **fingerprint**example.
* **Open**the **Serial Monitor.**
* **Select**the **baud rate** to be **9600**.
* **Place**the **finger**you need to test **on** the **sensor**.
* **Check** for fingerprint id number for **OUTPUT**.

1. Modify the Main Code to Interface Relay and blue led to can give signal for remote



Now let’s add a led to indicate our Fingerprint match, to do so follow these steps:

**use**"*digitalWrite(12, HIGH); delay(30000); digitalWrite(12, LOW);* "on line 144 to turn on and off the LED

* **Define**the **LED**on pin 12 **as output**in void setup part
* **Find**then on **line 135**we see a **fingerprint id match loop**, where in we use if statements along with library definitions to find our fingerprint id.
* We will **give**a **high pulse** for **1 second** and **turn** the **LED off**.
* After this line and our code will turn on the LED every time sensor finds a match.

**Prototype**

