

Vehicle Anti-theft system by implementing two module method

Ritesh kumar

Electronics and communication Engineer, India.

Corresponding Author: aeriteshkumar061997@gmail.com

Abstract: - This Vehicle Anti-theft system will stop a vehicle from getting theft and call immediately the owner of the vehicle if anyone else is try to operate the vehicle without the owner permission.

Key Words:— *Raspberry pi , Arduino , GSM Module , Python programming ,C programming, SMTP.*

I. INTRODUCTION

With the increase theft across all over the nation, it is very important to have a security for the individual vehicle, that's how we could able to safe our vehicle if we install any new type of intelligence device into our vehicle or car, this system deals with the same concept. This system is divided into two different parts; in one part it is manually password locked system and other part is totally intelligence system.

II. COMPONENTS INVOLVED

A. Raspberry Pi 3 Model B+

The Raspberry Pi 3 Model B+ Is The Final Revision in The Raspberry Pi 3 Range.

1. Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz.
2. 1GB LPDDR2 SDRAM.
3. 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE.
4. Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps).
5. Extended 40-pin GPIO header.
6. Full-size HDMI.
7. 4 USB 2.0 ports.
8. CSI camera port for connecting a Raspberry Pi camera.
9. DSI display port for connecting a Raspberry Pi touchscreen display.
10. 4-pole stereo output and composite video port
11. Micro SD port for loading your operating system and storing data.
12. 5V/2.5A DC power input.

13. Power-over-Ethernet (PoE) support (requires separate PoE HAT).

B. Raspberry Pi Camera

The Raspberry Pi Camera Module v2 replaced the original Camera Module in April 2016. The v2 Camera Module has a Sony IMX219 8-megapixel sensor (compared to the 5-megapixel OmniVision OV5647 sensor of the original camera). The Camera Module can be used to take high-definition video, as well as stills photographs. It's easy to use for beginners, but has plenty to offer advanced users if you're looking to expand your knowledge. There are lots of examples online of people using it for time-lapse, slow-motion, and other video cleverness. You can also use the libraries we bundle with the camera to create effects. You can read all the gory details about IMX219 and the Exmor R back-illuminated sensor architecture on Sony's website, but suffice to say this is more than just a resolution upgrade: it's a leap forward in image quality, color fidelity, and low-light performance. It supports 1080p30, 720p60 and VGA90 video modes, as well as still capture.

It attaches via a 15cm ribbon cable to the CSI systems such as Microsoft Exchange and IBM Notes and webmail systems such as Outlook.com, Gmail and Yahoo! Mail may use non-standard protocols internally, but all use SMTP when sending to or receiving email from outside their own systems. SMTP servers commonly use the Transmission Control Protocol on port number 25.

User-level email clients typically use SMTP only for sending messages to a mail server for relaying, and typically submit outgoing email to the mail server on port 587 or 465 as per RFC 8314. For retrieving messages, IMAP and POP3 are standard, but proprietary servers also often implement proprietary protocols. port on the Raspberry Pi. The camera works with all models of Raspberry Pi 1, 2, 3 and 4. It can be accessed through the MMAL and V4L APIs, and there are numerous third-party libraries built for it, including the Picamera Python library. See the Getting Started with

Picamera resource to learn how to use it. The camera module is very popular in home security applications, and in wildlife camera traps.

C. Servomotors

A *servomotor* is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration.[1] It consists of a suitable motor coupled to a sensor for position feedback. It also requires a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors

D. Arduino Uno

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc.[2][3] The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits.[1] The board has 14 digital I/O pins (six capable of PWM output), 6 analog I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable.[4] It can be powered by the USB cable or by an external 9-volt battery, though it accepts voltages between 7 and 20 volts. It is similar to the Arduino Nano and Leonardo.[5][6] The hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available.

E. GSM module Sim 900A

The SIM900A is a readily available GSM/GPRS module, used in many mobile phones and PDA. The module can also be used for developing IOT (Internet of Things) and Embedded Applications. SIM900A is a dual-band GSM/GPRS engine that works on frequencies EGSM 900MHz and DCS 1800MHz. SIM900A features GPRS multi-slot class 10/ class 8 (optional) and supports the GPRS coding schemes CS-1, CS-2, CS-3 and CS-4.

III. SOFTWARE INVOLVED

The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in functions from C and C++ 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. 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 toolchain, 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. By default, avrdude is used as the uploading tool to flash the user code onto official Arduino boards.

A. C programming

C (/si:/, as in the letter c) is a general-purpose, procedural computer programming language supporting structured programming, lexical variable scope, and recursion, with a static type system. By design, C provides constructs that map efficiently to typical machine instructions. It has found lasting use in applications previously coded in assembly language. Such applications include operating systems and various application software for computers architectures that range from supercomputers to PLCs and embedded systems.

B. Embedded c

Embedded C is a set of language extensions for the C programming language by the C Standards Committee to address commonality issues that exist between C extensions for different embedded systems.

Embedded C programming typically requires nonstandard extensions to the C language in order to support enhanced microprocessor features such as fixed-point arithmetic, multiple distinct memory banks, and basic I/O operations. In 2008, the C Standards Committee extended the C language to address such capabilities by providing a common standard for all implementations to adhere to. It includes a number of features not available in normal C, such as fixed-point arithmetic, named address spaces and basic I/O hardware addressing. Embedded C uses most of the syntax and semantics of standard C, e.g., main () function, variable definition, datatype declaration, conditional statements (if, switch case), loops (while, for), functions, arrays and strings, structures and union, bit operations, macros, etc.

C. Noobs operating system

NOOBS is an easy operating system installer which contains Raspberry Pi OS and LibreELEC. It also provides a selection of alternative operating systems which are then downloaded from the internet and installed.

NOOBS Lite contains the same operating system installer without Raspberry Pi OS pre-loaded. It provides the same operating system selection menu allowing Raspberry Pi OS

and other images to be downloaded and installed.

The Simple Mail Transfer Protocol (SMTP) is a communication protocol for electronic mail transmission. As an Internet standard, SMTP was first defined in 1982 by RFC 821, and updated in 2008 by RFC 5321 to Extended SMTP additions, which is the protocol variety in widespread use today. Mail servers and other message transfer agents use SMTP to send and receive mail messages.

D. Python

Python is dynamically typed and garbage-collected.

It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.[29]

Python was conceived in the late 1980s as a successor to the ABC language. Python 2.0, released in 2000, introduced features like list comprehensions and a garbage collection system with reference counting.

Python 3.0, released in 2008, was a major revision of the language that is not completely backward-compatible, and much Python 2 code does not run unmodified on Python 3.

The Python 2 language was officially discontinued in 2020 (first planned for 2015), and "Python 2.7.18 is the last Python 2.7 release and therefore the last Python 2 release." [30] No more security patches or other improvements will be released for it.[31][32] With Python 2's end-of-life, only Python 3.5.x[33] and later are supported.

Python interpreters are available for many operating systems. A global community of programmers develops and maintains CPython, an open source[34] reference implementation. A non-profit organization, the Python Software Foundation, manages and directs resources for Python and CPython development.

IV. WORKING

Working of this system is based upon two modules which is operated by switching digital or analog can be designed by the designer, now one phase of the system is based upon the use of Arduino, Arduino keypad, servo motor and GSM module. Firstly, the owner put a lock according to his own choice and the when the owner put the actual lock into keyboard the servo starts rotating and this indicate that system is working and in terms of car or vehicle implementing the spark wire can be connected through a relay and we can replace it from the servo, and if the person do not detect the correct password it will directly call the user by GSM module and all this coding is done in embedded C in Arduino ide and flash into Arduino

board.

Second phase of this system is owner can switch it into automatically tracking of the car / vehicle , it is based upon the face detection method using python programming , owner face is already saved into Raspberry pi memory and when anyone try to start a car , camera will detect it's photograph if the code detect that it is the same person it will allow the memory to start the car and give power to one respected pin or raspberry Pi and if it did not detect the face of owner it will stop the car and click photo and send it to the email address of owner by SMTP protocol using Python programming .

V. CONCLUSION

System works fine and exactly same as working. The servo works fine and email comes with the photograph of unknown person. If anyone enter wrong password it gives a call to owner. Implementation in model scale is achieved.

REFERENCES

- [1]. <https://www.raspberrypi.org>.
- [2]. <https://www.raspberrypi.org/education>.
- [3]. <https://create.arduino.cc/projecthub>.
- [4]. <https://www.raspberrypi.org/products/raspberrypi-3-model-b-plus>.
- [5]. <https://www.raspberrypi.org/help/noobs-setup/2>.