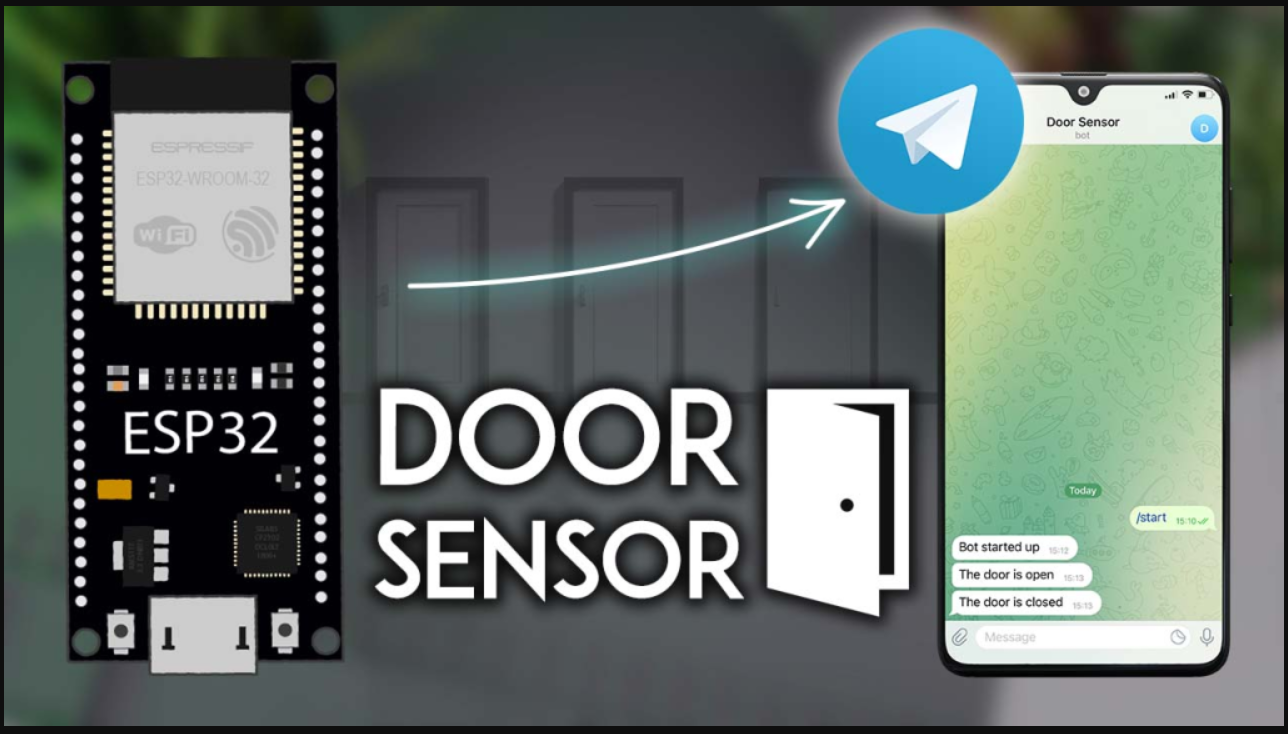
CPE316 – Embedded Systems Final Project Report

# **ESP32 Door Status Monitor with Telegram Notifications**

Semester II (2021-2022)



Contents

[1. Introduction 3](https://docs.google.com/document/d/1p5vC29DMGmUnSH58RNByaq4vzThP6eCZy8xCxyE4TbE/edit#heading=h.gjdgxs)

[2. Related Works 3](https://docs.google.com/document/d/1p5vC29DMGmUnSH58RNByaq4vzThP6eCZy8xCxyE4TbE/edit#heading=h.30j0zll)

[3. Project Design 3](https://docs.google.com/document/d/1p5vC29DMGmUnSH58RNByaq4vzThP6eCZy8xCxyE4TbE/edit#heading=h.1fob9te)

[3.1. Project Layout 3](https://docs.google.com/document/d/1p5vC29DMGmUnSH58RNByaq4vzThP6eCZy8xCxyE4TbE/edit#heading=h.3znysh7)

[3.2. Project Mechanism 4](https://docs.google.com/document/d/1p5vC29DMGmUnSH58RNByaq4vzThP6eCZy8xCxyE4TbE/edit#heading=h.2et92p0)

[4. Conclusion and Results 5](https://docs.google.com/document/d/1p5vC29DMGmUnSH58RNByaq4vzThP6eCZy8xCxyE4TbE/edit#heading=h.tyjcwt)

[5. Lesson Learnt 5](https://docs.google.com/document/d/1p5vC29DMGmUnSH58RNByaq4vzThP6eCZy8xCxyE4TbE/edit#heading=h.3dy6vkm)

[6. References 5](https://docs.google.com/document/d/1p5vC29DMGmUnSH58RNByaq4vzThP6eCZy8xCxyE4TbE/edit#heading=h.1t3h5sf)

[7. Attachments 6](https://docs.google.com/document/d/1p5vC29DMGmUnSH58RNByaq4vzThP6eCZy8xCxyE4TbE/edit#heading=h.4d34og8)

[7.1. Project Images 6](https://docs.google.com/document/d/1p5vC29DMGmUnSH58RNByaq4vzThP6eCZy8xCxyE4TbE/edit#heading=h.2s8eyo1)

[7.2. Code Modules 7](https://docs.google.com/document/d/1p5vC29DMGmUnSH58RNByaq4vzThP6eCZy8xCxyE4TbE/edit#heading=h.17dp8vu)

**INTRODUCTİON**

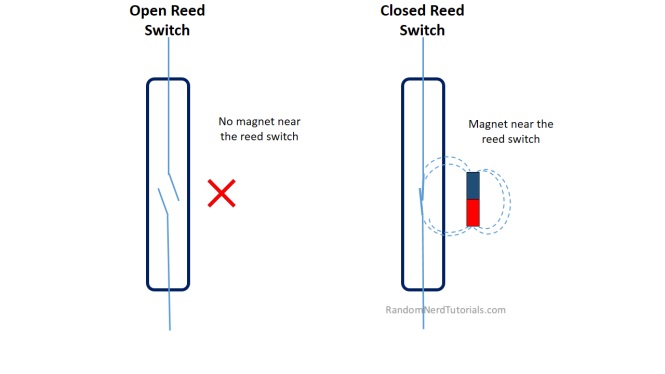
In this project, we’re going to monitor the status of a door using an ESP32 board and a magnetic reed switch. We’ll receive a message in our Telegram account whenever the door changes state: opened or closed. As long as we have access to the internet on our smartphone, we’ll be notified no matter where we are. The ESP32 board will be programmed using Arduino IDE.

In this project, we’ll create a Telegram Bot that will send messages to your Telegram account whenever a door changes state. To detect the change, we’ll use a magnetic contact switch.

A magnetic contact switch is basically a reed switch encased in a plastic shell so that you can easily apply it to a door, a window, or a drawer to detect if it is open or closed.



The electrical circuit is closed when a magnet is near the switch—door closed. When the magnet is far away from the switch—door open—the circuit is open.



We can connect the reed switch to an ESP32 GPIO to detect changes in its state.

**RELATED WORKS**

Because the technology involved in an entry sensor is so elegant, it can be used in a lot of creative ways—especially if you can program different responses for alarm events. Some security companies allow customers to decide whether a triggered sensor means a full-on alarm response, an alarm response with a detailed event report sent to your email or phone (a Smart Alert), or a private text or email, without a public alarm event (a Secret Alert).

Here are some ways to take advantage of those features and get the most out of your entry sensors.

* ***Doorbell:***Your SimpliSafe entry sensor doubles as a doorbell: If your system is off, it will still sound a chime if someone triggers a sensor. Great way to know when your husband has finally brought home dinner—or if your toddler has managed to open a window in his room.
* ***"Virtual Bouncer":***Entry sensors don't just keep out the outside world — they can warn you about breaches within your home, too. Stick one on the entrance to the game room or the door to the liquor cabinet. Rig it up with a Secret Alert so that it only tells you if the seal is broken — so you can crack down on those rule-breakers.
* ***Spy:***Savvy SimpliSafe employee Ryan K. positioned his TV right where he likes it, stuck on an entry sensor, and then set up a Secret Alert. "Now I know if anyone moves it," he says. I wish I'd done something similar with my mini-fridge in college.
* ***Party Fails:***   
  Want guaranteed entertainment at your next house party? Put an entry sensor on a medicine cabinet in your bathroom and set up a Secret Alert. You'll get a text message every time someone peeks, and you'll finally know which of your friends can't control their curiosity.

**PROJECT DESİGN**

[***3.1. Project Layout***](https://docs.google.com/document/d/1p5vC29DMGmUnSH58RNByaq4vzThP6eCZy8xCxyE4TbE/edit#heading=h.3znysh7)



***3.2 Project design***

Here’s the hardware that you need to complete this project:

* [ESP32](https://makeradvisor.com/tools/esp32-dev-board-wi-fi-bluetooth/) – read [Best ESP32 Development Boards](https://makeradvisor.com/esp32-development-boards-review-comparison/)
* [1× Magnetic Reed Switch](https://makeradvisor.com/tools/magnetic-reed-switch/)
* 1× 10kΩ resistor
* 1× breadboard
* Jumper wires

We attach the sensor and 1.terminal with gpio. Then attach that terminal to the magnet with cable. Also we attach the sensor directly to the magnet with cable. The sensor was attached to the resistance, the resistance was attached to the 2.terminal which is attached to the 1.terminal.

**4.CONCLUSİON AND RESULTS**

At the end of the project, We decided that the usage area is wide and the working principle of this project is useful and can be developed further.

**5.Lesson Learnt**

We learnt about some mechanical information and got different ideas about the software.

**6.References**

https://randomnerdtutorials.com/esp32-door-status-telegram/#more-106942

https://simplisafe.com/blog/door-sensor-secrets

## 

## 

## 

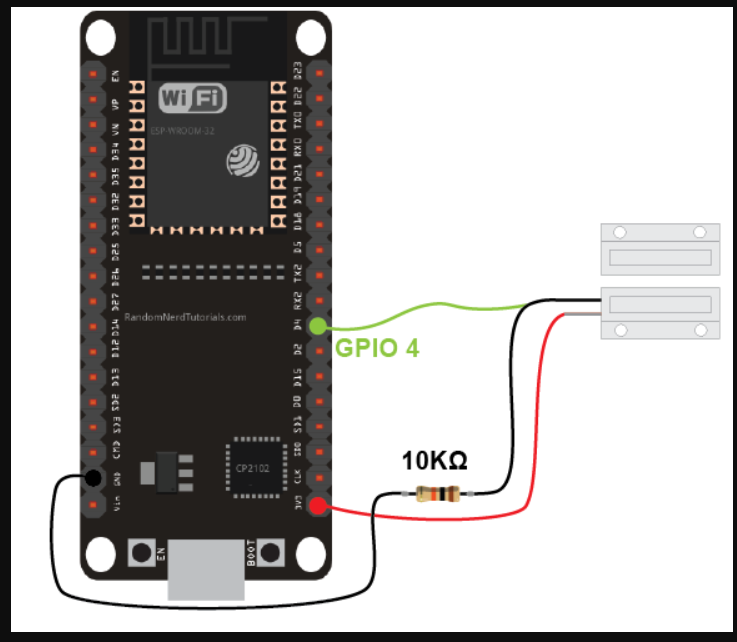
## 

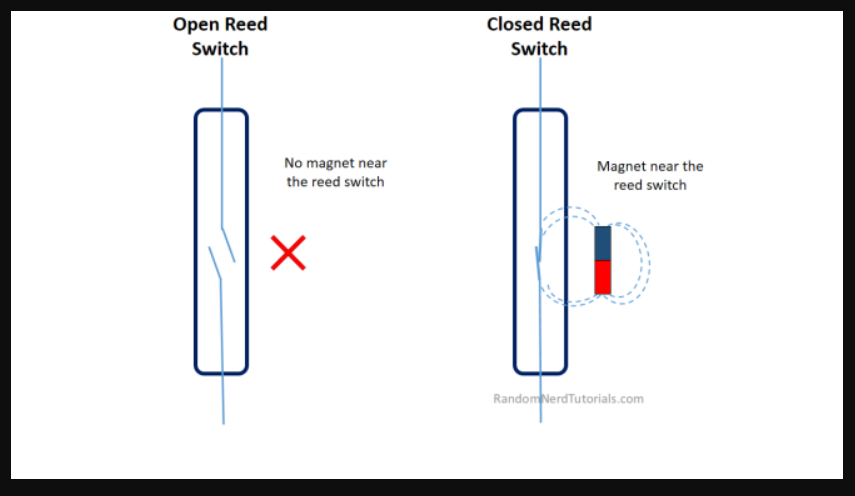
## 

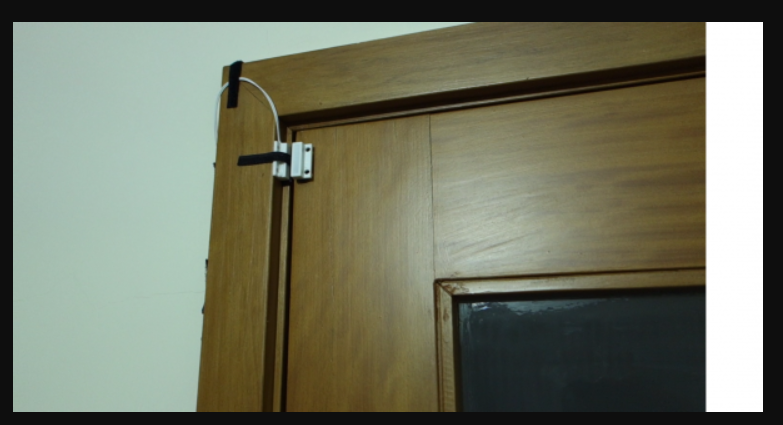
## 

## **7.** **Attachments**

7.1 Project Images









**7.2 Code Modules**

**Codes**

**/\***

**Rui Santos**

**Complete project details at https://RandomNerdTutorials.com/esp32-door-status-telegram/**

**Permission is hereby granted, free of charge, to any person obtaining a copy**

**of this software and associated documentation files.**

**The above copyright notice and this permission notice shall be included in all**

**copies or substantial portions of the Software.**

**\*/**

**#include <WiFi.h>**

**#include <WiFiClientSecure.h>**

**#include <UniversalTelegramBot.h>**

**#include <ArduinoJson.h>**

**// Set GPIOs for LED and reedswitch**

**const int reedSwitch = 4;**

**const int led = 2; //optional**

**// Detects whenever the door changed state**

**bool changeState = false;**

**// Holds reedswitch state (1=opened, 0=close)**

**bool state;**

**String doorState;**

**// Auxiliary variables (it will only detect changes that are 1500 milliseconds apart)**

**unsigned long previousMillis = 0;**

**const long interval = 1500;**

**const char\* ssid = "REPLACE\_WITH\_YOUR\_SSID";**

**const char\* password = "REPLACE\_WITH\_YOUR\_PASSWORD";**

**// Initialize Telegram BOT**

**#define BOTtoken "XXXXXXXXXX:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX" // your Bot Token (Get from Botfather)**

**// Use @myidbot to find out the chat ID of an individual or a group**

**// Also note that you need to click "start" on a bot before it can**

**// message you**

**#define CHAT\_ID "XXXXXXXXXX"**

**WiFiClientSecure client;**

**UniversalTelegramBot bot(BOTtoken, client);**

**// Runs whenever the reedswitch changes state**

**ICACHE\_RAM\_ATTR void changeDoorStatus() {**

**Serial.println("State changed");**

**changeState = true;**

**}**

**void setup() {**

**// Serial port for debugging purposes**

**Serial.begin(115200);**

**// Read the current door state**

**pinMode(reedSwitch, INPUT\_PULLUP);**

**state = digitalRead(reedSwitch);**

**// Set LED state to match door state**

**pinMode(led, OUTPUT);**

**digitalWrite(led, !state);**

**// Set the reedswitch pin as interrupt, assign interrupt function and set CHANGE mode**

**attachInterrupt(digitalPinToInterrupt(reedSwitch), changeDoorStatus, CHANGE);**

**// Connect to Wi-Fi**

**WiFi.mode(WIFI\_STA);**

**WiFi.begin(ssid, password);**

**client.setCACert(TELEGRAM\_CERTIFICATE\_ROOT); // Add root certificate for api.telegram.org**

**while (WiFi.status() != WL\_CONNECTED) {**

**delay(500);**

**Serial.print(".");**

**}**

**Serial.println("");**

**Serial.println("WiFi connected");**

**bot.sendMessage(CHAT\_ID, "Bot started up", "");**

**}**

**void loop() {**

**if (changeState){**

**unsigned long currentMillis = millis();**

**if(currentMillis - previousMillis >= interval) {**

**previousMillis = currentMillis;**

**// If a state has occured, invert the current door state**

**state = !state;**

**if(state) {**

**doorState = "closed";**

**}**

**else{**

**doorState = "open";**

**}**

**digitalWrite(led, !state);**

**changeState = false;**

**Serial.println(state);**

**Serial.println(doorState);**

**//Send notification**

**bot.sendMessage(CHAT\_ID, "The door is " + doorState, "");**

**}**

**}**

**}**