

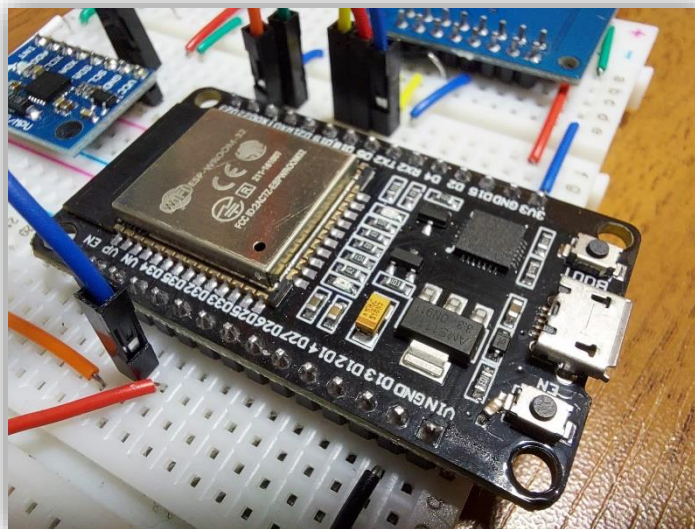
SERWER FTP Z ZABEZPIECZENIEM „TAMPER”

DOKUMENTACJA TECHNICZNA

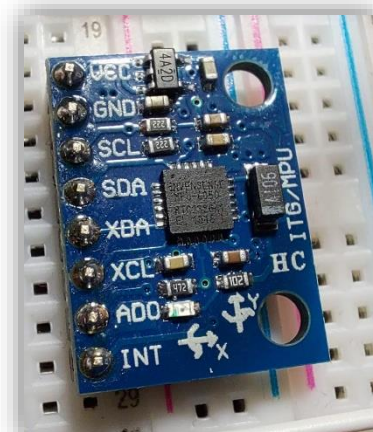
BUDOWA UKŁADU

UŻYTE KOMPONENTY

- Moduł rozwojowy ESP-32 (tu: ESP-WROOM-32 – wersja 20-pin) – [link](#)



- Czytnik kart SD z interfejsem SPI – dostępny np. [link](#)
- Układ MPU6050 - żyroskop i akcelerometr – [link](#)
- Układ RC522 – czytnik kart RFID 13.56MHz - [link](#)

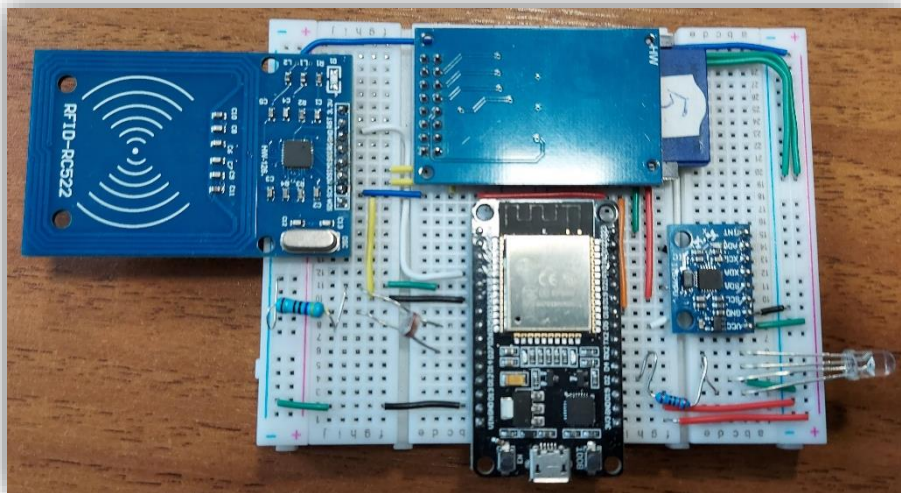


Dodatkowo do działania płytki użyte zostały następujące elementy:

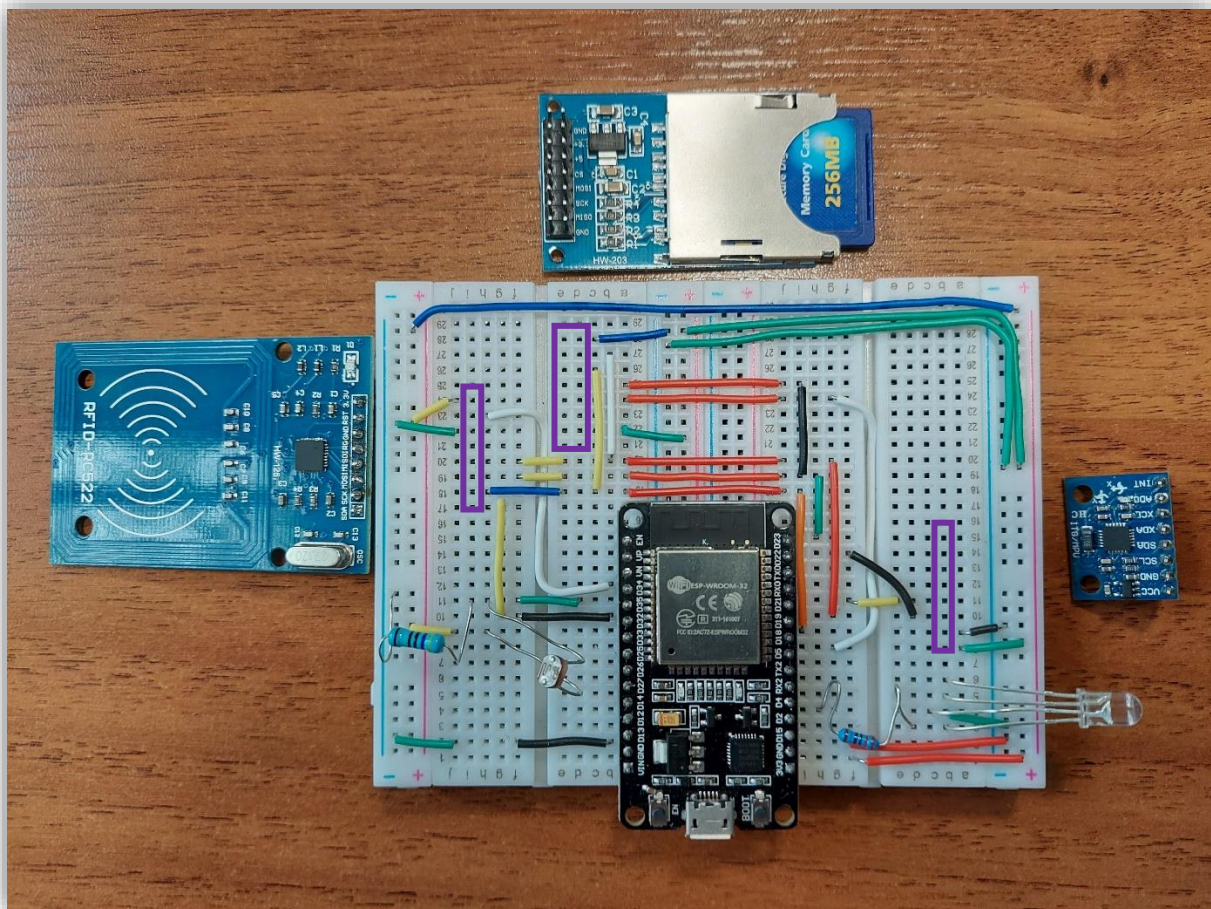
- 2 szt. płytek stykowych – 400 otworów
- Dioda RGB (wspólna katoda)
- Fotorezystor
- Rezystor 220 Ω (do diody)
- Rezystor 4.7 k Ω (pull-down do fotorezystora)
- Zestaw zworek (przewodów) do płytek stykowych
- Karta SD (z systemem plików FAT)

PODŁĄCZENIE KOMPONENTÓW

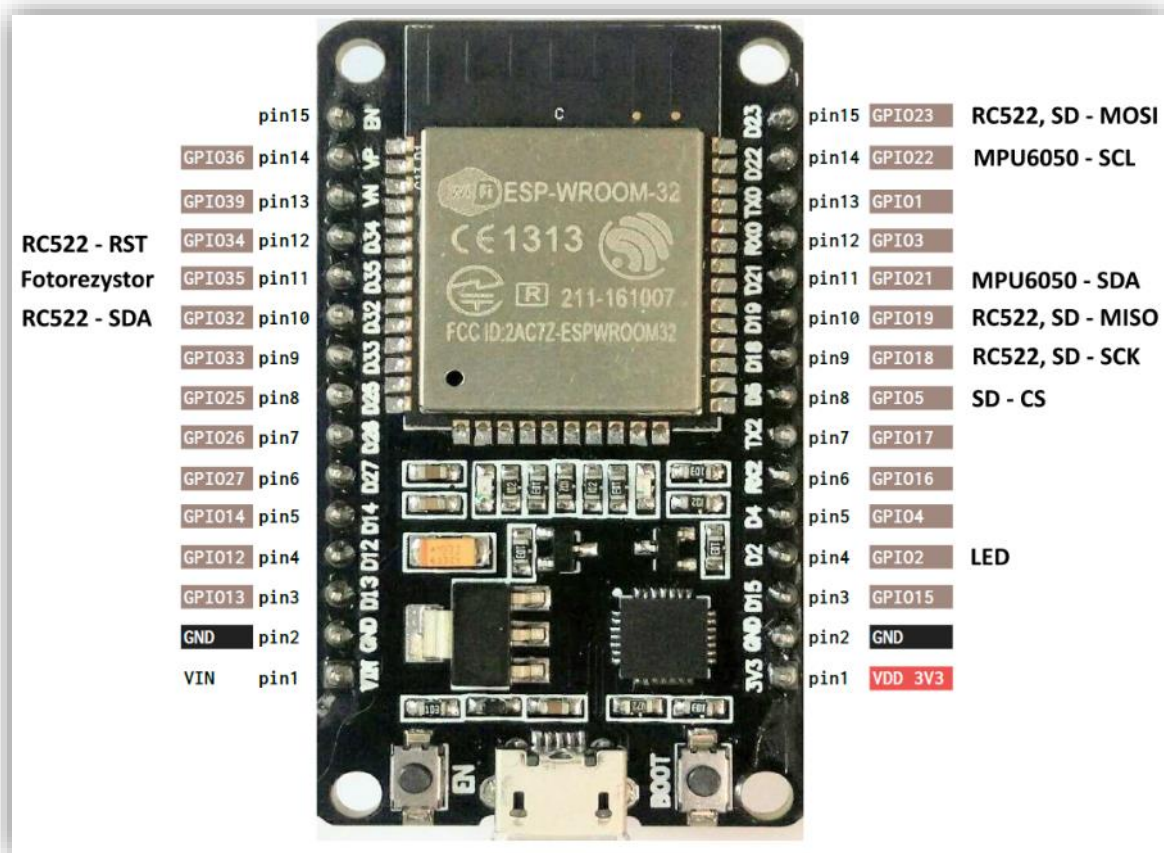
Gotowy układ po podłączeniu elementów prezentuje się następująco:



Na pełnym schemacie połączeń, przedstawionym poniżej, fioletowymi obszarami zaznaczono miejsce podłączenia sensorów:

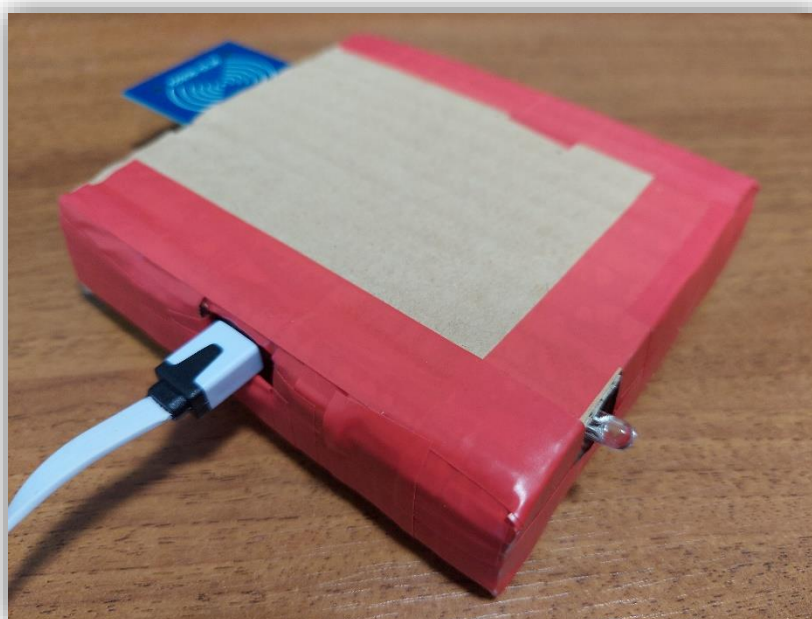


W Module ESP zostały wykorzystane następujące piny GPIO:



GOTOWY MODUŁ

Finalnie układ po dodaniu kartonowej obudowy może prezentować się następująco:



OPROGRAMOWANIE

Kod źródłowy dostępny jest do pobrania z serwisu GitHub:

<https://github.com/def-a1t/esp32-ftp>

KONFIGURACJA

Przed wgraniem programu na esp32 wymagana jest wcześniejsza konfiguracja.

Należy skopiować plik `src/credentials.h.sample` i nazwać go `src/credentials.h`. Należy tam umieścić zgodnie z przykładem następujące parametry:

- Nazwa i hasło sieci WiFi do której ma zostać podłączone urządzenie
- Nazwa użytkownika i hasło dostępowe do serwera FTP
- Identyfikatory zaufanych kart RFID

KOMPILACJA

Do wgrania programu polecamy wykorzystać program Visual Studio Code z rozszerzeniem *PlatformIO*.

Po złożeniu układu według instrukcji użytkownika, i podłączeniu esp32 do komputera za pomocą kabla USB, należy przejść do zakładki *PlatformIO*, a następnie wybrać opcję *Upload and Monitor*. Program może zapytać o adres urządzenia - wybieramy odpowiedni port szeregowy.

Po wgraniu programu na ekranie wyświetli się adres IP serwera FTP - możemy nawiązać z nim połączenie za pomocą klienta FTP.

POŁĄCZENIE

Zalecane jest użycie oprogramowania Filezilla do testowania połączenia z serwerem FTP. Przed połączeniem konieczna jest zmiana liczby jednoczesnych połączeń opisana w instrukcji użytkownika.

STRUKTURA KODU ŹRÓDŁOWEGO

SRC/MAIN.CPP

W tym pliku inicjalizowane są wszystkie funkcje programu - połączenie z wifi, odczyty z sensorów, obsługa karty RFID oraz serwer FTP.

Program operuje na zadaniach uruchamianych poprzez funkcje systemu **FreeRTOS**:

- xTaskCreatePinnedToCore
- vTaskSuspend
- vTaskResume
- vTaskDelay
- vTaskDelayUntil

Wykorzystywana jest wielowątkowość – w użyciu są oba rdzenie modułu ESP-32:

- Rdzeń 1: Obsługa połączeń i serwera FTP
- Rdzeń 2: Odczyt i analiza danych z sensorów.

Zaimplementowane jest również wykrywanie anomalii - odpowiedzialne są za nie funkcje void LightThread(void *params) oraz void AccThread(void *params) odpowiednio dla światła i ruchu.

Kiedy urządzenie znajduje się w stanie bezpiecznym, wykrycie anomalii powoduje wyczyszczenie zawartości karty SD, co potwierdzone jest szybkim miganiem diody urządzenia.

LIB/FTPSERVER/FTPSERVER.H

Plik zawiera implementację serwera FTP.

Możliwe stany serwera FTP:

- RESET - resetowanie połączenia - rozpoczęcie pracy od początku
- WAIT_CONNECTION - oczekiwanie na połączenie
- IDLE - WAIT_USERNAME - oczekiwanie na podanie nazwy użytkownika
- WAIT_PASSWORD - oczekiwanie na podanie hasła
- WAIT_COMMAND - oczekiwanie na podanie komendy

Zdarzenia obsługiwane są w pętli void mainFTPLoop().

Najważniejszą funkcją serwera jest boolean processCommand(String command, String params), zajmująca się przetwarzaniem komend użytkownika.

Obsługiwane komendy:

- ABOR
- CDUP
- CWD
- DELE
- FEAT

- LIST
- MKD
- MLSD
- MODE
- NOOP
- PASV
- PWD
- QUIT
- RETR
- RMD
- RNFR
- RNT0
- STOR
- STRU
- SYST
- TYPE

Pozwala to wykonywać wszystkie podstawowe operacje na serwerze FTP, pod warunkiem wykorzystywania jednego wątku klienta.

LIB/MPU6050/MPU6050.H

Plik jest odpowiedzialny za komunikację z modułem MPU6050 i wykrywanie anomalii dotyczących położenia urządzenia. Po uruchomieniu urządzenie jest kalibrowane, a następnie w pętli obliczane są zmiany w położeniu.

W razie wykrycia zbyt dużych zmian, funkcja `checkForAnomalies()` zwraca `true`, co powoduje uruchomienie działań niszczących dane na karcie, jeśli urządzenie jest w odpowiednim stanie.

LIB/RFIDREADER/RFIDREADER.H

Plik jest odpowiedzialny za komunikację z czytnikiem kart RFID. Zawiera również listę zaufanych kart (legitymacji studenckich i tokenów RFID mogących zmieniać stan urządzenia).

Rozpoznawanie kart następuje w funkcjach `bool verifyLoop()` oraz `bool isValidID(byte uid[10])`.