



## **IMP - Mikroprocesorové a vestavěné systémy**

### **Dokumentácia projektu**

ESP32: Přístupový terminál

# Obsah

<b>1</b>	<b>Úvod</b>	<b>2</b>
<b>2</b>	<b>Klávesnica</b>	<b>2</b>
2.1	Zapojenie klávesnice k ESP-32 . . . . .	3
<b>3</b>	<b>LED diódy</b>	<b>3</b>
3.1	Zapojenie diód k ESP-32 . . . . .	3
<b>4</b>	<b>Popis implementácie</b>	<b>4</b>
4.1	Užívateľ nechce zmeniť prístupový kód . . . . .	4
4.2	Užívateľ chce zmeniť prístupový kód . . . . .	4
<b>5</b>	<b>Záver</b>	<b>4</b>
<b>6</b>	<b>Použitá literatúra</b>	<b>5</b>

## Video k implementácii:

<https://youtu.be/f-X9w9CKGHs>

## 1 Úvod

Úlohou projektu bolo naprogramovať prístupový terminál na vývojovej doske ESP-32.

Ako tlačítka pre vstupný kód nám bude slúžiť klávesnica o rozmeroch 3x4. Zariadenie načíta vstupný 4-miestny kód z klávesnice a ak je správny, rozsvieti zelenú LED diódu. V opačnom prípade zostáva rozsvietená červená LED dióda.

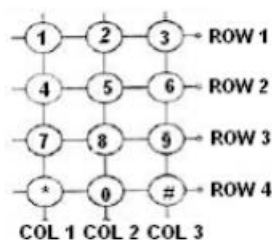
Zariadenie taktiež umožňuje zmenu aktuálneho prístupového kódu na ľubovoľný zadáním príkazu "# aktualnyKod \* novyKod".

## 2 Klávesnica

Klávesnica obsahuje 12 tlačidiel a 7 pinov. Tri piny sú na ovládanie stĺpcov a štyri na ovládanie riadkov klávesnice. Kvôli obmedzenému počtu pinov (7) na počet tlačidiel (12) sa na detekciu stlačenia klávesy musia aktivovať a deaktivovať postupne všetky stĺpce.

K tomu slúži funkcia `static void set_col(int colnb)`, kde `colnb` je číslo stĺpca, ktorý chceme aktivovať. Ostatné deaktivuje. Na začiatku aktivujem výstup na stĺpec COL1, snažím sa detekovať vstup na riadkoch ROW1-4. V prípade ak nedetekujem stlačenie klávesy, stĺpec COL1 sa deaktivuje a aktivuje sa stĺpec COL2. Následne prebehne detekcia vstupu na riadkoch ROW1-4 ako pri prvom stĺpci. Identický postup je aj pre stĺpec COL3.

Celý tento proces prepínania prebieha niekoľkokrát za sekundu stále dookola až pokiaľ sa nedetekuje stlačená klávesa. Po stlačení klávesy je tento proces zastavený a čaká na znovuspustenie pre načítanie ďalšieho znaku - funkcia `waiting_for_button(void)`.



Output Arrangement	
Output Pin Number	Symbol
1	COL 2
2	ROW 1
3	COL 1
4	ROW 4
5	COL 3
6	ROW 3
7	ROW 2

## 2.1 Zapojenie klávesnice k ESP-32

Pre vstupné porty ROW1-4 je zapnutá funkcia "Pull-Down" rezistorov. Použitím "Pull-Down" rezistorov dosiahneme 100% logickú nulu aj pri rozpojenom obvode. Bez ich použitia mikrokontrolér detekuje falošné stláčania tlačidiel vplyvom indukovaného napätia apod.

Klávesnica	ESP-32
COL 1	GPIO 12
COL 2	GPIO 18
COL 3	GPIO 13
ROW 1	GPO 19
ROW 2	GPIO 14
ROW 3	GPIO 12
ROW 4	GPIO 27

## 3 LED diódy

Pre zablikanie červenej diódy používam funkciu `void red_LED_blink()`.

Pre zablikanie zelenej diódy obdobne funkciu `green_LED_blink()`. Oba tieto funkcie fungujú na sekvencii "zasviet-počkaj-zhasni-počkaj" ktorá sa opakuje 5x za sebou. Týmto sa vytvorí efekt blikania diódy. Okrem blikania diód taktiež používam funkciu `turn_on_green_LED()`, ktorá iba zasvieti zelenú diódu na 1 sekundu bez blikania.

### 3.1 Zapojenie diód k ESP-32

Diódy sú zapojené rezistormi 330R k ESP-32:

LED DIODY	ESP-32
Zelená	GPIO 16
Červená	GPIO 17

## 4 Popis implementácie

Po spustení programu dôjde ku konfigurácii jednotlivých GPIO portov. Následne sa spustí nekonečný cyklus v ktorom program čaká na vstup z klávesnice. Nasleduje test, či prvý načítaný zadáný znak je "#". Ak áno znamená to, že užívateľ chce zmeniť aktuálny kód - totiž chce zadávať 10-miestny vstup z klávesnice. V opačnom prípade program čaká na 4-miestny vstup.

### 4.1 Užívateľ nechce zmeniť prístupový kód

Po úspešnom načítaní 4-miestneho kódu program porovná zadáný kód so správnym kódom po jednotlivých čísliciach. Ak je zhoda 100%, dôjde k rozsvieteniu zelenej LED diódy na 1 sekundu a zmazaniu súčasného vstupu od užívateľa - tzv. rest. V opačnom prípade dôjde k zablikaniu červenej LED diódy a vynulovaniu vstupu. Následne môže užívateľ zadávať ďalší nový kód.

### 4.2 Užívateľ chce zmeniť prístupový kód

Po úspešnom načítaní 10-miestneho vstupu od užívateľa dôjde k porovnaniu číslic na indexe 1-4 (index 0 obsahuje znak #) s číslicami správneho prístupového kódu. Ak sa niektoré z číslic nezhodujú, kód nemôže byť zmenený a dôjde k blikaniu červenej LED diódy a zmazaniu súčasného vstupu od užívateľa. Následne môže užívateľ zadať nový vstup (4-miestny kód alebo 10-miestnu sekvenciu ako nový pokus pre zmenu kódu).

Ak sa číslice pôvodného kódu zhodujú s užívateľovým vstupom, kód môže byť úspešne zmenený. Dôjde k prepisu číslic nastaveného kódu na nové a zároveň k rozblikaniu zelenej LED diódy - čo signalizuje, že prístupový kód bol zmenený. Následne môže užívateľ zadať nový vstup (4-miestny kód alebo 10-miestnu sekvenciu ako nový pokus pre zmenu kódu).

Zadávanie rôznych vstupov funguje neobmedzený počet krát a nikdy nedôjde k zastaveniu vykonávania programu.

## 5 Záver

Riešenie projektu je plne funkčné. V projekte boli implementované všetky požiadavky na danú variantu projektu. Behom riešenia a testovania tohto projektu som nezaznamenal žiadne problémy a nedostatky.

## 6 Použitá literatura

- [1] <https://www.baldengineer.com/arduino-keyboard-matrix-tutorial.html>
- [2] <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/>