Univerzitet u Sarajevu
Elektrotehnički fakultet
Ugradbeni sistemi (RI) 2024/25

Dokumentacija implementacije

Tema vježbe: Pametni sigurnosni sistem sa PIN autentikacijom i vizuelnim interfejsom

Članovi tima:
Emina Hamamdžić
Ajdin Kanlić

Sadržaj

1.1 Uvod

Ovaj sistem implementira jednostavan sigurnosni mehanizam koristeći Raspberry Pi Pico. Sistem reaguje na pokret pomoću PIR senzora, aktivira alarm (zvučni signal), i zahtijeva unos sigurnosnog PIN-a za deaktivaciju. Interakcija korisnika se realizuje pomoću matrične tastature i TFT displeja. Pored deaktivacije, omogućena je i promjena PIN koda putem posebnog tastera. Korišteni hardver i softver za rješavanje i realizaciju projekta, sistematski su prikazani u tabeli 1.

Softver	***Hardver***
Thonny IDE	**Raspberry Pi Pico (PicoETF)**
Wokwi simulator	**TFT displej Banggood (ILI9341)**
	Matrična tastatura
	PIR senzor
	Buzzer
	Dodatna 2 tastera

Tabela 1: Pregled korištenog hardvera i softvera.

1.2 Režimi rada sistema

Pametni sigurnosni sistem implementiran na Raspberry Pi Pico platformi koristi tri režima rada: **pasivni režim**, **aktivni režim**, i **režim promjene PIN-a**. Svaki od ovih režima ima jasno definisanu ulogu u okviru funkcionalnosti sistema i omogućava intuitivnu upotrebu od strane krajnjeg korisnika.

1.2.1 Pasivni režim

Pasivni režim je **podrazumijevano stanje sistema**. U ovom režimu:

- Sistem ne reaguje na pokret,
- Zvučni alarm je deaktiviran,
- Na TFT ekranu je prikazana poruka: PASIVNI MOD,
- Uključen je **zelena LED dioda**, što vizuelno označava bezbjedno stanje

.

Ovaj režim omogućava korisniku da, pomoću tastera, pokrene:

- Aktivaciju sistema (prelazak u aktivni režim),
- Promjenu PIN koda (prelazak u režim promjene PIN-a).

Prekid se inicira putem **IRQ-a na tasterima**:

- taster1 pokreće aktivaciju,
- taster2 inicira promjenu PIN-a.

1.2.2 Aktivni režim

Aktivni režim označava da sistem „osluškuje“ stanje, te je spreman da reaguje na pokret. Prepoznaje se po:

- Crvenoj LED diodi (zelena se gasi),
- Poruci na displeju: AKTIVNI MOD.

U ovom režimu, ukoliko **PIR senzor detektuje pokret**, automatski se:

- Aktivira **zvučni alarm** (PWM buzzer),
- Poziva funkcija deaktivacija() koja prikazuje upozorenje i traži unos PIN-a.

Korisnik mora unijeti **tačan PIN kod** da bi se alarm deaktivirao i sistem vratio u pasivni režim. U suprotnom:

- Broj pokušaja se broji do maksimalno 3,
- Nakon 3 neuspjela pokušaja, sistem se **zaključava na 30 sekundi**.

Ova logika je realizovana u petlji koja se izvršava u okviru ***while not pasivni_mod***: bloka.

1.2.3 Režim promjene PIN-a

Treći režim omogućava korisniku da bezbjedno **promijeni postojeći PIN kod**. On se aktivira iz pasivnog režima (nemoguće ga je aktivirati iz aktivnog,odnosno nakon aktivacije alarma), pritiskom na *taster2*. Implementiran je kao posebna grana logike unutar glavne petlje.

Postupak promjene:

1. Traži se trenutni (važeći) PIN kod za autentifikaciju.
2. Ukoliko je tačan, traži se novi PIN i njegova potvrda.
3. Ako su oba nova unosa identična, PIN se ažurira i režim se vraća u pasivno stanje.
4. U slučaju greške (pogrešan trenutni PIN ili neslaganje novih PIN-ova), sistem prikazuje odgovarajuću poruku i vraća se u pasivni režim bez izmjene.

Ovaj režim koristi zastavicu ***promjenaPinaFlag*** da bi se razlikovao od ostalih režima, a kontrola toka se obavlja funkcijom ***promjenaPina()***.

1.3 Implementacija

Nakon detaljne analize i kvalitetnog specificiranja problema, krenuli smo u pisanje i testiranje programskog koda u Wokwi simulatoru, koji nam je poslužio za precizniju ideju i vizualizaciju projekta prije testiranja na stvarnom mikrokontroleru. Izgled projekta i dodanih komponenti u simulatoru prikazan je na slici 1. Prije implementacije programskog koda u laboratoriji morali smo spojiti potrebne hardverske komponente, ranije prikazane u tabeli 1. U toj fazi realizacije projekta, najveći problem nam je predstavljao konflikt između matrične tastature i tastera, ove komponente su koristile iste pinove pa smo se odlučili na dodavanje dva dodatna tastera preko pomoćne ploče. Tastere smo spojili na način opisan u predavanjima, s tim da smo se ,umjesto dodavanja vanjskog otpornika, odlučili na korištenje internog otpornika. Da bismo uspješno realizirali tu mogućnost, u programskom kodu morali smo dodati informaciju da koristimo ***PULL_DOWN*** otpornik, na način prikazan ispod:

```
taster1 = Pin(10, Pin.IN, Pin.PULL_DOWN)
taster2 = Pin(13, Pin.IN, Pin.PULL_DOWN)
```

Kada smo spojili sve potrebne komponente, ostatak realizacije projekta se sveo na kvalitetno pisanje koda. U kodu, nakon inicijalizacije digitalnih ulaza i izlaza, spajanja matrične tastature i uspostavljanja SPI komunikacije za TFT displej, koristili smo pomoćne funkcije poput: ***displayMessage(msg)*** koja ispravno prikazuje string parametar *msg* na displej, ***led_mod(pasivni), scan_keypad(), enter_pin()*** za unos PIN koda putem tastarure, ***deaktivacija(), promjenaPina(),*** koje čine glavne režime programa, ***promijeni_pin(pin), aktivniMod(pin)*** funkcije koje se pozivaju kao handler funkcije vezane za tastere koristeći sistem prekida:

```
taster1.irq(handler = aktivniMod,trigger = Pin.IRQ_RISING)
taster2.irq(handler = promijeni_pin,trigger = Pin.IRQ_RISING)
```

U sklopu programskog dijela, posebnu pažnju smo posvetili obradi tastature, kako bi unos PIN koda bio pouzdan i otporan na višestruka očitavanja uslijed fizičkih karakteristika tipki. Funkcija *scan_keypad()* koristi princip aktivacije po redovima i očitavanja kolona, uz dodatno vremensko kašnjenje za eliminaciju tzv. "*bounce*" efekta. Time je omogućeno stabilno i tačno detektovanje svakog pojedinačnog pritiska. Za bolji korisnički doživljaj, unos svake cifre PIN-a se maskira i prikazuje na TFT displeju u obliku zvjezdica, čime se dodatno doprinosi sigurnosti sistema.

Pri detekciji pokreta, sistem odmah reaguje uključivanjem alarma (*buzzer*) putem PWM signala. Ova funkcionalnost implementirana je korištenjem *duty_u16* parametra, čime se alarm može aktivirati i deaktivirati bez promjene frekvencije. Kada je sistem u alarmnom stanju, korisniku se nudi mogućnost unosa PIN-a za deaktivaciju. Ukoliko unese pogrešan PIN, broj pokušaja se bilježi, a nakon tri neuspjela pokušaja sistem se zaključava na unaprijed definisano vrijeme od 30 sekundi, bez potrebe za dodatnim hardverom za brojače ili tajmere.

Jedna od naprednijih funkcionalnosti implementiranih u sistem je i mogućnost promjene sigurnosnog PIN-a. Ova opcija je omogućena samo korisnicima koji poznaju postojeći PIN, čime se sprječava neovlašteno mijenjanje sigurnosnih podataka. Cijeli proces promjene PIN-a vodi

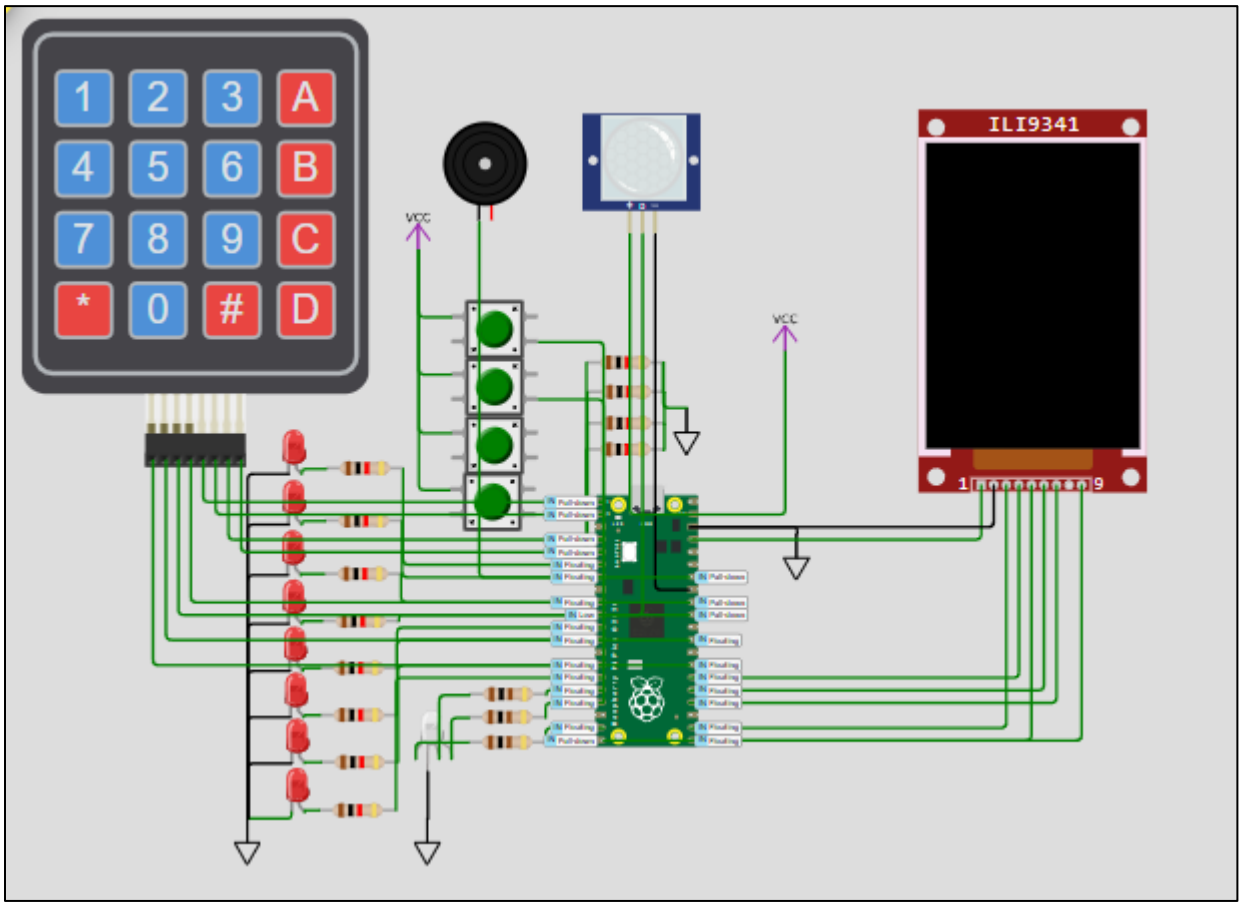
korisnika kroz tri uzastopna unosa – postojeći PIN, novi PIN, te potvrdu novog PIN-a. Svaka faza je jasno prikazana korisniku putem poruka na TFT ekranu, čime se obezbjeđuje jednostavno i intuitivno korištenje.

Svi režimi rada – **pasivni**, **aktivni**, **alarmni**, te režim za **promjenu PIN-a** – povezani su jasno definisanim *zastavicama* **(**flag varijablama**)** ***pasivni_mod*** i ***promjenaPinaFlag***. Umjesto korištenja složenih višezadaćnih mehanizama, odlučili smo se za jednostavnu i efikasnu kontrolu toka izvršavanja programa pomoću uslovnog grananja u glavnoj beskonačnoj petlji. Ove logičke zastavice omogućavaju da se sistem u svakom trenutku ponaša tačno prema aktivnom režimu, pri čemu:

- ***pasivni_mod*** označava stanje pripravnosti sistema. U tom stanju PIR senzor se ignoriše, a sistem čeka aktivaciju prekidom.
- ***promjenaPinaFlag*** se koristi isključivo za upravljanje režimom promjene PIN-a. Postavlja se na *True* prilikom pritiska drugog tastera i automatski se resetuje nakon izvršene promjene.

Na ovaj način osigurano je **determinističko i sekvencijalno ponašanje** sistema u svakom trenutku, bez mogućnosti neželjenog preklapanja režima rada. Upravo takav pristup upravljanju stanjima čini ovaj sistem robusnim i pouzdanim za realne aplikacije.

Cijela arhitektura koda oslanja se na modularnost – svaka funkcionalnost je jasno izdvojena u posebnu funkciju (enter_pin, deaktivacija, promjenaPina, scan_keypad itd.), što omogućava lakše testiranje, održavanje i eventualne nadogradnje sistema u budućnosti.



Slika 1: Prikaz dodanih komponenti u Wokwi simulatoru

1.4 Zaključak

Implementacijom ovog sistema uspješno je demonstrirana upotreba mikrokontrolera Raspberry Pi Pico u funkciji jednostavnog, ali funkcionalnog sigurnosnog uređaja. Kroz pažljivo planiranje, modularnu strukturu programskog koda i jasno definisane režime rada, postignut je pouzdan mehanizam zaštite koji korisniku nudi vizuelnu i zvučnu povratnu informaciju, kao i mogućnost fleksibilnog upravljanja PIN kodom.
Rješenje se odlikuje jednostavnošću u korištenju, ali i tehničkom ozbiljnošću u pogledu upravljanja stanjima, rukovanja prekidima, obradom ulaznih podataka sa tastature, kao i integracijom sa TFT displejem. Poseban fokus stavljen je na robusnost i otpornost na greške, što čini sistem pogodnim za proširenja i integraciju u složenije scenarije. Realizovani sistem je lako prilagodljiv i može poslužiti kao osnova za buduće nadogradnje, poput bežične kontrole, autentifikacije preko RFID/NFC tehnologije ili povezivanja sa mobilnom aplikacijom. Time se potvrđuje da je krajnji cilj ovog rada – funkcionalan, stabilan i edukativno vrijednostan sigurnosni sistem – u potpunosti ostvaren.