

Универзитет у Бањој Луци  
Електротехнички факултет  
Програмирање у реалном времену

Пројектни рад на тему  
**Безбједносни систем за пословне и личне просторе**  
(енг. *Home Security System*)

Аутори: Ана Вулин, Ања Таковић, Радислав Косијер, Дејана Смиљанић  
Професори – ментори: доц. др Срђан Попић, проф. др Младен Кнежић  
Академска 2023/24.

## Садржај:

1. Увод .....	3
1.1 Опис проблема.....	3
1.2 Циљеви пројектног задатка .....	3
1.3 Опсег пројекта .....	3
2. Технологије и алати .....	4
2.1 Развојни алати.....	4
2.2 Софтверски алати .....	4
2.3 Оперативни систем.....	5
3. Архитектура система .....	5
3.1 Дијаграм архитектуре .....	5
3.2 Коришћене компоненте и њихов опис .....	5
4. Детаљан опис система .....	7
4.1 Сензори, диоде и пиезо звучник .....	7
4.2 Остваривање комуникације и контроле .....	8
4.3 Управљање у реалном времену.....	8
4.4 Интеракција са корисником .....	9
4.5 Физичко повезивање система.....	9
5. Имплементација .....	9
5.1 Груба анализа.....	9
5.2 Фрагменти значајнијих дијелова кода.....	10
6. Закључак .....	13
6.1 Анализа постигнутих функционалности .....	13
6.2 Проблеми.....	13
6.3 Потенцијална побољшања.....	13
7. Референце .....	13
8. Прилози.....	14

## 1. Увод

### 1.1 Опис проблема

У савременом пословном окружењу и све чешћим „паметним“ домаћинствима, сигурност објеката постаје један од кључних аспеката за заштиту имовине, запослених и повјерљивих података. Са порастом броја случајева неовлашћених приступа и провала, јавља се потреба за ефикасним системима који могу правовремено детектовати и реаговати на потенцијалне пријетње. Традиционални сигурносни системи често захтијевају присуство обезбјеђења или комплексну инфраструктуру, што може бити скупо и тешко одрживо за мања предузећа или објекте личне намјене.

Проблем који овај пројекат рјешава односи се на потребу за једноставним, а ефикасним сигурносним рјешењем које може аутоматски детектовати присуство особе, провјерити њене приступне податке, и у случају неовлашћеног покушаја приступа, аутоматски покренути сигурносне мјере, као што су активација аларма и обавјештавање одговорне особе.

Комбиновањем *Raspberry Pi 3* рачунара са сензорима присуства, графичким интерфејсом за унос лозинке и камером, "*Home Security Систем*" представља приступачно и аутоматизовано рјешење за заштиту пословних или личних објеката. Тиме се елиминишу потребе за сталним надзором, а осигурава се брз и адекватан одговор на пријетње, чиме се минимизира ризик од провала и неовлашћеног приступа.

### 1.2 Циљеви пројектног задатка

Главна идеја при избору теме пројектног задатка била је та да се обухвати градиво обрађено током семестра, а које се тиче крос компајлинга, писање драјвера у кернел простору и њихово коришћење, те смислено повезивање система који треба да одговори на тражене захтјеве у реалном времену. Имплементација оваквог система била је погодна из разлога што је сама тема приступачна и општепозната, може се примијенити у пракси, те на добар начин прилагодити потребама самог курса предмета. Уз градиво које је обрађено користе се и комуникациони протоколи и знање из електронике, чиме је остварено повезивање знања из више различитих области.

### 1.3 Опсег пројекта

Овај пројекат обухвата развој сигурносног система за пословни објекат који укључује следеће компоненте и функционалности:

#### 1. Хардверске компоненте:

- *Raspberry Pi 3B V1.2* као централна контролна јединица.
- Сензор за детекцију покрета врата (*ADC12 click* и конвертор).
- Звучни сигнал (пиезо звучник) који ће се активирати у случају неуспешне валидације приступа.
- Црвено свјетло (LED диода) које се активира при алармантној ситуацији.
- Зелено свјетло (LED диода) које се активира након уношења исправних приступних података.

- Камера, за снимање фотографије у случају неовлашћеног приступа.

## 2. Софтверске функционалности:

- Детекција присуства: Сензор детектује присуство особе када уђе у објект.
- Валидација лозинке: Корисник мора унијети тачну лозинку у року од 30 секунди. У случају тачног уноса, систем се деактивира.
- Активација аларма: Ако корисник унесе погрешну лозинку или не унесе лозинку у предвиђеном времену, систем активира аларм и обавјештава шефа обезбјеђења путем СМС-а или е-поште.
- Слање фотографије: Уз поруку шефу обезбјеђења шаље се и фотографија особе која покушава неовлашћено приступити.

## 3. Интерфејс за обавјештавање:

- СМС или е-пошта за обавјештавање одговорне особе о потенцијалном безбједносном инциденту.

## 4. Тестирање у реалним условима:

- Тестирање функционалности у условима сличним стварним сценаријима (детекција присуства, валидација лозинке, активација аларма).

Пројекат не обухвата:

- Напредне методе верификације као што су биометрија.
- Могућност даљинског управљања системом преко мобилне апликације.
- Интеграцију са постојећим системима за надзор.

Циљ је да се имплементира основни сигурносни систем који је флексибилан и може се касније надоградити са додатним функционалностима, у складу са специфичним потребама корисника.

## 2. Технологије и алати

### 2.1 Развојни алати

Централни развојни алат је *Raspberry Pi 3B V1.2*, преко чијих се *GPIO* пинова врши комуникација и повезивање дијелова система. Њиме се омогућава хардверско везивање цјелина и спровођење комуникационих протокола на физичком слоју. Од хардверских компонената користе се још и *LED* диоде, пиезо звучник, *ADC12* кликер и *ADC* конвертор.

### 2.2 Софтверски алати

Код је написан у програмском језику *C*. Кориштено интегрисано радно окружење је *Visual Studio Code*, те *Notepad++* са *plug-in* – ом за *C* језик. За компајлирање се користи *GCC (GNU Compiler Collection)* верзије 12.2.0. Да би се омогућио рад са *GPIO* пиновима користи се *gpio.h* библиотека. Интеракција са *Raspberry Pi* се одвија путем командне линије.

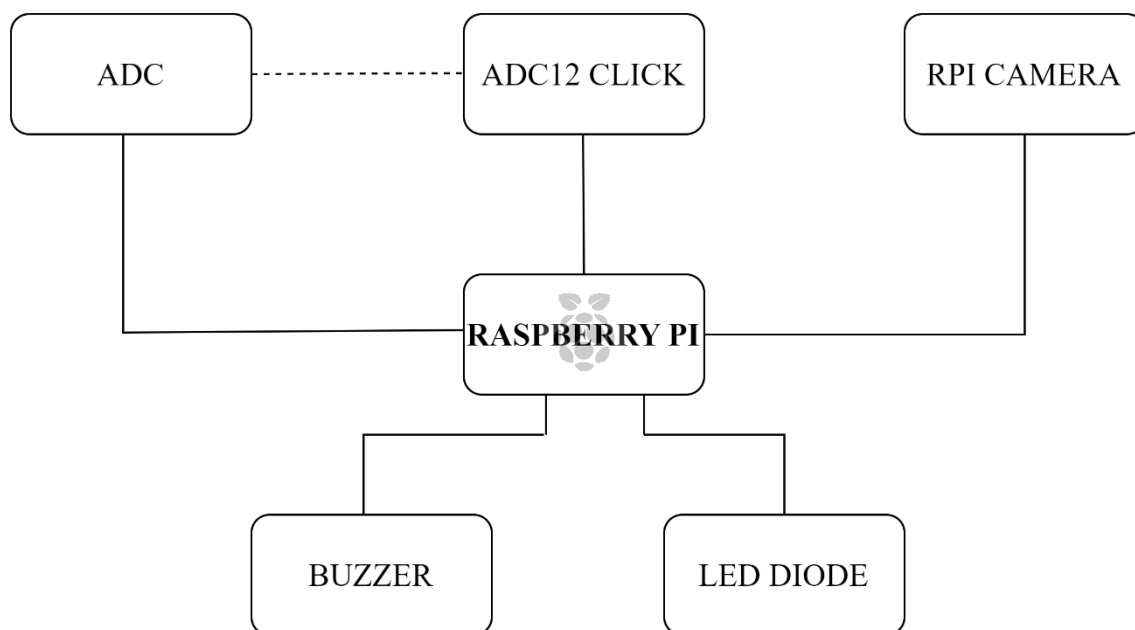
## 2.3 Оперативни систем

Инсталирани оперативни систем је *Raspbian*, 32 – битни, верзије 6.6.51.

## 3. Архитектура система

### 3.1 Дијаграм архитектуре

На Слици 1. дат је дијаграм архитектуре.



Слика 1. Дијаграм архитектуре

Централна компонента је *Raspberry Pi* који остварује везу између свих дијелова система. Комуникација се одвија путем *GPIO* пинова (за звучник, диоде и *ADC*), користећи *i2c* комуникациони протокол (за *ADC*). Контрола диода, звучника и АД конвертора одвија се путем драјвера, гдје је тачно дефинисано шта и како која компонента треба да извршава. Како су драјвери дио кернел простора, потребно је имплементирати корисничке апликације које ће комуницирати са кернелом, а за шта су имплементиране заједничке и појединачне тестне апликације.

Архитектура система одговара опису проблема - сваки дио треба да пружи могућност интеграције у систем, те упоредо да функционише као засебна (под)цјелина.

### 3.2 Кориштене компоненте и њихов опис

У Табели 1. приказане су кориштене компоненте, те њихов груби опис. За сваку од компонената кориштен је одговарајући *datasheet* и описи произвођача, а који ће бити излисати у одјељку Референце.

Назив:	Вриједност:	Опис:
Raspberry Pi 3B	/	Процесор: ARM Cortex-A53 64-битни четворојезгарни процесор.

		<p>Тактна фреквенција: 1.2 GHz.</p> <p>Меморија: 1 GB LPDDR2 SDRAM.</p> <p>Складиште: microSD картица за инсталацију оперативног система и складиштење података.</p> <p>Портови: USB портови: 4 x USB 2.0 порта.</p> <p>HDMI порт: Стандардни HDMI за повезивање са мониторима и ТВ-има.</p> <p>Ethernet: 10/100 Mbps Ethernet порт.</p> <p>Audio: 3.5mm аудио излаз.</p> <p>Бежична конекција: Вг интегрисан Wi-Fi 802.11n.</p> <p>Вг интегрисан Bluetooth 4.2 за повезивање са бежичним уређајима.</p> <p>GPIO: 40-пински GPIO заглавље, компатибилан са свим претходним моделима, за повезивање разних сензора и уређаја.</p> <p>Оперативни систем: Подржава Raspberry Pi OS (раније познат као Raspbian), али може да покреће и друге Linux дистрибуције као што су Ubuntu Mate, OSMC, LibreELEC, итд.</p>
<b>LED црвена</b>	$\lambda = 635\text{nm}$	<p>Црвена LED (Light Emitting Diode) је полупроводничка компонента која емитује свјетлост када кроз њу пролази струја (црвене боје).</p> <p>Карактеристике: Напон: Обично између 1.8V и 2.2V. Струја: Обично 20 mA.</p> <p>Примјена: Сигнализација грешке или недозвољеног стања.</p>
<b>LED жута</b>	$\lambda = 590\text{nm}$	<p>Карактеристике: Напон: Обично између 2.0V и 2.5V. Струја: Обично 20 mA.</p> <p>Примјена: Често се користи у семафорима или као индикатор упозорења.</p>
<b>LED зелена</b>	$\lambda = 520\text{nm}$	<p>Карактеристике: Напон: Обично између 2.0V и 3.5V. Струја: Обично 20 mA.</p> <p>Примјена: Често се користи у разним уређајима као статусна индикација (за означавање успеха неке радње).</p>
<b>ADC</b>	/	<p>АДЦ је уређај који конвертује аналогни сигнал у дигитални сигнал. Карактеристике: Резолуција: Изражава се у битима (нпр. 8-бит, 10-бит, 12-бит). Максимална фреквенција: Како брзо може да конвертује аналогни сигнал.</p> <p>Примјена: Користи се у пројектима гдје је потребно мјерење напона, температуре, влажности итд.</p>
<b>ADC12 click</b>	/	<p>ADC12 Click је додатна плоча (click board) за Raspberry Pi или MikroBUS платформе, која садржи 12-битни АЦД (Analog-to-Digital Converter) уређај.</p> <p>Карактеристике: Резолуција: 12 бита, што омогућава прецизно мјерење. Интерфејс: Обично се комуницира преко I2C или SPI.</p>

		Примјена: Често се користи за мјерење температура, напона или других аналогних сигнала.
<b>Buzzer</b>	$I \geq 85\text{dB}$	Buzzer је уређај који производи звук када пролази струја кроз њега. Карактеристике: Напон: Обично ради на 5V. Тон: Може да произведе различите фреквенције у зависности од начина рада (активни или пасивни). Овдје: активни. Примјена: Користи се у алармима, сигнализацији, обавјештењима и другим пројектима гдје је потребно производити звук.
<b>Пасивне компоненте</b>	$\sim 270\Omega$	Отпорници који служе за ограничавање струје, у конфигурацији pull-down.
<b>Везице (жице)</b>	/	Проводничка компонента која омогућава спајање и исправан електрични рад система.

Табела 1. *Коришћене компоненте и њихов опис*

#### 4. Детаљан опис система

Детаљнији преглед система, његових цјелина и механизма остваривања тражених функционалности дат је у наредним потпоглављима (4.2 - 4.4).

##### 4.1 Сензори, диоде и пиезо звучник

*ADC12 click* генерише напон на свом излазу који је осјетљив на покрет изнад сензора покрета. Оптимална осјетљивост сензора постиже се постављањем сензора на максимално 15 центиметара <sup>1</sup>од објекта од значаја (у овом случају плафон и горњи дио врата). Покрет се препознаје на основу промјене напона који се генерише на излазу. Да би се одредила вриједност напона која ће се генерисати у условима када нема покрета (назовимо то стање мировања), потребно је одредити доње и горње границе мјерења сензора и смјестити вриједност из стања мировања у добијени опсег.

Када се подесе основни параметри, потребно је остварити комуникацију са сензором и дјеловање у складу са његовим мјерењима. Пошто је напон на излазу сензора аналогни, користи се 12-битни АД конвертор који претвара аналогни напон у дигитални.

АД конвертор комуницира са *Raspberry Pi* путем i2c комуникационог протокола. Детаљи комуникације биће објашњени у наредном потпоглављу. Оно што је битно јесте да ће се на *Raspberry Pi* 12-битни податак конвертовати у 16-битни податак, који је лакши и за репрезентацију и за манипулацију у раду. За АД конвертор написан је драјвер у кернелу који се односи на све аспекте рада конвертора – пријем података, обраду и тумачење података, те генерисање потребних акција и одговора.

Након детекције покрета покреће се систем акција које се предузимају у случају да је детектовано присуство особе у простору. Жута диода се укључује на двије секунде и сигнализира да је започео процес уношења и валидације лозинке, те одбројавање. Особи се приказује екран на ком може да се региструје као запосленик или станар (у зависности од области примјене). У имплементираном систему се користи тестна, замјенска

<sup>1</sup> На основу података из datasheet-a

апликација која комуницира са корисником путем командног промпта. За практичне употребе било би потребно или интегрисати механичко-електричну тастатуру у систем или неку врсту екрана на ком би корисник могао да комуницира са системом.

Корисник има право да уноси лозинку у оквиру дефинисаног временског интервала. Овдје се ради од 30 секунди. Уколико корисник у том оквиру унесе тачну лозинку, укључује се зелена диода на двије секунде која сигнализира да је лозинка тачна и приступ омогућен. Уколико се унесе погрешна лозинка, тада се укључује црвена диода на двије секунде. Омогућено је уношење лозинке све док не истекне дефинисани период.

Након истека периода уношења и валидације лозинке, пали се пиезо звучник на десет секунди који оглашава стање покушаја неовлаштеног приступа, те се камером снима фотографија и шаље шефу обезбјеђења.

Манипулација кориштеним компонентама се врши помоћу функција дефинисаних унутар драјвера за сваку од њих. Дакле, функције *read*, *write*, *open*, *exit*, *release*, *stop*, *start* које су имплементирани у драјвер се позивају када је то потребно, што је прилично једноставан и лако разумљив процес, погодан за објашњавање и евентуално приближавање крајњем кориснику.

#### 4.2 Остваривање комуникације и контроле

Да би систем функционисао у складу са очекивањима, потребна је правовремена и неометана комуникација.

Директна комуникација између *Raspberry Pi*-а и диода је директна и одвија се путем *GPIO* пинова. Када је потребно да се диода укључи, вриједност пина се директно сетује / поставља на висок логички ниво. То је омогућено због уграђених функција у оквиру библиотеке *gpio.h*. Мапирање пинова се врши кроз драјвер, при чему је потребно водити рачуна о распореду пинова, њиховом сетовању за одређени режим рада и сл. Манипулација стањем диода је једноставна; потребно је позвати одређене, предефинисане функције и водити рачуна о ослобађању ресурса.

Исти тип комуникације се користи и за пиезо звучник.

Када је ријеч о АД конвертору, ту се користи *i2c* комуникација. АД конвертор је повезан на портове, помоћу којих доставља податке *Raspberry Pi*.

#### 4.3 Управљање у реалном времену

Систем је пројектован као систем за рад у реалном времену. Из тог разлога мора да се извршава у оквиру одређених рокова. Конкретно, овај систем може да трпи евентуална кашњења у одговору јер само вријеме слања и примања одговора није од примарног значаја. Да би се обезбиједила поузданост и тачност, за временске захтјеве користи се високо прецизни уграђени тајмер *Raspberry Pi* - а, који се путем системских позива може покренути или зауставити. Да би се омогућило повезивање путем системских позива, написан је драјвер који мапира функције које треба да се позову када корисничка апликација у системски фајл упише наредбу.

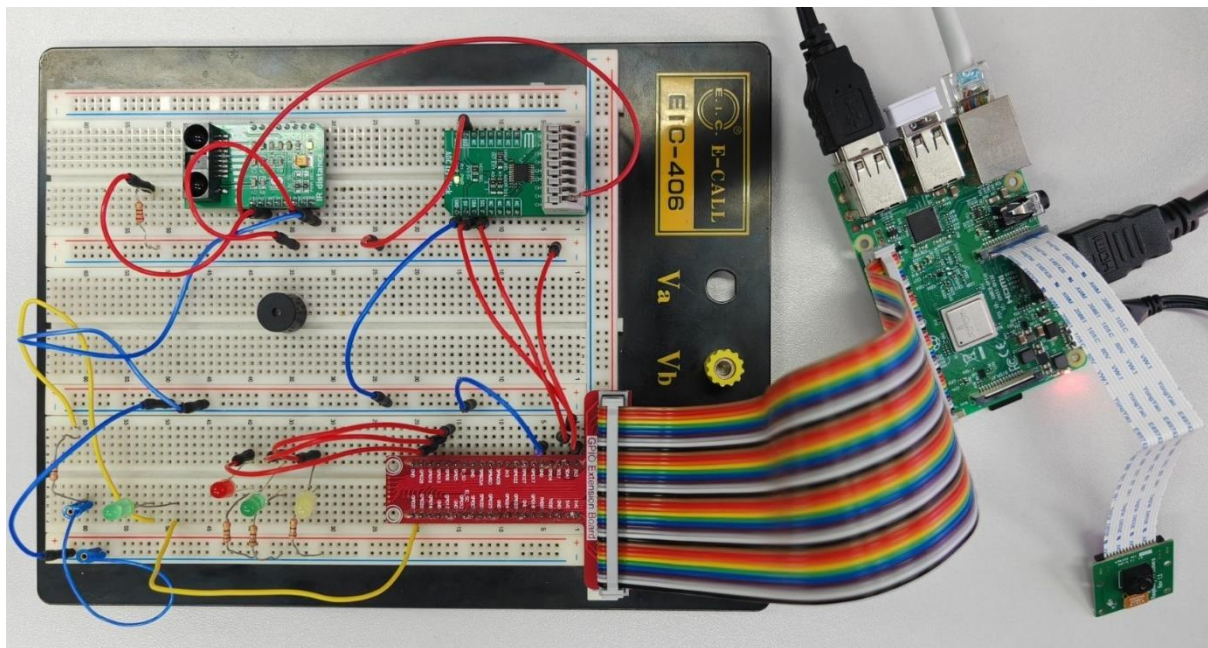


#### 4.4 Интеракција са корисником

Интеракција са корисником је (у имплементираном систему) на ниском нивоу. За потребе овог система, потребно је да је интерфејс једноставан и употребљив, пошто имплементира само унос лозинке.

#### 4.5 Физичко повезивање система

На Слици 2. је приказана шема система, са повезаним компонентама.

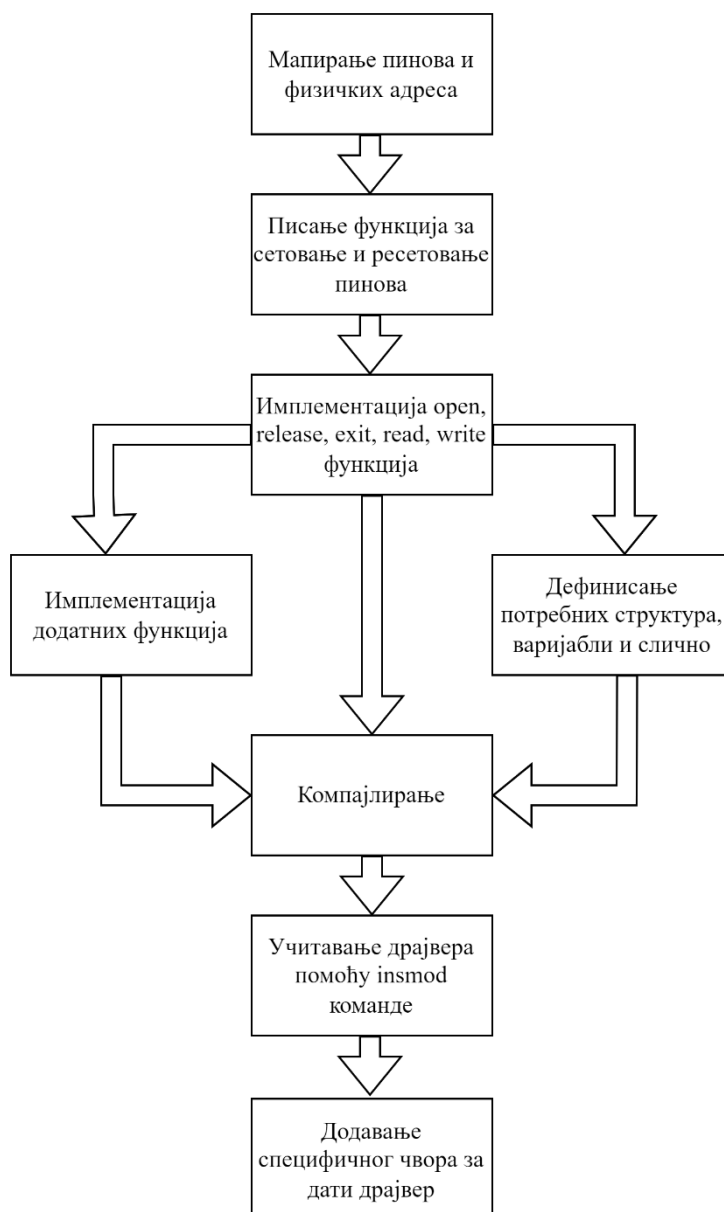


Слика 2. Систем у реалном времену, са приказаним повезаним компонентама

### 5. Имплементација

#### 5.1 Груба анализа

Драјвери се реализују у кернелу. Њихова реализација има неколико заједничких сегмената. На Слици 3. је приказан општи процес имплементације.



Слика 3. Основни кораци при писању драјвера

## 5.2 Фрагменти значајнијих дијелова кода

Приказаћемо неке од значајнијих дијелова кода, који осликавају појединачне функционалности.

У Листингу 1. је приказан дио који се односи на комуникацију између *Raspberry Pi* и АД конвертора. Прво се шаље захтјев за успостављањем комуникације, потом се чека једну секунду на одговор и започиње пријем података са сензора. Тада започиње пријем података. Читање података се врши двадесет пута, а потом се тражи средња вриједност добијених података, зарад тачности и прецизности.

```
/* Slanje poruke za pokretanje ADC konverzije */
i2c_master_send(i2c_client_device, &INIT_ADC_CONVERSION_MESSAGE, 1);

hrtimer_start(&mytimer, ktime_set(1,0), CLOCK_MONOTONIC);

/* Citanje podataka sa senzora (I2C klijenta) u 2B buffer */
i2c_master_recv(i2c_client_device, digital_voltage_value, 2);

i2c_master_send(i2c_client_device, &SHUTDOWN_ADC_CONVERSION_MESSAGE, 1);

data_size = strlen(digital_voltage_value);
uint16_t temp = digital_voltage_value[0]<<8 | digital_voltage_value[1];
avg += temp;
```

Листинг 1. *Део функције за читање који се односи на комуникацију путем i2c протокола*

Укључивање диода се врши паралелно покретању тајмера, односно, истовремено се провјерава и која диода треба да се укључи. Боја диода се бира у корисничком простору, те се на основу *write* функције сигнализира која диода треба да се укључи.

```
enum hrtimer_restart timer_callback(struct hrtimer* timer)
{
    printk(KERN_INFO "hrtimer handler called!\n");
    if (strcmp(flag,"YELLOW")==0){
        printk(KERN_INFO "YELLOW LED (GPIO_06) ON\n");
        ClearGpioPin(GPIO_06);
    }
    else if (strcmp(flag,"GREEN")==0){
        printk(KERN_INFO "GREEN LED (GPIO_13) ON\n");
        ClearGpioPin(GPIO_13);
    }
    else if (strcmp(flag,"RED")==0){
        printk(KERN_INFO "RED LED (GPIO_19) ON\n");
        ClearGpioPin(GPIO_19);
    }
    kfree(flag);
    return HRTIMER_NORESTART;
}
```

Листинг 2. *Укључивање диода уз паралелно покретање тајмера*

Да би се звучник огласио, потребно је да се из корисничког простора експлицитно позове *write* функција, након чега се звучник укључује на сто секунди. На тај начин се сигнализира да је дошло до покушаја неовлашћеног приступа. Део имплементације је приказан у Листингу 3.

```
static void buzz(int duration)
{
    double period = 1 / FREQUENCY;
    period *= MULTIPLIER;
    long int end_time = jiffies + msecs_to_jiffies(duration);
    while (time_before(jiffies, end_time))
    {
        SetGpioPin(GPIO_21);
        hrtimer_start(&mytimer, ktime_set(1,0), CLOCK_MONOTONIC);
    }
}
```

Листинг 3. Укључивање пиезо звучника

Заустављање тајмера се врши у засебном драјверу који служи за комуникацију између кернел и корисничког простора. Имплементација у комуникационом и кернел драјверу тајмера је приказана у Листингу 4. и Листингу 5.

```
static ssize_t stop_timer_store(struct kobject *kobj, struct kobj_attribute *attr, const char *buf, size_t count) {
    if (strncmp(buf, "stop", 4) == 0) {
        // Implement your timer stop functionality here
        printk(KERN_INFO "Timer stopped\n");
    }
    return count;
}
```

Листинг 4. Део драјвера који приказује имплементацију стопирања тајмера

```
static ssize_t stop_timer(struct device *dev, struct device_attribute *attr, const char *buf, size_t count)
{
    printk(KERN_INFO "Timer interrupted by user.\n");
    running = false;
    hrtimer_cancel(&countdown_timer);
    running = 0;
    return count;
}
```

Листинг 5. Заустављање тајмера у драјверу

Снимање фотографије након што се активира аларм је имплементирано кроз функцију приказану на Листингу 6. Ток извршавања је везан за родитељски и процес дијете, при чему је потребно вршити провјере да ли је процес завршио и како је завршио.

```
void takePic (char* filename)
{
    //ako je povratna vrijednost fork()=0, nalazimo se u child procesu
    if((pid = fork()) == 0){
        execl("/usr/bin/rpicam-still",
            "/usr/bin/rpicam-still",
            "-n",
            "-o",
            filename,
            NULL);
        //Ako execl ne uspije, exit sa greskom
        exit(1);
    }
```

```
    }

    /*ako je povratna vrijednost fork() pozitivno, znaci da se nalazimo
    u parent procesu i povratna vrijednost je PID child procesa
    */
    else if (pid > 0){
        int status;
        waitpid(pid, &status, 0); // Cekaj da child proces zavrshi
    }
    else{
        // Ako fork ne uspije
        perror("Fork failed");
    }
}
```

Листинг 6. Функција задужена за просљеђивање фотографије

## 6. Закључак

### 6.1 Анализа постигнутих функционалности

Овај пројекат је имао за циљ имплементацију основног сигурносног система за пословне и личне просторе, заснованог на *Raspberry Pi 3* платформи и различитим сензорима и компонентама. Систем је успјешно реализован, омогућавајући детекцију неовлашћеног приступа, валидацију корисничких података и активацију аларма у реалном времену. Кроз развој пројекта, остварени су сви постављени циљеви – укључене компоненте су функционално повезане, а систем је функционалан.

### 6.2 Проблеми

Уочене проблеме изложићемо по ставкама:

- Недостатак заштите система
- Осјетљивост на промјене у радном амбијенту
- Недостатак интеграције са стварним системом

### 6.3 Потенцијална побољшања

Имплементирани систем може послужити као основа за комплетније и поузданије сигурносне системе. Нека од побољшања су свакако могућност напредније валидације приступних података (према отisku, скенирањем картице, биометријом...), омогућење аутоматског закључавања након нетачних приступних података и слично. Прије свега, било би потребно заштити и сами хардверски дио од утицаја околних фактора. Како је за напредније концепте потребно више компонената него што је било доступно, функционалност је морала бити ограничена у складу са тим.

## 7. Референце

[1] Б. Докић, Дигитална електроника, [2012], [АКАДЕМСКА МИСАО, Бул. Краља Александра 160, Београд, ЕЛЕКТРОТЕХНИЧКИ ФАКУЛТЕТ, Патре 5, Бањалука

[2] <https://docs.rs-online.com/7e1f/A700000009001732.pdf>, датум приступа: 27.09 – 10.10.2024, уз пратеће документе

[3] [https://product.tdk.com/en/system/files/dam/doc/product/sw\\_piezo/sw\\_piezo/piezo-buzzer/catalog/piezo-electronic\\_buzzer\\_ps\\_en.pdf](https://product.tdk.com/en/system/files/dam/doc/product/sw_piezo/sw_piezo/piezo-buzzer/catalog/piezo-electronic_buzzer_ps_en.pdf), датум приступа 25.09.2024.

## 8. Прилози

- *Github* репозиторијум: <https://github.com/AnjaDj/Home-Security-System.git>
- Контакт информације:  
Ања Ђаковић: [anja.djakovic@student.etf.unibl.org](mailto:anja.djakovic@student.etf.unibl.org)  
Ана Вулин: [ana.vulin@student.etf.unibl.org](mailto:ana.vulin@student.etf.unibl.org)  
Дејана Смиљанић: [dejana.smiljanic@student.etf.unibl.org](mailto:dejana.smiljanic@student.etf.unibl.org)  
Радислав Косијер: [radislav.kosijer@student.etf.unibl.org](mailto:radislav.kosijer@student.etf.unibl.org)