



# MATURITNÍ PRÁCE

Informatika

## Zařízení pro realizaci chytré domácnosti

Vladislav Aulich 4.C

Prohlašuji tímto, že jsem práci vypracoval samostatně pod vedením Emila Milera a uvedl v seznamu literatury veškerou použitou literaturu a další zdroje včetně internetu.

Prohlašuji rovněž, že tištěná a elektronická verze této práce jsou shodné.

V Praze dne 9. března 2021

---

Podpis autora

## **Poděkování**

Touto cestou bych rád poděkoval svému vedoucímu práce Emilu Milerovi za podporu a odbornou pomoc při tvorbě projektu.

## **Anotace**

Tento projekt se zabývá realizací chytré domácnosti. Pro tvorbu projektu byl zvolen mikrokontrolér Arduino uno a čip ESP32. Komunikace mezi zařízeními je realizovaná bezdrátově za použití otevřené rádiové frekvence 433 MHz. Součástí projektu je i výroba hardware s použitím metody 3D tisku na tiskárně Ender 3 pro. Vývoj kódu probíhal v prostředí Arduino IDE, pro trasování změn byl využit verzovací systém git.

# Obsah

<b>I</b>	<b>Popis projektu</b>	<b>7</b>
<b>1</b>	<b>Zařízení</b>	<b>7</b>
1.1	Centrála . . . . .	7
1.1.1	Komunikace s uživatelem . . . . .	7
1.1.2	Komunikace mezi zařízeními . . . . .	8
1.1.3	Správa uložených zařízení . . . . .	9
1.2	Koncové zařízení . . . . .	10
<b>2</b>	<b>Použitý hardware</b>	<b>10</b>
2.1	Kalkulace nákladů . . . . .	11
<b>3</b>	<b>Testování</b>	<b>12</b>
<b>4</b>	<b>Automatizace</b>	<b>12</b>
<b>5</b>	<b>Porovnání s existujícími projekty</b>	<b>12</b>
<b>6</b>	<b>Možnosti rozšíření</b>	<b>12</b>
<b>II</b>	<b>Uvedení do provozu</b>	<b>12</b>



## Úvod

Toto téma jsem si zvolil, protože jsem chtěl blíže prozkoumat práci s platformou Arduino a ESP32. S programováním těchto zařízení jsem měl minimální zkušenosti, proto pro mě byla práce na projektu výzvou k objevování nového. Použití komunikace na rádiové frekvenci jsem zvolil z důvodu široké škály použití a velkého množství příkladů. Zároveň jsem měl doma nevyužívaný ovladač pracující s touto frekvencí.

Motivací k výběru tématu „chytré domácnosti“ mi bylo její čím dál větší nasazování v domácnostech a snaha vytvořit si ji po svém. Na mnohých komerčních řešeních mi totiž nevyhovoval způsob ovládání, stejně jako velký zásah do soukromí uživatelů.

## Část I

# Popis projektu

## 1 Zařízení

Projekt se skládá ze třech hlavních zařízení:

- Centrály - brány ovládající další zařízení
- Koncového zařízení ovládajícího spotřebič
- Analyzátoru kódu komerčních zásuvek

Jejich vzájemná komunikace probíhá na frekvenci 433,92 MHz.

### 1.1 Centrála

Centrála zajišťuje několik funkcí:

- Komunikace s uživatelem
- Komunikace mezi zařízeními
- Správa uložených zařízení

#### 1.1.1 Komunikace s uživatelem

Pro komunikaci s uživatelem je vytvořeno jednoduché webové rozhraní umožňující dynamicky vypsat uložená zařízení a přidat nové. Pro přidání nového zařízení je nutná autentifikace. Webové



rozhraní je přístupné po zadání adresy `http://esp32.local` nebo po zadání přidělené IP adresy.

Pro přidání nového zařízení má uživatel na výběr mezi přidáním koupeného komerčního zařízení nebo přidáním zařízení vyrobeného v rámci tohoto projektu. Postup při zadávání nového zařízení je nastíněn ve webovém rozhraní.

Komunikace programu a webového rozhraní je zajištěna pomocí HTTP GET requestů, které program zpracuje a provede potřebné akce.

#### **1.1.2 Komunikace mezi zařízeními**

Komunikace mezi zařízeními je realizovaná bezdrátově po frekvenci 433 MHz. Tento druh jsem zvolil z důvodu rozšířenosti a nízkých pořizovacích nákladů modulů.

Ovládání modulů jsem chtěl realizovat pomocí jednoduché knihovny VirtualWire, ale zjistil jsem, že není funkční na čípech ESP. K ovládání jsem tak použil knihovnu RadioHead.

Pro ovládání „Koncového zařízení“ je vyslána zpráva obsahující ID zařízení a tag ON nebo OFF, „koncové zařízení“ následně odchytí ID a provede příkaz. Tento systém je do budoucna rozšiřitelný o další tagy pro zařízení, které potřebují k ovládní více příkazů než ON nebo OFF.

Do projektu jsem chtěl přidat i možnost ovládání komerčních

zařizování. K tomuto účelu jsem začlenil knihovnu RC switch, která umožňuje zobrazit protokol, na jehož základě zařízení komunikují. Poté dokáže vysílat tímto protokolem. Cílem mého projektu nebylo ovládání komerčních zařízení, proto má tato část drobné nedostatky. Doma jsem neměl potřebný hardware, takže tato část není ani otestovaná. Do projektu byla začleněna, protože jsem měl doma nevyužitý ovladač, který pracoval na této frekvenci. Přesnou strukturou jeho vysílání jsem se ale nezabýval.

### 1.1.3 Správa uložených zařízení

Zařízení se ukládají přímo do flash paměti ESP32 o velikosti 4 MB. Soubor je nazván „zarizeni.csv“.

Kvůli potřebě ovládání komerčních zařízení jsem zvolil následující schéma.

Název	kod_ovladac	kod_zarizeni
Zásuvka z projektu	x	12345
Komerční zásuvka	1354	1364

Do sloupce „název“ se uloží název, který si zadá sám uživatel v uživatelském rozhraní. Do sloupce „kod\_ovladac“ se v případě zařízení vyrobeného v rámci projektu uloží „x“, protože k ovládání stačí ID zařízení, které je v tomto případě ve sloupci „kod\_zarizeni“.

Při zadávání komerčního zařízení záleží na způsobu ovládání daného zařízení. Při tvorbě jsem vycházel z mého ovladače, který má

zvlášť kód pro vypnutí a zapnutí. Jiná zařízení mají odlišná schémata.

Pro implementaci takového typu zařízení je za potřebí analyzovat, jaký způsob ovládání zařízení používá. K tomu slouží example kód knihovny, který jsem nahrál na jedno ze svých zařízení. Pro funkčnost ovládání je také potřeba prostudovat dokumentaci k této knihovně a dopsat správný typ ovládání do místa v kódu (Toto místo je označeno přímo v kódu).

## **1.2 Koncové zařízení**

Toto je hlavní částí mého projektu. Koncové zařízení provádí následující funkce:

- přijímá požadavek od centrály
- zapne/vypne spotřebič
- reguje na pohyb
- reaguje na vstup uživatele z tlačítka

## **2 Použitý hardware**

Pro „centrálu“ jsem si vybral platformu ESP32 a to hned z několika důvodů. Čip má integrovanou wifi, k dispozici je velké množství dokumentace, hardwaru s příklady a knihovnamí. Další výhodou je velká komunita, což může pomoci při řešení problémů. Čip lze

také integrovat do prostředí Arduino IDE, což umožňuje snadnější práci při tvorbě kódu.

Při výběru jsem zvažoval i platformu raspberry pi, ale odradila mě přítomnost operačního systému, který je zbytečný pro tak malý projekt a velké pořizovací náklady oproti ESP32.

Pro „koncová zařízení“ a „analýzátor kódu“ jsem zvolil jako vývojovou platformu Arduino uno. Po odladění kódu a hardwaru jsem vyrobil prototyp, který již využíval Arduino Pro mini.

## 2.1 Kalkulace nákladů

Počet kusů	Název	Cena [Kč]
1	NodeMCU-32S ESP32	249
2	433 MHz vysílač a přijímač	79
2	Spirálová anténa 433 MHz	10
1	Mikrospínač	4
2	Relé modul s optickým oddělením	65
1	Rezistor 10k	1
1	Arduino Pro Mini	98
1	Arduino uno	599
1	PCB prototypová deska	18
1	PIR detektor pohybu	38

Celkové náklady na „centrálu“ jsou přibližně 290 Kč. Celkové náklady za prototyp „koncového zařízení“ jsou 265 Kč.

### **3 Testování**

p

### **4 Automatizace**

p

### **5 Porovnání s existujícími projekty**

p

### **6 Možnosti rozšíření**

p

## **Část II**

## **Uvedení do provozu**

p

### **7 Závěr**

p