

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ  
ÚSTAV AUTOMATIZACE A INFORMATIKY

FACULTY OF MECHANICAL ENGINEERING  
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

RADIOVÝ WATCHDOG  
RADIO WATCHDOG

DIPLOMOVÁ PRÁCE  
MASTER'S THESIS

AUTOR  
AUTHOR

BC. JAN MATUŠKA

VEDOUCÍ PRÁCE  
SUPERVISOR

ING. PETR KRČEK

BRNO 2008



ZDE VLOŽ LIST ZADÁNÍ



## **Abstrakt**

Práce se zabývá návrhem a implementací dohledového systému počítačové sítě občanského sdružení Starhill. Součástí návrhu je vývoj a realizace hardwarových modulů, které umožňují restartovat vzdálené síťové zařízení. Přenos povelu je realizován prostřednictvím libovolných dostupných radiostanic, jejichž povolovací podmínky tyto přenosy umožňují. Předpokládá se využití frekvenčního pásma 466 MHz. V práci je popsána koncepce a technická realizace. Navržený systém je integrovaný do stávajícího IS občanského sdružení.

## **Abstract**

This thesis deals with a design and implementation monitoring system of Starhill society computer network. Part of the design is development and realization hardware modules that provide restarting of remote network devices. Command transmission is realized through discretionary transceivers available, whose regulations can make such transmissions possible. A usage of 466 MHz frequency band is the main assumption. Both concept and implementation are described in the thesis. The designed system is integrated to the existing IS of the association.

## **Klíčová slova**

watchdog, dohledový systém, radiostanice, restart, vzdálené zařízení, bezdrátová síť

## **Keywords**

watchdog, monitoring system, transceiver, restart, remote device, wireless network



Chtěl bych touto formou poděkovat všem uživatelům sítě občanského sdružení Starhill za jejich trpělivost při ověřování korektní funkčnosti dohledového systému. Dále administrátorům komunitní počítačové sítě Hunet za jejich podnětné a konstruktivní připomínky. V neposlední řadě přítelkyni a rodině za morální podporu.





## Obsah

<b>1 Úvod</b>	<b>13</b>
<b>2 Bezdrátová síťová zařízení</b>	<b>15</b>
2.1 Kvalitativní segmenty . . . . .	15
2.1.1 Profesionální technika . . . . .	15
2.1.2 Neprofesionální technika . . . . .	15
2.2 Problémy Wi-Fi zařízení v reálném provozu . . . . .	16
2.3 Možnosti obnovení činnosti Wi-Fi zařízení . . . . .	17
2.3.1 Situace bez přenosu povelu k restartu vzdáleného prvku . . . . .	17
2.3.2 Situace s přenosem povelu k restartu vzdáleného prvku . . . . .	17
2.3.3 Přenos povelu pomocí GSM sítě . . . . .	17
2.3.4 Přenos povelu pomocí radiostanic . . . . .	18
<b>3 Koncepce navrženého systému</b>	<b>19</b>
3.1 Provozní prostředí . . . . .	19
3.2 Souhrn požadavků na realizovaný systém . . . . .	20
3.3 Požadavky na interface . . . . .	20
<b>4 PMR Radiostanice</b>	<b>21</b>
4.1 Povolovací podmínky . . . . .	21
4.2 Obvodové řešení radiostanice . . . . .	22
4.3 Pracovní frekvence a VCO . . . . .	22
4.4 Šumová brána . . . . .	24
4.5 Úsporný režim . . . . .	24
4.6 CTCSS . . . . .	25
4.7 Připojení radiostanice k interface . . . . .	25
4.7.1 Napájení . . . . .	25
4.7.2 Audio vstup a výstup . . . . .	26
4.7.3 Úroveň modulace . . . . .	26
4.8 Dosah . . . . .	27
<b>5 Interface</b>	<b>29</b>
5.1 Hardware . . . . .	29
5.2 DTMF . . . . .	31
5.2.1 Příjem DTMF . . . . .	31
5.2.2 Generování DTMF . . . . .	32
5.2.3 DTMF generátor s využitím PWM . . . . .	32
5.2.4 Parametry generátoru DTMF . . . . .	34
5.3 Komunikační protokol . . . . .	34
5.3.1 Autentizace . . . . .	34
5.3.2 Přenos požadavku radiově dostupné jednotce . . . . .	36
5.3.3 Přenos požadavku prostřednictvím opakováče . . . . .	38
5.4 Ověření dostupnosti . . . . .	39
5.5 Firmware . . . . .	39
5.5.1 Formát příkazů . . . . .	39
5.5.2 Informační příkazy . . . . .	39
5.5.3 Konfigurační příkazy . . . . .	40
5.5.4 Komunikační příkazy . . . . .	40

<b>6</b>	<b>Konfigurační aplikace</b>	<b>41</b>
6.1	Terminály pro Microsoft Windows	41
6.2	Konzolové terminály pro Linux	42
6.3	Aplikace RadioReset device configurator	42
6.3.1	Komunikační vazba aplikace s interface	43
6.3.2	Struktura tříd aplikace	44
6.3.3	Výpis stavových informací	45
6.3.4	Význam vstupně-výstupních polí formuláře	45
6.4	Funkce aplikace pro konfiguraci interface	45
6.4.1	Funkce aplikace pro komunikační režim interface	46
<b>7</b>	<b>Dohledový systém</b>	<b>47</b>
7.1	Struktura dohledového systému	47
7.2	Topologie sítě Starhill	48
7.3	Implementace datových struktur v DBMS PostgreSQL	49
7.3.1	Tabulka prvků	50
7.3.2	Tabulka emailových adres	50
7.3.3	Tabulka IP adres prvků	51
7.3.4	Tabulka parametrů pro interface	51
7.3.5	Tabulka textových popisů stavů	51
7.4	Typy prvků	52
7.4.1	Testovatelný prvek	52
7.4.2	Netestovatelný prvek	52
7.4.3	Resetovatelný prvek	52
7.4.4	Neresetovatelný prvek	53
7.4.5	Informovatelný a neinformovatelný prvek	53
7.5	Způsob detekce dostupnosti prvků	53
7.5.1	Nástroj ping a argumenty	54
7.6	Chybový kód prvku	54
7.7	Činnost dohledového systému	56
7.7.1	Periodické spouštění	56
7.7.2	Prohledávání stromu prvků v detekční části	57
7.7.3	Prohledávání stromu prvků v akční části	57
7.7.4	Rozhodovací mechanismus neresetovatelných prvků	57
7.7.5	Rozhodovací mechanismus resetovatelných prvků	59
7.7.6	Stavové informační zprávy	60
7.8	Implementace detekční a akční části	61
7.8.1	Struktura	61
7.8.2	Třída tradioreset	62
<b>8</b>	<b>Uživatelské rozhraní dohledového systému</b>	<b>63</b>
8.1	Vlastnosti WWW rozhraní	63
8.1.1	Centralizace zdrojového kódu aplikace	63
8.1.2	Nestavovost a interaktivita	63
8.1.3	WWW Server	64
8.2	Grafická reprezentace prvku	64
8.2.1	Chybový kód	65
8.2.2	Barva pozadí	65
8.3	Stromová struktura prvků	65

OBSAH	11
8.4 Administrace prvků . . . . .	66
8.5 Export do PDF . . . . .	68
<b>9 Závěr</b>	<b>69</b>
<b>10 Literatura</b>	<b>71</b>
<b>Přílohy</b>	<b>73</b>
<b>A Schéma zapojení a deska plošných spojů</b>	<b>75</b>
<b>B Seznam součástek</b>	<b>77</b>
<b>C Vybrané texty konfiguračního prostředí</b>	<b>79</b>
C.1 Nápověda . . . . .	79
C.2 Stavové zprávy . . . . .	79
<b>D Obsah přiloženého DVD</b>	<b>81</b>



## 1 Úvod

Poslední desetiletí je v České republice, podobně jako i ve světě, spojeno s masivním rozvojem bezdrátových síťových prvků. Jejich zásadní přínos spočívá v usnadnění realizace počítačových sítí i v těch místech, kde je využití metalických a optických spojů vyloučeno nebo je ekonomicky neúnosné. Jedná se především o síťové prvky založené na standartu IEEE 802.11 a využívající nelicencované sdílené pásmo 2.4 a 5 GHz. Provoz je na základě všeobecného oprávnění, vydaného Českým telekomunikačním úřadem, povolen libovolné fyzické osobě nebo podnikatelskému subjektu. Podobná situace je i v jiných zemích.

Vzhledem k uvedenému liberálnímu přístupu státních správ se stala tato neprofesionální technologie velmi populární. Odpadla nutná licenční řízení, poplatky na pronájem kmitočtového pásma a jiné nadbytečné byrokratické kroky. Přírozenou reakcí výrobců na rostoucí poptávku byla široká nabídka různých zařízení, antén a jiného příslušenství. Tento trend je pozorovatelný samozřejmě i v současnosti, kdy se bezdrátové technologie staly integrální součástí přenosných počítačů, osobních organizérů a vybraných mobilních telefonů. Silný konkurenční boj nutí výrobce k neustálým inovacím, snižování prodejní ceny a nákladů. Tento fakt je velmi povzbudivý pro zákazníky, ale negativně se odráží ve kvalitě a spolehlivosti výrobků. Na trh se tak dostávají zařízení, která plně nerespektují požadované specifikace, obsahují nedostatečně ověřený firmware nebo nekvalitní hardware.

Cílem této diplomové práce je navrhnout systém, který bude realizovat dohled nad bezdrátovou počítačovou sítí. Klíčovým požadavkem na výsledný systém je, aby umožňoval restart vzdáleného síťového prvku, který se z důvodu chyby hardware nebo firmware dostal do stavu nečinnosti. Hlavní motivací je tedy jisté zjednodušení a zefektivnění správy takové počítačové sítě, kde se méně spolehlivé síťové prvky vyskytují a není v silách administrátora z ekonomických nebo vlastnických důvodů tyto prvky nahradit takovými, jejichž provozní spolehlivost je vyšší.

Systém je primárně určen pro využití v síti občanského sdružení Starhill [22] a volně navazuje na základ správního systému popsaného v bakalářské práci [11]. Autor uvedené bakalářské práce a této diplomové práce je ve zmíněném sdružení angažován od jeho vzniku v roce 2001. Vzhledem k nekomerčnímu charakteru občanského sdružení je požadována snaha o minimalizaci pořizovacích a provozních nákladů celého systému.

Koncepce práce je rozdělena do dvou logických částí. První popisuje realizaci interface určeného pro přenos povelu ke vzdáleným síťovým prvkům a druhá část dohledový systém jejich dostupnosti. Číslování kapitol je spojitě, aby byla zajištěna homogenita práce jako celku. Důvodem uvedeného logického rozdělení je předpokládaná možnost využití navrženého interface lidmi z jiných komunitních počítačových sítí. Dohledový systém je s existující infrastrukturou sdružení Starhill svázán výrazněji a pro jiné sítě může být aplikovatelný jen s velkými obtížemi.



## 2 Bezdrátová síťová zařízení

### 2.1 Kvalitativní segmenty

V oblasti mikrovlnných bezdrátových technologií existuje široká řada technických prostředků, kterými lze tyto sítě realizovat. Platí zde, stejně jako v každé jiné oblasti, kde panuje bouřlivý rozvoj vlastní technologie, že existují zcela odlišné kvalitativní segmenty výrobků s velmi rozdílnými technickými a provozními parametry.

#### 2.1.1 Profesionální technika

Tyto technické prostředky mají výrazně delší historii než technika neprofesionální. Jsou určeny pro budování datových sítí, kde jakýkoliv výpadek může znamenat výraznou finanční ztrátu pro organizaci, která tyto spoje provozuje. Cílem výrobců, kteří mají zájem uspět v konkurenčním prostředí, je nabídnout velmi kvalitní a spolehlivé zařízení, které bude po celou dobu své životnosti plnit svoji funkci s téměř 100% spolehlivostí. Samozřejmostí bývá podpora a servis.

Cena je v tomto případě méně významný faktor, protože se ve většině případů jedná o investiční náklady s garancí návratnosti. Soukromé osoby si tuto technologii obvykle nepořizují. Finanční náročnost jednoho spoje se pohybuje v řádu statisíců Kč. Je samozřejmě závislá na přenosové rychlosti, dosažitelné vzdálenosti a využívaném frekvenčním pásmu. Radiové jednotky různých výrobců nejsou vzájemně kompatibilní a využívají firemní know-how.

V současné době tyto mikrovlnné spoje ve velké míře využívají operátoři sítí mobilních telefonů, kteří tímto způsobem řeší propojení svých základnových stanic. Dále se jedná o telekomunikační firmy, kteří formou outsourcingu zabezpečují svým zákazníkům vzájemné propojení jejich poboček. Levnější varianty spojů pracujících v nelicencovaném pásmu 10 GHz lze nalézt u lokálních poskytovatelů připojení k Internetu. Síť občanského sdružení Starhill takový mikrovlnný spoj využívá pro připojení k Internetu. Jeho přenosová kapacita je 40 Mbit/s na vzdálenost 20 km.

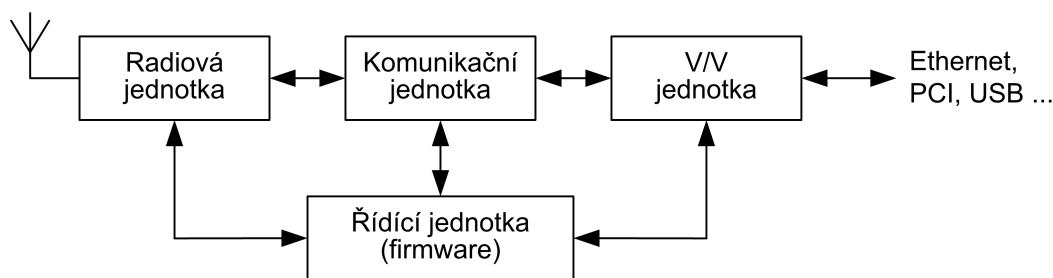
#### 2.1.2 Neprofesionální technika

S tímto segmentem bezdrátových technologií jsou pevně spjata všechna zařízení založená na standartu IEEE 802.11 a rozšiřujících 802.11a/b/g, souhrnně označované jako Wi-Fi. Od počátku vývoje těchto zařízení se předpokládalo, že budou provozována výhradně v domácnosti nebo kanceláři a to zpravidla k bezdrátovému připojení notebooků k domácí nebo firemní síti.

Ve státech s dlouhodobě rozvinutým a konkurenčním trhem na poli telekomunikačních operátorů zůstala tato technologie opravdu tam, kde byla původně určena. V zemích střední a východní Evropy byla situace odlišná v tom, že v době obrovské poptávky po připojení k Internetu nebyly polostátní telekomunikační firmy schopny pružně reagovat na vzniklou situaci a nabídnout cenově dostupné připojení pro běžné nefiremní uživatele. Reakcí na tuto situaci vzniklo velké množství veřejných bezdrátových sítí založených na technologii, která neměla původně k tomuto účelu sloužit. Je tedy zřejmé, že od těchto zařízení nelze čekat spolehlivost takovou jako u profesionální techniky. Tento nedostatek je však vyvážen o několik řádů nižší cenou a reálnou dostupností i pro běžné uživatele, kteří jsou nuceni občasně problémy tolerovat.

Existuje však typ laických uživatelů bezdrátových sítí, kteří za cenu průměrného Wi-Fi zařízení požadují absolutní spolehlivost a odmítají přijmout argumenty o původním zacílení této technologie. Tito uživatelé jsou poté velmi problémoví zákazníci komerčních sítí nebo členy občanských sdružení.

## 2.2 Problémy Wi-Fi zařízení v reálném provozu



Obr. 2.1 Blokové schéma typického Wi-Fi zařízení.

- *Radiová jednotka:* zajišťuje bezdrátovou komunikaci s ostatními Wi-Fi zařízeními v rádiovém dosahu. Její parametry jsou specifikovány všeobecným oprávněním Českého telekomunikačního úřadu. Každé zařízení musí projít před uvedením do provozu povinnou certifikací, a proto lze předpokládat, že parametry radiové jednotky budou v akceptovatelných mezích u všech radiových jednotek. Přesto však existují zařízení, které svou nízkou odolností proti interferencím nemohou být provozovány v silně zarušeném prostředí – přijímač se zahltí a znemožňuje komunikaci.
- *Komunikační jednotka:* představuje základní logickou část Wi-Fi zařízení. Implementuje v sobě přenosové protokoly, řízení provozu a zabezpečení. V mnoha případech je komunikační jednotka pevně spjata s radiovou jednotkou. Komunikační jednotka by měla striktně respektovat stanovené normy, ale ve skutečnosti tomu tak není. Výrobci ve snaze ušetřit náklady standarty zjednodušují a tak není neobvyklé, že spolu 2 zařízení označené „Wi-Fi 802.11b“ nekomunikují a nebo komunikují s mnoha omezeními jako např. nemožnost použití zabezpečení, nemožnost dosažení plné rychlosti, výpadky při přenosu apod. U některých jednotek dochází k maximálnímu vytížení již při několika připojených klientech i přesto, že výrobce garantuje desítky.
- *Vstupně výstupní jednotka:* zajišťuje propojení komunikační a řídicí jednotky na metalickou počítačovou síť. Pokud je Wi-Fi zařízení ve formě přídatné karty, pak přes PCI nebo USB sběrnici přímo do PC. Vstupně výstupní jednotka spolehlivě pracuje i v nejlevnějších zařízeních. Výrobci využívají ověřených řadičů, které lze nalézt i v mnoha jiných počítačových perifériích.
- *Řídící jednotka:* realizuje provozní funkce celého Wi-Fi zařízení. Umožňuje externí konfiguraci parametrů pomocí integrovaného webového rozhraní, SNMP, Telnetu nebo SSH. U jednoduchých zařízení je řídicí jednotka tvořena jednočipovým mikrokontrolérem s jednoúčelovým firmware. Složitější řídicí jednotky, které umožňují pokročilé funkce jako IP směrování, firewall, NAT, řízení přenosové rychlosti apod., jsou tvořeny výkonným mikrokontrolérem s desítkami MB RAM a flash paměti. V těchto jednotkách je provozován upravený operační systém Linux. Řídící jednotka je nejčastějším zdrojem problémů v reálném provozu zařízení. Trh výrobce tlačí k neustálým a velmi rychlým inovacím, ke snižování cen a z tohoto důvodu není nově vyvinutý firmware řádně otestovaný. Není výjimkou, že se použitelný a stabilní firmware objeví až po několika měsících od uvedení zařízení na trh. Chyby ve firmware řídicí jednotky se projevují tak, že Wi-Fi zařízení přestane plnit svoji funkci celé a nebo jen jeho část. Důvody jsou zpravidla takové, že se v zařízení vyskytnou takové provozní stavy, které firmware nešetruje a dojde k jeho pádu. Jediným řešením je dané zařízení odpojit od elektrické energie a poté znovu připojit – provést restart.



Na základě uvedených problémů by se mohlo být zřejmé, že každý Wi-Fi produkt je špatný. Existuje však řada velmi dobrých a spolehlivých zařízení, jejichž cena není paradoxně automaticky nejvyšší. V tomto případě záleží na referencích a provozních zkušenostech než na technické specifikaci od výrobce. V mnoha sítích, zpravidla nekomerčních a komunitních, se však vyskytuje velké množství horších prvků, které si zakoupili sami uživatelé a jejich výměna je tedy nereálná. Aktivní bezdrátová síť je velmi dynamický systém, který je těmito méně kvalitními zařízeními silně ovlivňován a to převážně sníženou propustností, resp. stabilitou spojení mezi přístupovým bodem a ostatními klienty. V některých případech může dojít i k negativnímu ovlivnění komunikační a řídicí jednotky přístupového bodu a k následnému výpadku.

## 2.3 Možnosti obnovení činnosti Wi-Fi zařízení

### 2.3.1 Situace bez přenosu povelu k restartu vzdáleného prvku

Pokud nějaký síťový prvek přestane plnit svoji funkci, administrátor se o tomto stavu dozví od uživatelů, kterých se výpadek týká nebo ze systému kontroly dostupnosti uzlů v síti. V obou případech je nutný jeho osobní zásah na místě, kde je problémový prvek umístěn. Jelikož se bezdrátové prvky umísťují na místa, která mají nejlepší rozhled do okolí, tedy střechy, stožáry, komíny, je přístup k samotnému prvku komplikovaný. Administrátor musí kontaktovat majitele nemovitosti, požádat o přístup, zapůjčit klíče a v případě, že nejsou špatné povětrnostní podmínky, tak provést odpojení a připojení prvku k elektrické energii. Je zřejmé, že tato činnost je časově velmi náročná a v některých podmínkách, zpravidla v zimě, obtížná.

V sítích občanských sdružení je nezanedbatelná i finanční náročnost zásahu. Jedná se především o telefonní poplatky, pohonné hmoty apod. Ekonomická situace občanských sdružení bývá vzhledem k nekomerčnímu pojetí sítě poměrně napjatá a každé takto vynaložené náklady pak chybí jinde. Je nutné podotknout, že většina administrátorů nekomerčních sítí se této činnosti věnuje ve volném čase při studiu nebo zaměstnání. Prostor pro výrazné odměny v rozpočtu sdružení obvykle není.

### 2.3.2 Situace s přenosem povelu k restartu vzdáleného prvku

Administrátor je informován o výpadku podobným způsobem jako v předchozí kapitole 2.3.1. Rozdíl je v tom, že danou lokalitu nemusí navštívit osobně, ale může využít technických prostředků k tomu, aby nekorektně pracující prvek vzdáleně restartoval. Celý proces je ve srovnání s osobním zásahem velmi rychlý a výpadek datové linky relativně krátký – záleží na rychlosti reakce administrátora.

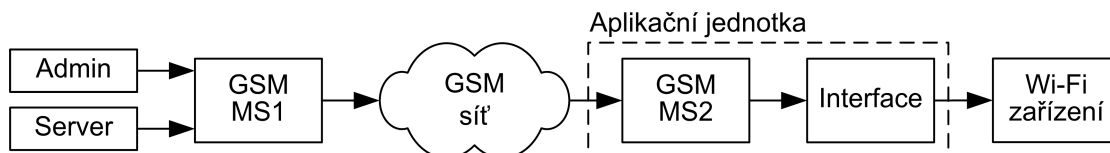
Povel může být přenášen těmito, snadno realizovatelnými, způsoby:

- GSM síť.
- Radiostanice nebo radiomodul.

### 2.3.3 Přenos povelu pomocí GSM sítě

Princip využití GSM sítě spočívá v existenci dvou mobilních stanic MS1 a MS2. MS1 vlastní administrátor ve formě běžného mobilního telefonu nebo je připojena k nadřazenému systému pro kontrolu dostupnosti. MS2 je připojena k interface, který sestává z jednoduchého obvodu řízeného MCU (mikrokontrolér). Tento obvod periodicky prověřuje příchozí SMS zprávy v MS2. Pokud detekuje příchozí zprávu od MS1 ve stanoveném tvaru, provede např. pomocí relé odpojení Wi-Fi zařízení (i jiného spotřebiče) od elektrické energie na stanovenou dobu a následné připojení zpět.

Využití tohoto způsobu přenosu povelu je v síti sdružení Starhill realizováno od roku 2003. Provozní zkušenost s mobilními telefony Siemens C35-55 však ukazuje, že spolehlivost těchto mobilních telefonů je v některých případech nižší než spolehlivost chráněného Wi-Fi zařízení. Pro zajištění korektní funkce je tedy nutné použít výrazně nákladnější průmyslový GSM modul. Celková cena za interface, GSM modul a nezbytné montážní prvky je téměř shodná s cenou samotného Wi-Fi zařízení v malém uzlu sítě.



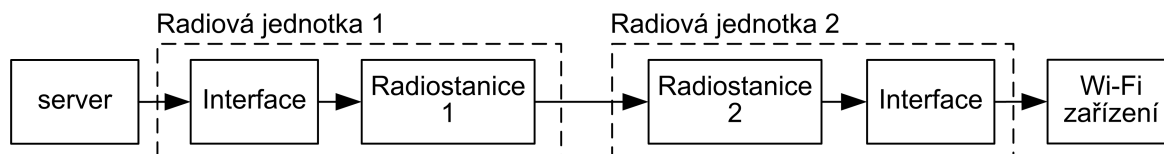
Obr. 2.2 Přenos povelu prostřednictvím GSM sítě.

- Výhody: pokrytí na území ČR, snadná realizace.
- Nevýhody: nespolehlivost MS, cena GSM modulu, tarifní provozní náklady.

V síti Starhill je tohoto způsobu přenosu povelu využito jen v hlavním uzlu sítě Jiráskova.

### 2.3.4 Přenos povelu pomocí radiostanic

Tento způsob přenosu povelu využívá ke vzdálené komunikaci audio signál vysílaný radiostanicí 1 a přijímaný radiostanicí 2. Elektronické obvody v nezbytných interface slouží k řízení relací při komunikaci radiostanic a kódování/dekódování analogových signálů. Pokyn k restartu vydá nadřazený systém, protože se nepředpokládá, že administrátor u sebe nosí radiostanici s interface a vyskytuje se v radiovému dosahu protější radiostanice.

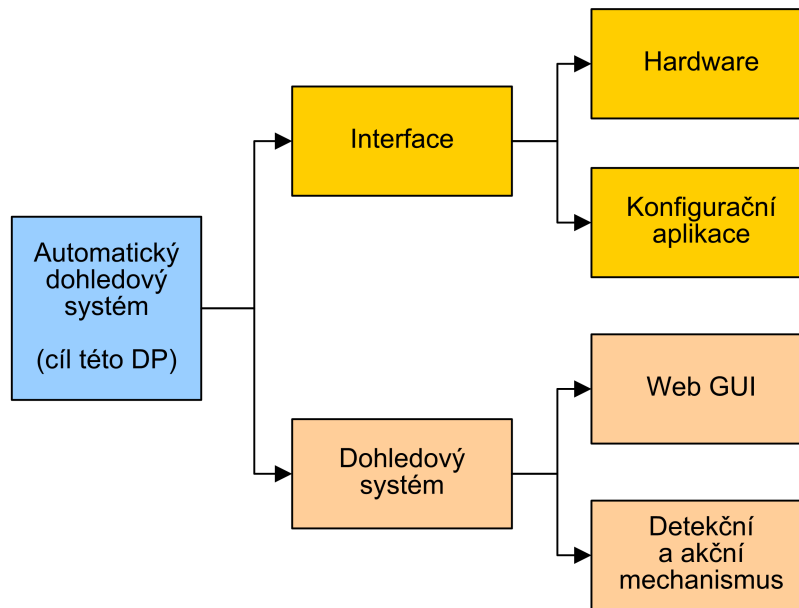


Obr. 2.3 Přenos povelu prostřednictvím radiostanic.

- Výhody: nulové provozní a nižší pořizovací náklady.
- Nevýhody: lokální dosah radiostanic, nutnost existence nadřazeného systému.

Realizace tohoto způsobu přenosu je záměrem této diplomové práce. Veškeré další podrobnosti jsou uvedeny v následujících kapitolách.

### 3 Konceptce navrženého systému



Obr. 3.1 Bloková struktura navrženého systému

Význam jednotlivých bloků uvedené struktury:

- *Interface*: hardware zajišťující komunikaci mezi dohledovým systémem a vzdálenou lokalitou. K přenosu povelu využívá FM radiostanici v pásmu 446 MHz.
- *Hardware*: návrh interface je uveden v kap. 5. Vlastnosti vhodných radiostanic jsou zmíněny v kap. 4.
- *Konfigurační aplikace*: software k nastavení provozních parametrů interface. Popis je v kap. 6 a vlastní software v příloze D.
- *Dohledový systém*: software realizující dohled nad dostupností prvků v síti.
- *Web GUI*: administrační rozhraní dohledového systému. Řešení v kap. 8.
- *Detekční a akční mechanismus*: výkonná část dohledového systému. Realizuje test dostupnosti prvků a případné akční zásahy s využitím interface. Řešení v kap. 7.

#### 3.1 Provozní prostředí

- Síť občanského sdružení Starhill.
- Administrátorská práva k síťovým prvkům.
- Přístup do síťových rozvaděčů ve všech lokalitách.
- Server s OS Linux v centrálním uzlu sítě M. Distribuce Debian 4.0 Etch.

Obecná topologie sítě včetně alfanumerického označení uzlů je uvedena v kapitole 7.2. Verze zakreslená na mapovém podkladu je umístěna v příloze D.

### 3.2 Souhrn požadavků na realizovaný systém

- Realizovat systém vzdáleného restartu síťových prvků prostřednictvím radiostanic dle konceptuálního návrhu uvedeného v kapitole 2.3.4.
- Umístit hlavní radiovou jednotku v centrálním uzlu M a testovací jednotku v uzlu D. Reálné nasazení předpokládá nasazení 7 ks jednotek. Budou doplněny postupně dle potřeby.
- Z bezpečnostních důvodů instalovat jednotky do oddělených elektroinstalačních krabic a pro realizaci odpojení napájení síťového prvku využívat výhradně nízkého napětí za příslušným síťovým adaptérem.
- Dohledový systém realizovat na serveru v uzlu M. Předpokládá se možnost provozu i bez radiových jednotek.

### 3.3 Požadavky na interface

- Použití elektronických součástek běžně dostupných v maloobchodní síti.
- Minimalizace pořizovacích nákladů bez negativního vlivu na funkcionalitu.
- Zhotovitelnost v amatérských podmínkách bez speciálních postupů.
- Možnost snadného připojení širokého spektra radiostanic.
- Minimálně 3 výkonové rozpínací výstupy. Proudové zatížení min. 5 A.
- Ke vzájemné komunikaci mezi interface využít DTMF (tónová volba).

## 4 PMR Radiostanice

PMR<sup>1</sup> radiostanice patří mezi typ radiokomunikační techniky, která je v současné mezi spotřebiteli velmi populární. Lze ji zakoupit jak ve specializovaných prodejnách, tak i v běžné obchodní síti, obchodních domech apod. Výhodou je relativně nízká pořizovací cena, různé formy provedení a malá hmotnost. Stejně jako u jiných výrobků spotřební elektroniky i zde platí, že vyšší cena automaticky neznamená vyšší kvalitu výrobku. Vhodnou radiostanicí je nutné vybírat na základně referencí nebo vlastní zkušenosti.

I přes značnou různorodost značek, výrobců a jednotlivých modelů jsou všechny tyto radiostanice koncepčně velmi podobné a lze tak provést zobecnění jejich parametrů. Jen velmi málo výrobců si dovolí, zvláště u výrobků této cenové kategorie, navrhnout nějaké nestandardní nebo inovativní řešení. Spíše dochází k tomu, že mnozí výrobci hledají cesty, jak minimalizovat náklady již ověřené koncepce bez snížení kvalitativních parametrů radiostanice. Smyslem této kapitoly je poskytnout základní pohled na vybrané parametry radiostanic a to takovým způsobem, aby byly zřejmé navazující souvislosti vzhledem k hardwarovému interface. Diagnostika případných problémů je poté snazší.

Radiostanice je v kontextu této diplomové práce chápána jako dále nedělitelná entita, která je využita takovým způsobem, že není v žádném případě nutné zasahovat do jejich elektronických obvodů.

### 4.1 Povolovací podmínky

Provoz PMR radiostanic v ČR podléhá všeobecnému oprávnění č. VO-R/3/07.2005-16. Český telekomunikační úřad<sup>2</sup> vydal toto oprávnění na základě mezinárodní dohody členů CEPT<sup>3</sup> a umožnil tak využívat tyto radiostanice jako v mnoha jiných zemích Evropy i mimo EU.

Základní pravidla pro provoz jsou následující:

- *Využití bez individuálního povolení.* Radiostanicí může využívat naprosto každý bez omezení. Komerční subjekty nejsou vyloučeny.
- *Určeno pro hlasovou komunikaci.* Tato provozní podmínka je na první pohled v rozporu se smyslem využití radiostanic v rámci této diplomové práce. Přenos povelu je možné chápat i jako přenos dat, tedy nevhodné informace. Z tohoto důvodu byl na ČTÚ vznesen dotaz, který požadoval objasnění pojmu „hlasová komunikace.“ Dle vyjádření pracovníků ČTÚ je možné prostřednictvím těchto radiostanic přenášet jakýkoliv frekvenčně modulovaný signál, který spadá do hovorového spektra. Toto omezení má primárně vyloučit různé formy digitálních modulací, jako je např. QPSK<sup>4</sup> apod. Pro přenos povelu nemohou být tyto modulace využity již na základě vlastností samotné radiostanice – nedisponuje jimi.
- *Radiostanice nesmí být elektricky a mechanicky měněna.* Zakázány jsou jakékoliv úpravy elektronických obvodů a připojování externích antén. Důsledkem neodborného zásahu bez příslušného měřicího vybavení může být výrazné zhoršení parametrů vysílače a následně možné rušení kritických radiových služeb – záchranné složky, řízení letového provozu apod.
- *Technické parametry.* Tyto parametry jsou uvedeny v následujících kapitolách s příslušnými komentáři.

<sup>1</sup>Personal Mobile Radio. Analogové simplexní radiostanice pracující v pásmu 446 MHz.

<sup>2</sup>ČTÚ je hlavním kontrolním orgánem ve využívání radiového spektra v ČR.

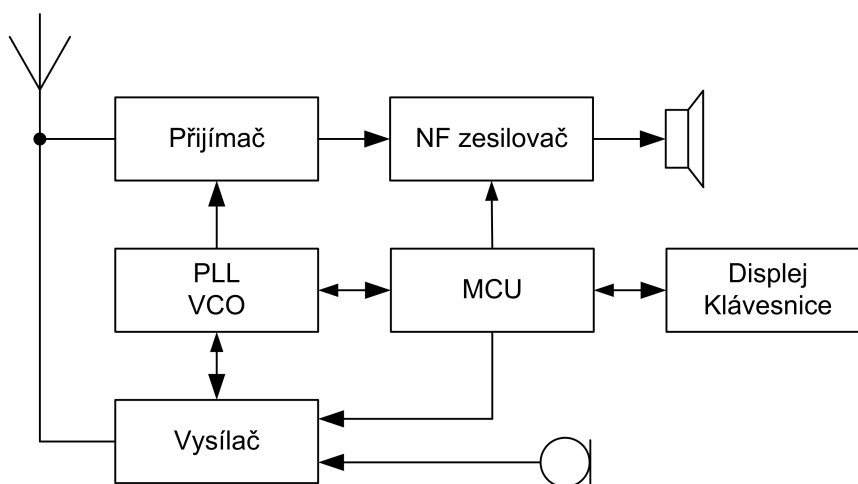
<sup>3</sup>Konference evropských správ pošt a telekomunikací. Zajišťuje mezinárodní koordinaci v rámci Evropy.

<sup>4</sup>Čtyřstavová fázová modulace.

## 4.2 Obvodové řešení radiostanice

Základními funkčními bloky jsou:

- *PLL/VCO*: zdroj frekvenčně stabilního harmonického vf signálu pro přijímač a modulovaného pro vysílač. Základní princip funkce je popsán v kapitole 4.3
- *Přijímač*: je řešen jako superheterodyn s dvojitým směřováním. Vstup přijímače je osazen pásmovým keramickým filtrem a následovaný směšovačem. Směřováním signálu přijímaného a generovaného z VCO vznikne rozdílový a součtový mezifrekvenční produkt, který je dále zpracováván. Kmitočet VCO musí být o požadovanou hodnotu rozdílu vyšší nebo nižší než je přijímaný kmitočet. Obvyklá hodnota mezifrekvence je 21,7 nebo 10,7 MHz. Následuje mezifrekvenční filtr, druhý směšovač, filtr a demodulátor. Obvody druhé mezifrekvence a demodulátoru jsou řešeny obvykle formou speciálního integrovaného obvodu.
- *Vysílač*: vstupní signál z mikrofonu je přiveden na předzesilovač s regulovatelným ziskem. Následně pak do VCO, kde je jeho prostřednictvím ovlivňován výstupní kmitočet – princip FM modulace. Takto získaný modulovaný signál je dále v koncovém stupni zesílen na 150–250 mW a to v závislosti na zisku použité antény. Maximální vyzářený výkon je povolenými podmínkami omezen na 500 mW EIRP<sup>5</sup>. K eliminaci harmonických signálů je mezi koncovým stupněm vysílače a anténou zařazena dolní propust ve formě dvojitého  $\pi$ -článku.
- *MCU*: realizuje ovládaní funkčních bloků radiostanice, zobrazení stavových informací na displeji a reakci na pokyny uživatele.



Obr. 4.1 Blokové schéma PMR radiostanice

## 4.3 Pracovní frekvence a VCO

Pro provoz radiostanic je vyhrazeno frekvenční pásmo 446,0–446,1 MHz. Tento úsek je rozdělen na 8 kanálů. Radiostanice obvykle nezobrazuje pracovní kmitočet, ale pouze číslo zvoleného kanálu. Z tab. 4.1 je patrné, že jejich vzájemný rozestup je 12,5 kHz. Jedná se o standardní běžně užívanou hodnotu, která umožňuje dostatečnou eliminaci přeslechů ze sousedních kanálů.

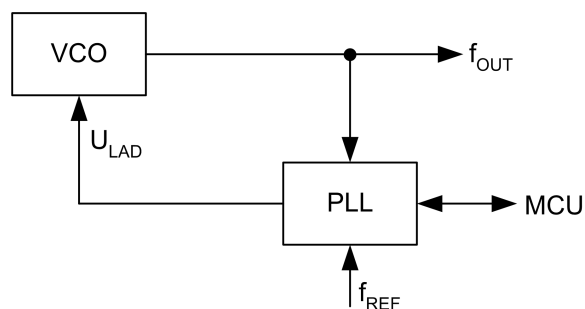
K přenosu modulačního signálu je využita úzkopásmová frekvenční modulace s modulačním indexem  $m \leq 1$ . Maximální dovolený kmitočtový zdvih pro tyto komunikační účely je stanoven na 3 kHz. Modulační signál je přenášen v rozsahu cca 60–2800 Hz a jeho maximální strop bývá obvykle omezen dolní propustí v mikrofonním předzesilovači. Pro přenos běžného hlasu jsou tyto parametry dostačující.

<sup>5</sup>Ekvivalentní izotropně vyzářený výkon. Platí vztah  $P_{EIRP} = P_{TX} 10^{0,1G}$  [W; W; dB]. Přejvato z [9].

Tab. 4.1 Rozdělení kanálů

Kanál	Střední kmitočet [MHz]
1	446,00625
2	446,01875
3	446,03125
4	446,04375
5	446,05625
6	446,06875
7	446,08125
8	446,09375

VCO je jednou z nejdůležitějších součástí radiostanice. Jedná se o napětově řízený zdroj harmonického vysokofrekvenčního signálu, který je využíván vysílačem i přijímačem. Bývá realizován formou rezonančního LC obvodu s varikapem, který umožňuje v závislosti na přivedeném stejnosměrném napětí měnit svoji kapacitu a tím i rezonanční frekvenci. Varikap nenahrazuje celou kapacitní složku LC obvodu. Je zapojen v sério-parallelním obvodu s fixními kapacitami kondenzátorů. Přeladitelnost oscilátoru je omezena a jeho rozsah závisí zpravidla na požadovaném kmitočtovém rozsahu a použité první mezifrekvenci přijímače. U PMR radiostanic bývá obvykle 25–30 MHz. Podobným způsobem je implementován FM modulátor, ovšem sério-parallelní fixní kapacity jsou zvoleny tak, aby změna frekvence v závislosti na přivedeném napětí byla v řádu desítek kHz.



Obr. 4.2 Zapojení VCO v kombinaci s PLL

Samotný VCO je teplotně a časově nestálý. K udržení přesné frekvence je nutná neustálá korekce ladícího napětí. Z tohoto důvodu obsahuje každá radiostanice regulátor (PLL), který zajišťuje stabilní výstupní frekvenci s možností číslicového nastavení její hodnoty.

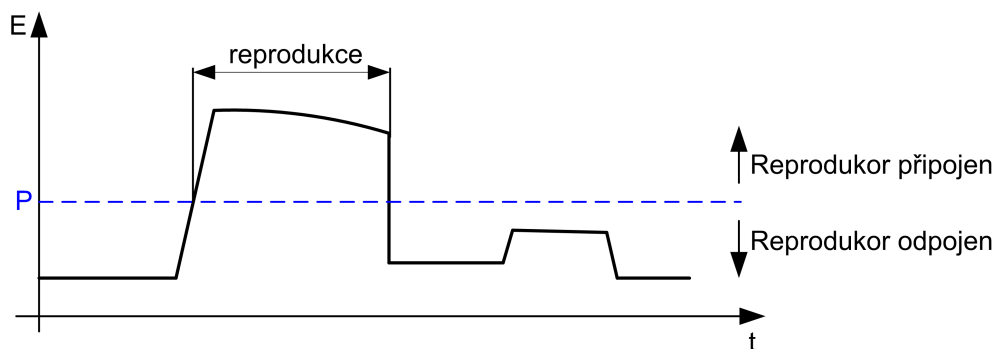
Princip činnosti PLL spočívá v porovnávání fázi výstupního signálu  $f_{OUT}$  s referenčním  $f_{REF}$ . Na základě zjištěného fázového rozdílu je vytvořeno ladící napětí  $U_{LAD}$ , které ovlivňuje frekvenci VCO. Signály  $f_{OUT}$  a  $f_{REF}$  jsou před fázovým detektorem přivedeny na interní konfigurovatelné děličky. Na základě nastavených dělicích poměrů lze zvolit požadovanou frekvenci VCO. Referenční signál  $f_{REF}$  je získán pomocí stabilního krystalového oscilátoru s několikanásobně nižším kmitočtem než je pracovní kmitočet VCO a bývá zpravidla součástí integrovaného obvodu, který realizuje PLL – vyjma externího krystalu.

Z uvedeného popisu činnosti PLL je zřejmé, že přesnost VCO je závislá na přesnosti referenčního oscilátoru. V případě, kdy by krystalový oscilátor vykazoval nějakou frekvenční odchylku od své jmenovité hodnoty, je možné ji ve velmi malém rozsahu eliminovat proměnným kondenzátorem. Bývá volně přístupný na DPS radiostanice. Vliv na kmitočet VCO je obvykle do 10 kHz. Zde je nutno upozornit, že žádná nově zakoupená radiostanice by neměla mít odchylku VCO

větší jak cca 1 kHz – do této hodnoty nelze pozorovat výraznější zhoršení kvality přeneseného modulačního signálu. Pokud je odchylka VCO větší, je vhodnějším řešením reklamacce.

#### 4.4 Šumová brána

Výhodou frekvenční modulace je poměrně kvalitní přenos modulačního signálu a nepříliš velká citlivost na průmyslové rušení. Nevýhodou je naopak velmi silný šum v případě, kdy přijímač v radiostanici nemá na vstupu žádný nebo dostatečně silný nosný signál. Aby obsluha nebyla tímto šumem obtěžována, obsahuje každá radiostanice funkci šumové brány. Princip její činnosti je znázorněn na obr. 4.3. Jedná se pouze o ilustrační příklad. Charakter intenzity vf signálu nemá v reálném prostředí tvar přímky, ale je zatížen šumem.



Obr. 4.3 Princip šumové brány

Demodulátor v přijímači radiostanice sleduje intenzitu vstupního vf signálu  $E$  a porovnává ji s nastavenou prahovou hodnotou  $P$ . Pokud se signál dostane nad tuto hodnotu, demodulátor připojí výstupní demodulovaný nf signál k nízkofrekvenčnímu zesilovači. V opačném případě zůstane odpojen. Výrobci obvykle nastavují šumovou bránu tak, aby byl reprodukován pouze srozumitelný hlasový přednes za obvyklých provozních podmínek. Z důvodu, že ve většině radiostanic nelze prahovou hodnotu nastavit bez zásahu do elektronických obvodů, je nutné zajistit, aby vzájemně komunikující radiostanice měly za každých okolností dostatečně silný signál.

#### 4.5 Úsporný režim

Radiostanice je z principu svého užití koncipována jako přenosné zařízení s bateriovým napájením. Proto je zcela přirozenou snahou výrobců maximalizovat provozní dobu i za cenu určitých kompromisů. Jedná se totiž o kvalitativní parametr, který je potenciálními zákazníky snadno porovnatelný s konkurencí. Provozní doba radiostanice závisí především na kapacitě použitých akumulátorů a na způsobu řízení odběru v jednotlivých částech radiostanice. Je zřejmé, že v režimu příjmu je zcela zbytečné napájet mikrofonní předzesilovač nebo v režimu vysílání kompletní obvody přijímače.

Existuje však ještě jedna možnost, výrobci hodně využívanou, jak průměrnou spotřebu radiostanice velmi výrazně snížit. Je založena na předpokladu, že se radiostanice většinu svého provozního času nachází v režimu příjmu s uzavřenou šumovou bránou. V tomto klidovém stavu MCU udržuje přijímač ve vypnutém stavu s odpojeným napájením. Spotřeba se tak pohybuje do 1 mA. Aby byla zachována komunikační schopnost radiostanice, MCU periodicky spouští přijímač na několik ms. Spotřeba vzrůstá na desítky mA. Pokud se na vstupu přijímače v okamžik jeho spuštění nachází signál, který otevře šumovou bránu, MCU okamžitě zareaguje dočasným odpojením tohoto úsporného režimu po celou dobu vysílání protistanice. Pro zajištění plynulé komunikace je úsporný režim neaktivní dalších cca 15 sekund po ukončení vysílání protistanice.



Nevýhodou uvedeného způsobu snižování spotřeby je fakt, že radiostanice je mezi dvěma testy přítomnosti signálu pro protistanici absolutně nedostupná. Výrobci proto volí kompromisní interval testů v rozsahu 1–2 s. Tato hodnota pro většinu případů postačuje. Využití radiostanice v rámci této DP nepočítá s bateriovým napájením, ale u většiny radiostanic nelze tento úsporný režim trvale vyřadit z činnosti. Vhodnou konfigurací interface je tedy nutné zajistit, aby k přenosu povelu došlo až v momentě, kdy bude přijímací radiostanice skutečně modulační signál k protějším interface přenášet. V opačném případě by mohlo docházet k ořezávání začátku povelu na přijímací straně a následnému selhávání komunikace.

## 4.6 CTCSS

S funkcí CTCSS<sup>6</sup> se lze setkat u lepších PMR radiostanic. Rozšiřuje funkci šumové brány o možnost selektivního výběru signálu. Princip CTCSS je založen na faktu, že vysílací radiostanice do modulačního signálu přidává harmonický subtón v intervalu 67–255 Hz. Pokud je na přijímací straně detekován dostatečně silný vf signál a současně je v demodulovaném signálu zjištěna přítomnost požadovaného subtónu, je připojen reproduktor. Ve všech ostatních případech je odpojen. Obvykle je k dispozici 38 volitelných subtónů.

Výhodou CTCSS je, že přijímací stanice ignoruje všechny ostatní vysílače, které nevysílají požadovaný subtón. Je zde možné použít analogii k VLAN v počítačových sítích, kdy jedno přenosové médium sdílí logicky nezávislé sítě.

Velmi často lze v reklamních materiálech k radiostanicím nalézt tvrzení, že umožňuje provoz na 308 kanálech apod. Jedná se však o mystifikaci spotřebitele. Dle kapitoly 4.3 je k využití přiděleno 8 nezávislých kanálů – kmitočtů. Na každém lze vytvořit 38 logických skupin, ovšem vysílat může na daném kanále vždy jen jedna radiostanice bez ohledu na nastavený subtón.

## 4.7 Připojení radiostanice k interface

### 4.7.1 Napájení

Napájení přenosných PMR radiostanic je řešeno výhradně pomocí tužkových a mikrotužkových NiCd/NiMH akumulátorových článků. Lze sice najít některé výjimky využívající interní speciální akumulátor, ale většina výrobců využití tužkových článků preferuje. Důvodem je skutečnost, že provozní doba článků může při intenzivním využití radiostanice klesnout na pouhých několik hodin a po této době by byla radiostanice nepoužitelná. Uživatel však může vlastnit jinou sadu nabitých akumulátorových článků, které je možné vyměnit a radiostanici dále využívat.

Z kontextu této diplomové práce je poměrně lhostejné, jakou formou je bateriové napájení realizováno. Důležitý je pouze fakt, že výrobci využívají k napájení typicky 4 kusy článků a většina radiostanic neobsahuje konektor pro externí připojení napájecího napětí. Je tedy nutné externí napájení přivést přímo na kontakty pro baterie a zajistit jejich řádné, trvalé a nedestruktivní uchycení. Tento způsob sice neodporuje povolovacím podmínkám, ale není také příliš elegantní. Je však nejjednodušší a provozně spolehlivý.

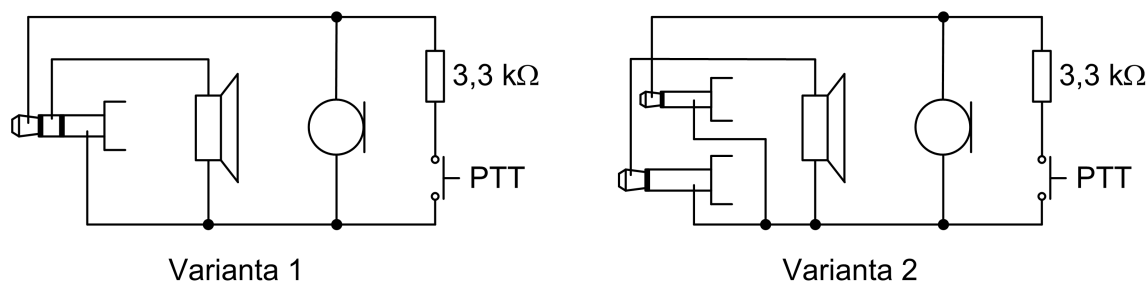
Velikost napájecího napětí je odvozena od počtu použitých článků a jejich typu. Jmenovité napětí akumulátoru je 1,2 V, suchého článku 1,5 V. Rozmezí celkového napětí bateriových sad se může pohybovat v intervalu 4,8–6 V. Většina radiostanic obsahuje interní stabilizátor na provozní napětí 3,3 V pro řídicí část, přijímač a VFO. Koncový stupeň vysílače je připojen mimo stabilizátor. Kompromisní velikost externího napájecího napětí byla zvolena na 5 V. Spotřeba radiostanice v režimu vysílání je při tomto napětí typicky 250–350 mA.

<sup>6</sup>Continuous Tone-Coded Squelch System. Využíváno převážně v profesionálních radiových sítích.

### 4.7.2 Audio vstup a výstup

Téměř každá radiostanice obsahuje na svém těle konektor nebo konektory, jejichž prostřednictvím lze k radiostanici připojit náhlavní soupravu. Ta je vhodná především z toho důvodu, že umožňuje uživateli využívat radiostanici bez nutnosti držení v ruce např. při sportu, jízdě na motocyklu apod. Fyzická realizace připojení bývá provedena obvykle dvěma způsoby. Oba využívají podobného zapojení, rozdíl je pouze v použitých konektorech.

- *Varianta 1*: pro připojení externího reproduktoru i mikrofonu je využit společný konektor typu JACK 3,5 mm ve stereofonním provedení. Ve velmi ojedinělých případech se lze setkat s menší verzí tohoto konektoru i v rámci této varianty.
- *Varianta 2*: pro připojení externího reproduktoru je využit konektor typu JACK 3,5 mm v monofonním provedení. Externí mikrofon je připojen prostřednictvím typu JACK 2,5 mm, také v monofonním provedení.



Obr. 4.4 Zapojení náhlavních sad

Dle obr. 4.4 je zřejmé, že každá náhlavní souprava obsahuje tlačítko PTT (Push To Talk). Po jeho stisku začne radiostanice vysílat a je možné hovořit. Vzájemné propojení radiostanice a interface je realizováno prostým spojením jejich audio vstupů a výstupů. Využit lze stíněný dvoužilový kabel s příslušnými konektory. Hardwarová emulace zapojení náhlavní sady, zvláště funkce tlačítka PTT, je realizována v zapojení interface.

### 4.7.3 Úroveň modulace

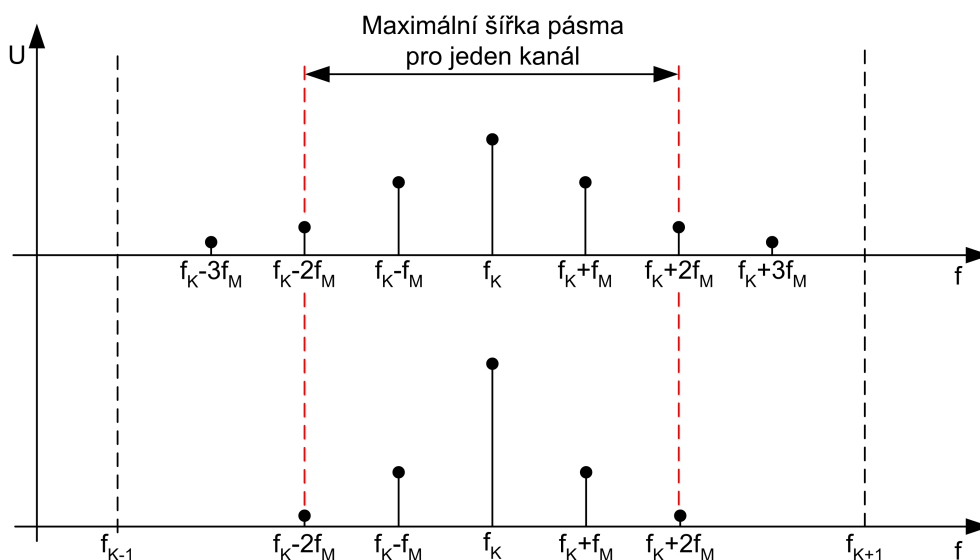
Z principu frekvenční modulace vyplývá, že kmitočet VCO je závislý na úrovni modulačního signálu. Maximální dovolená změna je dle kapitoly 4.3 stanovena na 3 kHz. Mikrofonní předzesilovač v radiostanici je nastaven tak, aby tento maximální zdvih nebyl za obvyklých podmínek překročen. Jelikož radiostanice nemá ve většině případů implementovanou funkci hlasového kompresoru nebo alespoň limiteru, obvyklou podmínkou se rozumí fakt, že uživatel hovoří do mikrofonu běžnou hovorovou hlasitostí. V případě překročení modulačního zdvihu dochází k následujícím jevům:

- Zvyšuje se zkreslení přednesu na přijímací straně. Pokud je frekvenční zdvih překročen velmi výrazně, stává přenesený hovor zcela nesrozumitelný. Představa, že zvýšený hlas zajistí lepší srozumitelnost na druhé straně je zcela nesprávná.
- Vlivem větší šířky využívaného pásma dochází k přeslechům na sousední kanály. Tento stav je z provozního hlediska zcela nepřijatelný a jedná se o hrubé porušení povolovacích podmínek.

Na obr. 4.5 je zobrazeno frekvenční spektrum FM modulace s mírně zvýšeným a normálním zdvihem. Střední frekvence pracovního kanálu  $f_K$  vychází z tab 4.1. Stejně je tomu i u sousedních kanálů  $f_{K-1}$  a  $f_{K+1}$ . Pro názornou ilustraci přeslechů u zvýšeného zdvihu byl zvolen harmonický

modulační signál  $f_M$  s limitním kmitočtem 3 kHz. Krajiní spektrální čáry normálního zdvihu jsou umístěny na hranici povolené kanálové šířky pásma, u zvýšeného již zasahují do prostoru sousedního kanálu a způsobují přeslechy.

Pro zpětnou rekonstrukci modulačního signálu využívá demodulátor uvedenou kanálovou šířku pásma. PMR radiostanice vzhledem ke své ceně neobsahuje příliš strmý druhý mezifrekvenční filtr. Důsledkem tohoto faktu je jistá tolerance demodulátoru k mírnému překročení modulačního zdvihu nebo k posunutí kmitočtu VCO, jak je uvedeno v kapitole 4.3.



Obr. 4.5 Amplitudy spektrálních čar zvýšeného (nahore) a normálního (dole) zdvihu

Interface obsahuje na svém nf výstupu prvek, kterým lze úroveň výstupního signálu v širokém rozsahu ovlivňovat. Je tedy nutné, aby tato úroveň byla přizpůsobena citlivosti externího mikrofonního vstupu radiostanice a zároveň nebyl překročen maximální frekvenční zdvih. K přesnému nastavení je vhodné použít spektrální analyzátor. Jedná se o přístroj velmi nákladný, dostupný pouze v laboratořích a servisních centrech. Alternativní metoda, která však pro účel této diplomové práce zcela postačuje, spočívá v nastavení úrovně modulace poslechem. Zvuk z reproduktoru přijímací radiostanice musí být jasný, nezkrácený a jeho hlasitost mírně nižší než běžný hovor.

## 4.8 Dosah

Dosah PMR radiostanic resp. jejich vzájemná radiová dostupnost je velmi relativní pojem, který závisí na mnoha faktorech. Může se pohybovat od 100 m v husté městské zástavbě do cca 100 km v ideálních podmínkách. Výběr několika nejpodstatnějších faktorů je seřazen dle důležitosti v následujících bodech:

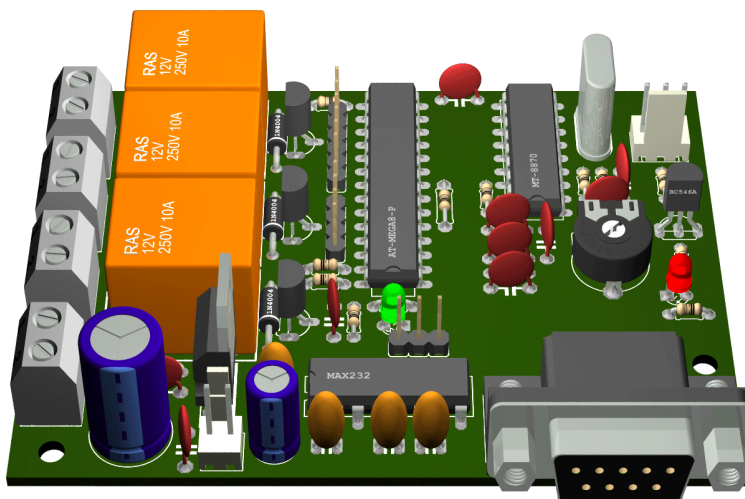
- *Poloha*: jedná se o nejdůležitější faktor a to především z důvodu, že charakter šíření přímé vlny na pracovních frekvencích PMR pásma má již jisté známky chování světelného paprsku. Mezi vlastnosti šíření patří neprůchodnost terénem, silný útlum při průchodu železobetonovými zdi a plechovými objekty. Naopak nevdí cihlové zdi, sklo, plasty. Ideální poloha komunikujících radiostanic je taková, aby mezi nimi byla zajištěna optická viditelnost. V případě, že to není možné, jsou akceptovatelné zdi budov, ve kterých je radiostanice umístěna. Jakákoliv další zástavba výrazně zkracuje dosažitelnou vzdálenost.

- *Anténa*: povolovací podmínky stanovují, že anténa musí být integrální a neodnímatelnou součástí radiostanice. Její zisk musí být navrhnut tak, aby nedošlo k překročení maximálního vyzářeného výkonu. Z hlediska příjmu je samozřejmě výhodnější směrová zisková anténa, ovšem u PMR radiostanic je nutné využít integrovanou. Výrazněji tak stoupá význam polohy, pro dosažení většího dosahu.
- *Citlivost přijímače*: bývá u všech PMR radiostanic velmi podobná a je z hlediska uživatele neovlivnitelná. Zde je nutné podotknout, že vysoká citlivost přijímače automaticky nezaručuje lepší příjem. Důležitými vlastnostmi jsou také mezikanálová selektivita a intermodulační odolnost. Tyto kvalitativní parametry musí být s citlivostí v rovnováze.
- *Průmyslové rušení*: je forma elektromagnetického smogu, který vzniká průmyslovým provozem elektrických spotřebičů, jízdou dopravních prostředků využívající trolejové sběrače apod.
- *Odráživost*: je schopnost lomu elektromagnetické vlny o nějaký objekt. Této vlastnosti může být využito při realizaci spojení mezi dvěma body, které jsou formou přímé vlny radiově nedostupné. Využije se přitom vlny odražené, například o nějaký železobetonový dům. Odráživost je silně závislá na místních podmínkách a nemůže být považována za regulérní parametr určující dosah radiostanice. Je to ovšem možnost, kterou lze v ojedinělých případech využít.
- *Útlum trasy*: je závislý na meteorologických podmínkách. Pro krátké vzdálenosti v řádu jednotek km je útlum přenosové trasy velmi málo významný. Projevuje se převážně na delších vzdálenostech a výrazněji na vyšších pracovních kmitočtech.

Reálný dosah PMR radiostanic je v případě optické viditelnosti a prostředí malého města 3–4 km. Na venkově se tato vzdálenost prodlužuje na dvojnásobek. Nasazení PMR radiostanic v rámci velkého města se značným průmyslovým rušením degraduje dosažitelnou vzdáleností na 1–2 km a tato vzdálenost koresponduje s omezeným dosahem jakýchkoliv bezdrátových Wi-Fi spojů. Ve všech případech jsou uvedené vzdálenosti stanoveny na základě provozní zkušenosti a garantují přenos modulačního signálu bez výrazného zkreslení a šumu.

## 5 Interface

Interface je elektronické zařízení, které umožňuje využít PMR radiostanici k sestavení přenosového kanálu a distribuci povelu. Disponuje vlastním komunikačním protokolem s autentizací. Navržené DPS v systému Eagle, 3D modely a reálné fotografie jsou umístěny v příloze D.



Obr. 5.1 3D model DPS interface

### 5.1 Hardware

Schéma zapojení interface je uvedeno v příloze D. Je rozděleno na tři funkční bloky. Jejich význam je následující:

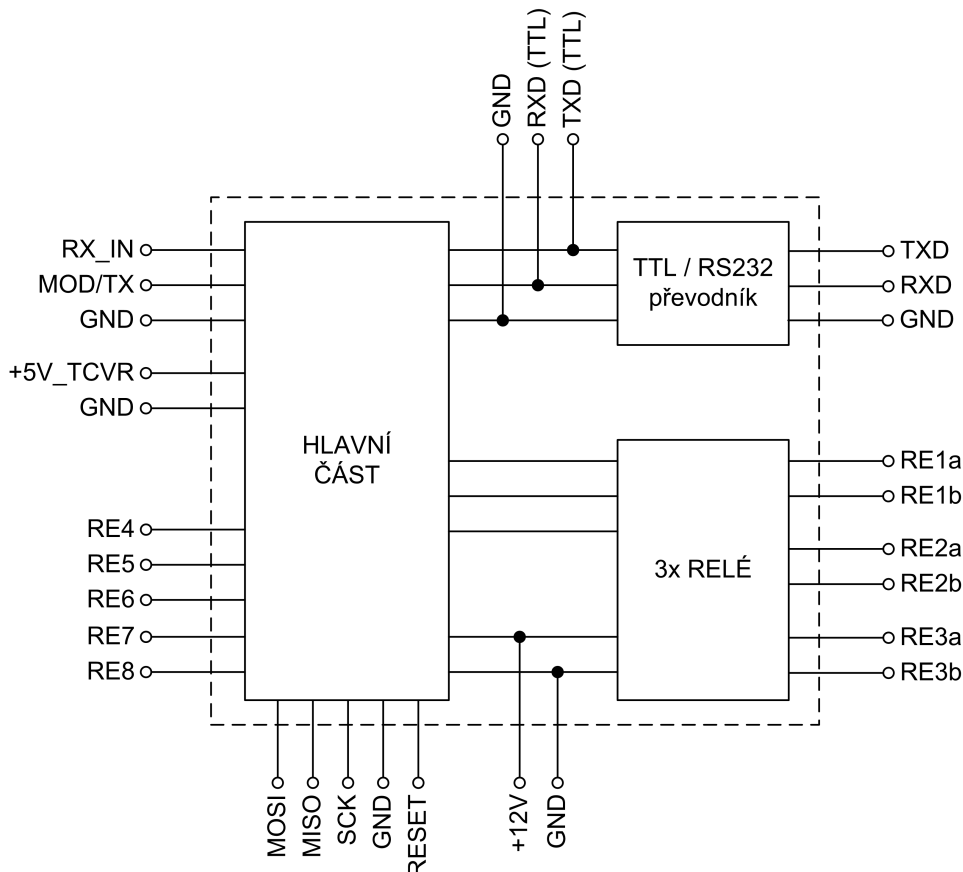
- *Hlavní část*: obsahuje základní funkční prvky interface. Jedná se o MCU, DTMF dekodér, napěťový stabilizátor a obvody umožňující připojení radiostanice. Součástí zapojení jsou dvě LED na signalizaci vysílání (červená) a příjmu DTMF tónu (zelená). Podrobněji je problematika DTMF popsána v kapitole 5.2.
- *TTL/RS232 převodník*: realizuje obousměrnou konverzi sériové linky mezi standardní úrovní RS-232 a TTL. Využit je IO MAX232. Pokud je interface umístěn v lokalitě, kde není požadováno připojení k nadřazenému systému, není nutné součástky tohoto převodníku osazovat. Sériová linka v TTL úrovni je vyvedena formou jumper lišty v každém případě. Konfiguraci interface je tak možné provést s využitím externího převodníku.
- *Relé*: jsou určena pro odpojení spotřebiče od napájení. Jejich počet vychází z obvyklých požadavků, resp. uspořádání typického uzlu bezdrátové sítě – WiFi klient, WiFi přístupový bod a síťový switch. V případě, že interface nerealizuje funkci odpojování spotřebičů, je možné obvody relé neosazovat. Jedná se zpravidla o interface, který je připojený k nadřazenému systému a lokálně nic neodpojuje.

MCU zajišťuje veškerou logickou funkcionalitu a tvoří tak zásadní součást interface. Jeho volba je založena na základě nalezení optima mezi maloobchodní dostupností kusovém množství, nízkou cenou, možnostmi implementovaných periférií, počtem vstupně/výstupních pinů a rozměry. Při porovnání produktové nabídky firem Atmel, Microchip a Motorola je nevhodnější MCU 8-mi bitový člen rodiny AVR<sup>7</sup> Atmel ATmega8.

Komunikace s interface prostřednictvím RS-232 byla zvolena z důvodu své jednoduchosti a podpory téměř ve všech operačních systémech. Implementace preferovanějšího USB rozhraní

<sup>7</sup>Označení pro rodinu 8-mi bitových RISC mikrokontrolérů z produkce firmy Atmel.

do interface je možný, ale došlo by k výraznému nárůstu ceny. A to je u jednotek, které nebudou trvale připojeny k nadřazenému systému, velmi negativní faktor. V případě nutnosti je vhodnější využít externí USB/RS-232 převodník, který je běžně dostupný v maloobchodní síti. Vyzkoušeny byly převodníky založené na obvodech Proflic PL2303 a FTDI FT232BM.



Obr. 5.2 Blokové schéma interface

Význam vstupů a výstupů:

- **+12V**: napájení interface. Jeho velikost je volena s ohledem na připojená relé. V hlavní části zapojení je umístěn stabilizátor, který zajišťuje napájení MCU, DTMF dekodéru a případně radiostanice. Klidový odběr interface bez sepnutého relé je 15 mA. Napájecí zdroj musí být patřičně proudově dimenzován. Vhodná hodnota maximálního proudového zatížení zdroje bez poklesu napětí pod 12 V je 1 A. Lze využít různé spínané síťové adaptéry.
- **+5V\_TCVR**: napájecí napětí pro radiostanici. Toto napětí je získáno ze stabilizátoru zmíněného v předchozí odrážce. Maximální možný krátkodobý proudový odběr je 800 mA a to z důvodu, že stabilizátor není chlazen. Další informace o napájení PMR radiostanic jsou uvedeny v kapitole 4.7.1.
- **TXD, TXD(TTL)**: datový výstup sériové linky RS-232 ve standardní a TTL úrovni.
- **RXD, RXD(TTL)**: datový vstup sériové linky RS-232 ve standardní a TTL úrovni.
- **MISO, MOSI, SCK, RESET**: systémové rozhraní pro programování MCU.
- **RE1a, RE1b**: výkonové výstupy relé. V klidovém stavu jsou kontakty relé sepnuty. Po příchodu povelu dojde k jejich rozpojení na stanovenou dobu. Výstupy RE2a, RE2b, RE3a a RE3b jsou funkčně analogické.

- *RE4-8*: logické výstupy. Po příchodu povelu je na těchto výstupech po stanovenou dobu nastavena log.1. Lze jej využít např. pro připojení dalších relé apod.
- *RX\_IN*: vstup nf signálu z radiostanice. Jeho úroveň není vzhledem k vlastnostem DTMF dekodéru uvedeným v kapitole 5.2.1 podstatná. Optimální hodnota je polovina rozsahu hlasitosti radiostanice.
- *MOD/TX*: sdružený výstup modulačního signálu a povelu k zahájení vysílání. Odporovým trimrem P1 je možné nastavit úroveň modulace tak, aby splňovala podmínky uvedené v kapitole 4.7.3.
- *GND*: místo s nulovým potenciálem vzhledem k napájecímu napětí a ostatním signálům.

## 5.2 DTMF

Princip DTMF spočívá na přenosu tónu, který je složen ze dvou harmonických signálů  $f_H$  a  $f_L$  o přesně definované frekvenci. Jedním tónem lze přenést 16 různých stavů – znaků. Jejich výčet je uveden v tab. 5.1.

Tab. 5.1 Tabulka frekvencí DTMF tónů

		$f_H$ [Hz]			
		1209	1336	1447	1633
$f_L$ [Hz]	697	1	2	3	A
	770	4	5	6	B
	852	7	8	9	C
	941	*	0	#	D

$$f(t) = \sin(2\pi f_H t) + \frac{3}{4} \sin(2\pi f_L t) \quad (1)$$

Klíčové vlastnosti DTMF:

- Přenosová rychlost je omezena na max. 10 znaků za sekundu a odpovídá minimálnímu garantovanému časovému rozložení 50 ms tón, 50 ms mezera. Tónová volba není primárně určena k přenosu dat, ale pouze k přenosu signalizace.
- Vzhledem k nízké přenosové rychlosti a přenosu informace ve frekvenční oblasti vykazuje velmi malou chybovost. Pokud se chyba vyskytne, má charakter nepřijatého znaku, nikoliv chybně přijatého znaku. Za předpokladu, že komunikující strany přesně ví, jaký formát zprávy mají od svého protějšku v daný moment očekávat, není nutné komunikaci zabezpečovat detekčními a opravnými kódy. U navrženého komunikačního protokolu je tento předpoklad splněn.

### 5.2.1 Příjem DTMF

Detekci DTMF lze realizovat následujícími způsoby:

- *FFT*: implementace rychlé fourierovy transformace do MCU je možná dle [14]. Prakticky však bylo zjištěno, že k dosažení přesnosti v řádu Hz by byl potřeba podstatně výkonnější hardware než je zvolený MCU. Speciální DSP (digital signal processing) mikrokontroléry jsou sice v maloobchodní síti k dispozici, ale jejich kusová cena výrazně narušuje původní nízkonákladovou koncepci interface.

- *Goertzelův algoritmus*: vyniká výrazně větší efektivitou než FFT a to především z důvodu, že výpočet probíhá pouze pro omezený počet bodů – frekvencí. V případě DTMF je jich pouze 8. Realizace v jazyce C dle [16] prokázala dostatečnou funkčnost. Z hlediska hardware je nutné, aby byl před vstupem A/D převodníku umístěn předzesilovač s limiterem a nedocházelo k překročení maximální vstupní úrovně převodníku vstupním signálem.
- *Integrovaný dekodér*: jedná se o dostupný, jednoúčelový integrovaný obvod MT8870 [13]. Jeho funkce spočívá v převodu DTMF na 4-bitové slovo. Činnost MCU je omezena na obsluhu přerušení, kdy MT8870 signalizuje, že dekodoval nový příchozí znak a jeho následné zpracování. Součástí vstupních obvodů tohoto IO je nízkofrekvenční kompresor s limiterem. Je tak zajištěn široký rozsah úrovní vstupního signálu, který bude korektně dekodován.

Z hlediska finančních nákladů, počtu použitých součástek a místa na DPS je využití integrovaného dekodéru ve všech uvedených parametrech téměř shodné s předzesilovačem nutného u Goertzelova algoritmu. Jediným zásadním rozdílem je, že při využití integrovaného dekodéru dojde k úspoře flash paměti MCU. Pouhá obsluha přerušení je paměťově méně náročnější. Integrovaný dekodér se tak jeví jako nejvhodnější řešení a byl v návrhu interface použit.

### 5.2.2 Generování DTMF

Generování DTMF je výrazně jednodušší než jeho detekce. Lze využít tyto způsoby:

- *Integrovaný transceiver*: jedná se o jednoúčelový obvod MT8880, který je schopen DTMF vytvářet, ale i přijímat. Jeho hlavní nevýhodou je maloobchodní cena, která dosahuje dvojnásobku ceny zvoleného MCU. Výhodou je naopak vynikající spektrální čistota výstupního signálu.
- *Software*: k vytváření DTMF lze využít časovače MCU, který je schopen pracovat v režimu pulzně šířkové modulace - PWM.

Pro realizaci generátoru DTMF byl zvolen softwarový způsob a to především z důvodu, že zbytečně nezvyšuje náklady a poskytuje přitom dostatečně jakostní signál.

Mezi hardwarové požadavky PWM, které je nutné splnit, patří stabilní taktovací kmitočet MCU a filtrace vzorkovacího kmitočtu. MCU má ve svém pouzdře umístěn interní RC oscilátor, který ovšem nevyniká dlouhodobou kmitočtovou stabilitou a pro realizaci PWM DTMF generátoru jej proto nelze použít. Problém stabilního taktovacího kmitočtu byl však nepřímo vyřešen volbou DTMF přijímače MT8870. Tento integrovaný obvod vyžaduje ke své činnosti externí krystalový rezonátor o frekvenci 3,579545 MHz. Signál z oscilátoru tohoto obvodu byl využit pro taktování MCU.

### 5.2.3 DTMF generátor s využitím PWM

K realizaci PWM je využit osmibitový Timer/Counter2. Je nastaven tak, aby k inkrementaci čítače TCNT2 docházelo přímo systémovým taktem MCU  $f_{takt}$  bez předděličky a režim pulzně šířkové modulace byl Fast PWM. V tomto režimu má časový průběh hodnoty tohoto čítače charakter pily. Po dosažení maximální hodnoty (255) je vždy vyvoláno přerušení a současně pokračuje inkrementace od nuly. Logický stav fyzického výstupního pinu OC2 je závislý na porovnání čítače TCNT2 s komparačním registrem OCR2.

$$TCNT2 < OCR2 \Rightarrow OC2 = \text{log.1} \quad (2)$$

$$TCNT2 \geq OCR2 \Rightarrow OC2 = \text{log.0} \quad (3)$$



Hodnotou komparačního registru OCR2 lze ovlivňovat střidu obdélníkového signálu na pinu OC2. Jeho perioda je dána vztahem 4. Uvedená hodnota současně vyjadřuje interval mezi voláním jednotlivých přerušení po přetečení čítače TCNT2.

$$T_{vz} = \frac{2^8}{f_{takt}} \quad [s] \quad (4)$$

Princip vytvoření signálu frekvence  $f$  ve tvaru sinusoidy spočívá v určení jeho okamžité hodnoty v momentě přerušení a následném nastavení komparačního registru OCR2 na tuto hodnotu. Funkce sinus je v MCU realizována formou tabulky  $T$  7-mi bitových vzorků v intervalu  $\langle 0; 2\pi \rangle$ . Velikost tabulky je  $N = 128$  vzorků. Vzhledem k tomu, že hodnota  $T_{vz}$  je konstantní, leží i okamžité hodnoty generovaného signálu  $f$  v tabulce  $T$  ve vzájemně konstantních vzdálenostech – nazváno krokem  $k$ .

$$k = \frac{2^8 N f}{f_{takt}} \quad [-] \quad (5)$$

Z důvodu vyšší rychlosti probíhají v MCU pouze celočíselné matematické operace. Zaokrouhlená hodnota uvedeného kroku  $k$  by způsobovala příliš výraznou odchylku od požadované frekvence  $f$ . Z tohoto důvodu se využívá zaokrouhlený šestnáctinásobek  $k_{16}$ . V tabulce 5.2 jsou uvedeny konkrétní hodnoty pro  $f_{takt} = 3\,579\,545$  Hz.

Tab. 5.2 Hodnoty kroků pro obě skupiny pracovních frekvencí DTMF

$f$ [Hz]	$k$	$k_{16}$	$f$ [Hz]	$k$	$k_{16}$
697	6,381	102	1209	11,067	177
770	7,049	113	1336	12,230	196
852	7,799	125	1477	13,521	216
941	8,614	138	1633	14,949	239

K určení hodnoty ukazatele  $p$  na aktuální vzorek ze sinus tabulky je využito rozšířeného ukazatele  $p_{16}$ . Při každém přetečení čítače TCNT2 je hodnota  $p_{16}$  upravena dle rovnice 6.

$$p_{16} = p_{16} + k_{16} \quad (6)$$

Pro získání skutečného ukazatele  $p$  na aktuální vzorek ze sinus tabulky se ukazatel  $p_{16}$  celočíselně vydělí 16 a současně odstraní vše, kromě spodních 7-mi bitů. Uvedená operace je pro snazší pochopení uvedena v jazyce C:

```
p = (char) ( ( p16 >> 4 ) & 0x007F )
```

Následuje získání vzorku  $s$  z tabulky  $T$ :

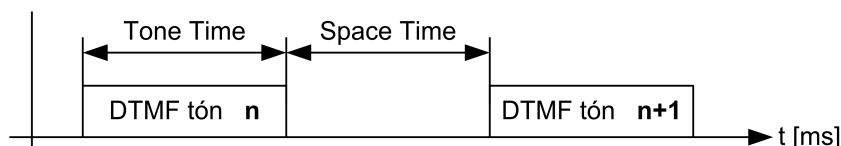
$$s = T[p] \quad (7)$$

Dle vztahu 1 je zřejmé, že uvedený postup k získání vzorku  $s$  je nutné vykonat pro obě frekvence  $f_H$  a  $f_L$ . Výsledkem jsou dva vzorky  $s_H$  a  $s_L$ . Obsah komparačního registru je pak dán vztahem 8.

$$OCR = s_H + s_L - \frac{1}{4}s_L \quad (8)$$

### 5.2.4 Parametry generátoru DTMF

Firmware umožňuje parametrizovat časové rozložení tónu formou konstant Tone Time a Space Time. Interval lze volit v rozsahu 10–2550 ms. Způsob modifikace je popsán v kapitole 5.5.3. Důvodem pro zavedení těchto konstant je skutečnost, že ani minimální časové rozložení nemusí garantovat spolehlivý přenos a detekci.



Obr. 5.3 Význam konstant Tone Time a Space Time

## 5.3 Komunikační protokol

Implementace komunikačního protokolu do firmware interface je jedním z klíčových požadavků navrhovaného systému. Účelem je, aby nadřazený systém, kterým je zpravidla server nebo router, vyslal pouze požadavek na provedení vzdáleného restartu a poté již jen po stanovenou dobu čekal na výsledek požadované operace. Jedná se tedy o jistou formu zapouzdření komunikačního protokolu a jednotný projev vzhledem k nadřazeným systémům.

Vlastnosti komunikačního protokolu:

- Unikátní adresace.
- Komunikace typu peer to peer.
- Autentizace před provedením požadovaného povelu.
- Využití dostupných nenumerických znaků v DTMF – A, B, C, D, \*, #.

Každá jednotka<sup>8</sup>, která má být schopna komunikovat s ostatními, musí mít přidělenou unikátní adresu. Tato adresa je tvořena trojicí numerických znaků a nabývá hodnot od 000 do 999. Vzhledem k reálnému dosahu radiostanic a hustotě rozložení bezdrátových síťových prvků je zvolený adresní prostor výrazně naddimenzovaný. V současné době by v síti sdružení Starhill postačovala pouze jednociferná adresa, ve střednědobém horizontu dvouciferná. Třetí cifra může být použita například k rozlišení lokality.

Všechny jednotky jsou schopny plnohodnotně komunikovat vzájemně mezi sebou. To znamená, že každá jednotka je schopna vyslat požadovaný příkaz k jakékoliv jiné jednotce nebo tento příkaz přijmout a provést. V reálném nasazení je však filozofie peer-to-peer mírně potlačena a to z důvodu, že příkaz může vyslat pouze ta jednotka, která je připojena k nějakému nadřazenému systému. V síti Starhill jsou tyto jednotky dvě. Jedná se však pouze o aplikační omezení, nikoliv implementační.

### 5.3.1 Autentizace

Smyslem autentizační schopnosti komunikačního protokolu je snížit pravděpodobnost případného zneužití jednotek cizími osobami. I když je motivace k této činnosti velmi rozporuplná, nelze zcela naivně předpokládat, že by se o to někdo nepokusil.

Nejjednodušší metodou vedoucí ke zneužití neautentizovaného přenosu povelu je záznam a opětovné odvysílání části zachycené relace mezi dvěma jednotkami. V zásadě se jedná o předstírání činnosti regulérní jednotky někým jiným. Technická náročnost tohoto postupu spočívá ve vlastnictví jednoho kusu PMR radiostanice a libovolného záznamového zařízení, např. PC

<sup>8</sup>Interface s připojenou radiostanicí.

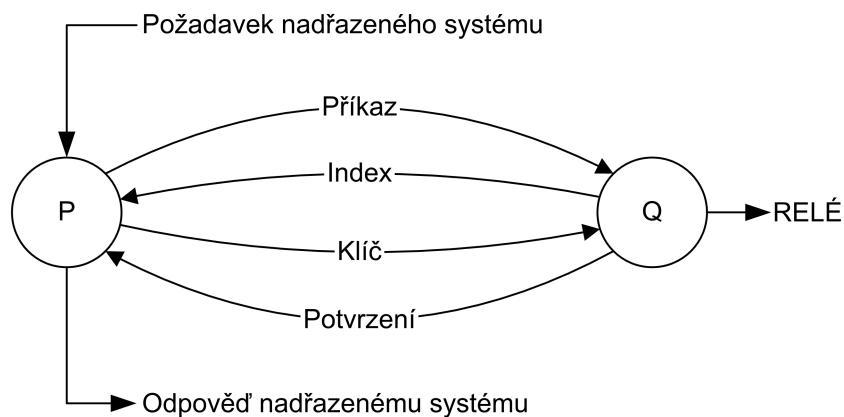
s připojeným mikrofonem. Jedná se tedy o velmi snadný úkol. Konkrétním cílem autentizace je zajistit, aby opětovným odvysíláním zachycené relace nemohlo dojít k vykonání příkazu a zároveň aplikovat takový mechanismus, který by umožňoval ověření identity regulérních jednotek. Není možné sestavit absolutně bezpečný způsob autentizace, to zde však není ani cílem. Je však nutné překonat pomyslnou hranici mezi náročností aktivit vedoucích ke zneužití jednotek a osobním profitem útočníka.

Základem navrženého způsobu autentizace je existence tabulky kódů v EEPROM paměti MCU. Velikost tabulky je omezena vzhledem k velikosti EEPROM na 200 numerických záznamů čtveřice numerických znaků v intervalu 0000–9999.

Tab. 5.3 Ukázka tabulky kódů

Index	Kód
1	2821
2	3283
...	...
200	0291

Tabulka je v MCU realizována formou statického pole proměnných typu neznaménkový integer. Jednotlivé číslice jsou uloženy v bitových čtveřicích formou BCD kódu. Přitom platí, že číslice kódu jsou dle své sestupné váhy umístěny od MSB směrem k LSB. Kód 3283 je uložen v tabulce jako hodnota 0x3283h. Tento způsob uložení kódu je výhodný především z hlediska čtení a zápisu, kde k získání jednotlivých cifer v ASCII formátu postačuje pouhý bitový posuv, vymaskování a přičtení konstanty. Nevýhodou je naopak omezení rozsahu použitelných kódů vzhledem k celému integeru. V praxi ovšem není tento nedostatek významný.



Obr. 5.4 Transakce mezi dvěma jednotkami *P* a *Q*

Na obr. 5.4 je znázorněna transakce s provedenou autentizací mezi dvěma jednotkami. Písmenem *P* je označena jednotka s připojeným nadřazeným systémem. Tato jednotka inicializuje autentizační sekvenci vysláním požadavku na vzdálený reset nebo restart. Úkolem jednotky označené *Q* je požadavek fyzicky provést.

Na vyslaný příkaz reaguje jednotka *Q* vygenerováním pseudonáhodného čísla z intervalu  $\langle 1; 200 \rangle$ . Toto číslo reprezentuje index z tabulky kódů a označuje jeden konkrétní řádek. Generování skutečně náhodného čísla je poměrně komplikované a vzhledem k četnosti využití jednotek by s použitím nějakého sofistikovaného algoritmu nedošlo k výraznému zvýšení bezpečnosti au-

tentizace. Byla proto zvolena jednoduchá implementace využívající 8-mi bitový čítač v MCU.

```
table_index = TCNT0 + 1;
if( table_index>200 ) table_index -= (table_index >> 1);
```

Čítač je inkrementován systémovým taktom MCU. Po příchodu požadavku na vygenerování pseudonáhodného čísla je k jeho aktuální hodnotě v registru TCNT0 přičtena 1. Tato operace zajistí splnění dolní hranice intervalu v případě, že aktuální hodnota čítače by byla 0. Pokud je hodnota vyšší než horní hranice intervalu, je od aktuální hodnoty odečteno číslo, které vznikne posunem o jeden bit doprava. Takto získaný index je odeslán jednotce  $P$ .

Jednotka  $P$  index přijme a na základě navržené funkce 9 provede výpočet a odeslání klíče. Podmínkou je, aby obsah tabulky kódů byl shodný s jednotkou  $Q$  a obě jednotky měly ve svém firmwaru implementovanou stejnou funkci. Na rozdíl od tabulky kódů je však tato funkce ve firmwaru zapsána fixně a nelze ji měnit jinak, než úpravou zdrojových kódů.

$$klic(k, a) = \frac{k+1}{2} + 2a - [(a+1) \bmod 3] + 11 \quad (9)$$

$k$ : kód z tabulky kódů na základě přijatého indexu

$a$ : adresa jednotky  $Q$

Kód  $k$  je z tabulky kódů přečten jako hodnota typu integer a nerespektuje původní dekadický zápis čtyřciferné hodnoty. Pokud tedy jednotka  $P$  přijme index 2, je dle ukázkové tab. 5.3 hodnota  $k=0x3283h$ , nikoliv 3283. Její vlastností je, že hodnota výsledného klíče leží pro všechny možné kombinace kódů a adres v intervalu neznaménkového integeru, tedy  $\langle 0; 65535 \rangle$ .

Jednotka  $Q$  provede před přijetím klíče od jednotky  $P$  naprosto shodný výpočet a následně ho porovná s přijatým klíčem. Pokud jsou oba klíče shodné, dojde k odeslání potvrzení o úspěšném provedení požadované akce. Pokud je však přijat chybný klíč nebo není přijat žádný, jednotka  $Q$  se z preventivních důvodů na 60 sekund zablokuje.

Uvedený způsob autentizace umožňuje případnému útočníkovi získat kombinace *adresa, index* a *klíč*. Vzhledem k tomu, že klíč je funkcí kódu z tabulky a adresy jednotky  $Q$ , je jeho výsledek pro každou jednotku typu  $Q$  různý. Na základě získaných údajů se může útočník pokoušet navazovat spojení s konkrétní jednotkou  $Q$ , jejíž údaje má k dispozici a čekat, až mu jednotka  $Q$  vygeneruje index, který útočník má k dispozici. Vzhledem k četnosti použití jednotek v řádu týdnů (dle reálných výpadků) a 60-ti sekundové blokaci jednotek typu  $Q$ , je postup odposlechu nutných údajů a jejich následné použití velmi zdlouhavé. Navíc od útočníka vyžaduje jisté odborné znalosti, protože prostý záznam komunikace nelze v tomto případě použít.

### 5.3.2 Přenos požadavku radiově dostupné jednotce

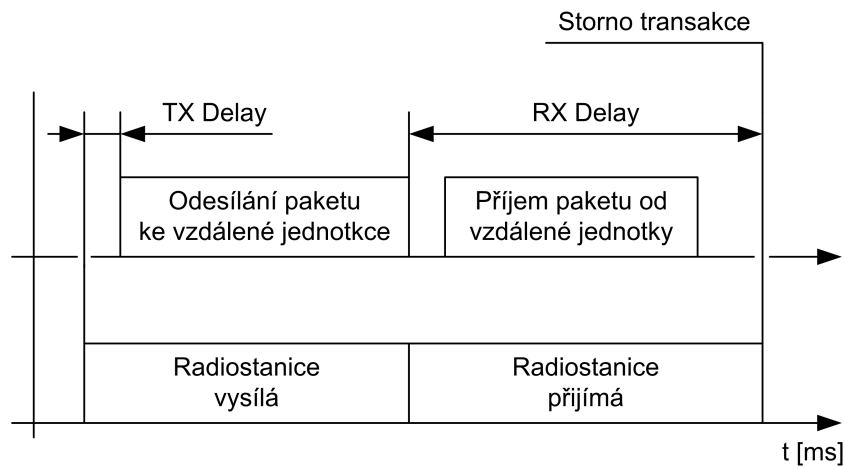
Jedná se o základní komunikační funkci jednotky, která realizuje sepnutí relé (rozepnutí napájecího obvodu spotřebiče) nebo nastavení log. 1 na požadovaném výstupu po stanovený čas. Spuštění této funkce se provede zasláním příkazu  $C$  na komunikační port jednotky. Syntax je uveden v kapitole 5.5.4.

Požadované argumenty jsou:

- Adresa vzdálené jednotky.
- Požadovaný výstup v intervalu  $\langle 1; 8 \rangle$ . Označen jako  $V$ .
- Čas požadované akce v desítkách sekund. Povolený interval je  $\langle 0; 9 \rangle$ . Hodnota 0 způsobí trvalé sepnutí relé nebo log. 1 na daném výstupu. Tento stav je následně uložen do interní EEPROM vzdálené jednotky. Opětovné rozepnutí relé nebo nastavení log. 0 lze provést zasláním příkazu s nenulovým časem na daný výstup. Čas je označen jako  $S$ .

Kromě výše uvedených parametrů je komunikace řízena dalšími nastavitelnými konstantami:

- *TX Delay*: jedná se o časovou prodlevu mezi zahájením vysílání radiostanice a počátkem přenosu paketu. Důvodem tohoto opatření existence úsporných režimů u většiny radiostanic. Podrobně popsáno v kapitole 4.5. Vhodná hodnota je 2 s.
- *RX Delay*: představuje komunikační timeout resp. čas, kdy jednotka přijímá paket od jednotky vzdálené. Pokud během přenosu dojde k chybě, po uplynutí tohoto času jednotka stornuje stávající transakci a vrátí se do výchozího stavu. Vhodná hodnota této konstanty závisí především na časové charakteristice DTMF tónů vzhledem k nejdelšímu paketu, která činí 12 znaků při přenášení klíče. Je nutné uvažovat také TX Delay, tedy dobu, než se paket začne skutečně přenášet. Orientační hodnota je cca 10 s.



Obr. 5.5 Význam konstant TX Delay a RX Delay

Popis transakce vychází z uspořádání definovaného na obr. 5.4. Jednotka *P* s adresou *ppp* předá příkaz jednotce *Q* s adresou *qqq*. Požadovaný výstup je *V* a čas *S*. Typ příkazu je *A*. Využívá se nenumerických znaků, kterými DTMF disponuje. Znak \* je startovacím znakem každého paketu.

Jednotka *P* vyše inicializační paket s potřebnými argumenty.

P: \*qqpppAVS

Jednotka *Q* odpoví zasláním indexu *iii*.

Q: \*ppqqqiii

Následuje předání klíče *kkkkk*.

P: \*qqpppkkkkk

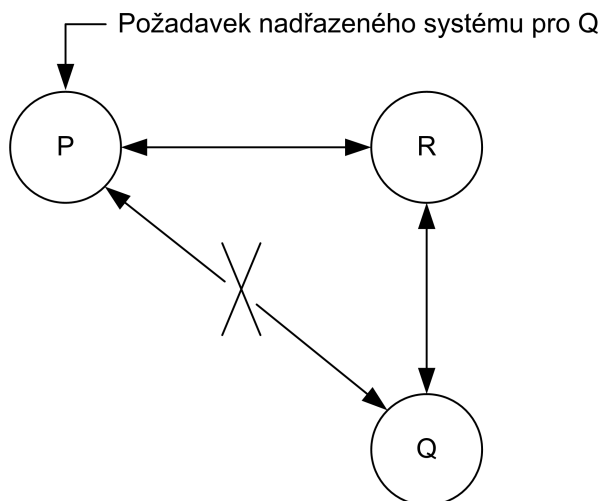
Pokud je klíč v pořádku, dojde k zaslání potvrzení o provedení požadované akce.

Q: \*ppqqq1

Následuje fyzické provedení formou sepnutí relé nebo nastavením log. 1 požadovaného výstupu *V*. Po stanovený čas *S* je jednotka nedostupná. Potvrzení o provedení je proto odesláno již předem. V každém okamžiku transakce je přijímací jednotce známo, jak dlouhý paket a s jakými argumenty má očekávat. Pokud přijme paket, kde je libovolný uvedený údaj chybný, transakci stornuje okamžitě nebo po uplynutí času RX Delay.

### 5.3.3 Přenos požadavku prostřednictvím opakovače

V případě, že mezi jednotkami není vzájemná radiová viditelnost (jsou umístěny mimo svůj vzájemný dosah), lze k uskutečnění spojení využít libovolnou jinou jednotku, která s požadovanou jednotkou radiovou viditelnost má. Tento provozní režim je nazván jako opakovač.



Obr. 5.6 Jednotka *R* ve funkci opakovače pro spojení mezi *P* a *Q*

Spuštění této funkce se provede zasláním příkazu *D* se shodnými argumenty, jako je tomu u požadavku na radiově dostupnou jednotku. Navíc je však nutné specifikovat adresu jednotky *R*, která vzájemné spojení zprostředkuje. Adresa této jednotky je označena *rrr*.

Jednotka *P* vyšle inicializační paket s potřebnými argumenty směrem k *R*. Typ příkazu je *B*.

P: \*rrrpppBVSqqq

Jednotka *R* odpoví zasláním indexu *iii*.

R: \*pprrriii

Následuje předání klíče *kkkkk*.

P: \*rrrpppkkkkk

Pokud byl klíč v pořádku, jednotka *R* si ověřila identitu jednotky *P* a nyní předá povel jednotce *Q*. Jednotka *P* bude čekat na odpověď po dobu definovanou konstantou RPTR Delay. Její hodnota je konfigurovatelná a závisí na době trvání celé transakce mezi *R* a *Q* včetně potvrzovacího paketu pro jednotku *P*. Orientační hodnota může být cca 30 s.

R: \*qqrrrAVS

Jednotka *Q* odpoví zasláním indexu *iii*.

Q: \*rrrqqqiii

Následuje předání klíče *kkkkk*.

R: \*qqrrrkkkkk

Pokud je klíč v pořádku, dojde k zaslání potvrzení o provedení požadované akce jednotce *R*. Zároveň jednotka *Q* fyzicky provede požadovanou akci.

Q: \*rrrqqq1

Jednotka *R* přeošle potvrzení inicializační jednotce *P*.

R: \*ppprrr1

## 5.4 Ověření dostupnosti

Test dostupnosti je doplňková funkce, vhodná především na diagnostické účely. Pokud se vyjde z uspořádání definovaného na obr. 5.4, je pro spuštění testu dostupnosti nutné zaslat příkaz L společně s adresou vzdálené jednotky *Q*. Vlastní test pak probíhá bez autentizace a to z důvodu, že v podstatě jakákoliv odpověď od vzdálené jednotky znamená, že jednotka je dostupná.

Jednotka *P* vyšle inicializační paket. Příkaz je typu *C*.

P: \*qqqpppC

Jednotka *Q* odpoví potvrzením své dostupnosti.

Q: \*ppqqq1

## 5.5 Firmware

Firmware<sup>9</sup> interface je vytvořen v programovacím jazyce C. Ke kompilaci zdrojových kódů byl využit volně dostupný kompilátor `avr-gcc`. Je primárně určen pro operační systémy unixového typu, ale lze jej provozovat i v prostředí Microsoft Windows. Zde je však vhodnější využít již předpřipraveného balíku WinAVR [18], který výrazně zjednodušuje instalaci a to včetně všech základních knihoven. Podrobně komentované zdrojové kódy jsou uvedeny v příloze D.

### 5.5.1 Formát příkazů

Uživatelská komunikace s jednotkou probíhá prostřednictvím sériové sběrnice v terminálovém režimu. Parametry sériového portu jsou uvedeny v kapitole 6 společně s několika ukázkami vhodného a ověřeného software. Vzhledem k omezenému prostoru flash paměti MCU bylo nutné při realizaci uživatelského terminálového prostředí pojmout minimalistickým způsobem.

Formát příkazu: <příkaz><argument>

Význam jednotlivých polí:

- *příkaz*: přesně jeden alfanumerický znak z intervalu A–Z.
- *argument*: žádný nebo několik numerických znaků. Následuje bez mezery za příkazem. V průběhu zpracování argumentu je testována korektnost zadaných hodnot. Pokud je argument nesprávný, je vypsána příslušná chybová zpráva a příkaz stornován.

### 5.5.2 Informační příkazy

- H: zobrazí nápovědu včetně základní syntaxe. Výpis nápovědy je uveden v příloze C.1.
- P: test dostupnosti lokálně připojeného interface.
- V: aktivuje/deaktivuje režim výpisu podrobných stavových informací v průběhu komunikačních operací. Lze tak snadněji nalézt příčinu případných problémů.

<sup>9</sup>Programové vybavení elektronických zařízení. Úzce vázáno na hardware.

### 5.5.3 Konfigurační příkazy

Tyto příkazy jsou určeny pro zápis a čtení konfigurace interface. Hodnoty jsou uloženy v paměti EEPROM a jsou tedy trvalé i po odpojení napájení.

- **A**: zobrazí aktuální adresu interface.
- **N<adresa>**: nastaví novou adresu interface. Argument adresa je trojčíferná numerická hodnota v intervalu 000-200.
- **R<index>**: vypíše kód z tabulky kódů. Argument index je trojčíferná hodnota ukazující na konkrétní řádek v této tabulce. Povolený interval je 000–200.
- **W<index><kód>**: zapíše nový kód do tabulky kódů. Význam argumentu index je shodný s příkazem R. Povolený rozsah čtyřčíferného argumentu kód je 0000–9999.
- **I**: zobrazí aktuální nastavení časových konstant. Výpis je proveden formou 5-tice tříčíferných numerických řetězců oddělených mezerou. První tři řetězce z levé strany reprezentují konstanty TX Delay, RX Delay a RPTR Delay. Skutečná hodnota jejich časového údaje v ms vznikne vynásobením zobrazeného čísla základním krokem 150 ms. Zbylé dva řetězce představují konstanty Tone Time a Space Time. Jejich základní krok je 10 ms.
- **K<TX Delay><RX Delay><RPTR Delay><Tone Time><Space Time>**: jedná se o funkci, která hromadně uloží všechny časové konstanty. Akceptovatelný tvar argumentů konstant TX Delay, RX Delay a RPTR Delay vznikne celočíselným podílem příslušné konstanty v ms základním krokem 150 ms. Následuje doplnění o počáteční nuly tak, aby výsledkem byl tříčíferný numerický řetězec. Modifikace argumentů konstant Tone Time a Space Time je podobný jako u předchozích konstant, pouze základní krok je zde 10 ms. Pro všechny argumenty platí, že vypočtená hodnota musí ležet v intervalu <0; 255>.

### 5.5.4 Komunikační příkazy

Následující tři příkazy jsou určeny ke spuštění komunikačních funkcí mezi lokálně připojeným a vzdáleným interface. Komunikační protokol je popsán v kapitole 5.3.

Mezi společné rysy příkazů patří argumenty adresa a opakovač, které musí obsahovat tříčíferný numerický řetězec v intervalu 000-999. Význam argumentů výstup a čas je uveden v kapitole 5.3.2.

- **C<adresa><výstup><čas>**: požadavek na přenos povelu k interface s adresou adresa a parametry výstup a čas.
- **D<adresa><opakovač><výstup><čas>**: požadavek na přenos povelu k interface s adresou adresa a parametry výstup a čas. K realizaci přenosu má být využito interface s adresou opakovač jako zprostředkovatele.
- **L<adresa>**: požadavek na ověření dostupnosti vzdáleného interface s adresou adresa. Dostupnost je potvrzena výpisem „L: Remote device alive.“
- **X<výstup><čas>**: požadavek na vykonání povelu na lokálně připojeném interface. Parametry jsou výstup a čas.

V průběhu komunikace jsou na terminál zasílány stavové zprávy ve tvaru, které mohou být snadno zpracovány nadřazeným systémem pomocí regulárních výrazů. Současně jsou však srozumitelné i pro člověka. Jedná se o tři druhy zpráv, které se liší svým prefixem.

- *Prefix EM – error message*: fatální chyba a stornování komunikace.
- *Prefix IM – info message*: informativní zpráva vhodná do logu.
- *Prefix GM – good message*: potvrzení úspěšného předání povelu.

Výpis stavových zpráv je uveden v příloze C.2.



## 6 Konfigurační aplikace

Hardware interface je navržen tak, aby umožnil konfiguraci provozních parametrů, spouštění komunikačních funkcí a zasilání stavových informací pomocí 3-žilového propojení standardní sériové sběrnice RS-232. Komunikace probíhá výhradně v terminálovém režimu v otevřeném textu. Volba tohoto způsobu vzájemné interakce mezi uživatelem a interface dovolila realizovat uživatelsky přívětivé konfigurační a provozní prostředí s minimálními softwarovými a hardwarovými nároky. Podrobný popis příkazů a jejich syntax je uveden v kapitole 5.5.

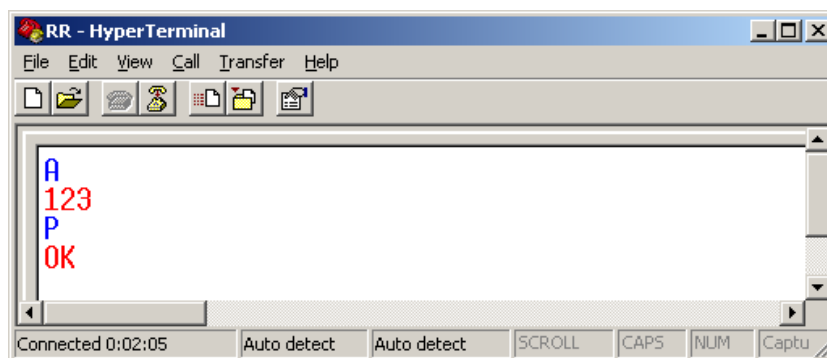
Principem terminálové komunikace uživatele s interface je odeslání příkazu na RXD vstup interface zakončený znakem konce řádku LF, CR nebo jejich kombinací a následný příjem odpovědi v podobě textového řetězce z TXD výstupu, který je rovněž zakončen znaky LF a CR. K tomu, aby mohl uživatel s interface takto komunikovat je nutné, aby na svém počítači spustil software, který tuto činnost zvládá. Zpravidla se jedná o programy typu RS-232 terminál, softwarový sériový terminál, emulátor terminálu apod. Alternativních názvů existuje celá řada, ale funkcionality všech těchto programů je v té nezákladnější podobě shodná.

Tab. 6.1 Parametry nastavení sériového portu

Parametr	Hodnota
Přenosová rychlost	9600 baud
Počet bitů v rámci	8 bit
Počet stop bitů	1
Parita	není
Řízení provozu	žádné

### 6.1 Terminály pro Microsoft Windows

- *Hyperterminal*: základní terminálový program, který je přítomen již v základní instalaci OS Windows. Spuštění se provede zadáním příkazu hypertrm v příkazovém řádku nebo prostřednictvím nabídky Start.
- *Termite*: jednoduchý terminálový software, který je možné zdarma provozovat a libovolně šířit. Jeho funkce jsou omezené, ale pro komunikaci s interface zcela dostatečné. Jednoduchost a přímočarost v nastavení je výhodou. Program je umístěn v příloze D.
- *The Terminal*: tento software byl využíván při vývoji firmware. Mezi jeho hlavní přednosti patří široké možnosti nastavení, automatické ukládání přijatých dat a využití maker. Tento program je také umístěn v příloze D.



Obr. 6.1 Pracovní okno aplikace Hyperterminal s výstupy příkazů A a P

## 6.2 Konzolové terminály pro Linux

Následující dva příklady konzolových sériových terminálů byly testovány v prostředí distribuce Linux Debian 4.0 (Etch). Je nutné podotknout, že pro manuální komunikaci je značně nepraktické využívat přímý přístup k sériovému portu pomocí zápisu a čtení souboru `/dev/ttySn` (resp. `/dev/cuaan` v BSD systémech) a to zejména pro nízkou interaktivitu. Lze ji výrazně zvýšit jednoduchým skriptem pro BASH nebo PERL, ale popis řešení tohoto problému není cílem této kapitoly.

- *cu*: byl vyvíjen v rámci projektu UUCP (Unix to Unix CoPy). V uvedené distribuci je k dispozici v rámci binárního balíku `uucp`.

Spuštění terminálu na sériový port `/dev/ttyS0` (COM1) a provedení 2 příkazů A a P:

```
starhill:~# cu -l /dev/ttyS0 -s 9600 -h
Connected.
A
785
P
OK
```

- *Minicom*: interaktivní textový emulátor terminálu. Je dostupný v binárním balíku `minicom`.

```
Welcome to minicom 2.2
OPTIONS: I18n
Compiled on Jan 7 2007, 18:00:43.
Port /dev/ttyS0
Press CTRL-I Z for help on special keys
A
785
P
OK
CTRL-I Z for help | 9600 8N1 | NOR | Minicom 2.2 | VT102 | Online 00:00
```

## 6.3 Aplikace RadioReset device configurator

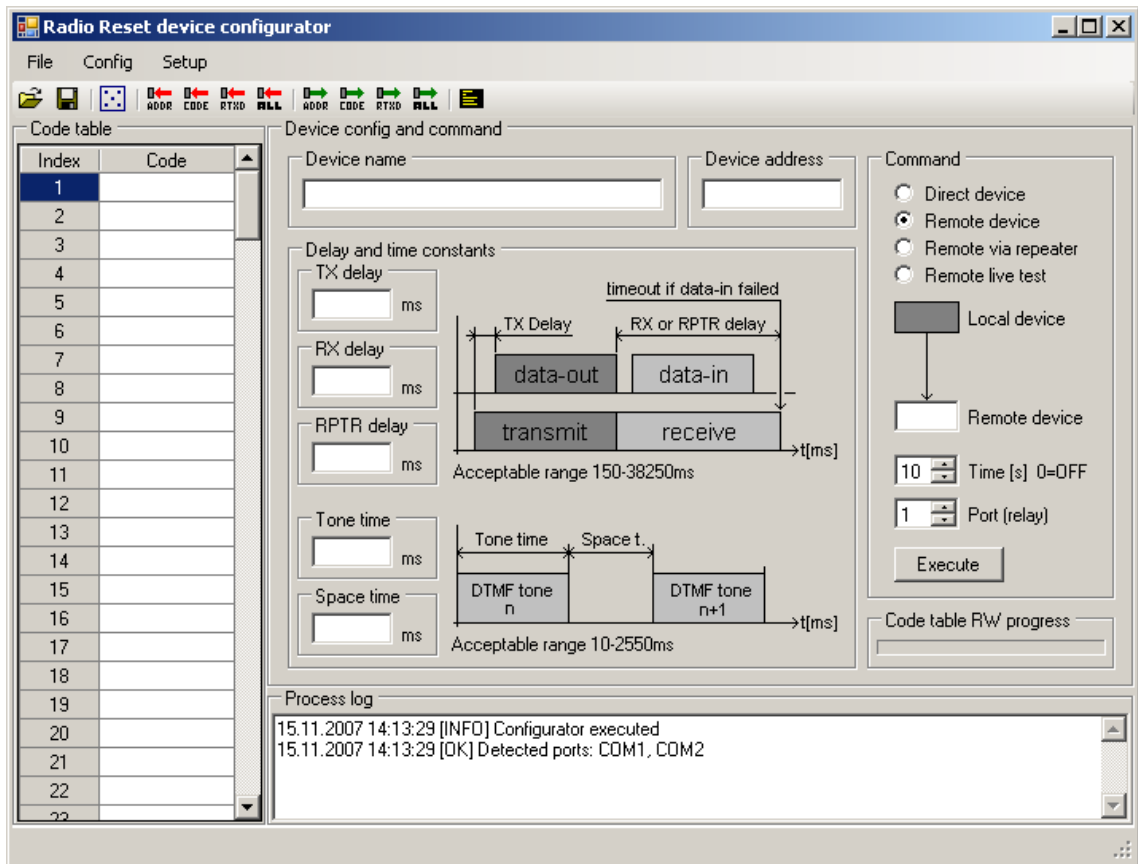
Jedná se o jednoúčelovou aplikaci s grafickým uživatelským rozhraním, která byla vyvinuta výhradně na použití s navrženým interface. Je primárně určena pro operační systém Microsoft Windows, ale lze ji spustit i v prostředí OS Linux s využitím software z projektu Mono. Hlavní motivací ke vzniku této aplikace byla snaha o zjednodušení a zrychlení konfigurace interface. Zdrojový kód aplikace je umístěn v příloze D.

Klíčové vlastnosti aplikace:

- Možnost exportu a importu konfigurace interface.
- Naplnění tabulky kódů pseudonáhodnými hodnotami.
- Ovládání všech konfiguračních funkcí jedním kliknutím myši.
- Jednoduchý terminál pro diagnostické účely.
- Standardní uspořádání a vzhled aplikace.

Systémové požadavky:

- Microsoft Windows 98 a novější
- Microsoft .NET Framework 2.0
- Fyzický sériový port nebo převodník USB/RS232

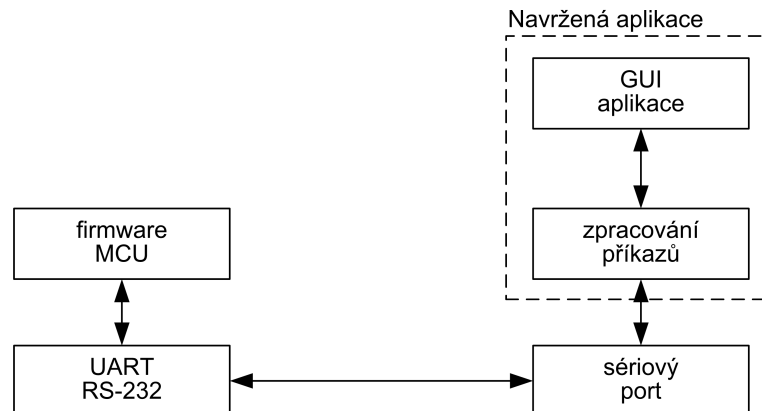


Obr. 6.2 Aplikace Radio Reset device configurator

### 6.3.1 Komunikační vazba aplikace s interface

Koncepce aplikace je navržena tak, aby grafické uživatelské rozhraní (GUI) tvořilo další (vyšší) vrstvu nad terminálovou komunikací. Uživatel díky tomu nemusí znát přesnou syntax příkazů interface, ale stačí mu pouze znalost významu jednotlivých konfiguračních parametrů.

Nastavení sériového portu dle tab. 6.1 je v aplikaci implementováno na úrovni zdrojového kódu. Uživatel má možnost výběru konkrétního portu, které má ve svém operačním systému k dispozici.



Obr. 6.3 Vzájemná vazba navržené aplikace s interface

### 6.3.2 Struktura tříd aplikace

Aplikace je vytvořena v současné době velmi populárním objektově orientovaném programovacím jazyce C#. Microsoft uvolnil k užívání vývojové prostředí Visual Studio 2005 C# Express Edition, které poskytuje rozdílnou od plné verze Visual Studia zcela zdarma. Tento programovací nástroj se tak jeví jako velmi perspektivní a pro vývoj konfigurační aplikace vhodný.

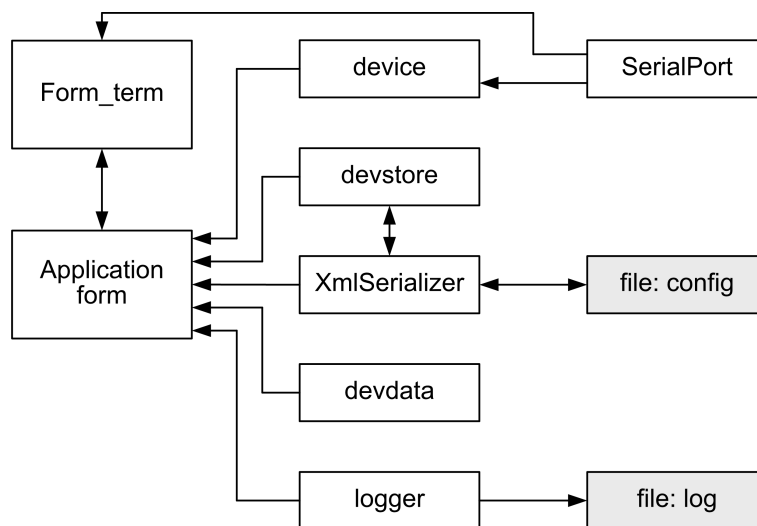
Nutnou podmínkou pro vývoj a spuštění aplikace je přítomnost Microsoft .NET Framework 2.0 nebo jakékoliv vyšší verze v operačním systému. Aktuálně je k dispozici již verze 3.0. Důvodem pro instalaci frameworku minimální verze 2.0 je využívání třídy `SerialPort` z jmenného prostoru `System.IO.Ports` pro komunikaci pomocí sériového portu počítače. Tato třída není v nižších verzích frameworku implementována a musela by být nahrazena externím řešením.

Popis klíčových tříd implementovaných ve frameworku:

- *Application Form*: třída základního formuláře.
- *SerialPort*: třída pro komunikaci prostřednictvím sériového portu počítače.
- *XmlSerializer*: export/import atributů libovolné třídy do XML struktury.

Popis externích tříd:

- *device*: třída obsahuje metody pro komunikaci s interface pomocí dostupných příkazů a realizuje reakci na přijatá stavová data.
- *devdata*: metody této třídy zajišťují detekci a korekci typové a syntaktické správnosti hodnot v prvcích formuláře.
- *devstore*: jedná se serializovatelnou třídu, která obsahuje výhradně atributy datových položek popisující konfiguraci interface. Těchto atributů využívá třída `XmlSerializer` při operacích uložit (otevřít) konfiguraci do (ze) souboru.
- *logger*: elementární logovací mechanismus. Popsán v kapitole 6.3.3.



Obr. 6.4 Bloková struktura klíčových tříd a jejich vzájemných logických vazeb

### 6.3.3 Výpis stavových informací

Pro výpis stavových a chybových hlášení byl zvolen způsob událostního logu ve formátu:

*datum čas [typ] zpráva*

Typ zpráv může být:


- *OK*: zpráva o úspěšném provedení nějaké operace.
- *ERROR*: zpráva o zjištěné chybě.
- *DEVICE*: řetězec přijatý z interface.
- *INFO*: informativní zpráva.




Výstup těchto zpráv je prováděn do Process logu na formuláři a zároveň do souboru log.txt umístěného ve stejném adresáři jako binární soubor aplikace. V konstruktoru třídy logger je proveden test na úspěšné otevření souboru log.txt. Pokud selže, je ukládání událostí do souboru po dobu běhu aplikace potlačeno.

### 6.3.4 Význam vstupně-výstupních polí formuláře

- *Device name*: alfanumerický identifikátor konkrétních konfiguračních parametrů. Tento řetězec se do interface neukládá, slouží pouze pro snazší identifikaci při uložení nebo otevření konfigurace ze souboru. Zpravidla reprezentuje název interface nebo lokality. Např. „AP Světlov.“ Lze použít i diakritiku.
- *Device address*: numerická unikátní 3-ciferná adresa interface. Povoleny jsou všechny kombinace mezi 000 až 999.
- *TX delay*: čas mezi zahájením vysílání radiostanice a přenosem DTMF paketu v milisekundách. Akceptovatelný interval je 150–38200 ms v kroku 150 ms. Pokud je zadána hodnota mimo tento krok, metoda GetDelays() třídy devdata ji na požadovaný krok upraví.
- *RX delay*: doba, po kterou přijímací interface čeká na doručení kompletního DTMF paketu od vysílací strany v milisekundách. Pravidla pro zadání času jsou shodná s předchozím prvkem.
- *RPTR delay*: doba, po kterou přijímací interface čeká na doručení kompletního DTMF paketu od vysílací strany v případě čekání na potvrzení od opakovače. Hodnota je opět v milisekundách a pravidla pro zadání času jsou opět shodná s předchozím prvkem.
- *Tone time*: délka trvání DTMF tónu v milisekundách. Akceptovatelný interval je 10–2550 ms v kroku 10 ms. Pokud je zadána hodnota mimo tento krok, metoda GetDelays() třídy devdata ji na požadovaný krok opět upraví.
- *Space time*: délka trvání mezery mezi jednotlivými DTMF tóny v milisekundách. Pravidla pro zadání času jsou shodná s předchozím prvkem.
- *Code table*: obsahuje 200 kódů určené pro autentizaci. Povoleny jsou 4-ciferné numerické hodnoty v rozsahu 0000–9999. Pokud je zadána numerická hodnota v jiném formátu, je zkrácena nebo doplněna nulami zleva.









## 6.4 Funkce aplikace pro konfiguraci interface

-  *Load config*: po výběru správného souboru s konfigurací dojde k načtení prvků z XML struktury v souboru do atributů instance třídy devstore a k následnému naplnění prvků formuláře.

-  *Save config*: obsah všech konfiguračních prvků formuláře se přesune do atributů instance třídy devstore a v podobě XML struktury uloží do zvoleného souboru.
-  *Generate random codes*: tabulka kódů se naplní pseudonáhodnými hodnotami kódů s využitím třídy Random.
-  *Terminal*: spustí instanci formuláře Form\_term, který umožňuje pracovat s interface podobně jako ze softwarového RS-232 terminálu.

Dále následují funkce pro zápis a čtení. Tyto funkce využívají metod třídy device. Na počátku každé z nich dojde k ověření, zda-li je k vybranému sériovému portu interface připojen a zda-li je funkční. Využívá se zde příkazu P.

Při zvolení funkcí pro zápis dochází k otestování povolených hodnot obsažených v polích dle předchozí kapitoly 6.3.4 a po úspěšném zápisu k následné verifikaci. Pokud vše proběhne v pořádku, je až nyní zápis prohlášen za úspěšný. Při volbě funkcí na čtení testuje příslušná metoda třídy device korektnost přijatých dat. Pokud je tento test úspěšný, je aktualizováno příslušné pole ve formuláři.

-  *Write address*: zápis adresy prostřednictvím příkazu N.
-  *Write code table*: zápis hodnot do tabulky kódů. 200x opakuje příkaz W se vzrůstajícím indexem a kódem z tabulky. Průběh zápisu je indikován prvkem Code table RW progress.
-  *Write delay and times*: zápis časových konstant prostřednictvím příkazu K.
-  *Write all*: zápis adresy, tabulky kódů a časových konstant najednou.
-  *Read address*: načtení adresy příkazem A.
-  *Read code table*: načtení všech hodnot tabulky kódů pomocí opakování příkazu R se vzrůstajícím indexem. Průběh čtení je indikován prvkem Code table RW progress.
-  *Read delay and times*: načtení časových konstant příkazem I.
-  *Read all*: načtení adresy, tabulky kódů a časových konstant najednou.

#### 6.4.1 Funkce aplikace pro komunikační režim interface

Komunikační funkce interface je možné ovládat prvky v seskupení Command. Jsou k dispozici 4 volby:

- *Direct device*: provede sepnutí relé na lokálně připojeném interface. Je nutné zvolit výstupní port a čas sepnutí relé. Pokud na daném výstupu není relé připojeno, dojde k nastavení log. 1 na daném výstupním pinu.
- *Remote device*: zahájí přenos povelu ke vzdálenému interface s adresou Remote device. Význam portu a času je shodný jako v předchozím případě.
- *Repeater device*: zahájí přenos povelu ke vzdálenému interface s adresou Remote device. K realizaci spojení využije opakováče s adresou Repeater device. Význam portu a času je opět shodný jako v předchozím případě.
- *Live test*: provede test dostupnosti na jednotku s adresou Remote device.

## 7 Dohledový systém

V předchozích kapitolách byl popsán návrh hardware interface, který je schopen ve spojení s vhodnou radiostanicí přenést povel k restartu ke vzdálenému síťovému zařízení – prvku. Již tato základní funkce umožňuje rychle a poměrně efektivně řešit celou řadu problémů, které mohou na vzdálené straně vzniknout. Nevýhodou stále zůstává nutnost manuální obsluhy administrátorem nebo jiné pověřené osoby.

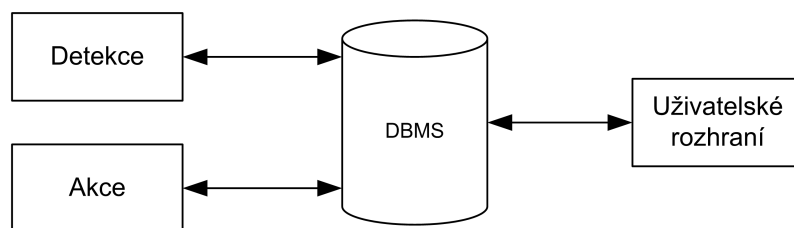
Mezi hlavní koncepční požadavky této diplomové práce patří návrh dohledového systému, který by byl do jisté míry autonomní ve vztahu k problémovým vzdáleným prvkům. Je zcela zřejmé, že v provozu bezdrátové počítačové sítě mohou vzniknout situace, kdy je zásah administrátora nutný vzhledem k jeho analytickým schopnostem a zkušenostem. Každý výpadek síťového prvku má svoji příčinu a úkolem administrátora je, aby se pokusil o nápravu vedoucí ke snížení četnosti výpadků, nejlépe k jejich úplné eliminaci. To lze provést například vhodnou úpravou konfigurace prvku v závislosti na předchozích zkušenostech s produkty daného výrobce a jejich chováním. Navrhnout zcela autonomní systém, který by byl schopen i těchto pochodů je náročné a vzhledem k úhrnné četnosti výpadků prvků v řádu týdnů i poměrně zbytečné.

Zcela dostatečná funkcionalita dohledového systému spočívá v detekci síťové dostupnosti testovaného prvku a případnému využití navržených interface k jeho vzdálenému restartu. Samozřejmostí je i informování administrátora, že k restartu prvku došlo a je pak na jeho uvážení, jaké kroky provede dál. Výhodou tohoto řešení je snadná implementovatelnost a nízká reakční doba při řešení výpadků, kdy vzdálený restart vyhovuje.

### 7.1 Struktura dohledového systému

Dohledový systém je založen na vzájemné interakci čtyř funkčních celků:

- *DBMS (Database management system)*: datová implementace topologie sítě, jednotlivých prvků, jejich vzájemných vazeb a provozních stavů.
- *Detekce*: detekuje dostupnost nebo nedostupnost prvků v závislosti na jejich vzájemných vazbách. Spouštěno periodicky.
- *Akce*: komunikuje s interface a realizuje případný restart lokálních nebo vzdálených síťových prvků. Spouštěno periodicky po detekci.
- *Uživatelské rozhraní*: prostředí pro administraci prvků v DBMS a konfiguraci parametrů pro detekční a akční funkční celky. Podrobný popis je uveden v kapitole 8.



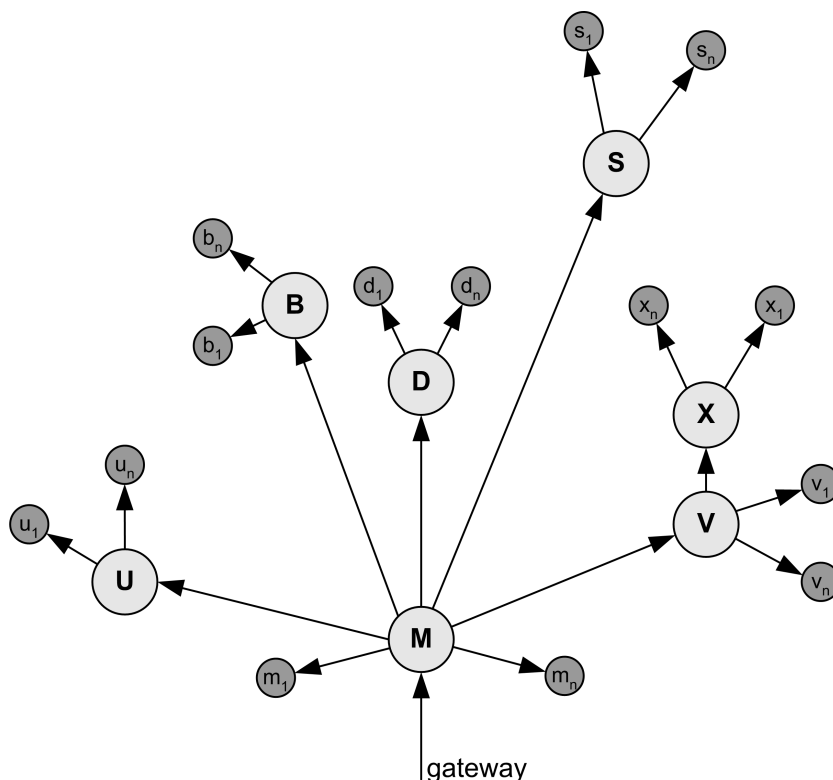
Obr. 7.1 Struktura funkčních celků dohledového systému

## 7.2 Topologie sítě Starhill

Na obrázku 7.2 je znázorněna obecná topologie bezdrátové počítačové sítě Starhill. Z uvedeného obrázku je patrné, že topologie je zakreslena ve formě orientovaného stromového grafu.

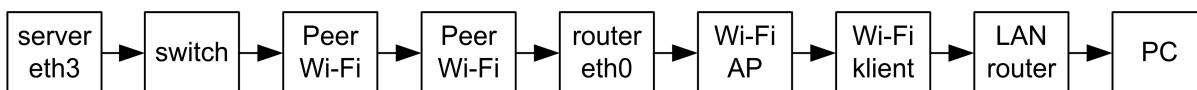
Faktický význam atributů grafu:

- *Vrcholy (velké)*: jedná se o souhrn aktivních síťových zařízení v jednotlivých lokalitách, které jsou v majetku sdružení. Např. směrovače, přístupové body nebo přepínače. Ve vrcholu M je umístěn hlavní server zajišťující většinu síťových služeb a je pro provoz sítě klíčový.
- *Vrcholy (malé)*: jedná se o uživatelské podsítě, které jsou tvořeny zpravidla klientem bezdrátového spoje a následným metalickým vedením k uživateli nebo skupině uživatelů.
- *Orientované hrany*: představují spoje mezi lokalitami. Jsou tvořeny ve většině případů bezdrátovými linkami. Orientace hran v topologii je uvedena ve smyslu procházení grafem od počátečního vrcholu směrem ke hranici grafu. Datové linky jsou samozřejmě duplexní nebo poloduplexní.



Obr. 7.2 Obecná topologie sítě Starhill

Reálná topologie sítě se od uvedené obecné liší především expanzí vrcholů na jednotlivé konkrétní prvky, které jsou na dané lokalitě obsaženy. Popisuje tak skutečnou cestu (větev) od počátečního vrcholu až k uživateli ve formě posloupnosti prvků, prostřednictvím kterých datová komunikace probíhá. Tuto posloupnost musí dohledový systém znát nebo si ji být schopen prohledáním grafu sestavit.



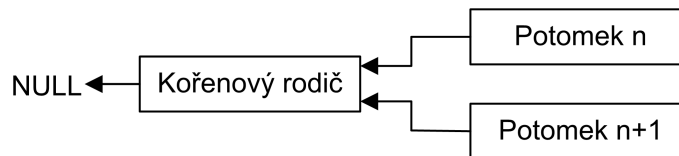
Obr. 7.3 Příklad skutečné posloupnosti prvků od vrcholu M k uživateli v uzlu v1



Z uvedené obecné topologie sítě je zřejmé, že neexistuje žádný způsob, jak se k nějakému vrcholu dostat více než jednou cestou. Z toho vyplývá, že v síti nejsou použity žádné záložní linky. Tento nedostatek lze sice považovat za zásadní, ale je nutné podotknout, že operační rádius použitých radiostanic a Wi-Fi spojů je 2–5 km. Nasazení redundantních bezdrátových spojů v tak malém prostoru by ještě více zkomplikovalo situaci v již tak velmi intenzivně využívaném frekvenčním pásmu 2.4 GHz a 5 GHz.

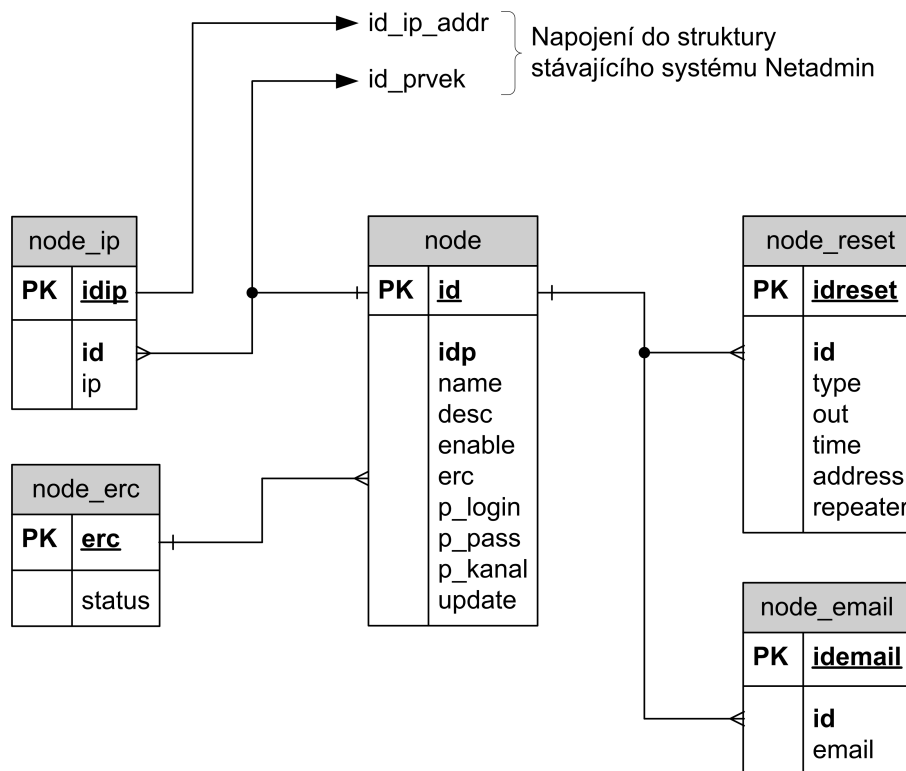
### 7.3 Implementace datových struktur v DBMS PostgreSQL

Jedna z nejjednodušších metod popisu vrcholů a hran stromového grafu je využití vazby rodič – potomek. Každý prvek obsahuje ve svém popisu informaci o prvku, který ho v dané větvi předchází. Vyjímkou je prvek kořenový s hodnotou NULL.



Obr. 7.4 Znárodnění vazby mezi rodičem a potomky

K implementaci stromu byl zvolen relační databázový systém PostgreSQL z důvodu, že na serveru síť Starhill je od roku 2004 provozován systém správy sítě Netadmin, který PostgreSQL využívá. Základní jádro tohoto správního systému bylo vytvořeno v rámci bakalářské práce [11]. Dohledový systém je v rámci této diplomové práce chápán jako samostatná aplikace a lze jej takovým způsobem provozovat. Ve skutečnosti je však požadována možnost kooperace se zmíněným správním systémem Netadmin. Propojení je naznačeno v relačním diagramu na obr. 7.5.



Obr. 7.5 Entitně-relační diagram datové struktury dohledového systému

### 7.3.1 Tabulka prvků

Obsahuje seznam síťových prvků včetně jejich vzájemných vazeb. Součástí jsou také doplňkové atributy pro uschování dalších informací o prvku.

```
CREATE TABLE node (
  id serial NOT NULL,
  idp integer NOT NULL,
  name character varying(50),
  desc text,
  enable boolean DEFAULT true,
  erc smallint NOT NULL DEFAULT 1,
  p_login text,
  p_pass text,
  p_kanal text,
  update timestamp without time zone DEFAULT now(),
  CONSTRAINT node_pkey PRIMARY KEY (id) )
```

Význam atributů:

- *id*: unikátní autoinkrementační identifikátor prvku. Referenční integrita relačních vazeb s ostatními tabulkami je zajištěna pravidlem *rule\_node\_delref*. Uvedeno v příloze D.
- *idp*: identifikátor předka. Případná hodnota NULL je z důvodu snazší programové obsluhy v detekčním a akčním funkčním celku reprezentována numerickou nulou.
- *name*: krátké textové označení prvku. Je využíváno při výpisu stromu prvků v uživatelském rozhraní a událostními funkcemi.
- *desc*: atribut pro poznámku o vlastnostech prvku.
- *enable*: povoluje resp. zakazuje zpracování prvku a celé následující větve.
- *erc*: chybový kód popisující stav prvku. Vysvětleno v kapitole 7.6.
- *p\_login*, *p\_pass*, *p\_kanal*: informativní textové atributy pro uložení přihlašovacích údajů a pracovního kanálu Wi-Fi prvku. Údaje jsou v tabulce uloženy ve formě prostého textu. Veškerá komunikace s DBMS však probíhá zabezpečeně.
- *update*: časový údaj poslední modifikace záznamu v tabulce.

### 7.3.2 Tabulka emailových adres

Tato tabulka obsahuje seznam emailových adres, na kterých bude administrátor informován při určitých změnách stavu daného prvku.

```
CREATE TABLE node_email (
  idemail serial NOT NULL,
  id integer NOT NULL,
  email character varying(50) CHECK (email::text ~ '^.+@.+$'::text),
  CONSTRAINT node_email_pkey PRIMARY KEY (idemail) )
```

Význam atributů:

- *idemail*: autoinkrementační primární klíč.
- *id*: referenční ukazatel do tabulky node na identifikátor konkrétního prvku.
- *email*: emailová adresa. S atributem je svázán regulární výraz, který je platný pouze pro nenulový řetězec s výskytem znaku @ a minimálně jedním dalším znakem na obou jeho stranách. Jedná se o základní ochranu proti překlepům, nikoliv pravidlo pro zaručení absolutní syntaktické platnosti emailové adresy.

### 7.3.3 Tabulka IP adres prvků

Obsahem této tabulky je seznam IP adres těch prvků, které jsou spravovatelné a mají přidělenou IP adresu na některém ze svých rozhraní.

```
CREATE TABLE node_ip (
  idip serial NOT NULL,
  id integer NOT NULL,
  ip inet NOT NULL,
  CONSTRAINT node_ip_pkey PRIMARY KEY (idip) )
```

Význam atributů:

- *idip*: autoinkrementační primární klíč.
- *id*: referenční ukazatel do tabulky node na identifikátor konkrétního prvku. Umožňuje realizovat vazbu mezi prvkem a jeho IP adresou, resp. několika IP adresami.
- *ip*: IP adresa prvku. Datový typ inet zajistí validitu IPv4 nebo IPv6 adresy. Případná síťová maska je odstraněna a je zbytečné ji zapisovat.

### 7.3.4 Tabulka parametrů pro interface

Tabulka `node_reset` je určena pro uložení parametrů interface (adresy, výstupní port, část). Dohledový systém tyto údaje využívá při vytváření příkazu pro interface, který je připojen k lokálnímu sériovému portu a zajišťuje spojení se vzdálenými interface.

```
CREATE TABLE node_reset (
  idreset serial NOT NULL,
  id integer NOT NULL,
  type character varying(1) NOT NULL CHECK (email ~ '^[DPR]{1}$'),
  out smallint CHECK (out ~ '^[1-8]{1}$'),
  time smallint CHECK (time ~ '^[1-9]{1}$'),
  address character varying(3) CHECK (time ~ '^([0-9]{3})?$'),
  repeater character varying(3) CHECK (time ~ '^([0-9]{3})?$'),
  CONSTRAINT node_reset_pkey PRIMARY KEY (idreset) )
```

- *idreset*: autoinkrementační primární klíč.
- *id*: referenční ukazatel do tabulky node na identifikátor konkrétního prvku.
- *time*, *out*: výstupní port a čas aktivace výstupu. Hodnoty jsou testovány na správnost.
- *address*, *repeater*: adresa vzdáleného interface, případně opakovače.
- *type*: Typ příkazu, který bude lokálně připojenému interface odeslán. Možnosti jsou:
  - *D (direct)*: bude sepnuto relé na lokálně připojeném interface.
  - *P (remote)*: povel bude zaslán vzdálenému interface v radiové dostupnosti.
  - *R (repeater)*: k přenosu povelu bude využito funkce opakovače.

### 7.3.5 Tabulka textových popisů stavů

Tabulka obsahuje krátké textové popisy stavů, ve kterých se prvek nachází.

```
CREATE TABLE node_erc (
  erc smallint NOT NULL,
  status character varying(255),
  CONSTRAINT node_erc_pkey PRIMARY KEY (erc) )
```

Význam atributů:

- *erc*: identifikátor chybového stavu s vazbou na atribut `erc` z tabulky `node`.
- *text*: krátký textový popis příslušného chybového stavu. Maximální velikost 255 znaků.

## 7.4 Typy prvků

Dle obsahu tabulek `node_ip`, `node_reset` a `node_email` lze stanovit šest typů prvků. Uvažovány jsou tyto faktory:

- Testovatelnost.
- Resetovatelnost.
- Informovatelnost.

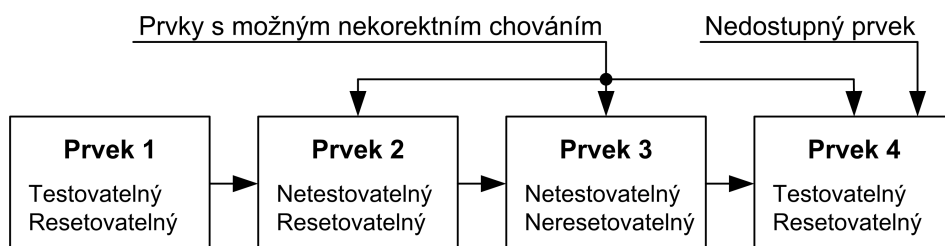
### 7.4.1 Testovatelný prvek

Jedná se o aktivní prvek ve smyslu jeho vlastní síťové komunikace na úrovni TCP/IP. Alespoň na jednom ze svých rozhraních má nastavenou IP adresu, resp. několik adres. V síti Starhill se jedná zpravidla o přístupové body, klientské adaptéry, hardware peer-to-peer spojů a spravovatelné přepínače. Dohledový systém je schopen ověřit dostupnost prvku na všech jeho aktivních rozhraních. Předpokladem je korektní nastavení směrovacích tabulek u směrovačů, které jsou předcházejícími prvky testovanému prvku na stejné větvi. IP prostor prvku nesmí být izolován.

Nutnou podmínkou testovatelnosti je jeden nebo několik záznamů v tabulce `node_ip` v referenčním vztahu k danému prvku.

### 7.4.2 Netestovatelný prvek

Tento prvek obsahuje hardware pro obsluhu komunikace na fyzické a linkové vrstvě dle ISO/OSI. Jedná se převážně o nespravovatelné přepínače a média konvertory z optického kabelu na metalický apod. Dohledový systém není schopen otestovat dostupnost těchto prvků. Je však schopen odvodit korektní funkčnost netestovatelného prvku za podmínky, kdy libovolný následující prvek dané větve je testovatelný a současně dostupný.



Obr. 7.6 Znárodnění vazby mezi rodičem a potomky

Z obr. 7.6 vyplývá, že v případě nedostupného prvku 4 nemusí být jako problémový okamžitě označen právě on, ale nekorektní funkci mohou vykazovat také jeho netestovatelní předchůdci 3, 2 a to až do úrovně testovatelného předchůdce 1. Tato vlastnost byla při návrhu systému zohledněna.

### 7.4.3 Resetovatelný prvek

Pokud lze k resetu nebo restartu prvku využít navrženého interface, je takový prvek označen jako resetovatelný. Není důležité, zda-li k předání povelu dojde lokálně, vzdáleně nebo s využitím opakováče. Klíčový je pouze fakt, aby se dohledový systém mohl pokusit obnovit korektní činnost sám bez asistence administrátora. Podmínkou resetovatelnosti je záznam, resp. záznamy v tabulce `node_reset` pro daný prvek.

Pokud je resetovatelný prvek zároveň prvkem netestovatelným, dohledový systém vykoná restart tohoto prvku za předpokladu, že za tímto prvkem následuje v dané větvi prvek, který je

testovatelný a nedostupný. Modelová situace je zobrazena na obr. 7.6 a týká se prvku 2. Tímto způsobem lze vyřešit problém nekorektně pracujících síťových přepínačů.

#### 7.4.4 Neresetovatelný prvek

Neresetovatelný prvek nelze prostřednictvím navržených interface resetovat nebo restartovat. V případě chybné funkce je nutné, aby administrátor provedl osobní zásah v lokalitě, kde je prvek umístěn.

Pokud je tento prvek testovatelný, dohledový systém jej označí jako nedostupný a informuje administrátora o vzniklé situaci. V opačném případě dojde k situaci, kdy není možné určit chybný prvek, nelze se pokusit činnost prvku obnovit a daná větev sítě zůstane nedostupná.

#### 7.4.5 Informovatelný a neinformovatelný prvek

Informovatelný prvek poskytuje dohledovému systému komunikační kanál, který je využit pro předávání zpráv o klíčových stavech prvku administrátorovi nebo jiným pověřeným osobám. Jedná se například o detekci nedostupnosti, selhání pokusu o restart nebo opětovnou detekci dostupnosti. Neinformovatelný prvek tímto komunikačním kanálem nedisponuje a stavové informace jsou zobrazeny v uživatelském rozhraní nebo v záznamu událostí – v logu.

### 7.5 Způsob detekce dostupnosti prvků

Realizovat detekci dostupného nebo nedostupného prvku lze několika způsoby. Jeho volba závisí na schopnostech detekčního systému a především na vlastnostech testovaného prvku. Následující výčet možností detekce je seřazen od nejvíce vhodného po nejméně vhodný způsob.

- *SNMP (Simple Network Management Protocol)*: jedná se o komunikační protokol, který je primárně určen na správu síťových zařízení, sběr dat a jejich vyhodnocování. Síťový prvek musí mít tento protokol implementovaný ve svém software nebo firmware a být schopný realizovat službu SNMP agent. Velmi vhodná je i schopnost prvku odesílat zprávy o nastalé události bez přímého vyžádání. SNMP umožňuje pružně reagovat na stav některých vnitřních parametrů prvku a tím snížit pravděpodobnost jeho selhání. V levnějších prvcích podpora SNMP často chybí.
- *ICMP (Internet Control Message Protocol)*: alespoň částečná implementace tohoto protokolu je ve všech síťových prvcích využívající TCP/IP a to bez ohledu na pořizovací cenu nebo kvalitu. ICMP datagramy Echo Request a Echo Reply jsou základními diagnostickými metodami jak ověřit dostupnost prvku. V mnoha operačních systémech je k dispozici konzolový nástroj ping, jehož výstupem je čas odezvy testovaného prvku. Nejedná se však o konstantní hodnotu, ale o závislost na typu přenosových cest, radiové interferenci okolních bezdrátových spojů a zatížení sítě v oblasti konvergence ke kapacitnímu maximu linky. Na základě změny doby odezvy není možné o prvku cokoliv usuzovat, je možné vyvodit pouze binární závěr, kde první stav reprezentuje libovolnou odezvu a druhý žádnou odezvu.
- *Síťová služba*: detekce spočívá v navázání spojení s nějakou službou (např. http, ftp, dns, ..), kterou síťový prvek realizuje a základě úspěšnosti komunikace stanoví závěr, zda-li je prvek dostupný nebo nikoliv. Tento způsob detekce není příliš vhodný a to především z důvodu, že aplikační služba má výrazně vyšší nároky na systémové zdroje než ICMP protokol. Může tak způsobovat u méně výkonných prvků zpomalení jejich užitečné činnosti. Jedná se však o jedinou detekční variantu v případě, že není možné využívat SNMP nebo ICMP.

V síti Starhill jsou v současnosti využívány prvky s podporou SNMP velmi málo. Důvodem je větší finanční náročnost bez přidané hodnoty v podobě vyšší spolehlivosti. ICMP protokol

je v rámci celé sítě povolen a jeho bezpečnostní omezení na úrovni firewallu nepředstavují pro detekci dostupnosti podstatný problém.

Vzhledem k aktuálnímu stavu sítě je nyní vhodné využít k detekci dostupnosti pouze ICMP. SNMP bude možné implementovat do dohledového systému později za předpokladu, že počet prvků disponujících SNMP agentem bude výrazněji vyšší.

### 7.5.1 Nástroj ping a argumenty

Detekce dostupnosti prvků je realizována programem ping s následujícími argumenty:

```
ping <ipv4_adresa> -s1 -i1 -w2
```

respektive

```
ping6 <ipv6_adresa> -s1 -i1 -w2
```

Význam jednotlivých argumentů:

- <ipvx\_adresa>: IP adresa testovaného prvku
- s1: velikost paketu 1 byte + ICMP hlavička
- i1: interval mezi požadavky na odezvu, nastavena 1 sekunda
- w2: maximální čas běhu programu ping, nastaveny 2 sekundy

Vzhledem ke zvoleným časovým hodnotám dojde po spuštění programu ping s uvedenými argumenty k odeslání 2 požadavků. Pokud je přijata odpověď na alespoň jeden z těchto požadavků, je prvek označen jako dostupný.

Z relačního diagramu na obr. 7.6 vyplývá, že vztah mezi tabulkou `node` a `node_email` je typu 1:N. Tabulka `node_email` tak může obsahovat více záznamů s IP adresami vztaženými k jednomu prvku. To ve skutečnosti znamená, že prvek má více síťových rozhraní nebo je na jednom rozhraní nastaveno více IP adres. Dohledový systém tyto několikanásobné záznamy zpracovává konjunktivně – aby byl prvek prohlášen za dostupný, musí být dostupný na všech jeho IP adresách. Příkladem takového prvku může být bezdrátový klientský směrovač, který obsahuje 2 nezávislá rozhraní: wireless a LAN. Každé rozhraní má jinou IP adresu a v případě selhání prvku může dojít k chybné funkci pouze jednoho rozhraní. Druhé poté pracuje zcela korektně a je dostupné. V případě disjunktního přístupu k detekci jednotlivých IP adres by byl takový prvek vždy prohlášen za dostupný i přesto, že je jeho funkce chybná.

## 7.6 Chybový kód prvku

Chybový kód je numerická reprezentace aktuálního stavu prvku. Stavem prvku je zde chápána informace o jeho dostupnosti a o způsobu řešení nedostupnosti dohledovým systémem. Jedná se o celočíselnou numerickou hodnotu, která je uložena v atributu `erc` tabulky `node`. V současné době je využit interval hodnot  $\langle 1; 10 \rangle$ , kde 1 reprezentuje korektně pracující dostupný prvek a 10 prvek, kde již nelze jeho činnost obnovit a je nutný zásah administrátora.

Dohledový systém v závislosti na aktuálním chybovém kódu, dostupnosti a možnosti restartu vyhodnocuje další možné varianty operací s prvkem a podle zvolené akce nastavuje jiné chybové kódy. Zvyšující se hodnota chybového kódu vyjadřuje i snižující se míru pravděpodobnosti, že dohledový systém bude schopen dostupnost prvku obnovit.

Význam jednotlivých chybových kódů:

- *Kód 1*: tento kód je výchozí pro všechny prvky a popisuje stav bezchybného a kompletně dostupného testovatelného prvku. Za předpokladu, že je prvek netestovatelný, tak je tímto kódem označen vždy.

- *Kód 2:* jedná se o speciální kód v rámci celé řady. Akční část dohledového systému provede u takto označeného resetovatelného prvku vynucený restart. Následuje nastavení chybového kódu 1. V případě neresetovatelného prvku dojde pouze k nastavení kódu 1. Důvodem pro zavedení tohoto chybového kódu je implementace funkce manuálního restartu prvku administrátorem.
- *Kód 3:* u testovatelného prvku byla detekována první nedostupnost. Ta může být způsobena krátkodobým výpadkem přenosových cest, rekonfigurací prvků nebo servisním zásahem. Prvek není v tomto případě prohlášen za nedostupný a vyčká se do dalšího spuštění detekce dostupnosti.
- *Kód 4:* byla zjištěna opětovná nedostupnost. V tomto případě je již prvek prohlášen za nedostupný a bude následovat akční řešení této nedostupnosti metodou restartu nebo informováním administrátora, pokud je to možné.
- *Kód 5:* pokud je prvek resetovatelný, byl proveden úspěšný restart prostřednictvím interface. Úspěšnost je zde chápána ve smyslu potvrzení předání povelu vzdálenému interface, nikoliv v obnovení síťové komunikace. Ta již samozřejmě může být obnovena, ale bude zjištěna až při dalším spuštění detekce dostupnosti.
- *Kód 6:* v případě úspěšného restartu resetovatelného vzdáleného prvku nebyla obnovena síťová komunikace nebo je prvek neresetovatelný. V obou případech byli nalezeni resetovatelní a netestovatelní předchůdci, kteří byli restartováni.
- *Kód 7:* ani po restartu resetovatelných a netestovatelných předchůdců tohoto prvku nebyla obnovena jeho dostupnost. Byl proveden druhý pokus o obnovu síťové komunikace opětovným restartem resetovatelných a netestovatelných předchůdců.
- *Kód 8:* byl proveden neúspěšný pokus o restart resetovatelného prvku ve smyslu potvrzení předání povelu interface. Tento stav mohl být zapříčiněn rušením na přenosové trase, výpadkem napájení ve vzdálené lokalitě apod. Pokud nedojde k obnovení dostupnosti, bude proveden druhý pokus.
- *Kód 9:* byl proveden druhý neúspěšný pokus o restart resetovatelného prvku. Pokud i po tomto pokusu nedojde k obnovení dostupnosti, bude proveden poslední pokus.
- *Kód 10:* koncový stav. Již není možné dostupnost prvku obnovit a je definitivně nedostupný. O této skutečnosti je informován administrátor, aby mohl učinit kroky vedoucí k obnovení dostupnosti.

Varianta zkráceného popisu chybových kódů je využita pro zobrazení nápovědy v uživatelském rozhraní dohledového systému. Texty jsou uloženy v tabulce `node_erc`.

Tab. 7.1 Zkrácené verze popisu chybových kódů

Kód	Zkrácený textový popis
1	Všechno je OK
2	Bude proveden vynucený reset
3	Dočasně nedostupný. Čekám na další test
4	Nedostupný. Bude proveden restart, pokud je to možné
5	Restart prvku byl úspěšně proveden
6	Restart neobnovil dostupnost, netestovatelní resetovatelní předchůdci restartováni
7	Druhý pokus restartu netestovatelných resetovatelných předchůdců
8	První pokus o restart prvku selhal
9	Druhý pokus o restart prvku selhal
10	Prvek je definitivně nedostupný, je nutný osobní zásah

## 7.7 Činnost dohledového systému

### 7.7.1 Periodické spouštění

Detekční a akční část dohledového systému není koncipována ve smyslu trvale běžící služby (daemon), ale jako periodicky spouštěný programový kód, který provede požadovanou činnost a následně se ukončí. Volbou exekuční periody lze velmi snadno ovlivnit reakční dobu systému a také míru zatížení OS. Tento faktor je významný především při realizaci dohledového systému na jednoúčelových základních deskách určených pro malé routery s nízkými systémovými rezervami.

Detekční a akční část dohledového systému jsou nezávislé funkční celky, které pracují se společnými daty v DBMS. Z tohoto důvodu jsou veškeré operace s DB uzavřeny do transakcí pro případ časového souběhu obou částí. Z hlediska efektivity je souběh značně nevhodný a to především z důvodu možných opakovaných akčních zásahů. Ty mohou nastat v případě, kdy původně nedostupný resetovatelný prvek detekční část ještě neoznačila jako dostupný, ale souběžně spuštěná akční část provedla opětovný restart takového prvku. Vhodné je spustit akční část bezprostředně po ukončení detekční části. Znázorněno na obr. 7.7.

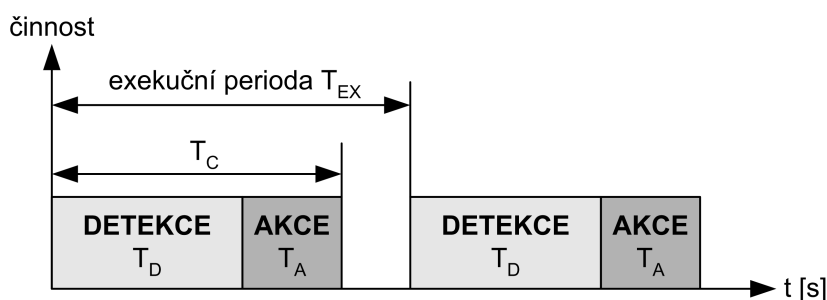
Volba exekuční periody závisí na době běhu detekční a akční části. Hlavním požadavkem na tuto hodnotu je, aby v klidovém stavu (bez akčních zásahů k nedostupným prvkům) nedocházelo k souběhu dvou instancí detekce dostupnosti.

Celkovou dobu běhu detekční části  $T_D$  a akční části  $T_A$  lze určit ze vztahu:

$$T_C = T_D + T_A \quad [s] \quad (10)$$

Pro exekuční interval platí podmínka:

$$T_{EX} > T_C \quad (11)$$



Obr. 7.7 Časová posloupnost spouštění detekce a akce

Hodnota  $T_A$  je v klidovém stavu bez akčních zásahů téměř nulová. Režijní čas na připojení k DBMS a prohledání stromu prvků je v tomto případě nevýznamný. V případě restartu nedostupného prvku se hodnota  $T_A$  zvyšuje až na desítky sekund a to v závislosti na nastavení interface. Z praktického hlediska je vhodné ponechat v exekuční periodě prostor pro  $T_A$  cca 60 s.

Hodnota  $T_D$  je závislá na celkovém počtu aktivních prvků, na počtu záznamů v tabulce `node_ip` a časových parametrech nástroje ping. Při uvedeném nastavení je doba detekce jedné IP adresy 2 s. V síti Starhill je nyní testováno 132 IP adres, hodnota  $T_D$  je tedy 264 s. Ze vztahu 10 plyne, že celková hodnota  $T_C$  je 324 s a na základě podmínky 11 je exekuční perioda  $T_{EX}$  nastavena na hodnotu 360 s.



### 7.7.2 Prohledávání stromu prvků v detekční části

Po každém spuštění detekční části dohledového systému je zahájeno prohledávání stromu prvků. Jeho počátek je určen kořenovými prvky, které nemají žádného předka. K realizaci prohledávání je využita jednoduchá rekurzivní funkce (prohledávání do hloubky), která v seznamu prvku prvků hledá ty prvky, jenž mají nastaveny jako předka aktuálně zpracováváný prvek.

Během prohledávání jsou u každého prvku seskupovány tyto údaje:

- Aktuální chybový kód.
- Seznam IP adres, resp. testovatelnost.
- Zjištěná dostupnost nebo nedostupnost.

Prohledávání následujícího zbytku větve za aktuálně zpracováváním prvkem je přerušeno po splnění jedné z uvedených podmínek:

- Prvek má nastaven atribut `enable` na `false`.
- Prvek má nastaven chybový kód v intervalu  $\langle 4; 10 \rangle$ . Jedná se o stav, kdy všechny ostatní prvky za prvkem, který je označen jako nedostupný, jsou teoreticky také nedostupné a je zbytečné je prohledávat a jakkoliv je označovat jako nedostupné.

Na základě seskupených údajů o aktuálně prohledávaném prvkem je soustavou pravidel vyvozen nový chybový kód, který je danému prvkem nastaven. Jakákoliv další změna chybového kódu aktuálně prohledávaného prvku je možná až při následujícím spuštění detekční části dohledového systému.

### 7.7.3 Prohledávání stromu prvků v akční části

Metoda prohledávání stromu prvků je shodná s detekční částí. Rozdíl je v informacích, které jsou během prohledávání o prvkem seskupovány.

- Aktuální chybový kód.
- Seznam IP adres, resp. testovatelnost.
- Seznam parametrů pro interface, resp. resetovatelnost.
- Seznam netestovatelných a resetovatelných předchůdců aktuálního prvku.

Podmínky pro přerušování prohledávání následujícího zbytku větve jsou podobné jako v detekční části. Pokud chybový kód aktuálně prohledávaného prvku leží v intervalu  $\langle 4; 10 \rangle$ , je využito seskupených údajů k provedení restartu. Následně je dle soustavy pravidel stanoven nový chybový kód a je prvkem nastaven.

### 7.7.4 Rozhodovací mechanismus neresetovatelných prvků

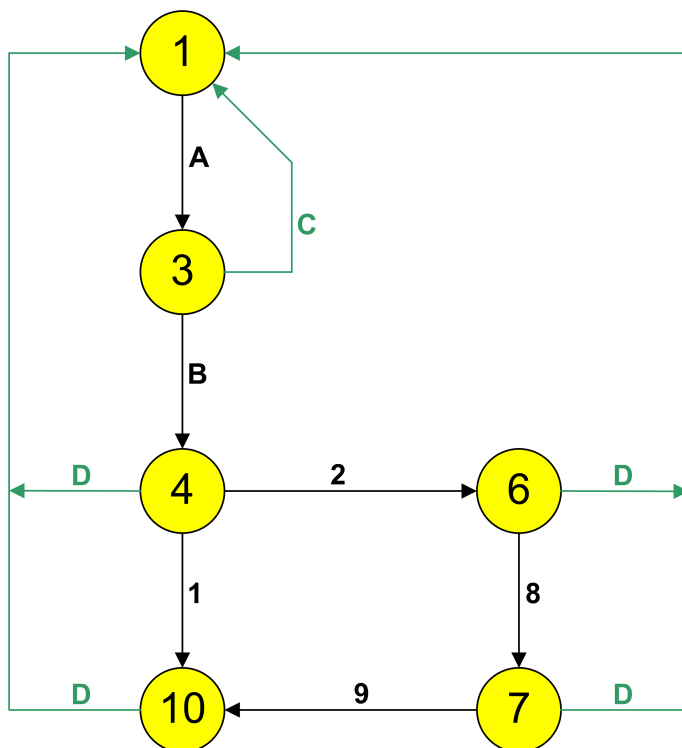
Obrázek 7.8 popisuje vzájemné vazby chybových kódů (vrcholy), kterých může nabývat neresetovatelný testovatelný prvek. Hrany v uvedeném grafu reprezentují přechody mezi chybovými kódy a jsou výsledkem posouzení seskupených informací daného prvku rozhodovacím mechanismem (sadou pravidel). V rámci jednoho spuštění detekční nebo akční části může být proveden pouze jeden přechod pro každý prvek. Z důvodu vyšší přehlednosti jsou obě dvě části zobrazeny v jednom grafu a odděleny stylem označení hran.

- Hrany A-D realizuje detekční část dohledového systému.
- Hrany 1, 2, 8, 9 realizuje akční část dohledového systému.

Význam jednotlivých hran:

- *A*: detekční část dohledového systému zjistila nedostupnost doposud dostupného prvku. V tomto okamžiku není prvek chápán už jako nedostupný, dojde pouze k nastavení nového kódu 3.
- *B*: došlo k opětovné detekci nedostupnosti. Je nastaven chybový kód 4 a detekční část dohledového systému označila prvek jako nedostupný. Další průchody grafem bude realizovat akční část.
- *C*: dohledový systém zjistil opětovnou dostupnost, bude nastaven chybový kód 1.
- *D*: pokud má prvek chybový kód v intervalu  $\langle 4; 10 \rangle$  a je detekována dostupnost, dojde k nastavení chybového kódu 1. Nebere se přitom ohled na stádium rozpracovanosti případných restartů. Na informační emailové adresy bude odeslána příslušná zpráva z tab. 7.2.
- *1*: jedná se o nejkratší cestu ke koncovému stavu, ve kterém již nemá dohledový systém schopnost obnovit dostupnost prvku. Tato situace nastane za podmínky, kdy neresetovatelný prvek nemá žádné netestovatelné resetovatelné předchůdce. Na informační emailové adresy bude odeslána příslušná zpráva z tab. 7.2.
- *2*: pokud má neresetovatelný prvek netestovatelné resetovatelné předchůdce, změní se jeho chybový kód na 6. Současně se všichni netestovatelní resetovatelní předchůdci restartují.
- *8*: společně s nastavením chybového kódu 7 bude proveden druhý pokus o restart netestovatelných resetovatelných předchůdců.
- *9*: opakovaný restart netestovatelných resetovatelných předchůdců nezajistil obnovení dostupnosti. Bude nastaven konečný chybový kód 10 a odeslána zpráva o nedostupnosti na emailové adresy odeslána příslušná zpráva z tab. 7.2.

Při průchodu hranami *B*, *D*, *1*, *2*, *8* a *9* je do záznamu událostí (logu) vložena informace se jménem prvku, aktuálním časem a provedenou akcí.

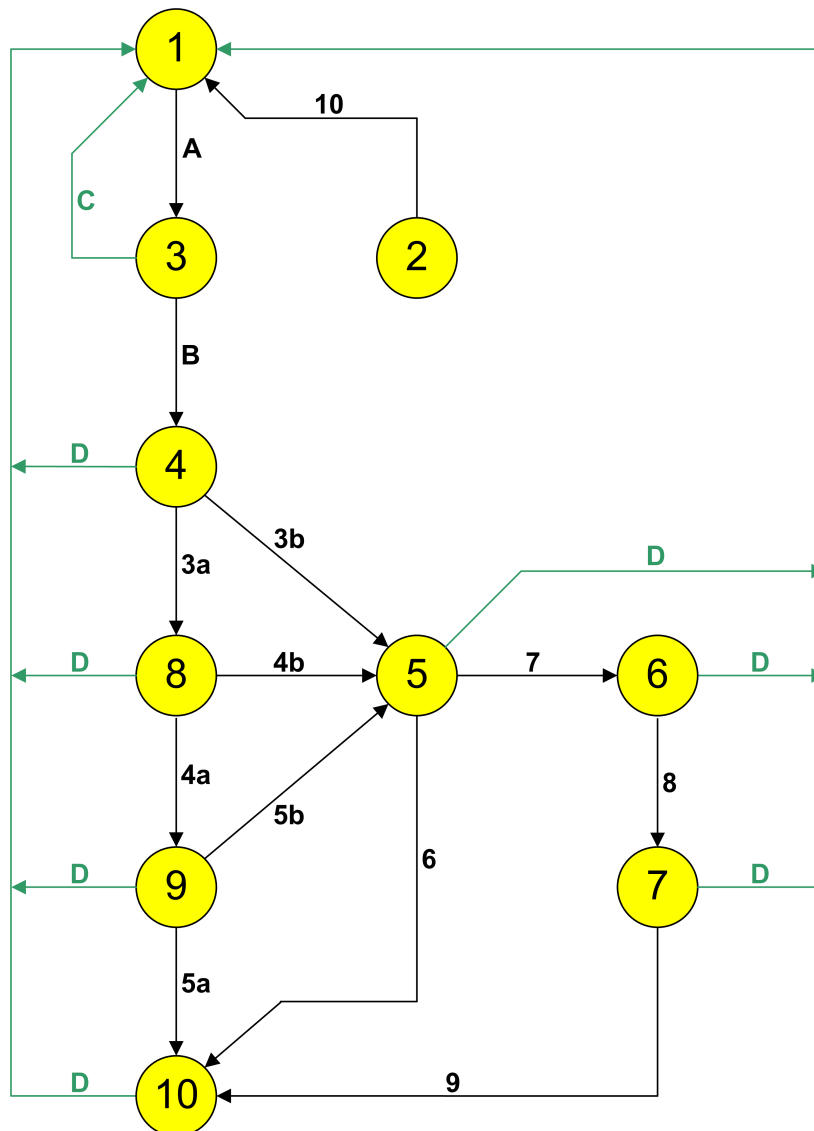


Obr. 7.8 Graf vztahů mezi chybovými kódy neresetovatelného prvku

### 7.7.5 Rozhodovací mechanismus resetovatelných prvků

Obrázek 7.9 popisuje vzájemné vazby chybových kódů (vrcholy), kterých může nabývat resetovatelný testovatelný prvek. Význam hran a vrcholů je shodný s grafem pro neresetovatelný prvek.

- Hrany A-D realizuje detekční část dohledového systému.
- Hrany 3a, 3b, 4a, 4b, 5a, 5b, 6-10 realizuje akční část dohledového systému.



Obr. 7.9 Graf vztahů mezi chybovými kódy resetovatelného prvku

Význam jednotlivých hran:

- A-D: činnost je shodná jako u neresetovatelného prvku. Kapitola 7.7.4.
- 3a: byl proveden první pokus o restart prostřednictvím interface s negativním výsledkem ve smyslu předání povelu.
- 3b: byl proveden první pokus o restart prostřednictvím interface. Předání povelu proběhlo úspěšně.

- *4a*: na základě předchozího neúspěšného pokusu o restart prvku byl proveden druhý pokus. Výsledek je opět negativní.
- *4b*: druhý pokus o restart prvku byl vykonán a povel úspěšně předán vzdálenému interface. Zde je vhodné, aby administrátor analyzoval, proč došlo k předání povelu až na druhý pokus. Zda-li bylo příčinou jednorázové rušení přenosové cesty nebo se jedná o stav dlouhodobějšího charakteru.
- *5a*: poslední pokus o restart prvku selhal a bude nastaven koncový stav s chybovým kódem 10. V tomto okamžiku již nelze s prvkem provést žádnou akci, která by vedla k obnovení dostupnosti. O vzniklé situaci bude informován administrátor zasláním příslušné zprávy z tab. 7.2 na emailové adresy patřící danému prvku.
- *5b*: poslední pokus o restart prvku byl úspěšný. Zde je opět vhodné analyzovat příčinu dvojnásobného selhání při předávání povelu k restartu.
- *6*: pokud akční část dohledového systému nalezne prvek s chybovým kódem 5, znamená to, že povel k provedení restartu prvku byl vzdálenému interface řádně předán (byl potvrzen), ale síťová dostupnost prvku nebyla obnovena. Jelikož prvek nemá žádného netestovatelného neresetovatelného předchůdce, je nastaven koncový stav s chybovým kódem 10. Na informační emailové adresy bude odeslána příslušná zpráva z tab. 7.2.
- *7*: prvek má minimálně jednoho netestovatelného resetovatelného předchůdce, který byl restartován. Pokud má takových předchůdců více, restartování byli všichni.
- *8*: první pokus o restart netestovatelných resetovatelných předchůdců nezajistil obnovení síťové dostupnosti prvku. Byl proveden druhý restart těchto prvků.
- *9*: opakovaný restart netestovatelných resetovatelných předchůdců selhal. Je nastaven koncový stav s chybovým kódem 10. Na informační emailové adresy bude odeslána příslušná zpráva z tab. 7.2.
- *10*: tato hrana mezi chybovým kódem 2 a 1 řeší funkci vynuceného manuálního restartu resetovatelného prvku. Není brán ohled na dostupnost prvku. K provedení restartu dojde za každých okolností a následně pak bude nastaven chybový kód 1.

Při průchodu všemi hranami bude do záznamu událostí (logu) vložena informace se jménem prvku, aktuálním časem a provedenou akcí. Vyjímkou jsou hrany A a C.

### 7.7.6 Stavové informační zprávy

Tabulka 7.2 obsahuje seznam zpráv, které se zašlou administrátorovi nebo pověřeným osobám při průchodu přes určité hrany grafu. Označení hran je v obou grafech unikátní, tabulka je tedy platná pro resetovatelný i neresetovatelný prvek. Forma zprávy je email nebo SMS.

Tab. 7.2 Stavové zprávy zasílané administrátorovi

Hrana	Text zprávy
1	OFFLINE: <název> <dd.MM. hh:mm>
5a	OFFLINE - NEUSPESNE RESETY: <název> <dd.MM. hh:mm>
6	OFFLINE - PO USPESNEM RESETU: <název> <dd.MM. hh:mm>
9	OFFLINE - PO RESETECH PREDCH: <název> <dd.MM. hh:mm>
D	ONLINE: <název> <dd.MM. hh:mm>

Význam polí:

- <název>: název prvku s odstraněnou interpunkcí
- <dd.MM. hh:mm>: datum a čas odeslání zprávy. Př. 27.08. 13:35

## 7.8 Implementace detekční a akční části

V kapitole 7.3 je uveden požadavek na kooperaci dohledového systému se správním systémem Netadmin. Modulární koncepce dohledového systému a systému Netadmin je založena na vzájemném sdílení dat v DBMS a z tohoto pohledu je sloučení obou systémů snadné.

Implementace aplikačních modulů systému Netadmin je provedena v jazyce PERL jako sada skriptů, jejichž ústřední činností je rekonfigurace různých síťových služeb dle aktuálních dat. V současné době existuje několik dalších perspektivních skriptovacích jazyků, jejichž prostřednictvím by byla implementace detekční a akční části realizovatelná. Jsou jimi například Python a Ruby. V obou případech se jedná o jazyky s výraznou podporou objektového programování a v některých vlastnostech převyšující možnosti PERLu. Je však vhodné zachovat určitou společnou homogenitu programového řešení a z tohoto důvodu byl PERL využit i pro dohledový systém.

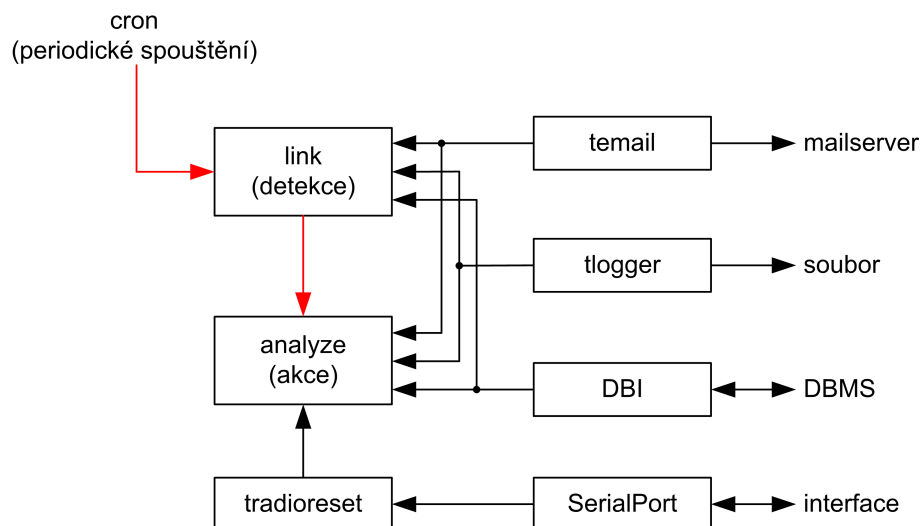
### 7.8.1 Struktura

Z důvodu přehlednosti, snadné modifikovatelnosti a opakované použitelnosti byla realizace detekční a akční části dohledového systému rozdělena do následujících souborů:

- *link*: realizuje detekční část dohledového systému.
- *analyze*: realizuje akční část dohledového systému.
- *tradioreset*: třída s metodami pro obsluhu interface.
- *tlogger*: třída pro jednoduché logování (záznam událostí) do souboru.
- *temail*: třída realizující vytvoření a odesílání zpráv na emailové adresy.

K úspěšnému spuštění skriptu *link* a *analyze* je nutné, aby měl interpret jazyka PERL k dispozici následující dvě třídy třetích stran:

- *DBI*: spojení s DBMS, zpracování SQL dotazů.
- *Device::SerialPort*: rozhraní pro komunikaci po sériové lince RS-232.



Obr. 7.10 Vzájemné vazby tříd detekční a akční části dohledového systému

Třída *DBI* je obvykle součástí základní instalace jazyka PERL ve většině distribucí OS Linux a FreeBSD. Pokud tomu tak není, je dostupná pomocí rozšiřujících instalačních balíčků zvolené distribuce nebo z archivu tříd a modulů CPAN. Třída *Device::SerialPort* v základní instalaci

nebývá a je nutné ji ve většině případů doinstalovat výhradně z archivu CPAN. Pokud má operační systém přístup k internetu, instalaci této třídy lze uskutečnit spuštěním následujícího příkazu:

```
perl -MCPAN -e 'install Device::SerialPort'
```

Na obr. 7.10 jsou znázorněny vazby mezi jednotlivými třídami detekční a akční části dohledového systému. Periodické spouštění je realizováno pomocí časového plánovače cron. V uvedeném obrázku je znázorněno šipkami červené barvy. Z nich je zřejmé, že je nejprve spuštěn skript link a až po jeho ukončení akční část analýzy. Periodicita spouštění je dána volbou exekuční periody a následnou konfigurací programu cron.

### 7.8.2 Třída *radioreset*

Motivací pro vznik této třídy byla snaha o zapouzdření metod pro komunikaci s připojeným interface. Důvodem je teoretická možnost využití více instancí této třídy pro připojení několika interface na různých sériových portech.

Pro komunikaci po sériové lince RS-232 je využita třída `Device::SerialPort` a to z důvodu, že poskytuje uživatelsky přívětivé metody pro přenos dat. Nahrazuje tak prosté využití příkazů `cat` a `echo` pro přímý přístup k souboru reprezentující sériový port v operačních systémech unixového typu. Hlavní výhodou této třídy je snadná konfigurace parametrů pomocí jednoúčelových metod a především možnost nastavení časového limitu pro příjem dat ze sériového portu. Tímto způsobem lze komunikační metody třídy *radioreset* ošetřit tak, aby respektovaly předpokládanou dobu odpovědi od interface a nezpůsobovaly zbytečnou časovou latenci v případě, že interface nekomunikuje nebo je jeho komunikace chybná.

Metody pro komunikaci s interface jsou následující:

- *ping*: ověření dostupnosti interface. Návrátová hodnota je 1 v případě, že interface zašle odpověď na příkaz `P` řetězec „OK.“
- *dreset*: uskuteční restart prvku prostřednictvím lokálního interface. Argumenty jsou čas a výstupní port. Hodnota 1 bude vrácena, pokud interface odešle metodě řetězec „GM-2 Direct reset accepted.“
- *remotereset*: realizuje restart vzdáleného prvku. Argumenty jsou čas, výstupní port a adresa. Návrátová hodnota 1 bude vrácena za předpokladu, že interface odešle metodě odpověď začínající podřetězcem „GM-“. Pokud tato metoda přijme od interface chybovou zprávu začínající podřetězcem „EM-“, bude okamžitě ukončena s návratovou hodnotou 0 a nebude čekat na uplynutí časového limitu.
- *rptrreset*: k přenosu povelu využije funkci opakovače. Funkčně je shodná s metodou *remotereset*. Mezi argumenty patří navíc adresa opakovače.

## 8 Uživatelské rozhraní dohledového systému

Koncepce dohledového systému je zvolena tak, aby všechny jeho části mohly pracovat zcela nezávisle na sobě. Jediným styčným bodem, který je navzájem spojuje ve funkční celek, jsou společná data v DBMS. Činnost uživatelského rozhraní je omezena výhradně na manipulaci s těmito daty.

Na základě uvedených vlastností lze realizaci uživatelského rozhraní pojmout několika způsoby. Je možné vytvořit klasickou GUI aplikaci (tlustý klient) s kompletní vnitřní logikou, která se k DBMS prostřednictvím sítě připojí a umožní tak správu dostupných dat. Výhodou je vysoká míra interaktivity, nevýhodou závislost na platformě (PC, PDA, mobilní telefon) a decentralizace různých verzí na různých místech. Vzhledem k síťově orientované povaze dohledového systému a absence jakýchkoliv činností prováděných offline, je forma tlustého klienta z provozního hlediska výrazně komplikovanější.

Tenký klient představuje způsob, jímž lze vnitřní logiku uživatelského rozhraní odstranit z lokální aplikace a přenést ji na centrální distribuční místo – server. V současnosti je pojem tenký klient spojován převážně s formou webových aplikací. Pro realizaci uživatelského rozhraní je tento způsob výhodnější a byl proto zvolen.

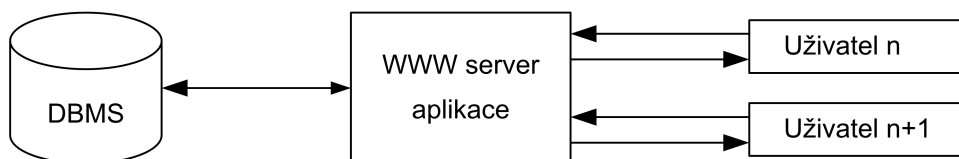
Charakteristika uživatelského rozhraní:

- Zobrazení kompletního stromu prvků včetně chybového kódu.
- Barevné odlišení prvku pro dostupný a nedostupný stav.
- Administrace všech parametrů prvku.
- Export stromu prvků do formátu PDF.
- Přehledné zobrazení v minimalistických prohlížečích (textové a mobilní).
- Využití šablony uživatelského rozhraní systému Netadmin.

### 8.1 Vlastnosti WWW rozhraní

#### 8.1.1 Centralizace zdrojového kódu aplikace

Možnost umístění zdrojového kódu odděleně od klienta představuje zřetelnou výhodu především z pohledu vývoje a údržby aplikace. Mezi uživatele není nutné šířit nové nebo opravené verze, ale úpravy provedené na centrálním místě (zpravidla WWW server) jsou k dispozici všem uživatelům ihned bez zbytečného prodlení. Přístup k takové aplikaci je pak v zásadě možný z jakéhokoliv počítače, který disponuje prohlížečem WWW stránek.



Obr. 8.1 Princip činnosti webové aplikace

#### 8.1.2 Nestavovost a interaktivita

Nestavovost je klíčovou vlastností webových aplikací – nikde neexistuje žádný začátek ani konec aplikace. Z tohoto důvodu musí být všechny logické celky aplikace rozděleny do funkčních kroků, které musí být schopny provozu na základě předaných parametrů, identifikačního řetězce dané relace (session.id) nebo cookies uložených v prohlížeči uživatele. Důležitým faktorem spolehlivé aplikace je ošetření vstupních parametrů proti zneužití a eliminace následků nežádoucích projevů, které vzniknout například kliknutím na tlačítko krok zpět nebo obnovit v prohlížeči.

Omezená interaktivita webové aplikace je daná samotnou technologií. Způsobuje nemožnost pružné a okamžité reakce na změnu stavu, který na straně serveru nastane. Týká se to především změn hodnot v DB a jejich následné prezentace uživateli. Webová stránka se může zobrazit pouze na pokyn uživatele (kliknutí na odkaz) nebo po uplynutí stanoveného časového intervalu (metatag, JavaScript). V obou případech se jedná o podnět k obnově obsahu ze strany klienta, nikoliv zdroje dat. Periodická aktualizace stránky je tak nutností v případě, že požadavkem aplikace je udržet aktuální stav dat na obrazovce uživatele. Důsledkem této aktivity je zvýšená zátěž serveru a je proto nutné stanovit aktualizací interval vzhledem k povaze problému.

V současné době je mezi tvůrci webových aplikací velmi populární technologie se souhrnným názvem AJAX. Jedná se o kombinaci JavaScriptu a objektu XMLHttpRequest. Zásadní výhodou této technologie je možnost zpracování http požadavků na pozadí zobrazené stránky. Její struktura se nemusí načítat celá, zůstane stále vykreslena. Aktualizují se pouze ty elementy, které jsou pro popis situace důležité. Stránka pak působí dojmem interaktivní desktopové aplikace, přestože v pozadí stále dochází k periodickým požadavkům na server. Ty však mají značně snížené nároky na systémové zdroje serveru. Interaktivita je v tomto případě pouze iluzorní.

Moderní prohlížeče AJAX podporují. V reálném provozu dohledového systému ovšem může nastat situace, kdy bude nutné k uživatelskému rozhraní přistupovat pomocí prohlížečů s omezenou funkcionalitou. Jsou jimi například konzolové textové prohlížeče, minimalistické verze v PDA nebo mobilních telefonech. Pro návrh uživatelského rozhraní byl proto zachován konzervativní přístup a využity pouze základní vlastnosti prohlížeče.

### 8.1.3 WWW Server

Nasazení administračního rozhraní do reálného provozu předpokládá využití šablony uživatelského rozhraní správního systému Netadmin. V rámci této šablony je již vyřešeno formátování vzhledu výsledných stránek, optimalizace pro textové prohlížeče a autentizace uživatelů.

V současné době provozuje sdružení Starhill z důvodu zabezpečení dva webové servery – uživatelský a administrační. Pro přístup k administrační části je požadováno zabezpečené spojení pomocí SSL s příslušným certifikátem a vlastnictví uživatelského jména a hesla. SSL je možno v případě nouzového přístupu vypnout a zajistit tím správnou funkčnost starších textových prohlížečů. Realizace obou těchto serverů je provedena s využitím dvou nezávislých instancí http serveru Apache. V obou instancích je povolena možnost využití skriptovacích jazyků PHP, PERL, Python a C# prostřednictvím technologie ASP.NET (mod\_mono).

Šablona uživatelského rozhraní systému Netadmin je vytvořena v jazyce PHP, a přestože je prakticky možné kombinovat v rámci webových aplikací více jazyků, uživatelské rozhraní dohledového systému využívá rovněž PHP.

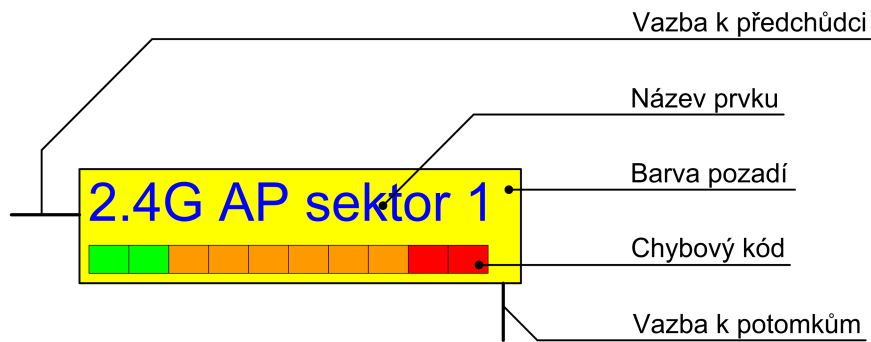
## 8.2 Grafická reprezentace prvku

Důležitým aspektem při zobrazení stromu prvků je jejich grafické ztvárnění a nalezení optimálního poměru mezi estetickým a funkčním návrhem. Na základě tohoto požadavku a s přihlédnutím ke specifikům webového rozhraní byl navrhnout vzhled prvku, který je uveden na obr. 8.2.

Mezi vlastnosti navrženého vzhledu patří:

- Malé rozměry. Jsou uvedeny pouze základní údaje o prvku.
- Chybový kód je zobrazen ve stylu barevného indikačního proužku.
- Barva pozadí je závislá na dostupnosti nebo nedostupnosti prvku.
- Vazby k ostatním prvkům jsou umístěny ve fixních polohách.





Obr. 8.2 Grafická reprezentace prvku.

### 8.2.1 Chybový kód

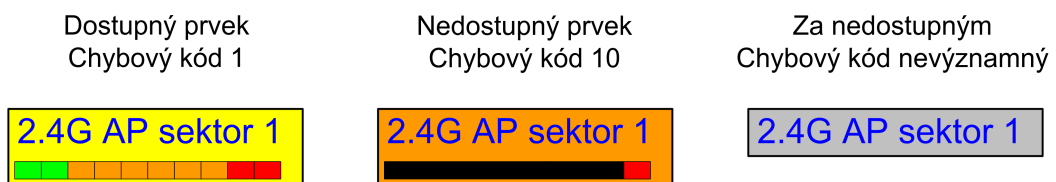
Znázornění chybového kódu prostřednictvím barevného indikačního proužku bylo vytvořeno z důvodu snazší orientace administrátora při pohledu na objekt prvku. Usnadnění spočívá v tom, že administrátor nemusí znát přesný význam numerické reprezentace chybového kódu, ale dle barev zobrazených polí je schopen určit, v jakém stavu se prvek nachází a jaká je orientační pravděpodobnost, že se dohledovému systému podaří dostupnost prvku obnovit. Využita je přitom vlastnost rostoucí hodnoty chybového kódu popsána v kapitole 7.7.6.

Na obr. 8.2 je znázorněn plně dostupný prvek s chybovým kódem 1. Všechna pole jsou zobrazena ve svých základních barvách – zelená, oranžová a červená. Pokud dojde u tohoto prvku k inkrementaci chybového kódu, první pole z levé strany, které bylo doposud vyplněno svou základní barvou, bude černé. Ukázka prvku s chybovým kódem 10 je na obr. 8.3.

Zástupný text proužku je získán z databázové tabulky `node_erc` a v prohlížeči se zobrazí zpravidla po najetí myši nad jeho libovolnou část. Administrátor je tak schopen velmi rychle zjistit, pokud má zájem, co přesně se prvkem děje. Přehled zástupných textů je uveden v tab. 7.1.

### 8.2.2 Barva pozadí

Při řešení výpadků v reálném provozu sítě je třeba, aby administrátor získal pohledem na strom prvků okamžitý přehled o tom, který prvek je dostupný a který nikoliv. Tuto informaci lze sice odvodit na základě chybového kódu, ale při větším počtu zobrazených prvků je forma barevného proužku nepřehledná. Z tohoto důvodu byla signalizace nedostupnosti realizována změnou barvy pozadí ze žluté na oranžovou, resp. šedou, pokud se prvek nachází mezi následníky za nedostupným prvkem. Na obr. 8.3 jsou uvedeny všechny varianty zobrazení prvku.



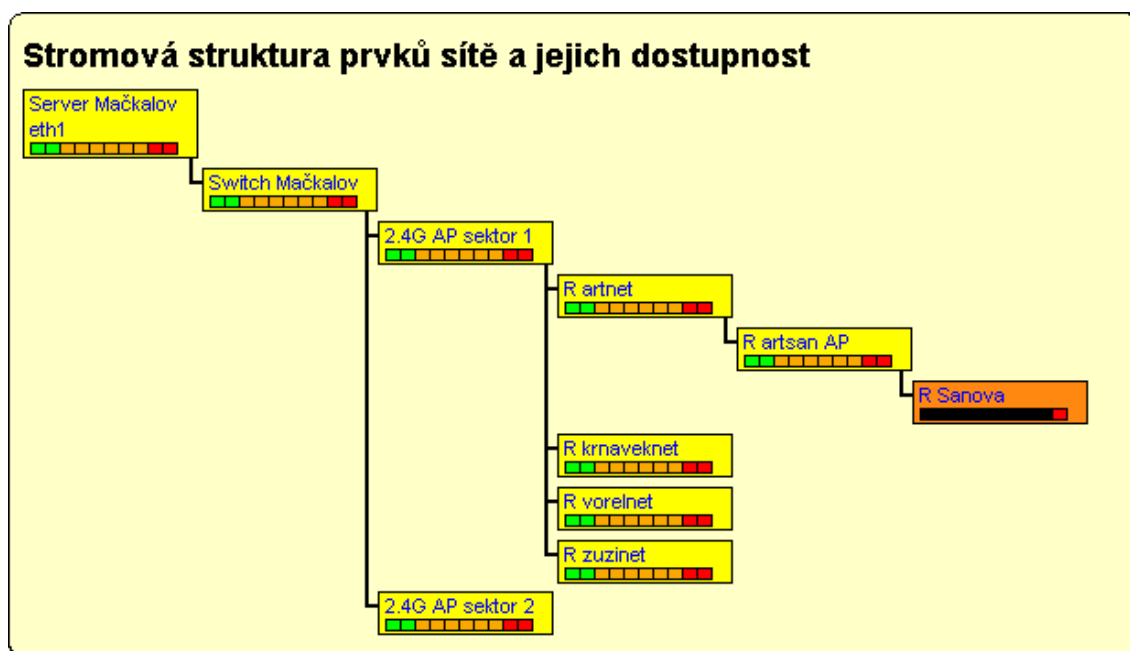
Obr. 8.3 Varianty zobrazení prvku dle jeho dostupnosti.

## 8.3 Stromová struktura prvků

Grafická reprezentace prvků a jejich vzájemných vazeb je základní vizuální forma, která je administrátorovi nabídnuta. Umožňuje získat snadný přehled o stavu sítě a současně velmi rychle odvodit závislosti mezi jednotlivými prvky. Na obr. 8.4 je uveden příklad stromové struktury

několika prvků z celkových 147, které jsou v rámci sítě Starhill provozovány. Kompletní graf je z prostorových důvodů umístěn v příloze D.

Vykreslení struktury je realizováno rekurzivní funkcí využívající seznamu prvků v databázové tabulce `node`. Pokud je nalezen prvek, který má nastaven atribut `erc` na hodnotu 0, je vykreslování zbytku větve přerušeno. Dohledový systém tento zbytek ignoruje a je zbytečné jej zobrazovat. Během prohledávání je současně u každého prvku analyzován chybový kód a na základě jeho hodnoty jsou nastaveny vlastnosti zobrazení prvku, případně celé následující větve.



Obr. 8.4 Ukázka vykreslení stromu prvků ve webové aplikaci.

Zobrazený objekt prvku obsahuje pouze omezené množství podstatných informací. Z tohoto důvodu byl do textu s jeho názvem vložen hypertextový odkaz, který směřuje na ucelený přehled všech jeho parametrů. Ukázka pro prvek „2.4G AP sektor 1“ je na obr. 8.6.

Pokud je prvek testovatelný a současně konfigurovatelný prostřednictvím vlastního WWW rozhraní, lze kliknutím na zobrazenou IP adresu do tohoto prostředí vstoupit. Podmínkou je funkční síťové spojení mezi klientským počítačem a prvkem. Kliknutím na ikonu „EDIT“ lze upravit příslušné parametry prvku ve vztahu k dohledovému systému.

## 8.4 Administrace prvků

Účelem tabulky na obr. 8.5 je zobrazit prvky bez jejich vzájemných vazeb, ale s možností jejich editace. Administrátor zde může nové prvky přidávat, upravovat nebo mazat. Tato funkcionality není v režimu zobrazení stromového grafu dostupná.

Sloupec *stav* obsahuje aktuální chybový kód prvku. Barva pozadí pole je přitom závislá na jeho hodnotě podobně jako u grafické reprezentace prvku. Sloupec *aktivní* obsahuje text ano, pokud je atribut `enable` databázové tabulky `node` nastaven na hodnotu 1. Následující 3 sloupce seznamu prvků popisují, zda-li je prvek testovatelný, resetovatelný a informovatelný. Kliknutím na ikonu „EDIT“ je zpřístupněna možnost změny požadovaného parametru. Kliknutím na název prvku se zobrazí ucelený přehled jeho parametrů, stejně jako je tomu u stromu prvků.

**Administrace prvků dohledového systému**

#	Název prvku	Stav	Aktivní	IP Test	Reset	Email					
1	R artnet	1	ano	ano	EDIT	ano	EDIT	ano	EDIT	upravit	smazat
2	R artsan AP	1	ano	ano	EDIT	ano	EDIT	ne	EDIT	upravit	smazat
3	R knaveknet	1	ano	ano	EDIT	ne	EDIT	ano	EDIT	upravit	smazat
4	R Sanova	10	ano	ano	EDIT	ne	EDIT	ano	EDIT	upravit	smazat
5	R vorelnet	1	ano	ano	EDIT	ne	EDIT	ano	EDIT	upravit	smazat
6	R zuzinet	1	ano	ano	EDIT	ne	EDIT	ano	EDIT	upravit	smazat
7	Server Mačkalov eth1	1	ano	ano	EDIT	ne	EDIT	ano	EDIT	upravit	smazat
8	Switch Mačkalov	1	ano	ne	EDIT	ano	EDIT	ano	EDIT	upravit	smazat
9	2.4G AP sektor 1	1	ano	ano	EDIT	ano	EDIT	ano	EDIT	upravit	smazat
10	2.4G AP sektor 2	1	ano	ano	EDIT	ano	EDIT	ano	EDIT	upravit	smazat

Vložit nový prvek

Obr. 8.5 Ukázka seznamu prvků.

**Parametry prvku**

Název prvku: **2.4G AP sektor 1**  
Aktivní: **ano**  
Chybový kód: **1**  
Slovní stav: **Všechno je OK**  
Jméno předka: **Switch Mačkalov**  
Popis: 2.4GHz AP na Mačkalově natočený směrem do centra města.  
Login: [zobrazit](#)  
Kanál: **8**

EDIT

**IP test**

Testovatelný na tyto IP adresy:

#	IP adresa
1	192.168.91.1
2	192.168.100.11

EDIT

**Reset**

Resetovatelný s těmito parametry:

#	Typ resetu	Adresa	Repeater	Výstup	Čas [s]
1	P	745		1	30 sec

EDIT

**Stavové emaily**

Informovatelný na tyto emaily:

#	Email
1	standa@starhill.org

EDIT

Obr. 8.6 Přehled parametrů prvku.

## 8.5 Export do PDF

Výstup seznamu prvků do formátu PDF je vytvořen zejména pro servisní a dokumentační účely. Administrátor si může tento seznam vytisknout a při své činnosti mít k dispozici podklady s aktuálním rozložením prvků v síti. Vzájemné vazby mezi nimi jsou naznačeny odsazením z levé strany. V pravé části dokumentu je místo pro případné poznámky, které mohou být následně zaneseny do dohledového nebo správního systému.

Pro generování PDF je využita volně dostupná open-source knihovna fpdf [23].

---

**Server Mačkalov eth1 | login: /**  
IP: 20.1 100.1 115.1 157.1 | Kanál:

---

**Switch Mačkalov | login: /**  
IP: | Kanál:

---

**2.4G AP sektor 1 | login: /**  
IP: 91.1 100.11 | Kanál: 8

---

**R artnet | login: /**  
IP: 111.1 | Kanál:

---

**R artsan AP | login: /**  
IP: 111.254 | Kanál:

---

**R Sanova | login: /**  
IP: 51.1 111.253 | Kanál:

---

**R krnaveknet | login: /**  
IP: 103.1 | Kanál:

---

**R vorelnet | login: /**  
IP: 116.1 | Kanál:

---

**R zuzinet | login: /**  
IP: 117.1 | Kanál:

---

**2.4G AP sektor 2 | login: /**  
IP: 92.1 100.12 | Kanál: 6

Obr. 8.7 Příklad struktury PDF dokumentu.

## 9 Závěr

Cílem této diplomové práce bylo zvýšit efektivitu správy bezdrátové sítě občanského sdružení Starhill. Jak již bylo naznačeno v úvodu, koncepce práce byla rozdělena do dvou logických celků. Oba tyto celky se podařilo úspěšně navrhnout, realizovat a nasadit do reálného provozu. V současné době je navíc 8 ks interface úspěšně využíváno v podobné bezdrátové síti v Humenném.

Interface byl realizován na jednovrstvé desce plošných spojů a využívá součástek, které lze běžně zakoupit v maloobchodní síti v České republice. Nebyly využity žádné komponenty, které by vyžadovaly speciální nástroje nebo postupy. Kompletaci interface je schopen zvládnout kdokoli se základními znalostmi elektrotechniky a průměrnou zručností.

Vývoj a testování interface probíhal přímo v reálném prostředí sítě Starhill. Volba tohoto způsobu práce poukázala na mnoho nedostatků již v samém počátku vývoje a umožnila tak velmi rychlé vytvoření programového vybavení pro interface bez fatálních chyb. Finální verze je výsledkem dlouhodobého testování od března 2007 do listopadu 2007. Během této doby byla navržena deska plošných spojů, uvedená v příloze A. Od prosince 2007 je interface, společně s radiostanicí, nasazen v rutinním provozu.

Elementární dohledový systém oznamující dostupnost nebo nedostupnost libovolného síťového zařízení je ve správním systému sítě implementován od roku 2005. Jednalo se však o verzi se špatně navrženou strukturou uložení informací o síťových prvcích. Nový dohledový systém, popsáný v této diplomové práci, využívá z původní verze pouze seznam prvků s jejich atributy. V provozu je od května 2007. Je nutné podotknout, že v dohledovém systému stále existuje prostor pro další inovaci. Změny budou probíhat dle potřeby.

Vzhledem ke krátké době provozu realizovaného systému je velmi obtížné hodnotit celkový přínos nějakou exaktní formou. Pokus o vytvoření statistického přehledu je, vzhledem k obecně nízké četnosti výpadků, značně neobjektivní. Období vývoje bylo specifické simulovanými výpadky a případně chybami ve firmware interface. Ale i v tomto přechodném období byla funkce navrženého systému využita při řešení skutečných problémů. V prvním případě šlo o fyzickou poruchu hardware přístupového bodu, kde byl nutný osobní zásah administrátora. Ve zbylých čtyřech byla činnost síťových prvků automaticky obnovena dálkovým přenosem povelu k restartu. Je nutné zmínit, že orientačně zjištěné přímé náklady na urgentní zásahy administrátora by nyní tvořily cca 45% pořizovací ceny 6-ti ks interface včetně radiostanic. Krátkodobá tendence vedoucí k úspoře provozních nákladů je tedy zřejmá.



## 10 Literatura

- [1] MOMJIAN, B. *PostgreSQL – Praktický průvodce*. Brno: Computer Press, 2003. ISBN 80-722-6954-2
- [2] HEROUT, P. *Učebnice jazyka C*. 4. přepracované vydání. České Budějovice: Kopp, 2005. ISBN 80-7232-220-6
- [3] SATRAPA P. *Perl pro zelenáče*. Praha: Neocortex, 2000. ISBN: 80-86330-02-8
- [4] VÁŇA, V. *Mikrokontroléry ATMEL AVR - popis procesoru a instrukční soubor*. 1. vydání. Praha: BEN - technická literatura, 2003. ISBN: 80-7300-083-0
- [5] VÁŇA, V. *Mikrokontroléry ATMEL AVR - programování v jazyce C*. 1. vydání. Praha: BEN - technická literatura, 2003. ISBN: 80-7300-102-0
- [6] MANN, B. *C pro mikrokontroléry*. 1. české vydání. Praha: BEN - technická literatura, 2004. ISBN: 80-7300-077-6
- [7] KABELOVÁ, A.; DOSTÁLEK, L. *Velký průvodce protokoly TCP/IP a systémem DNS*. Brno: Computer Press, 2003. ISBN 80-7226-675-6
- [8] VIRIUS, M. *C# - Hotová řešení*. Brno: Computer Press, 2006. ISBN 80-251-1084-2
- [9] MATUSZCZYK, J. *Antény prakticky*. Praha: BEN - technická literatura, 2002. ISBN 80-7300-178-0
- [10] NEVŘIVA, P. *Analýza signálů a soustav*. Praha: BEN - technická literatura, 2002. ISBN 80-7300-004-0
- [11] MATUŠKA, J. *IS nekomerční Wi-Fi sítě*. Brno, 2005. Bakalářská práce. UAI FSI VUT v Brně.
- [12] ATMEL Co. *ATmega8 dokumentace* [online]. [cit. 7.11.2006]. Dostupné z: <[http://www.atmel.com/dyn/resources/prod\\_documents/doc2486.pdf](http://www.atmel.com/dyn/resources/prod_documents/doc2486.pdf)>
- [13] ZARLINK Co. *MT8870 dokumentace* [online]. [cit. 7.11.2006]. Dostupné z: <[http://assets.zarlink.com/DS/zarlink\\_MT8870D\\_OCT\\_06.pdf](http://assets.zarlink.com/DS/zarlink_MT8870D_OCT_06.pdf)>
- [14] RONEŠOVÁ, A. *Rychlá Fourierova transformace na AVR* [online]. [cit. 15.11.2006]. Dostupné z: <<http://home.zcu.cz/~ronesova/index.php?menuitem=avrfft>>
- [15] SIWY, R. *Generation and Recognition of DTMF Signals* [online]. 5.10.1997 [cit. 15.11.2006]. Dostupné z: <<http://focus.ti.com/lit/an/slaae16/slaae16.pdf>>
- [16] BANKS, K. *The Goertzel Algorithm* [online]. 28.8.2002 [cit. 15.11.2006]. Dostupné z: <<http://www.embedded.com/story/OEG20020819S0057>>
- [17] AVR Freaks. Sdružení vývojářů pro AVR mikrokontroléry [online]. Dostupné z: <<http://www.avrfreaks.net/>>
- [18] WinAVR. Webová stránka projektu [online]. Dostupné z: <<http://winavr.sourceforge.net/>>
- [19] PostgreSQL. Webová stránka projektu [online]. Dostupné z: <<http://www.postgresql.org/>>
- [20] PHP: Hypertext Preprocessor. Dokumentace k PHP [online]. Dostupné z: <<http://www.php.net/>>

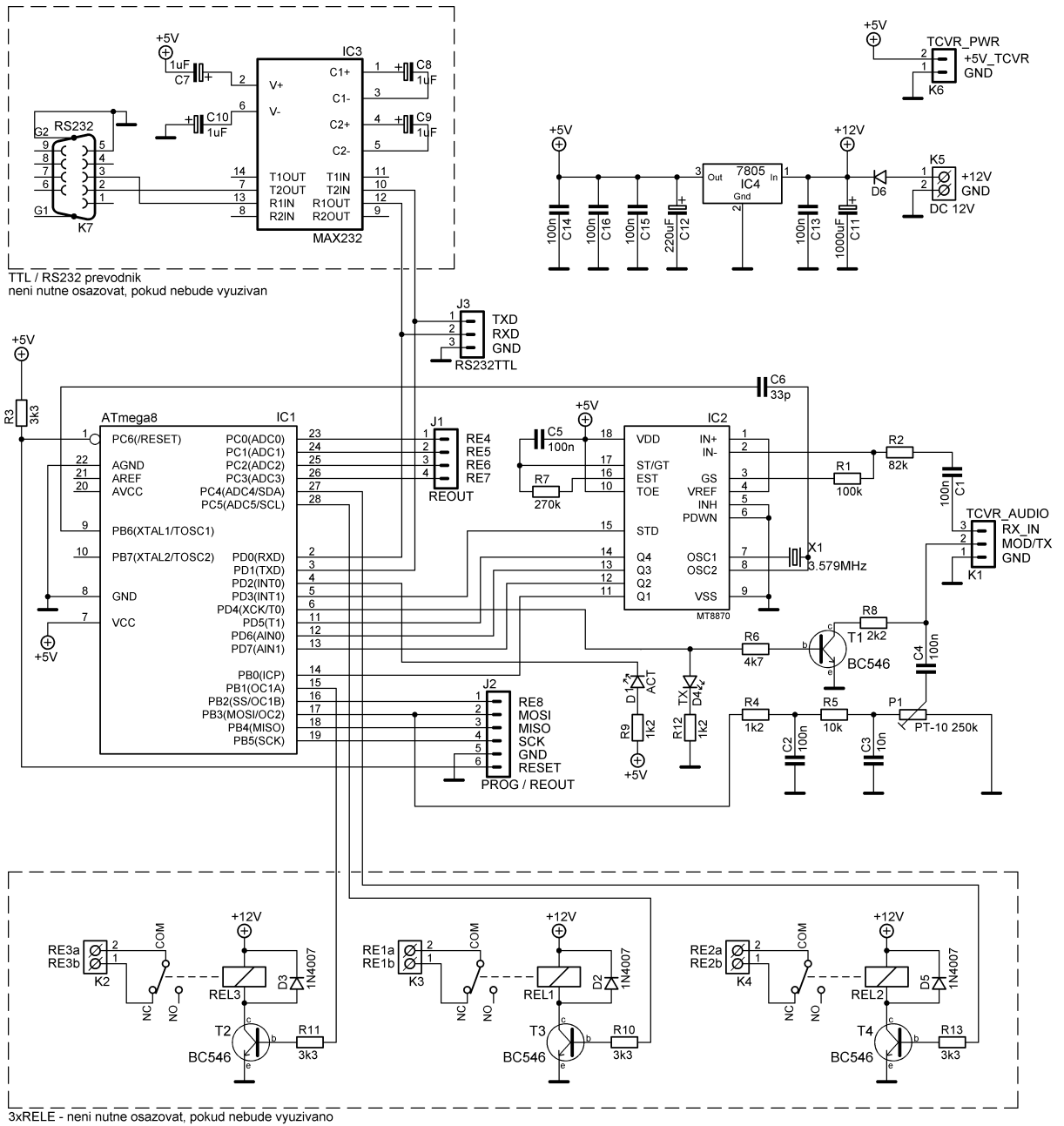
- [21] CPAN: Comprehensive Perl Archive Network [online].  
Dostupné z: <<http://www.cpan.org/>>
- [22] Starhill Network. Webové stránky sítě Starhill [online].  
Dostupné z: <<http://www.starhill.org/>>
- [23] FPDF Library. Webová stránka projektu [online].  
Dostupné z: <<http://www.fpdf.org/>>



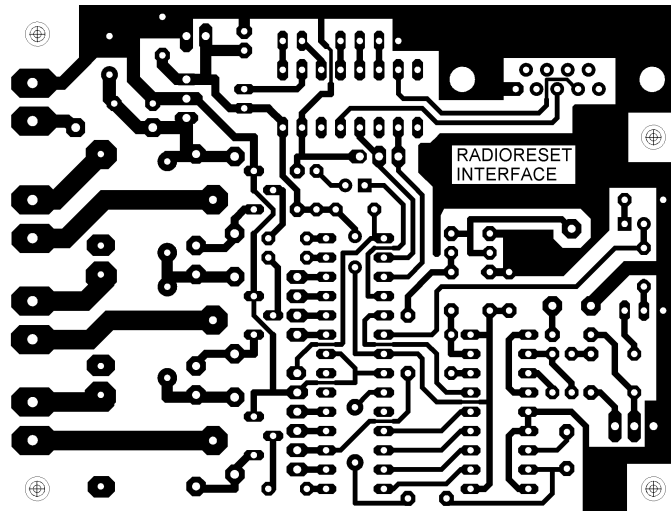
## **Přílohy**



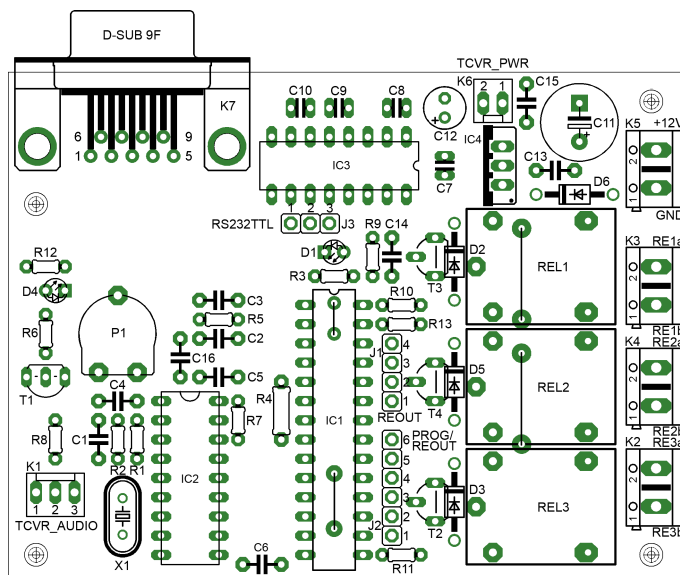
## A Schéma zapojení a deska plošných spojů



Obr. A.1 Schéma zapojení interface



Obr. A.2 Strana spojů. Rozměry 90 x 67 mm. Měřítko 1:1.



Obr. A.3 Strana součástek. Měřítko 1:1.

## B Seznam součástek

Tab. B.1 Seznam součástek hlavní části interface

Prvek	Hodnota nebo typ	Pouzdro
IC1	Atmel ATmega8	PDIP
IC2	MT8870	PDIP
IC4	7805	TO220
T1	BC546	TO92
D1	LED 3mm zelená	
D4	LED 3mm červená	
D6	1N4007	DO41
R4, R9, R12	1k2	R0204
R8	2k2	R0204
R3	3k3	R0204
R6	4k7	R0204
R5	10k	R0204
R2	82k	R0204
R1	100k	R0204
R7	270k	R0204
P1	250k / PT10VK250	
C6	33pF keramický	RM5
C3	10nF keramický	RM5
C1, C2, C4, C5, C13, C14, C15, C16	100n keramický	RM5
C11	1000 $\mu$ F/6V elektrolytický	RM5
C12	220 $\mu$ F/10V elektrolytický	RM5
X1	3.579545 MHz	HC-49U
J1	S1G04 jumper lišta	
J3	S1G06 jumper lišta	
K5	ARK300/2	
K1	PSH02-03PG	
K6	PSH02-02PG	

Tab. B.2 Seznam součástek bloku s relé

Prvek	Hodnota nebo typ	Pouzdro
T2, T3, T4	BC546	TO92
D2, D3, D5	1N4007	DO41
R10, R11, R13	3k3	R0204
K2, K3, K4	ARK300/2	
REL1, REL2, REL3	RAS1215	

Tab. B.3 Seznam součástek TTL/RS-232 převodníku

<b>Prvek</b>	<b>Hodnota nebo typ</b>	<b>Pouzdro</b>
IC3	MAX232	PDIP
C7, C8, C9, C10	1 $\mu$ F/35V tantalový	
K6	D-Sub 9F, 90° do DPS	

## C Vybrané texty konfiguračního prostředí

### C.1 Nápověda

K výpisu nápovědy dojde zasláním příkazu H připojenému interface.

Available commands:

```
H          -this help page
P          -online test
A          -show address of this device
Nxxx      -set new address, xxx is numeric value in interval 000-999
Rxxx      -read code form code table. xxx is numeric index in interval 001-200
Wxxxxyyy  -write code to code table. xxx is index 001-200, yyyy is code 0000-9999
Cxxxvs    -reset remote device with address xxx, output v and time s in 10 sec steps
Dyyyxxxvs -reset remote device yyy via repeater xxx, output v and time s in 10sec steps
Xvs       -direct reset, v is output and s time in 10sec steps
I         -show delays setup TXD RXD RPT TOT SPT
Kttttrraaaddsss -set delays ttt=TX rrr=RX aaa=RPTR ddd=tone sss=space
Lxxx      -live test to address xxx
V         -on/off verbose mode
```

### C.2 Stavové zprávy

Tyto zprávy jsou zasílány na terminál během komunikace se vzdáleným interface.

```
EM-1: AddrErr remote_device=local
EM-2: AddrErr remote_device=repeater
EM-3: Code table index receiving timeout
EM-4: Acknowledgement receiving timeout

IM-1: Sending init packet to remote device
IM-2: Receiving code table index
IM-3: Code table index received
IM-4: Sending key
IM-5: Receiving acknowledgement

GM-1: Successful reset
GM-2: Direct reset accepted
```





## D Obsah příloženého DVD

Obsah jednotlivých adresářů:

- `dp_text` – obsahuje tuto DP ve formátech  $\LaTeX$  a PDF.
- `latex` – obsahuje software TexLive2005 a návod k instalaci pro OS Microsoft Windows.
- `dohled_perl` – skripty dohledového systému.
- `dohled_web` – skripty uživatelského rozhraní dohledového systému.
- `iface_pics` – 3D modely a fotografie interface.
- `iface_el` – Schéma zapojení a DPS ve formátech PNG a Eagle.
- `iface_source` – zdrojové kódy firmware interface.
- `sw_terminals` – terminálový software pro OS Microsoft Windows.
- `sw_rrconf` – konfigurační aplikace včetně zdrojových kódů.
- `topologie` – mapa s topologií sítě Starhill.