

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Fakulta elektrotechnická

DIPLOMOVÁ PRÁCE

2017

Vojtěch Roztočil

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta elektrotechnická

Katedra telekomunikační techniky

**Analogová VoIP brána s pokročilými
audio kodeky**

kveten 2017

Diplomant: Bc. Vojtěch Roztočil

Vedoucí práce: Ing. Pavel Troller CSc.

Čestné prohlášení

Prohlašuji, že jsem zadanou diplomovou práci zpracoval sám s přispěním vedoucího práce a konzultanta a používal jsem pouze literaturu v práci uvedenou. Dále prohlašuji, že nemám námitek proti půjčování nebo zveřejňování mé diplomové práce nebo její části se souhlasem katedry.

Datum:

.....

Podpis diplomanta

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Roztočil** Jméno: **Vojtěch** Osobní číslo: **392919**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra telekomunikační techniky**
Studijní program: **Komunikace, multimédia a elektronika**
Studijní obor: **Sítě elektronických komunikací**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Analogová VoIP brána s pokročilými audio kodeky

Název diplomové práce anglicky:

Analog VoIP gateway with advanced audio codecs

Pokyny pro vypracování:

Navrhněte a zkonstruuje prototyp analogové brány internetové telefonie umožňující využití pokročilých kodeků, jako je G.722, AMR Wide Band nebo OPUS. Použijte dostupných komponent jako je Raspberry Pi a destičky periferních obvodů z nějaké starší analogové pobočkové ústředny. Software řešte s využitím systému Asterisk.

Seznam doporučené literatury:

- [1] Internet: Dokumentace a zdrojové texty systému Asterisk [on-line]
- [2] Firemní dokumentace použitých hardware komponent (účastnické sady, atd.)

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Pavel Troller CSc., katedra telekomunikační techniky FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **16.02.2017**

Termín odevzdání diplomové práce: **26.05.2017**

Platnost zadání diplomové práce: **30.09.2018**

Podpis vedoucí(ho) práce

Podpis vedoucí(ho) ústavu/katedry

Podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

Datum převzetí zadání

Podpis studenta

Anotace:

Cílem této práce je návrh a následná realizace analogové brány pro VoIP, která nabízí oproti dostupným podobným řešením pokročilé širokopásmé kodeky pro přenos hlasu. Celé řešení je modulární s použitím volně dostupných komponent včetně zdrojových kódů a kodeků. Výsledkem je pak unikátní řešení analogové brány pro internetovou telefonii, které nabízí kvalitní přenos zvuku a zároveň je snadno reprodukovatelné.

Klíčová slova:

Asterisk, FXO, FXS, GPIO, Raspberry Pi, RasPBX, SIP, širokopásmé zvukové kodeky s liberální licencí, tradiční analogový telefonní systém, VoIP

Summary:

The aim of this diploma thesis is a scheme and a consequent realization of a analog VoIP gateway with enabling advanced broadband audio codecs compare to available similar solutions. The solution is modular, uses available freely components and including source codes and codecs. The result is an unique analog voice over internet protocol gateway solution that offers high-quality audio transmission and that is reproduced easily.

Index Terms:

Asterisk, free wideband audio codec, FXO, FXS, GPIO, plain ordinary telephone system, Raspberry Pi, RasPBX, SIP, VoIP

Obsah

1. Úvod.....	1
1.1 Rozbor práce	1
2. Rozhraní	2
2.1 Ethernet.....	3
2.2 FXS	3
2.3 FXO	11
2.4 GPIO	12
2.5 AUDIO.....	12
3. VoIP	14
3.1 Signalizace	14
3.1.1 Protokol H.323	14
3.1.2 Protokol SIP.....	15
3.1.3 SDP	22
3.2 Přenos médií	23
3.2.1 RTP.....	23
3.2.2 RTCP (Real Time Control Protocol).....	24
4. Kodeky	26
4.1 G.722.....	28
4.2 SILK.....	29
4.3 OPUS	30
5. Modulární koncepce.....	33
5.1 Řídící mikropočítač Raspberry Pi.....	33
5.2 Zvukové I/O zařízení	34
5.3 Karta vnitřních linek	34

5.4	Karta vnější linek	37
5.5	Propojení účastnické karty s mikropočítačem	37
6.	Programové vybavení.....	44
6.1	RasPBX.....	44
6.2	Řídící program	48
7.	Závěr.....	51
7.1	Seznam použitých zkratek	52
7.2	Seznam použitých obrázků	54
7.3	Seznam použitých tabulek	55
7.4	Seznam použité literatury:	56
8.	Přílohy	59

1. Úvod

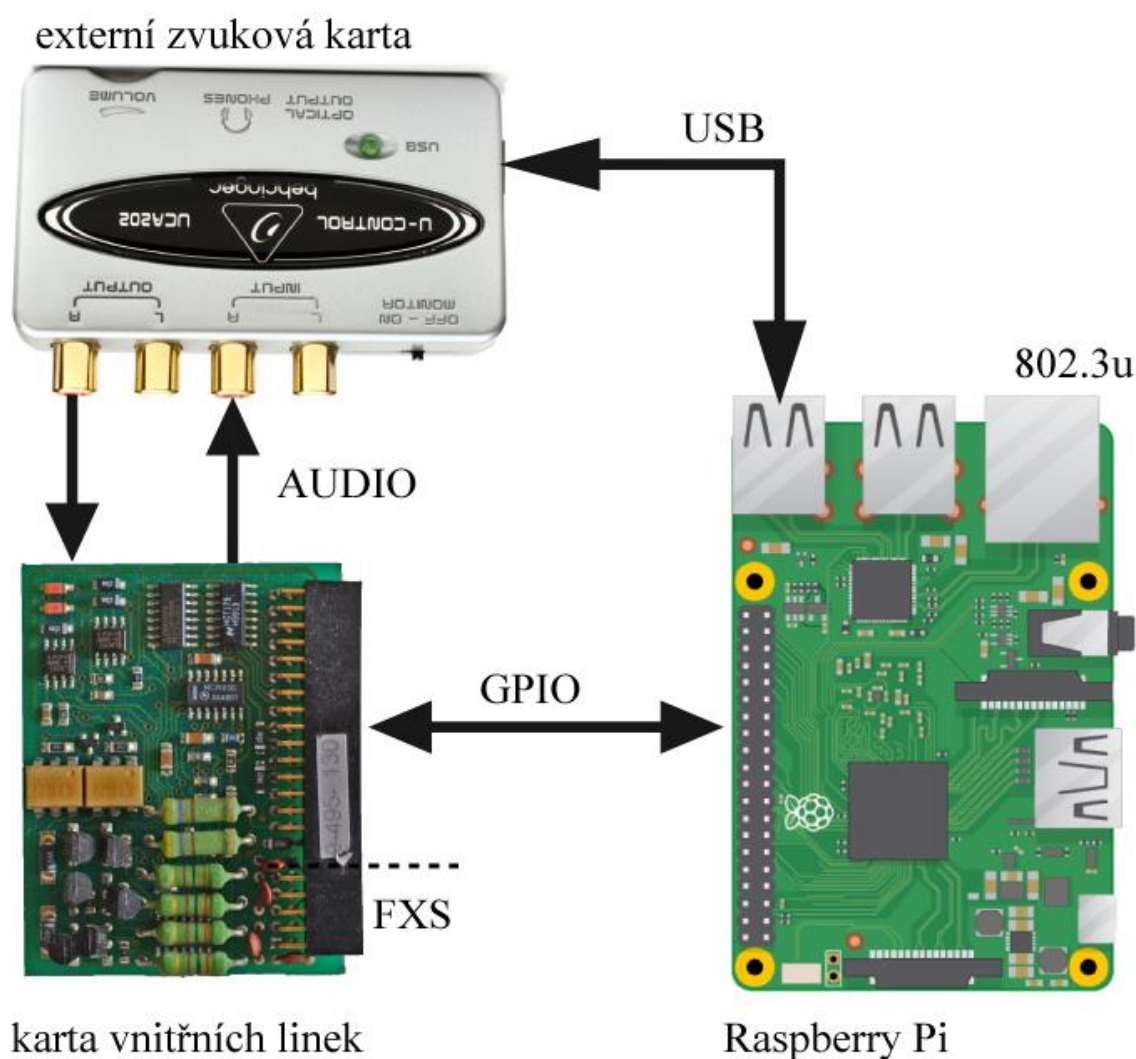
1.1 Rozbor práce

Cílem této práce je návrh a vytvoření analogové brány pro VoIP (ATA), podporující klasické analogové rozhraní (FXS) a přenos média prostřednictvím běžných nelicencovaných kodeků včetně kodeků s širším pásmem (tj. např. G.722, OPUS, SILK). Celé zařízení je modulární, sestává se z řídicího mikropočítače Raspberry, zvukového I/O zařízení (víceportová USB zvuková karta) a karty vnitřních linek z telefonní ústředny Ateus Omega 2N. Softwarové vybavení bude využívat systému s otevřeným kódem Asterisk.

2. Rozhraní

Tak jako většina telekomunikačních zařízení má i analogová brána pro VoIP několik rozhraní. V první řadě to jsou rozhraní vnější, pomocí kterých můžeme připojit bránu například do sítě internet či k analogovému telefonu.

Kromě vnějších rozhraní má ATA i rozhraní vnitřní, po kterých probíhá komunikace mezi jednotlivými moduly celého systému.



Obrázek 2.1: Rozhraní analogové brány pro VoIP

Každé rozhraní má své specifické mechanické vlastnosti (například typ konektoru), elektrické vlastnosti a protokolové vlastnosti. V dalších částech této kapitoly se budu věnovat popisu vnitřních i vnějších rozhraní, která jsou pro tuto práci klíčová.

2.1 Ethernet

Jak již zkratka VoIP napovídá, jedná se o přenos hlasu pomocí sítě založené na protokolu IP. Dle referenčního modelu ISO/OSI se IP protokol nachází na 3. (síťové) vrstvě a může být přenášén téměř libovolnou technologií splňující požadavek na minimální velikost MTU (Maximum transmission unit) 68 B. Nicméně v současné době je nejpoužívanější technologií pro přenos IP paketů Ethernet, kterým je vybaven i mikropočítač Raspberry Pi. Konkrétně se jedná o ethernetový standard s metalickým konektorem RJ-45. [1][2]

2.2 FXS

Rozhraní FXS (Foreign Exchange Station) by se zjednodušeně dalo popsat jako rozhraní pro přímé připojení analogového koncového zařízení, kterým nejčastěji bývá telefon, fax či modem.

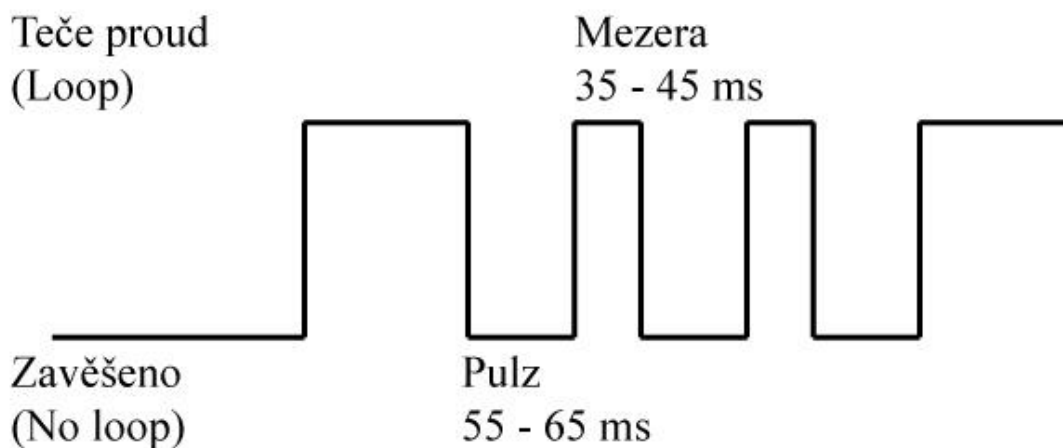
Fyzická vrstva tohoto rozhraní je metalický pár, též často označována jako a, b dráty s odporem každé žíly maximálně 800 Ω , vzájemným svodovým odporem mezi vodiči minimálně 20 k Ω a kapacitou mezi vodiči maximálně 0,5 μ F. Pro spojování vodičů se používá nejčastěji metalický konektor RJ-11.

Pokud budeme rozhraní FXS sledovat na linkové vrstvě, zjistíme, že v době klidu (IDLE), tedy ve stavu, kdy neprobíhá žádný hovor a sluchátko mikrotelefonu je zavěšené je smyčka rozpojená. To znamená, že jí neprotéká žádný proud kromě svodového a na a, b drátech je napájecí napětí ústředny. To bývá zpravidla u pobočkových ústředen v rozmezí 24 V až 48 V stejnosměrných. O trochu vyšší napětí mají veřejné ústředny, kde se počítá s úbytkem napětí na účastnickém vedení vlivem větší vzdálenosti účastníků od ústředny a bývá v rozmezí 48 V až 60 V. Avšak důležitějším a přesnějším ukazatelem než je napětí na a, b drátech je neprotékající proud skrze smyčku. Moderní ústředny používají napájení účastnické smyčky většinou zdrojem proudu přibližně 25 mA.

Ve chvíli, kdy dojde k vyzvednutí mikrotelefonu, tedy přihlášení (SEIZURE), se uzavře smyčka, na a, b drátech dojde k poklesu napětí a ústředna detekuje, že prochází proud. Obvyklé je zvýšení proudu na 15 až 60 mA po dobu delší než 200 ms. Pokud ústředna detekovala, že prochází proud, je ve stavu SEIZURE

ACKNOWLEDGE a čeká na další signál od účastníka. Tím je nejčastěji volba čísla pomocí impulzní (DEC) nebo frekvenční (DTMF) volby.

Impulzní volba čísla je generována koncovým zařízením, které číslo předává pomocí série krátkých přerušení účastnické smyčky, to je snížením proudu stejnosměrné smyčky jako při zavěšení nebo FLASH, ale s dobou přerušení kratší než jsou tyto signály. Každé číslici účastnické volby odpovídá tolik přerušení smyčky, jaká je volená číslice. Jednotlivá přerušení smyčky mají délku 55 ms až 65 ms a mezery mezi impulzy volby mají délku 35 ms až 45 ms. Jednotlivé číslice volby jsou odděleny mezičíslicovou mezerou, tedy uzavřením smyčky o délce minimálně 200 ms. Impulzní volbu je možné vysílat z telefonů s rotační i tlačítkovou číselnicí a je podporována téměř všemi telefony i ústřednami. Impulzní volba je historicky starší než tónová, vznikala pro analogové telefony s rotační číselnicí, kde dochází pohybem číselnice k mechanickému přerušování vedení. Nevýhodou je zhruba dvakrát pomalejší rychlost volby oproti DTMF a nelze ji používat pro komunikaci s hlasovými automaty. Ilustraci průběhu pulzní volby číslice tři v čase je vidět na obrázku č. 2.2.1.



Obrázek 2.2.1: Příklad pulzní volby číslice tři

S rozvojem technologie ústředen byla vyvinuta tónová nebo též kmitočtová volba, která je rychlejší než impulzní ale hlavně umožňuje přenos dat při sestaveném spojení, což pulzní volba neumožňovala. Signál vysílaný koncovým zařízením

je tvořen dvojicí kmitočtů a trvá vždy minimálně 70 ms. Mezera mezi vysíláním musí být pak alespoň 75 ms. Odchylka kmitočtů od jmenovité hodnoty může být +/- 1,8% a doba náběhu signálu maximálně 7 ms. Tabulka 2.2.1 uvádí kombinace použitých frekvencí.

	1209 Hz	1336 Hz	1477 Hz	1633 Hz
697 Hz	1	2	3	A
770 Hz	4	5	6	B
852 Hz	7	8	9	C
941 Hz	*	0	#	D

Tabulka 2.2.1: Pulzní volba

Dalším stavem na rozhraní je krátké přerušení smyčky (FLASH), které se zprvu dělalo krátkým poklepem na vidlici telefonu. Později na telefonech vzniklo pro tuto funkci speciální tlačítko FLASH. Jedná se o přerušení smyčky na dobu delší než impuls dekadické volby ale kratší než je zavěšení.

Dále jsou účastníkovi vysílány informační tóny, například po vyzvednutí mikrotelefonu ústředna reaguje vysláním oznamovacího tónu na frekvenci 425 Hz v impulzech a mezerách 330 ms / 330 ms / 660 ms / 660 ms a říká účastníkovi, že může začít s volbou čísla. Moderní telefonní ústředny umožňují nastavit různé druhy oznamovacích tónů, avšak je dobré zachovávat běžnou konvenci, na kterou jsou uživatelé již zvyklí. Pro síťově generované tóny ve veřejných sítích elektronických komunikací v ČR existují dvě varianty specifikace. První varianta specifikace tónů je založena na stávající národní specifikaci tónů. Druhá varianta specifikace tónů je, v souladu s článkem 17 Směrnice Evropského parlamentu a Rady 2002/21/EC ze dne 7. března 2002 o společném předpisovém rámci pro sítě a služby elektronických komunikací, založena na doporučení z dokumentu:

ETSI SR 002 211 V1.1.1 (2004-02) List of standards and/or specifications for electronic communications networks, services and associated facilities and services; in accordance with Article 17 of Directive 2002/21/EC.[3]

Tón	Frekvence (Hz)	Úroveň (dBm0)	Doba trvání (ms)	Poznámka
Oznamovací tón	425 ±20	-5 +2 -3	impuls 330±30 mezera 330±30 impuls 660±60 mezera 660±60	1) 13)
Oznamovací tón CENTREX	425 ±20	-5 +2	Spojité tón	2) 13)
Oznamovací tón služeb	425 ±20	-5 +2 -3	impuls 165±16 mezera 165±16 impuls 165±16 mezera 165±16 impuls 165±16 mezera 165±16 impuls 660±60 mezera 660±60	3) 13)
Okamžitý vyzváněcí tón	425 ±20	-5 +2 -3	a) impuls 1000±100 mezera > 200 a) impuls 400±40	4) 13)
Periodický vyzváněcí tón	425 ±20	-5 +2 -3	impuls 1000±100 mezera 4000±400	5) 13) 14) 15)
Obsazovací tón	425 ±20	-5 +2 -3	impuls 330±30 mezera 330±30	6) 13)
Tón neprůchodnosti	425 ±20	-5 +2 -3	impuls 165±16 mezera 165±16	7) 13)
Napojovací tón	425 ±20	-11 +2 -3	impuls 330±30 mezera 330±30 impuls 330±30 mezera 1500±150	8) 13)
Odkazovací tón	950 ±50 1400±50 1800±50	-5 +2 -3	impuls 330±70 mezera max. 30 impuls 330±70 mezera max 30 impuls 330±70 mezera 1000±250	9) 13)
Čekací tón	425 ±20	-5 +2 -3	impuls 1000±100 mezera 170±30 impuls 330±30 mezera 3500±300	10) 13)

Upozorňovací tón	425 ±20	-11 +2	impuls 330±30 mezera 9000±500	11)
		-3		13)
Konferenční tón	425 ±20	-5 +2	impuls 660±60	12)
		-3		13)

Tabulka 2.2.2: Specifikace tónů – národní varianta

Poznámky:

1. Oznamovací tón se vysílá jako první oznamovací tón účastníkům veřejných ústředen po vyzvednutí mikrotelefonu nebo po přivolání registru a jako druhý oznamovací tón na spojovací vedení od pobočkových ústředen.
2. Oznamovací tón CENTREX se vysílá jako první oznamovací tón účastníkům skupin pobočkových ústředen ve veřejných ústřednách (technika CENTREX) po vyzvednutí mikrotelefonu nebo po přivolání registru.
3. Oznamovací tón služeb se vysílá účastníkovi po vyzvednutí mikrotelefonu, má-li účastník aktivovanou službu, která přesměrovává příchozí volání. Tento tón připomíná účastníkovi, že služba je aktivována. Při některých účastnických službách se tento tón vysílá volajícímu účastníkovi po přivolání registru.
4. Okamžitý vyzváněcí tón se vysílá před připojením periodického vyzváněcího tónu. V případě a) je zajištěna minimální mezera po vyslání impulsu. Jestliže není možné zajistit minimální mezeru, je třeba použít případ b), kdy v případě spojení okamžitého vyzváněcího tónu a periodického vyzváněcího tónu je délka impulsu maximálně 1400ms.
5. Periodický vyzváněcí tón nemusí být synchronizován s periodickým vyzváněcím proudem. Periodický vyzváněcí proud se vysílá do účastnického přístroje volaného účastníka ve stejném rytmu.
6. Obsazovací tón se vysílá volajícímu účastníkovi, když je volaný účastník obsazen.
7. Tón neprůchodnosti se vysílá volajícímu účastníkovi při neprůchodnosti v síti (obsazené spojovací cesty, prošlá časová kontrola, signalizační chyba, požadovaná služba je nedostupná), tedy ve všech případech, kdy je třeba dát účastníkovi pokyn k zavěšení mikrotelefonu.

8. Napojovací tón je vysílán po celou dobu napojení meziměstské spojovatelky do hovoru mezi účastníky A a B. Tento tón se vysílá ze zařízení ústředny, která zajišťuje napojení a je používán k upozornění účastníků, že jejich hovor je připsloucháván.

9. Odkazovací tón informuje účastníka, že volané číslo je nedostupné z jiného důvodu než z důvodu obsazení nebo neprůchodnosti (např. nezařizené číslo, poruchová smyčka) a jestliže účastník žádá službu, pro kterou nemá oprávnění. Tento tón může být kombinován s odpovídající hláskou.

10. Čekací tón se vysílá volajícímu účastníkovi, který čeká na uvolnění obsazeného účastníka (volající účastník má kategorii II-2 "účastník s předností" nebo volaný účastník má aktivovanou službu "upozornění na čekající volání").

11. Upozorňovací tón se vysílá k volanému účastníkovi, který má aktivovanou službu "upozornění na čekající volání", jako informace o novém čekajícím příchozím volání.

12. Konferenční tón lze použít k informování účastníků v konferenci, že se jiný účastník připojil nebo odpojil.

13. Všechny tóny jsou vysílány bez signálu přihlášení.

14. Periodický vyzváněcí tón může být podložen hudební znělkou.

15. Úroveň periodického vyzváněcího tónu i hudební znělky při jejich současném vysílání bude stanovena dodatečně.

Druhá varianta specifikace tónů je přednostně zamýšlena pro nově zaváděné sítě a služby elektronických komunikací. Doporučuje se její postupné zavádění i do stávajících sítí jednotlivých operátorů.[3]

Tón	Frekvence (Hz ± 3%)	Doba trvání (ms)	Poznámka
Oznamovací tón (dial tone)	425	Spojité	

Oznamovací tón CENTREX (second dial tone)			1)
Oznamovací tón služeb (special dial tone)		impuls 1000 mezera 100	1)
Okamžitý vyzváněcí tón (immediate ringing tone)			1)
Periodický vyzváněcí tón (ringing tone)	425	impuls 1000 mezera 4000	
Obsazovací tón (busy tone)	425	impuls 500 mezera 500	
Tón neprůchodnosti (congestion tone)	425	impuls 500 mezera 200	
Napojovací tón (intrusion tone)			1)
Odkazovací tón (special information tone)	950 1400 1800	impuls 333 impuls 333 impuls 333 mezera 1000	
Čekací tón (caller waiting tone)			1)
Upozorňovací tón (call waiting tone)	425	impuls 200 mezera 600 impuls 200 mezera 3000	2)
Konferenční tón (conference tone)			1)

Tabulka 2.2.3: Specifikace tónů – varianta ETSI

Poznámky:

1. Dokumenty ETSI jednoznačně nespecifikují tento typ tónu.
2. Vysílá se opakovaně po dobu odměřování časové kontroly.

Důležitým signálem vysílaným z ústředny je účastnické vyzvánění, tedy střídavý signál s frekvencí kolem 50 Hz o efektivním napětí přibližně 45V. Hodnoty jsou orientační, jelikož záleží na konkrétním výrobci ústředny a na rozdíl od oznamovacích tónů se vyzváněcí tón nepřenáší sítí a není tedy standardizován. Moderní telefonní ústředny umožňují definovat různé druhy vyzvánění, které

se liší jejich časováním, v praxi je však vhodné zachovávat běžné konvence. U starších ústředen bývala proměnná délka mezi tzv. prvním a druhým vyzváněním (např. 0 až 4 s), dnes už je tato mezera konstantní, což umožňuje vložit do ní signál CLIP.

DTMF identifikace volajícího (CLIP) je signál, který pomocí tónové volby přenáší číslo volajícího do koncového zařízení. První číslice se začne vysílat 250 ms po doznění prvního vyzváněcího signálu. Následně se přenáší další číslice vždy s délkou 80 ms a mezerou 80 ms. To dává dostatečný prostor k přenosu devítimístného čísla volajícího.

FSK identifikace volajícího (tzv. CLIP)

Alternativou může být identifikace volajícího pomocí asynchronního modemového přenosu dat (tj. rychlost 1200 bit/s, 1 start bit, 8 info bitů a 1 stop bit, modulovaných dle standardu V.23 či BELL202). První číslice smí být vyslána až se zpožděním 250 ms po konci prvního vyzvánění a poslední číslice musí být vyslána nejpozději 200 ms před začátkem druhého vyzvánění. FSK značky mají délku přenosu 500 až 2000 ms. Max. počet vyslaných znaků je tedy 240, což překračuje délku čísla volajícího a lze tak navíc přidat i další informace (např. jméno volajícího). Pokud to umožní délka displeje telefonu, pak je tedy možné tímto způsobem na telefon předat z ústředny i text doplňující tzv. CLIP.

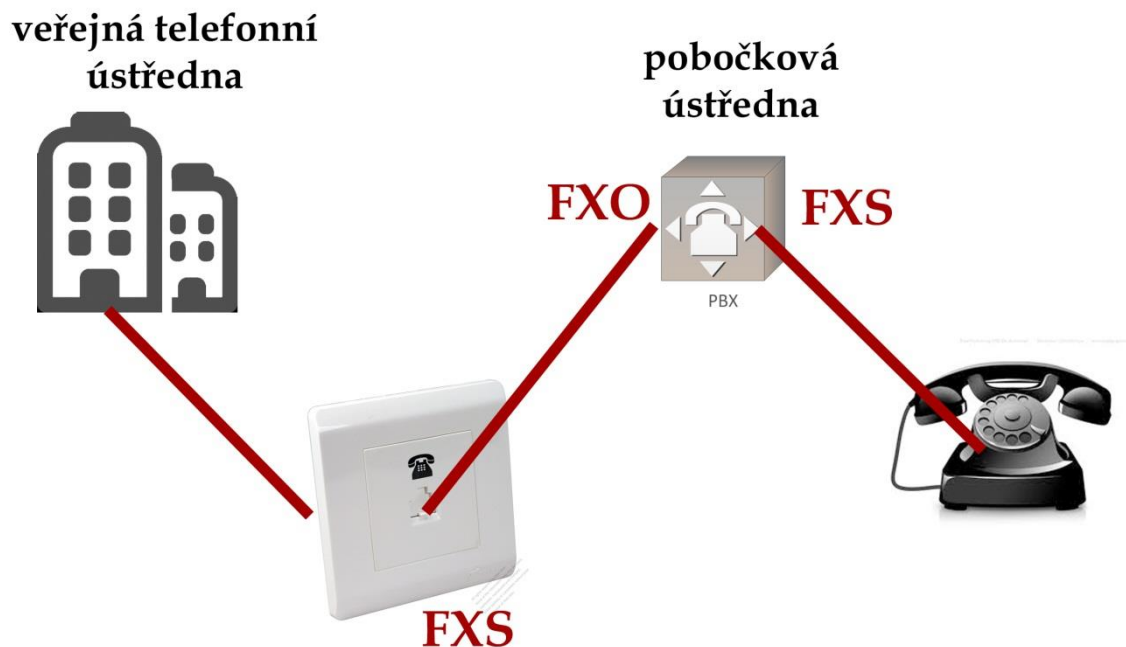
Posledním a dnes již téměř nepoužívaným signálem vysílaným z ústředny ke koncovému zařízení jsou takzvané tarifikační impulzy. Ty se přenáší pomocí impulzů délky 30 až 180 ms na frekvenci 16 kHz o napětí 22 až 44 mV. Jejich kadence odpovídá množství tarifních jednotek příslušejících konkrétnímu hovoru a shoduje se tak s počtem tarifních impulzů registrovaných na tzv. účastnickém počítadle. Jelikož se však dnes hovorné již zpravidla účtuje z tzv. CDR záznamů o jednotlivých hovorech a principy účtování hovorů mohou být součástí různých balíčků, byla tato orientační metoda signalizace ceny hovorného zrušena a už se nepoužívá.

Posledním stavem, respektive změnou stavu, je zavěšení. Tedy návrat do stavu IDLE. Signál vysílaný koncovým zařízením, který je předán pomocí rozpojení

účastnické smyčky, tj. snížením proudu stejnosměrné smyčky min. o 2 mA pod proud uzavřené smyčky na dobu delší než 200 ms. Zpravidla se stejnosměrný odpor smyčky zvýší nad 100 kohm (typicky nad 1 Mohm) a proud klesne pod 0,5 mA (typicky pod 0,05 mA). Příliš dlouhé přerušení smyčky FLASH může být již považováno za zavěšení (např. při poklepu na vidlici telefonu označovaném jako nekalibrované přerušení smyčky).[3][4][5]

2.3 FXO

Rozhraní FXO (Foreign Exchange Office) je zjednodušeně řečeno druhá strana k rozhraní FXS. V praxi to bývá nejčastěji analogový telefon, který je připojen k ústředně. Na straně ústředny je pak tedy port FXS a pomocí metalického páru popsaného v kapitole 2.2 je na druhé straně vedení připojen rozhraním FXO analogový telefon, fax či pobočková ústředna. Pro názornější pochopení může posloužit obrázek 2.3.1.



Obrázek 2.3.1: Příklad rozmístění FXO a FXS portů

Jednotlivé stavy na rozhraních FXO a FXS pak shrnuje tabulka 2.3.1.

Stav	Směr	FXO	FXS
IDLE	-	Smyčka rozpojena	Smyčka rozpojena
Přihlášení	->	Smyčka	Vyzvánění
Potvrzení přihlášení	<-	Teče proud	-
Zavěšení	-	Smyčka rozpojena	Smyčka rozpojena
Volba	->	DCT/DTMF	X
Upozornění	<-	Oznamovací tón	X

Tabulka 2.3.1: Signalizace na rozhraních FXO a FXS

2.4 GPIO

General Purpose Input/Output je vstupně / výstupní rozhraní k obecnému použití. V podstatě neexistuje žádná standardizace či doporučení, které by definovalo GPIO rozhraní. Jediný předpoklad je schopnost čtení a zápisu na port. Detailnějšímu popisu implementaci GPIO rozhraní v mikropočítači Raspberry Pi se budu věnovat v kapitole 5.4. [2]

2.5 AUDIO

Zvuková karta integrovaná na desce mikropočítače Raspberry Pi disponuje jedním výstupním audio portem s konektorem typu jack 3,5 mm. Pokud ale chceme s někým komunikovat, nestačí jen poslouchat. Musíme nějakým způsobem zajistit, aby se to, co říkáme dostalo na druhou stranu spojení. Součástí klasických telefonů je mikrotelefon, který obsahuje mikrofón a reproduktor. Pomocí hovorového transformátoru je pak signál z mikrofónu a reproduktoru transformován do a, b drátů telefonního vedení, které dopraví v našem případě hovor do karty vnitřních linek. Zde je signál transformován zpět z takzvaného dvou-drátu na čtyř-drát. S laskavou pomocí konstruktéra karty vnitřních linek jsme našli bod, z kterého lze získat vstupní audio signál a po odpájení dvou

rezistorů i bod s výstupem audio signálu. Jelikož integrovaná zvuková karta Raspberry Pi disponuje pouze výstupem audia, bylo třeba použít externí zvukovou kartu, která disponuje jak vstupním, tak výstupním audio portem a zároveň umí zpracovávat zvuk v dostatečné kvalitě.

Právě kvalita zvuku je zde zásadní parametr, jelikož výsledkem této práce má být analogová brána pro VoIP, která nabízí kvalitnější zvukové kodeky s větší šířkou pásma než komerčně nabízené brány. V této práci použita externí zvuková karta Behringer UCA202 nabízí maximální vzorkovací frekvenci 48 kHz s rozlišením převodníku 16 bitů. Vstupní konektor RCA s vstupní impedancí 27Ω a maximálním napětím 2 dBV. Výstupní konektor je též typu RCA impedancí 400Ω a maximálním napětí 2 dBV. Připojení k mikropočítači Raspberry Pi je řešeno pomocí rozhraní USB 1.1. Deklarované zpoždění zpracováním a převodem zvuku je deklarováno jako extrémně nízké s parametrem 2 pro vstup a 2 pro výstup.[16]

3. VoIP

První snahy o přenos hlasu prostřednictvím paketové sítě se objevují již v roce 1973, kdy Danny Cohen demonstroval přenos hlasu pomocí tehdejší sítě ARPANET. V roce 1990 byla první síť s přepojováním paketů ARPANET definitivně odpojena a byla nahrazena Internetem. Pokusy s přenosem hlasu pokračovaly více i méně úspěšně v celosvětové síti Internet. Důležitý milník ale nastal až v roce 1996, kdy byla mezinárodní telekomunikační unií (ITU) publikována první verze protokolu H.323 umožňující videokonference skrze síť LAN. Protokolu H.323 se rychle ujal průmysl a začal vyvíjet zařízení pro přenos hlasu nejen v místních sítích ale i sítích WAN a internetu. K dalšímu zlomu došlo až o tři roky později, kdy vyšlo doporučení RFC2543 specifikující The Session Initiation Protocol (SIP), které popisuje signalizaci a kontrolu multimediálního toku pro přenos hlasu prostřednictvím IP sítí. Protokol SIP na rozdíl od H.323 vychází z HTTP protokolu a je mu v mnoha částech velice podobný. Zatímco SIP protokol se stará o signalizaci, samotný přenos multimediálního toku je v dnešní době nejčastěji realizován pomocí Real-time Transport Protocol (RTP). A ačkoli měl protokol H.323 oproti protokolu SIP velký náskok, skupina 3GPP v roce 2000 rozhodla, že právě SIP bude sloužit jako signalizační protokol v IP Multimedia Subsystem (IMS) architektuře mobilních sítí, což přineslo protokolu SIP perspektivní výhled do budoucnosti. [6]

3.1 Signalizace

Pro přenos signalizace jsou v IP telefonii používány zejména protokoly H.323 a SIP.

3.1.1 Protokol H.323

Byl navržen odborníky na klasické telekomunikační služby z Mezinárodní telekomunikační unie (International Telecommunication Union – ITU), je velice robustní a zaměřený čistě na IP telefonii. Ve spoustě věcí je podobný ISDN

signalizaci. Jednotlivé zprávy jsou kódovány do binární podoby, čímž protokol šetří datový tok, avšak pro člověka je v případě problémů špatně čitelná. Většina logiky je obsažena v ústředně a často vyžaduje pouze podporované telefonní terminály. Pro přenos signalizace na transportní vrstvě je využíván stavový protokol TCP. H.323 je přesto dál využíván hlavně v plně profesionálních produktech velkých společností, které na jeho bázi dodávají kompletní systémy zahrnující všechny prvky telefonní sítě. H.323 je také mnohem častěji nasazován pro videokonference, protože technologie na bázi SIP má horší podporu videokonferencí. Pokud se budeme rozhodovat pro nasazení systému, který má komunikovat nejen v rámci své uzavřené sítě ale i s veřejnou sítí, je dobré brát v potaz technologii, kterou používají místní telefonní operátoři. V České Republice téměř všichni operátoři nabízejí VoIP prostřednictvím SIP, proto se nadále budu věnovat právě tomuto protokolu. [7]

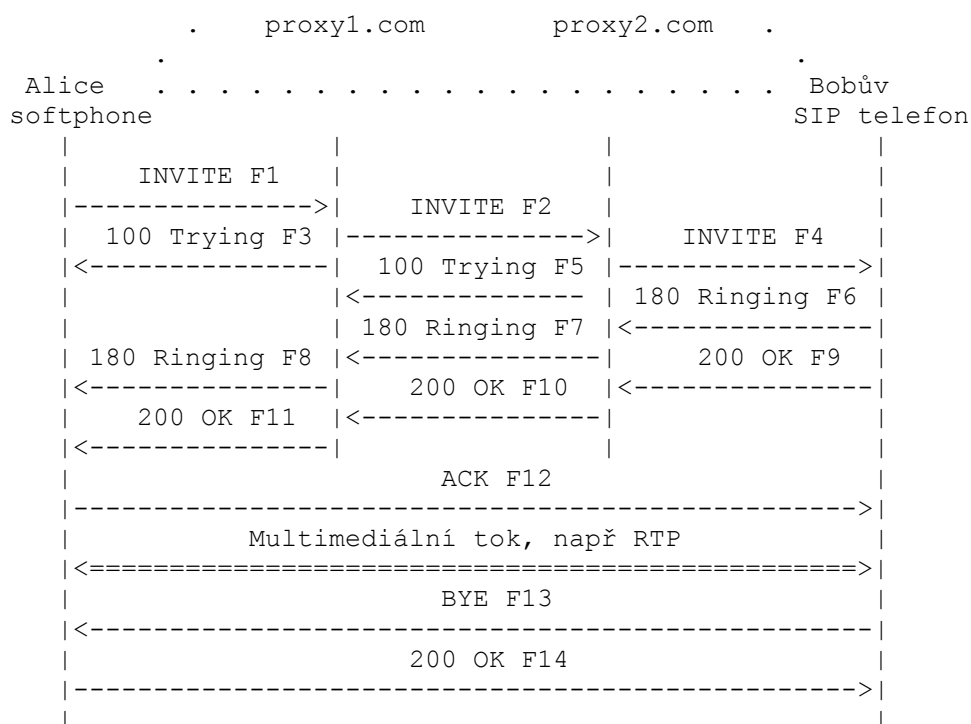
3.1.2 Protokol SIP

(Session Initiation Protocol) je popsán v doporučení IETF RFC 3261 (Internet Engineering Task Force Request For Comments) a je zaměřen na sestavování multimediálních relací, kdy telefonní hovor je považován za typ takové relace. Na rozdíl od protokolu H.323 je SIP textově orientován, snadno čitelný a není ucelenou platformou, ale jedná se v podstatě jen o signalizační protokol. Pro přenos multimediálních dat je, stejně jako u H.323, využito protokolu RTP. Ačkoliv je možné využít pro signalizaci TCP, v drtivé většině případů je využito UDP, což má příznivý efekt na zatížení sítě. [8] Pro úplnost dodejme, že SIP umožňuje také použití protokolu Transport Layer Security Protocol (TLS), který zajišťuje zabezpečení signalizačních dat.

SIP vytváří spojení typu klient – server, kde se stejně jako například u HTTP jedná o bezstavový protokol. Každý z uzlů musí být tedy vybaven jak klientskou, tak i serverovou částí. Nejdůležitější a zároveň jedinou povinnou výbavou SIP sítě jsou klientské terminály, nazývané User Agent. V tomto se tedy SIP od H.323 neodlišuje. Dalšími, tentokrát již volitelnými součástmi jsou SIP Proxy, SIP Registrar a SIP Redirect servery. Stejně jako u H.323 jsou však tyto součásti nepovinné jen virtuálně. Pokud chcete mít opravdu funkční telefonní

systém, budete je potřebovat. SIP Registrar je klíčovým prvkem funkční telefonní sítě, slouží k registraci User Agentů a jejich následné adresaci. SIP Proxy slouží k routování hovorů na základě informací od SIP Registrar. Veškerá signalizace je na SIP proxy překládána a SIP Proxy tedy není plně transparentní. SIP Redirect server slouží k podobnému účelu jako SIP Proxy, ale nepředává signalizaci, nýbrž pouze odkazuje na další uzel. V běžných podmínkách není jeho využití příliš časté.

Obrázek 3.2.1.1 zobrazuje typický příklad výměny SIP zpráv mezi Alicí a Bobem, kdy Alice používá SIP aplikaci v jejím počítači, někdy též nazývanou softphone k sestavení hovoru s Bobem. Na obrázku je též vidět dvojice SIP proxy serverů, které komunikují jménem Alice a Boba k usnadnění sestavení relace.



Obrázek 3.2.1.1: Sestavení a ukončení SIP relace

Při vytváření spojení Alice používá svoji a Bobovu SIP identitu nazvanou SIP Uniform Resource Identifier (SIP URI). SIP URI je jednotný identifikátor zdroje s definovanou strukturou, který slouží k identifikaci účastníka. Obecná syntaxe SIP URI je ve tvaru: sip:uživatel:heslo@hostitel:port;uri-parametr?hlavička,

kde uživatel je identifikátor účastníka v konkrétním systému a nejčastěji nabývá číselné hodnoty. Heslo slouží k ověření uživatelské totožnosti avšak vzhledem k bezpečnostnímu riziku je toto pole volitelné. Dvojice hostitel a port obsahuje plně specifikované doménové jméno počítače nebo přímo IPv4, případně IPv6 adresu a číslo portu, kam se má žádost odeslat.

Žádost o sestavení spojení začíná vždy na straně klienta, v našem případě u Alice, kdy její softphone pošle INVITE žádost, kterou proxy servery přepošlou až k Bobovi.

```
INVITE sip:bob@proxy2.com SIP/2.0
Via: SIP/2.0/UDP pcAlice.proxy1.com;branch=z9hG4bK776asdhds
Max-Forwards: 70
To: Bob <sip:bob@proxy2.com>
From: Alice <sip:alice@proxy1.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.proxy1.com
CSeq: 314159 INVITE
Contact: <sip:alice@pcAlice.proxy1.com>
Content-Type: application/sdp
Content-Length: 142
```

První řádek určuje název metody (INVITE), s kým se má uskutečnit spojení a verzi protokolu. V druhém řádku je adresa, na kterou očekáváme, že dorazí odpověď a branch parametr, který pomáhá identifikovat transakci. Parametr Max-Forwards slouží k eliminaci zacyklení, jeho hodnota je typu celého čísla v rozsahu 0-255 a indikuje zbývajících počet možných přesměrování proxy servery. Čtvrtý řádek a pole "To" obsahuje zobrazované jméno a v závorce SIP URI komu je žádost určena. Další řádek naopak určuje od koho je zpráva adresována. Pole tag obsahuje náhodný řetězec, který softphone přidal do URI pro další účely identifikace. Call-ID obsahuje globálně unikátní identifikátor hovoru, který je generován z pole tag a IP adresy nebo doménového jména volajícího. Command Sequence (CSeq) je celé číslo, které se s každou další žádostí o jedno zvyšuje a následuje název metody (INVITE). Pole Contact obsahuje oproti poli From SIP URI přímo na klientský počítač, tedy uživatelské jméno a plně kvalifikované doménové jméno nebo IP adresu počítače. Content-Type pak obsahuje popis těla zprávy a Content-Length udává počet bytů v textu zprávy. Podrobnosti o samotné relaci jako je například typ médií, použitý kodek nebo vzorkovací frekvence nejsou popisovány přímo pomocí SIP protokolu, ale jsou zakódované v těle SIP

zprávy pomocí jiného protokolu. Jedním z nich je Session Description Protocol (SDP) popsáný v RFC 2327.

Pokud se vrátíme zpět k sestavování hovoru, dokud Alice nezná přesnou URI adresu Boba nebo jeho SIP serveru, pošle se zpráva INVITE na vlastní SIP server. Adresu vlastního SIP serveru má Alice uloženou v softphonu. Alice SIP server slouží jako proxy server, který přijatou zprávu INVITE předává dál a zároveň odešle Alici zprávu 100 (Trying) o tom, že obdržel zprávu INVITE a snaží se ji doručit adresátovi. Ve zprávě jsou také pro kontrolu stejné parametry To, From, Call-ID, CSeq a branch. Alice SIP server proxy1 najde pomocí DNS záznamu IP adresu proxy2, na kterou odkazuje Bobova SIP URI a před přeposláním přidá do INVITE zprávy další pole s hlavičkou Via, které tentokrát obsahuje jeho vlastní adresu. Ve chvíli, kdy Bobův SIP server proxy2 obdrží zprávu INVITE, odešle zpět zprávu 100 (Trying) a vyhledá ve své databázi záznam s aktuální IP adresou Boba. Do zprávy INVITE stejně jako proxy1 přidá další pole Via s vlastní adresou a přepoše INVITE Bobovi. Bobův telefon po obdržení zprávy INVITE upozorní Boba na příchozí hovor od Alice, začne vyzvánět a vrátí svému SIP serveru proxy2 zprávu 180 (Ringing) o tom, že začal vyzvánět. Zpráva 180 (Ringing) je směrována stejným mechanismem přes proxy2 a proxy1 až k Alici, kde jí telefonu může informovat vyzváněcím tónem nebo zobrazit stav na display.

V našem případě se Bob rozhodne hovor přijmout. Když zvedne sluchátko, odešle se zpráva 200 (OK), která indikuje, že byl hovor přijmut. Tato zpráva obsahuje tělo s SDP protokolem, popisující multimediální detaily a v našem případě může vypadat následovně:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP server10.proxy2.com
    ;branch=z9hG4bKnashds8;received=192.0.2.3
Via: SIP/2.0/UDP pcAlice.proxy1.com
    ;branch=z9hG4bK776asdhds ;received=192.0.2.1
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
From: Alice <sip:alice@proxy1.com>;tag=1928301774
Call-ID: a84b4c76e66710@pcAlice.proxy1.com
CSeq: 314159 INVITE
Contact: <sip:bob@192.0.2.4>
Content-Type: application/sdp
Content-Length: 131
```

(Následuje SDP)

Zpráva projde přes proxy servery stejným principem jako zpráva 180 (Ringing) až k telefonu Alice, který přestane dávat vyzváněcí tón a pošle Bobovi potvrzení ACK, kterým stvrzuje, že je vytvořeno spojení. Toto potvrzení může být posláno již přímo na Bobův telefon bez použití proxy, protože během sestavování hovoru si účastníci pomocí zpráv INVITE a 180 (Ringing) vyměnili své adresy.

V této chvíli může začít multimediální seance s pakety ve formátu, na kterém se dohodli Alice s Bobem pomocí SDP protokolu. Mediální tok je zcela nezávislý na SIP signalizaci, může být vytvořen přímo mezi Bobem a Alicí a dokonce putovat jinou cestou než putovala SIP signalizace. Během multimediální seance se mohou Alice s Bobem dohodnout na změně formátu. K tomu slouží signalizační zpráva re-INVITE obsahující nový popis multimediálního toku. Na zprávu re-INVITE pak druhý účastník odpoví 200 (OK) a žadatel potvrdí pomocí zprávy ACK. Pokud druhá strana změnu nepřijímá, odešle zápornou odpověď například 488 (Not Acceptable Here), na kterou žadatel též odpoví ACK avšak nedojde ke změně ani selhání probíhajícího hovoru.

Na konci hovoru v našem případě Bob položí jako první a jeho telefon vygeneruje zprávu metody BYE. Zpráva BYE je směrována již přímo bez proxy serverů k Alici, která přijetí zprávy potvrdí zprávou 200 (OK) a ukončí celou relaci.

Další zajímavou a důležitou metodou je zpráva REGISTER. Registrace je cesta jak poskytnout svému SIP serveru informace o svém aktuálním umístění, což je nezbytná informace například při sestavování hovoru. SIP server se dívá do své databáze a z ní vyhledá údaj, kde se zrovna nacházíte. Aby měl SIP server aktuální informace, jsou zprávy REGISTER vysílány z User Agentů v periodicky se opakujících intervalech do SIP serveru. Zprávy REGISTER pak spojují určitou SIP URI (např. bob@proxy2.com) s konkrétním zařízením a IP adresou. Další úlohou registračního serveru může být autorizace a autentifikace. Ve skutečnosti a i v našem příkladu je často registrační server zároveň proxy serverem. Fyzicky se pak jedná o jeden server, avšak z pohledu logické topologie se jedná stále o dva různé servery. [9]

Pro potřebu získávat spolehlivější informace během sestavování hovoru vznikla metoda PRACK, která je definována v doporučení RFC 3262. Metoda je založena na zrcadlení mechanismu odpovědí typu 2xx (OK) a je posílána periodicky transakčním uživatelem, dokud nedojde k potvrzení o sestavení spojení metodou ACK. Mohlo by se zdát, že tento mechanismus dělá stejnou věc jako ACK, avšak jsou tu dva důležité rozdíly. PRACK je normální SIP zpráva stejně jako například BYE a dojde tedy k jejímu potvrzení na rozdíl od ACK. To zvyšuje spolehlivost a zároveň se zprávy PRACK směřují vždy skrze proxy servery, takže SIP server má přehled o stavu spojení. [10]

Mnohé systémy jako automatické spojovatelky či jiné interaktivní hlasové systémy vyžadují během hovoru zadání čísla pomocí tónové volby. Protože původní doporučení nspecifikovalo, jak bude tónová volba přenášena, vzniklo následně hned několik doporučení, které popisují přenos tónové volby. Jedním z nich je metoda INFO popsaná v doporučení RFC 6086. Ta umožňuje přenášet nejrůznější doplňkové informace k hovoru a DTMF volba může být právě jednou z takovýchto informací. Zpráva INFO pak může vypadat například takto:

```
INFO sip:7007471000@example.com SIP/2.0
Via: SIP/2.0/UDP alice.uk.example.com:5060
From: <sip:7007471234@alice.uk.example.com>;tag=d3f423d
To: <sip:7007471000@example.com>;tag=8942
Call-ID: 312352@myphone
CSeq: 5 INFO
Content-Length: 24
Content-Type: application/dtmf-relay

Signal=5
Duration=160
```

Problémem je však ve zdoluhavém vývoji doporučení RFC 6086, které vyšlo až v roce 2011 se snahou sjednotit a nahradit starší doporučení i proprietární řešení. Výsledkem je poněkud obecné doporučení, které neříká jak přesně tónovou volbu přenášet a záleží opět na výrobcí, jakým způsobem bude DTMF pomocí metody INVITE přenášet. O něco starší a v dnešní době pravděpodobně více používaná možnost přenosu tónové volby vychází z doporučení RFC 4733, respektive z jeho předchůdce RFC 2833, které popisuje jakým způsobem přenést tónovou volbu skrze RTP pakety. Jednotlivé značky tónové volby totiž není vhodné přenášet stejným kanálem, kterým probíhá hovor. Pro hovor jsou často

využívány úsporné kodeky s nízkou bitovou rychlostí jako je například G.723.1 nebo G.729, u kterých není možné zajistit spolehlivou detekci DTMF značky. Doporučení RFC 4733 definuje přenášení DTMF značek pomocí zvláštního RTP kanálu s předem danými parametry. V případě potřeby přenášet DTMF značku je pomocí protokolu SDP vytvořen nový kanál s dostatečně kvalitními parametry na přenos tónové volby a následně mohou být přenášeny jednotlivé značky.

Poslední rozšíření protokolu SIP, kterému se budu v této kapitole věnovat, je metoda MESSAGE, která umožňuje funkcionalitu zvanou Instant Messaging (IM). Jedná se o volitelné rozšíření, které dokáže v reálném čase posílat a přijímat krátké zprávy mezi uživateli. Zprávy mohou být posílány přímo mezi koncovými klienty, avšak pokud neznáme aktuální adresu adresáta, musíme využít model s proxy serverem, kdy proxy server zprávu přepošle adresátovi. Charakteristické pro metodu MESSAGE je, že na ní proxy server nijak nereaguje a jen jí přeposílá.

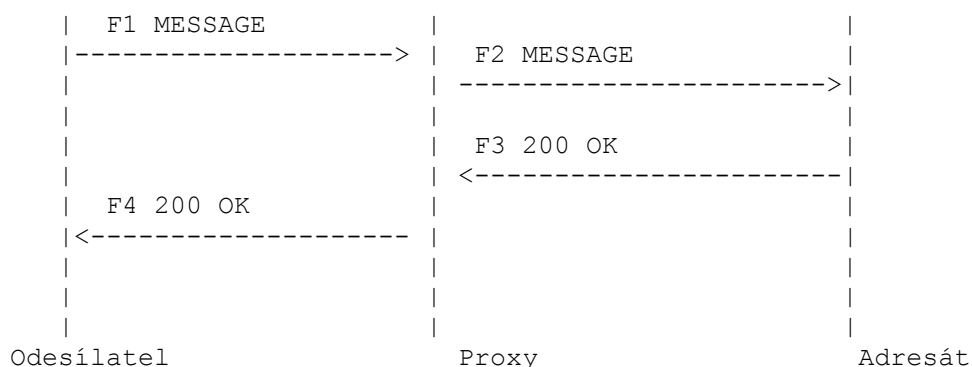


Schéma přeposílání SIP zprávy MESSAGE

Zpráva F1 pak může vypadat následovně:

```

MESSAGE sip:Odesílatel@domain.com SIP/2.0
Via: SIP/2.0/TCP user1pc.domain.com;branch=z9hG4bK776sgdkse
Max-Forwards: 70
From: sip: Odesílatel @domain.com;tag=49583
To: sip:Adresát@domain.com
Call-ID: asd88asd77a@1.2.3.4
CSeq: 1 MESSAGE
Content-Type: text/plain
Content-Length: 18

Text vlastní zprávy...
  
```

[11], [12]

3.1.3 SDP

Pro zahájení multimediálních telefonních konferencí, hlasových volání přes IP, streamování videa nebo jiných multimediálních relací v reálném čase, je nutné předem stanovit, v jakém formátu bude probíhat přenos dat. Pro tento účel slouží Session Description Protocol specifikovaný v doporučení RFC 4566, který popisuje formát sestavované multimediální relace. Zpráva protokolu SDP je obsažena v těle zprávy protokolu SIP a specifikuje následující parametry multimediálního obsahu:

- Typ multimediálního obsahu (video, audio, ...)
- Transportní protokol (RTP/UDP/IP, H.320, ...)
- Formát přenášených médií (H.261 video, MPEG video, ...)
- IP adresa a port, na který má být multimediální tok posílán

Protokol SDP je textově orientován a pro popis jednotlivých parametrů je použita standardní znaková sada (UTF-8), kdy na každém řádku je uveden typ parametru, který budeme specifikovat a za znaménkem rovná se je uvedena hodnota, kterou danému typu přiřadíme <Typ> = <hodnota>. První a povinný parametr ve zprávě SDP je vždy "v", který udává verzi protokolu SDP. Následují další parametry v přesně stanoveném pořadí, z nichž některé jsou pouze volitelné a nemusí být uvedeny. U parametrů, které nejsou povinné, je v následujícím seznamu uvedena hvězdička.

- v= (verze protokolu)
- o= (identifikátor iniciátoru a relace)
- s= (název relace)
- i=* (informace o relaci)
- u=* (URI popisované relace)
- e=* (emailová adresa)
- p=* (telefonní číslo)
- c=* (informace o připojení – IP adresa)
- b=* (informace o šířce pásma)
- z=* (úprava časového pásma)
- k=* (šifrovací klíč)

- a=* (nula nebo další parametry popisující média)
- t= (doba aktivní relace)
- r=* (doba opakování relace)
- m= (druh médií – audio/video/text/aplikace/zpráva, port, protokol pro přenos médií – udp/RTP/SRTP)

Celá zpráva protokolu SDP může pak vypadat následovně:

```
v=0
o=jdoe 2890844526 2890842807 IN IP4 10.47.16.5
s=SDP Ukazka
i=Ukázka SDP protokolu pro diplomovou práci
u=http://www.fel.cvut.cz
e=roztovoj@fel.cvut.cz (Vojtěch Roztočil)
c=IN IP4 224.2.17.12/127
t=2873397496 2873404696
a=recvonly
m=audio 49170 RTP/AVP 0
m=video 51372 RTP/AVP 99
a=rtpmap:99 h263-1998/90000
```

Protokol SDP tedy vyjednává veškeré informace o multimediálním toku, ale sám se ho již neúčastní. Pro přenos médií v IP telefonii se využívá protokol RTP, případně jeho zabezpečená varianta SRTP. [13]

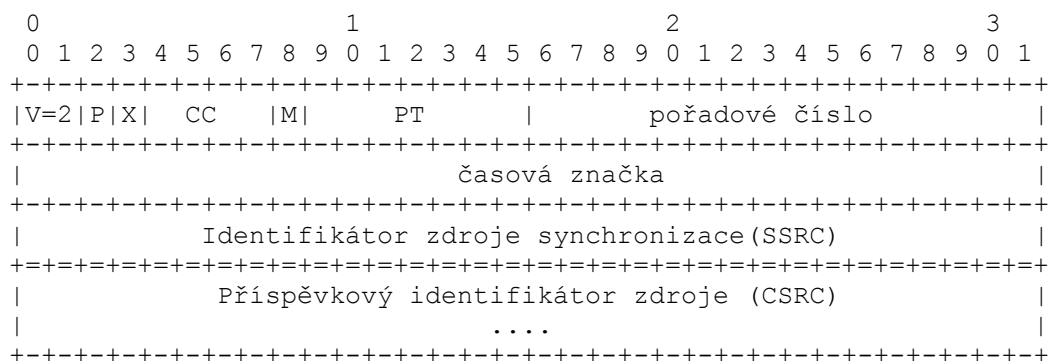
3.2 Přenos médií

3.2.1 RTP

Přenos dat v reálném čase umožňuje nejen v oblasti IP telefonie protokol RTP. Protože o sestavení hovoru a vyjednání informací ohledně multimediálního toku se starají protokoly SIP a SDP, protokol RTP už jen začne v daný moment přenášet multimediální data mezi koncovými zařízeními. Přestože se přenáší pouze multimediální data, obsahuje RTP paket ještě malé záhlaví, kde jsou uloženy informace o typu přenášených dat, způsobu kódování, zdroji synchronizace, jakož i sekvenční číslo paketu a časová značka pro obnovu synchronizace a real-time vlastností. Příjemací zařízení prostřednictvím sekvenčních čísel paketů zabezpečuje interpretaci dat z paketů RTP ve správném

pořadí. Podle definovaného prostředku synchronizace umožní časová značka v každém paketu načtení dat z paketů v odpovídajících časových intervalech. [14]

Záhlaví RTP protokolu obsahuje vždy následujících 12 B a dále můžou být volitelně připojena další data hlavičky.



Parametr V popisuje verzi RTP protokolu, P je výplňový bit, X říká, jestli bude hlavička obsahovat nějaká rozšíření, CC určuje počet CSRC identifikátorů, které budou následovat, M je bit, který může určovat určité důležité události jako je hranice rámců, pole PT identifikuje formát multimediálních dat, které se pak přenáší v těle RTP protokolu. Pořadové číslo, jehož hodnota se s každým odeslaným pakem zvyšuje o jedničku, slouží pouze pro přijímací zařízení, aby dokázalo správně sestavit multimediální obsah. 32 bitů časové značky udává okamžik vzorkování prvního oktetu v datové části RTP paketu, což slouží k synchronizaci a výpočtu parametru jitter. Identifikátor zdroje synchronizace (SSRC) je náhodně vygenerován a slouží k jednoznačné identifikaci multimediálního zdroje. Příspěvkový identifikátor zdroje pak pomáhá identifikovat zdroj, pokud je relace generována z více zdrojů zároveň.[15]

3.2.2 RTCP (Real Time Control Protocol)

Součástí doporučení RFC 3550, které specifikuje protokol RTP je i specifikace řídicího protokolu RTCP, který spolupracuje s protokolem RTP a poskytuje koncovým aplikacím informace týkající se kvality vysílaných dat v RTP protokolu a další údaje pro diagnostické účely probíhajícího spojení. RTCP protokol je periodicky vysílán všem účastníkům relace s intervalem v jednotkách

vteřin. RTCP vytváří zpětnou vazbu mezi účastníky relace protokolu RTP, ve které periodicky probíhá výměna RTCP paketů. RTCP pakety obsahují informace, podle kterých může strana vysílající multimediální proud dynamicky měnit např. rychlost přenosu na základě požadavků strany přijímající. Protokol RTCP tak poskytuje služby řízení toku a kontroly zahlcení sítě. Pro přenos RTCP používá stejně jako RTP transportní protokol UDP s číslem portu o jedna vyšší, než které je použito pro RTP. [14], [15]

4. Kodeky

Jedním z hlavních přínosů této analogové brány pro VoIP je, jak již bylo zmíněno dříve, poskytnout kvalitnější přenos hlasu v porovnání s dostupnými komerčními produkty na trhu. Pojem kvalita řeči se začal hojně používat již v době vzniku klasických telekomunikačních sítí. Úkol definovat kvalitu řeči je velmi obtížný, protože kvalita je velmi individuální záležitostí. Ať už dojdeme sebelepším způsobem k nějakému číslu reprezentující ohodnocení kvality nějakého úseku hovoru, vždy se najde někdo, komu bude daná kvalita nadmíru vyhovovat, a naopak někdo, kdo bude danou kvalitu považovat za nepřijatelnou. Dá se dokonce říci, že jeden a ten samý pozorovatel bude považovat kvalitu jistého degradovaného hovorového signálu po každém pozorování jinak, protože bude záležet i na tom, jakou má pozorovatel například náladu nebo bude ovlivněn jiným hovorovým signálem, který mu byl přehrán dříve. Je tedy prakticky nemožné označit nějaký úsek hovorového signálu (promluvu) konkrétním ohodnocením, které bude vždy, všude a pro každého stejné. Terminologii problematiky měření kvality hlasu upravuje doporučení ITU-T P.10. Toto doporučení obsahuje stručnou charakteristiku a případně další odkazy na jednotlivé termíny vyskytující v oblasti měření kvality, jejím cílem je sjednocení této terminologie a zabránění nejasnostem v interpretaci naměřených výsledků, bez ohledu na to, jakým způsobem byly tyto výsledky získány.

Samotný termín MOS (Mean Opinion Score) je definován normou ITU-T P.10 jako hodnota z rozsahu předdefinované stupnice, pomocí které testovací subjekty vyjadřují své hodnocení výkonnosti telefonního přenosového řetězce ve smyslu konverzačním nebo pouze poslechovém. Podle doporučení ITU-T P.800 existuje několik druhů stupnic a používá se právě ta, která je pro daný prováděný experiment vhodná. Nejčastěji používanou stupnicí je stupnice poslechové kvality, která je definována v tabulce 4.1. Jedná se o pětibodovou stupnici s pevně daným číselným rozsahem 1 – 5. Jednotlivým hodnotám odpovídá kvalitativní ohodnocení hlasového signálu. Toto slovní ohodnocení je rovněž přesně definované a mělo by být neměnné. Pro lepší pochopení se ještě často uvádí kratší slovní doprovod, který lépe postihuje ona jednoslovná ohodnocení. Výsledná kvalita se pak označuje anglickým termínem mean listening-quality opinion

score, zkráceně pouze mean opinion score, z toho pak vyplývá označení jednotky MOS.

MOS	Kvalita	Popis
5	Vynikající	Neznamenatelné rušení.
4	Dobrá	Rušení lze rozpoznat, ale není obtěžující.
3	Průměrná	Rušení lze rozpoznat a mírně obtěžuje.
2	Nízká	Rušení obtěžuje, je nutno vyvinout úsilí při snaze porozumět.
1	Špatná	Rušení velmi obtěžuje, řeč je nesrozumitelná.

Tabulka 4.1.: Tabulka stupnice MOS

Zatímco kvalita přenosu hlasu skrze analogové rozhraní závisí pouze na fyzikálních/elektrických parametrech telefonu a vlastního spojení, tedy metalickém páru, v digitálním světě je situace složitější. Začátek přenosového řetězce je stejný jako na analogovém rozhraní a podle Shannonova modelu komunikace jím je zdroj. Za zdroj lze považovat u telefonních systému mikrofon, do kterého člověk mluví a výsledná kvalita je dána fyzikálními parametry mikrofonu. Pokud chceme signál dále přenášet v digitální formě, je třeba signál v dostatečně kvalitě vzorkovat, kvantovat a následně kódovat. Při dodržení Shannonovy vzorkovací podmínky nedojde během vzorkování k degradaci signálu a bude ho stále možné obnovit. Během kvantování již ale dochází k zaokrouhlení úrovně aktuálního vzorku na celé číslo, což vede ke vzniku takzvaného kvantizačního šumu. Samozřejmě záleží na zvoleném počtu kvantizačních úrovní, a čím více kvantizačních úrovní máme, tím vzniká menší chyba v zaokrouhlení

a menší kvantizační šum. Klíčovou částí je pak kódování, kdy číslo získané během kvantování musíme vhodným způsobem zakódovat, aby bylo možné číslo později správně dekódovat a interpretovat. V dobách, kdy se začínal přenášet hlas v digitální formě, měly telekomunikační kanály z dnešního pohledu nízké přenosové rychlosti a bylo nutné hovorový signál kódovat tak, abychom co nejvíce šetřili šířku kanálu. Ukázkou mohou být kodeky GSM použité v mobilních sítích druhé generace, které ze vstupního datového toku 64kbit/s (PCM) produkují výstupní datový tok v závislosti na konkrétní verzi 5,6 kbit/s

až 13,2 kbit/s. Abychom snížili datový tok, museli jsme použít ztrátovou kompresi, což se negativně promítá na výsledné kvalitě hovoru. Algoritmů pro kódování zvuku existuje celá řada ale jak je vidět v tabulce 4.2, kodeky s opravdu malou přenosovou rychlostí nedosahují příliš dobré hodnocení kvality zvuku. [18]

Kodek	Přenosová rychlost [kbit/s]	MOS
G.711	64	4.1
G.723.1 - ACELP	5,3	3.65
GSM 06.10	13	3.5
G.729	8	3.92
G.726	32	3.85

Tabulka 4.2.:Přehled přenosové rychlosti a MOS vybraných kodeků

Od vývoje kodeku GSM uplynulo přes dvacet let a běžně používané přenosové rychlosti v datových sítích se zmnohonásobnily. Ve většině aplikací již není potřeba tolik dbát na co možná nejmenší přenosovou rychlost a můžeme využít vyšší přenosovou rychlost sítě pro kodek s lepší kvalitou.

V době vzniku metodiky ITU-T pro hodnocení kvality řeči se nepočítalo s využitím širokopásmových kodeků pro telefonii a stupnice MOS tedy není příliš vhodná pro hodnocení kvality hlasu s vysokým rozlišením. U širokopásmových kodeků sledujeme kromě kvality přenášeného zvuku a bitové rychlosti i například zpoždění vlivem použitého algoritmu. Tyto i další parametry podrobněji proberu v následujících kapitolách, kde se budu věnovat konkrétním širokopásmovým kodekům dostupným pod liberální licenci.

4.1 G.722

Standard G.722 vydaný mezinárodní telekomunikační unií v roce 1988 popisuje širokopásmový kodek pro přenos zvuku v šířce pásma 7 kHz s přenosovými rychlostmi podle zvolené varianty 48, 56 nebo 64 kbit/s. Kodek je uzpůsoben pro přenos zvuku konkrétně v pásmu od 50 Hz do 7 kHz. Při použití přenosové rychlosti 64 kbit/s je kodek schopen přenášet téměř dvojnásobné frekvenční

spektrum oproti kodeku G.711. Kódovací algoritmus dělí vstupní vzorky s rozlišením 16 kHz, 14 bitů pomocí digitálních filtrů na nižší frekvenční pásmo (0 až 4 kHz) a vyšší frekvenční pásmo (4 až 8 kHz). Každé dílčí pásmo je poté zakódováno pomocí adaptivní diferenční pulsní kódové modulace. Tato metoda využívající dílčí pásma a adaptivní diferenční kódovou modulaci se také označuje zkratkou SB-ADPCM a pracuje s nerovnoměrným vnímáním frekvencí lidským uchem. Nižší pásmo generuje datový tok 48 kbit/s, který spolu s datovým tokem 16 kbit/s z vyššího pásma tvoří vstup multiplexoru, jehož produktem je výstupní datový tok 64 kbit/s.

Protože patentové nároky na kodek G.722 vypršely, je v současné době kodek otevřen k volnému použití. Jinak je tomu u varianty G.722.1 a G.722.2, které používají stále patentově chráněné kompresní technologie. G.722.1 je založen na kódování Siren7 a nabízí nižší přenosové rychlosti. Novější typ G.722.2 také známý pod názvem AMRWB (Adaptive Multirate Wideband) je založena na kódování ACELP, které kromě nižší přenosové rychlosti nabízí schopnost rychle měnit kompresní poměr a při přetížení sítě změnit šíři pásma.[17]

4.2 SILK

Zvukový kodek SLIC pro přenos hlasu v reálném čase není na rozdíl od zmíněného kodeku G.722 nebo OPUS standardizován. Byl vyvinut totiž v laboratořích společnosti Skype Limited pro vlastní potřebu. Po vypršení licenčních ochranných lhůt byl kodek uvolněn včetně zdrojových kódů, je zdarma k dispozici a v současné době probíhá jeho standardizace.

Vstupní vzorkovací frekvence může být 8, 12, 16 nebo 24 kHz a přenosová rychlost se poté pohybuje v rozmezí 6 až 40 kbit/s. Výhodou je i poměrně malé zpoždění 25 ms vlivem použitého algoritmu, z čehož 20 ms je přenášený časový úsek a 5 ms je čas k lineární predikci (LPC), kterou SILK využívá.

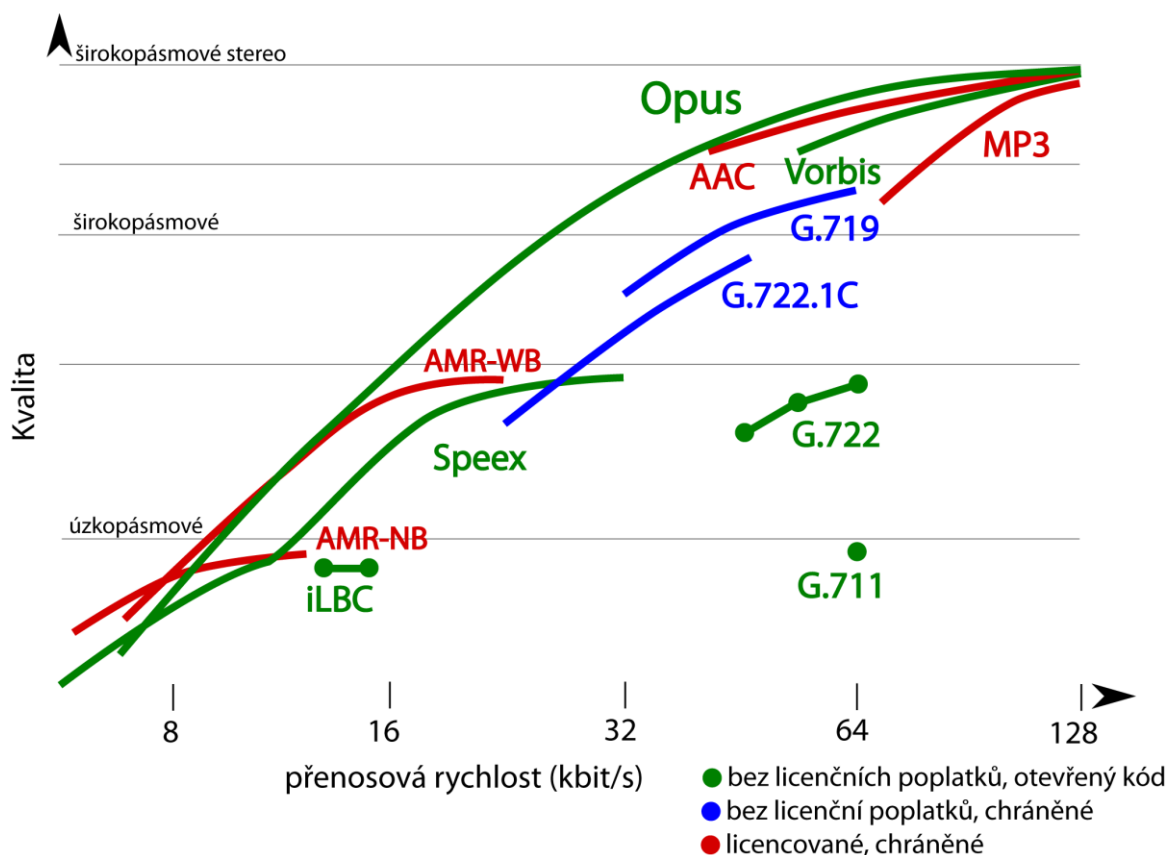
Výhod kodeku SILK si v době jeho uvolnění všimli vývojáři z Xiph.Org Foundation a rozhodli se ho zapracovat do nově vyvíjeného kodeku OPUS, který

by měl být široce škálovatelný, šetrný k operačnímu výkonu a nabízet lepší kvalitu než běžně používané kodeky vhodné pro přenos zvuku v reálném čase.

4.3 OPUS

OPUS je zcela otevřený, licenčními poplatky nezatížený, všestranný zvukový kodek určený jak k interaktivnímu přenosu řeči a hudby přes internet, tak pro ukládání a streamování zvuku. Jeho první verze byla standardizována v roce 2012 jako doporučení RFC 6716.

Opus spojuje technologie dvou zvukových formátů. První z nich je kodek SILK vyvinutý pro Skype a druhým je CELT z rodiny kodeků Ogg. Pro dosažení efektivní komprese mluveného projevu i hudebního záznamu je detekován typ zvuku a poté použita lineární predikce (nízký bitrate, SILK režim) nebo modifikovaná diskrétní kosinová transformace (vysoký bitrate, CELT režim) a „hybridní režim“, kdy řeč do frekvence 8 kHz je kódována LP, zatímco řeč nad 8 kHz je kódována pomocí MDCT. Obrázek 4.3.1 vykresluje závislost kvality přenášeného zvuku na přenosové rychlosti při použití různých kodeků. Přestože data na obrázku musíme brát s jistou rezervou, jelikož se jedná o prezentační obrázek vývojářů kodeku, je na něm dobře vidět možnost všestranného použití.

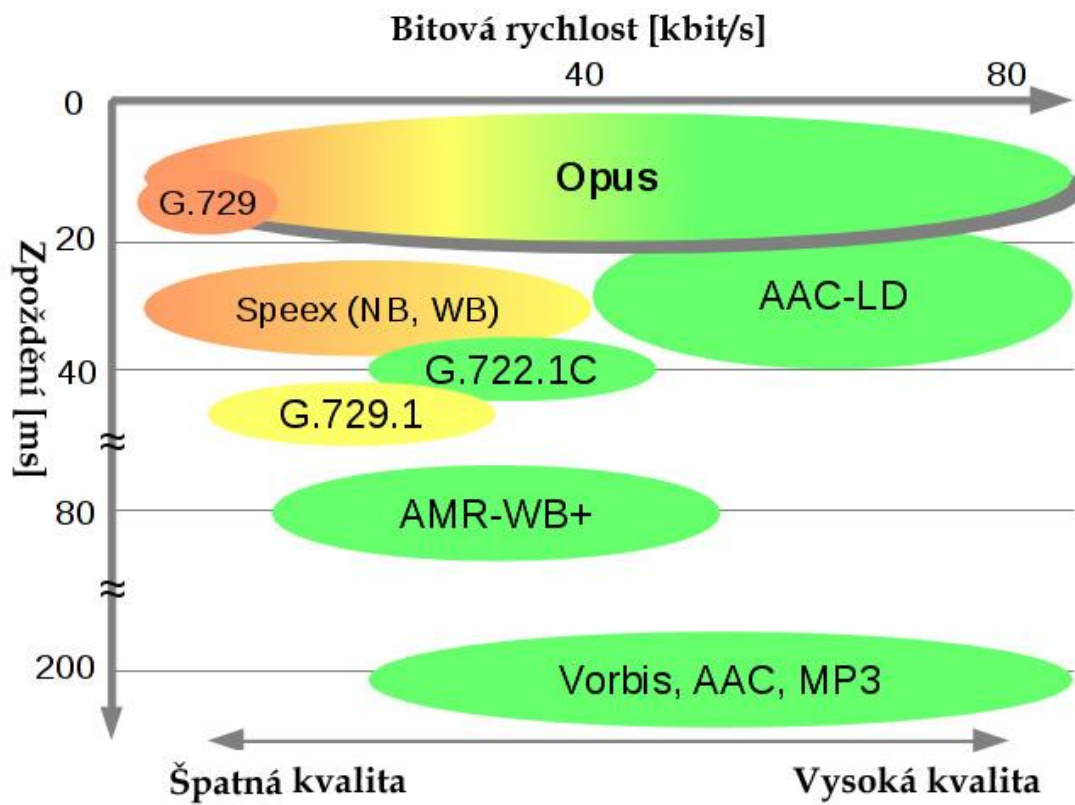


Obrázek 4.3.1: Závislost kvality zvuku na přenosové rychlosti kodeku OPUS a jiných vybraných kodeků

Kodek OPUS lze využít na celou škálu zvukových aplikací včetně přenosu hlasu přes internet, internetového vysílání rádia, hudby nebo například vzdálených živých hudebních vystoupení. Jsou podporovány následující formáty: [19], [20]

- Přenosová rychlost od 6 kbit/s do 510 kb/s
- Vzorkovací frekvence od 8 kHz do 48 kHz
- Doba trvání rámce od 2.5 ms do 60 ms
- Podpora konstantního i variabilního datového toku (CBR / VBR)
- Přenos od úzkopásmového až po širokopásmový zvuk
- Podpora pro přenos slova i hudby
- Přenos mono i stereo zvuku
- Dynamicky nastavitelná přenosová rychlost a doba trvání rámce.
- Dobrá odolnost proti ztrátě paketu (PLC)

Závěrem ještě stojí zmínit relativně nízké zpoždění algoritmizace kodeku v porovnání s jinými formáty. To ilustruje následující obrázek 4.3.2.



Obrázek 4.3.2: Porovnání zpoždění vybraných kodeků

5. Modulární koncepce

5.1 Řídící mikropočítač Raspberry Pi

Raspberry Pi je jednočipový počítač velikosti platební karty, založený na procesoru BCM2835 s architekturou ARM, vyvinutý britskou nadací Raspberry Pi Foundation s cílem podpořit výuku informatiky ve školách. Nápad cenově dostupného počítače se ujal a od roku 2012, kdy byla představena první verze, vzniklo postupně mnoho dalších verzí, které přináší lepší výkon či další funkcionalitu. Model Raspberry Pi B+ použitý v této práci disponuje procesorem 700 MHz single-core ARM1176JZF-S s architekturou ARMv6, čipem Broadcom BCM2835 a operační pamětí 512 MB. Pro připojení periférií slouží čtveřice USB 2.0 rozhraní, Ethernet konektor, HDMI (rev. 1.3) video výstup, analogový audio výstup přes 3,5 mm jack a v neposlední řadě GPIO pin hlavičky s 40 piny. Úložný prostor poskytuje slot na MicroSDHC paměťové karty.

Velikou výhodou Raspberry Pi oproti jiným vývojovým platformám je dostatečný výpočetní výkon, který umožňuje běh standardního operačního systému. V případě vývoje programu, který musí komunikovat s perifériemi na nízké úrovni, disponuje Raspberry Pi GPIO piny. Program může být kompilován i spuštěn na jednom zařízení na rozdíl od Arduina. V případě programů komunikujících na vyšší vrstvě lze pak bez problémů využít všechny dostupné protokoly a knihovny operačního systému a tím si ušetřit spoustu práce. V neposlední řadě - program napsaný a spouštěný na úrovni operačního systému je mnohem více kompatibilní než program psaný pro konkrétní hardware. Použití Raspberry Pi ideálně splňuje modulární koncepci analogové VoIP brány, kdy v případě potřeby není problém vyměnit Raspberry Pi za libovolný jiný počítač disponující GPIO piny, například výkonnější mikropočítač Banana Pi.

5.2 Zvukové I/O zařízení

Jak již bylo uvedeno v kapitole 2.5, kvalita zvuku je zásadní parametr této práce a v mém řešení je použita externí zvuková karta Behringer UCA202 s maximální vzorkovací frekvencí 48 kHz, při rozlišení převodníku 16 bitů. Připojení k mikropočítači Raspberry Pi je řešeno pomocí rozhraní USB 1.1 a tímto rozhraním může být připojena libovolná jiná externí zvuková karta, která nabízí alespoň srovnatelné parametry kvality zpracování zvuku. Použití jiné zvukové karty by nemělo mít vliv na funkčnost této práce.

5.3 Karta vnitřních linek

Pro připojení analogové účastnické přípojky do digitální ústředny se využívá účastnická sada, která zajišťuje jisté funkce. Protože analogová brána pro VoIP je částečně také digitální ústředna, měla by zastávat funkce účastnické sady, kterými jsou:

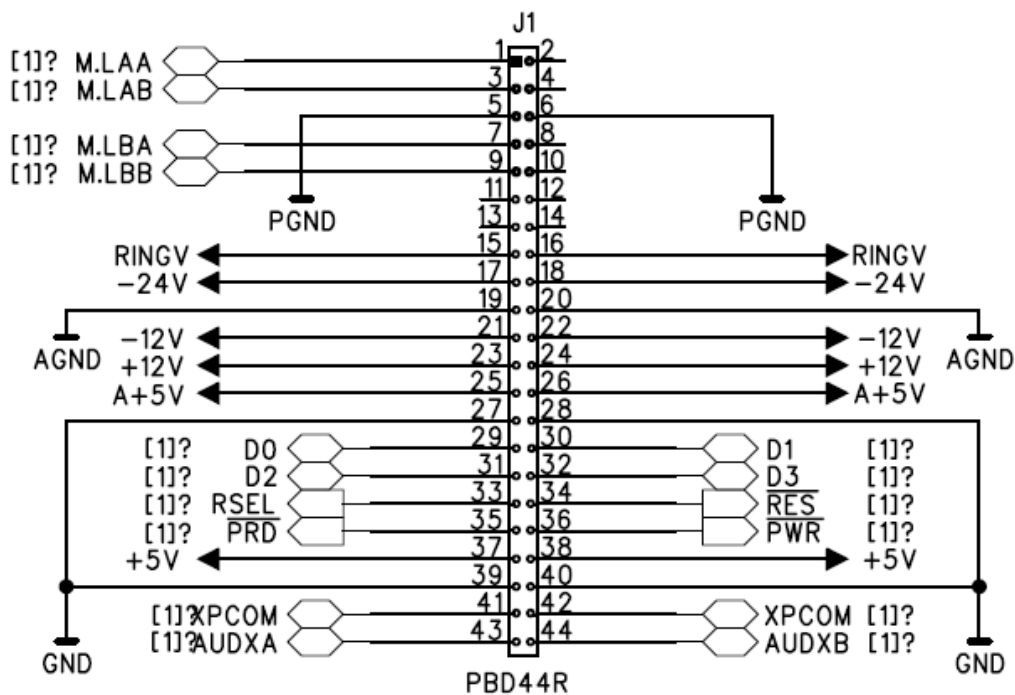
- Battery – stejnosměrné napájení účastnického vedení
- Overvoltage – napěťová ochrana proti přepětí na lince
- Ringing – vyzvánění účastníka
- Supervision – dohled nad spojením
- Coding – kódování – proces digitalizace signálu
- Hybrid – vidlice – pro oddělení směrů přenosu
- Testing – diagnostika

S ohledem na modulární koncepci celé práce, jsem se rozhodl využít kartu vnitřních linek z pobočkové ústředny Ateus Omega 2N. Tento modul zajišťuje základní funkce účastnické sady, nabízí kvalitní přenos zvuku a v současné době je na trhu stále k dispozici dostatečné množství těchto karet. Karta vnitřních linek Ateus Omega 2N nezajišťuje všechny funkce účastnické sady ale jen některé. Stará se o napájení účastnického vedení a částečně zajišťuje vyzvánění, dohled nad spojením a hybrid. Naopak karta nezajišťuje žádným způsobem coding, testing a ochranu před přepětím, což ale není vzhledem k tomu, že se jedná o kartu do pobočkové ústředny striktně vyžadováno.

Pokud potřebujeme upozornit na přichozí hovor pomocí vyzvánění, je třeba do datového pinu D1 zapsat logickou 1 a karta propojí vyzváněcí napětí, které je trvale přivedeno na vstup karty na a, b dráty účastnického vedení. Ručně je také třeba generovat mezery mezi vyzváněním a hlídat, aby nedošlo k vyzvánění ve chvíli, kdy má účastník přiložené sluchátko k uchu. Dohled nad spojením lze indikovat pomocí malé LED diody přímo na kartě vnitřních linek, která se v případě detekce protékajícího smyčkového proudu červeně rozsvítí. Druhou možností je čtení logického stavu na datovém pinu D1. To je bohužel stejný datový pin, který se používá pro připojení vyzváněcího napětí na a, b dráty. Pokud chceme z datových pinů D0 až D3 číst, musíme zapnout pomocí pinu PRD výstup integrovaného obvodu 74HCT257. Pro zápis do datových pinů D0 až D3 je nutné výstup z tohoto multiplexoru vypnout, nastavit požadovanou logickou úroveň na datových pinech a následně vytvořit vzestupnou hranu na pinu PWR. Pin PWR je zapojen do vstupu CLK obvodu 74HCT175. Integrovaný obvod 74HCT175 disponuje čtyřmi D obvodů, které nastaví na výstup logickou hodnotu ze vstupu v době vzestupné hrany. Tím je zajištěno, že lze nastavit na datové porty požadovanou úroveň a zároveň z nich číst. Schématické zapojení datových pinů je vidět na obrázku 5.3.2.

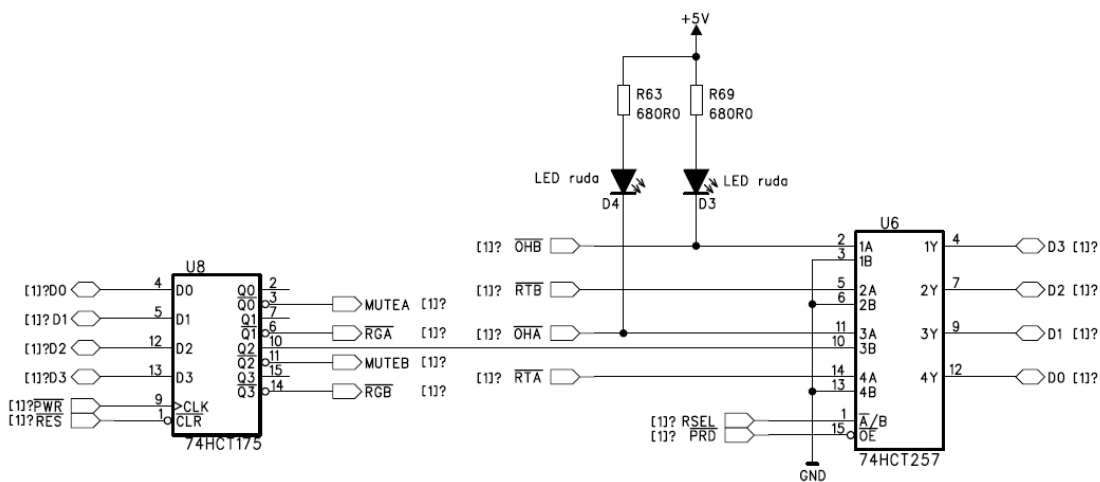
Hybrid, tedy převod z takzvaného dvou-drátu na čtyř-drát je na kartě vnitřních linek realizován pomocí elektronické vidlice, respektive několika operačních zesilovačů, avšak výstup z vidlice není nikam připojen a pro jeho použití je třeba demontovat dva rezistory z karty vnitřních linek. Původně byla karta konstruována pro přenos obousměrného zvuku do ústředny, kde se nacházela další vidlice. Obousměrný zvuk byl přenášen přes pin AUDXA, avšak po demontáži rezistorů z karty vnitřních linek slouží tento pin pouze jako vstup zvuku. Výhodou je, že karta žádným způsobem nezajišťuje převod analogového zvuku na digitální a můžeme využít kvalitního zvukové karty spolu s moderními kodeky popsané v kapitole 4 pro kvalitní přenos zvuku v digitální podobě.

Fyzickým rozhraním této karty, vyobrazené na obrázku 5.4.1 je 44 pinová dutinková lišta PBD.



Obrázek 5.3.1: Zapojení karty vnitřních linek

Piny 1,3 a 7,9 slouží k připojení dvou analogových telefonů. Dle technické dokumentace výrobce je linkové napájecí napětí rovno 24V, linkový napájecí proud je přibližně 25mA a vyzváněcí napětí s frekvencí 50Hz má efektivní hodnotu napětí 45V.



Obrázek 5.3.2.: Schématické zapojení datových pinů karty vnitřních linek

5.4 Karta vnější linek

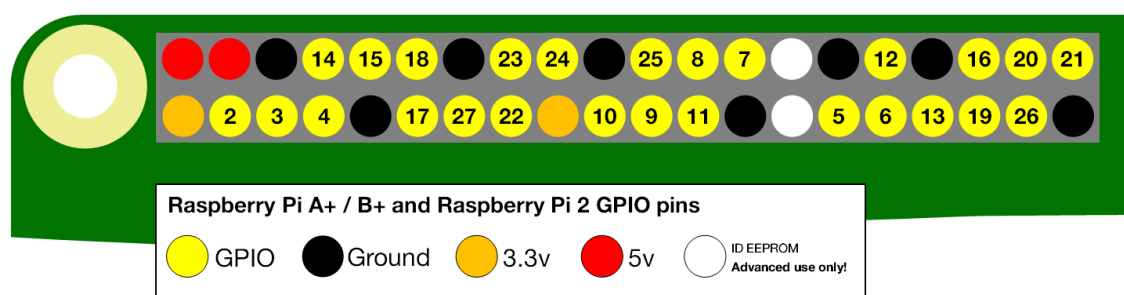
Stejným principem, kterým karta vnitřních linek zpřístupňuje FXS port pro připojení analogového telefonu je možné připojit kartu vnějších linek označenou OCOVL, která zpřístupní port FXO. Jediným rozdílem oproti kartě vnitřních linek je obrácená logika vyzvánění a on-hook stavu. Zatímco u modulu vnitřních linek je třeba detekovat stav on-hook a generovat vyzvánění, u státních linek je nutné naopak detekovat vyzvánění a ovládat stav smyčky. Propojení karty vnějších linek s mikropočítačem Raspberry Pi bude probíhat stejným způsobem jako propojení karty vnitřních linek popsané níže v kapitole 5.5.

Jelikož jsem neměl k dispozici kartu vnějších linek, není FXO port součástí řešení, avšak vzhledem k modulárnímu konceptu této diplomové práce je možné kartu jednoduše připojit a pouze drobnou úpravou řídicího programu popsaného v kapitole 6.2 FXO port zpřístupnit.

5.5 Propojení účastnické karty s mikropočítačem

K propojení mikropočítače Raspberry Pi s kartou účastnických linek, jak již bylo naznačeno v úvodní kapitole, slouží rozhraní GPIO, k němuž lze jednoduše přistupovat přímo z vrstvy operačního systému pomocí adresáře `/sys/class/gpio`. Tento adresář je ve skutečnosti speciální částí paměti a na disku bychom ho nenašli. Zapisováním či čtením z tohoto adresáře lze přímo ovládat GPIO piny bez nutnosti používání jakýchkoliv knihoven a k rozhraní lze tímto způsobem jednoduše přistupovat z libovolného programovacího jazyku. Na druhou stranu toto řešení přináší i svá úskalí. Protože se jedná o přímé propojení CPU, respektive části paměti, je číslování pinů poněkud komplikovanější. Ne všechny piny CPU totiž slouží pro GPIO a ne všechny jsou vyvedeny na pinovou hlavici. Pro snadnější práci s GPIO porty lze využít některou z dostupných knihoven. V mém případě jsem zvolil pravděpodobně nerozšířenější knihovnu pro práci s GPIO, kterou je WiringPi. Knihovnu napsal v jazyce C Gordon Henderson pro mikročip BCM2835 použitý v Raspberry Pi modelech A, B, B+, Compute Module, a Raspberry Pi Zero. Knihovna je šířena pod licencí GNU LGPLv3 a je použitelná nejen pro jazyk C ale též C++, RTB (BASIC) a další. Model

Raspberry Pi B+ použitý v této práci disponuje hlavicí s 40 piny, z nichž většinu lze použít v režimu GPIO, avšak některé piny jsou vyhrazeny na speciální funkce a použití v režimu GPIO je pak omezené. Rozložení pinů na hlavicí, číslovaných dle propojení na mikročip BCM je vidět na obrázku 5.4.1 níže. Obrázek také ilustruje a přibližuje důvody k využití knihovny. Značení pinů nezačíná od nuly, nejsou nikterak seřazeny a největším problémem je nekompatibilita s jiným modelem Raspberry Pi.



Obrázek 5.4.1: Zapojení pinů mikročipu BCM na pinové hlavici Raspberry Pi B+

Knihovna WiringPi definuje jako pin 0 BCM pin 17. Důvod je sice částečně historický ale hlavně se jedná o pragmatismus. Na obrázku 5.4.1. si lze povšimnout, že zde není GPIO pin 0 a 1. Ty má totiž BCM čip vyhrazen pro komunikaci s EEPROM. BCM piny 2-4 jsou sice pro GPIO využitelné, avšak mají zas jiná omezení, protože jsou vyhrazeny tentokrát pro komunikaci I2C a nelze vypnout 1,8 k Ω pull-up rezistor.[21],[22]

Pro komunikaci s účastnickou kartou jsem využil piny označené v knihovně WiringPi 0 - 3, 21, 22, 11, 10, což odpovídá u mého modelu Raspberry Pi BMC pinům 18, 17, 27, 22, 5, 6, 7, 8. Dále v této kapitole budu pro zjednodušení už vždy uvádět pouze značení pinů z knihovny WiringPi. Pro úplnost přikládám tabulku 5.4.1. se značením pinů.

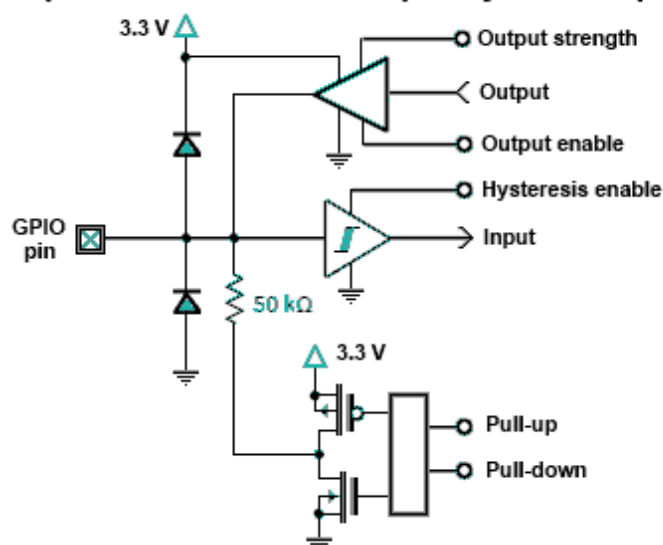
Značení WiringPi	Značení BCM	Značení WiringPi	Značení BCM
WiringPi 0	BCM 17	WiringPi 14	BCM 11
WiringPi 1	BCM 18	WiringPi 15	BCM 14

WiringPi 2	BCM 27	WiringPi 16	BCM 15
WiringPi 3	BCM 22	WiringPi 21	BCM 5
WiringPi 4	BCM 23	WiringPi 22	BCM 6
WiringPi 5	BCM 24	WiringPi 23	BCM 13
WiringPi 6	BCM 25	WiringPi 24	BCM 19
WiringPi 7	BCM 4	WiringPi 25	BCM 26
WiringPi 8	BCM 2	WiringPi 26	BCM 12
WiringPi 9	BCM 3	WiringPi 27	BCM 16
WiringPi 10	BCM 8	WiringPi 28	BCM 20
WiringPi 11	BCM 7	WiringPi 29	BCM 21
WiringPi 12	BCM 10	WiringPi 30	BCM 0
WiringPi 13	BCM 9	WiringPi 31	BCM 1

Tabulka 5.4.1: Rozložení pinu GPIO

Raspberry Pi má své GPIO piny poměrně univerzální a snadno pomocí softwaru můžeme upravit celou řadu jejich charakteristik. Lze zapnout / vypnout hysterezi na bázi Schmittova klopného obvodu, měnit výstupní proud od 2 mA do 16 mA s krokem 2 mA nebo zapnout pull-up či pull-down rezistor. Bohužel z důvodu patentové ochrany čipu BCM, výrobce Broadcom neuvádí přesné schéma zapojení GPIO pinů. Přibližné zapojení, které se podařilo zjistit reverzním inženýrstvím, je na obrázku 5.4.2. [23]

Equivalent Circuit for Raspberry Pi GPIO pins



Obrázek 5.4.2: Odhad vnitřního zapojení GPIO pinů.

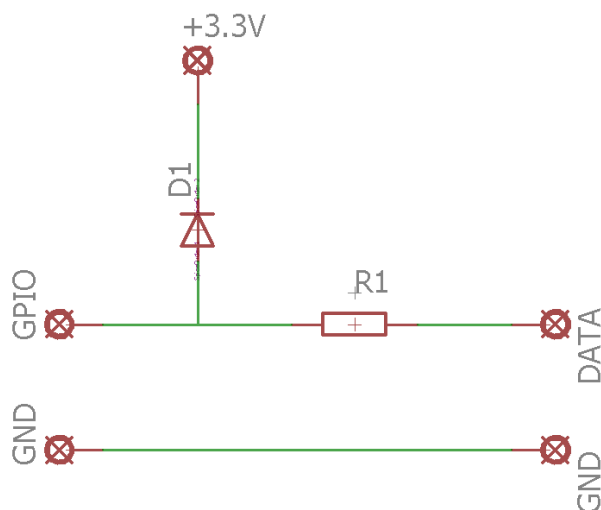
Pro komunikaci GPIO pinů Raspberry Pi s kartou vnitřních linek bylo potřeba vyvinout speciální převodní modul, který zajistí kompletní napájení pro kartu vnitřních linek a vyřeší rozdílné napěťové úrovně mezi GPIO piny a datovými vývody z karty vnitřních linek. Datové piny karty vnitřních linek mají 5 V logiku, respektive vstup zajišťuje integrovaný obvod 74HCT175 a výstupní režim obvod 74HCT257 zatím co GPIO piny Raspberry Pi mají logiku 3,3 V. Celkově tak můžou nastat čtyři rozdílné stavy. První dva z nich nastávají, pokud jsou GPIO piny nastaveny jako vstupní a čteme data z karty vnitřních linek.

- Napětí na výstupu integrovaného obvodu 74HCT257 v logické 1 je přibližně 5 V. Na vstup GPIO pinu Raspberry Pi lze ale zapojit pouze napětí 0 – 3,3 V. Pro tento případ je tedy potřeba napětí nějakým způsobem snížit. Jako nejvhodnější způsob, který zároveň neomezuje funkcionalitu GPIO pinu se ukázalo připojení diody mezi GPIO pin a 3,3 V, jejíž proud je nastaven rezistorem. Schématické zapojení je uvedeno na obrázku 5.4.3. Napětí na GPIO pinu sice nebude přesně 3,3 V, protože se k němu bude přičítat napětí na diodě, ale dle dostupných informací lze toto napětí už bezpečně připojit na vstup Raspberry Pi. [24]
- Napětí na výstupu integrovaného obvodu 74HCT257 v logické 0 je v rozmezí 0 V až 0,4 V v extrémním případě. Diodou v tomto případě

nepoteče proud, odpor nebude mít žádný efekt a Raspberry Pi bude správně detekovat úroveň.

V případě použití GPIO pinu jako výstupního můžeme definovat opět dvě možnosti:

- Napětí v logické 0 na GPIO pinu Raspberry Pi je 0 V s typickou maximální hodnotou 0,4 V. Pro integrovaný obvod 74HCT175 na kartě vnitřních linek do kterého zapisujeme je maximální hodnota napětí pro logickou 0 = 0,8 V. V tomto případě je vše v pořádku a logická úroveň by měla být v pořádku detekována.
- Posledním případem je logická 1 na GPIO pinu, která odpovídá napětí 3,3 V. Maximální vstupní proud integrovaného obvodu 74HCT175 je $\pm 1 \mu\text{A}$, tedy úbytek napětí na odporu bude zanedbatelný a vzhledem k minimálnímu napětí pro detekování logické 1, které je 2 V by měla být úroveň i zde v pořádku detekována.



Obrázek 5.4.3: Propojení GPIO pinů s kartou vnitřních linek

Tímto způsobem jsem propojil všechny datové piny (D0 - D3) a čtyři piny (RSEL, PRD, RES, PWR) ovládající integrované obvody 74HCT257 a 74HCT175.

Další úlohou tohoto převodního modulu je dodat správné referenční napětí na vstup karty vnitřních linek XPCOM. Ten slouží jako napěťová reference ke správné detekci stavu účastnických linek a bez něj karta není schopna správně fungovat. Jeho napětí bylo zjišťováno experimentálně, proto je na převodním modulu umístěn odporový trimr a napětí přiváděné na pin XPCOM lze plynule měnit v rozsahu 0 až 12 V.

Poslední funkcí převodní destičky je dodávat správné napájecí napětí pro kartu vnitřních linek. Zde byl pro jednoduchost použit laboratorní zdroj, který dodává všechna potřebná napětí a na převodní destičce dochází pouze k oddělení digitálního napětí +5 V od analogového napětí +5 V, které je poté ještě kapacitorem 100 μ F a 22 nF filtrováno proti případným napěťovým špičkám z integrovaných obvodů. Ze zdroje je pak ještě dodáváno napětí ± 12 V a -24 V. V případě potřeby by nebyl problém všechna napětí generovat přímo na kartě, avšak v tomto stádiu práce to nebylo vyžadováno.

6. Programové vybavení

Analogová bránu pro VoIP se skládá nejen z elektronických obvodů, sběrnic, pamětí, konektorů a dalšího hardwaru ale nedílnou součástí je též programové vybavení, které se stará o chod celého systému. Má koncepce analogové brány pro VoIP odpovídá ve své podstatě malé digitální ústředně, což znamená, že hardware je řízen programy, které se starají o spojování hovorů, signalizaci ale též testování, konfiguraci či komunikaci s dalšími moduly.

Při výběru softwaru máme na výběr ze tří základních možností. První z nich a zároveň nejvíce pracná možnost je napsání vlastního software. Výhody závisí na zručnosti programátora a kromě duševního vlastnictví kódu také žádné výhody být nemusí. Druhou možností je zakoupení softwaru, což přináší často snadné a rychlé nasazení, avšak většinou nemáme k dispozici zdrojové kódy a nelze provádět zásadní změny v systému. Poslední možností je využít software s otevřeným kódem, který je většinou zdarma a zároveň nabízí možnost libovolně systém upravovat. Pro zachování modulární koncepce ATA brány a potřeby komunikace s kartou vnitřních linek se jeví jako nejvýhodnější využít právě poslední možnost, tedy software s otevřeným kódem.

Budeme-li vybírat řešení z množiny programů s otevřeným kódem, měli bychom požadovat funkcionality pobočkové ústředny, podporu SIP protokol, případně jeho zabezpečené verze SIPS a šetrnost vzhledem k omezenému výpočetnímu výkonu mikropočítače Raspberry Pi. Všechny tyto parametry splňuje platforma Asterisk, jejíž odlehčená verze RasPBX je uzpůsobena pro snadné nasazení na mikropočítač Raspberry Pi.

6.1 RasPBX

Veškerá potřebná dokumentace včetně zdrojových kódů a souborů pro instalaci je k dispozici na webových stránkách www.raspberry-asterisk.org věnovaných projektu odlehčené verze Asterisku pro Raspberry Pi nazvaným RasPBX.

Vývoj softwarové telekomunikační platformy Asterisk prošel poměrně dlouhým vývojem a nikdy se úplně neobešel bez sponzorů. V současné chvíli je RasPBX nabízeno pouze s grafickým rozhraním FreePBX, vyvíjené společností Sangoma Technologies Corporation, která je hlavním sponzorem projektu Asterisk a FreePBX sice šíří zdarma pod svobodnou licenci GNU, avšak spousta modulů není v základní verzi k dispozici a je nutné je dokoupit.

Prvním krokem pro nasazení systému je stažení připraveného instalačního obrazu. Ten zahrnuje operační systém Raspbian, který je založen na aktuální verzi operačního systému Debian 8 / Jessie a dále obsahuje Asterisk a FreePBX. Stažený instalační obraz poté stačí zapsat na paměťovou kartu, z které je pak v Raspberry Pi zaveden operační systém včetně nainstalovaných a přednastavených všech potřebných programů. Pro přihlášení, ať už vzdáleně pomocí SSH nebo lokálně, využijeme přednastavené přihlašovací údaje uživatel: root a heslo: raspberry.

V této chvíli je nutné učinit rozhodnutí, zda využívat grafické rozhraní FreePBX či nikoli. Bohužel moc dobře nefunguje možnost využívat jak grafického rozhraní FreePBX tak klasické konfigurace Asterisk pomocí konfiguračních souborů současně. Grafické rozhraní FreePBX přepisuje konfigurační soubory umístěné v adresáři /etc/asterisk/ a zároveň si udržuje konfiguraci ve svoji SQL databázi. To v případě ručního zásahu do konfiguračního souboru může vést neočekávatelným stavům. Vzhledem k ne úplně standartnímu scénáři použití distribuce RasPBX doporučuji grafickou nadstavbu FreePBX vypnout. Dle dokumentace projektu RasPBX nelze nadstavbu FreePBX odebrat, ale lze ji deaktivovat pomocí následujících příkazů a skriptu asterisk, který je třeba stáhnout z oficiálních stránek z adresy <http://repo.raspbx.org/download/asterisk> a umístit do adresáře /etc/ini.d/

```
update-rc.d freepbx remove
update-rc.d apache2 remove
chmod 755 /etc/init.d/asterisk
update-rc.d asterisk defaults
[25]
```

Veškeré potřebné konfigurační soubory se nachází, jak již bylo napsáno, v adresáři `/etc/asterisk/`, kde jako první je nutné zapnout modul kanálu ALSA `chan_alsa.so`. Ten povoluje asterisku aby byly hovory přijímány / vytvářeny s použitím zvukové karty jako telefonního zařízení. Modul lze zapnout úpravou konfiguračního souboru `modules.conf`, kde je nutné přepsat `"noload = chan_alsa.so"` na `"load = chan_alsa.so"` nebo pomocí nástroje Asterisk command line interface (CLI) a příkazu `module load chan_alsa.so`.

Asterisk command line interface je speciální administrátorské rozhraní pro ovládání, konfiguraci, zobrazení výpisu protokolů, chyb a varování v reálném čase. Základní příkaz pro spuštění této konzole je `asterisk -r`. Pokud se podařilo zapnout modul `chan_alsa.so` je nutné restartovat jádro systému Asterisk například příkazem `core restart` z konzole.

V adresáři s konfiguračními soubory se nachází soubor `alsa.conf`, který se konfiguruje zvukové kanály ALSA přítomné v operačním systému. Advanced Linux Sound Architecture (ALSA) je projekt, který zpřístupňuje funkcionalitu zvukových zařízení Linuxovému operačnímu systému. V konfiguračním souboru `alsa.conf` je potřeba nastavit následující parametry:

```
autoanswer=no
context=local
input_device=plughw:2,0
output_device=plughw:2,0
```

Tyto parametry v postupném pořadí definují, že příchozí hovor nemá být přijmut automaticky, kontext v číslovacím plánu, vstupní a výstupní zvukové zařízení což je v našem případě externí zvuková karta.

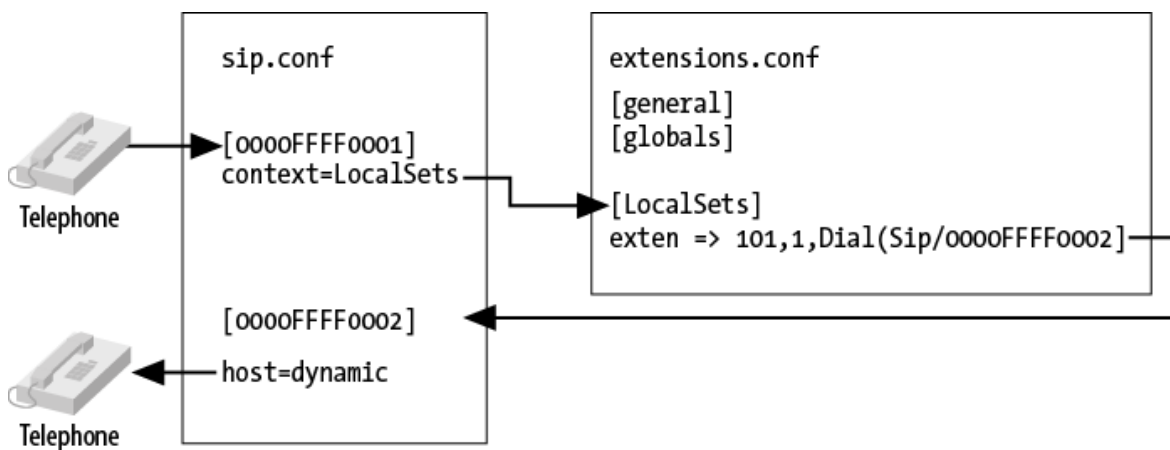
Tímto je nastavená "analogová část" brány a pomocí konfiguračního souboru `sip.conf` zbývá nastavit část pro internetové volání přes protokol SIP. První možnost je vytvoření nové SIP pobočky, čímž bude brána v roli SIP serveru. Pomocí následující konfigurace v souboru `sip.conf` vytvoříme nového uživatele "vojta" s číslem pobočky 17 v lokálním kontextu s použitím kvalitních zvukových kodeků OPUS, G.722, SLIC a pro zachování kompatibility s zařízeními, které nepodporují širokopásmový přenos zvuku G.711 A-law.

```
[vojta]
context=local
username=vojta
secret=123
userid=Vojtech Roztocil <17>
host=dynamic
allow=opus,g722,slic,alaw
```

Druhou možností je použít bránu Asterisk k SIP registraci například do veřejné telefonní sítě a tím bude brána v režimu klienta. :

```
register => uživatel:heslo@SIPserver:port/číslo
```

Poslední částí konfigurace je nastavení takzvaného číslovacího plánu. Ten je uložen v souboru `extensions.conf` a je srdcem systému Asterisk. Řídí totiž jak se má aplikovat logika hovorů na jakémkoliv připojení z jakéhokoli kanálu, co se stane, když zařízení vytočí pobočku 101 nebo so dělat s příchozím voláním od externího poskytovatele. Pro lepší představu propojení číslovacího plánu a například kanálu SIP je níže obrázek popisující jejich vzájemný vztah.



Obrázek 6.1.1: Vztah konfiguračních souborů `sip.conf`, `extensions.conf` a kontextu

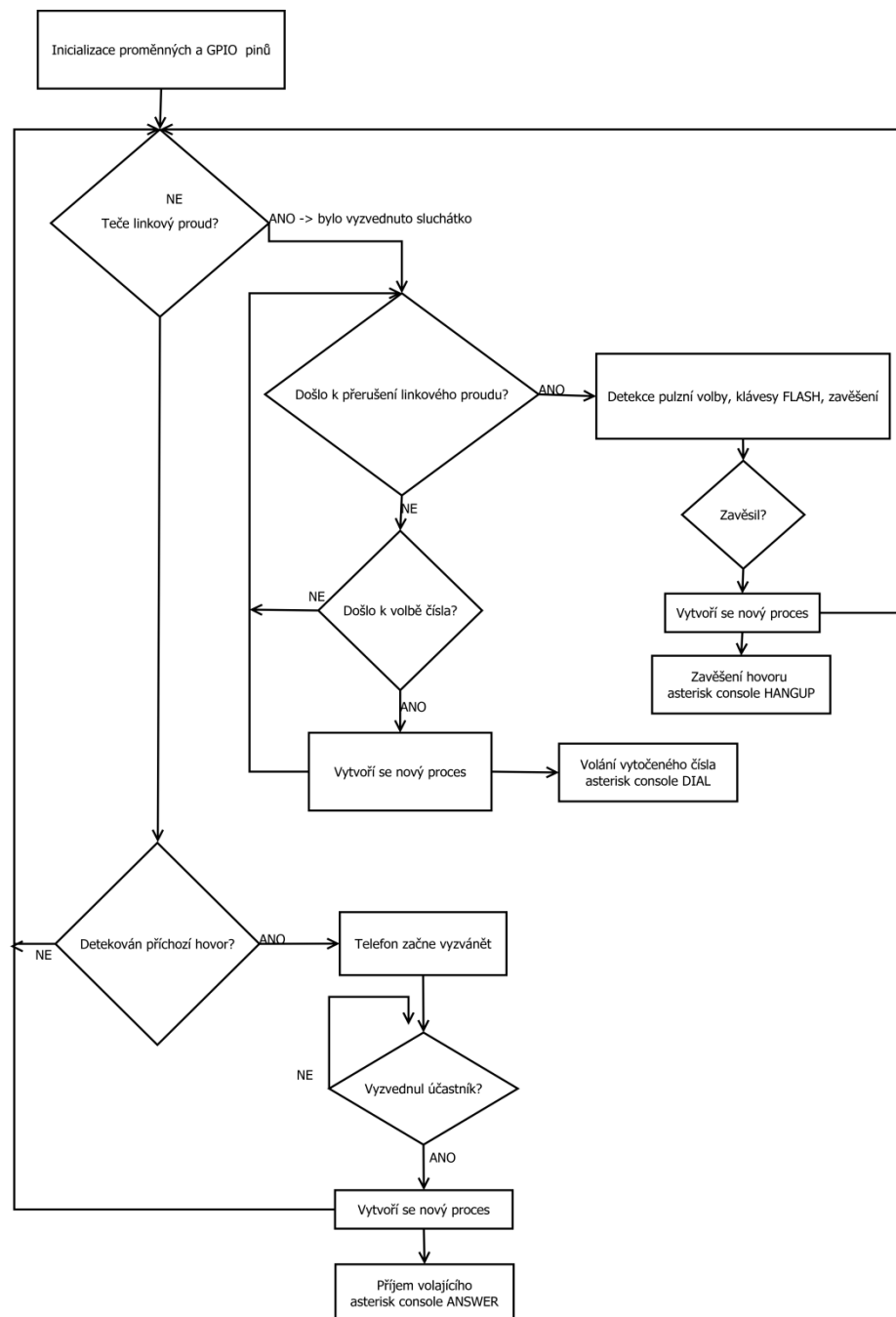
Pro vzájemnou komunikaci mezi Asteriskem a řídicím programem slouží soubor `console_call.txt`, do kterého Asterisk v případě příchozího hovoru zapíše 1.

```
exten => 101,1,System(/bin/echo "1" > /etc/asterisk/console_call.txt)
exten => 101,n,Dial(console/alsa)
```

6.2 Řídící program

Závěrečnou částí, která spojuje všechny dílčí bloky a vytváří z nich funkční řešení analogové brány pro VoIP je řídicí program. Ten jsem pojmenoval ATA a je umístěn v adresáři `/etc/init.d` aby docházelo k jeho automatickému spuštění při startu mikropočítače Raspberry Pi. Program zprostředkovává komunikaci mezi kartou vnitřních linek, respektive převodním modulem a softwarovou ústřednou Asterisk. Zásadními funkcemi programu jsou detekce stavu účastnické linky (on-hook / off-hook), rozpoznání znaků pulzní volby s následným předáním a vytočením volaného čísla. Pro příchozí hovory musí program zajistit rozpoznání příchozího volání, upozornění účastníka vyzváněním a následné spojení hovoru. Program je psaný v jazyce C pro překladač GNU Compiler Collection (GCC) a kromě knihovny `wiringPi.h` popsané v páté kapitole neobsahuje žádné nadstandardní knihovny nebo funkce.

Jak je naznačeno na obrázku 6.2.1, po startu programu dojde k inicializaci proměnných a nastavení GPIO portů. Datové porty D0 – D3 včetně portů, které řídí integrované obvody na kartě vnitřních linek jsou nastaveny jako výstupní. Následně program přejde do nekonečné smyčky, kde je pomocí funkce `digitalRead` z knihovny `wiringPi` čtena logická úroveň na datovém pinu D1, která v případě logické 0 znamená detekci smyčkového proudu na účastnickém vedení. Pokud není detekován smyčkový proud a podmínka není splněna, je v nekonečné smyčce druhá podmínka, která vyhodnocuje obsah souboru `/etc/asterisk/console_call.txt`. Do tohoto souboru Asterisk při příchozím volání na kanál ALSA zapíše 1 a tím program poté dokáže detekovat příchozí hovor. Je-li v souboru 1, je pak kladně vyhodnocena podmínka `if(file_i == 1)` a program se dostává z hlavního cyklu do vnořeného cyklu, který má dvě hlavní funkce. První je generovat ve správném intervalu vyzvánění a druhou je testovat, zda účastník hovor nepřijal vyzvednutím mikrotelefonu. Ve chvíli, kdy účastník přijme hovor, dojde pomocí funkce `fork` k vytvoření procesu potomka, který je identický s rodičem. Rozlišení mezi rodičem a potomkem je pomocí testování návratového kódu funkce `fork`. Rodič získá po návratu funkce PID procesu potomka. Potomek získá jako návratový kód 0 (nulu). Potomek má na starost informovat Asterisk o spojení hovoru. K tomu slouží funkce `execvp("asterisk", "asterisk", "-x", "console`



Obrázek 6.2.1: Funkční diagram programu ATA

answer", 0, 0), která nahradí současný proces procesem v parametru funkce. Původní proces bylo nutné rozdvojit na rodiče a potomka, protože potomek je nahrazen procesem “asterisk –x console answer“ a poté jeho úloha končí.

Proces rodiče dál pokračuje v hlavním cyklu programu a dostává se znovu k podmínce detekce smyčkového proudu. Podmínka je splněna a program se

dostává do vnořeného cyklu, který testuje, zda nebyla provedena pulzní volba, stisknutu tlačítko FLASH a nebo zda účastník nezavěsil. Tato detekce probíhá na základě jednoduché funkce `getPulzTime`, která vrací jako parametr délku pulzu. Ten je následně pomocí tří podmínek testován, a pokud je v rozmezí 1- 65 ms, jedná se o pulz pulzní volby. Pokud je délka pulzu větší než 65 ms ale kratší než 125 ms, jedná se o zmáčkuté tlačítko FLASH. Když je časový interval delší, vyhodnotí program akci uživatele jako zavěšení a stejným mechanismem jako u vyzvánění dojde pomocí funkce `fork` k vytvoření potomka původního procesu, který se sám nahradí procesem `“asterisk -x console hangup“`, čímž Asterisk informuje patřičným způsobem druhého účastníka hovoru, že bylo zavěšeno a ukončí spojení.

V případě kdy byla detekována pulzní volba, následuje další cyklus, který zajišťuje správné sestavení celého voleného čísla z dílčích pulzů a následně stejným mechanismem dojde k vytvoření potomka procesu, který se nahradí procesem `“asterisk -x console dial ret“`, kde `ret` je kompletní detekované číslo. Detekce zda je detekované číslo kompletní je založena na časové prodlevě 2,5 s a konjunkci nenulového počtu číslic ve voleném čísle.

Aby nedocházelo k zbytečnému vytěžování procesoru a ostatních prostředků mikropočítače Raspberry Pi, je ve většině smyček příkaz `delay` s parametrem 1 ms během kterého se chod programu pozastaví a dochází k úspoře výkon. Zároveň zpoždění o velikosti jedné milisekundy nemá pro lidské vnímání daných funkcí téměř žádný vliv a můžeme jej považovat za zanedbatelné.

7. Závěr

Cílem diplomové práce je nabídnout způsob, jak využít kvalitní zvukový aparát starých analogových telefonů, k přenosu hlasu paketovými sítěmi založených na protokolu IP s minimální ztrátou kvality přenášeného zvuku.

V této práci se podařilo navrhnout analogovou VoIP bránu s pokročilými audio kodeky pro širokopásmý přenos zvuku. Analogová část brány využívá kartu vnitřních linek z pobočkové ústředny Ateus Omega 2N, která je řízena mikropočítačem Raspberry Pi a zvuk je zpracováván externí zvukovou kartou. Přenos digitálně zpracovaného zvuku a signalizace sítí internet zajišťuje softwarová ústředna Asterisk, která je instalována na počítači Raspberry Pi.

Vlastní realizace projektu, která mimo jiné zahrnovala vývoj testovací desky pro plné zprovoznění karty vnitřních linek, výrobu převodního modulu zajišťující odlišné napěťové úrovně mikropočítače a karty vnitřních linek, napsání programu pro obsluhu karty či instalaci a konfiguraci softwaru, probíhala převážně v laboratorních podmínkách. Pokud bychom zvažovali použití výrobku pro laickou veřejnost, bylo by třeba vyvinout speciální zdroj, který by byl přijatelný svými rozměry, cenou a poskytoval všechna potřebná napětí popsaná v kapitole 5.5. Dále by bylo vhodné jednotlivé moduly umístit do společného krytu či pouzdra.

Závěrem bych chtěl velmi poděkovat panu Ing. Pavlu Trollerovi CSc., vedoucímu mé diplomové práce, za cenné připomínky, ochotu, trpělivost a účinnou pomoc při zpracování této práce.

7.1 Seznam použitých zkratek

AMRWB - Adaptive Multi-Rate Wideband

ATA - Analog Telephone Adapter

CDR - Call Detail Record

CPU - Central Processing Unit

DTMF - Dual-Tone Multi-Frequency

EEPROM - Electrically Erasable Programmable Read-Only Memory

FXO - Foreign Exchange Office

FXS - Foreign Exchange Station

GNU - GNU's Not Unix!

GPIO - General-Purpose Input/Output

GSM - Groupe Spécial Mobile

HDMI - High-Definition Multimedia Interface

HTTP - HyperText Transfer Protocol

IM – Instant Messaging

IMS – IP Multimedia Subsystem

IP – Internet Protocol

ISDN - Integrated Services Digital Network

ISO/OSI - International Organization for Standardization / Open Systems
Interconnection model

ITU - International Telecommunication Union

LED - Light-Emitting Diode

MOS - Mean opinion score

MTU - Maximum transmission unit

PCM - Pulse-Code Modulation

RFC - Requests For Comments

RTCP - RTP Control Protocol

RTP - Real-time Transport Protocol

SB-ADPCM Sub Band - Adaptive Differential Pulse-Code Modulation

SDHC - Secure Digital High Capacity

SDP - Session Description Protocol

SIP - Session Initiation Protocol

SRTP - Secure Real-time Transport Protocol

TLS - Transport Layer Security

UDP - User Datagram Protocol

URI - Uniform Resource Identifier

USB - Universal Serial Bus

VoIP - Voice over Internet Protocol

7.2 Seznam použitých obrázků

Obrázek 2.7: Rozhraní analogové brány pro VoIP

Obrázek 8.2.1: Příklad pulzní volby číslice tři

Obrázek 2.9.1: Příklad rozmístění FXO a FXS portů

Obrázek 4.3.1: Závislost kvality zvuku na přenosové rychlosti kodeku OPUS a jiných vybraných kodeků

Obrázek 10.3.2: Porovnání zpoždění vybraných kodeků

Obrázek 11.3.1: Zapojení karty vnitřních linek

Obrázek 5.3.2.: Schématické zapojení datových pinů karty vnitřních linek

Obrázek 5.4.1: Zapojení pinů mikročipu BCM na pinové hlavici Raspberry Pi B+

Obrázek 5.4.2: Odhad vnitřního zapojení GPIO pinů.

Obrázek 5.4.3: Propojení GPIO pinů s kartou vnitřních linek

Obrázek 5.4.4: Celkové schéma převodního modulu

Obrázek 12.1.1: Vztah konfiguračních souborů sip.conf, extensions.conf a kontextu

Obrázek 13.2.1: Funkční diagram programu ATA

7.3 Seznam použitých tabulek

Tabulka 2.2.1: Pulzní volba

Tabulka 2.2.2: Specifikace tónů – národní varianta

Tabulka 2.2.3: Specifikace tónů – varianta ETSI

Tabulka 2.3.1: Signalizace na rozhraních FXO a FXS

Tabulka 4.1.: Tabulka stupnice MOS

Tabulka 4.2.: Přehled přenosové rychlosti a MOS vybraných kodeků

Tabulka 5.4.1: Rozložení pinu GPIO

7.4 Seznam použité literatury:

- [1] RFC: 791: *INTERNET PROTOCOL, DARPA INTERNET PROGRAM, PROTOCOL SPECIFICATION* [online]. 1981 [cit. 2017-05-01]. Dostupné z: <https://ietf.org/rfc/rfc791>
- [2] RASPBERRY PI HARDWARE. *RASPBERRY PI* [online]. [cit. 2017-05-01]. Dostupné z: <https://www.raspberrypi.org/documentation/hardware/raspberrypi/>
- [3] *Síťový plán signalizace*. In: Ctu.cz [online]. 2004 [cit. 2017-05-01]. Dostupné z: <https://www.ctu.cz/sites/default/files/obsah/clanky/navrh-sitoveho-planu-signalizace-verejnych-telekomunikacnich-siti-poskytujicich-verejne-dostupnou/soubory/priloha-7-specifikace-tonu-1113656123.pdf>
- [4] *Principy - signalizace U+CLIP*. *Phonet.cz* [online]. [cit. 2017-05-01]. Dostupné z: http://www.phonet.cz/sig_u.html
- [5] 2N® - OMEGA 48 Technický popis: PŘÍRUČKA PRO INSTALACI Technický popis. In: 2N.cz [online]. [cit. 2017-05-01]. Dostupné z: 2.) https://www.2n.cz/cs_CZ/documents/22902/472859/Technick%C3%BD+popis+-+2N%C2%AE+Omega+48/1c07d11a-465b-4d7a-9945-27a4e3780859/searchTitle-Technick%C3%BD+popis+-+2N%C2%AE+Omega+48
- [6] IEEE SIGNAL PROCESSING MAGAZINE: The 1974 Origins of VoIP [online]. 2005, 2005(6) [cit. 2017-05-01]
- [7] HLAVÁČEK, J a R BEŠŤÁK. *Aktuální problémy řízení kvality služeb v IP telefonii* [online]. [cit. 2017-05-17]. ISSN 1214-9675. Dostupné z: <http://access.fel.cvut.cz/view.php?navezclanku=aktualni-problemy-rizeni-kvality-sluzeb-v-ip-telefonii&cislocclanku=2010010003>
- [8] Srovnání technologií SIP a H.323. *Intelek.cz* [online]. 2008 [cit. 2017-05-01]. Dostupné z: http://www.intelek.cz/art_doc-105E7A1C373BEE63C12573FC002C37C9.html
- [9] ROSENBERG, J. Request for Comments: 3261: SIP: Session Initiation Protocol [online]. 2002. [cit. 2017-05-01]. Dostupné z: <https://ietf.org/rfc/rfc3261>
- [10] ROSENBERG, J. *Request for Comments: 3262: Reliability of Provisional Responses in the Session Initiation Protocol (SIP)* [online]. 2002. [cit. 2017-05-01]. Dostupné z: <https://ietf.org/rfc/rfc3262>

- [11] HOLMBERG, C. *Request for Comments: 6086: Session Initiation Protocol (SIP) INFO Method and Package Framework* [online]. 2011. [cit. 2017-05-01]. Dostupné z: <https://tools.ietf.org/html/rfc6086>
- [12] SCHULZRINNE, H. *Request for Comments: 4733: RTP Payload for DTMF Digits, Telephony Tones, and Telephony Signals* [online]. 2006. [cit. 2017-05-01]. Dostupné z: <https://tools.ietf.org/html/rfc4733>
- [13] JACOBSON, V. *Request for Comments: 4566: SDP: Session Description Protocol* [online]. 2006. [cit. 2017-05-01]. Dostupné z: <https://tools.ietf.org/html/rfc4566>
- [14] PUŽMANOVÁ, Rita. *Streaming media (4): transportní protokoly RTP/RTCP* [online]. 2004 [cit. 2017-05-01]. Dostupné z: <http://www.dsl.cz/clanky/60-streaming-media-4-transportni-protokoly-rtp-rtcp>
- [15] SCHULZRINNE, H. *Request for Comments: 3550: RTP: A Transport Protocol for Real-Time Applications* [online]. 2003. [cit. 2017-05-01]. Dostupné z: <https://www.ietf.org/rfc/rfc3550.txt>
- [16] Behringer. *U-CONTROL UCA202* [online]. [cit. 2017-05-17]. Dostupné z: <https://www.music-group.com/Categories/Behringer/Computer-Audio/Audio-Interfaces/UCA202/p/P0484>
- [17] *International Telecommunication Union: G.722 : 7 kHz audio-coding within 64 kbit/s* [online]. [cit. 2017-05-17]. Dostupné z: <http://www.itu.int/rec/T-REC-G.722/en>
- [18] BRADA, M a J ZELENKA. *POSUZOVÁNÍ KVALITY HLASU* [online]. [cit. 2017-05-17]. Dostupné z: <http://www.ip-telefon.cz/data/download/40.pdf>
- [19] *Opus Interactive Audio Codec: Codec Landscape* [online]. [cit. 2017-05-17]. Dostupné z: <https://opus-codec.org/comparison/>
- [20] VALIN, JM, MOZILLA CORPORATION, K VOS, SKYPE TECHNOLOGIES S.A. a T TERRIBERRY. *Request for Comments: 6716: Definition of the Opus Audio Codec* [online]. 2012. [cit. 2017-05-01]. Dostupné z: <https://www.ietf.org/rfc/rfc6716.txt>
- [21] *Luketopia: Raspberry Pi GPIO via the Shell* [online]. [cit. 2017-05-17]. Dostupné z: <https://luketopia.net/2013/07/28/raspberry-pi-gpio-via-the-shell/>
- [22] *WiringPi: GPIO Interface library for the Raspberry Pi* [online]. [cit. 2017-05-17]. Dostupné z: <http://wiringpi.com/>
- [23] *Mosaic Documentation Web: GPIO Electrical Specifications* [online]. [cit. 2017-05-17]. Dostupné z: <http://www.mosaic-industries.com/embedded-systems/microcontroller-projects/raspberry-pi/gpio-pin-electrical-specifications>

[24] *Dr. Arno Wagner: Interfacing the Raspberry Pi to 5V Logic* [online]. [cit. 2017-05-17]. Dostupné z: <http://tansi.info/rp/interfacing5v.html>

[25] *Asterisk for Raspberry Pi: Documentation* [online]. [cit. 2017-05-17]. Dostupné z: <http://www.raspberry-asterisk.org/documentation>

8. Přílohy

Přílohou této diplomové práce je zdrojový kód řídicího programu ATA včetně detailních komentářů. Dále je přiložena konfigurace zásadních programů v mikropočítači Raspberry Pi, zejména softwarové ústředny Asterisk.

Hardware popisovaný v kapitolách 5.3 a 5.5 je znázorněn pomocí přiložených schémat zapojení.