

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

IMPLEMENTACE JEDNODUCHÉ POBOČKOVÉ ÚSTŘEDNY NA
OPENWRT

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. IVAN STRAČÁR

BRNO 2014



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ

ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

IMPLEMENTACE JEDNODUCHÉ POBOČKOVÉ ÚSTŘEDNY NA OPENWRT

SIMPLE PRIVATE BRANCH EXCHANGE IMPLEMENTATION IN OPENWRT

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. IVAN STRAČÁR

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. MILAN BARTL

BRNO 2014



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student: Bc. Ivan Stračár

ID: 129363

Ročník: 2

Akademický rok: 2013/2014

NÁZEV TÉMATU:

Implementace jednoduché pobočkové ústředny na OpenWRT

POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je naprogramovat jednoduchou demonstrační verzi pobočkové ústředny pro operační systém OpenWRT.

DOPORUČENÁ LITERATURA:

[1] FAINELLI, Florian. The OpenWrt embedded development framework. 2008.

[2] SELLS, Chris. Windows telephony programming: a developer's guide to TAPI. 1. vyd. Reading, Mass.: Addison Wesley Longman, c1998, xix, 295 p. ISBN 02-016-3450-3. Dostupné z:
http://www.google.cz/books?id=3M_mlvtGqUC&printsec=frontcover&hl=cs

[3] KIM, Do-Hyung, Min-Hong YUN, Sun-Ja KIM a Cheol-Hoon LEE. Design and Implementation of Smartphone Edition based on Embedded Linux. 2008 10th International Conference on Advanced Communication Technology. IEEE, 2008, s. 328-331. DOI: 10.1109/ICACT.2008.4493771.

Termín zadání: 10.2.2014

Termín odevzdání: 28.5.2014

Vedoucí práce: Ing. Milan Bartl

Konzultanti diplomové práce:

doc. Ing. Jiří Mišurec, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Diplomová práca sa zaoberá systémom OpenWRT. Inštaláciou tohto systému na smerovač Siemens Gigaset SX762. Popisuje kompiláciu a nahranie jednoduchého balíčka *helloworld* do tohto systému. Daný balíček mal za úlohu vypísanie jednoduchého slovného spojenia „Hello World“ do konzole systému OpenWRT. Balíček slúžil iba na ukážku, že systém OpenWRT umožňuje užívateľovi prispôbiť si ho podľa svojich potrieb. Následne sa do systému nainštalovala pobočková ústredňa Asterisk. Správne fungovanie PBX Asterisk bolo overené uskutočnením hovoru medzi dvomi softvérovými telefónmi ZoiPer. Ďalej je v práci opísané telefónne aplikačné programovacie rozhranie (TAPI). Niektoré jeho funkcie, rozhrania a potrebné balíčky pre komunikáciu so systémom OpenWRT. Na záver je predstavená testovaná topológia a overenie funkčnosti uskutočňovania hovorov medzi analógovými ale aj softvérovými telefónmi.

KLÚČOVÉ SLOVÁ

OpenWRT, Pobočková ústredňa Asterisk, SIP, TAPI, Zoiper

ABSTRACT

The diploma thesis deals with the system OpenWRT. Installing this system on the router Siemens Gigaset SX762. Describes how to compile and upload the simple package *helloworld* into this system. The package was tasked invitation simple phrase "Hello World" to the system console of OpenWRT. Package only serve to show that the system OpenWRT allows users to customize it according to their needs. After that it was installed PBX Asterisk into the system OpenWRT. Proper functioning of PBX Asterisk has been verified to make a call between two software phones ZoiPer. Furthermore, the work described telephony application programming interface (TAPI). Some of its functions, interfaces and packages needed to communicate with the system OpenWRT. In conclusion, the presented test topology and verify the operation of making calls between analog as well as softphones.

KEYWORDS

OpenWRT, PBX Asterisk, SIP, TAPI, Zoiper

STRAČÁR, Ivan *Implementace jednoduché pobočkové ústředny na OpenWRT*: diplomová práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2014. 44 s. Vedúci práce bol Ing. Milan Bartl

Výzkum popsáný v této diplomové práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

PREHLÁSENIE

Prehlasujem, že som svoju diplomovú prácu na tému „Implementace jednoduché pobočkové ústředny na OpenWRT“ vypracoval samostatne pod vedením vedúceho diplomovej práce, využitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej diplomovej práce ďalej prehlasujem, že v súvislosti s vytvorením tejto diplomovej práce som neporušil autorské práva tretích osôb, najmä som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných a som si plne vedomý následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona č. 121/2000 Sb., vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovenia § 152 trestného zákona č. 140/1961 Sb.

Brno

.....

(podpis autora)

POĎAKOVANIE

Rád by som poďakoval vedúcemu diplomovej práce pánovi Ing. Milanovi Bartlovi za odborné vedenie, konzultácie, trpezlivosť a podnetné návrhy k práci.

Brno

.....

(podpis autora)

OBSAH

Úvod	12
1 OpenWRT	13
1.1 História a vývoj verzií	13
1.2 Kompilácia OpenWRT	14
1.2.1 Kompilácia vlastného balíčka	14
1.2.2 Štruktúra súboru Makefile pre balíček Helloworld v OpenWRT	16
1.3 Grafické prostredie LuCI OpenWRT	17
1.4 Inštalácia OpenWRT na smerovač	18
1.5 Prvé prihlásenie do systému OpenWRT	19
1.5.1 Protokol SSH	20
1.5.2 Protokol Telnet (Telecommunications Network)	20
1.5.3 Protokol HTTPS (Hypertext Transfer Protocol Secure)	21
1.6 DHCP server v OpenWRT – Dnsmasq	22
1.7 Nastavenie LED diód na smerovači	22
2 Pobočková ústredňa Asterisk	23
2.1 Protokol SIP	23
2.2 Konfigurácia Asterisk	23
2.2.1 Sip.conf	24
2.2.2 Extensions.conf	25
3 Softvérový telefón ZoiPer	26
4 Telefónne aplikačné programovacie rozhranie (TAPI)	27
4.1 Verzie TAPI	27
4.2 Nastavenie TAPI v OpenWRT pomocou PBX Asterisk	28
4.2.1 Funkcie v súbore chan_lantiq.c	28
4.2.2 Významné rozhrania pre TAPI	30
5 Testovaná topológia	33
5.1 Smerovač Siemens Gigaset SX762	33
5.2 Telefón Planet VIP-101T	35
5.3 Overenie funkčnosti topológie	35
6 Záver	37
Literatúra	38

Zoznam symbolov, veličín a skratiek	40
Zoznam príloh	42
A Makefile pre balíček Helloworld	43
B Prepínač stavov v PBX Asterisk na rozhraní TAPI	44

ZOZNAM OBRÁZKOV

1.1	Balíček <i>helloworld</i> v konfiguračnom menu OpenWRT	15
1.2	Spustenie balíčka <i>helloworld</i> v OpenWRT	15
1.3	Internetové grafické prostredie LuCI	17
1.4	Webové rozhranie pôvodného softvéru na smerovači	18
1.5	Nastavenie siete v OpenWRT	19
1.6	Nastavenie LED diód v LuCI OpenWRT	22
3.1	Volanie zo softvérového telefónu na operačnom systéme Windows a aplikáciou na mobilnom telefóne	26
5.1	Schéma testovanej topológie	33
5.2	Doska smerovača	34
5.3	Smerovač Siemens Gigaset SX762	34
5.4	Telefón Planet VIP-101T	35
5.5	Fyzické zapojenie testovanej topológie	36
5.6	Zobrazenie smerovania hovorov medzi zariadeniami v PBX Asterisk .	36

ÚVOD

Veľakrát sa stáva, že ľudia hľadajú/potrebuju zaradenia, ktoré im budú ušité na mieru. Niekedy sa takéto zariadenia ani nevyrábajú, alebo sú príliš drahé. OpenWRT umožňuje prispôbiť si smerovač tak, že niekedy by bol aj výrobca prekvapený, čo jeho zariadenie dokáže. OpenWRT je vlastne distribúcia určená hlavne pre malé smerovače, plniace funkciu internetovej brány. Dokonca je podporovaná aj architektúra x86, ktorá umožňuje spustiť systém na bežných počítačoch alebo virtuálnych strojoch pre uľahčenie ladenia systému. V tejto práci bola použitá najnovšia stabilná verzia OpenWRT Attitude Adjustment. Táto verzia bola nainštalovaná na smerovač Siemens Gigaset SX762.

Pre vývojárov aplikácií poskytuje OpenWRT prostredie pre jednoduché pridávanie nových alebo existujúcich programov z iných platforiem. A pre užívateľov tak, ako už bolo spomenuté OpenWRT im poskytuje slobodu si prispôbiť dané zariadenie podľa svojich potrieb.

Prvá kapitola práce predstavuje OpenWRT, jeho históriu a vývoj verzií. Ďalej sa venuje opisu kompilácie jednoduchého balíčka, ktorý bol následne nahraný do OpenWRT a bola overená jeho funkčnosť. Následne je v tejto kapitole opísané grafické prostredie LuCI, ktoré sa používa pre jednoduchšiu správu systému OpenWRT. Kapitola pokračuje možnosťami inštalácie samotného systému na smerovač a prvým prihlásením sa do systému. Taktiež sú tu opísané použité protokoly pri prvom prihlásení. Na záver kapitoly je predstavený Dnsmasq, ktorý je obdobou DHCP (Dynamic Host Configuration Protocol) servera a možnosť ovládania LED (Light-Emitting Diode) diód na danom smerovači cez OpenWRT.

Druhá kapitola je venovaná pobočkovej ústredni Asterisk a jej nastaveniu. Táto ústredňa bola spustená pod systémom OpenWRT. Pomocou PBX Asterisk, boli realizované telefónne spojenia medzi dvomi softvérovými telefónmi ZoiPer a dvomi analógovými telefónmi Planet VIP-101T. V tejto kapitole je aj spomenutý protokol SIP pomocou ktorého komunikovali softvérové telefóny.

Tretia kapitola práce sa zaoberá overením funkčnosti PBX Asterisk a softvérovým telefónom Zoiper.

Následujúca kapitola je venovaná zoznámeniu sa s telefónnym aplikačným programovacím rozhraním (TAPI) a jeho verziám. Je v nej opísané aj nastavenie TAPI v OpenWRT. A spomenuté sú taktiež potrebné balíčky, ktoré OpenWRT potrebuje aby vedelo pracovať s TAPI.

Posledná kapitola práce popisuje použitú topológiu zapojenia potrebných zariadení pri plnení tejto práce. Ďalej sa venuje opisu použitých zariadení a funkčnosti danej topológie.

1 OPENWRT

OpenWRT je otvorený operačný systém v neustálom vývoji (GNU). Funguje pod Linuxovým operačným systémom a slúži ako distribúcia pre zariadenia (smerovače), so zabudovaným systémom ¹. OpenWRT poskytuje plne zapisovateľný súborový systém so správou balíkov ². To umožňuje, použiť balíčky prispôbené zariadeniu tak, aby vyhovovali ľubovoľným aplikáciám. Vývojárom poskytuje rámec pre vytvorenie aplikácie, bez nutnosti vytvárať kompletný firmvér a distribúciu okolo neho. Pre užívateľov OpenWRT znamená slobodu si zariadenie prispôbiť svojim vlastným potrebám. Už ako bolo spomenuté vyššie, ide o otvorený operačný systém, ktorý je úplne zadarmo. Oproti iným riešeniam je lepší vo výkone, stabilite, rozšíriteľnosti, dizajne atď. Cieľom vývojárov je zaistiť rozvoj a použitie OpenWRT v novátorských a dômyselných riešeniach.

1.1 História a vývoj verzií

Projekt OpenWRT začal v januári 2004. Prvá verzia bola vyvíjaná pre Linksys série WRT54. Táto verzia bola známa ako „**stabilná verzia**“ a bola široko používaná. Čísla stabilných verzií pozostávajú z roku a mesiaca, kedy boli vytvorené. A ďalšie tretie číslo identifikuje údržbu, alebo priebežnú správu verzie. Od roku 2005 OpenWRT oficiálne používa GNU/Linux jadro a už iba pridáva záplaty pre systém a ovládače pre sieťové rozhranie. Všetky stabilné verzie sú pomenované podľa niektorého nápoja, ktorého recept sa objaví po načítaní OpenWRT systému.

Verzie OpenWRT:

- **White Russian** – táto verzia je stále k dispozícii, ale nie je už ďalej podporovaná a udržiavaná. Posledná verzia White Russian je 0.9 z roku 2007.
- **Kamikaze** – vznikala v roku 2006. Kamikaze 7.07 je dostupná pre 6 procesorových a 7 systémových platforiem. Podporované sú aj x86 procesory.
- **Backfire** – prvá verzia bola vydaná v apríli 2010 pod označením 10.03. Táto verzia je postavená na linuxovom jadre 2.6.32 a priniesla podporu viacerých hardvérových platforiem.
- **Attitude Adjustment** – prvá verzia bola vydaná v apríli 2013 pod označením 12.09. Je postavená na linuxovom jadre 3.3 a oproti predchádzajúcej verzii pribudla podpora ďalších platforiem.
- **Barrier Breaker** – vo vývoji.

¹Ide o jednúčelový systém, v ktorom je riadiaci počítač s celo zabudovaným zariadením, ktoré ovláda

²Podľa [3] v súčasnosti existuje viac ako 2000 softvérových balíčkov

1.2 Kompilácia OpenWRT

Kompilácia prebieha v operačnom systéme Linux. V tejto práci bola skompilovaná OpenWRT verzia Attitude Adjustment. Prvým krokom kompilácie, bolo nainštalovanie podverzie Attitude Adjustment so všetkými jej balíčkami pomocou príkazu:

```
svn co svn://svn.openwrt.org/openwrt/branches/attitude_adjustment
```

Druhým krokom bolo zaktualizovanie a nainštalovanie všetkých dostupných balíčkov:

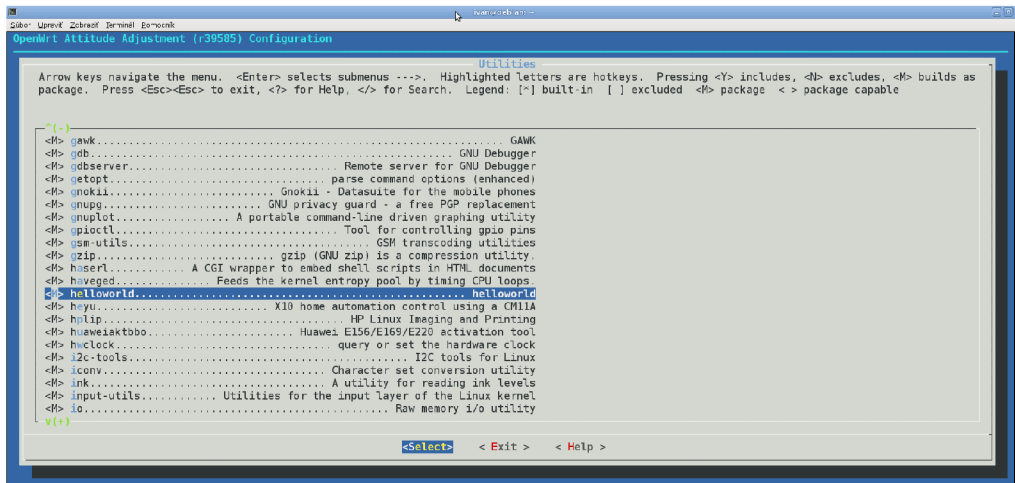
```
./scripts/feeds update -a  
./scripts/feeds install -a
```

V ďalšom kroku bolo spustené konfiguračné prostredie OpenWRT pomocou `make menuconfig`. V tomto prostredí sa dá nastaviť OpenWRT podľa požadovaných vlastností užívateľa. V tejto práci je použité nastavenie od tvorcov OpenWRT pre smerovač na jadre typu lantiq so systémom danube. Po tomto nastavení sa museli do prostredia Linux doinštalovať potrebné balíčky pre kompiláciu. Následne bola spustená samotná kompilácia OpenWRT príkazom `make`. Kompilácia zvyčajne trvá 2–6 hodín. Ak by sa na konci kompilácie vyskytol problém, tak po odstránení tohto problému, sa musí opäť spustiť kompilácia systému. Táto kompilácia systému trvá už výrazne kratšiu dobu. Po úspešnej kompilácii bol získaný obraz systému, ktorý sa v tomto prípade nainštaloval na daný smerovač.

1.2.1 Kompilácia vlastného balíčka

Pre názornú ukážku bol vytvorený a následne skompilovaný jednoduchý balíček *helloworld*. Pre kompiláciu tohoto balíčka boli vytvorené dva súbory `helloworld.c` a `Makefile`. Súbor `helloworld.c` obsahuje kód s jednoduchým výpisom do konzoly a súbor `Makefile` kompiluje tento kód a zároveň vytvára archív `tar.gz` týchto dvoch súborov. To, aby sa tento balíček objavil v balíčkoch konfiguračného menu OpenWRT, sa musí vytvoriť zložka *helloworld* v balíčkoch OpenWRT systému. A zároveň je potrebné v tejto zložke vytvoriť `Makefile`, ktorý okrem samotnej kompilácie už vytvoreného archívu `tar.gz` definuje sekciu a samotné meno balíčku, pod ktorým sa objaví v konfiguračnom menu (viz obr. 1.1). Aby sa vytvorený balíček objavil v konfiguračnom menu, je potrebné zaktualizovať a doinštalovať vytvorený balíček pomocou:

```
./scripts/feeds update -i  
./scripts/feeds install helloworld
```



Obr. 1.1: Balíček *helloworld* v konfiguračnom menu OpenWRT

Kompilácia samotného balíčka sa vykoná nasledujúcim príkazom:

```
make package/helloworld/compile
```

Následne bol skompilovaný balíček prenesený a nainštalovaný do OpenWRT, kde sa overila jeho funkčnosť (viz obr. 1.2).

```
Using username "root".
root@192.168.1.1's password:

BusyBox v1.19.4 (2013-03-12 15:13:26 UTC) built-in shell (ash)
Enter 'help' for a list of built-in commands.

 _ _ _ _ _
|_ | W I R E L E S S   F R E E D O M
|_|

-----
ATTITUDE ADJUSTMENT (12.09, r36088)
-----
* 1/4 oz Vodka      Pour all ingredients into mixing
* 1/4 oz Gin        tin with ice, strain into glass.
* 1/4 oz Amaretto
* 1/4 oz Triple sec
* 1/4 oz Peach schnapps
* 1/4 oz Sour mix
* 1 splash Cranberry juice
-----

root@Penzista:~# opkg install helloworld_1.0.0-1_lantiq.ipk
Installing helloworld (1.0.0-1) to root...
Configuring helloworld.
root@Penzista:~# helloworld
Hello World
```

Obr. 1.2: Spustenie balíčka *helloworld* v OpenWRT

1.2.2 Štruktúra súboru Makefile pre balíček Helloworld v Open-WRT

Tento súbor je veľmi dôležitý, pretože popisuje cestu k balíčku (prípadne adresu, z ktorej sa má stiahnuť) a jeho kompiláciu. Ako prvý krok, je nutné si definovať premenné, z ktorých bude balíček zostavený:

```
include $(TOPDIR)/rules.mk
PKG_NAME:=helloworld          ;meno pod ktorým sa zobrazí balíček v me-
                               nuconfigu
PKG_VERSION:=1.0.0           ;verzia balíčka
PKG_RELEASE:=1               ;verzia balíčka po kompilácii

PKG_SOURCE:=$(PKG_NAME)-$(PKG_VERSION).tar.xz ;umiestnenie balíčka,
                                               ktorý sa má kompilovať
PKG_BUILD_DIR:=$(BUILD_DIR)/$(PKG_NAME)-$(PKG_VERSION)
;umiestnenie balíčka po kompilácii
```

V strednej časti súboru Makefile sa nachádza definícia balíčka v menuconfigu Open-WRT. Napríklad kategória, pod ktorou sa bude balíček nachádzať v menuconfigu a jeho meno, poprípade jeho poznámka, ktorá balíček popisuje.

```
include $(INCLUDE_DIR)/package.mk

define Package/$(PKG_NAME)    ;názov balíčka v menuconfigu
SECTION:=utils                ;typ balíčka
CATEGORY:=Utilities           ;kategória v menuconfigu
TITLE:=$(PKG_NAME)            ;poznámka ktorá špecifikuje daný balíček
endef
```

Do Makefilu sa môže pridať aj ľubovoľná poznámka pomocou nasledujúceho príkazu. Táto poznámka slúži na popis daného balíčka.

```
define Package/$(PKG_NAME)/description
    Toto je hello world balíček ktorý slúži na jednoduché vypísanie
    slovného spojenia "Hello World" do konzole.
endef
```

Nasledujúci príkaz definuje, ako sa má daný balíček kompilovať.

```
define Package/helloworld/Build/Compile
    $(MAKE) -C $(PKG_BUILD_DIR) helloworld
endef
```


Sada príkazov pre skopírovanie súborov z kompilácie a vloženie do ipkg³ (Itsy Package Management System), ktorý je reprezentovaný adresárom \$(1).

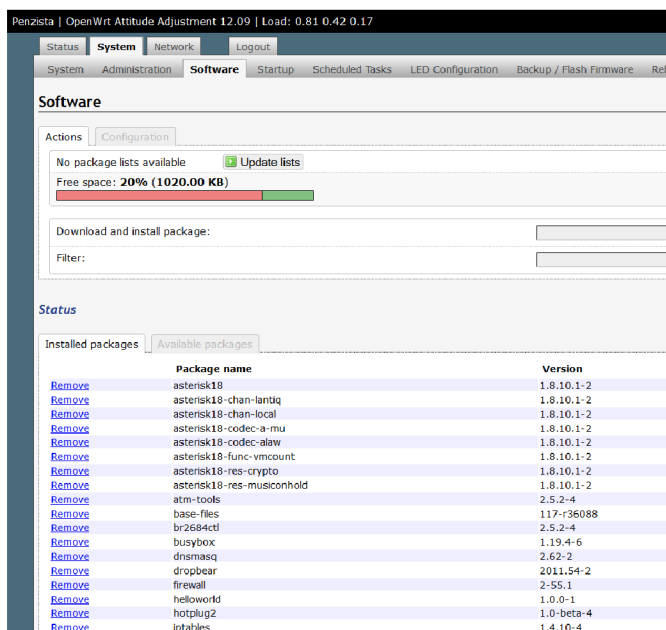
```
define Package/${PKG_NAME}/install
$(INSTALL_DIR) $(1)/usr/bin
$(CP) $(PKG_BUILD_DIR)/$(PKG_NAME) $(1)/usr/bin
endef
```

Na záver dokumentu sa volá balíček pre kompiláciu. Obsahuje iba jeden argument a to názov kompilovaného balíčka. Všetky ostatné informácie sa vyberajú z tzv. define blokov.

```
$(eval $(call BuildPackage,$(PKG_NAME)))
```

1.3 Grafické prostredie LuCI OpenWRT

OpenWRT ponúka internetové grafické prostredie LuCI, pomocou ktorého sa dajú jednoduchým spôsobom nastaviť požadované vlastnosti smerovača. Ponúka prehľadnú štatistiku inštalovaných a dostupných balíčkov. Na obrázku 1.3 je vidieť časť zoznamu inštalovaných balíčkov v OpenWRT vrátane balíčka *helloworld*, ktorého vytvorenie a kompilácia boli popísané vyššie.

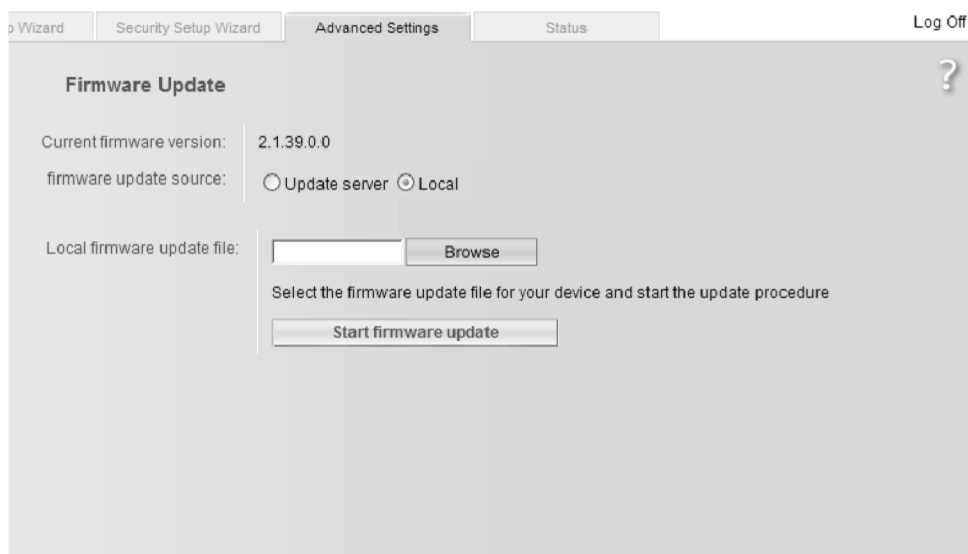


Obr. 1.3: Internetové grafické prostredie LuCI

³ systém pre správu balíčkov pre určitý systém. Podobá sa systému Debian dpkg

1.4 Inštalácia OpenWRT na smerovač

Pri inštalácii OpenWRT na smerovač, je nutné si overiť, či OpenWRT podporuje daný hardvér. Ak je daný hardvér podporovaný vo viacerých verziách OpenWRT, treba sa rozhodnúť, ktorá z verzií bude vyhovovať požadovaným vlastnostiam. Samozrejme najlepším riešením je, použitie poslednej stabilnej verzie (podpora, najnovšie balíčky, ...). V tejto práci bol použitý smerovač **Siemens Gigaset SX762**, do ktorého bola nainštalovaná verzia Attitude Adjustment. Inštalácia prebiehala cez webové rozhranie (viz obr. 1.4), kde bol zvolený zdroj aktualizácie skompilovaný obraz systému OpenWRT.



Obr. 1.4: Webové rozhranie pôvodného softvéru na smerovači

Okrem inštalácie OpenWRT cez webové rozhranie existujú ešte ďalšie tri možnosti inštalácie:

1. **Bootloader (zavádzač) a ethernet port** – väčšina, ak nie všetky zavádzače poskytujú integrované funkcie pre tento účel. Niektoré používajú TFTP/FTP-klienta, iné TFTP/FTP-server, niektoré webový server a niektoré používajú XMODEM-protokol.
2. **Zavádzač a sériový port**
3. **JTAG**⁴ – nahranie nového obrazu ramdisku na zariadenie. Existujúci firmvér na flash pamäti zostáva bezo zmeny. Nahranie nového ramdisku sa realizuje cez sériový kábel.

⁴Štandard pre porty testovacieho prístupu používaného pre testovanie plošných spojov metódou boundary scan.

1.5 Prvé prihlásenie do systému OpenWRT

Pri prvom prihlásení sa do systému, bolo nutné nastaviť na počítači IP adresu (DHCP server na danom smerovači nebol ešte zapnutý). Bola zvolená ľubovoľná IP adresa z rozsahu adres siete, kam patril aj daný smerovač. Pre pripojenie sa na smerovač, bol použitý protokol telnet. Následne bolo zmenené heslo pre administrátora (roota), pomocou príkazu `passwd`. Po zadaní tohto príkazu, systém vyzve užívateľa aby zadal nové heslo. Po zadaní hesla, sa telnet démon zablokuje. Bez toho aby sa systém musel reštartovať, je od zadania hesla dostupný protokol SSH (Secure Shell) a HTTPS (Hypertext Transfer Protocol Secure) pre webové rozhranie LuCI. Je možné nastaviť sieťové parametre v konzoly (viz obr. 1.5), alebo pomocou webového rozhrania systému OpenWRT. Tieto parametre sa nachádzajú v nastavení siete, kam sa dá dostať pomocou príkazu:

```
vi etc/config/network
```

Pomocou príkazu `ping` v konzoly OpenWRT, sa dala overiť komunikácia systému s internetom. Pre pohodlnejšiu správu systému bol nainštalovaný balík pre grafické prostredie (LuCI).

```
opkg install luci luci-theme-openwrt
```

Vďaka tomuto nastaveniu sa dalo cez internetový prehliadač prihlásiť do grafického prostredia systému OpenWRT. Samozrejme sa dá systému OpenWRT prihlásiť aj pomocou SSH (šifrovaný prístup) alebo telnet (nešifrovaný prístup).

```
|root@Penzista:~# cat /etc/config/network

config interface 'loopback'
    option ifname 'lo'
    option proto 'static'
    option ipaddr '127.0.0.1'
    option netmask '255.0.0.0'

config interface 'lan'
    option type 'bridge'
    option proto 'static'
    option ipaddr '192.168.1.1'
    option netmask '255.255.255.0'
    option ifname 'eth0.1'

config switch
    option name 'eth0'
    option reset '1'
    option enable_vlan '1'
```

Obr. 1.5: Nastavenie siete v OpenWRT

1.5.1 Protokol SSH

Je softvérové riešenie sieťového zabezpečenia. Implementuje moderné šifrovacie algoritmy, môžeme ho teda považovať za bezpečný. SSH pracuje na porte číslo 22 a pre svoje spojenie využíva TCP (spoľahlivé/potvrdzované) spojenie. Je založené na architektúre klient – server. Slúži ako náhrada nezabezpečených protokolov (telnet, rsh, atď.). SSH iba vytvára bezpečný kanál, nie je interpretom príkazov. Taktiež ponúka služby spojené s autentizáciou, integritou a samozrejme so šifrovaním. Základnými vlastnosťami SSH protokolu sú:

- sprostredkovanie bezpečnej autentizácie medzi serverom a klientom, ako na základe znalostí (užívateľské meno a heslo), tak na základe vlastníctva (verejný a tajný kľúč),
- tvorba bezpečného kanála (šifrovanie),
- spúšťanie vzdialených príkazov cez bezpečný kanál,
- manažment kľúčov,
- riadenie prístupu bez nutnosti vlastníctva administrátorských oprávnení.

Verzie protokolu SSH1 a SSH2

Postupom času boli vyvinuté dve verzie SSH protokolu, SSH1 a SSH2. Zásadnou odlišnosťou je skutočnosť, že SSH1 je monolitický celok, oproti tomu SSH2 je skôr modulárny. SSH2 pozostáva z troch jednotlivých protokolov:

- protokol transportnej vrstvy,
- autentizačný protokol,
- spojovací protokol.

Ďalšie rozdiely v prospech pre SSH2 sú:

- rozšírené vyjednávanie algoritmov medzi klientom a serverom,
- viacej metód pre výmenu kľúčov,
- certifikáty pre verejné kľúče,
- väčšia flexibilita pri autentizácií, možná aj čiastočná autentizácia,
- silnejšia kontrola integrity vďaka kryptografickým metódam (CRC-32 vs. MAC a HMAC).
- pravidelná zmena kľúčov relácie.

1.5.2 Protokol Telnet (Telecommunications Network)

Patrí medzi sieťové protokoly. Má svoju serverovú a klientskú časť. Klienti sa pripájajú na port 23 zariadenia, kde je spustený telnet server. Typicky poskytuje prístup k príkazovému riadku servera, komunikácia ovšem prebieha cez počítačovú sieť.

Tri najzásadnejšie problémy, ktoré v priebehu doby očiernili protokol telnet, akoby vhodného prostriedku pre vzdialené prihlasovanie:

- bežné užívané telnet servery sú zraniteľné. Existujú rôzne programy schopné preniknúť skrz telnet server.
- Nie sú použité šifrovacie algoritmy, a to ani pri prenose hesla. Je tak veľmi jednoduché, odpočúvaním prenosu na sieti zistiť informácie nutné k prihláseniu na server a potom ich patrične zneužiť.
- Chýba autentizačný mechanizmus, vďaka čomu je možný útok zvaný „ukradnutie spojenia“ – nikdy nie je isté, či prebieha komunikácia priamo so zariadením, ku ktorému sme sa chceli pripojiť, alebo s niekým kto sa zaradil medzi vás a cieľový server.

Použitie telnetu sa pre vzdialené prihlasovanie k serveru nedoporučuje.

1.5.3 Protokol HTTPS (Hypertext Transfer Protocol Secure)

Je nadstavbou protokolu HTTP (Hypertext Transfer Protocol), ktorý je rozšírený o zabezpečenie pomocou protokolu SSL (Secure Socket Layer) alebo TLS (Transport Layer Security). HTTPS komunikuje pomocou protokolu TCP na porte 443. Pre fungovanie tohto zabezpečenia, sa musí vytvoriť zabezpečovací certifikát, ktorý musí byť podpísaný a overený, aby nedošlo k jeho zneužitiu. Bezpečnosť závisí na použitom šifrovacom algoritme a správnej implementácii do prehliadača.

Protokol SSL/TLS

Predchodcom protokolu TLS je protokol SSL. SSL je plne duplexným zabezpečovacím protokolom, používajúci architektúru klient-server. Ide o vrstvu, ktorá je vložená medzi aplikačnú a transportnú vrstvu. Zabezpečuje autentizáciu, integritu a šifrovanie dát. Pri nadväzovaní spojenia dochádza k autentizácii servera. Pri komunikácii klient – server sa používajú iné zabezpečujúce kľúče, ako pri komunikácii server – klient. Vlastnosti SSL/TLS:

- dochádza k autentizácii serveru.
- Pre autentizáciu sa využíva asymetrickej kryptografie na základe certifikátov.
- Pri komunikácii s neznámym serverom nedochádza k autentizácii klienta.
- Nedochádza k elektronickému podpisu dát.
- Dáta sú doplnené o kontrolný súčet.
- SSL/TLS využíva na šifrovanie správ symetrickú šifru (prenos šifrovacieho (privátneho) kľúča od klienta k serveru).

1.6 DHCP server v OpenWRT – Dnsmasq

Dnsmasq je odľahčený, ľahko konfigurovateľný DNS (Domain Name System) a DHCP server. Je navrhnutý tak, aby poskytoval DNS a DHCP server pre malé siete. Môže slúžiť na pomenovanie lokálnych počítačov, ktoré nie sú v globálnom DNS. DHCP server sa zlúči s DNS serverom a pridelené adresy DHCP serverom sa objavia v DNS servery, kde sa zlúčia s názvami nakonfigurovanými buď v každom hostiteľovi, alebo v centrálnom konfiguračnom súbore. Dnsmasq podporuje statické aj dynamické pridelovanie adries. V OpenWRT je Dnsmasq nainštalovaný a predkonfigurovaný. Konfigurácia sa dá skontrolovať v:

```
/etc/config/dhcp
```

1.7 Nastavenie LED diód na smerovači

OpenWRT ponúka pohodlnú správu LED diód na danom smerovači. Podľa toho ako to bude užívateľovi vyhovovať, môže využiť tieto diódy na rôzne signalizácie.

Device	<input type="text"/>
Trigger Mode	<input checked="" type="checkbox"/> Link On <input checked="" type="checkbox"/> Transmit <input checked="" type="checkbox"/> Receive

Name	<input type="text" value="USB"/>	<input type="button" value="Delete"/>
LED Name	<input type="text" value="soc:green:usb"/>	
Default state	<input type="checkbox"/>	
Trigger	<input type="text" value="usbdev"/>	
USB Device	<input type="text"/>	

Name	<input type="text" value="Phone1"/>	<input type="button" value="Delete"/>
LED Name	<input type="text" value="soc:green:phone1"/>	
Default state	<input type="checkbox"/>	
Trigger	<input type="text" value="phy0rx"/>	

Name	<input type="text" value="Phone2"/>	<input type="button" value="Delete"/>
LED Name	<input type="text" value="soc:green:phone2"/>	
Default state	<input type="checkbox"/>	
Trigger	<input type="text" value="phy0rx"/>	

Obr. 1.6: Nastavenie LED diód v LuCI OpenWRT

2 POBOČKOVÁ ÚSTREDŇA ASTERISK

Asterisk je softvérová pobočková telefónna ústredňa (PBX) s otvoreným kódom. Funguje na platformách Linux a Unix. Poskytuje možnosť pripojenia na technológie klasickej telefónie, ako sú analógové a ISDN (digitálna sieť integrovaných služieb) linky. Z VoIP (prenos hlasu po internete) technológie podporuje riešenie na báze SIP (Session Initiation Protocol) protokolu, kde môže pracovať ako SIP klient, proxy server aj ako registračný server. Ďalšou možnosťou je využitie protokolu IAX (InterAsterisk eXchange), ktorý bol navrhnutý s dôrazom na maximálnu jednoduchosť a minimálnu réžiu pri prenose signalizácie. Asterisk je zdarma a preto si nájde uplatnenie pri rôznych riešeniach (napr. malé/veľké podniky, volacie (call) centrá, ...).

2.1 Protokol SIP

Je najpoužívanejší protokol pre signalizáciu vo VoIP. Jedná sa o signalizačný protokol, ktorý zaistuje inicializáciu, zmenu a ukončenie interaktívnej multimediálnej komunikácie. Protokol SIP je možné použiť aj na posielanie krátkych správ. Podobá sa protokolom HTTP a SMTP (Simple Mail Transfer Protocol). Je založený na posielaní žiadostí a k nim relevantných odpovedí. Základné vlastnosti a možnosti protokolu SIP sú:

- **vyhľadávanie užívateľa** – pri inicializácii spojenia vyhľadá volaného. Podporovaná je taktiež mobilita užívateľa.
- **Nastavenie relácie** – možnosť nastaviť a stanoviť prostriedky pri zahájení relácie.
- **Zmena relácie** – behom relácie je možné reláciu modifikovať, presmerovať alebo ukončiť.
- **Stanovenie schopností užívateľa** – umožňuje špecifikovať vlastnosti jednotlivých zariadení (použitie audio/video kodeky, ...).
- **Dostupnosť užívateľa** – prenos informácií o dostupnosti užívateľa.

2.2 Konfigurácia Asterisk

Pri riešení tejto práce boli použité tri súbory (`sip.conf`, `extensions.conf` a `lantiq.conf`) pre nastavenie PBX Asterisk. Možnosti nastavenia funkcií súbora `lantiq.conf` je opísané nižšie (viz kapitola 4.2).

2.2.1 Sip.conf

Tento súbor slúži k správe SIP účtov v PBX Asterisk. A to nielen účtov, ktoré sa prihlasujú k PBX Asterisk, ale aj účtov ktorými sa PBX prihlasuje k ďalším ústredniam. V tomto súbore sa definovali všeobecné nastavenia pre všetky účty a následne sa nastavili jednotlivé užívateľské účty ([101] a [102]).

```
[general]                ;určuje spoločné nastavenia pre všetky účty
context=default          ;určuje kontext, „skupinu“ do ktorej užívateľ
                          patrí
bindport=5060            ;port na ktorom počúva SIP
videosupport=yes        ;podpora videa
disallow=all             ;zakazuje všetky kodeky
allow=alaw               ;povolíme iba vybrané
allow=gsm
allow=h261
allow=h263
language=cz              ;východzí jazyk, napr. pre chybové hlášky

[101]                    ;názov - užívateľský účet
type=friend              ;friend - autorizácie v oboch smeroch
secret=welcome           ;heslo
username=101             ;názov (najčastejšie tel. číslo)
callerid=PC              ;identifikácia volajúceho
host=dynamic             ;dynamicky získaná IP hosta registráciou
auth=md5                 ;autentizácia
context=default          ;určuje kontext, „skupinu“ do ktorého užívateľ
                          patrí

[102]
type=friend
secret=welcome
username=102
callerid=Mobile
host=dynamic
auth=md5
context=default
```


2.2.2 Extensions.conf

Je súbor pre nastavenie samotného smerovacieho plánu. Určuje akým spôsobom a ktorými cestami bude hovor spojený. Rozdeľuje sa na tri časti: general (obecné nastavenie), globals (globálne premenné, volajú sa pomocou znaku \$) a na jednotlivé kontexty. Telefóny v danom kontexte „nevidia“ na telefóny v inom kontexte. Aby bolo umožnené smerovanie hovorov z telefónov z jedného kontextu, na telefóny z iného kontextu. Musíme v danom kontexte hovory smerovať na nejaký externý server pomocou tzv. chobota (trunku). Na servery sa hovory presmerujú do daného kontextu. Toto riešenie je vhodné pre siete s veľkým počtom telefónnych prístrojov. Ďalším riešením je zahrnutie jedného kontextu do druhého kontextu. Vhodné pre hierarchické usporiadanie telefónnych prístrojov v smerovacom pláne. Ponúka možnosť nastavenia smerovania hovorov z jedného kontextu do druhého s tým, že nemusí byť umožnené smerovanie hovorov v opačnom smere. Posledné možné riešenie bolo použité aj v tejto práci. Využíva sa pri malom počte telefónnych prístrojov a spočíva v umiestnení všetkých telefónov do jedného kontextu. Všetci na seba „vidia“.

```
[general]
```

```
dtmfmode = rfc2833
```

```
static=yes
```

```
writeprotect=no ;umožňuje ukladať dialplan z CLI, pri nastavení NO  
autofallthrough=no
```

```
extenpatternmatchnew=no
```

```
clearglobalvars=no ;určuje či budú pri reloade dialplánu zmazané glo-  
bálne premenné
```

```
usercontext=default
```

```
[globals]
```

```
[default]
```

```
exten => 101,1,Dial(SIP/101) ;klapka, priorita, príkaz
```

```
exten => 101,n,Hangup()
```

```
exten => 102,1,Dial(SIP/102)
```

```
exten => 102,n,Hangup()
```

```
exten => 103,1,Dial(TAPI/1,60) ;klapka, priorita, príkaz, rozhranie,
```

```
exten => 103,n,Hangup() ; doba reakcie na zdvihnutie sluchátka
```

3 SOFTVÉROVÝ TELEFÓN ZOIPER

Pre overenie funkčnosti PBX Asterisk v OpenWRT, bol použitý softvérový telefón ZoiPer. Tento telefón ponúka na výber dva komunikačné protokoly SIP a IAX. Dá sa voľne stiahnuť verzia so základnými funkciami, ktorá postačovala pre overenie správneho fungovania PBX Asterisk v tejto práci. ZoiPer ponúka aj voľne stiahnuteľné aplikácie pre operačný systém iOS aj Android. Na softvérových telefónoch bolo potrebné vytvoriť a nastaviť dané telefónne účty. Nastavovala sa IP (internet protokol) adresa PBX Asterisk, názov daného telefónneho účtu, heslo pre tento účet a identifikácia volajúceho. Väčšina tých vlastností bola definovaná v `sip.conf`. Bolo uskutočnené volanie v operačnom systéme Windows medzi dvomi účtami. Na obrázku 3.1 je znázornené volanie medzi operačným systémom Windows a ZoiPer aplikáciou na mobilnom telefóne iPhone. Mobilný telefón bol pripojený na WiFi, ktorú rozhlasoval použitý smerovač so systémom OpenWRT.



Obr. 3.1: Volanie zo softvérového telefónu na operačnom systéme Windows a aplikáciou na mobilnom telefóne

4 TELEFÓNNE APLIKAČNÉ PROGRAMOVACIE ROZHRANIE (TAPI)

TAPI umožňuje počítačom s operačným systémom Windows, využívať telefónne služby. Rôzne verzie TAPI sú k dispozícii k rôznym verziám systému. TAPI sprístupňuje aplikáciám ovládanie telefónnych funkcií medzi počítačom a telefónnou sieťou pre dáta, fax a hovory. Zahŕňa základné funkcie, ako je napr. volanie, záznamník a polohovanie hovoru. Podporuje taktiež doplnkové funkcie, ako napr. držanie alebo prepojenie hovoru, konferenčný hovor a nájdenie „zaparkovaného“ hovoru v PBX, ISDN, alebo v inom telefónnom systéme. TAPI sa predovšetkým používa na ovládanie modemov alebo pobočkových ústrední. Ovládače pre PBX sú dodávané výrobcom. Existujú v dvoch prevedeniach. Prvá skupina ovládačov sa nazýva „third-party control“ (riadenie tretou stranou). Tieto ovládače povoľujú aplikáciám sledovanie a/alebo riadenie viacerých spojení súčasne. Druhá skupina ovládačov sa nazýva „first-party control“ (riadenie prvou stranou). Ako už z názvu vyplýva, slúžia na sledovanie a/alebo riadenie jedného spojenia súčasne. Telefónne systémy prirodzene umožňujú prepojenie niekoľkých hovorov súčasne. Modemy odjakživa spadali do skupiny riadenia prvou stranou.

4.1 Verzie TAPI

Prvá verejne dostupná verzia TAPI bola verzia 1.3. Podporovala iba 16-bitové aplikácie. S MS (Microsoft) Windows 95 bolo TAPI integrované do operačného systému. Prvá verzia na Windows 95 bola 1.4, ktorá podporovala 32-bitové aplikácie. Verzia TAPI 2.0 urobila významný krok vpred s podporou automatickej distribúcie hovorov a špecifických funkcií PBX. TAPI 3.0 bola uvedená na trh spolu s Windows 2000. Táto verzia podporuje VoIP s tým, že poskytuje jednoduché a všeobecné metódy pre vytváranie spojenia medzi dvomi (pomocou H.323¹), alebo medzi viacerými počítačmi (použitím skupinového vysielania IP datagramov). Verzie TAPI 3.x a 2.x nie sú kompatibilné. Čiže verzie TAPI 3.x nemôžu slúžiť ako náhrada za verzie 2.x.

¹doporučenie, ktoré definuje protokoly pre audio-vizuálne relácie komunikácie v akejkoľvek paketovej sieti.

4.2 Nastavenie TAPI v OpenWRT pomocou PBX Asterisk

Pred samotným nastavením TAPI rozhrania je nutné do systému OpenWRT nainštalovať príslušné balíčky:

- **kmod-ltq-vmmc** – ovládač napätia pre hlasový podsystem pre čip danube,
- **kmod-ltq-tapi** – ovládače pre TAPI rozhranie,
- **kmod-ltq-ixos** – knižnica pre OS (operačný systém) lantiq,
- **asterisk18-chan-lantiq** – balíček pre TAPI rozhranie a nastavenie tohto rozhrania v PBX asterisk,
- **asterisk18-codec-alaw** — balíček pre kodek alaw.

Všetky premenné pomocou ktorých je TAPI ovládané sú uložené v súbore `chan_lantiq.c`. Úpravou tohto súboru sa dá prispôsobiť TAPI podľa potrieb užívateľa. Po každej úprave je nutné daný balíček (`asterisk18-chan-lantiq_1.8.10.1-2_lantiq.ipk`) skompilovať a nahráť do systému OpenWRT. Podobne, ako `sip.conf` môžeme nastaviť aj `lantiq.conf`. Avšak `lantiq.conf` má viac obecnjšie nastavenie oproti `sip.conf`.

```
[interfaces]
channels = 2      ;nastavenie počtu analógových portov.
```

```
[general]
echocancel = es  ;potlačenie ozveny na kanále.
```

Ďalej tento súbor umožňuje nastaviť kolísanie oneskorenia (jitter), detekciu hlasovej aktivity podľa príslušného kodeku a nastavenie identifikácie volajúceho (callerID).

4.2.1 Funkcie v súbore `chan_lantiq.c`

Pri uskutočňovaní hovoru z jedného TAPI rozhrania (TAPI/1) na druhé (TAPI/2) (viz obr 5.6). Je vidno, že jedna z funkcií, ktorá sa volá a vypisuje daný stav do konzole v PBX Asterisk zo súboru `chan_lantiq.c` je funkcia²:

```
static void lantiq_dial(struct lantiq_pvt *pvt)
{
    struct ast_channel *chan = NULL;
    if (ast_exists_extension(NULL, pvt->context, pvt->dtmfbuf, 1, NULL))
    {
```

²Daná funkcia nie je kompletná. Slúži iba na ukážku.

```

strcpy(pvt->ext, pvt->dtmfbuf);

ast_verbose(VERBOSE_PREFIX_3 "Daná klapka existuje, PBX vytáča %s
\n",pvt->ext);

chan = lantiq_channel(AST_STATE_UP, pvt->port_id, pvt->ext+1,
pvt->context);

chan->tech_pvt = pvt;
pvt->owner = chan;
strcpy(chan->exten, pvt->ext);
ast_setstate(chan, AST_STATE_RING);
pvt->channel_state = INCALL;
pvt->call_setup_start = now();

:
}

```

Kde je:

- `lantiq_pvt` – statická štruktúra kde sú definované premenné pre číslo portu, stav kanálu, sekvenčné číslo pre RTP (Real-time Transport Protocol) pakety, atď.,
- `dtmfbuf` – vyrovnávacia pamäť pre volané číslo,
- `ast_channel` – kanál do ktorého telefón patrí (definovaná premenná v `lantiq_pvt`),
- `ast_verbose` – výpis do konzole asterisku,
- `ext` – (smerovanie) port ktorý sa pripája,
- `call_setup_start` – začiatok vytáčania v ms.

Je definovaných sedem stavov, v ktorých sa môže nachádzať daný kanál:

1. ONHOOK – zavesené sluchátko,
2. OFFHOOK – vyvesené sluchátko,
3. DIALING – vytáčanie,
4. INCALL – prebieha hovor,
5. CALL_ENDED – hovor bol ukončený,
6. RINGING – zvoní,
7. UNKNOWN – neznámy.

Príklad definovania stavu **ONHOOK** vo funkcii `ast_lantiq_call`:

```
if (pvt->channel_state == ONHOOK) {
    ast_log(LOG_DEBUG, "port %i zvoní\n", pvt->port_id);

    char *cid = ast->connected.id.number.valid ? ast->connected.id.num-
        ber.str : NULL;
    ast_log(LOG_DEBUG, "port %i ID užívateľa: %s\n", pvt->port_id, cid
        ? cid : "none");

    lantiq_ring(pvt->port_id, 1, cid);
    pvt->channel_state = RINGING;

    ast_setstate(ast, AST_STATE_RINGING);
    ast_queue_control(ast, AST_CONTROL_RINGING);
}
```

Pre identifikovanie čo sa práve deje na kanály, slúži funkcia `ast_lantiq_indicate`. Táto funkcia vypisuje do konzoly PBX Asterisk kontrolné hlášky stavu kanála. Tak-
tiež tieto hlášky je možno vidieť na obr. 5.6. Nachádzajú sa za slovným spojením
„telefón indikuje, že“. Príklad možných hlášok pri daných stavoch:

```
switch (c) {

case AST_CONTROL_HANGUP: return "Druhý koniec ukončil hovor";
case AST_CONTROL_RING: return "Miestne volanie";
case AST_CONTROL_RINGING: return "Vzdialený koniec vyzváňa";

:
}
```

Ďalšie možné hlášky sú vypísané v prílohe B.

4.2.2 Významné rozhrania pre TAPI

Pre spustenie alebo zastavenie generovania oznamovacieho tónu na miestne porty TAPI používa rozhranie:

```
#define IFX_TAPI_TONE_LOCAL_PLAY _IOW (IFX_TAPI_IOC_MAGIC, 0x9B,  
IFX_int32_t)
```

Ak je parameter v tomto rozhraní väčší ako 0, tak ukazuje v tabuľke tónov na index tónu, ktorý bol hraný. Ak je parameter rovný 0, tak sa zastaví generovanie oznamovacieho tónu. Parameter `IFX_int32_t` udáva index tónu, ktorý bol preddefinovaný v tabuľke tónov (rozsah 1 – 31), alebo vlastný preddefinovaný tón (rozsah 32 – 255). Rozsah 1 – 31 je preddefinovaný ovládačom a vzťahuje sa k originálnemu TAPI.

Parameter `IFX_TAPI_IOC_MAGIC` udáva „magické“ číslo, pre riadenie vstupu a výstupu.

Toto rozhranie spustí neblokujúce vyzvanie na telefónnej linke pomocou predkonfigurovaného zvoniaceho rytmu.

```
#define IFX_TAPI_RING_START          _IO (IFX_TAPI_IOC_MAGIC,  
0x87)
```

Toto rozhranie neposkytuje službu identifikácie volajúceho (caller ID). Ak je potrebná táto služba je nutné použiť rozhranie `IFX_TAPI_CID_TX_SEQ_START`

Vopred definovaná tónová služba pre obsadzovací tón je reprezentovaná rozhraním:

```
#define IFX_TAPI_TONE_BUSY_PLAY      _IO (IFX_TAPI_IOC_MAGIC,  
0xA1)
```

Pri tomto rozhraní sa neočakáva žiadny parameter. Vždy by malo byť nastavené na 0. Tieto tóny sú definované frekvenciami vo vnútri TAPI definovaného pre USA (United States of America).

Ďalšou preddefinovanou tónovou službou je tón pre vytočenie čísla:

```
#define IFX_TAPI_TONE_DIALTONE_PLAY  _IO (IFX_TAPI_IOC_MAGIC,  
0xA3)
```

Rovnako ako pri rozhraní pre obsadzovací tón sa tu neočakáva žiadny parameter a hodnota iná ako 0. Tóny sú taktiež definované frekvenciami vo vnútri TAPI pre USA.

Rozhranie, ktoré zastaví generovanie tónov je:

```
#define IFX_TAPI_TONE_STOP           _IOW (IFX_TAPI_IOC_MAGIC,  
0xA4, IFX_int32_t)
```

Nastaví index parametera(`IFX_int32_t`) na 0. Nastavením indexu na 0 budú zastavené prehrávania všetkých tónov vo všetkých spojeniach.

Pre nastavenie hlasitosti modulu analógovej linky pre prichádzajúce a odchádzajúce hlasové cesty sa používa rozhranie:

```
#define IFX_TAPI_PHONE_VOLUME_SET          _IOW (IFX_TAPI_IOC_MAGIC,  
0x42, IFX_TAPI_LINE_VOLUME_t)
```

Jedná sa o nastavenie hlasitosti mikrofónu a reproduktora.

Pre nastavenie vyrovnávacej pamäte jittera (kolísania oneskorenia) slúži rozhranie:

```
#define IFX_TAPI_JB_CFG_SET                _IOW (IFX_TAPI_IOC_MAGIC,  
0x17, IFX_TAPI_JB_CFG_t)
```

Vyberie pevne alebo prispôsobí vyrovnávaciu pamäť jittera a nastaví potrebné parametre. Modifikuje nastavenia vyrovnávacej pamäte jittera počas behu.

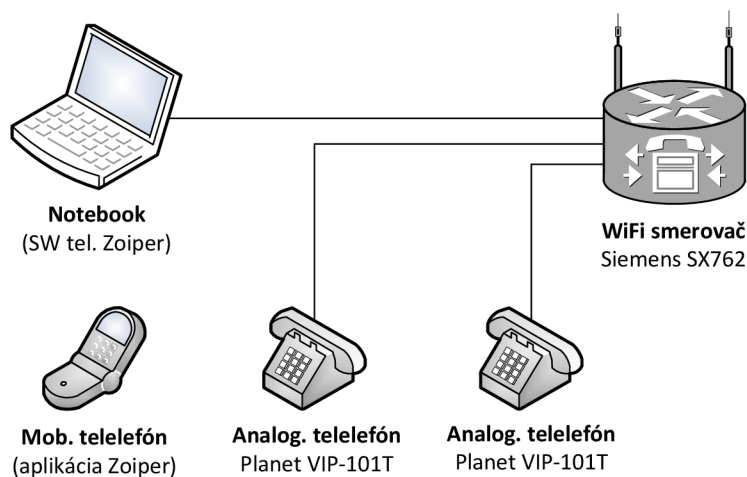
Toto rozhranie pripojí telefónny kanál na iný telefónny kanál (tzv. mapovanie).

```
#define IFX_TAPI_MAP_PHONE_ADD             _IOW (IFX_TAPI_IOC_MAGIC,  
0x2A, IFX_TAPI_MAP_PHONE_t)
```

Užívateľ sa musí postarať o riadenie zdrojov.

5 TESTOVANÁ TOPOLOGIA

Topológia bola navrhnutá tak, aby bolo na nej možné demonštrovať volanie nielen medzi analógovými ale aj IP telefónmi.



Obr. 5.1: Schéma testovanej topológie

Analógové telefóny komunikovali pomocou rozhrania TAPI, ktoré bolo opísané vyššie. IP telefóny boli v tomto prípade reprezentované SW telefónmi Zoiper. Použitý smerovač v tejto práci vedel vytvoriť WiFi sieť, vďaka ktorej nerobilo žiadny problém pripojenie bezdrôtových zariadení so SW telefónmi.

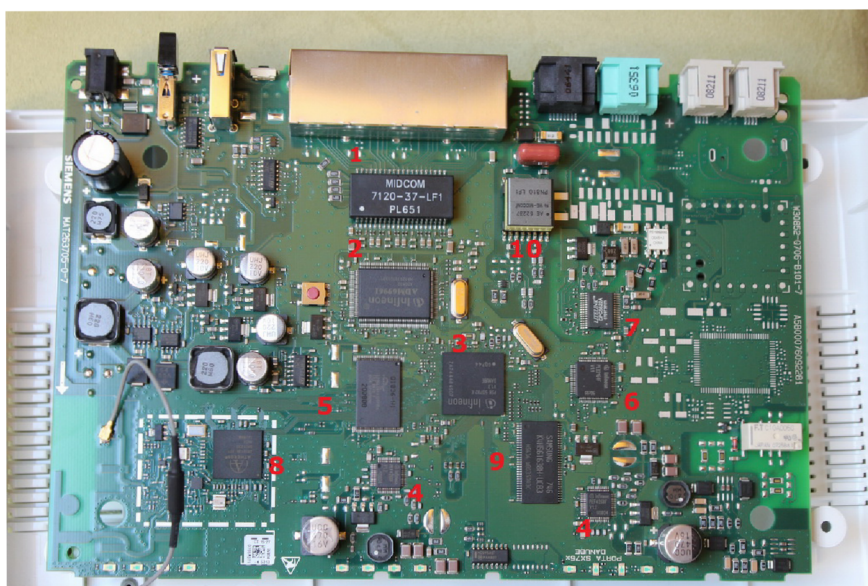
5.1 Smerovač Siemens Gigaset SX762

V tejto práci bol použitý smerovač s lantiqkovským jadrom. Daný smerovač bol vybavený štyrmi LAN (lokálna sieť) portami (jeden z nich plní aj funkciu WAN (sieť na veľkom geografickom území) portu), dvomi portami pre pripojenie analógových telefónov (RJ-11), ADSL (Asymmetric Digital Subscriber Line) portom, portom pre pripojenie telefónnej linky cez rozbočovač a USB portom (viz obr. 5.3). Už ako bolo spomenuté vyššie dokázal vytvoriť vlastnú WiFi sieť.

Čipy na smerovači (viz obr. 5.2):

1. transformátor pre sieť LAN; 10/100 BASE-T.
2. LAN prepínač.

3. Čip (danube) pre ADSL2/2+ s integrovanými VoIP DSP (procesor pre spracovanie digitálneho signálu) riešením pre domáce brány. Pripojenie analógových telefónov.
4. Adaptér pre analógový telefón.
5. Flash pamäť.
6. Čip pre prístupový hlasový kodek.
7. Čip na ovládanie faxu, PBX, hlasových odkazov (voicemail systems), VoIP.
8. Čip na ovládanie WiFi.
9. DRAM (druh pamäti) čip.
10. Transformátor xDSL signálu.



Obr. 5.2: Doska smerovača



Obr. 5.3: Smerovač Siemens Gigaset SX762

5.2 Telefón Planet VIP-101T

Tento telefón umožňuje komunikáciu cez VoIP ale aj cez analógové rozhranie. V tejto práci boli použité dva telefóny tohto druhu ako analógové telefóny. Pri tomto použití sa nemuseli telefóny napájať pomocou napájacieho adaptéru. Napájanie bolo vyriešenie cez port RJ-11. V tomto režime sa telefón tvári, že je vypnutý, ale dokáže volať a prijímať hovory.



Obr. 5.4: Telefón Planet VIP-101T

5.3 Overenie funkčnosti topológie

Na obr. 5.5 je vidno reálne zapojenie danej schémy. Zrovna na danom obrázku je uskutočnené volanie z TAPI rozhrania 1 na TAPI rozhranie 2. A zároveň je uskutočnené volanie z mobilnej aplikácie (ZoiPer) na softvérový telefón (ZoiPer) nainštalovaný v operačnom systéme windows. Boli testované všetky možné kombinácie volaní z každého telefónu. Čo je znázornené vo výpise smerovania hovor v konzoly PBX Asterisk (viz obr. 5.6).



Obr. 5.5: Fyzické zapojenie testovanej topológie

```

192.168.1.1 - PuTTY
-- TAPI/1 is ringing
== Spawn extension (default, 103, 1) exited non-zero on 'SIP/102-00000021'
-- Using SIP RTP CoS mark 5
-- Executing [104@default:1] Dial("SIP/102-00000022", "TAPI/2,60") in new stack
-- Called TAPI/2
-- TAPI/1 is ringing
-- Spawn extension (default, 104, 1) exited non-zero on 'SIP/102-00000022'
-- Daná klapka existuje, FBX vytáča 104
-- Executing [104@default:1] Dial("TAPI/1", "TAPI/2,60") in new stack
called TAPI/2
-- TAPI/1 is ringing
-- telefón indikuje že "Vzdialený koniec vyzváňa"
== Spawn extension (default, 104, 1) exited non-zero on 'TAPI/1'
-- Daná klapka existuje, FBX vytáča 102
-- Executing [102@default:1] Dial("TAPI/1", "SIP/102") in new stack
Using SIP RTP CoS mark 5
-- Called SIP/102
-- telefón indikuje že "Pripojenie linky"
-- SIP/102-30000023 is ringing
-- telefón indikuje že "Vzdialený koniec vyzváňa"
== Spawn extension (default, 102, 1) exited non-zero on 'TAPI/1'
-- Daná klapka existuje, FBX vytáča 101
-- Executing [101@default:1] Dial("TAPI/1", "SIP/101") in new stack
Using SIP RTP CoS mark 5
-- Called SIP/101
-- telefón indikuje že "Pripojenie linky"
-- SIP/101-30000024 is ringing
-- telefón indikuje že "Vzdialený koniec vyzváňa"
Apr 1 02:33:43] WARNING[1350]: db.c:115 dbinit: Unable to open Asterisk database '/var/lib/asterisk/astdb': No such file or directory
== Spawn extension (default, 101, 1) exited non-zero on 'TAPI/1'
-- Daná klapka existuje, FBX vytáča 101
-- Executing [101@default:1] Dial("TAPI/1", "SIP/101") in new stack
== Using SIP RTP CoS mark 5
-- Called SIP/101
-- telefón indikuje že "Pripojenie linky"
-- SIP/101-30000025 is ringing
-- telefón indikuje že "Vzdialený koniec vyzváňa"
== Spawn extension (default, 101, 1) exited non-zero on 'TAPI/1'
Apr 1 02:34:13] WARNING[1350]: db.c:115 dbinit: Unable to open Asterisk database '/var/lib/asterisk/astdb': No such file or directory
== Using SIP RTP CoS mark 5
-- Executing [101@default:1] Dial("SIP/102-00000026", "SIP/101") in new stack
== Using SIP RTP CoS mark 5
-- Called SIP/101
-- SIP/101-30000027 is ringing
-- Daná klapka existuje, FBX vytáča 104
-- Executing [104@default:1] Dial("TAPI/1", "TAPI/2,60") in new stack
-- Called TAPI/2
-- TAPI/1 is ringing
-- telefón indikuje že "Vzdialený koniec vyzváňa"
penzista*CLI>

```

Obr. 5.6: Zobrazenie smerovania hovorov medzi zariadeniami v PBX Asterisk

6 ZÁVER

V práci sme sa stručne zoznámili so systémom OpenWRT, pobočkovou ústredňou Asterisk a rozhraním TAPI. Systém OpenWRT bol nainštalovaný na smerovač Siemens Gigaset SX762. Pobočková ústredňa bola nainštalovaná a spustená pod systémom OpenWRT. V pobočkovej ústredni boli nakonfigurované dva telefónne SIP účty, ktoré ako už z názvu vyplýva pre komunikáciu používali protokol SIP. V smerovacom pláne PBX Asterisk boli okrem smerovacích plánov pre SIP účty pridané aj smerovacie plány pre analógové telefóny, ktoré komunikovali pomocou TAPI rozhrania. Overenie funkčnosti PBX Asterisk bolo uskutočnené hovorom medzi dvomi softvérovými telefónmi ZoiPer. Jeden softvérový telefón bol nainštalovaný na operačnom systéme MS Windows 8 a druhý bol nainštalovaný v podobe aplikácie na mobilný telefón, ktorý používa operačný systém iOS.

Práca sa zaoberala aj kompiláciou a inštaláciou jednoduchého balíčka *helloworld*. Balíček mal za úlohu vypísať jednoduché slovné spojenie „Hello World“ do konzoly systému OpenWRT. Správna inštalácia a následné spustenie balíčka je zobrazené na obrázku 1.2. Príklad kompilácie a inštalácie vlastného balíčka bol použitý iba pre ukážku, že systém OpenWRT je naozaj otvorený a dovoľuje užívateľovi si ho prispôbiť podľa seba.

Dôležitou súčasťou tejto práce bolo nastavenie rozhrania TAPI. Pomocou tohto rozhrania komunikovali analógové telefóny s PBX Asterisk. V práci boli spomenuté niektoré významné funkcie a rozhrania ktoré TAPI používa. Taktiež boli spomenuté aj potrebné balíčky, ktoré je nutné doinštalovať do systému OpenWRT, aby samotné TAPI rozhranie komunikovalo so systémom a pobočkovou ústredňou Asterisk.

Možným pokračovaním práce by mohlo byť využitie USB portu na smerovači. Do tohto portu by sa dal pripojiť 3G modem. V podstate je to GSM (Global System for Mobile Communications) telefón bez klávesnice, reproduktoru, batérie a displeja. A pomocou neho by bolo možné smerovať hovory z IP a analógových telefónov pripojených do smerovača cez sieť GSM.

LITERATÚRA

- [1] BEDNAŘÍK J.: *Analýza bezpečnosti síťových protokolů*. Brno, 2009. Bakalárska práca. FEKT Vysoké učení technické v Brně. Vedoucí práce ING. Jan Hajný. Dostupné z URL: <https://dspace.vutbr.cz/bitstream/handle/11012/11665/BK_Analyza_bezpecnosti.pdf?sequence=1>.
- [2] ČÍKA P.: *Multimediální služby: Internetová telefonie*. Brno. Skripta. FEKT Vysoké učení technické v Brně.
- [3] LOREMA: *About OpenWRT*. OpenWRT Wiki [online]. Posledná aktualizácia 13.9.2013 [cit. 28.10.2013]. Dostupné z URL: <<http://wiki.openwrt.org/about/start>>.
- [4] LOREMA: *Dnsmasq*. OpenWRT Wiki [online]. Posledná aktualizácia 11.7.2013 [cit. 27.4.2014]. Dostupné z URL: <<http://wiki.openwrt.org/doc/howto/dhcp.dnsmasq>>.
- [5] GRABELL: *Installing OpenWRT*. OpenWRT Wiki [online]. Posledná aktualizácia 8.9.2013 [cit. 29.4.2014]. Dostupné z URL: <<http://wiki.openwrt.org/doc/howto/generic.flashing>>.
- [6] MOLNÁR K., SOUMAR M.: *Praktikum z informačních sítí: Protokol Telnet*. Brno. Skripta. FEKT Vysoké učení technické v Brně.
- [7] RENNEBARTH N.: *Creating packages*. OpenWRT Wiki [online]. Posledná aktualizácia 5.12.2013 [cit. 23.12.2013]. Dostupné z URL: <[http://wiki.openwrt.org/doc/devel/packages?s\[\]=makefile](http://wiki.openwrt.org/doc/devel/packages?s[]=makefile)>.
- [8] ROSENBERG, M.: *Návrh, správa a bezpečnost počítačových sítí - cvičení: SSH - SECURE SHELL*. Brno, 2013. Skripta ke cvičením. FEKT Vysoké učení technické v Brně.
- [9] SPACEKPE: *OpenWrt - First Login*. OpenWRT Wiki [online]. Posledná aktualizácia 20.10.2013 [cit. 28.4.2014]. Dostupné z URL: <<http://wiki.openwrt.org/doc/howto/firstlogin>>.
- [10] ŠILHAVÝ, P.: *PBX Asterisk - Úvod, SIP, IAX, DAHDI, jednoduchý číslovací plán: Konfigurační soubory*. Brno. Laboratórne cvičenie. FEKT Vysoké učení technické v Brně.

- [11] *OpenWrt, využívame router a internetové pripojenie na maximum: Verzie a webové rozhranie*. DLS.SK. Digitálny Svet pod Lupou [online]. 20. 9. 2007 [cit. 6. 11. 2013]. Dostupné z URL: <<http://www.dsl.sk/article.php?article=4552> >.
- [12] *Step By Step Guide for Starting "Hello, World!ön OpenWRT*. Componentality [online]. Lappeenranta, Finland [cit. 23. 11. 2013]. Dostupné z URL: <www.componentality.com/res/Step-By-Step-Instruction-To-Run-Apps-On-FlexRoad-HW.en.pdf>.
- [13] *Telephony Application Programming Interface*. Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 23. 3. 2002, 17. 9. 2013 [cit. 23. 12. 2013]. Dostupné z URL: <http://en.wikipedia.org/wiki/Telephony_Application_Programming_Interface>.

ZOZNAM SYMBOLOV, VELIČÍN A SKRATIEK

ADSL (Asymmetric Digital Subscriber Line) asymetrická digitálna účastnícka linka

Atď. a tak ďalej

DHCP (Dynamic Host Configuration Protocol) protokol pre automatické pridelenie IP adresy

DNS (Domain Name System) systém doménových mien

DRAM (Dynamic Random Access Memory) druh počítačovej pamäte, ktorá uchováva dáta v podobe elektrického náboja v kondenzátore

DSP (Digital Signal Processor) procesor pre spracovanie digitálneho signálu

GSM (Global System for Mobile Communications) globálny systém mobilných komunikácií

HTTP (Hypertext Transfer Protocol) hypertextový prenosový protokol

HTTPS (Hypertext Transfer Protocol Secure) zabezpečený (šifrovaný) hypertextový prenosový protokol

GNU (General Public License) licencia pre slobodný softvér (celý názov GNU GPL)

IAX (InterAsterisk eXchange) komunikačný protokol systému Asterisk

ISDN (Integrated Services Digital Network) digitálna sieť integrovaných služieb

JTAG (Joint Test Action Group) umožňuje čítať registre, pamäť a podobne z procesora

LED (Light-Emitting Diode) dióda emitujúca svetlo

MAC (Message Authentication Code) autentizačný kód správy

HMAC (Hash Message Authentication Code) Hašovaný autentizačný kód správy

Napr. napríklad

OS operačný systém

PBX (Private Branch Exchange) súkromná pobočková ústredňa

RTP (Real-time Transport Protocol) protokole pre prenos dát v reálnom čase

SIP (Session Initiation Protocol) signalizačný protokol

SMTP (Simple Mail Transfer Protocol) protokol pre prenos elektronickej pošty

SSH (Secure Shell) zabezpečený (šifrovaný) komunikačný protokol

SSL (Secure Socket Layer) vrstva bezpečných socketov

TAPI (Telephony Application Programming Interface) Telefónne aplikačné programovacie rozhranie

TCP (Transmission Control Protocol) protokol pre spoľahlivý prenos

Telnet (Telecommunications Network) nezabezpečený komunikačný protokol

TLS (Transport Layer Security) predchodca SSL: protokoly, ktoré slúžia na šifrovanie dát

tzv takzvaný

VoIP (Voice over Internet Protocol) prenos hlasu po internete

WAN (Wide area network) sieť na veľkom geografickom území

USA (United States of America) spojené štáty americké

USB (Universal Serial Bus) univerzálna sériová zbernica

ZOZNAM PRÍLOH

A	Makefile pre balíček Helloworld	43
B	Prepínač stavov v PBX Asterisk na rozhraní TAPI	44

A MAKEFILE PRE BALÍČEK HELLOWORLD

```
include $(TOPDIR)/rules.mk
PKG_NAME:=helloworld
PKG_VERSION:=1.0.0
PKG_RELEASE:=1

PKG_SOURCE:=$(PKG_NAME)-$(PKG_VERSION).tar.xz
PKG_BUILD_DIR:=$(BUILD_DIR)/$(PKG_NAME)-$(PKG_VERSION)

include $(INCLUDE_DIR)/package.mk

define Package/$(PKG_NAME)
SECTION:=utils
CATEGORY:=Utilities
TITLE:=$(PKG_NAME)
endef

define Package/$(PKG_NAME)/description
    Toto je hello world balíček ktorý slúži na jednoduché vypísanie
    slovného spojenia "Hello World" do konzole.
endef

define Package/helloworld/Build/Compile
    $(MAKE) -C $(PKG_BUILD_DIR) helloworld
endef

CONFIGURE_VARS+= \
CC="$(TOOLCHAIN_DIR)/bin/$(TARGET_CC)"

define Package/$(PKG_NAME)/install
$(INSTALL_DIR) $(1)/usr/bin
$(CP) $(PKG_BUILD_DIR)/$(PKG_NAME) $(1)/usr/bin
endef

$(eval $(call BuildPackage,$(PKG_NAME)))
```

B PREPÍNAČ STAVOV V PBX ASTERISK NA ROZHRAŇÍ TAPI

V prílohe sú zobrazené možné stavy pri nadväzovaní hovoru pomocou TAPI rozhrania. Text z daného prepínača sa vypisuje v PBX Asterisk.

```
static const char *control_string(int c)
{
switch (c) {
case AST_CONTROL_HANGUP: return "Druhý koniec ukončil hovor";
case AST_CONTROL_RING: return "Miestne volanie";
case AST_CONTROL_RINGING: return "Vzdialený koniec vyzvára";
case AST_CONTROL_ANSWER: return "Vzdialený koniec odpovedá";
case AST_CONTROL_BUSY: return "Vzdialený koniec je zaneprázdnený";
case AST_CONTROL_TAKEOFFHOOK: return "Uistite sa či je vyvesené slu-
chátko";
case AST_CONTROL_OFFHOOK: return "Linka je vyradená";
case AST_CONTROL_CONGESTION: return "Preťaženie (obvod je zanepráz-
dnený)";
case AST_CONTROL_FLASH: return "Obdržanie spojenia";
case AST_CONTROL_WINK: return "Bliknutie/signál";
case AST_CONTROL_OPTION: return "Nastavenie voľby nízkeho levelu";
case AST_CONTROL_RADIO_KEY: return "Kontrola rádiotelefonických za-
riadení";
case AST_CONTROL_RADIO_UNKEY: return "Nekontrolovanie rádiotelefon-
ických zariadení";
case AST_CONTROL_PROGRESS: return "Vzdialený koniec sa vyvíja";
case AST_CONTROL_PROCEEDING: return "Vydialený koniec spracúva in-
formácie";
case AST_CONTROL_HOLD: return "Podržanie hovoru";
case AST_CONTROL_UNHOLD: return "Priятие hovoru";
case AST_CONTROL_SRCUPDATE: return "Aktualizácia medií";
case AST_CONTROL_CONNECTED_LINE: return "Pripojenie linky";
case AST_CONTROL_REDIRECTING: return "Presmerovanie";
case AST_CONTROL_INCOMPLETE: return "Neúplné zadanie";
case -1: return "Ukončenie tónu";
default: return "Neznámy";
}
}
```