

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

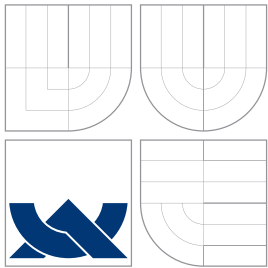
BEZDRÁTOVÉ OVLÁDÁNÍ PREZENTACÍ POMOCÍ
KAPESNÍHO POČÍTAČE

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

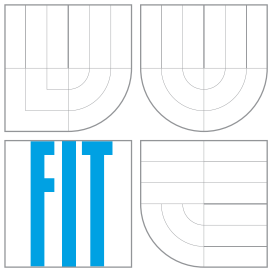
AUTOR PRÁCE
AUTHOR

JAN LIŠKA

BRNO 2007



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

BEZDRÁTOVÉ OVLÁDÁNÍ PREZENTACÍ POMOCÍ KAPESNÍHO POČÍTAČE

WIRELESS CONTROL OF PRESENTATIONS BY POCKET COMPUTER

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

JAN LIŠKA

VEDOUCÍ PRÁCE

SUPERVISOR

ING. PETR NOVOSAD

BRNO 2007

Zadání

Bezdrátové ovládání prezentací pomocí kapesního počítače

1. Seznamte se s mobilní platformou Pocket PC a s tvorbou programů pro příslušné kapesní počítače.
2. Navrhněte způsob komunikace mezi stolním a kapesním počítačem tak, aby bylo možné z kapesního počítače bezdrátově ovládat základní funkce prezentačních programů spuštěných na stolním počítači.
3. Navržené aplikace pro stolní a kapesní počítač implementujte.
4. Zhodnoťte dosažené výsledky a naznačte možné pokračování projektu.

Kategorie: Počítačové sítě

Licenční smlouva

Licenční smlouva je uložena v archivu Fakulty informačních technologií Vysokého učení technického v Brně.

Abstrakt

Cílem projektu je navrhnout a realizovat aplikaci pro PDA, která osvobodí osobu provádějící prezentaci z notebooku nebo i stolního počítače od nutnosti ovládat prezentaci z klávesnice. Taková osoba totiž běžně stojí daleko od plátna nebo ještě hůře neustále přechází od klávesnice k plátnu a zpět. Takovéto ovládání umožní prezentujícímu lepší kontakt s posluchači. Dokument obsahuje krátké seznámení s kapesními počítači a tvorbou programů pro tato zařízení. V hlavní části se zabývám návrhem a realizací aplikace pomocí architektury klient-server. Na závěr uvádím návrhy pro další vývoj aplikace.

Klíčová slova

PDA, Pocket PC, Kapesní počítač, Prezentace, PowerPoint, Microsoft Visual Studio, C/C++, WinAPI, Microsoft Windows, Windows CE, Architektura klient-server, Bluetooth

Abstract

This work deals with design and realization of an application for Pocket PC which controls presentation on notebook or desktop PC. The goal of this project is to free person from staying behind the computer during presentation. With this remote controller this person can move freely in the whole room and can be in better contact with audience. This document consists of short introduction to Pocket PCs and development tools for this devices. The main part describes concept and implementation of this application by client-server architecture. In the end there are suggestions for future development of this application.

Keywords

PDA, Pocket PC, presentation, PowerPoint, Microsoft Visual Studio, C/C++, WinAPI, Microsoft Windows, Windows CE, Client-server architecture, Bluetooth

Citace

Jan Liška: Bezdrátové ovládání prezentací pomocí kapesního počítače, bakalářská práce, Brno, FIT VUT v Brně, 2007

Bezdrátové ovládání prezentací pomocí kapesního počítače

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Ing. Petra Novosada

.....

Jan Liška
15. května 2007

© Jan Liška, 2007.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1	Úvod	3
2	PDA	4
2.1	Charakteristika PDA	4
2.2	Platformy na PDA	5
2.3	Tvorba programů pro Pocket PC	5
2.3.1	Visual Studio .NET 2003	5
2.3.2	eMbedded Visual C++ 3.0	6
2.3.3	eMbedded Visual C++ 4.0	6
2.3.4	Jazyk C#	6
2.3.5	Jazyk C/C++	6
3	Analýza problému	8
3.1	Motivace	8
3.2	Funkční požadavky	8
3.3	Architektura klient - server	9
3.4	Klientská část - PDA	9
3.5	Serverová část - PC	9
3.6	WinAPI	9
3.6.1	Rozhraní API a paměťové modely	9
3.6.2	Komponenty WinAPI	10
3.6.3	Možnosti jazyka	10
3.6.4	Programovací prostředí	10
3.7	Bluetooth	11
3.7.1	Specifikace Bluetooth	11
3.7.2	Programové řešení Bluetooth	11
4	Návrh	12
4.1	Komunikace mezi stolním počítačem a PDA	12
4.2	Klientská část - PDA	12
4.2.1	Zahájení komunikace	12
4.2.2	Odeslání zprávy	13
4.2.3	Uživatelské rozhraní	13
4.3	Serverová část - PC	14
4.3.1	Zahájení komunikace	14
4.3.2	Uživatelské rozhraní	14
4.3.3	Provádění příkazů	14

5 Implementace	15
5.1 Klient	15
5.1.1 Použité vývojové prostředí	15
5.1.2 První implementace	15
5.1.3 Výsledná implementace	16
5.1.4 Některé důležité funkce	19
5.2 Server	20
5.2.1 Použité vývojové prostředí	20
5.2.2 Řešení	20
5.2.3 Některé důležité funkce	20
6 Podobné produkty	22
6.1 Hardwarová řešení	22
6.2 Softwarová řešení	23
6.3 Srovnání	23
7 Budoucí vývoj	24
8 Závěr	25
Seznam použitých zdrojů	26
Seznam příloh	27
A Obsah CD	28
B Uživatelská příručka	29

Kapitola 1

Úvod

Během posledních let velmi vzrostla obliba používání mobilních zařízení, jako jsou mobilní telefony, přenosné počítače a další, které dokáží velmi zefektivnit každodenní práci. Jsou používanější čím dál tím víc zejména proto, že jejich rozměry jsou stále kompaktnější, a přitom jejich výkon se neustále zvyšuje.

V těchto dnech, kdy si téměř každý potřebuje organizovat čas, mít při sobě své kontakty a přistupovat na internet a email téměř odkudkoliv, se začínají mezi lidmi rozšiřovat kapesní počítače. Kapesní počítač sice nemůže plně nahradit stolní počítač, ale umožní dostatečnou náhradu jeho běžných funkcí v místě a době, kdy stolní počítač využít nelze. Hmotnost těchto zařízení většinou nepřesáhne 200 gramů a je tedy možné mít je při sobě 24 hodin denně. Navíc se na těchto zařízeních dají spouštět programy podobné těm, které známe ze stolních počítačů.

Při představování nových myšlenek, produktů, nebo při výuce se nezdírá využívat prezentací, které jsou spuštěny na stolním počítači a zobrazovány na promítacím plátně. Prezentace slouží především posluchačům, aby se ve výkladu neztratili, a také prezentujícímu jako vodítko, kterého se chce při svém výstupu držet. Ovládání prezentace vyžaduje stát u počítače, na kterém je prezentace spuštěna. Často by však prezentující chtěl přejít od počítače k plátnu, nebo blíž k posluchačům, avšak odtud by nemohl prezentaci ovládat.

Vytvořením aplikace, která by pomocí kapesního počítače umožnila ovládat prezentaci spuštěnou na jiném počítači, se zabývá tato práce.

Práce byla rozčleněna do osmi kapitol. První kapitola uvádí čtenáře do problematiky z širšího pohledu. V druhé kapitole jsou popsány kapesní počítače a problematika vývoje aplikací pro tato zařízení.

Třetí kapitola rozebírá analýzu problému, podle které jsou nastíněna možná řešení v souvislosti s požadavky na funkčnost. V další kapitole je podle analýzy proveden návrh řešení, navržen přibližný vzhled uživatelského rozhraní a chování aplikace. Pátá kapitola popisuje konkrétní implementaci podle návrhu.

V šesté kapitole jsou popsány podobné produkty, které rovněž umožňují ovládání prezentace na dálku. Je zde provedeno srovnání s navrhovaným řešením.

Sedmá kapitola naznačuje možný budoucí vývoj aplikace. Poslední kapitola uzavírá řešení problému a sumarizuje dosažené výsledky a zkušenosti s vývojem.

Kapitola 2

PDA

2.1 Charakteristika PDA

Podle [8, 5] se zkratkou PDA označuje malý kapesní počítač. Tento akronym je odvozen z názvu Personal Digital Assistant - osobní digitální pomocník. Původně měly PDA za cíl především pomoci s organizováním času a kontaktů. Současné PDA jsou velmi výkonné a zvládají i přehrávání videa a spoustu dalších aplikací, které jsou podobné těm ze stolních počítačů.

Historie PDA sahá do roku 1980, kdy firma Psion definovala tento název. Výkon a vybavení těchto zařízení se vyvíjí velmi rychle. Nyní již mají některé PDA procesor o frekvenci vyšší než 600 MHz a až 128 MB RAM operační paměti. Dále mohou obsahovat modul pro komunikaci přes Bluetooth, Wi-Fi pro bezdrátové připojení, GSM modul pro používání PDA současně jako mobilního telefonu, nebo modul GPS pro určení pozice uživatele na Zemi. Navíc každé PDA obsahuje slot pro paměťovou kartu, který nejčastěji slouží pro zvýšení kapacity úložného prostoru, avšak rovněž umožňuje přidat další přídatná zařízení, která PDA neobsahuje, jako např. fotoaparát.

- Hmotnost kapesního počítače se pohybuje mezi 100 a 250 gramy a rozměry obvykle nepřesahují $15 \times 10 \times 2$ cm. Používá baterii jako zdroj energie.
- Kapesní počítač je vybaven dotykovým displejem, který se ovládá tužkou, nazývanou stylus. Ovládání probíhá prostřednictvím ukázování na grafické objekty, zobrazené na displeji.
- PDA nemá plnohodnotnou klávesnici jako stolní počítač. Text se zadává pomocí systému rozpoznávání písma, nebo malé klávesnice, která se zobrazí na displeji. Tento způsob vyhovuje pouze při psaní krátkých textů, jako např. poznámek nebo kontaktů.
- PDA je schopné komunikovat s okolím buď pomocí bezdrátových technologií, nebo přes kabel.

Budoucnost kapesních počítačů je naznačována mnoha trendy, které jsou ale nezdědka protichůdné. Z jedné strany je zde snaha přibližovat se komunikátorům a implementovat funkce mobilních telefonů, na druhé straně vznikají modely s velkými displeji a vysokým rozlišením. Dokonce se objevily i přístroje s pevným diskem. Asi neexistuje jedna optimální cesta, protože každému vyhovuje něco trochu jiného. Díky tomu ale může vznikat široká nabídka různých přístrojů se sice podobnými, avšak v detailech rozdílnými vlastnostmi.

2.2 Platformy na PDA

Mezi nejčastější operační systémy používané na PDA patří Windows Mobile, PalmOS, Linux a Symbian. Pro tento projekt byla vybrána platforma Pocket PC od firmy Microsoft, a proto se jí budu nyní zabývat.

První úvahy o implementaci tohoto operačního systému (též zvaném Pocket PC) pro kapesní počítače přišly na svět v roce 1992. Do projektu se zapojily zejména společnosti Microsoft, Compaq, Motorola, NEC a Sharp. V roce 1995 byl vydán Windows CE 1.

V roce 1998 vyšel Windows CE 2.0, který přinesl větší rozlišení a také podporu Wi-Fi pro přístup do bezdrátových sítí.

V dubnu 2000 byl vydán Windows CE 3.0, který se od předchozích verzí dost odlišoval, a byl nápadně podobný konkurenčnímu Palmu. Nepoužíval totiž klávesnici, displej byl vyšší než širší s rozlišením 320*240 a primárně se vše ovládalo přes dotykový displej. Pouze nejčastěji používané funkce byly namapovány na hardwarová tlačítka a tento systém se v zásadě používá dodnes. Současně s těmito změnami došlo k další důležité věci, a to k výraznému snížení cen PDA s tímto operačním systémem, a Windows CE získal převahu nad Palmem.

Od verze Windows CE 4.0 je systém označován jako Pocket PC 2003, nebo spíše jako Windows Mobile 2003, k výrazným změnám, kterých by si uživatelé všimli, ale nedošlo. Windows Mobile 2003 SE přinesl podporu rozlišení VGA.

Aktuální verze je Windows Mobile 2005 (Windows CE 5.0), která především umožňuje synchronizaci s operačním systémem Windows Vista na stolních počítačích. Více viz [10].

2.3 Tvorba programů pro Pocket PC

Vývoj aplikací pro PDA se příliš neliší od vývoje aplikací pro stolní počítače. Při vývoji je však potřeba brát v úvahu slabší výkon procesoru, méně operační paměti, menší displej, ovládání stylusem a také výdrž baterie PDA.

V současné době je k dispozici několik vývojových nástrojů. Zde se budu věnovat těm od společnosti Microsoft, neboť jsou zdarma a pro vývojáře volně k dispozici. Microsoft těmito nástroji pokrývá v podstatě všechna Pocket PC a Smartphone zařízení založená na architektuře Windows Mobile. Tato kapitola poskytne přehled základních možností a využití nabízených nástrojů tak, aby bylo možné vybrat si ten, který nejvíce vyhovuje požadavkům tohoto projektu. Více viz [1].

2.3.1 Visual Studio .NET 2003

Tento nástroj využívá možností, které přináší Microsoft .NET Compact Framework a umožňující vývoj distribuovaných aplikací. Vývojáři poskytují knihovnu tříd pro pohodlný a rychlý vývoj aplikací, stejně jako i tvorbu tříd (komponent) nových a opětovně použitelných. Vývojář vyvíjí aplikace, které používají stejné nástroje jak na Pocket PC, tak i na stolním počítači. Tím se celý vývojový proces sjednocuje a poskytuje přenositelný a univerzální zdrojový kód nezávislý na procesoru a operačním systému. Programovacím jazykem se zde stává nově vzniklé C# a upravený Visual Basic .NET. Ve výčtu vlastností nesmí chybět také podpora Web procesů.

2.3.2 eMbedded Visual C++ 3.0

Microsoft eMbedded Visual Tools 3.0 poskytuje koncové vývojářské prostředí s jednoduchým a intuitivním grafickým uživatelských rozhraním (dále GUI), integrovaný kompilátor, ladící nástroj a samozřejmě kompletní dokumentaci k platformě. Tento nástroj je primárně určen pro vývoj aplikací pro operační systém Windows Mobile 2002 (též známý jako Pocket PC 2002). Je možné v něm vyvíjet aplikace určené pro OS Windows Mobile 2003, i když výsledný kód není optimalizovaný přímo pro něj. Ladění aplikací je možné dvěma způsoby. Buď přes MS ActiveSync nebo pomocí integrovaného emulátoru. Emulátor je dodáván automaticky jako součást SDK (Software Development Kit) pro WM2002, plně emuluje možnosti Pocket PC zařízení včetně vzhledu a umožňuje spouštět zkompileované binární soubory pro WM2002 přímo ve virtuálním prostředí. Mám-li provést srovnání, pak je nutné ztotožnit tento nástroj s nástrojem známým ze stolních počítačů – MS Visual Studio. Součástí je eMbedded Visual C++ 3.0 (odlehčené Visual C++) a také eMbedded Visual Basic 3.0 (odlehčený Visual Basic). Stejně jako na stolních počítačích, i zde je možné používat ActiveX komponenty, COM servery nebo grafické rozhraní poskytované pomocí Game API (GAPI).

2.3.3 eMbedded Visual C++ 4.0

Microsoft eMbedded Visual C++ 4.0 poskytuje koncové vývojářské prostředí pro vývoj aplikací pro Windows CE .NET 4.2 zařízení a přes dostupný opravný balíček (SP2) také pro WM2003. Z principu vychází z eVC++ 3.0 a přidává k němu nové možnosti, které dříve nebyly vývojářům dostupné. Tyto možnosti zahrnují JIT (Just-In-Time) ladící nástroj pro diagnostiku nezachycených vyjímek, strukturovaná správa vyjímek známá z VC++, „Attach-To” proces (neboli připojení k jinému procesu) pro lepší ladící možnosti (známý problém ladění např. Today doplňků, které nebylo v eVC++ 3.0 prakticky možné) a samozřejmě také nový emulátor založený na WM2003. Kód produkovaný tímto nástrojem je možné spustit pouze na zařízeních založených na WM2003 nebo Windows CE .NET 4.2.

2.3.4 Jazyk C#

C# (vyslovované anglicky jako C sharp) je vysokoúrovňový objektově orientovaný programovací jazyk vyvinutý firmou Microsoft zároveň s platformou .NET. Microsoft založil C# na jazycích C++ a Java (a je tedy nepřímým potomkem jazyka C, ze kterého čerpá syntaxi). C# se využívá hlavně k tvorbě databázových programů, webových aplikací, webových služeb, formulářových aplikací ve Windows apod. Výhodou je přenositelnost mezi platformami, která vyplývá z toho, že zdrojový kód je kompilován až za běhu aplikace. Nevýhodou je pomalejší běh programu a pro spuštění aplikace je nutné mít nainstalován .NET framework.

2.3.5 Jazyk C/C++

C je programovací jazyk, který vyvinuli Ken Thompson a Dennis Ritchie pro potřeby operačního systému Unix v roce 1973. V současné době je to jeden z nejpobulárnějších jazyků, zřejmě nejčastější pro psaní systémového softwaru, ale velmi rozšířený i pro aplikace. Pojmenování „C” zvolili, protože mnoho vlastností přebírali ze staršího jazyka zvaného „B”, jehož název byl zase odvozen od jazyka BCPL. Jedná se o jazyk kompilovaný, a tudíž nepotřebuje žádné další nainstalované prostředí pro spuštění aplikace.

C++ je objektivě orientovaný programovací jazyk, který vyvinul Bjarne Stroustrup a další v Bellových laboratořích AT&T jako rozšíření jazyka C. C++ podporuje několik programovacích stylů (paradigmat), jako je procedurální programování, objektivě orientované programování a generické programování. Z toho vyplývá, že to není jazyk čistě objektivý.

Jazyk C je až na několik jasně definovaných výjimek podmnožinou C++. Jak uvádí Bjarne Stroustrup, všechny programy uvedené ve slavné učebnici jazyka C *The C Programming Language* od Briana W. Kernighana a Dennise M. Ritchieho jsou zároveň programy v C++.

První překladače C++ byly preprocesory, které překládaly z C++ do čistého C. Považovat jazyk C++ za pouhé rozšíření jazyka C by ale bylo chybou, protože není s jazykem C zcela kompatibilní. Některé programy v jazyce C nelze překládat překladači pro C++.

Kapitola 3

Analýza problému

3.1 Motivace

Během posledních let se většina přednášek a konferencí přenesla od prezentací na průsvítkách k prezentacím, které jsou vytvořeny a ovládány pomocí počítače. Zobrazovací technika umožňuje zobrazit prezentaci na velkém plátně, které je dobře viditelné všem posluchačům v mnohdy velkých přednáškových sálech.

Prezentace slouží přednášejícímu jako vodítko, kterého se má při prezentování držet, ale hlavně posluchačům, kteří z ní vyčtou nejdůležitější pojmy. Bývají do ní zahrnuty také složitější části, které chce prezentující podrobněji vysvětlit. K tomu často potřebuje přejít od počítače, ze kterého prezentaci ovládá, k plátnu a z tohoto místa na plátně konkrétní věci ukazovat a vysvětlovat. Pokud ale chce pokračovat v ovládání prezentace, musí se opět vrátit od plátna ke klávesnici. Podobný problém může nastat tehdy, pokud chce být prezentující v bližším kontaktu s posluchači, například mezi nimi procházet a bavit se s nimi z očí do očí. Pak opět nemá možnost prezentaci na počítači ovládat.

Je to podobné jako s televizí. Dokáže si dnes ještě někdo představit přepínání programu nebo zesílení zvuku bez dálkového ovládání? Elektronika má lidem usnadňovat život a tato práce se usnadnění a zefektivnění snaží docílit.

3.2 Funkční požadavky

Uživatel by měl být schopen pomocí kapesního počítače ovládat prezentaci spuštěnou na stolním počítači. Tzn. uživatel na PDA stiskne tlačítko a podle tlačítka se příslušná akce na tomto počítači vykoná. Ovládáním prezentace se myslí tyto funkce:

- přechod na další / předchozí snímek prezentace
- přechod na začátek / konec prezentace
- zesílení / zeslabení zvuku prezentace
- ovládání kurzoru myši

PDA a PC spolu musí nějakým způsobem komunikovat. Jedním ze způsobů komunikace, který se hodí pro tento případ, je technologie Bluetooth (bude probrána v kapitole 3.7).

Projekt se tedy skládá ze dvou částí: serverové a klientské. Klientská část je program pro PDA. To slouží jako dálkový ovladač, který odesílá požadavky. Serverová část je program

pro PC, který realizuje příkazy od klienta. Dále se již budu věnovat klientské i serverové části samostatně.

3.3 Architektura klient - server

Jedná se o síťovou architekturu, která odděluje klienta od serveru. Klient může být aplikace s grafickým uživatelským rozhraním, která na vyžádání zobrazuje data přijatá od serveru. Typickým příkladem je webový prohlížeč. Jakmile webový prohlížeč (klient) odešle serveru požadavek na webovou stránku, server požadavek zpracuje, vyhledá všechny součásti potřebné pro zobrazení stránky a tu odešle zpět klientovi, který následně stránku zobrazí.

Jiný případ architektury klient - server je tzv. vzdálená prezentace, kdy server přijímá data od klienta a zobrazuje je (zpracovává). To je případ naší aplikace. Více viz [6].

3.4 Klientská část - PDA

Po spuštění programu se zobrazí obrazovka pro ustavení komunikace mezi klientem a serverem. V případě komunikace přes Bluetooth musí uživatel zvolit pouze příslušný port a dále již vidí obrazovku s ovládacími tlačítky, pomocí kterých může ovládat spuštěnou prezentaci.

Klientská část by měla obsahovat hezké grafické rozhraní, spočívající v ergonomickém rozložení tlačítek. Aplikace by měla prezentujícímu ulehčit ovládání prezentace tak, aby mohl přecházet mezi snímky bez nutnosti dívat se na PDA a nedocházelo přitom k častým stisknutím jiných tlačítek.

3.5 Serverová část - PC

Server nejprve čeká na připojení klienta a po navázání spojení přijímá a vykonává jeho požadavky.

Ovládání prezentace by mělo být nezávislé na programu, ve kterém je prezentace spuštěna. Takové programy vždy rozumí klávesovým povelům *page up*, *page down* apod. Pro takové ovládání je nutná spolupráce aplikace s operačním systémem – v našem případě Microsoft Windows, a přesně tyto služby nabízí Windows API (viz následující kapitola).

3.6 WinAPI

WinAPI [3, 9] je neformální zkratka pro Windows Application Programming Interface – aplikační programové rozhraní operačního systému Microsoft Windows.

3.6.1 Rozhraní API a paměťové modely

Z pohledu programátora je operační systém definován svým API. API obsahuje všechna volání funkcí, které od operačního systému může aplikační program žádat. Navíc obsahuje definice souvisejících datových typů a struktur.

Obecně platí, že Windows API zůstává zcela konzistentní od Windows 1.0. Funkce, které byly dostupné ve Windows 1.0, jsou stejné i v dnešních verzích Windows, nebo byly rozšířeny z důvodu přechodu z 16bitové architektury na 32bitovou. První Windows podporovaly méně než 450 funkcí, zatímco dnes jich jsou tisíce.

3.6.2 Komponenty WinAPI

Podle funkcí, které WinAPI nabízí, můžeme komponenty rozdělit do sedmi kategorií. Překlad některých názvů do češtiny by byl umělý a nepoužívá se, a proto ponechám názvy v angličtině. Soubory, které poskytují jednotlivé funkce, mají různé názvy v závislosti na tom, jestli je operační systém 16bitový nebo 32bitový. Já zde vypíši pouze jejich 32bitové verze.

Base Services – poskytují přístup k základním zdrojům operačního systému. Jedná se např. o souborový systém, zařízení, procesy, vlákna, přístup k registrům a zpracování chyb. Tyto funkce se nacházejí v souborech `kernel32.dll` a `advapi32.dll`.

Graphics Device Interface - poskytují funkce pro grafický výstup na monitor, tiskárnu apod. Najít je můžeme v `gdi32.dll` ve 32bitových Windows.

User Interface - pomocí této komponenty můžeme vytvářet okna na obrazovce a většinu kontrolních prvků, jako jsou tlačítka a posuvníky, zpracovávat zprávy od myši a klávesnice a další. Tato komponenta také obsahuje funkce, spojené s uživatelským rozhraním systému Windows. Obsahuje je soubor `user32.dll`. Od verze Windows XP se tyto funkce, společně s Common Control Library, nacházejí v `comctl32.dll`.

Common Dialog Box Library - tato komponenta zpřístupňuje dialogová okna pro otevírání a ukládání souborů, vybírání druhů písma a podobná. Nachází se v `comdlg32.dll`.

Common Control Library - umožňuje aplikaci přistupovat k pokročilejším prvkům systému, jako je stavový řádek a další součásti okna. Tyto funkce se nacházejí v `comctl32.dll`.

Windows Shell - komponenta, umožňující přístup k funkcím, které poskytuje shell operačního systému. Tyto funkce se nacházejí v `shlwapi.dll`

Network Services - poskytuje funkce pro využívání síťových služeb, jako jsou NetBIOS, Winsock a další.

3.6.3 Možnosti jazyka

Jazyk C a jemu vlastní API není jediným způsobem psaní programů ve Windows. Tento postup však přináší nejlepší výkon a nejširší možnosti funkcí Windows. Spustitelné soubory jsou poměrně malé a pro spuštění nepotřebují další knihovny (samozřejmě kromě samotných knihoven DLL ve Windows).

3.6.4 Programovací prostředí

Pro vývoj aplikace, která využívá WinAPI, existuje několik vývojových prostředí. Jednou z nejlepších voleb je Microsoft Visual C++, které kromě překladače jazyka C a dalších souborů a nástrojů potřebných pro překlad a sestavení programů pro Windows obsahuje také prostředí pro úpravu zdrojových kódů a interaktivní vytváření zdrojů, jako jsou ikony nebo dialogová okna. Dále umožňuje spouštět a odlaďovat přeložené programy.

Mezi další bychom mohli zařadit překladač od firmy Borland. Tento, spolu s překladači od firmy Microsoft, byly dlouhou dobu jedinnými nástroji, které byly schopné program pro Windows přeložit. V dnešní době lze použít i projekty MinGW nebo Cygwin, které poskytují prostředí založené na GNU Compiler Collection [7].

3.7 Bluetooth

3.7.1 Specifikace Bluetooth

Bluetooth, v překladu „modrý zub“, vznikl v devadesátých letech minulého století jako technologie, která chtěla spojit všechna mobilní zařízení a poskytnout jim tak možnost vzájemné spolupráce. Je to specifikace bezdrátové komunikace určené zejména pro malou vzdálenost (většinou do 10m), která nahrazuje hardwarově a softwarově nekompatibilní kabelová řešení. Z hlediska mobilních zařízení má Bluetooth dvě obrovské výhody:

- malý příkon (vyplývající z omezeného dosahu)
- automatickou konfiguraci (uživatelsky jednoduchá technologie)

Z výše uvedených vlastností vyplývá, že tato komunikace je vhodná právě pro spojení mobilních počítačů, mobilních telefonů, PDA, tiskáren, fotoaparátů a dalších přenosných zařízení. Stejně tak jako většina jiných specifikací, i Bluetooth specifikace zahrnuje jak pravidla pro implementaci rozhraní, tak i pro návrh periferií.

Technologie je založena na radiovém spojení v pásmu 2,4GHz na frekvencích 2400 MHz - 2483,5 MHz. Toto pásmo je definováno ve většině států jako bezlicenční. Spojení může přenášet spolehlivě a poměrně rychle jak data, tak i hlasový signál. Podle maximálního výstupního výkonu definuje specifikace 3 třídy zařízení:

- 10 mW (20 dBm) - dosah cca 100m
- 2,5 mW (4 dBm) - dosah cca 50m
- 1 mW (0 dBm) standardní specifikace - dosah cca 10m

3.7.2 Programové řešení Bluetooth

Možné aplikace jsou definovány tzv. profily, které tvoří základní část specifikace. Zajišťují vzájemnou slučitelnost zařízení na nejvyšší softwarové úrovni. Profily určují způsob komunikace a možné využití Bluetooth zařízení.

Aby zařízení mohla smysluplně komunikovat, musí podporovat obě komunikující strany stejný profil (Sériový, Dial-up, Headset...). Je logické, že například bezdrátový headset bude podporovat Headset profil, bude moci komunikovat s GSM přístrojem podporujícím stejný profil a nebude komunikovat s digitálním fotoaparátem).

V tomto projektu budu využívat Serial Port Profile, který definuje nastavení virtuálních sériových portů pomocí Bluetooth. Jedná se tak o klasickou náhradu sériového propojení s rychlostí do 128kbps. Více o Bluetooth se lze dočíst zde [4, 2].

Kapitola 4

Návrh

4.1 Komunikace mezi stolním počítačem a PDA

Jak bylo uvedeno v předchozích kapitolách, zařízení spolu budou komunikovat bezdrátově pomocí technologie Bluetooth. Jedná se o sériovou komunikaci a v našem případě bude stačit komunikace jednosměrná, tj. směrem od klienta k serveru. Klient nepotřebuje přijímat od serveru žádné počáteční nastavení, ani potvrzování o správném doručení zprávy. Spojení se ustaví tak, že uživatel zvolí příslušné Bluetooth porty na PDA a na PC. Klient i server mají předem domluvenou sadu zpráv, kdy každý z nich ví, co která zpráva znamená. Následná komunikace spočívá v odeslání domluvené zprávy od klienta serveru, který zprávu přijme a provede příslušnou činnost.

Zprávy můžeme rozdělit do dvou kategorií:

- zprávy, které jsou generované softwarovými tlačítky klientského programu, a to:
 - zprávy, sloužící pro navigování v prezentaci (vpřed, vzad, začátek a konec)
 - zprávy pro zesílení nebo zeslabení zvuku
- zprávy generované hardwarovými tlačítky PDA (joystickem), které umožňují ovládat kurzor myši

4.2 Klientská část - PDA

4.2.1 Zahájení komunikace

Po spuštění programu musí uživatel pouze zahájit komunikaci přes Bluetooth. To spočívá ve vybrání příslušného portu. Pokud uživatel spustil program poprvé, má dvě možnosti. Buď vybere port ze seznamu ručně, nebo požádá program, aby se pokusil příslušný port najít automaticky. Automatické hledání portu není a nemůže být s jistotou přesné, protože sériových portů je v PDA celá řada a pokud bychom ze záznamů v registrech vybrali jen ty, které přísluší Bluetooth, nachází se jich zde více. Když jsem zkoušel vyřešit tento problém obousměrnou komunikací klienta a serveru, nastaly dva problémy. Prvním byl ten, že musela být spuštěna serverová část, ve které rovněž uživatel musí vybrat správný port (který taktéž nemusí znát). Druhý problém nastal, když jsem se pokoušel odesílat zprávy na všechny porty postupně a čekal, ze kterého portu přijde odpověď od serveru. Při pokusu

o otevření určitých portů a následným odesláním zprávy ale celé PDA zamrzlo a pomohl pouze restart.

Pokud už komunikace jednou úspěšně probíhá, má uživatel možnost uložit si aktuální profil s nastavením programu a s ním se uloží i konkrétní číslo portu, který se používá. Při dalších spuštěních programu už může uživatel obnovit dříve uložený profil a problém s výběrem portu odpadá.

4.2.2 Odeslání zprávy

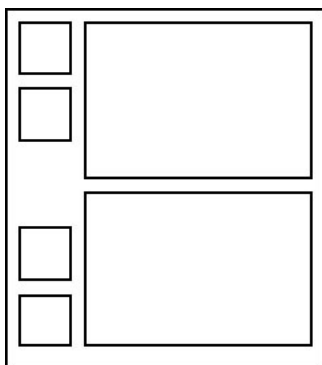
Po správném otevření příslušného portu je již samotné odeslání zprávy jednoduché. Po stisknutí tlačítka stačí zapsat příslušný kód akce na tento port.

4.2.3 Uživatelské rozhraní

Uživatelské rozhraní pro klientskou část je jednou z nejdůležitějších částí této práce. PDA se přes dotykovou obrazovku většinou ovládá stylusem, ale v tomto případě bude program spíše ovládán přímo palcem. Tento způsob ovládání není příliš přesný, a proto je potřeba, aby měla tlačítka velké rozměry a nedocházelo k nechtěným stisknutím jiných tlačítek. Obrazovka PDA je necelých 10 cm vysoká a přibližně 6 cm široká. Pokud má palec najít tlačítko intuitivně, nemůže být tlačítek na obrazovce velké množství.

Nejpoužívanější funkcí je zřejmě přechod na další snímek, a tedy tlačítko pro tuto funkci by mělo být umístěno v co nejpřirozenější poloze pro palec vzhledem k držení PDA. Po několika testech jsem zjistil, že když člověk drží PDA a pak palcem sáhne „někam“ na dotykovou obrazovku, ve většině případů palec sahá přibližně do poloviny šířky obrazovky a liší se pouze výška dotyku vzhledem ke spodnímu okraji displeje (v závislosti na způsobu uchopení PDA). Potěšující je fakt, že nezáleželo na tom, zda bylo PDA drženo v levé ruce nebo pravé.

Podle tohoto průzkumu tedy vyšlo jasné rozložení tlačítek. Tlačítka *vpřed* a *vzad* tedy budou největší a budou mít šířku větší, než je polovina šířky displeje. Ostatní tlačítka budou podstatně menší, musí však být stále možné stisknout je palcem. Návrh je vidět na obrázku:



Obrázek 4.1: Návrh rozložení tlačítek

Na obrázku je vidět šest pozic pro tlačítka. Čtyři menší na levé straně slouží k zesílení a zeslabení zvuku a k přechodu na začátek a konec prezentace. Tyto funkce se nepoužívají tak často, takže není tak důležité, jestli je ovládání zvuku v horní části nebo ve spodní, uživatel se v případě potřeby podívá a snadno je najde. Umístění ale není tak jednoduché,

pokud se jedná o velká tlačítka pro přechod na následující nebo předchozí snímek. Ty totiž uživatel používá v podstatě neustále a ovládání pro něho musí být intuitivní. Někdo je zvyklý číst dokumenty z počítače, a tedy na další stránku přechází směrem dolů (šipkou dolů). Někomu naopak může přijít přirozené na následující stránku přejít směrem dopředu (šipkou nahoru). Každý má proto možnost nastavit si v programu takové chování šipek, které mu vyhovuje.

PDA má (jak již bylo řečeno) kromě dotykového displeje také hardwarová tlačítka. Z nich se dá v této aplikaci s výhodou využít joysticku, primárně pro pohyb kurzoru myši. Joystick má oproti softwarovým tlačítkům jednu velkou výhodu. Pokud totiž stiskneme tlačítko na displeji a držíme ho delší dobu, akce se nezávisle na délce držení provede jen jednou. Joystick se chová jinak, při jeho držení se akce stále opakuje, a tak při ovládání myši se může kurzor pohybovat plynule. Tento způsob se dá ale využít i při procházení v prezentaci. Pokud obsahuje prezentace mnoho stránek, vyhledání určité stránky přecházením po jedné by bylo zdouhavé a každý přechod by vyžadoval jedno stisknutí. Uživatel má tedy možnost změnit si v programu chování joysticku tak, aby s ním neovládal myš, ale směrové klávesy pro přechod mezi stránkami.

Prezentace mohou probíhat i v setměné místnosti, a protože displej kapesního počítače může dost zářit, je vhodné mít možnost přepnout zobrazení do tzv. nočního režimu, ve kterém displej nebude zbytečně osvětlovat jinak tmavou místnost.

4.3 Serverová část - PC

4.3.1 Zahájení komunikace

Po spuštění programu je opět potřeba pouze vybrat port, který používá Bluetooth jako sériový port. Tento port lze většinou snadno najít v programu, který se nainstaluje společně s Bluetooth zařízením.

4.3.2 Uživatelské rozhraní

Uživatelské rozhraní pro tuto část není tak důležité, jako v případě části pro PDA. K plné funkčnosti programu stačí pouze seznam pro výběr portů, ze kterých uživatel vybírá při zahajování komunikace, dále tlačítka *Start* a *Stop* pro zahájení a ukončení komunikace a posuvník, kterým se nastavuje rychlost pohybu kurzoru myši při ovládání joystickem.

Program poté pracuje na pozadí a aby nezabíral místo ve spodní liště, je možné ho minimalizovat do systémové oblasti, kde je zobrazena pouze malá ikona.

4.3.3 Provádění příkazů

Všechny námi potřebné příkazy umožňuje realizovat WinAPI.

Jedním z typů příkazů jsou příkazy umožňující pohyb v prezentaci. Tyto příkazy vlastně simulují stisknutí klávesy na klávesnici. Jedná se o klávesy *šipka nahoru*, *šipka dolů*, *šipka vlevo*, *šipka vpravo*, *home* a *end* s příslušnými virtuálními kódy.

Dalším typem jsou příkazy pro pohyb kurzoru myši. Přestože to není příliš pohodlné, lze touto aplikací částečně nahradit myš. Joystick na PDA se dá i stisknout a stisknutí má tedy logicky přiřazenu myši funkci *stisknutí levého tlačítka*.

Posledním typem jsou příkazy, které umožňují zesílit a zeslabit zvuk na PC. WinAPI poskytuje struktury, které velmi usnadňují přístup ke zvukovým zařízením počítače, a proto jich využijeme.

Kapitola 5

Implementace

Jak již bylo zmíněno dříve, obě části (klientská i serverová) jsou implementovány v jazyce C/C++ a pro volání funkcí operačního systému je použito WinAPI. Nyní se opět budu věnovat každé části samostatně.

5.1 Klient

5.1.1 Použité vývojové prostředí

Po prostudování vývojových prostředí jsem vybral vývojové prostředí Microsoft eMbedded Visual C++ 4.0 s aktuálním opravným balíčkem SP4, které velmi usnadňuje vytváření aplikací pro kapesní počítače. Editor tohoto nástroje zvýrazňuje syntaxi používaného jazyka, umožňuje pohled na stromovou strukturu souborů a všech tříd projektu a také velmi usnadňuje vytváření a úpravu zdrojů.

Další výhodou tohoto nástroje je to, že po zkompilování a sestavení programu umožňuje spustit program buď na klasickém fyzicky připojeném PDA (pomocí synchronizačního nástroje Microsoft ActiveSync), nebo přímo ve vestavěném emulátoru. Pro vývoj např. uživatelského rozhraní tedy není nutné PDA vlastnit. Pomocí emulátoru jsem ale nevyužíval příliš dlouho, neboť není možné simulovat Bluetooth adaptér a bohužel se mi ani nepodařilo nastavit virtuální port, který by Bluetooth simuloval.

5.1.2 První implementace

Nejprve jsem si myslel, že celou aplikaci založím na dialogu. Značně se tím ulehčilo vytváření tlačítek pomocí editoru zdrojů, který dokáže vytvořit tlačítko na dialogu pouze jedním kliknutím myši. Zanedlouho jsem však narazil na dva problémy, z nichž jeden byl neřešitelný.

První problém byl ten, že při použití modálního dialogu (takového, který je nad všemi ostatními okny) nebylo možné používat v programu nabídku menu. Tento problém by se dal vyřešit použitím nemedálního dialogu. Pouze by se muselo kontrolovat, zda je okno aplikace nad všemi ostatními okny před stisknutím hardwarového tlačítka.

Druhý problém, který se týkal právě hardwarových tlačítek – joysticku – však již řešitelný nebyl. Pokud je totiž spuštěn dialog, joystick (směrové šipky) slouží k pohybu po prvcích na dialogu, např. po tlačítkách. Zprávy od joysticku tedy není možné odchytnout a přiřadit jim jiné chování. Protože joystick hraje významnou roli v mé aplikaci, byl jsem nucen od této implementace upustit.

5.1.3 Výsledná implementace

Abych tedy mohl využívat joystick, nezbyvá jiná alternativa, než umístit ovládací tlačítka do hlavního okna programu a dialogy používat pouze pro různá nastavení.

Po spuštění se nejprve vytvoří hlavní okno programu se všemi ovládacími tlačítky a poté se ihned otevře dialog pro zvolení způsobu komunikace, které hlavní okno překrývá (obrázek 5.1). Při vytváření grafiky jsem zohlednil i možná budoucí rozšíření programu a jsou zde dvě tlačítka. Jedno pro výběr komunikace přes Bluetooth a druhé pro komunikaci přes Wi-Fi. Druhé tlačítko je ale zatím bez funkce.



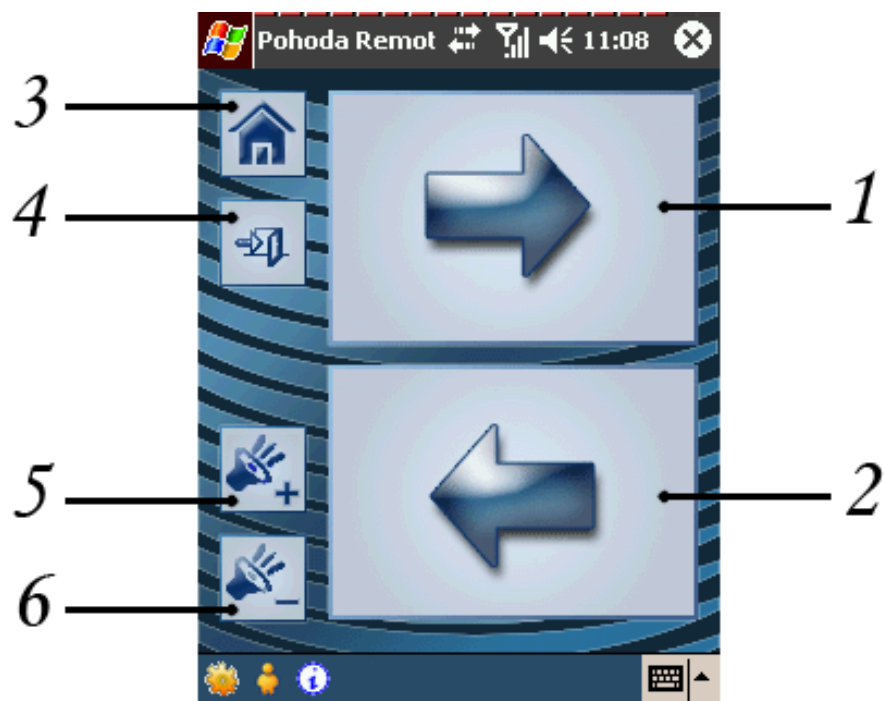
Obrázek 5.1: Zvolení způsobu komunikace

Po kliknutí na tlačítko Bluetooth se zobrazí další dialog s nastavením komunikace (obrázek 5.2). Ještě před zobrazením dialogu proběhne kontrola, zda je zařízení Bluetooth v zařízení přítomné. Pokud zařízení Bluetooth neobsahuje, uživatel obdrží varovné hlášení. Pokud je Bluetooth vypnuté, program se ho pokusí zapnout. V této fázi opět nastaly komplikace, protože v kapesních počítačích jsou ovladače pro Bluetooth od dvou různých výrobců. Jedná se o Bluetooth stack od společností Microsoft a od společnosti Widcomm. Microsoft poskytuje, narozdíl od Widcommu, své API pro přístup k Bluetooth zařízení zdarma a není tedy problém pomocí dostupných funkcí Bluetooth zapnout nebo vypnout. V případě PDA s ovladači od společnosti Widcomm jsem situaci vyřešil hledáním a zapisováním přímo do registrů systému, a API tedy nebylo potřeba kupovat (ačkoliv by to velmi usnadnilo implementaci).

Po zahájení komunikace se dialog ukončí a uživatel vidí hlavní okno programu (obrázek 5.3). Tlačítka pro ovládání nejsou vytvořena jako běžná tlačítka, ale jako klikací oblasti, kdy se při každém dotyku displeje testuje místo, na které bylo kliknuto, a pokud to bylo v oblasti některého tlačítka, provede se příslušná akce. Zároveň je po dobu stisknutí tlačítka změněna jeho grafická podoba, a tak má uživatel kontrolu, které tlačítko zasáhl. Pokud je tlačítko uvolněno, odešle se kód akce na otevřený sériový port. Tlačítka 1 a 2 umožňují



Obrázek 5.2: Nastavení komunikace přes Bluetooth

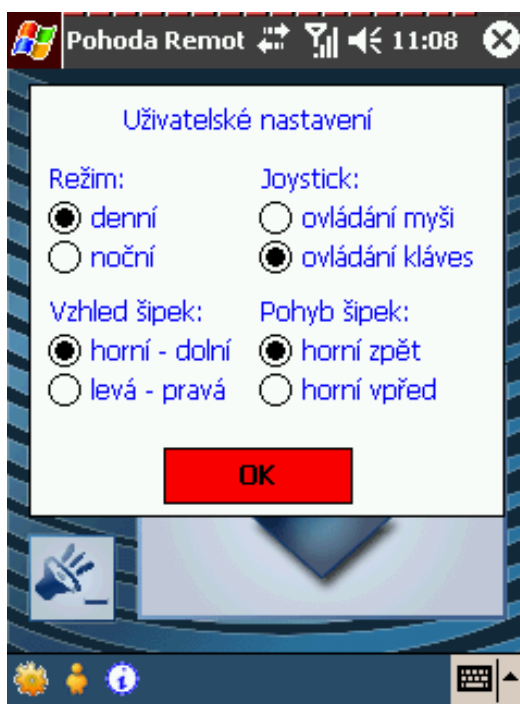


Obrázek 5.3: Hlavní obrazovka s ovládacími tlačítky a menu

pohyb vpřed a vzad v prezentaci. Je možné změnit rozložení tlačítek (vlevo, vpravo) a také funkčnost (jakým směrem se v prezentaci posune). Každý má tedy možnost nastavit si vzhled a chování tlačítek přesně podle sebe. Důvody k tomuto byly probrány v kapitole 4.2.3. Tlačítka 3 a 4 slouží k presunu na začátek nebo konec prezentace (klávesy Home a End na klávesnici). Zbylá dvě tlačítka 5 a 6 zesilují nebo zeslabují zvuk. Z této obrazovky je přístupné menu na spodním okraji displeje. Položka vlevo umožňuje

- uložit aktuální profil
- znovu zahájit komunikaci (přejít na úvodní obrazovku)
- ukončit program

Položka uprostřed otevře dialog s uživatelským nastavením (obrázek 5.4) a poslední položka zobrazí informace o programu.



Obrázek 5.4: Dialog pro nastavení uživatelského profilu

V případě, že uživatelský profil byl již uložen, údaje jsou zapsány v souboru `settings.txt` na stejné adresářové úrovni, jako spustitelný soubor aplikace. V tomto souboru každý řádek obsahuje klíč a hodnotu a nezáleží na pořadí řádků. Tato implementace umožňuje uživateli jednoduše editovat jeho profil, případně používat více profilů. Uživatelský profil je v aplikaci definovaný jako struktura. V případě existence souboru s profilem se tato struktura při spuštění aplikace naplní údaji ze souboru. Pokud ještě profil uložen nebyl, nebo je z nějakého důvodu poškozený, struktura se naplní předem nastavenými výchozími hodnotami. Pokud uživatel z menu zvolí položku *uložit profil*, údaje ze struktury se zapíše do souboru.

Implementace struktury:

```
typedef struct profile {
```



```

int nm;           // night mode (1 - nocni rezim, 0 - normalni)
int bt_port;     // cislo bluetooth portu
int sipky;       // 0 - up down, 1 - left right
int pohyb_sipek; // 0 - horni sipka pohyb zpet, 1 - horni vpred
int joystick;    // 0 - ovladani mysi, 1 - ovladani sipek
} PROFILE;

```

Takhle může vypadat soubor `settings.txt`:

```

nightmode 0
bt_port 7
sipky 1
pohyb_sipek 0
joystick 0

```

5.1.4 Některé důležité funkce

```
void zpracujBt()
```

tato funkce je volána při otevírání dialogu pro nastavení komunikace přes Bluetooth. Má za úkol zjistit, zda zařízení obsahuje Bluetooth modul, typ BT stacku a případně Bluetooth zapnout. Nejprve ze systémového registru zjistí, jaký Bluetooth stack dané zařízení má a následně ho aktivuje. V případě MS stacku pomocí systémové funkce `BthSetMode()`, v případě Widcomm stacku virtuálním kliknutím přímo na ikonu Bluetooth v systému Windows Mobile, protože knihovna od firmy Widcomm je placená.

```
HANDLE OpenComPort(char cislo, BOOL test)
```

funkce, která se pokusí otevřít port, jehož číslo je předáno jako první parametr, a v případě úspěchu vrátí handle otevřeného portu. Port se otevírá stejnou funkcí jako klasický soubor, a to funkcí `OpenFile()`, jen s tím rozdílem, že první parametr musí mít přesný formát sériového portu (např. „COM5:“). Druhý parametr slouží k povolení nebo zakázání výpisu chyb. Pokud probíhá automatické vyhledávání portu, není potřeba u většiny portů vypisovat, že se port nepodařilo otevřít.

```
int WriteToCom(HANDLE hComm, int kod)
```

funkce, která zapíše na otevřený port (první parametr funkce) kód akce, která se má vykonat (druhý parametr). Zápis je realizován pomocí stejné funkce, která se používá pro zápis do souboru, a to `WriteFile()`. Návrátový kód informuje o stavu provedené operace.

```
int TestRegion(HRGN *regiony[], int dlg, LONG x, LONG y, HWND hwnd)
```

funkce, která obdrží jako parametry ukazatel na pole regionů, identifikátor okna (dialogu), na které bylo kliknuto, souřadnice kliknutí a handle hlavního okna programu. Pozici kliknutí testuje pomocí systémové funkce `PtInRegion()`. Funkce vrátí identifikátor tlačítka, na které bylo kliknuto, případně 0, pokud bylo kliknuto jinam.

5.2 Server

5.2.1 Použité vývojové prostředí

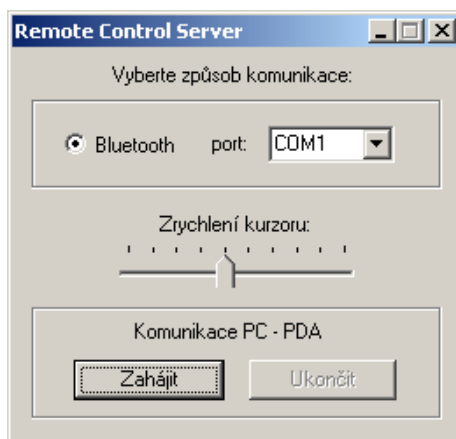
Pro vývoj serverové části programu jsem použil Microsoft Visual Studio 2005. Tento nástroj je svým vzhledem i poskytovanými funkcemi velice podobný eMbedded Visual Studiu, které bylo popsáno v kapitole 5.1.1. Rovněž v sobě obsahuje kompilátor, umožňuje spustit a ladit program a v neposlední řadě zvýraznit syntaxi. Nejvíc mi ale usnadnil práci na tomto projektu vestavěný editor zdrojů, pomocí kterého je vytvořeno celé uživatelské rozhraní programu.

5.2.2 Řešení

Celá aplikace je založena na dialogu (viz obrázek 5.5). V horní části je umístěno pole pro výběr příslušného Bluetooth portu, dále posuvník pro nastavení rychlosti pohybu kurzoru myši a dvě ovládací tlačítka pro zahájení a ukončení komunikace.

Po spuštění programu se zobrazí ikona v systémové oblasti Windows, odkud je možné program obnovit, pokud byl minimalizován, a nezabíral tak místo ve spodní liště.

Program je implementován jako vícevláknová aplikace. Primární vlákno vytvoří hlavní okno aplikace a všechny procedury tohoto okna a zpracovává všechny zprávy. Také jako jediný komunikuje s uživatelem. Druhé vlákno je vytvořeno po zahájení komunikace (stisknutím tlačítka *Start*). Toto vlákno představuje výkonný mechanismus na pozadí, odebírá data, která přijdou na sériový port, a vykonává podle nich příslušné operace. Vícevláknová implementace byla nevyhnutelná, protože funkce `ReadFile`, která čte data z portu, je v souvislosti s předem neznámým intervalem příchodu dat na port nastavena jako blokující.



Obrázek 5.5: Serverová část aplikace

5.2.3 Některé důležité funkce

```
HANDLE OpenPort(int cisloPortu)
```

funkce, která se pokusí otevřít port, jehož číslo je předáno jako první parametr, a v případě úspěchu vrátí handle otevřeného portu. Port se otevírá stejnou funkcí jako klasický soubor, a to funkcí `OpenFile()`, jen s tím rozdílem, že první parametr musí mít přesný formát sériového portu (např. „COM5:“).

```
void GenerateKey(int keycode)
```

funkce, která jako první parametr obdrží virtuální kód klávesy a provede simulaci jejího stisknutí, jako kdyby klávesu někdo doopravdy stiskl na klávesnici. Pro aplikaci Microsoft PowerPoint nelze použít systémovou funkci `PostMessage()`, proto používám funkci `SendInput()`, která posílá vstup do okna, které je aktivní („má focus“).

```
void GenerateMove(int direction, int count)
```

funkce, která posouvá kurzor myši v závislosti na směru (první parametr) a rychlosti (druhý parametr). Pohyb je realizován jako zrychlený, čím déle je joystick držen, tím rychleji se kurzor pohybuje. Pro nastavení polohy kurzoru je použita funkce `SetCursorPos()`.

```
void LeftClick()
```

funkce, která simuluje stisknutí levého tlačítka myši a opět používá funkci `SendInput()`.

```
void SetVolume(WPARAM wParam)
```

funkce, která v závislosti na parametru zesílí nebo zeslabí zvuk. Jako základní objekt je použit `HMIXER`, který slouží k získání a uchování parametrů a který následně využívají funkce

`mixerGetControlDetails()` a `mixerSetControlDetails()` pro nastavení hlasitosti.

Kapitola 6

Podobné produkty

6.1 Hardwarová řešení

Inspirací pro aplikaci, která by umožňovala ovládat prezentaci na počítači pomocí kapesního počítače, byly dálkové ovladače, které přesně toto umožňují. Jedním ze zástupců je Genius Wireless Pointer (6.1). Jedná se o univerzální rádiové dálkové ovládání pro prezentace a multimedia, vybavené vestavěným laserovým ukazovátkem. Disponuje všesměrovým ovladačem a dvěma tlačítky pro ovládání kurzoru myši, otočným tlačítkem pro listování nebo nastavení hlasitosti a třemi multifunkčními tlačítky. Pracuje ve dvou módech: prezentačním, kdy lze snadno ovládat laserové ukazovátko, posouvat zobrazení prezentace a listovat stránkami, nebo multimediálním, kdy umožňuje spouštět a zastavovat přehrávání audio a video souborů a regulovat hlasitost.



Obrázek 6.1: Genius Wireless Pointer

Protože je zařízení navrženo pouze pro tento jeden účel, má lépe vyřešenou ergonomii držení a ovládání. Naopak nevýhodou tohoto řešení je nutnost zapojení přijímače do USB portu počítače a také to, že se jedná o další zařízení, které musí prezentující nosit s sebou. Další nevýhodou může být cena. Navíc ne pro každého může být ovládání směrovým tlačítkem intuitivní.

6.2 Softwarová řešení

Další produkt, který umožňuje podobné funkce, je program *ShowDirector 2.0 for PowerPoint* od Signal Beach Software (<http://www.signalbeach.com>). Tento program ovládá prezentaci přes webový prohlížeč a na straně kapesního počítače se tedy nejedná o samostatnou aplikaci. Je ale nutné být připojen pomocí Bluetooth nebo Wi-Fi k internetu, což není dostupné vždy. Další nevýhodou je to, že spolupracuje pouze s programem Microsoft PowerPoint. Dokáže však kromě klasického přechodu mezi snímky také pomocí kapesního počítače upravovat snímky prezentace a ke každému snímku mít na PDA poznámku, kterou nechce prezentující zapomenout říct. Program také není zdarma.

6.3 Srovnání

Tato bakalářská práce nemá tak rozsáhlý výběr funkcí, jako výše popsané produkty, dostačují však k plnohodnotnému ovládní prezentace. Velkou výhodou oproti oběma zmíněným však vidím v možnosti úplného si přizpůsobení rozložení a funkčnosti ovládacích tlačítek. Tato tlačítka přitom jsou pro ovládní prezentace nejdůležitější. Další nespornou výhodou oproti výše popsaným produktům je cena, aplikace je dostupná zdarma. Navíc oproti programu *ShowDirector* není závislá na jednom konkrétním programu. Dokáže ovládat všechny programy, které lze ovládat klávesnicí, případně jednoduše myší.

Kapitola 7

Budoucí vývoj

Tato bakalářská práce poskytuje úplný výčet funkcí, které jsou potřebné pro ovládání prezentace. Zároveň je zde však velký prostor pro rozšíření. Zde je seznam možností, které by nějakým způsobem mohly tuto práci vylepšit nebo rozšířit:

- Jako jedna z hlavních možností rozšíření této aplikace je spojení kapesního a stolního počítače přes síť Wi-Fi. Modul Wi-Fi je nyní již ve většině notebooků vestavěný a pro mnoho uživatelů, kteří nemají vestavěný Bluetooth modul, by to bylo usnadnění. Funkce pro ovládání prezentace zůstanou beze změny, pouze se přidá dialog pro nastavení spojení a přidají se položky v uživatelském profilu.
- Hlavní nedostatek v jiných řešeních, popsaných v kapitole 6, vidím v nemožnosti používat program v jiném operačním systému, než je Microsoft Windows. Proto by bylo přínosné implementovat aplikaci také pro operační systém Linux, aby i uživatelé tohoto operačního systému mohli prezentaci snadněji ovládat. Jednalo by se pouze o serverovou část aplikace, část na PDA by zůstala beze změny. Použita by mohla být architektura X Window a knihovna Xlib, která pro tento účel poskytuje všechny potřebné funkce.
- Co se týče ovládání programu, bylo by pro prezentujícího příjemné doplnit funkci poznámek ke každému snímku, kterou obsahuje zmíněný program *ShowDirector*. Prezentující by tak u každého snímku viděl na svém PDA text, který by chtěl navíc ke snímku říct.
- Dalším vylepšením, které by udělalo program ještě více univerzálním, by byla možnost ručně nadefinovat klávesové zkratky. Toto rozšíření by vyžadovalo poměrně rozsáhlé změny v aplikaci, avšak program by pak dokázal ovládat libovolnou jinou aplikaci.

Kapitola 8

Závěr

Během vývoje této práce jsem postupoval přesně podle kroků, které mi určovalo zadání. Seznámil jsem se s mobilní platformou Pocket PC a nastudoval principy tvorby programů pro kapesní počítače na této platformě. Dále jsem navrhl způsob komunikace mezi kapesním a stolním počítačem. Komunikace těchto dvou zařízení probíhá pomocí technologie Bluetooth.

Následně jsem provedl analýzu a návrh aplikace, která by pomocí kapesního počítače dokázala ovládat prezentaci, spuštěnou na stolním počítači. Tuto aplikaci jsem podle provedeného návrhu implementoval. Aplikace poskytuje všechny funkce, které jsou pro ovládání prezentace potřebné. Nad rámec zadání jsem přidal možnost uložení vlastního profilu a uživateli tedy stačí jednou si program nastavit podle svých představ a při dalších spuštěních může vše obnovit jedním kliknutím. Jako další rozšíření jsem přidal možnost volby denního a nočního režimu a uživatel tak může přizpůsobit vzhled aplikace aktuálnímu prostředí. Aplikace je v reálu použitelná a dokáže lidem usnadnit a zpříjemnit prezentování. Dokazuje to i odezva uživatelů, kteří program využívají a od kterých dostávám návrhy na další možná vylepšení a rozšíření

Práce na tomto projektu pro mě byla velmi přínosná, neboť jsem se při ní naučil v podstatě od základů programování pomocí funkcí, které poskytuje WinAPI. Tím jsem zároveň lépe pochopil, jak funguje komunikace v operačním systému Microsoft Windows prostřednictvím zasílání zpráv. Jsem také rád, že jsem si vyzkoušel tvorbu programu pro kapesní počítače, která je velice zajímavá a podle mého názoru velice perspektivní. Zajímavé bylo také zjišťování možností v komunikaci přes Bluetooth. Zároveň jsem si vyzkoušel tvorbu rozsáhlejšího programu a uvědomil si při tom, že pro vývoj většího projektu pouhé programování nestačí, ale že je velmi důležité vytvořit si na začátku kvalitní návrh.

Vývoji aplikace bych se chtěl věnovat i v budoucnu a rád bych implementoval funkce, které jsem popsal v kapitole 7. Program je volně šiřitelný a ani v budoucnu neplánuji jeho komerční využití.

Seznam použitých zdrojů

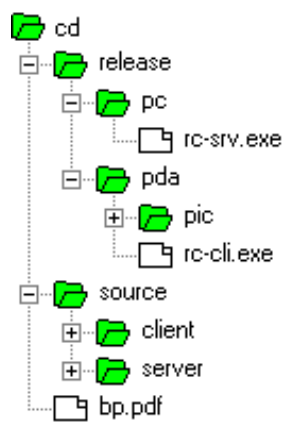
- [1] eXEeden - CE4YOU: *Úvod do programování pro Windows Mobile 2002 a 2003 - Vývojové nástroje* [ONLINE]. Poslední modifikace 18. září 2003. [cit. 2007-04-20]. Dostupné na URL: <http://www.ce4you.cz/articles/detail.asp?a=54>
- [2] Kobe - specialista na mobilní technologie: *Popis Bluetooth* [ONLINE]. [cit. 2007-05-06]. Dostupné na URL: http://www.kobe.cz/inshop/scripts/show.asp?page=technologie_bluetooth.htm&rnd=2964950
- [3] PETZOLD, C.: *Programování ve Windows*. Computer Press, první vydání, 1999, ISBN 80-7226-206-8, 1216 s.
- [4] The Official Bluetooth Membership Site: *Bluetooth* [ONLINE]. Poslední modifikace 2007. [cit. 2007-05-06]. Dostupné na URL: <http://www.bluetooth.org>
- [5] TKÁČ J., ZAORAL O.: *Průvodce světem kapesních počítačů aneb PDA na dlani*. Grada Publishing, první vydání, 2005, ISBN 80-247-1227-X, 205 s.
- [6] Wikipedia: *Client-server* [ONLINE]. Poslední modifikace 4. května 2007, 9:37. [cit. 2007-05-06]. Dostupné na URL: <http://en.wikipedia.org/wiki/Client/server>
- [7] Wikipedia: *GNU Compiler Collection* [ONLINE]. Poslední modifikace 6. května 2007, 5:35. [cit. 2007-05-06]. Dostupné na URL: http://en.wikipedia.org/wiki/GNU_Compiler_Collection
- [8] Wikipedia: *Personal digital assistant* [ONLINE]. Poslední modifikace 4. května 2007, 9:37. [cit. 2007-05-06]. Dostupné na URL: http://en.wikipedia.org/wiki/Personal_digital_assistant
- [9] Wikipedia: *WinAPI* [ONLINE]. Poslední modifikace 4. května 2007, 9:37. [cit. 2007-05-06]. Dostupné na URL: <http://en.wikipedia.org/wiki/WinAPI>
- [10] ŽIVNÝ, P.: *Historie Pocket PC* [ONLINE]. [cit. 2007-04-20]. Dostupné na URL: <http://www.pdasoft.cz/modules.php?name=News&file=article&sid=1794>

Seznam příloh

- A Obsah CD
- B Uživatelská příručka

Příloha A

Obsah CD



Obrázek A.1: Adresářová struktura

- Složka `release` - obsahuje spustitelné soubory aplikace pro stolní i kapesní počítač
- Složka `source` - obsahuje zdrojové kódy klientské i serverové části včetně projektových souborů do Microsoft Visual Studia, ze kterých je možné spustitelné soubory přeložit a sestavit
- `bp.pdf` - textová část práce

Příloha B

Uživatelská příručka

Tato příručka obsahuje návod na zprovoznění aplikace a popisuje způsob jejího používání. Na přiloženém CD jsou umístěny jak zdrojové kódy programu, tak i přeložené programy, připravené k okamžitému spuštění.

Požadavky na systém

Serverová část je spustitelná v operačním systému Microsoft Windows (testováno na Microsoft Windows 2000 a Windows XP). Klientská část požaduje pro své spuštění Microsoft Windows Mobile (testováno na WM2003, 2003SE a 2005).

Přeložení programu

Zdrojové kódy programu jsou umístěny na CD v adresáři `source`.

Serverová část je vyvíjena v prostředí Microsoft Visual Studio 2005. Po otevření projektu v tomto prostředí a následném přeložení se vytvoří spustitelný soubor pro PC.

Klientská část je vyvíjena v prostředí Microsoft eMbedded Visual Studio 2003 a po přeložení zdrojových kódů v tomto prostředí vznikne spustitelný soubor pro PocketPC. Pro správnou funkčnost je však ještě potřeba zkopírovat do PDA adresář `pic`, který je rovněž na CD umístěn. Tento adresář musí být v PDA na stejné adresářové úrovni, jako spustitelný soubor.

Instalace programu

Program není třeba instalovat, stačí pouze nahrát příslušné soubory buď na PC nebo na PDA a spustit je. Spustitelné soubory jsou na CD umístěny v adresáři `release`.

Adresář `PC` obsahuje jeden soubor `rc-srv.exe` – serverová část programu.

Adresář `PDA` obsahuje soubor `rc-cli.exe` a adresář `pic`. Jedná se o klientskou část programu. Soubor i adresář musí být v PDA na stejné adresářové úrovni. V opačném případě se po spuštění zobrazí pouze bílé pozadí bez tlačítek.

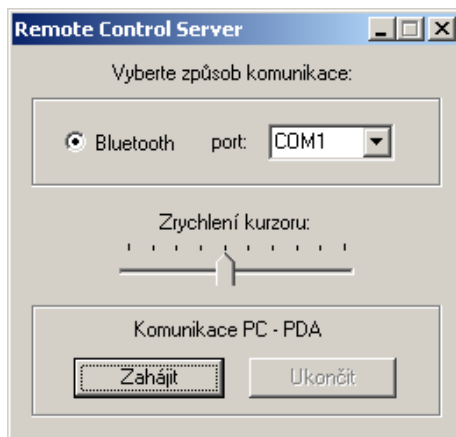
Spuštění a ovládání

Při komunikaci dvou zařízení přes Bluetooth je potřeba tato zařízení nejprve spárovat (tzv. vytvořit partnerství). To se provede pomocí průvodce nastavení, který se objeví po zapnutí

Bluetooth.

Serverová část - PC

- spustit na PC serverovou část (obrázek B.1)
- vybrat Bluetooth port, který používá PC na sériovou komunikaci (lze nalézt např. v aplikaci, která je dodávána společně s ovladačem pro dané bluetooth zařízení)
- kliknout na *Zahájit*



Obrázek B.1: Serverová část aplikace

Klientská část - PDA

- spustit na PDA klientskou část
- vybrat Bluetooth (pokud není Bluetooth zapnuté, program jej zapne) (obrázek B.2)
 - vybrat Bluetooth port, který používá PDA na sériovou komunikaci a kliknout na *Start* (obrázek B.3)
 - pokud už je uložený profil, lze ho vyvolat kliknutím na *Můj profil*
 - program se může pokusit vyhledat port automaticky (není 100%)
- po úspěšném zahájení komunikace se objeví obrazovka s ovládacími tlačítky a s menu pro nastavení (obrázek B.4)

Popis menu

- první položka v menu
 - **uložit profil** - uloží aktuální profil (Bluetooth port, rozložení šipek, denní/noční režim)
 - **komunikace** - přechod na úvodní obrazovku s výběrem komunikace
 - **konec** - ukončí program
- druhá položka v menu - zobrazí dialog s uživatelským nastavením
- třetí položka - zobrazí informace o programu



Obrázek B.2: Zvolení způsobu komunikace



Obrázek B.3: Nastavení komunikace přes Bluetooth



Obrázek B.4: Hlavní obrazovka s ovládacími tlačítky a menu