

VŠB-Technická univerzita Ostrava

Fakulta elektrotechniky a informatiky

BAKALÁŘSKÁ PRÁCE

2010/2011

Antonín Kučera

VŠB-Technická univerzita Ostrava

Fakulta elektrotechniky a informatiky

Katedra měřicí a řídicí techniky

**Mobilní aplikace pro vzdálený monitoring fyzikálních
veličin pomocí SMS, MMS nebo GPRS komunikace**

**Mobile Application for Remote Monitoring of Physical
Quantity using SMS, MMS or GRPS Communication**

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Antonín Kučera

Datum odevzdání bakalářské práce: 6. 5. 2011

Poděkování

Touto cestou bych chtěl poděkovat vedoucímu mé bakalářské práce panu Ing. Ondřeji Krejcarovi, Ph.D. a jeho kolegovi Ing. Vladimíru Kašíkovi, Ph.D. za poskytnutí materiálů, cenných rad a konzultací, které mi ochotně během tvorby bakalářské práce poskytli. Dále chci poděkovat Vlastě Kubíkové za pomoc s korekturou a opravou stylistických chyb.

Abstrakt

Tato práce spočívá ve vyvinutí aplikace pro mobilní telefony, která pomocí zobrazených hodnot na dotykovém displeji jednoduše zobrazí aktuální stav na vzdáleném zařízení. Zvýšení komfortu je dosaženo vizualizací aplikace pomocí vytvořených komponent. Zpracováním snadno pochopitelné a ovladatelné aplikace, jež je aktuálně navržena pro GSM komunikátor GD-06 Allegro, je usnadněné ovládání vzdáleného zařízení. Druhá část práce se zaměřuje na přesně opačně fungující aplikaci, která je v přenosném zařízení PDA. Aplikace simuluje GSM komunikátor a chová se jako vzdálené zařízení. Obě dvě aplikace jsou navrženy a konstruovány pro platformu .NET Compact Framework.

Klíčová slova

.NET Compact Framework, vzdálené zařízení, mobilní telefon, GSM komunikátor

Abstract

This work is in developing applications for mobile phones that use the values displayed on the touch screen simply displays the current status of the remote device. Increased comfort is achieved by visualization applications by using components created . Processing and easy-to-use application that is currently designed for GSM communicator GD-06 Allegro, is facilitated by a remote control device. The second part is exactly the opposite functional application that is portable PDA. The application simulates a GSM communicator, and acts as a remote device. Both applications are designed and constructed for the platform. NET Compact Framework.

Keywords

.NET Compact Framework, remote, mobile phone, GSM communicator

Seznam použitých zkratek a symbolů

- PDA - Zkratka pro kapesní počítač z anglického spojení *Personal Digital Assistant*
- GSM - (*Global System for Mobile Communications*) komunikační síť pro mobilní telefony
- SMS - (*Short Message Service*) krátká textová zpráva
- MMS - (*Multimedia Messaging Service*) multimediální zpráva
- 3G/GPRS - (*General Packet Radio Service*) typ datové komunikace
- Kb - (*Kilobyte*) Velikost posílaných a přijímaných dat
- Atd. - A tak dále, je možno více variant
- txt - Textový dokument
- Tel. - Telefonní (například: Telefonní číslo)
- LED - (*Light-Emitting Diode*) dioda emitující světlo
- Wi-fi - (*Wireless LAN*) bezdrátová komunikace mezi počítači a mobilními zařízeními
- IP - (*Internetový protokol*) IP adresa- identifikační číslo
- SIM - (*Subscriber identity module*) identifikační karta účastníka v mobilní síti
- PC - (*Personal computer*) osobní počítač užívaný jednotlivcem
- WM - (*Windows Mobile*) operační systém pro mobilní zařízení
- SQL - (*Structured Query Language*) používaný pro práci s daty v databázích
- FTP - (*File Transfer Protocol*) protokol pro přenos souborů a dat
- Pdf - (*Portable Document Format*) typ uložení textového dokumentu
- HMI - (*Human Machine Interface*) uživatelské rozhraní mezi člověkem a přístrojem
- OS - Operační systém
- MHz - (*Mega Hertz*) jednotka frekvence
- PIN - (*Personal identification number*) identifikační číslo sloužící k přihlašování
- GND - (*Ground*) označení uzemění v elektrotechnice
- ČTÚ - Český telekomunikační úřad
- ČSN EN - Označení československých státních norem
- ON - Zapnuto
- OFF - Vypnuto
- MAN - Manuální chod
- AUTO - Automatický chod
- FALSE - Označení proměnné v hodnotě nepravda nebo vypnuto
- TRUE - Označení proměnné v hodnotě pravda nebo zapnuto
- STRING - Textový typ uložení proměnné

Seznam použitých obrázků

Obrázek 1 - Ukázka použitelných verzí v postupu vývojem systému Windows	2
Obrázek 2- Ilustrační obrázek komunikátoru	7
Obrázek 3 - Zobrazení GSM komunikátoru GD-06 <i>Allegro</i>	10
Obrázek 4- vzhled komunikátoru <i>Allegro</i>	10
Obrázek 5- Závislost izolační mezery na velikosti napětí[17]	12
Obrázek 6 – Původní a současný Vzhled Aplikace	14
Obrázek 7 – Blokové schéma aplikace	15
Obrázek 8-Výpis použitých metod	15
Obrázek 9 – Přidávání telefonního čísla	16
Obrázek 10 - Přepínač mezi manuálním a automatickým chodem.	17
Obrázek 11 - Nastavování hodnoty teploty	17
Obrázek 12 – Komponenta ALLEGRO.....	18
Obrázek 13 – Komponenta Svorkovnice	19
Obrázek 14- Blokové schéma ovládané aplikace.....	21
Obrázek 15- Aplikace nahrazující GSM komunikátor.....	22
Obrázek 16 – Komponenta SMS	23
Obrázek 17- Komponenta Teploměr.....	23

Obsah

1.	Úvod	1
1.1.	Co je to PDA.....	1
2.	Komunikace pomocí SMS, MMS, GPRS rozhraní v jazyku C# a platformě .NET Compact.....	2
2.1.	Framework na mobilních zařízeních. Platforma .NET Framework	2
2.2.	Platforma .NET Compact Framework	2
2.3.	Komunikace přes GSM síť	3
3.	Analýza možností reálné využitelnosti SMS, MMS a GPRS na mobilních zařízeních pro vzdálený monitoring.	5
3.1.	Použití SMS.....	5
3.2.	Použití MMS	5
3.3.	Použití GPRS	6
4.	Seznámení s GSM komunikátory	7
4.1.	Další dostupné produkty na trhu a jejich porovnání	7
4.2.	Použití GSM komunikátoru <i>Allegro GD-06t</i>	10
4.3.	Druhy komunikací s GSM komunikátorem	11
4.4.	Technické parametry komunikátoru	11
5.	Aplikace pro mobilní zařízení realizující monitoring vzdáleného zařízení.....	13
5.1.	Tvorba komponent	13
5.2.	Řešení aplikace pro ovládání.....	13
5.3.	Komunikace mezi aplikací a GSM komunikátorem	20
6.	Aplikace pro mobilní zařízení nahrazující GSM komunikátor	21
6.1.	Definice problému	21
6.2.	Řešení ovládané aplikace.....	22
6.3.	Vlastní realizované komponenty pro vzdálené řízení.....	23
7.	Testování vyvinutého řešení dle navržené metodiky.....	24
7.1.	Navržení metodiky.....	24
7.2.	Black Box	24
7.3.	White Box.....	26
8.	Diskuze výsledků	28
9.	Závěr.....	30
10.	Literatura:	31

1. Úvod

Díky stále se rozšiřujícímu trhu s PDA a jejich četnému využití existuje mnoho způsobů komunikace mezi jednotlivými zařízeními. PDA nemusí komunikovat jen mezi sebou, ale i se vzdálenými zařízeními a GSM komunikátory. Možnosti komunikace jsou pomocí SMS, MMS, GPRS atd.

Vytvářením aplikací pomocí platformy .NET Compact Framework lze využít velké množství funkcí daného PDA. Zároveň máme velkou možnost využít grafického prostředí, které nám umožňuje tato platforma. Pomocí vizualizací a grafických komponent je ve výsledné fázi aplikace snadno a rychle pochopitelná a jednoduše ovladatelná. Pomocí vytvořené aplikace lze komunikovat s GSM komunikátory. Podle toho k jakému účelu a pomocí jakého protokolu aplikace funguje, můžeme přesně specifikovat jakýkoli proces.

Předmětem mojí bakalářské práce je v první části vytvořit jednoduchou univerzální aplikaci pro komunikaci a ovládání vzdáleného zařízení. Podle zadání pracuji s komerčním GSM komunikátorem Allegro GD-06. Pomocí graficky zpracované aplikace snadno a rychle komunikuji pomocí SMS.

V druhé části se budu zabývat vytvořením obrácené aplikace, která simuluje GSM komunikátor a k PDA, ve kterém by měla být spuštěna vytvořená aplikace, je připojené ovládané zařízení (topení).

1.1. Co je to PDA

Zkratka PDA pochází z anglického spojení Personal Digital Assistant, které v překladu znamená osobní digitální pomocník. Ovládání se realizuje z velké části pomocí dotykového displeje podle provedení buď pomocí dotykového pera stylus, nebo přímo prsty na ruce. V současné době se v PDA používají jako operační systémy: Windows Mobile, PalmOS, Symbian OS a Linux Android. [1]

V počátku vývoje elektronických pomocníků vznikaly jen Databanky, které sloužily k organizaci kontaktů a krátkých poznámek. Vývojem dalších technologií a komunikací jsou PDA. První modely se dají zařadit mezi multifunkční kalkulačky nebo elektronické diáře, které se skládaly z malé klávesnice, monochromatického displeje a z velmi malé paměti. Postupem času byla zvětšována interní paměť a v pozdější době vkládány i externí paměťové karty. Displeje byly zvětšovány a z monochromatických se přešlo na barevné a dotykové. Mimo to vývojáři zvyšovali výkonnost procesoru a přidávali různé možnosti komunikace. Jako jedna z prvních možností byl Infraport, následně i Bluetooth, Wi-fi, a samozřejmě i GSM komunikace.

2. Komunikace pomocí SMS, MMS a GPRS v rozhraní jazyku C# a platformě .NET Compact

Na začátku této kapitoly se seznámíme s teoretickým rozbohem vývojového prostředí a jeho možnostmi využití. V další části je čtenáři nastíněno, jaké existují druhy komunikací a jejich základní popis s možnostmi využití.

2.1. Framework na mobilních zařízeních. Platforma .NET Framework

.NET Framework je základní prostředí potřebné pro běh aplikací, jež nabízí potřebné rozhraní a také potřebné dynamické knihovny. Je to nejrozšířenější platforma pro použití v osobních počítačích s operačním systémem Microsoft Windows. Poprvé byla využita v operačním systému Windows 98. [2]

.NET verze	1.0	1.1	2.0	3.0	3.5	4.0
Windows 95	nelze	nelze	nelze	nelze	nelze	nelze
Windows NT	lze doinstalovat	lze doinstalovat (SP6a)	nelze	nelze	nelze	nelze
Windows 98, Windows 98 SE	lze doinstalovat	lze doinstalovat	lze doinstalovat	nelze	nelze	nelze
Windows Me	lze doinstalovat	lze doinstalovat	lze doinstalovat	nelze	nelze	nelze
Windows 2000	lze doinstalovat	lze doinstalovat	lze doinstalovat (SP3)	nelze	nelze	nelze
Windows XP	lze doinstalovat	lze doinstalovat	lze doinstalovat (SP2)	lze doinstalovat (SP2)	lze doinstalovat	lze doinstalovat (SP3)
Windows Server 2003	??	součást systému	lze doinstalovat	lze doinstalovat (SP1)	lze doinstalovat	lze doinstalovat (SP2)
Windows Vista	částečná kompatibilita	částečná kompatibilita	součást systému	součást systému	lze doinstalovat	lze doinstalovat (SP1)
Windows Server 2008	??	??	??	součást systému	lze doinstalovat	lze doinstalovat
Windows Server 2008 R2	??	??	??	??	součást systému	lze doinstalovat
Windows 7	částečná kompatibilita	částečná kompatibilita	součást systému	součást systému	součást systému	lze doinstalovat

Obrázek 1 - Ukázka použitelných verzí v postupu vývojem systému Windows

2.2. Platforma .NET Compact Framework

Platforma .NET Compact Framework je navazující produkt, který byl vytvořen pro použití v kapesních počítačích nebo mobilních telefonech s operačním systémem Windows Mobile. Tato platforma má s platformou pro osobní počítače mnoho společných věcí. Veškeré knihovny jsou použitelné v obou verzích až na několik výjimek, které jsou upraveny tak, aby nebyly náročné na místo, nebo jsou konstruované přímo pro dané využití [3]

Druhů programů, v nichž lze vyvíjet aplikace s použitím .NET Compact Framework, je mnoho. Tyto programy vytváří společnost Microsoft a používají se např. tyto: Visual Studio.NET 2003, Visual Studio 2005, Visual Studio 2008, Visual Basic.NET. Aplikace vytvořené v některém z těchto uvedených programů jsou použitelné pro speciální mobilní zařízení, která mají o něco vyšší výkon. Je možné je použít i ve starších modelech s nižším výkonem, ale není zaručena plynulost aplikace a její správná funkčnost.

Historie vzniku platformy .NET

Microsoft začal vyvíjet platformu ke konci roku 1990 původně pod názvem Nová generace Windows. Do konce roku 2000 vyšla nová verze, zatím jako beta, a byla testována pro odstranění chyb. V roce 2002, ve vývojovém prostředí Visual Studio .NET, byl uveden do provozu jazyk C#. V současné době se na trhu nachází verze 4.0, která byla spuštěna v roce 2010. Navržená verze Visual Studia byla spojená se stejnou verzí .NET Framework. Pro zařízení s Windows Mobile odpovídají z počátku stejné verze, ale postupem vývoje se došlo jen k verzi 3.7 s mnohem větším množstvím Beta verzí.

2.3. Komunikace přes GSM síť

GSM síť je digitální buňková radiotelefonní síť, pomocí níž můžeme přenášet hovory, SMS a data formou MMS, internetu (GPRS) a faxu. Buňková síť se jí říká proto, že ji tvoří buňky (přístupové body), na které se připojujeme pomocí mobilních telefonů, PDA a dalších typů zařízení.

2.3.1. Komunikace pomocí SMS v C#

Využitím dynamických knihoven, vytvořených pro Windows Mobile, jsme schopni posílat a přijímat SMS. Definování zdrojového kódu na posílání SMS je značně jednodušší než pro příjem. Každá funkce využívá jiné dynamické knihovny.

Pomocí metody SmsMessage() z dynamické knihovny PocketOutlook.dll byla vytvořena nová SMS a následně i její tělo. Pomocí metody Send() byla poté nová SMS zpráva odeslána na předem definované telefonní číslo. [4]

```
SmsMessage smsMessage = new SmsMessage();
smsMessage.Body = "Zde je text, který se má poslat.";
smsMessage.To.Add(new Recipient("Jméno kontaktu", "Telefonní číslo"));
smsMessage.RequestDeliveryReport = true;
smsMessage.Send(); //Metoda na posílání SMS.
return;
```

Příjem SMS v aplikaci definuje dynamická knihovna InteropServices.dll, jež se na rozdíl od PocketOutlook.dll nemusí přidávat do referencí, ale stačí knihovnu pouze přidat tak, aby aplikace věděla, že ji má použít a sama si ji v příslušném zařízení zavolá. Kód pro čtení SMS se dělí třemi způsoby. První možností je, že aplikace přečte SMS podle určeného jména v adresáři. Druhou variantou je to, že aplikace přečte jen ty příchozí SMS podle daného telefonního čísla. Při třetím způsobu čte SMS podle příchozího textu. Když je začátek SMS stejná jako předdefinovaný text, tak přečte přijatou SMS.[5]

```
void SmsInterceptor_MessageReceived_OnThread(object sender, MessageInterceptorEventArgs e)
{
    SmsMessage newMessage = e.Message as SmsMessage;
    if (newMessage != null){
        textBox_read.BeginInvoke((MethodInvoker)delegate {
            textBox_read.Text = newMessage.Body;
        });
        Debug.WriteLine(string.Format("Sender: {0} - Body: {1}",
            newMessage.From.Address, newMessage.Body));
    }
}
```

2.3.2. Komunikace pomocí MMS

Komunikace je velmi podobná jako u obyčejných SMS pouze s tím rozdílem, že do MMS se vejde podstatně více textu, můžeme přiložit obrázky, video soubory a hudební soubory. Zatímco zprávy SMS jsou posílány přímo po GSM pásmu, tak MMS jsou posílány pomocí GPRS, tedy jako datový tok. Funkce posílání MMS určuje knihovna IMessage [6]. U vytváření MMS se definuje nejen tělo a číslo příjemce, ale také jaký soubor má být přidán, smazání staré přílohy a přidání nové.

```
LPMAPITABLE pContentsTable = NULL;
LPSRowSet pRows = NULL;
HRESULT hr = _message->GetAttachmentTable(0, &pContentsTable);
if(FAILED(hr)) {
    return res;
}
ULONG amt = 0;
pContentsTable->GetRowCount(0, &amt);
while(!FAILED(hr))
{
    hr = pContentsTable->QueryRows(1, 0, &pRows);
    if(FAILED(hr) || pRows->cRows == 0)
    {
        break;
    }
}
```

2.3.3. Komunikace pomocí GPRS

V první řadě musí být karta SIM v komunikátoru nastavena tak, že je povolena funkce 3G/GPRS. V dalším kroku musí být komunikátor zaregistrován na serveru www.gsmlink.cz. Pro komunikaci aplikace se zařízením je nutné otevřít spojení po internetu a mít nastavenou IP adresu. Pro běžné nastavení by nebylo možné se navzájem propojit se zařízeními určenými pro plné ovládání a komunikaci. [7] V ukázce kódu vidíme jak nastavit zařízení pro správnou funkčnost.

```
public static string SetDDNS_Prov(string UserID , string Password ){
    System.Xml.XmlDocument d= new System.Xml.XmlDocument();
    String provString = "<wap-provisioningdoc>";
    provString += "<characteristic type='CM_Mappings'>";
    provString += "<characteristic type='900'>";
    provString += "<parm name='Pattern' value='*://'" + UserID + "':" + Password +
    provString += "<parm name='Network'
    value='{436EF144-B4FB-4863-A041-8F905A62C572}'/>";
    provString += "</characteristic>";
    provString += "</characteristic>";
    provString += "</wap-provisioningdoc>";
    provString = provString.Replace("'", Convert.ToChar(34).ToString());
    Debug.WriteLine("provString = " + provString);
    d.LoadXml(provString);
}
```

3. Analýza možností reálné využitelnosti SMS, MMS a GPRS na mobilních zařízeních pro vzdálený monitoring

V této kapitole je vysvětleno reálné využití zadaných komunikací. V každé podkapitole je řešen jeden druh komunikace a vysvětlen její princip fungování. Na konci každé kapitoly jsou ukázky zdrojových kódů, pomocí kterých jednotlivé komunikace fungují.

3.1. Použití SMS

Použití komunikace se vzdáleným zařízením pomocí SMS je nejjednodušší řešení v psaní zdrojového kódu aplikace. Pro domácí použití, kde bývá řízeno jen několik procesů pomocí příkazů, které se vejdu do jedné SMS, je to ideální řešení. Pomocí aplikace je nadefinován kód, který je poslán do vzdáleného zařízení. Na komunikátoru jsou provedeny všechny požadované příkazy a vytvořena odpověď, kde je vypsán stav na zařízení. Aplikace si tuto SMS přečte a podle hodnot nastaví vizuální prvky tak, aby odpovídaly skutečnému stavu. Výhoda použití této komunikace oproti ostatním spočívá v jednoduchosti programování. Kódem na odesílání stačí jen nadefinovat, jaký text se má poslat. Definovaný text lze vyčíst z proměnných nebo grafických funkčních prvků jako jsou např. textbox, listbox atd. Za pomoci druhé části kódu získáme z proměnné uložené telefonní číslo. Nevýhoda komunikace pomocí SMS tkví v omezení kapacity zprávy. V rámci jedné SMS můžeme poslat pouze 170 znaků.

V teoretickém popisu byl vložen zdrojový kód tak, jak je napsán v originále, ale v reálném využití byl zkrácen a upraven.

```
SmsMessage s = new SmsMessage(Convert.ToString(t_cislo), text
```

První řádek řeší, jak už bylo zmíněno, výčet telefonního čísla z proměnné *t_cislo*. Do globální proměnné *text* typu STRING jsou ukládány požadované příkazy a následně používány.

```
s.Body = this.text; //Tvorba tela sms
```

Příkazem *Body* vytvoříme tělo zprávy. Bez tohoto příkazu by nebylo možné text poslat, protože je zde z načtené proměnné použito a uloženo k odeslání.

```
s.Send();
```

Posledním řádkem je volána knihovna *Telephone.dll*, jež využívá vnitřní struktury mobilního zařízení a pošle zprávu s nadefinovaným textem na určené telefonní číslo.

3.2. Použití MMS

Řízení pomocí MMS je ideální v případě, jestliže je k napsání ovládacího kódu zapotřebí více textových příkazů než se vejde do jedné SMS. MMS komunikace se vzdáleným zařízením je lepší využívat v opačné komunikaci a to tak, že ovládací aplikaci je poslán příkaz pomocí SMS, ale odpověď od komunikátoru přijde v MMS. Tento způsob provedení je výhodný při

posílání fotek z GSM komunikátoru. Toto řízení lze využít např. při zabezpečení objektu, kde se komunikátor nachází. Při zaznamenání pohybu je vyfocena fotografie a poslána majiteli. Složitostí programování je podstatně náročnější než SMS. Reálné využití komunikace s GSM komunikátorem je proveditelné, ale neekonomické. Kapacita jedné MMS je průměrně 300Kb, ale když komunikátor odešle odpověď tímto způsobem, nenaplní obsah zprávy a tím se stává posílání ztrátovým. Po každém odeslání jedné MMS se účtuje stejná cena a nezáleží na velikosti posílaného množství dat. Tvoření algoritmu, který zajišťuje správnost odesílání MMS, je velmi podobný kódu pro SMS.

```
UriBuilder urlBuilder = new UriBuilder();
urlBuilder.Query = string.Format("PhoneNumber=%2B" + PhoneNumber +
"&MMSFrom=YourChoiceofName&MMSSubject=" + subject + "&MMSText=" +
message + "&MMSFile=http://127.0.0.1/" + fileName);
```

U MMS existuje funkce UriBuilder, která pracuje na podobném principu jako u SMS SMSMessage, s níž se vytvářely nové zprávy. Oproti SMS musíme do příkazu na tvoření MMS definovat kromě telefonního čísla a textu zprávy navíc soubor, jenž má být poslán, a jeho název pro správnost odeslání. [8]

3.3. Použití GPRS

Typ ovládání vzdáleného zařízení pomocí GPRS je nejvhodnější v průmyslu, kde při řízení složitějších procesů potřebujeme rychlost odezvy a přesnost aktuálních dat. Při řízení náročnějších procesů obsahuje výstup z GSM komunikátoru velké množství dat, která by se nevešla do SMS, a tudíž by bylo posílání MMS velmi nákladné. GPRS je datový kanál, kde se platí za poslaný/stažený Kb. Do jednoho Kb se vejde přibližně 16-17 znaků textu.[8] V porovnání s SMS je to nejlevnější komunikace. Do jedné SMS se vejde 170 znaků, což znamená, že jedna zpráva má velikost 10-11Kb, a to je v posílání pomocí GPRS podstatně levnější. V porovnání s MMS má výhodu, že se platí za poslané množství dat a ne za poslanou zprávu. Díky této komunikaci není problém odesílat jak Byte, tak i Kb. Při navazování spojení pomocí GPRS je nejdříve nutné otevřít správce připojení.

```
ConnectionManager connMgr = new ConnectionManager();
```

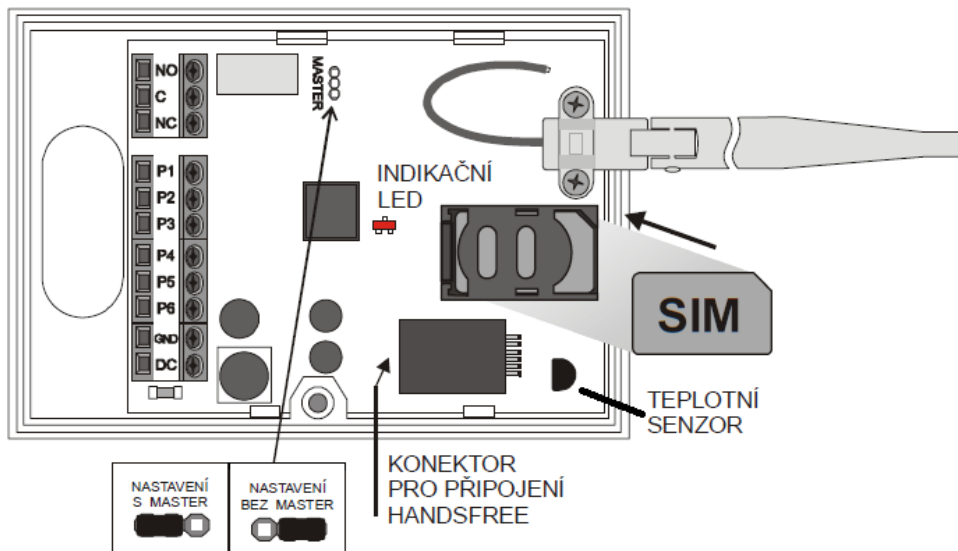
Máme-li otevřeného správce připojení, musíme zvolit typ možného připojení, po jehož volbě určíme, kam se má navázat.

```
DestinationInfoCollection DIC = connMgr.EnumDestinations();
if (di.description == "Internet")
{
    connMgr.Connect(di.guid, true, ConnectionMode.Asynchronous);
}
```

Po funkčním připojení do internetové sítě komunikuje aplikace pomocí GPRS protokolu se vzdáleným zařízením. Propojení lze zajistit přes různé programy, např. skrz WinLoad, nebo pomocí SQL databáze či ukládání přes FTP server.

4. Seznámení s GSM komunikátory

GSM komunikátory využívají, stejně jako mobilní zařízení, ke komunikaci síť GSM, jak už také vyplývá z názvu. Přímou v zařízení nalezneme slot na SIM kartu. Komunikátory se liší tím, že oproti mobilním telefonům nemají displej ani žádná tlačítka na psaní SMS či něco podobného. Jejich konstrukce se skládá z antény, indikační LED diody a mnoha ovládatelných vstupů a výstupů. Pomocí povoleného druhu komunikace můžeme řídit provoz. V základu komunikátory využívají komunikace pomocí SMS, další rozšiřující funkcí jsou MMS a GPRS. Velmi záleží na typu zařízení a konfiguraci od výrobce.



Obrázek 2- Ilustrační obrázek komunikátoru

Jako další funkci komunikátoru je důležité zmínit, že umožňuje volání jako z mobilního telefonu nebo pevné linky. Po připojení sady pro Handsfree můžeme telefonovat. Nejsme schopni z komunikátoru vytáčet čísla, ale když nám někdo volá, tak hovor můžeme přijmout nebo odmítnout. Do komunikátoru můžeme nastavit určitý počet telefonních čísel a nastavit co se má posílat na která čísla. Do některých zařízení je ještě přidávána baterie, jež slouží jako záložní zdroj energie v případě vypnutí přívodu elektřiny. Tato baterie není schopná pokrýt celý výkon komunikátoru, ale slouží tu k tomu, aby nebyla vymazána paměť.

4.1. Další dostupné produkty na trhu a jejich porovnání

GSM komunikátor - SIP300

GSM komunikátor SIP300 je vybaven celkem třemi univerzálními vstupy, na něž je možné připojit bezpotenciálový kontakt jakéhokoliv čidla či výstup libovolné zabezpečovací centrály apod. Po změně jejich stavu může být odeslána SMS s názvem vstupu (např. VLOUPANI, POZAR, VYPADEK KOTLE) a volitelně může být provedeno volání až na tři telefonní čísla. Každý ze vstupů je libovolně konfigurovatelný a nastavitelný jako 24h smyčka, která nepodléhá deaktivaci, nebo může být nastaveno příchodové a odchodové zpoždění. [10]

JA-60GSM GSM komunikátor

Modul komunikátoru nabízí ve spojení se zabezpečovacím systémem odesílání informačních SMS textových zpráv až na 8 mobilních telefonů. Z komunikátoru můžeme zavolat na nastavená telefonní čísla a přehrát akustické upozornění či předávat údaje na pult centrální ochrany. Můžeme ho dálkově ovládat a nastavovat systém z klávesnice telefonu (mobilní i pevné sítě), dálkově ovládat spotřebiče v domě z telefonu (mobilní i pevné sítě). Z připojeného telefonního přístroje lze telefonovat podobně jako z pevné linky. [11]

Level GB 060 300A








Komunikátor GB 060 3** Isis je určen pro dálkový sběr informací - dohled na stav ve vodojemech, vodních elektrárnách, čističkách odpadních vod atd. Je vybaven modulem GSM, interní záložní baterií, třemi vstupy pro přenos limitních stavů, dvěma informačními vstupy pro zjišťování stavů nebo čítání spotřeby, jedním výstupem ovládaným prozvoněním ze zadaných čísel, jedním vstupem přednastaveným pro manuální ovládání výstupu, dvěma výstupy pro spínání připojených spotřebičů, jedním výstupem pro indikační diodu LED, jednou sběrnici pro připojení 10 teploměrů, analogových vstupů, čipových klíčů nebo karet, jedním audio vstupem pro mikrofon, jedním audio výstupem pro reproduktor a sériovým kanálem RS232 pro změnu nastavení z PC a čtení reportu. Komunikátor je možné ovládat mobilním telefonem nebo modemem ze vzdáleného PC. [12]

T341L - nový GSM komunikátor

GSM komunikátor T341L lze použít i jako jednoduchou ústřednu jak pro drátové, tak i bezdrátové systémy. Má 16 digitálních drátových vstupů oddělených optočleny a dva releové výstupy s prepínacími kontakty. Můžeme uložit až 64 telefonních čísel, které lze libovolně přiřadit kterémukoliv vstupu. Dále zahrnuje 16 podsystémů, z nichž každý podsystém lze aktivovat/deaktivovat samostatně. Při použití jako bezdrátový systém výrobce doporučuje použít bezdrátový přijímač 868MHz včetně bezdrátových PIR868MHz, bezdrátových magnetů 868MHz a dalších detektorů 868MHz, které jsou dodávány jako příslušenství v krabici. [13]

GD-04 „David“

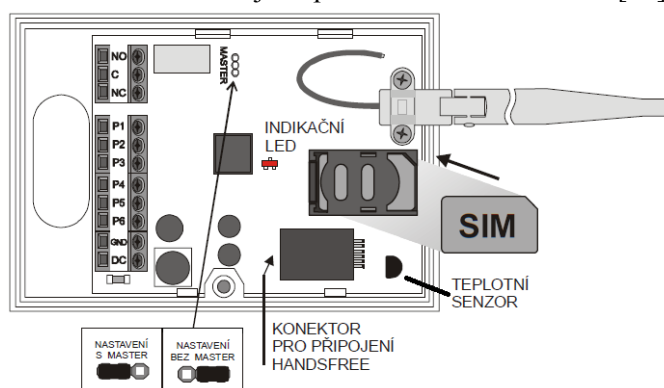
Jeho funkčnost je velmi podobná s GD-06 Allegro, ale disponuje menším množstvím vstupů a výstupů. Komunikátor obsahuje 2 výstupní kontakty silových relé X a Y (každý až 5A/250V), 4 vstupní svorky A až D pro automatické odeslání SMS zpráv (reagují na spojení nebo rozpojení se společnou svorkou GND). Můžeme ho využít jako vypínač ovládaný pomocí SMS, také ho aktivovat jako časový spínač a též jako zapínání a vypínání RELÉ. [14]

Název	Vzhled	Napájení	Odběr při komunikaci	Rozměry[mm]	Počet Svorek	Počet uložení tel. čísel	Cena Kč s DPH
GD-06 Allegro		8-15V DC	300mA	76x110x33	7	8	8 040,-
GD-04 David		11-13V DC	500mA	76x110x33	5	8	3790,-
T341L		7-14V DC	500mA	105x74x18	18	64	4920,-
GB 060 300A		9-25V DC	500mA	144 × 44 × 88	12	Min 1	8625,-
JA-60GSM		12V DC	1A	139x78x34	4	8	7788,-
SIP300		12V DC	400mA	72x56x20	7	9	4200,-
VT-21		10-16V DC	800mA	100x76x25	8	10-24	6000,-

Tabulka 1-Porovnání GSM komunikátorů

4.2. Použití GSM komunikátoru *Allegro GD-06*

Zařízení Allegro plní funkci vícekanálového universálního GSM zařízení. Řízení se uskutečňuje pomocí SMS, přes internet, pomocí GPRS či prozvoněním ze zadaného telefonního čísla. Informace o aktuálním stavu může být zaslána až na osm telefonních čísel najednou. V případě připojení handsfree sady a následném zavolání na číslo komunikátoru lze i telefonovat. Tento výrobek je určen pro interiérové použití. Nejvhodnější umístění je tam, kde je největší síla signálu. Před vložením SIM je doporučeno zrušit kód PIN. [15]



Obrázek 3 - Zobrazení GSM komunikátoru GD-06 *Allegro*.

Konstrukční stránka zařízení je velmi jednoduchá a při tvorbě krytu nebyla příliš řešena estetická stránka zařízení. Bílá barva byla zvolena jako univerzální, díky ní také lze na stěně komunikátor snadno přehlédnout. Uprostřed zařízení se nachází indikační LED signalizující aktuální stav. Připojená anténa má ohebný kloub pro lepší manipulaci. V místě se slabším signálem ho můžeme nastavit do polohy, ve které je síla signálu nejvyšší.



Obrázek 4- vzhled komunikátoru *Allegro*

Komunikátor GD-06 Allegro je určen pro vnitřní použití. Instalaci mimo budovu nebo zastřešený objekt vystavujeme komunikátor zvýšenému nebezpečí poškození. Zařízení není vodotěsné, proto nesmí být použito v prostředí se zvýšenou koncentrací vlhkosti.

4.3. Druhy komunikací s GSM komunikátorem

SMS

Na telefonní číslo je poslána běžná textová SMS s definovanými příkazy.

Příklad: STAV RELE TEMP x ZAPNI Px VYPNI Px.

STAV - Příkaz pro zpětné posláni SMS s aktuálním stavem na komunikátoru.

RELE TEMP X – Příkaz pro zapnutí automatické regulace topení na X teplotu.

ZAPNI/VYPNI Px – Zapnutí nebo vypnutí svorky Px. Když je na svorkách připojené zařízení (bojler, osvětlení), tak je zapnuto nebo vypnuto.

SMS+

Na telefonní číslo bude poslána běžná textová SMS a následně prozvoněno.

SMSSD

Na telefonní číslo bude poslána speciální datová SMS v technickém formátu.

GPRS

Pro komunikaci pomocí GPRS je nutno nejdříve zaregistrovat zařízení na adrese www.gsmlink.cz. Také musí být tato služba aktivovaná na kartě SIM. Při tomto spojení lze zjistit aktuální stav na zařízení, ovládat svorky, nastavit všechny parametry SMS, nastavit doplňkové parametry a texty.

CALL

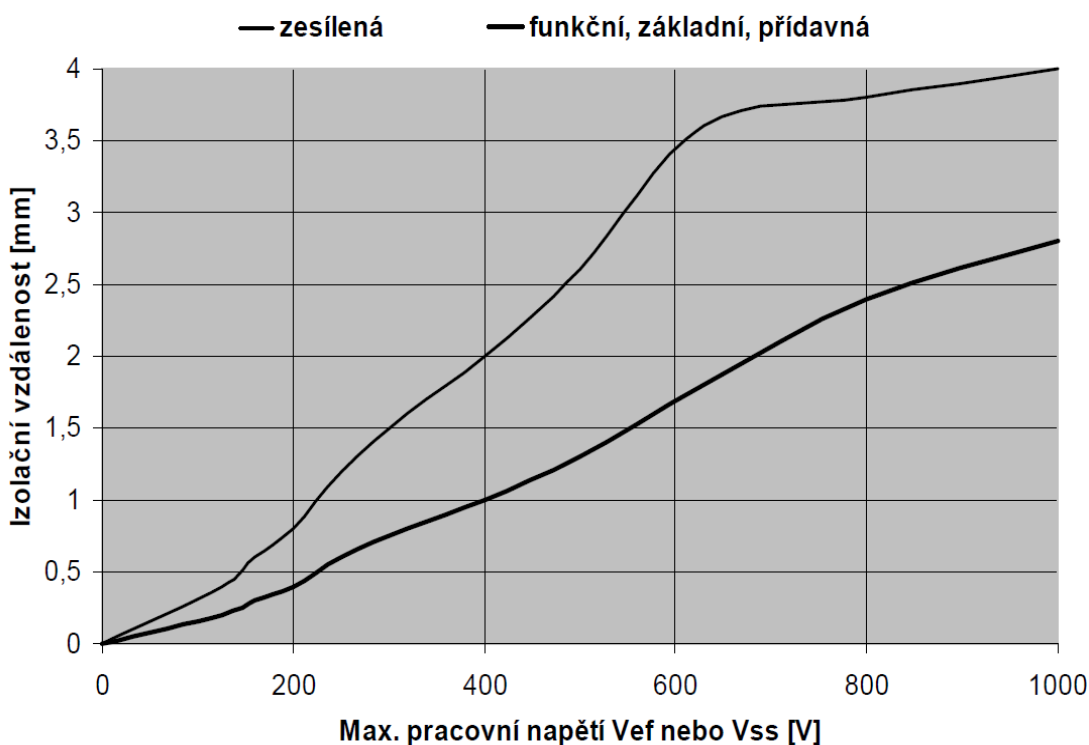
Funkce CALL umožňuje ovládání zařízení pomocí prozvonění. Reakcí na prozvonění může být sepnutí výstupu na určitou dobu nebo zjištění aktuálního stavu a zpětné zaslání SMS. Podle nastavení může každá svorka reagovat jinak.

4.4. Technické parametry komunikátoru

– napájení:	8 ÷ 15 V DC
– odběr klidový:	20 mA
– odběr při komunikaci:	500 mA
– pracovní pásmo GSM modulu:	E-GSM/GPRS 900/1800MHz
– výstupní výkon vysílače:	2 W pro GSM900, 1 W pro GSM1800
– svorky:	P1 až P6
– režim digitální vstup:	aktivace propojením s GND
– režim výstup:	zatížitelnost 100 mA spíná na GND
– režim analogový vstup:	rozsah měření 0 ~ napájecí napětí,
– pull-up rezistor:	10kΩ na napájecí napětí
– svorky NO,C,NC výstup relé:	přepínací kontakt max. 2A / 24V
– splňuje: podmínky provozování:	generální licence ČTÚ č. 1/R/2000
– bezpečnost:	podle normy ČSN EN 60950
– určeno pro prostředí:	II. vnitřní všeobecné (-10°C až +40°C)
– rozměry (bez antény):	76 x 110 x 33

Norma ČSN EN 60950 – Bezpečnost zařízení informační techniky

Tato norma, podle které je navrženo GSM zařízení Allegro, určuje způsob konstruování plošného spoje. V závislosti na velikosti napětí musí být rozmezí mezi spoji adekvátní. Norma byla vytvořena v roce 2001. V roce 2006 byla vytvořena nová norma s označením IEC 60950-1, která starou normu nahrazuje. Norma se vztahuje na zařízení informační technologie napájené ze sítě nebo z baterií. Dále také pokrývá bezpečnost kancelářských elektrických zařízení a přidružených zařízení se jmenovitým napětím nepřesahujícím 600 V. Nevztahuje se však k požadavkům parametrů a charakteristikám zařízení. Vydání nové normy nahrazující EN 60950 se liší především zvýšením požadavků baterie v zařízeních a omezení výstupního výkonu u datových portů pro přídatná zařízení.[16]



Obrázek 5- Závislost izolační mezery na velikosti napětí[17]

Na grafu vidíme dvě charakteristiky, které určují závislost izolační vzdálenosti na pracovním napětí. Čím silnější cestu na desce plošného spoje použijeme, tím menší můžeme použít izolační mezery. V ideálním řešení při dostatečně velkém spoji by izolační mezera mohla být minimální.

5. Aplikace pro mobilní zařízení realizující monitoring vzdáleného zařízení

Aplikace je vytvořena v prostředí vyvinutém společností Microsoft v produktu Visual Studio na platformě .NET Compact Framework. Komunikuje pomocí textových SMS. V aplikaci byly podle zadání použity vytvořené komponenty od pana Čajky. [18]

Řešení komunikace v aplikaci pomocí SMS se zdálo být nejjednodušším způsobem a pro toto zadání dostačujícím. Cíl vytvoření ovládní vzdáleného zařízení pomocí SMS přímo z navržené aplikace usnadňuje proces komunikace a odstraňuje problém při komunikaci přímým psaním textových zpráv. Aplikace byla řešena zpočátku jen pomocí základní nabídky Toolboxu v programu Visual Studio C#. Postupem vývoje aplikace byla změněna na ovládní pomocí komponent. Využití komponent v aplikaci zvyšuje celkový estetický dojem, usnadňuje a urychluje práci s aplikací. Jedním z nabízejících se řešení ovládní je jeho využití přímo z počítače. Problém nastává v okamžiku, kdy je potřeba řízení GSM komunikátoru a nelze použít PC. Ovládní přímo z PC je nepraktické i z těch důvodů, že musí být připojeno zařízení, jež podporuje posílání dat do GSM sítě nebo posílání přes internet [19].

5.1. Tvorba komponent

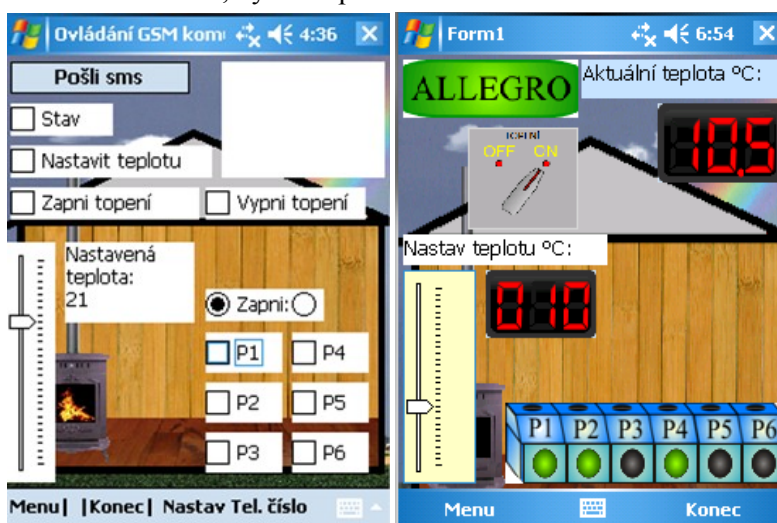
V zásadách programování jsou jistá pravidla, která je dobré dodržovat. Když vytvoříme novou komponentu (samostatnou kódovou strukturu), tak docílíme zrychlení aplikace. Kód v komponentě je volán jen tehdy, je-li potřeba. V případě, kdy komponenta není volána nebo používána, není v činnosti a zbytečně nezatěžuje procesor. U mobilních zařízení jako jsou WM, PDA a jiné produkty, využívající operační systém, je důležité řešení pomocí komponent. Čím méně aplikace zatěžuje mobilní zařízení, tím více šetří baterii. Vyvíjené komponenty mohou být vytvářeny jako grafický prvek nebo také jako dynamická knihovna, která má ve své struktuře napsaný kód na určitou funkci. Každé vytvořené komponentě můžeme nadefinovat také vlastnosti a její popis. Grafické zpracování komponenty lze provést více způsoby. Jednou z možností řešení je použití základních grafických a funkčních prvků, které poskytuje Visual studio. Lepším způsobem je však nahrání graficky pěkně zpracovaných obrázků a pomocí kódu pak nadefinujeme funkčnost komponenty [20].

5.2. Řešení aplikace pro ovládní

Podle základů programování mobilních aplikací [21] byla vytvořena aplikace, která komunikuje s GSM komunikátorem pomocí SMS. Aplikace sama o sobě je universální a pomocí proměnných komunikuje s vytvořenou komponentou, jež slouží jako zástupce GSM komunikátoru GD-06 Allegro. Toto řešení v aplikaci bylo vytvořeno pro další vylepšení funkcí a přidávání dalších komponent napsaných pro jakékoliv druhy GSM komunikátorů i s jinými možnostmi komunikace. Grafické zpracování aplikace je provedeno tak, aby člověk, který aplikaci nikdy neviděl, ihned pochopil, jak se tato aplikace ovládá.

5.2.1. Architektura aplikace

Při spuštění aplikace se zobrazí hlavní plocha, kde se nachází nejvíce ovládacích prvků. Zároveň lze také vidět poslední známý stav na vzdáleném zařízení. Pro zjištění stavu označíme položku stav a stiskneme zelené tlačítko ALLEGRO v levé horní části. Během krátké chvíle se přímo na displeji PDA změní hodnoty a zobrazení jednotlivých částí. V horní pravé části se na displeji ukáže hodnota aktuální teploty v místě nainstalování vzdáleného zařízení. Pod tlačítkem ALLEGRO je umístěn přepínač na změnu mezi manuálním a automatickým druhem řízení topení. V levé spodní části obrazovky nalezneme modrou svorkovnici, která slouží jako demonstrace svorkovnice ve vzdáleném zařízení. Každá svorka je označena jako P1 - 6. Při tvorbě bylo počítáno s tím, že více svorek v ovládaném zařízení nebude. V případě ovládání jiného zařízení, které má více svorek, by tato aplikace nedostačovala.



Obrazek 6 – Původní a současný Vzhled Aplikace

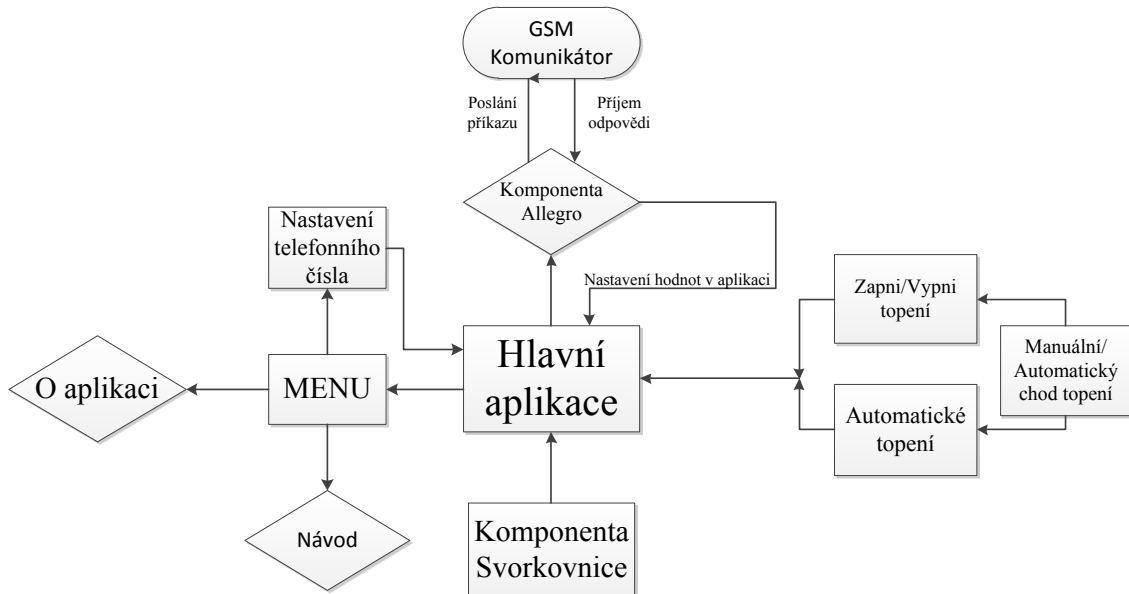
Provázání posuvníku s potřebnými komponentami:

```
private void trackBar1_ValueChanged(object sender, EventArgs e)
{
    ledDisplej2.Value = (float)trackBar1.Value + 0.5f;
    allegroGD061.RELE = true;
    ledDisplej1.Value = (float)trackBar1.Value;
    allegroGD061.n_teplota = Convert.ToInt16(ledDisplej1.Value);
}
```

Zajištění funkčnosti čtení SMS v aplikaci:

```
private void Form1_Load(object sender, EventArgs e)
{
    ifon();
    allegroGD061.cteniSMS();
}
private void Form1_Closed(object sender, EventArgs e)
{
    allegroGD061.cteniSMS_Closed();
}
```

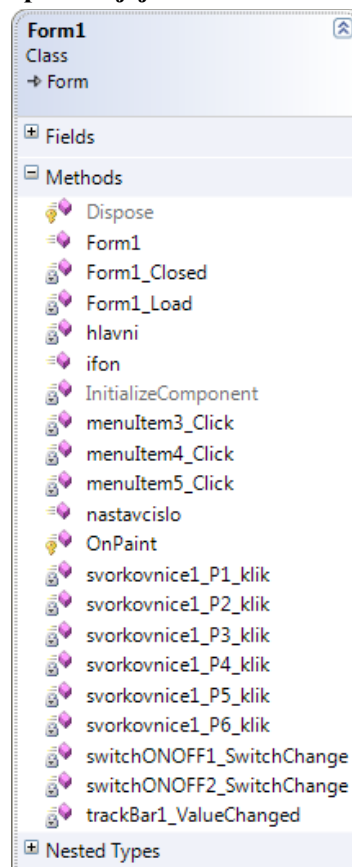
Blokové schéma řešení aplikace:



Obrázek 7 – Blokové schéma aplikace

Pomocí hlavního Formu (základní aplikace) jsou spojeny všechny komponenty jednoduchým kódem. Pro každé provázání existuje různý kód podle typu funkce komponent.

Diagram třídy hlavní aplikace a použití jejich metod



Obrázek 8-Výpis použitých metod

5.2.2. Implementace řešení

Nastavování telefonního čísla

V nabídce menu na tlačítkové liště jsou nabídky O programu a Nastav Tel. čísla. Při kliknutí na položku Nastavení telefonního čísla se otevře nové okno, ve kterém můžeme jak přidávat nové kontakty, tak i nastavit jedno ze seznamu. Na obrázku 4 jsou telefonní čísla pouze ilustrační a ve skutečnosti to telefonní čísla nejsou. Po přidání jakýchkoli telefonních čísel musíme stisknout Uložit a čísla ve světlezeleném poli se uloží do textového dokumentu. Při ukládání se všechna čísla kódují tak, že při otevření samotného textového dokumentu není zřejmé, co čísla přesně znázorňují. Kdybychom to i přesto dokázali identifikovat, tak nelze přijít na to, jak z hodnot dostaneme telefonní čísla. Když označíme jedno z čísel a klikneme, na položku Nastav, tak se číslo nastaví do aplikace a na toto číslo budou posílány veškeré provedené příkazy.

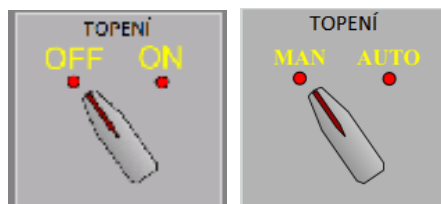


Obrázek 9 – Přidávání telefonního čísla

Uložení Tel. čísla do txt souboru:

```
Stream l2Stream = null;
l2Stream = new FileStream(@"Program Files\Chata3\Tel_cislo.txt",
    FileMode.Create);
l2Stream.Close();
for (int p = 0; p < listBox1.Items.Count; p++)
{
    Stream l11Stream = null;
    l11Stream = new FileStream(@"Program Files\Chata3\
        Tel_cislo.txt", FileMode.Append);
    StreamWriter lWriter = new StreamWriter(l11Stream);
    lWriter.WriteLine(listBox1.Items[p]);
    lWriter.Close();
}
```

Funkce ON/OFF, MAN/AUTO



Obrázek 10 - Přepínač mezi manuálním a automatickým chodem.

Funkce MAN/AUTO má za úkol přepínání mezi manuálním a automatickým chodem ovládání topení. Je-li přepnuta na manuální řízení, na displeji se zobrazí přepínač ON/OFF. Tímto přepínačem uživatel nastavuje, zda má topení topit nebo ne. V originálním nastavení po poslání příkazu „Zapni RELE“ topení topí maximálním výkonem bez omezení. Vypne se jen tehdy, jakmile nastane přehřátí, nebo když přijde nový příkaz. Pokud je nastaveno na automatické řízení, tak přepínač ON/OFF zmizí a objeví se trackbar (posuvník) a displej, na němž se nám zobrazuje hodnota při změně polohy na posuvníku. Hodnota na displeji se uloží do globální proměnné a tu potom jiná komponenta uloží do příkazu, který je poslán do vzdáleného zařízení. V následující části kódu je vidět, co se stane, když přepínáme mezi manuálním a automatickým chodem. V obou případech se nastavuje viditelnost jednotlivých prvků, ale v základu (když je nastaven manuální chod) je ošetřeno to, aby se nenastavoval příkaz pro automatickou regulaci.

```
if (onoff == true)
{
    trackBar1.Visible = true;
    ledDisplej1.Visible = true;
    switchONOFF2.Visible = false;
}
else
{
    allegroGD061.RELE = false;
    trackBar1.Visible = false;
    ledDisplej1.Visible = false;
    switchONOFF2.Visible = true;
}
```

Nastavení hodnoty teploty pro regulaci



Obrázek 11 - Nastavování hodnoty teploty

Při změně hodnoty na posuvníku, jenž má rozmezí od 0 do 30, displej zobrazí aktuální hodnotu. Hodnota na displeji je použita při řízení pomocí automatického chodu komunikátoru teploty v místnosti. Příkaz pro zapnutí regulace teploty u použitého GSM komunikátoru je RELE TEMP X (X značí hodnotu nastavené teploty).

5.2.3. Vlastní realizované komponenty pro vzdálené řízení

Komponenta Allegro

Komponenta Allegro slouží čistě ke komunikaci se vzdáleným zařízením, které bylo dodáno k testování. Vytvořením této komponenty bylo dosaženo větší kompatibility aplikace a zároveň zjednodušení a zkrácení hlavního kódu. Pomocí globálních proměnných komunikuje aplikace s komponentou ALLEGRO. Vytvořením nové komponenty lze komunikovat s dalšími zařízeními a pomocí jiných periférií, než jen pomocí SMS.



Obrázek 12 – Komponenta ALLEGRO

Komponenta ALLEGRO má stejnou funkci jako tlačítko. Při stisku na zelené pole s černým nápisem je proveden příkaz, jenž odešle SMS na předem definované číslo s příkazy, které uživatel nastavil v aplikaci. Když přijde z komunikátoru SMS, tato komponenta ji přijme, rozdělí text na jednotlivá slova, nastaví globální proměnné k použití a je-li hlavní aplikace propojena s proměnnými, nastaví vizualizaci podle aktuálního stavu na GSM komunikátoru. Například je schopná změnit svorky na zapnuté nebo vypnuté, přepne mezi manuálním a automatickým ovládáním topení atd. Když není v mobilním zařízení, ve kterém se nachází ovládací aplikace, vložena karta SIM a je stisknuto tlačítko ALLEGRO, aplikace zobrazí chybovou zprávu na displeji.

Rozdělení slov v příchozí SMS tak, aby fungovaly jen povolené příkazy:

```
public bool P1 = false;
public bool P2 = false;
public bool P3 = false;
.
.
.
string pattern = "RELE Zapnuto|P1 Zapnuto|...";
string data = textBox1.Text;

MatchCollection mc = Regex.Matches(data, pattern);

foreach (Match cp in mc)
{
    switch (Convert.ToString(cp.Value))
    {
        case "P1 Zapnuto": PT1 = true; break;
        case "P2 Zapnuto": PT2 = true; break;

        case "2,0C": akt_tep = 2.0; break;
        case "2,1C": akt_tep = 2.1; break;
        case "2,2C": akt_tep = 2.2; break;
    }
}
```

Komponenta Svorkovnice



Obrázek 13 – Komponenta Svorkovnice

Toto provedení svorkovnice nahrazuje původní řešení pomocí checkboxů. Její funkčnost je velmi jednoduchá. Pro zapnutí nebo vypnutí svorek stačí stisknout šedé kolečko. Když zezelená, znamená to, že svorka je zapnuta. Provedeme-li ten samý úkon ještě jednou, tlačítko zšedne a tudíž i vypne svorku. Pomocí globálních proměnných komunikuje s jinými komponentami nebo s hlavní aplikací. Záleží, ve kterém místě tyto proměnné voláme. Při programování komponenty Svorkovnice byly vytvořeny „Eventy“, události, se kterými můžeme pracovat. Tyto události slouží jako jednotlivá kliknutí na puntíky. Definování, aby „KLIK“ fungoval na jednotlivých svorkách, uskutečňujeme pomocí Handlu a tříd. Pozadí této komponenty je voláno pomocí kódu, ale jednotlivá kolečka jsou vloženy obrázkové boxy, které při kliknutí mění obrázek.

Deklarace vytvoření nového handlu tak, aby fungoval klik přímo na kulaté tlačítko:

```
//definice handlu aby jsme měli nový event
public delegate void P1_clickHandler(object sender, P1_click e);

        P1_click klik1 = new P1_click();           //nastavení aby
při kliku na obrázek se provedla funkce klik
        if (P1_klik != null)
            P1_klik(this, klik1);
public class P1_click : EventArgs           //vytvoření třídy pro event
aby byl funkční
{
}
```

Po stisku na kolečko je volána bool hodnota, která má stavy true a false. Návazností na tuto hodnotu se nastaví barva tlačítka. Je-li hodnota TRUE, změní se na zelenou, když je FALSE, tak zšedne.

```
PT1 == true
pictureBox2.Image = new Bitmap(Assembly.GetExecutingAssembly().
GetManifestResourceStream("svorkovnice.Resources.puntik.png"));

PT1 == false
pictureBox2.Image = new Bitmap(Assembly.GetExecutingAssembly().
GetManifestResourceStream("svorkovnice.Resources.puntikCerny.png"));
```

5.3. Komunikace mezi aplikací a GSM komunikátorem

Odesílání

Komunikace je založena na vytvořené samostatné komponentě, která v aplikaci slouží jako distributor mezi aplikací a GSM komunikátorem GD-06 Allego, vyhodnocuje nastavení vizuálních prvků přímo v aplikaci a po stisku tlačítka odešle příkaz na řízení pomocí textové zprávy na určené telefonní číslo. Je-li kód správně poslán a nadefinován, komunikátor nastaví hodnoty dle požadavku a pošle zpět odpověď s aktuálním stavem. Kód pro čtení SMS je možno napsat více způsoby. Nejpoužívanějším je vytvoření, definice telefonního čísla, konstruování těla SMS a poslání.

```
SmsMessage smsMessage = new SmsMessage();
smsMessage.Body = "Zde je text, který se má poslat.";
smsMessage.To.Add(new Recipient("Jméno kontaktu", "Telefonní číslo"));
smsMessage.RequestDeliveryReport = true;
smsMessage.Send(); //Metoda na poslání SMS.
return;
```

Příklad definice na vytváření SMS je zbytečně moc dlouhý a cílem vytvoření aplikace má být naprogramování aplikace tak, aby byla co nejvíce energeticky úsporná [22], co nejméně náročná. Každý řádek kódu navíc, který nemusí být použit, pouze rozkládá jednu funkci na dvě menší a zatěžuje telefon, ve kterém se aplikace nachází. I když může být zatížení minimální a nepoznatelné, v konečném výsledku je to velký rozdíl. Úpravou doporučeného kódu na SMS bylo dosaženo tří řádků, kde v prvním byla vytvořena SMS a zároveň i definice telefonního čísla komunikátoru a text SMS, který má být poslán. V druhém řádku je vytvořeno tělo z proměnných, které mají být použity. V posledním řádku je volána metoda pro posílání.[4]

```
SmsMessage s = new SmsMessage("Telefonní číslo","Text posílené sms");//vytvori sms
s.Body = this.label_sms2.Text; //Tvorba tela sms
s.Send(); //poslani sms
```

Příjem

Po příchodu SMS komponenta vyhodnotí, zda je od GSM komunikátoru, přečte ji a podle definovaného kódu, který má ve své struktuře, nastaví správně proměnné. Jsou-li proměnné nastavené a v hlavní aplikaci provázané s vizuálními prvky, tak se v okamžiku příchozích hodnot aplikace nastaví do takových poloh, aby uživateli bylo hned zřejmé, jaký je stav na zařízení. Když nastavujeme jakékoliv hodnoty, tak se vzhled příliš nezmění. Avšak v případě, kdy je zařízení nastaveno a v aplikaci je vše vypnuto nebo zapnuto a pošleme SMS jen s příkazem „STAV“, tak se aplikace ztelně změní. Po každé příchozí SMS se s největší pravděpodobností bude měnit hodnota na displeji pro zobrazení aktuální hodnoty teploty na vzdáleném zařízení. I v případě, že by byly poslány dva stejné příkazy pár sekund po sobě, tak vzhledem k neustále měnícímu se prostředí bude hodnota v řádech desetín, po každém vrácení hodnot, rozdílná. Přesné hodnoty nejsou zapotřebí, protože v této aplikaci stačí pouze informativní hodnoty a rozdíl jednoho °C nehraje žádnou roli.

6. Aplikace pro mobilní zařízení nahrazující GSM komunikátor

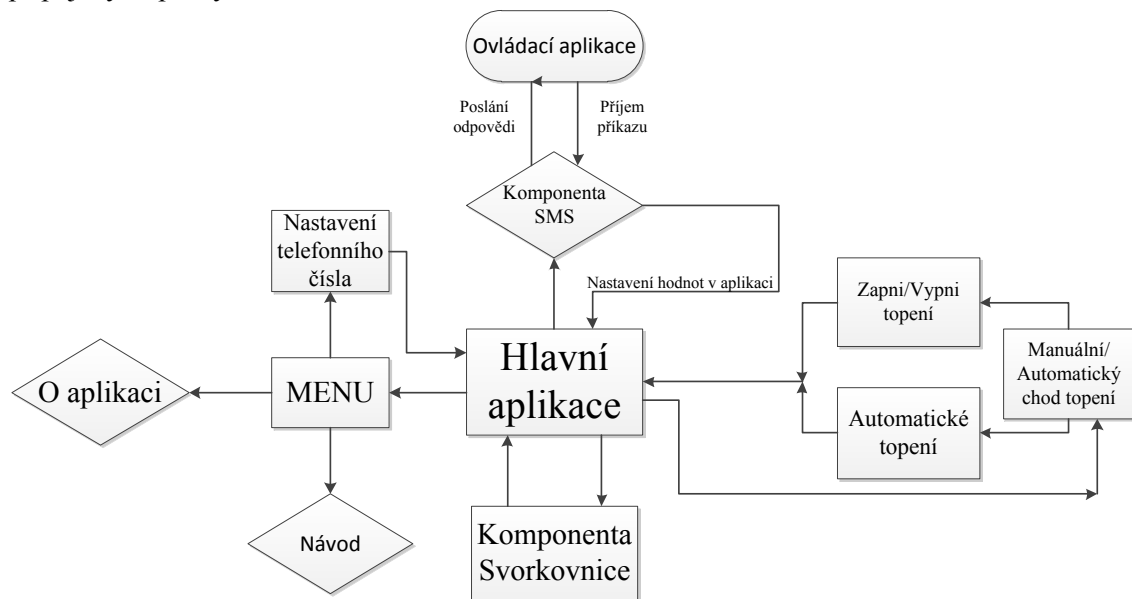
Šestá kapitola je druhou částí práce, která řeší aplikaci nahrazující GSM komunikátor. První částí je rozbor způsobu řešení aplikace. V dalších částech je popsán rozbor tvoření aplikace, grafických prvků a nakonec i nově vytvořená komponenta.

6.1. Definice problému

Aplikace je vytvořena ve stejném prostředí jako ovládací aplikace. Vytváření bylo velmi podobné. Rovněž je tomu tak i u vizuálního vzhledu. Vložené grafické prvky jsou použity z předchozí aplikace a jejich funkčnost je stejná.

V případě, kdy nemáme jakýkoliv GSM komunikátor, nabízí se možnost využití mobilního telefonu nebo PDA s operačním systémem WM. Aplikace nahrazující GSM komunikátor má tu výhodu, že nejsme omezeni počtem výstupů a je možno naprogramovat nezměrné množství funkcí. Jsme omezeni pouze možnostmi mobilního zařízení. Z hlediska pořizovacích nákladů je pořízení mobilního telefonu a této aplikace levnější než nákup GSM komunikátoru.

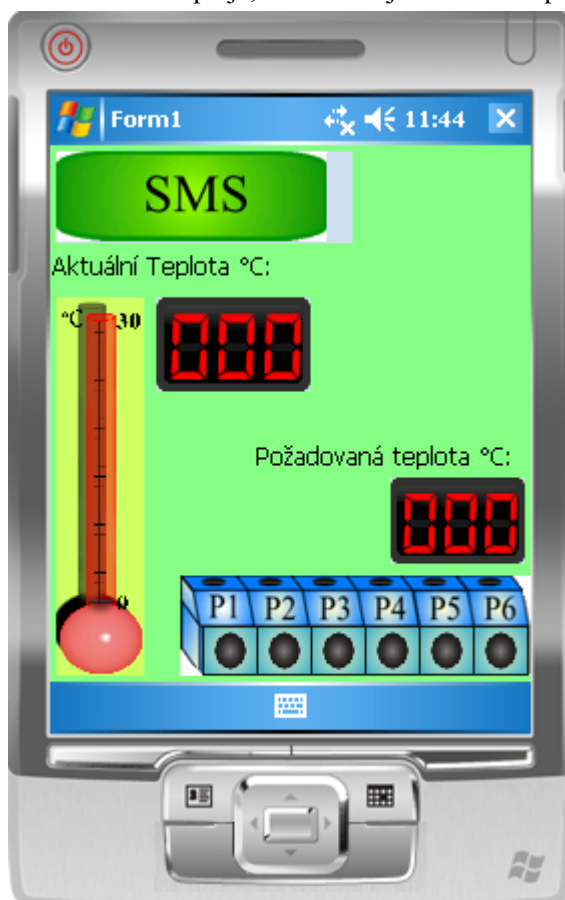
Nevýhodou využití tohoto způsobu ovládání a řízení je to, že GSM komunikátor pracuje stabilněji než WM, nezasekává se a jeho spolehlivost je vysoká. U mobilních zařízení vznikají problémy se zamrzáním. Občas může zkolabovat spuštěná aplikace nebo komunikace mezi připojenými prvky.



Obrázek 14- Blokové schéma ovládané aplikace

6.2. Řešení ovládané aplikace

Tato aplikace je zkonstruována podobně jako aplikace pro ovládání vzdálených zařízení. Její funkčnost nahrazuje GSM komunikátor Allegro GD-06, protože funkční příkazy, na které reaguje, jsou napsané pro toto zařízení. Stavební prvky (komponenty) jsou použity stejné jako v ovládací aplikaci. Komponenta Allegro, jež slouží ke čtení příchozích SMS a odesílání povelů, je předělána a pojmenována jako SMS, viz další popis. Když přijde povel od telefonního čísla, které je nastavené, aplikace vyhodnotí příkazy a provede veškeré úkony. Nastaví vizualizaci na požadované hodnoty a pošle zpět SMS s hodnotami v telefonu. Přidaná komponenta Teploměr zobrazuje aktuální hodnotu teploty v místě uložení PDA. V tomto řešení aplikace generuje teplotu pouze pomocí algoritmu a zobrazuje se na displeji telefonu pomocí komponenty Teploměr a přidaného Led displeje, na kterém je zobrazena přesná hodnota teploty.



Obrázek 15- Aplikace nahrazující GSM komunikátor

Svorkovnice

Využívá stejné svorkovnice jako v předchozí aplikaci. Její rozdíl ve funkčnosti je ten, že reaguje přesně opačně. Přijde-li příkaz, tak nastaví všechny svorky požadovaným způsobem a vrátí hodnoty nastavených svorek. Zapínání svorek je pouze virtuální a reálného spínání nebylo dosaženo. Po příchodu SMS se zadanými příkazy jsou sepnuty ty svorky, které se mají sepnout podle příchozího příkazu. Následně vrátí hodnoty, které svorky jsou sepnuty a které vypnuty. U reálného řešení by musela být ošetřena zpětná vazba od spínaného zařízení.

6.3. Vlastní realizované komponenty pro vzdálené řízení

Komponenta SMS

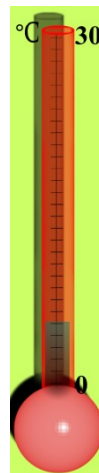
Tato komponenta je velmi podobná komponentě ALLEGRO. Struktura kódu je zkopírována a upravena na takové hodnoty, aby byly kompatibilní a funkční. Komponenta SMS funguje také jako tlačítko, na které když klikneme, tak odešle SMS na definované číslo, ale zároveň funguje automaticky. Po nastavení hodnot a podle příchozí SMS komponenta sama vyhodnotí stav, uloží všechny údaje do SMS a odešle je zpět na číslo, z něhož přišel příkaz. V případě chyby při odesílání, SMS aplikace neodešle a pokračuje dál ve své funkčnosti.



Obrázek 16 – Komponenta SMS

Komponenta Teploměr

Komponenta Teploměr slouží k demonstraci změřené aktuální hodnoty teploty v okolí. Po přijmutí příkazu na zapnutí nebo vypnutí topení nebo na automatickou regulaci teploty komponenta provede příslušný příkaz. Bude-li komponenta signalizovat například 3°C a v přijaté SMS se zobrazí příkaz „ZAPNI RELE (zapni topení)“, spustí se časovač, pomocí něhož se bude pomalu přidávat hodnota teploty na teploměru. V reálném použití sepne topení a bude měřit danou hodnotu. Ve funkčnosti komponenty bude jen signalizovat hodnotu teploty. Průběh změny zobrazení hodnot na teploměru není plynulý. Funkce změna vzhladu a změna výšky hladiny je tvořena překrýváním obrázků. Obrázky jsou definovány pro určité hodnoty, tudíž se obrázek mění přibližně a neurčuje přesnou hodnotu teploty. Teploměr má rozsah od 0 do 30°C. Při vyšší teplotě nereaguje na změny. Pro přesnou hodnotu zobrazení teploty je dobré mít k této komponentě připojen displej nebo textový box, ve kterém je hodnota zobrazena.



Obrázek 17- Komponenta Teploměr

```
new Bitmap(Assembly.GetExecutingAssembly().  
    GetManifestResourceStream("Teplomer.imagy.Teplomer.png"));  
new Bitmap(Assembly.GetExecutingAssembly().  
    GetManifestResourceStream("Teplomer.imagy.11C.png"));
```


7. Testování vyvinutého řešení dle navržené metodiky

Testování aplikací bylo provedeno pomocí navržené metodiky, která je popsána v první části této kapitoly. V druhé části je řešeno testování HMI a třetí část řeší testování vlastní struktury aplikace ze strany SW části.

7.1. Navržení metodiky

Navržená metodika pro testování aplikace byla vyhotovena formou jednoduchých dotazníků a pomocí diskuze. U obou aplikací byly použity stejné otázky. Vyhodnocení je sumarizováno do jedné odpovědi u každé otázky, při rozdílných odpovědích byly uvedeny všechny možnosti. Případné nevhodně formulované odpovědi byly upraveny nebo nebyly vůbec uvedeny. Celkový počet respondentů, kteří testovali obě aplikace, bylo 20.

V hodnocení grafického prostředí Black Box byly použity dotazy:

- Je aplikace přehledná?
- Je aplikace lehce ovladatelná?
- Je aplikace stabilní?
- Projevují se nějaké chyby při chodu?
- Kolabuje nebo se zasekává?
- Je návod k aplikaci pochopitelný?
- Jaké jsou případné návrhy na změnu?

V hodnocení softwarové části White Box byly použity dotazy:

- Je zdrojový kód přehledný v celé aplikaci a komponentách?
- Je řešení kódu optimální?
- Je SW dostatečně rozšiřitelný?
- Je dobře napsaná z hlediska struktury psaní?
- Nachází se v kódu na první pohled zbytečné pasáže?
- Jaké jsou další návrhy na změnu?

7.2. Black Box

Primárně se jedná o testování HMI, tedy grafického a funkcionálního zpracování aplikací.

Aplikace pro mobilní zařízení realizující monitoring vzdáleného zařízení

Otázka č.1- Je aplikace přehledná?

- ANO
- pozadí mírně ruší
- komponenty na displeji jsou moc u sebe a aplikace vypadá přeplněná

Otázka č.2- Je aplikace lehce ovladatelná?

- ANO
- Posuvník na nastavení teploty se hůře ovládá.

Otázka č.3- Je aplikace stabilní?

- ANO
- V simulátoru funguje bezchybně, ve WM se stala občas chyba na straně zařízení

Otázka č.4- Projevují se nějaké chyby při chodu?

- NE, žádné

Otázka č.5- Kolabuje nebo se zasekává?

- NE

Otázka č.6- Je návod k aplikaci pochopitelný?

- ANO
- Ano, lze ho pochopit, ale nejdříve by chtělo vysvětlit, k čemu aplikace slouží, i když to po krátké chvíli zjistíme

Otázka č.7- Jaké jsou další návrhy na změnu?

- Aplikaci vytvořit tak, aby nepřesahovala display
- Na zjištění stavu použít hezčí tlačítko
- U nastavování automatického topení použít menší display
- Na emulátoru byla aplikace zobrazena trochu větší. Bylo by dobré vytvořit elastické prostředí, nezávislé na rozlišení.

Aplikace pro mobilní zařízení nahrazující GSM komunikátor

Otázka č.1- Je aplikace přehledná?

- ANO
- Komponenta teploměr by mohla mít výraznější příčky

Otázka č.2- Je aplikace lehce ovladatelná?

- ANO
- ANO, ale v podstatě není potřeba ovládat manuálně

Otázka č.3- Je aplikace stabilní?

- ANO
- V simulátoru funguje bez problémů, nevím, jak v reálném využití, neměl jsem to, jak otestovat.

Otázka č.4- Projevují se nějaké chyby při chodu?

- NE
- Na začátku spouštění aplikaci chvíli trvá, než se rozjede

Otázka č.5- Kolabuje nebo se zasekává?

- NE, zdá se být v pořádku

Otázka č.6- Je návod k aplikaci pochopitelný?

- ANO
- Ano, ale i bez návodu jde snadno pochopit.
- Návodem je spíše upřesněno, jaké příkazy se mají posílat do aplikace

Otázka č.7- Jaké jsou další návrhy na změnu?

- LED displej na zobrazení aktuální teploty udělat větší než na zobrazení požadované teploty
- Komponenta teploměr by mohla být menší
- Sladit všechny prvky do podobných barev, ale to je věc názoru.

7.3. White Box

Metoda testování pomocí White Boxu se zabývá vlastní strukturou kódu v aplikaci. Hodnotí, jakým zpracováním je proveden algoritmus, hledá chyby v kódu a neošetřené části.

Aplikace pro mobilní zařízení realizující monitoring vzdáleného zařízení

Otázka č.1- Je zdrojový kód přehledný v celé aplikaci a komponentech?

- ANO
- V některých částech chybí popisky

Otázka č.2- Je řešení kódu optimální?

- ANO
- Ano, proč ne, ale pro použití ovládání složitějšího zařízení, kdy nestačí jen velikost SMS, by bylo lepší řešit přes GPRS komunikaci.
- Řekl bych, že je v pořádku. Některé věci bych udělal jinak, ale to už záleží na úhlu pohledu

Otázka č.3- Je dostatečně rozšiřitelné?

- ANO
- Ano, ale ještě by to šlo zlepšit pomocí interface

Otázka č.4- Je dobře napsaná z hlediska struktury psaní?

- ANO
- Jen v komponentě ALLEGRO lépe řešit výčet hodnot

Otázka č.5- Nachází se v kódu na první pohled zbytečné pasáže?

- Na první pohled se zdá aplikace jako velmi dobře řešená
- Není nic závažného poznat. Jestli jsou, tak nepatrné.

Otázka č.6- Jaké jsou další návrhy na změnu?

- Když jsou používány regulérní názvy, tak ať jsou „v plné kráse“ (soubor usercontrol1.cs)
- Komponentu Svorkovnice a Allegro pojmenovat ve všech částech tak, jak mají být.

Aplikace pro mobilní zařízení nahrazující GSM komunikátor

Otázka č.1- Je zdrojový kód přehledný v celé aplikaci a komponentech?

- ANO
- ANO, i popisky jsou dobře napsané

Otázka č.2- Je řešení kódu optimální?

- ANO
- Chtělo by to drobné úpravy, ale na funkčnosti by to nic nezměnilo

Otázka č.3- Je dostatečně rozšiřitelné?

- ANO
- Zdá se, že ano, ale pro použití rozšíření stávajících funkcí by bylo zapotřebí předělat komponenty.

Otázka č.4- Je dobře napsaná z hlediska struktury psaní?

- ANO
- Jen v komponentě Teploměr je řešení kódu hůře provedeno.

Otázka č.5- Jsou na první pohled v kódu zbytečné pasáže?

- Asi NE, zběžně to je dobré.
- Není poznat, jestli jsou tak nepatrné.

Otázka č.6- Jaké jsou další návrhy na změnu?

- Kód v komponentě Teploměr by bylo lepší zkrátit a nahradit lepšími funkcemi.

Odpovědi na všechny otázky byly použity tak, jak byly zodpovězeny v dotaznících od jednotlivých uživatelů, kterých bylo celkem 20. Stejně nebo podobné odpovědi nebyly znovu uváděny.

8. Diskuze výsledků

Cílem práce byl teoretický rozbor možností použití komunikace mezi PDA a vzdáleným zařízením a praktická realizace vytvoření aplikace pomocí platformy .NET Compact Framework.

V úvodu je popsána základní myšlenka, o čem je tato bakalářská práce. Teoretický rozbor je realizován nejdříve základním pojednáním o tom, co je to platforma .NET Framework a .NET Compact Framework.

Popisem v teoretické části je čtenáři vysvětleno, jak fungují uvedené typy komunikací. Nastíněním částí kódu u každého z typů komunikace si můžeme udělat obraz o tom, jak náročné je docílit správné funkčnosti.

Porovnáním rozdílů mezi druhy řízení jsem zjistil, že komunikace pomocí GPRS je naprosto odlišná a komunikace přes SMS nebo MMS nejsou tak podobné, jak se zdají. Postupnou dedukcí byly díky rozborům typů spojení vyřazeny ty, které nejméně splňovali požadavky. Použití SMS vyšlo jako jediné, které není složitostí programování tolik náročné. Navíc výběr, pomocí kterého bude aplikace komunikovat, byl podmíněn tím, že stávající produkt už byl ovládán pomocí SMS a také na tomto typu komunikace bylo vysvětleno, jak zařízení funguje a jak jej ovládat.

V tvoření programované aplikace byly použity podle zadání již vytvořené komponenty. Podle prvních očekávání jsem si myslel, že spojení programu s přidávanými prvky bude problematické, ale nebylo. Vytvoření nové komponenty je daleko složitější díky nastavování různých možností komunikace. Pro pokrytí všech funkcí ovládání komunikátoru musely být vytvořeny i nové komponenty, protože nabídka prvků dodaných k práci byla nedostačující. Vytváření komponent bylo účelné a aplikace se tím zpřehlednily a také bylo docíleno větší kompatibility. Vytvořené komponenty jsou popsány v této práci a případné rozšíření jejich funkcí není problematické. Struktura kódu je jednoduše popsána tak, aby člověk, který bude s prvky dále pracovat, snadno pochopil, jakým způsobem vše funguje.

Nejdříve byla vytvořena aplikace pro ovládání GSM komunikátoru. To především proto, že je v této práci více důležitá. Logika aplikace nejdříve stála na návrhu využívající pomoc základní nabídky grafických a funkčních prvků v programu Visual Studio od společnosti Microsoft. Toto provedení bylo však nedostačující a nevzhledné. Nahrazením prvků graficky pěknými komponentami bylo dosaženo graficky příjemnějšího prostředí. Uživatelské prostředí v této aplikaci působí jak jednoduše, tak i na pohled přitažlivě. Aplikace na ovládání vzdáleného zařízení slouží spíše jako nosná a využívá k veškerým funkcím vložené komponenty a ostatní prvky. Zdrojovým kódem v aplikaci bylo dosaženo jen propojení jednotlivých komponent.

U komponenty ALLEGRO byl problém se čtením příchozích SMS. Když byl vyřešen způsob čtení určitých SMS, byl problém s roztříděním příchozího textu a jeho použitím v aplikaci. V případě příchodu pouze jednoslovných SMS by to nebyl problém. Od komunikátoru však chodí SMS, kde jsou slovní spojení a jejich rozdělení tak, aby fungovala jako celek, což se ukázalo jako nedefinovatelné. Dalším problémem bylo čtení hodnoty teploty,

kdy je příchozí text spojením čísel a písmen. Konečné řešení fungování všech příkazů tak, jak mají, bylo nalezeno po hledání problému v internetových diskuzních fórech.

Návrhem komponenty svorkovnice byl odstraněn problém původního nevhledného stavu aplikace. Po vytvoření komponenty a jejím naprogramování vznikly problémy při testování. Bylo-li kliknuto na jakoukoli svorku, byla barva kolečka změněna, ale příkaz nebyl proveden. Toto byl největší zádrhel v tvoření komponenty. Proměnná, která zajišťuje změnu hodnoty, musí být při kliknutí negována. Komponenta není sama o sobě nijak rozšiřitelná, poněvadž je řešena pro určitý počet svorek. V případě, kdy by bylo nutné ji rozšířit, musela by se vytvořit nová, ale princip stavby komponenty by mohl zůstat stejný.

Druhá aplikace, která nahrazuje funkci GSM komunikátoru, byla záměrně řešena až po vytvoření a doladění předešlé aplikace. Prvky použité v této aplikaci jsou stejné až na malé rozdíly ve funkčnosti. Některé komponenty zůstaly stejné beze změny a některé poupraveny. Provedené změny byly v kódu a to spíše jen v prohození funkčnosti nebo definice čtených příkazů. V aplikaci nahrazující GSM komunikátor byla přidána nová komponenta s názvem a funkčností Teploměru. Hlavním problémem v ovládané aplikaci bylo ošetření komunikace mezi komponentami, aby jednotlivé funkce mezi sebou nebyly v rozporu. Při čtení příchozí ovládací SMS bylo problematické nastavení hodnot na displeji. Nebylo možné, aby svorky P1 - P6 sepnuly a zároveň komponenta SMS mohla použít proměnné pro textovou zprávu, v níž posílá zpět odpověď o stavu.

U komponenty Teploměru bylo složité řešení automatického nastavování hodnot. Pro nastavení teploty při simulaci topení bylo nutné nastavit časovač na určitý interval. Po vložení však program nešel zkompileovat a nešlo otestovat ani jeho funkci. Po vyřešení problému byla komponenta doladěna tak, aby plynule simulovala chod připojeného topení.

Psaní jednotlivých částí této práce probíhalo přibližně tak, jak byly vytvářeny aplikace. Postupně byly zaznamenávány poznatky, jakými byly řešeny jednotlivé části programů a komponent. Závěrem práce je testování aplikací kolegy, kteří někdy měli zkušenosti s programováním, a to především v programovacím prostředí C# nabízeném programem Visual Studio od společnosti Microsoft. Za pomoci jednoduchých dotazníků společně s vytvořenými aplikacemi mi zodpověděli otázky a připsali navíc další možná řešení. Testování by bylo daleko přesnější a s více možnými odpověďmi, ale nebylo možné předložit dotazník dostatečnému množství kompetentních lidí.

Nakonec můžu zhodnotit, že bakalářská práce se mi podařilo vypracovat podle zadání. Komplikace, které vznikly během tvorby celé práce, byly z velké většiny odstraněny. Ty, jež se mi nepodařilo odstranit formou nahrazení jiným typem řešení, byly poupraveny tak, aby splnily správně svoji funkci.

9. Závěr

Na začátku vytváření této práce jsem teoreticky rozebíral prostředí nabízené společností Microsoft. Popis jednotlivých kapitol nám ukázal, jaké jsou možnosti programování jednotlivých druhů možných prostředí. Teoretickým rozbohem komunikace pomocí SMS, MMS a GPRS byl čtenáři ukázán základní aspekt dané problematiky týkající se komunikací. Jak již bylo zmíněno v předešlé kapitole, je možné udělat si celkový obraz o tom, jak lze programovat tyto typy komunikací. Na první pohled lze snadno usoudit, že programování komunikace pomocí SMS je nejjednodušší. Tento druh posílání a přijímání dat se zdá v této aplikaci vhodný, ale musíme zvážit, kolik a jaká data budou posílána a podle toho určíme typ komunikace.

Analýzou reálného využití zadaných typů komunikací jsem našel jednotlivá řešení v reálném použití a také princip, na němž fungují. U komunikace pomocí SMS jsem tvořením aplikace zjistil, že kód, který je doporučován společností Microsoft, je možné omezit a funkčnost se tím nezmění. Reálné využití MMS jsem nemusel v této práci řešit. V rozboru reálného využití jsem popsal jednotlivé výhody a nevýhody této komunikace, které jsem našel v dostupných literaturách. Po hlubším rozboru komunikace pomocí GPRS vyšlo najevo, že je nejpříjemnějším řešením s mnoha výhodami. Tento typ komunikace má asi nejširší možnosti využití v programování aplikací a jejich vzájemné spolupráci.

V jazyce C# jsem naprogramoval aplikaci, která slouží k ovládní již existujícího vzdáleného zařízení GD-06 ALLEGO, pomocí SMS. Aplikace byla vytvořena z několika komponent, jak z vytvořených v předešlých letech, které jsem dostal k dispozici, tak i z komponent, jež jsem musel sám vytvořit. Vyvíjení komponent bylo řešeno pomocí dostupné literatury nebo zdrojů v pdf formátu uložených na internetu. Aplikace byla zhotovena tak, aby uživateli usnadnila práci při komunikaci, zároveň byla přehledná a graficky hezky zpracovaná. Její podoba je udělaná tak, aby byla kompatibilní a kdykoli rozšířitelná přidáním jakékoli nové komponenty, která bude např. ovládat jiný typ komunikace.

Návrhem aplikace realizující vzdálené zařízení je ukázán způsob možného nahrazení GSM komunikátoru. Vytvořená aplikace nefunguje jako komunikátor, protože k ní není připojeno žádné zařízení prostřednictvím mobilního telefonu. Tímto řešením je simulátor GSM komunikátoru GD-06 Allegra. V případě potřeby reálného využití je možné poměrně jednoduchým připsáním kódu nadefinovat spojení s připojeným zařízením.

V poslední kapitole jsou pomocí kolegů, studujících stejný obor, vyhodnoceny hodnoty z jednoduchého dotazníku. Kolegové si prohlédli aplikace, podle dotazníku ověřili funkčnost a napsali připomínky a případné návrhy, jak lze aplikace upravit. Úroveň posuzování aplikací se lišila, neboť se skupina hodnotitelů dělila na ty, kteří jsou na střední až mírně pokročilé úrovni v programování, a na začátečníky v této oblasti. Méně znalí hodnotili aplikace jako velmi dobré, ale zkušenější uživatelé byli více kritičtí a psali různé návrhy a připomínky, které jsou však užitečné pro další práci na aplikacích.

Na závěr bych chtěl podotknout, že při opětovném zpracování tohoto tématu bych snad více řešil komunikaci pomocí GPRS a ukládání do SQL databází. Toto téma nám otevírá mnoho dalších dílčích otázek a problémů, kterými by bylo dobré se dále zabývat, a nové zkoumání by nám jistě přineslo řadu užitečných poznatků.

10. Literatura

- [1] < <http://ronoath.signaly.cz>>
<http://ronoath.signaly.cz/0710/co-je-to-pda> [Datum citace 5.10.2010]
- [2] <<http://cs.wikipedia.org>>
<http://cs.wikipedia.org/wiki/.NET> [Datum citace 6.10.2010]
- [3] < <http://en.wikipedia.org>>
http://en.wikipedia.org/wiki/.NET_Compact_Framework [Datum citace 6.10.2010]
- [4] <<http://msdn.microsoft.com>>
<http://msdn.microsoft.com/en-us/library/microsoft.windowsmobile.pocketoutlook.smsmessage.send.aspx> [Datum citace 6.10.2010]
- [5] <<http://msdn.microsoft.com>>
<http://msdn.microsoft.com/en-us/library/bb932385.aspx> [Datum citace 6.10.2010]
- [6] <<http://msdn.microsoft.com>>
<http://msdn.microsoft.com/en-us/library/ms837571.aspx> [Datum citace 6.10.2010]
- [7] <<http://translate.googleusercontent.com>>
http://translate.googleusercontent.com/translate_c?hl=cs&sl=en&tl=cs&u=http://www.codeproject.com/KB/windows/CompactDynDNS.aspx&rurl=translate.google.cz&twu=1&usg=ALkJrhijX8OJ67KMGvA2LFTgWRKH6wZiDw [Datum citace 6.10.2010]
- [8] <<http://www.nowsms.com>>
<http://www.nowsms.com/discus/messages/485/21018.html> [Datum citace 6.10.2010]
- [9] <<http://wiki.answers.com>>
http://wiki.answers.com/Q/How_many_pages_of_text_will_fit_into_a_gigabyte_of_storage [Datum citace 4.11.2010]
- [10] <<http://pandatron.cz>>
http://pandatron.cz/?593&univerzalni_gsm_komunikator_-_sip300 [Datum citace 10.3.2010]
- [11] <<http://www.zabezpecovacky.cz>>
<http://www.zabezpecovacky.cz/ja-60gsm-gsm-komunikator/> [Datum citace 10.3.2010]
- [12] <<http://marcomplet.cz>>
<http://marcomplet.cz/zbozi/produkt-2881/level-level-gb-060-300a.html?sid=b0fe889f5de8862d74c62f5554d8cdde> [Datum citace 10.3.2010]
- [13] <<http://www.selax.cz>>
<http://www.selax.cz/product/t3411-novy-gsm-komunikator-16-vstupu-2-rele-s-prepinacimi-kontakty-16-podsystemu-703/> [Datum citace 10.3.2010]
- [14] <<http://www.zefin.cz>>
<http://www.zefin.cz/files/gd%2004.pdf> [Datum citace 10.3.2010]
- [15] <<http://www.zefin.cz>>
<http://www.zefin.cz/files/gd06.pdf> [Datum citace 8. 10. 2010]
- [16] <<http://www.odbornecasopisy.cz>>
http://www.odbornecasopisy.cz/index.php?id_document=27136 [Datum citace 10.3.2010]
- [17] <<http://www.micro.feld.cvut.cz>>
http://www.micro.feld.cvut.cz/home/zahlava/ppn/prednasky/Vlastnosti_DPS.pdf [Datum citace 10. 3. 2010]
- [18] Čajka Jiří, Bakalářská práce, 2010, podpora řízení procesů pomocí PDA zařízení na platformě .NET Compact Framework
- [19] < <http://www.codeproject.com> >
<http://www.codeproject.com/KB/cs/SMS.aspx> [Datum citace 22. 10. 2010]

- [20] <<http://studiodavinci.czechian.net>>
http://studiodavinci.czechian.net/MalyUvodDoCsharp.pdf [Datum citace 15.3. 2010]
- [21] LACKO, I. [i]Programujeme mobilní aplikace ve Visual Studiu .NET.[/i] 1. vyd. Brno: Computer Press, 2004. 479 s. ISBN 80-251-0176-2.
- [22] Haerick, W., Wauters, T., Develder, C., De Turck, F., Dhoedt, B., ANNALS OF TELECOMMUNICATIONS, Volume 65, Issue 7-8, pp. 419-432, 2010, DOI: 10.1007/s12243-010-0168-0

Seznam příloh

Extended Abstract.....	1
------------------------	---

Extended Abstract

This work is in developing applications for mobile phones that use the values displayed on the touch screen simply displays the current status of the remote device. Increased comfort is achieved by visualization applications by using components created. Processing and easy-to-use application that is currently designed for GSM communicator GD-06 Allegro is facilitated by a remote control device. The second part is exactly the opposite functional application that is portable PDA. The application simulates a GSM communicator, and acts as a remote device. Both applications are designed and constructed for the platform .NET Compact Framework.

1. Introduction

With the ever expanding market for PDAs and numerous uses, there are many ways to communicate between devices. PDA may communicate only among themselves but also with remote devices and GSM communicators. Communication skills are using SMS, MMS, GPRS, etc.

By developing applications using .NET Compact Framework, you can use many features of the PDA. We also have a great opportunity to use graphical environment that allows us the platform. The visualization and graphical components are in the final stage of the application easily and quickly understand and easy to use. Using the generated application can communicate with GSM communicators. Accordingly, for what purpose and with what protocol the application works, you can specify any process.

The subject of my thesis is the first part of a single universal application for communication and remote control devices. According to the assignment to work with commercial GSM communicator, GD-06 Allegro. With graphics applications processed quickly and easily communicate via SMS.

The second part will deal with creating a reverse application that simulates a GSM communicator and PDA, which should be started creating an application, is connected controlled device (heating).

PDA acronym comes from the English connection Personal Digital Assistant, which translates to a personal digital assistant. Control is implemented largely through the touch screen depending on design, either with a stylus with the stylus or your fingertips on the hand. Currently in use as a PDA operating systems: Windows Mobile, Palm OS, Symbian, OS and Linux, Android.

In the early development of electronic helpers formed a data bank, which served to organize contacts and short notes. The development of other technologies and communications PDA. The first models can be classified among the multifunctional calculators or electronic diaries, which consisted of a small keypad, monochrome display and a very small memory. Over time, the incremented internal memory and later placed at the external memory card. The displays have been enlarged and moved to the monochrome and color touch. In addition, developers have increased the performance of the processor and adding a variety of communication options. As one of the first option was to infrared, then the Bluetooth, Wi-Fi, and of course the GSM communication

2. Problem Definition

In this chapter, the definitions of each application. Each section deals with analysis of how applications should operate and what is the proposal for their construction.

2.1. Applications for mobile devices that perform remote device monitoring

Application is developed in an environment developed by Microsoft Visual Studio .NET Compact Framework. It communicates via text message. The application was created by award used components from Mr. Čajka.

Communication solutions in the application via SMS, it seemed to be the simplest way for this award and sufficient. The aim of a remote control device using SMS directly from the proposed application process facilitates communication and eliminates the problem of communicating directly by typing text messages. The application was initially dealt only with the basic toolbox menu in Visual Studio C#. Over development application has been modified to operate with components. The use of components in the application increases the overall aesthetic impression easier and faster with the application. One of offering solutions to control its use directly from your computer. The problem occurs when the management needs of GSM communicator and not a PC. Control directly from a PC is impractical from the grounds, it must be connected device that supports sending data to the GSM network or sent over the internet

2.2. Applications for mobile devices replacing the GSM communicator

The application is created in the same environment as the application driver. Building was very similar. It is also true for the visual appearance. Embedded graphics are used from the previous applications and their functionality is the same.

In cases where we do not have any GSM communicator, offers the possibility of using a mobile phone or PDA running WM. Application to replace the GSM communicator has the advantage that we are limited in number and outputs can be programmed immense amount of features. We are only limited possibilities for mobile devices. In terms of the cost of the acquisition of mobile phone applications cheaper than buying a GSM communicator.

The disadvantage of using this method of control and management is that the GSM communicator stable than WM freezes and its reliability is high. For mobile devices, problems arise with freezing. Occasionally, it may collapse running application or communication between the connected elements.

3. New Solution

The new solution shows how applications are created and their description of their functionality is outlined. The elements of the display, has its own function serving as one of the important processes.

3.1. Applications for mobile devices that perform remote device monitoring

When you start, the main area where the most controls. At the same time you can also see the last known state of the remote device. To determine the status of the state of the mouse click and press the green button Allegro in the upper left. Within a short time directly on the PDA screen changes the values and views of individual parts. In the upper right of the display shows the current value of temperature at the remote device is installed. Under the button Allegro is a switch to change between manual and automatic control of heating type. In the bottom left of the screen you will find a blue block, which serves as a demonstration of the remote terminal device. Each terminal is identified as P1 – 6. The formation was calculated with the fact that more terminals in a controlled facility will be. In the case of controlling another device that has multiple terminals, such applications would be insufficient.



Picture 1 - Original and contemporary look of the application

3.2. Applications for mobile devices replacing the GSM communicator

This application is designed for applications like remote control device. Its functionality is replaced by the GSM communicator GD-06 Allegro, because the functional commands, which respond to, are written for this device. Construction elements (components) are the same as used in the control application. Allegro component, which used to read incoming SMS and send commands is remodeled and named as SMS, see further description. When the command comes from a telephone number that is set, commands will evaluate applications and make all operations. Set the visualization to the desired value and sends back an SMS with the values in the phone. The added component thermometer shows the temperature value in the location of the PDA. In this solution, the application generates the temperature only through the algorithm and displayed on your phone using the added component thermometer and LED displays, showing the exact value of temperature.



Picture 2 - Application of replacing the GSM communicator

3.3. Custom made components for remote control



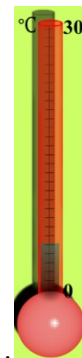
Picture 3 - component ALLEGRO

Allegro component is purely to communicate with a remote device, which was delivered for testing. By creating this component to achieve greater compatibility, while simplifying and shortening the main code. Using global variables to communicate with the application component Allegro. Creating new components can communicate with other devices and other peripherals with than just using SMS.



Picture 4 - Component Terminal

This version replaces the original terminal solution using the checkboxes. Its functionality is very simple. To turn on or off the terminal, just press the gray circle. When you turn green, it means that the terminal is turned on. If we act the same again, will turn gray button and thus turns off the terminal. Using global variables to communicate with other components or the main application. It depends on which point we call these variables. The programming terminal components were created "event", an event with which we work. These events serve as a single click on the dots. Defining the "click" to work on individual terminals, made using the handle and classes. The background of this component is called by the code, but the individual wheels are embedded picture boxes, which when clicked changes the picture

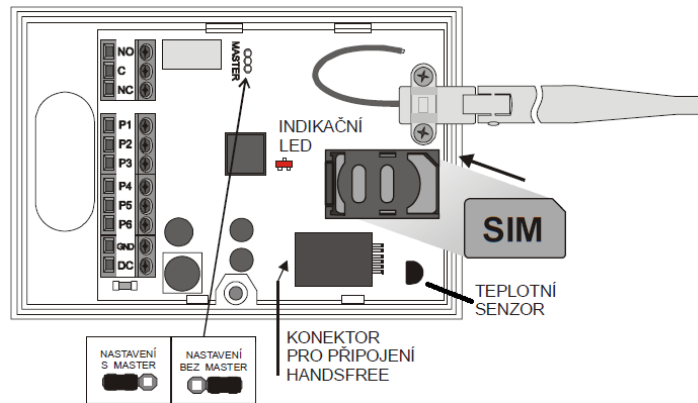


Picture 5 - Component Thermometer

The thermometer component is used to demonstrate the actual measured temperature values in the neighborhood. After receiving the command to turn on or off the heating system or an automatic temperature control component perform the appropriate command. If the component signal as 3 ° C and received SMS shows Order "switch a relay (turn on the heater), the timer starts, which is used to slowly add the value of temperature on the thermometer. Change the appearance and function of change in level is formed by overlapping images. The thermometer has a range from 0 to 30 ° C.

4. GSM Communicator GD-06 Allegro

Allegro device functions as multi-channel universal GSM equipment. Control is carried out via SMS, via the Internet, through GPRS or by dialing the specified telephone number. Information about the current state can be sent to up to eight phone numbers at once. If you connect the hands free kit and then call the number and dialer can make calls. This product is intended for indoor use. The best location is where the greatest signal strength.



Picture 6 - Views GSM communicator GD-06 Allegro

5. Conclusions

At the beginning I was creating this work environment, discussing the theory offered by Microsoft. Description of individual chapters shows us what the options are different types of programming environments. The theoretical analysis of communication via SMS, MMS and GPRS readers were shown a fundamental aspect of the issues relating to communications. As already mentioned in the previous chapter, it is possible to get a full picture of how to program these types of communications. At first glance you can easily conclude that programming of communication via SMS is simple. This kind of sending and receiving data, it seems appropriate in this application, but we need to consider how and what data will be sent and determines the type of communication.

In C # I wrote an application that is used to control existing remote device allegro GD-06, via SMS. Application was made up of several components, both created in the previous years; I've got available, as well as components that I had myself created. Its form is done so that compatible and expandable at any time by adding any new components that will operate as another type of communication.

Design of applications that perform remote device is shown how the possible replacement of GSM communicator. Create an application does not function as a communicator, because it is not connected to any device using a mobile phone. This solution is a simulator of the GSM communicator GD-06 Allegro. If necessary, the real recovery is possible by crediting a relatively simple code to define the connection with the device.

This theme opens up many more issues and sub issues, which would be a good deal further, and new research would certainly bring us a variety of useful knowledge.