

uSMS

DEFINICE ZÁKLADNÍCH KONSTANT A TYPŮ PRO KOMUNIKAČNÍ KNIHOVNU CHNSMS

Příručka uživatele a programátora



SofCon[®] spol. s r.o.
Střešovická 49
162 00 Praha 6
tel/fax: +420 220 180 454
E-mail: sofcon@sofcon.cz
www: <http://www.sofcon.cz>

Informace v tomto dokumentu byly pečlivě zkontrolovány a SofCon věří, že jsou spolehlivé, přesto SofCon nenese odpovědnost za případné nepřesnosti nebo nesprávnosti zde uvedených informací.

SofCon negarantuje bezchybnost tohoto dokumentu ani programového vybavení, které je v tomto dokumentu popsáno. Uživatel přebírá informace z tohoto dokumentu a odpovídající programové vybavení ve stavu, jak byly vytvořeny a sám je povinen provést validaci bezchybnosti produktu, který s použitím zde popsaného programového vybavení vytvořil.

SofCon si vyhrazuje právo změny obsahu tohoto dokumentu bez předchozího oznámení a nenese žádnou odpovědnost za důsledky, které z toho mohou vyplynout pro uživatele.

Datum vydání: 08.02.2004

Datum posledního uložení dokumentu: 08.02.2004

(Datum vydání a posledního uložení dokumentu musí být stejné)

Upozornění:

V dokumentu použité názvy výrobků, firem apod. mohou být ochrannými známkami nebo registrovanými ochrannými známkami příslušných vlastníků.

Obsah :

1. O dokumentu	5
1.1. Revize dokumentu	5
1.2. Účel dokumentu	5
1.3. Rozsah platnosti	5
1.4. Související dokumenty	5
2. Termíny a definice	5
3. Úvod	6
4. Konstanty a typy	6
4.1. Řetězcové typy	6
4.2. Výčtové typy	6
4.3. Struktura vysílacích a přijímacích bufferů	6
4.4. Konstanty typů telefonních čísel	7
4.5. Konstanty identifikátorů protokolu	8
4.6. Konstanty kódování dat SMS zprávy	8
5. Struktura SMS zpráv v PDU formátu	8
5.1. Struktura zpráv v přijímacím formátu	8
5.2. Struktura zpráv ve vysílacím formátu	10
6. Procedury a funkce	11
6.1. GetSMS_PDUStr funkce	11
6.2. GetPDU_SMSStr funkce	11
6.3. DecodeRelTime procedura	11
6.4. CodeRelTime funkce	11
6.5. DecodeRecMsg funkce	11
6.6. CodeSendMsg funkce	12

1. O dokumentu

1.1. Revize dokumentu

Verze dokumentu	Verze SW	Autor	Datum vydání	Popis změn
1.00	1.XX	Wi		První vydání.
1.10	4.XX	Tu	02.06.2003	Úprava dokumentu dle ISO9000.
1.11	4.XX	Wi	08.02.yyyy	Úprava formátu dokumentu.

1.2. Účel dokumentu

Tento dokument slouží jako popis jednotky definující základní konstanty a typy pro komunikační knihovnu ChnSMS.

1.3. Rozsah platnosti

Určen pro programátory a uživatele programového vybavení SofCon.

1.4. Související dokumenty

Pro čtení tohoto dokumentu není potřeba číst žádný další manuál, ale je potřeba se orientovat v používání programového vybavení SofCon.

Popis formátu verze knihovny a souvisejících funkcí je popsán v manuálu „LibVer“.

2. Termíny a definice

Obecně používané termíny a definice jsou popsány v samostatném dokumentu „Termíny a definice“.

Termíny týkající se přenosu krátkých textových zpráv (SMS) jsou popsány v dokumentu „ChnSMS“.

3. Úvod

Knihovna **uSMS** definuje základní konstanty, typy a konverzní funkce využívané knihovnou **ChnSMS** i aplikací pro posílání a příjem krátkých textových zpráv (SMS) prostřednictvím GSM modemu.

4. Konstanty a typy

Tato jednotka má význam jen spolu s jednotkou ChnSMS a její verze odpovídá verzi uvedené jednotky - **uSMS** tedy neobsahuje konstanty popisující verzi jednotky: cVerNo a cVer.

4.1. Řetězcové typy

```
tStringSMS = string[160];
```

Typ řetězce pro vlastní text SMS zprávy v textovém formátu.

```
tStringPDU = string[255];
```

Typ řetězce pro vlastní text SMS zprávy v protokolovém (PDU) formátu.

```
tPhoneNumStr = string[16];
```

Typ řetězce pro telefonní číslo.

4.2. Výčtové typy

```
tTypTime = (tmAbsTime, tmRelTime, tmNoTime);
```

Typ **tTypTime** definuje výčet různých formátů času v SMS zprávě. Obsahuje následující položky:

tmAbsTime - zpráva obsahuje absolutní datum a čas

tmRelTime - zpráva obsahuje relativní datum a čas

tmNoTime - zpráva neobsahuje datum a čas vůbec

```
tMsgStat = (tsRecUnRead, tsRecRead, tsStoUnSent, tsStoSent, tsAll);
```

Typ **tMsgState** definuje výčet základního statusu přijaté SMS zprávy. Obsahuje následující položky:

tsRecUnRead - jedná se o přijatou a ještě nepřečtenou (novou) zprávu

tsRecRead - jedná se o přijatou a již přečtenou (starou) zprávu

tsStoUnSent - jedná se o zprávu pro vyslání, ale která ještě nebyla vyslána

tsStoSent - jedná se o zprávu pro vyslání, která již byla vyslána

tsAll - jedná se o jakoukoliv zprávu

4.3. Struktura vysílacích a přijímacích bufferů

```
pSendRecord = ^tSendRecord;
```

```
tSendRecord = record
```

```

  MsgStat      : tMsgStat;      { základní status zprávy }
  PDU_Type     : byte;          { typ PDU zprávy }
  MsgReference : byte;          { !pouze pro vysílanou zprávu }
  PhoneNumType : byte;          { typ telefonního čísla }
  PhoneNum     : tPhoneNumStr;  { telefonní číslo (Originator
                                Adress/Destination Adress) }
  Protocol_ID  : byte;          { Protocol Identifier }

```

```

DataCodScheme: byte;          { Data Coding Scheme }
SMS_Message   : tStringSMS;   { vlastní zpráva }
case TypTime  : tTypTime of
  tmAbsTime:
    (Time      : DateTime;     { absolutní datum a čas }
     TimeZone: shortInt;      { posunutí oproti světovému času }
    );
  tmRelTime:
    (Mins      : word;         { relativní počet minut }
     Days      : word;         { relativní počet dnu }
    );
  tmNoTime:
    ();
end;

```

TSendRecord je typ variantního záznamu, který svou strukturou odpovídá struktuře SMS zprávy v protokolovém (PDU) režimu pro vysílání zpráv. Položka **MsgStat** udává, zda se jedná o novou přijatou zprávu, starou přijatou zprávu, zprávu pro vyslání apod. (viz. typ **tMsgStat**). Položka **PDU_Type** reprezentuje první znak SMS zprávy (viz. Konstanty kódování dat SMS zprávy). Položka **MsgReference** se používá pouze v případě zprávy ve vysílacím formátu (většinou se nastavuje na hodnotu 00h). Položka **PhoneNumType** určuje typ zadaného telefonního čísla (viz. Konstanty typů telefonních čísel). Položka **PhoneNum** obsahuje vlastní telefonní číslo včetně předvoleb. Položka **Protocol_ID** obsahuje kód identifikátoru zprávy (viz. Konstanty identifikátorů protokolu). Položka **DataCodScheme** obsahuje schéma kódování vlastních dat SMS zprávy (viz. Konstanty kódování dat SMS zprávy). Položka **SMS_Message** obsahuje vlastní text SMS zprávy. Položka **TypTime** určuje, jaký je použit formát datumu a času (viz. **tTypTime**). V případě použití absolutního času se dále používá položka **Time**, která obsahuje datum a čas ve struktuře záznamu **DateTime**, a položka **TimeZone**, která obsahuje číslo časového pásma. V případě použití relativního času se používá položka **Mins**, která udává relativní počet minut, a položka **Days**, která udává relativní počet dnů.

```

pRecRecord = ^tRecRecord;
tRecRecord = tSendRecord;

```

TRecRecord je typ variantního záznamu, který svou strukturou odpovídá struktuře SMS zprávy v protokolovém (PDU) režimu pro příjem zpráv. Svou strukturou odpovídá záznamu **TSendRecord**.

4.4. Konstanty typů telefonních čísel

Tyto konstanty jsou pro položku **PhoneNumType** ve struktuře vysílacího a přijímacího bufferu.

```

cPNT_National = $81;

```

Konstanta **cPNT_National** určuje, že telefonní číslo je národní.

```

cPNT_Internat = $91;

```

Konstanta **cPNT_Internat** určuje, že telefonní číslo je mezinárodní.

Pozn: Mohou existovat i další konstanty např. pro číslo reprezentující E-mail apod.

4.5. Konstanty identifikátorů protokolu

Tyto konstanty jsou pro položku Protocol_ID ve struktuře vysílacího a přijímacího bufferu.

cPID_SMS = \$00;

Zpráva je SMS.

cPID_Telex = \$01;

Zpráva je Telex (v této verzi neimplementováno).

cPID_TeleFax3 = \$02;

Zpráva je TeleFax 3 skupiny (v této verzi neimplementováno).

cPID_TeleFax4 = \$03;

Zpráva je TeleFax 3 skupiny (v této verzi neimplementováno).

Pozn: Pro vysílání a příjem SMS zpráv se používá pouze konstanta cPID_SMS. Mohou existovat i další konstanty.

4.6. Konstanty kódování dat SMS zprávy

cDCS_Impl7Bit = \$0000;

Tato konstanta určuje, že vlastní text SMS zprávy je zakódován v implicitním 7-mi bitovém kódování.

5. Struktura SMS zpráv v PDU formátu

Podle nejspodnějších dvou bitů prvního znaku SMS zprávy rozlišujeme několik formátů zpráv.

Určení formátu zprávy dle 1. znaku SMS zprávy (PDU_Type):

bity	7 .. 2	1 0
význam	závisí na formátu (viz níže)	MTI

MTI = 00b - zpráva je v přijímacím formátu (pro přijaté SMS zprávy od operátora SMSC nebo případnou odpověď na přijatou zprávu operátorovi)

MTI = 01b - zpráva je ve vysílacím formátu (pro SMS zprávy pro odvysílání a přijaté odpovědi od operátora SMSC zpráv na vyslání SMS zprávy)

MTI = 10b - rezervováno

MTI = 11b - rezervováno

5.1. Struktura zpráv v přijímacím formátu

PDU_Type	LenPN	PNT	Phone Number	PID	DCS	Time	LenSM	Data of SMS
----------	-------	-----	--------------	-----	-----	------	-------	-------------

Všechny položky zprávy jsou v ASCII a odpovídají příslušným položkám ve struktuře přijímacího bufferu typu **tRecRecord** (viz výše)

PDU_Type - 1. znak SMS zprávy v PDU formátu (byte)

7	6	5	4	3	2	1 0
RP	UDHI	SRI	0	0	MMS	MTI

RP (Reply Path) - bit indikující, že existuje "Reply Path"

UDHI (User Data Header Indicator) - bit indikující, že textové pole Data of SMS obsahuje hlavičku

SRI (Status Report Indication) - bit indikující, že byla přijata status odpověď od operátora SMSC

MMS (More Messages to Send) - bit indikuje, že není dalších zpráv

MTI - (viz výše)

LenPN - byte určující délku telefonního čísla (Length(PhoneNum))

PNT - byte určující typ telefonního čísla (položka PhoneNumType v tRecRecord) - viz. Konstanty typů telefonních čísel

Phone Number - řetězec telefonního čísla včetně případných předvoleb kódovaný z položky PhoneNum v tRecRecord následujícím způsobem:

Cifry ve dvojicích čísel v položce PhoneNum jsou v PhoneNumber prohozené a pokud Phone Number má lichý počet cifer doplní se znakem \$F, např. telefonní číslo „1234567“ zadávané do položky PhoneNum se do položky Phone Number zakóduje jako „214365F7“

PID - položka Protocol_ID v tRecRecord - viz. Konstanty identifikátorů protokolu

DCS - položka DataCodScheme v tRecRecord - viz. Konstanty kódování dat SMS zprávy

Time - absolutní datum a čas (položka TypTime v tRecRecord = tmAbsTime), které se z položek Time a TimeZone z tRecRecord kóduje následovně:

Jako první se udává poslední dvojčíslí roku, potom číslo měsíce zarovnané na dvě cifry, potom den v měsíci rovněž zarovnaný na dvě cifry, dále následují hodiny, minuty a sekundy rovněž zarovnané na dvě cifry a jako poslední je číslo časové zóny také na dvě cifry. Čili všechny části datumu a času jsou zadány v podobě pevných dvou cifer. Další důležitou věcí je to, že obě cifry u všech částí jsou vždy prohozeny. Např. datum a čas 13.6.1998 12:05:50 v položce Time v tRecRecord a časová zóna 0 v položce TimeZone v tRecRecord se zakóduje jako „89 60 31 21 50 05 00“.

LenSM - délka vlastní textové zprávy ve znacích (Length(SMS_Message))

Data of SMS Message - řetězec vlastní textové zprávy kódovaný (v případě DataCodScheme = cDCS_Impl7Bit) z položky SMS_Message v tRecRecord následujícím způsobem:

Každý znak zprávy se vejde do 7-mi bitové reprezentace. Proto je nejvyšší bit prvního znaku použit jako nejnižší bit následujícího znaku. Z tohoto druhého znaku je už jen třeba 6 bitů, a proto nyní zbudou 2 nejvyšší bity, které jsou použity jako nejnižší bity následujícího (třetího) znaku atd. atp.. Z toho plyne, že 8 znaků zprávy lze do Data of SMS Message zakódovat v podobě jen 7-mi znaků. např. zpráva „hello“ se zakóduje následovně:

bit 7	1	h	110 0000	h	1 110 1000	E8h
		e	11 0010 1	e	00 11 0010 1	32h
		l	1 1011 00	l	110 1 1011 00	9Eh
		l	1101 100	l	1111 1101 100	FDh
		o	110 1111	o	0000 0 110 1111	06h

Data of SMS Message = „E8 32 9B FD 06“

5.2. Struktura zpráv ve vysílacím formátu

PDU_Type	MR	LenPN	PNT	Phone Number	PID	DCS	Time	LenSM	Data of SMS
----------	----	-------	-----	--------------	-----	-----	------	-------	-------------

Všechny položky zprávy jsou v ASCII a odpovídají příslušným položkám ve struktuře přijímacího bufferu typu **tSendRecord** (viz výše)

PDU_Type - 1. znak SMS zprávy v PDU formátu (byte)

7	6	5	4	3	2	1	0
RP	UDHI	SRR	VPF		RD	MTI	

RP (Reply Path) - bit indikující, že existuje "Reply Path"

UDHI (User Data Header Indicator) - bit indikující, že textové pole Data of SMS obsahuje hlavičku

SRR (Status Report Request) - bit indikující, že GSM modem (MS) požaduje status odpověď od operátora SMSC

VPF (Validity Period Format) - 2 bity indikující formát datumu a času v položce zprávy Time

VPF význam

00 položka zprávy Time vůbec neexistuje

01 rezervováno

10 položka zprávy Time je byte a obsahuje relativní čas

11 položka zprávy Time obsahuje absolutní datum a čas

RD (Reject Duplicate)

MTI - (viz výše)

MR - položka MsgReference v tSendRecord

Ostatní položky zprávy mají stejný význam jako u zprávy v přijímacím formátu, kromě položky **Time**, která nemusí být zastoupena nebo může obsahovat relativní čas s následujícím kódováním:

Time ve zprávě	Mins či Days v tSendRecord
0..143	Mins = (Time + 1) * 5
144..166	Mins = 720 + (Time - 143) * 30
167	Days = 1
168..196	Days = (Time - 166) * 1
197..255	Days = (Time - 192) * 7

6. Procedury a funkce

6.1. GetSMS_PDUStr funkce

```
function GetSMS_PDUStr(S:tStringPDU):tStringSMS;
```

Funkce **GetSMS_PDUStr** převede řetězec znaků SMS zprávy v PDU módu v implicitním 7-mi bitovém kódování na řetězec znaků ASCII (viz. výše).

6.2. GetPDU_SMSStr funkce

```
function GetPDU_SMSStr(S:tStringSMS):tStringPDU;
```

Funkce **GetPDU_SMSStr** převede řetězec znaků SMS zprávy v ASCII na řetězec znaků v PDU módu v implicitním 7-mi bitovém kódování (viz. výše).

6.3. DecodeRelTime procedura

```
procedure DecodeRelTime(B:byte; var Mins,Days:word);
```

Procedura **DecodeRelTime** rozdekóduje relativní čas zakódovaný v B a vyplní položky Mins a Days.

6.4. CodeRelTime funkce

```
function CodeRelTime(Mins,Days:word):byte;
```

Funkce **CodeRelTime** zakóduje relativní čas v Mins či Days do bytu, který vrátí jako svou funkční hodnotu.

6.5. DecodeRecMsg funkce

```
function DecodeRecMsg(RMess:pRecRecord; S:string):word;
```

Funkce **DecodeRecMsg** dekóduje celou přijatou SMS zprávu v PDU módu s implicitním 7-mi bitovým kódováním vlastního textu zprávy do bufferu RMess a vrátí jeho délku.

6.6. CodeSendMsg funkce

```
function CodeSendMsg(SMess:pSendRecord):string;
```

Funkce **CodeSendMsg** kóduje zprávu v záznamu vysílacího bufferu SMess do vysílacího řetězce SMS zprávy v PDU módu s implicitním 7-mi bitovým kódováním vlastního textu zprávy.