

## **Labo datacommunication : versturen van sms**

### ***Doel***

Het versturen van sms berichten via AT commando's. We maken gebruik van een GSM/GPRS module om sms'en te versturen en te ontvangen.

### ***Wat is een modem?***

Een modem (MOdulator/DEModulator) is een apparaat welke intern in de computer is ingebouwd of extern aan de computer wordt gekoppeld om met een andere computer te kunnen communiceren over de analoge telefoonlijn. Een modem zet de data, die verzonden moet worden, om naar analoge signalen. De ontvangen data wordt door de modem weer omgezet naar door de computer begrijpbare data (digitaal). In andere woorden: de modem maakt data gereed voor verzending en maakt bij het ontvangen van data deze weer leesbaar voor de computer.

In het verleden werden modems alleen gebruikt in combinatie met de telefoonlijn, tegenwoordig wordt de modem ook gebruikt voor het verzenden en ontvangen van data over de kabel (kabelmodem). Ook een adsl-modem is een modem, welke computerdata omzet naar analoge signalen. De snelheid van een adsl-modem is wel tien keer sneller dan een standaard modem, een adsl-modem gebruikt een andere techniek voor het verzenden/ontvangen van data.

### ***Wat is de Hayes AT-commandoset?***

Bijna alle modems werken met de Hayes AT-commandoset. Bijna alle commando's die naar de modem worden gestuurd, beginnen met AT. Deze commando's kunnen worden ingegeven in het terminal gedeelte van bijvoorbeeld Terminal of Hyperterminal voor Windows. Een commando begint altijd met AT van attention. Daarna wordt het daadwerkelijke commando ingegeven.

Tegenwoordig bevat de Hayes AT commando set commando's voor data, fax, spraak en SMS communicatie. Vanwege de eenvoud en de low-cost implementatie werd de Hayes AT commando set al snel overgenomen in modems van verschillende fabrikanten. Naarmate functionaliteit en complexiteit van modems vergrote in de loop der tijd, werd ook de Hayes AT commandotaal steeds uitgebreider en al snel implementeerde elke modemfabrikant zijn eigen dialect.

Via de seriële poort kunnen de at commando's naar het toestel verstuurd worden.

## Testen via hyperterminal of putty

Maak een verbinding via hyperterminal met de gsm/GPRS module. Stel de verbinding in op geen handshaking, een baudrate van 9600, 1 stopbit, een datalengte van 8 bits en geen pariteitsbit.

Als we een commando intikken, b.v. AT (niet case sensitive) dan stuurt de module onmiddellijk elk karakter terug, we krijgen dus het commando te zien in hyperterminal.

Na elk commando moeten we een return ingeven.

AT <CR>

OK

Als alles goed verlopen is moeten we een ok terug vinden.

In het pdf document "At commands interface" kan je al de commando's voor dat type toestel terug vinden. De meeste commando's zijn echter bruikbaar op alle toestellen.

Probeer eens enkele commando's uit.

- Vraag het type van het toestel op.
- Vraag het merk van het toestel op.
- Vraag de datum en de tijd van het toestel op.

Sluit de hyperterminal af.

## Ontwikkel een applicatie

Start een nieuwe applicatie op. Voorzie jullie project van een tekstvak om commando's naar de gsm te versturen en een tekstvak om de data die de gsm terugstuurt te ontvangen.

Maak eerst een verbinding met de GSM via de correcte COM poort. Maak hiervoor gebruik van de serialport component. Stel de serialPort component in op : geen handshaking, een baudrate van 9600, 1 stopbit, een datalengte van 8 bits en geen pariteitsbit, **RTS enabled true.**

Via de commando's read, readexisting en write kunnen we lezen en schrijven naar de seriële poort.

Test enkele commando's uit in jullie applicatie. **Vergeet de <CRLF> niet!!!**

Probeer een lijst van alle sms berichten te krijgen. De berichten kunnen gestockeerd worden op de sim kaart ofwel in het geheugen van het toestel zelf.

Afhankelijk van het geselecteerde geheugen krijg je al of niet de berichten te zien.

Zoek de instructies op om te switchen tussen het simgeheugen en het geheugen van het toestel zelf. Eventueel kunnen de beide geheugens geselecteerd worden.

Je krijgt een stream van bytes terug. Die bytes staan helaas in het PDU format. We gaan straks zien hoe we het PDU format kunnen omzetten.

De meeste GSM toestellen werken in het PDU format dit is een format die 7 bits voor een karakter gebruikt ipv 8. Van enkele toestellen kan je dat formaat aanpassen naar het tekstformaat dat bestaat uit 8 bits per karakter. Nu beschik je wel maar over 140 karakters per sms ipv 160. Maar met het tekstformaat werken is wel veel eenvoudiger.

Je merkt dat het omzetten van tekst naar dat format niet zo eenvoudig is. Het PDU format bevat niet enkel de te versturen data maar ook de lengte van het bericht(enkel bij het lezen van berichten), het serviceCenterNumber, telefoonnummer waar het bericht naar toe moet, format van telefoonnummers.

Gelukkig kan je op het internet wel klassen terugvinden die die klus voor ons opknappen.

Een voorbeeld van zo'n klasse kan je in de map benodigdheden terugvinden.

Voeg de referentie smsEncoder.dll toe aan je project. Die dll bevat de klassen smsPDUEncoderDA en de klasse SMS. In de klasse sms wordt er eens sms object aangemaakt. De DA klasse beschikt over twee functies die we nodig hebben om een sms te versturen.

Namelijk GetPDU, GetPDUALength

De functie GetPDU verzamelt al de info die in de PDU string moet verzameld worden. Het sms bericht moet meegegeven worden als argument, alsook nog enkele instellingen. Die instellingen staan in commentaar vermeld in de klasse zelf. Nog een argument van de GetPDU functie is het ServiceCenterNumber.

Proximus: +32 475 16 16 16

Mobistar: + 32 495 00 25 30

BASE: +32 486 00 00 05

Let op dat je al de GSM nummers met 11 karakters en een plus teken weergeeft.

Met #124# kan je het nummer van de GSM te weten komen.

De functie GetPDU heeft ons bericht terug in PDU format. Laat dit PDU format zien in jouw vb applicatie.

Vooraleer we een bericht kunnen versturen moeten we eerst de lengte van het bericht meegeven. De lengte van het bericht kunnen we opvragen met de functie GetPDULength.

Hoe het bericht nu verstuurd moet worden kan je terugvinden in de at command manuals. Na het meegeven van de lengte moet je eventjes wachten om het PDU format mee te geven. Na het PDU format moet er nog een ctrl z komen.

```
AT+CMGF=0      //Set PDU mode
AT+CSMS=0      //Check if modem supports SMS commands
AT+CMGS=23     //Send message, 23 octets (excluding the two initial zeros)
>0011000B916407281553F80000AA0AE8329BFD4697D9EC37 <ctrl-z>
```

Probeer het versturen van de sms berichten tot een minimum te beperken, zodat de groepen die na jullie komen nog over voldoende krediet beschikken.

Probeer nu eens om een sms te ontvangen. Zoek zelf een decoder klasse.