

Round Solutions

Embedded Internet mit Easy GPRS

Ben Hoelke
Juli 2003

The reproduction, transmission or use of the present documentation or its contents is not permitted without express written authorization of Telit Mobile Terminals S.p.A.
Offenders may be liable for damages - All right reserved

Copyright © DAI Telecom S.p.A. 2003

Contents

1	OVERVIEW	3
2	GPRS OPERATIONS	4
2.1	Introduction	4
2.1.1.1	CSD application example	Error! Bookmark not defined.
2.1.1.2	GPRS application example	Error! Bookmark not defined.

Telit

1 Overview

This document describes the differences between GSM and GPRS operations and evidences the advantage and the drawbacks of the GPRS system.

NOTE

The information presented in this document is believed to be accurate and reliable. However, no responsibility is assumed by Telit Mobile Terminals S.p.A. for its use, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent rights of Telit Mobile Terminals other than for circuitry embodied in Telit products. This document is subject to change without notice.

Telit

2 EASY GPRS

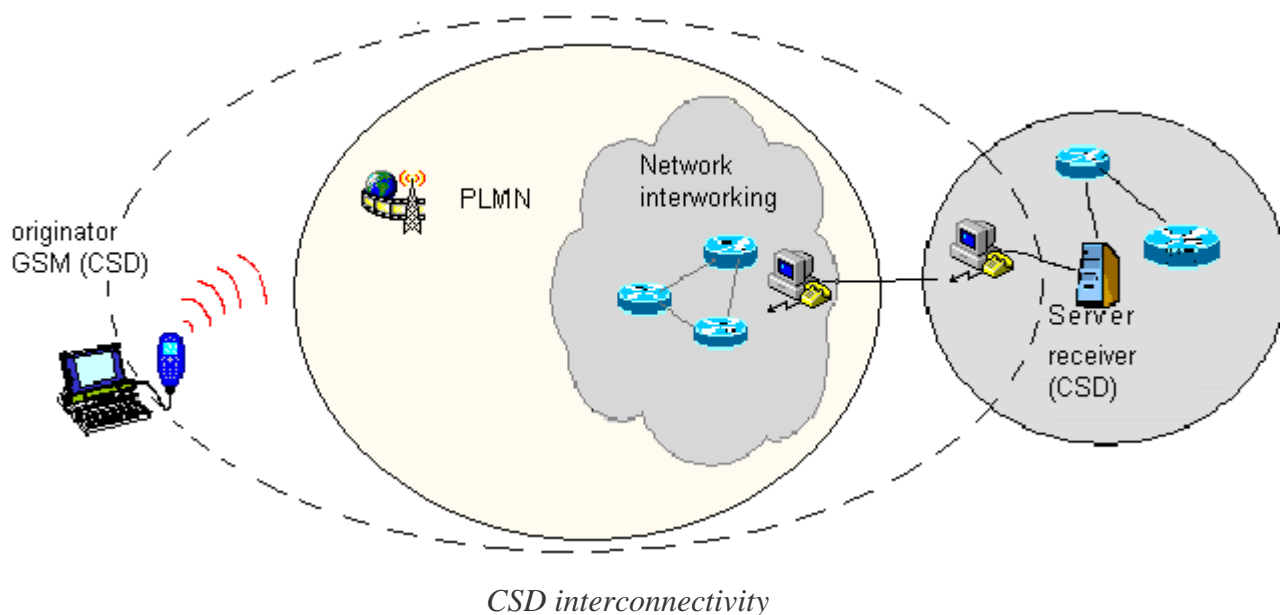
2.1 Introduction: CSD vs. GPRS

The General Packet Radio Services (GPRS) standard permits DATA transfers in a completely different way with respect to previous point to point communications made with Circuit Switch Data (CSD) modems .

In CSD operations the modem establishes a connection with the other party (another modem) in such a way that all the Network in between is transparent to the data exchanged, simulating a real point to point connection, just as if the other party is directly connected with the controlling application of the modem. The other party can be either a Internet Service Provider (ISP) or a private server, but in any case the arrival point must have a modem to connect to (either Landline, ISDN or GSM CSD). The connection establishment procedure defines a particular path where all the information exchanged between the two peers flows and this path is reserved for exclusive use of these 2 peers.

This approach has the drawbacks of a long time to set-up the link between the two peers (up to a minute) and a time counting bill which proceeds even if no data is exchanged because the path resources are reserved anyway; furthermore the speed of the data transfer is limited to 14400 bps.

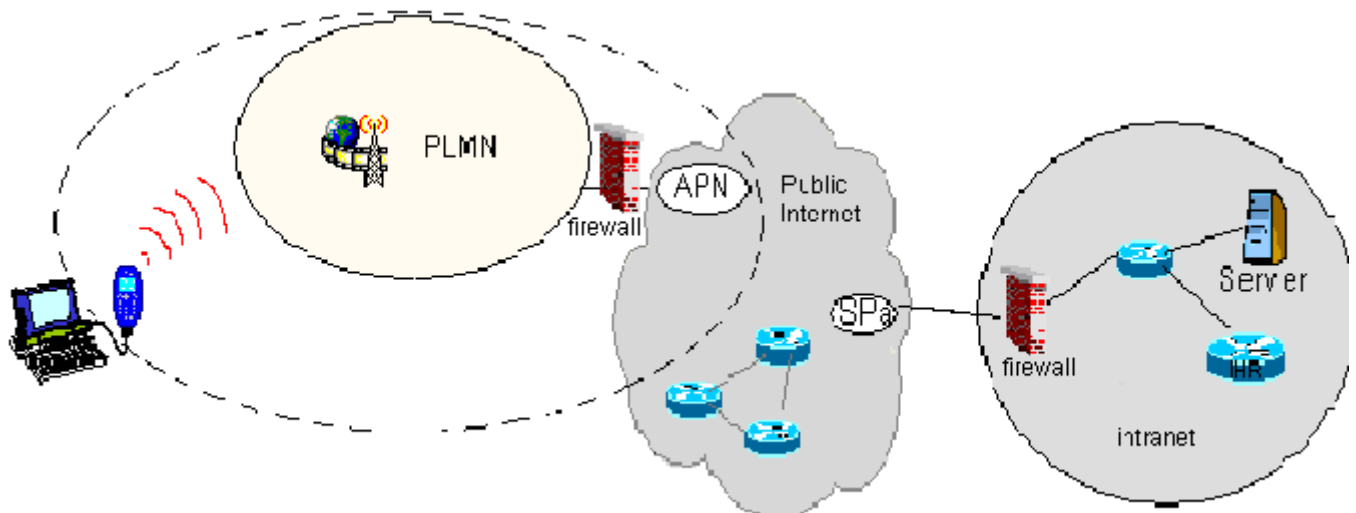
An example of this kind of operation is shown in the following picture, where the point to point connection is between the two peers as if all the devices inside the dashed line are not present:



In GPRS operations instead, the connection is made directly towards internet as if the GPRS modem was a network UDP/TCP/IP socket. There's no data path reserved for the data exchange

between the two peers, instead the resources are allocated dynamically on demand and the data exchanged is organized into UDP/TCP/IP packets, furthermore the maximum transfer speed can be theoretically up to 171,2 kbps, about ten time faster than GSM CSD.

An example of GPRS connection is shown in the following picture, where the GPRS connection is between the GPRS modem and the internet as if all the devices inside the dashed line are not present:



GPRS interconnectivity

Due to this kind of connection, when activating the GPRS connection you must provide the network parameters and not the phone number to be dialed; therefore it is not possible to establish a direct point to point GPRS connection between two GPRS modems, instead an internet TCP/IP connection must be done to achieve a point to point connection between two peers.

This approach as the immediate advantage of projecting the controlling application of the GPRS modem directly on the internet, ready to access the server virtually from anywhere in the world at the same cost; actually the billing of the GPRS connection is based on the amount of data exchanged (number of packets transferred) independently from the time the connection is active or where these packet must be delivered.

There are few considerations than must be done on the GPRS connections:

- the GPRS connection speed with a GPRS class 8 multislot device is asymmetrical, 4 time slots in reception (57600 bps max) and 1 time slot in sending (9600 bps max).
- the GPRS connection speed with a GPRS class 10 multislot device is asymmetrical, 4 time slots in reception (57600 bps max) and 2 time slot in sending (19200 bps max).
- The controlling application must relay on some APN (Access Point Name) that may be the one of the Network Operator of the SIM to gain access to the internet through the GPRS connection. It's the same thing as the Internet provider number for CSD internet access.
- As a consequence of the point before, the receiving application must have internet access either.



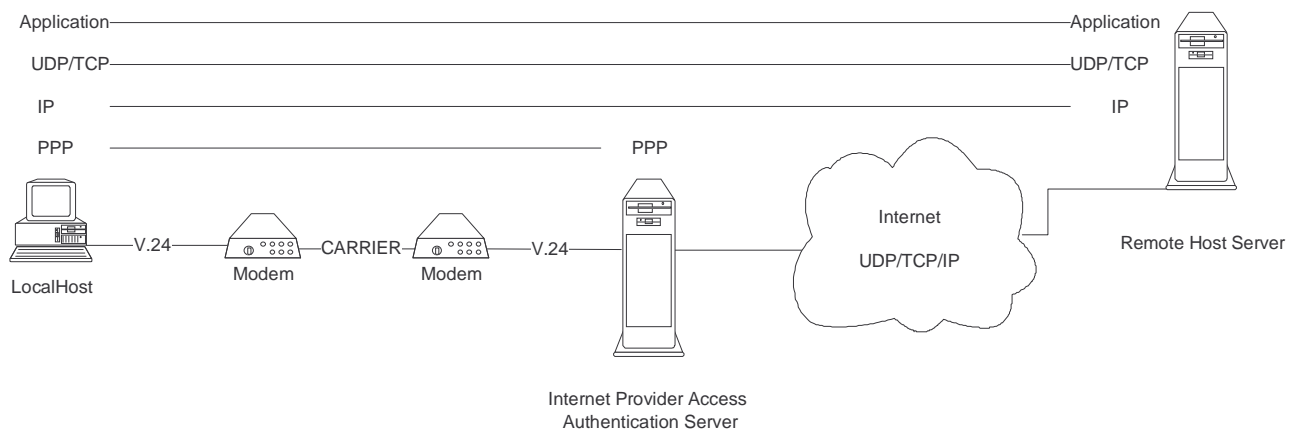
2.2 Seeing GPRS on the "user side"

On the "user side" (we will call it localhost) a GPRS device can be seen as the merge of a landline modem connection and the Internet Provider, since the GPRS device accesses directly internet and does not need an Internet Provider (but it still needs an access point to the internet, that is the APN usually from network operator...)

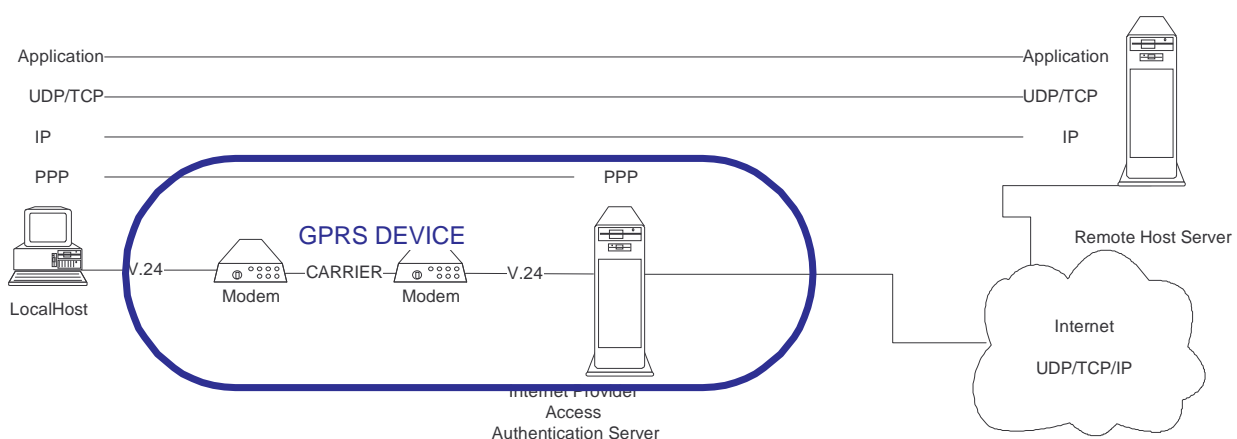
This also means that the GPRS device, once the modem "connect" message (fictitious) starts the PPP (point to point protocol described in RFC1661) between it and the localhost as if the GPRS device was the remote Internet Provider Access Authentication Server Machine.

(Just to let you know, after the PPP authentication PAP phase, the GPRS device uses the same data to authenticate itself on the APN machine with GPRS protocols).

Once established the PPP between the localhost and the GPRS device (this has been made in such a way to keep the GPRS procedure exactly the same as for a CSD internet provider call) the Localhost can exchange through the PPP the TCP/IP packets.



Typical CSD Internet connection

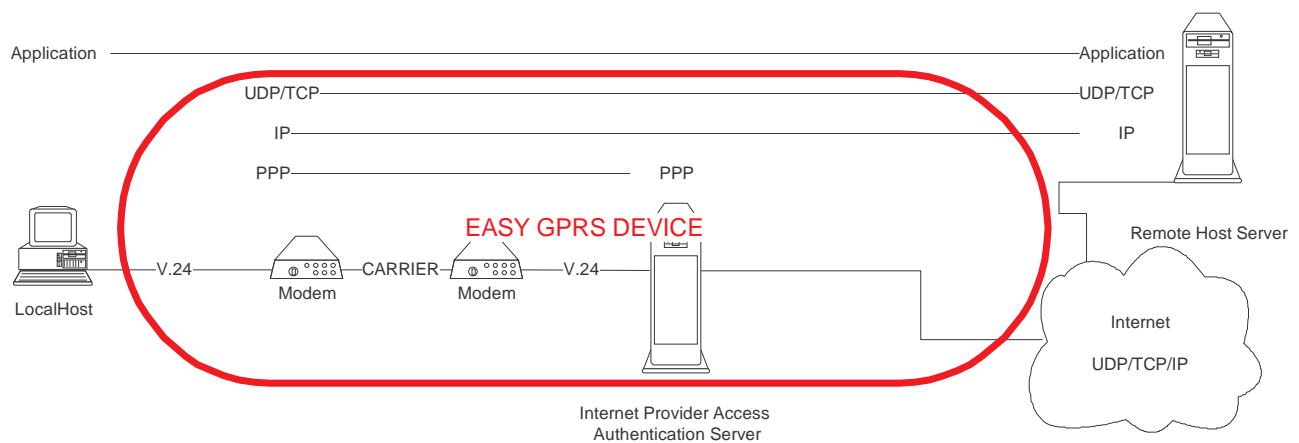


Rough GPRS Equivalent device (modem connection + ISP Access Machine)

2.3 Easy GPRS devices

A GM862-GPRS with its EASY GPRS feature can be seen as a special device, embedding and controlling the PPP/(UDP)TCP/IP protocol stack inside itself.

In this way the Localhost sees a "virtual serial line" connection with the application software on the Server machine.



Rough EASY GPRS Equivalent device (modem connection + ISP Access Machine + TCP/IP stack)

Differently from other GPRS devices that embed the TCP/IP protocol stack; an EASY GPRS device, such as the GM862-GPRS, does not provide a set of API functions to interface with the protocol stack but it automatically manages it internally as specified when starting the connection.

This completely relieves the designer from TCP/IP managing and knowledge.

The connection through the internet with a remote host Application layer residing on the Server can be done as easily as a CSD data call, by following the steps:

- 1- With an AT Command set the GPRS context 1 properties (to allow the GPRS modem to activate later the GPRS connection to internet on demand);
- 2- With an AT Command set the Authentication parameters (UserID and Password) to be used later during the authentication process on the GPRS network;
- 3- With an AT Command define the host by setting the remote host port to be contacted and with which transport protocol (TCP or UDP). There are three things to configure:
 - Transport layer protocol to be used: TCP or UDP
 - Server IP address or name (resolved with a DNS query automatically by the modem)
 - Server Port to be contacted
- 4- Start the connection with the server with a proprietary "internet dial" command.

When the device receives the CONNECT indication, then it's directly exchanging data with the remote host server Application Layer.



2.4 Example 1 - HTTP client application

Let's suppose we want to connect our embedded device to an HTTP server and retrieve an HTML page using the EASY GPRS feature.

Initial data:

Server to be contacted: www.roundsolutions.com

Application Layer Protocol: HTTP1.0 (RFC1945)

Page to be retrieved: homepage of server

GPRS settings:

APN: internet.gprs

IP of GPRS device: dynamically assigned by the network

DNS: assigned by the network

USERID: Ben Hoelke

PASSWORD: EASY GPRS

Checking on the RFC990 the HTTP service we can find that the port 80 is dedicated for HTTP service, therefore our HTTP server will be waiting for incoming connections on that port and we will fix the EASY GPRS port to be contacted on the remote server exactly to 80.

Second thing we have to discover is whether the transport protocol has to be TCP or UDP; on the RFC1945 we can read that the HTTP Application layer protocol is meant to be on top of TCP/IP protocol, therefore the transport protocol choice will fall on TCP.

Now we have all the information needed to configure our system.

With our microcontroller we issue to the GM862-GPRS the following AT commands:

`AT+CGDCONT = 1,"IP","internet.gprs","0.0.0.0",0,0<cr>` (1-GPRS context setting)

`AT#USERID = "Ben Hoelke"<cr>` (2-Authentication setting)

`AT#PASSW = "EASY GPRS"<cr>` (2-Authentication setting)

`AT#SKTSET= 0,80," www.roundsolutions.com"<cr>` (3-remote host setting)

For our convenience we store all these parameters with the command:

`AT#SKTSAV<cr>`

Now we can activate the GPRS connection and let the GM862-GPRS module contact the server:

`AT#SKTOP<cr>`

When we receive the CONNECT indication, then we are exchanging data with the HTTP server program on the remote host machine.

Now following the HTTP protocol we ask for the homepage by sending the following lines on the serial line:



```
GET / HTTP/1.0<cr><lf>  
Connection: keep-alive<cr><lf>  
<cr><lf>
```

As a response to our query the HTTP server will reply with the HTML code of the homepage and some debugging responses that we will see directly on the serial line:

```
HTTP/1.0 200 OK  
Date: Thu, 06 2003 10:21:58 GMT  
Server: Apache/1.3.27 (Unix)  
Last-Modified: Thu, 06 2003 10:21:58 GMT  
Content-Type: text/html  
X-Cache: MISS from proxy.telital.com  
Connection: close
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 FINAL//EN">  
<HTML>  
... here is all the HTML code of the page..  
</HTML>
```

```
DONE!  
Easy as EASY GPRS.
```



2.5 Example 2 - EMAIL sending application

Let's suppose we want to send with our embedded device an EMAIL by using a SMTP server.

Initial data:

Server to be contacted: smtp.roundsolutions.com

Application Layer Protocol: SMTP (RFC821)

Sender: "GM862-GPRS" <gm862-gprs@roundsolutions.com>

Receiver: "Receiver" <receiver@roundsolutions.com>

Subject: Email Test

Message body: this message is sent for test Easy GPRS feature. Hello World!

GPRS settings:

APN: internet.gprs

IP of GPRS device: dynamically assigned by the network

DNS: assigned by the network

USERID: Ben Hoelke

PASSWORD: EASY GPRS

Checking on the RFC990 the SMTP service we can find that the port 25 is dedicated for SMTP service, therefore our SMTP server will be waiting for incoming connections on that port and we will fix the EASY GPRS port to be contacted on the remote server exactly to 25.

Second thing we have to discover is whether the transport protocol has to be TCP or UDP; on the RFC821 we can read that the SMTP Application layer protocol is meant to be on top of TCP/IP protocol, therefore the transport protocol choice will fall on TCP.

Now we have all the information needed to configure our system.

With our microcontroller we issue to the GM862-GPRS the following AT commands:

`AT+CGDCONT = 1,"IP","internet.gprs","0.0.0.0",0,0<cr>` (1-GPRS context setting)

`AT#USERID = "Ben Hoelke"<cr>` (2-Authentication setting)

`AT#PASSW = "EASY GPRS"<cr>` (2-Authentication setting)

`AT#SKTSET= 0,25,"smtp.roundsolutions.com"<cr>` (3-remote host setting)

For our convenience we store all these parameters with the command:

`AT#SKTSAV<cr>`

Now we can activate the GPRS connection and let the GM862-GPRS module contact the server:

`AT#SKTOP<cr>`

When we receive the CONNECT indication, then we are exchanging data with the SMTP server program on the remote host machine.

Now following the SMTP protocol we proceed with the HELO presentation and mail delivery directly over the serial line (in blu the data sent by us, in violet the one received from host):

220 smtp.roundsolutions.com ESMTP ; Thu, 5 Jun 2003 14:45:11 +0200

HELO roundsolutions.com<cr><lf>

250 smtp.roundsolutions.com Hello [111.111.111.127], pleased to meet you

MAIL FROM: "GM862-GPRS"<gm862-gprs@roundsolutions.com><cr><lf>

250 2.1.0 "GM862-GPRS"<gm862-gprs@roundsolutions.com>... Sender ok

RCPT TO: "Receiver"<receiver@roundsolutions.com><cr><lf>

250 2.1.5 "fabiode@telit.net"... Recipient ok

DATA<cr><lf>

354 Enter mail, end with "." on a line by itself

From: "GM862-GPRS"<gm862-gprs@roundsolutions.com><cr><lf>

To: "Receiver"<receiver@roundsolutions.com><cr><lf>

Subject: Email Test<cr><lf>

this message is sent for testing Easy GPRS feature. Hello World!<cr><lf>

.<cr><lf>

250 2.0.0 h55CjBVI020859 Message accepted for delivery

DONE!

Easy as EASY GPRS.