SGSN integration and implementation

Papanash Om Prakash Muthuswamy
Abstract

**SGSN integration and implementation**

Papanash Om Prakash Muthuswamy

The goal of this Thesis project is to implement a prototype for Serving GPRS Support Node (SGSN) which is the central part of the General packet radio service (GPRS). SGSN is responsible for the transfer of data packets between the mobile stations covered in its geographical service area. The SGSN should be integrated into existing Mobile Arts products, open source projects and nanoBTS radio hardware to construct a minimal GPRS network that support sending and receiving Short Message Service (SMS). The prototype is tested with radio hardware and mobile phones.
Contents

1 Introduction 1
  1.1 Goal .................................................. 1
  1.2 Motivation ........................................... 1
  1.3 Overview ............................................. 2

2 Technical Background 3
  2.1 GPRS .................................................. 3
  2.2 GPRS network architecture ............................ 3
    2.2.1 The Mobile Station (MS) ........................ 3
    2.2.2 Base Transceiver Station (BTS) .................. 5
    2.2.3 Base station controller (BSC) .................... 5
    2.2.4 Mobile switching centre (MSC) ................... 5
    2.2.5 Home location register (HLR) .................... 5
    2.2.6 Short message service centre (SMSC) ............ 5
    2.2.7 Serving GPRS support node (SGSN) ............... 5
    2.2.8 Gateway GPRS support node (GGSN) .............. 6
  2.3 GPRS network protocols ............................... 6
    2.3.1 Network service layer ............................ 6
    2.3.2 BSSGP layer ..................................... 7
    2.3.3 LLC layer ....................................... 7
    2.3.4 GMM layer ....................................... 7
  2.4 Erlang/OTP ........................................... 7
  2.5 Mnesia ............................................... 7

3 Design and Implementation 9
  3.1 SGSN supervision tree .................................. 9
    3.1.1 Server ........................................... 10
    3.1.2 Child supervisor .................................. 10
    3.1.3 Children ......................................... 10
  3.2 Communication Interfaces .............................. 11
  3.3 Service procedures .................................... 11
    3.3.1 Parser and parse libraries ........................ 11
    3.3.2 Security procedures ............................... 12
    3.3.3 GPRS Attach Procedure ............................ 13
    3.3.4 GPRS Detach Procedure ............................ 14
Chapter 1

Introduction

General packet radio service (GPRS) is a packet switched mobile data service that enables second-generation (2G), third-generation (3G) and other mobile networks to transmit Internet Protocol (IP) packets to external networks like the Internet. GPRS was standardized by European Telecommunications Standards Institute (ETSI) [1] and is now maintained by the 3rd Generation Partnership Project (3GPP) [2].

The Thesis was conducted at Mobile Arts AB, a Telecommunication company situated in Stockholm, Sweden who work with Location based products and Messaging solutions. For this project, Mobile Arts provided Home location register (HLR), Short message service centre (SMSC) and a nanoBTS hardware.

The thesis project was done by Papanash OmPrakash Muthuswamy and Johan Drake and the goal was divided between the two of us. This report explains the work done by Papanash OmPrakash Muthuswamy.

1.1 Goal

Goal The goal of this Thesis project is to implement a prototype for Serving GPRS Support Node (SGSN) which is the central part of the GPRS. SGSN is responsible for the transfer of data packets between the mobile stations covered in its geographical service area. The SGSN should be integrated into existing Mobile Arts products, open source projects and a nanoBTS hardware to construct a minimal GPRS network that supports sending and receiving Short Message Service (SMS). The prototype was tested with GPRS supported mobile phones.

1.2 Motivation

The prototype of SGSN was implemented in Erlang Programming language, a functional language which was primarily developed for telecommunication applications. It is used to build massively scalable soft-real-time systems which have requirements on high availability. GPRS is one such a system and Erlang meets its essential requirements for concurrency and scalability. The implementation of SGSN uses Erlang OTP framework and the processes that perform the functionality in SGSN are developed using the behaviors of Erlang generic Servers and State Machines.
The project involved investigation of open source software OpenBSC and other Mobile Arts products like HLR and SMSC. It also involved implementing the interfaces between these products and SGSN. It also needed a good understanding to use the NanoBTS radio hardware.

1.3 Overview

In chapter 2, the background information about GPRS is explained with architecture along with a detailed explanation of all the nodes. It also covers the network protocols that are used in the GPRS network. Later it explains about Erlang, the programming language used to implement the SGSN and Mnesia database that is used by SGSN to store its information.

Chapter 3 explains the design decisions of the prototype. It shows the design of the SGSN application using an Erlang supervision tree. It later explains the communication interfaces used between various nodes and the implementation of every service procedure in SGSN. The Implementation details about Mobile Originating service, Mobile terminating service and SGSN location register and the communication interfaces between HLR, SMSC and SGSN will be explained in the report work of Johan Drake.

Chapter 4 explains how the prototype is evaluated and tested using mobile phones and radio hardware.

Chapter 5 talks about the conclusion which is the prototype and implementation of SGSN that is capable of sending an SMS from one mobile phone to another.

Chapter 6 explains the future work and how the project can be extended to provide more services using the GPRS network.
Chapter 2

Technical Background

2.1 GPRS

GPRS is a wireless communication data service that marked the transition from speech to data transmission. Unlike Global System for Mobile Communications (GSM) which uses the circuit switching, GPRS uses packet switching. Thereby, it enhanced the GSM mobile phones, allowing them to send and receive data rapidly providing the best-effort service. Following are some features provided by GPRS.

- GPRS provides data rates of 56-114 kbit/second.
- Latency and throughput depends on the number of users concurrently sharing the service.
- It enables the devices that are capable of handling packet data to exchange data directly with the Internet.
- Users can remain connected to the network even when they are not exchanging data with other users. They do not use any network resources when they are connected.
- Users are allocated with resources only when they require the resources for sending or receiving the data.
- GPRS is cheaper for users and operators compared to circuit-switched services as the communication channels are being shared, as opposed to being dedicated to only one user at a time.

2.2 GPRS network architecture

2.2.1 The Mobile Station (MS)

The mobile station includes the mobile equipment and the subscriber identity module (SIM). This mobile equipment can be a mobile telephone or any fixed terminals. The mobile phone that supports GPRS will be equipped with packet transmission capabilities.

An MS can operate in any one of the following three modes of operation

- Class-A: The MS supports both GPRS and GSM services simultaneously.
Figure 2.1: GPRS network architecture
• Class-B: The MS supports both GPRS and GSM services, but it can only operate one of the services at a time.

• Class-C: The MS only supports GPRS services.

SIM is a small chip provided by the network operator. It contains all the important data about the subscriber such as International mobile subscriber identity (IMSI) which serves as the user identifier. The SIM also has an authentication key which serves as a password for the subscriber.

2.2.2 Base Transceiver Station (BTS)
Base station subsystem(BSS) includes BTS and BSC. BTS is a hardware device that allows an MS to connect over air interface and is controlled by a BSC. It acts as a connector between the MS and the network. The BTS equipment consists of transceiver (TRX), power amplifier (PA), duplex, antenna, combiner etc.

2.2.3 Base station controller (BSC)
Base station controller mainly deals with the communication with BTS and MSC. A BSC can control several BTS at the same time. BSC is smart module in GSM that handles the handover when the mobile phone moves from BTS to BTS. It also allocates the radio channels for the MS. When mobile phone is turned on or when it moves across from one network to another it sends its mobile measurements to the network and BSC receives this information and handles it.

2.2.4 Mobile switching centre (MSC)
Mobile switching centre is one of the GSM network components which take care of the SMS, call and other services. This includes functions like ensuring the network connections, handling of location related information and maintaining the packet flow. It communicates with visiting location register (VLR), HLR and SMSC.

2.2.5 Home location register (HLR)
HLR is a network subscribers database that contains all the information about the subscribers of the network. If the network requires any information about the subscriber it is the HLRS responsibility to provide the necessary information.

2.2.6 Short message service centre (SMSC)
SMSC stores and forwards SMS messages. If the receiver is offline or unavailable due to any reason, then the SMSC stores the SMS and forwards the SMS as soon as it gets notified that the receiver is available again.

2.2.7 Serving GPRS support node (SGSN)
SGSN takes care of the delivery of data packets from and to the GPRS mobile stations that are connected to the network. It is responsible for routing and packet transfer,
mobility management services, session management functions, logical link management services, authorization, authentication, charging functions and SMS handling. The SGSN location register (SLR) stores subscriber-related information like the location information and the user profiles of all GPRS subscribers registered to the SGSN.

2.2.8 Gateway GPRS support node (GGSN)

The GGSN takes care of the interworking between GPRS data network and external packet data networks. It acts as a gateway for the GPRS network and it hides the network from other external networks. GGSN communicates to SGSN and when it receives data that is addressed to a user, it checks if the user is active and then forwards it to the SGSN that serves the mobile user. Although, if the user is not active, the data will be discarded by GGSN. It also enables the mobility of the users across the GPRS networks.

2.3 GPRS network protocols

SGSN communicates with BTS via Gb interface. The messages are in the form of binary which are encoded with various protocols at different layers.

![Protocol layers](image)

Figure 2.2: Protocol layers

2.3.1 Network service layer

This is the lowest layer above the physical layer and it provides network services for the upper protocol data units. This layer provides transmission and reception services for transporting packets between SGSN and BTS. It indicates the status about available transmission capabilities and network congestion.
2.3.2 BSSGP layer

Base Station Subsystem GPRS Protocol (BSSGP) is the next layer above the network service layer [6]. It allows the SGSN and BTS to provide radio related information in the communication messages. It also provides the node management functions for both BTS and SGSN.

2.3.3 LLC layer

Logical link control (LLC) layer is the one above the BSSGP and it provides a logical link between the MS and SGSN [7]. It makes the communication between the MS and SGSN independent from the radio interface and thus allowing different radio solutions to be used in the radio interface. It also allows more than one MS to connect to the SGSN using the physical channel. It allows the data transfer to be set with priority and supports confidentiality for the user identity. Flow control and error control can also be provided by this layer.

2.3.4 GMM layer

GPRS mobility management (GMM) layer is present above the LLC layer and it provides all the mobile management functionality to the MS at the radio interface [8]. This includes service procedures like GPRS Attach, Detach and routing area updating. The procedures are explained along with their implementation in the next chapter.

2.4 Erlang/OTP

Erlang [9] is a functional programming language mainly used to build massively scalable soft real-time systems which have requirements on high availability. It is a concurrent language with single assignment, strict evaluation, and dynamic typing. It enables concurrency by using message passing for inter-process communication instead of using shared variables.

It was designed by Ericsson to support fault-tolerant and distributed applications which was mainly required in telecommunication applications. It was later released as an open source Programming Language. It is primarily used to develop applications for telecoms, e-commerce, instant messaging and banking.

Erlang comes bundled with the Open telecom platform (OTP), which includes a bunch of libraries and design principles. OTP acts as a middle-ware to develop small and large systems in Erlang. It also includes a distributed database called Mnesia and a lot of other applications to interface with other programming languages.

2.5 Mnesia

Mnesia [10], [11] is a distributed database management system developed by Ericsson along with Erlang/OTP. It was developed in Erlang for soft real time and high availability applications such as Telecommunication applications. The query language is also written in Erlang. Erlang/OTP comes with a library to interface with Mnesia. The tables are
stored in the form of records that use Erlang data structures. It uses the same address space for code and the table data which increases the efficiency of Mnesia.
Chapter 3

Design and Implementation

This section explains our design and implementation of the SGSN. SGSN is implemented as an Erlang application which uses an Erlang supervision tree. SGSN stores the information in SLR database which is implemented using Mnesia. Section 3.1 explains the design of the supervision tree of SGSN application. In section 3.2, the communication interface between SGSN and BSC is explained. Section 3.3 explains the implementation of some of the procedures that implement the functionality of SGSN. The implementation of these procedures is explained with a sequence diagram that explains the communication flow between the nodes and services.

3.1 SGSN supervision tree

![SGSN supervision tree](image-url)
The SGSN node is implemented as an Erlang application with a supervisor and a number of servers and other Erlang processes. As the application is started, it starts the SGSN supervisor which starts the supervisors of all the services. The services have two kinds of structures. They can either have one process or a set of dynamic child processes to implement the service. In the later case, there is server and a child supervisor which supervises the set of dynamic children. This is explained in the following diagram.

3.1.1 Server
The servers are the process that handles the requests for SGSN and they contain the service application programming interface (API). It is implemented using gen server behavior. When the server receives a request it sends a message to the child supervisor to create a dynamic child to handle that request. The server also keeps track of the children using a table with child process identifier and the request identifier. This way, it can forward the related request messages to the appropriate child that handles that request.

3.1.2 Child supervisor
Child supervisor is responsible for creating the child processes dynamically when requested by the server. It supervises the children.

3.1.3 Children
The child processes are implemented using Erlang gen fsm behavior. One Child process is created to handle every specific request. Once it receives the request, it performs the service, which might include some communication with external nodes. Once the service is completed, the process is killed. It also dies in the case of timeout where it does not
get a reply from the external node during its service. Before it dies, it sends a message to the server so that the server can update its table.

3.2 Communication Interfaces

The communication between BSS and SGSN is done over the Gb interface and the messages are in the form of UDP binary messages. These messages are sent and received by the parser block of SGSN which is explained in the next section.

3.3 Service procedures

3.3.1 Parser and parse libraries

This service block includes a parser process implemented as a server and set of parse libraries. It is this parser server that receives and sends all the messages to and from SGSN. The messages are sent as UDP messages in the binary form. The parser has a set of libraries that decodes the messages at different layers. After the parsing is done, the parser sends the message to the respective server that can handle the message request.

![Parser control flow](image)

Figure 3.3: Parser control flow
• Parse ns
  This library decodes the messages from the mobile station on the lowest layer called
  the network service which has information about the physical layer. After decoding
  it forwards the decoded message to parse bssgp library. It is also responsible for
  encoding the message with network service header before sending it to the mobile
  station.

• Parse bssgp
  The messages from Parse ns have a BSSGP header which is decoded by this library.
  This header has information about channels that are responsible for establishing
  the virtual connection between the mobile and the SGSN. After the decoding is
  done, it forwards the message to Parse llc library. This library also adds the header
  to the messages being sent to the mobile station by SGSN.

• Parse llc
  This library encodes and decodes the message at the logical link layer being sent be-
  tween the mobile station and SGSN. It is also responsible for the logical connection
  between the mobile and the SGSN. After the message is decoded it is forwarded to
  either parse gmm or parse sms based on the kind of message.

• Parse gmm
  This library decodes the GPRS mobility management messages coming from the
  mobile station to the SGSN and encodes the messages with appropriate headers
  that go in the opposite direction.

• Parse sms
  This library decodes the Mobile originating messages coming from Mobile station
  to SGSN and encodes the Mobile terminating and acknowledgement messages going
  in the opposite direction.

3.3.2 Security procedures

Security procedure is used to authenticate the subscriber of the mobile station. In the
GPRS, it is always started and controlled by the SGSN node. Following are the security
procedures implemented in this project.

Identity Check Procedure

When a mobile station sends any request to SGSN, it uses its identifier which can either
be IMSI or P-TMSI. If the SGSN is not aware of this P-TMSI the SGSN follows the
identity check procedure and sends an identity request message to the mobile station. On
receiving this, the mobile station will send its IMSI back to SGSN which is authorized.

P-TMSI Reallocation

The P-TMSI is a temporary identifier which is used instead of IMSI for securing IMSI
in the air interface. When the Mobile station sends a request with an invalid P-TMSI to
SGSN, it starts the identity check procedure which is later followed by P-TMSI alloca-
tion. So, the SGSN allocates a new temporary identifier and sends it with the reply for
the request.

In this project, the security procedures are used in the Attach procedure. The sequence is explained in the sequence diagram of the Attach procedure.

### 3.3.3 GPRS Attach Procedure

This procedure is done by the Attach supervision tree. It includes a supervisor module that supervises the block, a server that handles the request and a set of dynamic children each one taking care of individual Attach requests. The server is implemented as Gen server behavior and the children as a finite state machine (Gen fsm). This procedure is done every time the mobile is connecting to GPRS. It is also done when the mobile station is connected to the network for the first time. This procedure uses the security procedures. Following is the sequence diagram.

![Sequence diagram for Attach procedure](image.png)

The mobile station sends a GPRS Attach request as an UDP binary message to the SGSN which is received by the parser. The parser decodes it and forwards the message
to the Attach server. This server creates dynamic children to handle every individual request. It also maintains a table called TLLI table where it stores the child process identifier (PID) and the message identifier (TLLI). This way the server can keep track of the children and the request messages that they handle.

When the Attach child gets the request, it checks the identifier of the mobile station. It authorizes the same by checking it against the SLR database. If it is not present, the identity check security procedure is performed. In the later case the security procedure is skipped.

The process which is initially in Attach state goes to update location state after the identity check procedure. Now, it sends an update location request to the HLR and goes to ins_sub_data state. The HLR authorizes the IMSI and sends insert subscriber data message with the subscriber information back to the Attach child. The child process updates the database and sends the insert subscriber ack back to the HLR and waits for the update location ack in the update_location_ack state. After the reception of this message it sends the GPRS Attach accept back to the Mobile station for which the mobile will respond with Attach complete. After receiving this Attach complete message, the child process dies after reporting to the server to update the TLLI table.

### 3.3.4 GPRS Detach Procedure

![Sequence diagram for Detach procedure](image)

This procedure includes a server process that handles all the Detach requests from the mobile stations. The mobile can initiate the Detach by turning off the GPRS. This
procedure is also performed when the mobile is turned off.

The mobile station sends the Detach request to SGSN which received and decoded by the parser server. The parser forwards this message to the Detach server which handles all the Detach requests. The request has the information about Power condition. If the power is off then the server does not respond. Otherwise it sends a Detach accept back to the mobile station through the parser which encodes the message. The Detach server also changes the state of the mobile to idle state in the SLR database.

### 3.3.5 Routing Area Update Procedure

This procedure block is done by the routing area (RA) supervision tree. It includes a supervisor module that supervises the block, a server that handles the request and a set of dynamic children each one taking care of individual routing area update requests. The server is implemented as Gen server behavior and the children as finite state machine.

In our implementation, as we used single nanoBTS as the radio tower, the mobile stations never change the routing area. Although it updates the routing area the first time it connects to the network.

![Sequence diagram for routing area update procedure](image)

The mobile station sends the routing area update message as a UDP binary message to the SGSN. The parser receives this request and parses the binary, converts it to a tuple and forwards the message to RA server. The RA server creates a dynamic child to handle this request and forwards the request to that child. The server also stores the information about the child and the request message in a TLLI table.
The child process which is initially in update request state receives this message from server. Based on the request, it updates the routing area in the database if it is new. Since the mobile always sends the Attach message first it is already authorized. After the updating it sends the update accept message back to parser, which encodes the reply into a binary message and forwards it to the mobile station. The child goes to update complete state after sending the reply and waits for the update complete message back from the mobile station. When the mobile receives the update accept, it sends back the update complete to the server in the same way. The server finds the child from the TLLI table that handles the specific request and forwards it to that child. The child process on receiving the update complete confirmation dies.
Chapter 4

Evaluation and Testing

A minimal GPRS network is established by integrating the SGSN with OpenBsc, NanoBTS, HLR and SMSC. The communication messages between the nodes are tested using a packet sniffer called Wireshark. Wireshark records all the messages that SGSN sends and receives from BTS and all other nodes. It also decodes the binary messages from each layer of the message.

We did not perform any stress/load testing as our project goal is rather a proof of concept than having a good performance. Following is the explanation of the evaluation of our goal which is shown by demonstrating how the GPRS test network enables sending an SMS from one mobile station to another.

The communication starts with BTS establishing the channels with the openBSC and SGSN. This includes communication with binary messages encoding information about protocols used in physical layer, BSSGP layer, logical link layer and the GPRS mobility management layer. The parser in the SGSN successfully decodes these messages with the help of various libraries at different stages. This is proved by the establishment of channel for communication between the SGSN node and radio hardware. Once the channel is established the mobile phone gets the authorization from the SGSN by executing procedures like GPRS Attach and Identity check procedure. Then, the mobile updates its location by following routing area update procedure. After the location is updated, a mobile phone is used to send an SMS which triggers an MO messages to the SGSN via radio hardware. SGSN sends an acknowledgement for MO message back to the mobile station through the same radio and also forwards the SMS as an MT message to the recipient mobile. This is another mobile station also connected to our GPRS test network via the NanoBTS. This complete procedure is explained using a test scenario.

Initial setup includes switching on the mobile phones and setting the name of the network provider with the name of our GPRS test network. Then the channels between the mobile phones and NanoBTS are established. NanoBTS is configured with the IP address of SGSN. Then, the servers implementing each of the services in SGSN node are started. SMSC, BSC, HLR nodes are also started.

This scenario includes each mobile sending GPRS Attach to SGSN via nanoBTS by turning on the GPRS option in the mobile phones. SGSN approves the request by sending the Attach complete message back to the mobile phones. This is done by Attach service...
Figure 4.1: Test scenario for sending and receiving SMS
which also uses authorization service in case the mobile is not authorized in the network. Later the mobile phones send routing area update messages and get accepted by SGSN which is handles by the routing area update service of SGSN. While performing these services, the SGSN contacts the HLR if it requires any information about the user.

The mobile stations can stop using GPRS services by sending a Detach message to the SGSN. The detach service in the SGSN handles it by updating the HLR database and acknowledging the mobile station by sending a reply. All the messages that are sent to SGSN are parsed by the parser. The parser decodes various protocols and sends the message to the respective service that will handle it. Parser is also involved when SGSN replies to the mobile station which encodes all the messages with the respective protocols.

When the SMS is sent from one mobile, it does it by sending an MO message to SGSN. This is taken care of by Mobile originating service which sends an acknowledgement to the sender and forwards the message to SMSC. The SMSC sends an MT message back to SGSN which is forwarded to the other mobile phone. Hence, an SMS is sent via GPRS network from one mobile phone to another.
Chapter 5

Conclusion

The goal of the project which is to design and implement a prototype for SGSN using Erlang programming language was achieved. The implementation included the necessary functionality to support sending and receiving of SMS. Integrating the SGSN with GPRS nodes like OpenBsc, HLR, SMSC and the radio hardware NanoBTS, a minimal GPRS test network capable of sending SMS was constructed.

This was demonstrated by sending an SMS from one mobile phone to another mobile phone through our GPRS test network. The successful delivery of the SMS in the receiver mobile proved that both the mobile phones are attached to the GPRS network. This involved a number of procedures and authorization of mobile of phones by the SGSN. It also proved that the SGSN successfully received and forwarded the SMS message from the sender mobile to the receiver mobile after analyzing and decoding the communication messages from the mobile phones.

It was very interesting to implement the SGSN completely using Erlang programming language. SGSN is a soft real-time system which takes care of the delivery of data packets from and to the GPRS mobile station. As the services provided are critical in the GPRS network, it also requires fault tolerance, high scalability and availability. Erlang, a robust and concurrent language is designed for the same purpose; serves the needs of SGSN perfectly. The concurrency property was exploited very well in the implementation of various GPRS services. For instance, with every SMS being sent across the network via SGSN, the work is delegated to two processes; one receiving it from the sender and another forwarding it to the receiver. This way, the numbers of processes taking care of various services are independent of each other and are organized into supervision trees being supervised by their supervisors.

This thesis project work contributed to us in several ways. It involved extensive study of a number of 3GPP documents about GPRS and network protocols used in GPRS. It also required a good knowledge about several other network protocols that are concerned with mobile technology. The implementation language being Erlang required good skills in functional programming language. It was challenging to implement in Erlang using OTP framework and behaviors. We learnt to use the Mnesia database and protocol sniffing tools like Wireshark. The project also needed some understanding of the openBSC application which was implemented in C and using the radio hardware nanoBTS.
The SGSN prototype can be integrated with other GPRS nodes to construct a test GPRS network in Mobile Arts. They can use it for testing SMS related applications on this GPRS network.
Chapter 6

Future Work

This project can be extended by implementing Gateway GPRS Support Node (GGSN) which is another main component of GPRS and thus constructing a complete GPRS core network. GGSN acts as a gateway for GPRS network when it communicates with external networks. GGSN is also responsible for charging and authentication functions. It communicates with SGSN through an IP based interface using a protocol called GPRS tunneling protocol. It is the GGSN that allows mobility to the mobile station. This is done by establishing a tunnel with MS through SGSN that is currently serving the MS. Another future work could be encrypting the SMS when sent over the air interface in the GPRS network.
Bibliography


[5] 3GPP General Packet Radio Service (GPRS); Base Station System (BSS) - Serving GPRS Support Node (SGSN) Interface; Network Service (3GPP TS 08.16 version 8.0.1 Release 1999).

[6] 3GPP General Packet Radio Service (GPRS); Base Station System (BSS) - Serving GPRS Support Node (SGSN); BSS GPRS Protocol. (3GPP TS 08.18 version 8.12.0 Release 1999).

[7] 3GPP General Packet Radio Service (GPRS); Mobile Station - Serving GPRS Support Node (MS-SGSN). Logical Link Control (LLC) layer specification. (3GPP TS 04.64 version 8.7.0 Release 1999).


