University of Crete
School of Sciences and Engineering
Computer Science Department

MONITORING AND MEASUREMENT OF GSM MOBILE TELEPHONY SIGNALS

by

CHARITON D. MELISSARIS

Master's Thesis

Heraklion, July 2005

University of Crete
School of Sciences and Engineering
Computer Science Department

MONITORING AND MEASUREMENT OF GSM MOBILE TELEPHONY SIGNALS

by

CHARITON D. MELISSARIS

A thesis submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

Author:

_____

Chariton Melissaris, Computer Science Department

Supervisory
Committee:

_____

Apostolos Traganitis, Professor, Supervisor

_____

Vassilis Siris, Assistant Professor, Member

_____

Panagiotis Tsakalides, Associate Professor, Member

Approved by:

_____

Dimitris Plexousakis, Associate Professor
Chairman of the Graduate Studies Committee

Heraklion, July 2005

Abstract

The GSM standard is the most widely used cellular technology. It has been designed to be a secure digital mobile telecommunication system with strong subscriber authentication and over-the-air transmission encryption. Currently it is supposed to be one of the most secure systems for mobile communications. However, the study of the security mechanism proves that GSM systems suffer from critical errors, enabling an attacker to go through the security model and perform an interception or a phone cloning.

The objective of this thesis is the design and development of a system for *Monitoring and Measurement of GSM Mobile Telephony Signals*, taking advantage of the easy access to the physical interface – air-interface as well as exploiting recently discovered vulnerabilities of such a system.

For achieving this goal we conducted a study of the GSM architecture and specially the *air-interface* and set the requirements of such a system. We implemented the basic set of the GSM protocol stack, spanning from *source coding* and *channel coding* to *ciphering* and *accessing the physical media – the air-interface*, developing the fundamental software and using special hardware modules. Furthermore, we made an extended study of the GSM security model and mechanism and presented a variety of possible interception attacks exploiting system vulnerabilities, using the implemented architecture.

Finally, we evaluated this architecture emphasizing on the most important constraints that make such a system difficult to built as well as to be used on existent networks. Besides, we proposed further optimizations and possible extensions of this work.

Supervisor: Apostolos Traganitis
Professor

Περίληψη

Το GSM είναι η πιο διαδεδομένη κυψελοειδής τεχνολογία ασύρματης προσωπικής επικοινωνίας. Σχεδιάστηκε με βάση την ψηφιακή τεχνολογία παρέχοντας υψηλή ασφάλεια τόσο στην πιστοποίηση των συνδρομητών όσο και στην κρυπτογράφηση των δεδομένων που μεταδίδονται στο ασύρματο κανάλι. Παρόλο το γεγονός ότι θεωρείται ένα από τα ασφαλέστερα συστήματα κινητής τηλεφωνίας, η μελέτη των μηχανισμών ασφάλειας αποδεικνύει ότι παρουσιάζει διάφορα μειονεκτήματα, επιτρέποντας σε έναν επιτιθέμενο να παρακάμψει το μοντέλο ασφάλειας, παρέχοντας του την δυνατότητα να υποκλέψει δεδομένα ή ακόμα και να δημιουργήσει ένα κλώνο τηλεφώνου προσποιούμενος την ταυτότητα ενός άλλου χρήστη.

Στόχος της παρούσας εργασίας είναι η σχεδίαση και ανάπτυξη ενός συστήματος για την *Παρακολούθηση και Μέτρηση Σημάτων Κινητής Τηλεφωνίας*, που αξιοποιεί την ευκολία πρόσβασης στο ασύρματο κανάλι, μεταξύ ενός κινητού και ενός σταθμού βάσης, εκμεταλλευόμενο τα κενά ασφάλειας του GSM.

Για την επίτευξη του στόχου αυτού εξετάσαμε σε βάθος την τεχνολογία και την αρχιτεκτονική του GSM, επικεντρώνοντας το ενδιαφέρον στα χαρακτηριστικά της ασύρματης διεπαφής, θέτοντας τις απαιτήσεις ενός τέτοιου συστήματος. Υλοποιήσαμε το κυρίως πρωτόκολλο GSM, καλύπτοντας τα επίπεδα από την κωδικοποίηση πηγής και καναλιού έως την κρυπτογράφηση και την πρόσβαση στο φυσικό μέσο, αναπτύσσοντας το απαραίτητο λογισμικό και χρησιμοποιώντας εξειδικευμένο υλικό (hardware). Επιπροσθέτως, κάναμε εκτενή αναφορά στο μοντέλο και τους μηχανισμούς ασφάλειας και παρουσιάσαμε πιθανά σενάρια επίθεσης, τα οποία μπορούν να εφαρμοστούν εκμεταλλευόμενα την ύπαρξη τρωτών σημείων.

Τέλος, κάναμε μια αξιολόγηση της αρχιτεκτονικής δίνοντας έμφαση στους σημαντικότερους περιορισμούς που καθιστούν δύσκολη τόσο την κατασκευή της,

όσο και την λειτουργία της στα σημερινά δίκτυα κινητής τηλεφωνίας. Επιπλέον, προτείναμε περαιτέρω βελτιώσεις και πιθανές επεκτάσεις της εργασίας αυτής.

Επόπτης: Απόστολος Τραγανίτης
Καθηγητής

To my parents Konstantina and Dimitrios, and my brothers Theofilos and Vassilis for their provision and support in all aspects of my life

Acknowledgements

This work is the outcome of my two-year M.Sc. study at Computer Science Department of University of Crete along with the cooperation of Institute of Computer Science (ICS) of Foundation for Research and Technology of Hellas (FORTH) from March 2002 to June 2003.

I would like to thank the following people for their help and support during this work and my academic years in graduate and postgraduate studies. First of all, I am foremost grateful to my supervisor, Professor Apostolos Traganitis, for always being available, his support during my studies, for his invaluable advices and guidance throughout this work and all opportunities I have been given from.

I would also like to thank the members of the Networks and Telecommunication Laboratory of ICS-FORTH for their support, their collaborative spirit and the harmonic and enjoyable coexistence they provided.

I am also grateful to the Computer Science Department of University of Crete and the Institute of Computer Science, Foundation for Research and Technology (ICS-FORTH) for their support in terms of a graduate fellowship and above all for providing an academic and research environment.

During my studies, lot of people help and support me and I would also like to thank them. I would like to thank Maria Mamalaki and Christina Vaglini for the significant knowledge they provided me and trustfulness they showed at my first step in the university, Yannis Surlatzis for his cooperation and especially Alex Saloustros for his support to many aspects of my life, Evangelos Antoniadis for helping and giving me the opportunity to collaborate with, professor and friend Ioannis Papaefstathiou, for consulting and supporting my efforts.

Above all, I would like to thank my friends for supporting my attempts.

Chariton Melissaris
Heraklion, June 2005

Abbreviations

AB	Access Burst
AGCH	Access Grant Channel
AuC	Authentication Center
BCCH	Broadcast Control Channel
BCH	Broadcast Channels
bps	Bits Per Second
BSC	Base Station Controller
BTS	Base Transceiver Station
BTS	Base Transceiver Station
CCCH	Common Control Channel
CDMA	Code Division Multiple Access
CM	Connection Management
DB	Dummy Burst
DCCH	Dedicated Control Channels
EIR	Equipment Identification Register
ETSI	European Telecommunications Standards Institute
FB	Frequency Correction Burst
FCCH	Frequency Correction Channel
FDMA	Frequency Division Multiple Access
FN	Frame Number
FPGA	Field programmable Gate Array
FSM	Finite States Machine
GSM	Global System for Mobiles
HLR	Home Location Register
IP	Internet Protocol
ISI	Intersymbolic Interference
ME	Mobile Equipment
MS	Mobile Station
MSC	Master Switching Centre
MSISDN	MS ISDN Number
PLMN	Public Land Mobile Network
SACCH	Slow Associate Control Channel
SB	Synchronization Burst
SCH	Synchronization Channel
SIM	Subscriber Identity Module
SMS	Short Message Service
TCP	Transfer Control Protocol
TDMA	Time Division Multiple Access
TMSI	Temporary Mobile Subscriber Identity
TRAU	Transcoding Range Adaptation Unit
Um	Air interface in GSM
VLR	Visitor Location Register

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Contents

1.1 Overview

The recent years have experienced an explosive growth of the wireless personal communications and an increase in the number of subscribers on the telecommunications networks. GSM, the most widely used technology for mobile communication increases rapidly from year to year.

GSM is a communication system that is based on digital technology and provides security mechanisms so as to ensure authentication and confidentiality of user and data security against interception. The security and the authentication mechanisms incorporated in GSM make it the most secure mobile communication standard currently available. Current challenges for security in cellular telecommunications systems include the security of conversations and signalling data from interception as well as to prevent cellular telephone fraud.

Unlike the case of a fixed phone, which offers some level of physical security due to the need of physical connection, in the case of a radio link, one may be able to passively monitor the airways by employing a receiver. This process seems to be much more realistic if we consider that GSM cryptographic algorithms and specifications have become public and that critical errors which permit interception have been found.

In this work we study the requirements of a system for monitoring and measurement of mobile telephony signal, proposing and developing a new architecture to achieve this goal. The remainder of the thesis is organized as follows:

1.2 Organization

Chapter 2 introduces an overview of GSM mobile telephony system explaining its basic architecture and main components it consist of, providing the appropriate background theory for helping the reader understand the objectives of this work and how they achieved.

In Chapter 3 we provide a more detailed analysis and background theory of Air interface of GSM, the radio link carrying the air waves of speech and signalling data.

Chapter 4 sets the hardware and software requirements of a system for passively monitoring mobile telephony signals, and comprises the proposed architecture of a prototype device. Moreover, implementation details of every component, utilized on this system, are stated presenting development procedure and technique.

Chapter 5 presents the security model of GSM, authentication and encryption, and proposes various methods for interception using the system developed. It also specifies the flaws and weakness of the set of security algorithms.

Chapter 6 concludes the results of this thesis and discusses ways to further extend the capabilities of the architecture we developed.

Finally, there is a chapter committed to relative bibliography and references that is referred throughout this thesis.

Chapter 2

Overview of GSM Telecommunication System

Contents

2.1 Some GSM History

During the early 1980s, analog cellular telephone systems were experiencing rapid growth in Europe, particularly in Scandinavia and the United Kingdom, but also in France and Germany. Each country developed its own system, which was incompatible with everyone else's in equipment and operation. This was an undesirable situation, because not only was the mobile equipment limited to operation within national boundaries, which in a unified Europe were increasingly unimportant, but there was a very limited market for each type of equipment, so economies of scale, and the subsequent savings, could not be realized.

In 1982 was started the development of a pan-European standard for digital cellular mobile radio by the *Group Special Mobile.* After the founding of ETSI (*European Telecommunications Standards Institute*), GSM group became its technical

Committee in 1989 and GSM responsibility was transferred to the new institute. Later on the name of GSM has been reinterpreted as *Global System for Mobiles,* which is today's official designation.

Global System for Mobile Communication (GSM) is standardized in Europe, but is not only a European standard. Over 200 GSM networks are operational in over 100 countries around the world. The official start of GSM networks is placed during the summer of 1992, since the number of subscribers has increased rapidly, such that, worldwide, the number of subscribers has reached **1.27 billions** globally by the end of 2004, sustaining a rate of **26 million** subscribers per month.

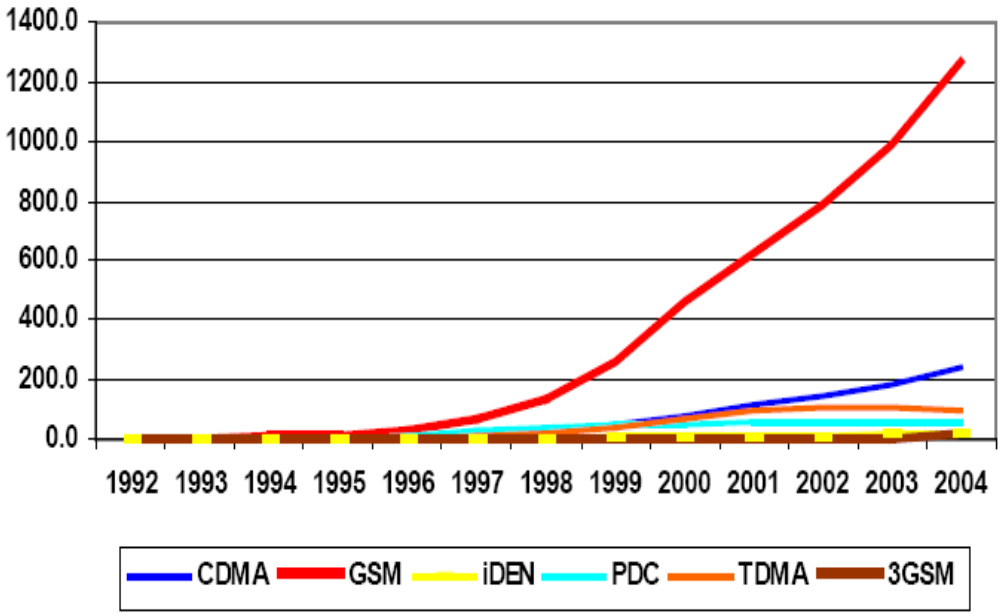


Figure 2.1-1: GSM facts and figures

2.2 An Overview of GSM Network Architecture

In this section we briefly examine the different components that together make up a GSM network. Many of these components are common to other cellular networks; however, a few are prominent to GSM. We also note that GSM sometimes uses its own terminology to describe familiar components.

Like all modern mobile networks, GSM utilizes a cellular structure as illustrated in **Figure 2.2-1**. The basic idea of a cellular network is to partition the available frequency range (physical resources), to assign only parts of that frequency spectrum to any base transceiver station, and to reduce the range of a base station in order to reuse the scarce frequencies as often as possible. One of the major goals of network planning is to reduce interference between different base stations.

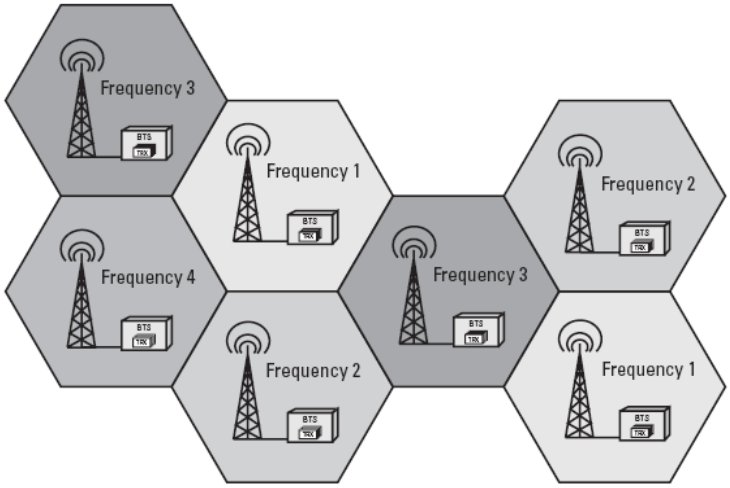


Figure 2.2-1: Cellular structure of GSM

2.3 Synopsis of the GSM Subsystems

A GSM network comprises several elements: the mobile station (MS), the subscriber identity module (SIM), the base transceiver station (BTS), the base station controller (BSC), the transcoding rate and adaptation unit (TRAU), the mobile services switching center (MSC), the home location register (HLR), the visitor location register (VLR), and the equipment identity register (EIR). Together, they form a public land mobile network (PLMN). A block diagram showing the simplified hierarchical structure of the GSM public land mobile network (PLMN) is given in **Figure 2.3-1**.

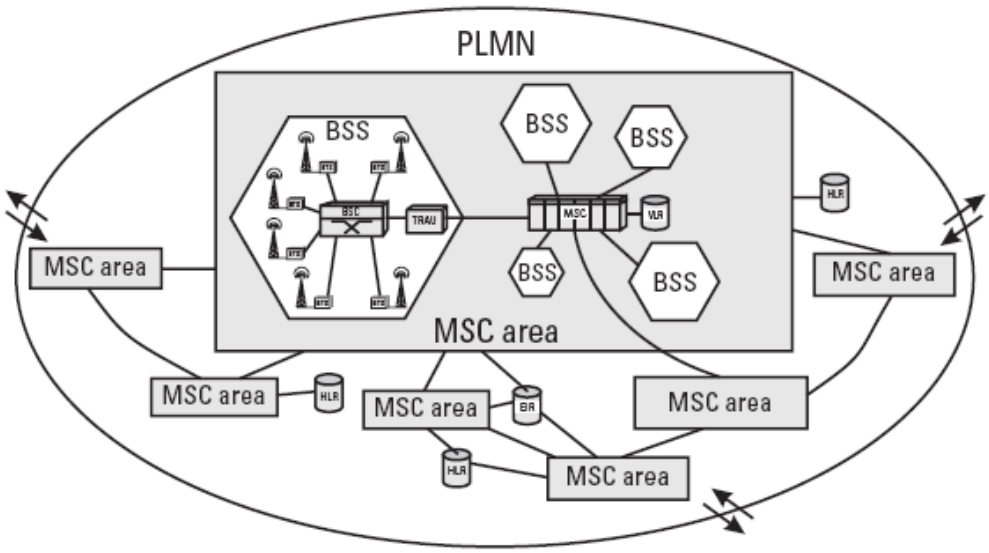


Figure 2.3-1: GSM network architecture

Figure 2.3-2 illustrates the interfaces between the main components of the GSM networks. Such interfaces are operating over microwave or leased line connections.

The *Mobile Station* communicates with the *Base Station Subsystem* over the radio interface. The BSS consists of many *Base Station Controllers* which connect to a single MSC using *Abis-interface* and the protocol that connects BSCs to MSC is called *A-interface.*

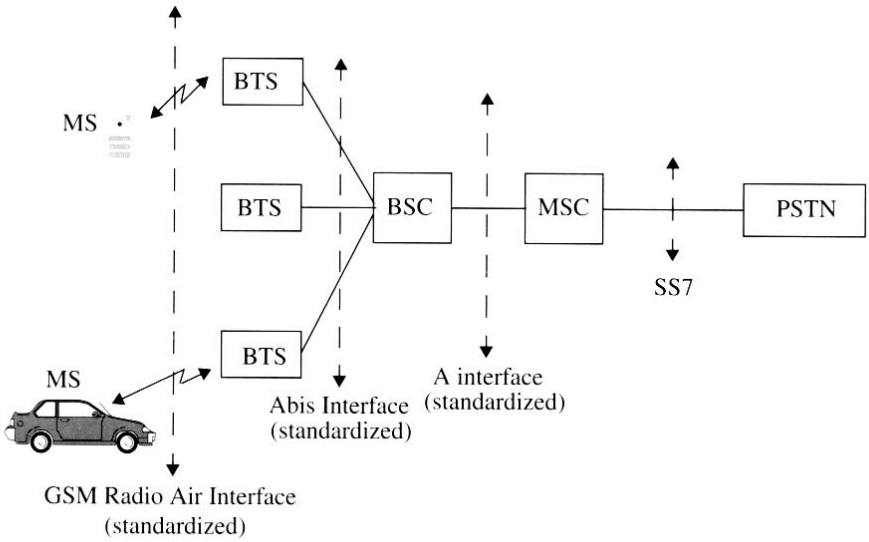


Figure 2.3-2: GSM network interfaces

Thereinafter follows a short description of the main components mentioned in previous paragraph.

2.3.1 Mobile Station

Mobile station is referred to any mobile device that uses a SIM card to gain access to the Network such as Mobile Phones, PDA's, laptops etc. GSM-PLMN contains as many MSs as possible, available in various styles and power classes.

2.3.2 Subscriber Identity Module

GSM distinguishes between the identity of the subscriber and that of the mobile equipment. The SIM determines the directory number and the calls billed to a subscriber. The SIM is a database on the user side. Physically, it consists of a chip, which the user must insert into the GSM telephone before it can be used. The SIM communicates directly with the VLR and indirectly with the HLR.

2.3.3 Base Transceiver Station

A large number of BTSs take care of the radio-related tasks and provide the connectivity between the network and the mobile station via the *Air-interface*.

2.3.4 Base Station Controller

The BTSs of an area (e.g., the size of a medium-size town) are connected to the BSC via an interface called the *Abis-interface*. The BSC takes care of all the central functions and the control of the subsystem, referred to as the base station subsystem (BSS). The BSS comprises the BSC itself and the connected BTSs.

2.3.5 Transcoding Rate and Adaptation unit

One of the most important aspects of a mobile network is the effectiveness with which it uses the available frequency resources. Effectiveness addresses how many calls can be made by using a certain bandwidth, which in turn translates into the necessity to compress data, at least over the Air-interface. In a GSM system, data compression is performed in both the MS and the TRAU. From the architecture perspective, the TRAU is part of the BSS.

2.3.6 Master Switching Centre

A large number of BSCs are connected to the MSC via the *A-interface*. The MSC is very similar to a regular digital telephone exchange and is accessed by external networks exactly the same way. The major tasks of an MSC are the routing of incoming and outgoing calls and the assignment of user channels on the A-interface.

2.3.7 Authentication Center

The Authentication Center is the central system where confidential data and keys are stored or generated. The keys serve for user authentication and authorization to respective services.

2.3.8 Home Location Register

The MSC is only one sub-center of a GSM network. Another sub-center is the HLR, a repository that stores the data of a large number of subscribers. An HLR can be regarded as a large database that administers the data of literally hundreds of thousands of subscribers. Every PLMN requires at least one HLR.

2.3.9 Visitor Location Register

The VLR was devised so that the HLR would not be overloaded with inquiries on data about its subscribers and can be considered to work like a cache. Like the HLR, a VLR contains subscriber data, but only part of the data in the HLR and only while the particular subscriber roams in the area for which the VLR is responsible. When the

subscriber moves out of the VLR area, the HLR requests removal of the data related to a subscriber from the VLR. The geographic area of the VLR consists of the total area covered by those BTSs that are related to the MSCs for which the VLR provides its services.

2.4 Mobile Station and the Subscriber Identity Module

Before we continue to the detailed description of the *GSM Air-interface*, which is a significant point for this work, it is important to understand how a mobile station works as well as the main components of this and their operation.

Mobile stations are used by mobile service subscribers, for access to the services, and consist of two major components: the *Mobile Equipment* and the *Subscriber Identification Module* (SIM).

2.4.1 Subscriber Identity Module

Except for emergency calls a mobile phone cannot be used without a SIM card. The SIM is a smart card microchip that turns mobile equipment into a *Mobile Station* and gives a kind of personalization to the subscriber. All the cryptographic algorithms as well as keys for data encryption are kept confidential in the SIM, which implements important functions for the authentication and data encryption. The major task of a SIM card is to store not only subscriber data but also other information. The most important parameters of a SIM are listed in **Table 2.4-1**.

Table 2.4-1: Data stored on a SIM

Parameter	Remarks
Administrative data	
PIN/PIN2	Personal identification number, provides access to the SIM
PUK/PUK2	PIN unblocking code
SIM service table	List of the optional functionality of the SIM
Last dialed numbers	Redial
Charging meter	Charges and time counter
Language	Determines the language
Security related data	
Algorithm A3 and A8	Required for authentication and to determine Kc
Key Ki	Known only on SIM and the HLR
Key Kc	Result of the A8, Ki and a random number (RAND)
CKSN	Ciphering key sequence number
Roaming data	
TMSI	Temporary mobile subscriber identity
Value T3212	For location update
Location update Status	
LAI	Location area identification
Network color codes of restricted PLMNs	Maximum 4 PLMNs can be stored on a SIM
NCCs of preferred	What PLMNs should the MS select

Table 2.4-1 (continued)

Parameter	Remarks
PLMN data	
NCC, mobile country code (MCC), mobile network of the home PLMN	Network identifier
Absolute radio frequency channel numbers of home PLMN	Frequencies for which PLMN is licensed

SIM provides the basis for personal mobility. The subscriber to a GSM system is not determined of the mobile equipment but only by the SIM. Because of the SIM customers can use any kind of different equipment and still be reachable under the same directory number.

2.4.2 Mobile Station Architecture

The *Mobile Station* consists of the physical equipment, such as radio transceiver, display and digital signal processors, as well as the SIM card. The basic architecture of a mobile station is shown on **Figure 2.1-1**. On the uplink direction (MS-BTS), first the sound from microphone is sampled at a specific rate; samples are fed into the speech encoder which compresses the information with a ratio of 1/10. Afterwards GSM system uses a combination of several procedures. Besides a block code, which generates parity bits for error detection, a convolutional code generates the redundancy needed for forward error correction. Furthermore, complicated interleaving of data over several blocks reduces the damage done by burst errors. Finally, the coded and interleaved blocks are enciphered, distributed across bursts, modulated and transmitted on the carrier frequency. Respectively on the downlink direction the opposite proceeding is taking place. Signals received on the radio interface on a specific frequency are demodulated and deciphered and forwarded to the source decoding module. Thereinafter, reformatting and de-interleaving processes, reconstructs and reorders the data blocks. Before source decoding of data blocks from the voice decoder a special and effective algorithm, the Viterbi one, is used for channel decoding and error correction. Finally, the decoded sound packets are played on mobile station's speaker.

All these modules and processes are described in detail on Chapter 4, since all of them are used on the system developed.

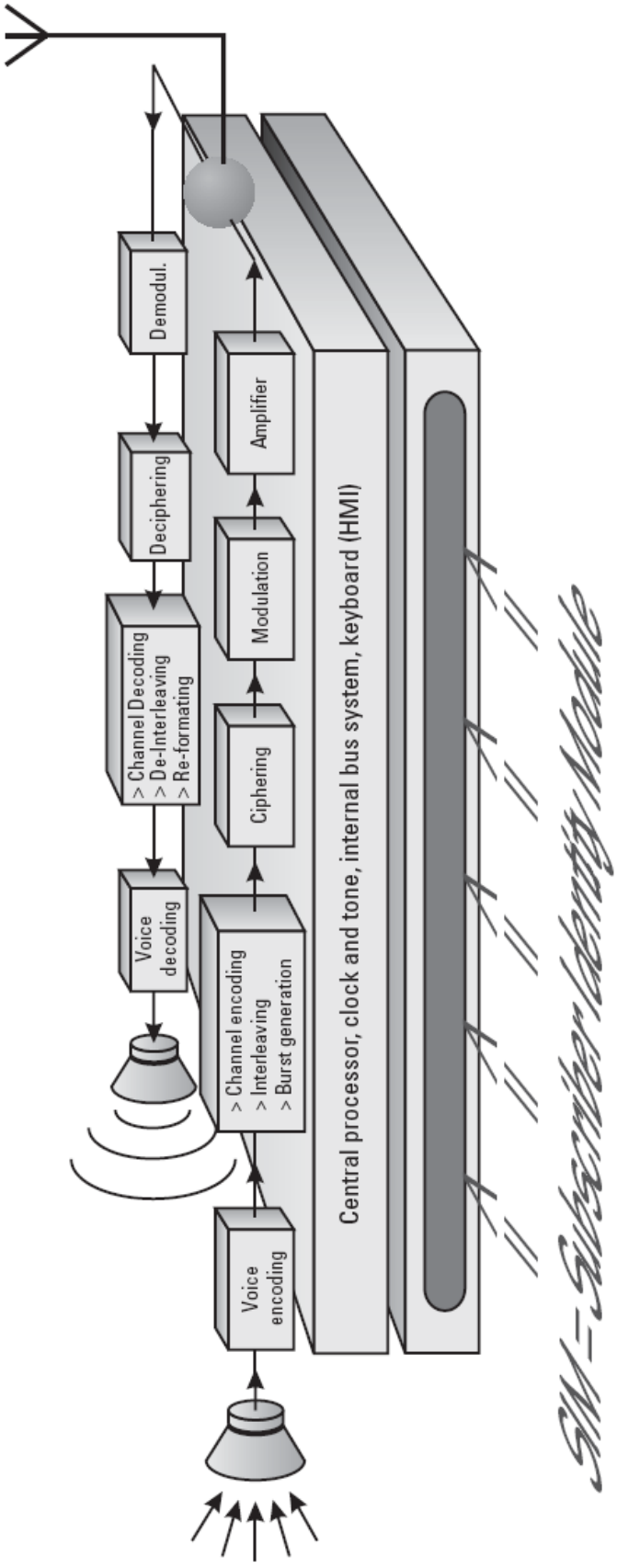


Figure 2.4-1: Block diagram of a GSM MS

Chapter 3

The Air-Interface of GSM

Contents

3.1 Structure of Air-interface

The Air-interface is the central interface of every mobile system and typically the only one to which a customer is exposed. The physical characteristics of the Air-interface are particularly important for the quality and success of any mobile standard. For this work, air interface is the main point of interest for achieving its goals. On top of the physical channels, a series of logical channels have been defined at the User-Network Interface (UNI) to perform a set of functions such as signalling, broadcast of system information, synchronization, paging, payload transport etc. In order to achieve the best bandwidth efficiency, the logical control channels are mapped onto physical channels in a certain time-multiplexed combinations.

Table 3.1-1: Classification of logical channels in GSM

Traffic Channels	Signalling Channels		
Bidirectional	**Unidirectional, Downlink**	**Unidirectional, Down- or uplink**	**Bidirectional**
Traffic Channel TCH	**Broadcast Channel BCH**	**Common Control Channel CCCH**	**Dedicated Control Channel DCCH**
Full-Rate Channel TCH/F	Broadcast Control Channel BCCH	Random Access Channel RAGH	Standalone Dedicated Control Channel SDCCH
Half-Rate Channel TCH/H	Synchronization Channel SCH	Access Grant Channel AGCH	Associated Control Channel ACCH
	Frequency Correction Channel FCH	Paging Channel PCH	Slow Associated Control Channel SACCH
			Fast Associated Control Channel FACCH

3.2 Physical Media Access Scheme

GSM utilizes a combination of frequency division multiple access (FDMA) and time division multiple access (TDMA). The difference between TDMA and FDMA is that in a TDMA system, each user sends an impulse-like signal only periodically, while a user in FDMA system sends the signal permanently.

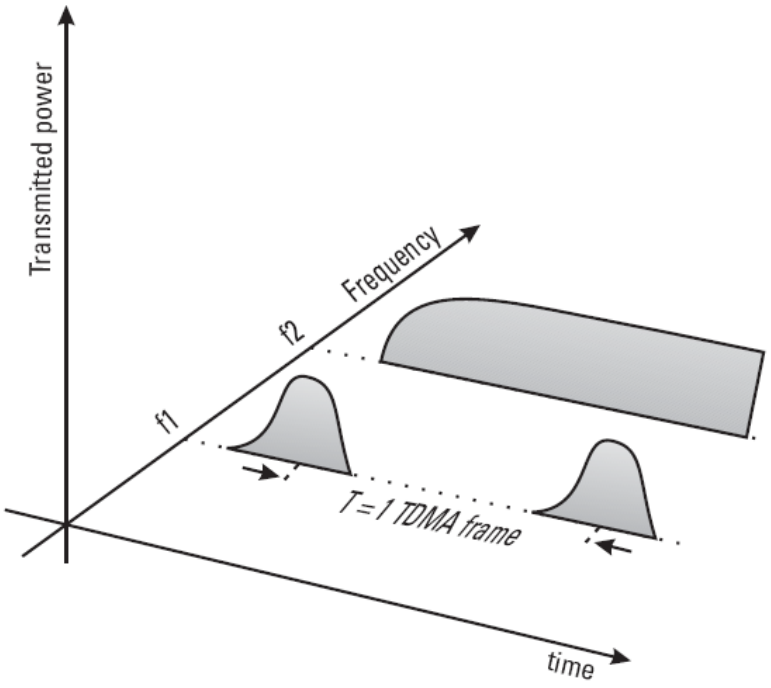


Figure 3.2-1: TDMA vs. FDMA

Figure 3.2-1 illustrates the difference between FDMA and TDMA. Frequency f1 represents a GSM frequency with an active time slot, where a signal is transmitted once per TDMA frame. That allows TDMA to serve seven other channels on the same frequency and manifest the major advantage of TDMA over FDMA (f2). That results in a two-dimensional channel structure, which is represented on **Figure 3.2-2**.

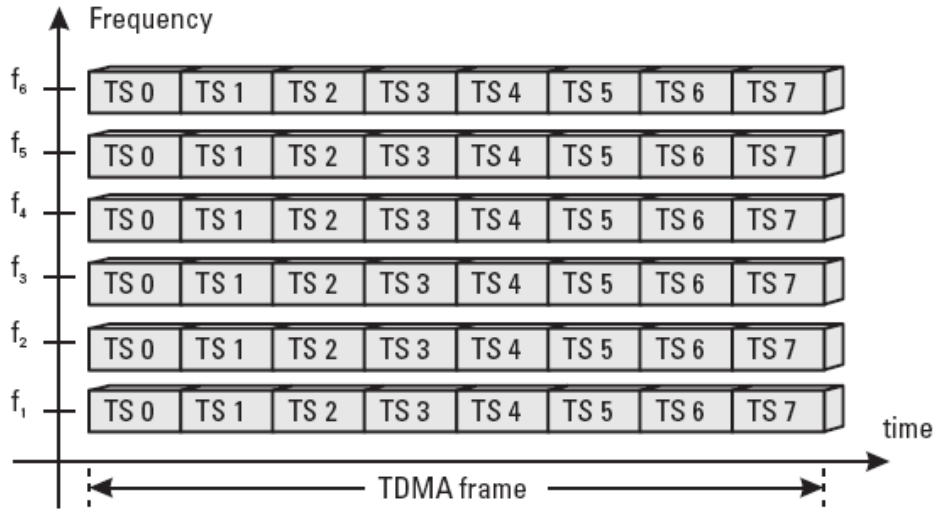


Figure 3.2-2: FDMA/TDMA structure of GSM

3.2.1 Physical versus Logical Channels

Physical channel are all the available Time Slots of a Base Transceiver Station, whereas every Time Slot corresponds to a physical channel. Two types of channels need to be distinguished, the half-rate channel and the full-rate channel. For example, a BTS with 6 carriers, as shown in **Figure 3.2-2**, has 48 (8 times 6) physical channels (in full-rate configuration). On the other hand Logical channels are piggybacked on the physical channels and are laid over the grid of physical channels, performing a specific task.

3.2.2 GSM Physical Layer Modulation

GSM uses Gaussian-Filtered Minimum Shift Keying (GMSK) as its modulation scheme which belongs to a family of continues-phase modulation procedures, which have the special advantages of narrow transmitter power spectrum with low adjacent channel interference on one hand and a constant amplitude envelope on the other hand. GMSK is a simple binary modulation scheme which may be viewed as a derivative of MSK (Minimum Shift Keying). MSK uses changes in phase to represent 0's and 1's, but unlike most other keying schemes, the pulse sent to represent a 0 or a 1, not only depends on what information is being sent, but what was previously sent. In GMSK, the side-lobe levels of the spectrum are further reduced by passing the modulation NRZ data waveform through a pre modulation Gaussian pulse shaping filter (**Figure 3.2-3**).

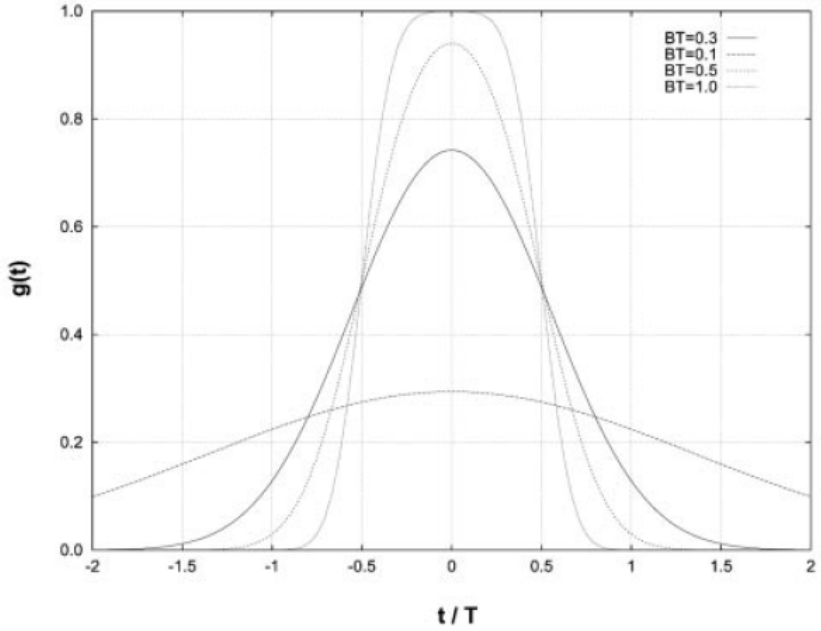


Figure 3.2-3: Impulse response of different frequency filters

The digital modulation procedure for the GSM air interface comprises several steps for the generation of a high-frequency signal from channel-coded and enciphered data blocks (**Figure 3.2-4**).

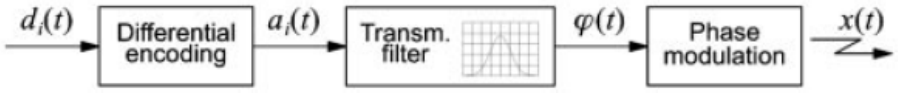


Figure 3.2-4: Steps of GSM digital modulation

Data d_i takes the value 0 or 1 and arrives at the modulator with a bit rate of 1625/6 kbits/s = 270.833 kbits/s (gross data rate) and are first differential-coded:

$$\hat{d}_i = d_i \otimes d_{i-1}, d_i \in (0,1)$$

Where $\hat{d}_i$ is the differentially encoded *i*-th data bit and $\otimes$ denotes modulo-2 addition. The output of the differential encoder represents a sequence of Dirac pulses:

$$a_i = 1 - 2\hat{d}_i$$

The above process has the effect of mapping the differential encoded data bits d_i into logical levels of ± 1 such that:

$$\hat{d}_i = 0 \rightarrow a_i = +1,$$
$$\hat{d}_i = 1 \rightarrow a_i = -1$$

This bipolar sequence is fed into the transmitter filter (linear filter with Gaussian-shaped impulse response) h(t) given by:

$$h(t) = \frac{1}{\sqrt{2\pi}\sigma T}\exp\left(\frac{-t^2}{2\sigma^2 T^2}\right)$$

where

$$\sigma = \frac{\sqrt{\ln 2}}{2\pi BT}, BT = 0.3$$

The *BT* product is the relative bandwidth of the baseband Gaussian filter and in GSM it is set to 0.3. This means that each bit is spread over three modulation symbols. The resulting ISI must be removed at the receiver using an equaliser. The impulse response, $h(t)$, and the frequency response $H(f)$ of this filter are shown in **Figure 3.2-5** and **Figure 3.2-6** respectively.

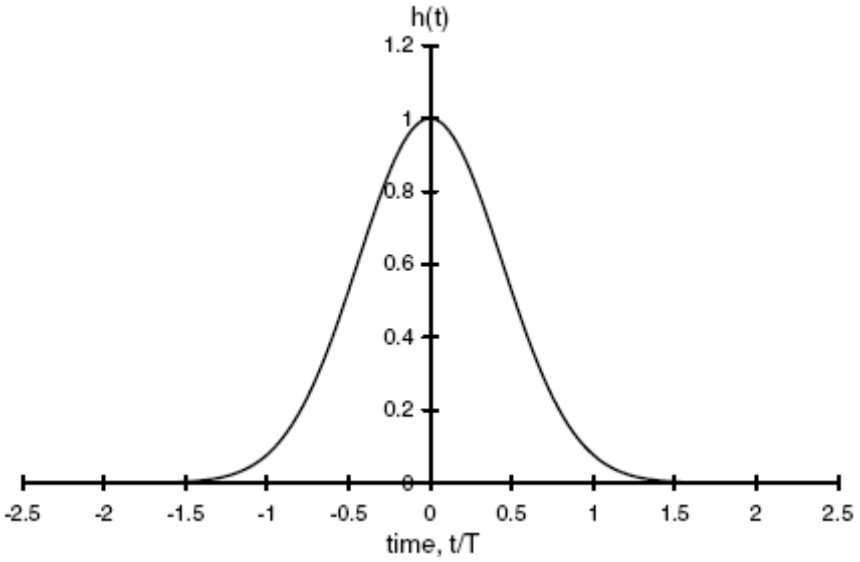


Figure 3.2-5: Impulse response

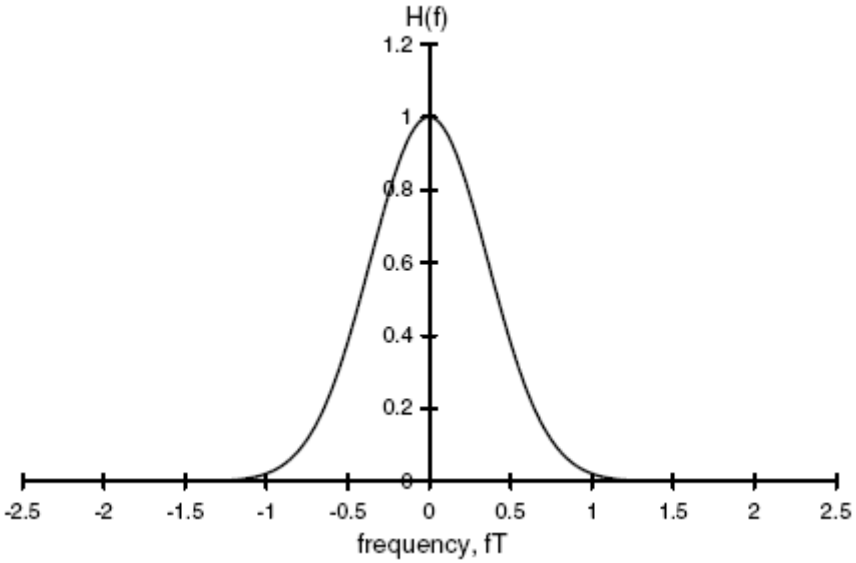


Figure 3.2-6: Frequency response

The pulse response g(t) of this filter is given by:

$$g(t) = h(t) * rect(t/T)$$

Where rect(t/T) is defined by:

$$rect(t/T) = \begin{cases} 1/T & \text{for} \quad |t| < T/2 \\ 0 & \text{for} \quad |t| \geq T/2 \end{cases}$$

The Gaussian low pass filtering has the effect of additional smoothing but also broadening the impulse response g(t). This means that on one hand the power spectrum of the signal is made narrower, but on the other hand the individual impulse response is spread across several bit durations which leads in increased intersymbol interference, as already mentioned.
The formula for the phase of a GMSK signal at a given instant relative to a differentially encoded bit stream is given by:

$$\varphi(t) = 2\pi h \int_{-\infty}^{t} \sum_{i=-\infty}^{\infty} d_i g(\tau - iT) d\tau$$

The following figure (**Figure 3.2-7**) shows the filter impact on the phase trellis diagram.

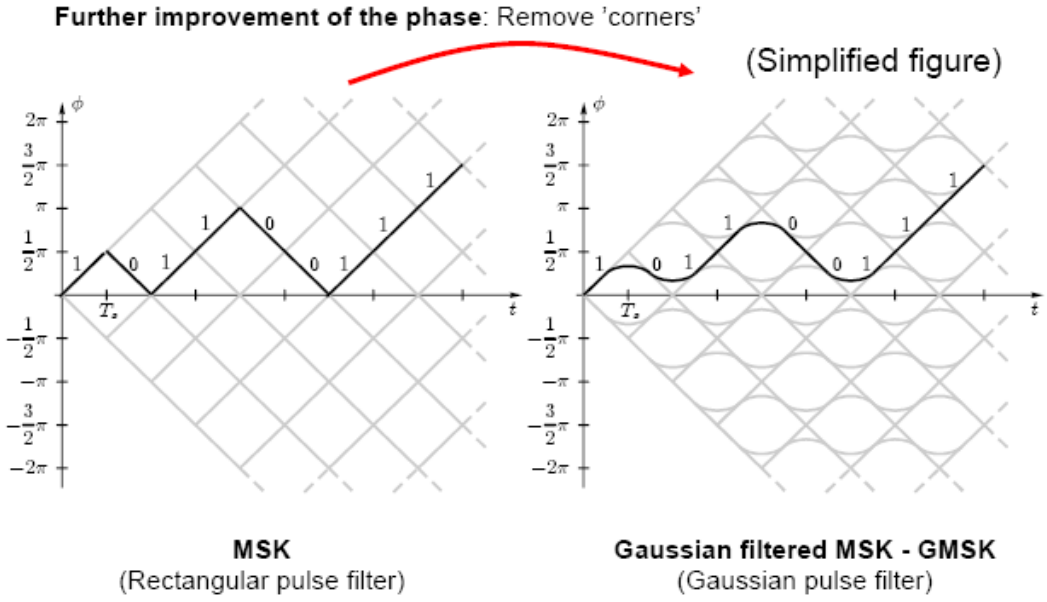


Figure 3.2-7: MSK versus GMSK

The signal constellation is drawn on the **Figure 3.2-8**, and as we can observe since the phase difference can only be π/2 the magnitude remains constant, that means that only phase errors are introduced that have to be recovered.

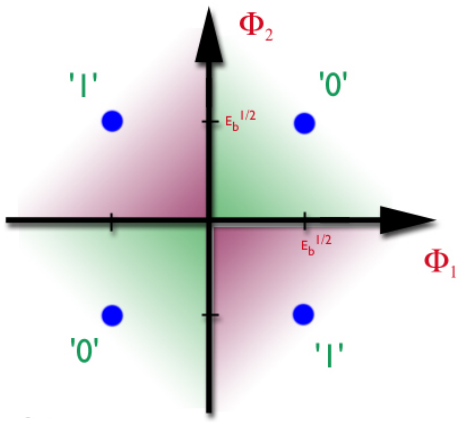


Figure 3.2-8: GMSK constellation

3.2.3 Radio Channels

GSM uses paired radio channels for simultaneously full duplex communication. Two frequency bands 45 MHz apart have been reserved for GSM operation, 890 – 915 MHz for transmission from mobile station (uplink) and 935 – 960 for transmission from base station (downlink). Each of these bands of 25 MHz is divided into 124 single carrier channels of 200 kHz width. Each channel is uniquely numbered and each pair of channels with the same number has a duplex distance of 45 MHz. **Figure 3.2-9** shows the channel organization over frequencies and time.

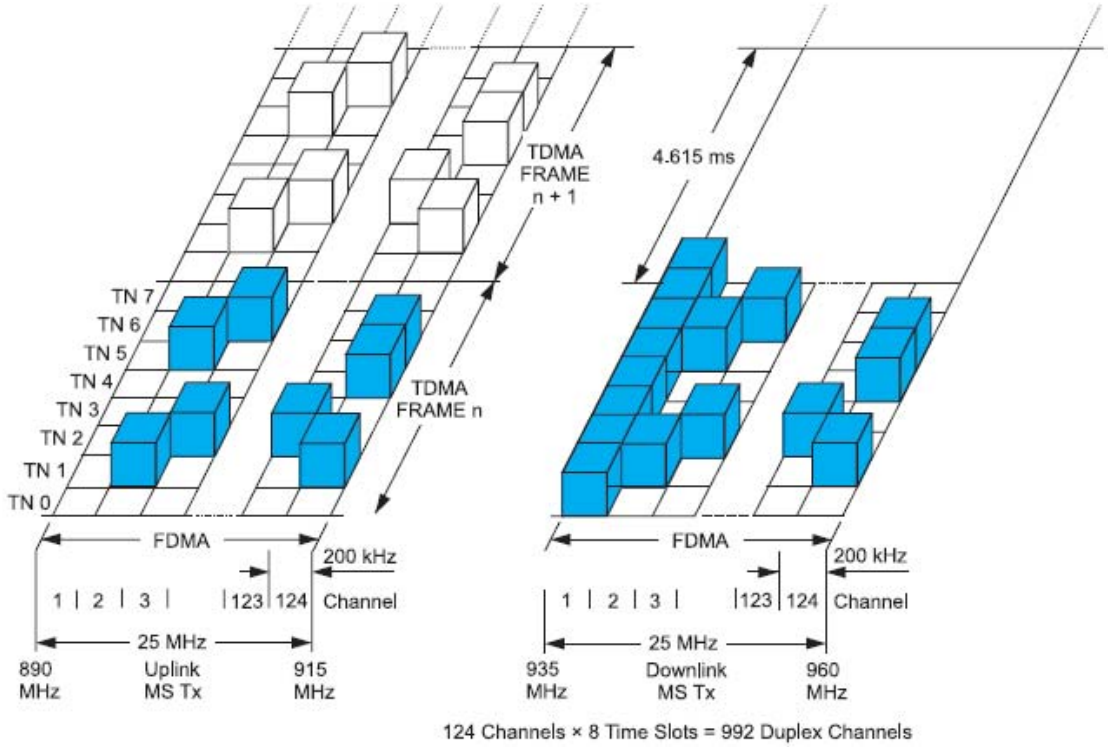


Figure 3.2-9: GSM TDMA/FDMA scheme

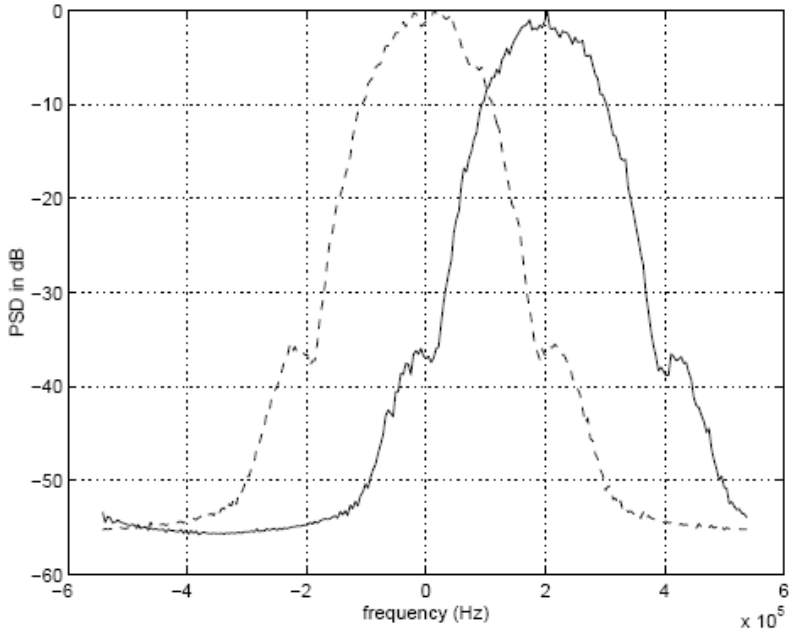


Figure 3.2-10: Spectrum for two adjacent channel GMSK signals

3.3 Frame Hierarchy and Frame Numbers

In a GSM system, every TDMA frame is assigned a fixed number, which repeats itself in a time period of 3 hours, 28 minutes, 53 seconds, and760 milliseconds. This time period is referred to as hyperframe. Multiframe and superframe are layers of hierarchy that lie between the basic TDMA frame and the hyperframe. **Figure 3.3-1** presents the various frame types, their periods, and other details, down to the level of a single burst as the smallest unit. Two variants of multiframes, with different lengths, need to be distinguished. There is the 26-multiframe, which contains 26 TDMA frames with duration of 120 ms and which carries only traffic channels and the associated control channels. The other variant is the 51-multiframe, which contains 51 TDMA frames with duration of 235.8 ms and which carries signalling data exclusively. Each superframe consists of twenty-six 51-multiframes or fifty-one 26-multiframes. This definition is purely arbitrary and does not reflect any physical constraint. The frame hierarchy is used for synchronization between BTS and MS, channel mapping, and ciphering. Every BTS permanently broadcasts the current frame number over the synchronization channel (SCH) and thereby forms an internal clock of the BTS (frame number also is used for ciphering procedures). There is no coordination between BTSs; all have an independent clock, except for synchronized BTSs. An MS can communicate with a BTS only after the MS has read the SCH data, which informs the MS about the frame number, which in turn indicates the chronologic sequence of the various control channels.

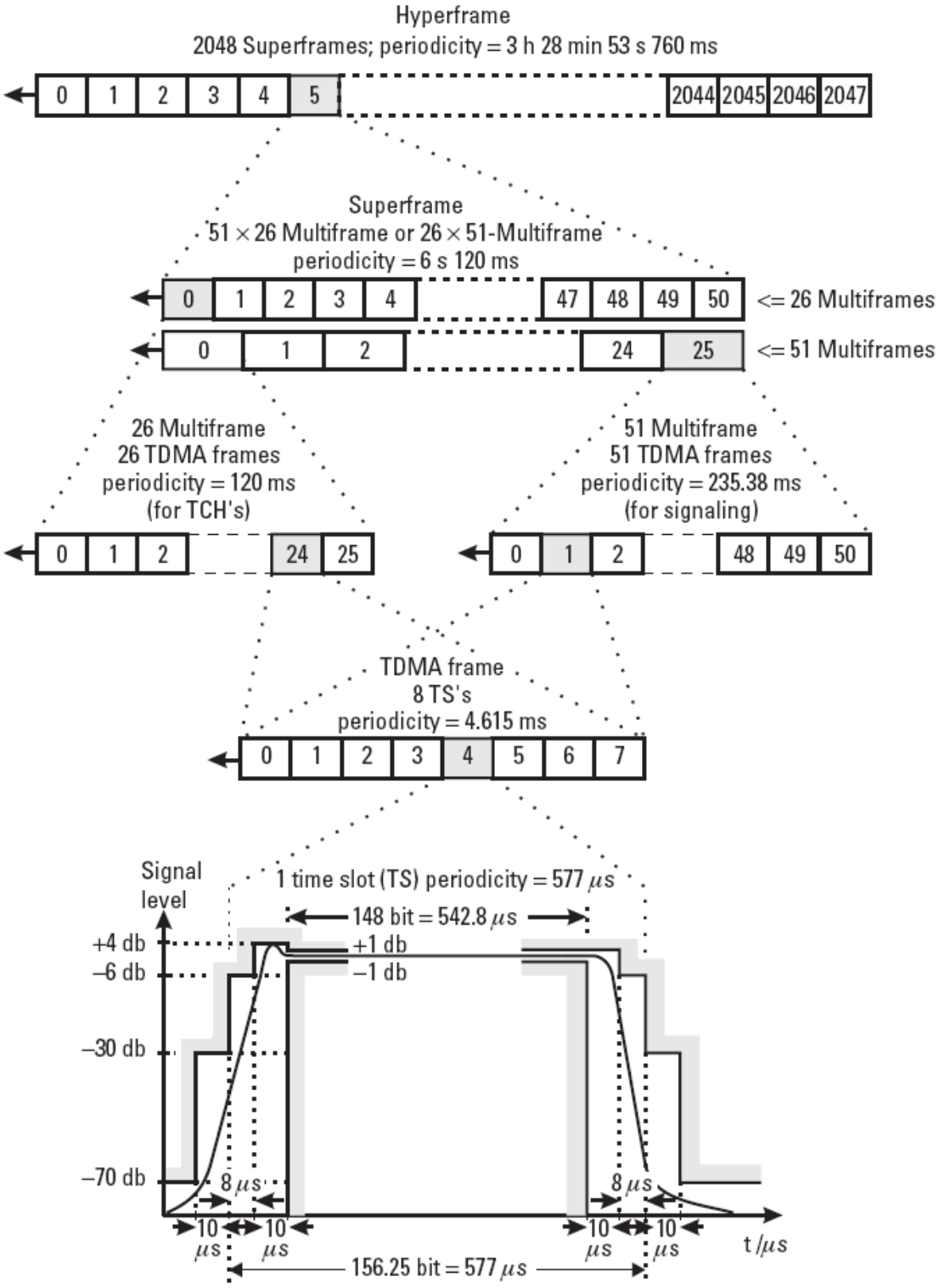


Figure 3.3-1: Frame Hierarchy in GSM

3.4 Logical Channel Configuration

On Layer 1 of the OSI Reference Model GSM defines a series of logical channels that are divided into two categories (**Table 3.1-1**), traffic (TCH) and control channels (CCH). The first category comprises the traffic channels:

- *Traffic Channel* (TCH)

Traffic channels are used to carry user payload data such as speech, fax and data. They do not carry any control information.

- *Control Channels*

Control or signaling channels are briefly explained on **Table 3.4-1**.

Table 3.4-1: Channel description

Name	Abbreviation	Task
Frequency Correction Channel	FCCH	The 'lighthouse' of a Base Station
Synchronization Channel	SCH	Base station identifier plus synchronization data (Frame number)
Broadcast Common Control Channel	BCCH	To transmit system information
Access Grant Channel	AGCH	SDCCH Channel assignment message
Paging Channel	PCH	Carries paging request message
Broadcast Control Channel	BCCH	Transmits cell broadcast messages
Standalone Dedicated Control Channel	SDCCH	Exchange of signaling information between MS and BTS when TCH is not active
Slow Associated Control Channel	SACCH	Transmission of signaling data when a connection is active
Fast Associated Control Channel	FCCH	Transmission of signaling data during a connection (used only if necessary)
Random Access Channel	RACH	Communication request from MS to BTS

Figure 3.4-1 shows in an example for an incoming call connection setup at the air interface and how the various logical channels are used in principle.

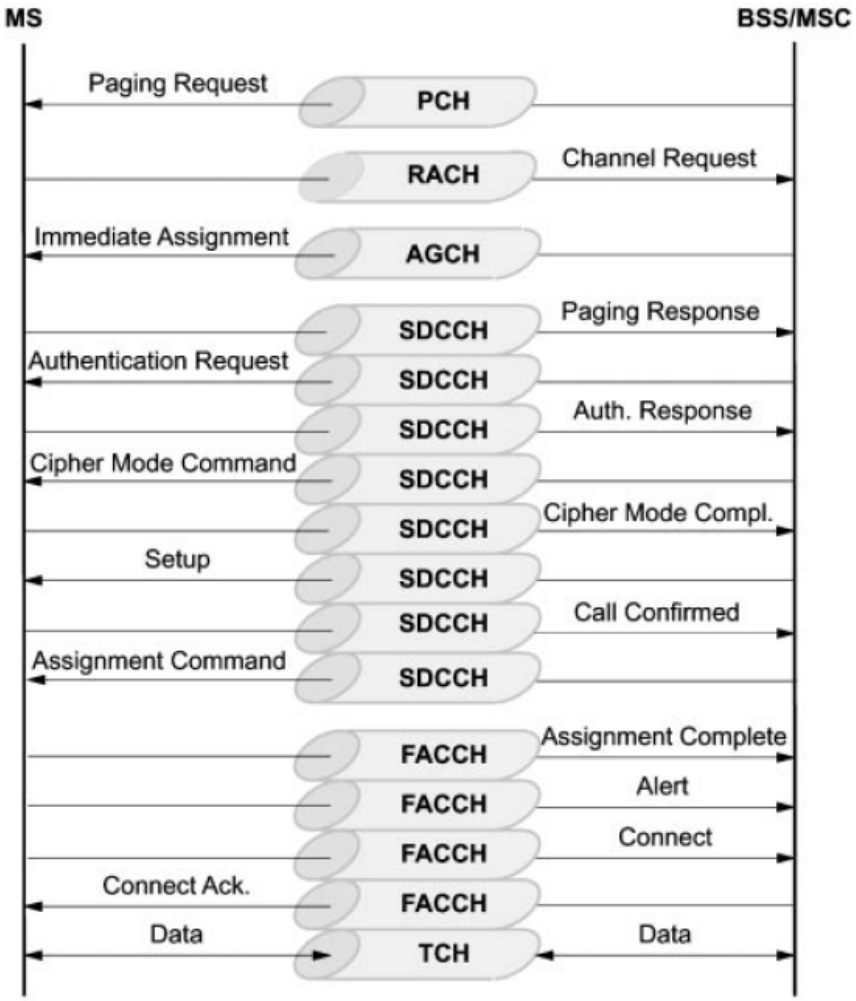


Figure 3.4-1: Logical channels and signaling

3.5 Bursts

There are five kinds of burst in GSM, each one has a duration of 156.25 bit times and last for 15/26 = 576.9 μs. A frame contains 8 Time slots which results duration of 4.613 ms. the structure of the five bursts is shown on **Figure 3.5-1**. Next paragraph shortly describes these bursts and their purpose.

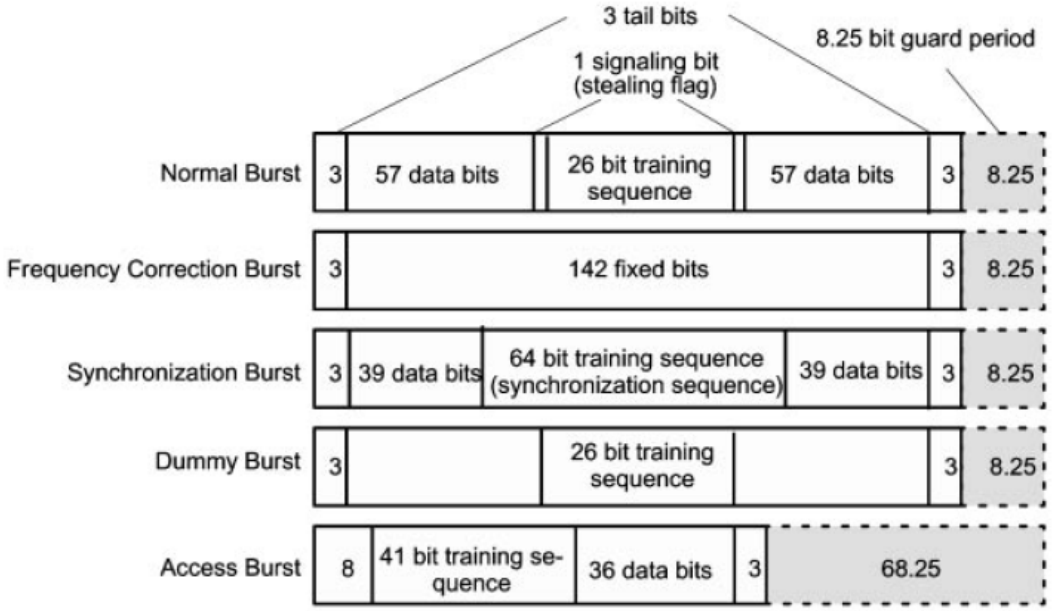


Figure 3.5-1: Bursts of the GSM TDMA procedure

3.5.1 Burst description

Each burst on GSM is separated each other by guard periods during which no bits are transmitted. Moreover, to overcome the power ramp up or ramp down time, every burst has 3 tail bits at its start and its end that are logically set to '0' (tail bits is also used in the demodulation process).

- *Normal Burst*

The normal burst is used to transmit information on traffic and control channels and contains two blocks of 57-bits each of error protected and channel coded user data. These blocks are separated by a 26-bit training sequence consisting predefined bit patterns, which are used for channel estimation. Consequently the first section of the burst must be stored before demodulation can proceed. The training sequence consists of a 16-bit sequence extended in both directions by copying the first five bits at the end of the sequence and the last five bits at the beginning. The central 16 bits are chosen to have a highly-peaked autocorrelation function, following GMSK modulation, and the repeated bits at either end ensure that the resulting channel estimate may be up to five bits wide before being corrupted by the information bits.

- *Frequency Correction Burst*

This burst is used for frequency synchronization of mobile station. It is used by FCCH channel for periodically re-synchronizing mobile with base station. All bits of this burst are set to zero broadcasting an un-modulated carrier with a

frequency shift of 1625/24 kHz = 67.708 kHz above the nominal frequency. This procedure permits the exact tuning to the carrier frequency.

- *Synchronization Burst*

The Synchronization burst is used to transmit information which allows the time-wise synchronization of a mobile station with base station. It is carried out by SCH channel and broadcasts the current *Frame Number*.

- *Dummy Burst*

For the identification of a broadcast channel a base station must transmit in all eight time slots. Dummy burst is used for dummy timeslot transmission of broadcast channel BCCH, in order to keep the frequency signal power in certain levels. This enables mobile station to perform signal power measurements of the BCCH (quality monitoring)

- *Access Burst*

Finally, access burst is used for random access to the RACH. The significant longer guard time is used to reduce the probability of collisions due to lack of synchronization

3.6 Mapping Logical onto Physical Channels

The mapping of a logical channel onto a physical channel in the frequency domain is based on the TDMA frame number FN. The various logical channels described above may be combined in one of six different ways, before being mapped onto a single physical channel. The simplest mapping is the full-rate traffic channel (TCH/F) and its SACCH. When combined these channels fit exactly into a single physical channel. We note that the mapping between the TCH and the physical channel is the same regardless of whether the TCH is used to carry speech or user data. A single physical channel will also support two half-rate traffic channels (TCH/H) and their SACCHs or eight SDCCHs and their associated SACCHs. The remaining three logical channel combinations are a little more complicated and these are explained below. The basic broadcast and common control channel combination consists of a single FCCH, SCH and BCCH on the down-link, along with a full-rate PCH and a full-rate AGCH. The up-link is entirely dedicated to the RACH, and for this reason we shall term this a full-rate RACH. This type of channel configuration is generally used in medium capacity or large capacity cells where the access capacity of a full-rate PCH, AGCH and RACH channel is justified. This control channel combination may only occur on time slot zero of a carrier. The carrier that supports these channels at a BTS is called the *BCCH carrier* and it will be unique within each cell, or sector. In smaller capacity cells, i.e. cells with a smaller number of RF carriers, the capacity of the full-rate PCH, AGCH and RACH may not be justified. For this reason, a second combination of the access channels is employed. The down-link continues to support an FCCH, SCH and BCCH; however, the rate of the down-link PCH and AGCH is reduced to around one-

third of their full rate. The extra slots that have been created as a result of this rate reduction on the down-link are used to support four SDCCHs and their associated SACCHs. The SDCCHs will also occupy a number of up-link slots and the number of timeslots allocated to the RACH on the up-link is reduced accordingly. Once again, this control channel combination may only occur on time slot zero of the BCCH carrier. The final control channel combination is defined for use in large capacity cells where the access capacity of a single PCH, AGCH and RACH is insufficient. This combination consists of a BCCH and a full-rate PCH and AGCH on the down-link and a full-rate RACH on the up-link. This channel combination may only occur on slot two, or slots two and four, or slots two, four and six of the BCCH carrier. The reason for this restriction is the timing advance mechanism. We note that each BTS must only transmit a single FCCH and SCH and, consequently, these channels are not included in the extension channel set. Each extension set contains its own BCCH for two reasons. Firstly, the BCCH contains information that applies only to the RACH occupying the same time slot within the TDMA frame, and secondly, it is easier for the MS to monitor bursts occurring on the same physical channel.

3.6.1 Possible combinations

Channel configuration is restricted by a number of constraints, for that reason a network operator has to consider the peculiarities of a service area and the frequency situation to optimize configuration. GSM ETSI standard 05.02 provides guidelines, which need to be taken into account when setting up control channels.

Following figures, **Figure 3.6-1** and **Figure 3.6-2** shows a typical configuration for downlink and uplink frequency channels respectively. The first figure illustrates an example of the downlink part of a full-rate channel configuration of FCCH/SCH + CCCH + SDCCH/4 + CBCH on TS 0, SDCCH/8 on TS 1, and TCHs on TSs 2–7.There is no SDCCH/2 on TS 0, because of the CBCH. The second figure's example shows a configuration of the uplink part of a full-rate channel configuration. RACHs can be found only on TS 0 of the designated frame numbers. The missing SACCHs on TS 0 and TS 1, in both examples, can be found in the next multiframe, which is not shown.

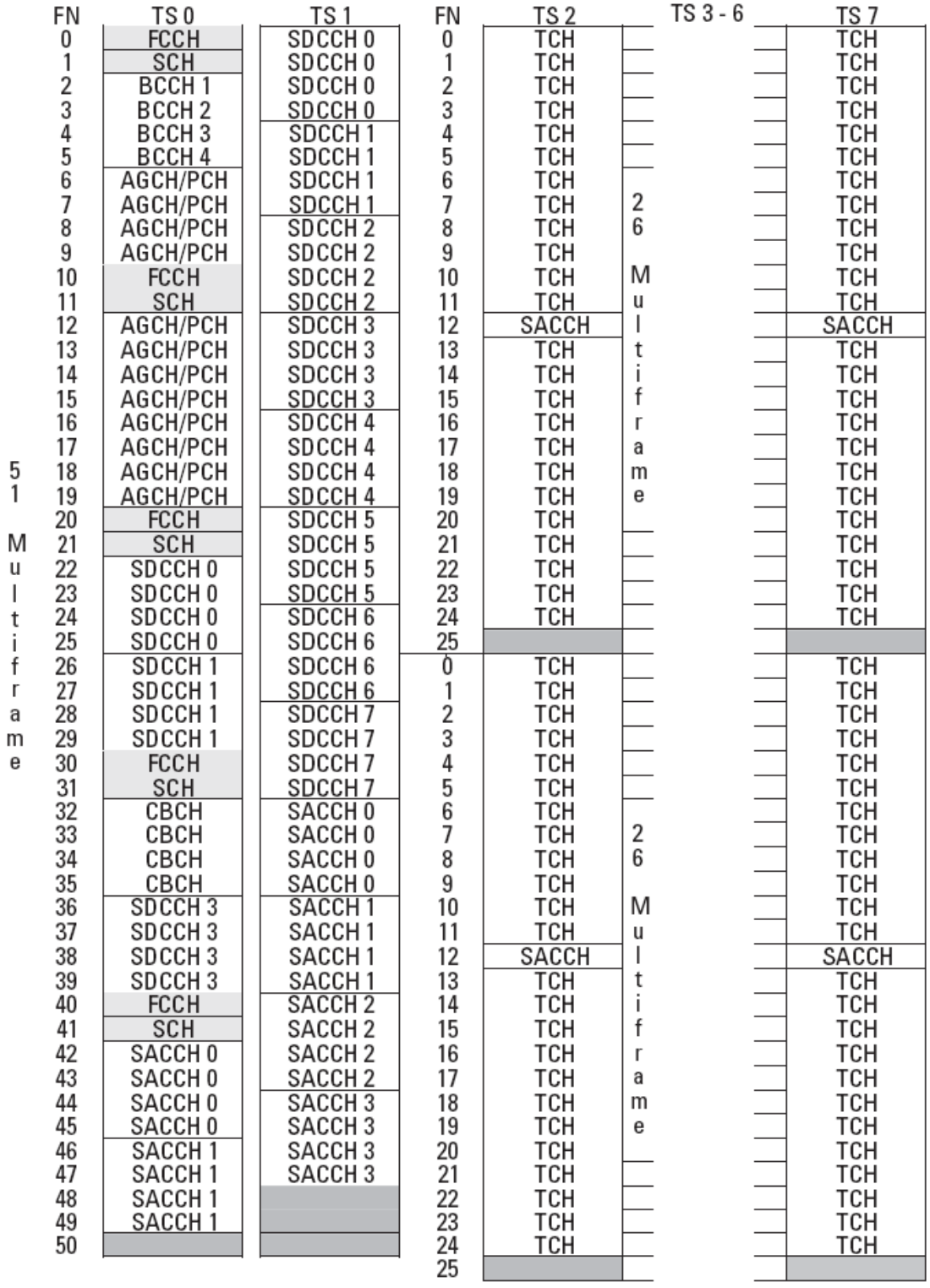


Figure 3.6-1: Example of a channel configuration for the downlink channel

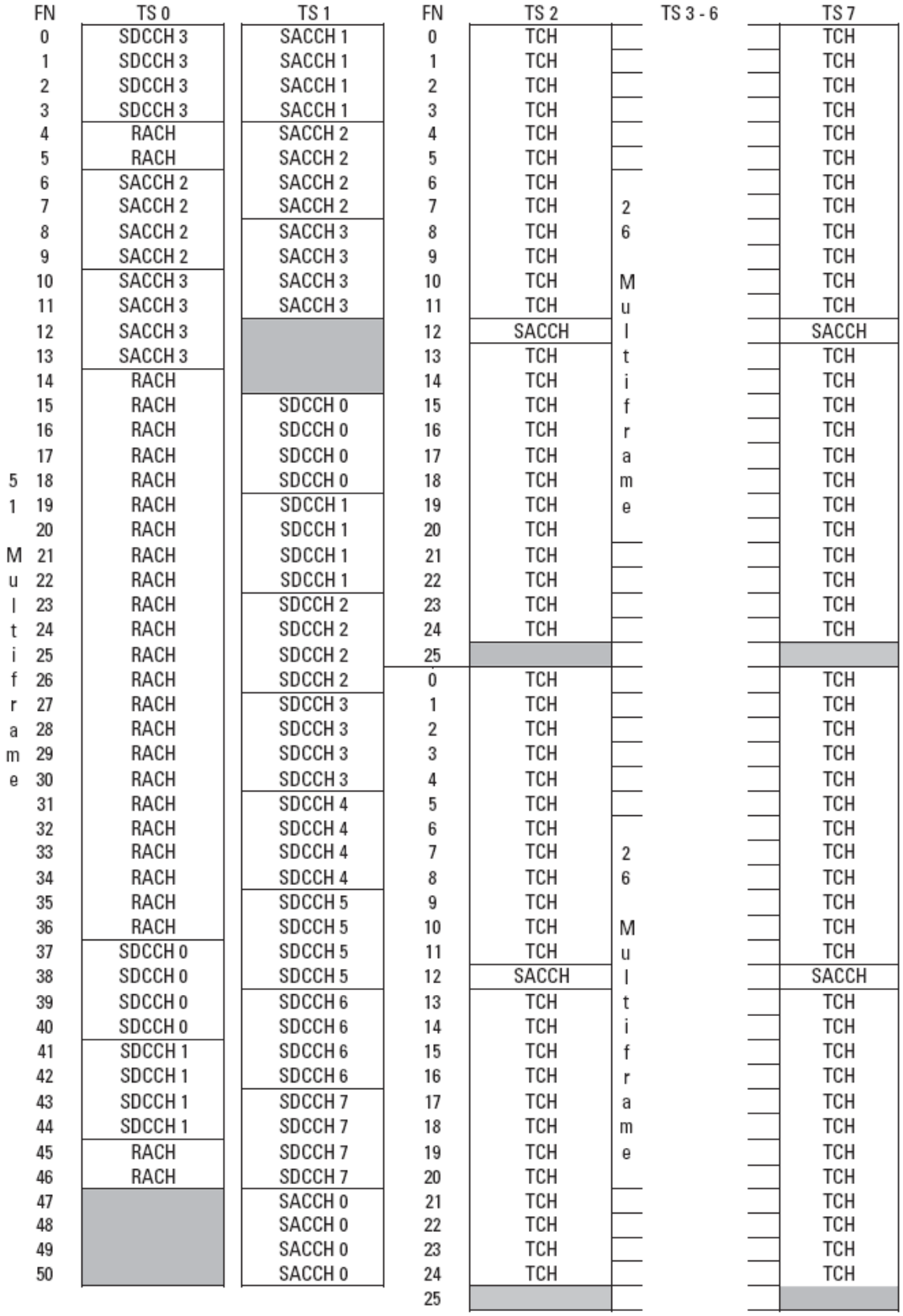


Figure 3.6-2: Example of a channel configuration for the uplink channel

Chapter 4

Architecture of GSM Monitoring System

Contents

4.1 Requirements

In previous sections we presented an overview of the main architecture of GSM telecommunication system as well as the main modules the system consists of. The reference to *Mobile Station's* architecture is very important because the core of the system architecture is based on that. Especially, we introduced a more detailed statement in the characteristics of *Air-Interface* and channels of GSM architecture that will help to better understand both the objectives of the project and system implementation issues. In this section we describe the basic requirements of a device for monitoring GSM mobile telephony signals.

Since a monitoring system is a mobile station, it is essential for reader to understand that the core of such a system is a *Mobile Telephone* similar to the one stated in **Figure 2.4-1**. The main difference is that we do not need all the functionality of a mobile phone, moreover we need to bypass, even essential for the GSM networks,

functions; i.e. one of the most important issues is the SIM card independency of the hardware. However, it is absolutely compulsory for our device to meet some other significant hardware and software requirements stated on next paragraphs.

For development reasons we expand this architecture to both uplink and downlink direction. In other words, to develop a GSM receiver module we also developed a base station (transmitter) system. The functions performed in receiver are the reverse procedures of a base station.

4.1.1 Hardware Requirements

Hardware components are all the modules that enable the access to the physical media. Respectively to a mobile phone and according its architecture, there are two important modules we need in order to receive digital raw data from the *Air-interface*:

- A Cellular band (800-1000 MHz) receiver and
- A GMSK demodulator

Moreover, there is a need to define an interface between physical media access hardware and a host running the appropriate software. Host can be a personal computer or a microcontroller daughterboard, or even a custom FPGA (Field programmable Gate Array) design. In present architecture the communication between hardware and software is based on TCP/IP interface and LAN I/O hardware that is connected to the GMSK demodulator. The above two modules, as well as LAN I/O hardware, are daughterboards connected each other, all provided by ComBlock. Modules are fully controlled and configurable by controller software running on host, based on register Read/Write commands (control communication is performed through hardware registers).Hardware requirements comprise the functionality listed in **Table 4.1-1** that follows.

Table 4.1-1: Hardware requirements

Requirement	Description
Cellular Band Receiver	
Frequency Selection	Tune capability to all GSM900 downlink frequencies (890-915 MHz) with step of 200kHz
Fast frequency tuning	Local oscillator must be able to tune on a frequency in less than 4,6 ms (Frame time)
High RF input Sensitivity	< -70 dB
Baseband filtering < 300kHz	GMSK demodulation requires BT = 0,3[1] and GSM channel spacing is 200kHz
Gain control	Gain control interface

[1] BT is the product of B: 3db bandwidth and T: bit duration

Table 4.1-1 (continued)

Requirement	Description
GMSK Digital demodulator	
GMSK	Modulation index h = 0,5
Bit Rate 270,833 kbps	GSM aggregate bit rate
Channel Spacing	200kHz channel spacing
Monitor	Frequency error and magnitude monitor
M-array	2-array FSK
Automated gain control	AGC module to self adjust received gain

The base station implementation (downlink channel) uses modules with the same characteristics and requirements as above, namely an RF transmitter on 800-1000 MHz and a GMSK modulator. Moreover, we use another separate daughterboard for converting digital to analog signals all from ComBlock [Appendix I].

4.1.2 Software Requirements

As already mentioned, hardware components are responsible of feeding a personal computer or workstation with GSM stream raw data captured from the *Air-interface* throughout a LAN interface. The workstation is in charge of running the appropriate software for transforming the received bit stream to audio information on sound device. **Figure 4.1-1** illustrates a full set of procedures and interfaces a mobile phone or a base station implements on the uplink and downlink direction respectively. Each channel has its own coding and interleaving scheme. However, the channel coding and interleaving is organized in such a way as to allow, as much as possible, a unified decoder structure.

- the information bits are coded with a systematic block code, building words of information + parity bits;
- these information + parity bits are encoded with a convolutional code, building the coded bits;
- Reordering and interleaving the coded bits, and adding a stealing flag, gives the interleaved bits.

All these operations are made block by block, the size of which depends on the channel. However, most of the channels use a block of 456 coded bits which is interleaved and mapped onto bursts in a very similar way for all of them.

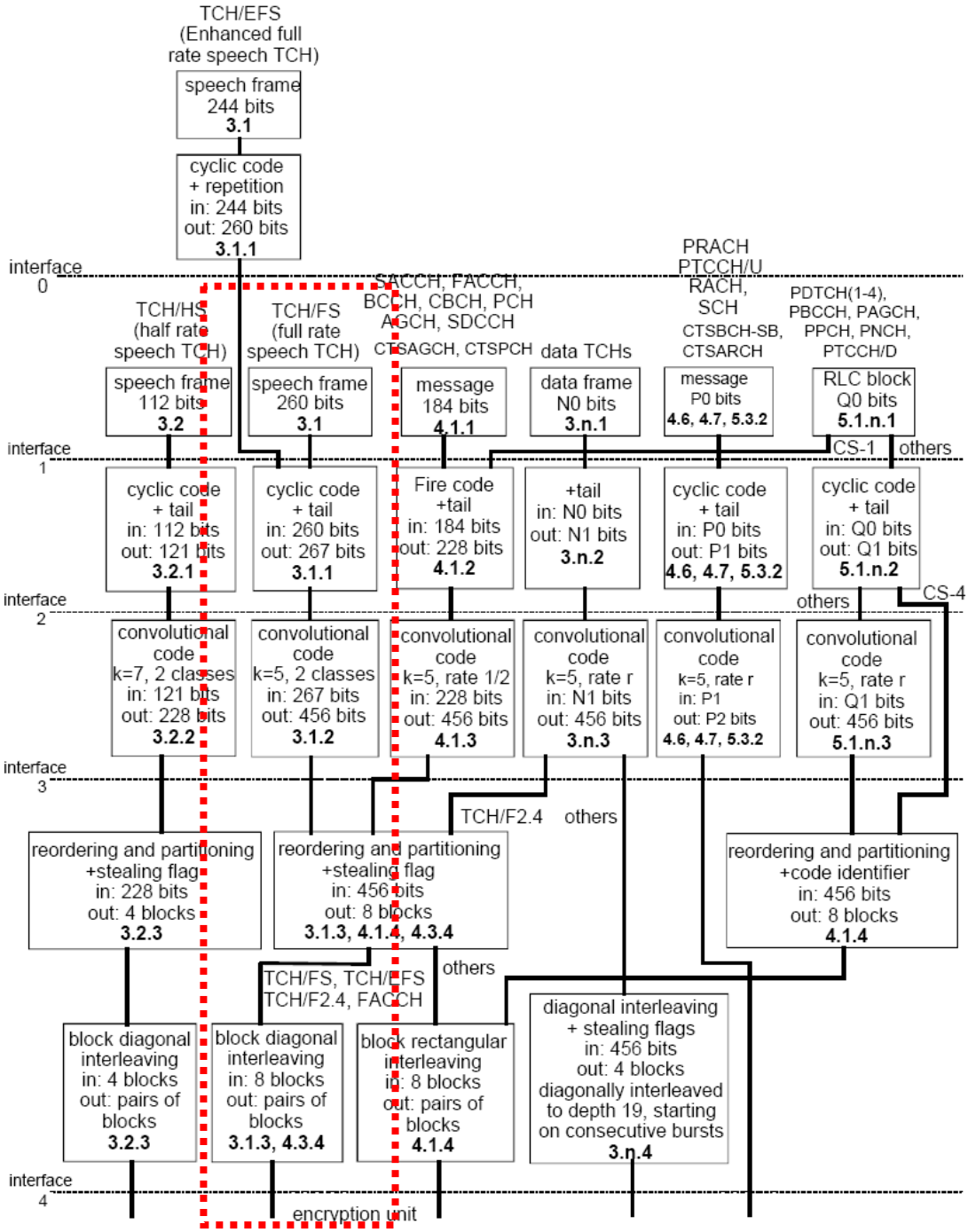


Figure 4.1-1: Channel coding and interleaving organization (by 3GPP)

In our architecture we only need to implement the set of modules in the red dotted frame shown in **Figure 4.1-1**, both for uplink and downlink, and some modules for the Synchronization channel decoding (partially implemented). Server application must be capable to perform a full decoding of a Full Rate Traffic Channel. The main difference between receiver and transmitter implementation is the convolutional decoder that implements Viterbi algorithm for decoding. Detailed information on implementation is proposed on next sections.

The following tables, **Table 4.1-2** and **Table 4.1-3**, summarize software modules and its requirements.

Table 4.1-2: Software module specifications (MS uplink)

Module	Description	Input bits	Output bits
		20ms data block	
Speech encoder	Performs RPE-LTP[2] source encoding	2080	260
CRC unit	Adds 3 CRC bits and adds 4 tail bits for resetting Convolutional encoder	260	267
Reordering	Reorders some bits	456	456
Convolutional encoder	Performs Convolutional Encoding CC(2,1,5)	267	456
Interleaver	Diagonal bit interleaving and block interleaving	456	456
Encryption unit	Block ciphering	114 (x4)	114 (x4)

Table 4.1-3: Software module specifications (MS downlink)

Module	Description	Input bits	Output bits
		20ms data block	
Decryption unit	Block deciphering	114 (x4)	114 (x4)
De-interleaver	Block de-interleaving and diagonal bit deinterleaving	456	456
Viterbi decoder	Implements Viterbi algorithm for convolutional decoding (hard decision)	456	267
Reordering	Reorders back some bits	456	456
CRC check unit	Checks the 3 CRC bits and removes CRC and tails bits	267	260
Speech decoder	Performs RPE-LTP source decoding	260	2080

[2]Speech compression algorithm *Long-Term Prediction – Linear Predictive Coder*

4.2 System Architecture

As mentioned in previous section, we have to distinguish two architectures, a transmitter one and a receiver one. These two architectures implemented according the standards of ETSI to totally conform to a real GSM system. Principally, we are interesting on receiver than transmitter implementation; however it was important to implement both a transmitter and receiver, both for debugging and succeeded accomplishment reasons. **Figure 4.2-1** shows the development platform used to implement the system including implementation modules.

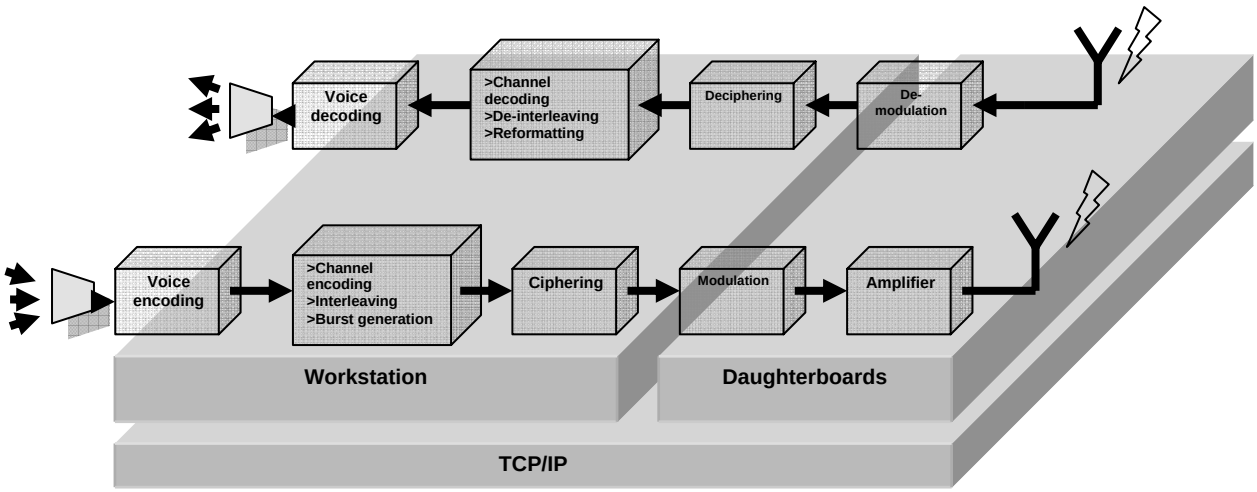


Figure 4.2-1: Development platform

More detailed architecture separating transmitter and receiver and assuming them as separate procedures is illustrated in the following figures. **Figure 4.2-2** shows the whole procedure of receiving a signal and playing it on a speaker while **Figure 4.2-3** shows the inverse procedure of sampling an audio input device and transmitting the data. Also, it is shown the control modules and the way they communicate each other.

Following paragraphs describes in hardware as well software components providing implementation details primarily for the second one.

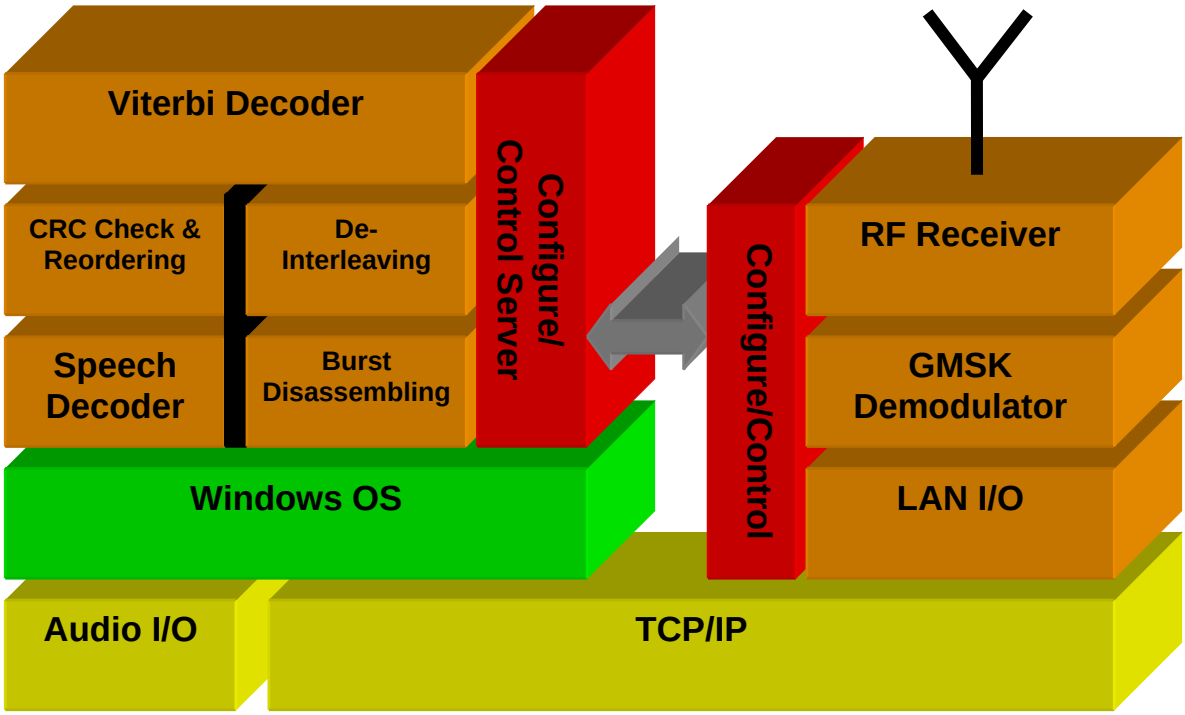


Figure 4.2-2: Receiver's architecture

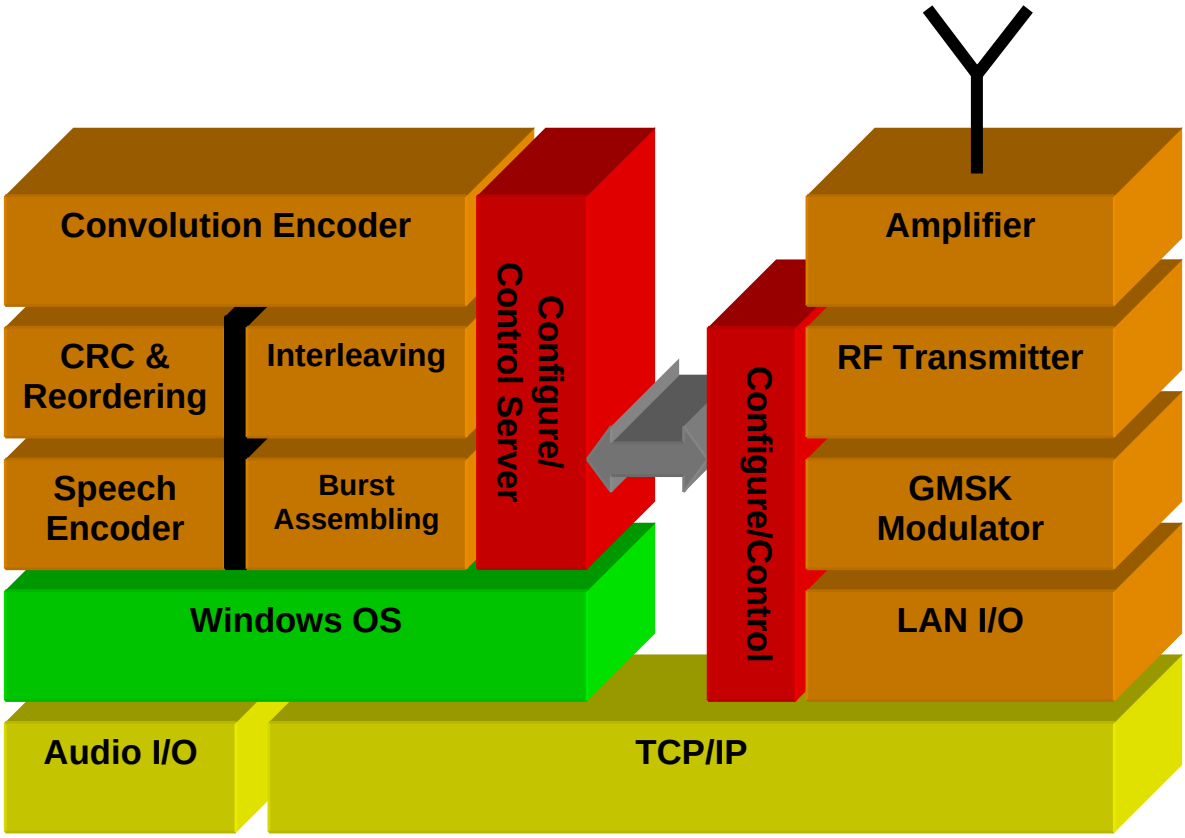


Figure 4.2-3: Transmitter's architecture

4.2.1 Hardware Architecture

Similar to every telecommunication system, hardware components consists of an RF receiver and a demodulator. System modules and the way they are connected are listed on **Figure 4.2-4**, full and detailed description of each device are cited on **Appendix I**.

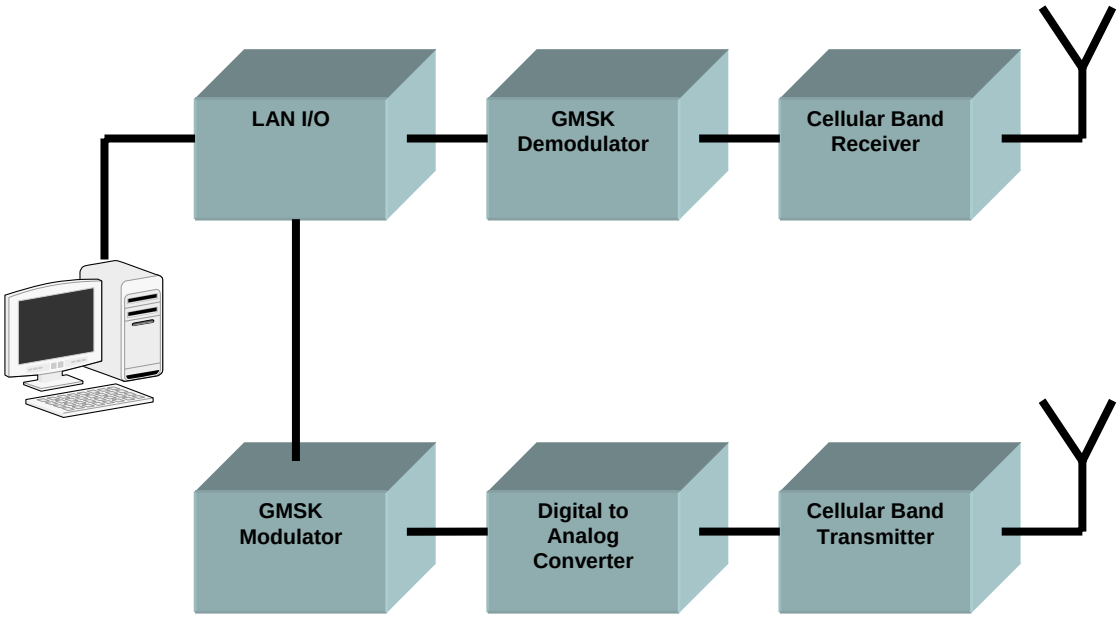


Figure 4.2-4: System hardware components

RF receiver front end has three basic elements as shown on **Figure 4.2-5**, a frequency synthesizer for tuning to the appropriate frequency, a set of wideband and narrow band low pass filters and analog to digital converters. The output of analog to digital converters feeds the GMSK demodulator that performs extra filtering, according to GSM parameters, and generates four soft-quantized bits demodulated data bits symbols. A transmitter, respectively, consists of a digital modulator, and an analog RF transmitter [see **Appendix I**].

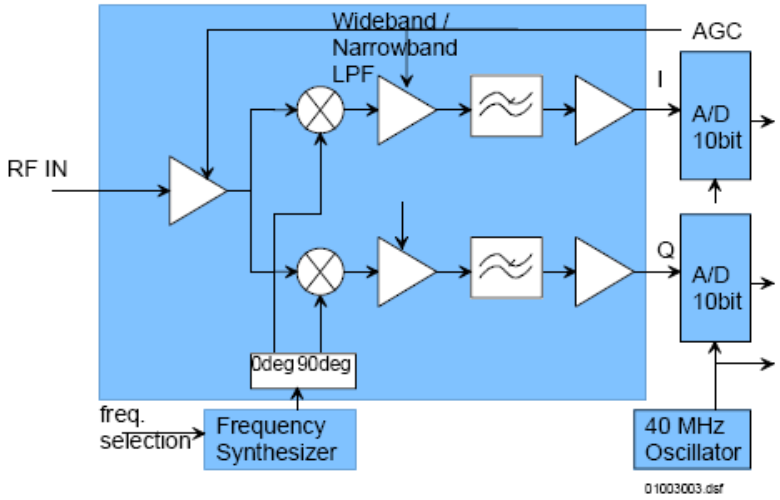


Figure 4.2-5: RF front end and ADCs

The block diagram of GMSK demodulator is illustrated on **Figure 4.2-6**. Although that the four soft-quantized bits could feed a soft-decision Viterbi module, interconnection module incompatibility forced the use only of the most significant bit and use hard decision Viterbi based on the hamming weight. Note that a soft decision algorithm could give a better noise margin for the wireless channel.

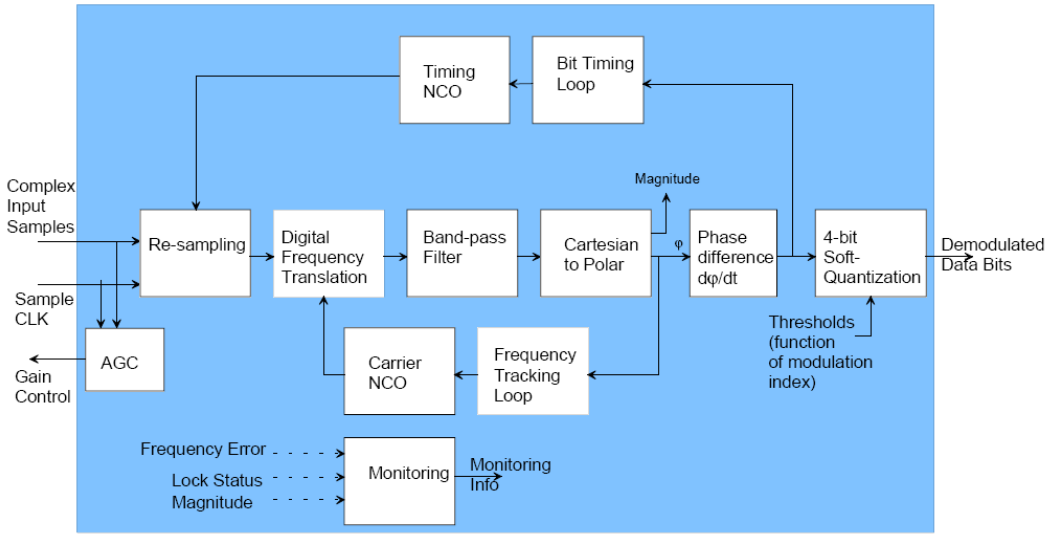


Figure 4.2-6: GMSK demodulator

4.2.2 Software Architecture

Main software has been developed in C language using windows API and Visual Studio as an *Integrated Development Environment* and comprises a library API performing GSM protocol stack. Another part of software has been developed in Java Standard Edition for controlling-setting the hardware providing the capability of integrating a complete system. The full software stack architecture is presented on next figure (**Figure 4.2-7**).

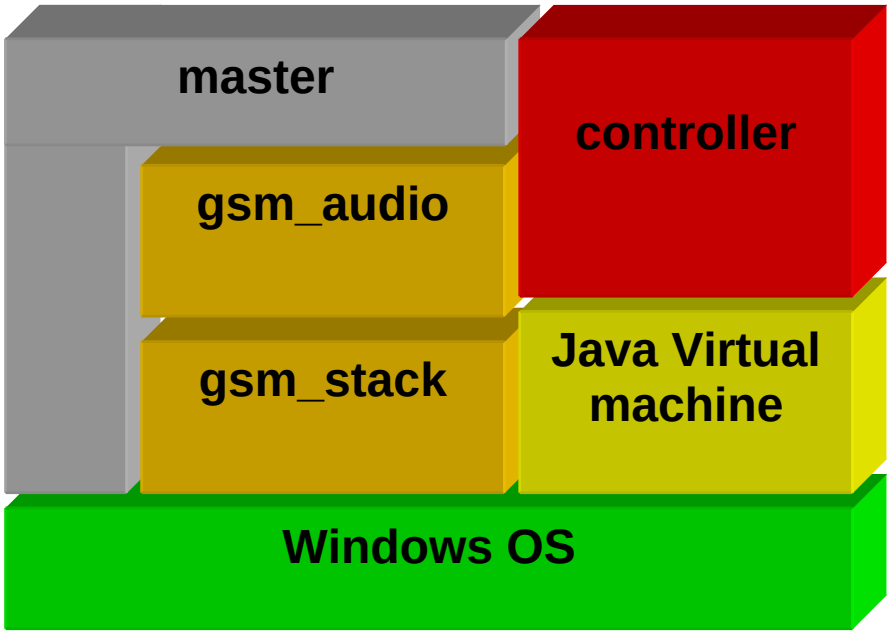


Figure 4.2-7: Software architecture

4.3 Software Implementation

Software stack consists of three basic projects, the *gsm_audio* project which implements all source coding and decoding algorithms, the *gsm_stack* project for source coding and decoding as well as encryption modules and finally a *master* project responsible for accessing Operating System and network resources.

4.3.1 Source Coding and Speech Processing

Source coding reduces redundancy in the speech signal resulting compression in signal, which means that an important lower bit rate is achieved than needed by the original speech signal. The speech coder/decoder is central part of any GSM speech processing function, both on transmitter (**Figure 4.3-1**) as well as at the receiver (**Figure 4.3-2**).

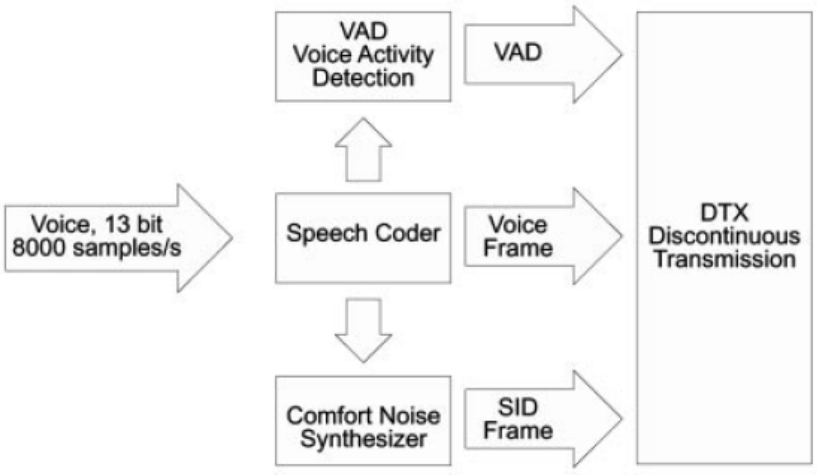


Figure 4.3-1: Schematic representation of speech functions on the transmitter

The analog source (pc microphone) or a .wav[3] file) is sampled at a rate of 8000 samples per second with a quantized resolution of 13 bits per sample. Actually sound device is sampled with a 16 bits resolution to a bit rate of 128 kbit/s. GSM speech processor uses input samples of 13 bits each one, that's why the three least significant bits are discarded. This corresponds to a bit rate of 104 kbit/s for the speech signal. Every speech frame, at the input of speech coder, has 160 samples of 13 bits arriving every 20 ms. The speech coder compresses this speech signals into a source-coded speech signal of 260-bit blocks at a bit rate of 13 kbit/s (GSM full rate payload voice throughput), achieving a compression ratio of 1 to 8. The speech coder uses a procedure known as *Regular Pulse Excitation - Long term Prediction – Linear Predictive Coder* (RPE-LTP). Details on this algorithm presented on ETSI 6.10 standard [12] and will not be discussed here since it does not constitute an important subject of this work.

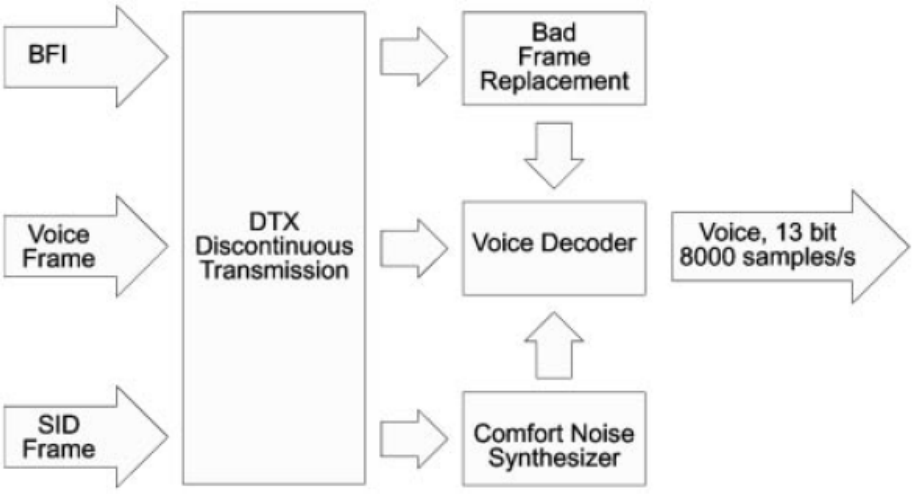


Figure 4.3-2: Schematic representation of speech functions on the receiver

[3] Wave file format mono at 8000 samples per second, 16bit each

4.3.2 Channel Coding

The varying properties of the mobile radio channel result in a very high bit error ratio on the order of 10^{-3} to 10^{-1}. Suitable error correction procedures are therefore necessary to reduce the bit error probability into an acceptable range of 10^{-5} to 10^{-6}. Channel coding, in contrast to source coding, adds redundancy to the stream of data to enable detection and correction of transmission errors.

The GSM system uses a combination of several procedures like a block code, which generates parity bits for error detection, a convolutional code generating the redundancy needed for error detection and a sophisticated interleaving of data over several block for reducing of damage done by burst errors.

Figure 4.3-3 shows the individual steps of channel coding that comprise:

- Calculation of parity bits (block code) and addition of fill bits
- Error protection coding through convolutional coding and
- Interleaving

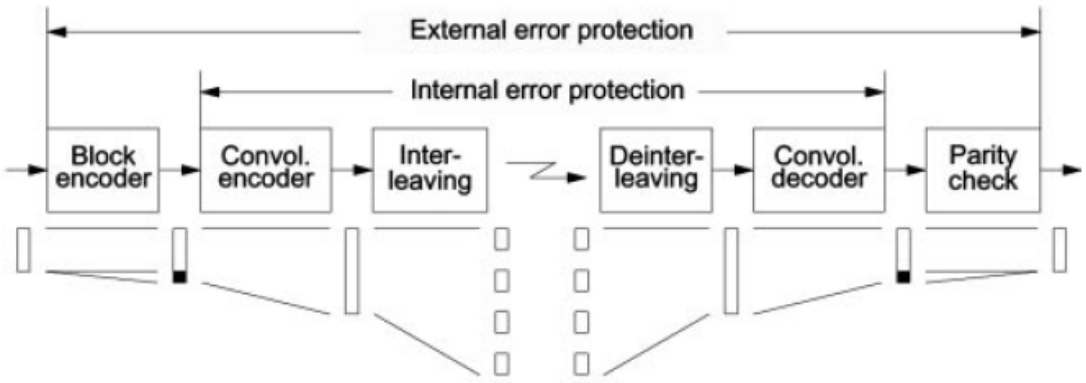


Figure 4.3-3: Stages of channel coding

The speech coder delivers to the channel encoder a sequence of blocks of data. In case of a full rate one block of data corresponds to one speech frame, each block contains 260 information bits, including 182 bits of class I (protected bits), and 78 bits of class II (no protection), (see **Figure 4.3-4**). The bits delivered by the speech coder are received in the order indicated in GSM 06.10 and have to be rearranged according to table 2 before channel coding. The rearranged bits are labelled { d(0), d(1), ..., d(259)}, defined in the order of decreasing importance. The class I bits are further divided into the class Ia and class Ib, class Ia bits being protected by a cyclic code and the convolutional code whereas the class Ib are protected by the convolutional code only.

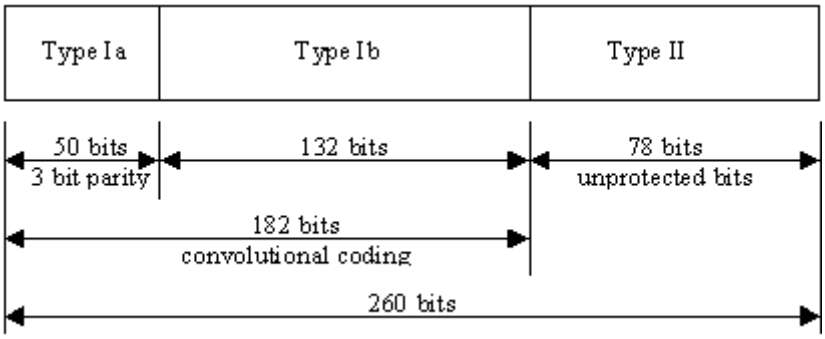


Figure 4.3-4: 1 audio block of 260 bits (20 ms)

A more detailed diagram of channel coding stages is illustrated on **Figure 4.3-5** bellow.

- 182 bits are protected by a convolutional block code with a convolutional efficiency of ½,
- Among these 182 bits, 50 are additionally protected by a detection code adding 3 redundancy bits. These 50 bits are the category Ia bits; the other 132 bits are category Ib bits,
- The other 78 bits are not protected.

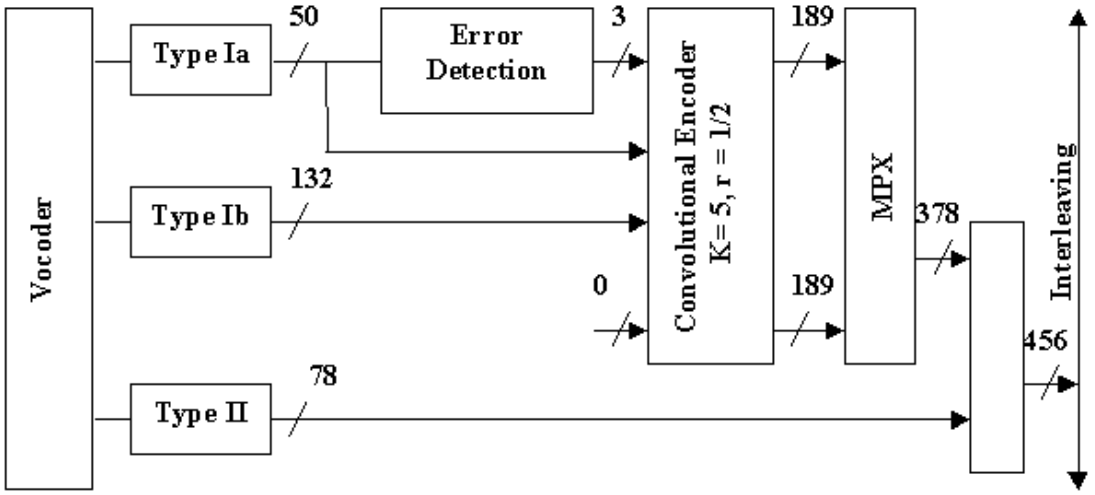


Figure 4.3-5: Traffic Channel Full rate transmission mode

4.3.3 External Error Protection

The block coding stage in GSM has the purpose of generating the parity bits for a block of data which allows the error detection in this block. A brief overview showing the codes used in GSM for each channel is listed on next figure. For our system we are interesting only on TCH and SCH CRC code with 260 bits input and 267 bits output.

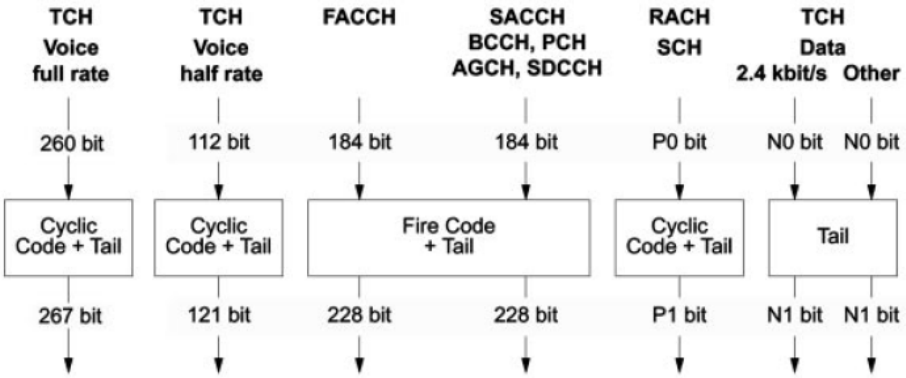


Figure 4.3-6: Overview of block coding for logical channels

A 3-bit *Cyclic Redundancy Check* (CRC) is calculated for the first 50 bits of Class I bits of a traffic frame. These parity bits are added to the 50 bits, according to a generate cyclic code (53, 50, 2), using the generator polynomial

$$G_{CRC}(x) = x^3 + x + 1$$

The encoding of the cyclic code is performed in a systematic form, which means that the polynomial:

$$d(0)D(52) + d(1)D(51) + ... + p(0)D2 + p(1)D + P(2)$$

Where p are the parity bits, when divided by g(D) yields a reminder equal to

$$1 + D + D^2$$

Since cyclic codes are easily generated with a feedback register, they can also be defined in their register form **in Figure 4.3-7**

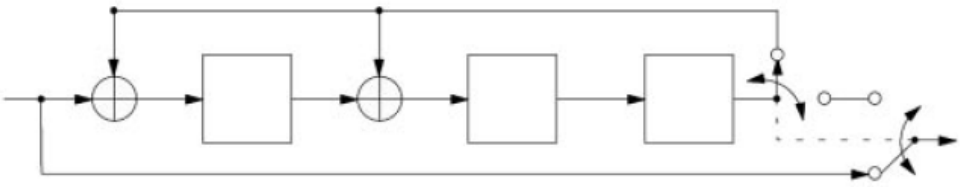


Figure 4.3-7: Feedback shift register of CRC

In our implementation, using blocks of 5 bits ($10x5 = 50$ bits) we used a CRC FSM translation matrix, resulting from a set of operations on data and polynomial where:

$$P(0) = d(4) \oplus d(1) \oplus R2 \oplus R3 \oplus R1$$
$$P(1) = d(3) \oplus d(0) \oplus d(1) \oplus R3$$
$$p(2) = d(2) \oplus R3 \oplus d(0) \oplus R2$$

Where R1, R2, R3 are register taps (R1 = least significant).

Finally tail bits are added and reordering is performed to construct the 189 bits long defined by:

$u(k)$	$= d(2k)$ and	$u(184-k) = d(2k+1)$	for k=0,1,...,90
$u(91+k)$	$= p(k)$		for k=0,1,2
$u(k)$	$= p(0)$		for k=185,186,187,188 (tail bits)

Where u(0) to u(188) are block encoder output bits.

4.3.4 Internal Error Protection

After block coding has supplemented the data with redundancy bits for error detection, as shown on **Figure 4.3-5**, the next stage is calculation of additional redundancy for error correction to correct transmission errors introduced by channel. The internal correction of GSM is based exclusive on convolutional codes. **Figure 4.3-6** shows different convolutional schemes for the different logical channels, all have the same constrain length K=5, but different generating polynomials and output rate. Our interest is on traffic channel's convolutional encoder CC(2, 1, 5).

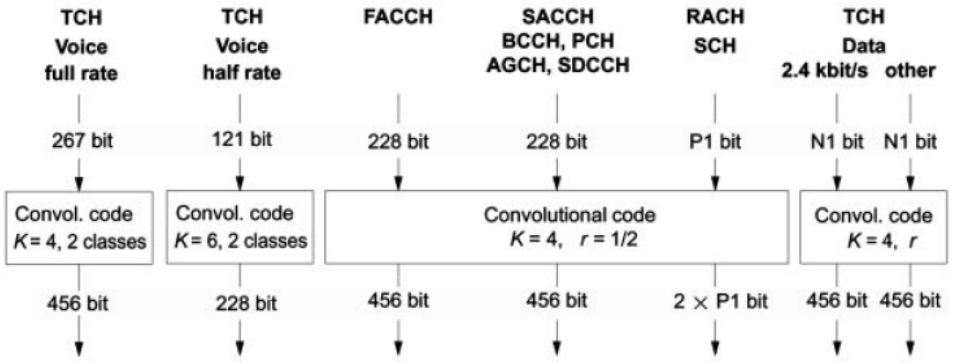


Figure 4.3-8: Overview of convolutional coding of logical channels

Like CRC, convolutional codes can be defined using shift registers and generating polynomials. The principle of convolutional encoding of a traffic channel is shown on

Figure 4.3-9. The output rate r of a convolutional encoder defines how many output bits are generated for each input coded bit. GSM has a rate r of 2, by multiplexing two polynomials G0 and G1.

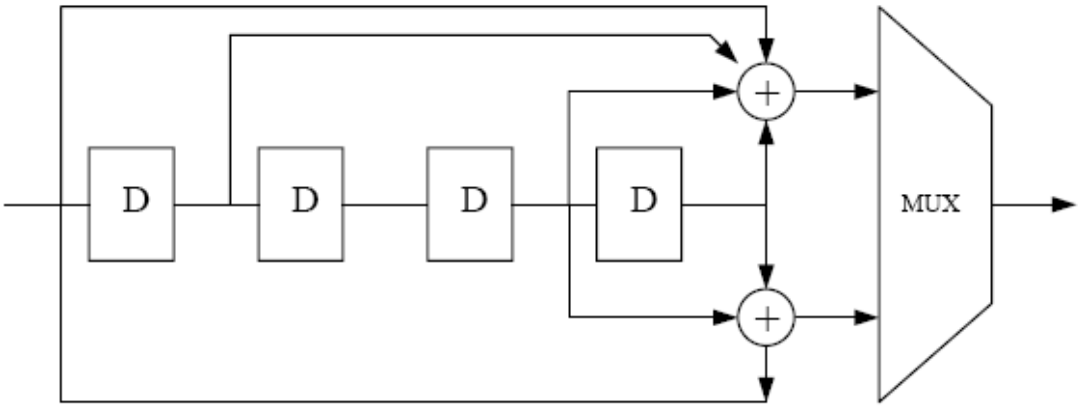


Figure 4.3-9: Principle of convolutional encoder for GSM

The coding procedure proper is expressed in the combinatorial operations (modulo 2 additions). In case of convolutional encoder of **Figure 4.3-9**, the two generating polynomials are:

$$G_0(d) = d^4 + d^3 + 1$$
$$G_1(d) = d^4 + d^3 + d + 1$$

The above polynomials are used both in TCH and SCH channels. For traffic channels the Class I bits are encoded using this polynomials and produces 378 bits that are concatenated with the rest, unprotected, 78 bits to produce a 456 bits data block. The coded bits {c(0), c(1), c(2), …, c(455)} are then defined by:

-Class I:

$$c(2k) = \quad u(k) + u(k-3) + u(k-4)$$
$$c(2k+1) = \quad u(k) + u(k-1) + u(k-3) + u(k-4) \qquad \text{for k=0,1,...,188}$$
$$u(k) \;=\; 0 \text{ for } k < 0$$

-Class II

$$c(378 + k) = \quad d(182 + k) \qquad \text{for k=0,1,...,77}$$

This coding scheme can be expressed as a *Finite State Machine* encoder like the scheme on **Figure 4.3-10**. The encoder on **Figure 4.3-9** can be represented as Mealy machine. The content on the shift register could be used to denote the states. Since each state is defined by a 4-bit register, the number of states is 16. Next state depends on current state as well as the input information bit of the encoder. This transition designates also the 2-bit output of the convolution coder.

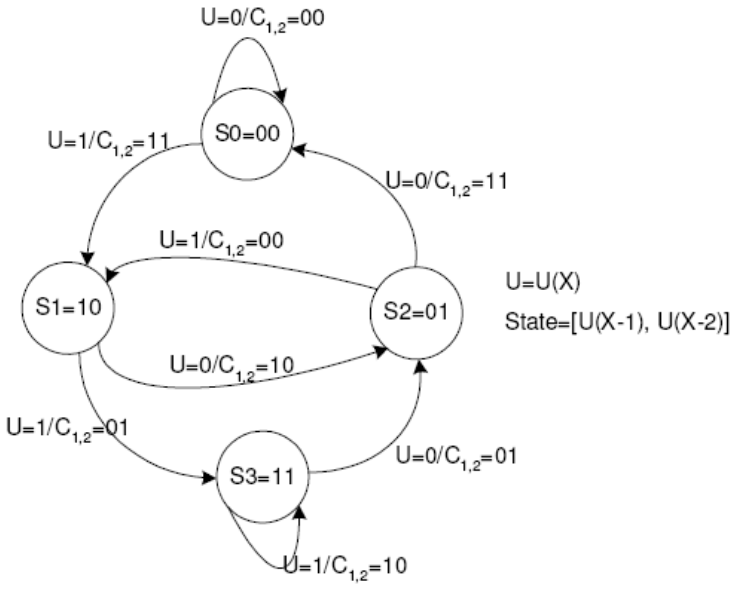


Figure 4.3-10: Encoder state machine

Table 4.3-1, formulates the FSM on a table format. This scheme used during implementation as a $16x2x2$ 3D static matrix to employ a fast mapping scheme. The 2-bits output symbol is been generated by a function with parameters the input bit u_i and current state S_j.

Table 4.3-1: Finite State Machine for Convolutional Encoder

FSM State		Next State		Output Symbol	
		in = 0	in = 1	in = 0	in = 1
S0	0000	0000	1000	00	11
S1	0001	0000	1000	11	00
S2	0010	0001	1001	11	00
S3	0011	0001	1001	00	11
S4	0100	0010	1010	00	11
S5	0101	0010	1010	11	00
S6	0110	0011	1011	11	00
S7	0111	0011	1011	00	11
S8	1000	0100	1100	10	01
S9	1001	0100	1100	01	10
S10	1010	0101	1101	01	10
S11	1011	0101	1101	10	01
S12	1100	0110	1110	10	01
S13	1101	0110	1110	01	10
S14	1110	0111	1111	01	10
S15	1111	0111	1111	10	01

An alternative presentation of the state diagram is the trellis diagram. The **Figure 4.3-11** is the trellis presentation of the state diagram in Figure 4.3-10. The trellis

diagram presents the time progress of the state transition. The time is expressed as the horizontal axis on the diagram. The left side states on the diagram are all the possible states at t=n, while the right side states are the states at t=n+1. The 8 edges are all the possible state transitions from time n to time n+1.

The trellis diagram can be used to express the encoding process better than the state diagram. Supposing the encoder is reset to state 0 at time t=0, if there is a stream of data 101100 coming from the cyclic encoder, the encoding activity could be indicated by the bold path in **Figure 4.3-12**. The output and the state transition of the encoder could be read out from the noted path.

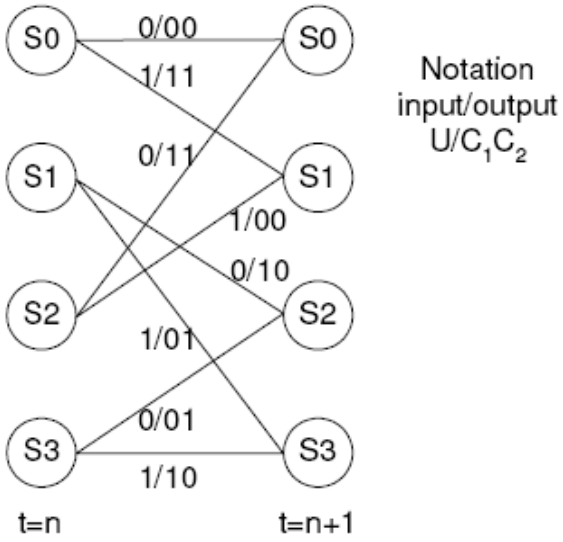


Figure 4.3-11: Trellis diagram

There are three well known decoding techniques for convolutional coding: the Viterbi algorithm, the sequential decoding and the feedback decoding. Sequential decoder's complexity is independent of constraint length K (K could be 41), thus can achieve very good error protection. But the main drawback of the sequential decoding is the requirement of a large buffer memory. Also the time needed for the decoding process is random. Feedback algorithm could only be used for hard-decision bit, which is not suitable for the targeting application. As a maximum likelihood decoding, Viterbi algorithm is the most used algorithm for low constraint length codes. Since the decoding complexity grows exponentially as K increases, the Viterbi algorithm is scarcely used if K is larger than 13. The GSM standard mostly uses low constraint length code like 5 and 7, thus makes the Viterbi algorithm a good choice.

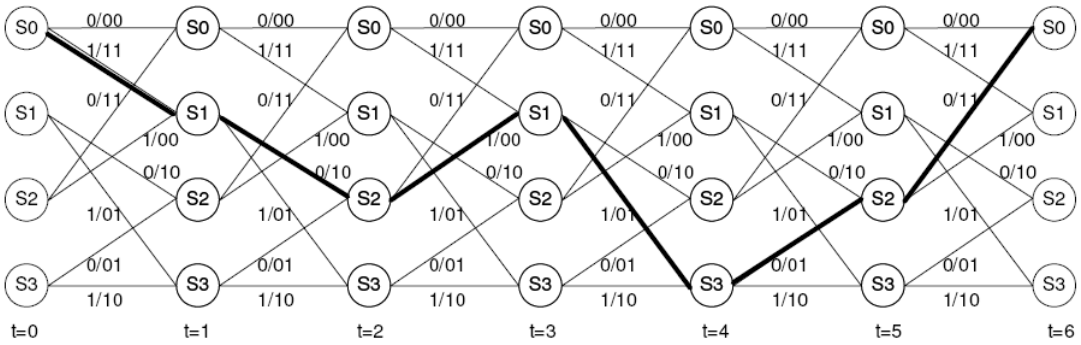


Figure 4.3-12: Trellis encoding scheme

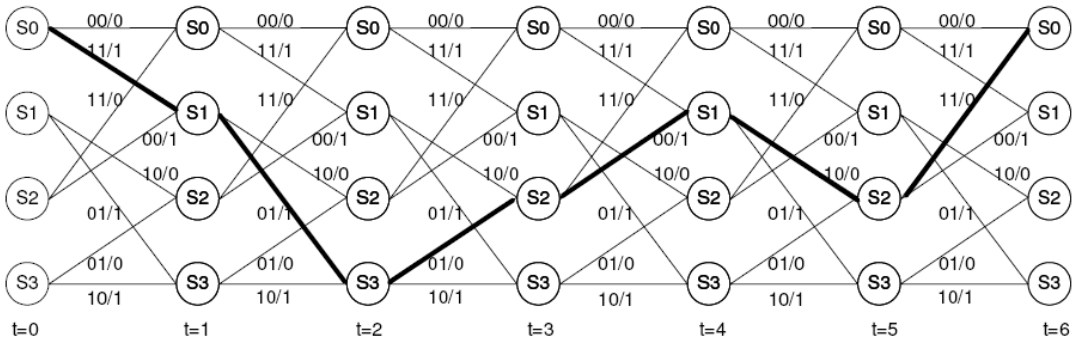


Figure 4.3-13: Trellis decoding diagram

4.3.5 Viterbi decoder

The concept of trellis diagram is of little use for the convolutional encoder, but it is the basic of Viterbi decoding. The decoding process could be described as looking for the best path on the trellis diagram that matches the decoder input. Consider a decoder that receives the transmitted signal 11 01 01 00 10 11 going from t=0 to t=6. Assume the encoding and decoding trellises were both reset to state S0 before t=0. The decoding path could be shown as **Figure 4.3-13**. Notice that the data on each edge is shown as C_1C_2 /U, where C is the decoder input and the U is the decoder output. The decoded data could be read from the path as 110100.

Just like encoding, decoding can be represented as an FSM. **Table 4.3-2** shows the state transition table for decoding process. This table implemented as a $16x2x4$ 3D static matrix to employ a fast mapping scheme. The 2-bits input symbol with current state as parameters generates the output bit u_i and next state S_j.

Table 4.3-2: Viterbi decoder Finite State Machine for GSM

FSM State		Next State				Output Symbol			
		in = 00	in = 01	in = 10	in = 11	in = 00	in = 01	in = 10	in = 11
S0	0000	0000	X	X	1000	0	X	X	1
S1	0001	1000	X	X	0000	0	X	X	1
S2	0010	1001	X	X	0001	1	X	X	0
S3	0011	0001	X	X	1001	0	X	X	1
S4	0100	0010	X	X	1010	0	X	X	1
S5	0101	1010	X	X	0010	1	X	X	0
S6	0110	1011	X	X	0011	1	X	X	0
S7	0111	0011	X	X	1011	0	X	X	1
S8	1000	X	1100	0100	X	X	1	0	X
S9	1001	X	0100	1100	X	X	0	1	X
S10	1010	X	0101	1101	X	X	0	1	X
S11	1011	X	1101	0101	X	X	1	0	X
S12	1100	X	1110	0110	X	X	1	0	X
S13	1101	X	0110	1110	X	X	0	1	X
S14	1110	X	0111	1111	X	X	0	1	X
S15	1111	X	1111	0111	X	X	1	0	X

Note that not all input combinations of state and input symbol defines an output symbol. These cases are marked with an x on the following table, and represent an error received symbol. In the previous example the channel is assumed as ideal. If the noise is present, the decoder input could be any data. In that case the trellis path is split in two parallel paths, each one for every valid transition (**Figure 4.3-14**)[4]. How to find a decoding path then? A way to measure the likelihood of the decoder input and the edge must be defined in order to keep only the right path.

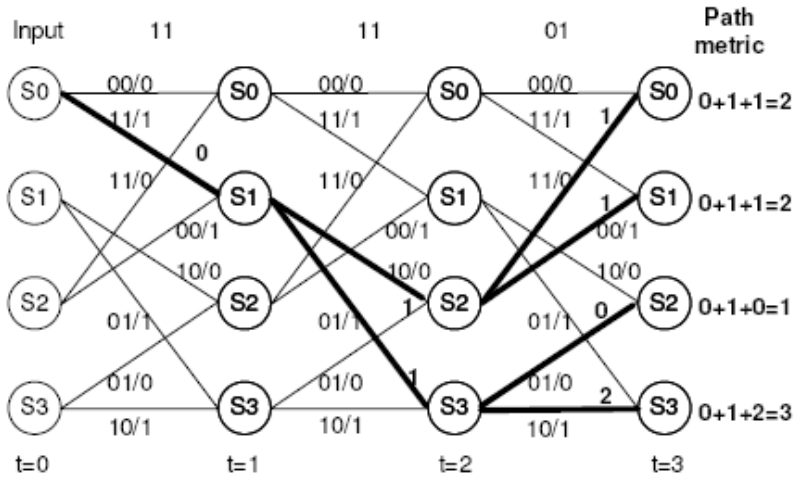


Figure 4.3-14: Path split and path metric

[4] Note that figure examples are simple and are not represent a GSM encoder/decoder process

Branch Metric

The Hamming distance is the simplest decoding-correctness measurement to use. Compare the bits in the same positions in two different binary numbers. The number of positions that are different is the Hamming distance. For example, the distance between 00 and 11 is 2, while the distance between 00110 and 10100 is also 2. In trellis diagram, each state has two edges that connect to the states in the previous decoding stages and two edges extend to the states of the next stage. Those edges are called *branches.*

Table 4.3-3: Hamming metric

Bits Received	Valid Codeword 1	Valid Codeword 2	Hamming Metric 1	Hamming Metric 2
00	00	11	2	0
01	10	01	0	2
10	00	11	1	1

Path Metric

Suppose an input stream 11 11 01 is received and the initial state is S0, the decoder extends paths as shown in **Figure 4.3-14**. The first decoding stage has only one survivor S0-S1, since the other branch have metric which is larger than 0. The second stage has more than one survivor, since both branches have a metric of 1. At the third stage, the two previous survivors extend themselves into four branches, each of which has its own branch metric. Starting from S0 at t=0, the branches fork into four paths at t=3. To evaluate which path is most likely to be the winner, the *branches metrics* along each path are accumulated to form a *path metric*. In this case, the path S0-S1-S3-S2 has the lowest *path metric*, thus is the potential winner.

In our implementation this procedure of finding the most possible (maximum likelihood) survivor is performed in a more optimal way. It is not necessary to receive the full block of 189 symbols (378 bits) in order to perform a trace back. Trace back is the decoding run through for finding the sole survivor of the decoding stage. During the decoding phase, a linked list of current survivors is kept and increases its nodes in every erroneous symbol received. At any point that this list exceed a certain predefined number (i.e. greater than 32 since we have 16 states) of active survivors a trace back is initiated to keep only the most possible one.

In that way the errors introduced by wireless channel are corrected. It is important that these errors occur infrequently and the probability of sequential or grouped errors is small. Especially, negative for error correction are burst errors during longer and deeper fading period. Therefore, burst errors occurring frequently on the radio channel should be distributed uniformly across the transmitted codeword. The spreading of these errors is possible using interleaving. After convolutional encoder, on the

transmitter, and before Viterbi decoder on the receiver, interleaving and de-interleaving procedures are taking place respectively.

4.3.6 Interleaving

To further protect against burst errors common to the radio interface, each sample is interleaved. The 456 bits output by the convolutional encoder are divided into 8 blocks of 57-bit, and these blocks are transmitted in eight consecutive time slot bursts. Since each time slot bursts can carry two 57-bit blocks, each burst carries traffic from two different speech samples.

GSM defines interleaving as a combination of two interleaving types, a diagonal and a block one. **Table 4.3-4** summarizes the way a 456 bits data word is interleaved across eigth 57-bit long interleaving blocks.

Table 4.3-4: Interleaving algorithm of a full rate traffic channel

0	8	.	.	.	448	Even bits of burst N+0
1	9	.	.	.	449	Even bits of burst N+1
2	10	.	.	.	450	Even bits of burst N+2
3	11	.	.	.	451	Even bits of burst N+3
4	12	.	.	.	452	Odd bits of burst N+4
5	13	.	.	.	453	Odd bits of burst N+5
6	14	.	.	.	454	Odd bits of burst N+6
7	15	.	.	.	455	Odd bits of burst N+7
←	57 columns				→	

The data of nth code word are distributed across eight interleaving blocks at 114 bits each, beginning with block B=B0+4n. In this way, only the even bits of the first four blocks (B+0,1,2,3) and the odd bits of the last four blocks (B+4,5,6,7) are used. The even bits of the last four blocks (B+4,5,6,7) are occupied by data from block n+1. Each interleaving block thus contains 57 bits of the current data block n and 57 bits of the following data block n+1 as shown on **Figure 4.3-15**.

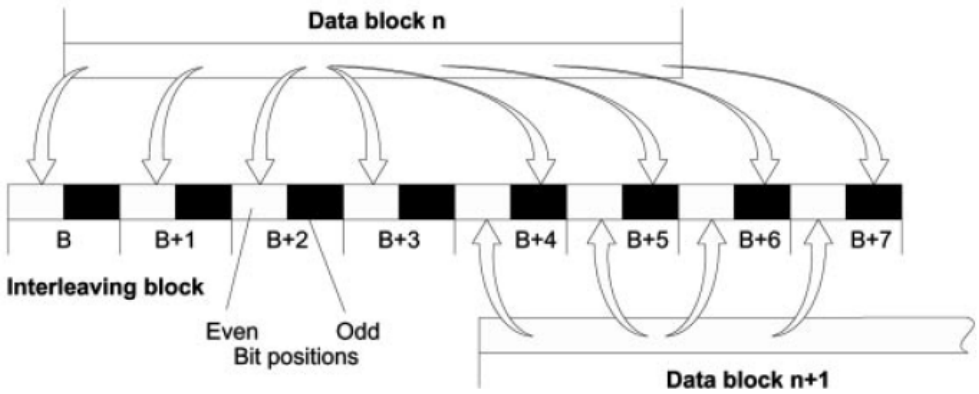


Figure 4.3-15: Interleaving TCH/FS block mapping

The coded bits are reordered and interleaved according to the following rule:

$$i(B, j) = c(n, k), \quad \text{for k} = 0,1,...,455$$
$$\text{n} = 0,1,...,N,N+1,...$$
$$B = B0 + 4n + (k \bmod 8)$$
$$j = 2((49k) \bmod 57) + ((k \bmod 8) \mathit{div} 4)$$

Note that an inverse procedure is performed on a receiver to build a 456-bit block before feeding the Viterbi decoder module. Both in transmitter and receiver, interleaving and de-interleaving is executed using bit mapping matrices to produce the resulting blocks.

4.3.7 Mapping on a burst

After convolutional coding and interleaving, the data is available in form of 114-bit interleaving blocks. This corresponds exactly to the amount of data a burst can carry. Each interleaving block is mapped directly onto one burst (**Figure 4.3-16**)

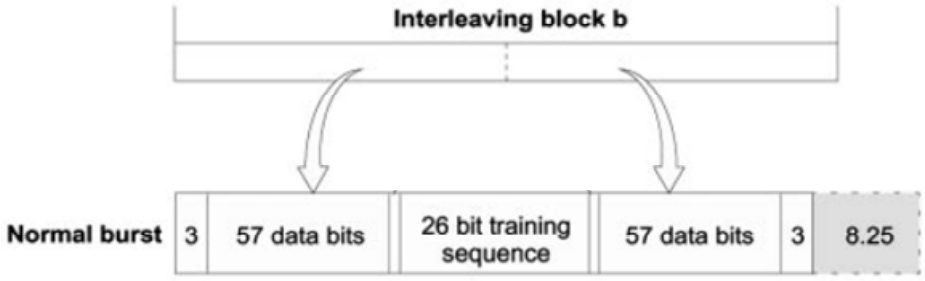


Figure 4.3-16: Mapping onto a burst

GSM uses a set of training sequences to perform channel equalization, this training sequence set consists of 26-bit long words described on the following table. Neighbor cells with same frequency set, uses different training sequence to eliminate interference.

Table 4.3-5: The GSM training sequences

Training sequence Code TSC	Training sequence bits (b61, b62, …, b86)
0	(0,0,1,0,0,1,0,1,1,1,0,0,0,0,1,0,0,0,1,0,0,1,0,1,1,1)
1	(0,0,1,0,1,1,0,1,1,1,0,1,1,1,1,0,0,0,1,0,1,1,0,1,1,1)
2	(0,1,0,0,0,0,1,1,1,0,1,1,0,1,0,0,0,1,0,0,0,1,1,1,1,0)
3	(0,1,0,0,0,1,1,1,1,0,1,1,0,1,0,0,0,1,0,0,0,1,1,1,1,0)
4	(0,0,0,1,1,0,1,0,1,1,1,0,0,1,0,0,0,0,0,1,1,0,1,0,1,1)
5	(0,1,0,0,1,1,1,0,1,0,1,1,0,0,0,0,0,1,0,0,1,1,1,0,1,0)
6	(1,0,1,0,0,1,1,1,1,1,0,1,1,0,0,0,1,0,1,0,0,1,1,1,1,1)
7	(1,1,1,0,1,1,1,1,0,0,0,1,0,0,1,0,1,1,1,0,1,1,1,1,0,0)

Finally this packet is encapsulated into an Ethernet packet and is delivered over LAN to the hardware for modulation and transmission. The only difference that is introduced is the guard period. Guard period differs and is only 4 bits due to implementation and hardware restrictions. Since hardware has not any clock synchronized with GSM bit synchronization by a FCCH channel, it is difficult to perform an accurate burst relative synchronization. Synchronization is a very important constraint of implemented architecture and is detailed discussed in a later paragraph.

4.4 Encryption

Every couple of 57-bit payload of a burst is ciphered using a 114 bit long key. The 114-bit payload is XORed with 114 bit key according to the diagram on **Figure 4.4-1**. The key used for encryption is generated according A5 algorithm and depends on Kc key and *Frame Number* FN.

Encryption procedures as well as security mechanism of GSM will be not discussed here. Next chapter, **Chapter 5**, makes a detailed analysis of GSM security model and its functions.

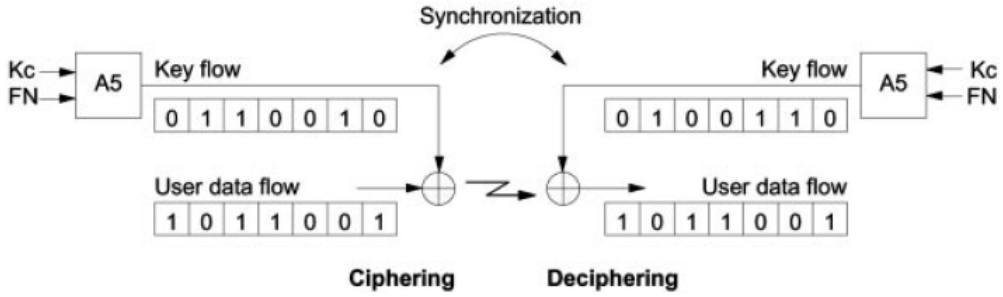


Figure 4.4-1: Combining payload data stream and ciphering stream

4.5 Synchronization

Successful operation of a mobile radio system is based on synchronization between the mobile and base station. Frequency agreement must be established between transmitter and receiver, for that reason GSM defines two kinds of synchronization: frequency synchrony and time synchrony.

Frequency synchrony is used to synchronize transmitter and receiver oscillators. FCCH channel is responsible for such synchronization. Since, FCCH contains a data sequence of logical zero bits "0" and cause the BT product of 0.3 of GMSK modulator, there exists a pure sine wave signal with a frequency shift of 1625/24 kHz (67,7 kHz).

Bit synchronization is performed by the SCH channel which provides information about *Frame Number* along with an extended 64-bit synchronizing training sequence.

In system implemented, only bit synchronization can be performed using SCH channel while FCCH channel must be interpreted and manipulated by specific hardware. Current hardware uses only a frequency tracking loop.

GSM also uses slow frequency hopping techniques to reduce intra-cell interference and also, to achieve higher security levels. In other words, to be able for a system to monitor a traffic channel it has to maintain a frequency change every 4.615 ms or about 217 times per second. This fact introduces another synchronization constraint, since hardware must be able to change to and synchronize to a new frequency in less than 4 ms.

Figure 4.5-1 shows a non frequency hopping scheme at time frequency domain while **Figure 4.5-2** illustrates a frequency hopping scheme. Most of the operators use a frequency hopping scheme, hopping between two frequencies. However, more complicated schemes define algorithms for a more complicated frequency hopping among a set of frequencies.

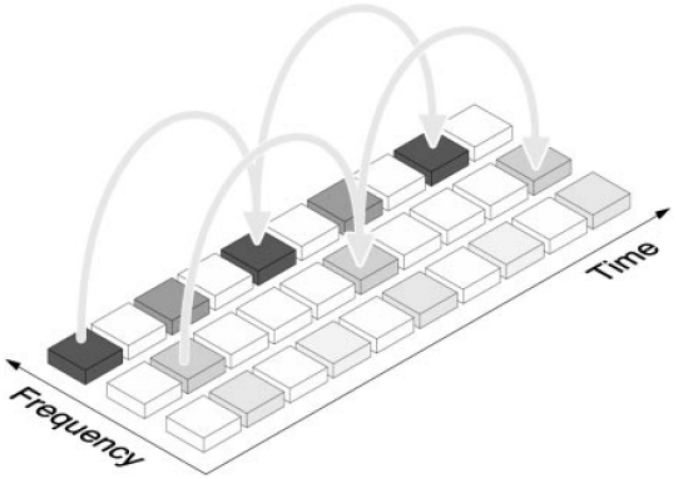


Figure 4.5-1: Non frequency hopping scheme

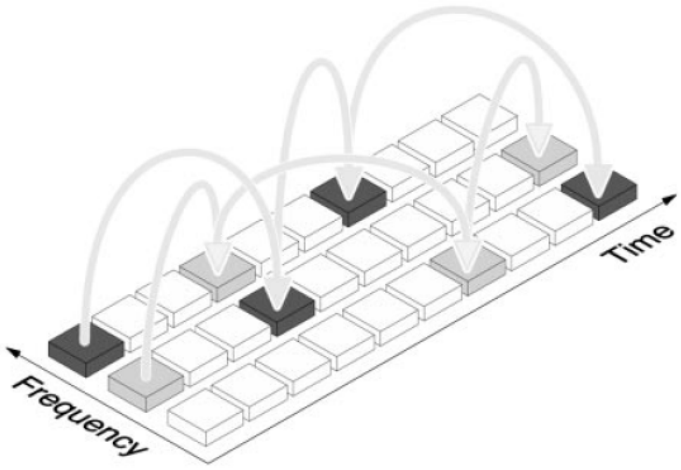


Figure 4.5-2: Frequency hopping scheme

Chapter 5

Security in the GSM System

Contents

5.1 Introduction to the GSM Security Model

The purpose of security for GSM system is to make the system as secure as the public switched telephone network and to prevent phone cloning. The use of air interface at the transmission media allows a number of potential threats from eavesdropping.

GSM specification 02.09 identifies three areas of security that are addressed by GSM.

- Authentication of a user – this deal with the ability for a mobile phone to prove that it has access to a particular account with the operator

- Data and signaling confidentiality – this requires that all signaling and user data (such as text messages and speech) are protected against interception by means of ciphering

- Confidentiality of a user – this deals with the fact that when the network needs to address a particular subscriber, or during the authentication process, the unique IMSI (*international mobile subscriber identity*) should not be disclosed in plaintext. This means someone intercepting communications should not be able to learn if a particular mobile user is in the area.

These 3 areas are covered in detail below.

The GSM Security Model is based on a shared secret between the subscriber's home network's HLR and the subscriber's SIM (**Figure 5.1-1)**. The shared secret, called Ki, is a 128-bit key used to generate a 32-bit signed response, called SRES, to a Random Challenge, called RAND, made by the MSC, and a 64-bit session key, called Kc, used for the encryption of the over-the-air channel. When a MS first signs on to a network, the HLR provides the MSC with five triples containing a RAND, a SRES to that particular RAND based on the Ki and a Kc based again on the same Ki. Each of the triples is used for one authentication of the specific MS. When all triples have been used the HLR provides a new set of five triples for the MSC.

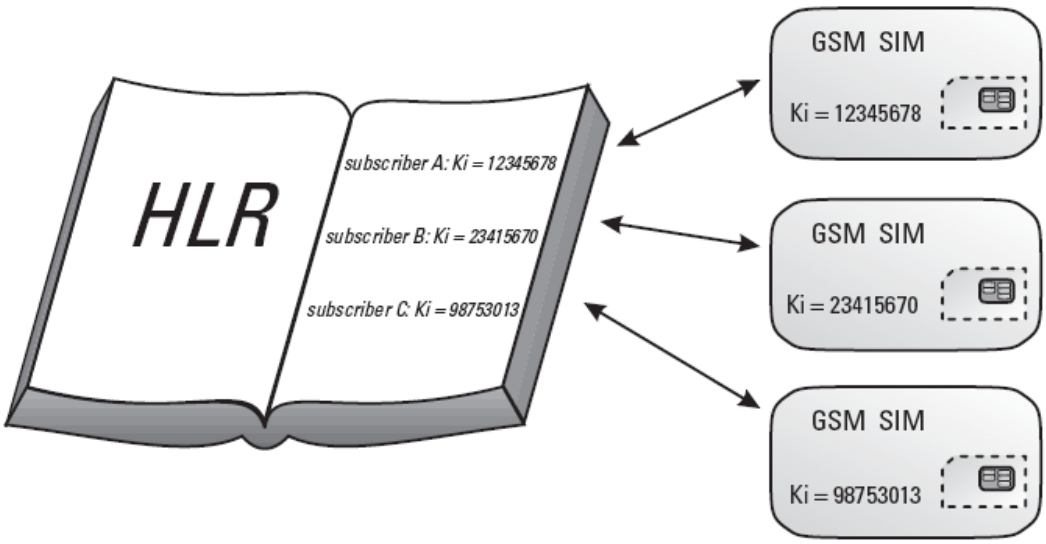


Figure 5.1-1: Only SIM and HLR know the value of Ki

On its own, the phone has no association with any particular network. The appropriate account with a network is selected by inserting the SIM into the phone. Therefore the SIM card contains all of the details necessary to obtain access to a particular account. These details come down to just 2 items of information.

- The IMSI – International Mobile Subscriber Identity – a unique number for every subscriber in the world.

- The Ki – the root encryption key. This is a randomly generated 128-bit number allocated to a particular subscriber that seeds the generation of all keys and challenges used in the GSM system. The Ki is highly protected, and is only known in the SIM and the network's AuC (Authentication Centre). The phone itself never learns of the Ki, and simply feeds the SIM the information it needs to know to perform the authentication or generate ciphering keys.

5.2 Authentication

Authentication is needed in a cellular system to prohibit an unauthorized user from logging into the network claiming to be a mobile subscriber. If this were possible, it would be easily possible to "hijack" someone's account and impersonate that person (or simply making that person pay for the services). In fact, this was possible in some earlier cellular systems. In order to solve this problem, some sort of challenge needs to be issued by the network which the mobile phone (MS) must respond to correctly.

When the MS first comes to the area of a particular MSC, the MSC sends the Challenge of the first triple to the MS. The MS calculates a SRES with the A3 algorithm using the given Challenge and the Ki residing in the SIM as input, and generates a 32-bit output (**Figure 5.2-2)**. The A3 is the authentication algorithm in the GSM security model. Both the RAND and the Ki secret are 128 bits long. The MS then sends the SRES to the MSC, which can confirm that the SRES really corresponds to the Challenge sent by comparing the SRES from the MS and the SRES in the triple from the HLR. Thus, the MS has authenticated itself to the MSC.

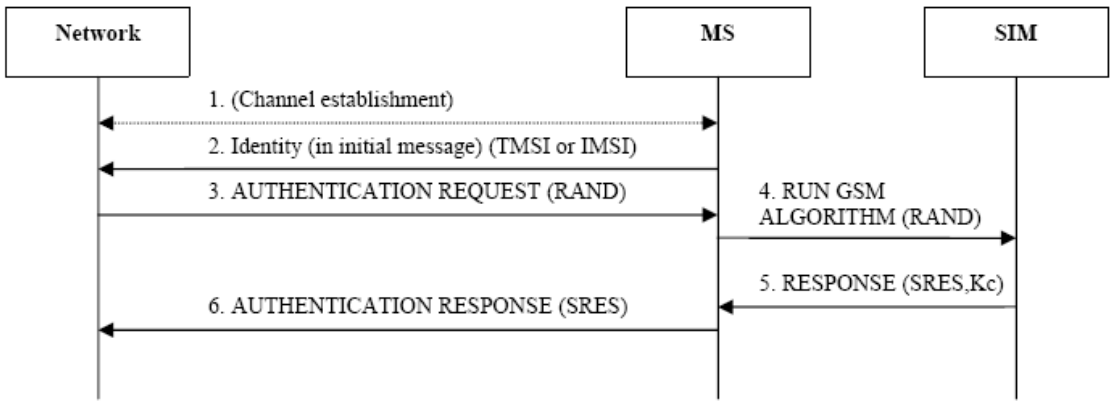


Figure 5.2-1: Authentication process

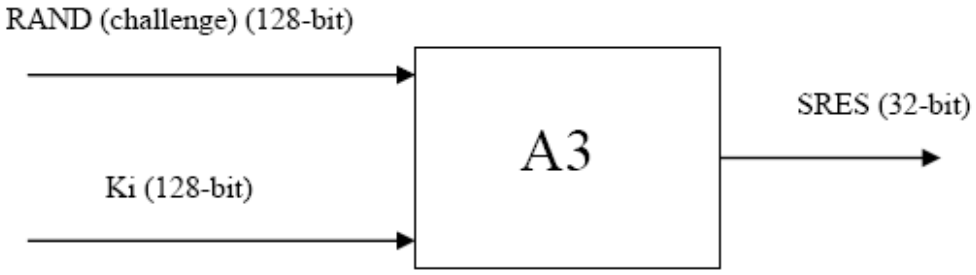


Figure 5.2-2: SRES computation on MSC

The A3 algorithm does not refer to a particular algorithm, rather the algorithm the operator has chosen to be implemented for authentication. The most common implementations for A3 are COMP128v1 and COMP128v2. In fact, both of these algorithms perform the function of both A3 and A8 (the ciphering key generation algorithm – discussed later) in the same stage.

Whenever the SIM is asked to compute the SRES (with the RUN GSM ALGORITHM command) it also computes a new Kc (ciphering key – discussed later). Thus not only is the authentication procedure used to verify a user, it is also used whenever the network wishes to change keys.

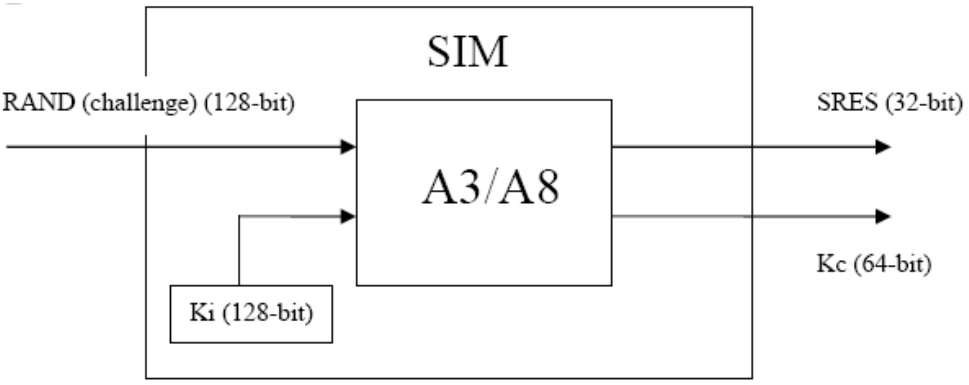


Figure 5.2-3: SIM authentication concept

5.3 Encryption

Ciphering is highly important to protect user data and signaling data from interception. The GSM system uses symmetric cryptography - the data is encrypted using an algorithm which is 'seeded' by the ciphering key Kc. This same Kc is needed by the decryption algorithm to decrypt the data. The idea is that the Kc should only be known by the phone and the network. If this is the case, the data is meaningless to

anyone intercepting it. The Kc should also frequently change, in case it is eventually compromised.

5.3.1 Generating Security Data

The method of distributing the Kc to the phone is closely tied in with the authentication procedure discussed above. Whenever the A3 algorithm is run (to generate SRES), the A8 algorithm is run as well (in fact the SIM runs both at the same time). The A8 algorithm uses the RAND and Ki as input to generate a 64-bit ciphering key (**Figure 5.3-1)**, the Kc, which is then stored in the SIM and readable by the phone. The network also generates the Kc and distributes it to the base station (BTS) handling the connection and performing deciphering.

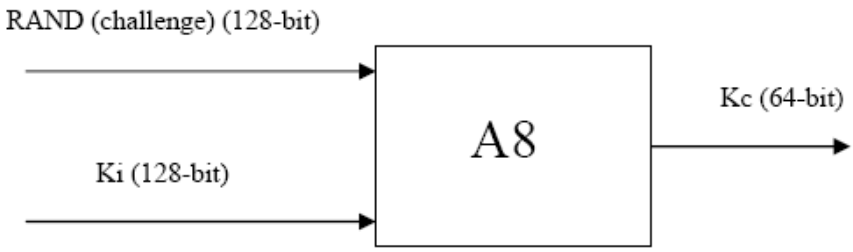


Figure 5.3-1: A8 algorithm

5.3.2 Encryption of Payload Data

At any time, the network can then order the phone to start ciphering the data (once authenticated) using the Kc generated. The ciphering algorithm works by generating a stream of binary data (the cipher block), which is modulo-2 added (XORed) with the user data, to produce the ciphered text which is transmitted over the air. The data is decrypted by XORing the received data with the cipher block, which should be the same if the Kc is the same.

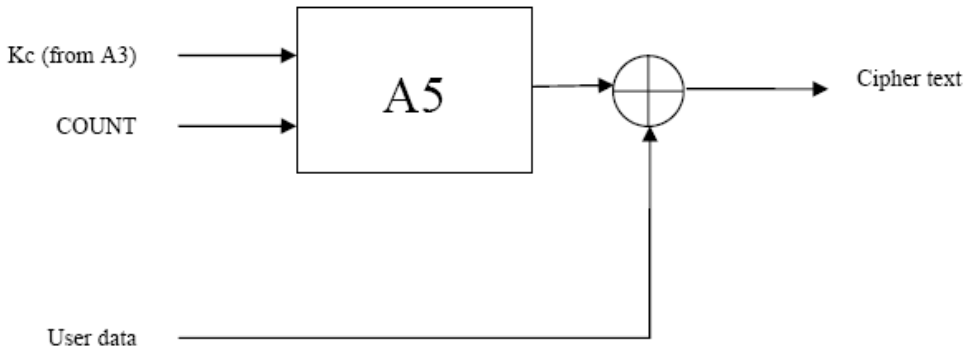


Figure 5.3-2: A5 Algorithm

The algorithm is also 'seeded' by the value COUNT, which is based on the TDMA *frame number*, sequentially applied to each 4.615ms GSM frame. Internal state of the algorithm is flushed after each burst (consisting of 2 blocks of 57 bits each). In the case of multi-slot configurations, different cipher contexts are maintained for each timeslot. The same base Kc is used, however it is manipulated for each timeslot by XORing bits 32-34 of the Kc with the 3-bit timeslot number (0-7).

5.3.3 Implementations of A3, A8

Although the design of the GSM system allows an operator to choose any algorithm they like for A3 & A8, many decided on the one that was developed in secret by the GSM association, COMP128. COMP128 eventually ended up in public knowledge due to a combination of reverse engineering and leaked documents, and serious flaws were discovered (as discussed below).

Some GSM operators have moved to a newer A3/A8 implementation, COMP128-2, a completely new algorithm which was also developed in secret. This algorithm for now seems to have addressed the faults of the COMP128 algorithm, although since it has yet to come under public scrutiny it may potentially be discovered via reverse-engineering and any possible flaws could be learned.

Finally, the COMP128-3 algorithm can also be used, it is simply the COMP128-2 algorithm, however all 64-bits of the Kc are generated, allowing maximal possible strength from the A5 ciphering algorithm (COMP128-2 and COMP128 still sets the 10 rightmost bits of the Kc to 0), deliberately weakening the A5 ciphering.

5.3.4 Frequency Hopping

Finally, as we previously show, slow frequency hopping is used in GSM, where the transceiver changes physical carrier every frame (4.615ms), or about 217 times per second. The hopping sequence is defined by two parameters – the HSN (Hopping Sequence Number), and the MAIO (Mobile Allocation Index Offset). There are 2 modes of hopping – cycling hopping and non cyclic hopping. In both modes, the MAIO simply chooses the phase in the hopping sequence that is to be used. If the HSN is 0, cyclic hopping is used where the mobile station simply steps through a set of frequencies (called the mobile allocation). In non-cyclic hopping, the (publicly known) frame number is used to seed a more complex hopping algorithm.

In both cases, the hopping adds a layer of security. In order to decipher the stream with no knowledge (at the time of eavesdropping) of the hopping sequence, the entire bandwidth needs to be sampled. In both GSM900 and GSM1800 this can amount to tens of megahertz (although can be reduced if the operator's frequency allocation for that area is known).

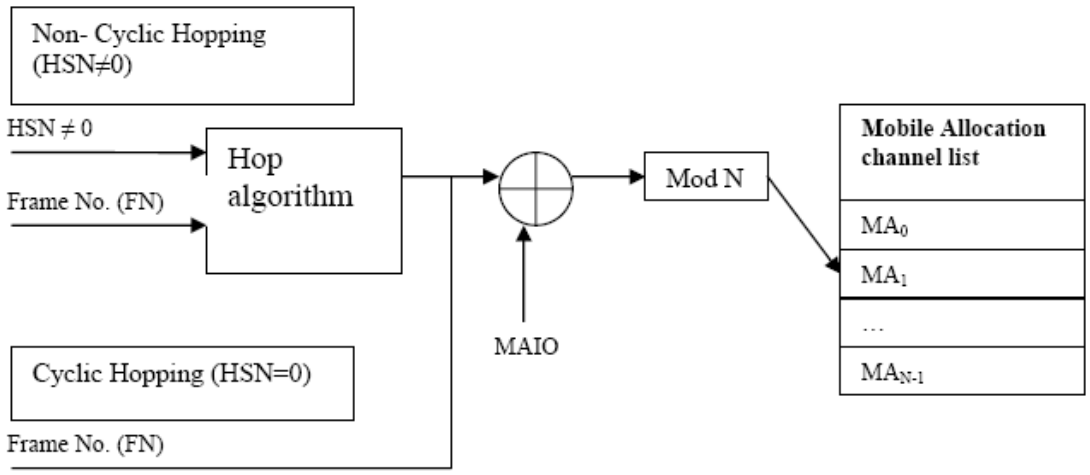


Figure 5.3-3: Frequency hopping algorithm

If the channel information is allocated to the phone in plaintext, as it is at the start of the connection, then that sequence can be easily followed. But, typically, when setting up a data or voice (TCH) channel, an additional channel is allocated by sending messages on the initial channel when encryption has been enabled, thus making it significantly more difficult for an attacker to learn the TCH sequence.

5.4 Protection of Subscriber Identity

As mentioned above, one of the main goals of GSM security was to avoid having to use the IMSI (International Mobile Subscriber Identity) in plaintext over the radio link, thus stopping an eavesdropper from determining if a particular subscriber was in an area and what services they were using.

This is avoided by addressing the phones by a 32-bit TMSI (Temporary Mobile Subscriber Identity), which is only valid in a particular Location Area (i.e. one paging domain). The subscriber addresses itself or is paged by the 32-bit TMSI from then on. The TMSI is updated at least during every location update procedure (i.e. when the phone changes LA or after a set period of time). The TMSI can also be changed at any time by the network. The new TMSI is sent in ciphered mode whenever possible so an attacker cannot maintain a mapping between an old TMSI and a new one and "follow" a TMSI.

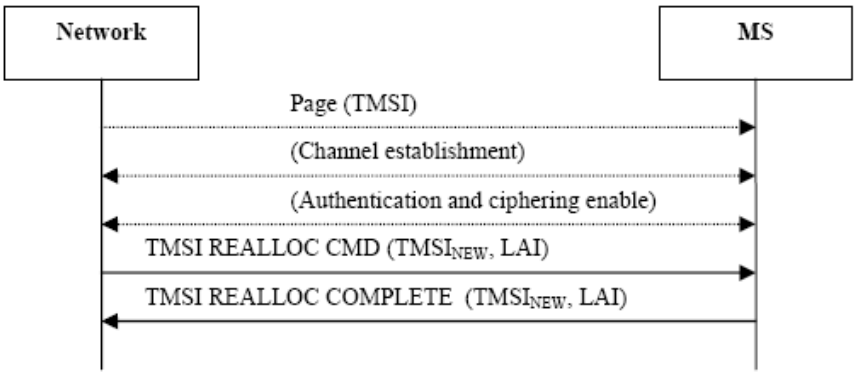


Figure 5.4-1: TMSI reallocation

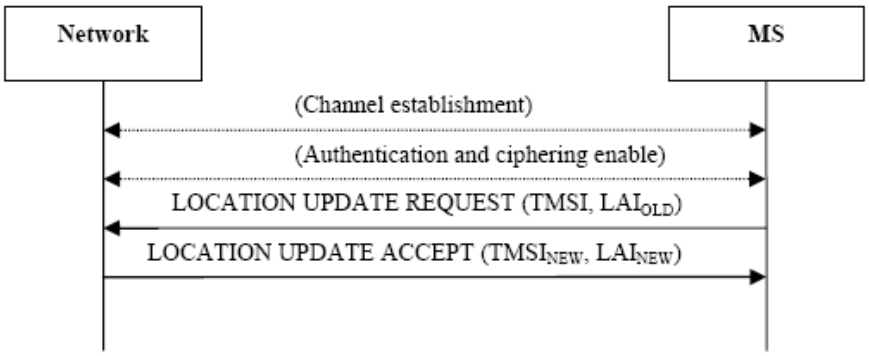


Figure 5.4-2: TMSI location update

However, a statistical practice could determine a TMSI. Performing an iteration of "blind calls" or "blind SMS messages" on a known MSISDN number we could be able to determine TMSI by monitoring the PCH (paging channel) for a frequently asked TMSI. This could be feasible if TMSI lasts for long time durations and does not change on every authentication request.

5.5 GSM Interception

The interesting question about the GSM security model is whether a call can be eavesdropped, since the algorithms it depends is known and has been proven faulty. The implemented architecture could provide the base where possible attacks could be applied on.

5.5.1 Man-in-the-middle attack

Although that most serious fault with the GSM authentication system, is that network does not authenticate itself to the phone. The authentication procedure described above does not require the network to prove its knowledge of the Ki, or any other authentication context to the phone.

Thus it is possible for an attacker to setup a false base station with the same Mobile Network Code as the subscriber's network. Since the authentication procedure initiation is up to the network's discretion, the false network may choose not to authenticate at all, or simply send the RAND and ignore the response. It does not have to activate ciphering either. The subscriber could then unknowingly be making calls or sending text messages that could be intercepted using this man-in-the-middle attack (as the false network could then route the calls back to the public telephone network).

This kind of attack is feasible but difficult to implement because you must implement a full functional base station system that provides transparency between mobile station and a GSM network. Moreover, mobile station must tune to this provided fake frequency, while more than one will be available. Current architecture could be evolved to implement a "man-in-the-middle" attack with much more effort.

5.5.2 Attack against A3/A8 – Retrieving Ki

Common implementation of A3/A8 is flawed – contains a narrow pipe. The most common implementation of the A3 and A8 algorithms is rolled into a single algorithm, COMP128, which generates the 64-bit Kc and the 32-bit SRES from the 128-bit RAND and the 128-bit Ki input. This algorithm is seriously flawed, in that carefully chosen values for the input RAND will provide enough information to determine the Ki in significantly less than the ideal number of attempts (a brute force on the order of 2^{128} values). The flaw exists in the fact that in the second round of the algorithm, a narrow pipe exists (such that individual bytes in isolated groups of 4 bytes in the second round output depend only on unique groups of 4 bytes in the input, 2 of which are in the Ki, 2 are in the RAND)), and thus a collision attack can be performed.
Earlier attacks based on repeated 2R attacks could typically crack a SIM in approximately 2^{17} RANDs. Dejan Kaljevic has written a utility (Sim Scan) which uses 2R, 3R, 4R and 5R attacks to obtain various bytes of the Ki, which he estimates can recover the Ki in around $2^{13} – 2^{15}$ RAND values on average. Ki can be extracted within an hour but this is not a serious attack, as it requires physical access to the SIM (and the PIN, if one is used). It is only useful for cloning SIM cards.

5.5.3 Over the air cracking of Ki

The previous mentioned flaws combined can result in a more serious attack. The same attack could be done over-the-air, using a fake over powered Base Station and bomb the target Mobile Station with challenges and retrieve the Ki. Again the MS has to be available to the attacker over the air for the whole time it takes to conduct the attack. Time to recover the Ki in this method would be around 2^{17} RANDs / 4 RAND/s = 2^{15} seconds or 9 hours. The subscriber would be unaware of such an attack though the fact that the battery of the phone has run out slightly quicker than usual might make him suspicious. The attack can also be performed in parts: instead of performing an eight-hour attack, the attacker could tease the phone for twenty minutes every day.

After retrieving Ki, a GSM phone could be cloned and interception could be feasible using our implementation while the attacker could be able to monitor challenges requests – network authentication procedure –and generate Kc using A5 algorithm.

.

This attack will work on any GSM phone, without any previous access to the phone (or even knowledge of the IMSI) – a random phone's TMSI can be chosen by monitoring radio traffic. Being an over-the-air attack, it can even be performed from many kilometers away.

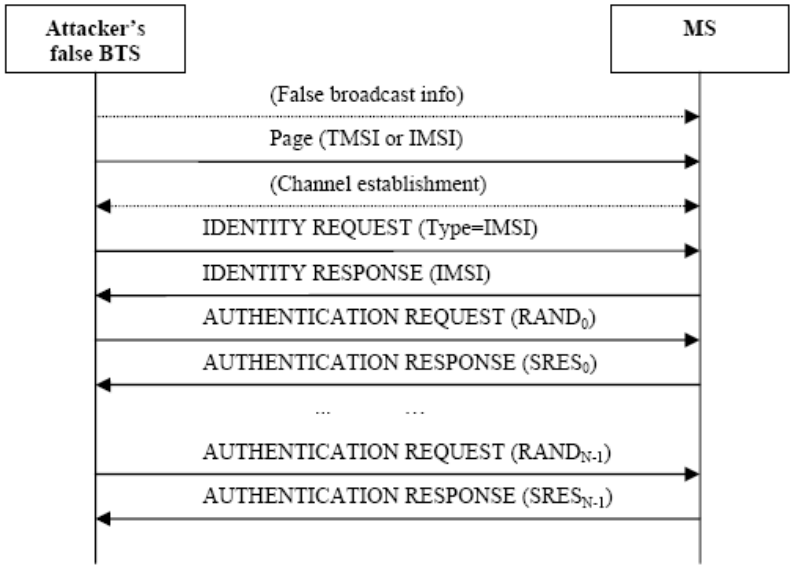


Figure 5.5-1: Over the air cracking of Ki

5.5.4 Brute-Force Attack against A5

A real-time brute-force attack against the GSM security system is not feasible, as stated above. The time complexity of the attack is 2^54 (2^64 if the ten bits were not zeroed out). This requires too much time in order to be feasible in eavesdropping on GSM calls in real time. It might be possible to record the frames between the MS and the BTS and launch the attack afterwards though.

Table 5.5-1: Brute-force key search times for various key sizes

Key length in bits	32	40	56	64	128
Time required to test all possible keys	1.19 hours	12.7 days	2291 years	584542 years	10.8 x 10^24 years

The common implementation of A3/A8, COMP128 has yet another "flaw". When generating the 64-bit Kc, it always sets the least significant 10 bits of the Kc to 0. This effectively reduces the strength of the data ciphering algorithm to 54 bits (a reduction by a factor of 1024), regardless of which ciphering algorithm is used. This same deliberate weakening is also present in the other algorithm of choice, COMP128v2.

Table 5.5-2: Number of machines required to search a key space in a given time

Key length in bits	**1 day**	**1 week**	**1 year**
40	13	2	-
56	836788	119132	2.291
64	2.14x10^8	3.04x10^6	584542
128	3.9x10^27	5.6x10^26	10.8x10^24

5.5.5 Passive Ciphertext-Only Cryptanalysis of GSM A5/1

In the GSM system, like all digital radio communications systems, forward error correction (FEC) is used over the radio link to assist in the correction from errors caused by noise or signal fading as we described on previous chapter.

The problem in GSM is that ciphering occurs after FEC, meaning the redundant stream of bits is then modulo-2 (XORed) added with the ciphering stream, meaning the known redundancy patterns could be used to assist in a crypt-analytical attack.

Of course it is far more complicated than that; in GSM the data is interleaved over many blocks and the coding on most encrypted channels is a ½ rate convolutional code generated by polynomials with m from 5 to 7. Furthermore, on voice channels, only certain bits are protected with the convolutional code (unequal error protection).

The A5/1 output is based on the modulo-2 summed output of 3 LFSRs whose clock inputs are controlled by a majority function of certain bits in each LFSR. However, Alex Biryukov, Adi Shamir and David Wagner [1][2] demonstrated in a paper that A5/1 could be cracked in about 2 seconds on a typical PC (however large pre-computed tables are required, amounting to about 2^{48} bytes or 4 disks of 73G each).

The attack exploits flaws in the algorithm when storing these tables utilizing a combination of what has been learnt through statistical analysis of the states the algorithm steps through, as well as exploiting the poor single-bit taps used to control the LFSR clocks. Following table, **Table 5.5-3**, summarizes the results of their work.

Table 5.5-3: Three possible tradeoff points in the attacks on A5/1

Attack type	Preprocessing steps	Available data	Number of 73GB disks	Attack time
Biased Birthday attack (1)	2^{42}	2 minutes	4	1 second
Biased Birthday attack (2)	2^{48}	2 minutes	2	1 second
Random Subgraph attack	2^{48}	2 seconds	4	minutes

Chapter 6

Conclusions

Contents

6.1 Summary

In this study we developed a system for *monitoring and measuring mobile GSM telephony signals,* the most common system worldwide for wireless personal digital communications, analyzing all aspects of such a system and its operation.

The current implementation of the system has set the basic architecture towards an integrated prototype device for GSM monitoring and interception. We have set the hardware and software requirements of such a system and partly implemented its basic architecture. However, there are some open problems that have to be overcome. Actually, hardware restrictions and constrains appoints current implementation incomplete.

At the beginning, we described the architecture of GSM telecommunication system, the basic components and devices it consists of and how they communicate each other. We concentrated our interest at the structure of the *air-interface* of GSM giving a detailed description and analysis of its architecture and operation, since it comprises a significant aspect of this study.

Subsequently, we set the requirements of our architecture and gave a thorough description of the modules implemented and the way they employed. We analyzed most of the characteristics and performed a custom development, from application layer to physical layer, fully based on ETSI standards that govern GSM technology. Together, we accented the main restrictions set by the hardware making real operation complicated.

Afterwards, we performed an extended analysis of the GSM security model showing the way our implemented system can be used for possible interception attacks. We presented that the security mechanisms specified in the GSM standard make it a secure cellular telecommunications system. The use of authentication, encryption, and temporary identification numbers ensures the privacy and anonymity of the system's users, as well as safeguarding the system against fraudulent use. However, the GSM security model is broken on many levels and is thus vulnerable to numerous attacks targeted at different parts of an operator's network.

Further more, the secretly designed security algorithms incorporated into the GSM system have been proven faulty. The A5 algorithm used for encrypting the over-the-air transmission channel is vulnerable even against cipher-text only attacks and the intentionally reduced key space is small enough to make a brute-force attack feasible as well. The COMP128 algorithm used in most GSM networks as the A3/A8 algorithm has been proved faulty so that the secret key Ki can be reverse-engineered over-the-air through a chosen challenge attack in approximately ten hours.

All this means that, with the proposed architecture, if somebody wants to intercept a GSM call, he could do so. The required resources depend on the attack chosen. The current GSM standard and implementation enables both subscriber identity cloning and call interception. Although the implementation of cloning or call interception is a little bit more difficult, due to the digital technology that is used the threat is still very real and the implementation feasible.

6.2 Extensions and Future Work

The architecture described in **Chapter 4** and implemented, suffers from various technical problems that have to be corrected in a next version. Specially, hardware limits prevents the system from working on real world while software meets absolutely the requirements that have been set.

One of the basic problems that exist in hardware is, as mentioned in previous chapter, synchronization. Synchronization between mobile and the base station is essential for the successful operation of a mobile radio. The subsistence of LAN I/O interface for controlling the hardware introduces a meaningful delay that makes system incomplete and partial. Moreover, currently used hardware daughterboards bring in another set of working problems. The slow rate of read/write operations on hardware registers comprises another significant issue for completing a full functional system. As we note, communication delays between host and hardware must be totally eliminated.

An interesting study for potential future work is the confrontation of the above problems and the evaluation of proposed interception attacks. There are many ways someone can advance the current system to a better design, keeping the software architecture untouched or making small additions. First of all, since the interface between software and hardware is based on LAN I/O and TCP protocol, many other hardware kits with better performance could be attached. Another interesting

optimization is the use of an extra daughterboard, comprising a microcontroller and a FPGA module. This can be used as an addition to the current hardware and optimize the hardware control procedures reducing delays. Moreover, some of the software modules could be embedded on the microcontroller evolving to a best performance system. Still, the presence of a workstation is necessary for logging and post processing captured data and high performance real time analysis.

Finally, the ideal solution regarding the hardware is the use of a complete development system for mobile phones. Many of companies, like *Analog Devices*, that put in the mobile telephony industry and develop telecommunication products, offers a wide range of integrated development kits for mobile phone development. Unfortunately, these kits are available only to telecommunication manufactures and are very difficult for anyone to obtain one.

References and Bibliography

[1] A. Biryukov, A. Shamir, “Real time cryptanalysis of the alleged A5/1 on a PC”, preliminary draft, December 1999.

[2] A. Biryukov, A. Shamir, D. Wagner, "Real time cryptanalysis of A5/1 on a PC", in FSE 2000, LNCS No. 1978, Springer Verlag, Berlin, 2000.

[3] Biham, Barkan and Keller, “Instant Ciphertext- Only Cryptanalysis of GSM Encrypted Communication” - http://www.cs.technion.ac.il/~biham/

[4] Briceno, Goldberg & Wagner, “An implementation of the GSM A3A8 algorithm. (Specifically, COMP128.)” - http://www.mirrors.wiretapped.net/security/cryptography/hashes/a3a8/a3a8.c

[5] Goldberg, Briceno & Wagner, “A pedagogical implementation of the GSM A5/1 and A5/2 ‘voice privacy’ encryption algorithms” http://kiwibyrd.chat.ru/gsm/a512.htm

[6] GSM 02.09 - Digital cellular telecommunications system (Phase 2+); Security aspects (GSM 02.09 version 8.0.1 Release 1999)

[7] GSM 03.20 - Digital cellular telecommunications system (Phase 2+); Security related network functions (GSM 03.20 version 8.1.0 Release 1999)

[8] GSM 04.08 - Digital cellular telecommunications system (Phase 2+); Mobile radio interface layer 3 specification (GSM 04.08 version 6.11.0 Release 1997)

[9] GSM 05.02 - Digital cellular telecommunications system (Phase 2+); Multiplexing and multiple access on the radio path (GSM 05.02 version 8.5.1 Release 1999)

[10] GSM 05.03 - Digital cellular telecommunications system (Phase 2+); Channel coding (GSM 05.03 version 8.5.1 Release 1999)

[11] GSM 05.08 - Digital cellular telecommunications system (Phase 2+); Radio subsystem link control (GSM 05.08 version 8.5.0 Release 1999)

[12] GSM 06.10 - Digital cellular telecommunications system (Phase 2+);Full Rate Speech; Transcoding(3GPP TS 06.10 version 8.2.0 Release 1999)

[13] Gunnar Heine, “GSM Networks - Protocols, Terminology and Implementation”, Artech House, 1998

[14] J. R. Rao, P. Rohatzi and H. Scherzer, “Partitioning Attacks: Or How to Rapidly Clone Some GSM Cards”, IBM Watson Research Center, in 2002 IEEE Symposium on Security and Privacy, Oakland, CA, May 2002.

[15] Jorg Eberspacher, Hans-Jorg Vogel, “GSM, Switching, Services and Protocols”, Wiley, John & Sons, 1999

[16] Marc Kahabka, “GSM Pocket Guide vol. 2”, Acterna Eningen GmbH

[17] Raymond Steele, Chin-Chun Lee, Peter Gould, “GSM, cdmaOne and 3G Systems”, Wiley, John & Sons, 2001

[18] Robert H. Morelos-Zaragoza, “The Art of Error Correcting Coding”, Wiley, John & Sons, 2002

[19] Schneier, B., “Applied Cryptography,” J. Wiley & Sons, 1994.

[20] Theodore S. Rappaport, “Wireless Communications, principle and practice”, Prentice Hall, 2002

[21] www.comblock.com

[22] www.etsi.org

[23] www.gsmworld.com

[24] www.jya.com

Appendix I

COM-3005 CELLULAR BAND [800-1000 MHz] RECEIVER

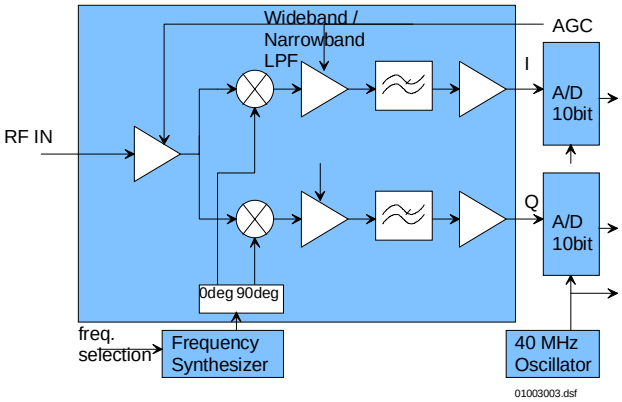
Key Features

- 800 – 1000 MHz receiver, designed for use in cellular and unlicensed ISM bands.
- Sensitivity: -56 dBm RF input for full scale 10-bit output samples.
- Built-in RF AGC, 70 dB dynamic range.
- Low phase-noise frequency synthesizer can be tuned over entire range by steps of 100, 31.25 or 25 KHz.
- 8 preset frequencies for fast (<2ms) local oscillator frequency tuning.
- Selectable internal / external 10 MHz frequency reference for the frequency synthesizer.
- Dual 10-bit Analog-to-Digital converters, 40 Msamples/s.
- Two baseband filtering options:
  - Narrow-band (<300 KHz)
  - Wideband applications (< 20 MHz).
- SMA connectors. Single 5V supply. Connectorized 3”x 3” module for ease of prototyping.

For the latest data sheet, please refer to the **ComBlock** web site: www.comblock.com/download/com3005.pdf. These specifications are subject to change without notice.

For an up-to-date list of **ComBlock** modules, please refer to www.comblock.com/product_list.htm .

Block Diagram



Electrical Interface

Inputs / Outputs

Inputs	Definition
RF_IN	800 - 1000 MHz J3 SMA male connector. 50 Ohm impedance. Receiver sensitivity: -56 dBm at RF input for full scale signal at A/D converter. Maximum input (operating): -5 dBm Maximum input (no damage): +10 dBm AGC range: 70 dB.
EXT_REF_CLK	External 10 MHz frequency reference for frequency synthesis. Sinewave, clipped sinewave or squarewave. Minimum level 0.5Vpp. Maximum level: 3.3Vpp. J7 SMA male connector.
Digital Output Signals	**Definition**
DATA_I_OUT[9:0]	In-phase baseband signal. 10-bit digital samples. 40 Msamples/s. Unsigned.
DATA_Q_OUT[9:0]	Quadrature baseband signal. 10-bit digital samples. 40 Msamples/s. Unsigned.
CLK_OUT	Digital clock. 40 Msamples/s. Read the samples at the rising edge of CLK_OUT.
ADC_CLK_OUT	Same as CLK_OUT.
AGC_IN	Pulse-width modulated signal to control the RF to baseband gain. The higher the mean value, the smaller the RF receiver gain.
Control Lines	**Definition**
PLL_STROBE	Low-voltage (3.3V / 0V) TTL input control. Used to increment the modulo- N_{freq} frequency pointer (where N_{freq} is defined in Register 35) RF frequency 0 ->RF frequency 1 ->RF frequency 2 ->RF frequency 0 > etc… Rising edge triggered. Minimum pulse width: 10 µsec. Connector J6 Pin A3.
Serial Monitoring & Control	DB9 connector. 115 Kbaud/s. 8-bit, no parity, one stop bit. No flow control.
Power Interface	4.75 – 5.25VDC. Terminal block. Power consumption is 180mA typ.

Configuration (via Serial Link / LAN)

Complete assemblies can monitored and controlled centrally over a single serial or LAN connection.

The module configuration parameters are stored in non-volatile memory. All control registers are read/write.

Parameters	Configuration
Lower-band RF synthesizer frequency	Valid range 800 MHz – 1000 MHz, steps 100 KHz. Expressed in Hz. REG0: bit 7:0 (LSB) REG1: bit 15:8 REG2: bit 23:16 REG3: bit 31:24 (MSB)
External/Internal frequency reference	0 = internal 1 = external. REG4 bit 0
External controls enabled/disabled	Enable or disable the PLL_STROBE external control on the J6 connector. 0 = external control disabled 1 = external control enabled REG6: bit 1
Step size selection	Chose between 100, 31.25 or 25 KHz step size. 00 = 100 KHz step 01 = 31.25 KHz step 10 = 25 KHz step REG6 bits 4-3.
Frequency selection	Use to switch local oscillator frequency among preselected values. Range 0 through 7 REG6 bits 7-5.
RF frequency 1	Preselected frequency 1. Same format as RF frequency 0. REG7: bit 7:0 (LSB) REG8: bit 15:8 REG9: bit 23:16 REG10: bit 31:24 (MSB)
RF frequency 2	Preselected frequency 2. Same format as RF frequency 0. REG11: bit 7:0 (LSB) REG12: bit 15:8 REG13: bit 23:16 REG14: bit 31:24 (MSB)
RF frequency 3	Preselected frequency 3. Same format as RF frequency 0. REG15: bit 7:0 (LSB) REG16: bit 15:8 REG17: bit 23:16 REG18: bit 31:24 (MSB)
RF frequency 4	Preselected frequency 4. Same format as RF frequency 0.

	REG19: bit 7:0 (LSB) REG20: bit 15:8 REG21: bit 23:16 REG22: bit 31:24 (MSB)
RF frequency 5	Preselected frequency 5. Same format as RF frequency 0. REG23: bit 7:0 (LSB) REG24: bit 15:8 REG25: bit 23:16 REG26: bit 31:24 (MSB)
RF frequency 6	Preselected frequency 6. Same format as RF frequency 0. REG27: bit 7:0 (LSB) REG28: bit 15:8 REG29: bit 23:16 REG30: bit 31:24 (MSB)
RF frequency 7	Preselected frequency 7. Same format as RF frequency 0. REG31: bit 7:0 (LSB) REG32: bit 15:8 REG33: bit 23:16 REG34: bit 31:24 (MSB)
Number of RF frequencies N_{freq} in the scanning list	Each time a PLL_STROBE pulse is received, the frequency pointer increments modulo N_{freq}. N_{freq} is in the range 1 – 8. REG35: bit 7:0.

Note: Fine frequency tuning (down to Hz precision) is typically implemented digitally at the demodulator. See digitaly demodulator specifications (COM-1001, 1011, 1018, etc) for details.

Monitoring (via Serial Link / LAN)

Parameters	Monitoring
Version	Returns '3005-A or B' when prompted for version number.

Operations

Internal vs External frequency reference for frequency synthesizer

The RF local oscillator frequency generated by the frequency synthesizer is frequency-locked onto a 10 MHz reference clock. The source of this 10 MHz reference clock (internal versus external) is user-selected by software commands.

In order to use the external frequency reference, connect a 10 MHz sinewave, clipped sinewave or square wave to the SMA connector J7. Then select external frequency reference by software command from the ComBlock control center.

In order to use the internal frequency reference, either physically disconnect the external 10 MHz signal at SMA connector J7, or place the external input signal in high impedance mode. Then select internal frequency reference by software command from the ComBlock control center.

Test Points

Test points are provided for easy access by an oscilloscope probe.

Test Point	Definition
TP1	Baseband signal, I-channel, at A/D converter input. The nominal amplitude is 1Vpp when the AGC loop is closed with the following demodulator (COM-1001, COM-1011, or equivalent).
TP2	Baseband signal, Q-channel, at A/D converter input. Nominal amplitude is 1Vpp when the AGC loop is closed.
PLL_LOCK	Frequency synthesizer PLL lock status. Active low: '0' when locked.
PLL_REF	Reference clock (10 MHz external or 20 MHz internal)

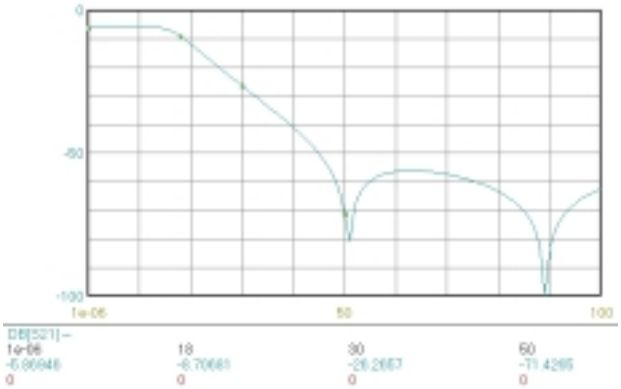
Performance

Internal Clock Reference

The internal crystal has a stability of +/- 50 ppm.

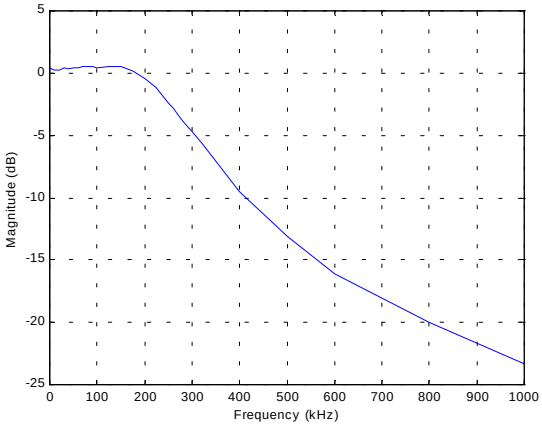
Low Pass Filter

Each A/D converter is preceeded by a 4th order elliptic low-pass filter. The 3 dB cutoff frequency for model COM-3005-B (wideband applications) is 20 MHz.



COM-3005-B baseband low-pass filter frequency response. Span 100 MHz, 10dB/div.

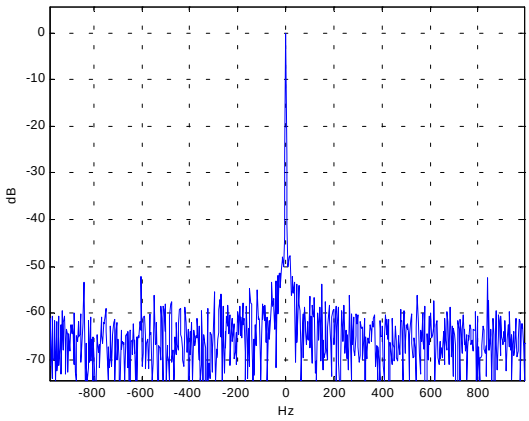
The 3 dB cutoff frequency for model COM-3005-A (narrow band applications) is 265 KHz. In-band ripple within +/- 150 KHz is less than +/- 0.1 dB.



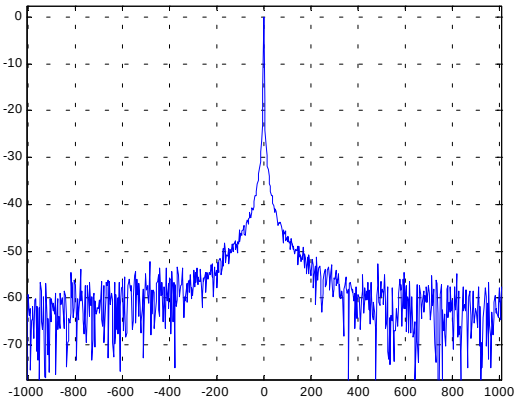
COM-3005-A baseband low-pass filter frequency response. Span 1 MHz, 5dB/div.

Phase noise

Typical phase noise performances are:
-50 dBc @ 100 Hz away from the carrier
-65 dBc @1 KHz
-65 dBc @10 KHz
-100 dBc @ 100 KHz



Phase noise, 800 MHz, 200 Hz/division span, 3Hz resolution bandwidth. Internal reference clock. Measured from RF in to digital samples out.

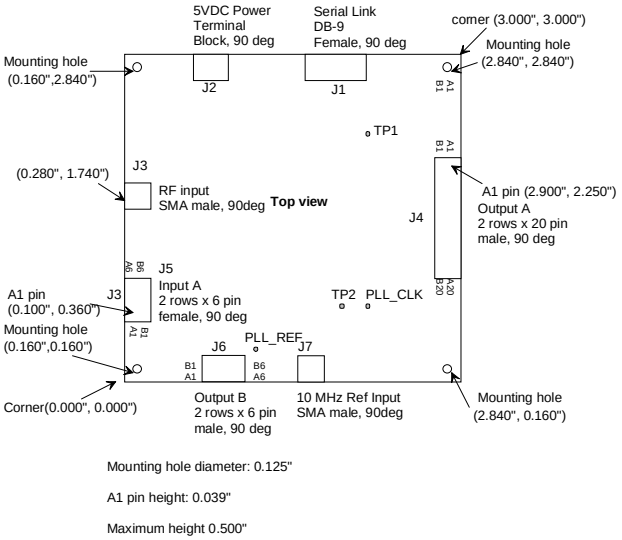


Phase noise, 1.0GHz, 200 Hz/division span, 3Hz resolution bandwidth. Internal reference clock. Measured from RF in to digital samples out.

Spectral spurious lines are at –60 dBc or lower.

LO frequency switching time: <2 ms

Mechanical Interface



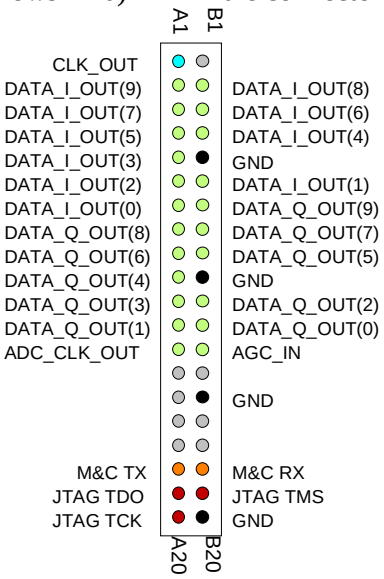
Mounting hole diameter: 0.125"

A1 pin height: 0.039"

Maximum height 0.500"

Output Connector J4

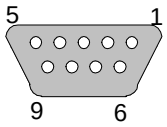
40-pin (2 rows x 20) 2mm male connector.



Pinout

Serial Link J1

The DB-9 connector is wired as data circuit terminating equipment (DCE). Connection to a PC is over a straight-through cable. No null modem or gender changer is required.

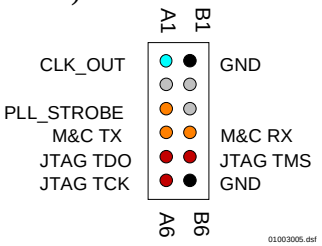


DB-9 Female

2 Transmit
3 Receive
5 Ground

Connector J6

12-pin (2 rows x 6) 2mm male connector.



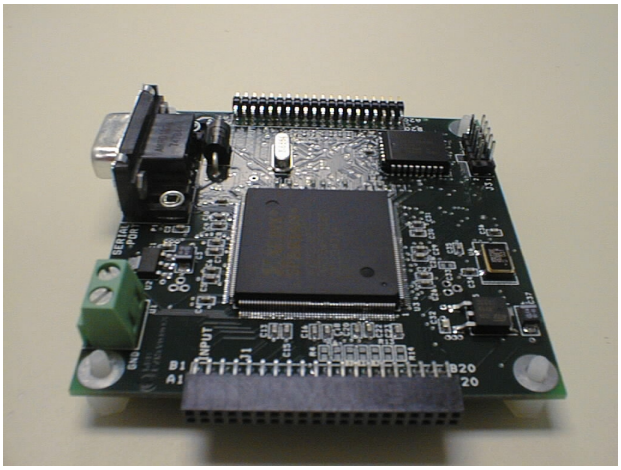
COM-1027 FSK/MSK/GFSK/GMSK DIGITAL DEMODULATOR

Key Features

- Demodulator for continuous phase FSK (CPFSK) and its derivatives:
  - Minimum shift keying (MSK)
  - Gaussian frequency shift keying (GFSK)
  - Gaussian minimum shift keying (GMSK)
- Programmable 2-, 4-, 8-ary FSK
- Programmable modulation index h [0.125 to 4]
- Programmable data rates up to 30/20/10 Mbps. (8-, 4-, 2-ary FSK).
- Coherent demodulator for better BER performances.
- **ComScope** –enabled: key internal signals can be captured in real-time and displayed on host computer.
- Connectorized 3”x 3” module for ease of prototyping. Standard 40 pin 2mm dual row connectors (left, right). Single 5V supply with reverse voltage and overvoltage protection. Interfaces with 3.3V LVTTL logic.

For the latest data sheet, please refer to the **ComBlock** web site: www.comblock.com/download/com1027.pdf.
These specifications are subject to change without notice.

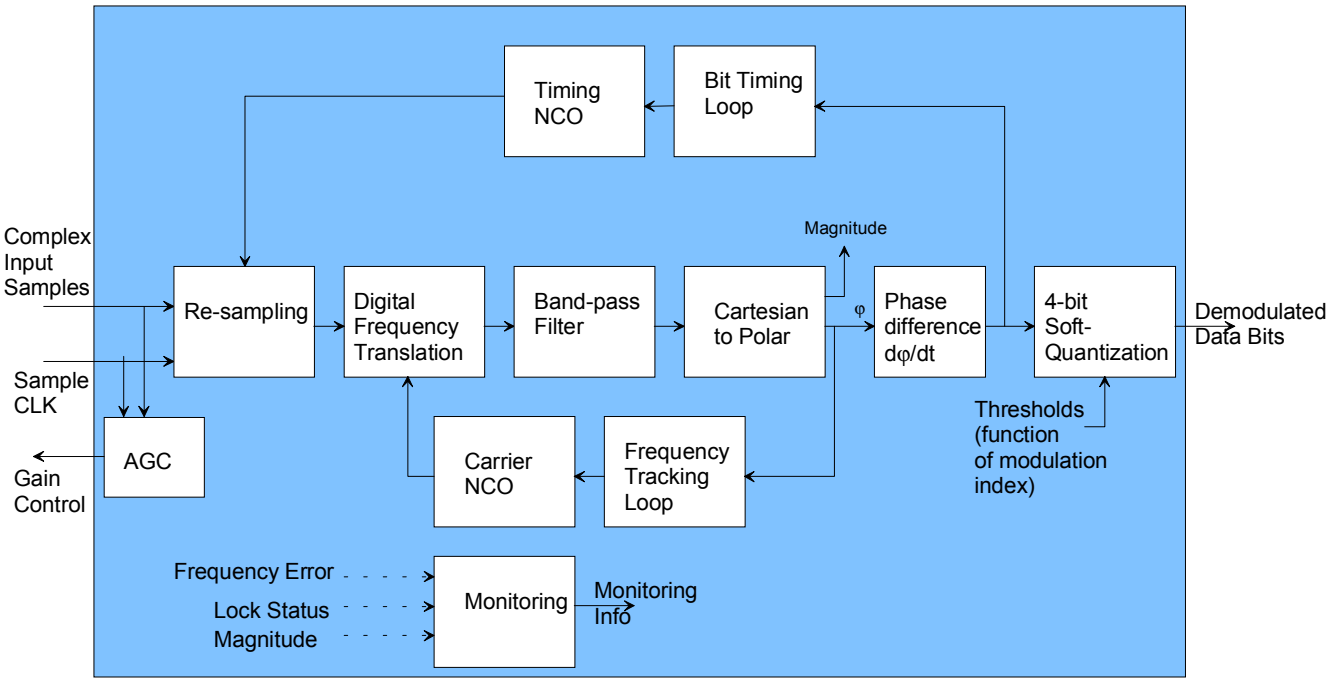
For an up-to-date list of **ComBlock** modules, please refer to www.comblock.com/product_list.htm .



Typical Applications

- GSM:
  - GMSK, modulation index h=0.5, BT = 0.3, bit rate: 270.833 Kbps, channel spacing: 200 KHz
- Bluetooth:
  - GFSK, modulation index h = 0.32, BT = 0.5
- DECT:
  - GFSK, BT = 0.5, bit rate: 1.152 Mbps, channel spacing: 1.728 MHz

Block Diagram



Electrical Interface

Input Module Interface	Definition
DATA_I_IN[9:0]	Modulated input signal, real axis. 10-bit precision. Format: 2's complement or unsigned. Unused LSBs are pulled low.
DATA_Q_IN[9:0]	Modulated input signal, imaginary axis. 10-bit precision. Same format as DATA_I_IN. Unused LSBs are pulled low.
SAMPLE_CLK_IN	Input signal sampling clock. One CLK-wide pulse. Read the input signal at the rising edge of CLK when SAMPLE_CLK_IN = '1'. The minimum input sampling rate is 8 samples per symbol. Sampling above 16 samples per symbol may cause aliasing whereby adjacent channels may interfere with the main signa. Samples can be consecutive. For example, SAMPLE_CLK_IN can be fixed at '1' to indicate that new input samples are provided once per CLK_IN clock period. Signal is pulled-up.
AGC_OUT	Output. When this demodulator is connected directly to an analog receiver, it generates a pulse-width modulated signal to control the analog gain prior to A/D conversion. The purpose is to use the maximum dynamic range while preventing saturation at the A/D converter. 0 is the maximum gain, +3V is the minimum gain.
CLK_IN	Input reference clock for synchronous I/O. DATA_x_IN and SAMPLE_CLK_IN are read at the rising edge of CLK_IN. Maximum 40 MHz.

Output Module Interface	Definition
DATA_OUT[3:0]	4-bit soft-quantized demodulated bits, real axis. Unsigned representation: 0000 for maximum amplitude '0', 1111 for maximum amplitude '1'.
BIT_CLK_OUT	Demodulated bit clock. One CLK-wide pulse. Read the output signal at the rising edge of CLK when BIT_CLK_OUT = '1'.
RX_LOCK	'1' when the demodulator is locked, '0' otherwise.
CLK_OUT	40 MHz output reference clock. Generated by dividing the internal processing clock: $\mathbf{f_{clk}}$ /2
Serial Monitoring & Control	DB9 connector. 115 Kbaud/s. 8-bit, no parity, one stop bit. No flow control.
Power Interface	4.75 – 5.25VDC. Terminal block. Power consumption is approximately proportional to the symbol clock rate (f_{symbol_clk}). The maximum power consumption is 650mA.

Important: I/O signals are 0-3.3V LVTTL. Inputs are NOT 5V tolerant!

Configuration (via Serial Link / LAN)

Complete assemblies can be monitored and controlled centrally over a single serial or LAN connection.

The module configuration parameters are stored in non-volatile memory. All control registers are read/write.

This module operates at a fixed internal clock rate $\mathbf{f_{clk}}$ of 80 MHz.

Most processing is done at the sampling rate / $\mathbf{f_{sample\_clk}}$ = 8 * symbol rate.

Parameters	Configuration
Symbol rate (f_{symbol_clk})	24-bit signed integer expressed as fsymbol rate * 2^{24} / $\mathbf{f_{clk}}$. $\mathbf{f_{clk}}$ is 80 MHz. The maximum symbol rate is 10 Msymbols/s (0x1FFFFF). The data rate is 1x, 2x or 3x the symbol rate depending on the M-ary number set in REG8. REG0 = bit 7-0 (LSB) REG1 = bit 15 – 8 REG2 = bit 23 – 16 (MSB)
Center frequency ($\mathbf{f_c}$)	Nominal center frequency. This value is substracted from the received signal actual center frequency. 24-bit signed integer (2's complement representation) expressed as $\mathbf{f_c} * 2^{24} / (8*f_{symbol\_clk})$ Safe range to avoid aliasing: +/- 2 * f_{symbol_clk} Note: the definition of the center frequency is different for the COM-1028 modulator and this demodulator. REG3 = bits 7 – 0 REG4 = bits 15 – 8 REG5 = bits 23 - 16
Inverse Modulation Index **1/h**	1/(Modulation index **h**). Format 8.8 Thus, 0x0200 represents the inverse of a modulation index of 0.5. (MSK or GMSK modulation imply h = 0.5). Valid range for **1/h**: 0.125 – 4 REG6: bits 7:0 LSB REG7: bit 15:8: MSB
M-ary number	Size of the symbol alphabet: 00 = 2-ary, 2-FSK, M=2 01 = 4-ary, 4-FSK, M=4 10 = 8-ary, 8-FSK, M=8 REG8 bits 1-0
Input sample format	0 = 2's complement 1 = unsigned REG8 bit 2
Spectrum inversion	Invert Q bit. 0 = off 1 = on REG8 bit 3
Freeze monitoring data	As the monitoring data is constantly changing, it is important to be able to prevent changes while reading a multi-byte parameter. Write a zero in bit 7 to freeze the monitoring data prior to reading it. Write a one to re-enable the update. REG8 bit 7

Baseline configurations can be found at www.comblock.com/tsbasic_settings.htm and imported into the ComBlock assembly using the ComBlock Control Center File | Import menu.

Monitoring (via Serial Link / LAN)

Monitoring registers are read-only.

Parameters	Monitoring
Carrier frequency offset (fcdelta)	Residual frequency offset with respect to the nominal carrier frequency. 24-bit signed integer (2's complement) expressed as fcdelta * 2^{24} / (8*fsymbol rate). REG12 = bit 7 – 0 REG13 = bit 15 – 8 REG14 = bit 23 – 16
Received signal magnitude after channel filtering	8-bit unsigned REG15 bit 7-0.
AFC lock status	0 = unlocked 1 = locked REG16 bit 0
Signal power detection	0 = below threshold 1 = signal power detection REG16 bit 1
Option o / Version v	Returns '1027ov' when prompted for option o and version v numbers.

ComScope Monitoring

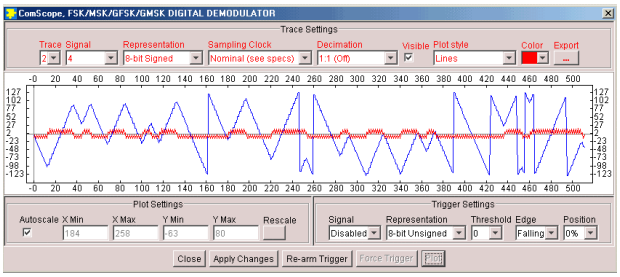
Key internal signals can be captured in real-time and displayed on a host computer using the ComScope feature of the ComBlock Control Center. The COM-1027 signal traces and trigger are defined as follows:

Trace 1 signals	Format	Nominal sampling rate	Buffer length (samples)
1: Input I signal	8-bit signed or unsigned. (8MSB/10)	Input sampling rate	512
2: Cartesian-to-Polar conversion: phase	8-bit signed (8MSB/10)	8 samples /symbol	512
3: Cartesian-to-Polar conversion: magnitude	8-bit signed (8MSB/14)	8 samples /symbol	512
4: Filtered Phase difference at optimum sampling instant	8-bit signed (8MSB/13)	1 sample /symbol	512
5: Recovered carrier frequency offset. Resolution is fsymbol/32.	8-bit signed (8MSB/24)	8 samples /symbol	512
Trace 2 signals	**Format**	**Nominal sampling rate**	**Capture length (samples)**
1: Input Q signal	8-bit signed or unsigned. (8MSB/10)	Input sampling rate	512
2: front-end AGC	8-bit unsigned (8MSB/10)	$\mathbf{f_{clk}}$ (80 MHz)	512
3: I signal after elastic buffer, interpolation and resampling at 8 samples/symbol	8-bit signed (8MSB/12)	8 samples /symbol	512
4: Phase difference	8-bit signed (8MSB/10)	8 samples /symbol	512
5: Recovered phase, after scaling for modulation index (i.e. prior to soft quantization)	8-bit signed (8MSB/14)	1 samples /symbol (optimum sampling instant)	512
Trigger Signal	**Format**		
1: AFC lock status	Binary		
2: Signal power detection	Binary		

Signals sampling rates can be changed under software control by adjusting the decimation factor and/or selecting the $\mathbf{f_{clk}}$ processing clock as real-time sampling clock.

In particular, selecting the $\mathbf{f_{clk}}$ processing clock as real-time sampling clock allows one to have the same time-scale for all signals.

The ComScope user manual is available at www.comblock.com/download/comscope.pdf.



ComScope Window Sample: showing GMSK demodulated phase (blue) and reconstructed unfiltered symbols (red).

Operation

FSK Modulation

The FSK modulation and its derivatives (CPFSK, MSK, GMSK, GFSK) are best described by the following equations for the modulated signal s(t). The first equation describes a phase modulator, with the modulated centered around the center frequency f_c.

$$s(t) = \sqrt{\frac{2E_s}{T}}.\cos(2\pi f_c t + \theta(t) + \theta_0)$$

where

- E_s is the energy per symbol
- T is the symbol period
- f_c is the center frequency
- $\theta(t)$ is the phase modulation

The COM-1027 implements a continuous phase FSK demodulator. It assumes that there are no phase discontinuities between symbols. The CPFSK phase modulation can be described as:

$$\theta(t) = \frac{\pi h}{T}\int_0^t a_i(t)dt$$

where:

- h is the modulation index. A modulation index of 0.5 yields a maximum phase change of π/2 over a symbol.

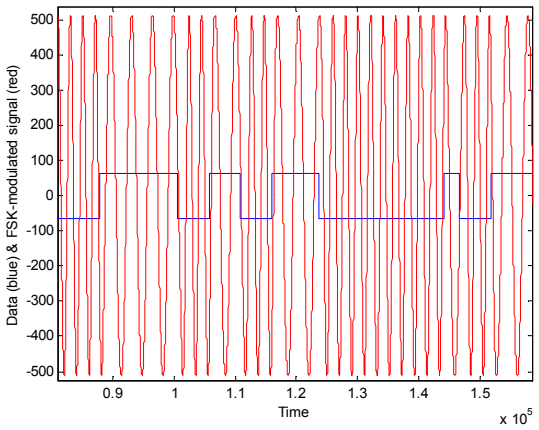
a_i are the symbols. With 2-FSK, the binary data is represented as –1 (for '0') and +1 (for '1').

M-ary Number M

The transmitted data is grouped into symbols of size 1, 2, or 3 consecutive bits. The size of the symbol alphabet is thus M = 2, 4 or 8. The symbol MSB is sent first to the DATA_OUT output.

The mapping between modulation symbol a_i and symbol alphabet is described in the table below:

Modulation symbol a_i	Symbol alphabet
-1	2-FSK '0'
+1	2-FSK '1'
-3	4-FSK "00"
-1	4-FSK "01"
+1	4-FSK "10"
+3	4-FSK "11"
-7	8-FSK "000"
-5	8-FSK "001"
-3	8-FSK "010"
-1	8-FSK "011"
+1	8-FSK "100"
+3	8-FSK "101"
+5	8-FSK "110"
+7	8-FSK "111"



Continuous FSK modulated signal example

FSK modulation is sometimes characterized by the frequency separation between symbols. The relationship between modulation index h and frequency separation is $f_{separation}$ = 0.5 h f_{symbol_clk}

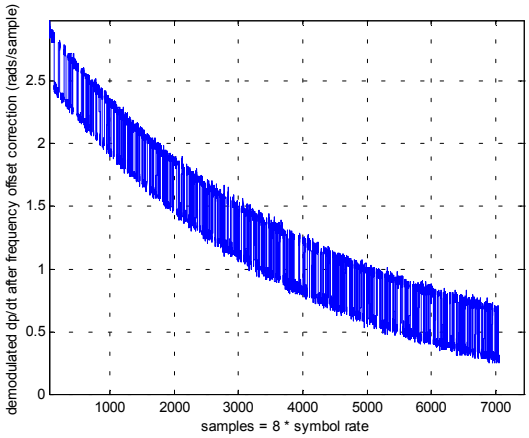
Frequency Acquisition and Tracking

The demodulator comprises an automatic frequency control (AFC) loop to acquire and track the residual frequency offset of the modulated signal.

The tracking range $f_{tracking}$ is bound by the following constraints:
abs($\mathbf{f_{tracking}}$/(4* $\mathbf{f_{symbol\_clk}}$)) + **h**/8 < 1

For example, if the modulation index **h** is 0.5, the maximum tracking range is ± 3.75 $\mathbf{f_{symbol\_clk}}$. We recommend an additional 10% implementation margin.

The AFC response time is illustrated below for an initial frequency offset of (3.37* $\mathbf{f_{symbol\_clk}}$) and modulation index h = 0.5.



AFC response to input frequency step

An AFC lock status is provided in status register REG16 and at a test point. AFC lock is declared when the residual frequency error is below $1/16^{th}$ of the symbol rate.

Bit Timing Tracking

A first order loop is capable of acquiring and tracking bit timing differences between the transmitter and the receiver, up to ± 500 ppm.

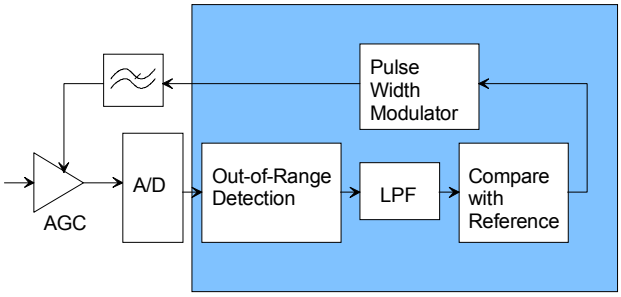
AGC

The purpose of this AGC is to prevent saturation at the external A/D converter(s) while making full use of the 10-bit A/D converter dynamic range. The principle of operations is outlined below:

(a) out-of-range at the A/D converter is detected. An out-of-range condition occurs if the quantized A/D samples are equal to either "1111111111" or "0000000000".

(b) The AGC will adjust the analog circuitry gain so that out-of-range conditions do not occur more than 1 in 64 samples in the average.

(c) The resulting gain control signal is a pulse-width modulated (PWM) signal with 10-bit precision.



AGC principle

The analog circuit shall filter this 3.3V low-voltage TTL PWM signal with a low-pass filter prior to controlling the analog gain. The PWM is randomized and its spectral distribution shifted to the higher frequencies so as facilitate the analog low-pass filter design.

The AGC loop bandwidth is typically 1 Hz when used in conjunction with COM-30xx receivers and a 40 MHz processing clock. The loop response time is assymetrical: it responds faster to a saturation condition than to a 'low signal' condition.

The gain control signal will increase if too many out-of-range conditions occur.

Implementation

The incoming samples are first stored in an elastic buffer to switch between the input clock (CLK_IN, up to 40 MHz) and the internal processing clock (CLK, $\mathbf{f_{clk}}$ = 80 MHz). All subsequence signal processing is performed at a clock rate $\mathbf{f_{clk}}$ of 80 MHz.

The incoming In-phase (I) and quadrature (Q) samples are subsampled at a rate of 8 samples per symbol under the control of the bit timing NCO.

The signal center frequency is then translated to zero to compensate for known ($\mathbf{f_c}$) and unknown frequency offsets. The known frequency offset $\mathbf{f_c}$ is under user control by means of the control registers REG3/4/5. Unknown frequency offsets are detected by the carrier tracking loop.

The resulting signal undergoes channel filtering to reject the out-of-band noise.

The coordinates for the filtered complex signal are converted from Cartesian (I,Q) to Polar (|A|, φ).

Differentiation dφ/dt reveals the modulated symbols and any residual frequency offset .

The Automatic Frequency Control (AFC) loop accumulates the differentiated phase dφ/dt. The resulting sum is used to control the carrier NCO as a first order loop. The key underlying assumption is that the transmitted data is random and balanced, i.e. contains an equal number of 0's and 1's.

Timing

Clocks

An 80 MHz internal clock $\mathbf{f_{clk}}$ is generated by frequency doubling of the 40 MHz oscillator installed on the COM-1027 board. $\mathbf{f_{clk}}$ is <u>not</u> related to the CLK_IN clock. $\mathbf{f_{clk}}$ is used for internal processing and for generating the output clock CLK_OUT = $\mathbf{f_{clk}}/2$.

Input data DATA_IN is first written into an input elastic buffer at the rising edge of CLK_IN when SAMPLE_CLK_IN = '1'.

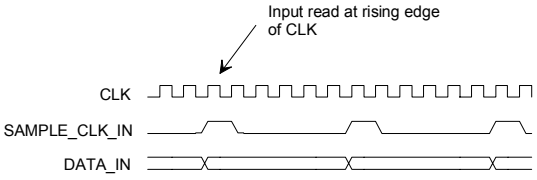
The data is read out of the input elastic buffer at the symbol rate * 1 (2-ary FSK), * 2 (4-ary FSK) or * 3 (8-ary FSK).

The input buffer size is 256 symbols.

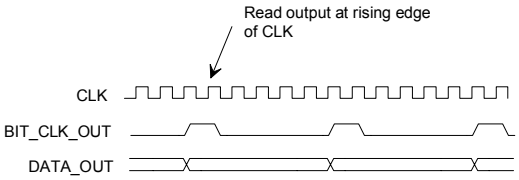
I/Os

All I/O signals are synchronous with the rising edge of the reference clock CLK_IN or CLK_OUT (i.e. all signals transitions always occur after the rising edge of clock). The maximum frequency for CLK_IN is 40 MHz. The frequency for CLK_OUT is fixed at 40 MHz ($\mathbf{f_{clk}}/2$) .

Input



Output

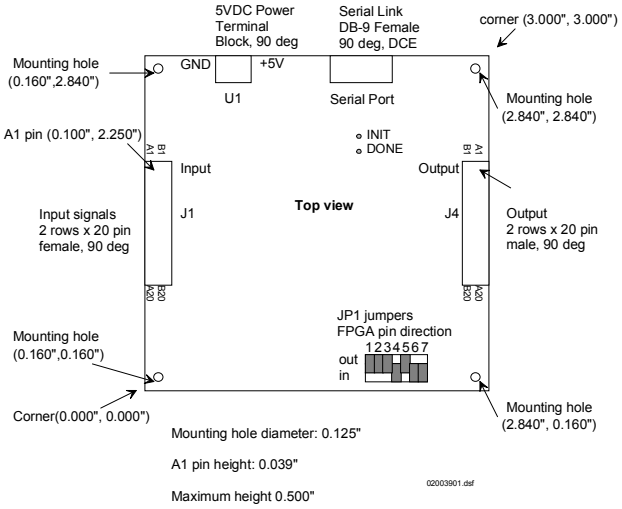


Test Points

Test points are provided for easy access by an oscilloscope probe at the J4 male connector.

Test Point	Definition
J4/A7	AFC lock status (1 = locked, 0 = unlocked)
J4/B7	Signal power detection (1 = signal power presence detected , 0 = signal power below threshold)
J4/A8	Recovered carrier (carrier NCO output MSB)
J4/B8	Recovered bit timing Compare with modulator bit timing.
J4/A9	Saturation condition detected at input samples DATA_I_IN or DATA_I_Q.
J4/B9	Internal 80 MHz clock

Mechanical Interface

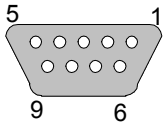


Note: All seven JP1 jumpers must be in the 'IN' location.

Pinout

Serial Link P1

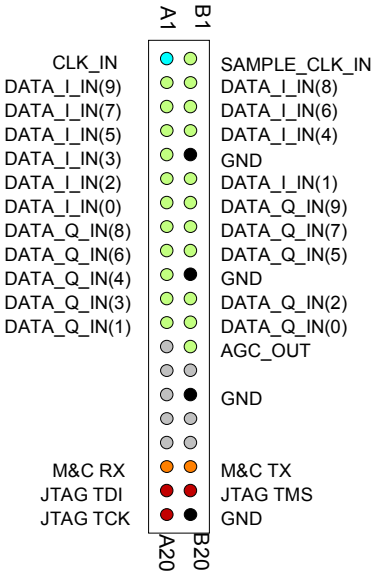
The DB-9 connector is wired as data circuit terminating equipment (DCE). Connection to a PC is over a straight-through cable. No null modem or gender changer is required.



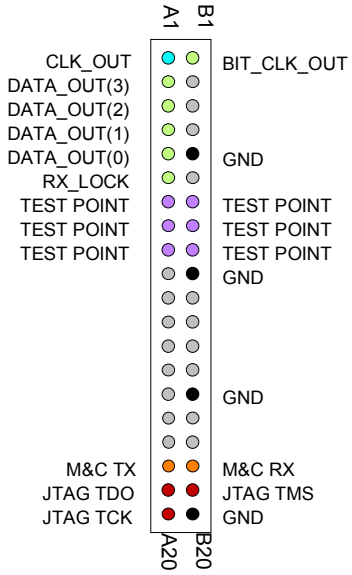
DB-9 Female

2 Transmit
3 Receive
5 Ground

Input Connector J1



Output Connector J4



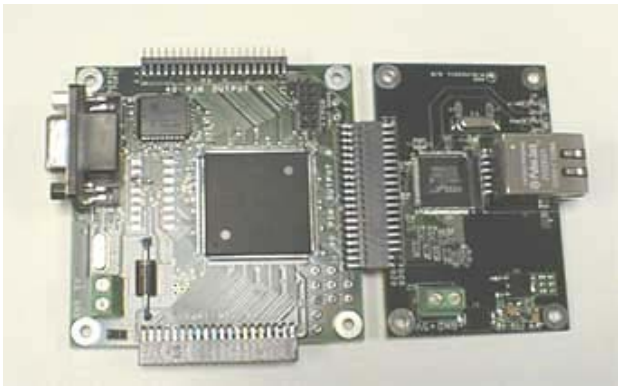
COM-5002 LAN / IP NETWORK INTERFACE

Key Features

- TCP-IP server connects ComBlock assemblies to network clients for data transfer, monitoring and control.
- Standard 100Base-Tx/10Base-T, RJ-45 connector. Autonegotiation or manual settings: 10/100 Mbit/s, full/half duplex.
- Maximum sustained throughput:
  25 Mbits/s (100Base-Tx).
  5.3 Mbits/s (10Base-T).
  Actual speed depends on host computer.
- Elastic buffering and flow-control on each transmit and receive link.
- Monitoring and control of ComBlock assemblies over LAN or serial link from a graphical user interface.
- Single 5V supply. Standard 40 pin 2mm dual row connectors (right, left)

For the latest data sheet, please refer to the **ComBlock** web site: www.comblock.com/download/com5002.pdf.
These specifications are subject to change without notice.

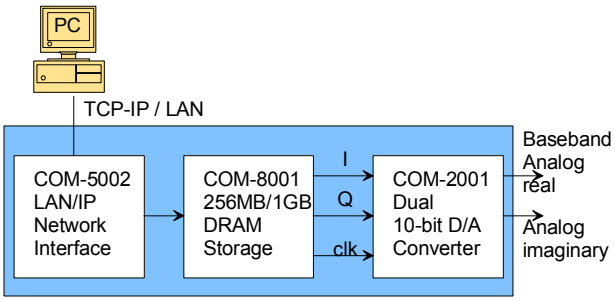
For an up-to-date list of **ComBlock** modules, please refer to www.comblock.com/product_list.htm .



Typical Applications

Arbitrary Waveform Signal Generation

Files representing binary or analog sampled signals can be uploaded through the COM-5002 to the COM-8001 SDRAM memory over the network, then played back at the selected speed. Various ComBlocks can be used to generate analog signals at baseband, 70 MHz intermediate frequency or radio-frequency.



Arbitrary waveform generator, analog baseband example

Interfaces

Input Interface	Definition
DATA_IN[7:0]	Input signal. The input width is user programmable as a function of the data source.
SAMPLE_CLK_IN	Input signal sampling clock. One CLK_IN-wide pulse. Read the input signal at the rising edge of CLK_IN when SAMPLE_CLK_IN = '1'. Samples can be consecutive. For example, SAMPLE_CLK_IN can be fixed at '1' to indicate that new input samples are provided once per CLK_IN clock period. Signal is pulled-up.
CLK_IN	Input reference clock for synchronous I/O. DATA_IN and SAMPLE_CLK_IN are read at the rising edge of CLK_IN. Maximum 40 MHz.
Output Interface	**Definition**
DATA_OUT[7:0]	Output signal. The output width is user programmable as a function of the data sink.
SAMPLE_CLK_OUT	Output signal sampling clock. One CLK_OUT-wide pulse. Read the output signal at the rising edge of CLK_OUT when SAMPLE_CLK_OUT = '1'.
CLK_OUT	40 MHz output reference clock. (from internal oscillator).
Other Interfaces	**Definition**
LAN	4 wire. 10Base-T/100Base-TX. RJ45 connector. NIC wiring. Use standard category 5 cable for connection to a Hub. Use crossover cable for connection to a host computer.
Serial Monitoring & Control	DB9 connector. 115 Kbaud/s. 8-bit, no parity, one stop bit. No flow control.
Power Interface	4.75 – 5.25VDC. Terminal block. Power consumption is typically 350mA.

Initial Configuration (via Serial Link)

The IP address must first be configured over serial link. This network setting is saved in non-volatile memory. Once the correct network setting is configured, the Comblock Control Center and this ComBlock assembly can communicate over the intranet or internet as well as over a serial link.

Configuration (via Serial Link / LAN)

Complete assemblies can be monitored and controlled centrally over a single serial or LAN connection.

The module configuration parameters are stored in non-volatile memory. All control registers are read/write.

Undefined control registers or register bits are for backward software compatibility and/or future use. They are ignored in the current firmware version.

Parameters	Configuration
IP address	4-byte IP address. Example : 0x AC 10 01 80 designates address 172.16.1.128 The new address becomes effective immediately (no need to reset the ComBlock). REG0: MSB REG1 REG2 REG3: LSB
Reserved	REG4-19 Reserved for other network configurations. No need to write any data.
Input format	00000 = J2 input is disabled 00001 = 1-bit wide from J2 01000 = 8-bit wide from J2 11110 = test mode. Internally generated 8-bit wide periodic counting sequence (0-255) as input. J2 input is disabled. The throughput is determined by the TCP-IP client. REG20 bits 4-0
Output format	00001 = 1-bit wide 01000 = 8-bit wide REG21 bits 4-0
COM-8001 external trigger	Special use: Writing to REG22 with a '1' in bit 1 will generate a 1 CLK wide pulse on pin J3/B6. The main application is to trigger the COM-8001 file playback/download. There is no need to reset this bit to '0' prior to writing a '1'. REG22 bit 1.
10Base-T / 100Base-TX LAN selection	00 = 10Base-T 01 = 100Base-TX 10 = Auto negotiation Changes will take effect at the next power up. REG22 bits 3-2

Half / Full duplex LAN link	0 = half duplex 1 = full duplex Changes will take effect at the next power up. REG22 bit 4
Promiscuous (listen) mode	Test mode. Incoming packets are not checked for matching destination address. 0 = disabled. 1 = enabled. REG22 bit 5

Baseline configurations can be found at www.comblock.com/tsbasic_settings.htm and imported into the ComBlock assembly using the ComBlock Control Center File | Import menu.

Monitoring (via Serial Link / LAN)

Monitoring registers are read-only.

Parameters	**Monitoring**
TCP-IP connection on port 1024 (data stream)	1 = connected, 0 otherwise. REG23 bit 0
TCP-IP connection on port 1028 (Monitoring & Control)	1 = connected, 0 otherwise. REG23 bit 2
Transmit data elastic buffer empty	1 = empty, 0 otherwise REG23 bit 3
Transmit data elastic buffer full	1 = full, 0 otherwise REG23 bit 4
Receive data elastic buffer empty	1 = empty, 0 otherwise REG23 bit 5
Receive data elastic buffer more than half full	1 = more than half full, 0 otherwise REG23 bit 6
Number of bytes transmitted from LAN to digital device	Total number of bytes transmitted over data and signaling channels. 32-bit byte count. Counter rolls over when reaching 0xFFFFFFFF. REG24: bits 7-0 (LSB) REG25: bits 15-8 REG26: bits 23-16 REG27: bits 31-24 (MSB)
Number of bytes received from digital device and forwarded to LAN	Total number of bytes received over data and signaling channels. 32-bit byte count. Counter rolls over when reaching 0xFFFFFFFF. REG28: bits 7-0 (LSB) REG29: bits 15-8 REG30: bits 23-16 REG31: bits 31-24 (MSB)
Option o / Version v	Returns '5002ov' when prompted for option o and version v numbers.

As the monitoring data is constantly changing, it is important to be able to prevent changes while reading a multi-byte parameter. The monitoring data is latched upon reading register 23. Therefore, register 23 should always be read first.

IP Protocols

This module supports the following IP protocols:

- Ping
- ARP
- TCP-IP

Ping

The module responds to ping requests with size up to 470 bytes. Ping can be used to check the module response over the network. Ping can be used at any time, concurrently with other transmit and receive transactions. For example, on a Windows operating system, open the Command prompt window and type "ping –t –l 470 172.16.1.128" to send pings forever of length 470 bytes to address 172.16.1.128.

TCP-IP

As a Server, the module opens the following sockets in listening mode:

Port 1024: transmit and receive data streams

Port 1028: monitoring and control port

Operation

Concept

The COM-5002 converts a TCP-IP socket stream into a simple data stream and vice versa. On the transmit side, the COM-5002 decodes the TCP-IP protocol and extracts the data from the network client. TCP, IP and Network information, and in particular routing information, are not transmitted from one end to the other.

At the receiving end, the network client must first connect to the COM-5002 to receive data.

The COM-5002 maintains the flow-control information between the TCP-IP socket and the

input/output interfaces. For example, if the COM-5002 is connected to a COM-1001 QPSK modulator configured for 1 Mbit/s data throughput, the network client (i.e. data source) will be asked for 1 Mbit/s throughput over the TCP-IP link.

Throughput Benchmarks

The COM-5002 is capable of a sustained (average) throughput of 25 Mbits/s over 100base-Tx and 5.3 Mbit/s over 10base-T. In most cases, the sustained throughput is limited by the TCP-IP client computer and the application running on the client computer as illustrated in the one-way data transfer benchmarks below:

Throughput tests conditions	Throughput
Client: Intel Pentium 4 2.6 GHz running winsock-based console application. Direct cross-over LAN cable. No network connection. No other application running. COM-5002 configured as 'Auto Negotiation". 100Base-Tx connection.	25 Mbits/s 100Mbytes transferred in 32 seconds.
Client: Intel Celeron 766 MHz running winsock-based console application. Direct cross-over LAN cable. No network connection. No other application running. COM-5002 configured as 'Auto Negotiation". 100Base-Tx connection.	14 Mbits/s 100Mbytes transferred in 57 seconds.
Client: Intel Pentium 4 2.6 GHz running winsock-based console application. Direct cross-over LAN cable. No network connection. No other application running. COM-5002 or client computer configured as '10Base-T". 10Base-T connection.	5.36 Mbits/s 100Mbytes transferred in 149 seconds.
Client: Intel Celeron 766 MHz running winsock-based console application. Direct cross-over LAN cable. No network connection. No other application running. COM-5002 or client computer configured as '10Base-T". 10Base-T connection.	4.16 Mbits/s 100Mbytes transferred in 192 seconds.
Client: Intel Pentium 4 2.6 GHz running Java JRE-based application (ComBlock Control Center). Direct cross-over LAN cable. No network connection. No other application running. COM-5002 or client computer configured as '10Base-T". 10Base-T connection.	4.82 Mbits/s 208Mbytes transferred in 345 seconds.

Format Conversion

Parallel to serial conversion occurs at the output when a 8-bit byte received over the TCP-IP link is converted to n-bit serial, where the sample width n is selected by the user. The key rule for parallel to serial conversion is that the most significant bit (MSb) is transmitted first.

Likewise, in the serial-to-parallel conversion which occurs at the input, the first received bit is placed at the MSb position in the byte.

Client Programming

This section is intended to help designers who want to design their own client application. It can be skipped by users of ready-to-use applications such as Hyperterminal, ComBlock Control Center, etc.

In network terminology, the COM-5002 is a server. It awaits connection establishment and connection termination under the initiation of clients. It never initiate any connection establishment or termination.

An example of C-language Winsock programming for Windows OS clients is shown below. More information about Winsock programming can be found at http://msdn.microsoft.com/library/default.asp?url=/library/en-us/winsock/winsock/finished_server_and_client_code.asp

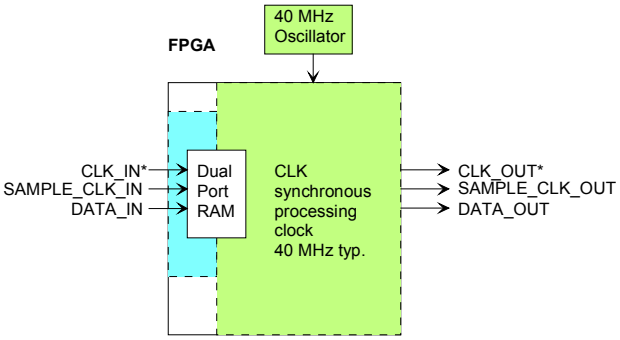
Be sure to include a reference to the Winsock2 library (WS2_32.lib) in the project release and/or debug settings.

```
#include <stdio.h>
#include "winsock2.h"

void main() {

    // Initialize Winsock.
    WSADATA wsaData;
    int iResult = WSAStartup( MAKEWORD(2,2), &wsaData );
    if ( iResult != NO_ERROR )
        printf("Error at WSAStartup()\n");

    // Create a socket.
    SOCKET m_socket;
    m_socket = socket( AF_INET, SOCK_STREAM, IPPROTO_TCP );

    if ( m_socket == INVALID_SOCKET ) {
        printf( "Error at socket(): %ld\n", WSAGetLastError() );
        WSACleanup();
        return;
    }

    // Connect to a server.
    sockaddr_in clientService;

    clientService.sin_family = AF_INET;
// insert destination address below
    clientService.sin_addr.s_addr = inet_addr( "172.16.1.128" );
// insert destination port below
    clientService.sin_port = htons(1024);

    if ( connect( m_socket, (SOCKADDR*) &clientService, sizeof(clientService) ) ==
SOCKET_ERROR) {
        printf( "Failed to connect.\n" );
        WSACleanup();
        return;
    }

    // Send and receive data.
    int bytesSent;
    int bytesRecv = SOCKET_ERROR;
    char sendbuf[32] = "Client: Sending data.";
    char recvbuf[32] = "";

    bytesSent = send( m_socket, sendbuf, strlen(sendbuf), 0 );
    printf( "Bytes Sent: %ld\n", bytesSent );

    while( bytesRecv == SOCKET_ERROR ) {
        bytesRecv = recv( m_socket, recvbuf, 32, 0 );
        if ( bytesRecv == 0 || bytesRecv == WSAECONNRESET ) {
            printf( "Connection Closed.\n");
            break;
        }
        if (bytesRecv < 0)
            return;
        printf( "Bytes Recv: %ld\n", bytesRecv );
    }

    return;
}
```

Timing

Clocks

The clock distribution scheme embodied in the COM-5002 is illustrated below.



Baseline clock architecture
Green = 40 MHz processing zone
Light blue = user defined input clock
**** indicates edge-trigger signal***

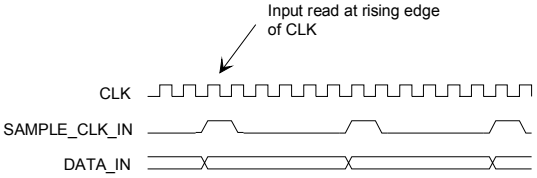
The core signal processing performed within the FPGA is synchronous with the processing clock $\mathbf{f_{clk}}$. The processing clock is derived from a 40 MHz oscillator. $\mathbf{f_{clk}}$ is not related to the external CLK_IN clock.

A 512-sample Dual-port RAM elastic buffer is used at the boundary between inputs and internal processing area. Thus, the input clocks frequencies can be independent from the internal processing clock frequency.

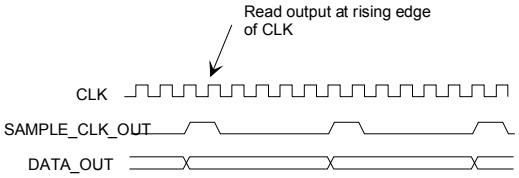
The input signals at the J2 connector are synchronous with the CLK_IN clock at J2/A1. This clock can be up to 40 MHz.

The output signals are synchronous with the rising edge of the 40 MHz reference clock CLK_OUT (i.e. all signals are stable at the rising edge of the reference clock CLK_OUT).

Input



Output



LEDs

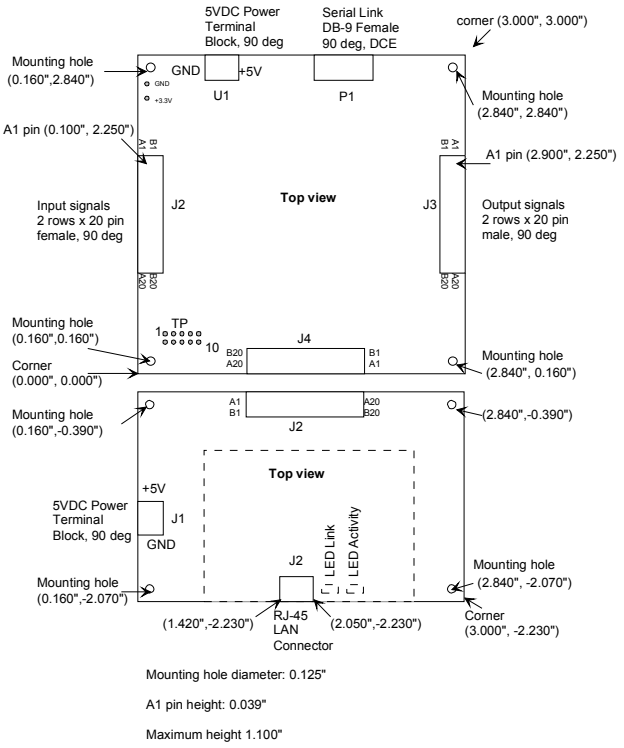
2 LEDs located close to the LAN RJ-45 jack provide summary information as to the LAN: Link and activity.

Test Points

Test points are provided for easy access by an oscilloscope probe. The main focus of these test points is to help monitor proper flow control operation.

Test Point	Definition
TP 1	TCP-IP connection on port 1024 (data stream) 1 = connected, 0 otherwise
TP 2	Future use
TP 3	TCP-IP connection on port 1028 (Monitoring & Control) 1 = connected, 0 otherwise
TP 4	Transmit data elastic buffer empty 1 = empty, 0 otherwise
TP 5	Transmit data elastic buffer full 1 = full, 0 otherwise
TP 6	Receive data elastic buffer empty 1 = empty, 0 otherwise
TP 7	Receive data elastic buffer more than half full 1 = more than half full, 0 otherwise
TP 8	Sample requests received through the J3 connector
TP9	Future use
TP10	Future use

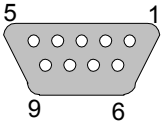
Mechanical Interface



Pinout

Serial Link P1

The DB-9 connector is wired as data circuit terminating equipment (DCE). Connection to a PC is over a straight-through cable. No null modem or gender changer is required.

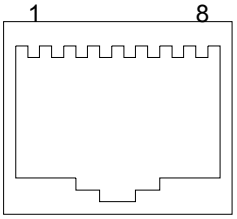


DB-9 Female

2 Transmit
3 Receive
5 Ground

LAN Connector J2

The RJ-45 Jack is wired as a standard PC network interface card. Connection to a LAN Hub is over a straight-through cable.



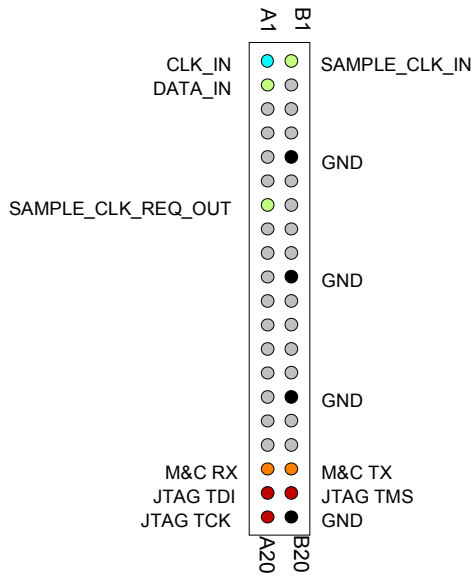
RJ-45 Jack

1 Tx+
2 Tx-
3 Rx+
6 Rx-

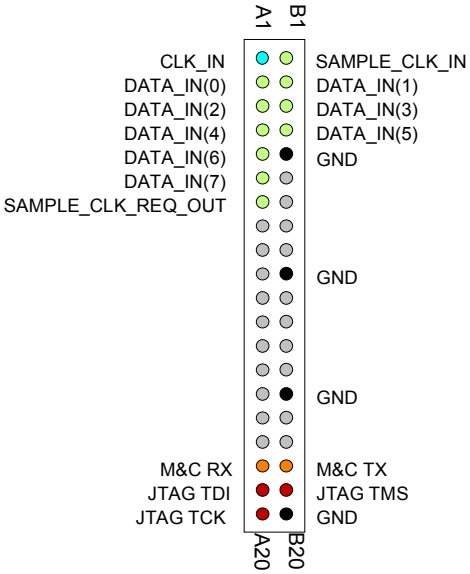
Input Connector J2

There are several possible connector configurations, depending on the application:

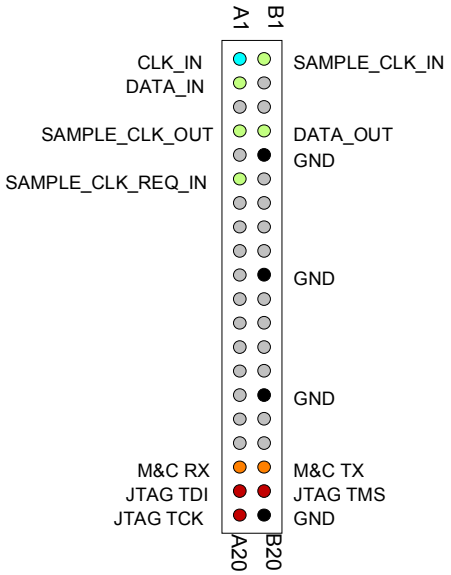
(a) 1-bit wide connection to another ComBlock [COM-1001, COM-1011,etc]

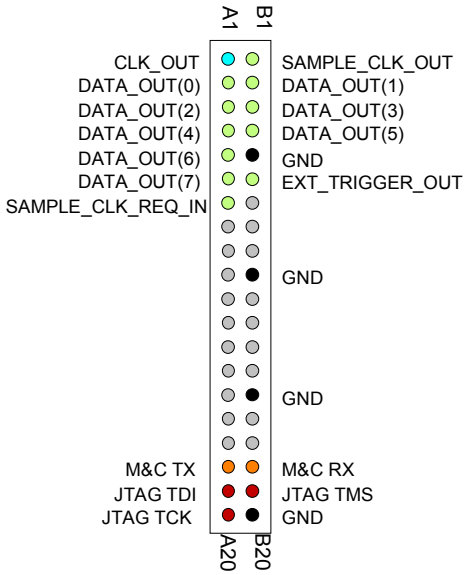


(b) 8-bit wide connection to another ComBlock [COM-8002,etc]



(c) Special case: input connector is used for bi-directional connection to COM-7001 module.

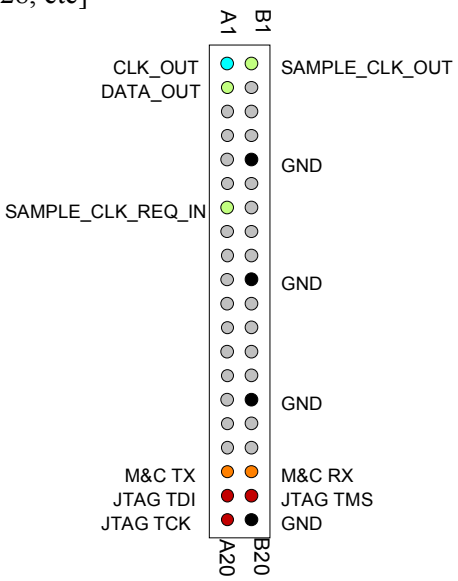




Output Connector J3

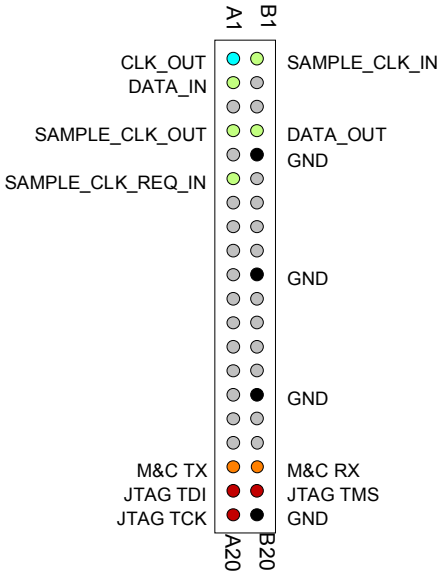
There are several possible connector configurations, depending on the application:

(a) 1-bit wide connection to another ComBlock [COM-1002, COM-1012, COM-1019, COM-1028, etc]



(b) 8-bit wide connection to another ComBlock [COM-8001, etc]

(c) Special case: output connector is used for bi-directional connection to COM-7001 module.



COM-2001
DIGITAL-TO-ANALOG CONVERSION
(I & Q COMPLEX BASEBAND)

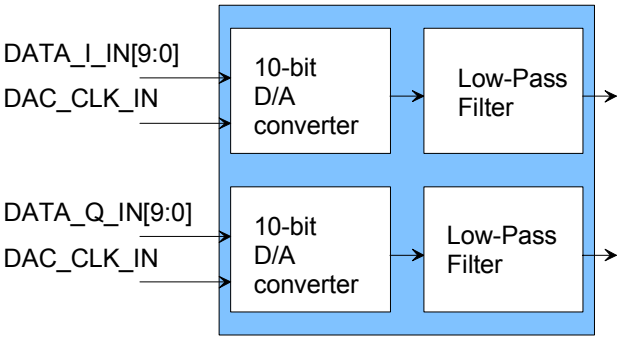
Key Features

- Converts the complex baseband digital signal to two analog baseband signals.
- Dual 125 Msamples/s 10-bit D/A converters.
- 6-pole Butterworth clock rejection filters Maximum bandwidth: +/- 13 MHz @±0.4dB ripple.
- A/D clock rejection @40 MHz > 84 dBc.
- Output voltage: 1Vpp with 0.85V DC bias.
- Single 5V supply
- Connectorized 3"x 3" module for ease of prototyping.
- Analog: SMA connectors
- Digital: standard 40 pin 2mm dual row connectors (left)

For the latest data sheet, please refer to the **ComBlock** web site: www.comblock.com/download/com2001.pdf. These specifications are subject to change without notice.

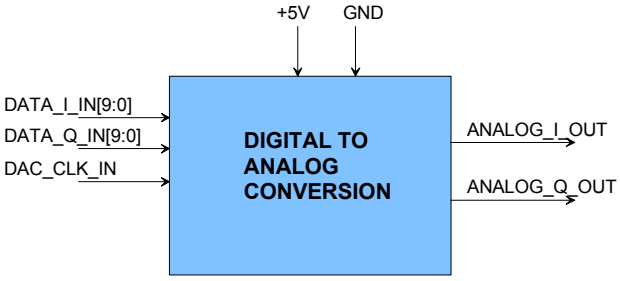
For an up-to-date list of **ComBlock** modules, please refer to www.comblock.com/product_list.htm .

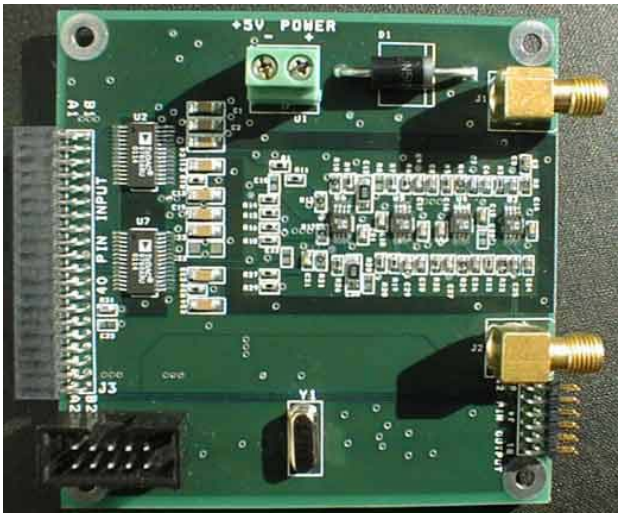
Block Diagram



Electrical Interface

Inputs / Outputs



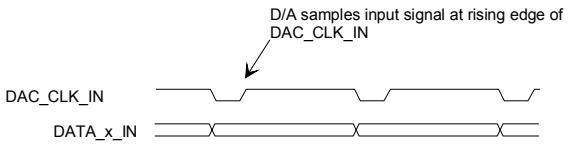


Input Module Interface	Definition
DATA_I_IN[9:0]	Modulated input signal, digital, baseband, real axis. 10-bit unsigned format. 0x000 for maximum output level 0x3FF for minimum output level 0x1FF or 0x200 for near center level. This data word is read at the rising edge of DAC_CLK_IN, and ignored at all other times.
DATA_Q_IN[9:0]	Modulated input signal, digital, baseband imaginary axis. Same format as DATA_I_IN.
DAC_CLK_IN	Input signal sampling clock. The input samples are stable at the rising edge of DAC_CLK_IN. Maximum sampling rate is 125 MHz.
CLK_IN	This signal is not used within the COM-2001. It is only forwarded to other ComBlocks in the assembly.
Analog Output Signals	**Definition**
ANALOG_I_OUT	Analog output, baseband, real-axis. Peak amplitude: 1.0Vpp DC bias: 0.85V. SMA female connector.
ANALOG_Q_OUT	Analog output, baseband, imaginary-axis. Peak amplitude: 1.0Vpp DC bias: 0.85V. SMA female connector.
Serial Monitoring & Control	DB9 connector. 115 Kbaud/s. 8-bit, no parity, one stop bit. No flow control.
Power Interface	4.75 – 5.25VDC. Terminal block. Power consumption is 120mA.

Timing

The input signals DATA_x_IN are read at the rising edge of the DAC_CLK_IN sampling clock on pin A13 of the J3 connector. The maximum sampling clock frequency is 125 MHz.

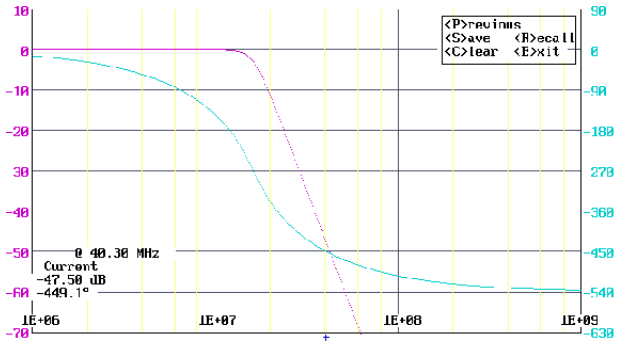
Input



Performance

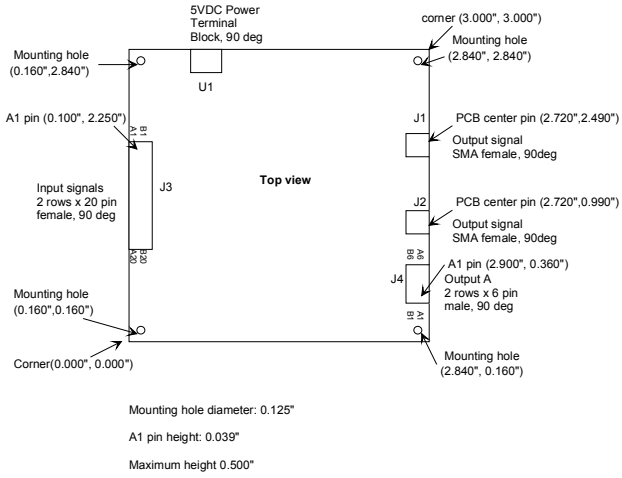
Low Pass Filter

Each D/A converter is followed by a 6-pole Butterworth low-pass filter to suppress harmonics. The filter response is as follows:



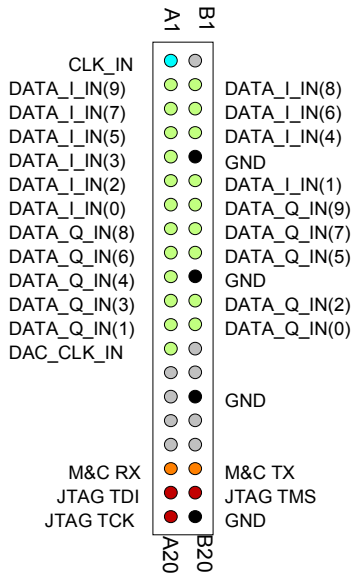
Out of band spectral spurious lines: < -84dBc in any 3 KHz band.

Mechanical Interface

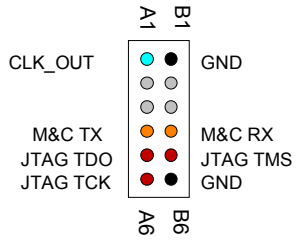


Pinout

Input Connector J3



Output Connector J4



This connector is to forward JTAG, GND and other monitoring and control signals to subsequent analog modules.

COM-1028 FSK/MSK/GFSK/GMSK DIGITAL MODULATOR

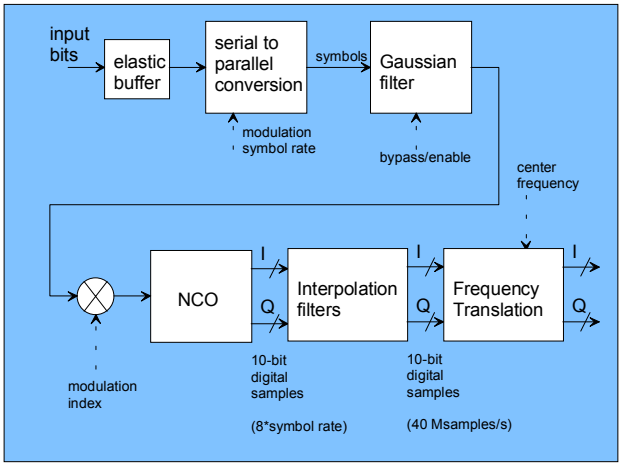
Key Features

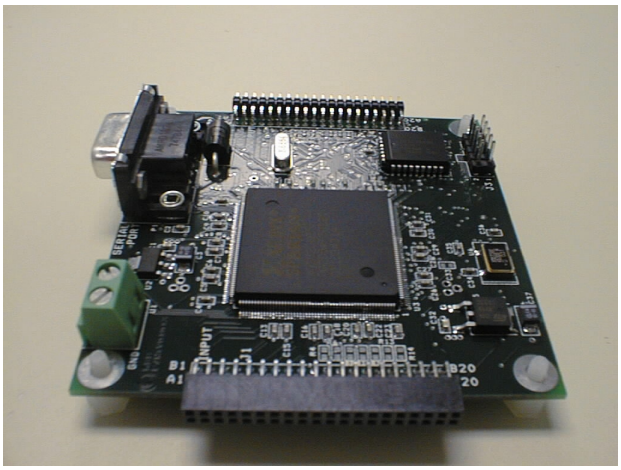
- Modulations:
  - Continuous phase FSK (CPFSK)
  - Minimum shift keying (MSK)
  - Gaussian frequency shift keying (GFSK)
  - Gaussian minimum shift keying (GMSK)
- Programmable 2-, 4-, 8-ary FSK
- Programmable modulation index h.
- Two selectable Gaussian filter BT product: 0.5 and 0.3.
- Programmable data rates up to 30/20/10 Mbps. (8-, 4-, 2-ary FSK)
- Internal generation of pseudo-random bit stream and unmodulated carrier for test purposes.
- On-board or external clock selection.
- **ComScope** –enabled: key internal signals can be captured in real-time and displayed on host computer.
- Connectorized 3”x 3” module for ease of prototyping. Standard 40 pin 2mm dual row connectors (left, right). Single 5V supply with reverse voltage and overvoltage protection. Interfaces with 3.3V LVTTL logic.

For the latest data sheet, please refer to the **ComBlock** web site: www.comblock.com/download/com1028.pdf. These specifications are subject to change without notice.

For an up-to-date list of **ComBlock** modules, please refer to www.comblock.com/product_list.htm .

Block Diagram





Typical Applications

- GSM:
  - GMSK, modulation index h=0.5, BT = 0.3, bit rate: 270.833 Kbps, channel spacing: 200 KHz
- Bluetooth:
  - GFSK, modulation index h = 0.32, BT = 0.5

- DECT:
    - GFSK, BT = 0.5, bit rate: 1.152 Mbps, channel spacing: 1.728 MHz

Electrical Interface

Input Module Interface	**Definition**
DATA_IN	Input data stream.
SAMPLE_CLK_IN	Input bit clock. One CLK-wide pulse. Read the input signals at the rising edge of CLK when SAMPLE_CLK_IN = '1'.
SYMBOL_CLK_IN	Input symbol clock. '1' identifies the first bit in a symbol. If not used, the symbol boundaries are automatically (internally) generated every 1 (2-FSK), 2 (4-FSK) or 3 (8-FSK) bits.
SAMPLE_CLK_IN_REQ	Output. One CLK-wide pulse. Requests a data bits from the module upstream. For flow-control purposes.
CLK_IN	Input reference clock for synchronous I/O. DATA_IN, SAMPLE_CLK_IN and SYMBOL_CLK_IN are aread at the rising edge of CLK_IN. Maximum 40 MHz.
Output Module Interface (Output data pushed out)	**Definition**
DATA_I_OUT[9:0]	Modulated output signal, real axis. 10-bit precision. Format: 2's complement or unsigned, selected by configuration bit 1.
DATA_Q_OUT[9:0]	Modulated output signal, imaginary axis. 10-bit precision. Same format as DATA_I_OUT.
SAMPLE_CLK_OUT	Output signal sampling clock. Read the output signal at the rising edge of CLK when SAMPLE_CLK_OUT = '1'. SAMPLE_CLK_OUT is fixed at '1' when the modulator is enabled. Fixed at '0' otherwise.
DAC_CLK_OUT	Output sampling clock for Digital to Analog Converters. DAC reads the output sample at the rising edge.
CLK_OUT	40 MHz output reference clock. Generated by dividing the internal processing clock: f_{clk} /2

Output Module Interface (Output data pulled)	**Definition**
SAMPLE_CLK_REQ_IN	Input. 100 MHz clock requesting output samples.
DATA_OUT[13:0]	Output. Quadrature baseband samples, 14-bit precision, 2's complement format. Bit 13 is the most significant bit. The in-phase (I) and quadrature (Q) samples alternate. Output samples are synchronous with the falling edge of SAMPLE_CLK_REQ_IN.
TX_ENABLE	Output. Transmit enable. Active high. The first sample after TX_ENABLE becomes active is an in-phase (I) sample.
Serial Monitoring & Control	DB9 connector. 115 Kbaud/s. 8-bit, no parity, one stop bit. No flow control.
Power Interface	4.75 – 5.25VDC. Terminal block. Power consumption is approximately proportional to the CLK frequency. The maximum power consumption is 650mA.

Important: I/O signals are 0-3.3V LVTTL. Inputs are NOT 5V tolerant!

Configuration (via Serial Link / LAN)

Complete assemblies can be monitored and controlled centrally over a single serial or LAN connection.

The module configuration parameters are stored in non-volatile memory. All control registers are read/write.

Parameters	**Configuration**
Symbol rate (f_{symbol_clk})	24-bit signed integer expressed as fsymbol rate * 2^{24} / $\mathbf{f_{clk}}$. $\mathbf{f_{clk}}$ is 80 MHz. The maximum symbol rate is 10 Msymbols/s (0x1FFFFF). The data rate is 1x, 2x or 3x the symbol rate depending on the M-ary number set in REG9.

	REG0 = bit 7-0 (LSB) REG1 = bit 15 – 8 REG2 = bit 23 – 16 (MSB)
Center frequency (f_c)	24-bit signed integer (2's complement representation) expressed as $f_c * 2^{24} / f_{clk}$. f_{clk} is 80 MHz. Valid range: +/- 20 MHz. (+/- 10 MHz for flat gain with COM-2001 D/A converter). REG3 = bits 7 – 0 REG4 = bits 15 – 8 REG5 = bits 23 - 16
Signal gain	Signal level. 8-bit unsigned integer. Applies equally to the I and Q channels. We recommend a maximum settings of 0x80 to avoid saturation in the subsequent digital-to-analog conversion module (COM-2001, COM-4004, etc). REG6 = bits 7-0
Modulation Index h	Modulation index h. Format 3.8 Thus, 0x0080 represents an index of 0.5. Ignored if MSK or GMSK modulation is selected (MSK implies h = 0.5). Valid range: 0 – 7.996 REG7: bits 7:0 LSB REG8: bit 10:8: MSB
M-ary number	Size of the symbol alphabet: 00 = 2-ary, 2-FSK, M=2 01 = 4-ary, 4-FSK, M=4 10 = 8-ary, 8-FSK, M=8 REG9 bits 1-0
Modulation	000 = FSK 001 = MSK 010 = GFSK 011 = GMSK REG9 bit 6-4
Output sample format	0 = 2's complement 1 = unsigned See also REG10 bit 2 for additional settings. REG10 bit 1
Output data flow	0 = output data is pushed to the next module (for example to COM-2001, or COM-1027) 1 = output data is pulled by next module (for example by the COM-4004) REG10 bit 2
Test mode	00 = disabled 01 = internal generation of 2047-bit periodic pseudo-random bit sequence as modulator input. (overrides external input bit stream). 10 = unmodulated carrier. (overrides external input bit stream) REG10 bit 5 – 4

Configuration example:

COM-1028 FSK modulator -> COM-2001 baseband D/A converters

In this setup, the COM-1028 generates a 2047-bit pseudo-random data stream at a rate of 1 Mbit/s. The modulation is 2-FSK with a modulation index of 0.5. The center frequency is 2 MHz. Mid-amplitude setting.

Settings:
COM-1028: 33 33 03 66 66 06 80 80 00 00 10
COM-2001: n/a

Monitoring (via Serial Link / LAN)

Monitoring registers are read-only.

Parameters	Monitoring
Version	Returns '1028x' when prompted for version number.

ComScope Monitoring

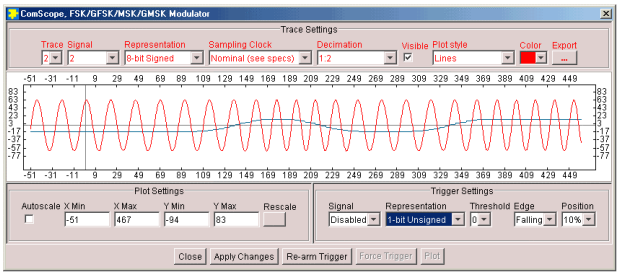
Key internal signals can be captured in real-time and displayed on a host computer using the ComScope feature of the ComBlock Control Center. The COM-1028 signal traces and trigger are defined as follows:

Trace 1 signals	Format	Nominal sampling rate	Buffer length (samples)
1: Data symbols before Gaussian filter	8-bit signed	8 samples /symbol	512
2: Data symbols after Gaussian filter	8-bit signed	8 samples /symbol	512
3: Frequency modulator phase	8-bit signed	8 samples /symbol	512
4: Modulated Signal I-channel	8-bit signed	$\mathbf{f_{clk}}$ (80 MHz)	512
Trace 2 signals	**Format**	**Nominal sampling rate**	**Capture length (samples)**
1: Internal PRBS-11 test sequence	Binary	1 sample / bit	4096
2: Modulated Signal Q-channel	8-bit signed	$\mathbf{f_{clk}}$ (80 MHz)	512
Trigger Signal	**Format**		
1: Start of internal PRBS11 test sequence	binary		

Signals sampling rates can be changed under software control by adjusting the decimation factor and/or selecting the $\mathbf{f_{clk}}$ processing clock as real-time sampling clock.

In particular, selecting the $\mathbf{f_{clk}}$ processing clock as real-time sampling clock allows one to have the same time-scale for all signals.

The ComScope user manual is available at www.comblock.com/download/comscope.pdf.



ComScope Window Sample: showing GMSK modulator output (red) and Gaussian filter output (blue).

Operation

FSK Modulation

The FSK modulation and its derivatives (CPFSK, MSK, GMSK, GFSK) are best described by the following equations for the modulated signal s(t). The first equation describes a phase modulator, with the modulated centered around the center frequency f_c.

$$s(t) = \sqrt{\frac{2E_s}{T}}.\cos(2\pi f_c t + \theta(t) + \theta_0)$$

where

- E_s is the energy per symbol
- T is the symbol period
- f_c is the center frequency
- $\theta(t)$ is the phase modulation

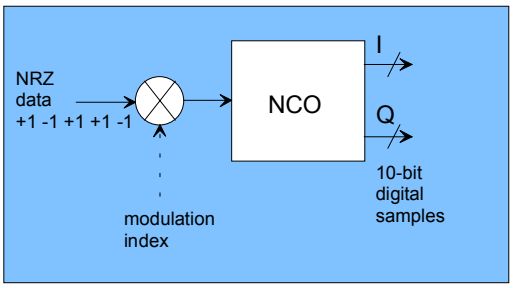
The COM-1028 implements a continuous phase FSK modulator. There are no phase discontinuities between symbols. The CPFSK phase modulation can be described as:

$$\theta(t) = \frac{\pi h}{T}\int_0^t a_i(t)dt$$

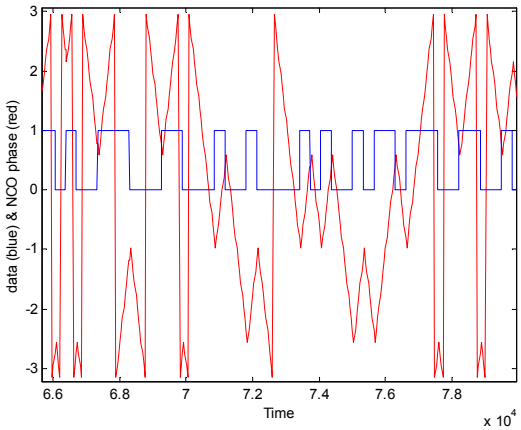
where:

- h is the modulation index. A modulation index of 0.5 yields a maximum phase change of π/2 over a symbol.
- a_i are the symbols. With 2-FSK, the binary data is represented as –1 (for '0') and +1 (for '1').

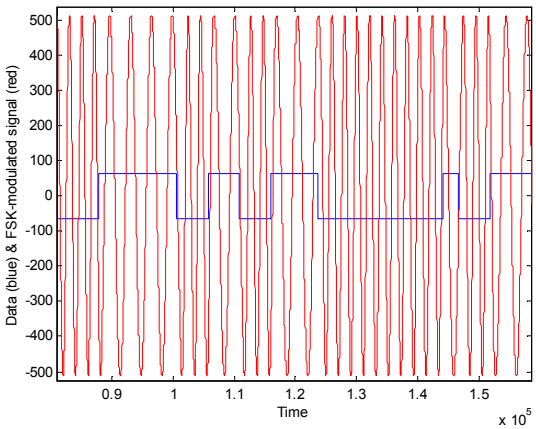
The generic implementation of a CPFSK modulator is based on the use of a numerically controlled oscillator (NCO) as shown in the block diagram below:



CPFSK modulator



NCO phase, continuous phase FSK 2-FSK, center frequency fc = 0, modulation index h = 0.5



Continuous FSK modulated signal example

FSK modulation is sometimes characterized by the frequency separation between symbols. The relationship between modulation index h and frequency separation is $f_{separation} = 0.5\ h\ f_{symbol\_clk}$

M-ary Number M

Transmitted data is grouped into symbols of size 1, 2, or 3 consecutive bits. The size of the symbol alphabet is thus M = 2, 4 or 8. The packing of serial data bits into alphabet symbols is such that the MSB is received first at the DATA_IN serial input.

The mapping between symbol alphabet and modulation symbol a_i is described in the table below:

Symbol alphabet	Modulation symbol a_i
2-FSK '0'	-1
2-FSK '1'	+1
4-FSK "00"	-3
4-FSK "01"	-1
4-FSK "10"	+1
4-FSK "11"	+3
8-FSK "000"	-7
8-FSK "001"	-5
8-FSK "010"	-3
8-FSK "011"	-1
8-FSK "100"	+1
8-FSK "101"	+3
8-FSK "110"	+5
8-FSK "111"	+7

Gaussian Filter

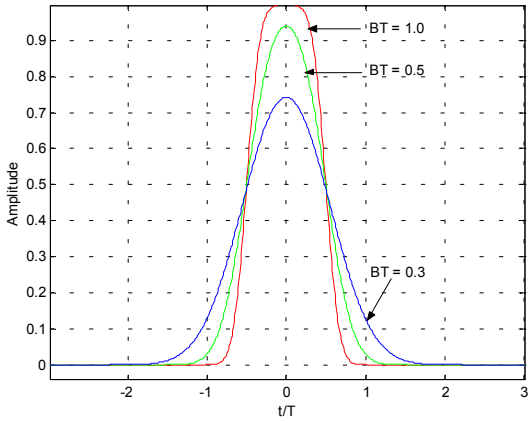
A filter with Gaussian impulse response can be used as pre-filtering of the symbols prior to the continuous phase modulation. Its purpose is to control the modulated signal bandwidth.

The Gaussian filter is characterized by its BT product (B is the –3 dB bandwidth, T is the symbol period = $1/f_{\text{symbol rate}}$). The lower the BT product, the narrower the modulation bandwidth and the higher the inter-symbol interference.

The filter impulse response is expressed analytically as: $h(t) = \frac{1}{\sqrt{2\pi}\sigma T}\exp\left(\frac{-t^2}{2\sigma^2 T^2}\right)$

where $\sigma = \frac{\sqrt{\ln(2)}}{2\pi BT}$

The impulse response h(t) is further convoluted with the rectangular waveform representing the symbol width T. The resulting impulse is illustrated below for BT = 0.3, 0.5 and 1.0.



Shaping pulses for BT = 0.3, 0.5 and 1.0 (Gaussian convoluted with rectangle window)

Configuration Files

In order to provide for configuration flexibility without unduly increasing the hardware complexity, some features require uploading different firmware into the ComBlock using the ComBlock control center.

- Channel filter (Gaussian filter) BT product: 0.3 and 0.5.

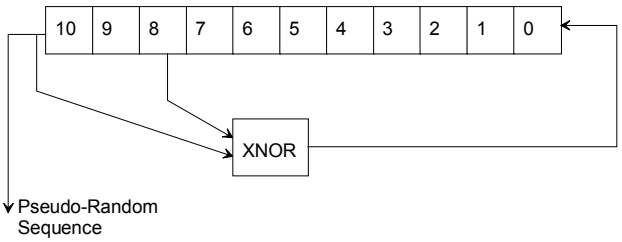
All firmware versions can be downloaded from www.comblock.com/download.

COM-1028A FSK/MSK/GFSK/GMSK modulator, Gaussian filter BT = 0.3 .

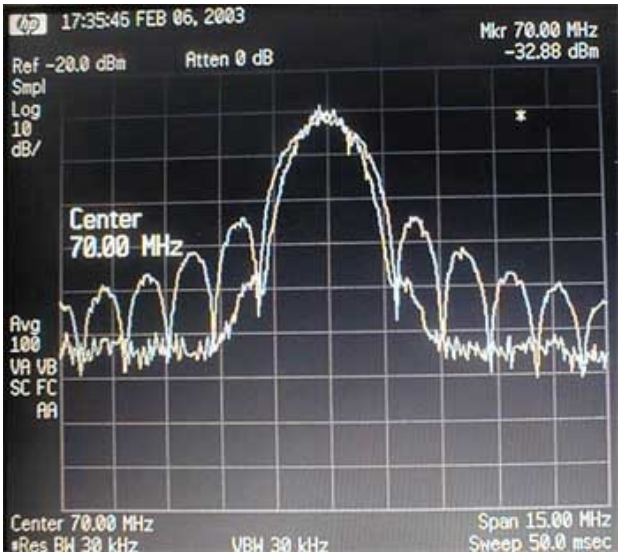
COM-1028B FSK/MSK/GFSK/GMSK modulator, Gaussian filter BT = 0.5.

Pseudo-Random Bit Stream (PRBS-11)

A periodic pseudo-random sequence can be used as modulator source instead of the input data stream. A typical use would be for end-to-end bit-error-rate measurement of a communication link. The sequence is 2047-bit long maximum length sequence generated by a 11-tap linear feedback shift register:



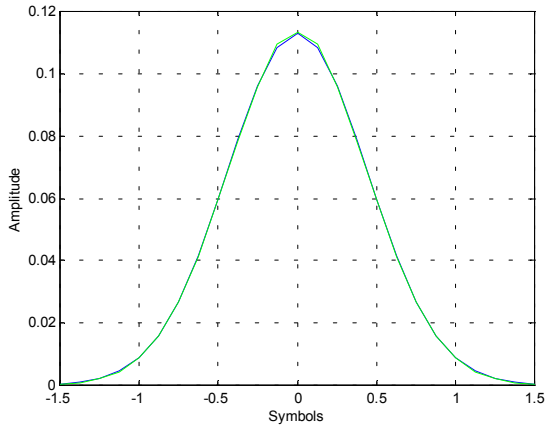
The first 100 bits of the PN sequence are as follows:
0000000000 0111111111 0011111110 0001111100
1100111000 0000010011 1111010001 1110110100
1101001100 0011000001



70 MHz MSK versus GMSK spectrum Using COM-1028 & COM-4004

Implementation

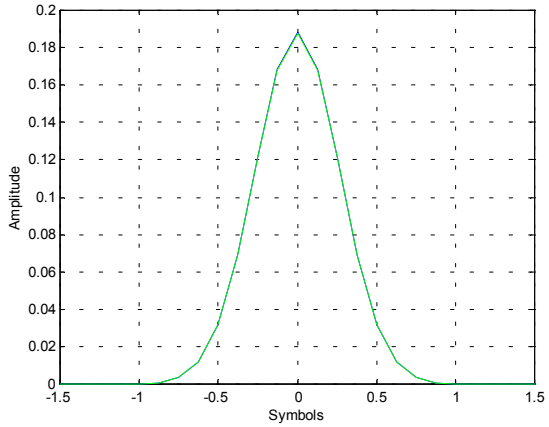
Gaussian Filter Response BT = 0.3



Filter impulse response. 8 samples/symbol ideal (blue), implemented (green).

The gaussian filter with BT=0.3 is a 25-tap FIR filter with the following impulse response:
Coeff(0) = 3/1024
Coeff(1) = 7/1024
Coeff(2) = 17/1024
Coeff(3) = 36/1024
Coeff(4) = 72/1024
Coeff(5) = 130/1024
Coeff(6) = 220/1024
Coeff(7) = 336/1024
Coeff(8) = 488/1024
Coeff(9) = 640/1024
Coeff(10) = 784/1024
Coeff(11) = 896/1024
Coeff(12) = 928/1024
Coeff(j=13:24) = coeff(24-j);

Gaussian Filter Response BT = 0.5



Filter impulse response. 8 samples/symbol ideal (blue), implemented (green).

The gaussian filter with BT=0.5 is a 17-tap FIR filter with the following impulse response:
Coeff(0) = 1/1024
Coeff(1) = 6/1024
Coeff(2) = 28/1024
Coeff(3) = 95/1024
Coeff(4) = 258/1024
Coeff(5) = 568/1024
Coeff(6) = 992/1024
Coeff(7) = 1376/1024
Coeff(8) = 1536/1024
Coeff(j=9:16) = coeff(16-j);

Timing

Clocks

An 80 MHz internal clock f_{clk} is generated by frequency doubling of the 40 MHz oscillator installed on the COM-1028 board. f_{clk} is not related to the CLK_IN clock. f_{clk} is used for internal processing and for generating the output clock CLK_OUT = $f_{clk}/2$.

Input data DATA_IN is first written into an input elastic buffer at the rising edge of CLK_IN when SAMPLE_CLK_IN = '1'.

The data is read out of the input elastic buffer at the symbol rate * 1 (2-ary FSK), * 2 (4-ary FSK) or * 3 (8-ary FSK).

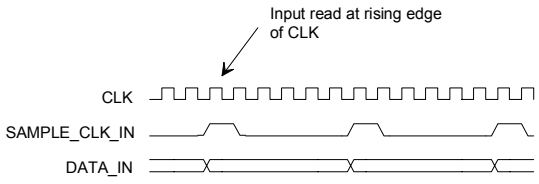
The input buffer size is 256 symbols.

I/Os

In general, the I/O signals are synchronous with the rising edge of the reference clock CLK_IN or CLK_OUT (i.e. all signals transitions always occur after the rising edge of clock). The maximum frequency for CLK_IN is 40 MHz. The frequency for CLK_OUT is fixed at 40 MHz ($f_{clk}/2$) .

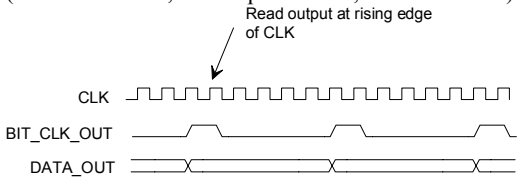
A special case is when the output is connected to the COM-4004 70 MHz IF modulator. The data samples are then pulled out by a 100 MHz clock. The complex I and Q samples are time-multiplexed for a maximum throughput of 50 Msamples/s. For timing details, please refer to the COM-4004 specifications.

Input



Output

(REG10 bit2 = 0, data is pushed out, CLK=40 MHz)

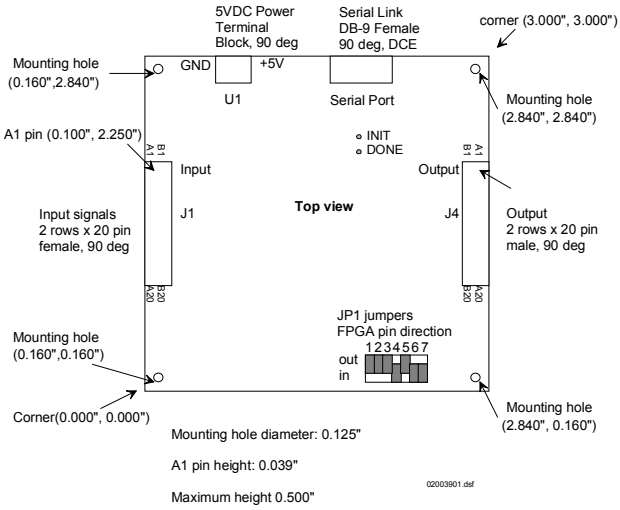


Test Points

Test points are provided for easy access by an oscilloscope probe.

Test Point	Definition
J1connector pin B7	Symbol clock
J1 connector pin B8	Bit clock
J1 connector pin B9	Sample clock
J1 connector pin A9	PRBS11 start of 2047-bit period
TP1	FPGA DONE pin. High indicates proper download of the FPGA configuration file.

Mechanical Interface

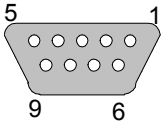


Note: All seven JP1 jumpers must be in the 'OUT' location.

Pinout

Serial Link P1

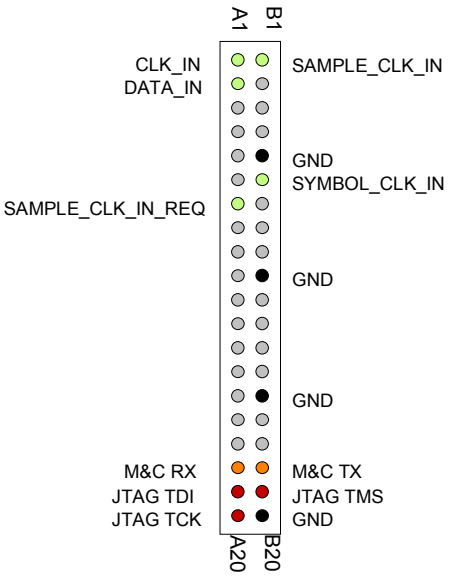
The DB-9 connector is wired as data circuit terminating equipment (DCE). Connection to a PC is over a straight-through cable. No null modem or gender changer is required.



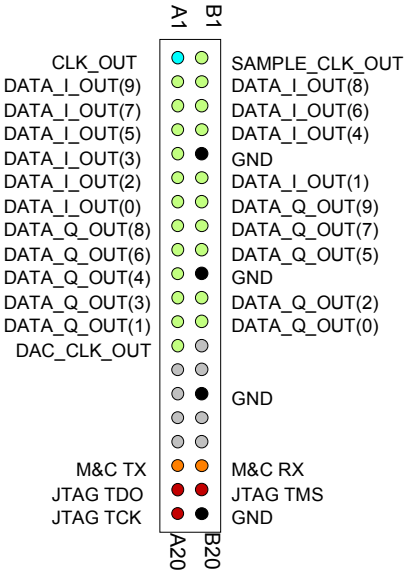
DB-9 Female

2 Transmit
3 Receive
5 Ground

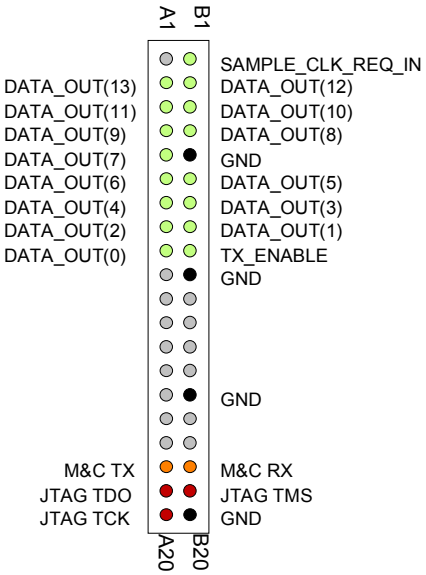
Input Connector J1



Output Connector J4



This connector is used when output data is pushed out (configuration REG10 bit2 = 0).



This connector is used when output data is pulled out by the next module (configuration REG10 bit 2 = 1).

COM-4005 CELLULAR BAND [800 -1000 MHz] QUADRATURE RF MODULATOR

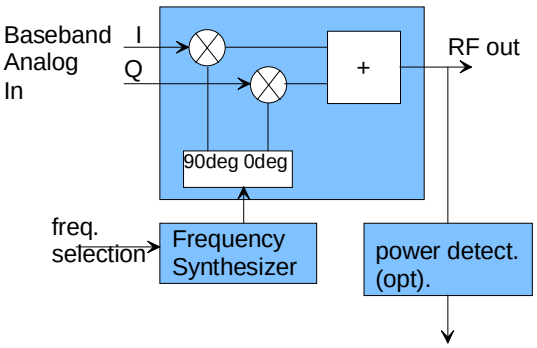
Key Features

- Quadrature modulator 800 – 1000 MHz center frequency.
- Low-noise frequency synthesizer can be tuned over entire range by steps of 100 KHz.
- Optional output power measurement has 0.1 dB resolution.
- Output power can be controlled over 20 dB range using 10-bit control words. Non-linear scale.
- Selectable internal / external 10 MHz frequency reference for the frequency synthesizer.
- Single 5V supply
- Connectorized 3"x 3" module for ease of prototyping. SMA connectors.

For the latest data sheet, please refer to the **ComBlock** web site: www.comblock.com/download/com4005.pdf. These specifications are subject to change without notice.

For an up-to-date list of **ComBlock** modules, please refer to www.comblock.com/product_list.htm .

Block Diagram



Electrical Interface

Inputs / Outputs

Input Module Interface	**Definition**
ANALOG_I_IN	Modulated input signal, analog, baseband, real axis. 1Vpp max. 0.85V DC bias. SMA connector.
ANALOG_Q_IN	Modulated input signal, analog, baseband, imaginary axis. 1Vpp max. 0.85V DC bias. SMA connector
EXT_REF_CLK	External 10 MHz frequency reference for frequency synthesis. Sinewave, clipped sinewave or squarewave. Minimum level 0.5Vpp. Maximum level: 3.3Vpp. Use square wave for best phase noise performances.
Analog Output Signals	**Definition**
RF_OUT	Modulated RF output. 800 – 1000MHz Maximum output level: -3 dBm. Impedance: 50 Ohms. SMA connector

Serial Monitoring & Control	DB9 connector. 115 Kbaud/s. 8-bit, no parity, one stop bit. No flow control.
Power Interface	4.75 – 5.25VDC. Terminal block. Power consumption is 250mA max.

Configuration (via Serial Link / LAN)

Complete assemblies can be monitored and controlled centrally over a single serial or LAN connection.

The module configuration parameters are stored in non-volatile memory. The installation default values are highlighted in bold.

Parameters	Configuration
RF frequency	Range 800 MHz to 1000 MHz, steps 100 KHz, expressed in Hz. Default: **950 MHz**. REG0: bit 7:0 (LSB) REG1: bit 15:8 REG2: bit 23:16 REG3: bit 31:24 (MSB)
Gain control	10-bit control. Non-linear scale. Zero is lowest power. AGC range : 22 dB @ 800 MHz (typ.) 26 dB @ 1000 MHz (typ.) Default: **00 00000000** REG4: bit 7-0 (LSB) REG5: bit 1-0 (MSB)
External/Internal frequency reference	0 = internal 1 = external. Default:**0** REG6: bit 0
Modulator on/off	0 = modulator off 1 = modulator on Default:**0** REG6: bit 2

Default configuration at manufacturing:
REG0 = 0x80
REG1 = 0xD9
REG2 = 0x9F
REG3 = 0x38
REG4 = 0x00
REG5 = 0x00
REG6 = 0x00
950 MHz, minimum gain, internal frequency reference, modulator off.

Monitoring (via Serial Link / LAN)

Parameters	Monitoring
Version	Returns '4005A or B' when prompted for version number.
Power measurement (option)	10-bit number. The higher the number, the lower the power. The power measurement linearity is shown below. REG7 bits 7-0: bit 7-0 (LSB) REG8 bits 1-0: bits 9-8 (MSB)

Operations

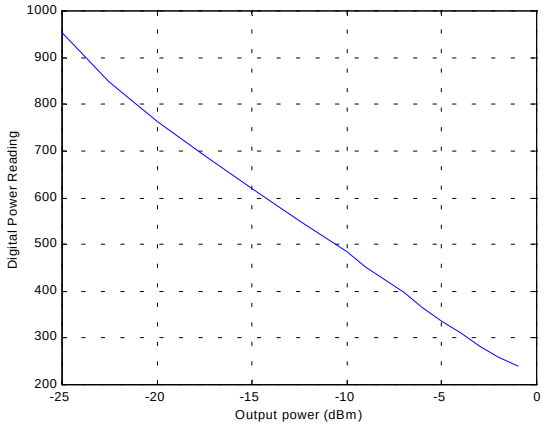
Internal vs External Frequency Reference

In order to use the external frequency reference, connect a 10 MHz sinewave, clipped sinewave or square wave to the SMA connector J2. Then select external frequency reference by software command from the ComBlock control center.

In order to use the internal frequency reference, either physically disconnect the external 10 MHz signal at SMA connector J2, or place the external input signal in high impedance mode. Then select internal frequency reference by software command from the ComBlock control center.

Power Measurement (Option -B)

Output power measurement is provided as an option (-B). Output power measured with +/- 0.2 dB accuracy over a range from –25 dBm to the maximum output power. The 10-bit measurement linarity is shown below [800 MHz output signal]:



Test Points

Test points are provided for easy access by an oscilloscope probe.

Test Point	Definition
TP1	Internal / External reference clock
TP2	Frequency synthesizer PLL lock status

Performance

Quadrature phase error: 1. deg rms. typ
I/Q amplitude balance error: 0.2 dB.typ
ON/OFF rejection: > 80 dB

LO leakage (at output, maximum AGC gain, +20 KHz input signal):

-36 dBm @ 800 MHz, typ.
-37 dBm @ 1.00 GHz, typ.

Sideband suppression (at output, maximum AGC gain, + 20KHz input signal):

-42 dBc @ 800 MHz, typ.
-51 dBc @ 1.0 GHz, typ.

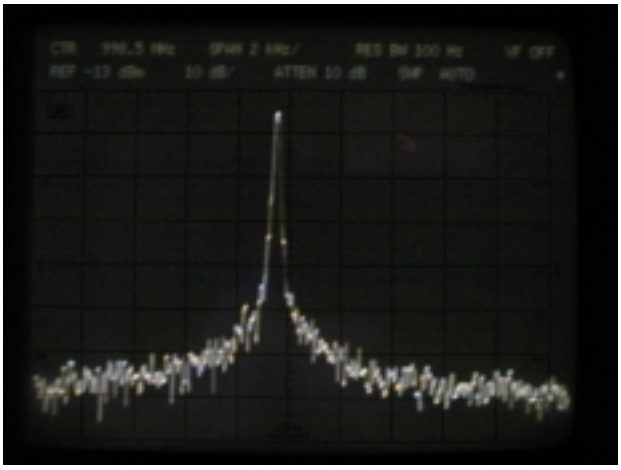
Out-of-band spurious spectral lines: < -60 dBc (Exception: a –47dBc spectral line may be present at 120 MHz from the center frequency).

Power detection (option -B) resolution: 0.1 dB.

Phase noise:

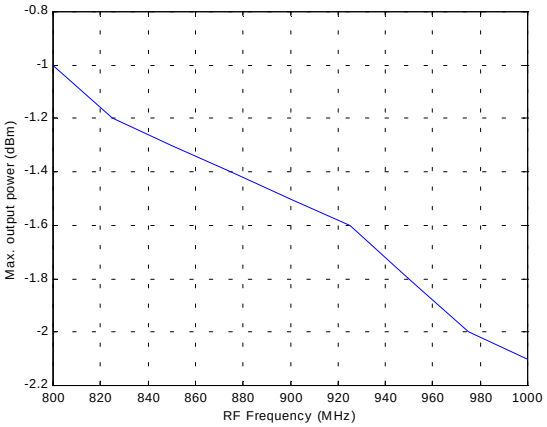
<-50 dBc @ 100 Hz
< -65 dBc @ 1 KHz
< -82 dBc @ 10 KHz
< -110 dBc @ 100 KHz

The phase noise measurements are similar when internal or external frequency references are used.



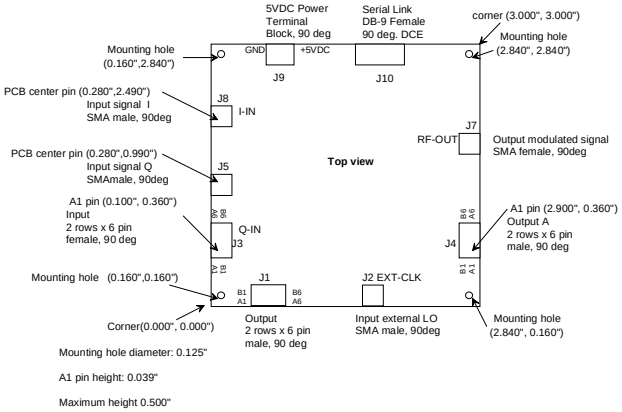
Phase noise measurement: internal reference clock 2 KHz/div, 10 dB/div. 998.5 MHz center freq., 100 Hz resolution bandwidth.

Maximum output power level (for a 1Vpp input):



Minimum output power: -25 dBm (800-1000 MHz).

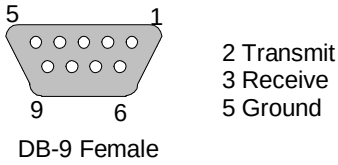
Mechanical Interface



Pinout

Serial Link J10

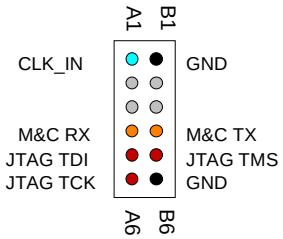
The DB-9 connector is wired as data circuit terminating equipment (DCE). Connection to a PC is over a straight-through cable. No null modem or gender changer is required.



DB-9 Female

Input Connector J3

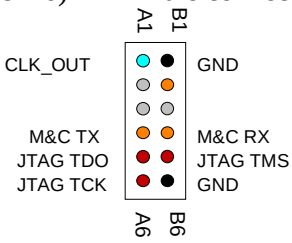
12-pin (2 rows x 6) 2mm female connector.



This module is is designed for direct connection to the COM-2001 baseband digital-to-analog conversion module.

Output Connectors J1,J4

12-pin (2 rows x 6) 2mm male connector.



This connector is to forward JTAG, GND and other monitoring and control signals to subsequent analog modules.