UNIVERSITY OF CRETE
DEPARTMENT OF COMPUTER SCIENCE

DVB-RCT Physical Layer and Implementation of the Frame Adaptation Unit

MSc. Thesis of

Evangelatou Anastasia Vassiliki

Supervisor: Panagiotis Tsakalides

Heraklion March 2005

UNIVERSITY OF CRETE
DEPARTMENT OF COMPUTER SCIENCE

DVB-RCT Physical Layer and Implementation of the Frame Adaptation Unit

Submitted to the
Department of Computer Science
in partial fulfillment of the requirements for the degree of
Master of Science

Author: ________________________________
Anastasia-Vassiliki Evangelatou
Department of Computer Science

Committee:
Supervisor ________________________________
Panagiotis Tsakalides, Associate Professor

Member ________________________________
Manolis Katevenis, Professor

Member ________________________________
Apostolos Traganitis, Professor

Accepted by:
Chairman of the
Graduate Studies Committee ________________________________
Dimitris Plexousakis, Associate Professor

Heraklion, March 2005

'Cheshire Puss,' she began, rather timidly, as she did not at all know whether it would like the name: however, it only grinned a little wider. 'Come, it's pleased so far,' thought Alice, and she went on. 'Would you tell me, please, which way I ought to go from here?'

'That depends a good deal on where you want to get to,' said the Cat.

'I don't much care where--' said Alice.

'Then it doesn't matter which way you go,' said the Cat.

'As long as I get somewhere,' Alice added as an explanation.

'Oh, you're sure to do that,' said the Cat, 'if you only walk long enough.'

Alice felt that this could not be denied, so she tried another question. 'What sort of people live about here?'

'In that direction,' the Cat said, waving its right paw round, 'lives a Hatter: and in that direction,' waving the other paw, 'lives a March Hare. Visit either you like: they're both mad.'

'But I don't want to go among mad people,' Alice remarked.

'Oh, you can't help that,' said the Cat: 'we're all mad here. I'm mad. You're mad.'

'How do you know I'm mad?' said Alice.

'You must be,' said the Cat, 'or you wouldn't have come here.'

(Extract from "Alice in Wonderland", Lewis Carroll, 1971)

Abstract

The addition of interactive services to Digital Television is the key element to the transition from Analog to Digital Broadcasting. Such services reflect the current wishes and expectations of television viewers from Digital TV. The way to realize this goal is the definition of a return channel using the same medium as the existing broadcasted. In defining a return channel for the Digital Video Broadcasting Terrestrial system, the DVB forum produced DVB-RCT (Digital Video Broadcasting Return Channel Terrestrial), an ETSI standard (EN 301 958) providing specification for interactivity in the VHF/UHF bands. The result is the definition of a high bandwidth Wireless Interactive Terrestrial Digital TV system using a multiple access Orthogonal Frequency Division Multiplexing arrangement.

The next step that must be taken in the direction of DVB-T becoming a bidirectional system is the design of a VLSI solution for the integration of DVB-RCT in a TV set top box. The goal of this thesis is the implementation, as part of a future ASIC, of the stage that constitutes the cornerstone of the DVB-RCT physical layer: the Frame Adaptation Unit. First, we present the physical layer and the system parameters. A careful study of the transmission parameters and the physical layer is necessary in order to comprehend the Frame Adaptation requirements. Then, we define the unit's functionality and main architecture. The main accomplishment of the thesis is the design specification of the Frame Adaptation Unit and its implementation with the creation of a validated synthesizable VHDL model. The delivered model has been integrated in an FPGA platform employed for a successful real time demonstration of the DVB-RCT system.

Περίληψη

Η προσθήκη αλληλεπιδραστικών υπηρεσιών στη Ψηφιακή Τηλεόραση είναι μία από τις κρίσιμες κινήσεις για την μετάβαση από την αναλογική ευρεία μετάδοση σήματος στη ψηφιακή. Τέτοιου είδους υπηρεσίες αντανακλούν τις επιθυμίες και προσδοκίες των τηλεθεατών από την Ψηφιακή Τηλεόραση. Η πραγματοποίηση αυτού του στόχου μπορεί να επιτευχθεί με την χρήση ενός καναλιού επιστροφής που να χρησιμοποιεί το ίδιο μέσο με ένα υπάρχον σύστημα μετάδοσης. Στο πλαίσιο ορισμού ενός τέτοιου καναλιού για το σύστημα Ψηφιακής Επίγειας Μετάδοσης (DVB-T), το φόρουμ του DVB δημιούργησε το DVB-RCT (Digital Video Broadcasting Return Channel Terrestrial), ένα πρότυπο προδιαγραφών του ETSI (EN 301 958), για αμφίδρομη επικοινωνία στις VHF/UHF συχνότητες. Το αποτέλεσμα είναι ο ορισμός ενός ασύρματου επίγειου τηλεοπτικού συστήματος, διπλής κατεύθυνσης, το οποίο χρησιμοποιεί μία διάταξη πολλαπλής πρόσβασης βασισμένη σε OFDM διαμόρφωση.

Το επόμενο βήμα προς την πραγματοποίηση ενός αμφίδρομου DVB-T συστήματος είναι ο σχεδιασμός ενός ολοκληρωμένου κυκλώματος για την ενσωμάτωση του DVB-RCT στη τηλεοπτική συσκευή ή μία περιφερειακή μονάδα αυτής. Ο στόχος αυτής της μεταπτυχιακής εργασίας είναι η υλοποίηση, ως μέρος ενός μελλοντικού ASIC, του σταδίου που αποτελεί τον ακρογωνιαίο λίθο του Φυσικού επιπέδου του DVB-RCT: της μονάδας Προσαρμογής Πλαισίου (Frame Adaptation). Αρχικά παρουσιάζουμε το Φυσικό επίπεδο και τις παραμέτρους του συστήματος. Η προσεκτική μελέτη των παραμέτρων μετάδοσης και του Φυσικού επιπέδου είναι απαραίτητη για την εις βάθος κατανόηση των απαιτήσεων της Προσαρμογής Πλαισίου. Μετά από αυτή την ανάλυση, ορίζουμε τις λειτουργίες της μονάδας και την κύρια αρχιτεκτονική της. Το κύριο επίτευγμα αυτής της μεταπτυχιακής εργασίας είναι οι σχεδιαστικές προδιαγραφές της μονάδας Προσαρμογής Πλαισίου και η υλοποίησή της με την δημιουργία ενός έγκυρου μοντέλου στη γλώσσα περιγραφής υλικού VHDL. Το παραδοτέο μοντέλο ενσωματώθηκε σε μία FPGA πλατφόρμα, η οποία χρησιμοποιήθηκε για την επιτυχή επίδειξη σε πραγματικό χρόνο του DVB-RCT συστήματος.

Acknowledgments

First of all I would like to thank Assistant Professor P. Tsakalidis for accepting the supervision of this thesis and helping me patiently comprehend in depth the telecommunication background required in my work, as well as to resolve any issues that troubled me. In addition, I thank Professor M.Katevenis, whose experience and methodic way of thinking led me to organize correctly my work and realize that a hardware engineer must always search for the “big picture” behind every design choice. I also want to thank Professor Apostolos Traganitis, who kindly accepted to be the third member of my committee.

I wish to thank Mr C.Papadas who gave me the opportunity and encouraged me to continue my studies. The list of people with who I have collaborated during this two and a half years is long: C.Karakonstantis, D.Daniolos (rip), D.Fragopoulos, N.Venios, A.Vlahos. I especially thank D.Kapsalis and V.Papaefstathiou who were my closest partners in this project, but also friends and advisors when needed. Finally, I thank S.Lyberis for helping me in the review of the present document.

Many friends have stood by my side during this period. There aren’t enough words to express my gratitude towards Despina Evangelatou who has been to me dear family, patient house-mate, close friend, confessor, psychoanalyst, cook :o) and so many, many other things. I honestly don’t know what I would have done without you, cousin! Nassia Papagianni, my “sister”, wasn’t physically present, but was only a phone call away, ready to discuss trivial news or existential issues with the same ease. She has been the person that made me realize that even if I want to travel and experience life in distant places, my final destination is near the people I care the most. I also want to thank Olga Dokianaki, for “adopting” me, being my Cretan family and sharing the good and bad moments of our graduate studies. I had the chance to make many new friends in the University. We have shared a lot of laughs with Anastasia Katranidou, Lefteris Sidirourgos and Roza Akkus. We listened to a lot of music with Antonis Zachopoulos and enjoyed many colourful discussions with Panagiotis Koutsourakis. I also thank Vangelis Aggelakis and Stephanos Papadakis for their valuable help and support, as well as my office neighbours Stratos Politis, Yannis Haritakis and Dimitris Kalopsinakis who were always available when I needed a friendly advise and someone to talk to.

The last two persons I would like to thank are also the most important ones: my mother Rena Karahaliou and my father Grigoris Evangelatos. They have always supported my choices and encouraged me to pursue my goals. They were my faithful advisors when I needed help, as well as strict and impartial judges when I deserved it. Although, you know it without me writing it down, I still want to say it. Thank you!

Contents

List of Figures

List of Tables

1 Introduction

Nowadays, the TV industry has concentrated its interest into the transition from Analog to Digital Video Broadcasting. The most efficient standard for terrestrial Digital Video Broadcasting is DVB-T [1] and is already being used not only in Europe but also in several countries throughout the world.

As digital television evolves, new ways are investigated, in order to make it more appealing to its users, the television viewers. A television user is not generally interested in switching from Analogue to Digital TV just for the pure digital broadcasting TV service. Consequently, the need for new innovative services appears, which will convince the users for the superiority of Digital TV services. The model, which meets the requested services' profile, is the one of Interactive Services. These services can be bandwidth demanding as Internet, or can be totally new, presenting a strong real-time relationship with the TV programs as T-Commerce (Television Commerce), interactive advertising, tele-voting, tele-quiz and others. Following the public demands, a new standard was developed which describes an interactive version of DVB-T and more specifically, the return channel transmission.

Real-time interactive Digital Terrestrial Television Services need a wireless interaction channel, and the European Telecommunications Standard Committee composed the DVB-RCT standard [2] as a response to that need. The implementation of DVB-RCT with an application specific integrated circuit as part of a set-top box is recommended because the silicon solution is cost effective. In addition, the user can optionally add the interactive feature to his existing Digital TV terminal, by just connecting the set-top box. That way all digital televisions can be easily upgraded.

Some of the DVB-RCT advantages are its low cost and a flexible Wireless Multiple Access based on the well-known OFDM technique that is well suited for transmission in the terrestrial channel. It is important that the DVB-RCT standard is designed to exploit the already existing resources with only small relatively adaptations. The interactive system consists of a downstream channel (forward interaction) to the user established over a DVB-T compliant terrestrial broadcast network, and a return interaction channel based on a Wireless VHF/UHF transmission (upstream) of the same type. As far as it concerns the Base Station side, all that is needed is to demodulate the upstream signal transmitted by the terminal.

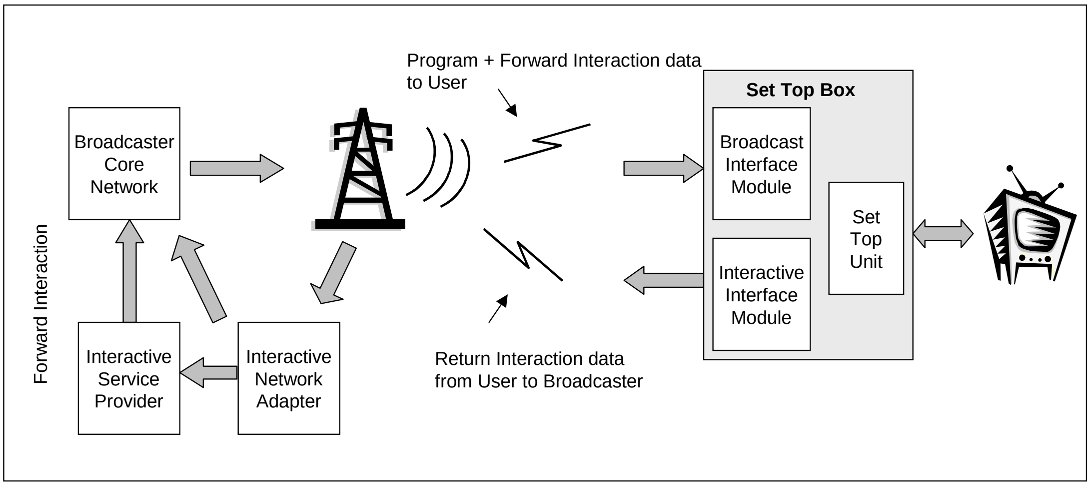

Figure 1 DVB-RCT network concept

The concept of the DVB-RCT network is illustrated in Figure 1. The downstream transmission from the Base Station consists of the regular television program, plus forward interaction information and it also provides synchronization to all the users' television terminals. The same antenna used for reception by the terminal can be used for upstream transmission.

An implementation prototype has been scheduled in the framework of making DVB-RCT a reality. This thesis work consists part of this effort towards this direction by specifying the design of Frame Adaptation, one of the critical Physical Layer's units. The system described by the standard has the ambition of being fully configurable by offering a variety of different physical parameters. That permits the implemented DVB-RCT system to respond to different needs and requirements. As already mentioned the Return Interaction Channel is based on multiple access OFDM technique. Six different transmission modes are supported as a combination of three available carrier spacing values (~1KHz, ~2KHz, ~4KHz) and two alternative OFDM carrier sets (1024 or 2048 carriers). The carrier shaping can be either Nyquist or Rectangular and in the second case the guard interval size can be the 1/4, 1/8, 1/16, 1/32 of the OFDM symbol. There are two Transmission Frame types and three burst structure types whose combinations define three Medium Access Schemes, providing different trade-offs of frequency diversity and burst structure duration. The applied coding can be Turbo or Concatenated (Reed Solomon and Convolutional). Then according to the encoding rate (½ or ¾) and the modulation (available constellation types: QPSK, 16QAM, 64QAM) we can have variable useful data payloads per burst. All the above parameters ensure a flexible net bit rate per carrier that ranges from 0.6 Kbps to 15 Kbps (depending on the mode) and the service range can reach a cell radius of 65km.

The present document's goal is twofold. Initially we present the DVB-RCT physical layer from the terminal side. Then we will concentrate our interest to the design of a critical unit in the physical layer chain: the Frame Adaptation. More specifically in the next chapters there will be an introduction to Orthogonal Frequency Division Multiplexing followed by a presentation of the basic system principles and definitions. Then the physical layer's diagram will be given along with a brief description of the various stages of the chain. The chain stages are: Randomization, Encoding, Interleaving, Frame Adaptation, Ranging and Pilot Code Generation, Symbol Mapping, Inverse FFT, Shaping and Up-Converting. The main attention will be given to the analytical functional and design specification of Frame Adaptation that has been done on our way to the unit's implementation.

2 Orthogonal Frequency Division Multiplexing

DVB-RCT is a parallel transmission system based on Orthogonal Frequency Division Multiplexing. OFDM is a multiple carrier modulation technique. In this chapter we present an introduction of the basic theory concerning OFDM and its advantages.

2.1 Multi-carrier transmission

The basic idea of multi-carrier transmission is to divide the input symbol stream into several parallel streams used to modulate more than one carrier. The main problem of digital transmission is the inter-symbol interference, which occurs when system frequency is significantly smaller than the duration of the real non-ideal channel's impulse response.

In multi-carrier transmission, this problem is handled by dividing the available channel bandwidth into a number of sub-channels with equal bandwidth. The aim is to select an appropriate sub-channel bandwidth, so that their frequency response characteristics are nearly ideal. Thus, different information symbols can be transmitted in parallel over $N = W/\Delta f$ sub-channels, where W is the entire bandwidth and Δf a sub-channel's bandwidth. Obviously the data transmission is achieved by means of Frequency Division Multiplexing (FDM).

2.2 OFDM concept

OFDM is a form of multi-carrier modulation where its carrier spacing is selected in such a way so that each sub-carrier is orthogonal to the other sub-carriers. It makes a much more efficient use of the spectrum by spacing the channels much closer together. The available bandwidth is split into many narrow-band channels whose carriers are made orthogonal to one another, allowing them to be spaced very close together.

In previous multi-tone techniques the allocated bandwidth of each sub-channel should be made wider than the minimum required amount, to prevent channels from interfering with one another. In OFDM, efficient spectrum use is succeeded through the division of the total bandwidth to sub-channels of orthogonal carriers (Figure 2).

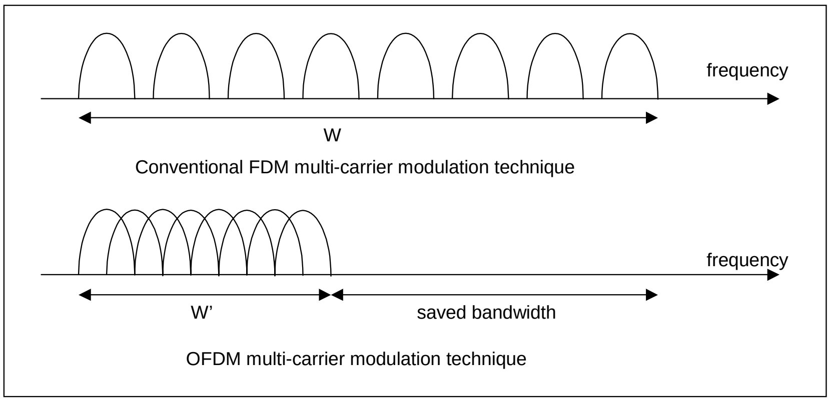

Figure 2 Comparison of the bandwidth utilization between FDM and OFDM

The application of orthogonality between the carriers means that each carrier has an integer number of cycles over a symbol period. As a result, the spectrum of each carrier has a null at the center frequency of each of the other carriers in the system (Figure 3). Since there is no crosstalk from other sub-channels at the central frequency of each sub-channel, there is no interference between them and the carriers can theoretically be spaced as close as possible.

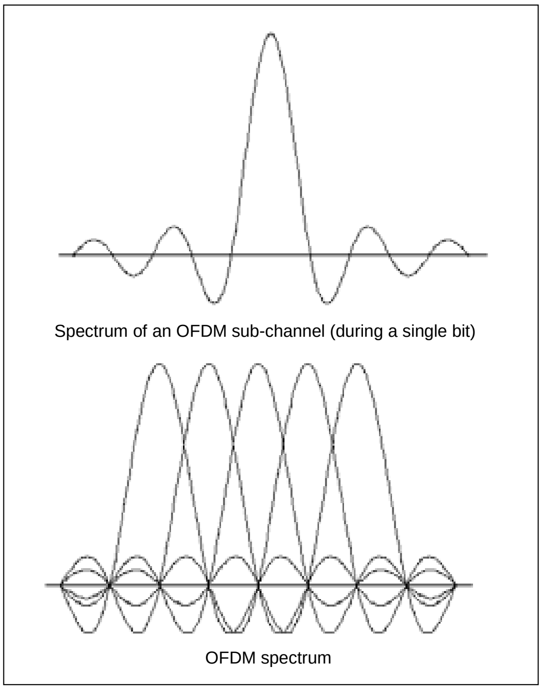

Figure 3 Spectrum of the composite signal (OFDM symbol)

More specifically, lets associate each sub-channel with a carrier $x_n(t) = sin\ 2\pi\ f_n\ t$ where n ranges from 0 to N-1. We have defined N to be the number of the divided sub-channels ($N = W/\ \Delta f$) and we consider f_n to be the middle frequency of the n^{th} sub-channel. If we select Δf to be equal to the symbol rate $1/T$, then the carriers are orthogonal over the symbol period T and there is no dependence between the relative phases of the carriers. This is expressed by the following equation:

$$\int_0^T \sin\left(2\pi f_n t + \varphi_n\right)\ \sin\left(2\pi f_k t + \varphi_k\right) dt = 0\ \text{, where } f_n - f_k = m/T$$

The common point of OFDM and the common Frequency Division Multiplexing Access is that multiple user access is achieved by subdividing the available bandwidth into multiple channels that are then allocated to users. In the OFDM case the bandwidth of the sub-channels is very narrow, a characteristic that results in a low symbol rate. If we define as T_s the symbol duration in the single carrier system, then the symbol interval in the OFDM system is $T = N\ T_s$.

The transmitted signal presents high tolerance to the effect of multi-path delay spread. Multi-paths occur from the random reflection of the signal in various environment obstacles. This results to the receiver detecting multiple echoes of the sent signal. By selecting a significantly large N (usually a power of 2), the symbol duration can be made larger than the time dispersion of the channel. The low symbol rate in OFDM systems resolves the problem since the delay spread must be very long to result to Inter-symbol Interference. Hence, in an OFDM system Inter-symbol Interference is avoided through the division of the available spectrum into many orthogonal carriers, each one being modulated by a low-rate data stream

2.3 OFDM generation

The modulator in an OFDM system is based on the Inverse Discrete Fourier Transform (IDFT) and in reality it is implemented with the use of the Inverse Fast Fourier Transform (IFFT) algorithm.

It is very important to maintain the orthogonality between all the carriers. The selection of the required spectrum for OFDM is based on the input data and the employed modulation scheme. Each carrier is assigned to transmit a number of bits/symbol. Several constellation types can be used to calculate the required amplitude and phase of the carrier. In addition different constellation types permit to adjust the symbol bit capacity according to the Signal to Noise Ratio (SNR) of the channel. The used modulation schemes are typically differential BPSK, QPSK, or QAM of different constellation sizes.

The symbols to be transmitted are mapped to a complex number. The resulting parallel stream of complex weights is used as an input to basic functions associated with IDFT. IFFT performs an efficient transformation of the required spectrum (amplitude and phase of each component) to its time domain signal and it ensures the orthogonality of the produced carrier signals. In other words, an IFFT converts a number of complex data points corresponding to input symbols, whose length is a power of 2, into a time domain signal of the same number of points.

Let us examine the role of IFFT with a mathematical approach. Each carrier can be corresponded to the description of a complex wave, whose real part corresponds to the real signal and its amplitude and phase depend on the transmitted symbol:

$$x_n(t) = A_n(t)e^{j(\omega_n+\varphi_n(t))}$$

where $A_n(t) = A_n$ and $\varphi_n(t) = \varphi_n$ are constant over the symbol duration τ and $\omega_n = \omega_0 + n\Delta\omega$ or, without a loss of generality $\omega_n = n\Delta\omega$. Considering the above as well as the fact that OFDM consists of many carriers, if the signal is sampled over the period of one data symbol ($\tau = NT$) we have the following relationship:

$$s(kT) = \frac{1}{N}\sum_{n=0}^{N-1} A_n e^{j(n\Delta\omega kT+\varphi_n)} = \frac{1}{N}\sum_{n=0}^{N-1} A_n e^{j\varphi_n} e^{jn\Delta\omega kT}$$

The inverse Fourier transform of the signal *s(kT)* would be described by the following equation:

$$s(kT) = \frac{1}{N}\sum_{n=0}^{N-1} S\left(\frac{n}{NT}\right)e^{j2\pi nk/N}$$

If we compare the two expressions of *s(kT)*, we observe that $A_n e^{j\varphi_n}$ is a definition of the signal in the sampled frequency domain. Therefore, if we consider $A_n e^{j\varphi_n} = S\left(\frac{n}{NT}\right)$, in order for the two equations to be equivalent the following relationship must be true:

$$\Delta f = \frac{\Delta\omega}{2\pi} = \frac{1}{NT} = \frac{1}{\tau}$$

This relationship coincides with the condition that is required for orthogonality. Therefore, since we wish to maintain orthogonality, we conclude that the OFDM signal can be defined by using Fourier transform procedures.

In the previous paragraph it has been discussed how the robustness against multi-path delay spread is achieved by having a long symbol period. The tolerance level to ISI can be increased further, by the addition of a guard period between transmitted symbols. That way, there is enough time from a previous symbol's multi-path signals to fade before the current symbol's information is gathered. This guard period can be achieved in practice with the cyclic extension of the symbol. If a mirror in time, of the end of the symbol waveform is put at the start of the symbol, the length of the symbol is extended, while the orthogonality of the waveform is maintained. The addition of a guard period with the use of this cyclic extended symbol, permits the FFT to decode the symbol by taking the required samples anywhere over the length of the symbol. This provides multi-path immunity as well as symbol time synchronization tolerance.

2.4 OFDM in DVB-RCT

The OFDM technique is applied in our system by partitioning the VHF/UHF return channel both in the frequency and time domains, using Frequency Division Multiplex (FDM) and Time Division Multiplex (TDM).

DVB-RCT supports three different carrier spacing values (~1KHz, ~2KHz, ~4KHz) and there is also an option between two OFDM carriers sets of 1024 or 2048 carriers. That way the system can operate in six different transmission modes depending on the combination of carrier spacing and OFDM carrier set. In addition the applied guard interval can be the 1/4, 1/8, 1/16, 1/32 of the OFDM symbol.

An OFDM symbol consists of the symbols transmitted over all the carriers. Visually, consecutive OFDM symbols form a frequency-time grid. Our interest is to allow access to multiple users. Any terminal/user can contribute to the content of the grid; the available carriers are allocated to certain users for a defined time slot. The Base Station is responsible for the allocation policy. The goal is to serve a big number of users, in parallel, imposing an efficient usage of the available spectrum.

3 Basic principles and definitions

Before continuing with the stages of the Physical Layer and the detailed description of the Frame Adaptation, we will devote this chapter to present some information related to the DVB-RCT system operation, including the employed modulation and the available transmission modes. It is also necessary to introduce the concepts and definitions regarding the Transmission Frame (TF), the Medium Access Schemes (MAS), the Burst Structures (BS) and their formatting, as well as the Ranging signals.

3.1 Transmission modes

There are six different transmission modes depending on the maximum number of used carriers (1K, 2K) and their inter-carrier distance (CS1, CS2, CS3) but they cannot be mixed: only one shall be used. The inter-carrier distance is necessary for the system protection against synchronization misalignment (caused by the Doppler shift). The wider the spacing, the more robust the system is. The whole DVB-RCT Radio Frequency channel shall be populated with either 1 024 (1K) carriers or 2 048 (2K) carriers.

We present the following definitions which will make the transmission modes analysis more clear and which will also show their dependency from the selected carrier spacing and maximum number of used carriers.

- **Total system carriers (T_{sc}):** It is the total number of carriers managed by DVB-RCT. As already mentioned, the frequency channel can be populated by 1024 carriers (1K mode) or 2048 carriers (2K mode).
- **Used carriers (C_u):** It is the maximum number of carriers that can be effectively used. It results by removing from T_{sc}, the guard band carriers. The excluded carriers are the extreme ones, which remain unused, forming that way a guard band for the protection of the adjacent channels. The number of used carriers is 1712 in 2K or 842 in 1K.
- **Useful symbol duration (T_u):** It is the useful duration of the modulated symbol.
- **RCT system clock (T):** It is the DVB-RCT system clock and it is derived from the DVB-T downstream. A reference clock is defined depending on whether DVB-T is an 8 MHz, 7 MHz or 6 MHz system. In our application, we consider an 8 MHz DVB-T system, which reflects the European standard and implies a 64/7 MHz reference clock. The RCT system clock is defined as:

- ➢ four times the DVB-T system clock period in case of CS1
- ➢ two times the DVB-T system clock period in case of CS2
- ➢ one time the DVB-T system clock period in case of CS3

It is expressed by the formula $T = T_u / T_{sc}$. Therefore the relation of T in 1K and 2K mode for a given carrier spacing is $T_{1K} = 2 \times T_{2K}$.

- **Carrier Spacing (C_s):** It is the inter-carrier distance and is expressed as $C_s = 1 / T_u$.
- **RCT channel bandwidth (B_u):** It is the DVB-RCT channel bandwidth and is calculated by the formula $B_u = C_s \times C_u$.

	8 MHz DVB-T system, reference clock 64/7 MHz (0.109 µsec)		
	Total system carriers	**2048**	**1024**
	Used carriers	**1712**	**842**
CS1	RCT system clock	4 x T = 0.438 µsec	8 x T = 0.875 µsec
	Useful symbol duration	896 µsec	
	Carrier spacing	1116 Hz	
	RCT channel bandwidth	1.911 MHz	0.940 MHz
CS2	RCT system clock	2 x T = 0.219 µsec	4 x T = 0.438 µsec
	Useful symbol duration	448 µsec	
	Carrier spacing	2232 Hz	
	RCT channel bandwidth	3.821 MHz	1.879 MHz
CS3	RCT system clock	1 x T = 0.109 µsec	2 x T = 0.219 µsec
	Useful symbol duration	224 µsec	
	Carrier spacing	4464 Hz	
	RCT channel bandwidth	7.643 MHz	3.759 MHz

Table 1 DVB-RCT transmission mode parameters for the 8MHz DVB-T system

The table above lists the values of the various transmission parameters as calculated from the above definitions. It must be made clear that the parameters that actually define the six transmission modes are the carrier spacing and the number of system carriers. For different DVB-T systems (7 MHz or 6 MHz) the value of a selected carrier spacing varies a little. In the following calculation the exact carrier spacing value is presented, but for practical reasons from now on we will refer to the three types with their approximate targeted values, which are 1 KHz for CS1, 2 KHz for CS2 and 4 KHz for CS3. As a conclusion, we can say that, each transmission mode has a trade-off between frequency diversity and time diversity and then between coverage range and portability capability.

The transmitted symbol's actual duration depends on the shaping function applied to the carriers. In case of Rectangular Shaping the useful symbol duration is increased by the guard interval duration, which can be 1/4, 1/8, 1/16 or 1/32 of the useful symbol duration. If the applied shaping is Nyquist then the useful symbol duration does not actually have a physical signification but the total symbol duration is 1.25 the inverse of the carrier spacing. For later design purposes we can conceive the total symbol duration in an equivalent way with the Rectangular Shaping case: it is the useful symbol duration multiplied with the factor 1.25, as if a guard interval of ¼ was inserted.

3.2 Adaptive modulation

One of the main features of the DVB-RCT is the capability for dynamically assigned adaptive modulation. Different types of modulation are supported within the same sell. These types are defined by the selected combination of constellation type and coding rate.

The available constellation types are QPSK, 16QAM or 64QAM, whereas the coding rate can be 1/2 or 3/4. The combination of constellation type and coding rate defines a symbol's bit capacity, thus, the data payload within a burst. Consequently, they are two of the factors that define the robustness of the system depending on the SNR. Since different modulation types can co-exist in the same cell, the users positioned near the outer limits of the cell can be assigned a more robust form (i.e. QPSK, ½), which allows them to use the minimum possible amount of power. On the other hand, users near the centre of the cell can use more power and enjoy high data throughput, since they are further away from the other cells and will cause less interference. The following table lists the data payload of a burst in relation to the constellation type and modulation rate.

	QPSK		16QAM		64QAM	
Encoding rate R	½	¾	½	¾	½	¾
Data payload of a burst (bytes)	18	27	36	54	54	81

Table 2 Useful data payload of a burst

3.3 Sub-carrier Shaping

The aim of sub-carrier shaping is to provide additional protection from inter-carrier and inter-symbol interference. Two kinds of shaping are supported and are never mixed but used alternatively: Nyquist Shaping and Rectangular Shaping.

3.3.1 Nyquist Shaping

It provides immunity against jammers and inter-carrier, inter-symbol interference by applying in time Nyquist filtering. The transmission of a user over a carrier can pollute the transmission of another user over neighbouring carriers. With Nyquist Shaping the carrier spectra are disjoint. That way the effect of misalignment is limited. This kind of shaping is recommended in case of large cells were the users are far away from each other and from the Base Station.

3.3.2 Rectangular Shaping

It provides immunity against inter-carrier, inter-symbol interference by the orthogonal arrangement of the carriers and the insertion of Guard Intervals between the modulated symbols. It also deals with the problem of multi-path propagation effects. The use of orthogonal carriers allows their spectra to overlap, but it also allows the correct demodulation and separation of data. On the other hand it is more sensitive to misalignments. This kind of shaping is recommended for small cells in dense networks.

3.4 OFDM symbol

The Transmission Frame is a repetitive structure transmitted over the radio-frequency channel. It consists of time-frequency slices, which contain various symbols required for three basic tasks: ranging, synchronization and data transmission. We will define the term OFDM symbol as a vertical time-frequency slice.

Depending on the transmission mode in operation, one OFDM symbol is made of 2048 carriers (2K mode) or 1 024 carriers (1K mode). Not all carriers are used. Some carriers remain unused and are located on each edge of the channel, providing a guard band to protect adjacent channels. From the total number of 2K (2048) FFT points, 1712 are usable carriers and there is also a lower and upper guard band of 168 points each. Correspondingly for 1K (1024 carriers), 842 carriers are used and the guard bands have a width of 91 carriers each. An OFDM symbol carries a symbol per carrier from one of the following categories:

- Ranging symbols: they are needed to establish the connection, request access or extra bandwidth.
- Pilot symbols: they are used for channel estimation at the receiver end. Along with the data symbols they are considered user symbols and are required for the data transmission.
- Data symbols: they are the actual data. The first time the data symbol concept appears is at the output of the Interleaver stage (presented in a later chapter) after the data stream being processed by the previous stages of the chain.
- Null symbols: they are used to fill the guard bands and the unused points (even entire OFDM symbols).

In the Frame Adaptation block analysis the terms OFDM symbol and time slice are synonym. The same stands for the term symbol and point.

3.5 Transmission Frame

As already mentioned, Transmission Frame is a repetitive structure transmitted over the radio-frequency channel and is consisted of OFDM symbols. Two types of Transmission Frames are defined, depending on whether the OFDM symbols are dedicated to one or more activities (ranging, synchronization, data transmission).

3.5.1 TF1

The first type of Transmission Frame is organized in the time domain. Therefore each OFDM symbol is dedicated to a unique activity. The structure of TF1 is described below and depicted in the first part of Figure 4:

- The first OFDM symbol is a slice of null symbols. During that period no transmission occurs in order to allow the Base Station to detect jammers.

– A group of OFDM symbols follows, carrying Ranging symbols, information for the establishment and conservation of the connection.

– At the end come the OFDM symbols with the user symbols, which transmit the data bursts.

The duration of TF1 (length in OFDM symbols) is the sum of the Ranging Frame duration, plus the Data Frame duration, plus one null OFDM symbol.

3.5.2 TF2

The second type of Transmission Frame is organized in the frequency domain. Each OFDM symbol is of general purpose and does not serve a unique activity. The carriers (FFT points of the OFDM symbol) are divided into sub-channels, and each sub-channel is dedicated either to carry ranging functions or data bursts. In order for the sub-channels to synchronize, the OFDM symbols are organized in the time domain into eight groups of six OFDM symbols. The structure of TF2 is depicted in Figure 4. The duration of TF2 is always 8 x 6 = 48 OFDM symbols.

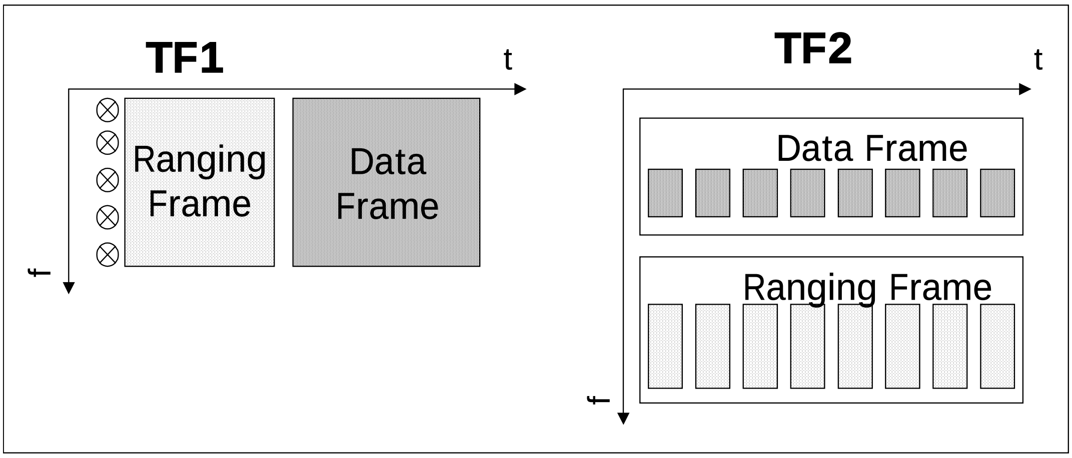

Figure 4 Organization of Transmission Frame 1 and 2

3.6 Burst Structure

Each terminal transmits bursts of data based on an integer number of ATM cells. Whatever the encoding and the physical modulation, the data bursts are made up of 144 data symbols, which contain the user data. A collection of pilot symbols is spread among the data symbols to allow coherent demodulation in the Base Station. Three Burst Structure types are defined according to the partitioning of the data bursts and the pilots among the time-frequency axes. The choice of the burst structure type used is a trade-off between frequency diversity and burst duration. Small burst duration provides the system robustness against interference but it also requires the allocation of more carriers to a single user, therefore fewer users can be served. A draft comparative illustration of the three types placement in frequency and time appears in the figure below.

The duration of a Burst Structure shall be called from now on **Time Slot**.

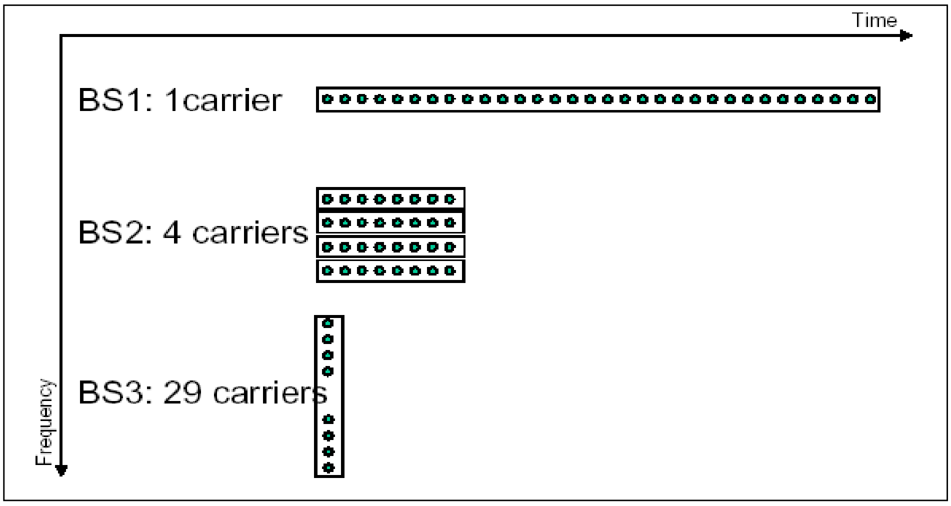

Figure 5 Burst Structures' comparative placement in frequency and time

3.6.1 BS1

Burst Structure 1 places the data consecutively in time over a unique carrier. The duration of BS1 is 180 OFDM symbols. Each BS1 comprises of four consecutive mini-bursts (carrying Data, Boosted and Non Boosted Pilot symbols) and each mini-burst makes use of one carrier at a time. The 144 data symbols of the data payload are split into 4 mini-bursts of 36 data symbols and 9 pilot symbols each. The 2 of them are non-boosted (one at the beginning and the end of each mini-burst) and the other 7 are boosted. A boosted pilot symbol is transmitted every five Data Symbols (Boosted Pilot Insertion Ratio 1/6). The various parameters that define the internal format of BS1 is:

- Frequency Hopping option: dictates whether the mini-bursts will be transmitted over a unique carrier or each one over a separate carrier.

- Sub-Channel Numbering (SCN): defines implicitly the sub-channel (used carriers) allocated for the transmission of the data burst (carrier allocation will be presented analytically in a following paragraph).
- Used carrier value modulo 4: the location of the carrier affects the distribution pattern of the symbols inside a mini-burst, as illustrated in the following figure.

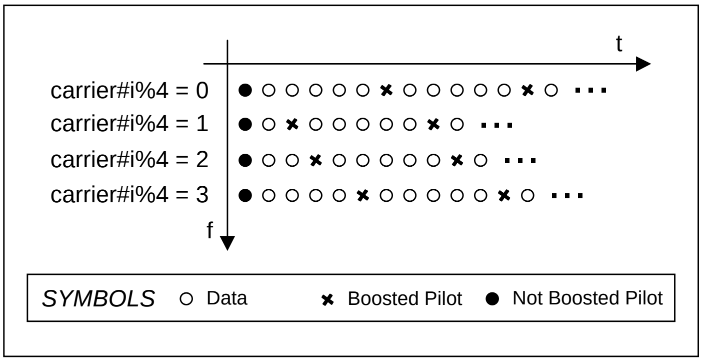

Figure 6 Symbol pattern dependency to used carrier value modulo 4

3.6.2 BS2

Burst Structure 2 places the data consecutively in time and over four carriers. The duration of BS2 is 45 OFDM symbols. Each BS2 comprises of four mini-bursts (carrying Data, Boosted and Non Boosted Pilot symbols) transmitted in parallel. The 144 data symbols of the data payload are split into 4 mini-bursts of 36 data symbols and 9 pilot symbols each, the same way it is done in BS1. The various parameters that define the internal format of BS2 is (as for BS1):

- Sub-Channel Numbering (SCN).
- Used carrier value modulo 4.

3.6.3 BS3

Burst Structure 3 places the data consecutively in frequency over 29 carriers. The duration of BS3 is 6 OFDM symbols. The 144 data symbols of the data are organized in a “block” with 30 boosted pilot symbols spread among them in a specific pattern (illustrated in the following figure). The Boosted Pilot Insertion Ratio is also approximately 1/6. The only parameter that defines the internal format of BS3 is Sub-Channel Numbering (SCN).

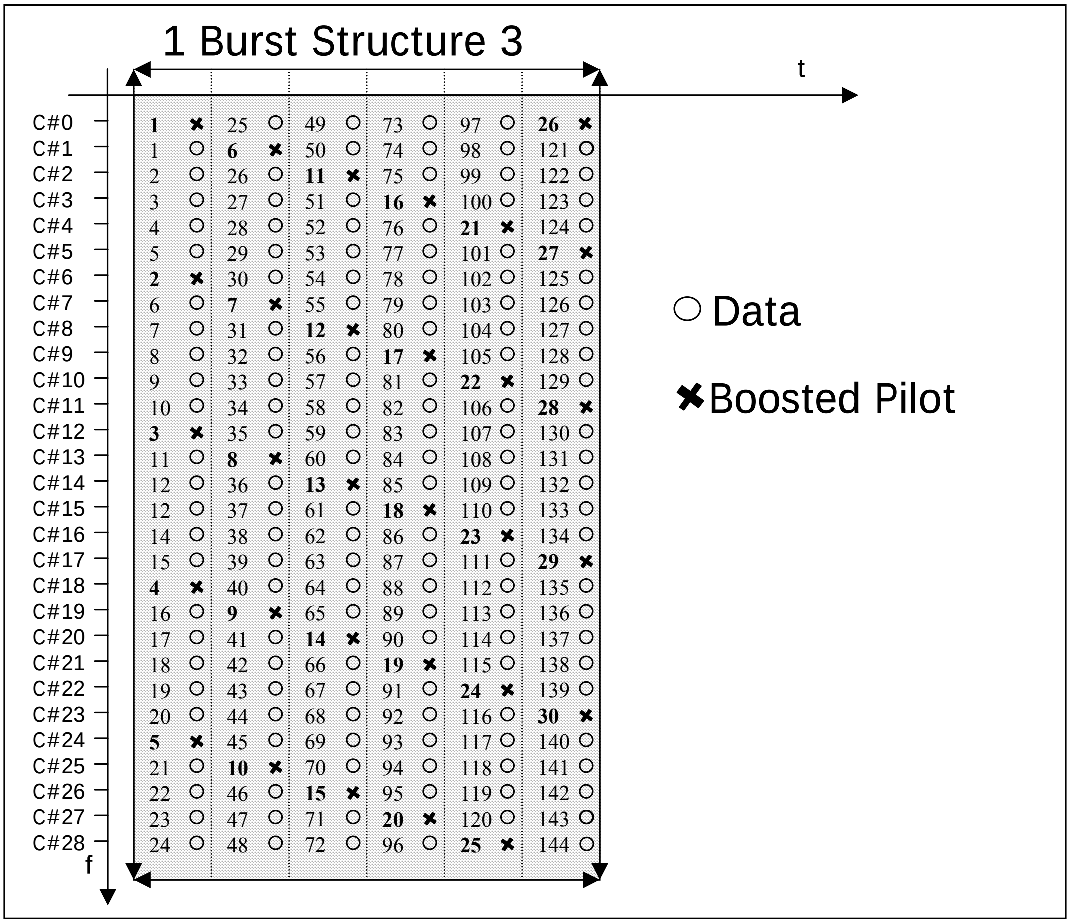

Figure 7 Burst Structure 3 symbol pattern

3.7 Medium Access Schemes

The mapping of the burst structures into the Transmission Frame depends on the transmission parameters of the system and the allocated carriers for the specific terminal. The Base Station provides this information, through a MAC (Medium Access Control) process. Three methods of mapping exist and are called Medium Access Schemes. During the presentation of these three schemes, the formatting of the Burst Structures will be studied in more detail.

3.7.1 MAS1

Medium Access Scheme 1, maps in time one Burst Structure 1 onto Transmission Frame 1. Thus TF1 in MAS1 contains one Time Slot. The duration of a Time Slot in MAS1 is at least 180 OFDM symbols. The constellation type and carrier shaping function dictate the number of bursts inside a Time Slot. Due to memory space limitations, as it will be explained later, for the specific implementation, we can have up to 128 BS1s per Time Slot, transmitted in parallel.

The various options that influence the format of the TF are:

- *Rectangular or Nyquist Shaping:* defines whether Nyquist pre- and post-amble symbols are going to be transmitted. The Nyquist symbols are not real symbols. Their function is to reserve space needed for the filter process, in case of Nyquist Shaping. For our purposes the Nyquist symbols are null symbols.
- *Frequency Hopping option:* dictates whether transmission will take place over four or one carrier.
- *Sub-Channel Numbering (SCN):* defines implicitly the carriers used by a burst structure.
- *Used carrier value modulo4:* the location of the carrier affects the distribution pattern of the symbols inside a mini-burst.

In the following figures, only one BS per Time Slot is illustrated (the actual capacity is 128 BS1s in parallel). Also at this point our interest is concentrated to the data part of TF1. Ranging transmission will be presented later.

3.7.1.1 Rectangular Shaping, without Frequency Hopping

When transmission is performed with Rectangular Shaping and without Frequency Hopping, the BS is transmitted over a unique carrier and its duration is 180 OFDM symbols. The transmitted symbols are Data, Not Boosted Pilots, and Boosted Pilots. The TF appears in the following figure. In the depicted example, it is assumed carrier mod 4=0.

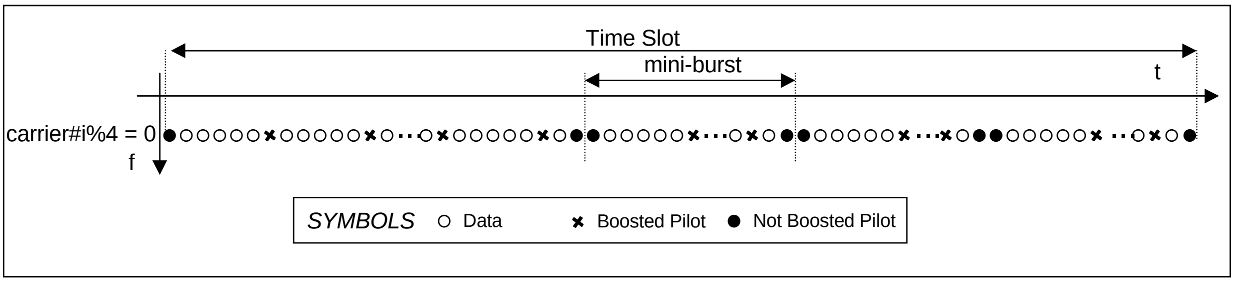

Figure 8 Burst Structure formatting inside the Transmission Frame for MAS1 without Frequency Hopping, with Rectangular Shaping and carrier#i%4=0

3.7.1.2 Rectangular Shaping, with Frequency Hopping

When transmission is performed with Rectangular Shaping and Frequency Hopping, the BS is transmitted over a Sub-Channel of four carriers, one carrier per mini-burst and its duration is again 180 OFDM symbols. The transmitted symbols are Data, Not Boosted Pilots and Boosted Pilots. The TF appears in the figure below.

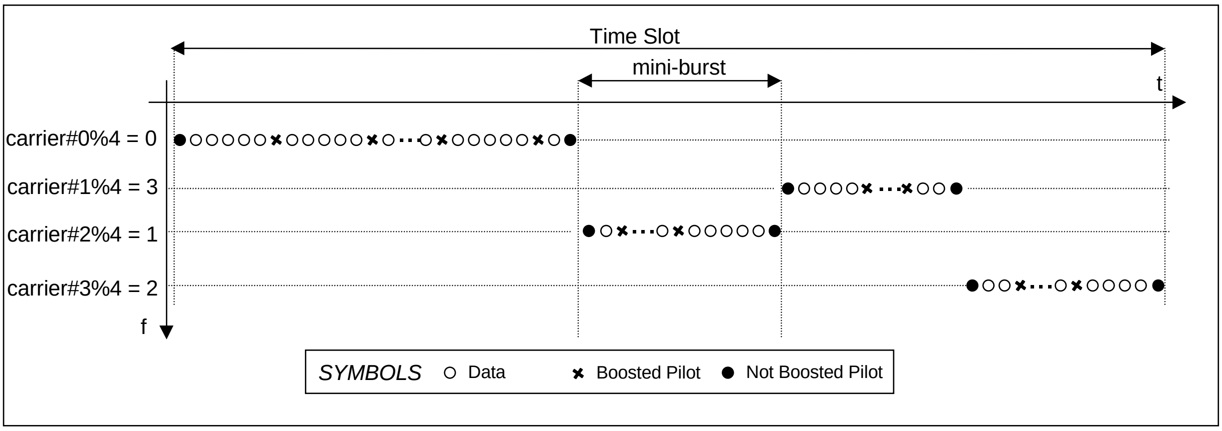

Figure 9 Burst Structure formatting inside the Transmission Frame for MAS1 with Frequency Hopping, Rectangular Shaping and various carrier#i%4

3.7.1.3 Nyquist Shaping, without Frequency Hopping

When transmission is performed with Nyquist Shaping and without Frequency Hopping, the BS is transmitted over a unique carrier but due to the insertion of Nyquist symbols its duration is now 180 + 8 = 188 OFDM symbols. The transmitted symbols are Data, Not Boosted Pilots, Boosted Pilots, Nyquist Pre- and Post-amble symbols. The TF appears in the above figure. The formatting is identical to the case of Rectangular Shaping without Frequency Hopping, with the unique difference that the Burst Structure is preceded by four pre-amble and followed by four post-amble Nyquist symbols. In the depicted example it is assumed carrier mod 4=0.

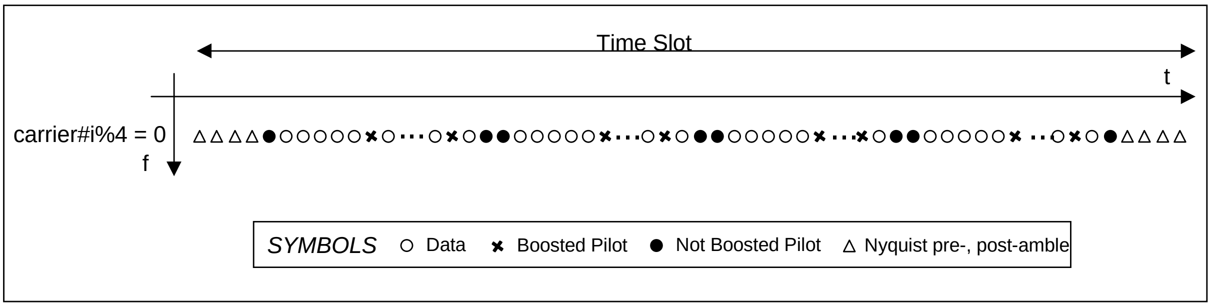

Figure 10 Burst Structure formatting inside the Transmission Frame for MAS1 without Frequency Hopping, with Nyquist Shaping and carrier#i%4=0

3.7.1.4 Nyquist Shaping, with Frequency Hopping

When transmission is performed with Nyquist Shaping and Frequency Hopping, the BS is transmitted over a Sub-Channel of four carriers, one carrier per mini-burst but due to the insertion of Nyquist symbols at each mini-burst its duration is now 180 + 32 = 212 OFDM symbols.. The transmitted symbols are Data, Not Boosted Pilots, Boosted Pilots, Nyquist Pre- and Post-amble symbols. The formatting is identical to the case of Rectangular Shaping with Frequency Hopping, with the unique difference that each mini-burst is preceded by four pre-amble and followed by four post-amble Nyquist symbols. The TF appears in the figure below.

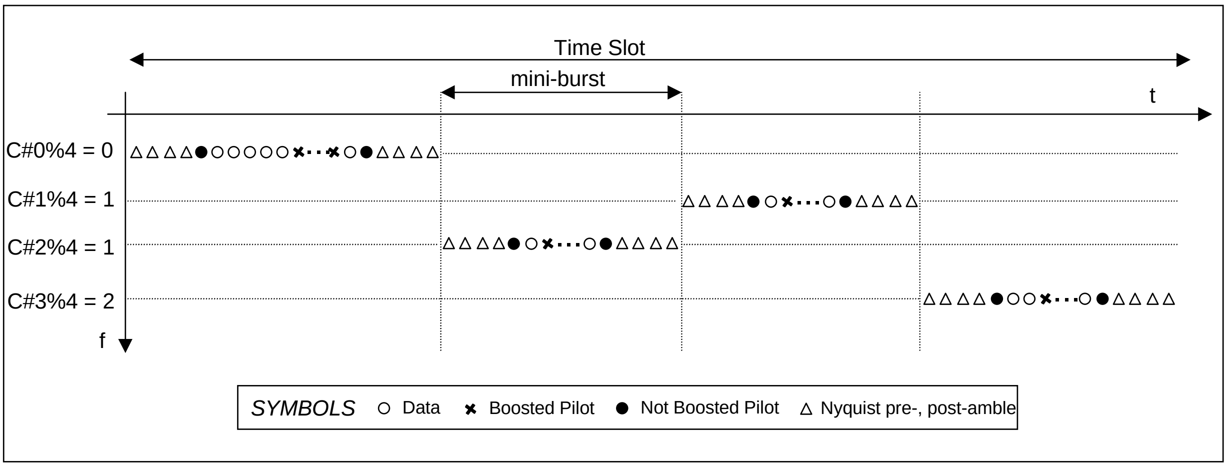

Figure 11 Burst Structure formatting inside the Transmission Frame for MAS1 with Frequency Hopping, Nyquist Shaping and various carrier#i%4

3.7.2 MAS2

Medium Access Scheme 2, maps in time four Burst Structures 2, onto Transmission Frame 1. The duration of a Time Slot in MAS2 is 45 OFDM symbols. TF1 must have the same duration independently of the MAS. Thus TF1 in MAS2 contains four Time Slots. Inside a TF we can have up to 128 BS2s per Time Slot, transmitted in parallel for the same reasons as in MAS1.

Only Rectangular Shaping is performed in the present implementation. The various options that influence the format of the transmission frame are:

- *Sub-Channel Numbering (SCN).*
- *Used carrier value modulo 4.*

When transmission of a MAS2 Time Slot is performed, a BS2 is transmitted over a four carrier Sub-Channel and its duration is 45 OFDM symbols. The transmitted symbols are Data, Not Boosted Pilots and Boosted Pilots. As we can see the formatting of the mini-bursts is identical to the one used in MAS1. The difference is that the four mini-bursts comprising a Burst Structure 2 are transmitted in parallel. In the figure, only one BS per Time Slot is illustrated. Also at this point out interest is concentrated to the data part of TF1. Ranging transmission will be presented later.

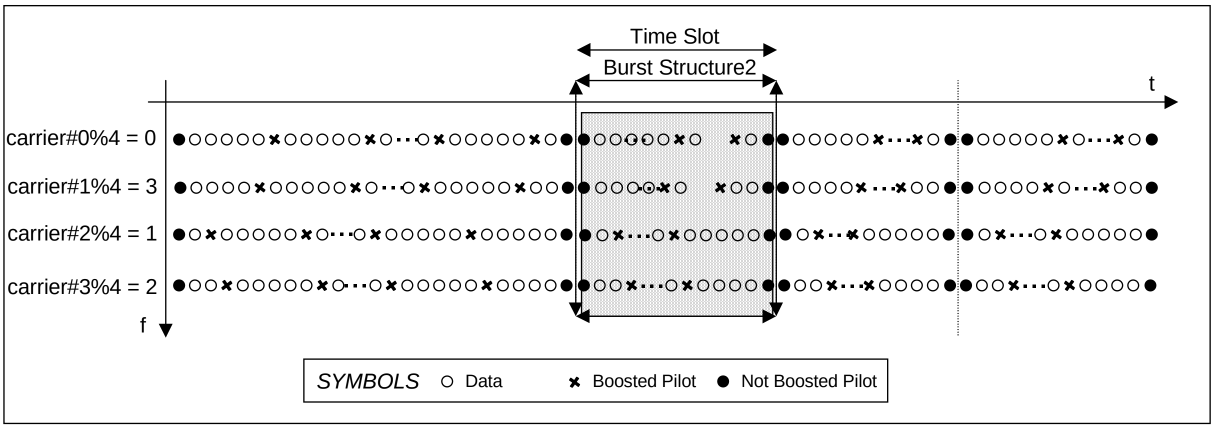

Figure 12 Burst Structure formatting inside the Transmission Frame for MAS2 with Rectangular Shaping and various carrier#i%4

3.7.3 MAS3

Medium Access Scheme 3, maps in time eight Burst Structures 3 onto Transmission Frame 2. Thus TF2 in MAS3 contains eight Time Slots. The duration of a Time Slot in MAS3 is 6 OFDM symbols. In a TF2 we can have up to 54 BS3s per Time Slot transmitted, in parallel, in different Sub-Channels. Nyquist Shaping is not applicable in MAS3.

The unique option that influences the format of the transmission frame is *Sub-Channel Numbering (SCN).*

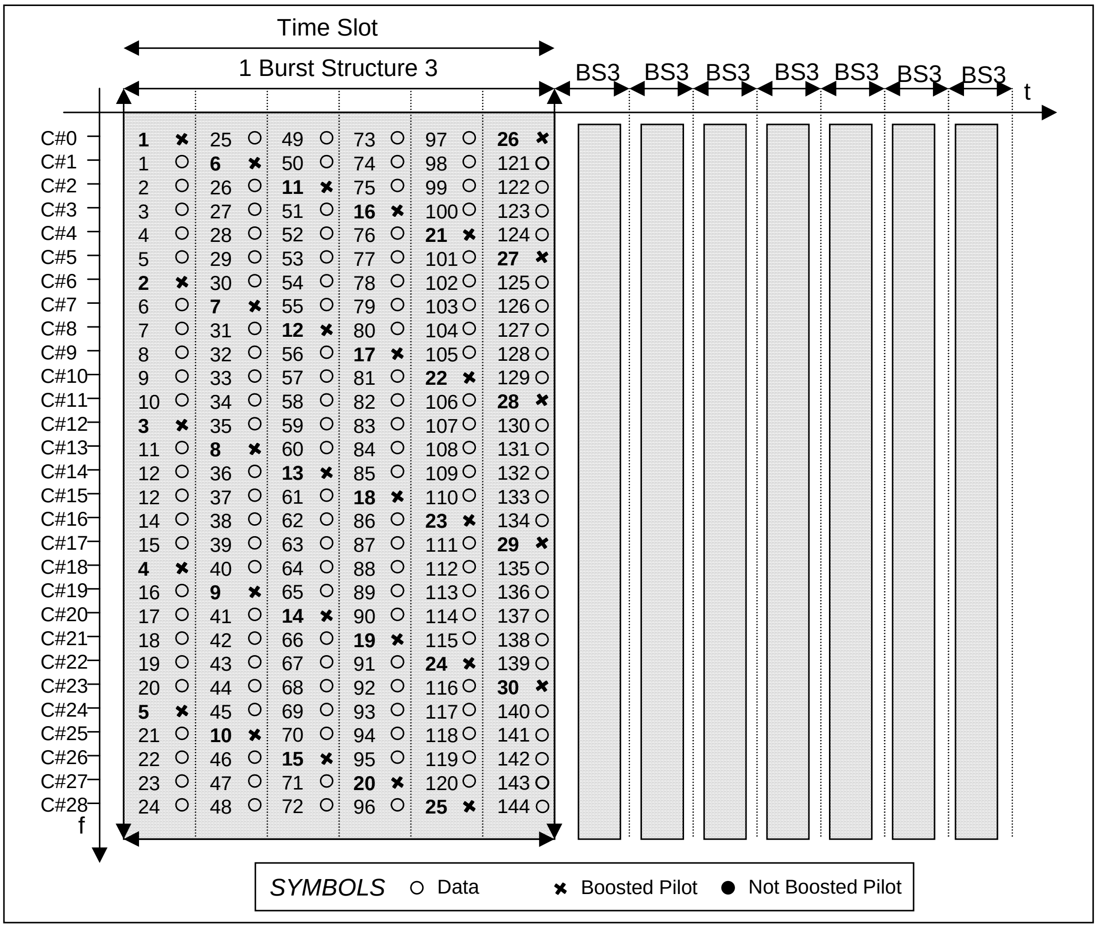

Figure 13 Transmission Frame of MAS3 with Burst Structure 3 mapped

3.8 Carrier Allocation

Each Burst Structure is transmitted over a sub-channel and more than one Burst Structures can be transmitted in parallel over different sub-channels. The carrier allocation is the process that calculates the carriers of a sub-channel. A sub-channel is defined by a Sub-Channel Number (SCN), which is used for the carrier calculation. For each transmitted Burst Structure a sub-channel is occupied and the respective SCN is given. The used carriers are calculated from the SCN with various algorithms depending on the Medium Access Scheme and the Transmission mode (1K or 2K). The numbering of the allocated carriers corresponds to the range of the carriers where transmission is permitted. More specifically, as already mentioned in section 3.4 , from the 2048 carriers of the 2K mode, 1712 are usable whereas the rest form two guard bands. Therefore the range of the allocated carriers in 2K mode is from 0 to 1711. Equivalently, in the 1K mode the range is from 0 to 841, because from the 1024 carriers 842 are usable. The following table describes the general carrier distribution.

	2K mode	**1K mode**
Virtual number of carriers in OFDM	2048	1024
Overall carriers	1712	842
Guard Band on each side	168	91
Usable carriers' numbering	0 to 1711	0 to 841
DC carrier	856	421

Table 3 Carrier distribution in an OFDM symbol

For the treatment of the Nyquist Shaping the same carriers numbering is used, but the OFDM symbol's points are less. More specifically, we consider the carriers of the OFDM symbol to be 256, instead of 1K or 2K. The maximum permitted number of allocated Sub-Channels is 64 and their Sub-Channel Numbers must be consecutive. That way all the used carriers are concentrated in a narrow range of used carriers that is defined by the first Sub-Channel. When frequency hopping is applied, the position of this band of carriers varies because the first SCN is interpreted to four carriers. The concept of the narrow OFDM symbol is introduced so that the carrier allocation is treated by FRM, equivalently for both shaping types. The final number of points will be again 1K or 2K by inserting 4 or 8 null carriers respectively, after each carrier of the described "short" time slice. The carrier distribution table for Nyquist Shaping appears below and it is a function of the first allocated carrier.

	Nyquist Shaping
First allocated carrier	x
Virtual number of carriers in OFDM	256
Overall carriers	64 (x to x+63)
Guard Band on each side	96 (x-96 to x-1 and x+64 to x+159)
central carrier	x+31

Table 4 Carrier distribution in a shorter OFDM symbol for Nyquist Shaping

3.8.1 Medium Access Scheme 1

In Medium Access Scheme 1, BS1s are transmitted. The Sub-Channel Number has frequency dimensions and it is employed for the calculation of the Sub-Channel's carriers used for a Burst Structure's (BS1) transmission. The only parameter, besides SCN, that influences the carrier allocation is whether there is Frequency Hopping.

- **Without Frequency Hopping:** Calculations for the carrier allocation are not necessary as the frequency of the unique carrier is given by SCN.
- **With Frequency Hopping:** A sub-channel consists of four carriers in total but only one is occupied per mini-burst.

Sub-Channel for 2≤SCN≤428	Sub-Channel for 429≤SCN≤855	Sub-Channel for 857≤SCN≤1283	Sub-Channel for 1284≤SCN≤1710
Carrier SCN	Carrier SCN	Carrier SCN	Carrier SCN
Carrier SCN+855	Carrier SCN+855	Carrier SCN-428	Carrier SCN-1282
Carrier SCN+427	Carrier SCN-427	Carrier SCN+427	Carrier SCN-427
Carrier SCN+1282	Carrier SCN+428	Carrier SCN-855	Carrier SCN-855

Table 5 Frequency Hopping Law for 2K mode

Sub-Channel for 1≤SCN≤210	Sub-Channel for 211≤SCN≤420	Sub-Channel for 422≤SCN≤631	Sub-Channel for 632≤SCN≤841
Carrier SCN	Carrier SCN	Carrier SCN	Carrier SCN
Carrier SCN+421	Carrier SCN+421	Carrier SCN-211	Carrier SCN-631
Carrier SCN+210	Carrier SCN-210	Carrier SCN+210	Carrier SCN-210
Carrier SCN+631	Carrier SCN+211	Carrier SCN-421	Carrier SCN-421

Table 6 Frequency Hopping Law for 1K mode

Carrier allocation is performed for 2K and 1K mode as described in Table 5 and Table 6, respectively. Table 7 lists the various parameters related to the carrier allocation and distribution in MAS1.

	2K mode	1K mode
Usable carriers	1708	840
Excluded carrier numbers	0, 1, 856 (dc), 1711	0, 421(dc)
SCN range	2-855 and 857-1710	1-420 and 422-841

Table 7 Carrier allocation parameters for MAS1 (BS1)

3.8.2 Medium Access Scheme 2

In Medium Access Scheme 2, BS2s are transmitted. A sub-channel for the BS2 consists of four carriers. The Sub-Channel Number has frequency dimensions and the carrier allocation of the used Sub-Channel is done using the formulas:

$$Carrier\#0 = SCN$$

$$Carrier\#1 = SCN+X$$

$$Carrier\#2 = SCN+Y$$

$$Carrier\#3 = SCN+Z$$

where the dependence of X, Y, Z from the SCN range is given in Table 8. Table 9 lists the various parameters related to the carrier allocation and distribution in MAS2.

	2K mode	1K mode
SCN	2 to 428	1 to 210
X	427	210
Y	855	421
Z	1282	631

Table 8 Sub-Channel Numbering range and definition of auxiliary variables

	2K mode	1K mode
Usable carriers	1708	840
Excluded carrier numbers	0, 1, 856 (dc), 1711	0, 421(dc)
SCN range	2-428	1-210

Table 9 Carrier allocation parameters for MAS2 (BS2)

3.8.3 Medium Access Scheme 3

In MAS3 the carrier allocation for the transmission of a BS3 is more complicated. The following table lists the various carrier allocation parameters.

	2K mode	1K mode
Usable carriers	1710	840
Excluded carrier numbers	856	421
SCN range	0, 10, 20, ...580	0, 10, 20, ...280

Table 10 Carrier allocation parameters for MAS1 (BS1)

3.8.3.1 2K Mode

In 2K mode the total number of usable carriers over which the Burst Structures are transmitted is 1712. That range is divided into 59 Sub-Channels of 29 carriers each (59x29=1711 and the DC carrier which is not used).

The carrier allocation for each Sub-Channel is computed using a permutation of the following series of numbers:

11, 3, 33, 9, 40, 27, 2, 22, 6, 7, 18, 21, 54, 4, 44, 12, 14, 36, 42, 49, 8, 29, 24, 28, 13, 25, 39, 16, 58, 48, 56, 26, 50, 19, 32, 57, 37, 53, 0, 52, 41, 38, 5, 55, 15, 47, 45, 23, 17, 10, 51, 30, 35, 31, 46, 34, 20, 43, 1

(initial series)

A new series results by rotating the previous series by one to the left. From each series we keep the first 29 numbers. That way we end up with 59 sets of 29 numbers each (indexed 0 to 28). We assume that one set corresponds to each Sub-Channel. The carriers are calculated with the following formula:

$$Carrier\#n = 59 \times n + (Index(n) + Unique\_Key) \bmod 59$$

and

$$\text{if } Carrier\#n \geq 856 \text{ then } Carrier\#n = Carrier\#n + 1$$

where:

- *n* the index of each number inside the set (0 to 28)
- *Index(n)* is the actual number which corresponds to index n. For example Index(0) for the initial series of numbers is 11.
- *Unique_Key* denotes a key (0 to 255), provided by the MAC process and which will be unique to each Upstream Channel

The result is 59 Carrier-Sets/Sub-Channels (0 to 58) of 29 carriers each (0 to 28). To compute a Sub-Channel from a SCN, we rotate the initial series by SCN times. SCN is just increasing, meaning 0, 1, 2, ... 58 for the 1st, 2nd, 3rd, … 59th Sub-Channel.

3.8.3.2 1K Mode

In 1K mode the total number of usable carriers over which the Burst Structures are transmitted is 842. That range is divided into 29 Sub-Channels of 29 carriers each (29x29=841 and the DC carrier which is not used.

The carrier allocation for each Sub-Channel is computed using a permutation of the following series of numbers:

10, 13, 14, 24, 8, 22, 17, 25, 18, 6, 2, 20, 26, 28, 19, 16, 0, 15, 5, 21, 7, 12, 4, 11, 23, 27, 9, 3, 1

(initial series)

The method is proportional to the one described in 2K mode. From that series we extract 29 cyclic permutations. Each new series results by rotating the basic one to the left. We assume that one series corresponds to each Sub-Channel. The carriers are calculated with the following formula:

$$Carrier\#n = 29 \times n + (Index(n) + Unique\_Key) \bmod 29$$

and

$$\text{if } Carrier\#n \geq 421 \text{ then } Carrier\#n = Carrier\#n + 1$$

where:

- *n* the index of each number inside the set (0 to 28)
- *Index(n)* is the actual number which corresponds to index n
- *Unique_Key* denotes a key, provided by the MAC process and which will be unique to each Upstream Channel

The result is 29 Carrier-Sets/Sub-Channels (0 to 28) of 29 carriers each (0 to 28). To compute a Sub-Channel from a SCN, we rotate the initial series by SCN times. SCN is just increasing, meaning 0, 1, 2, ... 58 for the 1st, 2nd, 3rd, … 59th Sub-Channel.

3.9 Ranging Transmission

With the Ranging Transmission the terminal communicates various parameters of the RCTT connection with the Base Station. These parameters are sent from the terminal by using Ranging Codes transmitted over specific time intervals (Ranging Intervals) and frequencies (Ranging Sub-Channels). The mapping of the Ranging Intervals inside the frame depends on the type of the Transmission Frame as presented in a previous section. The Ranging Transmission uses only Rectangular Shaping, independently of the shaping type of the Data.

3.9.1 Ranging Codes

The RCTT has to perform Ranging Transmission for the establishment and conservation of the connection. In order to do that, it makes use of Ranging Codes, whose size is 145 bits. A PRBS (Pseudo-random Bit Sequence) generator produces these codes. The first 96 codes are used and they are divided into three categories depending on the purpose they serve:

- The first 32 codes are used to synchronize to the DVB-RCT RF channel (Long Ranging).
- The next 32 codes are used to maintain the connection with the Base Station (Short Ranging).
- The last 32 codes are used to request additional bandwidth (BWrequest Ranging).

Ranging_Code_id is a parameter provided by the MAC layer and indicates which the requested code is and thus it can be deduced whether it is a Long, Short or BWrequest Ranging Code. If *Ranging_Code_id* is given an illegitimate value then Ranging Transmission is performed but it is empty.

3.9.2 Ranging Sub-Channels definition

The Ranging Codes are transmitted over dedicated Sub-Channels, consisted of 145 carriers each (1 carrier/bit). The allocation of the required carriers is based on the carrier calculation algorithm used for MAS3 and the Ranging Sub-Channel Number (RSCN) gives the identity of the Ranging Sub-Channel.

3.9.2.1 2K mode

For the 2K mode the 59 Sub-Channels defined in the section that describes the MAS3 carrier allocation, are grouped to form a total of 12 Ranging Sub-Channels (0 to

11). The first 11 Ranging Sub-Channels consist of five consecutive MAS3 Sub-Channels and the 12th one consists of the last four remaining MAS3 Sub-Channels (11x5 + 1x4 = 59). Consequently every Ranging Sub-Channel makes use of 145(=5x29) carriers and the last Ranging Sub-Channel makes use of 116(=4x29) carriers. The RSCN ranges from 0 to 11, where 0 corresponds to the first Ranging Sub-Channel, 1 to the second etc.

The carriers belonging to a given Ranging Sub-Channel must be sorted in ascending way. The 145 bits of the Ranging Code will be transmitted over the 145 sorted carriers (1 bit/carrier), with the convention that the 1st bit of the output of the Ranging Code Generator Block will be transmitted on the carrier with the smallest value. When we have to use the last Ranging Sub-Channel (116 carriers), only the first 116 bits (instead of the 145 bits) of the Ranging Code shall be modulated.

3.9.2.2 1K mode

For the 1K mode the 29 Sub-Channels defined in section 3.8.3.2 are grouped to form a total of 6 Ranging Sub-Channels (0 to 5). The first 5 Ranging Sub-Channels consist of five consecutive Sub-Channels and the 6th one consists of the last four remaining Sub-Channels (5x5 + 1x4 = 29). Consequently every Ranging Sub-Channel makes use of 145(=5x29) carriers and the last Ranging Sub-Channel makes use of 116(=4x29) carriers.

The transmission of the Ranging Codes over a Ranging Sub-Channel is performed as in the 2K mode. The same stands for the interpretation of the RSCN parameter.

3.9.3 Ranging Interval

A Ranging transmission is performed during one to eight Ranging Intervals. A Ranging Interval has duration of 6 OFDM symbols over which a Ranging Sub-Channel is required. The position of a Ranging Code inside an Interval depends on the kind of the Ranging Transmission.

- **Long Ranging Transmission:** the same Long Ranging Code will be transmitted during the two first OFDM symbols out of the six available. The remaining four shall not be used and will be filled with null symbols.
- **Short/BWrequest Ranging Transmission:** one Short/BWrequest Code will be transmitted during one of the six available OFDM symbols (the rest will remain

unused). The selection of the position is made by the MAC layer and given through the parameter *Time_Symbol_Number*. The remaining five OFDM symbols shall not be used and will be filled with null symbols.

3.9.4 Ranging Frame types

A Ranging Frame may last for more than one contiguous Ranging Intervals. The actual Ranging Transmission is performed during only one Ranging Interval whereas the rest remain empty. The parameter *Time_Symbol_Number* points the position of the Ranging Code inside the Ranging Transmission.

When we make a Long Ranging Transmission, we do not transmit Data. The explanation is that Long Ranging is transmitted in order to synchronize the RCT channel for the first time. Therefore, since the connection is not securely established, it would not be wise to transmit Data risking that way their distortion. Nevertheless, the size of the Transmission Frame will not be reduced and the Data Frame will be transmitted but filled with null symbols.

3.9.4.1 Transmission Frame 1

The Ranging Frame inside Transmission Frame 1 is placed after the transmission of one null OFDM symbol and before the Data Transmission. It can consist of 1, 2, 4 or 8 contiguous Ranging Intervals. The number of Ranging Intervals is expressed in number of OFDM symbols and is specified by the parameter *Ranging_Size*.

The Ranging Code will be transmitted during only one Ranging Interval. The rest will remain unused. The parameter *Time_Symbol_Number* (expressed in number of OFDM symbols) specifies the position of the code inside the Ranging Transmission (0 to 5 corresponds to the 1^{st} interval, 6-11 the 2^{nd},..., 42-47 the 8^{th}). The unused Ranging Intervals will be filled with null symbols. If no Ranging Transmission is performed, the Ranging Frame is reserved space and it will be transmitted either way filled with null symbols. If Long Ranging Transmission is performed, data are not transmitted and the Data Frame is filled with null symbols. *Ranging Sub-Channel Number (RSCN)* indicates over which Ranging Sub-Channel we will transmit the specified Ranging Code.

3.9.4.2 Transmission Frame 2

The Ranging Frame inside Transmission Frame 2, is sent in parallel with the Data Transmission. It consists of 8 contiguous Ranging Intervals and all eight are modulated over the same Ranging Sub-Channel.

In the case of TF2, the parameters of the Ranging Transmission are given per Time Slot (in MAS3 a Ranging Interval corresponds to a Time Slot). Therefore *Time_Symbol_Number* specifies the position of the code inside the Ranging Interval and not the Ranging Transmission (0 to 5). *RSCN* indicates over which Ranging Sub-Channel we will transmit the specified Ranging Code of the Ranging Interval.

3.10 Transmission capacities

From the previous paragraphs of this chapter, it has been made clear that the DVB-RCT transmission is controlled by a variety of parameters. Consequently the transmission capacity also varies according to the configuration of the system.

The physical net bit rate per carrier is affected by the following factors.

- **Transmission mode:** As it has already been described, the transmission mode is defined by a combination of the total system carriers and the inter-carrier distance. The useful symbol duration is a function of the selected mode (see Table 1).
- **Guard interval:** It affects the total symbol duration. It indicates the fraction of the useful symbol by which, it will be increased, in order to calculate the total duration.
- **Modulation type:** It depends on the applied constellation type and coding rate. It defines the useful data payload of a burst.
- **Medium Access Scheme:** It defines the size (number of modulated symbols) of a Transmission Frame, thus, its duration.
- **Shaping type:** It affects the total symbol duration (due to guard interval duration) and also the Transmission Frame duration (insertion of nyquist symbols).

Globally, in typical applications, the DVB-RCT system offers a net bit-rate per carrier ranging from 0,6Kbps to 15Kbps. When all carriers are used, the Base Station is able to collect up to 1Mbbs to 26Mbps of user data in the DVB-RCT channel, whereas the maximum net bit rate for a single user ranges from 86Kbps to 24Mbps. The following chart gives a global view of a user's bit rate for a ¼ guard interval. We can observe the differences between the bit rates for various transmission parameters. Obviously the most robust modes offer the lowest bit-rate, over a large radius cell whereas the weakest modes offer the largest bit-rate over a small radius cell. Furthermore, the selection of the Medium Access Scheme is an important factor. MAS3 is more robust but it requires the allocation of more carriers to a single user. That way few users can enjoy high data throughput, but the Base Station cannot serve many terminals.

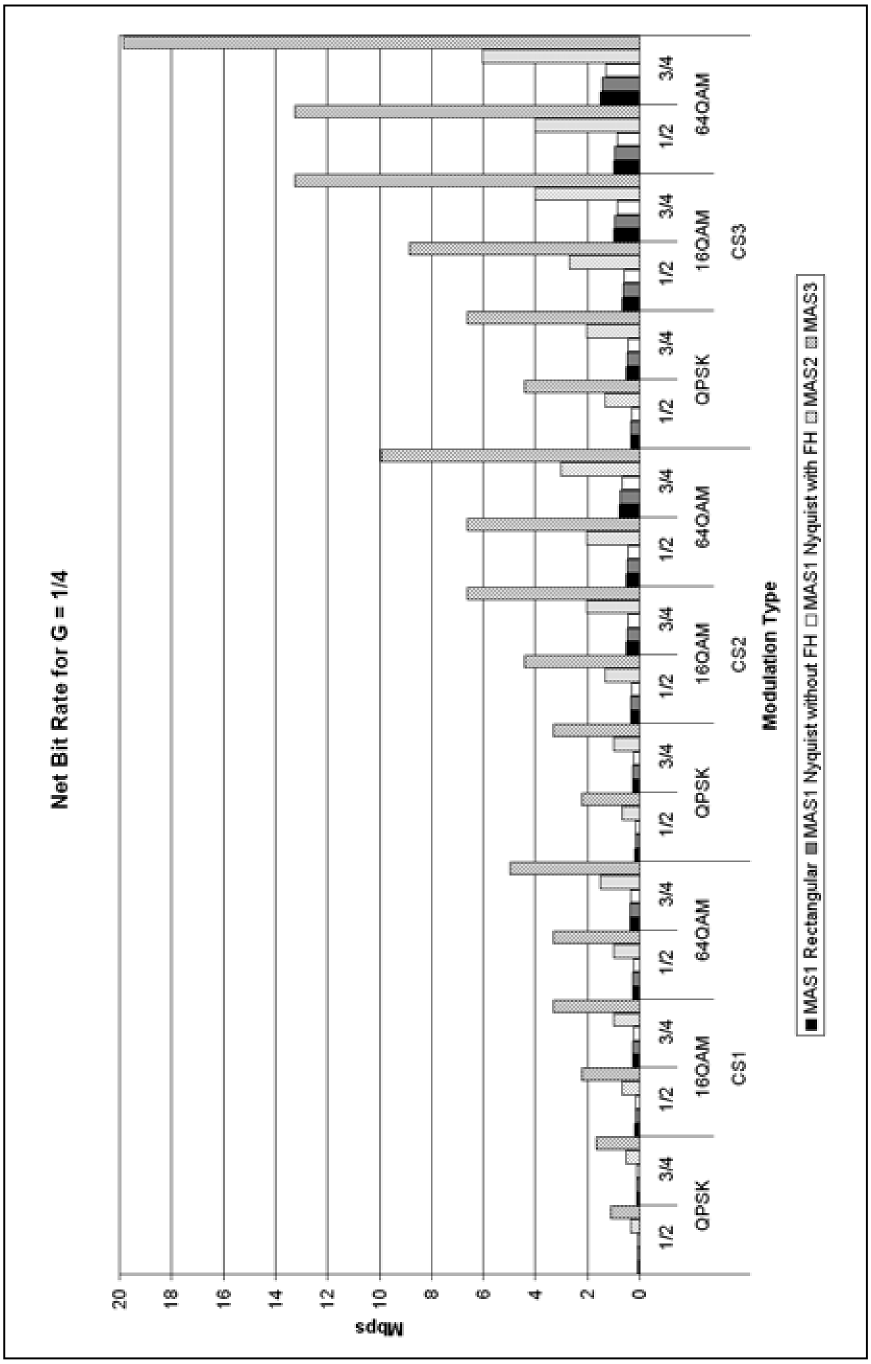

Figure 14 Net Bit rate chart for G = 1/4

4 Physical Layer Organization

In the following picture it is depicted the diagram of the physical layer from the side of the user terminal. As it has already been mentioned, the goal is to achieve upstream data transmission with the simple addition of a set top box to the digital TV set.

The television receives as always the downstream signal and provides to the user the data and information transmitted by the base station. The DVB-T demodulator issues the recovered DVB-T clock (frequency information). The user data as well as part of the received downstream information are forwarded to the MAC management and to the Synchronisation management. MAC (Medium Access Control) management is responsible for gathering all the information related to the transmission resources and configuration assigned to the terminal. It is the decision centre for all the parameters regarding the return transmission. MAC provides the data and the transmission parameters to the main chain of the Physical Layer whose blocks are described below.

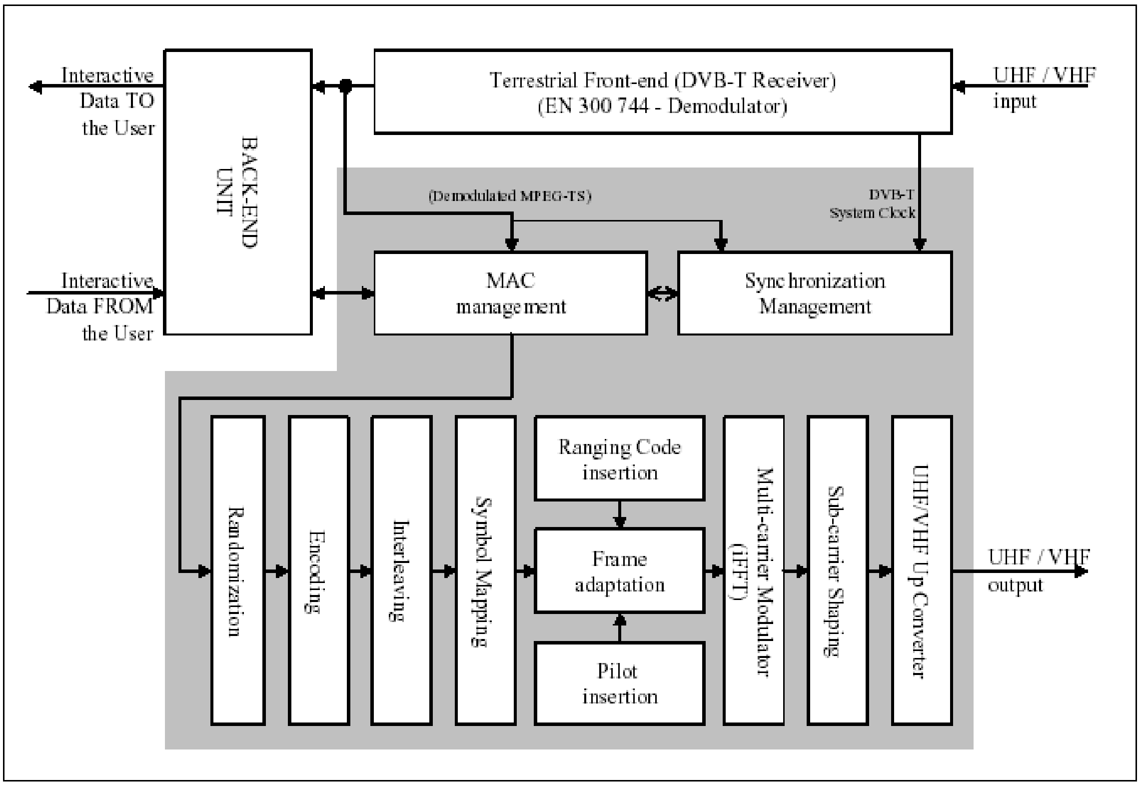

Figure 15: Conceptual Block Diagram for the DVB-RCT

4.1 Randomization

The randomization of the data bits is a basic process for every transmitting system. It redistributes the information in such a way in order to ensure an even distribution of ones and zeros. The even distribution is necessary for optimal performance of demodulation and also for an encoding which will achieve the minimum information loss. It functions as a first encoder, which does not add redundant bits, but rather changes the signal in such a way that allows the easy recovery of the original signal.

The randomization block consists of a pseudo random binary sequence generator. It applies to the incoming bits a randomisation function based on the polynomial $1 + x^{14} + x^{15}$. The generator is implemented with a 15-bit shift register and some XOR gates operating on the bits indicated by the polynomial. The initial binary value of the PRBS generator's shift register is 100101010000000. The following figure depicts the basic block diagram for a PRBS generator.

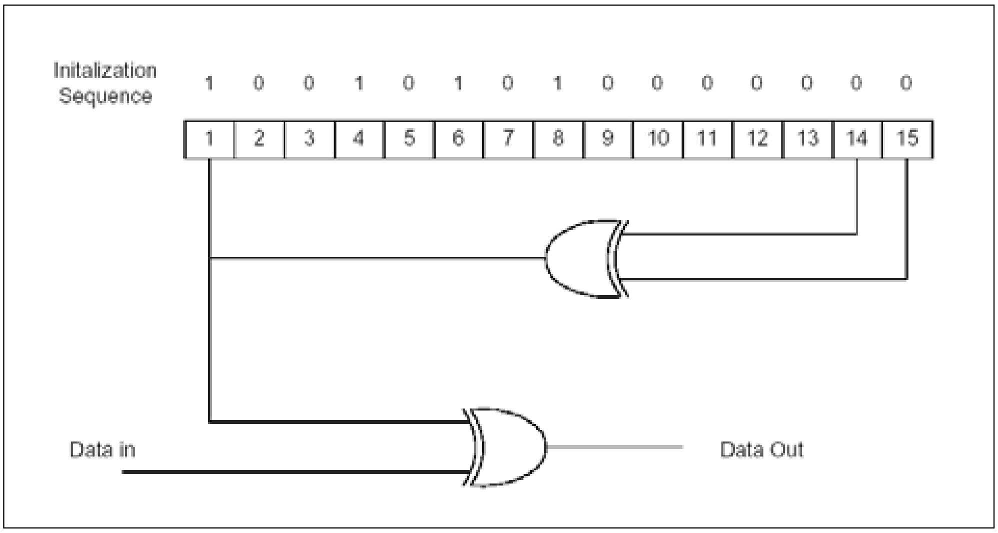

Figure 16 PRBS generator block diagram

Input data from the MAC management are actually bit couples and not just a serial bit stream. For that reason the basic implementation of the PRBS generator is adjusted in a way to process two bits in every clock cycle.

The input data are divided into bursts. Defined quantities of data bits form burst structures, which are sent in bit couples at the input of the Randomizer. The useful data payload of a burst depends on the constellation type that is employed, as well as on the encoding rate to be used in the Encoding stage. Furthermore, the MAC decides how many bursts shall be transmitted inside a Time Slot of a Transmission Frame. It has already been described how Transmission Frames of different types of Medium Access Schemes consist of different number of Time Slots. In addition a Time Slot also has a different burst capacity depending on the Medium Access Scheme. For the above reasons all the parameters, related to the burst size, the Medium Access scheme and the Time Slot capacity are provided to the Randomization block. This is necessary since the PRBS generator must be reset every new burst. Finally, the block has an additional functionality apart from the randomization. This is the task to generate the appropriate flags, which indicate the start and the end of a burst, a Time Slot and a Transmission Frame. The flags, burst limits, will accompany the data through all the stages up to the Frame Adaptation block input. The rest of the flags will continue to be issued until the final digital stage of the physical layer.

4.2 Encoder

After the randomization of the data, the ones and zeros of the information are evenly distributed. That way the data are prepared for the next stage, the encoding. The air channel inserts a big error rate to the transmitted information. Thus the encoding of the data is indispensable because it permits the detection and correction of errors that may occur during the transmission. The target is to perform an encoding which will permit the correction of a limited yet important number of errors, while keeping the level of added parity bits at a low level.

In our system two different encoding algorithms are supported: Turbo Coding or Concatenated Coding, which consists of Reed Solomon followed by convolutional encoding. From the available encoding algorithms only one is being used and it is selected during the system initialization. The option to choose between two encoding algorithms has important advantages. In the past Reed-Solomon encoding was used followed by a convolutional encoding. In the last years it was proven that iterative encoding (as Turbo Code) could result to an improvement of the bit error rate for a given C/N level. This improvement can be used to achieve a bigger cell radius for a given transmitting power level. When approached from a different point of view, for a given network topology the transmitting power can be reduced, leading to less interference and pollution. Apart from the two coding types two different encoding rates can be used, alternatively. More precisely, we can have combinations from three different constellation types and two coding rates. This is done in order to be able to choose between a lower bit-rate and a more robust encoding scheme. A puncturing process is applied to the generated parity bits depending on the encoding rate. That way a reasonable bit rate is achieved by reducing the number of parity bits, which is large. At this point a varying latency is added, due to the computation and addition of parity bits.

4.3 Interleaver

After the encoding, the issued data must be interleaved. This is essential in order to avoid sequential errors in the same code word. Because of the air transmission the signal can be altered in such away that errors occur in sequential positions. The decoding process can support the correction of a message up to a number of errors. By interleaving the bits of different words, the possibility to have sequential transmission errors in the same word is reduced. Consequently, the danger to exceed the error correcting capability of the decoding process is avoided. This way even if sequential errors occur, during the reverse process at the receiver end, they will be distributed over multiple code words.

The interleaver's task is to inter-mix the input bit quantities generated through encoding. This process adds latency. It is inevitable, since the entire encoded burst structure must be collected. When all the data are available, the interleaving can start. The Burst Structure is issued as 144 data symbols. Each symbol is 6 bits long. It contains 6 bits of encoded and interleaved data or less bits padded with zeros, depending on the applied constellation type. Specifically, in the case of 64QAM, 6 out of 6 bits are encoded data. In the case of 16QAM, 4 out of 6 bits are encoded data. Finally, when using QPSK, only 2 bits of the symbol are data. The length of the encoded data payload according to the constellation type is listed in the following table.

	QPSK	16QAM	64QAM
Bits/data symbol	2	4	6
Encoded Data payload of a burst (bits)	288	576	864

Table 11 Useful encoded data payload of a burst

The interleaving process can be performed with a Pseudo Random Sequence generator, which uses the polynomial $1 + x^3 + x^{10}$ and is initialized with the value 0001011010. Each time a bit arrives the PRBS generator produces a value, which corresponds to the position of the incoming bit, inside the output burst. Consequently, the generated values must belong in the range 1 to n, where n is the length of the burst (depending on the constellation type as described in Table 2). Therefore, if the generated value is greater than n, then a new value is generated. This step is repeated until a valid index is produced. We can observe that the generated values are fixed for a given burst size. In order to save time and speed up the process, the PRBS generator is bypassed and the valid index values are stored inside a ROM. Therefore, the Interleaver first selects the valid bits that are indicated by the mask that accompanies the incoming data. Then it stores them in their natural order, meaning in the order they are received. Afterwards, it reads the bits from the memory locations that are provided by the ROM area, which corresponds to the current constellation type. Finally, it groups the required number of bits in order to form a data symbol and issues the symbol along with a valid flag.

4.4 Frame Adaptation

The frame adaptation block receives the data to be transmitted, in the form of 6-bit data symbols. Its task is to collect the symbols of the bursts that will be transmitted in parallel over the same Time Slot of a Transmission Frame and to organise them in frequency and time. Then it issues the symbols of the entire Time Slot in frequency (time slices) having added the necessary null symbols and extra control symbols.

The frame adaptation is the process where we are going to focus our interest. The block's basic function is to reorder the incoming user symbols as dictated by the system configuration. It is the cornerstone of the system and it will be presented in detail in a separate chapter from both the functional and the implementation point of view.

As already mentioned, apart from the data symbols, the block must also issue some control symbols. There are two kinds of control symbols: Ranging symbols and pilots. The first are used for issues related to the connection with the Base Station, whereas the later are necessary to allow coherent detection. The generation of the Ranging and Pilot Codes is integrated in the Frame Adaptation block. Their functionality is simple and is implemented with simple look-up tables.

4.4.1 Ranging Pilot generation

The Ranging codes are used to transmit information related to the connection's establishment and conservation. They are requested and placed in the Transmission Frame by the Frame Adaptation stage. Ranging transmission has already been explained in a previous chapter. The total number of Ranging codes is 96 and they are divided into three groups depending on their purpose of use.

The code generation is based on a Pseudo Random Binary Sequence Generator using the polynomial $x^{15} + x^7 + x^4 + x + 1$ and being initialized with the binary value 000000010101001. Instead of implementing the generator, the codes are stored inside a 96 x 145 bits ROM, addressed with the code identity number. The block will provide bit by bit, the required code after request of the Frame Adaptation control block.

4.4.2 Pilot generation

The pilots are distributed among the data of a burst in specific positions and they serve to the coherent detection of the transmitted data by the Base Station. The generated pilot symbols are 1-bit quantities, and their value depends on the carrier over which they are transmitted.

In theory, their values are produced by a PRBS generator based on the polynomial $x^{11} + x^2 + 1$ and initialized with the value 11111111111. Once again in order to simplify the design, a ROM will be used instead of a PRBS generator, which will provide the pilot corresponding to a specific carrier, upon request of the Frame Adaptation control block.

4.5 Symbol Mapper

The Symbol Mapper receives the 7-bit symbols issued by the Frame Adaptation block and maps them to a complex number (a real and imaginary part, 10 bits each). The issued symbols can be data, pilots and boosted pilots, ranging pilots or null symbols. In order for the Symbol Mapper to recognize the symbol type, a pre-mapping has taken place in the Frame Adaptation stage.

The mapping of the data symbols with respect to the constellation type is depicted in the following figure, which illustrates the position of each data symbol in the axes system of complex numbers.

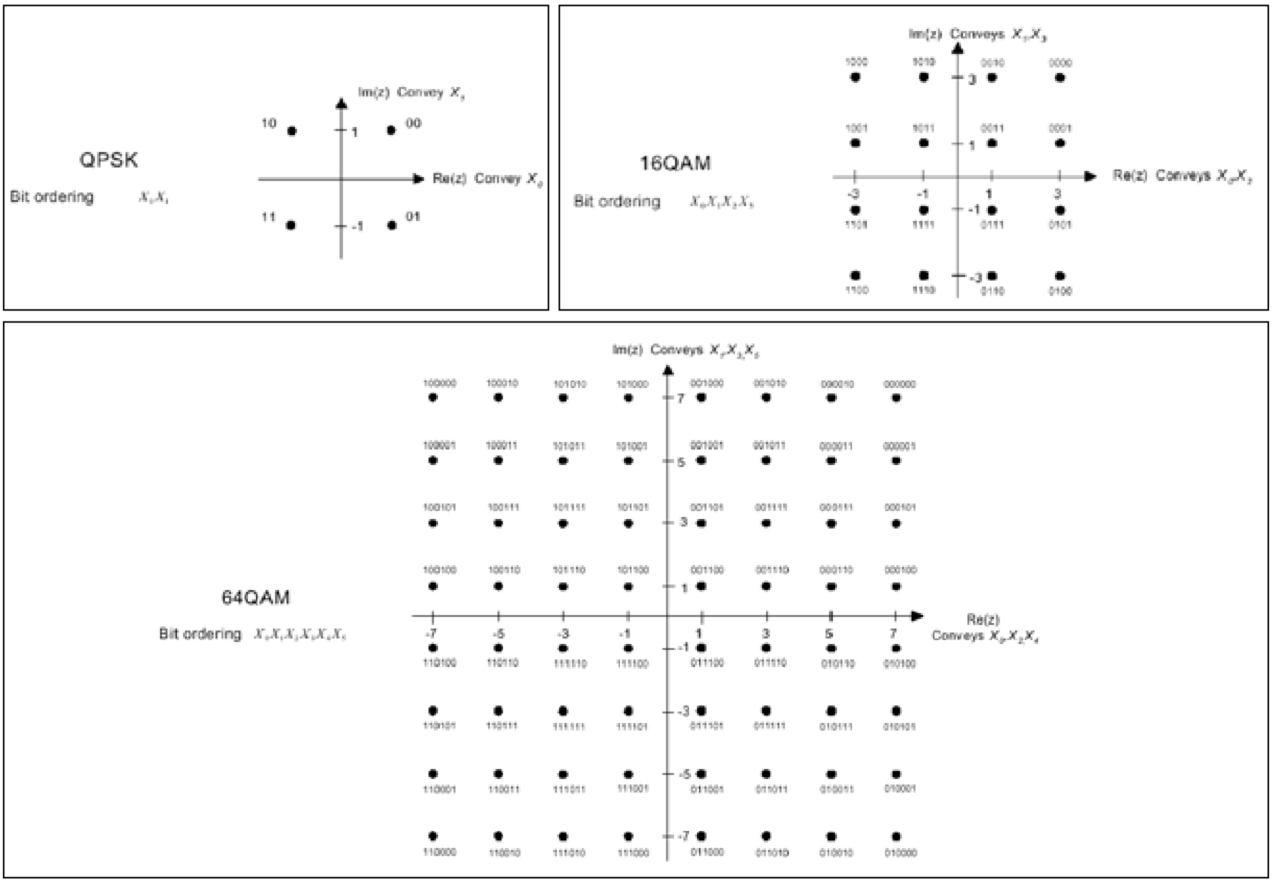

Figure 17 QPSK, 16QAM and 64QAM constellations

After the mapping of the data symbols, the real and imaginary parts must be divided with a normalization factor, which depends on the constellation type as listed in the following table.

Constellation Type	Normalization factor
QPSK	$\sqrt{2}$
16QAM	$\sqrt{10}$
64QAM	$\sqrt{42}$

Table 12 Normalization factor

As far as it concerns the mapping of the other symbols, null symbol corresponds to complex zero (both real and imaginary parts are zero) and the other pilot symbols are translated to a complex number as appears in the next table.

Pilot	Re formula	value 0		value 1	
		real	imaginary	real	imaginary
Boosted	4/3 x 2 (1/2 - pilot bit)	4/3	0	-4/3	0
Non-boosted	2 (1/2 - pilot bit)	1	0	-1	0
Ranging	2 (1/2 - ranging bit)	1	0	-1	0

Table 13 Pilot Mapping

The Symbol Mapper's implementation is straightforward. The complex values are stored inside a ROM. The pre-mapping of the incoming symbols is done in such a way that their value corresponds to the address, which contains the complex value of the symbol. The block receives the symbols and issues them in their complex form after a clock cycle.

4.6 IFFT, Shaper, Up Converter

These three blocks perform the final stages of the Physical Layer. IFFT is a component that performs an Inverse Fast Fourier Transformation to the incoming data. In the OFDM chapter we have already analyzed the connection between IFFT and the OFDM generation. Many companies have designed and provided hardware modules that implement an IFFT process and one of these has been used. The Shaper block applies the appropriate filters to the incoming signals before forwarding the signal to the final stage. Two shaping types are supported: Rectangular and Nyquist. For each Transmission Frame only one is selected and applied. Finally, the Up-converter block is the last stage of the chain, where the data are converted from their binary form to an analogue signal, appropriate for broadcasting.

5 Frame Adaptation Unit Analysis

The Frame Adaptation block (FRM) is the "cornerstone" of the Physical Layer. Here, all data, pilot and ranging pilot symbols are gathered. The task of the FRM is to reorder the data and place all the symbols in the frequency-time grid that consists the Transmission Frame. Then it issues the entire Transmission Frame, symbol by symbol, in columns. The data arrive serially from the previous block (Interleaver) in the form of data symbols (144 data symbols define a Burst Structure). The pilot and ranging pilot symbols are requested from the blocks responsible for their generation, which is based on the use of ROMs.

After the careful study of the Physical layer's stages, the system parameters and the main definitions, introduced in previous chapters we can start the analysis of the Frame Adaptation block's functionality. This analysis will lead us to select the structure and design of the block and each decision will be defended with a complete argumentation.

The first step is to examine the input and output rates. Then we will approach the FRM functionality by studying the general structure of the Transmission Frame contents and calculating the data occupancy of the frame. Then, since a reordering task is performed, a storage scheme will be required. That scheme will have to be effective for all the different configurations of the system and its choice will be based on the occupancy results. An Input Controller (ICTR) will be responsible for the storage of the Transmission Frame and an Output Controller (OCTR) for the reading of the symbols from the memory and their issuing. Finally the Transmission Frames must be issued consecutively and that implies the need for a double buffer mechanism. Such a mechanism will permit OCTR to issue a frame while ICTR will prepare and store the next one.

5.1 I/O Rates and Operation frequency

The data symbols of a burst arrive from the Interleaver block with an input rate of one symbol every 2, 4 or 6 clock cycles depending on the applied constellation type. Each symbol is received along with a valid signal. In addition, three more pairs of flags accompany the incoming symbols and they indicate the start and end of a burst, Time

Slot and Transmission Frame (sob and eob, sos and eos, sof and eof). Their timing appears in Figure 18.

The bursts are not continuous and the time between the end of a burst and the start of the next one varies. It depends on the latency of the previous stages as results for a given system configuration and also it is determined by the rate that the data stream is fed to the Physical Layer's chain. In any case, the latencies are taken into consideration and the initial input rate must be regulated by the Medium Access Control, in order to ensure that the Frame Adaptation block will receive the data, transmitted over of a Time Slot, in a specific time margin. This period is equal to the time required by the block to issue the contents of the entire Time Slot. Obviously it is not constant but depends on numerous parameters as the Transmission mode, the FFT length, the Guard Interval length, the Ranging transmission duration and the applied Medium Access Scheme. For the moment lets keep a note that after measurements of the previous stages latencies the worst case has been estimated to be one burst every 2000 clock cycles when the constellation type is 64QAM and the code rate 3/4.

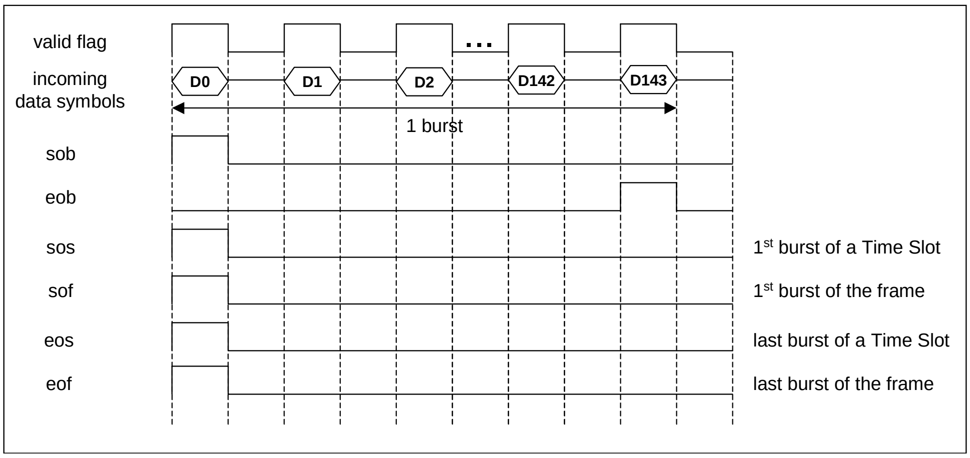

Figure 18 Timing of the incoming data flags and interpretation

The clock of the Frame Adaptation Unit is dictated by the RCT system frequency. In a previous chapter, it has been described how the RCT system frequency depends on the applied Carrier Spacing. The reference clock calculated for an 8MHZ DVB-T system (used in Europe) is 64/7 MHz. Following the analysis of Table 1 the RCT system clock may be 64/7 MHz, 64/14 MHz, 64/28 MHz or 64/56 MHz depending on the inter-carrier distance. The FRM operation frequency must have such a value that it will permit the block to adjust its output rate according to the set inter-carrier distance. A conventional solution is to select the clock to be four times the fastest RCT system clock, meaning approximately 36 MHz (4 x 64 / 7 MHz). That gives FRM, four cycles to process the output of a symbol. Then the symbol remains latched for 4, 8, 16 or 32 cycles, according to the transmission mode. That design choice gives FRM the flexibility to support different bit rates, by adjusting its output rate to the configuration of the transmission mode (FFT length and carrier spacing).

It has been mentioned in the beginning of this paragraph that the condition for the correct operation of the unit is the data to arrive in a specific margin of time. Through an example we will prove that this condition is not always met and thus measures should be taken. From calculations of the previous stages' latencies the worst case is approximately one burst every 2000 cycles when the constellation type is 64QAM and the code rate 3/4. The Time Slots of Medium Access Scheme 3 are the smallest ones, because they transmit only six time slices. The capacity of the Time Slot is 54 burst structures. Therefore, the maximum quantity of contents is received in 54 x 2000 = 108000 clock cycles. The fastest transmission mode is 2K and CS3, where the output is latched for four output clock cycles and the smallest guard interval is 1/32 of the time slice. That gives (2048 + 64) points x 6 time slices x 4 cc = 50688 clock cycles. Unfortunately we discover that the Time Slot's data need time to be received that exceeds the permitted margin. The ratio of input clock cycles to output clock cycles equals 108000 / 50688 = 2.13. The above remarks lead us to the conclusion that the stages preceding the FRM must be accelerated. Since the FRM clock is 36 MHz, the operation frequency of the previous stages must be at least 2.13 x 36 = 77 MHz. A 100 MHz frequency has been selected, which offers an additional safety margin, while the FRM clock remains at 36 MHz.

5.2 Transmission Frame Structure and Occupancy

Every Transmission Frame consists of a Ranging and a Data Transmission. Two types of Transmission Frame have been introduced. For the first type the two kinds of transmission are consecutive in time and thus they occupy distinguished parts of the frequency-time grid. On the other hand, in the case of the second type, the two transmissions cannot be distinguished since they are performed in parallel and over variable sub-channels (there are not fixed frequencies dedicated to each transmission). In addition there are areas in the Transmission Frame grid that always contain null symbols (i.e. the unused carriers that form guard bands). Following the above remarks, we conclude that the Transmission Frame can be viewed as a table divided to smaller matrixes that correspond to the following three areas:

- Ranging area: It contains ranging pilot symbols and null symbols.
- Data area: It contains data symbols, pilot symbols and null symbols. For the second type of Transmission Frame, the Data area is considered to include the Ranging Transmission as well, thus it also contains ranging pilot symbols.
- Null area: It contains only null symbols.

We define as Time Slot the part of the Transmission Frame, which contains all the burst structures that are supposed to be transmitted in parallel. A Transmission Frame is divided into one or more Time Slots depending on the employed Medium Access Scheme. A Time Slot may include, in addition to the data, the Ranging transmission, according to its order inside the Transmission Frame. FRM issues the frequency-time grid in columns and each column is equivalent to an OFDM symbol. Therefore, it can start issuing, when all the information that must be transmitted in a Time Slot is available.

The data reordering can be visualized as the placement of the data symbols in the Time Slot matrix. Since the functionality of FRM has to do with reordering some intermediate storage of the matrix or data is necessary in a memory scheme whose size and organization will be defined.

Figure 19 illustrates in rough lines, the flow of the data reordering in a Time Slot for Medium Access Scheme 1. It is not the general case but it is easily depicted and demonstrates in a clear way the main idea of the data redistribution and why all the data of a Time Slot must be received before the matrix is output. When a column’s output is completed, the next one starts to be issued after a pause corresponding to the required Guard Interval time. The output timing is depicted in Figure 20 where we can see the Guard Interval spacing among the issued columns.

To start the FRM analysis we present the structure of the Transmission Frame table for the main configuration cases. It will be explained why only the Data area concerns us and after each presentation, we will focus our interest on that part of the Time Slot matrix.

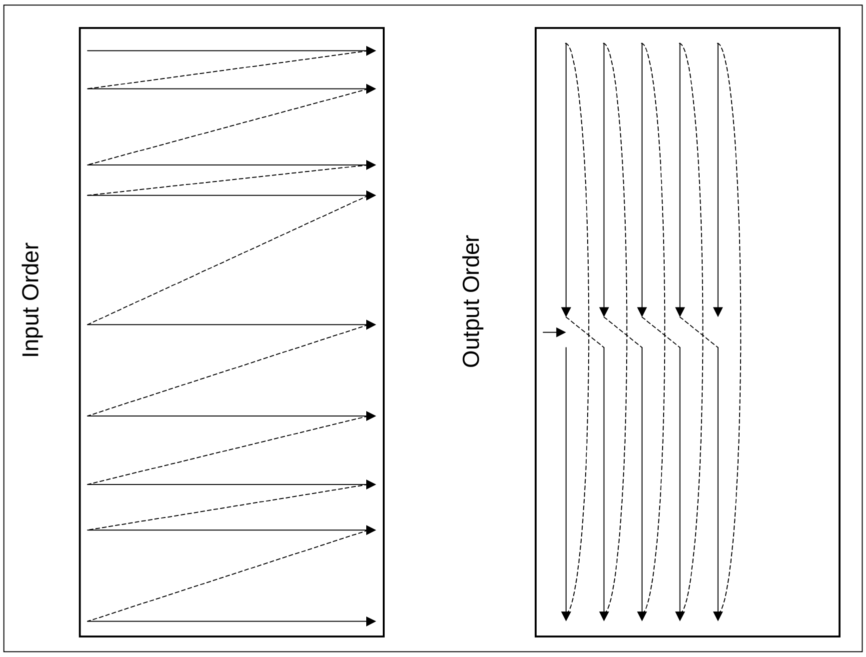

Figure 19 Example of reordering task

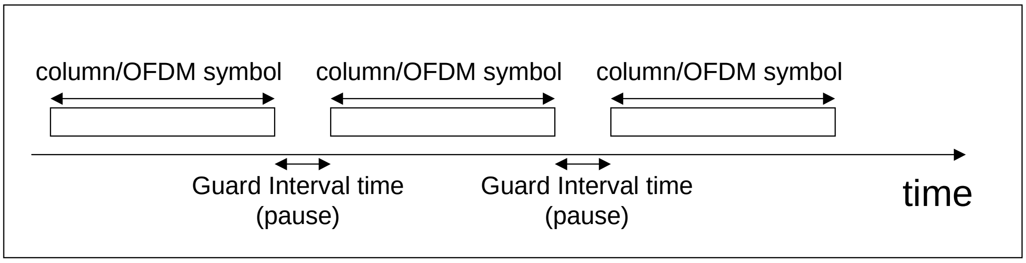

Figure 20 Output OFDM symbols timing

5.2.1 Medium Access Scheme 1

In Medium Access Scheme 1, the concepts of Transmission Frame and Time Slot coincide. The constructed and transmitted matrix consists of the three known area types. The vertical dimensions can vary depending on whether the system is configured to work with 2K or 1K carriers. The horizontal dimensions depend on the selected shaping type (Rectangular or Nyquist) and the frequency hopping option. Figure 21 to Figure 23 depict the different organization that the transmitted matrix can have in Medium Access Scheme 1.

Lets examine the areas of the Transmission Frame a bit more carefully. The Null area consists of the carrier guard bands and the first column of the table (as dictated by the format of the first type of Transmission Frame). In case of Nyquist shaping we observe that vertical zones of null symbols are inserted. The Ranging area's width is 6 to 48 columns depending on the parameters of the Ranging transmission, whereas its height occupies the range of useful carriers. The same stands for the height of the Data area. The width of the Data area is always 180 columns which correspond to the 144 data symbols of a burst structure (placed in one line of the matrix) plus 36 pilot symbols scattered in specific positions in the line occupied by data. It must be noted that the 180 columns width stands even for the case of Nyquist Shaping with frequency hopping. This can be justified if we imagine the Data area to be the same matrix as in the other two cases, with the interleaving of vertical null zones every quarter of the area's width.

From the three areas only the Data area attracts our interest, since it is the only one that a part of its contents (data symbols) are not available (any kind of pilot symbol is available upon request). When all the data of a Time Slot are received, they will be placed inside the Data area matrix. The question that arises is whether it is worth storing the entire matrix. An alternative solution is to store only the data and then reorder them on the fly. In order to investigate these two alternative solutions we will examine the placement scheme of the data in the Data area matrix and the final occupancy.

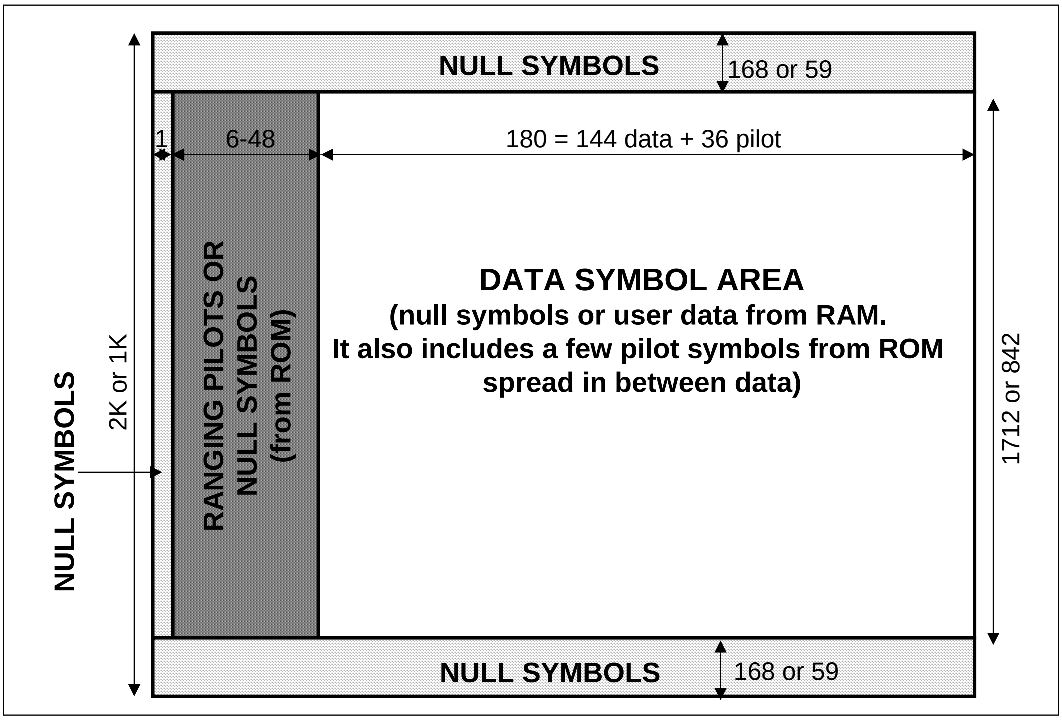

Figure 21 Transmitted matrix for Medium Access Scheme 1 and Rectangular Shaping

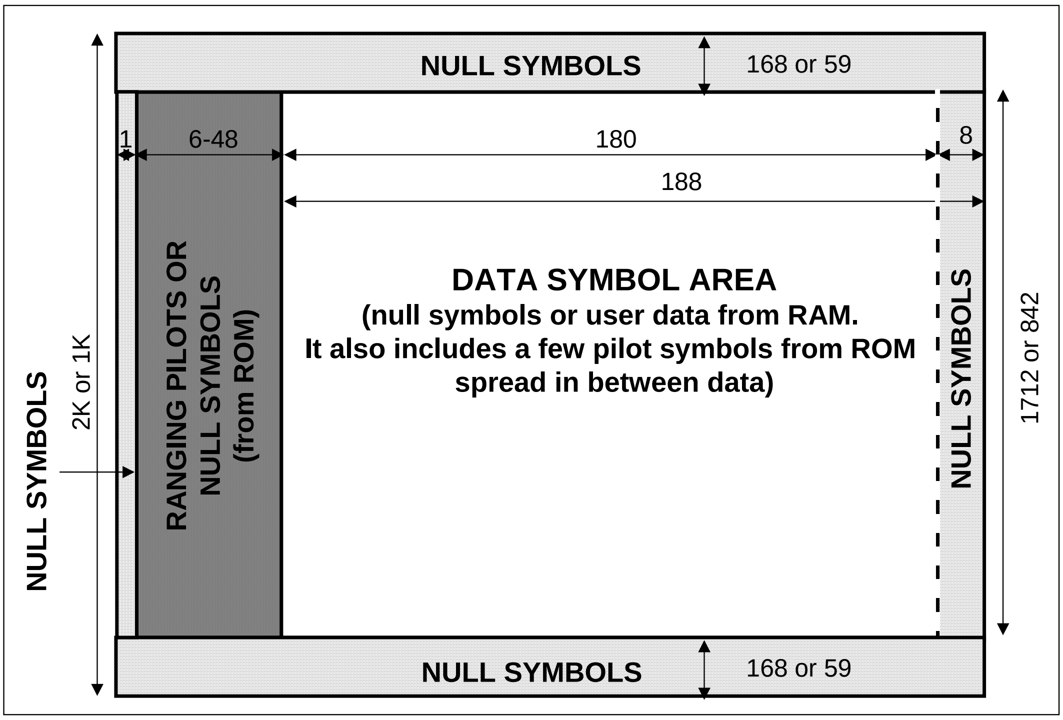

Figure 22 Transmitted matrix for Medium Access Scheme 1 with Nyquist Shaping and no frequency hopping

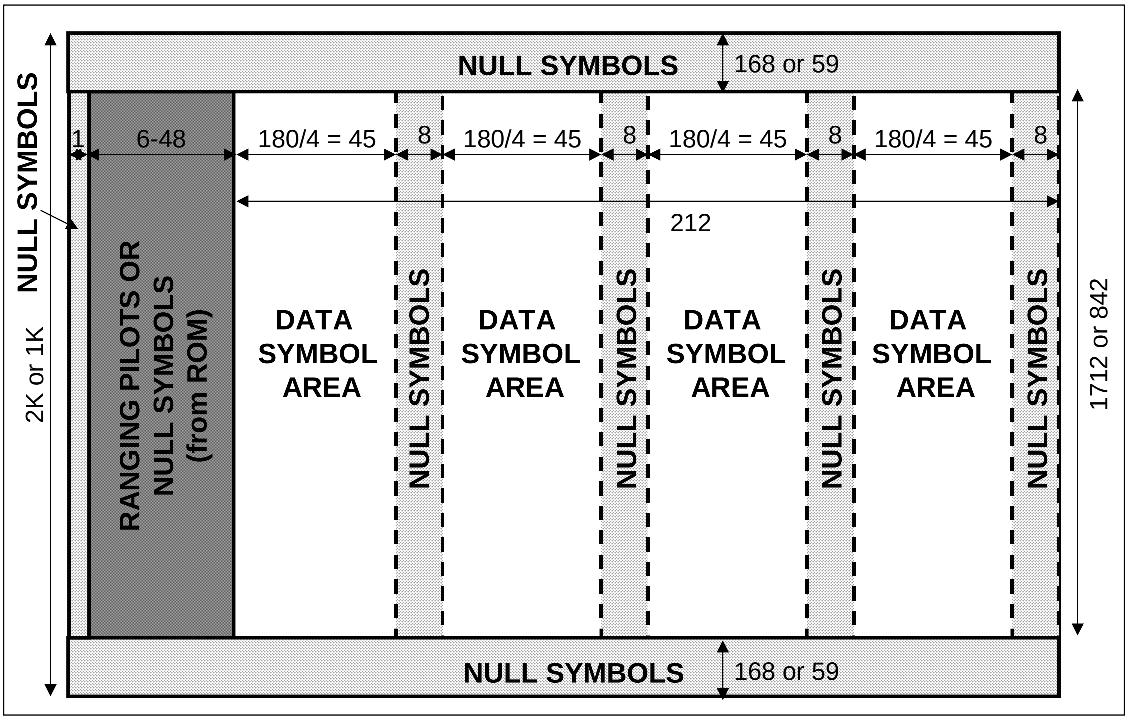

Figure 23 Transmitted matrix for Medium Access Scheme 1 with Nyquist Shaping and frequency hopping

In Medium Access Scheme 1, the data are organized in Burst Structures of the first type (BS1). Though a burst structure consists of 144 data symbols, BS1 contains 180 symbols because a collection of pilot symbols is scattered among the data. An element of the matrix can be data or null, pilot or null but never data or pilot. The pilot insertion pattern has already been presented in a previous paragraph and it is a function of the carrier (line) number modulo 4, exclusively.

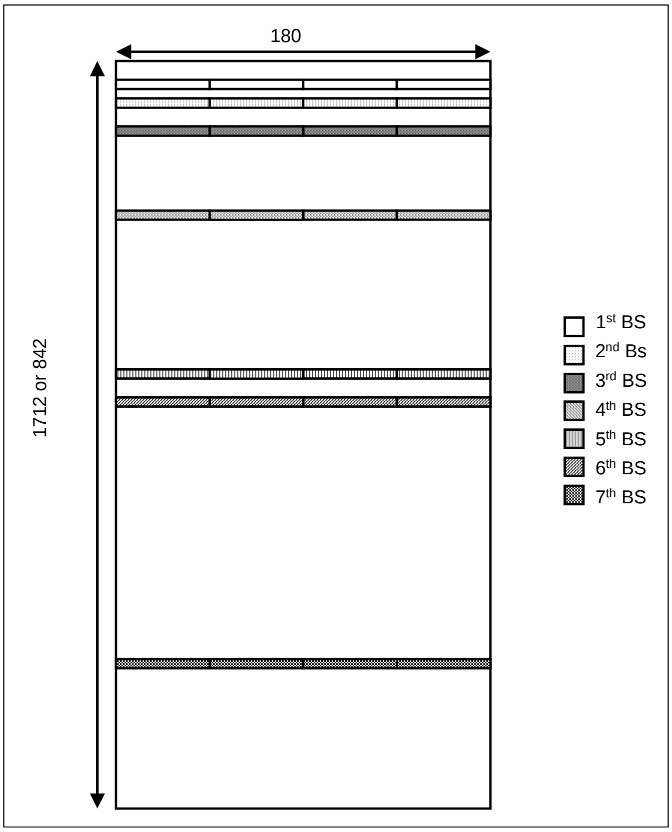

Figure 24 Placement of Burst Structures 1 in Data area matrix without frequency hopping

Figure 24 illustrates a Data area matrix filled with seven burst structures to be transmitted without frequency hopping. When no frequency hopping is applied a sub-channel consists of one carrier only. Each burst structure occupies a line of the matrix, which corresponds to the used sub-channel. The line is filled from left to right with incoming data. The bursts' order of arrival is also reflected in the order of the used sub-channels numbers: the sub-channel occupied by a burst structure has a smaller number than the ones of the bursts that arrive later.

Figure 25 illustrates a Data area matrix filled with four burst structures to be transmitted with frequency hopping. In the frequency hopping case, a burst structure is divided into four mini-bursts. In addition, a sub-channel uses four carriers. Each mini-burst occupies one quarter of four matrix lines, which correspond to the four carriers of the used sub-channel. The result is the same as before: a burst structure occupies in total one line of the matrix. Once again the order of the used sub-channels number and the order of the incoming bursts coincide. An important detail is that the sub-channel numbers cannot take a random value. The entire usable range is divided to four zones of consecutive sub-channel numbers. The sub-channels reserved for the data transmission of a Time Slot, can belong to only one of the four zones. That way a line may contain the data of only one mini-burst and the remaining three quarters will be empty. In the depicted example the sub-channels start from the first quarter whereas in Figure 26 we can see two alternative forms of the matrix if the sub-channels belonged to the second or third zone. More details about the sub-channel calculation have already been presented in the Carrier Allocation paragraph of the previous chapter.

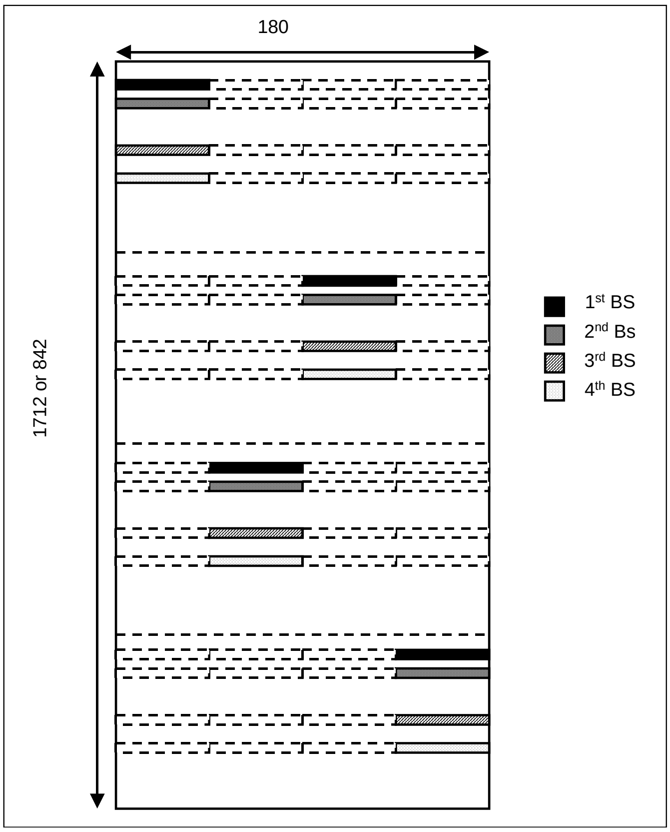

Figure 25 Placement of Burst Structures 1 in Data area matrix with frequency hopping

After seeing how bursts are placed inside the matrix we will estimate what percentage of the Data area is occupied by the burst's symbols. Lets examine the biggest case since it will be the one dictating our final choice of memory requirements. The total size of the matrix in symbols is 1712 x 180 = 308160, thus approximately 300K symbols. The maximum number of burst structures transmitted during a Time Slot, in Medium Access Scheme 1 is 128. That results to a total of 128 x (144 data + 36 pilot) = 18432 data + 4608 pilot = 23040 = 22.5K symbols. This number is only the 7.5% of the total matrix positions whereas the actual data occupy even less space (6%).

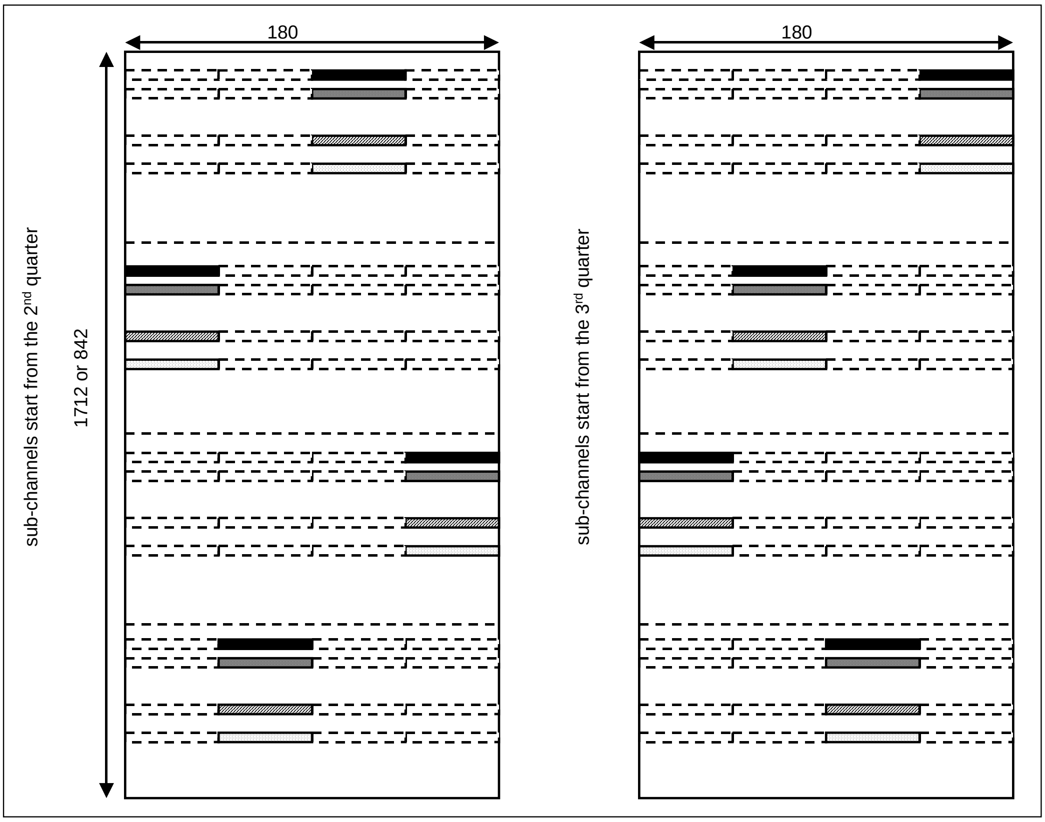

Figure 26 Placement of Burst Structures 1 in Data area matrix with frequency hopping for two alternative sub-channel groups

Already from these first occupancy calculations, it appears that it is not in our interest to attempt to store the entire Data area matrix because it is sparse. Of course that decision remains to be taken after we examine the other schemes as well.

5.2.2 Medium Access Scheme 2

In Medium Access Scheme 2, a Transmission Frame consists of four Time Slots. The same area types exist in this matrix and the vertical dimensions vary the same way as before (2K or 1K mode). On the other hand the horizontal dimensions are always the same for a given Ranging transmission configuration. Figure 27 depicts the organization that the transmitted matrix can have in Medium Access Scheme 2.

We will examine in more details the areas of the Transmission. The Null and Ranging areas are the same as before. So is the height of the Data area. The width of the Data area is 45 columns per Time Slot, which corresponds to a mini-burst's size (36 data and 9 pilot symbols following the same pattern as in Medium Access Scheme 1). That way the Transmission Frame's size is the same for both Medium Access Schemes 1 and 2. This is logical since both cases involve the same type of Transmission Frame: the first one.

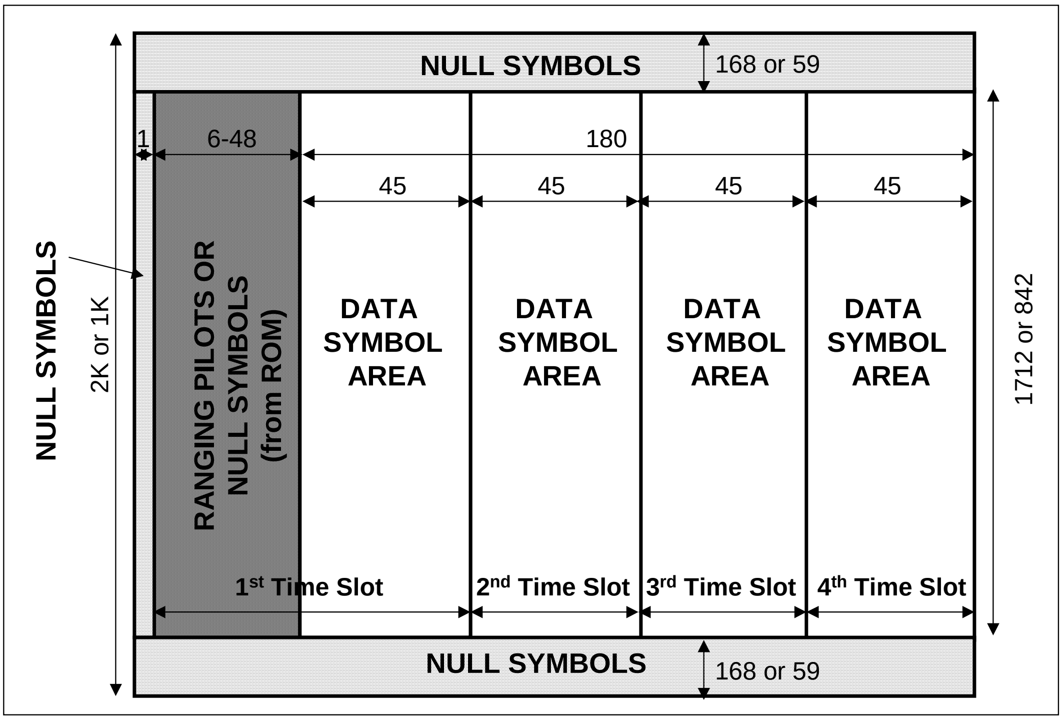

Figure 27 Transmitted matrix for Medium Access Scheme 2

Once again we will concentrate our interest on a Time Slot's Data area. Now its size is the one quarter of the matrix presented in the previous paragraph. We will examine the placement scheme of the data in the Data area matrix and the final occupancy.

In Medium Access Scheme 2, the data are organized in Burst Structures of the second type (BS2). The placement of BS2 has many similarities with the placement of BS1 with frequency hopping. It consists of 144 data symbol and 36 pilots (scattered with the same pattern) and is divided in four mini-bursts. The sub-channel for BS2 also consists of four carriers. Each line of the matrix corresponds to a carrier and is occupied by one mini-burst. An example is illustrated in Figure 28 where four burst structures are placed inside a time Slot's Data area matrix. The order of the used sub-channels number and the order of the incoming bursts coincide. For more details about the sub-channel calculation the reader can refer to the Carrier Allocation paragraph of the previous chapter. As a conclusion we can state that the case of MAS2 is like MAS1 with frequency hopping with the difference that the matrix is shrunk and the mini-bursts are thus placed (transmitted) in parallel.

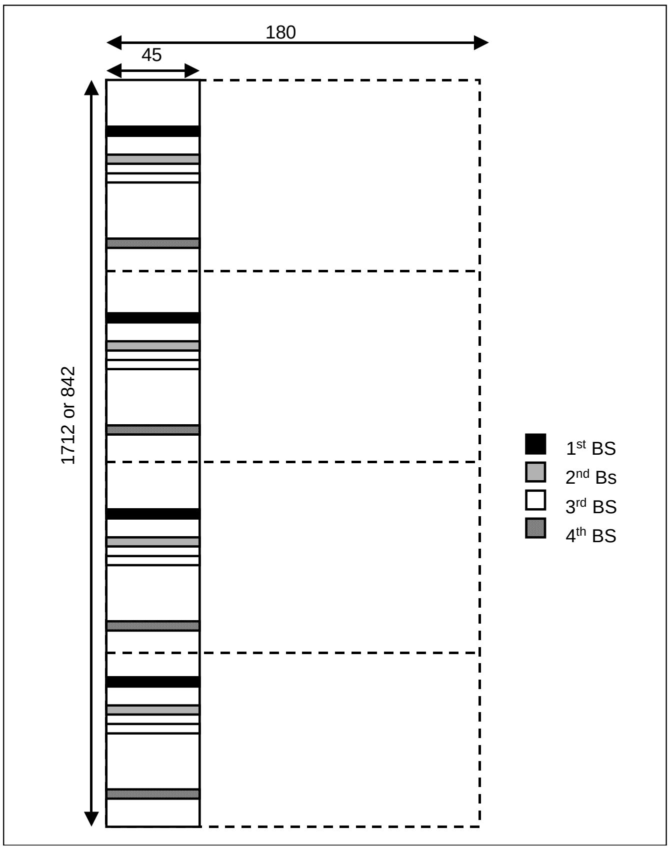

Figure 28 Placement of Burst Structures 2 in Data area matrix

The procedure to calculate the matrix's data occupancy is the following one. The total maximum size of the matrix in symbols is 1712 x 45 = 77040, thus approximately

75K symbols. The maximum number of burst structures transmitted during a Time Slot, in Medium Access Scheme 2 is 128. That results to a total of 128 x (144 data + 36 pilot) = 18432 data + 4608 pilot = 23040 = 22.5K symbols. This number is the 30% of the total matrix positions and the actual data occupy 24% of the matrix space. Still the matrix is sparse. Since the MAS1 case requires the use of a larger matrix, a more useful calculation is the occupancy of BS2s in relation to that matrix. We can picture that the data are placed in a 1712x180 matrix where not all columns are occupied but instead the input to output reordering is limited to the first 45 columns. The result is that only the 7% is filled. These new calculations reinforce the selection not to store the entire matrix, because it would be translated to the waste of memory space.

5.2.3 Medium Access Scheme 3

In Medium Access Scheme 3, a Transmission Frame consists of eight Time Slots. Its type is the second one, which means that the Ranging area is merged inside the Data area. The vertical dimensions vary the same way as before for 2K and 1K mode. The horizontal dimensions are constant and equal to 8x6 = 48 columns. Figure 29 depicts the organization of the transmitted matrix.

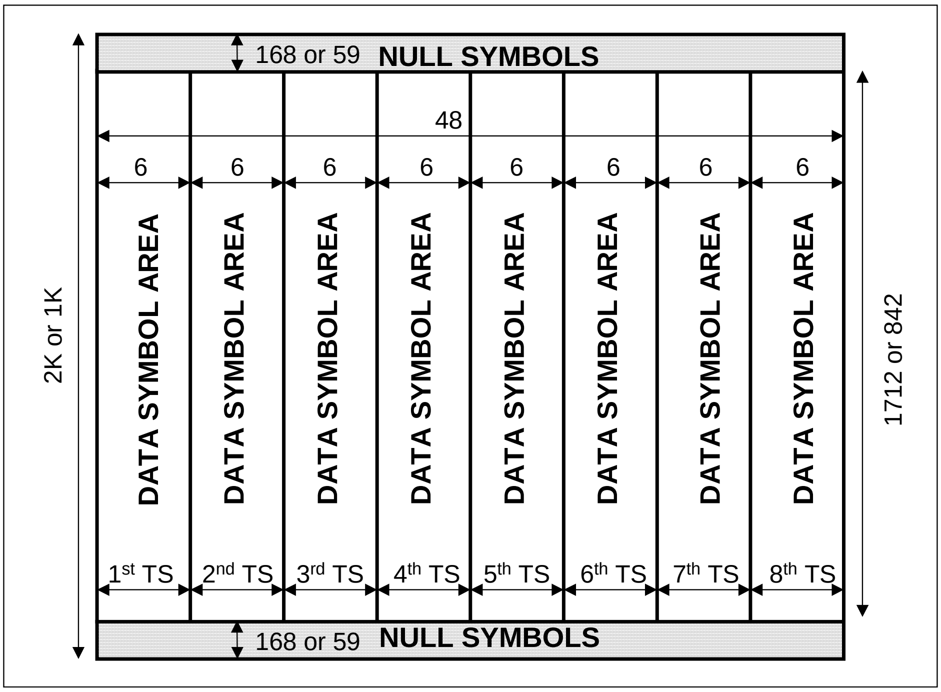

Figure 29 Transmitted matrix for Medium Access Scheme 3

We will examine in more details the areas of the Transmission. The Null area is the same as before. The height of the Data area always corresponds to the usable carriers, whereas its width is 6 columns per Time Slot, which results from the duration of the third Burst Structure type (BS3). Viewed from a different angle the Data area's width is equivalent to the duration of a Ranging Interval, which is placed in parallel to the data.

As before we will concentrate our interest on the Data area of a single Time Slot. We will examine the placement scheme of the data in the Data area matrix and the final occupancy.

In Medium Access Scheme 3, the data are organized in Burst Structures of the third type (BS3). The placement of BS3 is totally different from the placement of the other burst structure types. It consists of 144 data symbol and 30 pilots, scattered this time inside the burst structure with a fixed pattern. The BS3 sub-channel consists of 29 carriers and each line of the matrix corresponds to a used carrier. The data are placed in frequency as they arrive filling the burst structure. An example is illustrated in Figure 30 where two burst structures are placed inside a time Slot's Data area matrix. The order of the used sub-channels number and the order of the incoming bursts coincide. For more details about the sub-channel calculation the reader can refer to the Carrier Allocation paragraph of the previous chapter.

The matrix's data occupancy is calculated to be the following one. The total maximum size of the matrix in symbols is 1712 x 6 = 10272, thus approximately 10K symbols. The maximum number of burst structures transmitted during a Time Slot, in Medium Access Scheme 3 and 2K carriers is 54. That results to a total of 54 x (144 data + 30 pilot) = 7776 data + 1620 pilot = 9396 = 9.2K symbols. This means that up to 91.5% of the total matrix positions can be occupied by burst structures and the actual data reserve the 76% of the matrix space. So in this case the matrix is not so sparse. Nevertheless if we picture that the data are placed in a 1712x180 matrix where not all columns are occupied but instead the input to output reordering is limited to the first 6 columns, the result is that only the 3% is filled. If we observe the results we can conclude to some very interesting remarks. The Data area matrix itself is not necessarily sparse. On the contrary in the case of the maximum number of bursts it is almost full. The data in relation to the 1712x180 matrix, which consists the maximum case, are very few. That supports once more that the maximum matrix should not be stored entirely. On the other hand if we compare the size of the Data area matrix in Medium Access Scheme 3 (10272 symbols) to the maximum number of burst structure symbols in the previous schemes (23040 symbols), we observe that it is almost the half. That makes us suspect that the entire MAS3 Data area matrix could be stored.

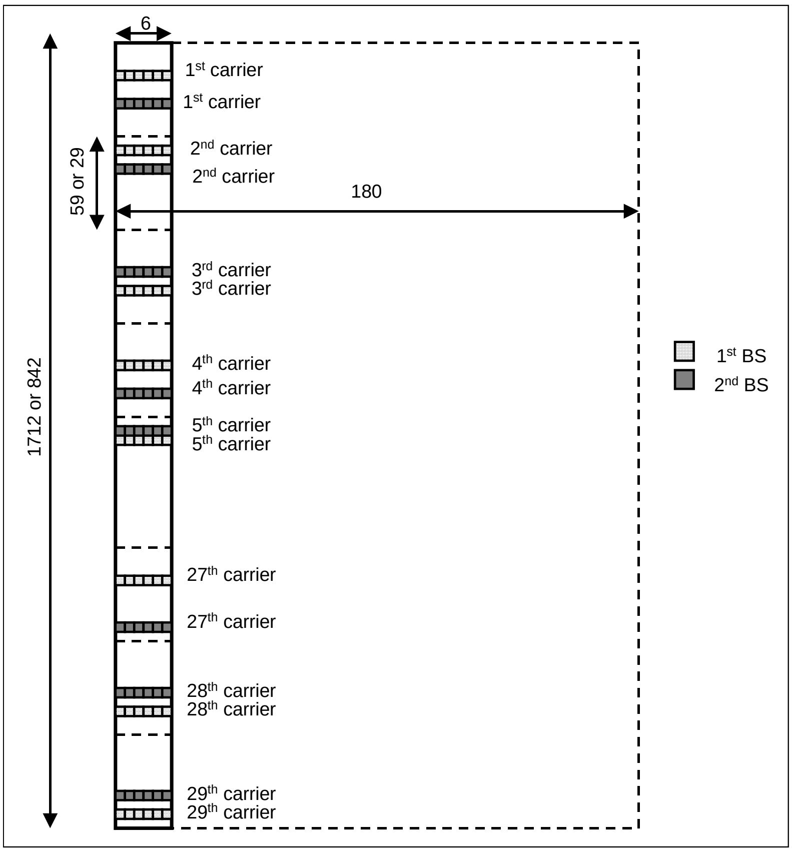

Figure 30 Placement of Burst Structures 3 in Data area matrix

5.3 Data Area Storage Scheme

In this paragraph, we will investigate the storage scheme for a minimum implementation cost. We know that each symbol is 6 bits long. The Data area matrix is generally sparse thus we will try to minimize the required memory space by storing only the absolutely necessary information. The selection will be based on the occupancy results provided in the previous section and it must be effective for all the Medium Access Schemes.

5.3.1 Medium Access Scheme 1

The occupancy of the Data area matrix in MAS1 is only 7%. The matrix is quite big (1712 lines x 180 column x 6 bits = 1.76Mbits) and very sparse. Therefore we turn down the alternative to store all its contents. Instead we will investigate what is the minimum number of symbols that must be stored and how to keep in memory the information about their placement in the Data area. The schematic of the idea for no frequency hopping appears in Figure 31, whereas the frequency hopping case is illustrated in Figure 32. We will explain in more details the concept depicted in the two figures.

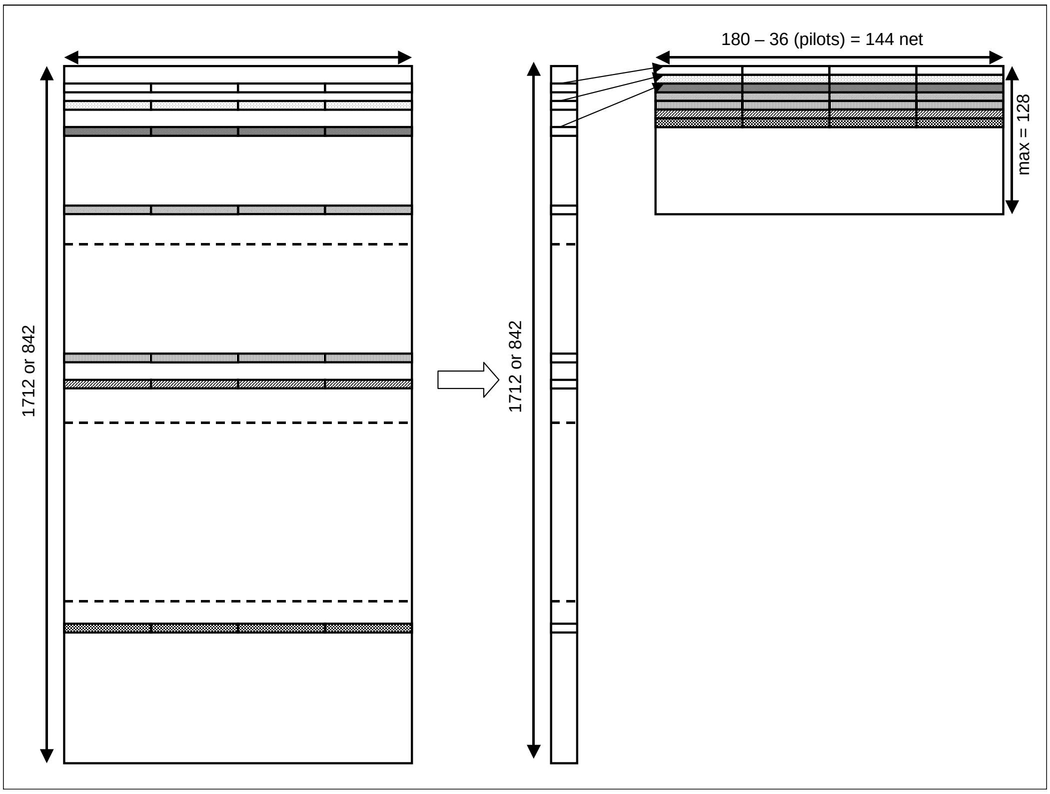

Figure 31 Scheme for minimizing storage in MAS1 without frequency hopping

The maximum number of burst structure information in the matrix is 23040 symbols or 23040x6 bits = 135Kbits, 27Kbits of which are pilot symbols. Can we save more space by avoiding storing the pilot symbols? The answer is yes since the pilot positions depend only on the number of the line they are placed in. Therefore all we have to know is whether a line is occupied by data and then the pilot pattern is fixed. So what we have to do is store the data symbols in their order of arrival. The maximum quantity of data symbols that will be reordered in a Time Slot, dictates the required memory space. Therefore the needed memory capacity is 108Kbits (128 max number of BS1 x 144 data symbols x 6 bits). From now on will call the memory where data are stored the Data memory (DMEM). Its exact dimensions will be defined after all three Medium Access Schemes are studied.

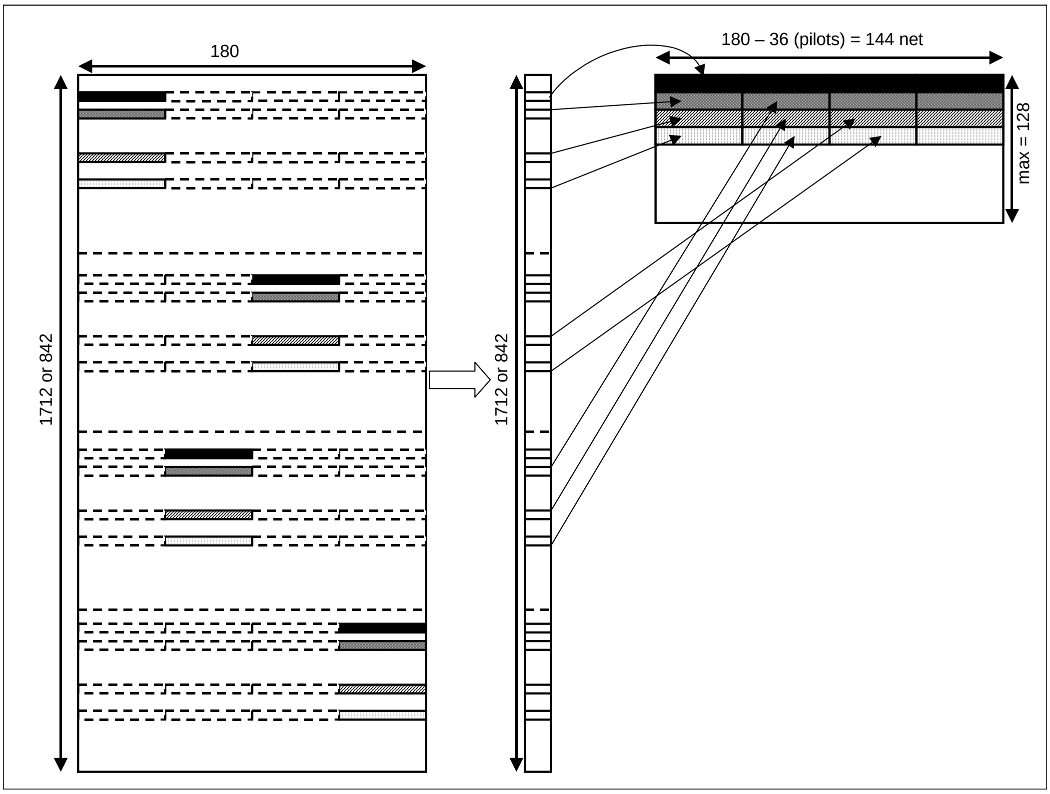

Figure 32 Scheme for minimizing storage in MAS1 with frequency hopping

The next step after storing the data is how to keep the information about which lines are occupied and by which data. Therefore apart from DMEM, another memory is required for the storage of a pointer table. That table, which from now on will be called PMEM, keeps the information of the original matrix's lines. Its height is equal to the matrix height, in other words to the number of usable carriers. The idea is to match each line with a pointer that will direct us in the DMEM position where the data related to that line/carrier can be found.

5.3.2 Medium Access Scheme 2

The occupancy of the Data area matrix in MAS2 is the one fourth of the same matrix in MAS1. The contained data represent the 30% of its space whereas the occupancy is 7% in relation to maximum case matrix of MAS1. In any case from this numbers we draw the same conclusions as before: the matrix is too big and sparse to be reproduced in a memory. Only the Time Slot's data symbols are stored. The followed analysis is exactly the same as for the frequency hopping case of Medium Access Scheme. The data symbols are stored inside DMEM whose required capacity is again 135Kbits. Pilot symbols follow the same pattern and do not need to be stored. The pointer table (PMEM) has the same functionality to correspond a line to its data. The idea is illustrated in Figure 33.

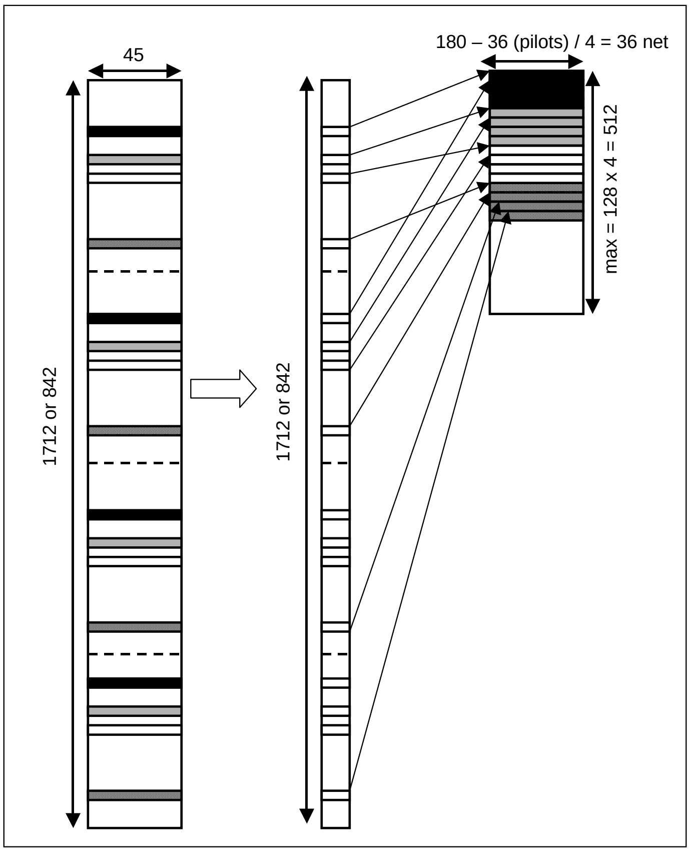

Figure 33 Scheme for minimizing storage in MAS2

5.3.3 Medium Access Scheme 3

The Data area matrix in MAS3 can be occupied by burst structures up to 91.5% and its total size is 1712 lines x 6 column x 6 bits = 61632bits = 60.2Kbits. Obviously the remarks made in the other cases are not valid here. If the system supported only Medium Access Scheme 3 we would decide to store the entire area in a memory. Even now we observe that the previous schemes require DMEM to store 135Kbits, which is double the capacity, required for the storage of the MAS3 Data area. So we can still consider the option to reorder the data directly and reproduce the entire matrix inside DMEM. Nevertheless, although this case can be handled differently, we select to find a storage scheme similar to the previous ones. The advantage of that choice is that all cases are treated in a more uniform way. Of course the specific decision is somehow subjective and a different designer might have decided to implement it otherwise. The illustration of the storage idea for MAS3 appears in Figure 34.

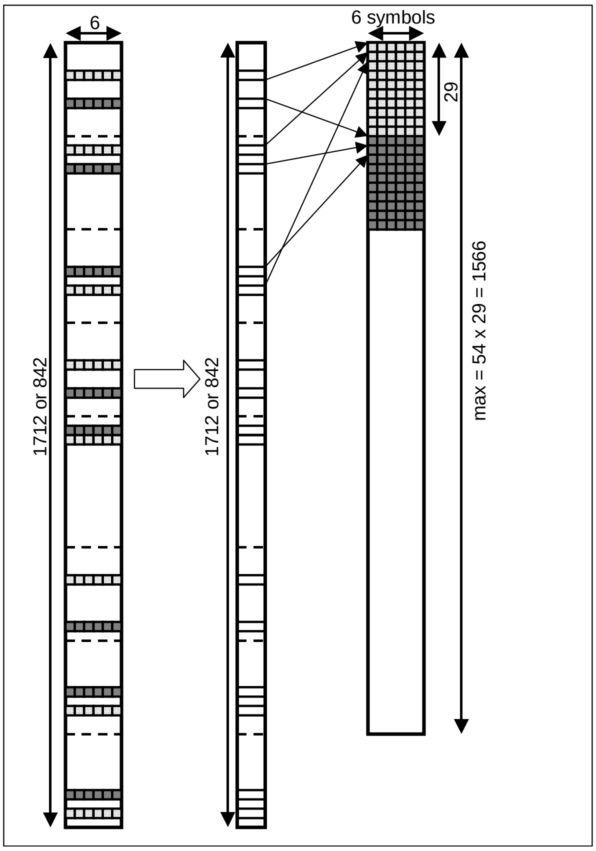

Figure 34 Scheme for minimizing storage in MAS3

The maximum number of burst structure information in the matrix is 9396 symbols or 9396x6 = 56376bits = 55Kbits, 9720bits (9.5Kbits) of which are pilot symbols. Once again we would like to avoid the pilot storage. The problem is that the pilot pattern in burst structures of the third type (BS3) is fixed for each burst and is a function of the burst structure's line number and not the matrix's. That is interpreted that the pilot positions influence the data positions in the matrix lines. Since the data will be reordered in frequency (columns) and we try to keep an alignment between data that are placed in the same matrix line, we have to omit the pilot positions in DMEM. The actual pilot symbols are not stored in DMEM but will be output on the fly as for the other Medium Access Schemes. Therefore the burst structure is reproduced inside DMEM by storing a burst's data in columns and skipping the pilot positions. That way, incoming data symbols are not stores in sequential DMEM positions but all data that must occupy the same Data area matrix line are actually placed in sequential positions.

A pointer table stored in PMEM is also needed. Each line/carrier has a pointer that directs us in the DMEM position where the data related to it can be found.

5.4 Ranging Area Storage Scheme

The matrix of a Time Slot also includes Null areas and sometimes a Ranging area. This is the case for Medium Access Scheme 1 and also the first Time Slot of Medium Access Scheme 2. In the third Medium Access Scheme the Ranging area is merged with the Data area, but the scheme introduced in this paragraph applies to all cases.

Null areas do not concern us since they are filled with null symbols. Ranging area contains apart from null symbols, ranging pilot symbols that are placed in the same column. The column and its occupied positions are defined by the Ranging transmission configuration. The carriers that consist the Ranging sub-channel (145 or 116 carriers wide) correspond to the used lines of the area. The parameter Ranging_Size indicates the width of the area (always 6 in MAS3) and Time_Symbol_Number the used column. Thus, if we allocate the used carriers and mark them in a 1712x1 table, stored in a memory (RMEM), then we have all the necessary information to reproduce the Ranging area since ranging pilots are available on request and can be output on the fly. The idea is illustrated in Figure 35.

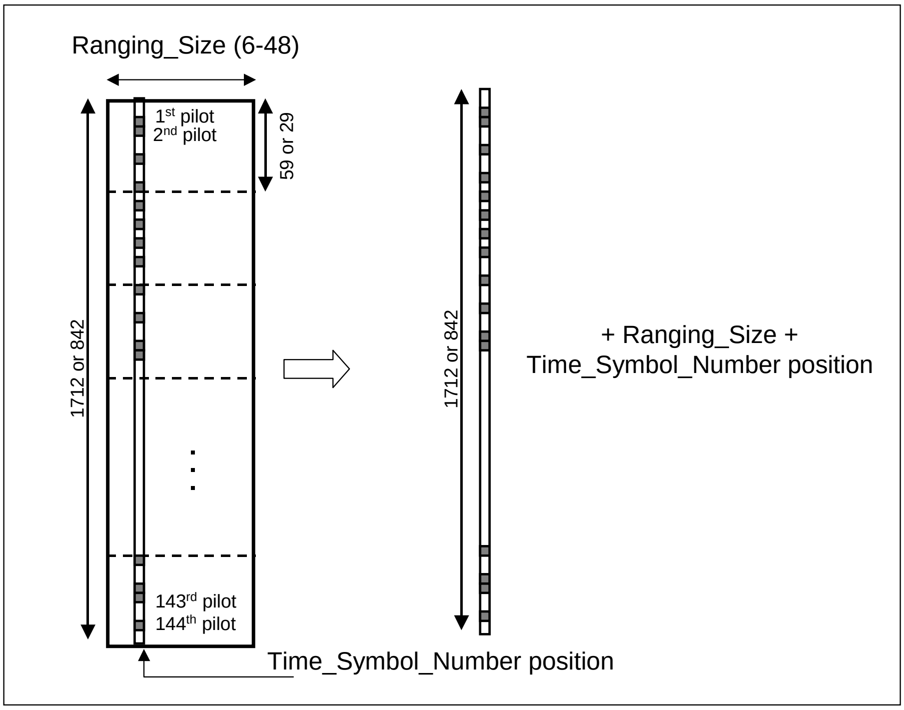

Figure 35 Scheme for Ranging area storage

5.5 Internal Organization

Following the so far analysis, we conclude that Frame Adaptation Block works on a Time Slot basis and is partitioned to three sub-blocks: an Input Controller, an Output Controller separated by a Storage Block.

The system parameters that configure the FRM operations are listed below. The Input stage, the Output stage or both can use them. The sub-blocks requiring them are noted in a parenthesis next to their name and then a brief description of their usage follows.

- **Constellation Type (OCTR):** It defines the Constellation Type used. It is used at the last output stage, to pre-map the issued data symbols accordingly. Three types are supported: QPSK, 16QAM and 64QAM.
- **FFT Size (ICTR, OCTR):** It indicates the number of points of the OFDM symbol (height of the transmitted slot). More specifically, it defines whether the transmission is done over a range of 1024 carriers (1K mode) or 2048 carriers (2K mode).

- **Unique Key (ICTR):** It is a constant required for the carrier allocation of MAS3 and Ranging Transmission. It ranges from 0 to 255.
- **Carrier Spacing (OCTR):** It indicates the spacing between the transmitted carriers. In the FRM, the spacing corresponds to the output rate of the symbols. There are three alternative values: 1KHz, 2KHz and 4KHz. These values are interpreted to an output rate of 1 symbol/4 clock cycles, 1 symbol/8 clock cycles and 1 symbol/16 clock cycles, respectively. The reasons will be explained in the design chapter.
- **Guard Interval Size (OCTR):** It indicates the length of the guard interval applied to an OFDM symbol, when Rectangular Shaping is used. In other words, the pause between issued columns, during which null symbols are output. Its length can be the 1/4, 1/8, 1/16 or 1/32 of the useful symbol duration. When the selected shaping type is Nyquist, then the guard interval is considered to be always ¼. The Guard Interval between symbols provides immunity against inter-carrier and inter-symbol interference.
- **Medium Access Scheme (ICTR, OCTR):** It indicates the Medium Access Scheme of the transmission. As mentioned before, there are three available MAS types. The Burst Structure and the Transmission Frame type can be deduced from the applied Medium Access Scheme.
- **Sub-Channel Numbers (ICTR):** Each Time Slot contains a number of Burst Structures. Each BS will be transmitted over a specific Sub-Channel. For that reason, the MAC must provide a number of Sub-Channel Numbers, equal to the number of BSs contained in the current Time Slot. The used carriers can be calculated from the SCNs, following the carrier allocation algorithms. Since the maximum capacity a Time Slot is 128 BSs, the required SCNs are kept inside a FIFO. They are stored in the arrival order of the Burst Structures they correspond to. Then ICTR can use them for the Time Slot's carrier allocation, requesting one SCN at a time.
- **Frequency Hopping (ICTR, OCTR):** It indicates whether the Frequency Hopping option is applied to the MAS1 carrier allocation. This information is needed by ICTR for the used carriers calculation and by OCTR for the correct interpretation of the stored information.
- **Shaping Type (OCTR):** It signals whether the Shaping Type is Rectangular or Nyquist. OCTR needs to know the Shaping Type for the insertion of the extra Nyquist symbols and the correct issuing of the time slices.

- **Ranging Sub-Channel Number (ICTR):** It is the identity number of the Ranging Sub-Channel and IDC it is needed for the calculation of the carriers over which the Ranging transmission is performed.
- **Ranging Size (OCTR):** It declares the size of Ranging Transmission, which is defined by the number of Ranging intervals. The parameter is expressed in number of OFDM symbols (columns) and the optional values are 6, 12, 24 or 48 OFDM symbols (1, 2, 4 or 8 Ranging Intervals, respectively).
- **Time Symbol Number (OCTR):** It specifies the position of the code inside the Ranging Transmission (0 to 5 corresponds to the 1^{st} interval, 6-11 the 2^{nd},..., 42-47 the 8^{th}). The OCTR needs it so it can issue in the correct column the Ranging Code.
- **Ranging Code Identity Number (OCTR):** It provides the identity number of the Ranging Code to be transmitted. From its range, OCTR deduces information about the type of Ranging Transmission (Long, Short or Bandwidth request). It is also used to request the actual Ranging pilot bits from the RNG block.

5.5.1 Input Controller

The Input Controller is the Frame Adaptation Block's interface with the previous stage of the Physical Layer chain, the Interleaver. In addition it is the part of FRM that receives the system configuration parameters. According to the configuration, the Controller performs the carrier allocation and stores the incoming data symbols and Time Slot information as dictated by the storage scheme defined in the previous paragraphs.

Apart from the configuration parameters required by OCTR, ICTR calculates and stores some other characteristic parameters of the Time Slot.

- **First TS:** It informs whether the current TS is the first one of a Transmission Frame. This information is required by OCTR in MAS2, in order to know whether Ranging Transmission will precede the Data Transmission.
- **Central carriers:** In case of Nyquist Shaping only a band of the OFDM slice is issued, and the central frequency of this band is different than the usual dc carriers. The location of this band inside the OFDM slice differs according to which mini-burst is transmitted when frequency hopping exists. Thus, IDC calculates one or four central carriers of the bands as described in Table 4.

- **Ranging bit offset:** It is the number of used carriers for Ranging Transmission whose value is greater then the dc carrier. Its use will be explained in the section describing the ODC functionalities.
- **Ranging length:** It flags that the used Ranging Sub-Channel is the one with the greatest RSCN (the last), thus it consists of 116 carriers instead of 145.

The arriving data symbols are stored in the memory destined for their storage (DMEM). The input rate does not affect the data storage process because a valid signal accompanies each data symbol. This valid flag is used to write enable DMEM. The order in which data are written in DMEM depends on the selected Medium Access Scheme. For the first two types they are stored in sequential positions, whereas for the third type a sort of structured reordering is required as described in the corresponding storage scheme paragraph.

The other important task of the Input Controller is the allocation of the used carriers. Each burst is transmitted over a sub-channel and each sub-channel consists of a group of one or more carriers depending on the Medium Access Scheme. The carrier allocation algorithms have already been presented in the chapter of the main concepts and definitions. The Sub-Channel Numbers are stored in the order their corresponding bursts arrive. It is the responsibility of the Input Controller to select the correct algorithm and apply it for each given Sub-Channel number. Each time a new burst arrives, the used carriers are calculated and are marked in PMEM as defined by the appropriate storage thing. Since the data of each burst are written in DMEM in a strict order it is easy to calculate the pointers that correspond to each used line without actually waiting for the storage of the burst to be completed. This process is repeated for all the sub-channels of the Time Slot, including the Ranging sub-channel, whose calculated carriers are stored in RMEM.

Finally Input Controller must store any other remaining information of the Time Slot. That requires an additional memory called CMEM. CMEM is devoted to the storage of Time Slot parameters. These are the configuration variables required by the Output Controller for the correct interpretation of the storage scheme, plus the extra parameters mentioned above that are generated by ICTR itself.

5.5.2 Storage Block

Storage block consists of two sets of all the memories plus a double buffering mechanism. We will refer as buffer to a memory set. The memories are the ones that have been introduced so far meaning:

- **DMEM:** The data memory where all the data symbols are stored. Its capacity is the maximum required by the storage scheme, thus, 108Kbits.
- **PMEM:** The pointer memory where each line corresponds to a used carrier and the pointers that match the carriers with their data are stored. Its capacity will be defined in the Design Chapter after defining the exact dimensions of DMEM.
- **RMEM:** The ranging memory where the lines correspond to the carriers occupied for Ranging transmission and its size is 1712bits.
- **CMEM:** The configuration memory where the remaining parameters and information of the Time Slot is written. If count the stored parameters listed in the ICTR paragraph and set a standard length of 32bits per word then its capacity is 14x32 = 448bits.

The double buffering mechanism controls which buffer is accessed by each controller. While Input Controller writes in first buffer, Output Controller is permitted to access the other buffer and vice versa. This allows the issuing of a Time Slot and the preparation of the next one to take place at the same time. This is after all a necessary condition for the output stream to be continuous.

5.5.3 Output Controller

The Output Controller is responsible for reading the data and information from the memories and issuing the Time Slot symbol by symbol in columns according to the system configuration. The purpose of this sub-block is twofold: it must supply the data - contained in the other buffer than the one that ICTR is writing to and the additional pilot symbols (provided by RNG and PLT), in a format appropriate for the forthcoming inverse FFT, as well as code the data correctly in order to pass through the Symbol Mapper (SYM) block.

In regard to the first task, the iFFT block processes the data of the Time Slot in a column-by-column basis. OCTR must therefore correctly deduce this information from the buffer (where data symbols are kept), the Pilot Generator block (to fetch the correct pilot symbols, depending on the carrier location) and the Ranging block (to fetch the correct ranging symbols if needed). Pilots and ranging pilots are accessible by a simple handshake protocol with the Pilot and Ranging Code Generator blocks. Thus, such a symbol is required, it must simply request it one cycle in advance from the appropriate block.

The information, about which lines of the Data area carry valid symbols, is extracted by reading for each position of a column, the corresponding line of PMEM. If a line is used, then the combination of the read pointer, the global column pointer and the pilot patterns permits us to decide what kind of symbol must be issued and where to retrieve it from (DMEM or pilot generation block). A similar process is performed for the Ranging area, only that RMEM is the accessed memory; while in case of MAS3 both memories (PMEM and RMEM) are read at the same time. Obviously PMEM (RMEM) is scanned a number of times equal to the width of the Data area (Ranging area).

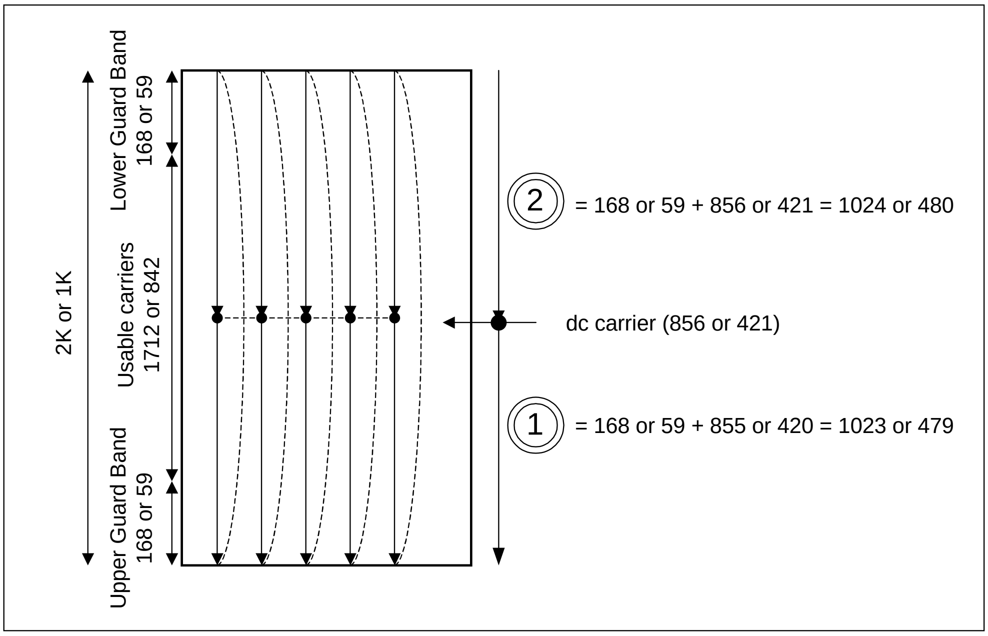

Figure 36 Symbol transmission order

Although the Time Slot matrix is issued on a column-by-column basis, there is the constraint that the iFFT algorithm considers the dc carrier to be the first one. Thus a column's symbols are not output from top to bottom. The first transmitted point of an OFDM slice is the one corresponding to the central frequency (dc carrier), whose value is

856 in 2K or 421 in 1K. After that, OCTR issues the symbols placed on the carriers whose value is greater than the dc carrier. Then follow the upper and lower guard band's null symbols. Finally, the symbols belonging to the first usable carriers are issued. The output order makes now clear the necessity of the Ranging offset parameter, generated by ICTR. The RNG block needs to know which is the first code bit that corresponds to the first ranging carrier above the dc value. Resuming, we see that the output order of a column's symbols starts from the middle to the end; it wraps around and continues from the beginning back to the middle. The guard interval points follow afterwards. The scheme appears in Figure 36.

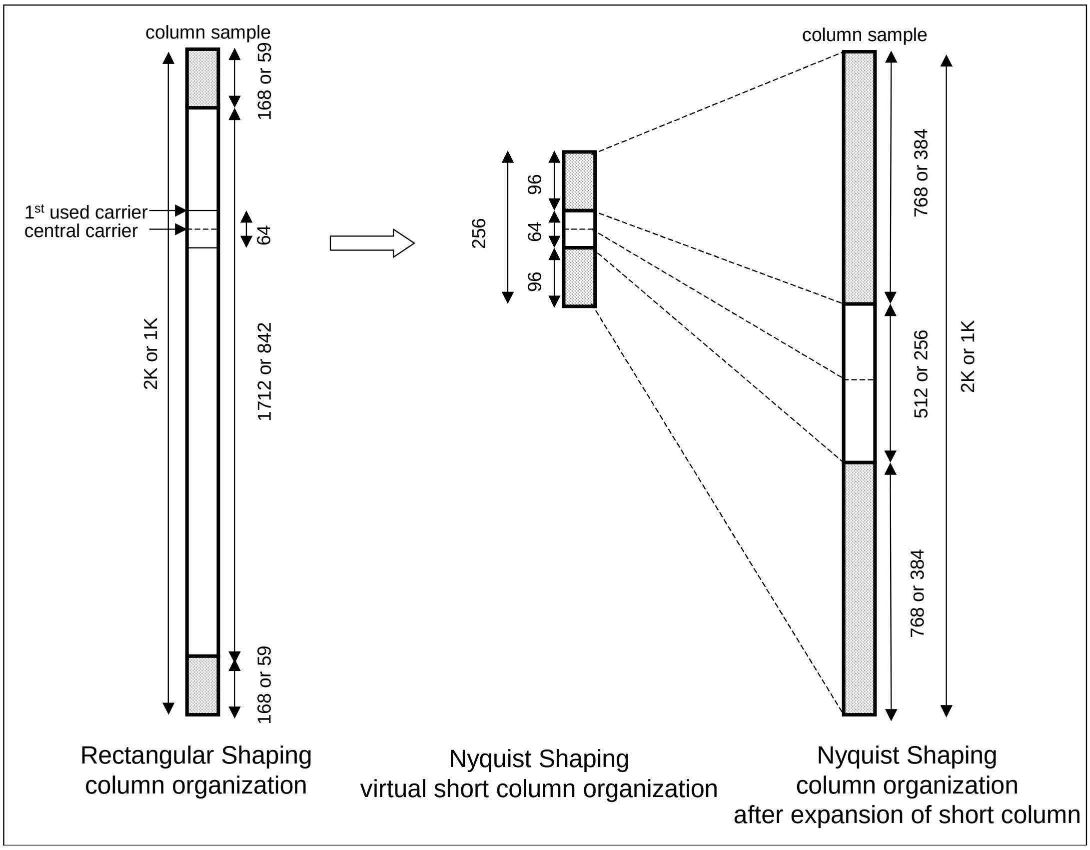

Figure 37 Expansion of virtual short column in case of Nyquist shaping

The above stand in case of Rectangular Shaping. When Nyquist Shaping is applied, the maximum value of the allocated Sub-Channels is 64 and their SCNs are consecutive. Thus, we can picture the data being concentrated in a horizontal band. This new slice has a different central frequency in the place of the usual dc carrier. The central carrier for Nyquist transmission is calculated in relation to the first sub-channel number of the Time

Slot, as presented in Table 4 of the Carrier Allocation paragraph. This information is generated and provided through the buffer by ICTR and it is a function of the carrier used by the first burst structure of the Time Slot. In case of frequency hopping four different central carriers are estimated, one for each mini-burst. Therefore, the position of the horizontal band varies. All this arrangement is just a trick, which aims to treat both shaping types in a uniform way. In reality, the horizontal band is expanded to the full matrix height, 2K or 1K lines, by issuing three or seven null symbols after each symbol of the band. Figure 37 illustrates the idea. The first column depicts how the common column format with Rectangular Shaping. The next two columns illustrate the expansion from the virtual short to column to full sized one. Let's remind that in that last column 7 out of 8 symbols in 2K mode, or 3 out of 4 in 1K, are null.

Finally, the last task of OCTR before a symbol is issued is to code them in a single 7-bit word, which indicates to the Symbol Mapper whether the symbol is data, pilot, ranging or the Null symbol. Table 14 lists this pre-mapping.

Representation (decimal)	**Symbol**
0 - 3	QPSK data symbol
4 - 19	16QAM data symbol
20 - 83	64QAM data symbol
84	Boosted pilot 0
85	Boosted pilot 1
86	Non-boosted pilot 0
87	Non-boosted pilot 1
88	Ranging pilot 0
89	Ranging pilot 1
90	Null symbol

Table 14 Symbol pre-mapping to 7-bit word

6 Frame Adaptation Unit Design

After introducing the main concepts and definitions of the system and the stages of the Physical Layer, we've analyzed the Frame Adaptation Unit's main functional tasks. Through this analysis, the main architecture of the unit has been defined. In this chapter we will study in depth the design details.

The issues that will be resolved concern the memory, the change of clock domain, the implementation of the memory access controls, as well as the datapath of a time consuming calculation. Although the buffer capacity has been defined, it remains to select the exact dimensions and type of each memory. In addition, we must handle the problems that occur from the fact that the input arrives from a different clock domain (100MHz) than the unit's target clock (36MHz). Furthermore, the processes performed by the Input and Output Controllers and described verbally in the previous section, must be translated to precise Finite State Machines. Finally, we will present an implementation of a modulo operation, required in the carrier allocation algorithm of MAS3 and Ranging sub-channels.

6.1 Memory specifications

So far we have defined the capacity of DMEM, RMEM and CMEM to be 108Kbits, 1712 bits and 448 bits respectively. Still remain two pending issues. The first concerns the exact dimensions of the above memories. The last issue concerns the size of PMEM and it will be resolved after the first one is answered. Finally we will indicate the required memory type.

For Ranging and Configuration memories the decision is a one way. RMEM is a 1712x1 memory since one bit at a time is set and also it is read bit-by-bit. CMEM, which keeps the Time Slot's parameters, will be sized 14x32 bits, providing a 32bit word for each parameter.

As far as it concerns DMEM, we take into consideration its input and output throughput. Both the Input and Output Controller access one symbol at a time, thus, never more than 6 bits per clock cycle. Consequently, we chose the width of a DMEM word to be 6 bits. Since the capacity is 108Kbits, the final DMEM dimensions are

18432x6 bits and each stored symbol is addressed separately. For the addressing`15 bits are required.

Now that the DMEM dimensions are defined and we know that its addresses consist of 15 bits, we can specify the exact contents of PMEM. Each PMEM word serves to indicate whether the carrier that corresponds to its address is used, and where to find the data to be placed over that carrier. In MAS1 without frequency hopping a line may contain one burst. In MAS1 with frequency hopping and in MAS2 a carrier is used for the transmission of a mini-burst. Finally, in MAS3 a carrier may be occupied by a BS3 line. From the above we understand that we must store a pointer that will direct us to the beginning of the burst, mini-burst or BS3 line. The question is what pointer length we must choose.

Lets suppose that we store the entire address of a carrier's first symbol. In that case the PMEM word should be 15 bits. Since there are address values that remain unused (15 bits can address 32768 memory lines and DMEM has only 18432 lines), we can exploit that to mark the validity of a carrier. Therefore, if all the bits of DMEM are set when it is cleared, then after a Time Slot is stored, the lines containing words whose value is different than 111111111111111, are used and the stored value is the required pointer. All other PMEM lines filled with ones are not used for transmission. That way, we avoid the addition of an extra bit as validity flag, and save 1712 memory bits while the total PMEM size is 1712x15bits = 25680bits = 25.08Kbits.

On the other hand we could store the pointer information in less bits. That could be done if we were to store not the entire address but just an identity number of a burst, mini-burst or BS3 line. We are going to examine how the PMEM word signification can be organized in each Medium Access Scheme case.

In the no frequency hopping case of MAS1, it is sufficient to correspond each line to a burst number expressing its arrival order. Since the data of a burst arrive together and are stored in sequential DMEM positions, the burst's starting position is sufficient. The starting address can be calculated by multiplying 144 with the burst number described above. If frequency hopping is applied, the same thing must be done for each mini-burst. The reason is that in this case each used line contains only one quarter of the burst. The pointer not only indicates which mini-burst is placed over that line but furthermore its modulo 4 implies which quarter of the line is occupied. The mini-burst's starting address is 36 times its increasing number. Following the above it is concluded that the pointers' range can be 0 to 127 bursts or 0 to 511 mini-bursts. That requires 9 bits per pointer. In addition, the used lines must be distinguished from the empty ones. That could be indicated with an invalid pointer value but since the entire range is used, an extra bit is added, used as a flag. The total size of PMEM for Medium Access Scheme 1 is 1712x10 = 17120 bits = 16.72Kbits. Figure 38 depicts the fields of a PMEM word.

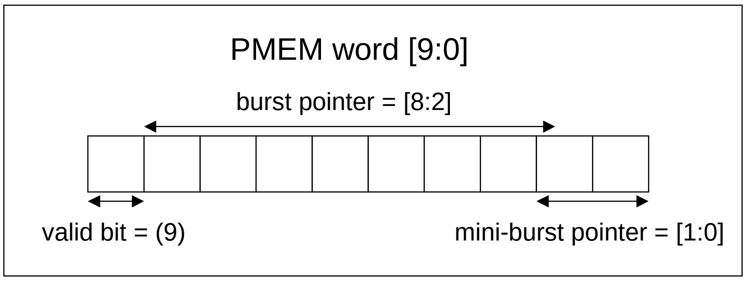

Figure 38 PMEM word structure for MAS1 and MAS2

In Medium Access Scheme 2, the pointer use is similar to Medium Access Scheme 1 with frequency hopping. It corresponds each line to a mini-burst number. The difference is that the line is now shorter and filled entirely by one mini-burst. Therefore the order of the mini-burst inside the burst does not interest us as it did when frequency hopping was applied. The pointer's length is again 9 bits (range from 0 to 511) and an extra validity bit is added. We observe that PMEM has the structure as for Medium Access Scheme 1. Its total size is again 16.72Kbits.

We will use the term burst line for the quantity of symbols that consist a BS3 line. In Medium Access Scheme 3 the lines of PMEM are corresponded to a burst line number expressing its order. We have in total 54x29 = 1566 burst lines. They require an 11bit pointer in order to be indexed. Since many values of the pointer range remain unused we can use them to flag an empty line. More specifically, in order to be in line with the other Medium Schemed, when all the bits are set, that signals an unused carrier (we consider the valid bit in the previous cases to be of negative logic) and the pointer range will be 1 to 54x29 - 1 = 1565. That way we avoid adding an extra valid bit and exchange the cost of 1712 bits of memory space with an 11bit comparator to zero. The total size of PMEM for Medium Access Scheme 3 results to be 1712x11 = 18832 bits = 18.4Kbits. The PMEM word contains only the 11-bit pointer.

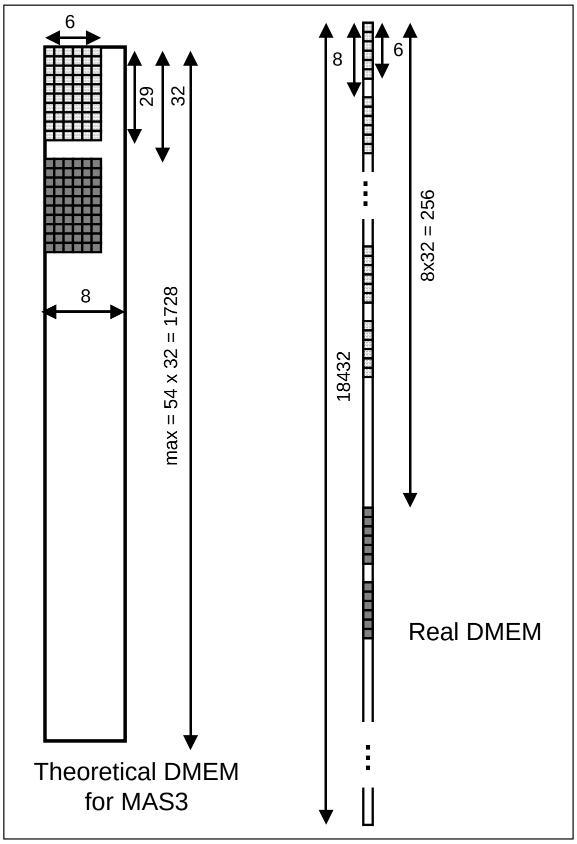

Figure 39 Comparison of BS3 placement in theoretical and real DMEM

The initial address of a burst line's first symbol can be calculated depending on the order data are stored inside DMEM. In the previous schemes the incoming symbols were stored in consecutive positions. In Medium Access 3 we can still store the symbols in the order they arrive. The difference is that since the maximum quantity of data is less than the total capacity of DMEM we can try to arrange the stored symbols to be aligned in powers of 2. More specifically, each burst line occupies 6 DMEM words, but if we were to reserve 2 more words, then the columns would follow a 8-word alignment. This can be extended to a burst structure level. Although a BS3 can be stored in 174 DMEM positions, we can align the burst storage every 256 addresses. Figure 39 depicts the idea, where the theoretical DMEM (presented originally in Figure 34) is extended and then the placement of data in the real DMEM is illustrated. This arrangement is equivalent to storing 32 lines of eight symbols each per burst structure. The pointer range changes (1 to 54x32 = 1728) but its required length is still 11 bits. The specific burst structure arrangement offers us the advantage to extract the following information from the pointer: the general order of the burst line (thus its beginning in DMEM) the BS3 order (pointer div 32) and the order of the burst line inside the BS3 it belongs to (pointer mod 32). Then, the initial address of each burst line is 32 times (pointer value –1). The PMEM word structure for MAS3 is shown in Figure 40.

The conclusion is that with 11 bits we can store the necessary pointer information for all the Medium Access Schemes without keeping the entire address. For the first two schemes the most significant bit is not used.

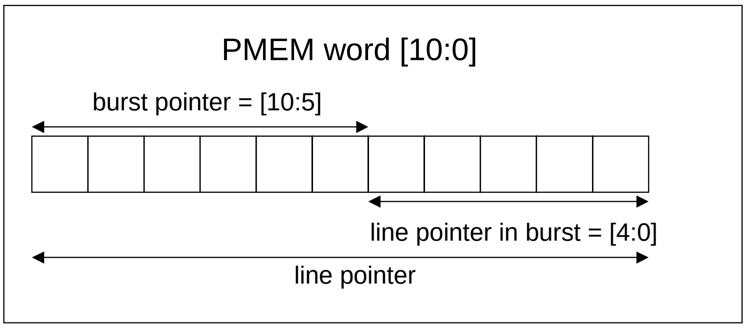

Figure 40 PMEM word structure for MAS3

It remains to decide which pointer forms we will implement: the 15-bit full address pointer or the 11-bit “group” pointer. We must prove that gained memory space is worth the implementation of the second pointer structure. The full address storage requires PMEM capacity to be 1712x15 = 25680bits whereas for the second solution 1712x11 = 18832bits are sufficient. If we consider that the Storage block consists of two identical memory sets, then the 6848-bit difference is doubled. Consequently, choosing the “group” pointer means we can save the space of approximately 80K transistors, a considerable quantity.

After the above analysis, we can conclude that the Storage block includes two identical buffers, each of which consists of the following four memories:

- **DMEM:** 18432x6 = 110592 bits
- **PMEM:** 1712x11 = 18832 bits
- **RMEM:** 1712x1 = 1712 bits
- **CMEM:** 14x32 = 448 bits

Therefore each buffer’s capacity is 128.5Kbits, which makes a total of 257Kbits for the entire Storage block.

The used memories are single ported RAMS. The general memory interface is listed in Table 15. Read data are available at the out of the memory one cycle after their request.

Pin	Width	I/O	Description
CLK	1	i	Clock
WEN	1	i	Write Enable signal (negative logic)
OEN	1	i	Output Enable signal (negative logic)
CSN	1	i	Chip select signal (negative logic)
ADR	L	i	Address
D	W	i	Input Data
Q	W	o	Output Data

Table 15 Generic pin list of an HxW memory cut, where $2^{L-1}<H\leq 2^{L}$

6.2 Input Resynchronization

In the first paragraph of the previous chapter we have discussed the input rate of the Frame Adaptation Unit and we have also defines the target clock to be 36MHz. Also given the latencies of the previous stages, we have concluded that the condition for the uninterrupted operation of the unit can be met only if the clock of the blocks up to FRM is 100MHz. Apparently, that means that the input interface of our unit crosses two clock domains and thus needs to be resynchronized.

A solution would be to use dual port memories. In that case the clock domain boundary would be shifted from the input interface to the Storage block. The clock of the port on the Input Controller's side would be 100MHz, whereas the Output Controller's port would work at 36MHz. The disadvantage is that such a memory has almost a double cost in transistors. Fortunately this implementation can be avoided.

The alternative solution is to add an elastic buffer at the input of the Frame Adaptation Unit. The data arrive in bursts of 144 symbols and the fastest rate is one symbol every two cycles, which corresponds to 50MHz frequency. Since the input stage can store one word per cycle, the maximum input to output frequency rate is 1.4. In other words in the worst case, Input Controller is half a time slower. A 144x6 show ahead elastic buffer can cover our needs. The input valid flag will function as a write enable signal while the inversed empty flag of the buffer will play its role. The other input flags can be synchronized by passing through 2 (or even better 3) flip-flops to prevent any occurrences of metastability. A basic block diagram of the unit appears in Figure 41.

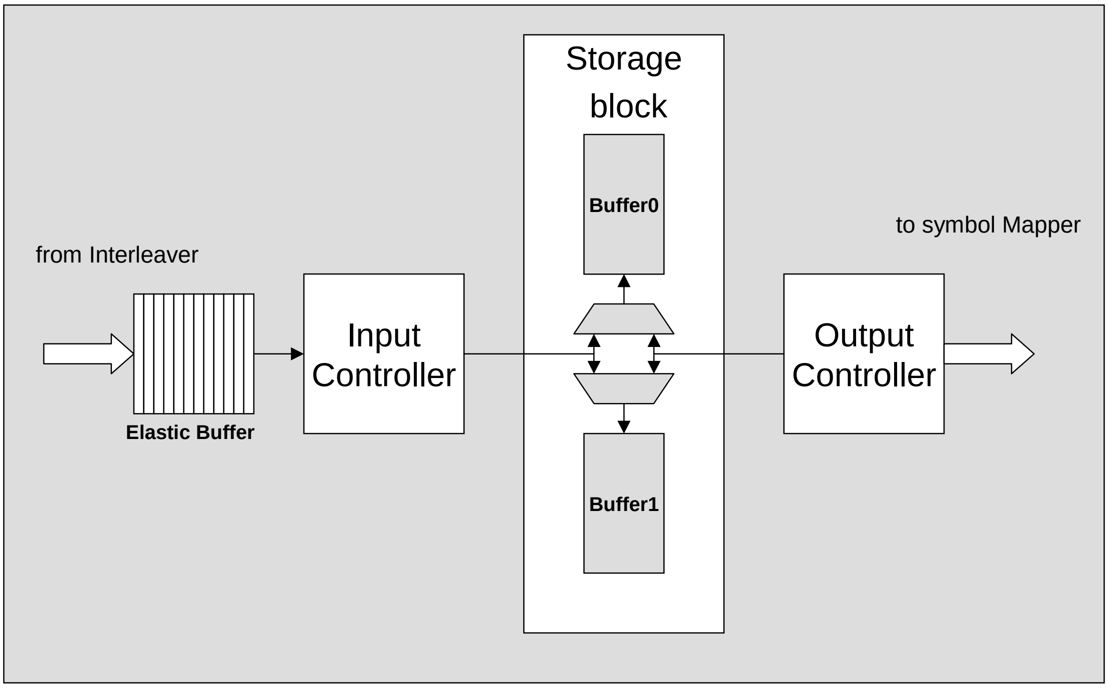

Figure 41 Block diagram of Frame Adaptation Unit

6.3 Input Controller Implementation

The Input Controller's main tasks, on whose implementation we will concentrate our interest, are the data storage and the carrier allocation along with the pointer storage. Two Finite State machines control the DMEM and PMEM access. We repeat that the received data are read from an Elastic Buffer and the main input control flags are synchronized through three flip-flops. The inverted empty flag of the buffer now replaces the valid symbol flag: as long as the buffer is not empty the symbol at its output is valid. Each FSM is presented in detail through a state diagram and a detailed description of each states input and output signals.

6.3.1 Data Storage FSM

The data symbol storage s controlled by a FSM, whose state diagram is depicted in Figure 42. It consists of one idle state, then one state for operation in Medium Access Schemes 1 and 2, and four more states for Medium Access Scheme 3.

The state transitions depend on some basic control. The basic control signal is the empty flag of the Elastic buffer, since this signal is the one that informs about the validity of the buffer's output and enables the data storage and the reading of the next position of the Elastic Buffer. In addition another flag called TS_ok is used to inform that the last symbols of the last burst of a Time Slot have been received in the Elastic Buffer. When the end of slot signal (eos) arrives, an intermediate flag, last_bs, is asserted in order to mark that the incoming burst is the last for the current Time Slot. TS_ok is set when last_bs is true and the end of burst (eob) signal arrives.

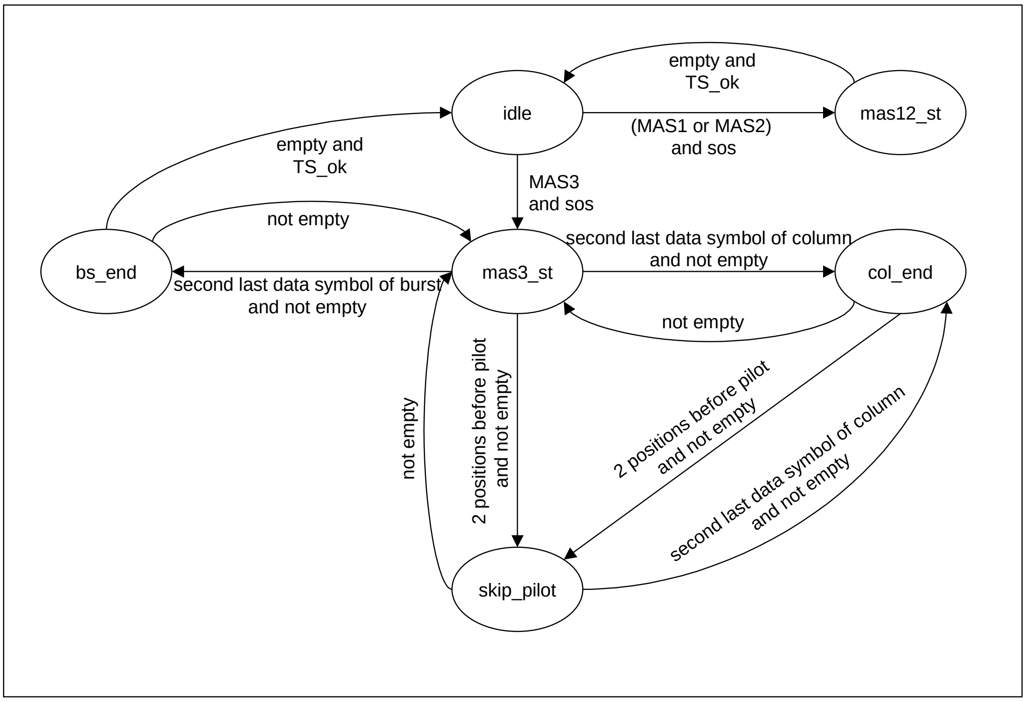

Figure 42 Data storage FSM

Two memory accesses take place at the same time. The first one is a read access from the Elastic Buffer and it is enabled with a signal called rd_en. It is important to remember that the Elastic Buffer functions as a show ahead FIFO, meaning that the oldest stored symbol's value is valid at the output and rd_en generates the preparation of the next symbol. The second memory operation is the write access of DMEM and it is enabled through dmem_wr_en. Both enable signals are asserted when we've left idle state and the Elastic Buffer is not empty.

A set of counters is used for the generation of the accessed address as well as for the control of the state transition. One is used for MAS1 and MAS2, whereas three different counters are employed in MAS3

- **Data_cnt:** It counts the number of incoming data. It ranges from 0 to 18432, thus it is 15 bits long. It is reset when all the data symbols of the Time Slot have been stored.
- **Line_cnt:** It counts the number of a BS3column's symbols that are stored or omitted. It ranges from 0 to 28 and consists of 5 bits.
- **Col_cnt:** It keeps track of a BS3 stored columns. It ranges from 0 to 5 and its length is 3 bits.
- **Bs_cnt:** It counts how many bursts of the third type (BS3) have been stored. Its range is from 0 to 53 and requires 6 bits.

The enabling of the counters is done through data_en, line_en, col_en and bs_en respectively, whose assertion depends on the current state of the FSM.

Dmem_addr is the DMEM address where the symbols are stored and it takes its value from the counters. More specifically for the first two Medium Access Schemes the address corresponds to data_cnt, since the symbols are stored in consecutive positions in their order of arrival. In MAS3 the dmem_addr corresponds to the concatenation of bs_cnt, occupying the most significant bits of the address, then line_cnt and finally col_cnt, placed in the least significant address bits. That way, the symbols are stored following the pattern described in the previous chapter and depicted in Figure 39 and the required allignment is automatically applied. The lenght of the bs_cnt, line_cnt and col_cnt deduce the 256, 32 and 256 alignment.

The initial state is idle state. While the FSM is in that state the input stage remains idle and all output signals take their reset values. That means that the memories' accesses and the counters are disabled. From all the counters only line_cnt is set to one because the first position of a BS3 is a pilot, thus it is skipped. All the other counters are reset to zero.

In the two first schemes the storage is simple and straightforward since the data symbols are stored in consecutive addresses of DMEM. Only one state, mas12_st, is used to indicate that the memory access process is enabled.

In MAS3 the calculation of the accessed address is a bit more complicated since the data must follow a specific pattern as explained in the Memory specifications paragraph: pilot positions must be omitted and the stored burst structures are stored aligned inside the memory. In order to control that, the counters increase depending on the current state and whether they are enabled (empty flag reset). We will examine each state in more details.

While in state mas3_st, line_cnt increases normally by one. This state corresponds to the consecutive positions of a BS3 column that are occupied by data symbols. No other counter increases. When a pilot position is approached we move to skip_pilot state. The condition that controls the transition depends on the col_cnt and line_cnt values since the pilot pattern is different for each column. A state transition to col_end state occurs when the end of a column is near and it is detected by the line_cnt value. Finally if the burst's storage is about to be completed, in other words if the last column comes to an end, then instead of going to col_end we move to the fourth MAS3 state, bs_end.

The skip_pilot state resembles to mas2_st. The difference between these two states is that line_cnt increases by two when enabled. That way a position is reserved empty in DMEM, omitting that way a pilot. Under normal circumstances our machine will return to mas3_st. On the other hand, if col_cnt and line_cnt indicate that the column is about to be completed, the next state is col_end.

In the col_end state line_cnt is not the only enabled counter. Line_cnt is reset to an initial value that varies from column to column. The explanation is once again the different pilot patterns between columns. In addition col_cnt augments by one. If a pilot is right after the start of a column then the next state will be skip_pilot, whereas under ordinary conditions it is mas3_st.

The final state that interests us is bs_end. In this state it is the turn of bs_cnt to increase, whereas col_cnt and line_cnt are reset to zero and one respectively. If more bursts remain to be received for transmission in the same Time Slot then we return to mas3_st and the process is repeated. On the other hand if TS_ok is set, that means that the stored burst was the last one and consequently we return to idle state until the next Time Slot's data arrive.

The following table lists all the actions taken in each state, and the required conditions for a transition.

STATE	TRANSITION	DESCRIPTION
idle	mas12_st: sos = 1 and MAS = 1 or 2 mas3_st: sos = 1 and MAS = 3	rd_en, dmem_wr_en, data_cnt, data_en, bs_cnt, bs_en, col_cnt, col_en, line_en, dmem_addr = 0 line_cnt = 1
mas12_st	idle: empty = 1 and TS_ok = 1	rd_en, dmem_wr_en, data_en = not empty data_cnt + 1 if enabled dmem_addr = data_cnt
mas3_st	skip_pilot: empty = 0 and {(col_cnt = 0 and line_cnt = 4 or 10 or 16 or 22) or (col_cnt = 1 and line_cnt = 5 or 11 or 17 or 23) or (col_cnt = 2 and line_cnt = 0 or 6 or 12 or 18 or 24) or (col_cnt = 3 and line_cnt = 1 or 7 or 13 or 19 or 25) or (col_cnt = 4 and line_cnt = 2 or 8 or 14 or 20) or (col_cnt = 5 and line_cnt = 3 or 9 or 15 or 21)} col_end: empty = 0 and {(col_cnt /= 5 and line_cnt = 27) or (col_cnt = 4 and line_cnt = 26)} bs_end: empty = 0 and (col_cnt = 5 and line_cnt = 27)	rd_en, dmem_wr_en, line_en = not empty line_cnt + 1 if enabled dmem_addr = bs_cnt & line_cnt & col_cnt
skip_pilot	col_end: empty = 0 and (col_cnt = 3 and line_cnt = 26) mas3_st: empty = 0 and any other case	rd_en, dmem_wr_en, line_en = not empty line_cnt + 2 if enabled dmem_addr = bs_cnt & line_cnt & col_cnt
col_end	skip_pilot: empty = 0 and col_cnt = 0 mas3_st: empty = 0 and col_cnt /= 0	rd_en, dmem_wr_en, line_en, col_en = not empty line_cnt = 0 if enabled and col_cnt /= 4 line_cnt = 1 if enabled and col_cnt = 4 col_cnt + 1 if enabled dmem_addr = bs_cnt & line_cnt & col_cnt
bs_end	mas3_st: empty = 0 idle: empty = 1 and TS_ok = 1	rd_en, dmem_wr_en, line_en, col_en, bs_en = not empty line_cnt = 1 if enabled col_cnt = 0 if enabled bs_cnt + 1 if enabled dmem_addr = bs_cnt & line_cnt & col_cnt

Table 16 Data storage FSM actions

It must be clarified that the detection of a pilot or the end of a column is done two positions before because the counters take their increased value after one cycle. That way when in mas3_st and line_cnt points two positions before a pilot, we go to skip_pilot stateand the new line_cnt points exactly before the pilot position. A general timing diagram is depicted in Figure 43, where p is the skipped position (a line_cnt value for a given col_cnt value) that omits a pilot. The values for the pilot and end of column detection have been extracted by numbering the rows and columns of a burst structure 3. The pilot pattern is depicted in Figure 44 from where we can deduce the coordinates of each symbol.

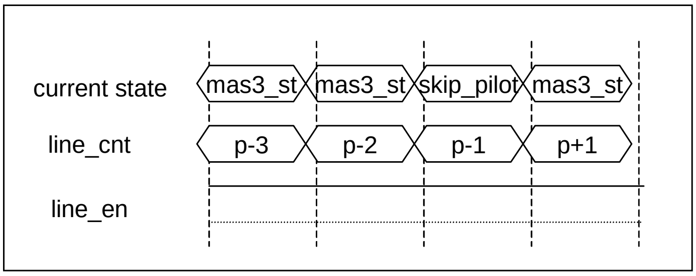

Figure 43 General timing of state transition and line_cnt increase in MAS3

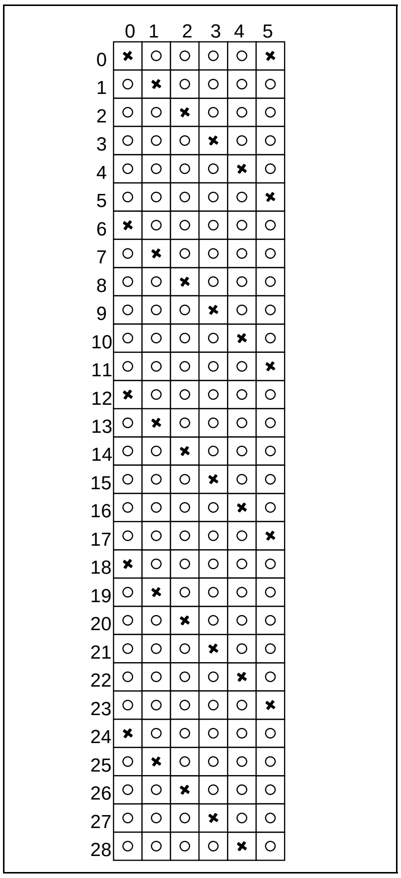

Figure 44 Burst Structure 3 pattern coordinates

6.3.2 Carrier Information Storage

The second main task of the FRM is the storage of the carrier related information. For that purpose carrier allocation of each burst's sub-channel is performed and then the results are used for the storage inside PMEM of pointers that link the used carriers with their data.

Each Time Slot may have a different configuration and uses a number of sub-channels each of which is assigned to a burst. For that purpose, a part of the Frame Adaptation Unit is dedicated to register all the configuration parameters of the Time Slot and also includes an Elastic Buffer, that stores in a FIFO way the required Sub-Channel Numbers. This FIFO is "show ahead" and each time a sub-channel's calculation is completed, the next Sub-Channel Number is de-queued.

The selection of the carrier allocation algorithm is performed according to the applied MAS. The implementation of carrier allocation in case of MAS1 and MAS2 is quite simple and straightforward. All the required constants are hardwired and the calculation is just a matter of multiplexing depending on the Sub-Channel Number range and the order of the computed carrier. The MAS3 carrier allocation is more complicated as it requires complex operations. Two ROMS, one for 2K mode and one for 1K mode, are used for keeping the series of numbers, which consist the initial permutation. Below, we repeat the formula used for the carrier calculation:

Carrier#n = total_channels x n + (Index(n) + Unique_Key) mod total_channels

and

if Carrier#n≥856 then Carrier#n=Carrier#n+1

where:

- *total_channels* the total number of Sub-Channels (29 in 1K, 59 in 2K)
- *n* the index of each number inside the selected permutation (0 to 28)
- *Index(n)* is the actual number which corresponds to index n.
- *Unique_Key* denotes a key (0 to 255), provided by the MAC process and which will be unique to each Upstream Channel

The used permutation depends on the Sub-Channel Number. This number indicates the first read ROM position. Each carrier is calculated in two steps, which form a pipeline: while the final value of a carrier is generated, the next carrier's calculation is prepared. In the first stage, we calculate the sum *Index(n)* + *Unique*_Key, as well as total_*channels x n*. The multiplication is done through multiplexing of different results according to n. The second stage performs a modulo operation on the previous sum and adds the two terms of the equation.

One of the interesting parts of this formula's implementation is the calculation of modulo total_channels. Total_channels is either 29 or 59. None of these value is a power of two, which makes the calculation to be complicated and time consuming. Based on the fact that Index(n), Unique_Key and total_channels belong to a specific range, it was decided that modulo total_channels (29 or 59) of Unique_Key and Index(n) sum, will be calculated with field checking and subtraction. That way the operation can fit in one cycle.

The constant Unique_Key takes a value from 0 to 255. The ROM output (Index(n)) is a number that varies from 0 to 58 (in 2K) or 0 to 28 (in 1K). Consequently the range of these two numbers' sum is 0 to 314 (in 2K) or 0 to 284(in 1K). The sum range can be divided in the fields listed in Table 17.

	2K (mod 59)		1K (mod 29)	
	sub-ranges (lower to upper limit)	subtraction number	sub-ranges (lower to upper limit)	subtraction number
field 0	0 to 58	0	0 to 28	0
field 1	59 to 117	59	29 to 57	29
field 2	118 to 235	118	58 to 86	58
field 3	236 to 294	236	87 to 115	87
field 4	295 to 314	295	116 to 144	116
field 5			145 to 173	145
field 6			174 to 202	174
field 7			203 to 231	203
field 8			232 to 260	232
field 9			261 to 284	261

Table 17 Range fields for mod 29 or 59 of Unique_Key and Index(n) sum

The lower limits of each field are hardwired. The sum is compared whether it is less than each limit value (except the value 0). The results of the nine comparisons are concatenated into a vector, which is encoded to the number of the sub-range that the sum belongs to. Depending on the field number, it is decided which value will be subtracted from un_key_sum in order to take the result of mod 59 or 29. The datapath of the operation is depicted in Figure 45 and the result vector's encoding in Table 18.

	2K (mod 29)		1K (mod 59)	
	encoded value	result vector	encoded value	result vector
field 0	0	111111111	0	111111111
field 1	1	111111110	1	111111110
field 2	2	111111100	2	111111100
field 3	3	111111000	3	111111000
field 4	4	111110000	4	111110000
field 5	5	111100000	5	111100000
field 6			6	111000000
field 7			7	110000000
field 8			8	100000000
field 9			9	000000000

Table 18 Encoding of result vector

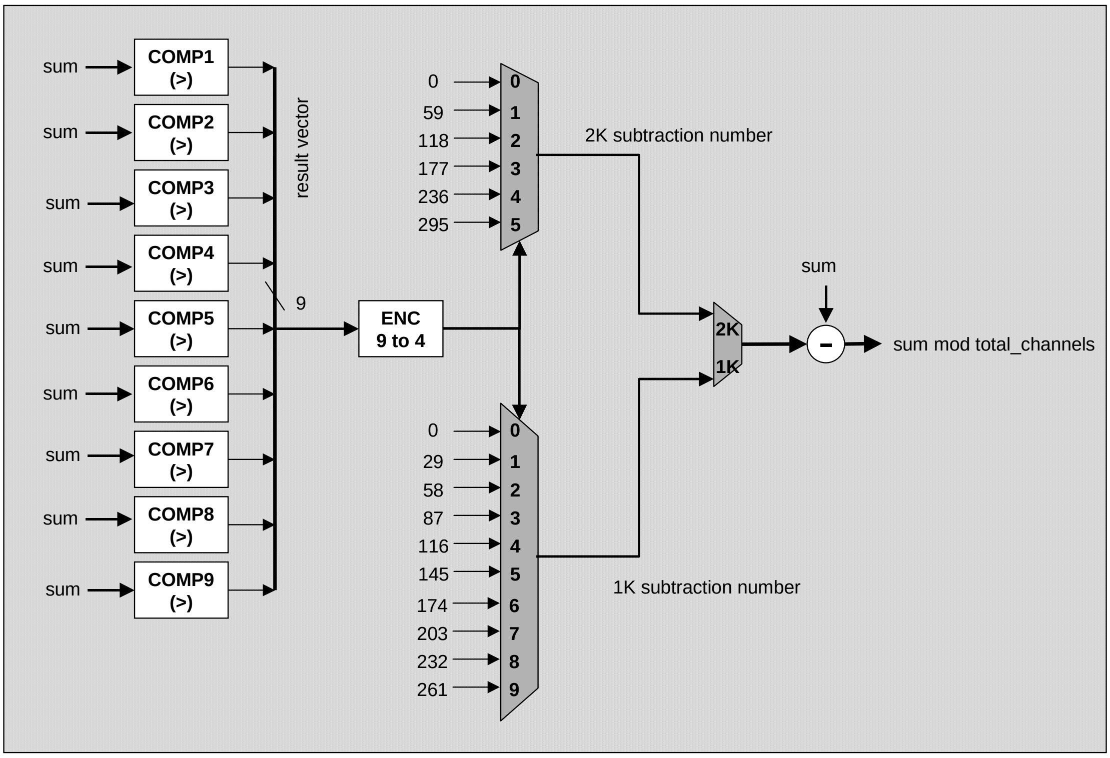

Figure 45 Datapath of modulo total_channels

For the purpose of the pointer generation and memory accesses, a FSM is used, whose state diagram is illustrated in Figure 46. The Pointer Storage FSM requests the new Sub-Channel Numbers and controls the write access to PMEM, storing the calculated pointers in the addresses provided by the allocated carriers' values. Since the data storage organization follows a well-specified structure, the pointers to be stored can be easily calculated through internally kept and increased counters, without any information exchange with the Data Storage FSM.

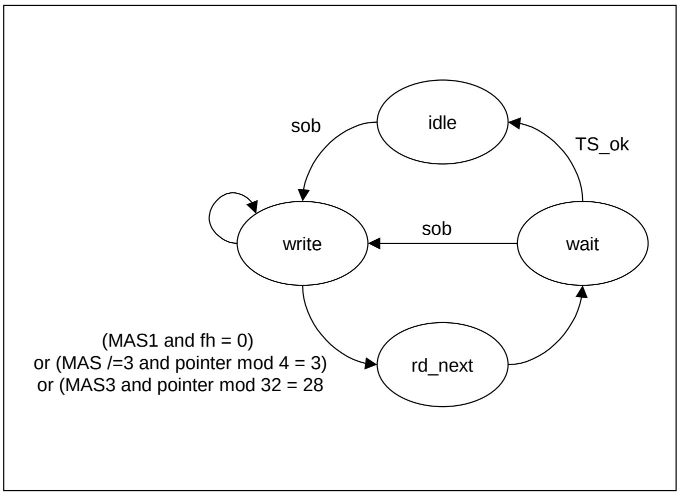

Figure 46 Pointer Storage FSM

The main idea of the FSM is quite simple. While in idle state, no memory access is performed and the unit waits for a new Time Slot's data to arrive. When the start of a burst is detected (sob = 1), there is a transition to the write state. While in write, PMEM write access is enabled (pmem_wr_en = 1) as well as the pointer generation. When the storage of the sub-channel's pointers is completed, we move to rd_next state. In rd_next, the FSM activates the de-queuing of a Sub-Channel Number from the FIFO (scn_req = 1). In the next cycle the FSM is set in a wait state. We remain in that state until a new burst arrives and the process is repeated, unless the Time Slot has been completed (TS_ok = 1) which means that we will pass to idle state.

The written addresses are provided by the carrier allocation circuit, which operates in parallel with the FSM. Both the FSM and the carrier allocation are triggered with the assertion of sob signal. Since the FIFO that keeps the Sub-Channel Numbers is "show-ahead" and the Time Slot configuration is known before the data reception starts, the first carrier of a sub-channel is calculated long before a burst starts. Then after sob arrives the circuit provides one carrier per cycle. That permits the carrier calculation to be synchronized with Pointer Storage FSM.

As far as it concerns the stored pointer, it is a counter that is enabled while in write state and increases according to the configuration. For the first two Medium Access

Schemes it augments by one for each calculated carrier. In case of the third Medium Access Scheme, the same thing happens with the following difference. The stored pointer value is the counter value. When a channel's allocation is completed the 5 bits of lower significance are reset and the 6 most significant ones stand as a separate counter of the incoming bursts and their value is increased by one.

The pointer value is used along with the parameters of Medium Access Scheme and frequency hopping to define the transition conditions from write to rd_next state. In other words this conditions indicate when the carrier information storage of a burst has been completed. In MAS1 without frequency hopping the sub-channel consists of one carrier, therefore only one write access to PMEM is required for each burst. In the frequency hopping case or when the second Medium Access Scheme is applied, for carriers are allocated. Consequently, the next state will be rd_next when modulo 4 of the pointer is 3. With the same logic, when in MAS3, we check for modulo 32 to equal 28.

The following table lists all the actions taken in each state, and the required conditions for a transition, whereas an example of the signals' timing is illustrated in .

STATE	**TRANSITION**	**DESCRIPTION**
idle	<u>write:</u> sob = 1	pmem_wr_en, pmem_addr, pmem_di, scn_req, pointer = 0
write	<u>rd_next:</u> (MAS1 and frequency hopping = 0) or (MAS/=3 and pointer mod 4 = 3) or (MAS3 and pointer modulo 32 = 28)	pmem_wr_en = 1 pmem_addr = calculated carrier pmem_di = 10 & pointer & 00 when MAS1 and no frequency hopping 10 & pointer when MAS/=3 pointer + 1 when MAS3 scn_req = 0 pointer + 1
rd_next	<u>wait:</u> always	pmem_wr_en = 1 pmem_addr, pmem_di unused scn_req = 1 pointer[10:5] + 1 when MAS3 pointer[4:0] = 0 when MAS3
wait	<u>idle:</u> TS_ok = 1 <u>write:</u> sob = 1	pmem_wr_en = 0 pmem_addr, pmem_di unused scn_req = 0 pointer unchanged

Table 19 Pointer storage FSM actions

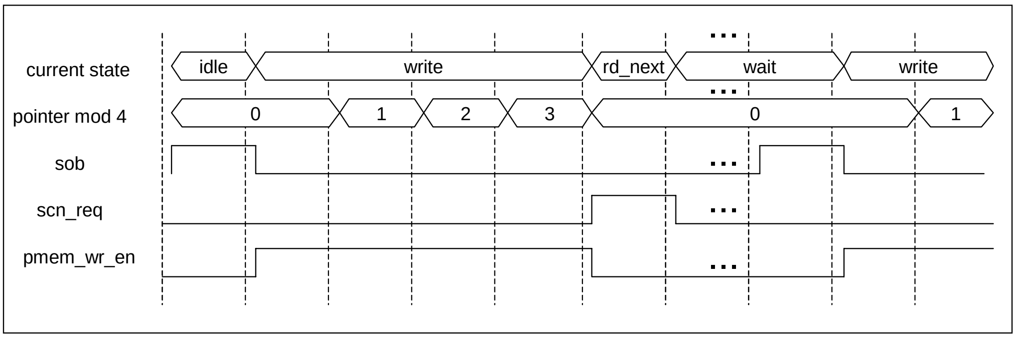

Figure 47 General timing of signals in MAS1 with frequency hopping or MAS2

6.4 Double Buffer Mechanism

The double buffering mechanism keeps the ownership status of each buffer. In other words it controls which buffer is the target of the write accesses and which of the read accesses by routing the access signals from each controller to the correct buffer. The Controllers on the other hand do not know which buffer they access. For the above purposes, the block, uses the external interface listed in the next table. The width of each signal is one bit.

Each buffer contains the information of one Time Slot. Consequently it is not permitted that the same Controller accesses both buffers. Expressed otherwise, each buffer is accessed by only one Controller at a given time. The Output Controller must transmit the Time Slots contiguously. That means, that when a Time Slot's transmission is completed, a buffer must be available and loaded with the data of the next Time Slot. Although the reading of a Time Slot takes more time than the storage, there is a sufficient delay among a Time Slot's data. Therefore, the input data rate guarantees that the Input Controller always has an available buffer when a new Time Slot arrives and also a Time Slot's storage will be completed before the output stage finishes the issuing of the previous Time Slot. That way the double buffering is well synchronized and the Output Controller succeeds to issue a contiguous flow of data.

After the system start-up both buffers are empty. That means that there is no available buffer for the Output Controller and two available buffers for IDC. Only one will be selected and we consider that BUF0 will be the first accessed buffer. For that reason, there is no need to flag the buffer availability to the Input Controller. The beginning of the Time Slot is flagged at the input and this is sufficient to inform the input stage to start writing. On the other hand, at system start-up, there is no available buffer to the Output Controller, and the sub-block has no means of knowing when to start the transmission. For that reason as well as for the better synchronization of the two stages the MAC Layer issues a pulse that indicates to the Output Controller that a Time Slot must start. The timing diagram of Figure 48 illustrates the toggling of the access operations between the two buffers.

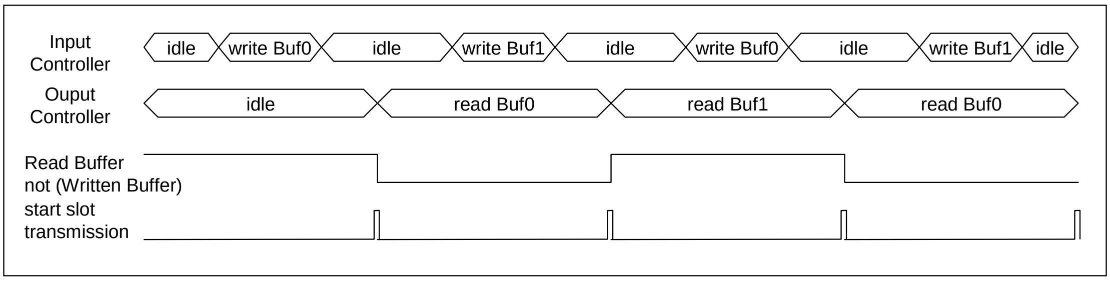

Figure 48 Double buffering mechanism timing

6.5 Output Controller Implementation

The Output Controller is responsible for the issuing in columns of the matrix to be transmitted. For that purpose it read accesses the memories in the buffers where the Input Controller has stored the data symbols and the Time Slot information. Two main counters are kept internally to point the position of the matrix that must be issued. Then based on the stored information it decides the type of the position's contents and if it is not empty (null), it retrieves the appropriate symbol, either from DMEM, either from one of the pilot generating blocks. Before the final issuing, the symbol is pre-mapped according to Table 14, presented in the previous chapter. The final pre-mapped output is kept latched for as many cycled required in order to succeed the rate required by the configured transmission mode. Each time a symbol/point is issued the line counter of the matrix is increased. When a column has been completed, there is a pause whose length depends on the guard interval size and during which null symbols are issued.

The main controls that keep in order the read accesses and the type decisions are succeeded basically with two Finite State Machines. The first one keeps track of the matrix's area type that is currently issued (Null, Data or Ranging Area) and we will call it

Area Control FSM. A second one is used for the actual output symbol generation and it called Output FSM. The two FSMs exchange information. More specifically the Area Control FSM flags to the Output FSM the area type, and the Output FSM generates a pulse each time a symbol is issued. The following paragraphs are devoted to a more detailed description of the FSMs concept and structure.

6.5.1 Area Control FSM

The Area Control FSM (Figure 49) keeps track of the matrix transmission and defines at any given moment which area is issued. The different areas distribution has already been introduced in the previous chapter of the FRM Unit's functional analysis. There are 3 main types of areas: Null, Data and Ranging. Depending on the type we can define the nature of their contents and thus from where they can be retrieved, an information that will be exploited by the Output FSM. The position and range of the various areas is being depicted for different system configurations in Figure 21, Figure 22, Figure 23, Figure 27 and Figure 29.

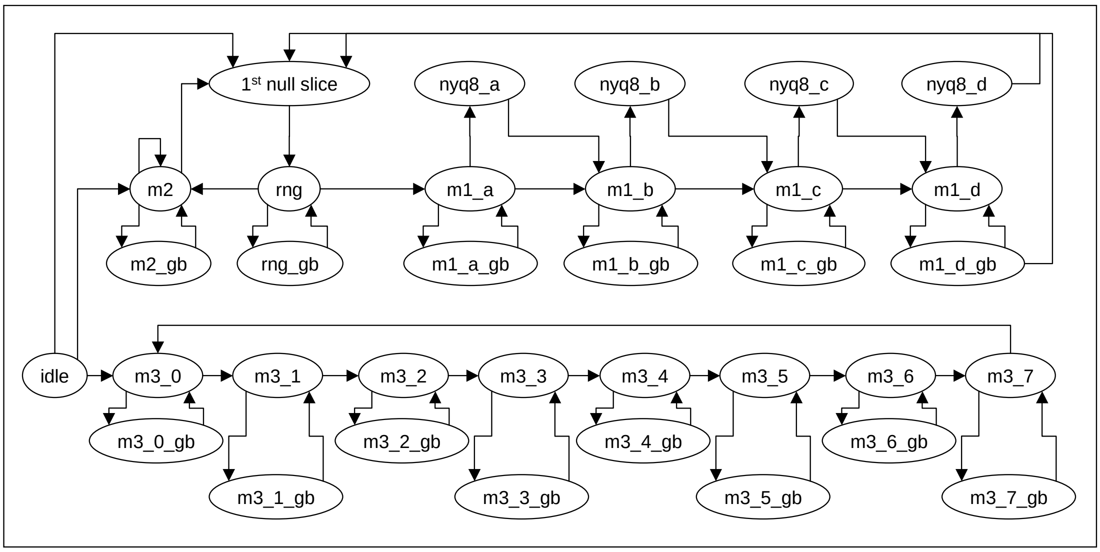

Figure 49 Area Control FSM

When the system is initiated, the FSM's status is idle. The Output Controller cannot operate until a first Time Slot is stored and a start pulse is received. Then after reading the configuration parameters of the Time Slot from PMEM, the FSM moves to an active state. From that point, we will transit only to active states according to the transmitted area and the FSM will return to idle state only when the system is reset.

The transitions among the active states can be easily extracted from the analysis made so far. More specifically, if we consider the fact that the transmitted matrix is issued in columns and that the issuing starts from the middle of the column and also consult the figures that illustrate the different matrix organizations, the states and the flow of the FSM are easily deduced. It is understandable that the conditions for the state transitions depend on the values of the matrix's column and line counters.

In Medium Access Scheme 1 the first column is always filled with null symbols (1^{st} null slice state). The next area in time is dedicated to the Ranging transmission (rng state) and then follows the transmission of the Data area, which is divided, to four sub-areas, one for each mini-burst (m1_a, m1_b, m1_c, m1_d states). In case of Nyquist Shaping and frequency hopping the data area is interrupted by sets of eight null columns after each mini-burst (nyq8_a, nyq8_b, nyq8_c, nyq8_d states). On the other hand if no frequency hopping is applied, eight null columns are added only after the last mini-burst (nyq8_d state after m1_d state). After all the transmission of the matrix, we move back to 1^{st} null slice state and the process is repeated for the next Time Slot matrix.

In Medium Access Scheme 2, if the transmitted Slot is the first one of a Transmission Frame, we will start again with a column of null symbols (1^{st} null slice state) followed by the Ranging transmission (rng state). Then the Data area is represented by m2 state. If the transmitted matrix does not correspond to the head Time Slot, it consists only of a Data area (m2 state). Again, when a Time Slot is completely issued the FSM will start from 1^{st} null slice state or m2 state depending on the next Time Slot's position inside the frame.

In Medium Access Scheme 3, Data and Ranging areas are merged. Since the entire Ranging transmission is spread in eight contiguous Time Slots that consist a Transmission Frame, the FSM keeps different states for each one of them (m3_0, m3_1, m3_4, m3_5, m3_6, m3_7 states). Each time a matrix is issued we move from state m3_n to state m3_n+1.

It is known that the Data Ranging areas occupy almost the entire height of the matrix apart from two guard band zones at the top and bottom lines that are null areas. For that reason each state is accompanied by a corresponding _gb state (ie m2 state is paired with m2_gb state). As the symbols of the top carriers (lower lines) are issued first, we start from a regular state then move to the *gb* state, in order to issue the guard band points and finally move back again to the regular state. The states that have already been corresponded to null areas as 1st null slice and nyq8_x states do not have matching _gb states.

6.5.2 Output FSM

The Output FSM, whose state diagram appears in Figure 50, is responsible of the actual symbol decision, retrieval, pre-mapping and issuing. More specifically depending on the status of the Area Control FSM, Output FSM has knowledge of the current area's identity. That way it can limit the number of candidate symbols types. So, when a guard band is transmitted, it is known that only null symbols are issued, whereas in Ranging area there are null symbols or ranging pilots, thus there is no question whether a data symbol must be retrieved.

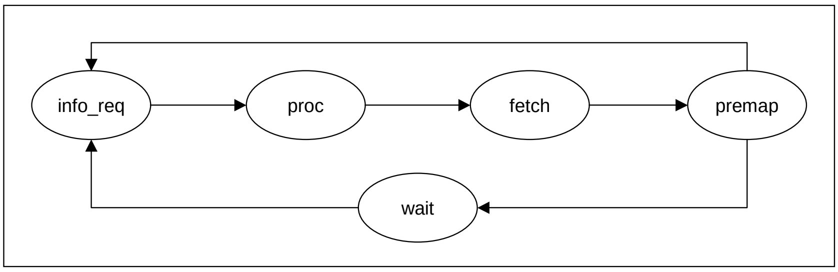

Figure 50 Output FSM

The first state is info_req, during which the Output Controller requests the information related to the point of the currently processed carrier. If the area type is Data then we read the PMEM address corresponding to that carrier. Respectively in case of Ranging area RMEM is read, whereas in Medium Access Scheme 3 both memories are read.

In the next cycle, the current FSM state is proc. During that state, the read words arrive and the contained information is processed. As it has already been explained in the presentation of PMEM and RMEM, the basic element deduced is whether the carrier is used for transmission. Apart from the validity of the carrier, a PMEM word contains a pointer that serves the retrieval of a data symbol from DMEM. Therefore, the output stage decides whether the current symbol is null or valid. If it is valid it resolves the type question based on the column and line counters and the pilot patterns and prepares the corresponding request.

The next transition is to fetch state. In that state, we read the DMEM address calculated in the previous stage or request a pilot from one of the Pilot generating blocks. The final state is premap, during which the requested symbol is received and pre-mapped according to its type. The pre-mapped point is latched until the next one is issued.

Apart from the main states described above, there is a supplementary wait state. During that state the Controller is idling, keeping latched the last output, in order to succeed the desired output rate defined by the configuration. The fastest mode is one symbol every four cycles and in that case the wait state is redundant and is bypassed. Other modes provide the option of one symbol every 8, 16 or 32 cycles and then the FSM must stay in wait state for a respective number of cycles minus 4.

Figure 51 illustrates the timing of the basic operations. The depicted example concerns the issuing of a data and then a pilot symbol, in case of the two first Medium Access Schemes for the fastest output rate (wait state skipped).

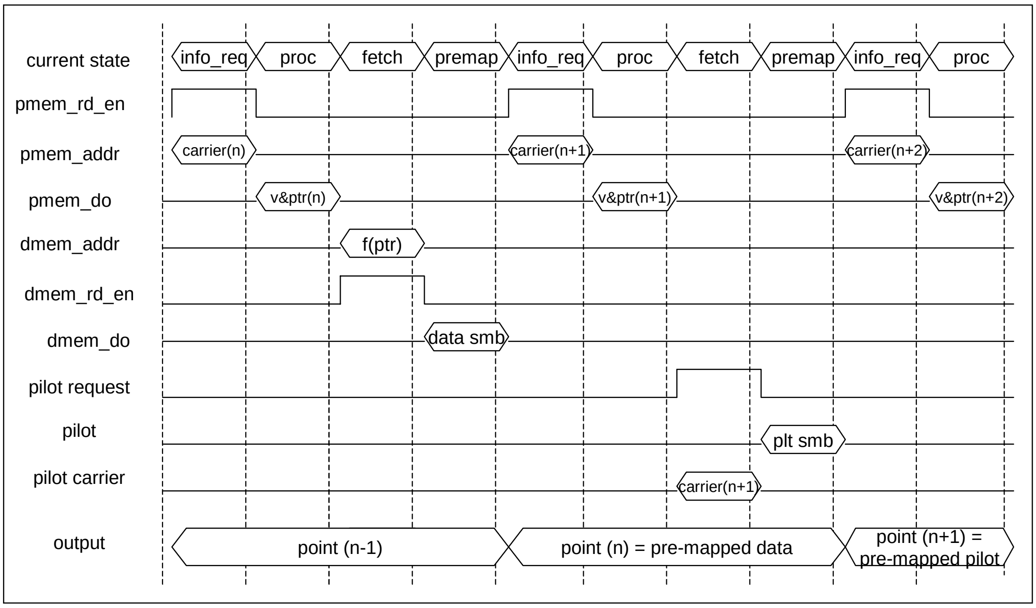

Figure 51 Timing example of a data and pilot issuing in case of MAS1 or MAS2

Let's examine in more detail the counters and pointers used in the processing state for the calculation of DMEM address and various other conditions. The basic pair of counters is always the matrix line and column counter. Each time a symbol is issued the matrix's line counter is increased, and when an entire column has been transmitted it is reset and column counter augments. Both counters are reset when the transmission of a new Time Slot starts. Apart from the line counter there is also a similar carrier counter, which ranges from 0 to 1711 (usable carriers numbering) and increases only during when PMEM or RMEM access is involved, since it is used for the addressing of these two memories. In addition the carrier value must be provided along with a pilot request in order for the Pilot Generation block to extract the correct symbol value from its internal ROM.

The position of the symbol in the matrix and the pilot patterns play the most important role to the exact calculation of DMEM address based on the offset provided inside the read PMEM word. In Medium Access Scheme 3, the pointer provides information about the BS3 line the current symbol belongs to. Along with the column counter we can detect the exact position of the point in the BS3 sub-matrix and thus deduce whether it is data or pilot. If it is data then the position is translated to a DMEM address resulting from the concatenation of the burst's increasing number (provided by the read pointer), the line number (contained in the 5 LSBs of the read pointer) and the 3 LSBs of the column counter. The alignment that has been followed during the data storage helps retrieve the desired data symbol with the simple concatenation of the counters and pointers.

For the first two Medium Access Schemes the followed process is different. The modulo 4 of the carrier value indicates the pilot pattern and along with the column counter we can draw the conclusion whether the current point is data or pilot. On the other hand there is a complexity involved in the retrieval of a data symbol from DMEM. The complexity occurs from the fact that the n^{th} stored data symbol of a burst is not placed on the n^{th} slice of the Data area, nor is necessarily column aligned with the n^{th} stored data symbol of another burst. On the other hand, since the pilot patters of a mini-burst depends on the value of the carrier over which it is transmitted (mod 4) we can observe that the n^{th} stored data symbols of bursts using carriers with the same modulo 4 are column aligned. Hence we use four data counters, one for each carrier modulo 4, which show how many data symbols of a burst have been read so far. When a column is completed the data counters related to a pilot pattern indicating a pilot for the specific column, are not increased, whereas the rest are. That way the DMEM address of a requested data symbol can be calculated by 36 times the read from PMEM pointer increased by the data counter of the corresponding carrier's modulo 4. In case of Medium Access Scheme 1 without frequency hopping the read pointer is multiplied 36 x mini-burst number.

6.6 Design Flow and Implementation Results

A sequence of steps was followed for the completion and delivery of the design. Starting from the study and comprehension of the DVB-RCT standard, a functional and design description was composed. Then the design was described in a hardware description language. The next step was the debugging and verification of the block. The last step before the final delivery was the synthesis of the code. During this process the design was revisited and updated, according to the required modifications and corrections.

The first action was the functional specification of the block. The I/O interface was precisely defined and then the main algorithms were conceived. When the functionality of the block was finalized, the design specification of the block could be described in detail. The partitioning of the block was decided, as well as the precise interface between the sub-blocks. Then the implementation of each sub-block was designed to the datapath level, using basic modules as gates, multiplexers, adders or Finite State Machines.

Following the design specification, the block was described to the Register Transfer Level in VHDL. The verification of the block was performed individually as well as each time that a stage was written and added to the DVB-RCT chain. First FRM was connected to the previous stages of the chain. The system was configured as described by the scheduled verification plan. When all the tests passed, the next stage was attached. That way FRM was tested as an output and as an intermediate stage.

For the implementation and verification of the design a set of tools was used. Cadence tools (ncvhdl, ncelab, ncsim) were used for the compilation and simulation of the VHDL code. The test benches were generated using a company propriety tool, whereas a configurable C model of the system generated the output reference patterns. Synthesis sessions were launched using Synopsys design analyzer and a target library of 18µ. That way the synthesizability of the code was checked and a first estimation about the feasibility of the target frequency.

The chain was developed having in mind the future production of an ASIC. Nevertheless the implementation target was the chain to be ported in a FPGA board for the purposes of the system's demonstration. The synthesis was performed with Synplify. The used FPGA characteristics were:

- Technology: Xilinx Virtex 2
- Part: XC2V6000
- Speed: -4
- Package: FF1152

It must be noted that the ported implementation is a different design version than the one presented in the current document. The delivered design was based in a first analysis of the Frame Adaptation Unit. This analysis wasn't complete due to deadline constraints. The functionality and implementation of the Frame Adaptation Unit was reviewed and completed resulting in the design described in the above chapters, and it is characterized by lower cost, reduced complexity and "cleaner structure".

7 Contribution and Future Work

The present thesis has highlighted the basic DVB-RCT Physical Layer's features and characteristics. Furthermore it has provided a design specification for the Frame Adaptation Unit. Then design has been described in a hardware description language (VHDL) and verified. The description of the system in VHDL is flexible providing the ability to integrate the same design in a different FPGA platform or be used for the future manufacturing of an integrated circuit.

DVB-RCT is a standard aiming to provide terrestrial broadcasters with a flexible wireless return channel for interactive DVB-T Services. The acceptance f DVB-T is an important argument in favor of this new standard's adoption. Due to the compatibility of the standard, DVB-RCT is constituted the most appropriate solution for the creation of a return path supporting applications of interactive digital terrestrial TV.

An important characteristic of DVB-RCT is the fact that it combines a set of technical elements that are considered to be the basis of future broadband transmission systems in general. Such elements are the Orthogonal Frequency Division Multiplexing Multiple Access technique (OFDMA), the Adaptive Modulation feature and others. For that reason, any work and study related to DVB-RCT technology is a promising basis for the development of future high performances for broadband wireless access and advanced mobile systems.

The creation of the DVB-RCT standard isn't just a theoretical study but reflects actual commercial needs for interactivity. Thus, the partners of the DVB-RCT forum have decided the availability of an efficient low cost silicon solution. The design and manufacture of low cost integrated interactive terminals for digital terrestrial TV based on DVB-RCT technology using a dedicated silicon solution is the final target

The first steps have been taken to make the DVB-RCT from a theoretical standard to a commercial reality. The initial phase dictated the need for a prototype aiming to test critical parts of the DVB-RCT specification. The work presented in this thesis is part of this effort. The current status of this design is that it has been successfully implemented and tested. It has been integrated in an FPGA prototype version that demonstrated real time back-to-back transmission and reception. These results provided the green light to continue the work on making DVB-RCT a reality.

Of course an important amount of work remains to be done. One of the main pending issues is the definition of regulation policies regarding the use of UHF/VHF bands for Interactive DTV services. DVB-RCT was created in order to permit the evolution of digital television with the addition of new interactive services. These services must be created providing challenging and attractive applications to the television users.

As far as it concerns the hardware design of the Frame Adaptation Unit it could use further investigation. The currently implemented Physical Layer supports almost all the parameters described in the DVB-RCT standard. Some configurations of secondary importance have been judged to insert an undesirable complexity and cost in relation to their advantages. Such cases are Medium Access Scheme 2 with Nyquist Shaping or Transmission Frame 2 containing bursts of the second type. Another example is the capacity of Transmission Frame 1. In out implementation we have considered that the maximum number of transmitted bursts per Time Slot is 128. In reality this number depends on the modulation and 128 is the minimum permitted value whereas for QPSK it can reach 384 bursts per slot. In the future we could work on the expansion of the design to cover these additional options.

Even if work remains, from now, the DVB-RCT constitutes a real asset for a successful deployment of the interactive terrestrial digital TV worldwide. The encouraging thing is that the effort that has been dedicated so far gave positive results. Therefore, DVB-RCT is moving forward, from theory to reality.

8 References

[1] DVB Project, ETSI: "Digital Broadcasting System for Television, Sound and Data Services; Framing Structure, Channel Coding and Modulation for Digital Terrestrial Television" - ETS 300 744 DVB-T standard, 1996

[2] DVB Project, "Digital Video Broadcasting (DVB); Interaction channel for Digital Terrestrial Television (DVB-RCT) incorporating Multiple Access OFDM", DVB-TM document TM2361r3, March 2001 and ETSI, EN 301 958 DVB-RCT standard, 2001

[3] "Communication Systems Engineering" John G. Proakis and Masoud Salehi, Prentice-Hall Inc, 2nd edition

[4] "An introduction to orthogonal frequency-division multiplexing", Edfors et al, Technical report Luleå University of Technology, 1996

[5] "OFDM as a possible modulation technique for multimedia applications in the range of mm waves", Dušan Matiæ, 10/30/98/TUD-TVS, Introduction to OFDM, 2nd edition

[6] "OFDM transmission and interfering terms in case of CIR longer than the prefix", J.Dagres, T.Vaidis, A.Polydoros, P.Tsakalides, 2001, IST-1999-10025.

[7] "802.11 Wireless Networks, The Definitive Guide", Mathew S. Gast, O'Reilly editions, April 2002

[8] " Signals and Systems", Simon Haykin and Barry Van Veen, John Wiley & Sons Inc, 1999

[9] "Digital Design", M. Morris Mano, Prentice-Hall Inc, 2nd edition