

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ

**Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων**



Πτυχιακή Εργασία

“ Σχεδίαση, υλοποίηση και αξιολόγηση απόδοσης ενός δικτυακού περιβάλλοντος σύγκλισης IP/DVB για την παροχή υπηρεσιών triple-play σε αστικές και απομακρυσμένες περιοχές ”

**ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΜΠΟΥΡΔΕΝΑ ΑΘΗΝΑ
ΗΜΕΡΟΜΗΝΙΑ: 25/08/2007**

ΕΙΣΗΓΗΤΗΣ: ΠΑΛΛΗΣ ΕΥΑΓΓΕΛΟΣ

Σε αυτούς που στάθηκαν δίπλα μου,
με ιδιαίτερη εκτίμηση και αγάπη

Ευχαριστίες

Με την ολοκλήρωση της πτυχιακής μου εργασίας, η οποία υλοποιήθηκε στο Εργαστήριο Τηλεπικοινωνιών και Δικτύων (ΠΑΣΙΦΑΗ) του Τ.Ε.Ι Κρήτης, θα ήθελα να ευχαριστήσω τους ανθρώπους οι οποίοι βοήθησαν στην περάτωση αυτής της εργασίας. Θα ήταν παράλειψη να μην αναφερθώ σε όλους εκείνους που συμπαραστάθηκαν σε αυτήν την προσπάθεια.

Κατά κύριο λόγο, θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου από το Τ.Ε.Ι Κρήτης κ. Πάλλη Ευάγγελο, ο οποίος με υποστήριξε καθ'όλη τη διάρκεια της πτυχιακής εργασίας και μου εξασφάλισε την παροχή πλούσιας υλικοτεχνικής υποδομής, πολύτιμης για μια μελέτη όπως αυτή.

Θα ήθελα να εκφράσω την εκτίμηση μου προς τον υποψήφιο διδάκτορα, κ. Μαστοράκη Γεώργιο, με τον οποίο με χαρά συνεργάστηκα και να τον ευχαριστήσω για την καθοδήγηση του προς την επιτυχή ολοκλήρωση της πτυχιακής μου εργασίας.

Ιδιαίτερες ευχαριστίες οφείλονται επίσης και στους εργαστηριακούς συνεργάτες του Εργαστηρίου ΠΑΣΙΦΑΗ, κ. Σιδέρη Ανάργυρο και κ. Μαρκάκη Ευάγγελο, για την ανεκτίμητη βοήθεια και την απεριόριστη ψυχολογική υποστήριξη και κατανόηση που μου παρείχαν όποτε αυτή χρειάστηκε.

Ηράκλειο, Αύγουστος 2007

Μπουρδένα Αθηνά

Περίληψη

Η διπλωματική εργασία αυτή προτείνει την ιδέα για την υλοποίηση ενός δικτυακού περιβάλλοντος σύγκλισης IP/DVB το οποίο δίνει τη δυνατότητα παροχής υπηρεσιών triple-play (υπηρεσίες βίντεο, υπηρεσίες φωνής, υπηρεσίες δεδομένων) 24ωρης σύνδεσης σε αστικές και απομακρυσμένες περιοχές. Ειδικότερα, με το δικτυακό αυτό περιβάλλον σύγκλισης των τεχνολογιών τηλεπικοινωνιών και των δικτύων ευρείας κάλυψης (broadcasting) πραγματοποιείται μία ενοποιημένη υποδομή η οποία επιτρέπει στους παραδοσιακά παθητικούς αστικούς τελικούς χρήστες να γίνουν ενεργοί, συμμετέχοντας στην Κοινωνία της Πληροφορίας δημιουργώντας, επεξεργάζοντας και παρέχοντας το δικό τους περιεχόμενο/υπηρεσίες με τη χρήση μίας κοινά αξιοποιήσιμης υποδομής. Επιπλέον το δικτυακό περιβάλλον αυτό επιτρέπει την παροχή υπηρεσιών triple/play σε απομακρυσμένες περιοχές στις οποίες δεν υπάρχει σύνδεση του τοπικού βρόχου με τα κεντρικά δίκτυα κορμού (core backbone). Έτσι με μία ενοποιημένη πλατφόρμα IP/DVB επιτυγχάνεται η συνέργια των νέων τεχνολογιών ασύρματων και ενσύρματων δικτύων τηλεπικοινωνιών καθώς και αυτών της ευρείας κάλυψης για τη γεφύρωση του ψηφιακού χάσματος (digital divide) το οποίο υπάρχει μέχρι σήμερα όχι μόνο μεταξύ διαφορετικών κρατών αλλά και σε πολλές περιοχές των τεχνολογικά και οικονομικά ανεπτυγμένων χωρών.

Στόχος της διπλωματικής αυτής εργασίας αποτελεί η εξέταση των κατάλληλων αρχιτεκτονικών για τη δημιουργία ευρυζωνικών μητροπολιτικών δικτύων, οι οποίες υλοποιούν τη ροή μεταφοράς της επίγειας ψηφιακής τηλεόρασης ως κορμό διασύνδεσης ενδιάμεσων κόμβων διανομής, μέσω των οποίων επιτυγχάνεται η παροχή και λήψη διαδραστικών πολυμεσικών υπηρεσιών κάνοντας χρήση του πρωτοκόλλου TCP/IP. Δεδομένου ότι ένα ασύμμετρο δίκτυο παρουσιάζει διαφορετική συμπεριφορά ανάλογα με το χρησιμοποιούμενο uplink, η μελέτη εστιάζεται στη χρήση τεχνολογιών μεγάλου ρυθμού διαμεταγωγής στο δίκτυο πρόσβασης (μεταξύ κόμβου και τελικών χρηστών), έτσι ώστε να είναι δυνατή η παροχή ευρυζωνικών υπηρεσιών από τους τελικούς χρήστες. Η αξιολόγηση της απόδοσης του δικτύου θα γίνει με αντικειμενικά μέτρα (Throughput, RTT Delay, Jitter, One-way Delay) κάτω από διαφορετικές συνθήκες φόρτισης του δικτύου, αλλά και με διαφορετικές τεχνολογίες καναλιού επιστροφής. Συγκεκριμένα, στο δίκτυο πρόσβασης (access network) θα εξετασθούν ως πιθανά σενάρια η ψηφιακή (ADSL) γραμμή και ασύρματα δίκτυα δεδομένων (Wireless LAN) IEEE 802.11g τα οποία παρέχουν μεγάλες ταχύτητες διαμεταγωγής.

Περιεχόμενα

1. ΕΙΣΑΓΩΓΗ	12
1.1. Γενική Περιγραφή	12
1.2. Στόχοι	12
1.3. Δομή	13
1.4. Θεωρητικό Υπόβαθρο για την Τεχνολογία Digital Video Broadcasting – Terrestrial (DVB-T)	14
1.5. Σύνοψη	14
2. ΨΗΦΙΑΚΗ ΤΗΛΕΟΡΑΣΗ ΚΑΙ ΥΠΗΡΕΣΙΕΣ ΔΕΔΟΜΕΝΩΝ	15
2.1. Εισαγωγή	15
2.2. Σήμα Βασικής Ζώνης – Ροή Μεταφοράς MPEG-2	16
2.3.Εισαγωγή στα Πρότυπα DVB	21
2.4. Γενικές Αρχές Επίγειας Ψηφιακής Μετάδοσης Ευρείας Κάλυψης (DVB-T)	22
2.4.1 Γενικά	23
2.4.1.1. Προσαρμογή ροής Μεταφοράς και Τυχαιοποίηση για Διασπορά Ενέργειας	26
2.4.1.2. Εξωτερική Κωδικοποίηση και Διεμπλοκή	27
2.4.1.3. Εσωτερική Κωδικοποίηση και Διεμπλοκή	29
2.4.2. Ωφέλιμο Bit Rate	30
2.5. Πρωτόκολλα Μεταφοράς Δεδομένων	31
2.5.1. Το Πρωτόκολλο TCP και η Σημασία του	31
2.5.2. Το Πρωτόκολλο UDP και η Σημασία του	33
3. ΠΑΡΟΥΣΙΑΣΗ ΔΙΚΤΥΟΥ ΠΑΡΟΧΗΣ ΑΜΦΙΔΡΟΜΩΝ ΔΙΑΔΡΑΣΤΙΚΩΝ ΥΠΗΡΕΣΙΩΝ	35
3.1. Εισαγωγή	35
3.2. Τοπολογία Πειραματικού Δικτύου Επίγειας Ψηφιακής Τηλεόρασης	35
3.3. Χαρακτηριστικά Ηλεκτρονικών Υπολογιστών που Χρησιμοποιήθηκαν για την Υλοποίηση του Δικτύου	40
3.4. Ενδιάμεσος Κόμβος Διανομής (Cell Main Node)	47
3.4.1. Απαιτήσεις Υλικού (Hardware Requirements)	49
3.4.2. Απαιτήσεις Λογισμικού (Software Requirements)	52
3.5. Η Συμπεριφορά του CMN σε Δικτυακή Κίνηση	54
3.5.1. Μετάδοση Δεδομένων με την Αξιοποίηση Κίνησης TCP/IP και UDP	54
4. ΣΤΟΙΧΕΙΟΘΕΤΙΣΗ ΚΑΙ ΣΧΟΛΙΑΣΜΟΣ ΠΕΙΡΑΜΑΤΙΚΩΝ ΜΕΤΡΗΣΕΩΝ	63
4.1. Εισαγωγή	63
4.2. Παρουσίαση Αποτελεσμάτων και Γραφικών Παραστάσεων Δικτυακής Κίνησης TCP	64
4.2.1. Γενικά	64
4.2.2. Μεμονωμένες μετρήσεις	64
4.2.2.1. Αξιολόγηση Δικτυακής Κίνησης από τον τελικό Χρήστη προς τον Εξυπηρετητή Δεδομένων	64
4.2.2.2. Αξιολόγηση Δικτυακής Κίνησης από τον τελικό Χρήστη προς την Αναγεννητική Πλατφόρμα	67
4.2.2.3. Αξιολόγηση Δικτυακής Κίνησης από τον τελικό Χρήστη προς τον Ενδιάμεσο Κόμβο Διανομής 2	69
4.2.2.4. Αξιολόγηση Δικτυακής Κίνησης από τον Ενδιάμεσο Κόμβο Διανομής των Τελικών Χρηστών προς την Αναγεννητική Πλατφόρμα	72
4.2.2.5. Αξιολόγηση Δικτυακής Κίνησης από τον Εξυπηρετητή Δεδομένων προς την Αναγεννητική Πλατφόρμα	74
4.2.2.6. Αξιολόγηση Δικτυακής Κίνησης από τον Εξυπηρετητή Δεδομένων προς τον Ενδιάμεσο Κόμβο Διανομής 1	76

4.2.2.7. Αξιολόγηση Δικτυακής Κίνησης από τον Ενδιάμεσο Κόμβο Διανομής του Εξυπηρετητή Δεδομένων προς την Αναγεννητική Πλατφόρμα	78
4.2.3. Αξιολόγηση Δικτυακής Κίνησης Μεταβαλλόντας το Window Size	80
4.2.3.1. Αξιολόγηση Δικτυακής Κίνησης με Window Size 16k	84
4.2.3.2. Αξιολόγηση Δικτυακής Κίνησης με Window Size 64k	86
4.2.3.3. Αξιολόγηση Δικτυακής Κίνησης με Window Size 128k	88
4.2.3.4. Αξιολόγηση Δικτυακής Κίνησης με Window Size 512k	90
4.2.4. Αξιολόγηση Δικτυακής Κίνησης με Δύο Ταυτόχρονους Χρήστες	92
4.2.5. Αξιολόγηση Δικτυακής Κίνησης Μεταβαλλόντας το Μέγεθος του ICMP Packet	94
4.2.6. Αξιολόγηση Δικτυακής Κίνησης Μεταβαλλόντας το Bit Rate στον Amber	96
4.3. Παρουσίαση Αποτελεσμάτων και Γραφικών Παραστάσεων Δικτυακής Κίνησης UDP	98
5. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΠΡΟΤΑΣΕΙΣ	100
Παράρτημα Α	101
Α.1. Προγράμματα Δημιουργίας και Ανάλυσης της Κίνησης ενός Δικτύου	101
Α.1.1. Δημιουργία Κίνησης	101
Α.1.1.1. Πρόγραμμα Iperf	101
Α.1.1.2. Πρόγραμμα Multi-Generator (MGEN)	101
Α.2.1. Ανάλυση Κίνησης	102
Α.2.1.1. Πρόγραμμα Tcpdump	102
Α.2.1.2. Πρόγραμμα Tcptrace	102
Α.2.1.3. Πρόγραμμα Xp12gpl	103
Α.2.1.4. Προγράμματα της Γλώσσας Προγραμματισμού Perl	103
Α.2. Παράδειγμα Δημιουργίας και Ανάλυσης της Κίνησης ενός Δικτύου	104
Α.2.1. Δημιουργία Δικτυακής Κίνησης TCP	104
Α.2.1.1. Ανάλυση Δικτυακής Κίνησης TCP	106
Α.2.2. Δημιουργία Δικτυακής Κίνησης UDP	109
Α.2.2.1. Ανάλυση Δικτυακής Κίνησης UDP	112
Παράρτημα Β	114
Β.1. SCRIPT REPLICID THE PERL	119
Β.2. SCRIPT CREATEFILES THE PERL	123
Β.3. SCRIPT LOSSES THE PERL	128
Β.4. SCRIPT THROUGHPUT THE PERL	130
Β.5. SCRIPT JITTER THE PERL	132
Β.6. SCRIPT TIMESTAMP THE PERL	134
Β.7. SCRIPT TIMESTAMP ONE WAY DELAY THE PERL	136
Β.8. SCRIPT SMOOTHED JITTER (RFC 3550) THE PERL	142
Β.9. SCRIPT JITTER THE PERL	
ΒΙΒΛΙΟΓΡΑΦΙΑ	144

Πίνακας Σχημάτων

Σχ.2.1. Πολυπλέκτης MPEG-2 TS	17
Σχ.2.2. Δομή Πακέτου PES	17
Σχ.2.3. Δομή Πακέτου MPEG-2 TS	19
Σχ.2.4. Πολυπλεξία πολλαπλών Ροών Μεταφοράς MPEG-2	20
Σχ.2.5. Λειτουργικό Διάγραμμα Διαμορφωτή DVB-T	25
Σχ.2.6. Τυχαιοποίηση Δεδομένων Ροής Μεταφοράς MPEG-2	26
Σχ.2.7. Στάδια Προσαρμογής Τυχαιοποίησης, Εξωτερικής Κωδικοποίησης και Διεμπλοκής	28
Σχ.2.8. Διάγραμμα Εξωτερικού Διεμπλοκέα και Αποδιεμπλοκέα	29
Σχ.2.9. Μητρικός Συνελικτικός Κώδικας με Ρυθμό Κώδικα $\frac{1}{2}$	29
Σχ.2.10. Εσωτερική Κωδικοποίηση και Εσωτερική Διεμπλοκή	30
Σχ.2.11. Ωφέλιμος Ρυθμός Μετάδοσης (Mb/s) για όλους τους Συνδυασμούς Ρυθμού Κωδικοποίησης, Σχήματος Διαμόρφωσης και Διαστήματος Φρούρησης (Κανάλι 8MHz)	31
Σχ.2.12. Η Επικεφαλίδα TCP	32
Σχ.2.13. Η Επικεφαλίδα UDP	33
Σχ.3.1. Τοπολογία Πειραματικού Δικτύου DVB-T	39
Σχ.3.2. Ο Διαμορφωτής COFDM που Παρουσιάζει τα Στοιχεία της Διαμόρφωσης	42
Σχ.3.3. Ο Διαμορφωτής COFDM που Παρέχει το DVB-T Stream στο Κανάλι 29 σε Επίπεδο Δύναμης 0 dBm	43
Σχ.3.4. Αρχιτεκτονική CMN	48
Σχ.3.5. Ενδιάμεσος Κόμβος Διανομής (CMN), Κάρτα Επιγείας Ψηφιακής Τηλεόρασης, Κεραία Λήψης	49
Σχ.3.6. Κάρτα Επιγείας Ψηφιακής Τηλεόρασης WinTV-NOVA-T-USB-2	50
Σχ.3.7. Κάρτα Δικτύου Intel® Rro/100s Desktop Adapter	51
Σχ.3.8. ISDN Network Termination Unit της INDRACOM	51
Σχ.3.9. Level One Wireless Lan PCI Adapter 11g	52
Σχ.3.10. Εντολή ip-forwarding	54
Σχ.3.11. Εντολή xconfig	55
Σχ.3.12. Εφαρμογή xconfig	56
Σχ.3.13. Επιλογή Κάρτας Επιγείας Ψηφιακής Τηλεόρασης	57
Σχ.3.14. Επιλογή rp-filter	58
Σχ.3.15. Εντολή ip-route για τη Δρομολόγηση της Κίνησης από την eth1 προς την eth0	58
Σχ.3.16. Δρομολόγηση των Κινήσεων Μέσα στον CMN	59
Σχ.3.17. Εντολή iptables	60
Σχ.3.18. Αρχείο “Athena”	61
Σχ.3.19. Διαθέσιμα Τηλεοπτικά Προγράμματα	61
Σχ.3.20. Συντονισμός στο Τηλεοπτικό Πρόγραμμα “DSO TV”	62
Σχ.4.1. Αρχιτεκτονική Δικτύου	65
Σχ.4.2. Throughput Δικτύου (σχ. 4.1)	66
Σχ.4.3. RTT Δικτύου (σχ. 4.1)	66
Σχ.4.4. Δίκτυο Πρόσβασης και Αλυσίδα Uplink – Downlink για τον CMN 2	67
Σχ.4.5. Throughput Δικτύου (σχ. 4.4)	68
Σχ.4.6. RTT Δικτύου (σχ. 4.4)	68
Σχ.4.7. Δίκτυο Πρόσβασης για τον CMN 2	69
Σχ.4.8. Throughput Δικτύου (σχ. 4.7)	70
Σχ.4.9. RTT Δικτύου (σχ. 4.7)	70
Σχ.4.10. Retransmissions και Triple Duplicate Packets Δικτύου (σχ. 4.7)	71
Σχ.4.11. Αλυσίδα Uplink – Downlink για τον CMN 2	72

Σχ.4.12. Throughput Δικτύου (σχ. 4.11)	73
Σχ.4.13. RTT Δικτύου (σχ. 4.11)	73
Σχ.4.14. Δίκτυο Πρόσβασης και Αλυσίδα Uplink – Downlink για τον CMN 1	74
Σχ.4.15. Throughput Δικτύου (σχ. 4.14)	75
Σχ.4.16. RTT Δικτύου (σχ. 4.14)	75
Σχ.4.17. Δίκτυο Πρόσβασης για τον CMN 1	76
Σχ.4.18. Throughput Δικτύου (σχ. 4.17)	77
Σχ.4.19. RTT Δικτύου (σχ. 4.17)	77
Σχ.4.20. Αλυσίδα Uplink – Downlink για τον CMN 1	78
Σχ.4.21. Throughput Δικτύου (σχ. 4.20)	79
Σχ.4.22. RTT Δικτύου (σχ. 4.20)	79
Σχ.4.23. Μέσο Throughput για τις διάφορες τιμές του TCP Window Size	80
Σχ.4.24. Μέσο RTT για τις διάφορες τιμές του TCP Window Size	81
Σχ.4.25. Throughput για 16k Window Size	84
Σχ.4.26. RTT για 16k Window Size	84
Σχ.4.27. Sequence Number για 16k Window Size	85
Σχ.4.28. Sequence Number Zoom για 16k Window Size	85
Σχ.4.29. Throughput για 64k Window Size	86
Σχ.4.30. RTT για 64k Window Size	86
Σχ.4.31. Sequence Number για 64k Window Size	87
Σχ.4.32. Sequence Number Zoom για 64k Window Size	87
Σχ.4.33. Throughput για 128k Window Size	88
Σχ.4.34. RTT για 128k Window Size	88
Σχ.4.35. Sequence Number για 128k Window Size	89
Σχ.4.36. Sequence Number Zoom για 128k Window Size	89
Σχ.4.37. Throughput για 512k Window Size	90
Σχ.4.38. RTT για 512k Window Size	90
Σχ.4.39. Sequence Number για 512k Window Size	91
Σχ.4.40. Sequence Number Zoom για 1512k Window Size	91
Σχ.4.41. Αρχιτεκτονική Δικτύου με Δύο Χρήστες	92
Σχ.4.42. Throughput για Δύο ADSL Χρήστες	93
Σχ.4.43. Μέσο RTT για Διαφορετικό ICMP Packet Size και Διαφορετικό Bit Rate	94
Σχ.4.44. Μέσο Throughput για διαφορετικό Bit Rate	96
Σχ.4.45. Μέσο RTT για διαφορετικό Bit Rate	97
Σχ.4.46. Inter – arrival Jitter	98
Σχ.4.47. One Way Delay	99
Σχ.4.48. Packet-to-Packet Delay Variation	99
Σχ. A.1 Εντολή tcpdump στον Server	104
Σχ. A.2 Εντολή tcpdump στον Client	105
Σχ. A.3 Εντολή iperf στον Server	105
Σχ. A.4 Εντολή iperf στον Client	106
Σχ. A.5 Εντολή tcpdump στον Server	110
Σχ. A.6 Εντολή tcpdump στον Client	110
Σχ. A.7 Εντολή Mgen στον Server	111
Σχ. A.8 Εντολή Mgen στον Client	111

Ακρωνύμια

A

AAI ATM Adaption Layer

ACK Acknowledgement packet

ADSL Asymmetric Digital Subscriber Line

AIX Advanced Interactive eXecutive

AMD Advanced Micro Devices

AoD Audio on Demand

API Application Program(ming) Interface

ATM Asynchronous Transfer Mode

B

BDP Bandwidth Delay Products

BSD Berkeley Software Distribution

C

COFDM Coded Orthogonal Frequency Division Multiplexing

CMN Cell Main Node

D

DSL Digital Subscriber Line

DSLAM Digital Subscriber Line Access Multiplexer

DVB-C Digital Video Broadcasting-Cable

DVB-H Digital Video Broadcasting-Handover

DVB-S Digital Video Broadcasting-Satellite

DVB-T Digital Video Broadcasting-Terrestrial

DVD Digital Video Decoder

E

ES Elementary Stream

ETSI European Telecommunications Standard Institute

F

FEC Forward Error Correction

G

GOP Group Of Pictures

GPRS General Packet Radio Service

GSM Global System For Mobile Communications

H

HDTV High Definition Television

HP-UX Hewlett Packard UniX

I

IEEE Institute of Electrical & Electronics Engineers

IFFT Inverse Fast Fourier Transform

IP Internet Protocol

IPTV Internet Protocol Television

ISDB-T International Standard Digital Broadcasting-Terrestrial

ISDN Integrated Services Digital Network

ISM Industrial, Scientific, Medical

ISP Internet Service Provider

ITU-R International Telecommunications Union – Radio Communication

ISDN Integrated Services Digital Network

J

JPEG Joint Photographic Experts Group

M

MAC Media Address Controller

MAC OS X Macintosh Operating System X

MHP Multimedia Home Platform

MPE Multi Protocol Encapsulation

MPEG Motion Pictures Experts Group

MSS Maximum Segment Size

MTU Maximum Transmission Unit

N

NAT Network Address Translation

NRL Naval Research Laboratory

NTSC National TV Standards Committee

O

OPAL Omni-Purpose Apparatus at LEP

P

PAT Programm Association Table

PCR Program Clock Reference

PES Packetised Elementary Stream

PID Programm Identifier

PMT Programm Map Table

POSIX Portable Operating System Interface for uniX

PS Program Stream

Q

QAM Quadrature Amplitude Modulation

QoS Quality of Service

QPSK Quadrature Phase Shift Keying

R

RAM Random-Access Memory

RDVB-T Regenerative Digital Video Broadcasting-Terrestrial

RS Reed Solomon

RTO Retransmission Time Out

S

SFN Single Frequency Networks

SNR Signal-to-noise ratio

T

TCP Transmission Control Protocol

TPS Transactions Per Second

TS Transport Stream

TV Television

U

UDP User Datagram Protocol

UHF Ultra High Frequency

ULE Ultra-Light Encapsulation

UMTS Universal Mobile Telecommunications System

V

VCR Video Cassette Recorder

VLC VideoLAN Client

VoD Video on Demand

W

WLAN Wireless Lan

WEP Wired Equivalent Privacy

1. ΕΙΣΑΓΩΓΗ

1.1. Γενική Περιγραφή

Το θέμα με το οποίο ασχολείται η συγκεκριμένη πτυχιακή εργασία αφορά τη μελέτη, τη σχεδίαση, την υλοποίηση και την αξιολόγηση ενός υβριδικού δικτύου παροχής αμφίδρομων διαδραστικών υπηρεσιών (TCP/IP, UDP/IP) με τη χρήση πλατφόρμας επίγειας ψηφιακής τηλεόρασης. Στα πλαίσια αυτά, ο σχεδιασμός και η μελέτη ενός τέτοιου τύπου δικτύου στηρίζονται στην εισήγηση μίας κατάλληλης αρχιτεκτονικής, η οποία επιτρέπει την πρόσβαση σε διαδραστικές υπηρεσίες, κάνοντας χρήση της τεχνολογίας του Digital Video Broadcasting -Terrestrial (DVB-T) στην κατερχόμενη ζεύξη (downlink) και των τεχνολογιών ISDN και WLAN στα κανάλια επιστροφής (uplinks). Προς αυτήν την κατεύθυνση θα μελετηθεί η αρχιτεκτονική ενός τέτοιου δικτύου και θα υλοποιηθεί μία πρότυπη υποδομή σύγκλισης IP/DVB-T, η οποία θα αποτελέσει τη βάση για την πραγματοποίηση μίας σειράς ερευνητικών πειραμάτων αξιολόγησης και εκτίμησης της απόδοσής της.

1.2. Στόχοι

Ο βασικός στόχος της πτυχιακής εργασίας είναι να προσδιορίσει, υλοποιήσει, επιδείξει και ελέγξει μία νέα αρχιτεκτονική δικτύου επίγειας ψηφιακής τηλεόρασης για την παροχή αμφίδρομων διαδραστικών υπηρεσιών. Το δίκτυο θα χρησιμοποιηθεί για τη διασύνδεση ενδιάμεσων κόμβων διανομής και θα δοκιμαστεί, επιδειχθεί και ελεγχθεί για την ορθότητά του πάνω από την πλατφόρμα DVB-T του Εργαστηρίου Έρευνας και Ανάπτυξης Τηλεπικοινωνιακών Συστημάτων, ΠΑΣΙΦΑΗ. Η πτυχιακή εργασία θα συμπεριλαμβάνει την υλοποίηση, δοκιμή και έλεγχο ορθότητας της συνολικής απόδοσης του δικτύου με τη χρήση ενσύρματων επίγειων δικτύων συνεισφέροντας έτσι στην περαιτέρω σύγκλιση της τεχνολογίας DVB-T με τα δίκτυα τηλεπικοινωνιών σε ένα ολοκληρωμένο και συμπαγή περιβάλλον. Εκτενείς μετρήσεις αξιολόγησης της απόδοσης του προτεινόμενου δικτύου θα παρθούν μετά το στάδιο υλοποίησης και τα κατάλληλα συμπεράσματα θα παρατεθούν και θα σχολιασθούν.

1.3. Δομή

Σε μια προσπάθεια να καλυφθούν ικανοποιητικά τα παραπάνω θέματα, επιχειρείται να εξετασθούν όλες οι παράμετροι που καθορίζουν τη λειτουργία ενός τέτοιου δικτύου με την πραγματοποίηση των κατάλληλων μετρήσεων έτσι ώστε να επιτευχθούν οι απαιτούμενες βελτιστοποιήσεις για την καλύτερη δυνατή απόδοση του συστήματος.

Στην ενότητα 2 παρουσιάζονται θεωρητικά στοιχεία σχετικά με την τεχνολογία της επίγειας ψηφιακής τηλεόρασης και τον δημιουργία σημάτων βασικής ζώνης. Ακολούθως, επιχειρείται η περιγραφή του τρόπου δημιουργίας της τελικής ροής μεταφοράς MPEG-2 (MPEG-2 TS). Στη συνέχεια, γίνεται μία εκτενέστερη αναφορά στο Ευρωπαϊκό πρότυπο DVB-T πάνω στο οποίο στηρίζεται η λειτουργία της επίγειας ψηφιακής τηλεόρασης. Τέλος, ακολουθεί μια αναφορά στα πρωτόκολλα TCP και UDP καθώς παρουσιάζονται και συγκεκριμένες τεχνολογίες που χρησιμοποιούνται εναλλακτικά στην υλοποίηση καναλιού εκπομπής, επιστροφής και δικτύου πρόσβασης.

Αντικείμενο της ενότητας 3 αποτελεί η παρουσίαση μιας υβριδικής αρχιτεκτονικής ασύμμετρων δικτύων (πελάτη-εξυπηρετητή) στα οποία οι τεχνολογίες ISDN και WLAN εφαρμόζεται στα κανάλια επιστροφής. Πιο συγκεκριμένα, το προτεινόμενο δίκτυο παρουσιάζεται σχεδιαστικά, περιγράφεται η λειτουργία του και εξηγείται αναλυτικά ο ρόλος του κάθε στοιχείου από το οποίο αποτελείται. Επίσης, περιγράφεται η δημιουργία ενός ενδιάμεσου κόμβου διανομής (CMN) και αναλύεται ο ρόλος του μέσα στο συγκεκριμένο ευρυζωνικό δίκτυο.

Η ενότητα 4 ασχολείται με τη στοιχειοθέτηση των πειραματικών μετρήσεων οι οποίες πραγματοποιήθηκαν στα πλαίσια της πτυχιακής αυτής εργασίας. Στη συνέχεια, μελετάται και εξετάζεται η συμπεριφορά του προτεινόμενου αμφίδρομου διαδραστικού δικτύου DVB-T, το οποίο υλοποιήθηκε στα πλαίσια της πτυχιακής εργασίας, με βάση δύο διαφορετικά είδη κίνησης (TCP και UDP). Τα αποτελέσματα των μετρήσεων καταγράφηκαν και αναλύθηκαν με στόχο τη εξαγωγή όσο το δυνατόν χρησιμότερων συμπερασμάτων σε μία προσπάθεια βελτιστοποίησης των επιδόσεων αυτών.

Η ολοκλήρωση της εργασίας αυτής πραγματοποιείται με την ενότητα 5 καταγράφοντας τα συμπεράσματα που προκύπτουν από τη μελέτη του αμφίδρομου διαδραστικού δικτύου και προτείνονται θέματα για την επέκταση της συγκεκριμένης μελέτης στην ίδια θεματική περιοχή.

1.4. Θεωρητικό Υπόβαθρο για την Τεχνολογία Digital Video Broadcasting- Terrestrial (DVB-T)

Η εξέλιξη του προτύπου επίγειας ψηφιακής τηλεόρασης DVB-T και η εφαρμογή του πάνω σε επίγεια κανάλια είναι μία από τις πιο σημαντικές τεχνολογικές εξελίξεις στις ασύρματες τηλεπικοινωνίες. Μια πλατφόρμα DVB-T μπορεί να αξιοποιηθεί όχι μόνο για την εκπομπή "μπουκέτων" τηλεοπτικών προγραμμάτων σε ένα μεγάλο αριθμό τελικών χρηστών κατανεμημένων πάνω σε μεγάλες γεωγραφικά περιοχές αλλά και για την παροχή αμφίδρομων Διαδικτυακών υπηρεσιών. Η χαρακτηριστική ιδιότητα του DVB-T να συνδυάζει τηλεοπτικά προγράμματα MPEG-2 με υπηρεσίες IP στο ίδιο κανάλι μεταφοράς, επιτρέπει την ανάπτυξη δικτυακών υποδομών οι οποίες μπορούν να διασυνδέσουν ενδιάμεσους κόμβους διανομής για την παροχή υπηρεσιών κατά απαίτηση.

1.5. Σύνοψη

Η μελέτη αυτή υλοποιεί την ιδέα της σύγκλισης των τηλεπικοινωνιών και των τεχνολογιών εκπομπής διερευνώντας ένα μοντέλο υβριδικού ευρυζωνικού δικτύου βασισμένου στην επίγεια ψηφιακή τηλεόραση (DVB-T). Η αρχιτεκτονική που προτείνεται συνδυάζει την τεχνολογία του DVB-T με μια ήδη υπάρχουσα δικτυακή υποδομή, για να προσφέρει στον τελικό χρήστη ψηφιακά τηλεοπτικά προγράμματα μαζί με υψηλής ταχύτητας υπηρεσίες δεδομένων.

2. ΨΗΦΙΑΚΗ ΤΗΛΕΟΡΑΣΗ ΚΑΙ ΥΠΗΡΕΣΙΕΣ ΔΕΔΟΜΕΝΩΝ

2.1. Εισαγωγή

Η τηλεοπτική τεχνολογία έχει διατηρήσει την αναλογική μέθοδο εκπομπής σε μια εποχή που έχει επικρατήσει για τις περισσότερες μορφές πληροφορίας η ψηφιοποιημένη αποθήκευση και μεταφορά. Μόνο την τελευταία δεκαετία και χάρη στα πρότυπα συμπίεσης και πολυπλεξίας εικόνας της οικογένειας MPEG και το ευρωπαϊκό σύνολο προδιαγραφών DVB για διαμόρφωση και μετάδοση έχει αποκτήσει η ψηφιακή τηλεόραση μια συγκεκριμένη προοπτική.

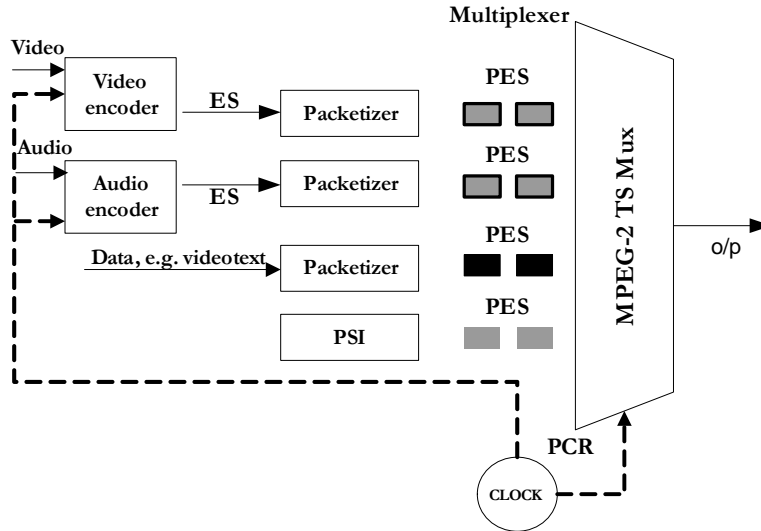
Είναι αριετὰ τα πλεονεκτήματα τα οποία εισάγει η ψηφιακή τεχνολογία στην διαμόρφωση και διανομή του τηλεοπτικού σήματος[Rysdale96]. Μερικά από αυτά είναι:

- Σταθερή ποιότητα εικόνας, με μεγαλύτερη ανοχή στις ατέλειες του ασύρματου ή ενσύρματου διαύλου. Εξάλειψη φαινομένων όπως αυτά της θόλωσης, των πολλαπλών ειδώλων ή του θορύβου («χιονιού»).
- Μειωμένος λόγος σήματος προς θόρυβο που απαιτείται σε σύγκριση με την αναλογική μετάδοση. Αυτό επιτρέπει τη μείωση της εκπεμπόμενης ισχύος μέχρι και κατά 30 dB χωρίς να διαταράσσεται η ποιότητα της εικόνας.
- Καλύτερη εκμετάλλευση του φάσματος λόγω της συμπίεσης του σήματος βασικής ζώνης. Για παράδειγμα, ένα επίγειο κανάλι UHF αντιστοιχεί σε ένα και μόνο αναλογικό τηλεοπτικό πρόγραμμα, ενώ το ίδιο εύρος μπορεί να φιλοξενήσει μέχρι και 4 ψηφιακά προγράμματα υψηλής ποιότητας.
- Ευέλικτες τεχνικές πολυπλεξίας των ψηφιακών συστημάτων, που επιτρέπουν την συνύπαρξη πολλών προγραμμάτων και υπηρεσιών επιλεγόμενης ποιότητας και ευκρίνειας.
- Μεταβλητή ταχύτητα (bit rate) εκπομπής, ανάλογα με τις απαιτήσεις ποιότητας του προγράμματος, κάτι που μεταξύ άλλων αυξάνει το κέρδος πολυπλεξίας (multiplexing gain) στην περίπτωση της ταυτόχρονης μετάδοσης πολλών προγραμμάτων μεταβλητού ρυθμού.
- Δυνατότητα επεξεργασίας της εικόνας στο δέκτη μετά τη λήψη μέσω αλγορίθμων ψηφιακής επεξεργασίας (digital image post-processing), όπως χρωματική διόρθωση, αποκοπή ορίων, αυξομείωση του μεγέθους ή αφαίρεση θορύβου.

- Εύκολος εμπλουτισμός των τηλεοπτικών προγραμμάτων μέσω τυποποιημένων αρχιτεκτονικών (π.χ. MHP, OpenTV) με τοπικές εφαρμογές που εκμεταλλεύονται τις δυνατότητες των σύγχρονων «έξυπνων δεκτών».
- Ενσωμάτωση διαφόρων πολυμεσικών εφαρμογών και υπηρεσιών δεδομένων, όπως αμφίδρομων υπηρεσιών και διαδικτυακής πρόσβασης σε μια κοινή ψηφιακή πλατφόρμα, με προϋπόθεση ότι υπάρχει διαθέσιμο κανάλι επιστροφής (reverse path).

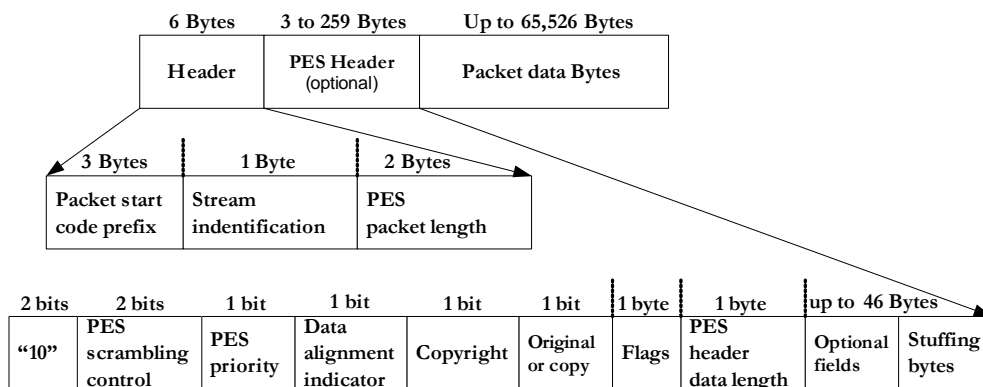
2.2. Σήμα Βασικής Ζώνης – Ροή Μεταφοράς MPEG-2

Για την ψηφιακή τηλεοπτική μετάδοση δεν αρκεί μόνο η απλή ψηφιοποίηση του τηλεοπτικού σήματος σε ασυμπιεστή παλμοσειρά PCM η οποία δημιουργεί ένα ψηφιακό σήμα ρυθμού μετάδοσης της τάξης των 270Mbps, αλλά απαιτείται η κατάλληλη αξιοποίηση αλγορίθμων συμπίεσης εικόνας έτσι ώστε να είναι εφικτή η ασύρματη εκπομπή των δεδομένων. Το πρότυπο για τη συμπίεση του τηλεοπτικού σήματος που χρησιμοποιείται σε όλα τα συστήματα μετάδοσης ψηφιακής τηλεόρασης της οικογένειας προτύπων DVB είναι το MPEG-2 το οποίο αξιοποιείται για την κωδικοποίηση σημάτων βίντεο και ήχου [1], [2]. Κατά τη σύσταση της κοινοπραξία του DVB Project αποφασίσθηκε ότι η κωδικοποίηση καθώς και η πολυπλεξία διαφορετικών πρωτογενών σημάτων που καθορίζονται από το πρότυπο MPEG-2 έπρεπε να αξιοποιηθεί αφενός λόγω της ευρείας αποδοχής του σε παγκόσμιο επίπεδο, αφετέρου για είναι εφικτή η κατάλληλη χρήση προσιτών ολοκληρωμένων κυκλωμάτων από διαφορετικούς κατασκευαστές ειδικά στους αποκωδικοποιητές. Έτσι τα συμπιεσμένα οπτικοακουστικά σήματα μαζί με άλλου είδους ψηφιακή πληροφορία πολυπλέκονται μαζί με βάση τις προδιαγραφές οι οποίες καθορίζονται στο πρότυπο MPEG-2 Systems [3] για τη δημιουργία μίας κοινής Ροής Μεταφοράς MPEG-2 (MPEG-2 Transport Stream) που αποτελεί το σήμα βασικής ζώνης για όλα τα συστήματα μετάδοσης DVB (Σχ. 2.1).



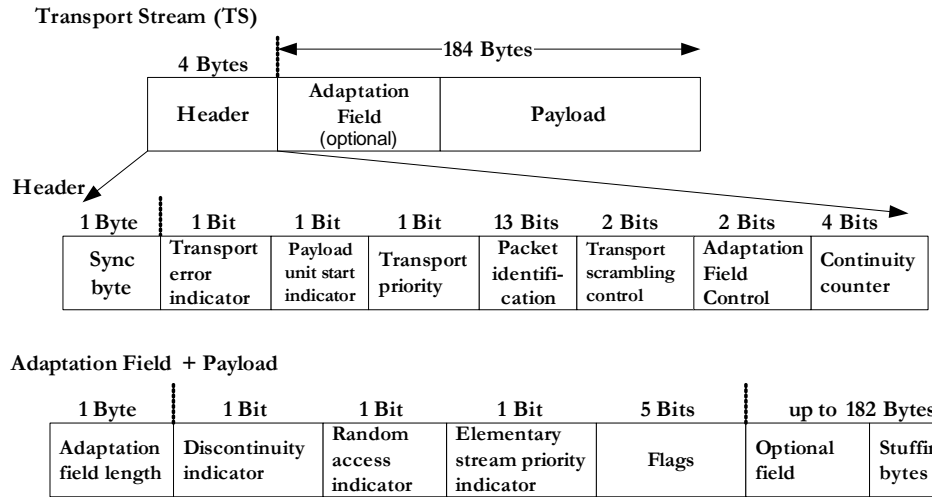
Σχ. 2.1 Πολυπλέκτης MPEG-2 TS

Για τη δημιουργία του σήματος βασικής ζώνης τα αναλογικά σήματα βίντεο και ήχου πρώτα κωδικοποιούνται με βάση το πρότυπο MPEG-2 έτσι ώστε να δημιουργηθούν ξεχωριστές Βασικές Ροές δεδομένων (Elementary Streams – ES) για κάθε ένα από αυτά [4]. Οι βασικές ροές στην πορεία διαμορφώνονται σε μορφή πακέτων δημιουργώντας τις Βασικές Ροές Πακέτων (Packetized Elementary Streams – PES). Τα πακέτα των ροών αυτών, τα οποία είναι μεταβλητού μήκους, περιέχουν πληροφορίες κεφαλίδας (header) και παρουσιάζονται στο Σχ. 2.2. Οι ροές πακέτων αυτές παραμένουν ξεχωριστές για τα σήματα βίντεο, ήχου και δεδομένων.



Σχ. 2.2 Δομή Πακέτου MPEG-2 PES

Το επόμενο στάδιο δημιουργίας του σήματος βασικής ζώνης αφορά την πολυπλεξία των βασικών ροών πακέτων για το σχηματισμό της τελικής ροής μεταφοράς MPEG-2 η οποία αποτελείται μία διαδοχή πακέτων σταθερού μήκους (transport packets) των 188 bytes. Στο Σχ. 2.3 παρουσιάζεται η δομή ενός πακέτου MPEG-2 TS το οποίο περιέχει μία κεφαλίδα των 4 bytes. Το αναγνωριστικό πακέτου (PID – Packet Identification) περιέχεται μέσα στην κεφαλίδα του πακέτου MPEG-2 TS και χρησιμοποιείται για τον εντοπισμό διαφορετικών τηλεοπτικών προγραμμάτων και υπηρεσιών μέσα σε μία κοινή ροή μεταφοράς MPEG-2. Έτσι τα πακέτα τα οποία μεταφέρουν το οπτικοακουστικό περιεχόμενο ενός συγκεκριμένου ψηφιακού τηλεοπτικού προγράμματος μέσα σε ένα κοινό μπουκέτο υπηρεσιών, περιέχουν το ίδιο PID. Δεδομένα τα οποία είναι μεγαλύτερα σε μέγεθος (πακέτα IP) υπόκεινται κατάτμηση και το περιεχόμενό τους διαμοιράζεται σε περισσότερα από ένα πακέτα MPEG-2 TS με βάση κάποια από τις διαθέσιμες τεχνικές ενθυλάκωσης οι οποίες είναι οι εξής: Data riping, Data streaming, Multiprotocol Encapsulation – MPE, Ultra-Light Encapsulation – ULE, Object carousel. Στην τελική ροή μεταφοράς MPEG-2 περιλαμβάνεται επίσης επιπρόσθετη πληροφορία δεδομένων PSI (Program Specific Information) η οποία καθορίζει τη σχέση μεταξύ των διαθέσιμων ψηφιακών τηλεοπτικών προγραμμάτων και των τιμών PID τους. Η πληροφορία PSI η οποία καθορίζεται από το πρότυπο MPEG-2 Systems περιλαμβάνει τέσσερις τύπους πινάκων και αξιοποιείται κυρίως έτσι ώστε να είναι εφικτή η αυτόματη ρύθμιση μίας συσκευής λήψης ψηφιακών τηλεοπτικών προγραμμάτων. Τα δεδομένα τα οποία περιλαμβάνονται στα δεδομένα PSI είναι τα παρακάτω: PMT - Program Map Table (συσχετίζει τα πακέτα με το ρολόι προγράμματος), PAT - Program Association Table (λίστα όλων των προγραμμάτων που περιέχονται στη ροή μεταφοράς MPEG-2), NIT - Network Information Table (δίνει πληροφορίες σχετικά με το δίκτυο εκπομπής) και CAT - Conditional Access Table (συσχετίζει περιπλεγμένα προγράμματα – scrambled programs). Κατά τη διάρκεια της διαδικασίας πολυπλεξίας για το σχηματισμό της ροής μεταφοράς MPEG-2 χρησιμοποιείται μία αναφορά ρολογιού προγράμματος (PCR – Program Clock Reference) η οποία αξιοποιείται για την αναπαραγωγή του ρολογιού των 27MHz και μεταδίδεται τουλάχιστον κάθε 0,1 sec.

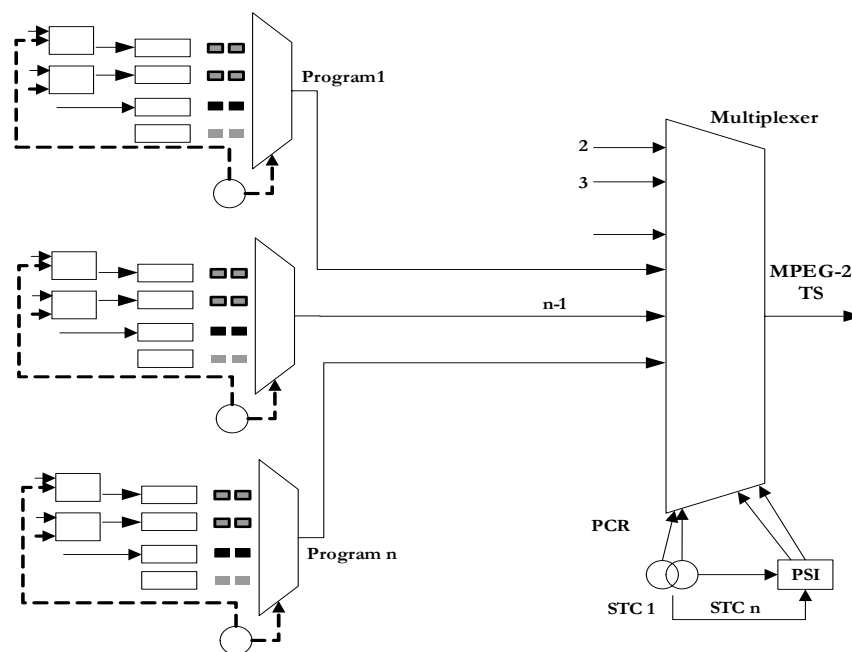


Σχ. 2.3 Δομή Πακέτου MPEG-2 TS

Η ροή μεταφοράς MPEG-2 αποτελεί μία μορφή πολυπλεξίας σχεδιασμένη για σύνθετες εφαρμογές όπως τα ψηφιακά τηλεοπτικά μπουκέτα τα οποία απαρτίζονται από πολλαπλά τηλεοπτικά προγράμματα και υπηρεσίες δεδομένων. Το πεδίο προσαρμογής (adaptation field) αξιοποιείται για να συμπληρώσει το τελευταίο από μία σειρά πακέτων τα οποία φέρουν ένα κατακερματισμένο σύνολο δεδομένων. Η χρήση μικρών και σταθερού μεγέθους πακέτων μεταφοράς διευκολύνει τη διαδικασία στην οποία υπόκεινται αυτά στην πορεία για την προστασία του ψηφιακού τηλεοπτικού σήματος από λάθη κατά τη διάρκεια της μετάδοσης. Η διαδικασία αυτή αφορά την προστασία των πακέτων με τη χρήση κώδικα τύπου Reed-Solomon και περιέχεται στις προδιαγραφές των προτύπων DVB. Η προστασία έναντι λαθών της ροής μεταφοράς MPEG-2 TS θεωρείται απαραίτητη έτσι ώστε να μειωθεί η πιθανότητα απώλειας δεδομένων κατά τη διάρκεια της εκπομπής σε κάποιο περιβάλλον ασύρματης μετάδοσης. Οι προδιαγραφές οι οποίες περιγράφονται από το πρότυπο MPEG-2 Systems καθορίζουν την ακριβή μορφή του σήματος βασικής ζώνης για τη δημιουργία της ροής μεταφοράς MPEG-2. Οι περαιτέρω διαδικασίες κωδικοποίησης καναλιού και διαμόρφωσης καθώς και οι μέθοδοι προστασίας έναντι λαθών οι οποίες εφαρμόζονται πριν την εκπομπή των δεδομένων, καθορίζονται από τα πρότυπα DVB που αφορούν τις προδιαγραφές μετάδοσης στο φυσικό επίπεδο.

Ένας πολυπλέκτης MPEG-2 TS μπορεί να αξιοποιηθεί όχι μόνο για τη πολυπλεξία των στοιχείων για τη δημιουργία μίας ροής μεταφοράς ενός μόνο τηλεοπτικού προγράμματος αλλά και για την πολυπλεξία μεταξύ πολλαπλών τηλεοπτικών προγραμμάτων. Αυτή η δυνατότητα επιτρέπει μία κοινή ροή μεταφοράς MPEG-2 να μεταφέρει πολλές τηλεοπτικές υπηρεσίες διαφορετικού

ρυθμού μετάδοσης η κάθε μία προς τους τελικούς χρήστες. Η τεχνική πολυπλεξίας πολλαπλών ψηφιακών προγραμμάτων παρουσιάζεται στο Σχ. 2.4. Κάθε τηλεοπτικό πρόγραμμα έχει τη δική του αναφορά ρολογιού προγράμματος (PCR - Program Clock Reference). Αυτά τα ανεξάρτητα ρολόγια των 27MHz δεν είναι κλειδωμένα μαζί και αυτό επιτρέπει την πολυπλεξία διαφορετικών προγραμμάτων από διαφορετικές πηγές τα οποία έχουν κωδικοποιηθεί σε ξεχωριστές χρονικές στιγμές. Οι πολυπλεξία πακέτων πρέπει να ακολουθεί μερικούς κανόνες έτσι ώστε να εξασφαλιστεί ότι η ροή μεταφοράς έχει απόλυτα συγχρονισμένα τα δεδομένα ήχου και εικόνας. Διαφορετικά είναι πιθανόν να σταλούν τα πακέτα του ήχου σε διαφορετικό χρονικό διάστημα από τα αντίστοιχα πακέτα της εικόνας. Για την αποφυγή αυτής της περίπτωσης όλα τα πακέτα ήχου, εικόνας και δεδομένων πολυπλέκονται στην τελική ροή μεταφοράς σε αντιστοιχία με το στιγμιαίο ρυθμό μετάδοσής τους. Αυτό εξασφαλίζει ότι ο συγχρονισμός μεταξύ όλων των πηγών διατηρείται. Η διαδικασία του σωστού συγχρονισμού επιτυγχάνεται χρονοπρογραμματίζοντας την μεταφορά των πακέτων χρησιμοποιώντας τον χρονοπρογραμματιστή πακέτων (packet scheduler). Ο χρονοπρογραμματιστής πακέτων ελέγχει τα επίπεδα του καταχωρητή ενός υποθετικού αποκωδικοποιητή ο οποίος ονομάζεται αποκωδικοποιητής συστήματος στόχου ροής μεταφοράς (TS-STD – Transport Stream-System Target Decoder). Η λειτουργία του συγκεκριμένου υποθετικού αποκωδικοποιητή εξασφαλίζει ότι κανένας από τους καταχωρητές βίντεο, ήχου και συστήματος δεν βρίσκεται σε κατάσταση υπερχείλισης (overflow) ή υπορροής (underflow).



Σχ. 2.4 Πολυπλεξία Πολλαπλών Ροών Μεταφοράς MPEG-2

2.3. Εισαγωγή στα πρότυπα DVB

Η οικογένεια προτύπων DVB (Digital Video Broadcasting) αναπτύχθηκε για να προσφέρει υπηρεσίες ψηφιακής τηλεόρασης εκμεταλλευόμενη ένα ευρύ φάσμα μέσων διανομής, δορυφορικών, καλωδιακών και επίγειων [DVB00]. Όλα τα DVB standards έχουν υιοθετήσει τα πρότυπα MPEG-2 για συμπίεση ήχου και κινούμενης εικόνας καθώς και για πολυπλεξία. Χάρη στη χρήση των πακέτων μεταφοράς MPEG-2 ως γενικευμένων "μεταφορέων δεδομένων" (data containers), ένα MPEG-2 Transport Stream -και συνεπώς ένα σύστημα DVB- μπορεί να μεταφέρει σχεδόν ο,τιδήποτε μπορεί να ψηφιοποιηθεί, από τηλεόραση υψηλής ευκρίνειας (HDTV), πολλαπλά κανάλια PAL/SECAM/NTSC, μέχρι και υψηλής ταχύτητας υπηρεσίες πολυμέσων και δεδομένων.

- Το δορυφορικό σύστημα DVB-S (Digital Video Broadcasting - Satellite) είναι το παλαιότερο και πιο διαδεδομένο από την οικογένεια προτύπων DVB και έχει αδιαμφισβήτητα τύχει παγκόσμιας αποδοχής. Το DVB-S σχεδιάστηκε για να εκμεταλλεύεται πλήρως το εύρος ζώνης των δορυφορικών τηλεοπτικών αναμεταδοτών. Χρησιμοποιεί ρυθμό μεταφοράς των 54Mbps με διαμόρφωση QPSK σε συνδυασμό με ένα σχήμα διπλής κωδικοποίησης και διεμπλοκής (coding/interleaving). Είναι το μόνο πρότυπο από την οικογένεια DVB που έχει τύχει εμπορικής εφαρμογής και στη χώρα μας.
- Το καλωδιακό σύστημα DVB-C (Digital Video Broadcasting - Cable) έχει τεχνικά αρκετές ομοιότητες με το DVB-S. Η διαφορά του έγκειται στο ότι χρησιμοποιεί την αποδοτικότερη -αλλά και πιο ευαίσθητη σε παρεμβολές- διαμόρφωση 64QAM αντί για την QPSK. Έτσι, ένα καλωδιακό κανάλι των 8MHz μπορεί να μεταφέρει 38.5Mbps. Εναλλακτικά μπορεί να χρησιμοποιηθεί QAM λιγότερων ή περισσότερων επιπέδων. Σε κάθε περίπτωση, υπάρχει μια αντιστάθμιση μεταξύ ταχύτητας και αξιοπιστίας.
- Το επίγειο σύστημα DVB-T (Digital Video Broadcasting - Terrestrial) αποτελεί το τελευταίο χρονικά μέλος της οικογένειας DVB. Επιτυγχάνει ψηφιακή μετάδοση υψηλών ταχυτήτων πάνω από το "δύσκολο" επίγειο κανάλι, χρησιμοποιώντας διαμόρφωση πολλαπλών φερόντων στο σχήμα της πολυπλεξίας με ορθογωνική διαίρεση συχνότητας (Orthogonal Frequency Division Multiplexing - OFDM). Το σχήμα OFDM του DVB-T χρησιμοποιεί ένα μεγάλο αριθμό φερόντων (6817 ή 1704 για μετάδοση 8K και 2K αντίστοιχα), κάθε ένα από τα οποία διαμορφώνεται κατά QPSK, 16QAM ή 64QAM. Έτσι, η πληροφορία κατανέμεται ομοιόμορφα στο φάσμα και, σε συνδυασμό με

κωδικοποίηση και διεμπλοκή δύο στρωμάτων, το σήμα αποκτά μεγάλη ευρωστία ακόμη και σε περιβάλλοντα με ισχυρές διαλείψεις και φαινόμενα πολυδιαδρομικής μετάδοσης (multipath).

Παρόλο που υβριδικές αρχιτεκτονικές σαν την προτεινόμενη μπορούν να υλοποιηθούν με οποιοδήποτε από τα παραπάνω πρότυπα, η διπλωματική επικεντρώνεται στη μελέτη του επίγειου συστήματος (DVB-T). Το DVB-T είναι το πιο πρόσφατο (και πιο εξελιγμένο τεχνολογικά) πρότυπο, και οι προοπτικές χρήσης του για μετάδοση δεδομένων IP όχι μόνο είναι πολυάριθμες, αλλά και δεν έχουν ακόμη αναδειχθεί (δίκτυα παροχής δεδομένων βασισμένα στο DVB-S και DVB-C έχουν ήδη αναπτυχθεί αλλά οι δυνατότητες και οι προοπτικές τους είναι σχετικά περιορισμένες). Επίσης, το επίγειο σύστημα δεν απαιτεί ιδιαίτερο εξοπλισμό από πλευράς χρήστη (π.χ. δορυφορικό δέκτη ή καλωδιακή υποδομή) και από την πλευρά του παροχέα αποτελεί την πιο προσιτή και πιο ευέλικτη λύση σε σχέση με την δορυφορική μετάδοση ή το καλωδιακό δίκτυο. Τέλος, ένα επίγειο ψηφιακό σύστημα μπορεί να οργανωθεί σε περιοχές κάλυψης με κυψελωτή δομή και να προσφέρει υπηρεσίες και σε κινούμενους χρήστες, μια δυνατότητα που οι υπόλοιπες τεχνολογίες δεν μπορούν να προσφέρουν.

2.4. Γενικές Αρχές Επίγειας Ψηφιακής Μετάδοσης Ευρείας Κάλυψης (DVB-T)

Το πιο πρόσφατο χρονικά πρότυπο DVB-T (2004) [5] καθορίζει τις προδιαγραφές ενός βασικού συστήματος μετάδοσης για την επίγεια ψηφιακή ειπομπή ευρείας κάλυψης. Το πρότυπο αυτό αποτελεί τη συνέχεια του προγενέστερου προτύπου DVB-T (ETS 300 744: Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital Terrestrial television (DVB-T)) [6] και περιγράφει την κωδικοποίηση καναλιού/διαμόρφωση συστήματος που χρησιμοποιείται για την επίγεια μετάδοση του ψηφιακού τηλεοπτικού σήματος. Το πρότυπο DVB-T αναπτύχθηκε από το DVB Forum [7] και εγκρίθηκε από το Ευρωπαϊκό Ινστιτούτο Τηλεπικοινωνιακών Συστημάτων ETSI [8] το Νοέμβριο του 2004. Μέχρι σήμερα συστήματα μετάδοσης DVB-T αξιοποιούνται σε χώρες της Ευρωπαϊκής Ένωσης, τη Ρωσία, την Ανατολική Ευρώπη, την Ινδία, τη Σιγκαπούρη, τη Σαουδική Αραβία και την Αυστραλία [9]. Τα αντίστοιχα πρότυπα που αξιοποιούνται για την επίγεια ψηφιακή μετάδοση υπηρεσιών στις Ηνωμένες Πολιτείες Αμερικής και στην Ιαπωνία είναι το ATSC [10] και το ISDB-T [11]. Το πρότυπο ATSC υπολείπεται ως προς κάποια χαρακτηριστικά σε σχέση με το αντίστοιχο Ευρωπαϊκό όπως στην κινητικότητα του τελικού χρήστη, την αντοχή σε φαινόμενα πολυδιαδρομικών διαλείψεων και τη

δημιουργία Single Frequency Networks (SFN), ενώ το πρότυπο ISDB-T βασίζεται στη φιλοσοφία του προτύπου DVB-T.

2.4.1. Γενικά

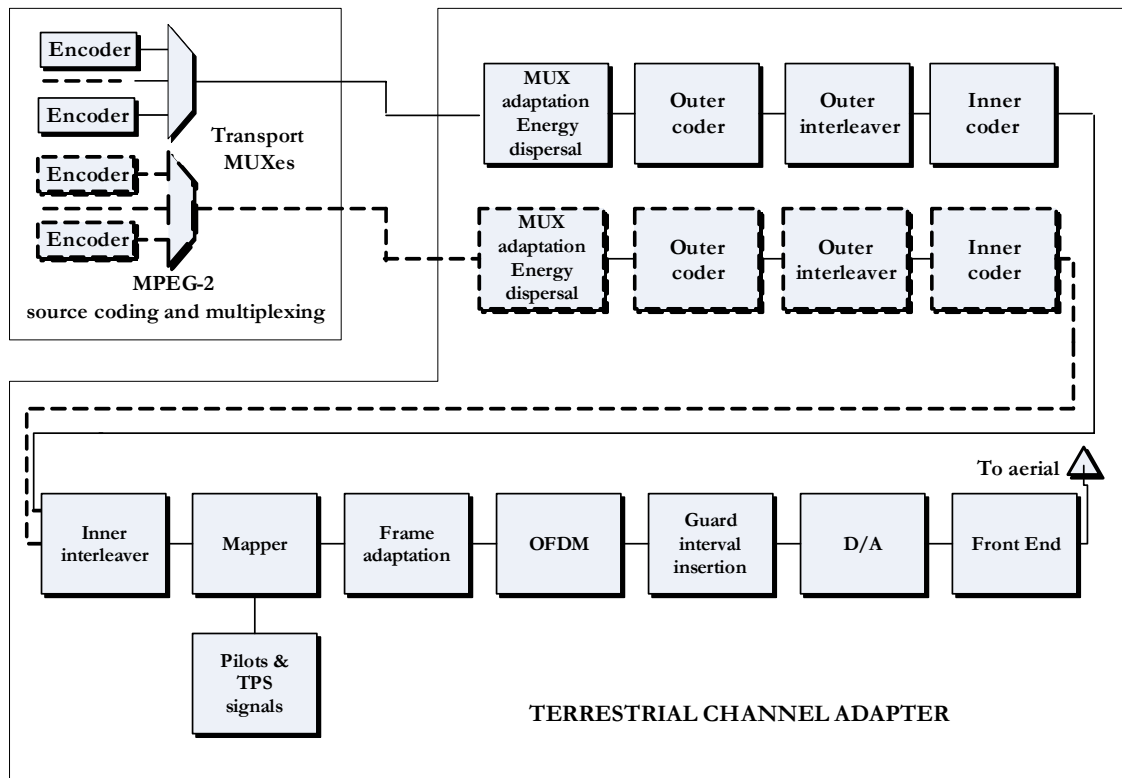
Το πρότυπο DVB-T καθορίζει τις προδιαγραφές ενός συστήματος επίγειας ειπομπής ευρείας κάλυψης και περιορίζεται αποκλειστικά στην περιγραφή των λειτουργιών του διαμορφωτή παραλείποντας τον καθορισμό των τεχνολογιών για την υλοποίηση των καναλιών επιστροφής και την παροχή αμφίδρομων πολυμεσικών και Διαδικτυακών υπηρεσιών, καθώς και τις προδιαγραφές για τη δημιουργία του σήματος βασικής ζώνης. Ο διαμορφωτής δέχεται από τον πολυπλέκτη/ενθυλακωτή την τελική ροή μεταφοράς MPEG-2 στην οποία περιέχονται πολυπλεγμένες οι υπηρεσίες εικόνας, ήχου και δεδομένων υπό τη μορφή σήματος βασικής ζώνης και στην έξοδό του παράγει το προς μετάδοση σήμα RF. Το εύρος ζώνης του σήματος αυτού είναι 8MHz και μεταδίδεται σε ένα από τα διαθέσιμα κανάλια UHF (21ch. – 69ch.).

Ο διαμορφωτής DVB-T χρησιμοποιεί σχήμα COFDM [12] στην επίγεια μετάδοση για την αντιμετώπιση του φαινομένου των επιλεκτικών ως προς τη συχνότητα διαλείψεων. Η τεχνική COFDM περιλαμβάνει μία μέθοδο για τη διαμόρφωση πολλαπλών φερόντων και αποτελεί ένα αντικείμενο μελέτης ευρέως ενδιαφέροντος για πολλά χρόνια. Έγινε πρώτα ελκυστική για τη μετάδοση σημάτων με τη χρήση διακριτών Μετασχηματισμών Fourier και πιο πρόσφατα χρησιμοποιώντας μετασχηματισμούς FFT. Είναι αρκετά ανθεκτική στη λήψη πολυδιαδρομικών σημάτων και είναι χρήσιμη για κανάλια που παρουσιάζουν γραμμικές παραμορφώσεις. Είναι η διαμόρφωση που επιλέχθηκε από την οικογένεια προτύπων DVB για την ανάπτυξη του Ευρωπαϊκού προτύπου επίγειας ψηφιακής τηλεόρασης (DVB-T) και ως τεχνική μετάδοσης γενικά είναι η κύρια υποψήφια για τις τεχνολογίες προσωπικών επικοινωνιών 4ης γενιάς [13] και των ασύρματων επικοινωνιών υψηλής ταχύτητας. Η τεχνική διαμόρφωσης COFDM χρησιμοποιεί χιλιάδες ξεχωριστά φέροντα σήματα για τη μεταφορά του σήματος δεδομένων, διαμοιράζοντας τα δεδομένα σε κάθε φέρον. Το σήμα δεδομένων έπειτα διαμορφώνει αυτά τα φέροντα σύμφωνα με τα πρότυπα διαμόρφωσης QPSK και QAM. Επιλέγοντας το σωστό τύπο του σήματος φέροντος είναι δυνατόν να σχηματιστεί ένα σύνολο φερόντων τα οποία είναι κοντά τοποθετημένα μαζί και δεν υπάρχει η ανάγκη για την ύπαρξη κενού εύρους συχνότητας μεταξύ τους. Αυτό είναι ένα βασικό στοιχείο της τεχνικής COFDM όπου τα φέροντα σήματα είναι κοντά τοποθετημένες με τέτοιο τρόπο σε ένα κανάλι RF δίχως την ύπαρξη κενού εύρους ενώ κάθε ξεχωριστό φέρον μπορεί ληφθεί από το σύνολο τους. Σύμφωνα με τα χαρακτηριστικά μετάδοσης και τις ρυθμίσεις λειτουργίας για τη μετάδοση

DVB-T είναι δυνατόν να επιτευχθεί ωφέλιμος ρυθμός μετάδοσης μεταξύ 4Mb/s έως και 31Mb/s. Αυτό αποτελεί μία δυνατότητα για περισσότερη ευελιξία στο σχεδιασμό δικτύων επίγειας ψηφιακής μετάδοσης ευρείας κάλυψης. Το πρότυπο DVB-T συνδυάζει την τεχνική COFDM με κωδικοποίηση και διεμπλοκή δύο επιπέδων που καθιστούν το σήμα ιδιαίτερα ανθεκτικό σε καταστάσεις πολυδιαδρομικής διάδοσης και παρεμβολών [14].

Το σύστημα επίγειας ψηφιακής μετάδοσης καθορίζεται στο Σχ. 2.5 όπου παρουσιάζεται το λειτουργικό μπλοκ διάγραμμα με βάση το οποίο πραγματοποιείται η προσαρμογή των τηλεοπτικών σημάτων βασικής ζώνης από την έξοδο του πολυπλέκτη/ενθυλακωτή MPEG-2 TS στα χαρακτηριστικά του επίγειου καναλιού. Οι βασικές λειτουργίες της μετατροπής του σήματος βασικής ζώνης στο προς μετάδοση σήμα παρουσιάζονται παρακάτω με τη σειρά που εφαρμόζονται στη ροή μεταφοράς MPEG-2:

- Προσαρμογή ροής μεταφοράς και τυχαιοποίηση
- Εξωτερική κωδικοποίηση (προστασία έναντι λαθών με κώδικα Reed-Solomon)
- Εξωτερική συνελικτική διεμπλοκή (convolutional interleaving)
- Εσωτερική κωδικοποίηση με διατρητό συνελικτικό κώδικα (punctured convolutional code)
- Εσωτερική διεμπλοκή (inner interleaving)
- Αντιστοίχιση και διαμόρφωση φερόντων σημάτων
- Μετάδοση με τη χρήση πολυπλεξίας με διαίρεση σε ορθογώνιες συχνότητες (OFDM)
- Άνω μετατροπή στη συχνότητα RF (up conversion)



Σχ. 2.5 Λειτουργικό Διάγραμμα Διαμορφωτή DVB-T

Το σύστημα αυτό είναι απόλυτα συμβατό με τα κωδικοποιημένα τηλεοπτικά σήματα MPEG-2 σύμφωνα με τα πρότυπα ISO/IEC 13818 [3], [15], [2] και έχει σχεδιαστεί για τη μετάδοση επίγειων ψηφιακών τηλεοπτικών υπηρεσιών στην ήδη υπάρχουσα περιοχή φάσματος των συχνοτήτων UHF που χρησιμοποιείται για τη μετάδοση των αναλογικών τηλεοπτικών προγραμμάτων σήμερα. Επιπλέον απαιτείται από το σύστημα μετάδοσης να παρέχει επαρκή προστασία στα υψηλά επίπεδα των παρεμβολών παρακείμενου καναλιού και των ομοκαναλικών παρεμβολών που προέρχονται από τις υπάρχουσες υπηρεσίες PAL/SECAM/NTSC και επίσης απαιτείται ότι το σύστημα αυτό επιτρέπει την καλύτερη δυνατή απόδοση φάσματος όταν χρησιμοποιείται στις συχνότητες UHF. Αυτή η απαίτηση μπορεί να επιτευχθεί όταν αξιοποιηθεί η λειτουργία SFN.

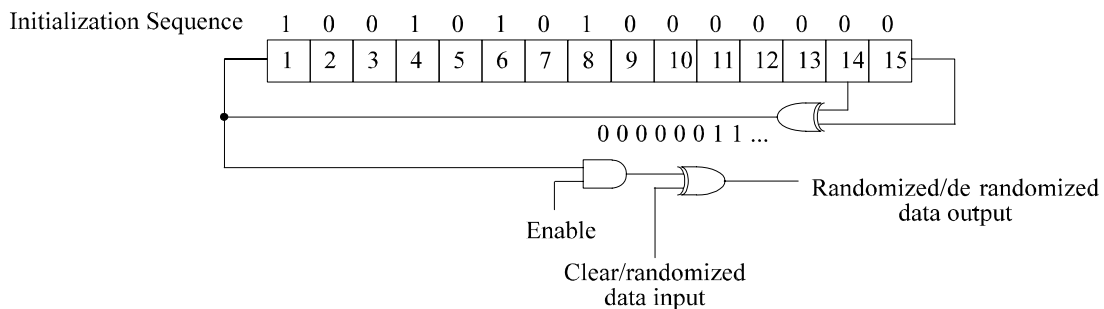
Για την αξιοποίηση της κοινής χρήσης με τα πρότυπα του ETSI για τη δορυφορική [16] και καλωδιακή [17] μετάδοση, η εξωτερική κωδικοποίηση και διεμπλοκή είναι κοινές με αυτές του προτύπου DVB-T ενώ η εσωτερική κωδικοποίηση είναι κοινή μόνο με το πρότυπο της δορυφορικής εκπομπής. Επιπλέον για τον βέλτιστο συμβιβασμό μεταξύ της τοπολογίας του δικτύου επίγειας ψηφιακής τηλεόρασης και της απόδοσης της συχνότητας, ένα διάστημα φρούρησης

καθορίζεται που επιτρέπει διαφορετικές δικτυακές διατάξεις όπως δικτυα SFN μεγάλης εμβέλειας διατηρώντας παράλληλα τη μέγιστη απόδοση συχνότητας.

Το σύστημα μετάδοσης DVB-T επιτρέπει τη χρήση διαφορετικών επιπέδων διαμόρφωσης QAM και διαφορετικούς ρυθμούς εσωτερικής κωδικοποίησης και παράλληλα επιτρέπει την αξιοποίηση δύο επιπέδων ιεραρχικής κωδικοποίησης και διαμόρφωσης καναλιού. Σε αυτήν την περίπτωση το λειτουργικό μπλοκ διάγραμμα του συστήματος επεκτείνεται περιλαμβάνοντας τα στοιχεία που φαίνονται στο Σχ. 2.5 με διακεκομμένες γραμμές. Στην περίπτωση της ιεραρχικής διαμόρφωσης το σήμα βασικής ζώνης διαχωρίζεται σε δύο ροές μεταφοράς: μία υψηλής και μία χαμηλής προτεραιότητας και τα δύο αυτά σήματα διαμορφώνονται ταυτόχρονα σε ένα ιεραρχικό σχήμα QAM. Έτσι ένας δέκτης με κακές συνθήκες λήψης λαμβάνει μόνο τα δεδομένα υψηλής προτεραιότητας ενώ ένας με καλύτερες λαμβάνει το σύνολο των υπηρεσιών. Η λειτουργία αυτή δίνει τη δυνατότητα ευελιξίας στο σύστημα μετάδοσης της επίγειας ψηφιακής τηλεόρασης ειδικά όταν συνοδευτεί από κλιμακωτή κωδικοποίηση της κινούμενης εικόνας κατά MPEG-2.

2.4.1.1. Προσαρμογή Ροής Μεταφοράς και Τυχαιοποίηση για Διασπορά Ενέργειας

Η ροή μεταφοράς MPEG-2 που προέρχεται από την έξοδο του πολυπλέκτη/ενθυλακωτή και εισέρχεται στον διαμορφωτή COFDM είναι σταθερού ρυθμού (Constant Bit Rate - CBR) και αποτελείται από πακέτα σταθερού μήκους των 188 bytes (Σχ. 2.7 α) τα οποία περιλαμβάνουν το κάθε ένα από ένα byte συγχρονισμού. Προκειμένου να εξασφαλιστούν επαρκείς δυαδικές μεταβολές, τα δεδομένα εισόδου της ροής μεταφοράς MPEG-2 δέχονται μία διαδικασία τυχαιοποίησης η οποία παρουσιάζεται στο Σχ. 2.6



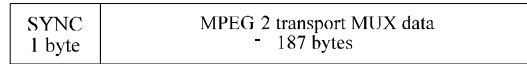
Σχ. 2.6 Τυχαιοποίηση Δεδομένων Ροής Μεταφοράς MPEG-2

Με τη διαδικασία αυτή εξασφαλίζεται ότι το σήμα COFDM έχει ισότιμα κατανεμημένη την πληροφορία σε όλο το διαθέσιμο εύρος ζώνης. Αυτό επιτυγχάνεται εισάγοντας τη ροή μεταφοράς

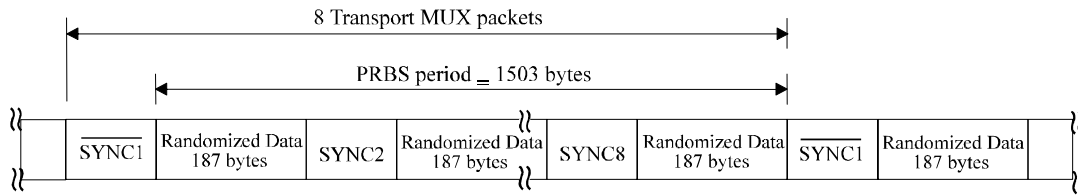
σε έναν περιπλέκτη (scrambler) ο οποίος χρησιμοποιεί μία γεννήτρια ψευδοτυχαίας δυαδικής ακολουθίας (Pseudo Random Binary Sequence – PRBS generator) έτσι ώστε όλη η πληροφορία του σήματος MPEG-2 να διαμοιραστεί τυχαία σε όλη τη ροή δεδομένων. Αυτή η διαδικασία επιπλέον εξασφαλίζει ότι θα υπάρχουν αραιές δυαδικές μεταβολές και αυτό βοηθάει για την αντιμετώπιση λαθών κατά τη διάρκεια της μετάδοσης. Η γεννήτρια PRBS που χρησιμοποιείται έχει μία περίοδο των 1503 bytes και έχει τη δυνατότητα να διαχειρίζεται 8 πακέτα δεδομένων κάθε φορά (Σχ. 2.7 b). Για την παροχή ενός σήματος έναρξης για τον δέκτη, το byte συγχρονισμού του πρώτου από τα 8 συνεχή πακέτα μέσα στη ροή μεταφοράς αναστρέφεται δυφιδήν (bit-wise inverted) και το πρώτο bit από την έξοδο της γεννήτριας PRBS εφαρμόζεται στο πρώτο bit (MSB) του πρώτου byte που ακολουθεί το ανεστραμμένο αυτό byte συγχρονισμού. Για τη δυνατότητα αξιοποίησης πρόσθετων λειτουργιών συγχρονισμού, κατά τη διάρκεια των bytes συγχρονισμού των υπόλοιπων 7 συνεχόμενων πακέτων, η γεννήτρια PRBS συνεχίζει τη λειτουργία της αλλά η έξοδος της απενεργοποιείται διατηρώντας αυτά τα bytes όπως είναι δίχως να έχουν υποστεί τη διαδικασία της τυχαιοποίησης. Έτσι, η περίοδος της ακολουθίας PRBS όπως αναφέρθηκε παραπάνω είναι 1503 bytes.

2.4.1.2. Εξωτερική Κωδικοποίηση και Συνελικτική Διεμπλοκή

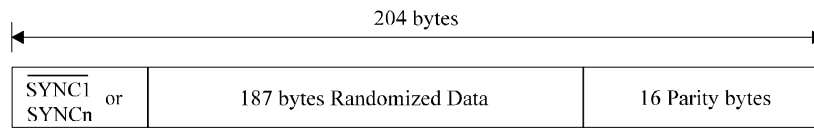
Μετά την προσαρμογή της ροής μεταφοράς και την τυχαιοποίηση ακολουθούν τα στάδια της εξωτερικής κωδικοποίησης και της συνελικτικής διεμπλοκής. Συντομευμένος κώδικας Reed-Solomon RS (204,188, t=8), ο οποίος προκύπτει από τον αρχικό συστηματικό κώδικα RS (255,239, t=8), εφαρμόζεται σε κάθε τυχαιοποιημένο πακέτο (188 bytes) έτσι ώστε να προκύψουν προστατευμένα από λάθη πακέτα των 204 bytes (Σχ. 2.7 c). Ο κώδικας Reed-Solomon εφαρμόζεται επίσης στο byte συγχρονισμού των πακέτων είτε είναι ανεστραμμένο είτε όχι και επιτρέπει τη διόρθωση μέχρι και 8 λανθασμένων bytes σε τυχαίες θέσεις μέσα στο προστατευμένο πακέτο. Ο συντομευμένος κώδικας Reed-Solomon μπορεί να υλοποιηθεί προσθέτοντας 51 bytes (όλα μηδέν) πριν από τα bytes της πληροφορίας στην είσοδο του κωδικοποιητή RS (255,239, t=8). Μετά τη διαδικασία της κωδικοποίησης RS αυτά τα μηδενικά bytes απορρίπτονται οδηγώντας σε μία λέξη κώδικα RS των N=204 bytes.



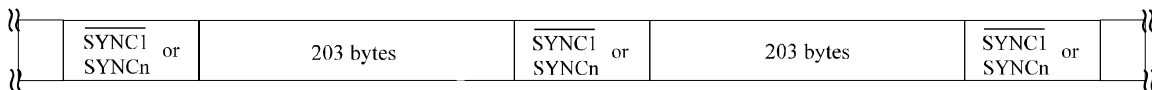
a) MPEG-2 transport MUX packet



b) Randomized transport packets: Sync bytes and Randomized Data bytes



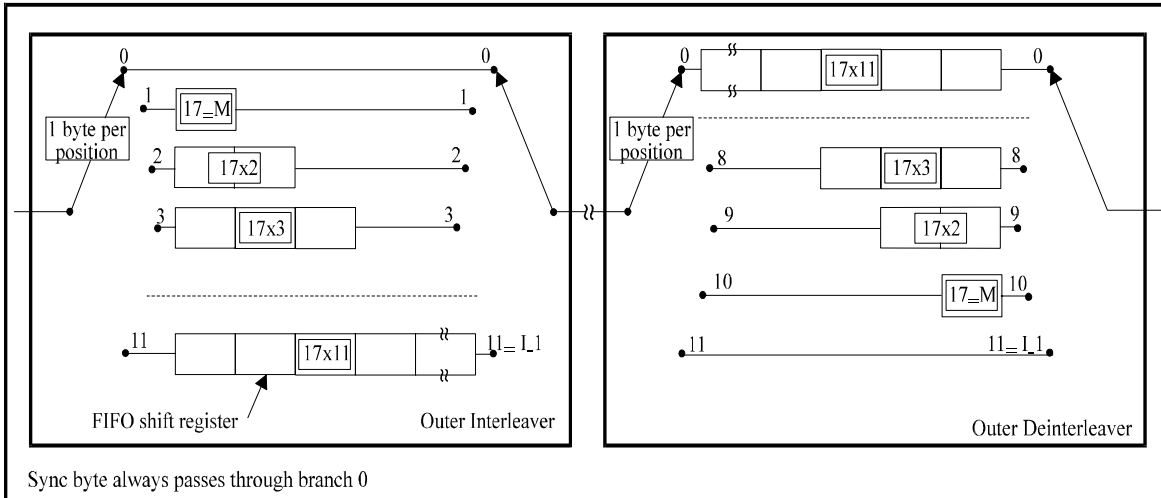
c) Reed-Solomon RS(204,188,8) error protected packets



d) Data structure after outer interleaving; interleaving depth I = 12 bytes

Σχ. 2.7 Στάδια Προσαρμογής, Τυχαιοποίησης, Εξωτερικής Κωδικοποίησης και Διεμπλοκής

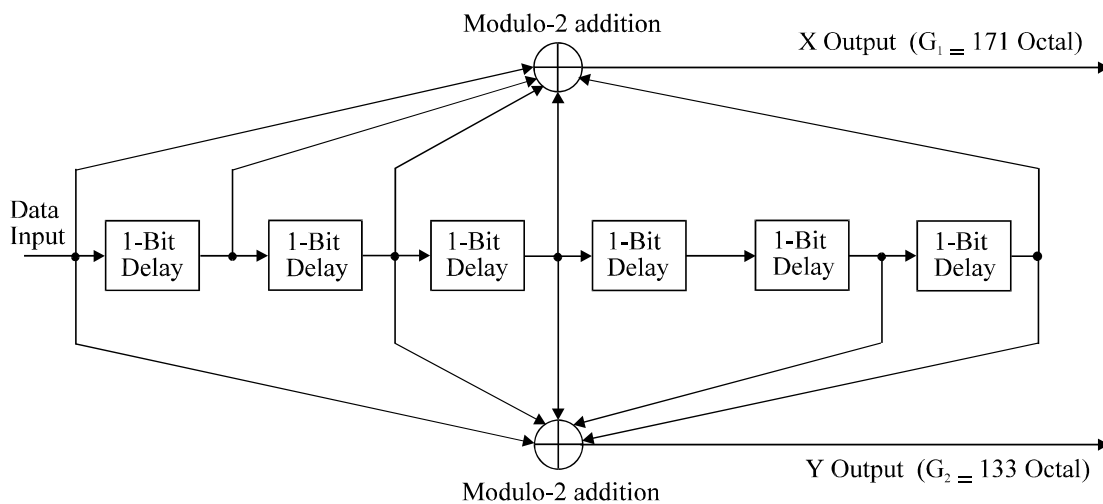
Μετά το στάδιο της εξωτερικής κωδικοποίησης, συνελικτική διεμπλοκή με βάθος $I=12$ (convolutional interleaving) εφαρμόζεται στα προστατευμένα από λάθη πακέτα και αυτό οδηγεί στη δομή δεδομένων του Σχ. 2.7 d. Η διαδικασία της συνελικτικής διεμπλοκής βασίζεται στη μέθοδο Forney η οποία είναι συμβατή με τη μέθοδο Ramsey τύπου III. Ο διεμπλοκέας (Σχ. 2.8) αποτελείται από $I=12$ κλάδους, κυκλικά συνδεδεμένους στην είσοδο της ροής των bytes στον διακόπτη εισόδου. Κάθε κλάδος αποτελεί έναν καταχωρητή μετατόπισης (shift register) First-In, First-Out (FIFO) του οποίου τα κελιά περιέχουν 1 byte και οι διακόπτες εισόδου και εξόδου είναι συγχρονισμένοι. Για λόγους συγχρονισμού τα bytes SYNC και δρομολογούνται πάντα στον κλάδο "0" του διεμπλοκέα.



Σχ. 2.8 Διάγραμμα Εξωτερικού Διεμπλοκέα και Αποδιεμπλοκέα

2.4.1.3. Εσωτερική Κωδικοποίηση και Διεμπλοκή

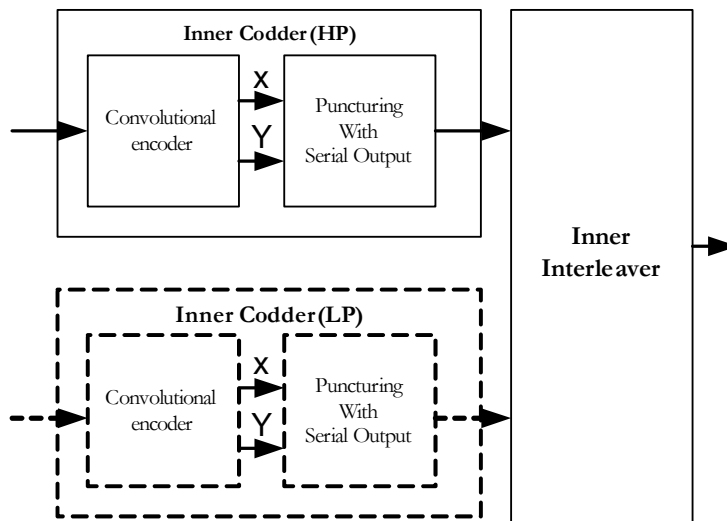
Το πρότυπο DVB-T επιτρέπει μία ευρεία επιλογή διάτρητων συνελικτικών κωδίκων (punctured convolutional codes) οι οποίοι βασίζονται σε έναν μητρικό συνελικτικό κώδικα με ρυθμό $1/2$. Αυτό επιτρέπει την επιλογή του πιο κατάλληλου επιπέδου προστασίας λαθών για μία δεδομένη υπηρεσία ή ένα ρυθμό μετάδοσης είτε στη μέθοδο της ιεραρχικής είτε της μη-ιεραρχικής μετάδοσης. Τα πολυώνυμα-γεννήτριες του μητρικού κώδικα είναι $G_1 = 171 \text{ OCT}$ για την έξοδο X και $G_2 = 133 \text{ OCT}$ για την έξοδο Y (Σχ. 2.8). Επιπρόσθετα με τον μητρικό κώδικα ρυθμού $1/2$ το πρότυπο επιτρέπει τη χρήση διάτρητων κωδίκων με ρυθμούς $2/3$, $3/4$, $5/6$ και $7/8$.



Σχ. 2.9 Μητρικός Συνελικτικός Κώδικας με Ρυθμό Κώδικα $1/2$

Στο επόμενο στάδιο μετά την έξοδο του εσωτερικού κωδικοποιητή τα δεδομένα υφίστανται εσωτερική διεμπλοκή (Σχ. 2.10) η οποία περιλαμβάνει διεμπλοκή σε επίπεδο bits και σε επίπεδο συμβόλων. Η ακριβής αντιστοιχία των bits εισόδου στα τελικά διαμορφωμένα σύμβολα έχει άμεση σχέση με το σχήμα διαμόρφωσης το οποίο αξιοποιείται.

Inner Coder and Interleaver



Σχ. 2.10 Εσωτερική Κωδικοποίηση και Εσωτερική Διεμπλοκή

Μετά τη διαδικασία της εσωτερικής διεμπλοκής παράγεται μία ακολουθία δυαδικών ψηφίων (bits) η οποία είναι ήδη οργανωμένη σε σύμβολα και τα δυνατά σχήματα διαμόρφωσης τα οποία μπορούν αξιοποιηθούν με βάση τις προδιαγραφές του προτύπου DVB-T (ETSI 300 744) [5] είναι: QPSK, 16QAM, 64QAM (2bits/symbol, 4bits/symbol και 6bits/symbol σε κάθε φέρον σήμα αντίστοιχα). Τα δεδομένα προς εμπομπή μεταδίδονται στη συνέχεια με τη χρήση της τεχνικής κωδικοποιημένης πολυπλεξίας με ορθογωνική διαίρεση συχνότητας (COFDM). Με τη χρήση της τεχνικής αυτής τα δεδομένα αντιστοιχίζονται με βάση το κατάλληλο σχήμα διαμόρφωσης σε κάθε ένα από τα χιλιάδες φέροντα σήματα τα οποία χρησιμοποιούνται για τη μετάδοσή τους σε όλη την περιοχή κάλυψης DVB-T.

2.4.2. Ωφέλιμο Bit Rate

Ο διαθέσιμος ωφέλιμος ρυθμός μετάδοσης για την εμπομπή δεδομένων σε συστήματα DVB-T δεν εξαρτάται από την κατάσταση λειτουργίας (2K ή 8K) του διαμορφωτή COFDM αλλά από το

σχήμα διαμόρφωσης που αξιοποιείται, από τη διάρκεια του διαστήματος φρούρησης και το ρυθμό της εσωτερικής κωδικοποίησης που επιλέγεται όπως φαίνεται από τον παρακάτω πίνακα (Σχ. 2.11)

Modulation	Code rate	Guard interval			
		1/4	1/8	1/16	1/32
QPSK	1/2	4,98	5,53	5,85	6,03
	2/3	6,64	7,37	7,81	8,04
	3/4	7,46	8,29	8,78	9,05
	5/6	8,29	9,22	9,76	10,05
	7/8	8,71	9,68	10,25	10,56
16-QAM	1/2	9,95	11,06	11,71	12,06
	2/3	13,27	14,75	15,61	16,09
	3/4	14,93	16,59	17,56	18,10
	5/6	16,59	18,43	19,52	20,11
	7/8	17,42	19,35	20,49	21,11
64-QAM	1/2	14,93	16,59	17,56	18,10
	2/3	19,91	22,12	23,42	24,13
	3/4	22,39	24,88	26,35	27,14
	5/6	24,88	27,65	29,27	30,16
	7/8	26,13	29,03	30,74	31,67

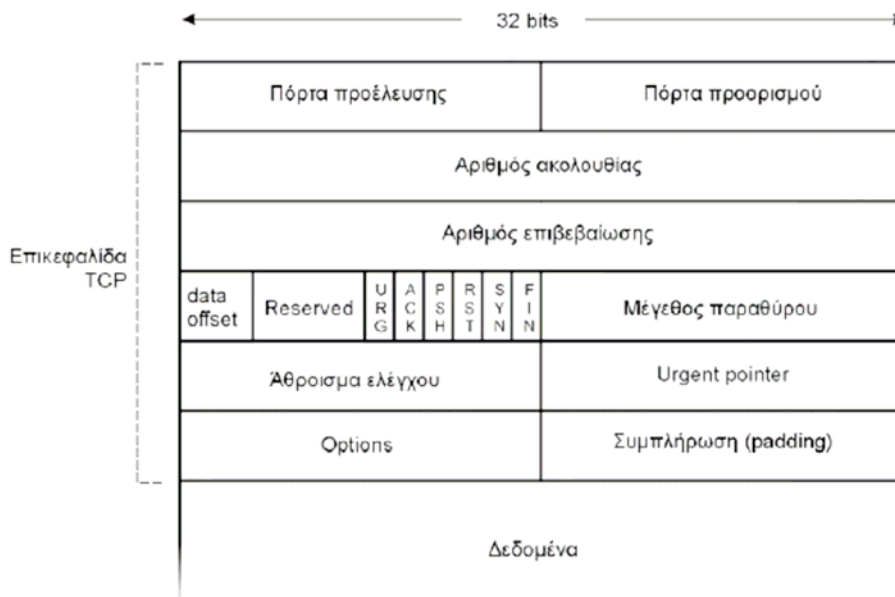
Σχ. 2.11 Ωφέλιμος Ρυθμός Μετάδοσης (Mbps) για όλους τους Συνδυασμούς Ρυθμού Κωδικοποίησης, Σχήματος Διαμόρφωσης και Διαστήματος Φρούρησης (Κανάλι 8MHz)

2.5. Πρωτόκολλα Μεταφοράς Δεδομένων

2.5.1. Το πρωτόκολλο TCP και η σημασία του

Το πρωτόκολλο μεταφοράς TCP (Transmission Control Protocol) είναι σήμερα το πιο δημοφιλές ίσως πρότυπο στις επικοινωνίες δεδομένων μέσω του Internet και όχι μόνο. Σχεδιάστηκε για πρώτη φορά το 1981 για χρήση στο αμερικανικό DARPA για να εξασφαλίσει αξιόπιστη επικοινωνία μεταξύ δύο τερματικών. Από τότε, έχει γίνει αντικείμενο πάμπολλων ερευνητικών προσπαθειών και έχει υποστεί πολλές προσθήκες και βελτιώσεις. Πολλές από τις βελτιώσεις αυτές θα υιοθετηθούν στα πλαίσια αυτής της πτυχιακής για να μεγιστοποιηθεί η επίδοση του TCP σε ασύμμετρα δίκτυα DVB-T.

Το TCP παρέχει υπηρεσίες μεταφοράς δεδομένων με σύνδεση (connection-oriented) και έλεγχο ροής (flow control) χρησιμοποιώντας το IP ως επίπεδο δικτύου. Οργανώνει τα δεδομένα σε τμήματα (segments) με επικεφαλίδα, της οποίας τα πεδία διευκολύνουν τις λειτουργίες ελέγχου ροής και αποφυγής λαθών, όπως φαίνεται και από το σχήμα που ακολουθεί.



Σχ. 2.12 Η Επικεφαλίδα TCP

Πολλές νοητές παράλληλες συνδέσεις μπορούν να εγκατασταθούν στην ίδια φυσική διαδρομή IP χάρη στις νοητές «πόρτες» (ports) των οποίων οι αριθμοί δηλώνονται στην αρχή της κεφαλίδας. Για ανίχνευση χαμένων πακέτων, το κάθε τμήμα αριθμείται με έναν συγκεκριμένο αριθμό ακολουθίας (sequence number) τον οποίο κάθε φορά ο αποστολέας αυξάνει κατά τον αριθμό των bytes που έχουν ως τώρα σταλεί επιτυχώς. Ο παραλήπτης απαντά δηλώνοντας στο πεδίο του αριθμού επιβεβαίωσης (acknowledgment number) τον αριθμό ακολουθίας του επόμενου τμήματος που αναμένει. Τμήματα για τα οποία η επιβεβαίωση καθυστερεί περισσότερο από ένα χρονικό διάστημα RTO (Retransmission TimeOut) επανειπέμπονται.

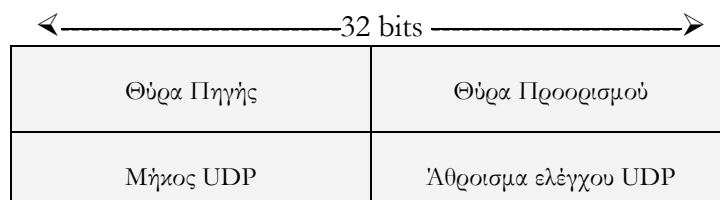
Προκειμένου να μην καθυστερείτε η ανταλλαγή δεδομένων από τη φυσική καθυστέρηση του καναλιού, το TCP επιχειρεί να στείλει έναν συγκεκριμένο όγκο δεδομένων χωρίς να περιμένει την επιβεβαίωση για το πρώτο τμήμα. Ο όγκος αυτός των δεδομένων που ανα πάσα στιγμή βρίσκονται ανεπιβεβαίωτα στο δίκτυο λέγεται παράθυρο (TCP window) ή παράθυρο συμφόρησης (congestion window). Το παράθυρο συμφόρησης αρχίζει με την τιμή του ενός τμήματος και αυξάνεται με κάθε επιτυχή επιβεβαίωση, αυξάνοντας έτσι και την ταχύτητα μετάδοσης. Η αύξηση είναι αρχικά ειθητική (διαδικασία αργής εκκίνησης - «slow start») και αργότερα γραμμική (διαδικασία αποφυγής συμφόρησης - «congestion avoidance»). Το παράθυρο που χρησιμοποιεί ο αποστολέας δεν μπορεί να υπερβεί την ονομαστική τιμή (receiver advertised window) που δηλώνει ο παραλήπτης σε κάθε

επιβεβαίωση χρησιμοποιώντας το αντίστοιχο πεδίο της επικεφαλίδας. Σε περίπτωση απώλειας πακέτου, ο αποστολέας μειώνει το παράθυρο συμφόρησης στο ήμισυ της προηγούμενης τιμής του, με αποτέλεσμα να μειώνεται αντίστοιχα και η ταχύτητα αποστολής.

Είναι γεγονός ότι το TCP είναι το πιο διαδεδομένο πρωτόκολλο στις διαδικτυακές συνδέσεις. Είναι χαρακτηριστικό ότι το 95% των bytes, το 90% των πακέτων και το 75% των συνδέσεων στο Internet σήμερα χρησιμοποιούν το TCP. [18], [19], [20], [21].

2.5.2. Το πρωτόκολλο UDP και η σημασία του

Η στοίβα πρωτοκόλλων του Internet υποστηρίζει επίσης ένα πρωτόκολλο μεταφοράς πληροφοριών χωρίς σύνδεση, το Πρωτόκολλο Δεδομενογραφμάτων Χρήστη UDP (User Datagram Protocol). Το UDP προσφέρει έναν τρόπο για να στέλνουν οι εφαρμογές ενθουλακωμένα ακατέργαστα δεδομενογραφήματα IP χωρίς να πρέπει να εγκαταστήσουν μια σύνδεση. Πολλές εφαρμογές πελάτη-εξυπηρετητή, που έχουν μία αίτηση και μία απόκριση, προτιμούν να χρησιμοποιήσουν το UDP παρά να μπου στον κόπο να εγκαταστήσουν και κατόπιν να απολύσουν μια σύνδεση.



Σχ. 2.13 Η Επικεφαλίδα UDP

Ένα τεμάχιο UDP αποτελείται από μια επικεφαλίδα των 8 byte (64 bit), ακολουθούμενη από δεδομένα. Η επικεφαλίδα φαίνεται στο παραπάνω σχήμα. Οι δύο θύρες εξυπηρετούν την ίδια λειτουργία όπως και στο TCP: την αναγνώριση των ακραίων σημείων στα μηχανήματα πηγής και προορισμού. Το πεδίο Μήκος UDP (UDP length) αφορά στην επικεφαλίδα 8 byte και στα δεδομένα. Το πεδίο Άθροισμα ελέγχου UDP (UDP checksum) περιλαμβάνει την ίδια ψευδο-εποκεφαλίδα με το TCP, την επικεφαλίδα UDP και τα δεδομένα UDP, συμπληρωμένα ώστε να σχηματίζουν έναν άρτιο αριθμό, αν χρειασθεί. Το Άθροισμα ελέγχου UDP είναι προαιρετικό και καταχωρείται ως 0 όταν δεν υπολογίζεται. (Το πραγματικά υπολογισμένο 0 καταχωρείται μ'όλα τα bit ίσα με 1, που είναι το ίδιο σε συμπλήρωμα ως προς 1). Το να μην χρησιμοποιηθεί είναι ανόητο, εκτός εάν η ποιότητα των δεδομένων δεν έχει μεγάλη σημασία (π.χ. η ψηφιοποιημένη φωνή).

Το UDP είναι ένα μη αξιόπιστο πρωτόκολλο, για εφαρμογές που δεν θέλουν τον έλεγχο της ακολουθίας ή της ροής του TCP και επιθυμούν να χρησιμοποιήσουν δικό τους. Επίσης χρησιμοποιείται ευρέως σε γρήγορες εφαρμογές και ερωταποκρίσεις, τύπου πελάτη-εξυπηρετητή, όπου η άμεση παράδοση είναι σπουδαιότερη από τη σωστή παράδοση, όπως π.χ. είναι η μετάδοση φωνής ή βίντεο. [19], [22].

3. ΠΑΡΟΥΣΙΑΣΗ ΔΙΚΤΥΟΥ ΠΑΡΟΧΗΣ ΑΜΦΙΔΡΟΜΩΝ ΔΙΑΔΡΑΣΤΙΚΩΝ ΥΠΗΡΕΣΙΩΝ

3.1. Εισαγωγή

Μια πλατφόρμα επίγειας ψηφιακής τηλεόρασης είναι δυνατόν να παρέχει εκτός από ψηφιακά τηλεοπτικά προγράμματα (MPEG-2) και υπηρεσίες IP. Το ενδογενές αυτό χαρακτηριστικό σε συνδυασμό με την ύπαρξη καναλιών επιστροφής (Reverse Path), επιτρέπει την υλοποίηση ευρυζωνικών υποδομών για την παροχή αμφίδρομων διαδραστικών υπηρεσιών. Στο κεφάλαιο αυτό γίνεται η παρουσίαση ενός προτεινόμενου δικτύου στο κανάλι επιστροφής του οποίου, υλοποιείται η τεχνολογία ISDN. Οι υπηρεσίες που παρέχονται από το σύστημα DVB-T της συγκεκριμένης πτυχιακής μπορούν να κατηγοριοποιηθούν σε δύο περιοχές:

- Διαδραστική Εκπομπή – Interactive Broadcasting. Περιλαμβάνει πραγματικά αμφίδρομες υπηρεσίες που συνδέονται ή είναι τελείως ανεξάρτητες με άλλες, broadcast υπηρεσίες. Απαιτείται κανάλι επιστροφής.
- Πρόσβαση στο Internet – Internet Access. Αυτή η περιοχή εστιάζεται στην παροχή πρόσβασης στο Internet μέσω μιας πλατφόρμας DVB.

3.2. Τοπολογία Πειραματικού Δικτύου Επίγειας Ψηφιακής Τηλεόρασης

Η τοπολογία του πειραματικού δικτύου DVB-T για την υλοποίηση της υβριδικής δικτυακής μητροπολιτικής υποδομής παρουσιάζεται στο Σχ. 3.1. Η συγκεκριμένη αρχιτεκτονική αποτελείται από τρία διακριτά μέρη με τα οποία υλοποιείται το περιβάλλον σύγκλισης του τομέα των δικτύων εκπομπής ευρείας κάλυψης και των δικτύων τηλεπικοινωνιών για την παροχή πολλαπλών υπηρεσιών διαφορετικού ρυθμού μετάδοσης σε αστικές και απομακρυσμένες περιοχές. Τα μέρη αυτά είναι μία πλατφόρμα επίγειας ψηφιακής τηλεόρασης η οποία αποτελείται από όλες τις κατάλληλες λειτουργικές μονάδες έτσι ώστε να είναι δυνατή η εκπομπή σημάτων COFDM και δύο ενδιάμεσοι κόμβοι διανομής με διαφορετικά κανάλια επιστροφής και δίκτυο πρόσβασης των τελικών χρηστών και των πάροχων των υπηρεσιών. Η δικτυακή υποδομή αυτή επιτρέπει τη μεταφορά κίνησης βασισμένη στα πρωτόκολλα TCP και UDP η οποία ενθυλακώνεται στην τελική ροή μεταφοράς

MPEG-2 με τη μορφή πακέτων IP σύμφωνα με την τεχνική MPE (Multi Protocol Encapsulation) [1]. Η διαδικασία της ενθυλάκωσης πραγματοποιείται από μία μονάδα που δρα ως Πύλη IP/DVB (IP-to-DVB Gateway) και η έξοδος της σε μορφή σήματος βασικής ζώνης οδηγείται στον διαμορφωτή COFDM ο οποίος αναλαμβάνει να δημιουργήσει το προς εκπομπή RF σήμα σύμφωνα με το πρότυπο του DVB-T [31]. Τα επεμπόμενα δεδομένα λαμβάνονται έτσι από τις διεπαφές λήψης σημάτων COFDM των ενδιάμεσων κόμβων διανομής και δρομολογούνται κατάλληλα στους τελικούς χρήστες και στους πάροχους των υπηρεσιών οι οποίοι είναι συνδεδεμένοι στο δίκτυο πρόσβασής τους.

Πιο αναλυτικά, η επικοινωνία του τελικού χρήστη με τον εξυπηρετητή δεδομένων επιτυγχάνεται όπως περιγράφεται αναλυτικά παρακάτω. Το τερματικό του τελικού χρήστη στέλνει τα δεδομένα επιστροφής (αιτήσεις και επιβεβαιώσεις) στον ενδιάμεσο κόμβο διανομής 2 μέσω του δικτύου ADSL στο οποίο είναι συνδεδεμένος με τη χρήση μίας συσκευής modem ADSL. Τα δεδομένα αυτά λαμβάνονται από την συσκευή IP-DSLAM και την διεπαφή του δικτύου πρόσβασης και δρομολογούνται κατάλληλα στην διεπαφή του καναλιού επιστροφής (dial up modem). Από εκεί μεταφέρονται στο κεντρικό σημείο εκπομπής DVB-T μέσω του δημόσιου τηλεφωνικού δικτύου μεταγωγής. Η πλατφόρμα DVB-T είναι εξοπλισμένη με μία διεπαφή για το Δίκτυο επιστροφής από τον κόμβο 2 η οποία έχει ρυθμιστεί κατάλληλα για την δρομολόγηση των πακέτων IP προς την Πύλη IP/DVB για την ενθυλάκωσή τους και τελικά την εκπομπή τους σε όλη την περιοχή κάλυψης του DVB-T. Τα δεδομένα επιστροφής λαμβάνονται με αυτόν τον τρόπο από τον δέκτη COFDM του ενδιάμεσου κόμβου διανομής 1 και δρομολογούνται στην διεπαφή του δικτύου πρόσβασής του. Στο συγκεκριμένο κόμβο το δίκτυο πρόσβασης υλοποιείται με την αξιοποίηση μίας ασύρματης πολυ-σημειακής ζεύξης (point-to-multipoint) σύμφωνα με το πρότυπο IEEE 802.11g [107], η οποία αναλαμβάνει να δρομολογήσει τα δεδομένα επιστροφής στον εξυπηρετητή δεδομένων. Η ταχεία εξάπλωση των ασύρματων δικτύων δεδομένων (WLANs) και ιδίως της οικογένειας προτύπων IEEE 802.11x έχει καταστήσει την τεχνολογία αυτή κυρίαρχη στον χώρο της ασύρματης μετάδοσης δεδομένων σε μικρές περιοχές με ρυθμούς μετάδοσης οι οποίοι μπορούν να αγγίξουν τα 54Mbps (full duplex) ή και ακόμα υψηλότερα. Δεδομένου ότι οι συσκευές αυτές είναι πλέον προσιτές εμπορικά, είναι πολύ διαδεδομένη η ανάπτυξη τοπικών μικροκυψελών (hot-spots) υψηλής ταχύτητας εντός των οποίων μπορούν να συνδεθούν αρκετοί τελικοί χρήστες αλλά και εν δυνάμει πάροχοι υπηρεσιών. Η χρήση έτσι μίας ευρυζωνικής τεχνολογίας για την υλοποίηση του δικτύου πρόσβασης στον ενδιάμεσο κόμβο διανομής 1 δίνει τη δυνατότητα στους εξυπηρετητές δεδομένων και πολυμέσων να στέλνουν δεδομένα με υψηλό ρυθμό μετάδοσης προς τους τελικούς χρήστες οι

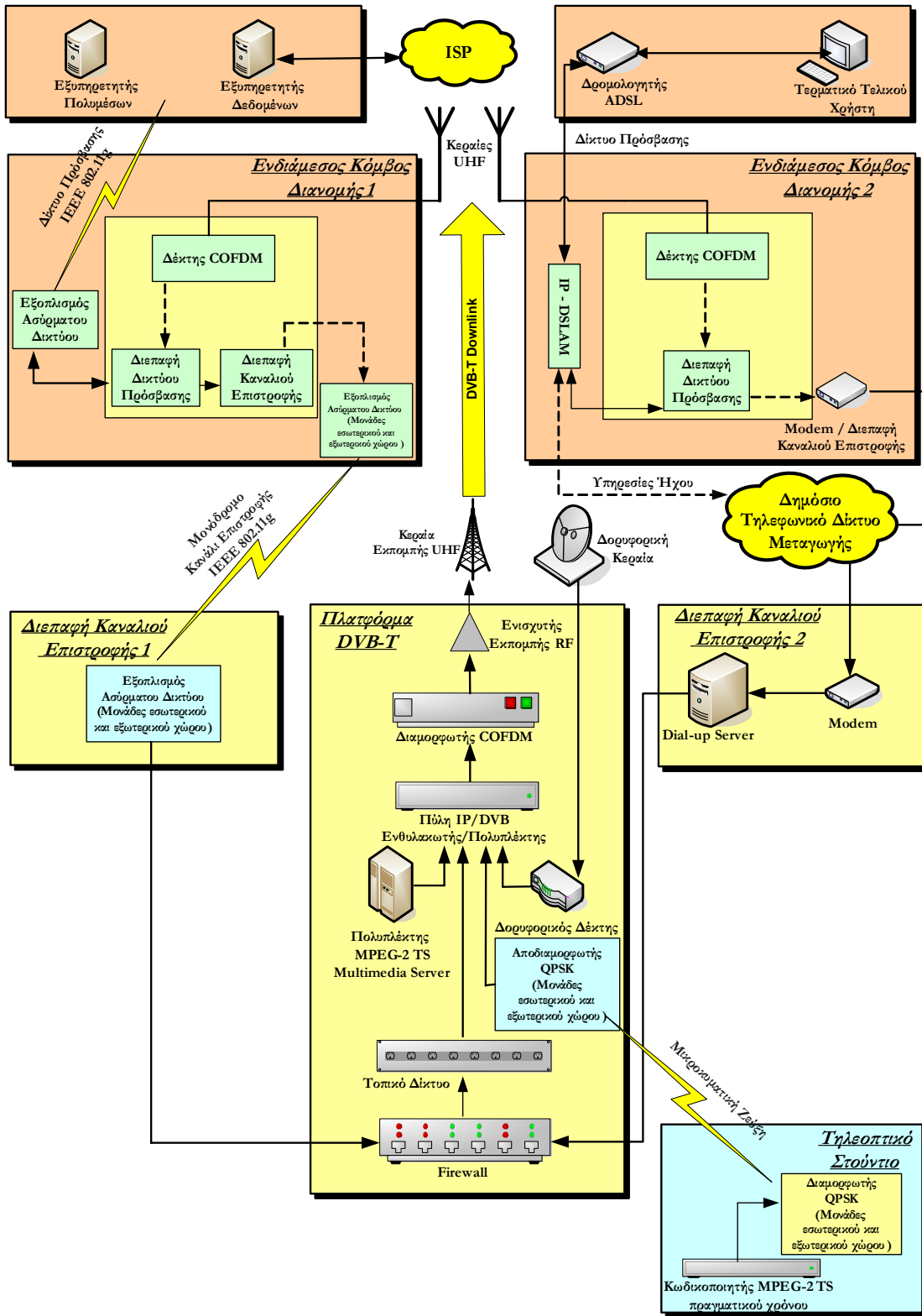
οποίοι βρίσκονται συνδεδεμένοι στο δίκτυο πρόσβασης του ενδιάμεσου κόμβου διανομής 2. Υπάρχει με αυτόν τον τρόπο η δυνατότητα ακόμα και οι τελικοί χρήστες οι οποίοι επιθυμούν να γίνουν “ενεργοί” και εν δυνάμει πάροχοι υπηρεσιών να στέλνουν τις υπηρεσίες τους αξιοποιώντας μία ουδέτερη πλατφόρμα DVB-T που αναλαμβάνει να παρέχει τα δεδομένα σε όλη την περιοχή εκπομπής. Η τεχνολογία ασύρματης τοπικής δικτύωσης που αξιοποιήθηκε για την υλοποίηση του δικτύου πρόσβασης στον κόμβο 1 λειτουργεί στην ελεύθερη (license-free) περιοχή συχνοτήτων των 2,4GHz, γνωστή και ως ζώνη ISM (Industrial, Scientific, Medical).

Οι κατάλληλες απαντήσεις του συγκεκριμένου εξυπηρετητή στα δεδομένα επιστροφής για την παροχή υπηρεσιών, βασισμένες σε κίνηση με βάση το πρωτόκολλο TCP, δρομολογούνται από το τερματικό του στον εξοπλισμό του ασύρματου δικτύου μέσω της κατάλληλης κάρτας (station adapter IEEE 802.11g) και από εκεί στον ενδιάμεσο κόμβο διανομής 1. Σε αυτό το σημείο του δικτύου η διεπαφή δικτύου πρόσβασης του κόμβου 1 δρομολογεί τα δεδομένα στην διεπαφή του καναλιού επιστροφής και από εκεί προς το κεντρικό σημείο εκπομπής. Το κανάλι επιστροφής του κόμβου 1 υλοποιείται σε αυτήν την περίπτωση με τη χρήση μία κατευθυντικής ασύρματης ζεύξης (point-to-point) με βάση το πρότυπο IEEE 802.11g [107], υψηλού ρυθμού μετάδοσης δεδομένων. Με αυτόν τον τρόπο η κίνηση TCP από τον εξυπηρετητή δεδομένων λαμβάνεται από τη διεπαφή καναλιού επιστροφής 1 της πλατφόρμας DVB-T η οποία αναλαμβάνει να την δρομολογήσει κατάλληλα στην πύλη IP/DVB μέσω του τοίχους προστασίας (firewall) ο οποίος δεν επιτρέπει την επιστροφή δεδομένων από το κεντρικό σημείο εκπομπής προς τους κόμβους μέσω των καναλιών επιστροφής. Στο επόμενο στάδιο μετά την ενθυλάκωση της κίνησης TCP με τη μορφή πακέτων IP, εκπέμπεται η κοινή ροή μεταφοράς DVB-T η οποία περιλαμβάνει εκτός από αυτήν και πολλαπλά ψηφιακά τηλεοπτικά προγράμματα από διαφορετικές πηγές. Τα δεδομένα έτσι λαμβάνονται από το δέκτη COFDM του κόμβου 2 και δρομολογούνται στην διεπαφή δικτύου πρόσβασης. Η συσκευή IP-DSLAM αναλαμβάνει στην πορεία να προωθήσει την κίνηση στον δρομολογητή ADSL και αυτός με τη σειρά του στο τερματικό του τελικού χρήστη.

Επιπλέον τα δεδομένα πολυμεσικού περιεχομένου κίνησης βασισμένης στο πρωτόκολλο UDP από τον εξυπηρετητή πολυμέσων δρομολογούνται στον τελικό χρήστη με τον ίδιο ακριβώς τρόπο μέσω της πλατφόρμας DVB-T. Η ροή δεδομένων UDP δρομολογείται στη διεπαφή δικτύου πρόσβασης του ενδιάμεσου κόμβου διανομής 1 μέσω της ασύρματης πολυσημειακής ζεύξης IEEE 802.11g και από εκεί στη διεπαφή καναλιού επιστροφής έτσι ώστε να είναι εφικτή η λήψη της από την πλατφόρμα DVB-T. Τα δεδομένα αυτά ενθυλακώνονται μέσα στην τελική ροή μεταφοράς MPEG-2 και εκπέμπονται σε όλη την περιοχή κάλυψης DVB-T. Λαμβάνονται έτσι από τον δέκτη

COFDM του ενδιάμεσου κόμβου διανομής 2 και δρομολογούνται στη διεπαφή δικτύου πρόσβασης. Τελικά λαμβάνονται από το τερματικό του τελικού χρήστη μέσω του δικτύου ADSL στο οποίο είναι συνδεδεμένος.

Η πλατφόρμα DVB-T (Σχ. 3.1) έχει επιπλέον τη δυνατότητα μετάδοσης πολλαπλών ψηφιακών τηλεοπτικών προγραμμάτων σε όλη την περιοχή εκπομπής. Στην εργαστηριακή υλοποίηση αξιοποιήθηκε η αναμετάδοση δύο ψηφιακών τηλεοπτικών προγραμμάτων από μία δορυφορική κεραία και έναν τοπικό τηλεοπτικό σταθμό αντίστοιχα καθώς και η εκπομπή ενός ψηφιακού τηλεοπτικού προγράμματος κωδικοποιημένου κατάλληλα και αποθηκευμένου στην συσκευή πολυπλέκτης MPEG-2 TS, multimedia server (Σχ. 3.1).



Σχ. 3.1 Τοπολογία Πειραματικού Δικτύου DVB-T

3.3. Χαρακτηριστικά Ηλεκτρονικών Υπολογιστών που Χρησιμοποιήθηκαν για την Υλοποίηση του Δικτύου

Στην εργαστηριακή υλοποίηση που χρησιμοποιήθηκε στη διπλωματική ο εξοπλισμός του τελικού χρήστη περιλαμβάνει έναν τυπικό Η/Υ, Pentium 4 / 3GHz με 512MB μνήμη RAM και λειτουργικό σύστημα LINUX με διανομή DEBIAN. Το τερματικό αυτό επικοινωνεί με μία κάρτα δικτύου 100Mbps και με τη χρήση καλωδίου τύπου RJ-45 με ένα modem ADSL το οποίο έχει επιπλέον δυνατότητες δρομολόγησης. Ο τύπος της συγκεκριμένης συσκευής modem είναι το Linksys WAG200G (ADSL Home Gateway) και διασυνδέεται με τηλεφωνικό καλώδιο (RJ-11) με τη συσκευή IP-DSLAM. Η συσκευή IP-DSLAM έχει ρυθμιστεί κατάλληλα έτσι ώστε ο ρυθμός μετάδοσης του καναλιού καθόδου να είναι 8Mbps και του καναλιού ανόδου 512Kbps. Ο τύπος της συσκευής αυτής είναι ZYXEL IP Express IE5-1000 και διασυνδέεται με καλώδιο δικτύου RJ-45 με τη διεπαφή δικτύου πρόσβασης του ενδιάμεσου κόμβου διανομής 2. Για την υλοποίηση του συγκεκριμένου κόμβου αξιοποιήθηκε ένας Η/Υ, Pentium 4 / 3GHz με 1GB μνήμη RAM και λειτουργικό σύστημα LINUX με διανομή DEBIAN. Ο ενδιάμεσος κόμβος διανομής 2 αποτελείται από τρεις διαφορετικές διεπαφές. Η πρώτη αφορά τη λήψη σημάτων COFDM, η δεύτερη τη διεπαφή του δικτύου πρόσβασης και η τρίτη τη διεπαφή του μονόδρομου καναλιού επιστροφής προς την πλατφόρμα DVB-T. Η διεπαφή λήψης σημάτων COFDM υλοποιήθηκε με τη χρήση ενός δέκτη DVB-T (WinTV-NOVA-T, USB2) ο οποίος χρησιμοποιεί το κατάλληλο λογισμικό για την αποθλάκωση σε πραγματικό χρόνο των πακέτων IP από τη λαμβανόμενη ροή μεταφοράς MPEG-2 και την παράδοσή τους στη στοίβα πρωτοκόλλων TCP/IP του λειτουργικού συστήματος. Για τη διεπαφή του δικτύου πρόσβασης αξιοποιήθηκε μία κάρτα δικτύου ρυθμού μετάδοσης 100Mbps και για τη διεπαφή του καναλιού επιστροφής ένα σειριακό ψηφιακό dial up ISDN modem τύπου US-Robotics Courier I-modem ISDN / v.34 με μέγιστη ονομαστική τιμή ρυθμού μετάδοσης από τον κόμβο προς το κεντρικό σημείο εκπομπής 64Kbps. Τα δεδομένα επιστροφής (αιτήσεις και επιβεβαιώσεις) της κίνησης TCP μεταφέρονται έτσι μέσω του δημόσιου τηλεφωνικού δικτύου μεταγωγής σε ένα ISDN modem το οποίο διασυνδέεται με σειριακή διεπαφή με έναν τυπικό Η/Υ (Pentium 2 / 4GHz, μνήμη RAM 128MB) με λειτουργικό σύστημα LINUX με διανομή DEBIAN ο οποίος έχει ρυθμιστεί κατάλληλα για την αξιοποίησή του ως εξυπηρετητής dial up και αποτελεί τη διεπαφή καναλιού επιστροφής 2 της πλατφόρμας DVB-T. Η διαδικτυακή κίνηση επιστροφής μεταφέρεται με αυτόν τον τρόπο στη μονάδα του τοίχους προστασίας (firewall) μέσω της κάρτας δικτύου (100Mbps) του εξυπηρετητή dial up. Το τοίχος προστασίας υλοποιήθηκε

με την αξιοποίηση μίας συσκευής ProCurve Switch 2424M τύπου HP J4093A η οποία ρυθμίστηκε κατάλληλα έτσι ώστε να μην είναι εφικτή η μεταφορά των δεδομένων από την πλατφόρμα DVB-T προς τον εξυπηρετητή dial up διατηρώντας έτσι μία μονόδρομη επικοινωνία χαμηλού ρυθμού μετάδοσης με ενεργό μόνο το κανάλι ανόδου από τον ενδιάμεσο κόμβο διανομής 2 προς την πλατφόρμα DVB-T.

Στη συσκευή του τοίχου προστασίας συνδέεται επίσης και η διεπαφή καναλιού επιστροφής 1 της πλατφόρμας DVB-T για την μεταφορά την εμπρόσθιας διαδικτυακής κίνησης TCP η οποία προέρχεται από τον εξυπηρετητή δεδομένων ο οποίος διασυνδέεται στο δίκτυο πρόσβασης του ενδιάμεσου κόμβου διανομής 1. Όλα τα δεδομένα τα οποία προέρχονται από τους δύο κόμβους με προορισμό την πλατφόρμα DVB-T προωθούνται από τη συσκευή του τοίχους προστασίας σε μία συσκευή Switch τύπου Level One RSW-0808TX με μέγιστο ρυθμό μετάδοσης 100Mbps και από εκεί δρομολογούνται στην Πύλη IP/DVB για την ενθυλάκωσή τους στην κοινή ροή μεταφοράς DVB-T και την τελική εκπομπή τους στην περιοχή κάλυψης του DVB-T. Η Πύλη IP/DVB υλοποιήθηκε με την αξιοποίηση της συσκευής τύπου AMBER THALES η οποία έχει επιπλέον δυνατότητες πολυπλεξίας πολλαπλών ροών μεταφοράς MPEG-2 για τη δημιουργία μίας τελικής κοινής ροής προς εκπομπή στα UHF. Η Πύλη IP/DVB ρυθμίστηκε κατάλληλα έτσι ώστε να είναι εφικτή η εκπομπή μέσα στο κοινό μπουκέτο DVB-T τριών ψηφιακών τηλεοπτικών προγραμμάτων από διαφορετικές πηγές με αθροιστικό ρυθμό μετάδοσης και για τα τρία περίπου 11Mbps. Το ένα ψηφιακό τηλεοπτικό πρόγραμμα το οποίο αναμεταδόθηκε είναι πραγματικού χρόνου και προέρχεται από τη δορυφορική κεραία λήψης η οποία έχει συνδεθεί με ένα δέκτη QPSK. Το δεύτερο ψηφιακό τηλεοπτικό πρόγραμμα προέρχεται από έναν τοπικό τηλεοπτικό σταθμό και κωδικοποιείται με τη χρήση ενός MPEG-2 TS encoder πραγματικού χρόνου. Με την αξιοποίηση μία μικροκυματικής ζεύξης η συγκεκριμένη ροή μεταφοράς μεταφέρεται από τον διαμορφωτή QPSK του τηλεοπτικού σταθμού στον αποδιαμορφωτή QPSK της πλατφόρμας DVB-T ο οποίος διασυνδέεται στην Πύλη IP/DVB. Το τρίτο ψηφιακό τηλεοπτικό πρόγραμμα είναι κωδικοποιημένο και αποθηκευμένο στην συσκευή του εξυπηρετητή πολυμέσων OPAL (Πολυπλέκτης MPEG-2 TS) η οποία διασυνδέεται με την Πύλη IP/DVB. Όπως αναφέρθηκε παραπάνω η συσκευή ενθυλάκωσης IP/DVB αποτελεί και έναν πολυπλέκτη MPEG-2 TS ο οποίος έχει τη δυνατότητα να πολυπλέκει πολλαπλές ροές ψηφιακών τηλεοπτικών προγραμμάτων για τη δημιουργία μίας κοινής τελικής ροής προς εκπομπή DVB-T. Έτσι τα τρία ψηφιακά τηλεοπτικά προγράμματα πολυπλέκονται μαζί και μεταδίδονται σε όλη την περιοχή εκπομπής μαζί με την διαδικτυακή κίνηση IP. Η Πύλη IP/DVB ρυθμίστηκε επιπλέον κατάλληλα έτσι ώστε να είναι δυνατή η εκπομπή δεδομένων multicasting

βασισμένα σε κίνηση πρωτοκόλλου UDP με μέγιστο ρυθμό μετάδοσης 1,5Mbps ενώ η κατανομή του ρυθμού μετάδοσης για τα δεδομένα βασισμένα σε κίνηση πρωτοκόλλου TCP επιλέχθηκε να είναι 8Mbps. Το τελικό επίγειο μπουκέτο εκπομπής το οποίο περιλάμβανε όλες τις πολυμεσιές και διαδικτυακές υπηρεσίες από την έξοδο της Πύλης IP/DVB διαμορφώθηκε έτσι με συνολικό ρυθμό μετάδοσης περίπου 21Mbps.

Η τελική ροή μεταφοράς η οποία περιέχει τα πολλαπλά ψηφιακά τηλεοπτικά προγράμματα και τα διαδικτυακά δεδομένα κίνησης IP (δεδομένα επιστροφής και εμπρόσθια διαδικτυακή κίνηση) που προέρχονται από τους ενδιάμεσους κόμβους διανομής, μεταφέρεται στη μονάδα διαμόρφωσης και εκπομπής DVB-T για την τελική μετάδοση στην περιοχή κάλυψης με τη μίας διεπαφής ASI και καλώδιο. Για την κατάλληλη διαμόρφωση και εκπομπή αξιοποιήθηκε ένας διαμορφωτής DVB-T τύπου UBS PT 5780 ο οποίος ρυθμίστηκε να χρησιμοποιεί σχήμα διαμόρφωσης 16QAM, ρυθμό κώδικα 7/8 και διάστημα φρούρησης (guard interval) ίσο με 1/32 του μήκους συμβόλου (Σχ. 3.2) και (Σχ. 3.3). Οι παράμετροι αυτές αντιστοιχούν σε ωφέλιμο ρυθμό μετάδοσης ίσο με 21,11Mbps (Σχ. 2.11). Η τελική εκπομπή RF και ενίσχυση πραγματοποιείται με τη χρήση ενός ενισχυτή με ισχύ εξόδου της τάξης των 3W και την αξιοποίηση μίας κεραίας μετάδοσης UHF (abe LB13-SA) στη συχνότητα 538MHz (κανάλι 29 στις συχνότητες UHF).



Σχ. 3.2 Ο Διαμορφωτής COFDM που Παρουσιάζει τα Στοιχεία της Διαμόρφωσης



Σχ. 3.3 Ο Διαμορφωτής COFDM που Παρέχει το DVB-T Stream στο Κανάλι 29 σε Επίπεδο Δύναμης 0dBm

Στην εργαστηριακή υλοποίηση, χρησιμοποιήθηκε επίσης, ο εξοπλισμός του εξυπηρετητή δεδομένων και του εξυπηρετητή πολυμέσων περιλαμβάνει από έναν τυπικό Η/Υ, ο κάθε ένας τύπου AMD 64bit / 2GHz με 512MB μνήμη RAM και λειτουργικό σύστημα LINUX με διανομή UBUNTU. Οι δύο αυτοί Η/Υ αξιοποιήθηκαν για την αποστολή των δεδομένων προς τον τελικό χρήστη. Η προτεινόμενη υλοποίηση του δικτυακού περιβάλλοντος σύγκλισης έχει ως στόχο την ταχεία διασύνδεση του τελικού χρήστη με το Internet μέσω κάποιου πάροχου διαδικτυακών υπηρεσιών (ISP). Οι μετρήσεις επιδόσεων εν τούτοις για την αξιολόγηση της δικτυακής υποδομής DVB-T με βάση την κίνηση TCP επιλέχθηκε να πραγματοποιηθούν με έναν τοπικό εξυπηρετητή (εξυπηρετητής δεδομένων) ούτως ώστε να μην επηρεάζονται από πιθανές συμφορήσεις (bottlenecks) εκτός του προτεινόμενου δικτύου κάτι που ενδεχομένως να συνέβαινε στην περίπτωση ενός απομακρυσμένου εξυπηρετητή. Οι δύο εξυπηρετητές που αξιοποιήθηκαν για τις μετρήσεις αξιολόγησης της απόδοσης διασυνδέθηκαν με τις κάρτες ασύρματου δικτύου τους (τύπου Netgear WG311V.3, PCI) στον εξοπλισμό πρόσβασης ασύρματου δικτύου (τύπου LINKSYS WAG54G) στον ενδιάμεσο κόμβο διανομής 1. Η ονομαστική τιμή του μέγιστου ρυθμού μετάδοσης αυτής της ασύρματης πολυσημειακής ζεύξης με βάση το πρότυπο IEEE 802.11g είναι της τάξης των 27Mbps και στα δύο κανάλια (ανόδου και καθόδου). Τα δεδομένα αποστολής μεταφέρονται έτσι από τον εξοπλισμό της ασύρματης πρόσβασης στη διεπαφή δικτύου πρόσβασης του ενδιάμεσου κόμβου διανομής 1 μέσω της κάρτας δικτύου (100Mbps) της συσκευής αυτής και ενός καλωδίου RJ-45. Ο

ενδιάμεσος κόμβος διανομής 1 αποτελείται και αυτός όπως και ο 2 από τρεις διαφορετικές διεπαφές και για την υλοποίησή του αξιοποιήθηκε ένας H/Y, AMD 64bit / 2GHz με 512MB μνήμη RAM και λειτουργικό σύστημα LINUX με διανομή UBUNTU. Η πρώτη διεπαφή του συγκεκριμένου κόμβου αφορά τη λήψη σημάτων COFDM, η δεύτερη τη διεπαφή του δικτύου πρόσβασης και η τρίτη τη διεπαφή του μονόδρομου καναλιού επιστροφής προς την πλατφόρμα DVB-T. Η διεπαφή λήψης σημάτων COFDM υλοποιήθηκε με τη χρήση ενός δέκτη DVB-T (WinTV-NOVA-T, USB2) και οι άλλες δύο με τη χρήση καρτών δικτύου ρυθμού μετάδοσης 100Mbps. Η εμπρόσθια διαδικτυακή κίνηση TCP από τους εξυπηρετητές δρομολογείται έτσι κατάλληλα μέσω του κόμβου και της διεπαφής καναλιού επιστροφής στον εξοπλισμό ασύρματου δικτύου του και από εκεί στην πλατφόρμα DVB-T. Η μεταφορά των δεδομένων πραγματοποιείται με αυτόν τον τρόπο μέσω της κατευθυντικής ασύρματης ζεύξης IEEE 802.11g της οποίας η μέγιστη ονομαστική τιμή του ρυθμού μετάδοσης είναι 27Mbps και για τα δύο κανάλια (ανόδου και καθόδου). Η συσκευή (εξοπλισμός ασύρματου δικτύου) η οποία αξιοποιήθηκε για τη λήψη των δεδομένων στην πλατφόρμα DVB-T διασυνδέεται με καλώδιο δικτύου RJ-45 στο τοίχος προστασίας. Το τοίχος προστασίας όπως αναφέρθηκε παραπάνω έχει ρυθμιστεί κατάλληλα έτσι ώστε να μην είναι εφικτή η μεταφορά των δεδομένων από την πλατφόρμα DVB-T προς τον ενδιάμεσο κόμβο διανομής 1 μέσω του εξοπλισμού ασύρματου δικτύου, διατηρώντας έτσι μία μονόδρομη επικοινωνία υψηλού ρυθμού μετάδοσης με ενεργό μόνο το κανάλι ανόδου από τον κόμβο 1 προς την πλατφόρμα.

Η συμπεριφορά και οι επιδόσεις της προτεινόμενης δικτυακής αρχιτεκτονικής DVB-T με βάση την κίνηση πρωτοκόλλου TCP, μετρήθηκαν και αξιολογήθηκαν στα πλαίσια μίας σειράς εργαστηριακών πειραμάτων τα οποία πραγματοποιήθηκαν με τη μαζική μεταφορά δεδομένων (TCP bulk data transfer) από τον εξυπηρετητή δεδομένων προς το τερματικό του τελικού χρήστη αξιοποιώντας το πρόγραμμα δημιουργίας διαδικτυακής κίνησης Iperf [114]. Το μετρητικό αυτό πρόγραμμα πραγματοποιεί μεταφορά εικονικών δεδομένων (dummy data) από ένα τερματικό σε ένα άλλο για μία συγκεκριμένη χρονική διάρκεια (180 δευτερόλεπτα στα πειράματα που θα περιγραφθούν) με σκοπό να προσδιοριστούν οι επιδόσεις και η συμπεριφορά των πρωτοκόλλων δικτύου πάνω από τα συγκεκριμένα κανάλια. Κατά τη διάρκεια μεταφοράς των δεδομένων στα πλαίσια μίας εργαστηριακής μέτρησης, τα πακέτα που ανταλλάσσονται στις διεπαφές της δικτυακής υποδομής DVB-T καταγράφονται από ένα πρόγραμμα σύλληψης πακέτων (tcpdump) [116] και οι κεφαλίδες τους αποθηκεύονται σε ένα τοπικό dump file. Τα dump files αναλύονται εις των υστέρων από έναν αναλυτή πρωτοκόλλου (protocol analyzer TCPTrace) [116] έτσι ώστε να εξαχθούν πληροφορίες σχετικά με το ρυθμό διαμεταγωγής (Throughput), τους χρόνους πλήρους διαδρομής

(Round-Trip Times) [117], το ποσοστό των ανεπιβεβαιωτών δεδομένων (Outstanding Data) στο δίκτυο, τον αριθμό των πακέτων επανειπομπής (Retransmitted Packets) και την πρόοδο του αριθμού ακολουθίας (Sequence Number) των δεδομένων και των επιβεβαιώσεων κατά τη μεταφορά. Είναι σημαντικό να τονιστεί ότι καμία ειδική ρύθμιση ή τροποποίηση στη στοίβα πρωτοκόλλων δεν απαιτείται από την πλευρά του τελικού χρήστη για να λάβει τα δεδομένα όταν διασυνδεθεί στην προτεινόμενη δικτυακή υποδομή DVB-T. Αρκετές φορές όμως η καθυστέρηση που υπεισέρχεται τόσο από τα κανάλια επιστροφής και ειδικά από αυτό της dial up σύνδεσης χαμηλού ρυθμού μετάδοσης, όσο και από τις λειτουργικές μονάδες επεξεργασίας του καναλιού εκπομπής, σε συνδυασμό με τον υψηλό ρυθμό μετάδοσης δεδομένων του DVB-T, μπορούν μερικές φορές να προκαλέσουν σχετικά μεγάλα γινόμενα καθυστέρησης-εύρους ζώνης (bandwidth-delay products). Έτσι είναι απαραίτητο να προσαρμοστεί το μέγιστο παράθυρο λήψης (TCP receive window) στο τερματικό του τελικού χρήστη έτσι ώστε να επιτευχθούν οι βέλτιστες δυνατές επιδόσεις. Στο επόμενο κεφάλαιο παρουσιάζονται αναλυτικά οι μετρήσεις αξιολόγησης της απόδοσης της δικτυακής υποδομής DVB-T βασισμένες σε κίνηση πρωτοκόλλου TCP στις οποίες παρατηρείται μεταξύ των άλλων και η σχέση μεταξύ του παραθύρου λήψης στο τερματικό του δέκτη και των διαφορετικών ρυθμών διαμεταγωγής σε κάθε περίπτωση.

Η συμπεριφορά και οι επιδόσεις της προτεινόμενης δικτυακής υποδομής DVB-T αξιολογήθηκαν επίσης με βάση την κίνηση πρωτοκόλλου UDP στα πλαίσια μίας σειράς εργαστηριακών πειραμάτων κατά τη διάρκεια των οποίων πραγματοποιήθηκε η μαζική μεταφορά δεδομένων από τον εξυπηρετητή πολυμέσων προς το τερματικό του τελικού χρήστη (Σχ. 3.1). Το πρόγραμμα το οποίο αξιοποιήθηκε για τη δημιουργία της κίνησης UDP ήταν σε αυτήν την περίπτωση το MGEN [118] και οι μετρήσεις πραγματοποιήθηκαν για χρονικό διάστημα 180 δευτερολέπτων με σκοπό να προσδιοριστεί η επίδοση της μητροπολιτικής υποδομής για την παροχή πολυμεσιών διαδικτυακών υπηρεσιών. Κατά τη διάρκεια πραγματοποίησης μίας εργαστηριακής μέτρησης τα πακέτα τα οποία μεταφέρονται από τον εξυπηρετητή προς τον τελικό χρήστη μέσω της πλατφόρμας DVB-T καταγράφονται από το πρόγραμμα σύλληψης πακέτων tcpdump και οι κεφαλίδες τους αποθηκεύονται σε ένα τοπικό dump file. Μετά την ανάλυση των dump files από τον αναλυτή πρωτοκόλλου (TCPTrace) εξάγονται πληροφορίες σχετικά με τη διακύμανση καθυστέρησης διαδοχικών πακέτων (Packet-to-packet Delay Variation) [119], τη μέση απόκλιση – εξομαλυμένη απόλυτη τιμή της διακύμανσης καθυστέρησης διαδοχικών πακέτων (Interarrival Jitter) [119] και τη μονόδρομη καθυστέρηση (One-way Delay) [120]. Για την

πραγματοποίηση της μέτρησης της μονόδρομης καθυστέρησης, γίνεται χρήση του NTP πρωτοκόλλου, προκειμένου να συγχρονιστούν τα ρολόγια των Server και Client.

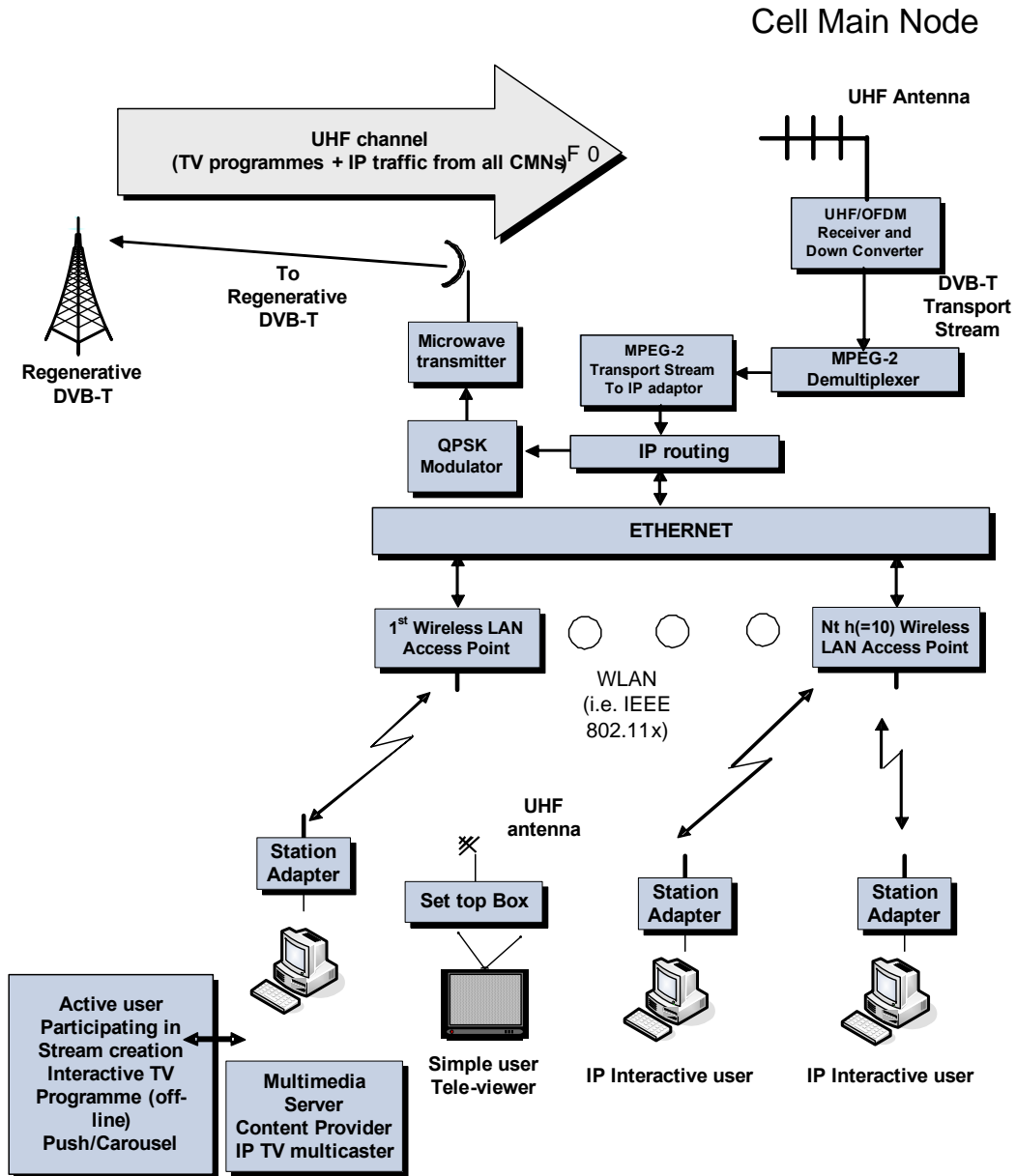
Η γενική τοπολογία του προτεινόμενου δικτύου (Σχ. 3.1) αποτελείται όπως έχει ήδη περιγραφεί από μία κεντρική πλατφόρμα DVB-T και δύο ενδιάμεσους κόμβους διανομής οι οποίοι μεσολαβούν στη διασύνδεση των τελικών χρηστών και των πάροχων των διαδικτυακών και πολυμεσικών υπηρεσιών της μητροπολιτικής αρχιτεκτονικής. Οι ενδιάμεσοι κόμβοι διανομής αξιοποιούν τις γενικές προδιαγραφές των προτύπων ETS 300 802 [80] και ETS 300 801 [81] για την υλοποίηση ασύμμετρων δικτυακών τοπολογιών μεταξύ των λειτουργικών μονάδων τους και των μονάδων της αλυσίδας εκπομπής της κεντρικής πλατφόρμας DVB-T. Όσον αφορά τη διεθυνσιοδότηση, για λόγους τάξης αλλά και σαφέστερης δρομολόγησης, η τοπολογία του συνολικού δικτύου χωρίστηκε σε τρία διαφορετικά υποδίκτυα με ιδιωτικές (private) IP διευθύνσεις. Το υποδίκτυο Α με διευθύνσεις στο χώρο 10.0.67.x περιλαμβάνει όλη την αλυσίδα εκπομπής DVB-T (Πύλη IP/DVB, τοπικό δίκτυο, τοίχος προστασίας), τις δύο διεπαφές καναλιού επιστροφής στην πλατφόρμα DVB-T και στους ενδιάμεσους κόμβους διανομής καθώς και τους δέκτες COFDM σε αυτούς. Το υποδίκτυο Β (δίκτυο πρόσβασης πάροχων υπηρεσιών) με διευθύνσεις στο χώρο 172.16.0.x περιλαμβάνει όλο το αμφίδρομο δίκτυο πρόσβασης των εξυπηρετητών του δικτυακού περιβάλλοντος σύγκλισης στον ενδιάμεσο κόμβο διανομής 1. Επίσης το υποδίκτυο Γ (δίκτυο πρόσβασης τελικών χρηστών) με διευθύνσεις στο χώρο 192.168.0.x περιλαμβάνει το αμφίδρομο δίκτυο πρόσβασης των τελικών χρηστών της μητροπολιτικής υποδομής DVB-T στον ενδιάμεσο κόμβο διανομής 2. Τα υποδίκτυα Β και Γ γεφυρώνονται μέσα στους κόμβους διανομής 1 και 2 αντίστοιχα με το υποδίκτυο Α με την αξιοποίηση των κατάλληλων εντολών δρομολόγησης του λειτουργικού συστήματός τους. Στην περίπτωση που η προτεινόμενη δικτυακή υποδομή DVB-T συνδεθεί στο Internet για την παροχή διαδικτυακών υπηρεσιών, χρησιμοποιείται ένας διακομιστής μεσολάβησης (proxy server) ο οποίος συνδέεται με τον εξυπηρετητή δεδομένων και αναλαμβάνει τη μετατροπή των ιδιωτικών IP διευθύνσεων σε δημόσιες (public) με την αξιοποίηση ενός μηχανισμού μετάφρασης διευθύνσεων (NAT- Network Address Translation). Έτσι η υιοθέτηση της μητροπολιτικής αυτής υποδομής και της υβριδικής δικτυακής αρχιτεκτονικής DVB-T προσφέρει μία προσιτή, βιώσιμη και πολύ αποδοτική λύση για την παροχή υπηρεσιών δεδομένων υψηλών ταχυτήτων στους τελικούς χρήστες οι οποίοι βρίσκονται σε απομακρυσμένες και αστικές περιοχές μέσα στην περιοχή εκπομπής.

3.4. Ενδιάμεσος Κόμβος Διανομής (Cell Main Node)

Κάθε CMN καθιστά δυνατή την πρόσβαση από έναν αριθμό τελικών χρηστών (οι οποίοι βρίσκονται γεωγραφικά κοντά στο CMN) σε IP υπηρεσίες που παρέχονται από το δίκτυο. Η επικοινωνία ανάμεσα στους χρήστες και στον αντίστοιχο κόμβο επιτυγχάνεται με την χρήση ασύρματων τεχνολογιών πρόσβασης, όπως WLAN. Κάθε κεντρικός κόμβος συγκεντρώνει όλη την IP κίνηση, η οποία προέρχεται από τους χρήστες του, και την προωθεί στο κεντρικό σημείο ευρυεκπομπής (regenerative DVB-T) με τη χρήση point-to-point ζεύξεων (uplinks). Η IP κίνηση που προέρχεται από κάθε CMN, λαμβάνεται από το κεντρικό σημείο ευρυεκπομπής, όπου φιλτράρεται, αναγεννάται και πολυπλέκεται με ψηφιακά τηλεοπτικά προγράμματα (που προέρχονται από τους φορείς ευρυεκπομπής – TV Studio), σε μία κοινή ροή μεταφοράς (Transport Stream), η οποία αποτελεί το τελικό μπουκέτο DVB-T (DVB-T bouquet). Η αναγεννημένη κίνηση εκπέμπεται σε ένα κανάλι UHF (π.χ. κανάλι 29), σύμφωνα με το πρότυπο DVB-T. Κάθε χρήστης λαμβάνει τις κατάλληλες IP απαντήσεις (replies) μέσω του αντίστοιχου CMN, ενώ λαμβάνει τα ψηφιακά τηλεοπτικά προγράμματα απευθείας μέσω του καναλιού UHF.

Η κυβελωτή προσέγγιση η οποία έχει υιοθετηθεί, χρησιμοποιεί το DVB-T κανάλι μεταφοράς ως δικτυακό κορμό backbone, ο οποίος διασυνδέει όλους τους κόμβους διανομής (CMN) που βρίσκονται μέσα στην περιοχή ευρυεκπομπής (broadcasting area). Έτσι, δημιουργείται ένας μοναδικός κορμός Ethernet, που είναι διαθέσιμος στους τελικούς χρήστες (μέσω του κατάλληλου κόμβου – CMN). Η IP κίνηση του δικτύου Ethernet παρέχεται μέσω του συρμού DVB-T.

Η γενική αρχιτεκτονική ενός CMN που χρησιμοποιεί την WLAN τεχνολογία για την υλοποίηση του καναλιού επιστροφής παρουσιάζεται στο Σχ. 3.4. Το συγκεκριμένο τμήμα του δικτύου υλοποιεί το δίκτυο πρόσβασης σύμφωνα με το πρότυπο IEEE 802.11b, κάνοντας χρήση τεχνικών ευρέου φάσματος (Spread Spectrum). Ένα τέτοιο δίκτυο επιτρέπει την point-to-multipoint επικοινωνία ανάμεσα στο CMN και στους τελικούς χρήστες.



Σχ. 3.4 Αρχιτεκτονική CMN

Οι ενεργοί και διαδραστικοί χρήστες δημιουργούν κίνηση IP, η οποία μεταφέρεται μέσω του αντίστοιχου κόμβου στο κεντρικό σημείο ευρυεκπομπής (regenerative DVB-T) και από εκεί σε όλη την πόλη μέσω του καναλιού των UHF.

3.4.1. Απαιτήσεις Υλικού (Hardware Requirements)

Ο ενδιάμεσος κόμβος διανομής (CMN) που χρησιμοποιείται στο δίκτυο της επίγειας ψηφιακής τηλεόρασης έχει υλοποιηθεί με τη χρήση ενός ηλεκτρονικού υπολογιστή Σχ. 3.5 και των κατάλληλων μονάδων hardware και software. Ο Η/Υ απαιτεί ως ελάχιστα τεχνικά χαρακτηριστικά έναν επεξεργαστή Pentium 3, ένα σκληρό δίσκο χωρητικότητας 10GB και μία μνήμη RAM 256MB. Ο κόμβος για την σωστή λειτουργία του απαιτεί επίσης τρία δικτυακά interfaces τα οποία περιγράφονται παρακάτω.

Η πλατφόρμα DVB-T εκπέμπει μία κοινή ροή μεταφοράς πακέτων MPEG-2 TS μέσα στην οποία έχει ενθυλακωθεί Διαδικτυακή κίνηση πακέτων IP στο κανάλι 29 της μπάντας συχνοτήτων UHF. Η συγκεκριμένη ροή μεταφοράς λαμβάνεται από την κάρτα επίγειας ψηφιακής τηλεόρασης που έχει τοποθετηθεί στον ενδιάμεσο κόμβο διανομής και έτσι με αυτόν το τρόπο υπάρχει η δυνατότητα λήψης της κίνησης πακέτων IP από το interface του DVB-T.

Η κάρτα αυτή διαδοχικά προωθεί τα πακέτα προς την κάρτα Ethernet του δικτύου πρόσβασης των τελικών χρηστών, όπου απορρίπτονται όσα πακέτα δεν αντιστοιχούν στο συγκεκριμένο υποδίκτυο. Η βασική λειτουργία της κάρτας δικτύου πρόσβασης είναι να προωθεί την κίνηση προς τους τελικούς χρήστες μέσω ενός δικτύου Ethernet. Με αυτόν τον τρόπο οι τελικοί χρήστες έχουν τη δυνατότητα να λάβουν τα πακέτα και να στείλουν τις επιβεβαιώσεις (ACKs) της συγκεκριμένης κίνησης μέσω του δικτύου πρόσβασης και του δικτύου επιστροφής. Η κίνηση των πακέτων αυτών προωθείται μέσω της κάρτας του δικτύου πρόσβασης στην κάρτα του δικτύου επιστροφής που διασυνδέει τον ενδιάμεσο κόμβο διανομής με το κεντρικό σημείο ειπομπής.



Σχ. 3.5 Ενδιάμεσος Κόμβος Διανομής (CMN), Κάρτα Επίγειας Ψηφιακής Τηλεόρασης, Κεραία Λήψης

Όπως έχουμε προαναφέρει για την υλοποίηση του δικτύου μας κάναμε χρήση δυο ενδιαμέσων κόμβων διανομής ,έναν με δίκτυο επιστροφής ISDN και έναν άλλο με ασύρματο 802.11b/g. Ο κάθε CMN αναλαμβάνει να προωθήσει τα δεδομένα στο αντίστοιχο υποδίκτυο. Και οι δυο CMN χρησιμοποιούν σαν interface της λήψης από μία κάρτα επίγειας ψηφιακής τηλεόρασης DVB-T. Το μοντέλο της κάρτας αυτής είναι η WinTV-NOVA-T-USB-2 (Σχ. 3.63.6) που υποστηρίζεται από τον πυρήνα του λειτουργικού συστήματος Linux που εγκαταστάθηκε στους Η/Υ. Η κάρτα αυτή ονομάστηκε dnb_0 και χρησιμοποιείται για τη λήψη της κίνησης που εκπέμπεται από το κεντρικό σημείο εκπομπής και περιέχει τις IP απαντήσεις για τους τελικούς χρήστες. Απαραίτητη για τη λήψη είναι η χρήση μίας κεραίας UHF εσωτερικού χώρου.



Σχ. 3.6 Κάρτα Επίγειας Ψηφιακής Τηλεόρασης WinTV-NOVA-T-USB-2

Άλλο κοινό χαρακτηριστικό των δύο CMN είναι το interface του δικτύου πρόσβασης το οποίο υλοποιείται με τη χρήση καρτών δικτύου, eth_1, τύπου Intel(R) Pro/100 s Desktop Adapter (Σχ. 3.7) οι οποίες λαμβάνουν την προωθημένη Διαδικτυακή κίνηση από την κάρτα επίγειας ψηφιακής τηλεόρασης και την προωθούν στους τελικούς χρήστες του αντίστοιχου υποδικτύου. Οι κάρτες αυτές επίσης συγκεντρώνουν όλη την IP κίνηση, που προέρχεται από τους τελικούς χρήστες του κάθε CMN και την προωθούν προς το άλλο δυκτιακό interface του αντίστοιχου CMN.



Σχ. 3.7 Κάρτα Δικτύου, Intel(R) Pro/100 s Desktop Adapter

Το τρίτο δικτυακό interface (δύκτιο επιστροφής), το οποίο χρησιμοποιείται για την αποστολή των πακέτων προερχόμενα από τους χρήστες του αντίστοιχου υποδικτύου από τον κεντρικό κόμβο διανομής προς το σταθμό εκπομπής, διαφέρει στους δύο CMN. Ο πρώτος CMN μέσω ενός ψηφιακού ISDN modem το οποίο κάνει χρήση του πρωτοκόλλου PPP στέλνει τα IP δεδομένα σε έναν Dial-up Server ο οποίος βρίσκεται στον κεντρικό σταθμό εκπομπής. Το modem που χρησιμοποιήθηκε είναι της ISDN Network Termination Units της INTRACOM (Σχ. 3.8)



Σχ. 3.8 ISDN Network Termination Unit της INTRACOM

Στον δεύτερο CMN για το κανάλι επιστροφής χρησιμοποιήσαμε μια ασύρματη on-board κάρτας η οποία κάνοντας χρήση του πρωτοκόλλου 802.11 αποστέλλει τα ACK στον κεντρικό σταθμό βάσης. Η κάρτα που χρησιμοποιήσαμε για να πετύχουμε τη ασύρματη ζεύξη ήταν της Level-One (Σχ. 3.9)



Σχ. 3.9 Level-One Wireless Lan PCI Adapter 11g

3.4.2. Απαιτήσεις Λογισμικού (Software Requirements)

Οι ενδιαμέσοι κόμβοι διανομής έχουν υλοποιηθεί με βάση το λειτουργικό σύστημα των Linux το οποίο είναι ένα ελεύθερου "τύπου-Unix" λειτουργικό σύστημα και αποτελεί μία ανεξάρτητη POSIX υλοποίηση (συλλογικό όνομα μιας οικογένειας σχετικών προτύπων που διευκρινίζονται από το IEEE για να καθορίσουν τη διεπαφή προγραμματισμού εφαρμογής (API) για λογισμικό συμβατό με τις παραλλαγές του λειτουργικού συστήματος Unix). Στα χαρακτηριστικά του συμπεριλαμβάνει πολλαπλές διεργασίες, πραγματικό πολυχρηστικό περιβάλλον, εικονική μνήμη, κοινές βιβλιοθήκες, φόρτωση απαιτήσεων, δικτύωση TCP/IP και πολλά άλλα χαρακτηριστικά που δικαιολογούν τον τίτλο "τύπου-Unix". Ο πηγαίος κώδικας του λειτουργικού συστήματος Linux είναι διαθέσιμος σε όλους τους χρήστες ενώ για τη συγκεκριμένη υλοποίηση του ενδιαμέσου κόμβου διανομής χρησιμοποιήθηκε η έκδοση Debian ETCH.

Πριν την εγκατάσταση των εντολών για τη σωστή δρομολόγηση των πακέτων IP έπρεπε να δοθούν οι IP διευθύνσεις των τριών δικτυακών interfaces (dwb_0, eth_1 και eth_0) σε κάθε CMN. Στην περίπτωση του CMN που κάνει χρήση του δικτύου επιστροφής modem 56k η κατανομή των ip είναι η παρακάτω. Η dwb_0, η οποία λαμβάνει τη ροή μεταφοράς MPEG-2 TS που εκπέμπεται από το κεντρικό σημείο εκπομπής που περιέχει τις Διαδικτυακές και πολυμεσιές υπηρεσίες για τους τελικούς χρήστες, τέθηκε η διεύθυνση IP 14.14.14.14. Το interface του δικτύου πρόσβασης υλοποιήθηκε με την κάρτα eth_1. Το συγκεκριμένο interface λαμβάνει την προωθημένη κίνηση με τις δικτυακές υπηρεσίες από την κάρτα επίγειας ψηφιακής τηλεόρασης και την προωθεί έπειτα στους τελικούς χρήστες του υποδικτύου που υλοποιείται με κάποια τεχνολογία δικτύου πρόσβασης. Το συγκεκριμένο interface συγκεντρώνει επίσης όλη την IP κίνηση (αιτήσεις, επιβεβαιώσεις) που προέρχεται από τους τελικούς χρήστες του CMN και την προωθεί προς την άλλη κάρτα δικτύου

eth_0. Έτσι τέθηκε ως διεύθυνση IP η 192.168.0.1 και ως Subnet Mask η 255.255.255.0 ανάλογα με το υποδίκτυο των τελικών χρηστών. Πρέπει να σημειωθεί ότι δε χρειάζεται να δοθεί συγκεκριμένη Gateway διεύθυνση και αυτό γιατί η κίνηση των πακέτων δρομολογείται με βάση συγκεκριμένων εντολών δρομολόγησης (routing commands). Το τρίτο interface έχει υλοποιηθεί με ένα ψηφιακό modem το οποίο προωθεί όλη την IP κίνηση (αιτήσεις, επιβεβαιώσεις) που προέρχεται από την κάρτα eth_1 προς το κεντρικό σημείο εκπομπής. Στο modem αυτό τέθηκε ως IP διεύθυνση η 10.0.67.20 , ως Subnet Mask η 255.255.255.0 και ως Default Gateway η 10.0.67.1 η οποία αποτελεί την IP διεύθυνση του Cisco router (firewall) που βρίσκεται στο κεντρικό σημείο εκπομπής της πλατφόρμας DVB-T.

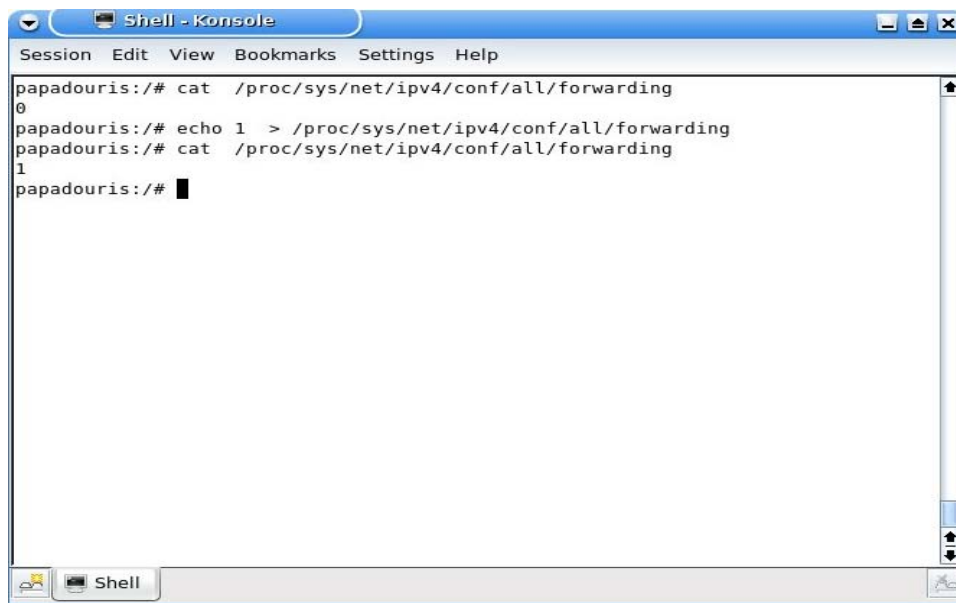
Όσον αφορά το CMN που κάνει χρήση του ασύρματου δικτύου επιστροφής 802.11g η κατανομή των ip είναι η παρακάτω. Η dnb_0, η οποία λαμβάνει τη ροή μεταφοράς MPEG-2 TS που εκπέμπεται από το κεντρικό σημείο εκπομπής που περιέχει τις Διαδικτυακές και πολυμεσιές υπηρεσίες για τους τελικούς χρήστες, τέθηκε η διεύθυνση IP 13.13.13.13. Το interface του δικτύου πρόσβασης υλοποιήθηκε με την κάρτα eth_1. Το συγκεκριμένο interface λαμβάνει την προωθημένη κίνηση με τις δικτυακές υπηρεσίες από την κάρτα επιλογής ψηφιακής τηλεόρασης και την προωθεί έπειτα στους τελικούς χρήστες του υποδικτύου που υλοποιείται με κάποια τεχνολογία δικτύου πρόσβασης. Το συγκεκριμένο interface συγκεντρώνει επίσης όλη την IP κίνηση (αιτήσεις, επιβεβαιώσεις) που προέρχεται από τους τελικούς χρήστες του CMN και την προωθεί προς την άλλη κάρτα δικτύου eth_0. Έτσι τέθηκε ως διεύθυνση IP η 172.16.0.1 και ως Subnet Mask η 255.255.255.0 ανάλογα με το υποδίκτυο των τελικών χρηστών. Πρέπει να σημειωθεί ότι δε χρειάζεται να δοθεί συγκεκριμένη Gateway διεύθυνση και αυτό γιατί η κίνηση των πακέτων δρομολογείται με βάση συγκεκριμένων εντολών δρομολόγησης (routing commands). Το τρίτο interface έχει υλοποιηθεί με μια ασύρματη pci κάρτα η οποία προωθεί όλη την IP κίνηση (αιτήσεις, επιβεβαιώσεις) που προέρχεται από την κάρτα eth_1 προς το κεντρικό σημείο εκπομπής. Στην κάρτα αυτή τέθηκε ως IP διεύθυνση η 10.0.67.99, ως Subnet Mask η 255.255.255.0 και ως Default Gateway η 10.0.67.1 η οποία αποτελεί την IP διεύθυνση του Cisco router (firewall) που βρίσκεται στο κεντρικό σημείο εκπομπής της πλατφόρμας DVB-T.

3.5. Η συμπεριφορά του CMN σε δικτυακή κίνηση

Στα πλαίσια της υλοποίησης του ενδιάμεσου κόμβου διανομής πρέπει να προηγηθεί η εκτέλεση κάποιων εντολών που θα βοηθήσουν στην δρομολόγηση των πακέτων IP έτσι ώστε να εξασφαλιστεί η σωστή μεταφορά της Διαδικτυακής κίνησης του δικτύου. Παρακάτω παρουσιάζεται αναλυτικά η υλοποίηση του CMN που κάνει χρήση του ασύρματου δικτύου επιστροφής. Ομοίως υλοποιείται και ο CMN που κάνει χρήση ISDN καναλιού επιστροφής.

3.5.1. Μετάδοση δεδομένων με την αξιοποίηση κίνησης TCP/IP και UDP

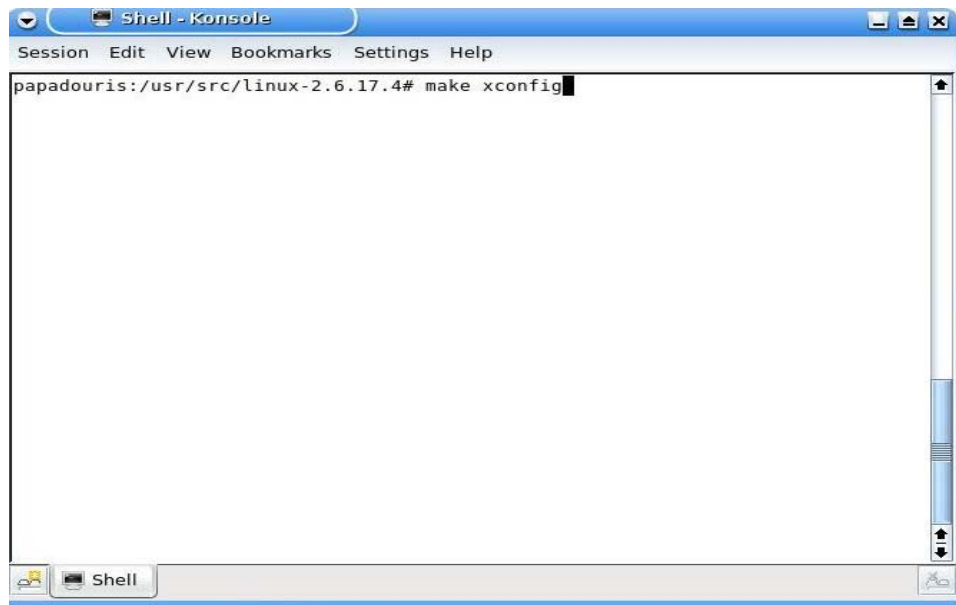
Για να μπορέσουν να οριστούν οι εντολές δρομολόγησης (routes), βασική προϋπόθεση είναι να ενεργοποιηθεί το **ip_forwarding**, το οποίο είναι μία εφαρμογή που προωθεί τα πακέτα μεταξύ διαφορετικών interfaces (dwb_0 → eth_1 και eth_1 → eth_0). Πρώτα, πρέπει να γίνει ο έλεγχος για το αν έχει ενεργοποιηθεί η συγκεκριμένη εντολή, πληκτρολογώντας την σε ένα terminal των linux. Εφόσον το αποτέλεσμα της είναι 0, όπως φαίνεται στην πρώτη εντολή του screenshot στο Σχ. 3.10, στο terminal, τότε είναι απενεργοποιημένη. Αν το αποτέλεσμα είναι 1 τότε είναι ενεργοποιημένη και την αφήνουμε ως έχει. Για την ενεργοποίησή της πρέπει να πληκτρολογηθεί η δεύτερη εντολή που φαίνεται στο screenshot του ίδιου σχήματος. Τέλος, επαναλαμβάνεται ο έλεγχος, όπου φαίνεται πως ενεργοποιήθηκε κι έγινε 1.



```
Shell - Konsole
Session Edit View Bookmarks Settings Help
papadouris:~# cat /proc/sys/net/ipv4/conf/all/forwarding
0
papadouris:~# echo 1 > /proc/sys/net/ipv4/conf/all/forwarding
papadouris:~# cat /proc/sys/net/ipv4/conf/all/forwarding
1
papadouris:~#
```

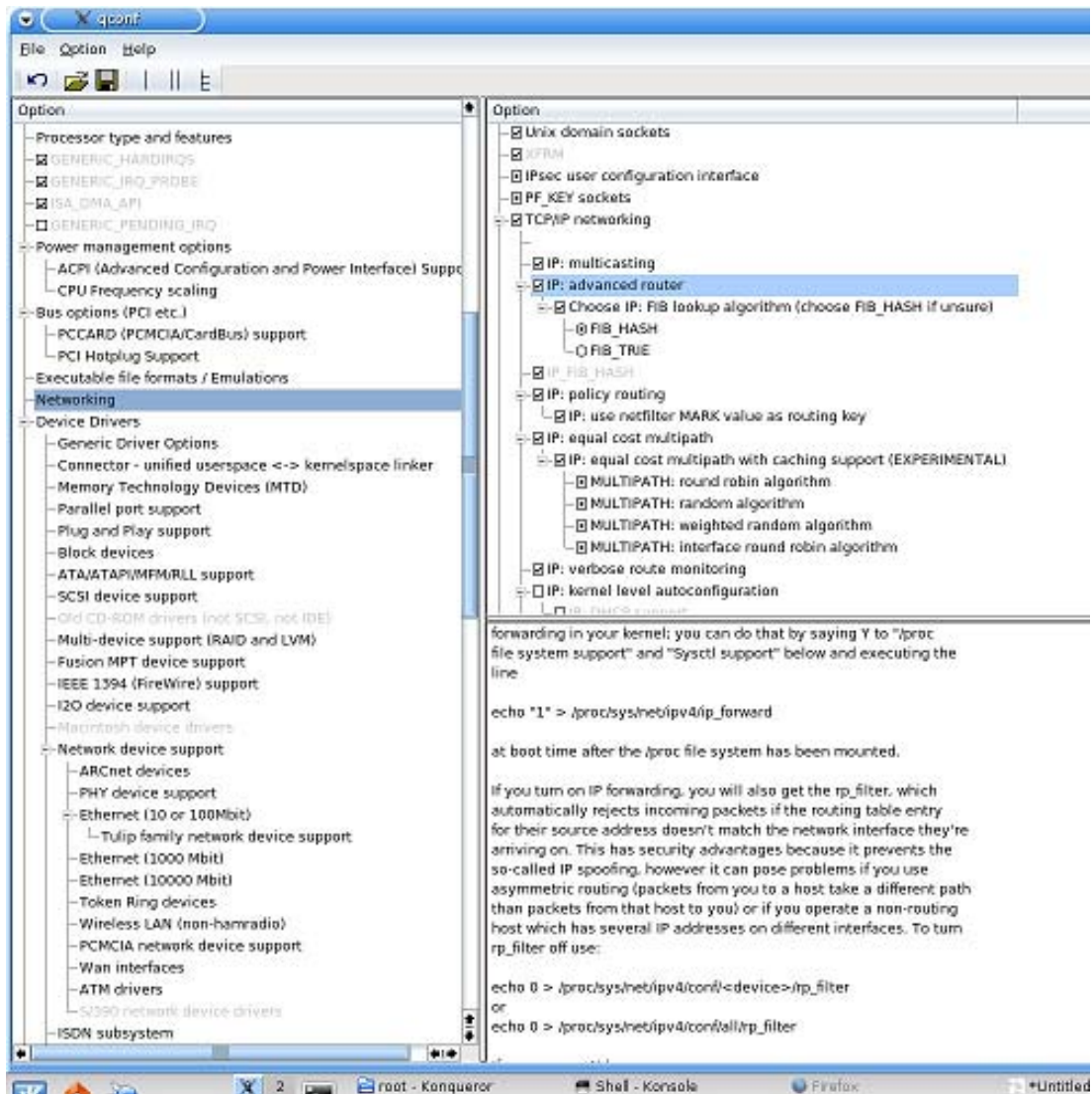
Σχ. 3.10 Εντολή ip_forwarding

Άλλη εντολή που χρησιμοποιείται είναι η εντολή **xconfig** που παριστάνει μια γραφική προσομοίωση του πυρήνα του λειτουργικού συστήματος και παρουσιάζει όλα τα στοιχεία του. Αρχικά, πρέπει να γίνει είσοδος στο CMN ως root στο φάκελο με τον πηγαίο κώδικα του πυρήνα μας. Έπειτα πληκτρολογείται η αντίστοιχη εντολή ώστε να ενεργοποιηθεί η xconfig εφαρμογή όπως φαίνεται στο παρακάτω Σχ. 3.11



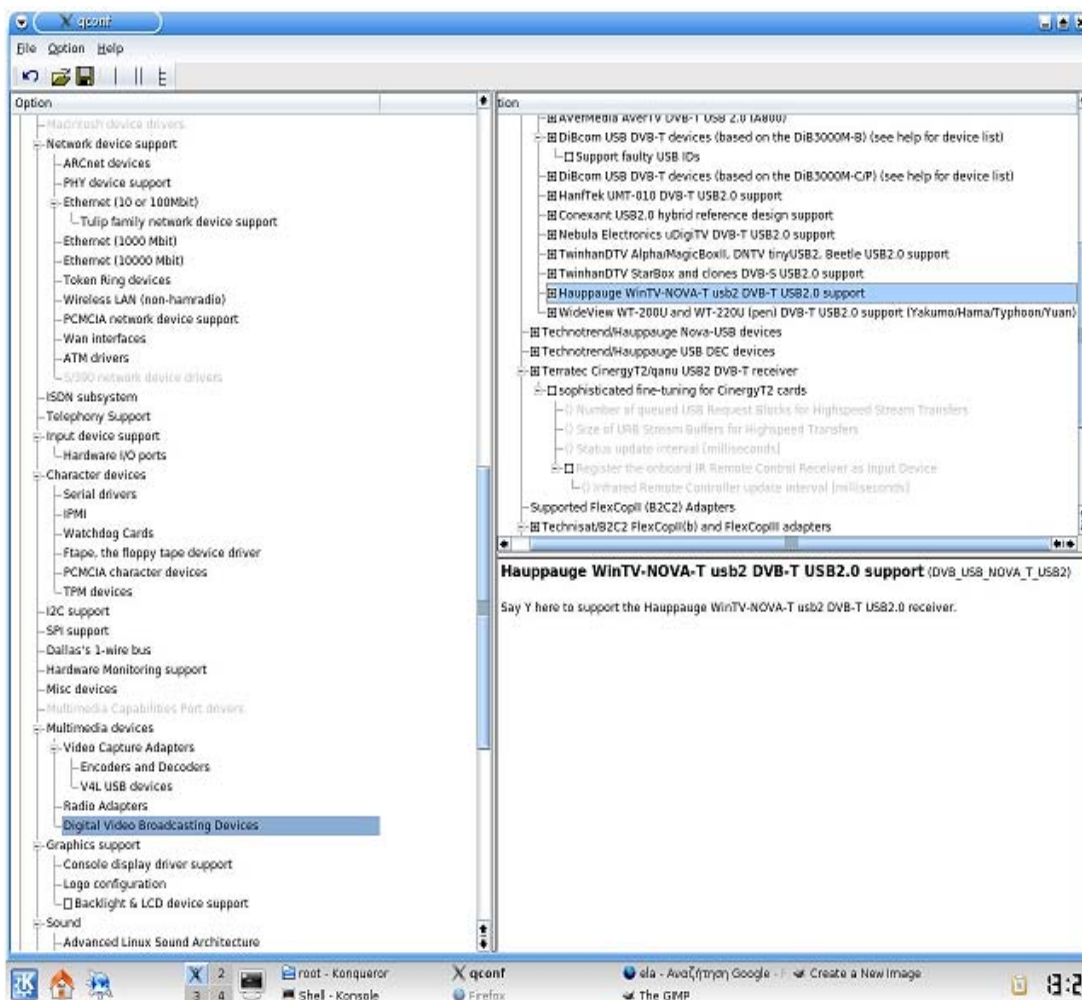
Σχ. 3.11 Εντολή xconfig

Όταν ανοίξει η εφαρμογή πρέπει να ενεργοποιηθούν οι παρακάτω επιλογές στο menu Networking. Networking → tcp/ip networking → Ip:advanced router, όπως φαίνεται και στο screenshot του Σχ. 3.12 που ακολουθεί.



Σχ. 3.12 Εφαρμογή xconfig

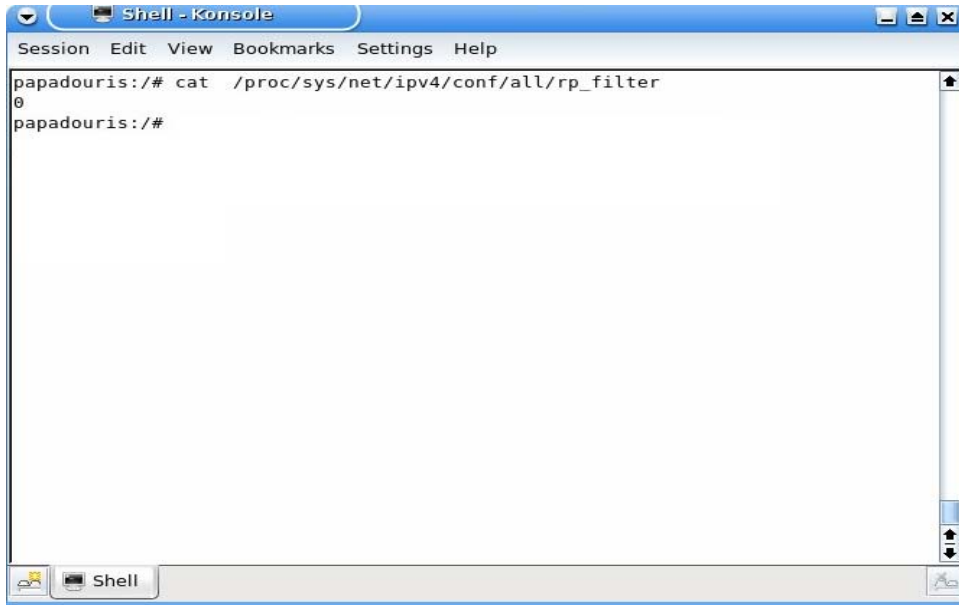
Για να ενεργοποιηθεί η κάρτα επιχειας ψηφιακής τηλεόρασης πρέπει στην ίδια εφαρμογή να βρεθεί και να επιλεγεί, όπως φαίνεται και στο screenshot του Σχ. 3.13 που ακολουθεί.



Σχ. 3.13 Επιλογή Κάρτας Επίγειας Ψηφιακής Τηλεόρασης

Επίσης στον ενδιάμεσο κόμβο διανομής χρησιμοποιείται η εντολή **rp-filter**, η οποία χρησιμοποιείται για να εξασφαλιστεί η αποδοχή των πακέτων μόνο από τα δικτυακά interfaces στα οποία απευθύνονται τα πακέτα IP. Ακόμα δίνει τη δυνατότητα σε αυτά τα interfaces αυτόματα να απορρίπτουν εισερχόμενα πακέτα των οποίων ο πίνακας διευθύνσεων δεν ταιριάζει με τις δικές τους διευθύνσεις.

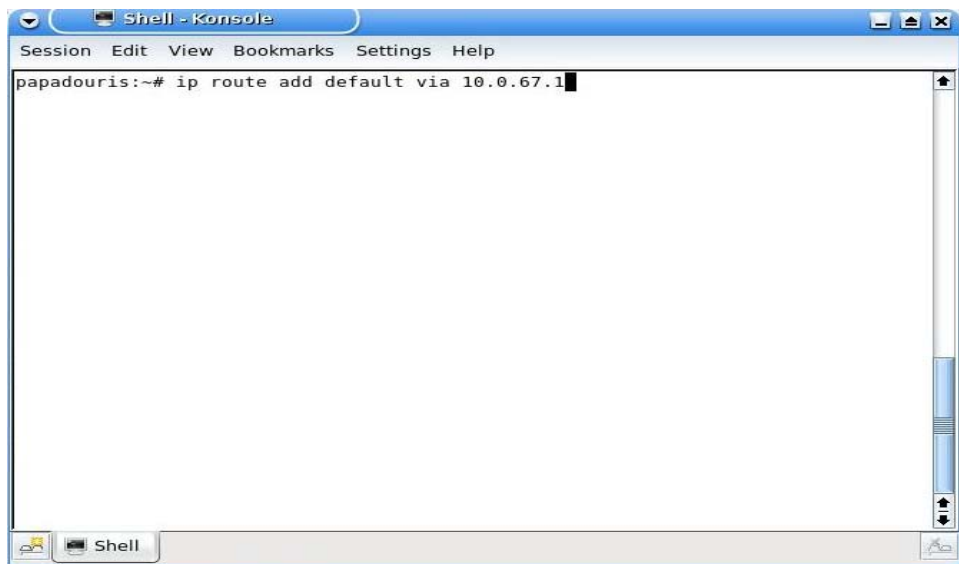
Ο έλεγχος για το αν έχει ενεργοποιηθεί η συγκεκριμένη εντολή γίνεται πληκτρολογώντας την σε ένα terminal των linux. Εφόσον το αποτέλεσμα της είναι 0, όπως φαίνεται στην εντολή του screenshot στο Σχ.3.14 στο terminal, τότε είναι ενεργοποιημένη. Αν είναι 1, τότε είναι απενεργοποιημένη και κόβει όλα τα πακέτα που έρχονται από ένα interface και φεύγουν από άλλο. Έτσι πρέπει να πληκτρολογηθεί η αντίστοιχη εντολή με την `ip_forwarding`, αλλά αυτή τη φορά θα έχουμε “`echo 0 > proc/sys/net/ipv4/conf/all/rp_filter`”.



```
Shell - Konsole
Session Edit View Bookmarks Settings Help
papadouris:~# cat /proc/sys/net/ipv4/conf/all/rp_filter
0
papadouris:~#
```

Σχ. 3.14 Εντολή `rp-filter`

Σημαντική εντολή είναι η **ip-route** με την η οποία δίνεται η δυνατότητα να δηλωθούν default εντολές δρομολόγησης της κίνησης, μέσα στον CMN. Έτσι από το screenshot που ακολουθεί βλέπουμε ότι μέσω της εντολής `ip-route` προωθούνται οποιαδήποτε IP πακέτα που προέρχονται από του χρήστες, στην `eth_1` και από εκεί στην `eth_0`, έχοντας ως τελικό προορισμό την πλατφόρμα (Σχ. 3.15).

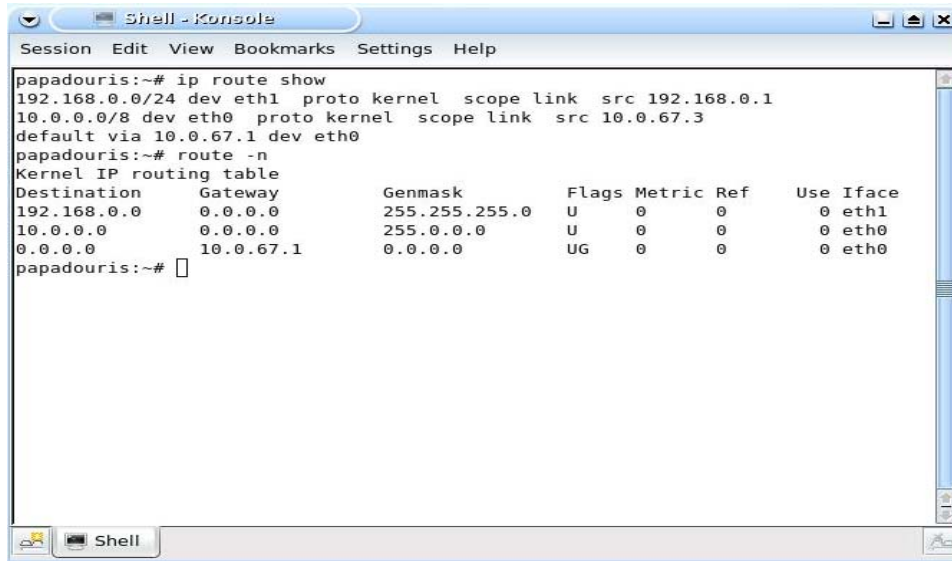


```
Shell - Konsole
Session Edit View Bookmarks Settings Help
papadouris:~# ip route add default via 10.0.67.1
```

Σχ. 3.15 Εντολή `ip-route` για την Δρομολόγηση της Κίνησης από την `eth_1` προς την `eth_0`

Ενώ αντίθετα η δρομολόγηση της κίνησης που έρχεται από την κάρτα ψηφιακής τηλεόρασης `dnb_0` και προωθείται προς την κάρτα δικτύου `eth_1` ενεργοποιείται αυτόματα, με τον καθορισμό της IP Address της κάρτας δικτύου `eth_1`.

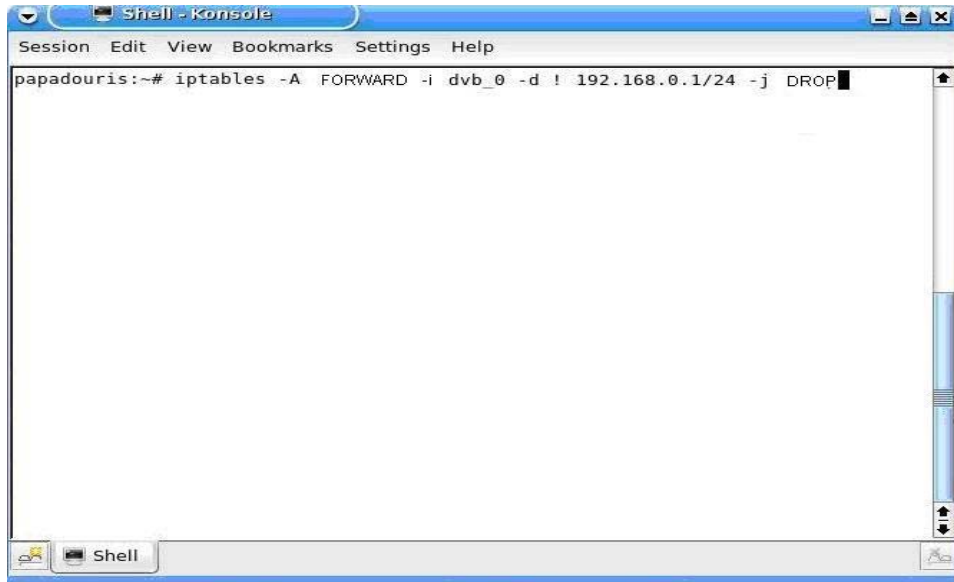
Ακολουθούν οι δρομολογήσεις των κινήσεων μέσα στον ενδιάμεσο κόμβο διανομής (CMN) και οι οποίες παρουσιάζονται με δύο τρόπους. Είτε με την εντολή “`ip route show`” είτε με την εντολή “`route -n`” (Σχ. 3.16).



```
Shell - Konsole
Session Edit View Bookmarks Settings Help
papadouris:~# ip route show
192.168.0.0/24 dev eth1 proto kernel scope link src 192.168.0.1
10.0.0.0/8 dev eth0 proto kernel scope link src 10.0.67.3
default via 10.0.67.1 dev eth0
papadouris:~# route -n
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
192.168.0.0      0.0.0.0        255.255.255.0   U        0      0      0 eth1
10.0.0.0         0.0.0.0        255.0.0.0       U        0      0      0 eth0
0.0.0.0         10.0.67.1     0.0.0.0         UG       0      0      0 eth0
papadouris:~#
```

Σχ. 3.16 Δρομολόγηση των Κινήσεων Μέσα στον CMN

Στο συγκεκριμένο δίκτυο υπάρχει μία προεπιλογή η οποία πρέπει να αποκλειστεί. Πρόκειται για την περίπτωση που φτάσουν στον ενδιάμεσο κόμβο διανομής IP πακέτα μέσω της `dnb_0` τα οποία δεν προορίζονται για αυτόν. Το αποτέλεσμα είναι να στέλνονται πίσω στον αποστολέα με αποτέλεσμα να φορτώνεται το δίκτυο με επιπλέον κίνηση. Έτσι λοιπόν για την αποφυγή τέτοιου είδους προβλημάτων χρησιμοποιείται ένα φίλτρο που «κόβει» την κίνηση η οποία φτάνει στην κάρτα `dnb_0` και δεν προορίζεται για το υποδίκτυό μας. Η εντολή για την ενεργοποίηση του φίλτρου είναι η **iptables** που παρουσιάζεται στο screenshot του Σχ. 3.17 που ακολουθεί.

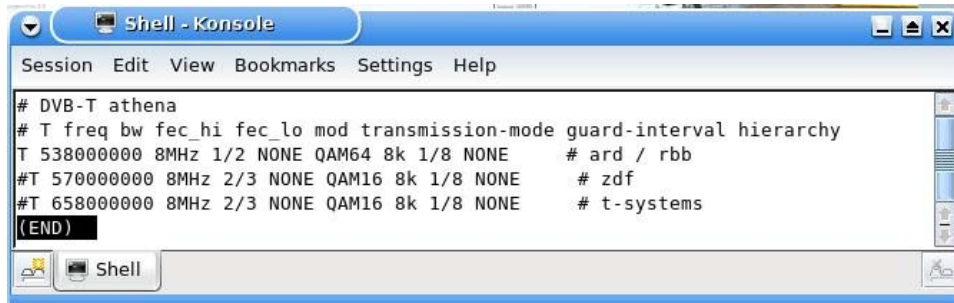


Σχ. 3.17 Εντολή iptables

Όταν ολοκληρωθούν όλες οι παραπάνω διαδικασίες, πρέπει να γίνει στον κόμβο ο εντοπισμός των τηλεοπτικών προγραμμάτων που εκπέμπονται από το κεντρικό σημείο εκπομπής. Για αυτό τον σκοπό χρησιμοποιείται ένα αρχείο που έχουμε ονομάσει «athena» το οποίο περιέχει όλα τα απαραίτητα στοιχεία του modulator της πλατφόρμας DVB-T. Τα στοιχεία αυτά είναι τα εξής:

Carrier	8k
Code rate	1/2
Guard interval	1/8
Constellation	64 QAM
Frequency	538 MHz

και τα έχουμε καθορίσει εμείς και στον modulator αλλά και στον ίδιο τον CMN. Τα παραπάνω στοιχεία φαίνονται και από το screenshot που ακολουθεί στο Σχ. 3.18

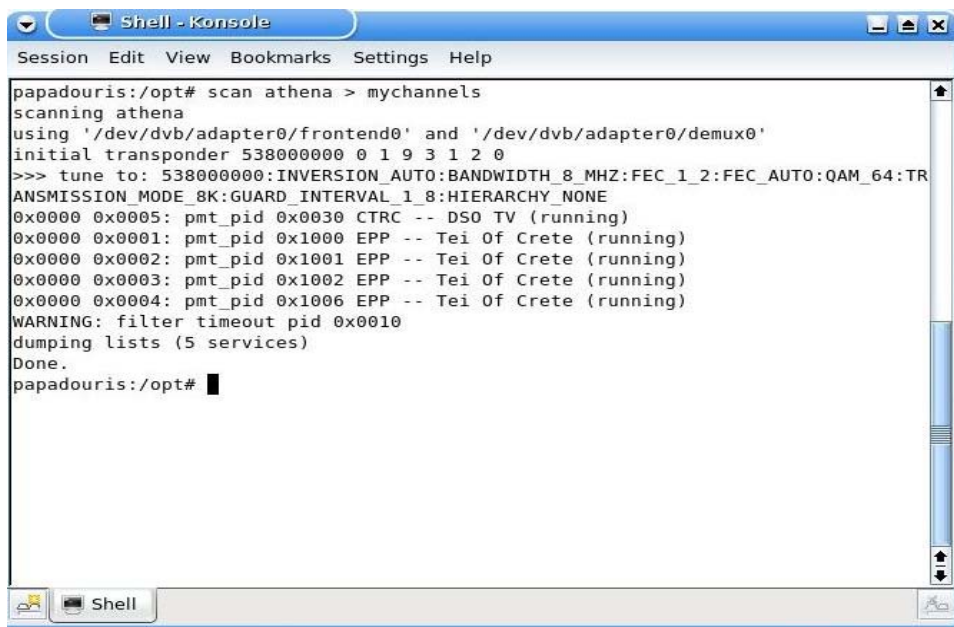


```
Shell - Konsole
Session Edit View Bookmarks Settings Help

# DVB-T athena
# T freq bw fec_hi fec_lo mod transmission-mode guard-interval hierarchy
T 538000000 8MHz 1/2 NONE QAM64 8k 1/8 NONE # ard / rbb
#T 570000000 8MHz 2/3 NONE QAM16 8k 1/8 NONE # zdf
#T 658000000 8MHz 2/3 NONE QAM16 8k 1/8 NONE # t-systems
(END)
```

Σχ. 3.18 Αρχείο “Athena”

Με την εντολή **scan** εντοπίζονται όλα τα τηλεοπτικά κανάλια που εκπέμπονται από την πλατφόρμα DVB-T, όπως φαίνεται και στο παρακάτω screenshot στο Σχ. 3.19. Παρατηρείται ότι ανιχνεύονται 5 τηλεοπτικά προγράμματα τα οποία αποθηκεύονται στη συνέχεια σε ένα αρχείο που το ονομάζουμε mychannels.



```
Shell - Konsole
Session Edit View Bookmarks Settings Help

papadouris:/opt# scan athena > mychannels
scanning athena
using '/dev/dvb/adapter0/frontend0' and '/dev/dvb/adapter0/demux0'
initial transponder 538000000 0 1 9 3 1 2 0
>>> tune to: 538000000:INVERSION_AUTO:BANDWIDTH_8_MHZ:FEC_1_2:FEC_AUTO:QAM_64:TRANSMISSION_MODE_8K:GUARD_INTERVAL_1_8:HIERARCHY_NONE
0x0000 0x0005: pmt_pid 0x0030 CTCR -- DSO TV (running)
0x0000 0x0001: pmt_pid 0x1000 EPP -- Tei Of Crete (running)
0x0000 0x0002: pmt_pid 0x1001 EPP -- Tei Of Crete (running)
0x0000 0x0003: pmt_pid 0x1002 EPP -- Tei Of Crete (running)
0x0000 0x0004: pmt_pid 0x1006 EPP -- Tei Of Crete (running)
WARNING: filter timeout pid 0x0010
dumping lists (5 services)
Done.
papadouris:/opt#
```

Σχ. 3.19 Διαθέσιμα Τηλεοπτικά Προγράμματα

Τέλος, με την εντολή **tzap**, παρέχεται η δυνατότητα για συντονισμό σε κάποιο από τα ανιχνεύσιμα προγράμματα σε μια συγκεκριμένη συχνότητα, όπως παρατηρείται και από το screenshot στο Σχ. 3.20.

```
Shell - Konsole
Session Edit View Bookmarks Settings Help
papadouris:/opt# tzap -c mychannels 'DSO TV'
using '/dev/dvb/adapter0/frontend0' and '/dev/dvb/adapter0/demux0'
tuning to 538000000 Hz
video pid 0x00a5, audio pid 0x0164
status 03 | signal f70e | snr 0001 | ber 001fffff | unc 0000ffff |
status 1f | signal f70d | snr 0001 | ber 00000000 | unc 00000000 | FE_HAS_LOCK
status 1f | signal 0000 | snr ffff | ber 00000000 | unc 00000000 | FE_HAS_LOCK
status 1f | signal 0000 | snr ffff | ber 00000000 | unc 00000000 | FE_HAS_LOCK
status 1f | signal 0000 | snr ffff | ber 00000000 | unc 00000000 | FE_HAS_LOCK
status 1f | signal 0000 | snr ffff | ber 00000000 | unc 00000000 | FE_HAS_LOCK
status 1f | signal 0000 | snr ffff | ber 00000000 | unc 00000000 | FE_HAS_LOCK
status 1f | signal 0000 | snr ffff | ber 00000000 | unc 00000000 | FE_HAS_LOCK
status 1f | signal 0000 | snr ffff | ber 00000000 | unc 00000000 | FE_HAS_LOCK
```

Σχ. 3.20 Συντονισμός στο Τηλεοπτικό Πρόγραμμα “DSO TV”

Αυτή ήταν και η διαδικασία υλοποίησης ενός ενδιάμεσου κόμβου διανομής (CMN) με σκοπό την εξυπηρέτηση των ενεργών χρηστών-πολιτών, παρέχοντας τους πρόσβαση στο ευρυζωνικό δίκτυο της επίγειας ψηφιακής τηλεόρασης.

4. ΣΤΟΙΧΕΙΟΘΕΤΗΣΗ ΚΑΙ ΣΧΟΛΙΑΣΜΟΣ ΠΕΙΡΑΜΑΤΙΚΩΝ ΜΕΤΡΗΣΕΩΝ

4.1. Εισαγωγή

Κύριος σκοπός της διπλωματικής αυτής, είναι η αξιολόγηση των επιδόσεων του δικτύου που περιγράφηκε στο προηγούμενο κεφάλαιο. Θα πρέπει να πραγματοποιηθούν οι απαραίτητες πειραματικές μετρήσεις μέσω των οποίων θα γίνει τελικά η αξιολόγηση του δικτύου.

Οι μετρήσεις που πραγματοποιήθηκαν, προσπαθούν να καλύψουν όσο το δυνατόν καλύτερα τις δυνατότητες και τα χαρακτηριστικά λειτουργίας του δικτύου. Γι' αυτόν το λόγο έπρεπε να δοθεί ιδιαίτερη προσοχή στην επιλογή των κριτηρίων αξιολόγησης του δικτύου. Τα κριτήρια λοιπόν τα οποία σε κάθε μέτρηση έκριναν τις επιδόσεις των δύο δικτύων είναι:

Για την μέτρηση δικτυακής κίνησης από σημείο σε σημείο κάνοντας χρήση του πρωτοκόλλου TCP:

- 1) Η μέση χρήση-φόρτος του δικτύου, ανά δευτερόλεπτο, οι οποίες όμως χαρακτηρίζουν σε μεγάλο βαθμό, τη συνολική χρήση του δικτύου (THROUGHPUT).
- 2) Η μέση καθυστέρηση του δικτύου (RTT).
- 3) Οι απώλειες και αναμεταδόσεις πακέτων (LOSSES AND RETRANSMISSIONS).
- 4) Sequence number

Και για την μέτρηση δικτυακής κίνησης πάλι από σημείο σε σημείο ,αλλά κάνοντας χρήση του πρωτοκόλλου UDP:

- 1) Η καθυστέρηση που είχαμε μεταξύ των πακέτων (Packet_to_Packet_Delay)
- 2) One way delay
- 3) Η διακύμανση της καθυστέρησης (JITTER)

4.2. Παρουσίαση Αποτελεσμάτων και Γραφικών Παραστάσεων Δικτυακής Κίνησης TCP

4.2.1. Γενικά

Στο προηγούμενο κεφάλαιο έγινε μια λεπτομερή αναφορά πάνω στην αρχιτεκτονική του δικτύου επίγειας ψηφιακής τηλεόρασης με κανάλια επιστροφής ISDN και Wlan.

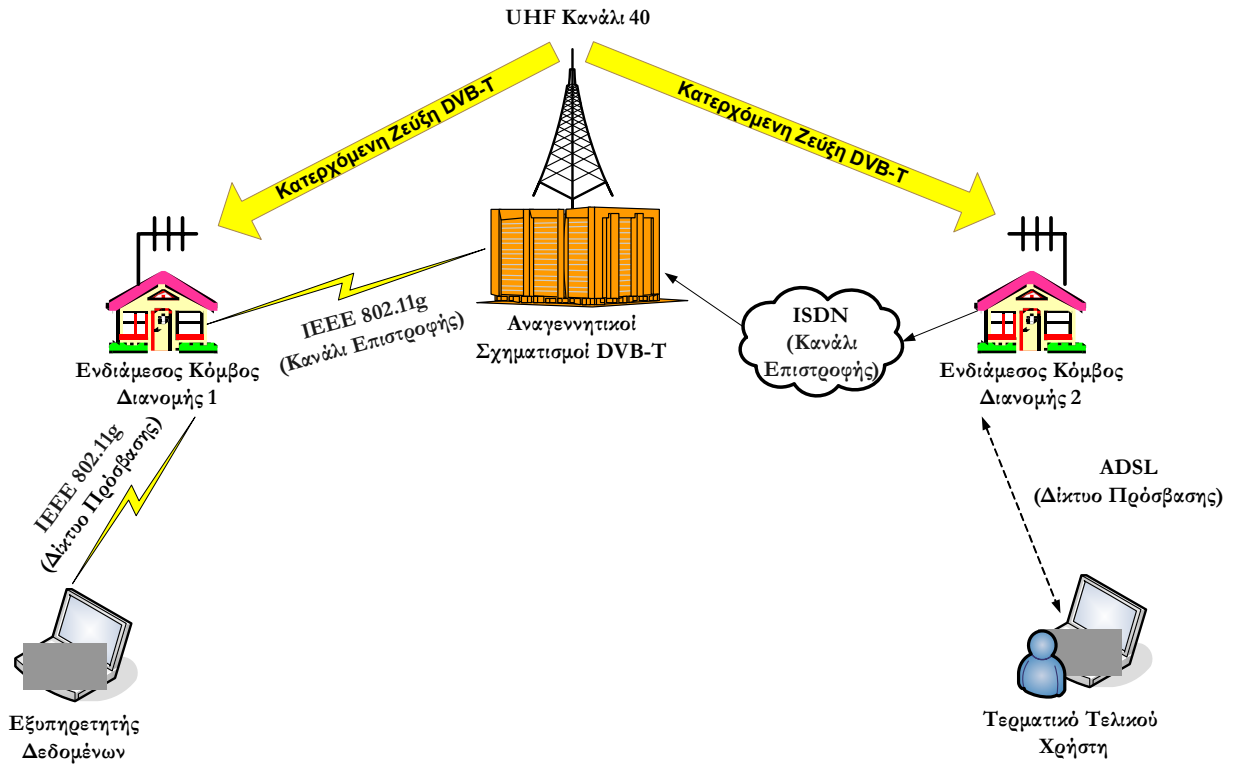
Παρακάτω αναφέρονται τα σενάρια που υιοθετήθηκαν πάνω στο προγραμματισμό των συγκεκριμένων πειραματικών μετρήσεων και για τα δύο είδη της δικτυακής κίνησης που δημιουργήθηκαν (TCP και UDP).

Επίσης γίνεται η ανάλυση και ο σχολιασμός των πειραματικών μετρήσεων με σκοπό την εξαγωγή χρήσιμων συμπερασμάτων για τις επιδόσεις των προαναφερθέντων δικτύων (σύμφωνα με τις προτεινόμενες αρχιτεκτονικές). Είναι αναμφισβήτητο το σημαντικότερο κομμάτι της πτυχιακής εξ' αιτίας του ότι κρίνεται η προσφορά της όσον αφορά τις επιδόσεις του προτεινόμενου δικτύου, σε σύγκριση με ένα δίκτυο που είναι ήδη σε λειτουργία. Στο διάστημα λοιπόν που διήρκησαν οι μετρήσεις, ολοκληρώθηκαν αρκετές μετρήσεις για κάθε σενάριο έτσι ώστε να γίνουν όσο το δυνατόν αντιπροσωπευτικότερα τα δείγματα που παρουσιάζονται.

4.2.2. Μεμονωμένες Μετρήσεις

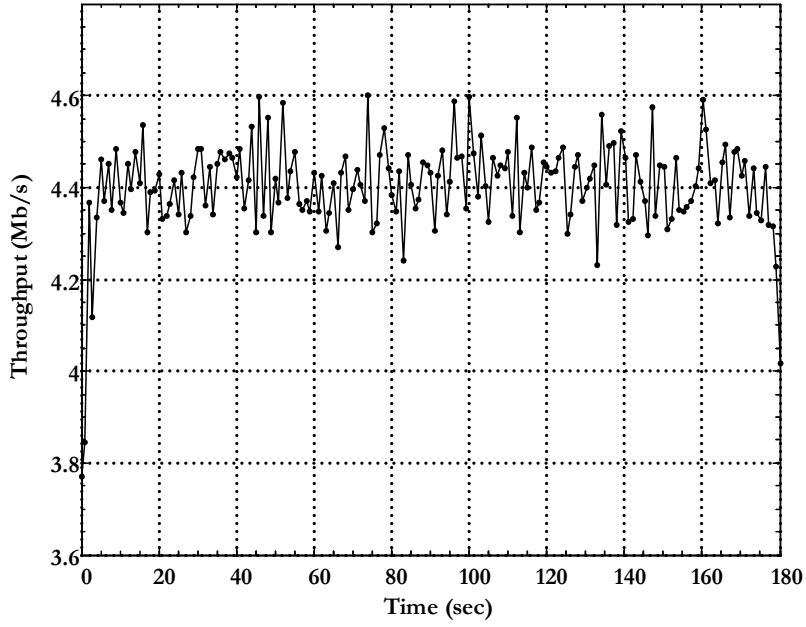
4.2.2.1. Αξιολόγηση Δικτυακής Κίνησης από τον τελικό Χρήστη προς τον Εξυπηρετητή Δεδομένων

Στο σχήμα 4.1 παρουσιάζεται το κύριο σενάριο που σχεδιάστηκε κατά τη διάρκεια της διπλωματικής, όπου ένας σταθερός χρήστης ζητάει υπηρεσίες internet από έναν εξυπηρετητή δεδομένων, κάνοντας χρήση των δικτύων ISDN και WLAN ως κανάλια επιστροφής και του δικτύου της ψηφιακής τηλεόρασης ως κανάλι καθόδου. Η δικτυακή κίνηση από το χρήστη προς τον εξυπηρετητή δεδομένων και αντίστροφα, γίνεται με τη βοήθεια ενδιάμεσων κόμβων διανομής (CMNs).

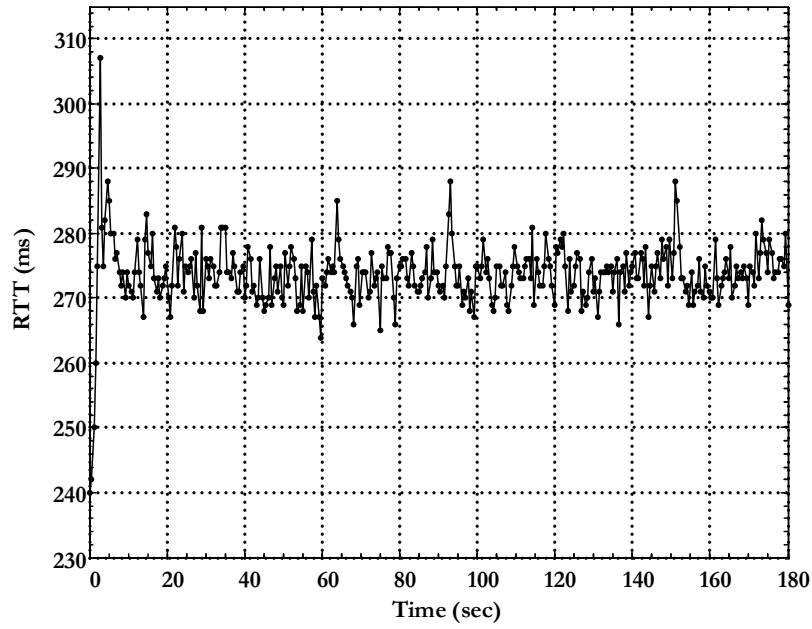


Σχ. 4.1 Αρχιτεκτονική Δικτύου

Τα αποτελέσματα του παραπάνω σεναρίου δίνουν ότι το μέσω Throughput, όταν ο χρήστης του CMN 2 ζητάει TCP υπηρεσίες, οι οποίες παρέχονται από τον εξυπηρετητή δεδομένων μέσω του CMN 1, είναι 4.40 Mb/s, ενώ το μέσο RTT είναι της τάξεως των 273.69 ms. Τα αποτελέσματα των Throughput και RTT παρουσιάζονται στα γραφήματα Σχ. 4.2 και Σχ. 4.3 αντίστοιχα. Σημειώνεται ότι στο σενάριο αυτό δεν παρουσιάστηκαν απώλειες (Losses & Retransmissions).



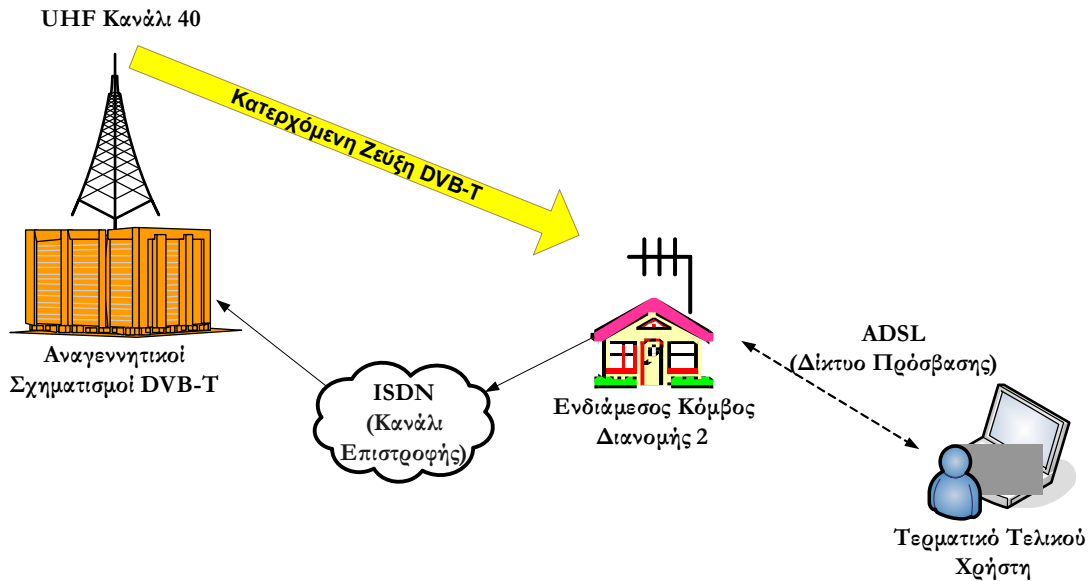
Σχ. 4.2 Throughput Δικτύου (σχ. 4.1)



Σχ. 4.3 RTT Δικτύου (σχ. 4.1)

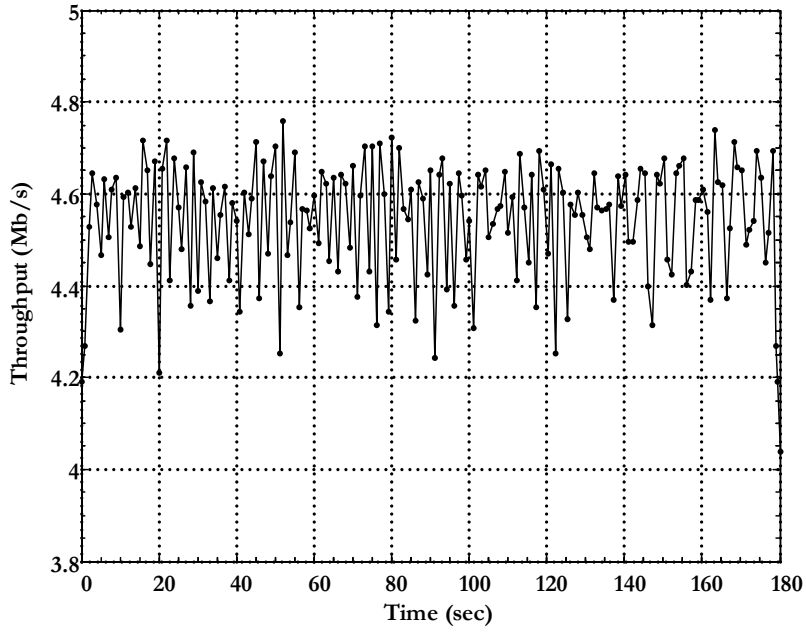
4.2.2.2. Αξιολόγηση Δικτυακής Κίνησης από τον τελικό Χρήστη προς την Αναγεννητική Πλατφόρμα

Στο σενάριο του σχήματος 4.4, έχει σκοπό να αξιολογήσει την απόδοση της επικοινωνίας μεταξύ του τελικού χρήστη και της αναγεννητικής πλατφόρμας, όπου χρησιμοποιούν ως κατερχόμενη ζεύξη το DVB-T κανάλι και ως ανερχόμενη ζεύξη (ή κανάλι επιστροφής), ένα ISDN κανάλι και ως κανάλι πρόσβασης χρησιμοποιείται το ADSL δίκτυο.

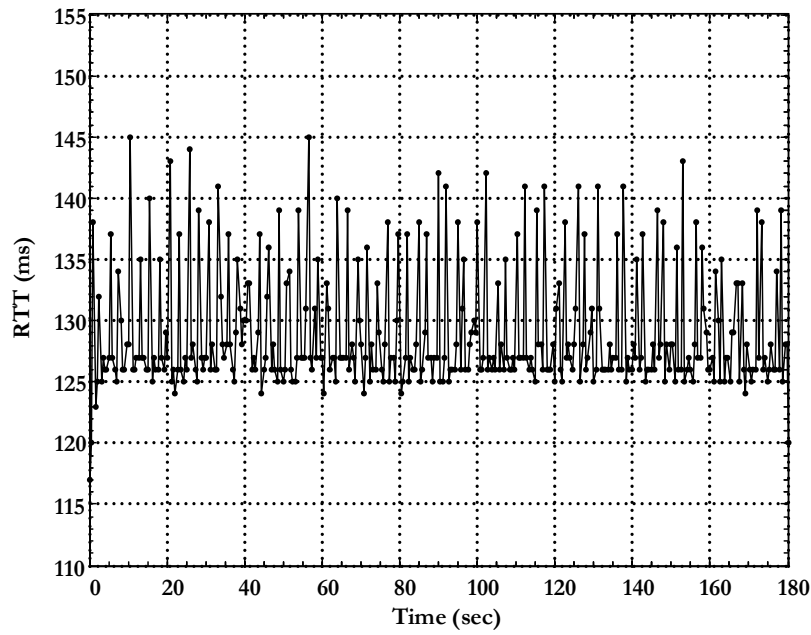


Σχ. 4.4 Δίκτυο Πρόσβασης και Αλυσίδα Uplink – Downlink για τον CMN 2

Τα αποτελέσματα του παραπάνω σεναρίου δίνουν ότι το μέσω Throughput είναι 4.54 Mb/s, ενώ το μέσο RTT είναι 128.83 ms. Τα αποτελέσματα των Throughput και RTT παρουσιάζονται στα γραφήματα Σχ. 4.5 και Σχ. 4.6 αντίστοιχα. Σημειώνεται ότι στο σενάριο αυτό δεν παρουσιάστηκαν απώλειες (Losses & Retransmissions).



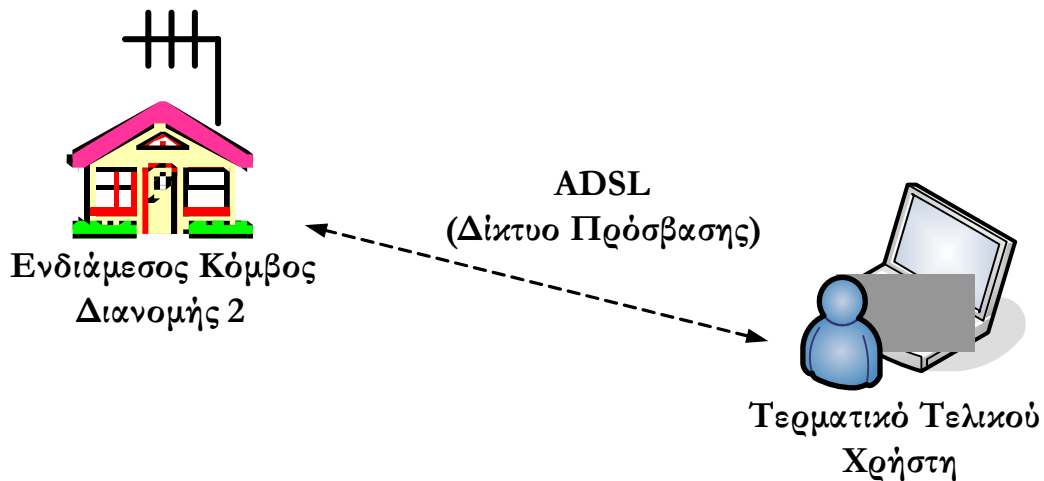
Σχ. 4.5 Throughput Δικτύου (σχ. 4.4)



Σχ. 4.6 RTT Δικτύου (σχ. 4.4)

4.2.2.3. Αξιολόγηση Δικτυακής Κίνησης από τον τελικό Χρήστη προς τον Ενδιάμεσο Κόμβο Διανομής 2

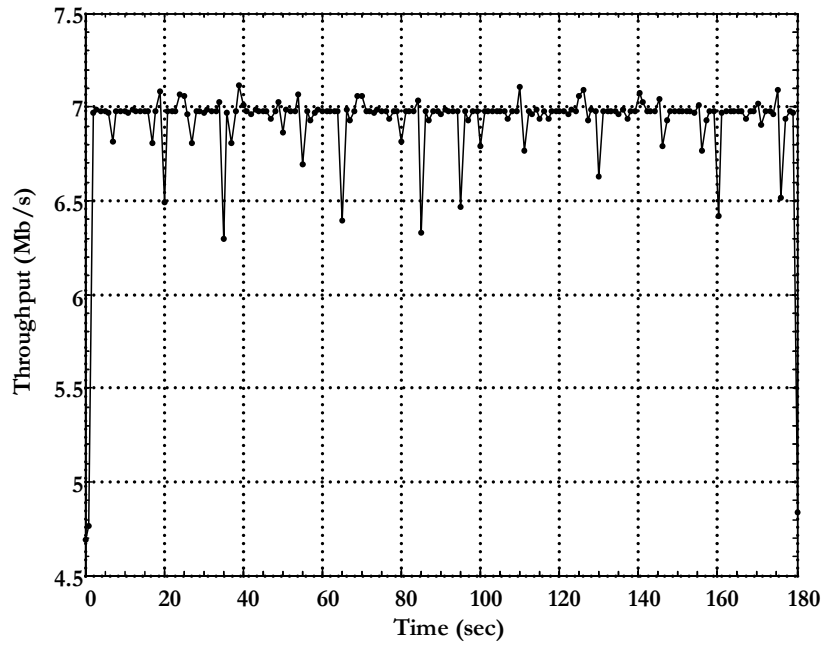
Το επόμενο σενάριο (Σχ. 4.7), έχει ως σκοπό να αξιολογήσει την απόδοση του καναλιού πρόσβασης ADSL, μέσω του οποίου ο τελικός χρήστης συνδέεται στον CMN 2, ώστε να ζητήσει και στη συνέχεια να λάβει τις επιθυμητές υπηρεσίες.



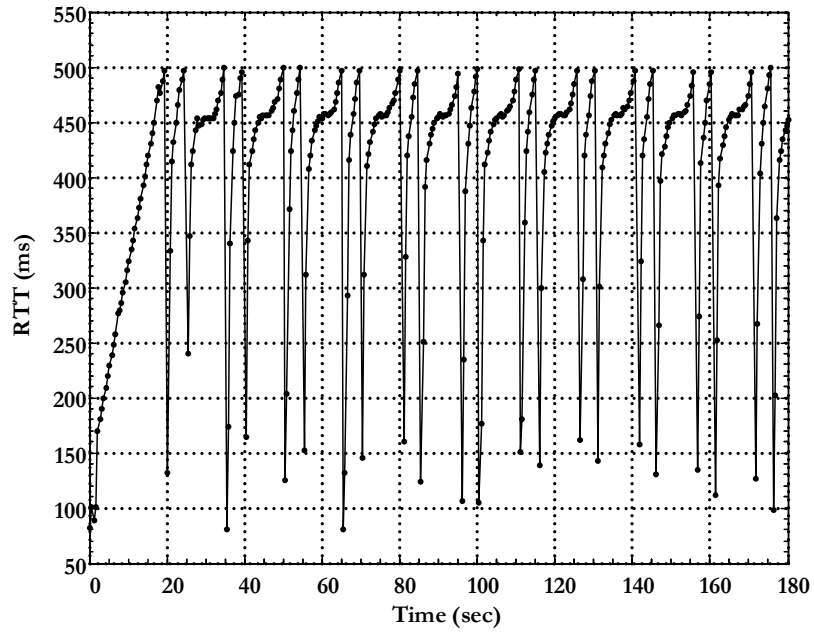
Σχ. 4.7 Δίκτυο Πρόσβασης για τον CMN 2

Τα αποτελέσματα του παραπάνω σεναρίου δίνουν ότι το μέσο Throughput είναι 6.90 Mb/s, ενώ το μέσο RTT είναι 404.67 ms. Τα αποτελέσματα των Throughput και RTT παρουσιάζονται στα γραφήματα Σχ. 4.8 και Σχ. 4.9 αντίστοιχα.

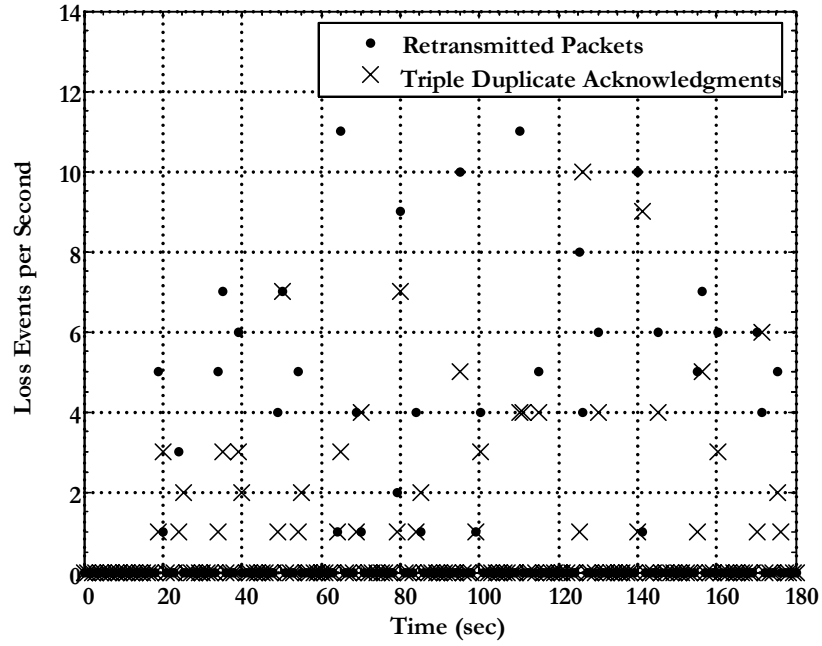
Επίσης, από τη γραφική παράσταση του Throughput (Σχ. 4.8), παρατηρούμε αρκετές βυθίσεις, οι οποίες οφείλονται σε απώλειες (Σχ. 4.10). Οι απώλειες στο ADSL δίκτυο αιτιολογούνται ακολούθως. Η μέση τιμή του Throughput είναι σύμφωνα με τις μετρήσεις (Σχ. 4.8) της τάξεως των 6.90 Mb/s, έχοντας θέσει στον Amber bit rate ίσο με 8 Mb/s. Αντίθετα παρατηρούμε ότι στο σενάριο του Σχ. 4.1, δεν παρουσιάστηκαν απώλειες και αυτό συνέβη διότι ολόκληρη η αλυσίδα από τον τελικό χρήστη μέχρι τον εξυπηρετητή δεδομένων, φτάνει στα 4.40 Mb/s, έχοντας θέσει 8 Mb/s στον Amber, οπότε ποτέ δε φτάνουμε σε κορεσμό για να έχουμε ως συνέπεια τις απώλειες. Αντίθετα στο ADSL δίκτυο, φτάνουμε στον κορεσμό, με αποτέλεσμα να έχουμε αναμεταδώσεις πακέτων και μεγάλο RTT Delay, όπου είναι η μέση καθυστέρηση του δικτύου για τη μεταφορά των πακέτων.



Σχ. 4.8 Throughput Δικτύου (σχ. 4.7)



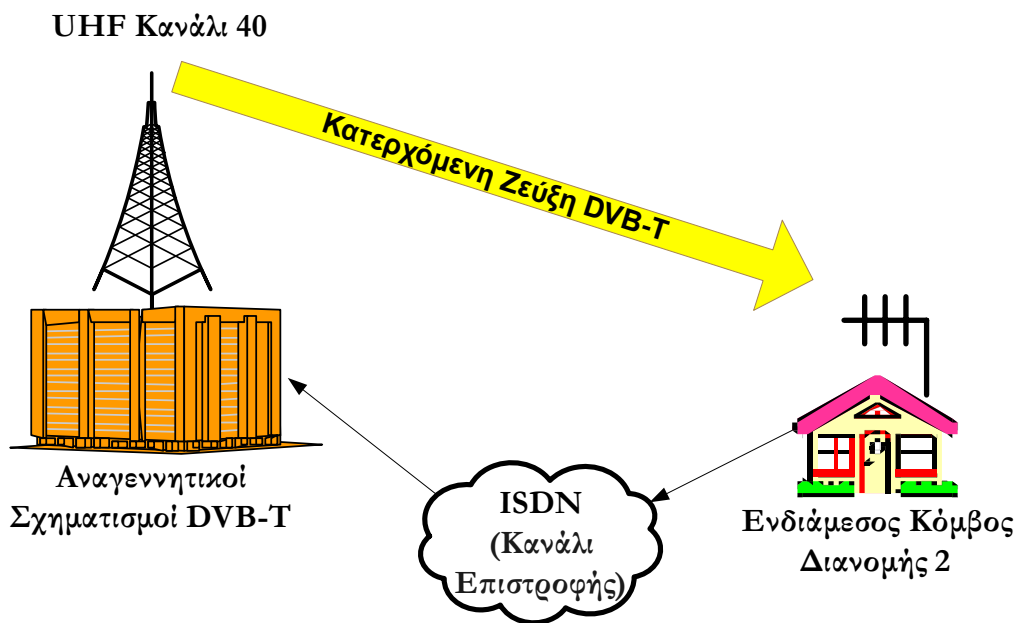
Σχ. 4.9 RTT Δικτύου (σχ. 4.7)



Σχ. 4.10 Retransmissions και Triple Duplicate Packets Δικτύου (σχ. 4.7)

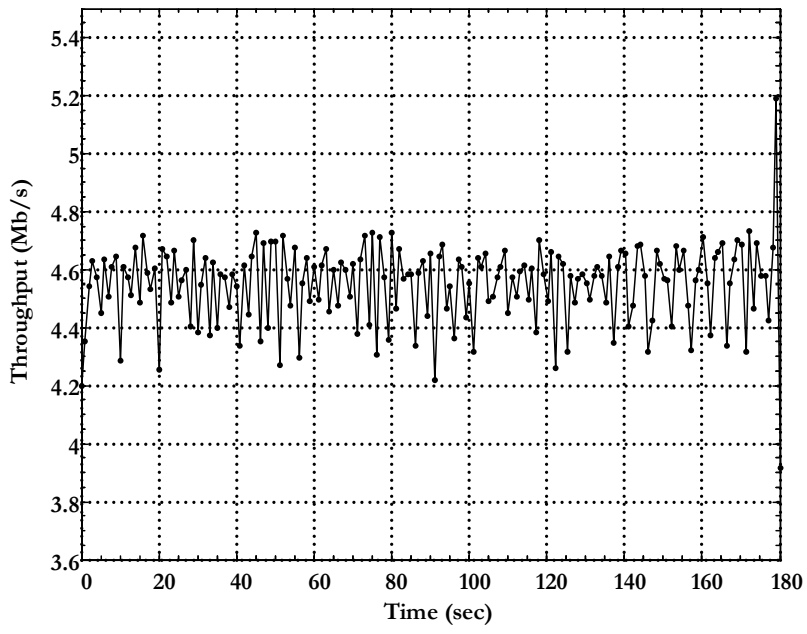
4.2.2.4. Αξιολόγηση Δικτυακής Κίνησης από τον Ενδιάμεσο Κόμβο Διανομής των Τελικών Χρηστών προς την Αναγεννητική Πλατφόρμα

Το σενάριο του Σχ. 4.11, αξιολογεί την απόδοση της επικοινωνίας μεταξύ του CMN, στον οποίο συνδέονται οι τελικοί χρήστες, και της αναγεννητικής πλατφόρμας, όπου χρησιμοποιούν ως κατερχόμενη ζεύξη το DVB-T κανάλι και ως ανερχόμενη ζεύξη ένα ISDN κανάλι.

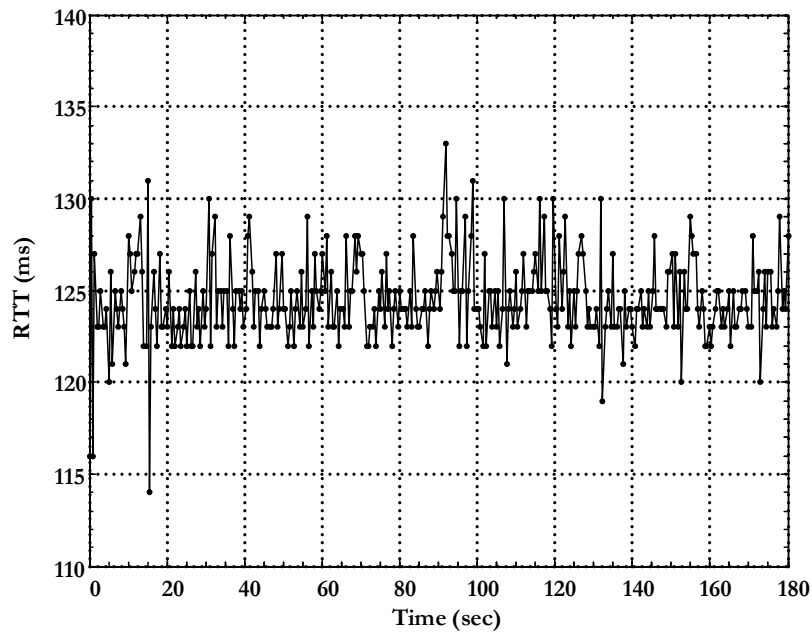


Σχ. 4.11 Αλυσίδα Uplink – Downlink για τον CMN 2

Τα αποτελέσματα του παραπάνω σεναρίου δίνουν ότι το μέσω Throughput είναι 4.54 Mb/s, ενώ το μέσο RTT είναι 124.48 ms. Τα αποτελέσματα των Throughput και RTT παρουσιάζονται στα γραφήματα Σχ. 4.12 και Σχ. 4.13 αντίστοιχα. Σημειώνεται ότι στο σενάριο αυτό δεν παρουσιάστηκαν απώλειες (Losses & Retransmissions).



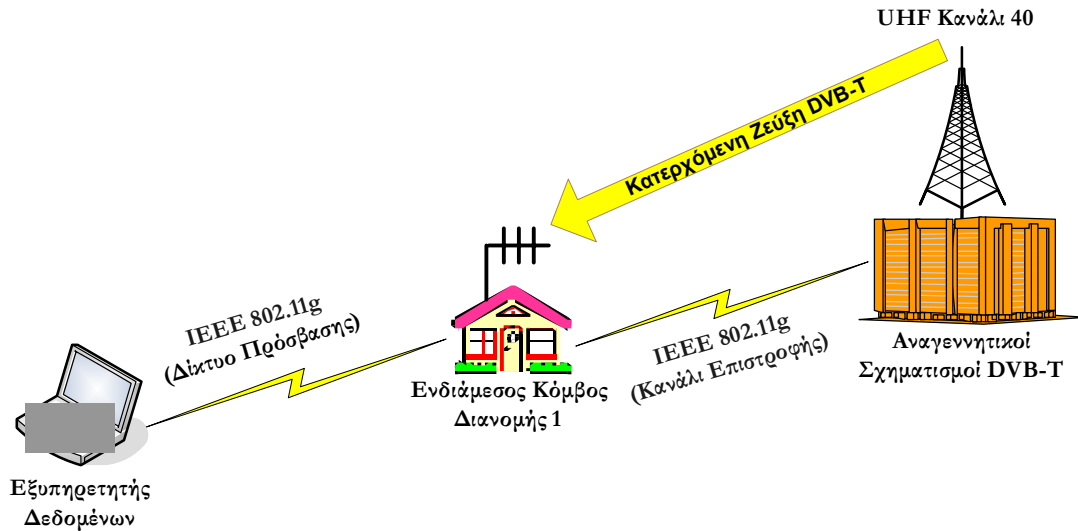
Σχ. 4.12 Throughput Δικτύου (σχ. 4.11)



Σχ. 4.13 RTT Δικτύου (σχ. 4.11)

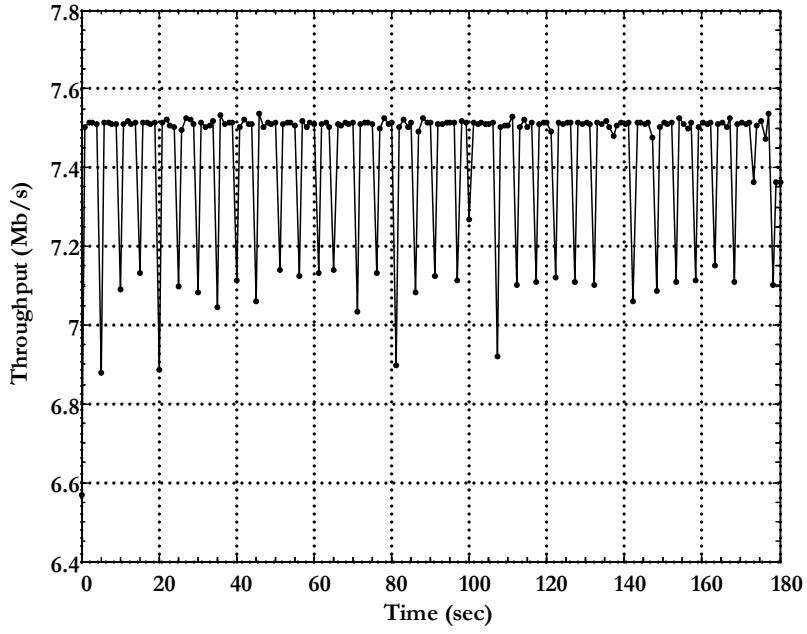
4.2.2.5. Αξιολόγηση Δικτυακής Κίνησης από τον Εξυπηρετητή Δεδομένων προς την Αναγεννητική Πλατφόρμα

Το σενάριο του Σχ. 4.14, έχει σκοπό να αξιολογήσει την απόδοση της επικοινωνίας μεταξύ του εξυπηρετητή δεδομένων και της αναγεννητικής πλατφόρμας, όπου χρησιμοποιούν ως κατερχόμενη ζεύξη το DVB-T κανάλι και ως ανερχόμενη ζεύξη ένα ασύρματο κανάλι (802.11g), ως κανάλι πρόσβασης χρησιμοποιείται ένα ασύρματο δίκτυο (802.11g).

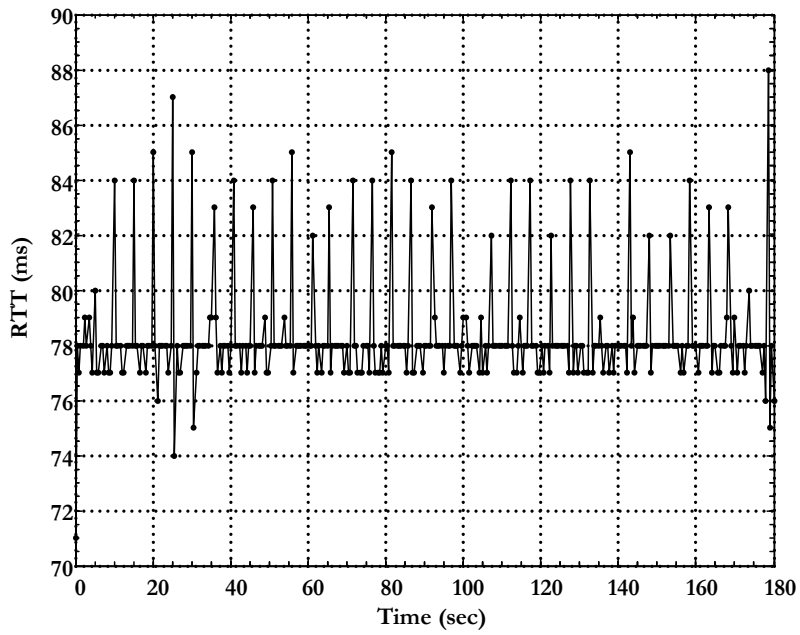


Σχ. 4.14 Δίκτυο Πρόσβασης και Αλυσίδα Uplink – Downlink για τον CMN 1

Τα αποτελέσματα του παραπάνω σεναρίου δίνουν ότι το μέσω Throughput είναι 7.42 Mb/s, ενώ το μέσο RTT είναι 78.27 ms. Τα αποτελέσματα των Throughput και RTT παρουσιάζονται στα γραφήματα Σχ. 4.15 και Σχ. 4.16 αντίστοιχα. Σημειώνεται ότι στο σενάριο αυτό δεν παρουσιάστηκαν απώλειες (Losses & Retransmissions).



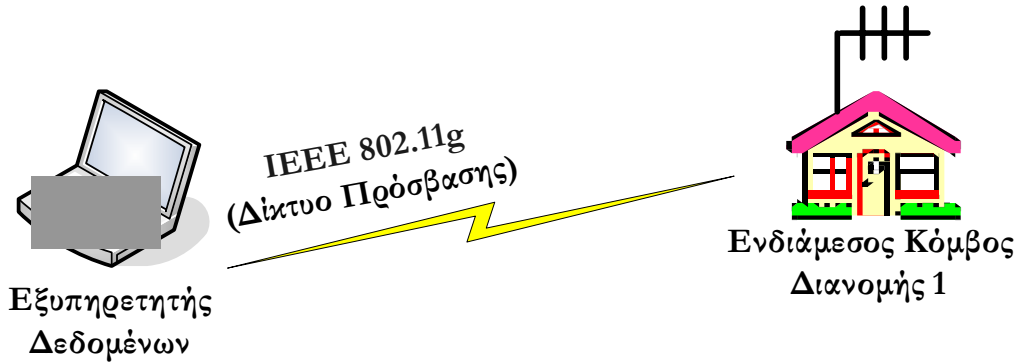
Σχ. 4.15 Throughput Δικτύου (σχ. 4.14)



Σχ. 4.16 RTT Δικτύου (σχ. 4.14)

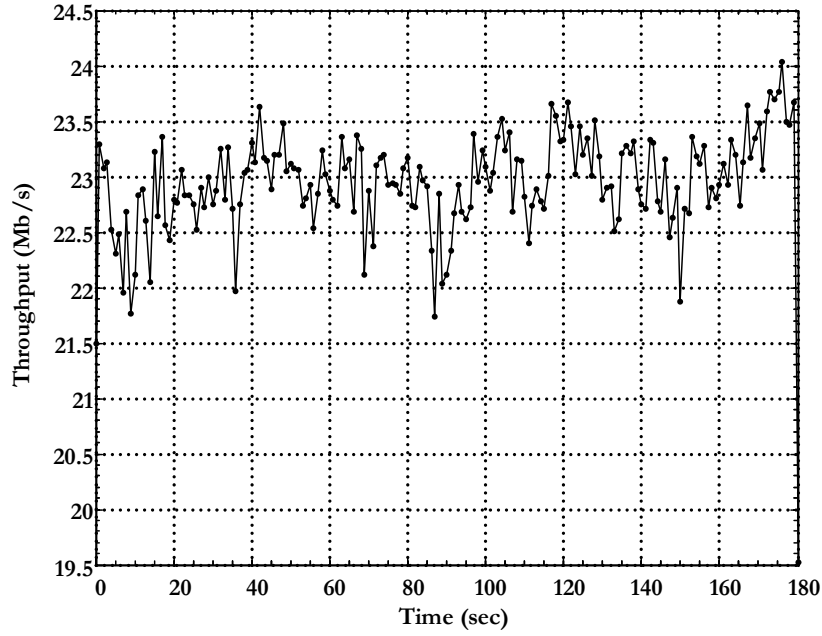
4.2.2.6. Αξιολόγηση Δικτυακής Κίνησης από τον Εξυπηρετητή Δεδομένων προς τον Ενδιάμεσο Κόμβο Διανομής 1

Το σενάριο του Σχ. 4.17, έχει σκοπό να αξιολογήσει την απόδοση του καναλιού πρόσβασης Wlan (802.11g), μέσω του οποίου ο εξυπηρετητής δεδομένων συνδέεται στον CMN του, ώστε να παρέχει υπηρεσίες, στην αναγεννητική πλατφόρμα και στην συνέχεια, μέσω αυτής, στους τελικούς χρήστες.

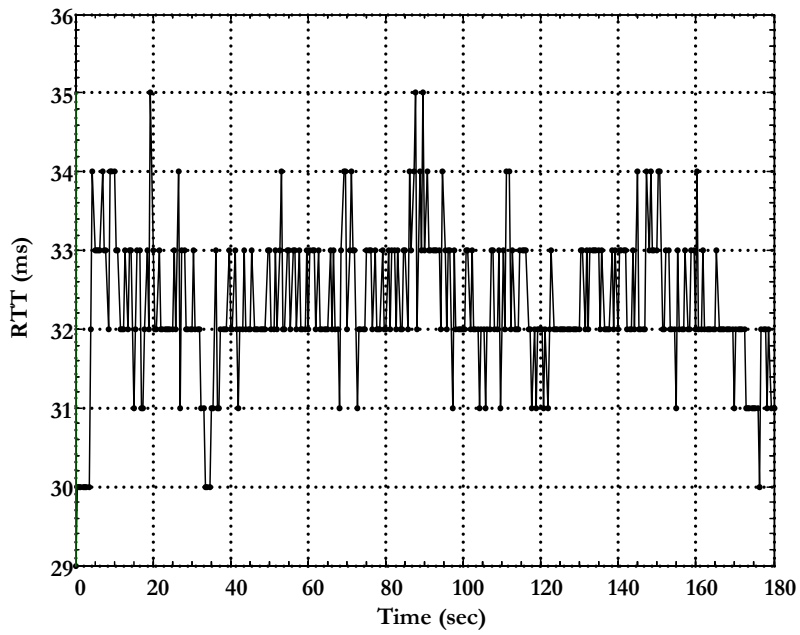


Σχ. 4.17 Δίκτυο Πρόσβασης για τον CMN 1

Τα αποτελέσματα του παραπάνω σεναρίου δίνουν ότι το μέσο Throughput είναι 22.93 Mb/s, ενώ το μέσο RTT είναι 32.29 ms. Τα αποτελέσματα των Throughput και RTT παρουσιάζονται στα γραφήματα Σχ. 4.18 και Σχ. 4.19 αντίστοιχα. Σημειώνεται ότι στο σενάριο αυτό δεν παρουσιάστηκαν απώλειες (Losses & Retransmissions).



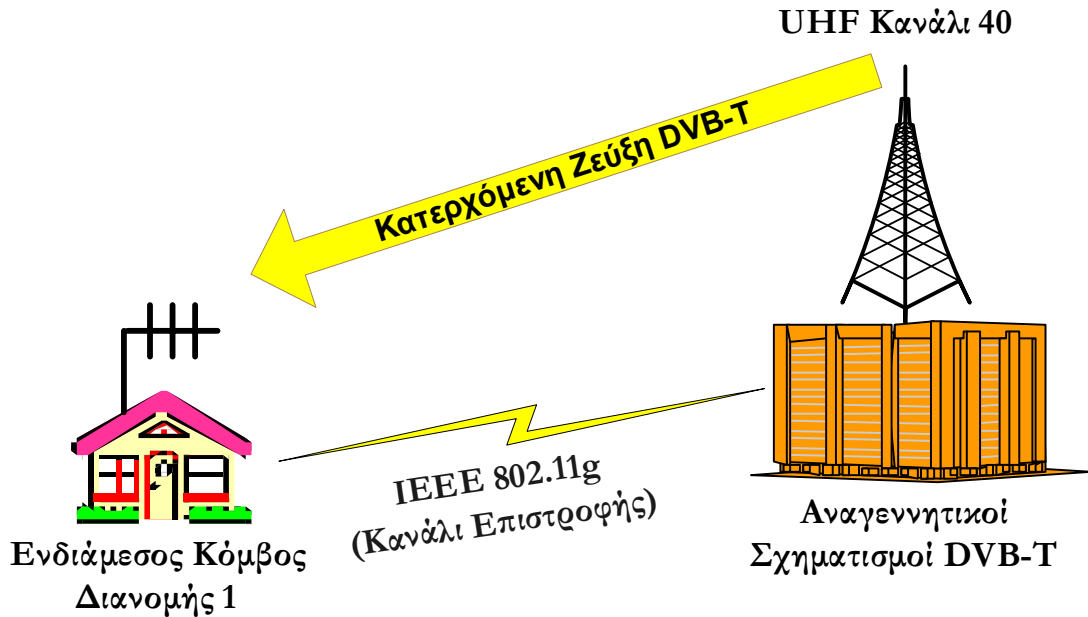
Σχ. 4.18 Throughput Δικτύου (σχ. 4.17)



Σχ. 4.19 RTT Δικτύου (σχ. 4.17)

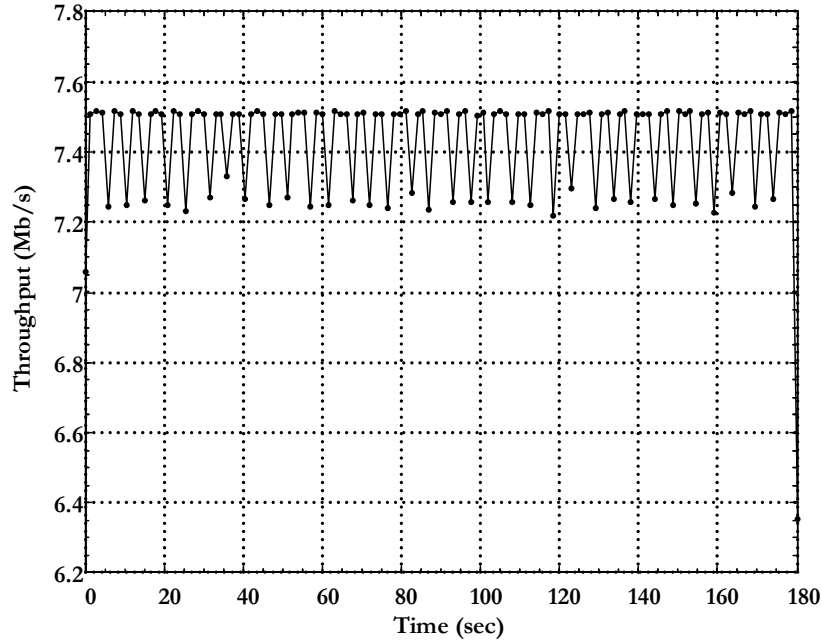
4.2.2.7. Αξιολόγηση Δικτυακής Κίνησης από τον Ενδιάμεσο Κόμβο Διανομής του Εξυπηρετητή Δεδομένων προς την Αναγεννητική Πλατφόρμα

Το σενάριο του Σχ. 4.20, αξιολογεί την απόδοση της επικοινωνίας μεταξύ του CMN, στον οποίο συνδέεται ο εξυπηρετητής δεδομένων, και της αναγεννητικής πλατφόρμας, όπου χρησιμοποιούν ως κατερχόμενη ζεύξη το DVB-T κανάλι και ως ανερχόμενη ένα ασύρματο κανάλι (802.11g).

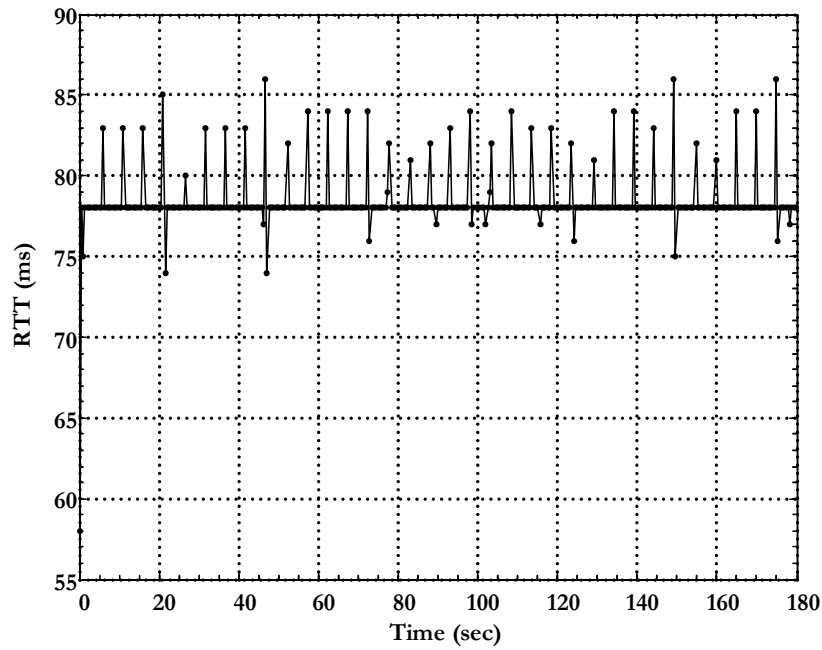


Σχ. 4.20 Αλυσίδα Uplink – Downlink για τον CMN 1

Τα αποτελέσματα του παραπάνω σεναρίου δίνουν ότι το μέσο Throughput είναι 7.42 Mb/s, ενώ το μέσο RTT είναι 78.36 ms. Τα αποτελέσματα των Throughput και RTT παρουσιάζονται στα γραφήματα Σχ. 4.21 και Σχ. 4.22 αντίστοιχα. Σημειώνεται ότι στο σενάριο αυτό δεν παρουσιάστηκαν απώλειες (Losses & Retransmissions).



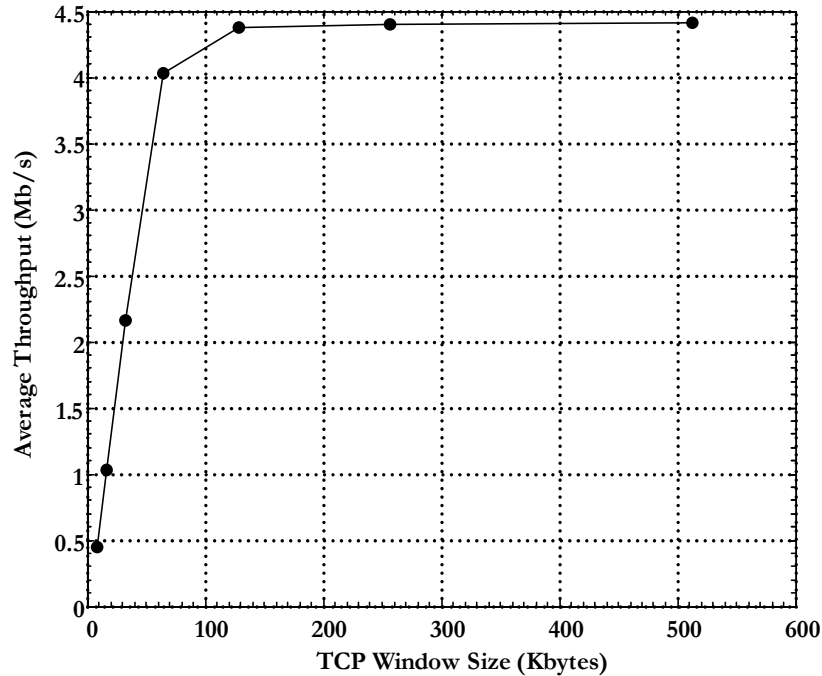
Σχ. 4.21 Throughput Δικτύου (σχ. 4.20)



Σχ. 4.22 RTT Δικτύου (σχ. 4.20)

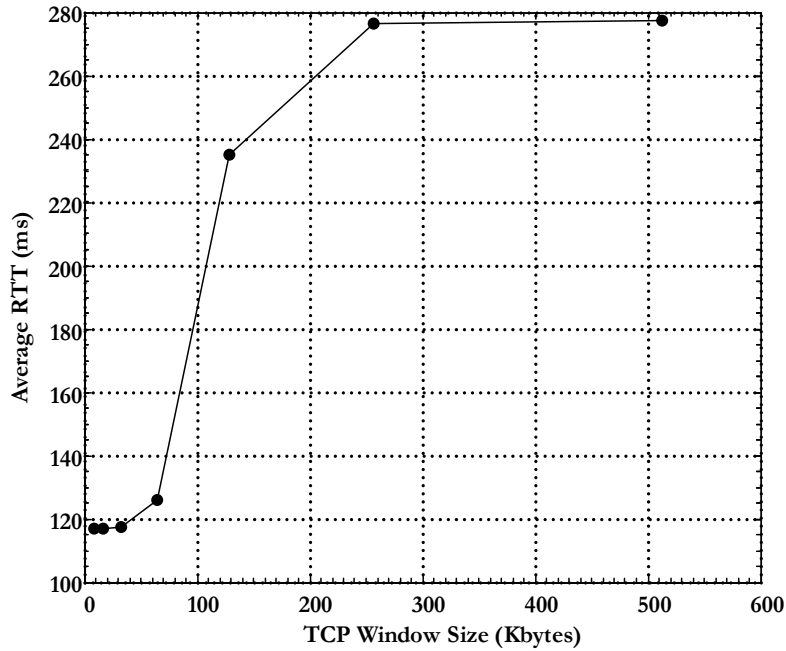
4.2.3. Αξιολόγηση Δικτυακής Κίνησης Μεταβαλλόντας το Window Size

Στο σενάριο αυτό μεταβάλλουμε το Window Size στον Client, προκειμένου να παρατηρήσουμε τη συμπεριφορά του δικτύου σε διαφορετικές καταστάσεις. Συγκεκριμένα, στους παρακάτω πίνακες αναφέρονται οι τιμές των TCP Window Size που χρησιμοποιήθηκαν κατά τη διάρκεια των μετρήσεων, και τα αποτελέσματα των μέσων Throughput και RTT που εξήχθησαν. Παρακάτω παραθέτουμε τα αποτελέσματα.



Σχ. 4.23 Μέσο Throughput για τις διάφορες τιμές του TCP Window Size

Average Throughput	
TCP Window Size (Kbytes)	Average Throughput (Mb/s)
8	0.45
16	1.03
32	2.16
64	4.03
128	4.37
256	4.4
512	4.41



Σχ. 4.24 Μέσο RTT για τις διάφορες τιμές του TCP Window Size

Average RTT	
TCP Window Size (Kbytes)	Average RTT (ms)
8	116.8
16	116.8
32	117.5
64	125.9
128	235.1
256	276.3
512	277.4

Από τα γραφήματα των Σχ.4.23 και Σχ.4.24, παρατηρούμε τα εξής αναμενόμενα αποτελέσματα. Όσο αυξάνει το “μέγεθος παραθύρου” του χρήστη, τόσο αυξάνεται ο ωφέλιμος ρυθμός μετάδοσης, διότι περισσότερα δεδομένα μεταφέρονται ανά πακέτο. Αντίστοιχα, αναμένεται η καθυστέρηση πλήρους διαδρομής να αυξάνεται όσο μεγαλύτερο Window Size θέτουμε, και αυτό λόγω του γεγονότος ότι τα μεγάλα πακέτα καθυστερούν στη μετάδοση. Το συμπέρασμα που εξάγεται, σύμφωνα με τις μετρήσεις που πραγματοποιήθηκαν, είναι ότι η τιμή 128k (Window Size) μας δίνει έναν καλό συνδυασμό Throughput και RTT. Το Throughput κυμαίνεται στα 4.37 Mb/s, ενώ το RTT στα 235.1 ms. Στόχος μας είναι η λειτουργία ενός αποδοτικότερου δικτύου, με υψηλό ρυθμό μετάδοσης (Throughput) και χαμηλές τιμές καθυστέρησης (RTT).

Επίσης από τα γραφήματα αυτά (Σχ.4.23 και Σχ.4.24), διαπιστώνουμε ότι από την τιμή 128k και πάνω, φτάνουμε σε κορεσμό, δηλαδή δεν έχουμε κάποια σημαντική βελτίωση, αντίθετα το δίκτυο μένει σταθερό.

Μια ακόμη παρατήρηση που θα μπορούσαμε να σχολιάσουμε, είναι η σύγκριση των Throughput και RTT αυτού του σεναρίου με το σενάριο την ενότητας 4.2.2.1, όπου αξιολογούμε επίσης ολόκληρο το δίκτυο, χωρίς όμως να θέσουμε κάποιο συγκεκριμένο Window Size, αλλά αφήνοντας το στην default κατάσταση λειτουργίας. Από τη σύγκριση αυτή παρατηρούμε ότι η default κατάσταση, λειτουργεί ανάμεσα στις τιμές 128k και 512k, δίνοντας μέσο Throughput 4.40 Mb/s και μέσο RTT 273.69 ms.

Στις επόμενες τέσσερις ενότητες παραθέτουμε αναλυτικά, για επιλεγμένα Window Size, τα γραφήματα των Throughput, RTT και Sequence Number, καθώς δεν υπήρχαν αναμεταδώσεις και απώλειες στο δίκτυο.

ΣΧΟΛΙΑΣΜΟΣ ΑΠΟΤΕΛΕΣΜΑΤΩΝ

Throughput

Όπως διαπιστώσαμε και από τα παραπάνω γραφήματα, το Throughput αντιπροσωπεύει το ποσοστό των δεδομένων που μεταφέρονται από το δίκτυο για κάποια χρονική στιγμή. Το throughput εξαρτάται από το διαθέσιμο εύρος ζώνης, και αυτό το εξετάζουμε στην ενότητα 4.2.6. Από τις μετρήσεις που πραγματοποιήθηκαν παρατηρούμε ότι η μέση τιμή του Throughput φτάνει μέχρι τα 4.40 Mb/s, έχοντας θέσει 8 Mb/s.

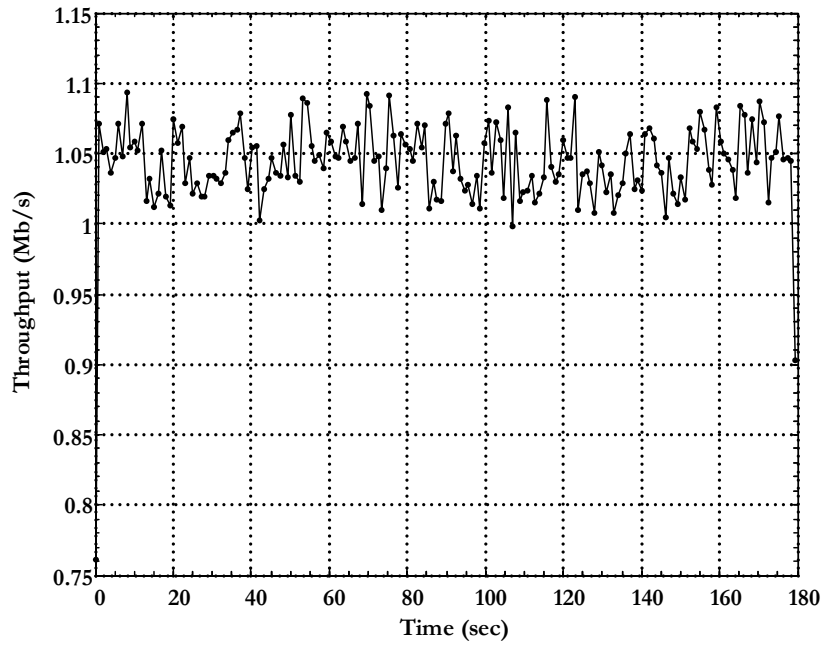
RTT Delay

Το RTT είναι η καθυστέρηση του δικτύου, δηλαδή ο χρόνος που χρειάζεται για να μεταφερθεί ένα πακέτο και να επιστρέψει το Acknowledgment (επιβεβαίωση). Όσο αυξάνεται το μέγεθος του παραθύρου τόσο αυξάνεται το RTT γιατί χρειάζεται περισσότερος χρόνος, για τη μεταφορά του πακέτου. Επίσης, αναμένουμε όσο μικρότερο Bit rate θέτουμε, τόσο μεγαλύτερη καθυστέρηση να έχουμε. Αυτό επιβεβαιώνεται και στην ενότητα 4.2.6.

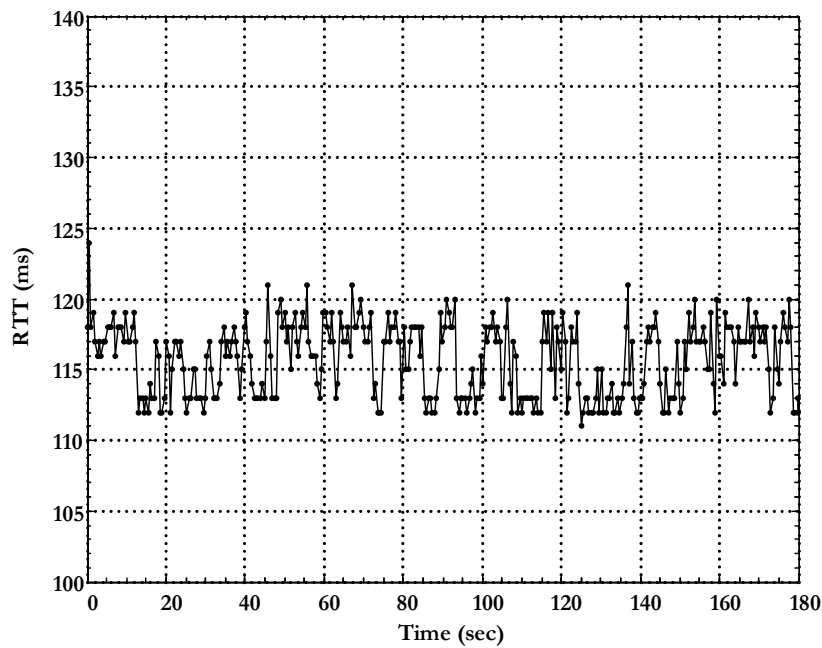
Sequence Number

Στα γραφήματα του Sequence Number φαίνεται η εξέλιξη του αριθμού ακολουθίας των πακέτων TCP στο πέρασμα του χρόνου για συγκεκριμένο χρονικό διάστημα που επιλέξαμε. Όπως παρατηρούμε δεν έχουμε απώλειες πακέτων και αναμεταδόσεις στο δίκτυο μας. Τα μαύρα βέλη αναπαριστούν την αρχή και το τέλος του κάθε πακέτου δεδομένων που στέλνονται σε κάθε χρονική στιγμή, ενώ η κάτω γραμμή, κόκκινη, δείχνει τον αριθμό ακολουθίας των δεδομένων για τα οποία μέχρι τώρα έχει φτάσει επιβεβαίωση (acknowledged data). Η άνω γραμμή, μπλε υπολογίζεται αν προστεθεί στην γραμμή επιβεβαιώσεων (κόκκινη) το μέγεθος παραθύρου του client (window size) και ουσιαστικά δείχνει το μέγιστο όριο της απόστασης-διαφοράς μεταξύ του αριθμού ακολουθίας του τελευταίου επιβεβαιωμένου πακέτου και αυτού που μόλις εκπέμφθηκε. Η διαφορά αυτή ισούται με το ποσό των δεδομένων που έχουν σταλεί και βρίσκονται ανεπιβεβαίωτα στο δίκτυο.

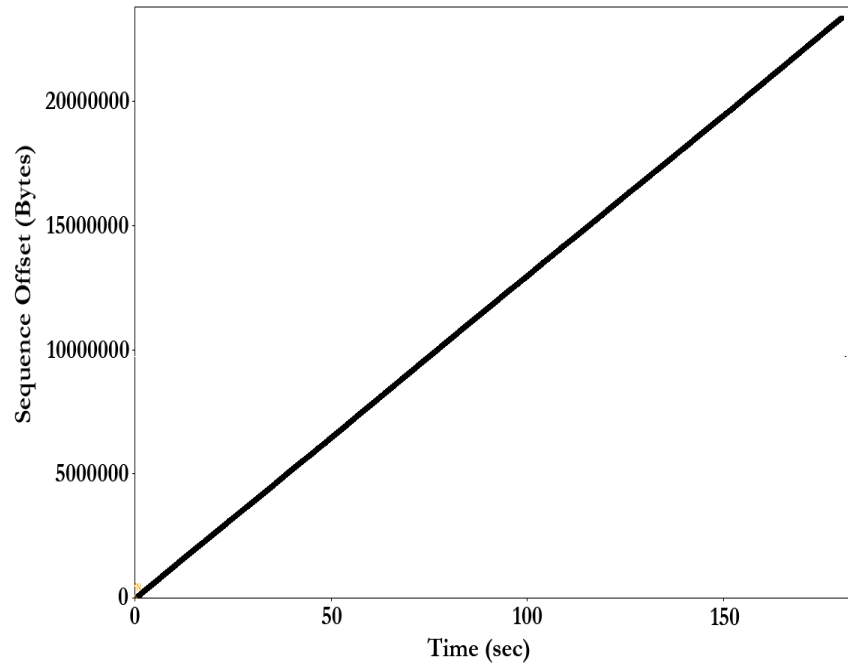
4.2.3.1. Αξιολόγηση Δικτυακής Κίνησης με Window Size 16k



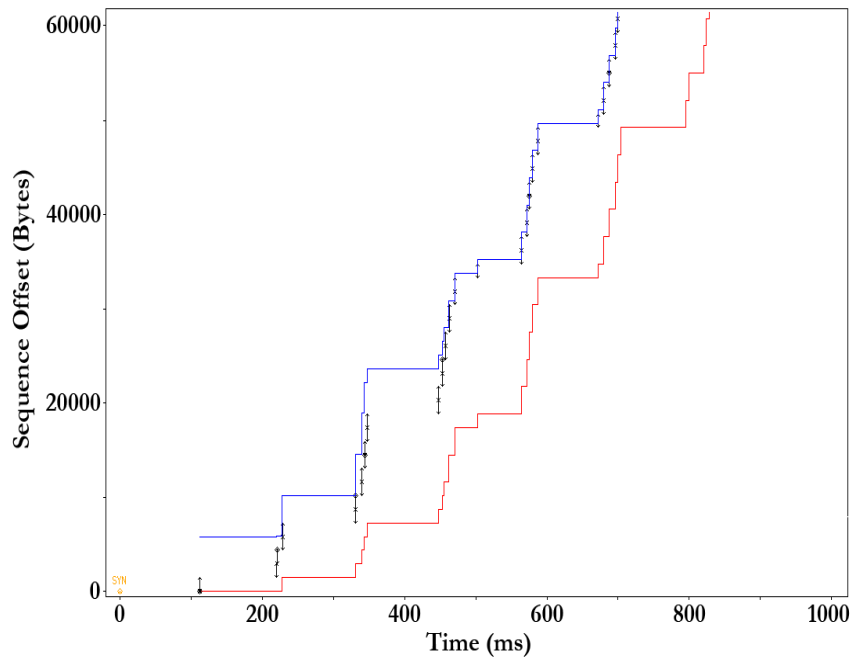
Σχ. 4.25 Throughput για 16k Window Size



Σχ. 4.26 RTT για 16k Window Size

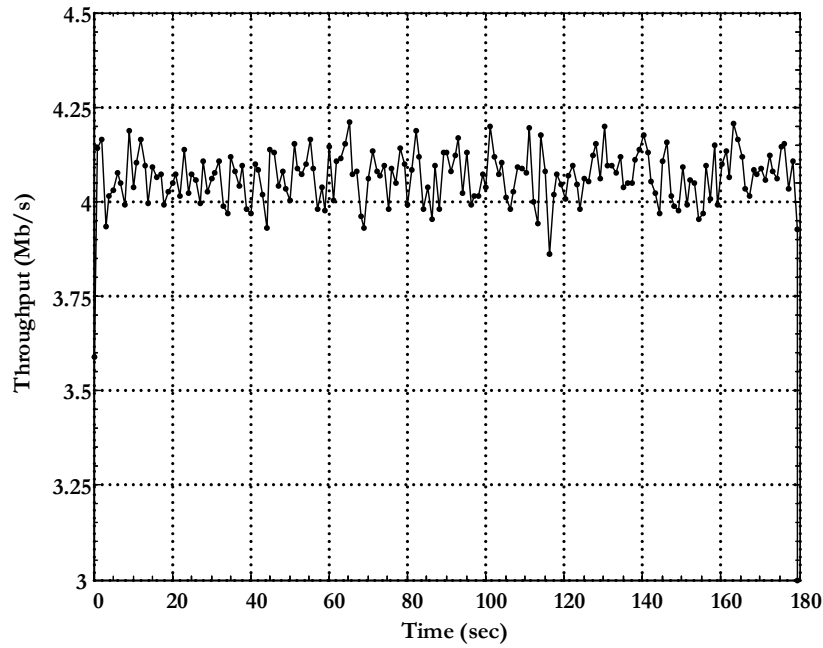


Σχ. 4.27 Sequence Number $\gamma\alpha$ 16k Window Size

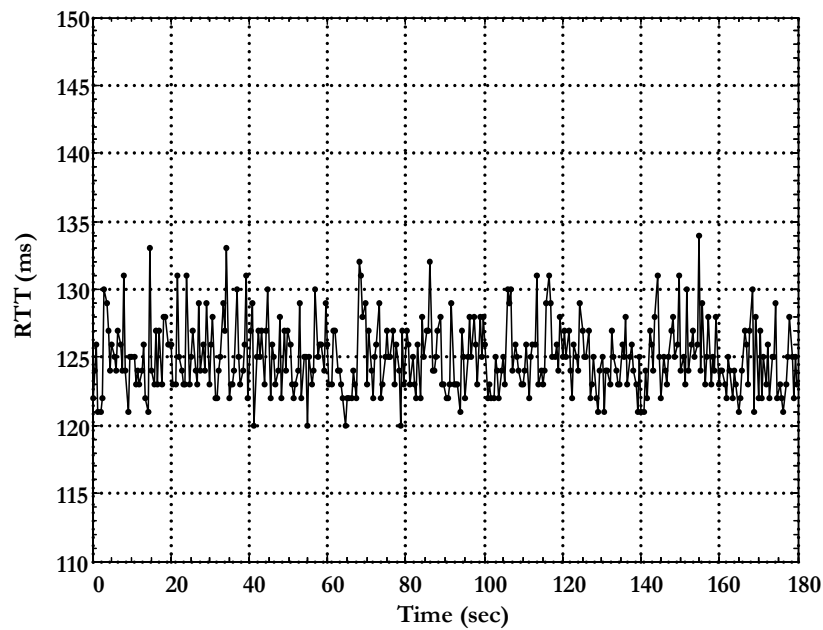


Σχ. 4.28 Sequence Number Zoom $\gamma\alpha$ 16k Window Size

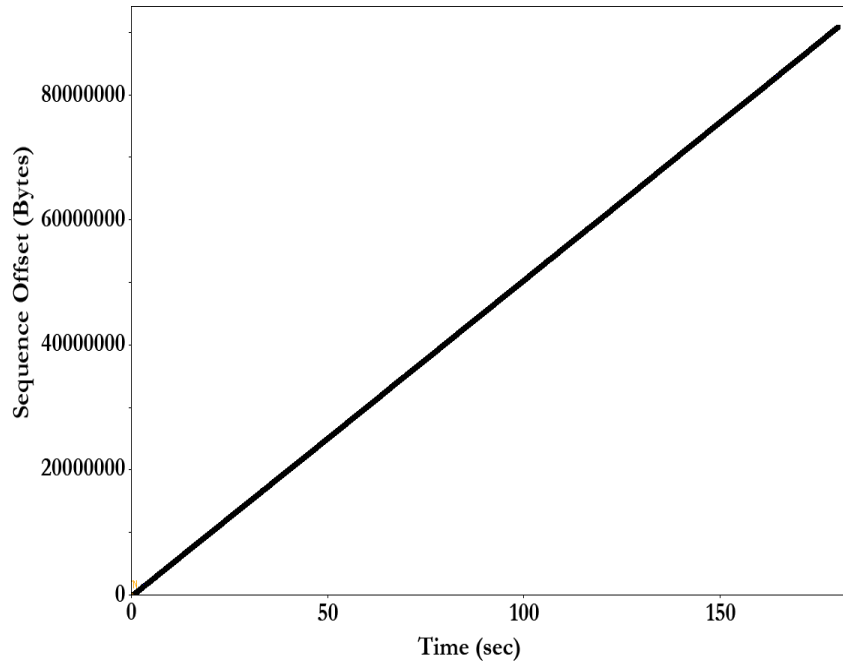
4.2.3.2. Αξιολόγηση Δικτυακής Κίνησης με Window Size 64k



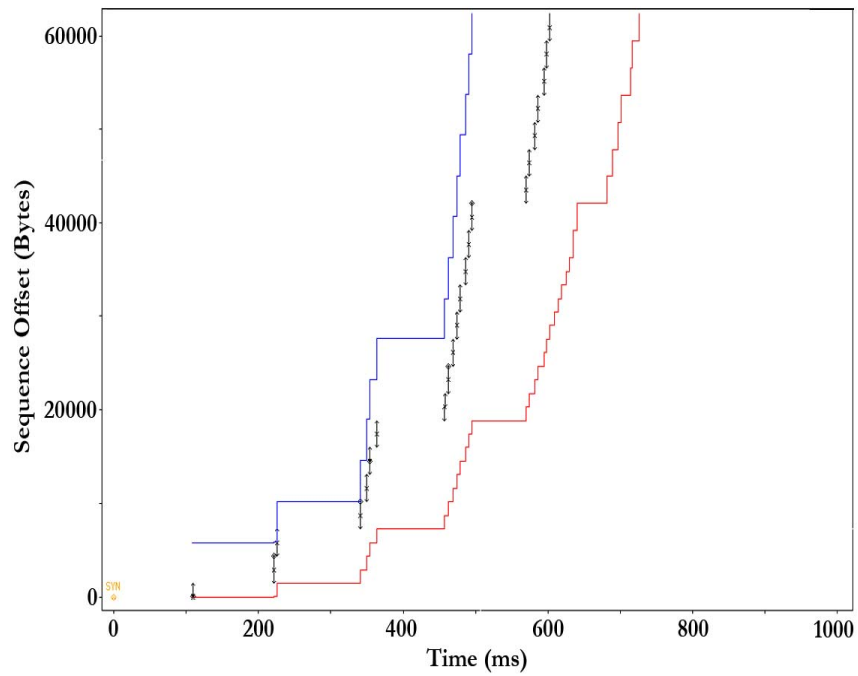
Σχ. 4.29 Throughput για 64k Window Size



Σχ. 4.30 RTT για 64k Window Size

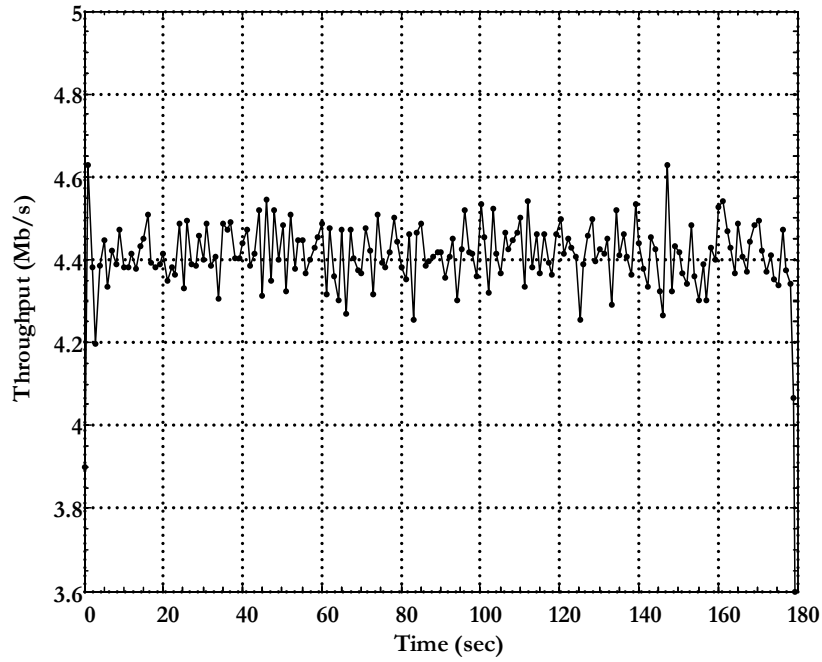


Σχ. 4.31 Sequence Number $\gamma\alpha$ 64k Window Size

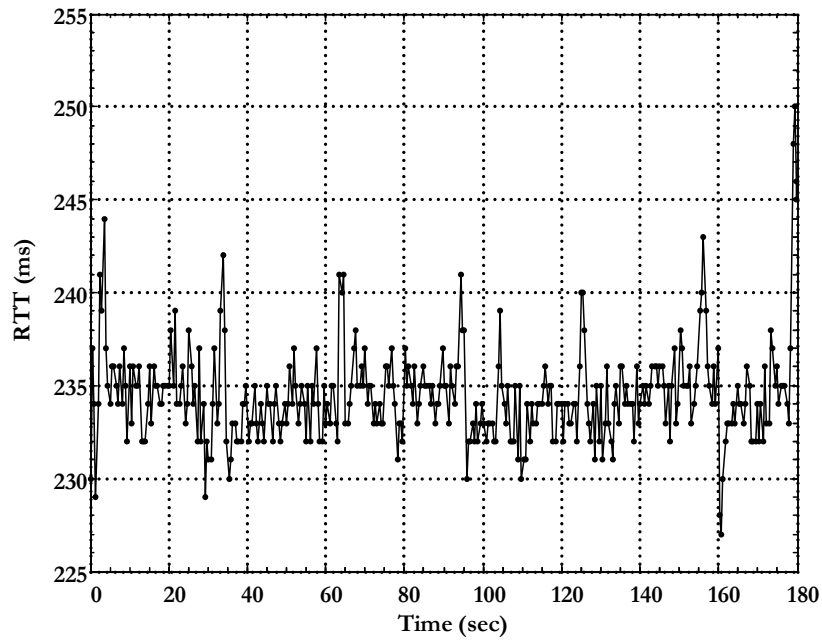


Σχ. 4.32 Sequence Number Zoom $\gamma\alpha$ 64k Window Size

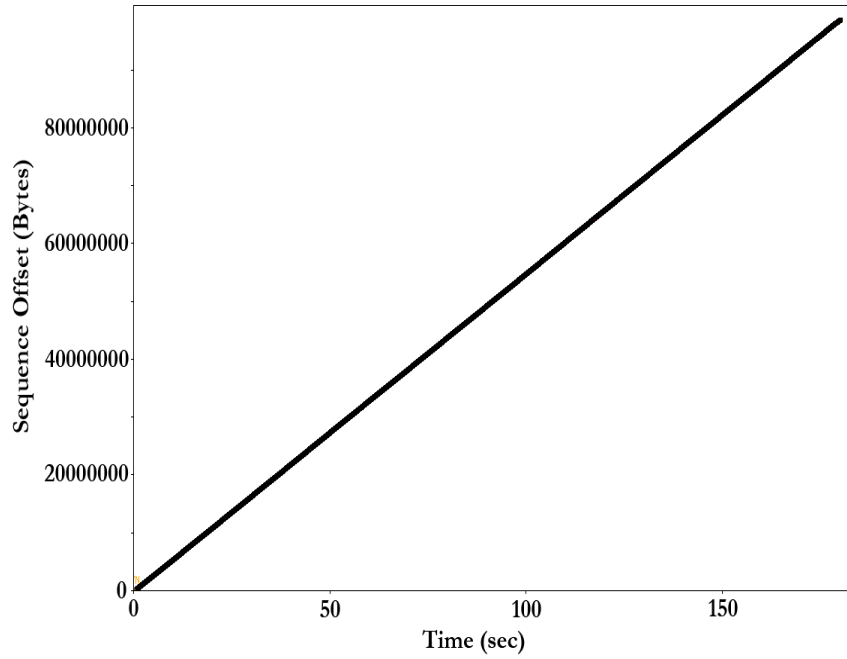
4.2.3.3. Αξιολόγηση Δικτυακής Κίνησης με Window Size 128k



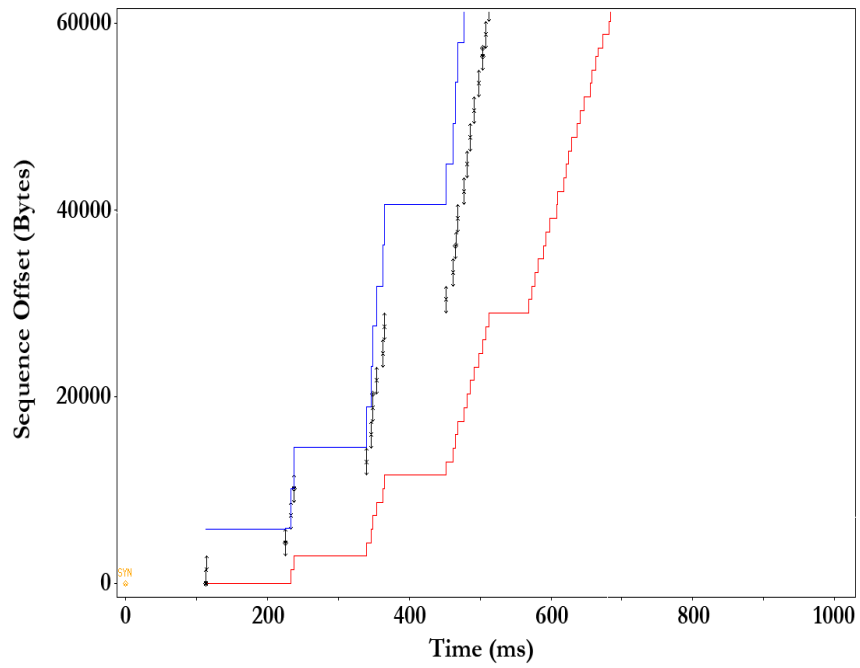
Σχ. 4.33 Throughput για 128k Window Size



Σχ. 4.34 RTT για 128k Window Size

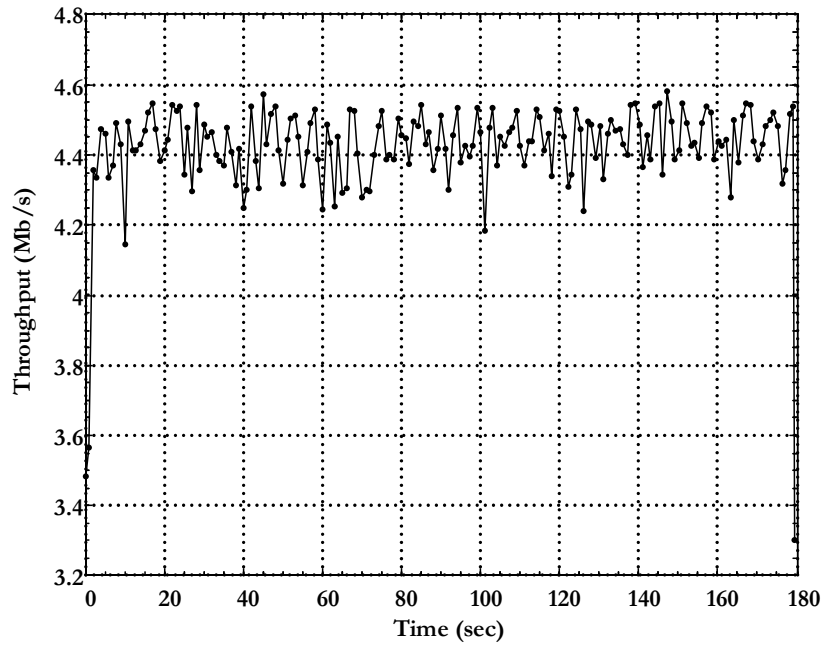


Σχ. 4.35 Sequence Number για 128k Window Size

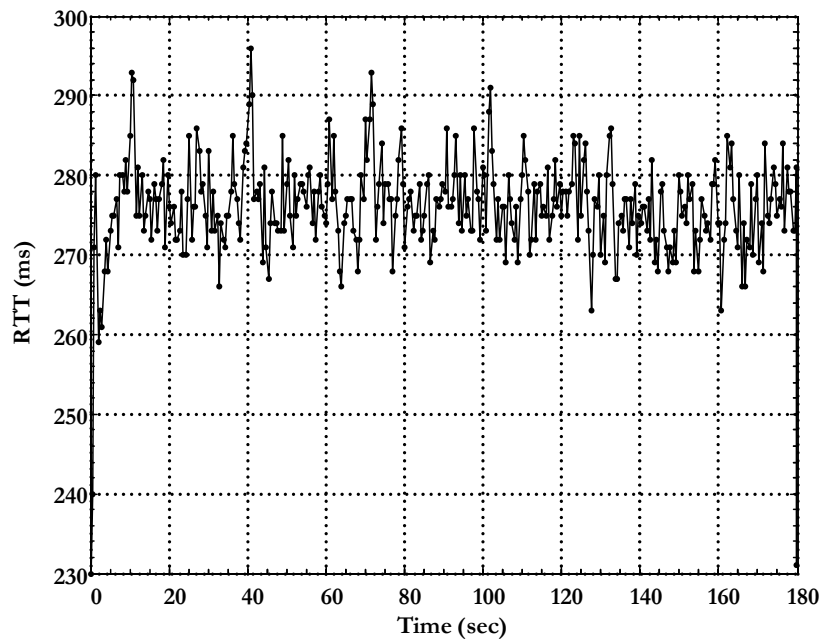


Σχ. 4.36 Sequence Number Zoom για 128k Window Size

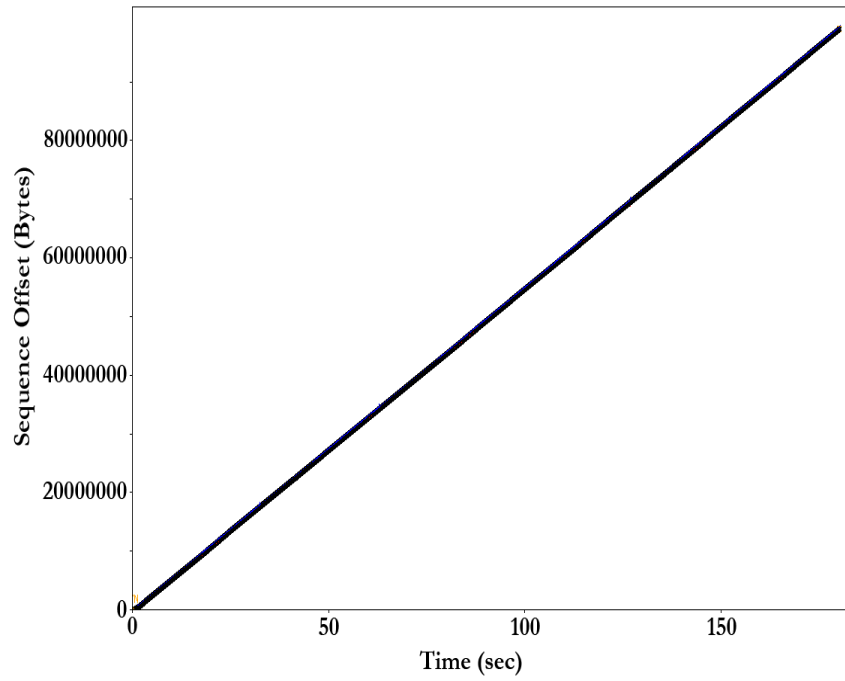
4.2.3.4. Αξιολόγηση Δικτυακής Κίνησης με Window Size 512k



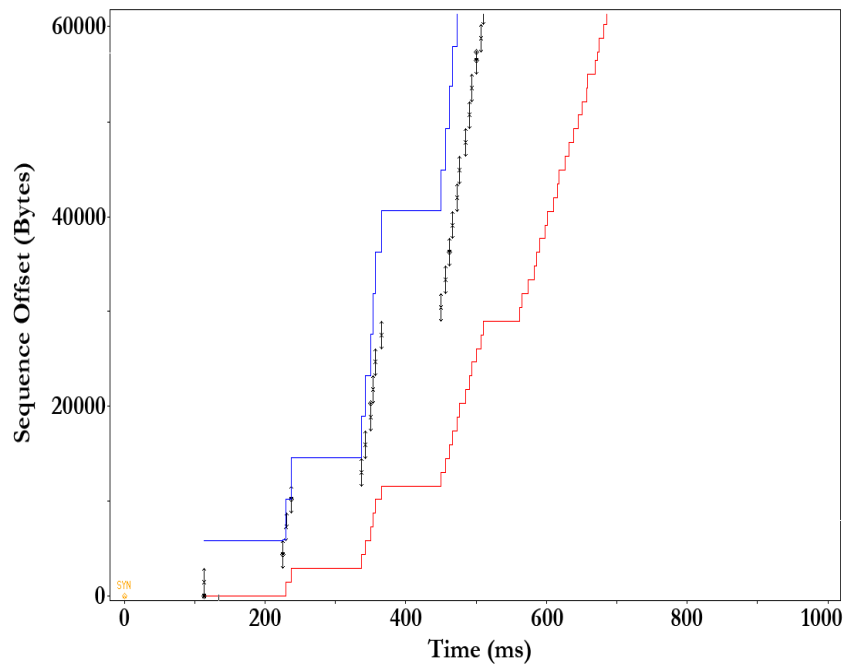
Σχ. 4.37 Throughput για 512k Window Size



Σχ. 4.38 RTT για 512k Window Size



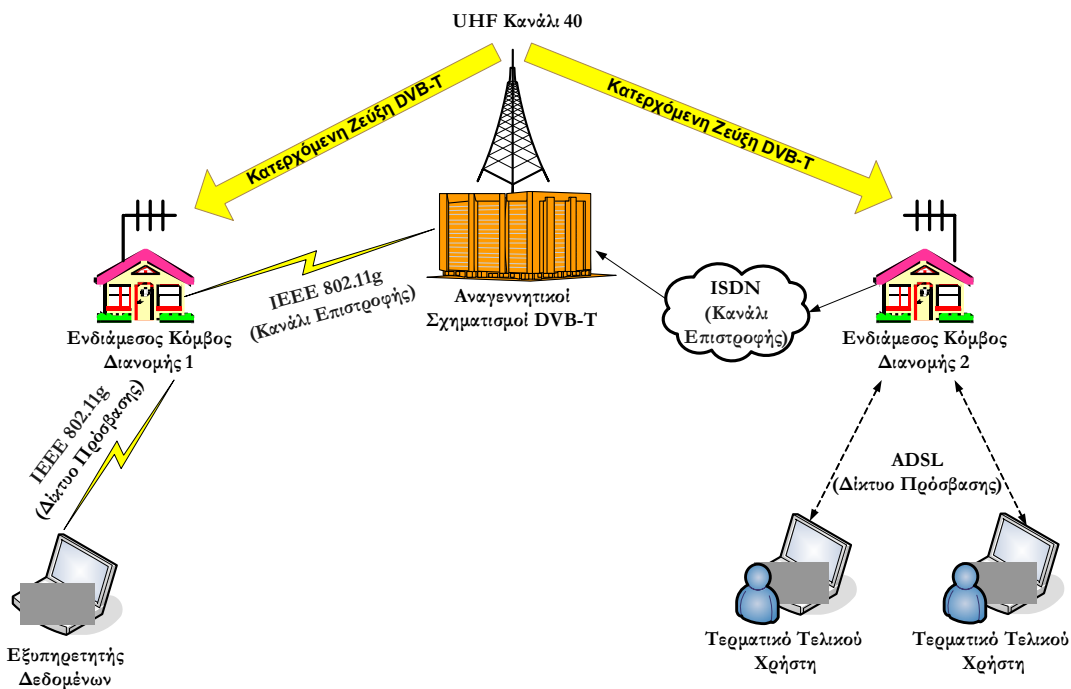
Σχ. 4.39 Sequence Number $\gamma\alpha$ 512k Window Size



Σχ. 4.40 Sequence Number Zoom $\gamma\alpha$ 512k Window Size

4.2.4. Αξιολόγηση Δικτυακής Κίνησης με Δύο Ταυτόχρονους Χρήστες

Το σενάριο του Σχ. 4.41, έχει σκοπό να αξιολογήσει το δίκτυο DVB-T/ISDN, όταν στον CMN στον οποίο έχουν πρόσβαση οι τελικοί χρήστες (CMN2 στο σενάριο), συνδέονται δύο χρήστες. Η ιδέα είναι η εξής, να αξιολογήσουμε τον ωφέλιμο ρυθμό μετάδοσης (Throughput) για τα 60 πρώτα δευτερόλεπτα όταν είναι ενεργός ο πρώτος χρήστης, στη συνέχεια (επόμενα 60 δευτερόλεπτα) όταν είναι ενεργοί και οι δύο χρήστες και τέλος, (τελευταία 60 δευτερόλεπτα) όταν είναι ενεργός ο δεύτερος χρήστης.

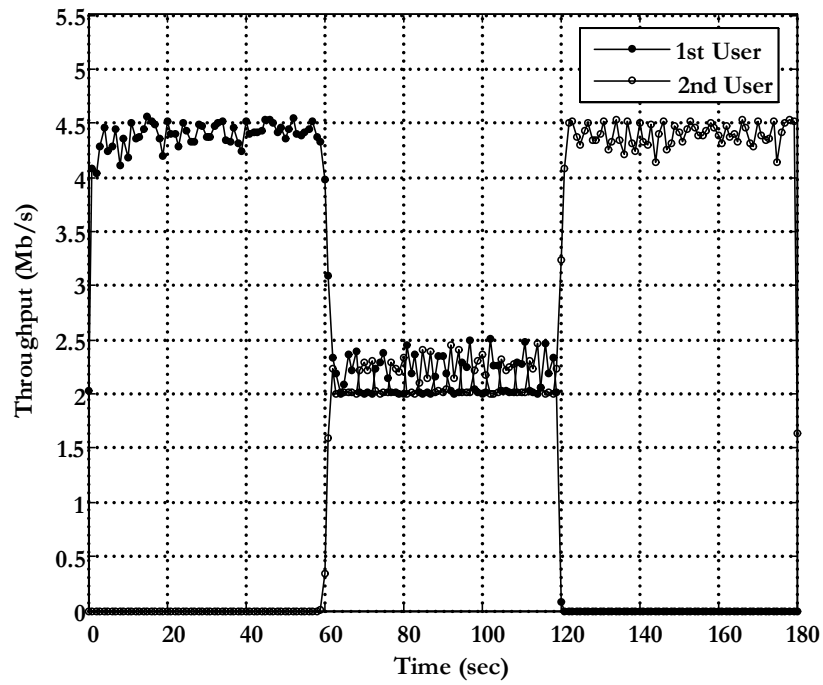


Σχ. 4.41 Αρχιτεκτονική Δικτύου με Δύο Χρήστες

Από το γράφημα του Σχ.4.42, παρατηρούμε αναμενόμενα αποτελέσματα. Στις περιπτώσεις που έχουμε μόνο ένα χρήστη ενεργοποιημένο, διαπιστώνουμε ότι το μέσο Throughput είναι της τάξεως των 4.40 Mb/s, αναμενόμενο αποτέλεσμα σύμφωνα με τις προηγούμενες μετρήσεις, όπου είχαμε μόνο ένα χρήστη και το μέσο Throughput ήταν 4.40 Mb/s.

Στην περίπτωση όμως που έχουμε και τους δύο χρήστες ταυτόχρονα να ζητούν υπηρεσίες μέσω της DVB-T πλατφόρμας, διαπιστώνουμε ότι το μέσο Throughput πέφτει στο μισό, γεγονός αναμενόμενο διότι μοιράζονται τον ωφέλιμο ρυθμό. Για την ακρίβεια ο πρώτος χρήστης

χρησιμοποιεί τα 2,17 Mb/s από τα 4.40 Mb/s, που είναι η τιμή όταν είναι ενεργός μόνο ένας χρήστης, ενώ ο δεύτερος 2.15 Mb/s από τα 4.40 Mb/s.

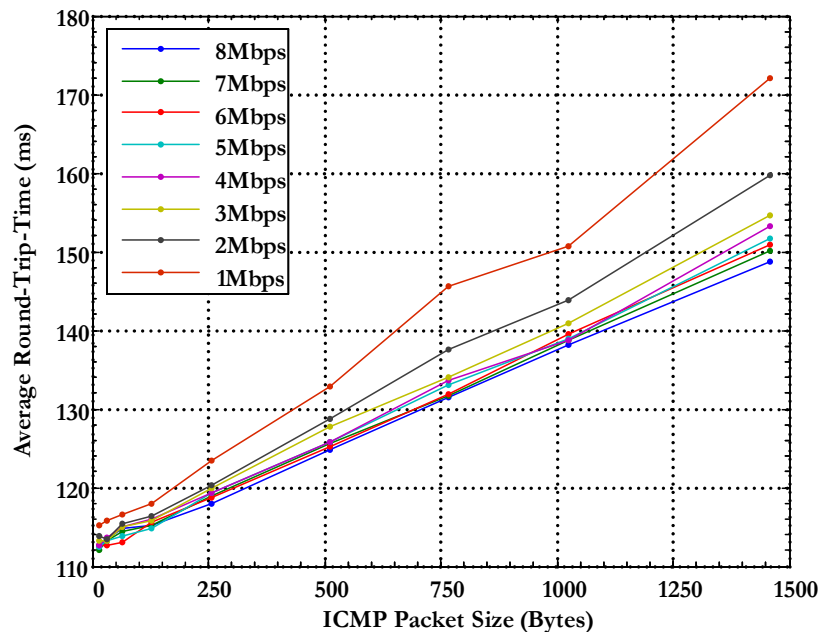


Σχ. 4.42 Throughput για Δύο ADSL Χρήστες

4.2.5. Αξιολόγηση Δικτυακής Κίνησης Μεταβαλλόντας το Μέγεθος του ICMP Packet

Ένα άλλο σενάριο για την αξιολόγηση του δικτύου DVB-T/ISDN, περιλαμβάνει μετρήσεις πλήρους διαδρομής μέσω ICMP request/ replies (“ping”), με τα πακέτα echo request να κατευθύνονται από το χρήστη στον εξυπηρετητή δεδομένων μέσω του δικτύου ISDN, και τις απαντήσεις (echo replies), να επιστρέφουν μέσω της κατερχόμενης ζεύξης DVB-T.

Τα αποτελέσματα παρουσιάζονται στο Σχ.4.43. Συγκεκριμένα, στο σενάριο αυτό μεταβάλλαμε την τιμή του Bit rate στον Amber από 8 Mb/s μέχρι 1 Mb/s και μετρήσαμε την μέση τιμή της καθυστέρησης πλήρους διαδρομής αλλάζοντας το ICMP Packet Size με τις εξής τιμές, 16, 32, 64, 128, 256, 512, 768, 1024, 1460 Bytes.



Σχ. 4.43 Μέσο RTT για Διαφορετικό ICMP Packet Size και Διαφορετικό Bit Rate

Παρατηρούμε ότι όσο το Bit rate μειώνεται, τόσο μεγαλύτερη είναι η μέση καθυστέρηση πλήρους διαδρομής. Επίσης, όσο μικρότερο είναι το μέγεθος του ICMP Packet, τόσο μικρότερη είναι και η μέση καθυστέρηση πλήρους διαδρομής.

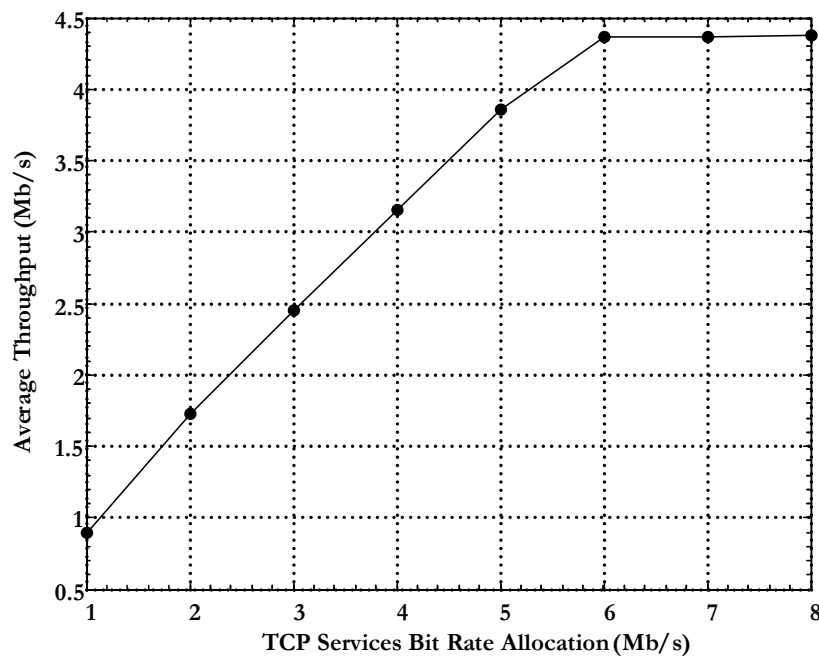
Στον πίνακα που ακολουθεί αναφέρονται οι τιμές του μέσου RTT για τις διαφορετικές τιμές του Bit rate που θέσαμε στον Amber.

Average RTT	
ICMP Packet Size (Bytes)	Average RTT (ms)
1	132.21
2	127.65
3	126.01
4	125.34
5	124.76
6	124.44
7	124.41
8	124.14

4.2.6. Αξιολόγηση Δικτυακής Κίνησης Μεταβαλλόντας το Bit Rate στον Amber

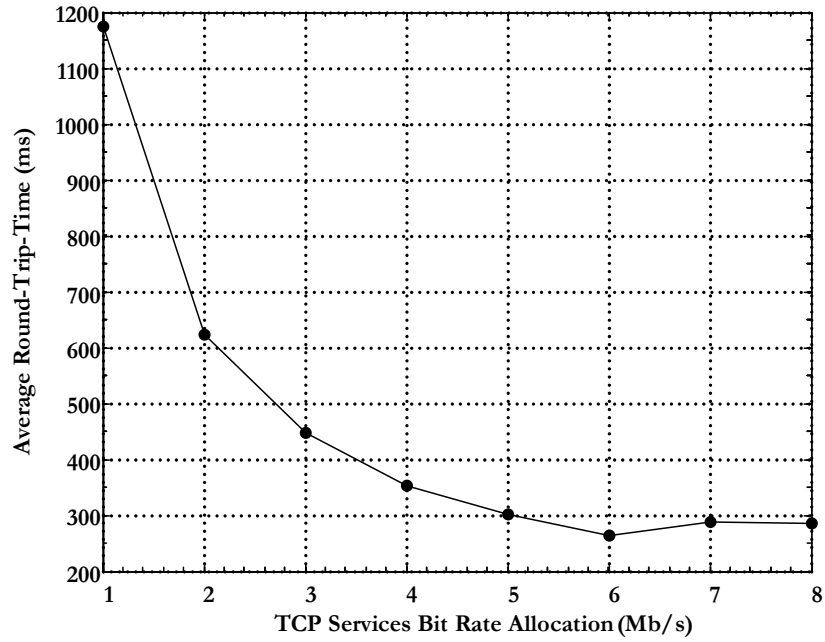
Τα επόμενα δύο γραφήματα (Σχ.4.44 και Σχ.4.45), έχουν σκοπό να αξιολογήσουν το δίκτυο DVB-T/ISDN, βάση μετρήσεων μέσω Throughput και RTT σε σχέση με τη μεταβολή του Bit rate στον Amber. Οι τιμές στον Amber κυμαίνονται από 1 Mb/s έως και 8 Mb/s.

Από τα γραφήματα (Σχ.4.44 και Σχ.4.45), διαπιστώνουμε, ότι όσο αυξάνεται το Bit rate, τόσο αυξάνεται το μέσω Throughput, ενώ το μέσω RTT μειώνεται. Τα αποτελέσματα επιβεβαιώνονται και από τα γραφήματα του Σχ.4.2 και Σχ.4.3 (με Bit Rate 8 Mb/s), όπου το μέσω Throughput είναι 4.40 Mb/s και το μέσω RTT είναι 273.69 ms. Στους παρακάτω πίνακες οι τιμές των γραφημάτων φαίνονται αναλυτικά.



Σχ. 4.44 Μέσο Throughput για διαφορετικό Bit Rate

Average Throughput	
Bit Rate Allocation (Mb/s)	Average Throughput (Mb/s)
1	0.9
2	1.73
3	2.45
4	3.15
5	3.86
6	4.36
7	4.37
8	4.38



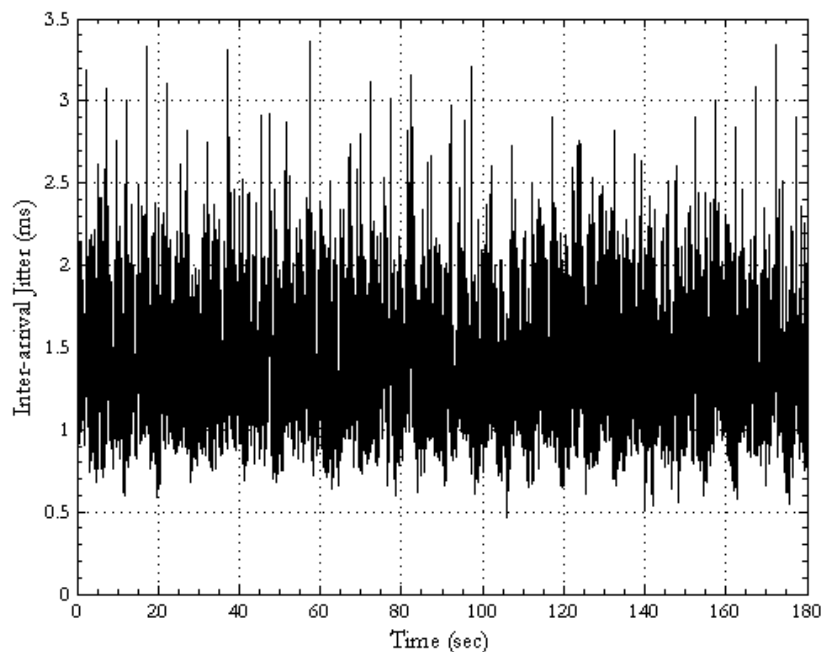
Σχ. 4.45 Μέσο RTT για διαφορετικό Bit Rate

Average RTT	
Bit Rate Allocation (Mb/s)	Average RTT (ms)
1	1175.5
2	622
3	448.3
4	353.5
5	301.1
6	263.2
7	287.4
8	285.1

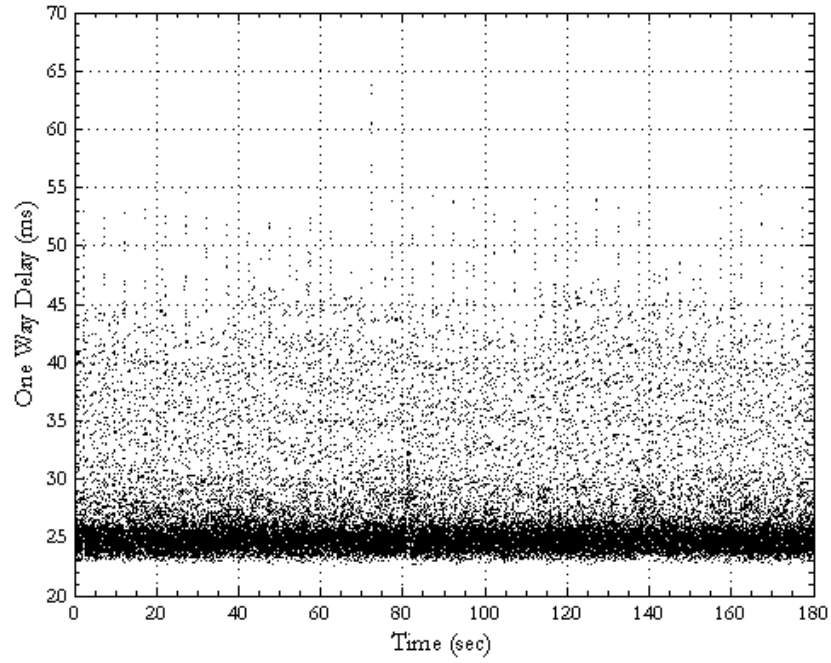
4.3. Παρουσίαση Αποτελεσμάτων και Γραφικών Παραστάσεων Δικτυακής Κίνησης UDP

Στην ενότητα αυτή περιγράφεται ένα δίκτυο, το οποίο προσφέρει υπηρεσίες ψηφιακής τηλεόρασης με την χρήση του πρωτοκόλλου Internet μέσω ενός ευρυζωνικού δικτύου από έναν εξυπηρετητή δεδομένων, κάνοντας χρήση του δικτύου ψηφιακής τηλεόρασης ως κανάλι καθόδου. Η δικτυακή κίνηση είναι UDP και γίνεται χρήση των ενδιάμεσων κόμβων διανομής (CMNs) για την επικοινωνία του εξυπηρετητή δεδομένων και των τελικών χρηστών.

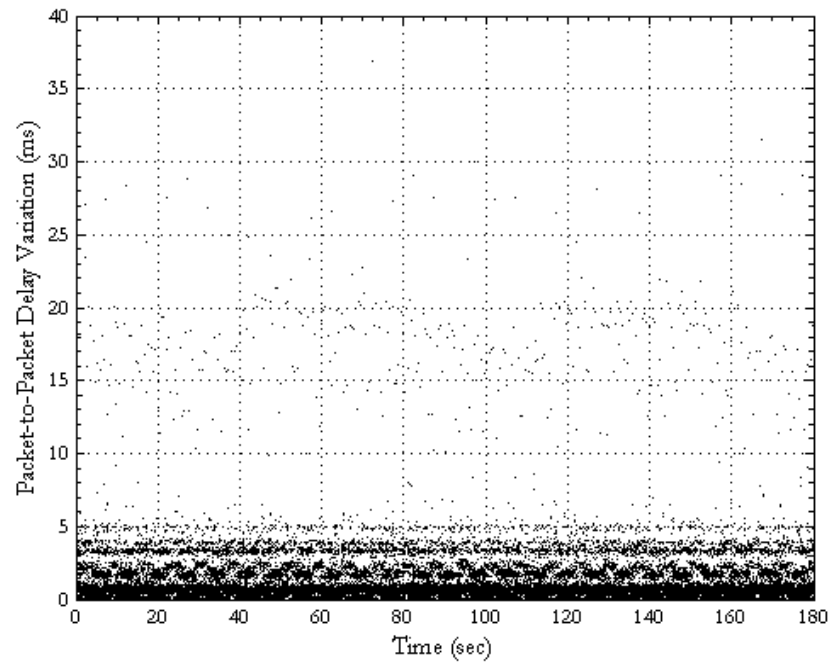
Στα γραφήματα των Σχ.4.46, Σχ.4.47 και Σχ.4.48 παρουσιάζονται τα αποτελέσματα του σεναρίου αυτού. Η μέση τιμή του Inter-arrival Jitter είναι 1.38 ms, το μέσο One way delay είναι 27.12 ms και τέλος η μέση τιμή του Packet to packet delay variation υπολογίστηκε 1.38 ms. Συμπεράνουμε λοιπόν, ότι το δίκτυο που αξιολογήσαμε είναι ικανοποιητικό, αφού οι απώλειες είναι μηδενικές και η μέση καθυστέρηση είναι 27.12 ms, τιμή που δεν ξεπερνά τα 5-6 sec. Αν η μέση καθυστέρηση πλησιάζει τα 5-6 sec, σημαίνει ότι το δίκτυο δεν είναι αξιόπιστο και αποδοτικό.



Σχ. 4.46 Inter – arrival Jitter



Σχ. 4.47 One Way Delay



Σχ. 4.48 Packet-to-Packet Delay Variation

5. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΠΡΟΤΑΣΕΙΣ

Η πτυχιακή εργασία εστίασε την έρευνα πάνω στη χρήση της σύγχρονης και προτυποποιημένης τεχνολογίας της επίγειας ψηφιακής τηλεόρασης (DVB-T) ως ευρυζωνικό δίκτυο πρόσβασης για υπηρεσίες δεδομένων. Μελετήθηκε μία υβριδική αρχιτεκτονική ασύμμετρου δικτύου σε δύο διαφορετικές παραλλαγές, στις οποίες η κατερχόμενη ζεύξη (downlink) υλοποιήθηκε με ένα ευρυζωνικό κανάλι DVB-T ενώ η ανερχόμενη (uplink) παρεχόταν από την ενσύρματη τεχνολογία ISDN και WLAN. Σημαντικό ρόλο στην αρχιτεκτονική των δικτύων είχαν οι ενδιάμεσοι κόμβοι διανομής (CMNs). Εξετάστηκε και αξιολογήθηκε η συμπεριφορά του ενός δικτύου σε μεταδόσεις από σημείο σε σημείο κάνοντας χρήση των πρωτοκόλλων tcp/ip και udp/ip.

Όσον αφορά τα αποτελέσματα των μετρήσεων, κρίνονται αναμενόμενα με βάση την αρχιτεκτονική και τη λειτουργία του δικτύου επίγειας ψηφιακής τηλεόρασης με κανάλι επιστροφής την τεχνολογία ISDN. Παρόλο που πρόκειται για μια τεχνολογία με σημαντικά μειονεκτήματα, ο συνδιασμός της με την τεχνολογία DVB-T είναι αποτελεσματικός και δημιουργείται μια αρχιτεκτονική παροχής πολλαπλών Διαδικτυακών υπηρεσιών και υπηρεσιών πολυμέσων.

Το ερευνητικό αντικείμενο που μελετήθηκε στην παρούσα πτυχιακή εργασία μπορεί να επεκταθεί έτσι ώστε να αποτελέσει αντικείμενο μελλοντικής έρευνας στο τομέα της σύγκλισης των τεχνολογιών ευρυ-εμπομπής και τηλεπικοινωνιών. Σχετικά με το κανάλι επιστροφής τεχνολογίας ISDN που αξιοποιήθηκε στο προτεινόμενο δίκτυο, περαιτέρω διερεύνηση μπορεί να επεκταθεί και να περιλάβει και άλλες ασύρματες και ενσύρματες τεχνολογίες ανάλογα με το είδος του τελικού χρήστη (κινητός, σταθερός).

Παράρτημα Α

Για να μπορέσει ο αναγνώστης να κατανοήσει πλήρως την διεξαγωγή των μετρήσεων αυτών, παρουσιάζεται μία λεπτομερή περιγραφή της δημιουργίας και της ανάλυσης τους, καθώς και των προγραμμάτων που χρησιμοποιήθηκαν.

A.1 Προγράμματα Δημιουργίας και Ανάλυσης της Κίνησης ενός Δικτύου

A.1.1. Δημιουργία Κίνησης

A.1.1.1. Πρόγραμμα Iperf

Το Iperf είναι μια εναλλακτική λύση για τη δημιουργία TCP και UDP κίνησης καθώς και για τη μέτρηση της απόδοσης του εύρους ζώνης ενός δικτύου με TCP ή UDP κίνηση. Η εφαρμογή αυτή εφαρμόζεται τόσο σε Unix λειτουργικά όσο και σε Windows.

Επιτρέπει το συντονισμό διαφόρων παραμέτρων, μεγεθών και χαρακτηριστικών όπως μέτρηση εύρους ζώνης, απώλεια πακέτων, διακύμανση καθυστέρησης (jitter), μέγεθος αναφορών MSS/MTU και μεγέθη διαβασμένων παρατηρήσεων. Υποστηρίζει TCP window size μέσω των socket buffers, πολλαπλές ταυτόχρονες συνδέσεις μεταξύ server και client, δημιουργία UDP κίνησης με διευκρινισμένο εύρος ζώνης, multicast κίνηση και ικανότητα IPv6. Όλες οι παράμετροι πρέπει να διευκρινιστούν με το K (kilo-) και το M (mega-).

Σε καθορισμένο χρόνο μεταφοράς μπορεί να τρέξει περισσότερα από ένα σύνολα στοιχείων, επιλέγει τις καλύτερες μονάδες για τα μεγέθη της αναφοράς, και χρειάζεται πολλές περισσότερες συνδέσεις για τον server, ο οποίος μπορεί να οργανωθεί και ως daemon. Τυπώνει κατά περιόδους το μέσο εύρος ζώνης, το jitter, και τις απώλειες αναφοράς σε συγκεκριμένα χρονικά διαστήματα. Τέλος, χρησιμοποιεί αντιπροσωπευτικά streams που εξετάζουν πώς η συμπίεση στο στρώμα συνδέσεων έχει επιπτώσεις στο εύρος ζώνης του δικτύου.

A.1.1.2. Πρόγραμμα Multi-Generator (MGEN)

Το πρόγραμμα MGEN είναι ένα λογισμικό ανοικτού κώδικα, και αποτελεί δημιουργία του Εργαστηρίου Έρευνας Naval (NRL) και συγκεκριμένα της ερευνητικής ομάδας που ασχολείται με τα πρωτόκολλα προηγμένων δικτύων. Το MGEN παρέχει τη δυνατότητα να εκτελεστούν

μετρήσεις για την απόδοση δικτύων που παρέχουν IP υπηρεσίες, χρησιμοποιώντας UDP/IP κίνηση. Το σύνολο των εργαλείων του, δημιουργεί κίνηση σε πραγματικό χρόνο έτσι ώστε το δίκτυο να μπορεί να φορτωθεί με ποικίλους τρόπους. Η κίνηση που δημιουργείται μπορεί επίσης να παραληφθεί και να καταγραφεί για τις αναλύσεις. Χρησιμοποιούνται script αρχεία, για να καθοδηγήσουν τα αρχεία που δημιουργούνται, φορτώνοντας patterns καθ'όλη τη διάρκεια της κίνησης. Αυτά τα scripts χρειάζονται για την προσομοίωση unicast ή και multicast εφαρμογών UDP/IP. Τα log στοιχεία του MGEN μπορούν να χρησιμοποιηθούν για να υπολογίσουν τις στατιστικές απόδοσης όσον αφορά το throughput, τα packet loss rates, το communication delay, τα ποσοστά απώλειας πακέτων, την καθυστέρηση επικοινωνίας και άλλα. Το MGEN αυτή την περίοδο εκτελείται τόσο σε διάφορα λειτουργικά τύπου Unix όσο και σε πλατφόρμες WIN32. Τέλος, κάποιες εκδόσεις του MGEN υποστηρίζουν graphical user interface.

A.2.1. Ανάλυση Κίνησης

A.2.1.1. Πρόγραμμα Tcpdump

Το Tcpdump είναι μια εύχρηστη εφαρμογή ανάλυσης της κίνησης ενός δικτύου και εκτελείται σε γραμμή εντολών. Επιτρέπει στους χρήστες να ανιχνεύσουν και να εμφανίσουν κάθε είδους κίνηση, στον υπολογιστή που είναι εγκατεστημένο το πρόγραμμα, καθώς επίσης και την κίνηση πακέτων από άλλα υποδίκτυα.

Το Tcpdump έχει γραφτεί από τους Van Jacobson, Craig Leres και Steven McCanne. Συνεργάζεται με τις πιο πολλές Unix εφαρμογές όπως είναι οι Linux, Solaris, BSD, Mac OS X, HP-UX και AIX. Σε τέτοια συστήματα το tcpdump βασίζεται στην libpcap, η οποία είναι μια βιβλιοθήκη που περιέχει τις απαραίτητες ρουτίνες για να χρησιμοποιηθεί σαν φίλτρο πακέτων.

A.2.1.2. Πρόγραμμα Tcptrace

Το Tcptrace είναι ένα χρήσιμο εργαλείο, γραμμένο από τον Shawn Ostermann στο Ohio University, και χρησιμοποιείται για την ανάλυση TCP dump αρχείων. Δέχεται πακέτα που έχουν δημιουργηθεί από διάφορα packet-capture προγράμματα όπως είναι το tcpdump, snoop, etherpeek, HP Net Metrix, και WinDump. Το tcptrace μπορεί να παράγει αρχεία τα οποία περιέχουν διαφόρους τύπους πληροφοριών για κάθε υπαρκτή κίνηση. Όπως για παράδειγμα χρόνος εξομοίωσης, bytes που έχουν σταλεί και παραληφθεί, retransmissions, round trip times, window advertisements, throughput, και άλλα. Μπορεί επίσης να παράγει γραφικές παραστάσεις για κάθε μια από τις παραπάνω πληροφορίες, για περαιτέρω ανάλυση.

A.2.1.3. Πρόγραμμα Xpl2gpl

Το xpl2gpl είναι ένα AWK script που μετατρέπει τα αρχεία που προκύπτουν από το tcptrace και είναι τύπου .xpl σε αρχεία τύπου .gpl. Έτσι έχουμε τη δυνατότητα να εισάγουμε τα παραγόμενα .gpl αρχεία στο Excel και να δημιουργήσουμε τις γραφικές παραστάσεις τους.

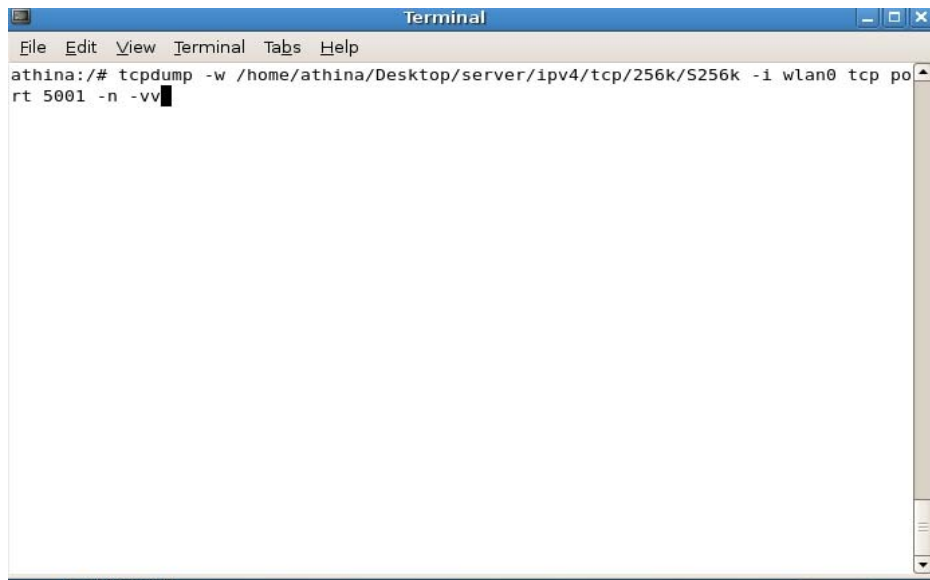
A.2.1.4. Προγράμματα της Γλώσσας Προγραμματισμού Perl

Η Perl είναι μια πρακτική και δυναμική, ταυτόχρονα γλώσσα προγραμματισμού που δανείζεται τα χαρακτηριστικά της γνωρίσματα από τη C, τη shell scripting (sh), AWK, sed, Lisp, και από πολλές άλλες γλώσσες προγραμματισμού. Προγράμματα, που ο αλγόριθμός τους είναι δομημένος με βάση τη γλώσσα αυτή, χρησιμοποιούνται για τον υπολογισμό της διακύμανσης της καθυστέρησης των πακέτων (jitter).

A.2 Παράδειγμα Δημιουργίας και Ανάλυσης της Κίνησης ενός Δικτύου

A.2.1. Δημιουργία Δικτυακής Κίνησης TCP

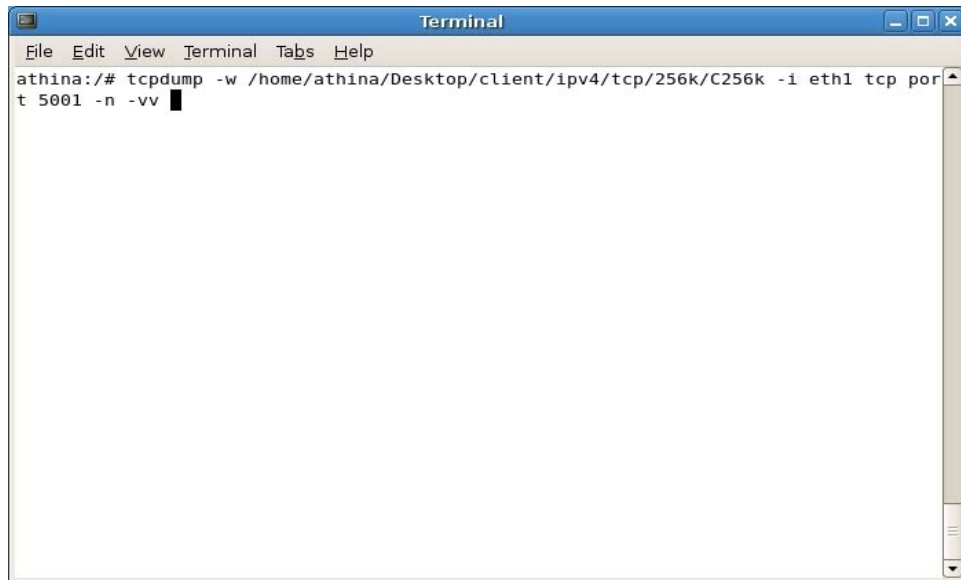
Παρακάτω θα αναλυθεί η δημιουργία και η ανάλυση μιας TCP κίνησης έτσι ώστε να γίνει κατανοητή η λειτουργία των εφαρμογών που προαναφέρθηκαν. Πριν το στάδιο της δημιουργίας της κίνησης μας πρέπει να ενεργοποιήσουμε το πρόγραμμα tcpdump τόσο στον Server (Σχ.Α.1) όσο και στον Client (Σχ.Α.2) για να μπορεί να ανιχνεύσει όλη την κίνηση (sniffing), πριν αρχίσει η ροή της. Τα στοιχεία της κίνησης αυτής θα καταχωρούνται στο αρχείο με το όνομα S256k και C256k για τον server και client αντίστοιχα.

A terminal window titled "Terminal" with a menu bar containing "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal content shows the command:

```
athina:/# tcpdump -w /home/athina/Desktop/server/ipv4/tcp/256k/S256k -i wlan0 tcp port 5001 -n -vv
```

 with a cursor at the end of the line.

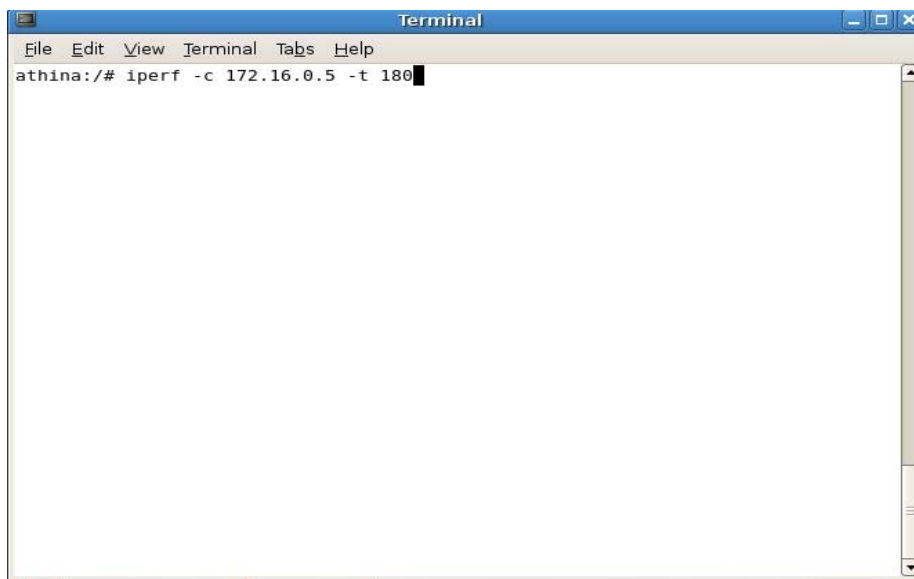
Σχ. Α.1 Εντολή tcpdump στον Server

A terminal window titled "Terminal" with a menu bar containing "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal content shows the command:

```
athina:/# tcpdump -w /home/athina/Desktop/client/ipv4/tcp/256k/C256k -i eth1 tcp port 5001 -n -vv
```

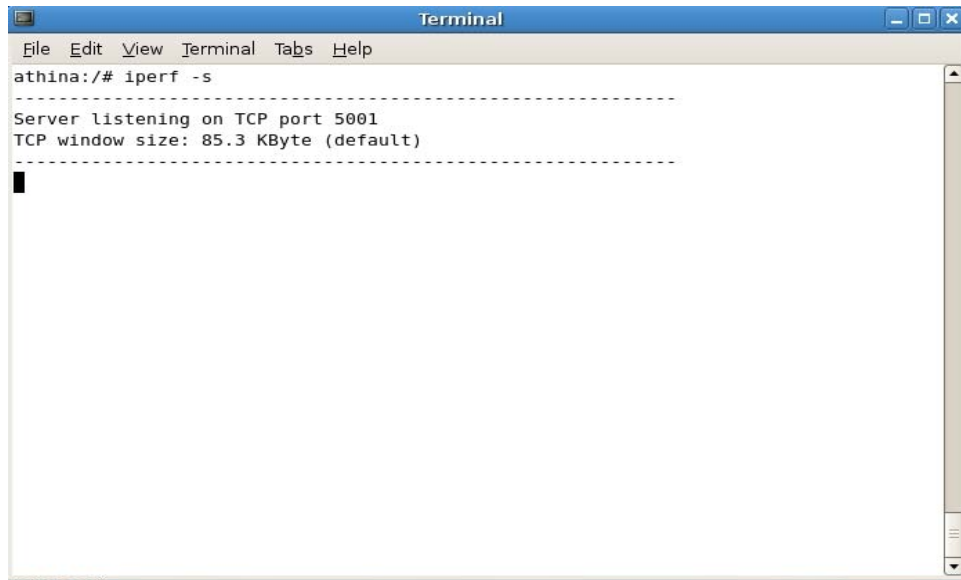
Σχ. A.2 Εντολή tcpdump στον Client

Κι αφού τώρα είναι έτοιμα τα φίλτρα της κίνησης, μπορούμε να δημιουργήσουμε την ροή της. Έτσι στον Server, όπως φαίνεται και στο screenshot του Σχ.Α.3 εκτελείται η εντολή του iperf για να δημιουργήσει δικτυακή κίνηση προς τον Client για χρονικό διάστημα 180 sec. Ενώ για τον client παρουσιάζεται στο screenshot του Σχ. Α.4.

A terminal window titled "Terminal" with a menu bar containing "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal content shows the command:

```
athina:/# iperf -c 172.16.0.5 -t 180
```

Σχ. A.3 Εντολή iperf στον Server



Σχ. Α.4 Εντολή iperf στον Client

Μετά το τέλος της κίνησης αυτής, παρατηρούμε ότι αναφέρονται τα ανιχνευμένα πακέτα τα οποία έχουν σταλθεί από τον Server.

A.2.1.1. Ανάλυση Δικτυακής Κίνησης TCP

Στο επόμενο στάδιο έχουμε την ανάλυση αυτής κίνησης. Η ανάλυση πρέπει να γίνει σε υπολογιστή με λειτουργικό σύστημα Linux. Η διαδικασία ακολουθεί στα παρακάτω βήματα.

1. Μετατρέπουμε το dump αρχείο σε txt

```
tcpdump -lwr 128k > 128k.txt
```

2. Στην TCP κίνηση μετράω στον :

Client: Throughput


Server: RTT, Retransmission, Sequence Number

3. Throughput (time)

```
tcpdump -zxy -xtraffic'-B -i1' 128k (dump αρχείο client)
```


4. Throughput (segments)

Το dump αρχείο που έχουμε



128k

Το txt αρχείο που δημιουργήσαμε, για να δούμε το report με τα αποτελέσματα στατιστικά.



128k.txt

OPTIONS

- l: δημιουργία εκτενούς αναφοράς
- w: αναφορά για τη συμφόρηση του παραθύρου (TCP)
- r: εμφάνιση του rtt

tcptrace -zxy -T -y -A100 128k (dump αρχείο client)

5. **RTT (samples)**

tcptrace -zxy -xtraffic'-R -i0.5' 128k (dump αρχείο server)

6. **RTT (segments)**

tcptrace -zxy -R 128k (dump αρχείο server)

7. **Retransmission**

tcptrace -zxy -xtraffic'-L -i1' 128k (dump αρχείο server)

8. **Time Sequence**

tcptrace -zxy -S 128k (dump αρχείο server)

9. Μετατρέπω το .xpl → .gpl αρχείο, ώστε στη συνέχεια να κάνω τη γραφική

xpl2gpl -s a2b_tput.xpl [Κάθε φορά χρησιμοποιώ το αντίστοιχο .xpl αρχείο]

10. Τέλος χρησιμοποιούμε xplot, gnuplot, matlab, excel ή ότι άλλο θέλουμε για να εμφανίσουμε τη γραφική.

Στη συνέχεια περιγράφεται λεπτομερώς η λειτουργία του κάθε option, που χρησιμοποιήθηκαν κατά τη διαδικασία της ανάλυσης.

THROUGHPUT	RTT
<ul style="list-style-type: none">-zxy: κάνει plot και τους δύο άξονες από το μηδέν-xtraffic: Για να βγάλει Throughput σε time-B: εμφάνιση του Throughput (Time)-T: εμφάνιση του Throughput (segment)-i1: κάθε 1 sec να βάζει ένα spot-y: Δημιουργία αρχείου με στιγμιαίο throughput-A100: 100 δείγματα σε ένα segment	<ul style="list-style-type: none">-zxy: κάνει plot και τους δύο άξονες από το μηδέν-xtraffic: Για να βγάλει RTT σε samples-R: εμφάνιση του RTT (Time & segment)-i0.5: κάθε 0.5 sec να βάζει ένα spot
	RETRANSMISSION & SEQUENCE NUMBER
	<ul style="list-style-type: none">-zxy: κάνει plot και τους δύο άξονες από το μηδέν-L: εμφάνιση των Retransmissions-S: εμφάνιση του Time Sequence

Κάνοντας ανάλυση, μας εμφανίζονται διάφορα αρχεία κάθε φορά, όμως θα πρέπει να επιλέγουμε το σωστό προκειμένου να δημιουργήσουμε το επιθυμητό γράφημα.

- Όταν θέλουμε Throughput σε Time χρησιμοποιούμε το Blue.dot αρχείο, το οποίο μας δίνει το Average Throughput. Δηλαδή έχοντας θέσει $-i1$ κάθε sec θα εμφανίζεται ένα spot με το average throughput. Οπότε για 3 min λειτουργίας του δικτύου αναμένουμε 180 spot.



traffic_bytes.datase
t.blue.dot

- Όταν θέλουμε Throughput σε Segments χρησιμοποιούμε το Yellow.dot αρχείο, το οποίο μας δίνει το στιγμιαίο Instantaneous Throughput για κάθε πακέτο που μεταφέρεται. Οπότε για 3 min λειτουργίας του δικτύου θα πάρουμε αρκετές χιλιάδες spot. Μπορούμε επίσης να χρησιμοποιήσουμε το Red.dot ώστε να πάρουμε το Average throughput για τα δείγματα ανά segment που ορίζουμε με το $-A100$ ή $-A50$ κτλ. Αν δεν ορίσουμε πόσα δείγματα θέλουμε, τότε το πρόγραμμα χρησιμοποιεί την default τιμή.

0.114988 208.00	0.715576 195512
0.119663 308021	0.927955 674981
0.224631 13718.	1.124316 725276
0.226111 972972	1.371219 577781
0.229066 487309	1.634094 540485
0.230776 840405	1.877242 584335
0.333786 139791	2.124860 578116
0.335503 338672	2.389487 539083
0.337721 649233	2.660351 528070
0.339437 839169	2.924097 539928
0.341193 820045	3.200289 514424

a2b_tput.
dataset.yellow.
dot

a2b_tput.
dataset.red.dot

- Όταν θέλουμε RTT σε Time χρησιμοποιούμε το Blue.dot αρχείο, το οποίο μας δίνει το Average RTT. Δηλαδή έχοντας θέσει $-i0.5$ κάθε μισό sec θα εμφανίζεται ένα spot με το average RTT. Οπότε για 3 min λειτουργίας του δικτύου αναμένουμε περίπου 360 spot.

```
0.000000 123.00
0.503403 146.00
1.008928 219.00
1.517586 226.00
2.021019 234.00
2.527623 241.00
3.028507 247.00
3.529893 244.00
4.032605 237.00
4.536667 235.00
5.037831 234.00
```

traffic_rtt.
dataset.blue.
dot

- Όταν θέλουμε RTT σε Segments χρησιμοποιούμε το Red.dot αρχείο, το οποίο μας δίνει το RTT για κάθε πακέτο που μεταφέρεται. Οπότε για 3 min λειτουργίας του δικτύου θα πάρουμε αρκετές χιλιάδες spot. Αν θέλουμε χρησιμοποιούμε το Script **ipv6_rtt.pl** του Αργύρη για να μας βγάξει το Average RTT ανά δοθέν χρονικό διάστημα.

```
0.000000 141.00
0.503403 184.00
1.008928 237.00
1.517586 239.00
2.021019 251.00
2.527623 284.00
3.028507 253.00
3.529893 264.00
4.032605 249.00
4.536667 248.00
5.037831 251.00
```

traffic_rtt.
dataset.red.dot

- Για το Sequence Number χρησιμοποιούμε το a2b_tsg.xpl.



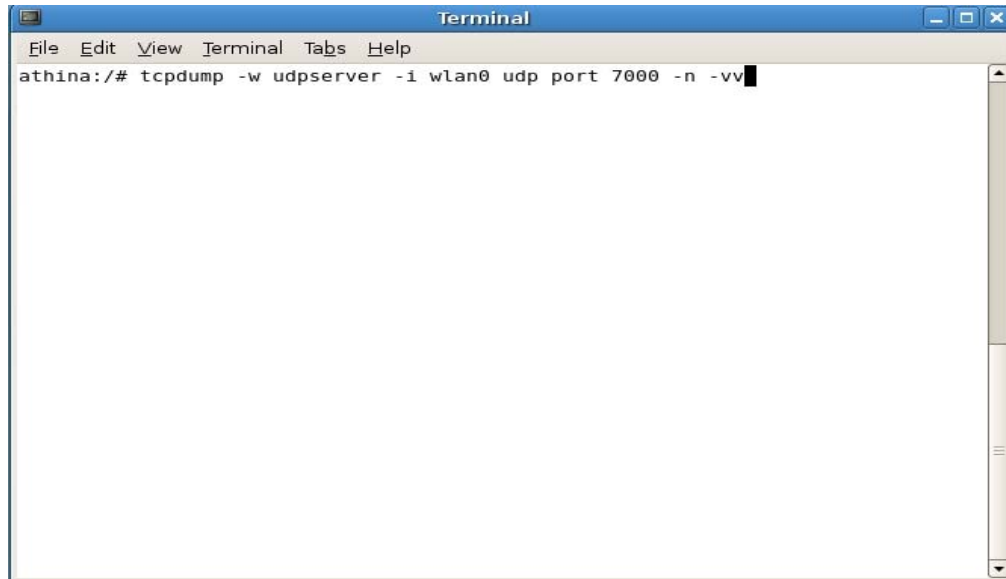
a2b_tsg.xpl

A.2.2. Δημιουργία Δικτυακής Κίνησης UDP

Για να δημιουργηθεί αυτή η δικτυακή κίνηση χρειάζεται όπως είδαμε και προηγουμένως το πρόγραμμα Mgen. Και για να ανιχνευθεί και να εμφανιστεί χρειάζεται το πρόγραμμα tcpdump στον Server και στον Client.

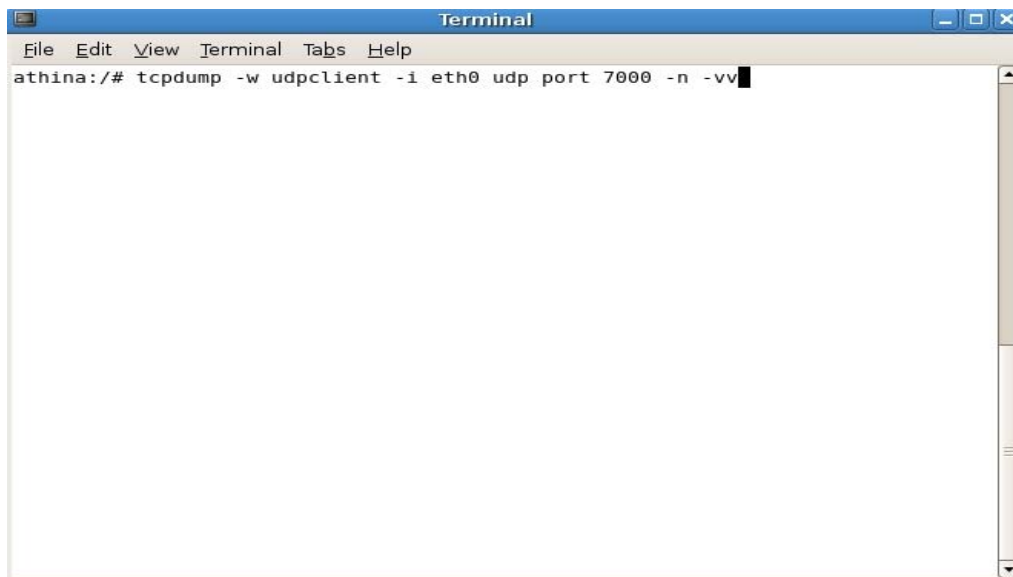
Αρχικά πρέπει ο Server, όπως φαίνεται και στο screenshot του Σχ.Α.5, να εκτελέσει την εντολή του tcpdump, για να μπορεί να ανιχνεύσει όλη την κίνηση (sniffing), πριν αρχίσει η ροή της. Τα στοιχεία της κίνησης αυτής θα καταχωρούνται στο αρχείο με το όνομα S256k. Αντίστοιχα και ο

Client, όπως φαίνεται και στο screenshot του Σχ.Α.6, εκτελεί την εντολή tcpdump, για να μπορεί να ανιχνεύσει όλη την κίνηση (sniffing), πριν αρχίσει η ροή της. Τα στοιχεία της κίνησης αυτής θα καταχωρούνται στο αρχείο με το όνομα C256k.



```
athina:/# tcpdump -w udpserver -i wlan0 udp port 7000 -n -vv
```

Σχ. Α.5 Εντολή tcpdump στον Server

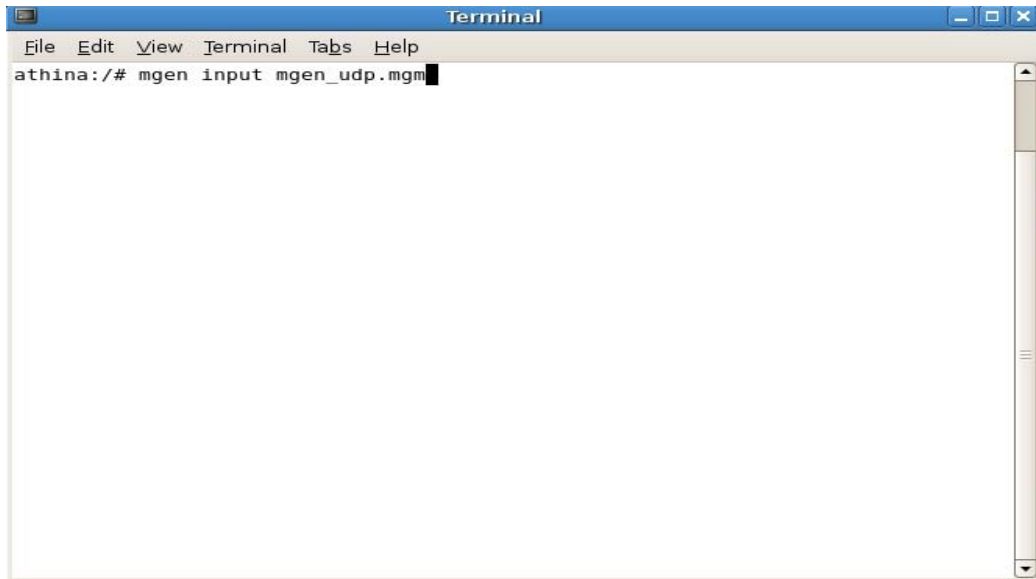


```
athina:/# tcpdump -w udpclient -i eth0 udp port 7000 -n -vv
```

Σχ. Α.6 Εντολή tcpdump στον Client

Κι αφού τώρα είναι έτοιμα τα φίλτρα της κίνησης, μπορούμε να δημιουργήσουμε την ροή της. Έτσι στον Server, όπως φαίνεται και στο screenshot του Σχ.Α.7, εκτελείται η εντολή του Mgen για

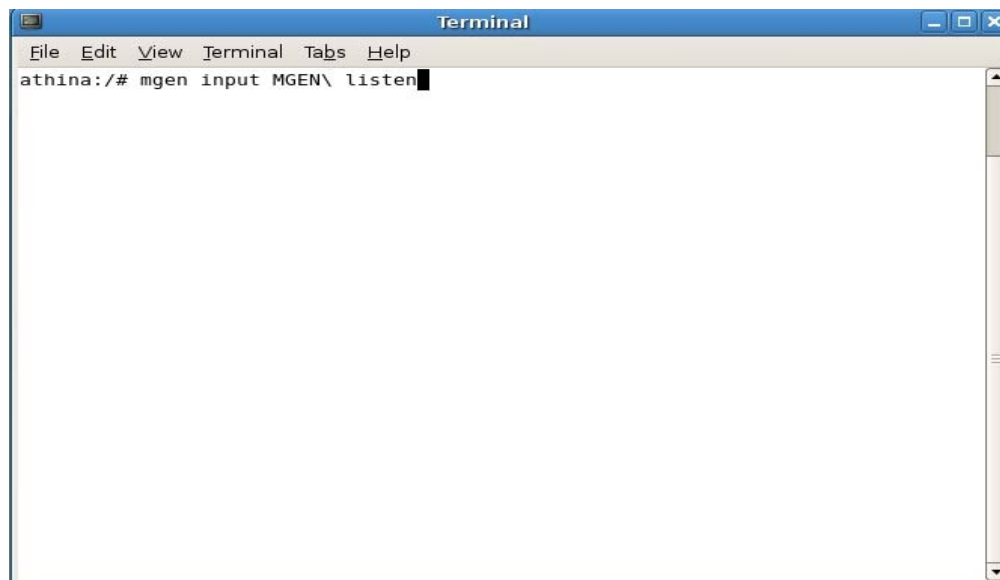
να δημιουργήσει δικτυακή κίνηση προς τον Client για χρονικό διάστημα 180 sec. Ομοίως και στον Client, όπως φαίνεται και στο screenshot του Σχ.Α.8, εκτελείται η αντίστοιχη εντολή του Mgen για να δεχτεί την δικτυακή κίνηση από τον Server.



```
athina:/# mgen input mgen_udp.mgm
```

Σχ. Α.7 Εντολή Mgen στον Server

Σε αυτό το σημείο πρέπει να σημειωθεί ότι το `mgen_udp.mgm` είναι ένα αρχείο που διαβάζεται από το `mgen` και περιέχει όλες της πληροφορίες που χρειάζεται το πρόγραμμα αυτό για τη δημιουργία της κίνησης μας, όπως είναι ο χρόνος εκτέλεσης και το bit rate (σχήμα).



```
athina:/# mgen input MGEN\ listen
```

Σχ. Α.8 Εντολή Mgen στον Client

Τελικά, έχουμε την δημιουργία μιας δικτυακής κίνησης από τον Server στον Client η οποία ανιχνεύεται και στους δύο υπολογιστές.

A.2.2.1. Ανάλυση Δικτυακής Κίνησης UDP

Στο επόμενο στάδιο έχουμε την ανάλυση αυτής κίνησης. Η ανάλυση πρέπει να γίνει σε υπολογιστή με λειτουργικό σύστημα Linux. Η διαδικασία ακολουθεί στα παρακάτω βήματα.

1. Μετατρέπουμε τα dump αρχεία σε txt

```
tcpdump -tt -vv -n -r udpServer > udp_sender.txt  
tcpdump -tt -vv -n -r udpClient > udp_receiver.txt
```

Θα ήταν καλό να σώζουμε τα .txt με τα ονόματα που φαίνονται δίπλα, διότι στα script υπάρχουν αυτά τα ονόματα

Αυτά είναι τα ονόματα των dump files των Client & Server αντίστοιχα

2. Script replicid (ipv4_replicid.pl)

Σε κάθε αρχείο θα πρέπει τα id να είναι μοναδικά και με τη βοήθεια αυτού του προγράμματος γίνεται ο έλεγχος για την μοναδικότητά τους.

OPTIONS

-tt: Print an unformatted timestamp on each dump line
-vv: Even more verbose output. For example, additional fields are printed from NFS reply packets.
-n: Don't convert addresses to names.
-r: Read packets from *file* (which was created with the -w option). Standard input is used if *file* is `-'.

3. Script createedfiles (ipv4_createendfiles.pl)

Δημιουργεί δύο αρχεία (κίνηση sender & receiver)

4. Script losses (ipv_all_losses.pl)

Βρίσκει τις απώλειες κατά την ενεργό δράση του δικτύου

5. Script throughput (ipv_all_sender_receiver_rate.pl)

Βρίσκει το datarate των sender & receiver

6. Script align for jitter (ipv_all_align_for_delay_jitt.pl)

Ευθυγραμμίζει κάθε πακέτο του αρχείου του receiver με κάθε πακέτο του αρχείου του sender, ώστε στη συνέχεια να πάρουμε το Jitter

7. Script timestamp (ipv_all_timestamp.pl)

Δημιουργεί δύο αρχεία με τους χρόνους των sender & receiver για τον υπολογισμό του Jitter

8. Script smoothed jitter (RFC 3550) (ipv_all_inter_arrival_jitter.pl)

9. Script timestamp one way delay (ipv_all_one_way_delay.pl)

10. Script Jitter (ipv_all_jitter.pl)

Υπολογίζει το jitter

Τα script χρησιμοποιούνται με την παραπάνω σειρά.

Τα script μπορούν να χρησιμοποιηθούν για όλες τις κινήσεις (π.χ. mgen & vlc) .

Θα χρησιμοποιήσουμε τα αρχεία (**one_way_delayvstime** μας δίνει το one way delay, **timed_final_jitter** μας δίνει το smooth jitter, **jittervstime** μας δίνει το jitter) για τις γραφικές παραστάσεις.

Παράρτημα Β

Ακολουθούν οι κώδικες των προγραμμάτων, της γλώσσας προγραμματισμού Perl, που χρησιμοποιήθηκαν για την ανάλυση της UDP κίνησης.

B.1. SCRIPT REPLICID THE PERL

```
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 2 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307,
# USA.

#!/usr/bin/perl -w

open(AFIL, "<udp_sender.txt") || die ("cannot open udp_sender
file\n");
open(BFILE, "<udp_receiver.txt") || die ("cannot open udp_receiver file
\n");
my ($temp);
my ($temp1);
my (@rec_lines) = <AFIL>;

    $start = time;

for ($j=0;$j<@rec_lines;$j++)
{
    @udp_id=split(/\t +/,$rec_lines[$j]);
    for ($k=0;$k<@udp_id;$k++)
    {
        $temp=0;
        if ($udp_id[$k] eq "cid")
        {
            $temp=$udp_id[$k+1];
            last;
        }
    }
}

for ($k=0;$k<@udp_id;$k++)
```

```

{
    if ($udp_id[$k] eq "seq")
    {
        $temp=$temp.($udp_id[$k+1]);
        last;
    }
}
for ($k=0;$k<@udp_id;$k++)
{
    if ($udp_id[$k] eq "ser")
    {
        $temp=$temp.($udp_id[$k+1]);
        last;
    }
}
for ($k=0;$k<@udp_id;$k++)
{
    if ($udp_id[$k] eq "id")
    {
        $temp=$temp.($udp_id[$k+1]);
        last;
    }
}
}
#print "$temp\n";
for ($i=$j+1;$i<@rec_lines;$i++)
{
    @udp_id_compare=split(/\t +/, $rec_lines[$i]);
    for ($k=0;$k<@udp_id;$k++)
    {
        $temp1=0;
        if ($udp_id_compare[$k] eq "cid")
        {
            $temp1=$udp_id_compare[$k+1];
            last;
        }
    }
    for ($k=0;$k<@udp_id;$k++)
    {
        if ($udp_id_compare[$k] eq "seq")
        {
            $temp1=$temp1.($udp_id_compare[$k+1]);
            last;
        }
    }
}
}

```

```

for ($k=0;$k<@udp_id;$k++)
{
    if ($udp_id_compare[$k] eq "ser")
    {
        $temp1=$temp1.($udp_id_compare[$k+1]);
        last;
    }
}
for ($k=0;$k<@udp_id;$k++)
{
    if ($udp_id_compare[$k] eq "id")
    {
        $temp1=$temp1.($udp_id_compare[$k+1]);
        last;
    }
}
#print "$temp1\n";
if ($temp eq $temp1)
{
    print "Found two instances (sender file) of $temp1 at line $i $j\n";
    exit(12);
}
}

#print "$sen_line\n";

}

$elapsed_sec = time - $start;

my $second = $elapsed_sec%60;
my $minute = ($elapsed_sec/60)%60;
my $hour = ($elapsed_sec/(60*60))%24;
    print "Time elapsed for senders file: $hour hours:$minute min:$second
sec\n";

close(AFILE);

my (@rec_lines) = <BFILE>;

for ($j=0;$j<@rec_lines;$j++)
{

@udp_id=split(/\t +/,$rec_lines[$j]);

```

```

for ($k=0;$k<@udp_id;$k++)
{
    $temp=0;
    if ($udp_id[$k] eq "cid")
    {
        $temp=$udp_id[$k+1];
        last;
    }
}

for ($k=0;$k<@udp_id;$k++)
{
    if ($udp_id[$k] eq "seq")
    {
        $temp=$temp.($udp_id[$k+1]);
        last;
    }
}

for ($k=0;$k<@udp_id;$k++)
{
    if ($udp_id[$k] eq "ser")
    {
        $temp=$temp.($udp_id[$k+1]);
        last;
    }
}

#print "$temp\n";
for ($i=$j+1;$i<@rec_lines;$i++)
{
    @udp_id_compare=split(/\t +/,$rec_lines[$i]);
    for ($k=0;$k<@udp_id;$k++)
    {
        $temp1=0;
        if ($udp_id_compare[$k] eq "cid")
        {
            $temp1=$udp_id_compare[$k+1];
            last;
        }
    }
    for ($k=0;$k<@udp_id;$k++)
    {
        if ($udp_id_compare[$k] eq "seq")
        {
            $temp1=$temp1.($udp_id_compare[$k+1]);
            last;
        }
    }
}

```

```

}
for ($k=0;$k<@udp_id;$k++)
{
    if ($udp_id_compare[$k] eq "ser")
    {
        $stemp1=$stemp1.($udp_id_compare[$k+1]);
        last;
    }
}
#print "$stemp1\n";

if ($stemp eq $stemp1)
{
    print "Found two instances (receiver file) of $stemp1 at line $i
$j\n";
    exit(12);
}

}

#print "$sen_line\n";

}

$elapsed_sec = time - $start;

my $second = $elapsed_sec%60;
my $minute = ($elapsed_sec/60)%60;
my $hour = ($elapsed_sec/(60*60))%24;
    print "Total Time elapsed: $hour hours:$minute min:$second sec\n";

close(BFILE);

```

B.2. SCRIPT CREATEFILES THE PERL

```
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 2 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307,
# USA.

#!/usr/bin/perl -w

open(INPUTFILE, "<udp_sender.txt") || die ("cannot open udp_sender file
1\n");

unless (open (OUTFILE, ">final.tx"))
{
    die ("cannot open output file outfile\n");
}

$start = time;

my (@rec_lines) = <INPUTFILE>;

foreach $rec_line (@rec_lines)
{
    chomp($rec_line);
    @line=split(/\t +/,$rec_line);

    for ($k=0;$k<@line;$k++)
    {
        $id=0;
        if ($line[$k] eq "cid")
        {
            $id=$line[$k+1];
            last;
        }
    }

    for ($k=0;$k<@line;$k++)
    {
        if ($line[$k] eq "seq")
        {
            $id=$id.($line[$k+1]);
        }
    }
}
```

```

        last;
    }
}
for ($k=0;$k<@line;$k++)
{
    if ($line[$k] eq "ser")
    {
        $id=$id.($line[$k+1]);
        last;
    }
}
for ($k=0;$k<@line;$k++)
{
    if ($line[$k] eq "id")
    {
        $id=$id.($line[$k+1]);
        last;
    }
}

for ($k=0;$k<@line;$k++)
{
    if ($line[$k] eq "length:")
    {
        $size=$line[$k+1];
        last;
    }
}

$time=$line[0];
#$id=substr($id,0,length($id)-1);
#$size=substr($size,0,length($size)-1);
#print (" $id $size \n");
print OUTFILE (" $id\t$id\t$time\t$size\n");
}

close(INPUTFILE);
close(OUTPUTFILE);

#second file

open(INPUTFILE, "<udp_receiver.txt") || die ("cannot open udp_receiver
file 1\n");

unless (open (OUTFILE, ">final.rx"))
{

```



```

        die ("cannot open output file outfile\n");
    }

my (@rec_lines) = <INPUTFILE>;

foreach $rec_line (@rec_lines)
{
    chomp($rec_line);
    @line=split(/\t +/,$rec_line);
    for ($k=0;$k<@line;$k++)
    {
        $sid=0;
        if ($line[$k] eq "cid")
        {
            $sid=$line[$k+1];
            last;
        }
    }

    for ($k=0;$k<@line;$k++)
    {
        if ($line[$k] eq "seq")
        {
            $sid=$sid.($line[$k+1]);
            last;
        }
    }

    for ($k=0;$k<@line;$k++)
    {
        if ($line[$k] eq "ser")
        {
            $sid=$sid.($line[$k+1]);
            last;
        }
    }

    for ($k=0;$k<@line;$k++)
    {
        if ($line[$k] eq "id")
        {
            $sid=$sid.($line[$k+1]);
            last;
        }
    }
}

```

```

for ($k=0;$k<@line;$k++)
{
    if ($line[$k] eq "length:")
    {
        $size=$line[$k+1];
        last;
    }
}
$time=$line[0];
#$id=substr($id,0,length($id)-1);
$size=substr($size,0,length($size)-1);

print OUTFILE ("{$id\t$id\t$time\t$size\n"});
}

$elapsed_sec = time - $start;

my $second = $elapsed_sec%60;
my $minute = ($elapsed_sec/60)%60;
my $hour = ($elapsed_sec/(60*60))%24;
    print "Total Time elapsed: $hour hours:$minute min:$second sec\n";

close(INPUTFILE);
close(OUTPUTFILE);

```

B.3. SCRIPT LOSSES THE PERL

```
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 2 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307,
# USA.
#Author irons
#Mail irons@pasiphae.teiher.gr
#This file calculates the losses in a udp transmission.

#!/usr/bin/perl -w

$start = time;

& calc_loss;

$elapsed_sec = time - $start;

my $second = $elapsed_sec%60;
my $minute = ($elapsed_sec/60)%60;
my $hour = ($elapsed_sec/(60*60))%24;
    print "Total Time elapsed: $hour hours:$minute min:$second sec\n";

sub calc_loss
# Simple loss calculation
{
    my ($sender_packets);
    my ($receiver_packets);
    my ($loss_rate);

    $sender_packets = 0;
    $receiver_packets = 0;
    $loss_rate = 0;

# Sender file
    open (SENDER, "<final.tx") || die ("cannot open input file 1\n");

    while (<SENDER>)
    {

        $sender_packets++;

    }
}
```

```

        close (SENDER);

# Receiver file
    open (RECEIVER, "<final.rx") || die ("cannot open input file
2\n");

    while (<RECEIVER>)
    {

        $receiver_packets++;

    }

    close (RECEIVER);

#calculation
    $loss_rate =
        (($sender_packets -
            $receiver_packets) / $sender_packets) * 100;

    print "sender packets $sender_packets, receiver packets
$receiver_packets, losses $loss_rate%\n";

    if (($sender_packets - $receiver_packets)!=0)
    {
        & lossvstime;
    }
}

sub lossvstime
{

unless (open (OUTFILE, ">pack_num_seq_loss_vs_time"))
{

    die ("cannot open output file outfile\n");

}

    open (SENDER, "<final.tx");
open (RECEIVER, "<final.rx");
my (@sen_lines) = <SENDER>;
my ($sen_line);
my (@rec_lines) = <RECEIVER>;
my ($rec_line);
my ($temp);
my ($templ);
my ($lock);
my ($lock_ref_time);
my ($send_time);

```

```

my ($start_ref_time);
my ($packet_counter);
close (SENDER);
close (RECEIVER);
$size=@rec_lines;
$size1=@sen_lines;
$packet_counter=0;
$lock_ref_time=0;

foreach $sen_line (@sen_lines)
{
$lock=0;
$packet_counter++;
    chomp($sen_line);
        @line_sender=split(/\t +/,$sen_line);
        $temp1=$line_sender[0].$line_sender[1];

        if ($lock_ref_time==0)
        {
            $start_ref_time=$line_sender[2];
            $lock_ref_time=1;
            #print ("\n$start_ref_time\n");
        }
foreach $rec_line (@rec_lines)
{
    chomp($rec_line);
        @line_receiver=split(/\t +/,$rec_line);
        $temp=$line_receiver[0].$line_receiver[1];
if ($temp eq $temp1)
    {

        #unless (open (OUTFILE, ">>aligned_sender"))
        #{

            #    die ("cannot open output file outfile\n");

        #}

        #close (OUTFILE);
            $lock=1;
            last;
#print " sender $sen_line receiver $rec_line\n";
        }

    }
if ($lock==0)
{
$send_time=$line_sender[2]-$start_ref_time;
print OUTFILE "$send_time\t$packet_counter \n";

}
#if ($lock==1)
#{

```

```

#$send_time=$line_sender[2]-$start_ref_time;
#print OUTFILE "$send_time\t0 \n";

#}
}
close (OUTFILE);
& avg_lossvstime;
}
sub avg_lossvstime{

unless (open (OUTFILE, ">lossvstime"))
{

    die ("cannot open output file outfile\n");

}

open (LOSSES, "<pack_num_seq_loss_vs_time");

my (@sen_lines) = <LOSSES>;
my ($sen_line);
my ($temp1);
my ($temp);
my ($lock);
my ($lock1);
my ($send_time);
my ($packet_counter);
my ($line_counter);

close (LOSSES);

$size=@rec_lines;
$size1=@sen_lines;
$packet_counter=0;
$lock_ref_time=0;
$line_counter=0;
$lock=0;

$lock1=0;
foreach $sen_line (@sen_lines)
{

$packet_counter++;
    chomp($sen_line);
    @line_sender=split(/\t +/,$sen_line);
    $temp=$line_sender[0];
    if ($lock==0)
    {
        $temp1=$line_sender[0];
#print "$temp1\n";
@line_sender_last=split(/\t +/,$sen_lines[(@sen_lines) -1]);
#print "$line_sender_last[0]\n";
        # $packet_counter=0;
$lock=1;
    }
}

```

```

if (($temp1+1.0) < ($temp))
{
    if ($lock1==0)
    {
        $pack_count=$packet_counter-1;
        print OUTFILE "$send_time\t$pack_count \n";
        #print "$send_time\t$packet_counter\t$temp \n";
        $packet_counter=1;
    }
    $temp1=$temp;
}
}
if (($temp1+1) > ($line_sender_last[0]))
{
    $lock1=1;
    if ($temp==$line_sender_last[0])
    {
        $send_time=$temp;
        # print "$send_time\t$packet_counter\t$temp \n";
        print OUTFILE "$send_time\t$packet_counter \n";
    }
}

}

$send_time=$temp;
$line_counter++;
}
close (OUTFILE);
#print "$pack_count\t$packet_counter\t$line_counter\n";
}

```

B.4. SCRIPT THROUGHPUT THE PERL

```
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 2 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307,
# USA.

#!/usr/bin/perl -w

my($sum_pack_size);
my($lock);
my($start_time);
my($end_time);
my($transfer_time);
my($data_rate);
$data_rate=0;
$transfer_time=0;
$sum_pack_size=0;
$lock=0;
$start_time=0;
$end_time=0;

open(SENDER, "<final.tx") || die ("cannot open input file 1\n");

$start = time;

while (<SENDER>)
{
    my($sen_line) = $_;
    chomp($sen_line);

    @line_sender=split(/\t +/, $sen_line);
    $sum_pack_size+=$line_sender[3];
    if ($lock==0)
    {
        $start_time=$line_sender[2];
        $lock=1;
    }

    #print "$papa[3]\n";

}
$end_time=$line_sender[2];
#print "stime $start_time etime $end_time\n";
$transfer_time=$end_time-$start_time;
```



```

$data_rate=$sum_pack_size/$stransfer_time;
print "SENDER RESULTS\n";
print "total bytes transferred $sum_pack_size in $stransfer_time sec.
Sender output data rate is $data_rate bytes/sec \n";
close(SENDER);

open(RECEIVER, "<final.rx") || die ("cannot open input file 2\n");
$data_rate=0;
$stransfer_time=0;
$sum_pack_size=0;
$lock=0;
$start_time=0;
$end_time=0;
while (<RECEIVER>)
{
    my($sen_line) = $_;
    chomp($sen_line);

    @line_receiver=split(/\t +/,$sen_line);
    $sum_pack_size+=$line_receiver[3];
    if ($lock==0)
    {
        $start_time=$line_receiver[2];
        $lock=1;
    }

    #print "$papa[3]\n";

}
$end_time=$line_receiver[2];
print "stime $start_time etime $end_time\n";
$stransfer_time=$end_time-$start_time;
$data_rate=$sum_pack_size/$stransfer_time;
print "RECEIVER RESULTS\n";
print "total bytes transferred $sum_pack_size in $stransfer_time sec.
Receiver input data rate is $data_rate bytes/sec \n";

close(RECEIVER);

$elapsed_sec = time - $start;

my $second = $elapsed_sec%60;
my $minute = ($elapsed_sec/60)%60;
my $hour = ($elapsed_sec/(60*60))%24;
    print "Total Time elapsed: $hour hours:$minute min:$second sec\n";

```

B.5. SCRIPT JITTER THE PERL

```
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 2 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307,
# USA.

#!/usr/bin/perl -w

unless (open (OUTFILE, ">aligned_sender"))
{
    die ("cannot open output file outfile\n");
}

open (SENDER, "<final.tx");
open (RECEIVER, "<final.rx");
my (@sen_lines) = <SENDER>;
my ($sen_line);
my (@rec_lines) = <RECEIVER>;
my ($rec_line);
my ($temp);
my ($temp1);

close (SENDER);
close (RECEIVER);
$size=@rec_lines;
$size1=@sen_lines;
print "Receiver packets $size --- Sender packets $size1\n";

$start = time;

foreach $rec_line (@rec_lines)
{
    chomp($rec_line);
    @line_receiver=split(/\t +/,$rec_line);
    $temp=$line_receiver[0].$line_receiver[1];
    foreach $sen_line (@sen_lines)
    {
        chomp($sen_line);
        @line_sender=split(/\t +/,$sen_line);
        $temp1=$line_sender[0].$line_sender[1];
```

```

if ($temp eq $temp1)
{
    #unless (open (OUTFILE, ">>aligned_sender"))
    #{
        #    die ("cannot open output file outfile\n");
    #}

    print OUTFILE "$sen_line\n";

    #close (OUTFILE);

        last;
#print " sender $sen_line receiver $rec_line\n";
    }

}

$elapsed_sec = time - $start;

my $second = $elapsed_sec%60;
my $minute = ($elapsed_sec/60)%60;
my $hour = ($elapsed_sec/(60*60))%24;
    print "Total Time elapsed: $hour hours:$minute min:$second sec\n";

close (OUTFILE);

```

B.6. SCRIPT TIMESTAMP THE PERL

```
#!/usr/bin/perl -w

unless (open (SENDER, "<aligned_sender"))
{
    die ("cannot open input file outfile\n");
}

unless (open (OUTFILE, ">sender_timestamp"))
{
    die ("cannot open output file outfile\n");
}

$start = time;

while (<SENDER>)
{
    my($sen_line) = $_;
    chomp($sen_line);

    @line_sender=split(/\t +/,$sen_line);

    print OUTFILE "$line_sender[2]\n";

    #print "$papa[3]\n";
}

close(OUTFILE);
close(SENDER);

unless (open (RECEIVER, "<final.rx"))
{
    die ("cannot open input file outfile\n");
}

unless (open (OUTFILE, ">receiver_timestamp"))
{
    die ("cannot open output file outfile\n");
}
```

```
while (<RECEIVER>)
{
  my($sen_line) = $_;
  chomp($sen_line);

  @line_receiver=split(/\t +/,$sen_line);

  print OUTFILE "$line_receiver[2]\n";

  #print "$papa[3]\n";

}

close(OUTFILE);

close(RECEIVER);

$elapsed_sec = time - $start;

my $second = $elapsed_sec%60;
my $minute = ($elapsed_sec/60)%60;
my $hour = ($elapsed_sec/(60*60))%24;
  print "Total Time elapsed: $hour hours:$minute min:$second sec\n";
```

B.7. SCRIPT TIMESTAMP ONE WAY DELAY THE PERL

```
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 2 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307,
# USA.

#!/usr/bin/perl -w

unless (open (SENDER, "<sender_timestamp"))
{
    die ("cannot open input file outfile\n");
}
unless (open (RECEIVER, "<receiver_timestamp"))
{
    die ("cannot open input file outfile\n");
}

my (@sen_lines) = <SENDER>;
my (@rec_lines) = <RECEIVER>;
my($delay);
my($counter);
my($avg_delay);
my($sample_time);
my($min);
my($max);
$counter=0;
$delay=0;
$avg_delay=0;
$sample_time=0;
$max=-1;
$min=1000000;
close(SENDER);
close(RECEIVER);

unless (open (OUTFILE, ">one_way_delayvstime"))
{
    die ("cannot open output file jittervstime\n");
}
```

```

$start = time;

# One way delay formula is  $D_i = \text{abs}(R_i - S_i)$ 
#Avg One way Delay is  $\text{Sum}(D_i)/n$ 

for ($i=0;$i<@sen_lines;$i++)
{
    #print("Sender line $sen_lines[$i]\n");
    #print("Receiver line $rec_lines[$i]\n");
    $delay=abs(($rec_lines[$i])-( $sen_lines[$i]))*1000;
    $avg_delay+=$delay;
    if ($min>$delay)
    {
        $min=$delay;
    }
    if ($max<$delay)
    {
        $max=$delay;
    }
    $sample_time=$rec_lines[$i]-$rec_lines[0];
    #print (" $sample_time $jitter\n");
    print OUTFILE "$sample_time $delay\n";
    $counter++;
}
$avg_delay=($avg_delay/$counter);
print ("Average One way Delay is $avg_delay ms. Max One way Delay is
$max ms. Min One way Delay is $min ms\n");

close(OUTFILE);

$elapsed_sec = time - $start;

my $second = $elapsed_sec%60;
my $minute = ($elapsed_sec/60)%60;
my $hour = ($elapsed_sec/(60*60))%24;
    print "Total Time elapsed: $hour hours:$minute min:$second sec\n";

```

B.8. SCRIPT SMOOTHED JITTER (RFC 3550) THE PERL

```
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 2 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307,
# USA.

#!/usr/bin/perl -w

unless (open (SENDER, "<sender_timestamp"))
{
    die ("cannot open input file outfile\n");
}
unless (open (RECEIVER, "<receiver_timestamp"))
{
    die ("cannot open input file outfile\n");
}

my (@sen_lines) = <SENDER>;
my ($sen_line);
my (@rec_lines) = <RECEIVER>;
my ($rec_line);
my($stransit);
my($delta_transit);
my($last_transit);
my($jitter);
my($counter);
$counter=0;
$stransit=0;
$delta_transit=0;
$last_transit=0;
$jitter=0;

close(SENDER);
close(RECEIVER);

unless (open (OUTFILE, ">final_jitter"))
{
    die ("cannot open output file outfile\n");
}
```



```

unless (open (OUTFILE1, ">final_pack2packdelay"))
{
    die ("cannot open output file outfile\n");
}

$start = time;

foreach $sen_line (@sen_lines)
{
    chomp($sen_line);

    $transit=$rec_lines[$counter]-$sen_line;
    if ($last_transit!=0)
    {
        $delta_transit=$transit-$last_transit;
        if ( $delta_transit < 0 ) {
            $delta_transit = -$delta_transit;
        }

        $jitter+=$(delta_transit-$jitter)/16.0;

    }
    $last_transit=$transit;
    $result=$jitter*1000;
    $timerec=$rec_lines[$counter];
    chomp($timerec);
    print OUTFILE "$timerec $result\n";
    $pack_delay=$delta_transit*1000;
    print OUTFILE1 "$timerec $pack_delay\n";
    #print "$sen_line $rec_lines[$counter] $result\n";
    $counter++;
}

close(OUTFILE);
close(OUTFILE1);

unless (open (INFILE, "<final_jitter"))
{
    die ("cannot open input file outfile\n");
}

$min=100000;
$max=0;
$counter=0;
$result=0;
$lock=0;
while (<INFILE>)
{

```

```

my($sen_line) = $_;
  chomp($sen_line);
@values=split(/\t +/,$sen_line);
$value=$values[1];
$result+=$value;
  if ($counter==1)
  {
$min=$value;

}

if ($value>$max)
{
  $max=$value;

}
if (($value<$min))
{
  $min=$value;

}
$counter++;

}

close (INFILE);
$result=$result/$counter;

print "aver jitter is $result max is $max min $min\n";

unless (open (INFILE, "<final_pack2packdelay"))
{

    die ("cannot open input file outfile\n");

}
$min=100000;
$max=0;
$counter=0;
$result=0;
$lock=0;
while (<INFILE>)
{

my($sen_line) = $_;
  chomp($sen_line);

@values=split(/\t +/,$sen_line);
$value=$values[1];
$result+=$value;
  if ($counter==1)
  {
$min=$value;

}

```

```

if ($value>$max)
{
    $max=$value;
}
if (($value<$min))
{
    $min=$value;
}
$counter++;

}

close (INFILE);
$result=$result/($counter);

print "aver pack2packdelay is $result max is $max min $min\n";

unless (open (INFILE, "<final_pack2packdelay"))
{
    die ("cannot open input file outfile\n");
}

unless (open (OUTFILE, ">timed_final_pack2packdelay"))
{
    die ("cannot open input file outfile\n");
}

my (@times) = <INFILE>;
close(INFILE);
$time=0;
$lock=0;
for ($i=0;$i<@times-1;$i++)
{
    if ($lock==0)
    {
        @valuesplits=split(/\t +/,$times[$i]);
        $valuesplit=$valuesplits[1];
chomp($valuesplit);
        print OUTFILE "$time $valuesplit\n";
$lock=1;
    }
    #chomp($sen_line);
    @timesplit=split(/\t +/,$times[$i+1]);
    $temp_timel=$timesplit[0];
    @timesplit=split(/\t +/,$times[$i]);

```

```

$temp_time2=$timesplit[0];
$time=($temp_time1-$temp_time2)+$time;
#$time=$timesplit[0];
@valuesplits=split(/\t +/,$times[$i+1]);
$valuesplit=$valuesplits[1];
chomp($valuesplit);
print OUTFILE "$time $valuesplit\n";
#print "$time\n";
}

close(OUTFILE);

unless (open (INFILE, "<final_jitter"))
{
    die ("cannot open input file outfile\n");
}

unless (open (OUTFILE, ">timed_final_jitter"))
{
    die ("cannot open input file outfile\n");
}

my (@times) = <INFILE>;
close(INFILE);
$time=0;
$lock=0;
for ($i=0;$i<@times-1;$i++)
{
    if ($lock==0)
    {
        @valuesplits=split(/\t +/,$times[$i]);
        $valuesplit=$valuesplits[1];
chomp($valuesplit);
        print OUTFILE "$time $valuesplit\n";
$lock=1;
    }
    #chomp($sen_line);
    @timesplit=split(/\t +/,$times[$i+1]);
    $temp_time1=$timesplit[0];
    @timesplit=split(/\t +/,$times[$i]);
    $temp_time2=$timesplit[0];
    $time=($temp_time1-$temp_time2)+$time;
    #$time=$timesplit[0];
    @valuesplits=split(/\t +/,$times[$i+1]);
    $valuesplit=$valuesplits[1];
chomp($valuesplit);
    print OUTFILE "$time $valuesplit\n";
    #print "$time\n";
}
}

```

```
close(OUTFILE);

$elapsed_sec = time - $start;

my $second = $elapsed_sec%60;
my $minute = ($elapsed_sec/60)%60;
my $hour = ($elapsed_sec/(60*60))%24;
    print "Total Time elapsed: $hour hours:$minute min:$second sec\n";
```

B.9. SCRIPT JITTER THE PERL

```
# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 2 of the License, or
# (at your option) any later version.
#
# This program is distributed in the hope that it will be useful,
# but WITHOUT ANY WARRANTY; without even the implied warranty of
# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
# GNU General Public License for more details.
#
# You should have received a copy of the GNU General Public License
# along with this program; if not, write to the Free Software
# Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307,
# USA.

#!/usr/bin/perl -w

unless (open (SENDER, "<sender_timestamp"))
{
    die ("cannot open input file outfile\n");
}
unless (open (RECEIVER, "<receiver_timestamp"))
{
    die ("cannot open input file outfile\n");
}

my (@sen_lines) = <SENDER>;
my (@rec_lines) = <RECEIVER>;
my($jitter);
my($counter);
my($avg_jitter);
my($sample_time);
my($min);
my($max);
$counter=0;
$jitter=0;
$avg_jitter=0;
$sample_time=0;
$max=-1;
$min=1000000;
close(SENDER);
close(RECEIVER);

unless (open (OUTFILE, ">jittervstime"))
{
    die ("cannot open output file jittervstime\n");
}
```

```

$start = time;

# Jitter formula is  $D_i = \text{abs}(R_i - R_{i-1}) - (S_i - S_{i-1})$ 
#Avg jitter is  $\text{Sum}(D_i)/n$ 

for ($i=0;$i<@sen_lines-1;$i++)
{
    #print("Sender line $sen_lines[$i]\n");
    #print("Receiver line $rec_lines[$i]\n");
    $jitter=abs(($rec_lines[$i+1]-$rec_lines[$i])-( $sen_lines[$i+1]-
$sen_lines[$i]))*1000;
    $avg_jitter+=$jitter;
    if ($min>$jitter)
    {
        $min=$jitter;
    }
    if ($max<$jitter)
    {
        $max=$jitter;
    }
    $sample_time=$rec_lines[$i+1]-$rec_lines[0];
    #print (" $sample_time $jitter\n");
    print OUTFILE " $sample_time $jitter\n";
    $counter++;
}
$avg_jitter=( $avg_jitter/$counter);
print ("Average Jitter is $avg_jitter ms. Max jitter is $max ms. Min
jitter is $min ms\n");

close(OUTFILE);

$elapsed_sec = time - $start;

my $second = $elapsed_sec%60;
my $minute = ($elapsed_sec/60)%60;
my $hour = ($elapsed_sec/(60*60))%24;
    print "Total Time elapsed: $hour hours:$minute min:$second sec\n";

```

Βιβλιογραφία

- [1] ISO/IEC 13818-2, “Generic Coding of Moving Pictures and Associated Audio Information (MPEG-2) Part 2: Video”, ISO, 1996.
- [2] ISO/IEC 13818-3, “Generic Coding of Moving Pictures and Associated Audio Information (MPEG-2) Part 3: Audio”, ISO, 1996.
- [3] ISO/IEC 13818-1, “Generic Coding of Moving Pictures and Associated Audio Information (MPEG-2) Part 1: Systems”, ISO, 1996.
- [4] P. A. Sarginson, “MPEG-2: Overview of the systems layer”, BBC Research and Development Report, February 2006.
- [5] ETSI EN 300 744, “Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital terrestrial television”, v1.5.1, ETSI, 2004.
- [6] ETS 300 744, “Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital Terrestrial television (DVB-T)”, ETSI, 1997.
- [7] Digital Video Broadcasting (DVB) Forum, February 2007, <http://www.dvb.org>
- [8] European Telecommunications Standards Institute (ETSI), February 2007, <http://www.etsi.org>
- [9] Digital Video Broadcasting (DVB) Worldwide, February 2007, http://www.dvb.org/about_dvb/dvb_worldwide
- [10] ATSC A53E, “ATSC Digital Television Standard (A53) Revision E, with Amendments No.1 and 2”, ATSC, 2005.
- [11] Y. Wu, E. Pliszka, B. Caron, P. Bouchard, G. Chouinard, “Comparison of terrestrial DTV transmission systems: the ATSC 8-VSB, the DVB-T COFDM, and the ISDB-T BST-OFDM”, IEEE Transactions on Broadcasting, vol. 46, no. 2, June 2000, pp. 101-113.
- [12] W. Y. Zou, W. Yiyan, “COFDM: an overview”, IEEE Transactions on Broadcasting, vol. 41, no. 1, March 1995, pp. 1-8.

- [13] J. Chuang, N. Sollenberger, “Beyond 3G: Wideband Wireless Data Access Based on OFDM and Dynamic Packet Assignment”, IEEE Communications Magazine, vol. 38, no. 7, July 2000, pp 78-87.
- [14] J. Stott, “The how and why of COFDM”, EBU Technical Review, Winter 1998.
- [15] U. Reimers, “Digital Video Broadcasting”, IEEE Communications Magazine, vol. 36, no. 6, June 1998, pp.104-110.
- [16] ETSI EN 300 421, “Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for 11/12 GHz satellite services”, v1.1.2, ETSI, 1997.
- [17] ETSI EN 300 429, “Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for cable systems”, v1.2.1, ETSI, 1998.
- [18] E.M.Πάλλης, “Towards the Provision of High Quality Interactive Multimedia Services via a Wireless Broadband Networking Infrastructure” PhD thesis, 2002
- [19] A.S.Tanenbaum, “Δίκτυα Υπολογιστών”, Τρίτη έκδοση
- [20] http://en.wikipedia.org/wiki/Transmission_Control_Protocol
- [21] Γ.Γαρδικής, “Παροχή υπηρεσιών TCP/IP σε συστήματα επίγειας ψηφιακής τηλεόρασης (DVB-T)”, PhD thesis, 2004
- [22] http://en.wikipedia.org/wiki/User_Datagram_Protocol