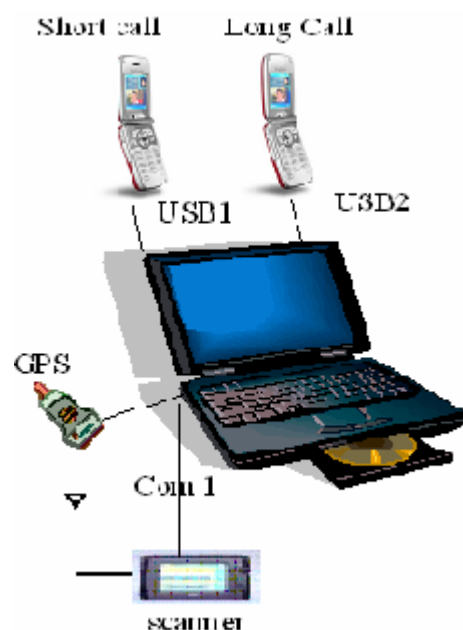




TECHNOLOGICAL EDUCATIONAL
INSTITUTE OF CRETE
BRAUCH OF CHANIA
DEPT OF ELECTRONICS

Degree Project

Analysis of radio coverage measurements from third generation 3G mobile communications systems.



Students:

Fotinopoulou Aimilia

Hohlidaki Theodosia

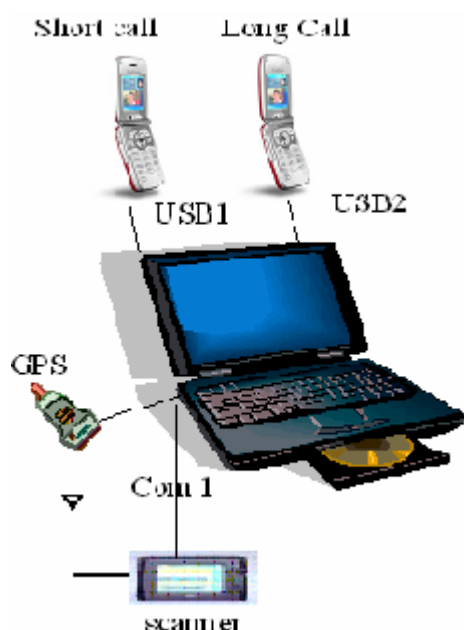
Chania November 2005



ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ
ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΗΣ
ΤΟΜΕΑΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

Πτυχιακή εργασία

Θέμα: Ανάλυση Μετρήσεων Ραδιοκάλυψης Κινητών
Επικοινωνιών Τρίτης Γενιάς



Σπουδάστριες:
Φωτεινοπούλου Αιμιλία
Χοχλιδάκη Θεοδοσία

Επιβλέπων Καθηγητής:
Επίκουρος Καθηγητής Κος Ευάγγελος Κόκκινος

Χανιά Νοέμβριος 2005

ΠΕΡΙΕΧΟΜΕΝΑ

<i>Πρόλογος</i>	<i>σελ. 2</i>
<i>Λίγα λόγια για τη κινητή τηλεφωνία τρίτης γενιάς (3G)</i>	<i>σελ. 4</i>
<i>Μετρήσεις ραδιοκάλυψης</i>	<i>σελ. 6</i>
<i>Ανάλυση γραμμής ECN0</i>	<i>σελ. 6</i>
<i>VISUAL BASIC 6</i>	<i>σελ. 10</i>
<i>Εκκίνηση της Visual Basic 6</i>	<i>σελ. 10</i>
<i>Σχέση Visual Basic και Access και τρόποι σύνδεση τους</i>	<i>σελ. 15</i>
<i>Κώδικας σε Visual Basic</i>	<i>σελ. 31</i>
<i>MATLAB 7</i>	<i>σελ. 56</i>
<i>Βήματα πριν την δημιουργία γραφικών</i>	<i>σελ.56</i>
<i>Σημαντικοί ορισμοί</i>	<i>σελ.59</i>
<i>Δημιουργία γραφικών στο Matlab</i>	<i>σελ.62</i>
<i>Βιβλιογραφία</i>	<i>σελ.91</i>

Πρόλογος



Τα συστήματα τρίτης γενιάς (3G) σχεδιάζονται για την επικοινωνία με πολυμέσα:

- α. Με την άτομο προς άτομο επικοινωνία που μπορεί να ενισχυθεί με τις υψηλής ποιότητας εικόνες και βίντεο.
- β. Με τη πρόσβαση στις υπηρεσίες πληροφοριών στα δημόσια και ιδιωτικά δίκτυα που θα ενισχυθεί από τις υψηλότερες ταχύτητες δεδομένων και τις νέες εύκαμπτες ικανότητες επικοινωνίας των συστημάτων τρίτης γενιάς.

Αυτό μαζί με τη συνεχιζόμενη εξέλιξη των συστημάτων δεύτερης γενιάς (2G) θα δημιουργήσει τις νέες επιχειρησιακές ευκαιρίες όχι μόνο για τους κατασκευαστές και τους χειριστές αλλά και για τους προμηθευτές του περιεχομένου και των εφαρμογών που χρησιμοποιούν αυτά τα δίκτυα.

Κύριος σκοπός της πτυχιακής αυτής είναι η ανάλυση μετρήσεων ραδιοκάλυψης κινητών επικοινωνιών τρίτης γενιάς 3G. Στο βιβλίο αυτό γίνεται ανάλυση των μετρήσεων που δόθηκαν από το δίκτυο της COSMOTE χρησιμοποιώντας το μετρητικό σύστημα TOM. Η συγκεκριμένη ανάλυση γίνεται με τέτοιο τρόπο ώστε να παρέχονται οι απαραίτητες πληροφορίες για την κάλυψη σήματος σε ένα συγκεκριμένο σημείο κάθε χρονική στιγμή.

Το βιβλίο αυτό περιλαμβάνει τα ακόλουθα:

-  Στο πρώτο κεφάλαιο γίνεται η ανάλυση του κώδικα που φτιάχτηκε σε VISUAL BASIC, τρόποι σύνδεσης VISUAL BASIC με ACCESS (βάση δεδομένων) και τρέξιμο του προγράμματος.
-  Στο δεύτερο κεφάλαιο γίνεται η σύνδεση της ACCESS με το MATLAB 7 έχοντας ως αποτέλεσμα τη δημιουργία γραφικών.

Φωτεινοπούλου Αιμιλία
Χοχλιδάκη Θεοδοσία

Preface

Third generation systems (3G) are designed for multimedia communication:

- a. The person-to-person communication that can be enhanced with high quality images and video.
- b. The access to information services on public and private networks that will be enhanced by the higher data rates and new flexible communication capabilities of third generation systems.

These facts are reinforced by the continuing evolution of the second generation systems (2G) which will create new business opportunities not only for manufacturers and operators but also for the providers of content and applications using these networks.

The main objective of this project is the analysis of radio coverage measurements from third generation (3G) mobile communications systems. The measurements were given by a Greek operator (COSMOTE) and were produced using the TOM measurement system. This specific analysis is carried out in such a way that the essential information concerning radio coverage in a specific point is provided each time.

This book includes two chapters.

- ✚ In the first chapter there is an analysis of the code that was made in VISUAL BASIC, the way that VISUAL BASIC is connected with ACCESS (data base) and how the program runs.
- ✚ In the second chapter is described the way that the data are transferred from ACCESS to MATLAB 7, in order to create the graphs.

Fotinopoulou Aimilia
Hohlidaki Theodosia

ΛΙΓΑ ΛΟΓΙΑ ΓΙΑ ΤΗ ΚΙΝΗΤΗ ΤΗΛΕΦΩΝΙΑ ΤΡΙΤΗΣ ΓΕΝΙΑΣ3G

Η κινητή τηλεφωνία έχει διεισδύσει για τα καλά στη ζωή μας, καθιστώντας το κινητό τηλέφωνο απαραίτητο εργαλείο για όλους μας. Νέες υπηρεσίες δημιουργούνται συνεχώς από τα δίκτυα κινητής τηλεφωνίας για να καλύψουν όλο και περισσότερες ανάγκες και να παρέχουν νέες ευκολίες στους χρήστες, ενώ νέες τεχνολογίες ξεπροβάλλουν διαρκώς για να διευκολύνουν τη δημιουργία νέων υπηρεσιών. Μια από αυτές τις τεχνολογίες είναι το 3G για το οποίο σίγουρα όλοι έχουμε ακούσει. Τι είναι όμως... αυτό και τι μπορεί να μας προσφέρει;

Το 3G είναι τα αρχικά των λέξεων 3rd Generation και αποτελεί ένα γενικό όρο ο οποίος αναφέρεται στην τρίτη γενιά τεχνολογίας κινητής τηλεφωνίας. Ως γενιά χαρακτηρίζεται το σύνολο των ασύρματων τεχνολογιών που επιτρέπουν τη μετάδοση φωνής ή και δεδομένων στα δίκτυα κινητής τηλεφωνίας. Μεταξύ των τεχνολογιών αυτών είναι οι WCDMA, CDMA2000 (Το CDMA2000, γνωστό και ως IMT-CDMA Multi Carrier (1X/3X) είναι η τεχνολογία τρίτης γενιάς (3G), που προέκυψε από την σταδιακή εξέλιξη του CDMAone. Η συγκεκριμένη τεχνολογία παρέχει στους παρόχους υπηρεσιών κινητής τηλεφωνίας δεύτερης γενιάς (2G) να αναβαθμίσουν χωρίς προβλήματα τα δίκτυά τους σε δύο φάσεις. Οι δυνατότητες της πρώτης φάσης περιγράφονται με το πρότυπο 1X, το οποίο επιτρέπει τη μετάδοση δεδομένων σε ταχύτητες που φθάνουν τα 144 kbps. Η δεύτερη φάση του CDMA2000 ονομάζεται 3X και επιτρέπει τη μετάδοση δεδομένων σε ταχύτητες που φθάνουν τα 2 Mbps, αλλά και την παροχή προηγμένων multimedia υπηρεσιών), UMTS και EDGE (Το EDGE είναι η τεχνολογία, που επιτρέπει στα υπάρχοντα δίκτυα κινητής τηλεφωνίας δεύτερης γενιάς (2G) την απαιτούμενη «χωρητικότητα» και ταχύτητα, ώστε να επιτραπεί η χρήση τους για την παροχή υπηρεσιών, που χαρακτηρίζουν τα δίκτυα τρίτης γενιάς (3G)! Με τη χρήση του EDGE μπορούν να εξυπηρετηθούν τρεις φορές περισσότεροι συνδρομητές από ότι με το GPRS, να τριπλασιαστεί η ταχύτητα μεταφοράς δεδομένων ή απλά να απελευθερωθεί «χώρος» για την βελτίωση της ποιότητας των φωνητικών υπηρεσιών. Έτσι θεωρητικά γίνονται εφικτές υπηρεσίες όπως τηλεσυνδιάσκεψη, αναπαραγωγή video streaming, και πραγματικό Internet browsing με ιδιαίτερα υψηλές ταχύτητες. Το EDGE ουσιαστικά αποτελεί μια αναβάθμιση του GPRS και δεν μπορεί να λειτουργήσει αυτόνομα σε κάποιο δίκτυο GSM.). Το WCDMA (Wideband Code Multiple Division Access), έχει ήδη επιλεγεί ως το "σύστημα" τρίτης γενιάς, που θα χρησιμοποιηθεί στην Ευρώπη, Ιαπωνία και στις ΗΠΑ. Το WCDMA είναι η τεχνολογία τρίτης γενιάς (3G) που επιτρέπει την υψηλής ταχύτητας και ποιότητας μετάδοση ψηφιακών δεδομένων. Το W-CDMA βασίζεται στο CDMA, μόνο που σε αντίθεση με αυτό, χρησιμοποιεί περισσότερες από μια συχνότητες για τη διαχείριση της συνολικής κίνησης. Μολονότι απαιτείται μεγαλύτερο εύρος συχνοτήτων για τη λειτουργία του, καθιστά εφικτή, με τις τεχνικές που χρησιμοποιεί, τη μετάδοση πληροφοριών σε ταχύτητες που φθάνουν θεωρητικά τα 2Mbps.

Στο σημείο αυτό ας αναφέρουμε δυο λόγια για τη πρώτη και τη δεύτερη γενιά κινητής τηλεφωνίας. Ως πρώτη γενιά (1G) χαρακτηρίστηκε η αναλογική τεχνολογία κινητής τηλεφωνίας, ενώ ως δεύτερη γενιά (2G) η ψηφιακή τεχνολογία που ακολούθησε. Τεχνολογίες όπως το GSM, TDMA και CDMA αποτελούν τρέχουσες τεχνολογίες μετάδοσης δεδομένων στα δίκτυα δεύτερης γενιάς. Υπάρχουν βέβαια και τεχνολογίες που προσφέρουν υψηλότερες ταχύτητες στα δίκτυα δεύτερης γενιάς όπως αυτή του GPRS που αποκαλείται 2.75G αλλά και η πολλά υποσχόμενη τεχνολογία **EDGE** που έχει ήδη ονομαστεί χαρακτηριστικά 2.75G, υποδηλώνοντας μια ενδιάμεση μεταβατική γενιά πριν το 3G.

Ας δούμε τώρα πως λειτουργεί το 3G: Σε αντίθεση με προηγούμενες τεχνολογίες που βασίζονται σε μετάδοση δεδομένων πάνω από απευθείας σύνδεση των δύο μερών (circuit-switched), οι τεχνολογίες που ανήκουν στην ομάδα τρίτης γενιάς βασίζονται σε υψηλής ταχύτητας μετάδοση δεδομένων μοιρασμένων σε πακέτα (packet-switched). Η τεχνολογία στηρίζεται στα γνωστά δίκτυα GSM με μια παραλλαγή του CDMA με το όνομα WCDMA (Wideband-CDMA) η οποία είναι ικανή να επιτύχει ταχύτητες μετάδοσης έως και 2Mbps! Στα άμεσα επερχόμενα δίκτυα τρίτης γενιάς όμως, ο συνδυασμός του WCDMA με τις υπό διάθεση συσκευές θα είναι ικανός να προσφέρει στον τελικό χρήστη ταχύτητες έως και 384Kbps, οι οποίες όμως είναι αρκετές για να μετατρέψουν το κινητό σε μια ασύρματη συσκευή πολυμέσων.

Οι σημαντικότερες υπηρεσίες του 3G:

- Οι βίντεο-κλήσεις είναι χωρίς αμφιβολία μια από τις πιο πολυσυζητημένες υπηρεσίες των δικτύων 3G. Πλέον, εκτός από το να ακούμε το συνομιλητή μας θα μπορούμε και να τον... απολαμβάνουμε ζωντανά στην οθόνη του κινητού μας. Φυσικά, θα πρέπει να έχουμε τόσο εμείς, όσο και ο συνομιλητής μας κάποια συμβατή συσκευή.
- Οι υψηλές ταχύτητες ασύρματης μεταφοράς δεδομένων είναι ένα ακόμη από τα πλεονεκτήματα των δικτύων 3G. Η σύνδεση στο Internet εκτός από άμεση και απρόσκοπτη, θα μας δώσει πλέον και ταχύτητες που φθάνουν τα 384kbps - ανάλογες δηλαδή με αυτές της σταθερής τεχνολογίας xDSL.
- Οι υψηλές ταχύτητες μεταφοράς δεδομένων βοηθούν αρκετά στην πιο γρήγορη και άμεση χρήση διαφόρων multimedia εφαρμογών. Έτσι, αν επιχειρήσουμε να στείλουμε ένα MMS σε κάποιον άλλο συνδρομητή, τότε δεν θα χρειαστούμε να περιμένουμε περισσότερο από 10 δευτερόλεπτα - όταν στα δίκτυα 2G ο χρόνος αυτός μπορούσε να ξεπεράσει και το 1 λεπτό.
- Το video-streaming είναι μια ακόμη από τις υπηρεσίες που παρέχουν τα δίκτυα 3G. Το αυξημένο bandwidth επιτρέπει τη μετάδοση σε πραγματικό χρόνο, κινούμενης εικόνας και ήχου υψηλής ανάλυσης. Έτσι, θα μπορούμε να παρακολουθούμε τηλεοπτικά προγράμματα, ζωντανά ή μαγνητοσκοπημένα, ανεξαρτήτως τόπου και χρόνου.
- Υψηλής ποιότητας παιχνίδια, τα οποία θα μπορούν να παίζονται online σε πραγματικό χρόνο και ταυτόχρονα με άλλους παίκτες.
- Υπηρεσίες εύρεσης θέσεως, σε συνδυασμό με την τεχνολογία GPS, οι οποίες θα μπορούν να παρέχουν χάρτες της περιοχής που βρισκόμαστε, εύρεση βέλτιστης διαδρομής προς τον προορισμό μας, γειτονικά σημεία ενδιαφέροντος κλπ.

Μετά την ευρεία διείσδυση της τεχνολογίας 3G αναμένεται να διατεθούν ακόμη περισσότερες υπηρεσίες, όπως μετάδοση τηλεοπτικών εκπομπών και υπηρεσίες παγκόσμιας περιαγωγής.

ΜΕΤΡΗΣΕΙΣ ΡΑΔΙΟΚΑΛΥΨΗΣ

Τα μετρητικά συστήματα είναι ένα από τα βασικά input που έχουν οι operator για να βελτιώσουν την κάλυψη τους.

Στην πτυχιακή μας πήραμε μετρήσεις από το μετρητικό σύστημα TOM το οποίο είναι το μόνο που έχει ανοιχτά τα αρχεία του. Ας δούμε τις κύριες παραμέτρους που αναλύσαμε.

- **RSCP-Received Signal Code Power** (Λαμβανόμενη Ισχύς Κωδικού Σήματος). Χρησιμοποιείται κυρίως για να υπολογίσει την απώλεια διαδρομής (path loss) από τη στιγμή που η ισχύς εκπομπής του CPICH (κοινού δοκιμαστικού καναλιού) είναι γνωστή ή μπορεί να διαβιβαστεί από το σύστημα πληροφοριών.
- **RSSI-Received Signal Strength Indicator** η λαμβανόμενη ευρυζωνική ισχύς στο B.W του καναλιού στο downlink.
- **Ec/No** είναι η πιο σημαντική μέτρηση στο WCDMA για τους σκοπούς σχεδίασης δικτύου λόγω του ότι είναι ο βασικός δείκτης κάλυψης αλλά επίσης έχει καλή ακρίβεια αφού η μέτρηση μπορεί να γίνει στη βασική ζώνη. Μετρείται μετά την αποδιαμόρφωση και μας δείχνει πόσο ισχυρός είναι ο κώδικας σε σχέση με την ισχύ των 5MHz.

Spreading(διεύρυνση φάσματος)

Η έννοια του **Spreading** εφαρμόζεται στα φυσικά κανάλια και αποτελείται από 2 λειτουργίες . Η πρώτη είναι η **Channelisation** η οποία μετατρέπει κάθε σύμβολο δεδομένων(bits) σε ένα αριθμό από chips, έτσι αυξάνει το εύρος ζώνης του σήματος. Ο αριθμός των chips ανά σύμβολο δεδομένων ονομάζεται SF(Spreading factor). Η δεύτερη λειτουργία είναι η **Scrambling** η οποία εφαρμόζεται πάνω στο διευρυμένο σήμα και δεν επηρεάζει το εύρος ζώνης εκπομπής.

- **Scrambling Codes** : χρησιμοποιείται για να ξεχωρίσει τις διαφορετικές κυψέλες στην downlink και τα διαφορετικά UE(user equipment=κινητά) στην uplink κατεύθυνση. Στη downlink : σε κάθε κυψέλη(cell) εκχωρείται ένας και μόνο ένας πρωτεύον Scrambling Code (primary code). Υπάρχουν 512 primary codes που χωρίζονται σε 64 ομάδες των 8 Scrambling Code σε κάθε ομάδα. Στην uplink : υπάρχουν μερικά εκατομμύρια Scrambling Code και κάθε κινητό χρησιμοποιεί ένα Scrambling Code που του παρέχει το δίκτυο.

ΑΝΑΛΥΣΗ ΓΡΑΜΜΗΣ ECN0 ΣΥΜΦΩΝΑ ΜΕ ΤΟ ΠΑΡΑΚΑΤΩ ΠΙΝΑΚΑ ΤΟΥ ΜΕΤΡΗΤΙΚΟΥ ΣΥΣΤΗΜΑΤΟΣ TOM

4.12.4 EC/NO (ECN0)

Event ID	ECN0
Cellular systems	WCDMA
Mode	All
Recorded	Always Recorded when measurement sample is received from mobile
Nemo Tools	TOM

Parameters

Name	Type	Description
Cellular system	Integer	12 = WCDMA FDD 2100 13 = WCDMA TDD 2100 20 = WCDMA FDD 1900 21 = WCDMA TDD 1900
Number of channels	Integer	Number of carriers.
Channel number	Integer	Carrier channel number
Carrier RSSI	Integer	Carrier RSSI.
Number of parameters per cell	Integer	Number of parameters for each cell.
Cell count active	Integer	Number of cells in the active set
Channel	Integer	Channel number.
Scrambling code	Integer	Scrambling code
Ec/NO active cell	Integer	Ec/NO
STTD status	Integer	BS STTD status 0 = STTD not active on PCCPCH. 1 = STTD active on PCCPCH
Cell count monitored	Integer	Number of cells in the monitored set
Channel	Integer	Channel number.
Scrambling code	Integer	Scrambling code
Ec/NO monitored cell	Integer	Ec/NO
STTD status	Integer	BS STTD status 0 = STTD not active on PCCPCH. 1 = STTD active on PCCPCH
Cell count detected	Integer	Number of cells in the detected set
Channel	Integer	Channel number.

ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)

Scrambling code	Integer	Scrambling code
Ec/N0 detected cell	Integer	Ec/N0
STTD status	Integer	BS STTD status 0 = STTD not active on PCCPCH. 1 = STTD active on PCCPCH
Cell count undetected	Integer	Number of cells in the neighbor list that are undetected.
Channel	Integer	Channel number.
Scrambling code	Integer	Scrambling code
Ec/N0 undetected cell	Integer	Ec/N0
STTD status	Integer	BS STTD status 0 = STTD not active on PCCPCH. 1 = STTD active on PCCPCH

Ας δούμε τώρα ένα παράδειγμα για το πώς αναλύεται η γραμμή ECN0

```
ECN0 0.000000 0.000000 0 0 0 0 13:29:35.292 12 1 10764 -84 4 3 10764 24 -8 0
10764 192 -11 0 10764 152 -12 0 3 10764 120 -13 0 10764 376 -15 0 10764 144 -19
0 2 10764 360 -12 1 10764 216 -18 1 0
```

Τα αρχικά νούμερα μέχρι την ώρα αντιστοιχούν σε φυσικές παραμέτρους.

0.000000 → longitude
 0.000000 → latitude
 0 → height
 0 → distance
 0 → GPS status
 0 → number of satellites
 0 → velocity

13:29:35.292 → hour:min: sec.thousands of seconds

Από εδώ και κάτω η ανάλυση γίνεται σύμφωνα με τον παραπάνω πίνακα

12 → WCDMA FDD 2100
 1 → Number of carriers
 10764 → carrier channel number
 -84 → carrier RSSI
 4 → number of parameters for each cell

3 → number of cells in the active set (ότι νούμερο υπάρχει εδώ αντιστοιχεί στον αριθμό των server και θα κάνει τόσες φορές το loop channel number, scrambling code, ECN0, STTD not active on PCCPCH)

10764 → channel number
 24 → scrambling code
 -8 → ECN0

0→STTD not active on PCCP

10764→channel number

192→scrambling code

-11→ECN0

0→STTD not active on PCCPCH

10764→channel number

152→scrambling code

-12→ECN0

0→STTD not active on PCCPCH

3→number of cells in the monitored set (ότι νούμερο υπάρχει εδώ αντιστοιχεί στον αριθμό των neighbors monitored και θα κάνει τόσες φορές το loop channel number, scrambling code, ECN0, STTD not active on PCCPCH)

10764→channel number

120→scrambling code

-13→ECN0

0→STTD not active on PCCPCH

10764→channel number

376→scrambling code

-15→ECN0

0→STTD not active on PCCPCH

10764→channel number

144→scrambling code

-19→ECN0

0→STTD not active on PCCPCH

2→number of cells in the detected set(ότι νούμερο υπάρχει εδώ αντιστοιχεί στον αριθμό των neighbors detected και θα κάνει τόσες φορές το loop channel number, scrambling code, ECN0, STTD not active on PCCPCH)

10764→channel number

360→scrambling code

-12→ECN0

1→STTD active on PCCPCH

10764→channel number

216→scrambling code

-18→ECN0

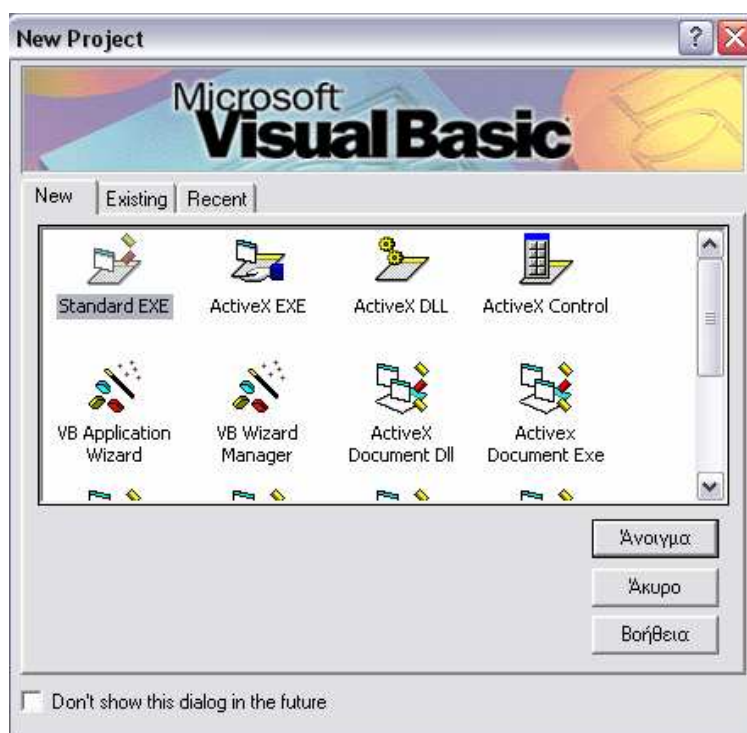
1→STTD not active on PCCPCH

0→δεν υπάρχουν neighbors undetected

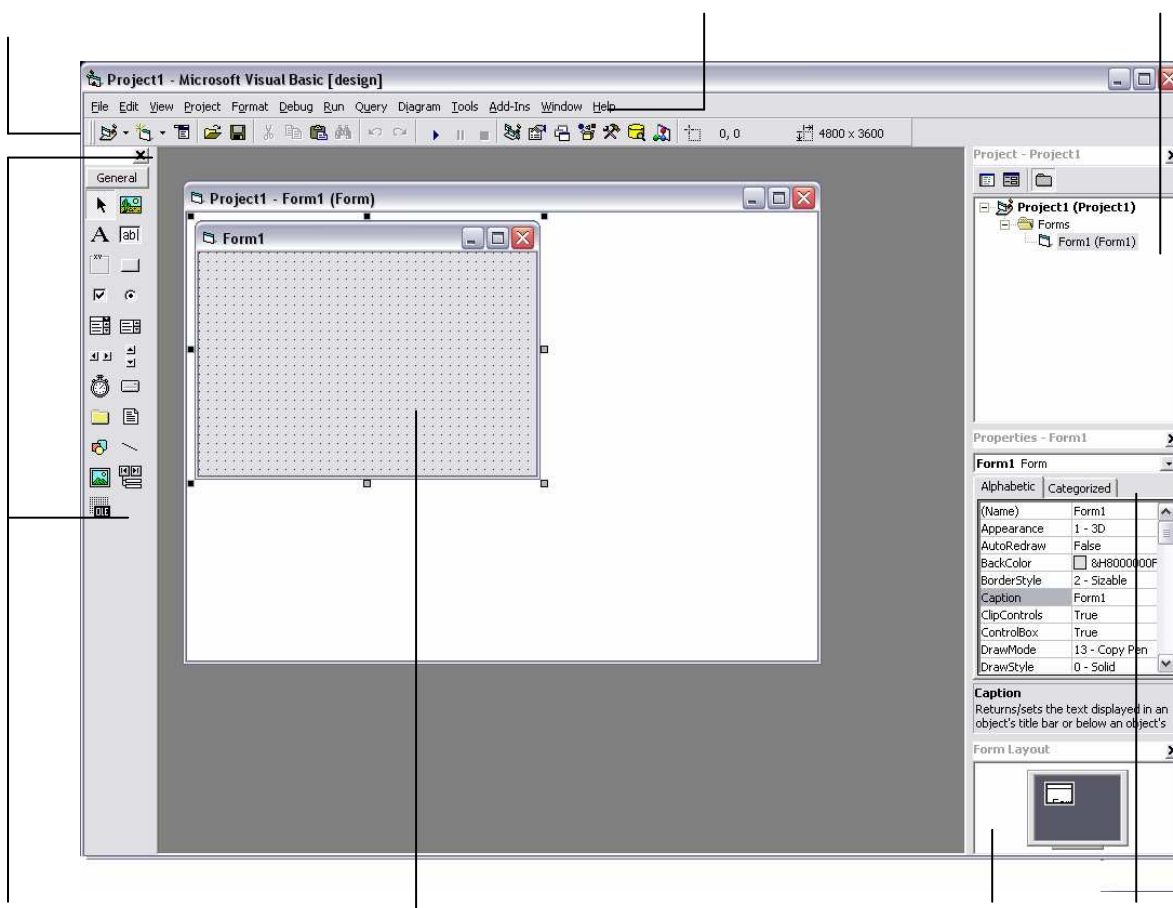
VISUAL BASIC 6

ΕΚΚΙΝΗΣΗ ΤΗΣ VISUAL BASIC 6

Το πρώτο πράγμα που κάνουμε είναι να ανοίξουμε ένα πρόγραμμα της VISUAL BASIC. Από το μενού Start των Windows επιλέγουμε Programs, Microsoft Visual Studio 6.0 και κατόπιν Microsoft Visual Basic 6.0. Εμφανίζεται το παράθυρο διαλόγου New Project από το οποίο επιλέγουμε τον τύπο Standard EXE και κάνουμε κλικ στο κουμπί Open.



Με την έναρξη της Visual Basic βρισκόμαστε στο ολοκληρωμένο περιβάλλον ανάπτυξης Integrated Development Environment (IDE) της Visual Basic. Αυτός είναι ο χώρος προγραμματισμού όπου θα δημιουργήσουμε το πρόγραμμά μας.



ΕΙΚΟΝΑ 1

Το ολοκληρωμένο περιβάλλον ανάπτυξης της Visual Basic

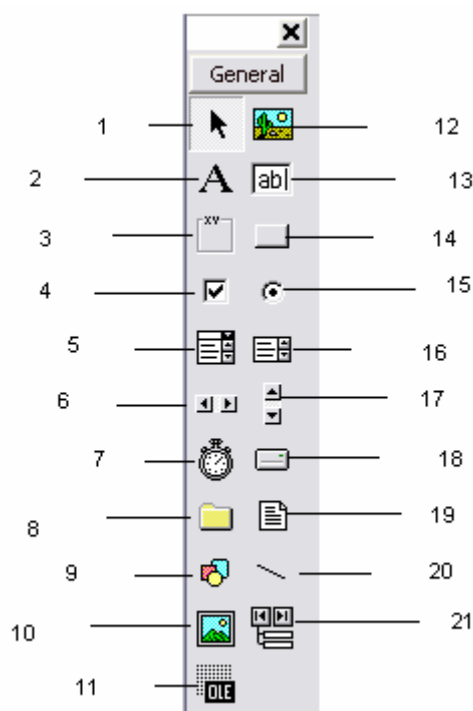
- 1 Εργαλειοθήκη
- 2 Μενού
- 3 Βασική γραμμή εργαλείων
- 4 Παράθυρο φόρμας
- 5 Παράθυρο έργου
- 6 Παράθυρο ιδιοτήτων
- 7 Παράθυρο διάταξης φόρμας

Παρατηρούμε ότι το περιβάλλον IDE μοιάζει με ένα παράθυρο που περιέχει κάποιες κουκίδες και μια γραμμή τίτλου με την λέξη Form1. Πρόκειται για το παράθυρο μιας φόρμας η οποία αποτελεί το βασικό δομικό στοιχείο για τις εφαρμογές της Visual Basic. Μια εφαρμογή μπορεί να έχει οποιοδήποτε αριθμό φορμών ή ακόμα να αποτελείται και από μια μόνο φόρμα. Η εφαρμογή που αναπτύξαμε διαθέτει μόνο μια φόρμα.

[Τι είναι οι Φόρμες και τα στοιχεία ελέγχου](#)

Στη Visual Basic οι φόρμες αποτελούν τα θεμελιώδη συστατικά που χρησιμοποιούνται για τη δημιουργία προγραμμάτων. Μια φόρμα είναι ο χώρος όπου τοποθετούμε όλα τα προγράμματα με τα οποία πρόκειται να έρθουν σε επαφή οι χρήστες του προγράμματος μας. Τα προγράμματα αυτά τα οποία τοποθετούμε στη φόρμα είναι τα στοιχεία ελέγχου τα οποία δίνουν στους χρήστες την δυνατότητα να κάνουν διάφορες ενέργειες όπως να καταχωρούν κείμενα, να κάνουν κλικ στα κουμπιά κλπ.

Αριστερά της εικόνας 1 βλέπουμε την παρακάτω εργαλειοθήκη:



ΕΙΚΟΝΑ 2
Η εργαλειοθήκη

- | | |
|---|--|
| 1 Pointer (Δείκτης) | 12 PictureBox (Πλαίσιο εικόνας) |
| 2 Label (Ετικέτα) | 13 TextBox (Πλαίσιο κειμένου) |
| 3 Frame (Πλαίσιο) | 14 CommandButton (Κουμπί εντολής) |
| 4 CheckBox (Πλαίσιο ελέγχου) | 15 OptionButton (Κουμπί επιλογής) |
| 5 combobox (Πλαίσιο) | 16 ListBox (Πλαίσιο λίστας) |
| 6 ScrollBar (Οριζόντια γραμμή κύλισης) | 17 ScrollBar (Κατακόρυφη γραμμή κύλισης) |
| 7 Timer (Χρονόμετρο) | 18 DriveListBox(Πλαίσιο λίστας για μονάδες δίσκου) |
| 8 DirListBox (Πλαίσιο λίστας καταλόγων) | 19 FileListBox (Πλαίσιο λίστας αρχείων) |
| 9 Share (Σχήμα) | 20 Line (Γραμμή) |
| 10 OLE Container Control (Στοιχείο ελέγχου Container Control) | 21 Data (Δεδομένα) |

Από την εργαλειοθήκη μας προσθέσαμε για το πρόγραμμα μας τα εξής:
Δυο CommandButton, DriveListBox, δυο FileListBox, Label τα οποία εκτελούν τα παρακάτω:

CommandButton: το πατάμε όταν θέλουμε να τρέξουμε το πρόγραμμα. Αρχικά πατάμε το Run για να τρέξει το πρόγραμμα, έπειτα θα εμφανιστεί η φόρμα μας στη οποία θα πατάμε το CommandButton το οποίο έχουμε ονομάσει Run για να ξεκινήσει να τρέχει το πρόγραμμα. Το άλλο CommandButton το οποίο έχουμε ονομάσει ClearDatabase θα το πατάμε όταν θέλουμε να καθαρίσουμε τους πίνακες της βάσης.

DriveListBox: μας επιτρέπει να επιλέξουμε μια μονάδα του δίσκου. Αν έχουμε δυο σκληρούς θα μας εμφανίσει δυο επιλογές δηλ c: και d:

DirListBox: μας επιτρέπει να επιλέξουμε ένα φάκελο.

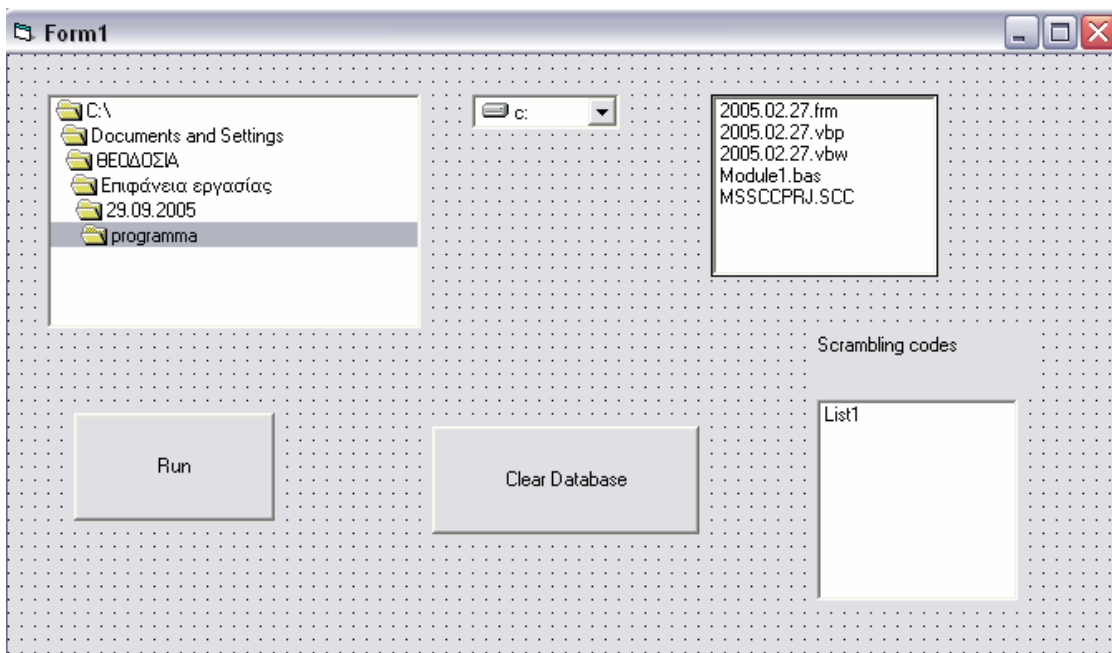
FileListBox: μας επιτρέπει να επιλέξουμε ένα αρχείο από το φάκελο που είχαμε

επιλέξει στο DirListBox. Επίσης το δεύτερο FileListBox μας εμφανίζει όλα τα scrambling codes που έχουμε στο αρχείο μας.

Label: μας εμφανίζει πόσα σε αριθμό διαφορετικά scrambling codes έχουμε στο αρχείο μας.


Τα βήματα που ακολουθήσαμε για την προσθήκη των εργαλείων μας στην εφαρμογή μας και για να ολοκληρώσουμε το πρόγραμμα μας είναι τα εξής:

1. Επιλέξαμε το στοιχείο ελέγχου **CommandButton** δυο φορές στην εργαλειοθήκη.
2. Προσθέσαμε ένα κουμπί εντολής στη φόρμα σύροντας με το δρομέα του ποντικιού το αντίστοιχο εικονίδιο του στοιχείου ελέγχου από την εργαλειοθήκη στη φόρμα. Ονομάσαμε το ένα CommandButton **Run** και το άλλο **Clear Database**
3. Χρησιμοποιήσαμε την τεχνική μεταφοράς με το ποντίκι όπως κάναμε στο βήμα 2 για να προσθέσουμε ένα **DriveListBox**, **DirListBox**, δυο **FileListBox**. Έτσι η φόρμα μας τώρα έχει την παρακάτω μορφή:



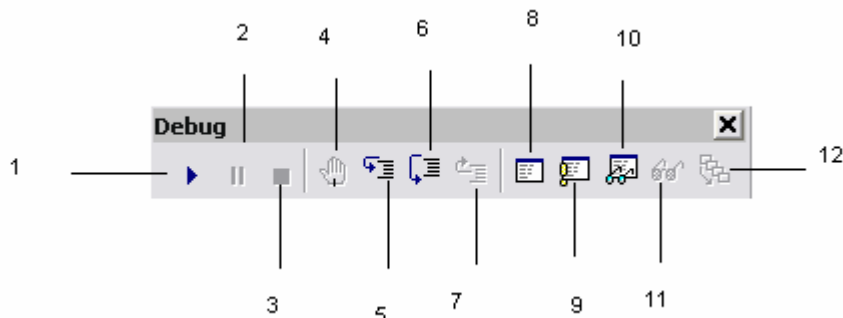
ΕΙΚΟΝΑ 3

4. Κάναμε **διπλό κλικ στο Run** για να ενεργοποιήσουμε τη διαδικασία συμβάντος Click().
5. Ενεργοποιήσαμε και τις διαδικασίες των DriveListBox DirListBox και Command Button (Clear Database).
6. Γράψαμε τον κώδικα μας.
7. Αποθηκεύσαμε τον κώδικά μας επιλέγοντας **File→Save project As→ δώσαμε το όνομα 2005.02.27** και στη συνέχεια εμφανίστηκε το παράθυρο για να **σώσουμε τη φόρμα** και την σώσαμε με το όνομα **2005.02.27**.

8. Τρέχουμε τον κώδικά μας με **Run→Start** ή πατώντας το κουμπί  από την γραμμή εργαλείων Debug ή πατώντας **F5**.
 Για να δούμε πως τρέχει το πρόγραμμα βήμα βήμα πατάμε **F8** και όχι Run.
 Μετά κάνουμε κλικ στο CommandButton και πατώντας F8 προχωράμε γραμμή γραμμή τον κώδικα.
 Για να δούμε συγκεκριμένα σημεία δηλ να δούμε τι τιμές παίρνουν κάποιες συγκεκριμένες μεταβλητές **μαρκάρουμε την μεταβλητή που θέλουμε→δεξί κλικ→Add Watch →βλέπουμε την μεταβλητή μας στο Expression →OK**.
 Όταν όμως αρχίσουμε να τρέχουμε το πρόγραμμά μας με το F8 το watch Window δεν εμφανίζεται στο κάτω μέρος της οθόνης μας και πάμε **View→ Watch Window** και εμφανίζεται.

Λίγα λόγια για την γραμμή εργαλείων Debug

Η γραμμή εργαλείων Debug (εντοπισμός και διόρθωση σφαλμάτων) μας δίνει την δυνατότητα να ελέγχουμε το πρόγραμμά μας και να λύνουμε διάφορα σφάλματα. Κατά τη διόρθωση ενός προγράμματος κάνουμε διάφορα πράγματα όπως η εκτέλεση του κώδικα κατά μια γραμμή τη φορά, εξέταση των τιμών μιας μεταβλητής και η διακοπή του κώδικα σε συγκεκριμένα κρίσιμα σημεία ή κάτω από ορισμένες συνθήκες.



ΕΙΚΟΝΑ 4

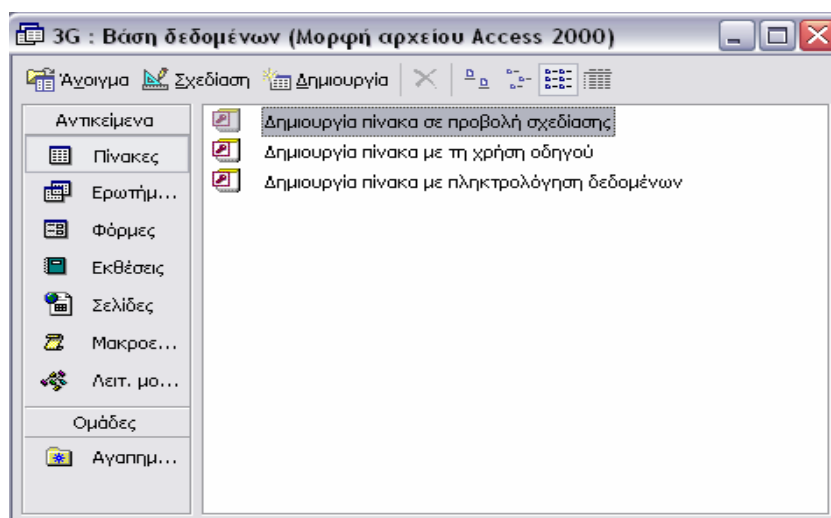
Η γραμμή εργαλείων Debug

1. Θέτει το πρόγραμμα σε κατάσταση εκτέλεσης
2. Προσωρινά διακόπτει το πρόγραμμα έτσι ώστε να εξετάσουμε τις τιμές μεταβλητών και τη ροή του προγράμματος
3. Διακόπτει το πρόγραμμα και επαναθέτει το περιβάλλον IDE σε κατάσταση σχεδίασης
4. Ορίζει ένα σημείο μέσα στο κώδικα για να θέσει το πρόγραμμα σε κατάσταση break
5. Στην κατάσταση break η μετακίνηση γίνεται μέσα στον κώδικα κατά μια γραμμή τη φορά
6. Στην κατάσταση break η μετακίνηση μέσω του κώδικα κατά μια διαδικασία τη φορά
7. Στην κατάσταση Step Into μεταβαίνει έξω από τη διαδικασία έτσι ώστε να μπορούμε να μετακινηθούμε μέσω μεγάλων τμημάτων του κώδικα γρήγορα
8. Εμφανίζει το τοπικό παράθυρο το οποίο περιέχει τα παράθυρα Watch και Call Stack και μας επιτρέπει να δούμε την κατάσταση των διαφόρων μεταβλητών του κώδικα
9. Εμφανίζει το παράθυρο Immediate έτσι ώστε να καταχωρούμε μικρά αποσπάσματα κώδικα για έλεγχο
10. Εμφανίζει το παράθυρο Watch προκειμένου να δούμε τη τιμή μιας μονής μεταβλητής ή έκφρασης
11. Εμφανίζει το παράθυρο Call Stack για να δούμε όλες τις μεταβλητές και ρουτίνες.

Σχέση Visual Basic και Access και τρόποι σύνδεση τους

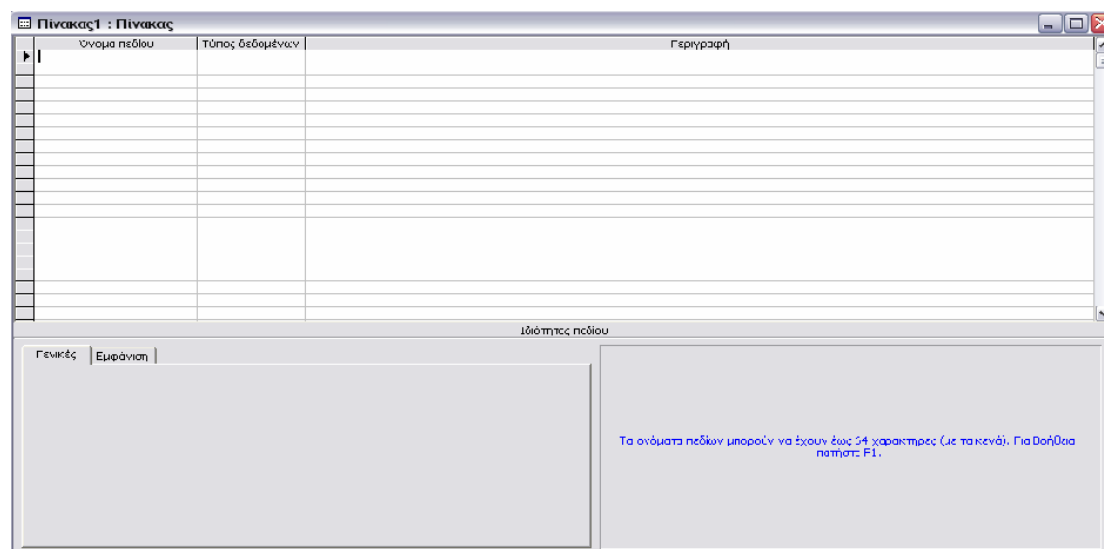
Στο πρόγραμμα μας αναλύσαμε ένα text αρχείο μετρήσεων του μετρητικού συστήματος TOM που χρησιμοποιεί η Cosmote. Αφού πήραμε το αρχείο από το μέρος που το είχαμε αποθηκεύσει στον υπολογιστή, το ανοίξαμε, το διαβάσαμε γραμμή γραμμή και όταν βρήκαμε τη γραμμή του ECN0 αναλύσαμε ένα ένα τα νούμερα που είχε και τα τοποθετήσαμε στους κατάλληλους πίνακες της βάσης που είχαμε δημιουργήσει στην Access.

Ας δούμε τώρα πως δημιουργήσαμε την βάση μας. Αρχικά ανοίξαμε την Microsoft Access **Start → Programs → Microsoft Access** και ακολουθήσαμε τα παρακάτω. Επιλέξαμε από το περιβάλλον της Microsoft Access **Δημιουργία → Κενή Βάση δεδομένων → δώσαμε το όνομα 3G.mdb** επιλέξαμε να δημιουργηθεί στο C: και πατήσαμε το κουμπί **Δημιουργία**. Η Access εμφάνισε το παρακάτω παράθυρο:



ΕΙΚΟΝΑ 5

Στη συνέχεια επιλέξαμε **Δημιουργία πίνακα σε προβολή σχεδίασης**

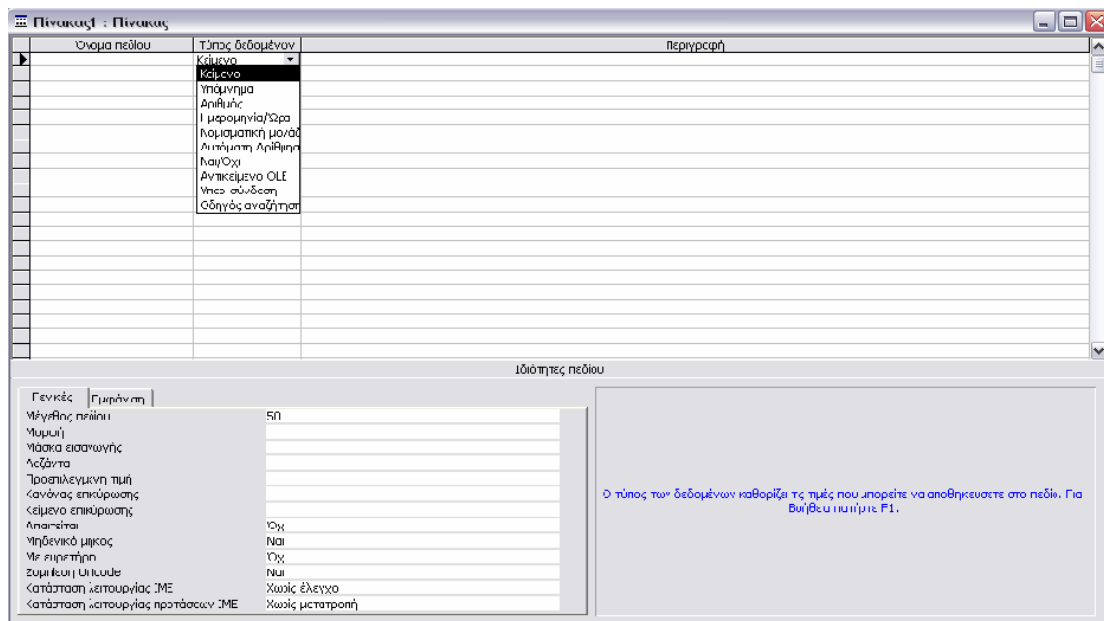


ΕΙΚΟΝΑ 6

ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)

Στην προβολή σχεδίασης μας πληκτρολογήσαμε αρχικά τα ονόματα των πεδίων που θέλαμε να φτιάξουμε σε κάθε πίνακα και στη συνέχεια ορίσαμε τον τύπο κάθε πεδίου. Ο τύπος δεδομένων ενός πεδίου καθορίζει το είδος των πληροφοριών που μπορούν να καταχωρηθούν στο πεδίο και τον τρόπο με τον οποίο η Access μπορεί να χρησιμοποιήσει αυτές τις πληροφορίες. Για να εξετάσουμε τους τύπους δεδομένων που χρησιμοποιούνται στον πίνακα ακολουθήσαμε τα παρακάτω βήματα:

1. Πρώτα πατάμε **Enter** για να μεταφερθούμε στη στήλη Τύπος δεδομένων.
2. Πατάμε στο κουμπί βέλος της στήλης για να εμφανιστεί ο επόμενος κατάλογος με τύπους από τον οποίο επιλέγουμε τι τύπο θέλουμε να δώσουμε σε κάθε πεδίο μας.



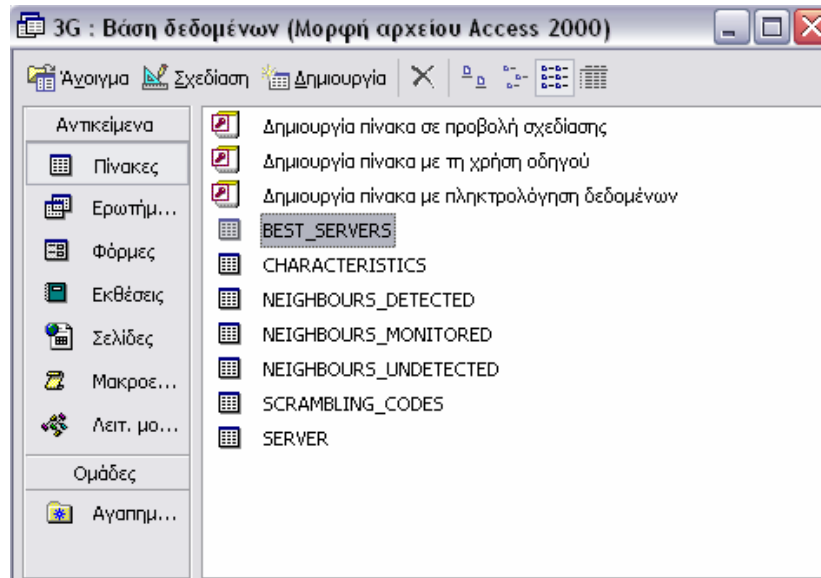
ΕΙΚΟΝΑ 7

3. Σώζουμε τον πίνακα μας επιλέγοντας **Αρχείο → Αποθήκευση**



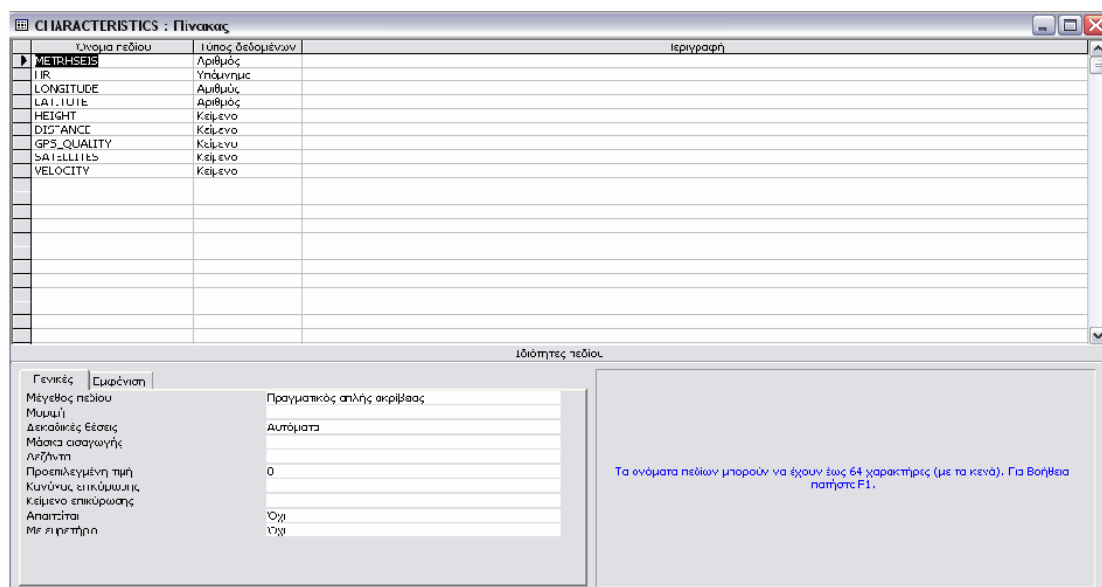
ΕΙΚΟΝΑ 8

Με τα παραπάνω βήματα δημιουργήσαμε 7 πίνακες με τα εξής ονόματα: CHARACTERISTICS, NEIGHBOURS_DETECTED, NEIGHBOURS_MONITORED, NEIGHBOURS_UNDETECTED, SERVER, SCRAMBLING_CODES και BEST_SERVERS.



ΕΙΚΟΝΑ 9

Τα ονόματα που δώσαμε στα πεδία του κάθε πίνακα καθώς και οι τύποι που ορίσαμε σε κάθε πεδίο φαίνονται παρακάτω:



ΕΙΚΟΝΑ 10
Πίνακας CHARACTERISTICS

ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)

Όνομα πεδίου	Τύπος δεδομένων	Περιγραφή
MLTRICCID	Αριθμός	
HR	Υπόμνημα	
C_RSSI	Αριθμός	
NO_CELLS_DET_SET	Κείμενο	
CH_NO	Κείμενο	
SCR_CODE	Αριθμός	
ECMO	Αριθμός	
BS_STTD	Κείμενο	
RSCP	Αριθμός	

Γενικές		Εμφάνιση	
Μέγεθος πεδίου	Ακέραιος μεγάλου μήκους		
Μορφή	Αυτόματα		
Λειτουργίες: Άρνησις			
Μάσκα εισαγωγής			
Λεζάντα	0		
Προεπιλεγμένη τιμή			
Κανόνες ειδικού κ. κείμενου επικύρωσης	Όχι		
Απαιτείται	Όχι		
Με ευρετήριο			

Τα ονόματα πεδίων μπορούν να έχουν έως 64 χαρακτήρες (με τα κενά). Για Βοήθεια πατήστε F1.

ΕΙΚΟΝΑ 11
Πίνακας NEIGHBOURS_DETECTED

Όνομα πεδίου	Τύπος δεδομένων	Περιγραφή
MLTRHSEIS	Αριθμός	
HR	Υπόμνημα	
C_RSSI	Αριθμός	
NO_CELLS_MON_SET	Κείμενο	
CH_NO	Κείμενο	
SCR_CODE	Αριθμός	
ECMO	Αριθμός	
BS_STTD	Κείμενο	
RSCP	Αριθμός	

Γενικές		Εμφάνιση	
Μέγεθος πεδίου	Ακέραιος μεγάλου μήκους		
Μορφή	Αυτόματα		
Δεκαδικές θέσεις			
Μάσκα εισαγωγής			
Λεζάντα	0		
Προεπιλεγμένη τιμή			
Κανόνες επικύρωσης	Όχι		
Κείμενο επικύρωσης	Όχι		
Απαιτείται			
Με ευρετήριο			

Τα ονόματα πεδίων μπορούν να έχουν έως 54 χαρακτήρες (με τα κενά). Για Βοήθεια πατήστε F1.

ΕΙΚΟΝΑ 12
Πίνακας NEIGHBOURS_MONITORED

ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)

Όνομα πεδίου	Τύπος δεδομένων	Περιγραφή
NEIGHBOURS	Αριθμός	
HR	Υπόμνημα	
C_RSSI	Αριθμός	
NC_CE_L5_UND	Κείμενο	
CH_NO	Κείμενο	
SCR_CODE	Αριθμός	
ECNO	Αριθμός	
BS_>1 ID	Κείμενο	
RSCP	Αριθμός	

Γενικές	Εμφάνιση
Μέγεθος πεδίου	Ακέραιος μεγάλου μήκους
Μορφή	Αυτόματα
Δεκαδικές θέσεις	
Μάσκα εισαγωγής	
Λεζάντα	0
Προπιλεγμένη τιμή	
Κανόνες επικύρωσης	Όχι
Κείμενο επικύρωσης	Όχι
Απαιτείται	
Με ευρετήριο	

Τα ονόματα πεδίων μπορούν να έχουν έως 64 χαρακτήρες (με τα κενά). Για Βοήθεια πατήστε F1.

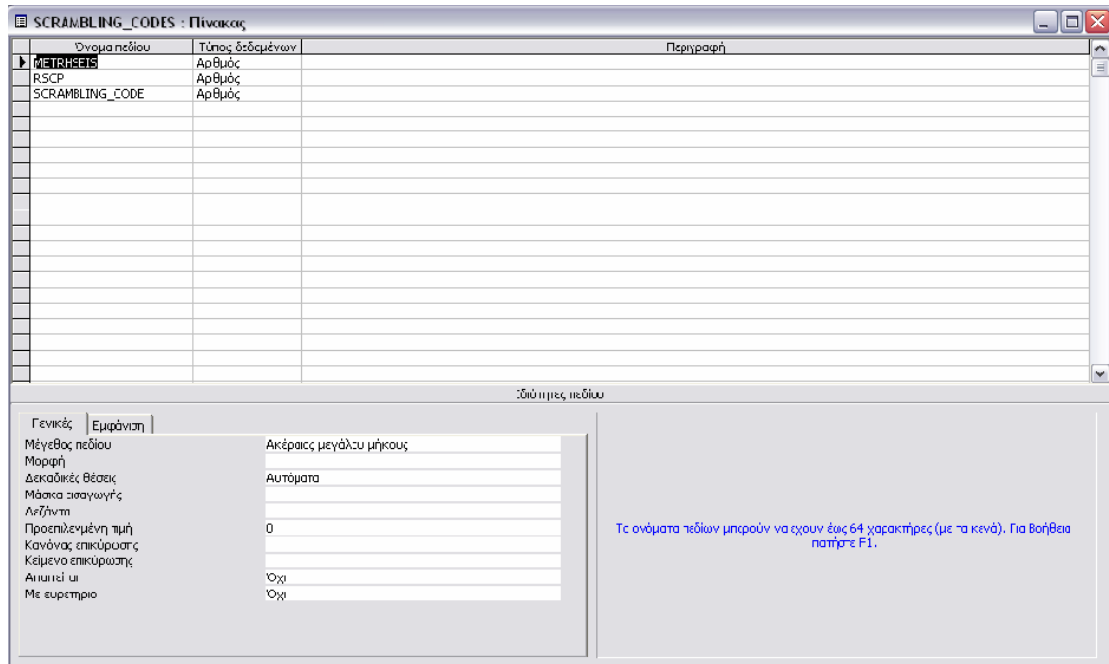
**Πίνακας NEIGHBOURS_UNDETECTED
ΕΙΚΟΝΑ 13**

Όνομα πεδίου	Τύπος δεδομένων	Περιγραφή
NEIGHBOURS	Αριθμός	
HR	Υπόμνημα	
WCDMA	Κείμενο	
NC_C	Κείμενο	
CH_NO_C	Κείμενο	
C_>5SI	Αριθμός	
NC_PAR	Κείμενο	
NC_OF_CELLS_ACT_SET	Κείμενο	
CH_NO	Κείμενο	
SCR_CODE	Αριθμός	
ECNO	Αριθμός	
BS_STTD	Κείμενο	
RSCP	Αριθμός	

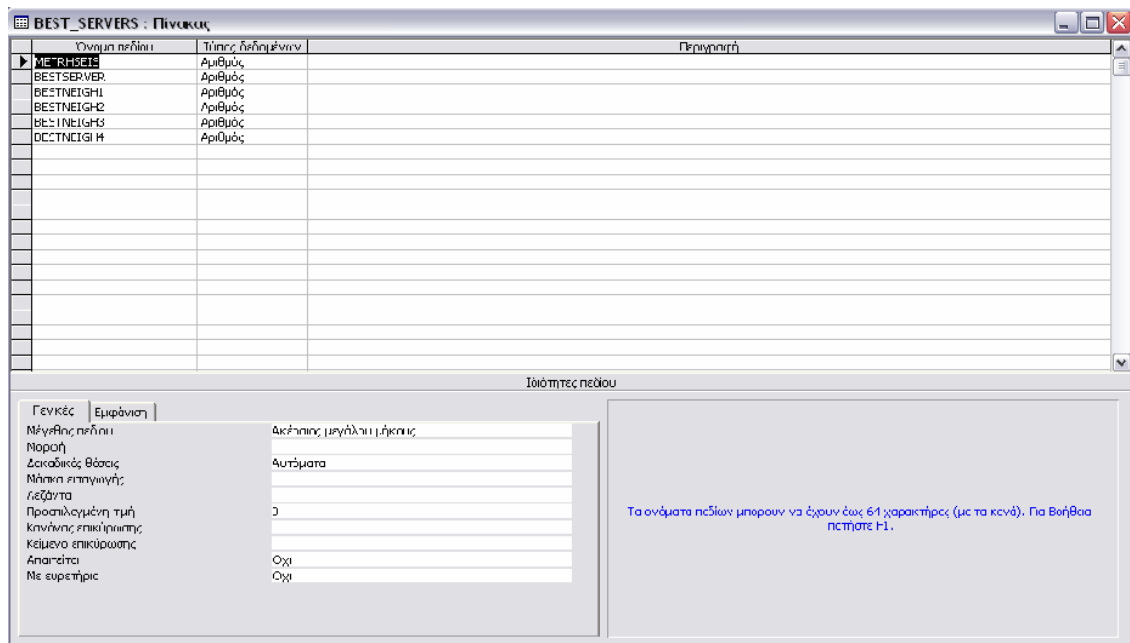
Γενικές	Εμφάνιση
Μέγεθος πεδίου	Ακέραιος μεγάλου μήκους
Μορφή	Αυτόματα
Δεκαδικές θέσεις	
Μάσκα εισαγωγής	
Λεζάντα	0
Προπιλεγμένη τιμή	
Κανόνες επικύρωσης	Όχι
Κείμενο επικύρωσης	Όχι
Απαιτείται	
Με ευρετήριο	

Τα ονόματα πεδίων μπορούν να έχουν έως 64 χαρακτήρες (με τα κενά). Για Βοήθεια πατήστε F1.

**Πίνακας SERVER
ΕΙΚΟΝΑ 14**



**Πίνακας SCRAMBLING_CODES
ΕΙΚΟΝΑ15**



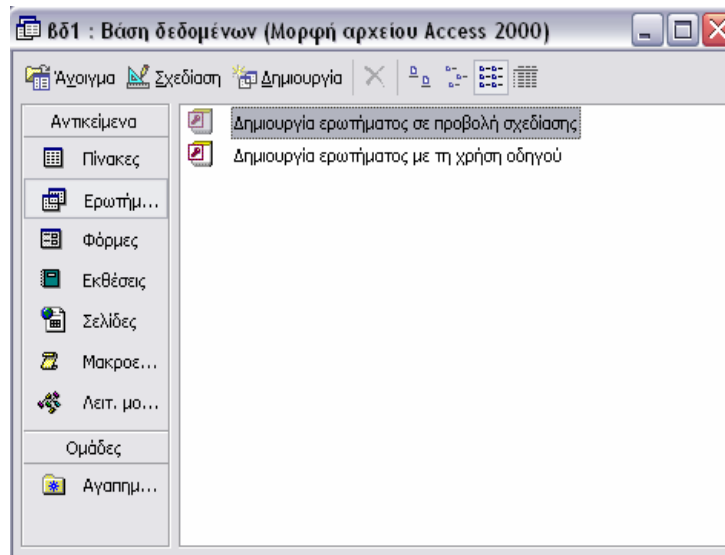
**Πίνακας BEST_SERVERS
ΕΙΚΟΝΑ 16**

Έτσι όταν θα τρέχουμε το πρόγραμμα μας στην Visual Basic θα γεμίζουν οι αντίστοιχοι πίνακες της βάσης με τα νούμερα που θέλουμε.

Στη συνέχεια φτιάξαμε κάποια ερωτήματα στη βάση μας τα οποία θα μας βοηθήσουν στη δημιουργία των γραφικών μας. Έτσι σύμφωνα με τις ομαδοποιήσεις που φτιάξαμε απεικονίζεται με διαφορετικό χρώμα η κάθε ομάδα. Περισσότερα όμως με τις γραφικές και τα χρώματα αναφέρονται στο κεφάλαιο του MATLAB στο οποίο και φτιάξαμε τις γραφικές.

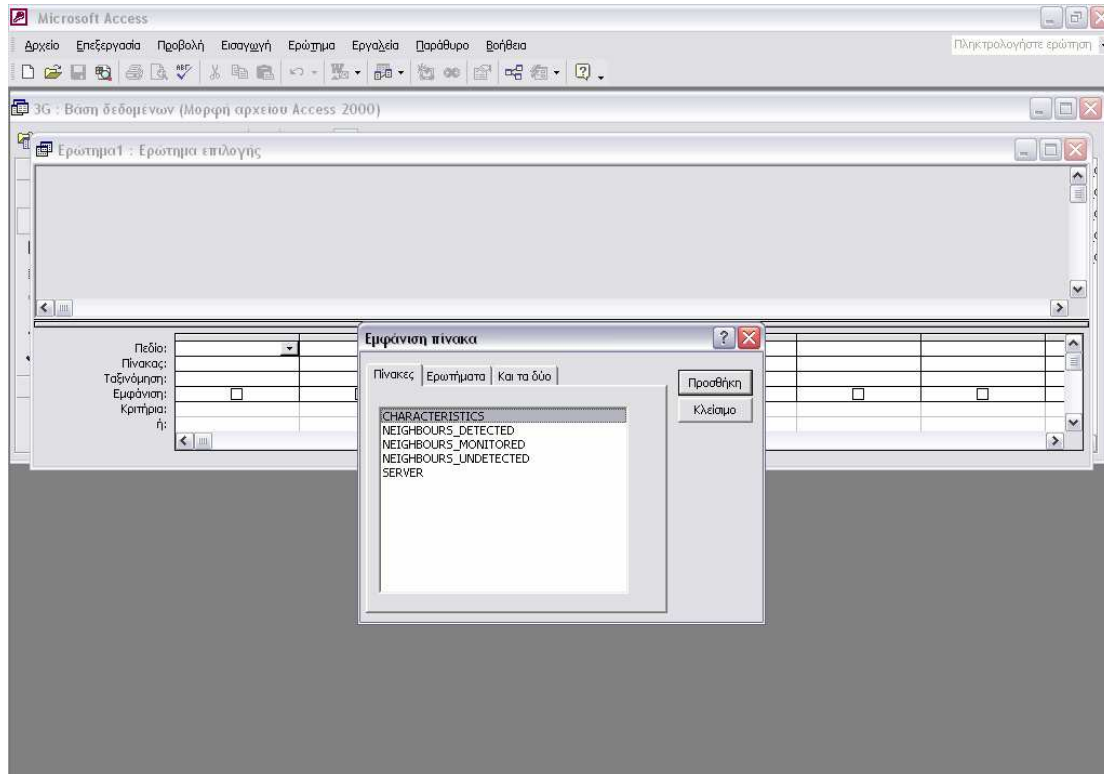
Για τη δημιουργία των ερωτημάτων κάναμε την εξής διαδικασία που περιγράφεται παρακάτω:

Επιλέξαμε από τα αντικείμενα της βάσης τα **ερωτήματα** και επιλέξαμε με διπλό κλικ **δημιουργία ερωτήματος σε προβολή σχεδίασης** επιλέξαμε τον πίνακα **CHARACTERISTICS** και τον πίνακα **NEIGHBOURS_DETECTED** έπειτα επιλέξαμε κλείσιμο



ΕΙΚΟΝΑ 17

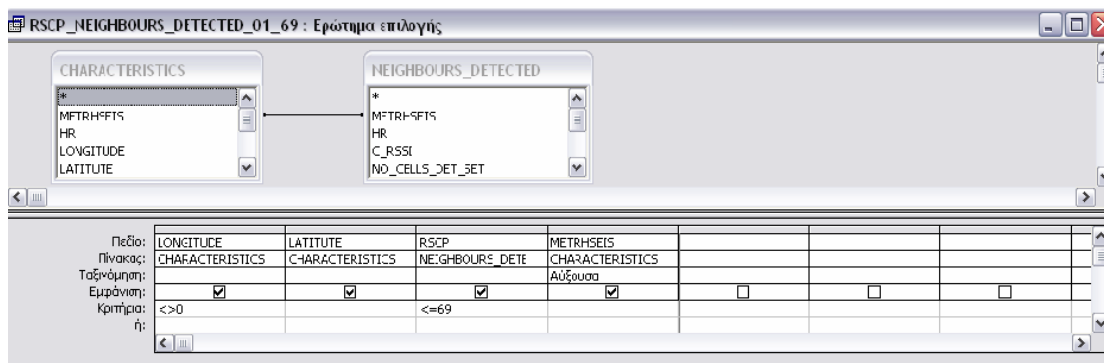
ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)



ΕΙΚΟΝΑ 18

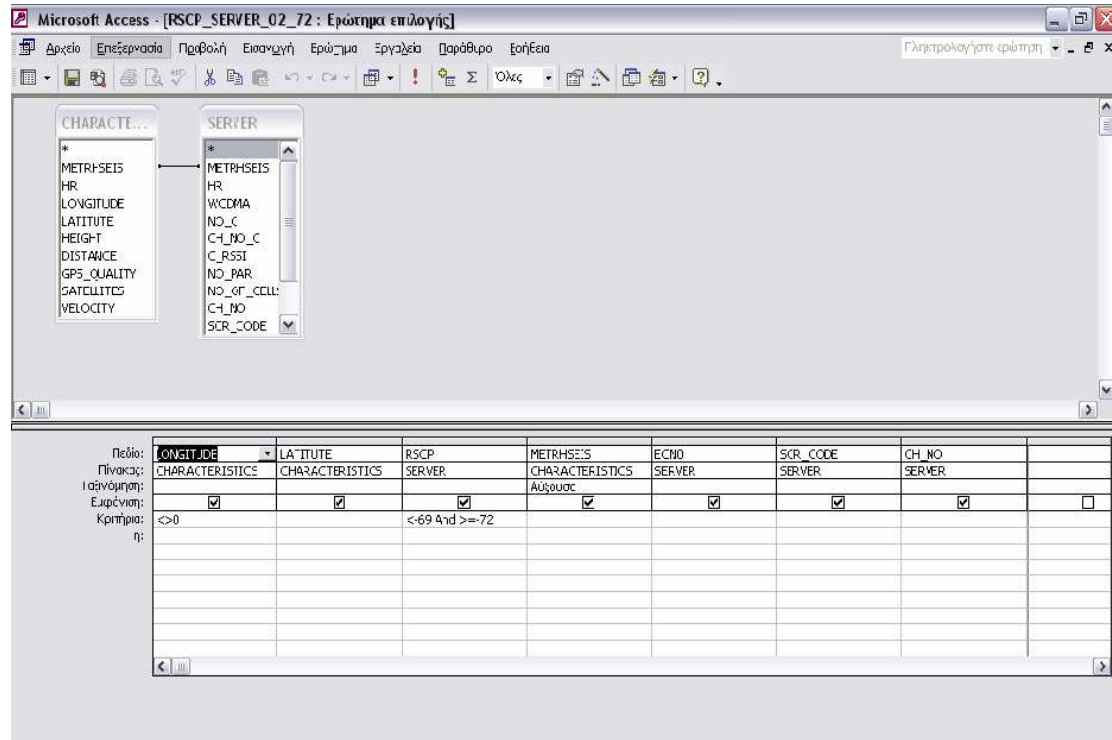
Έπειτα αφού εμφανιστούν οι πίνακες **CHARACTERISTICS** και **NEIGHBOURS_DETECTED** επιλέγω **METRHSSEIS** από τον πίνακα **CHARACTERISTICS** και έχοντας το επιλεγμένο τραβάω να φτάσει στο **METRHSSEIS** του πίνακα **NEIGHBOURS_DETECTED** έτσι ώστε να εμφανιστεί μια γραμμή που να ενώνει τις **METRHSSEIS** του πίνακα **CHARACTERISTICS** με τις **METRHSSEIS** του πίνακα **NEIGHBOURS_DETECTED**. Αυτό το κάνουμε για να παίρνουμε όλες τις τιμές που αντιστοιχούν για παράδειγμα στην δέκατη μέτρηση. Αν δεν ενώσουμε τις μετρήσεις δεν μπορούμε να πάρουμε όλες τις τιμές. Στη συνέχεια επιλέγουμε τα πεδία που θα πάρουμε, τους πίνακες και τα κριτήρια τα οποία θα χρησιμοποιήσουμε. Τέλος αποθηκεύουμε το ερώτημα από το αρχείο → αποθήκευση ως και δίνουμε το όνομα.

Παρακάτω φαίνονται το πώς έχουν διαμορφωθεί τα ερωτήματα σύμφωνα με την παραπάνω διαδικασία που περιγράψαμε.

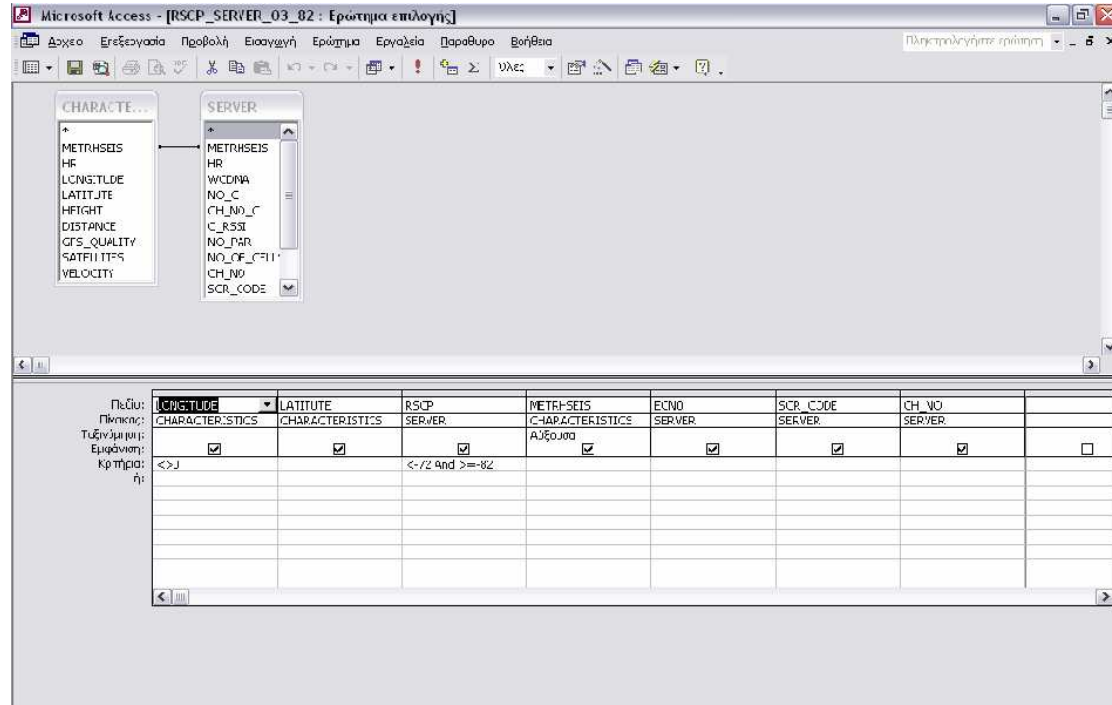


ΕΙΚΟΝΑ 19

ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)

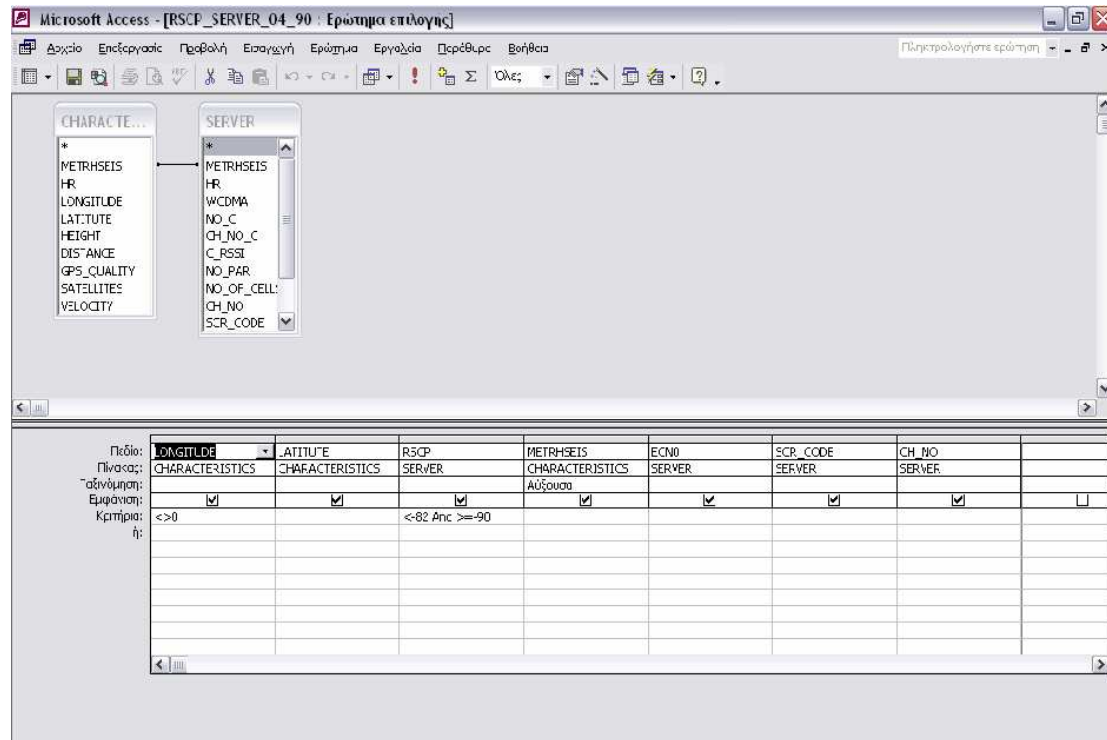


ΕΙΚΟΝΑ 20

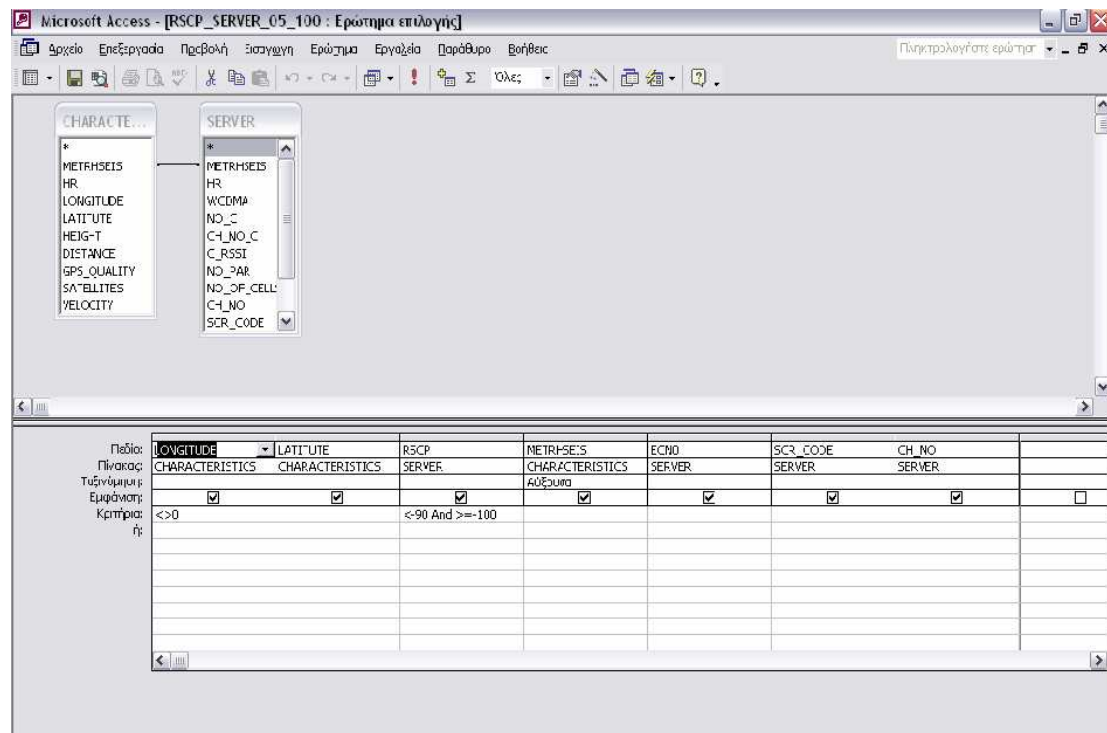


ΕΙΚΟΝΑ 21

ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)

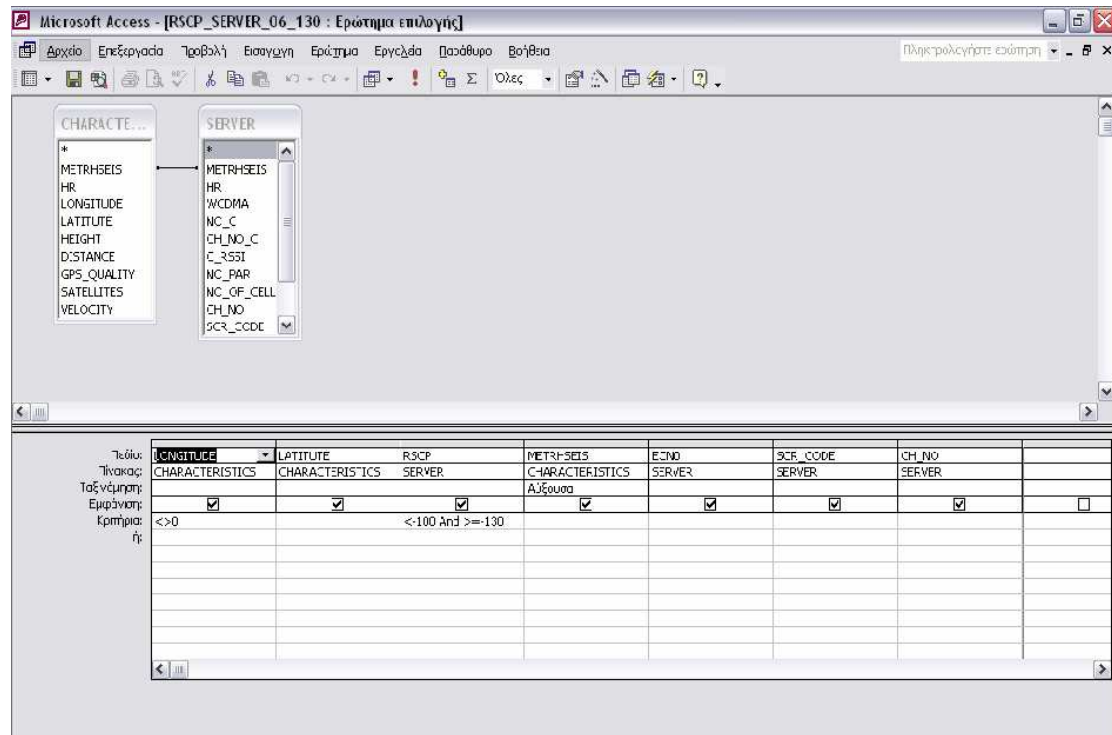


ΕΙΚΟΝΑ 22

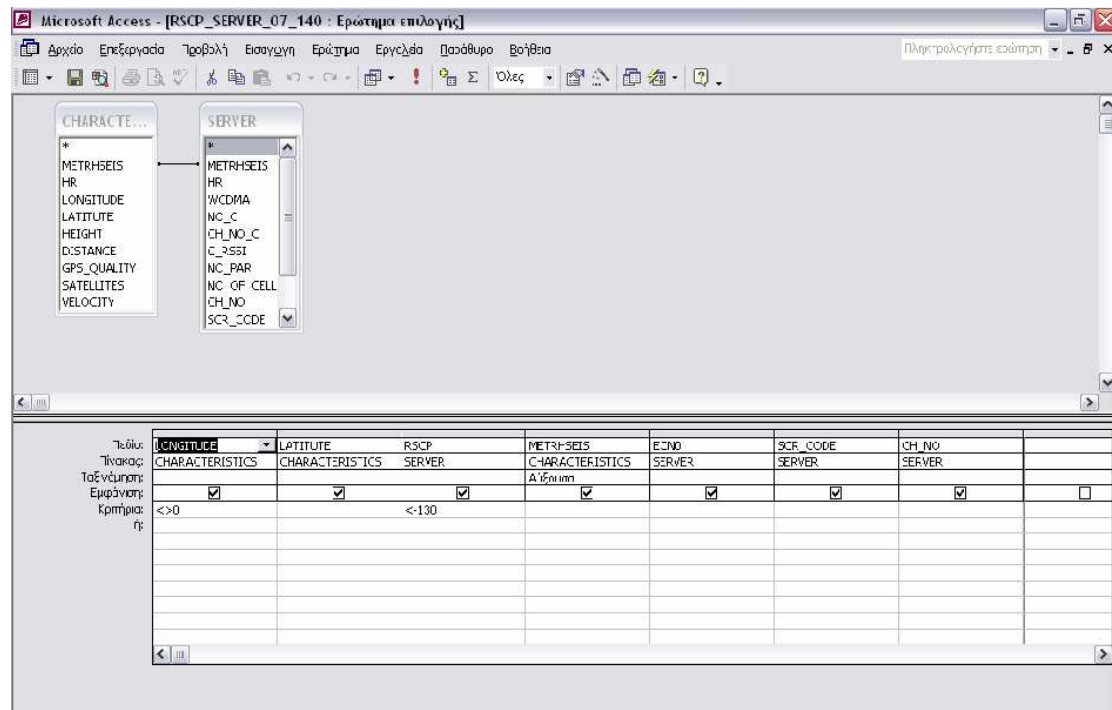


ΕΙΚΟΝΑ 23

ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)

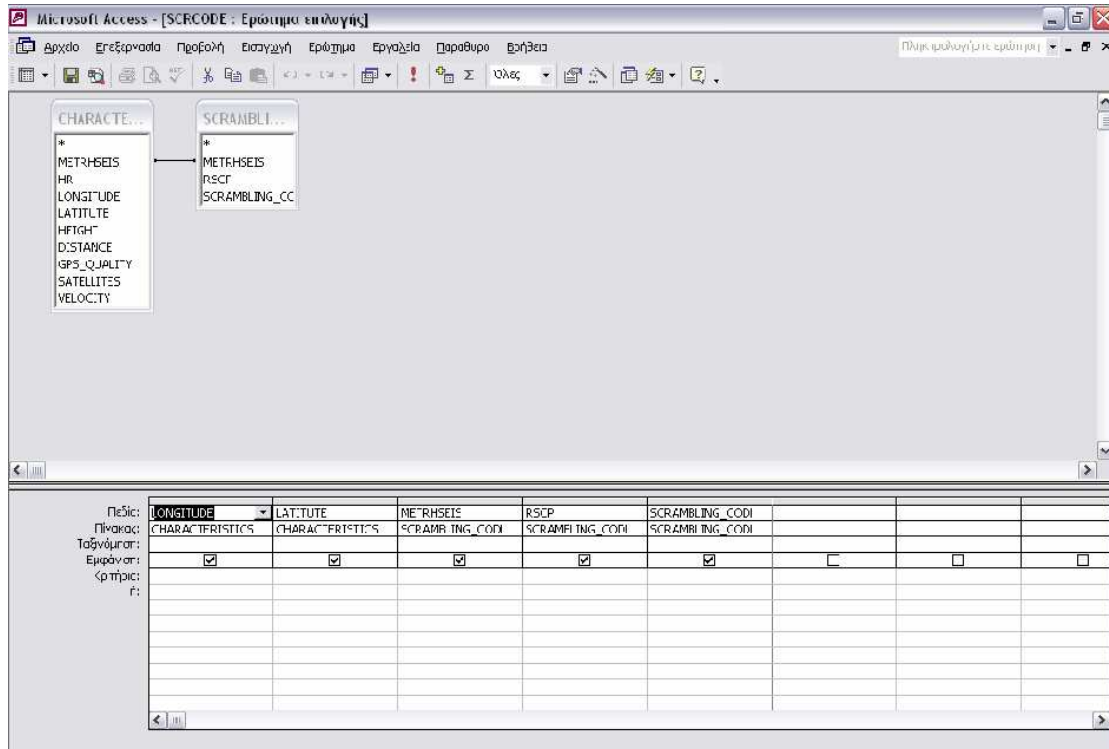


EIKONA 24

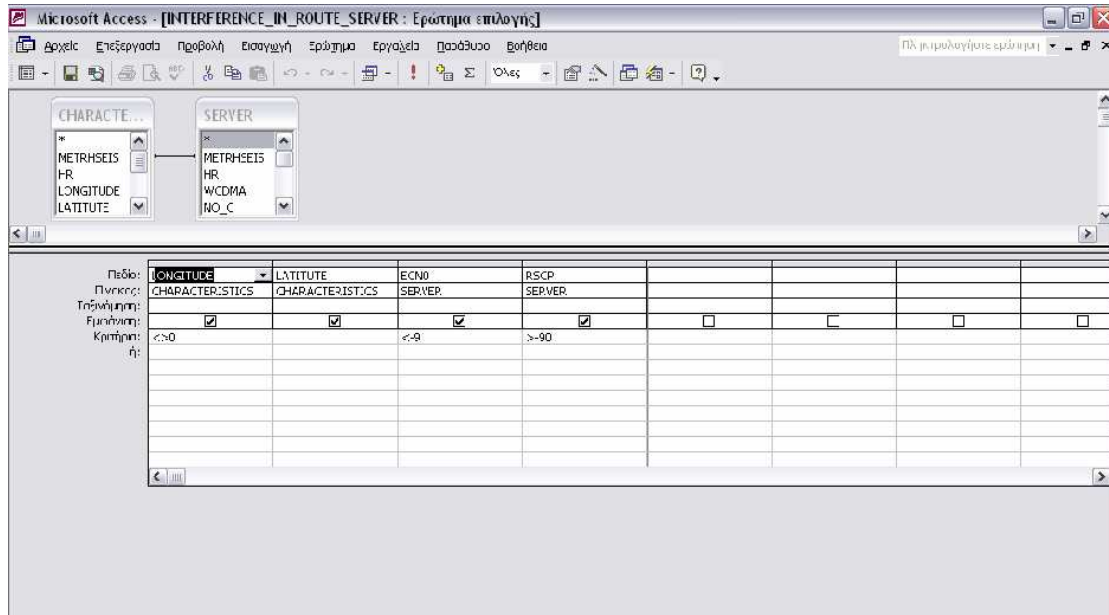


EIKONA 25

ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)

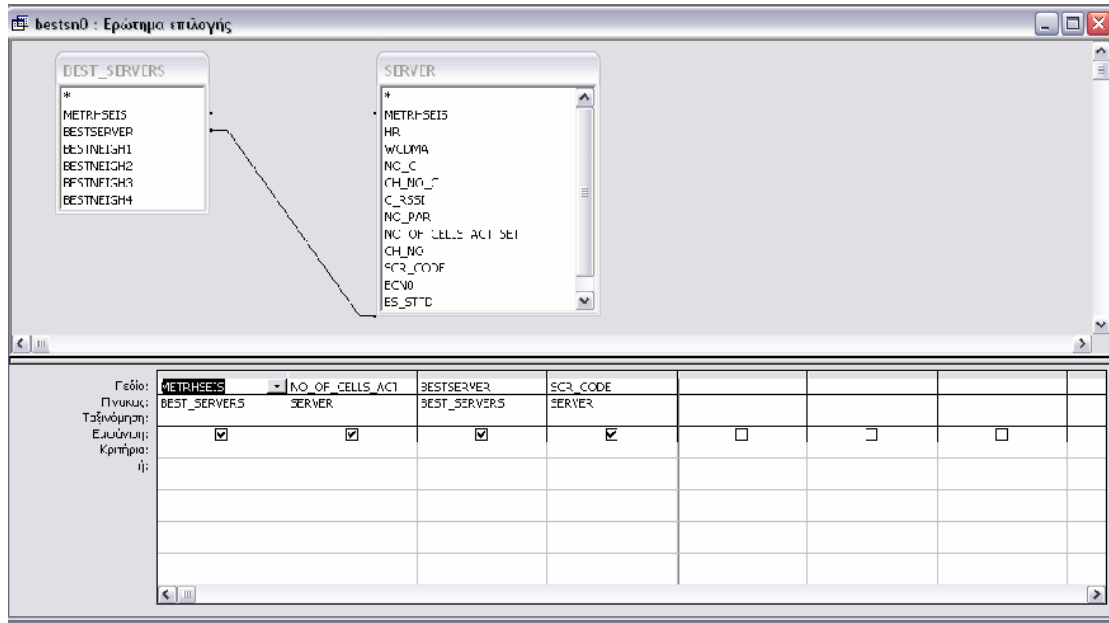


ΕΙΚΟΝΑ 26

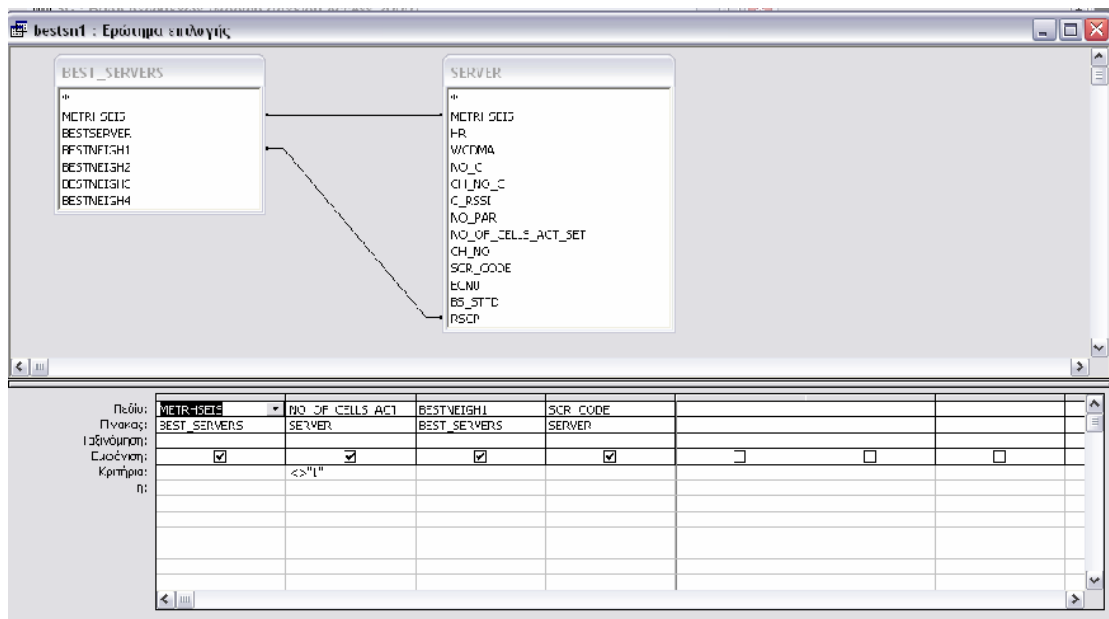


ΕΙΚΟΝΑ 27

ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)

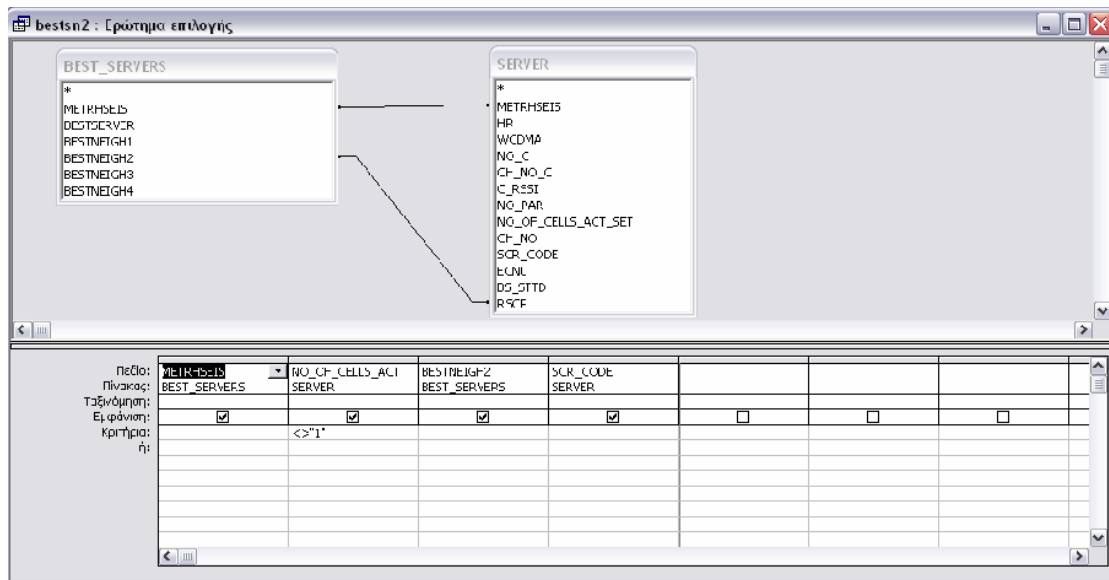


ΕΙΚΟΝΑ 30

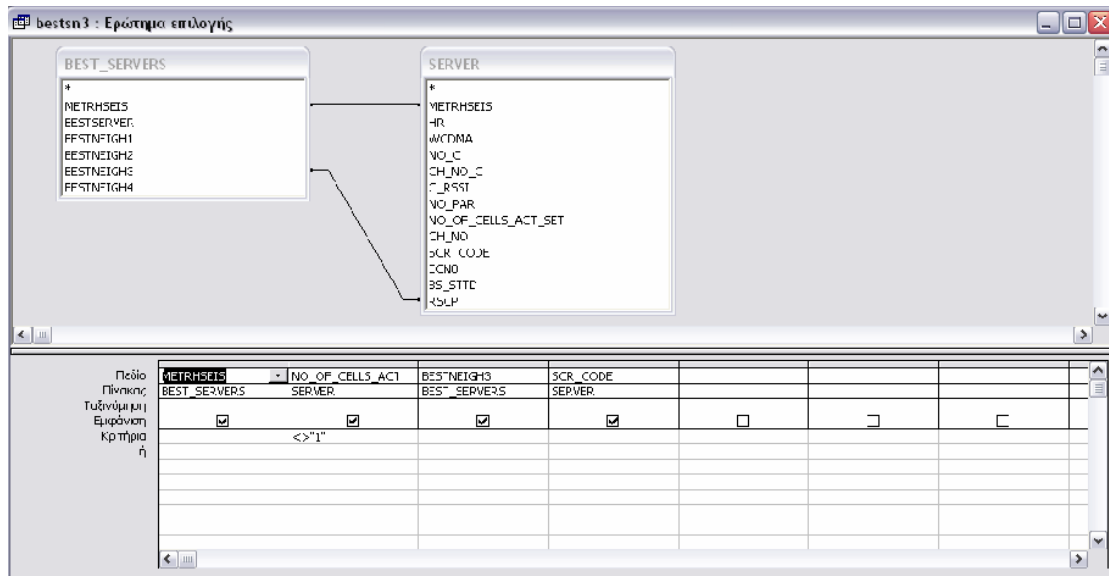


ΕΙΚΟΝΑ 31

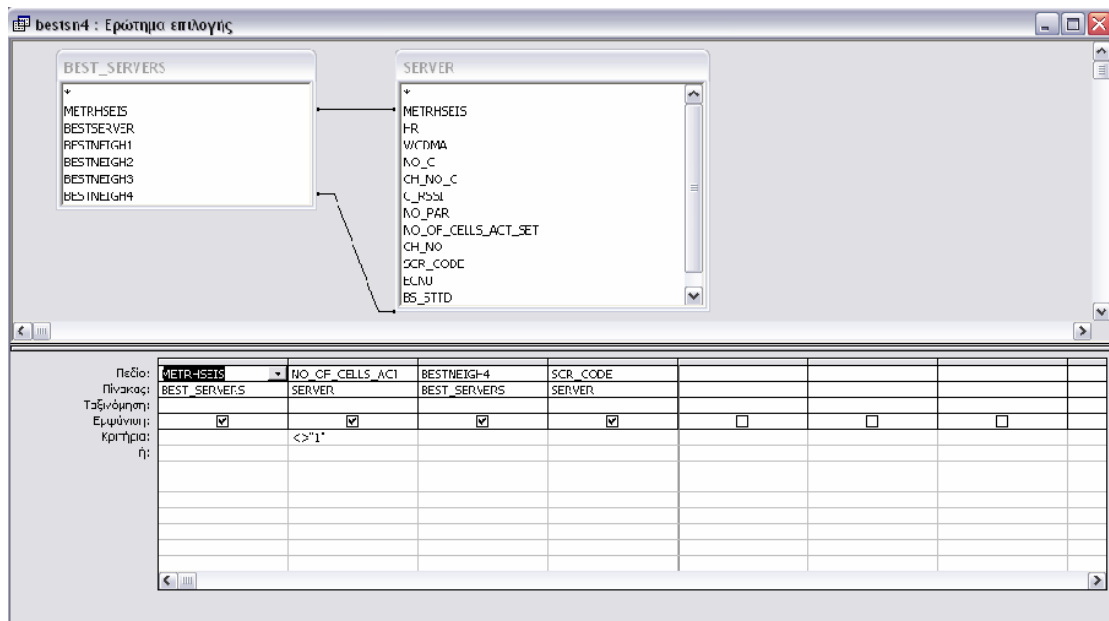
ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)



EIKONA 32

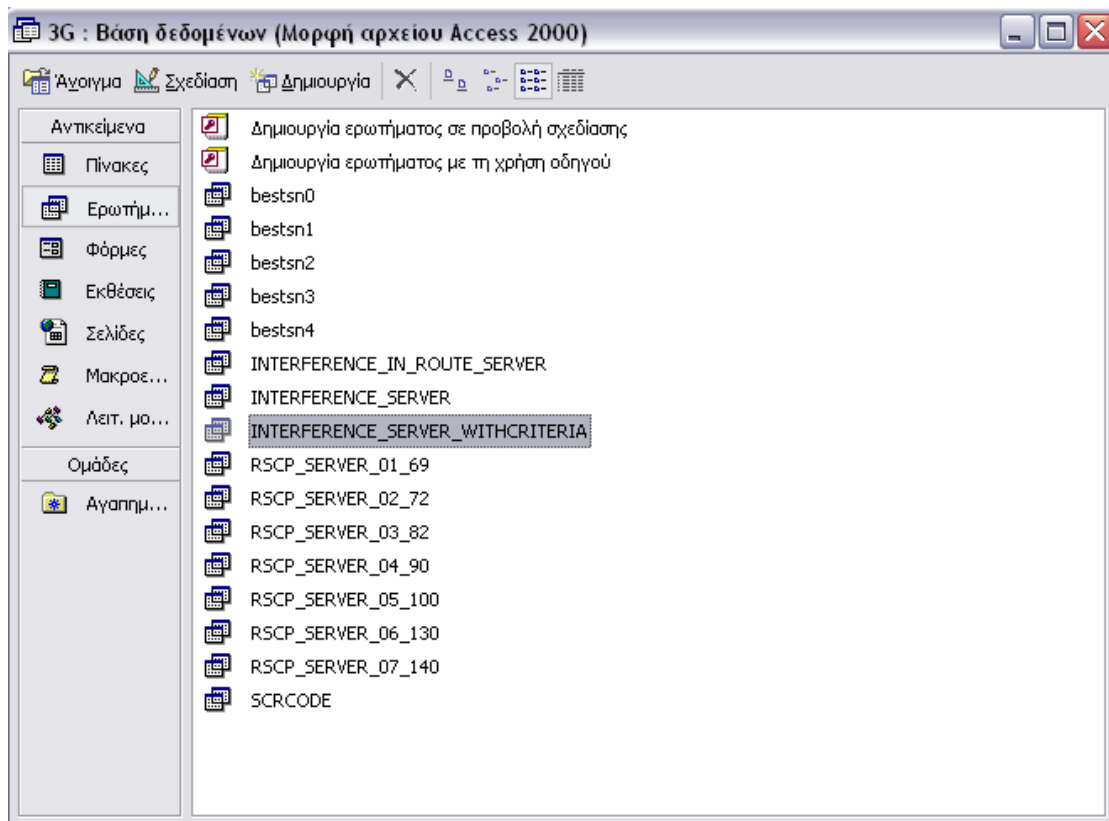


EIKONA 33



ΕΙΚΟΝΑ 34

Έτσι τα ερωτήματα που δημιουργήσαμε φαίνονται στην παρακάτω εικόνα.



ΕΙΚΟΝΑ 35

Επιστροφή στη Visual Basic και κάποια σημαντικά πράγματα που πρέπει να προσέξουμε στο πρόγραμμα μας.

Παρακάτω ακολουθεί ο κώδικας μας. Για να τρέξει σωστά και να συμπληρώσει επίσης σωστά τους πίνακες της βάσης δεδομένων μας θα πρέπει να συνδέσουμε τη βάση 3G.mdb με τη Visual Basic. Έτσι λοιπόν πριν τρέξουμε το πρόγραμμά μας **επιλέγουμε Project →References →τσεκάρουμε Microsoft DAO 3.6 Object Library →OK**. Δεν ξεχνάμε να σώσουμε ξανά το πρόγραμμά μας έχοντας τσεκαρισμένη τη βιβλιοθήκη DAO

Ας δούμε τώρα τον κώδικά μας στη Visual Basic και στη συνέχεια να επισημάνουμε ορισμένα σημεία. Στον κώδικά έχουμε σχολιάσει κάθε του γραμμή (τα σχόλια απεικονίζονται με πράσινους χαρακτήρες) έτσι ώστε να μπορούν οι αναγνώστες να καταλάβουν τη ροή του προγράμματος. Ο παρακάτω κώδικας είναι τελειοποιημένος, χωρίς λάθη και τρέχει χωρίς πρόβλημα.

Option Explicit 'με την εντολή Option Explicit καθιστούμε υποχρεωτική τη δήλωση ' όλων 'των μεταβλητών. Αν δεν τη χρησιμοποιήσουμε οι μεταβλητές 'που δεν δηλώνονται ορίζονται αυτόματα σαν variant

Public adeia

Private Sub Command1_Click() 'Αυτή η εντολή μας λέει ότι θα τρέχουμε κάθε φορά ' τον κώδικά μας κάνοντας κλικ στο Command Button που έχουμε 'σχεδιάσει στη Form. Η εντολή αυτή εμφανίζεται όταν κάνουμε 'διπλό κλικ στο Command Button που έχουμε 'σχεδιάσει στη Form '(Sub:λέξη κλειδί της VB η οποία δηλώνει τον τύπο της ' διαδικασίας). '(Command Button:με διπλό κλικ ενεργοποιείται η διαδικασία ' του κλικ).

adeia = strup

Call TextStream 'καλώντας τη διαδικασία TextStream έχουμε πρόσβαση σ'ένα αρχείο 'η TextStream μας δίνει τη δυνατότητα να διαβάσουμε να γράψουμε 'και να επισυνάψουμε αρχείο.

End Sub 'Τέλος της διαδικασίας τρεξίματος και διαβάσματος του αρχείου

Sub TextStream() 'δηλώνονται τα διάφορα μέρη της TextStream

Dim fs, f, ts, s, s2, variable 'δηλώνονται οι μεταβλητές μας 'η δήλωση των μεταβλητών γίνεται ψμε την Dim 'fs->αντικείμενο που μας δίνει τη δυνατότητα να έχουμε ' πρόσβαση σε αρχείο.

```
'f-> διαδικασία όπου παίρνουμε το αρχείο από κάποιο μέρος
'   του συστήματος
'ts-> όνομα διαδικασίας ανοίγματος του αρχείου μας
's-> μεταβλητή στην οποία εκχωρούμε το διάβασμα της
'   γραμμής του αρχείου
's2-> μεταβλητή που μας δείχνει τους 4 χαρακτήρες που
'   είναι πριν το πρώτο κενό και αν είναι ECN0 μας
'   δείχνει στη συνέχεια έναν έναν τους χαρακτήρες.
'variable-> παίρνει έναν έναν τους χαρακτήρες που
'           εμφανίζει το s2 και τους δείχνει ομαδοποιημέ-
'           νους ώσπου να βρεί κενό
```

Dim i, metr, k, cellcount, g,j, num As Integer 'δηλώνονται οι μεταβλητές μας σαν Integer

```
'i-> μετράει τον αριθμό των χαρακτήρων έναν έναν μόνο
'   όταν μπει σε γραμμή ECN0
'metr-> εμφανίζει τον αριθμό όλων των χαρακτήρων της
'   κάθε γραμμής ECN0
'k-> μετράει τον αριθμό των κενών μόνο όταν μπει σε
'   γραμμή ECN0
'cellcount-> ότι νούμερο έχει η μεταβλητή cellcount
'           τόσες φορές θα επαναληφθεί η τετράδα
'           CH_NO,SCR_CODE,ECN0,BS_STTD
```

'g-> μετράει τις γραμμές ECN0 μόνο.

Dim Flug As Boolean 'δήλωση της μεταβλητής Flug.H Flug παίρνει λογικές τιμές 0 ή '1 true ή false

Dim server() As String 'δήλωση του πίνακα server σαν αλφαριθμητικό.Στον πίνακα 'server θα καταχωρούνται μια μια οι τιμές της γραμμής ECN0

Dim dbs,rd As Database 'δήλωση της μεταβλητής dbs σαν βάση δεδομένων

Dim rst As Recordset 'η rst δηλώνεται σαν μεταβλητή εγγραφής

Dim emptflg As Boolean

Dim bestmon() As Integer

emptflg = True

Set fs = CreateObject("Scripting.FileSystemObject") 'με την Set γίνεται εκχώρηση ' μιας αναφοράς σε αντικείμενο σε μια ' μεταβλητή ή ιδιότητα.Ονομάζουμε fs ' το αντικείμενο που μας δίνει τη ' δυνατότητα να έχουμε πρόσβαση σε ' αρχείο του συστήματος. ' Η εντολή CreateObject συντάσσεται ως:

'Scripting.FileSystemObject
'και η οποία μας παρέχει πρόσβαση
'στο μσύστημα αρχείων του H/Y

Set f = fs.GetFile(Dir1.Path & "\" & File1.FileName) 'εκχωρούμε στο f την
διαδικασία κατά
'την οποία παίρνουμε το αρχείο
'του συστήματος το οποίο το
'διαλέγουμε από το DirListBox
'που έχουμε σχεδιάσει στη φόρμα μας.
'Το DirListBox μας εμφανίζει μια
'σειρά φακέλων από τους οποίους
'διαλέγουμε τον φάκελο που περιέχει
'το αρχείο που θέλουμε.

Set ts = f.OpenAsTextStream(1, -2) 'εκχωρούμε στο f την διαδικασία ανοίγματος
' του αρχείου μας (που βρίσκεται στο c με το όνομα
'test1.txt) το οποίο διαβάζουμε(1) και ανοίγουμε(-2)
'η εντολή OpenAsTextStream συντάσσεται ως:
'object.OpenAsTextStream([iomode],[format])
'Εμείς στο iomode έχουμε δώσει την τιμή 1 η οποία
'αντιστοιχεί στο άνοιγμα του αρχείου το οποίο μόνο
'διαβάζουμε. Δεν μπορούμε να γράψουμε σε αυτό το
'αρχείο. Στο format έχουμε δώσει την τιμή -2 η οποία
'αντιστοιχεί στο άνοιγμα του αρχείου χρησιμοποιώντας
'το προκαθορισμένο σύστημα.

Set dbs = OpenDatabase("c:\3G.mdb") 'εκχωρούμε στο dbs την διαδικασία
' ανοίγματος της βάσης δεδομένων η οποία βρίσκεται στο c με
' το όνομα 3G.mdb

cellcount = 0 'αρχικοποίηση της μεταβλητής cellcount η οποία ανάλογα με το τι
'αριθμό έχει τόσες φορές θα επαναλάβει την καταχώρηση στην τετράδα
'CH_NO,SCR_CODE,ECN0,BS_STTD

On Error GoTo tei 'ενεργοποίηση της χρήσης μιας διαδικασίας διαχείρισης
σφαλμάτων
'σύνταξη: On Error GoTo γραμμή
'Η δήλωση On Error μπορεί να αναφέρεται σε μια ετικέτα ή αριθμό
'γραμμής(που υποδηλώνει το όρισμα γραμμής)για την εκτροπή της ροής
'του προγράμματος όταν σημειωθεί σφάλμα ενεργοποιώντας τη
' διαδικασία σφάλματος

g = 0 'αρχική τιμή για το g που είναι το πλήθος των γραμμών του ECN0

Do While True 'Κάνε τις παρακάτω εντολές ενώ επαληθεύεται μια συνθήκη(ο όρος
Είναι αληθής)

ReDim server(0) 'επανακαθορισμός μιας ή περισσότερων μεταβλητών δυναμικών
' πινάκων και εκ νέου εκχώρηση χώρου στη μνήμη γι αυτές.Δηλαδή
'επανακαθορισμός
'των μεταβλητών του δυναμικού πίνακα server και εκ νέου εκχώρηση
'χώρου στη μνήμη γι αυτές
'σύνταξη:ReDim όνομα-μεταβλητής (δείκτες)
'δείκτες=διαστάσεις της μεταβλητής του πίνακα

s = ts.readline 'η μεταβλητή readline διαβάζει μια ολόκληρη γραμμή από ένα αρχείο
' και επιστρέφει την προκύπτουσα σειρά
'oyioaic:object.readline
'Όποτε s ονομάζουμε τη μεταβλητή στην οποία εφόσον ανοίγει το αρχείο
'μας θα αρχίσει να το διαβάζει γραμμή γραμμή και να μας εμφανίζει κάθε
'φορά τη γραμμή που διαβάζει

metr = Len(s) 'η συνάρτηση Len()μετράει τους χαρακτήρες της γραμμής και τους
' αποθηκεύει στη μεταβλητή metr
'σύνταξη της Len :Len(συμβολοσειρά|όνομα_μεταβλητής)

metr = metr - 4 'μειώνουμε τους χαρακτήρες της γραμμής κατά 4 έτσι ώστε να μας
'εμφανίζονται στο metr οι 4 πρώτοι χαρακτήρες οι οποίοι αν είναι ECN0
'θα συνεχίσουμε να κάνουμε την ανάλυση της γραμμής διαφορετικά θα
'πάμε στην επόμενη γραμμή.

s2 = Mid(s, 1, 4) 'η συνάρτηση Mid μας επιστρέφει από τους χαρακτήρες της
' γραμμής αρχίζοντας από τον πρώτο τους 4 πρώτους τους οποίους
' καταχωρεί στη μεταβλητή s2
'σύνταξη της Mid:Mid(συμβολοσειρά,αρχή,μήκος)

If s2 <> "ECN0" Then GoTo skpln 'αν η μεταβλητή s2 είναι διάφορη του ECN0 τότε
' πάει στο label skpln και κάνει loop στη συνθήκη
'DoWhile True

g = g + 1 'όταν βρεί την πρώτη γραμμή του ECN0 αρχίζει να μετράει το g τις
' γραμμές του ECN0 που βρίσκει.Στη γραμμή g = g + 1 δηλαδή ουσιαστικά
' για g =1 μπαίνει στη γραμμή του ECN0

variable = "" 'αρχικοποίηση της μεταβλητής variable

k = 0 'αρχικοποίηση μετρητή του πίνακα server

For i = 1 To metr 'μετακινείται ο μετρητής i από το πρώτο ως τον τελευταίο
' χαρακτήρα.

s2 = Mid(s, 4 + i, 1) 'μας επιστρέφει τους 4 + i χαρακτήρες έναν έναν στο s2

If s2 = " " Then 'αν το s2 είναι κενό (δηλ.δεν έχει νούμερο)τότε

flug = True 'η σημαία μας είναι αληθής

If variable <> "" Then 'αν το variable διάφορο του κενού τότε

ReDim Preserve server(UBound(server) + 1) 'ξαναπροσδιορίζουμε τον πίνακα
 'του server δηλ. του αλλάζουμε
 'το μέγεθος με τη λέξη κλειδί
 'ReDim. Επειδή όμως θέλουμε με την
 'αλλαγή μεγέθους τα στοιχεία να
 'μην χάσουν την τιμή τους χρησιμο-
 'ποιούμε τη λέξη κλειδί Preserve
 'UBound: μας επιστρέφει μια τιμή
 'long η οποία αντιπροσωπεύει
 'τον μέγιστο δείκτη ενός πίνακα
 'σύνταξη UBound:
 'UBound(όνομα_πίνακα(διαστάσεις))
 'άρα με τη δήλωση
 'UBound(server) + 1 αυξάνουμε
 'τον δείκτη του server κατά ένα
 'κάθε φορά

server(k) = variable 'ο πίνακας του server αναλόγως την τιμή του k παίρνει
 'την τιμή του variable. Δηλαδή αν k=3 ο 3ος server
 'θα πάρει την τιμή του variable.

variable = ""
 k = k + 1 'αυξάνουμε το k
 End If 'τέλος συνθήκης If που αρχίζει με το αν το variable είναι διάφορο του
 'κενού

Else 'διαφορετικά

If flug = True Then variable = variable & s2 'αν το flug = True δηλαδή είναι
 'κενό τότε το variable θα'ναι ίσο
 'με ότι ήταν στο variable πριν
 'συν την τιμή του s2

End If 'τέλος συνθήκης If που αρχίζει με το αν το s2 είναι κενό

If i = metr And s2 <> " " Then server(k) = variable 'αν το i = metr και το s2
 'διάφορο του κενού τότε το
 'τελευταίο στοιχείο του
 'πίνακα του server θα πάρει
 'την τιμή του variable

Next i 'πάμε στο επόμενο i δηλαδή στην επόμενη μέτρηση χαρακτήρων της
 'επόμενης γραμμής 'ECN0 που θα βρούμε

server(0) = teleiatocomma(server(0)) 'ο server 0 θα είναι ίσος με το server 0 που έχει
 'αντικτασταθεί η τελεία σε κόμμα.

server(1) = teleiatocomma(server(1)) 'ο server 1 θα είναι ίσος με το server 1 που έχει
 'αντικτασταθεί η τελεία σε κόμμα

Set rst = dbs.OpenRecordset("CHARACTERISTICS") 'θέτει στη μεταβλητή
rst(εγγραφής) τη
'διαδικασία ανοίγματος της βάσης και
'συγκεκριμένα ανοίγουμε τον πίνακα με
'το όνομα CHARACTERISTICS στον οποίο
'θα εκχωρήσουμε τις τιμές που εξηγούμε
'παρακάτω.

With rst 'με την rst αρχίζουμε να γράφουμε στον πίνακα CHARACTERISTICS της
βάσης.

If adeia <> 0 Then

.MoveLast

g = adeia + 1

adeia = 0

emptflg = False

End If

.AddNew 'εισάγουμε νέα εγγραφή

!METRHSEIS = g 'στην στήλη με το όνομα METRHSEIS να καταχωρηθεί το
νούμερο i της μεταβλητής g

!HR = server(7) 'στη στήλη με το όνομα HR να καταχωρηθεί το νούμερο του
server 7

!LONGITUDE = server(0) 'στη στήλη με το όνομα LONGITUDE να καταχωρηθεί
το νούμερο του server 0

!LATITUDE = server(1) 'στη στήλη με το όνομα LATITUDE να καταχωρηθεί το
νούμερο του server 1

!Height = server(2) 'στη στήλη με το όνομα Height να καταχωρηθεί το νούμερο
'του server

!DISTANCE = server(3) 'στη στήλη με το όνομα DISTANCE να καταχωρηθεί το
νούμερο του server 3

!GPS_QUALITY = server(4)'στη στήλη με το όνομα GPS_QUALITY να
καταχωρηθεί το νούμερο του server 4

!SATELLITES = server(5) 'στη στήλη με το όνομα SATELLITES να καταχωρηθεί
'το νούμερο του server 5

!VELOCITY = server(6) 'στη στήλη με το όνομα VELOCITY να καταχωρηθεί το
'νούμερο του server 6

```

.Update 'ενημέρωση του πίνακα

.Close 'κλείνει ο πίνακας CHARACTERISTICS
End With

Set rst = dbs.OpenRecordset("SERVER")      'θέτει στη μεταβλητή rst(εγγραφής)
      ' τη διαδικασία ανοίγματος της βάσης και
      'συγκεκριμένα ανοίγουμε τον πίνακα με
      'το όνομα SERVER στον οποίο
      'θα εκχωρήσουμε τις τιμές που εξηγούμε
      'παρακάτω.

cellcount = server(13) 'δίνουμε στη μεταβλητή cellcount την τιμή που έχει το 13ο
      'στοιχείο του πίνακα του server

With rst
  For i = 0 To cellcount - 1 'για i=0 μέχρι το στοιχείο του 13ου server

    If emptflg = False Then
      .MoveLast
    End If

    .AddNew 'εισάγουμε νέα εγγραφή

    !METRHSEIS = g 'στη στήλη με το όνομα METRHSEIS να καταχωρηθεί
      'το νούμερο της μεταβλητής g

    !HR = server(7) 'στη στήλη με το όνομα HR να καταχωρηθεί
      'το νούμερο του server 7

    !WCDMA = server(8) 'στη στήλη με το όνομα WCDMA να καταχωρηθεί
      'το νούμερο του server 8

    !NO_C = server(9) 'στη στήλη με το όνομα NO_C να καταχωρηθεί
      'το νούμερο του server 9

    !CH_NO_C = server(10) 'στη στήλη με το όνομα CH_NO_C να καταχωρηθεί
      'το νούμερο του server 10

    !C_RSSI = server(11) 'στη στήλη με το όνομα C_RSSI να καταχωρηθεί
      'το νούμερο του server 11

    !NO_PAR = server(12) 'στη στήλη με το όνομα NO_PAR να καταχωρηθεί
      'το νούμερο του server 12

    !NO_OF_CELLS_ACT_SET = server(13) 'στη στήλη με το όνομα
      NO_OF_CELLS_ACT_SET
      'να καταχωρηθεί το νούμερο του server 13

    !CH_NO = server(14 + i * 4) 'η στήλη με το όνομα CH_NO παίρνει την τιμή του

```

' server 14 συν την τιμή του $i * 4$. Δηλαδή αν το
' $i=0$ παίρνει την τιμή του 14ου server ενώ αν
' $i=1$ παίρνει την τιμή του 18ου server.

!SCR_CODE = server(15 + $i * 4$) 'η στήλη με το όνομα SCR_CODE παίρνει την
' τιμή του server 15 συν την τιμή του $i * 4$. Δηλαδή αν το
' $i=0$ παίρνει την τιμή του 15ου server ενώ αν
' $i=1$ παίρνει την τιμή του 19ου server.

!ECN0 = server(16 + $i * 4$) 'η στήλη με το όνομα ECN0 παίρνει την τιμή του
' server 16 συν την τιμή του $i * 4$. Δηλαδή αν το
' $i=0$ παίρνει την τιμή του 16ου server ενώ αν
' $i=1$ παίρνει την τιμή του 20ου server.

!BS_STTD = server(17 + $i * 4$) 'η στήλη με το όνομα BS_STTD παίρνει την
' τιμή του server 17 συν την τιμή του $i * 4$. Δηλαδή αν το
' $i=0$ παίρνει την τιμή του 17ου server ενώ αν
' $i=1$ παίρνει την τιμή του 21ου server.

!RSCP = !C_RSSI + !ECN0 'προσθέτουμε το νούμερο της στήλης C_RSSI με
' το νούμερο της στήλης ECN0 και το αποτέλεσμα της
' πρόσθεσης το καταχωρούμε στη στήλη με το όνομα RSCP

.Update 'ενημέρωση του πίνακα

Next i 'πάμε στο επόμενο i

.Close 'κλείνει ο πίνακας SERVER

End With

'μέχρι το End If γίνεται η συγκέντρωση όλων των RSCP σ'ένα δυναμικό πίνακα της Visual Basic που ονομάζεται bestserver και από τον οποίο μετά θα διαλέξουμε τα 5 καλύτερα που θέλουμε.

Dim best1, best2, bestserver(), koryfh1, trith1 As Integer

ReDim bestserver(0)

best1 = Val(server(11)) + Val(server(16))

If server(13) <> 1 Then

For i = 0 To cellcount - 1

best2 = Val(server(11)) + Val(server(16 + $i * 4$))

bestserver(i) = best2

ReDim Preserve bestserver(UBound(bestserver) + 1)

bestserver(UBound(bestserver)) = 0

If best1 < best2 Then

best1 = best2

End If

Next i

koryfh1 = UBound(bestserver) - 1

Do While True


```

For i = 0 To koryfh1
  If bestserver(i) > bestserver(i + 1) Then
    trith1 = bestserver(i)
    bestserver(i) = bestserver(i + 1)
    bestserver(i + 1) = trith1
  End If
Next i
koryfh1 = koryfh1 - 1
If koryfh1 <= 0 Then Exit Do
Loop
End If

```

num = 18 + (cellcount - 1) * 4 'ορίζουμε την μεταβλητή num ίση με την τιμή του
'server 18 συν την τιμή που έχει το i (δηλ. το cellcount - 1)
'επί την τετράδα που θα συμπληρώσει. Δηλ. αν το i=0 num=18.

cellcount = server(num) 'το cellcount ορίζεται να είναι ίσο με την τιμή του
server(num)
'δηλ.αν num=18 το cellcount παίρνει την τιμή του server 18

Set rst = dbs.OpenRecordset("NEIGHBOURS_MONITORED") 'θέτει στη μεταβλητή
rst(εγγραφής) τη
'διαδικασία ανοίγματος της βάσης και
'υγκεκριμένα ανοίγουμε τον πίνακα με
'το όνομα NEIGHBOURS_MONITORED στον οποίο
'θα εκχωρήσουμε τις τιμές που εξηγούμε
'παρακάτω

ReDim bestmon(0)

With rst

For i = 0 To cellcount - 1 'μέχρι το στοιχείο του 18ου server

If emptflg = False Then

.MoveLast

End If

.AddNew

!METRHSEIS = g 'στη στήλη με το όνομα METRHSEIS να καταχωρηθεί
'το νούμερο της μεταβλητής g

!HR = server(7) 'στη στήλη με το όνομα HR να καταχωρηθεί
'το νούμερο του server 7

!C_RSSI = server(11) 'στη στήλη με το όνομα C_RSSI να καταχωρηθεί
'το νούμερο του server 11

!NO_CELLS_MON_SET = server(num) 'στη στήλη με το όνομα
NO_CELLS_MON_SET
'να καταχωρηθεί το νούμερο του server num

```

!CH_NO = server(1 + num + i * 4) 'στη στήλη με το όνομα CH_NO
        'να καταχωρηθεί το νούμερο του server num
        'αυξάνοντας τον κατά ένα και προσθέτοντας το
        'αποτέλεσμα της πράξης i * 4

!SCR_CODE = server(2 + num + i * 4) 'στη στήλη με το όνομα SCR_CODE
        'να καταχωρηθεί το νούμερο του server num
        'αυξάνοντας τον κατά δυο και προσθέτοντας το
        'αποτέλεσμα της πράξης i * 4

!ECN0 = server(3 + num + i * 4) 'στη στήλη με το όνομα ECN0
        'να καταχωρηθεί το νούμερο του server num
        'αυξάνοντας τον κατά τρία και προσθέτοντας το
        'αποτέλεσμα της πράξης i * 4

!BS_STTD = server(4 + num + i * 4) 'στη στήλη με το όνομα BS_STTD
        'να καταχωρηθεί το νούμερο του server num
        'αυξάνοντας τον κατά τέσσερα και
        'προσθέτοντας το αποτέλεσμα της πράξης
        'i * 4

!RSCP = !C_RSSI + !ECN0 'προσθέτουμε το νούμερο της στήλης C_RSSI με
        ' το νούμερο της στήλης ECN0 και το αποτέλεσμα της
        'πρόσθεσης το καταχωρούμε στη στήλη με το όνομα RSCP

.Update 'ενημέρωση του πίνακα
    
```

'μέχρι το .Close γίνεται έλεγχος για το καλύτερο RSCP του πίνακα

```

NEIGHBOURS_MONITORED
Dim ii As Integer
Dim denyparxei As Boolean
denyparxei = False
For ii = 0 To UBound(bestmon)
    If bestmon(ii) <> (Val(server(11)) + Val(server(3 + num + i * 4))) Then
        denyparxei = True
    Else
        denyparxei = False
        GoTo exwif
    End If
Next ii
exwif:
If denyparxei = True Then
    bestmon(UBound(bestmon)) = Val(server(11)) + Val(server(3 + num + i * 4))
    ReDim Preserve bestmon(UBound(bestmon) + 1)
End If

Next i '?Uia ooi a?uiaii i
    
```

```
Dim lasti As Integer
lasti = UBound(bestmon)
```

```
.Close 'κλείνει ο πίνακας NEIGHBOURS_MONITORED
```

```
End With
```

```
num = 5 + num + (cellcount - 1) * 4 'ορίζουμε την μεταβλητή num ίση με την τιμή
' της προηγούμενης num συν την τιμή 5 συν την τιμή
' που έχει το i (δηλ.τοcellcount-1 )
' επί την τετράδα που θα συμπληρώσει.Δηλ. αν το i=0 num=23.
```

```
cellcount = server(num) 'το cellcount ορίζεται να είναι ίσο με την τιμή του
server(num)
'δηλ.αν cell=23 το cellcount παίρνει την τιμή του server 23
```

```
Set rst = dbs.OpenRecordset("NEIGHBOURS_DETECTED") 'θέτει στη μεταβλητή
rst(εγγραφής) τη
' διαδικασία ανοίγματος της βάσης και
' συγκεκριμένα ανοίγουμε τον πίνακα με
' το όνομα NEIGHBOURS_DETECTED στον οποίο
' θα εκχωρήσουμε τις τιμές που εξηγούμε
' παρακάτω
```

```
With rst
```

```
For i = 0 To cellcount - 1
If emptflg = False Then
.MoveLast
End If
.AddNew
!METRHSEIS = g 'στη στήλη με το όνομα METRHSEIS να καταχωρηθεί
'το νούμερο της μεταβλητής g
```

```
!HR = server(7) 'στη στήλη με το όνομα HR να καταχωρηθεί
'το νούμερο του server 7
```

```
!C_RSSI = server(11) 'στη στήλη με το όνομα C_RSSI να καταχωρηθεί
'το νούμερο του server 11
```

```
!NO_CELLS_DET_SET = server(num) 'στη στήλη με το όνομα
NO_CELLS_DET_SET
'να καταχωρηθεί το νούμερο του server num
```

```
!CH_NO = server(1 + num + i * 4) 'στη στήλη με το όνομα CH_NO
'να καταχωρηθεί το νούμερο του server num
'αυξάνοντας τον κατά ένα και προσθέτοντας το
'αποτέλεσμα της πράξης i * 4
```

```
!SCR_CODE = server(2 + num + i * 4) 'στη στήλη με το όνομα SCR_CODE
'να καταχωρηθεί το νούμερο του server num
'αυξάνοντας τον κατά δυο και προσθέτοντας το
'αποτέλεσμα της πράξης i * 4
```

```
!ECN0 = server(3 + num + i * 4) 'στη στήλη με το όνομα ECN0
'να καταχωρηθεί το νούμερο του server num
'αυξάνοντας τον κατά τρία και προσθέτοντας το
'αποτέλεσμα της πράξης i * 4
```

```
!BS_STTD = server(4 + num + i * 4) 'στη στήλη με το όνομα BS_STTD
'να καταχωρηθεί το νούμερο του server num
'αυξάνοντας τον κατά τέσσερα και προσθέτοντας το
'αποτέλεσμα της πράξης i * 4
```

```
!RSCP = !C_RSSI + !ECN0 'προσθέτουμε το νούμερο της στήλης C_RSSI με
' το νούμερο της στήλης ECN0 και το αποτέλεσμα της
'πρόσθεσης το καταχωρούμε στη στήλη με το όνομα RSCP
```

```
.Update 'ενημέρωση του πίνακα
```

'μέχρι το .Close γίνεται έλεγχος για το καλύτερο RSCP του πίνακα
'NEIGHBOURS_DETECTED και τα καταχωρούμε στον δυναμικό πίνακα
'του bestserver της Visual Basic

```
For ii = 0 To UBound(bestmon)
denyparxei = False
If bestmon(ii) <> (Val(server(11)) + Val(server(3 + num + i * 4))) Then
denyparxei = True
Else
denyparxei = False
GoTo exwif2
End If
Next ii
```

exwif2:

```
If denyparxei = True Then
bestmon(UBound(bestmon)) = Val(server(11)) + Val(server(3 + num + i * 4))
ReDim Preserve bestmon(UBound(bestmon) + 1)
End If
```

```
Next i '?Uia ooi a?uiaii i
```

```
.Close 'eea?iae i ??iaeo NEIGHBOURS_DETECTED
```

```
End With
```

'μέχρι το .End with γίνεται υπολογισμός των καλύτερων RSCP και τα καταχωρούμε
στον πίνακα της βάσης BEST_SERVERS που έχουμε φτιάξει.

```
If bestserver(0) <> 0 Then
ReDim Preserve bestmon(UBound(bestmon) + UBound(bestserver) - 2)
For i = 0 To (UBound(bestserver) - 2)
bestmon(i + UBound(bestmon) + 2 - UBound(bestserver)) =
bestserver(UBound(bestserver) - 2 - i)
```

```

Next i
End If
Dim best() As Integer
Dim koryfh, trith As Integer
Select Case UBound(bestmon)
Case Is <= 4
    ReDim best(4)
    For i = 0 To UBound(bestmon)
        best(i) = bestmon(i)
    Next i
    For i = UBound(bestmon) + 1 To 3
        best(i) = 0
    Next i
    koryfh = UBound(best) - 1
    Do While True

        For i = 0 To koryfh
            If best(i) < best(i + 1) And best(i + 1) <> 0 Then
                trith = best(i)
                best(i) = best(i + 1)
                best(i + 1) = trith
            End If
        Next i
        koryfh = koryfh - 1
        If koryfh <= 0 Then Exit Do
    Loop
Case Else
    ReDim best(4)

    koryfh = UBound(bestmon) - 1
    Do While True

        For i = 0 To koryfh
            If bestmon(i) > bestmon(i + 1) Then
                trith = bestmon(i)
                bestmon(i) = bestmon(i + 1)
                bestmon(i + 1) = trith
            End If
        Next i
        koryfh = koryfh - 1
        If koryfh <= 0 Then Exit Do
    Loop

    For i = 0 To 3
        best(i) = bestmon(UBound(bestmon) - i)
    Next i

End Select

Set rst = dbs.OpenRecordset("BEST_SERVERS")

```

With rst

```
.AddNew
!metrhseis = g
!bestserver = best1
!BESTNEIGH1 = best(0)
!BESTNEIGH2 = best(1)
!BESTNEIGH3 = best(2)
!BESTNEIGH4 = best(3)
```

.Update

.Close

End With

num = 5 + num + (cellcount - 1) * 4 'ορίζουμε την μεταβλητή num ίση με την τιμή τ
'τηςπροηγούμενης num συν την τιμή 5 συν την τιμή
'που έχει το i (δηλ. tocellcount-1)
'επί την τετράδα που θα συμπληρώσει. Δηλ. αν το i=0 num=28.

cellcount = server(num)

Set rst = dbs.OpenRecordset("NEIGHBOURS_UNDETECTED") 'θέτει στη
μεταβλητή rst(εγγραφής) τη
'διαδικασία ανοίγματος της βάσης και
'συγκεκριμένα ανοίγουμε τον πίνακα με
'το όνομα NEIGHBOURS_UNDETECTED στον οποίο
'θα εκχωρήσουμε τις τιμές που εξηγούμε
'παρακάτω

With rst

For i = 0 To cellcount - 1

If emptflg = False Then

.MoveLast

End If

.AddNew

!METRHSEIS = g 'στη στήλη με το όνομα METRHSEIS να καταχωρηθεί
'το νούμερο της μεταβλητής g

!HR = server(7) 'στη στήλη με το όνομα HR να καταχωρηθεί
'το νούμερο του server 7

!C_RSSI = server(11) 'στη στήλη με το όνομα C_RSSI να καταχωρηθεί
'το νούμερο του server 11

!NO_CELLS_UND = server(num) 'στη στήλη με το όνομα NO_CELLS_UND
'να καταχωρηθεί το νούμερο του server num

!CH_NO = server(1 + num + i * 4) 'στη στήλη με το όνομα CH_NO
'να καταχωρηθεί το νούμερο του server num
'αυξάνοντας τον κατά ένα και προσθέτοντας το
'αποτέλεσμα της πράξης i * 4

!SCR_CODE = server(2 + num + i * 4) 'στη στήλη με το όνομα SCR_CODE

'να καταχωρηθεί το νούμερο του server num
'αυξάνοντας τον κατά δυο και προσθέτοντας το
'αποτέλεσμα της πράξης $i * 4$

!ECN0 = server(3 + num + $i * 4$) 'στη στήλη με το όνομα ECN0
'να καταχωρηθεί το νούμερο του server num
'αυξάνοντας τον κατά τρια και προσθέτοντας το
'αποτέλεσμα της πράξης $i * 4$

!BS_STTD = server(4 + num + $i * 4$) 'στη στήλη με το όνομα BS_STTD
'να καταχωρηθεί το νούμερο του server num
'αυξάνοντας τον κατά τέσσερα και προσθέτοντας το
'αποτέλεσμα της πράξης $i * 4$

!RSCP = !C_RSSI + !ECN0 'προσθέτουμε το νούμερο της στήλης C_RSSI με
' το νούμερο της στήλης ECN0 και το αποτέλεσμα της
'πρόσθεσης το καταχωρούμε στη στήλη με το όνομα RSCP

.Update 'ενημέρωση του πίνακα

Next i 'πάμε στο επόμενο i

.Close 'κλείνει ο πίνακας NEIGHBOURS_UNDETECTED

End With

skpln: 'το label skpln κάνει Loop

Loop

home: 'το label home

'μέχρι το End With βρίσκουμε τα scrambling code του πίνακα SERVER

Dim scrcode(), metrhs(), longit(), lat(), rscp2() As Integer

ReDim Preserve scrcode(0)

ReDim metrhs(0)

ReDim longit(0)

ReDim lat(0)

ReDim rscp2(0)

Set rst = dbs.OpenRecordset("SERVER")

i = 0

With rst

.MoveFirst

Do While True

scrcode(i) = !SCR_CODE

metrhs(i) = !metrhseis

rscp2(i) = !RSCP

ReDim Preserve scrcode(UBound(scrcode) + 1)

ReDim Preserve metrhs(UBound(metrhs) + 1)

```
ReDim Preserve longit(UBound(longit) + 1)
ReDim Preserve lat(UBound(lat) + 1)
ReDim Preserve rscp2(UBound(rscp2) + 1)
```

```
i = i + 1
```

```
.MoveNext
```

```
If .EOF Then GoTo Label1
```

```
Loop
```

```
Label1:
```

```
.Close
```

```
End With
```

'μέχρι το End With βρίσκουμε τα scrambling code του πίνακα
'NEIGHBOURS_MONITORED

```
Set rst = dbs.OpenRecordset("NEIGHBOURS_MONITORED")
```

```
With rst
```

```
.MoveFirst
```

```
Do While True
```

```
scrcode(i) = !SCR_CODE
```

```
metrhs(i) = !metrhseis
```

```
rscp2(i) = !RSCP
```

```
ReDim Preserve scrcode(UBound(scrcode) + 1)
```

```
ReDim Preserve metrhs(UBound(metrhs) + 1)
```

```
ReDim Preserve longit(UBound(longit) + 1)
```

```
ReDim Preserve lat(UBound(lat) + 1)
```

```
ReDim Preserve rscp2(UBound(rscp2) + 1)
```

```
i = i + 1
```

```
.MoveNext
```

```
If .EOF Then GoTo label2
```

```
Loop
```

```
label2:
```

```
.Close
```

```
End With
```

'μέχρι το End With βρίσκουμε τα scrambling code του πίνακα
'NEIGHBOURS_DETECTED

```
Set rst = dbs.OpenRecordset("NEIGHBOURS_DETECTED")
```

```
With rst
```

```
.MoveFirst
```

```
Do While True
```

```
scrcode(i) = !SCR_CODE
```

```
metrhs(i) = !metrhseis
```



```

rscp2(i) = !RSCP
ReDim Preserve scrcode(UBound(scrcode) + 1)
ReDim Preserve metrhs(UBound(metrhs) + 1)
ReDim Preserve longit(UBound(longit) + 1)
ReDim Preserve lat(UBound(lat) + 1)
ReDim Preserve rscp2(UBound(rscp2) + 1)
i = i + 1
.MoveNext

If .EOF Then GoTo label3
Loop
label3:
.Close
End With

'καταχωρούμε τα scrambling code στον πίνακα SCRAMBLING_CODES
Set rst = dbs.OpenRecordset("SCRAMBLING_CODES")
With rst
For i = 0 To UBound(scrcode) - 1
.AddNew
    !metrhseis = metrhs(i)
    !RSCP = rscp2(i)
    !SCRAMBLING_CODE = scrcode(i)

.Update
Next i

.Close

End With

Dim numofscrcode() As Integer
ReDim numofscrcode(0)
Dim plscr As Integer
Dim newscr As Boolean
plscr = 0
newscr = False
For i = 0 To UBound(scrcode)

For plscr = 0 To UBound(numofscrcode)
    If numofscrcode(plscr) <> scrcode(i) Then
        newscr = True

    Else
        newscr = False
        Exit For
    End If
Next plscr
If newscr = True Then
    numofscrcode(UBound(numofscrcode)) = scrcode(i)
    ReDim Preserve numofscrcode(UBound(numofscrcode) + 1)

```

```

    End If
Next i
For i = 0 To UBound(numofscrcode) - 1
    List1.AddItem numofscrcode(i)
Next i
Label1.Caption = " Scrambling code: " & UBound(numofscrcode)
' μας εμφανίζει όταν τελειώσει
'το τρέξιμο του αρχείου σε νομερο πόσα SCRAMBLING_CODES βρήκε στο αρχείο
που τρέξαμε

dbs.Close 'κλείνει τη βάση

ts.Close 'κλείνει το αρχείο

MsgBox g 'όταν τελειώσει το τρέξιμο του προγράμματος μας εμφανίζει πόσες
γραμμές ECN0 βρέθηκαν.

Exit Sub 'τέλος διαδικασίας

tei: 'to label tei

If Error = Error(62) Then 'αν το λάθος είναι το 62(που είναι ότι το αρχείο έχει
'τελειώσει ) να πάει στο label home να κλείσει τη βάση
'και το αρχείο

    GoTo home

Else 'αλλιώς

MsgBox Error 'να μας εμφανίσει το μήνυμα λάθους

End If
End Sub

Private Sub Drive1_Change()
    Dir1.Path = Drive1.Drive 'διαλέγουμε από το DriveListBox τη μονάδα του δίσκου
    Που θα χρησιμοποιήσουμε

End Sub
Private Sub Dir1_Change()
    File1.Path = Dir1.Path 'διαλέγουμε από το FileListBox το αρχείο που
    'θα χρησιμοποιήσουμε
End Sub

Private Function teleiatocomma(str As String) As String 'δήλωση της μεταβλητής
    teleiatocomma σαν String
Dim s_start, s_end, str2 As String 'δήλωση των μεταβλητών s_start, s_end, str2 σαν
String οι οποίες κάνουν τα εξής:
s_start->καταχωρούνται οι τιμές που είναι πριν την τελεία
s_end -> καταχωρούνται οι τιμές που είναι μετά την τελεία

```

ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)

str2 -> διαβάζει έναν έναν τους χαρακτήρες.

Dim i As Integer 'δήλωση της μεταβλητής σαν Integer

For i = 1 To Len(str) - 1 'για i=1 μέχρι να μετρήσει όλους τους χαρακτήρες

str2= Mid(str, i, 1) 'η συνάρτηση Mid μας επιστρέφει από τους χαρακτήρες του str
'αρχίζοντας από τον i έναν έναν τους οποίους καταχωρεί
'στη μεταβλητή str2

If str2 = "." Then 'αν η μεταβλητή str2 είναι τελεία τότε

str = s_start & "," & Mid(s_end, 2, Len(s_end) - 1) 'η μεταβλητή str θα είναι
ίση με την αρχή του string
και το κόμμα και τους υπόλοιπους
χαρακτήρες μέχρι το τέλος
αρχίζοντας από το δεύτερο χαρακτήρα.

Exit For 'έξοδος από την for
End If 'τέλος της συνθήκης if

s_start = Mid(str, 1, i) 'η μεταβλητή s_start είναι ίση με
s_end = Mid(str, i + 1, Len(str) - Len(s_start))

Next i

teleiatocomma = str

End Function

'πατώντας το κουμπί ClearDatabase καθαρίζονται οι παρακάτω πίνακες της βάσης

Private Sub Command2_Click()

Call cleardatabase("c:\3G.mdb", "CHARACTERISTICS")

Call cleardatabase("c:\3G.mdb", "NEIGHBOURS_DETECTED")

Call cleardatabase("c:\3G.mdb", "NEIGHBOURS_MONITORED")

Call cleardatabase("c:\3G.mdb", "SERVER")

Call cleardatabase("c:\3G.mdb", "BEST_SERVERS")

Call cleardatabase("c:\3G.mdb", "SCRAMBLING_CODES")

End Sub

Private Function cleardatabase(dbsloc As String, rstname As String) 'δήλωση της
συνάρτησης που σβήνει τις εγγραφές της βάσης όπου με το dbsloc
δηλώνουμε την τοποθεσία που βρίσκεται η βάση και με rstname το
όνομα του πίνακα της βάσης.

Dim dbs As Database

ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)

```
Dim rst As Recordset
Set dbs = OpenDatabase(dbsloc)

Set rst = dbs.OpenRecordset(rstname)
With rst
    .MoveFirst
    Do While True

        .Delete

        .MoveNext

    If .EOF Then Exit Do

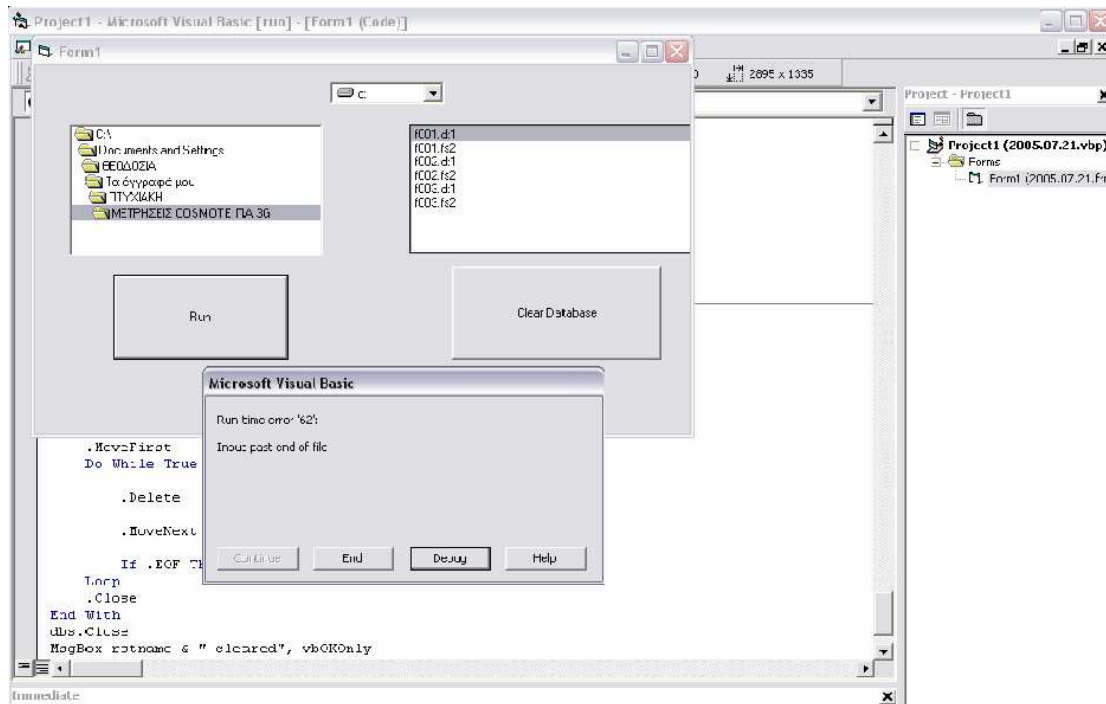
Loop
.Close
End With
dbs.Close
MsgBox rstname & " cleared", vbOKOnly 'όταν τελειώσει το σβήσιμο να εμφανιστεί
                                     μήνυμα με το όνομα του πίνακα που καθαρίστηκε

End Function

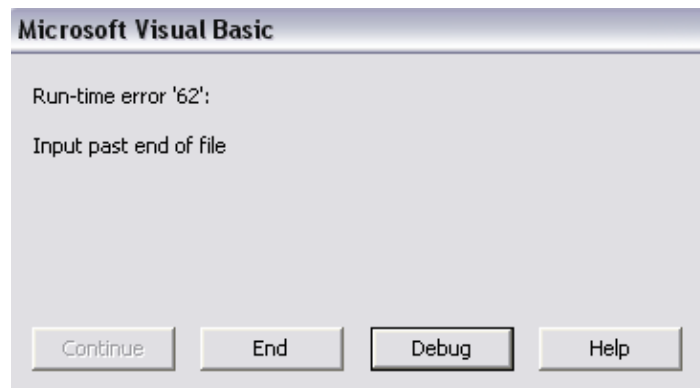
Private Sub Form_Load()
Call strup
Dir1.Path = "c:\"
End Sub
```

Διευκρίνιση διαφόρων σημείων του κώδικα μας

1. Το Error 62 σημαίνει ότι το αρχείο έχει τελειώσει και δεν υπάρχει άλλη γραμμή για να διαβάσουμε. Αν λοιπόν τρέξουμε τον κώδικά μας χωρίς τη συνθήκη **If Error=Error (62)** τότε μπορούμε να δούμε το μήνυμα λάθους που μας εμφανίζεται και μας ειδοποιεί για το τέλος του αρχείου.

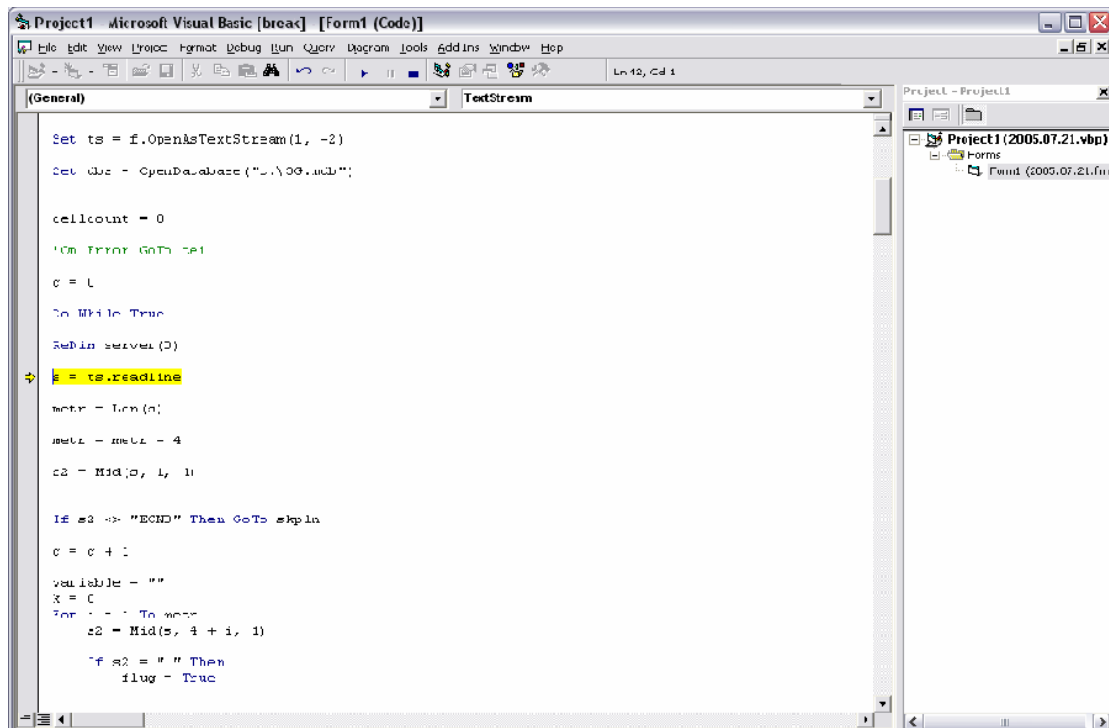


ΕΙΚΟΝΑ 36



ΕΙΚΟΝΑ 37

Αν πατήσουμε στο κουμπί Debug θα μας δείξει με κίτρινο βελάκι ολόκληρη τη γραμμή όπου υπάρχει το πρόβλημα




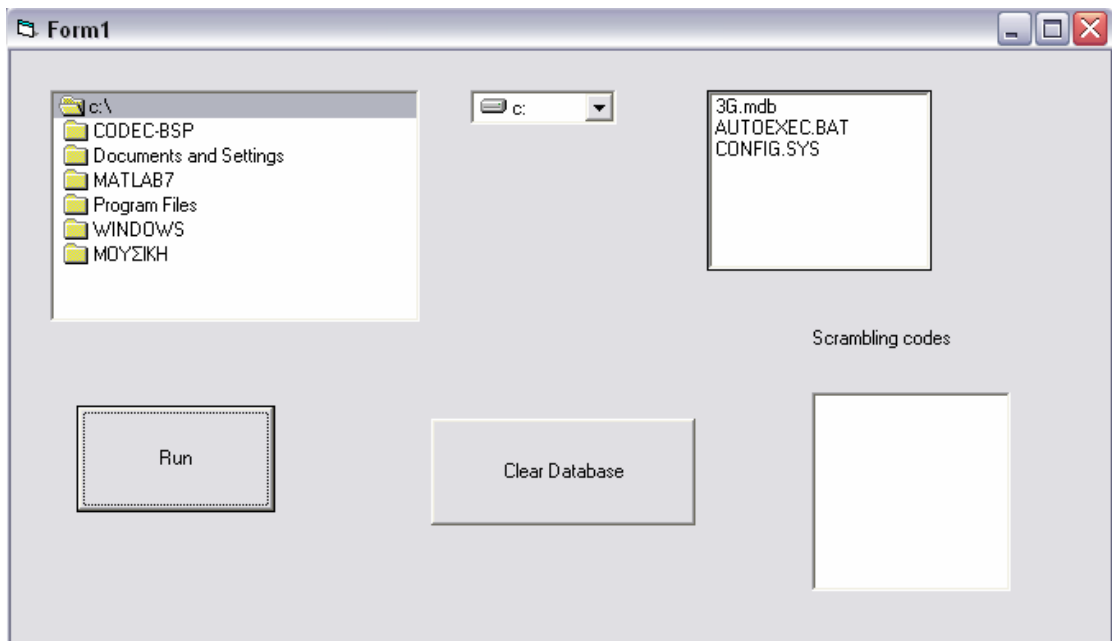
ΕΙΚΟΝΑ 38

Βλέπουμε λοιπόν ότι το λάθος εμφανίζεται στη μεταβλητή S η οποία διαβάζει τη γραμμή και αυτό γιατί δεν υπάρχει άλλη γραμμή για διάβασμα. Το Error 3021 δηλώνει ότι δεν υπάρχει εγγραφή στον πίνακα της βάσης που έχει ανοιχτεί. Με θαυμαστικό (!) δηλώνουμε εγγραφή στη βάση. Για παράδειγμα !METRHSEIS = g σημαίνει να γραφτεί στη στήλη με το όνομα METRHSEIS το νούμερο της γραμμής που έχει μετρηθεί.

Τρέξιμο του προγράμματος ολοκληρωμένα αλλά και βήμα βήμα

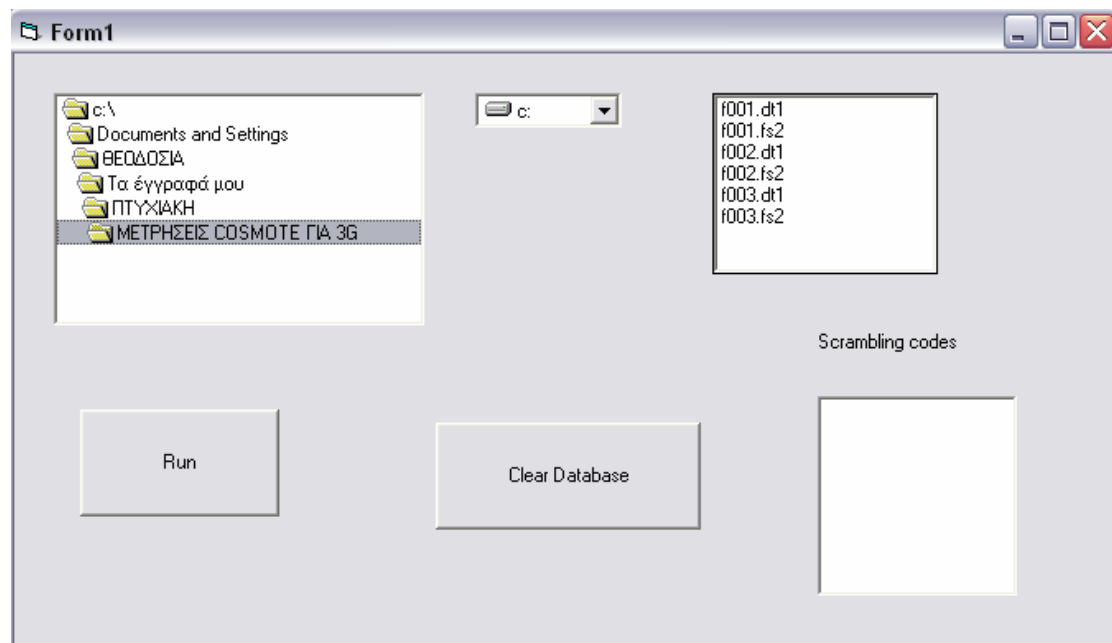
Ολοκληρωμένο τρέξιμο

- Τρέχουμε τον κώδικά μας με **Run→Start** ή πατώντας το κουμπί  από την γραμμή εργαλείων Debug ή πατώντας **F5**.
- Εμφανίζεται η Form :



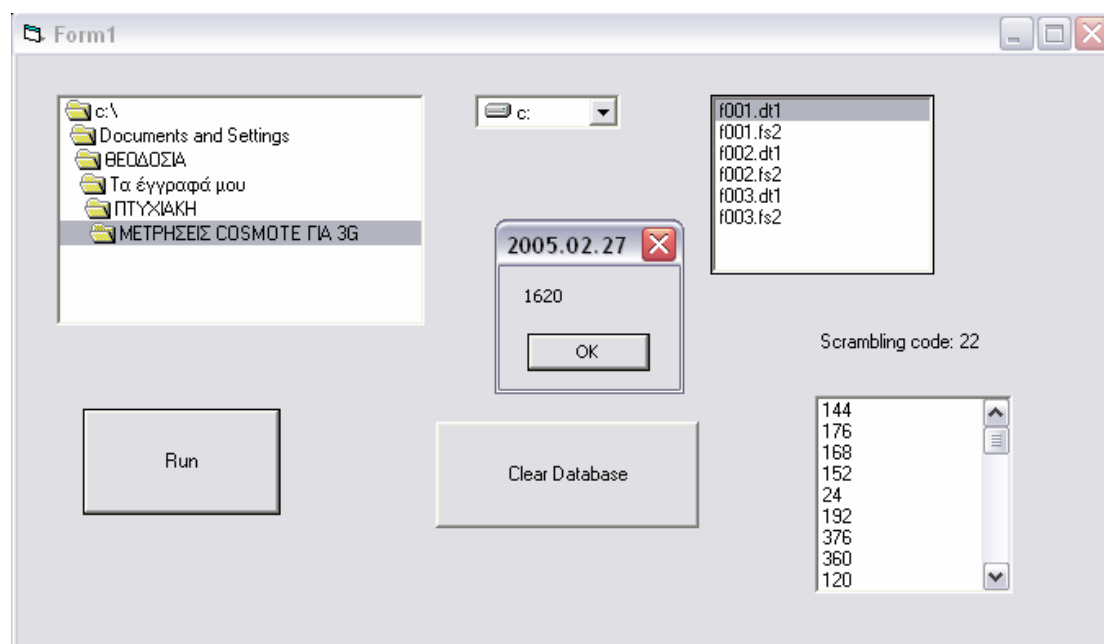
ΕΙΚΟΝΑ 39

Από το **DirListBox** επιλέγουμε το φάκελο **ΠΤΥΧΙΑΚΗ** με διπλό κλικ
 → επιλέγουμε με διπλό κλικ το φάκελο **ΜΕΤΡΗΣΕΙΣ COSMOTE** → επιλέγουμε
 από το **FileListBox** το αρχείο **f001.dt1** → πατάμε το κουμπί **Run** και τρέχει.



ΕΙΚΟΝΑ 40

Όταν τελειώσει το τρέξιμο θα μας εμφανίσει μήνυμα με το πόσες γραμμές ECNO βρήκε στο αρχείο μας, πόσα scrambling code και τον αριθμό αυτών.



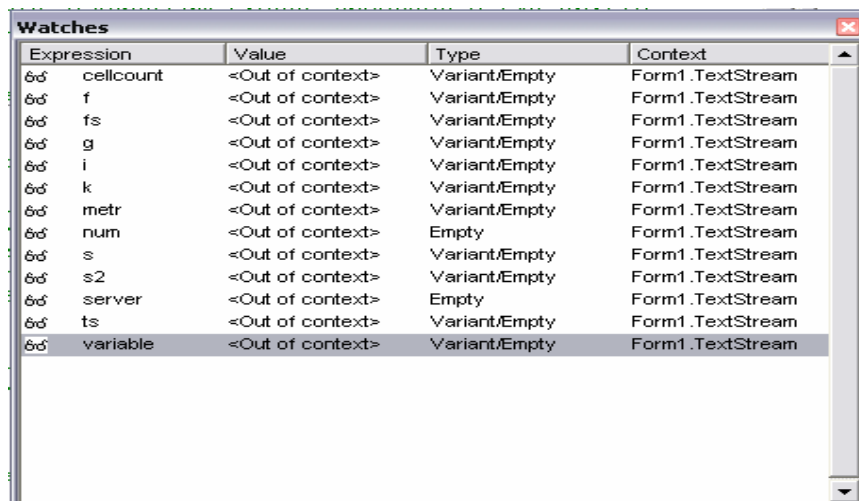
ΕΙΚΟΝΑ 41

Θα πατήσουμε **OK** και θα επιστρέψει στη Form της εικόνας 40 στη συνέχεια **επιλέγοντας απλά το αρχείο f002.dt1** →πατάμε **Run** και τρέχει για το 2^ο αρχείο. Θα εμφανίσει τελειώνοντας το τρέξιμο το μήνυμα των γραμμών όπως παραπάνω και ο αριθμός των scrambling code. Τέλος θα επιλέξουμε να τρέξει και για το αρχείο f003.dt1 με την ίδια διαδικασία που περιγράψαμε για τα άλλα δυο αρχεία. Πρέπει να σημειώσουμε σε αυτό το σημείο ότι τρέχουμε μόνο τα αρχεία με προέκταση .dt γιατί σε αυτά μόνο υπάρχουν οι γραμμές του ECNO που εξετάζουμε.

- Ανοίγουμε την βάση μας και βλέπουμε ότι έχουν συμπληρωθεί οι πίνακες μας σωστά.
- Αν τώρα θάλουμε να καθαρίσουμε τη βάση από τις εγγραφές τότε πατάμε το κουμπί **Clear Database** και εμφανίζεται κάθε φορά σε Msg Box ποιος πίνακας έχει καθαριστεί.

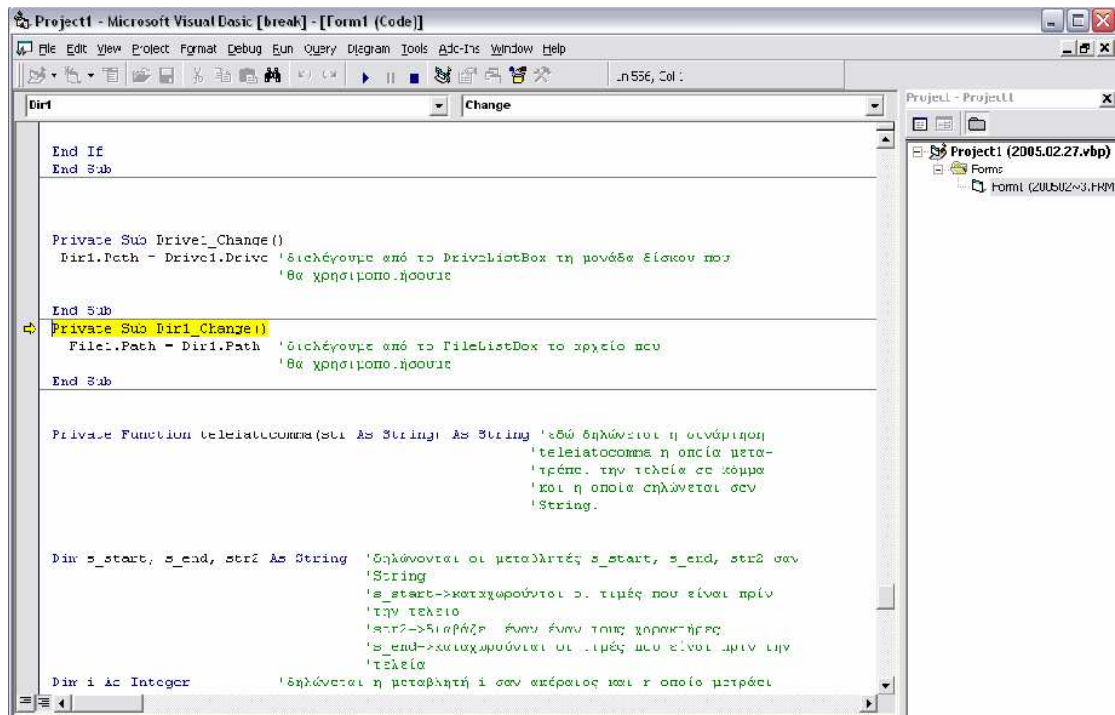
Τρέξιμο αρχείων βήμα βήμα.

- Το τρέξιμο κατά βήμα μας εξυπηρετεί στο να παρακολουθούμε τις τιμές που παίρνουν οι μεταβλητές μας καθώς προχωράμε γραμμή γραμμή το πρόγραμμα μας. Αρχικά εισάγουμε τις μεταβλητές μας στο watch window ως εξής: **μαρκάρουμε την μεταβλητή που θέλουμε→δεξί κλικ→Add Watch** →**βλέπουμε την μεταβλητή μας στο Expression** →**OK**. Στην εικόνα 42 φαίνονται οι μεταβλητές που έχουμε βάλει για παρακολούθηση. Όταν όμως αρχίσουμε να τρέχουμε το πρόγραμμα μας με το F8 το watch Window δεν εμφανίζεται στο κάτω μέρος της οθόνης μας και πάμε **View**→**Watch Window** και εμφανίζεται.



ΕΙΚΟΝΑ 42

- Στη συνέχεια πατάμε **F8** → διπλό κλικ στο φάκελο **ΠΤΥΧΙΑΚΗ** που είναι στο **My Document** → μας πετάει στο κώδικα μας στο σημείο Private Sub Dir1_Change()



ΕΙΚΟΝΑ 43

- Πατάμε **F8** ώσπου να φτάσει στο End Sub →πάμε στη Form και επιλέγουμε από το **DirListBox** το φάκελο **ΜΕΤΡΗΣΕΙΣ COSMOTE** με διπλό κλικ → μας πετάει ξανά στο σημείο Private Sub Dir1_Change()
- Πατάμε **F8** ώσπου να φτάσει στο End Sub →πάμε στη Form και επιλέγουμε από το **FileListBox** το **f001.dt1** →πατάμε το Run και στη συνέχεια με **F8** τρέχουμε τον κώδικα μας βήμα βήμα και βλέπουμε τις τιμές που παίρνουν οι μεταβλητές μας.

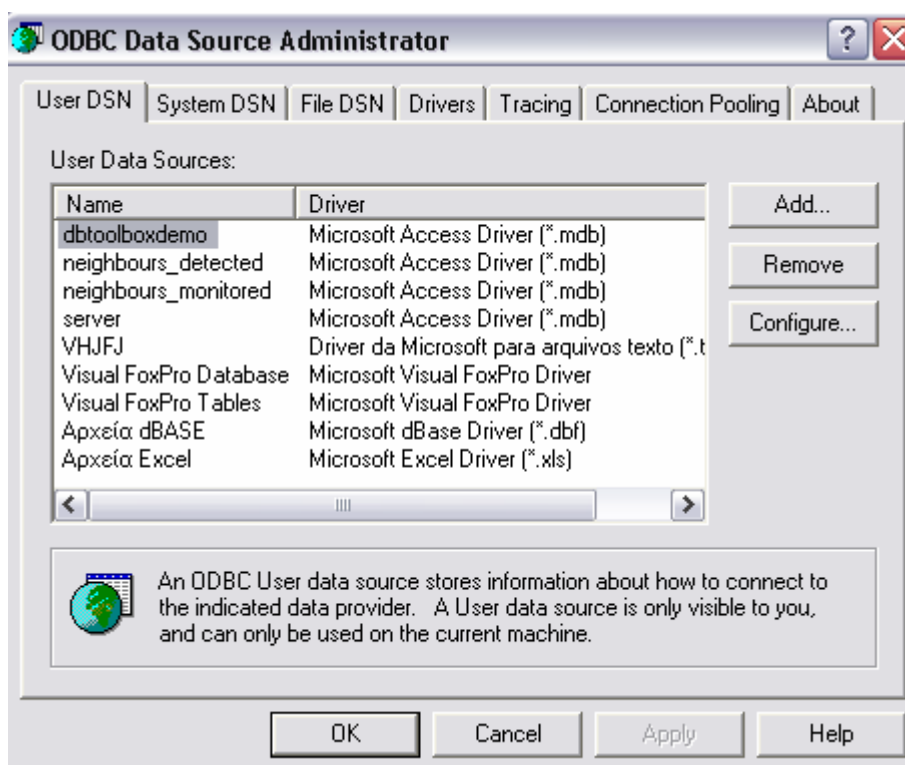
MATLAB 7

ΒΗΜΑΤΑ ΠΡΙΝ ΤΗΝ ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΩΝ

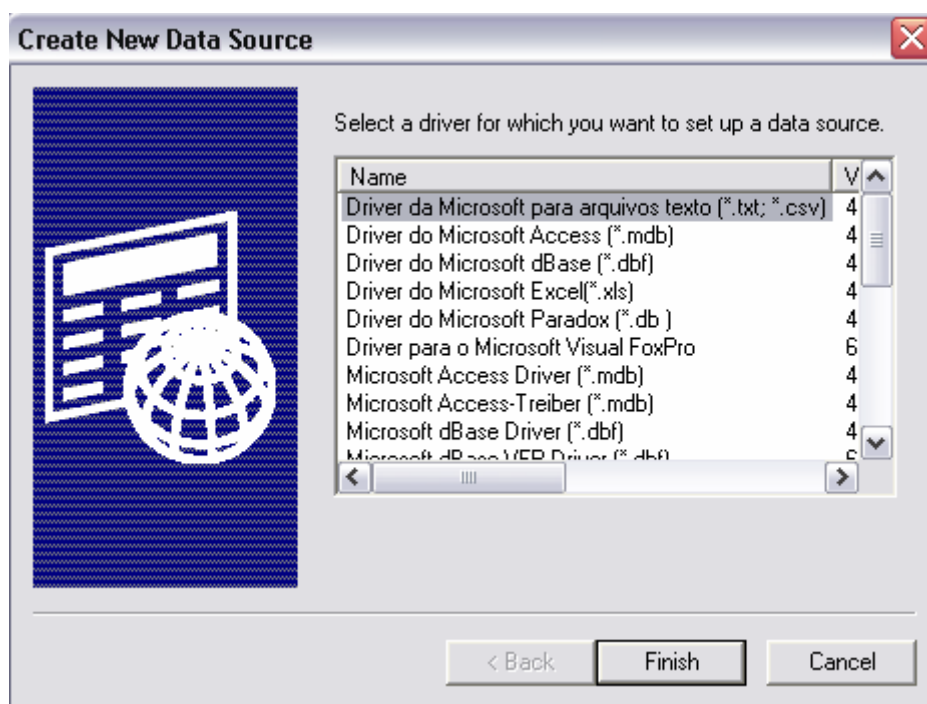
ODBC

Μετά που θα τρέξει το πρόγραμμα και γεμίσει η βάση δεδομένων κάνουμε τα εξής βήματα:

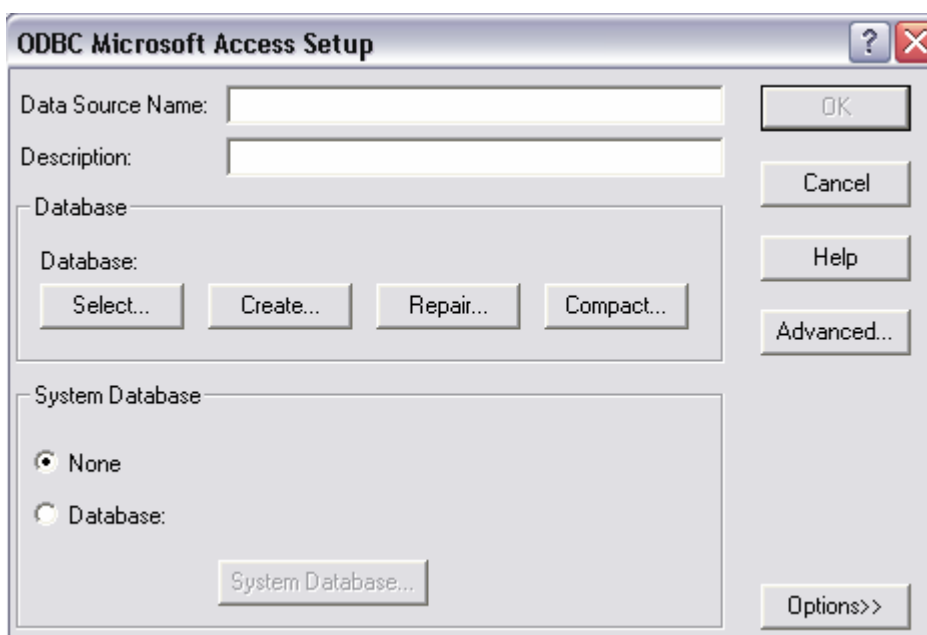
1. Από την μπάρα των Windows Start menu επιλέγουμε **Control Panel**
→ **Administrative Tools**→ **Data Sources (ODBC)**.



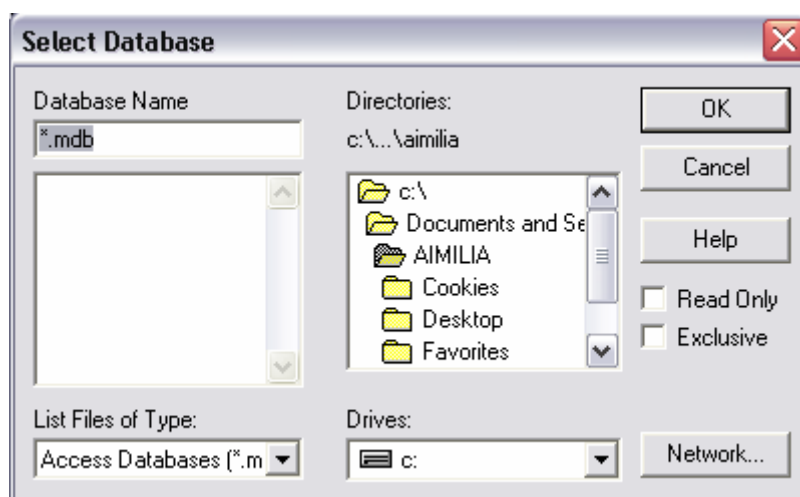
2. Επιλέγουμε το **User DSN**.
3. Κάνουμε κλικ στο **Add** και δημιουργούμε μία νέα πηγή δεδομένων (**Create New Data Source**) επιλέγοντας τον Driver που θέλουμε να εντάξουμε την νέα μας πηγή δεδομένων δηλ. Driver Do Microsoft Access (*.mdb)



4. Κάνουμε κλικ στο **Finish**.



5. Στο **Data Source Name** γράφουμε το όνομα της νέας πηγής δεδομένων π.χ. test.
Στο **Description** γράφουμε την περιγραφή της π.χ. base.
6. Επιλέγουμε **Select**.



7. Βρίσκουμε στο Directories **πού** θέλουμε έχει αποθηκευτεί η βάση δεδομένων(στο c:\) και στο Database Name τσεκάρουμε **3G.mdb**
8. Κάνουμε κλικ στο **OK** του Select Database.
9. Κάνουμε κλικ στο **OK** του ODBC Microsoft Access Setup.
10. Κάνουμε κλικ στο **OK** του ODBC Data Source Administrator.

ΣΗΜΑΝΤΙΚΟΙ ΟΡΙΣΜΟΙ

- **Active Set**: είναι ορισμένο ως το σύνολο των σταθμών βάσης του κινητού που είναι ταυτόχρονα συνδεδεμένο.
- **Monitored Set**: κυψέλες, οι οποίες δεν συμπεριλαμβάνονται στο Active Set αλλά συμπεριλαμβάνονται στη λίστα πληροφοριών της κυψέλης.
- **Detected Set**: εντοπισμένες κυψέλες από το κινητό που δεν είναι ούτε στη λίστα πληροφορίας κυψέλης ούτε στο Active Set
- **RSCP-Received Signal Code Power** (Λαμβανόμενη Ισχύς Κωδικού Σήματος). Δείχνει τη στάθμη του σήματος που έχουμε εκείνη τη στιγμή. Έτσι ξέρει το κινητό ποιος σταθμός είναι πιο ισχυρός δηλ. πιο κοντινός. Χρησιμοποιείται κυρίως για να υπολογίσει την απώλεια διαδρομής (path loss) από τη στιγμή που η ισχύς εκπομπής του CPICH (κοινού δοκιμαστικού καναλιού) είναι γνωστή ή μπορεί να διαβιβαστεί από το σύστημα πληροφοριών.
- **RSSI-Received Signal Strength Indicator** η λαμβανόμενη ευρυζωνική ισχύς στο B.W του καναλιού στο downlink. Είναι ένας δείκτης που δείχνει πόσο ισχυρό είναι το κανάλι των 5MHz.
- **Ec/No** είναι η πιο σημαντική μέτρηση στο WCDMA για τους σκοπούς σχεδίασης δικτύου λόγω του ότι είναι ο βασικός δείκτης κάλυψης αλλά επίσης έχει καλή ακρίβεια αφού η μέτρηση μπορεί να γίνει στη βασική ζώνη. Μετριέται μετά την αποδιαμόρφωση. Μας δείχνει πόσο ισχυρός είναι ο κώδικας σε σχέση με την ισχύ των 5MHz.

Spreading(διεύρυνση φάσματος)

Η έννοια του **Spreading** εφαρμόζεται στα φυσικά κανάλια και αποτελείται από 2 λειτουργίες .

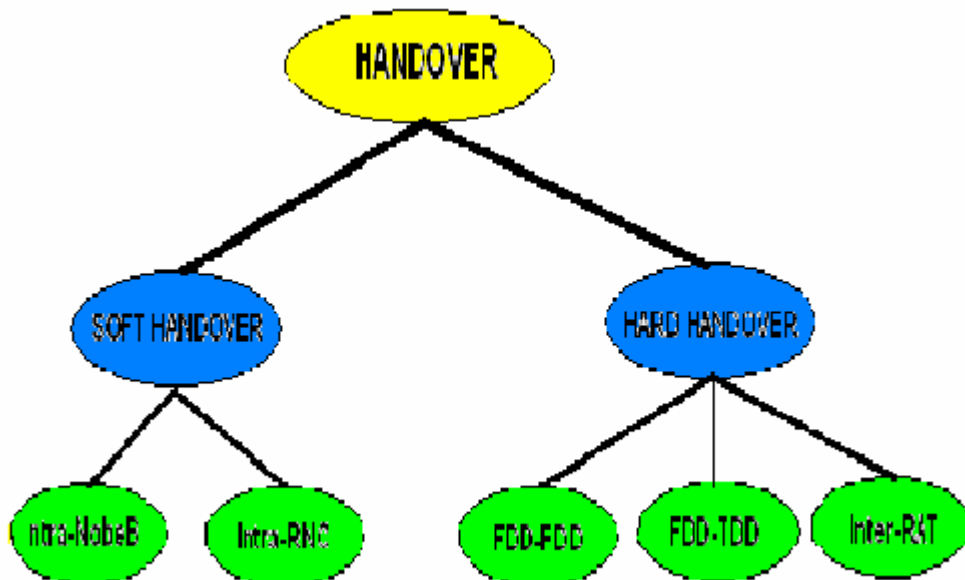
- Η πρώτη είναι η **Channelisation** η οποία μετατρέπει κάθε σύμβολο δεδομένων(bits) σε ένα αριθμό από chips, έτσι αυξάνει το εύρος ζώνης του σήματος. Ο αριθμός των chips ανά σύμβολο δεδομένων ονομάζεται SF(Spreading factor)
- Η δεύτερη λειτουργία είναι η **scrambling** η οποία εφαρμόζεται πάνω στο διευρυμένο σήμα και δεν επηρεάζει το εύρος ζώνης εκπομπής.
- **Scrambling Codes** : χρησιμοποιείται για να ξεχωρίσει τις διαφορετικές κυψέλες στη downlink και τα διαφορετικά UE(user equipment=κινητά) στην uplink κατεύθυνση.
- ✓ **Στη downlink** : σε κάθε κυψέλη(cell) εκχωρείται ένας και μόνο ένας πρωτεύον Scrambling Code (primary code). Υπάρχουν 512 primary codes που χωρίζονται σε 64 ομάδες των 8 Scrambling Code σε κάθε ομάδα.

- ✓ Στην uplink : υπάρχουν μερικά εκατομμύρια Scrambling Code και κάθε κινητό χρησιμοποιεί ένα Scrambling Code που του παρέχει το δίκτυο.

HANDOVER

Η μεταπομπή (handover) είναι ο μηχανισμός που εναλλάσσει την σύνδεση ενός χρήστη από συχνότητες οι οποίες χρησιμοποιούνται σε μια κυψέλη ενός ψηφιακού συστήματος κινητής τηλεφωνίας, είτε σε άλλες συχνότητες στην ίδια κυψέλη, είτε σε συχνότητες σε παρακείμενη κυψέλη. Η έναρξη του μηχανισμού προκαλείται, είτε με πτώση του σήματος κάτω από ένα επιτρεπτό επίπεδο λειτουργίας (threshold level), είτε με τον εντοπισμό ενός μεγάλου αποθέματος ισχύος από παρακείμενο σταθμό βάσης, είτε από φτωχό σε ποιότητα σήμα (όπως υποδεικνύεται π.χ. από έναν ρυθμό σφάλματος - error rate - ο οποίος ξεπερνά ένα επιτρεπτό μέγεθος) ή ακόμη από μία υπέρμετρη χρονική αύξηση (excess timing advance) ενώ το κινητό εντοπίζεται πολύ μακριά από τον σταθμό βάσης. (από το γλωσσάρι)

ΣΧΗΜΑ ΓΙΑ ΤΟ HANDOVER



Το Handover χωρίζεται σε Soft Handover και σε Hard Handover

Το Soft Handover με τη σειρά του χωρίζεται σε **Soft Handover** και **Softer Handover**.

- **Soft Handover:** σημαίνει ότι ο κινητός σταθμός αλλάζει το κανάλι του από ένα σέκτορα σ'άλλον σέκτορα μέσα σε μια κυψέλη. Αυτό εξασφαλίζει επιπλέον κέρδος κατευθυντικότητας και δεν δημιουργεί έξτρα μεταφορά ανάμεσα στον σταθμό βάσης και στον κινητό σταθμό.
- **Softer Handover:** ο κινητός σταθμός επιτρέπεται να είναι συνδεδεμένος ταυτοχρόνως σε πολλούς σταθμούς βάσης οι οποίοι εκλέγονται χρησιμοποιώντας κατώφλι αναφοράς. Ένας κινητός σταθμός εισέρχεται στο επίπεδο του Soft Handover όταν η δύναμη του σήματος της γειτονικής κυψέλης υπερβεί το προκαθορισμένο όριο αλλά είναι ακόμα κάτω από την ισχύουσα δύναμη του σήματος του σταθμού.

- **Hard Handover:** στο το WCDMA Hard Handover είναι χρήσιμο ή για τα μοιρασμένα κανάλια και τα κοινά ποιοτικά κανάλια με ενδιαμέση συχνότητα ή να αλληλεπιδρά για να εξισορροπήσει τα φορτία του δικτύου ή χρησιμοποιείται για να κάνει Handover σε χρήστες των άλλων συστημάτων όπως το GSM ή IS-95 από τα συστήματα του WCDMA.

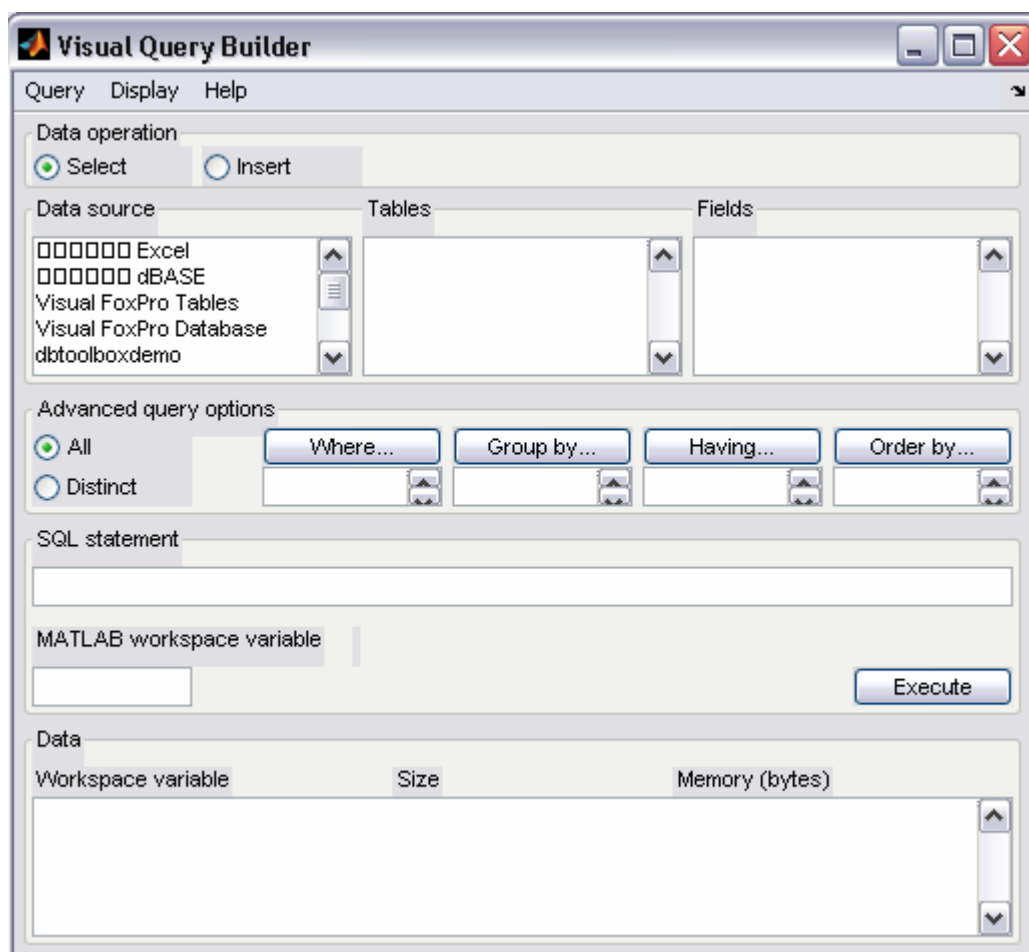
ΑΙΤΙΑ ΤΟΥ HANDOVER

- Ποιότητα σήματος του Handover
 - Συμβαίνει όταν η ποιότητα ή η ισχύς του σήματος πέσει κάτω από τις διευκρινισμένες τωρινές παραμέτρους στα κριτήρια του Handover.
 - Η αλλαγή σήματος εντοπίζεται από σταθερές μετρήσεις του σήματος που έγιναν και από το κινητό και από το σταθμό βάσης.
 - Η ποιότητα του σήματος εξαιτίας του Handover μπορεί να εκτελεστεί και για την uplink και downlink κατεύθυνση.
- Κίνηση Handover
 - Συμβαίνει όταν η χωρητικότητα της κίνησης μιας κυψέλης έχει φτάσει στο maximum.
 - Το κινητό κοντά στις κορυφές της κυψέλης με ψηλό φορτίο μπορεί να αναλάβει τον έλεγχο των γειτονικών κυψελών με την χαμηλότερη κίνηση.

ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΩΝ ΣΤΟ MATLAB

Visual Query Builder

Το **Visual Query Builder (VQB)** είναι ένα εύκολο στην χρησιμότητά του εργαλείο γραφικών που χρησιμεύει στην ανάκτηση των δεδομένων της βάσης μας. Για να ξεκινήσει το Visual Query Builder ανοίγουμε το **MATLAB** και στο Command Window γράφουμε τη λέξη **querybuilder**.



1. Επιλέγουμε την **Data Source**(πηγή δεδομένων) π.χ. test από την οποία θέλουμε να χρησιμοποιήσουμε συγκεκριμένα δεδομένα.
2. Επιλέγουμε στο **Table** τον πίνακα ή το ερώτημα που θα περιέχει τις στήλες(πεδία) που θέλουμε.
3. Επιλέγουμε τα **Fields**(πεδία) του πίνακα ή του ερωτήματος που περιέχουν τα δεδομένα που θέλουμε.

4. Τα Fields(πεδία) που επιλέχθηκαν φαίνονται στο **SQL statement**.
5. Στο **MATLAB workspace variable**, εκχωρούμε ένα γράμμα για το αποτέλεσμα των δεδομένων (π.χ. A).
6. Κάνουμε κλικ στο **Execute** για να τρέξει το query και να ανακτηθούν τα δεδομένα.
7. Πληροφορίες για τα δεδομένα εμφανίζονται στην περιοχή **Data**.
8. Κάνοντας διπλό κλικ στην περιοχή **Data** εμφανίζονται τα αποτελέσματα σε **Array Editor**.
9. Επιστρέφουμε στο **Command Window** του Matlab και γράφουμε το όνομα του M-file που μας ενδιαφέρει και μέσα σε παρένθεση το γράμμα που αποθηκεύσαμε τα αποτελέσματα π.χ. **ptyx (A)**.

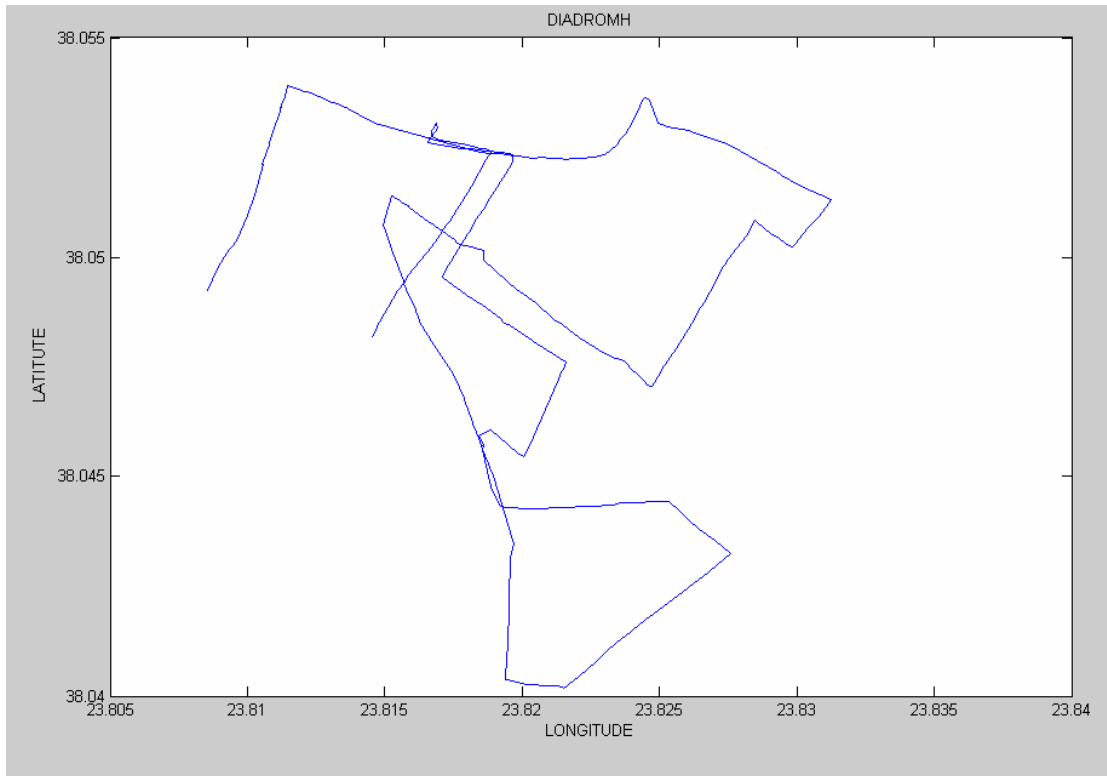
Αφού έχουμε κάνει τα βήματα στο ODBC που πρέπει **ανοίγουμε το MATLAB** κάνοντας κι εκεί τα βήματα που πρέπει και που προαναφέραμε με το εργαλείο **Visual Query Builder** σε συνδυασμό με το M-file.

Αναλυτικά

- **Για το αρχείο f001:**

ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ ΤΗΣ ΔΙΑΔΡΟΜΗΣ

- 1) Αφού έχουμε τρέξει το πρώτο αρχείο κι έχει γεμίσει η Βάση Δεδομένων, **ανοίγουμε το MATLAB και γράφουμε τη λέξη ‘querybuilder’.**
- 2) Επιλέγουμε **test** από το πεδίο **Data Source**.
- 3) Επιλέγουμε **CHARACTERISTICS** από το πεδίο **Tables**.
- 4) Επιλέγουμε **LONGITUDE, LATITUDE, METRHSIS** κρατώντας **πατημένο το Ctrl** από το πεδίο **Fields**.
- 5) Στο **Where** επιλέγουμε **LONGITUDE<>0** → Apply → OK.
- 6) Στο **MATLAB workspace variable** γράφουμε **A** και πατάμε **Execute**.
- 7) Από το **Display** επιλέγουμε **Chart**. Στο **Xdata** διαλέγουμε **LONGITUDE** και στο **Ydata** διαλέγουμε **LATITUDE** και πατάμε **Display**.



ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ RSCP SERVER.

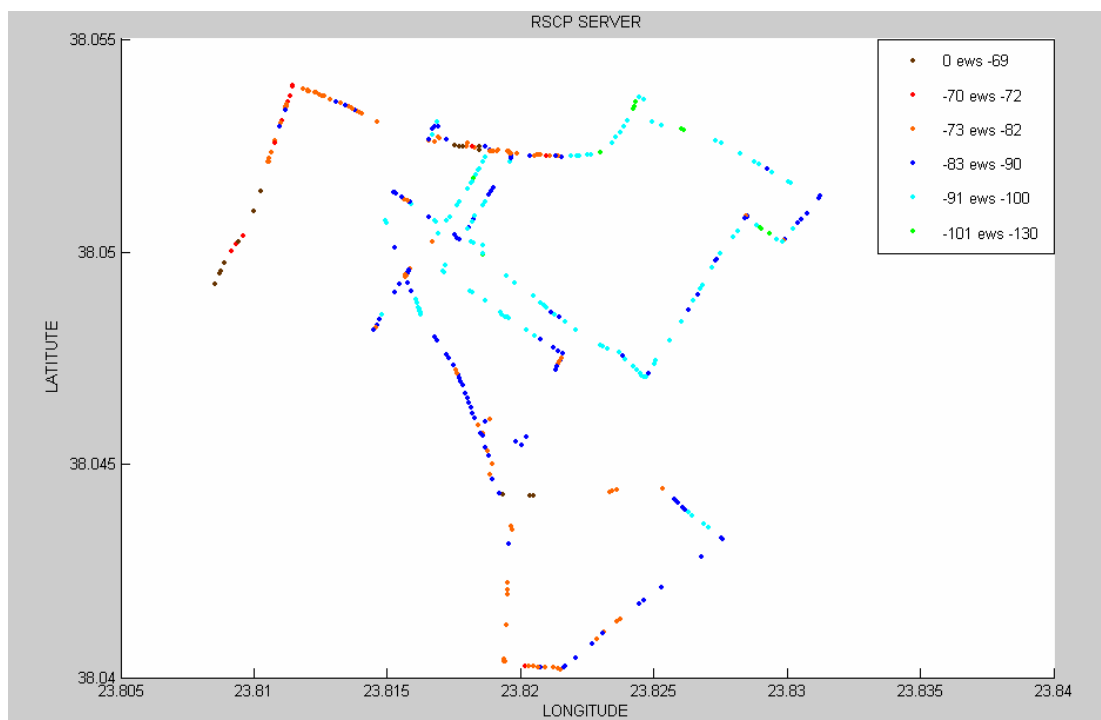
- 1) Αφού έχουμε τρέξει το πρώτο αρχείο κι έχει γεμίσει η Βάση Δεδομένων, ανοίγουμε το **MATLAB** και γράφουμε τη λέξη **'querybuilder'**.
- 2) Επιλέγουμε **test** από το πεδίο **Data Source**.
- 3) Επιλέγουμε **RSCP_SERVER_01_69** από το πεδίο **Tables**.
- 4) Επιλέγουμε **LONGITUDE, LATITUDE, METRHSEIS** κρατώντας πατημένο το **Ctrl** από το πεδίο **Fields**.
- 5) Στο **Where** επιλέγουμε **LONGITUDE<>0** → Apply → OK.
- 6) Στο **MATLAB workspace variable** γράφουμε **A** και πατάμε **Execute**.
- 7) Κάνουμε τα ίδια βήματα για **RSCP_SERVER_02_72** και γράφουμε **B**, για **RSCP_SERVER_03_82** και γράφουμε **C**, για **RSCP_SERVER_04_90** και γράφουμε **D**, για **RSCP_SERVER_05_100** και γράφουμε **E**, για **RSCP_SERVER_06_130** και γράφουμε **F**, για **RSCP_SERVER_07_140** και γράφουμε **G**.

8) Χρησιμοποιούμε το M-File **rscp.m** .

```
function rscp(metab1,db)
metabm1=cell2mat(metab1);

figure(1)
switch db
case 69
    H=plot3(metabm1(:,1),metabm1(:,2),metabm1(:,3),'.','linewidth',2)
    set(H,'color',[0.4 0.2 0]) %kafe
    view(0.001,90)
case 72
    plot3(metabm1(:,1),metabm1(:,2),metabm1(:,3),'r','linewidth',2) %kokkino
    view(0.001,90)
case 82
    H=plot3(metabm1(:,1),metabm1(:,2),metabm1(:,3),'.','linewidth',2)
    set(H,'color',[1 0.4 0]) %portokali
    view(0.001,90)
case 90
    plot3(metabm1(:,1),metabm1(:,2),metabm1(:,3),'b','linewidth',2) %mble
    view(0.001,90)
case 100
    plot3(metabm1(:,1),metabm1(:,2),metabm1(:,3),'c','linewidth',2) %galazio
    view(0.001,90)
case 130
    plot3(metabm1(:,1),metabm1(:,2),metabm1(:,3),'g','linewidth',2) %prasino
    view(0.001,90)
case 140
    plot3(metabm1(:,1),metabm1(:,2),metabm1(:,3),'y','linewidth',2) %kitrino
    view(0.001,90)
otherwise
end
hold on;
end
```

9) Επιστρέφουμε στο Command Window του Matlab και γράφουμε rscp(A,69) και Enter, rscp(B,72) και Enter, rscp(C,82) και Enter, rscp(D,90) και Enter, rscp(E,100) και Enter, rscp(F,130) και Enter, rscp(G,140) και Enter.



Το γράφημα αυτό δείχνει την ποιότητα της στάθμης του σήματος (RSCP) στη διαδρομή. Βλέπουμε λοιπόν ότι όσο μεγαλύτερη είναι η στάθμη του σήματος τόσο καλύτερο είναι το σήμα μας. Δηλαδή με το καφέ χρώμα απεικονίζονται οι περιοχές με το καλύτερο σήμα, με το πράσινο οι περιοχές με το χειρότερο σήμα και τα ενδιάμεσα χρώματα δείχνουν τις περιοχές με το μέτριο σήμα.

ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ INTERFERENCE (ΠΑΡΕΜΒΟΛΗΣ) SERVER.

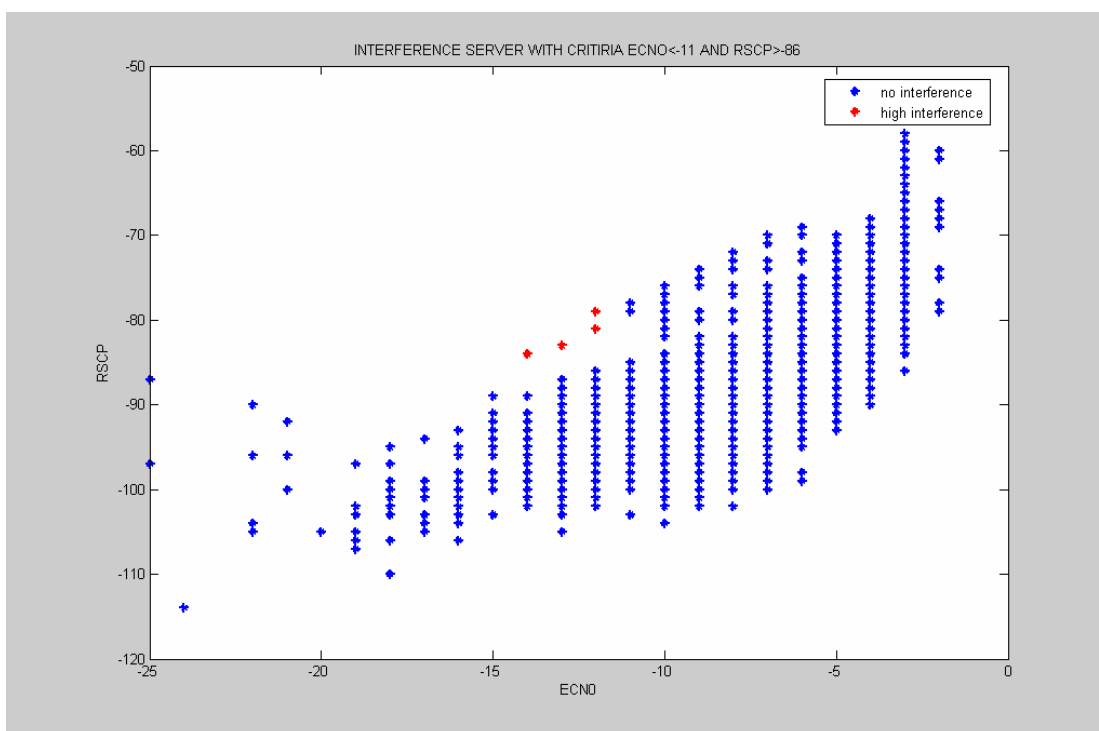
- 1) Αφού έχουμε τρέξει το πρώτο αρχείο κι έχει γεμίσει η Βάση Δεδομένων, ανοίγουμε το **MATLAB** και γράφουμε τη λέξη **'querybuilder'**.
- 2) Επιλέγουμε **test** από το πεδίο **Data Source**.
- 3) Επιλέγουμε **INTERFERENCE_SERVER** από το πεδίο **Tables**.
- 4) Επιλέγουμε **ECNO, RSCP** κρατώντας πατημένο το **Ctrl** από το πεδίο **Fields**.
- 5) Στο **MATLAB workspace variable** γράφουμε **A** και πατάμε **Execute**.
- 6) Επιλέγουμε **INTERFERENCE_SERVER_WITHCRITERIA** από το πεδίο **Tables**.
- 7) Επιλέγουμε **ECNO, RSCP** κρατώντας πατημένο το **Ctrl** από το πεδίο **Fields**.

8) Στο MATLAB **workspace variable** γράφουμε **B** και πατάμε **Execute**.

9) Χρησιμοποιούμε το M-File **intpoint.m** .

```
function intpoint(metab1,metab2)
metabm1=cell2mat(metab1);
metabm2=cell2mat(metab2); %with criteria
figure(1)
plot(metabm1(:,1),metabm1(:,2),'b*','linewidth',2) %mple
hold on
plot(metabm2(:,1),metabm2(:,2),'r*','linewidth',2) %with criteria kokkino
end
```

10) Επιστρέφουμε στο Command Window του Matlab και γράφουμε `intpoint(A,B)` και Enter.



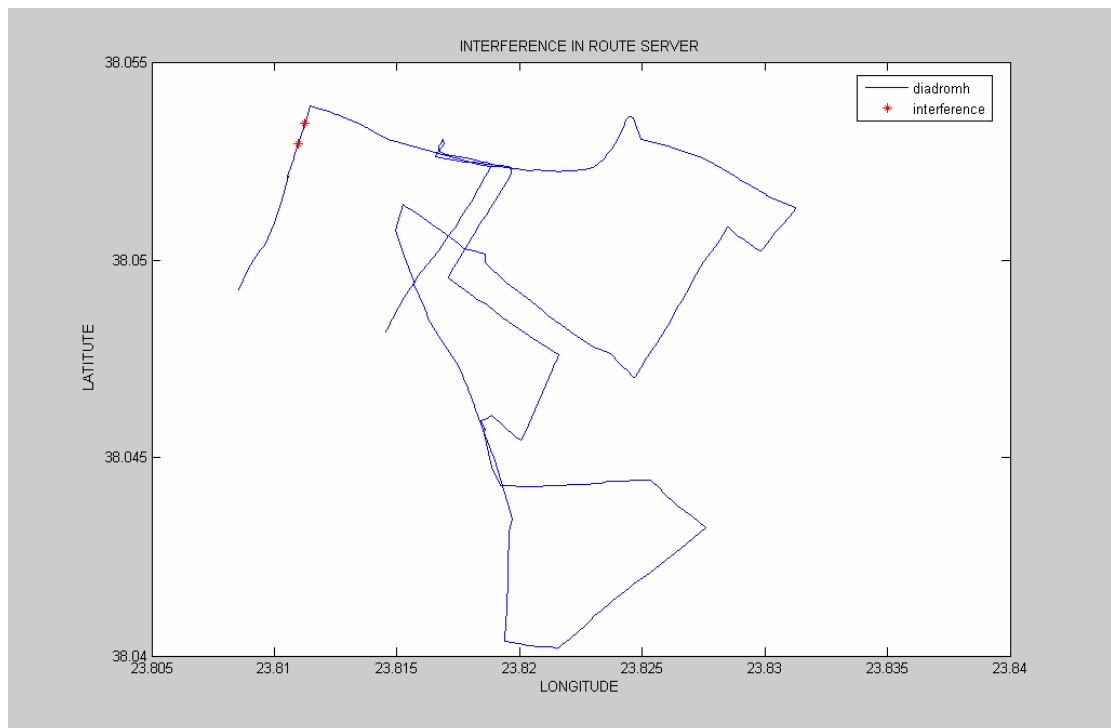
Όταν το $ECNO < -11$ και $RSCP > -86$ έχουμε παρεμβολή στο σήμα μας. Στο συγκεκριμένο γράφημα η προβληματική κάλυψη του σήματος απεικονίζεται κόκκινα σημεία.

**ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ INTERFERENCE (ΠΑΡΕΜΒΟΛΗΣ)
SERVER ΠΑΝΩ ΣΤΗ ΔΙΑΔΡΟΜΗ(IN ROUTE).**

- 1) Αφού έχουμε τρέξει το πρώτο αρχείο κι έχει γεμίσει η Βάση Δεδομένων, ανοίγουμε το **MATLAB** και γράφουμε τη λέξη **'querybuilder'**.
- 2) Επιλέγουμε **test** από το πεδίο **Data Source**.
- 3) Επιλέγουμε **CHARACTERISTICS** από το πεδίο **Tables**.
- 4) Επιλέγουμε **LONGITUDE, LATITUDE** κρατώντας πατημένο το **Ctrl** από το πεδίο **Fields**.
- 5) Στο **Where** επιλέγουμε **LONGITUDE<>0** → Apply → OK.
- 6) Στο **MATLAB workspace variable** γράφουμε **A** και πατάμε **Execute**.
- 7) Επιλέγουμε **INTERFERENCE_IN_ROUTE_SERVER** από το πεδίο **Tables**.
- 8) Επιλέγουμε **ECNO, RSCP, LONGITUDE, LATITUDE** κρατώντας πατημένο το **Ctrl** από το πεδίο **Fields**.
- 9) Στο **MATLAB workspace variable** γράφουμε **B** και πατάμε **Execute**.
- 10) Χρησιμοποιούμε το M-File **introute.m**.

```
function introute(metab1,metab2)
metabm1=cell2mat(metab1);
metabm2=cell2mat(metab2);
figure(1)
plot(metabm1(:,1),metabm1(:,2))
hold on
plot(metabm2(:,1),metabm2(:,2),'r*')
end
```

- 11) Επιστρέφουμε στο Command Window του Matlab και γράφουμε **introute(A,B)** και Enter.



Το παραπάνω γράφημα δείχνει την παρεμβολή του σήματος με βάση τα κριτήρια μας πάνω στη διαδρομή.

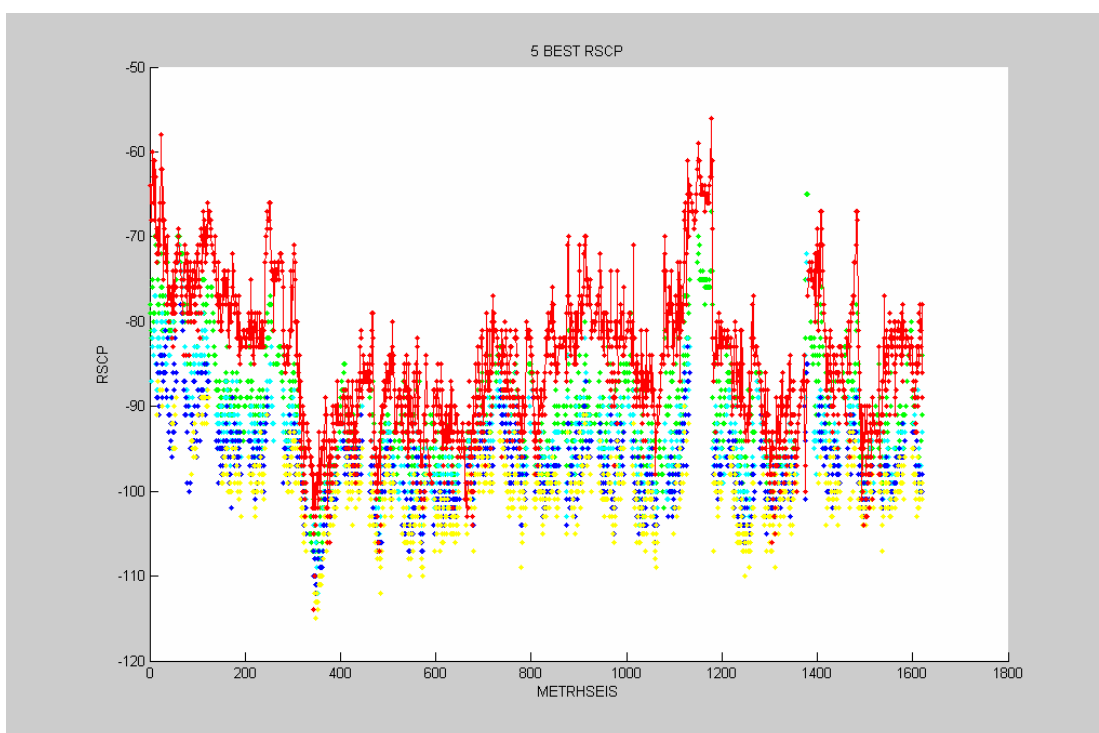
ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ BEST SERVERS ΚΑΙ BEST NEIGHBOURS.

- 1) Αφού έχουμε τρέξει το πρώτο αρχείο κι έχει γεμίσει η Βάση Δεδομένων, ανοίγουμε το MATLAB και γράφουμε τη λέξη 'querybuilder'.
- 2) Επιλέγουμε **test** από το πεδίο **Data Source**.
- 3) Επιλέγουμε **BEST_SERVERS** από το πεδίο **Tables**.
- 4) Επιλέγουμε **METRHSEIS, BESTSERVER, BESTNEIGH1, BESTNEIGH2, BESTNEIGH3, BESTNEIGH4** κρατώντας πατημένο το **Ctrl** από το πεδίο **Fields**.
- 5) Στο MATLAB **workspace variable** γράφουμε **all** και πατάμε **Execute**.
- 6) Επιλέγουμε **bestn0** από το πεδίο **Tables**.
- 7) Επιλέγουμε **METRHSEIS, BESTSERVER, SCR** κρατώντας πατημένο το **Ctrl** από το πεδίο **Fields**.
- 8) Στο MATLAB **workspace variable** γράφουμε **a** και πατάμε **Execute**.
- 9) Επιλέγουμε **bestn1** από το πεδίο **Tables**.

- 10) Επιλέγουμε **METRHSEIS, BESTSERVER, SCR** κρατώντας πατημένο το **Ctrl** από το πεδίο **Fields**.
- 11) Στο MATLAB **workspace variable** γράφουμε **b** και πατάμε **Execute**.
- 12) Επιλέγουμε **bestn2** από το πεδίο **Tables**.
- 13) Επιλέγουμε **METRHSEIS, BESTSERVER, SCR** κρατώντας πατημένο το **Ctrl** από το πεδίο **Fields**.
- 14) Στο MATLAB **workspace variable** γράφουμε **c** και πατάμε **Execute**.
- 15) Επιλέγουμε **bestn3** από το πεδίο **Tables**.
- 16) Επιλέγουμε **METRHSEIS, BESTSERVER, SCR** κρατώντας πατημένο το **Ctrl** από το πεδίο **Fields**.
- 17) Στο MATLAB **workspace variable** γράφουμε **d** και πατάμε **Execute**.
- 18) Επιλέγουμε **bestn4** από το πεδίο **Tables**.
- 19) Επιλέγουμε **METRHSEIS, BESTSERVER, SCR** κρατώντας πατημένο το **Ctrl** από το πεδίο **Fields**.
- 20) Στο MATLAB **workspace variable** γράφουμε **e** και πατάμε **Execute**.
- 21) Χρησιμοποιούμε το M-File **bestsrscp.m** .

```
function bestsrscp(met1,met2,met3,met4,met5,met6)
[a b c d e ]=ptyx41_1(met2,met3,met4,met5,met6);
ptyx51(met1)
ptyx6(a)
ptyx6(b)
ptyx6(c)
ptyx6(d)
ptyx6(e)
ptyx7(a,1)
```

- 22) Επιστρέφουμε στο Command Window του Matlab και γράφουμε **bestsrscp(all,a,b,c,d,e)** και **Enter**.

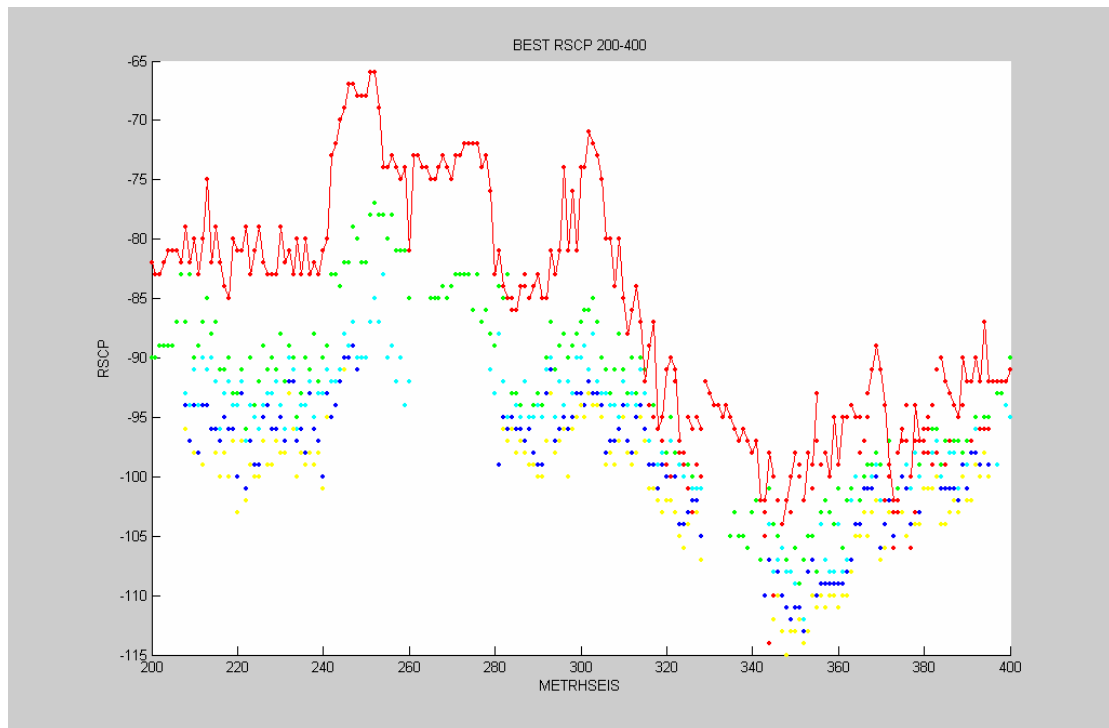
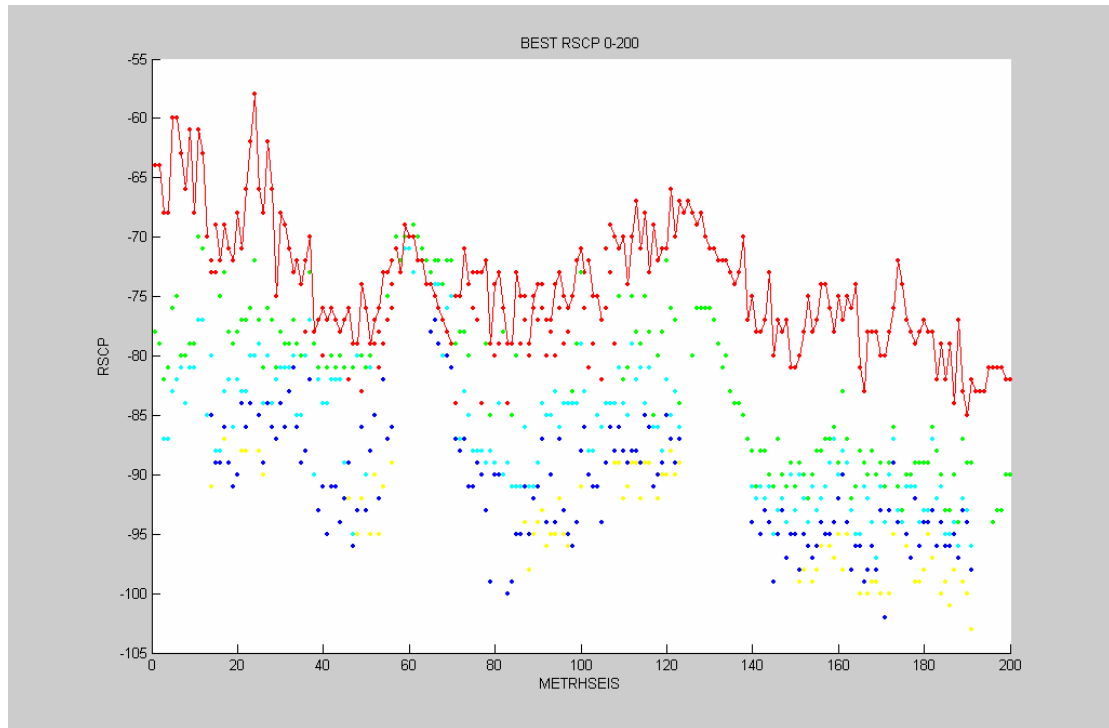


Στη γραφική αυτή φαίνονται τα 5 καλύτερα RSCP σε κάθε μέτρηση, τα οποία με κόκκινο χρώμα είναι του SERVER, με πράσινο απεικονίζεται το δεύτερο καλύτερο RSCP το οποίο είναι από MONITORED ή DETECTED εκτός αν είναι SERVER και εμφανιστεί με κόκκινο χρώμα, με γαλάζιο απεικονίζεται το τρίτο καλύτερο RSCP το οποίο είναι από MONITORED ή DETECTED εκτός αν είναι SERVER και εμφανιστεί με κόκκινο χρώμα, με μπλέ απεικονίζεται το τέταρτο καλύτερο RSCP το οποίο είναι από MONITORED ή DETECTED εκτός αν είναι SERVER και εμφανιστεί με κόκκινο χρώμα και με κίτρινο απεικονίζεται το πέμπτο καλύτερο RSCP το οποίο είναι από MONITORED ή DETECTED εκτός αν είναι SERVER και εμφανιστεί με κόκκινο χρώμα. Αν τώρα οι SERVER έχουν το ίδιο SCRAMBLING CODE σε διαδοχικές μετρήσεις τότε τα σημεία αυτά ενώνονται διαφορετικά δεν ενώνονται.

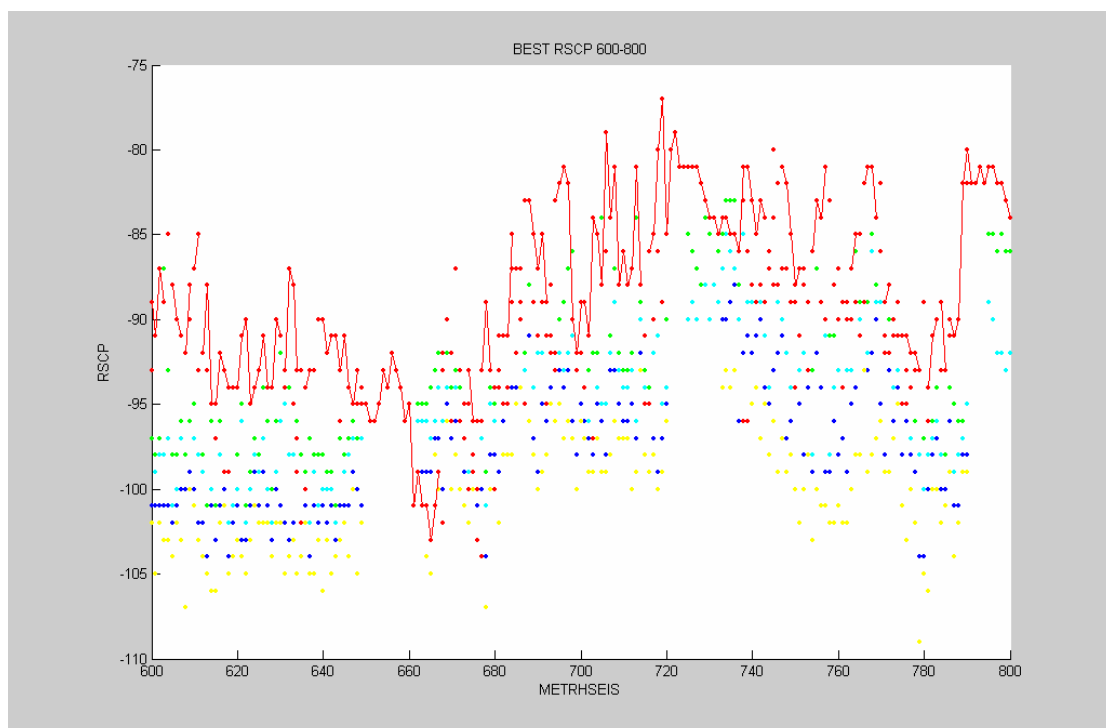
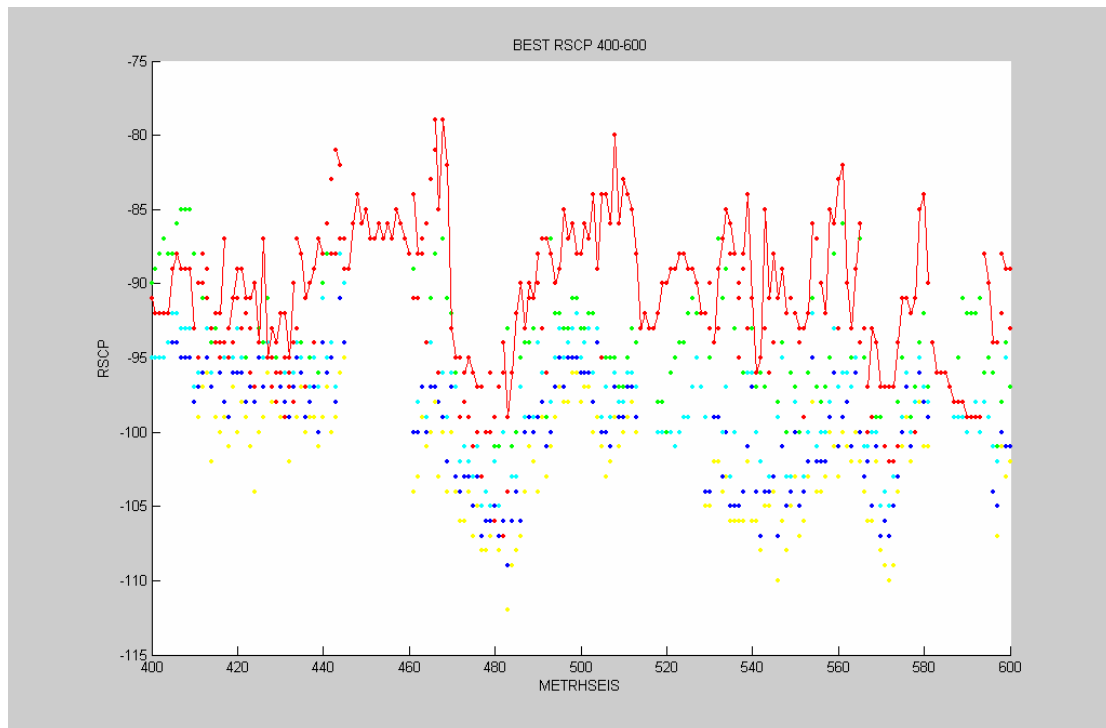
Το συμπέρασμα που βγαίνει από αυτή τη γραφική είναι ότι όταν έχουμε active set > 1 δηλαδή περισσότερους από έναν SERVER στην ίδια μέτρηση τότε έχουμε Soft Handover που σημαίνει ότι ο κινητός σταθμός αλλάζει το κανάλι του από ένα σέκτορα σ'άλλον σέκτορα μέσα σε μια κυψέλη. Αυτό εξασφαλίζει επιπλέον κέρδος κατευθυντικότητας και δεν δημιουργεί έξτρα μεταφορά ανάμεσα στον σταθμό βάσης και στον κινητό σταθμό.

ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)

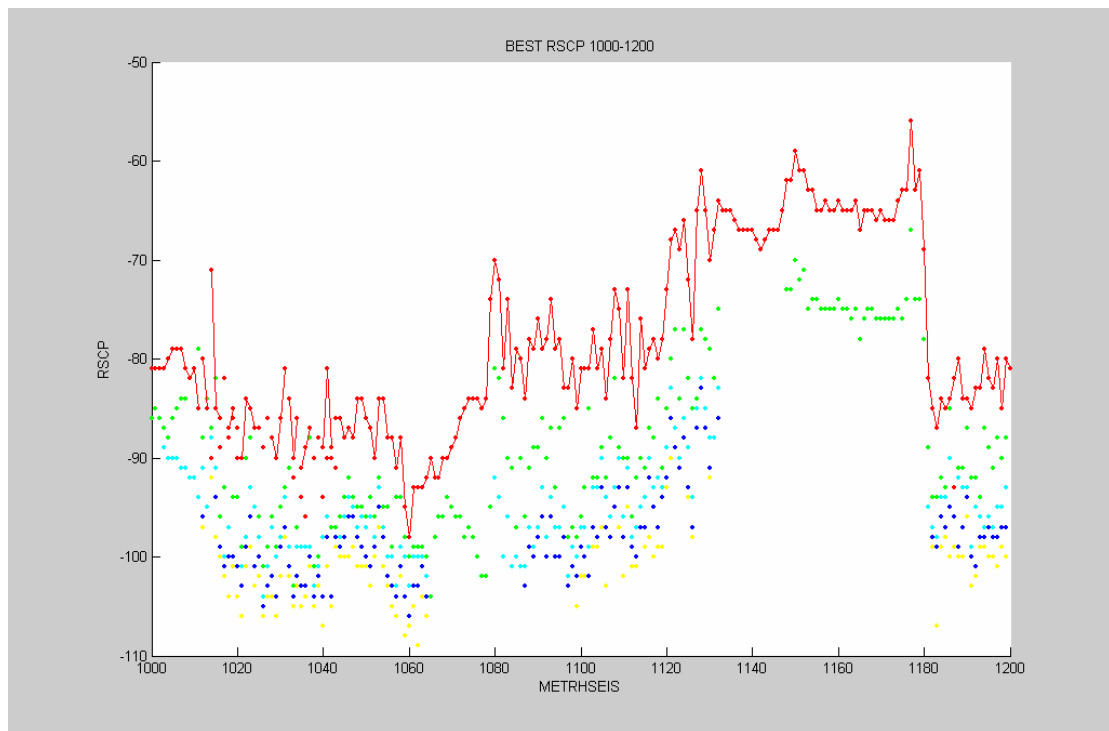
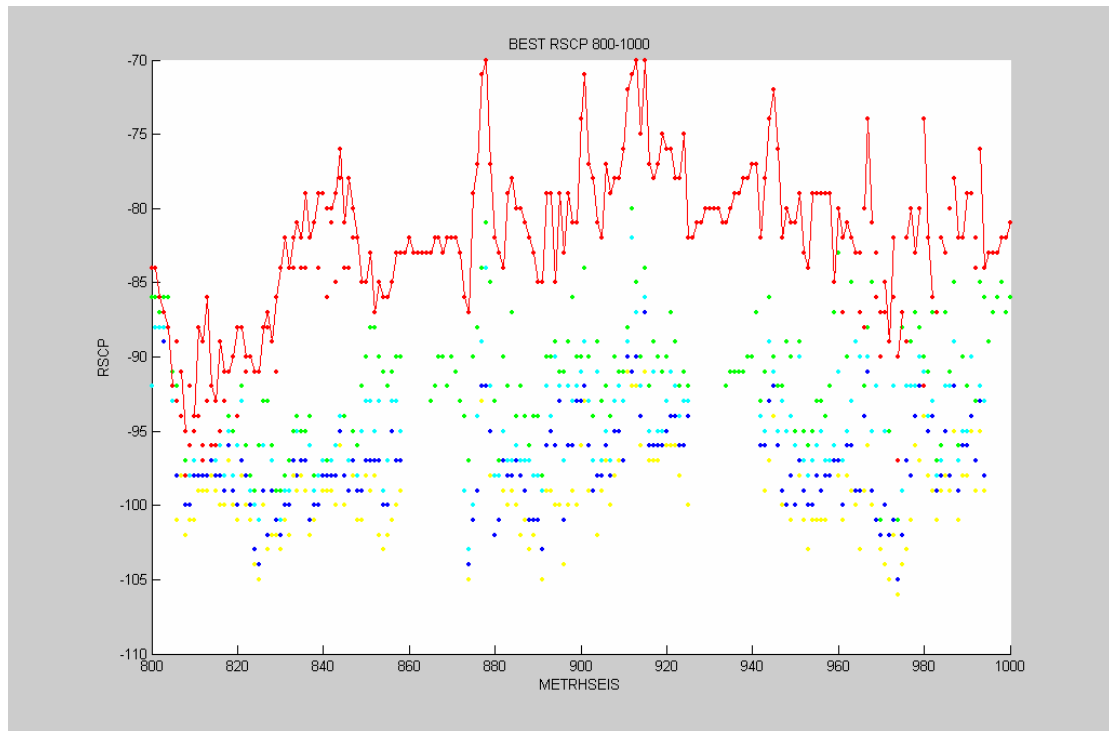
Οι παρακάτω γραφικές δείχνουν την προηγούμενη γραφική χωρισμένη ανά 200 μετρήσεις.

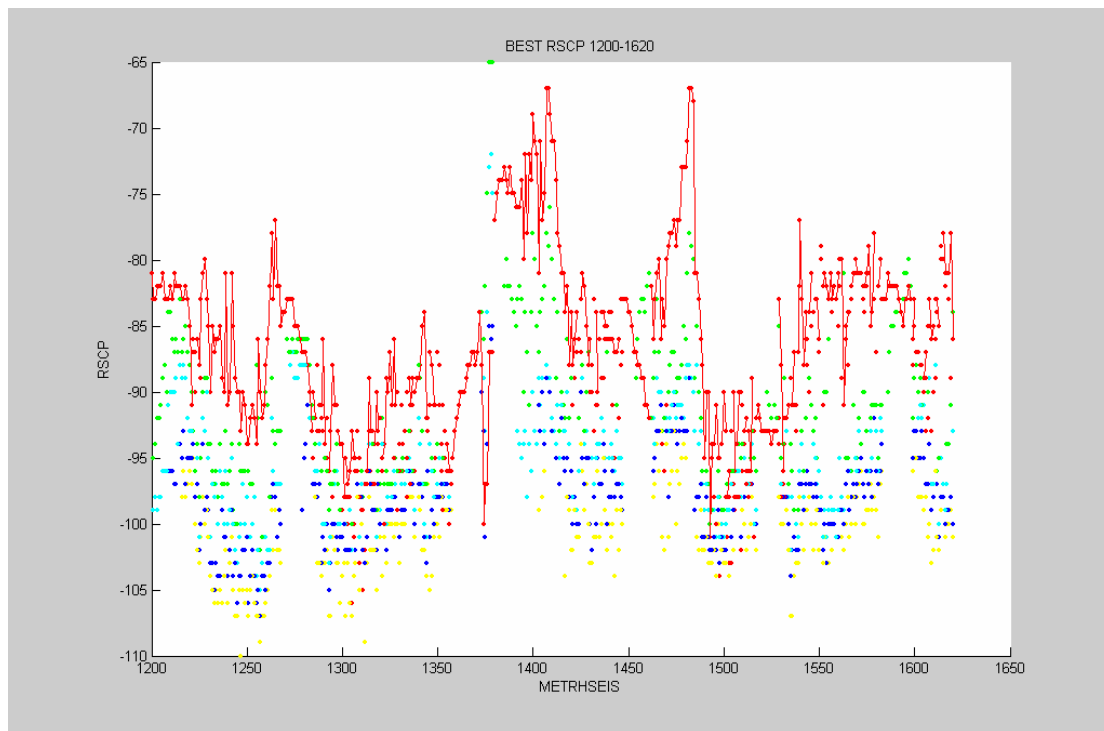


ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)



ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)





ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ SCRAMBLING CODE.

- 1) Αφού έχουμε τρέξει το πρώτο αρχείο κι έχει γεμίσει η Βάση Δεδομένων, ανοίγουμε το **MATLAB** και γράφουμε τη λέξη **'querybuilder'**.
- 2) Επιλέγουμε **test** από το πεδίο **Data Source**.
- 3) Επιλέγουμε **SCRCODE** από το πεδίο **Tables**.
- 4) Επιλέγουμε **LONGITUDE, LATITUDE, METRHSEIS, RSCP, SCRAMBLING_CODE** κρατώντας πατημένο το **Ctrl** από το πεδίο **Fields**.
- 5) Στο **Where** επιλέγουμε **LONGITUDE<>0 AND SCRAMBLING_CODE = π.χ. 144** → Apply → OK.
- 6) Στο **MATLAB workspace variable** γράφουμε **A** και πατάμε **Execute**.
- 7) Χρησιμοποιούμε το M-File **scrcode.m**.

```
function scrcode(metab1)
metabm1=cell2mat(metab1);
figure(1)

for i=1:size(metabm1,1)
    rscp=metabm1(i,4);
```

```

if rscp>=-69
    H=plot3(metabm1(i,1),metabm1(i,2),metabm1(i,4),'.');
    set(H,'color',[0.4 0.2 0]) %kafe
    view(0.001,90)
elseif rscp<-69 & rscp>=-72
    plot3(metabm1(i,1),metabm1(i,2),metabm1(i,4),'.r')
    view(0.001,90)
elseif rscp<-72 & rscp>=-82
    H=plot3(metabm1(i,1),metabm1(i,2),metabm1(i,4),'.');
    set(H,'color',[1 0.4 0]) %portokali
    view(0.001,90)
elseif rscp<-82 & rscp>=-90
    plot3(metabm1(i,1),metabm1(i,2),metabm1(i,4),'.b')
    view(0.001,90)
elseif rscp<-90 & rscp>=-100
    plot3(metabm1(i,1),metabm1(i,2),metabm1(i,4),'.c')
    view(0.001,90)
elseif rscp<-100 & rscp>=-130
    plot3(metabm1(i,1),metabm1(i,2),metabm1(i,4),'.g')
    view(0.001,90)
else
    plot3(metabm1(i,1),metabm1(i,2),metabm1(i,4),'.y')
    view(0.001,90)
end

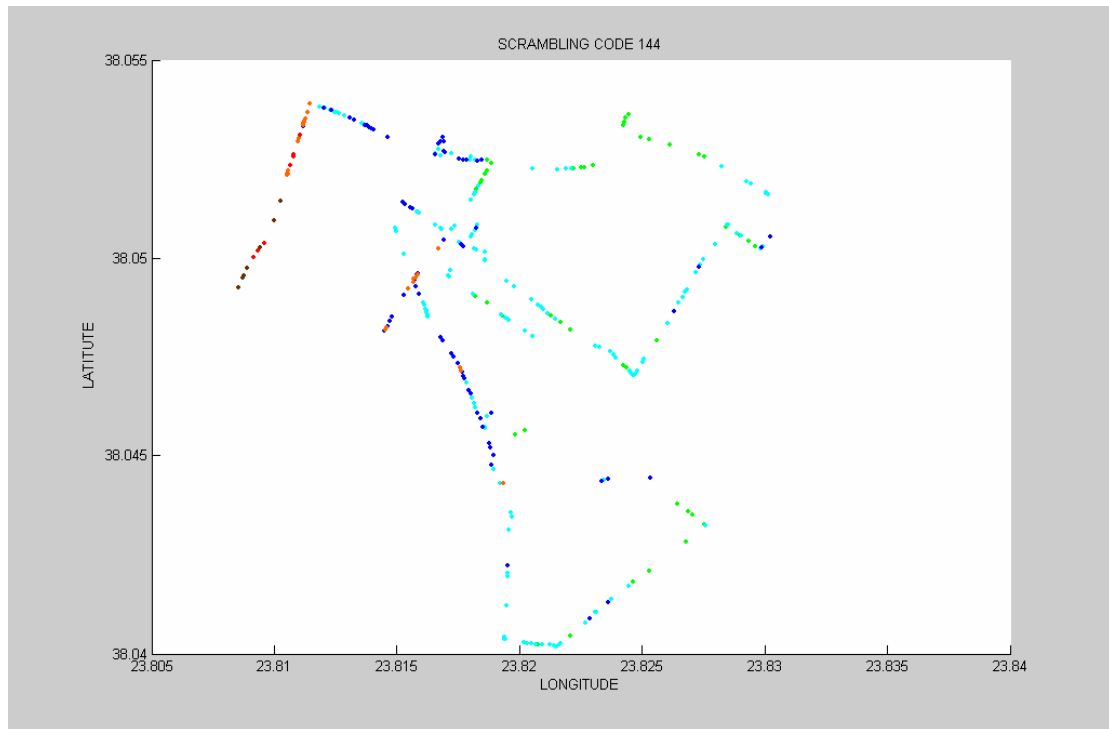
    hold on
end

end

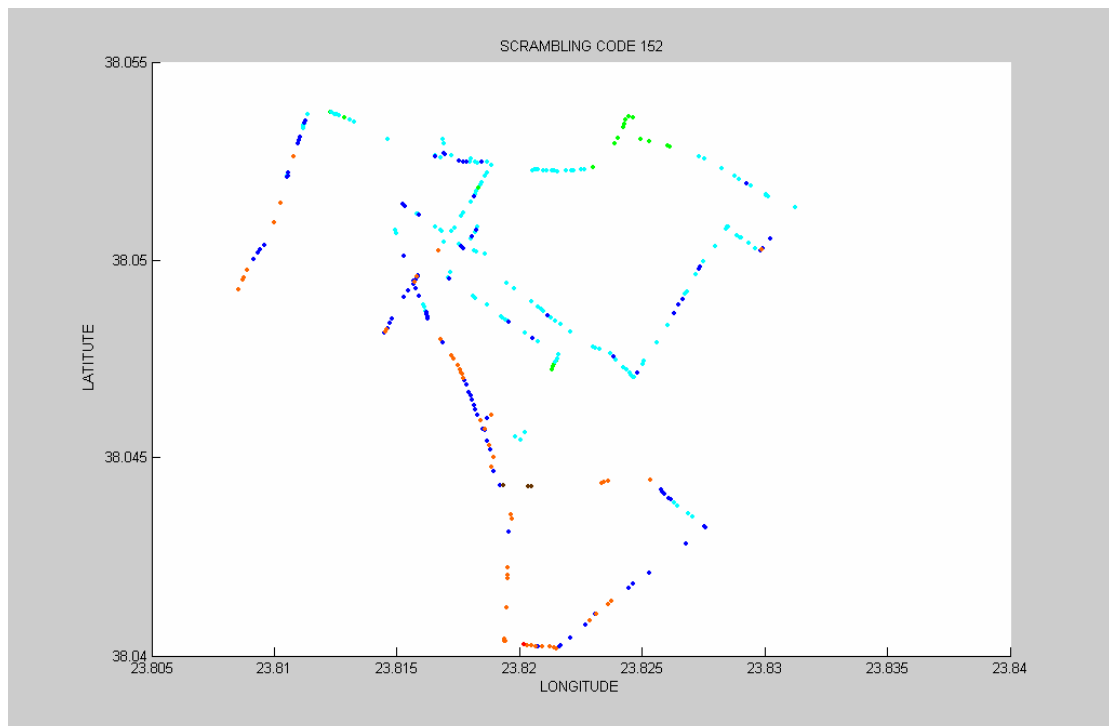
```

- 8) Επιστρέφουμε στο Command Window του Matlab και γράφουμε `scrconde(A)` και Enter.

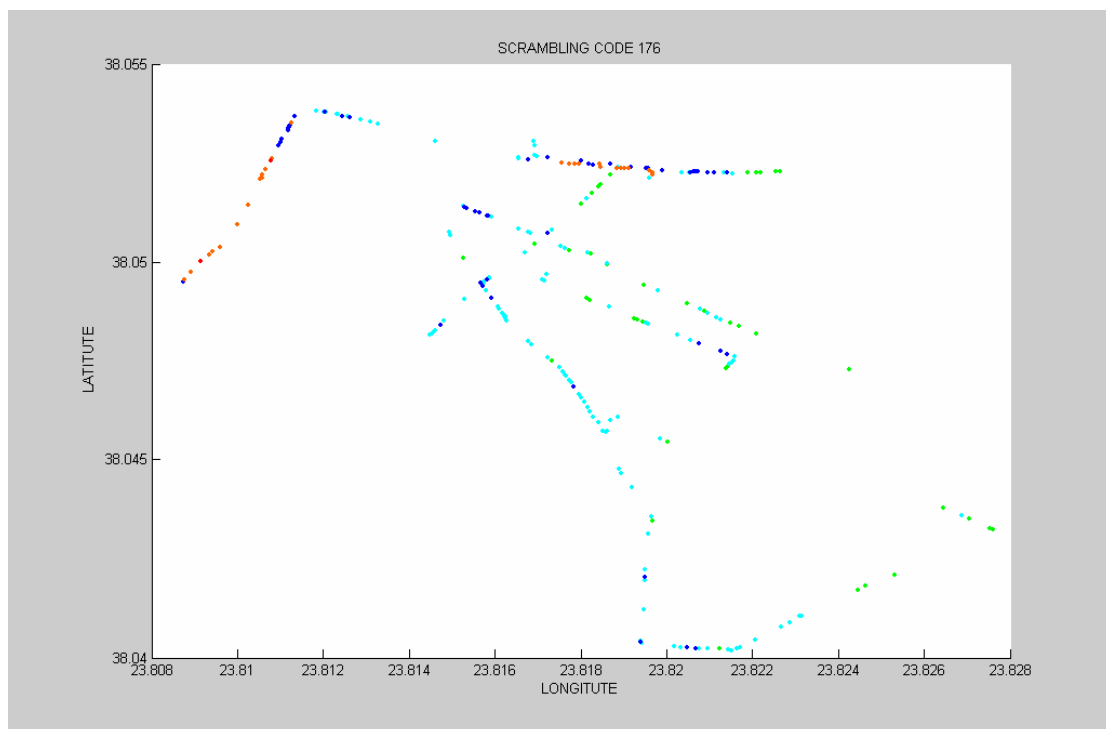
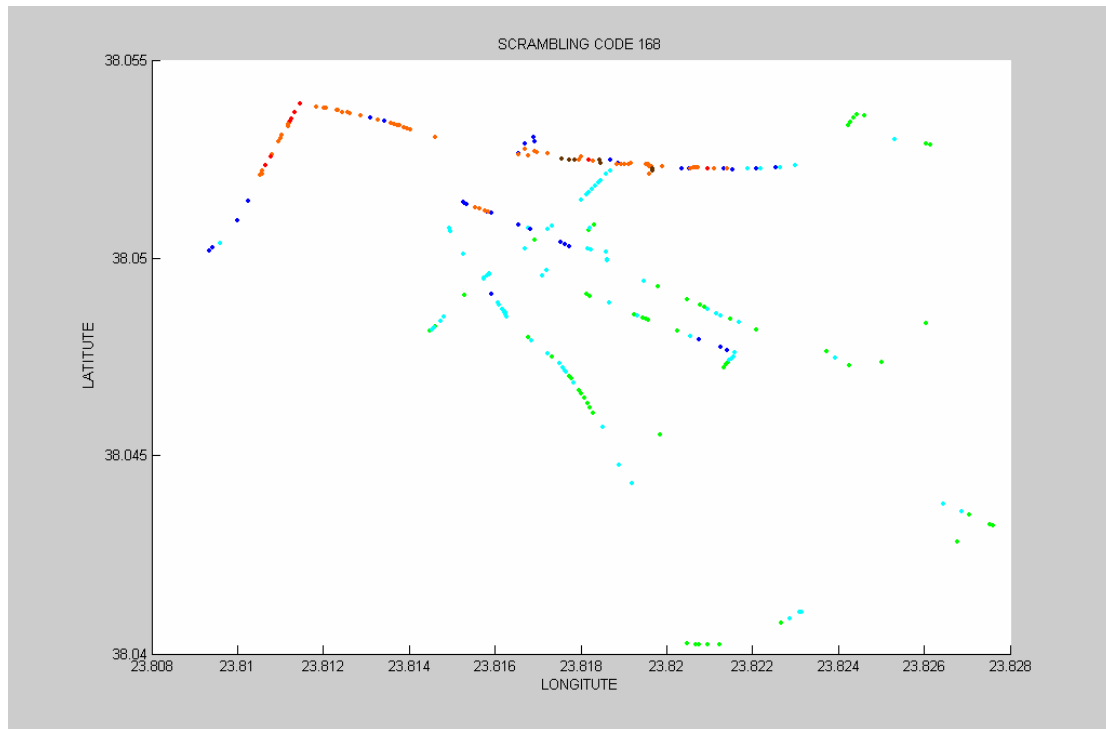
ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)



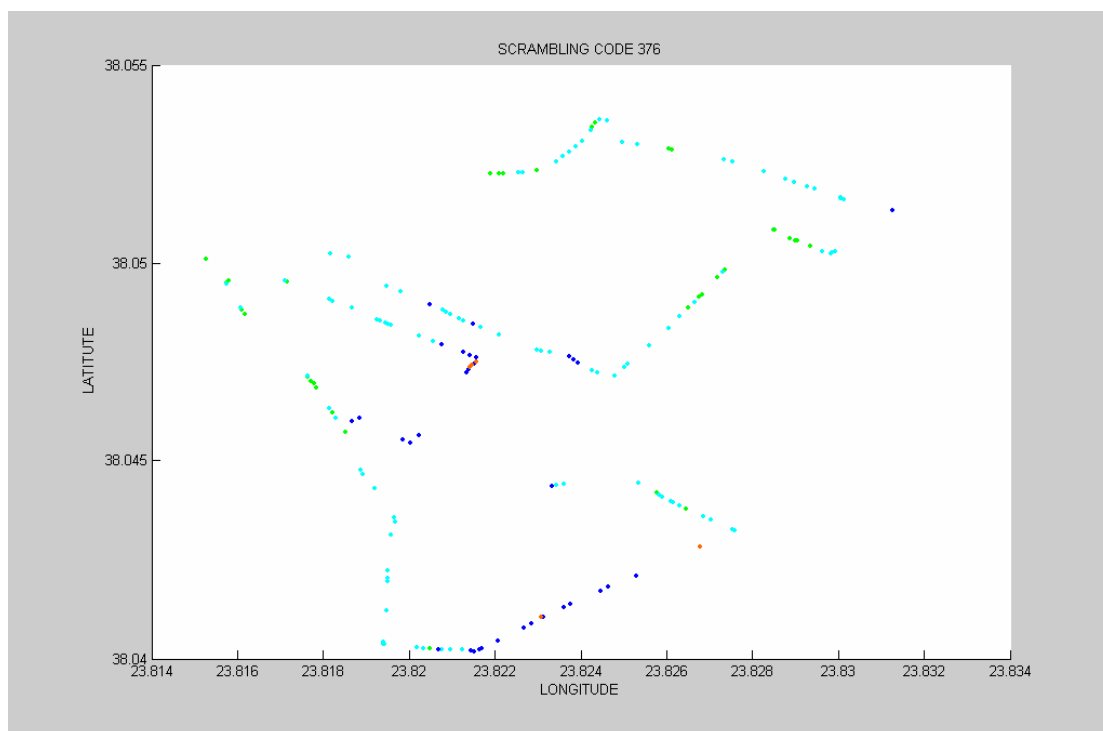
Το Scrambling Code χρησιμοποιείται για να ξεχωρίσει τις διαφορετικές κυψέλες στη downlink και τα διαφορετικά UE(user equipment=κινητά) στην uplink κατεύθυνση. Στη γραφική μας φαίνεται η διαδρομή με όλα τα RSCP που έχουν Scrambling code 144. Οπότε ο σέκτορας (cell) που μας σερβίρει είναι αυτός με το καλύτερο RSCP που απεικονίζεται με καφέ ή κόκκινο χρώμα.



ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)



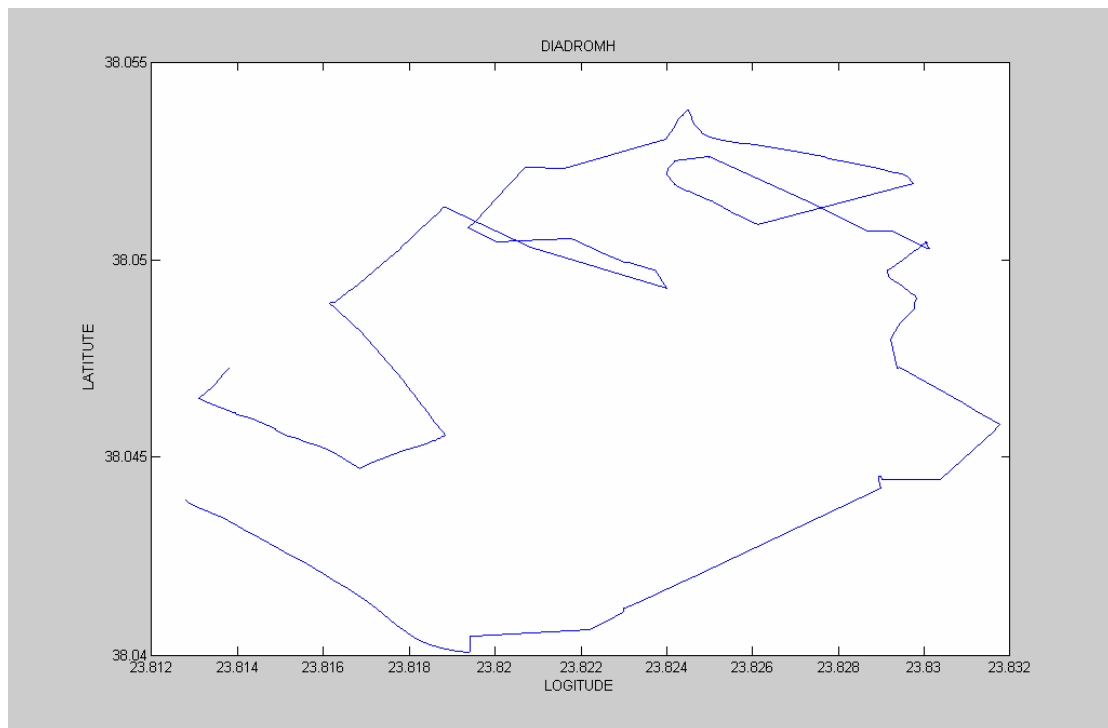
ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)



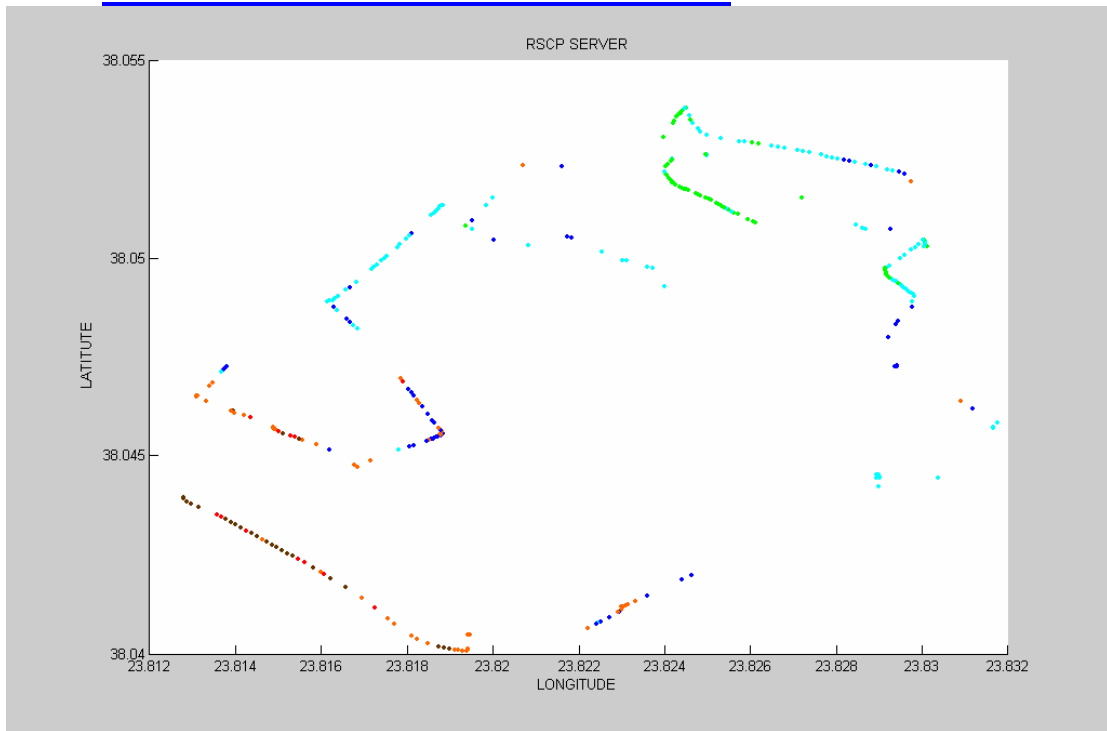
- Για το αρχείο f002:

Για τον τρόπο δημιουργίας γραφικών χρησιμοποιούμε την ίδια διαδικασία με αυτή του αρχείου F001 και τα συμπεράσματα αυτών είναι τα ίδια με τα προηγούμενα.

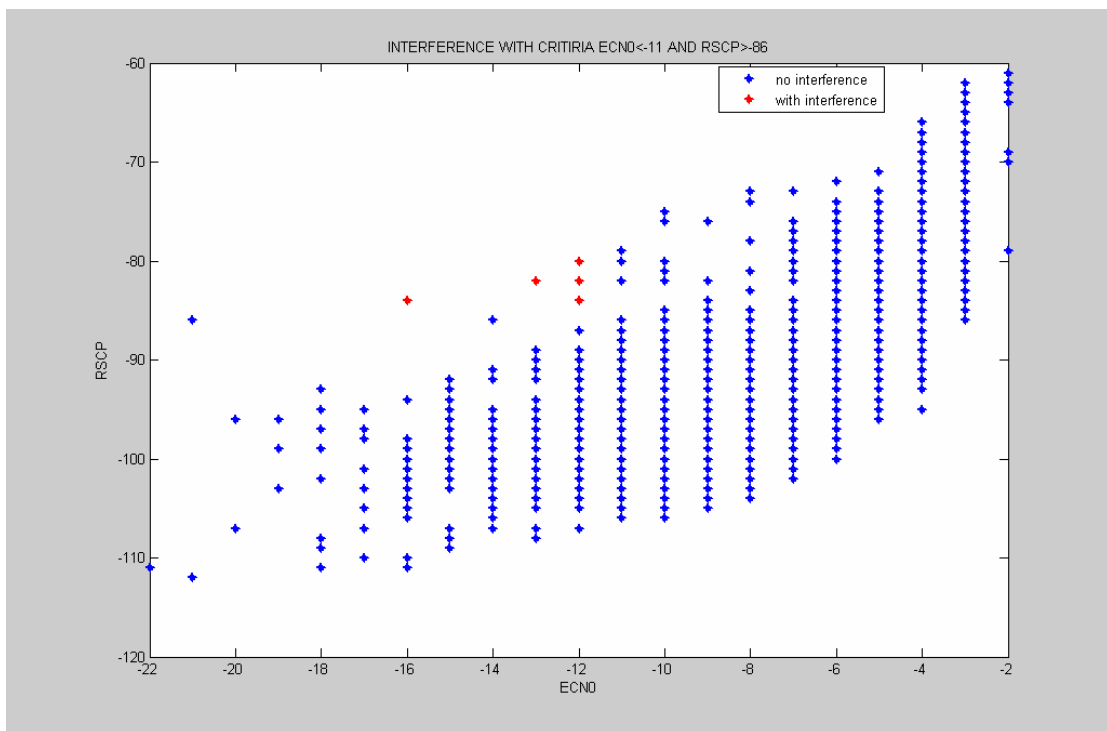
ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ ΤΗΣ ΔΙΑΔΡΟΜΗΣ



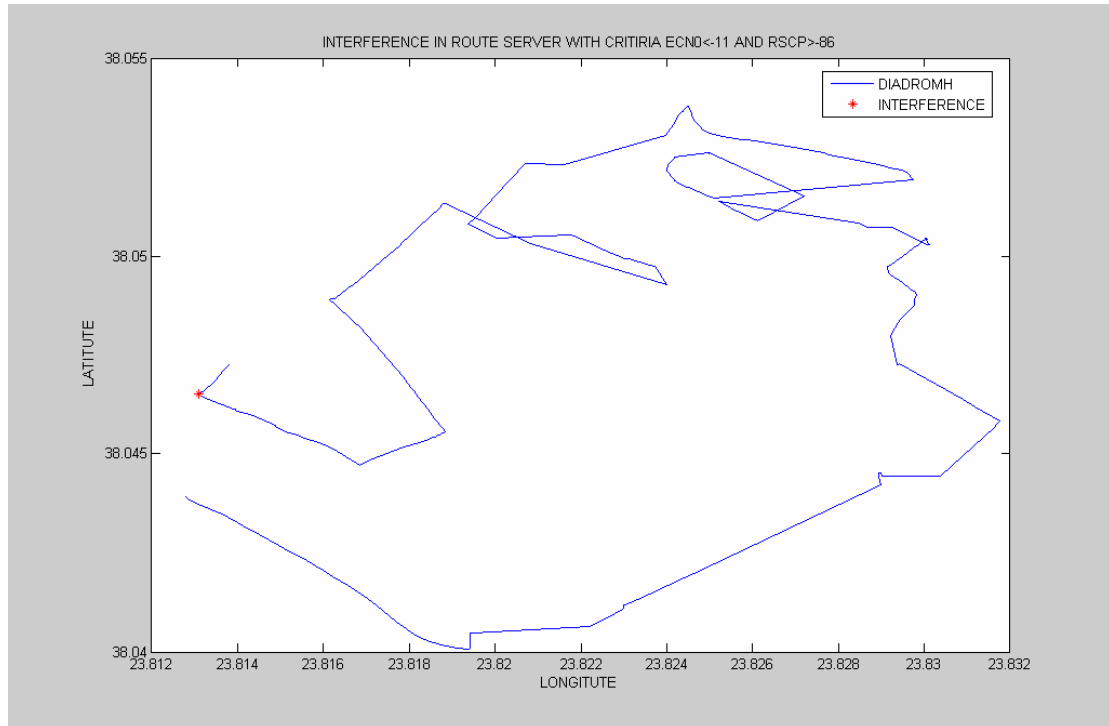
ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ RSCP SERVER.



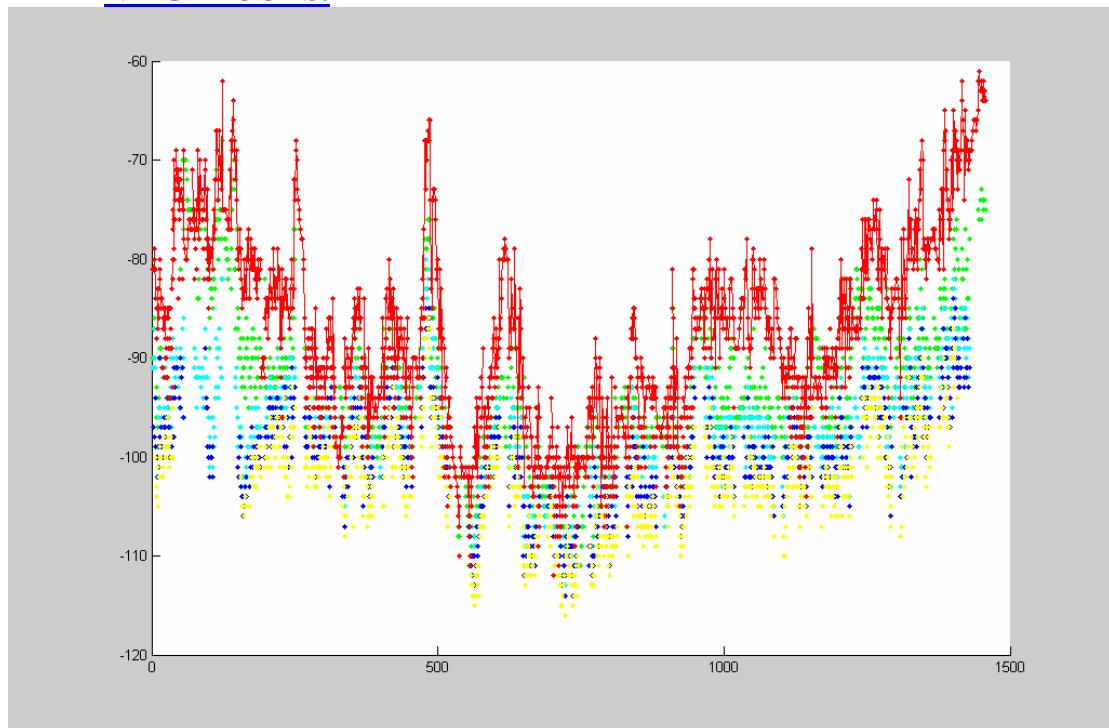
ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ INTERFERENCE (ΠΑΡΕΜΒΟΛΗΣ) SERVER.



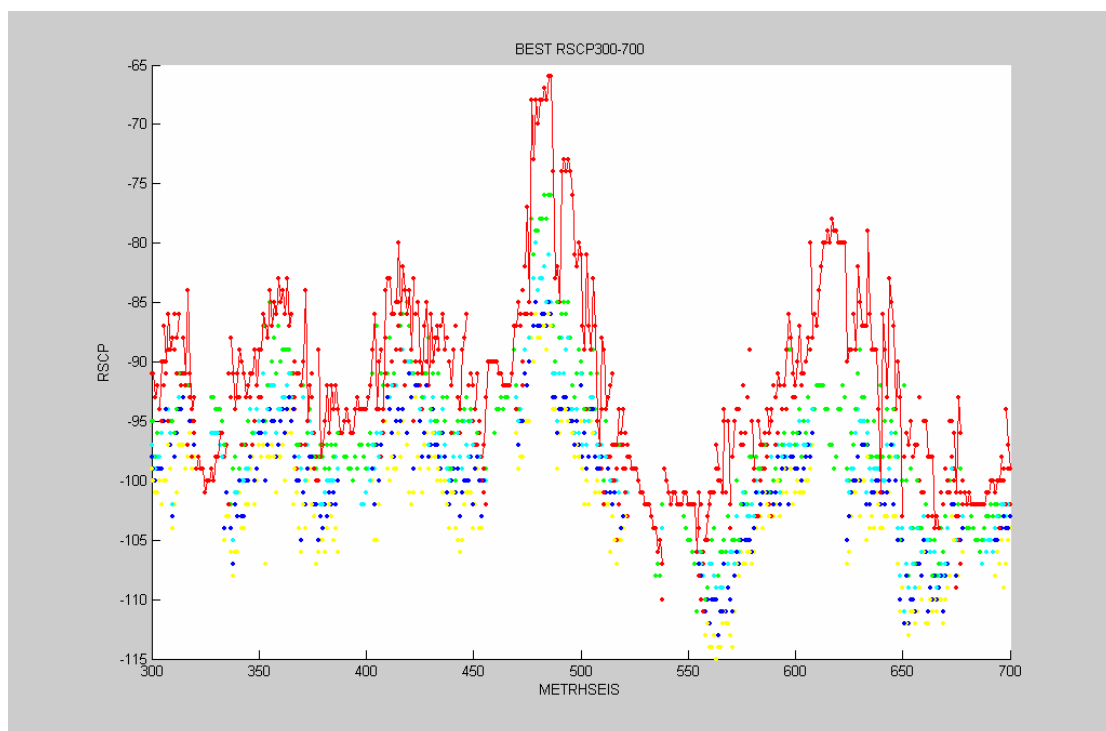
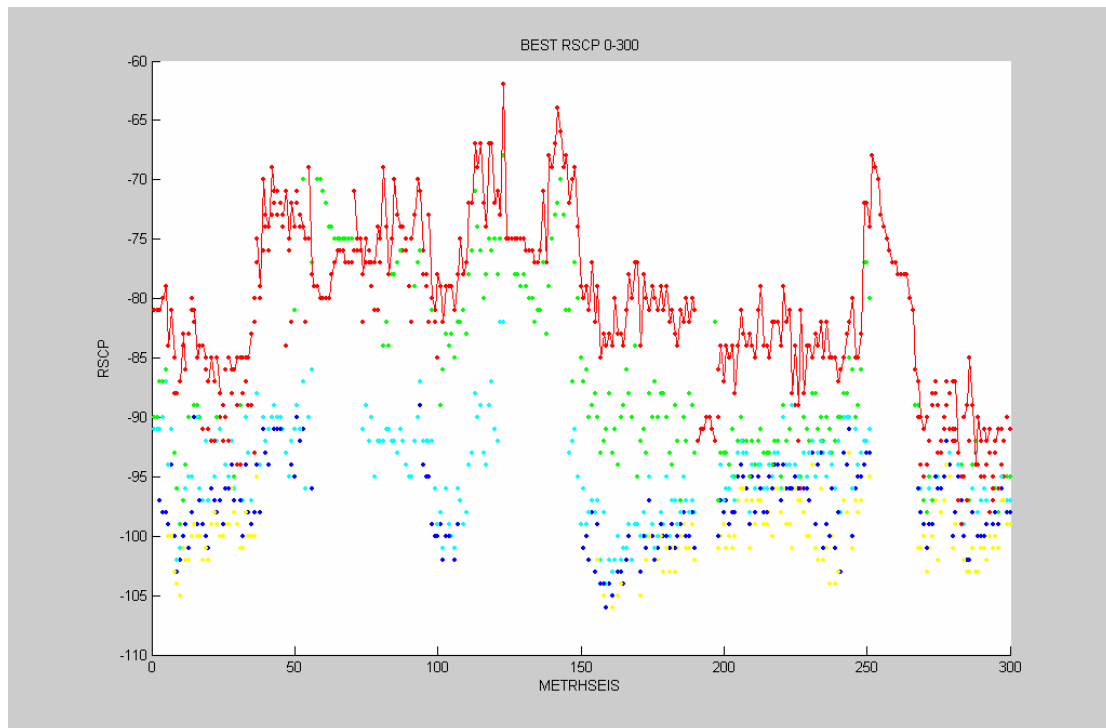
**ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ INTERFERENCE (ΠΑΡΕΜΒΟΛΗΣ)
SERVER ΠΑΝΩ ΣΤΗ ΔΙΑΔΡΟΜΗ(IN ROUTE).**



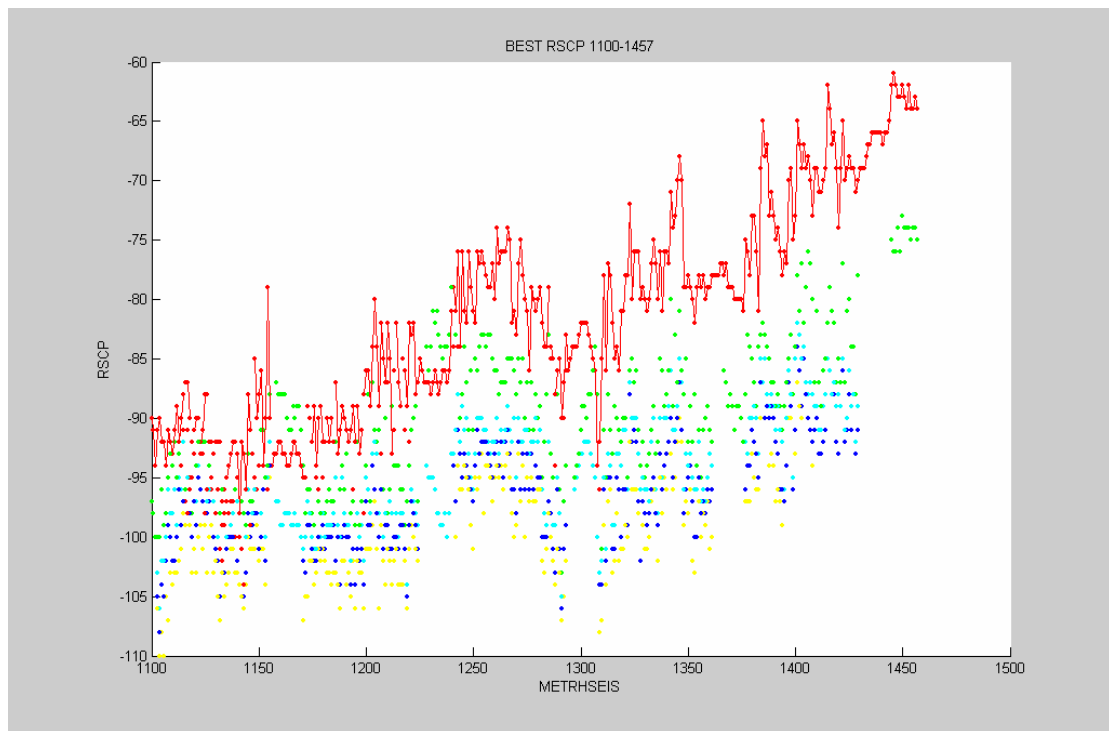
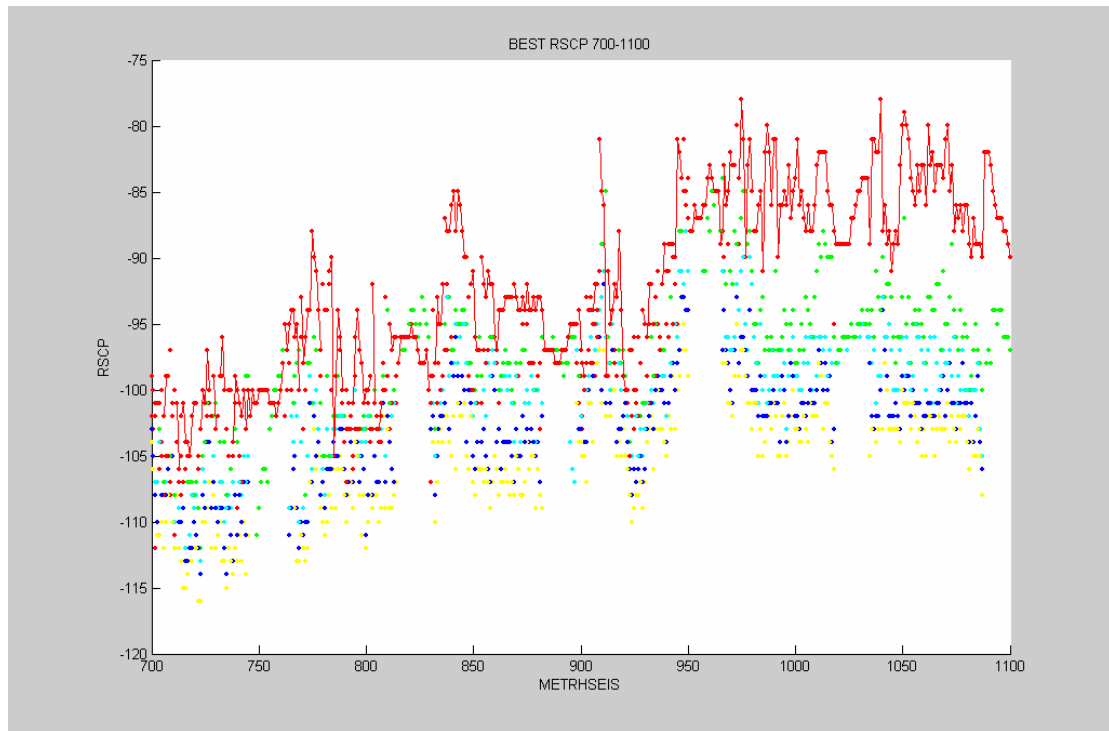
**ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ BEST SERVERS ΚΑΙ BEST
NEIGHBOURS.**



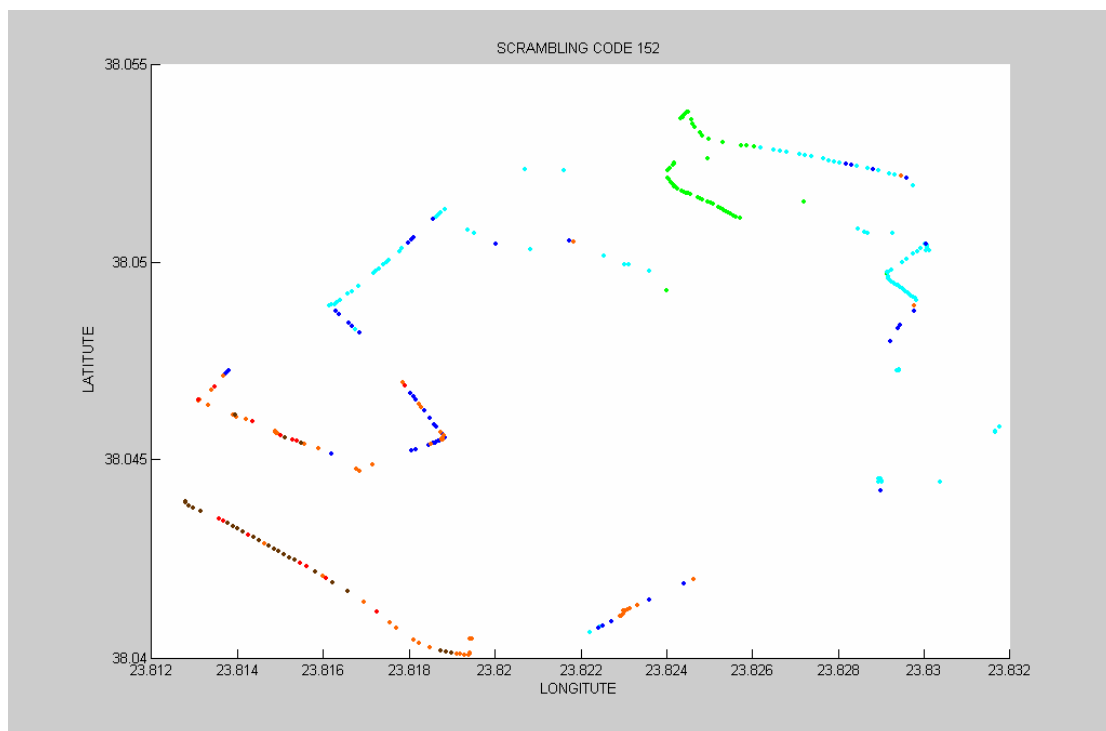
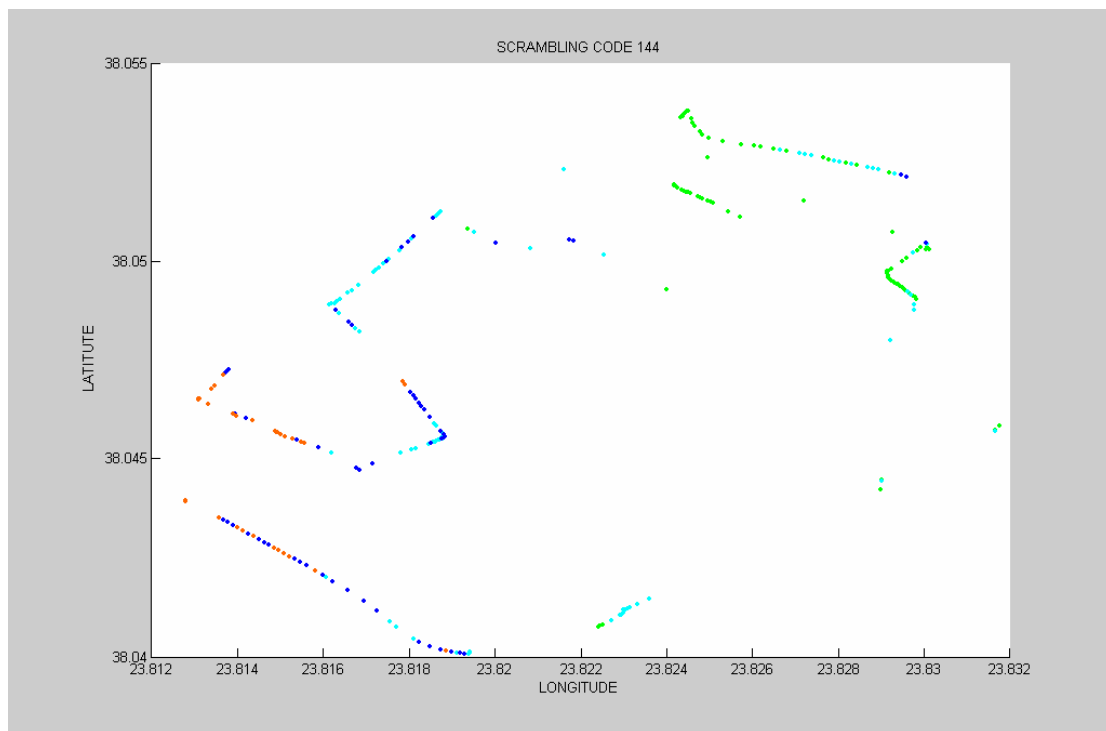
ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)



ΑΝΑΛΥΣΗ ΜΕΤΡΗΣΕΩΝ ΚΙΝΗΤΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΡΙΤΗΣ ΓΕΝΙΑΣ (3G)



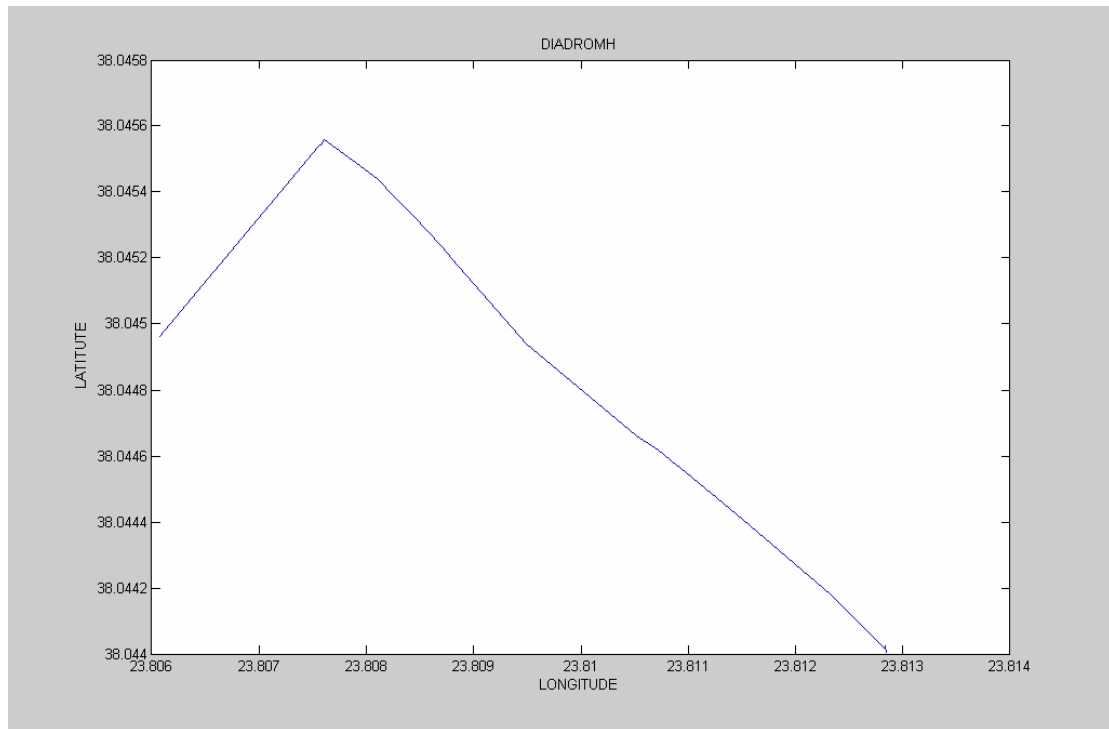
ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ SCRAMBLING CODE.



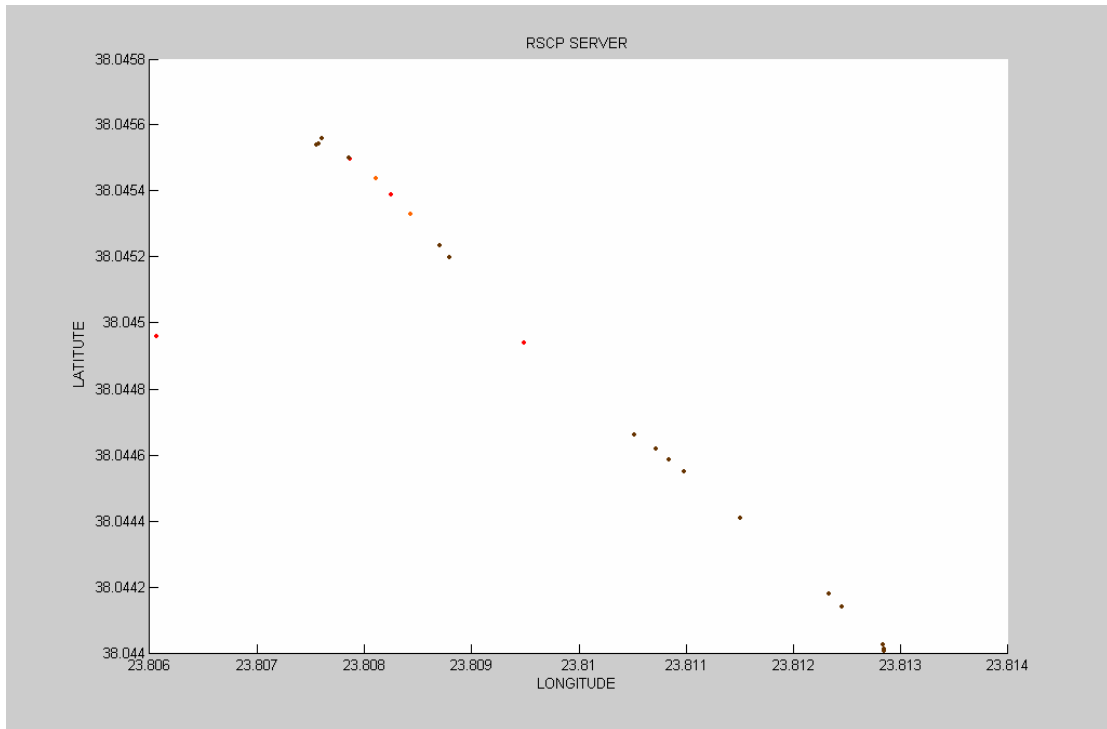
- Για το αρχείο f003:

Για τον τρόπο δημιουργίας γραφικών χρησιμοποιούμε την ίδια διαδικασία με αυτή του αρχείου F001 και τα συμπεράσματα αυτών είναι τα ίδια με τα προηγούμενα.

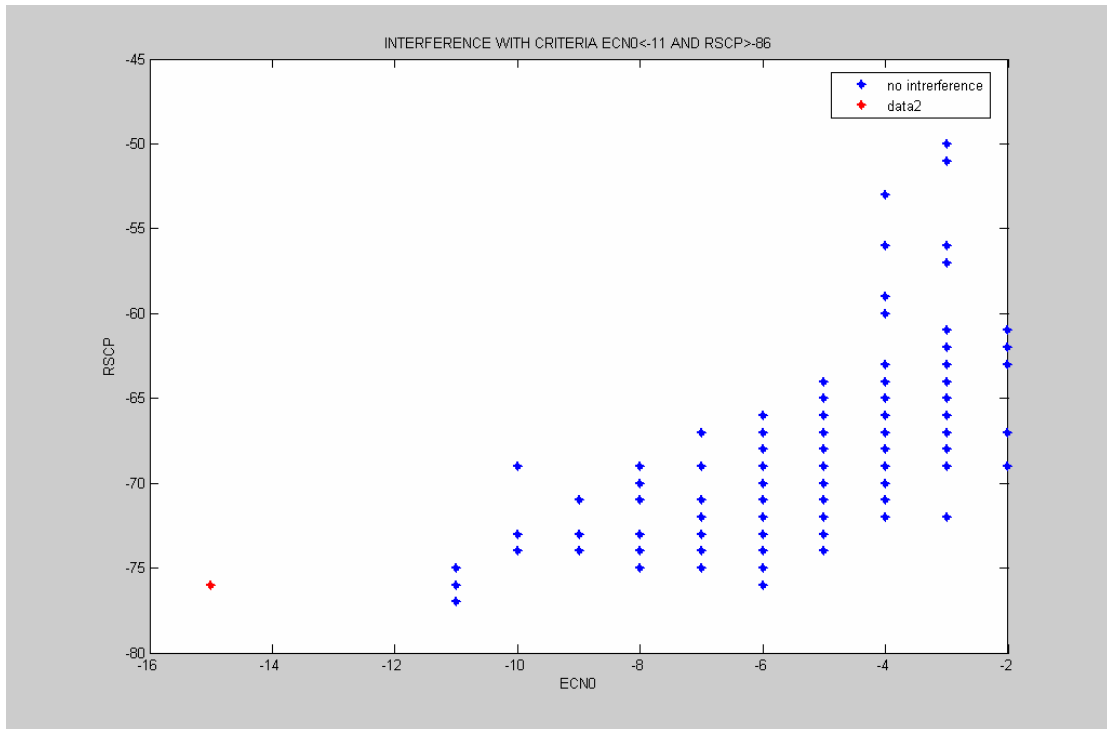
ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ ΤΗΣ ΔΙΑΔΡΟΜΗΣ



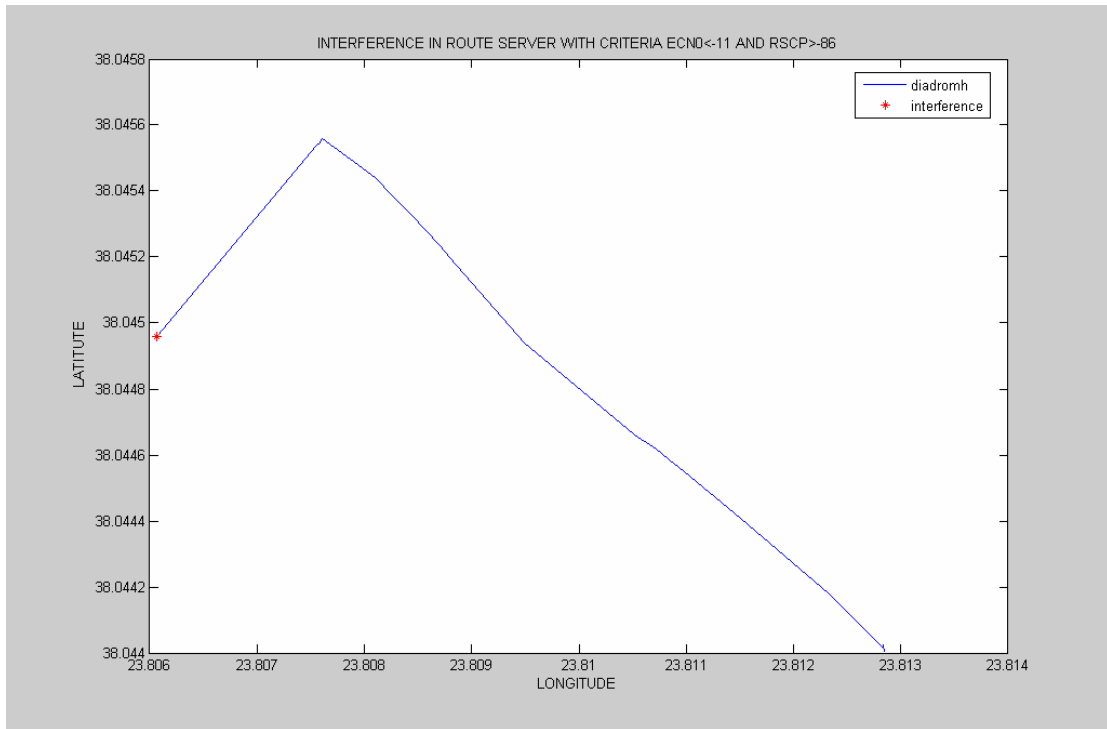
ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ RSCP SERVER.



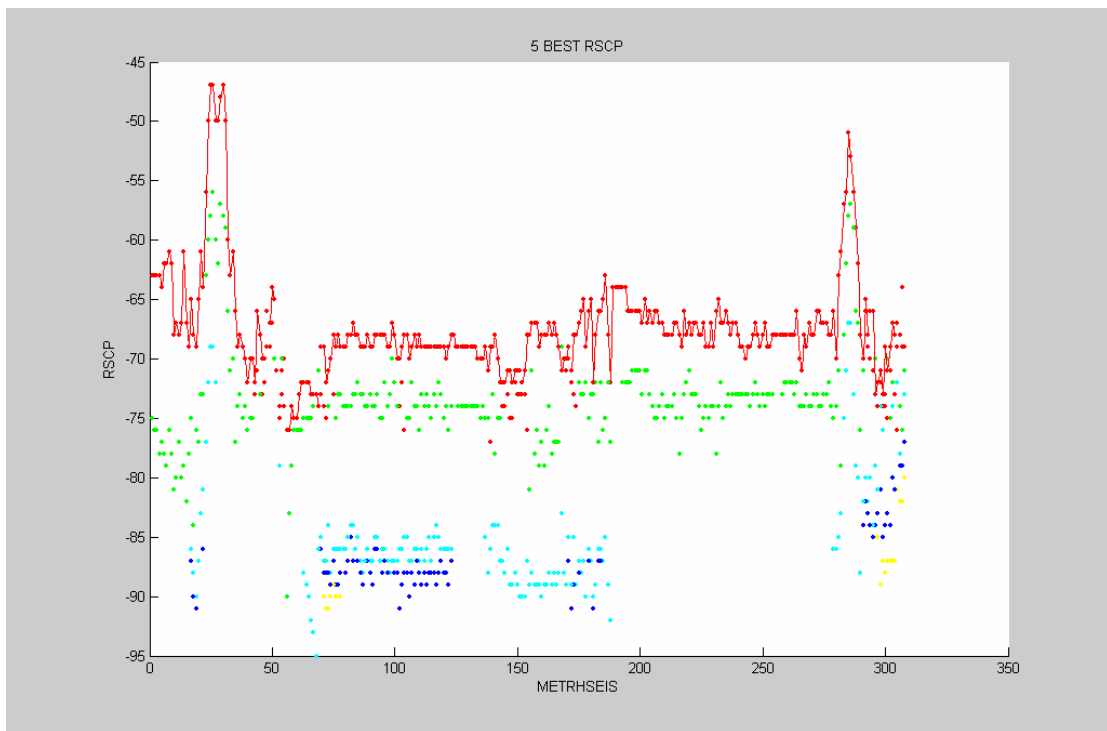
ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ INTERFERENCE (ΠΑΡΕΜΒΟΛΗΣ) SERVER.



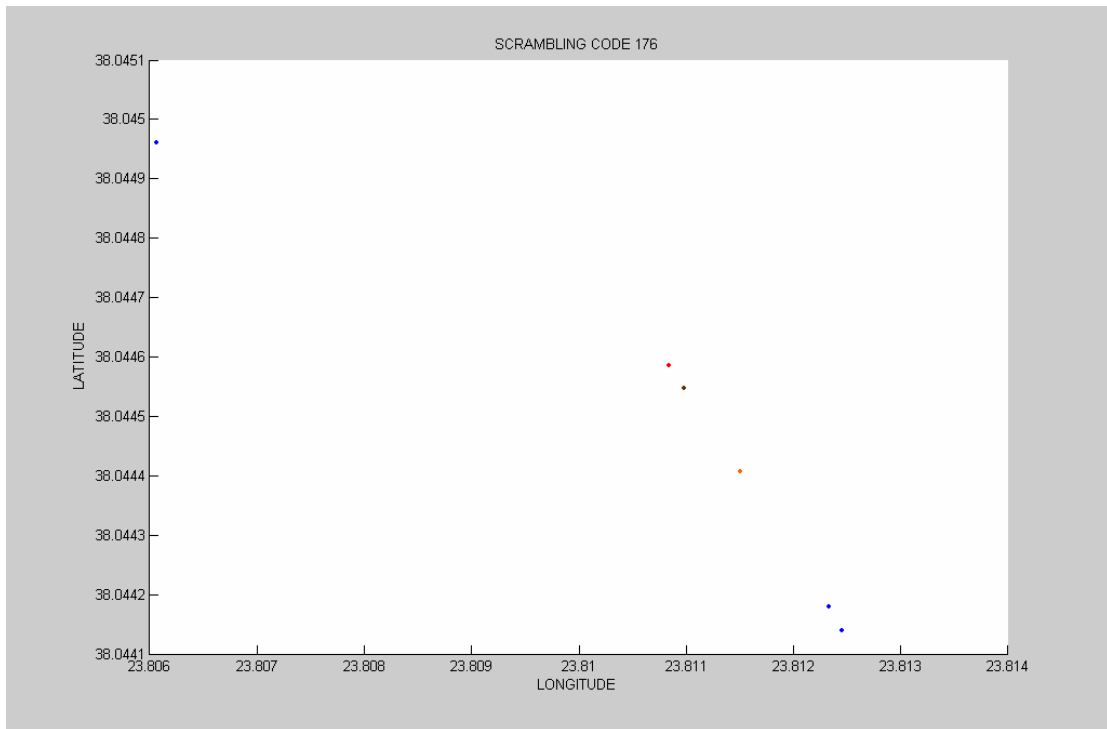
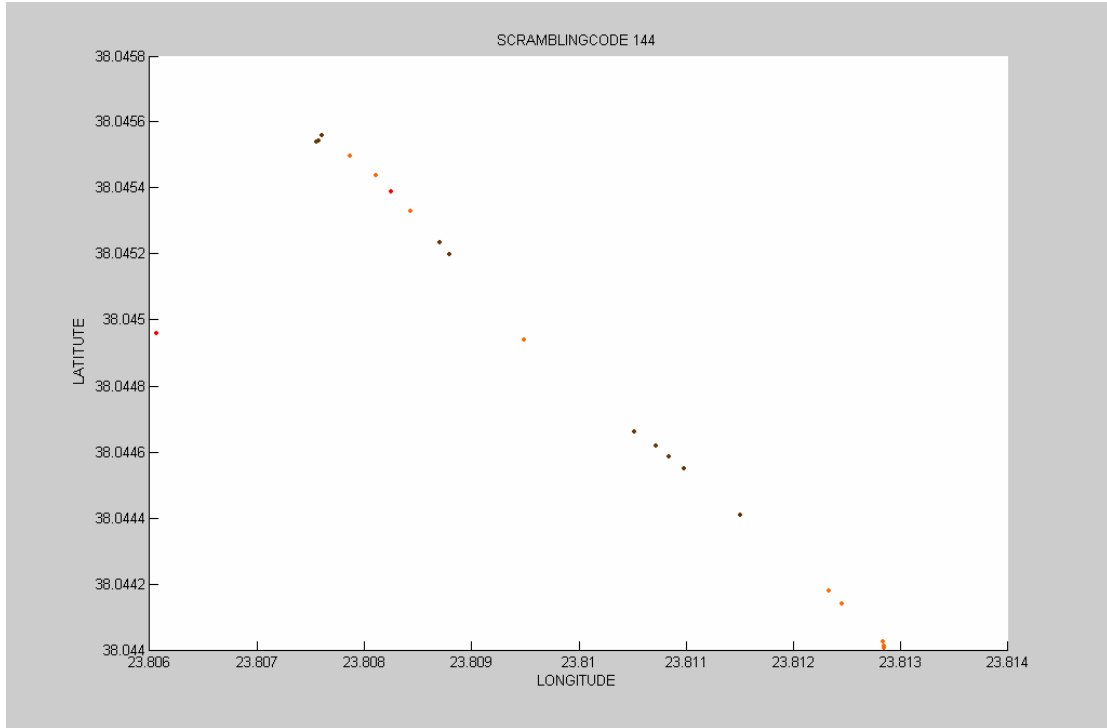
**ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ INTERFERENCE (ΠΑΡΕΜΒΟΛΗΣ)
SERVER ΠΑΝΩ ΣΤΗ ΔΙΑΔΡΟΜΗ(IN ROUTE).**



**ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ BEST SERVERS ΚΑΙ BEST
NEIGHBOURS.**



ΔΗΜΙΟΥΡΓΙΑ ΓΡΑΦΙΚΗΣ SCRAMBLING CODE.



ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Bob Reselman, Richard Peasley, Wayne Pruchniak **Visual Basic 6** Εκδόσεις Γκιούρδας.
2. Nemo Technologies **File Format Specification**
3. Πηγές INTERNET
4. **Help** Προγραμμάτων **Visual Basic 6** και **MATLAB 7**

Στο σημείο αυτό θα θέλαμε να ευχαριστήσουμε τον επιβλέπων καθηγητή της πτυχιακής μας κ.Ευάγγελο Κόκκινο καθώς και τον συμφοιτητή μας Εμμανουήλ Μιχαλοδημητράκη για την πολύτιμη βοήθεια του.