



Τ.Ε.Ι. ΚΡΗΤΗΣ

ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ

Τμήμα Ηλεκτρονικής

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

*“Μελέτη και κατασκευή έξυπνου
συστήματος ελέγχου Ηλεκτρικών φορτίων
μέσω δικτύου κινητής τηλεφωνίας ”*

*του σπουδαστή
Κοκολάκη Ιωάννη*

Εισηγητής :Ρηγάκης Ηρακλής

ΠΕΡΙΛΗΨΗ

Το αντικείμενο της παρούσας πτυχιακής εργασίας είναι η κατασκευή ενός έξυπνου συστήματος που θα χρησιμοποιηθεί στα σπίτια για τον έλεγχο ηλεκτρικών φορτίων μέσω δικτύου κινητής τηλεφωνίας.

Το σύστημα αποτελείται από τρεις κύριες μονάδες. Η πρώτη μονάδα είναι ένας επεξεργαστής 8051 για τον έλεγχο του όλου συστήματος. Η δεύτερη μονάδα είναι ένα κινητό τηλέφωνο όπου με την ανταλλαγή sms θα γίνεται η επικοινωνία με τον επεξεργαστή. Και τέλος υπάρχει η μονάδα της πλακέτας ισχύος όπου με τις κατάλληλες εντολές που θα λαμβάνει ο επεξεργαστής θα ενεργοποιεί και τις κατάλληλες εξόδους.

Με την εφαρμογή αυτής της πτυχιακής ο χρήστης θα έχει την δυνατότητα να ενεργοποιεί ή να απενεργοποιεί κάποιες συσκευές του στο σπίτι με την αποστολή ενός απλού γραπτού μηνύματος, όπου στην συνέχεια ο χρήστης θα λαμβάνει ένα μήνυμα απάντησης με την κατάσταση που βρίσκονται οι διακόπτες εκείνη την χρονική στιγμή. Ακόμα σε περίπτωση που ο χρήστης λείπει από το σπίτι του και κάποιος πατήσει το κουδούνι της εξώπορτας, η συσκευή μας έχει την δυνατότητα να κάνει κλήση και να μιλάει απευθείας ο ιδιοκτήτης με το άλλο πρόσωπο που τον ψάχνει.



*Technological Education Institute (T.E.I) of crete
Branch of Chania
Department of Electronics*

*“Study and construction of a smart
control system of electrical charges
via mobile telephony nets”*

Graduate Thesis
of
Giannis kokolakis

Supervisor: Iraklis Rigakis

CHANIA 2006

CD-ROM INCLUDED

Abstract

The object of the present thesis is the construction of a smart system which will be used at home for the control of electrical charges via mobile telephony nets.

The system consists of three main units. The first one is a micro controller 8051 who controls the whole system. The second unit is a cell phone which communicates with the controller by exchanging written message (sms). The last one is a unit of a printed circuit which will enable the right outputs by the proper orders which the controller will receive.

By making this project work, the user will have the potential to enable or disable home devices by sending a written message (sms) then, the user will receive a report which will show if the switches are at the moment on or off. Lastly, if someone rings the doorbell and the user is absent the device gives a signal to his cell phone and he can automatically communicate with the person that is looking for him.



Τ.Ε.Ι. ΚΡΗΤΗΣ

ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ

Τμήμα Ηλεκτρονικής

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

“Μελέτη και κατασκευή έξυπνου
συστήματος ελέγχου Ηλεκτρικών φορτίων
μέσω δικτύου κινητής τηλεφωνίας ”

του σπουδαστή
Κοκολάκη Ιωάννη



ΧΑΝΙΑ 2006



Τ.Ε.Ι. ΚΡΗΤΗΣ

ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ

Τμήμα Ηλεκτρονικής

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

*“ Μελέτη και κατασκευή έξυπνου
συστήματος ελέγχου Ηλεκτρικών φορτίων
μέσω δικτύου κινητής τηλεφωνίας ”*

*του σπουδαστή
Κοκολάκη Ιωάννη*

Εισηγητής : Ρηγάκης Ηρακλής

ΧΑΝΙΑ 2006

*ΤΗΝ ΠΑΡΟΥΣΑ ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ ΤΗΝ
ΑΦΙΕΡΩΝΩ ΣΤΗΝ ΟΙΚΟΤΕΝΕΙΑ ΜΟΥ*

ΓΙΑΝΝΗΣ

Θα ήθελα να ευχαριστήσω:

*Τον κ. Φηγάκη Ηρακλή, για
την αμέριστη συμπαράσταση του,
και την επιμελή καθοδήγηση του
καθώς και τις απεριόριστες ώρες
που αφιέρωσε μαζί μου.*

*Τους γονείς μου για ότι μου έχουν
προσφέρει και ότι θα μου
προσφέρουν ακόμα.*

*Τον κ. Παπαδάκη Γιάννη για την
βοήθεια του.*

*Τον φίλο και συνάδερφο
Κυριαζάκο Στέφανο για την
βοήθεια του και τις χρήσιμες
παρατηρήσεις του.*

*Καθώς και την φίλη μου και
Συνάδερφο **Κεχηρή Ματίνα**
για την βοήθεια της.*

ΠΕΡΙΕΧΟΜΕΝΑ

Αφιερώσεις.....	3
Ευχαριστίες.....	4
Πρόλογος.....	6
Κεφάλαιο 1	
1.1 Εφαρμογές του έξυπνου συστήματος.....	7
Κεφάλαιο 2	
2.1 Δίκτυα GSM.....	8-10
2.2 GSM μόντεμ.....	11-14
2.3 AT εντολές.....	15-29
Κεφάλαιο 3	
3.1 Ο DS89C420.....	30-32
3.2 Ο MAX232.....	33
Κεφάλαιο 4	
4.1 Μπλοκ διάγραμμα της κατασκευής.....	34
4.2 Διάγραμμα ροής του προγράμματος.....	35-36
4.3 Η κατασκευή μας.....	37-38
Κεφάλαιο 5	
5.1 Τα σχηματικά και τα PCB της κατασκευής μας.....	39-44
5.2 Τα σχηματικά του κυκλώματος τροφοδοσίας, καθώς και το βασικό σχηματικό της πλακέτας ισχύος.....	45-46
5.3 Διάγραμμα ροής υπό ρουτίνας διαβάσματος του μηνύματος.....	47-49
5.4 Διάγραμμα ροής υπό ρουτίνας αποθήκευσης του μηνύματος.....	50-51
Βιβλιογραφία.....	52
Κώδικας.....	53-69

ΠΡΟΛΟΓΟΣ

«...Είναι πολύ επώδυνο και ψυχοφθόρο με την πίεση χρόνου όπου έχουν οι άνθρωποι στις μέρες μας να γυρίζεις το απόγευμα στο σπίτι σου κουρασμένος από την δουλειά και να περιμένεις καθημερινά μισή ώρα ώσπου να ζεσταθεί το νερό στο θερμοσίφωνα. Γυρίζοντας αργά στο σπίτι να έχεις ενεργοποιήσει την κεντρική θέρμανση και να μην περιμένεις εκείνη την στιγμή να ζεσταθεί το σπίτι. Όπως και αν είσαι κάπου μακριά και να έχεις ξεχάσει ν' ανάψεις ένα εξωτερικό φως και να μην νοιώθεις άνετα μην τυχόν και σας κλέψουν...»

Τέτοια και άλλα πολλά προβλήματα απασχολούν ένα μεγάλο μέρος των εργαζομένων σήμερα. Η ρουτίνα της καθημερινότητας και ο περιορισμένος χρόνος των ανθρώπων είναι η αιτία της φόρτισης και τη μη απόλαυση της φυσιολογικής ζωής.

Στόχος της πτυχιακής αυτής είναι η κατασκευή συσκευής που θα πραγματοποιεί αυτόματα τις παραπάνω βασικές λειτουργίες ηλεκτρικών συσκευών. Τις εντολές για την εκτέλεση των εργασιών αυτών θα γίνετε από οπουδήποτε. Αυτό μπορεί να γίνει μόνο με ένα ασύρματο τηλεπικοινωνιακό δίκτυο παγκόσμιας εμβέλειας όπως είναι το δίκτυο κινητής τηλεφωνίας, με την χρήση ενός μικροελεκτή και ένα GSM μόντεμ, καθώς και πολλών άλλων υποσυστημάτων που περιγράφονται παρακάτω.

Η συσκευή που κατασκευάστηκε και περιγράφεται στα επόμενα κεφάλαια μπορεί να διαχειρίζεται ηλεκτρικούς διακόπτες (π.χ. τα Φώτα, το θερμοσίφωνα, το πότισμα του κήπου κ.α.), καθώς και ηλεκτρονικούς διακόπτες (A/C, την πόρτα του γκαράζ κ.α.) ενός σπιτιού από οποιοδήποτε μέρος του κόσμου με την χρήση απλών μηνυμάτων από οποιοδήποτε κινητό τηλέφωνο.

Ακόμα σε περίπτωση που κάποιος πατήσει το κουδούνι της εξώπορτας μας η συσκευή αυτή έχει την δυνατότητα και μας κάνει αυτόματα κλήση, έτσι ώστε να μπορέσουμε να επικοινωνήσουμε εκείνη την στιγμή και να μιλήσουμε με τον άνθρωπο που βρίσκετε έξω από το σπίτι μας.

ΚΕΦΑΛΑΙΟ 1

1.1 ΕΦΑΡΜΟΓΕΣ ΤΗΣ ΕΡΓΑΣΙΑΣ

Οι άνθρωποι στην εποχή μας εργάζονται συνέχεια και σε εξαντλητικά ωράρια ώστε να μπορούν να τα φέρουν πέρα στις καθημερινές ανάγκες που απαιτεί η ζωή. Και δεν φτάνει αυτό έχουν και διάφορα αλλά άγχη όσον αφορά τις ανάγκες του σπιτιού ή της επιχείρησής τους με αποτέλεσμα να υπάρχει υπερβολική φόρτιση, άγχος και στρες και να ξεχνάνε πολύ σημαντικά πράγματα που είναι απαραίτητα να γίνουν σε καθημερινή βάση.

Με την υλοποίηση της συσκευής αυτής οι άνθρωποι θα έχουν την δυνατότητα με την απλή αποστολή ενός γραπτού μηνύματος (SMS), να μπορούν να καλύπτουν κάποιες από τις βασικές αυτές ανάγκες ενός θα βρίσκονται στην καθιερωμένη καθημερινή τους εργασία χωρίς να έχουν περαιτέρω άγχος.

Η κατασκευή αυτή με τον κατάλληλο χειρισμό μα δίνει την δυνατότητα να μπορούμε να ενεργοποιήσουμε τον ηλεκτρικό *θερμοσίφωνα* του σπιτιού μας, ενώ λείπουμε από το σπίτι μας και καθώς γυρίζουμε να μπορούμε να απολαύσουμε ένα ζεστό μπάνιο, χωρίς να περιμένουμε εκείνη την στιγμή να ζεσταθεί το νερό και να σπαταλάμε περαιτέρω χρόνο.

Ακόμα θα μπορούμε αν κάποια μέρα λείπουμε από το σπίτι μας να ενεργοποιούμε ή να απενεργοποιούμε κάποιο εσωτερικό ή εξωτερικό φως, καθώς επίσης θα μπορούμε και να ενεργοποιούμε / απενεργοποιούμε το σύστημα ασφαλείας του σπιτιού μας χωρίς να χρειάζεται να επιστρέψουμε από κάπου μακριά για να το κάνουμε, ή να στείλουμε κάποιο άλλο γνωστό μας πρόσωπο για να το κάνει. Με αποτέλεσμα να γλιτώνουμε και χρόνο και κόπο, και να μην γινόμαστε βαρετοί σε γνωστούς μας.

Καθώς επίσης κατά τους καλοκαιρινούς μήνες και σε περίοδο καύσωνα θα μπορούμε από την δουλειά μας ή όπου αλλού βρισκόμαστε, και λίγο πριν πάμε σπίτι να ενεργοποιήσουμε το air contusion (A/C) του σπιτιού μας, με ένα (SMS) ώστε την ώρα που θα επιστρέψουμε από την δουλειά να έχουμε ένα ωραίο και δροσερό περιβάλλον, για να μπορέσουμε να καθίσουμε να ξεκουραστούμε χωρίς να έχουμε τις υπερθερμίες του καύσωνα.

Κατά τους χειμερινούς μήνες όπου το κρύο είναι υπερβολικά πολύ και επειδή δεν μπορούμε να έχουμε την κεντρική θέρμανση του σπιτιού μας ανοικτή τις ώρες που δεν θα βρισκόμαστε στο σπίτι για λόγους οικονομίας, και όταν πηγαίνουμε σπίτι θα πρέπει να καθόμαστε μέσα στην παγωνιά να περιμένουμε να ζεσταθεί το σπίτι για να μπορέσουμε να απολαύσουμε την οικογενειακή θαλπωρή. Με την υλοποίηση της συσκευής αυτής θα μπορούμε απ' όπου και αν είμαστε, και με ένα απλό μήνυμα να ανάψουμε την κεντρική θέρμανση, έτσι ώστε γυρίζοντας στο σπίτι να έχουμε ένα ζεστό περιβάλλον.

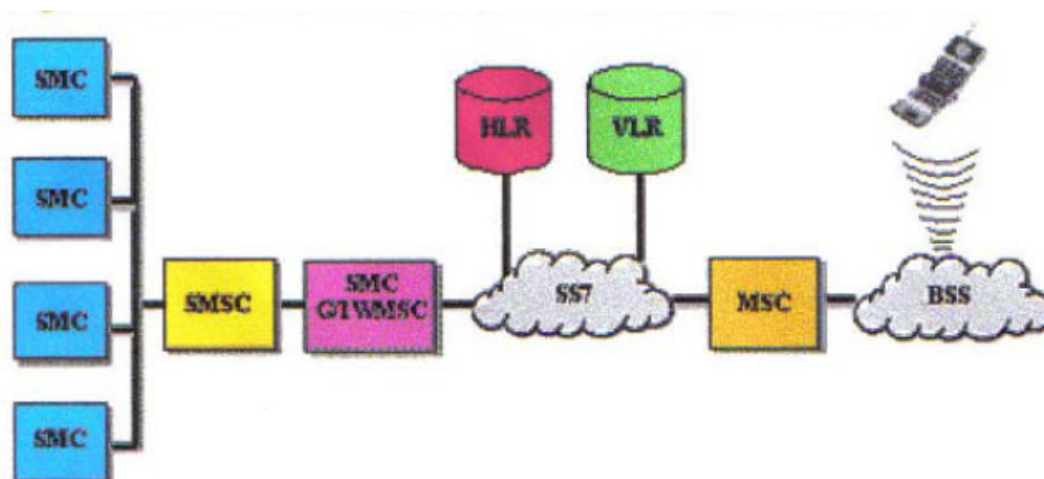
Επίσης θα έχουμε την δυνατότητα κατά την καλοκαιρινή περίοδο όπου τα φυτά χρειάζονται καθημερινά πότισμα, να μπορούμε να το πράξουμε απ' όπου και αν βρισκόμαστε με την αποστολή ενός γραπτού μηνύματος, θα ενεργοποιείτε το μοτέρ ώστε το πότισμα να εκτελείτε αυτόματα, και όταν εμείς επιθυμούμε θα μπορούμε να το απενεργοποιήσουμε χωρίς να χρειάζεται να είμαστε εμείς εκεί, και να χάνουμε χρόνο ή να πρέπει να φύγουμε από την εργασία μας για να το κάνουμε αυτό.

ΚΕΦΑΛΑΙΟ 2

2.1 ΔΙΚΤΥΑ GSM

Το πρότυπο κινητής τηλεφωνίας GSM (Global System for Mobile communications) δημιουργήθηκε από τον Ευρωπαϊκό Οργανισμό για τα ταχυδρομεία και τις τηλεπικοινωνίες (Conference of European Postal and Telecommunications - CEPT). Το πρότυπο GSM ακολουθεί την κυψελωτή αρχιτεκτονική δικτύου. Κάθε κυψέλη διαθέτει ένα σταθμό βάσης με κατάλληλη κεραία που καλύπτει όλη την έκταση της. Ο σταθμός βάσης συνδέεται ασύρματα με τις κινητές συσκευές και ενσύρματα με το κέντρο μεταγωγής του συστήματος. Το βασικό στοιχείο του GSM είναι η μετατροπή του σήματος φωνής σε ψηφιακό σήμα και η μετάδοση του σε συχνότητες UHF (Ultra High Frequencies) και συγκεκριμένα γύρω από τα **900 MHz**, **1800MHz** με κανάλια εύρους ζώνης 200 KHz. Ο κάθε συνδρομητής επιτρέπεται να κινείται τόσο μέσα στην ίδια κυψέλη όσο και μεταξύ διαφορετικών κυψελών, χωρίς να υπάρχει διακοπή της επικοινωνίας, ακόμα και όταν ο χρήστης κινείται με ταχύτητα 140 χιλιόμετρα την ώρα. Το κανάλι επικοινωνίας που χρησιμοποιείται κάθε φορά από ένα κινητό ορίζεται από το σταθμό βάσης. Ο τελευταίος μπορεί να χειρίζεται ταυτόχρονα μεγάλο αριθμό καναλιών με τη χρήση τεχνικής FDMA (Frequency Division Multiple Access). Ο ακριβής αριθμός καθορίζεται κατά τη σχεδίαση του δικτύου ανάλογα με τις ανάγκες που προβλέπεται να καλύψει. Επιπλέον σε κάθε κανάλι επικοινωνίας μπορούν να μιλούν ταυτόχρονα μέχρι οκτώ κινητά με χρήση της TDMA (Time Division Multiple Access). Έτσι το κινητό δεν εκπέμπει συνεχώς αλλά μία φορά κάθε 4,615 msec ή ισοδύναμα 217 φορές το δευτερόλεπτο.

Η βασική δομή του δικτύου για την παροχή SMS φαίνεται στο παρακάτω σχήμα:



Σχήμα 2.1

- **Short Messaging Capability (SMC)**

Το (SMC) είναι οτιδήποτε μπορεί να στείλει ή να λάβει μηνύματα. Το (SMC) μπορεί να αφορά συσκευή ενός κινητού δικτύου ή ενός σταθερού τηλεπικοινωνιακού δικτύου (κινητό τηλέφωνο, modem της παρούσας κατασκευής, H/Y, κ.λ.π).

- **Short Message Service Center (SMSC)**

Το (SMSC) είναι υπεύθυνο για την αναμετάδοση, την αποθήκευση και την προώθηση των μηνυμάτων μεταξύ του SME και του κινητού σταθμού.

- **SMS-Gateway/ Internet working Mobile Switching Center (SMS-G/IWMSC)**

Το (SMS-GMSC) είναι ένα κινητό κέντρο μεταγωγής (MSC) ικανό για λήψη ενός μηνύματος από το SMSC, επικοινωνίας με τον HLR για την λήψη πληροφοριών δρομολόγησης και για μεταφορά του μηνύματος στον (MSC) του αποδέκτη μεταγωγής.

Το (SMS-IWMSC) είναι ένα κινητό κέντρο μεταγωγής ικανό για την λήψη ενός μηνύματος από το κινητό δίκτυο και παράδοση αυτού στο κατάλληλο (SMSC).

- **Home Location Register (HLR)**

Ο HLR είναι μια βάση δεδομένων για μόνιμη αποθήκευση και διοίκηση των συνδρομητικών υπηρεσιών. Σύμφωνα με την ερώτηση του (SMSC) ο HLR εξασφαλίζει την πληροφορία δρομολόγησης για τον ενδεικνύμενο συνδρομητή.

- **Mobile Switching Center (MSC)**

Ο MSC εκτελεί τις λειτουργίες μεταγωγής του συστήματος και ελέγχει κλήσεις από και προς άλλα τηλέφωνα και βάσεις δεδομένων.

- **Visitor Location Register (VLR)**

Ο VLR είναι μια βάση δεδομένων η οποία περιέχει προσωρινές πληροφορίες για τους συνδρομητές. Αυτές οι πληροφορίες είναι χρήσιμες για το (MSC) στην εξυπηρέτηση των συνδρομητών.

- **The Base Station System (BSS)**

Όλες οι ασύρματες λειτουργίες εκτελούνται στη BSS. Η BSS αποτελείται από (BSCs) και (BTSs) και η κύρια αρμοδιότητα του είναι η μεταφορά φωνής και δεδομένων μεταξύ των κινητών σταθμών.

- **The Mobile Station (MS)**

Ο MS είναι το ασύρματο τερματικό ικανό για την λήψη και την αρχειοθέτηση των μηνυμάτων όπως στις κλήσεις φωνής. Το ασύρματο δίκτυο υποδομής σηματοδοσίας είναι βασισμένο στο σύστημα 7 (SS7). Το SMS κάνει χρήση του Mobile Application Part (MAP) το οποίο καθορίζει τις μεθόδους και τους μηχανισμούς της επικοινωνίας στα ασύρματα δίκτυα και χρησιμοποιεί τις υπηρεσίες του SS7 Transaction Capability Application Part (TCAP). Ένα επίπεδο υπηρεσιών SMS κάνει χρήση των δυνατοτήτων σηματοδοσίας του MAP και καθιστά δυνατή την μεταβίβαση των μηνυμάτων μεταξύ των ιδίων σημείων.

Τα πλεονεκτήματα του GSM σε σχέση με τα αναλογικά συστήματα κινητής τηλεφωνίας:

Καλύτερη εκμετάλλευση του φάσματος και άρα μεγαλύτερη χωρητικότητα καναλιών σε κάθε κυψέλη.

Ψηφιακή τεχνολογία που κάνει μικρότερα, ελαφρύτερα και φθηνότερα τα κινητά τηλέφωνα.

Σημαντικά καλύτερη ποιότητα φωνής .

Συμβατότητα με όλα τα διεθνή πρότυπα και ενσύρματα δίκτυα.

Ευρεία αποδοχή διεθνώς και εξάπλωση, πράγμα που σημαίνει συμβατότητα σε πολλές διαφορετικές χώρες και χαμηλότερο κόστος κατασκευής και λειτουργίας.

Η χρήση δικτύων GSM ξεκίνησε από την Ευρώπη τον Ιούνιο του 1991 και έχει εξαπλωθεί σε όλο τον κόσμο. Στην Ελλάδα λειτουργούν ήδη δύο δίκτυα κινητής τηλεφωνίας GSM (GSM900 GSM1800).

Μια από τις πολλές υπηρεσίες που μας παρέχουν τα δίκτυα κινητής τηλεφωνίας είναι και η αποστολή σύντομων γραπτών μηνυμάτων (Short Message System) ή αλλιώς SMS. Κατά τη λήψη ενός SMS από ένα κινητό τηλέφωνο ή κάποια άλλη GSM συσκευή λαμβάνουμε ένα μήνυμα της ακόλουθης μορφής:

+CMGR: "REC READ", "+30694000000", "05/3/06, 18:39:42+12"

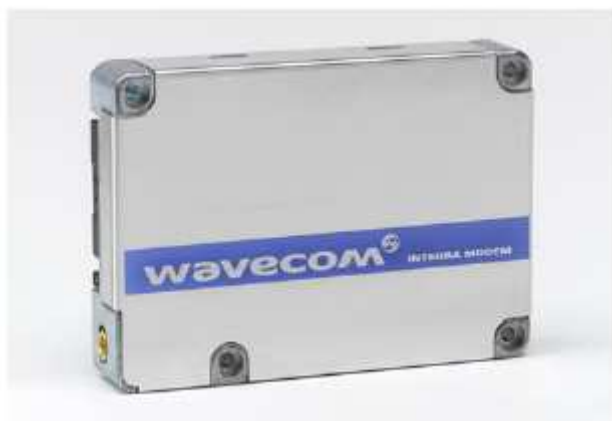
ΚΑΛΗΜΕΡΑ

OK

Στην πρώτη γραμμή διακρίνουμε την κατάσταση του μηνύματος (διαβασμένο ή αδιάβαστο), τον αποστολέα, την ημερομηνία και την ώρα αποστολής του μηνύματος. Στην δεύτερη γραμμή έχουμε το περιεχόμενο του μηνύματος, το οποίο εμφανίζεται στην οθόνη του κινητού μας, που μπορεί να φτάσει τους 160 χαρακτήρες. Στην Τρίτη γραμμή έχουμε την ένδειξη τέλος μηνύματος (OK).

2.2 GSM MODEM

Οι συσκευές που χρησιμοποιούν τα GSM δίκτυα είναι τα GSM μόντεμ. Το GSM μόντεμ είναι μια επέκταση του συνηθισμένου ενσύρματου μόντεμ, με την μόνη διαφορά ότι αυτό το GSM μόντεμ συνδέεται ασύρματα σε οποιοδήποτε δίκτυο κινητής τηλεφωνίας επιθυμούμε. Για να μπορέσει αυτό το GSM μόντεμ να συνδεθεί σε οποιοδήποτε δίκτυο κινητής τηλεφωνίας χρειάζεται η χρήση μιας ενεργοποιημένης κάρτας SIM, καθώς και μια κεραία.

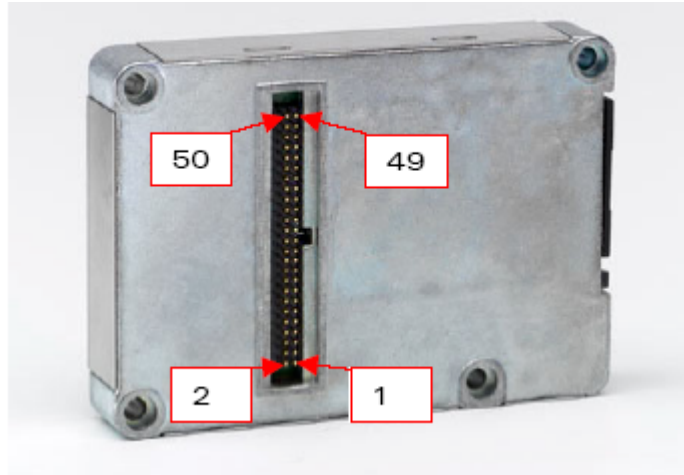


Σχήμα 2.2: Το GSM μόντεμ M2106 που χρησιμοποιήθηκε στο σύστημα



Σχήμα 2.3: Υποδοχή για την κάρτας SIM, και την κεραίας του GSM μόντεμ.

Ένας 50-pin τυποποιημένος συνδετήρας παρέχει τις αναλογικές και ψηφιακές διεπαφές ζωνών βάσης.



Σχήμα 2.4 50-pin συνδετήρας



(1)



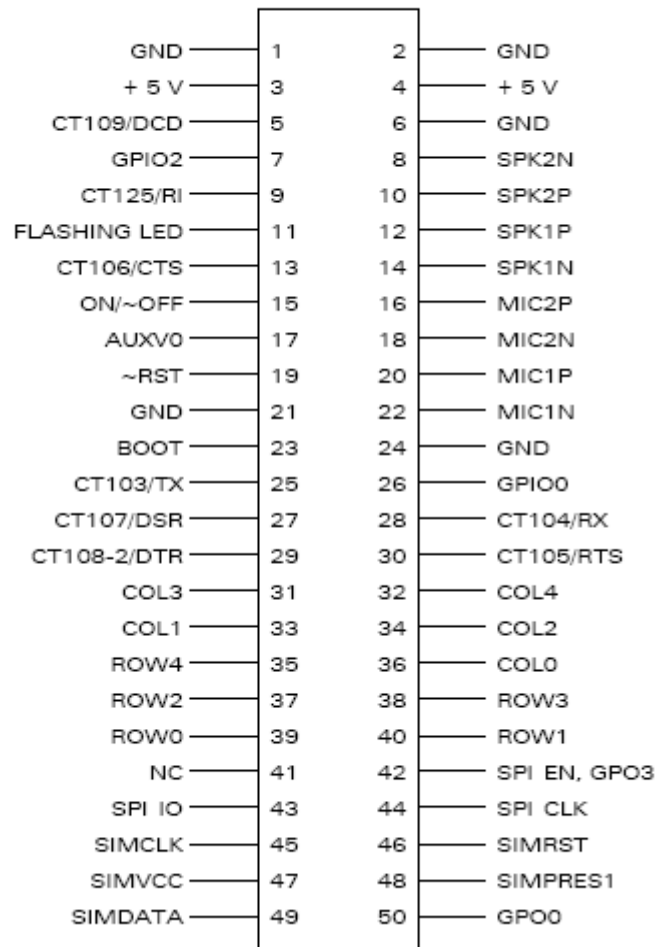
(2)



(3)

Σχήμα 2.5 μερικά από τα είδη κεραιών που μπορούμε να χρησιμοποιήσουμε.

Pin Allocation



Σχήμα 2.5 Η κατανομή των pin του συνδετήρα

Η επικοινωνία των μόντεμ, με τους ηλεκτρονικούς υπολογιστές παλιότερα ήταν κάτι πολύ δύσκολο για τον λόγο ότι κάθε κατασκευαστής επινοούσε τον δικό του τρόπο επικοινωνίας.

Η Αμερικάνικη εταιρία που καθιέρωσε πρώτη ένα συγκεκριμένο σύνολο εντολών είναι η Hayes, το σύνολο αυτό των εντολών είναι κατανοητό και αποδεκτό από όλους, για την επικοινωνία των μόντεμ με τις υπόλοιπες συσκευές. Σήμερα όλα τα μόντεμ είναι συμβατά με τις εντολές αυτές. Όλες οι εντολές Hayes ξεκινούν με το πρόθεμα AT γι' αυτό και ονομάζονται «AT εντολές».

Ένα GSM μόντεμ μπορεί να δεχθεί όλες τις AT εντολές των κλασικών μόντεμ, αλλά λόγω της εξελιγμένης τεχνολογίας του δέχεται περισσότερες και μεταγενέστερες AT εντολές. Με τη βοήθεια των AT αυτών εντολών μπορούμε να διαχειριστούμε τη μνήμη της κάρτας SIM και του μόντεμ και να εκτελέσουμε διάφορες λειτουργίες μέσω του GSM μόντεμ.

Οι AT εντολές των GSM μόντεμ χρησιμοποιούν συντακτικούς κανόνες για εκτεταμένες εντολές. Κάθε εκτεταμένη εντολή έχει ένα test command, για να δοκιμάζει την ύπαρξη των εντολών και να δίνει πληροφορίες σχετικά με τον τύπο των υποπαραμέτρων τους.

2.3 AT- ΕΝΤΟΛΕΣ

Σύντομες εντολές μηνυμάτων

Καθορισμός παραμέτρων

- <da> Διεύθυνση προορισμού, που κωδικοποιείται όπως το GSM 30,40 TP-DA
- <dc> Σχέδιο κωδικοποίησης στοιχείων, που κωδικοποιείται όπως στο έγγραφο [5]
- <dt> Χρόνος απαλλαγής με το σχήμα σειράς :
“yy/MM/dd,hh :mm :ss±_zz”(Έτος [00-99], Μήνας [01-12], Ημέρα [01-31], Ωρα, λεπτά, Ζώνη δευτερολέπτου και χρόνο [τέταρτα μιας ώρας])
- <fo> Η πρώτη ψηφιολέξη, που κωδικοποιείται όπως sms-υποβάλλει την πρώτη ψηφιολέξη στο έγγραφο [4], η προκαθορισμένη αξία είναι 17 για sms-υποβάλλει.
- <index> Θέση της αποθήκευσης στη μνήμη.
- <lenght> Τρόπος κειμένων (+CMGF=1): αριθμός χαρακτήρων
PDU τρόπος (+CMGF=0): μήκος της μονάδας στοιχείων TP στις Ψηφιολέξεις.
- <mem1> Η μνήμη που χρησιμοποιείται στον κατάλογο, που διαβάζεται και διαγράφει τα μηνύματα (+CMGL, +CMGR και +CMGD).
- <mem2> Μνήμη που χρησιμοποιείται για να γράψει και να στείλει τα μηνύματα (+CMGW, +CMSS).
- <mid> CBM προσδιοριστικό μηνυμάτων.
- <mr> Αναφορά μηνυμάτων
- <oa> Διεύθυνση δημιουργών
- <pid> Προσδιοριστικό πρωτοκόλλου
- < pdu> Για sms : GSM 04.11 SC διεύθυνση ακολουθούμενη από GSM 03.40 TPDU το δεκαεξαδικό σχήμα, που κωδικοποιείται με όπως διευκρινίζεται στο έγγραφο [4] Για CBS : GSM 03.41 TPDU το δεκαεξαδικό σχήμα.
- <ra> Λαμβάνουσα διεύθυνση
- <sca> Κεντρική διεύθυνση υπηρεσιών
- <scts> Γραμματόσημο κεντρικού χρόνου υπηρεσιών με το σχήμα σειράς:
“yy/MM/dd,hh :mm :ss±_zz” (Έτος/ Μήνας /Ημέρα /Ωρα: λεπτά: δευτερόλεπτα ± χρονική Ζώνη)
- <sn> CBM αύξων αριθμός
- <st> θέση ενός SMS - ΘΕΣΗ – ΈΚΘΕΣΗ
- <stat> θέση του μηνύματος στη μνήμη
- <tooa> Τύπος διεύθυνσης <oa>
- <tora> Τύπος διεύθυνσης <ra>
- <tosca> Τύπος διεύθυνσης <sca>
- <total1> Αριθμός θέσεων μηνυμάτων μέσα <mem1>.
- <total2> Αριθμός θέσεων μηνυμάτων μέσα <mem2>.
- <used1> συνολικός αριθμός θέσεων μηνυμάτων μέσα <mem1>
- <used2> συνολικός αριθμός θέσεων μηνυμάτων μέσα <mem2>.
- <vp> Περίοδος ισχύος σύντομου μηνύματος, η προκαθορισμένη τιμή είναι 167.

2.3.1 Προτιμημένο σχήμα μηνυμάτων +CMGF

Περιγραφή

Τα σχήματα μηνυμάτων που υποστηρίζονται από PDU τρόπους ο οποίος είναι τρόπος κειμένων.

Στον PDU τρόπο, ένα πλήρες μήνυμα SMS συμπεριλαμβανομένων όλων των πληροφοριών επιγραφών δίνεται ως δυαδική σειρά (με το δεκαεξαδικό σχήμα).

Επομένως, μόνο το ακόλουθο σύνολο χαρακτήρων επιτρέπεται: {'0','1','2','3','4','5','6','7','8','9' 'A','B','C','D','E','F'}. Κάθε ζευγάρι ή χαρακτήρας μετατρέπεται σε μια ψηφιολέξη (π.χ.: '41' μετατρέπεται στο χαρακτήρα " A "ASCII, ο οποίος κώδικας ASCII είναι 0x41 ή 65).

Στον τρόπο κειμένων, όλες οι εντολές και οι απαντήσεις είναι στους χαρακτήρες ASCII.

Το επιλεγμένο σχήμα αποθηκεύεται σε EEPROM από την +CSAS εντολή.

Σύνταξη εντολής : AT+CMGF=<mode> (Με 0 ή 1 επιλέγετε ο PDU ή ο τρόπος κειμένου(txt mode)) Ορίζει το format του SMS σε txt ή PDU mode.

Εντολή	Πιθανές απαντήσεις
AT+CMGF?	+CMGF:1 OK
Σημείωση: Τρέχον σχήμα μηνυμάτων	Σημείωση: τρόπος κειμένων
AT+CMGF=?	+CMGF(0,1) OK
Σημείωση: Πιθανό σχήμα μηνυμάτων	Σημείωση: το κείμενο ή οι τρόποι PDU είναι διαθέσιμοι.

Παράδειγμα, που στέλνει ένα μήνυμα SMS στον PDU τρόπο.

Εντολή	Πιθανές απαντήσεις
AT+CMGF=0	OK
Σημείωση: καθορισμένος τρόπος PDU	Σημείωση: Τρόπος PDU έγκυρος
AT+CMGS=14<CR> 0001030691214365000004C9E9340B	+CMGS: 4 OK
Σημείωση: Στείλετε πλήρες MSG στον PDU τρόπο, καμία SC διεύθυνση	Σημείωση: MSG που στέλνεται σωστά, < mt > επιστρέφεται.

0	PDU τρόπος
1	τρόπος κειμένου

Το < pdu> μήνυμα αποτελείται από τη διεύθυνση SC (00 καμία διεύθυνση Sc μέσα που να δίνεται, η διεύθυνση Sc προεπιλογής χρήσης διαβάζεται με την εντολή +CSCA) και το μήνυμα TPDU.

Σε αυτό το παράδειγμα, το μήκος στις ψηφιολέξεις του απομονωτή TPDU είναι 14, κωδικοποιημένος ως GSM 03.40.

Σε αυτήν την περίπτωση το TPDU είναι: 0x01 0x03 0x06 0x91 0x21 0x43 0x65 0x00 0x00 0x04 0xC9 0xE9 0x34 0x0B, ποια μέσα σχετικά με το GSM 03.40 :

<fo>	0x01(SMS- υποβάλατε, καμία περίοδος ισχύος)
<mr>	(TP-MR) 0x03 (αναφορά μηνυμάτων)
<da>	(TP-DA) 0x06 0x91 0x21 0x43 0x65 (διεύθυνση προορισμού +123456)
<pid>	(TP-PID) 0x00 (προσδιοριστικό πρωτοκόλλου)
<dc>	(TP-DCS) 0x00 (Σχέδιο κωδικοποίησης στοιχείων: 7 bits αλφάβητου)
<length>	(TP-UDL) 0x04 (Μήκος στοιχείων χρηστών, 4 χαρακτήρες του κειμένου)
TP-UD	0xC9 0xE9 0x34 0x0B (Στοιχεία χρηστών: ISSY)

Το TPDU με το δεκαεξαδικό σχήμα πρέπει να μετατραπεί σε δύο χαρακτήρες ASCII. Παραδείγματος χάριν, η ψηφιολέξη 0x2A παρουσιάζεται ME ως δύο χαρακτήρες "2" (ASCII 50) και "A" (ASCII 65).

ΕΝΔΕΙΞΗ ΝΕΟΥ ΜΗΝΥΜΑΤΟΣ +CNMI

ΠΕΡΙΓΡΑΦΗ

Αυτή η εντολή επιλέγει τη διαδικασία για την υποδοχή νέου μηνύματος για το διαδίκτυο.

Σύνταξη εντολής: AT +CNMI=<mode>,<mt>,<bm>,<ds>,<bfr> Ορίζουμε στο μόντεμ να επιστρέφει ειδοποίηση στη σειριακή μετά τη λήψη οποιουδήποτε SMS.

Εντολή	Πιθανές απαντήσεις
AT+CNMI=2,1,0,0,0 Σημείωση:<mt>=1	OK
	AT+CNMI : “SM”,1 Σημείωση: λαμβανόμενο μήνυμα
AT+CNMI=2,2,0,0,0 Σημείωση:<mt>=2	OK
	+CMT : “123456”, “98/10/01,12 :30 00+00”, 129,4 ,32,240, “15379”, 129,5<CR><LF> Σημείωση: λαμβανόμενο μήνυμα
AT+CNMI=2,0,0,1,0 Σημείωση:<ds>=1	OK
AT+CMGS=“ +33146290800” <CR> ευτυχή γενέθλια !<ctrl-Z> Σημείωση: Στείλετε ένα μήνυμα στον τρόπο κειμένων	+CMGS : 7 OK
	+CDS : 2, 116, “+33146290800”, 145, “98/10/01,12 :30 :07+04”, “98/10/01 12 :30 08+04”, 0 Σημείωση: το μήνυμα παραδόθηκε σωστά

ΚΑΘΟΡΙΣΜΕΝΕΣ ΤΙΜΕΣ

<mode>: ελέγχει την επεξεργασία των εκούσιων κωδικών αποτελέσματος
0 : εκούσιοι κώδικες αποτελέσματος απομονωτών στο TA. Αν ο απομονωτής κώδικα αποτελέσματος TA είναι πλήρης, οι ενδείξεις μπορούν να αποθηκευτούν σε ισχύ κάποια άλλη ή οι παλαιότερες ενδείξεις μπορούν να απορριφθούν και να αντικατασταθούν με τις νέες λαμβανόμενες ενδείξεις.

1 : Απορρίψτε την ένδειξη και απορρίψτε τους νέους λαμβανόμενους κώδικες αποτελέσματος μηνυμάτων εκούσιους όταν διατηρείται η σύνδεση TA-TE.
Διαφορετικά προς τα εμπρός αυτοί άμεσα στο TE

2 : Αποθηκεύστε τους εκούσιους κώδικες αποτελέσματος στο TA όταν διατηρείται η σύνδεση TA-TE και επίπεδος αυτοί στο TE μετά από την επιφύλαξη. Διαφορετικά προς τα εμπρός αυτοί άμεσα στο TE.

3 : Μπροστινοί εκούσιοι κώδικες αποτελέσματος άμεσα στο TE. Συγκεκριμένος inband συνδέσεων TA-TE που χρησιμοποιείται για να ενσωματώσει τους κώδικες και τα στοιχεία αποτελέσματος όταν είναι το TA σε απευθείας σύνδεση τρόπο στοιχείων.

ΣΗΜΑΝΤΙΚΗ ΣΗΜΕΙΩΣΗ: MONO <mode>=2 υποστηρίζεται

Οποιαδήποτε άλλη τιμή για <mode>(0,1 ή 3) γίνονται αποδεκτά (ο επιστροφής κώδικας θα είναι εντάξει), αλλά η επεξεργασία των εκούσιων κωδικών αποτελέσματος θα είναι η ίδια με with<mode>=2

<mt>: θέτει τη δρομολόγηση ένδειξης κώδικα αποτελέσματος για τα μηνύματα και παραδίδει τις ενδείξεις.

Η προεπιλογή είναι 1.

0: Κανένα μήνυμα δεν παραδόθηκε στις ενδείξεις που καθοδηγείται.

1: Τα μηνύματα καθοδηγούνται χρησιμοποιώντας τον κώδικα: +CMTI: "SM", <index>

2: Τα μηνύματα (εκτός από την κατηγορία 2 μηνυμάτων) καθοδηγούνται χρησιμοποιώντας τον κώδικα:

If PDU mode:

+CMT :[<alpha>], <length> <CR> <LF> <pdu>

If text mode:

+CMT: <oa>,[<alpha>], <sects> [,<tooa>, <fo>, <pid>, <dcs>, <sca>, <tosca>, <lenght>] <CR><LF><data>

3. Η κατηγορία 3 μηνυμάτων καθοδηγείται άμεσα χρησιμοποιώντας τον κώδικα <mt>=2;

Άλλα μηνύματα κατηγοριών οδηγούνται στην ένδειξη <mt>=1.

<bm>: καθορίζει τους κανόνες για τους λαμβανομένους τύπους CBMs(μήνυμα ραδιοφωνικής μετάδοσης κυττάρων). Εξαρτώνται επίσης από το σχέδιο κωδικοποίησης (κείμενο PDU) και τον καθορισμό των επίλεκτων CBM τύπων (βλ. εντολή + CSCB).

Η προεπιλογή είναι 0.

0: Καμία CBM ένδειξη δεν καθοδηγείται στο TE. Τα CBMs αποθηκεύονται.

1: Το CBM αποθηκεύεται και μια ένδειξη της θέσης μνήμης καθοδηγείται στην εφαρμογή πελατών χρησιμοποιώντας τον εκούσιο κώδικα αποτελέσματος : +CBMI : "BM", <δείκτης>

2: Τα νέα CBMs αποθηκεύονται άμεσα στο TE χρησιμοποιώντας εκούσιο κώδικα αποτελέσματος.

If PDU mode :

+CBM: <lenght><CR><LF><pdu> or

If text mode:

+CBM: <sn>,<mid>,<dcs>,<page>,<pages><CR><LF><data>

3: κατηγορία 3 CBMs : όπως <bm>=2.

Άλλες κατηγορίες CBMs : όπως <bm>=1.

<ds> για μηνύματα θέσης-έκθεσης. Η προεπιλογή είναι 0.

0 : Καθόλου μηνύματα θέσης-έκθεσης δεν καθοδηγούνται.

1: μηνύματα θέσης-έκθεσης καθοδηγούνται χρησιμοποιώντας εκούσιο κώδικα.

If PDU mode :

+CDS:<length> <CR> <LF> <pdu> (PDU mode)or

If text mode:

+CDS : <fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>,<st> (Τρόπος κειμένων)

2: Μηνύματα θέσης-έκθεσης αποθηκεύονται και καθοδηγούνται χρησιμοποιώντας εκούσιο κώδικα αποτελέσματος: +CDSI : “SR”,<index>

<bfr> Η προεπιλογή είναι 0.

0: Ο απομονωτής TA των εκούσιων κωδικών αποτελέσματος που καθορίζονται μέσα σε αυτήν την εντολή ξεπλένεται στο TE όταν εισάγεται <mode>=1 σε 3 (η απάντηση OK θα δοθεί πριν ξεπλύνει τους κώδικες.)

1: : Ο απομονωτής TA των εκούσιων κωδικών αποτελέσματος που καθορίζονται μέσα σε αυτήν την εντολή καθαρίζεται όταν εισάγεται <mode>=1...3

2.3.2 Διαβάστε το μήνυμα +CMGR

Περιγραφή

Αυτή η εντολή επιτρέπει την εφαρμογή για να διαβαστούν τα αποθηκευμένα μηνύματα. Τα μηνύματα διαβάζονται από τη μνήμη που επιλέγεται από την +CPMS εντολή.

Σύνταξη εντολής : AT+CMGR= <index> (Θέση της αποθήκευσης στη μνήμη)

Διαβάζει το μήνυμα που βρίσκεται στην <index> θέση μνήμης.

Σύνταξη απάντησης για τον τρόπο κειμένων:

+CMGR: <star>, <oa>,[<alpha>] <scts> [,<tooa>,<fo>,<pid>,<dcs>,<sca>,<tosca>,<length>] <CR> <LF> <data> (Για SMS-παράδοσης μόνο).

+CMGR: <stat>,<da>,[<alpha>] [,<toda>,<fo>,<pid>,<dcs> , [<vp>], <sca>,<tosca>,<length>]<CR><LF> <data> (Για SMS-Υποβολής μόνο).

+CMGR: <stat>,<fo>,<mr>,[<ra>] , [<tora>],<scts>,<dt>,<st> (Για SMS- θέσης-έκθεσης μόνο).

Σύνταξη απάντησης για τον PDU τρόπο:

+CMGR: <stat>,[<alpha>] <length> <CR> <LF> <pdu>

Ένα μήνυμα που διαβάζεται με τη θέση “REC ΑΔΙΑΒΑΣΤΟ” θα ενημερωθεί στη μνήμη με τη θέση “REC ΔΙΑΒΑΣΜΕΝΟ”.

Π.χ. Για να διαβάσουμε ένα SMS το οποίο να γράφει «*Hi John*,» με την χρήση GSM Modem το οποίο είναι συνδεδεμένο με έναν μικροελεγκτή της οικογένειας MCS51 πρέπει να γράψουμε το παρακάτω πρόγραμμα σε γλώσσα assembly:

```
ReceiveTimeFL      equ    20h.1

READ1:             db 'AT+CMGR=1',0ah,0
READ2:             db 'AT+CMGR=2',0ah,0
READ3:             db 'AT+CMGR=3',0ah,0
READ4:             db 'AT+CMGR=4',0ah,0
READ5:             db 'AT+CMGR=5',0ah,0
READ6:             db 'AT+CMGR=6',0ah,0
READ7:             db 'AT+CMGR=7',0ah,0
READ8:             db 'AT+CMGR=8',0ah,0
```

```
    Acall requestMessage
    Acall ReadMessage
    Sjmp $
RequestMessage
    mov  dptr,#READ1
    acall sendtoSerialA
    acall readmessage
    acall decodemessage
    ret
SendToSerialA:
    clr  a
    movc a,@a+dptr
    jz   oxo1a
    acall send_char
    inc  dptr
    sjmp SendToSerialA
oxo1a: mov  a,MessageIndex
    acall send_char
    mov  a,#0ah
    acall send_char
    ret
```

```

ReadMessage:
    mov    r0,#50h
    mov    r7,#100
    clr    ReceiveTimeFL
    mov    ReceiveDelay,#(OneSecond*4)
rm1:     clr    ri
rm1a:    jnb   ri,rt1
        sjmp  rm11
rt1:     jnb   ReceiveTimeFL,rm1a
        sjmp  message_error
rm11:    djnz  r7,rm2
        sjmp  message_error

rm2:     mov    a,sbuf
        cjne  a,#',',rm1
        ret

```

```

message_error:
    mov    r0,#50h
DL1:     mov    @r0,#0
        inc   r0
        cjne  r0,#87h,DL1
        ret

```

Σημείωση:

Το <star> η παράμετρος για τις εκθέσεις θέσης SMS πάντα "ΔΙΑΒΑΖΟΝΤΑΙ".

Παράδειγμα:

Εντολές	Πιθανές απαντήσεις +CMTI: "SM",1 Σημείωση: Νέο λαμβανόμενο μήνυμα
AT+CMGR=1 Σημείωση: Διαβάστε το μήνυμα	+CMGR: "ΑΔΙΑΒΑΣΤΟ", "0146290800" "98/10/01, 18:22 :11+00",<CR><LF> ABCdefGHI OK
AT+CMGR=1 Σημείωση: Διαβάστε το μήνυμα πάλι	+CMGR:"ΔΙΑΒΑΣΤΕ","0146290800" "98/10/01, 18:22 :11+00",<CR><LF> ABCdefGHI OK Σημείωση: το μήνυμα διαβάζεται τώρα
AT+CMGR=2 Σημείωση: διαβάστε σε έναν λανθασμένο δείκτη	+CMS ΛΑΘΟΣ: 321 Σημείωση: Λάθος: άκυρος δείκτης
AT+CMGF=0 ;+CMGR=1 Σημείωση: Στον PDU τρόπο	+CMGR:2,<length><CR><LF><pdu> OK Σημείωση: Το μήνυμα αποθηκεύεται αλλά Κανένας <alpha> τομέας.
AT+CMGF=1;+CPMS="SR";+CNMI=,,2 Η αναστοχιοθέτηση στον τρόπο κειμένων, σύνολο διάβασε τη μνήμη "SR" και επιτρέπει την αποθήκευση της περαιτέρω έκθεσης θέσης SMS στη μνήμη "SR"	OK
AT+CMSS=3 Στείλετε ένα SMS που αποθηκεύεται προηγουμένως	+CMSS: 160 OK
	+CDSI: "SR",1 Νέα έκθεση θέσης SMS που αποθηκεύεται στη μνήμη "SR" στο δείκτη 1.
AT+CMGR=1 Διαβάστε τα SMS θέσης έκθεσης	+CMGR: "REC ΑΔΙΑΒΑΣΤΟ",6,160, "+33612345678", 129, "01/05/31,15:15:0 9+00", "01/05/31,15:15:09+00",0 OK

2.3.3 Μήνυμα καταλόγων +CMGL

Περιγραφή

Αυτή η εντολή επιτρέπει την εφαρμογή για να διαβαστούν τα αποθηκευμένα μηνύματα, με την ένδειξη του τύπου του μηνύματος που διαβάζει.

Το μήνυμα διαβάζεται από τη μνήμη που επιλέγεται από την +CPMS εντολή.

Σύνταξη εντολής : AT+CMGL=<stat> (θέση του μηνύματος στη μνήμη)

Σύνταξη απάντησης για τον τρόπο κειμένων:

+CMGL: <index>,<stat>,<da/oa>,[<alpha>],[<scts>,<tooa/toda>,<length>]

<CR><LF><data>(Για SMS- ΠΑΡΑΔΩΣΕΙΣ και SMS - ΥΠΟΒΟΛΗΣ, μπορούν να ακολουθηθούν από άλλο δείκτη <CR> <LF>+CMGL: <index>....)

+CMGL: <index>,<stat>,<fo>,<mr>,[<ra>],[<tora>],<scts>,<dt>,<st> (Για SMS- Θέσης-Έκθεσης μόνο, μπορεί να ακολουθηθεί από άλλο δείκτη <CR><LF>+CMGL: <index>....)

Σύνταξη απάντησης για τον PDU τρόπο:

+CMGL: <index>,<stat>,[<alpha>], <length> <CR><LF> <pdu> (Για SMS- ΠΑΡΑΔΩΣΕΙΣ ,SMS – ΥΠΟΒΟΛΗΣ και SMS- Θέσης-Έκθεσης, μπορεί να ακολουθηθεί από άλλο δείκτη <CR><LF>+CMGL: <index>....)

Εντολές	Πιθανές απαντήσεις
AT+CMGL="ΑΔΙΑΒΑΣΤΟ" Σημείωση: Αδιάβαστά μηνύματα καταλόγων στον τρόπο κειμένων	+CMGL: 1,"ΑΔΙΑΒΑΣΤΟ", "0146290800" <CR><LF>Θα είμαι αργά +CMGL: 3,"ΑΔΙΑΒΑΣΤΟ", "46290800" <CR><LF>θα με δείτε απόψε OK Σημείωση: 2 μηνύματα είναι αδιάβαστά αυτά τα μηνύματα έπειτα θα αλλάξουν τη θέση τους "ΔΙΑΒΑΣΜΕΝΟ"
AT+CMGL="REC ΔΙΑΒΑΣΜΕΝΟ" Σημείωση: Κατάλογος διαβασμένων μηνυμάτων στον τρόπο κειμένων	+CMGL: 2,"ΑΔΙΑΒΑΣΤΟ", "0146290800" <CR><LF>Κρατήστε δροσερό OK
AT+CMGL="STO Σταλμένο" Σημείωση: Ο κατάλογος αποθήκευσε Και έστειλε τα μηνύματα στον τρόπο κειμένων	OK Σημείωση: Κανένα μήνυμα δε βρέθηκε
AT+CMGL=1 Σημείωση: Κατάλογος διαβασμένων μηνυμάτων στον PDU τρόπο.	+CMGL: <CR><LF> 07913366003000F3040B913366920547 F40013001190412530400741AA8E5A9C 5201 OK

Καθορισμένες τιμές

<stat> Πιθανές τιμές (θέση των μηνυμάτων στη μνήμη):

Πιθανές τιμές τρόπου Κειμένων	Πιθανές τιμές τρόπου PDU	Θέση των μηνυμάτων στη μνήμη
“REC ΑΔΙΑΒΑΣΤΟ”	0	Λαμβανόμενα αδιάβαστά μηνύματα
“REC ΔΙΑΒΑΣΜΕΝΟ”	1	λαμβανόμενα διαβασμένα μηνύματα
“STO Μη σταλμένο”	2	Αποθηκευμένο μη σταλμένο μήνυμα
“STO Σταλμένο”	3	Αποθηκευμένο σταλμένο μήνυμα
“ΟΛΑ”	4	Όλα τα μηνύματα

Σημείωση: Για τις εκθέσεις θέσης SMS, μόνο "ΟΛΟΙ"/ 4 και οι "ΔΙΑΒΑΣΜΕΝΕΣ" "τιμές /1 <stat> της παραμέτρου θα απαριθμήσουν τα μηνύματα, άλλες τιμές θα επιστρέψουν μόνο OK.

2.3.4 Στείλετε το μήνυμα +CMGS

περιγραφή

Ο <address> τομέας είναι η διεύθυνση του τερματικού στο οποίο το μήνυμα στέλνεται. Για να στείλετε το μήνυμα, δακτυλογραφήστε απλά, <ctrl-Z> και <ESC> (ASCII 27).

Αυτή η το εντολή μπορεί να είναι χρησιμοποίηση <ESC> χαρακτήρα κατά την είσοδο του κειμένου.

Στον PDU τρόπο, μόνο οι δεκαεξαδικοί χαρακτήρες χρησιμοποιούνται ('0'....'9', 'A'.....'F').

Σύνταξη εντολής στον τρόπο κειμένου:

AT+CMGS= <da> [,<today>]<CR>

Το κείμενο εισάγεται <ctrl-Z / ESC>

Σύνταξη εντολής στον PDU τρόπο:

AT+CMGS= <length> <CR>

Π.χ. Για να στείλουμε ένα SMS το οποίο να γράφει «*This is a Test*» με την χρήση GSM Modem το οποίο είναι συνδεδεμένο με έναν μικροελεγκτή της οικογένειας MCS51 πρέπει να γράψουμε το παρακάτω πρόγραμμα σε γλώσσα assembly:

```

        mov DPTR,#SEND_SMS
        acall send_message
L1:     clr    ri
        jnb   ri,$
        mov   a,sbuf
        cjne a,#'>',L1
        mov   DPTR,#message
        acall send_message
        sjmp  $

send_message:
        clr   a
        movc a,@a+dptr
        jz    oxo
        acall send_char
        inc  dptr
        sjmp send_message
oxo:    ret

SendToSerial:
        clr   a
        movc a,@a+dptr
        jz    oxo1
        acall send_char
        inc  dptr
        sjmp SendToSerial
oxo1:   ret
SEND_CHAR: clr   ti
        MOV  SBUF,A
        JNB TI,$
        CLR  TI
        RET

SEND_SMS: DB    'AT+CMGS=Αριθμός τηλεφώνου',0aH,0
Message db    'This Is A Test',1ah,0

```

Ο PDU εισάγεται <ctrl-Z / ESC >

Εντολές	Πιθανές απαντήσεις
AT+CMGS=" +33146290800" <CR>	>
Παρακαλώ καλέστε με σύντομα, Fred. <ctrl-Z> Σημείωση: στείλετε ένα μήνυμα στον τρόπο κειμένων.	
	+CMGS: <mr> OK Σημείωση: επιτυχής μετάδοση
AT+CMGS= <length><CR> Σημείωση: στείλετε ένα μήνυμα στον PDU τρόπο (1/3)	>
<rdu> Σημείωση: Εισάγετε το μήνυμα στον PDU τρόπο (2/3)	
<ctrl-Z> Σημείωση: Τελειώστε το μήνυμα (3/3)	+CMGS: <mr> OK Σημείωση: επιτυχής μετάδοση

Εξερχόμενο μήνυμα (επιτυχείς και περιπτώσεις αποτυχίας) είναι κυκλικό σε μια ψηφιολέξη (0 ακολουθούν 255).

Σημείωση: αυτός ο αριθμός είναι ένας αριθμός αποθήκευσης - τα εξερχόμενα μηνύματα δεν αποθηκεύονται.

2.3.5 Διαγράψτε το μήνυμα +CMGD

Περιγραφή

Αυτή η εντολή χρησιμοποιείται για να διαγράψει ένα ή περισσότερα μηνύματα από την προτιμημένη αποθήκευση μηνυμάτων ("BM" SMS-CB RAM αποθήκευση "SM" SMSPP αποθήκευση 'SIM αποθήκευσης' ή "SR" SMS θέση-εκθέσεων αποθήκευσης).

Αναφερθείτε επίσης στην προτιμημένες αποθήκευση μηνυμάτων με την εντολή +CPMS.

Σύνταξη εντολής: AT+CMGD=<Index> [,<DelFalg>] Σβήνει το μήνυμα της πρώτης θέσης μνήμης.

Εντολές	Πιθανές απαντήσεις
	+CMTI: "SM", 3 Σημείωση: νέο λαμβανόμενο μήνυμα
AT+CMGR=3 Σημείωση: διαβάστε το	+CMGR: "REC ΑΔΙΑΒΑΣΤΟ", "0146290800", "98/10/01,18 :19 :20+00" <CR><LF> Μήνυμα λαμβανόμενο! Σημείωση: Μήνυμα που παραλαμβάνεται αδιάβαστο από 0146290800 01/10/1998 στις 18H 19m 20s
AT+CMGD=3 Σημείωση: διαβάστε το	OK Σημείωση: Μήνυμα που διαγράφεται
AT+CMGD=1,0	OK Σημείωση: Το μήνυμα από την προτιμημένη αποθήκευση μηνυμάτων στη θέση 1 διαγράφεται
AT+CMGD=1,1	OK Σημείωση: Όλα τα ΔΙΑΒΑΣΜΕΝΑ μηνύματα από την προτιμημένη αποθήκευση μηνυμάτων διαγράφονται
AT+CMGD=1,2	OK Σημείωση: Όλοι ΔΙΑΒΑΖΟΥΝ τα μηνύματα και τα ΣΤΑΛΜΕΝΑ δημιουργημένα μηνύματα διαγράφονται
AT+CMGD=1,3	OK Σημείωση: Όλα που ΔΙΑΒΑΖΟΝΤΑΙ, ΣΤΑΛΜΕΝΑ και μη σταλμένα μηνύματα διαγράφονται.
AT+CMGD=1,4	OK Σημείωση: Όλα τα μηνύματα διαγράφονται

Καθορισμένες τιμές

<index> εάν DelFlag=0

- (1-20) εάν η προτιμημένη αποθήκευση μηνυμάτων είναι "BM"
- SIM εξαρτώμενες τιμές ακέραιων αριθμών (στη σειρά του αριθμού θέσης SIM) εάν η προτιμημένη αποθήκευση μηνυμάτων είναι "SM" ή "SR".

Εάν DelFlag είναι διαφορετικός από 0, <index> αγνοείται.

<DelFlag>

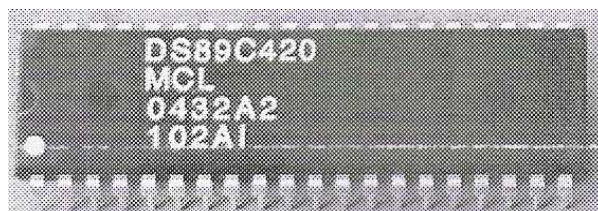
- 0 Διαγράψτε το μήνυμα στη θέση <index> (προκαθορισμένη τιμή).
- 1 Διαγράψτε όλα τα ΔΙΑΒΑΣΜΕΝΑ μηνύματα
- 2 Διαγράψτε όλα τα μηνύματα ΠΟΥ ΔΙΑΒΑΖΟΝΤΑΙ και ΣΤΕΛΝΟΝΤΑΙ
- 3 Διαγράψτε όλα τα μηνύματα ΠΟΥ ΔΙΑΒΑΖΟΝΤΑΙ, ΣΤΑΛΜΕΝΑ ΚΑΙ ΜΗ ΑΠΕΣΤΑΛΜΕΝΑ .
- 4 Διαγράψτε όλα τα μηνύματα.

Εάν <DelFlag> είναι omitted, η προκαθορισμένη τιμή χρησιμοποιείται

Σημείωση: όταν η προτιμημένη αποθήκευση μηνυμάτων είναι "SR", ως εκθέσεις θέσης SMS είναι υποτιθέμενη "ΔΙΑΒΑΣΜΕΝΗ" θέση, εάν <DelFlag> είναι μεγαλύτερος από 0, όλες οι εκθέσεις θέσης SMS θα διαγραφούν.

ΚΕΦΑΛΑΙΟ 3

3.1 Ο DS89C420



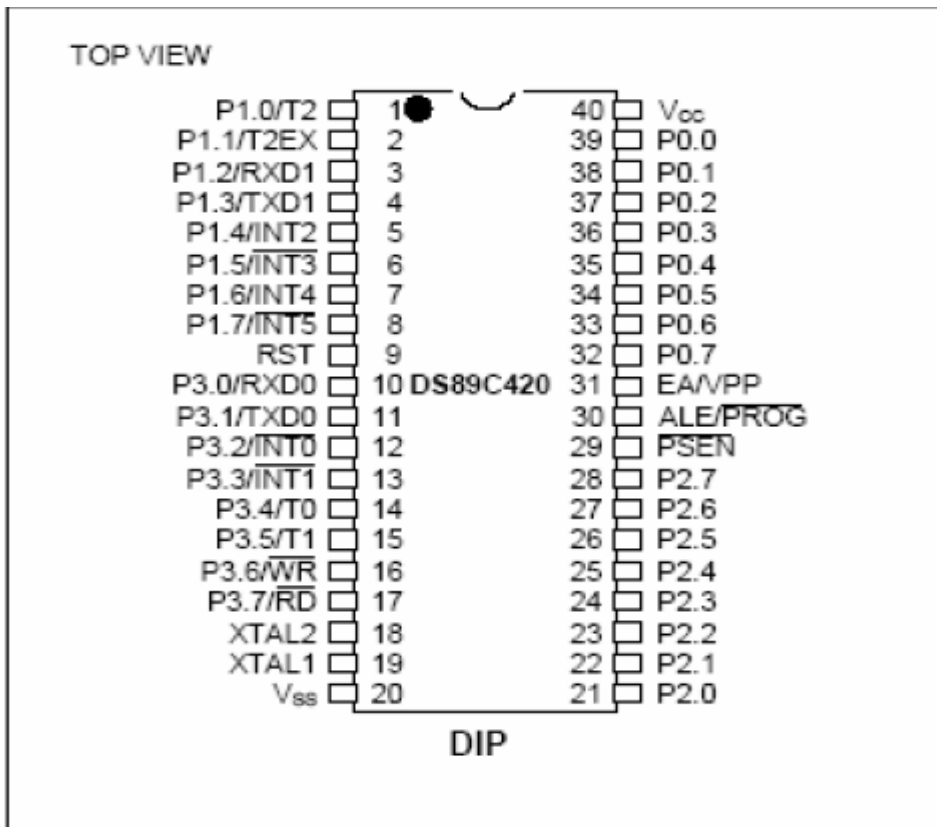
Ο 8051 εξακολουθεί να είναι ένας από τους πιο δημοφιλείς μικροελεγκτές, παρά τη σχετικά μεγάλη ηλικία του. Πολλοί άλλοι μικροελεγκτές έχουν από τότε αναπτυχθεί, οι οποίοι βασίζονται και είναι συμβατοί με τον 8051, ένας από αυτούς είναι ο DS89C420 της Dallas.

Ο μικροελεγκτής DS89C420 της Dallas, ο οποίος είναι εξαιρετικά υψηλής ταχύτητας, παρέχει τη βελτιωμένη απόδοση και την κατανάλωση ισχύος με την αρχική έκδοση του 8051. Διατηρεί συμβατότητα με τον 8051, όμως εκτελεί τις ίδιες διαδικασίες σε λιγότερους κύκλους ρολογιών. Συνεπώς, η μεγαλύτερη απόδοση ρυθμού είναι δυνατή για την ίδια ταχύτητα κρυστάλλου. Σαν εναλλακτική λύση, ο μικροελεγκτής DS89C420 μπορεί να λειτουργήσει σε μια μειωμένη συχνότητα για να μειώσει την κατανάλωση ισχύος. Το σχέδιο του επιτρέπει με μικρότερο κρύσταλλο χρονισμού να έχουμε τα ίδια αποτελέσματα με αυτά που θα είχαμε με έναν κλασικό 8051.

Η θεμελιώδης καινοτομία του μικροελεγκτή DS89C420 είναι ότι χρησιμοποιεί μόνο ένα ρολόι ανά κύκλο οδηγίας έναντι με 12 για τον αρχικά 8051. Επίσης η συχνότητά του φτάνει εως τα 33MHz. Ο DS89C420 παρέχει διάφορες περιφερειακές μονάδες και χαρακτηριστικά γνωρίσματα, όπως 16KB / 32KB / 64KB μνήμη, 1KB RAM, τέσσερις 8-bits I/O Ports, τρεις 16-bits timers / counters, δύο chip-UARTs, δύο data pointers, ένα watchdog timer, πέντε επίπεδα interrupts προτεραιότητας, και ένα πολλαπλασιαστή συχνότητας κρυστάλλου. Επίσης παρέχει 256 bytes RAM για τις μεταβλητές και σωρό 128 bytes, τα οποία μπορούν να επιτευχθούν χρησιμοποιώντας την άμεση ή έμμεση διευθυνσιοδότηση, ή χρησιμοποιώντας την έμμεση διευθυνσιοδότηση μόνο.

Ο DS89C420 φαίνεται στο Σχ.3.1. Τα pins του μικροελεγκτή περιγράφονται παρακάτω:

- Pin 40: Τροφοδοσία μικροελεγκτή στα +5V
- Pin 39-32: Τα pin αυτά αποτελούν την Port 0. Στην «πόρτα» αυτή έχουμε συνδέσει το LCD display της συσκευής μας.
- Pin 31: Το pin αυτό καθορίζει αν χρησιμοποιούμε την εσωτερική ή κάποια εξωτερική μνήμη. Όταν το pin είναι συνδεδεμένο με την Vcc (+5V) τότε ο DS89C420 χρησιμοποιεί εξωτερική μνήμη.
- Pin 30: Το pin 30 δημιουργεί μια σειρά παλμών για να μπορέσουμε να διαχειριστούμε την εξωτερική μνήμη, εφόσον έχουμε κάνει χρήση αυτής.
- Pin 29: Το pin 29 (Program Store Enable) ενεργοποιεί / απενεργοποιεί την αποθήκευση δεδομένων στην επιλεγμένη μνήμη του DS89C420.
- Pin 21-28: Τα pin αυτά αποτελούν την πόρτα 2 του DS89C420. Η πόρτα 2 χρησιμοποιείται ως έξοδος του μικροελεγκτή αφού με αυτήν την πόρτα ενεργοποιούμε και απενεργοποιούμε τα ρελέ της συσκευής μας.
- Pin 20: Γείωση του DS89C420.
- Pin 19 & 18: Στα δυο αυτά Pins συνδέεται ο κρύσταλλος που χρειάζεται ο DS89C420.
- Pin 10-17: Τα pin αυτά αποτελούν την Port 3 του DS89C420. Στα pin 10 και 11 βρίσκονται τα Rx (receive) και Tx (transmit) της σειριακής θύρας 1.
- Pin 9: Το pin 9 κάνει επανεκκίνηση στον DS89C420 όταν δεχθεί +5V.
- Pin 1-8: Τα pin αυτά αποτελούν την Port 1 του DS89C420. Στα pin 3 και 4 βρίσκονται τα Rx (receive) και Tx (transmit) της σειριακής θύρας 0.

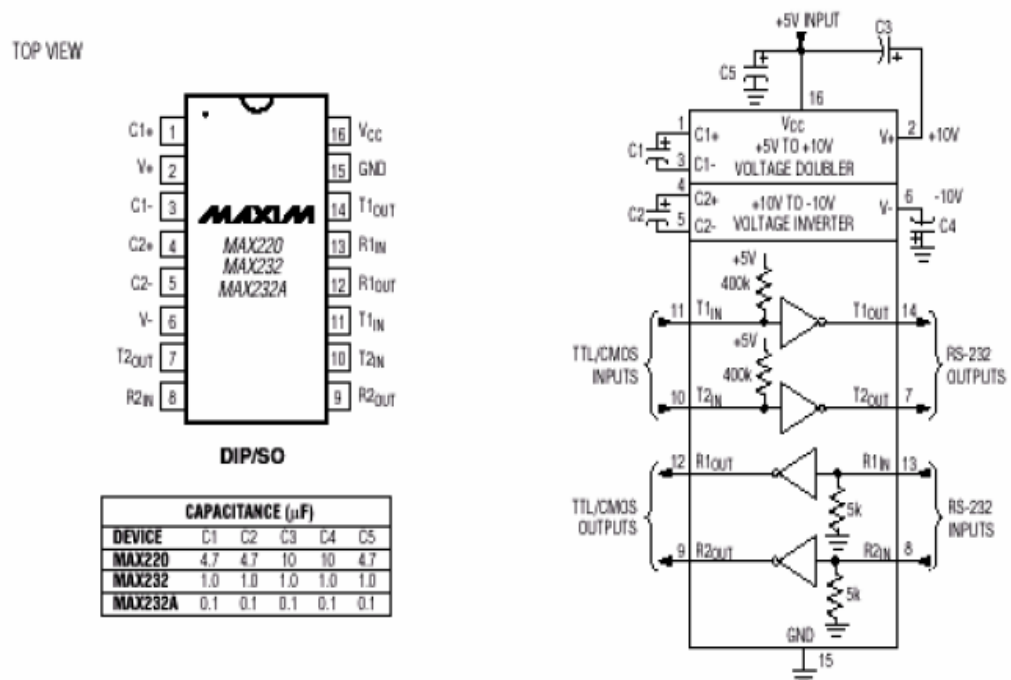


Σχήμα 3.1. Κάτοψη του DS89C420.

3.2 Μετατροπέας MAX 232

Στην συσκευή μας χρειαζόμαστε μια απλή RS232 διεπαφή, μεταξύ του DS89C420 και της σειριακής θύρας του μόντεμ, η οποία δεν μπορεί να λειτουργήσει αν δεν υπάρχει συμμετρική στάθμη τάσεως. Λέγοντας TTL output Level εννοούμε ότι, το λογικό 1 (Hi) λαμβάνει την τιμή των +5Volt και το λογικό 0 (Low) ισούται με 0 V. Στην περίπτωση των CMOS, το επίπεδο “HI” μπορεί να είναι από +5 έως +18V, αλλά και πάλι το “LOW” είναι μηδενικού Voltage. Η στάθμες αυτές ασύμβατες για κάποιες COM Ports, οι οποίες για να λειτουργήσουν αποδεκτά, απαιτούν το λογικό “1” να είναι ίσο με +5 έως 12Volts και το “0” ίσο με -5 έως -12Volts, δηλαδή απαιτούν συμμετρικότητα σε σχέση με το μηδέν.

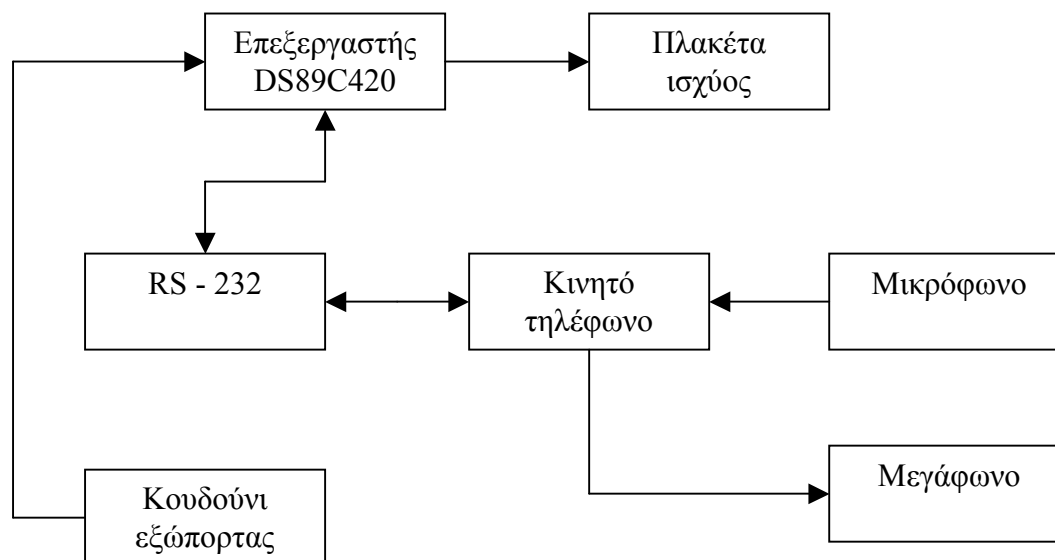
Το MAX 232 είναι ένα ειδικό IC, σκοπός του οποίου είναι να παρέχει “συμβατότητα” μεταξύ ενός κυκλώματος με TTL ή CMOS στάθμη και μίας άλλης πόρτας, που να απαιτεί +V / -V, όπως η RS-232.



Σχήμα 3.2. Ηλεκτρονικό διάγραμμα του MAX 232. Το IC περιλαμβάνει 4 εισόδους και 4 εξόδους, ανά δύο για TTL / CMOS & RS-232.

ΚΕΦΑΛΑΙΟ 4

4.1 ΜΠΛΟΚ ΔΙΑΓΡΑΜΜΑ ΤΗΣ ΣΥΣΚΕΥΗΣ



Σχήμα 4.1: Μπλοκ διάγραμμα της συσκευής.

Διαδικασία λειτουργίας της κατασκευής μας έχει ως εξής:

Όταν η συσκευή μας βρίσκεται σε λειτουργία, και βρισκόμαστε εκτός σπιτιού έχουμε την δυνατότητα να ενεργοποιήσουμε κάποιες συσκευές ή κάποια φορτία με την ακόλουθη διαδικασία.

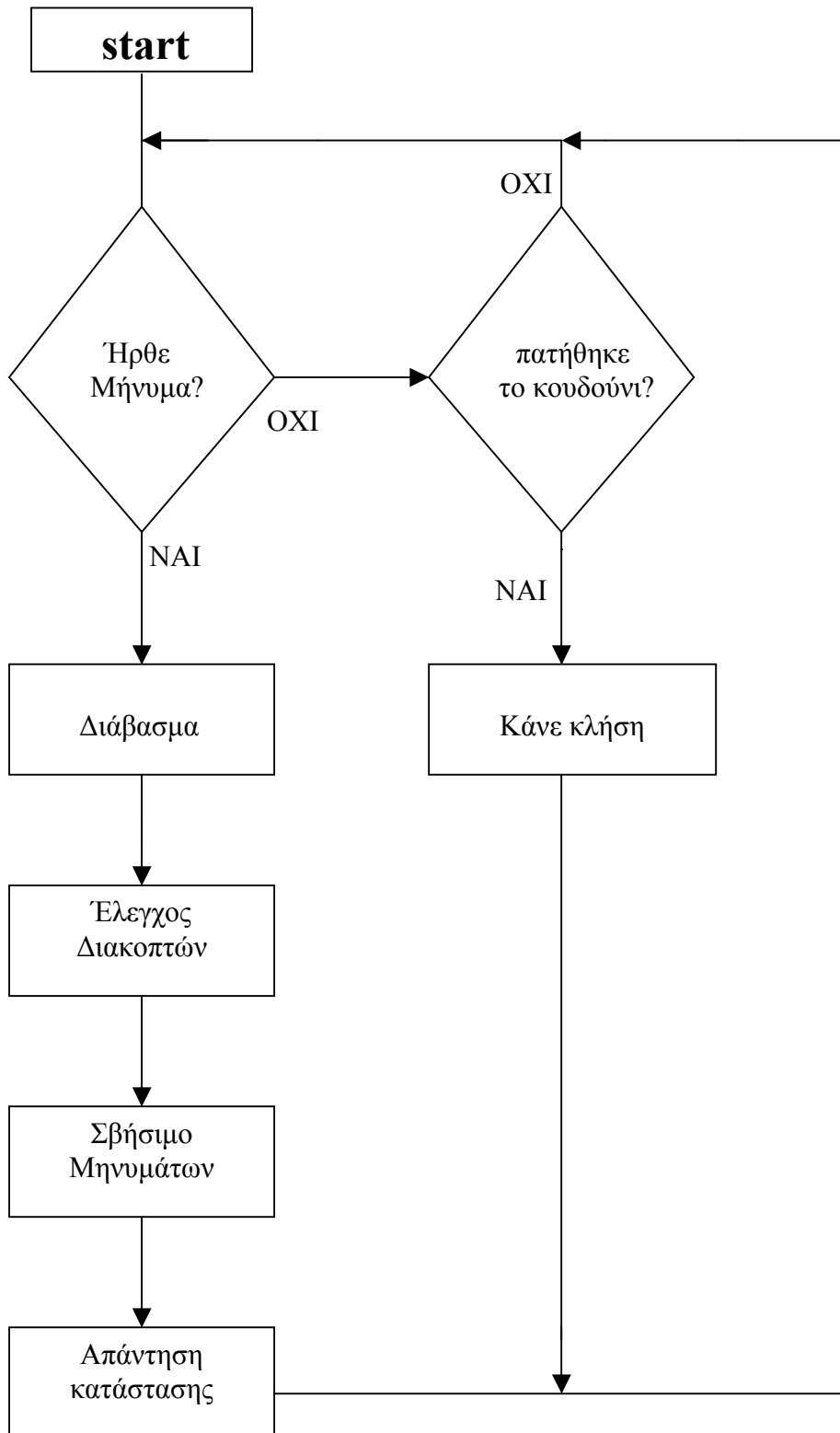
Συντάσσουμε το κωδικοποιημένο μήνυμα που έχουμε κατασκευάσει με τέτοιο τρόπο ώστε να ενεργοποιήσουμε ή να απενεργοποιήσουμε τις συσκευές ή τα φορτία που επιθυμούμε.

Στη συνέχεια το στέλνουμε στο κινητό τηλέφωνο της κατασκευής μας, όπου με την σειρά του και με την βοήθεια της θύρας RS – 232 το στέλνει σειριακά στον επεξεργαστή μας (DS89C420). Ο επεξεργαστής διαβάζοντας το μήνυμα ενεργοποιεί / απενεργοποιεί και τις κατάλληλες πόρτες, όπου είναι συνδεδεμένες με την πλακέτα ισχύος.

Όταν ολοκληρωθεί η διαδικασία της ανάγνωσης και η ενεργοποίηση / απενεργοποίηση των εξόδων ο επεξεργαστής μας κάνει έναν έλεγχο με την κατάσταση των πόρτων και με την βοήθεια πάλι της θύρας RS – 232 περνάει στο κινητό τηλέφωνο όπου αυτό με την σειρά του στέλνει μία απάντηση με τις καταστάσεις όπως έχουν εκείνη την στιγμή.

Ακόμα η κατασκευή μας έχει την δυνατότητα σε περίπτωση που λείπουμε από το σπίτι, και κάποιος μας πατήσει το κουδούνι της εξώπορτας, το κινητό τηλέφωνο της κατασκευής μας αυτόματα μας κάνει κλήση στο προσωπικό μας τηλέφωνο και με την βοήθεια ενός μικροφώνου και ενός μεγαφώνου όπου είναι τοποθετημένα μέσα στο κουδούνι έχουμε την δυνατότητα να επικοινωνήσουμε μαζί του σε οποιοδήποτε σημείο και αν βρισκόμαστε με την βοήθεια των δικτύων κινητής τηλεφωνίας όπου έχουν εξαπλωθεί σε όλων τον κόσμο.

4.2 ΔΙΑΓΡΑΜΜΑ ΡΟΗΣ ΤΗΣ ΚΕΝΤΡΙΚΗΣ ΙΔΕΑΣ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ ΜΑΣ



Σχήμα 4.2: Διάγραμμα ροής της κεντρικής ιδέας του προγράμματος.

Η Λογική η οποία ακολουθεί το πρόγραμμα μας είναι η εξής:

Το πρόγραμμα μας αρχίζει να δουλεύει και βρίσκετε σε ένα συνεχές loop και ελέγχει αν έχει έρθει μήνυμα, και αν έχει πατηθεί το κουδούνι του σπιτιού μας. πχ. Έχει έρθει μήνυμα; Αν (ΟΧΙ), ελέγχει πατήθηκε το κουδούνι; Αν (ΟΧΙ) ξανάπαι στην αρχή και ελέγχει πάλι αν ήρθε μήνυμα.

Αν έχει έρθει μήνυμα (ΝΑΙ), ακολουθείτε η διαδικασία του διαβάσματος και στη συνέχεια γίνετε ο έλεγχος τον διακοπών.

Δηλ. Ελέγχει ποιοι διακόπτες πρέπει να ανοίξουν και ποιοι διακόπτες πρέπει να κλείσουν.

Αφού γίνει ο έλεγχος στην συνέχεια διαγράφετε το μήνυμα αυτόματα από την μνήμη της κάρτας SIM του κινητού τηλεφώνου, έτσι ώστε να είναι έτοιμη να δεχθεί το επόμενο μήνυμα.

Από την στιγμή που έχει γίνει ο έλεγχος και έχει διαγράψει το μήνυμα, το κινητό στέλνει απάντηση με όλες τις καταστάσεις στις οποίες βρίσκονται οι διακόπτες εκείνη την στιγμή, έτσι ώστε αν έχουμε ξεχάσει κανένα διακόπτη ανοικτό ή κλειστό να μας το υπενθυμίσει.

Αν τώρα έχει πατηθεί και το κουδούνι το κινητό αυτόματα σου κάνει κλήση και σου δίνει την δυνατότητα να μιλήσεις εκείνη την στιγμή με το άτομο που βρίσκετε έξω από την πόρτα του σπιτιού σου. Και αμέσως μετά ξαναγυρίζει στην αρχή και ξαναρχίζει τον έλεγχο.

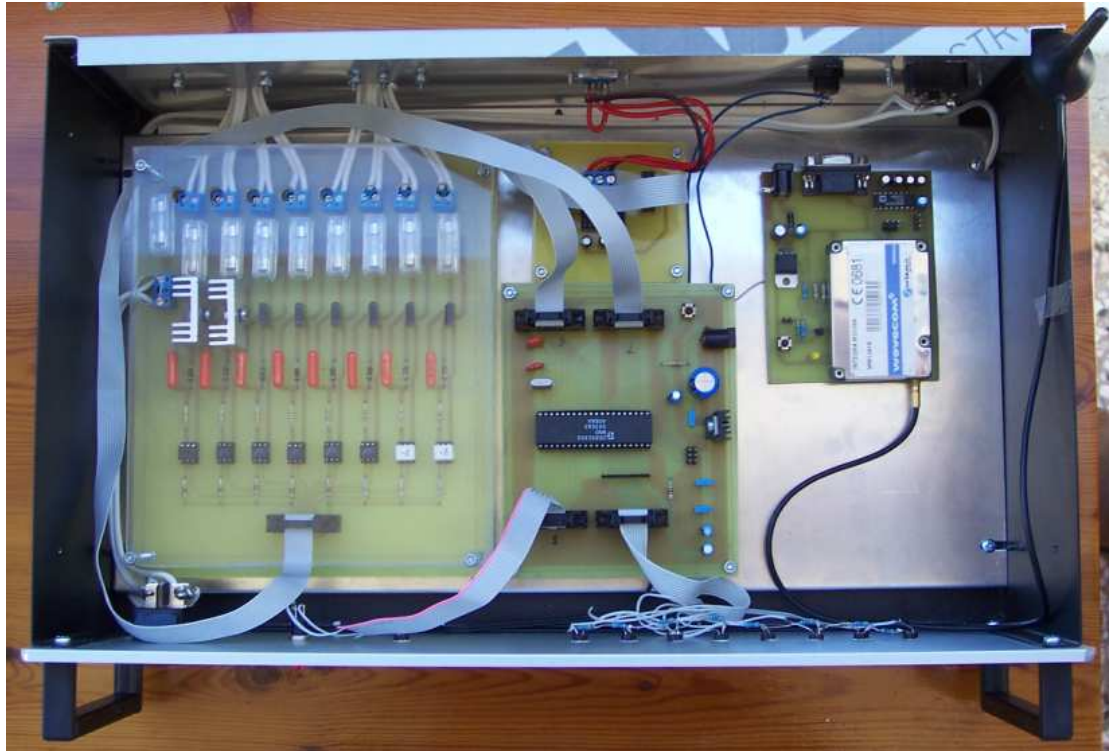
4.3 Η κατασκευή μας



Σχήμα 4.3: Μπροστινή όψη της συσκευής. Όπου βλέπουμε δεξιά 8-led με τις καταστάσεις που θα βρίσκέτε η συσκευή, και αριστερά βλέπουμε το led τροφοδοσίας +5V, το κουμπί του κουδουνιού καθώς και το διακόπτη για τα 220V.



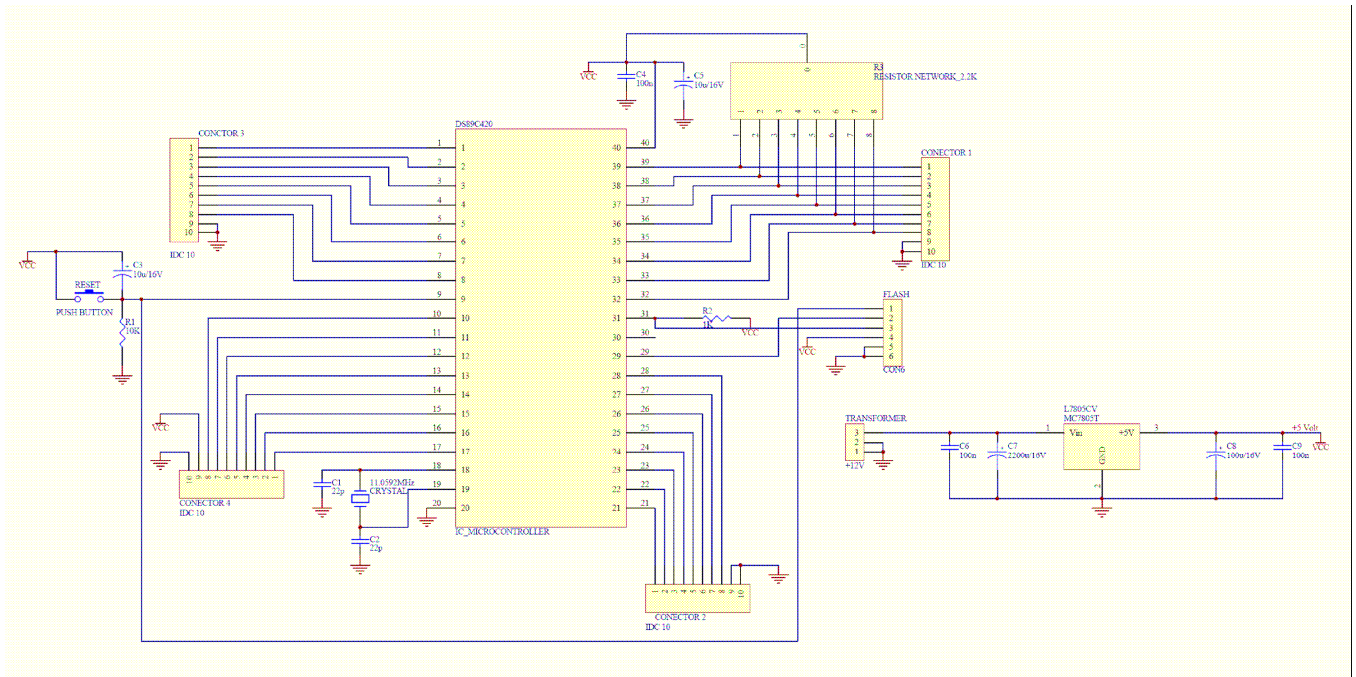
Σχήμα 4.4: Πίσω όψη της συσκευής. Βλέπουμε τις τροφοδοσίες της συσκευής 220V και +12V, την σειριακή θύρα όπου θα γίνετε η επικοινωνία με το κινητό τηλέφωνο, καθώς και τον συνδετήρα όπου θα συνδεθούν πάνω τα φορτία μας.



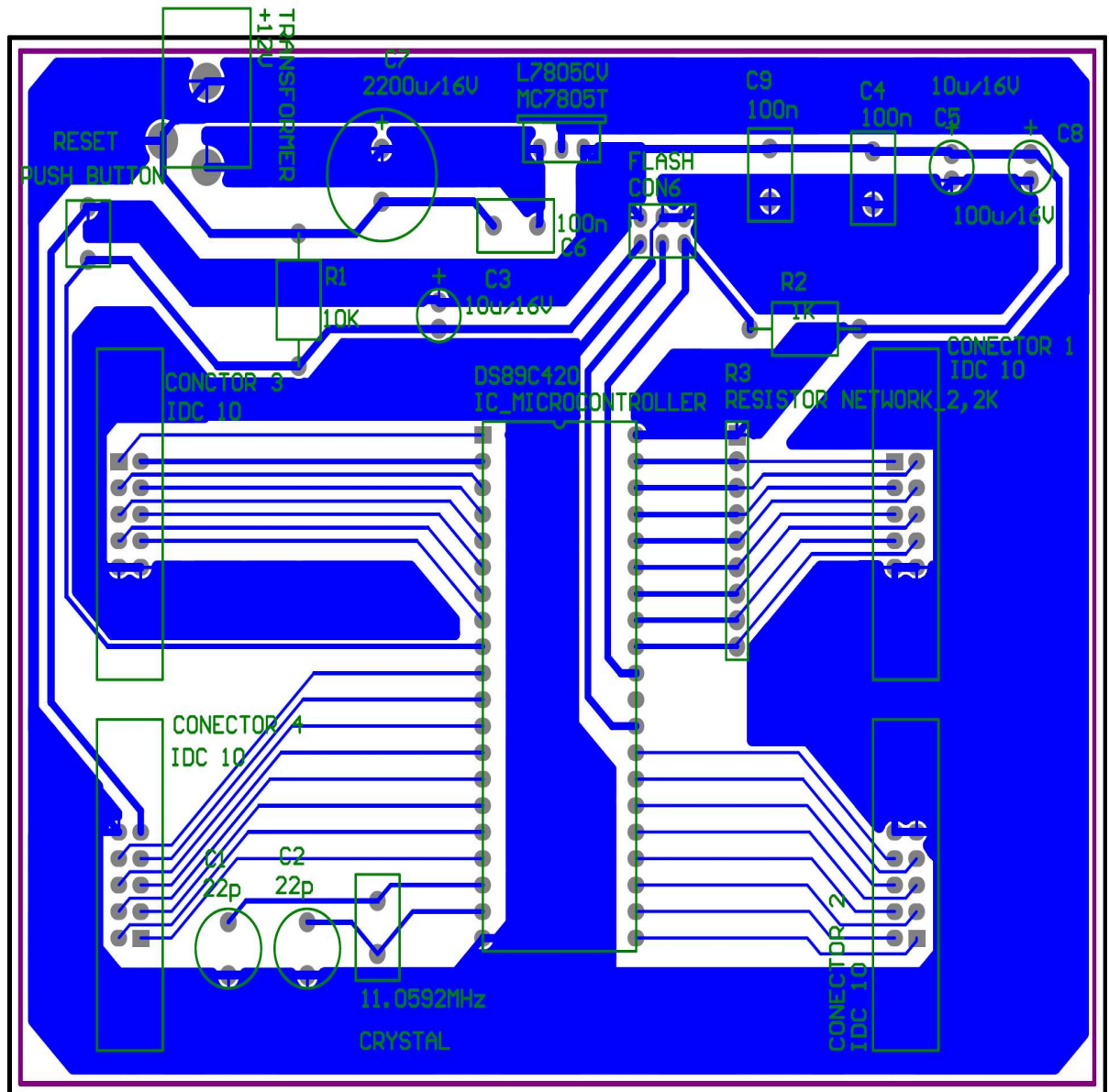
Σχήμα 4.5: Το εσωτερικό της κατασκευής. Βλέπουμε αριστερά την πλακέτα ισχύος, στην μέση την πλακέτα του επεξεργαστή και την πλακέτα του Max 232, και στην δεξιά μεριά βλέπουμε το κινητό μας τηλέφωνο της εταιρίας wavercom καθώς και την κεραία του τηλεφώνου πάνω δεξιά .

ΚΕΦΑΛΑΙΟ 5

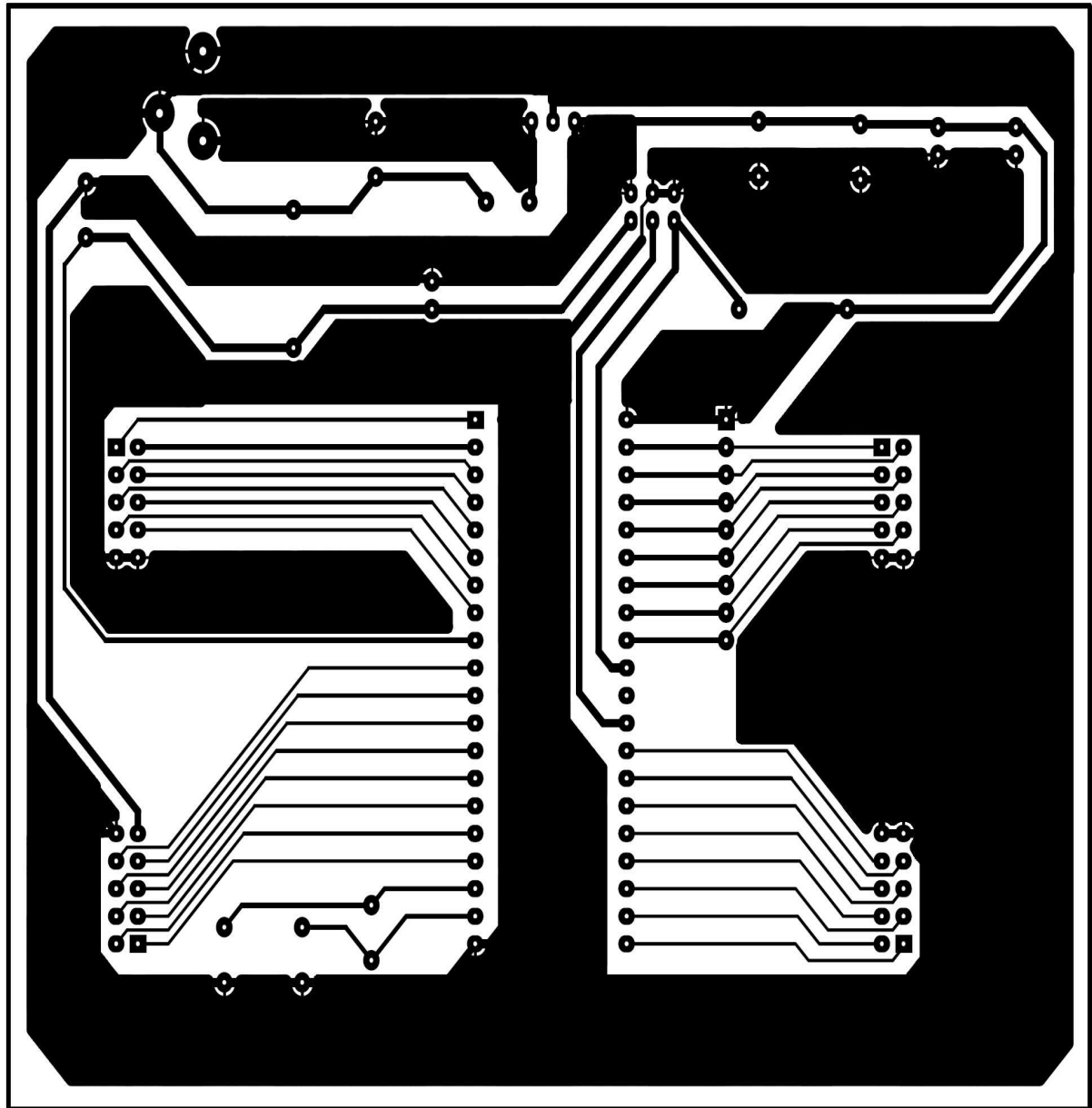
5.1 Τα μέρη από τα οποία αποτελείτε η κατασκευή μας.



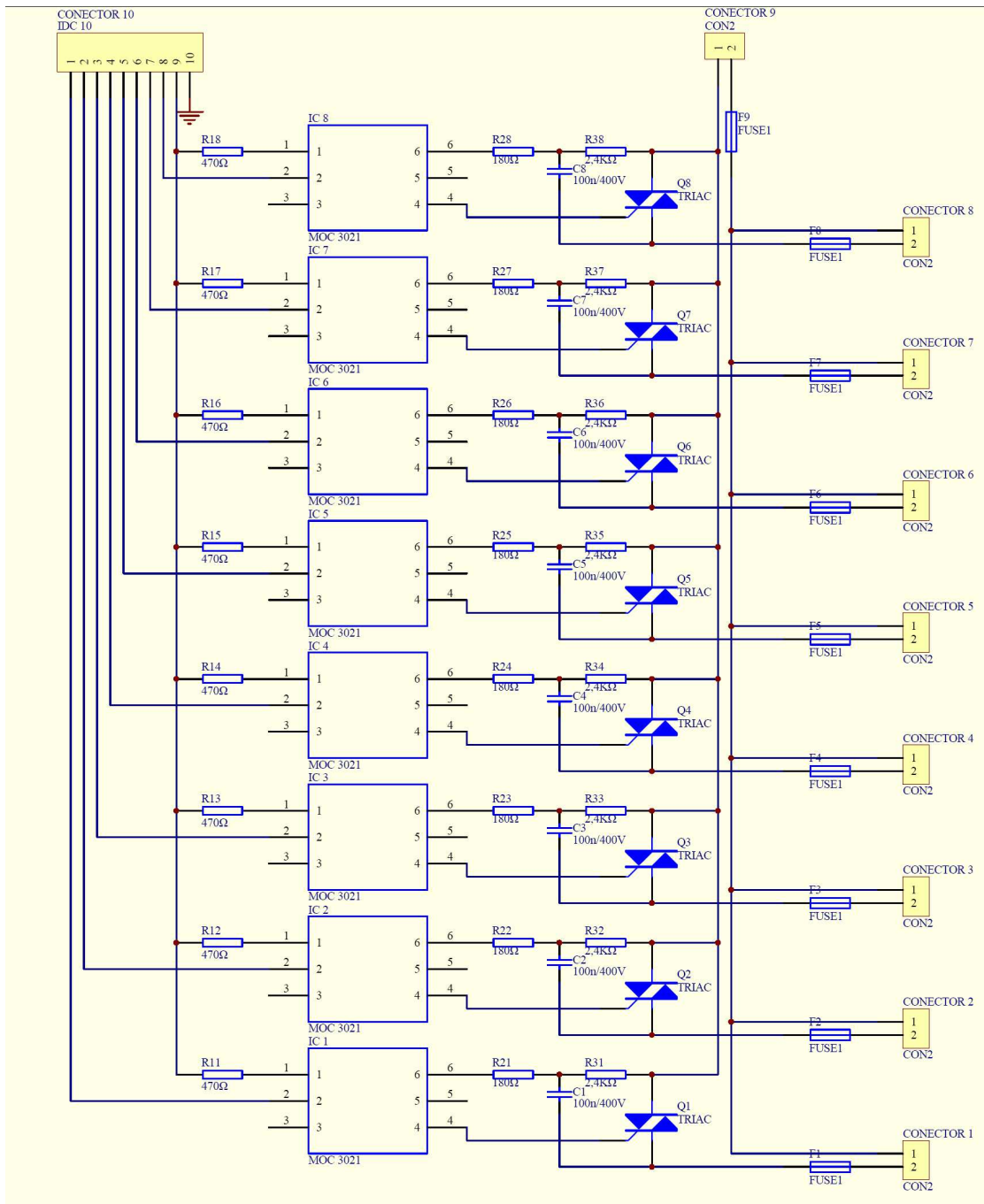
Σχήμα 5.1: το σχηματικό της πλακέτας του μικροελεγτή DS89C420.



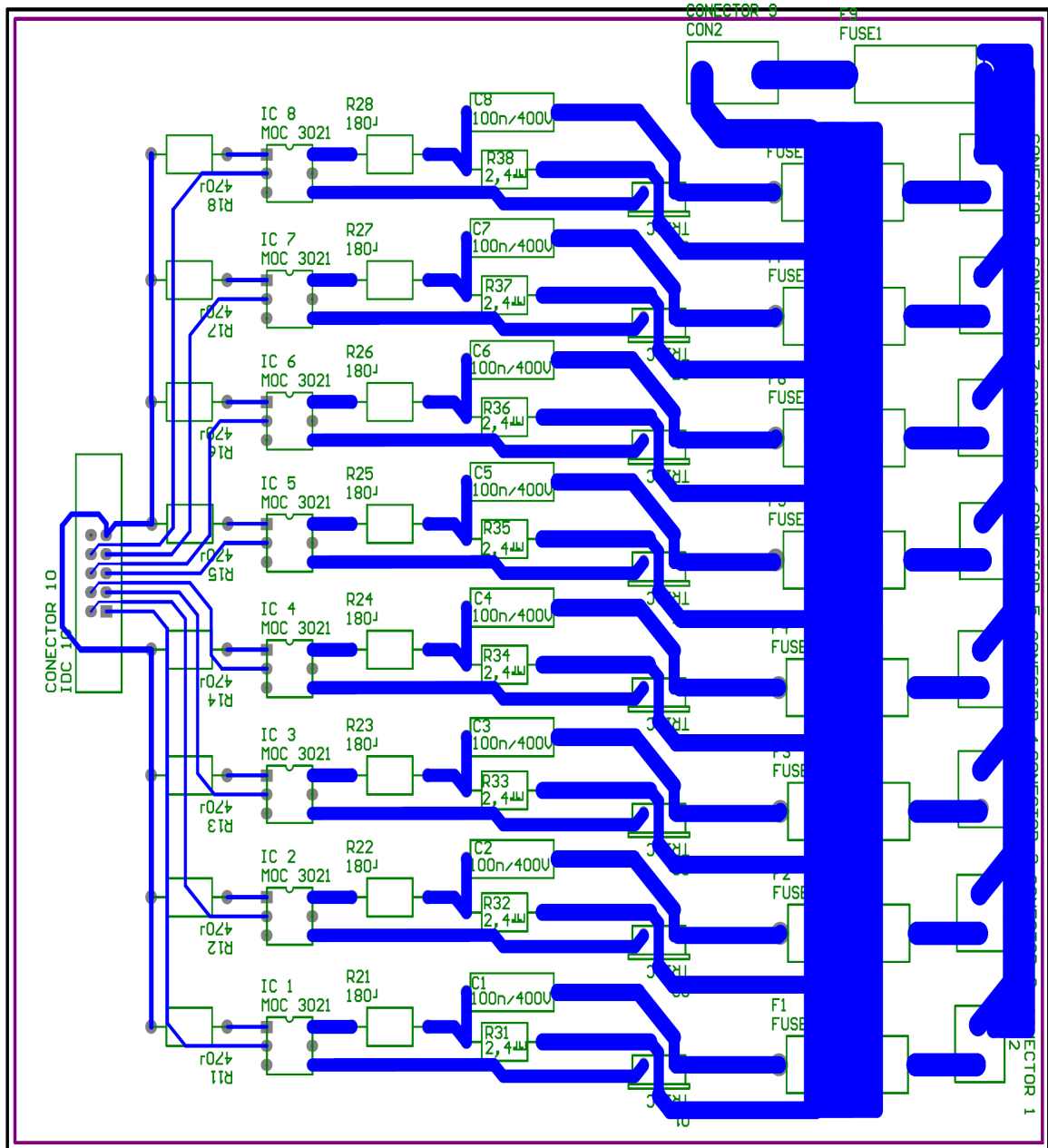
Σχήμα 5.2: Το PCB της πλακέτας του μικροελεγκτή όπως το βλέπουμε στο πρόγραμμα σχεδίασης Protel.



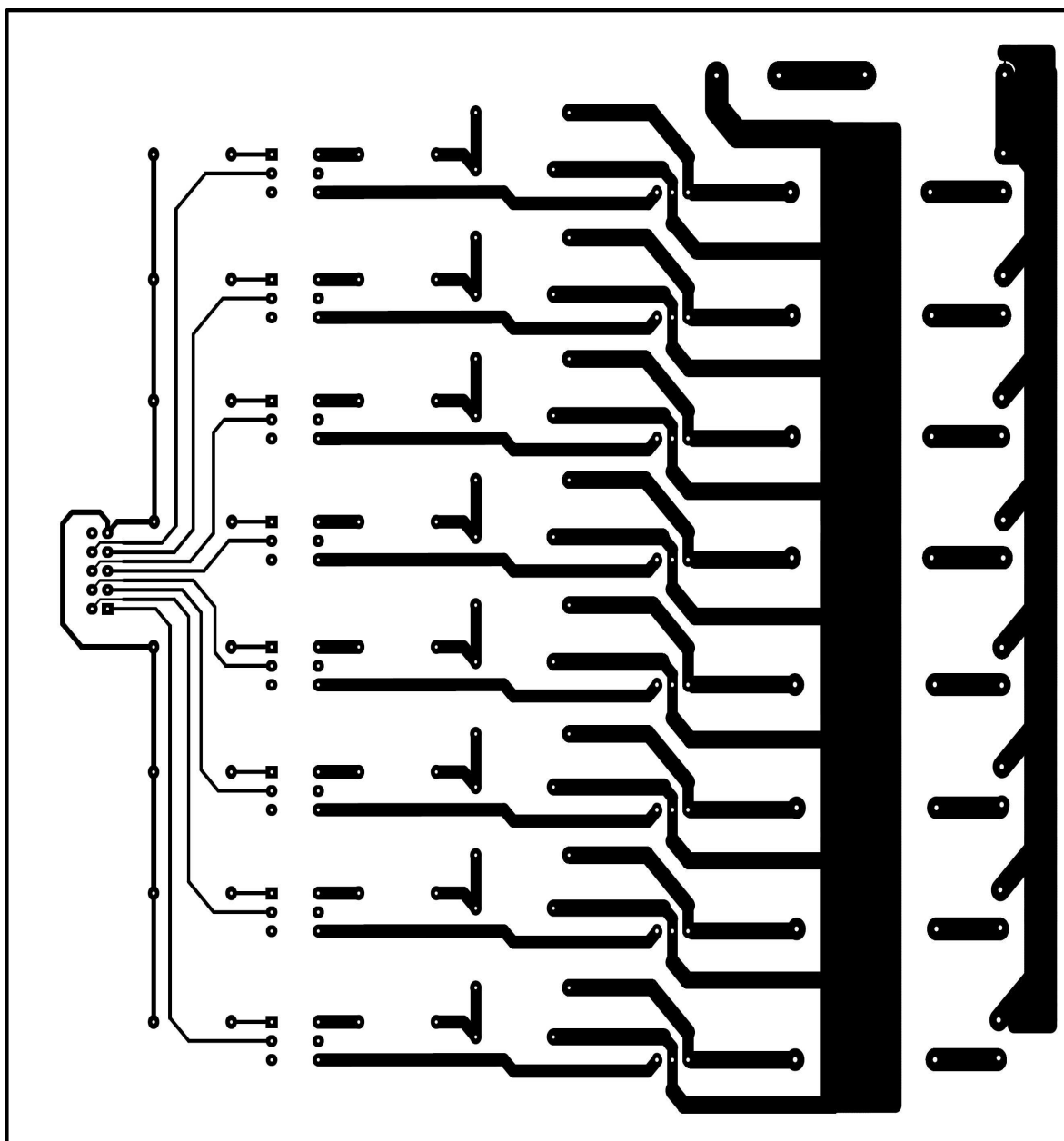
Σχήμα 5.3: Το PCB της πλακέτας του μικροελεγκτή όπως φαίνεται από την κάτω όψη.



Σχήμα 5.4: Το σχηματικό της πλακέτας ισχύος .

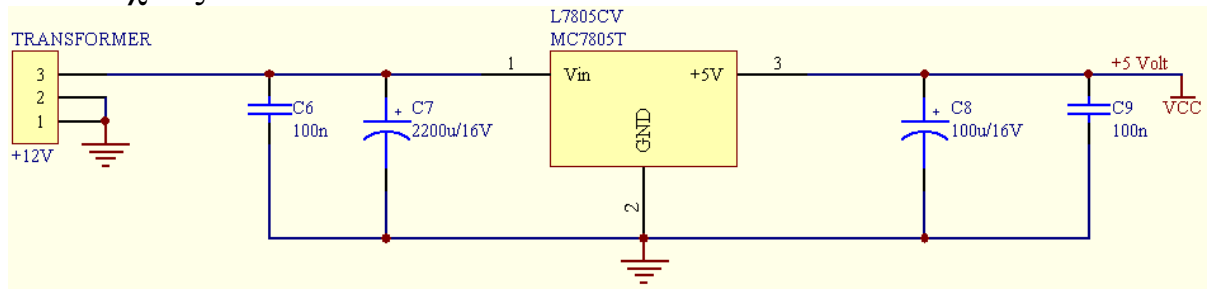


Σχήμα 5.5: Το PCB της πλακέτας ισχύος όπως το βλέπουμε στο πρόγραμμα σχεδίασης Protel.

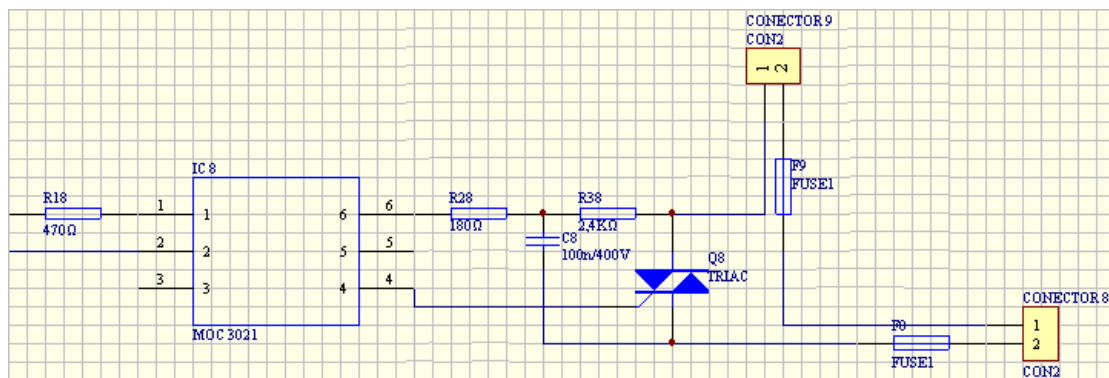


Σχήμα 5.6: Το PCB της πλακέτας ισχύος όπως φαίνεται από την κάτω όψη.

5.2 Σχηματικά του κυκλώματος τροφοδοσίας και της πλακέτας ισχύος



Σχήμα 5.7: Απλοποιημένο σχηματικό διάγραμμα του κυκλώματος τροφοδοσίας του μικροελεγκτή.



Σχήμα 5.8: Το βασικό σχηματικό διάγραμμα της πλακέτας ισχύος όπου ενεργοποιεί / απενεργοποιεί το triac ανάλογα με το σήμα που παίρνει από τον μικροελεγκτή.

Διασύνδεση των κυκλωμάτων ισχύος με τον μικροελεγκτή

Εδώ θα περιγράψουμε με λίγα λόγια πώς λειτουργεί η πλακέτα του μικροελεγκτή και πως γίνεται η διασύνδεση του μικροελεγκτή με την πλακέτα ισχύος. Έτσι με την ακόλουθη διαδικασία έχουμε την δυνατότητα να ενεργοποιούμε και να απενεργοποιούμε τα κατάλληλα mos καθώς και τα triac της κατασκευής μας.

Έτσι: Όταν έχουμε “0” από τον μικροεπεξεργαστή υπάρχει διαφορά δυναμικού και τα 5V περνάνε στην έξοδο, όταν έχουμε “1” δεν έχουμε διαφορά δυναμικού άρα δεν περνάνε.

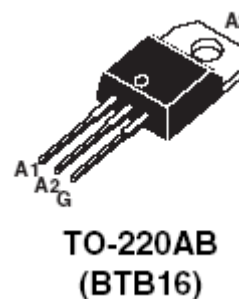
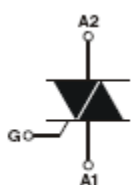
Το κάναμε αυτό για τον λόγο ότι με το που ανοίγει ο μικροεπεξεργαστής δίνει παντού “1” άρα θα άνοιγαν όλα τα triac μαζί κάτι που δεν το θέλουμε.

Οι αντιστάσεις R1 και R2 είναι για να μετριάζουν το ρεύμα που περνάει μέσα από το mos, επίσης έχουμε αντικαταστήσει τα ρελέ με τα triac ισχύος για να μπορούμε να οδηγούμε και μεγαλύτερα ρεύματα. Ακόμα με τα triac αποφεύγουμε τυχόν σπινθηρισμούς όπου πιθανών να μας δημιουργούσαν προβλήματα στην συσκευή μας, καθώς έχουμε και περισσότερη οικονομία ρεύματος σε σχέση με τα ρελέ.

Στο triac έχουμε μέσα στο ίδιο στοιχείο δηλ. σε ενιαία δομή τη διάταξη δύο αντιπαράλληλων θυρίστωρ S.C.R με ένα όμως, κοινό, ηλεκτρόδιο της πύλης. Μ’ αυτόν τον τρόπο το θυρίστωρ triac μπορεί από το κοινό ηλεκτρόδιο της πύλης G να σκανδαλιστεί τόσο με θετικό όσο και με αρνητικό παλμό.

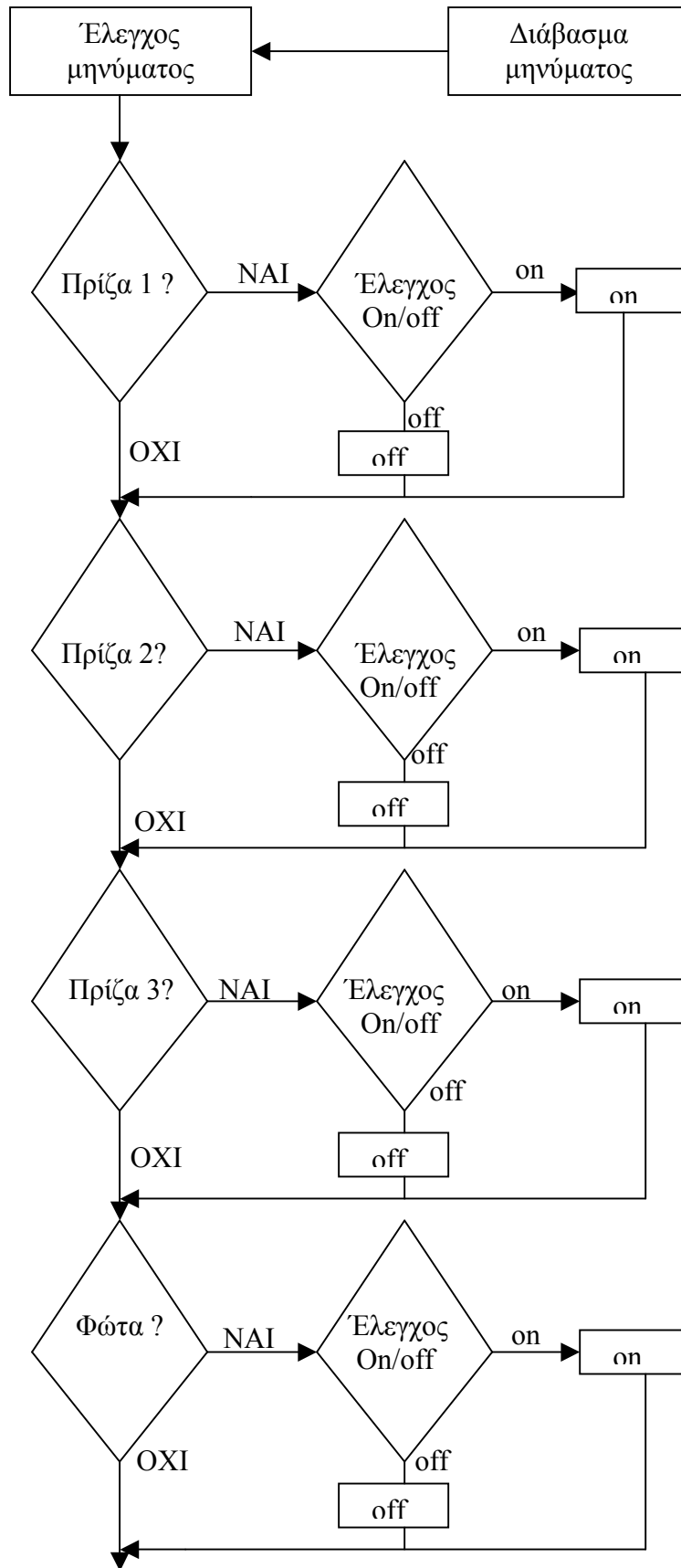
Η πολικότητα των παλμών έναυσης δεν επηρεάζει τη φορά αγωγιμότητας δηλ. είναι ανεξάρτητη από την πολικότητα που επικρατεί εκείνη την στιγμή στους κύριους ακροδέκτες A1, A2.

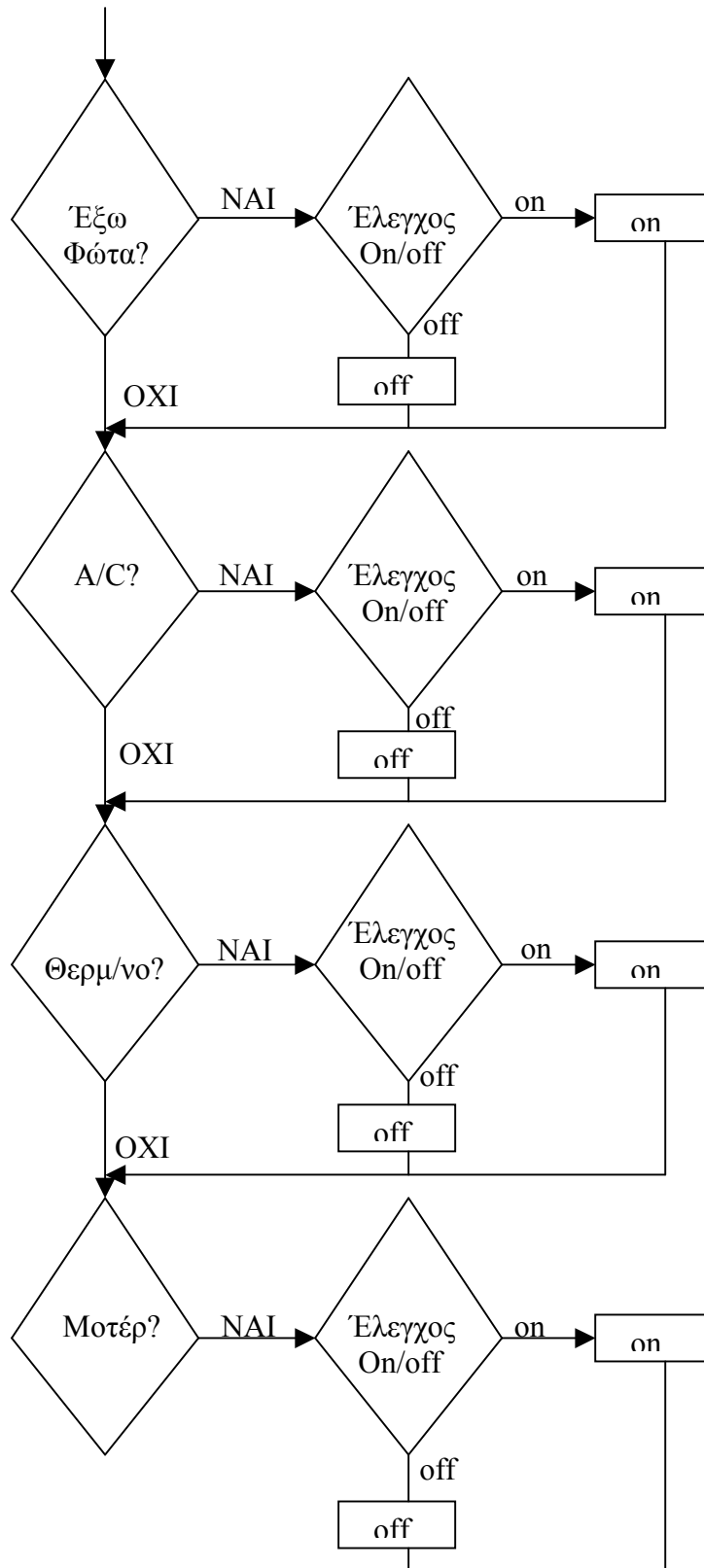
Το triac δεν χρησιμοποιείται για ανόρθωση, απλή ή πλήρης, παρά μόνο για τον αμφίπλευρο έλεγχο της ισχύος του εναλλασσόμενου ρεύματος από 100% έως 0. Συμπερασματικά λοιπόν επειδή το triac έχει δύο “back-to-back” διόδους μπορεί να άγει όταν οποιαδήποτε, εκείνη την στιγμή, είναι ορθά πολωμένη. Σε A.C λοιπόν λειτουργία το S.C.R άγει μόνο κατά τη θετική ημιπερίοδο ενώ το triac άγει και κατά τις δύο ημιπεριόδους.



Σχήμα 5.9: Το σχηματικό του triac TO- 220AB (BTB 16A).

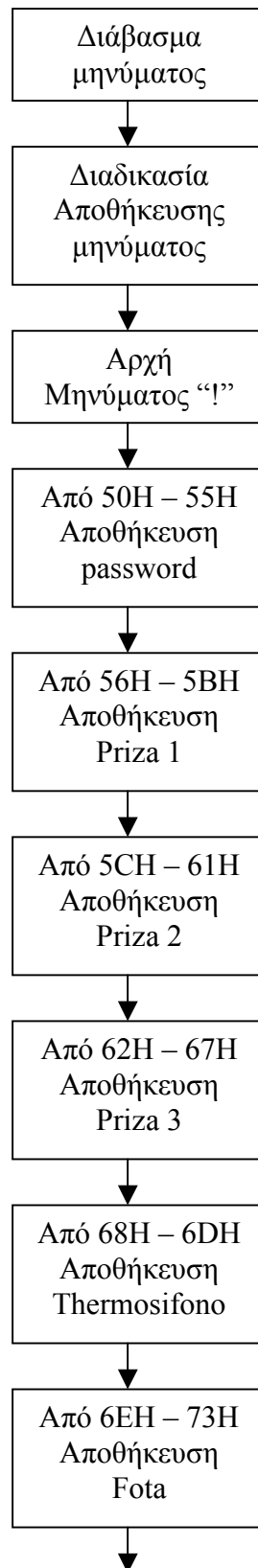
5.3 ΔΙΑΓΡΑΜΜΑ ΡΟΗΣ
 ΥΠΟ ΡΟΥΤΙΝΑΣ ΔΙΑΒΑΣΜΑΤΟΣ ΤΟΥ ΜΗΝΥΜΑΤΟΣ

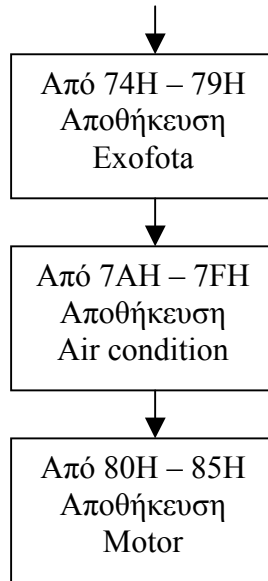




Η υπό ρουτίνα διαβάσματος και ελέγχου του μηνύματος λειτουργεί ως εξής:
Έρχεται το μήνυμα και αρχίζει αυτόματα η διαδικασία για τον έλεγχο, είναι η πρίζα 1; Αν **ναι** κάνει ένα δεύτερο έλεγχο όπου ελέγχει αν είναι **on** ή **off** για να κάνει "0" ή "1" την πόρτα p1.2 του μικροεπεξεργαστή. Όπου **ενεργοποιεί** ή **απενεργοποιεί** το τρίτο triac. Αν **όχι** προχωράει και ελέγχει παρακάτω είναι η πρίζα 2; Αν **ναι** κάνει ένα δεύτερο έλεγχο όπου ελέγχει αν είναι **on** ή **off** για να κάνει "0" ή "1" την πόρτα p1.3 του μικροεπεξεργαστή. Όπου **ενεργοποιεί** ή **απενεργοποιεί** το τέταρτο triac. Αν **όχι** προχωράει και ελέγχει παρακάτω είναι η πρίζα 3; Αν **ναι** ελέγχει αν είναι **on** ή **off** για να κάνει "0" ή "1" την πόρτα p1.4 του μικροεπεξεργαστή. Όπου **ενεργοποιεί** ή **απενεργοποιεί** το πέμπτο triac. Αν **όχι** προχωράει και ελέγχει παρακάτω είναι τα φώτα; Αν **ναι** ελέγχει αν είναι **on** ή **off** για να κάνει "0" ή "1" την πόρτα p1.0 του μικροεπεξεργαστή. Όπου **ενεργοποιεί** ή **απενεργοποιεί** το πρώτο triac. Αν **όχι** προχωράει και ελέγχει παρακάτω είναι τα έξω φώτα; Αν **ναι** ελέγχει αν είναι **on** ή **off** για να κάνει "0" ή "1" την πόρτα p1.1 του μικροεπεξεργαστή. Όπου **ενεργοποιεί** ή **απενεργοποιεί** το δεύτερο triac. Αν **όχι** προχωράει και ελέγχει παρακάτω είναι το aircondition; Αν **ναι** ελέγχει αν είναι **on** ή **off** για να κάνει "0" ή "1" την πόρτα p1.6 του μικροεπεξεργαστή. Όπου **ενεργοποιεί** ή **απενεργοποιεί** το έβδομο triac. Αν **όχι** προχωράει και ελέγχει παρακάτω είναι το θερμοσίφωνο; Αν **ναι** ελέγχει αν είναι **on** ή **off** για να κάνει "0" ή "1" την πόρτα p1.7 του μικροεπεξεργαστή. Όπου **ενεργοποιεί** ή **απενεργοποιεί** το όγδοο triac. Αν **όχι** προχωράει και ελέγχει παρακάτω είναι το μοτέρ; Αν **ναι** ελέγχει αν είναι **on** ή **off** για να κάνει "0" ή "1" την πόρτα p1.5 του μικροεπεξεργαστή. Όπου **ενεργοποιεί** ή **απενεργοποιεί** το έκτο triac.

5.4 ΔΙΑΓΡΑΜΜΑ ΡΟΗΣ
ΥΠΟ ΡΟΥΤΙΝΑΣ ΑΠΟΘΗΚΕΥΣΗΣ ΤΟΥ ΜΗΝΥΜΑΤΟΣ





Η διαδικασία της αποθήκευσης του μηνύματος είναι η εξής:

Το μήνυμα όταν ληφθεί από το κινητό το στέλνει στον μικροεπεξεργαστή όπου στην συνέχεια ξεκινά η διαδικασία αποθήκευσης του μηνύματος στην μνήμη του μικροεπεξεργαστή. Η αρχή του μηνύματος είναι πάντα με το χαρακτήρα “!” , και ακολουθεί ένα “**5ψήφιο**” password καθώς και τον χαρακτήρα “,” όπου αποθηκεύεται από την 50^H – 55^H θέση μνήμης. Στη συνέχεια αποθηκεύει τους χαρακτήρες “**priza1,**” από την 56^H – 5B^H θέση μνήμης, τους χαρακτήρες “**priza2,**” από την 5C^H – 61^H θέση μνήμης, τους χαρακτήρες “**priza3,**” από την 62^H – 67^H θέση μνήμης, Ακολουθεί η αποθήκευση των χαρακτήρων “**Thermosifono,**” από την 68^H – 6D^H θέση μνήμης, στη συνέχεια αποθηκεύει τους χαρακτήρες “**fota,**” από την 6E^H – 73^H θέση μνήμης, Ακολουθεί η αποθήκευση των χαρακτήρων “**Exofota,**” από την 74^H – 79^H θέση μνήμης, Μετά αποθηκεύει τους χαρακτήρες “**Aircondition,**” από την 7A^H – 7F^H θέση μνήμης, και τέλος αποθηκεύει τους χαρακτήρες “**Motor**” καθώς και τον χαρακτήρα “.” Όπου δηλώνει το τέλος του μηνύματος.

Βιβλιογραφία

- Σημειώσεις εργαστηρίου «Μικροϋπολογιστές Ι» Παν. Σολάνου. ΤΕΙ Κρήτης Παράρτημα Χανίων Τμήμα Ηλεκτρονικής.
- Σημειώσεις εργαστηρίου «Μικροελεκτές» Αντωνιδάκης – Μαρινάκης . ΤΕΙ Κρήτης Παράρτημα Χανίων Τμήμα Ηλεκτρονικής.
- Ultra High Speed Microcontroller DS89C420 Full Data Sheet . Maxim. Dallas semiconductor.
- AT Command Set for Wavecom AT x41 GSM Products.
- Integra modem M2106 Product Specification

Web Sites

- <http://www.maxim-ic.com>
- <http://www.wavecom.com>
- <http://www.alldatasheet.com>
- <http://www.roundsolutions.com>

ΚΩΔΙΚΑΣ

```
Defseg      start,absolute
seg         start
org         0000H
ljmp        main
org         000bh
ljmp        timer0_isr
org         0023h
ljmp        Serial_ISR

SerialRxPointer equ 08h
SerialRXmode   equ 0ah
MessageIndex   equ 0bh
DelayTime      equ 30h
ReceiveDelay   equ 31h
Instr          equ 32h
Phone         equ 90h

TimeFL        equ 20h.0
ReceiveTimeFL equ 20h.1
MessageFlag   equ 20h.2
Password      equ 20h.3

fota          equ p1.0
exofota       equ p1.1
Priza1        equ p1.2
Priza2        equ p1.3
Priza3        equ p1.4
motor         equ p1.5
Aircondition  equ p1.6
Thermosifono  equ p1.7
Koudouni      equ p2.0
Enable_Disable equ p2.1
OneSecond     equ 20
SerialRxBuffer equ 39h

org         0070H
main:
mov         ie,#0
mov         ip0,#0
mov         ip1,#0
mov         sp,#0b0h           ;Initialise Stack
;orl        pmr,#00000001b
mov         psw,#0
```

```

mov    t10,#low(-46080)    ; Reload @50msec
mov    th0,#high(-46080)   ; Reload @50msec
mov    tmod,#00000001b     ; Mode 1
clr    tf0
setb   tr0
setb   et0
setb   ea

mov    scon,#50H
mov    th1,#-3
orl    tmod,#00100000B
setb   tr1

setb   rs0
mov    r2,#0
clr    rs0
setb   koudouni
clr    messageflag
mov    p1,#0ffh
mov    p0,#0ffh
setb   Enable_Disable
acall  delay_1sec
mov    p1,#0a5h
acall  delay_1sec
mov    p1,#0ffh

acall  delay_1sec
mov    dptr,#speaker
acall  sendtoserialA
acall  delay_1sec
mov    dptr,#gainR
acall  sendtoserialA
acall  delay_1sec
mov    dptr,#gainT
acall  sendtoserialA
;*****
;*****
acall  readall
mov    p0,p1
MainLoop:
mov    sp,#0b0h
clr    rs0
clr    rs1
clr    ri
clr    ti
clr    MessageFlag
setb   es
mess:
p0,p1
jb    Enable_Disable,Dis
mov    p1,#0ffh
sjmp  mess

```

```

Dis:          jb MessageFlag,Minim      ;check if message received
              Jb koudouni,mess
              acall klisi
              sjmp mainloop
minim:        acall delay_1sec
              acall clear_ram
              mov dptr,#ReadIndex
              acall sendtoserialA
              acall readmessage      ;read message
              acall decodemessage
              mov dptr,#DeleteIndex
              acall sendtoserialA
              jnb password,LL2
              acall SendAnswer
LL2:          clr MessageFlag
              Clr ti
              clr ri
              sjmp mainloop
;*****
;*****
; Read All Messages At Startup
readall:
              mov MessageIndex,#030h
RA1:          inc MessageIndex
              acall delay_1sec
              acall clear_ram
              mov dptr,#ReadIndex
              acall sendtoserialA
              acall readmessage
              acall decodemessage
              mov dptr,#DeleteIndex
              acall sendtoserialA
              acall SendAnswer
              mov a,messageindex
              cjne a,#038h,RA1
              ret
;*****
;*****
klisi:        MOV DPTR,#Thlefono
              ACALL SEND_MESSAGE
              ACALL DELAY_12SEC
              ACALL DELAY_12SEC
              MOV DPTR,#HUNG_UP
              ACALL SEND_MESSAGE
              RET
              ret
;*****
;*****

```

```

SendAnswer:   clr es
              acall delay_2sec
              mov DPTR,#CMGS
              acall send_message
              mov r1,#Phone
              acall SendPhone
              acall delay_3sec
              mov a,sbuf
              mov a,#'P'
              acall send_char
              mov a,#'R'
              acall send_char
              mov a,#'1'
              acall send_char
              mov a,#'='
              acall send_char
              jb priza1, L1
              mov a,#'A'
              sjmp L2
L1:           mov a,#'K'
L2:           acall send_char
              mov a,#0dh ;','
              acall send_char
              mov a,#'P'
              acall send_char
              mov a,#'R'
              acall send_char
              mov a,#'2'
              acall send_char
              mov a,#'='
              acall send_char
              jb priza2, L12
              mov a,#'A'
              sjmp L22
L12:         mov a,#'K'

```

```

L22:      acall send_char
          mov a,#0dh ;','
          acall send_char
          mov a,#'P'
          acall send_char
          mov a,#'R'
          acall send_char
          mov a,#'3'
          acall send_char
          mov a,#'='
          acall send_char
          jb priza3, L13
          mov a,#'A'
          sjmp L23
L13:      mov a,#'K'
L23:      acall send_char
          mov a,#0dh ;','
          acall send_char
          mov a,#'F'
          acall send_char
          mov a,#'O'
          acall send_char
          mov a,#'T'
          acall send_char
          mov a,#'='
          acall send_char
          jb fota,L14
          mov a,#'A'
          sjmp L24
L14:      mov a,#'K'
L24:      acall send_char
          mov a,#0dh ;','
          acall send_char
          mov a,#'E'
          acall send_char
          mov a,#'F'
          acall send_char
          mov a,#'1'
          acall send_char
          mov a,#'='
          acall send_char
          jb Exofota, L15
          mov a,#'A'
          sjmp L25
L15:      mov a,#'K'

```

```

L25:      acall send_char
          mov a,#0dh ;','
          acall send_char
          mov a,#'T'
          acall send_char
          mov a,#'H'
          acall send_char
          mov a,#'E'
          acall send_char
          mov a,#'='
          acall send_char
          jb thermosifono, L16
          mov a,#'A'
          sjmp L26
L16:      mov a,#'K'
L26:      acall send_char
          mov a,#0dh ;','
          acall send_char
          mov a,#'A'
          acall send_char
          mov a,#'/'
          acall send_char
          mov a,#'C'
          acall send_char
          mov a,#'='
          acall send_char
          jb Aircondition, L17
          mov a,#'A'
          sjmp L27
L17:      mov a,#'K'
L27:      acall send_char
          mov a,#0dh ;','
          acall send_char
          mov a,#'M'
          acall send_char
          mov a,#'O'
          acall send_char
          mov a,#'T'
          acall send_char
          mov a,#'='
          acall send_char
          jb motor, L18
          mov a,#'A'
          sjmp L28
L18:      mov a,#'K'

```

```

L28:          acall send_char
              mov a,#'.'
              acall send_char
              mov a,#1ah
              acall send_char
              acall delay_2sec
              ret
;*****
;*****
Serial_ISR:
              Jb ri,Sr1
              reti
Sr1:          push psw
              Push acc
              setb rs0
              clr rs1
              clr ri
              cjne r2,#00,Sr2
              mov a,sbuf
              cjne a,#',',Sr_Exit
              mov r2,#01
              ljmp Sr_Exit
Sr2:          cjne r2,#01,Sr3
              Mov MessageIndex,sbuf
              mov r2,#00
              clr es
              setb MessageFlag
Sr3:
Sr_Exit:     Pop acc
              pop psw
              reti
;*****
;*****
ReadMessage:
              Mov r0,#50h
              Mov r7,#100
              clr ReceiveTimeFL
              mov ReceiveDelay,#(OneSecond*4)
rm1:         clr ri
rm1a:        jnb ri,rt1
              sjmp rm11
rt1:         jnb ReceiveTimeFL,rm1a
              sjmp message_error
rm11:        djnz r7,rm2
              sjmp message_error
rm2:         mov a,sbuf
              cjne a,#',',rm1
              clr ReceiveTimeFL
              clr ri

```

```

rr2:      jnb ri,rr1
          sjmp rma3
rr1:      jnb ReceiveTimeFL,rr2
          ljmp message_error
rma3:     mov a,sbuf
          mov r1,#Phone
rma4:     clr ri
rr5:      jnb ri,rr4
          sjmp rr6
rr4:      jnb ReceiveTimeFL,rr5
          ljmp message_error
rr6:      mov a,sbuf
          cjne a,#"",rm443
          mov @r1,#0
          sjmp rmc3
rm443:    mov @r1,a
          inc r1
          sjmp rma4
rmc3:     clr ri
rr8:      jnb ri,rr7
          sjmp rr9
rr7:      jnb ReceiveTimeFL,rr8
          ljmp message_error
rr9:      mov a,sbuf
          cjne a,#',,rmc3
          clr ri
rmb3:     mov r7,#100
          clr ReceiveTimeFL
          mov ReceiveDelay,#OneSecond
rm4:      clr ri
rm44:     jnb ri,rt2
          sjmp rm55
rt2:      jnb ReceiveTimeFL,rm44
          sjmp message_error
rm55:     djnz r7,rm3
          sjmp message_error
rm3:      mov a,sbuf
          mov @r0,a
          inc r0
          cjne a,#',,rm4
;         mov p1,#0xfd
          ret
message_error:
          mov r0,#50h
DL1:      mov @r0,#0
          Inc r0
          cjne r0,#87h,DL1
          ret
.*****
*****

```



```

decodemessage:
    acall checkpassword
    jnb password,passerror
    mov instr,#56h
    acall checkinstruction
    mov instr,#5ch
    acall checkinstruction
    mov instr,#62h
    acall checkinstruction
    mov instr,#68h
    acall checkinstruction
    mov instr,#6eh
    acall checkinstruction
    mov instr,#74h
    acall checkinstruction
    mov instr,#7ah
    acall checkinstruction
    mov instr,#80h
    acall checkinstruction

```

```

passerror:
    ret

```

```

,*****
*****

```

```

checkinstruction:
    mov r0,instr
    mov a,@r0
    cjne a,#'P',CH1
    inc r0
    mov a,@r0
    cjne a,#'R',CH1
    inc r0
    mov a,@r0
    cjne a,#'1',CH1
    inc r0
    mov a,@r0
    cjne a,#'=',CH1
    inc r0
    mov a,@r0
    cjne a,#'K',CP1
    setb Priza1
    sjmp exit_ch
CP1:
    cjne a,#'A',exit_ch
    clr Priza1
exit_ch:
    ljmp CHEXIT

```

```

CH1:      mov r0,instr
          Mov a,@r0
          Cjne a,#'P',CH1a
          Inc r0
          Mov a,@r0
          Cjne a,#'R',CH1a
          Inc r0
          Mov a,@r0
          Cjne a,#'2',CH1a
          Inc r0
          Mov a,@r0
          Cjne a,#'=',CH1a
          Inc r0
          Mov a,@r0
          Cjne a,#'K',CP1a
          setb Priza2
          sjmp exit_cha
CP1a:    cjne a,#'A',exit_cha
          Clr Priza2
exit_cha:
          ljmp CHEXIT
CH1a:    mov r0,instr
          Mov a,@r0
          cjne a,#'P',CH2
          inc r0
          mov a,@r0
          cjne a,#'R',CH2
          inc r0
          mov a,@r0
          cjne a,#'3',CH2
          inc r0
          mov a,@r0
          cjne a,#'=',CH2
          inc r0
          mov a,@r0
          cjne a,#'K',CP1b
          setb Priza3
          sjmp exit_chb
CP1b:    cjne a,#'A',exit_chb
          Clr Priza3
exit_chb:
          ljmp CHEXIT

```

```

CH2:    mov r0,instr
        Mov a,@r0
        cjne a,#'T',CH3
        inc r0
        mov a,@r0
        cjne a,#'H',CH3
        inc r0
        mov a,@r0
        cjne a,#'E',CH3
        inc r0
        mov a,@r0
        cjne a,#'=',CH3
        inc r0
        mov a,@r0
        cjne a,#'K',CP1c
        setb Thermosifono
        sjmp exit_chc
CP1c:   cjne a,#'A',exit_chc
        Clr Thermosifono
exit_chc:
        ljmp CHEXIT
CH3:    mov r0,instr
        Mov a,@r0
        cjne a,#'F',CH4
        inc r0
        mov a,@r0
        cjne a,#'O',CH4
        inc r0
        mov a,@r0
        cjne a,#'T',CH4
        inc r0
        mov a,@r0
        cjne a,#'=',CH4
        inc r0
        mov a,@r0
        cjne a,#'K',CP1d
        setb Fota
        sjmp exit_chd
CP1d:   cjne a,#'A',exit_chd
        Clr Fota

```

```

exit_chd:
    ljmp CHEXIT
CH4:
    mov r0,instr
    Mov a,@r0
    cjne a,#'E',CH5
    inc r0
    mov a,@r0
    cjne a,#'F',CH5
    inc r0
    mov a,@r0
    cjne a,#'1',CH5
    inc r0
    mov a,@r0
    cjne a,#'=',CH5
    inc r0
    mov a,@r0
    cjne a,#'K',CP1e
    setb ExoFota
    sjmp exit_che
CP1e:
    cjne a,#'A',exit_che
    Clr ExoFota
exit_che:
    ljmp CHEXIT
CH5:
    mov r0,instr
    Mov a,@r0
    cjne a,#'A',CH6
    inc r0
    mov a,@r0
    cjne a,#'/',CH6
    inc r0
    mov a,@r0
    cjne a,#'C',CH6
    inc r0
    mov a,@r0
    cjne a,#'=',CH6
    inc r0
    mov a,@r0
    cjne a,#'K',CP1f
    setb Aircondition
    sjmp exit_chf
CP1f:
    cjne a,#'A',exit_chf
    Clr Aircondition

```

```

exit_chf:
    ljmp CHEXIT
CH6:
    mov r0,instr
    Mov a,@r0
    cjne a,#'M',CHEXIT
    inc r0
    mov a,@r0
    cjne a,#'O',CHEXIT
    inc r0
    mov a,@r0
    cjne a,#'T',CHEXIT
    inc r0
    mov a,@r0
    cjne a,#'=',CHEXIT
    inc r0
    mov a,@r0
    cjne a,#'K',CP1g
    setb Motor
    sjmp exit_chg
CP1g:
    cjne a,#'A',CHEXIT
    Clr Motor
exit_chg:
CHEXIT: ret

```

```

;*****
;*****

```

```

checkpassword:
    mov instr,#50h
    mov r0,instr
    mov a,@r0
    cjne a,#'2',pigene
    inc r0
    mov a,@r0
    cjne a,#'2',pigene
    inc r0
    mov a,@r0
    cjne a,#'2',pigene
    inc r0
    mov a,@r0
    cjne a,#'2',pigene
    inc r0
    mov a,@r0
    cjne a,#'2',pigene
    setb password
    ret

```

```

pigene:
    clr password
    ret

```

```

;*****
;*****

```

```

; mov dptr,#send_sms
; acall send_message
; acall delay_1sec
; mov dptr,#mhnima
; acall send_Mynima
; sjmp $

send_message:
    clr a
    movc a,@a+dptr
    jz oxo
    acall send_char
    inc dptr
    sjmp send_message
oxo:    ret
,
*****
*****

SendPhone:
    Mov a,@r1
    Jz oxo3
    Acall send_char
    Inc r1
    Sjmp sendphone
oxo3:  mov a,#0ah
    acall send_char
    ret

,
*****
*****

SendToSerial:
    Clr a
    Movc a,@a+dptr
    Jz oxo1
    Acall send_char
    Inc dptr
    Sjmp SendToSerial
oxo1:  ret
,
*****
*****

SendToSerialA:
    clr a
    Movc a,@a+dptr
    jz oxo1a
    acall send_char
    inc dptr
    sjmp SendToSerialA

```

```

oxo1a: mov a,MessageIndex
        acall send_char
        mov a,#0ah
        acall send_char
        ret
;*****
;*****
SEND_CHAR: clr ti
           MOV SBUF,A
           JNB TI,$
           CLR TI
           RET
;*****
;*****
GET_CHAR: JNB RI,$
           CLR RI
           MOV A,SBUF
           RET
;*****
;*****
clear_ram: mov r0,#050h
CR1:      mov @r0,#0
           Inc r0
           Cjne r0,#090h,CR1
           ret
;*****
;*****
delay_1sec: mov delaytime,#OneSecond
            clr timefl
            jnb timefl,$
            ret
delay_2sec: mov delaytime,#(OneSecond*2)
            clr timefl
            jnb timefl,$
            ret
delay_3sec: mov delaytime,#(OneSecond*3)
            clr timefl
            jnb timefl,$
            ret
delay_4sec: mov delaytime,#(OneSecond*4)
            clr timefl
            jnb timefl,$
            ret
delay_5sec: mov delaytime,#(OneSecond*5)
            clr timefl
            jnb timefl,$
            ret

```

```

delay_6sec: mov delaytime,#(OneSecond*6)
            clr timefl
            jnb timefl,$
            ret
delay_7sec: mov delaytime,#(OneSecond*7)
            clr timefl
            jnb timefl,$
            ret
delay_8sec: mov delaytime,#(OneSecond*8)
            clr timefl
            jnb timefl,$
            ret
delay_9sec: mov delaytime,#(OneSecond*9)
            clr timefl
            jnb timefl,$
            ret
delay_10sec: mov delaytime,#(OneSecond*10)
            clr timefl
            jnb timefl,$
            ret
delay_11sec: mov delaytime,#(OneSecond*11)
            clr timefl
            jnb timefl,$
            ret
delay_12sec: mov delaytime,#(OneSecond*12)
            clr timefl
            jnb timefl,$
            ret

```

```

,*****
**

```

```

send_mynima:
            acall    send_message
            acall    delay
            mov     a,#1ah
            acall    send_char
            ret

```

```

,*****

```

```

delay: mov     r4,#15
La3:  mov     r5,#200
La2:  mov     r6,#8
La1:  mov     r7,#250
      djnz   r7,$
      djnz   r6,La1
      djnz   r5,La2
      djnz   r4,La3
      ret

```



```

;----- RTC Subroutine -----
Timer0_Isr: mov     th0,#high(-46076)      ; Reload @50msec
             mov     tl0,#low(-46076) ; Reload @50msec
             clr     tf0
             djnz    DelayTime,DT1a
             setb    TimeFL
DT1a:       djnz    ReceiveDelay,DT2a
             setb    ReceiveTimeFL
DT2a:       reti
;-----
MHNYMA:     DB 'Hello John',0
SEND_SMS:   DB 'AT+CMGS=Αριθμός τηλεφώνου',0aH,0
CMGS:       DB 'AT+CMGS=',0
DELETE_ALL: db 'AT+CMGD=1,4',0ah,0
ReadIndex:  db 'AT+CMGR=',0
DeleteIndex: db 'AT+CMGD=',0
READ1:      db 'AT+CMGR=1',0ah,0
READ2:      db 'AT+CMGR=2',0ah,0
READ3:      db 'AT+CMGR=3',0ah,0
READ4:      db 'AT+CMGR=4',0ah,0
READ5:      db 'AT+CMGR=5',0ah,0
READ6:      db 'AT+CMGR=6',0ah,0
READ7:      db 'AT+CMGR=7',0ah,0
READ8:      db 'AT+CMGR=8',0ah,0
Clear1:     db 'AT+CMGD=1',0ah,0
Clear2:     db 'AT+CMGD=2',0ah,0
Clear3:     db 'AT+CMGD=3',0ah,0
Clear4:     db 'AT+CMGD=4',0ah,0
Clear5:     db 'AT+CMGD=5',0ah,0
Clear6:     db 'AT+CMGD=6',0ah,0
Clear7:     db 'AT+CMGD=7',0ah,0
Clear8:     db 'AT+CMGD=8',0ah,0
Speaker:    db 'AT+SPEAKER=1',0ah,0
GainT:      DB 'AT+VGT=200',0dh      ; Microphone Gain (0-255)
GainR:      DB 'AT+VGR=200',0dh      ; Speaker Gain (0-255)
Passworda:  db '22222',0
thlefono:   db 'ATD Αριθμός τηλεφώνου;',13,10,0
Hung_up:    db 'ATH1',13,10,0

```

END