



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ & ΣΥΣΤΗΜΑΤΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ

«ΠΑΝΖΕΥΚΤΗΣ»: Ολοκληρωμένη Πλατφόρμα Υλικού
Ανάπτυξης Εφαρμογών Διάχυτης Ευφυΐας, Ασύρματης
Εποπτείας και Ευφυούς Ελέγχου Οικίας (domotics) με
Προηγμένες Υπολογιστικές Δυνατότητες.

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Επιβλέπων : Ευστάθιος Δ. Σοκάς

Καθηγητής Ε.Μ.Π.

Αθήνα, Σεπτέμβριος 2011



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ & ΣΥΣΤΗΜΑΤΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ

«ΠΑΝΖΕΥΚΤΗΣ»: Ολοκληρωμένη Πλατφόρμα Υλικού
Ανάπτυξης Εφαρμογών Διάχυτης Ευφυΐας, Ασύρματης
Εποπτείας και Ευφυούς Ελέγχου Οικίας (domotics) με
Προηγμένες Υπολογιστικές Δυνατότητες.

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Επιβλέπων : Ευστάθιος Δ. Συκάς

Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την

.....

Μιχαήλ Θεολόγου

Καθηγητής Ε.Μ.Π.

.....

ς Παπανάνος

ητής Ε.Μ.Π.

Αθήνα, Σεπτέμβριος 2011

.....
Ισαάκ Χ.Σολωμονίδης

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © Ισαάκ Σολωμονίδης, 2011

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Περίληψη

Σκοπός της παρούσας διπλωματικής εργασίας ήταν η σχεδίαση, κατασκευή και δοκιμή μιας ολοκληρωμένης πλατφόρμας υλικού για την ανάπτυξη εφαρμογών διάχυτης ευφυίας, ασύρματης εποπτείας και ευφυούς ελέγχου οικίας (domotics), με προηγμένες υπολογιστικές δυνατότητες. Ονόμασα αυτή τη συσκευή «ΠΑΝΖΕΥΚΤΗ».

Παράλληλα, ανέπτυξα και περιφερειακή συσκευή η οποία επικοινωνεί ασύρματα με τον «Πανζεύκτη» και προσφέρει πολλαπλή λειτουργικότητα εισόδου-εξόδου και μετρήσεων, ώστε να διευκολύνεται η άμεση υλοποίηση εφαρμογών. Ονόμασα αυτή τη συσκευή «Panzerterminal».

Ο ισχυρότατος μικροελεγκτής 32-bit ARM Cortex-M3 50-MHz που ο «Πανζεύκτης» διαθέτει, επιτρέπει την αποτελεσματική ανάπτυξη σύγχρονων εφαρμογών με υψηλές απαιτήσεις. Ακόμη, ο «Πανζεύκτης» προσφέρει εκτεταμένη συνδεσιμότητα καθώς διαθέτει Ethernet, μόντεμ XBEE (επιλογή ανάμεσα σε ZigBee, 802.15.4, Wi-Fi ή και άλλα ασύρματα δίκτυα), RS232, IrDA, USB(COM). Έτσι, σε συνδυασμό με μία ή περισσότερες τερματικές μονάδες «Panzerterminal» προσφέρεται ένα πλήρες φάσμα από χρήσιμες λειτουργίες.

Παρουσιάζεται η πλήρης θεωρητική μελέτη για την κατασκευή, καθώς και το υλικολογισμικό που έχει αναπτυχθεί για την παρουσίαση των λειτουργιών, μέσα από το φιλικό περιβάλλον μιας ενσωματωμένης web εφαρμογής. Η πρόσβαση σε όλες τις λειτουργίες της συσκευής γίνεται μέσα από τον φυλλομετρητή ιστού του χρήστη.

Η εργασία αυτή φιλοδοξεί:

- Να αποτελέσει μια ισχυρή πλατφόρμα ανάπτυξης ενσωματωμένων εφαρμογών, ιδανική για εφαρμογές Διάχυτης Ευφυίας, έχοντας ως βάση έναν ισχυρότατο ARM® 32bit μικροελεγκτή και παρέχοντας έτοιμη προς χρήση μια ποικιλία διασυνδέσεων, μέσω αποθήκευσης και εισόδων-εξόδων μέτρησης και ελέγχου.
- Να παρουσιάσει μια πρόταση για τη διασύνδεση συσκευών ή μεμονωμένων αισθητήρων και ενεργοποιητών με τον χρήστη, οπουδήποτε και αν αυτός βρίσκεται, μέσα από ένα φιλικό περιβάλλον, με τη χρήση ενός απλού περιηγητή ιστού.
- Να διευκολύνει σημαντικά την υλοποίηση εξειδικευμένων συσκευών τηλεμετρίας ή άλλων, καθώς η πολυεπίπεδη αρχιτεκτονική στην υλοποίηση της κεντρικής πλακέτας επιτρέπει την απρόσκοπτη προσθήκη επιπλέον πλακετών ειδικής χρήσης.
- Να δώσει τη δυνατότητα πειραματισμού στις time-critical εφαρμογές, καθώς υποστηρίζεται η υλοποίηση με λειτουργικό σύστημα πραγματικού χρόνου (R.T.O.S.).
- Να δώσει ένα παράδειγμα υλοποίησης με σύγχρονα υλικά. Έγινε προσπάθεια ώστε τα περισσότερα εξαρτήματα που χρησιμοποιήθηκαν να είναι επιφανειακής στήριξης.

Λέξεις-κλειδιά

Διάχυτη Ευφυία, Pervasive Computing, Domotics, Ενσωματωμένο Σύστημα, Ethernet, Web Control Box, RTOS, ΠΑΝΖΕΥΚΤΗΣ, PANZERMINAL , XBEE, ZigBee, Wi-Fi , Απομακρυσμένη Διαχείριση, Έξυπνο Σπίτι.

Abstract

The purpose of this thesis was to design, assemble and test an integrated hardware platform for the development of ambient intelligence and wireless domotics applications, providing enhanced processing capabilities. I gave this device the name “PANZEFKTIS”.

Alongside, a peripheral device which communicates wirelessly with “Panzeftkis” was developed, providing a multitude of i/o and measurements functionality, in order to allow for direct development of applications. I gave this device the name “Panzerterminal”.

The powerful 32-bit ARM Cortex-M3 50-MHz microprocessor “Panzeftkis” features, allows the effective development of modern, demanding applications. Also, “Panzeftkis” features extended connectivity as it is equipped with Ethernet, XBEE modem (one of 802.15.4, Wi-Fi, ZigBee or other wireless networks), RS232, IrDA, USB(COM). When combined with one or more “Panzerterminal” devices, a complete range of useful functions is provided.

The study for the design of the device hardware is presented, alongside with an extended reference to the firmware that has been developed for the presentation of the provided functionality, through the friendly user interface of an embedded web application. Access to all features is provided within the environment of a web browser.

This thesis targets to:

- provide a powerful embedded applications development platform, ideal for Ambient intelligence applications, around a very powerful ARM® 32bit microcontroller, providing a ready-to-use variety of connectivity options, storage media and measurement&control inputs and outputs.
- provide a solution for the remote connectivity of devices or individual sensors and actuators, through a friendly user interface, with the use of a web browser
- allow for easier development of specialized telemetry or other solutions, as the multi-layered architecture on which the main board is based allows for the easy installation of one or more additional special-purpose pcbs.
- allow for easy experimentation on time-critical applications, as implementation of Real Time Operating System (R.T.O.S.) is supported.
- provide an example of pcb development using modern electronic components. The effort was targeted on the use of surface mount technology.

Keywords

Ambient Intelligence, Pervasive Computing, Domotics, Embedded Systems, Ethernet, Web Control Box, RTOS, PANZEFKTIS, PANZERMINAL , XBEE, ZigBee, Wi-Fi ,Remote Administration, Smart Home.

ΠΕΡΙΕΧΟΜΕΝΑ

• Περίληψη	v
• Περιεχόμενα	vii
• Πρόλογος – Εισαγωγή	1
• Ενσωματωμένα συστήματα	3
○ Τι είναι ενσωματωμένο σύστημα;	3
○ Πού συναντά κανείς τα ενσωματωμένα συστήματα;	3
○ Γιατί επιλέγεται ένα ενσωματωμένο σύστημα στη θέση ενός Η/Υ	3
○ Ιστορική αναδρομή	4
• Η συσκευή «ΠΑΝΖΕΥΚΤΗΣ»	6
○ Γενική περιγραφή	6
○ Βασικά τεχνικά χαρακτηριστικά του «ΠΑΝΖΕΥΚΤΗ»	6
○ Το Υλικό	7
I. Το Stellaris® EK-LM3S6965 της Texas Instruments	7
II. Η κεντρική πλακέτα του «Πανζεύκτη»	8
A. Τα υποσυστήματα	
(Ανάλυση Λειτουργίας – Σχηματικά Διαγράμματα)	8
○ (A.1) Το κύκλωμα οδήγησης των ρελέ ελέγχου φορτίου	8
○ (A.2) Το κύκλωμα ανίχνευσης εισόδου 220V με οπτική απομόνωση	10
○ (A.3) Το δικτύωμα αναλογικής εισόδου με προστασία υπέρτασης	12
○ (A.4) Το υποσύστημα τροφοδοσίας	13
○ (A.5) Το υποσύστημα επικοινωνίας με το μόντεμ XBEE (ZigBee®, 802.15.4, Wi-Fi ή άλλα πρωτόκολλα)	15
○ (A.6) Το υποσύστημα επικοινωνίας με RS232 ή IrDA	16
○ (A.7) Η μνήμη I2C EEPROM	17
○ (A.8) Η πλακέτα του Evaluation Kit της T.I.	18
B. Διασύνδεση	23
○ (B.1) Η διασύνδεση των μονάδων ανίχνευσης εισόδων 220V με οπτική απομόνωση	23
○ (B.2) Η διασύνδεση των μονάδων ρελέ φορτίου	24
○ (B.3) Η διασύνδεση του υποσυστήματος με το XBEE	24
○ (B.4) Η διασύνδεση του τροφοδοτικού	25
○ (B.5) Η διασύνδεση των μονάδων για τη λήψη αναλογικών μετρήσεων	25
○ (B.6) Η διασύνδεση του Evaluation Board ή της πλακέτας του μικροελεγκτή.....	26
III. Το τυπωμένο κύκλωμα	27
IV. Όψεις της ολοκληρωμένης κατασκευής	28
V. Το “Panzerterminal”	33
○ Το σχηματικό διάγραμμα του “Panzerterminal”	35
○ Η πλακέτα του “Panzerterminal” (σε μεγέθυνση)	36
VI. Ο μικροελεγκτής Stellaris® LM3S6965 (ARM 32Bit)	37
○ Διαγράμματα	37
○ Περιγραφή	38
○ Χαρακτηριστικά – Δυνατότητες	38

○ Τεχνικά Χαρακτηριστικά	39
○ Ο Ethernet Controller του LM3S6965	41
VII. Η οικογένεια μόντεμ XBEE	43
○ Το Υλικολογισμικό (firmware)	44
▪ Η διασύνδεση Ethernet	44
• Η στοίβα TCP/IP lwIP	44
○ Χαρακτηριστικά	44
○ Υλοποίηση	45
○ Διεπαφή προγράμματος εφαρμογής (API)	46
○ Η Βιβλιοθήκη Οδηγών Περιφερειακών για Stellaris (Stellaris Peripheral Driver Library)	46
• Τα εργαλεία ανάπτυξης	47
• Η εφαρμογή χρήστη	47
○ Το δέντρο αρχείων του project	48
• Τα αρχεία web του http server	50
○ 1. index.htm	50
○ 2. io_http.htm	51
▪ Ημερομηνία / Ώρα	52
▪ Α. Ενδεικτικό Κατάστασης (Status) και επιλογέας (RS232-IrDA) (1 GPIO bit)	53
▪ Β. Αναλογικές Είσοδοι	55
▪ Γ. Είσοδοι Εποπτείας Φορτίου (4 οπτικά απομονωμένες είσοδοι 220V)	58
▪ Δ. Έξοδοι Ελέγχου Φορτίου (4 έξοδοι ρελέ 230V/3A)	61
▪ Ε. Σειριακή επικοινωνία (RS232 – IrDA)	64
▪ ΣΤ. Ασύρματος Έλεγχος Συσκευών (onboard XBEE module)	67
• α. Έλεγχος εξόδων γενικής χρήσης (GPIO) απομακρυσμένου modem	67
• β. Ασύρματη σειριακή επικοινωνία (XBEE 802.15.4 ή ZigBee)	70
▪ Ζ. Μαγνητικός Βομβητής ελεγχόμενος από παλμούς PWM	73
○ 3. about.htm	82
• Παράρτημα	
○ Διάσπαρτη Υπολογιστική (Ubiquitous Computing - ubicomp)	83
○ Διάχυτη Ευφυΐα (Ambient Intelligence)	84
○ Υλοποιήσεις: Διάχυτη Ευφυΐα σε Έξυπνο Σπίτι (Ambient Intelligence Domotics)	85
• Βιβλιογραφία – Αναφορές	87

Πρόλογος - Εισαγωγή

Παρακολουθώντας την εξέλιξη της τεχνολογίας των ηλεκτρονικών συσκευών κατά τις τελευταίες δεκαετίες, εύκολα διακρίνει κανείς την εκρηκτική αύξηση της συγκέντρωσης υπολογιστικής ισχύος των συσκευών που σήμερα μας περιβάλλουν και τις οποίες οι περισσότεροι χρησιμοποιούμε καθημερινά. Με την παράλληλη πρόοδο του λογισμικού που πλέον ενσωματώνεται σε όλες σχεδόν τις συσκευές, η χρήση πολύπλοκων εφαρμογών γίνεται απλή για οποιονδήποτε, μέσα από φιλικά περιβάλλοντα διεπαφής.

Τα περιβάλλοντα Διάχυτης Ευφυίας (Ambient Intelligence – Ami) αποτελούν σύνολα από συνεργαζόμενες συσκευές που συλλέγουν πληροφορίες από τις καθημερινές δραστηριότητες των χρηστών, τις αναλύουν και αντιδρούν κατάλληλα ώστε να διευκολύνουν τον χρήστη στην καθημερινή του ζωή. Παρά το γεγονός ότι η σχετική έρευνα έχει προχωρήσει, η ένταξη αυτών των συστημάτων στην καθημερινή πρακτική βρίσκεται ακόμη σε αρχικό στάδιο. Η υλοποίηση με τεχνολογίες προσιτού κόστους θα αυξήσουν την ταχύτητα διείσδυσης και τον ρυθμό ολοκλήρωσης αυτών των συστημάτων στη ζωή μας.

Ζητούμενο για την παρούσα διπλωματική εργασία ήταν να υλοποιηθεί:

« ένα Web on the Box Controller. Ουσιαστικά μια συσκευή, ένα κουτί «WebBox», το οποίο θα δέχεται μια διαδικτυακή σύνδεση τύπου Ethernet και θα διαθέτει μερικές εισόδους και εξόδους, δίνοντας έτσι τη δυνατότητα αλληλεπίδρασης του χρήστη με τον «πραγματικό» κόσμο, αρχικά μέσω μιας διαδικτυακής σελίδας. Οι εισοδοί θα έχουν την δυνατότητα να διαβάζουν αναλογικές και ψηφιακές καταστάσεις ενώ οι έξοδοι θα μπορούν να ελέγχουν μερικά relay. Σαν παράδειγμα μπορούμε να αναφέρουμε την εφαρμογή ενός WCB στη δημιουργία ενός έξυπνου σπιτιού. Ο χρήστης θα συνδέεται στο διαδίκτυο είτε μέσω Η/Υ είτε μέσω του κινητού τηλεφώνου του και με την βοήθεια μιας ιστοσελίδας ή του email του θα έχει πρόσβαση στο WebBox. Αποκτώντας πρόσβαση θα ενημερώνεται για την κατάσταση διαφόρων παραμέτρων του σπιτιού του όπως θερμοκρασία, υγρασία, ύπαρξη ατόμων, παραβίαση χώρων, ενώ ταυτόχρονα θα μπορεί να ελέγχει κάποιες συσκευές, όπως ρύθμιση θερμοστάτη, άνοιγμα/κλείσιμο ποτίσματος, φώτων κλπ.»

Ως προέκταση του ζητούμενου, σκοπός της παρούσας διπλωματικής εργασίας ήταν η σχεδίαση, κατασκευή και δοκιμή μιας ολοκληρωμένης πλατφόρμας υλικού για την ανάπτυξη εφαρμογών διάχυτης ευφυίας, ασύρματης εποπτείας και ευφυούς ελέγχου οικίας (domotics), με προηγμένες υπολογιστικές δυνατότητες. Ονόμασα αυτή τη συσκευή «ΠΑΝΖΕΥΚΤΗ».

Παράλληλα, ανέπτυξα και περιφερειακή συσκευή η οποία επικοινωνεί ασύρματα με τον «Πανζεύκτη» και προσφέρει πολλαπλή λειτουργικότητα εισόδου-εξόδου και μετρήσεων, ώστε να διευκολύνεται η άμεση υλοποίηση εφαρμογών. Ονόμασα αυτή τη συσκευή «Panzerterminal».

Ο ισχυρότατος μικροελεγκτής 32-bit ARM Cortex-M3 50-MHz που ο «Πανζεύκτης» διαθέτει, επιτρέπει την αποτελεσματική ανάπτυξη σύγχρονων εφαρμογών με υψηλές απαιτήσεις. Ακόμη, ο «Πανζεύκτης» προσφέρει εκτεταμένη συνδεσιμότητα καθώς διαθέτει Ethernet, μόντεμ XBEE (επιλογή ανάμεσα σε ZigBee, 802.15.4, Wi-Fi ή και άλλα ασύρματα δίκτυα), RS232, IrDA, USB(COM). Έτσι, σε συνδυασμό με μία ή περισσότερες τερματικές μονάδες «Panzerterminal» προσφέρεται ένα πλήρες φάσμα από χρήσιμες λειτουργίες.

Παρουσιάζεται η πλήρης θεωρητική μελέτη για την κατασκευή, καθώς και το υλικολογισμικό που έχει αναπτυχθεί για την παρουσίαση των λειτουργιών, μέσα από το φιλικό περιβάλλον μιας ενσωματωμένης web εφαρμογής. Η πρόσβαση σε όλες τις λειτουργίες της συσκευής γίνεται μέσα από τον φυλλομετρητή ιστού του χρήστη.

Η εργασία αυτή φιλοδοξεί:

- Να αποτελέσει μια ισχυρή πλατφόρμα ανάπτυξης ενσωματωμένων εφαρμογών, ιδανική για εφαρμογές Διάχυτης Ευφυΐας, έχοντας ως βάση έναν ισχυρότατο ARM® 32bit μικροελεγκτή και παρέχοντας έτοιμη προς χρήση μια ποικιλία διασυνδέσεων, μέσω αποθήκευσης και εισόδων-εξόδων μέτρησης και ελέγχου.
- Να προσφέρει μια λύση για τη διασύνδεση συσκευών ή μεμονωμένων αισθητήρων και ενεργοποιητών με τον χρήστη, οπουδήποτε και αν αυτός βρίσκεται, μέσα από ένα φιλικό περιβάλλον, με τη χρήση ενός απλού περιηγητή ιστού.
- Να διευκολύνει σημαντικά την υλοποίηση εξειδικευμένων συσκευών τηλεμετρίας ή άλλων, καθώς η πολυεπίπεδη αρχιτεκτονική στην υλοποίηση της κεντρικής πλακέτας επιτρέπει την απρόσκοπτη προσθήκη επιπλέον πλακετών ειδικής χρήσης.
- Να δώσει τη δυνατότητα πειραματισμού στις time-critical εφαρμογές, καθώς υποστηρίζεται η υλοποίηση με λειτουργικό σύστημα πραγματικού χρόνου (R.T.O.S.).
- Να δώσει ένα παράδειγμα υλοποίησης με σύγχρονα υλικά. Έγινε προσπάθεια ώστε τα περισσότερα εξαρτήματα που χρησιμοποιήθηκαν να είναι επιφανειακής στήριξης.

Θα ήθελα να εκφράσω τις ευχαριστίες μου προς τον καθηγητή μου κ. Συκά και τον υποψήφιο διδάκτορα κο Απόστολο Κοτοπούλη που μου εμπιστεύτηκαν αυτή την εργασία καθώς και για την υποστήριξη και τις πολύτιμες συμβουλές που τόσο γενναιόδωρα μου παρείχαν. Τέλος, θα ήθελα να ευχαριστήσω τη σύζυγό μου Γεωργία και να αφιερώσω την προσπάθεια αυτή στα παιδιά μας Χάρη και Αφροδίτη.

Ενσωματωμένα συστήματα

Τι είναι ενσωματωμένο σύστημα;

Τα ενσωματωμένα συστήματα είναι αυτόνομες συσκευές ή υπολογιστικές μονάδες που έχουν συνήθως σχεδιαστεί ώστε να εκτελούν περιορισμένες επεξεργαστικές λειτουργίες αξιόπιστα, με ασφάλεια και με ελάχιστο κόστος συντήρησης.

Πού συναντά κανείς τα ενσωματωμένα συστήματα;

Ενσωματωμένα συστήματα ή ενσωματωμένα συστήματα υπολογιστών συναντούμε πλέον παντού γύρω μας.

Μερικά είναι χαμηλού κόστους και μαζικής παραγωγής, όπως κινητά τηλέφωνα, DVD player, ηλεκτρονικοί ζυγοί.

Κάποια είναι υψηλού κόστους και χρησιμοποιούνται σε εξαιρετικά εξειδικευμένα συστήματα: σε ιατρικά όργανα, μηχανήματα υπερήχων, δορυφόρους αεροδιαστημικών επικοινωνιών, αυτοματισμούς εργοστασίων, βιομηχανικούς λέβητες, κεντρικές μονάδες ελέγχου των αυτοκινήτων αλλά στα συστήματα ελέγχου των αεροσκαφών.

Κάποια άλλα χρησιμοποιούνται στις συσκευές καθημερινής χρήσης που θέλουμε να λειτουργούν αξιόπιστα, όπως οι φωτεινοί σηματοδότες, τα συστήματα συλλογής των ναύλων, οι σαρωτές ασφαλείας των αερολιμένων και οι συσκευές καταγραφής δεδομένων των αεροσκαφών.

Επιπλέον, κάποια είναι εξαιρετικά αυστηρών προδιαγραφών, γρήγορης απόκρισης συστήματα σε κρίσιμης σημασίας μονάδες ελέγχου όπως τα ραντάρ ελέγχου εναέριας κυκλοφορίας, τα συστήματα ασφαλείας για τρένα και τα ιατρικά μηχανήματα υποστήριξης της ζωής.

Γιατί επιλέγεται ένα ενσωματωμένο σύστημα στη θέση ενός Η/Υ

- Αξιοπιστία (εργασία χωρίς αποτυχία για χρόνια)
- Ασφάλεια (ατρωσία από Hacker)
- Περιβάλλον (μεγάλο εύρος θερμοκρασιών, σφραγισμένα από τις χημικές ουσίες, ακτινοβολία)
- Ενεργειακή Απόδοση (αντλεί ελάχιστη ενέργεια για τη λειτουργία που επιτελεί)
- Ενσωμάτωση στη συσκευή για την οποία προορίζεται
- Προσαρμοσμένες απαιτήσεις τάσης / ρεύματος (DC: 12, 14, 24, 72.)
- Αποτελεσματική αλληλεπίδραση με τον χρήστη (λιγότερα κουμπιά, διακόπτες αφής)[1]

Ιστορική αναδρομή

Ένα από τα συστήματα που καταγράφησαν στις πρώτες σελίδες της ιστορίας των σύγχρονων ενσωματωμένων συστημάτων ήταν ο Υπολογιστής Προσανατολισμού του διαστημικού σκάφους Apollo (Apollo Guidance Computer), που αναπτύχθηκε από τον Charles Stark Draper στο Εργαστήριο Οργανολογίας του MIT. Κατά τη σύλληψη του έργου, ο υπολογιστής καθοδήγησης του Apollo θεωρήθηκε το πιο ριψοκίνδυνο στοιχείο στο Apollo Project, καθώς είχε υλοποιηθεί με τα, νέα εκείνη την εποχή, μονολιθικά ολοκληρωμένα κυκλώματα, για τη μείωση του βάρους και του μεγέθους του υπολογιστή.

Ένα από τα πρώτα μαζικής παραγωγής ενσωματωμένο σύστημα ήταν ο υπολογιστής καθοδήγησης Autonetics D-17 του πυραύλου Minuteman, που εμφανίστηκε το 1961. Είχε κατασκευαστεί από τρανζίστορ και είχε έναν σκληρό δίσκο ως κύρια μνήμη. Όταν ο Minuteman II βγήκε στην παραγωγή το 1966, ο D-17 έχει αντικατασταθεί με έναν νέο υπολογιστή που σηματοδότησε και την πρώτη μαζική χρήση των ολοκληρωμένων κυκλωμάτων. Το πρόγραμμα αυτό από μόνο του μείωσε τις τιμές των ολοκληρωμένων κυκλωμάτων τεσσάρων NAND πυλών (quad NAND IC) από 1000\$ σε 3\$ το κάθε ένα, καθιστώντας πλέον εφικτή τη χρήση τους σε εμπορικά προϊόντα.

Από την εμφάνιση αυτών των πρώτων εφαρμογών στη δεκαετία του 1960, έχει σημειωθεί μεγάλη μείωση του κόστους των ενσωματωμένων συστημάτων, ενώ παράλληλα υπάρχει και μια δραματική αύξηση της επεξεργαστικής ισχύος και της λειτουργικότητας τους. Ο πρώτος μικροεπεξεργαστής για παράδειγμα, ο Intel 4004, είχε σχεδιαστεί για υπολογιστές και άλλα μικρά συστήματα, αλλά απαιτούσε τη χρήση πολλών ολοκληρωμένων κυκλωμάτων εξωτερικής μνήμης και υποστήριξης.

Το 1978, η National Engineering Manufacturers Association κυκλοφόρησε ένα "πρότυπο" για προγραμματιζόμενους μικροελεγκτές, όπου συμπεριελάμβανε σχεδόν όλους τους ελεγκτές που βασίζονται σε υπολογιστή, όπως υπολογιστές μιας πλακέτας (single board computers), αριθμητικούς επεξεργαστές και ελεγκτές οδηγούμενους από γεγονότα.

Καθώς το κόστος των μικροεπεξεργαστών και μικροελεγκτών έπεσε, έγινε εφικτό να αντικατασταθούν τα δαπανηρά εξαρτήματα που αποτελούσαν τα «κουμπιά» των συσκευών, όπως τα ποτενσιόμετρα και οι μεταβλητοί πυκνωτές με πλήκτρα «πανω-κάτω» ή περιστρεφόμενους διακόπτες που ο μικροελεγκτής τους «διάβαζε» απευθείας, ακόμη και σε ορισμένα ορισμένα καταναλωτικά προϊόντα. Μέχρι τα μέσα της δεκαετίας του 1980, τα περισσότερα από τα κοινά εξαρτήματα του συστήματος που προηγουμένως ήταν εξωτερικά, είχαν ενσωματωθεί στο ίδιο τσιπ, καθώς ο επεξεργαστής και αυτή η σύγχρονη μορφή του μικροελεγκτή επέτρεψε μια ακόμα πιο εκτεταμένη χρήση, η οποία μέχρι το τέλος της δεκαετίας ήταν ο κανόνας και όχι την εξαίρεση για όλες σχεδόν τις ηλεκτρονικές συσκευές.

Η εξέλιξη των μικροελεγκτών έχει αυξήσει περαιτέρω τις εφαρμογές για τις οποίες τα ενσωματωμένα συστήματα χρησιμοποιούνται σε περιοχές όπου παραδοσιακά ένας υπολογιστής δεν θα είχε επιλεγεί. Ένας γενικού σκοπού και συγκριτικά χαμηλού κόστους μικροελεγκτής μπορεί συχνά να προγραμματιστεί για να εκπληρώσει τον ίδιο ρόλο με έναν μεγάλο αριθμό από επιμέρους εξαρτήματα. Παρά το γεγονός ότι στο πλαίσιο αυτό, ένα ενσωματωμένο σύστημα είναι συνήθως πιο πολύπλοκο από μια παραδοσιακή λύση, το μεγαλύτερο μέρος της πολυπλοκότητας περιέχεται μέσα στο ίδιο τον μικροελεγκτή. Πολύ λίγα επιπλέον στοιχεία μπορεί να απαιτηθούν και το μεγαλύτερο μέρος της προσπάθειας

του σχεδιασμού είναι στο λογισμικό. Ο άυλος χαρακτήρας του λογισμικού καθιστά πολύ πιο εύκολη την κατασκευή προτύπων και δοκιμή νέων εκδόσεων σε σύγκριση με τον σχεδιασμό και την κατασκευή ενός νέου κυκλώματος το οποίο δε χρησιμοποιεί ενσωματωμένο επεξεργαστή.[2]

Η συσκευή «ΠΑΝΖΕΥΚΤΗΣ»

Γενική περιγραφή

Πρόκειται για μια συσκευή που δίνει τη δυνατότητα στον χρήστη να ελέγχει μέσω του Ethernet (Internet, intranet, PC) ηλεκτρικές συσκευές, ασύρματους αισθητήρες και ενεργοποιητές καθώς και να επικοινωνεί μέσω USB και σειριακής θύρας με έναν ΗΥ ή άλλο ενσωματωμένο σύστημα. Διαθέτει το XBEE της Digi International, ένα μικρού μεγέθους, πλούσιο σε χαρακτηριστικά και οικονομικό μόντεμ για ασύρματη δικτύωση.

Βασικά τεχνικά χαρακτηριστικά του «ΠΑΝΖΕΥΚΤΗΣ»

ΜΙΚΡΟΕΛΕΓΚΤΗΣ

Texas Instruments® Stellaris® LM3S6965 (ARM 32Bit)

DEBUG INTERFACE

Ενσωματωμένο σύστημα αποσφαλμάτωσης μέσω USB (JTAG/SWD)

ΓΕΝΙΚΑ

- Οθόνη OLED (128x96px)
- Πλήκτρα πλοήγησης
- Μαγνητικός Βομβητής

ΔΙΑΣΥΝΔΕΣΗ ΕΠΙΚΟΙΝΩΝΙΩΝ

- Ethernet 10/100
- Virtual COM Port (USB-COM PORT Bridge)
- Σειρική Θύρα με Baud Rate έως 115,2kbps
- Πομποδέκτης IrDA
- ZigBee® ή 802.15.4 (Μόντεμ XBEE®)

ΕΙΣΟΔΟΙ ΕΛΕΓΧΟΥ

4 είσοδοι για 220VAC με οπτική απομόνωση

ΕΞΟΔΟΙ ΕΛΕΓΧΟΥ

4 ρελέ δύο θέσεων για φορτίο τάσης 250VAC/220VDC, 3A AC/DC

ΕΙΣΟΔΟΙ ΜΕΤΡΗΣΗΣ

4 προσατευμένες είσοδοι 0%3,3Volt

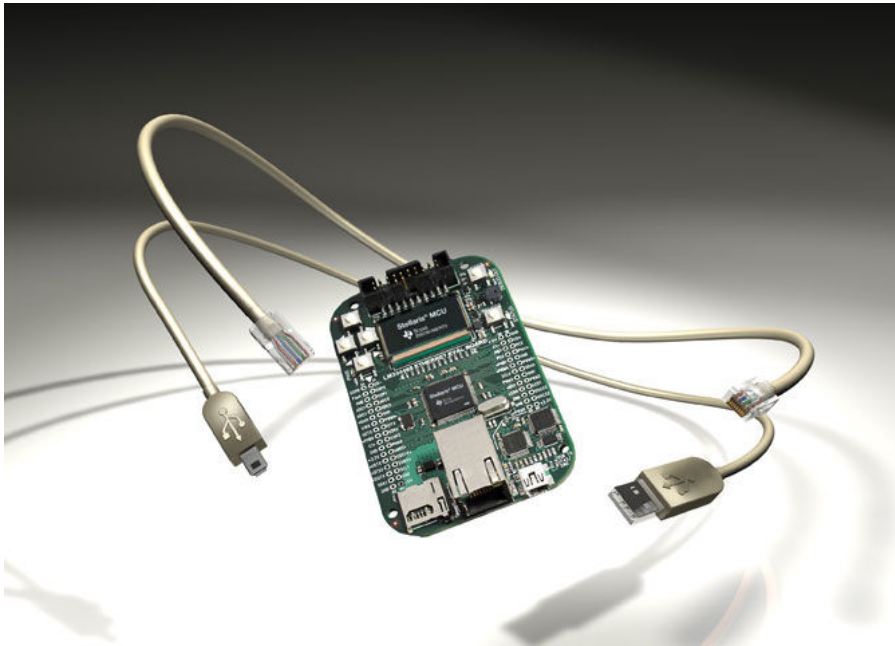
ΣΥΣΚΕΥΕΣ ΑΠΟΘΗΚΕΥΣΗΣ

- Κάρτα microSD(uSD) (σε SPI-mode)
- Μνήμη I2C χωρητικότητας 1MBit

Το Υλικό

Ο «Πανζεύκτης» αποτελείται από δύο πλακέτες και το μόντεμ XBEE:

I. Το Stellaris® EK-LM3S6965 της Texas Instruments



Πηγή: http://focus.ti.com/graphics/tool/EK_LM3S6965_8_09.jpg

Πρόκειται για ένα Evaluation Kit γύρω από τον 32Bit μικροελεγκτή Stellaris® LM3S6965 της Texas Instruments (ARM Cortex M3).

Το Evaluation Board διαθέτει:

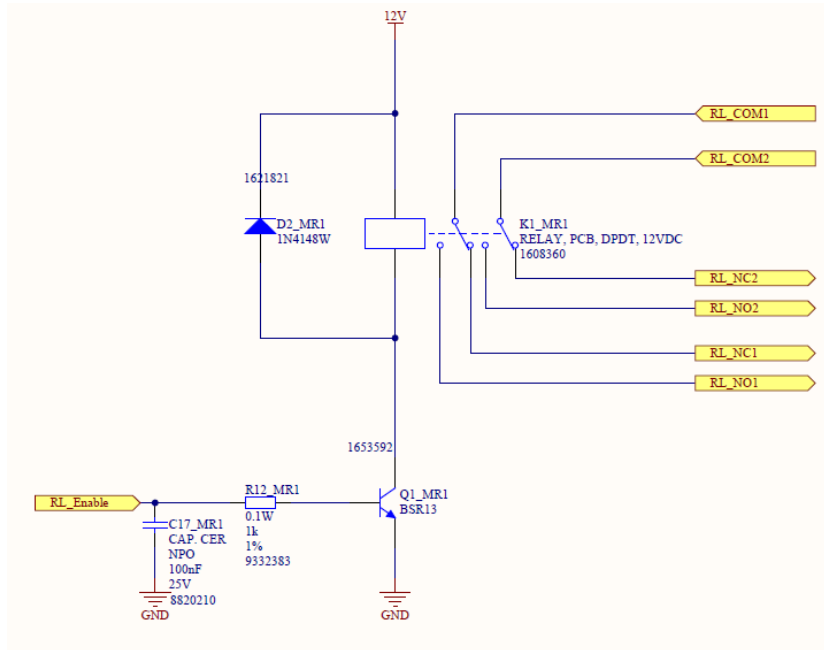
- Ενσωματωμένο σύστημα αποσφαλμάτωσης μέσω USB (JTAG/SWD)
- Ενσωματωμένο Ethernet 10/100 (MAC και PHY)
- Οθόνη OLED (128x96px)
- Πλήκτρα πλοήγησης
- Μαγνητικός Βομβητής
- Virtual COM Port (USB-COM PORT Bridge)
- Κάρτα microSD(υSD) (σε SPI-mode)
- Τροφοδοτικά 15V(boost), 5V και 3,3V.

Επίσης, προσφέρει δύο διπλές υποδοχές για ακιδοσειρές, που επιτρέπουν την εύκολη σύνδεση εξωτερικών κυκλωμάτων με ακροδέκτες τροφοδοσίας και περιφερειακών του μικροελεγκτή.[3]

II. Η κεντρική πλακέτα του «Πανζεύκτη»

A. Τα υποσυστήματα (Ανάλυση Λειτουργίας - Σχηματικά Διαγράμματα)

(A.1) Το κύκλωμα οδήγησης των ρελέ ελέγχου φορτίου



Σχ.Α.1) Το κύκλωμα οδήγησης για ένα ρελέ ελέγχου φορτίου

Η συσκευή διαθέτει τέσσερα ρελέ διπλού κυκλώματος, δύο επαφών (Dual Position Dual Terminal – DPDT), με ικανότητα ελέγχου ρεύματος 3A AC/DC, τάσης 250V AC/220VDC. Για το κάθε ένα από αυτά, υλοποίησα ένα κύκλωμα ελέγχου όπως στο παραπάνω σχήμα.

Το πηνίο του ρελέ έχει ονομαστική τάση ελέγχου 12VDC με ονομαστική ωμική αντίσταση 960Ω.

Υπολογισμός κυκλώματος:

Από το φύλλο δεδομένων του κατασκευαστή (Σύνδεσμος 1), η τάση τροφοδοσίας είναι 12VDC ενώ η ονομαστική ωμική αντίσταση του πηνίου είναι 960Ω. Το μέγιστο ρεύμα διέρχεται από το πηνίο μετά τη διέλευση των μεταβατικών φαινομένων και καθώς ο κατασκευαστής δίνει $V_{CE(SAT)max} \approx 0,4Volt$, είναι:

$$I_{C(SAT)} = \frac{V_{cc} - V_{CE(SAT)max}}{R_{coil}} = \frac{(12 - 0,4)Volt}{960\Omega} = 12mA$$

Το τρανζίστορ που οδηγεί το ρελέ έχει (Σύνδεσμος 2) μέγιστο ρεύμα συνεχούς λειτουργίας 500mA, δηλαδή 40 φορές περισσότερο από αυτό που ελέγχει εδώ. Επομένως, είναι κατάλληλο για την εφαρμογή αυτή. Φυσικά, το πλήθος των επιλογών τρανζίστορ για αυτήν την απλή διακοπτική λειτουργία είναι πραγματικά ανεξάντλητο.

Η διόδος D2 είναι απαραίτητη καθώς επιτρέπει τη διέλευση του αναστροφικού ρεύματος του πηνίου όταν αυτό απενεργοποιείται. Πρόκειται για μια διόδο (1N4148) γενικής χρήσης.

Για τον προσδιορισμό της μέγιστης τιμής του ρεύματος αυτού, καθώς οριακά πριν την απενεργοποίηση του πηνίου το ρεύμα του είναι $I_{C(SAT)}$ και επειδή κατά την απόδοση της αποθηκευμένης ενέργειας με τον σταδιακό εκμηδενισμό του μαγνητικού πεδίου στο πηνίο το ανάστροφο ρεύμα διέρχεται μέσα από την ίδια την εσωτερική αντίσταση του πηνίου, το μέγιστο αντιστροφο ρεύμα θα εμφανίζεται ακριβώς τη στιγμή της διακοπής και θα είναι ίσο με $-I_{C(SAT)} = -12mA$. Καθώς η διόδος είναι κατασκευασμένη για ρεύμα συνεχούς εφαρμογής 150mA (παλμικό 300mA) η διόδος ανταπεξέρχεται χωρίς καμία δυσκολία. Σημειώνεται επίσης ότι κατά την αγωγή του πηνίου εφαρμόζεται στη διόδο αναστροφη τάση 11,6Volt, πολύ μικρότερη από την τάση διάσπασης της διόδου κατά την αναστροφη πόλωση η οποία είναι τουλάχιστον 75Volt.

Ο πυκνωτής στην πλευρά της βάσης του τρανζίστορ συνδέεται για να κάνει την απαραίτητη απόξεση των παρασιτικών σημάτων που εμφανίζονται επάνω στη γραμμή που συνδέει την έξοδο του μικροελεγκτή με το δικτύωμα ελέγχου του ρελέ.

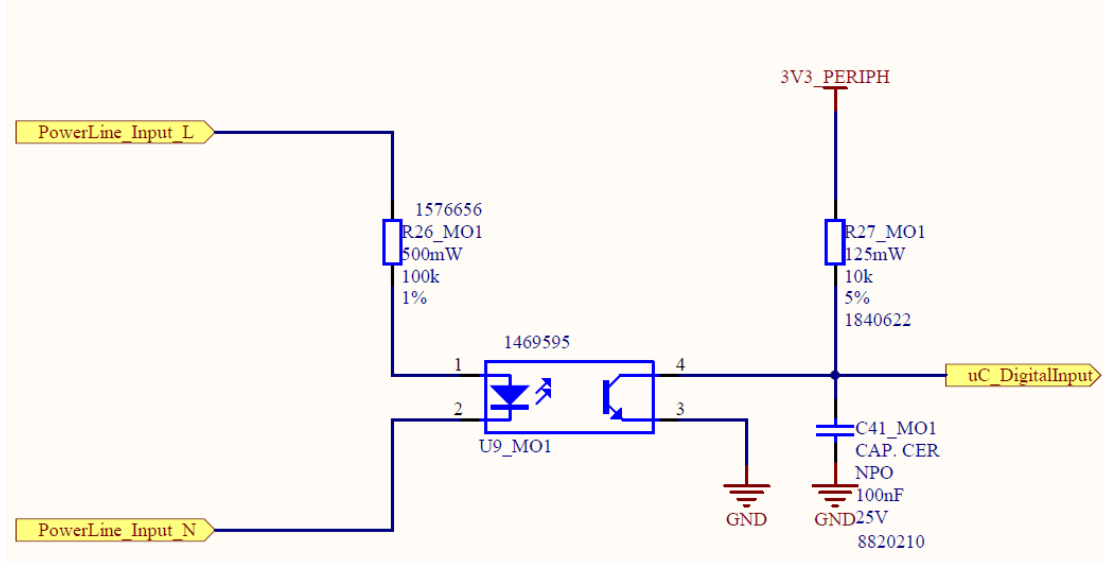
Όταν το σήμα ελέγχου από τον μικροελεγκτή είναι low, τότε το τρανζίστορ βρίσκεται σε αποκοπή και το ρελέ είναι απενεργοποιημένο. Όταν το σήμα ελέγχου από τον μικροελεγκτή είναι high, τότε το τρανζίστορ βρίσκεται σε κορεσμό και το ρελέ είναι απενεργοποιημένο.

Αναλυτικά, για την περίπτωση αυτή:

Όταν το τρανζίστορ είναι στην περιοχή κόρου, τότε το ρεύμα συλλέκτη είδαμε ότι είναι: $I_{C(SAT)} = 12mA$. Το ελάχιστο h_{fe} για αυτήν την περιοχή του I_C δίνεται ίσο με 75. Έτσι, στην χειρότερη περίπτωση, όπου $h_{fe}=75$, το απαιτούμενο ρεύμα βάσης είναι: $I_B = \frac{I_{C(SAT)}}{h_{fe_min}} = \frac{12mA}{75} = 160\mu A$. Συνεπώς, η ελάχιστη τιμή της τάσης ελέγχου με την οποία θα ενεργοποιηθεί το ρελέ είναι: $V_{in(min)} = V_{BE} + I_B R_{12} = 0,7V + 160\mu A * 1k\Omega = 0,86Volt$.

Συνεπώς, όταν το σήμα ελέγχου είναι high (3.3Volt nom.> 0,86Volt), τότε το τρανζίστορ εισέρχεται στον κόρο και το ρελέ ενεργοποιείται.

(A.2) Το κύκλωμα ανίχνευσης εισόδου 220V με οπτική απομόνωση



Σχ.Α.2) Το κύκλωμα ανίχνευσης εισόδου 220V με οπτική απομόνωση

Πρόκειται για ένα δικτύωμα γύρω από τον οπτοζεύκτη SFH690BT της Vishay Semiconductor, με το οποίο μπορεί ο μικροελεγκτής να παρακολουθεί την τάση που επικρατεί σε μια συσκευή ή εγκατάσταση 220V, με γαλβανική απομόνωση.

Κατά τη θετική ημιπερίοδο και όσο η τάση επί της φωτοδιόδου εντός του οπτοζεύκτη είναι μεγαλύτερη από την τάση ορθής πόλωσης αυτής V_F (Σύνδεσμος 2), η φωτοδίοδος είναι ενεργοποιημένη.

Από το φύλλο δεδομένων του κατασκευαστή (Σύνδεσμος 3), εξάγουμε τα ακόλουθα στοιχεία:

$$V_F = 1.15\text{Volt}, CTR_{MIN} = 100\%, CTR_{MAX} = 300\%$$

$$V_{IL_{max}} = 1.3\text{Volt}, V_{IH_{min}} = 2\text{Volt}$$

Η έξοδος του οπτοζεύκτη αναγνωρίζεται από τον μικροελεγκτή ως λογικό χαμηλό (low) όταν η τάση είναι μικρότερη ή ίση από $V_{IL_{max}} = 1.3\text{Volt}$. Για να συμβαίνει αυτό, πρέπει το ρεύμα που διέρχεται από την αντίσταση R_{27} και το φωτοτρανζίστορ να είναι ίσο με

$$I_C = \frac{V_{3V3PERIPH} - V_{IL_{max}}}{R_{27}} = \frac{3,3\text{Volt} - 1,3\text{Volt}}{10\text{k}\Omega} = 200\mu\text{A}$$

Επομένως, στη χειρότερη περίπτωση, όπου το Current Transfer Ratio είναι μικρό, πρέπει το ρεύμα που διέρχεται από τη φωτοδίοδο να είναι:

$$I_F = \frac{I_C}{CTR_{MIN}} = \frac{200\mu\text{A}}{100\%} = 200\mu\text{A}$$

Συνεπώς, αρκεί η τάση εισόδου (για τη χειρότερη περίπτωση όπου $V_F = V_{F_{max}}$, να είναι:

$$V_{in} = V_{F_{max}} + I_F R_{26} = 1.4\text{Volt} + 200\mu\text{A} * 100\text{k}\Omega = 28\text{Volt}$$

Όταν η είσοδος είναι εναλλασσόμενη 220V RMS ημιτονοειδής με συχνότητα 50Hz, τότε το αναμενόμενο duty-cycle προκύπτει ως εξής:

Ο οπτοζεύκτης άγει μόνο στο τμήμα της θετικής ημιπερίοδου όπου η τάση εισόδου είναι μεγαλύτερη των 18,2Volt, έστω στο χρονικό διάστημα $[t_{th_1}, t_{th_2}]$, όπου

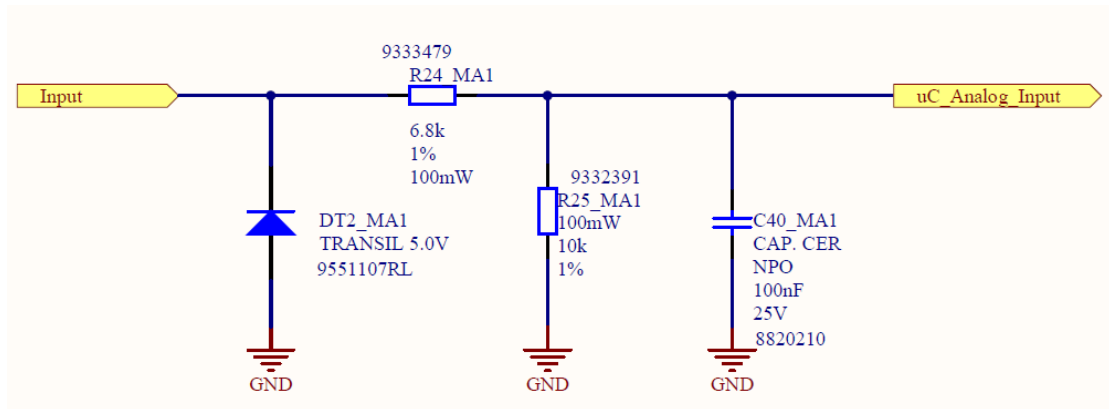
$$V_{th} = 220\sqrt{2} \sin(100\pi t_{th}) = 28\text{Volt} \Rightarrow t_{th_1} = \frac{\arcsin\left(\frac{28}{220\sqrt{2}}\right)}{100\pi} = 0.29\text{ms}$$

$$t_{th_2} = \frac{T}{2} - t_{th_1} = \frac{1}{2f} - t_{th_1} = \frac{1}{100} - 0.29\text{ms} = 9,71\text{ms}$$

Κατά το χρονικό διάστημα αυτό της πλήρους περιόδου, το σήμα εξόδου είναι σε στάθμη λογικά χαμηλού. Έτσι, το duty-cycle είναι : $D.C. = 1 - \frac{t_{th_2} - t_{th_1}}{T} = 1 - f(t_{th_2} - t_{th_1}) = 1 - 50\text{Hz}(9,71\text{ms} - 0.29\text{ms}) = 52.9\%$.

Συνεπώς, ο μικροελεγκτής μπορεί να αποφασίζει ότι η εναλασσόμενη τάση εισόδου είναι ενεργοποιημένη, όταν σε δειγματοληψία διάρκειας τουλάχιστον $\frac{1}{f} = \frac{1}{50\text{Hz}} = 20\text{ms}$, το πλήθος των δειγμάτων που ανιχνεύονται σε υψηλή στάθμη είναι τουλάχιστον στο 40% των συνολικών. Εισάγουμε δηλαδή μια μικρή ανοχή, ώστε να ανιχνεύονται και χαμηλότερες τάσεις. Επίσης, αξίζει να σημειώσουμε ότι όταν η τάση στον ακροδέκτη εισόδου του μικροελεγκτή είναι στην «γκρίζα» περιοχή 1,3 έως 2 volt, η λογική στάθμη που ανιχνεύεται δεν μπορεί να προβλεφθεί. Θεωρούμε ότι η στατιστική κατανομή των δειγμάτων από αυτήν την περιοχή είναι ισοβαρής για το αρμονικό σήμα της εισόδου.

(A.3) Το δικτύωμα αναλογικής εισόδου με προστασία υπέρτασης



Σχ.Α.3) Το δικτύωμα αναλογικής εισόδου

Υλοποιείται ένας διαιρέτης τάσης με τη χρήση των R_{24}, R_{25} .

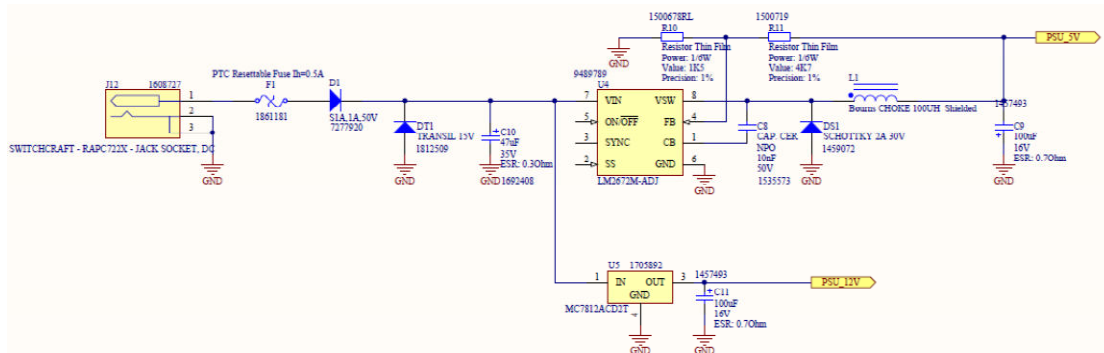
$$V_o = \frac{R_{25}}{R_{24} + R_{25}} V_i = \frac{10k\Omega}{6,8k\Omega + 10k\Omega} V_i = 0,59V_i$$

Όταν η τάση εισόδου είναι 5Volt, η τάση εξόδου είναι $V_o = 0,59 * 5Volt = 2,95Volt$ που είναι μικρότερη από το όριο των 3,3Volt που απαιτεί ο μικροελεγκτής για τις αναλογικές εισόδους του.

Μια διάδος προστασίας transil, προστατεύει τον μικροελεγκτή αν εφαρμοστούν προς μέτρηση τάσεις μεγαλύτερες των 5Volt.

Ο πυκνωτής εξομάλυνσης C_{40} χρησιμεύει στο φιλτράρισμα του σήματος προς μέτρηση.

(A.4) Το υποσύστημα τροφοδοσίας



Σχ.Α.4.1) Τα υποσυστήματα τροφοδοσίας 12V,5V

Γενικά: Το υποσύστημα τροφοδοσίας δέχεται εξωτερική παροχή 15VDC/0,5A. Εισάγεται μια ασφάλεια αυτοεπαναφοράς (F1-Auto Resettable Fuse) η οποία σε κανονικές συνθήκες έχει πολύ χαμηλή αντίσταση. Όταν όμως το ρεύμα ξεπεράσει το 0,5A, αυξάνει η θερμοκρασία της και αποκτά πολύ υψηλή αντίσταση. Έτσι, μέχρι την πιθανή άρση της ανωμαλίας που προκάλεσε την αρχική υπερένταση, το κύκλωμα διαρρέεται από χαμηλό ρεύμα και η συσκευή προστατεύεται από τις μόνιμες βλάβες και τον κίνδυνο πυρκαϊάς που εισάγει η υπερθέρμανση.

Εισάγεται μια δίοδος (D1) προστασίας από αναστροφή της εξωτερικά παρεχόμενης τροφοδοσίας με μέγιστη ικανότητα ρεύματος κατά την ορθή πόλωση 1A. Επίσης, μια δίοδος TRANSIL στα 15V προστατεύει από πιθανές αιχμές υπέρτασης στην τάση τροφοδοσίας, καθώς όταν η τάση ξεπεράσει τα {16.7 έως 18.5}Volt, αρχίζει να άγει, με αποτέλεσμα να διατηρείται η τάση κοντά στα 15V. Επίσης, αν από λάθος συνδεθεί στην είσοδο τροφοδοσία μεγαλύτερη από τα 15V που είναι η μέγιστη επιτρεπόμενη τάση της δίοδου (TRANSIL standoff voltage), η δίοδος αυτή άγει ισχυρά και επιτρέπει τη διέλευση υψηλού ρεύματος το οποίο προκαλεί την υπερθέρμανση της ασφάλειας F1 και συνεπώς τη διακοπή παροχής στη συσκευή, προστατεύοντας την.

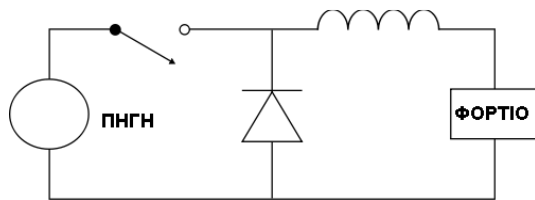
Ένας πυκνωτής τανταλίου (C10) με χαμηλή ισοδύναμη αντίσταση σειράς (ESR) φιλτράρει την τάση τροφοδοσίας.

Τροφοδοσία 12VDC (PSU_12V):

Μέσω μόνο ενός γραμμικού ολοκληρωμένου ρυθμιστή τάσης (MC7812ACD2T) και πυκνωτή εξόδου που προσφέρει καλύτερη μεταβατική απόκριση στον ρυθμιστή, κυρίως όταν η τάση εισόδου μεταβάλλεται ταχέως, παρέχεται η τάση των 12VDC που χρησιμοποιείται για την τροφοδοσία των ρελέ και είναι ανεξάρτητη από τις άλλες τροφοδοσίες.

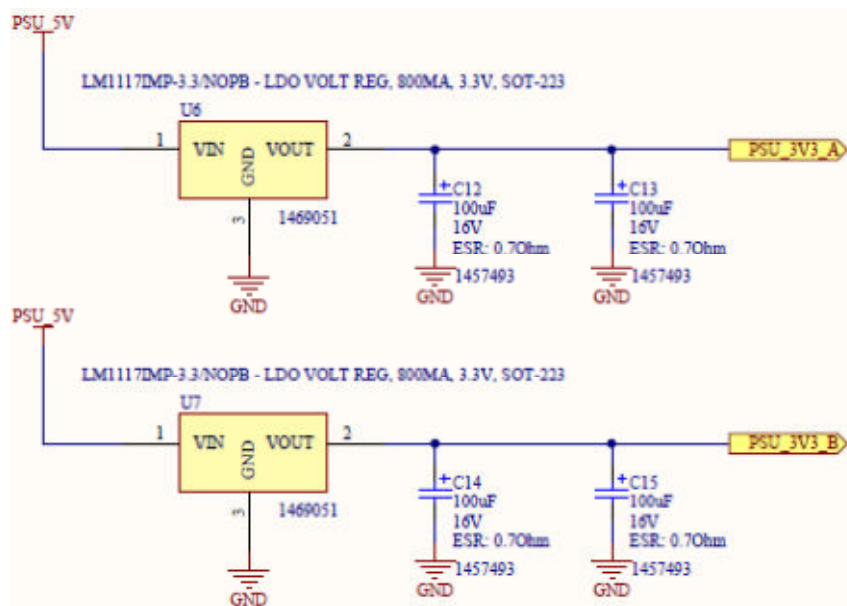
Τροφοδοσία 5VDC (PSU_5V):

Υλοποιείται ένα διακοπτικό τροφοδοτικό (Switching Mode Power Supply – SMPS) γύρω από το ολοκληρωμένο LM2672M-ADJ, το οποίο με τη βοήθεια ελαχίστων εξωτερικών εξαρτημάτων, συνδεσμολογεί ένα διακοπτικό τροφοδοτικό υποβιβασμού τάσης (buck converter).



Η αρχή λειτουργίας των μετατροπέων buck είναι αρκετά απλή. Με τη βοήθεια ενός πηνίου και δύο διακοπών (που εδώ υλοποιούνται με το τρανζίστορ εξόδου του LM και τη δίοδο schottky) οι οποίοι ελέγχουν το πηνίο. Γίνεται συνεχώς σύνδεση και επαναφορά του διακόπτη, με αποτέλεσμα σε πρώτη φάση να αποθηκεύει το πηνίο ενέργεια από την πηγή και στη συνέχεια το πηνίο να αποδίδει αυτή την ενέργεια στο φορτίο. Η τάση εξόδου καθορίζεται από το duty-cycle της αγωγής του τρανζίστορ. Στην παρούσα υλοποίηση, ένα δικτύωμα αντιστάσεων (R10,R11) ανατροφοδοτεί κατάλληλα δείγμα της τάσης εξόδου στο κύκλωμα ελέγχου, το οποίο καθορίζει το duty-cycle των παλμών ελέγχου του τρανζίστορ ώστε η έξοδος να σταθεροποιείται στα 5V.

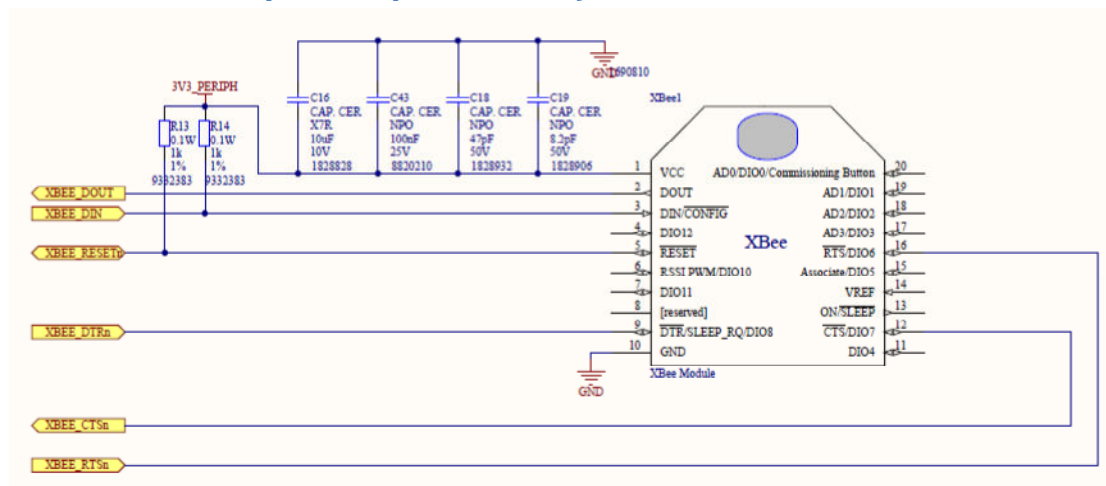
Τροφοδοσία 3,3VDC (PSU_3V3_A, PSU_3V3_B):



Σχ.Α.4.2) Τα δύο ανεξάρτητα υποσυστήματα τροφοδοσίας 3,3V

Υλοποιήθηκαν δύο πανομοιότυποι γραμμικοί ρυθμιστές τάσης με είσοδο 5V και έξοδο 3,3V, με τη βοήθεια του Low Dropout ρυθμιστή τάσης LM1117. Η έξοδος του ενός ρυθμιστή τροφοδοτεί τον μικροελεγκτή και η έξοδος του άλλου τα περιφερειακά εξαρτήματα επί της συσκευής ενώ η ίδια έξοδος προσφέρεται και στον χρήστη με τη βοήθεια ειδικού συνδέσμου.

(A.5) Το υποσύστημα επικοινωνίας με το μόντεμ XBEE (ZigBee®, 802.15.4, Wi-Fi ή άλλα πρωτόκολλα)

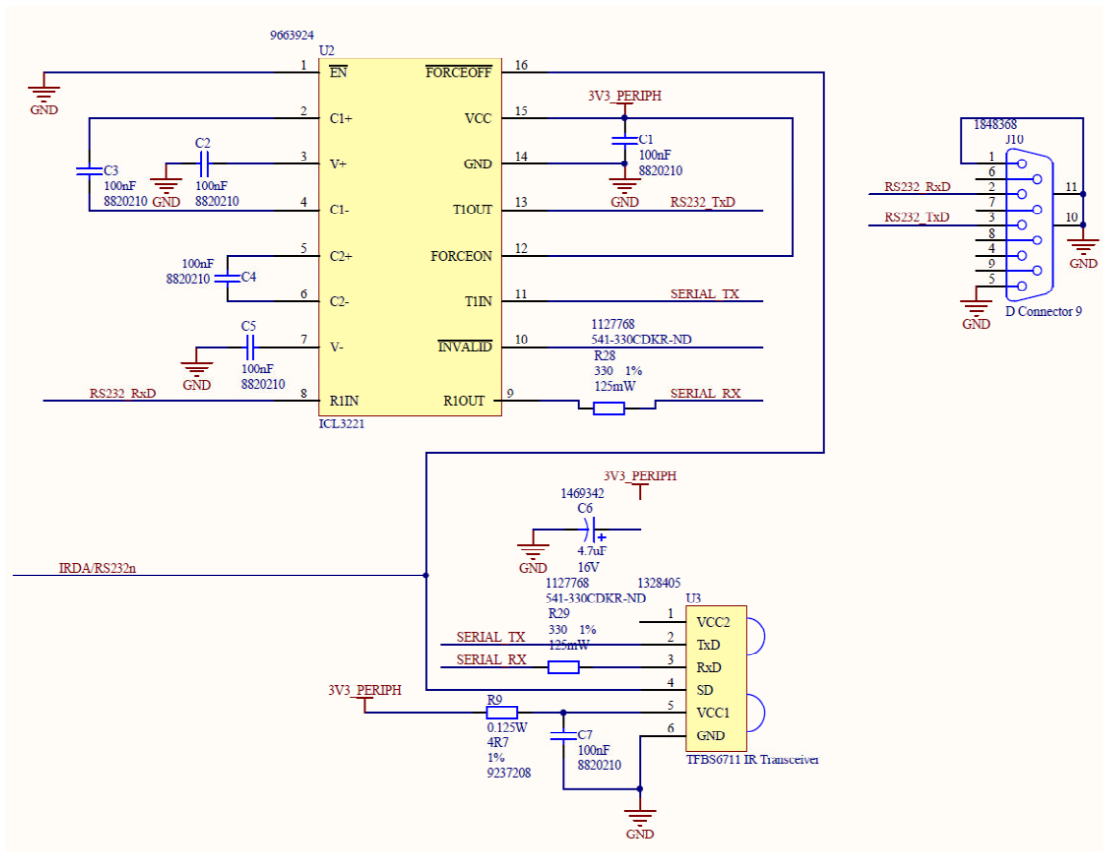


Σχ.Α.5) Το υποσύστημα επικοινωνίας γύρω από το XBEE

Για την αποτελεσματική διεύρυνση των δυνατοτήτων της συσκευής στην ασύρματη επικοινωνία, περιλαμβάνεται μονάδα διασύνδεσης σε δίκτυο 802.15.4 (XBEE24-ASI-001 της Digi®). Εναλλακτικά μπορεί να συνδεθεί κάποια άλλη μονάδα για σύνδεση σε άλλο δίκτυο (π.χ. το modem για ZigBee® XB24-Z7WIT-004).

Οι πυκνωτές C46,C43,C18,C19 απορρίπτουν διάφορες φασματικές συριστώσεις καθώς η τάση τροφοδοσίας απαιτείται να είναι εξαιρετικά καθαρή. Συνδέονται μόνο η τροφοδοσία και τα σήματα για σειριακή επικοινωνία και flow control. Οι αντιστάσεις R13,R14 είναι για pull-up των σημάτων #RESET και DIN.

(A.6) Το υποσύστημα επικοινωνίας με RS232 ή IrDA



Σχ.Α.6) Το υποσύστημα επικοινωνίας με RS232 ή IrDA

Η σειριακή θύρα UART2 μπορεί να χρησιμοποιηθεί είτε ως RS232 είτε ως IrDA. Η επιλογή γίνεται με το σήμα IRDA/RS232n

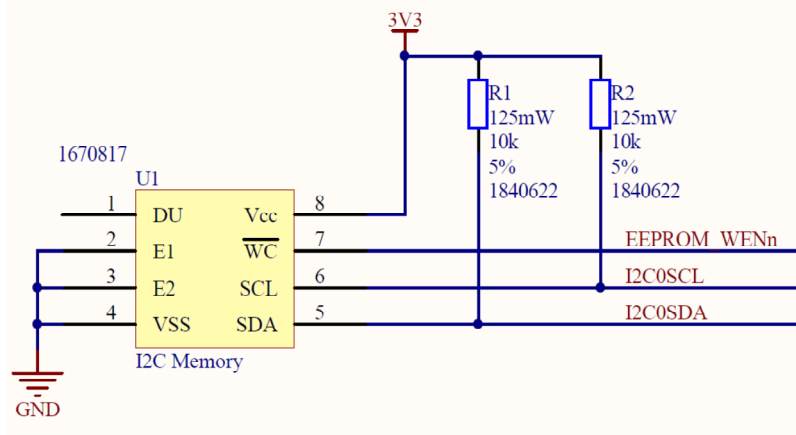
RS232(IRDA/RS232n high):

Με τη βοήθεια του πομποδέκτη για RS232 (ICL3221 της intersil), πραγματοποιείται η απαραίτητη προσαρμογή του επιπέδου των τάσεων από RS232 σε TTL και αντίστροφα. Οι πυκνωτές C2,C3,C4,C5 χρησιμοποιούνται ώστε να παράγονται από την τροφοδοσία των 3,3Volt οι τάσεις $\pm 5\text{Volt}$ για την έξοδο RS232. Για τον σκοπό αυτόν, εντός του ολοκληρωμένου υλοποιούνται αντλίες φορτίου και αντιστροφείς.

IrDA(IRDA/RS232n low):

Με τη βοήθεια του πομποδέκτη για IrDA (TFBS6711 της Vishay) είναι εφικτή η λήψη και εκπομπή σημάτων IrDA.

(Α.7) Η μνήμη I2C EEPROM

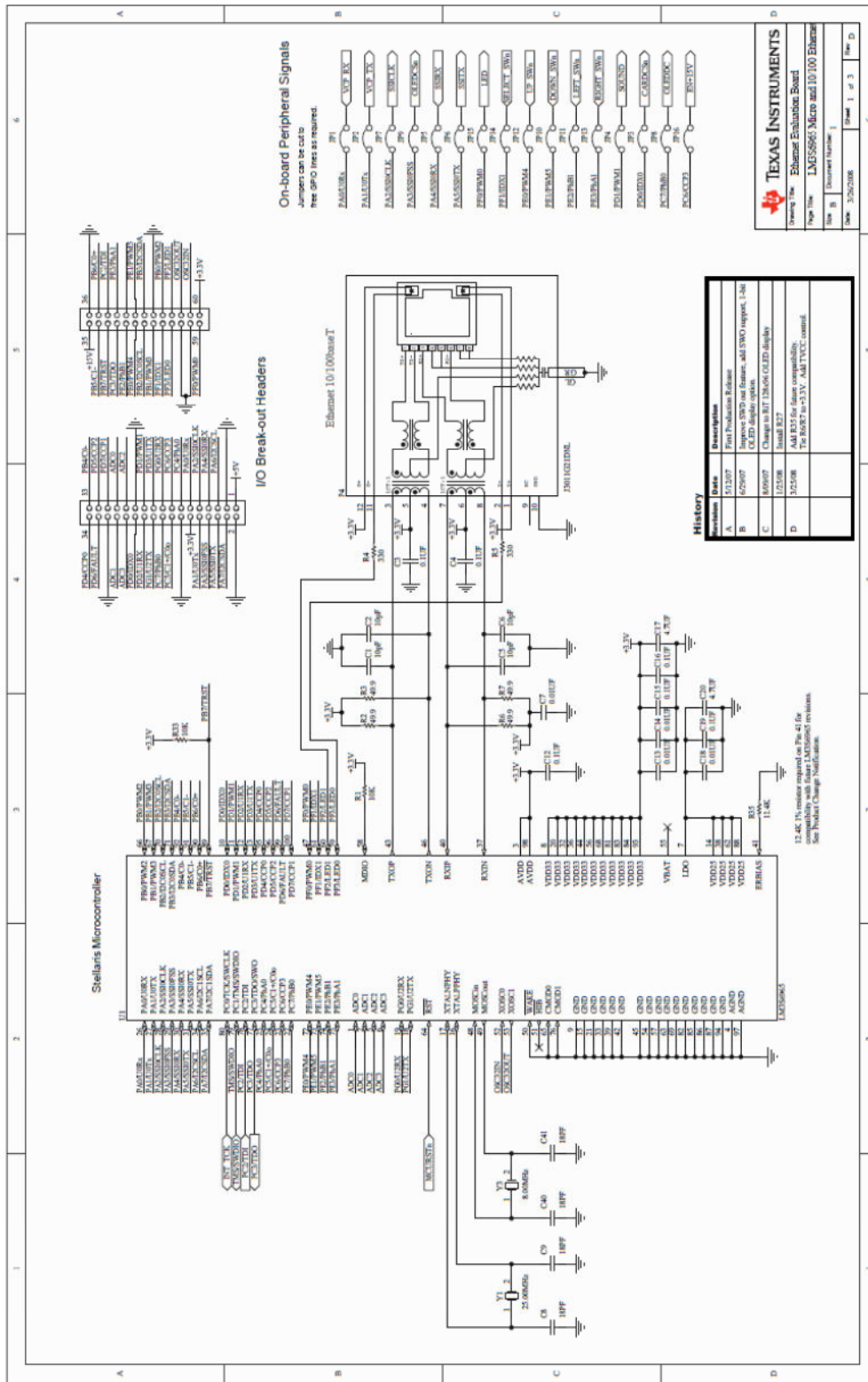


Σχ.Α.7) Η μνήμη I2C EEPROM

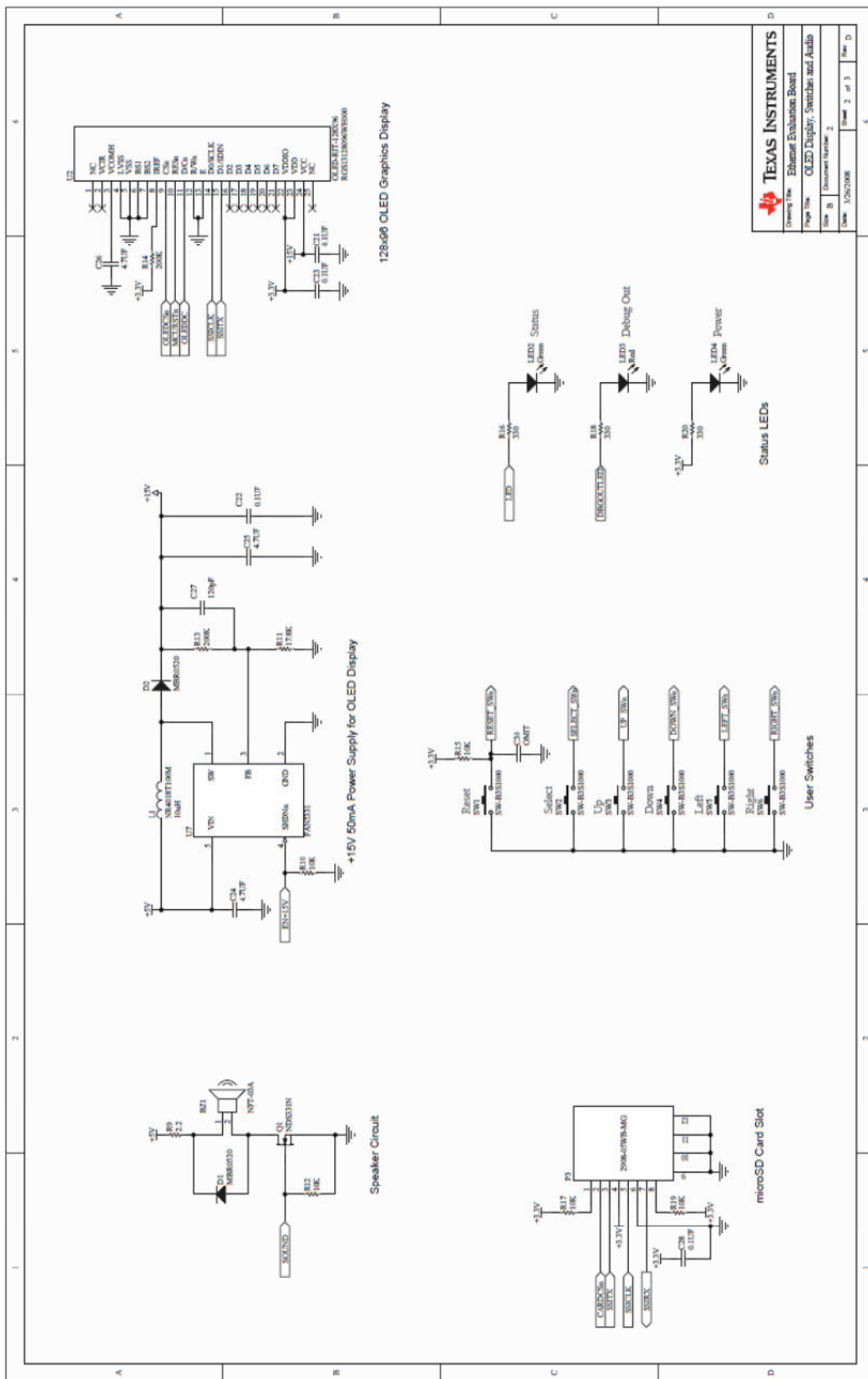
Η συσκευή διαθέτει και έναν επιπλέον τρόπο μόνιμης αποθήκευσης, με τη βοήθεια της μνήμης EEPROM I2C, μεγάλης χωρητικότητας (1MBit - M24M01).

(A.8) Η πλακέτα του Evaluation Kit EKLM3S6965 της Texas Instruments

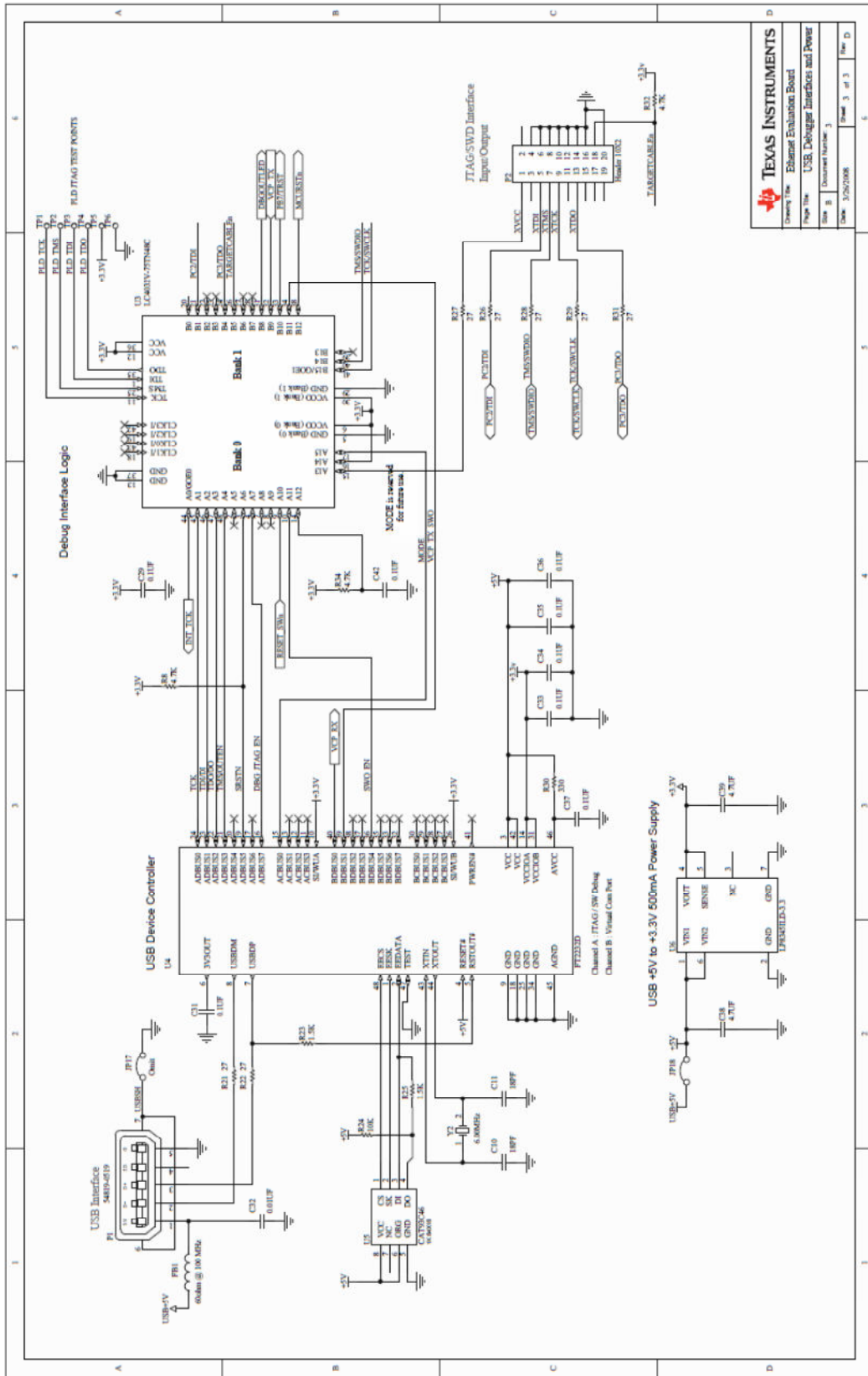
Το λογικό κύκλωμα του Evaluation Board, όπως δίδεται στο [3]



Ο μικροελεγκτής και το 10/100 Ethernet

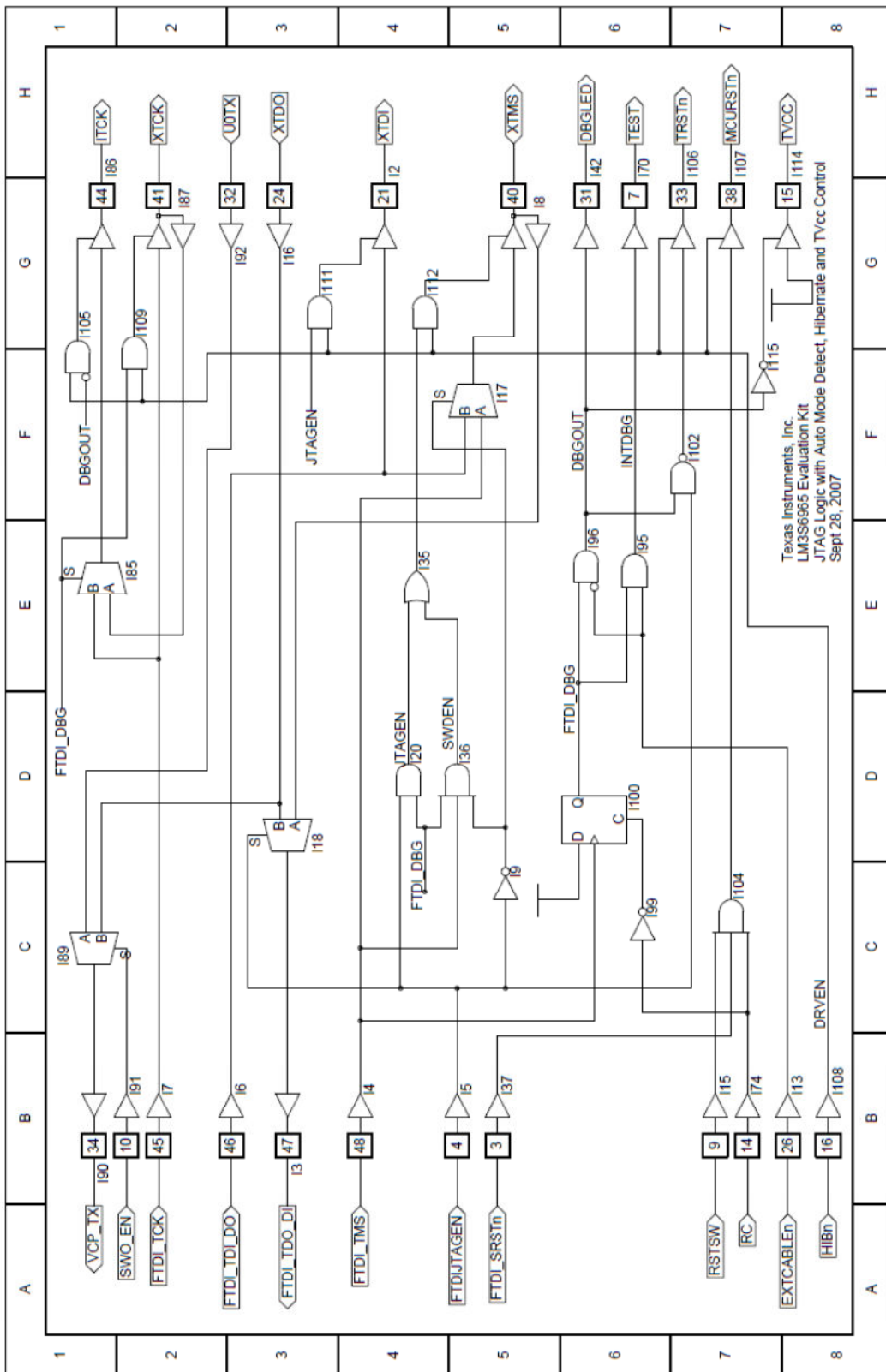


Το OLED display, οι διακόπτες και το σύστημα Audio

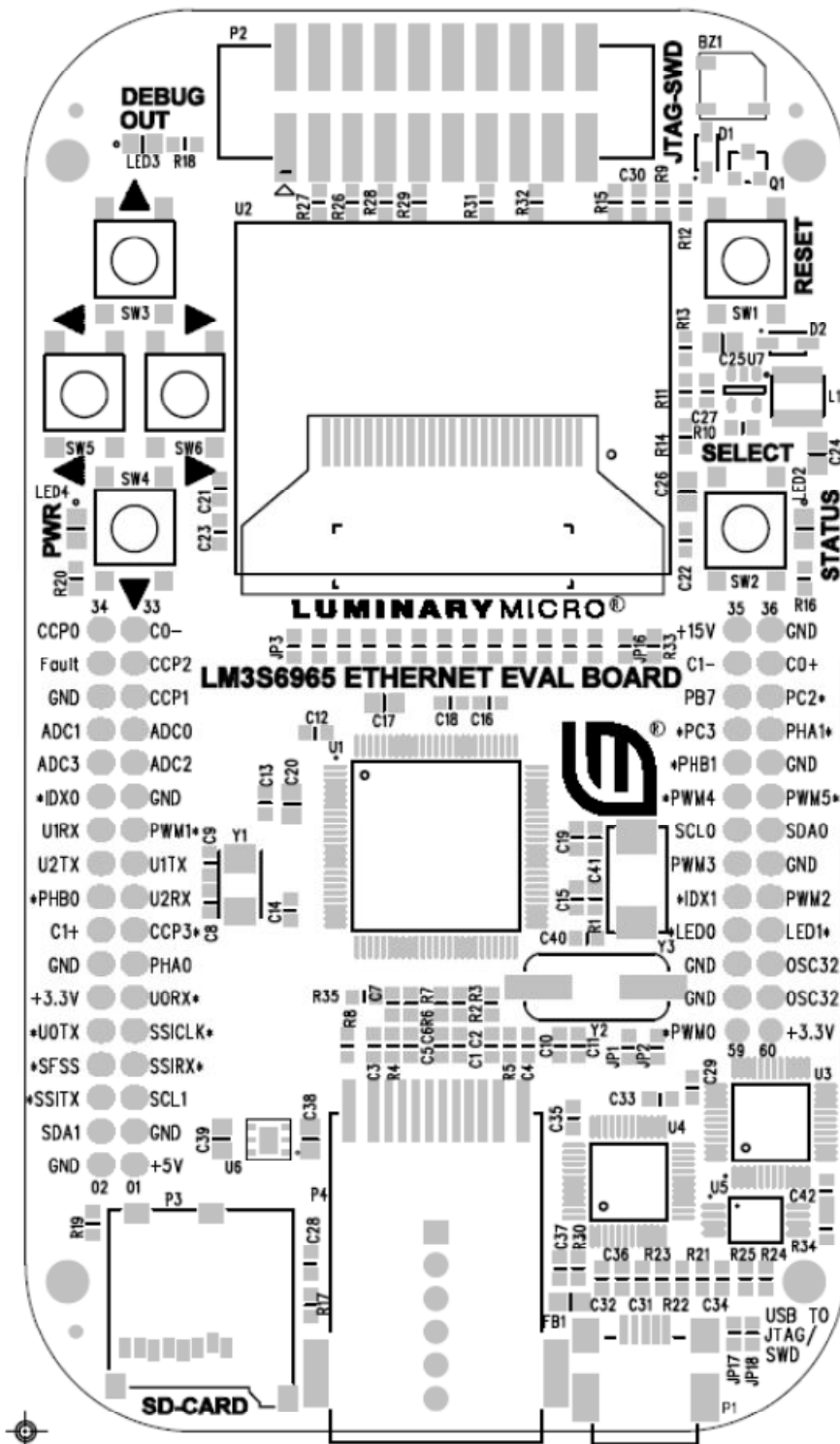


TEXAS INSTRUMENTS			
Drawn: Tim	Checked: Evgeniy	Approved: [Signature]	Rev: D
Evaluating Board			
USB Debug Interface and Power			
Sheet: 3	of: 3	Doc: 1-02020008	Doc Number: 3

To USB, to debugging interface και το τροφοδοτικό



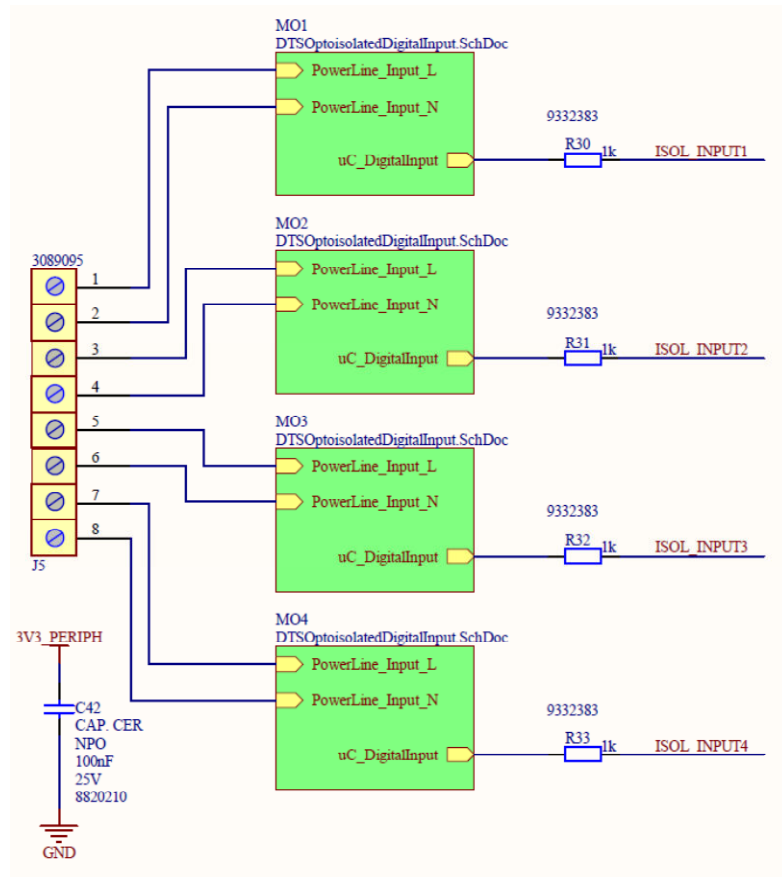
Προσαρμογή τάσεων, muxing, boolean op. και latching (glue logic)



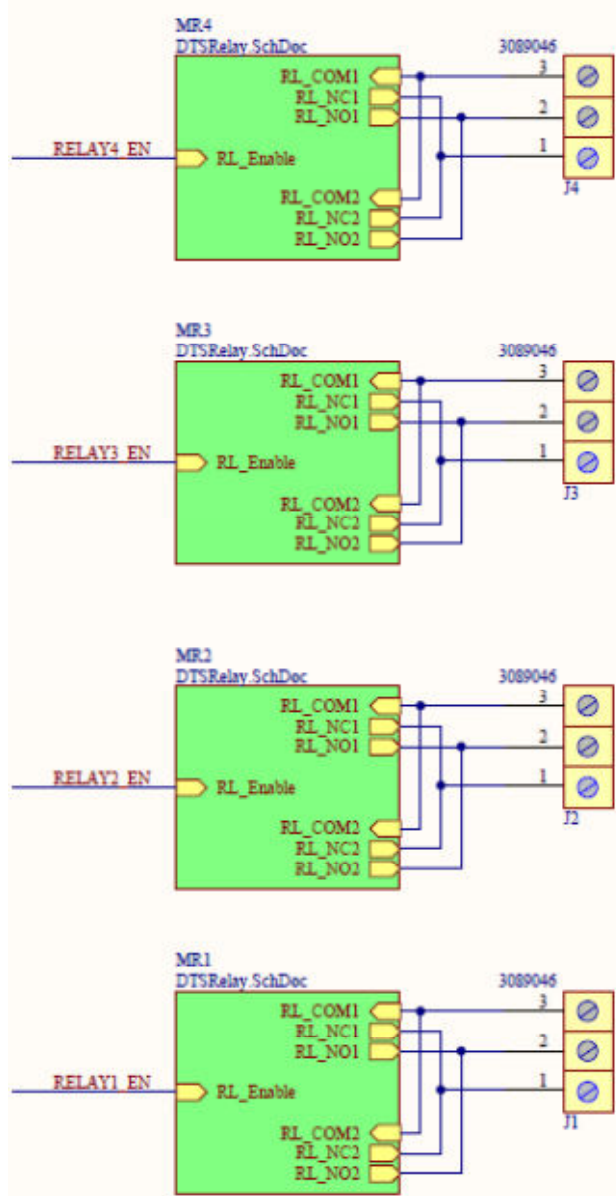
Σημαντική σημείωση: Η χρήση του Evaluation Board δεν είναι απαραίτητη. Μπορεί να αναπτυχθεί μια πλακέτα με τον LM3S6965 που θα ενσωματώνει όποιες λειτουργίες χρειάζεται ο σχεδιαστής, είτε διατηρώντας τα περιφερειακά που επιθυμεί είτε προσθέτοντας όποια άλλα εξαρτήματα χρειάζεται. Πάντως, επειδή ο «Πανζεύκτης» προσφέρει τις τροφοδοσίες 5V, 3,3V, δεν είναι απαραίτητη η υλοποίηση τροφοδοτικού παρά μόνο για τα 15V του OLED, αν αυτό χρησιμοποιείται.

(B) Διασύνδεση

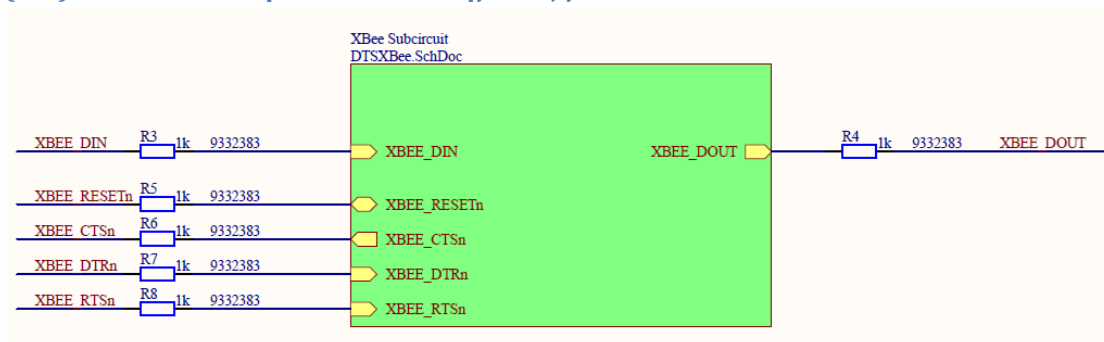
(B.1) Η διασύνδεση των μονάδων ανίχνευσης εισόδων 220V με οπτική απομόνωση



(B.2) Η διασύνδεση των μονάδων ρελέ φορτίου



(B.3) Η διασύνδεση του υποσυστήματος με το XBEE

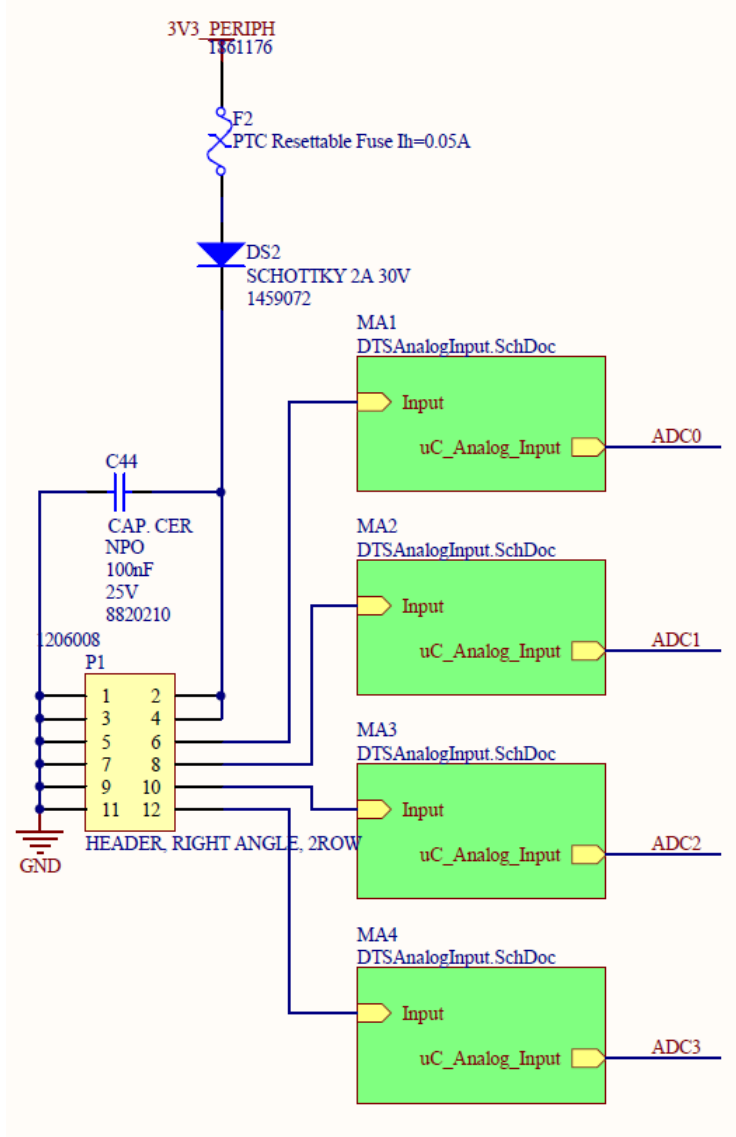


(B.4) Η διασύνδεση του τροφοδοτικού

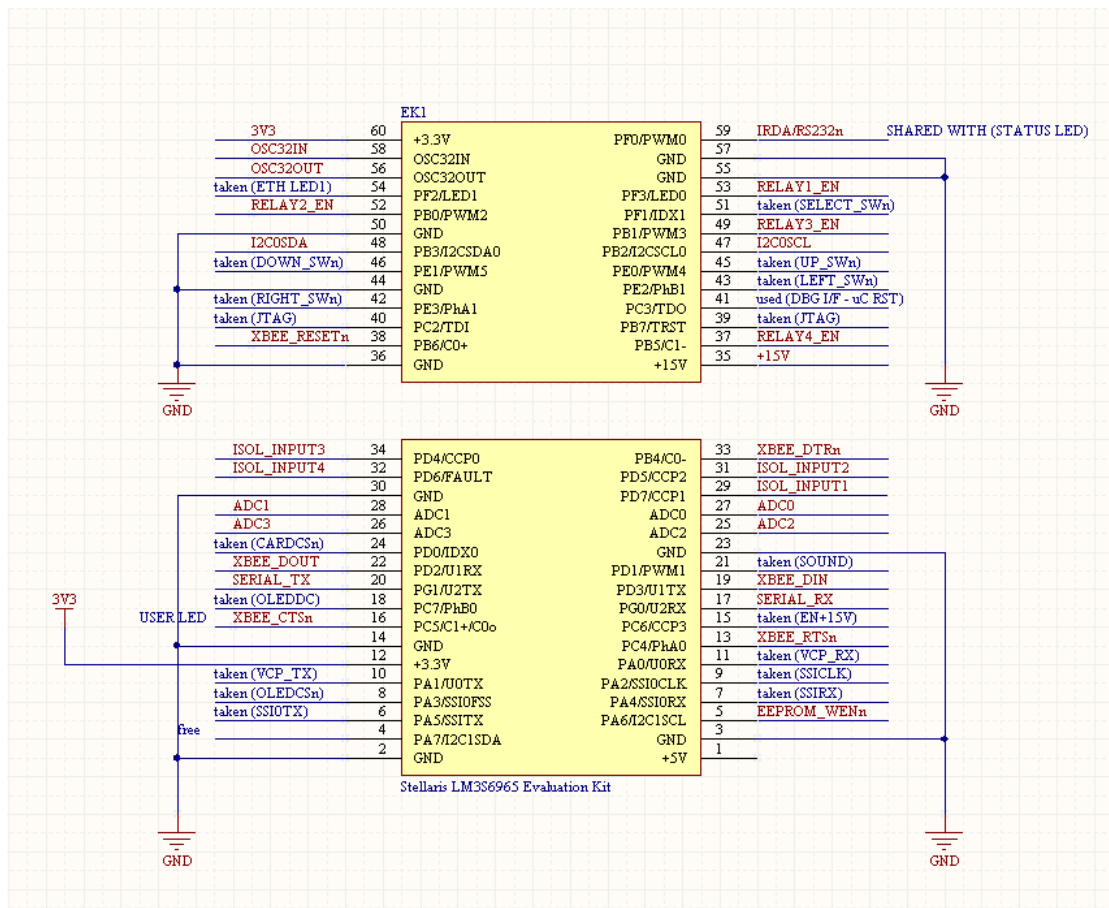
Power Supply Module
DTSPowerSupply.SchDoc



(B.5) Η διασύνδεση των μονάδων για τη λήψη αναλογικών μετρήσεων

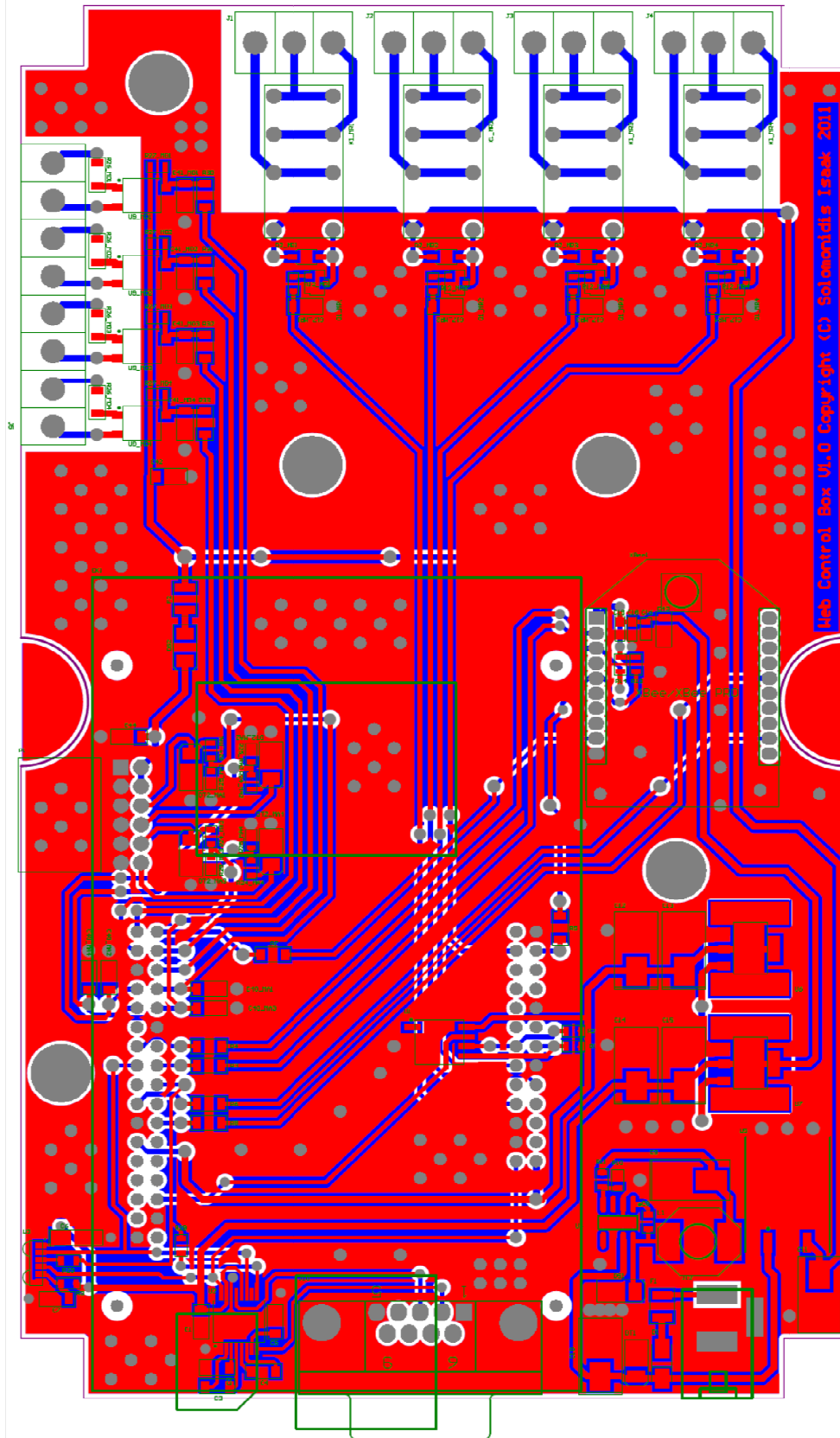


(B.6) Η διασύνδεση του Evaluation Board ή της πλακέτας του μικροελεγκτή



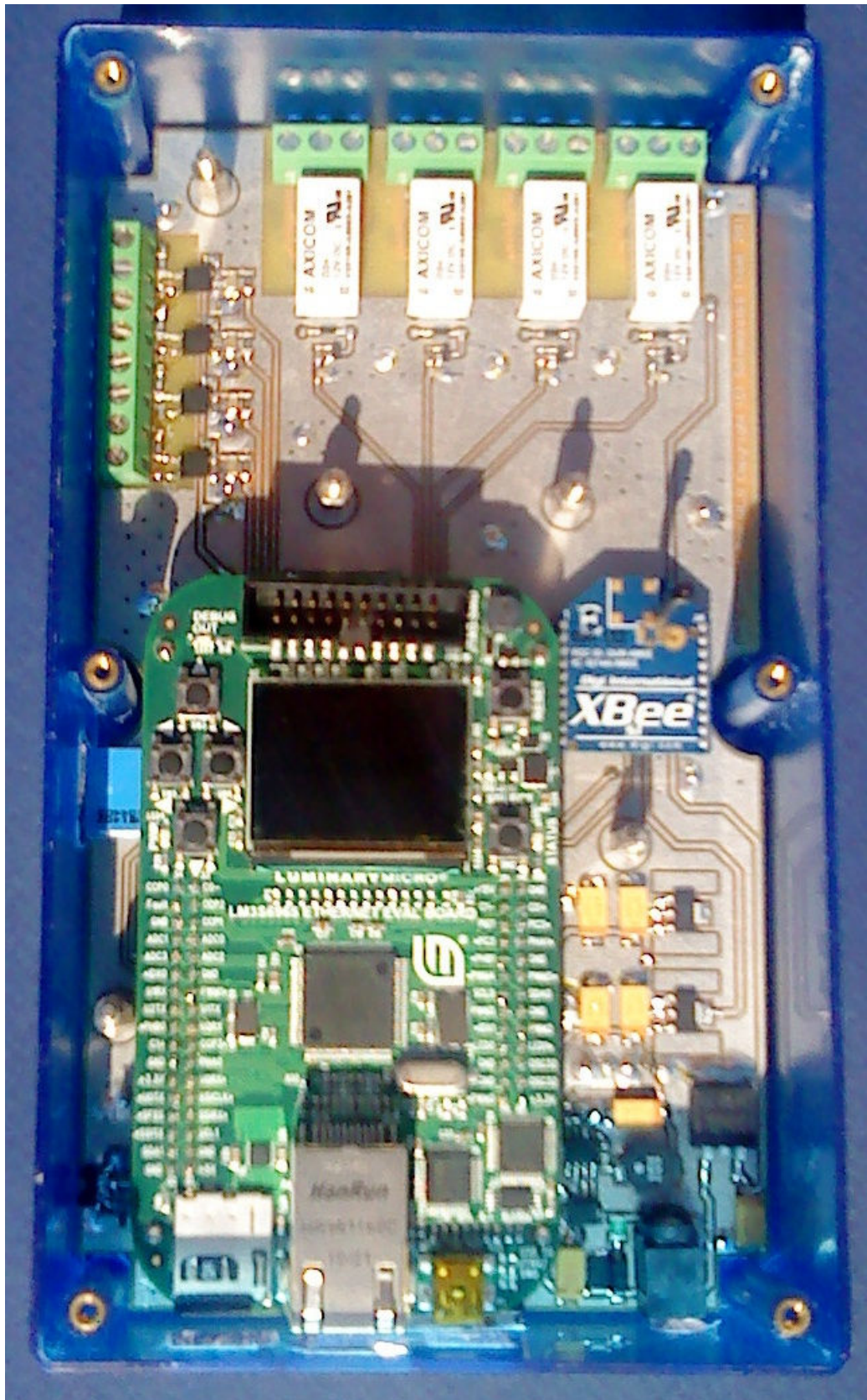
Προσοχή: Στην περίπτωση που συνδέεται το Evaluation Board, δε θα συνδεθεί η τροφοδοσία των 3,3V του “Πανζεύκτη», παρά μόνο η τροφοδοσία των 5V. Απαραίτητα, θα πρέπει τότε με ένα κοπίδι να κοπεί η γραμμή του JP18, ώστε να αποσυνδεθεί η γραμμή τροφοδοσίας 5V του USB από το regulator(U6) των 3,3V.

(III). Το τυπωμένο κύκλωμα

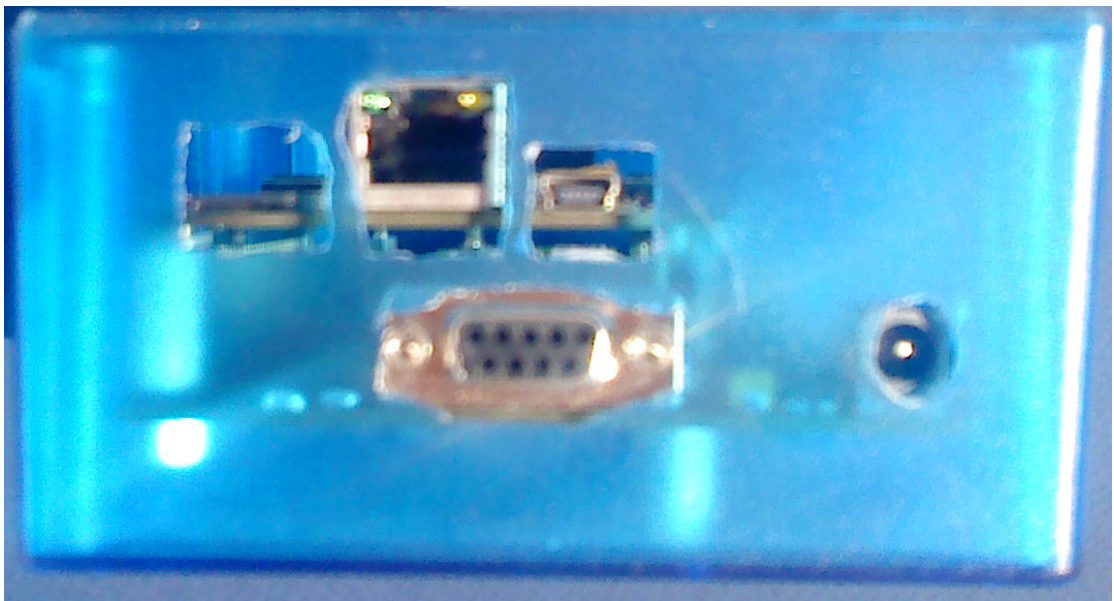
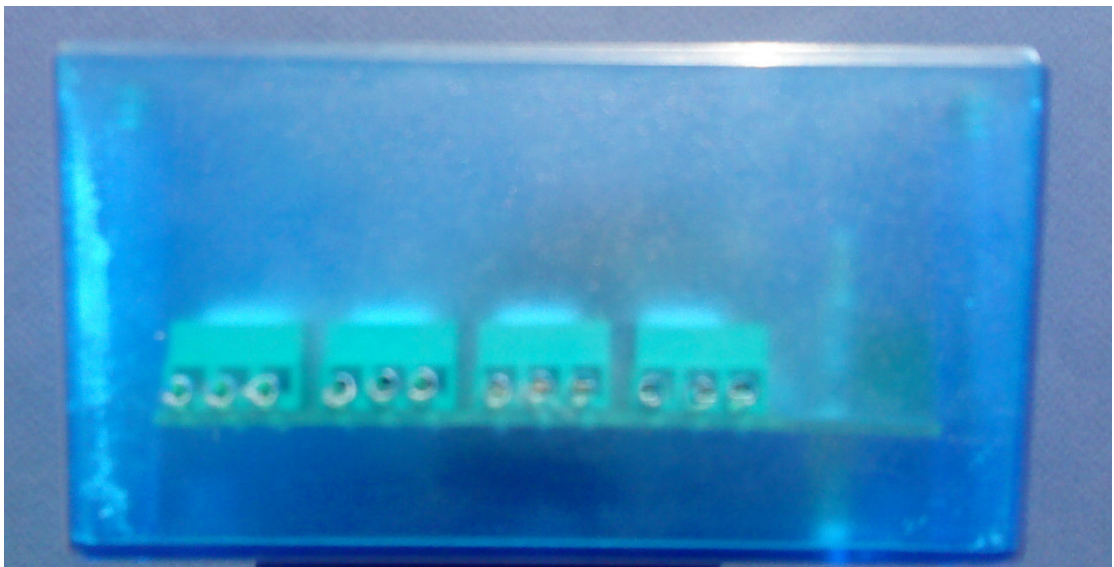


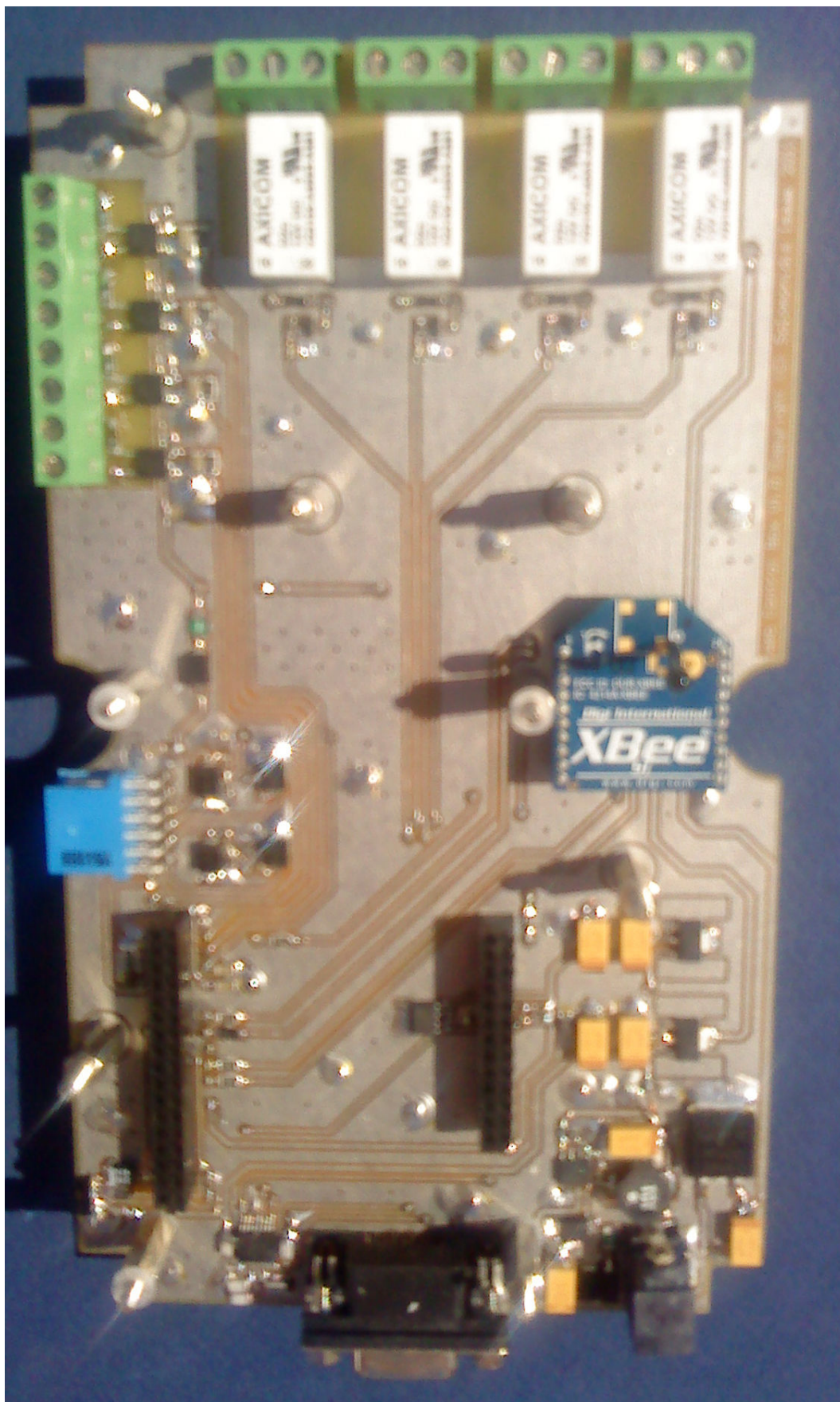
Η κατασκευή έγινε σε πλακέτα FR4 χωρίς επιμεταλλωμένες οπές.

(IV). Όψεις της ολοκληρωμένης κατασκευής

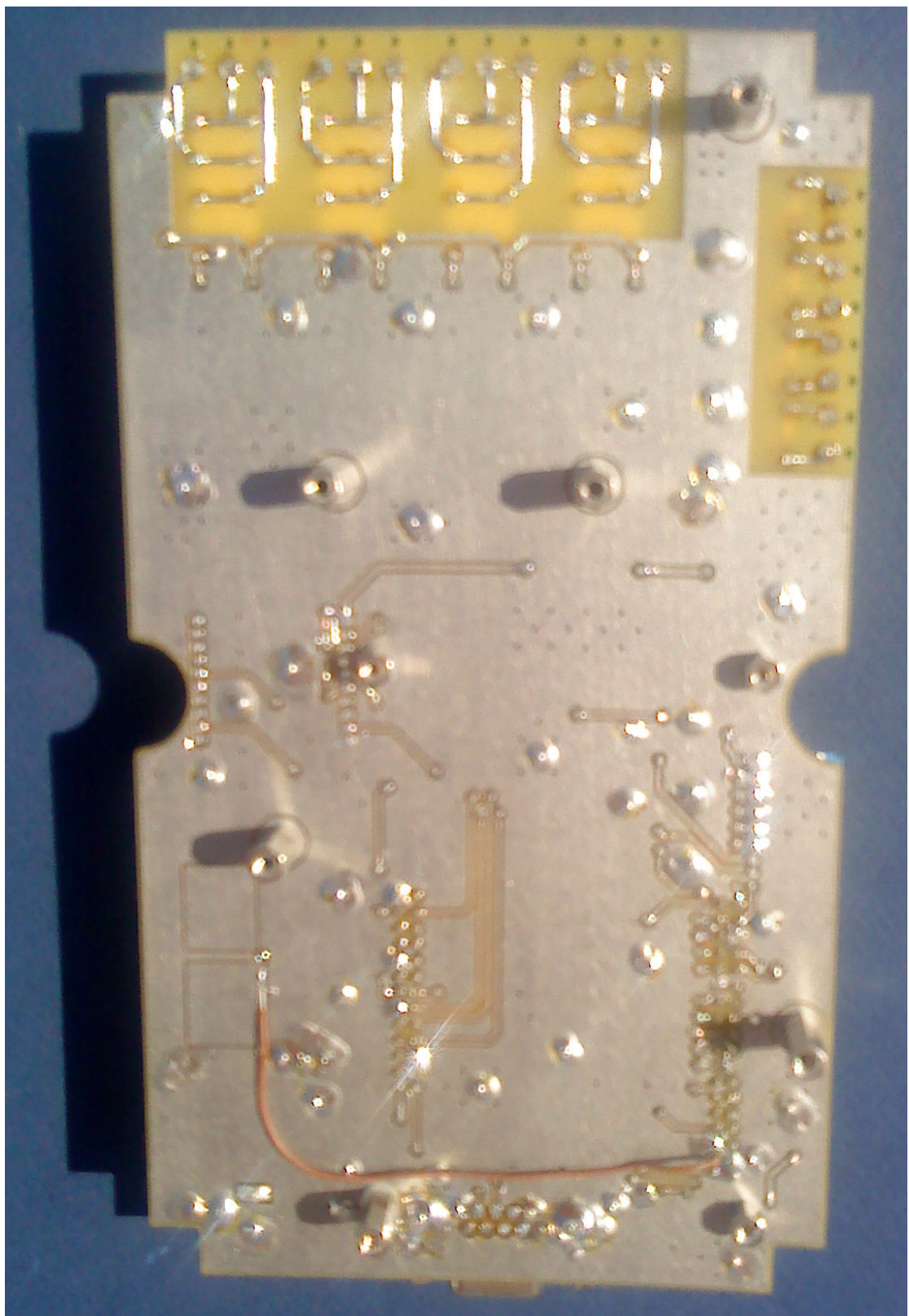








Η κεντρική πλακέτα (εμπρόσθια όψη)



Η κεντρική πλακέτα (πίσω όψη)

V. Το “Panzerterminal”

Για την άμεση υλοποίηση εφαρμογών πληροφόρησης και διαχείρισης στο σπίτι και σε κτίρια γενικότερα (domotics) και διάχυτης νοημοσύνης (ambient intelligence), είναι απαραίτητες και κάποιες περιφερειακές συσκευές που, αν και διαθέτουν σημαντικά μικρότερη υπολογιστική ισχύ, εντούτοις έχουν κύριο ρόλο στις εφαρμογές αυτές.

Με στόχο την επαρκή κάλυψη ενός μεγάλου εύρους αναγκών, αναπτύχθηκε το «pan-zerterminal». Πρόκειται για μια περιφερειακή συσκευή που επικοινωνεί ασύρματα με τον «Πανζεύκτη» και επιτρέπει την εποπτεία και έλεγχο συσκευών και συστημάτων. Έχει μικρό μέγεθος (148 x 63mm) και διαθέτει τα ακόλουθα χαρακτηριστικά:

- Τροφοδοσία 220VAC
- Ψηφιακή είσοδος 220V με οπτική απομόνωση
- Έξοδος ελέγχου φορτίου με ρελέ δύο θέσεων 220VAC/3A
- Έξοδος ρυθμιζόμενη PWM sink 50mA έως 12V
- Αναλογική είσοδος 0%3,3V
- Ενσωματωμένο ποτενσιόμετρο
- Επαφή διακόπτη push-on
- LED ένδειξης τροφοδοσίας
- LED ένδειξης σύνδεσης του μόντεμ XBEE
- 2 LED γενικής χρήσης ελεγχόμενα από το XBEE
- Παροχή τροφοδοσίας 12V σε εξωτερικά κυκλώματα

Περιγραφή του κυκλώματος.

Τροφοδοτικό: Η τάση του δικτύου, μέσω ενός μετασχηματιστή 220/18, ανορθώνεται από τη γέφυρα D1, περνά από τηκόμενη ασφάλεια και εξομαλύνεται με τη βοήθεια του C1. Αυτή η τάση διοζτεύεται σε ένα διακοπτικό τροφοδοτικό όμοιο με του «Πανζεύκτη», το οποίο όμως παρέχει τάση 3,3Volt αφού έχουν επιλεγεί κατάλληλά οι τιμές του δικτυώματος των R1,R3 (βλ. Αντίστοιχη ανάλυση στον «Πανζεύκτη») καθώς και στον γραμμικό μετατροπέα για την τροφοδότηση του ρελέ και παροχή 12V τροφοδοσίας σε εξωτερικές συσκευές.

Ψηφιακή είσοδος 220V με οπτική απομόνωση

Όπως και στον «Πανζεύκτη». Η είσοδος καταλήγει στο DIO11 του XBEE. Για την υποστήριξη αυτής της λειτουργίας, ανατρέξτε στο εγχειρίδιο του XBEE, στο «ADC and Digital I/O Line Support». Καθώς το XBEE υποστηρίζει sampling με ρυθμό έως και 1ms, μπορεί να ρυθμιστεί έτσι, ώστε να αποστέλει δεδομένα όταν έχει συμπληρώσει αρκετά δείγματα ώστε ο «Πανζεύκτης» να αποφασίσει για την κατάσταση της εισόδου AC. Η ίδια είσοδος μπορεί να χρησιμοποιηθεί και για DC σήματα.

Έξοδος ελέγχου φορτίου με ρελέ δύο θέσεων 220VAC/3A: Όπως και στον «Πανζεύκτη». Ο έλεγχος γίνεται από το DIO3 του XBEE.

PWM: Η έξοδος PWM του XBEE οδηγεί ένα τρανζίστορ, το οποίο με τη σύνδεση οιουδήποτε φορτίου στον συνδετήρα J3, προσφέρει PWM έλεγχο σε φορτία με sink έως 50mA, με μέγιστη τάση τα 12volt.

Αναλογική είσοδος 0%3,3V: Συνδέεται στην αναλογική είσοδο AD1 του XBEE.

Ενσωματωμένο ποτενσιόμετρο: Συνδέεται στην αναλογική είσοδο AD2 του XBEE.

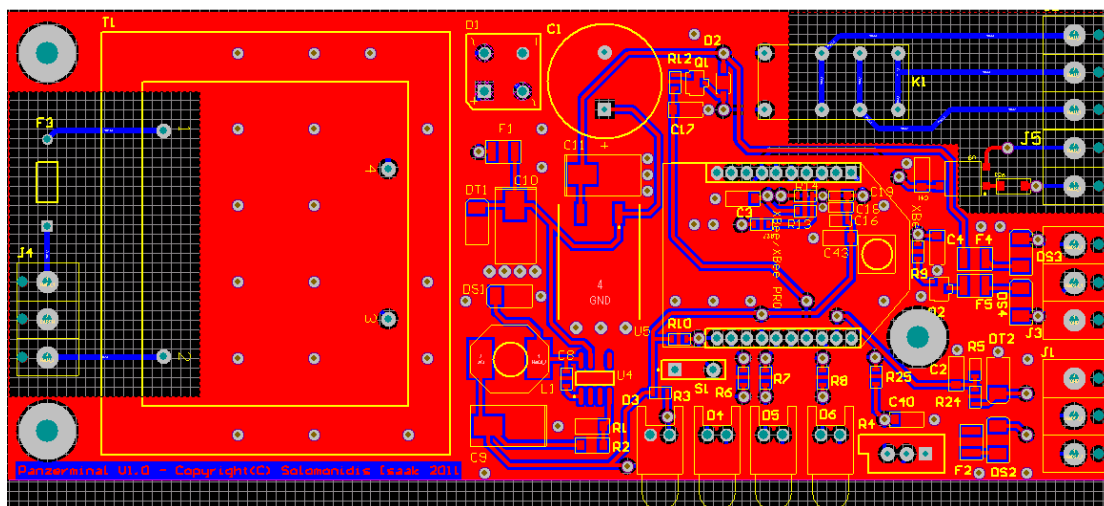
Επαφή διακόπτη push-on: Συνδέεται στην ψηφιακή είσοδο DIO4 του XBEE.

LED ένδειξης τροφοδοσίας: Καταδεικνύει την ύπαρξη των 3,3V

LED ένδειξης σύνδεσης του μόντεμ XBEE: Συνδέεται με την έξοδο Associate(DIO5) του XBEE και αναβοσβήνει όταν το τερματικό έχει συνδεθεί.

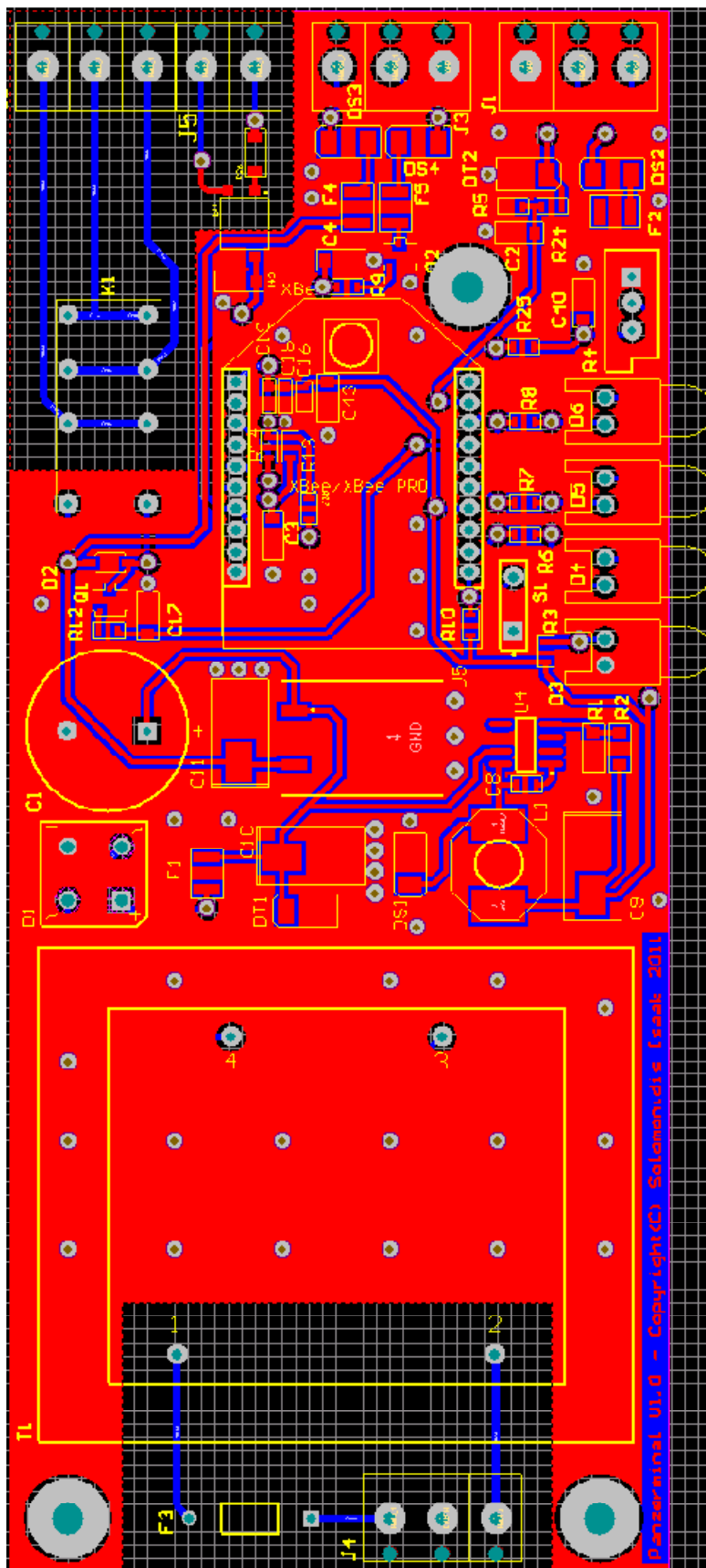
2 LED γενικής χρήσης ελεγχόμενα από το XBEE: Συνδέονται με τις εξόδους DIO7(το D4) και DIO6(το D6).

Παροχή τροφοδοσίας 12V σε εξωτερικά κυκλώματα: Παρέχει 12V τροφοδοσία έως 0,5A σε εξωτερικές συσκευές, μέσω του J3.



Η πλακέτα του “Panzerterminal” σε φυσικό μέγεθος

Σημείωση: Για δίκτυο ZigBee διατίθεται και έκδοση του XBEE module με επιπλέον επεξεργαστή ο οποίος διαθέτει 32kByte μνήμη flash και 2kByte RAM. Έτσι, μπορεί για παράδειγμα το “Panzerterminal” να λειτουργήσει αυτόνομα ως απλός ελεγκτής ή και να εκτελεί τοπικά δειγματοληψία, να επεξεργάζεται τα δεδομένα και να στέλνει τα αποτελέσματα στον “Πανζεύκτη».

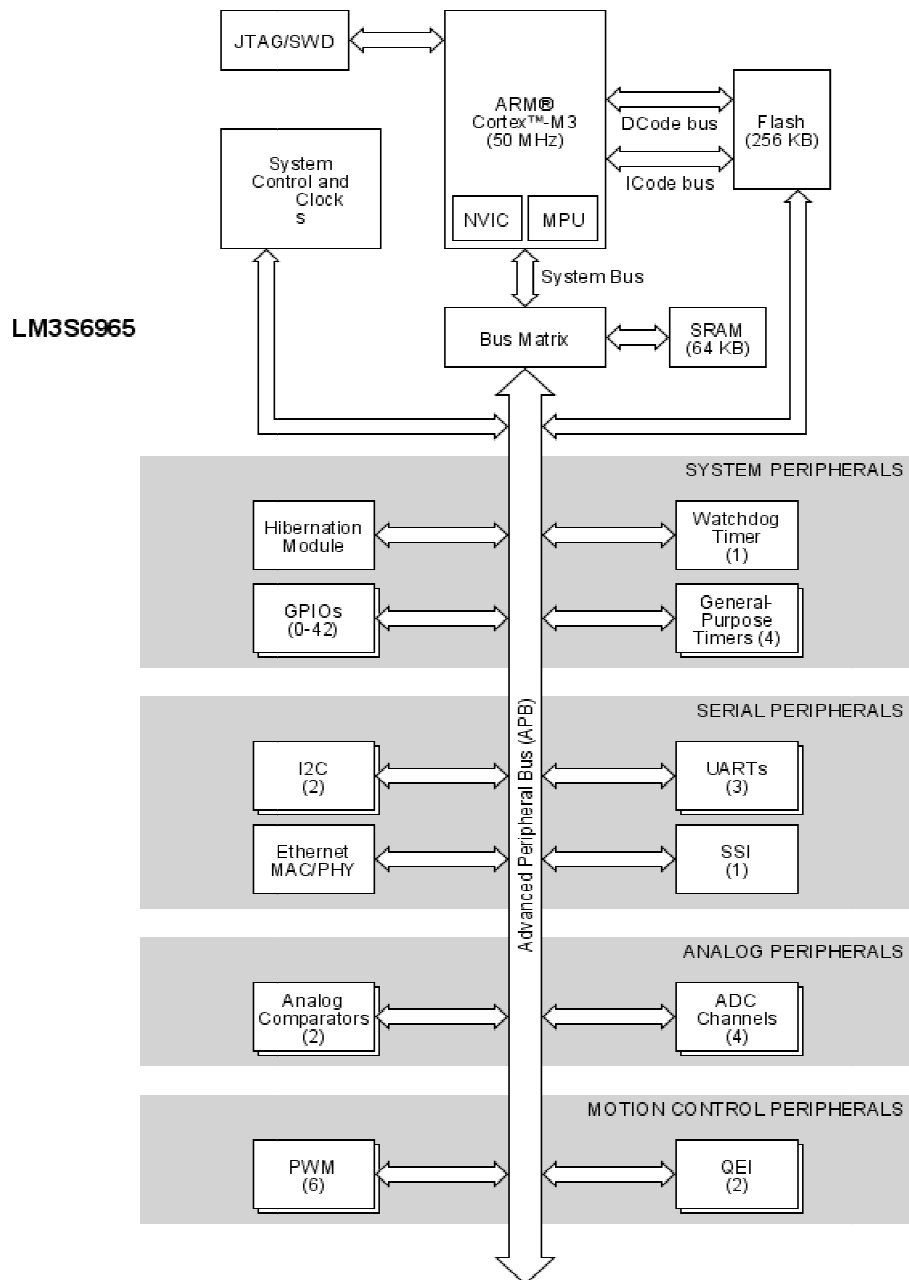


Η πλακέτα του “Panzerterminal” (σε μεγέθυνση)

Ο μικροελεγκτής Stellaris® LM3S6965 (ARM 32Bit)

Διαγράμματα

Λειτουργικό διάγραμμα



Περιγραφή

Ο μικροελεγκτής stellaris® LM3S6965 βασίζεται στον πυρήνα ARM Cortex®™-M3 με συχνότητα λειτουργίας μέχρι τα 50 MHz, με 256 kB flash και 64 kB SRAM. Ο LM3S6965 διαθέτει επίσης βιομηχανική συνδεσιμότητα, σε πραγματικό χρόνο με 10/100 Ethernet MAC / PHY, έναν ελεγκτή SSI / SPI ελεγκτή, 2 διεπαφές I2C και 3 UART. Ο μικροελεγκτής LM3S6965 διαθέτει προηγμένα χαρακτηριστικά για συστήματα ελέγχου κίνησης, συμπεριλαμβανομένων 6 PWM εξόδους ελέγχου κίνησης με νεκρούς-χρόνους, εισόδους κωδικοποιητή δύο τεταρτημορίων για ακριβή παρακολούθηση της θέσης, και 1 είσοδο προστασίας από τα σφάλματα για την άμεση απενεργοποίηση απενεργοποίηση. Ο μικροελεγκτής διαθέτει επίσης ευφυείς δυνατότητες διαχείρισης αναλογικού σήματος, μεταξύ των οποίων 2 αναλογικούς συγκριτές και 4 κανάλια μετατροπέα αναλογικό σε ψηφιακό υψηλής ακρίβειας 10-bit, με δυνατότητα δειγματοληψίας έως και 1 εκατ. δείγματα ανά δευτερόλεπτο.

Χαρακτηριστικά - Δυνατότητες

- 32-bit ARM Cortex®™-M3 50-MHz core επεξεργαστής με Χρονιστή Συστήματος (SysTick), ολοκληρωμένο Φωλιασμένο Διανυσματικό Ελεγκτή διακοπών (Nested Vectored Interrupt Controller - NVIC), Μονάδα Προστασίας Μνήμης (Memory Protection Unit - MPU), και σετ εντολών Thumb2.
- Αποσφαλμάτωση πλήρους λειτουργικότητας με πρόσβαση μέσω JTAG, διεπαφές Serial Wire, και ελεγκτή Test Access Port (TAP) συμβατό με IEEE 1149.1-1990.
- 256 KB ενός κύκλου Flash και 64 KB μόνο κύκλου SRAM.
- Μονάδα αδρανοποίησης χαμηλής ισχύος υποστηριζόμενη από μπαταρία με ρολόι πραγματικού χρόνου (Real-Time Clock – RTC).
- 0-42 Γενικής Χρήσης Είσοδοι/Εξοδοι (GPIO) (ανάλογα με τη διαμόρφωση) με προγραμματιζόμενο έλεγχο για διακοπές GPIO και διαμόρφωση pad.
- Συμβατός με ARM FiRM Χρονιστής Επιτήρησης(Watchdog Timer), καθώς και τέσσερις γενικής χρήσης μονάδες χρονοδιακόπτη (GPTM), κάθεμα από τις οποίες παρέχει δύο 16-bit χρονόμετρα / μετρητές και μπορεί να ρυθμιστεί ώστε να λειτουργεί ανεξάρτητα
- 10-bit αναλογικό σε ψηφιακό μετατροπέας (ADC) με τέσσερα αναλογικά κανάλια εισόδου και συχνότητα δειγματοληψίας ένα εκατομμύριο δείγματα / δευτερόλεπτο
- Τρία πλήρως προγραμματιζόμενα τύπου 16C550 UART s με υποστήριξη Ir-DA.
- Synchronous Serial Interface (SSI), που υποστηρίζουν Freescale SPI, MICROWIRE, ή σύγχρονες σειριακές διεπαφές της Texas Instruments.

- Δύο Inter-Integrated Circuit (I²C) μονάδες διεπαφής, παρέχοντας Standard(100 Kbps) και Fast (400 Kbps) επικοινωνία και υποστήριξη για την αποστολή και λήψη δεδομένων είτε ως master ή ως slave.
- 10/100 Ethernet Controller, με εκτεταμένες δυνατότητες διαμόρφωσης, που είναι σύμφωνο με τις προδιαγραφές IEEE 802.3-2002 με πλήρη και ημιαμφίδρομο τρόπο λειτουργίας τόσο για 100 Mbps όσο και για 10 Mbps λειτουργία, καθώς και αυτόματη διόρθωση MDI / MDI-X cross-over.
- Δύο ολοκληρωμένοι αναλογικοί συγκριτές, διαμορφώσιμοι για έξοδο για την εκκίνηση μιας ακολουθίας μετατροπής αναλογικό-σε-ψηφιακό, να οδηγήσει μια έναν ακροδέκτη εξόδου ή να παράξει μια διακοπή.
- Τρεις μονάδες γεννήτριας PWM, με μετρητή 16-bit η καθεμία, δύο συγκριτές PWM, γεννήτρια σήματος PWM, μια γεννήτρια νεκρού χρόνου και έναν επιλογέα διακοπή / σκανδαλισμός ADC.
- Δύο μονάδες Quadrature Encoder Interface (QEI) που διαθέτουν ολοκληρωτή θέσης για την παρακολούθηση της θέσης του κωδικοποιητή και την καταγραφή της ταχύτητας με τη χρήση ενσωματωμένου χρονοστή.

Τεχνικά Χαρακτηριστικά

	LM3S6965
Flash (KB)	256
RAM (KB)	64
StellarisWare σε ROM	Όχι
Φορτωτή Εκκίνησης σε ROM	0
DMA	0
SafeRTOS	0
Μέγιστη ταχύτητα (MHz)	50
Ταλαντωτές ακριβείας	Όχι
Μονάδα Προστασίας Μνήμης (MPU)	Ναι
Χρονοδιακόπτες	5
RTC	Ναι
Χρονοδιακόπτες Watchdog	1
Κίνηση PWM	6
Υποσύστημα Βλάβης PWM	1
Dead-Band Generator	Ναι
Σύλληψη Pins	4
QEI	2
EPI / EMIF	Όχι

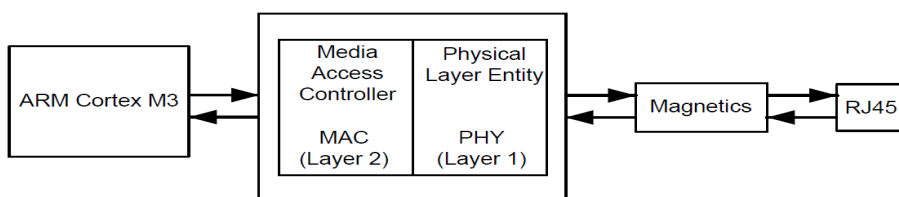
Ethernet	MAC + PHY
IEEE 1588	Όχι
CAN	0
USB D, H / D, ή OTG	Όχι
UART	3
UART Κατάσταση Modem	Όχι
I2C	2
SSI / SPI	1
I2S	Όχι
ADC Μονάδες	1
ADC Κανάλια	4
ADC Ανάλυση (bits)	10
ADC Ρυθμός δειγματοληψίας (kSPS)	1000
ADC Εξωτ. Αναφορά	Όχι
Ενσωματωμένος αισθητήρας θερμοκρασίας	1
Αναλογικοί συγκριτές	2
Ανεκτικά σε μέγιστη τάση 5-V GPIOs	42
Αποκλειστικά 5-V Tolerant GPIOs	0
Ψηφιακοί Συγκριτές	0
Ενότητα αδρανοποίησης με Υποστήριξη μπαταρίας	Ναί
Εσωτερικός ρυθμιστής τάσης LDO	Ναί
Ρεύμα κατά τη λειτουργία (mA) / MHz	εν αναμονή
Ρεύμα αναμονής (μΑ)	εν αναμονή
Θερμοκρασία λειτουργίας (C)	-40 Έως 105, -40 έως 85
Ακροδέκτες / Συσκευασία	100LQFP, 108NFBGA

Ο Ethernet Controller του LM3S6965

Ο Ethernet Controller του Stellaris, αποτελείται από έναν πλήρως ενσωματωμένο media access controller (MAC) και network physical (PHY) interface. Ο Ethernet Controller συμμορφώνεται στις προδιαγραφές *IEEE 802.3* και υποστηρίζει πλήρως τα πρότυπα 10BASE-T και 100BASE-TX.

Η μονάδα Stellaris Ethernet Controller έχει τα ακόλουθα χαρακτηριστικά:

- Συμμορφώνεται στις προδιαγραφές *IEEE 802.3-2002*
 - Συμβατό με 10BASE-T/100BASE-TX IEEE-802.3. Απαιτεί μόνο έναν διπλό 1:1 μετασχηματιστή απομόνωσης για διασύνδεση με τη γραμμή
 - 10BASE-T/100BASE-TX ENDEC, 100BASE-TX κρυπτογράφηση/αποκρυπτογράφηση
 - Πλήρες auto-negotiation
- Πολλαπλοί τρόποι λειτουργίας
 - Full- και half-duplex 100 Mbps
 - Full- και half-duplex 10 Mbps
 - Power-saving και power-down modes
- Εξαιρετικά παραμετροποιήσιμο
 - Προγραμματιζόμενη MAC address
 - Επιλογή για LED δραστηριότητας
 - Υποστήριξη για promiscuous mode
 - CRC error-rejection control
 - Διακοπές καθοριζόμενες από τον χρήστη
- Διαχείριση φυσικού μέσου
 - Αυτόματα διόρθωση MDI/MDI-X cross-over
 - Πλάτος μετάδοσης προγραμματιζόμενο από καταχωρητή
 - Αυτόματη διόρθωση πολικότητας και λήψη σήματος 10BASE-T

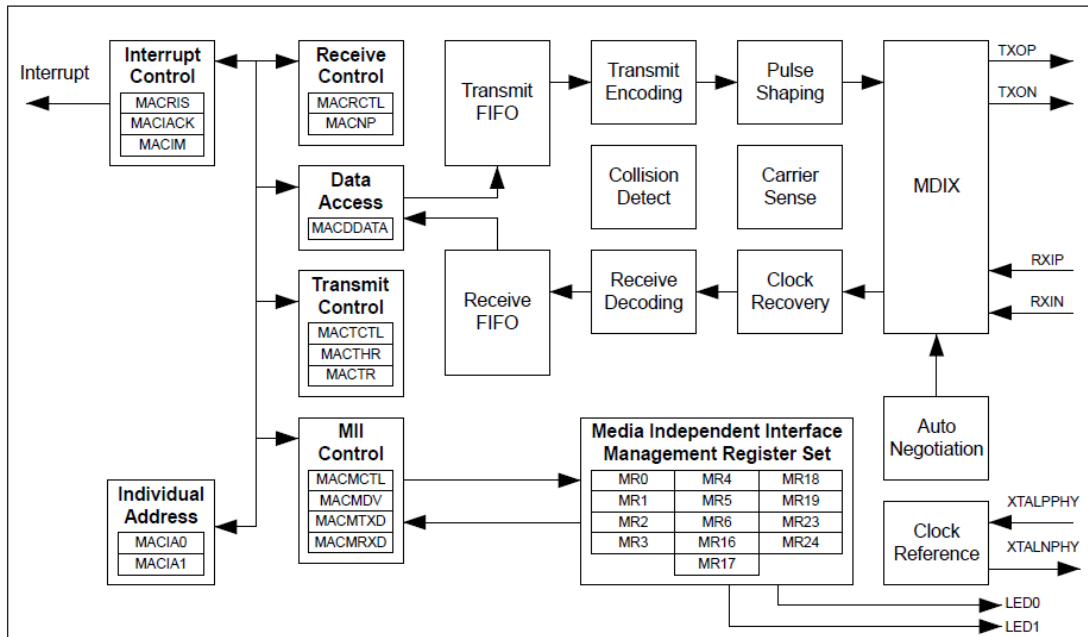


Όπως φαίνεται στο σχήμα, ο Ethernet Controller διαίρεται λειτουργικά σε δύο επίπεδα:

- Το επίπεδο Ελεγκτή Πρόσβασης Μέσου (Media Access Controller - MAC)
 - Αντιστοιχεί στο επίπεδο OSI 2
- Το επίπεδο Φυσικού Μέσου (Network Physical Layer - PHY)
 - Αντιστοιχεί στο επίπεδο OSI 1

Ο μικροελεγκτής αποκτά πρόσβαση στον Ethernet Controller μέσω του επιπέδου MAC. Το επίπεδο MAC προσφέρει επεξεργασία αποστολής και λήψης για πλαίσια Ethernet. Το

επίπεδο MAC παρέχει επίσης τη διεπαφή στο επίπεδο ΡΗΥ μέσω ενός ενσωματωμένου Media Independent Interface (MII). Το επίπεδο ΡΗΥ επικοινωνεί με τον δίαυλο Ethernet.



VII. Η οικογένεια μόντεμ XBEE

Η οικογένεια μόντεμ XBEE της Digi International διαθέτει αρκετά μοντέλα μικρών, οικονομικών ολοκληρωμένων συσκευών, με πλούσιες δυνατότητες σε αρκετά πρωτόκολλα δικτύωσης. Ο «Πανζεύκτης» έχει σχεδιαστεί έτσι, ώστε να δέχεται τόσο τα XBEE όσο και τα XBEE Pro μόντεμ. Το τρέχον λογισμικό έχει δοκιμαστεί με δύο XBee 802.15.4 OEM RF Module.



Εικόνα A.5.1: Το XBEE και το XBEE Pro

Διατίθενται τα ακόλουθα μοντέλα:

- ZigBee and Mesh Modules
 - XBee ZigBee
 - XBee® ZB ZigBee® RF Modules
 - XBee® ZB SMT
 - XBee DigiMesh
 - XBee® DigiMesh 2.4 RF Modules
 - XBee® DigiMesh 900 Mesh RF Modules
- Point-to-Multipoint RF Modules
 - XBee Wi-Fi
 - XBee Wi-Fi
 - XBee 802.15.4
 - XBee-PRO® 802.15.4 OEM RF Modules
 - XBee 868 MHz
 - XBee-PRO® 868 OEM RF Modules
 - XBee 900 MHz
 - XBee-PRO® 900 RF Modules
 - XBee-PRO® XSC
 - XTend® OEM RF Modules

Περισσότερες πληροφορίες μπορείτε να βρείτε στο [4].

Το υλικολογισμικό(firmware)

Η διασύνδεση Ethernet

Ο μικροελεγκτής LM3S6965, όπως και πολλοί μικροελεγκτές της οικογένειας stellaris® περιλαμβάνει έναν ελεγκτή Ethernet, ο οποίος αποτελείται από έναν πλήρως ολοκληρωμένο ελεγκτή πρόσβασης μέσου(Media Access Controller - MAC) και υποσύστημα φυσικής διασύνδεσης (PHY) με το δίκτυο. Ο ελεγκτής συμμορφώνεται με τις προδιαγραφές IEEE 802.3 και υποστηρίζει πλήρως τα πρότυπα 10BASE-T και 100BASE-TX πρότυπα.

Η στοίβα Lightweight IP (lwIP) είναι μια υλοποίηση ανοικτού κώδικα της στοίβας TCP/IP που αναπτύχθηκε με στόχο τη μείωση της χρήσης των πόρων, διατηρώντας παράλληλα μία στοίβα TCP / IP πλήρους κλίμακας. Το lwIP επιτρέπει στα ενσωματωμένα συστήματα να συνδέονται σε ένα τοπικό intranet ή στο Internet. Η στοίβα lwIP έχει προσαρμοστεί για την οικογένεια stellaris® των μικροελεγκτών.

Η στοίβα TCP/IP lwIP

Η στοίβα lwIP αναπτύχθηκε αρχικά από τον Adam Dunkels του Networked Embedded Systems group στο Swedish Institute of Computer Science. Πλέον αναπτύσσεται δυναμικά από μια παγκόσμια ομάδα με επικεφαλής τον Leon Woestenberg και χρησιμοποιείται σε πολλά εμπορικά προϊόντα. Γραμμένο σε γλώσσα προγραμματισμού C, το lwIP είναι μια στοίβα TCP/IP ανοικτού κώδικα, που αναπτύχθηκε για ενσωματωμένα συστήματα με έμφαση στην μείωση της χρήσης των πόρων. Το lwIP μπορεί να εκτελεστεί με ή χωρίς υποκείμενο λειτουργικό σύστημα. Το τυπικό μέγεθος του κώδικα που παράγεται είναι της τάξης των 25 έως 40 kilobyte ενώ οι απαιτήσεις σε RAM είναι περίπου 15 έως μερικές δεκάδες kilobyte. Μια βασική επισκόπηση της στοίβας lwIP επιχειρείται παρακάτω. Μπορείτε να βρείτε περισσότερες πληροφορίες σχετικά με το lwIP στις ιστοσελίδες:

<http://www.sics.se/adam/lwip> και <http://savannah.nongnu.org/projects/lwip>

Χαρακτηριστικά

Η στοίβα TCP/IP lwIP έχει τα ακόλουθα χαρακτηριστικά:

Internet Protocol (IP), συμπεριλαμβανομένης της προώθησης πακέτων σε πολλαπλές διεπαφές δικτύου.

Διαδικτυακό πρωτόκολλο ελέγχου μηνυμάτων (ICMP) για τη συντήρηση του δικτύου και τον εντοπισμό σφαλμάτων.

User Datagram Protocol (UDP), συμπεριλαμβανομένων των πειραματικών επεκτάσεων UDP-lite.

Transmission Control Protocol (TCP) με έλεγχο της κυκλοφοριακής συμφόρησης, RTT εκτιμήσεις, και γρήγορη ανάκτηση / μετάδοση.

Dynamic Host Configuration Protocol (DHCP)

Point-to-Point Protocol (PPP)

Address Resolution Protocol (ARP) για Ethernet

AutoIP αυτόματη link-local διαμόρφωση IP

Εξειδικευμένο raw API για βελτιωμένη απόδοση

Προαιρετικό Berkeley-like socket API

Υποστηρίζει πολλαπλές διεπαφές δικτύου και συνδέσεις

Υλοποίηση

Η στοίβα lwIP ασχολείται κυρίως με το πρωτόκολλο TCP / IP, αλλά επίσης παρέχει πρόσθετη υποστήριξη με ένα επίπεδο εξομοίωσης λειτουργικού συστήματος, προσωρινή μνήμη και υποσυστήματα διαχείρισης της μνήμης, συναρτήσεις διεπαφής με το δίκτυο και συναρτήσεις για τον υπολογισμό αθροίσματος ελέγχου (checksum) Διαδικτύου. Τα ανώτερου στρώματος πρωτόκολλα θεωρούνται μέρος της εφαρμογής. Τα πρωτόκολλα χαμηλότερων επιπέδων ελέγχονται από τον οδηγό του υλικού δικτύωσης.

Η στοίβα πρωτοκόλλων TCP/IP διατάσσεται σε πολλαπλά επίπεδα, όπου κάθε επίπεδο έχει μια συγκεκριμένη λειτουργία. Αυτός ο πολυεπίπεδος σχεδιασμός του πρωτοκόλλου έχει χρησιμεύσει ως οδηγός για την υλοποίηση της στοίβας lwIP. Κάθε πρωτόκολλο που υλοποιείται έχει τη δική του μονάδα με σημεία εισόδου στο πρωτόκολλο να παρέχονται μέσω κλήσης συναρτήσεων. Γίνονται πάντως ορισμένες παραβιάσεις ώστε να επιτυγχάνεται βελτίωση στις επιδόσεις σε όρους ταχύτητα επεξεργασίας και χρήσης της μνήμης. Για παράδειγμα, κατά τον υπολογισμό του αθροίσματος ελέγχου και την αποπολύπλεξη ενός εισερχόμενου TCP segment, η μονάδα TCP πρέπει να γνωρίζει την πηγή και τις IP διευθύνσεις προορισμού. Αντί η μονάδα IP να περνά αυτές τις διευθύνσεις στην μονάδα TCP μέσα από μια κλήση συνάρτησης, η μονάδα TCP γνωρίζει τη δομή του IP header και, ως εκ τούτου, μπορεί να εξάγει αυτές τις πληροφορίες η ίδια.

Μερικές υλοποιήσεις TCP/IP χωρίζουν τα πρωτόκολλα σε αυτόνομες διαδικασίες. Παρόλο που το μοντέλο αυτό έχει πλεονεκτήματα, το μειονέκτημα είναι η πρόσθετη επιβάρυνση κατά την εναλλαγή περιεχομένου (context switching). Η υλοποίηση lwIP χρησιμοποιεί ένα μοντέλο επεξεργασίας όπου όλα τα πρωτόκολλα εδράζουν σε μια ενιαία διαδικασία και διαχωρίζονται από τον πυρήνα του λειτουργικού συστήματος. Προγράμματα εφαρμογών μπορούν να βρίσκονται στη διαδικασία lwIP ή να είναι σε χωριστές διεργασίες. Έχοντας υλοποιήσει το lwIP εκτός του πυρήνα του λειτουργικού συστήματος, δύναται η δυνατότητα στη στοίβα lwIP να είναι φορητή σε λειτουργικά συστήματα ή να χρησιμοποιηθεί χωρίς λειτουργικό σύστημα.

Για να είναι το lwIP φορητό, δε χρησιμοποιούνται απευθείας από τον κώδικα συναρτήσεις ή δομές δεδομένων του λειτουργικού συστήματος. Αντ' αυτού, παρέχεται ένα επίπεδο εξομοίωσης λειτουργικού συστήματος, παρέχοντας μια ενιαία διεπαφή στις υπηρεσίες του λειτουργικού συστήματος όπως χρονόμετρα, συγχρονισμός διαδικασιών και μηχανισμοί διάδοσης μηνυμάτων. Το επίπεδο εξομοίωσης λειτουργικού συστήματος παρέχει λειτουργικότητα χρονισμού που χρησιμοποιείται από το TCP, συγχρονισμό διαδικασιών μέσω σηματοφορέων και διάδοση μηνυμάτων μέσω ενός απλού μηχανισμού που ονομάζεται "γραμματοκιβώτια" (mailboxes).

Το σύστημα διαχείρισης της μνήμης και της προσωρινής αποθήκευσης σε ένα σύστημα επικοινωνίας πρέπει να είναι προετοιμασμένο για λήψη buffer διαφόρων μεγεθών. Το lwIP χρησιμοποιεί buffer πακέτων που ονομάζονται rbufs. Η δομή rbuf επιτρέπει την κατανομή δυναμικής μνήμης για τα πακέτα όπως και την ανάθεση πακέτων στη στατική μνήμη. Ο διαχειριστής μνήμης που υποστηρίζει το σύστημα rbuf χειρίζεται αναθέσεις και απελευθερώσεις συνεκτικών τμημάτων μνήμης. Ο διαχειριστής μνήμης χρησιμοποιεί ένα ειδικό τμήμα του συνολικού συστήματος μνήμης, αποτρέποντας το σύστημα δικτύωσης από τη χρήση όλης της διαθέσιμης μνήμης.

Η στοίβα lwIP παρέχει μια δομή δεδομένων διεπαφής δικτύου που επιτρέπει στις διασυνδέσεις του δικτύου να καταχωρούνται σε μια συνδεδεμένη λίστα. Η δομή δεδομένων παρέχει έναν δείκτη στην επόμενη δομή διεπαφής δικτύου, το όνομα της διεπαφής και τη διεύθυνση IP. Υπάρχουν επίσης δύο δείκτες συναρτήσεων που παρέχονται στη δομή των δεδομένων: ο ένας καταδεικνύει μια λειτουργία για την επεξεργασία των εισερχόμενων δεδομένων και ο άλλος τον οδηγό συσκευής συσκευής που χρησιμοποιείται για τη μετάδοση δεδομένων στο φυσικό δίκτυο.

Διεπαφή προγράμματος εφαρμογής (API)

Η lwIP API μεγιστοποιεί την αποτελεσματικότητα αξιοποιώντας τη γνώση της εσωτερικής δομής του lwIP. Το API δεν απαιτεί τα δεδομένα να αντιγραφούν μεταξύ του προγράμματος και τη στοίβα TCP/IP, καθώς το πρόγραμμα εφαρμογής μπορεί να χειριστεί τα εσωτερικά δεδομένα άμεσα. Δεδομένου ότι το Berkeley (BSD) socket API είναι ευρέως κατανοητό, το lwIP παρέχει λειτουργίες BSD socket που έχουν ξαναγραφεί με τη χρήση του lwIP API.

Λόγω του μοντέλου επεξεργασίας της στοίβας lwIP, η υλοποίηση του API χωρίζεται σε δύο μέρη. Το πρώτο μέρος είναι μια βιβλιοθήκη που συνδέεται στο πρόγραμμα εφαρμογής και το δεύτερο μέρος υλοποιείται στη διεργασία του TCP/IP. Οι μηχανισμοί επικοινωνίας μεταξύ διεργασιών (InterProcess Communications - IPC) παρέχονται από το επίπεδο εξομίωσης λειτουργικού συστήματος.

Η Βιβλιοθήκη Οδηγών Περιφερειακών για Stellaris (Stellaris Peripheral Driver Library)

Η Stellaris Peripheral Driver Library είναι ένα σύνολο προγραμμάτων οδήγησης για την πρόσβαση στα περιφερειακά των μικροελεγκτών της οικογένειας stellaris®.

Τα εργαλεία ανάπτυξης

Η ανάπτυξη του υλικολογισμικού μπορεί να γίνει με μία από τις ακόλουθες οικογένειες εργαλείων:

- Keil RealView Microcontroller Development Kit
 - Στον ιστότοπο της TI διατίθεται δωρεάν μια evaluation έκδοση
- IAR Embedded Workbench
 - Στον ιστότοπο της TI διατίθεται δωρεάν το IAR Embedded Workbench-KickStart edition, μια περιορισμένη στα 32 KB evaluation έκδοση του IAR Embedded Workbench.
- CodeSourcery SourceryG++ GNU tool chain
 - Στον ιστότοπο της TI διατίθεται μια evaluation έκδοση 30 ημερών.
- Code Red Technologies tools
 - Στον ιστότοπο της TI διατίθεται μια evaluation έκδοση 90 ημερών.
- Code Composer Studio (CCStudio) IDE
 - Στον ιστότοπο της TI διατίθεται μια πλήρως λειτουργική έκδοση του Code Composer Studio (CCStudio) IDE που όμως θα λειτουργεί μόνο για ένα συγκεκριμένο Evaluation Board.

Επίσης, χρειάζεται και το EK-LM3S6965 Firmware Development Package, το οποίο διατίθεται δωρεάν από την Texas Instruments και περιλαμβάνει:

- Τη βιβλιοθήκη StellarisWare Driver Library
 - Είναι ένα δωρεάν σετ λειτουργιών για τον έλεγχο των περιφερειακών που υπάρχουν στην οικογένεια μικροελεγκτών ARM Cortex-M3 Stellaris.
- Τον Boot Loader
- Τη βιβλιοθήκη Graphics Library
- Τη βιβλιοθήκη USB Library

Ως βάση για την ανάπτυξη του υλικολογισμικού, χρησιμοποιήθηκε το παράδειγμα enet_io της βιβλιοθήκης EK-LM3S6965 Firmware Development Package της Texas Instruments[7]. Το περιβάλλον ανάπτυξης επιλογής ήταν το Sourcery CodeBench Stellaris EABI της codeSourcery, καθώς χρησιμοποιεί τη συλλογή GNU Toolchain και στηρίζεται στο γνώριμο και αξιόπιστο περιβάλλον του eclipse.

Η εφαρμογή χρήστη

Η εφαρμογή είναι ένας διακομιστής HTTP (Dynamic HTTP server) που εκτελείται στη στοίβα lwIP. Κατά την εκκίνηση, ξεκινά η διαδικασία απόδοσης διεύθυνσης IP δυναμικά με χρήση του DHCP. Αν συμβεί DHCP timeout, χρησιμοποιείται μια προκαθορισμένη διεύθυνση IP. Ο κύριος βρόχος ελέγχου αναμένει εισερχόμενα πακέτα, τα επεξεργάζεται με χρήση της στοίβας lwIP και του HTTP server και μεταδίδει πακέτα πίσω στο δίκτυο, όπως απαιτείται. Το παράδειγμα εφαρμόζει επίσης έναν περιοδικό χρονιστή ο οποίος χρησιμοποιείται για το TCP, το ARP, και το DHCP. Χρησιμοποιείται ένα προαποθηκευμένο εντός του μικροελεγκτή σύνολο σελίδων.

Ek-lm3s6965 – enet_io

- **Includes**

Σε αυτόν τον φάκελο αναφέρονται οι φάκελοι και τα μεμονωμένα αρχεία κεφαλίδων που χρησιμοποιούνται από το configuration.

- **Boards**

- **ek-lm3s6965**

- **drivers**

- **rit128x96x4.c**

- **rit128x96x4.h**

Αρχεία βιβλιοθήκης για υποστήριξη του OLED display

- **enet_io**

- **fs**

Φάκελος-πηγή, που περιέχει όλα τα αρχεία τα οποία μετά από μετατροπή με τη βοήθεια της εντολής:

```
"C:\StellarisWare\tools\bin\makefsfile -i fs -o io_fsdata.h -r -h", παράγουν το αρχείο io_fsdata.h (βλ. παρακάτω)
```

- **cgifuncs.c**

Εκτελεί αναζήτηση στη λίστα παραμέτρων που έχει περάσει σε έναν CGI handler και επιστρέφει τον δείκτη της παραμέτρου αυτής μέσα σε αυτή τη λίστα.

- **cgifuncs.h**

Το αρχείο κεφαλίδας για το cgifuncs.c

- **enet_io**

Το βασικό αρχείο της εφαρμογής. Περιέχει τη main, όπου γίνεται η αρχικοποίηση του συστήματος (όπως το lwIP, τα περιφερειακά και οι διακοπές).

- **io_fsdata.h**

Παράγεται αυτόματα με τη βοήθεια του makefsfile, όπως αναφέρθηκε παραπάνω. Περιέχει σε μορφή πινάκων byte όλα τα αρχεία του φακέλου fs, ώστε να είναι εφικτή η ανάκτησή τους από το lwIP.

- **io.c**

Σε αυτό το αρχείο περιέχονται όλες οι ρουτίνες που εκτελούν τις λειτουργίες που διαβιβάζονται μέσω της διεπαφής GUI αλλά και οι μετρήσεις ή άλλες εργασίες για την προετοιμασία των αποτελεσμάτων.

- **io.h**

Το αρχείο κεφαλίδας για το io.c

- **lmi_fs.c**

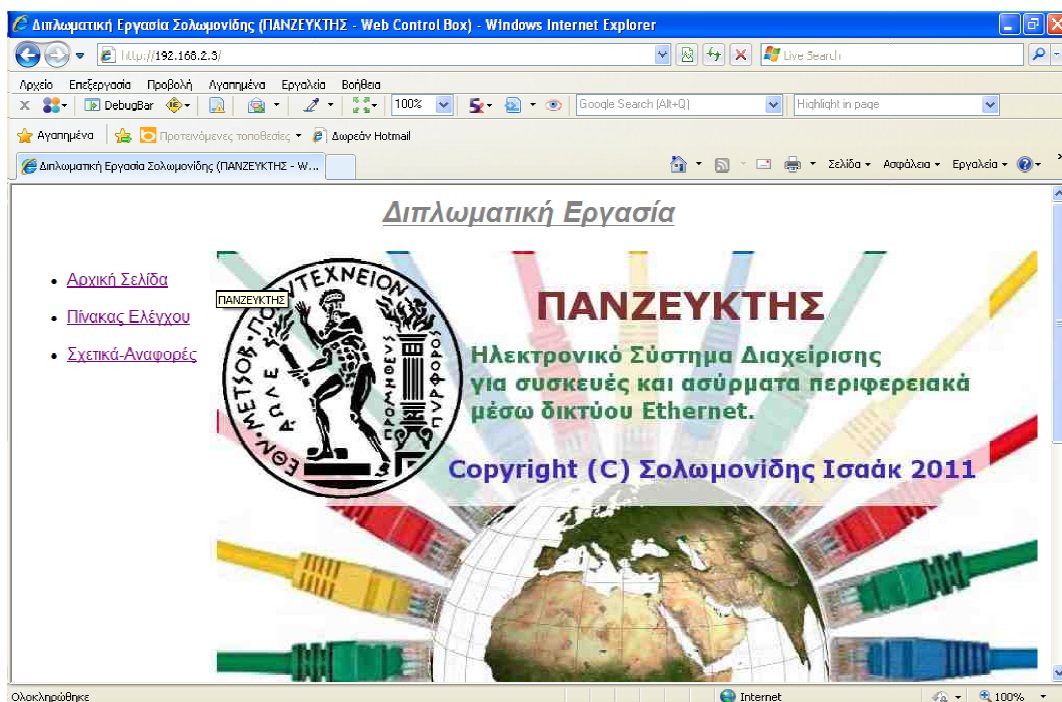
Εδώ γίνεται η επεξεργασία για το σύστημα αρχείων: ανάκτηση των αρχείων που ζητούν ο χρήστης μέσω του περιηγητή ιστού ή η εφαρμογή μέσω AJAX, ενώ ταυτόχρονα γίνεται και εκτέλεση κατάλληλων εντολών

και κλήση των κατάλληλων ρουτινών για την εκτέλεση των κατάλληλων λειτουργιών.

- **lwipopts.h**
Αρχείο διαμόρφωσης (παραμετροποίησης) του lwIP
- **enet_io_sourcerygxx.ld**
scatter file (αρχείο διαμόρφωσης μνήμης) για το Sourcery G++
- **readme.txt**
περιγραφή-οδηγίες της εφαρμογής
- **Release**
Περιέχει τα αρχεία που παράγονται με τη μεταγλώττιση(compilation) χωρίς πληροφορίες αποσφαλμάτωσης (debug)
- **third_party**
 - **lwip-1.3.2**
 - **apps**
 - **http_server_raw**
 - **fs.h**
Ορίζει την επεξεργασία για το σύστημα αρχείων ώστε να γίνεται από τις ρουτίνες του lmi_fs.c
 - **fsdata.h**
Ορίζει τη δομή fsdata_file
 - **httpd.c**
Υλοποιείται η λειτουργικότητα του δυναμικού http server
 - **httpd.h**
Το αρχείο κεφαλίδας για το httpd.c
- **utils**
 - **locator.c**
Υλοποιεί έναν εξυπηρετητή εύρεσης συσκευής με τη χρήση UDP στο lwIP
 - **lwiplib.c**
Υλοποιεί το Στρώμα Αφαιρετικότητας(Abstraction Layer) TCP/IP για το lwIP. Υλοποιούνται λειτουργίες του lwIP με τη βοήθεια του Stellaris.
 - **ustdlib.c**
Απλές, κοινές συναρτήσεις βιβλιοθήκης(Simple standard library functions)
 - **ek-lm3s6965 – enet_io.launch**
Αρχείο διαμόρφωσης(xml)

Τα αρχεία web του http server

1. index.htm



Πρόκειται για την πιο απλή σελίδα της εφαρμογής. Είναι στατική, με βασικά πληροφοριακά στοιχεία και συνδέσμους προς τις άλλες σελίδες. Μπορεί να επεκταθεί για να γίνει σελίδα ελέγχου πρόσβασης (login). Σε μια τέτοια περίπτωση, ο μικροελεγκτής δε θα εξυπηρετεί άλλες σελίδες μέχρι να κάνει ο χρήστης login.

Διπλωματική Εργασία
ΠΑΝΖΕΥΚΤΗΣ: Ηλεκτρονικό Σύστημα Διαχείρισης για συσκευές και ασύρματα περιφερειακά μέσω Ethernet
 Copyright © Σολωμονίδης Ισαάκ 2011

- [Αρχική Σελίδα](#)
- [Πίνακας Ελέγχου](#)
- [Σχετικά-Αναφορές](#)

Mon May 2 21:15:39 UTC+0300 2011
 Αυτόματη ανανέωση κάθε 1 sec
 Α. Ενδεικτικό Κατάστασης (Status) και επιλογέας (RS232 IrDA) (1 GPIO bit):
 ON

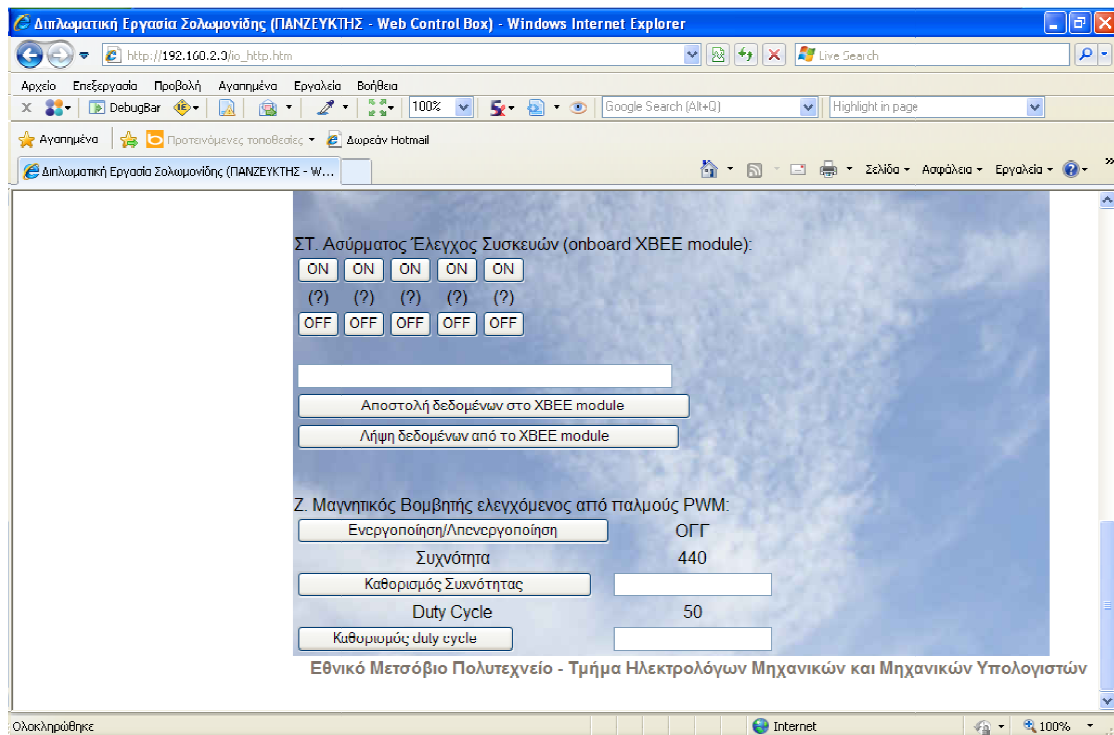
 Β. Αναλογικές Είσοδοι (4 είσοδοι Αναλογικό-σε-Ψηφιακό):
 0,1,1,552

Γ. Είσοδοι Εποπτείας Φορτίου (4 οπτικά απομονωμένες είσοδοι 220V)
 0,0,0,0

Δ. Έξοδοι Ελέγχου Φορτίου (4 έξοδοι ρελέ 230V/3A)

Ε. Σειριακή επικοινωνία (RS232 - IrDA):

ΣΤ. Ασύρματος Έλεγχος Συσκευών (onboard XBEE module):



Πρόκειται για τη βασική σελίδα της εφαρμογής. Με τη χρήση javascript αποστέλλονται αιτήματα στον http server ο οποίος είτε προκαλεί την εκτέλεση συγκεκριμένων λειτουργιών του συστήματος, είτε μετά την εκτέλεση των κατάλληλων εσωτερικών λειτουργιών ενημερώνει δυναμικά τα στοιχεία ελέγχου της σελίδας με τη χρήση AJAX ή εκτελεί και τις δύο αυτές εργασίες διαδοχικά.

Ημερομηνία / Ώρα

Στο επάνω μέρος του πίνακα ελέγχου διακρίνουμε μια ένδειξη της τοπικής ημερομηνίας και ώρας. Πρόκειται για την ημερομηνία και ώρα του συστήματος του χρήστη. Προτείνεται η επέκταση της εφαρμογής ώστε να υποστηρίζεται ο χρονοπρογραμματισμός εργασιών (π.χ. αυτόματο πότισμα, τυχαίο άναμμα και σβήσιμο φώτων κατά την απουσία από το σπίτι). Σε αυτήν την περίπτωση θα είναι καλό να προβλεφθεί μπαταρία υποστήριξης του Ρολογιού Πραγματικού Χρόνου (RTC). Τότε, η ημερομηνία και ώρα του ελεγκτή μπορεί να αποστέλλεται περιοδικά μέσα από επαναλαμβανόμενα αιτήματα http, με την χρήση του AJAX.

A. Ενδεικτικό Κατάστασης (Status) και επιλογέας (RS232-IrDA) (1 GPIO bit)

Λειτουργία: Ενεργοποιεί και απενεργοποιεί το Ενδεικτικό Κατάστασης (Status) και επιλέγει σειριακή θύρα (OFF: IrDA, ON-default: RS-232)

Περιγραφή λειτουργίας: Το πλήκτρο “Εναλλαγή Κατάστασης Ενδεικτικού Κατάστασης (Status)” καλεί τη συνάρτηση `toggle_led()`, η οποία αιτείται τη σελίδα `/ledstate` και ενημερώνει τον πίνακα ελέγχου με τα αποτελέσματα της σελίδας `/cgi-bin/toggle_led`.

```
function toggle_led() {
    var req = false;
    var led = false;

    function ledComplete() {
        if (led.readyState == 4) {
            if (led.status == 200) {
                document.getElementById("ledstate").innerHTML = "<div>" + led.responseText + "</div>";
            }
        }
    }

    if (window.XMLHttpRequest) {
        req = new XMLHttpRequest();
        led = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
        req = new ActiveXObject("Microsoft.XMLHTTP");
        led = new ActiveXObject("Microsoft.XMLHTTP");
    }

    if (req) {
        req.open("GET", "/cgi-bin/toggle_led?id" + Math.random(), true);
        req.send(null);
    }

    if (led) {
        led.open("GET", "/ledstate?id=" + Math.random(), true);
        led.onreadystatechange = ledComplete;
        led.send(null);
    }
}
```

Ο http server εκτελεί το εξής τμήμα κώδικα:

```
//
```

```

// Process request to toggle STATUS LED
//
if(strncmp(name, "/cgi-bin/toggle_led", 19) == 0)
{
    //
    // Toggle the STATUS LED
    //
    io_set_led(!io_is_led_on());

    //
    // Setup the file structure to return whatever.
    //
    ptFile->data = NULL;
    ptFile->len = 0;
    ptFile->index = 0;
    ptFile->pextension = NULL;

    //
    // Return the file system pointer.
    //
    return(ptFile);
}

```

Καλείται η συνάρτηση **io_set_led**, η οποία τοποθετεί κατάλληλα το pin ελέγχου, έτσι ώστε όταν ο χρήστης επιλέγει RS232 απενεργοποιείται ο πομποδέκτης του IrDA και ενεργοποιείται αυτός της εξωτερικής θύρας COM και αντίστροφα. Παράλληλα, ελέγχει την ενεργοποίηση/απενεργοποίηση της αυτόματης διαμόρφωσης IrDA (SIR).

```

void
io_set_led(tBoolean bOn)
{
    //
    // Turn the LED on or off as requested.
    //
    GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_0, bOn ? GPIO_PIN_0 : 0);

    if (!bOn)
    {
        // Configure the UART2 for IrDA operation.
        UARTEnableSIR(UART2_BASE, 0);
    }
    else
    {
        // Configure the UART2 for RS232 operation.
        UARTDisableSIR(UART2_BASE);
    }
}

```

B. Αναλογικές Είσοδοι (4 είσοδοι Αναλογικό-σε-Ψηφιακό)

Λειτουργία: Εκτελεί μέτρηση των αναλογικών εισόδων και επιστρέφει τις αριθμητικές τους τιμές, χωρισμένες με κόμματα.

Περιγραφή Λειτουργίας: Το πλήκτρο “Λήψη Αναλογικών Μετρήσεων” καλεί τη συνάρτηση `adcGet()`, η οποία αιτείται τη σελίδα `"/adcget"` και ενημερώνει τον πίνακα ελέγχου με τα αποτελέσματά της.

```
function adcGet() {
    var adcMeas = false;
    function adcMeasComplete() {
        if (adcMeas.readyState == 4) {
            if (adcMeas.status == 200) {
                document.getElementById("adc").innerHTML = "<div>"
+ adcMeas.responseText + "</div>";
            }
        }
    }
    if (window.XMLHttpRequest) {
        adcMeas = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
        adcMeas = new ActiveXObject("Microsoft.XMLHTTP");
    }
    if (adcMeas) {
        adcMeas.open("GET", "/adcget?id" + Math.random(), true);
        adcMeas.onreadystatechange = adcMeasComplete;
        adcMeas.send(null);
    }
}
```

Ο http server εκτελεί το εξής τμήμα κώδικα:

```
//
// Request the ADC Data
//

if(strncmp(name, "/adcget?id", 10) == 0)
{
    unsigned long adcMeasurements[4];
    static char pcBuf[32];

    io_get_adcdata(&adcMeasurements);
    sprintf(pcBuf, "%d, %d, %d, %d", adcMeasurements[0],
        adcMeasurements[1], adcMeasurements[2], adc-
Measurements[3] );

    ptFile->data = pcBuf;
    ptFile->len = strlen(pcBuf);
}
```

```

    ptFile->index = ptFile->len;
    ptFile->pextension = NULL;
    return(ptFile);
}

```

καλείται η συνάρτηση `io_get_adcdata()`:

```

void
io_get_adcdata(unsigned long* adcMeasurements)
{
    //
    // Disable sample sequence 1
    //
    ADCSequenceDisable(ADC_BASE, 1);
    //
    //
    ADCComparatorReset(ADC_BASE, 1, true, true);
    // Enable sample sequence 1
    //
    ADCSequenceEnable(ADC_BASE, 1);

    // Trigger the sample sequence.
    //
    ADCProcessorTrigger(ADC_BASE, 1);
    //
    // Wait until the sample sequence has completed.
    //
    while(!ADCIntStatus(ADC_BASE, 1, true))
    {
    }
    //
    // Read the value from the ADC.
    //
    //}
    if (ADCSequenceDataGet(ADC_BASE, 1, adcMeasurements) != 4)
    {
        adcMeasurements[0] = 10000;
        adcMeasurements[1] = 10000;
        adcMeasurements[2] = 10000;
        adcMeasurements[3] = 10000;
    }
}

```

Η μέτρηση των αναλογικών εισόδων έχει εξαρχής προγραμματιστεί. Όλες οι μετρήσεις γίνονται με τη βοήθεια του sequencer 1:

```

//
// Enable the clock to the ADC module
//
SysCtlPeripheralEnable(SYSCTL_PERIPH_ADC);

//
// Disable all four sample sequences
//
ADCSequenceDisable(ADC_BASE, 0);
ADCSequenceDisable(ADC_BASE, 1);
ADCSequenceDisable(ADC_BASE, 2);
ADCSequenceDisable(ADC_BASE, 3);

//
// Configure sample sequence 1: processor trigger, priority = 0
//
ADCSequenceConfigure(ADC_BASE, 1, ADC_TRIGGER_PROCESSOR, 0);

//

```



```

// Configure sample sequence 1 steps 0, 1, 2 and 3
//
ADCSequenceStepConfigure(ADC_BASE, 1, 0, ADC_CTL_CH0);
ADCSequenceStepConfigure(ADC_BASE, 1, 1, ADC_CTL_CH1);
ADCSequenceStepConfigure(ADC_BASE, 1, 2, ADC_CTL_CH2);
ADCSequenceStepConfigure(ADC_BASE, 1, 3, ADC_CTL_CH3 | ADC_CTL_IE
| ADC_CTL_END);

//
// Enable 64x hardware averaging
//
ADCHardwareOversampleConfigure(ADC_BASE, 64);

//
// Enable ADC Interrupt
//
ADCIntEnable(ADC_BASE, 1);

//
// Enable sample sequence 1
//
ADCSequenceEnable(ADC_BASE, 1);

```

Έχει χρησιμοποιηθεί και η ενσωματωμένη δυνατότητα για hardware over-sampling, ρυθμισμένη στα 64 δείγματα.

Γ. Είσοδοι Εποπτείας Φορτίου (4 οπτικά απομονωμένες εισοδοι 220V)

Λειτουργία: Εκτελεί ανάλυση των οπτικά απομονωμένων ψηφιακών εισόδων εποπτείας φορτίου, για την ανίχνευση σήματος 220V/50Hz.

Περιγραφή λειτουργίας: Έχει προγραμματιστεί ο χρονιστής TIMER2 σε επαναλαμβανόμενη λειτουργία μέτρησης προς τα κάτω με περίοδο 1ms.

```
//
// Enable the TIMER2 peripheral
//
SysCtlPeripheralEnable(SYSCTL_PERIPH_TIMER2);

//
// Initialize Timer 2 to trigger an Interrupt once every 1 ms
//
TimerDisable(TIMER2_BASE, TIMER_A);
TimerConfigure(TIMER2_BASE, TIMER_CFG_32_BIT_PER);
TimerLoadSet(TIMER2_BASE, TIMER_A, SysCtlClockGet() / 1000);
IntEnable(INT_TIMER2A);
TimerIntEnable(TIMER2_BASE, TIMER_TIMA_TIMEOUT);
TimerEnable(TIMER2_BASE, TIMER_A);
```

Επίσης, στο αρχείο configuration της εφαρμογής (.ld) έχει προστεθεί:

```
EXTERN(powerlineInputGetSample)
__cs3_isr_timer2a = powerlineInputGetSample;
```

για τον καθορισμό της συνάρτησης που εκτελείται κάθε φορά που ο χρονιστής TIMER2 φθάνει στο μηδέν (κάθε 1ms). Η συνάρτηση αυτή είναι:

```
#define OPTOISOLATED_INPUTS_TOTAL_SAMPLES (400)
void
powerlineInputGetSample(void)
{
    //
    // Clear the timer interrupt.
    //
    TimerIntClear(TIMER2_BASE, TIMER_TIMA_TIMEOUT);

    if (pInputSampleIndex++ < OPTOISOLATED_INPUTS_TOTAL_SAMPLES)
    {
        pInputSample[0] += (GPIOPinRead(ISOL_INPUT1_GPIO_PORT_BASE,
ISOL_INPUT1_GPIO_PIN)==0)?1:0;
        pInputSample[1] += (GPIOPinRead(ISOL_INPUT2_GPIO_PORT_BASE,
ISOL_INPUT2_GPIO_PIN)==0)?1:0;
        pInputSample[2] += (GPIOPinRead(ISOL_INPUT3_GPIO_PORT_BASE,
ISOL_INPUT3_GPIO_PIN)==0)?1:0;
```

```

        plInputSample[3] += (GPIOPinRead(ISOL_INPUT4_GPIO_PORT_BASE,
ISOL_INPUT4_GPIO_PIN)==0)?1:0;
    }
    else
    {
        plInputSampleResult[0]=plInputSample[0];
        plInputSampleResult[0]=plInputSample[1];
        plInputSampleResult[0]=plInputSample[2];
        plInputSampleResult[0]=plInputSample[3];
        plInputSampleIndex=0;
        plInputSample[0]=0;
        plInputSample[1]=0;
        plInputSample[2]=0;
        plInputSample[3]=0;
    }
}

```

Δηλαδή, συνεχώς εκτελείται ένας κύκλος λήψης 400 δειγμάτων με περίοδο δειγματοληψίας 1ms. Δηλαδή, λαμβάνονται δείγματα σε χρόνο 20 περιόδων της τάσης δικτύου ($1/50\text{Hz} * 20 = 400\text{ms}$). Όταν λαμβάνονται τα 400 δείγματα, γίνεται αποθήκευση στο διάνυσμα `plInputSampleResult`, απ'όπου ασύγχρονα γίνεται ανάγνωση των αποτελεσμάτων από τις άλλες συναρτήσεις. Δεν έχει προβλεφθεί μηχανισμός αποκλειστικής εγγραφής/ανάγνωσης(locking) για το διάνυσμα των αποτελεσμάτων. Αν η κρισιμότητα της εφαρμογής το απαιτεί, μπορεί εύκολα να υλοποιηθεί.

Το πλήκτρο “Λήψη Κατάστασης Εισόδων Εποπτείας Φορτίου” καλεί τη συνάρτηση `digInGet()`, η οποία αιτείται τη σελίδα `"/diginget"` και ενημερώνει τον πίνακα ελέγχου με τα αποτελέσματά της.

```

function digInGet() {
    var digInRet = false;
    function digInRetrievalComplete() {
        if (digInRet.readyState == 4) {
            if (digInRet.status == 200) {
                document.getElementById("digIn").innerHTML =
"<div>" + digInRet.responseText + "</div>";
            }
        }
    }
    if (window.XMLHttpRequest) {
        digInRet = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
        digInRet = new ActiveXObject("Microsoft.XMLHTTP");
    }
    if (digInRet) {
        digInRet.open("GET", "/diginget?id" + Math.random(), true);
        digInRet.onreadystatechange = digInRetrievalComplete;
    }
}

```

```

        digInRet.send(null);
    }
}

```

Ο http server εκτελεί το εξής τμήμα κώδικα:

```

#define OPTOISOLATED_INPUTS_SAMPLES_QUOTA (160)
//
// Request the optoisolated inputs Data
//

if(strncmp(name, "/diginget?id", 12) == 0)
{
    unsigned char optoisolatedInputsData[8];
    io_get_optoisolated_input_data(&optoisolatedInputsData);
    static char pcBuf[32];

    usprintf(pcBuf, "%s", optoisolatedInputsData);

    ptFile->data = pcBuf;
    ptFile->len = strlen(pcBuf);
    ptFile->index = ptFile->len;
    ptFile->pextension = NULL;
    return(ptFile);
}

```

Καλείται η συνάρτηση io_get_optoisolated_input_data:

```

void
io_get_optoisolated_input_data(unsigned char* optoisolatedInputsData)
{
    optoisolatedInputsData[0] = (plInputSampleResult[0] >= OPTOISO-
LATED_INPUTS_SAMPLES_QUOTA) ? '1' : '0';
    optoisolatedInputsData[1] = ',';
    optoisolatedInputsData[2] = (plInputSampleResult[1] >= OPTOISO-
LATED_INPUTS_SAMPLES_QUOTA) ? '1' : '0';
    optoisolatedInputsData[3] = ',';
    optoisolatedInputsData[4] = (plInputSampleResult[2] >= OPTOISO-
LATED_INPUTS_SAMPLES_QUOTA) ? '1' : '0';
    optoisolatedInputsData[5] = ',';
    optoisolatedInputsData[6] = (plInputSampleResult[3] >= OPTOISO-
LATED_INPUTS_SAMPLES_QUOTA) ? '1' : '0';
    optoisolatedInputsData[7] = 0;
}

```

Επειδή, όπως έχουμε υπολογίσει, κατ'ελάχιστο το 40% των δειγμάτων πρέπει να είναι ενεργά (low), το κατώφλι απόφασης είναι στα 160 δείγματα. Τα αποτελέσματα (ενεργή/ανενεργή είσοδος) αποθηκεύονται στο διάνυσμα optoisolatedInputsData.

Δ. Έξοδοι Ελέγχου Φορτίου (4 έξοδοι ρελέ 230V/3A)

Λειτουργία: Ελέγχει την ενεργοποίηση/απενεργοποίηση των τεσσάρων ρελέ ελέγχου φορτίου.

Περιγραφή λειτουργίας: Το πλήκτρο ON(OFF) κάθε εισόδου καλεί τη συνάρτηση

set_relay_state(μ,ν) με ορίσματα μ: αριθμός ρελέ {1,2,3,4} και ν: {0:OFF, 1:ON}

```
function set_relay_state(relayIndex, relayState) {  
    var req = false;  
    var relay = false;  
    function relayComplete() {  
        if (relay.readyState == 4) {  
            if (relay.status == 200) {  
                document.getElementById("relay" + relayIndex +  
"State").innerHTML = "<div>" + relay.responseText + "</div>";  
            }  
        }  
    }  
    if (window.XMLHttpRequest) {  
        req = new XMLHttpRequest();  
        relay = new XMLHttpRequest();  
    } else if (window.ActiveXObject) {  
        req = new ActiveXObject("Microsoft.XMLHTTP");  
        relay = new ActiveXObject("Microsoft.XMLHTTP");  
    }  
    if (req) {  
        req.open("GET", "/cgi-bin/set_relay?index=" + relayIndex  
            + "&state=" + relayState + Math.random(), true);  
        req.send(null);  
    }  
    if (relay) {  
        relay.open("GET", "/relaystate?index=" + relayIndex + ":"  
            + Math.random(), true);  
        relay.onreadystatechange = relayComplete;  
        relay.send(null);  
    }  
}
```

Γίνεται αίτηση για τη σελίδα "/cgi-bin/set_relay?index=μ&state=ν" για τον καθορισμό της κατάστασης του ρελέ και αμέσως μετά για τη σελίδα "/relaystate?index=μ" για την ενημέρωση του πίνακα ελέγχου με την πραγματική κατάσταση του ρελέ.

O http server εκτελεί το εξής τμήμα κώδικα:

```
//
// Process request to set RELAY STATE
//
if( (strcmp(name, "/cgi-bin/set_relay?index=", 25) == 0) &&
(strcmp(name+26, "&state=", 7) == 0) )
{
    //
    // Get Relay Index //
    unsigned short relayIndex = *(name+25)-'0';
    // Get Requested Relay State //
    tBoolean relayState = *(name+33)-'0';

    //
    // Set the RELAY state
    //
    if ( (relayIndex>=1) && (relayIndex <= 4) )
        io_set_relay(relayIndex, relayState);

    //
    // Setup the file structure to return whatever.
    //
    ptFile->data = NULL;
    ptFile->len = 0;
    ptFile->index = 0;
    ptFile->pextension = NULL;

    //
    // Return the file system pointer.
    //
    return(ptFile);
}

void
io_set_relay(unsigned char relayIndex, tBoolean bOn)
{
    //
    // Turn the RELAY on or off as requested.
    //
    switch(relayIndex)
    {
        case 1:
            GPIOWrite(RELAY1_EN_GPIO_PORT_BASE, RE-
LAY1_EN_GPIO_PIN, bOn ? RELAY1_EN_GPIO_PIN : 0);
            break;
        case 2:
            GPIOWrite(RELAY2_EN_GPIO_PORT_BASE, RE-
LAY2_EN_GPIO_PIN, bOn ? RELAY2_EN_GPIO_PIN : 0);
            break;
        case 3:
            GPIOWrite(RELAY3_EN_GPIO_PORT_BASE, RE-
LAY3_EN_GPIO_PIN, bOn ? RELAY3_EN_GPIO_PIN : 0);
            break;
        case 4:
            GPIOWrite(RELAY4_EN_GPIO_PORT_BASE, RE-
LAY4_EN_GPIO_PIN, bOn ? RELAY4_EN_GPIO_PIN : 0);
            break;
    }
}

//
// Request for RELAY State?
//
```

```

if(strncmp(name, "/relaystate?index=", 18) == 0)
{
    unsigned short relayIndex = *(name+18)-'0';
    static char pcBuf[4];

    //
    // Get the state of the RELAY
    //
    if ( (relayIndex>=1) && (relayIndex <= 4) )
        io_get_relaystate(relayIndex, pcBuf, 4);

    else usnprintf(pcBuf, 4, "inv");

    ptFile->data = pcBuf;
    ptFile->len = strlen(pcBuf);
    ptFile->index = ptFile->len;
    ptFile->pextension = NULL;
    return(ptFile);
}

void
io_get_relaystate(unsigned short relayIndex, char * pcBuf, int iBuf-
Len)
{
    //
    // Get the state of the RELAY
    //

    tBoolean relayIsOn = false;
    switch(relayIndex)
    {
        case 1:
            relayIsOn = GPIOPinRead(RELAY1_EN_GPIO_PORT_BASE, RE-
LAY1_EN_GPIO_PIN);
            break;
        case 2:
            relayIsOn = GPIOPinRead(RELAY2_EN_GPIO_PORT_BASE, RE-
LAY2_EN_GPIO_PIN);
            break;
        case 3:
            relayIsOn = GPIOPinRead(RELAY3_EN_GPIO_PORT_BASE, RE-
LAY3_EN_GPIO_PIN);
            break;
        case 4:
            relayIsOn = GPIOPinRead(RELAY4_EN_GPIO_PORT_BASE, RE-
LAY4_EN_GPIO_PIN);
            break;
    }

    if(relayIsOn)
    {
        usnprintf(pcBuf, iBufLen, "ON");
    }
    else
    {
        usnprintf(pcBuf, iBufLen, "OFF");
    }
}

```

ΠΡΟΣΟΧΗ: το εξωτερικό κύκλωμα θα πρέπει να ασφαλίζεται (2,5A/250VAC), καθώς σε περίπτωση βραχυκυκλώματος μπορεί να καταστραφεί το τυπωμένο κύκλωμα που συνδέει τα ρελέ ή/και τα ίδια τα ρελέ.

Ε. Σειριακή επικοινωνία (RS232 – IrDA)

Λειτουργία: Γίνεται αποστολή και λήψη δεδομένων προς και από το UART2.

Περιγραφή λειτουργίας αποστολής: Το πλήκτρο "Αποστολή δεδομένων στο UART" καλεί τη συνάρτηση UARTSendData:

```
function UARTSendData() {
    var req = false;
    var lenZeroPad = document.getElementById("UARTDataToSend").value.length + '';
    var dataToSend = document.getElementById("UARTDataToSend").value;
    while(lenZeroPad.length < 3)
        lenZeroPad= "0" + lenZeroPad;
    function UARTDataTransmissionComplete() {
        if (req.readyState == 4) {
            if (req.status == 200) {
                UARTReceiveData();
            }
        }
    }
    if (window.XMLHttpRequest)
        req = new XMLHttpRequest();
    else if (window.ActiveXObject)
        req = new ActiveXObject("Microsoft.XMLHTTP");
    if (req) {
        req.open("GET", "/cgi-bin/UARTSendData?length=" + lenZeroPad + "&data=" + dataToSend + Math.random(), true);
        req.onreadystatechange = UARTDataTransmissionComplete;
        req.send(null);
    }
}
```

Γίνεται αίτηση για τη σελίδα "/cgi-bin/UARTSendData" με προσαρτήματα το πλήθος των byte και τα ίδια τα byte προς αποστολή. Με τα δεδομένα που επιστρέφονται ενημερώνεται το πεδίο κειμένου, αντικαθιστώντας προηγούμενα δεδομένα αν υπάρχουν.

Ο http server εκτελεί το εξής τμήμα κώδικα:

```
//
// Process request to send UART2 (RS232 or IrDA)
//
if( strcmp(name, "/cgi-bin/UARTSendData?length=", 29) == 0)
{
```



```

        unsigned int dataCount = atoi(strndup(name+29, 3));
        for (i=0; i<dataCount; i++)
            UARTCharPut(UART2_BASE, name[38+i]);

//
// Setup the file structure to return whatever.
//
ptFile->data = NULL;
ptFile->len = 0;
ptFile->index = 0;
ptFile->pextension = NULL;

//
// Return the file system pointer.
//
return(ptFile);
}

```

Περιγραφή λειτουργίας λήψης: Το πλήκτρο "Λήψη δεδομένων από το UART" καλεί τη συνάρτηση UARTReceiveData:

```

function UARTReceiveData() {
    var UARTDataReception = false;
    function UARTDataReceptionComplete() {
        if (UARTDataReception.readyState == 4) {
            if (UARTDataReception.status == 200) {
                document.getElementById("UARTDataReceived").innerHTML =
"<div>" + UARTDataReception.responseText + "</div>";
            }
        }
    }
    if (window.XMLHttpRequest) {
        UARTDataReception = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
        UARTDataReception = new ActiveXObject("Microsoft.XMLHTTP");
    }
    if (UARTDataReception) {
        UARTDataReception.open("GET", "/UARTReceiveData" +
Math.random(), true);
        UARTDataReception.onreadystatechange = UARTDataReceptionCom-
plete;
        UARTDataReception.send(null);
    }
}

```

Γίνεται αίτηση για τη σελίδα "/UARTReceiveData". Με τα δεδομένα που επιστρέφονται ενημερώνεται το πεδίο κειμένου, αντικαθιστώντας προηγούμενα δεδομένα αν υπάρχουν.

O http server εκτελεί το εξής τμήμα κώδικα:

```
//  
// Process request to receive UART (RS232 or IrDA) data  
//  
if(strcmp(name, "/UARTReceiveData", 16) == 0)  
{  
    unsigned int i=0;  
    static char pcBuf[32];  
    while (UARTCharsAvail(UART2_BASE) && (i<31) )  
        pcBuf[i++] = UARTCharGet(UART2_BASE) & 0xFF;  
  
        if (i==0)  
            usnprintf(pcBuf, 32, "%s", "no data available");  
    else  
        pcBuf[i]=0;  
  
    ptFile->data = pcBuf;  
    ptFile->len = strlen(pcBuf);  
    ptFile->index = ptFile->len;  
    ptFile->pextension = NULL;  
    return(ptFile);  
}
```

ΣΤ. Ασύρματος Έλεγχος Συσκευών (onboard XBEE module)

α. Έλεγχος εξόδων γενικής χρήσης(GPIO) απομακρυσμένου modem

Λειτουργία: Γίνεται έλεγχος των GPIO του μόντεμ XBEE. Τα πλήκτρα "ON"("OFF") ενεργοποιούν(απενεργοποιούν) το αντίστοιχο GPIO pin ενός προκαθορισμένου (64bit address) end device.

Σημείωση: Για ασφάλεια, θα πρέπει ο χρήστης να προγραμματίσει τα pin που επιθυμεί να είναι διαθέσιμα προς έλεγχο, μέσω των παραμέτρων που αποθηκεύονται στη flash του μόντεμ προορισμού. Για τον σκοπό αυτό, μπορεί να χρησιμοποιηθεί το X-CTU.

Περιγραφή λειτουργίας: Τα πλήκτρα "ON"("OFF") καλούν τη συνάρτηση

set_XBEEOut_state(μ,ν) με ορίσματα μ: αριθμός εξόδου {1,2,3,4,5} και ν: {0:OFF, 1:ON}

```
function set_XBEEOut_state(XBEEOutIndex, XBEEOutState) {
    var req = false;
    var XBEEOut = false;

    function XBEEOutComplete() {
        if (XBEEOut.readyState == 4) {
            if (XBEEOut.status == 200) {
                document.getElementById("XBEEOut" + relayIndex +
                    "State").innerHTML = "<div>" + XBEEOut.responseText + "</div>";
            }
        }
    }

    if (window.XMLHttpRequest) {
        req = new XMLHttpRequest();
        XBEEOut = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
        req = new ActiveXObject("Microsoft.XMLHTTP");
        XBEEOut = new ActiveXObject("Microsoft.XMLHTTP");
    }

    if (req) {
        req.open("GET", "/cgi-bin/set_XBEEOut?index=" + XBEEOutIndex
            + "&state=" + XBEEOutState + Math.random(), true);
        req.send(null);
    }

    if (XBEEOut) {
        XBEEOut.open("GET", "/XBEEOutstate?index=" + XBEEOutIndex + ":"
            + Math.random(), true);
        XBEEOut.onreadystatechange = XBEEOutComplete;
        XBEEOut.send(null);
    }
}
```

```
}
```

Γίνεται αίτηση για τη σελίδα `"/cgi-bin/set_XBEEOut?index=μ&state=ν"` για τον καθορισμό της κατάστασης της εξόδου του μόντεμ και αμέσως μετά για τη σελίδα `"/XBEEOutstate?index=μ"` για την ενημέρωση του πίνακα ελέγχου με την πραγματική κατάσταση της εξόδου.

Ο http server εκτελεί το εξής τμήμα κώδικα:

```
//  
// Process request to set XBEEOut STATE  
//  
if( (strcmp(name, "/cgi-bin/set_XBEEOut?index=", 27) == 0) && (strcmp(name+28,  
"&state=", 7) == 0) )  
{  
    //  
    // Get XBEEOut Index      //  
    unsigned short XBEEOutIndex = *(name+27)-'0';  
    // Get Requested Relay State //  
    tBoolean XBEEOutState = *(name+35)-'0';  
  
    //  
    // Set the XBEEOut state  
    //  
    if ( (XBEEOutIndex>=1) && (XBEEOutIndex <= 5) )  
        io_set_XBEEOut(XBEEOutIndex, XBEEOutState);  
  
    //  
    // Setup the file structure to return whatever.  
    //  
    ptFile->data = NULL;  
    ptFile->len = 0;  
    ptFile->index = 0;  
    ptFile->pextension = NULL;  
  
    //  
    // Return the file system pointer.  
    //  
    return(ptFile);  
}
```

καλείται η συνάρτηση `io_set_XBEEOut`:

```
void  
io_set_XBEEOut(unsigned char XBEEOutIndex, tBoolean bOn)  
{  
    SysCtlDelay(20000000);  
    UARTCharPut(UART1_BASE, '+');  
    UARTCharPut(UART1_BASE, '+');  
    UARTCharPut(UART1_BASE, '+');  
    SysCtlDelay(20000000);  
    UARTCharPut(UART1_BASE, 'A');  
    UARTCharPut(UART1_BASE, 'T');  
    UARTCharPut(UART1_BASE, 'A');  
    UARTCharPut(UART1_BASE, 'P');  
}
```

```

UARTCharPut(UART1_BASE, '1');
UARTCharPut(UART1_BASE, 0x0D); //CR
SysCtlDelay(2000000);
UARTCharPut(UART1_BASE, 'A');
UARTCharPut(UART1_BASE, 'T');
UARTCharPut(UART1_BASE, 'C');
UARTCharPut(UART1_BASE, 'N');
UARTCharPut(UART1_BASE, 0x0D); //CR

//
// Turn the XBEEOut on or off as requested.
//
    unsigned int i;
    unsigned char command[] =
{0x7E,0x00,0x10,0x17,0x01,0x00,0x13,0xA2,0x00,0x40,0x69,0x39,0x27,0xFF,0xFE,0x02,'D',0x
35,5,0x7C};
    command[17]=0x30+XBEEOutIndex;
    if (bOn)
        command[18] = 5;
    else
        command[18] = 4;
    command[19] = 0;
    for (i=3;i<19;i++)
        command[19] += command[i];
    command[19] = 0xFF - command[19]%0xFF;

    for (i=0; i<=19; i++)
        UARTCharPut(UART1_BASE, command[i]);
}

```

Τα XBEE modem, στην προεπιλεγμένη διαμόρφωση των παραμέτρων τους στην ενσωματωμένη μνήμη flash που διαθέτουν, όταν ενεργοποιούνται λειτουργούν στην κατάσταση "transparent", δηλαδή ως RS232 bridge. Τα δεδομένα μεταφέρονται αυτούσια, σαν να υπήρχε ενσύρματη σύνδεση μεταξύ δύο UART. Για να εκμεταλλευτούμε τις γενικής χρήσης εισόδους-εξόδους του απομακρυσμένου μόντεμ ώστε να ελέγξουμε τα περιφερειακά που διαθέτει ενσωματωμένα (GPIO, PWM, ADC) ώστε να μη χρειαστεί να καταφύγουμε στη χρήση εξωτερικού μικροελεγκτή, πρέπει να επικοινωνήσουμε με το απομακρυσμένο μόντεμ μέσω ειδικών μηνυμάτων (API), χρησιμοποιώντας τις εντολές Remote Configuration Command και Remote Configuration Response. Για να γίνει αυτό, πρώτα στέλνουμε την AT εντολή ATAP1, ακολουθούμενη από την εντολή ATAC για άμεση έξοδο από το AT mode. Τώρα, το τοπικό modem επικοινωνεί με API μηνύματα (δε στέλνει δεδομένα σε transparent mode). Στέλνουμε το ειδικά διαμορφωμένο μήνυμα API και στη συνέχεια στέλνουμε την AT εντολή ATAP0, ακολουθούμενη από την εντολή ATAC για άμεση έξοδο από το AT mode. Τώρα, το τοπικό modem επικοινωνεί και πάλι σε transparent mode.

Επειδή η καθυστέρηση για την αλλαγή των mode είναι αισθητή, μπορεί να γίνει ρύθμιση της παραμέτρου GT(Guard Times) του τοπικού modem σε χαμηλότερη τιμή και αντίστοιχη μείωση της καθυστέρησης στην εφαρμογή (SysCtlDelay);

β. Ασύρματη σειριακή επικοινωνία (XBEE 802.15.4 ή ZigBee)

Λειτουργία: Γίνεται αποστολή και λήψη δεδομένων προς και από το XBEE modem

Περιγραφή λειτουργίας αποστολής: Το πλήκτρο “Αποστολή δεδομένων στο XBEE” καλεί τη συνάρτηση XBEEsendData:

```
function XBEEsendData() {
    var req = false;
    var lenZeroPad = document.getElementById("XBEEDataToSend").value.length + '';
    var dataToSend = document.getElementById("XBEEDataToSend").value;
    while(lenZeroPad.length < 3)
        lenZeroPad= "0" + lenZeroPad;
    function XBEEDataTransmissionComplete() {
        if (req.readyState == 4) {
            if (req.status == 200) {
                XBEEReceiveData();
            }
        }
    }
    if (window.XMLHttpRequest)
        req = new XMLHttpRequest();
    else if (window.ActiveXObject)
        req = new ActiveXObject("Microsoft.XMLHTTP");
    if (req) {
        req.open("GET", "/cgi-bin/XBEEsendData?length=" + lenZeroPad +
            "&data=" + dataToSend + Math.random(), true);
        req.onreadystatechange = XBEEDataTransmissionComplete;
        req.send(null);
    }
}
```

Γίνεται αίτηση για τη σελίδα “/cgi-bin/XBEEsendData” με προσαρτήματα το πλήθος των byte και τα ίδια τα byte προς αποστολή. Με τα δεδομένα που επιστρέφονται ενημερώνεται το πεδίο κειμένου, αντικαθιστώντας προηγούμενα δεδομένα αν υπάρχουν.

Ο http server εκτελεί το εξής τμήμα κώδικα:

```
//
// Process request to send XBEE data
//
if( strcmp(name, "/cgi-bin/XBEEsendData?length=", 29) == 0)
```

```

{
    // Set Local Modem to Transparent Mode
    SysCtlDelay(20000000);
    UARTCharPut(UART1_BASE, '+');
    UARTCharPut(UART1_BASE, '+');
    UARTCharPut(UART1_BASE, '+');
    SysCtlDelay(20000000);
    UARTCharPut(UART1_BASE, 'A');
    UARTCharPut(UART1_BASE, 'T');
    UARTCharPut(UART1_BASE, 'A');
    UARTCharPut(UART1_BASE, 'P');
    UARTCharPut(UART1_BASE, '0');
    UARTCharPut(UART1_BASE, 0X0D); //CR
    SysCtlDelay(20000000);
    UARTCharPut(UART1_BASE, 'A');
    UARTCharPut(UART1_BASE, 'T');
    UARTCharPut(UART1_BASE, 'C');
    UARTCharPut(UART1_BASE, 'N');
    UARTCharPut(UART1_BASE, 0X0D); //CR
    SysCtlDelay(20000000);

    //Transmit data to XBEE
    unsigned int dataCount = atoi(strndup(name+29, 3));
    for (i=0; i<dataCount; i++)
        UARTCharPut(UART1_BASE, name[38+i]);

    //
    // Setup the file structure to return whatever.
    //
    ptFile->data = NULL;
    ptFile->len = 0;
    ptFile->index = 0;
    ptFile->pextension = NULL;

    //
    // Return the file system pointer.
    //
    return(ptFile);
}

```

Σημείωση: Η εντολή AT για είσοδο σε transparent mode είναι χρήσιμη για την περίπτωση που για κάποιον λόγο το μόντεμ έχει παραμείνει σε κατάσταση API.

Περιγραφή λειτουργίας λήψης: Το πλήκτρο "Λήψη δεδομένων από το XBEE" καλεί τη συνάρτηση XBEEReceiveData:

```

function XBEEReceiveData() {
    var XBEEDataReception = false;

    function XBEEDataReceptionComplete() {
        if (XBEEDataReception.readyState == 4) {
            if (XBEEDataReception.status == 200) {

```


Z. Μαγνητικός Βομβητής ελεγχόμενος από παλμούς PWM

Λειτουργία: Γίνεται έλεγχος της συχνότητας και ενεργοποίηση/απενεργοποίηση του σήματος PWM που οδηγεί τον μαγνητικό βομβητή. Προέρχεται απο το παράδειγμα της βιβλιοθήκης enet_io.

Περιγραφή λειτουργίας :

Το πλήκτρο “Ενεργοποίηση/Απενεργοποίηση” καλεί τη συνάρτηση `pwm_onoff`:

```
function pwm_onoff() {
    var req = false;
    var pwm = false;
    function pwmComplete() {
        if (pwm.readyState == 4) {
            if (pwm.status == 200) {
                document.getElementById("pwmstate").innerHTML = "<div>"
                    + pwm.responseText + "</div>";
            }
        }
    }
    if (window.XMLHttpRequest) {
        req = new XMLHttpRequest();
        pwm = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
        req = new ActiveXObject("Microsoft.XMLHTTP");
        pwm = new ActiveXObject("Microsoft.XMLHTTP");
    }
    if (req) {
        req.open("GET", "/cgi-bin/pwm_onoff?id" + Math.random(), true);
        req.send(null);
    }
    if (pwm) {
        pwm.open("GET", "/pwmstate?id=" + Math.random(), true);
        pwm.onreadystatechange = pwmComplete;
        pwm.send(null);
    }
}
```

Γίνεται αίτημα για τη σελίδα “`cgi-bin/pwm_onoff`” και στη συνέχεια για τη σελίδα “`/pwmstate`” με τα αποτελέσματα της οποίας ενημερώνεται ο πίνακας ελέγχου για την τρέχουσα κατάσταση ενεργοποίησης του PWM.

Ο http server εκτελεί το εξής τμήμα κώδικα:

```
//  
// Process request to turn PWM ON/OFF  
//  
if(strcmp(name, "/cgi-bin/pwm_onoff", 18) == 0)  
{  
    //  
    // Turn PWM on/off  
    //  
    io_set_pwm(!io_is_pwm_on());  
  
    //  
    // Setup the file structure to return whatever.  
    //  
    ptFile->data = NULL;  
    ptFile->len = 0;  
    ptFile->index = 0;  
    ptFile->pextension = NULL;  
  
    //  
    // Return the file system pointer.  
    //  
    return(ptFile);  
}
```

καλείται η συνάρτηση:

```
void  
io_set_pwm(tBoolean bOn)  
{  
    //  
    // Enable or disable the PWM1 output.  
    //  
    PWMOutputState(PWM_BASE, PWM_OUT_1_BIT, bOn);  
}  
  
//  
// Request for PWM State?  
//  
if(strcmp(name, "/pwmstate?id", 12) == 0)  
{  
    static char pcBuf[4];  
  
    //  
    // Get the state of the PWM  
    //  
    io_get_pwmstate(pcBuf, 4);  
  
    ptFile->data = pcBuf;  
    ptFile->len = strlen(pcBuf);  
    ptFile->index = ptFile->len;  
    ptFile->pextension = NULL;  
    return(ptFile);  
}
```

καλείται η συνάρτηση:

```
void
io_get_pwmstate(char * pcBuf, int iBufLen)
{
    //
    // Get the state of the PWM1
    //
    if(HWREG(PWM_BASE + PWM_O_ENABLE) & PWM_OUT_1_BIT)
    {
        usnprintf(pcBuf, iBufLen, "ON");
    }
    else
    {
        usnprintf(pcBuf, iBufLen, "OFF");
    }
}
}
```

Το πλήκτρο “Καθορισμός Συχνότητας” καλεί τη συνάρτηση `pwm_freq_set`:

```
function pwm_freq_set() {
    var req = false;
    var pwmfreq = false;
    var FreqText = document.getElementById("pwmfreqtxt");
    function pwmfreqComplete() {
        if (pwmfreq.readyState == 4) {
            if (pwmfreq.status == 200) {
                document.getElementById("pwmfreq").innerHTML =
"<div>" + pwmfreq.responseText + "</div>";
            }
        }
    }
    if (window.XMLHttpRequest) {
        req = new XMLHttpRequest();
        pwmfreq = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
        req = new ActiveXObject("Microsoft.XMLHTTP");
        pwmfreq = new ActiveXObject("Microsoft.XMLHTTP");
    }
    if (req) {
        if (FreqText.value != "") {
            req.open("GET", "/pwm_freq?value=" + FreqText.value +
"&id=" + Math.random(), true);
            req.send(null);
        }
    }
}
```

```
if (pwmfreq) {  
    pwmfreq.open("GET", "/pwmfreqget?id=" + Math.random(), true);  
    pwmfreq.onreadystatechange = pwmfreqComplete;  
    pwmfreq.send(null);  
}  
}
```

Γίνεται αίτηση για τη σελίδα "/pwm_freq" για τον καθορισμό της συχνότητας και στη συνέχεια για τη σελίδα "/pwmfreqget", με τα δεδομένα της οποίας ενημερώνεται ο πίνακας ελέγχου για την τρέχουσα συχνότητα του PWM.

Ο http server εκτελεί το εξής τμήμα κώδικα:

```
//  
// Process request for PWM freq update  
//  
if(strncmp(name, "/pwm_freq?value=", 16) == 0)  
{  
    //  
    // Get Frequency String  
    //  
    data = name;  
    data += 16;  
    i = 0;  
    do  
    {  
        switch(data[i])  
        {  
            case 0:  
            case '&':  
                g_cSampleTextBuffer[i] = 0;  
                break;  
            case '+':  
                g_cSampleTextBuffer[i] = '+';  
                break;  
            default:  
                g_cSampleTextBuffer[i] = data[i];  
                break;  
        }  
        if(g_cSampleTextBuffer[i] == 0)  
        {  
            break;  
        }  
        i++;  
    }while(i < sizeof(g_cSampleTextBuffer));  
  
    //  
    // Set PWM Frequency  
    //  
    io_pwm_freq(ustrtoul(g_cSampleTextBuffer, NULL, 10));  
  
    //  
    // Setup the file structure to return whatever.  
    //  
    ptFile->data = NULL;  
    ptFile->len = 0;  
    ptFile->index = 0;  
    ptFile->pextension = NULL;  
  
    //  
    // Return the file system pointer.  
    //  
    return(ptFile);  
}
```

void

```

io_pwm_freq(unsigned long ulFreq)
{
    unsigned long ulPWMClock;

    //
    // Get the PWM clock
    //
    ulPWMClock = SysCtlClockGet()/4;

    //
    // Set the global frequency
    //
    g_ulFrequency = ulFreq;

    //
    // Set the period.
    //
    PWMGenPeriodSet(PWM_BASE, PWM_GEN_0, ulPWMClock / g_ulFrequency);

    //
    // Set the pulse width of PWM1
    //
    PWMPulseWidthSet(PWM_BASE, PWM_OUT_1,
        ((ulPWMClock * g_ulDutyCycle)/100) / g_ulFrequency);
}

//
// Request PWM Frequency?
//
if(strcmp(name, "/pwmfreqget?id", 14) == 0)
{
    static char pcBuf[16];

    //
    // Get the frequency of the PWM
    //
    usprintf(pcBuf, "%d", io_get_pwmfreq());

    ptFile->data = pcBuf;
    ptFile->len = strlen(pcBuf);
    ptFile->index = ptFile->len;
    ptFile->pextension = NULL;
    return(ptFile);
}

```

Το πλήκτρο “Καθορισμός *duty cycle*” καλεί τη συνάρτηση *pwm_dutycycle_set*:

```
function pwm_dutycycle_set() {
    var req = false;
    var pwmdutycycle = false;
    var DutyCycleText = document.getElementById("pwmdutycycleText");
    function pwmdutycycleComplete() {
        if (pwmdutycycle.readyState == 4) {
            if (pwmdutycycle.status == 200) {
                document.getElementById("pwmdutycycle").innerHTML
= "<div>" + pwmdutycycle.responseText + "</div>";
            }
        }
    }
    if (window.XMLHttpRequest) {
        req = new XMLHttpRequest();
        pwmdutycycle = new XMLHttpRequest();
    } else if (window.ActiveXObject) {
        req = new ActiveXObject("Microsoft.XMLHTTP");
        pwmdutycycle = new ActiveXObject("Microsoft.XMLHTTP");
    }
    if (req) {
        if (DutyCycleText.value != "") {
            req.open("GET", "/pwm_dutycycle?value=" + DutyCycle-
Text.value + "&id=" + Math.random(), true);
            req.send(null);
        }
    }
    if (pwmdutycycle) {
        pwmdutycycle.open("GET", "/pwmdutycycleget?id=" +
Math.random(), true);
        pwmdutycycle.onreadystatechange = pwmdutycycleComplete;
        pwmdutycycle.send(null);
    }
}
```

Γίνεται αίτηση για τη σελίδα “/pwm_dutycycle” για τον καθορισμό της συχνότητας και στη συνέχεια για τη σελίδα “/pwmdutycycleget”, με τα δεδομένα της οποίας ενημερώνεται ο πίνακας ελέγχου για το τρέχον *duty cycle* του PWM.

Ο http server εκτελεί το εξής τμήμα κώδικα:

```
//
// Process request for PWM Duty Cycle update
```

```

//
if(strncmp(name, "/pwm_dutycycle?value=", 21) == 0)
{
    //
    // Get Duty Cycle String
    //
    data = name;
    data += 21;
    i = 0;
    do
    {
        switch(data[i])
        {
            case 0:
            case '&':
                g_cSampleTextBuffer[i] = 0;
                break;
            case '+':
                g_cSampleTextBuffer[i] = ' ';
                break;
            default:
                g_cSampleTextBuffer[i] = data[i];
                break;
        }
        if(g_cSampleTextBuffer[i] == 0)
        {
            break;
        }
        i++;
    }while(i < sizeof(g_cSampleTextBuffer));

    //
    // Set PWM Duty Cycle
    //
    io_pwm_dutycycle(ustrtol(g_cSampleTextBuffer, NULL, 10));

    //
    // Setup the file structure to return whatever.
    //
    ptFile->data = NULL;
    ptFile->len = 0;
    ptFile->index = 0;
    ptFile->pextension = NULL;

    //
    // Return the file system pointer.
    //
    return(ptFile);
}

void
io_pwm_dutycycle(unsigned long ulDutyCycle)
{
    unsigned long ulPWMClock;

```



```

//
// Get the PWM clock
//
ulPWMClock = SysCtlClockGet()/4;

//
// Set the global duty cycle
//
g_ulDutyCycle = ulDutyCycle;

//
// Set the period.
//
PWMGenPeriodSet(PWM_BASE, PWM_GEN_0, ulPWMClock / g_ulFrequency);

//
// Set the pulse width of PWM1
//
PWMPulseWidthSet(PWM_BASE, PWM_OUT_1,
                  ((ulPWMClock * g_ulDutyCycle)/100) / g_ulFrequency);
}

```

```

//
// Request PWM Duty Cycle?
//
if(strncmp(name, "/pwmdutycycleget?id", 19) == 0)
{
    static char pcBuf[16];

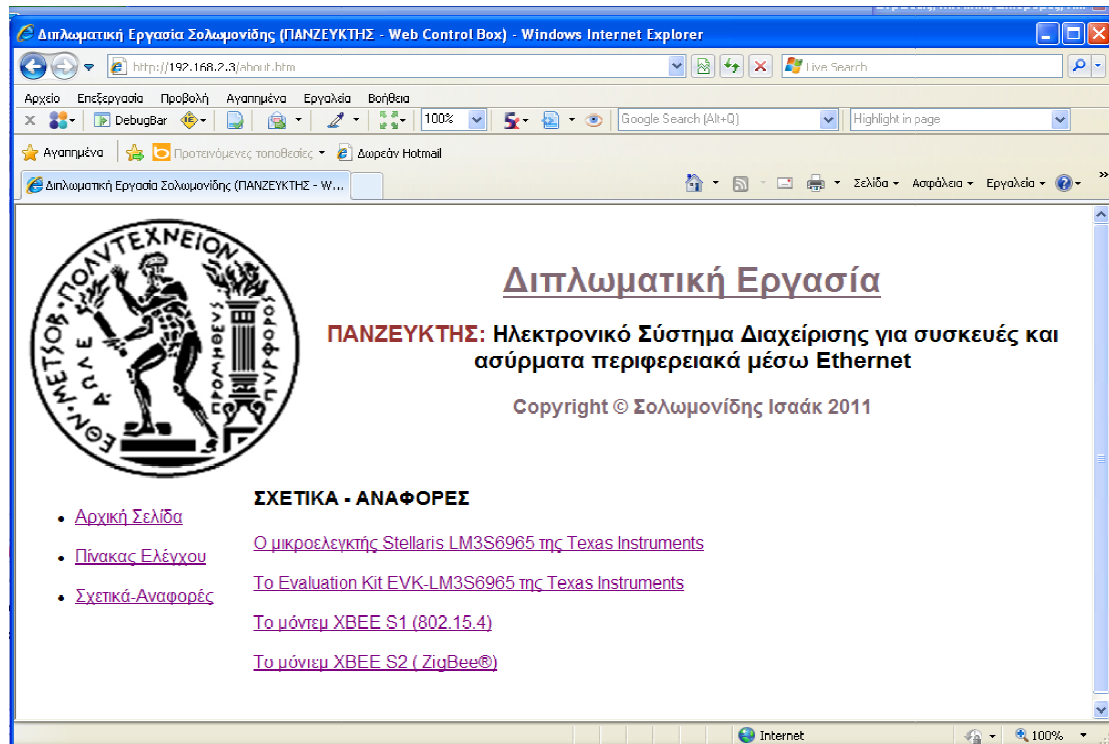
    //
    // Get the duty cycle of te PWM
    //
    usprintf(pcBuf, "%d", io_get_pwmdutycycle());

    ptFile->data = pcBuf;
    ptFile->len = strlen(pcBuf);
    ptFile->index = ptFile->len;
    ptFile->pextension = NULL;
    return(ptFile);
}

```

3.about.htm

Σχετικά – Αναφορές: Πρόκειται για μια στατική σελίδα με χρήσιμους συνδέσμους



ΠΑΡΑΡΤΗΜΑ

Διάσπαρτη Υπολογιστική (Ubiquitous Computing - ubicomp)

Η Διάσπαρτη Υπολογιστική είναι ένα μοντέλο επικοινωνίας ανθρώπου-μηχανής στο οποίο η συλλογή και επεξεργασία της πληροφορίας έχει ενσωματωθεί σε αντικείμενα καθημερινής χρήσης («έξυπνες» συσκευές). Στο πλαίσιο των καθημερινών του δραστηριοτήτων, κάποιος που «κάνει χρήση» τέτοιων συσκευών γίνεται υποκείμενο επεξεργασίας από πολλές συσκευές και συστήματα ταυτόχρονα, χωρίς πολλές φορές ο ίδιος να το αντιλαμβάνεται.

Πιο τυπικά, η Διάσπαρτη Υπολογιστική ορίζεται ως «Μηχανές που προσαρμόζονται στο ανθρώπινο περιβάλλον αντί να υποχρεώνουν τους ανθρώπους να εισέρχονται στο δικό τους».

Κατηγορίες «έξυπνων» συσκευών:

- Macro
 - Tabs: φορητές συσκευές μεγέθους λίγων εκατοστών
 - Pads: Συσκευές χειρός
 - Boards: Συσκευές διάδρασης μεγάλου μεγέθους
- Micro
 - Dust: μικροσκοπικές συσκευές μεγέθους από nm έως λίγα mm (π.χ. MEMS). Η «έξυπνη» σκόνη (Smart Dust) αποτελείται από πολλά «σωματίδια» που το καθένα διαθέτει αισθητήρες και όλα μαζί επικοινωνούν συνήθως ασύρματα.
 - Skin: «έξυπνο» ύφασμα που βασίζεται σε πολυμερή υλικά που φωτοβολούν ή είναι αγωγίμα και μπορούν να διαμορφωθούν σε προϊόντα όπως ρούχα και κουρτίνες (π.χ. οργανικές οθόνες – OLED). Ακόμη και μικροηλεκτρομηχανικά συστήματα (Micro Electro-Mechanical Systems- MEMS) μπορούν να αλειφθούν σε διάφορες επιφάνειες που μας περιβάλλουν ώστε αυτές να αποτελούν διασυνδεδεμένες επιφάνειες MEMS.
 - Clay: Πλήθος MEMS μπορεί να διαμορφωθούν στην επιθυμητή τρισδιάστατη μορφή και να αποτελούν ένα αντικείμενο καθημερινής χρήσης με ενσωματωμένη διαδραστικότητα ή και «ευφυΐα».[8]

Διάχυτη Ευφυΐα (Ambient Intelligence)

Η Διάχυτη Ευφυΐα αναφέρεται σε περιβάλλοντα ηλεκτρονικών συστημάτων τα οποία αντιλαμβάνονται την παρουσία των ανθρώπων και ανταποκρίνονται σε αυτή.

Τα συστήματα αυτά:

- Είναι ενσωματωμένα: πολλές διασυνδεδεμένες συσκευές ενσωματώνονται στο περιβάλλον.
- Διαθέτουν αντιληπτικότητα του περιβάλλοντός τους (context awareness): μπορούν να αναγνωρίσουν έναν άνθρωπο και τη θέση του.
- Είναι παραμετροποιήσιμα: Προσαρμόζονται στις ανάγκες του χρήστη.
- Είναι προσαρμοστικά: Μεταβάλλονται ανταποκρινόμενα στις επιλογές και τις συνήθειες του χρήστη.
- Κάνουν πρόβλεψη: Μπορούν να προβλέψουν τις επιθυμίες του χρήστη χωρίς τη διαμεσολάβηση του.[9]

Στόχος: Η ενσωμάτωση συστημάτων Διάχυτης Ευφυΐας σε οικιακό περιβάλλον.

Συστατικά μέρη:

Location Awareness: Στα ρούχα ή τα παπούτσια του κάθε χρήστη, έχει ενσωματωθεί ένα RFID tag. Δίκτυο αισθητήρων σε όλο το σπίτι παρακολουθεί την παρουσία χρηστών στα δωμάτια ή και συγκεκριμένους τομείς των δωματίων. Ένας κόμβος του δικτύου αυτών των αισθητήρων συνδέεται στη σειριακή θύρα του «Πανζεύκτη». Αισθητήρες από έξυπνο ύφασμα στις καρέκλες και τους καναπέδες και τα κρεβάτια παρέχουν πληροφορία για την κατανομή βάρους που δέχονται καθώς και τη μέση θερμοκρασία. Στη μπανιέρα, πιεζοηλεκτρικοί αισθητήρες αναγνωρίζουν το βάρος του χρήστη.

Φωτισμός: Όλα τα φωτιστικά σώματα του σπιτιού ελέγχονται από μονάδες «Pan-terminal» με μόντεμ Zigbee. Στο σαλόνι υπάρχει σύστημα πολυχρωματικού φωτισμού. Υπάρχουν μονάδες «Panterminal» εφοδιασμένες με φωτοαισθητήρες σε καίρια σημεία περιμετρικά του σπιτιού.

Σκίαση: Όλα τα συστήματα σκίασης και συσκότισης του σπιτιού (τέντες, ρολά στόρια, κουρτίνες) ελέγχονται από σερβομηχανισμούς που οδηγούνται από μονάδες «Panterminal».

Θέρμανση: Υπάρχει κεντρική θέρμανση. Μονάδες «Panterminal» με μόντεμ Zigbee καταγράφουν τη θερμοκρασία και υγρασία του κάθε δωματίου και οδηγούν ηλεκτροβάνες που ελέγχουν τα θερμαντικά σώματα αυτόνομα σε κάθε δωμάτιο.

Ζεστό νερό: Η θερμοκρασία του ηλιακού θερμοσιφώνου καθώς και η ροή του νερού(ροόμετρο) μετράται από ένα «Panterminal», το οποίο επίσης ελέγχει μέσω ενός ρελέ ισχύος την ενεργοποίηση ή όχι του ηλεκτρικού κυκλώματος του θερμοσιφώνου.

Υδρευση: Όλες οι βρύσες ελέγχονται από μικρούς κινητήρες που οδηγούνται από μονάδες «Panterminal» οι οποίες διαθέτουν φωτοκύτταρο και αισθητήρα θερμοκρασίας. Είναι έτσι δυνατή και η ρύθμιση της θερμοκρασίας απευθείας από τον χρήστη.

Τρόφιμα: Το «έξυπνο ψυγείο» διαθέτει σύνδεση Ethernet, αναγνωρίζει μέσω RFID ποιά τρόφιμα περιέχει, από πότε καθώς και τις ημερομηνίες λήξης αυτών. Επίσης παρέχει πληροφορίες στον «Πανζεύκτη» μέσω Ethernet για την κατάστασή του (διακυμάνσεις θερμοκρασίας, διακοπές). Αισθητήρες RFID τοποθετούνται και στα ντουλάπια των τροφίμων.

Διασκέδαση: Υπάρχει Home Entertainment System που επικοινωνεί με τον «Πανζεύκτη» μέσω Ethernet.

Πρόσβαση: Όλες οι εξωτερικές πόρτες του σπιτιού καθώς και πόρτες γκαράζ ελέγχονται από μονάδες «Panterminal» με μόντεμ Zigbee. Καταγράφεται η κατάστασή τους (ανοικτή/κλειστή) και εκτελείται κατάλληλη εντολή (άνοιγμα/κλείσιμο πόρτας). Στο γκαράζ υπάρχει και αισθητήρας RFID, όπως σε όλα τα δωμάτια. Συσκευή αναγνώρισης ίριδας βρίσκεται εγκατεστημένη στην εξώπορτα και συνδέεται στο Ethernet.

Ασφάλεια: Μονάδες «Panterminal» με μόντεμ Zigbee εφοδιασμένες με παθητικά ραντάρ υπερέθρων (PIR) τοποθετούνται σε στρατηγικά σημεία του σπιτιού. Σειρήνα ελέγχεται απευθείας από τον «Πανζεύκτη, όπως και Αισθητήρες καπνού και αερίων ZigBee.

Τηλεφωνία/Internet: Το ADSL router επικοινωνεί μέσω Ethernet με τον «Πανζεύκτη».

Υγεία: Το ρολόι που φορά ο χρήστης διαθέτει αισθητήρα που παρακολουθεί τη θερμοκρασία του και επικοινωνεί με RFID.

Λειτουργίες Διάχυτης Ευφυΐας:

Ο «Πανζεύκτης» θα διαθέτει τη δυνατότητα να καταγράφει όλες τις πρόσφατες δραστηριότητες, να τις αναλύει και να προτείνει δράσεις ή ακόμη και να αυτενεργεί.

Παράδειγμα 1: Όταν ο χρήστης πλησιάσει τη βρύση του νιπτήρα με το χέρι του, τότε αρχίζει να ρέει το νερό στη συνήθη για εκείνος θερμοκρασία. Αν θελήσει κάποια άλλη ρύθμιση, τότε πολύ απλά επεμβαίνει απευθείας στη βρύση. Η νέα ρύθμιση καταγράφεται και συνεκτιμάται στην επόμενη ανάλυση για αυτόματη επιλογή θερμοκρασίας. Το ίδιο συμβαίνει και με το ντους του μπάνιου.

Παράδειγμα 2: Όταν η οδοντόβουρτσα του χρήστη, η οποία διαθέτει ενσωματωμένο rfid tag, βγει από τη θήκη της περισσότερες φορές από κάποιο όριο, μετρώντας από την πρώτη φορά που τοποθετήθηκε, τότε αυτόματα προσθέτει στη λίστα με τα ψώνια μια καινούργια οδοντόβουρτσα στον επιθυμητό τύπο του χρήστη.

Παράδειγμα 2: Όταν το σύστημα ανιχνεύσει ότι κάποιο από τα τρόφιμα δεν έχει χρησιμοποιηθεί καιρό, τότε προτείνει κάποια συνταγή με αυτό, μόνο εφόσον τις τελευταίες ημέρες έχει μπει στο σπίτι κάποιος από τους χρήστες που την έχει επιλέξει έστω και μία φορά.

Όλα τα παραπάνω επιχειρηματικά μοντέλα μπορούν να υλοποιηθούν με την αποκλειστική χρήση του «Πανζεύκτη» με σύνδεση και στο Internet (π.χ. για λήψη των μαγειρικών συνταγών).

Βιβλιογραφία – Αναφορές

[1] <http://www.embeddedsystem.com/>

[2] http://en.wikipedia.org/wiki/Embedded_systems

[3] Το εγχειρίδιο χρήσης του Stellaris® LM3S6965 Evaluation Board
<http://www.ti.com/lit/ug/spmu029a/spmu029a.pdf>

[4] Η σελίδα των μόντεμ XBEE® της Digi International
<http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/>

[5] Σελίδα προϊόντος του μικροελεγκτή LM3S6965 της Texas Instruments
<http://focus.ti.com/docs/prod/folders/print/lm3s6965.html>

[6] Using the Stellaris® Ethernet Controller with Lightweight IP (lwIP)
<http://www.ti.com/lit/an/spma025b/spma025b.pdf>

[7] Η βιβλιοθήκη λογισμικού EK-LM3S6965 Firmware Development Package της Texas Instruments
<http://focus.ti.com/docs/toolsw/folders/print/sw-ek-lm3s6965.html>

[8] Διάσπαρτη Υπολογιστική (Ubiquitous Computing)
http://en.wikipedia.org/wiki/Ubiquitous_computing

[9] Διάχυτη Ευφυΐα (Ambient Intelligence)
http://en.wikipedia.org/wiki/Ambient_intelligence