



ΤΕΙ ΗΠΕΙΡΟΥ

**ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε.**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**ΕΛΕΓΧΟΣ ΘΥΡΩΝ ΣΕ ARM EMBEDDED COMPUT
ER ΜΕ ΤΗ ΧΡΗΣΗ LINUX**

Μαρία Καραδήμα

Επιβλέπων: Τζάλλας Αλέξανδρος

Άρτα Άυγουστος 2017

ΕΛΕΓΧΟΣ ΘΥΡΩΝ ΣΕ ARM EMBEDDED COMPUT ER ΜΕ ΤΗ ΧΡΗΣΗ LINUX

Δήλωση μη λογοκλοπής

Δηλώνουμε υπεύθυνα και γνωρίζοντας τις κυρώσεις του Ν. 2121/1993 περί Πνευματικής Ιδιοκτησίας, ότι η παρούσα πτυχιακή εργασία είναι εξ ολοκλήρου αποτέλεσμα δικής μας ερευνητικής εργασίας, δεν αποτελεί προϊόν αντιγραφής ούτε προέρχεται από ανάθεση σε τρίτους. Όλες οι πηγές που χρησιμοποιήθηκαν (κάθε είδους, μορφής και προέλευσης) για τη συγγραφή της περιλαμβάνονται στη βιβλιογραφία.

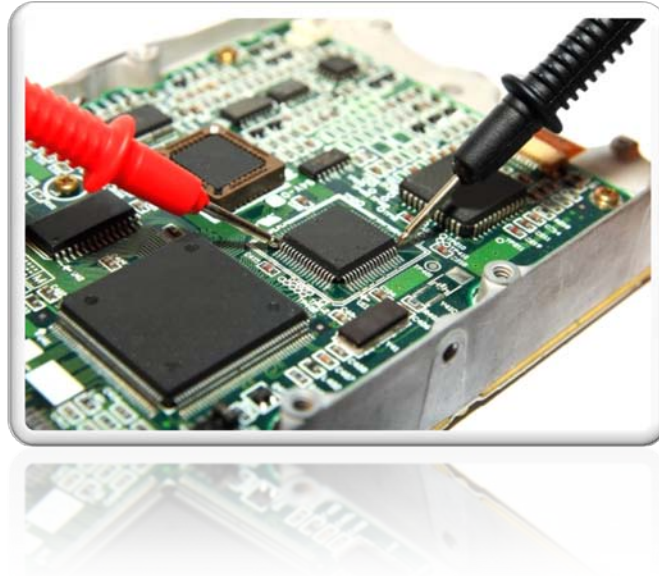
Μαρία Καραδήμα

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

1. ΜΙΚΡΟΕΛΕΓΚΤΕΣ	
1.1 Εισαγωγή στους μικροελεγκτες.....	5
1.2 Διαφορες απο τον μικροεπεξεργαστη.....	6
1.3 Συνηθη υποσυστηματα.....	7
1.4 Διαδεδομενες κατηγοριες μικροελεγκτων.....	8,9
1.5 Γενικη δομη ενος μικροελεγκτη.....	9
2. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ARM	
2.1 Πυρηνες ARM.....	10,11
2.2 Ενσωματωμενα λειτουργικα συστηματα.....	12,13
3. ΤΟ ΒΕΑΓΛΕΒΟΝΕ	
3.1 Ο ρόλος του Beaglebone.....	13,14
3.2 Εισαγωγή στην πλατφόρμα.....	15,16
4. ΥΛΟΠΟΙΗΣΗ	
4.1 Επικοινωνία με το Beaglebone	
4.2 Κώδικας	
4.2.1 Κώδικας Cpp.....	17,18
4.2.2 Κώδικας html.....	18,19
4.3 Αποτέλεσμα κώδικα.....	21,22
4.4 Βήματα και εικονες υλοποιησης.....	23-34
5. ΒΙΒΛΙΟΓΡΑΦΙΑ.....	35

1.ΜΙΚΡΟΕΛΕΓΚΤΕΣ

1.1 ΕΙΣΑΓΩΓΗ ΣΤΟΥΣ ΜΙΚΡΟΕΛΕΓΚΤΕΣ



Μικροελεγκτής είναι ένας τύπος επεξεργαστή. Ο μικροελεγκτής στην ουσία είναι μία παραλλαγή του μικροεπεξεργαστή, με την βασική διαφορά ότι ο μικροελεγκτής μπορεί να λειτουργήσει με ελάχιστα εξωτερικά συστήματα λόγω των πολλών ενσωματωμένων υποσυστημάτων που διαθέτει. Το συναντάμε σε όλα τα ενσωματωμένα συστήματα (embedded systems) ελέγχου χαμηλού και μεσαίου κόστους.

Οι μικροελεγκτές είναι αυτόνομοι και δεν χρειάζονται κανένα άλλο ολοκληρωμένο κύκλωμα για να λειτουργήσουν. Χάρη σε αυτό έχουν χαμηλή κατανάλωση ισχύος λόγω των απλούστερων διασυνδέσεων.

Η αρχιτεκτονική που χρησιμοποιούν είναι Harvard για την επικοινωνία της CPU και την μνήμης.

Οι κατηγορίες που χωρίζονται οι μικροελεγκτές είναι δύο με βάση τον τρόπο που χρησιμοποιούν τις εντολές για τον υπολογισμό μιας εργασίας ανεξάρτητου μεγέθους.

Οι κατηγορίες αυτές είναι:

- Η αρχιτεκτονική RISC
- Η αρχιτεκτονική CISC

Ο προγραμματισμός των μικροελεγκτών μπορεί να γίνει είτε από γλώσσες υψηλού επιπέδου, είτε σε γλώσσα κατανοητή από τον μικροελεγκτή (π.χ. Assembly). Η πιο διαδεδομένη γλώσσα προγραμματισμού είναι η C και C++.

1.2 ΔΙΑΦΟΡΕΣ ΑΠΟ ΤΟΝ ΜΙΚΡΟΕΠΕΞΕΡΓΑΣΤΗ

Ο ρυθμός με τον οποίο αναπτύσσονται οι εφαρμογές σήμερα είναι μεγάλη, καθώς η λειτουργικότητα του τελικού συστήματος καθορίζεται από τα εξωτερικά περιφερειακά τα οποία διασυνδέονται με την κεντρική μονάδα (μικροεπεξεργαστή) η οποία δεν είναι εξειδικευμένη. Σε αντίθεση με του μικροεπεξεργαστές για ενσωματωμένα συστήματα (μικροελεγκτές) οι οποίοι έχουν μικρότερες ή και μηδαμινές δυνατότητες συνεργασίας με εξωτερικά περιφερειακά η ευελιξία είναι περιορισμένη καθώς και η υπολογιστική ισχύ.

Πλεονεκτήματα μικροελεγκτών:

- Αυτονομία, μέσω της ενσωμάτωσης σύνθετων περιφερειακών υποσυστημάτων όπως μνήμες και θύρες επικοινωνίας. Έτσι πολλοί μικροελεγκτές δεν χρειάζονται κανένα άλλο ολοκληρωμένο κύκλωμα για να λειτουργήσουν.
- Η ενσωμάτωση περιφερειακών σημαίνει ευκολότερη υλοποίηση εφαρμογών λόγω των απλούστερων διασυνδέσεων. Επίσης, οδηγεί σε χαμηλότερη κατανάλωση ισχύος, μεγιστοποιώντας τη φορητότητα και ελαχιστοποιεί το κόστος της συσκευής στην οποία ενσωματώνεται ο μικροελεγκτής.
- Χαμηλό κόστος.
- Μεγαλύτερη αξιοπιστία, και πάλι λόγω των λιγότερων διασυνδέσεων.
- Μειωμένες εκπομπές ηλεκτρομαγνητικών παρεμβολών και μειωμένη ευαισθησία σε αντίστοιχες παρεμβολές από άλλες ηλεκτρικές και ηλεκτρονικές συσκευές. Το πλεονέκτημα αυτό προκύπτει από το μικρότερο αριθμό και μήκος εξωτερικών διασυνδέσεων καθώς και τις χαμηλότερες ταχύτητες λειτουργίας.
- Περισσότεροι διαθέσιμοι ακροδέκτες για ψηφιακές εισόδους-εξόδους (για δεδομένο μέγεθος ολοκληρωμένου κυκλώματος), λόγω της μη δέσμευσής τους για τη σύνδεση εξωτερικών περιφερειακών.
- Μικρό μέγεθος συνολικού υπολογιστικού συστήματος.

1.3 ΣΥΝΗΘΗ ΥΠΟΣΥΣΤΗΜΑΤΑ

Το ολοκληρωμένο σύστημα που αποτελεί τον μικροεπεξεργαστή περιέχει μόνο την Λογική και Αριθμητική Μονάδα(ALU) στοιχειώδεις καταχωρητές (registers), προσωρινή μνήμη RAM πολύ υψηλής ταχύτητας (cache memory) και, κάποιες φορές, τον ελεγκτή μνήμης (memory controller). Όμως, για τη λειτουργία ενός πλήρους ενσωματωμένου υπολογιστικού συστήματος, απαιτούνται πολλά εξωτερικά υποσυστήματα και περιφερειακά.

Κάποια απο τα υποσυστήματα και τα περιφερειακά:

- Κύκλωμα συνδετικής λογικής (glue logic) για τη σύνδεση των εξωτερικών μνημών και άλλων περιφερειακών παράλληλης σύνδεσης στην αρτηρία δεδομένων (bus) του επεξεργαστή.
- Μνήμη προγράμματος (τύπου ROM, FLASH, EPROM κλπ) η οποία περιέχει το λογισμικό του συστήματος. Σε κάποια μοντέλα, είναι δυνατό το κλείδωμα αυτής της μνήμης, μετά την εγγραφή της, ώστε να προστατευτεί το περιεχόμενό της από αντιγραφή.
- Μεγάλο μέγεθος μνήμης RAM.
- Μόνιμη μνήμη αποθήκευσης παραμέτρων λειτουργίας (τύπου EEPROM ή NVRAM) η οποία να μπορεί να γράφεται τον πυρήνα του μικροελεγκτή. Αυτή η μνήμη έχει, έναντι της FLASH, το πλεονέκτημα της δυνατότητας διαγραφής και εγγραφής οποιουδήποτε μεμονωμένου byte.
- Κύκλωμα αρχικοποίησης (reset).
- Διαχειριστή αιτήσεων διακοπής (interrupt request controller) από τα περιφερειακά.
- Κύκλωμα επιτήρησης τροφοδοσίας (brown-out detection) το οποίο παρακολουθεί την τροφοδοσία και αρχικοποιεί ολόκληρο το σύστημα όταν αυτή πέσει κάτω από τα ανεκτά όρια, προλαμβάνοντας έτσι την αλλοίωση των δεδομένων.
- Κύκλωμα επιτήρησης λειτουργίας (watchdog timer) το οποίο αρχικοποιεί το σύστημα, αν αυτό εμφανίσει σημάδια δυσλειτουργίας λόγω κολλήματος (hang).
- Τοπικό ταλαντωτή για την παροχή παλμών χρονισμού (clock).
- Έναν ή περισσότερους χρονιστές-απαριθμητές υψηλής ταχύτητας (hardware timer-counter) για τη δημιουργία καθυστερήσεων, μέτρηση διάρκειας γεγονότων, απαρίθμηση γεγονότων και άλλων λειτουργιών ακριβούς χρονισμού.

- Ρολόι πραγματικού χρόνου (Real Time Clock, RTC) το οποίο τροφοδοτείται από ανεξάρτητη μπαταρία και γι αυτό πρέπει να έχει πολύ χαμηλή κατανάλωση ρεύματος.
- Σειρά ανεξάρτητων ψηφιακών εισόδων και εξόδων (Parallel Input-Output, PIO).

1.4 ΔΙΑΔΕΔΟΜΕΝΕΣ ΚΑΤΗΓΟΡΙΕΣ ΜΙΚΡΟΕΛΕΓΚΤΩΝ

Εξαιτίας του ισχυρού ανταγωνισμού αλλά και της τάσης ενσωμάτωσης των μικροελέκτων σε κάθε ηλεκτρική και ηλεκτρονική συσκευή, η βιομηχανία μικροελεγκτών έχει καταλήξει στην παραγωγή ανταγωνιστικών μοντέλων μαζικής παραγωγής καθώς και μικροελεγκτών για πιο εξειδικευμένες εφαρμογές.

Έχουμε λοιπόν τις εξής κατηγορίες:

1. Μικροελεγκτές (καμιά φορά 4-bit αλλά συνήθως 8-bit) πολύ χαμηλού κόστους, γενικής χρήσης, με πολύ μικρό αριθμό ακροδεκτών (ακόμη και λιγότερους από 8). Σχεδιάζονται με έμφαση στη χαμηλή κατανάλωση ισχύος και την αυτάρκεια, ώστε να χρειάζονται ελάχιστα ή και καθόλου εξωτερικά εξαρτήματα και να μη μπορεί να αντιγραφεί εύκολα το εσωτερικό λογισμικό τους. Απουσιάζει η δυνατότητα επέκτασης της μνήμης τους. Μερικά μοντέλα είναι ευρέως γνωστά στους ερασιτέχνες ηλεκτρονικούς
2. Μικροελεγκτές (συνήθως 8-bit αλλά και 16 ή 32-bit) χαμηλού κόστους, γενικής χρήσης, με μέτριο έως σχετικά μεγάλο αριθμό ακροδεκτών. Διαθέτουν μεγάλο αριθμό κοινών περιφερειακών, όπως θύρες UART, I2C, SPI ή CAN, μετατροπείς αναλογικού σε ψηφιακό και ψηφιακού σε αναλογικό. Στους κατασκευαστές της Άπω Ανατολής (Ιαπωνία, Κορέα), συνηθίζεται η ενσωμάτωση ελεγκτών οθόνης υγρών κρυστάλλων και πληκτρολογίου. Μερικές φορές παρέχουν δυνατότητα εξωτερικής επέκτασης της μνήμης τους.
3. Μικροελεγκτές (κυρίως 32-bit) μέσου κόστους, γενικής χρήσης, με μεγάλο αριθμό ακροδεκτών. Χαρακτηρίζονται από έμφαση στην ταχύτητα εκτέλεσης εντολών, υψηλή αυτάρκεια περιφερειακών και μεγάλες δυνατότητες εσωτερικής ή εξωτερικής μνήμης προγράμματος (FLASH)

και RAM. Στο χώρο αυτό έχουν ισχυρή παρουσία οι αρχιτεκτονικές με υψηλή μεταφερσιμότητα λογισμικού (portability) από τον ένα στον άλλο κατασκευαστή.

4. Μικροελεγκτές εξειδικευμένων εφαρμογών, οι οποίοι ενσωματώνουν συνήθως κάποιο εξειδικευμένο πρωτόκολλο επικοινωνίας το οποίο υλοποιείται πάντοτε σε hardware. Τέτοιοι μικροελεγκτές χρησιμοποιούνται σε τηλεπικοινωνιακές συσκευές όπως τα μόντεμ.

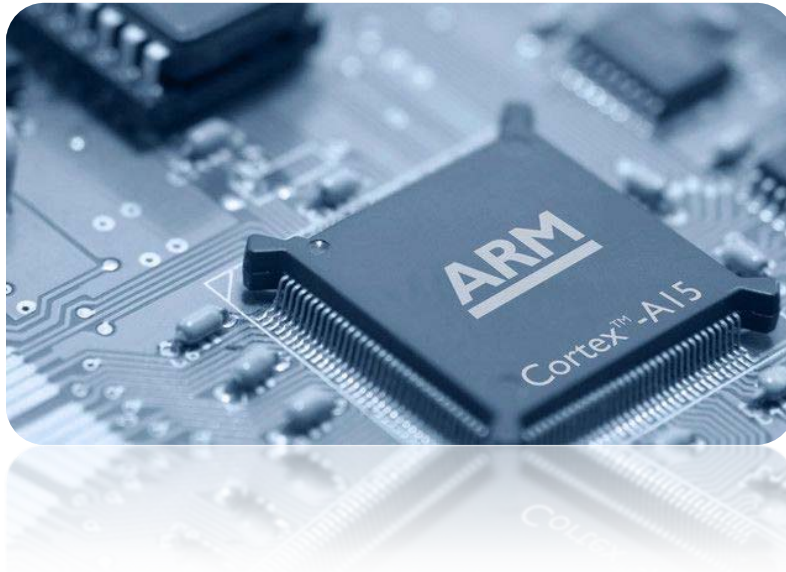
Οι περισσότερες πωλήσεις μικροελεγκτών αφορά αυτούς των 8-bit καθώς και η κατηγορία με το χαμηλότερο κόστος και το μικρότερο μέγεθος λογισμικού για το ίδιο αποτέλεσμα και αυτό συμβαίνει γιατί οι σύγχρονες οικογένειες μικροελεγκτών 8-bit έχουν πολύ βελτιωμένες επιδόσεις σε σχέση με το παρελθόν.

1.5 ΓΕΝΙΚΗ ΔΟΜΗ ΕΝΟΣ ΜΙΚΡΟΕΛΕΓΚΤΗ

Στη δομή ενός μικροελεγκτή διακρίνουμε τρία μέρη:

- Τον πυρήνα του μικροελεγκτή (Core)
 - Κεντρική μονάδα επεξεργασίας (CPU)
 - Μνήμη
- Τα περιφερειακά (Peripherals)
 - Θύρες εισόδου/εξόδου
 - Θύρες σειριακής επικοινωνίας
 - Οι μετρητές χρόνου
 - Μετατροπέας αναλογικού σήματος σε ψηφιακό
 - Θύρα παράλληλης επικοινωνίας
 - Η μονάδα διαμόρφωσης πλάτους

2.ΑΡΧΙΤΕΚΤΟΝΙΚΗ ARM



Η ARM είναι μια αρχιτεκτονική συνόλου εντολών RISC των 32-bit, η οποία έχει αναπτυχθεί από την ARM Holdings. Η αρχιτεκτονική ARM είναι η πιο συχνά χρησιμοποιούμενη αρχιτεκτονική συνόλου εντολών 32-bit όσον αφορά τους επεξεργαστές που παράγονται. Τα αρχικά σημαίνουν Προχωρημένη Μηχανή RISC (Advanced RISC Machine), ενώ παλαιότερα σήμαιναν Μηχανή RISC Acorn (Acorn RISC Machine). Η αρχιτεκτονική δημιουργήθηκε αρχικά από την Acorn Computers για χρήση στους προσωπικούς υπολογιστές της, με τα πρώτα προϊόντα που βασίζονταν στην αρχιτεκτονική ARM να είναι οι σειρά Acorn Archimedes που εμφανίστηκε το 1987.

Οι μικροεπεξεργαστές ARM είναι απλοί, για αυτό είναι κατάλληλοι για εφαρμογές χαμηλής ισχύος. Αυτό έχει ως αποτέλεσμα να έχουν υπερισχύσει στις αγορές των κινητών και των ενσωματωμένων συστημάτων, σαν μικροί και σχετικά χαμηλού κόστους μικροεπεξεργαστές και μικροελεγκτές. Το 2005, περίπου το 98% των πάνω από ένα δισεκατομμύριο κινητών τηλεφώνων που πωλούνται κάθε χρόνο είχαν τουλάχιστον έναν επεξεργαστή ARM. Το 2009 οι επεξεργαστές ARM αντιστοιχούσαν περίπου στο 90% όλων των ενσωματωμένων επεξεργαστών RISC 32-bit και χρησιμοποιούνται σε μεγάλο βαθμό σε καταναλωτικά ηλεκτρονικά προϊόντα, συμπεριλαμβανομένων των προσωπικών ψηφιακών βοηθών (personal digital assistants,

PDA's), των κινητών τηλεφώνων, των συσκευών ψηφιακής μουσικής και πολυμέσων, των φορητών κονσολών βιντεοπαιχνιδιών, των αριθμομηχανών και περιφερειακών υπολογιστών όπως οι σκληροί δίσκοι και οι δρομολογητές.

Οι επεξεργαστές ARM αναπτύσσονται από την ARM και τις εταιρείες που έχουν άδεια από την ARM. Σημαντικές οικογένειες επεξεργαστών ARM που έχουν αναπτυχθεί από την ARM Holdings είναι οι ARM7, οι ARM9, οι ARM11 και οι Cortex. Σημαντικοί επεξεργαστές ARM που έχουν αναπτυχθεί από τις εταιρείες με άδεια είναι ο StrongARM της DEC, ο i.MX της Freescale, ο XScale της Marvell (πρώην Intel), ο Tegra της Nvidia, ο Nova και ο NovaThor της ST-Ericsson, ο Snapdragon της Qualcomm, η σειρά προϊόντων OMAP της Texas Instruments, ο Hummingbird της Samsung και οι A4 και A5 της Apple.

2.1 ΠΥΡΗΝΕΣ ARM

Πυρήνες ARM [[Επεξεργασία](#) | [επεξεργασία κώδικα](#)]

Αρχιτεκτονική	Οικογένεια
ARMv1	ARM1
ARMv2	ARM2, ARM3
ARMv3	ARM6, ARM7
ARMv4	StrongARM, ARM7TDMI, ARM9TDMI
ARMv5	ARM7EJ, ARM9E, ARM10E, XScale
ARMv6	ARM11
ARMv7	Cortex
ARMv8	Δεν υπάρχουν πυρήνες διαθέσιμοι ακόμα. Θα υποστηρίξει διευθύνσεις 64-bit ^[15]

Η ARM παρέχει έναν κατάλογο όλων των εταιρειών που υλοποιούν πυρήνες ARM.^[16]

Οι πυρήνες ARM χρησιμοποιούνται σε πολλά προϊόντα όπως για παράδειγμα στα έξυπνα κινητά (smartphones). Χρησιμοποιείται επίσης στο φορητό αναπαραγωγέα πολυμέσων iPod της Apple, στην φορητή κονσόλα βιντεοπαιχνιδιών Nintendo DS και σε πολλά άλλα.

Για να διατηρηθεί η σχεδίαση καθαρή, απλή και γρήγορη, η αρχική υλοποίηση ARM είχε κατασκευαστεί κατευθείαν, χωρίς τη χρήση μικροκώδικα.

Η αρχιτεκτονική ARM περιλαμβάνει τα εξής χαρακτηριστικά RISC:

- Αρχιτεκτονική φόρτωσης/αποθήκευσης
- Ομοιόμορφο αρχείο $16 \times 32\text{-bit}$ καταχωρητών.
- Σταθερό εύρος εντολών 32-bit για εύκολη αποκωδικοποίηση και διοχέτευση (pipelining), με μειονέκτημα την μικρότερη πυκνότητα κώδικα.
- Συνήθως η εκτέλεση γίνεται σε έναν κύκλο.

2.2 ΕΝΣΩΜΑΤΩΜΕΝΑ ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ

Η αρχιτεκτονική ARM υποστηρίζεται από πολλά ενσωματωμένα λειτουργικά συστήματα και λειτουργικά συστήματα πραγματικού χρόνου.

➤ UNIX:

Η αρχιτεκτονική ARM υποστηρίζεται από λειτουργικά συστήματα τύπου Unix. Το Unix ή UNIX είναι λειτουργικό σύστημα Ηλεκτρονικών Υπολογιστών, το οποίο αναπτύχθηκε κατά τις δεκαετίες του 1960 και του 1970 από ομάδα εργαζομένων των εργαστηρίων Μπελ (Bell Labs) της εταιρείας AT&T. Σήμερα τα συστήματα Unix έχουν χωριστεί σε πολλούς κλάδους και αναπτύσσονται τόσο από την AT&T όσο και από άλλους εμπορικούς παράγοντες, όπως και από αρκετούς μη κερδοσκοπικούς οργανισμούς, όπως το πρόγραμμα GNU.

➤ LINUX:

Το Linux ή GNU/Linux είναι ένα λειτουργικό σύστημα που αποτελείται από ελεύθερο λογισμικό. Η χρήση του είναι παρόμοια με αυτή του Unix, αλλά όλος ο πηγαίος κώδικας του έχει γραφτεί από την αρχή ως ελεύθερο λογισμικό υπό την ελεύθερη άδεια χρήσης GNU General Public License.

➤ BSD:

Το BSD (Berkeley Software Distribution - μερικές φορές αναφέρεται και Berkeley Unix) είναι ένα σύστημα προερχόμενο

από το Unix, που διανέμεται από το πανεπιστήμιο του Μπέρκλεϊ της Καλιφόρνιας. Η ανάπτυξη του BSD ξεκίνησε τη δεκαετία του '70. Το όνομα χρησιμοποιείται επίσης για να περιγράψει το σύνολο των σύγχρονων διανομών που προέρχονται από τις αρχικές διανομές του BSD. Το BSD έτυχε ευρείας αναγνώρισης με τις εκδόσεις Unix που ήταν διαθέσιμες για "workstation-class" συστήματα. Αυτό μπορεί να αποδοθεί στην ευκολία με την οποία μπορούσε να χορηγηθεί άδεια για χρήση και την οικειότητα που είχαν με το σύστημα οι ιδρυτές πολλών τεχνολογικών επιχειρήσεων κατά τη δεκαετία του '80.

➤ **Windows:**

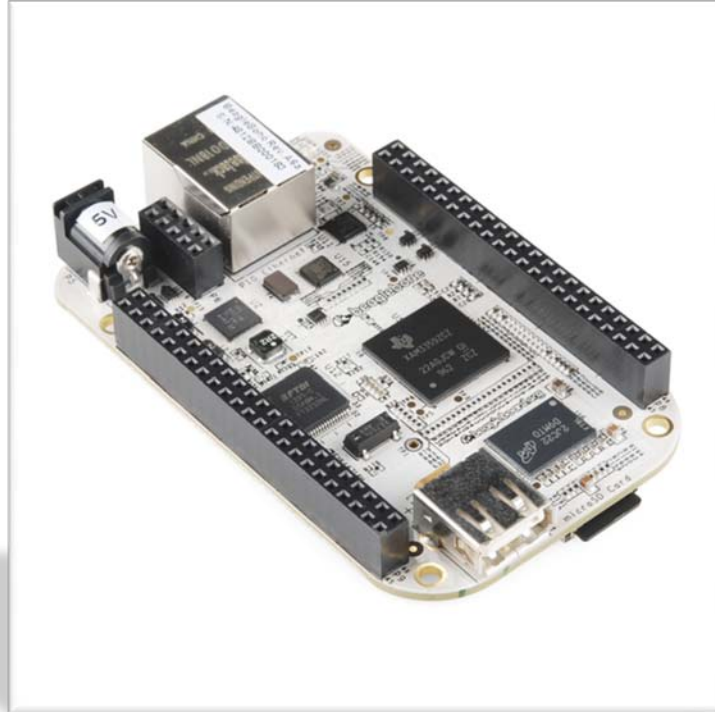
Τα Windows είναι μια σειρά από λειτουργικά συστήματα για προσωπικούς υπολογιστές και διακομιστές. Η σειρά εμφανίστηκε πρώτα από τη Microsoft το 1985. Τα Microsoft Windows είναι λογισμικό κλειστού κώδικα. Αναπτύχθηκαν για τους IBM συμβατούς μικροθπολογιστές, οι οποίοι ήταν βασισμένοι στην αρχιτεκτονική x86 της Intel και μέχρι και σήμερα σχεδόν όλες οι εκδόσεις είναι κατασκευασμένες για την πλατφόρμα αυτή. Τα Windows έχουν κυκλοφορήσει σε διάφορες εκδόσεις από τότε που πρωτοεμφανίστηκαν μέχρι σήμερα.

3. Το BeagleBone

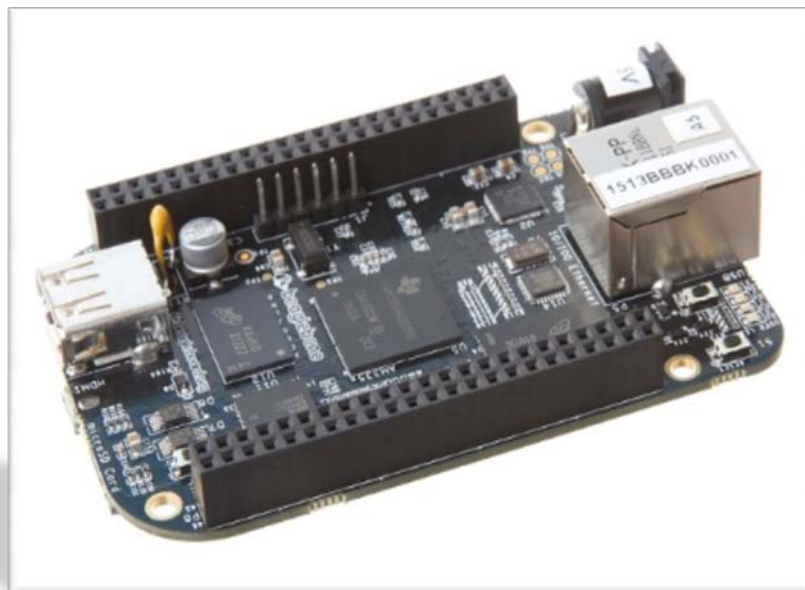
3.1 Ο ρόλος του Beaglebone

Το beaglebone είναι υλικό χαμηλής ισχύος ανοιχτού κώδικα υπολογιστή που παράγεται από την Texas Instruments σε συνεργασία με την Digi-Key και Newark element14. Η Beaglebone σχεδιάστηκε με την ανάπτυξη λογισμικού ανοιχτού κώδικα με τρόπο έτσι ώστε να αναδειχθεί η Texas Instrument's OMAP3530 system-on-a-chip. Το διοικητικό συμβούλιο αναπτύχθηκε από μία μικρή ομάδα μηχανικών ως εκπαιδευτικό συμβούλιο που θα μπορούσε να χρησιμοποιηθεί σε κολέγια από όλο τον κόσμο για να διδάξει τις ικανότητες ανοιχτού κώδικα πηγής και λογισμικού. Πωλείται στο κοινό με

την άδεια της Creative Commons. Στην οικογένεια του Beaglebone έχουμε το BeagleBoneoriginalκαι τοαναβαθμισμένο αλλά χαμηλού κόστους BeagleBoneblack.



Εικόνα 1:Beaglebone Original



Εικόνα 2: Beaglebone Black

	BeagleBone Enhanced \$69	BeagleBone Black \$55
Processor	AM3358BZCZ100, 1GHz	AM3358BZCZ100, 1GHz
Video Out	HDMI	HDMI
DRAM	1024MB DDR3L 800MHz	512MB DDR3L 800MHz
Flash	4GB eMMC, uSD	4GB eMMC, uSD
Boot Flash	8Mb	No
USB Ports	4	1
Network Speed	1000Mb	100Mb
Sensors	Accelerometer/Gyro/Compass + Barometer/Temperature	None
Wireless Connectivity	Wifi/Bluetooth	None

Εικόνα 3: Beaglebone original vs Beaglebone Black

3.2 Εισαγωγή στην πλατφόρμα

Το beagleboneoriginal(white) είναι μια υπολογιστική πλατφόρμα ανοιχτού κώδικα Linux χαμηλού κόστους. Η πλατφόρμα αυτή μπορεί να χρησιμοποιηθεί για να κατασκευάσουμε διάφορες πολύπλοκες εφαρμογές που διασυνδέουν λογισμικό υψηλού επιπέδου με ηλεκτρονικά κυκλώματα χαμηλού επιπέδου. Το Beaglebone διαθέτει pins και διάλυσους εισόδου/εξόδου δίνοντας την δυνατότητα σύνδεσης με ηλεκτρονικά εξαρτήματα, μονάδες και συσκευές USB.

Τα χαρακτηριστικά την πλατφόρμας είναι:

- ✓ Επεξεργαστής Sitara ARM Cortex-A8 που εκτελείται σε 720 MHz
- ✓ Διαθέτει 256 MB μνήμη RAM

- ✓ Δύο υποδοχές επέκτασης 46 ακίδων
- ✓ Ethernet on chip
- ✓ Υποδοχή microSD
- ✓ Θύρα υποδοχής USB
- ✓ Θύρα συσκευής πολλαπλών χρήσεων
- ✓ Περιλαμβάνει σειριακό έλεγχο χαμηλού επιπέδου και συνδέσεις εντοπισμού σφαλμάτων υλικού JTAG

4.ΥΛΟΠΟΙΗΣΗ

4.1 Επικοινωνία με το Beaglebone

Γενικά υπάρχουν πολλοί τρόποι για να συνδεθούμε με το Beaglebone και να έχουμε πρόσβαση στην γραμμή εντολών του. Σε λειτουργικό σύστημα Windows, είναι απαραίτητη η εγκατάσταση ενός τρίτου προγράμματος ενώ σε λειτουργικό σύστημα Linux γίνεται απλά μέσα από ένα terminal. Για την συγκεκριμένη υλοποίηση χρησιμοποιήσαμε υπολογιστή με Linux για αυτό και δεν εγκαταστήσαμε κάποιο επιπλέον πρόγραμμα.

4.2 ΚΩΔΙΚΑΣ

4.2.1 ΚΩΔΙΚΑΣ Cpp

```
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>
#include <unistd.h>

int main(int argc, const char *argv[])
{
    if (argc != 3)
    {
        printf("Usage:\n");
        printf("\tledCtl<led><on/off>\n");
        printf("\n");
        printf("<led> : Number between 0-3\n");
        printf("<on/off>: 1 = on, 0 = off\n");

        return 1;
    }

    int ledNum = atoi(argv[1]);
    if (ledNum < 0 || ledNum > 3)
    {
        printf("<led> : Number between 0-3\n");

        return 1;
    }

    char ledPath[1024];
    sprintf(ledPath,
"/sys/class/leds/beaglebone::usr%d/brightness", ledNum);
    int fid = open(ledPath, O_WRONLY);

    int onOff = atoi(argv[2]);
    switch (onOff)
    {
```

```

        case 0:
            write(fid, "0", 1);
            break;
        case 1:
            write(fid, "1", 1);
            break;
        default:
            printf("<on/off>: 1 = on, 0 = off\n");
            return 1;
    }

    close(fid);

    return 0;
}

```

4.2.2 ΚΩΔΙΚΑΣ HTML

```

<html>
<head>
<title>BeagleBone LED Changer</title>
<style type="text/css">
p{ display: table-cell; }
button {width:75px; margin: 2px auto;}
</style>
<?php
    If (isset($_GET['led']) &&isset($_GET['onoff']))
    {
        $led = $_GET['led'];
        $onoff = $_GET['onoff'];
exec("/www/lala1 $led $onoff" );
    }
?>
</head>
<body>

```

```

<div style="width: 200px; margin:0px auto;">
<div style="width: 100px; float: left;">
<p>LED #0:</p>
<button
type="button"onclick="location.href='test.php?led=0&onOff=1
' ">ON</button>
<button
type="button"onclick="location.href='test.php?led=0&onOff=0
' ">OFF</button>
</div>
<div style="width: 100px; float: left;">
<p>LED #1:</p>
<button
type="button"onclick="location.href='test.php?led=1&onOff=1
' ">ON</button>
<button
type="button"onclick="location.href='test.php?led=1&onOff=0
' ">OFF</button>
</div>

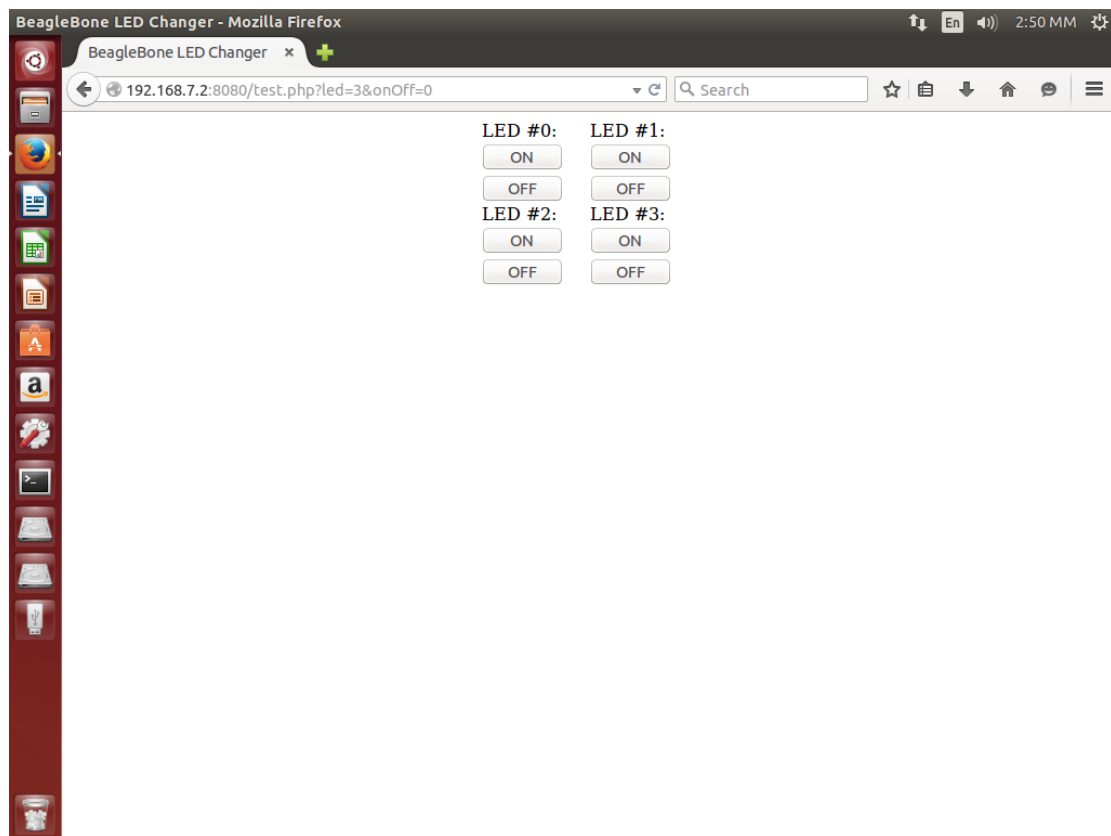
<div style="width: 100px; float: left;">
<p>LED #2:</p>
<button
type="button"onclick="location.href='test.php?led=2&onOff=1
' ">ON</button>
<button
type="button"onclick="location.href='test.php?led=2&onOff=0
' ">OFF</button>
</div>
<div style="width: 100px; margin-left: 100px;">
<p>LED #3:</P>
<button
type="button"
onclick="location.href='test.php?led=3&onOff=1'
">ON</button>
<button
type="button"
onclick="location.href='test.php?led=3&onOff=0'
">OFF</button>

```

</div>

</div>

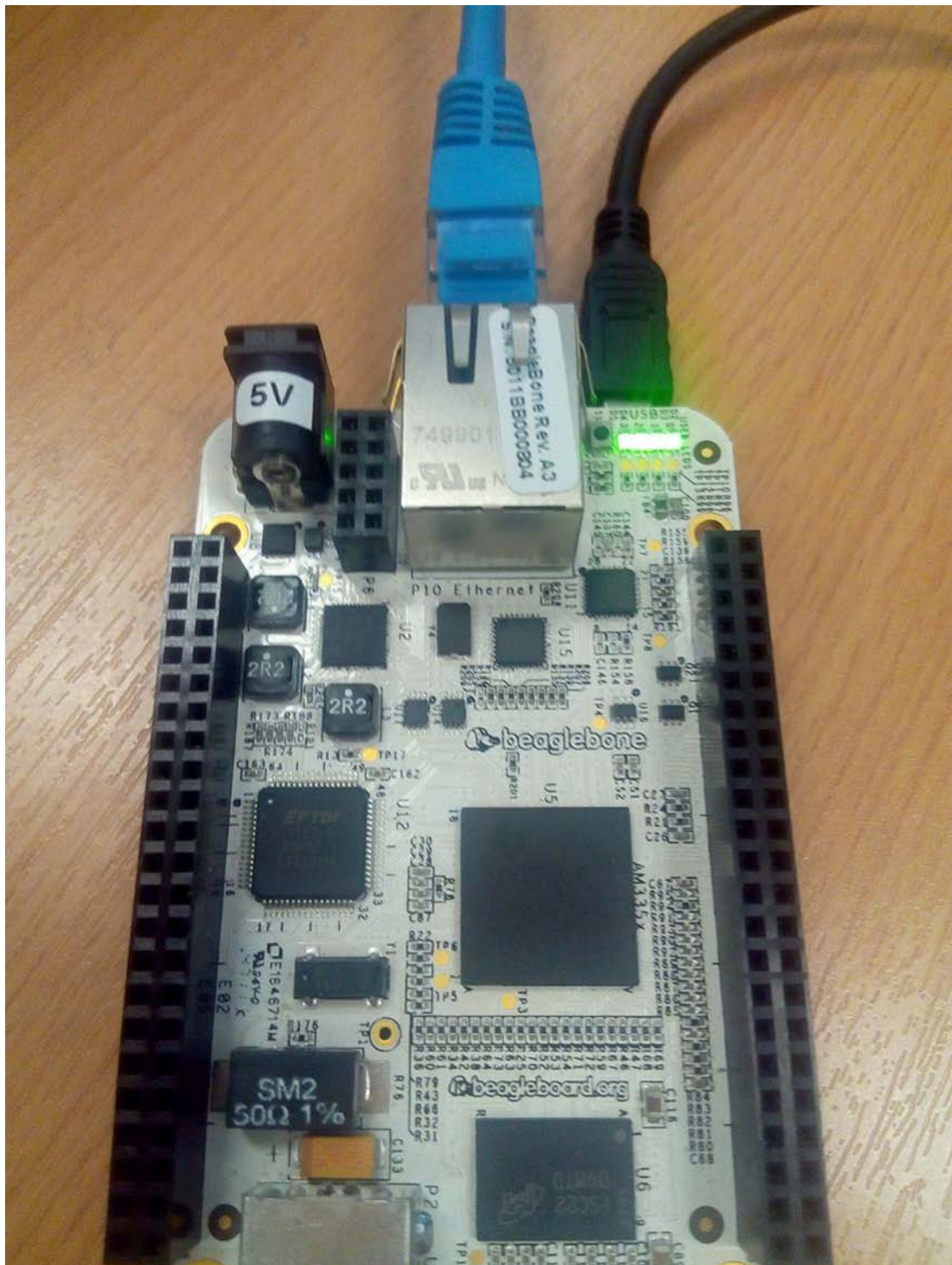
4.3 ΑΠΟΤΕΛΕΣΜΑ ΚΩΔΙΚΑ



ΕΙΚΟΝΑ 4:ΑΠΟΤΕΛΕΣΜΑ ΚΩΔΙΚΑ HTML

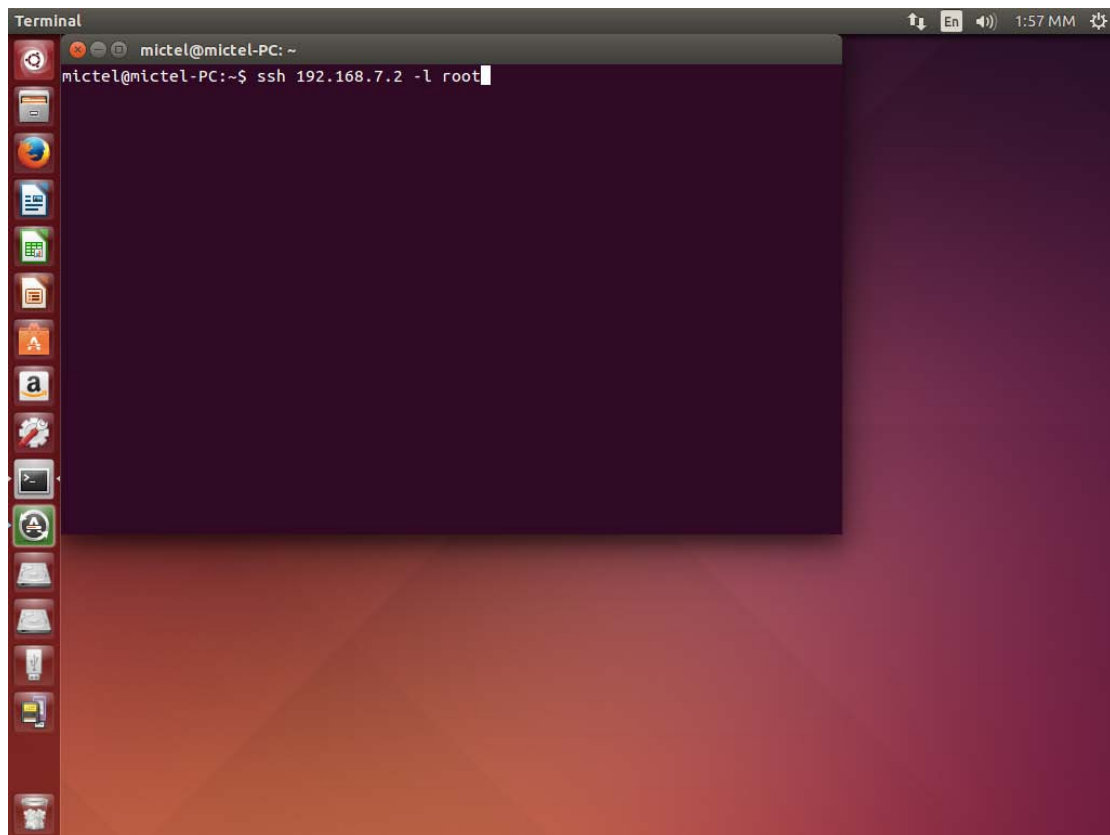
ΒΗΜΑ 1: Δημιουργία τοπικού δικτύου για τη σύνδεση και λειτουργία του BeagleBone με τη χρήση ενός hub και καλωδίων Ethernet.

ΒΗΜΑ 2: Συνδεση BeagleBone με υπολογιστημεσω usb και καλωδιο ethernet

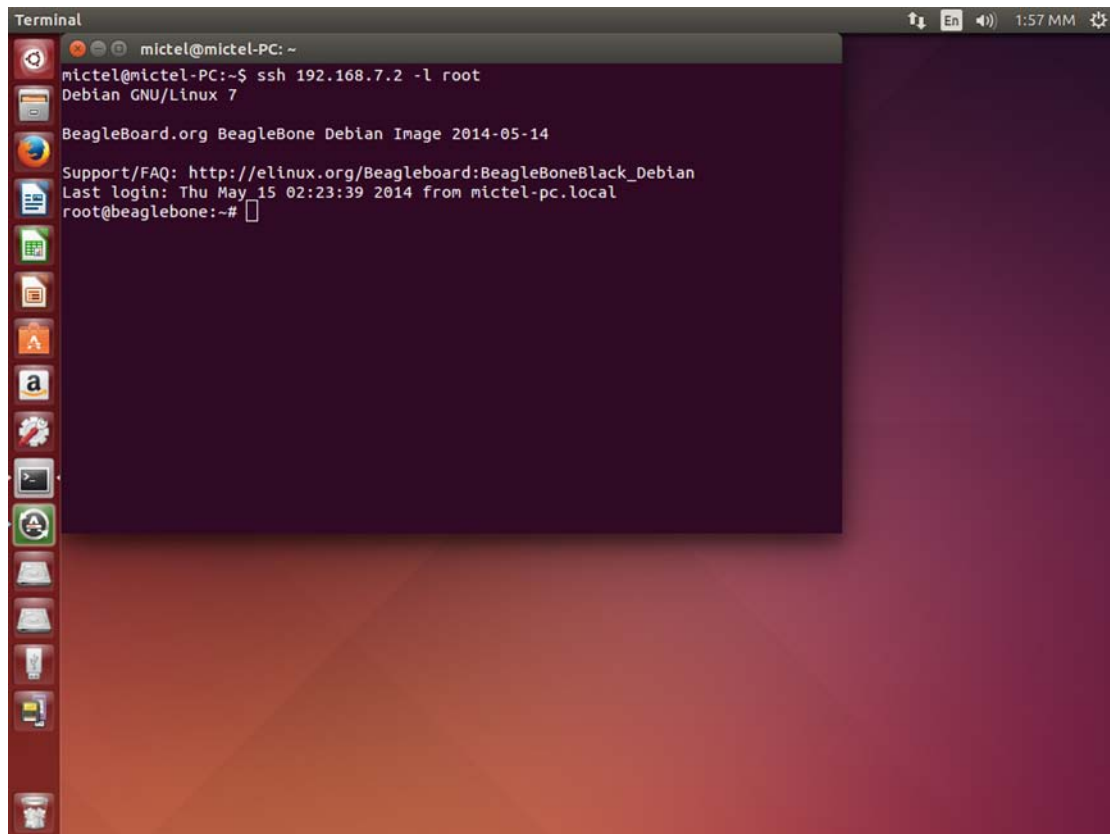


ΕΙΚΟΝΑ 5

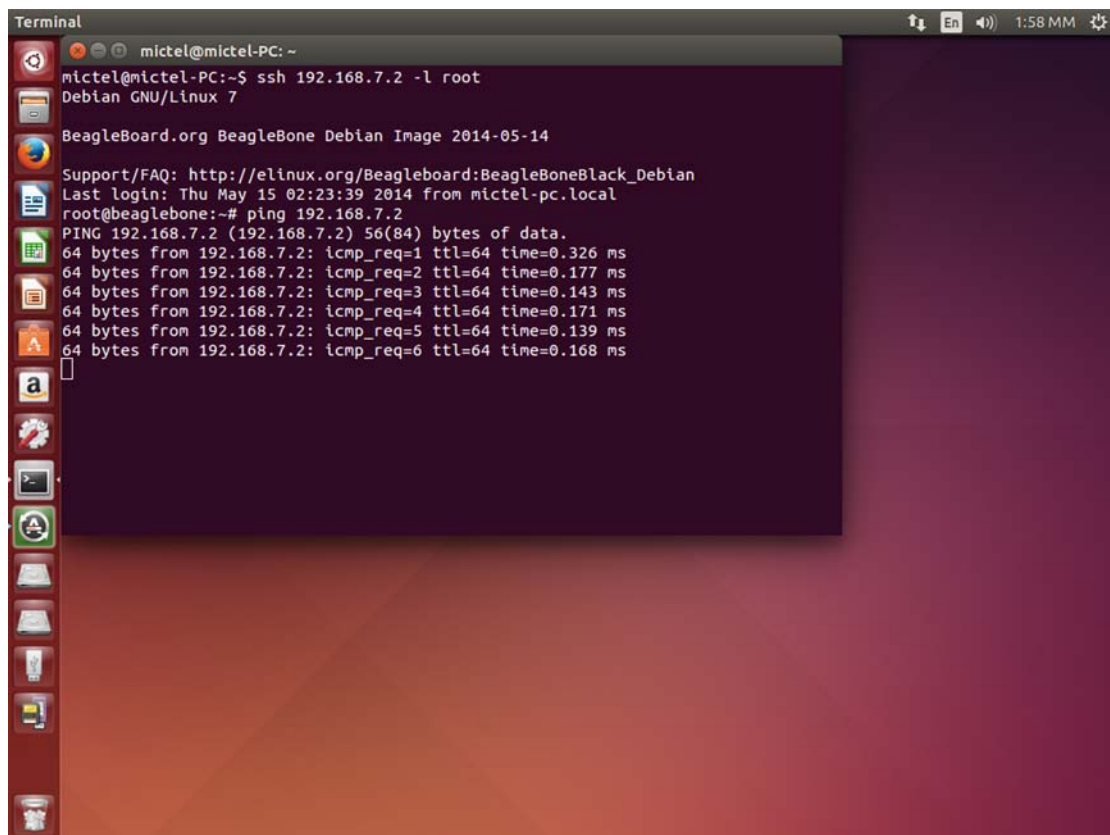
ΒΗΜΑ 3:ΕΠΙΚΟΙΝΩΝΙΑ ΥΠΟΛΟΓΙΣΤΗ ΜΕ BeagleBone



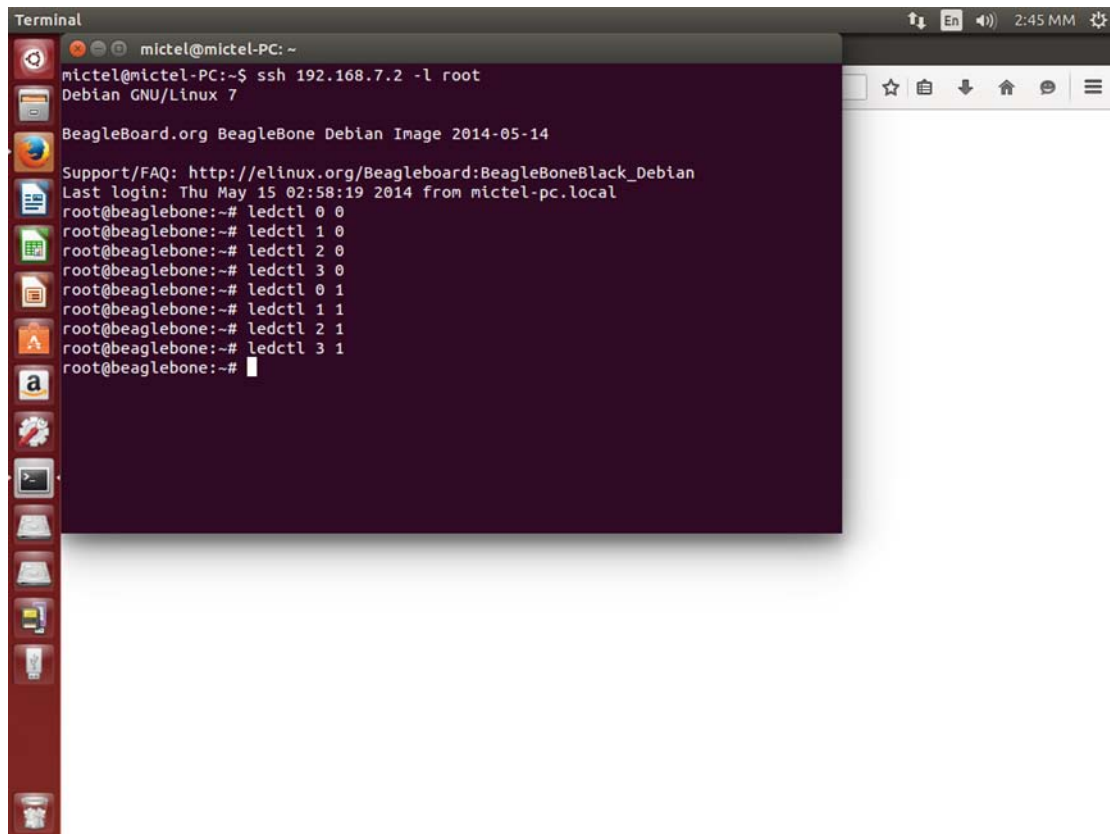
ΕΙΚΟΝΑ 6



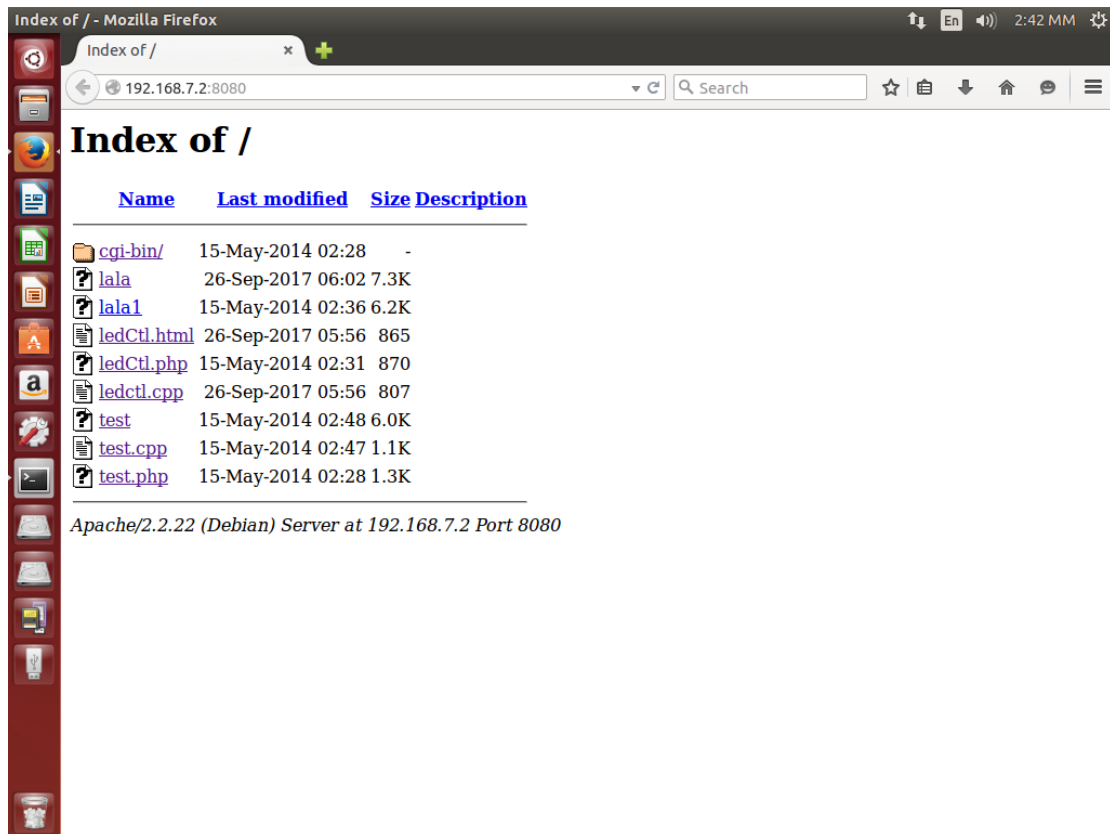
EIKONA 7



EIKONA8:Ανταποκριση συνδεσης

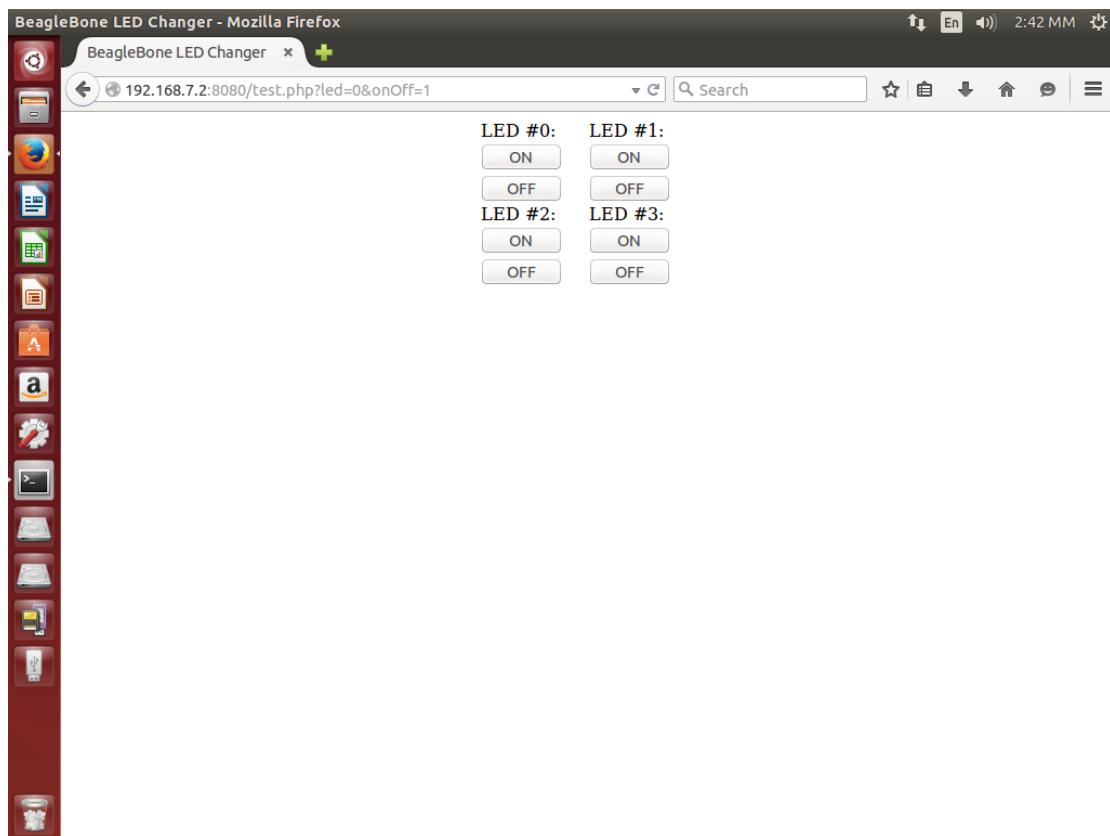


ΕΙΚΟΝΑ 9: ledon/off με χρήση εντολής μέσω τερματικού



ΕΙΚΟΝΑ 10: Συνδεσμε το BeagleBone διαδίκτυα με τη χρήση της διεύθυνσης 192.168.7.2:8080

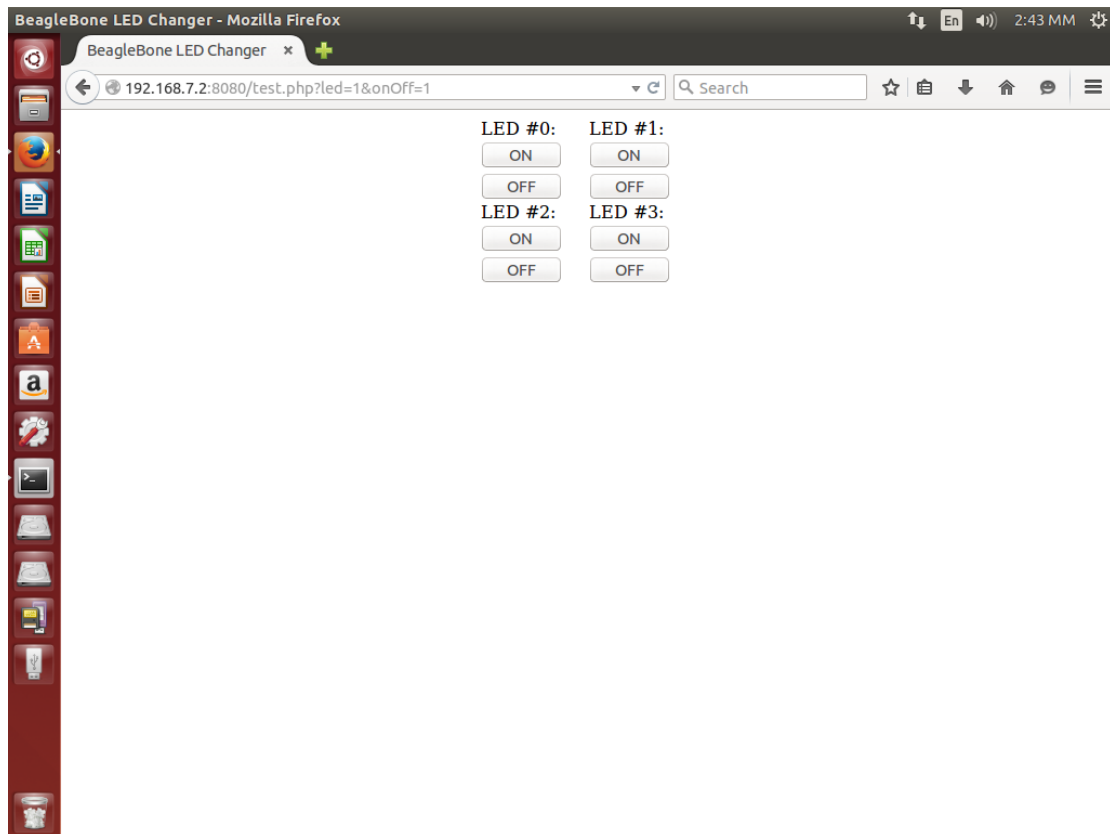
Βημα 4:Χρηση της Webσελιδας με τα κουμπια και ανταποκριση του BeagleBone



ΕΙΚΟΝΑ 11:ΠΑΤΗΜΕΝΟ ΤΟ ΚΟΥΜΠΙ LED #0>ON



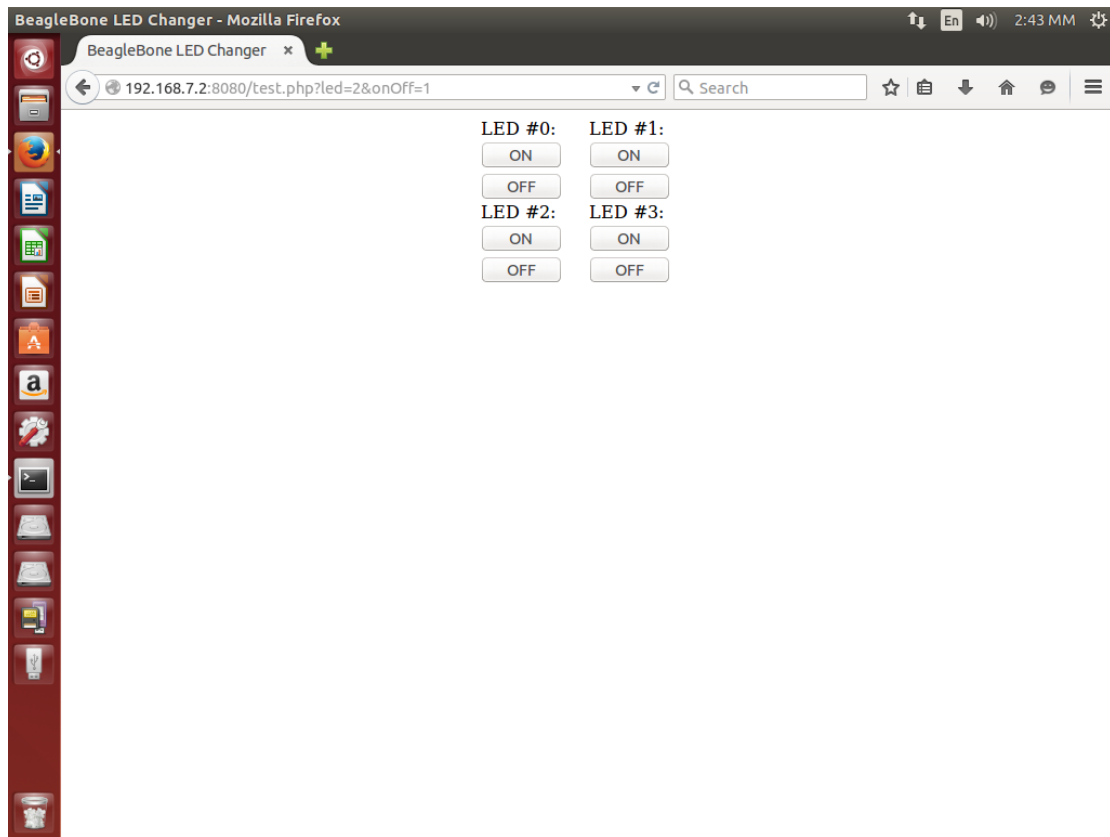
EIKONA 12:Ανάμενο το led #0



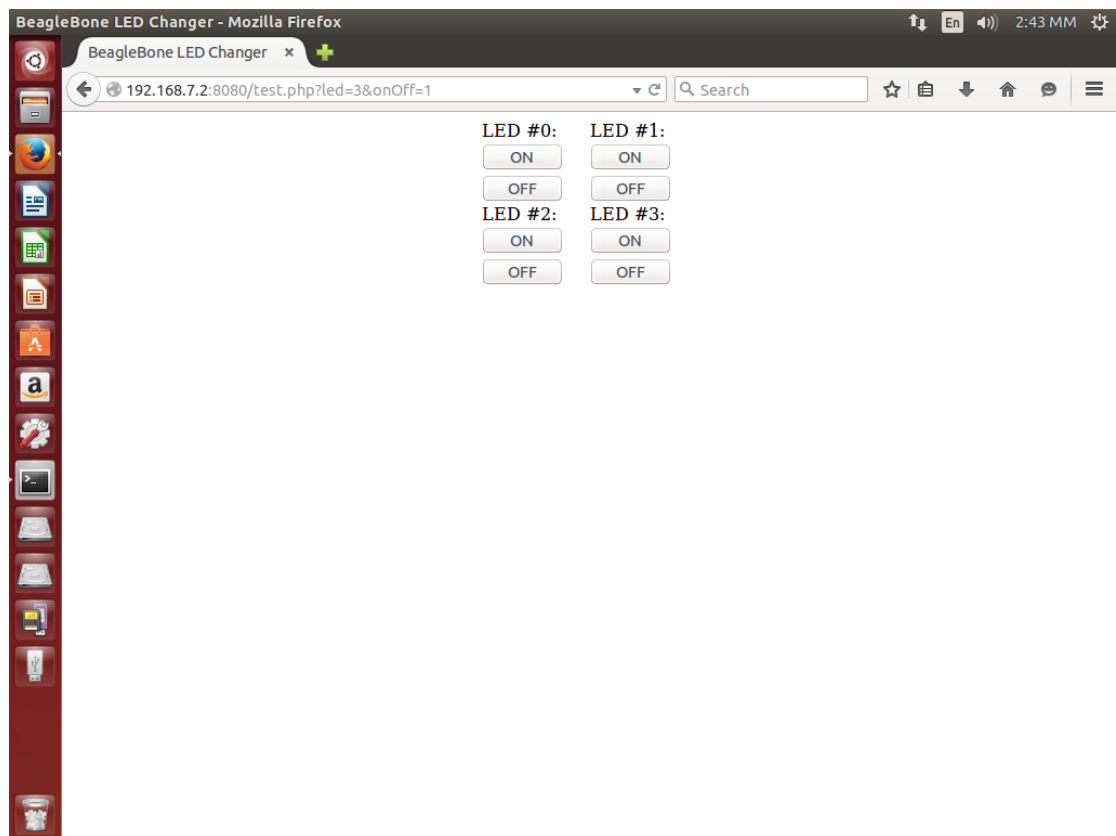
ΕΙΚΟΝΑ 13: Πατημένο το κουμπί LED #1>ON



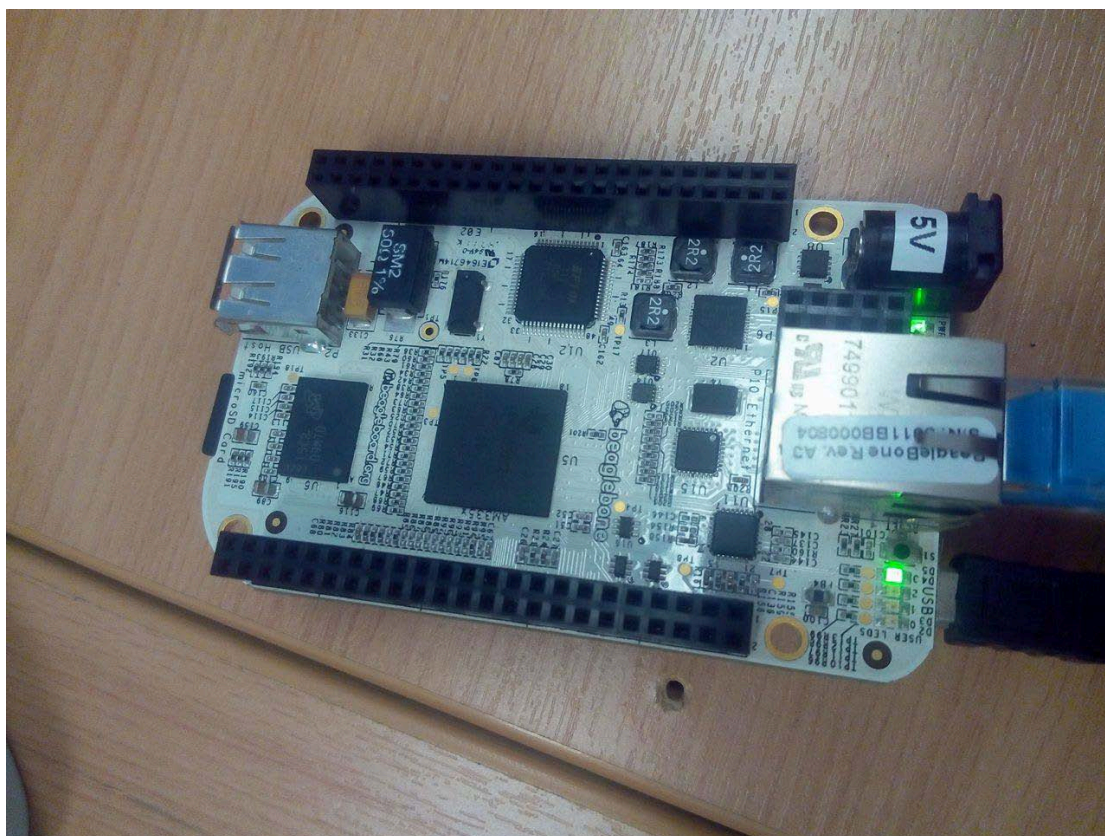
EIKONA 14:Αναμένο το led #1



ΕΙΚΟΝΑ 15: Πατημένο κουμπι LED #2>ON



ΕΙΚΟΝΑ 17:ΠΑΤΗΜΕΝΟ ΚΟΥΜΠΙ LED #3>ON



ΕΙΚΟΝΑ 18:Αναμένο το led#3

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. «<https://el.wikipedia.org/wiki/%CE%9C%CE%B9%CE%BA%CF%81%CE%BF%CE%B5%CE%BB%CE%B5%CE%B3%CE%BA%CF%84%CE%AE%CF%82>»
2. «https://el.wikipedia.org/wiki/%CE%91%CF%81%CF%87%CE%B9%CF%84%CE%B5%CE%BA%CF%84%CE%BF%CE%BD%CE%B9%CE%BA%CE%AE_ARM»
3. «<http://beagleboard.org/bone>»
4. «http://elinux.org/BeagleBone#Manuals_and_resources»
5. «http://elinux.org/BeagleBone_Community»
6. «<https://en.wikipedia.org/wiki/BeagleBoard>»
7. «<https://el.wikipedia.org/wiki/Unix>»
8. «https://el.wikipedia.org/wiki/Berkeley_Software_Distribution»