



**ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ**

---

## **Ενσωματωμένα Συστήματα**

**Ενότητα:** ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ Beagleboard SBC (single board computer)

**Απαιτούμενος Αρ. Εργατωρών:** 16 (προσεγγιστικά)

Δρ. Μηνάς Δασυγένης

[mdasyg@ieee.org](mailto:mdasyg@ieee.org)

με τη συνεργασία των: Ηλιάνα Ποσίκα & Ελένη Σακέρογλου

**Τμήμα Μηχανικών Πληροφορικής και Τηλεπικοινωνιών**

Εργαστήριο Ψηφιακών Συστημάτων και Αρχιτεκτονικής Υπολογιστών

<http://arch.icte.uowm.gr/mdasyg>

---

## Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



## Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ψηφιακά Μαθήματα του Πανεπιστημίου Δυτικής Μακεδονίας**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



## Περιεχόμενα

1.	Σκοπός της άσκησης.....	4
2.	Παραδοτέα .....	4
3.	Γνωριμία με το BeagleBoard-xM .....	5
4.	Βήματα εγκατάστασης Ubuntu στο BeagleBoard-xM.....	7
5.	Εύρεση της IP διεύθυνσης χρησιμοποιώντας DD-WRT + CRON + WGET (Track IP Address using DD-WRT + CRON + WGET) .....	16
6.	Εγκατάσταση της βιβλιοθήκης ανοικτού λογισμικού OpenCV .....	19
7.	Παραδείγματα.....	22
7.1	Επεξεργασία εικόνας .....	22
7.2	Αναγνώριση κύκλων και κόκκινων αντικειμένων .....	23
7.3	Λήψη frame με OpenCV .....	23
7.4	Λήψη live video και αναγνώριση χρωμάτων .....	24
7.5	Λήψη live video και αναγνώριση κυκλικών αντικειμένων με παραμετροποιήσιμο εύρος χρωμάτων .....	24

# 1. Σκοπός της άσκησης

- Γνωριμία με την αναπτυξιακή πλακέτα Beagleboard-xM.
- Βήματα εγκατάστασης λειτουργικού στο Beagleboard-xM.
- Καταγραφή της IP διεύθυνσης σε εξωτερικό αρχείο.
- Εγκατάσταση της βιβλιοθήκης ανοικτού λογισμικού OpenCV.

Ο σκοπός της παρούσας άσκησης είναι η γνωριμία των φοιτητών με την πλακέτα Single Board Computer (SBC) Beagleboard-Xm ( <https://elinux.org/Beagleboard:BeagleBoard-xM> ) που φέρει τον επεξεργαστή ARM, καθώς και με τη βιβλιοθήκη ανοικτού λογισμικού OpenCV.

Μπορείτε να διαβάσετε το επίσημο υποστηρικτικό υλικό για το BB:

<http://beagleboard.org/getting-started>

η τελευταία έκδοση του DataSheet για το BB-xM βρίσκεται online: [http://beagleboard.org/static/BBxMSRM\\_latest.pdf](http://beagleboard.org/static/BBxMSRM_latest.pdf)

Σε αυτή την εργαστηριακή άσκηση, αφού εγκατασταθεί το λειτουργικό σύστημα Ubuntu στην κάρτα SD, στη συνέχεια μέσω της Web κάμερας θα αναγνωρίζονται τα χρώματα των αντικειμένων που διέρχονται, σε πραγματικό χρόνο, με τη χρήση της OpenCV. Τα αντικείμενα θα διέρχονται σε ουδέτερο φόντο και θα καταμετράται ο συνολικός αριθμός τους, ο οποίος θα τυπώνεται από το Beagleboard στην οθόνη. Η εφαρμογή θα είναι παραμετροποιήσιμη ως προς το εύρος του κάθε χρώματος (π.χ. το κόκκινο δεν έχει μόνο μια τιμή RGB και συνεπώς θα πρέπει να δέχεται το εύρος του, π.χ. από 200 έως 255 κ.ο.κ.).

# 2. Παραδοτέα

(A) 14 ερωτήσεις

(C) 8 ασκήσεις

- **Παραδοτέο 1:** το αρχείο **c1.php**
- **Παραδοτέο 2:** το αρχείο **c2.php**
- **Παραδοτέο 3:** το αρχείο **c3 screenshot**
- **Παραδοτέο 4:** το αρχείο **lena\_gray.jpg** και **c4 screenshot**
- **Παραδοτέο 5:** το αρχείο **c5.jpg**
- **Παραδοτέο 6:** το αρχείο **c6.jpg**
- **Παραδοτέο 7:** το αρχείο **c7 screenshot**
- **Παραδοτέο 8:** το αρχείο **c8 screenshot**

### 3. Γνωριμία με το BeagleBoard-xM

Το BeagleBoard-xM (Εικόνα 1) αποτελεί μία χαμηλής κατανάλωσης ενέργειας, ανοιχτού κώδικα υλικού (open source hardware) και ανοιχτού κώδικα λογισμικού (open source software) υπολογιστική πλατφόρμα. Η εταιρία σχεδίασης είναι η Texas Instruments σε συνεργασία με την Digi-Key, ενώ η εταιρία κατασκευής είναι η CircuitCo.



Εικόνα 1: BeagleBoard-xM

Η συσκευασία προϊόντος (με τιμή 150\$ το 2014) περιλαμβάνει τα ακόλουθα:

1. BeagleBoard-xM, εντός αντιστατικού περιτυλίγματος
2. κάρτα τύπου microSD
3. προσαρμογέα κάρτας τύπου microSD σε MultiMediaCard (MMC)

#### **Χαρακτηριστικά του BeagleBoard-xM.**

Ο Πίνακας (Εικόνα 2) παρουσιάζει τα χαρακτηριστικά του BeagleBoard-xM:

	<b>Χαρακτηριστικά</b>	
<b>Επεξεργαστής DM3730</b>	ARM Cortex A8 1GHz	
	DSP TMS320 C64x 800 MHz	
	PowerVR SGX 2D/3D	
<b>Μνήμη POP<sup>1</sup></b>	Micron 4Gb MDDR SDRAM (512MB) 200MHz	
<b>PMIC TPS65950</b>	Ρυθμιστές ισχύος	
	Κωδικοποιητής/Αποκωδικοποιητής ήχου	
	Επανεκκίνηση	
	USB OTG PHY	
<b>Υποστήριξη αποσφαλμάτωσης</b>	JTAG	GPIO
	UART	3 LED
<b>PCB</b>	78,74 x 76,2 mm	6 επίπεδα
<b>Δείκτες</b>	Τροφοδοσίας, Σφάλματος τροφοδοσίας	USR0, USR1
	PMU	Τροφοδοσίας USB
<b>Θύρα HS USB 2.0 OTG</b>	Υποδοχή Mini AB USB	TPS65950 I/F
<b>Θύρες USB Host</b>	SMSC LAN9514 Ethernet HUB	
	4 FS/LS/HS	Έως 500 mA ανά θύρα, με επαρκή τροφοδοσία
<b>Συνδεσιμότητα Audio</b>	3,5 mm	3,5 mm
	Έξοδος L+R	Είσοδος L+R
<b>Συνδεσιμότητα SD/MMC</b>	MicroSD	
<b>Διεπαφή χρήση</b>	Πλήκτρο χρήση	Πλήκτρο επανεκκίνησης
<b>Συνδεσιμότητα Βίντεο</b>	DVI-D	S-Video
<b>Κάμερα</b>	Υποδοχή	Υποστήριξη του αρθρώματος πυρήνα Leopard Imaging
<b>Τροφοδοσία</b>	USB	DC
<b>Προστασία υπερτροφοδοσίας</b>	Τερματισμός λειτουργίας	
<b>Κύρια υποδοχή θυρών επέκτασης</b>	Τροφοδοσία (5 V & 1.8 V)	UART
	McBSP	McSPI
	I2C	GPIO
	MMC2	PWM
<b>2 υποδοχείς LCD</b>	Πρόσβαση σε όλους τους ελέγχους LCD και σε I2C	3.3 V, 5 V, 1.8 V
<b>Συμπλ/τική συνδεσιμότητα</b>	Υποδοχή 4 ακίδων	McBSP2
<b>Συμπλ/τικές θύρες επέκτασης</b>	MMC3	GPIO,ADC,HDQ
<b>Διαστάσεις</b>	82,55 x 82,55 mm	

Εικόνα 2: Πίνακας με τα χαρακτηριστικά του BeagleBoard-xM.

## 4. Βήματα εγκατάστασης Ubuntu στο BeagleBoard-xM

Η εγκατάσταση του λειτουργικού Ubuntu στο Beagleboard-xM είναι πολύ απλή, αλλά μερικές φορές μπορεί μια μικρή λεπτομέρεια να ταλαιπωρήσει πολύ το χρήστη. Δεν είναι τόσο εύκολη όσο μια εγκατάσταση Microsoft Windows, αλλά υπάρχουν αρκετές πληροφορίες (και σε αυτό το εργαστηριακό φυλλάδιο) ώστε να αντιμετωπίσετε όλα τα πιθανά προβλήματα.

Η πλακέτα BeagleBoard-xM είναι ένας πλήρης υπολογιστής, και μπορεί να χρησιμοποιηθεί ως πλήρης σταθμός εργασίας με τη σύνδεση ενός USB πληκτρολογίου, USB mouse και ενός καλωδίου HDMI σε HDMI για σύνδεση σε HDMI οθόνη ή HDMI σε DVI-D για σύνδεση σε DVI οθόνη.

Εντούτοις, ως ενσωματωμένο σύστημα, δε χρειάζεται πληκτρολόγιο, mouse και οθόνη (που αυξάνουν την επιβάρυνση του επεξεργαστή), αφού η πρόσβαση γίνεται απομακρυσμένα μέσω ασφαλούς σύνδεσης SSH σε ένα τυπικό τερματικό bash, μέσω δικτύου είτε ενσύρματου (καλώδιο LAN), είτε ασύρματου (USB wifi κάρτα δικτύου).

Πριν όμως ρυθμίσετε το δίκτυο, θα πρέπει να προβείτε σε κάποιες ρυθμίσεις, οι οποίες είτε γίνονται από USB πληκτρολόγιο και οθόνη ή καλύτερα (και πιο ορθά) μέσω σειριακής σύνδεσης από τον υπολογιστή μας στο BeagleBoard-xM (BBxM) με κατάλληλου καλωδίου USB σε RS232/DB9 (αυτό θα επιλέξουμε σε αυτό το εργαστήριο).

Τα υλικά που χρειάζεστε σε αυτό το εργαστήριο είναι τα εξής:

- 4GB SD card το ελάχιστο και αν έχετε 8GB να την προτιμήσετε
- 4GB USB Flash drive το ελάχιστο και αν έχετε 8GB να το προτιμήσετε (επειδή υπάρχει μεγάλη πιθανότητα το BBxM να μη μπορεί να συνεχίσει στο 4 στάδιο της διαδικασίας boot με την προσάρτηση του root filesystem από τη SD card και τότε θα ορίσουμε το 4 στάδιο να συνεχίσει στο USB Flash drive. Σημείωση: Το πρώτο στάδιο είναι η εκκίνηση από τη ROM του BBxM, η οποία πηγαίνει στο 2<sup>ο</sup> στάδιο που είναι η ανάγνωση του αρχείου **MLO** από τη SDcard που ονομάζεται και X-loader (είναι υποχρεωτική η χρήση SDcard, και το BBxM δεν παρέχει άλλο τρόπο εκκίνησης όπως το συγγενικό board BeagleBone). Το X-loader φορτώνει το 3<sup>ο</sup> στάδιο που είναι ο U-Boot loader και βρίσκεται στο αρχείο **u-boot.img** στη SDcard, ο οποίος στη συνέχεια φορτώνει τον πυρήνα του linux, π.χ. **vmlinux-4.12.5-armv7-x3** (4<sup>ο</sup> στάδιο διαδικασίας boot) και καθορίζει που βρίσκεται το σύστημα αρχείων (root file system).
- Αναπτυξιακή πλακέτα Beagleboard-XM
- SD Card Reader ή Micro-SD USB, ώστε να τη συνδέσετε σε ένα κανονικό σταθμό εργασίας (που ονομάζεται development host ή απλά host) και να αντιγράψετε το λειτουργικό σύστημα του beagleboard.
- Πρόσβαση σε υπολογιστή με λειτουργικό Linux (προτιμάται Ubuntu ή Debian, αλλά δεν είναι υποχρεωτικό).
- HDMI Cable (προαιρετικά, δε θα το χρειαστείτε υπό κανονικές συνθήκες)

- HDMI Monitor (προαιρετικά, δε θα το χρειαστείτε υπό κανονικές συνθήκες)
- Ethernet σύνδεση με το Beagleboard (ή καλύτερα ένα υποστηριζόμενο USB wifi περιφερειακό για την ασύρματη σύνδεση στο δίκτυο, όπως θα δείξουμε σε αυτό το φυλλάδιο)
- Μια πηγή ενέργειας που παρέχει 5 volts και 0.7 Amps
- Μια υποστηριζόμενη από το linux USB webcam.
- Ένα καλώδιο USB σε RS232 για τη σύνδεση στη κονσόλα του BBxM και τοποθέτηση των κατάλληλων ρυθμίσεων.

Για να είστε σίγουροι ότι όλα τα πακέτα που χρειάζονται είναι εγκατεστημένα στον υπολογιστή Linux Host που χρησιμοποιείτε, ανοίξτε ένα τερματικό και πληκτρολογήστε:

```
sudo apt-get install wget pv dosfstools parted
```

Για την εγκατάσταση του λειτουργικού συστήματος, θα χρησιμοποιηθεί ένα pre-built πακέτο για το beagleboard των Ubuntu. Επειδή το πακέτο αυτό ενημερώνεται αρκετά συχνά, θα πρέπει να βρείτε την τελευταία έκδοση του με την εξής διαδικασία: Μεταβείτε στην τοποθεσία <https://rcn-ee.com/rootfs/> και εισέλθετε σε έναν κατάλογο που φέρει ημερομηνία, π.χ. **2019-10-09** . Στη συνέχεια εισέλθετε στον υποκατάλογο elinux (δηλαδή, embedded linux). Εκεί θα βρείτε κάποια αρχεία που φέρουν ως πρώτη λέξη το Ubuntu ή το debian, την έκδοση του ΛΣ, τη λέξη console (ότι τα prebuilt images δεν έχουν γραφικό περιβάλλον και είναι πιο μικρά και ελαφριά, αν και μπορούμε να εγκαταστήσουμε στη συνέχεια κάποιο γραφικό περιβάλλον), τη λέξη armhf (ότι αφορούν επεξεργαστή arm με hardware floating point unit και όχι εξομοίωση πράξεων κινητής υποδιαστολής), την ημερομηνία δημιουργίας και την κατάληξη **.xz** ή **.sha256sum** (για την επιβεβαίωση της ψηφιακής υπογραφής). Για παράδειγμα, ένα τέτοιο αρχείο εικόνας linux είναι: **ubuntu-14.04.3-console-armhf-2015-10-09.tar.xz**

Μεταφορτώστε και τα 2 αρχεία (**.xz** , **.sha256sum**) στον υπολογιστή σας.

Για να επιβεβαιώσετε την ψηφιακή υπογραφή εκτελείτε σε Linux το πρόγραμμα **sha256sum** με παράμετρο το image .xz και συγκρίνετε τον αριθμό που υπολογίζετε με τον αριθμό που αναγράφεται μέσα στο αρχείο. Οι 2 αριθμοί θα πρέπει να είναι 100% ίδιοι.

Στη συνέχεια πρέπει να αποσυμπιέσετε το αρχείο. Η αποσυμπίεση γίνεται με το πρόγραμμα **tar** , αφού ανοίξετε ένα τερματικό και μεταβείτε στον φάκελο που βρίσκονται τα αρχεία που μεταφορτώσατε.

Δώστε την πλήρη εντολή για την αποσυμπίεση του αρχείου:  


---

**(A1)**

Αφού ολοκληρωθεί η αποσυμπίεση μπειτε στον φάκελο που δημιουργήθηκε με την κατάλληλη εντολή **cd**. Μετά πρέπει να βρείτε που είναι αντιστοιχισμένη η SD card. Τοποθετήστε την κάρτα SD στο USB SD card reader και στη συνέχεια συνδέστε το καλώδιο στον υπολογιστή. Η εντολή που θα σας φανεί χρήσιμη είναι η **dmesg**.



Δώστε την εντολή που πρέπει να εκτελεστεί για να γίνει το παραπάνω:  
**(A2)**

Αν η κάρτα τοποθετηθεί σε linux που αυτόματα προσαρτά τις SD κάρτες, τότε αν δώσετε `mount` θα δείτε ότι θα έχει προσαρτηθεί στο ΛΣ, όπως παρακάτω (για τη `/dev/sdb` κάρτα):

Filesystem	Size	Used	Avail	Use%	Mounted on
<code>/dev/sda3</code>	20G	16G	2.9G	85%	<code>/</code>
<code>udev</code>	2.9G	4.0K	2.9G	1%	<code>/dev</code>
<code>tmpfs</code>	1.2G	1016K	1.2G	1%	<code>/run</code>
<code>none</code>	5.0M	0	5.0M	0%	<code>/run/lock</code>
<code>none</code>	2.9G	80K	2.9G	1%	<code>/run/shm</code>
<code>none</code>	100M	56K	100M	1%	<code>/run/user</code>
<code>/dev/sdb2</code>	3.6G	2.5G	987M	72%	<code>/media/lion/rootfs</code>
<code>/dev/sdb1</code>	64M	11M	54M	17%	<code>/media/lion/boot</code>

Όπως βλέπετε, εδώ η SD card είναι τοποθετημένη στις θέσεις `/dev/sdb1` και `sdb2`, γιατί έχει 2 partitions. Ένα FAT partition για τα πρώτα στάδια της εκκίνησης και ένα ext3/ext4 για το σύστημα αρχείων.

Δώστε το screenshot που δείχνει αν έχει προσαρτηθεί ή όχι η SD card **(A3)**

Από-προσαρτήστε την κάρτα αν είναι προσαρτημένη με την εντολή `umount`.

Εισέλθετε στον κατάλογο που έχετε αποσυμπιέσει το image του Linux για το BBxM. Τώρα πληκτρολογήστε στο τερματικό (αν η κάρτα σας είναι στο `/dev/sdb`):

```
sudo ./setup_sdcard.sh --mmc /dev/sdb --dtb omap3-beagle-xm
```

Αν η SD card είναι τοποθετημένη στη διαδρομή `/dev/sdc`, για παράδειγμα, πρέπει να αλλάξετε την παραπάνω εντολή. Προσοχή μη δώσετε ένα device που αντιστοιχεί σε σκληρό δίσκο και σβηστούν όλα τα περιεχόμενα του σκληρού δίσκου του υπολογιστή host που χρησιμοποιείτε.

Μετά την εντολή αυτή, μπορεί να ερωτηθείτε αν επιλέξατε το σωστό δίσκο. Το script θα συνεχίσει να εκτελείται διαμορφώνοντας την κάρτα για την εκκίνηση του Beagleboard. Αυτό μπορεί να πάρει κάποια λεπτά. Όταν τελειώσει πρέπει να εμφανιστεί το παρακάτω:

```
Finished populating rootfs Partition
-----
setup_sdcard.sh script complete
-----
The default user:password for this image:
ubuntu:temprrwd
```

Δώστε το screenshot που δείχνει ότι η διαδικασία έχει ολοκληρωθεί (όπως το παραπάνω) **(A4)**

Κάντε ασφαλή κατάργηση υλικού και στη συνέχεια εξάγεται την κάρτα SD.

Σε αυτό το σημείο, **συνιστάται να τοποθετήσετε στον Host υπολογιστή ένα USB flash drive** (που θα διαγραφεί στη διαδικασία), και να επαναλάβετε τα προηγούμενα βήματα, ώστε το ΛΣ να βρίσκεται και στη SD card και στο USB Flash Drive (η διαδικασία είναι ακριβώς η ίδια).

Το επόμενο βήμα είναι η τοποθέτηση του σειριακού καλωδίου στο board και η σύνδεση με υπολογιστή. Να σημειώσετε ότι επειδή συνδέεστε στην κεντρική κονσόλα του BBxM θα σας εμφανίζονται διάφορα μηνύματα, ακόμη και ενώ γράφετε κάποια εντολή. Με το συνδυασμό πλήκτρων **control+1** (-ελ) θα καθαρίζουν αυτά τα μηνύματα (αλλά θα παραμένουν αυτά που είχατε γράψει ως εκείνο το σημείο).

Για να βρείτε το device που έχει συνδεθεί με το σειριακό καλώδιο (αν χρησιμοποιείτε USB serial cable), δώστε **dmesg** αμέσως μόλις συνδέσετε το καλώδιο και θα δείτε στις τελευταίες γραμμές το όνομα, π.χ. **tttyUSB0**

- Αν είστε σε σύστημα Microsoft Windows, μπορείτε να χρησιμοποιήσετε το putty με επιλογή το Serial και ως θύρα, τη θύρα που θα εμφανιστεί στο device manager για το καλώδιο USB-RS232 που θα προσθέσετε. Η πρώτη παράμετρος είναι η ρύθμιση της ταχύτητας στα 115200 bps που είναι η προεπιλογή για το BBxM. Μια ακόμη σημαντική παράμετρος στο serial είναι η επιλογή **None** στις ρυθμίσεις serial για το **Hardware Flow Control**. Σε περίπτωση που δεν απενεργοποιήσετε το Hardware Flow Control θα βλέπετε τα μηνύματα μόνο και δε θα μπορείτε να στείλετε κάποιο πλήκτρο.
- Αν είστε σε σύστημα Linux, τότε μπορείτε να χρησιμοποιήσετε είτε το **cu**, είτε το **minicom**, είτε το **screen**. Αν είστε σε Linux θα πρέπει να έχετε δικαιώματα χρήσης της θύρας /dev/ttyUSB0 (π.χ. να ανήκετε στο group dialout). Μπορείτε να δώστε ως root **chmod 777 /dev/ttyUSB0** για να μη σας εμφανιστεί κάποιο μήνυμα permission denied αν δεν είστε στο κατάλληλο group ή να επεξεργαστείτε το /etc/group και να τοποθετήσετε το username σας στο group dialout. Η χρήση των προγραμμάτων σειριακής σύνδεσης είναι ως εξής:
  - Για το screen η σύνταξη είναι **screen /dev/ttyUSB0 115200** [το device name το βρίσκετε με dmesg]. Για να βγείτε από το screen δίνετε **control+a** και μετά το \

- Για το minicom, αφού εκτελέσετε `minicom` θα πρέπει να πατήσετε `control+a` και μετά `z`. Εμφανίζεται ένα menu που πατώντας ένα πλήκτρο επιλέγετε το υπομενού. Οι ρυθμίσεις που μας ενδιαφέρουν βρίσκονται στο Comm Parameters, και μετά στο Configure Minicom και Serial Port Setup. Θα πρέπει να δώσετε 115200, 8bits, No parity, Stop bits 1 (8-N-1), και Hardware Flow Control None. Για να εγκαταλείψετε το minicom, δίνετε `control+a` και μετά `q`.
- Για το cu, θα πρέπει να εκτελέσετε `cu -l /dev/ttyUSB0 -s 115200`, και από ένα άλλο τερματικό του Host υπολογιστή να απενεργοποιήσετε το hardware flow control με την εντολή `stty -F /dev/ttyUSB0 -crtcts`. Εναλλακτικά, μπορείτε να δημιουργήσετε ένα αρχείο `/etc/uucp/port` και να τοποθετήσετε τα παρακάτω:
 

```
port ttyUSB0
type direct
device /dev/ttyUSB0
hardflow false
speed 115200
```

 Και να συνδεθείτε με `cu -I /etc/uucp/port -l /dev/ttyUSB0`  
 Για να βγείτε από το cu, πατάτε `enter ~ control+d enter`.

Σημειώστε ότι όταν ξεκινάει το σύστημα μπορείτε να πατήσετε ESC στην κατάλληλη προτροπή για να διακόψετε το 2 στάδιο εκκίνησης (u-boot), και στην προτροπή να δώσετε εντολές όπως `help`, `printenv`, `setenv` για να αλλάξετε τις ρυθμίσεις του φορτωτή ([http://wiki.dave.eu/index.php/Change\\_Linux\\_Command\\_Line\\_Parameter\\_from\\_U-boot](http://wiki.dave.eu/index.php/Change_Linux_Command_Line_Parameter_from_U-boot)).

Εναλλακτικά, αν δεν έχετε σειριακό καλώδιο, αλλά έχετε USB πληκτρολόγιο και HDMI οθόνη, προβείτε σε ρυθμίσεις μέσω πληκτρολογίου στο BBxM.

Όσο το Beagleboard κάνει boot, το LED D6 στην πλακέτα πρέπει να αναβοσβήνει (σαν καρδιακός παλμός), ενώ το D7 πρέπει να αναβοσβήνει ασταθώς.

Ένα συνηθισμένο πρόβλημα στα BBxM κατά το boot, είναι η εμφάνιση του μηνύματος:

```
mmc0: error -110 whilst initialising SD card
```

Το οποίο σημαίνει ότι λόγω κακής σχεδίασης του υποσυστήματος SD δε μπόρεσε να διαβάσει στο τελευταίο στάδιο boot το filesystem. Αν έχετε μόνο τη SD card, τότε σε αυτό το σημείο σταματάει η διαδικασία boot και δε μπορείτε να κάνετε τίποτα άλλο παρά `reset`. Αν δείτε ότι το πρόβλημα συνεχίζει μετά από 5-6 reboot, τότε μια εναλλακτική διαδικασία boot, είναι η χρήση του USB flash drive ως εξής:

- Συνδέετε το USB flash drive που έχετε δημιουργήσει προηγουμένως στο BBxM
- Απομακρύνετε τη SD card από το BBxM και την προσαρτάτε σε ένα σύστημα Linux για να ρυθμίσετε κατάλληλα την εκκίνηση. Η προσαρτήση γίνεται με την εντολή `mount`, π.χ. αν η κάρτα έχει το `/dev/sdb`, τότε προσαρτάται το πρώτο

partition με τις ρυθμίσεις εκκίνησης σε έναν υποκατάλογο /mnt όπως παρακάτω: `mount /dev/sdb1 /mnt .`

- Δημιουργείτε στο /mnt ή επεξεργάζεστε το αρχείο `uEnv.txt` και τοποθετείτε τις παρακάτω γραμμές που ορίζουν ότι το τελευταίο στάδιο της εκκίνησης, θα γίνει με το USB flash drive αφού (φορτωθεί το υποσύστημα usb), που θα είναι το /dev/sda. Το αρχείο τοποθετείται στο πρώτο partition FAT της SD κάρτας.

```
uenvcmd=usb start;
bootpart=0:1
bootdir=/
loaduimage=load usb ${bootpart} 0x80200000 /boot/uImage
mmcroot=/dev/sda1 ro rootwait verbose
mmcboot=echo Booting from USB ...; run mmcargs; bootm 0x80200000 - 0x815f0000
uname_r=4.12.5-armv7-x3
cmdline=coherent_pool=1M net.ifnames=0 verbose root=/dev/sda2
```

- Απομακρύνετε την κάρτα από τον υπολογιστή αφού την αποπροσαρτήσετε με `umount /mnt` και την τοποθετείτε στο BBxM.

Ένα άλλο πρόβλημα που ίσως αντιμετωπίσετε, είναι ότι η οθόνη που έχει συνδεθεί με HDMI (μόνο αν δεν χρησιμοποιείτε σειριακή σύνδεση) δεν εμφανίζει τίποτα κατά το πρώτο boot. Για να λύσετε το πρόβλημα, απενεργοποιείτε το Beagleboard, και αφαιρέστε την κάρτα SD (αν δεν τα κάνετε με αυτή τη σειρά μπορεί να καταστραφεί η κάρτα SD). Εισάγετε ξανά την κάρτα στον υπολογιστή και στο "boot" partition θα βρείτε ένα αρχείο με το όνομα uEnv.txt. Αν δεν υπάρχει δημιουργήστε το με την κατάλληλη ανάλυση οθόνης (πχ 1920x1080) (και προσαρτήστε στο τέλος του αρχείου τις παραπάνω ρυθμίσεις του uEnv.txt αν θέλετε να γίνει boot μέσω USB flash drive):

```
kernel_file=zImage
initrd_file=initrd.img

#Video: Uncomment to override U-Boots value:
defaultdisplay=dvi
dvmode="omapfb.mode=dvi:1920x1080"
vram=12MB

#Camera: Uncomment to enable:
#http://shop.leopardimaging.com/product.sc?productId=17
#camera=li5m03

#SPI: enable for userspace spi access on expansion header
#buddy=spidev

#LSR COM6L Adapter Board
#http://eewiki.net/display/linuxonarm/LSR+COM6L+Adapter+Board
#First production run has unprogrammed eeprom:
#buddy=lsr-com6l-adpt

#LSR COM6L Adapter Board + TiWi5
#wl12xx_clk=wl12xx_26mhz

console=ttyO2,115200n8

mmcroot=/dev/mmcblk0p2 ro
mmcrootfstype=ext4 rootwait fixrtc

boot_fstype=fat
xyz_load_image=${boot_fstype}load mmc 0:1 0x80300000 ${kernel_file}
xyz_load_initrd=${boot_fstype}load mmc 0:1 0x81600000 ${initrd_file}; setenv initrd_size ${filesize}
xyz_load_dtb=${boot_fstype}load mmc 0:1 0x815f0000 /dtbs/${dtb_file}

xyz_mmcboot=run xyz_load_image; run xyz_load_initrd; echo Booting from mmc ...

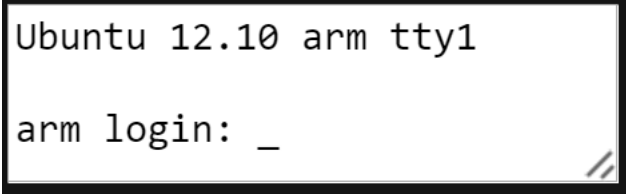
video_args=setenv video vram=${vram} omapfb.mode=${defaultdisplay}:${dvmode} omapdss.def_disp=${defaultdisplay}
device_args=run video_args; run expansion_args; run mmcargs
mmcargs=setenv bootargs console=${console} ${optargs} ${video} root=${mmcroot} rootfstype=${mmcrootfstype}
${expansion}
```

```
optargs=console=tty0
expansion_args=setenv expansion buddy=${buddy} buddy2=${buddy2} camera=${camera} w12xx_clk=${w12xx_clk}
loaduimage=run xyz_mmcbboot; run device_args; bootz 0x80300000 0x81600000:${initrd_size}
```

Αποθηκεύστε το αρχείο, κάντε ασφαλή κατάργηση, εισάγετε την κάρτα SD στο Beagleboard και ενεργοποιείστε το.

---

Η οθόνη πρέπει να ανοίξει και να εκτελείται το boot. Στη συνέχεια θα εμφανιστεί μια προτροπή, όπως η παρακάτω:



```
Ubuntu 12.10 arm tty1
arm login: _
```

Συνδεθείτε με το όνομα χρήστη και τον κωδικό πρόσβασης που σας έχει δοθεί παραπάνω.

Δώστε το όνομα χρήστη και τον κωδικό πρόσβασης που χρησιμοποιήσατε.  
\_\_\_\_\_ **(A5)**

Το τελευταίο τμήμα αρχικών ρυθμίσεων αφορά τη σύνδεση με το διαδίκτυο. Οι ρυθμίσεις τοποθετούνται με τον ίδιο τρόπο με ένα Linux μηχάνημα.

- Αν έχετε καλώδιο LAN συνδέστε το στο board και στην κατάλληλη πρίζα, δείτε με την εντολή `ifconfig` ποια είναι η κάρτα δικτύου (συνήθως η `eth0`), και δώστε `dhclient eth0` (για να λάβει μια αυτόματη διεύθυνση IP από το DHCP server του δικτύου, αν υπάρχει).

## Σύνδεση στο διαδίκτυο

Αν έχετε τοποθετήσει USB wifi adaptor, δείτε αν αναγνωρίζεται με `iwconfig`. Θα πρέπει να εμφανιστεί κάτι παρόμοιο με το παρακάτω:

```
wlan0      IEEE 802.11  ESSID:off/any
           Mode:Managed  Access Point: Not-Associated  Tx-Power=20 dBm
           Retry short limit:7   RTS thr=2347 B   Fragment thr:off
           Encryption key:off
           Power Management:off
```

Αν σας αναφέρει `wlan0 IEEE 802.11` ή κάτι παρόμοιο τότε έχει υποστηρίξει η USB κάρτα wifi. Υποθέτουμε ότι είναι το `wlan0` το όνομα της κάρτας σας. Αν δεν υπάρχει η USB wifi στο σύστημά, σας βρείτε στο διαδίκτυο τι απαιτείται για να υποστηριχθεί (εκτός της ύλης αυτού του εργαστηρίου). Ενεργοποιήστε το ασύρματο interface με `ifconfig wlan0 up` ή αν χρησιμοποιείτε ενσύρματο σύνδεση με `ifconfig eth0 up`.

Στη συνέχεια βρείτε τα SSID που υπάρχουν με: `iwlist wlan0 scan | grep ESSID`

Για παράδειγμα, στο εργαστήριο του διδάσκοντα, υπάρχει το

**ESSID:"UOWM-K7-WIFI"**

Το επόμενο βήμα είναι η επεξεργασία του αρχείου `/etc/network/interfaces` με το `nano`:

**nano /etc/network/interfaces**

Προσθέστε τις παρακάτω γραμμές στο τέλος (αν υπάρχουν και άλλες καταχωρήσεις για `wlan0` να τις απομακρύνετε). Να αλλάξετε το `XXXXXX` με τον κωδικό του δικτύου όπως και το SSID.

```
auto wlan0
iface wlan0 inet dhcp
wpa-psk XXXXXXXXXXXXXXXX
wpa-key-mgmt WPA-PSK
wpa-proto WPA2
wpa-ssid UOWM-K7-WIFI
pre-up sleep 5
```

Επειδή μάλλον θα εκτελείται το `wpa_supplicant` που θα δημιουργήσει προβλήματα, τερματίστε το ως εξής:

```
kill -9 `ps auxww | grep wpa_supplicant |awk '{print $2}'`
```

Αν έχουν γίνει όλα σωστά τότε απενεργοποιήστε τη διεπαφή και ενεργοποιήστε την πάλι προκειμένου να συνδεθεί στο δίκτυο και να λάβει την IP.

```
ifdown wlan0
```

```
ifup -v wlan0
```

Επιβεβαιώστε ότι έχει λάβει IP η wifi κάρτα σας με:

```
ifconfig wlan0
```

και ότι μπορείτε να κάνετε ένα ping (control + c , σταματάει το ping)

```
ping 83.212.19.218
```

όπως και DNS resolve

```
ping zafora.ict.e.uowm.gr
```

Προχωρήστε στα επόμενα βήματα μόνο αν έχετε ρυθμίσει το δίκτυο.

Ακολουθούν κάποιες επιπλέον εγκαταστάσεις πάνω στο BeagleM που έχει ξεκινήσει με επιτυχία.:

- `sudo apt-get update`
- `sudo apt-get install ssh`
- `sudo apt-get install man info`
- `sudo apt-get install libwebcam0`
- `sudo apt-get install uvcdynctrl`
- `sudo apt-get install streamer`

Στο τέλος, συνδέστε την web cam στο Beagleboard και ελέγξτε αν έχει αναγνωριστεί:

```
ls /dev/ | grep video
```

Δώστε το screenshot που δείχνει το αποτέλεσμα της παραπάνω εντολής. **(A6)**

Δώστε την εντολή που εμφανίζει όλες τις συνδεδεμένες συσκευές usb.  
\_\_\_\_\_ **(A7)**

Στη συνέχεια δώστε τις παρακάτω εντολές στο τερματικό:

- `sudo mkdir /tmp/ramdisk`
- `sudo chmod 777 /tmp/ramdisk`
- `sudo mount -t tmpfs -o size=64M tmpfs /tmp/ramdisk`

Τι κάνουν οι παραπάνω εντολές; \_\_\_\_\_ **(A8)**

Ο πρώτος έλεγχος της κάμερας μπορεί να γίνει με την εντολή:

```
streamer -c/dev/video1 -s 1920x1080 -j 100 -o  
/tmp/ramdisk/picture.jpeg
```

η οποία τραβάει ένα frame με την κάμερα και το αποθηκεύει σε μορφή jpeg.

Να μεταφέρετε την εικόνα picture.jpeg μέσω secure copy (scp). Αν χρησιμοποιείτε linux να ανοίξετε ένα τερματικό και να δώσετε:

```
scp ubuntu@IP:/tmp/ramdisk/picture.jpeg /tmp
```

Και μετά `feh /tmp/picture.jpeg`

Αν χρησιμοποιείτε Microsoft Windows, το πρόγραμμα winscp ή FileZilla θα σας φανεί χρήσιμο.

Τι κάνουν οι παράμετροι της παραπάνω εντολής; \_\_\_\_\_ **(A9)**

**\*\*Προσοχή:** Είναι καλό πάντα, μόλις συνδέεστε στο Beagleboard να εκτελείτε την εντολή `sudo bash` για να έχετε όλα τα δικαιώματα διαχειριστή. Επίσης, μπορείτε να προβάλετε ή να μεταφέρετε αρχεία από και προς το Beagleboard μέσω του προγράμματος WinSCP ή scp.

Τέλος Α' μέρους

---

## 5. Εύρεση της IP διεύθυνσης χρησιμοποιώντας DD-WRT + CRON + WGET (Track IP Address using DD-WRT + CRON + WGET) – ΠΡΟΑΙΡΕΤΙΚΑ για πιο προχωρημένους χρήστες στο Linux

Ύστερα από αρκετές φορές χρήσης του BeagleBoard-xM, προέκυψε η ανάγκη μιας πιο εύκολης πρόσβασης σε αυτό. Η διαδικασία σύνδεσης του BeagleBoard σε μια επιπλέον οθόνη μπορεί πολύ απλά να αντικατασταθεί μέσω ενός σεναρίου (script), το οποίο τρέχει στο BeagleBoard-xM και στέλνει αιτήσεις HTTP σε έναν server (πχ. okeano ή zafora). Έτσι, μόλις το BeagleBoard συνδεθεί στο διαδίκτυο, στέλνει την IP του στον server, ο οποίος την καταγράφει σε ένα αρχείο. Έτσι λοιπόν, μέσω σύνδεσης με ssh από το putty μπορούμε εύκολα να συνδεθούμε στο BeagleBoard.

**ΣΗΜΕΙΩΣΗ:** Αυτό το βήμα μπορεί να μη γίνει, αν δεν πρόκειται να χρησιμοποιηθεί το BB-xm ως ένας αυτόνομος σταθμός εργασίας που θα είναι συνεχώς ανοιχτός και ενδέχεται να αλλάζει IP κατά διαστήματα.

**Διαδικασία:**

### **Βήμα 1° ( ενεργοποίηση της περιοδικής εκτέλεσης της ανανέωσης της IP μέσω crontab στο beagleboard).**

Επεξεργασία του αρχείου Crontab (αρχείο που καθορίζει τις περιοδικές εργασίες του συστήματος). Δώστε την παρακάτω εντολή στο τερματικό του Beagleboard για να επεξεργαστείτε το αρχείο του crontab (ίσως ερωτηθείτε να επιλέξετε έναν επεξεργαστή κειμένου).

`crontab -e`



Στη συνέχεια αντιγράψτε το παρακάτω script και επικολλήστε το στο αρχείο που εμφανίστηκε με την παραπάνω εντολή που ξεκινάει από \*5 για να εκτελείται κάθε 5 λεπτά. Να αντικαταστήσετε το **eth0** με **wlan0** αν έχετε συνδέσει το board με ενσύρματο τρόπο στο δίκτυο. Για να καταλάβετε τι κάνει το script εκτελέστε διαδοδικά (και με προσοχή για τη σωστή σύνταξη) τις παρακάτω εντολές που συνθέτουν το ολοκληρωμένο script:

```
/sbin/ifconfig eth0
```

```
/sbin/ifconfig eth0 | /bin/grep -v inet6
```

```
/sbin/ifconfig eth0 | /bin/grep -v inet6 | /bin/grep inet
```

```
/sbin/ifconfig eth0 | /bin/grep -v inet6 | /bin/grep  
inet | /usr/bin/tr -s ' '
```

```
/sbin/ifconfig eth0 | /bin/grep -v inet6 | /bin/grep  
inet | /usr/bin/tr -s ' ' | /usr/bin/cut -f 2 -d ":"
```

```
/sbin/ifconfig eth0 | /bin/grep -v inet6 | /bin/grep  
inet | /usr/bin/tr -s ' ' | /usr/bin/cut -f 2 -  
d ":" | /usr/bin/cut -f 1 -d " "
```

\*σημείωση. Αν χρησιμοποιήσετε **wlan0** και όχι **eth0** τότε θα πρέπει το τελευταίο τμήμα της τελευταίας εντολής να γίνει `/usr/bin/cut -f 3 -d " "`

Η ολοκληρωμένη γραμμή λοιπόν που θα τοποθετήσετε στο crontab για να εκτελείται κάθε 30 λεπτά είναι:

```
*/30 * * * * /usr/bin/wget -O /dev/null -q  
http://zafora.ict.e.uowm.gr/~myusername/ping.php?mitsos=`/  
sbin/ifconfig wlan0 | /bin/grep -v inet6 | /bin/grep inet  
| /usr/bin/tr -s ' ' | /usr/bin/cut -f 2 -d ":" |  
/usr/bin/cut -f 1 -d " "` -qo /dev/null
```

**\*\*Προσοχή:** Στην εντολή wget θέλει το πλήρες path σε κάθε εντολή εσωτερικά γιατί αλλιώς το Crontab δεν λειτουργεί.

Κάνοντας `whereis + όνομα_εντολής` βρίσκουμε τη διαδρομή που βρίσκεται αυτό που ψάχνουμε, πχ `whereis wget`.

Με ποια παράμετρο μπορούμε να προβάλλουμε τις καταχωρήσεις του αρχείου Crontab; \_\_\_\_\_(A10)

Τι κάνει η παράμετρος `-q` στην παραπάνω εντολή; \_\_\_\_\_(A11)

Τι κάνει η παράμετρος -qo /dev/null στην παραπάνω εντολή; \_\_\_\_\_(A12)

**Βήμα 2<sup>ο</sup>:**(δημιουργία κατάλληλων αρχείων σε web server της επιλογής μας, π.χ. zafora.ict.e.uowm.gr)

Δημιουργήστε τα δύο αρχεία στον server σας (zafora ή okeano), όπως παρακάτω (C1) + (C2):

**pong.php** (C1)

```
<?php
$ip = $_REQUEST['mitsos'];
$handle = fopen("./ip.txt", 'w');
fwrite($handle, $ip);
fclose($handle);
echo $ip;
?>
```

**pong.php** (C2)

```
<?php
echo file_get_contents("ip.txt");
?>
```

**\*\*Προσοχή:** Η παραπάνω διαδικασία γίνεται είτε στον server του zafora.ict.e.uowm.gr δημιουργώντας τον κατάλογο public\_html (δείτε οδηγίες εδώ: <https://zafora.ict.e.uowm.gr/php-mysql-zafora.pdf> ), είτε σε οποιοδήποτε άλλο server πχ okeanos .grnet.gr.

Μετά από τα παραπάνω, μπορείτε πολύ απλά να περιηγηθείτε στον server σας πχ. <http://snf-xxx.vm.okeanos.grnet.gr/ip.txt> ή <http://zafora.ict.e.uowm.gr/~myusername/ip.txt> και να δείτε την τελευταία γνωστή διεύθυνση IP του BeagleBoard σας. Αντίστοιχα για τον zafora, συνδέεστε με putty και ανοίγετε το αρχείο ip.txt.

---

## B' ΜΕΡΟΣ

### 6. Εγκατάσταση της βιβλιοθήκης ανοικτού λογισμικού OpenCV

Η OpenCV είναι βιβλιοθήκη με έτοιμες συναρτήσεις που αποσκοπούν στην εφαρμογή συστημάτων Τεχνητής όρασης (Computer Vision) και Ψηφιακής επεξεργασίας εικόνας (Image processing). Η OpenCV θεωρείται απαραίτητη γνώση σε επαγγελματίες ή ακαδημαϊκούς για εφαρμογές όπως ρομποτική, εφαρμογές για smartphones, επαυξημένη πραγματικότητα, και άλλες. Υποστηρίζει Windows, Linux, Mac OS, iOS και Android και οι γλώσσες συγγραφής κώδικα που μπορούν να χρησιμοποιηθούν είναι η C++, η C, η Python και ηJava.

#### Διαδικασία εγκατάστασης OpenCV

**\*\*Προσοχή:** Το παρακάτω εγχειρίδιο έχει δοκιμαστεί σε Ubuntu 14.04.3, και η έκδοση που θα εγκατασταθεί είναι η OpenCV 3.0.0.

**\*\*Σημείωση1:** Η συγκεκριμένη άσκηση έχει εκτελεστεί και με επιτυχία στο OpenCV 2.4 με τα πακέτα που έχει το Ubuntu 16.04 χωρίς να χρειαστεί η μεταγλώττιση της τελευταίας έκδοσης του OpenCV. Αν όμως κάποιος θελήσει πιο προχωρημένες λειτουργίες του OpenCV που δε βρίσκονται στην 2.4 τότε θα πρέπει να κατεβάσει τον πηγαίο κώδικα της τελευταίας έκδοσης για να το κάνει compile. Τα πακέτα που χρησιμοποιήθηκαν είναι: `python-opencv opencv-data libcv2.4 libcv-dev libopencv-core2.4v5 libopencv-dev` .

**\*\*Σημείωση2:** Κάποια πακέτα ενδέχεται να έχουν αλλάξει όνομα ή έκδοση. Αν συναντήσετε κάποιο πρόβλημα ο διδάσκοντας θα σας βοηθήσει.

Για να εγκαταστήσετε την τελευταία έκδοση του OpenCV στο BeagleBoard επιβεβαιώστε ότι έχετε αφαιρέσει τη βιβλιοθήκη από το αποθετήριο με:

```
sudo apt-get autoremove libopencv-dev python-OpenCV
```

και ακολουθήστε τα παρακάτω βήματα.

Αν θέλετε να κάνετε τα παρακάτω βήματα αυτόματα, μπορείτε να εκτελέσετε αυτό το [σενάριο \(script\)](#) με bash.

#### 1. Εγκατάσταση του zip/unzip

```
sudo apt-get install zip unzip
```

Με ποια εντολή γίνεται unzip ένα αρχείο; \_\_\_\_\_ (A13)

## 2. Έλεγχος ενημερώσεων για το Ubuntu

Ανοίξτε τερματικό σας και εκτελέστε:

```
sudo apt-get update && sudo apt-get upgrade && sudo  
apt-get dist-upgrade && sudo apt-get autoremove
```

## 3. Εγκατάσταση βασικών βιβλιοθηκών

### Build tools:

```
sudo apt-get install build-essential cmake
```

### GUI:

Επειδή δε χρησιμοποιείται γραφικό περιβάλλον, εκτελέστε την παρακάτω εντολή:

```
sudo apt-get install libqt4-dev pkg-config
```

### Media I/O:

```
sudo apt-get install zlib1g-dev libjpeg-dev libwebp-  
dev libpng-dev libtiff5-dev libjasper-dev libopenexr-  
dev libgdal-dev
```

### Video I/O:

```
sudo apt-get install libdc1394-22-dev libavcodec-dev  
libavformat-dev libswscale-dev libtheora-dev  
libvorbis-dev libxvidcore-dev libx264-dev yasm  
libopencore-amrnb-dev libopencore-amrwb-dev libv4l-  
dev libxine2-dev
```

### Parallelism and linear algebra libraries:

```
sudo apt-get install libtbb-dev libeigen3-dev
```

### Python:

```
sudo apt-get install python-dev python-tk python-  
numpy python3-dev python3-tk python3-numpy
```

### Java:

```
sudo apt-get install ant default-jdk
```

### Documentation:

```
sudo apt-get install doxygen
```

## 4. Κατέβασμα και αποσυμπίεση της OpenCV

Μπείτε στον επίσημο δικτυακό τόπο της [OpenCV](#) και κατεβάστε την τελευταία έκδοση για Linux. Στη συνέχεια αποσυμπίεστε το αρχείο που κατεβάσατε.

## 5. Μεταγλώττιση και εγκατάσταση της OpenCV

Στο τερματικό σας, βεβαιωθείτε ότι είστε μέσα στον κατάλογο της OpenCV και εκτελέστε τις ακόλουθες εντολές:

- `cd opencv`
- `mkdir build && cd build`
- `cmake -D CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local -D WITH_CUDA=OFF -D WITH_CUFFT=OFF -D WITH_CUBLAS=OFF -D WITH_NVCUVID=OFF -D WITH_OPENCL=OFF -D WITH_OPENCLAMDFFT=OFF -D WITH_OPENCLAMDBLAS=OFF -D BUILD_opencv_apps=OFF -D BUILD_DOCS=OFF -D BUILD_PERF_TESTS=OFF -D BUILD_TESTS=OFF -D ENABLE_NEON=on ..`
- `make`
- `sudo make install`
- `sudo ldconfig`

**\*\*Προσοχή:** Αν δε γίνεται το compile, τότε μπορεί να έχει αλλάξει κάποια παράμετρος, οπότε συμβουλευτείτε το διδάσκοντα.

**\*\*Προσοχή:** Σε περίπτωση προβλήματος με τις παραπάνω εντολές απλά διαγράφουμε τον κατάλογο build με την παρακάτω εντολή και ξανακάνουμε τη διαδικασία του βήματος 5: `rm -r build`

Εάν έχετε ολοκληρώσει τα παραπάνω βήματα με επιτυχία, τότε έχετε εγκαταστήσει την τελευταία έκδοση της OpenCV στο BeagleBoard σας.

Ωρα για ένα μικρό τεστ! Έχοντας ήδη γίνει διαχειριστής, δημιουργήστε μέσα στον κατάλογο `opencv-3.0.0/build` έναν φάκελο με το όνομα `examples`. Αυτός είναι ο κατάλογος στον οποίο θα τοποθετείτε όλα τα παραδείγματα και τα αρχεία με τους κώδικες. Μέσα στον φάκελο `examples`, λοιπόν, δημιουργήστε το αρχείο `test1.cpp`, με τις παρακάτω γραμμές κώδικα:

```
// Test to check the OpenCV version

#include <opencv2/opencv.hpp>
#include <iostream>

int main()
{
    std::cout << "Hello, OpenCV version " << CV_VERSION << std::endl;
    return 0;
}
```

Υποθέτοντας ότι έχετε αποθηκεύσει τον παραπάνω κώδικα στον σωστό φάκελο με το όνομα test1.cpp, επιβεβαιώστε ότι το σύστημά σας γνωρίζει για τις βιβλιοθήκες του opencv με: `pkg-config opencv --cflags --libs`

Θα πρέπει να σας εκτυπώσει κάποιους καταλόγους του opencv.

Στη συνέχεια μπορείτε να δημιουργήσετε και να εκτελέσετε το εκτελέσιμο αρχείο ως εξής:

```
g++ test_1.cpp -o test_1 `pkg-config opencv --cflags --libs`  
./test_1
```

Θα πρέπει να δείτε κάτι σαν αυτό:

```
~Hello, OpenCV version 3.0.0-dev
```

**(C3)** Παραδώστε screenshot από την παραπάνω εκτέλεση.

## 7. Παραδείγματα

### 7.1 Επεξεργασία εικόνας

Μέσα στον φάκελο examples δημιουργήστε το αρχείο test2.cpp, με τις παρακάτω γραμμές κώδικα:

```
// Test to convert a color image to gray  
// Build on Linux  
  
#include <opencv2/opencv.hpp>  
#include <iostream>  
  
int main() {  
    // Load the image file and check for success  
    cv::Mat input = cv::imread("lena.jpg", 1);  
    if(!input.data) {  
        std::cout << "Unable to open the image file" << std::endl;  
        return -1;  
    }  
  
    // Convert the input file to gray  
    cv::Mat gray_image;  
    cvtColor(input, gray_image, cv::COLOR_BGR2GRAY);  
  
    // Save the result  
    cv::imwrite("lena_gray.jpg", gray_image);  
}
```

```
    return 0;
}
```

Κατεβάστε την εικόνα **lena.jpg** από το διαδίκτυο (ή μια δική σας εικόνα από τη **webcam**) και τοποθετήστε την και αυτή μέσα στον κατάλογο `examples`.

**\*\*Προσοχή:** Αν χρησιμοποιείτε παλιότερη έκδοση του OpenCV, θα χρειαστεί να αλλάξετε την επισημασμένη γραμμή από πάνω με:

```
cvtColor(input, gray_image, CV_BGR2GRAY);
```

Για compile:

```
g++ test2.cpp -o test2 `pkg-config opencv --cflags --libs`
```

**(C4)** Εκτελέστε τον κώδικα και παραδώστε την εικόνα **lena\_gray.jpg** που θα δημιουργηθεί, καθώς και ένα screenshot των περιεχομένων του καταλόγου `examples` στο οποίο θα φαίνεται ότι έχει δημιουργηθεί η εικόνα `lena_gray.jpg`.

## 7.2 Αναγνώριση κύκλων και κόκκινων αντικειμένων

Αντιγράψτε το αρχείο **circle\_detect.cpp** και την εικόνα **image8.jpg** στον κατάλογο `examples`. Κάντε compile με την εντολή:

```
g++ -std=c++11 circle_detect.cpp -o circle_detect `pkg-config opencv --cflags --libs`
```

Εκτελέστε με την εντολή:

```
./circle_detect image8.jpg
```

Δοκιμάστε όποια άλλη εικόνα θέλετε.

**(C5+C6)** Παραδώστε τις δύο εικόνες που θα δημιουργηθούν μετά την εκτέλεση.

## 7.3 Λήψη frame με OpenCV

Αντιγράψτε τις παρακάτω γραμμές κώδικα σε ένα νέο αρχείο με το όνομα **capture.cpp**:

```
#include <opencv2/opencv.hpp>
#include <iostream>

using namespace cv;

int main(int, char**){
```

```
VideoCapture cap(0); // open the default camera
Mat frame;
cap >> frame; // get a new frame from camera
imwrite("cam_output.png", frame);

return 0;
}
```

Με ποια εντολή θα κάνετε compile? \_\_\_\_\_(A14)

Εκτελέστε και βγάλτε κάποιες φωτογραφίες με την κάμερα που είναι συνδεδεμένη στο Beagleboard.

## 7.4 Λήψη live video και αναγνώριση χρωμάτων

Αντιγράψτε το αρχείο `color_detect_cam.cpp` στον κατάλογο `examples`. Κάντε `compile` και εκτελέστε.

(C7) Παραδώστε screenshot της εκτέλεσης.

## 7.5 Λήψη live video και αναγνώριση κυκλικών αντικειμένων με παραμετροποιήσιμο εύρος χρωμάτων

Αντιγράψτε το αρχείο `color_detector.cpp` στον κατάλογο `examples`. Κάντε `compile` και εκτελέστε.

(C8) Παραδώστε screenshot της εκτέλεσης.