

**ΙΟΝΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ
ΤΜΗΜΑ ΤΕΧΝΩΝ ΗΧΟΥ ΚΑΙ ΕΙΚΟΝΑΣ**

Ατομική Εργασία

**ΜΕΤΑΤΡΟΠΗ ΤΟΥ RASPBERRY ΠΙ ΣΕ ΔΙΑΔΡΑΣΤΙΚΟ ΜΟΥΣΙΚΟ
ΣΥΣΤΗΜΑ**

**Λουκάς Μπασούκος
ΤΧ2007063**

**Κέρκυρα,
Φεβρουάριος 2015**

Εισαγωγή

Για την εργασία αυτή χρησιμοποίησα το SuperCollider, το Emacs, το Node.js και το πρωτόκολλο επικοινωνίας OSC για να μετατρέψω ένα Raspberry Pi σε διαδραστικό μουσικό σύστημα.

Η εργασία έχει χωριστεί σε τρία μέρη:

1. Εγκατάσταση των τελευταίων εκδόσεων του SuperCollider (Emacs mode) και του Emacs, καθώς και η ρύθμισή τους για να χρησιμοποιούνται σε μη γραφικό περιβάλλον.
2. Εγκατάσταση της τελευταίας έκδοσης του Node.js και χρήση της εφαρμογής Beloi OSC για τη δημιουργία ενός browser-based μουσικού controller.
3. Ρύθμιση του Raspberry Pi ως ασύρματο access point χρησιμοποιώντας μια USB συσκευή wi-fi, έτσι ώστε η δημιουργία δικτύου να μην απαιτεί σύνδεση στο διαδίκτυο.

Επίσης, γίνεται αναφορά στα εξής θέματα:

- Αρχικές ρυθμίσεις στο Raspberry Pi.
- Αντιμετώπιση κάποιων βασικών σφαλμάτων του SuperCollider που σχετίζονται με τη χρήση του σε μη γραφικό περιβάλλον.
- Απομακρυσμένη σύνδεση στο Raspberry Pi χρησιμοποιώντας το πρωτόκολλο μεταφοράς δεδομένων SSH.

Ο στόχος της εργασίας είναι να δημιουργηθεί, ξεκινώντας από το μηδέν, ένα φτηνό και άμεσα αξιοποιήσιμο μουσικό σύστημα με δυνατότητες διάδρασης όσον αφορά τον μουσικό αυτοσχεδιασμό. Ως ηχητική πηγή λειτουργεί το SuperCollider το οποίο τρέχει στο Raspberry Pi. Χρησιμοποίησα το node.js για την δημιουργία ενός browser-based controller, έτσι ώστε οποιαδήποτε συσκευή με δυνατότητα πρόσβασης σε δίκτυο να μπορεί να χρησιμοποιηθεί ως μουσικό controller, στέλνοντας OSC μηνύματα στον server του SuperCollider και παραλάσσοντας τον ήχο με διάφορους τρόπους. Δεδομένου λοιπόν ότι οι περισσότεροι άνθρωποι σήμερα διαθέτουν τέτοιες συσκευές, και μάλιστα φορητές, η συμμετοχή σε έναν τέτοιου είδους αυτοσχεδιασμό γίνεται πολύ εύκολη και αυθόρμητη, οπουδήποτε και οποτεδήποτε αυτός λαμβάνει χώρα.

Αναλυτικότερες οδηγίες για τα επόμενα βήματα υπάρχουν στο repository της εργασίας στο GitHub (https://github.com/dakodeon/RPi_Music_System.git).

Σημειώνεται επίσης ότι όλα τα παρακάτω βήματα εκτός από την εγκατάσταση λειτουργικού συστήματος στο Raspberry Pi γίνονται σε επίπεδο τερματικού, χωρίς καθόλου χρήση γραφικού περιβάλλοντος.

Βασικές ρυθμίσεις του Raspberry Pi, εγκατάσταση του SuperCollider και του Emacs

Ξεκινάμε εγκαθιστώντας μια έκδοση του λειτουργικού συστήματος Raspbian (μια παραλλαγή του Debian Linux ειδικά για το Raspberry Pi) σε μια κάρτα αποθήκευσης micro SD. Ο τρόπος εγκατάστασης διαφέρει ανάλογα με το λειτουργικό σύστημα που χρησιμοποιείται ως πλατφόρμα για την εγκατάσταση. Ειδικές οδηγίες για τα τρία μείζονα λειτουργικά συστήματα (Mac OS, Windows και Linux) υπάρχουν στον ιστότοπο του Raspberry Pi Foundation. Αφού γίνει η εγκατάσταση, η micro SD τοποθετείται στην ειδική υποδοχή του Raspberry Pi και το λειτουργικό λειτουργεί αμέσως. Έπειτα χρησιμοποιούμε ένα καλώδιο Ethernet ή μια USB συσκευή wi-fi για να συνδεθούμε στο διαδίκτυο.

Το λειτουργικό σύστημα Raspbian έχει προεγκατεστημένη μια έκδοση του server του SuperCollider, καθώς και μια έκδοση του Jack Audio Connection Kit (JACK). Πρώτη κίνησή μας είναι να απεγκαταστήσουμε τις προεγκατεστημένες εκδόσεις με σκοπό να προχωρήσουμε σε μια “καθαρή” εγκατάσταση των τελευταίων εκδόσεων. Πρώτα εγκαθιστούμε τα dependencies, καθώς και τις τελευταίες εκδόσεις των εργαλείων για το compile. Στη συνέχεια εγκαθιστούμε τις τελευταίες εκδόσεις των JACK, Emacs και SuperCollider. Σ' αυτό το σημείο κάνουμε έναν έλεγχο για την ορθή λειτουργία των όσων εγκαταστήσαμε μέχρι τώρα. Τέλος, ενεργοποιούμε το SuperCollider mode για το Emacs μέσω της κατάλληλης εντολής στο init file του Emacs.

Όταν όμως θα προσπαθήσουμε να τρέξουμε το SuperCollider μέσω του Emacs, θα παρατηρήσουμε ότι προκύπτουν δύο σφάλματα: το ένα αφορά την απουσία γραφικού περιβάλλοντος, ενώ το άλλο δεν μας επιτρέπει την εκκίνηση του server του SuperCollider.

Το πρώτο σφάλμα, που είναι αβλαβές αλλά ενοχλητικό, λύνεται σχετικά απλά, εντοπίζοντας και μετονομάζοντας δύο αρχεία από την βιβλιοθήκη του SuperCollider, τα οποία αφορούν τον έλεγχο του χρησιμοποιούμενου GUI.

Το δεύτερο σφάλμα είναι πιο σοβαρό, καθώς άμα δεν το αντιμετωπίσουμε, δεν μπορούμε να χρησιμοποιήσουμε το SuperCollider μέσω του Emacs. Σχετίζεται πάλι με την μη ύπαρξη γραφικού περιβάλλοντος και πιο συγκεκριμένα με τα βήματα που ακολουθούμε για να τρέξουμε το SuperCollider. Σε περιβάλλον τερματικού, ενεργοποιούμε πρώτα το JACK, μετά το scsynth (τον server του SuperCollider) και τέλος την γλώσσα (sclang) και το SuperCollider λειτουργεί κανονικά. Στο περιβάλλον του Emacs όμως, τυπικά θα χρησιμοποιούσαμε την εντολή `Server.default.boot` η οποία επιχειρεί να ενεργοποιήσει το JACK και τον server μετά την ενεργοποίηση της γλώσσας, κάτι το οποίο δημιουργεί το συγκεκριμένο σφάλμα. Για να το αντιμετωπίσουμε, ενεργοποιούμε το JACK και τον server και τέλος το Emacs σε sclang mode. Επειδή όμως με αυτόν τον τρόπο δεν χρησιμοποιούμε την εντολή `boot`, δεν έχει αρχικοποιηθεί το βασικό group του SuperCollider, παρ' όλο που ήδη τρέχει ο server. Αυτό αντιμετωπίζεται χειροκίνητα, με την εντολή `Server.default.initTree`, την οποία τοποθετούμε στο αρχείο `startup.scd`, ένα αρχείο το οποίο εκτελείται κατά την ενεργοποίηση της sclang.

Σε αυτό το σημείο, και εάν έχουν ακολουθηθεί σωστά τα βήματα, θα μπορούμε να ενεργοποιήσουμε το SuperCollider μέσω του Emacs και να το χρησιμοποιήσουμε χωρίς προβλήματα.

Εγκατάσταση του Node.js και βασική χρήση της εφαρμογής Beloi OSC

Το Node.js είναι ένα περιβάλλον για την δημιουργία δικτυακών εφαρμογών σε JavaScript, οι οποίες να εκτελούνται στον server και όχι στον browser. Χρησιμοποιώντας αυτό το εργαλείο, μπορούμε να φτιάξουμε μια εφαρμογή σε μορφή ιστοσελίδας στον server μας (σε αυτήν την περίπτωση το Raspberry Pi), στην οποία θα περιέχονται controllers και αξιοποιώντας διάφορες βιβλιοθήκες της JavaScript, όπως η osc-min και η socket.io, θα αποστέλλονται μηνύματα OSC στον server του SuperCollider. Μια τέτοια εφαρμογή, την οποία και θα χρησιμοποιήσουμε, είναι η Beloi OSC. Με αυτόν τον τρόπο θα περιορίσουμε τις απαιτήσεις του συστήματος όσον αφορά το λογισμικό, καθώς είναι αυτονόητο ότι μια συσκευή με πρόσβαση σε δίκτυο διαθέτει κάποιον browser με δυνατότητα αποκωδικοποίησης JavaScript.

Πρώτα εγκαθιστούμε την τελευταία έκδοση του Node.js, κάνοντας build από τον πηγαίο κώδικα. Κατά την διαδικασία αυτή ενδέχεται προκύψει ένα σφάλμα σχετικό με τον C compiler, το οποίο αντιμετωπίζεται δημιουργώντας το κατάλληλο symbolic link στον φάκελλο που περιέχεται το εκτελέσιμο αρχείο του compiler.

Στη συνέχεια, κατεβάζουμε την εφαρμογή Beloi OSC από το repository στο GitHub. Όταν τρέξουμε την εφαρμογή και συνδεθούμε με έναν browser στην κατάλληλη διεύθυνση

και θύρα, θα δούμε μια ιστοσελίδα με πέντε απλά faders, ενώ στο τερματικό του server μας θα γίνει η αρχικοποίηση των μυνημάτων OSC. Θα παρατηρήσουμε επίσης μια σειρά τιμών με διαφορετική κάθε φορά διεύθυνση κάθε φορά που πειράζουμε κάποιο από τα faders. Αυτά είναι τα μυνήματα τα οποία πρέπει να προωθήσουμε στο SuperCollider. Για να γίνει αυτό, εντοπίζουμε μέσα στον φάκελο στον οποίο κατεβάσαμε την εφαρμογή το αρχείο που αφορά την ρύθμιση των θυρών και των διευθύνσεων IP μεταξύ των οποίων γίνεται η δρομολόγηση των μυνημάτων και αλλάζουμε την θύρα εξόδου να είναι ίδια με την θύρα εισόδου της slang (τυπικά αυτή είναι η 57120, αν και μπορούμε να την αλλάξουμε στο SuperCollider). Εάν η εφαρμογή Beloi OSC τρέχει σε διαφορετικό μηχάνημα από τον server του SuperCollider, αλλάζουμε και την διεύθυνση IP. Μπορούμε επίσης προαιρετικά να αλλάξουμε τις ονομασίες των faders, οι οποίες είναι και οι ταυτότητες των μυνημάτων OSC, για λόγους οργάνωσης και σαφήνειας.

Τέλος, πρέπει να ρυθμίσουμε το SuperCollider να δέχεται τα μυνήματα OSC. Αυτό μπορεί να γίνει με χρήση της κλάσης OSCdef ή OSCFunc.

Ρύθμιση του Raspberry Pi ως ασύρματο access point

Σε αυτό το σημείο θα προτιμήσουμε να συνδεόμαστε και να ελεγχουμε το σύστημά μας από απομακρυσμένη τοποθεσία μέσω του πρωτοκόλλου SSH. Ο λόγος είναι ότι με αυτόν τον τρόπο μπορούμε να αποφύγουμε τις πολλές καλωδιώσεις στο Raspberry Pi, κάτι πολύ χρήσιμο εάν π.χ. το Raspberry Pi αποτελεί μέρος μίας εγκατάστασης και πρέπει να μην είναι εμφανές ή απαιτείται από εμάς ο απομακρυσμένος χειρισμός του. Η σύνδεση είναι απλή, δεδομένου ότι χρησιμοποιούμε σύστημα Unix και ότι γνωρίζουμε τα username και password, καθώς και την διεύθυνση IP του Raspberry Pi.

Ανακύπτουν όμως δύο προβλήματα: Πρώτον η δικτύωση, η οποία άμα βασίζεται στην ύπαρξη router θα είναι ένας επιπλέον περιορισμός και δεύτερον το πως θα γνωρίζουμε την IP του Raspberry Pi χωρίς να έχουμε πρόσβαση στο τερματικό του. Και τα δύο αυτά προβλήματα λύνονται άμα μετατρέψουμε το Raspberry Pi, χρησιμοποιώντας μια USB συσκευή wi-fi, σε wireless access point με static IP. Πρέπει όμως να έχουμε υπ' όψιν ότι δεν θα μπορούμε να συνδεθούμε ταυτόχρονα στο διαδίκτυο με την ίδια συσκευή wi-fi. Για να γίνεται και αυτό θα πρέπει να χρησιμοποιήσουμε το καλώδιο Ethernet ή μια δεύτερη συσκευή wi-fi.

Πρώτα θα ανοίξουμε το αρχείο μας με τις ρυθμίσεις δικτύου και θα ορίσουμε μια static IP για το ασύρματο δίκτυο. Στη συνέχεια θα εγκαταστήσουμε έναν DHCP server ο οποίος θα διανέμει τις διευθύνσεις IP στα συνδεόμενα μηχανήματα. Στο αρχείο ρυθμίσεων του server θα ορίσουμε το επιθυμητό εύρος διευθύνσεων IP που θα διαθέσουμε, ενώ θα ορίσουμε την wi-fi συσκευή μας ως προεπιλεγμένη για αυτόν. Στην συνέχεια εγκαθιστούμε την εφαρμογή Hostapd, η οποία θα είναι η πλατφόρμα δημιουργίας του ασύρματου δικτύου μας. Έπειτα θα ανοίξουμε το αρχείο ρυθμίσεων της εφαρμογής (αν δεν υπάρχει θα το δημιουργήσουμε εμείς) και θα προσθέσουμε της πληροφορίες που θα ενεργοποιήσουν το δίκτυό μας, ορίζοντας παράλληλα και το όνομα του δικτύου καθώς και το passphrase που θα απαιτείται για την πρόσβαση σε αυτό. Εναλλακτικά θα μπορούσαμε να δημιουργήσουμε ανοικτό δίκτυο χωρίς passphrase, αλλά αυτό μπορεί να αποβεί επικίνδυνο για την σταθερότητα του συστήματος. Τέλος θα ενεργοποιήσουμε την δυνατότητα προώθησης IP για το Raspberry Pi.

Κάτι που πρέπει να προσέξουμε σε αυτό το σημείο είναι οι drivers που χρησιμοποιεί η συσκευή wi-fi που χρησιμοποιούμε. Ενδέχεται να χρειάζονται κάποιες επιμέρους ρυθμίσεις καθώς η εφαρμογή hostapd υποστηρίζει έναν συγκεκριμένο τύπο driver και ίσως χρειαστεί μια τροποποίηση των ρυθμίσεων καθώς και της ίδιας της εφαρμογής. Αφού ελέγξουμε και αυτό το ενδεχόμενο, μπορούμε να χρησιμοποιήσουμε το Raspberry Pi σαν wireless access point, ακόμα και για να συνδεθούμε στο διαδίκτυο.

Επίλογος

Συνδυάζοντας όλα τα παραπάνω θα έχουμε στα χέρια μας ένα μουσικό σύστημα με διαδραστικές δυνατότητες μουσικού αυτοσχεδιασμού, το οποίο θα λειτουργεί ασύρματα αλλά αυτόνομα από το διαδίκτυο, και του οποίου ο έλεγχος θα μπορεί να γίνεται από οποιονδήποτε θέλει και διαθέτει συσκευή με πρόσβαση σε δίκτυο (για παράδειγμα ένα οποιοδήποτε smartphone). Εξειδικευμένα λογισμικά (πχ SuperCollider, Emacs, Node.js) θα παρίστανται μόνο στο κεντρικό σύστημα, κάνοντας την συμμετοχή όσο το δυνατό απλούστερη. Το ηχητικό αποτέλεσμα, το πρωτογενές υλικό που θα χρησιμοποιηθεί καθώς και ο βαθμός επιτρεπόμενου ελέγχου από τους συμμετέχοντες εξαρτάται από τις προσωπικές μας προτιμήσεις (αλλά και από τους περιορισμούς του εκάστοτε κεντρικού συστήματος) καθώς έχουμε πλήρη ελευθερία στον προγραμματισμό του SuperCollider, στον καθορισμό των παραμέτρων που θα ελέγχονται από τα μηνύματα OSC και στην μορφοποίηση της ιστοσελίδας που θα λειτουργεί σαν controller. Το μουσικό αποτέλεσμα εδώ είναι δευτερεύον. Το βάρος δίνεται στην χρήση της σύγχρονης τεχνολογίας με δημιουργικό τρόπο, με έμφαση στη συνεργασία και στην ανθρώπινη επαφή.

Ιστογραφία

<http://www.raspberrypi.org/documentation/installation/installing-images/>
<http://supercollider.github.io/development/building-raspberrypi.html>
<http://sourceforge.net/p/supercollider/supercollider/ci/master/tree/editors/scel/>
<http://doc.sccode.org/Classes/Server.html>
<http://chrisperkins.blogspot.gr/2011/07/building-emacs-24.html>
http://ergoemacs.org/emacs/building_emacs_on_linux.html
https://www.gnu.org/software/emacs/manual/html_node/emacs/
<http://nodejs.org/dist/latest/>
http://elinux.org/Node.js_on_RPi
<https://www.youtube.com/watch?v=J6g53Hm0rq4>
<http://stackoverflow.com/questions/11696162/node-js-install-synology-ds212-error>
<http://www.maketecheasier.com/set-up-raspberry-pi-as-wireless-access-point/>