



# Βιοϊατρική τεχνολογία

## Ενότητα 4: Καρδιογράφος – Σπιρόμετρο

Αν. καθηγήτης Αγγελίδης Παντελής

e-mail: [paggelidis@uowm.gr](mailto:paggelidis@uowm.gr)

ΕΕΔΙΠ Μπέλλου Σοφία

e-mail: [sbellou@uowm.gr](mailto:sbellou@uowm.gr)

Τμήμα Μηχανικών Πληροφορικής και Τηλεπικοινωνιών

---



# 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 στο Πανεπιστήμιο 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 ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ  
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



# Καρδιογράφος



- Ο ηλεκτροκαρδιογράφος **SelfCheck ECG** είναι μια απλή στη χρήση φορητή, ιατρική συσκευή, η οποία επιτρέπει την καταγραφή ηλεκτροκαρδιογραφήματος 1 ή 12 απαγωγών.
- Τα αποτελέσματα μεταδίδονται συνεχώς κατά τη διάρκεια της εξέτασης μέσω Bluetooth στο PDA. Στη συνέχεια τα αποτελέσματα μπορούν να αποσταλούν σε τοπικό ή απομακρυσμένο σταθμό για ανάλυση από τον επαγγελματία της υγείας.
- Παρέχεται έτσι η δυνατότητα στους ασθενείς και στον ιατρό να παρακολουθούν τα βιοσήματά τους από οπουδήποτε και οποτεδήποτε.

# ECGApp.java (1/5)

---

- Θέτουμε:
  - sampling frequency:  $fs=250$
  - Period  $T_s$  of sampling:  $T_s=1f/fs$
  - Calibration time for each phase:  $calibrationTime=3500$
- Δημιουργούμε κι ένα αρχείο `ecgFILE` στο οποίο θα αποθηκεύονται τα δεδομένα.
- Ορίζουμε κανάλι επικοινωνίας, κανάλι εισερχόμενων και εξερχόμενων δεδομένων.
  - `static StreamConnection sc;`
  - `static DataOutputStream dos;`
  - `static DataInputStream dis;`



# ECGApp.java (2/5)

---

- Ορίζουμε αρχικούς πίνακες όπου θα αποθηκεύονται τα δεδομένα που θα αποστέλλονται και θα λαμβάνονται. Τους πίνακες δηλαδή που θα χρειαστούμε για να επικοινωνεί το σύστημά μας με τη συσκευή.
  - `byte[] initCommand1 = {0x20,0x20};`
- Ορίζουμε επίσης τον πίνακα όπου θα αποθηκεύεται η απόκριση της συσκευής, η οποία θα ορίζει με τη σειρά της το είδος της εξέτασης (ηλεκτροκαρδιογράφημα 1 ή 12 απαγωγών).
  - `byte[] initCommand2 = {0x51,0x51};`
- Ορίζουμε και τον πίνακα όπου θα αποθηκεύονται τα δεδομένα που προκύπτουν (response) όταν η συσκευή θα πιάσει τον παλμό του ασθενή.
  - `byte[] pingCommand = {0x45,0x45,0x00};`



# ECGApp.java (3/5)

---

- Στη συνέχεια μετά τις αποκρίσεις από τη συσκευή γίνεται έλεγχος για το είδος της εξέτασης (1 ή 12 απαγωγών).
- Ορίζουμε μήνυμα λάθους σε περίπτωση που δεν έχει επιλεχθεί συσκευή.
  - `JOptionPane.showMessageDialog(new JFrame(),"Δεν έχετε επιλέξει συσκευές.\nΠαρακαλώ επιλέξτε Ρυθμίσεις->Συσκευές.", "ΣΦΑΛΜΑ",JOptionPane.WARNING_MESSAGE) stop();`



# ECGApp.java (4/5)

- **Καρδιογράφημα 1 απαγωγού:**

- Ορίζουμε τη διάρκεια: `long duration=34000`, σε ms.
- Ορίζουμε το μέγιστο αριθμό των δειγμάτων, οι οποίοι αποθηκεύονται σε πίνακα: `int maxSamples = (int)((duration/1000)*(fs+50))`.
- Τα δεδομένα μετά στέλνονται στο `ecgFile` .

- **Καρδιογράφημα 12 απαγωγών:**

- Εδώ ορίζουμε 3 διαφορετικές φάσεις:
  - Φάση 1η (V1-V2): `long phase1Duration=12000` και μέγιστος αριθμός δειγμάτων `int maxSamplesPhase1 = (int)((phase1Duration/1000)*(fs+50))`
  - Φάση 2η (V3-V4): `long phase2Duration=5000` και μέγιστος αριθμός δειγμάτων `int maxSamplesPhase2 = (int)((phase2Duration/1000)*(fs+50))`
  - Φάση 3η (V5-V6): `long phase3Duration=5000` και μέγιστος αριθμός δειγμάτων `int maxSamplesPhase3 = (int)((phase3Duration/1000)*(fs+50))`





# ECGApp.java (5/5)

---

- Έπειτα ολοκληρώνεται η εξέταση, με την παράλληλη πλήρη φόρτωση της μπάρας.
- Αποθηκεύονται όλες οι μετρήσεις στο αρχείο `ecgFile`.



---

# Σπιρόμετρο



# Τι είναι το Σπιρόμετρο



- Το **Σπιρόμετρο Spiro Pro** είναι μια απλή στη χρήση, φορητή, ιατρική συσκευή, η οποία μετράει με ακρίβεια παραμέτρους αναπνευστικής λειτουργίας και αποθηκεύει τα αποτελέσματα των μετρήσεων στη μνήμη. Το σύστημα καταγράφει τις καμπύλες Όγκου-Χρόνου και Ροής-Όγκου καθώς και τις ακόλουθες παραμέτρους: FVC, FEV1, FEV3, PEF, FEF25%, FEF50%, FEF75%, FEV25%-75%. Στη συνέχεια με τη χρήση της τεχνολογίας Bluetooth τα αποτελέσματα μπορούν να αποσταλούν στο PDA για περαιτέρω ανάλυση.

# SpiroProApp.java (1/5)

---

- Ορίζουμε τις εξής βιβλιοθήκες:
  - Swing (Java GUI Widget Toolkit)
  - java.io
  - java.nio
  - javax.microedition.io



# SpiroProApp.java (2/5)

---

- Ορίζουμε τις μεταβλητές μας (serverURL, user).
- Καλούμε τη μέθοδο run (), η οποία με τη σειρά της καλεί την launchProgress () και την createConnection ().
- Η τελευταία δημιουργεί τη σύνδεση μεταξύ του PDA και του Σπιρόμετρου.
- Στη συνέχεια ορίζουμε τη συχνότητα και την περίοδο δειγματοληψίας (fs, Ts) των δεδομένων της εξέτασης.
- Δημιουργούμε ένα αρχείο (spiroFile) μέσα στο οποίο θα αποθηκευτούν τα δεδομένα.



# SpiroProApp.java (3/5)

---

- Ορίζουμε το πρωτόκολλο απόκρισης της συσκευής (initCommand 1,2,3, initResponse 1,2,3, stopCommand, stopResponse κτλ.).
- Γίνεται η σύνδεση μεταξύ του PDA και του Σπιρόμετρου, ενώ παράλληλα «γεμίζει» η μπάρα εξέλιξης της διαδικασίας, όπου και στο τέλος εμφανίζει μήνυμα «Συνδέθηκε».
- Έναρξη της εξέτασης, ενώ ορίζουμε πως σε περίπτωση λάθους θα εμφανιστεί μήνυμα σφάλματος.
- Εμφάνιση μηνύματος «Εκπνεύστε μετά τον ήχο» και ορισμός των μεταβλητών numberOfSamples, maxSamples, volume, oldFlow, maxDuration.
- Ορίζουμε τον πίνακα στον οποίο θα φορτώνονται οι τιμές της μέτρησης, αλλά και το beep tone, το οποίο σηματοδοτεί την έναρξη της εξέτασης.



# SpiroProApp.java (4/5)

---

- Αποθήκευση των δεδομένων ροής στη μεταβλητή flow (flowBuffer.put(flow), και υπολογίζουμε τον όγκο των πνευμόνων.
  - $\text{volume} += ((\text{flow} + \text{oldFlow})/2) * T_s;$
  - volumeBuffer.put(volume)
- Δίνουμε closeCommand και stopCommand για τον τερματισμό της εξέτασης, ενώ η μπάρα εξέλιξης της εξέτασης έχει φορτώσει πλήρως.
- Με τα δεδομένα που έχουμε λάβει θα πρέπει στη συνέχεια να υπολογίσουμε τις τιμές FVC, FEV1, FEV3, PEF, FEF25, FEF50, FEF75, FEF25-75, οι οποίες ορίζονται από το πρότυπο Σπυρομέτρησης.



# SpiroProApp.java (5/5)

- FVC: συνολικός όγκος του αέρα που έχει εκπνεύσει ο ασθενής.
- FEV1: όγκος αέρα που έχει εκπνευστεί στο πρώτο δευτερόλεπτο της εξέτασης.
- FEV3: όγκος αέρα που έχει εκπνευστεί στα τρία πρώτα δευτερόλεπτα .
- PEF: μέγιστη τιμή της ροής του αέρα (L/s).
- FEF25: μέσος όρος ροής που έχει εκπνευστεί κατά το πρώτο 25% της εξέτασης.
- FEF50: μέσος όρος ροής που έχει εκπνευστεί κατά το πρώτο 50% της εξέτασης.
- FEF75: μέσος όρος ροής που έχει εκπνευστεί κατά το 75% της εξέτασης.
- FEF25-75: μέσος όρος ροής που έχει εκπνευστεί από το 25% -75% της εξέτασης.
- Αποθήκευση όλων των δεδομένων στο αρχείο που δημιουργήσαμε αρχικά και τέλος απεικόνιση αυτών στην οθόνη.





---

# Τέλος Ενότητας



Ευρωπαϊκή Ένωση  
Ευρωπαϊκό Κοινωνικό Ταμείο



Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

