

ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ
ΚΑΙ ΕΦΑΡΜΟΓΕΣ ΤΗΣ ΣΕΙΡΑΣ
ROBO TECHNOLOGY ΤΗΣ LEGO

Υπεύθυνοι Υλοποίησης:
Κυριακίδης Χρήστος
Ντελής Γεώργιος

Υπεύθυνος Καθηγητής :
Σαπουνίδης Θεοδόσης

Πίνακας Περιεχομένων

- Ιστορική Αναδρομή
- Ρομποτική
- Η Lego το 1998 παρουσιάζει το RCX
- Τα εξωτερικά χαρ/κα του NXT
- Μπλοκ Διάγραμμα NXT
- Το εσωτερικό του NXT
- Connectors για τα Input Ports
- Connectors για τα Output Ports
- High speed communication Port
- Πώς θα φτιάξουμε το δικό μας καλώδιο διασύνδεσης
- Επαναφορτιζόμενη Μπαταρία
- Οθόνη απεικόνισης
- Αισθητήρες
- Μοτέρ Κίνησης
- Java
- VB.net
- C#
- Συμπεράσματα

Η ζωή είναι ένα παιχνίδι

So, Lets play

ή όπως λένε οι φίλοι μας οι Δανοί,

Leg Godt

Από αυτή τη φράση εμπνεύστηκε το όνομά της η εταιρία

LeGo

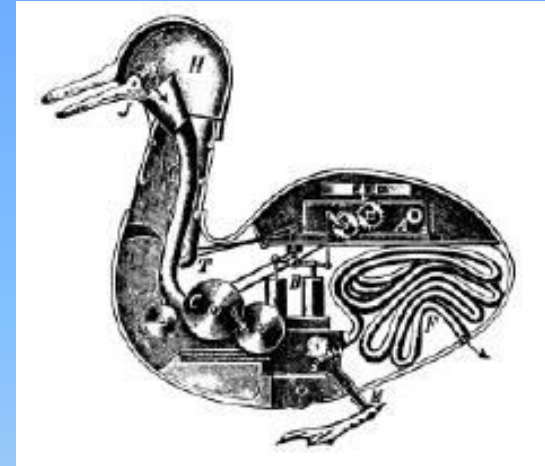
Leg Godt λοιπόν...

Ιστορική Αναδρομή



- Ο Μηχανισμός των Αντικυθήρων (150 π.Χ.)

Είναι ο αρχαιότερος αυτοματισμός που σώζεται ως σήμερα (Αρχ. Μουσείο Αθηνών). Μπορούσε να προβλέψει τις θέσεις των πλανητών.



- Η πάπια του Γάλλου Jacques de Vaucanson (1738)

Είναι μια ρομποτική πάπια που έχει τη δυνατότητα να τρώει σπόρους και να κουνάει τα φτερά της.

Modern Times (1936)



Οι μοντέρνοι
καιροί το 1936

Modern Times (c. 2000)



και οι
μοντέρνοι
καιροί σήμερα

Ρομποτική

- Η Ρομποτική είναι ένας νεοσύστατος τεχνολογικός κλάδος, παράγωγος της τεχνολογίας του αυτοματισμού και ασχολείται με τη μελέτη και την ανάπτυξη των ρομπότ.



- Ρομπότ είναι ένα προγραμματιζόμενο σύστημα αυτόματου ελέγχου, του οποίου η εμφάνιση και οι κινήσεις συχνά δείχνουν την εντύπωση πως ενεργεί κατά βούληση

Η Lego το 1998 παρουσιάζει το RCX



Χαρακτηριστικά:

- 16 bit επεξεργαστή
- MicroController (Renesas H8/300)
- 3 εισόδους (αισθητήρες αφής και φωτός)
- 3 εξόδου[A,B,C] (κινητήρες και λαμπάκια)
- Ηχείο
- Οθόνη LCD 4 ψηφίων
- Υπέρυθρη επικοινωνία IR

Υποστηριζόμενες Γλώσσες προγραμματισμού :

- RCX Code
- ROBOLAB (βασίζεται στο LabVIEW και αναπτύχθηκε στο Tufts University)
- C and C++ μέσω BrickOS
- Java μέσω leJOS
- NQC ("Not Quite C")
- Visual Basic
- RobotC (νέα γλώσσα συμβατή με την έκδοση NXT)

Η Lego το 1998 παρουσιάζει το RCX

- 1^η έκδοση (Με παροχή ρεύματος)

Για στατικά προγράμματα (ρομποτικοί βραχίονες)

- 2^η έκδοση (Με μπαταρίες)

Με περιορισμένο χρόνο λειτουργίας για μικροεφαρμογές

- Ο προγραμματισμός του RCX γίνεται υπέρυθρα και στις δύο εκδόσεις

Το RCX ήταν περιορισμένο στις δυνατότητες της εξέλιξης της ρομποτικής, οπότε η Lego προχώρησε στη δημιουργία του NXT

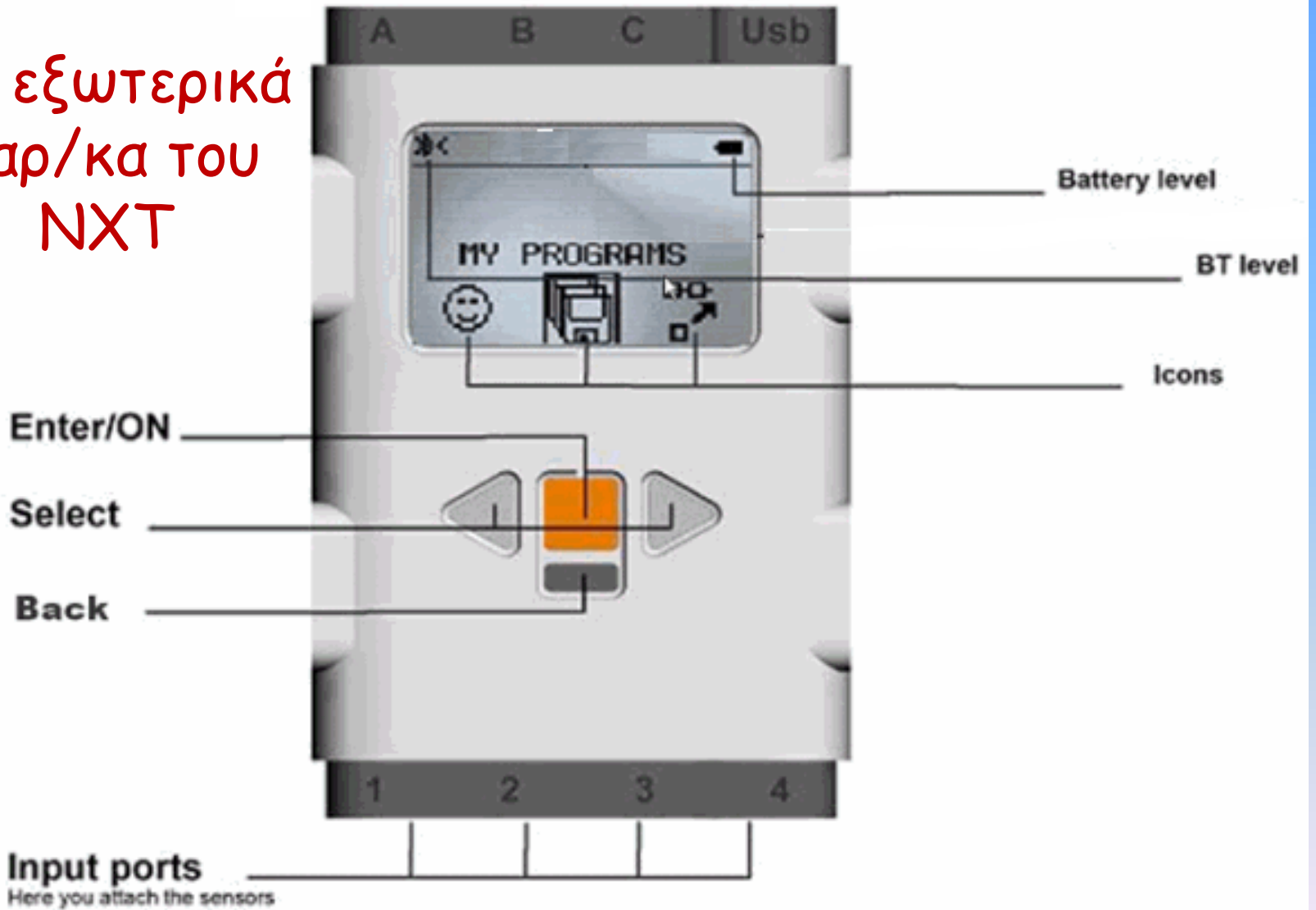
Output ports

Here you attach the motors

Usb port

You use this port to download and upload data.

Τα εξωτερικά
χαρ/κα του
NXT



Enter/ON

Select

Back

Battery level

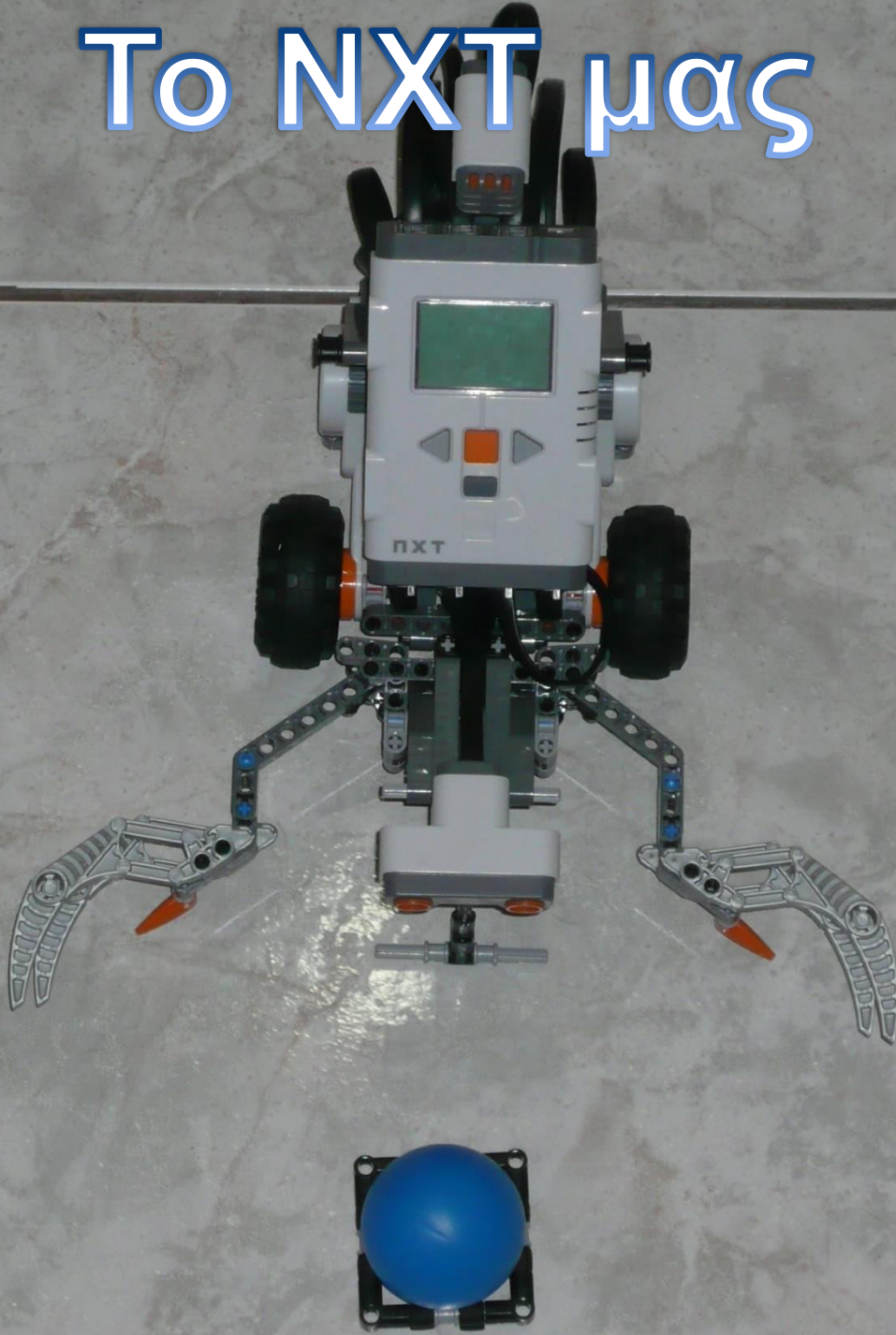
BT level

Icons

Input ports

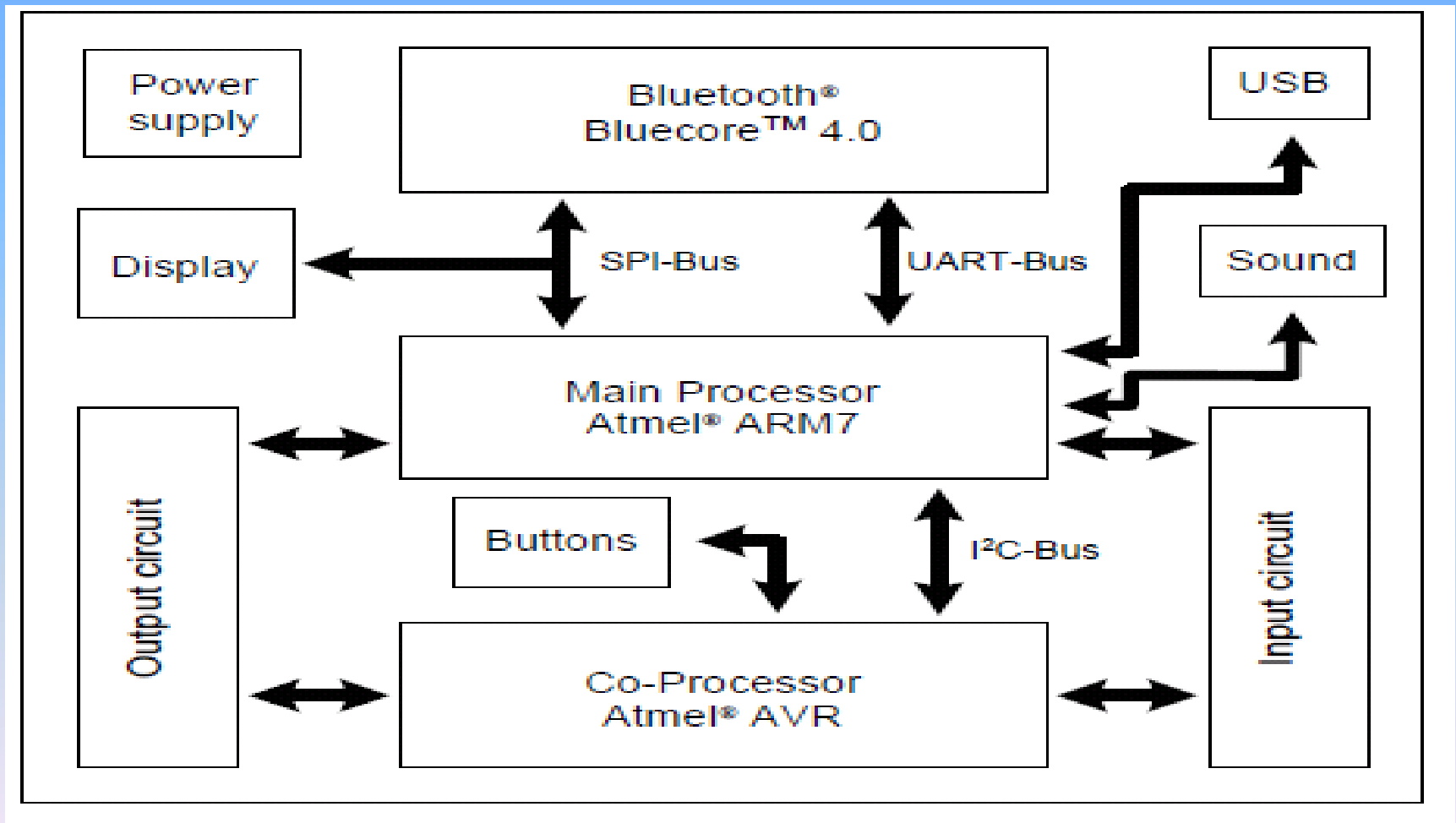
Here you attach the sensors

Το NXT μας

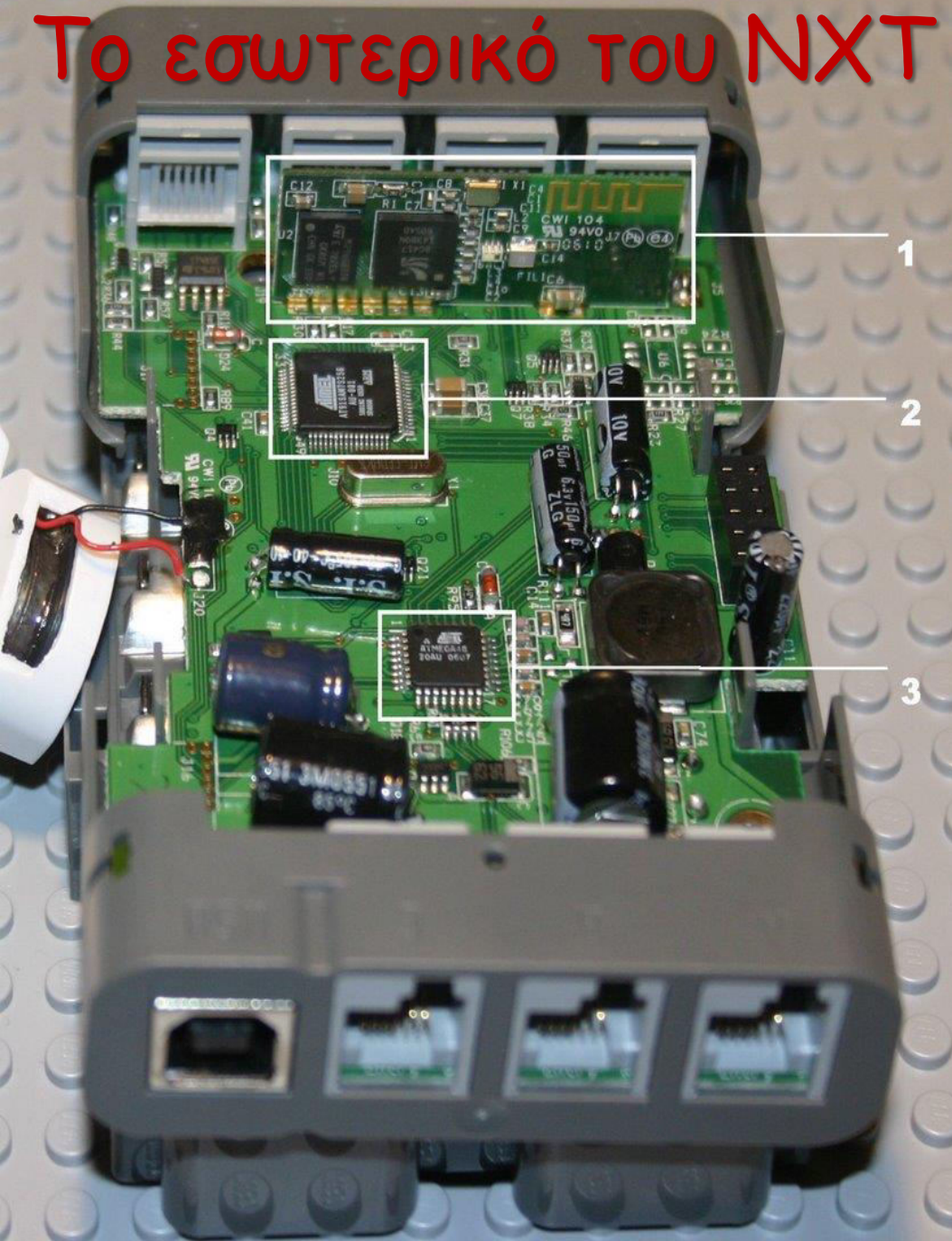


Κύρια μονάδα επεξεργασίας (MPU)	Atmel® 32-bit ARM® processor, AT91SAM7S256 - 256 KB FLASH - 64 KB RAM - 48 MHz
Μικροεπεξεργαστής	Atmel® 8-bit AVR processor - 4 KB FLASH- 512 Byte RAM - 8 MHz
Bluetooth wireless communication	CSR BlueCoreTM 4 v2.0
USB 2.0 communication	Full speed port (12 Mbit/s)
4 Εισόδους 3 Εξόδους	Τα καλώδια αποτελούνται από 6 συρματινές επαφές διασύνδεσης και υποστηρίζουν αναλογική και ψηφιακή διασύνδεση
Η διασύνδεση των εισόδων-εξόδων	Γίνετε με καλώδιο το οποίο προσαρμόζεται σε ένα συνδετήρα 6 επαφών , RJ12
Μια οθόνη (Display)	100 x 64 pixel LCD black & white graphical display - View area: 26 X 40.6 mm
Μεγάφωνο	Κανάλι ήχου εξόδου με 8-bit ανάλυση 2-16 KHz
4 Κουμπιά (χειρισμού Menu)	Τα κουμπιά είναι Φτιαγμένα από καουτσούκ
Τροφοδοσία	6 AA μπαταριές - Οι αλκαλικές μπαταρίες συνιστάτε. - επαναφορτιζόμενη λιθίου μπαταρία 1400 mAh μπορεί να χρησιμοποιηθεί

Μπλοκ Διάγραμμα NXT



Το εσωτερικό του NXT



➤ 1) Bluetooth

- το πρώτο Chip (M29W800DT) δέχεται τροφοδοσία 3.6 V, έχει 19 Memory Blocks, 64 bit Security Code και το λογισμικό ελέγχου του Cambridge Silicon Radio.

- Το τετράγωνο chip είναι ο μικροελεγκτής CSR BlueCore-4 (BC417143BON606AJ) όπου επικοινωνεί ασύρματα με τις συσκευές υποστηρίζοντας Bluetooth v.2.0

- Το ίχνος της χρυσής λωρίδας “ζικ-ζακ” είναι η κεραία του.

➤ 2) Κύρια Μονάδα Επεξεργασίας ARM-32bit,48 MHz (AT91SAM7s256)


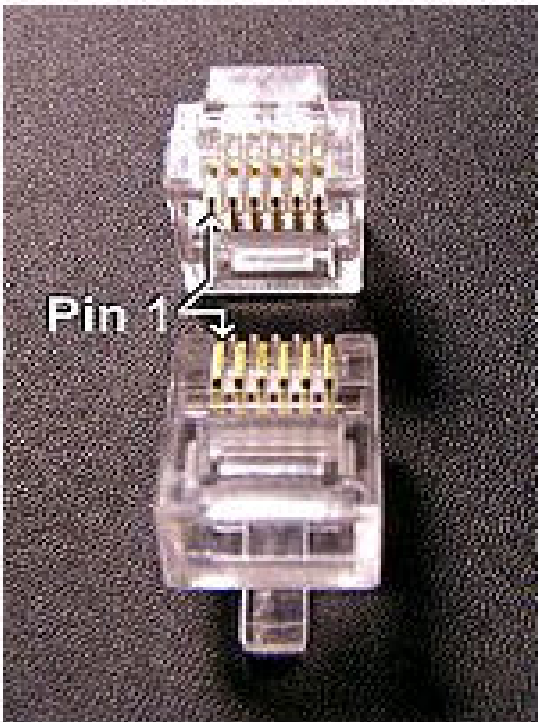





- Εδώ τρέχουν όλα τα προγράμματα
Περιλαμβάνει:

- Flash Memory (256 KB)
- Μνήμη RAM (64 KB)
- Τα αρχεία συστήματος
- Το πρωτόκολλο USB

➤ 3) Μικροεπεξεργαστής 8bit ATME1

- 4 KB Flash
- 512 RAM
- 8 MHz
- Οδηγεί την παλμογεννήτρια(PWM) (Έλεγχος μοτέρ)

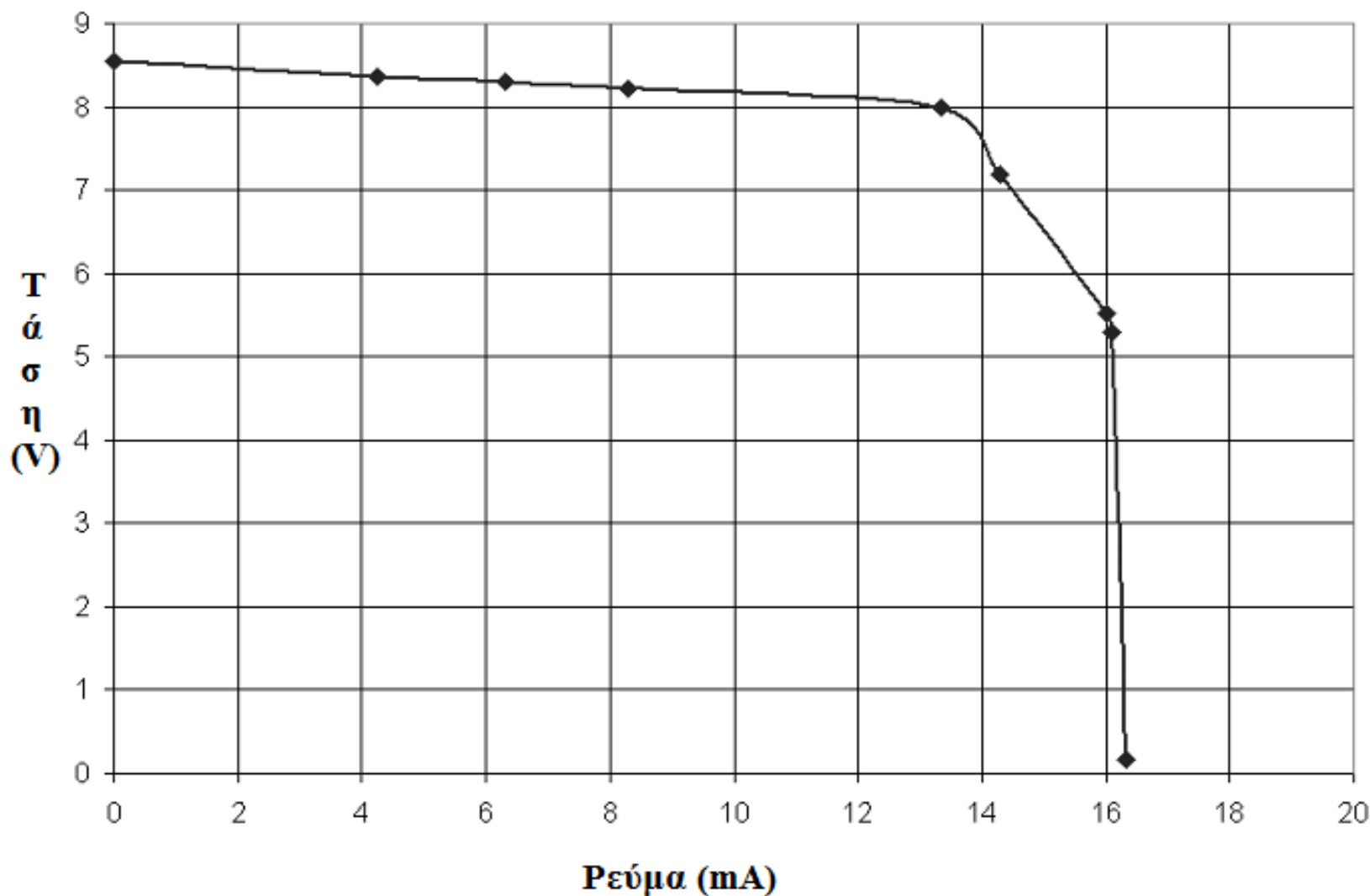
Connectors via TA Input Ports

Pin	Name	Function	Color	Pin Numbering
1	ANA	Analog interface, +9V Supply	 white	
2	GND	Ground	 black	
3	GND	Ground	 red	
4	IPOWERA	+4.3V Supply	 green	
5	DIGIAI0	I ² C Clock (SCL), RS-485 A	 yellow	
6	DIGIAI1	I ² C Data (SDA), RS-485 B	 blue	

Connectors για τα Input Ports

Pin 1

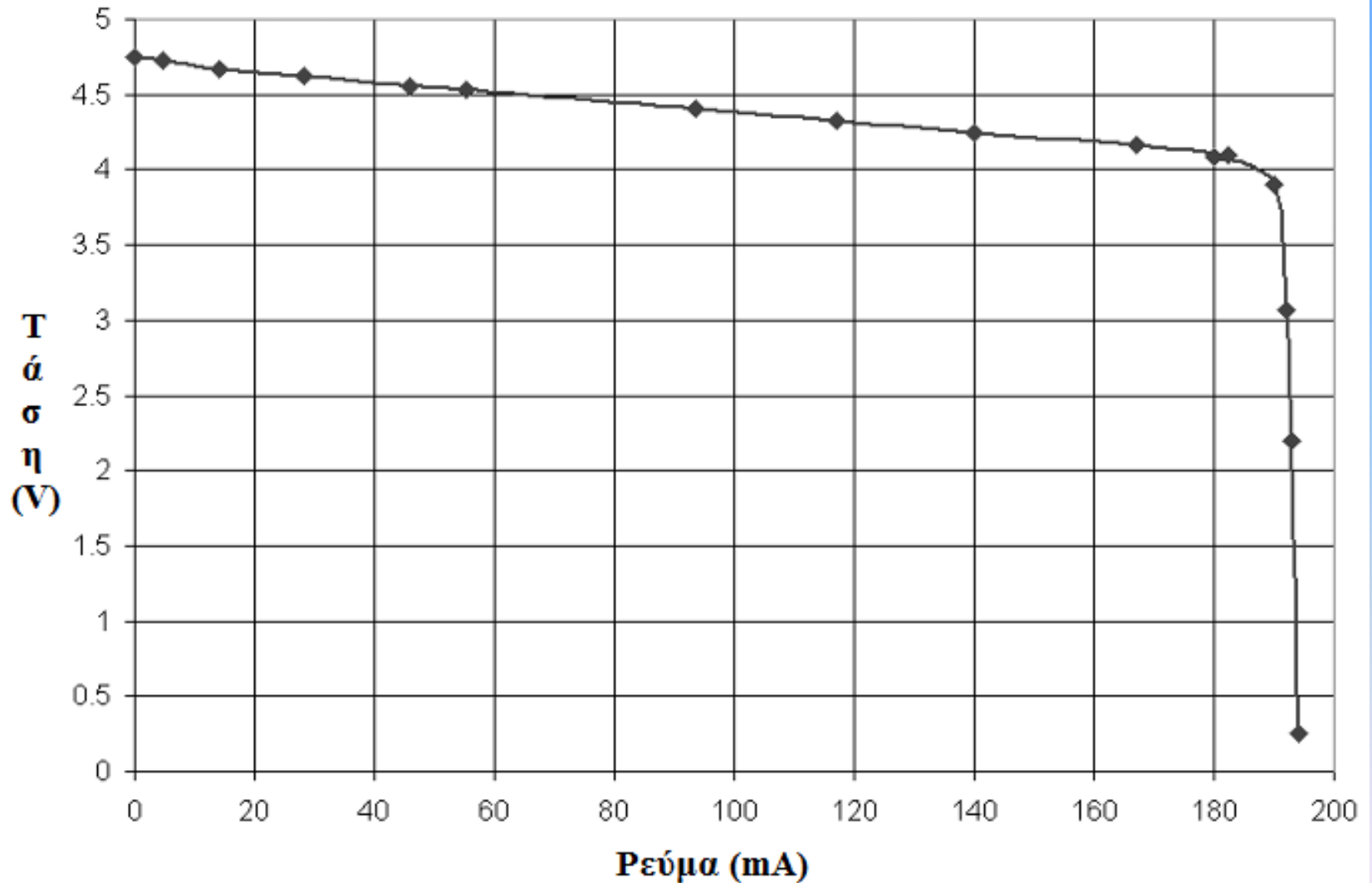
Τάση αισθητήρων 9V σε αντιδιαστολή με το ρεύμα



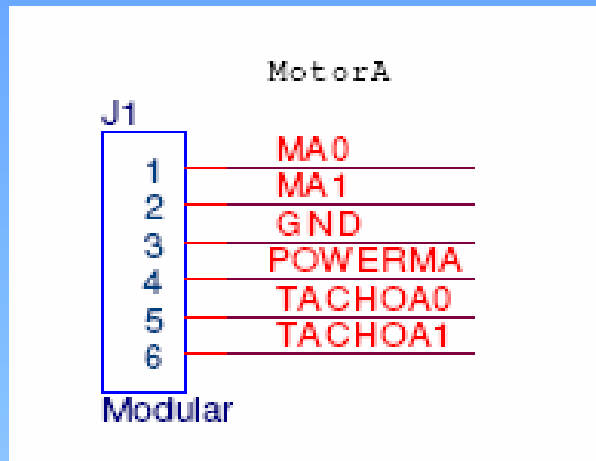
Connectors για τα Input Ports

Pin 4

Παρεχόμενη τάση 4,3 volt



Connectors για τα Output Ports



Αριθμός Pin ,Όνομα	Χρώμα	Λειτουργία
Pin 1,2 , MA0,1	Λευκό, Μαύρο	Μια παλμό-γεννήτρια PWM στέλνει σήμα για τον έλεγχο της ενεργοποίησης
Pin 3 , GND	Κόκκινο	Μέσω του σήματος περνάει η γείωση στη έξοδο που είναι χρήσιμη για την τροφοδοσία
PIN 4 , POWERMA	Πράσινο	Έχουμε 4,3volt τροφοδοσία στην έξοδο
PIN 5,6 , TACHOA0,1	Κίτρινο, Μπλέ	Δέχεται στη είσοδο μια τιμή που περιλαμβάνει αποδοχή και ενεργοποίηση της λειτουργίας της κίνησής του

High speed communication Port

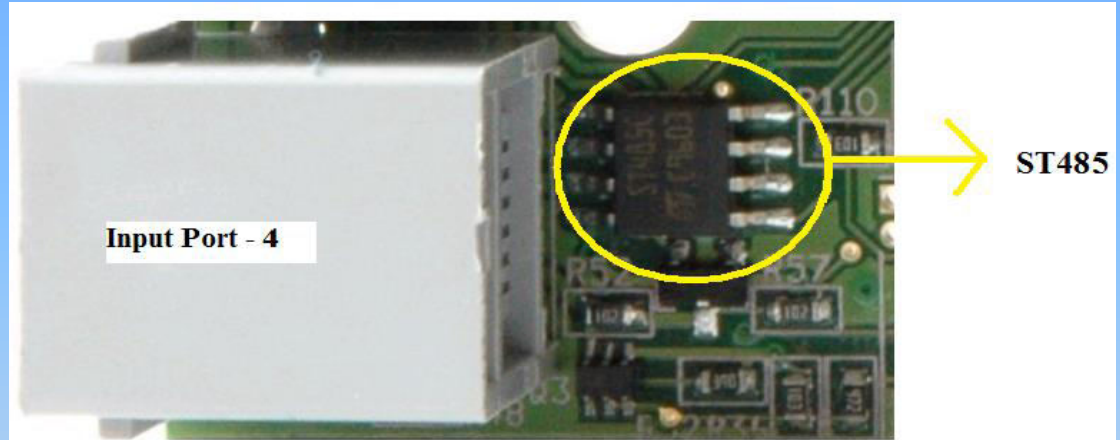
- Το NXT έτοιμο για μελλοντική μακρινή αμφίδρομη μετάδοση πληροφοριών

Αυτό επιτυγχάνεται χάρη στο πρωτόκολλο επικοινωνίας P-net και τη χρήση ενός ειδικού καλωδίου που επιτρέπει την επικοινωνία μέχρι 1200 μέτρα χωρίς επαναλήπτες.

- Είναι ένα πρωτόκολλο που μπορεί να χειριστεί μέχρι 300 bit/sec από 300 ανεξάρτητες διευθύνσεις.

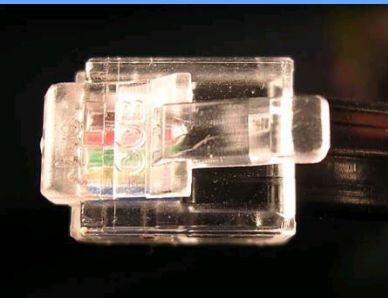
- Οι διευθύνσεις μπορούν να μεταφερθούν υπό τη μορφή επεξεργάσιμων πληροφοριών (θερμοκρασία, πίεση, ρεύμα, τάση κ.α.)

- Όλα τα παραπάνω πετυχαίνονται, διότι οι πληροφορίες επεξεργάζονται και στέλνονται παράλληλα



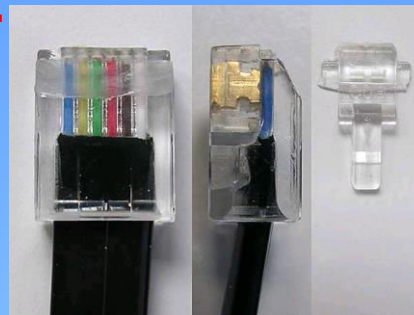
Πώς θα φτιάξουμε το δικό μας καλώδιο διασύνδεσης

1



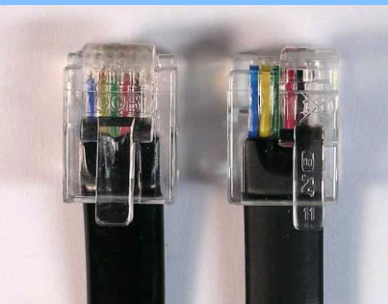
✓ Αρχίζουμε με ένα κανονικό βούλωμα RJ12, που εφαρμόστηκε στο καλώδιο. Το κλιψάκι πρέπει να έχει 6 υποδοχές διαθέσιμες το ίδιο και το καλώδιο αντίστοιχα για να μπορέσει να γίνει η τοποθέτηση των καλωδίων ανάλογα με τη σειρά χρώματος για να λειτουργήσει σωστά το καλώδιο.

4



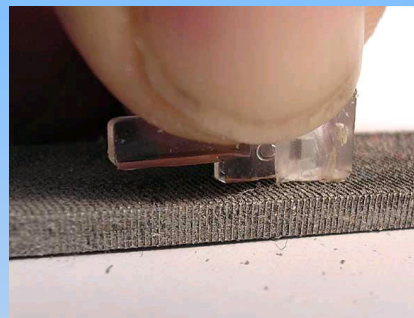
✓ Παρατηρούμε το αποτέλεσμα του κοψίματος όπου είναι φανερό το κλιψάκι χωρίς την ασφάλεια .

2



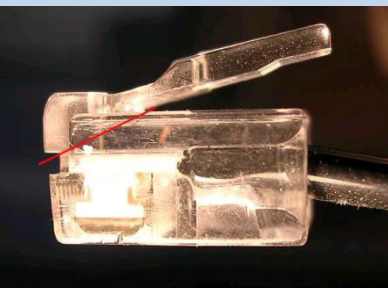
Συγκρίνοντας το κλιψάκι RJ-12 με ένα κλιψάκι Nxt παρατηρούμε δυο διαφορές :
✓ Η ασφάλεια πρέπει να μετακινηθεί δεξιά στο κλιψάκι του RJ-12.
✓ Στο καλώδιο RJ-12 το μπροστινό μέρος είναι πιο ευρύτερο και θα πρέπει να συρρικνωθεί για να ταιριάζει στο μέγεθος των πορτών (Ports) του Nxt .

5



✓ Ξύνουμε τις πλευρές τις ασφαλείας για να ταιριάζει με το πλάτος της ασφαλείας από το κλιψάκι του Nxt

3



✓ Το πρώτο βήμα είναι να κοπεί η ασφάλεια και πιο συγκεκριμένα να κοπεί μετά την κόκκινη γραμμή που παρουσιάζεται σε αυτήν την φωτογραφία.

6



✓ Ελέγχουμε την διαδικασία του κοψίματος βάζοντας την «ασφάλεια» στην υποδοχή ενός από τους αισθητήρες και πρέπει να ταιριάζει.

7 Πώς θα φτιάξουμε το δικό μας καλώδιο διασύνδεσης

7



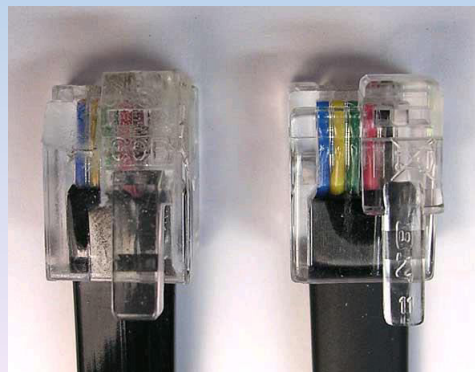
✓ Τοποθετούμε τον συνδετήρα σε μια μέγγενη, βάζουμε διπλά ένα μικρό κομμάτι χαρτί για να μην επεκταθεί η κόλλα στις επαφές του συνδετήρα και μετά βάζουμε την «ασφάλεια» πάνω στην κόλλα.

8



✓ Θερμαίνουμε την κόλλα με ένα λαμπτήρα για ταχύτερο και πιο καλό κόλλημα. Δεν πρέπει να τοποθετήσουμε πολύ κοντά τον συνδετήρα γιατί μπορεί να λιώσει..

9



✓ Τελικά , το αποτέλεσμα είναι εμφανές. Ο συνδετήρας που φτιάξαμε δεν έχει καμία διαφοροποίηση από τον συνδετήρα του Nxt.

Επαναφορτιζόμενη Μπαταρία

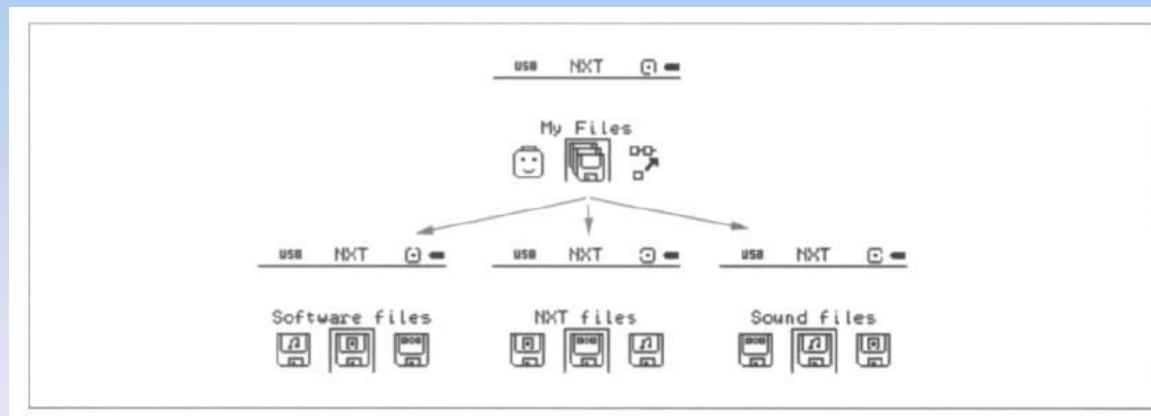


Supply voltage	Current		Effect (Battery = 9 Volts)	
	Max [mA]	Normal [mA]	Max [mW]	Normal [mW]
No load on motors				
9 Volt	339	114	5184	1422
5 Volt	271	112	1744	448
3.3 Volt	72	38	410	216
Load on motors				
9 Volt	2901	848	26109	7632
5 Volt	271	112	1142	307
3.3 Volt	72	38	410	137
Standby	46 uA assumed standby current due to brown out detection			

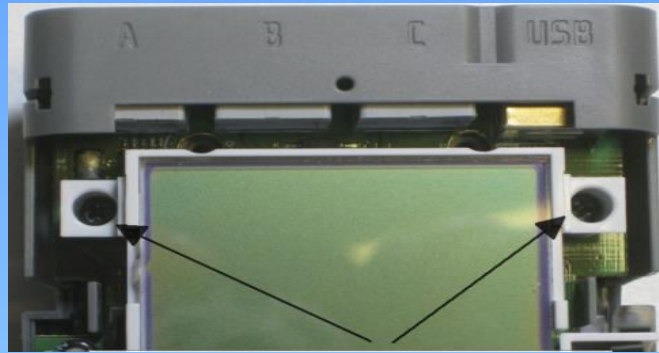
παρουσίαση των τρέχων τιμών του ρεύματος στο Nxt

Data File System

File Type	Extension(s)
Data files	.rdt
Executable files and Try Me programs	.rxex, .rtm
Icon files	.ric
Hidden menu files	.rms
Program files	.rpg
Sound files	.rso
Hidden system files	.sys
Temporary hidden files	.tmp



Οθόνη απεικόνισης



Format	100 x 64 dots
LCD mode	STN / Positive Reflective Mode / Gray
Viewing direction	6 o'clock
Driving scheme	1/65 duty cycle, 1/9 bias
Power supply voltage (VDD)	3.0V
LCD driving voltage (VLCD)	9.0V (for best contrast)

Αισθητήρες

1. Αισθητήρας αφής (Touch Sensor)



4. Αισθητήρας υπερήχων (Ultrasonic Sensor)



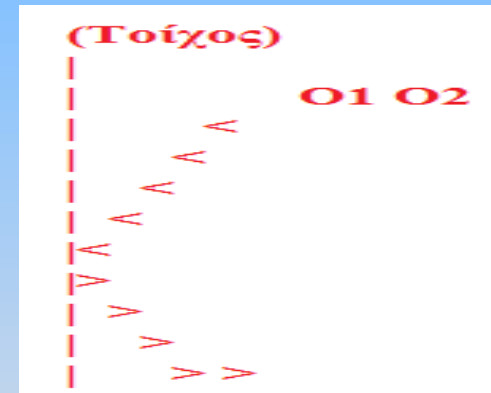
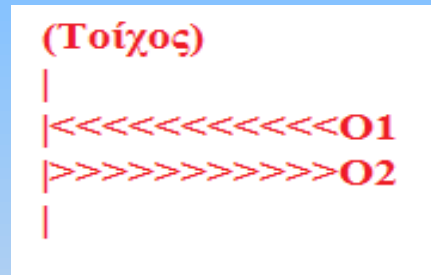
2. Αισθητήρας φωτός (Light Sensor)



Αυτά που βλέπει το ανθρώπινο μάτι



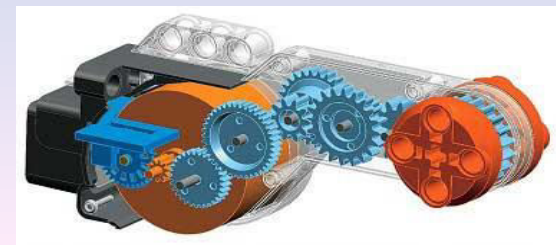
Αυτά βλέπει το robot χρησιμοποιώντας τον αισθητήρα φωτός



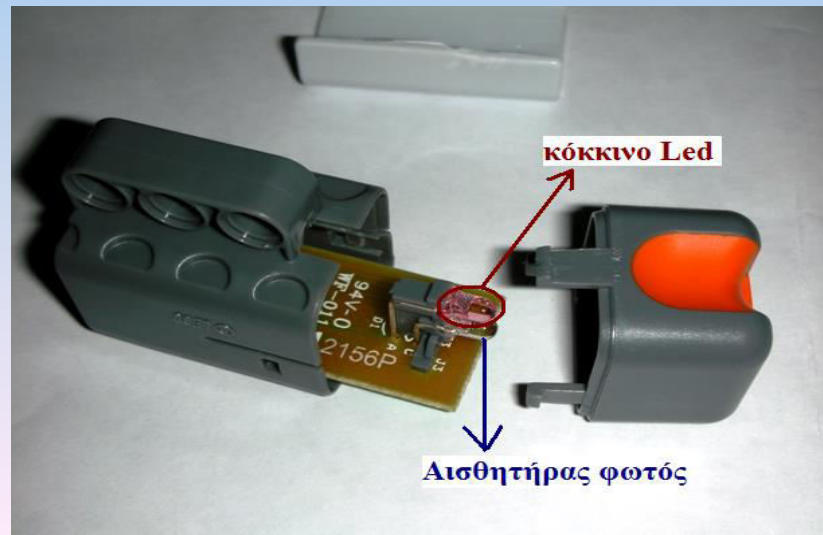
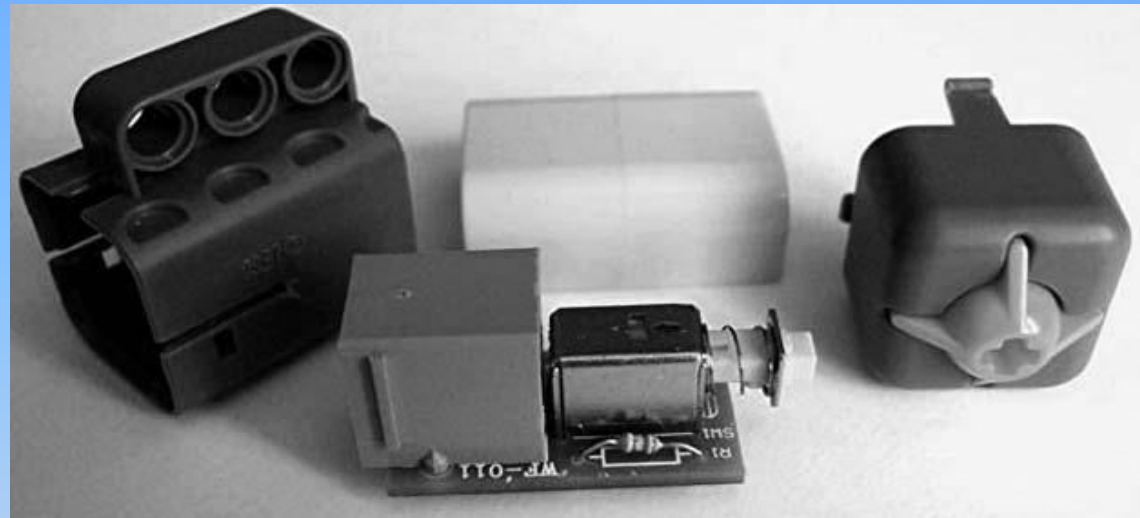
3. Αισθητήρας ήχου (Sound Sensor)



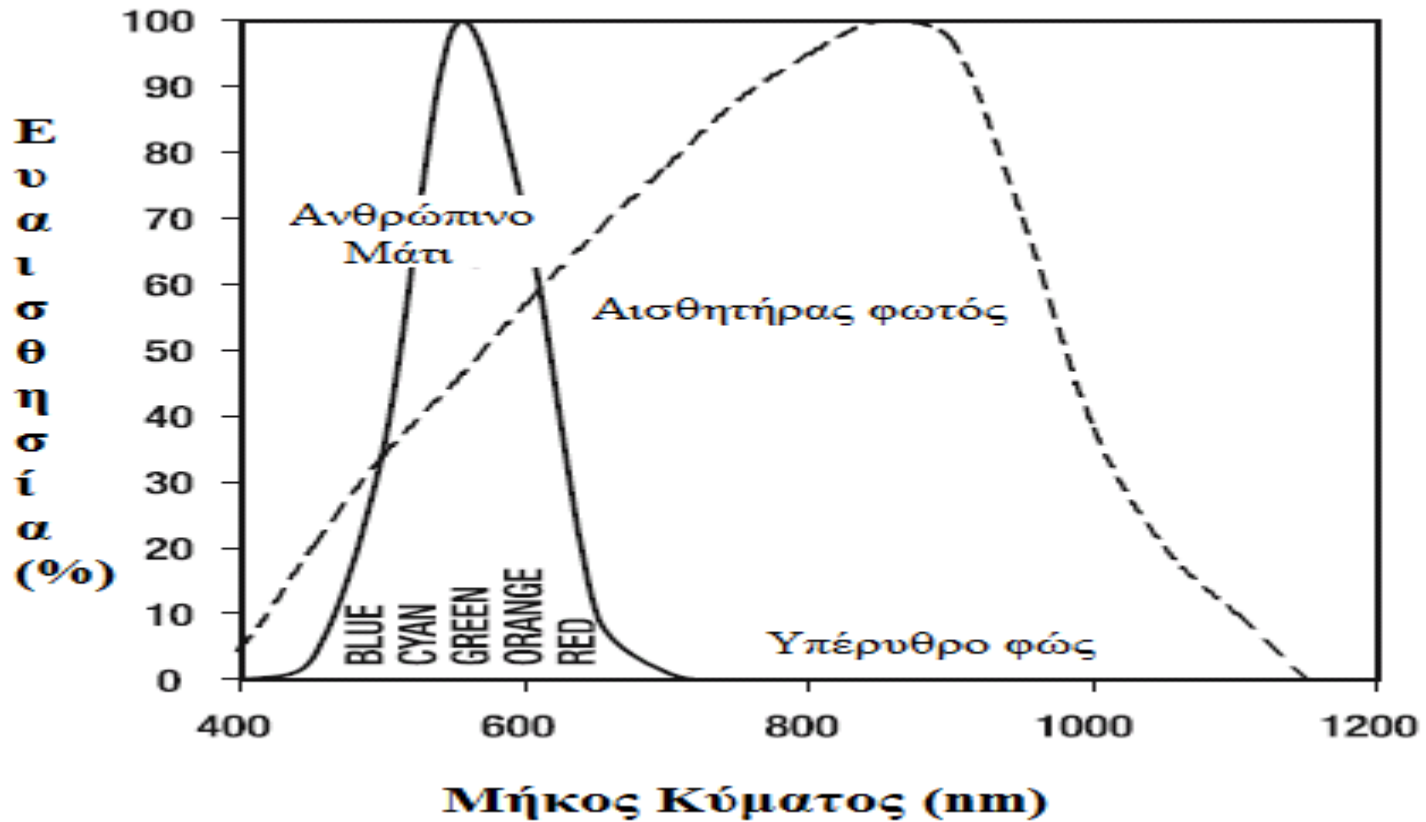
5. Σερβοκινητήρας (Rotation Sensor)



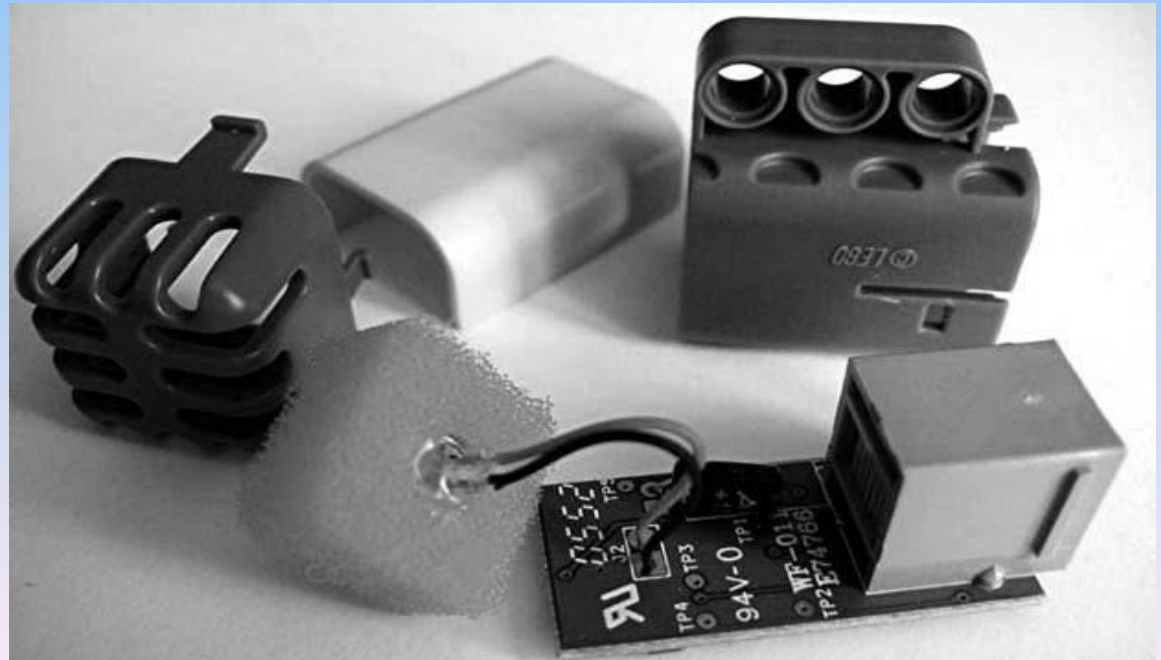
Αισθητήρας Αφής και Φωτός



Ευαισθησία Αισθητήρα Φωτός

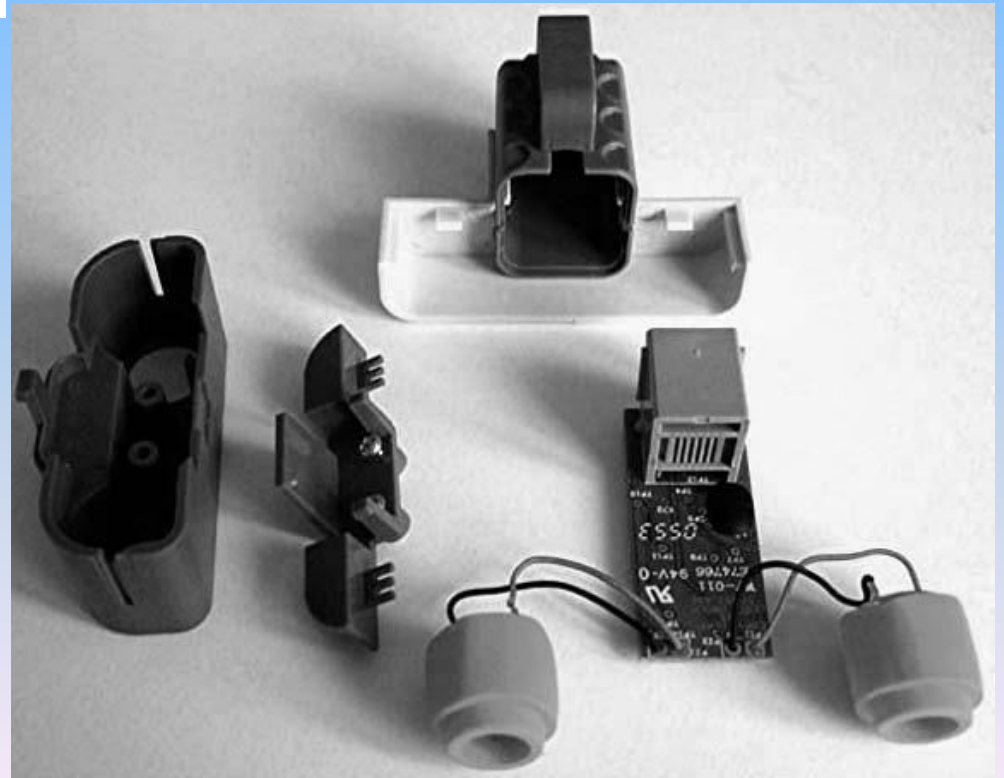


Αισθητήρας Ήχου

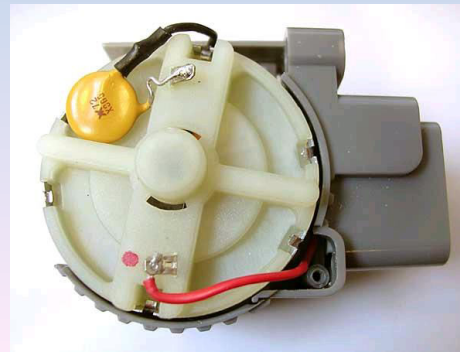
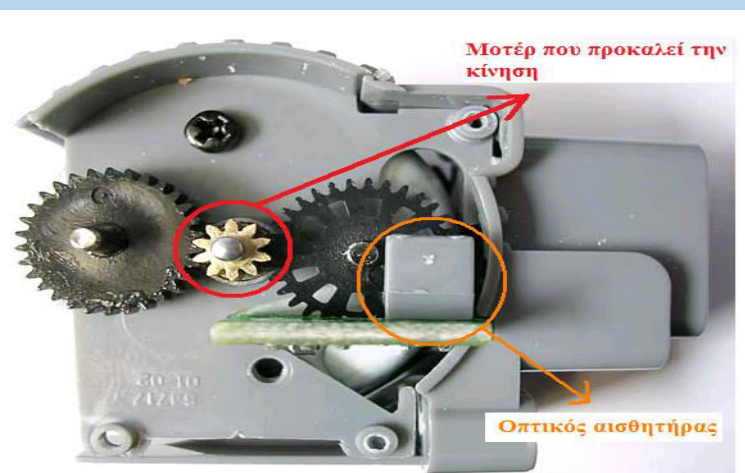
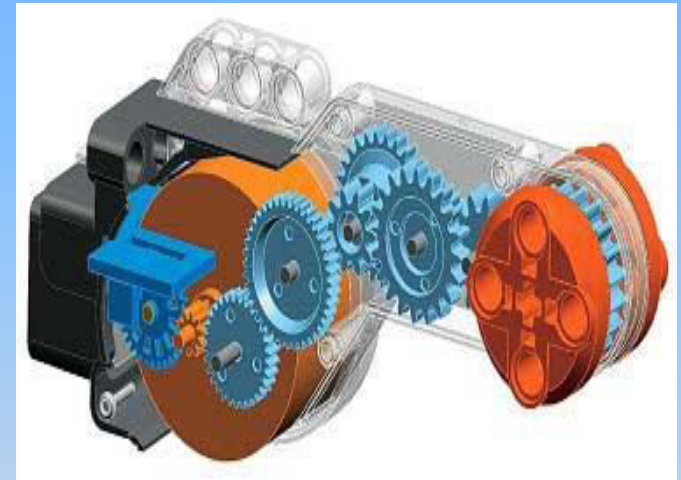
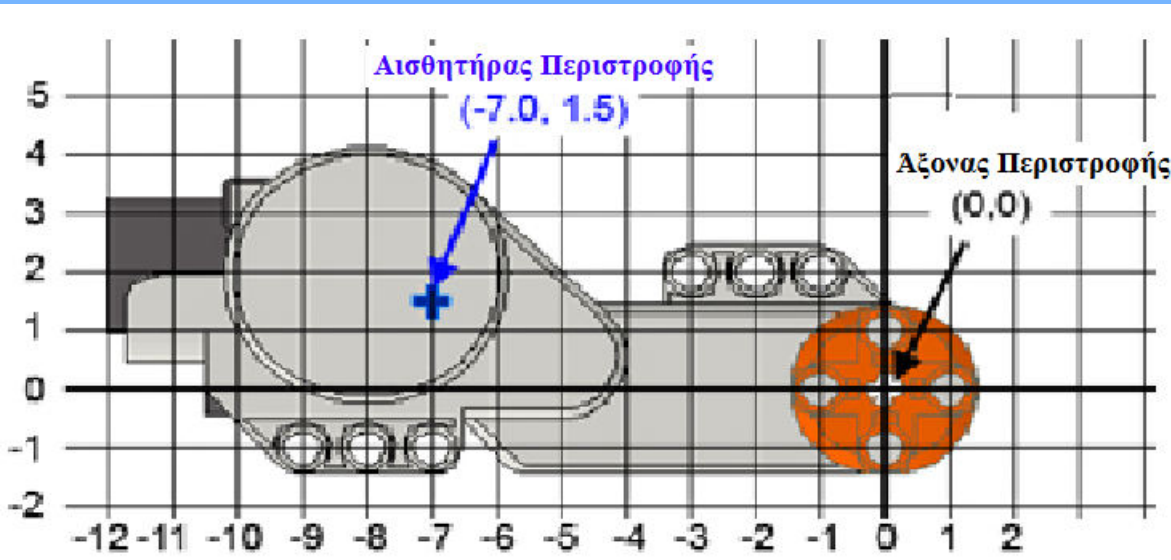




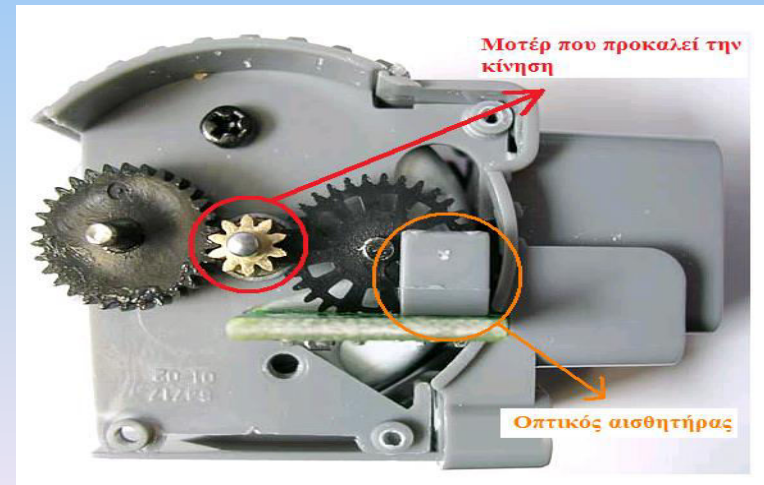
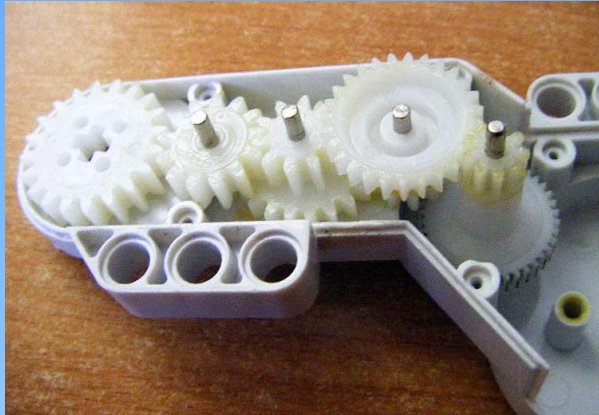
Αισθητήρας Υπερήχων



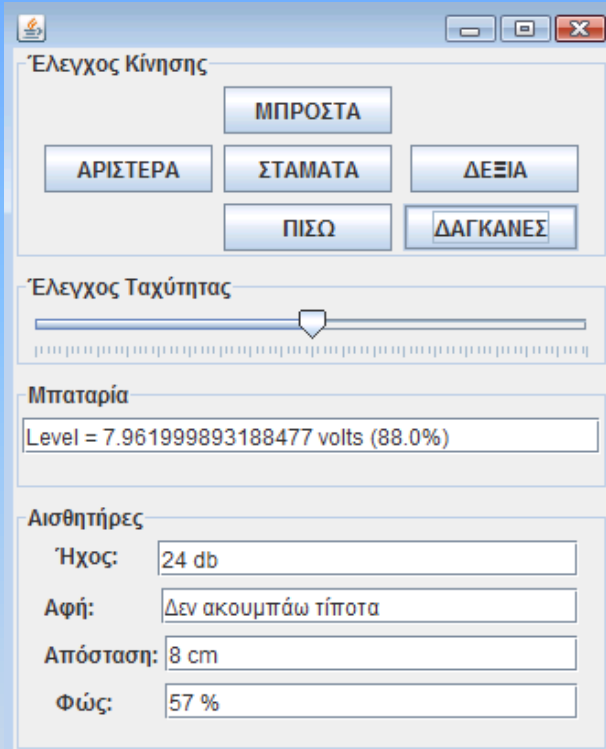
Μοτέρ Κίνησης



Το Εσωτερικό του Μοτέρ Κίνησης



Java (1 από 4)



✓ Πρόγραμμα: NetBeans IDE 6.5

✓ Βιβλιοθήκες: i-command 0.7 (Brian Bagnall)

Bluecove-2.1.0

JDK 1.6 (Java Development Key)

✓ Sensors:

Sound Sensor

Touch Sensor

Ultrasonic Sensor

Light Sensor

Motor A,B

Java (2 από 4)

✓ Ενδεικτικές εντολές στη βιβλιοθήκη i-command

• *getVoltage (Returns Battery level in Volts)*

```
public static getVoltage()
```

• *isPressed (Returns true if sensor is pressed)*

```
public TouchSensor(SensorPort sensor)
```

```
public boolean isPressed()
```

• *getDistance (Returns the distance to the closest object in centimeters)*

```
public UltrasonicSensor(SensorPort s)
```

```
public int getDistance()
```

• *getLightPercent (Returns light reading as a percentage, 0 = dark, 100 = bright)*

```
public LightSensor(SensorPort sensor) (LED light is active by default)
```

```
public int getLightPercent()
```

Java (3 από 4)

✓ Ενδεικτικά κομμάτια κώδικα

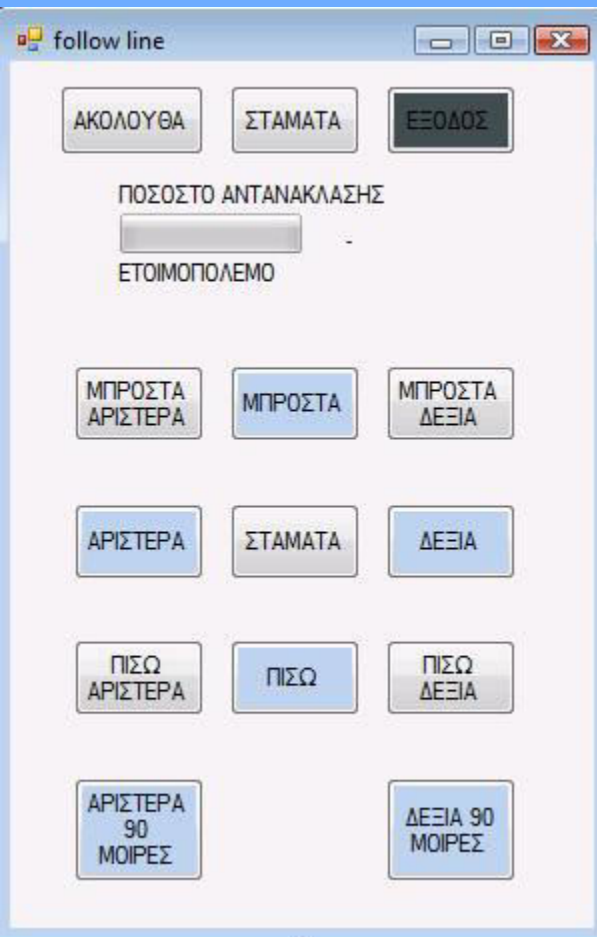
```
Import icommand.nxt.*;
public class NXTRobot{
    private TouchSensor _touch;
    private LightSensor _light;
    private UltrasonicSensor _usonic;
    public NXTRobot(){
        NXTCommand.open();    //Επικοινωνία με το NXT
        _touch = new TouchSensor(SensorPort.S1);
        _light = new LightSensor(SensorPort.S3);
        _usonic = new UltrasonicSensor(SensorPort.S4);
    }
    public void close(){
        NXTCommand.close();
    }
}
```


Java (4 από 4)

✓ Ενδεικτικά κομμάτια κώδικα

```
public boolean readTouchSensor(){
    return _touch.isPressed();
}
public int readLightLevel(){
    return _light.getLightPercent();
}
public int readDistance(){
    return _usonic.getDistance();
}
public void WriteTouchLevel(String s){
    touchTextField.setText(s);
}
public void WriteLightLevel(String s){
    lightTextField.setText(s);
}
Public void WriteDistanceLevel (String s){
    distanceTextField.setText (s );
}
```

VB.net (1 από 4)



✓ Πρόγραμμα: Visual Studio 2008

✓ Βιβλιοθήκη: NXT# (Bram Fokke)

✓ Sensors: Light Sensor
Motor A,B

VB.net (3 από 4)

✓ Ενδεικτικά κομμάτια κώδικα

```
Imports Bram.Lego
```

```
Public Class Form1
```

```
    #Region "Events"
```

```
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e  
        As System.EventArgs) Handles MyBase.Load
```

```
        NxtBrick1.Connect()
```

```
    End Sub
```

```
    Private Sub Form1_FormClosed(ByVal sender As Object, ByVal e  
        As System.Windows.Forms.FormClosedEventArgs) Handles Me.FormClosed
```

```
        NxtBrick1.Disconnect()
```

```
    End Sub
```

```
    #End Region
```

VB.net (4 από 4)

✓ Ενδεικτικά κομμάτια κώδικα

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles 1.Click
```

```
    NxtMotorA.Turn(20, 1022)
```

```
    NxtMotorB.Turn(20, 1022)
```

```
End Sub
```

$$1c=360^{\circ}=17.6cm(\text{περ.Ρόδας})$$

$$50/17.6=2.84c$$

$$2.84*360^{\circ}=1022^{\circ}$$

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
```

```
    NxtMotorA.Turn(0, 0)
```

```
    NxtMotorB.Turn(20,369)
```

```
End Sub
```

```
End Class
```

$$AΞΟΝΑΣ(AKT)=11.5 \text{ cm}$$

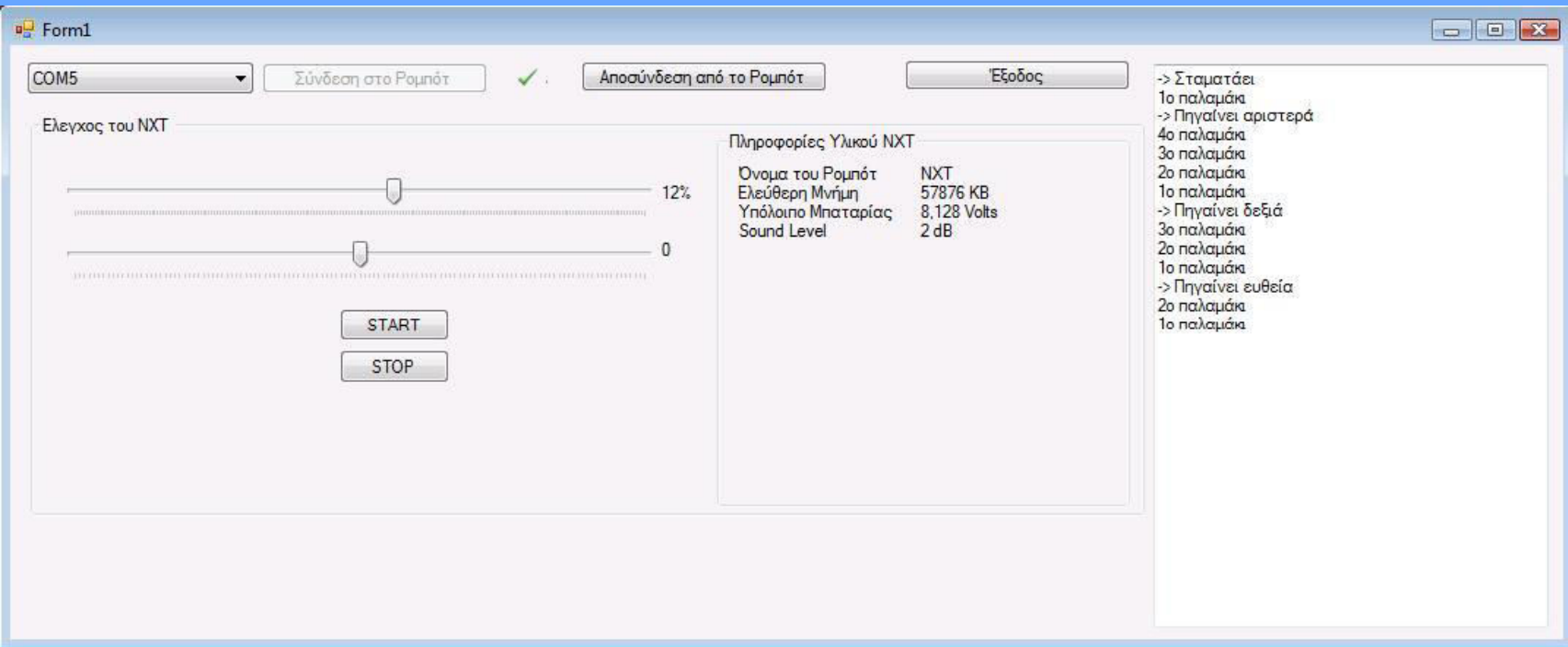
$$\Delta=2*\pi*R=72.22 \text{ cm}$$

$$\Delta/4=18.055 \text{ cm}$$

$$18.055cm/17.6cm=1.026$$

$$1.026*360^{\circ}=369.3^{\circ}$$

C# (1 από 3)



✓ Πρόγραμμα: Visual Studio 2008

✓ Βιβλιοθήκη: Mindsqualls V 1.2 (Niels Handest)

✓ Sensors: Sound Sensor
Motor A,B

C# (2 από 3)

✓ Ενδεικτικές εντολές στη βιβλιοθήκη Mindsqualls V 1.2

• NXTBrick connect via Bluetooth

```
NxtBluetoothConnection conn = new NxtBluetoothConnection (40);  
conn.Connect();  
brick.Disconnect ();
```

• Create a Motor

```
NxtMotor motor = new NxtMotor ();
```

• Connect to the port

```
brick.Motorb = motor;
```

• Run it at 75% power

```
motor.Run (75, 360);
```

• Synchronize two motors and Run them at 75% power

```
NxtMotorSync moterPair=new NxtMotorSync (brick.MotorB, brick.MotorC);  
moterPair.Run (Power[0,100], Degrees, TurnRatio[-100,100])
```

C# (3 από 3)

✓ Ενδεικτικά κομμάτια κώδικα

```
using NKH.MindSqualls;
public Form1()      {
    NxtBluetoothConnection conn = new NxtBluetoothConnection (40);
    conn.Connect();
    NxtSoundSensor soundsensor = new NxtSoundSensor ();
    Brick.Sensor1 = soundSensor;
    soundSensor.dBA = true;
    NxtGetDeviceInfoReply? devinfo = brick.CommLink.GetDeviceInfo();
    lblRobotName.Text = devinfo.Value.nxtName;
    lblFreeFlash.Text = ""+devinfo.Value.freeUserFlash + " KB";
    lblBatteryLevel.Text = ""+brick.CommLink.GetBatteryLevel().Value/1000.0 + " Volts";
    grpHardwareInfo.Refresh();
    label8.Text = "" +soundSensor.SoundLevel.Value+ " dB";
    brick.Disconnect ();
}
```


Συμπεράσματα-Δυσκολίες

Έπειτα από την ανηφόρα που λέγεται πτυχιακή, βγάλαμε κάποια συμπεράσματα και θα θέλαμε να τα μοιραστούμε μαζί σας.

- ✓ Το Lego NXTRobot, είναι το κατάλληλο μέσο για να ασχοληθεί κάποιος, σοβαρά με τη Ρομποτική
- ✓ Συνιστούμε την Java, διότι:
 - Τη βρήκαμε πιο ενδιαφέρουσα
 - Η i-command, υποστηρίζει το μεγαλύτερο εύρος εντολών για τους αισθητήρες που κυκλοφορούν σήμερα για το NXT
 - Έχει forum υποστήριξης
 - Μπορούμε να επέμβουμε άμεσα στα σωθικά του Lego NXT και να αλλάξουμε το Firmware αλλά και τα προγράμματα που περιέχει προεγκατεστημένα από τη Lego
- ✓ Μία σχετικά δύσκολη υπόθεση ήταν η εύρεση βιβλιοθηκών και η προσαρμογή τους στα Προγραμματιστικά περιβάλλοντα, και ειδικά στο Netbeans της Java
- ✓ Τέλος πήρε πολύ χρόνο να κατασκευάσουμε το ρομπότ έτσι ώστε να αξιοποιήσουμε με τον καλύτερο δυνατό τρόπο τους αισθητήρες

Ευχαριστούμε πολύ για το
πολύτιμο χρόνο σας,
καλό ακαδημαϊκό έτος.

Με εκτίμηση,
ΝΤΕΛΗΣ ΓΕΩΡΓΙΟΣ
και
ΚΥΡΙΑΚΙΔΗΣ ΧΡΗΣΤΟΣ