



Warsztaty Lego Mindstorms #2

Kontakt

- Mateusz Kaczor:

matiduck@gmail.com

- Mikołaj Buchwald:

mikolaj.buchwald@gmail.com

Plan gry

- Czy wszystko wszystkim działa?
- Five Minute Bot.
- Klasa Motor.
- Klasa LCD.
- Konkurs czy Machina Speculatrix?

Five Minute Bot



http://www.nxtprograms.com/five_minute_bot/steps.html

Klasa Motor.

Źródło:

<http://www.lejos.org/nxt/nxj/tutorial/MotorTutorial/ControllingMotors.html>

API dla Klasy Motor:

- http://www.lejos.org/p_technologies/nxt/nxj/api/lejos/nxt/Motor.html

Przypomnienie!:

leJOS udostępnia obiekty odpowiadające motorom podpiętym do portów:

- Motor.A
- Motor.B
- Motor.C

Należy pamiętać, że włączony motor będzie działać tak długo, aż nie zostanie zatrzymany, albo nie skończy się program

Motor #1

- Użyte metody:
 - Motor.forward()
 - Motor.backward()
 - Motor.Stop()
 - Button.waitForPress()
 - LCD.drawString(String str, int x, int y)

Wskazówka:” x ranges from 0 to 15, and y from 0 to 7.”

- 1.Display "Program 1 " in row 0 of the LCD.
- 2.Wait for a button to be pressed
- 3.Run motor A in the forward direction.
- 4.Display FORWARD in the top line.
- 5.Wait until a button is pressed.
- 6.Run the motor backward.
- 7.Display BACKWARD in next line.
- 8.Wait until a button is pressed.
- 9.Stop the motor.

Motor #2 – użycie tachometru.

- „The NXT motor has a built in tachometer that keeps track of the current angle (in degrees) of the motor axle.”

- Użyte metody:
 - `Motor.getTachoCount()`
 - `setSpeed(int speed)` – stopnie na sekundę.
 - `Delay.msDelay(int interval)`
- Wskazówka:
Użycie pętli while.

1. Display the program number as before.
2. Set the motor speed to 2 rev/sec
3. Run Motor.A forward.
4. Wait for 2 seconds.
5. Display the motor angle on the LCD. (what should it be?)
6. Stop the motor.
7. Display the tachometer reading on the on the next line LCD.
8. Start the motor rotating backward.
9. Wait till the tacho count reaches 0.
10. Display the tacho count on the next line.
11. Stop the motor.
12. Display the tacho count on the next line.
13. Wait for a button press so you can read the LCD.

Motor #3 - obrót

- „The basic method returns only when the rotation is complete(rotate), the other returns immediately but the motor stops when the rotation is completed(rotateTo).”

Użyte metody:

```
Motor.rotate(int angle);  
Motor.rotateTo(int angle);
```

1. Display the program name and wait for a button press.
2. Rotate the motor 4 complete revolutions.
3. Display the tachometer reading on the on the LCD.
4. Rotate the motor to angle 0.
5. Display the tachometer reading on the on the LCD, next row
6. Wait for a button press.

Motor #4 -przerywanie obrotu.

Użyte metody:

```
Motor.rotate(int angle);
```

```
Motor.rotateTo(int angle);
```

```
(boolean)isRotating();
```

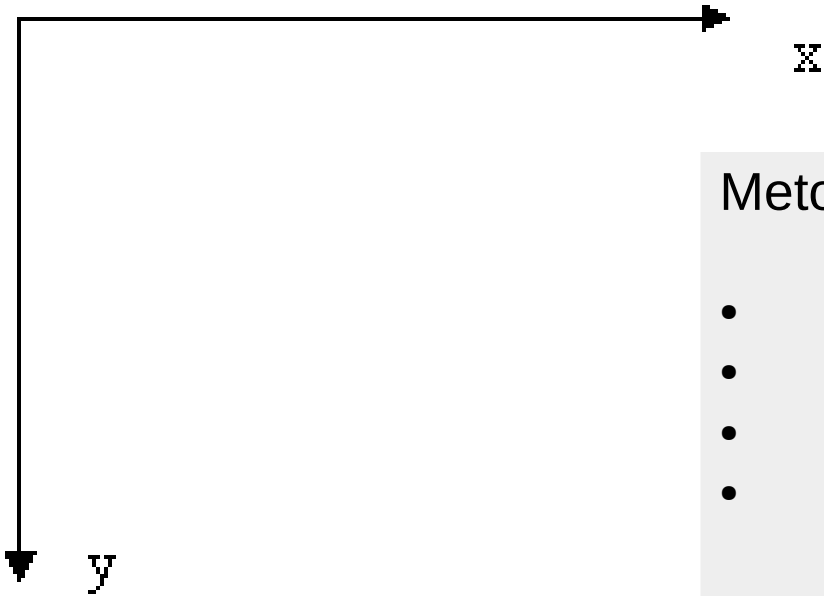
- 1.Display the program number.
- 2.Start a rotation of 4 revolutions backward.
- 3.While the motor is rotating, display the tachometer count.
- 4.When a button is pressed, stop the motor.
- 5.After the motor has stopped, display the tachometer count in another row.
- 6.Wait for a button press.
- 7.

Motor #5 -prędkość

```
public class MotorTutor5 {
```

```
public static void main(String[] args) {  
    LCD.drawString("Progam 5", 0, 0);  
    Button.waitForAnyPress();  
    LCD.clear();  
    int speed = 720;  
    Motor.A.setSpeed(speed);  
    Motor.B.setSpeed(speed);  
    Motor.C.setSpeed(speed);  
    Motor.A.rotate(720,true);  
    Motor.B.rotate(720,true);  
    Motor.C.rotate(720,true);  
    for (int i = 0 ; i < 8 ; i++)  
    {  
        Delay.msDelay(200);  
        LCD.drawInt(Motor.A.getTachoCount(), 0,i);  
        LCD.drawInt(Motor.B.getTachoCount(), 6,i);  
        LCD.drawInt(Motor.A.getTachoCount(), 12,i);  
    }  
    while(Motor.A.isMoving()|| Motor.B.isMoving()||Motor.C.isMoving());  
    Button.waitForAnyPress();  
}
```

LCD #1



Metody:

- `void drawString(String str, int x, int y);`
 - `void drawInt(int i, int x, int y);`
 - `void drawInt(int i, int places, int x, int y)`
 - `void clear()`
-
- Słowo, dobry: `System.out.println(String str);`

LCD #2

Przykład:

```
import lejos.nxt.LCD;
public class LCDTest {
    public static void main(String[] args) throws
Exception {
        LCD.drawString("Free RAM:", 0, 0);
        LCD.drawInt((int)
System.getRuntime().freeMemory(), 6, 9, 0);
        Thread.sleep(2000);
    }
}
```

LCD #3 - Grafika

API Graphic:

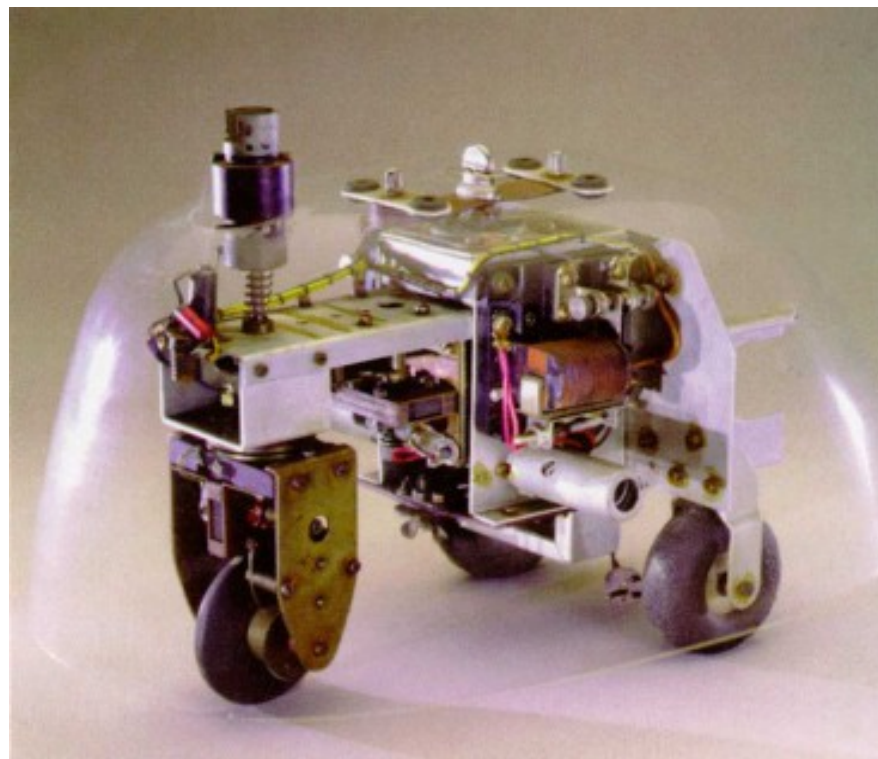
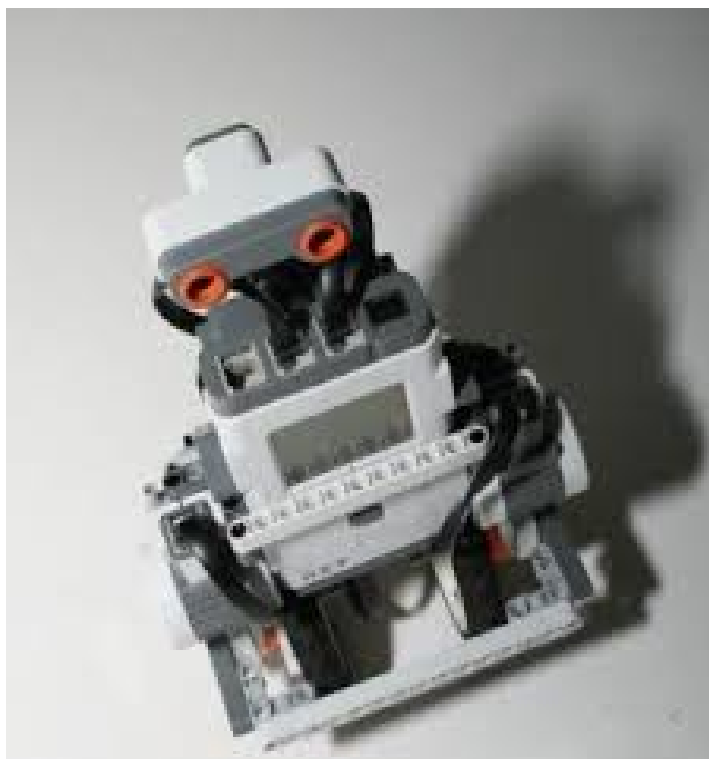
<http://www.lejos.org/nxt/nxj/api/javax/microedition/lcd/LcdGraphics.html>

```
import javax.microedition.lcd.Graphics;

public class GraphicsSample {

    public static void main(String [] options) throws
    Exception {
        Graphics g = new Graphics();
        g.drawLine(5,5,60,60);
        g.drawRect(62, 10, 25, 35);
        Thread.sleep(2000);
    }
}
```

Konkurs czy Machina?



Koniec.

- Zaprogramuj robota tak, aby podróżował po kwadracie. Robot powinien wykonać co najmniej trzy okrążenia pod rząd. Dodatkowo powinien zaczynać i kończyć zadanie w tym samym punkcie.
- Jak wyżej, ale tor ma kształt koła o wyznaczonej średnicy.