

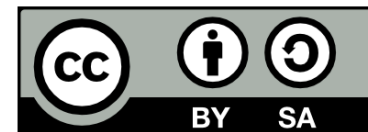
Tento vzdělávací materiál vznikl v rámci projektu
CZ.02.3.68/0.0/0.0/16_036/0005322 **Podpora rozvíjení informatického myšlení.**



EVROPSKÁ UNIE
Evropské strukturální a investiční fondy
Operační program Výzkum, vývoj a vzdělávání



Podléhá licenci Creative Commons Uveďte původ-Zachovejte licenci 4.0



LEGO Mindstorms EV3

Obsah

LEGO Mindstorms EV3

Programování

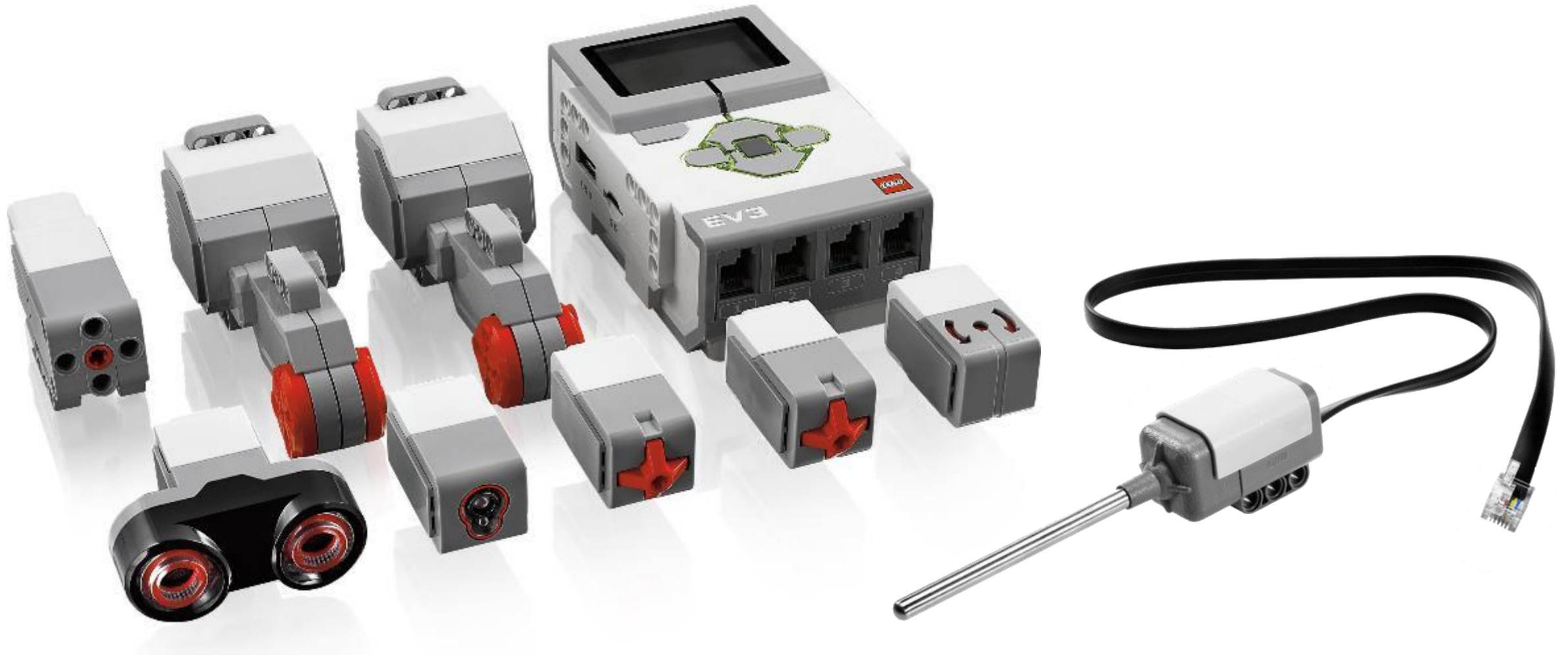
Vizuální programování

Programovací jazyk Java

LEGO Mindstorms EV3

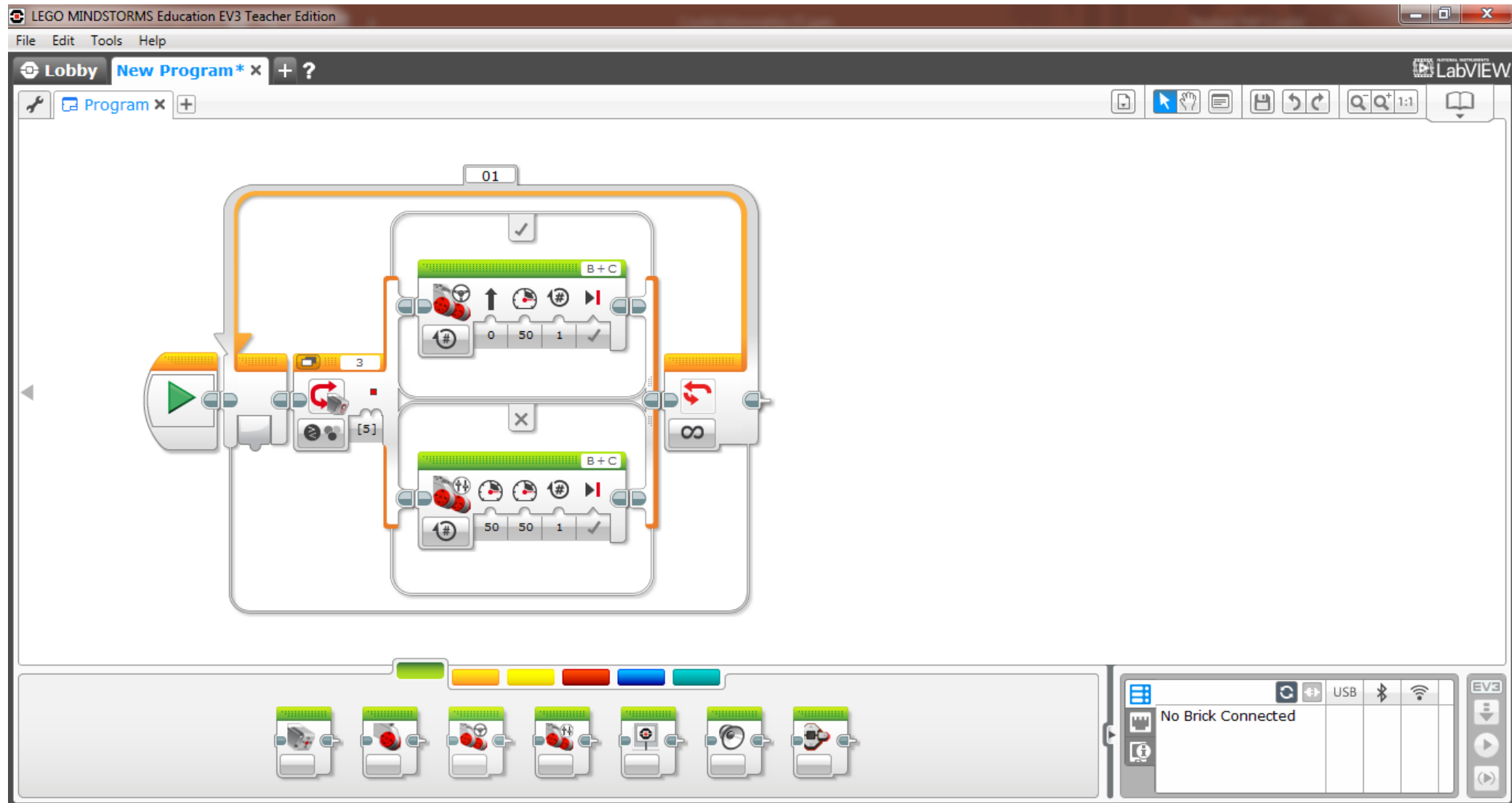


LEGO Mindstorms EV3 - senzory





Programování LEGO Mindstorms EV3






Programování LEGO Mindstorms EV3

The screenshot displays the LEGO Mindstorms EV3 software interface, which is used for programming the EV3 brick. The interface is divided into several sections:

- Top Bar:** Includes the "Beta" label, "LEGO education" logo, "home" and "share" buttons, a toggle between "Blocks" and "JavaScript" programming modes, and a "Microsoft" logo.
- Left Panel:** Shows a diagram of the EV3 brick with its ports and connected components. The ports are labeled 1, 2, 3, and 4. Port 1 is connected to a motor (50% power), port 2 to a sensor (50% power), port 3 to a motor (50% power), and port 4 to a sensor (50% power). The brick itself is labeled "EV3".
- Search Bar:** A search bar with the text "Search..." and a magnifying glass icon.
- Category List:** A list of categories for selecting blocks: Brick, Sensors, Motors, Music, Loops, Logic, Variables, and Math.
- Program Canvas:** The main area for building the program. It shows a sequence of blocks:
 - A green "forever" loop block.
 - An "on start" block containing:
 - A green "run large motor" block with port "A" and power "50 %".
 - A green "run large motor" block with port "C" and power "50 %".
 - A purple "pause until touch" block with port "1" and "pressed" condition.

At the bottom of the interface, there is a "Download" button, a text field labeled "Untitled", and a set of navigation controls (undo, redo, zoom in, zoom out).

Programování LEGO Mindstorms EV3

```
31 lines (22 sloc) | 987 Bytes | Raw | Blame | History |   
1 //*****
2 //Wei Lu Java Robotics Programming with Lego EV3      Hello World.java
3 //An example to display HelloWorld on the LCD screen of EV3 brick
4 //*****
5
6 // import EV3 hardware packages for EV brick finding, activating keys and LCD
7 import lejos.hardware.ev3.EV3;
8 import lejos.hardware.BrickFinder;
9 import lejos.hardware.Keys;
10 import lejos.hardware.lcd.TextLCD;
11
12 public class HelloWorld {
13
14     public static void main(String[] args) {
15
16         // get EV3 brick
17         EV3 ev3brick = (EV3) BrickFinder.getLocal();
18
19         // instantized LCD class for displaying and Keys class for buttons
20         Keys buttons = ev3brick.getKeys();
21         TextLCD lcddisplay = ev3brick.getTextLCD();
22
23         // drawing text on the LCD screen based on coordinates
24         lcddisplay.drawString("Hello World", 2, 4);
25
26         // exit program after any button pressed
27         buttons.waitForAnyPress();
28     }
29
30 }
```



	Refresh	Connect	USB	Bluetooth	Wi-Fi	EV3
No Brick Connected						

EV3



			USB			EV3
No Brick Connected						



EV3

No Brick Connected

USB Bluetooth Wi-Fi

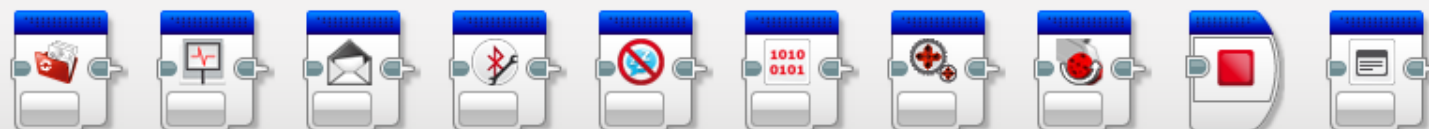
EV3

Download Run Stop



	USB	Bluetooth	WiFi
No Brick Connected			

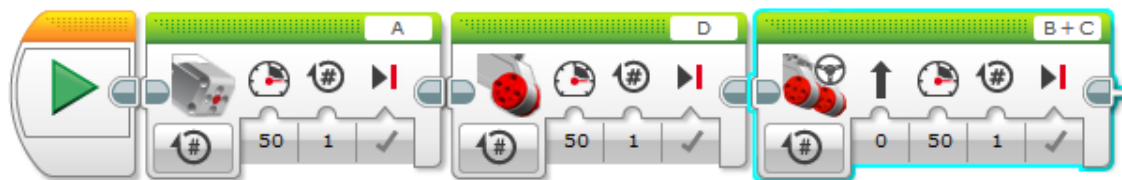
EV3



	Refresh	Connect	USB	Bluetooth	WiFi	EV3
No Brick Connected						



	Refresh	Connect	USB	Bluetooth	Wi-Fi	EV3
No Brick Connected						



EV3

No Brick Connected

USB Bluetooth Wi-Fi

EV3



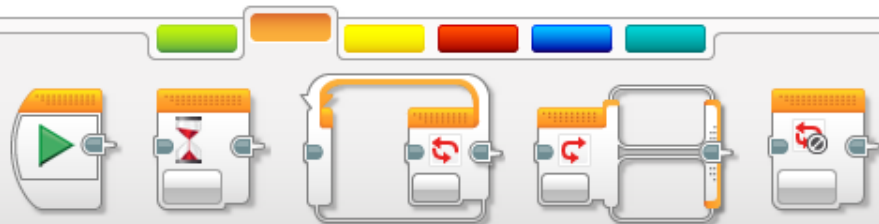
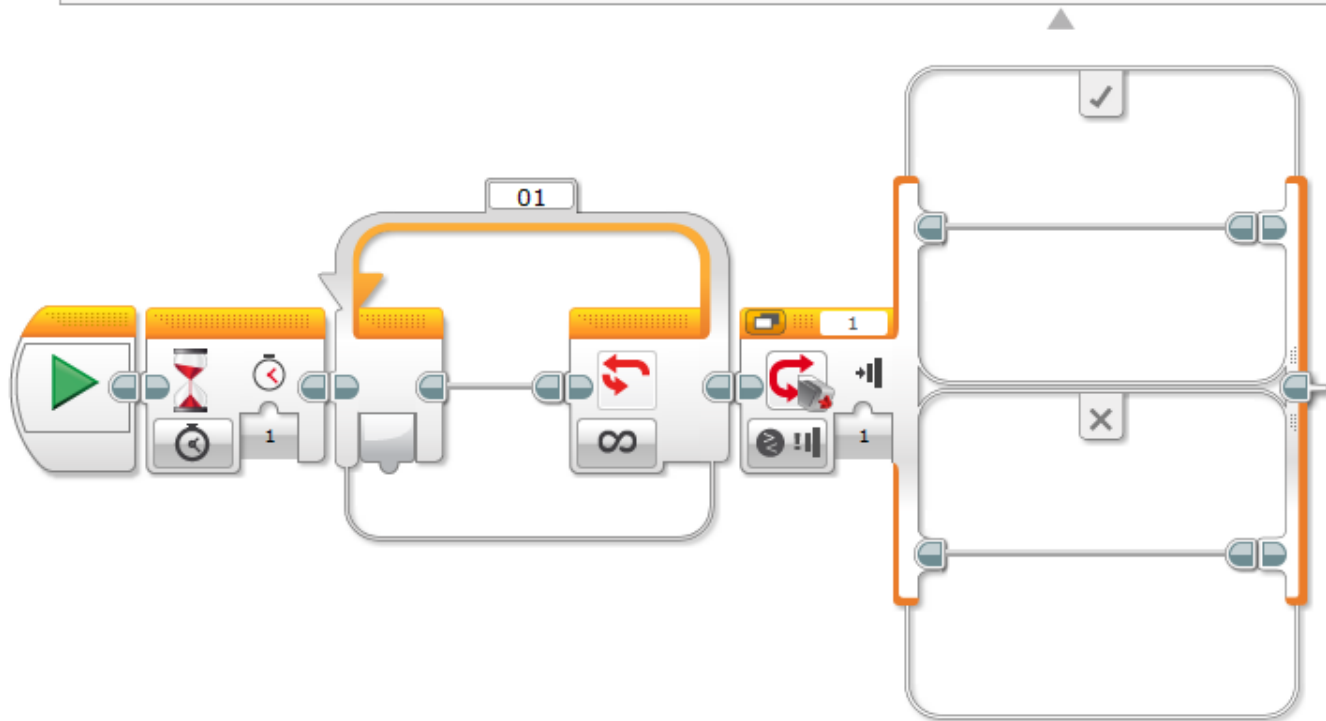
	USB	Bluetooth	WiFi
No Brick Connected			

EV3

Download

Play

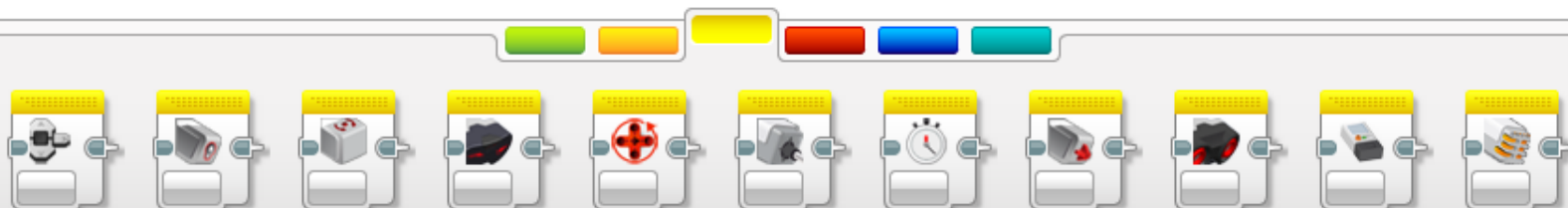
Stop



EV3

No Brick Connected

USB Bluetooth Wi-Fi

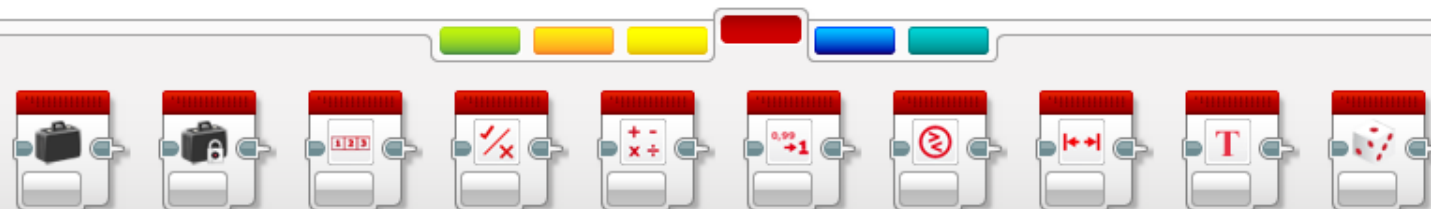
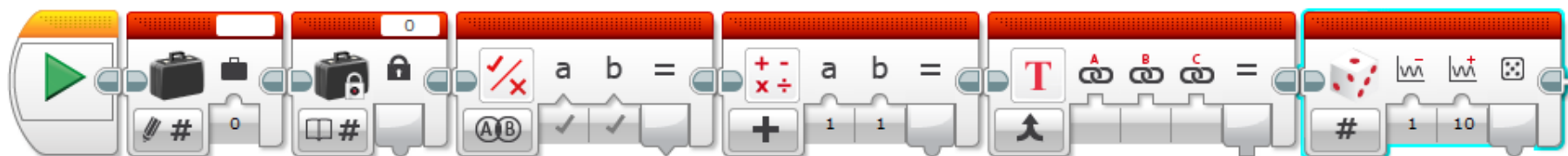


EV3

No Brick Connected

USB Bluetooth Wi-Fi

EV3

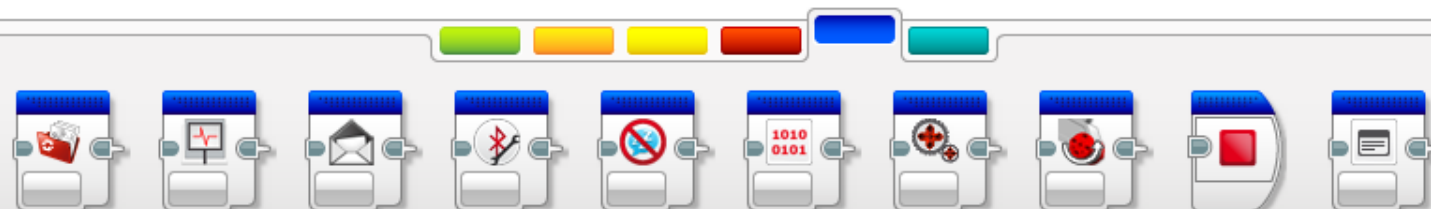
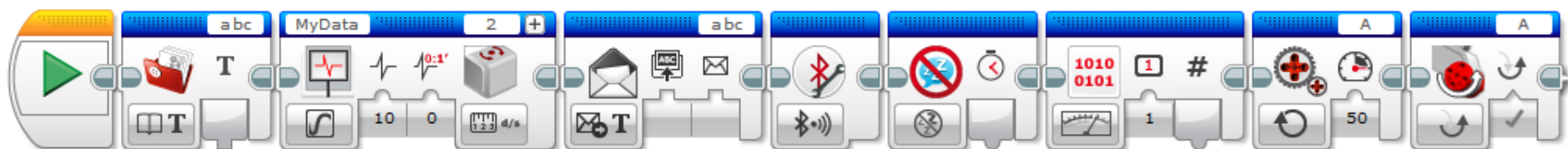


EV3

No Brick Connected

USB Bluetooth Wi-Fi

EV3



EV3

No Brick Connected

USB Bluetooth Wi-Fi

EV3

Programování Java - úvod

```
package ev3.exercises;  
  
import lejos.hardware.Button;  
import lejos.hardware.Sound;  
import lejos.hardware.motor.*;  
import lejos.hardware.port.*;  
import lejos.utility.Delay;
```

Programování Java - úvod

```
import lejos.hardware.*;

public class HelloWorld
{
    public static void main(String[] args)
    {
        System.out.println("Hello World!!");

        Button.waitForAnyPress();
    }
}
```


Programování Java - úvod

```
System.out.println("Drive Forward\nand Stop\n");  
System.out.println("Press any key to start");  
  
Button.LEDPattern(4);      // rozsvícení zelené LED  
Sound.beepSequenceUp();    // zvukový signál  
  
Button.waitForAnyPress();
```

Programování Java - motory

```
// vytvoření objektů motorů

UnregulatedMotor motorA = new UnregulatedMotor(MotorPort.A);
UnregulatedMotor motorB = new UnregulatedMotor(MotorPort.B);

// nastavení výkonu motorů na 50%
motorA.setPower(50);
motorB.setPower(50);

Delay.msDelay(2000); // doba jízdy motorů 2 vteřiny

motorA.stop(); // zastavení motorů
motorB.stop();

motorA.close(); // uzavření objektů motorů
motorB.close();
```

Programování Java - motory

```
motorA.setPower(50);  
motorB.setPower(50);
```

```
motorA.backward();  
motorB.forward();
```

```
motorA.setPower(70);  
motorB.setPower(30);
```

Programování Java - motory

```
// vytvoření objektů motorů
EV3LargeRegulatedMotor motorA = new EV3LargeRegulatedMotor(MotorPort.A);
EV3LargeRegulatedMotor motorB = new EV3LargeRegulatedMotor(MotorPort.B);

// nastavení rotací o 500 stupeň/vteřina
motorA.setSpeed(500);
motorA.forward(); // začátek otáčení

motorB.setSpeed(500);
motorB.forward(); // začátek otáčení

// doba 3 vteřiny
Delay.msDelay(3000);

// zastavení motorů
motorA.stop();
motorB.stop();
```

Programování Java - motory

```
// ukázka otáčení motorů, nejprve se otáčí první, potom druhý motor  
motorA.rotate(360);  
motorB.rotate(360);
```

```
// ukázka otáčení motorů, otáčejí se oba současně  
motorA.rotate(360, true);  
motorB.rotate(360, true);
```

```
// ukázka otáčení motorů o přesný úhel  
motorA.resetTachoCount();  
motorB.resetTachoCount();
```

```
motorA.rotateTo(180, true);  
motorB.rotateTo(180, true);
```


Programování Java – Ultrazvukový senzor

```
public class UltraSonicDemo
{
    public static void main(String[] args)
    {
        float                range;
        UltraSonicSensor     uss = new
UltraSonicSensor(SensorPort.S4);

        Button.waitForAnyPress();

        range = uss.getRange();

        Lcd.print(7, "range=");

        while (range > .25)
        {
            Lcd.clear(7, 7, 10);
            Lcd.print(7, 7, "%.3f", range);
            Delay.msDelay(500);

            range = uss.getRange();

        }

        uss.close();

        Sound.beepSequence();

        Lcd.clear(7, 7, 10);
        Lcd.print(7, 7, "%.3f", range);

        Button.waitForAnyPress();
    }
}
```

Programování Java – Gyroskopický senzor

```
public class GyroDemo
{
    public static void main(String[] args)
    {
        GyroSensor    gyro = new GyroSensor(SensorPort.S2);

        System.out.println("Gyro Demo");
        System.out.println("Press to start");

        Button.waitForAnyPress();

        while (Button.ESCAPE.isUp())
        {
            Lcd.clear(5);
            Lcd.print(5, "angle=%d av=%.3f", gyro.getAngle(),
gyro.getAngularVelocity());

            Delay.msDelay(250);
        }

        gyro.close();
    }
}
```

Programování Java – Senzor barev

```
public class ColorDemo
{
    public static void main(String[] args)
    {
        ColorSensor    color = new ColorSensor(SensorPort.S3);

        System.out.println("Color Demo");
        Lcd.print(2, "Press to start");

        Button.waitForAnyPress();
        Button.LEDPattern(0);

        // řízení programu pomocí tlačítka

        while (Button.ESCAPE.isUp())
        {
            Lcd.clear(4);
            Lcd.print(4, "ambient=%.3f", color.getAmbient());
            Delay.msDelay(250);
        }

        Delay.msDelay(1000);

        color.setRedMode();
        color.setFloodLight(Color.RED);
        color.setFloodLight(true);

        while (Button.ESCAPE.isUp())
        {
            Lcd.clear(5);
            Lcd.print(5, "red=%.3f", color.getRed());
            Delay.msDelay(250);
        }

        Delay.msDelay(1000);
    }
}
```

Programování Java – Senzor barev

```
color.setRGBMode();
color.setFloodLight(Color.WHITE);

Color rgb;

while (Button.ESCAPE.isUp())
{
    rgb = color.getColor();

    Lcd.clear(6);
    Lcd.print(6, "r=%d g=%d b=%d", rgb.getRed(),
rgb.getGreen(), rgb.getBlue());
    Delay.msDelay(250);
}

Delay.msDelay(1000);
```

```
color.setColorIdMode();
color.setFloodLight(false);

while (Button.ESCAPE.isUp())
{
    Lcd.clear(7);
    Lcd.print(7, "id=%s", ColorSensor.colorName
(color.getColorID()));
    Delay.msDelay(250);
}

color.close();

Sound.beepSequence();    // zvukový signál

Button.LEDPattern(4);
Button.waitForAnyPress();
}

}
```

Použité zdroje

Zdroje

<http://stemrobotics.cs.pdx.edu/node/4576?root=4576>

<http://www.lejos.org/ev3/docs/>

LEGO MINDSTORMS Education EV3 Teacher Edition

<https://makecode.mindstorms.com/#editor>

Zdroje obrázků

<https://www.lego.com/en-gb/mindstorms/downloads>

<https://roboter.schule/theorie/brick/>

<https://roboshop.lv/roboti/lego-mindstorms-education-ev3-core-set/>

<http://www.robomiku.ee/toode/nxt-temperatuuriandur/>

<https://github.com/Apress/beg-robotics-programming-in-java-w-lego-mindstorms/blob/master/Chapter%201/HelloWorld.java>

Tento vzdělávací materiál vznikl v rámci projektu
CZ.02.3.68/0.0/0.0/16_036/0005322 **Podpora rozvíjení informatického myšlení.**



EVROPSKÁ UNIE
Evropské strukturální a investiční fondy
Operační program Výzkum, vývoj a vzdělávání



Podléhá licenci Creative Commons Uveďte původ-Zachovejte licenci 4.0

