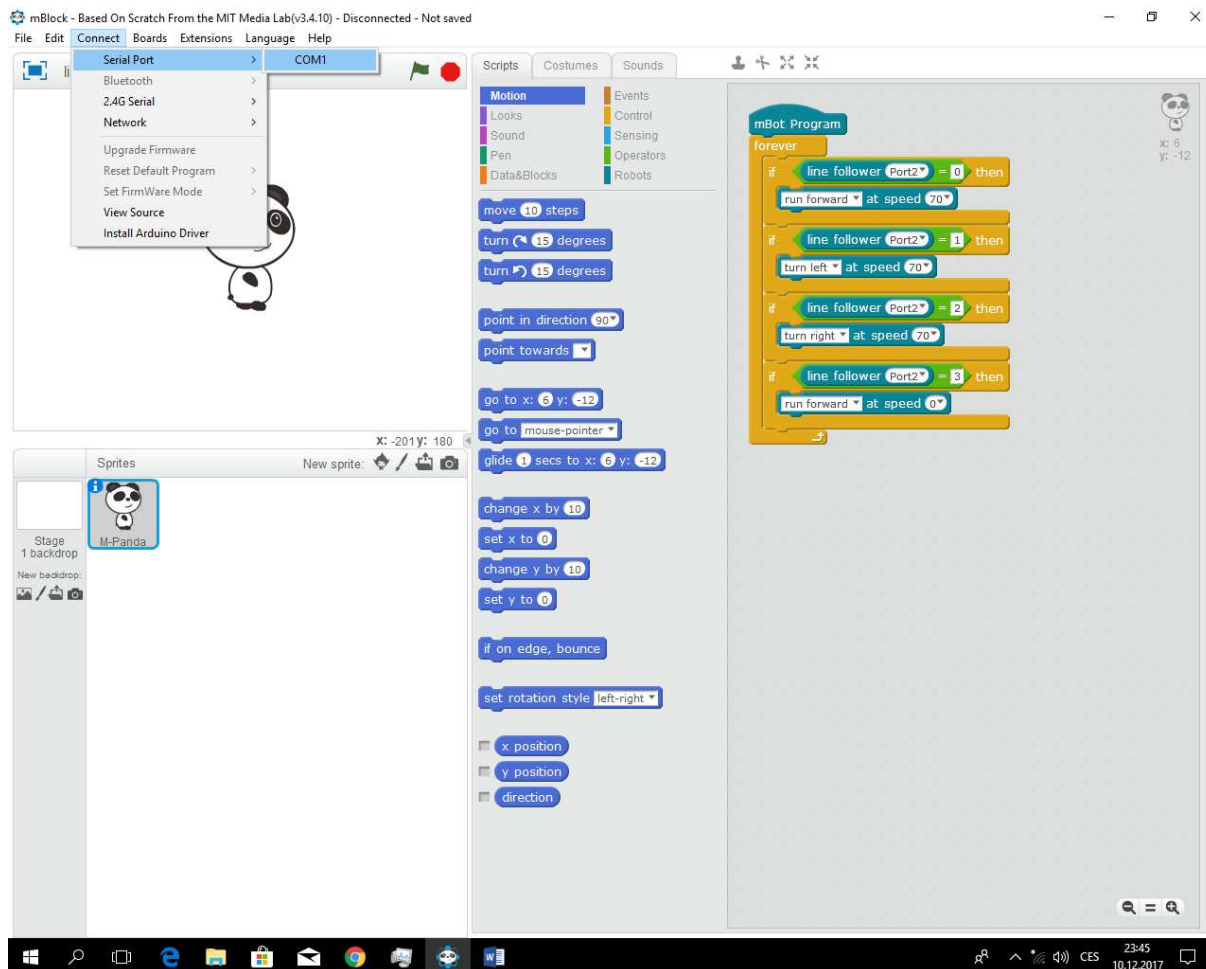


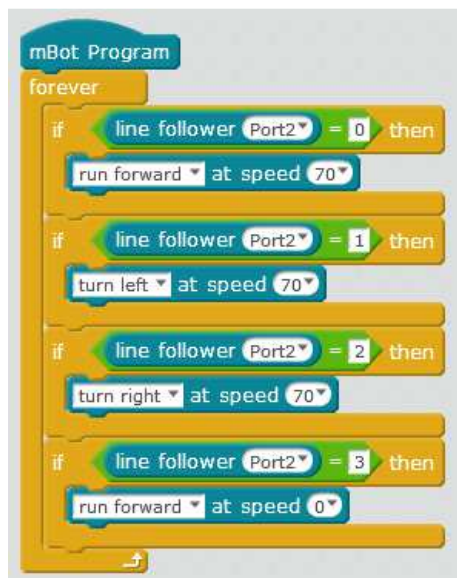
7. Kompilace programu a nahrávání do mBota pře USB kabel

Do USB portu se kabelem připojí zapnutý mBot a spustí se připojení.

Stále se jedná o sledování čáry pomocí line followeru.



Dále je potřeba celý program definovat jako kompilovatelný „mBot Program“.



Pro režim nahrávání je potřeba program přepnout do Arduino módu.

The screenshot shows the mBlock software interface. On the left, a Scratch script is visible, consisting of a 'forever' loop containing four 'if' blocks. Each 'if' block checks a 'line follower Port2' sensor (0, 1, 2, 3) and triggers a specific action: 'run forward at speed 70', 'turn left at speed 70', 'turn right at speed 70', and 'run forward at speed 0'. On the right, the Arduino code editor displays the corresponding C++ code. The code includes headers for Arduino, Wire, SoftwareSerial, and MeMCore. It defines two MeDCMotor objects (motor_9 and motor_10) and a 'move' function that takes a direction and speed as input. The 'move' function uses a series of 'if' and 'else if' statements to set the left and right motor speeds based on the direction. The 'setup' function is also defined but currently empty.

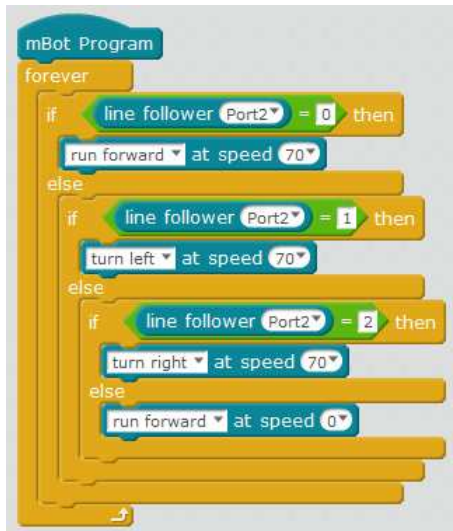
Následně pak stačí už jen zmáčknout tlačítko „Upload to Arduino“.

This is a close-up view of the Arduino code editor. The code is as follows:

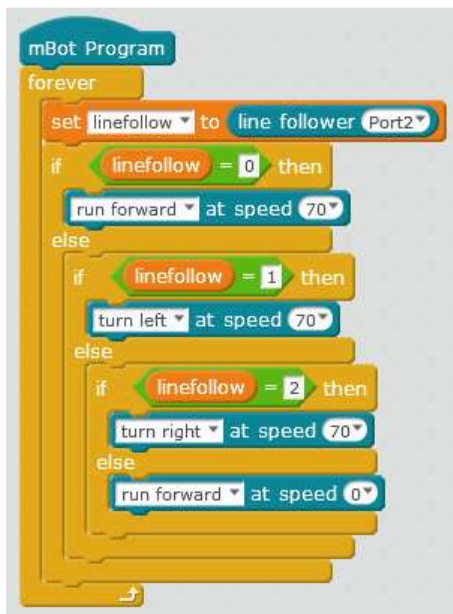
```
1 #include <Arduino.h>
2 #include <Wire.h>
3 #include <SoftwareSerial.h>
4
5 #include <MeMCore.h>
6
7 MeDCMotor motor_9(9);
8 MeDCMotor motor_10(10);
9 void move(int direction, int speed)
10 {
11     int leftSpeed = 0;
12     int rightSpeed = 0;
13     if(direction == 1){
14         leftSpeed = speed;
15         rightSpeed = speed;
16     }else if(direction == 2){
17         leftSpeed = -speed;
18         rightSpeed = -speed;
19     }else if(direction == 3){
20         leftSpeed = -speed;
21         rightSpeed = speed;
22     }else if(direction == 4){
23         leftSpeed = speed;
24         rightSpeed = -speed;
25     }
26     motor_9.run((9)==M1?- (leftSpeed) : (leftSpeed));
27     motor_10.run((10)==M1?- (rightSpeed) : (rightSpeed));
28 }
29 double angle_rad = PI/180.0;
30 double angle_deg = 180.0/PI;
31 MeLineFollower linefollower_2(2);
32
33 void setup() {
34 }
```

8. Úpravy programu směrem k procedurálnímu programování

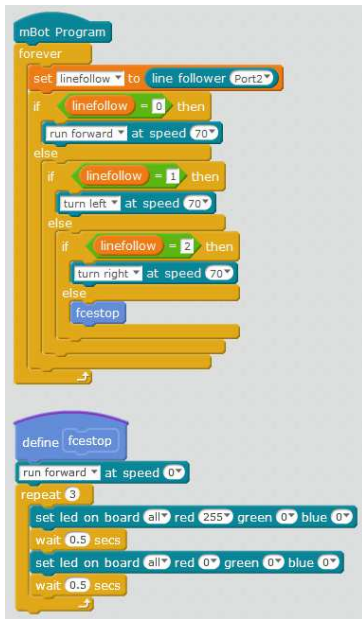
Celý program je upraven do **podoby větvení**.



Opakované odečítání hodnot z line followeru je **uloženo do proměnné**.



Jako zastavení je zde **definována funkce (block)** – fcestop, mBot zastaví a začne blikat červeně.



Každý pohyb je zde definován jako **samostatná funkce (block)** – fcestop, fceforward, fceright a fceleft.

