



Centrum
Mistrzostwa
Informatycznego

Zestawy mBot/CMI.

dr inż. Paweł Trajdos



Spis treści

Mikrokontrolery

Zestaw mBot

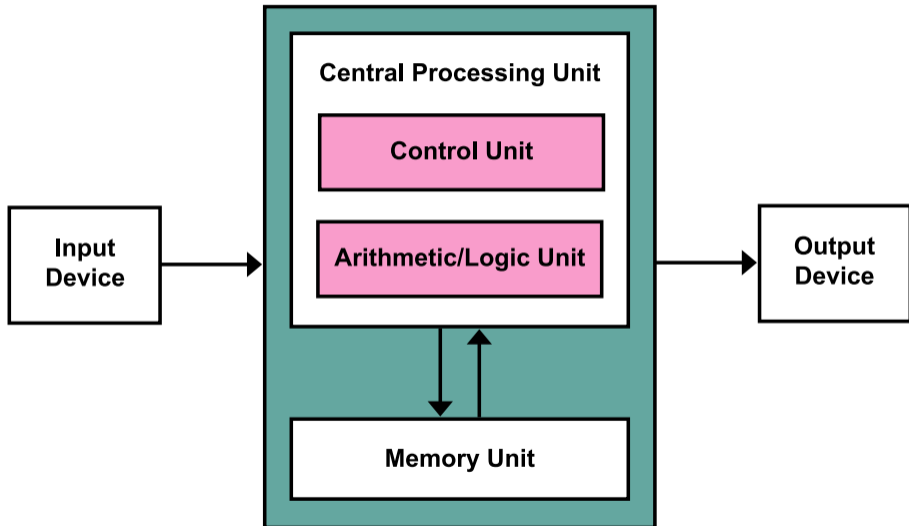
Programowanie zestawu

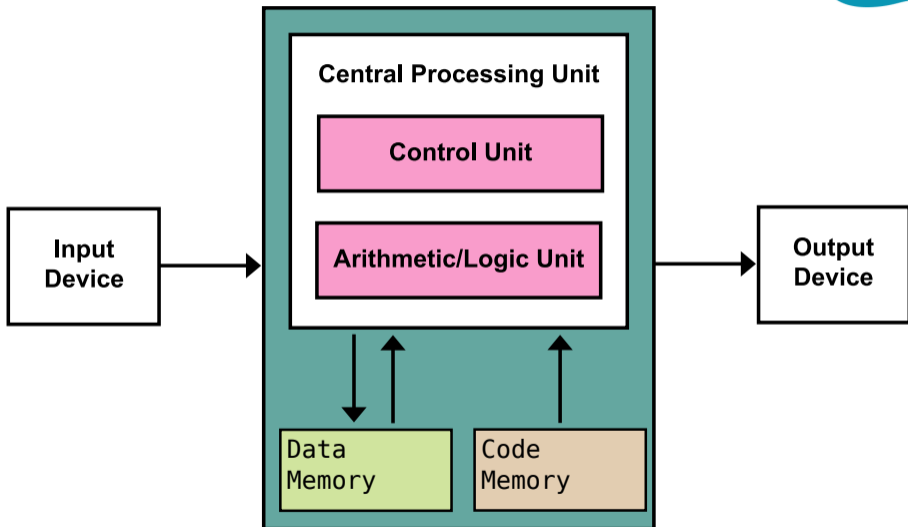
Section 1

Mikrokontrolery

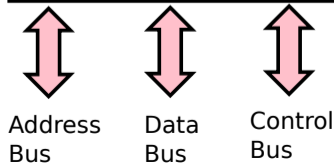
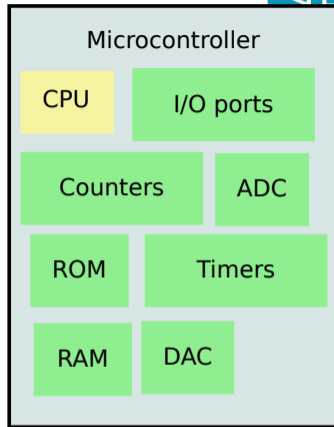
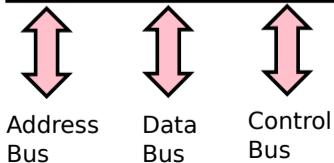
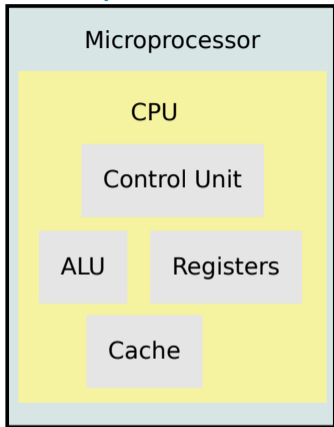
Model systemu komputerowego

Model Von Neumanna





Mikroprocesor a mikrokontroler



Section 2

Zestaw mBot

ATmega328

- ▶ Architektura Harwardzka
- ▶ RISC
- ▶ 131 instrukcji
- ▶ Pamięć:
 - ▶ 32 ośmiobitowe rejestry
 - ▶ 32KB – pamięć kodu (FLASH)
 - ▶ 2KB – pamięć SRAM
 - ▶ 1KB – pamięć EEPROM
- ▶ 8 bit (16 bitowa szyna danych)
- ▶ 20 Mhz
- ▶ 20 MIPS

20 MIPS. Czy to dużo?

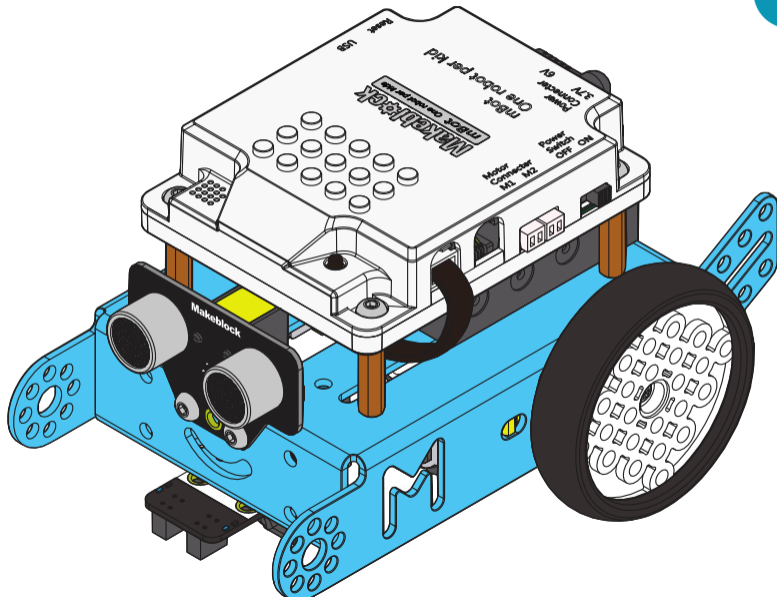
- ▶ Intel i386DX – 4.3 MIPS @ 33MHz; (32bit)
- ▶ Intel i486DX – 8.7 MIPS @ 25MHz; (32bit)
- ▶ Intel i486DX2 – 25 MIPS @ 66MHz; (32bit)
- ▶ Intel Pentium I – 188 MIPS @ 100MHz; (32bit)
- ▶ Intel Core i7 875K – 92100 MIPS @ 2.93GHz; (64bit)
- ▶ Raspberry Pi 2 – 4744 MIPS @ 1GHz (32bit)
- ▶ ARM Cortex A73 71120 MIPS @ 2.8 GHz (64bit)

8 bitowy. Co to znaczy?

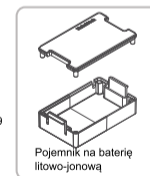
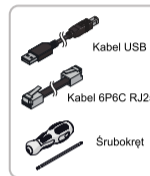
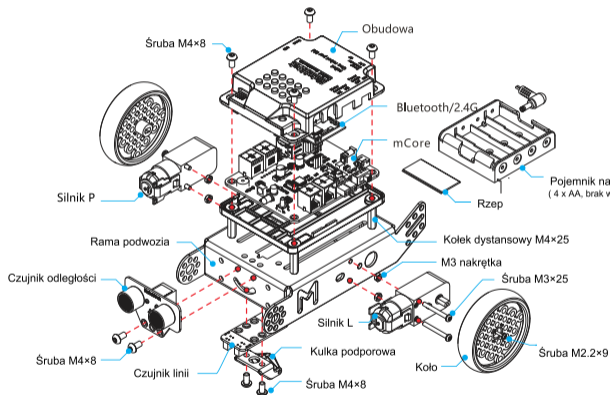
- ▶ Każdy z rejestrów składa się z 8 bitów.
- ▶ Daje to 2^8 różnych wartości, które można w nich zapisać
 - ▶ Liczby bez znaku 0 - 255
 - ▶ liczby ze znakiem -127 - 128

Operacje na liczbach 'rzeczywistych'

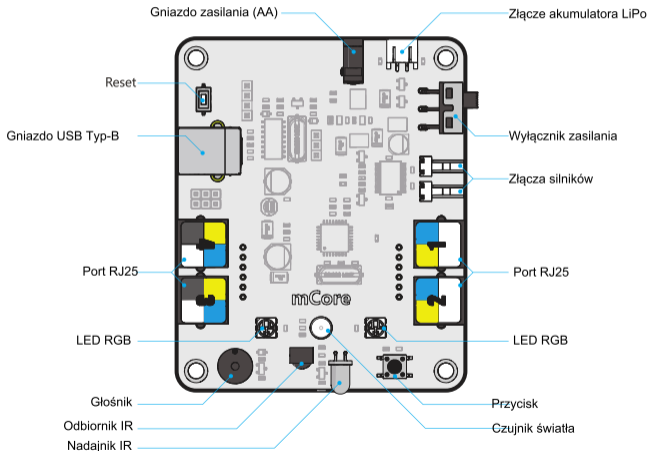
- ▶ Mikrokontroler ATmega328 nie posiada FPU.
- ▶ Możemy wykorzystywać tylko arytmetykę satłoprzecinkową.



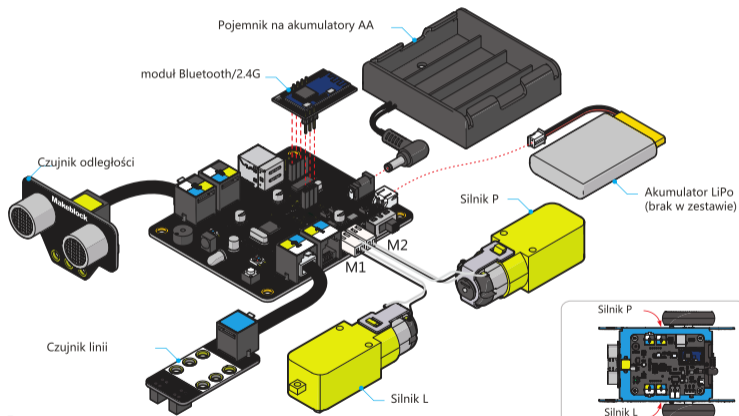
Lista elementów



mCore









Podłączanie



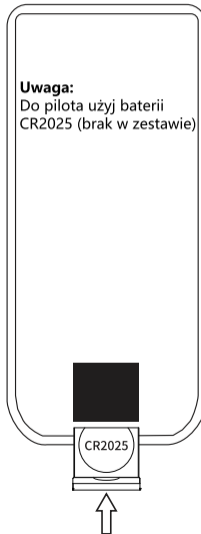
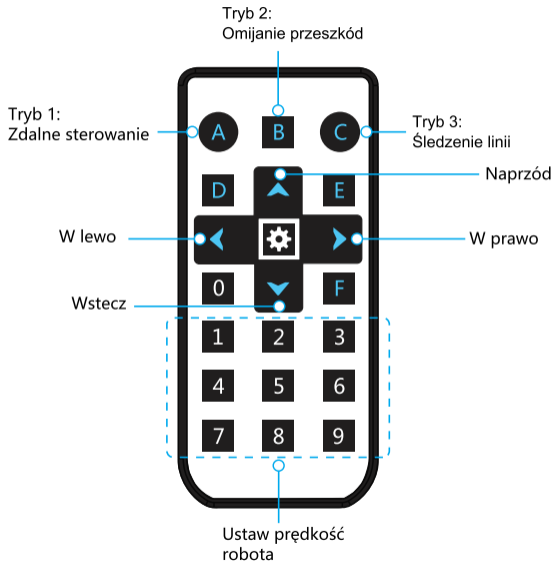
Podłącz silnik L (lewy) do gniazda M1, a silnik P (prawy) do gniazda M2 żeby uzyskać właściwy kierunek obrotów

Kolory portów

Color	Function	Module using this port
	Red means the output voltage is 6-12V, and it is usually connected to the motor driver module of 6-12V voltage	<ul style="list-style-type: none">● Me DC Motor Driver● Me Stepper Motor Driver● Me Encoder Motor Driver
	Single-digital port	<ul style="list-style-type: none">● Me Ultrasonic Sensor● Me RGB LED
	Double-digital port	<ul style="list-style-type: none">● Me 7-Segment Display● Me PIR Motion Sensor● Me Shutter Cable● Me Line Follower● Me IR Receiver
	Serial port of hardware	<ul style="list-style-type: none">● Me Bluetooth Module (Dual Mode)● Me WiFi Module

	Analog signal port	<ul style="list-style-type: none">● Me Light Sensor● Me Potentiometer● Me Joystick● Me 4-Button● Me Sound Sensor
	I ² C port	<ul style="list-style-type: none">● Me 3-Axis Accelerometer and Gyro Sensor● Me Compass

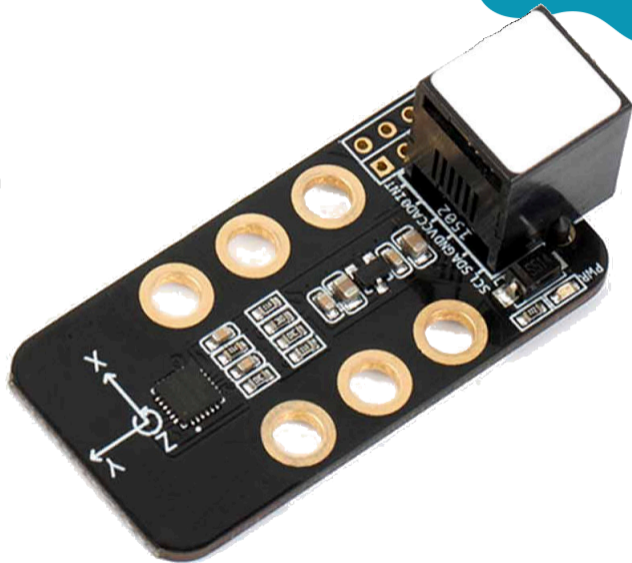
Pilot zdalnego sterowania



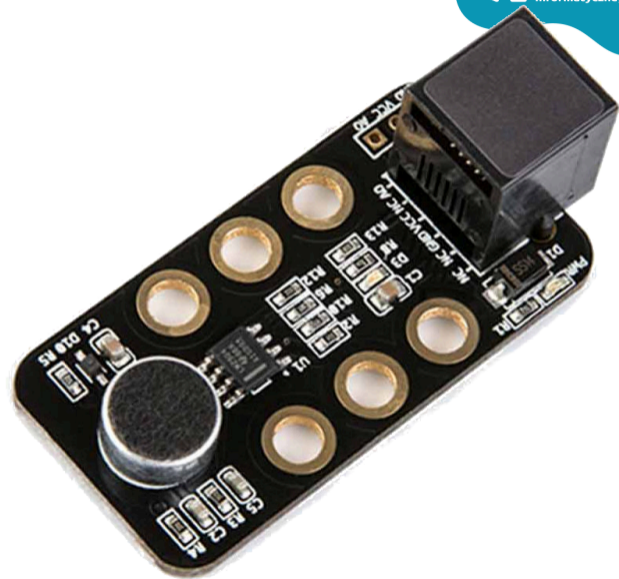
Matryca LED



- ▶ Akcelerometr 3-osiowy z żyroskopem
- ▶ Czujnik dźwięku
- ▶ Czujnik temperatury i wilgotności
- ▶ Czujnik koloru
- ▶ Czujnik dotyku
- ▶ Kompas

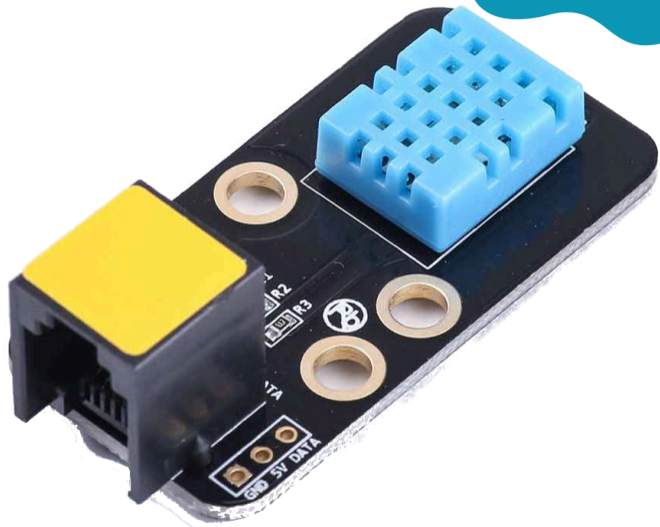


- ▶ Akcelerometr 3-osiowy z żyroskopem
- ▶ **Czujnik dźwięku**
- ▶ Czujnik temperatury i wilgotności
- ▶ Czujnik koloru
- ▶ Czujnik dotyku
- ▶ Kompas



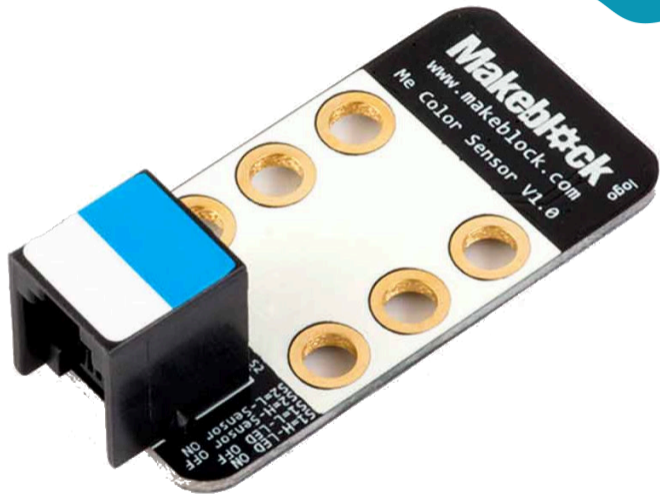
Czujniki dodatkowe

- ▶ Akcelerometr 3-osiowy z żyroskopem
- ▶ Czujnik dźwięku
- ▶ **▶ Czujnik temperatury i wilgotności**
- ▶ Czujnik koloru
- ▶ Czujnik dotyku
- ▶ Kompas



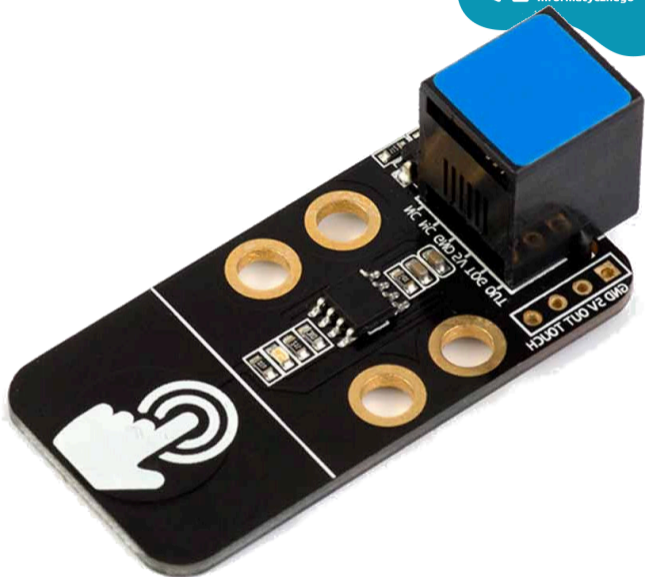
Czujniki dodatkowe

- ▶ Akcelerometr 3-osiowy z żyroskopem
- ▶ Czujnik dźwięku
- ▶ Czujnik temperatury i wilgotności
- ▶ **Czujnik koloru**
- ▶ Czujnik dotyku
- ▶ Kompas



Czujniki dodatkowe

- ▶ Akcelerometr 3-osiowy z żyroskopem
- ▶ Czujnik dźwięku
- ▶ Czujnik temperatury i wilgotności
- ▶ Czujnik koloru
- ▶ Czujnik dotyku
- ▶ Kompas



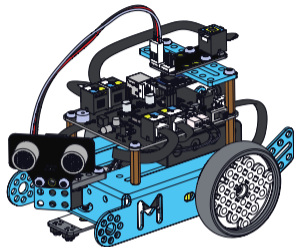
Czujniki dodatkowe

- ▶ Akcelerometr 3-osiowy z żyroskopem
- ▶ Czujnik dźwięku
- ▶ Czujnik temperatury i wilgotności
- ▶ Czujnik koloru
- ▶ Czujnik dotyku
- ▶ Kompas

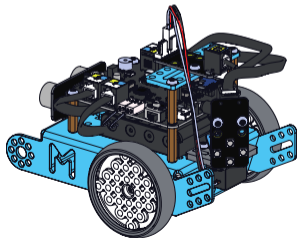


Elementy dodatkowe

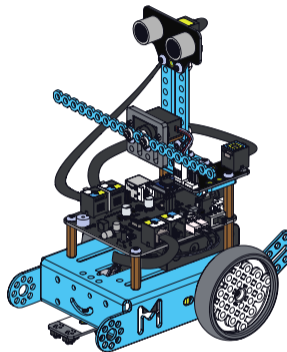
Zestaw z serwomotorem



Head-Shaking Cat



Light-Emitting Cat



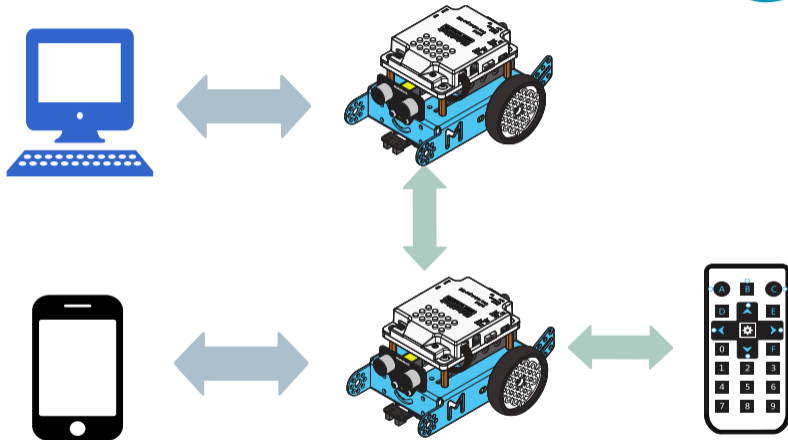
Dancing Cat

Section 3

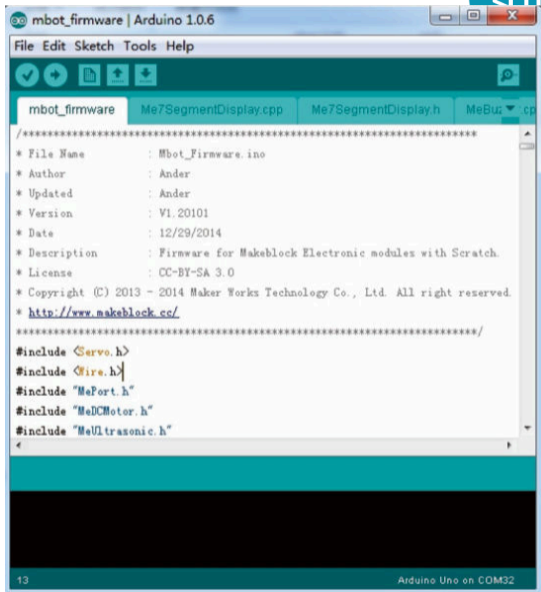
Programowanie zestawu

Komunikacja zestawem

- ▶ USB
- ▶ Bluetooth
- ▶ IR



► Arduino Studio



```
mbot_firmware | Arduino 1.0.6
File Edit Sketch Tools Help
mbot_firmware Me7SegmentDisplay.cpp Me7SegmentDisplay.h MeBus.cpp
/*****
 * File Name      : Mbot_Firmware.ino
 * Author         : Ander
 * Updated        : Ander
 * Version        : V1.20101
 * Date           : 12/29/2014
 * Description    : Firmware for Makeblock Electronic modules with Scratch.
 * License        : CC-BY-SA 3.0
 * Copyright (C) 2013 - 2014 Maker Works Technology Co., Ltd. All right reserved.
 * http://www.makeblock.cc/
 *****/
#include <Servo.h>
#include <Wire.h>
#include "MePort.h"
#include "MeDCMotor.h"
#include "MeUltrasonic.h"
13 Arduino Uno on COM32
```

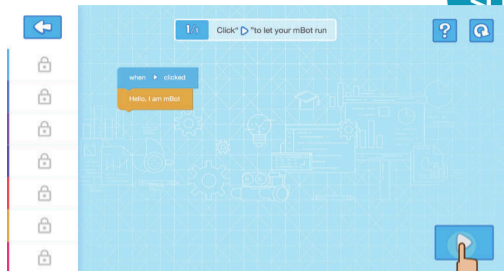
► mBlock

The screenshot displays the mBlock software interface. On the left, a hexagonal sprite is shown with a red outline. Below it, the 'Sprites' panel is active, showing a 'Panda' sprite with its properties: X: 0, Y: 0, Size: 100, Direction: 90. The main workspace contains a sequence of blocks: 'move 10 steps', 'turn 15 degrees', 'turn 15 degrees', 'go to random position', 'go to x: 0 y: 0', 'glide 1 secs to random position', 'glide 1 secs to x: 0 y: 0', 'point in direction 90', 'point towards mouse-pointer', 'change x by 10', 'set x to 0', 'change y by 10', and 'set y to 0'. A 'when clicked' event block is connected to a 'hide' block, followed by 'go to x: 0 y: 0', 'point in direction 90', 'erase all', 'set pen color to red', 'set pen size to 3', 'pen down', a 'repeat 6' loop containing 'move 100 steps' and 'turn 60 degrees', and finally 'pen up'. On the right, the Python code editor shows the following code:

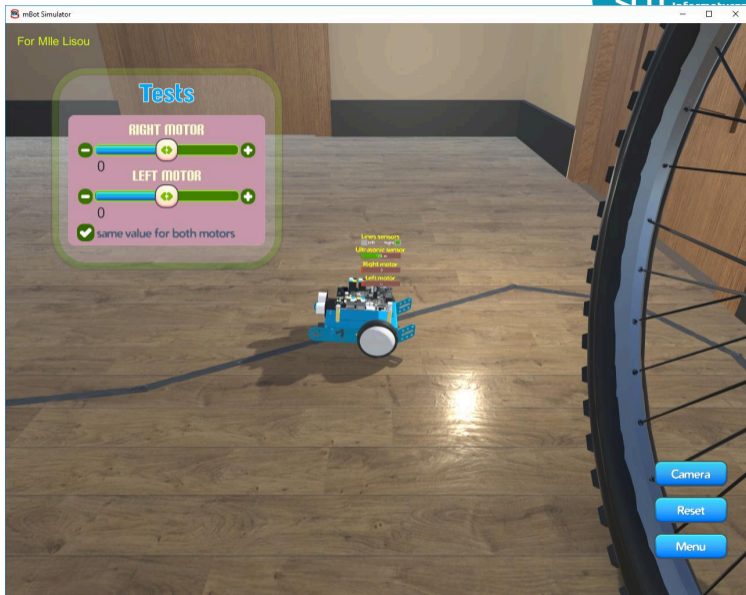
```
Python
1 from mblock import event
2
3 @event.greenflag
4 def on_greenflag():
5     sprite.hide()
6     sprite.x = 0
7     sprite.y = 0
8     sprite.direction = 90
9     sprite.clear()
10    sprite.pencolor('#808080')
11    sprite.pensize(3)
12    sprite.pendown()
13    for count in range(6):
14        sprite.forward(100)
15        sprite.right(60)
16
17    sprite.penup()
18
```

Środowiska programowania

- ▶ mBlock, Makeblock (Android)



- <http://www.irai.com/mbs/mBot%20simulator.pdf>





Dziękuję za Uwagę!



Centrum Mistrzostwa Informatycznego

www.cmi.edu.pl



**Fundusze
Europejskie**
Polska Cyfrowa



**Rzeczpospolita
Polska**



CENTRUM
PROJEKTÓW
POLSKA
CYFROWA

Unia Europejska
Europejski Fundusz
Rozwoju Regionalnego



Projekt "Centrum Mistrzostwa Informatycznego" współfinansowany jest ze środków Unii Europejskiej z Europejskiego Funduszu Rozwoju Regionalnego w ramach Programu Operacyjnego Polska Cyfrowa na lata 2014 - 2020