SYNTHIAM

synthiam.com

The Robot Program Episode 006: Introducing ARC

This lesson introduces the EZ-Builder Robot Software by exploring options and describing features. At the end of this lesson, the reader will be familiar with the overall layout and features of EZ-Builder. Follow along with The Robot Program Episode 006: Introducing EZ-Builder.

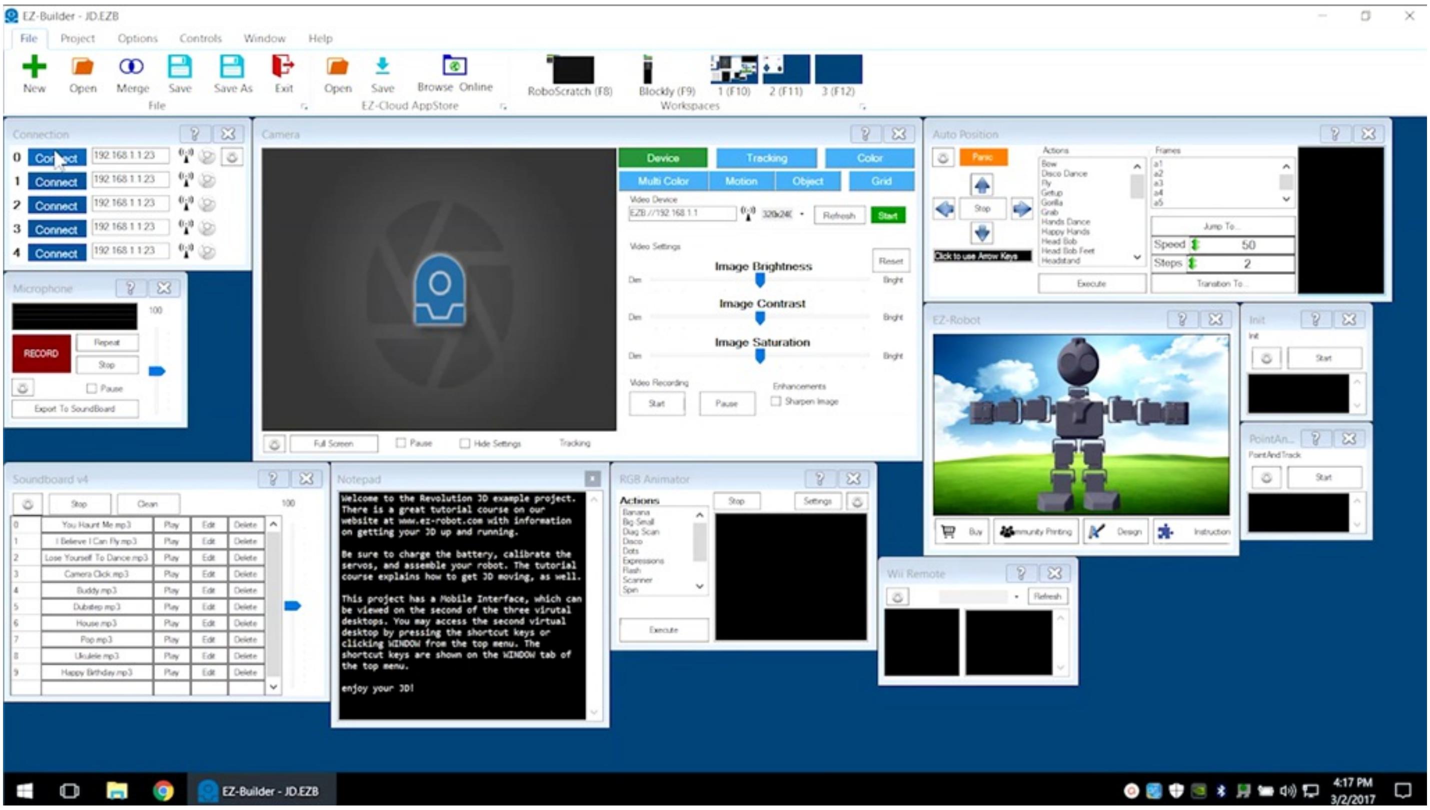
View the video episode here: https://www.ez-robot.com/Tutorials/Lesson/20

Last Updated: 6/1/2018

Professor E's Overview

This lesson introduces and demonstrates the EZ-Builder software. Follow along with The Robot Program Episode 006: Introducing EZ-Builder. At the end of this lesson, readers will be familiar with the layout, workspaces, and available controls of the EZ-Builder software.

The RoboScratch and Blockly workspaces will also be introduced, as well as EZ-Script and third-party plugins. Blue question marks and window questions marks can be used to find more information about a specific aspect or control.

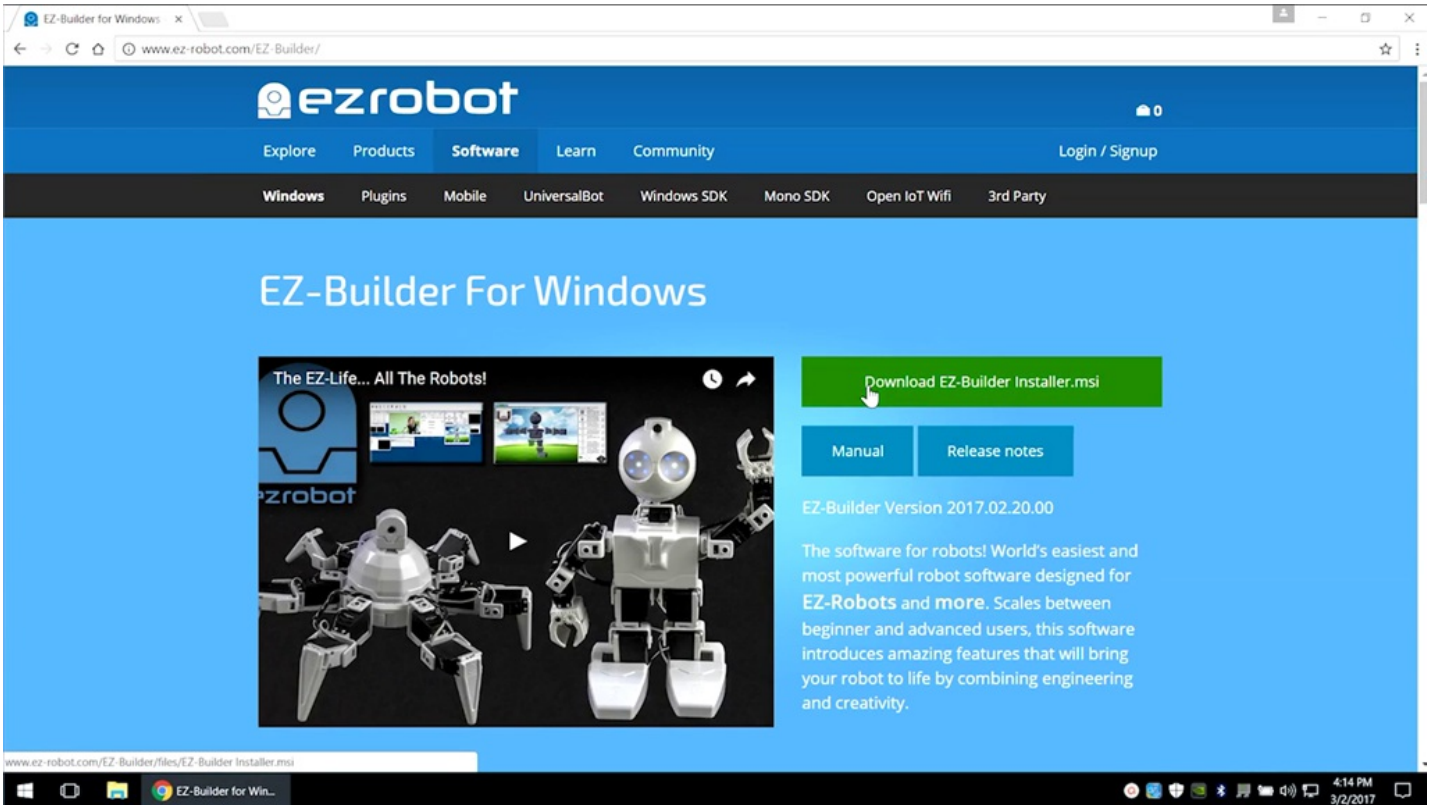


0	You Haunt Me.mp3	Play	Edit	Delete
1	I Believe I Can Fly.mp3	Play	Edit	Delete
2	Lose Yourself To Dance.mp3	Play	Edit	Delete
3	Camera Click.mp3	Play	Edit	Delete
4	Buddy.mp3	Play	Edit	Delete
5	Dubstep.mp3	Play	Edit	Delete
6	House.mp3	Play	Edit	Delete
7	Pop.mp3	Play	Edit	Delete
8	Ukulele.mp3	Play	Edit	Delete
9	Happy Birthday.mp3	Play	Edit	Delete

```
Welcome to the Revolution JD example project.
There is a great tutorial course on our
website at www.ez-robot.com with information
on getting your JD up and running.

Be sure to charge the battery, calibrate the
servos, and assemble your robot. The tutorial
course explains how to get JD moving, as well.

This project has a Mobile Interface, which can
be viewed on the second of the three virutal
desktops. You may access the second virtual
desktop by pressing the shortcut keys or
clicking WINDOW from the top menu. The
shortcut keys are shown on the WINDOW tab of
the top menu.

enjoy your JD!
```

Software Download

Download **EZ-Builder** from www.ez-robot.com. Find more tutorials at **EZ-Robot School**.



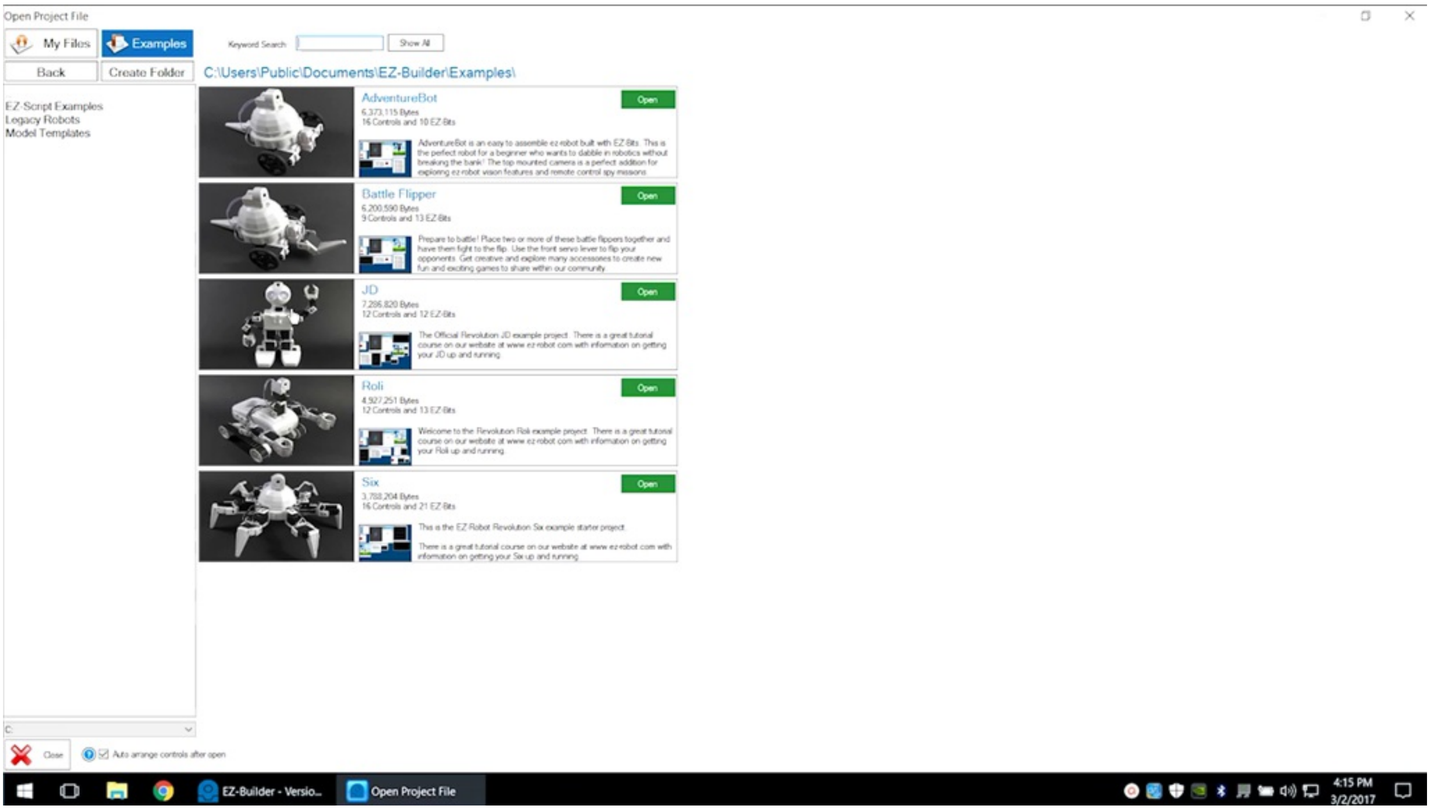
EZ-Builder For Windows

EZ-Builder Version 2017.02.20.00

The software for robots! World's easiest and most powerful robot software designed for **EZ-Robots** and **more**. Scales between beginner and advanced users, this software introduces amazing features that will bring your robot to life by combining engineering and creativity.

Loading an Example Project

Power on the robot. This example uses **Revolution JD**. Load an example project for the robot.



AdventureBot
6,373,115 Bytes
16 Controls and 10 EZ-Bits

AdventureBot is an easy to assemble ez-robot built with EZ-Bits. This is the perfect robot for a beginner who wants to dabble in robotics without breaking the bank! The top mounted camera is a perfect addition for exploring ez-robot vision features and remote control spy missions.

Battle Flipper
6,200,590 Bytes
9 Controls and 13 EZ-Bits

Prepare to battle! Place two or more of these battle flippers together and have them fight to the flip. Use the front servo lever to flip your opponents. Get creative and explore many accessories to create new fun and exciting games to share within our community.

JD
7,286,820 Bytes
12 Controls and 12 EZ-Bits

The Official Revolution JD example project. There is a great tutorial course on our website at www.ez-robot.com with information on getting your JD up and running.

Roli
4,927,251 Bytes
12 Controls and 13 EZ-Bits

Welcome to the Revolution Roli example project. There is a great tutorial course on our website at www.ez-robot.com with information on getting your Roli up and running.

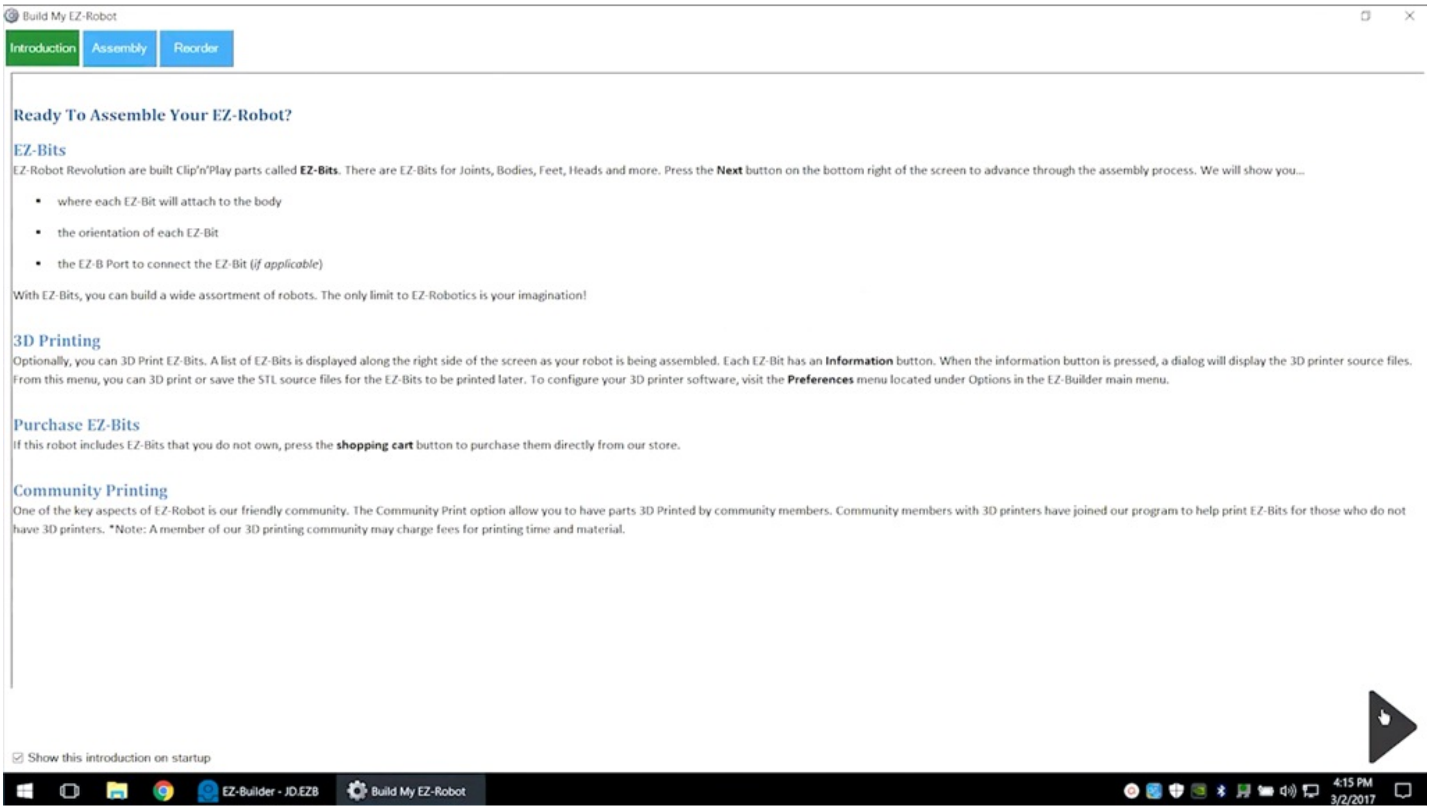
Six
3,788,204 Bytes
16 Controls and 21 EZ-Bits

This is the EZ-Robot Revolution Six example starter project.

There is a great tutorial course on our website at www.ez-robot.com with information on getting your Six up and running.

Building Instructions

View building instructions by selecting **Instructions** from the **Project** tab.



Ready To Assemble Your EZ-Robot?

EZ-Bits

EZ-Robot Revolution are built Clip'n'Play parts called **EZ-Bits**. There are EZ-Bits for Joints, Bodies, Feet, Heads and more. Press the **Next** button on the bottom right of the screen to advance through the assembly process. We will show you...

- where each EZ-Bit will attach to the body
- the orientation of each EZ-Bit
- the EZ-B Port to connect the EZ-Bit (*if applicable*)

With EZ-Bits, you can build a wide assortment of robots. The only limit to EZ-Robotics is your imagination!

3D Printing

Optionally, you can 3D Print EZ-Bits. A list of EZ-Bits is displayed along the right side of the screen as your robot is being assembled. Each EZ-Bit has an **Information** button. When the information button is pressed, a dialog will display the 3D printer source files. From this menu, you can 3D print or save the STL source files for the EZ-Bits to be printed later. To configure your 3D printer software, visit the **Preferences** menu located under Options in the EZ-Builder main menu.

Purchase EZ-Bits

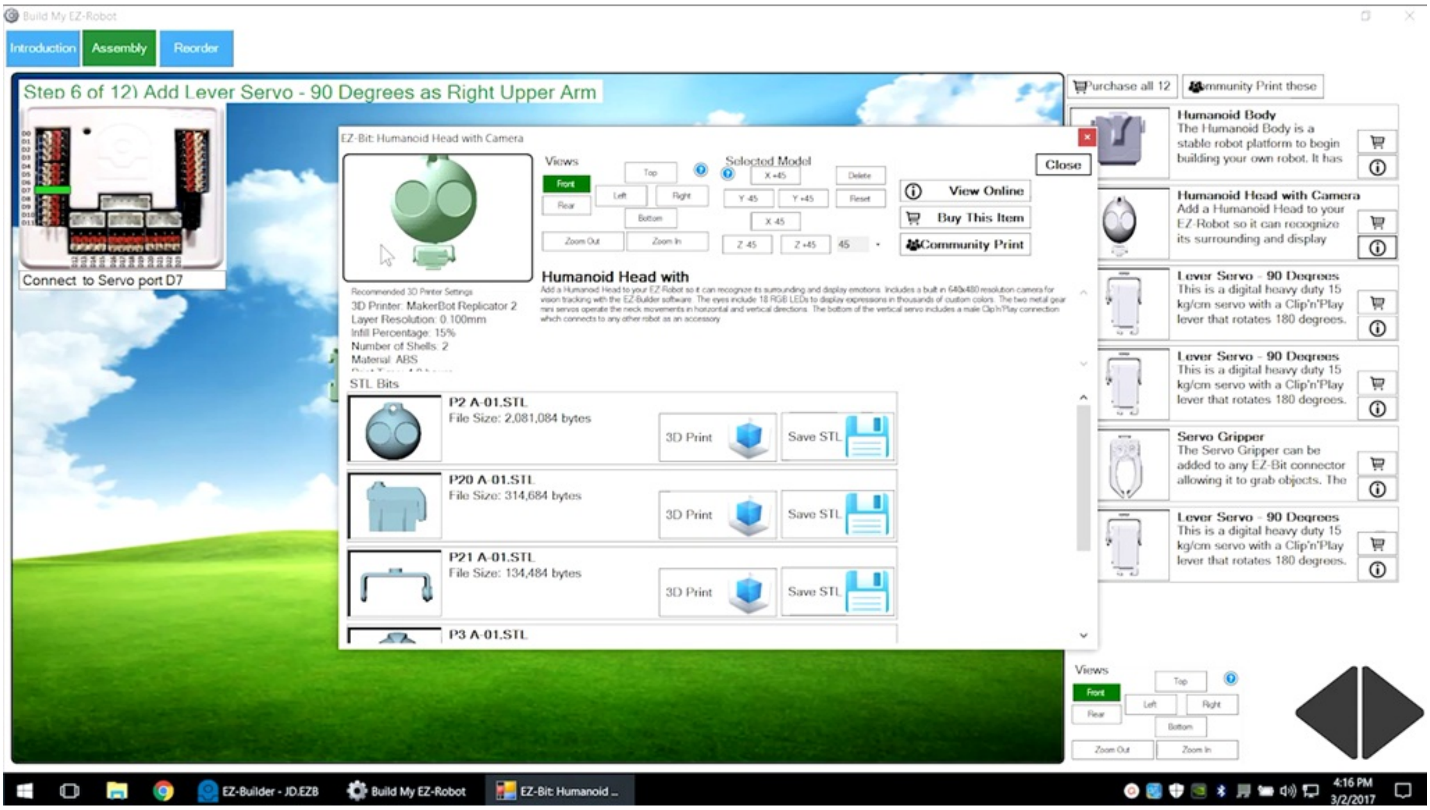
If this robot includes EZ-Bits that you do not own, press the **shopping cart** button to purchase them directly from our store.

Community Printing

One of the key aspects of EZ-Robot is our friendly community. The Community Print option allow you to have parts 3D Printed by community members. Community members with 3D printers have joined our program to help print EZ-Bits for those who do not have 3D printers. *Note: A member of our 3D printing community may charge fees for printing time and material.

☑ Show this introduction on startup

Click on the **i** icon for more information about a part, including 3D printing designs.



Step 6 of 12) Add Lever Servo - 90 Degrees as Right Upper Arm

Connect to Servo port D7

EZ-Bit: Humanoid Head with Camera

Humanoid Head with

Add a Humanoid Head to your EZ-Robot so it can recognize its surrounding and display emotions. Includes a built in 640x480 resolution camera for vision tracking with the EZ-Builder software. The eyes include 18 RGB LEDs to display expressions in thousands of custom colors. The two metal gear mini servos operate the neck movements in horizontal and vertical directions. The bottom of the vertical servo includes a male Clip'n'Play connection which connects to any other robot as an accessory.

Recommended 3D Printer Settings
3D Printer: MakerBot Replicator 2
Layer Resolution: 0.100mm
Infill Percentage: 15%
Number of Shells: 2
Material: ABS

STL Bits

P2 A-01.STL
File Size: 2,081,084 bytes

P20 A-01.STL
File Size: 314,684 bytes

P21 A-01.STL
File Size: 134,484 bytes

P3 A-01.STL

Humanoid Body
The Humanoid Body is a stable robot platform to begin building your own robot. It has

Humanoid Head with Camera
Add a Humanoid Head to your EZ-Robot so it can recognize its surrounding and display

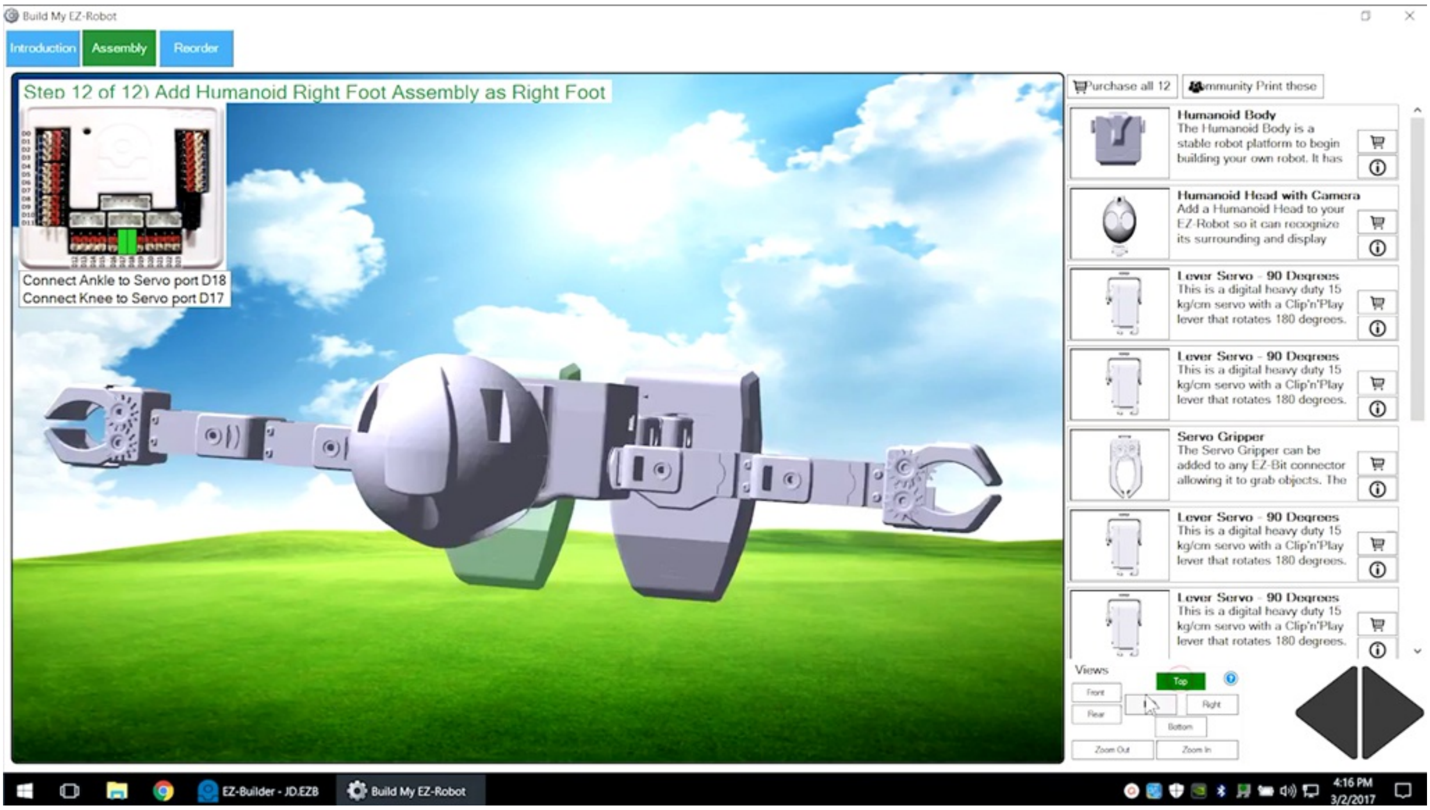
Lever Servo - 90 Degrees
This is a digital heavy duty 15 kg/cm servo with a Clip'n'Play lever that rotates 180 degrees.

Lever Servo - 90 Degrees
This is a digital heavy duty 15 kg/cm servo with a Clip'n'Play lever that rotates 180 degrees.

Servo Gripper
The Servo Gripper can be added to any EZ-Bit connector allowing it to grab objects. The

Lever Servo - 90 Degrees
This is a digital heavy duty 15 kg/cm servo with a Clip'n'Play lever that rotates 180 degrees.

Use the arrow buttons to view the robot from different angles.



Step 12 of 12) Add Humanoid Right Foot Assembly as Right Foot

Connect Ankle to Servo port D18
Connect Knee to Servo port D17

Humanoid Body
The Humanoid Body is a stable robot platform to begin building your own robot. It has

Humanoid Head with Camera
Add a Humanoid Head to your EZ-Robot so it can recognize its surrounding and display

Lever Servo - 90 Degrees
This is a digital heavy duty 15 kg/cm servo with a Clip'n'Play lever that rotates 180 degrees.

Lever Servo - 90 Degrees
This is a digital heavy duty 15 kg/cm servo with a Clip'n'Play lever that rotates 180 degrees.

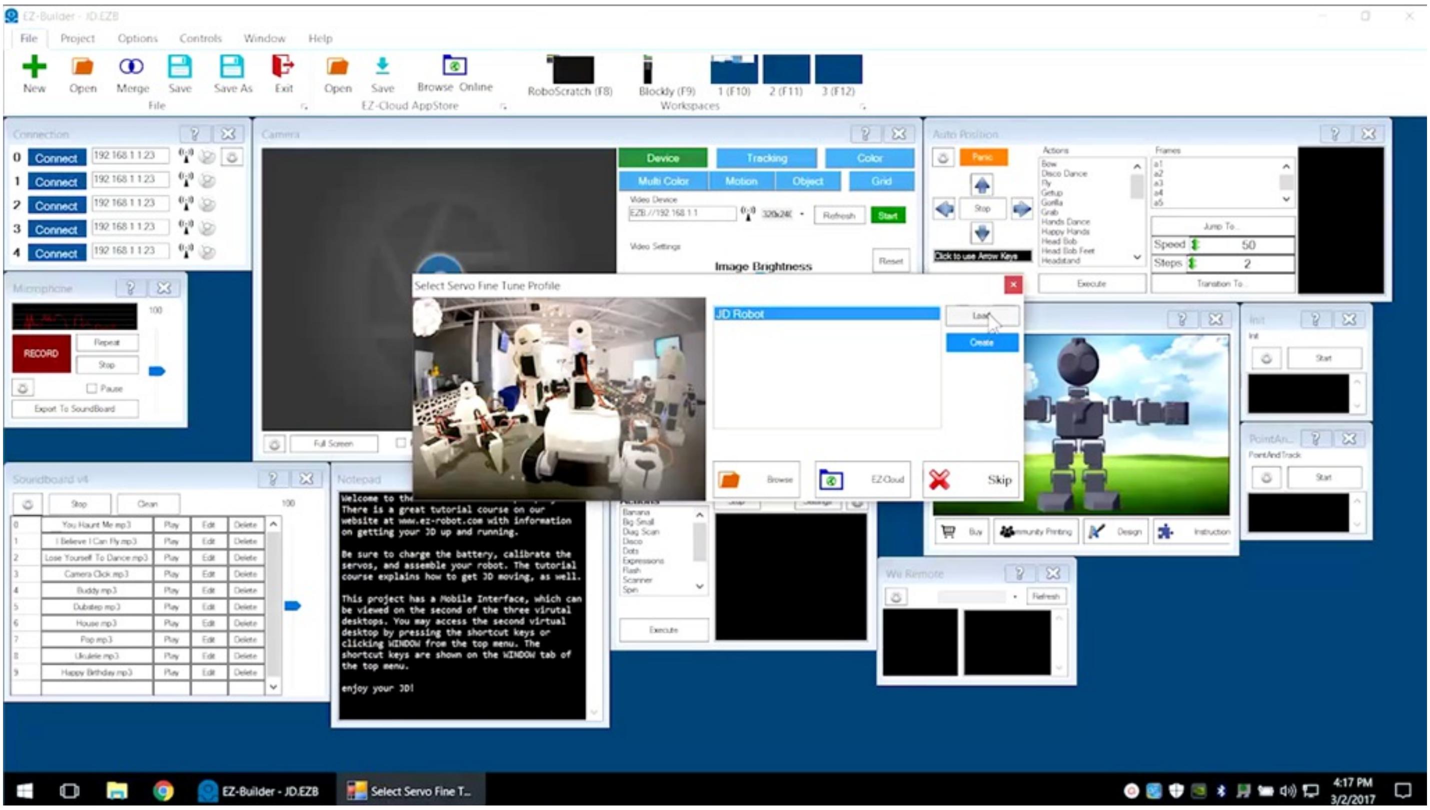
Servo Gripper
The Servo Gripper can be added to any EZ-Bit connector allowing it to grab objects. The

Lever Servo - 90 Degrees
This is a digital heavy duty 15 kg/cm servo with a Clip'n'Play lever that rotates 180 degrees.

Lever Servo - 90 Degrees
This is a digital heavy duty 15 kg/cm servo with a Clip'n'Play lever that rotates 180 degrees.

Servo Profile

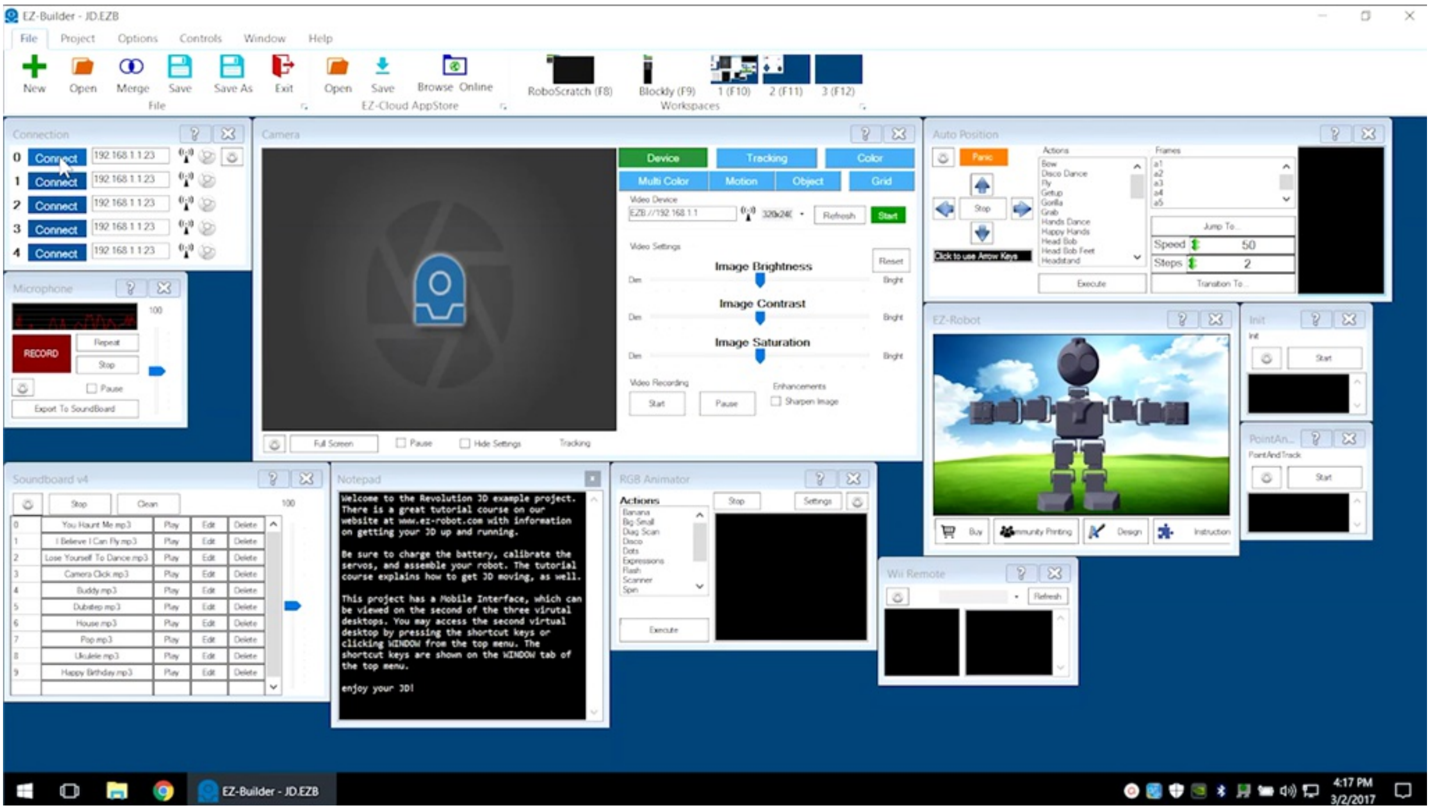
Load a servo profile if needed (humanoid robots only).



```
Welcome to the
There is a great tutorial course on our
website at www.ez-robot.com with information
on getting your JD up and running.

Be sure to charge the battery, calibrate the
servos, and assemble your robot. The tutorial
course explains how to get JD moving, as well.

This project has a Mobile Interface, which can
be viewed on the second of the three virutal
desktops. You may access the second virtual
desktop by pressing the shortcut keys or
clicking WINDOW from the top menu. The
shortcut keys are shown on the WINDOW tab of
the top menu.

enjoy your JD!
```

Connecting

Connect to the robot using the **EZ-B v4** Wi-Fi connection. Click on the blue **Connect** button. Once connected, each robot has an initialization pose.



0	Connect	192.168.1.1:23
1	Connect	192.168.1.1:23
2	Connect	192.168.1.1:23
3	Connect	192.168.1.1:23
4	Connect	192.168.1.1:23

EZB://192.168.1.1 320x240

Auto Position Actions: Bow, Disco Dance, Fly, Getup, Gorilla, Grab, Hands Dance, Happy Hands, Head Bob, Head Bob Feet, Headstand

Frames: a1, a2, a3, a4, a5

Speed 50

Steps 2

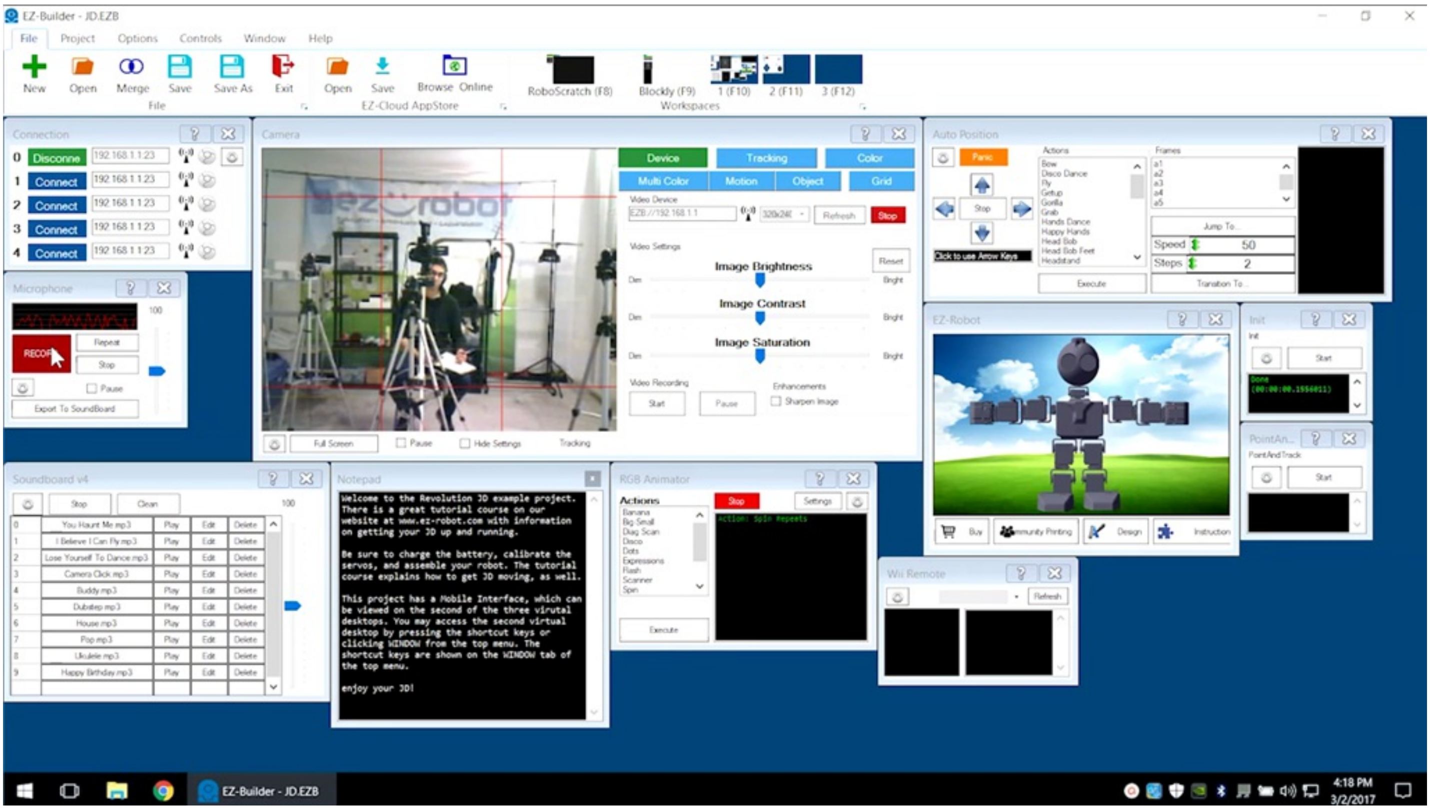
0	You Haunt Me.mp3	Play	Edit	Delete
1	I Believe I Can Fly.mp3	Play	Edit	Delete
2	Lose Yourself To Dance.mp3	Play	Edit	Delete
3	Camera Click.mp3	Play	Edit	Delete
4	Buddy.mp3	Play	Edit	Delete
5	Dubstep.mp3	Play	Edit	Delete
6	House.mp3	Play	Edit	Delete
7	Pop.mp3	Play	Edit	Delete
8	Ukulele.mp3	Play	Edit	Delete
9	Happy Birthday.mp3	Play	Edit	Delete

```
Welcome to the Revolution JD example project.
There is a great tutorial course on our
website at www.ez-robot.com with information
on getting your JD up and running.

Be sure to charge the battery, calibrate the
servos, and assemble your robot. The tutorial
course explains how to get JD moving, as well.

This project has a Mobile Interface, which can
be viewed on the second of the three virutal
desktops. You may access the second virtual
desktop by pressing the shortcut keys or
clicking WINDOW from the top menu. The
shortcut keys are shown on the WINDOW tab of
the top menu.

enjoy your JD!
```

RGB Animator Actions: Banana, Big Small, Diag Scan, Disco, Dots, Expressions, Flash, Scanner, Spin

Microphone Control

There are many different control windows. Use the **Microphone** control to record and playback a sound.



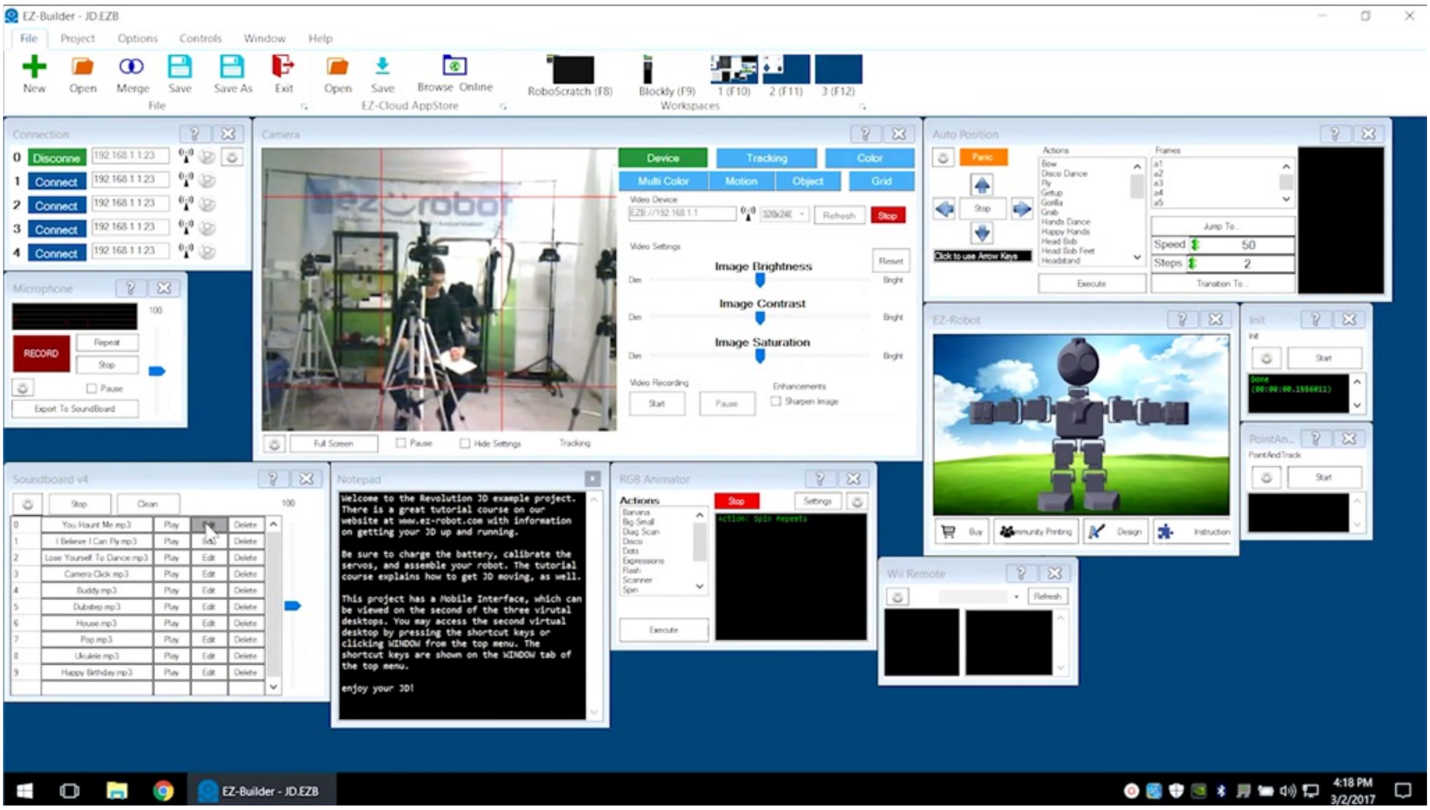
0	Disconne	192.168.1.1:23
1	Connect	192.168.1.1:23
2	Connect	192.168.1.1:23
3	Connect	192.168.1.1:23
4	Connect	192.168.1.1:23

0	You Haunt Me.mp3	Play	Edit	Delete
1	I Believe I Can Fly.mp3	Play	Edit	Delete
2	Lose Yourself To Dance.mp3	Play	Edit	Delete
3	Camera Click.mp3	Play	Edit	Delete
4	Buddy.mp3	Play	Edit	Delete
5	Dubstep.mp3	Play	Edit	Delete
6	House.mp3	Play	Edit	Delete
7	Pop.mp3	Play	Edit	Delete
8	Ukulele.mp3	Play	Edit	Delete
9	Happy Birthday.mp3	Play	Edit	Delete

```
Welcome to the Revolution JD example project.
There is a great tutorial course on our
website at www.ez-robot.com with information
on getting your JD up and running.

Be sure to charge the battery, calibrate the
servos, and assemble your robot. The tutorial
course explains how to get JD moving, as well.

This project has a Mobile Interface, which can
be viewed on the second of the three virutal
desktops. You may access the second virtual
desktop by pressing the shortcut keys or
clicking WINDOW from the top menu. The
shortcut keys are shown on the WINDOW tab of
the top menu.

enjoy your JD!
```

Soundboard Control

Use the **Soundboard** control to play and edit audio files. Code can also be added in sync with the audio waveform.

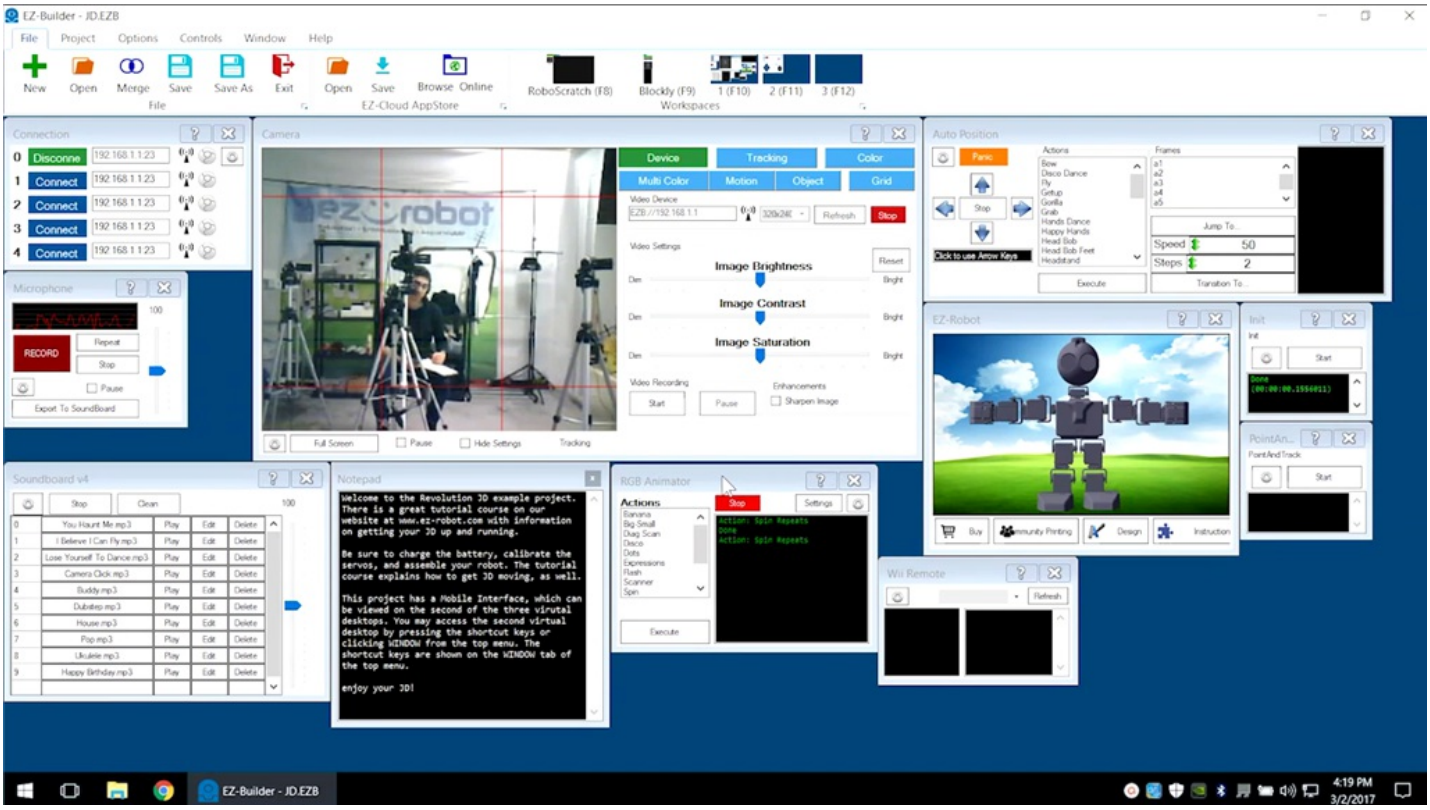


0	You Haunt Me.mp3	Play	Edit	Delete
1	I Believe I Can Fly.mp3	Play	Edit	Delete
2	Lose Yourself To Dance.mp3	Play	Edit	Delete
3	Camera Click.mp3	Play	Edit	Delete
4	Buddy.mp3	Play	Edit	Delete
5	Dubstep.mp3	Play	Edit	Delete
6	House.mp3	Play	Edit	Delete
7	Pop.mp3	Play	Edit	Delete
8	Ukulele.mp3	Play	Edit	Delete
9	Happy Birthday.mp3	Play	Edit	Delete

```
Welcome to the Revolution JD example project.
There is a great tutorial course on our
website at www.ez-robot.com with information
on getting your JD up and running.

Be sure to charge the battery, calibrate the
servos, and assemble your robot. The tutorial
course explains how to get JD moving, as well.

This project has a Mobile Interface, which can
be viewed on the second of the three virutal
desktops. You may access the second virtual
desktop by pressing the shortcut keys or
clicking WINDOW from the top menu. The
shortcut keys are shown on the WINDOW tab of
the top menu.

enjoy your JD!
```

Commonly Used Controls

Other controls include **Camera**, **RGB Animator**, and **PointAndTrack**.

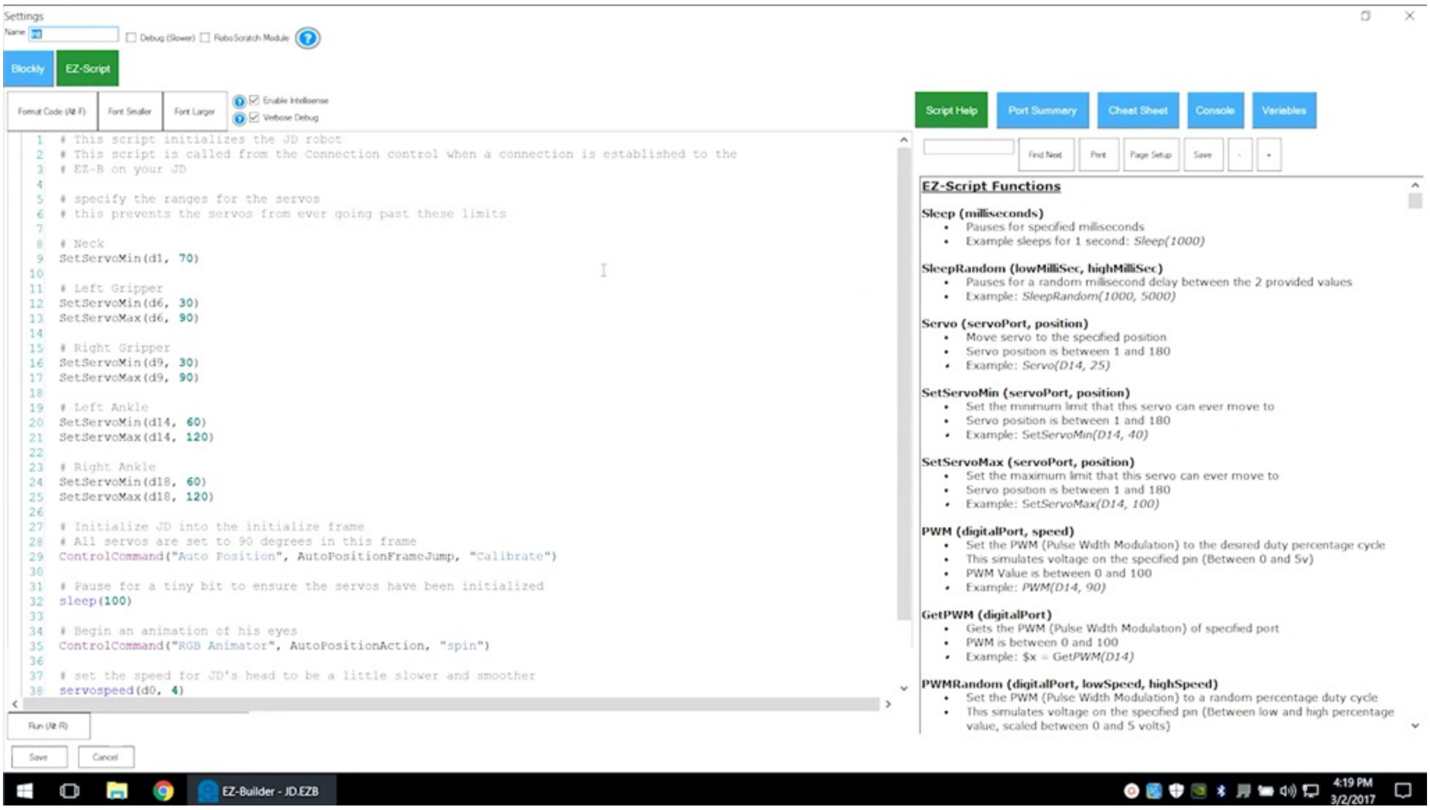


0	You Haunt Me.mp3	Play	Edit	Delete
1	I Believe I Can Fly.mp3	Play	Edit	Delete
2	Lose Yourself To Dance.mp3	Play	Edit	Delete
3	Camera Click.mp3	Play	Edit	Delete
4	Buddy.mp3	Play	Edit	Delete
5	Dubstep.mp3	Play	Edit	Delete
6	House.mp3	Play	Edit	Delete
7	Pop.mp3	Play	Edit	Delete
8	Ukulele.mp3	Play	Edit	Delete
9	Happy Birthday.mp3	Play	Edit	Delete

```
Welcome to the Revolution JD example project.
There is a great tutorial course on our
website at www.ez-robot.com with information
on getting your JD up and running.

Be sure to charge the battery, calibrate the
servos, and assemble your robot. The tutorial
course explains how to get JD moving, as well.

This project has a Mobile Interface, which can
be viewed on the second of the three virutal
desktops. You may access the second virtual
desktop by pressing the shortcut keys or
clicking WINDOW from the top menu. The
shortcut keys are shown on the WINDOW tab of
the top menu.

enjoy your JD!
```

Gear Icon

Click on the **Gear Icon** to see the control configuration code.



```
# This script initializes the JD robot
# This script is called from the Connection control when a connection is established to the
# EZ-B on your JD

# specify the ranges for the servos
# this prevents the servos from ever going past these limits

# Neck
SetServoMin(d1, 70)

# Left Gripper
SetServoMin(d6, 30)
SetServoMax(d6, 90)

# Right Gripper
SetServoMin(d9, 30)
SetServoMax(d9, 90)

# Left Ankle
SetServoMin(d14, 60)
SetServoMax(d14, 120)

# Right Ankle
SetServoMin(d18, 60)
SetServoMax(d18, 120)

# Initialize JD into the initialize frame
# All servos are set to 90 degrees in this frame
ControlCommand("Auto Position", AutoPositionFrameJump, "Calibrate")

# Pause for a tiny bit to ensure the servos have been initialized
sleep(100)

# Begin an animation of his eyes
ControlCommand("RGB Animator", AutoPositionAction, "spin")

# set the speed for JD's head to be a little slower and smoother
servospeed(d0, 4)
```

EZ-Script Functions

Sleep (milliseconds)

- Pauses for specified milliseconds
- Example sleeps for 1 second: *Sleep(1000)*

SleepRandom (lowMilliSec, highMilliSec)

- Pauses for a random millisecond delay between the 2 provided values
- Example: *SleepRandom(1000, 5000)*

Servo (servoPort, position)

- Move servo to the specified position
- Servo position is between 1 and 180
- Example: *Servo(D14, 25)*

SetServoMin (servoPort, position)

- Set the minimum limit that this servo can ever move to
- Servo position is between 1 and 180
- Example: Set*ServoMin(D14, 40)*

SetServoMax (servoPort, position)

- Set the maximum limit that this servo can ever move to
- Servo position is between 1 and 180
- Example: Set*ServoMax(D14, 100)*

PWM (digitalPort, speed)

- Set the PWM (Pulse Width Modulation) to the desired duty percentage cycle
- This simulates voltage on the specified pin (Between 0 and 5v)
- PWM Value is between 0 and 100
- Example: *PWM(D14, 90)*

GetPWM (digitalPort)

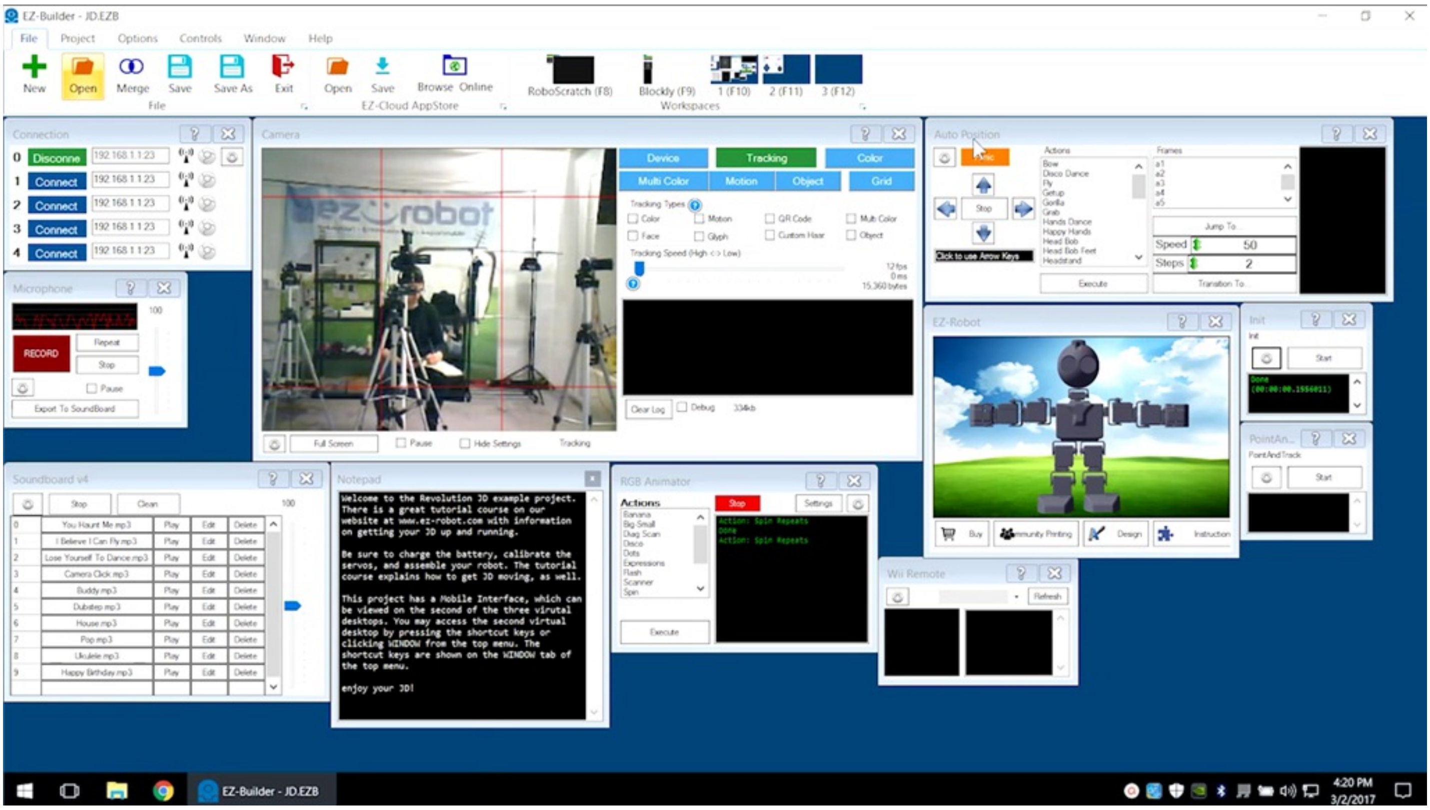
- Gets the PWM (Pulse Width Modulation) of specified port
- PWM is between 0 and 100
- Example: $x = Get*PWM(D14)*

PWMRandom (digitalPort, lowSpeed, highSpeed)

- Set the PWM (Pulse Width Modulation) to a random percentage duty cycle
- This simulates voltage on the specified pin (Between low and high percentage value, scaled between 0 and 5 volts)

Camera Control

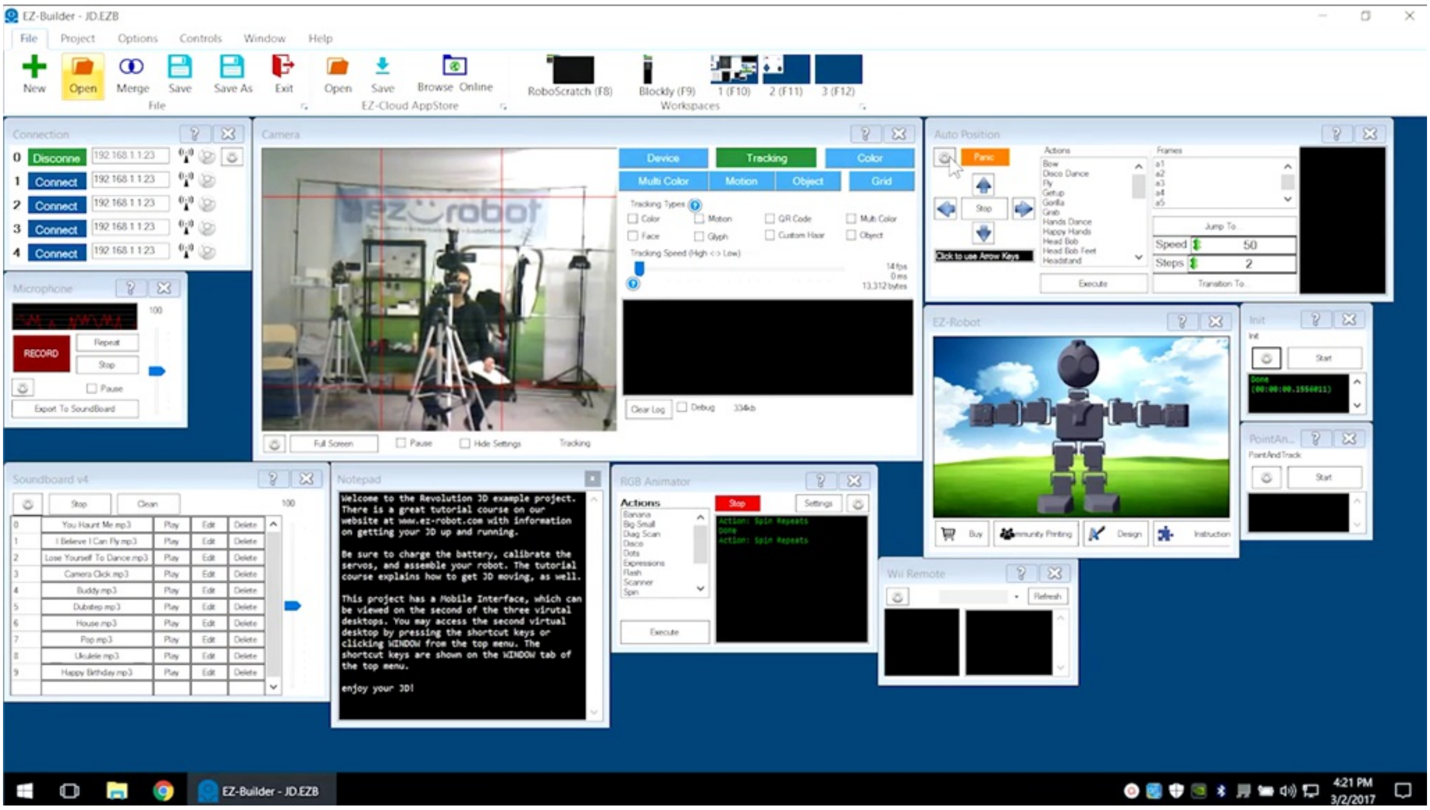
Camera control can be used to change the robot camera settings and to track objects.



```
Welcome to the Revolution JD example project.
There is a great tutorial course on our
website at www.ez-robot.com with information
on getting your JD up and running.

Be sure to charge the battery, calibrate the
servos, and assemble your robot. The tutorial
course explains how to get JD moving, as well.

This project has a Mobile Interface, which can
be viewed on the second of the three virutal
desktops. You may access the second virtual
desktop by pressing the shortcut keys or
clicking WINDOW from the top menu. The
shortcut keys are shown on the WINDOW tab of
the top menu.

enjoy your JD!
```

Auto Position Control

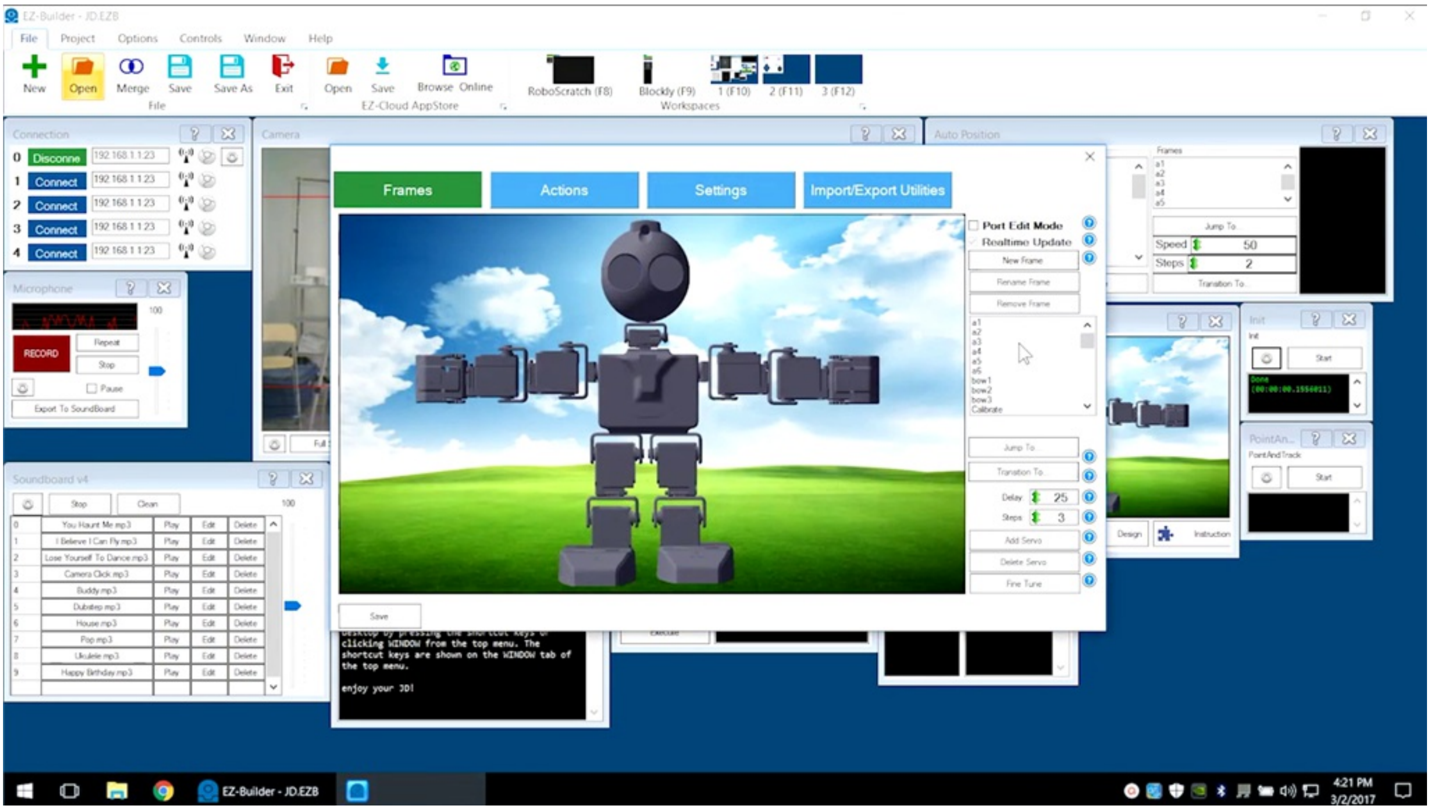
Auto Position is a movement panel. Each robot has its own type of movement panel for controlling motion.



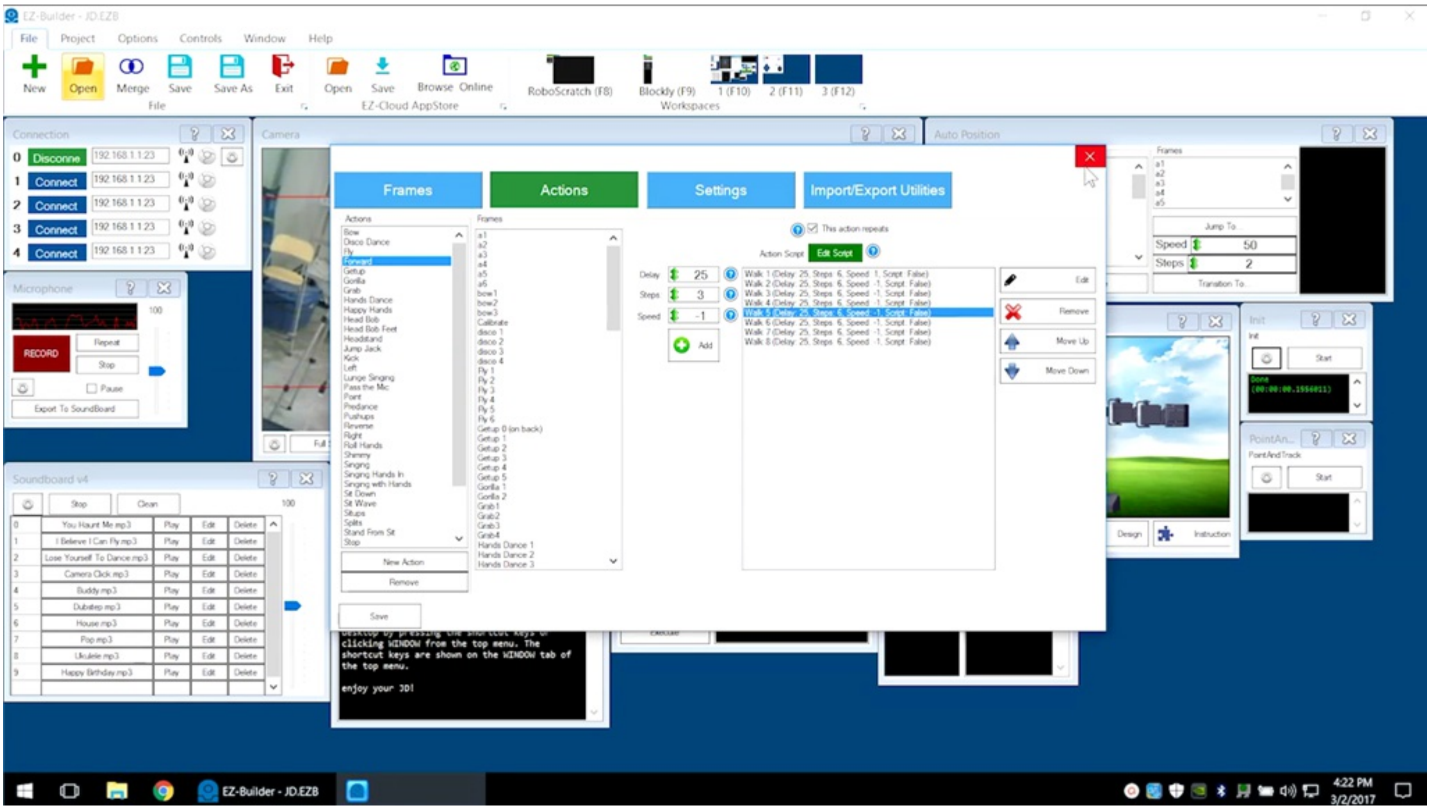
```
Welcome to the Revolution JD example project.
There is a great tutorial course on our
website at www.ez-robot.com with information
on getting your JD up and running.

Be sure to charge the battery, calibrate the
servos, and assemble your robot. The tutorial
course explains how to get JD moving, as well.

This project has a Mobile Interface, which can
be viewed on the second of the three virutal
desktops. You may access the second virtual
desktop by pressing the shortcut keys or
clicking WINDOW from the top menu. The
shortcut keys are shown on the WINDOW tab of
the top menu.

enjoy your JD!
```

Frame Creation

Click on the **Auto Position** gear icon to create frame-by-frame movement control.



```
clicking WINDOW from the top menu. The
shortcut keys are shown on the WINDOW tab of
the top menu.

enjoy your JD!
```



☑ This action repeats

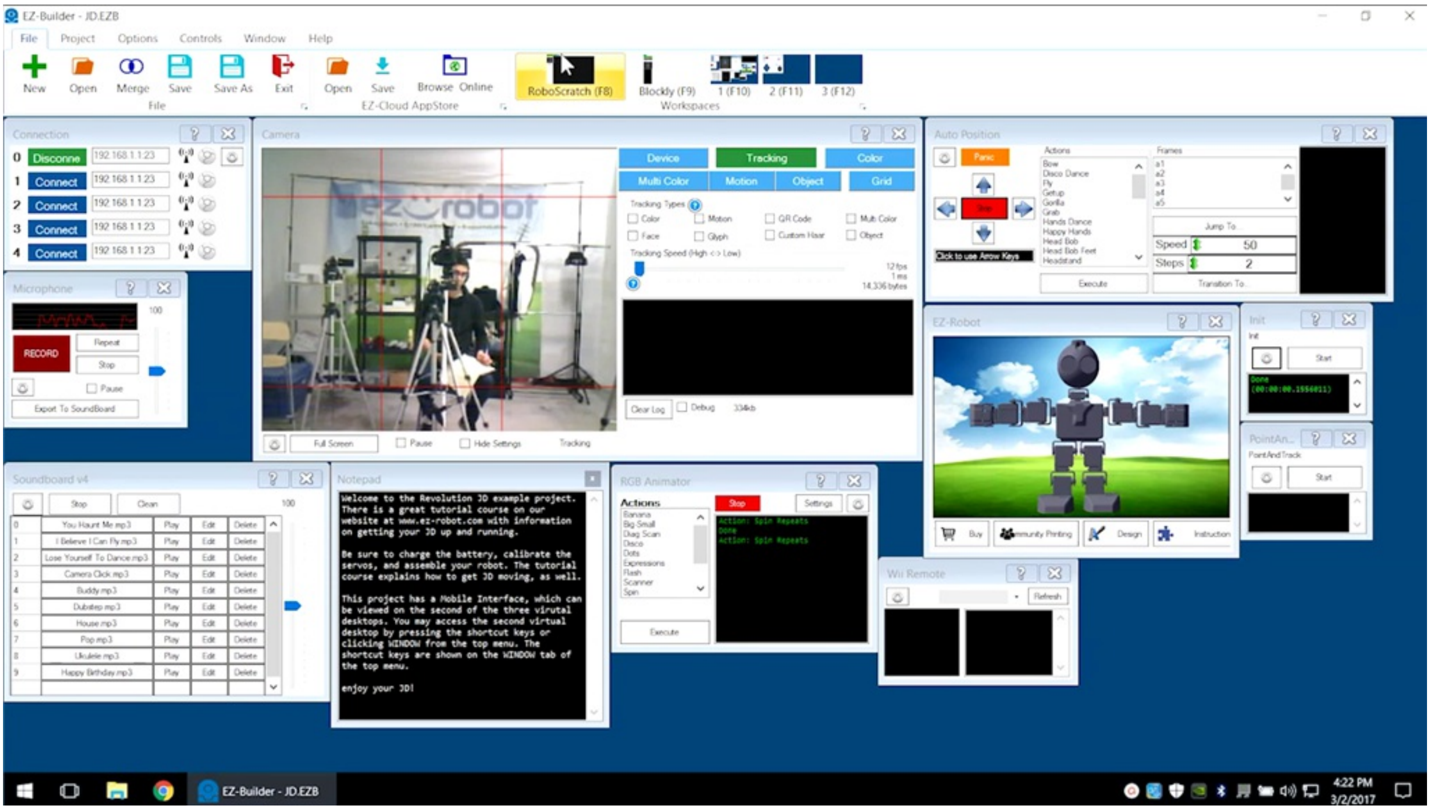
Action Script: Edit Script

Walk 1 (Delay: 25, Steps: 6, Speed: 1, Script: False)
Walk 2 (Delay: 25, Steps: 6, Speed: -1, Script: False)
Walk 3 (Delay: 25, Steps: 6, Speed: -1, Script: False)
Walk 4 (Delay: 25, Steps: 6, Speed: -1, Script: False)
Walk 5 (Delay: 25, Steps: 6, Speed: -1, Script: False)
Walk 6 (Delay: 25, Steps: 6, Speed: -1, Script: False)
Walk 7 (Delay: 25, Steps: 6, Speed: -1, Script: False)
Walk 8 (Delay: 25, Steps: 6, Speed: -1, Script: False)

```
clicking WINDOW from the top menu. The
shortcut keys are shown on the WINDOW tab of
the top menu.

enjoy your JD!
```

Coding Workspaces

Custom controls can also be created through coding. View available coding **Workspaces** using the **File** tab.

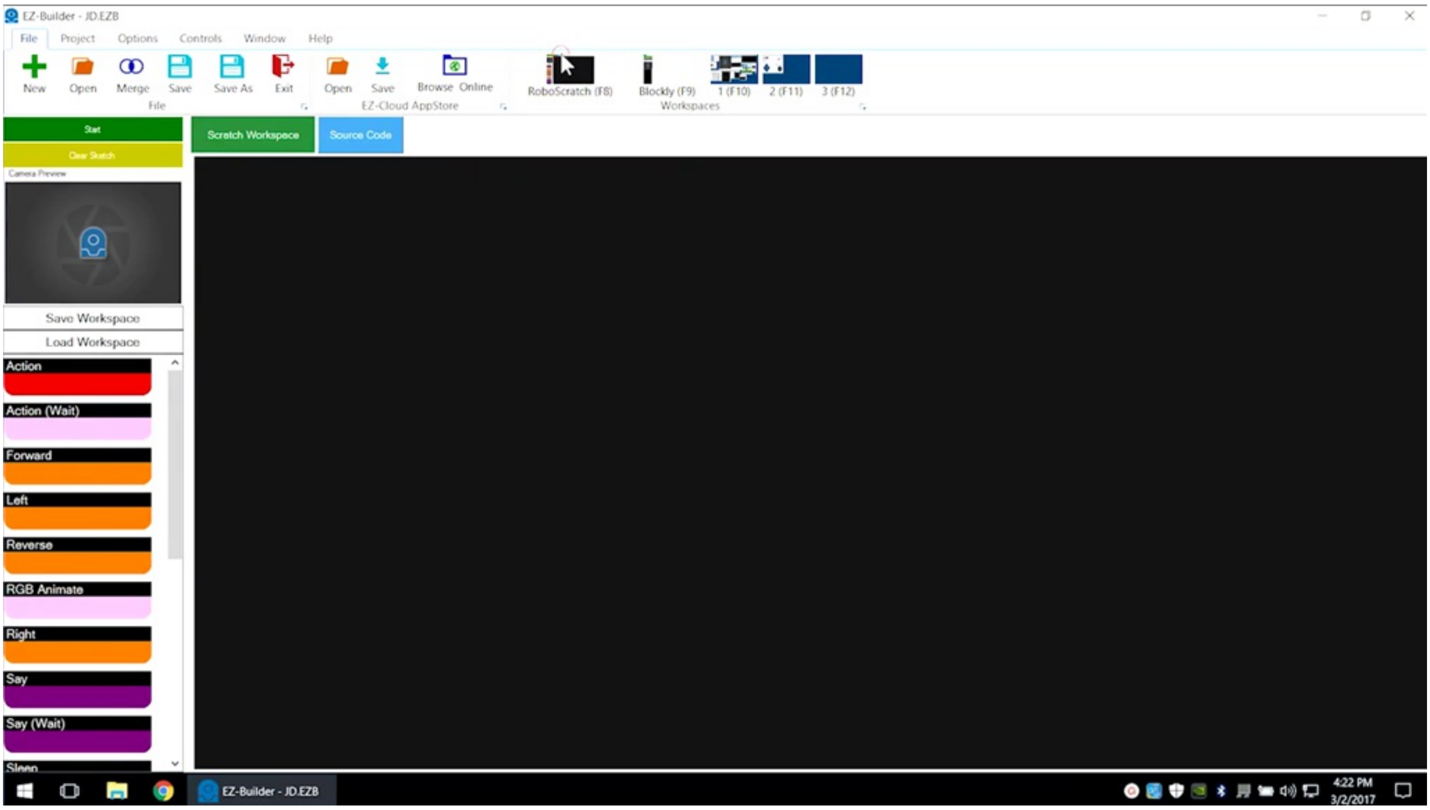


```
Welcome to the Revolution JD example project.
There is a great tutorial course on our
website at www.ez-robot.com with information
on getting your JD up and running.

Be sure to charge the battery, calibrate the
servos, and assemble your robot. The tutorial
course explains how to get JD moving, as well.

This project has a Mobile Interface, which can
be viewed on the second of the three virutal
desktops. You may access the second virtual
desktop by pressing the shortcut keys or
clicking WINDOW from the top menu. The
shortcut keys are shown on the WINDOW tab of
the top menu.

enjoy your JD!
```

0	You Haunt Me.mp3	Play	Edit	Delete
1	I Believe I Can Fly.mp3	Play	Edit	Delete
2	Lose Yourself To Dance.mp3	Play	Edit	Delete
3	Camera Click.mp3	Play	Edit	Delete
4	Buddy.mp3	Play	Edit	Delete
5	Dubstep.mp3	Play	Edit	Delete
6	House.mp3	Play	Edit	Delete
7	Pop.mp3	Play	Edit	Delete
8	Ukulele.mp3	Play	Edit	Delete
9	Happy Birthday.mp3	Play	Edit	Delete

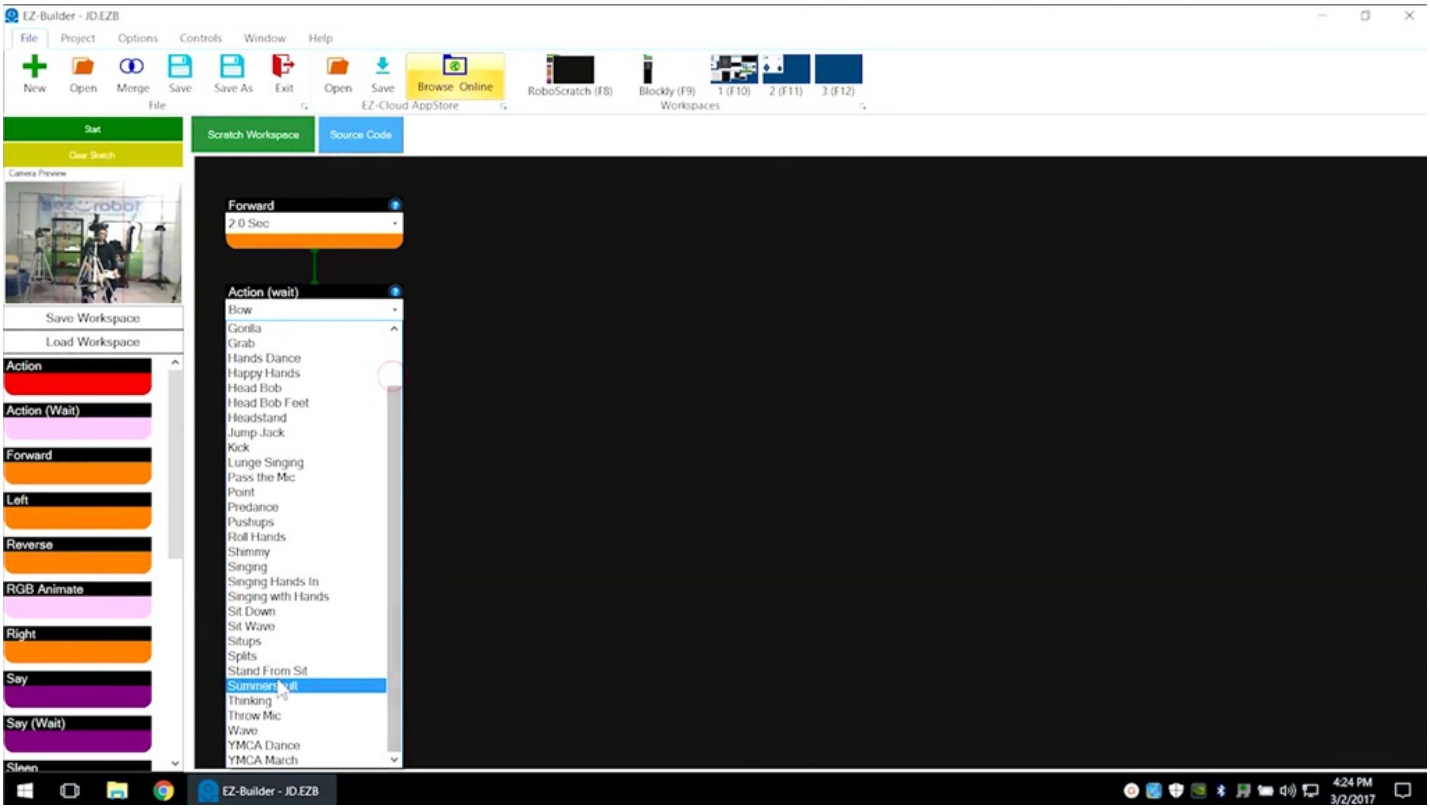
RoboScratch Workspace

Select **RoboScratch** from the **Workspaces** to create a linear program that runs step-by-step from start to finish.



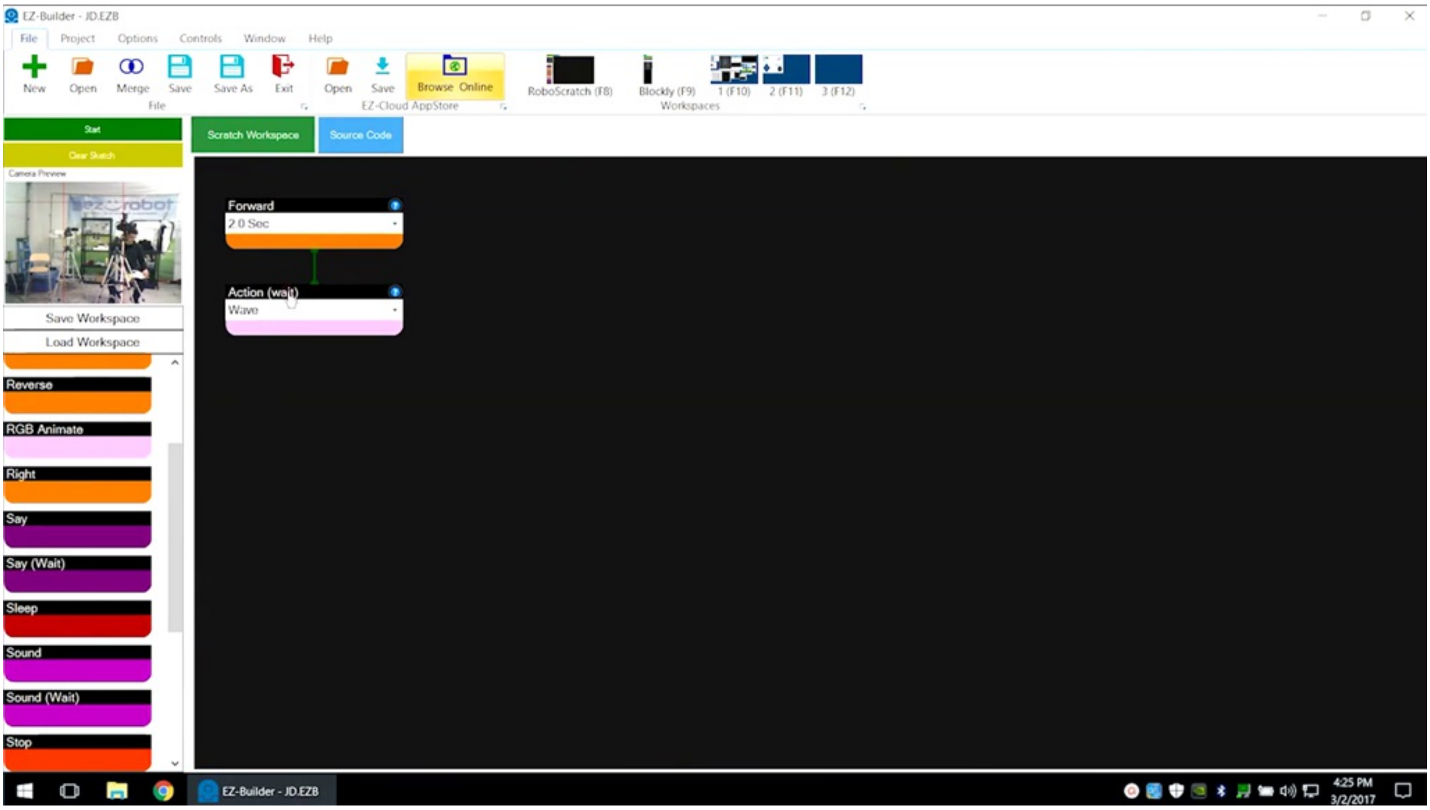
RoboScratch Commands

Click on commands, drag into position, and edit the parameters as desired.



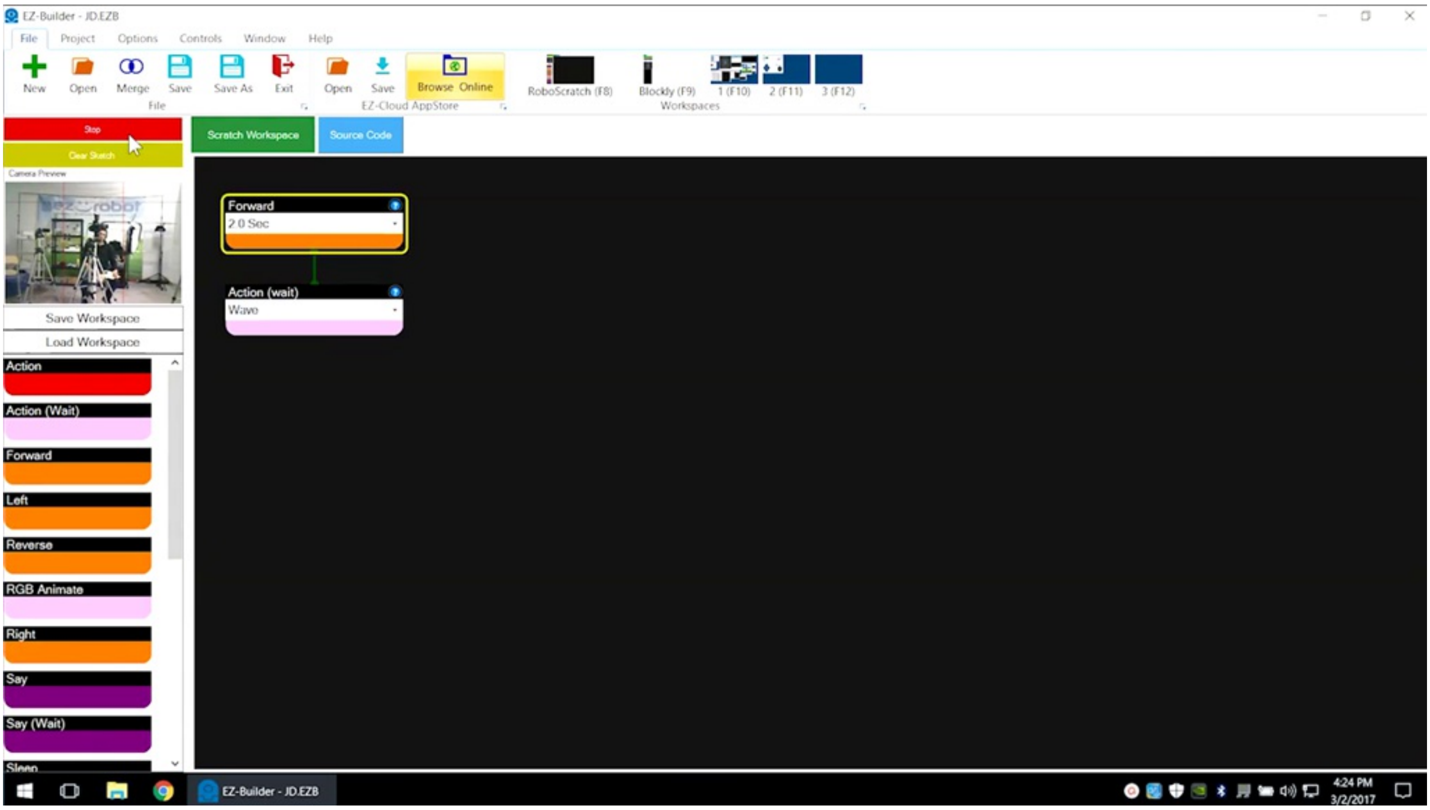
Using Wait

Wait will allow an **Action** to complete before moving to the next command.



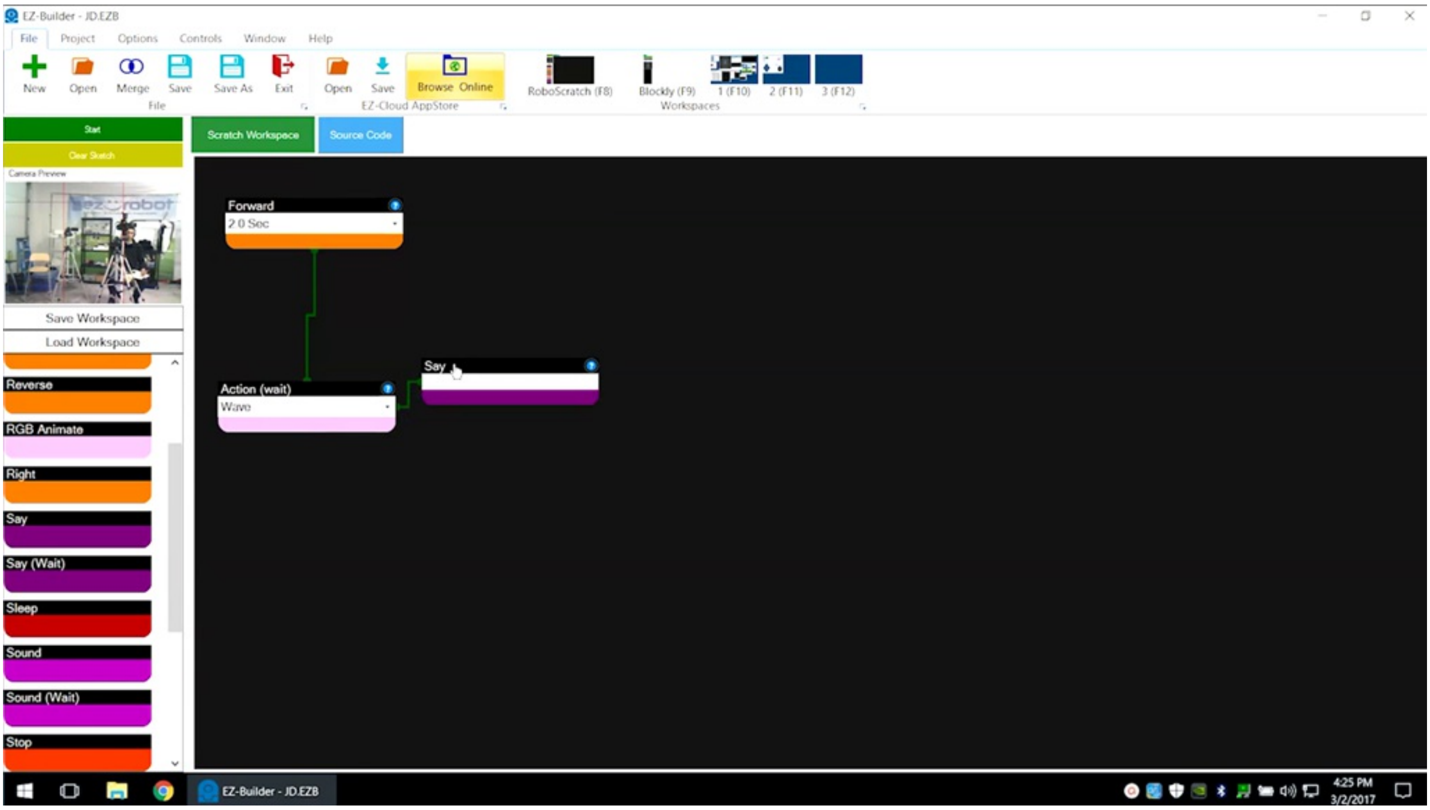
RoboScratch Program Execution

Click on **Start** to run the program. Each command is highlighted in yellow as it executes.



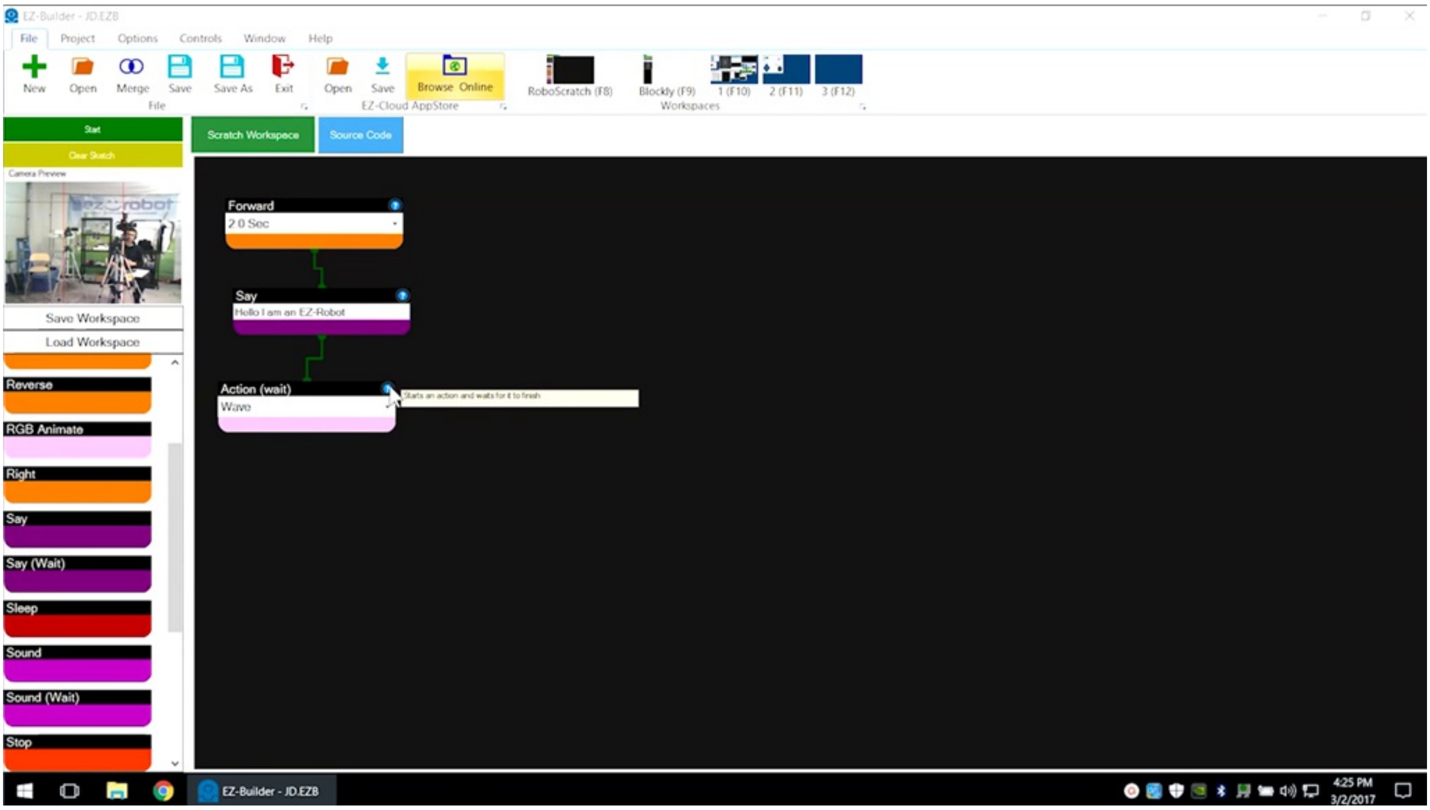
RoboScratch Program Flow

Follow the green line for program flow. Commands can be reordered by dragging into a new position.



Blue Question Marks

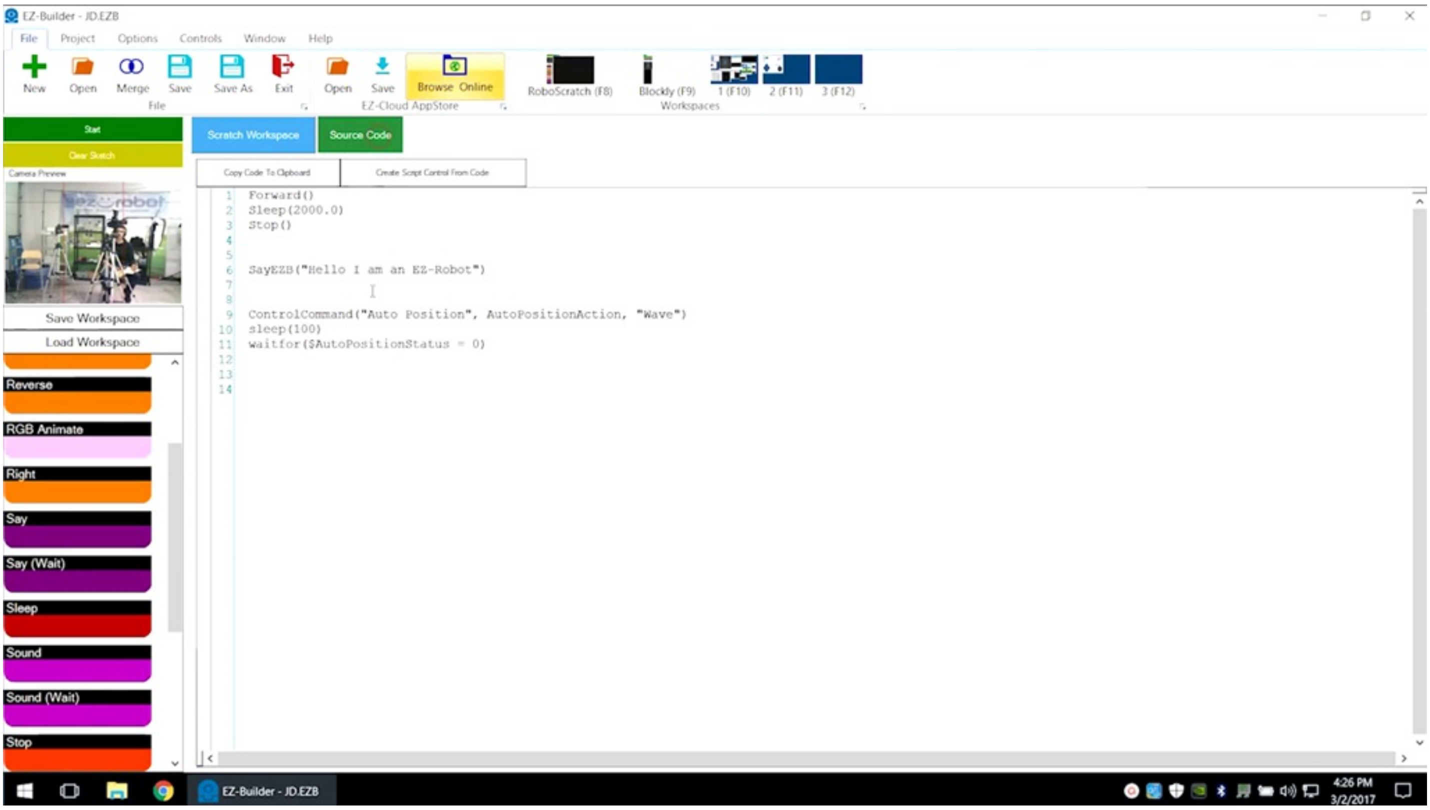
Hover over any blue question mark for more information. Click on window question marks for additional details.



Starts an action and waits for it to finish

RoboScratch Source Code

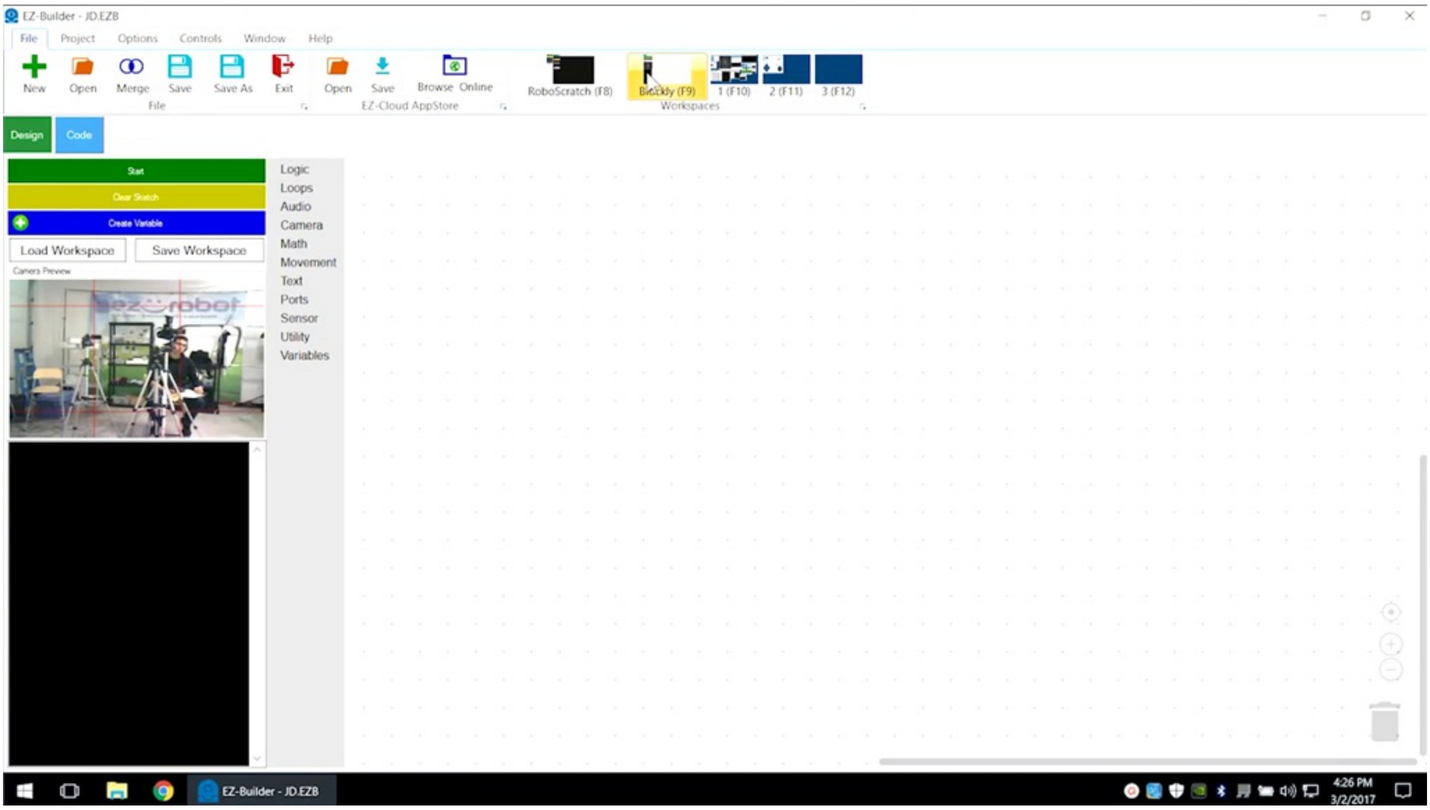
Click on the **Source Code** button to view the generated **EZ-Script** code.



```
Forward()
Sleep(2000.0)
Stop()


SayEZB("Hello I am an EZ-Robot")


ControlCommand("Auto Position", AutoPositionAction, "Wave")
sleep(100)
waitfor($AutoPositionStatus = 0)
```

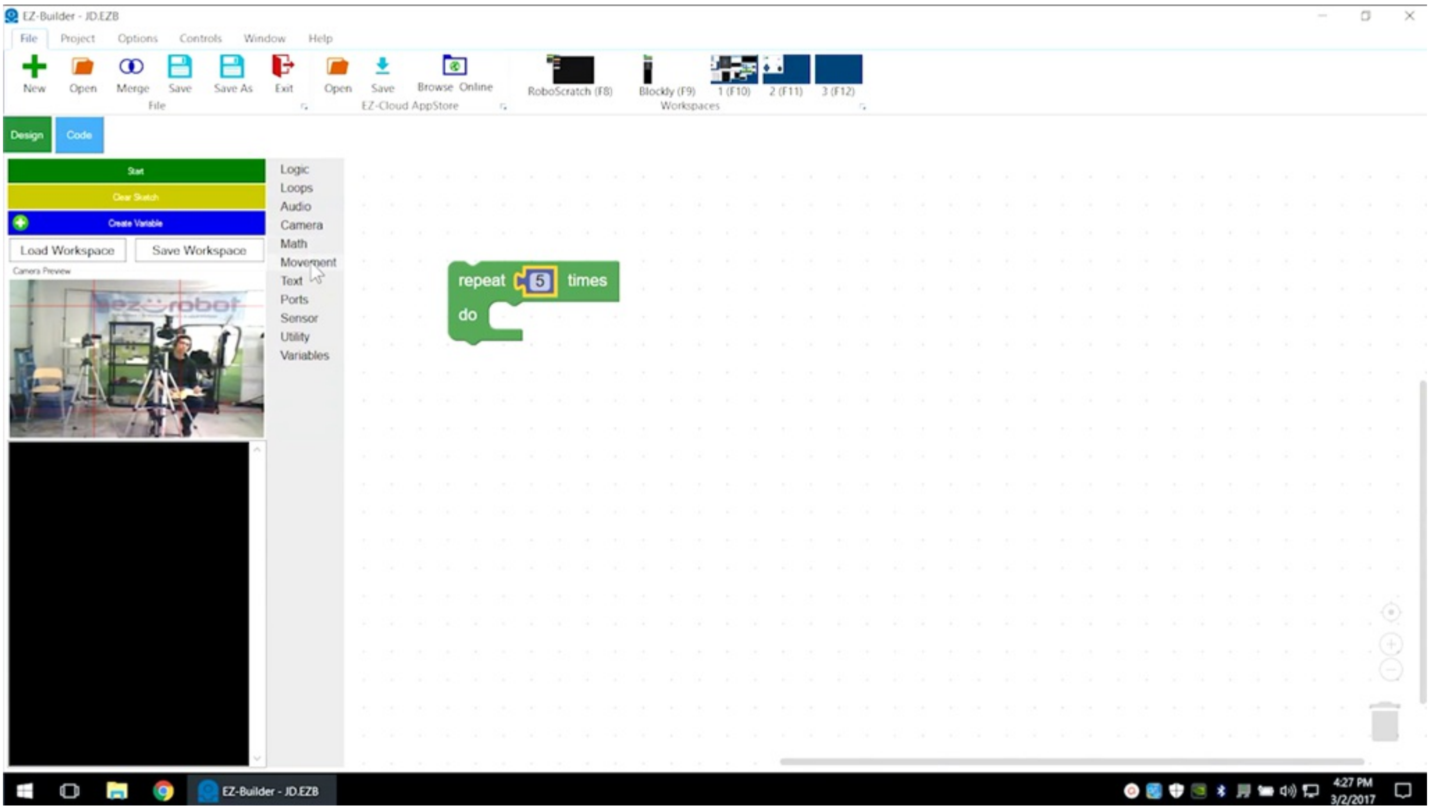
Blockly Workspace

Select **Blockly** from the **Workspaces** to create a more complicated program that uses logic, loops, and branches.



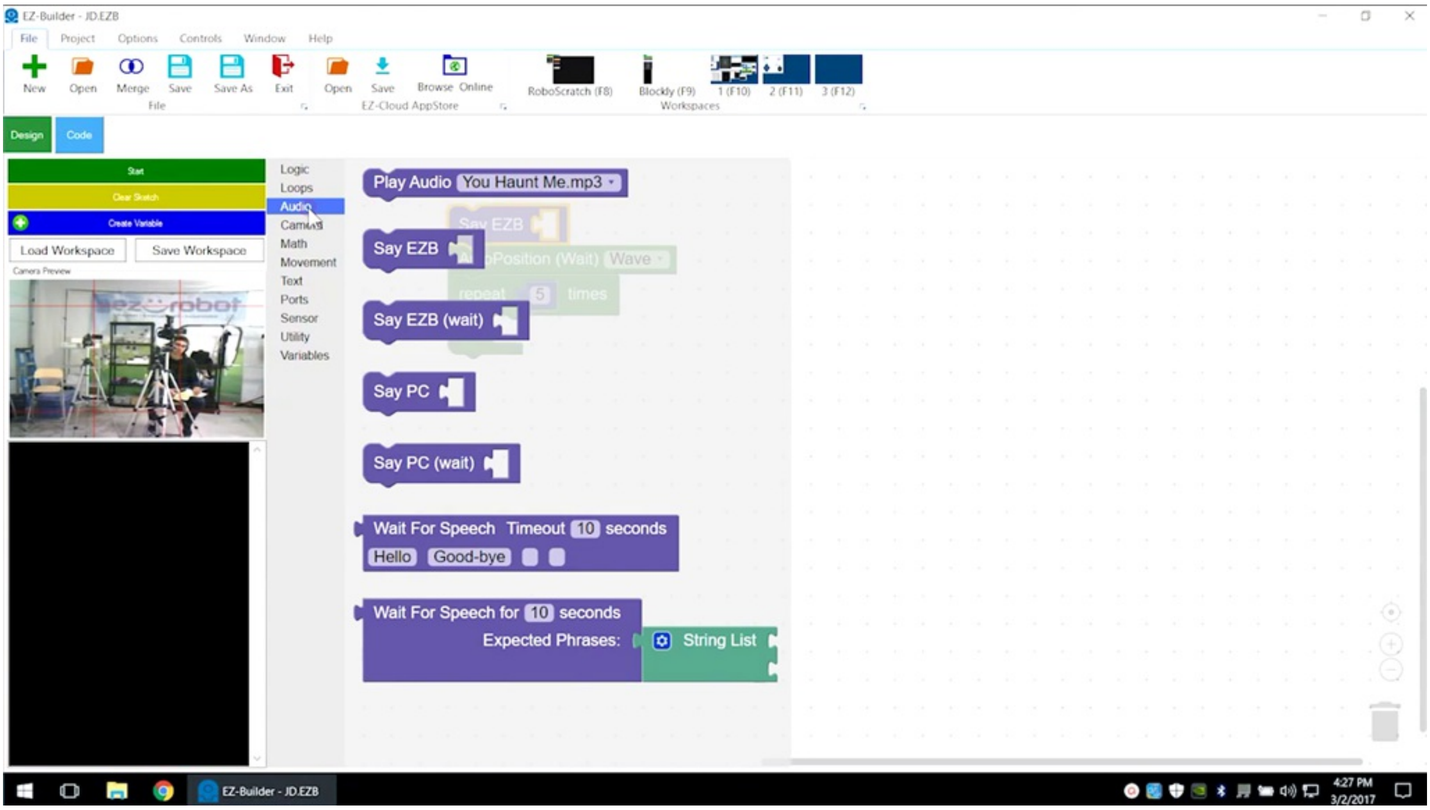
Blockly Commands

Click on commands, drag into position, and edit the parameters as desired.



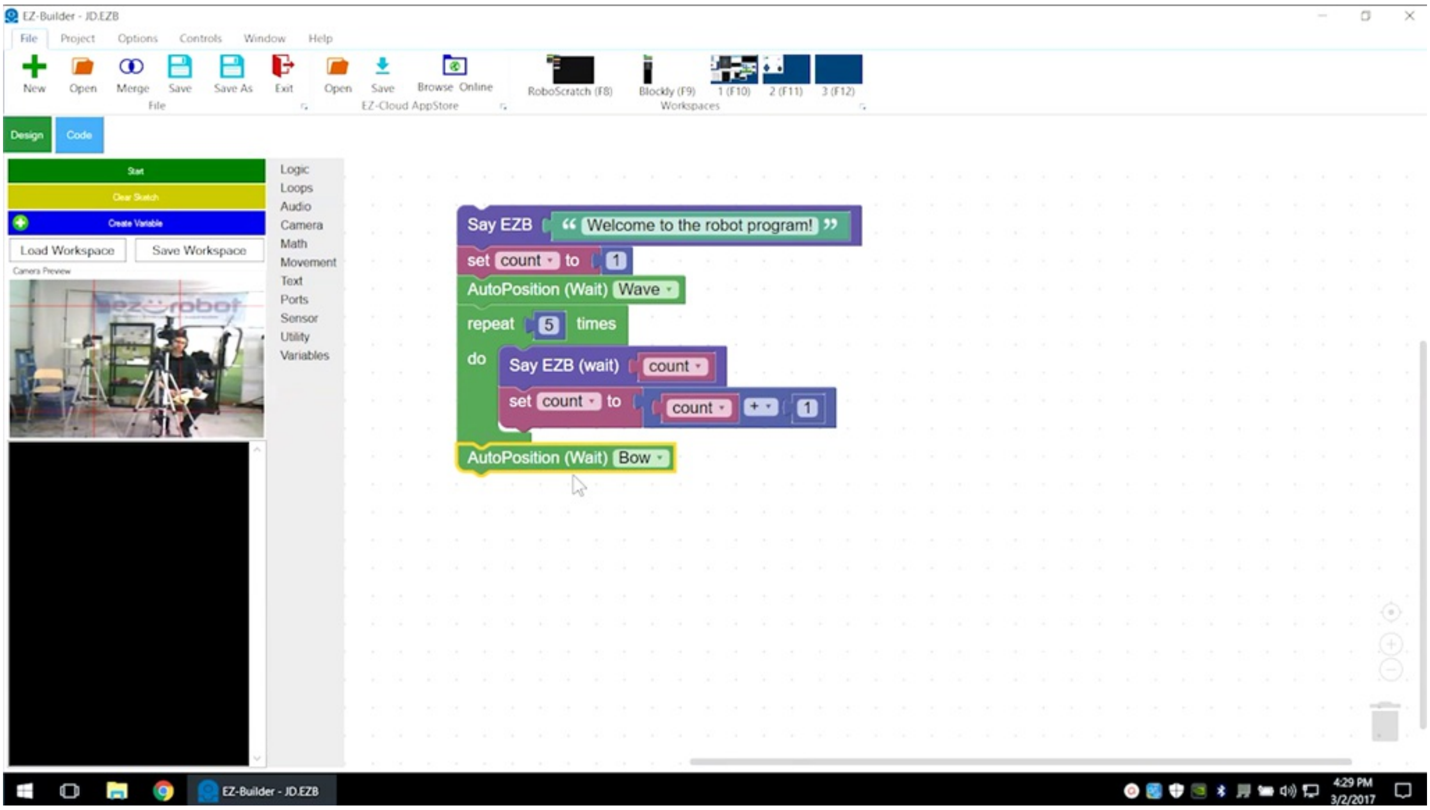
Blockly Audio

Audio can be output through either the **EZ-B** controller or the **PC** itself.



Blockly Logic

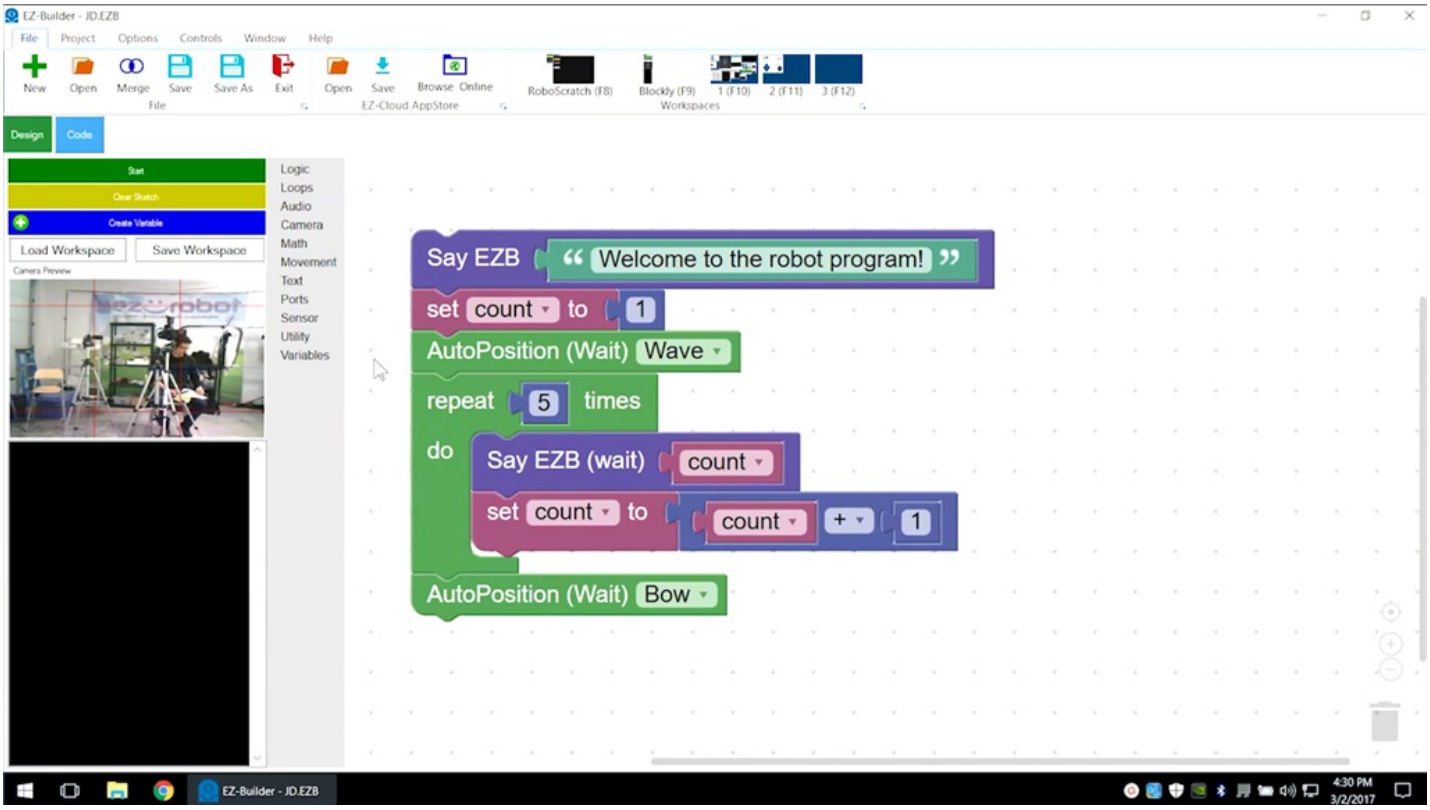
Blockly uses coding elements such as variables to count, track, and repeat.



```
Say EZB “ Welcome to the robot program! ”
set count to 1
AutoPosition (Wait) Wave
repeat 5 times
do  Say EZB (wait) count
    set count to count + 1
AutoPosition (Wait) Bow
```

Blockly Execution

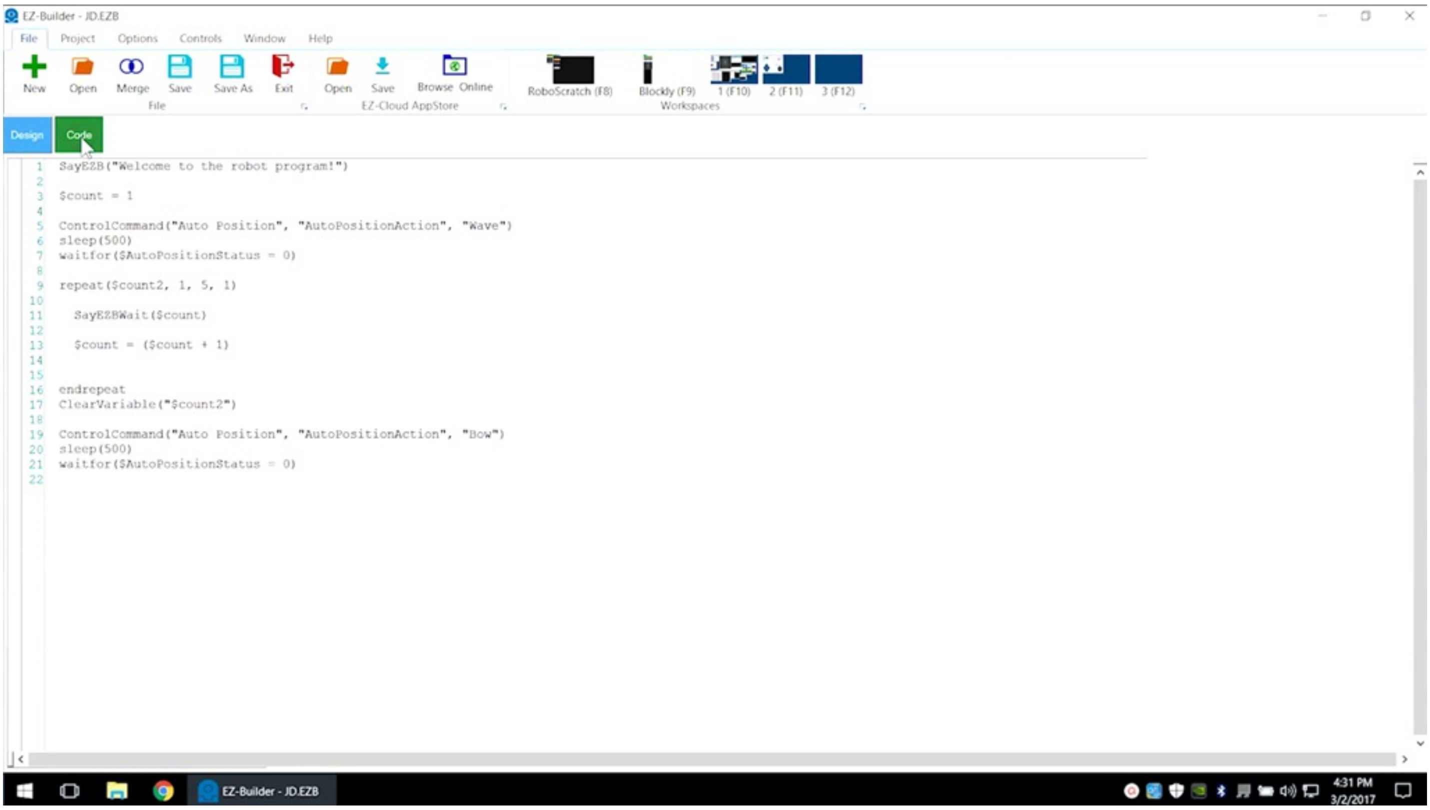
Click on the green **Start** button to execute the program.



```
Say EZB " Welcome to the robot program! "
set count to 1
AutoPosition (Wait) Wave
repeat 5 times
do  Say EZB (wait) count
    set count to count + 1
AutoPosition (Wait) Bow
```

Blockly Source Code

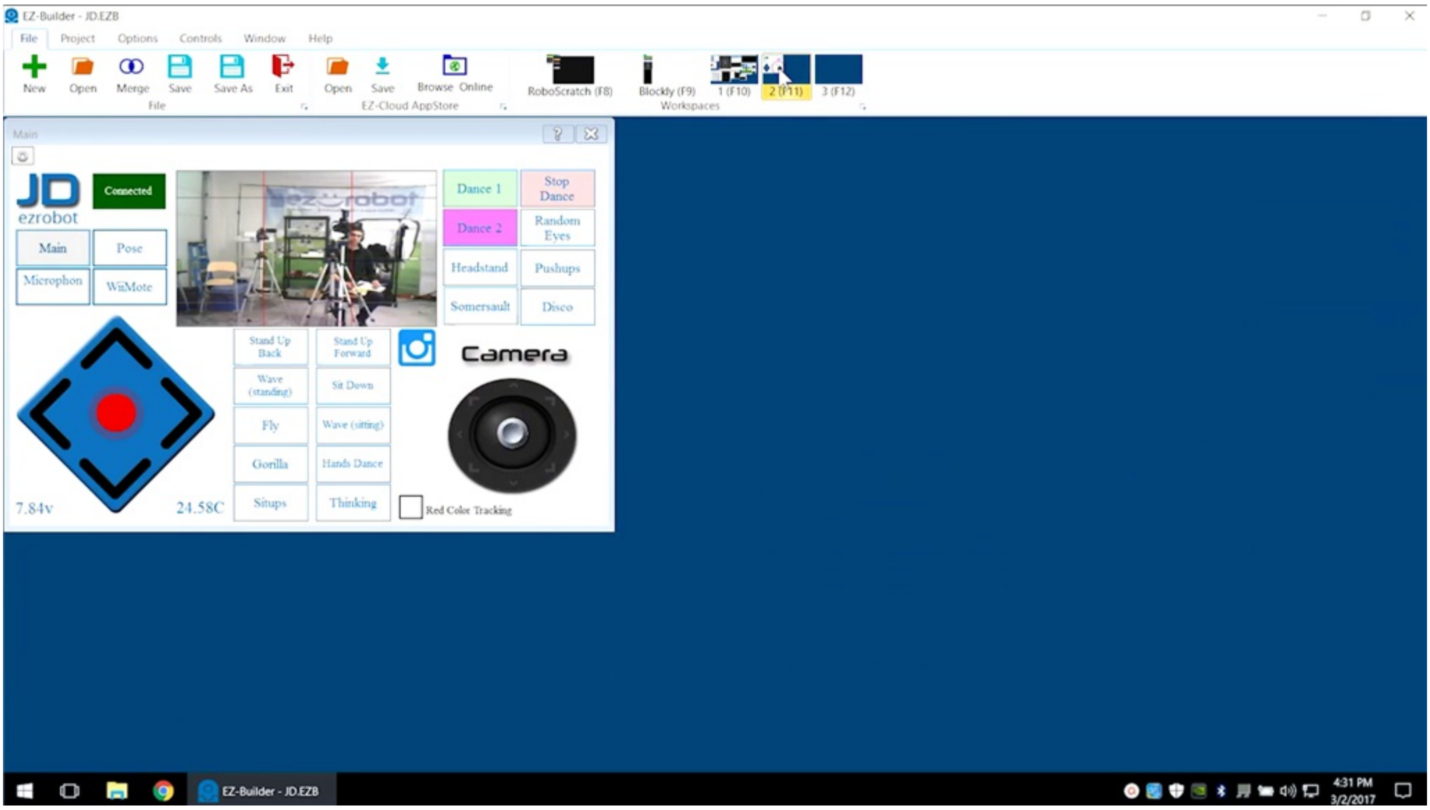
Click on the **Code** button to view the generated **EZ-Script** code.

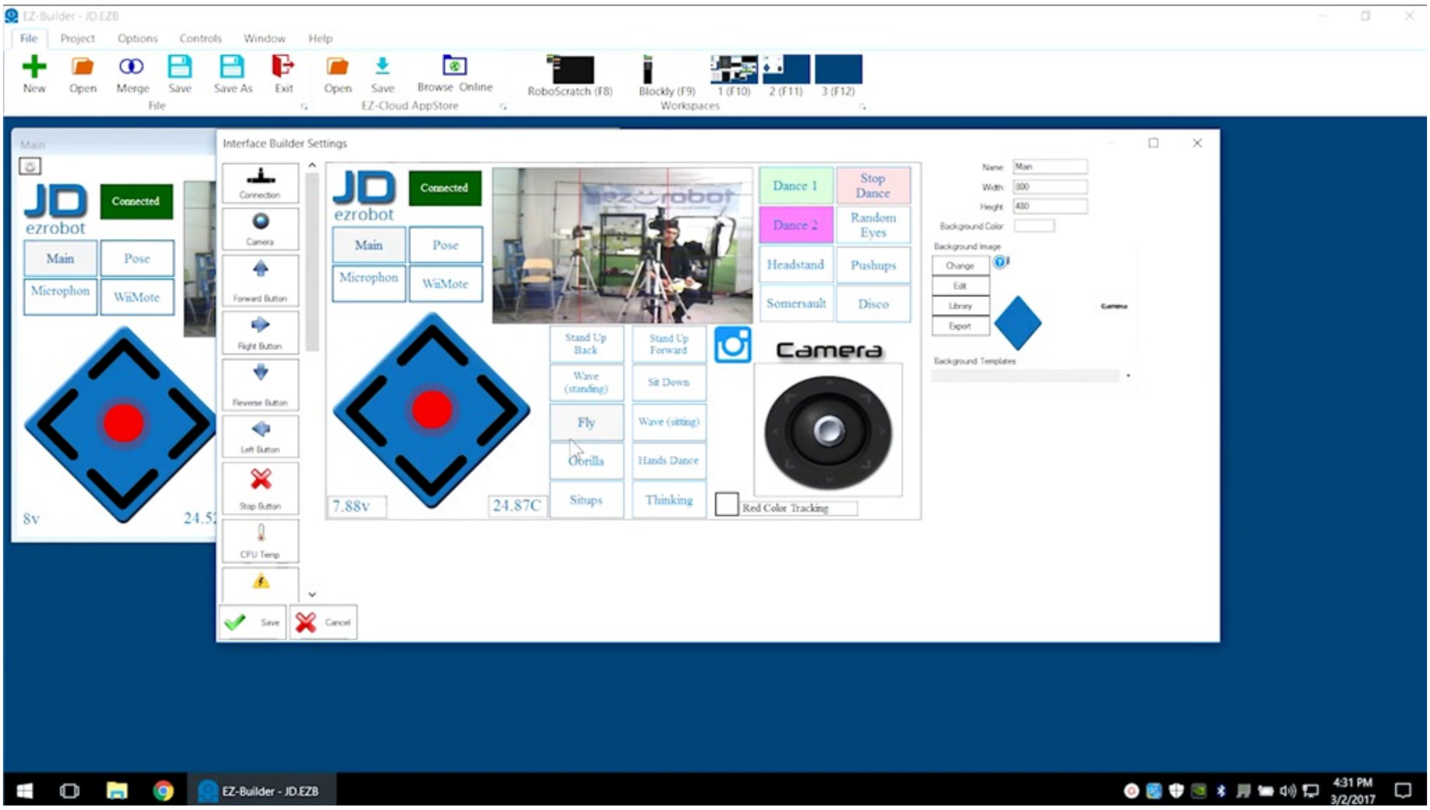


```
SayEZB("Welcome to the robot program!")

$count = 1

ControlCommand("Auto Position", "AutoPositionAction", "Wave")
sleep(500)
waitfor($AutoPositionStatus = 0)

repeat($count2, 1, 5, 1)

  SayEZBWait($count)

  $count = ($count + 1)


endrepeat
ClearVariable("$count2")

ControlCommand("Auto Position", "AutoPositionAction", "Bow")
sleep(500)
waitfor($AutoPositionStatus = 0)

```

Additional Workspaces

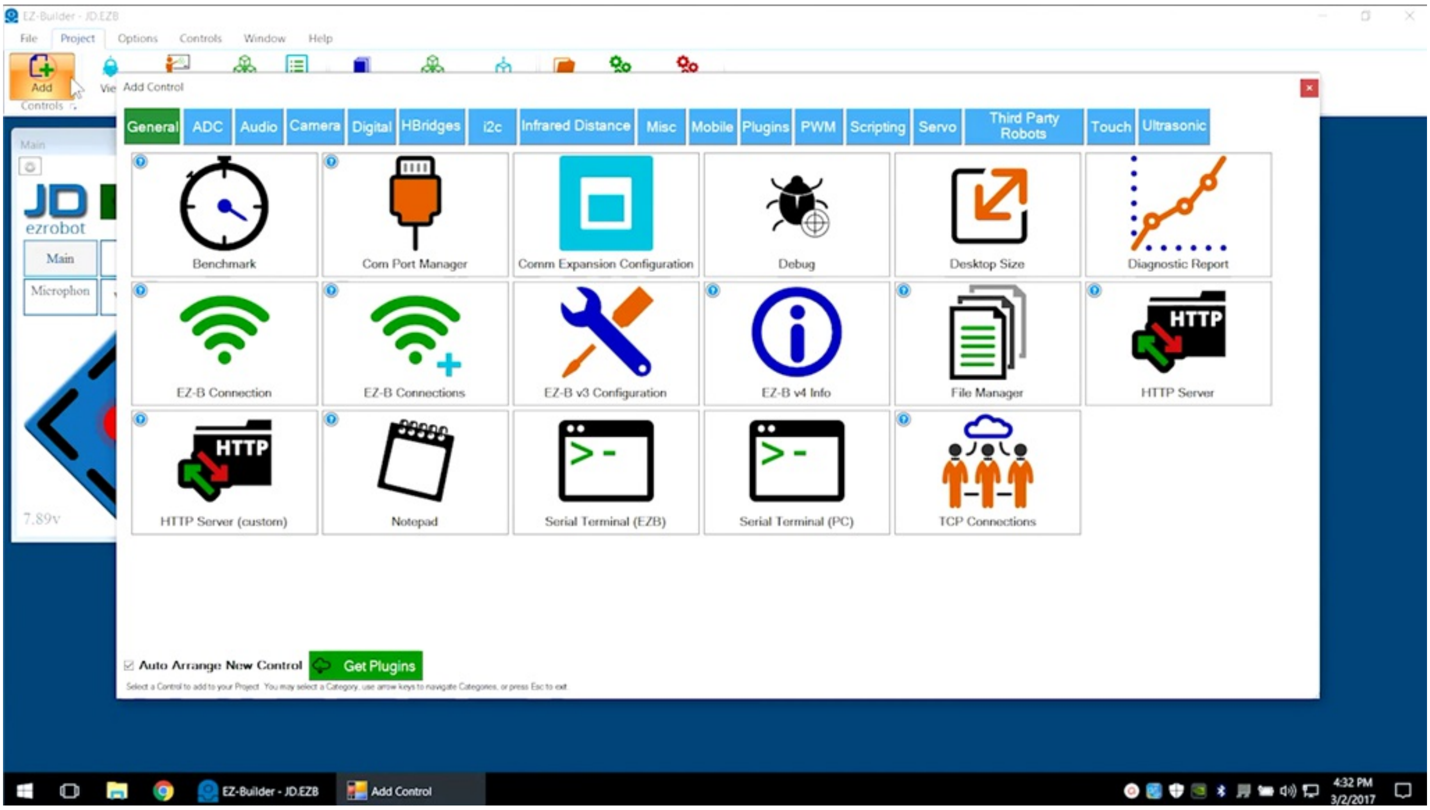
Select **Workspace 2** or **3** for more space to add controls. **Main** control shows the interface for creating an **EZ-Robot** mobile app. Click on the gear to customize the app interface.





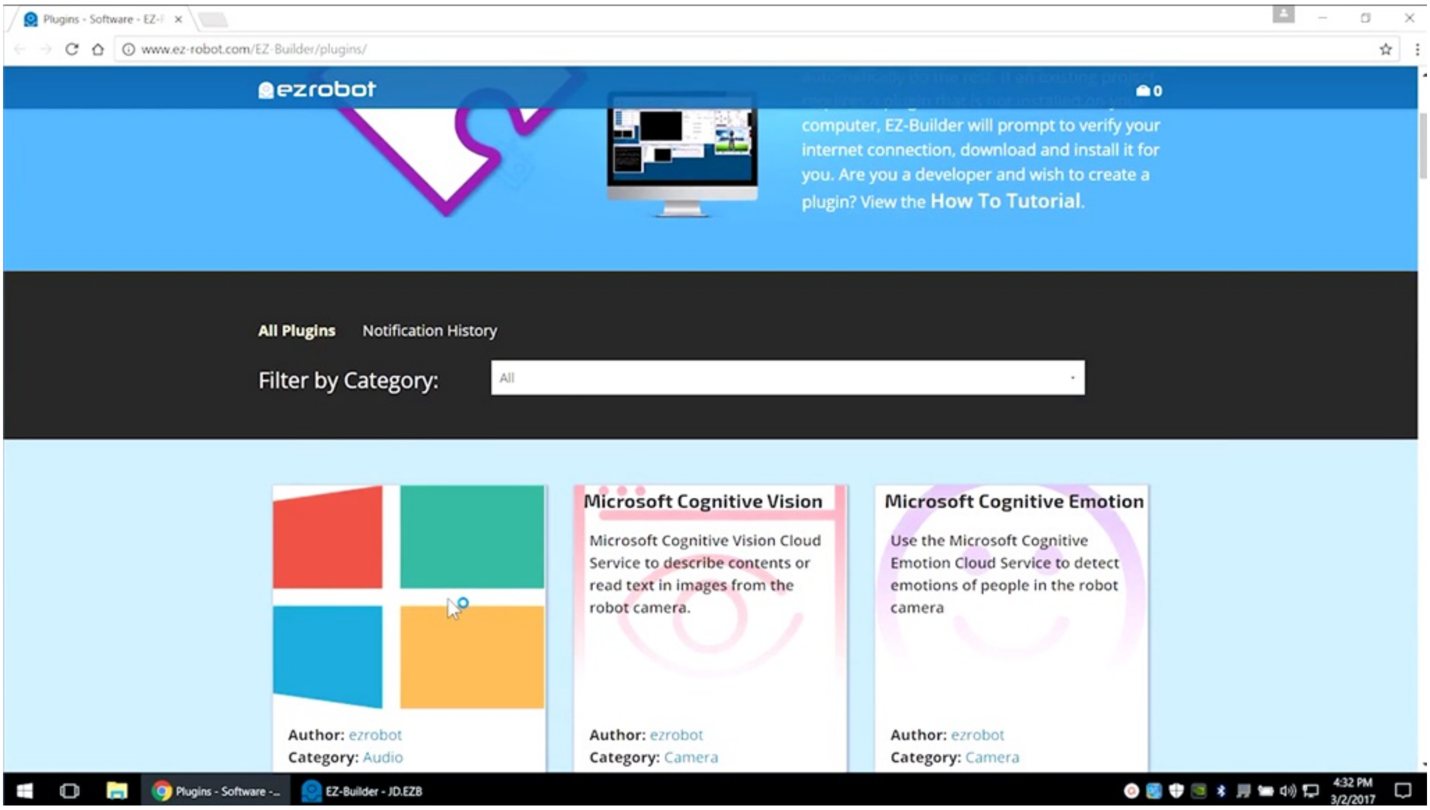
Adding Controls

Find more controls through the **Project** -> **Add** menu.



Third Party Plugins

Third party plugins can also be downloaded. These are added under **Plugins**.



automatically do the rest. If an existing project requires a plugin that is not installed on your computer, EZ-Builder will prompt to verify your internet connection, download and install it for you. Are you a developer and wish to create a plugin? View the **How To Tutorial**.

Filter by Category: All

Microsoft Cognitive Vision

Microsoft Cognitive Vision Cloud Service to describe contents or read text in images from the robot camera.

Author: ezrobot
Category: Camera

Microsoft Cognitive Emotion

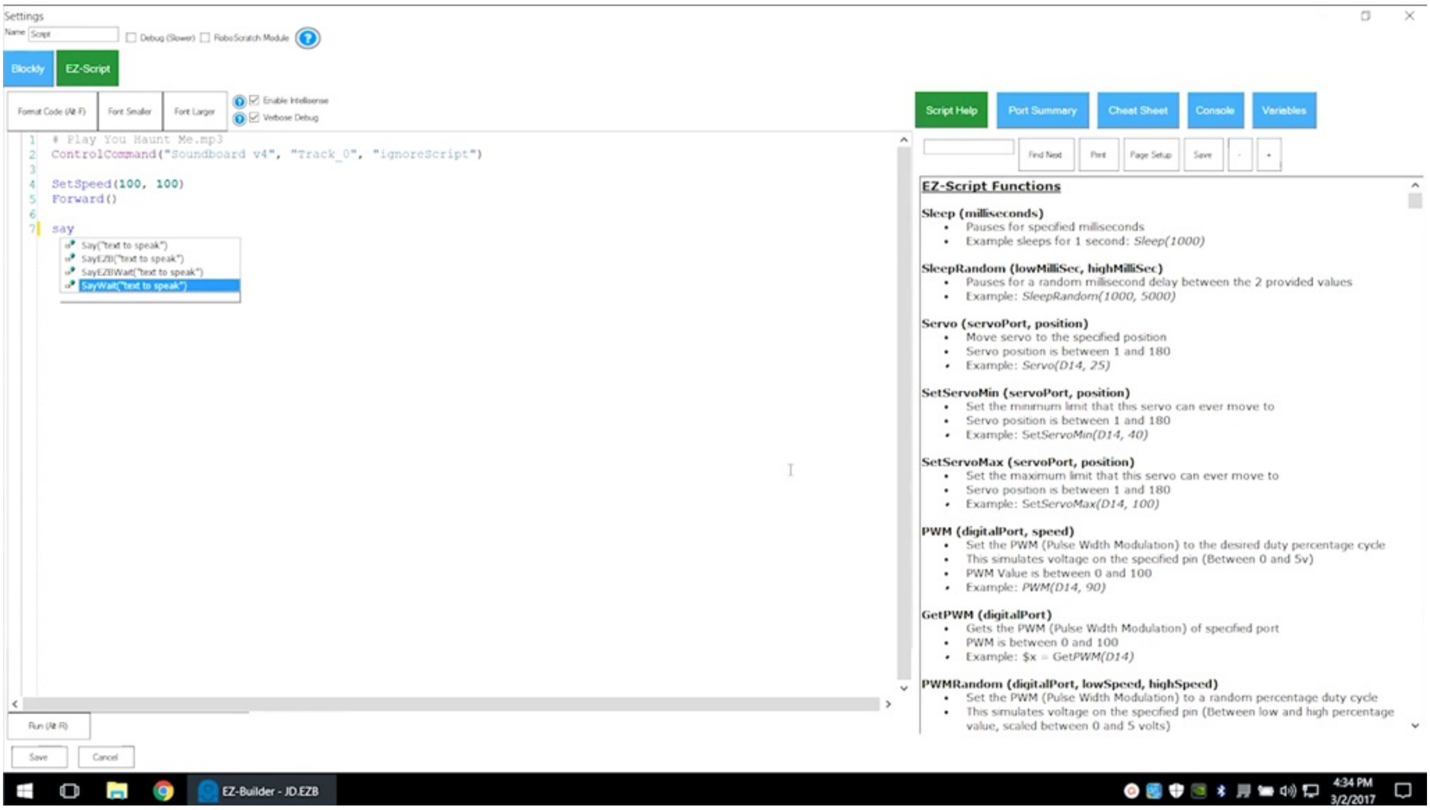
Use the Microsoft Cognitive Emotion Cloud Service to detect emotions of people in the robot camera

Author: ezrobot
Category: Camera

Author: ezrobot
Category: Audio

Scripting

Scripting can be used to create custom controls with the **Blockly** editor. **IntelliSense** will automatically show the available syntax options.



```
# Play You Haunt Me.mp3
ControlCommand("Soundboard v4", "Track_0", "ignoreScript")

SetSpeed(100, 100)
Forward()

say
```

EZ-Script Functions

Sleep (milliseconds)
- Pauses for specified milliseconds
- Example sleeps for 1 second: *Sleep(1000)*

SleepRandom (lowMilliSec, highMilliSec)
- Pauses for a random millisecond delay between the 2 provided values
- Example: *SleepRandom(1000, 5000)*

Servo (servoPort, position)
- Move servo to the specified position
- Servo position is between 1 and 180
- Example: *Servo(D14, 25)*

SetServoMin (servoPort, position)
- Set the minimum limit that this servo can ever move to
- Servo position is between 1 and 180
- Example: Set*ServoMin(D14, 40)*

SetServoMax (servoPort, position)
- Set the maximum limit that this servo can ever move to
- Servo position is between 1 and 180
- Example: Set*ServoMax(D14, 100)*

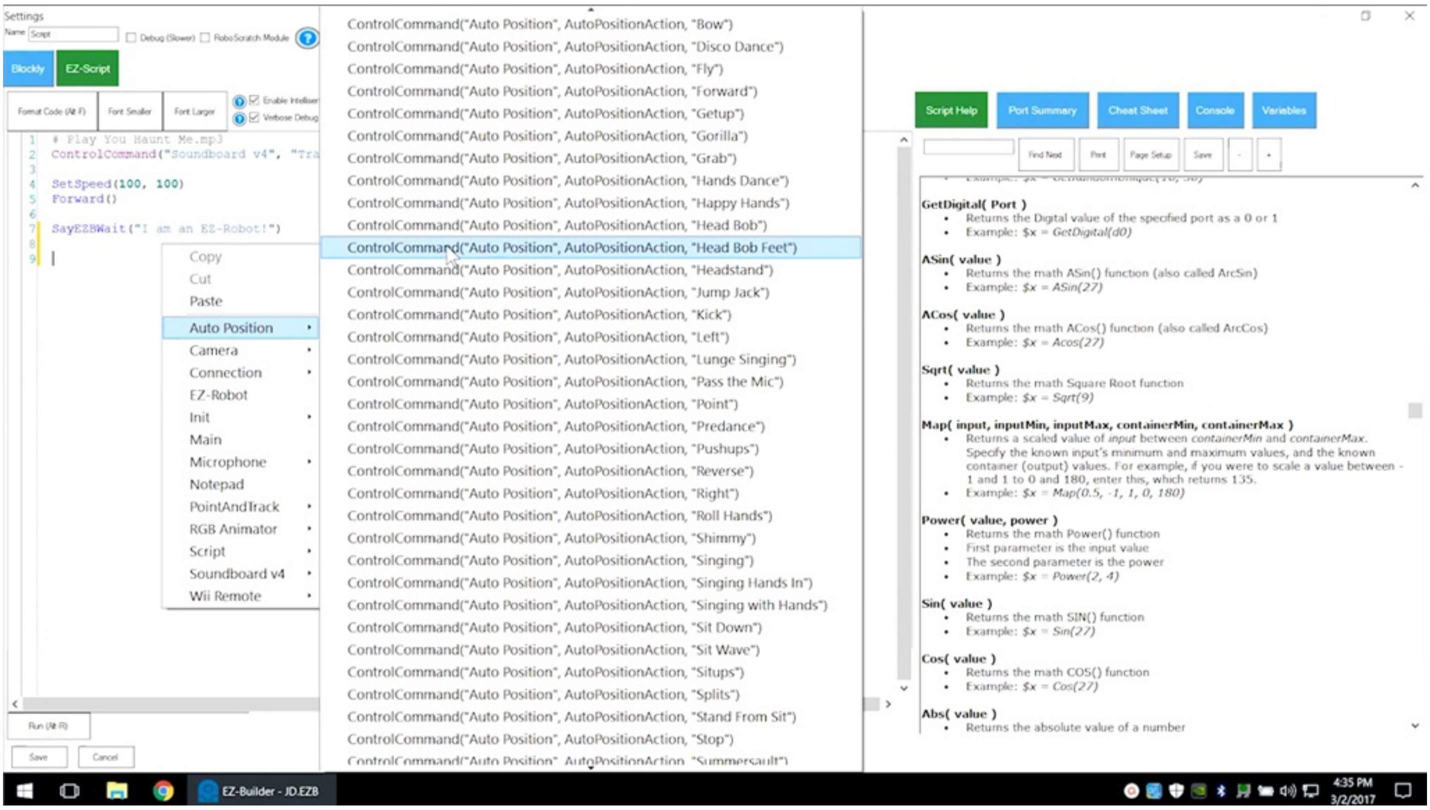
PWM (digitalPort, speed)
- Set the PWM (Pulse Width Modulation) to the desired duty percentage cycle
- This simulates voltage on the specified pin (Between 0 and 5v)
- PWM Value is between 0 and 100
- Example: *PWM(D14, 90)*

GetPWM (digitalPort)
- Gets the PWM (Pulse Width Modulation) of specified port
- PWM is between 0 and 100
- Example: $x = Get*PWM(D14)*

PWMRandom (digitalPort, lowSpeed, highSpeed)
- Set the PWM (Pulse Width Modulation) to a random percentage duty cycle
- This simulates voltage on the specified pin (Between low and high percentage value, scaled between 0 and 5 volts)

Scripting Control Options

See all of the control options by right-clicking in the editor, or by selecting the **Cheat Sheet**.



```
# Play You Haunt Me.mp3
ControlCommand("Soundboard v4", "Tra

SetSpeed(100, 100)
Forward()

SayEZBWait("I am an EZ-Robot!")

```

ControlCommand("Auto Position", AutoPositionAction, "Bow")
ControlCommand("Auto Position", AutoPositionAction, "Disco Dance")
ControlCommand("Auto Position", AutoPositionAction, "Fly")
ControlCommand("Auto Position", AutoPositionAction, "Forward")
ControlCommand("Auto Position", AutoPositionAction, "Getup")
ControlCommand("Auto Position", AutoPositionAction, "Gorilla")
ControlCommand("Auto Position", AutoPositionAction, "Grab")
ControlCommand("Auto Position", AutoPositionAction, "Hands Dance")
ControlCommand("Auto Position", AutoPositionAction, "Happy Hands")
ControlCommand("Auto Position", AutoPositionAction, "Head Bob")
ControlCommand("Auto Position", AutoPositionAction, "Head Bob Feet")
ControlCommand("Auto Position", AutoPositionAction, "Headstand")
ControlCommand("Auto Position", AutoPositionAction, "Jump Jack")
ControlCommand("Auto Position", AutoPositionAction, "Kick")
ControlCommand("Auto Position", AutoPositionAction, "Left")
ControlCommand("Auto Position", AutoPositionAction, "Lunge Singing")
ControlCommand("Auto Position", AutoPositionAction, "Pass the Mic")
ControlCommand("Auto Position", AutoPositionAction, "Point")
ControlCommand("Auto Position", AutoPositionAction, "Predance")
ControlCommand("Auto Position", AutoPositionAction, "Pushups")
ControlCommand("Auto Position", AutoPositionAction, "Reverse")
ControlCommand("Auto Position", AutoPositionAction, "Right")
ControlCommand("Auto Position", AutoPositionAction, "Roll Hands")
ControlCommand("Auto Position", AutoPositionAction, "Shimmy")
ControlCommand("Auto Position", AutoPositionAction, "Singing")
ControlCommand("Auto Position", AutoPositionAction, "Singing Hands In")
ControlCommand("Auto Position", AutoPositionAction, "Singing with Hands")
ControlCommand("Auto Position", AutoPositionAction, "Sit Down")
ControlCommand("Auto Position", AutoPositionAction, "Sit Wave")
ControlCommand("Auto Position", AutoPositionAction, "Situps")
ControlCommand("Auto Position", AutoPositionAction, "Splits")
ControlCommand("Auto Position", AutoPositionAction, "Stand From Sit")
ControlCommand("Auto Position", AutoPositionAction, "Stop")
ControlCommand("Auto Position", AutoPositionAction, "Summersault")

GetDigital(Port)
- Returns the Digital value of the specified port as a 0 or 1
- Example: $x = *GetDigital(d0)*

ASin(value)
- Returns the math ASin() function (also called ArcSin)
- Example: $x = *ASin(27)*

ACos(value)
- Returns the math ACos() function (also called ArcCos)
- Example: $x = *Acos(27)*

Sqrt(value)
- Returns the math Square Root function
- Example: $x = *Sqrt(9)*

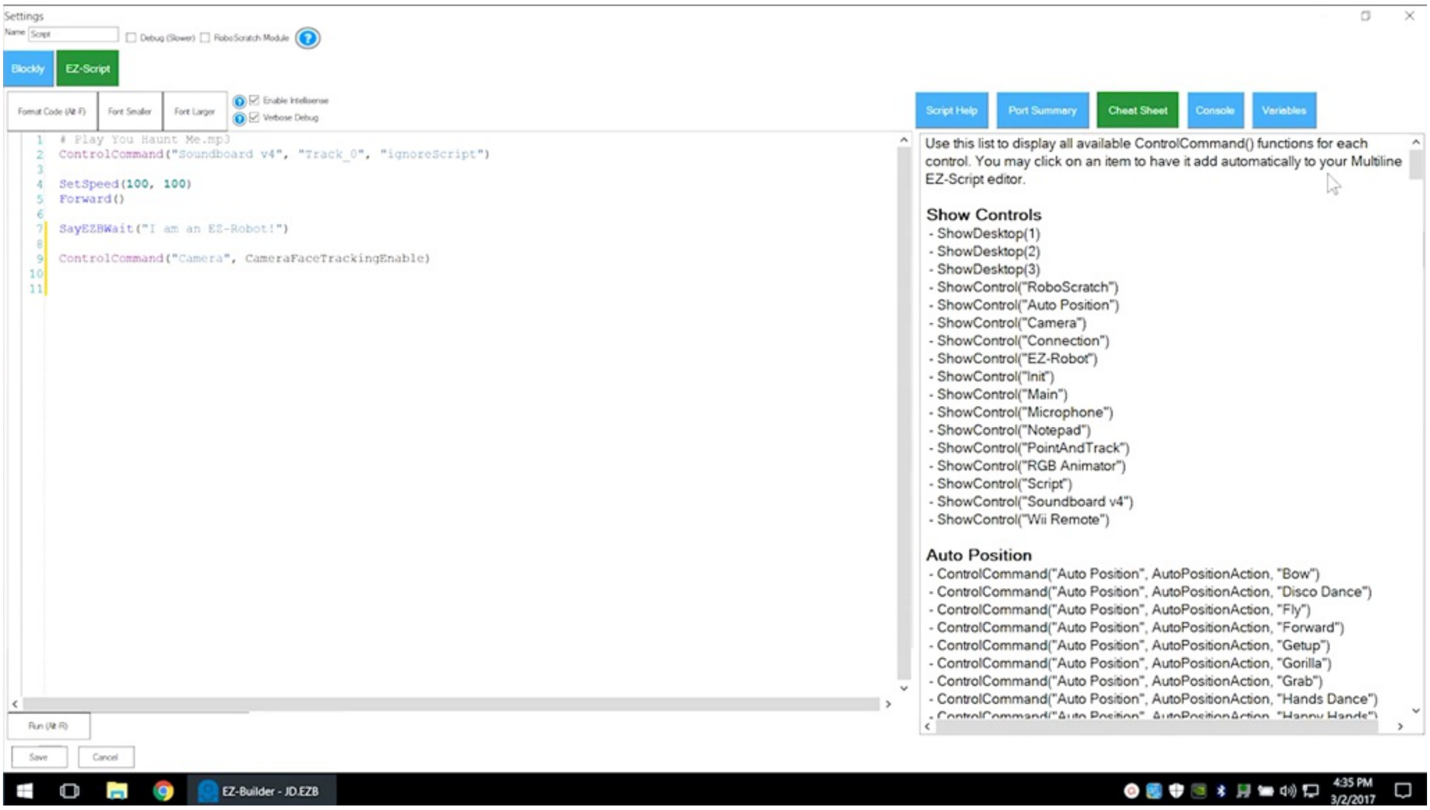
Map(input, inputMin, inputMax, containerMin, containerMax)
- Returns a scaled value of *input* between *containerMin* and *containerMax*. Specify the known input's minimum and maximum values, and the known container (output) values. For example, if you were to scale a value between -1 and 1 to 0 and 180, enter this, which returns 135.
- Example: $x = *Map(0.5, -1, 1, 0, 180)*

Power(value, power)
- Returns the math Power() function
- First parameter is the input value
- The second parameter is the power
- Example: $x = *Power(2, 4)*

Sin(value)
- Returns the math SIN() function
- Example: $x = *Sin(27)*

Cos(value)
- Returns the math COS() function
- Example: $x = *Cos(27)*

Abs(value)
- Returns the absolute value of a number



```
# Play You Haunt Me.mp3
ControlCommand("Soundboard v4", "Track_0", "ignoreScript")

SetSpeed(100, 100)
Forward()

SayEZBWait("I am an EZ-Robot!")

ControlCommand("Camera", CameraFaceTrackingEnable)
```

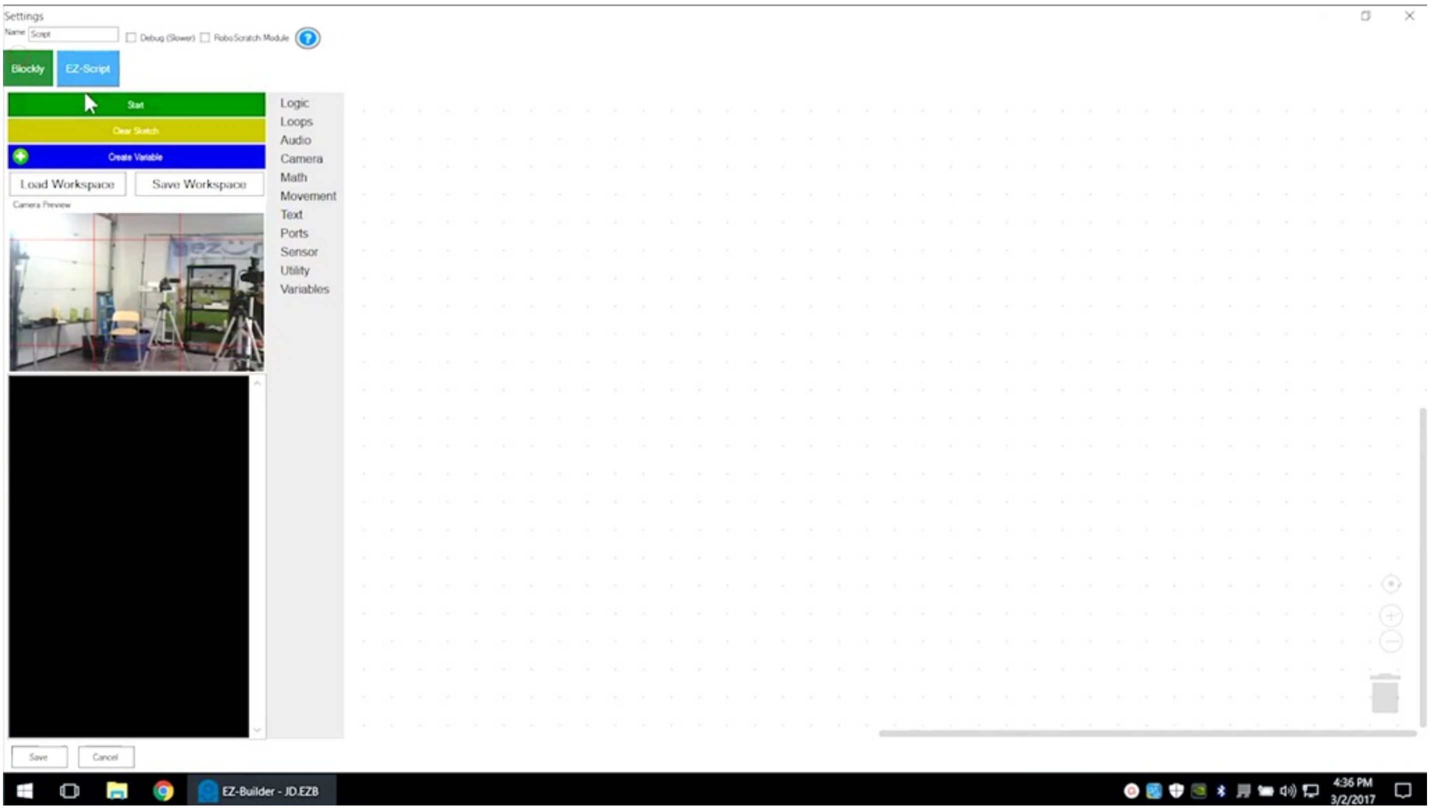
Use this list to display all available ControlCommand() functions for each control. You may click on an item to have it add automatically to your Multiline EZ-Script editor.

Show Controls
- ShowDesktop(1)
- ShowDesktop(2)
- ShowDesktop(3)
- ShowControl("RoboScratch")
- ShowControl("Auto Position")
- ShowControl("Camera")
- ShowControl("Connection")
- ShowControl("EZ-Robot")
- ShowControl("Init")
- ShowControl("Main")
- ShowControl("Microphone")
- ShowControl("Notepad")
- ShowControl("PointAndTrack")
- ShowControl("RGB Animator")
- ShowControl("Script")
- ShowControl("Soundboard v4")
- ShowControl("Wii Remote")

Auto Position
- ControlCommand("Auto Position", AutoPositionAction, "Bow")
- ControlCommand("Auto Position", AutoPositionAction, "Disco Dance")
- ControlCommand("Auto Position", AutoPositionAction, "Fly")
- ControlCommand("Auto Position", AutoPositionAction, "Forward")
- ControlCommand("Auto Position", AutoPositionAction, "Getup")
- ControlCommand("Auto Position", AutoPositionAction, "Gorilla")
- ControlCommand("Auto Position", AutoPositionAction, "Grab")
- ControlCommand("Auto Position", AutoPositionAction, "Hands Dance")
- ControlCommand("Auto Position", AutoPositionAction, "Happy Hands")

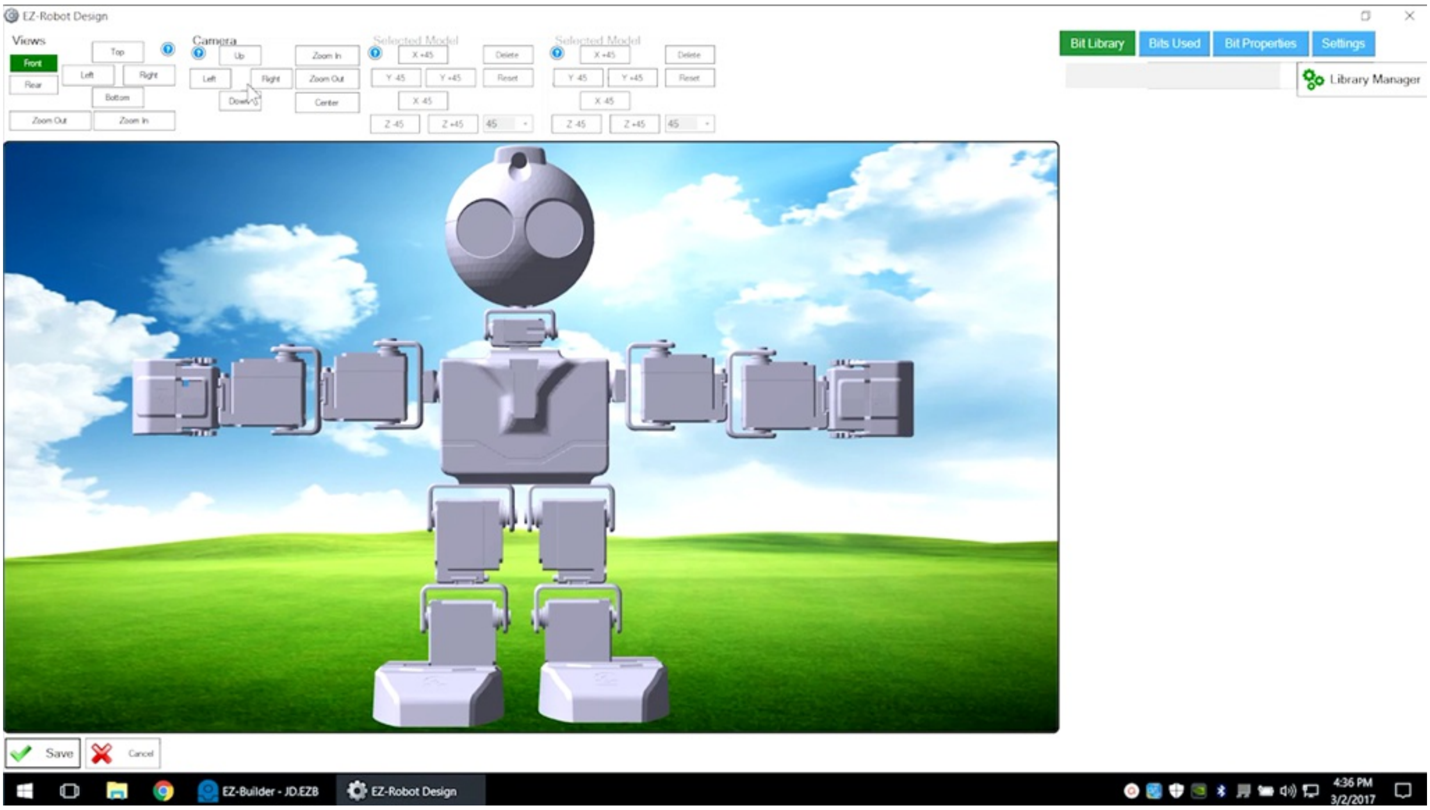
EZ-Script and Blockly

Editing in **EZ-Script** will clear the **Blockly** editor.



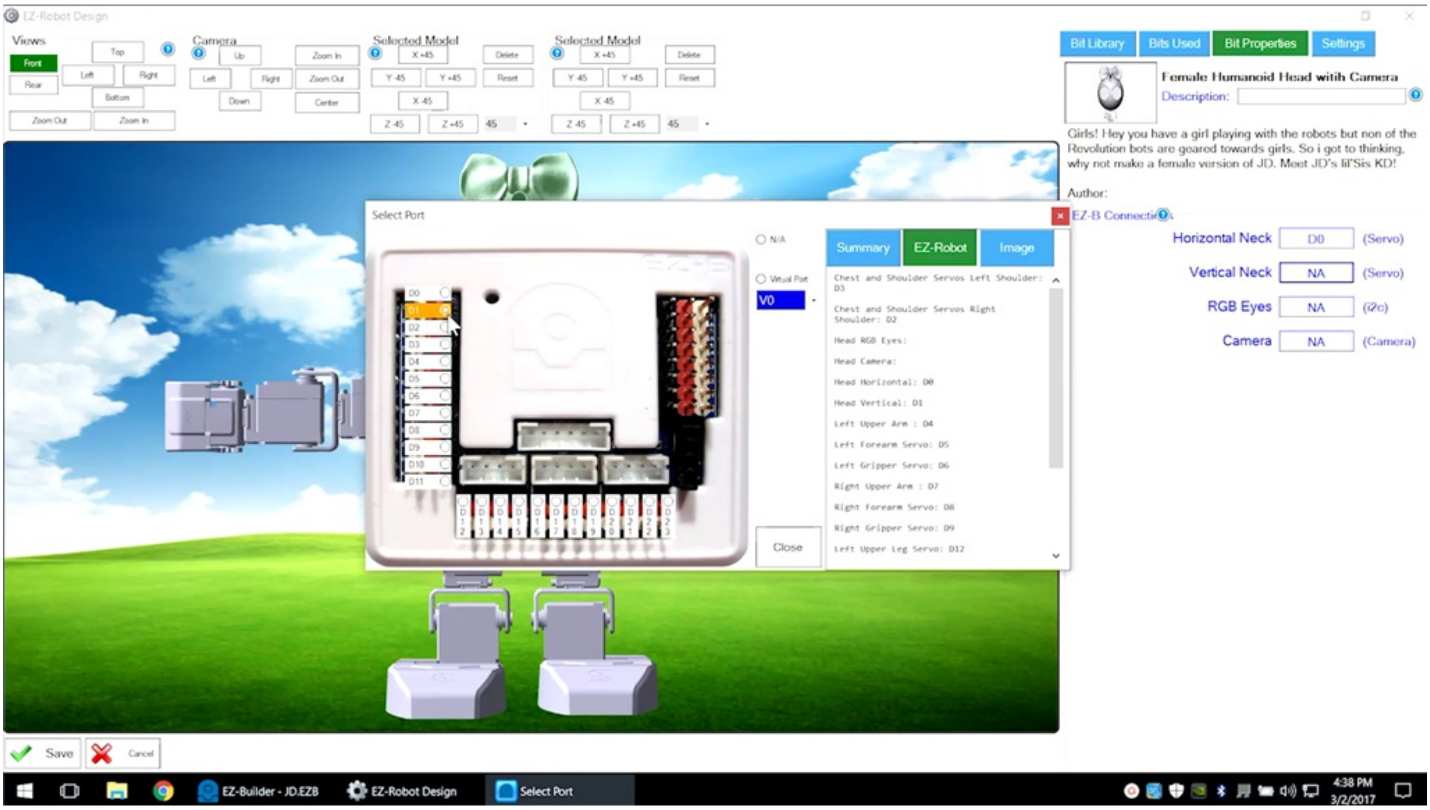
Design Mode

Enter **Design Mode** by selecting **Project -> Design**. Design options are available through the **Bit Library**.



Connecting Custom Components

Choose which ports will be used for connecting custom components.



Chest and Shoulder Servos Left Shoulder: D3

Chest and Shoulder Servos Right Shoulder: D2

Head RGB Eyes:

Head Camera:

Head Horizontal: D0

Head Vertical: D1

Left Upper Arm : D4

Left Forearm Servo: D5

Left Gripper Servo: D6

Right Upper Arm : D7

Right Forearm Servo: D8

Right Gripper Servo: D9

Left Upper Leg Servo: D12

Female Humanoid Head witih Camera

Description:

Girls! Hey you have a girl playing with the robots but non of the Revolution bots are geared towards girls. So i got to thinking, why not make a female version of JD. Meet JD's lil'Sis KD!

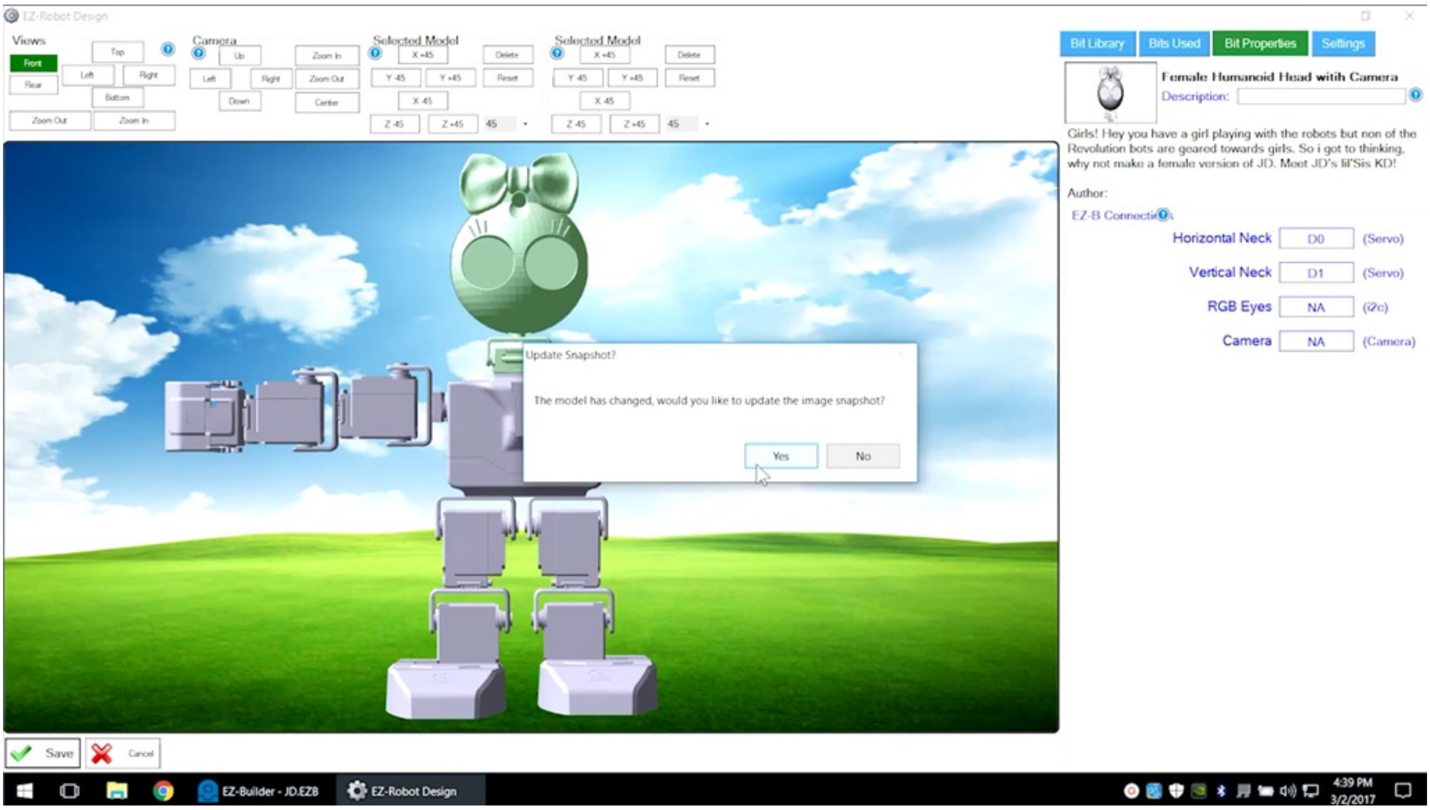
Author:

EZ-B Connections

Horizontal Neck	D0	(Servo)
Vertical Neck	NA	(Servo)
RGB Eyes	NA	(i2c)
Camera	NA	(Camera)

Updating Build Instructions

Select **Save**. Updating the robot image will change the build instructions.



Female Humanoid Head witih Camera

Description:

Girls! Hey you have a girl playing with the robots but non of the Revolution bots are geared towards girls. So i got to thinking, why not make a female version of JD. Meet JD's lil'Sis KD!

Author:

EZ-B Connections

Horizontal Neck	D0	(Servo)
Vertical Neck	D1	(Servo)
RGB Eyes	NA	(i2c)
Camera	NA	(Camera)

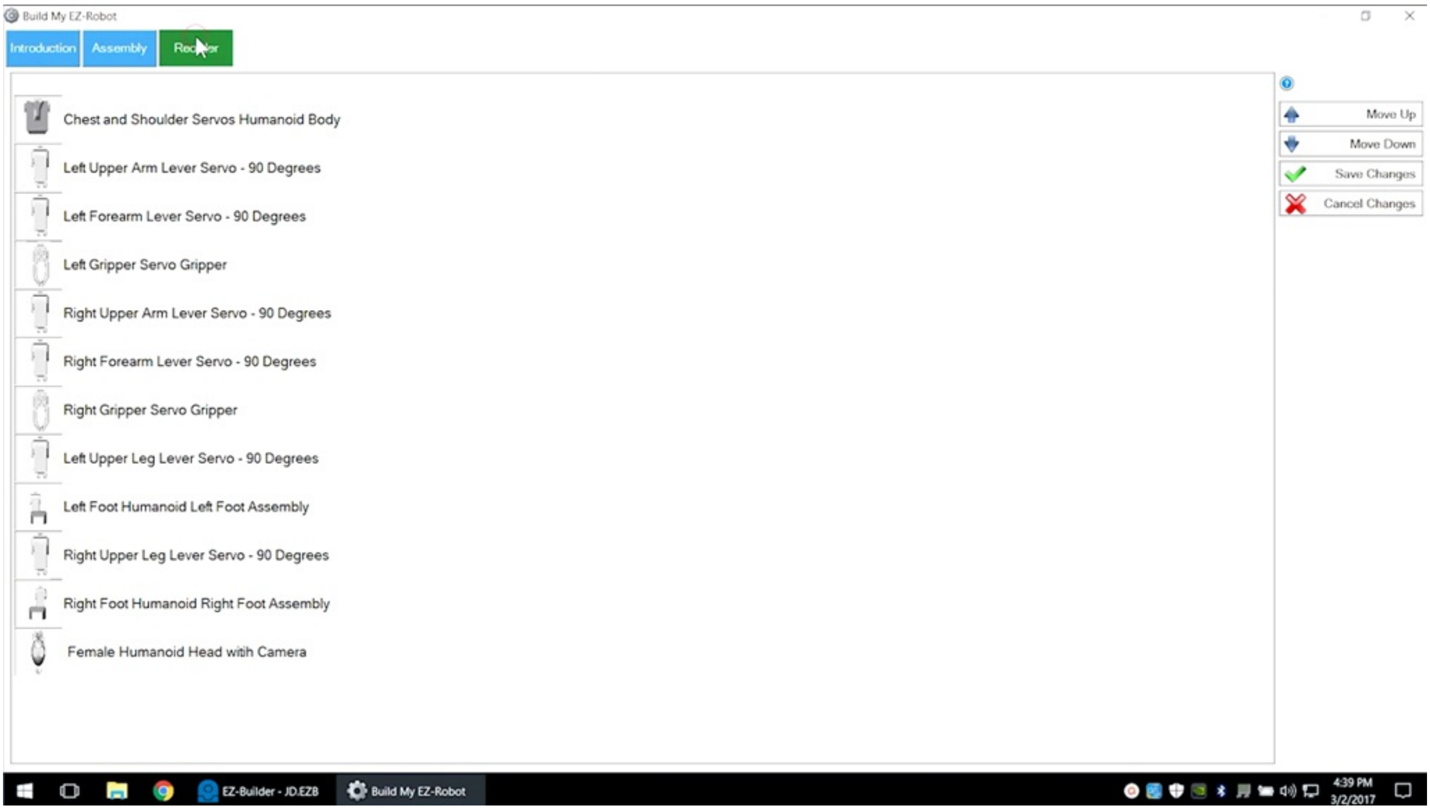
Update Snapshot?

The model has changed, would you like to update the image snapshot?

Yes No

Reorder Build Steps

Select the **Reorder** tab to change the order of build instructions.



Chest and Shoulder Servos Humanoid Body

Left Upper Arm Lever Servo - 90 Degrees

Left Forearm Lever Servo - 90 Degrees

Left Gripper Servo Gripper

Right Upper Arm Lever Servo - 90 Degrees

Right Forearm Lever Servo - 90 Degrees

Right Gripper Servo Gripper

Left Upper Leg Lever Servo - 90 Degrees

Left Foot Humanoid Left Foot Assembly

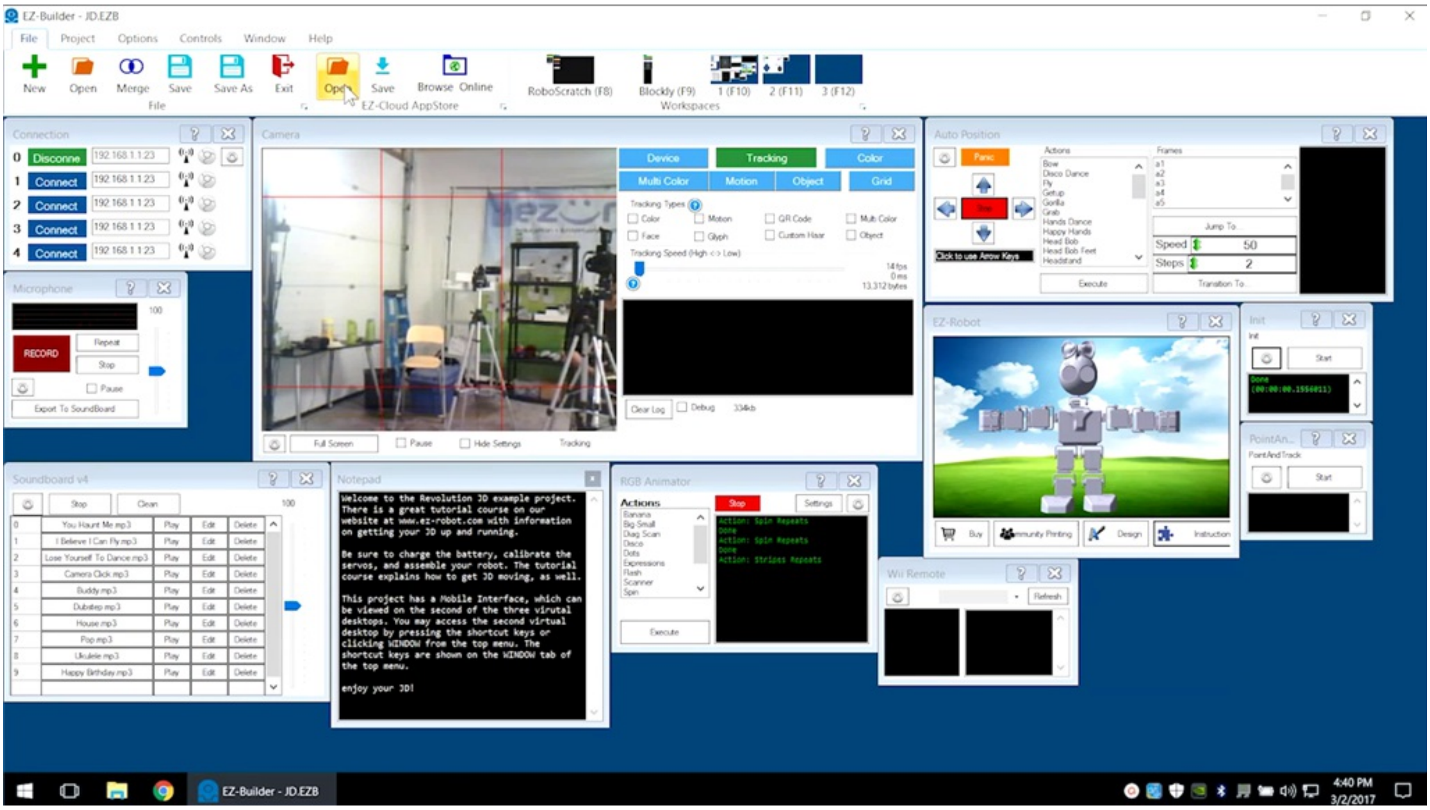
Right Upper Leg Lever Servo - 90 Degrees

Right Foot Humanoid Right Foot Assembly

Female Humanoid Head with Camera

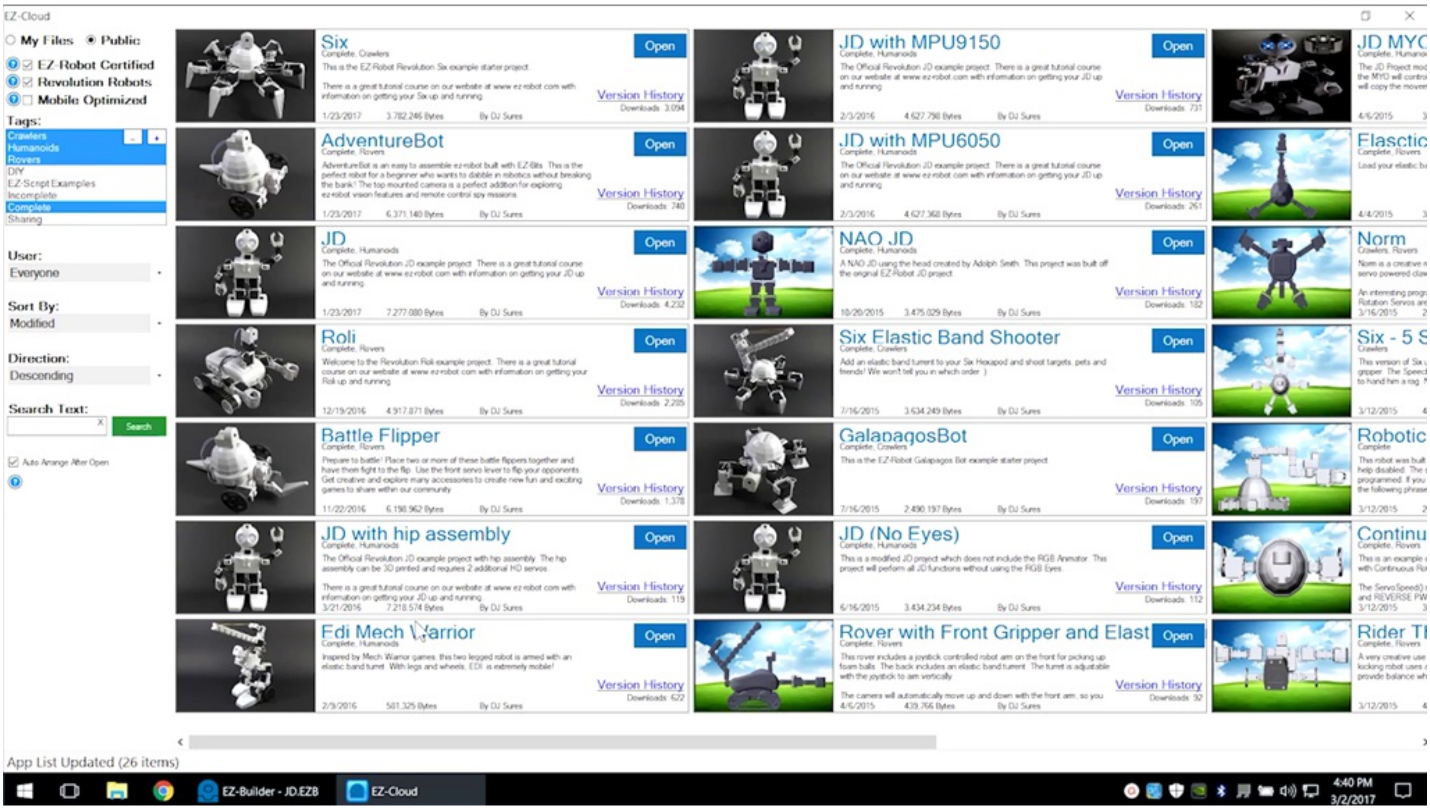
Share with EZ-Cloud

Access the **EZ-Cloud** through **File** -> **Open** to save and share programs.



Notepad

```
Welcome to the Revolution JD example project.
There is a great tutorial course on our
website at www.ez-robot.com with information
on getting your JD up and running.

Be sure to charge the battery, calibrate the
servos, and assemble your robot. The tutorial
course explains how to get JD moving, as well.

This project has a Mobile Interface, which can
be viewed on the second of the three virutal
desktops. You may access the second virtual
desktop by pressing the shortcut keys or
clicking WINDOW from the top menu. The
shortcut keys are shown on the WINDOW tab of
the top menu.

enjoy your JD!
```



Six — Complete, Crawlers
This is the EZ-Robot Revolution Six example starter project.
There is a great tutorial course on our website at www.ez-robot.com with information on getting your Six up and running.
1/23/2017 3,782,246 Bytes By DJ Sures — Downloads: 3,094

AdventureBot — Complete, Rovers
AdventureBot is an easy to assemble ez-robot built with EZ-Bits. This is the perfect robot for a beginner who wants to dabble in robotics without breaking the bank! The top mounted camera is a perfect addition for exploring ez-robot vision features and remote control spy missions.
1/23/2017 6,371,140 Bytes By DJ Sures — Downloads: 740

JD — Complete, Humanoids
The Official Revolution JD example project. There is a great tutorial course on our website at www.ez-robot.com with information on getting your JD up and running.
1/23/2017 7,277,080 Bytes By DJ Sures — Downloads: 4,232

Roli — Complete, Rovers
Welcome to the Revolution Roli example project. There is a great tutorial course on our website at www.ez-robot.com with information on getting your Roli up and running.
12/19/2016 4,917,871 Bytes By DJ Sures — Downloads: 2,205

Battle Flipper — Complete, Rovers
Prepare to battle! Place two or more of these battle flippers together and have them fight to the flip. Use the front servo lever to flip your opponents. Get creative and explore many accessories to create new fun and exciting games to share within our community.
11/22/2016 6,158,962 Bytes By DJ Sures — Downloads: 1,378

JD with hip assembly — Complete, Humanoids
The Official Revolution JD example project with hip assembly. The hip assembly can be 3D printed and requires 2 additional HD servos.
There is a great tutorial course on our website at www.ez-robot.com with information on getting your JD up and running.
3/21/2016 7,218,574 Bytes By DJ Sures — Downloads: 119

Edi Mech Warrior — Complete, Humanoids
Inspired by Mech Warrior games, this two legged robot is armed with an elastic band turret. With legs and wheels, EDI is extremely mobile!
2/9/2016 581,325 Bytes By DJ Sures — Downloads: 622

JD with MPU9150 — Complete, Humanoids
The Official Revolution JD example project. There is a great tutorial course on our website at www.ez-robot.com with information on getting your JD up and running.
2/3/2016 4,627,798 Bytes By DJ Sures — Downloads: 731

JD with MPU6050 — Complete, Humanoids
The Official Revolution JD example project. There is a great tutorial course on our website at www.ez-robot.com with information on getting your JD up and running.
2/3/2016 4,627,368 Bytes By DJ Sures — Downloads: 261

NAO JD — Complete, Humanoids
A NAO JD using the head created by Adolph Smith. This project was built off the original EZ-Robot JD project.
10/20/2015 3,475,029 Bytes By DJ Sures — Downloads: 182

Six Elastic Band Shooter — Complete, Crawlers
Add an elastic band turrent to your Six Hexapod and shoot targets, pets and friends! We won't tell you in which order :)
7/16/2015 3,634,249 Bytes By DJ Sures — Downloads: 105

GalapagosBot — Complete, Crawlers
This is the EZ-Robot Galapagos Bot example starter project.
7/16/2015 2,490,197 Bytes By DJ Sures — Downloads: 197

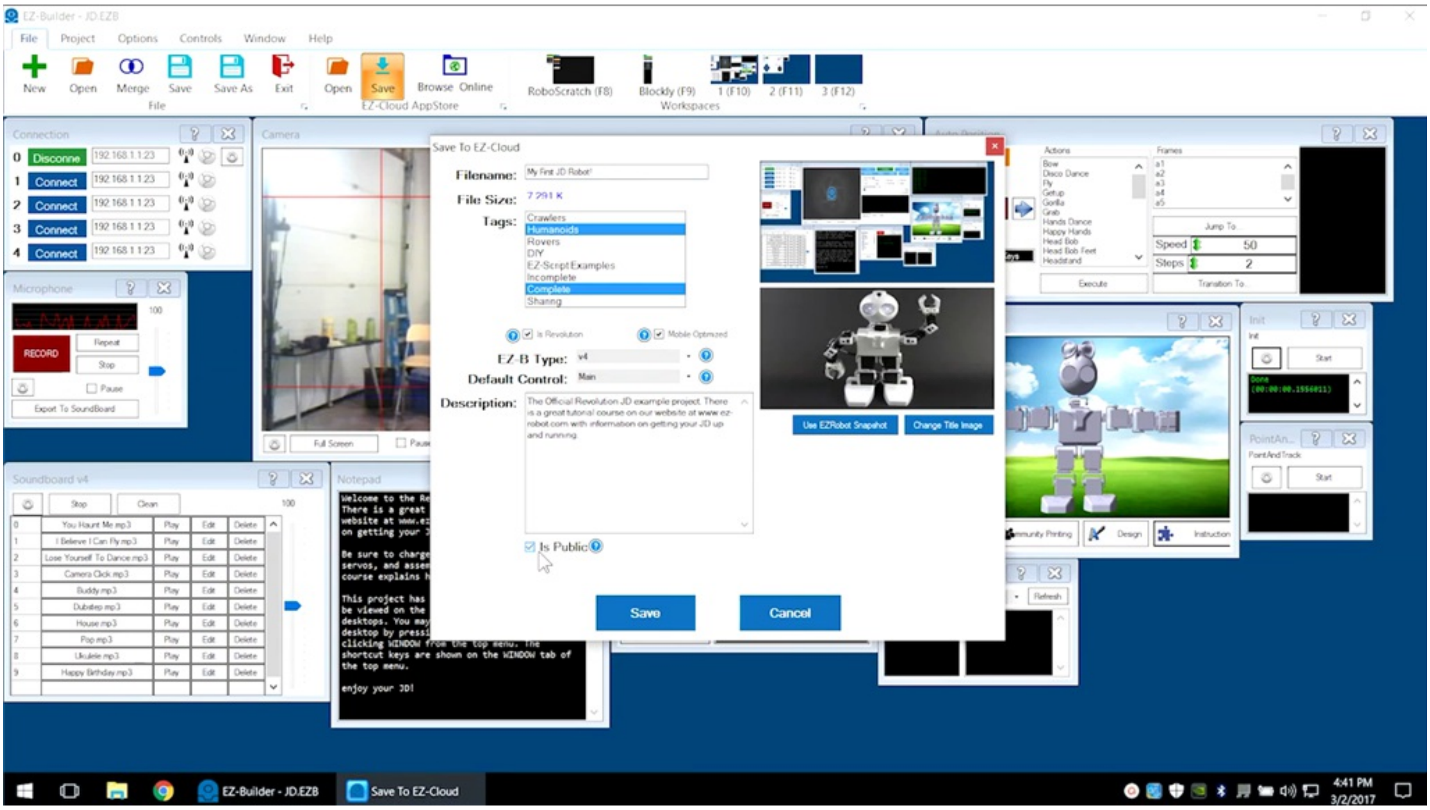
JD (No Eyes) — Complete, Humanoids
This is a modified JD project which does not include the RGB Animator. This project will perform all JD functions without using the RGB Eyes.
6/16/2015 3,434,234 Bytes By DJ Sures — Downloads: 112

Rover with Front Gripper and Elast — Complete, Rovers
This rover includes a joystick controlled robot arm on the front for picking up foam balls. The back includes an elastic band turrent. The turret is adjustable with the joystick to aim vertically.
The camera will automatically move up and down with the front arm, so you
4/6/2015 439,766 Bytes By DJ Sures — Downloads: 92

App List Updated (26 items)

Public Sharing

Check **Is Public** to share programs with others.



Save To EZ-Cloud

Filename: My First JD Robot!

File Size: 7 291 K

Tags: Crawlers, **Humanoids**, Rovers, DIY, EZ-Script Examples, Incomplete, **Complete**, Sharing

☑ Is Revolution ☑ Mobile Optimized

EZ-B Type: v4

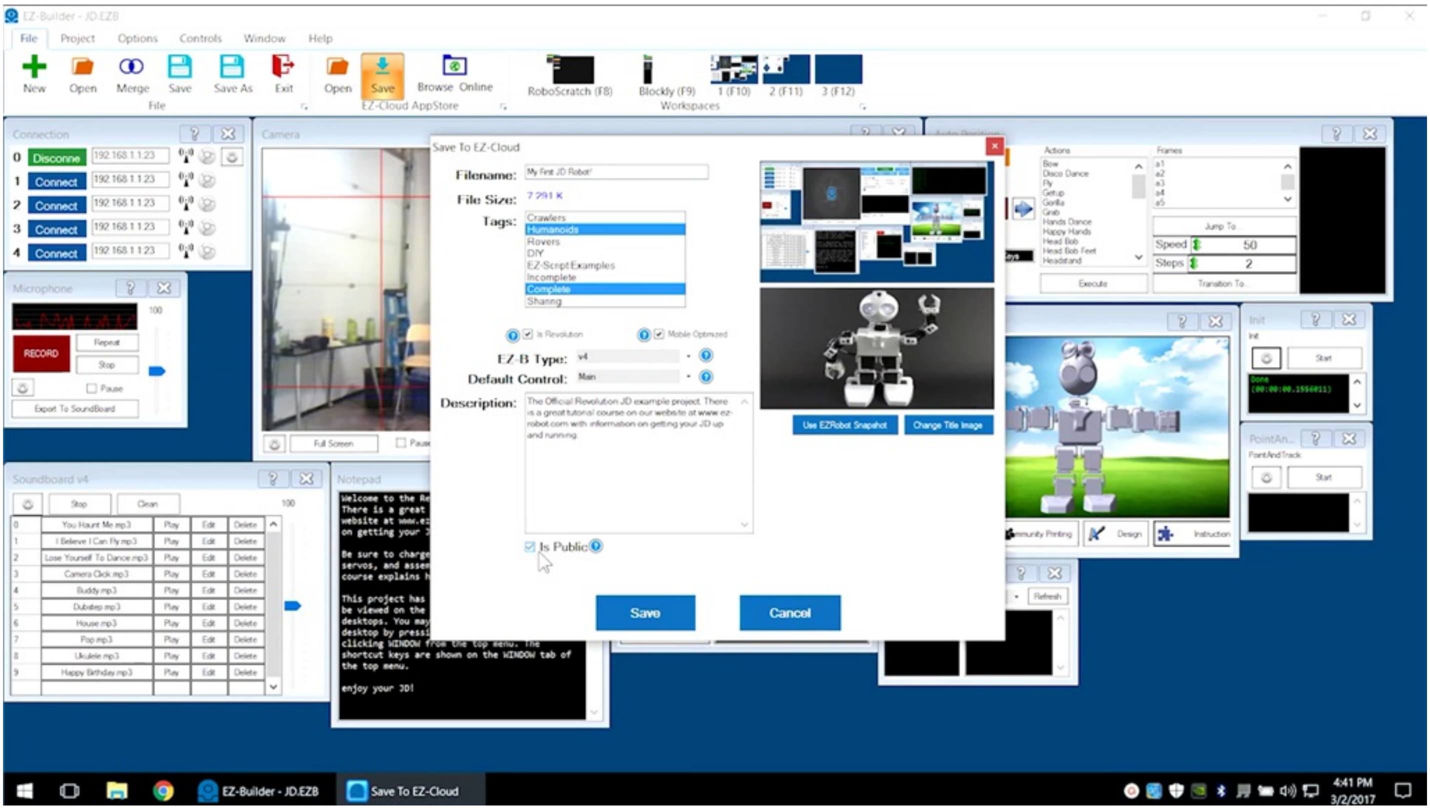
Default Control: Main

Description: The Official Revolution JD example project. There is a great tutorial course on our website at www.ez-robot.com with information on getting your JD up and running.

☑ Is Public

EZ-Cloud AppStore

The **EZ-Cloud AppStore** saves all revisions and logs change notes.



Save To EZ-Cloud

Filename: My First JD Robot!

File Size: 7 291 K

Tags: Crawlers, Humanoids, Rovers, DIY, EZ-Script Examples, Incomplete, Complete, Sharing

☑ Is Revolution ☑ Mobile Optimized

EZ-B Type: v4

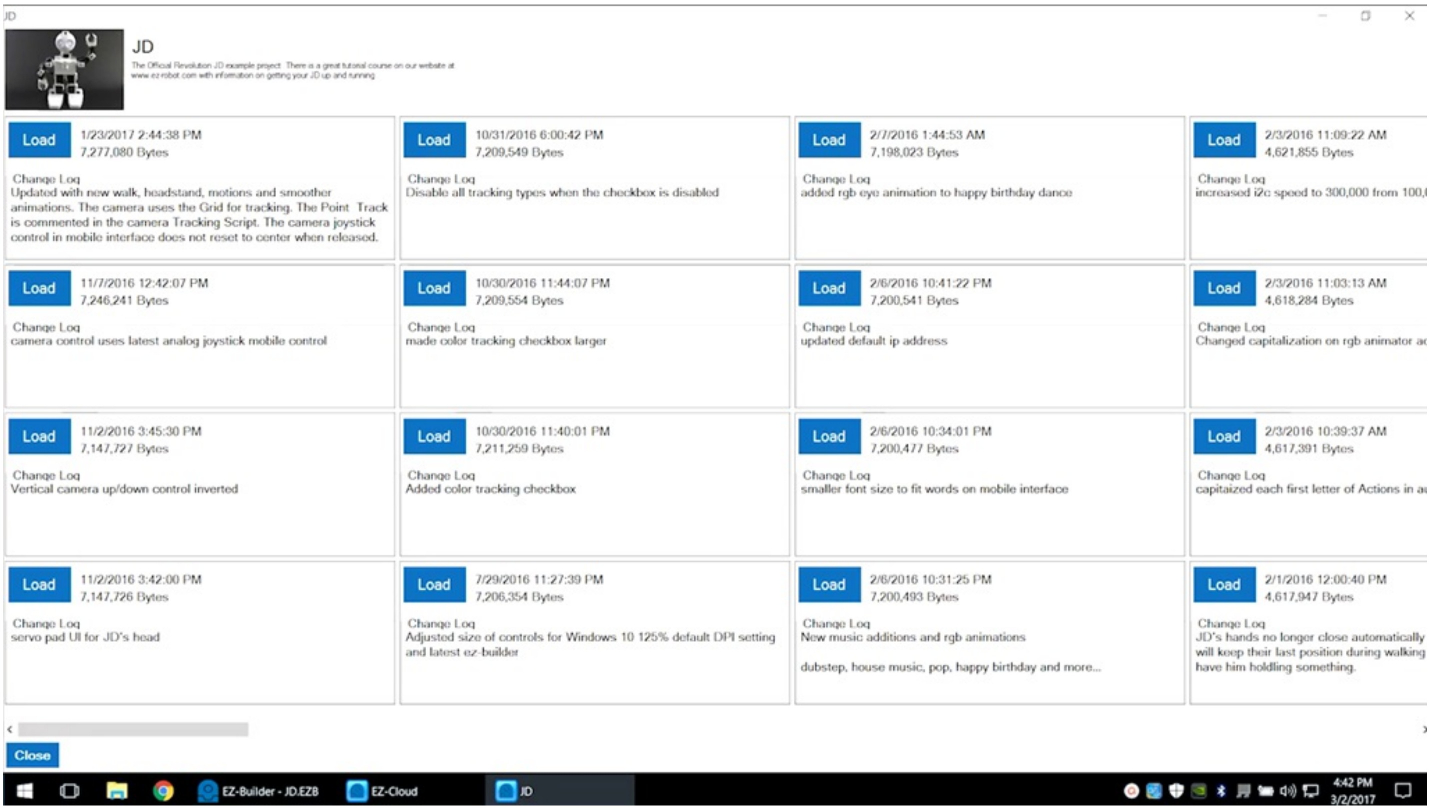
Default Control: Main

Description: The Official Revolution JD example project. There is a great tutorial course on our website at www.ez-robot.com with information on getting your JD up and running.

☑ Is Public

Load Revisions

Reload a previous version by selecting **File** -> **Open**. Click on the desired projectâ€™s **Version History** to see listed revisions.



JD

The Official Revolution JD example project. There is a great tutorial course on our website at www.ez-robot.com with information on getting your JD up and running

1/23/2017 2:44:38 PM
7,277,080 Bytes
Change Log
Updated with new walk, headstand, motions and smoother animations. The camera uses the Grid for tracking. The Point Track is commented in the camera Tracking Script. The camera joystick control in mobile interface does not reset to center when released.

11/7/2016 12:42:07 PM
7,246,241 Bytes
Change Log
camera control uses latest analog joystick mobile control

11/2/2016 3:45:30 PM
7,147,727 Bytes
Change Log
Vertical camera up/down control inverted

11/2/2016 3:42:00 PM
7,147,726 Bytes
Change Log
servo pad UI for JD's head

10/31/2016 6:00:42 PM
7,209,549 Bytes
Change Log
Disable all tracking types when the checkbox is disabled

10/30/2016 11:44:07 PM
7,209,554 Bytes
Change Log
made color tracking checkbox larger

10/30/2016 11:40:01 PM
7,211,259 Bytes
Change Log
Added color tracking checkbox

7/29/2016 11:27:39 PM
7,206,354 Bytes
Change Log
Adjusted size of controls for Windows 10 125% default DPI setting and latest ez-builder

2/7/2016 1:44:53 AM
7,198,023 Bytes
Change Log
added rgb eye animation to happy birthday dance

2/6/2016 10:41:22 PM
7,200,541 Bytes
Change Log
updated default ip address

2/6/2016 10:34:01 PM
7,200,477 Bytes
Change Log
smaller font size to fit words on mobile interface

2/6/2016 10:31:25 PM
7,200,493 Bytes
Change Log
New music additions and rgb animations

dubstep, house music, pop, happy birthday and more...

2/3/2016 11:09:22 AM
4,621,855 Bytes
Change Log
increased i2c speed to 300,000 from 100,

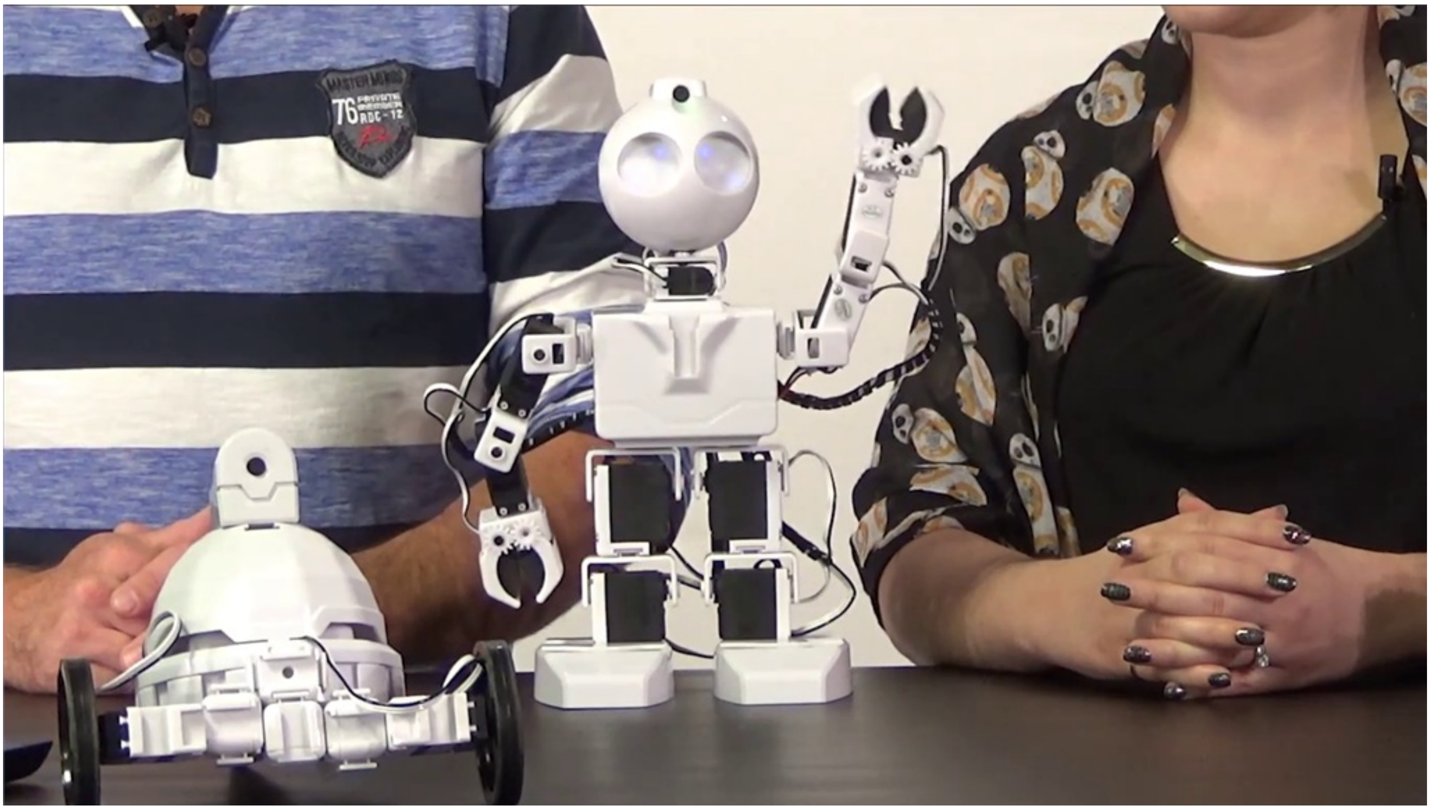
2/3/2016 11:03:13 AM
4,618,284 Bytes
Change Log
Changed capitalization on rgb animator ac

2/3/2016 10:39:37 AM
4,617,391 Bytes
Change Log
capitaized each first letter of Actions in a

2/1/2016 12:00:40 PM
4,617,947 Bytes
Change Log
JD's hands no longer close automatically will keep their last position during walking have him holdling something.

Learn More

Follow The Robot Program episodes to see all the features **EZ-Builder** has to offer.



Quiz

Question #1 Which workspace is designed for linear programming?

Question #2 Which workspace is designed for programming with logic, branches, and loops?

Question #3 What is the EZ-Robot scripting language called?

View the answers to this quiz at www.ez-robot.com/Tutorials/Lesson/20.

Visit www.TheRobotProgram.com for more episodes.