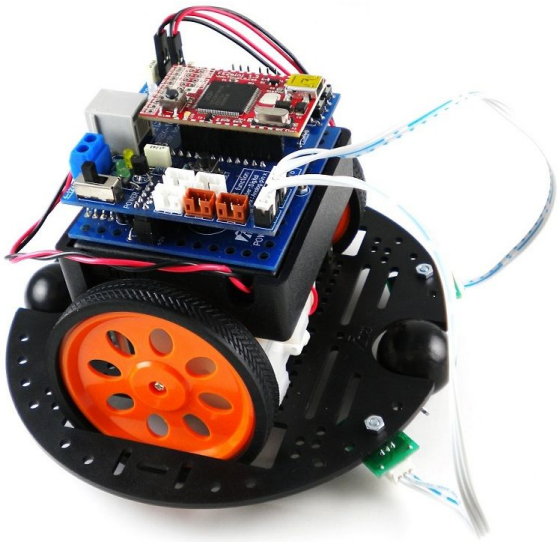
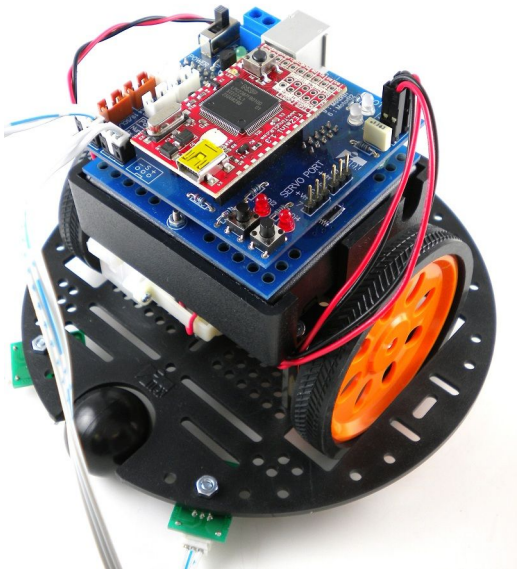
FEZ Robot Kit

Rev.1.00 March 21, 2011 step-by-step guide





Brought to you by

with the

Technology

www.ghielectronics.com

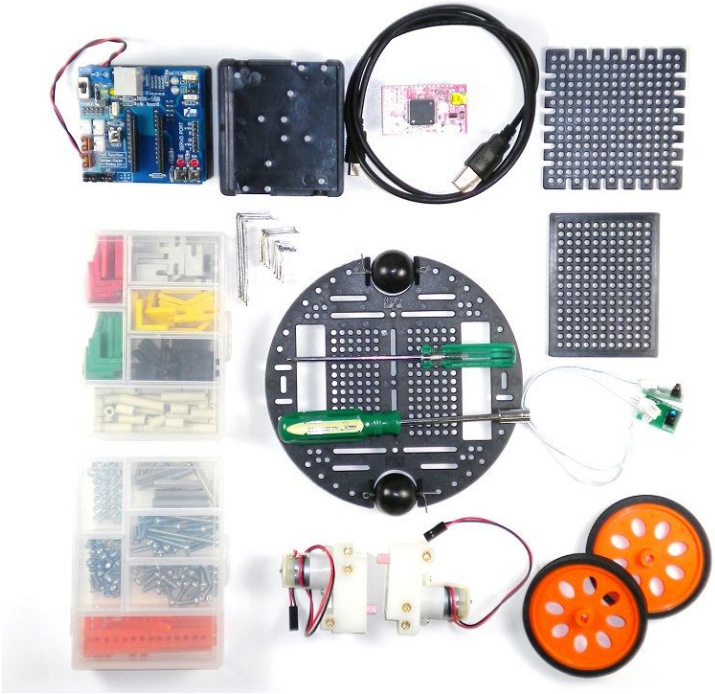
Table of Contents

Document Information

Information	Description
Abstract	This document provide information about FEZ Robot kit assembly with example code for robot control

1. Robot Kit Part List Summary

1. FEZ Mini board with Microsoft .NET Micro Framework with USB cable.
2. Robot controller board with 4-AA battery holder (Batteries are not included).
3. Reflective Sensor component with JST cable (2 sets).
4. 48:1 ratio 4.5V DC motor gearbox with IDC cable (2 sets).
5. Circle wheel and Threaded rubber wheel set with 2mm. tapped-screw. (2 sets)
6. 80x60 cm. and 80x80 cm. Plastic Grid plate set (2 sets)
7. Circle base with idle ball wheels
8. Plastic joiner and Strip joiner set (60 pieces of 3-type mix colored plastic joiner, 4 pieces of each 3/5/12 holes of Strip joiner)
9. Right-angled metal shaft set (4 pieces of each 1x2, 2x2, 2x5 Right-angled metal shaft)
10. Nuts and Screws set.



2. Parts Description

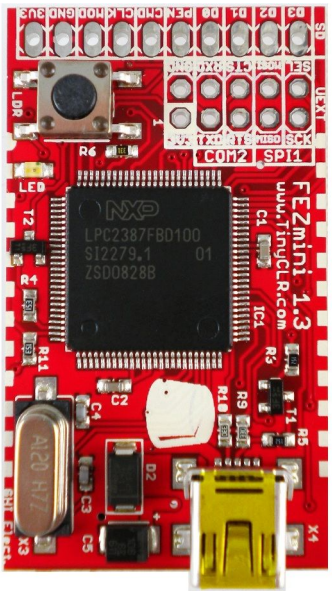
2.1. FEZ Mini

FEZ Mini is a tiny board running Microsoft .NET Micro Framework. This means, you can write code with much more efficiency using C# programming language under free Microsoft Visual C# express.

The pinout is a standard 0.1" header allowing FEZ Mini to plug directly into a bread-board.

Many libraries are already included like FAT file system, threading, UART, SPI, I2C, GPIO, PWM, ADC, DAC and many more.

To get started with FEZ please take a look at *FEZ Tutorial* and *Beginners Guide to .NET Micro Framework* eBook.

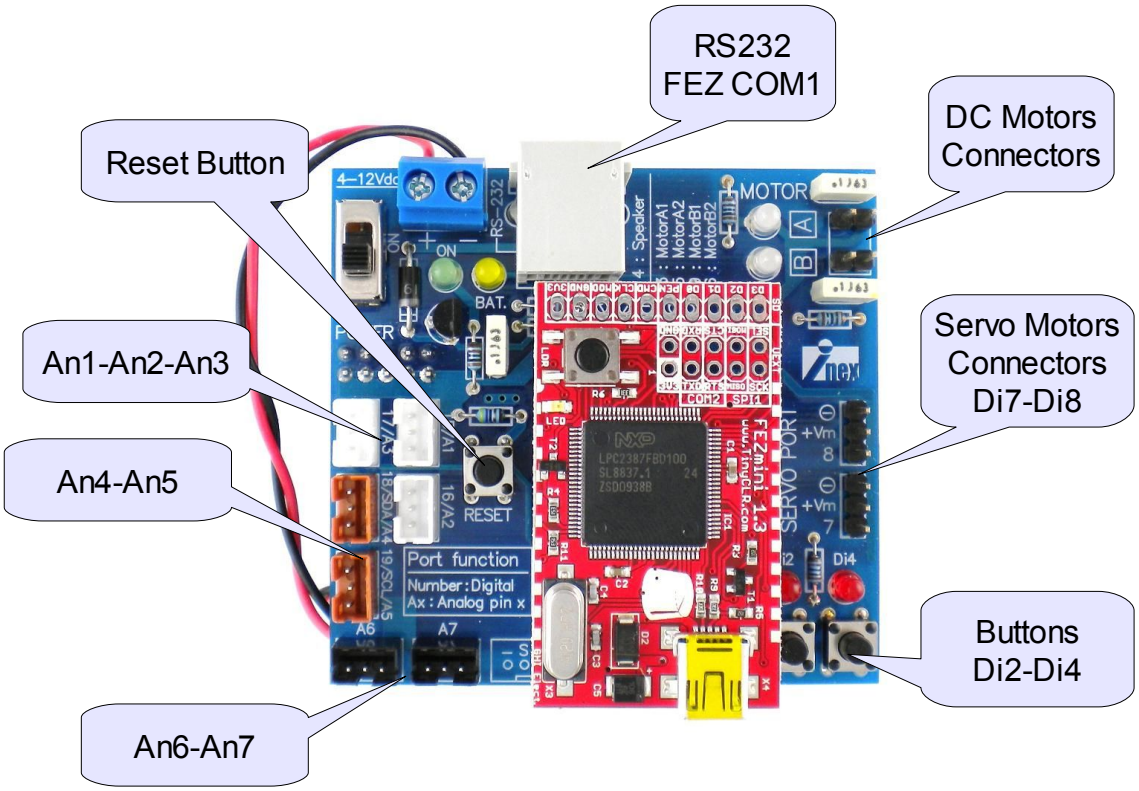


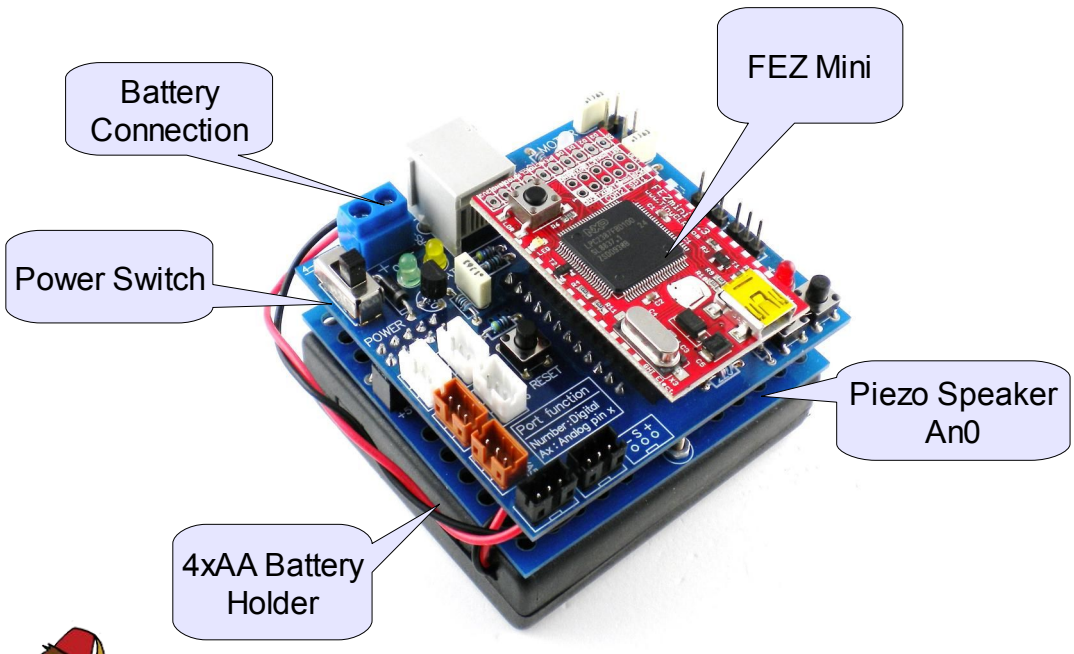
2.2. Robot controller board

The Robot controller board is a complete, low-cost development platform designed for those interested in learning and using FEZ Mini module in robotic applications. Its compact size, convenient features, and low price make it an ideal tool for students and educators. The summarized technical feature of the board is as follows :

- Screw terminal block for battery connections, operational +4.8 to +12V.
- +5V switching regulator power supply circuit.
- 2 push-button switches, with indicator LEDs, connected to Di2 and Di4.
- 5 IOs with standard JST3AA sockets, An1 to An7.
- RS-232 serial port interface.
- 2-Channel DC motor driver with LED indicators, Motor A: Di3-Di5, Motor B: Di6-D9.

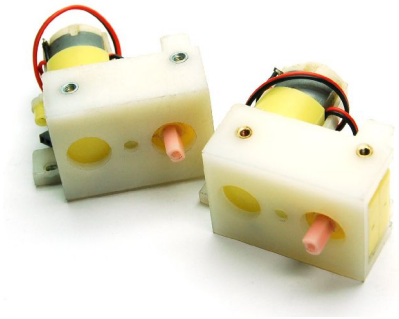
- 2-Servo motor output, connected to Di7 and Di8.
- Piezo speaker, at bottom circuit board, connected to An0.





2.3. DC motor gearbox

This robot kit included 48:1 ratio DC motor gearbox; model BO-2 with IDC connector cable. This is features :



- Operating voltage is +3 to +9Vdc
- Current consumption 130mA @ +6Vdc and no load)
- Average speed 170 to 250 round per minute (RPM) @ +6V and no load
- Weight is 30 grams
- Minimum torque is 0.5 kg/cm.
- Attached with the plastic mounting with 5 of insert nuts
- 42 x 45 x 22.7 mm. (WxLxH) dimension

2.4. Reflective sensor Component

This sensor emits an invisible infrared light and then tries to measure it back to see if the light is "reflecting". This can be used to detect a black line on a white sheet of paper (help guide the robot) or to detect table edges (avoid falling off). This component simply changes its output state from high to low when there is reflection.

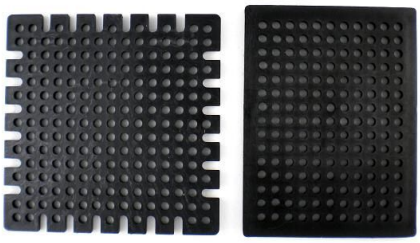
2.5. Mechanical Parts

Circle wheel and Tire set

Includes a pair of wheels for the DC motor gearbox with rubber tires. Fix the wheel with gearbox shaft by the included 2mm self-tapping screws.

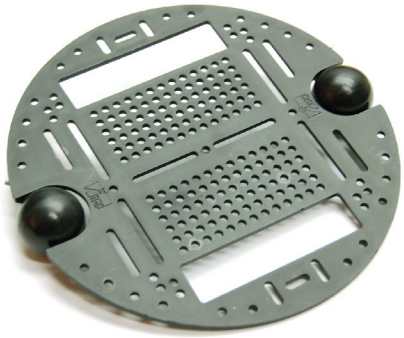
Plastic grid plate set

Includes each of the universal plastic grid plate 2 sizes; 80x60mm and 80x80mm. Each plate provides 3mm diameter holes with 5mm pitch.



Circle base

This base is injected from high quality ABS plastic. It has 2 free ball wheels at both side. This base has many 3mm holes to fix the controller board, sensors and more mechanical parts.



Plastic joiners

60 pieces of varied color joiners made from PVC plastic. They can be connected together or by using screws and 3 mm nuts in installation. There are 4 types; Right angle, Obtuse, Straight joiner and Hole straight joiner.

Strip joiners

Plastic joints with 3mm holes 5mm pitches. Usually used when extra length is required to attach sensor away from the robot for instance. Comes in 4 pieces of 3 sizes; 3, 5 and 12 holes type. Total 12 pieces.

Box holder

Injected plastic box for supporting the controller board with battery holder. It has some 3mm hole to fix it with the robot skeleton.

Right-angle metal shaft

7.5mm-width right-angle metal shaft. Each shaft has 3mm holes. The set includes 4 pieces of 1x2, 2x2 and 2x5 holes metal shaft.

Screw and Nut set

Includes 2 of 2mm self-tapping screws, 4 of 3x8mm M3 screws, 30 of 3x10mm M3 screws, 4 of 3x15 mm M3 screws, 4 of 3x40mm M3 screws, 10 of 3x8mm flat-head screws and 30 of 3mm. M3 nuts.

Metal spacer

Supporting metal parts for plate or sensor components. Includes 6 of 33mm. metal hexagonal spacers. Each standoff has 3mm thread through-hole.

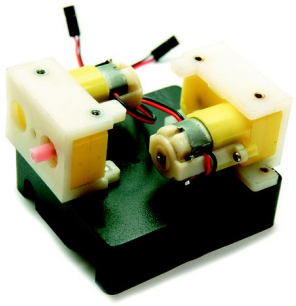
Plastic spacer

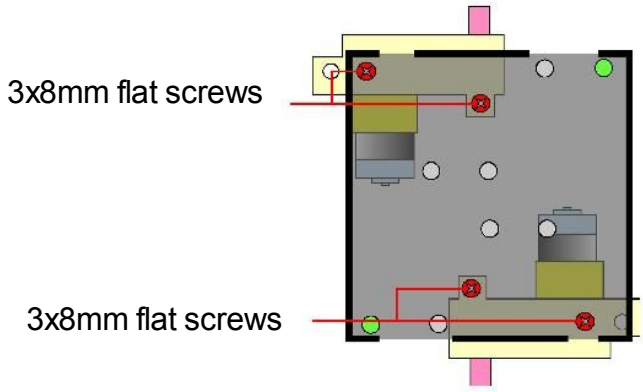
They are some mechanical parts for supporting the plate and sensor board. This kit includes 4 pieces set of plastic spacer (3mm., 10mm., 15mm. and 25mm.) 4 sets

3. Building The Robot

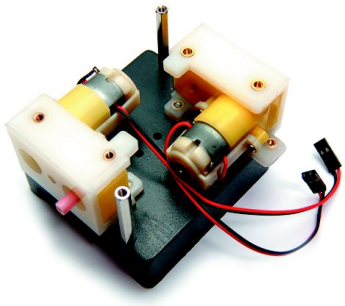
3.1. Building procedure

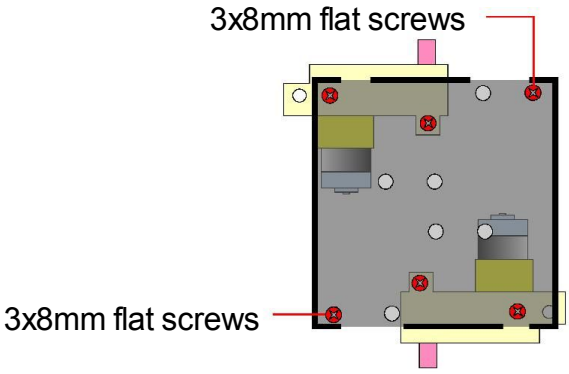
Attach both DC motor gearboxes with the Box holder by 3x8mm flat screws.



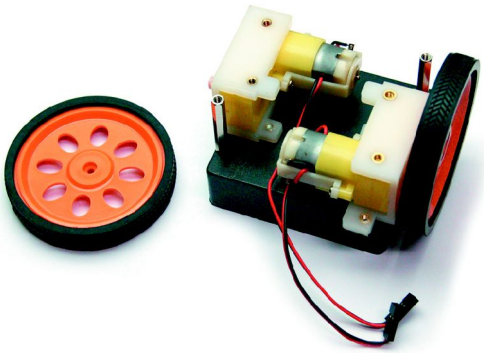


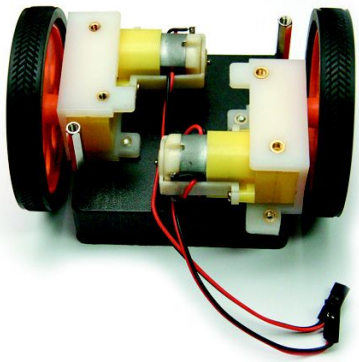
Attach 2 of 33mm metal spacers with the Box holder by by 3x8mm flat screws at the position following the pictures below.



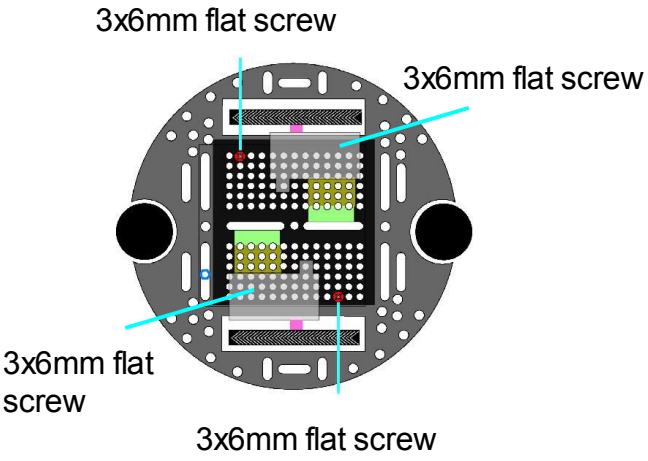


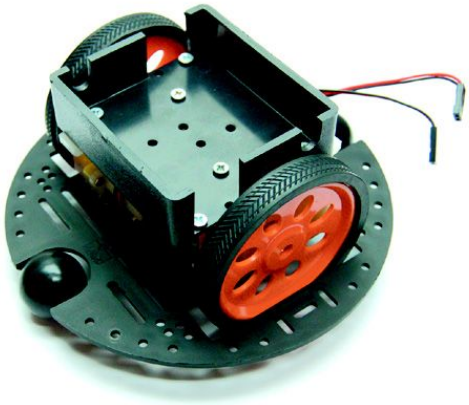
Attach the wheel with tire to DC motor gear boxes shaft and fix with 2mm self-tapping screws.



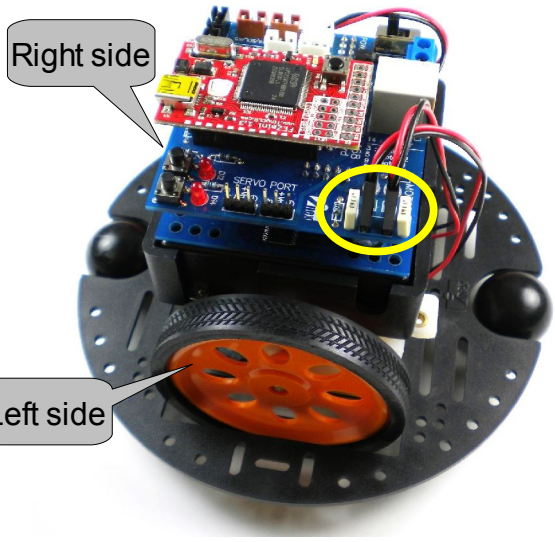


Attach the gearbox structure as shown with the circle base by using 3x6mm screws at the position following the pictures below. See the wheels position is the center of base.





Install Robot Controller board into the Box holder. Connect DC motors cables. Start with Left motor cable to MOTOR A output, Right motor cable to MOTOR B output. don't worry about the direction of cable connection since you will check that anyway when we try to move the robot to let motor run in the correct direction that matches the code description.



4. Controlling The Robot

4.1. Generating tones using on-board piezo speaker

Robot Control board includes a piezo speaker connected to pin An0 to FEZ Mini. To control this piezo, simply download **FEZMini_Robot.cs** driver source code file and add it to Visual C# project to use the example below.

FEZ tutorial Document shows how to create projects and add components drivers.

(Both files are available on www.tinyclr.com).

Code snippet:

```
using System;
using System.Threading;
using Microsoft.SPOT;
using GHIElectronics.NETMF.FEZ;

public class Program
{
    public static void Main()
    {
        //make a beep for 1sec
        FEZMini_Robot.Beep(7000, 1000);
        Thread.Sleep(1000);

        // make more noise!
        while (true)
        {
            for (int i = 3000; i < 8000; i += 100)
                FEZMini_Robot.Beep(i, 50);

            Thread.Sleep(1000);
        }
    }
}
```

Note: Piezo Speakers usually require PWM, but since An0 does not support PWM we made software PWM in this driver to resonate the piezo. Software PWM can slow down the system. An advice is to use this piezo only when necessary. there are separate seizor components on tinyclr.com that user can get and use with pin with hardware PWM feature

on this robot.

4.2. Moving the robot

The Robot Control board has PWM-based H-Bridge driver circuit gives FEZ Mini the ability to control the 2 DC motors (speed, direction break ...etc).

Every Channel (motor control) requires 2 signals, speed control signal acquired by PWM signal (Pulse Width Modulation) to control rotation speed and direction control signal, acquired by digital signal, to control the direction of the motor rotation.

Left channel control:

Speed control (PWM) connected to Di5, Direction signal is connected to Di3.

Right channel control:

Speed control (PWM) connected to Di6, Direction signal is connected to Di9.

We provide off-the-shelf source code driver **(FEZMini_Robot.cs)** for easy control of our Robot 's DC motors through the control board.

Simply download **FEZMini_Robot.cs** driver source code file and add it to Visual C# project to use the example below.

FEZ tutorial Document shows how to create projects and add components drivers. (Both files are available on www.tinyclr.com).

In the next few sections, we will try the methods provided by this driver to move the robot.

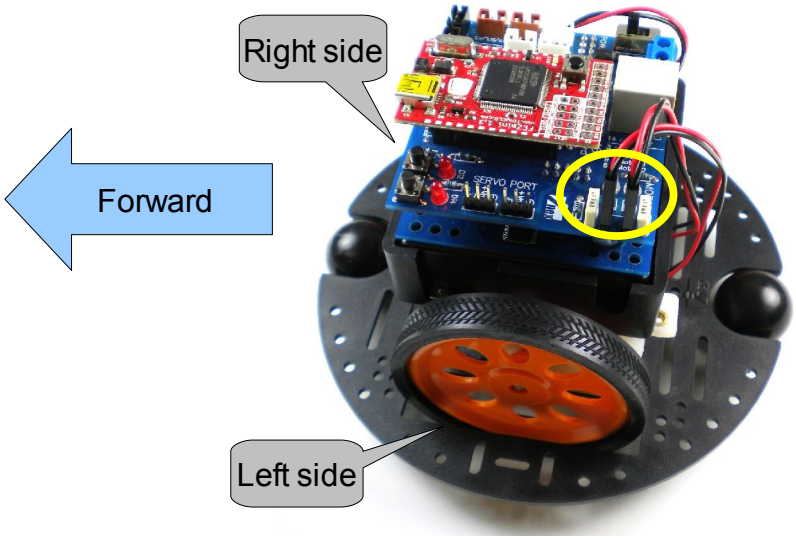
Forward and Backward

Code snippet: (Move Forward)

```
using System;
using System.Threading;
using Microsoft.SPOT;
using GHIElectronics.NETMF.FEZ;

public class Program
{
   public static void Main()
   {
      //make a beep for 1sec
      FEZMini_Robot.Beep(7000, 1000);
      Thread.Sleep(1000);

      //move forwards
```

```
        FEZMini_Robot.Move(50, 50);

        Thread.Sleep(Timeout.Infinite);
    }
}
```

Important Note: Try this example on your robot and see if robot move forward, if not, you may need to reverse the motor cables.



Code snippet: (Forward...Stop...Backward and repeat)

```
using System.Threading;
using GHIElectronics.NETMF.FEZ;

public class Program
{
    public static void Main()
    {
        //make a beep for 1sec
        FEZMini_Robot.Beep(7000, 1000);
        Thread.Sleep(1000);

        while (true)
        {
            // move forwards
            FEZMini_Robot.Move(70, 70);
            Thread.Sleep(500);
```

```
            // stop
            FEZMini_Robot.Move(0, 0);
            Thread.Sleep(500);

            // move backwards
            FEZMini_Robot.Move(-70, -70);
            Thread.Sleep(500);
        }
    }
}
```

Ramping Speed

Sudden fast movement generate a power surge that can cause the robot to reset. This is like pushing the gas peddle to the ground in your car. For that, we need to "ramp" the movement up or down. Ramping down is not required but ramping up is a must in many cases.

Code snippet:

```
using System.Threading;
using GHIElectronics.NETMF.FEZ;

public class Program
{
    public static void Main()
    {
        while (true)
        {
            //stop
            FEZMini_Robot.Move(0, 0);

            //make a beep for 1sec
            FEZMini_Robot.Beep(7000, 1000);
            Thread.Sleep(1000);

            // move forwards with full speed but with ramp up
            FEZMini_Robot.MoveRamp(100, 100, 10);
            Thread.Sleep(2000);

            // emergency stop do nto need ramp
            FEZMini_Robot.Move(0, 0);
            Thread.Sleep(2000);

            // ramp back to 100% but very slowley
            FEZMini_Robot.MoveRamp(100, 100, 100);
            Thread.Sleep(2000);
        }
    }
```

Making turns

The robot turn in the same way tank does. There is no front wheel that turn, instead the right motor goes backwards while the left motor goes forward to make a right turn.

Code snippet:

```
using System.Threading;
using GHIElectronics.NETMF.FEZ;

public class Program
{
   public static void Main()
   {
      while (true)
      {
         //make a beep for 1sec
         FEZMini_Robot.Beep(7000, 1000);
         Thread.Sleep(1000);

         // turn left
         FEZMini_Robot.MoveRamp(-100, 100, 150);
         Thread.Sleep(5000);

         //stop
         FEZMini_Robot.Move(0, 0);
         Thread.Sleep(1000);

         // turn right
         FEZMini_Robot.MoveRamp(100, -100, 150);
         Thread.Sleep(5000);

         //stop
         FEZMini_Robot.Move(0, 0);

      }
   }
}
```

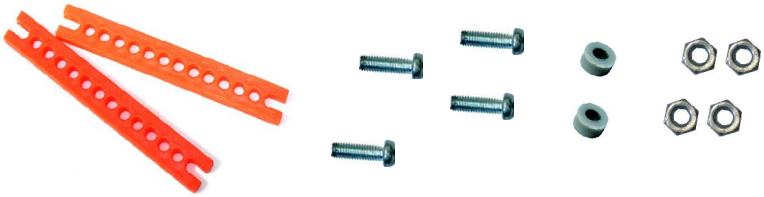
5. Table Edge Detection Example

Moving the robot was fun and simple but what about doing some more interesting stuff. We will make the robot smart enough to move on a table without falling off. This requires us to detect the edge of the table.

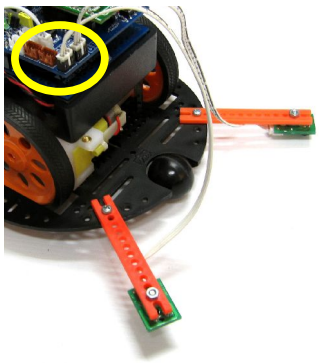
The included reflective sensor components are ideal for this task. This sensor emits an invisible infrared light and then tries to measure it back to see if the light is "reflected". This can be used to detect a black line on a white sheet of paper to help guide a robot. This component simply outputs different voltage level according to the returned reflection density.

Let's first attach the reflective sensor components, then we will explain how they would help finding the edge more.

Additional part list



Assembly



Connect right reflective sensor cable to An7 and the left one to An6.

the Idea

Now, while the sensor is facing the smooth table, the majority of the emitted infrared beam will be reflected back to the sensor even partially (supportingly more than 5% of the beam) thus it can tell the robot that the it is still on the table and it can move ahead. Then when any of the sensor becomes our of the table after the edge, the emitted beam will not be reflected on anything back to the sensor (supportingly more than 5% of the beam) thus it can tell the robot to stop and move back rotate and go in different direction or it will fall.

User might need to calibrate the reflection threshold in the code according to the environment (table color, the brighter the better the reflection)

Let's try to apply that in code

Create a new .NET Micro Framework project and added the required assemblies.

Add **FEZ_Components_ReflectiveSensor.cs** to Visual C# project to use the example below. This driver file include very simple methods to read any attached reflective sensor component.

Add **FEZMini_Robot.cs** driver source code file to Visual C# project to control the robot.

FEZ tutorial Document shows how to create projects and add components drivers. (All files are available on www.tinyclr.com).

Now the project is ready to add some code accomplish what we explained before

Code snippet:

```
using System.Threading;
using GHIElectronics.NETMF.FEZ;

public class Program
{
   public static void Main()
   {
      FEZ_Components.ReflectiveSensor LeftSensor =
         new FEZ_Components.ReflectiveSensor(FEZ_Pin.AnalogIn.An7, 1);
      FEZ_Components.ReflectiveSensor RightSensor =
         new FEZ_Components.ReflectiveSensor(FEZ_Pin.AnalogIn.An6, 1);

      // move forwards
      FEZMini_Robot.MoveRamp(70, 70, 10);
      while (true)
      {
         if (LeftSensor.GetState() ==
FEZ_Components.ReflectiveSensor.DetectingState.ReflectionNotDetected
            || RightSensor.GetState() ==
FEZ_Components.ReflectiveSensor.DetectingState.ReflectionNotDetected)
         {
            // no reflection on one of the sensors so we will back up and turn
            //quick stop
            FEZMini_Robot.Move(0, 0);
            // backup
            FEZMini_Robot.MoveRamp(-70, -70, 10);
            Thread.Sleep(50);
            //turn
            FEZMini_Robot.Move(-70, 70);
            Thread.Sleep(200);
            //quick stop
            FEZMini_Robot.Move(0, 0);
            //make a beep for 1sec
            FEZMini_Robot.Beep(7000, 1000);
            // move forwards
            FEZMini_Robot.MoveRamp(70, 70, 10);
         }
         Thread.Sleep(10);
      }
   }
}
```

6. What Is Next?

GHI Electronics supplies many components and extensions that can be used with this robot kit, Distance Detector, touch buttons, LCD , Easy Remote, GPS , accelerometer, Servo Motors and more...

... Be Creative.