



Support

[.NET Micro Framework](#)
[.NET Gadgeteer](#)
[File System](#)
[Documents](#)
[Consulting](#)
[Glide](#)

Analog Inputs

Contents

1. [Required](#)
2. [Introduction](#)
3. [Example 1: Light Sensor](#)

Required

Tutorials: [First NETMF Project](#), [Digital Inputs](#)

Introduction

Digital input pins can only read high and low (one or zero) but analog pins can read the voltage level. There are limitations on voltages that can be applied to analog inputs. For example, FEZ analog input can read voltages anywhere between zero and 3.3V. When the pins are digital they are tolerant of 5V but when the same pin is set to analog only up to 3.3V can be used. This limitation on analog inputs is not a big issue because most analog signals are usually conditioned to work with the analog inputs. A voltage divider or an op-amp circuit can be used to get a fraction of the actual signal, and then scale the signal.

Example 1: Light Sensor

To show Analog input, we will use the E-Block Light Sensor (which will show data due to varying voltage resistance) connected to an Analog pin via a JST connector. With the E-Block Light Sensor, the higher the number, the darker the light is. This example is using JST An3 and the light levels will be displayed every half second. The values returned from the input can be scaled. Instead of the default range of 0 - 3.3, you can scale it and add an offset as you need.

Remember to use the analog pin enumeration to determine what pins can be used as analog inputs on your device.

```
1. using System.Threading;
2. using Microsoft.SPOT;
3. using Microsoft.SPOT.Hardware;
4.
5. namespace change_this_to_your_namespace
6. {
7.     public class Program
8.     {
9.         public static void Main()
10.        {
11.            AnalogInput lightSensor = new AnalogInput((Cpu.AnalogChannel)Cpu.AnalogChannel.ANALOG_7);
12.            double lightSensorReading = 0;
13.
14.            while (true)
15.            {
16.                lightSensorReading = lightSensor.Read();
17.                Debug.Print(lightSensorReading.ToString());
18.                Thread.Sleep(500);
19.            }
20.        }
21.    }
22. }
```

lightSensor.SetLinearScale(0, 100); sets the minimum and maximum sample rate. If this were the E-Block Thermometer, it could be set up with a range of 0 ~ 32 degrees Celsius.

lightSensor.Read(); retrieves the current light reading. Remember, the larger the number, the darker the light is.

