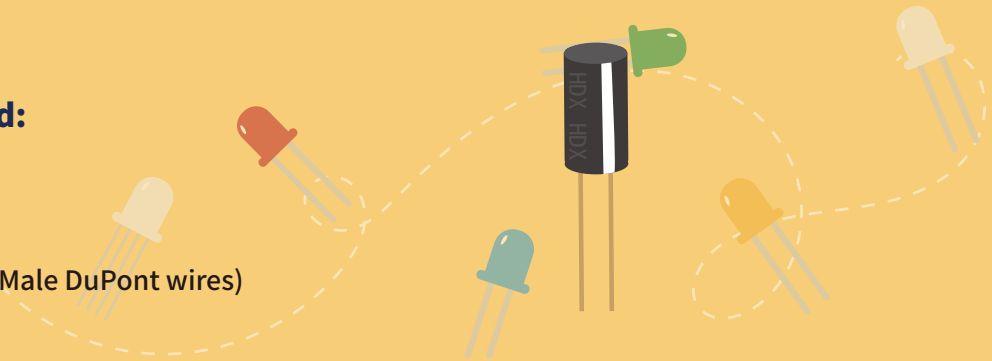# lesson

7

Tilt Ball Switch

Overview

In this lesson, you will learn how to use a tilt ball switch in order to detect a small angle of inclination.
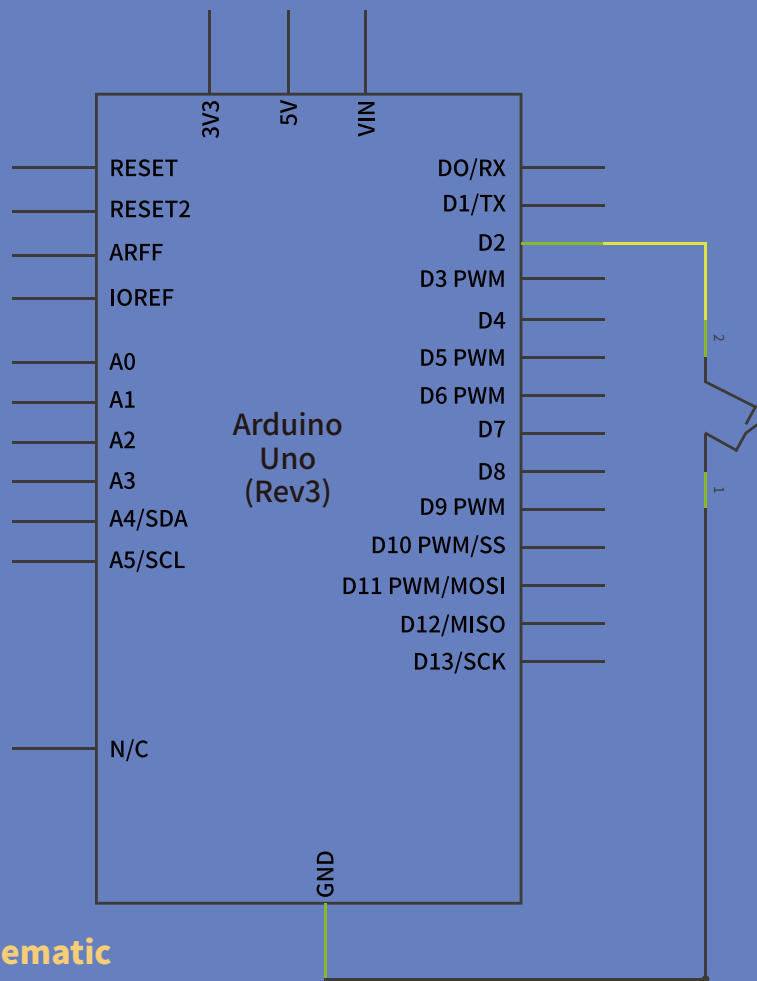
Component Required:

(1) x Elegoo Uno R3

(1) x Tilt Ball switch

(2) x F-M wires (Female to Male DuPont wires)
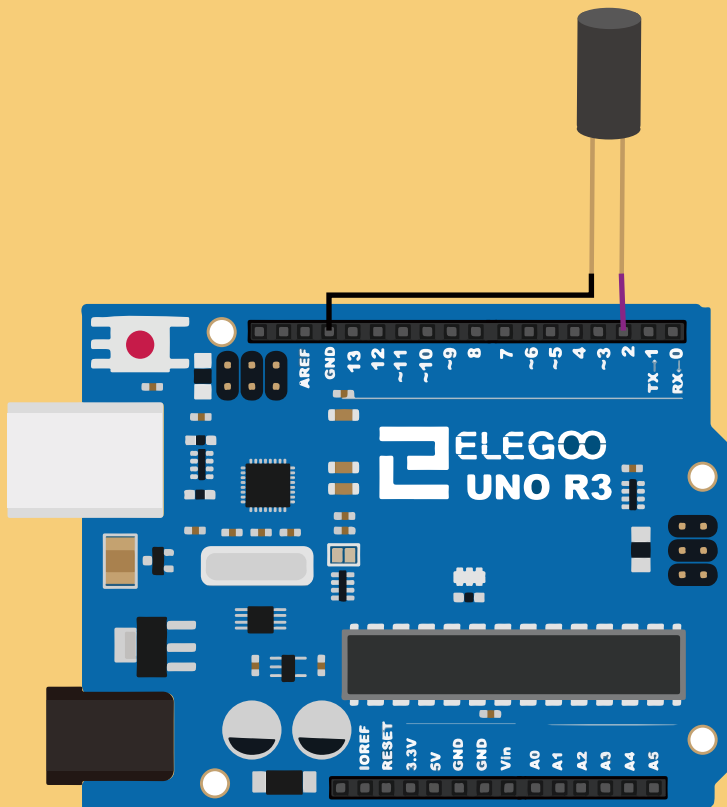
Component Introduction

Tilt sensor:

Tilt sensors (tilt ball switch) allow you to detect orientation or inclination. They are small, inexpensive, low-power and easy-to-use. If used properly, they will not wear out. Their simplicity makes them popular for toys, gadgets and appliances. Sometimes, they are referred to as "mercury switches", "tilt switches" or "rolling ball sensors" for obvious reasons.

They are usually made up of a cavity of some sort (cylindrical is popular, although not always) with a conductive free mass inside, such as a blob of mercury or rolling ball. One end of the cavity has two conductive elements (poles). When the sensor is oriented so that that end is downwards, the mass rolls onto the poles and shorts them, acting as a switch throw.

While not as precise or flexible as a full accelerometer, tilt switches can detect motion or orientation. Another benefit is that the big ones can switch power on their own. Accelerometers, on the other hand, output digital or analog voltage that must then be analyzed using extra circuitry.

Connection Schematic

3V3 5V VIN

RESET
RESET2
ARFF
IOREF

A0
A1
A2
A3
A4/SDA
A5/SCL

Arduino
Uno
(Rev3)

N/C

GND

DO/RX
D1/TX
D2
D3 PWM
D4
D5 PWM
D6 PWM
D7
D8
D9 PWM
D10 PWM/SS
D11 PWM/MOSI
D12/MISO
D13/SCK

2

1

Wiring diagram

ELEGOO
UNO R3

AREF GND 13 12 ~11 ~10 ~9 8   7 ~6 ~5 4 ~3 2 TX►1 RX►0

IOREF RESET 3.3V 5V GND GND Vin A0 A1 A2 A3 A4 A5

Code

After wiring, please open the program in the Folder **Ball_Switch** where the course is located and click UPLOAD to upload the program. See Lesson 5 in part 1 for details about program uploading if there are any errors.

```
const int ledPin = 13;
```

const

[Variable Scope & Qualifiers]
Description

The const keyword stands for constant. It is a variable qualifier that modifies the behavior of the variable, making a variable "read-only". This means that the variable can be used just as any other variable of its type, but its value cannot be changed. You will get a compiler error if you try to assign a value to a const variable.

Constants defined with the const keyword obey the rules of variable scoping that govern other variables. This, and the pitfalls of using #define, makes the const keyword a superior method for defining constants and is preferred over using #define.

Example Code

```
const float pi = 3.14;
float x;
// ....
x = pi * 2; // it's fine to use consts in math
pi = 7;    // illegal - you can't write to (modify) a constant
```

Notes and Warnings
#define or const

You can use either const or #define for creating numeric or string constants. For arrays, you will need to use const.In general const is preferred over #define for defining constants.