DualEncoders

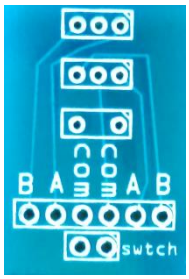
TUTORIAL

WHAT IS DualEncoders?

The DualEncoders Library allows you to use dual concentric rotary encoders with 2 steps per detent, like those sold by Propwash.

HOW DO WE WIRE THE ENCODERS?

Each encoder has 3 pins: A, B and Com. Pins A and B are connected to a digital pin on the Arduino, and the common (Com) pin is connected to ground. One of the switch pin is connected to a digital pin on the Arduino and the other one is connected to ground. (It does not matter which one is connected to what.)




```
//============================
// Encoder      Arduino
// outerB-------3
// outerA-------2
// Com--+---------+--Gnd
// Com--+         |
// innerA-------4 |
// innerB-------5 |
// Switch-------6 |
// Switch---------+
//----------------------------
```

HOW DOES IT WORK?

First, we have to include the Library in the sketch (before the setup() section).

```
#include <DualEncoders.h>
```

Then, we tell the DualEncoders Library which Arduino pins we will use:

```
DualEncoders anEncoder(2, 3, 4, 5, 6) ;
```

In the setup() section of the sketch, we include the following line that initialises for us the four encoders pins and the switches' pin. We don't have to write: `pinMode(x, INPUT_PULLUP)`.

```
anEncoder.begin();
```

The encoders are designated as "inner" and "outer". The Library's methods will return in which direction each encoder was turned :CW or CCW. It will return 0 (zero) if it has not turned or reached a detent (click).

```
int direction : anEncoder.inner();[1]
int direction : anEncoder.outer();
```

The switch is read with the following line. If the switch is pressed, it will wait until it is released and return 0. If the switch was not pressed, it will return 1.

```
byte aSwitch = anEncoder.readSwitch();
```

[1] The variable has to be of type int. The values can be negative (CW=1 and CCW=-1)

EXAMPLE SKETCH

```
#include <DualEncoder.h>

DualEncoder radioCOM1(2, 3, 4, 5, 6);

//setup=========================================================
void setup() {
  Serial.begin (9600);            //To display on the monitor
  radioCOM1.begin();              //To initialise the encoders
  Serial.println("starting”);
}//setup----------------------------------------------------------

//loop============================================================================
void loop(){
  int stat;
  stat = radioCOM1.readInner();
  if (stat == CWW) Serial.println("sim/radios/stby_com1_coarse_down");
  if (stat ==  CW) Serial.println("sim/radios/stby_com1_coarse_up");
  stat = radioCOM1. readOuter();
  if (stat == CCW) Serial.println("sim/radios/stby_com1_fine_down");
  if (stat ==  CW) Serial.println("sim/radios/stby_com1_fine_up");
  if (radioCOM1.readSwitch() == 0) Serial.println("sim/radios/com1_standy_flip");
}//loop-----------------------------------------------------------------------------
```

This is version 1.1: Added a very fast Debounce and EMF filter.
Code was rewritten entirely.

I sincerely hope that this DualEncoder Library will help you in your projects.
Jacques Bellavance