```cpp
// RPMonlyV2
#include <math.h>
#include<SoftwareSerial.h>
#include<EEPROM.h>

#define RPMTxPin 4
#define commonRxPin 8
#define rpmIntPin 7
#define alpha 0.2

// SparkFun serial 7 segments display
SoftwareSerialsDispRPM(commonRxPin,RPMTxPin);

int compticRPM = 0;
long ldrpm = 0;
float rpm = 0.0;
long timerRPM = 0;
int nbtrs = 4;    // Nombre de tours pour échantillonnage
int nbtic = 128; // Nombre d'impulsions par tour
long pow2Tab[4] = {1, 256, 65536, 16777216}; // Power of 2 array

float pfRPM = 0.0;     // Previous filtered RPM value
float lfRPM = 0.0;     // Last filtered RPM value

void setup()
{
 attachInterrupt(digitalPinToInterrupt(rpmIntPin),intRPM,FALLING);

 Serial.begin(9600);
 delay(10);

 // Resetting serial display
 long baudRates[14] = {300, 600, 1200, 2400, 4800, 9600, 14400,
19200, 28800, 38400,
                      57600, 76800, 115200, 153600
                      };

 for (int i = 0; i < 14; i++)
 {
  sDispRPM.begin(baudRates[i]);
   delay(10);
   sDispRPM.write(0x81);
```

```
    delay(10);
    sDispRPM.write(0x76);
    delay(10);
  }

 sDispRPM.begin(9600);
 delay(10);
 Serial1.begin(9600);
 delay(10);

 char sInit[5]  = "8888";
 char sClear[5] = "    ";
 dispText( &sDispRPM, sClear, 0 );
 dispText( &sDispRPM, sInit, 0b00111111 );  // Light on all
 delay(1000);
 dispText( &sDispRPM, sClear, 0 );
 timerRPM = 0;
 }

void loop()
{
  // Display RPM
  if ( millis() - timerRPM > 250 )
  {
    timerRPM = millis();

    if ( millis() - ldrpm > 5000 )
     {
       lfRPM = 0;
     }

     char sRpm[5];
    sprintf(sRpm, "%4d", int(lfRPM+0.5));
     dispText( &sDispRPM, sRpm, 0 );
   }
}

void dispText(SoftwareSerial  * afficheur, char *text, char dots)
{
 afficheur->write(0x76);
 afficheur->print(text);
 afficheur->write(0x77);
```

```
 afficheur->print(dots);
}


void intRPM()
{
  if (compticRPM >= nbtrs * nbtic )
   {
     rpm = nbtrs * 60000 / (millis() - ldrpm);
     lfRPM = alpha*pfRPM + (1.0-alpha)*rpm;  // Filtering RPM value
     pfRPM = lfRPM;
     compticRPM = 0;
     ldrpm = millis();
   }
  compticRPM++;
}
```