

```
#include "Arduino.h"
#include "Voltmeter.h"
```

```
Public:
int realVoltage;
int calibrated;
void get();
float getValue();
float getAverage();
float getVoltage();
bool getReady();

private:
int _sensor;
int _freq;
float _average;
float _voltage;
int _R;
int _debounce;
unsigned long _tMax;
unsigned long _tActu;
float _maxVal;
float _minVal;
// Movil Average
float _Y = 0.00;
float _alpha = 0.86;
float _S = _Y;
```

```
Voltmeter
(const int sensorPin, const int freq)
```

```
_sensor = sensorPin;
_freq = freq;
_debounce = 0;
```

```
void get()
```

```
_maxVal = 0.00;
_minVal = 1024.00;
_tActu = (unsigned long)millis();
_tMax = _tActu + _freq;
```

```
_tActu = _tActu;
_tMax > _tActu
```

```
_Y = (float)analogRead(_sensor)
```

```
_S = (_alpha*_Y)+((1-_alpha)*_S)
```

```
_S > _maxVal
```

```
_maxVal = _S
```

```
_S < _minVal
```

```
_minVal = _S
```

```
_average = (_maxVal - _minVal) / 2 + _minVal;
```

```
float getVoltage()
```

```
_R = (_maxVal - _average)*10
```

```
_voltage = (float)_R/10
```

```
return _voltage
```

```
bool getReady()
```

```
getVoltage()
```

```
_voltage > 190 && _voltage < 240
```

```
_debounce = 0
```

```
return false
```

```
_debounce < 3
```

```
return true
```

```
_debounce++
```

```
return false
```

```
float getValue()
```

```
return _maxVal
```

```
float getAverage()
```

```
return _average
```