

Electricity Sensor (AC - 5A)

The Electricity sensor module is based on the TA12-200 current transformer which can change large alternating current into a small amplitude. You can use it to test large alternating current up to 5A.

Specifications

Transformation ratio: 2000:1
Max Input Current: 5A (AC)
MAX Output Current: 2.5 mA
Working Frequency: 20K

Arduino Code

```
// Function: Measure the amplitude current of the alternating
current and
//           the effective current of the sinusoidal
alternating current.
// Hardware: Grove - Electricity Sensor
// Date:     Jan 19,2013

#define ELECTRICITY_SENSOR A0 // Analog input pin that sensor is
attached to

float amplitude_current;           //amplitude current
float effective_value;             //effective current

void setup()
{
  Serial.begin(9600);
  pins_init();
}
void loop()
{
  int sensor_max;
```

```

    sensor_max = getMaxValue();
    Serial.print("sensor_max = ");
    Serial.println(sensor_max);
    //the VCC on the Grove interface of the sensor is 5v
    amplitude_current=(float)sensor_max/1024*5/800*2000000;
    effective_value=amplitude_current/1.414;//
minimum_current=1/1024*5/800*2000000/1.414=8.6(mA)
                                //Only for sinusoidal alternating
current
    Serial.println("The amplitude of the current is(in mA)");
    Serial.println(amplitude_current,1);//Only one number after
the decimal point
    Serial.println("The effective value of the current is(in
mA)");
    Serial.println(effective_value,1);
}
void pins_init()
{
    pinMode(ELECTRICITY_SENSOR, INPUT);
}
/*Function: Sample for 1000ms and get the maximum value from the
SIG pin*/
int getMaxValue()
{
    int sensorValue;           //value read from the sensor
    int sensorMax = 0;
    uint32_t start_time = millis();
    while((millis()-start_time) < 1000)//sample for 1000ms
    {
        sensorValue = analogRead(ELECTRICITY_SENSOR);
        if (sensorValue > sensorMax)
        {
            /*record the maximum sensor value*/
            sensorMax = sensorValue;
        }
    }
    return sensorMax;
}

```