Making a Dust Sensor using an Ionisation Chamber

Intro

This was a project to satisfy a curiosity of mine: “Could a household smoke alarm sensor be used to monitor air quality?”

Ionisation chambers draw very little current and can run on a 9v battery for tens of years. How often do you change your smoke alarm battery?

Makers around the world are looking into using dust sensors like the Sharp gp2y1010, DSM501, ZPH01 and SDS011 for environmental monitoring. The problem is, they all draw currents from 20mA to 100mA and so are not suited to long term battery operation. They also cost between 5GBP and 20+GBP, depending where you buy them.

Please bear in mind that this project outcome is not the finished sensor. That needs more work but it may encourage you to investigate and also to refine the circuit/calibrate the response.

Principle of Operation

A voltage applied to the terminals of an ionisation chamber give rise to a small current due to the ions thrown out by a tiny piece of Americium radioactive material.

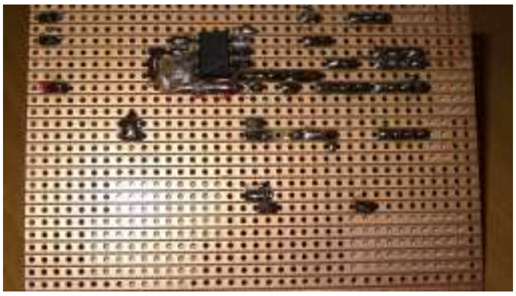
If smoke or dust enters the chamber the ions lose their charge and the current flow is reduced.

A very high impedance amplifier is needed to detect this voltage change.

Humidity will also affect the current flow, but, due to its conductive nature tends to make the current flow increase. Since smoke detectors are looking for a drop in current this doesn’t matter for the home smoke alarms but it will matter for air quality monitoring.

The development PCB

As you can see I used strip board to cobble this together. I figured the parallel tracks would enable me to provide a guard rail around the ionisation chamber output pin.

Top side of PCB		Track Side of PCB
		
The topside of the PCB has a couple of resistors to feed the ionisation chamber and test point plus a handful of capacitors to help reduce signal noise pick up.		The RE46C112 detector chip is soldered on the PCB tracks to minimise distance between the sense pin from the ionisation chamber and the detect input

This is a development rig to investigate fitness for purpose and a properly designed PCB is really needed to help reduce signal noise.		of the chip. Two tracks provide a guard rail for the output signal and the tracks are covered in wax to reduce moisture effects

Bill of Materials

Item	
NAP-07/HIS-07 ionisation chamber	Can be bought off Ebay for about £1.20 but come from China. On Aliexpress I found them for sale for £1.05. So, if you are going to wait anyway buy them from Aliexpress.
RE46C112 Smoke detector amp	Not available on Aliexpress. Price varies on Ebay – some advertise £1.11 but add £1.40 postage. Some advertise for £2.77 free postage. Bulk purchase from a chip supplier should see this come down to less than
2x 100uF Capacitors	Used for decoupling the supply voltages – low noise devices would be best.
2x0.1uf Capacitors	Used for decoupling the supply voltages – low noise devices would be best.
2x1m Ohm resistors	Some circuit diagrams use other values
Stripboard PCB	I used one 65mmx95mm. Price came in around £2. A dedicated PCB could be manufactured for a lot less.
header pins	I like my coloured header pins. I tend to use coloured pins to readily identify ground, supply and signals.

In all I made this for about £5 but I suspect it can be brought down to around £2 with a bespoke PCB and bulk purchasing of the RE46C112 chips.

Circuit Diagram

The RE46C112 datasheet is kind enough to provide a circuit diagram as below:-

Typical Application

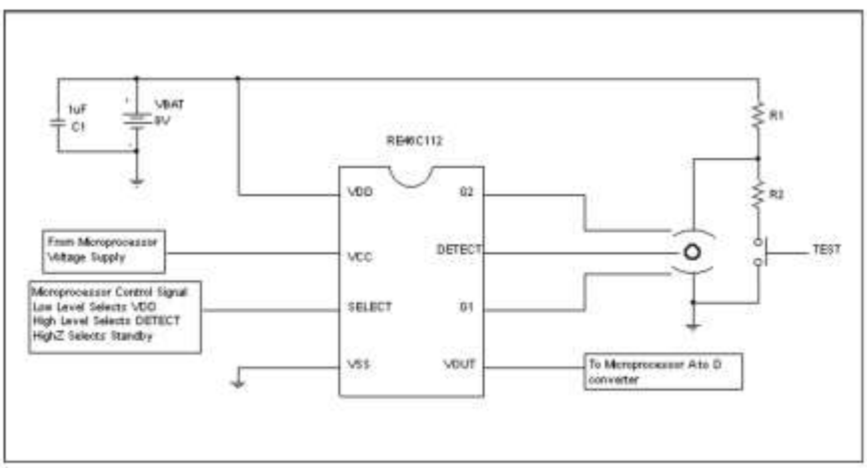


Figure 2

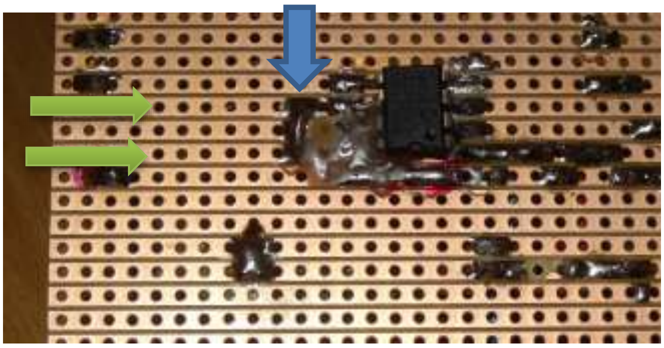
Most of this is self-explanatory. G1 and G2 are the guard amplifier output and guaranteed to be within 50mV of the DETECT signal and are used to shield the DETECT pin to reduce leakage.

R1 and R2 are not mentioned in the datasheet but other circuit diagrams use 1m Ohm and, sometimes, zero Ohm. I chose to use 1m Ohm because it means the normal current flow is limited to a maximum of 9uA. when using a 9v supply. Remember that dust/smoke causes the current to fall. The resistor used does affect the 'resting' voltage output from the amplifier chip. I did try 10K Ohm for a while (A load resistor of 10k was mentioned in the data sheet). The resting voltage was higher and I think the responsiveness to dust was greater – that's another thing to investigate but I really want R1 to be in the megohm region to limit current drain on the 9V battery.

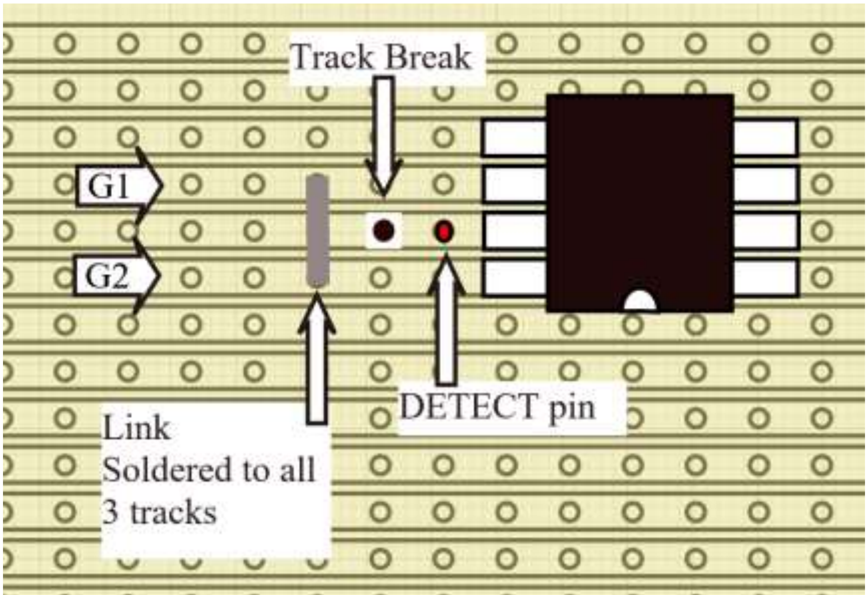
Guard Rail

Hard to see as I've covered the area with paraffin (candle) wax which is an insulator and keeps moisture away from the DETECT pin.

In this picture the two guard rails are highlighted (Green arrows). They are linked (Blue arrow) to form a complete guard. The DETECT input track is cut just before the link so that it is isolated.



Hopefully, this diagram clarifies the section under the wax.



After doing that cover the region with paraffin wax.

Arduino Sketch

To capture the sensor output as a real-time graph I used the Arduino-Plotter library https://github.com/devinaconley/arduino-plotter.

You can download the code from the above BUT when I manually installed it as a library it didn't work so I resorted to using the Arduino library manager – just search for Plotter. That worked except it didn't install the Listener code for the Serial Plotter in the Tools menu.

Having manually added that a window appears but I didn't get a chart so I used the Windows 64-bit listener outside the Arduino IDE.

If you go to the home page of the wiki you will find links to Listeners. These appear to be pre-set to use 115200 baud so don't mess with that setting in the plotter library.

Start your sketch then launch Listener.exe

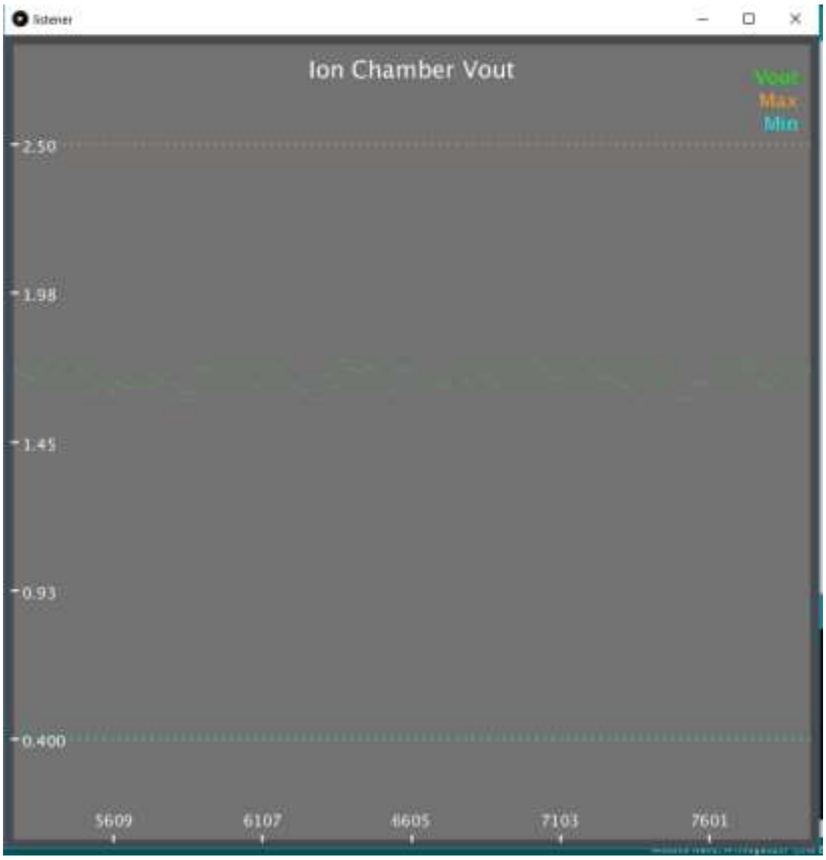
Here's my sketch. The Plotter code runs very quickly and there's some noise on my signal so I resorted to averaging a number of readings and

```
#include <Plotter.h>
#define GAIN    4.0         // multiply ADC voltage (Amplify signal)
#define NUM_AVG 20          // number of ADC readings to average
#define SIG A4              // analog input pin
#define CONTROL 10          // digital control pin
#define ADC_MAX_VOLT  5.0   // Depends on Arduino
#define ADC_MAX_REG 1024.0  // 10 bits on most


float Vout;      // Analogue voltage
float Max=2.50; // graph vertical range controllers
float Min=0.4;  // doing this stops plotter from adjusting vert axes

Plotter p;


void setup() {

  pinMode(CONTROL,OUTPUT);
  digitalWrite(CONTROL,HIGH);

  // now start the plotter

  p.Begin();

  p.AddTimeGraph( "Ion Chamber Vout", 100, "Vout",
Vout,"Max",Max,"Min",Min);

  Vout=ADC_MAX_VOLT*analogRead(SIG)/ADC_MAX_REG;
}

void loop()
{
```

```
  float avg=ADC_MAX_VOLT*analogRead(SIG)/ADC_MAX_REG;

  for (uint8_t c=0;c<NUM_AVG;c++)
    {
      avg=avg+ADC_MAX_VOLT*analogRead(SIG)/ADC_MAX_REG;
      delay(1);
    }

  Vout=GAIN*avg/NUM_AVG;

  p.Plot();
}
```

That's all there is to the code.

The output is shown below. The green line is the amplified (x4) averaged signal. The noise is present on the +ve connector of the ionisation chamber **but not on the 9v supply** hence I conclude that what we are seeing is the random nature of radiation and it's hard to smooth out other than by averaging.



Where Next?

To make this useful for Air Quality measurements it needs to be located with a known, calibrated, dust (PM) sensor to see if the AQI can be calculated from the output readings. It certainly responds quickly to being squirted with an aerosol.

I need to get some kicad whizkid to design a proper pcb even though I believe I tracked the noisy signal to the ion flow through the chamber