

# Using Virtual RGA Systems

Version 5.20 of Process Eye introduces the concept of Virtual RGA's. A Virtual RGA provides prospective users with a powerful means of evaluating Process Eye or EasyView and provides application developers with a great simulation platform for recipe code or customised software developed using the ASCII SDK.

## Part 1 - Introduction

### What is a Virtual RGA System?

A virtual RGA system is a complete PC-memory-based implementation of the RGA hardware along with any peripheral hardware such as an RVC, as well as a model of the vacuum system itself. A virtual RGA system appears to the application software as a fully functioning system in every respect. Here is just a summary of the features of a virtual RGA system.

The control unit and analyser are modelled, including

- Ion source parameters such as emission and electron energy
- Filter parameters including alignment and resolution
- Detector behaviour including a typical multiplier gain/voltage characteristic and optional peak noise
- Filament behaviour including filament trips and RF trip
- Analog and digital I/O
- External hardware such as an RVC or a Cirrus that controls external pumping

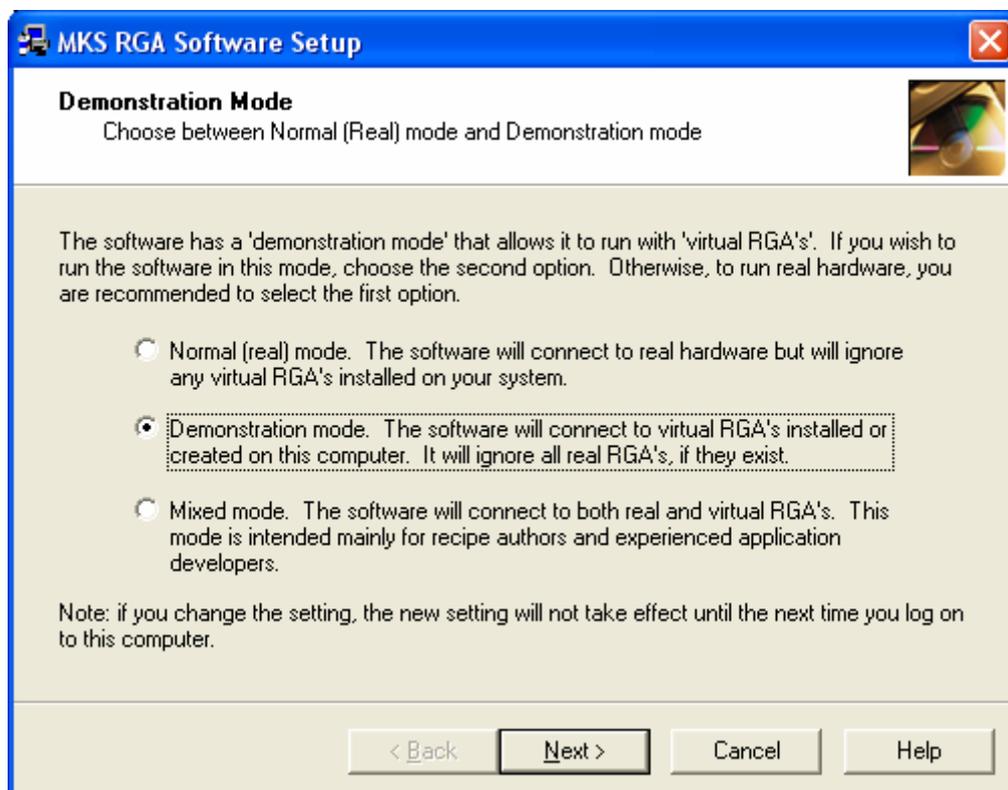
The vacuum environment is modelled such that

- The individual peak height of any mass can be set
- One of a selection of standard spectra can be displayed
- A dynamic profile can be 're-played' from a file
- Random variations in selected peaks can be incorporated.

All virtual systems are managed from a central control panel and changes to an individual system can be made interactively. On the other hand an ActiveX control is provided so that any input to a system can be programmatically changed as part of a complete simulation environment.

## How are Virtual RGA's Installed?

The software to exploit virtual RGA's is installed as standard from v5.20. You activate or de-activate virtual RGA's using a new page in the administrative setup application.



## How do Virtual RGA's Work?

A virtual system is a memory-mapped image of a control unit and its associated hardware. Version 5.20 of the RGA Server application has been modified to 'connect' to this memory mapping in just the same way as it connects to a 'real' instrument. Having connected to this virtual instrument it presents the same interface to the applications that connect to it, using the standard MKS ASCII protocol.

Thus, provided virtual RGA's have been enabled in the administrative setup, all the applications in the suite that communicate with the server as their source of data, behave exactly as though they were connected to a real RGA. You can write or run recipes for them, see them in RGA Device Manager, configure them with Hardware set-up, degas them and run diagnostics, calibrate them, save and recall data from them and even simulate what happens when the serial or network cable is pulled out.

You can control virtual RGA's using Process Eye, EasyView, the Developer's SDK or, at the most basic level, by using ASCII commands over telnet.

## **Are there any limitations to Virtual RGA's?**

Very few. You can create any number of virtual RGA's of a variety of types, including Microvision Plus, HPQ2, HPQ2S, e-Vision, e-Vision+, Cirrus, Vision 1000P, 1000C and 1000E and 300mm ResistTorr. The only limitations are

- There is no simulation of browsing to a network-enabled instrument
- There is no simulation of a VSC
- There is no simulation of rollover on standard analysers. If the vacuum on a standard analyser is set at  $1e^{-3}$  torr, the resultant peak height will be full scale  $1e^{-4}$  torr, rather than what would happen in practice where the peaks would be significantly lower than this.
- Likewise there is no internal simulation of the rollover correction procedure in an HPQ2S – that is to say that there is no attempt to simulate the actual (uncorrected) ion current at a given high pressure and then apply rollover correction to that signal.
- There is no core or app download.
- The filaments do not burn out when they are over-pressurised!

e-Vision control units are modelled as though they were serial instruments. They have all the same configuration as a real e-Vision and behave in EasyView (for example) exactly as a real e-Vision would, but they cannot be configured because they cannot be browsed to. However, the factory choices of 40eV or 70eV can be selected when the virtual e-Vision is created.

All the information in a virtual RGA system persists when you log off or close down your computer. This is obviously important for objects like an RVC (which must continue monitoring the state of vacuum all the time) and it also means that, like a real RGA, the source, filter and detector settings also persist through shut-downs and re-starts.

## Part 2 – Using the Software

### Getting Started

Run Administrative Setup and on the first of the administrative options pages, be sure to select the second (Demonstration mode) option. If you want the server still to connect to any real heads you have, select the third (Mixed mode) option. This third mode is really for specialist developers.

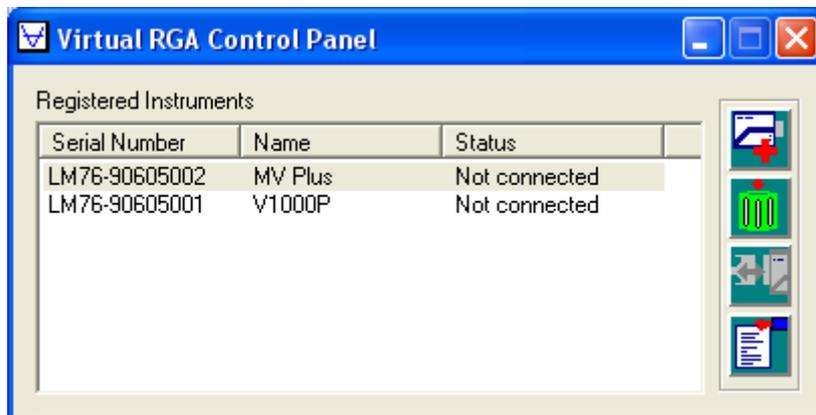
Log off as the administrator. Do this even if you run Process Eye as the administrator (which in version 5 you should **never** do!).

Log on as the regular RGA user. You will notice a new icon in the system tray.



This is the icon for the Virtual RGA Control Panel. (If you are wondering, it is the symbol for a vacuum ion gauge.)

Double click on the 'Virtual RGA Control Panel' (referred to from here on simply as the control panel). Until you create your first virtual instrument, it will have an empty list. Move the window down to the bottom of the screen to a convenient location.



Now, in the control panel, click the 'Add' button (the top one). From the drop down list choose the system you would like to create. The default is an LM76 Microvision Plus, but you can choose any one of a total of nine virtual systems. The control panel will suggest an unused serial number and name for your instrument, so you generally do not need to change what is proposed, unless you are creating a simulation of a real factory system. Click Ok and your first virtual system will be created. If you want further instruments, repeat the procedure again. Close the Virtual RGA Control Panel with the Close button.

Now run the regular user set-up. If you have just upgraded, run through to the end in the normal way, to register the new version. Then run it again. At the

hardware page select the first option to search for and configure the installed hardware. This, as usual, will launch RGA Device Manager. Your instrument(s) will be in the list. Right click on the instrument and select the 'Access...' menu to set 'exclusive use' of this instrument.

Close RGA Device Manager.

You are now ready to run Process Eye or EasyView and acquire virtual data.

### **The Virtual RGA Control Panel**

Run Process Eye or EasyView. Double click the control panel icon and check that the position is somewhere convenient on your screen. The position you select will be remembered the next time you use it. Double click an entry in the control panel and a new multi-tabbed window will appear, which is your interface to that virtual system. Position this window somewhere convenient. This position will also be preserved. You can click the Close box on any of these virtual RGA windows. The effect will merely be to hide the window. You restore a particular window by right mouse clicking in the system tray on

the  icon. This will display the main control panel menu.



The remaining buttons on the control panel are, in turn



The '**trash**' button. This removes the virtual system from your computer.



The '**disconnect / re-connect**' button. This simulates a link down between the computer and the control unit. Click the button once to invoke a link down. The status in the control panel will show 'link down'. Click the button a second time to re-connect with the server.



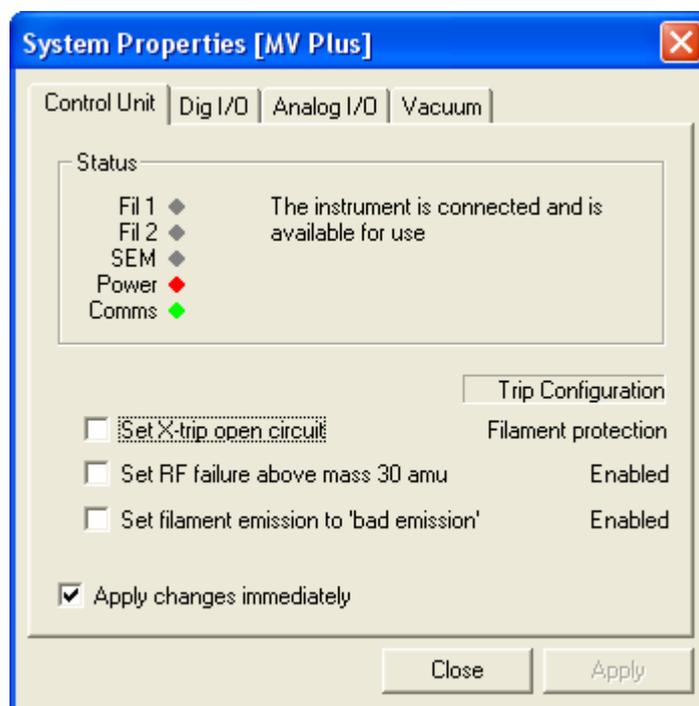
The '**properties**' button. This opens the multi-tabbed window that is the user interface to the selected virtual system. Clicking this button is exactly the same as double clicking the entry in the list.

The items on the preferences menu are global settings for the application as a whole.

Running the RGA Control Panel is the equivalent of powering up your control units. If you run the server but do not run the control panel, this is the exact equivalent of de-powering all your control units. Equally, if you run the control panel but close the server, this is the exact equivalent of plugging all your control units in to the 24V supply, but with no server running their COMMS LED's will be blinking slowly.

All the remaining interaction with a specific virtual RGA is carried out by activating its individual System Properties window as will now be described.

### **System Properties – The Control Unit Tab**



The first tab of the system properties window is a simulation of the standard control unit itself. You will notice it has the same LED's as a Microvision Plus/HPQ, which function in exactly the same way as a real control unit. If the control panel is running but the RGA Server is not, the comms light will be slowly blinking. If the server is running the comms light will be solid green.

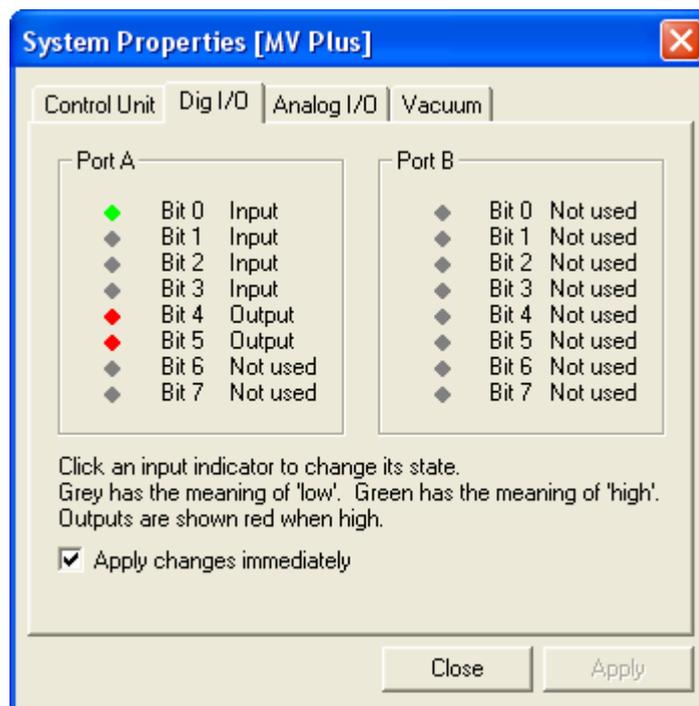
There are three options on this page.

- You can set the external trip open circuit. The effect of this depends on how the external trip has been configured.
- You can set the RF to fail above mass 30. If you do this you will see peaks below mass 30 but none above.

- You can set the filament emission to be 'bad' when filaments are switched on. Again the exact effect of this depends on how the unit has been configured.

The 'apply changes immediately' checkbox (which is checked by default) means that any change in the check state of the three main checkboxes is applied immediately. If this box is not checked you can make more than one change and no change will take effect until you click the 'Apply' button.

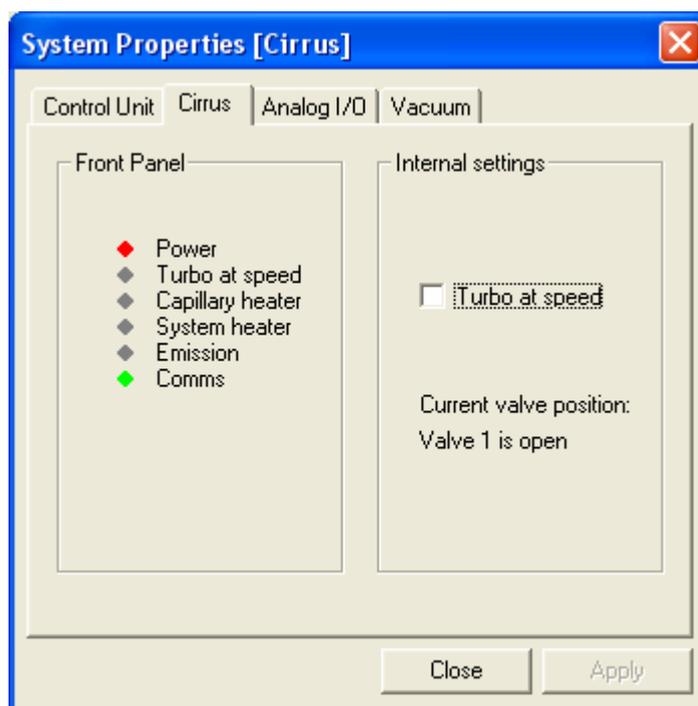
## **System Properties – The Digital I/O Tab**



If the control unit that you have created has available digital I/O, you will have a digital I/O tab which will show the state of the 16 digital I/O lines. Once again the exact content of this page depends on how the control unit is configured. Hardware set-up can of course be used to change the I/O configuration, or it can be set when the virtual instrument is created, using the Options... button.

Inputs are shown green when TTL high. Outputs are shown red when TTL high. You can change the state of any input simply by clicking it. You cannot change outputs. You have the choice of 'applying the changes immediately' or making more than one change and then clicking the 'Apply' button.

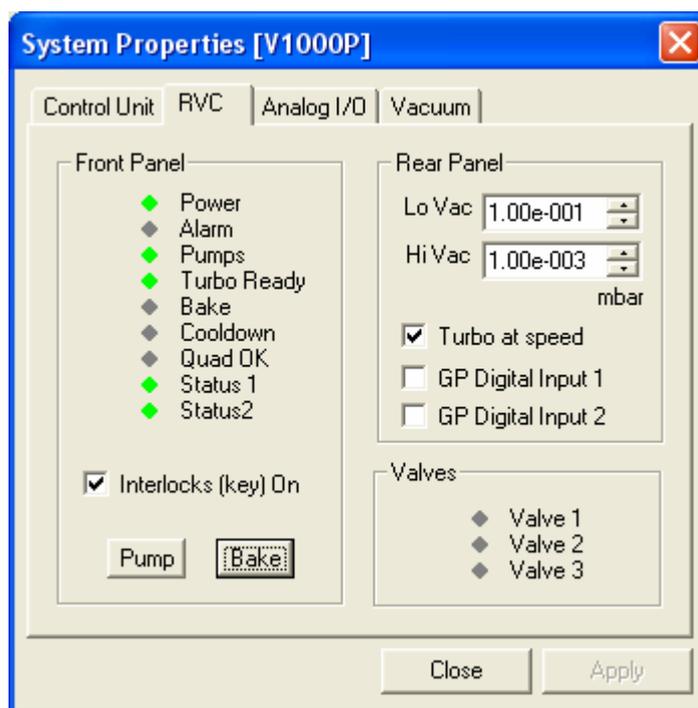
## System Properties – The Cirrus Tab



If the virtual system is a Cirrus, you will have a Cirrus tab. This shows the same LED's that are found on the front of the real Cirrus hardware. The only interactive 'input' that is available is the 'turbo at speed' setting. The behaviour of the filaments mirrors the real system. Filaments can only be on when the turbo is at speed and the system pressure is less than 5e-5 mbar.

If the Cirrus is configured to have a multi-valve inlet, the current valve position will be displayed.

## System Properties – The RVC Tab



If the system has an RVC, you will have an RVC tab, which is a full virtualisation of a real RVC. You will see exactly the same front panel as a real RVC and you have the ability to mimic the external input functionality using the user interface. Once again you can set the turbo 'at speed' state, the two general purpose digital inputs, the maintenance key switch and the settings for the pressure switch that governs the Status1 and Status2 lines. Like a real RVC, the Pump and Bake buttons can be used to toggle the state of the relevant item.

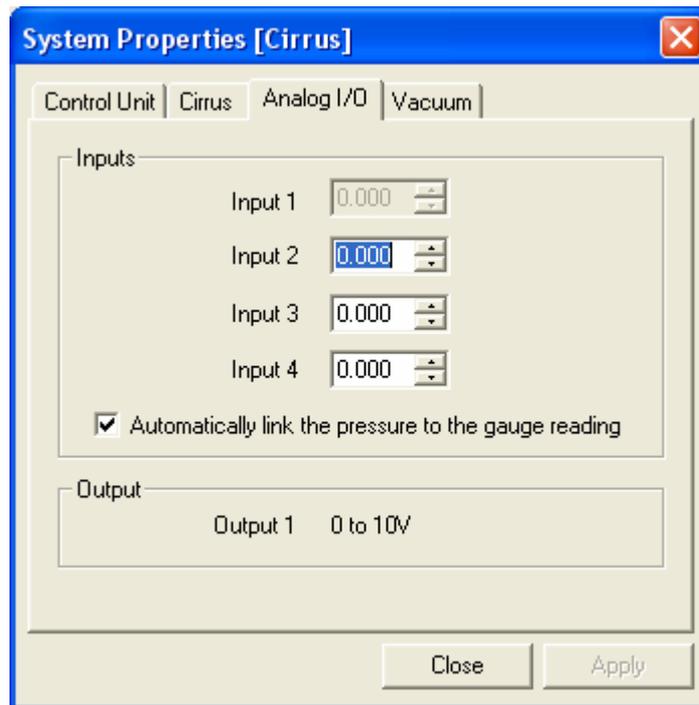
If you change the pressure switch set point pressures or the general purpose digital inputs you must click the 'Apply' button when you have the settings that you want.

In common with all virtual outputs, valves show red when open.

As you would expect, you can make changes to the RVC either through the virtual user interface or through the Process Eye or EasyView RVC interface. In either case the new state is updated in both places.

On a Vision 1000P system, which only has one valve, both the Status1 and Status2 lines are and-ed together and the two valve outputs work as one. This is equivalent to the standard behaviour.

## System Properties – The Analog Input Tab

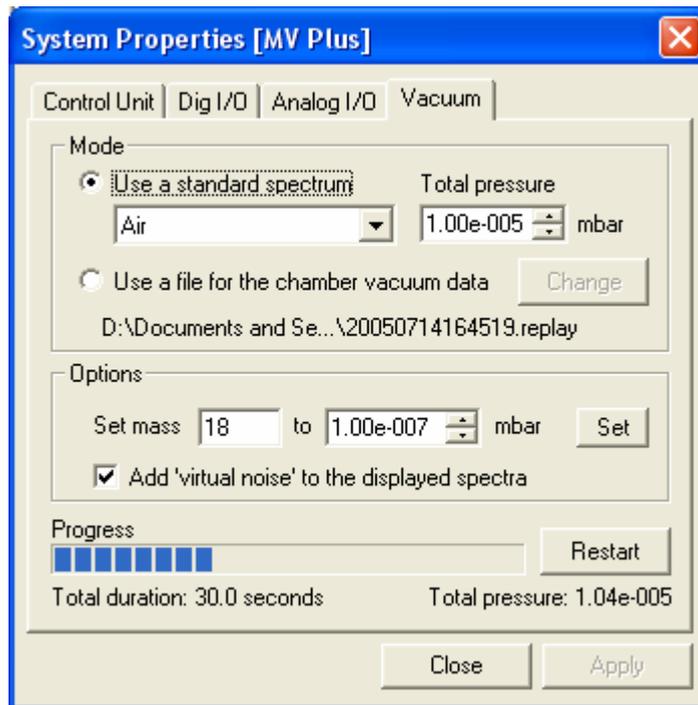


If the control unit has analog inputs, you can set their value on the analog input tab. In all cases of setting a numeric value such as the voltage where you have the option of entering the text or using the spin buttons, you make a change either by

- Editing the text and clicking Apply
- Editing the text and pressing the tab key to move to another control, then clicking Apply
- Using the spin button(s), then clicking Apply

If the control unit has a reserved analog input for a pressure gauge, you can choose whether the simulator automatically sets the value to the appropriate value for the particular gauge. A Cirrus has a gauge on Input 1, whereas an HPQ2S may have a gauge on Input 4. When you link the total pressure to the analog input in this way you cannot change that channel manually. If you break the link, you can change the input value and see your change reflected in the pressure reading in Process Eye or EasyView.

## System Properties – The Vacuum Tab



This tab is in many ways the most interesting since it permits interactive changes to the data displayed by the virtual RGA. There are two choices.

***The first choice is to pick one of a range of standard spectra*** at a pressure of your choice. Pressure units are always shown in the units that you have set for Process Eye or EasyView. Be aware that the pressure specified is the actual chamber pressure. If your system has a special inlet, the inlet factor will be applied to this value. If the pressure specified results in an individual partial pressure exceeding the maximum measurable pressure for the gain conditions in force at the time, the displayed peak height will be full scale. On the other hand, if the peak is so small that it cannot be measured, the peak will be 'in the noise'.

You can elect whether the simulation should add appropriate measurement 'noise' to the returned peak height.

Any change to the selected spectrum or pressure will take effect when you click the Apply button.

The information at the bottom of the window shows the actual sum of peaks currently measured – this will often differ just slightly from the requested value at the top.

When you are displaying a standard spectrum you can change the partial pressure of an individual mass using the 'Set mass to ...' and clicking the Set button. The new setting will apply for the remainder of the current 'replay scan' (whose length you can determine); it will then revert to the normal setting for that spectrum. The progress bar shows how much time remains for the scan. Some of the spectra are 'dynamic' which means that some

constituents such as helium, water or air change on a random basis. Each time the progress bar starts over, a new composition will be applied.

You can use the Restart button to restart the scan prior to clicking the Set button to change an individual mass.

The duration of the standard replay scan is an option that can be set on the preferences menu.

**The second choice is to 'replay' a vacuum profile from a file.** This file can have one or many individual 'scans'. The file is typically generated using your favourite spreadsheet. If you want to 'replay' real live data, you use Recall to save your data as a VVP file (virtual vacuum profile). This can be read directly by the virtual RGA.

The format of the file is explained in appendix 1.

The Restart button is used to synchronise the data back to the start of the first scan.

A replay file can be 'real' or entirely artificial. A real replay file is one derived from real data where each 'scan' has a similar duration, for example every 4 or 5 seconds. Replaying this file would produce a trend view that looked very similar to the one stored in the original data file. An 'artificial' replay file is typically created by a recipe developer to test the behaviour of the recipe in alarm conditions. The scans in this replay file may have widely varying durations for reasons that will become apparent later.

Once again, you can change an individual mass peak using the Set button, but remember that the change will only last until the next replay file scan.

Please understand that it is not the replay scan that **drives** the scan in Process Eye. Process Eye is a subscriber to data but has no mechanism for 'asking for' a particular reading at a particular time. The replay scan is the provider of data values during that replay scan's lifetime, but there is **no synchronisation** between the start of a Process Eye scan and the start of a replay scan. This means that if the data in the file was acquired at accuracy 5 but it is replayed at accuracy 3, the replay duration will remain the same but you will get more scans than in the original. Likewise, if it is replayed at accuracy 8, you will get fewer scans than the original. Where synchronisation is critical (usually only to developers rather than software evaluators) a synchronisation tool is provided and is described below.

If you are particularly interested in what is happening at low partial pressures, you will probably choose to un-check the 'Add 'virtual noise' to the spectra' checkbox because the simulator will be adding more noise to the noise already stored in the file.

More advanced information about the use of replay files is provided in a later section.

## **General Information about Partial Pressure Simulation**

When the simulation generates a partial pressure reading, it takes note of a number of virtual RGA configuration settings. The mass alignment is modelled exactly and the resolution is also modelled, particularly on the over-resolved side.

Electron emission is modelled so that the peak height is proportional to the emission up to a maximum of 10% over the recommended value. Electron energy is modelled to the degree that mass 4 is dependent on it and mass 20 is very small if the 20/40 ratio looks like argon.

The multiplier detector has a recognised voltage / gain characteristic.

For these reasons, please be aware that you will only get the partial pressure reading recorded in the file or standard spectrum when

- The alignment and resolution settings are all set to 32767
- The filament emission current is set to the default value for the source setting
- The electron energy is at its appropriate default for the source setting.
- The multiplier gain/voltage matches a two-part graph of 30/-650 to 300/-750 or 300/-750 to 3000/-900.
- The sensitivity (amps per unit of pressure) for the active source setting is the default value for that setting.

If any of these items are not the case, the peak heights will not match the value stated on the Vacuum tab of the virtual RGA window.

Noise is simulated in three parts – ADC noise, ion source noise and detector noise.

## **Calibrating a Virtual RGA**

It is perfectly possible to calibrate a virtual RGA using EasyView or the built in Process Eye methods, or indeed any recipe method of your own. The Vacuum tab of the virtual RGA window will show what is the current sum of the partial pressures, but you are free to calibrate the height of any peak to any value you choose.

For example, suppose you are displaying the argon spectrum, so that the simulation peak height of mass 40 is  $1e^{-5}$  torr (as determined by the vacuum tab).

Now run a calibration on mass 40 but state that you want to calibrate mass 40 to  $1e^{-6}$  torr. The calibration will run, and when you run a bar chart afterwards the height of mass 40 will indeed be  $1e^{-6}$  torr and the sensitivity (amps per unit of pressure) of the analyser will be 10 times higher than the 'standard' value.

## Modifying the Initial Configuration of a Virtual RGA

When you create a new virtual RGA using the 'Add' button, the 'Add New Instrument' dialog has an 'Options...' button which allows you to set a number of general options that will apply to all new systems that you create.

The options shown in this dialog will be applied to a new instrument when it is initially created. Note that not all options apply to all instruments.

**Control Unit**

Maximum mass: 200 amu

HPQ2S gauge type: GP ion gauge (Torr)

e-Vision 70 eV ion energy (default is 40 eV)

**Digital I/O**

Port A inputs: 4      Outputs: 2

Port B inputs: 0      Outputs: 0

Cirrus Valves: 4-position

Initialise available digital outputs high

**RVC**

Max 'on' seconds without cooldown: 10

Cooldown seconds: 30

V3 connection: Purge

**Trip actions**

Filament off if emission fails

External trip mode: Filament protection

OK      Cancel

In the example shown here, the following defaults have been selected.

- Where possible, analysers are assumed to be 200 amu, and when an HPQ2S is created it will have a Granville Phillips ion gauge with Torr units attached.
- e-Vision control units will be set to the 40 eV option.
- The default trip actions are as shown.
- Where a control unit allows it, the digital I/O will be set to have the first 4 bits as inputs and the next 2 bits as outputs on port A, with no inputs or outputs on port B. General purpose digital outputs will be initialised TTL high. Where a Cirrus system is created, it will be configured to have 4 valves.
- When an RVC is created, the cool down parameters will be set so that there are only 10 seconds of bake before cool down will be enforced, and the cool down period will last for 30 seconds. The valve mode for V3 behaviour is the standard 1000C/E nitrogen purge. The alternatives are to select the 300mm ResistTorr behaviour of admitting a calibration gas under scripted recipe control, or to have the 1000P behaviour where V3 is not connected.

## **Part 3 – Things to do with a Virtual RGA**

Once you have created your virtual RGA systems you might like to try some of the following using the EasyView recipe or installation.

- Set up Process Eye / EasyView so that it takes up the full height of your screen but leave space on the right hand side so that you can see the System Properties window for your installed system(s). (If the System Properties window is not visible, right-click on the Virtual RGA Control Panel icon and select the system from the pop-up menu.)
- Switch on the filament and check that the filament LED goes green on the virtual UI.
- Start a barchart scan and check that you see an air spectrum. Using the UI select the vacuum tab and change the system pressure, then click the Apply button – watch the peak heights change.
- Select a different spectrum from the drop-down list and click 'Apply'.
- On the Control Unit tab, check the 'X-trip' check box and watch the filaments turn off and the peaks disappear. Uncheck the box and manually turn the filaments back on.
- Check the 'RF fail above mass 30' check box and watch the spectrum.
- Do the same with the 'bad emission' check box. The behaviour depends on how the unit is configured.
- Use the user interface in the EasyView recipe to change the gain range and the detector. Notice how, as you use a higher gain you see peaks that you did not see at lower gains. Can you see any helium? What about Krypton? Any other unexpected peaks?
- Set up a spectrum of argon and see what you get on both 'standard' and 'low' electron energy.
- Now run an analog RGA spectrum and select the tuning tab. Experiment with the low and high mass resolution and alignment.
- Using the source settings tab, experiment with the filament emission current and see the effect on the peaks.
- See if you can observe any effect on the air spectrum of changes to electron energy.

Now create a Cirrus benchtop system. (See if you can create one with a 4-valve multi-inlet.)

- Run the EasyView recipe (or run the EasyView application) and select the Cirrus button in the recipe UI. Select the Cirrus tab in the Virtual System UI.

- Turn on the Cirrus pump – use the UI to get the turbo at speed. Notice that the gauge pressure readout is activated.
- Turn on the RGA filaments and run a barchart.
- Notice how the virtual UI LED's mimic exactly the ones shown on a real Cirrus unit.
- Turn on the capillary heater and/or the main heater.
- Can you find the effect of checking the 'Chamber hotter than 90C' checkbox? (Clue: run a scan using the multiplier).
- What happens if you turn off the pump?
- What happens if, after turning the pump back on, you increase the system pressure on the Vacuum tab of the virtual UI to about 6 atmospheres? (Actually make the Cirrus internal gauge read greater than  $5e-5$  mbar or  $3.75e-5$  torr). Don't forget to click the 'Apply' button when you make system pressure changes.
- Check that when you change the inlet valve that the virtual UI recognises the fact.

Now create a virtual system that uses an RVC – maybe a Vision 1000P or a Vision 1000C.

Start EasyView again and use this new virtual system.

- Check that you can start and stop the pumps using the EasyView UI or the virtual system UI. The virtual UI RVC front panel is laid out exactly like the real one.
- Experiment with the RVC key switch toggling between maintenance and manual / automatic mode.
- Do you know the steps required to get the valves open in automatic mode? You need to make sure that the system vacuum is low enough to get the status1 and status2 lines green. Then you need the filaments on and the Quad OK light to be green on the RVC. The behaviour is subtly different between a 1000P and a 1000C.
- Run this system and acquire a barchart. What happens to the spectrum when the valves close? Actually the answer depends on whether you created the virtual system with a purge or a cal gas or a direct connection in the virtual RGA UI options dialog.
- Try baking the chamber. What effect does this have on the scan? Check that if you leave the bake for long enough, the system goes into cooldown.
- Check that the general purpose digital inputs work correctly – you need to click 'Apply' for these.
- You can change the Status1 and Status2 set points on the virtual UI.

Back on the standard control unit with standard general purpose digital I/O

- Check that when you change an input on the virtual UI that you see the corresponding change in Process or EasyView.
- Try running diagnostics on the control unit.
- Try and Degas the ion source.

Close Process / EasyView and run RGA Device Manager. Right mouse click on the control unit and choose 'Configure or Install...'. You will be asked for a password – any non-empty password works for a virtual RGA. You can change any behaviour that you wish to – then run the control unit again in EasyView to see the effect. A new inlet factor? A new trip behaviour?

Try calibrating your virtual control unit. If the UI Vacuum tab says that the pressure is  $1e^{-6}$ , try calibrating at a pressure of  $1e^{-5}$ . Then run the barchart again. Is the peak height  $1e^{-5}$  or  $1e^{-6}$ ?

Bring up the main Virtual RGA Control Panel window and click the 'Link-down' button. (The third one down). You can restore the link by clicking the button again. What happened?

If you think you have made a mess of these virtual control units, just send them to the trash-can (button two) and create some new ones!

If you stored data files using EasyView, you can Recall them using the Recall application.

If you installed the Process Eye Professional option, you can try writing a new recipe for your virtual RGA using the Wizard and then running it. There are no restrictions on what is available to you in demonstration mode. All the features of the recipe object model work in the same way as when using real hardware.

## **Part 4 – Synchronising a Virtual System with a Full System Simulation**

This is an advanced topic that will mainly be of interest to developers.

You have already seen that the scans of vacuum data in a replay file are not directly linked to the scans in Process Eye. Some background concerning data acquisition using a real RGA may be helpful here. The control unit is (within limits which we will ignore here) free to acquire data asynchronously at a rate determined by the accuracy setting. The server (Windows or IP) continuously goes round the loop of asking the control unit if it has new data, processing that data and sending it to Process Eye and then asking the control unit again. Sometimes the control unit has no data at all (at high accuracies) because new values only come every one hundred milliseconds or so. On other occasions (low accuracy leak check, for example) the control unit may send back many items of data in one response.

The server pushes its calibrated data to Process Eye, which is a passive recipient. Items of data are 'labelled' so Process 'knows' where the new items belong. It also knows when it has received all the data for a 'scan' and when this is the case it calls the script functions to do user channels and end of scan actions.

In a virtual RGA the Process Eye end is completely unchanged, but the data server effectively talks to itself to get new data. This is implemented using a 100-millisecond timer. Each time this goes off, the server, in conjunction with the virtual UI, determines how many readings could have been acquired since the previous report, allowing for any unused time from the previous period. These readings are, of course partial pressures but also analog inputs from virtual gauges and other external 'hardware'. The partial pressures for the next masses in the Process Eye scan are calculated from the currently applicable partial pressures in the replay scan. The number of readings reported depends on the accuracy and the zero and detector delays that are part of the hardware setup.

Recipe application developers will need much tighter synchronisation of the replay file to the other drivers of the system, such as SECS messages or even digital inputs to the control unit. You can take control of all the virtual system inputs programmatically, using the supplied ActiveX control. A simple Visual Basic 6 project is available that demonstrates the use of this control.

### **The Synchronization ActiveX Control**

You can use this ActiveX control in a VB project or in a scripted recipe.

The object model contains two read only properties, one write-only property and several (write only) methods. None of the methods either cause or return

any errors. If you invoke a method that cannot be executed, for example because the particular virtual instrument is off-line, the method does nothing.

You invoke the ActiveX control either by adding a project reference to 'MKS Virtual RGA Synchronization v1.0', then coding

```
Dim sync As VRGASYNCLib.VrgaSynchronize 'VB declaration
Set sync = CreateObject("vrga.synchronize")
```

Or, for VB script

```
Dim sync
Set sync = CreateObject("vrga.synchronize")
```

This is a brief description of the properties and methods

### **Properties**

---

#### **IsInstalled**

returns 0 if the administrator has disabled virtual RGA's or nonzero if they are enabled. This property is read-only.

---

#### **CanConnect (Name As String)**

Returns non-zero if the virtual instrument named '**Name**' is powered up and connected to the server. Returns 0 if virtualRGA.exe is not running or the control unit is in a link-down state.

Note that if this property returns zero, none of the following methods will succeed, but they will not generate errors.

This property is read-only.

---

#### **StandardScanDuration (TimeInSeconds as Long)**

Sets the duration for standard spectra scans. Must be greater than 1. Does not have to be one of the standard preferences on the main menu.

This property is write-only.

## Methods

---

### AnalogInput

AnalogInput(Name As String, Port As Byte, Value As Single)

Sets analog input **port** on virtual instrument **name** to a value of **Value**.

**Port** is 0-based.

```
AnalogInput "MV Plus", 0, -5.00
```

sets analog input 1 to -5.00 volts

---

### DigitalInput

DigitalInput(Name As String, Port As Byte, FromBit As Byte, ToBit As Byte, SetTo As Byte)

Sets digital port **Port** on virtual instrument **name** to a value specified by **FromBit**, **ToBit** and **SetTo**. **Port**, **FromBit** and **ToBit** are all 0-based. Unconnected inputs are ignored, as are all outputs.

```
DigitalInput "MV Plus", 1, 2, 4, 1
```

sets digital inputs as follows: PB2=1, PB3=0 and PB4=0

---

### PeakHeight

PeakHeight(Name As String, Mass As Integer, Value As Single)

Sets the peak height of mass **Mass** to a value of **Value** Pascal on virtual instrument **Name**

```
PeakHeight "MV Plus", 28, 1e-4
```

sets the peak height of mass 28 on "MV Plus" to  $1\text{-e}^4$  Pa **for the duration of the current scan**.

---

## Restart

`Restart(Name as String)`

Performs the equivalent of clicking the Restart button on the Vacuum page. Restarts the scan or file. Functionally equivalent to setting `ScanNumber` to 1.

```
Restart "MV Plus"
```

---

## RVCGeneralPurposeDIs

`RVCGeneralPurposeDIs(Name As String, SetTo As Byte)`

Sets the two general purpose digital inputs on the RVC attached to virtual instrument **Name** to a bit pattern **SetTo**. **SetTo** has a value between 0 and 3 where bit 0 is the state of input 1 and bit 1 is the state of input 2.

```
RVCGeneralPurposeDIs "MV Plus", 2
```

sets input 1 low and input 2 high

---

## ScanNumber

`ScanNumber(Name As String, ScanNumber As Long)`

Sets the current scan number (1-based) being 'played' by virtual instrument **Name**.

```
ScanNumber "MV Plus", 19
```

sets the active scan to 19.

---

## ScanTime

`ScanTime(Name As String, HH As Integer, MM As Integer, SS As Integer)`

Sets the current scan to the one relevant to the specified time. HH can be greater than 23.

```
ScanTime "MV Plus" 2, 30, 0
```

sets the elapsed time in the file to be 2 hours 30 minutes. Note that the specified time does not have to explicitly match any particular elapsed time in the replay file.

---

## StandardSpectrum

StandardSpectrum(Name As String, SpectrumID As Integer, TP As Single)

Sets the active standard spectrum for virtual instrument **Name** to be the one corresponding to the 0-based list index **SpectrumID** at a total pressure **TP** Pascal.

```
StandardSpectrum "MV Plus", 2, 1.333e-3
```

sets the standard spectrum to air on virtual RGA "MV Plus" at a total pressure of 1e-5 torr.

---

## TurboAtSpeed

TurboAtSpeed(Name As String, bAtSpeed As Byte)

Sets the turbo 'at speed' on virtual instrument **Name** if the **bAtSpeed** is non-zero, otherwise sets it to 'not at speed'. This method works equally well for an RVC system and a Cirrus system.

```
TurboAtSpeed "V1000P", 1
```

sets the turbo 'at speed'.

---

## TurboPumpOn

TurboPumpOn(Name As String, bOn As Byte)

Sets the pump on or off on an RVC system (it has no effect on a Cirrus).

```
TurboPumpOn "V1000P", 0
```

sets the pump off on virtual instrument "V1000P".

You will remember that there is never any error when calling any of these functions, so on a real system without virtual RGA support installed they become do-nothing commands and can therefore safely exist in your shipping recipes. You just need to protect against failing to create the Synchronize object, which would occur if the recipe were used on a system running a version prior to v5.20.

```
Dim sync
```

```
Set Sync = Nothing  
If App.Version >= 5.20 Then
```

```
    Set Sync = CreateObject("vrga.synchronize")
End If

If Not Sync Is Nothing Then Sync.TurboPumpOn "V1000P", 1
```

## Appendix 1 – Vacuum Profile ‘Replay’ File Format

### **Basic Format**

The ‘Virtual Vacuum Profile’ or ‘replay’ file can be either a csv file or a tab-delimited text file, either of which can be created directly by Excel. Alternatively you can use a Recall data file to create a .vvp (virtual vacuum profile) file for you.

The rules for the file are as follows.

1. You can have any number of lines of text at the top of the file for description and comments.
2. The first critical line in the file is a line that contains [UNITS] in column A. Column B must then contain one of pascal, torr, mbar or millitorr. None of the text in a vacuum profile file is case-sensitive.
3. The next critical line is one that contains [DATA] in column A. Columns B onwards then contain the masses that are going to have specified peak heights. These masses must be in ascending order but there can be gaps, i.e. you can specify 2, 14, 17, 18, 28, 32. You can specify any masses in the range 1 to 300. You can optionally prefix the numeric value with the word ‘Mass’ (without quotes).
4. Every succeeding row must contain partial pressure data. There can be no blank rows, nor rows with comment text, although you can write comments in columns beyond the highest mass. Each row has an elapsed time in column A using the hh:mm:ss format. Columns B onwards contain the partial peak height for the mass specified above in the units specified. You must specify the partial pressures in the format x,yyE-nn. It does not matter what the decimal separator is but there must be exactly 3 significant figures to the mantissa. The exponent does not have to have a + sign. The exponent can have 1, 2 or 3 digits with or without leading zeros. If you follow these rules you can work with any international settings.

When the file is ‘played’ the simulator executes the first row (scan 1) until its elapsed time is completed. The simulator then moves on to scan 2 until the total elapsed time exceeds the time in column A of scan 2. Finally, when the total elapsed time exceeds the column A entry of the last row, the whole thing is replayed again.

## Examples

Here is an example of a tab delimited file

Any text here on multiple lines as a description  
Of what the file contains. Then a line beginning with [UNITS]

```
[UNITS]      mbar
[DATA]       14          28          29          32
0:00:15     1.12e-7     2.34e-5     5.67e-8     8.99e-6
0:00:30     1.12e-7     2.34e-5     5.67e-8     9.23e-6
0:00:45     1.12e-7     2.34e-5     5.67e-8     9.57e-6     a comment
0:01:00     1.12e-7     2.34e-5     5.67e-8     9.81e-6
```

or

```
[UNITS]      mbar
[DATA]       Mass 14      Mass 28      Mass 29      Mass 32
0:00:15     1.12e-7     2.34e-5     5.67e-8     8.99e-6
0:00:30     1.12e-7     2.34e-5     5.67e-8     9.23e-6
0:00:45     1.12e-7     2.34e-5     5.67e-8     9.57e-6     a comment
0:01:00     1.12e-7     2.34e-5     5.67e-8     9.81e-6
```

This plays a file for 1 minute showing a gradually increasing 32 level, while masses 14, 28 and 29 stay constant. After 1 minute the simulation repeats itself.

There is no limit to the number of 'scans' that your file contains – until the computer runs out of memory.

That was a simple example. A more complex example would be when you want to construct a replay file that models an entire semiconductor processing fab.

Here is how you might integrate with your fab simulation. Define the first hour of the replay period to correspond to idle. So from 0:00:00 to 1:00:00 define a scan profile that may change peak heights every 20 seconds for the first 20 minutes and then has a scan line that lasts until 1:00:00.

Now define that the period in the file from 1:00:00 to 1:30:00 is wafer processing and 1:30:00 to 2:00:00 is inter-wafer. Set up your desired profile for each of these, but only use scans every 2 or 3 seconds for the first minute, then add the 1:30 and 2:00 lines to use up all the remaining time.

Now in your VB simulation, when the time comes to idle, simply call

```
Sync.ScanTime MyChamberName, 0, 0, 0
```

or

```
Sync.Restart MyChamberName
```

And when you send a wafer start message to Process call

```
Sync.ScanTime MyChamberName, 1, 0, 0
```

And a wafer end message, call

```
Sync.ScanTime MyChamberName, 1, 30, 0
```

Then all you have to do is ensure that your idle simulation duration never normally exceeds 20 minutes and certainly never exceeds one hour (or your RGA data will run into your wafer processing profile). Likewise ensure that your wafer process and inter-wafer time last no more than one minute.

### ***Random Behaviour***

You can introduce random behaviour by setting a peak height to be negative. If you set a value for, say mass 18 to be  $-1.00e^{-8}$  torr, the simulator will generate a new random number at the time that the scan becomes active and display a partial pressure somewhere between  $1.00e^{-8}$  and  $1.00e^{-13}$ , ie somewhere evenly logarithmically distributed over 5 decades.

Different random numbers are generated for mass 4, mass 17/18, 14/16/28/32 and the remainder of the peaks. Over the course of a number of scans this means that sometimes helium will be critically high, whereas on other occasions air, water, nitrogen, oxygen or general contamination will be critically high. This allows you to evaluate many different alarm conditions.

If the ratio of 17/18 is between 0.2 and 0.3, 17 and 18 are considered to be water and get modified by the random water factor.

If the ratio of 32/28 is between 0.18 and 0.28 then 14/16/28 and 32 are air and get modified by the air random factor

Otherwise, if 16/32 is between 0.03 and 0.04 then 16 and 32 are Oxygen and get modified by the random oxygen factor

Otherwise, if 14/28 is between 0.046 and 0.056, then 14 and 28 are nitrogen and get modified by the random nitrogen factor

Otherwise 14, 16, 28 and 32 get modified by the general random factor.

Mass 4 is modified by the random helium factor and all other masses are modified by the general random factor.

Part 4 of this document described how, by using the ActiveX control, you can programmatically position the scan number or current elapsed time within the file mapping.

### ***Using a Recall File***

The steps in converting a Recall file to a Replay file are:

1. Open the file in Recall

2. Choose File | Save As... and be sure to select 'Virtual vacuum profile file (\*.vvp)' from the File Type drop-down list.
3. At the options dialog, the options on the left hand side are all pre-selected for you.
4. If there is more than one measurement listed on the right hand side, you must choose just one – usually the main barchart. You cannot use a user measurement or an analog scan but you can use a peak jump measurement. Recall will automatically sort peak jump measurements into ascending mass order.
5. Save the file.

### **Noise Behaviour**

Finally – and this is only really of interest to those who enjoy 'splitting hairs' – can you see what happens to file noise when it is replayed with no simulator noise added? A typical barchart file has a number of masses where no ion peaks exist so the data for that column is a mixture of positive and negative values around 'zero'. Take for example a traditional ¼ inch rod RGA on Faraday at low electronic gain. The range of values might be, say, from  $1.2e^{-9}$  torr to  $-1.2e^{-9}$  torr, representing +/- 100 ADC units. Now you already know that positive values are displayed as they are, but negative values are displayed as somewhere between the full positive value and 100,000 times less. So, in this example, and bearing in mind the logarithmic distribution of random values, two out of five occurrences of  $-1.2e^{-9}$  torr will actually be converted to a displayable positive reading. Likewise two out of fifty readings of  $-1.2e^{-10}$  torr will be displayed as a positive value.

Alternatively, at the dialog that follows the 'File | Save As' menu selection in Recall, you can clip all the data values at a very low positive value. This should have the effect of making the replay spectrum the same as the original because the clip value will usually be less than 1 ADC unit since the clip value is actually  $1e^{-16}$  mbar.