TOFSYS data system basis

This program has been created to replace the PASCAL TOFSYS program which was running on the Macintosh IIci computers. Now it is working on a PC under Microsoft Windows© and handles all control, data display, processing and data archiving and is written in Labview© v 6.0 for portability programming.

Experimental Control

The experiment control falls into two main areas; TOF timing and Signal acquisition.

TOF timing

The first is the experiment running and synchronisation signals. These are generated in hardware from the TOF timing unit. There are set via RS232 and include:

- Pulse beam period
- Chopper disc rotation speed
- Pulse beam firing delay
- Laser firing delay
- TOF delay
- Scalar window width
- collection mode (A, B, etc.)

The TOFSys program must:

- provide data entry of all parameters
- check the compatibility of parameters
- check ranges of parameters
- transfer parameters to TOF timing unit via RS232
- transfer parameters from TOF timing unit via RS232

The RS232 commands for the TOF timing unit are routed transparently through a 486 MS-DOS based PC computer.

Signal Acquisition – ion counting

The signal acquisition is via custom-built dual multichannel scalar (MCS) based in a 486 PC. The MCS is implemented in two parts: (i) a fast dual-channel single-shot MCS in hardware on a PC card for storing beam on/beam off data following a single gas/laser pulse and, (ii) data transfer from the card + signal averaging performed in software using the 486 PC processor and main memory under Quick Basic/machine-code control.

The MCS has several parameters to be set up

- number of channels
- channel width
- collection mode (A, B, etc.)
- number of shots

The TOFSys program must:

- provide data entry of all parameters
- check compatibility of parameters
- check ranges parameters
- transfer parameters to the 486 PC via RS232
- transfer parameters from the 486 PC via RS232
- Initiate data collection 486 PC via RS232
- interrupt/abort data collection 486 PC via RS232
- transfer data from 486 PC via RS232

Once again RS232 commands for the TOF timing unit are routed transparently through the 486 MS-DOS based PC computer

Data processing and display

As well as setting up the hardware requirements the TOFSys program provides:

- display of all running parameters
- collection/display other auxiliary experimental parameters: masses, pressures, T, etc.
- default running parameters sets (i.e. sets of commonly used parameters)
- an acquisition status display (running, stopped, no of shots, etc.)
- error checking for data acquisition
- display Beam on, Beam off and Difference data graphically
- calculate correct channel times allowing for all timing delays and chopper pulse widths
- perform data processing to correct for ion-times (iterative correction), and conversion from number density (ND)-time to velocity-ND and velocity-flux distributions (with Jacobians)
- smoothing of data
- tabular version of the data
- multiple data sets
- amalgamation of datasets
- saving of complete datasets with all running and auxiliary parameters
- reading of complete datasets with all running and auxiliary parameters
- exporting of data: time/ND, vel/ND, vel/flux, etc

Introduction to LabView

Dataflow programming and virtual instruments

LabView [1] (Laboratory Virtual Instrument Engineering Workbench) from National Instruments is a graphical development environment, that is, it relies on symbols rather than text based languages to describe programming actions. LabView is a product which is primarily aimed at scientists, engineers and alike for simulating different types of instruments. The programs in Labview are called virtual instruments or VI's.
A virtual instrument is a combination of a data-acquisition system and a processing system—the computer in this case. The processing involved is soft coded as a program rather than as complex circuitry. The advantages are obvious—cost for one.

The same hardware (DAQ + computer) can act as different instruments without any additional overhead of cost. Flexibility in experimentation—changing the parameters of instruments is as simple as clicking on some button, when dealing with real instruments, things are much more difficult.

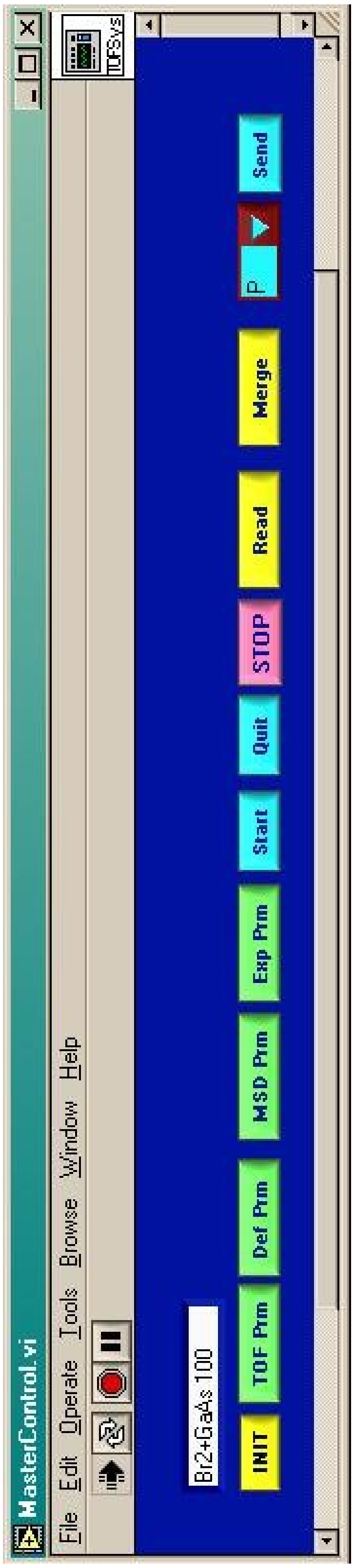
The graphical development environment uses the concept of data flow programming. Most programming students are exposed to the control flow paradigm of programming. In control flow programming, the sequence of the program is defined by the flow of control through the program. In contrast—in a data flow paradigm, the sequence of execution is determined by the flow of data. In LabView, graphical components that are familiar to technicians and engineers are used and the flow of data is implemented by just joining these components by means of "wires". This methodology has some inherent drawbacks, for example, it lacks flexibility provided by a text-based language say C or C++, but it has the advantage of greatly reduced development time and effort when used by adept hands. This is an extremely important characteristic—as the intended user is more likely to be an experimentalist than a dedicated programmer.

Layout of TOFSYS program

The remainder of this chapter will look at some of the key VI's that make up the TOFSys program.

The full range of VI's is shown in the two-hierarchy diagrams Figures 4.1a and 4.1b. The discussion will be limited to the important higher level VI's.

The TOF system is controlled from a master VI. This control panel has the all the necessary buttons to set the parameters and control acquisition. See Figure 4.2



Init	Initiates the program sending the default parameters for TOF control and acquisition
TOF Prm	Sets manually the parameters for TOF control and acquisition
Def Prm	Sets automatically the default parameters for mass spectrometer/detector
MSD Prm	Sets manually the parameters for mass spectrometer/detector
Exp Prm	Sets manually the experimental parameters
Start	Starts collecting data
Quit	Quit the program stopping all processes
Stop	Stops the program
Read	Read of complete datasets with all running and auxiliary parameters
Merge	Amalgamation of datasets
Send	Sends commands manually in debug mode

Figure 4.2 Front panel. Controls the communication and the run process.

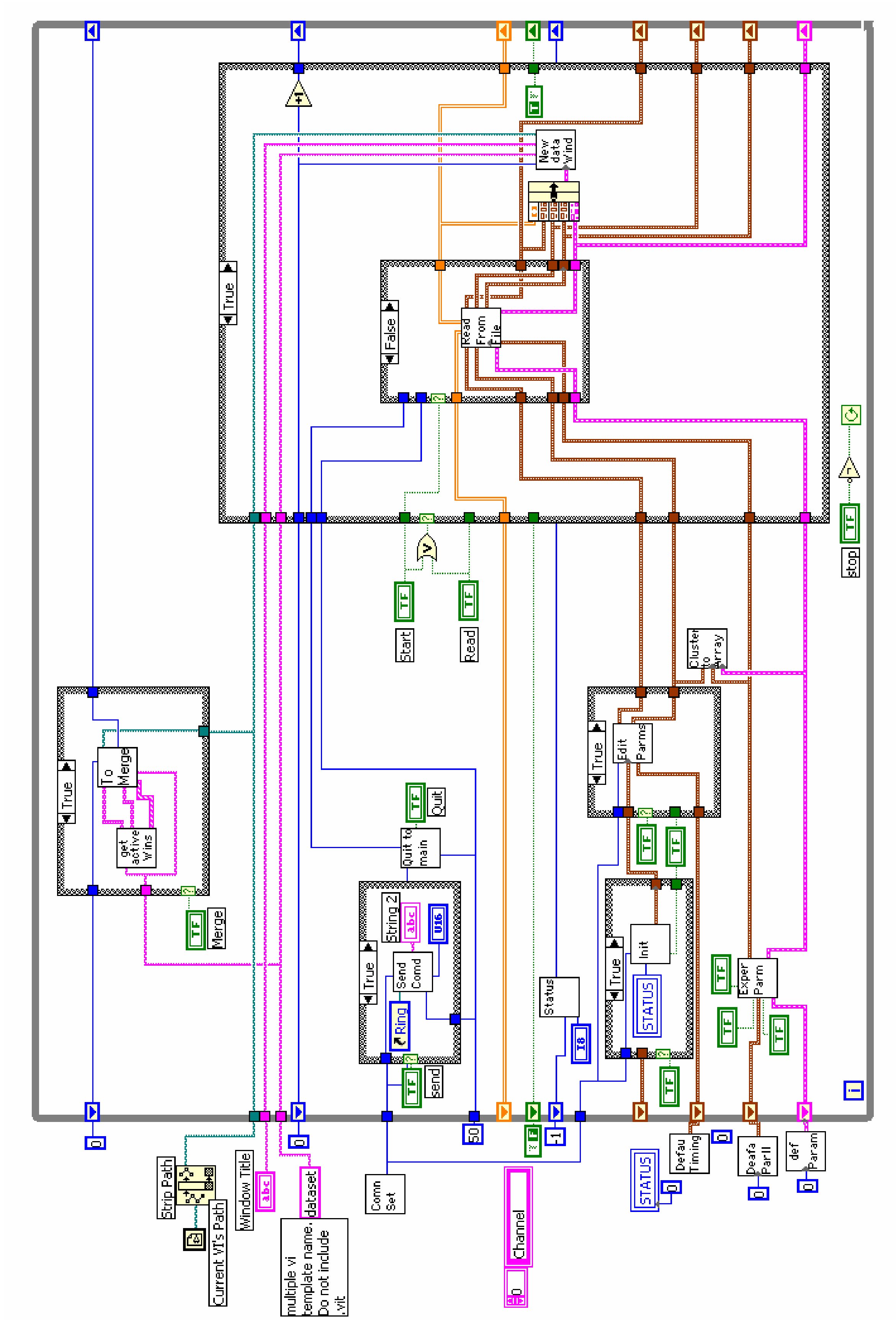
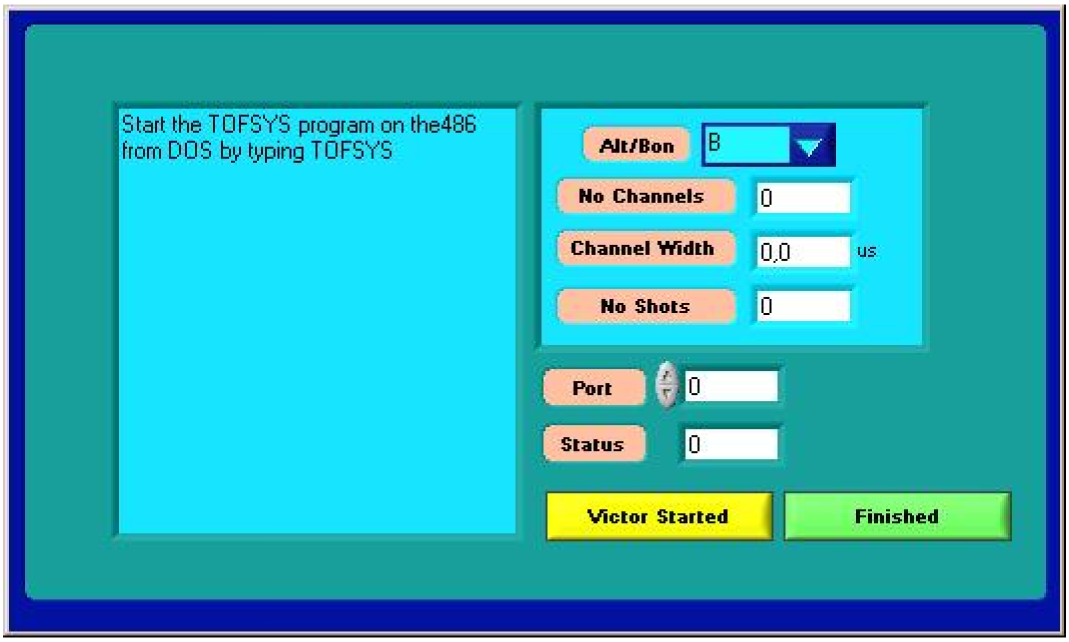


Figure 4.3 Diagram of the front panel.



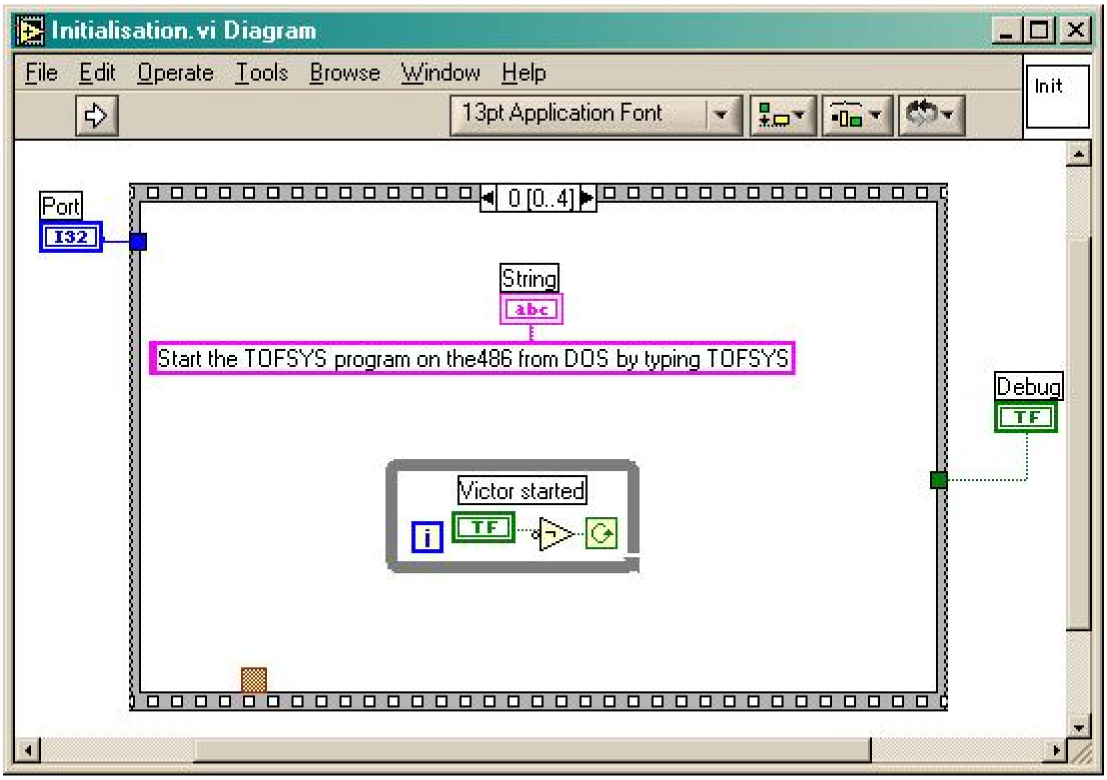


Figure 4.4 and 4.5. The running parameters settings and the diagram showing the first step of the sequence.

The initialisation button sends the initial commands and parameters to communicate the TOFSys program with the TOF INTERFACE (appendix 1) into PC 486 MS-DOS based computer. As it is shown initially it is sent a cluster of default parameters of number of shots, channel width and number of channel and the option of Beam On/Beam Off (alternative, A) or just Beam On (B). The port has been set internally.
The 'Initialisation' has the appearance shown in Figure 4.4. The running parameters settings sends the initial parameters to the 486-PC and starts the operating program, see Appendix 1. This is performed through of the 4-step sequence in Figures 4.5, 4.6, 4.7 and 4.9, starting with the button to set the default parameters (Victor Started).

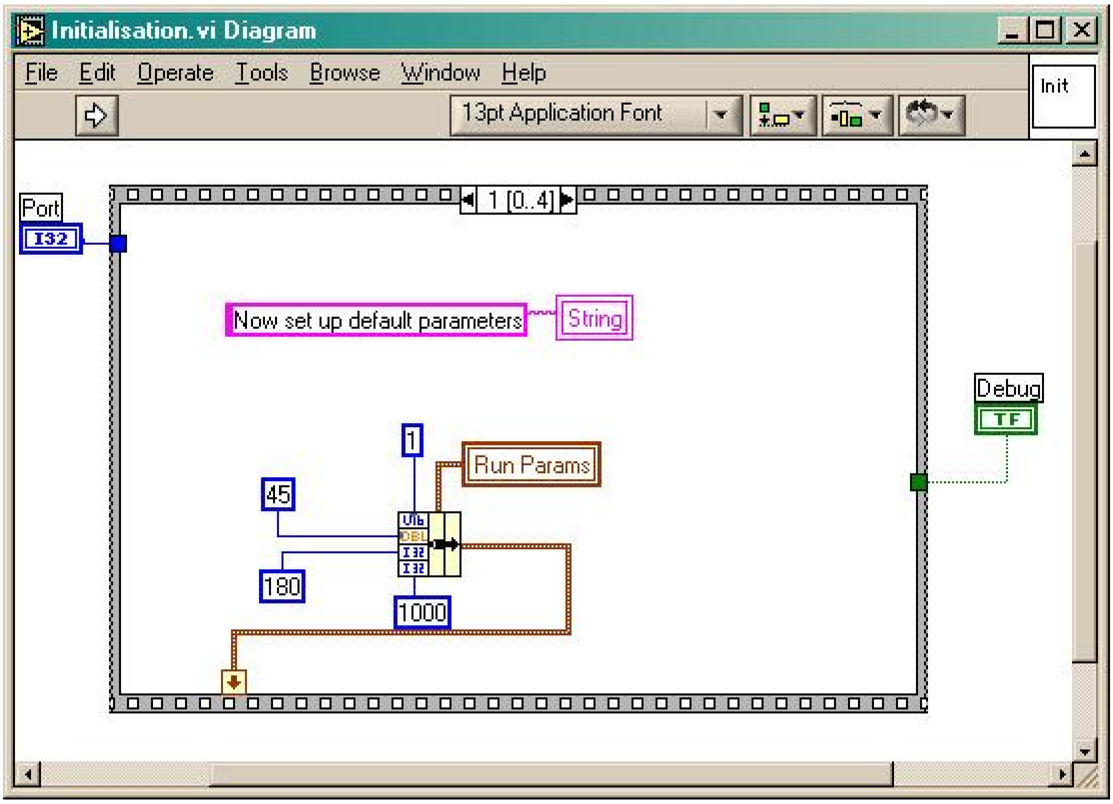


Figure 4.6 Second step at the sequence sending the initialisation running parameters.

After that, clicking on 'Finished' we are sending the parameters to the PC-486 to communicate with the TOF interface (appendix 1) which is waiting for a command. To send the parameters the command is "S".

At the second step of the sequence the default running parameters are set.

The third step, Figure 4.7, shows how the parameters are going to be sent with the "S" command through a subvi called 'Send and Reply' see Figure 4.8. The Send+Reply subvi is the heart of the RS232 communication. It sends a single character, a carriage return and waits for a single character reply from the 486-PC to show it is ready for a further character.

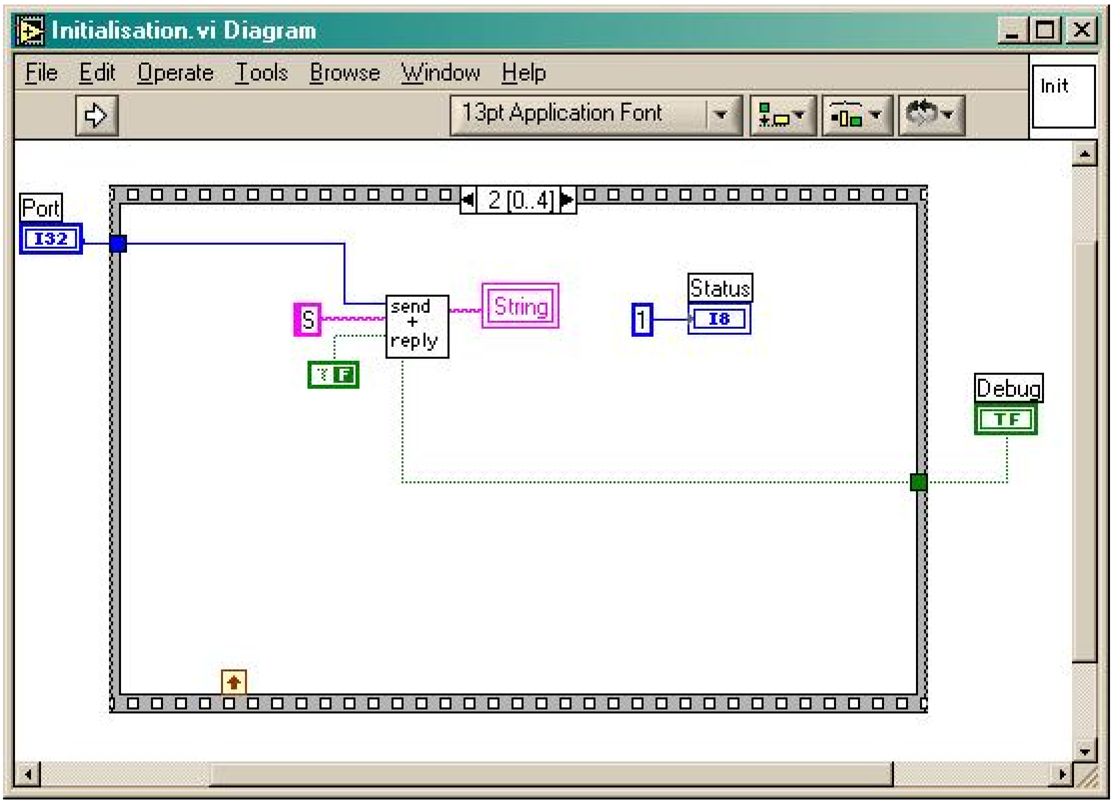


Figure 4.7 Third step for the initialisation running parameters.

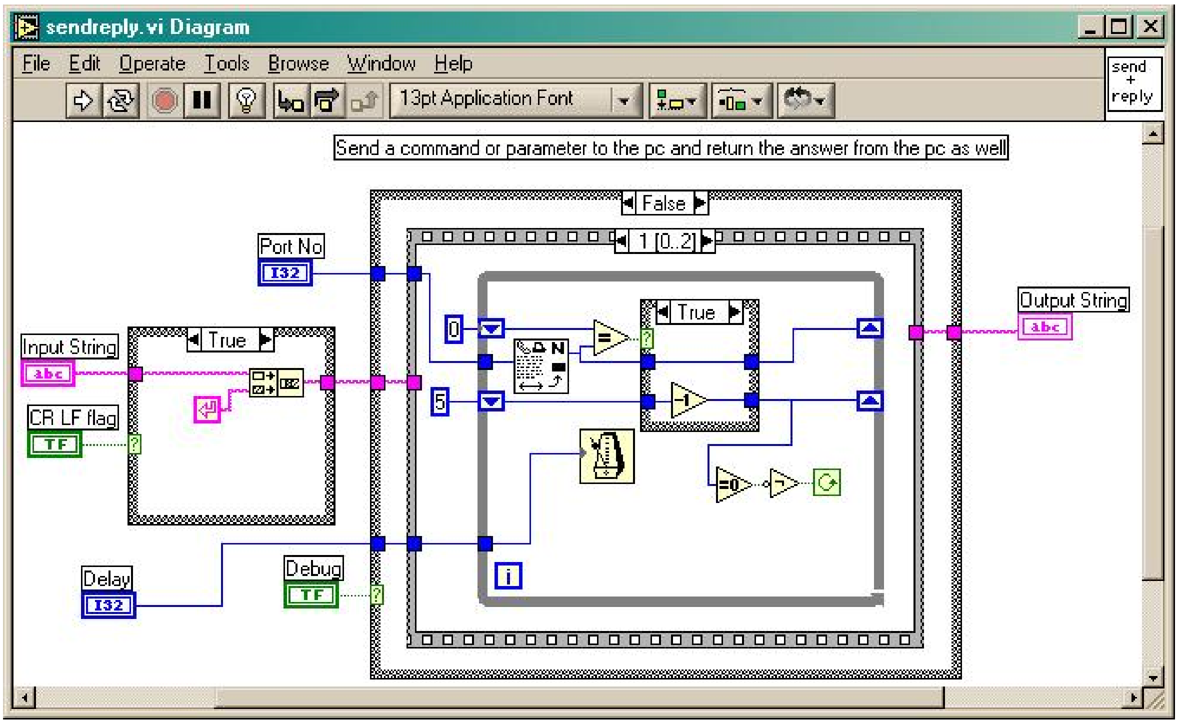


Figure 4.8 The Send+Reply subvi is used each time we want to send any character to the PC-486.

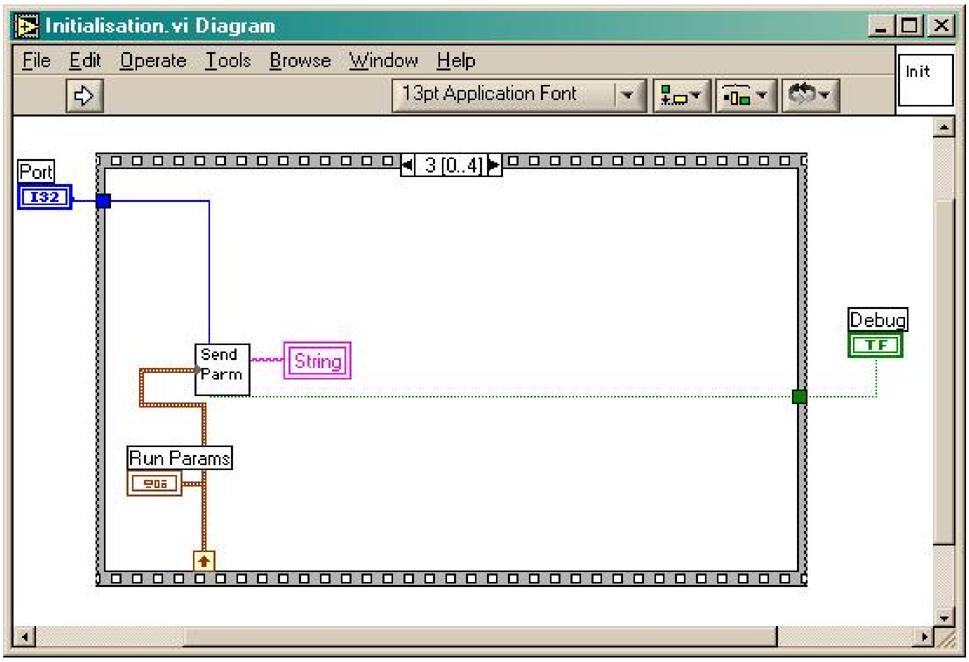


Figure 4.9 The fourth step finally sends the parameters which set in the LabView VI.

The subvi 'Send Parm' in Figure 4.9 sends the parameters one by one, character by character waiting 50 milliseconds between each one. See diagram in Figure 4.10.

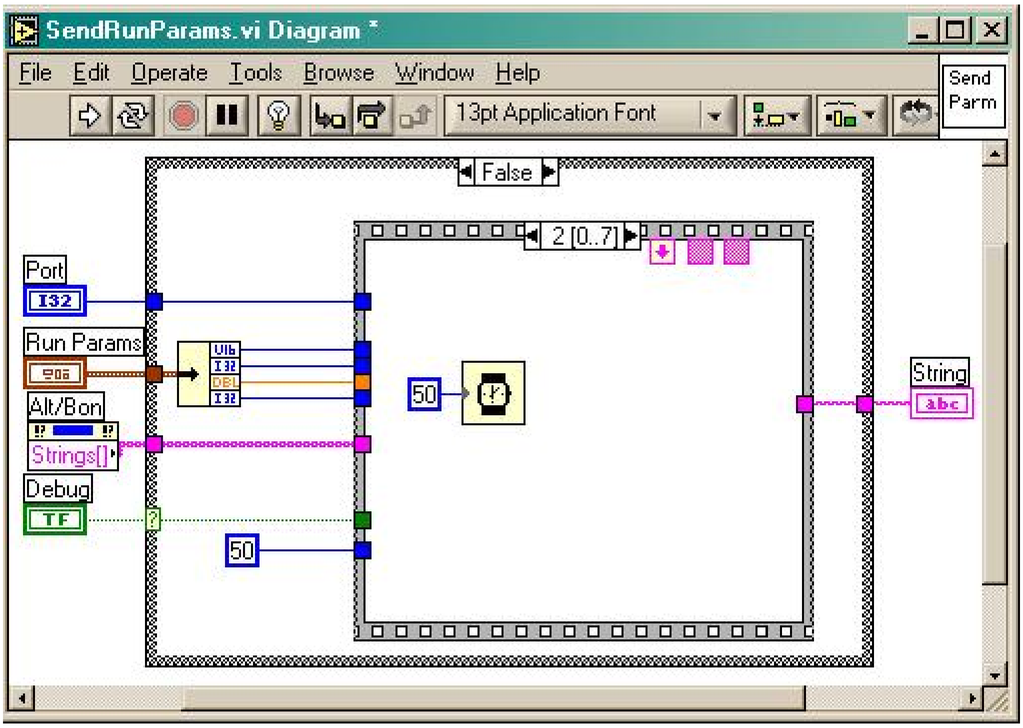


Figure 4.10 Sending parameters waiting 50 ms between the next one.

The last step of the initialisation sequence automatically closes the window after sending all the parameters.

After the initialisation we have to set and send the parameters to the TOF TIMING UNIT. These parameters are routed through the PC-486 to the TIMING UNIT. Also we have the ability to make further changes the TOF INTERFACE parameters. The subvi changing the TOF running parameters for the TOF INTERFACE and the TOF TIMING UNIT is shown in Figure 4.11. We also have the option to save the complete parameter sets for use later.

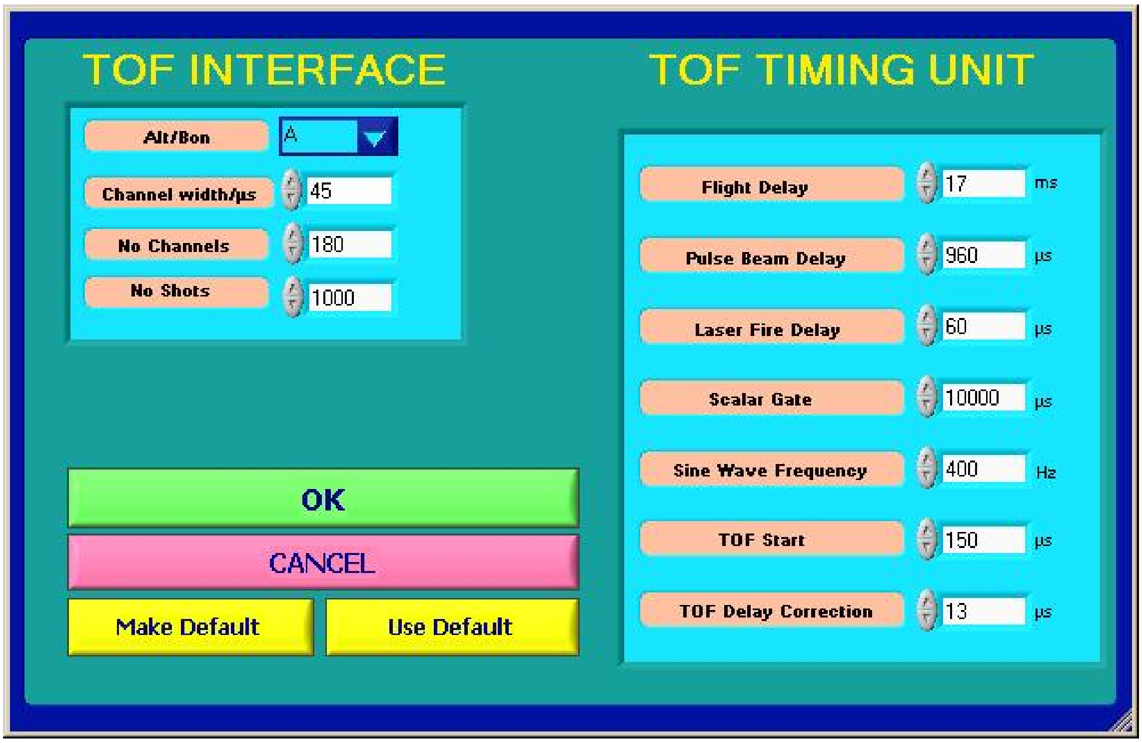


Figure 4.11 Editing running parameters and setting TOF Timing Unit

The internal structure of this subvi is as shows the Figure 4.12. It provides options to save/read the parameters. The subvi 'Send TOF-P' sends the parameters to the 486-PC to set the TOF controllers. This currently echos every character and a faster routine in the 486-PC would improve the transfer time.

With the parameters set we are ready to start the experiment.
Communication between the pc is made through the port RS-232 with a baud rate of 9600 with no parity. The Figure 4.13 shows all the parameters for a proper connection.

First we choose what kind of experiment we are going to do. Figure 4.14. The 'Def Prm' button puts up a window with 6 experimental options, which save time since each option is linked to a cluster of default parameters for both experimental and mass spectrometer/detector setting.

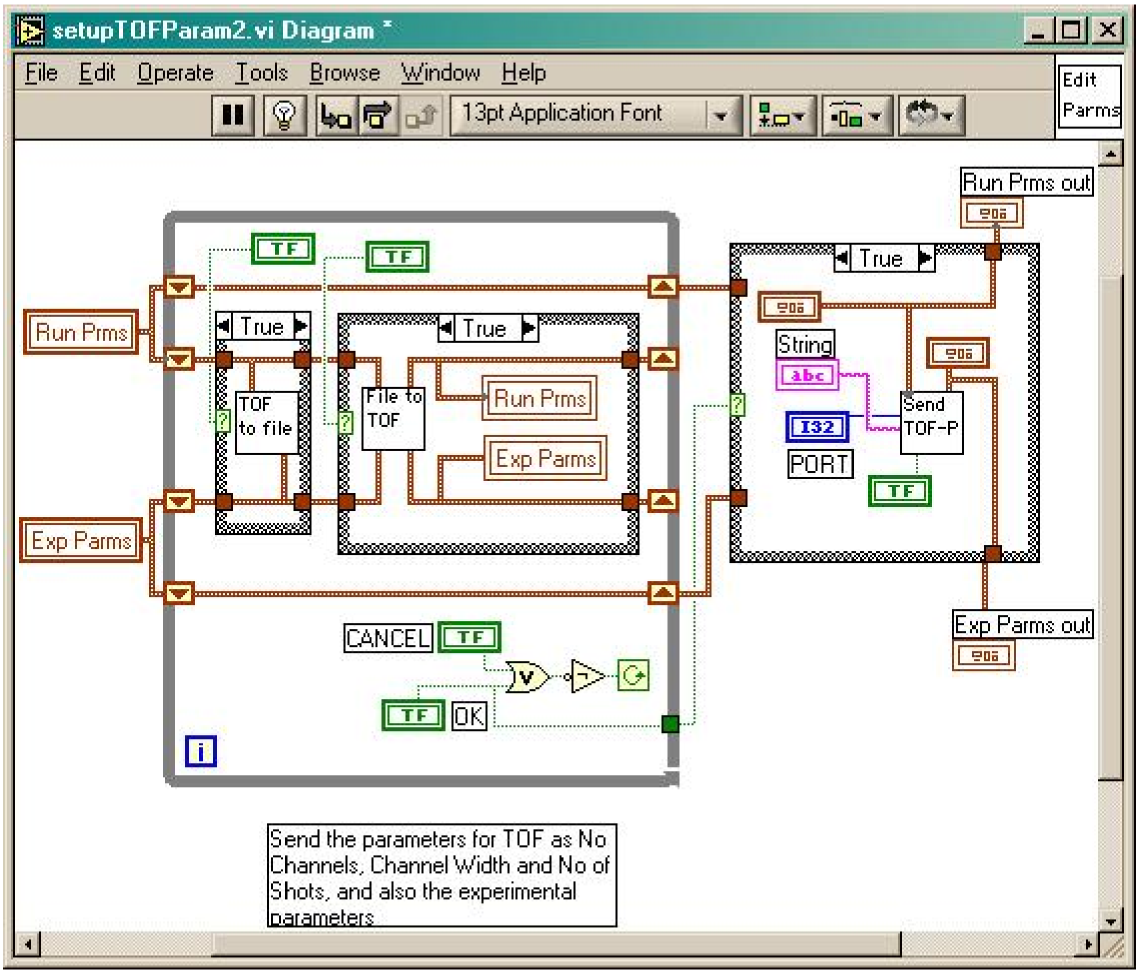


Figure 4.12 Diagram for the subvi to edit/save/read/set running parameters.

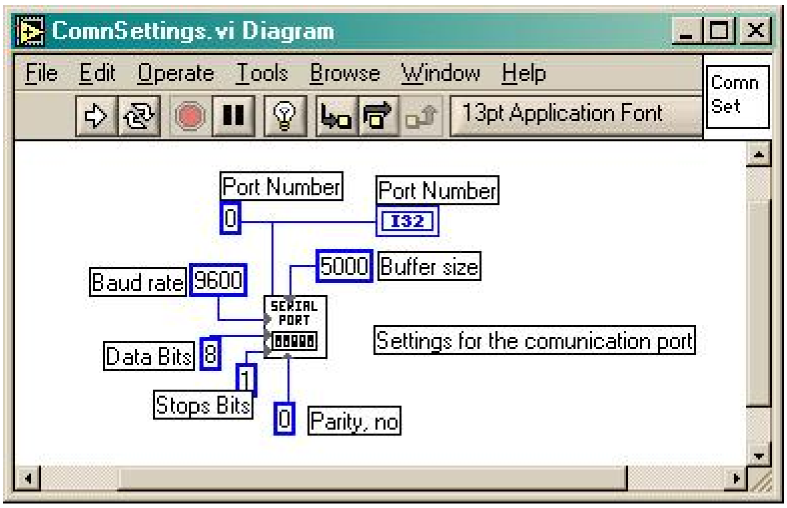


Figure 4.13 Communication settings

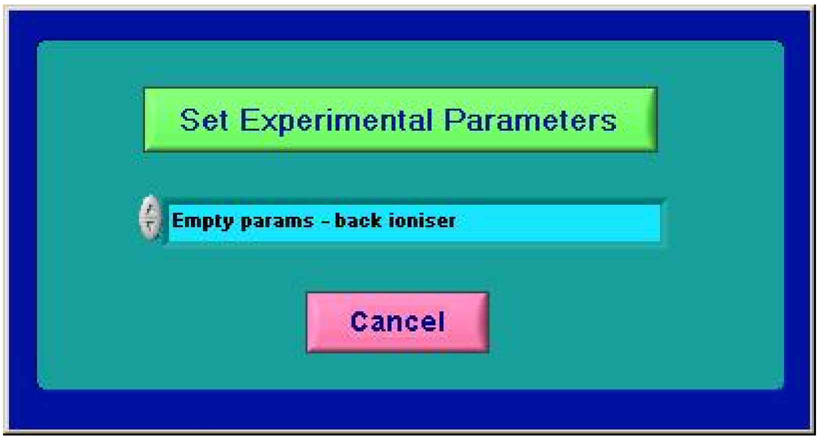


Figure 4.14 Subvi to set the default parameters depending on the type of the experiment being performed.

The six options are depend on whether we are using the back ioniser or the front ioniser and also if we are making a direct beam observation or performing a reactive scattering experiment. The other two options set all parameters to blank:

Empty parameters – back ioniser
Direct beam observation – back ioniser
Reactive scattering – back ioniser
Empty parameters – front ioniser
Direct beam observation – front ioniser
Reactive scattering – front ioniser

Cancelling close the window does nothing

The mass spectrometer/detector parameters are set manually using the MSD Prm button. This produces the window shown in Figure 4.15

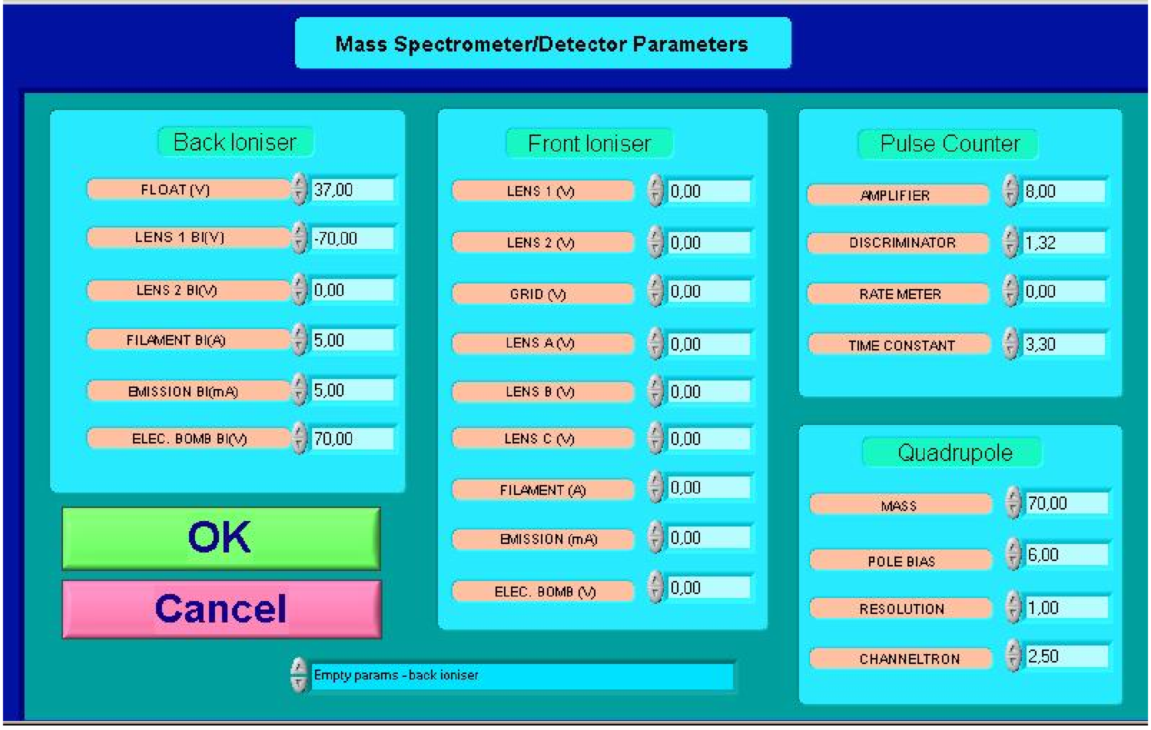


Figure 4.15 Window to set the Mass Spectrometer/Detector parameters

We see both the back and front ioniser options, the pulse counting parameters and the quadrupole control settings.

For the experimental parameters, where we store information about the pressures in the chamber, kind of surface, temperatures, ratio of mixed beam, etc. as shown in Figure 4.16.

Those parameters are additional information linked to the collected experimental data. We once again can save those parameters and read them as required.

Clicking the “Start” button initiates data collection. A progress bar, Figure 4.17, is displayed with several options. The “Pause” option stops the data collection and clicking “Continue” it restarts it. Aborting stops the data collection and no continuation is possible. If we are interested on the collected data before the abort we can get it by clicking the “Get” button.

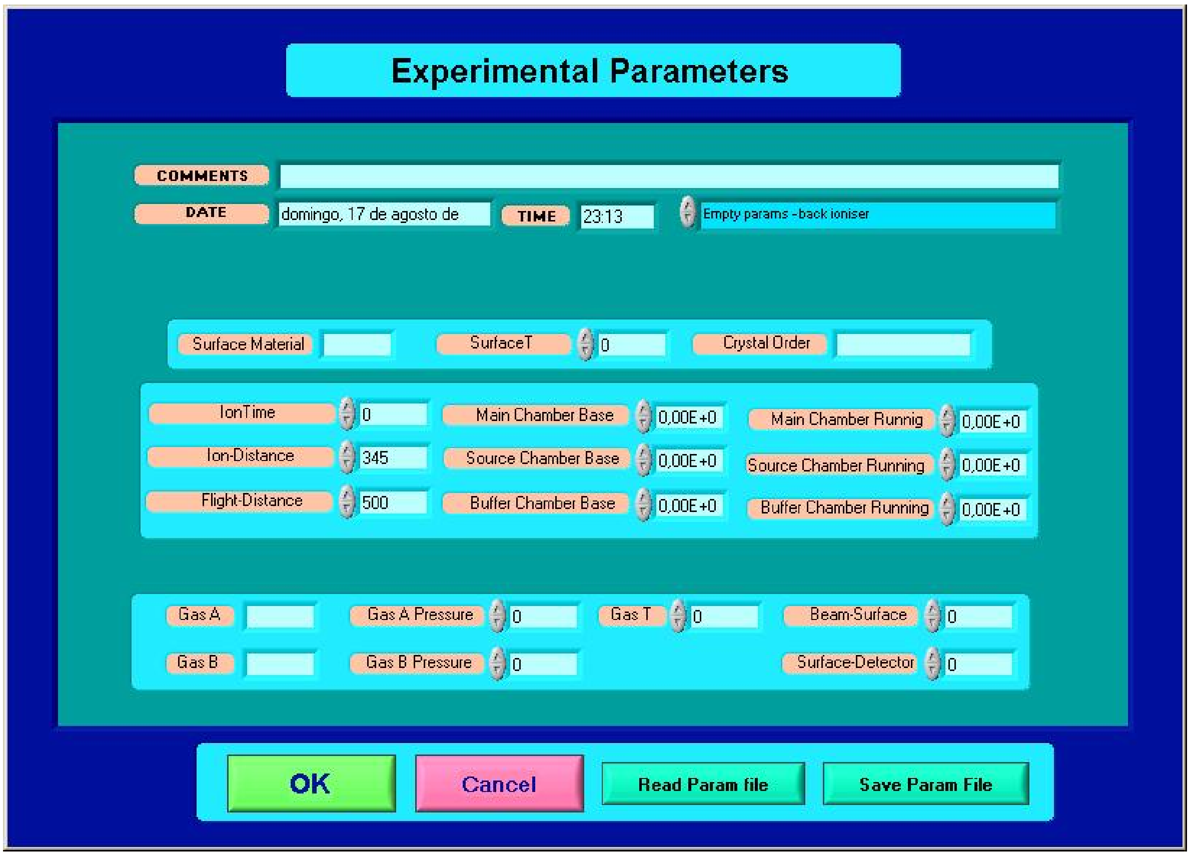


Figure 4.16 Storing the experimental parameters. We can save them to a file and read from a file.

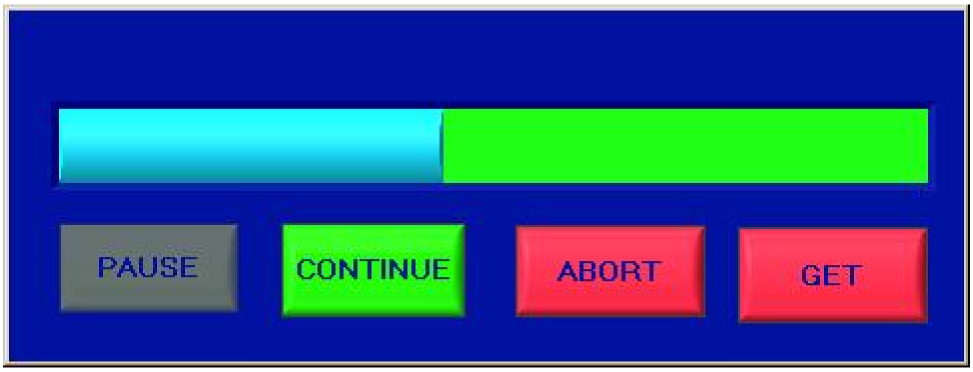


Figure 4.17 Progress bar collecting data with several options

In the diagram for this subvi, Figure 4.18, clicking the 'Start' button sends 'S' starting the collection. The 'Program Collect' subvi (the progress bar itself) responds to the pause, continue, abort and get buttons.

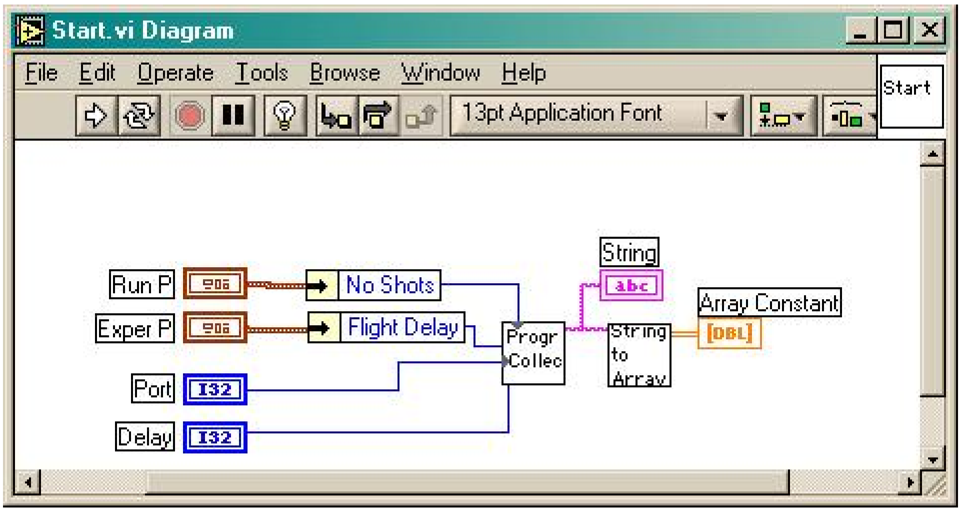


Figure 4.18 Diagram for the progress collecting data bar

When data collection is complete the data is transferred from the 486-PC to the Labview program as 16 bit integer in hexadecimal format (for speed). These are converted to numbers for data processing by the 'String to Array' subvi.

We can also retrieve saved data via the 'Read' button as shown in Figure 4.19. The computer will request a path to where the data are saved.

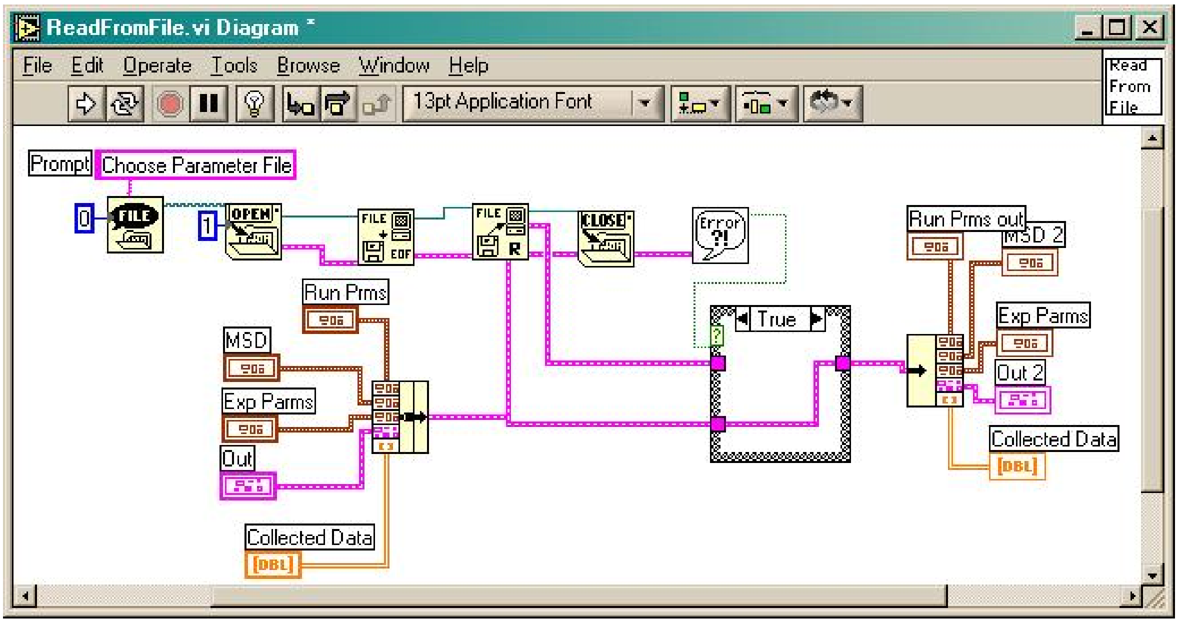


Figure 4.19 Diagram for read data from a file. The one to save the data is a similar diagram.

Data file Structures

Once the data are received the main data display window is brought up. This displays key parameters and graphs for Beam On, Beam Off and Beam On minus Beam Off. The collected data is also listed in a table showing the channel number, Channel A, Channel B, Time, velocity, the corrected time, the normalised signal, the number density and the flux.

The true TOF distribution must now be transformed to a velocity distribution [2]. If T represents the total flight time measured by the detector as the sum of the molecule flight time T_n taken for the neutral molecule to traverse the distance from the chopper disc to the centre of the ioniser D_n and the ionised molecule flight time T_i (ion time) taken for the resulting ionised molecule to traverse the distance from the centre of the ioniser to the exit of the quadrupole mass filter D_i, then:

$$T = T_n + T_i = \frac{D_n}{\nu_n} + \frac{D_i}{\sqrt{\nu_n^2 + \nu_i^2}} \quad (4.1)$$

where ν_n is the velocity of the neutral molecule, ν_i is the ion velocity.

Thus the velocity distribution may be obtained from the TOF spectrum using the following algorithm [3,4]:

$$T_n = nT_w + \frac{1}{2}T_w + \Delta T \quad (4.2)$$

$$\nu_n^0 = 0$$

$$T_n^{k+1} = T_n - \frac{D_i}{\left[\left(\nu_n^k\right)^2 + \nu_i^2\right]^{\frac{1}{2}}} \quad (4.3)$$

$$\nu_n^k = \frac{D_n}{T_n^k} \quad (4.4)$$

where T_w is the TOF channel width, n is the number of TOF channels, ΔT is the chopper disc reference pulse delay, T_n is the 'true' TOF in the n^{th} channel for the neutral molecule, k is the iteration index (k = 1~5). After typically five iterations, the velocity of the neutral molecule in the n^{th} TOF channel ν_n is obtained.

The number density velocity distribution is then given by:

$$N(\nu) = h(t)\left\|\left[\frac{dt}{d\nu}\right]\right\| \quad (4.5)$$

where the Jacobian is calculated from equation 4.2. Then we have:

$$\frac{dt}{d\nu} = -\frac{D}{\nu^2} \quad (4.6)$$

where D is the neutral molecule flight distance between the slot on the disc and the centre of the ioniser; ν is the neutral molecule flight velocity traversing the distance D.
Thus the velocity distribution in number density is given by:

$$N(\nu) = -\frac{h(t)}{\nu^2}D \quad (4.7)$$

Then the corresponding velocity distribution in flux is given by:

$$I(\nu) = \nu N(\nu) \quad (4.8)$$

In order to display multiple data set we make use of an advanced feature introduced in LabView 6. This allows us to create a VI dynamically based on a template. This means that we can create a VI for each dataset collected, complete with its own controls, collection and processing parameters, file handling and graphic options.

Each dataset window is 'live' allowing all processing, file and printing options to be carried out. The data remains live until the 'Close' button is used to remove it.

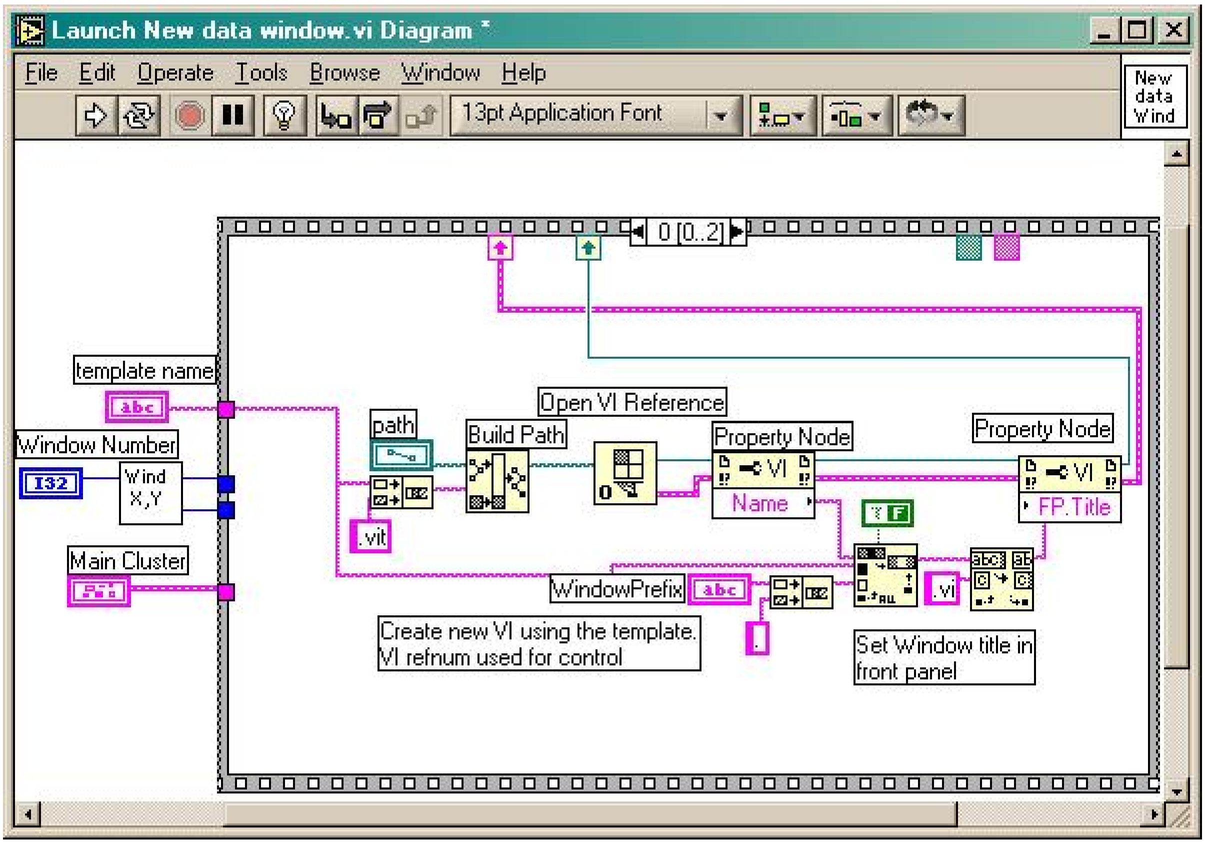
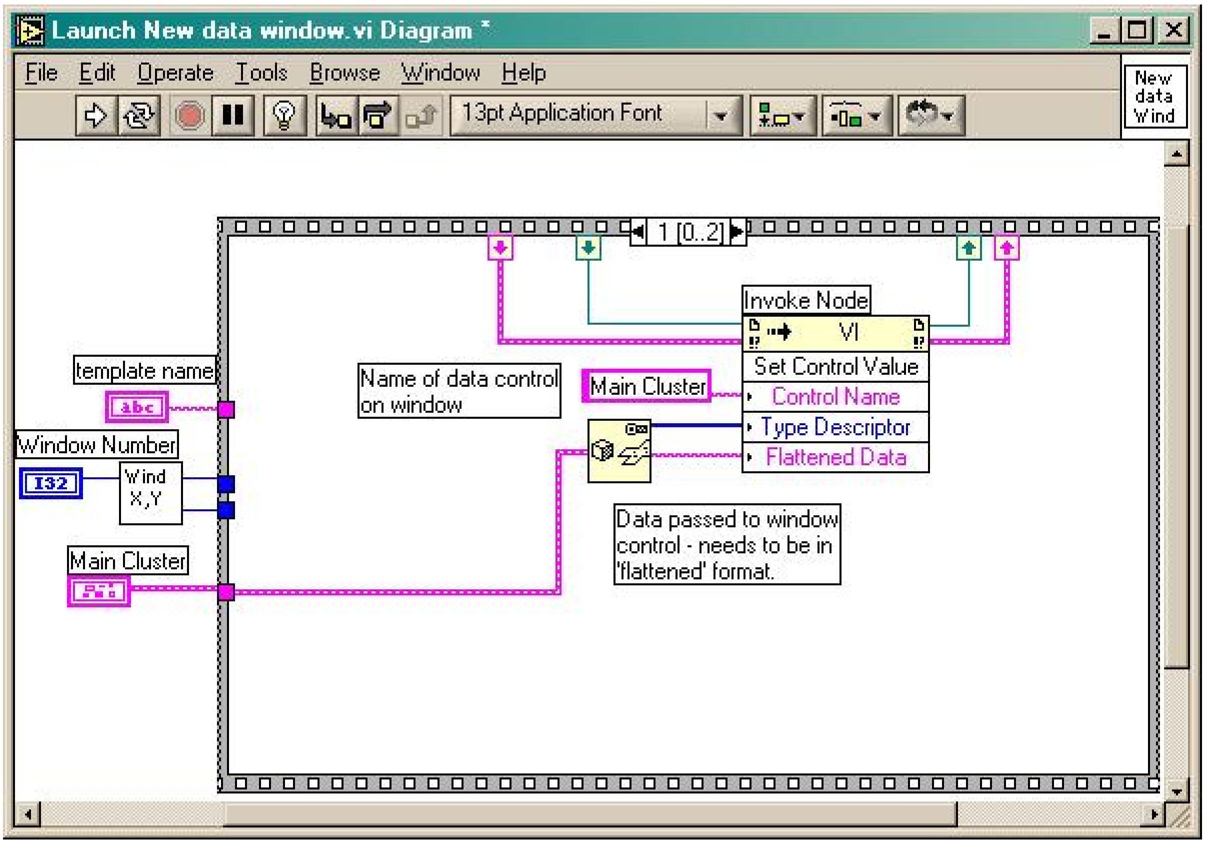
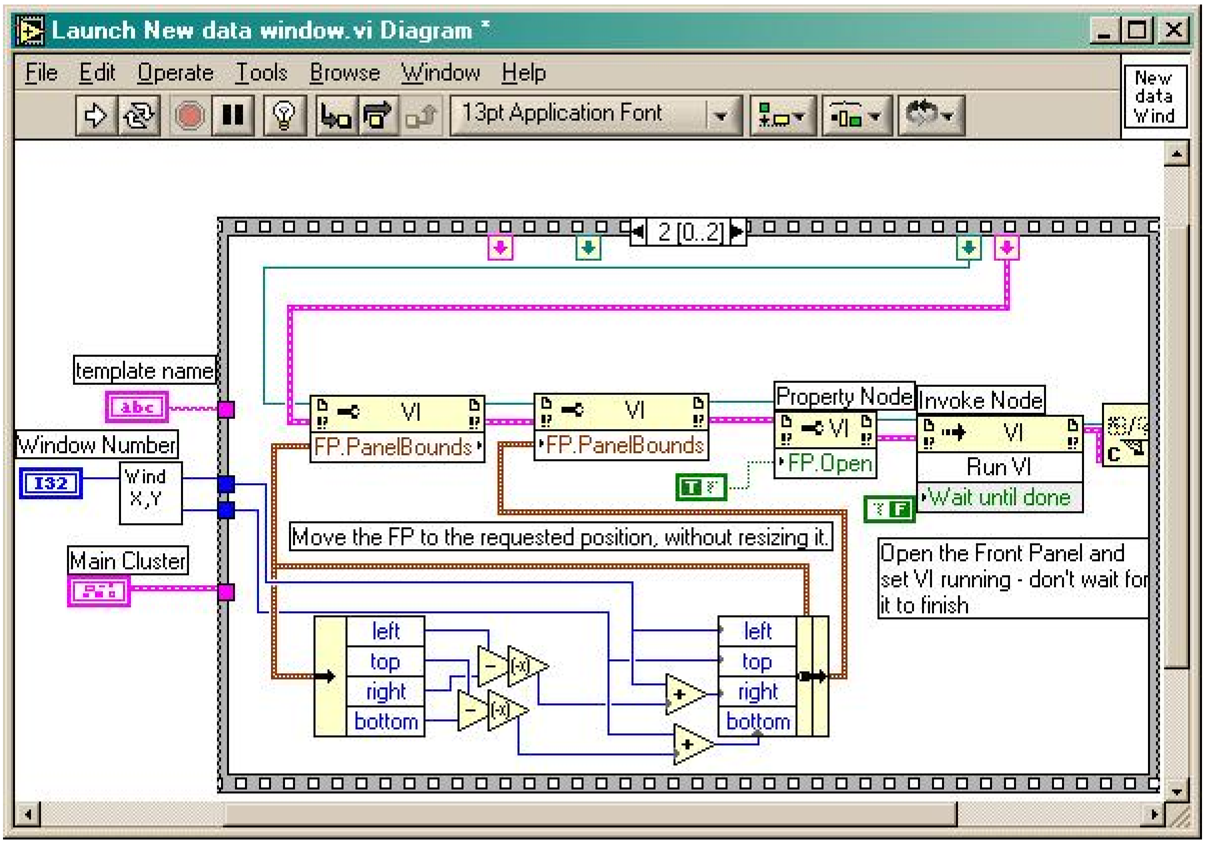


Figure 4.20 Diagram for the firs step launching a new data window





Figures 4.21 and 4.22 Second and third steps to launch a new data window.

The diagram to launch the window is as follow: it is a three-step sequence where the first one creates the new subvi using the template. See Figure 4.20.

The second one sets the name for the data control window. See Figure 4.21.

The last step opens the front panel and set the subvi running and allocates the window to a preset position. See Figure 4.22.

The appearance if the data window is as showed in Figure 4.23

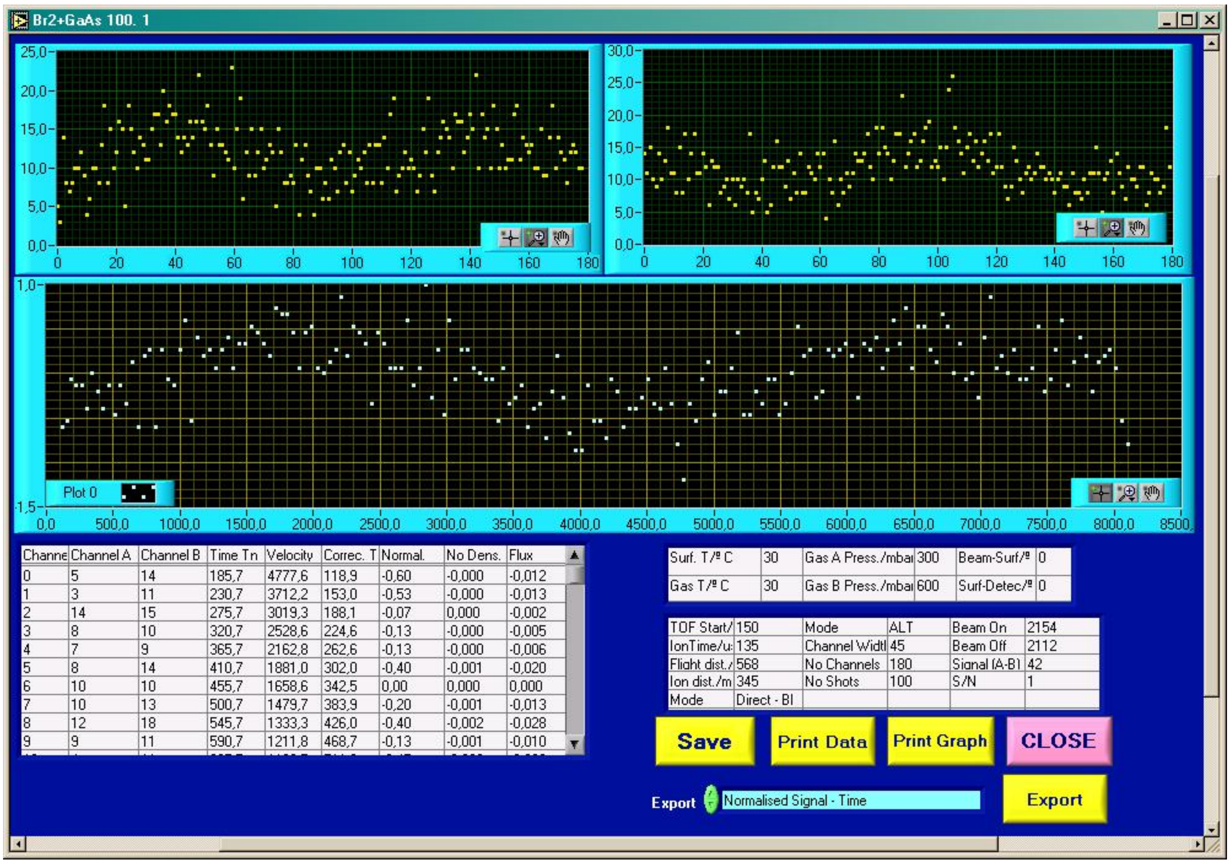


Figure 4.23 Window showing collected data and key parameters.

The window shows the collected data in a scrollable table. The top left graph is for Channel A, the right top one is for Channel B, and the bottom one is the subtraction Channel B – Channel A. Where the Channel B is the signal plus background and the Channel A is the background.

The graphics on the window have options for the zooming and all the normal properties provide by the LabView system. The subvi also saves the data set recording all the data, graphs and parameters. We also have the print option for both data table and graphs. Another important facility is the export option to export the data in spreadsheet format.

The exporting options are:

Normalised Signal – Time
Normalised Signal – Time (minus constant)
Number density – Velocity
Flux – Velocity
Raw Signal – Time
Raw Background – Time

The diagram for this subvi is as shown in Figure 4.24

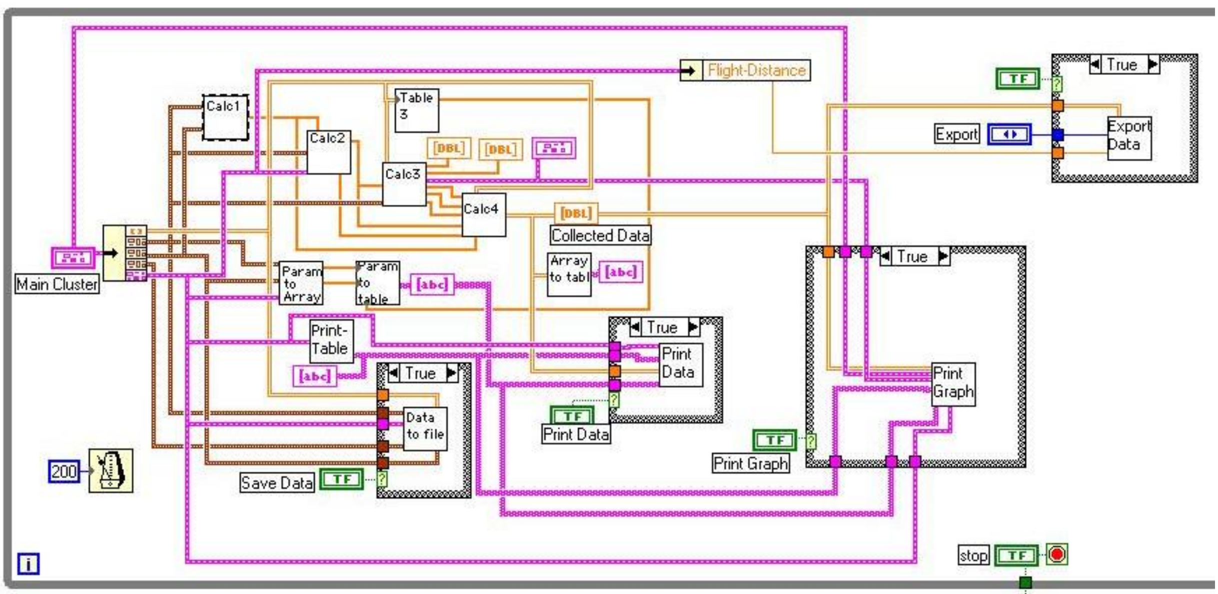


Figure 4.24 Diagram for the data window

The "Main Cluster" contains all the parameters and the collected data. The calculations vi's create the data table and the graphs. In order to display the values we convert the parameters to arrays of numbers and finally to a table. The subvi also provides options for saving, printing and exporting.

Multiple data windows

Finally we have the merge option shown in Figure 4.25. With the merge option we can add different multiple datasets from the same experiment. The datasets must match in all their parameters and settings. The merge option shows the available datasets in the left window and merges two or more (of the same kind) after adding to the right window. We can add and remove datasets to merge and produce as many new datasets as we want. A new window is created displaying the merged data.

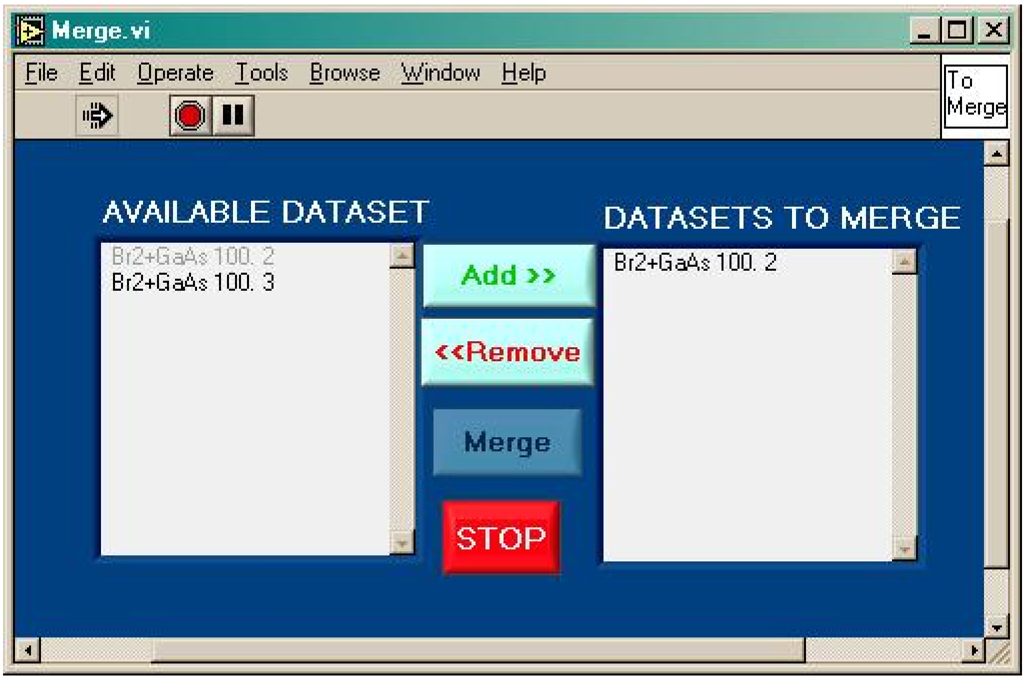
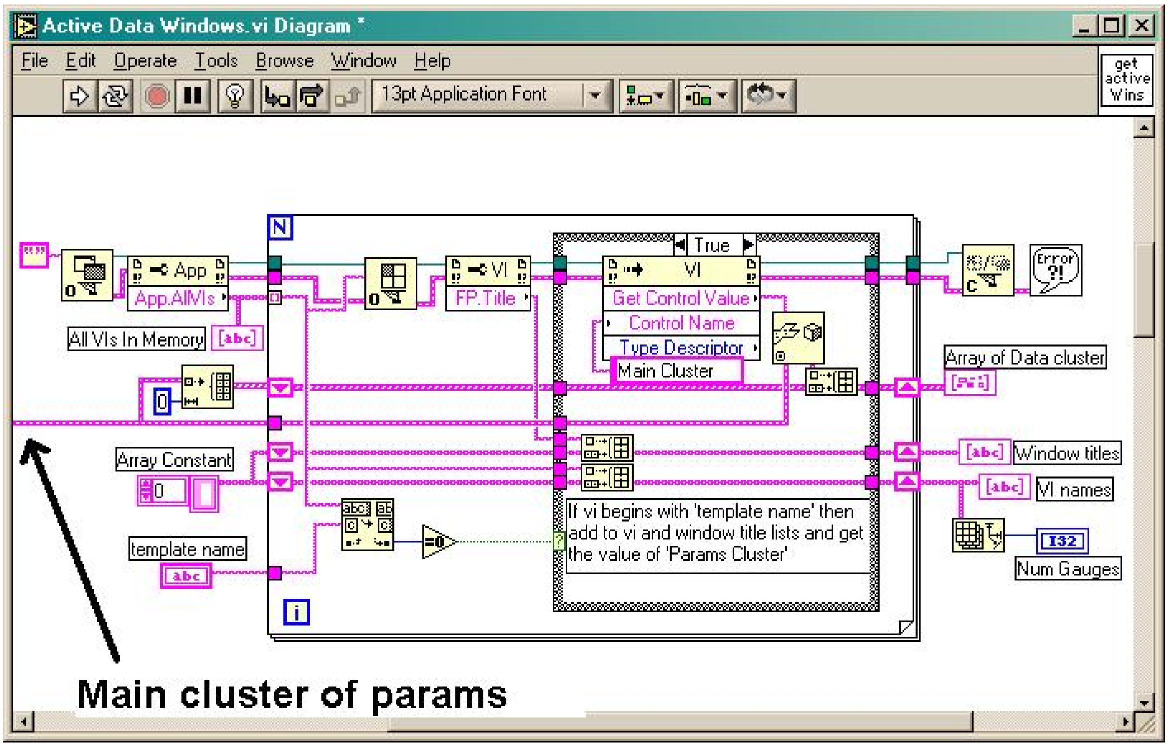


Figure 4.25 Window for the merge option



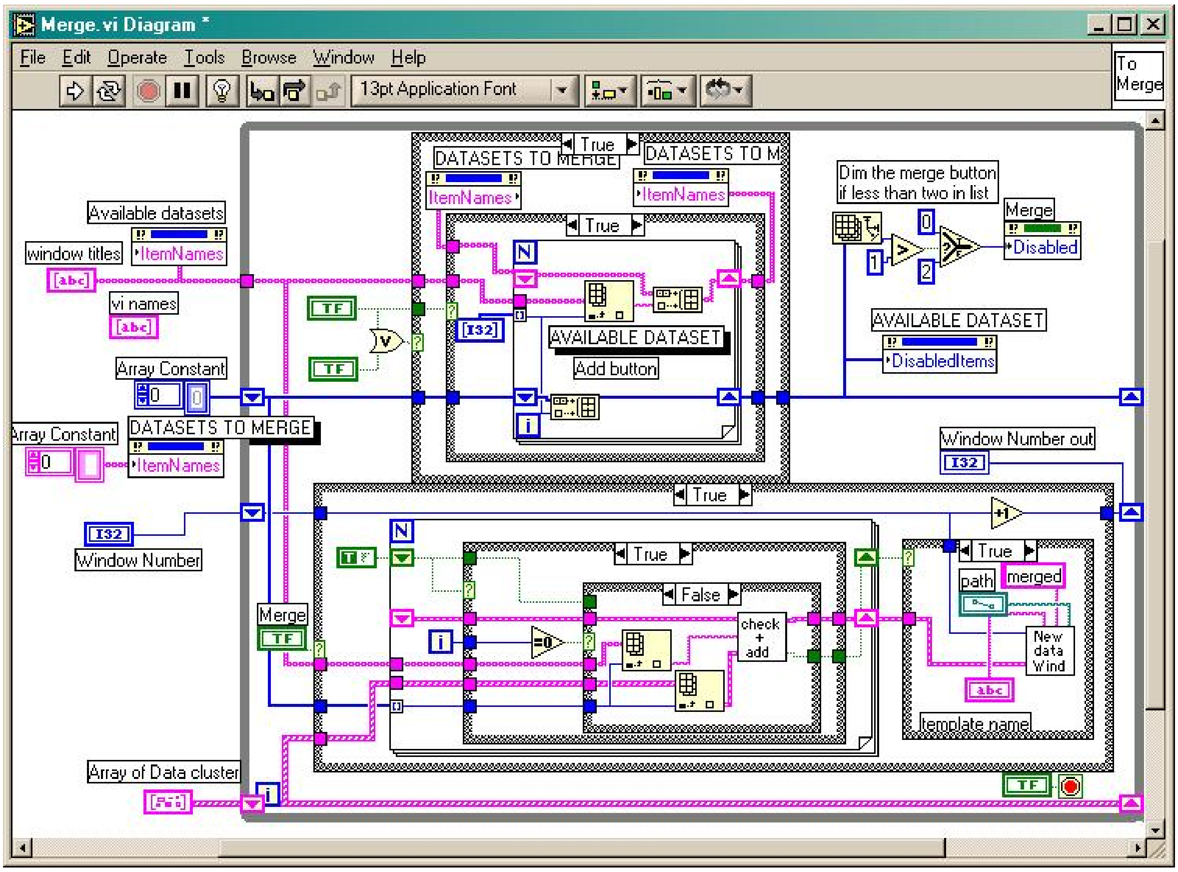


Figure 4.26 and 4.27 Top: The diagram to get the active data windows. Bottom: Diagram to merge the data.

References

[1] www.ni.com

[2] Martin P. Hall, M. Phil. University of Manchester, 1997

[3] P. A. Gorry; R. Grice; J. Phys. E: Sci. Instrum., 1979, 12, 857

[4] D. St. A. G. Radlein, PhD. Thesis (Cambridge), 1975