Using Igor as user interface together with FAST TDC card for PEPIPICO data acquisition: manual and overview

F. Burmeister,[1] M. Gisselbrecht,[2] J. R. Piton,[3] E. S. Cardoso,[3] S. Sorensen,[2] and A. Naves de Brito[3]

[1] *Department of Physics, Uppsala University, Box 530, S-751 21 Uppsala, Sweden*

[2] *Department of Synchrotron Radiation Research, Institute of Physics, University of Lund, Box 118, S-221 00 Lund, Sweden*

[3] *LNLS, Box 6192, 13083-970,Campinas SP, Brazil*

(August 15, 2002)

An overview and manual for the data acquisition setup using FAST TDC 7886S card with Igor as a user interface is presented. The setup is intended to be used for Photo-Electron Photo-Ion Photo-Ion COincidence (PEPIPICO) measurements with a Time-Of-Flight (TOF) spectrometer.

Contents

I. INTRODUCTION

The aim of this report is to make the user familiar to the setup of the Time-Of-Flight (TOF) FAST TDC card, using Igor as user interface. The steps for starting up an experiment and presenting data are explained.

FAST Cards [1] were bought in 2001 for Arnaldo Naves de Brito (LNLS in Campinas, Brazil) and Stacey Sorensen (MAX-lab in Lund, Sweden). The coincidence (PEPIPICO) community has traditionally been using Igor [2] as the interface for presenting and acquisitioning data. The aim of this project has been to knit the FAST Card to Igor, and to make data acquisition and data presentation as user-friendly as possible.

A. spectrometer overview

In Fig. 1, the overview of a Photo-Electron Photo-Ion Photo-Ion COincidence, (PEPIPICO), set-up is shown. The concept is used for the LNLS and MAX-lab spectrometers. The synchrotron light γ comes from the upper side. The gas molecules comes from a diffusive gas jet, perpendicular to the light and the spectrometer axis (in Fig. 1 into the plane of the sketch, not shown here). The region where the synchrotron light and the gas molecules interact is called the extraction region.

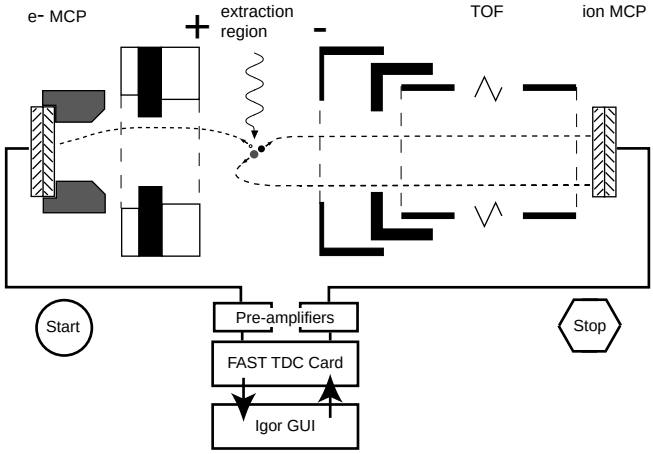

FIG. 1. PEPIPICO spectrometer overview.

1. The synchrotron light γ interacts with a molecule.

2. The molecule emits photo- and/or Auger electrons. The molecule might dissociate, with a probability of creation of one or several positively charged fragments.

3. The electrons are attracted by the positive (+ in Fig. 1) electric field of the grids on the left side of the extraction region, and hits the multi-channel plate, MCP.

4. The capture of a single electron triggers the MCP to produce a measurable pulse. The pulse is amplified and cleaned through discriminators.

5. The pulse triggers a "Start" in the FAST card.

6. Meanwhile, the remaining positively charged fragment(s) in the extraction region are attracted by the negative potential of the right side. These fragments fly through a field-free TOF, and hit upon the ion-side MCP.

7. The detection of a fragment generates a pulse just as the electron did, and triggers a "Stop" in the FAST card.

8. The time between the electron "Start" and the fragment "Stop" (one or several stops) is converted to digital information by the FAST time-to-digital-converter, (TDC) card.

9. The digital information is transferred to Igor grafical user interface, GUI, and presented in a user-friendly fashion.

The TOF contains information about the mass/charge ratio of the fragment. In Fig. 7, the two histograms are examples of TOF and its corresponding mass spectrum. Further, information about the fragment kinetic energy release during dissociation can be obtained. In Fig. 8, the middle right panel shows a double coincidence plot (singly charged oxygen ion and singly charged carbon atom from the molecule CO), and the shape of the feature contains information about the kinetic energy release. A detailed discussion about data treatment and physical information obtained regarding N_2O, and O_3 (Ozone) respectively, using the PEPIPICO technique, can be found in [3–5]. These measurements were done using precursors to the FAST TDC, namely a FLY TDC and a 4208 Multistop LeCroy TDC. More general information about PEPIPICO technique can be found in [6,7].

B. FAST TDC card

The LNLS and MAX-lab teams have bought FAST TDC cards used for time-of-flight treatment of pulses. The model chosen was P7886S. There are several reasons for choosing the FAST TDC card:

- 1 ns multistop resolution.
- Discriminator included in the FAST card. Other TDC cards need external discriminators, which reduces the multistop resolution to $\gtrsim$ 3 ns.
- The time window (`Range` in Fig. 2) can be set to as much as 64 seconds, which can be useful for clusters and large molecules.
- The high integration level with standard existing PC computers (Windows, Linux etc.). All tools needed for a home-made data acquisition set-up can be provided by the FASTComTec company.
- The possibility to buy FAST card version P7887 with 250 ps multihint resolution, with optimal price performance. The data acquisition set-up made for present card versions (P7886S) can be used with minor adjustments.

II. HOW TO START

The FASTIgor data acquisition setup works for operating system Windows2000NT. The DLL ("dynamic link library") package for the TDC card is required, and manufacured by the FASTComTec company.

Connect FASTCard cables: green for the electron start signal and red for the ion stop signal. When starting an experiment series, you need to start up the FAST Card server and Igor:

1. Double-click the shortcut P7886 on the desktop. Now you have the interface MCDWIN to the FAST Card, provided by FAST ComTec.
2. The P7886 Server is accessible, it was initialized by MCDWIN. Make P7886 Server window active from taskbar at the lower edge of Windows.
3. Under the menu `Settings`, choose `Hardware`.
4. Set the P7886 Settings parameters and flags the way they are shown in Fig. 2. `Write Listfile` option has to be checked, because then the card will create the `C:/Program Files/P7886/DATA/spec2.lst` list, which will be sorted later. `Range` value

can be changed to a smaller or larger time window.

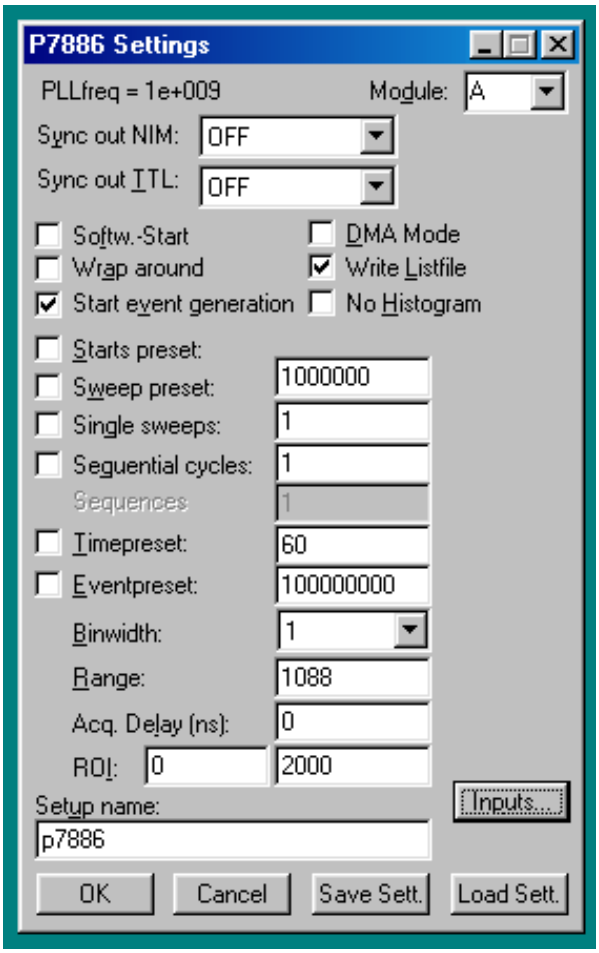

FIG. 2. Recommended P7886 settings.

5. Push button `Inputs...` in P7886 Settings window.

6. In Fig. 3, our choise of discriminator thresholds is shown. Here we use the internal discriminators of the FASTCard. However, if the pre-amplified pulses are potentially too strong (>2.5V), then a setup with external discriminators should be used. The thresholds in Fig. 3 are valid only for the internal discriminator setup.

7. Close windows `Input Threshold` and `P7886 Settings`.

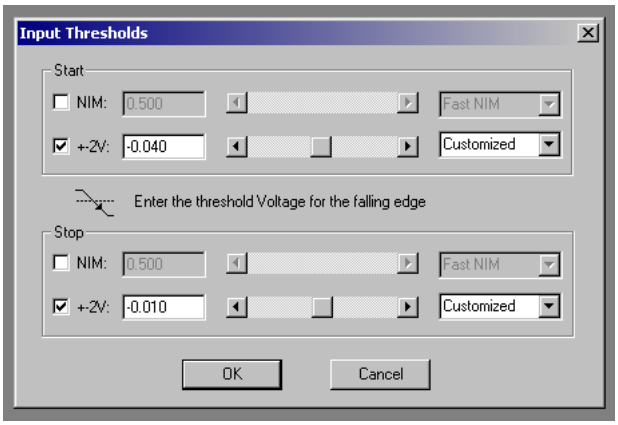

FIG. 3. Recommended input thresholds.

8. In P7886 Server window, choose `File:Data` in menu. Choose parameters as shown in Fig. 4.

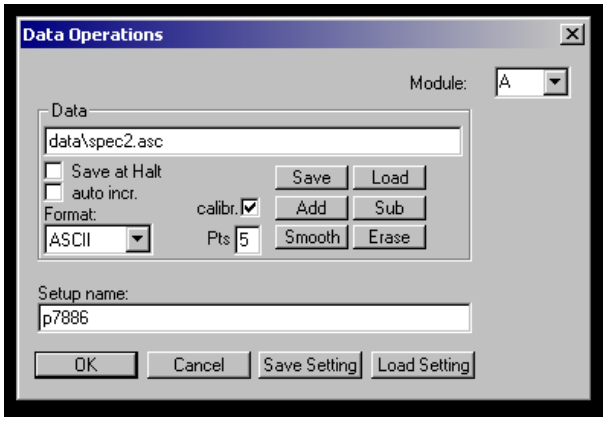

FIG. 4. Recommended Data settings.

9. Run FAST Card by pushing Start Button at the upper right panel (▷) in

MCDWIN. Click `yes` in possible overwrite/erase file check windows. If the FAST Card is working, you will see a growing histogram.

10. Stop FAST Card acquisition at the upper right panel.

11. Check whether folder `FASTIgor` (including FASTIgor_xxx.ipf and Coincidence_xxx.ipf) and shortcut to FASTIgor_MainControl are in place (upper two windows in Fig. 5, the required addresses are shown). Otherwise, place the FASTIgor folder (provided as freeware by the authors) and shortcut as shown.

12. Check whether `FASTIgor.xop` (or shortcut to `FASTIgor.xop`) and `tst7886.exe` are in place (two middle windows in Fig. 5, the required addresses are shown). Otherwise, place the `FASTIgor.xop` and `tst7886.exe` as shown. They are also included in the freeware package of the authors.

13. Check whether `dp7886.dll` is in the right folder: `C:/WINNT//System32` (lower window in Fig. 5). If not, you can find it in folder `C:/Program Files/P7886`, if you have bought the DLL program package of the FAST-Card company. Rename the original in `C:/Program Files/P7886` to `dp7886.dl_`

14. Check that all files in Fig. 5 are unique, throw away all obsolete copies of files.

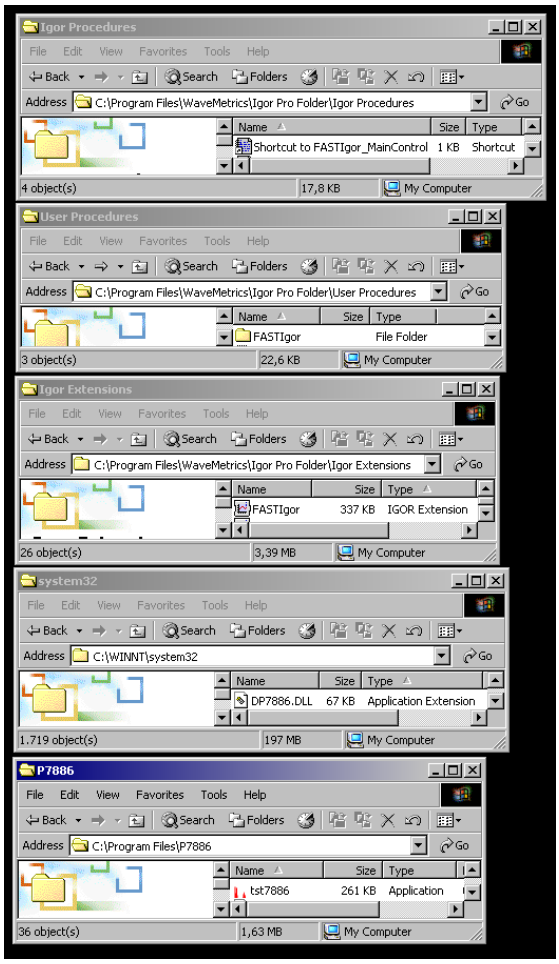

FIG. 5. Required Addresses for files.

15. Open an Igor Experiment. Now see next section: Igor user interface.

III. IGOR USER INTERFACE

A. getting started

Menu `Main Control`, see Fig. 6 gives the user access to all windows needed for running an acquisition and presenting data. As the interface is developed further, the menu will increase with features. The windows are as self-explainatory as possible, with the help line at the lower part of Igor window implemented for many buttons and features.

Go back to Igor. Windows `StartStopRefresh` and `Status` are used for controlling an experiment acquistion. `StartStopRefresh` provides `Start`, which starts, refreshes and provides a Single Coincidence histogram automatically after one second. When you start an experiment, a window will check whether you have saved last experiment. Furthermore, you will be asked for a server parameter. Set it to 0, further info can be found in Section IV.

B. running an acquisition

Windows `StartStopRefresh` and `Status` (windows to the right in Fig. 6) are used for controlling an experiment acquistion. `Start` redefines all waves in experiment, which means that you will lose information gathered in the waves. Always save previous experiment before starting again! `Refresh` button refreshes. `Stop` button stops FAST Card, without refreshing Igor experiment. However, the user has the opportunity to do one last `Refresh` after `Stop`, to include the remaining acquisition time data. In window `Status`, information about the FAST card is updated continuously. The Igor experiment parameters are updated after each `refresh`. The user can preset experiment time, i.e. time between `Start` and `Stop` of an acquisition. `Max Counts` gives max value of `TotalSum` parameter for FAST Card before the experiment is refreshed automatically. Both preset parameters work fine, but not together, for some unknown reason. If you want to use `Time Preset` option, set `Max Counts` to `Inf`. You can `refresh` manually without problems though. If you do not want to use `Time Preset` and/or `Max Counts` options, set `Inf`. You can change `Preset` parameters between two `Refreshes`. The P7886 Server window is also included in Fig. 6, the parameters are identical with the

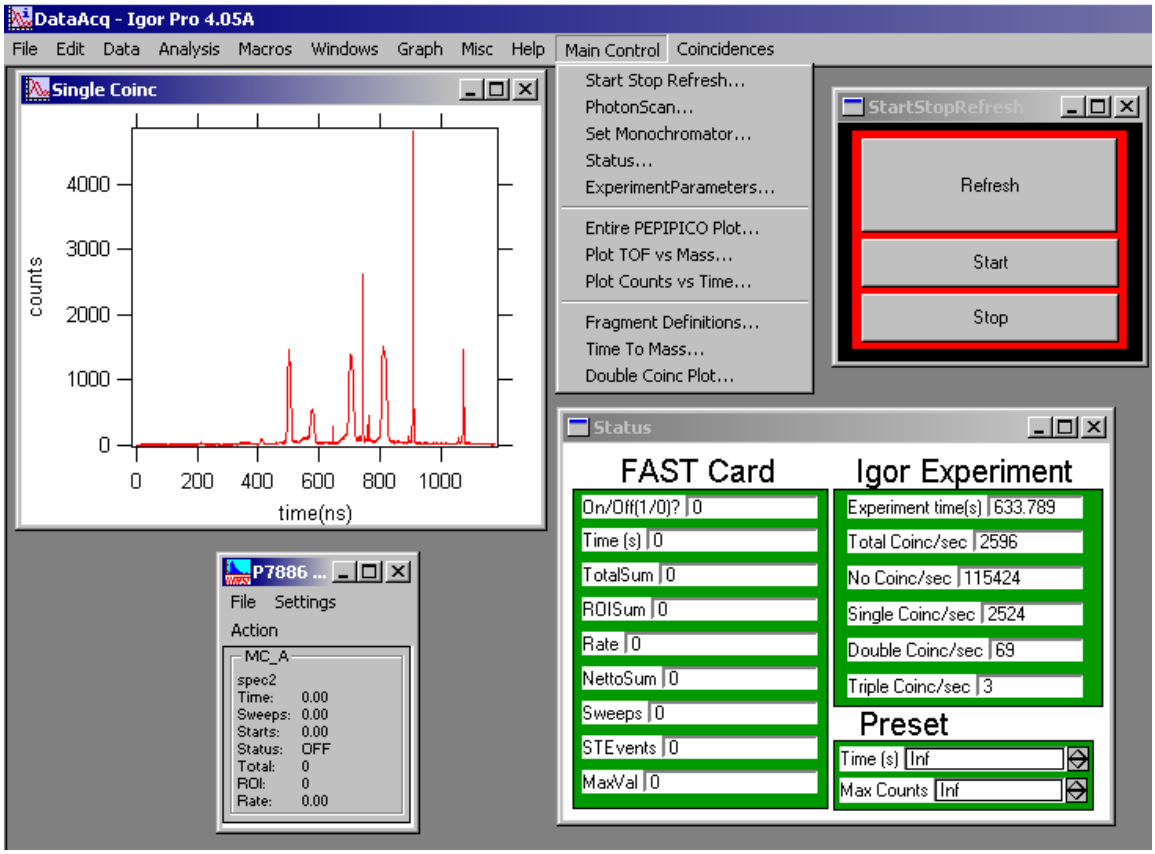

FIG. 6. Igor user interface panels for control of data acquisition.

parameter in `Status` window. A trick: the user can do Menu `Action:Halt`, and later Menu `Action:Continue` in the P7886 Server window, e. g. for changing of gas pressure in the experiment chamber, to "pause" the experiment.

Expression

$$t = \beta + \alpha * \sqrt{m}, \qquad (3.1)$$

where t is time and m is time for a fragment, can be used to transform time to mass/charge information. Two fragments have to be chosen (preferably as far away as possible), and time and mass has to be determined for the two fragments. In Fig. 7, in the lower part of `Single Coinc` window, the Info panel is shown, which can be activated by <Ctrl I> (click on the window to set it active first). The two peaks chosen have TOF 500 and 1074 ns, respectively. And a guess is that the peaks might be related to fragments C^+ and CO^+. Now open window `Cal. M from T` (Menu `Main Control:Time To Mass`). Enter the digits for the related peaks, and press button `Cal and Plot`. Calculated α and β

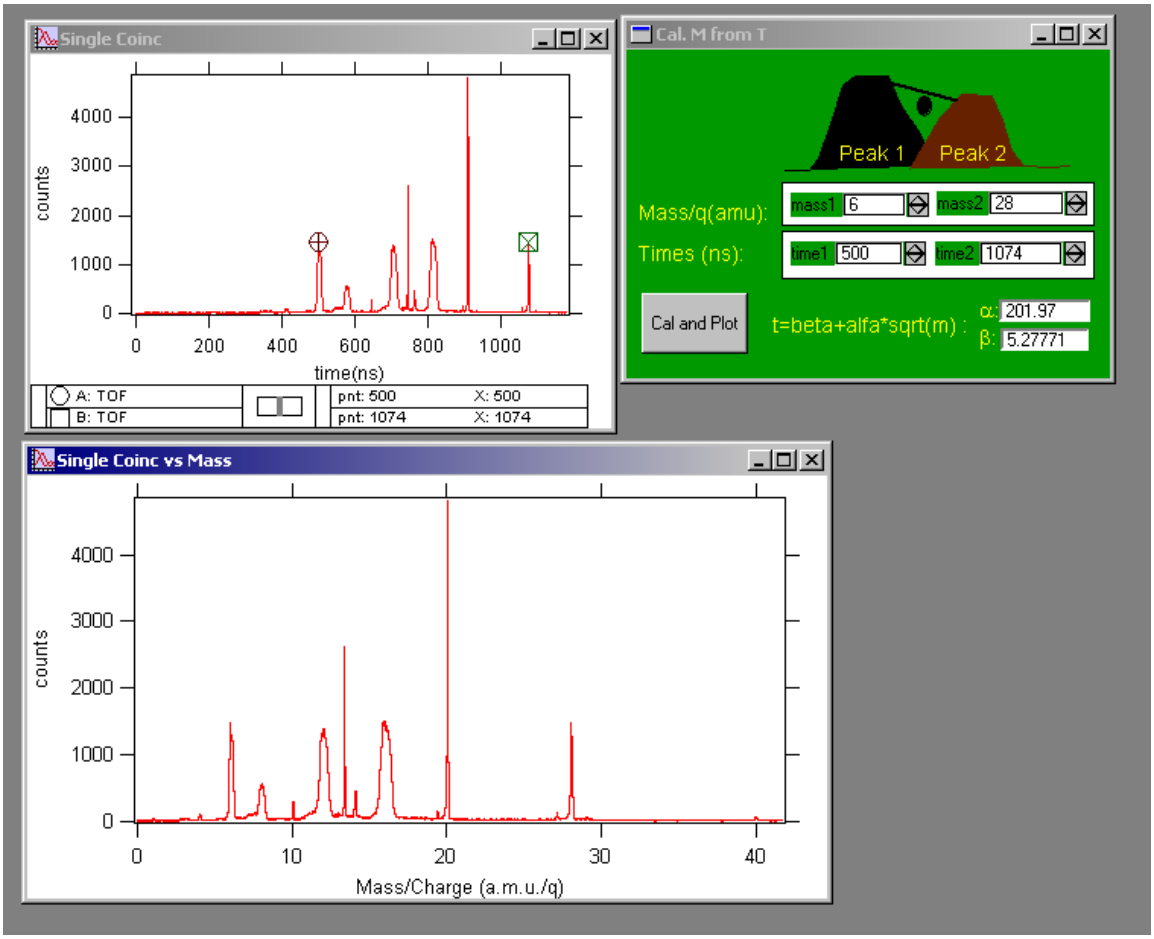

FIG. 7. Time-to-mass conversion.

will be shown in window `Cal. M from T`, and window `Single Coinc vs Mass` will appear. Usually the Time-to-Mass conversion is very convenient, and if the conversion seems strange for some peaks, you probably have to redo the definition of the two peaks in window `Cal. M from T`. In the case shown in Fig. 7, the conversion was successful. Singly and doubly charged CO fragments and CO ions, together with Ar ions, up to quadrupally charged, appear in the window. If the user starts with an experiment, where the Time-to-Mass definitions already have been set, then this work will not have to be redone.

Double coincidence (PEPIPICO) data can be presented as shown in Fig. 8. To get an overview over the total PEPIPICO data, plot window PEPIPICO_Full_Plot: menu `Main Control:Entire PEPIPICO plot`. The user can zoom the window, but the info in the lower left edge will not be updated. Even

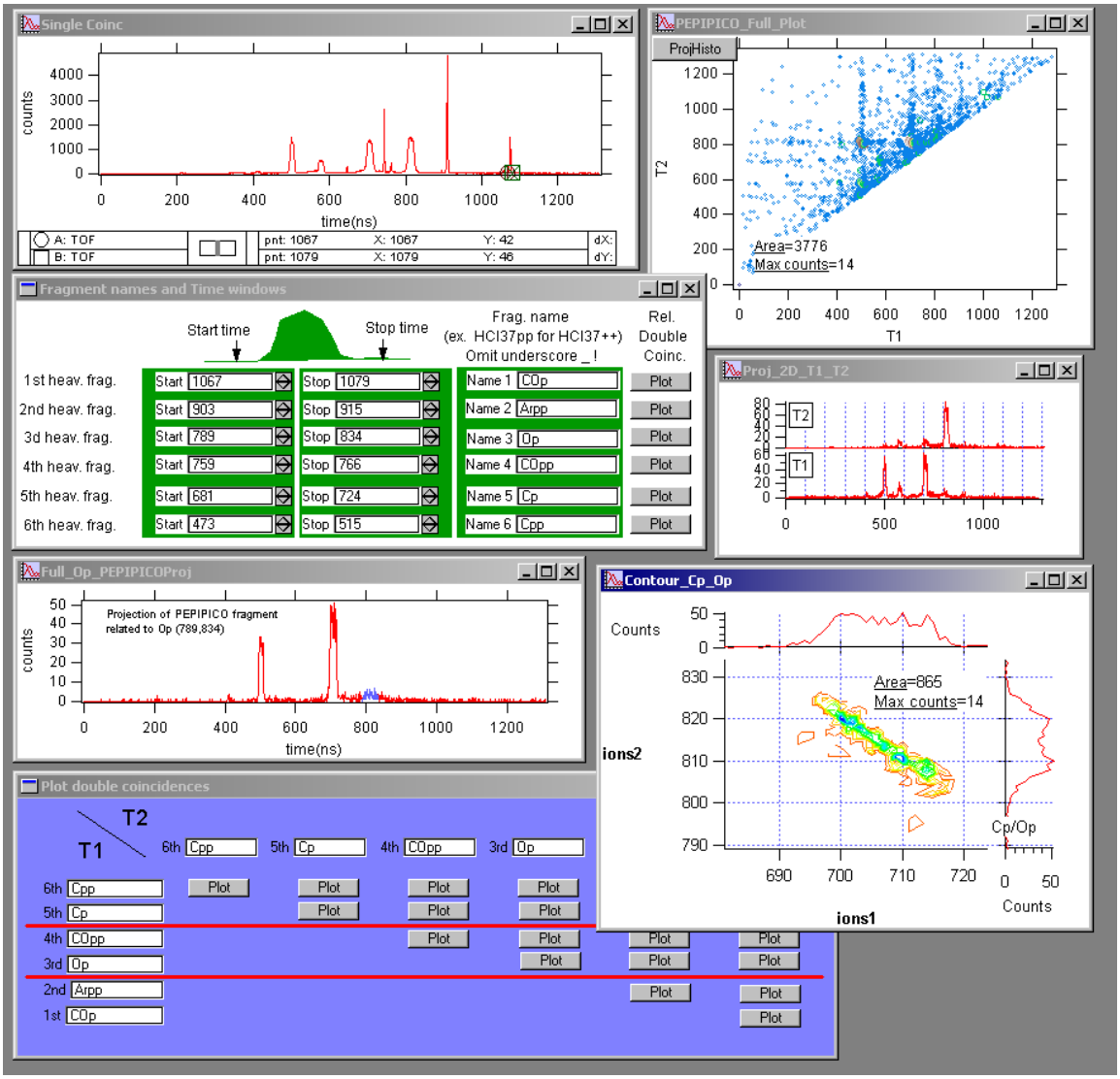

FIG. 8. Double coincidence (PEPIPICO) data presentation.

as the experiment is `Refreshed`, the update of the window will not be performed, instead the user has to plot the window again. However, the button `ProjHisto` uses the zoomed x- and y-axes of the window to plot the projected waves in window `Proj_2D_T1_T2`. Here, T1 is the projection of PEPIPICO data onto the T1 axis, and T2 is the projection onto the T2 axis.

For more detailed info about the fragment coincidences, the peak time windows and names have to be defined. Use window `Fragment Names and Time windows` (Menu `Main Control:Fragment Definitions`). To check the time windows, use window `Single Coinc`, with information

panel <Ctrl I>. For instance, peak related to ion CO^+, with the given name COp, is situated between 1067 and 1079 ns. If the user starts with an experiment, where the Fragment definitions already have been set, then this work will not have to be redone.

The Plot option on the right side of panel `Fragment names and time windows` plots the second fragment in coincidence with the fragment. As an example, the fragments in coincidence with O^+ are shown in window `Full_Op_PEPIPICOProj`. Obviously, the only fragments allowed are the C^{n+} fragments.

Now open window `Plot Double Coincidences` (menu `Main Control:Double Coinc Plot`). In Fig. 8, the C^+/O^+ plot was chosen. These windows are not updated as the user performs `Refresh`, they have to be replotted manually.

C. waves

PEPIPICO measurements create huge amounts of data. The FAST card creates a file `spec2.lst` (see section I B), which looks like in Fig. 9. The zeros are starts, i. e. detected electrons. Due to a non-optimized spectrometer, plenty of electron signals come with no ion fragment triggering a stop. The time window simply stops as time `Range` (Fig. 2) has passed, and the FAST card waits for another electron. In the `spec2.lst` file shown in Fig. 9, two simple coincidences (one stop (detected ion fragment)), one double coincidence (two stops) and one triple coincidence (three stops) are shown.

The `spec2.lst` file is typically far too large for loading into Igor. Instead the `tst7886.exe` file takes this data file, and sorts it into files looking like waves TOF, T1, T2, Counts in Fig. 9. The TOF wave can be plotted vs the Point array (`_calculated_`) in Igor `New Graph` window), since the time

spec2.lst
0
949
0
0
629
727
0
0
0
344
629
737
0
680
0
0
0

Point	TOF
942	427
943	451
944	552
945	566
946	777
947	2994
948	17954
949	59486
950	67677
951	38641
952	19875
953	7799
954	2654
955	896
956	482
957	337
958	309

Point	T1	T2	Counts
13151	629	723	8
13152	629	724	13
13153	629	725	37
13154	629	726	99
13155	629	727	331
13156	629	728	803
13157	629	729	1182
13158	629	730	992
13159	629	731	641
13160	629	732	434
13161	629	733	286
13162	629	734	153
13163	629	735	62
13164	629	736	17
13165	629	737	6
13166	629	738	1
13167	629	739	6

Point	CoinCounts
0	0
1	476221
2	97746
3	14538
4	3449
5	12
6	22
7	11

FIG. 9. The FAST card creates a file `spec2.lst`, and a fraction of the content of such a file is shown. Also shown are the waves created for the user in Igor: TOF (for single coincidences), T1, T2, Counts (for double coincidences) and CoinCounts (Coincidence Counts).

information is inherent here. Unfortunately, for historical reasons, the single coincidences (a single coincidence is defined as one stop only) wave has the name TOF. In other contexts TOF would inherent all (single, double, triple etc) stop times.

The double coincidences are stored in three waves: T1, T2 and Counts. E. g. double coincidence (629, 727) in spec2.lst is equivalent to one count in wave Counts, point 13155 in Fig. 9. Analogous sorting is made for triple coincidences waves T31, T32, T33 and Counts3. This way of sorting PEPICO into a TOF wave, PEPIPICO into waves T1, T2 and Counts and PE3PICO into T31, T32, T33 and Counts3 has its origin from the group of Paul Morin [7] at the LURE Synchrotron in Paris.

The `tst7886.exe` application program also creates a wave CoinCounts, see Fig. 9. Point 0 in CoinCounts gives the number of erratic coincidences in spec2.lst, e. g. sequence 0, 727, 629, 0 with second stop digit smaller than first digit. In this case no erratic coincidences were found (which always seems to be the case). Point 1 gives number of "zero coincidences", electron without ion fragment detected. Point 2, 3, ... 6 gives number of single, double, ... penta coincidences. Point 7 gives number of >penta coincidences. Selected info from CoinCounts is shown in `Status` window, under `Igor Experiment`.

D. miscellaneous functions

The macro provides code for various functions:

`PEPIPICOproj`

`PE3PICOproj`

`PE3PICOprojSD`

`completeTOF`

`P3PICO`

`P3PICO_B`

`Coinc_Abort`

`FilterDouble`

These functions have to be run from command window. Ask A. Naves de Brito or E. S. Cardoso for details.

IV. ADDING TWO EXPERIMENTS

If the user wants to add two experiments, window `Add two experiments`, menu `Main Control: AddTwoExperiments` shows the steps that have to be done.

V. LNLS FEATURES

At the LNLS synchrotron facility in Campinas, Brazil, a serial port connection to a beamline server is provided for controlling and reading of the status of the beamline, which is the source of the incoming photons (see Fig. 1). The XOP VDT (Very Dumb terminal), provided by the Wavemetrics Toolkit, has to be implemented in Igor, just the same way as the XOP FASTIgor (see Fig. 5). When starting up the beamline server computer, press `Cancel` when the `Enter Network Password` window appears. Launch program `WinDCM LINHA xGM` on the desktop. Choose Menu `Arquivo:Modo Local/Remoto:Remoto:R232IG`. The screen will turn into a radioactive colour. Check Menu `Parametros:Detetores`. The user has the possibility to define up to nine experiment parameters. For now, our group has used the reading of parameters Pressure, Photodiode, Ion counting and Electron counting. For non-integer parameters (typically reading of Keithley detectors etc.), the format (Formatacao) of the parameter has to be set to 0.00000E+0. For integer parameters (typically ion and electron counting) the format is 0. Now return to the FASTIgor computer.

A. experiment parameter reading

In window Menu `MainControl:ExperimentParameters`, an overview of the information from the server computer is shown: time, date, photon energy, ring current and the nine detectors DT1-DT9, see Fig. 10. To define the detectors in the Igor experiment, click `Change DTx parameters` to open window `Experiment Parameter Settings`. Give the parameters appropriate names. Be careful to spell pressure parameters *Pressure* (with capital P, * is your choise of string, e. g. HPSPressure1 is OK), then the program does the pressure conversion from Keithley reading to mbar automatically. Ask Paulo de Tarso at the LNLS staff for details. Now close window `Experiment Parameter Settings`. Click refresh experiment parameters. In check window, click `Refresh anyway`. If the setup works, all parameters will be updated. If good statistics of the reading is required, you can set `Integration Time` to a large number.

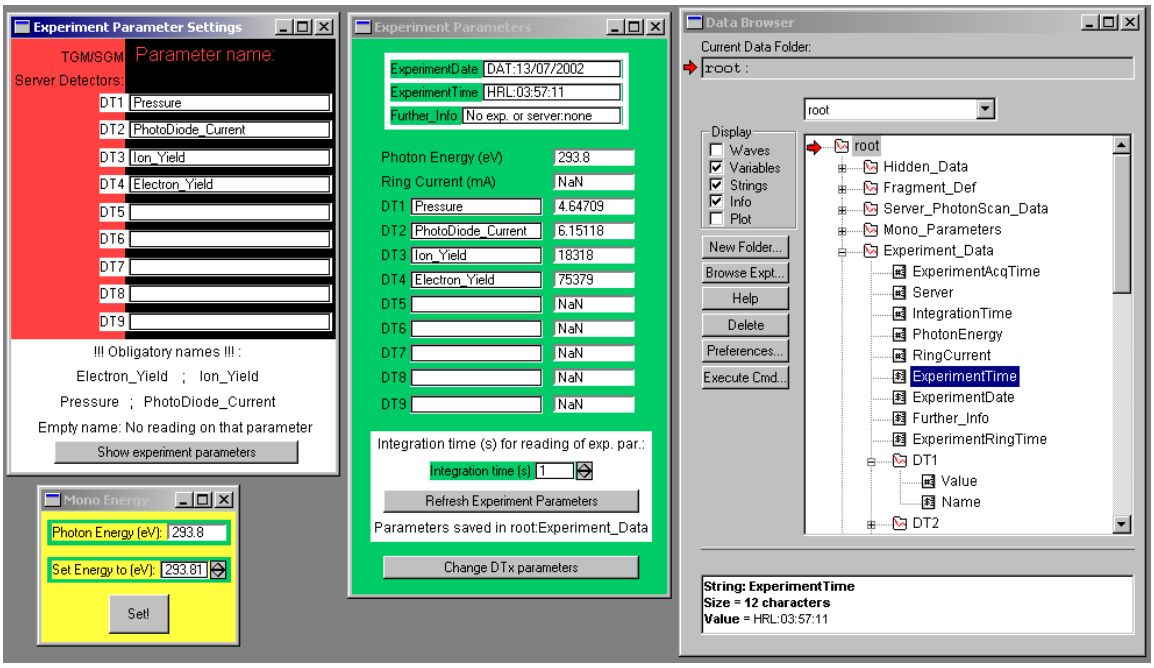

FIG. 10. Features at the LNLS Synchrotron in Campinas, Brazil, for reading of the experiment parameters provided by the beamline server.

B. set monochromator

The monochromator energy can be set by the small window to the lower left in Fig. 10, window `Mono Energy`, which can be opened with: Menu `Main Control:Set Monochromator`. You know that the operation is terminated as the `Photon Energy` is set to approximately the same value as the `Set energy` value.

C. photon scan

In Window `Photon Scan` Menu `Main Control: PhotonScan`, you can set the energy parameters for `Start Energy`, `Stop Energy`, `Delta Energy` and `TimeStep`, for a photon scan experiment. Another name for such an experiment is NEXAFS, Near-Edge X-ray Absorption Fine Structure.

D. server photon scan

Now click `Server Photon Scan` in window `Photon Scan`. In window `Server Photonscan Settings`, you choose which server detector parameters will be plotted, and with which parameters you want to normalize the parameters with.

Now `Start Photon Scan`. The Yield vs

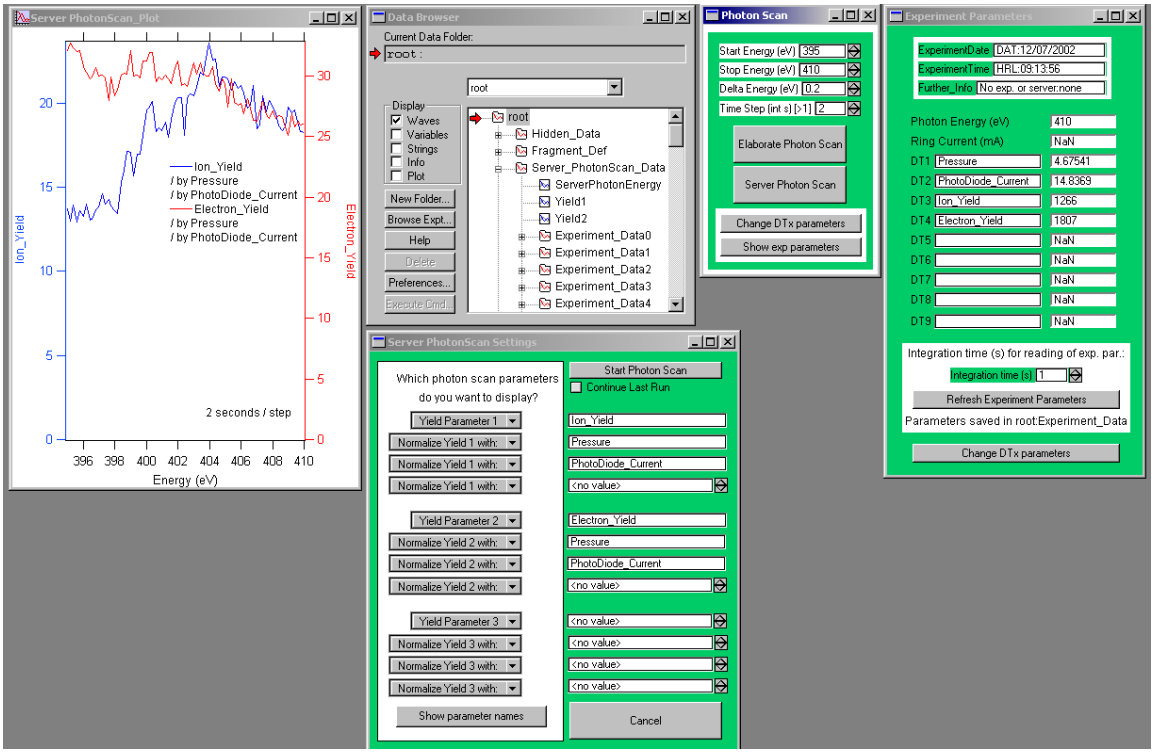

FIG. 11. Server Photon Scan appearence using information obtained by the serial-port connection at LNLS, Campinas, Brazil. N. b.: The Server Photon Scan runs independently of the FASTCard.

Energy plot is as self-explainatory as possible. If you want to pause the experiment, you have to Abort the experiment in the lower left corner. When you want to continue the previous run, you do the same steps as when you started the experiment, but without changing any parameters on the way. Before you click `Start Photon Scan`, click checkbox `Continue Last Run`. The last energy step will now be redone, and the experiment continues as before. The yield waves, the energy wave and the rawdata (i. e. non-normalized information is kept, in fact all parameters shown in window `ExperimentParameters`) are stored in `root:Server_PhotonScan_Data` see menu `Data:Data Browser`, upper second window in Fig. 11.

Both Server Photon Scan and Elaborate Photon Scan (next Section) are dependent on the mood of the beamline server computer. Sometimes it gets rusty, always check that `Experiment Time` in window `Experiment Parameters` is reasonable, the internal Igor

computer clock at the lower right of the screen can be used to check the time difference. So keep the clocks synchronized.

E. elaborate photon scan

Go to Menu `PhotonScan`, and set the parameters as before. Click `Elaborate Photon Scan`, and check the fragments you want to be shown. The fragments are defined in window `Fragment Definitions`, see Section III A. Be careful with the definitions of the time windows, otherwise background will contribute to the yields! The debugging of the PEPIPICO partial yield data treatment has not been finished yet. Even though the user checks PEPIPICO partial yield options, the option will automatically be unchecked by the program as the scan starts. The server yields can be plotted as well, just as in Server Photon Scan, see `Plot DTx Yield` options on the lower side of the panel. Normalizing with server DTx parameters can be done by setting `Norm Yields with:` parameters. All plotted yields are normalized with acquisition time for each photon step by default, since the step time is not always uniform. In fact, the acquisition time tends to overdraw with ≈10 seconds, due to delay of the reading of server parameters. All raw data (waves TOF, T1, T2, Counts, T31, T32, T33, Counts3, CoinCounts as well as all Server experiment parameters) are stored in `root:Elaborate_PhotonScan_Data` with indexed folders for each photon energy, see menu `Data:Data Browser`. The plotted yield waves are stored here as well. Repeat: In indexed folders non-normalized data are stored, only the yields are normalized, if the user has set the option at the start of the experiment.

Sometimes the yields for certain energy steps are not shown. The value of the yields for that energy step is then `NaN` (Not a Number). This is not as bad as it seems: Since the communication between the beamline server computer and the Igor computer is not always working, the DTx parameters can not always be read. And when the yields are normalized by non-valid values. However, the entire raw data including coincidence info is stored in the indexed data folder in `root:Elaborate_PhotonScan_Data`. And after acquisition run, the DTx data can always be interpolated by the other energy step values. And the Yield plot can be completed.

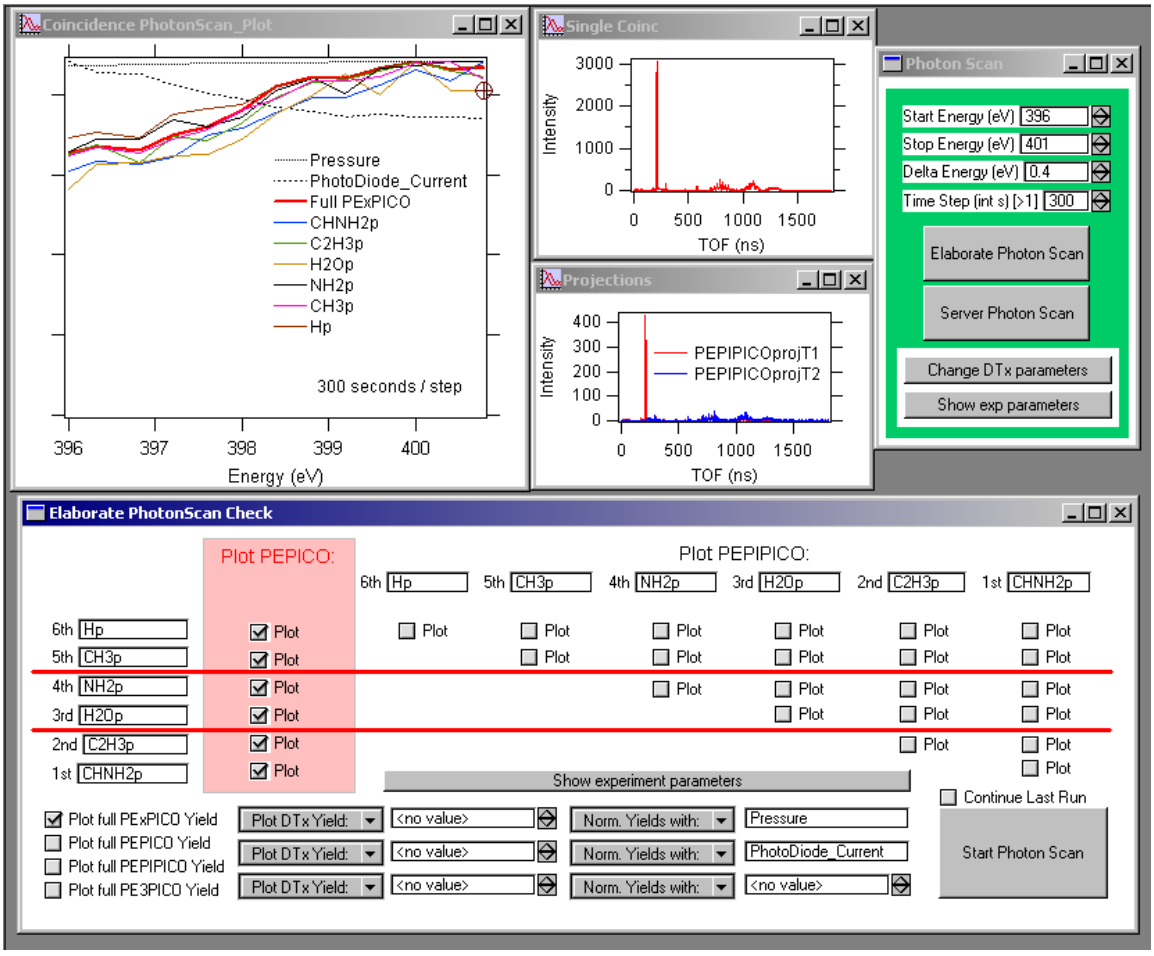

FIG. 12. Elaborate Photon Scan appearence. Full coincidence data (TOF, T1, T2, Counts, T31, T32, T33, Counts3 and Coincounts) is recorded for each photon energy.

If you want to pause the experiment, press Abort in the lower left corner of the Igor experiment. To continue, repeat the steps for starting a scan, but do not change any parameters. Check `Continue Last Run` option above the `Start Photon Scan` button, before pressing `Start Photon Scan`. The fastest way to see if the acquisition run has got stuck (due to communication problems with the server) is to open the P7886 Server window. If the Igor experiment is not updated soon after the FASTCard acquisition has come to the end, it is time to abort and to `Continue Last Run`.

VI. ACQUISITION OVERVIEW

In Fig. 13 the overview of the data acquisition setup is shown. Three levels of code are implemented: Igor macro (up-

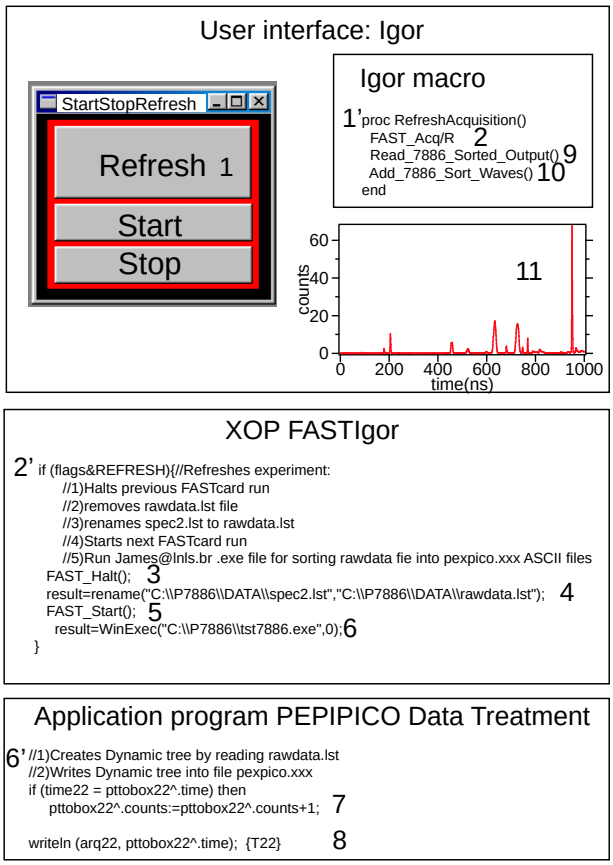

FIG. 13. Overview of the data acquisition set-up.

per section), XOP FASTIgor (middle section), and PEPIPICO Data Treatment application program (lower section). XOP FASTIgor code was developed in Visual C++ [8], and PEPIPICO Data Treatment application program was developed in Delphi (Pascal) [9]. The Igor interface provides a window `StartStopRefresh`, which contains three buttons `Start`, `Stop` and `Refresh`. These buttons are for data acquisition (see III A). As an example for showing the sequence of code implementation, we have chosen the `Refresh` button. In Fig. 13, numbers 1-11 are shown. Where one level calls another, the sequence is shown by a prime (e.g. macro at event 2 calls XOP at 2'). The code is "cleaned", so only essential lines for understanding of the code sequence are shown here.

1. User pushes button `Refresh`, calling macro code (1').

2. Macro

   calls the XOP command `FAST_Acq/R`, provided by XOP FASTIgor. The flag `/R` makes the XOP executing the code under `if(flags&Refresh)` shown in middle section of Fig. 13:2'. The flags `/S` and `/H` would have executed code under `if(flags&Start)` and `if(flags&Halt)`, respectively.

3. XOP halts FAST Card.

4. XOP

   renames `C:/P7886/DATA/spec2.lst` to `C:/P7886/DATA/rawdata.lst`. This trick makes it possible to restart the FAST Card immediately (5), otherwise it would have overwritten the file that

is needed for sorting (7).

5. XOP starts FAST Card.

6. XOP executes `C:/P7886/tst7886.exe`, which is the application program PEPIPICO Data Treatment (6').

7. A tiny sequence of the sorting of data in file `C:/P7886/DATA/rawdata.lst` into several dynamical lists is shown.

8. Dynamical lists are written into files.

9. Macro reads the files written by PEPIPICO Data Treatment into Igor waves.

10. Macro adds new waves to already existing waves in experiment.

11. Updated waves are accessible immediately for the user.

VII. ERROR HANDLING

Obviously, the FASTIgor data acquisition setup is far from crash-proof. A general rule is that Igor experiments should be saved regularly, so that a crash does as little harm as possible. Here is a list of known problems:

1. A Windows error message `I/O error 32` appears. We do not know why. However, the user can always restart the setup again.

2. An Igor window tells the user that some file `C:/P7886/Data/pexpico.xxx` can not be read. This is due to confusion between levels in Fig. 13. Just try one more `Start`, after pushing `Stop` in window `StartStopRefresh`.

This problem list has to be updated, with the help from you users. Tell us all about bugs and problems, and how they occurred, as specified as possible.

VIII. FUTURE DEVELOPMENTS

At this stage, we have a setup that works for data acquisitioning at one photon energy. Here is a list of improvements to be done. The order reflects to some degree the priorities.

1. We have a working photon scan setup for the Brazilian spectrometer. For the swedish spectrometer, code for control of the monochromator will have to be done by network programming in XOP.

2. A 3D plot of TOF waves as a function of monochromator energy.

3. The functions mentioned in III C will get a window, as user friendly as window `Cal. M from T`.

4. For historical reasons, the set-up shown in Fig. 13 shows three levels of code. The lower level application program is unnecessary though. The sorting of data can can be fully integrated on XOP level, so that Igor waves can be updated continuously.

IX. CONCLUDING REMARKS

For future improvements of the code and this manual, please contact me (florian@fysik.uu.se), Emerson (emerson@lnls.br) or Arnaldo (arnaldo@lnls.br). We need to know when you do not understand the user interface, how to set up the data acquisition, or the manual. Is something essential forgotten? What can be added?

---

[1] www.fastcomtec.com.

[2] www.wavemetrics.com.

[3] T. LeBrun, M. Lavollée, M. Simon, and P. Morin, J. Chem. Phys. **98**, 2534 (1993).

[4] A. N. de Brito *et al.*, Chem. Phys. Lett. **328**, 177 (2000).

[5] A. Mocellin *et al.*, accepted for publication in J. Chem. Phys. (unpublished).

[6] J. Eland, Mol. Phys. **61**, 725 (1987).

[7] M. Simon *et al.*, Nucl. Instrum. Methods B **62**, 167 (1991).

[8] msdn.microsoft.com/visualc.

[9] www.borland.com/delphi.