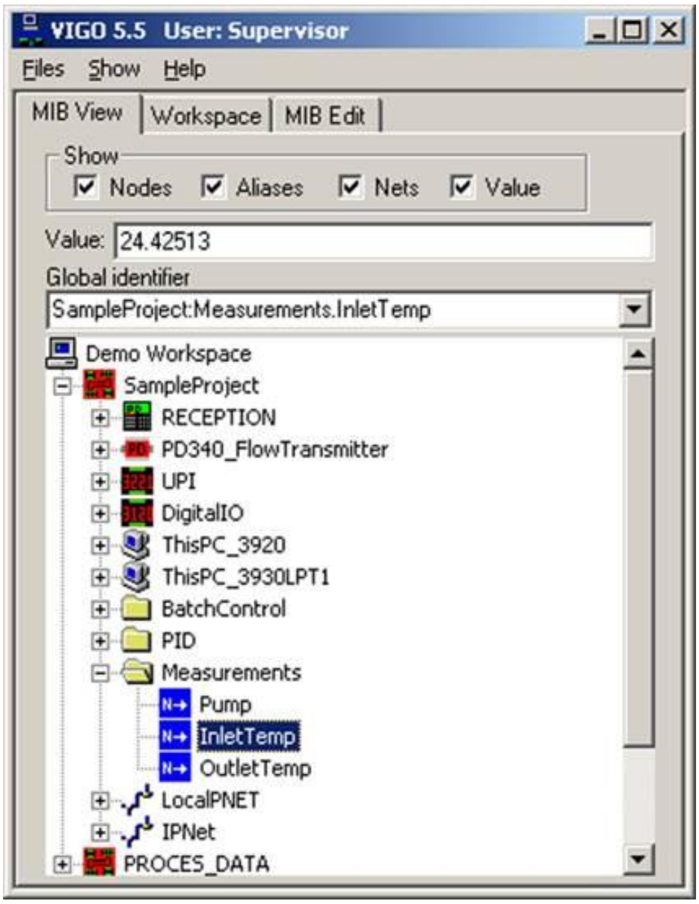
VIGO - The Fieldbus Management System

VIGO is a suite of interrelated PC applications and utilities, which enables designers, and commissioning and maintenance engineers to configure, monitor and debug P-NET enabled systems. At its heart is the mechanism to enable communication between devices connected to P-NET (and other standards), and applications running in a PC. Such PC applications communicate internally by means of what is known as OLE2 Automation (Object Linking and Embedding), which is part of operating systems based on Microsoft Windows.
VIGO is so powerful and versatile, that it is an essential tool when developing P-NET systems. As such, it is provided with a free licence when used with a PC/P-NET interface from PROCES-DATA.



Show
☑ Nodes ☑ Aliases ☑ Nets ☑ Value

Value: 24.42513

Global identifier
SampleProject:Measurements.InletTemp

The main user interface provides the means to build up the structure of a system using a variety of known and registered devices, in a tree structure. For example, if a cluster of 600 series devices is to be used, it is merely a matter of picking each node device type from a list and including it in the tree. Since the structure of each node device is known, VIGO knows how to access every register and what data type it contains. VIGO doesn't know at this stage what the node address of each device is, or the name the user would like to call each device. This is achieved by using a *property* window for each device, where the user just fills in the required information. This procedure is carried out to include all nodes that will form the complete system, including 600 series clusters, other controllers and I/O nodes from other series, and any PCs.

Once the structure of a system has been defined, it is an easy matter to see or change the contents of each channel register. This can be done in 3 ways, either one register at a time by highlighting each register from the expanded tree (MIB view/value), by defining a list of multiple registers (Monitor), or by using one of the utilities which displays all the registers within a standard channel at the same time (Channel configuration).

Controllers and DPIs (masters) are slightly different to I/O type devices (slaves), in that they are programmable. Some of these also contain standard channels, but the user (programmer) can define his/her own data structures and operational software. Once a program has been devised, it can be downloaded to the controller via VIGO. However, VIGO now knows the data structure of the program and where to find specific values of variables held in the controller or DPI. A procedure is provided within VIGO to display such variables within the system tree structure. This means that such programmed variables can be monitored and changed in the same way as the facilities provided to monitor and control variables defined in standard I/O channels.

The VIGO suite of programs also provides facilities to scan the network to display the type and address of every node connected to the network. Also, it provides the means to change node addresses.

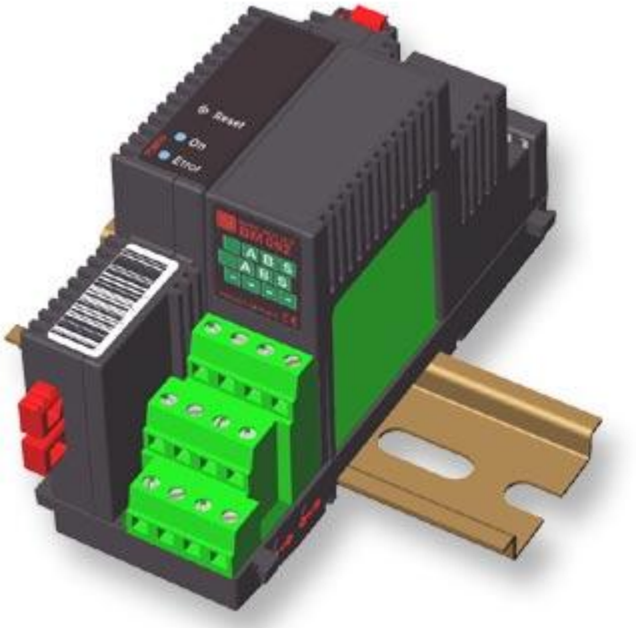
VIGO provides the ability to communicate via other communication standards. For example, two PCs can be connected via Ethernet, and transfer and display P-NET data between them. Also, P-NET data can be transferred between PCs over the Internet, assuming each PC has been set up to make such communication. Direct or dial-up connections can also be made, to transfer P-NET data using the PC's RS232 ports via modems.
For more detailed information about the use of VIGO, refer to the VIGO manual, or the VIGO on-line help.

Data Acquisition Software for DPIs

In order to set up a Data Acquisition system that doesn't rely on Visual VIGO running, or a PC having to be constantly powered, there needs to be a means to continually collect and log data from a remote source. One such device that can perform this task is a PD 60xL DPI (Distributed Process Interface) from the PD 600 series of P-NET modules. These are user programmable devices that can be accessed via standard P-NET, Light-Link, RS 232 or Ethernet.

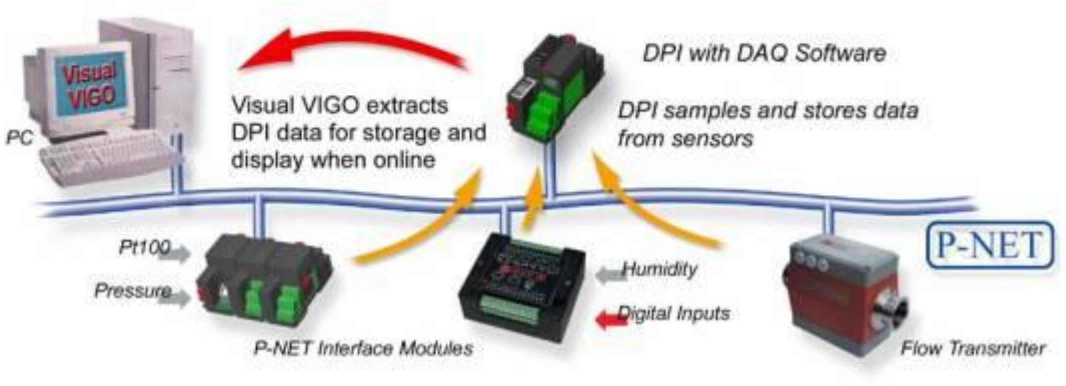
DataCollect Channels

Such a device needs to be programmed to create a number of individually configurable DataCollect Channels, each capable of being linked to a particular process variable located anywhere within the P-NET system. The **Remote Data Acquisition** component in Visual VIGO can be used to configure an individual DataCollect channel to log data from a specific variable, but the Data Collect Configuration Program can be used to configure and monitor many channels at the same time. When Visual VIGO is running, any ***new*** data collected by the DPI is automatically uploaded to the PC hard disc, which then enables the display of charts, curves and data tables to be displayed for that process variable. This data is then also available to other applications such as Excel, Access, SQL, by means of OLE2 Automation.



Free Software

PROCES-DATA offers the DPI software to perform the complete Data Acquisition process FREE OF CHARGE. This can be downloaded from the PROCES-DATA web site. It makes available up to 100 DataCollect channels, which, depending on the number of channels used and the configured frequency of log samples, enables many days, weeks or even months of unattended data to be stored. Once Visual VIGO is launched, any new data that has not already been stored on the PC hard disc will be automatically extracted from the DPI and saved on the PC hard disc. As long as Visual VIGO is run at least once before the newest unsaved data held on the DPI starts to overwrite the oldest unsaved data, then all stored logged data on the PC will always be complete.



Flexible Logging Method

Data Collect channels provide a wide collection of logging options ranging from the simple regular sample rate, through conditional logging based on change of value and/or the state of another variable, to memory saving processed logging to provide 'best fit' curve data.

Data Collect Configuration Program

Introduction

The Data Collect Configuration Program (DCConfig), provides a means to quickly set up and monitor a number of individual DataCollect Channels in a Data Collection enabled DPI (Datalogger, P-NET Controller). The data logged within DataCollect Channels is used by Visual VIGO to graphically display this in the form of curves and tables. Individual channels can also be configured and monitored using the property window of a Remote Data Acquisition component in Visual VIGO, but DCConfig provides the facilities to apply such activities to many channels at the same time. It also allows such configurations to be edited and stored on file, for immediate application to the same or additional controllers. The program is available as a licence free download.

Configuration and Monitoring Techniques

Configuration Preparation

The program enables a systems designer to firstly prepare (edit) and then download (store) a set of configurations for up to 100 DataCollect channels. The designer can define the location of each log variable by PhysID, set logging threshold values, define sample times, and select and configure a range of conditional filters. Facilities are also provided to name particular filter conditions so that they can be quickly identified and selected to configure other channels. Each channel can also be identified with a comment. When editing is complete, this configuration set would be saved as a user named file.

Downloading Configurations

The content of this file is now used to download (store) configuration data to the datalogger. The download mechanism first checks that there are no repeated log variable PhysIDs within the configuration file and that the settings defined for any thresholds etc. are legitimate. Next, channel numbers are allocated for each of the configurations, which are then downloaded.

Uploading Data

All of the channels configured as 'active' will now commence their data logging duties. This can be tested and monitored by using the 'Load data from datalogger' facility. This will provide values within the Channel Overview table in terms of channel number, current log value, sample timer, and decoded status information. This data can be updated by loading data again.

Uploading Configurations

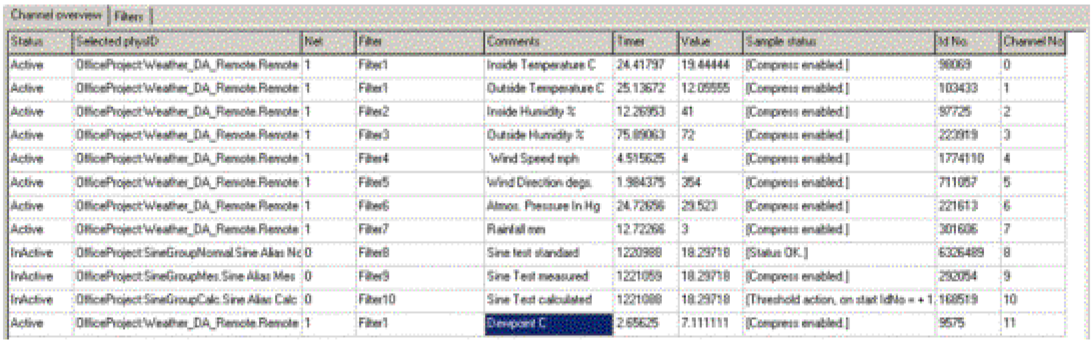
For Dataloggers that have already been configured, where the designer needs to be reminded of the current structure, or perhaps because additional channels need to be configured, the facility is provided to upload all configuration data from a selected datalogger. It is advisable to open a new configuration file before performing this procedure, since no user comments or user defined filter names are retained in a datalogger and will therefore be cleared from the table. To configure additional channels, or to reconfigure current settings, is a matter of adding additional rows in the table and defining the attributes. By selecting to download this configuration set to the datalogger, will allocate unused channels for the new configurations and, following a warning if you will be overwriting a currently configured channel, will download the data and start the logging activity. It should be noted that in modifying a currently active channel, any previously logged data could be lost. Data in current channels that are not being modified will not be affected.

Program User Interface

The program is presented in the form of two tab-identified tables. The first is referred to as "Channel Overview" and the other as "Filters". Channel Overview enables the user to see the configurations and latest data from all or a selected list of DataCollect channels. It also provides the ability to sort information by columns and to change the type of configuration (filter) and to activate/de-activate individual channels. The Filter table enables the user to define a set of named configurations (filters), which can then be applied to individual or groups of DataCollect channels. The program menu provides a number of general facilities such as saving files, inserting/deleting rows and up/downloading data to and from DataCollect channels.

Channel Overview

This table consists of 10 columns, each having a particular attribute. Clicking on any of the column headings will perform a sort on the whole table with respect to that column, in either ascending or descending order. The width of columns can be adjusted by dragging the header boundary in either direction with the mouse.



Status	Selected physID	Net	Filter	Comments	Timer	Value	Sample status	Id No.	Channel No
Active	OfficeProject\Weather_DA_Remote.Remote	1	Filter1	Inside Temperature C	24.41797	19.44444	[Compress enabled.]	98069	0
Active	OfficeProject\Weather_DA_Remote.Remote	1	Filter1	Outside Temperature C	25.13672	12.05555	[Compress enabled.]	103433	1
Active	OfficeProject\Weather_DA_Remote.Remote	1	Filter2	Inside Humidity %	12.26953	41	[Compress enabled.]	97725	2
Active	OfficeProject\Weather_DA_Remote.Remote	1	Filter3	Outside Humidity %	75.89063	72	[Compress enabled.]	223919	3
Active	OfficeProject\Weather_DA_Remote.Remote	1	Filter4	Wind Speed mph	4.515625	4	[Compress enabled.]	1774110	4
Active	OfficeProject\Weather_DA_Remote.Remote	1	Filter5	Wind Direction degs.	1.984375	354	[Compress enabled.]	711057	5
Active	OfficeProject\Weather_DA_Remote.Remote	1	Filter6	Atmos. Pressure In Hg	24.72656	29.523	[Compress enabled.]	221613	6
Active	OfficeProject\Weather_DA_Remote.Remote	1	Filter7	Rainfall mm	12.72266	3	[Compress enabled.]	301606	7
InActive	OfficeProject\SineGroupNormal.Sine Alias Nc	0	Filter8	Sine test standard	1220988	18.29718	[Status OK.]	6326489	8
InActive	OfficeProject\SineGroupMes.Sine Alias Mes	0	Filter9	Sine Test measured	1221059	18.29718	[Compress enabled.]	292054	9
InActive	OfficeProject\SineGroupCalc.Sine Alias Calc	0	Filter10	Sine Test calculated	1221088	18.29718	[Threshold action, on start IdNo = + 1.	168519	10
Active	OfficeProject\Weather_DA_Remote.Remote	1	Filter1	Dewpoint C	2.65625	7.111111	[Compress enabled.]	9575	11

Status

This column indicates whether a particular channel is active or inactive. By clicking on an individual cell will toggle its present state.

Selected PhysID

These cells show the actual Physical Identity of the variable to be logged in a DataCollect channel. Clicking on a particular PhysID or empty cell will highlight the cell and provide the opportunity to select a new one from the MIB.

Net

This editable column defines the network route through which the data-logging DPI can obtain the variable value. The syntax of this path definition is "[out of port No.][in through node address][out of port No.]…". The node address of the final module is added automatically when downloading the configuration, so there is no need to include this. Net = 0 means that the value is obtained internally from within the data-logging DPI. For a PD 600L DPI, Port 1 is RS 485 and Port 2 is Light-Link. So, for example, if the variable to be logged is within a module in the local cluster, Net = 2. If the variable resides in a module directly connected to the same RS 485 network as the DPI,

Net = 1.
If Net = 2 5 1, it means the path is out of Port 2 (Light-Link), in through node address 5 (e.g. another DPI) and out of Port 1 (e.g. 485).
According to the syntax there must be a space between each of the parameters defining the network route.

Filter

This column provides an indication as to the type of configuration that has been applied to this channel. The user has the opportunity (in the filter table) to give a meaningful name to each of the different configuration types. If a name has not been given, the program applies a default name such as “Unknown Filter 1…n”. Clicking on one of the cells in this column will provide a drop down list of all the configured filter types available, and the user may choose an alternative from these if required.

Comments

This text column is freely editable by the user to provide some means of describing what function each channel provides e.g. “Tank 1 Temperature”. This data is only stored in the PC file, NOT in the datalogger.

Timer

These cells indicate the state of the logging timers at a particular point in time when data is uploaded. On performing a sort on this column, it can be easily seen whether any timers have gone negative or have not changed, indicating that there might be a problem with those particular channels.

Value

This column contains the current value at the instant of data download of the variable that is being monitored for logging.

Sample Status

Provides an indication of how each channel is configured in terms of whether and what type of conditional processing is being applied.

Id No.

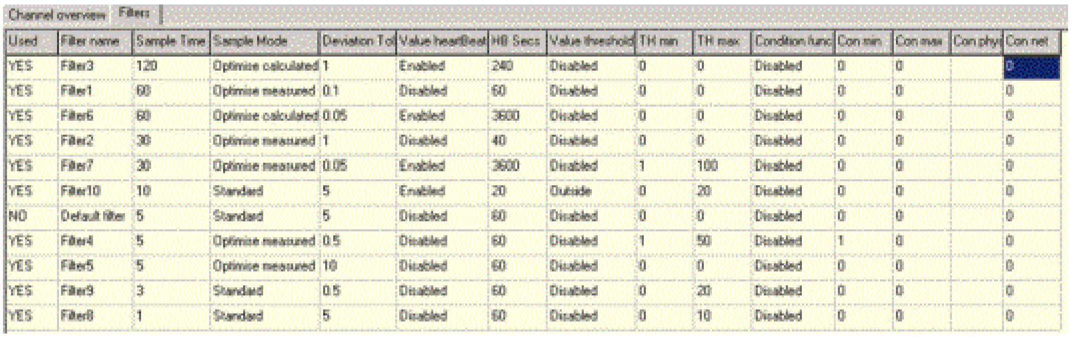
Indicates the current identification number of the latest logged data at the time of data download.

Channel No.

Identifies the channel number associated with a particular table row.

Filters

The 15-column Filters table provides an easy way to define and apply a set of logging conditions to individual or to a set of DataCollect channels. By providing a meaningful name to each of the unique condition types, enables that log condition to be selected for particular log definition in the Channel Overview table. As with Channel Overview, each column width can be adjusted by dragging, and the first three columns can be sorted in ascending or descending order.



Used	Filter name	Sample Time	Sample Mode	Deviation Tol	Value heartBeat	HB Secs	Value threshold	TH min	TH max	Condition func	Con min	Con max	Con phy	Con net
YES	Filter3	120	Optimise calculated	1	Enabled	240	Disabled	0	0	Disabled	0	0		0
YES	Filter1	60	Optimise measured	0.1	Disabled	60	Disabled	0	0	Disabled	0	0		0
YES	Filter6	60	Optimise calculated	0.05	Enabled	3600	Disabled	0	0	Disabled	0	0		0
YES	Filter2	30	Optimise measured	1	Disabled	40	Disabled	0	0	Disabled	0	0		0
YES	Filter7	30	Optimise measured	0.05	Enabled	3600	Disabled	1	100	Disabled	0	0		0
YES	Filter10	10	Standard	5	Enabled	20	Outside	0	20	Disabled	0	0		0
NO	Default filter	5	Standard	5	Disabled	60	Disabled	0	0	Disabled	0	0		0
YES	Filter4	5	Optimise measured	0.5	Disabled	60	Disabled	1	50	Disabled	1	0		0
YES	Filter5	5	Optimise measured	10	Disabled	60	Disabled	0	0	Disabled	0	0		0
YES	Filter9	3	Standard	0.5	Disabled	60	Disabled	0	20	Disabled	0	0		0
YES	Filter8	1	Standard	5	Disabled	60	Disabled	0	10	Disabled	0	0		0

Used

Provides an indication as to whether this condition has been selected to be used by any log channel.

Filter Name

This editable column provides the means to uniquely name a condition filter. Thereafter, this name can be selected when configuring new channels.

Sample Time

This editable column shows and provides the means to define the period in seconds between log samples applying this filter.

Sample Mode

Indicates the kind of conditioning that is to be applied by this filter type. By clicking on a particular cell in this column and selecting the '+' button, provides a pull down list to select the type of conditioning to be used.

- **Optimise Measured –** a means of data compression where only actual measured values that full outside a defined deviation tolerance are logged. See illustrated explanation.
- **Optimise Calculated –** another means of data compression where a logged value is calculated when the measured value falls outside a defined deviation tolerance. See illustrated explanation.

- **Standard –** all measured values are logged at sample time. See illustrated explanation.

- **Disable –** Sampling is de-activated.

Deviation Tolerance

Provides the means to define by how much a measured value must change either positively or negatively (deviate) before a new value is considered for logging. See fuller explanation.

Heartbeat Function

Provides a pull down list to select whether this configuration includes the means to force a value to be logged if no conditional log values have been stored. See Heartbeat.

HB Secs

These editable cells enable a frequency to be defined at which a measured value is logged when no other logs have been taken during this period. See Heartbeat.

Value Threshold Function

The cells within this column include a pull down list providing a selection of conditional tests to establish whether a measured value should be considered for logging. See fuller explanation.

- **Disabled –** the Threshold Function is disabled and any measured value will be considered for logging.
- **Outside –** Only measurements falling outside the defined **Min**imum and **Max**imum threshold values will be considered for logging.
- **Inside** - Only measurements falling inside the defined **Min**imum and **Max**imum threshold values will be considered for logging.
- **Over** - Only measurements exceeding the defined **Max**imum threshold value will be considered for logging.
- **Under** - Only measurements falling below the defined **Min**imum threshold value will be considered for logging.

TH Min

These editable cells define the minimum or low value that will be used in the Threshold truth test.

TH Max

These editable cells define the maximum or high value that will be used in the Threshold truth test.

Condition Function

An additional applied test to include the state of another defined secondary variable to establish whether the log variable is to be considered for logging. See fuller explanation.

- **Disabled –** the Sample Enable Function is disabled and will not influence the taking of log data.
- **Outside –** Only when a secondary variable value falls outside the defined **Min**imum and **Max**imum values will log sampling be enabled.
- Inside - Only when a secondary variable value falls inside the defined **Min**imum and **Max**imum values will log sampling be enabled.
- **Over** - Only secondary variable values exceed the defined **Max**imum value will log sampling be enabled.
- **Under** - Only secondary variable values falling below the defined **Min**imum value will log sampling be enabled.

Con Min

Used to define the minimum value used in the secondary variable test.

Con Max

Used to define the maximum value used in the secondary variable test.

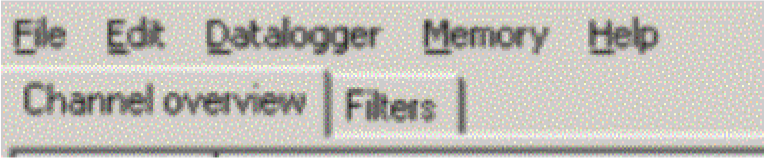
Con PhysID

Defines the PhysID of the secondary variable. Clicking any cell provides the means to choose the required variable from the MIB.

Con Net

This editable column defines the network route through which the data-logging DPI can obtain the variable value for testing. For an explanation of the way this number is defined and used, see Net above.

Menu Items



File

- **New –** Will open a new blank table. Provides a warning to save the current table if it has been changed.
- **Open –** Provides a file selection window to open a previously saved configuration table.
- **Save –** Save the changes made to the open configuration table using the current file name.
- **Save As –** Saves the changes made to the open configuration table using a new name.
- **Config-file Info –** Provides an editable dialogue box showing the file name and last edit date together with an area for users notes.

Edit

- **Delete channel-row (delete) –** Deletes the row in which a cell is selected in the Channel Overview table.
- **Add channel-row (insert) –** Adds a blank row to the bottom of the Channel Overview table.
- **Delete filter-row (delete)** - Deletes the row in which a cell is selected in the Filter table.
- **Add filter-row (insert)** - Adds a blank row to the bottom of the Filter table.

Datalogger

- **Select Datalogger –** Provides a selection dialogue box with the opportunity to select a data logging enabled controller from the MIB.
- **Store configurations to Datalogger –** Will download the configuration data from the currently opened file to the selected data-logging controller. A number of checks are made before this is performed:-
    - If a configuration file contains a repeated PhysID within the Channel Overview table, it will be prevented from being downloaded, and a warning given that “Two channels cannot have the same PhysID”. N.B. If it is necessary log a particular variable in different ways, create an alias to ensure that a PhysID is not repeated.
    - If one or more net definitions have not been entered, the configuration will be prevented from being downloaded, and a warning given that “A net must be chosen for all channels”.
    - If one or more status definitions (active/inactive), have not been entered, the configuration will be prevented from being downloaded, and a warning given that “All channels must be set to active or inactive”.
    - If one or more filter type definitions have not been entered, the configuration will be prevented from being downloaded, and a warning given that “A valid filter must be selected for all channels”.
    - If all the above tests are met correctly, a dialogue box asks “Are you sure you want to overwrite existing datalogger configurations?” By selecting “YES”, further checks are made to compare the configuration file with the current datalogger configuration. By selecting “NO”, no further download activity will take place.
    - If the configuration file to be downloaded does not define all the channels already configured in the datalogger, a message is given that “The datalogger contains one or more PhysIDs not included in the file you are storing. Do you wish to reset these channels?”. By answering “YES”, will delete those channels not included in the configuration file. Any newly specified PhysID will have a new channel created. All other specified channels will be unaffected. By answering “NO” will leave the unspecified channels unaffected during download.
    - If one or more currently configured datalogger channels’ PhyIDs are changed in the configuration file, the same message as above is displayed. By answering “YES”, will reconfigure those channels for the new PhyID and begin logging. The data collected by that channel using the previous PhysID will be lost.
- **Load configurations from Datalogger –** Updates the table with the current configurations from the selected data-logging controller. Provides a choice of whether to load “All Channels”, “All Configurations”, or “Enabled Configurations”. N.B. Since a datalogger does not store any user comments or user filter names, uploaded configuration data will nullify any comments and show “Unknown” filter names in the currently opened file.
- **Load data from Datalogger –** Provides a one-shot upload of the sample timers and the latest values at sample time (sample timer = 0). This can only be activated following the storing of configuration data from a previously saved file, or by loading configuration data into a new file.
- **Master Reset –** Provides the ability to perform a master reset on the data-logging controller. This would normally only be performed when using a newly installed controller to which a prepared library of configurations are to be downloaded.

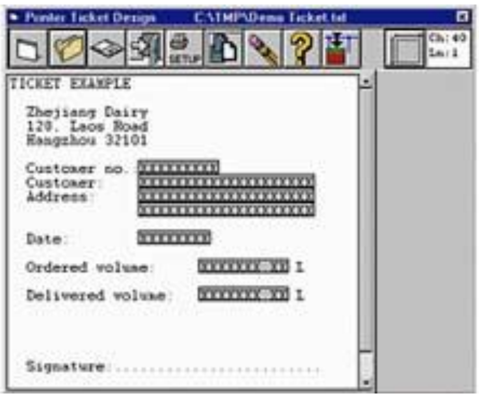
Memory

- Memory Status – Provides a dialogue box containing information about how much memory is available on the PC hard disc together with how long it is possible not to have a Visual VIGO design running and connected before memory in the Data-logging controller starts to become overwritten.

Help

- **About –** Shows information about the program and version
- **On-Line Help –** Provides a link to the PROCES-DATA.com web site where this document can be found.

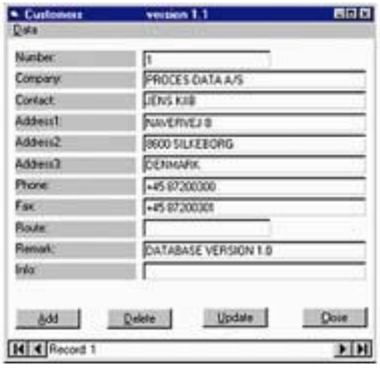
Printer Ticket Design Program



```
TICKET EXAMPLE

  Zhejiang Dairy
  120. Laos Road
  Hangzhou 32101

  Customer no.
  Customer:
  Address:

  Date:

  Ordered volume:                 l

  Delivered volume:               l



  Signature:........................
```

Program for design of printer tickets for the PD 4000/340 Flowmeter-Display. The program makes it possible to create a ticket layout on the PC monitor. This design is then downloaded to the PD 4000/340. A customized ticket layout may be saved on the PC hard disc for later use. Requires PD 3920/PD 3930 and VIGO software.
The Printer Ticket Design Program is sold on a try-before-buy basis.
Click the "files" button below to go to the download page.

PDFLOW Database Program



Number:	1
Company:	PROCES DATA A/S
Contact:	JENS KIIB
Address1:	NAVERVEJ 8
Address2:	8600 SILKEBORG
Address3:	DENMARK
Phone:	+45 87200300
Fax:	+45 87200301
Route:	
Remark:	DATABASE VERSION 1.0
Info:	

PDFLOW is a tool for generating a customer database for PD 4000/340 Flowmeter-Display. The database can hold up to 1200 customers. This information is then downloaded to the PD 4000/340. The PDFlow Database program is based on a standard database using Access 2.0 file format. Requires PD 3920/PD 3930 and VIGO software.

Download the fully featured 30-days-free-trial version

The PDFLOW Program is sold on a try-before-buy basis.
Click the "files" button below to go to the download page.

The Process-Pascal programming language

Process-Pascal, which is used to program Controllers from PROCES-DATA, is a high level programming language based on Standard Pascal.

Process-Pascal is an extension of Standard Pascal, with a number of facilities that make it possible to execute several programmes simultaneously in one computer. This is called multi-tasking.

Process-Pascal has been specially developed for use in connection with P-NET, which is a local area network (fieldbus) for transmission of data throughout distributed data acquisition and process control plants. Data, which are distributed in modules on the P-NET fieldbus, can be defined as variables in Process-Pascal.

Process-Pascal permits automatic reports of alarms in the event of error in the controller or in the interface modules. Furthermore, it is possible to automatically test all components within the plant during the starting phase.

Process-Pascal includes standard routines for interactive screen dialogue. Thus it is possible to define that a particular variable must be shown on the screen and be continuously updated. Data can be keyed into a variable by pointing at it using the screen cursor.

Process-Pascal programmes can be written using any general-purpose editor, which uses ASCII or ANSI files, or the editor included in the Process-Pascal complier suite can be used.

Process-Pascal programmes operate with several types of variables. The compiler automatically performs typecasting during compilation. This makes programmes safer and easier to develop. The Process-Pascal program suite includes a debugger, which is a very powerful tool that significantly speeds up the overall process of application development.

Process-Pascal programmes are compiled using a cross-compiler running on a PC under the Microsoft Windows operating systems. The compiler generates code, P-code that is stored in the controller in FLASH or RAM memory. The operating system in the controller interprets the P-codes and executes a piece of machine code for each P-code.

The Process-Pascal program suite is an integrated part of the VIGO installation package (VIGO version 5.4 and newer), and is therefore available for the user when VIGO is installed. No separate licence fees are required for using the Process-Pascal tools.