

# **VOLTECH INSTRUMENTS**

---

## **AT Server (Legacy Version)**

**For software V3.xxx**

## **User Manual**

---

# **AT Server (Legacy Version)**

## **For software V3.xxx**

### **User Manual**

**98-092 Issue 1 / 15 JUL 2020**

[sales@voltech.com](mailto:sales@voltech.com)  
[www.voltech.com](http://www.voltech.com)

© Voltech Instruments All rights reserved

No part of this publication may be produced, stored in a retrieval system, or transmitted in any form, or by means, electronic, mechanical photocopying, recording or otherwise without prior written permission of Voltech

# Table of Contents

<b>1. AT Series Server Software.....</b>	<b>5</b>
1.1. Introduction.....	6
1.2. Installing the Server Software.....	7
1.2.1. Computer Requirements.....	7
1.2.2. Installation Instructions.....	7
1.3. Starting for the first time.....	8
1.3.1. Running the software.....	8
1.3.2. Initial Configuration.....	8
1.4. Setting up the Options.....	9
1.4.1. Communications.....	9
1.4.2. Directories.....	13
1.4.3. Groups.....	14
1.4.4. Password.....	14
1.4.5. Results.....	15
1.4.6. Language.....	15
1.4.7. Storing Test Programs.....	16
1.4.8. Viewing a Test Program.....	17
1.4.9. Deleting a Test Program.....	18
1.5. Test Results Handling.....	19
1.5.1. Receiving Test Results.....	19
1.5.2. Result Display and Analysis.....	20
1.5.3. On-Line Monitoring.....	21
1.5.4. Batch Statistics On-line Analysis.....	22
1.5.5. On-Line Analysis via a Database.....	24
Introduction.....	24
Setting Up Database Results Storage.....	24
Database Tables.....	29
Charting and Analysis Using SPC/SQC Software.....	33
1.5.6. On-Line Analysis via OLE.....	35
1.5.7. Off-line Analysis.....	45



## **1. AT Series Server Software**

Centralized program and results data for the AT Series Transformer Testers.

**This manual is for the Legacy version of the AT Server.**

**Historically it was part of the AT3600+ AT5600 + ATi user manual but has been extracted as a separate document.**

**This manual supports V3.xx of the AT Server.**

**The new dotNET AT Server (V4.xx) has been rewritten and has new features.**

**Please see the separate dotNET AT Server Manual (98-122) for details.**

## 1.1. Introduction

This chapter will guide you through the operation of the AT Series Server,



Press the F1 key to get help at any time while using the software.

The server program supplied with your AT series tester may be considered as an on-line storage system.

The server will handle two types of data:

1. Test Programs
2. Test Results

Test programs that are created using the Editor application may be uploaded into the server file store. All testers connected to the server then have access to the file store. Programs are requested by an AT tester, recalled from the server, and then returned to the AT tester for execution.

Test results are generated from every program execution, if the option is set, and sent to the server for storage. The data is then stored in a file or a database for record keeping and analysis.

The features of the server include:

1. MS Windows application style dialogues and tools
2. Storage of all previously created test programs
3. Transfer of test programs to the AT tester on request
4. Viewing of test programs outside the test program editor environment
5. Test results may be received and stored in a database ready format for easy offline analysis
6. OLE 2 support allowing real-time analysis of your production test system

## **1.2. Installing the Server Software**

All software is available from the Voltech website [www.voltech.com](http://www.voltech.com).

### **1.2.1. Computer Requirements**

Before attempting to install any Windows software, check that the PC satisfies the following hardware requirements:

<b>PC Processor:</b>	486 or better
<b>Memory:</b>	4MB minimum
<b>Hard Disk Space:</b>	5MB minimum

(Note: 3MB does not include disk space required for storing programs or test results)

<b>Operating system:</b>	Windows 7, 8 and 10. 32 or 64 Bit
--------------------------	-----------------------------------

Supported screen resolutions:	640 x 480, 800 x 600, 1024 x 768, 1280 x 1024
-------------------------------	---

### **1.2.2. Installation Instructions**

Download and follow the instructions to install the AT Server software. The installation program will suggest a folder in which to place the installation files:

c:\Program Files\Voltech Instruments\Voltech AT Server

This directory will be created if it does not already exist. If you want to place the files elsewhere, browse to another directory.

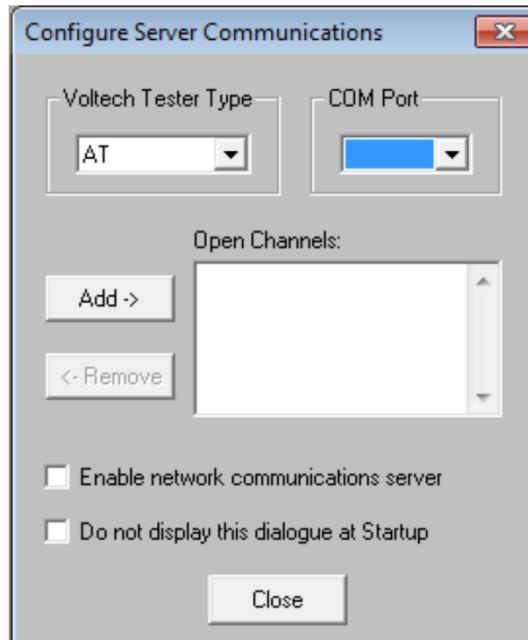
If you have a previous version of the Server software already installed on your PC, the procedure outlined above may also be used to install a new version. None of your program files will be lost. The same Server program is used for all AT Series testers (excluding the discontinued AT3500 and AT1000).

## 1.3. Starting for the first time

### 1.3.1. Running the software

When the installation of the software has been completed, you will see the Server icon in the program group 'Voltech Software'. Double-click this icon with the left mouse button to start the program. The first time the program runs it will show the communications configuration screen.

Note: On some Operating Systems, it may be necessary to run the Server software as an administrator. Right click on the shortcut > Run as Administrator.



For full details of the communications configuration, please see section 1.4.1.

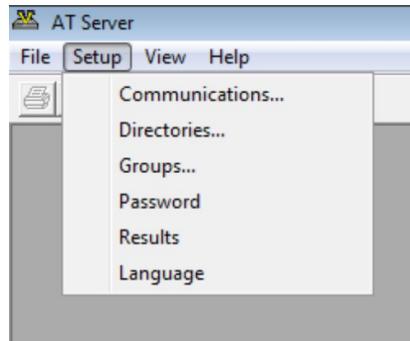
The check box 'Do not display this dialogue at Start-up' exists to prevent it from being displayed each time the Server application is launched. When this option is activated, this dialogue box can be accessed via the 'Setup', 'Communications...' menu.

### 1.3.2. Initial Configuration

Before starting to use the software, it is recommended to configure some of the options. Section 1.4 provides a step-by-step instruction to configure the Server software for proper operation.

## 1.4. Setting up the Options

All Server software options are configured through the Setup menu as shown



### 1.4.1. Communications

The communications configuration dialog may be displayed when the software starts up or by using the **Setup > Communications** option from the Server menu

There are two methods for communicating with the AT5600.

The recommended method is using an Ethernet network connection.

RS232 communications is also an option (as used for the AT3600 and ATi products.)

#### RS232 Communications

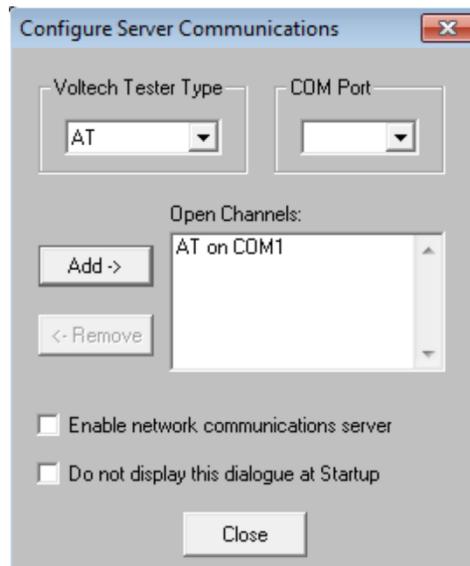
Clicking on clicking the COM port setting of the "Configure Server Communications" dialog will show a list of available PC COM ports as shown.



Select a COM port that is connected to the server port of an AT series tester, and then click the Add button to add this AT to the list of open channels as shown



**WARNING:** If the chosen 'COM' port is reserved by another Windows application; an error message will appear. You should select another port.



If successful, a new results window will be displayed as shown in **Error! Reference source not found**. The title bar of the results window shows which testers results will be shown in this window (in this case, “AT on COM1”). The Server software is now ready to communicate with the AT tester.



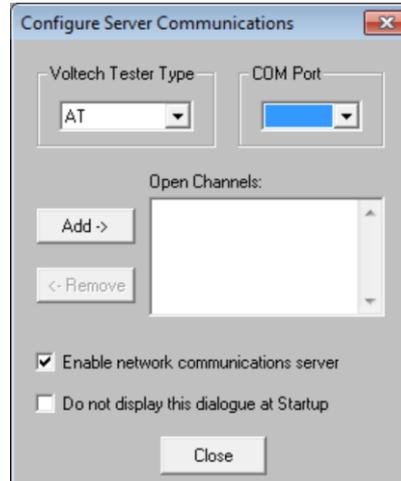
NOTE: So far, no communications will have taken place with the unit. To confirm it is set up correctly, communications must be tested from the AT series tester.

Highlight the open channel, and then click the Remove button in to remove the open channel and close the results window.

## Network Communications

The recommended method for communicating with an AT5600 is using Network Communications. This will communicate over a standard TCP/IP Ethernet network using the Ethernet interface on the computer and AT5600.

To enable network communications, select the option to “Enable network communications server”

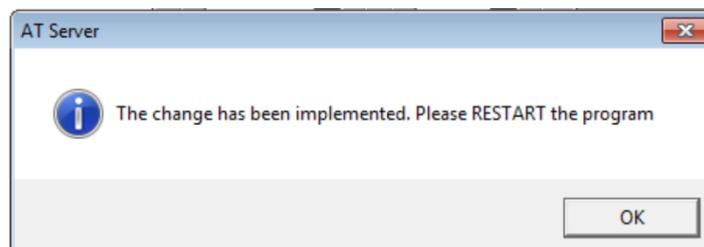


After selecting the option to enable the network communications server, a prompt will ask for the program to be restarted as shown. Click on OK, and close and reopen the Server again.

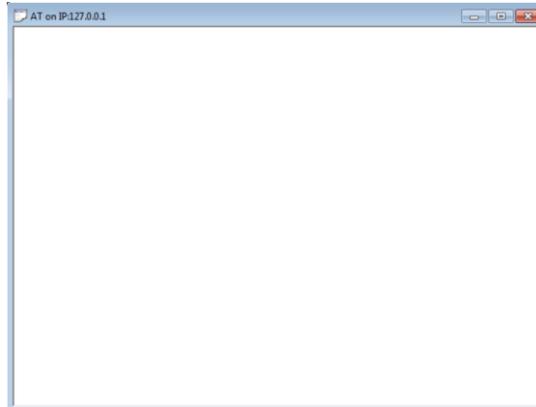


**When starting the server after enabling the network communications server, the computer firewall may give a warning and ask for permission for the server to communicate over the network.**

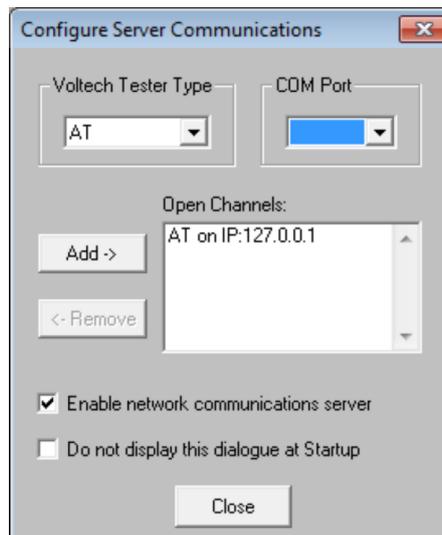
**This must be allowed otherwise communications cannot take place.**



With network communications, the results window is not shown until communication is initiated from the AT5600. When shown, the results window title bar will identify the AT by the network IP address it has connected from.

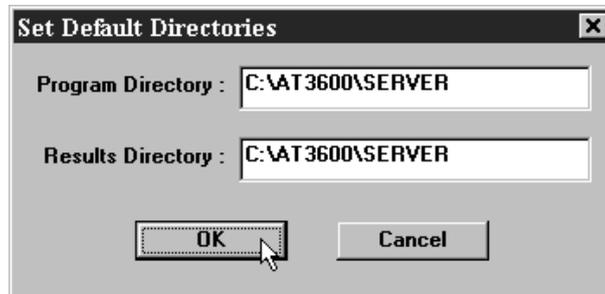


The AT is also identified by its IP address in the communications setting dialog as shown



## 1.4.2. Directories

The directory setup dialogue will prompt you with the current directory. Change these settings if you wish to nominate your own program and results storage directories. The Results directory can only be changed when there is no communication. Highlight and remove Open Channels before attempting to change the Results directory.

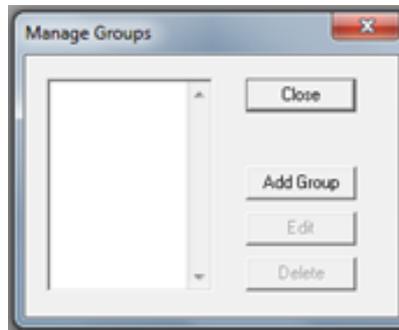


The setting of storage path directories may be altered at any time. The directory setup dialogue may be found under the SETUP section of the menu bar.

**IMPORTANT:** If you change the factory default settings, remember that the Server will only look in the directories specified for test programs or results.

### 1.4.3. Groups

Manages groups of programs Select Setup > Groups from the top menu. The following dialog box will appear.



Once the dialog box appears, you will be able to perform the following operation.

- Close
- Add Group
- Edit
- Delete

Note: The Groups function is not yet implemented in the AT5600.

### 1.4.4. Password

To specify a password to prevent unauthorized deletion of files via the Server.

Leave the password blank to disable this feature.

The set the password of the Server select **Setup > Password**.



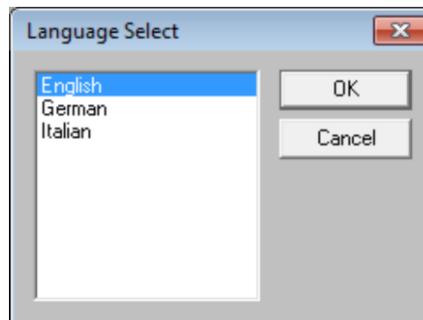
### 1.4.5. Results

To define how the Server will save the Test Results, select **Setup > Results**.



### 1.4.6. Language

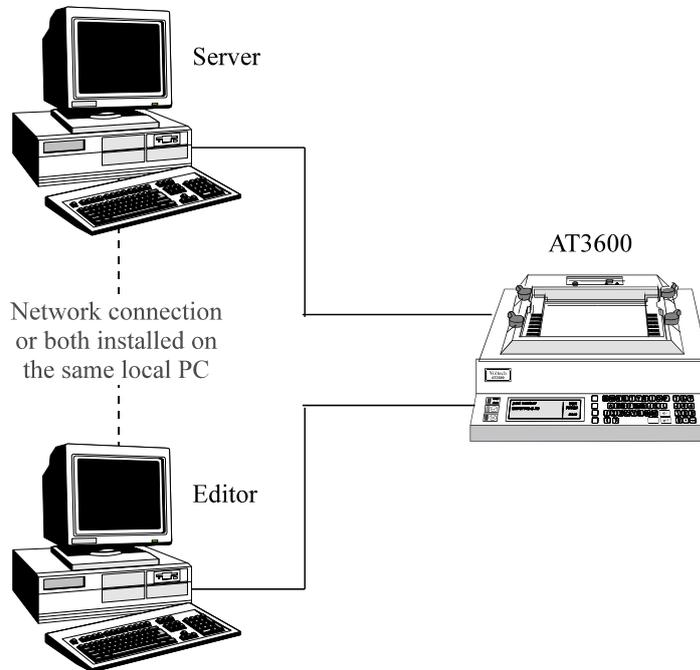
The operating language of the Server can be changed through the **Setup > Language** menu as shown.



Select the desired language and click on OK. A prompt will request the application to be restarted. Close and reopen the Server application for the new language to take effect.

### 1.4.7. Storing Test Programs

After a test program has been created using the Editor application it will be ready for use with your AT series tester. The diagram below shows a potential connection scheme for use with a single tester and two PCs.

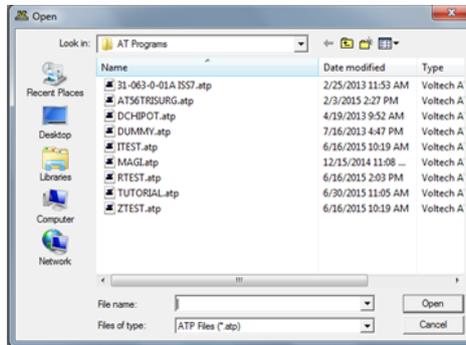


You may copy program files directly into the program directory used by the server application.

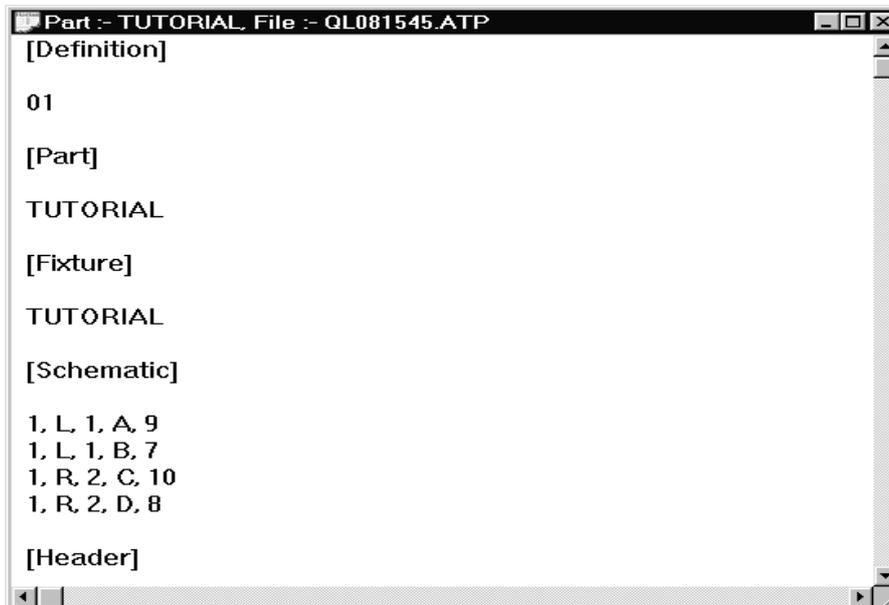
**IMPORTANT:** When a program file is copied to the server and a file with the same name is already resident within the server, the incoming file will replace the previous file.

### 1.4.8. Viewing a Test Program

Each test program has a part name. To view a particular test program, you will need to know its part name. Select FILE, PART and then OPEN from the menu bar. You may select the part to open from the list displayed or change the path and drive to search for other parts.



When you have selected the correct part to open, click on the OK button.

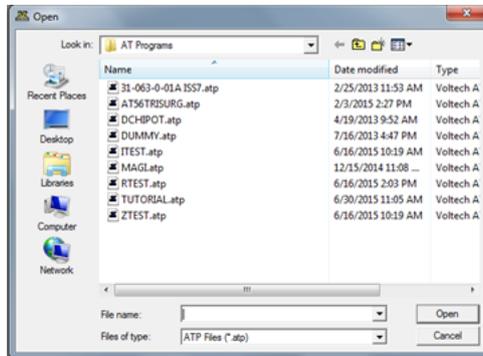


You will now be able to view the text listing of the test program chosen.

### 1.4.9. Deleting a Test Program

To remove apart from the Server program directory, select FILE, PART and then OPEN from the menu bar. Highlight the part to be deleted and press the Delete key on the keyboard or right click the mouse and select Delete.

The program will then be removed from the directory.



**IMPORTANT:** Once a part has been deleted it may not be recovered. Before deleting a part, it is advisable to open and view to ensure you are deleting the correct test program.

## 1.5. Test Results Handling

### 1.5.1. Receiving Test Results

By default, the AT series tester does not automatically send test results back to the server.

Results will only be sent back for test programs that have been set up with the 'Send Results to Server' option enabled (see AT EDITOR manual).

The server receives test results from the tester via the same serial connection used for transferring test programs to the tester. Results handling is a separate function of the server program and is not part of program transfer. Where more than one tester is connected to your server PC, the one server program handles all the transformer tester results as well as acting as the central server.

All results sent by one tester's channel for one day, are collected together and formed into one large file. This file is known as the batch results file.

One batch results file will be generated per open channel, per day. Batch files are stored under the specified results directory; section 1.4.2.

The DOS filename of the batch results file is generated as follows.

c<n><dd><mm><yy>.atr

where:

n channel number

dd day number

mm month number

yy year number

example:

**c1250297.atr** (Channel 1, 25-02-1997)

'atr' is the reserved file extension used by the server to indicate a stored test result file.

**IMPORTANT;** If you are using several installations of the AT SERVER program, and storing all results in one central directory, there is potential for file conflict to occur if the COM ports are not unique.

For example, two instances of the AT SERVER could both be (validly) using COM2 on the local PC they are running on.

Both these AT Servers would then attempt to write a file "C2ddmmy.ATR" to the same results directory.

This can easily be avoided by using unique COM PORTS across all AT SERVER instances. Windows will allow you to use up to 256 different COM numbers which will avoid this potential problem.

What follows is an example of the first part of a batch result file.

Note the DOS file name (c6200697.atr) and the part name ("Part #" "2") are both listed within the file.

```
"VOLTECH AT5600 RESULTS FILE"
"Server Program Version: 2.1"
"Communication Channel: 6"
"AT5600 ID: 0571"
"AT5600 Software Version: 2.2"
"File name: C6200697.atr"
"Test Date: 20 Jun 1997"

"Part #","2"
"Fixture ID","TEST"
"Operator"
"Batch #"
"No Fixture Compensation"

"Result #","Serial #","Time", "Pass/Fail", "", "001  R", "", "", "", "002
", "", "", "", "Minimum", "Maximum", "Reading", "", "Minimum", "Maximum", "R

1, "", "10:44:20", "Pass", "", 59,73,6.776850E+01, "Pass", 3,,3.223899E+00, "P
2, "", "10:44:23", "Pass", "", 59,73,6.776832E+01, "Pass", 3,,3.110549E+00, "P
3, "", "10:44:24", "Pass", "", 59,73,6.776363E+01, "Pass", 3,,3.125640E+00, "P
4, "", "10:44:26", "Pass", "", 59,73,6.777967E+01, "Pass", 3,,3.135034E+00, "P
5, "", "10:44:27", "Pass", "", 59,73,6.777842E+01, "Pass", 3,,3.185198E+00, "P
6, "", "10:44:29", "Pass", "", 59,73,6.778374E+01, "Pass", 3,,3.152627E+00, "P
```

...and so forth for each result until a change of batch or part. If a result window is closed, the connection to the transformer tester will be lost. A warning is issued before connections are removed.

### **1.5.2. Result Display and Analysis**

Four approaches have been provided to allow observation of your test result data:

- On-Line monitoring
- On-Line analysis via a Database
- On-Line analysis via OLE
- Off-Line analysis

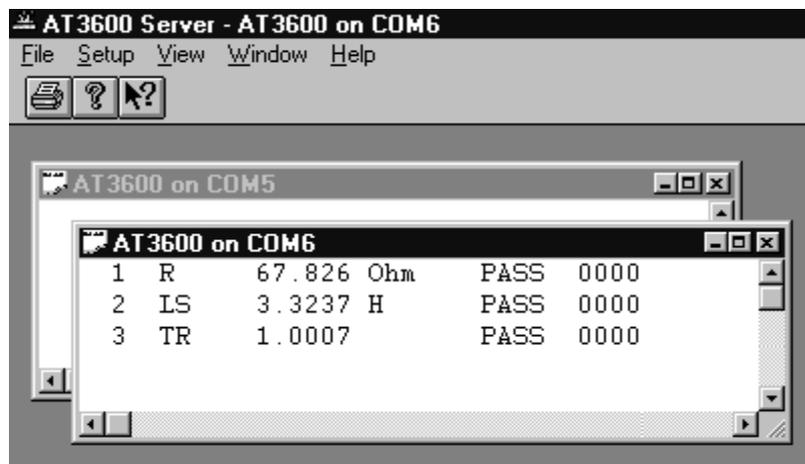
### 1.5.3. On-Line Monitoring

Each time a COM to AT series tester connection is opened using the SETUP, COMMUNICATION menu, an accompanying result monitoring window will be created.

One result monitoring window will be present for each open channel.

When results are sent from the tester, they will be displayed in the monitoring window before passing to the batch result file. The most recent result for each channel will be visible in the window.

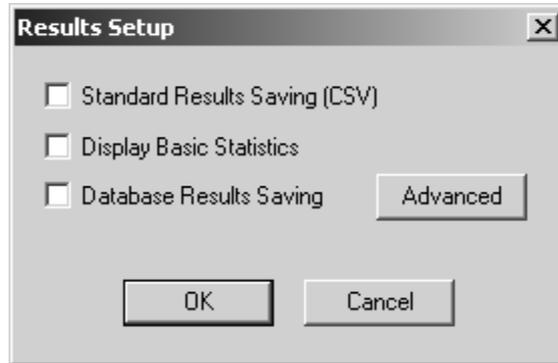
An open result window may be minimized at any time and subsequently maximized using the icons indicated.



### 1.5.4. Batch Statistics On-line Analysis

#### Setting up Batch Statistics

From the 'Set-up' menu, select 'Results' to see the following dialogue:

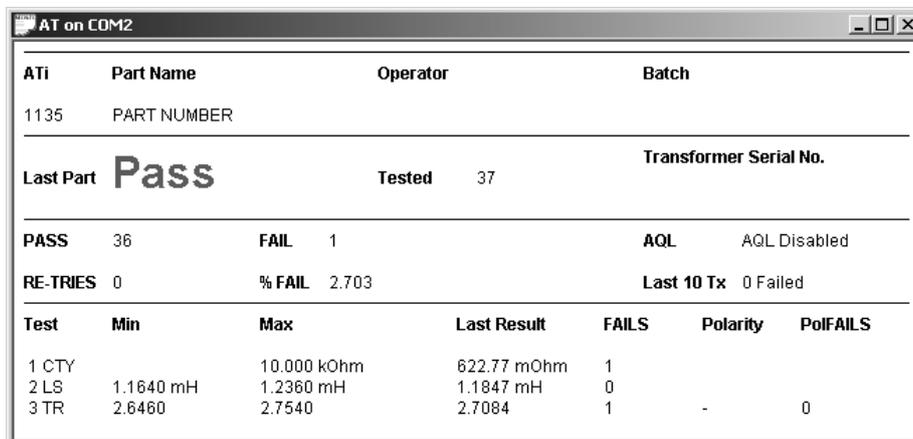


Select 'Display Basic Statistics' to save results to a database.

**For this feature to operate a license for use must first be obtained from Voltech. You will be then given a Key code to enable the option in the AT SERVER.**

All results formats can be saved simultaneously.

Basic statistics displays, on-line, a summary of the results for the current batch of transformers that is being tested on every AT connected to the Server. Results for previous batches can also be opened and the statistics displayed and printed.



ATi	Part Name	Operator	Batch			
1135	PART NUMBER					
<b>Last Part</b>	<b>Pass</b>	<b>Tested</b>	<b>Transformer Serial No.</b>			
		37				
<b>PASS</b>	36	<b>FAIL</b>	1			
<b>RE-TRIES</b>	0	<b>% FAIL</b>	2.703			
		<b>AQL</b>	AQL Disabled			
		<b>Last 10 Tx</b>	0 Failed			
Test	Min	Max	Last Result	FAILS	Polarity	PolFAILS
1 CTY		10.000 kOhm	622.77 mOhm	1		
2 LS	1.1640 mH	1.2360 mH	1.1847 mH	0		
3 TR	2.6460	2.7540	2.7084	1	-	0

The batch statistics window display is automatically shown once this feature is enabled. For each AT connected to the Server the following information is displayed:

The type of AT and its serial number, and the part name, operator and batch.

PASS or FAIL for the last part tested, the total tested so far, and the serial number of the last part tested.

The number that have passed and failed, and the AQL<+> (Acceptable Quality Level) of the transformer.

The number of re-tries that have been attempted, the percentage of failures so far and the number of transformers that failed out of the last 10 tested.

For each test in the program, the test limits, the last result and the accumulated number of fails for each test.

Results may also be exported or printed using the normal options found under the file menu.

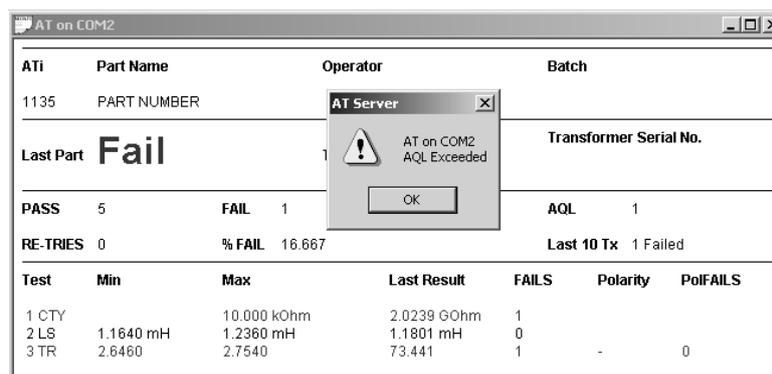
## AQL

AQL (Acceptable Quality Level) settings for production test are defined for each program using the Editor, **Program > Options**. AQL is the level of test failures in a batch that are acceptable. Exceeding the AQL is an indication that there is problem with the quality of the production process, or the quality of materials used. AQL monitoring is an optional feature of the Voltech PC Server which displays live summary results from multiple AT testers. When the AQL level set here is reached or exceeded, the Server will display a warning.

'AQL No. of Parts' is the number of failures in the batch at which a warning will be displayed on the Server. This is most often used when the batch size is always similar, or zero / very small numbers of defects are expected.

'AQL % of Parts' is the percentage level of failures at which a warning will be displayed on the Server. Note that this AQL is only applied after sufficient parts have been tested to make the percentage meaningful. When setting AQL % of Parts to be 1% for example, a warning will only be displayed after  $100/1 = 100$  parts have been tested. Both types of limit may be used at the same time. Reaching any one limit will cause a warning to be displayed on the Server.

When an AQL level is reached, or exceeded the Server will display a warning:



Click on 'OK' to acknowledge the warning. You will be given the option to disable the warning for further parts in the batch. Normally, you would select 'No' to leave the AQL checking active for the rest of the batch. When a new batch is started, the AQL level is reset according to the level entered for the part using the Voltech AT Editor.

## 1.5.5. On-Line Analysis via a Database

### Introduction

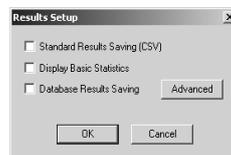
The AT Server software can store test results in an electronic database. This provides a very powerful and versatile method of recording and analysing test results historically or on-line. The Server uses standard Windows methods (including OLE DB and ODBC) to connect to a variety of databases that conform to these specifications.

**For this feature to operate a license for use must first be obtained from Voltech. You will be then given a Key code to enable the option in the AT SERVER.**

Once stored in a database the data may then be analysed using tools available in the database application or tools provided by specialist software packages such as those available for SPC (Statistical Process Control). One example of this is to monitor in real time the trend in a winding resistance to identify a defective drum of copper. You would then be able to rectify the fault during production rather than producing a large stock of QA failed transformers all wound with the defective wire.

### Setting Up Database Results Storage

From the 'Setup' menu, select 'Results' to see the following dialogue:

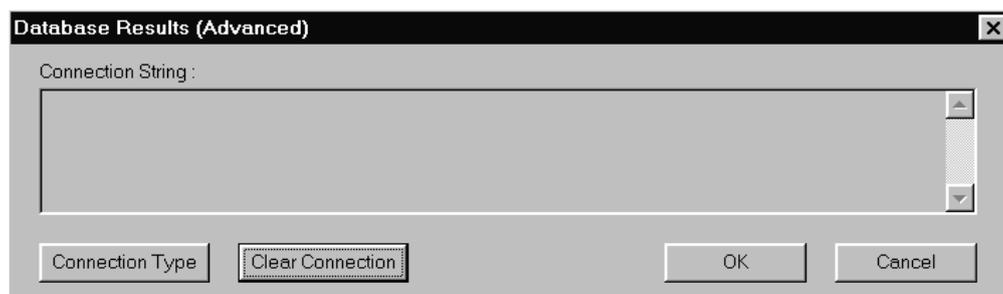


Select 'Standard Results Saving' (CSV) to save all results to a spreadsheet compatible file.

Select 'Database Results Saving' to save results to a database. For this feature to operate properly, a connection to the database must be set up first.

CSV and database format results can be saved simultaneously.

Click on 'Advanced' to set up your database connection:

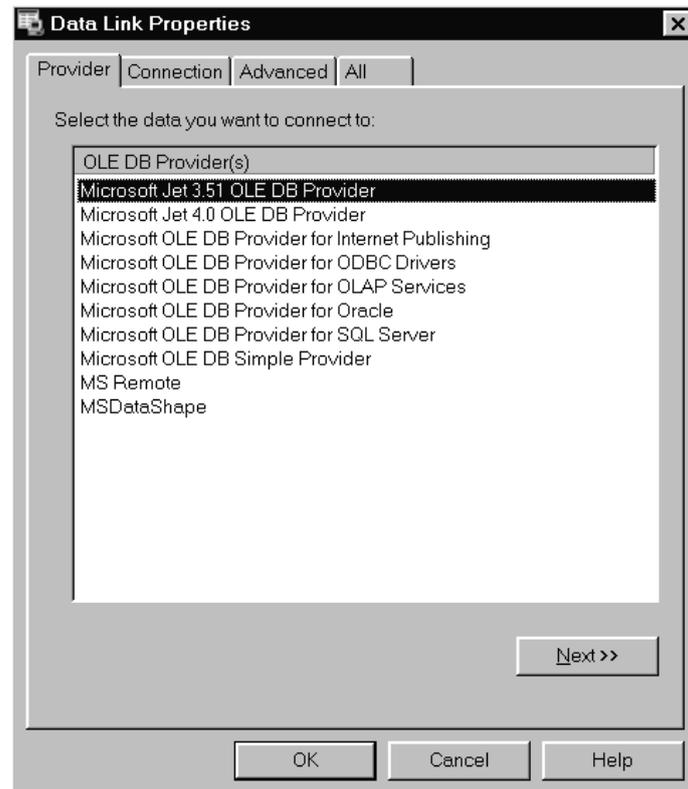


To connect to your database, the Server software uses a connection string that contains (amongst other things) the name of a provider, the name and location of

the database and a valid database username and password (if required). You may need to consult with the database or network administrator to obtain this information.

If a connection has already been established, you will see the connection string in the window inside this dialogue. If you wish to remove an existing connection string, click on 'Clear Connection'.

To establish a new connection, click on 'Connection Type'.



The 'Data Link Properties' dialogue is a standard one that is supplied by Windows. You should choose the provider that will provide the connection to the type of database that you are using.

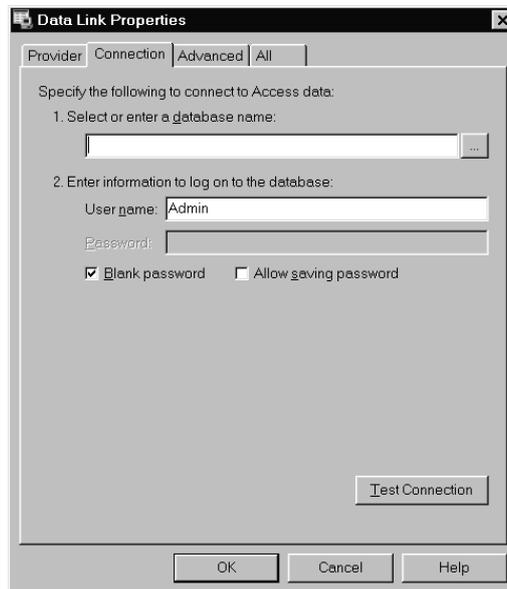
Several different types of provider are installed along with the Server software; others may be available if you have a database application installed on the same PC as the Server. If the provider you need is not available and the database is installed on another PC, ask the database administrator for the database client tools for the Server PC.

Typical provider types are:

Database	Provider
Microsoft Access 97	Microsoft Jet 3.51
Microsoft Access 2000	Microsoft Jet 4.0
Oracle	Microsoft OLE DB Provider for Oracle
SQL Server	Microsoft OLE DB Provider for SQL Server

In addition, it is possible to connect to ODBC compliant databases (including Microsoft Access) using the Microsoft OLE DB Provider for ODBC Drivers.

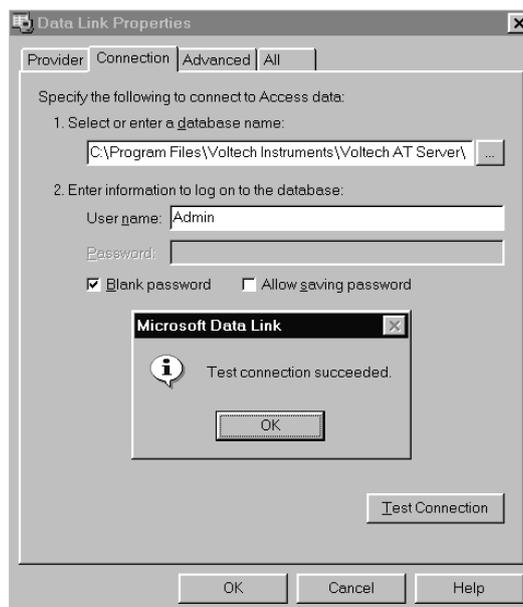
Click on 'Next' when you have chosen the correct provider.



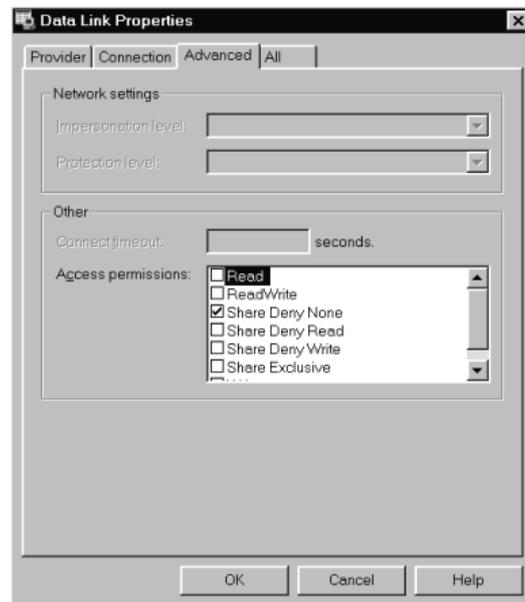
Select the database name that you will be using by clicking on 'Select'. If you wish to create a new database for AT Server results, do so now using your database program.

The default username 'Admin' with no password may be adequate. Consult your database administrator if you need a specific username and password.

When you click on 'Test Connection' you should see something similar to:

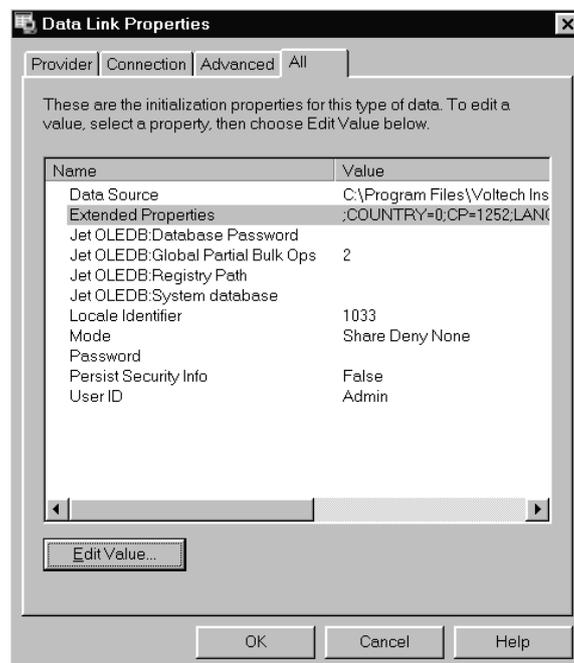


Next, click on 'OK' of the 'Test connection succeeded' dialogue and select the 'Advanced' tab.

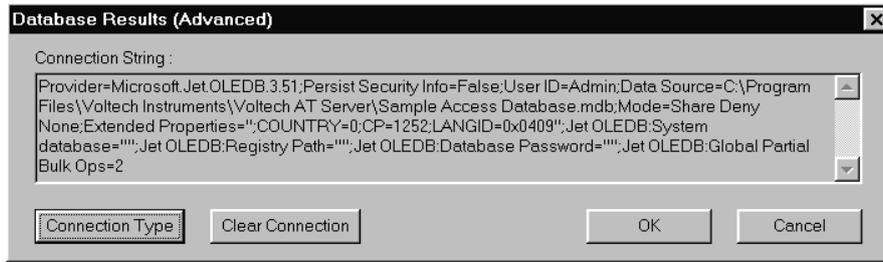


Check that the 'Access permissions' for the database are as required. 'Share Deny None' is the default and is usually the most suitable setting.

Under the 'All' tab all of the individual settings can be reviewed and adjusted if necessary.



Click on OK to complete the setting of the data link properties. A typical AT Server advanced database results dialogue would be:



Click on OK to return to the results dialogue. At this point the AT Server will automatically create the required database tables if they do not already exist.

Ensure that 'Database Results Saving' is checked and click on 'OK'. Whenever a results window is open, the results will now be stored in the selected database.

## Database Tables

The server software automatically creates the following tables in the selected database (if they do not already exist) and stores data within the tables using the structure defined here.

The tables store all the information that is available concerning the results (including time, date, limits, operator etc.) in a very compact and efficient structure.

Note that new data is not necessarily appended to the end of a table; it may appear anywhere within.

Table:	tblVoltechATUnitID	
Field	Format	Notes
<b>ATUnitID</b>	<b>Text (25)</b>	<b>Index ID</b>
UnitType	Text (10)	AT5600, AT3600, AT1600, ATi
UnitID	Text (10)	Unit Serial Number
FirmwareID	Text (10)	Unit Firmware Version

Table:	tblVoltechRunID	
Field	Format	Notes
<b>RunID</b>	<b>Text (25)</b>	<b>Index ID</b>
Date	Date/Time (Short Date)	Date of Test Run
Time	Date/Time (Long Time)	Time of Test Run
<b>ATUnitID</b>	<b>Text (25)</b>	<b>As tblVoltechATUnitID.ATUnitID</b>
PartID	Text (25)	Part name
FixtureID	Text (25)	Fixture name
OperatorID	Text (25)	Operator name
BatchID	Text (25)	Batch name
TransformerSerialNo	Text (25)	Transformer serial number
OverallResult	Text (10)	PASS/FAIL/RE-RUN/ABORT Overall test pass or fail result and Re-Run result.

<b>Table:</b>	<b>tblVoltechResults</b>	
<b>Field</b>	<b>Format</b>	<b>Notes</b>
<b>ResultID</b>	<b>Text (25)</b>	<b>Index ID</b>
<b>RunID</b>	<b>Text (25)</b>	<b>As tblVoltechRunID.RunID</b>
<b>TestID</b>	<b>Text (25)</b>	<b>As tblVoltechTestID.TestID</b>
TestNo	Number (Long Integer)	Test number as in test program
OC_Comp	Text (5)	YES, NO, N/A Open Circuit Compensation status
SC_Comp	Text (5)	YES, NO, N/A Short Circuit Compensation status
Load_Comp	Text (5)	YES, NO, N/A Load Compensation (future development)
Result	Number (Single)	Actual test result
TestStatus	Text (5)	DSP Status for test
PassFail	Text (5)	PASS/FAIL Individual test pass or fail result

<b>Table:</b>	<b>tblVoltechTestID</b>	
<b>Field</b>	<b>Format</b>	<b>Notes</b>
<b>TestID</b>	<b>Text (25)</b>	<b>Index ID</b>
TestMnemonic	Text (10)	Individual test mnemonic
TestUnits	Text (10)	Individual test units
CheckType	Text (5)	NONE/MIN/MAX/DIFF/POL Type of test limits (see *)
MinLimit	Number (Single)	Minimum limit for test (see *)
MaxLimit	Number (Single)	Maximum limit for test (see *)

\* Notes:

- (CheckType = NONE) If there are no Min or Max limits,  
MinLimit = 0 and MaxLimit = 0
- (CheckType = MIN) If there is a Min limit only, MaxLimit = 0
- (CheckType = MAX) If there is a Max limit only, MinLimit = 0
- (CheckType = DIFF) If there is a Min and Max limit
- (CheckType = POL) Polarity test  
If Positive Polarity test, MinLimit = 0 and MaxLimit = 1  
If Negative Polarity test, MinLimit = -1 and MaxLimit = 0

## INDEX IDS

The index IDs for the database storage (**shown in BOLD in the above tables**) are strings based on the following pattern:

- PPyyyyMMddHHmmssRRRRRRRR

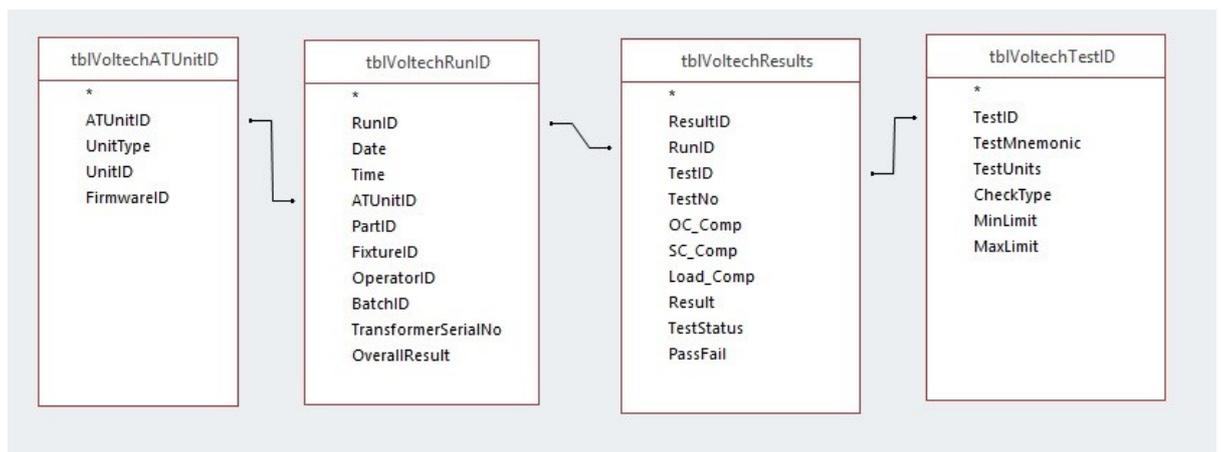
where,

- PP is two digits representing the COM port number that the AT is on.
- yyyy is the four-digit year the entry was created.
- MM is the two-digit month the entry was created.
- dd is the two-digit day the entry was created.
- HH is the two-digit hour (0-23) the entry was created.
- mm is the two-digit minute the entry was created.
- ss is the two-digit second the entry was created.
- RRRRRRRR is a 32-bit random number expressed in hexadecimal notation with characters 0-9 and A-F.

## Table Relationships

The relationship between the four tables is relatively simple.

It is best expressed using the below query diagram from ACCESS



## **Hints and Tips**

### **TR TRL VOC VOCX LVOC results**

These tests appear to contain two results, as they have

a, MIN / MAX / DIFF results (numerical measurement)

and

b, an optional POLARITY result (+1 or -1)

If you are performing numerical analysis (Min/Max/Standard Deviation etc) on these, remember to remove the CheckType = "POL" records from your query

### **Best Practice**

Its recommended to keep 1 database with the 4 tables connected to the ATs to collect the data, and then use a separate "front end" database, linked to the first, to run query's and analyse the data

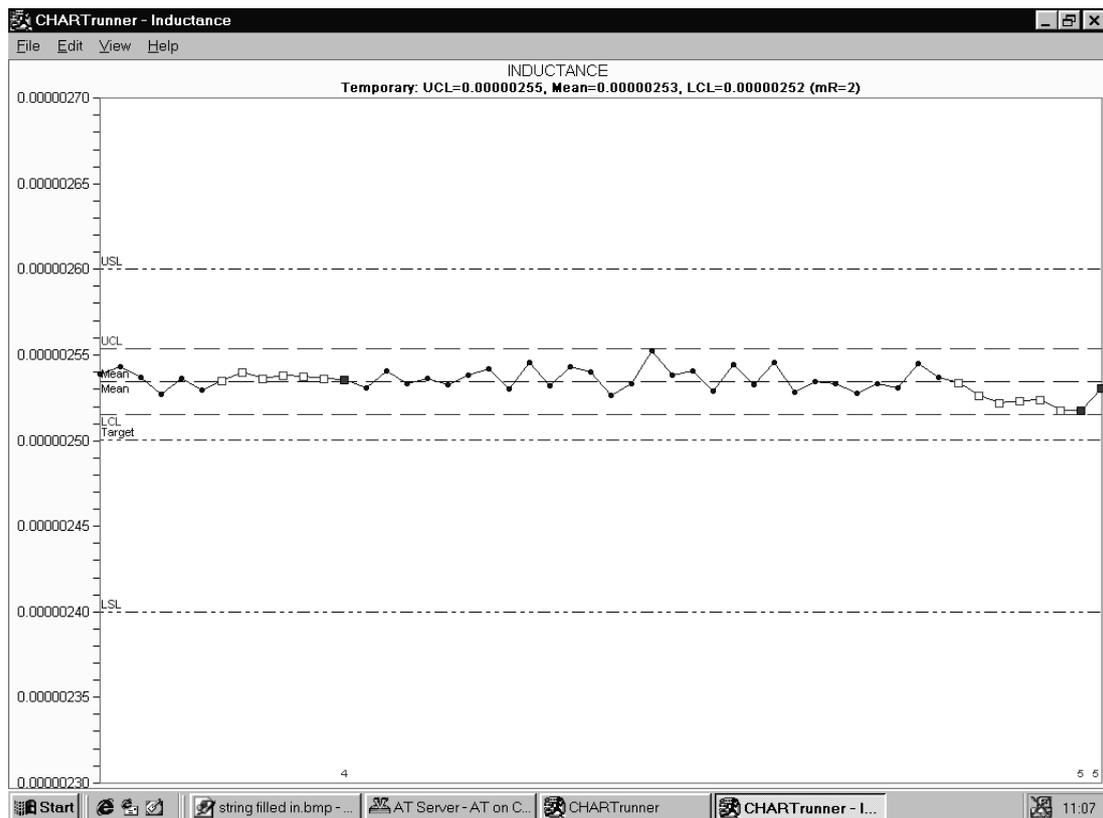
In this way you can develop a bespoke company front end which reads and analyses data, which is separate from the data, and easily distributable to many users.

## Charting and Analysis Using SPC/SQC Software

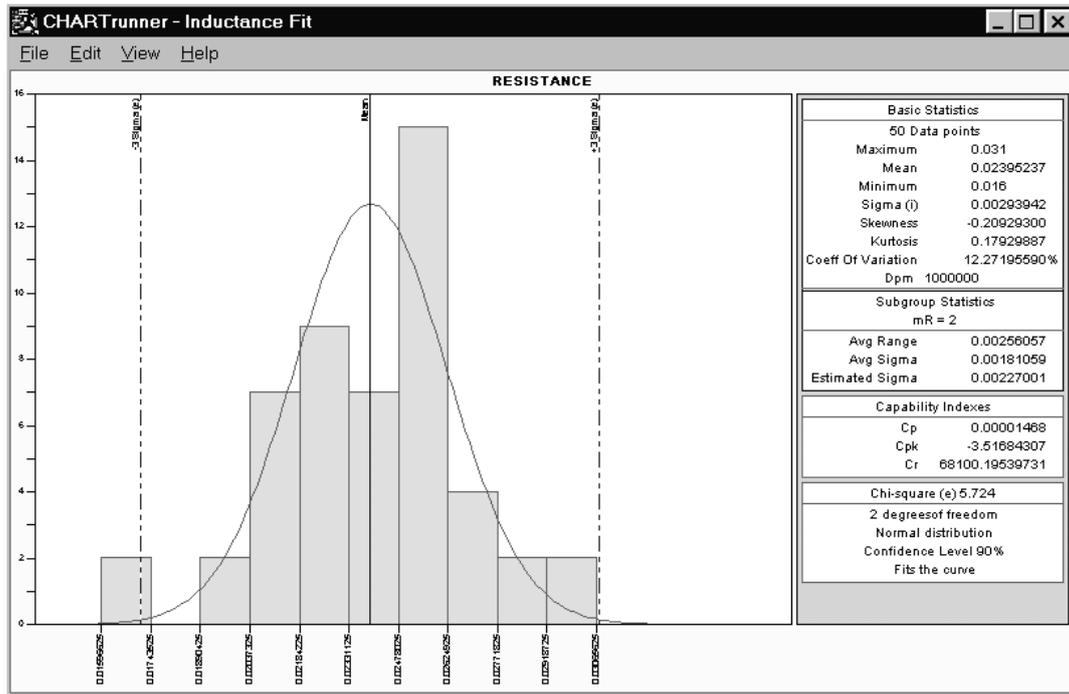
Once an AT Server to database connection has been made, it is possible to connect third party analysis software to the same database for live on-line (or offline) data analysis.

The example reports presented here was created using PQ Systems ChartRunner, but many other charting and analysis programs are available.

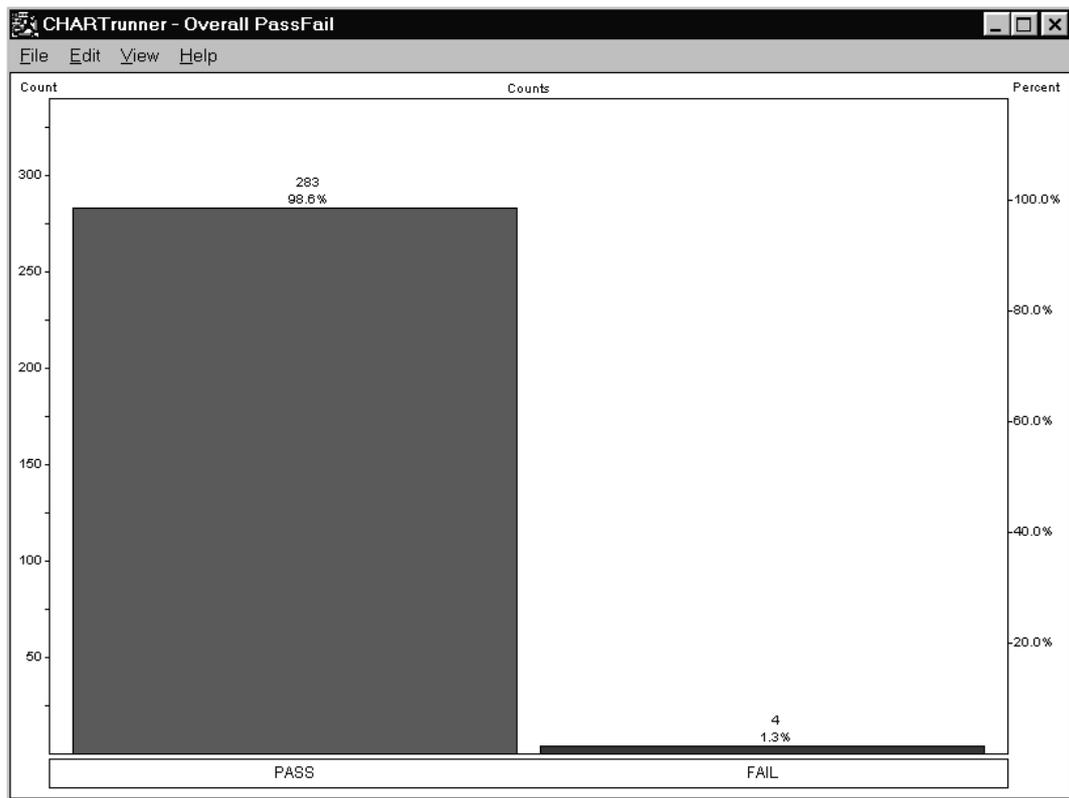
Voltech does not recommend one package more than any other. You should choose to use the one that best meets your needs.



Inductance Trend Analysis



Resistance Distribution Analysis



Overall Pass/Fail Analysis

### **1.5.6. On-Line Analysis via OLE**

The MS Windows environment provides a system of transferring data between active applications in real-time known as Object Link Embedding (OLE). OLE2 is the latest version of the transfer standard and is fully supported by the Server application.

Embedded data links allow test results received by the server to automatically pass into a second application, which may be your own custom software or a spreadsheet such as Microsoft Excel. Server OLE commands also allow you to load and execute test programs. OLE is the analysis and control method preferred by users who wish to operate the AT under software control.

OLE is already configured ready-to-go within the Server application. Your client application will need to be setup to request data from the Server. Refer to your own application handbook for details of OLE setup.

**NOTE This functionality is included for compatibility with older versions of the Voltech AT Server. It is not recommended for new applications, as Microsoft will soon depreciate OLE.**

**IMPORTANT: Attempting to make more than ten data requests per second may result in data loss or corruption.**

NOTES:

The automation object is called '**Server.Results**'

The exposed OLE automation methods are as follows

**GetVariantResult** (LPCSTR ComPort, short TestNumber)

where: LPCSTR ComPort is a pointer to a string ComPort

**ComPort**= "COM2" ... "COM9" = the port connected to the AT Series Tester which gives the required result

**TestNumber** =Integer number for test within program the number returned is a float representing the value of the result measured or a zero if no result was available.

This call returns a result on success or a failure string:

<b>"INVALID COM PORT"</b>	no valid com port found
<b>"COM PORT IS NOT OPEN"</b>	com port requested not connected to an AT Series Tester
<b>"INVALID RESULT NUMBER"</b>	the requested result number is out of range
<b>"NO RESULT"</b>	no updated result data available

Further Voltech Server OLE functions are described below. They use the same format and failure strings as 'GetVariant Result'. Only function specific responses and return values are shown.

**NewDataAvailable** (LPCSTR ComPort)

Returns TRUE or FALSE (Boolean 1 or 0 as variant) depending on whether new data is available. If this function returns true it means that new data can be read back via the OLE. If false, then no new data has come in since the last read

**GetPartID** (LPCSTR ComPort)

Returns the part name on success or a failure string:

<b>"NO PART ID"</b>	no valid part name found
---------------------	--------------------------

**GetFixtureID** (LPCSTR ComPort)

This call returns the fixture name on success or a failure string:

<b>"NO FIXTURE ID"</b>	no valid fixture name found
------------------------	-----------------------------

**GetOperatorID** (LPCSTR ComPort)

This call returns the operator name on success or a failure string:

“NO OPERATOR ID”                      no valid operator name found

**GetBatchID** (LPCSTR ComPort)

This call returns the batch number on success or a failure string:

“NO BATCH ID”                      no valid batch number found

**GetFirmwareID** (LPCSTR ComPort)

This call returns the firmware version number on success or a failure string:

“NO FIRMWARE ID”                      no valid firmware version found

**GetUnitID** (LPCSTR ComPort)

This call returns the tester serial number on success or a failure string:

“NO UNIT ID”                      no valid AT serial number found

**GetTransformerSerialNo** (LPCSTR ComPort)

This call returns the transformer serial number on success or a failure string:

“NO SERIAL NO”                      no valid transformer serial number found

**GetResultsFilename** (LPCSTR ComPort)

This call returns the results file filename on success or a failure string:

“NO RESULTS FILE”                      no valid filename found. (No results yet?)

**GetResultPolarity** (LPCSTR ComPort, short TestNumber)

“NO POLARITY”                      Test has no polarity result

“+”                      Polarity is positive

“-”                      Polarity is negative

**GetPolarityPass** (LPCSTR ComPort, short TestNumber)

“NO POLARITY”	Test number has no polarity result
“PASS”	Polarity result has passed
“FAIL”	Polarity result has failed

**GetTestPass** (LPCSTR ComPort, short TestNumber)

“PASS”	Test result has passed
“FAIL”	Test result has failed

**GetOverallPass** (LPCSTR ComPort)

“PASS”	All tests have passed
“FAIL”	One or more tests have failed

**GetNumberOfTests** (LPCSTR ComPort)

Returns a short integer variant type giving the number of tests in the program. Use for possible determination of the maximum value for TestNumber in other functions.

**GetMinLimit** (LPCSTR ComPort, short TestNumber)

Float value giving the minimum limit for the test TestNumber.

“NO LIMIT”	This test has no minimum limit.
------------	---------------------------------

**GetMaxLimit** (LPCSTR ComPort, short TestNumber)

Float value giving the maximum limit for the test TestNumber.

“NO LIMIT”	This test has no maximum limit.
------------	---------------------------------

**GetTestStatus** (LPCSTR ComPort, short TestNumber)

Returns four-digit hex string ("####") giving the test status as seen on editor results screen.

**GetTestMnemonic** (LPCSTR ComPort, short TestNumber)

Returns the test mnemonic as a string, i.e. "LBAL".

**GetTestUnits** (LPCSTR ComPort, short TestNumber)

Returns the test units as a string, i.e. "Ohm".

**GetCompensationStatus** (LPCSTR ComPort)

"NO COMPENSATION STATUS"	No status received yet.
"No Fixture Compensation"	Program has not been compensated.
"Fixture Compensation: successful"	Compensation all OK.
"Fixture Compensation: fail" + ExtraString	Compensation has failed.

**GetUnitType** (LPCSTR ComPort)

Returns "AT5600", "AT3600", "AT1600" or "ATi" depending on the type of instrument connected to the ComPort.

**GetResult** (LPCSTR ComPort, short TestNumber)

The number returned is a float representing the value of the result measured or a zero if no result was available.

This section shows 3 available functions for controlling an attached instrument via the server.

This is to allow basic instrument control in applications where the AT is being used entirely remotely.

The functions available allow a program to be loaded into an AT and give the ability to run and stop it.

#### **Variant LoadProgramToAT** (LPCSTR ComPort, LPCSTR ProgName)

"PROGRAM DOES NOT EXIST"

Program is not in the servers' program folder.

"INVALID PROGRAM"

Program fails AT test program validation.

"LOADING"

Returned when command is first sent to server, and every subsequent time it is sent until loading finishes.

"LOADED"

The program has been successfully loaded, this will continue to be returned, and the program will not load until either StopAT() is called or the server receives results on that COM port.

#### **Variant RunProgramInAT** (LPCSTR ComPort)

"NO PROGRAM"

The AT has no program to run.

"SENT"

Returned when the program has started to run and every subsequent time the function is called until the server receives results.

"DONE"

The program has finished running; results will be available. If the function is called after it has returned this value, it will cause the program to be run again.

#### **Variant StopAT** (LPCSTR ComPort)

"STOPPED"

The AT must stop whatever test it is running.

## **Example of On-Line Analysis**

This section contains a simple example to illustrate how you may use OLE to transfer the measured results into a spreadsheet as they happen in real-time.

The spreadsheet used in the example is Microsoft Excel 97, and the module of code illustrated is written in the Visual Basic supplied with the spreadsheet.

Before attempting to create the spreadsheet, set up the AT and server:

- Use the editor to create a program containing 5 tests, where test number 3 measures a winding resistance.
- Ensure that the program option 'Send Results to Server' is set to 'On'.
- Transfer the program to the directory being used by the server for storing programs.
- Ensure that the server program is running, and that the COM Port to be used has been enabled.
- Connect the AT to the server PC and use the PROGRAM - SET-UP menu to set the origin to SERVER.
- At the AT Top Level menu, press the EXECUTE soft-key, and type in the name of the program.
- Run the program and check that the correct results are displayed in the on-line monitoring window.

Leave the Server program running, and on the same PC, start Microsoft Excel 97. A new (empty) spreadsheet will be created.

From the Excel top level menu bar, select: Tools > Macro > Macros

Enter the name 'OLE\_Demo' and then create.

Excel will then bring up a module sheet labelled 'Module 1'. Type the following code into this module sheet:

```
*****
'OLE_Demo Sub
'
'Procedure to demonstrate linking AT results into an Excel spreadsheet
'in real time using OLE.
'
'The procedure will read the third test in the program and put the result
'into a cell on the third row of the spreadsheet.
'
'The procedure will do this for a total of 3 program runs, and then exit.

Sub OLE_Demo ()

    Dim objServerResult As Object
    Dim vntResult(1 To 99) As Variant

    Set objServerResult = CreateObject("Server.Results")

    nTest = 3
    nNumRuns = 3
    sComPort = "COM2"

    'Dummy read to clear any previous result
    vntResult(nTest) = objServerResult.GetVariantResult(sComPort, nTest)

    MsgBox ("Press the RUN key on the AT5600")

    For nRun = 1 To nNumRuns          'Get the results
    Do
    vntResult(nTest) = objServerResult.GetVariantResult(sComPort, nTest)
    Loop Until (vntResult(nTest) <> "NO RESULT") And _
    (vntResult(nTest) <> "INVALID RESULT NUMBER")

    Cells(3, 5 + nRun) = vntResult(nTest)    'Put result in spreadsheet

    Next nRun

    MsgBox ("Done")

End Sub
```

When you have finished typing the code, return to the 'Sheet 1' of the spreadsheet.

To run the code, again from the Excel top-level menu bar, select:

**Tools > Macro**

At the dialog box, select the name of the macro 'OLE\_Demo', and click on the RUN button.

The code will now start to run, and after a few seconds, you will see the first message box defined in the code:



Click on the OK button. When you have pressed the RUN soft-key and the first result has been read, the message box will disappear, and you will then see the first result entered into cell F3 of the spreadsheet.

Continue pressing the RUN soft key until the second message box is seen.



Click on the OK button to finish running the code.

### **Additional Notes for the Above Example**

If there appear to be any problems while the code is running, press the escape key. This will halt the execution and bring up a dialog box to allow you to either exit from the code execution, or to debug your code, for example by single stepping through it.

Once the results have been placed into cells in the spreadsheet, standard Excel functions may be used to perform further analysis on the data, or to present the data in different ways. As an example, you may wish to use the Excel Chart Wizard to produce a graph of the results. Alternatively, you may wish to add further code to the example shown above, as another way of extending the functionality. For guidance of what may be possible, please refer to the Microsoft Excel user manuals.

In the code example shown above, it is worth noting the following points:

- vntResult(1 To 99) is declared as a variant type because the GetVariantResult method can return either a numerical value corresponding to a valid result, or a string corresponding to an error message.
- The line

```
Set objServerResult = CreateObject("Server.Results")
```

creates the server.Results automation object within Visual Basic.

When trying to read the result with

Do

```
    vntResult(nTest) = objServerResult.GetVariantResult(sComPort, nTest)  
Loop Until (vntResult(nTest) <> "NO RESULT") And _  
           (vntResult(nTest) <> "INVALID RESULT NUMBER")
```

Two tests for 'No Result' and 'Invalid Result Number' are included.

The GetVariant method will normally read back 'No Result', until a measurement has occurred. The next time it is executed after a new result has been sent to the server, GetVariant will read back the actual result value. The act of doing this will set a flag in the Server to say that the data has been read, so that any further executions of the GetVariant statement before another result has been transferred to the server will then return the 'No Result' reply.

Although the test number is fixed (= 3) in this example, the 'Invalid Result Number' test is also included. This is because the GetVariant method can be activated at exactly the same time as the results are being transferred from the AT to the server. Depending on the speed and other parameters of your PC, there may be a small time during the transfer when the result number requested can be read as invalid. Including this test in the loop will trap out this error.

Instead of using the GetVariantResult function itself to determine if new data is available, the NewDataAvailable function can be used. This function is recommended for use when more than one item of data is to be read after new data is available. For example, the NewDataAvailable function can be read in the do loop until it returns true, after which any other OLE function can be called to gather their return values knowing that all the data will be valid. Using the NewDataAvailable function also means that checking for 'INVALID RESULT NUMBER' and 'NO RESULT' return values is no longer necessary.

### **1.5.7. Off-line Analysis**

If you have no requirement for real-time analysis but are interested in post-test analysis of test results, off-line data analysis should be adopted. An example of this would be to view the results file created the previous day.

The batch result file is stored in a text-based Comma Separated Variable format (CSV).

A spreadsheet is an ideal environment to analyse and manage your test data. Almost all spreadsheet applications are able to import a CSV file.

Imported test results may be graphed, calculated, edited and printed like any other data. Using a package such as Microsoft Excel, macros may be setup to sort and extract the appropriate result sections, calculate and compile reports and finally produce a printout for records.

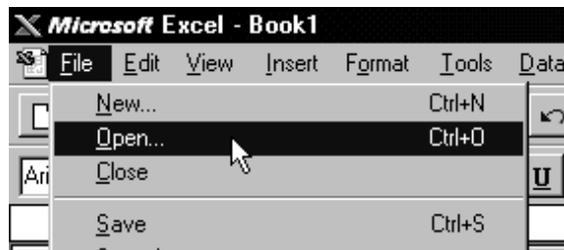
Refer to your application handbook for details of importing and manipulating data.

#### **Example of Off-Line Analysis**

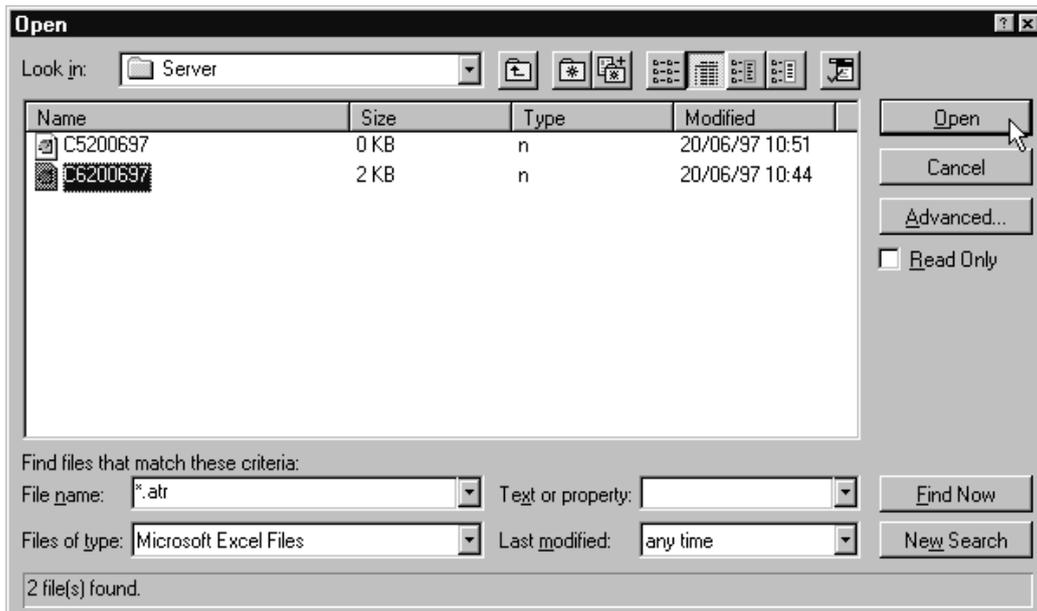
The spreadsheet used in the example is Microsoft Excel 97.

From the Excel top level menu bar, select:

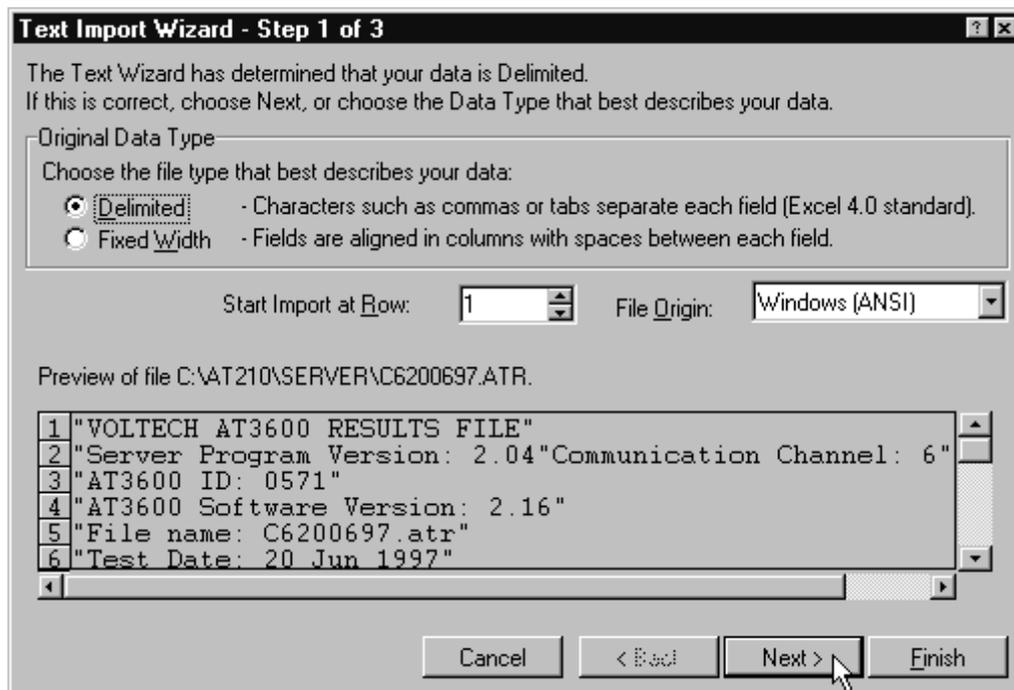
**File > Open**



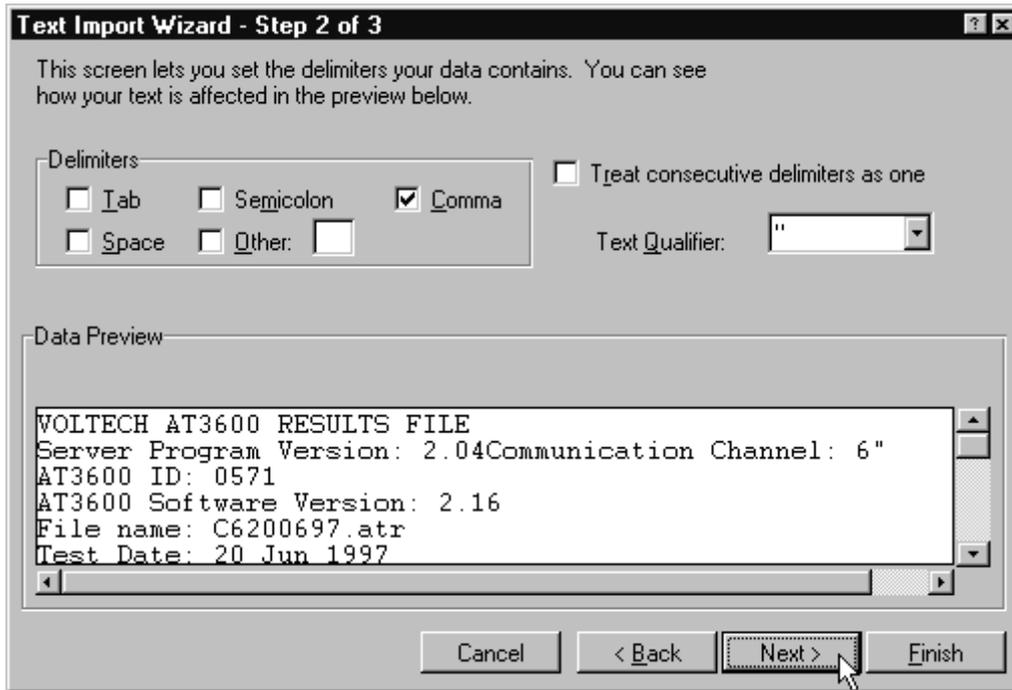
Excel will then bring up the 'Open File' dialogue box. Select the results file you wish to view and click on 'Open'.



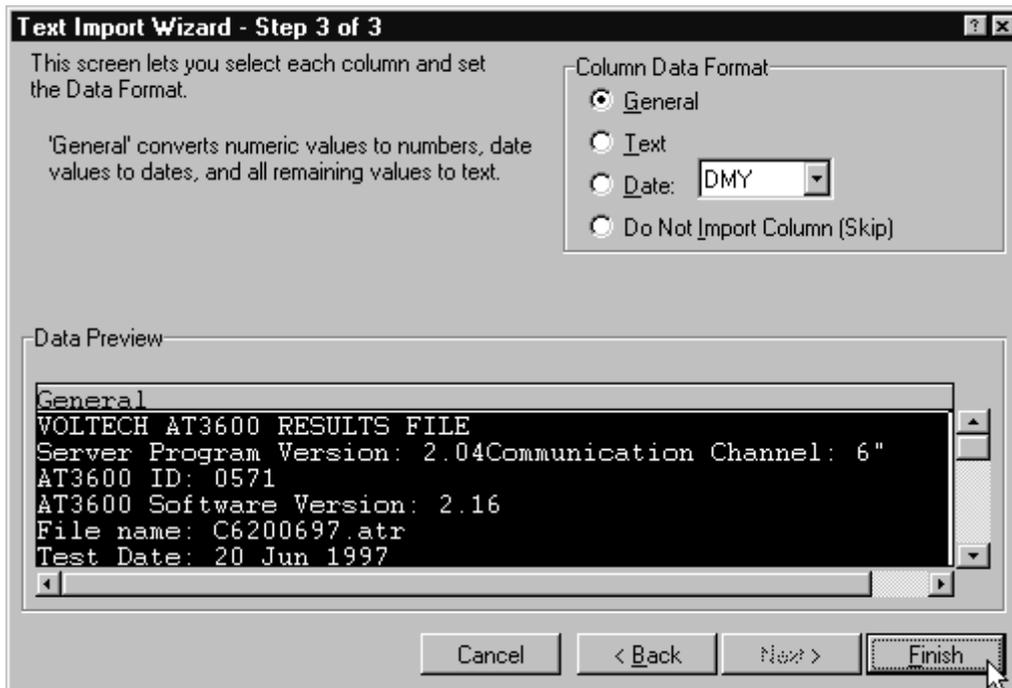
After clicking 'Open', the Text Import Wizard dialog box will appear:



Ensure that the 'Delimited' option is checked and click on 'Next >' to move to step two.



In the 'Delimiters' section of this screen, ensure that only the 'Comma' option is checked. Click on 'Next >' and proceed to step three:



Check that the 'Column Data Format' option is checked as 'General' and click on finish.

The results will now be displayed as shown below:

The screenshot shows a Microsoft Excel spreadsheet with the following data:

VOLTECH AT3600 RESULTS FILE											
1	VOLTECH AT3600 RESULTS FILE										
2	Server Program Version: 2.04										
3	Communications Channel: 6										
4	AT3600 ID: 0571										
5	AT3600 Software Version: 2.16										
6	File name: C6200697.atr										
7	Test Date: 20 Jun 1997										
8											
9	Part #	2									
10	Fixture ID:	TEST									
11	Operator										
12	Batch #										
13	No Fixture Compensation										
14											
15											
16	Result #	Serial #	Time	Pass/Fail	001 R			002 LS			
17					Minimum	Maximum	Reading	Minimum	Maximum	Reading	
18											
19	1		10:44:20	Pass	59	73	6.78E+01	Pass	3	3.22E+0	
20	2		10:44:23	Pass	59	73	6.78E+01	Pass	3	3.11E+0	
21	3		10:44:24	Pass	59	73	6.78E+01	Pass	3	3.13E+0	
22	4		10:44:26	Pass	59	73	6.78E+01	Pass	3	3.14E+0	
23	5		10:44:27	Pass	59	73	6.78E+01	Pass	3	3.19E+0	
24	6		10:44:29	Pass	59	73	6.78E+01	Pass	3	3.15E+0	
25											
26											
27											

You may need to adjust the Cell number format of the spreadsheet, and the width of the columns, in order to display the required number of decimal places.

Using functions built into the spreadsheet, the data obtained may now be used to create graphs and bar charts for a visual representation of your results. As an example, to produce a graph, select a group of results from one column; then click the Chart Wizard button and use this to create the graph you require in the normal way. For additional help on what may be possible, please refer to your spreadsheet user manual