ATEasy 5.0

TEST EXECUTIVE AND DEVELOPMENT STUDIO

- 32-bit object-oriented programming framework for Windows® 9x/ME and NT/2000/XP
- Rapid Application Development (RAD) environment enables short development cycles
- Instrumentation/Test System-like modular structure for easy development and maintenance
- Built-in Application Builder generates royalty-free run-time executables (EXE files)
- Built-in Test Executive and Profile editor for full control of test execution, sequence, and conditions
- User-defined, plain language commands offer program legibility for easier maintenance
- Automatic generation of HTML and text-based test logs instantly display test results
- Built-in Configuration Management tools allow simplified project management
- Connectivity with Microsoft® Source Safe enables direct check in, out, add, and compare files
- Multithreading support allows simultaneous execution of multiple code segments
- Visual Basic-like Form Editor generates graphical user interfaces, menus, and controls
- Event-driven programming support facilitates more efficient program execution
- Supports ActiveX, OCX, OLE, DDE, DLL, .NET components/assemblies and C/C++ header files (for automatic DLL functions and type declaration)
- Vendor-independent open architecture supports PXI, PCI/cPCI, VXI, GPIB, TCP/IP, RS-232, ISA, and more
- Supports VXI Plug-and-Play, Function Panel (.fp) drivers, Virtual Instruments (.vi/.llb), and IVI drivers

DESCRIPTION

ATEasy is a rapid application development framework for functional test, ATE, data acquisition, process control, and instrumentation systems. ATEasy provides all the necessary tools to develop and maintain software components, from instrument drivers to complex test programs. It is designed to support and simplify ATE projects with long life cycles. With ATEasy, test applications are faster to generate and easier to maintain.

ATEasy includes a complete test development suite as well as a test executive specifically designed for test applications. The ATEasy development environment combines the ease of Microsoft Visual Basic and the flexibility of Microsoft Visual C++ — a complete object-oriented, 32-bit Windows programming environment. Users of these programs are able to pick up on ATEasy quickly, while first time users not familiar with these programs can use the Application Wizard to generate applications quickly. Application components are displayed in an easy-to-browse tree view that serves as a basis for the application specific components.

FEATURES

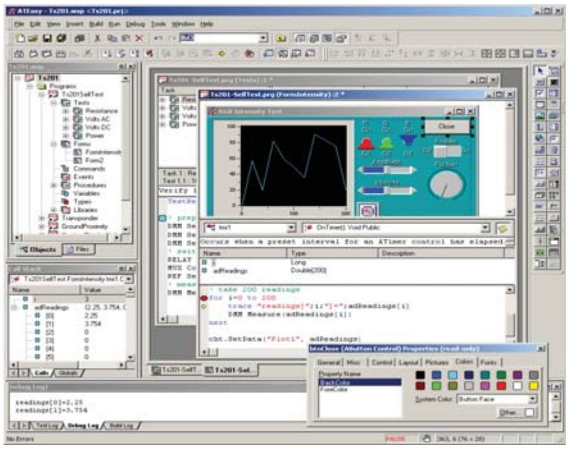
Test engineers build their test application from components modeled after real-world test systems. These components include a System, Drivers, Programs, Tests, Commands, and more. ATEasy provides a streamlined, easy-to-follow framework and wide array of features that enable the user to create re-usable components. The result is a test application that is faster to generate and easier to maintain.

Rapid Application Development (RAD)

ATEasy's application framework consists of well-organized components that allow engineers to partition and organize their test code during development. When debugging and validating, engineers can use these components to quickly isolate problems. Once a modification is applied, smaller portions of the application code can be executed independently without running the entire application, which provides rapid development cycles.

ATEasy Compiler

The ATEasy compiler is extremely fast. During debugging, ATEasy's Just-In-Time Compiler compiles only the necessary code as required. Once debugging is complete, the Build command creates an executable file.



Occurs when a preset interval for an ATimer control has elapsed

Name	Type	Description
i	Long	
adReadings	Double[200]	

```
' take 200 readings
for i=0 to 200
    trace "readings[";i;"]=";adReadings[i]
    DMM Measure(adReadings[i])
next

cht.SetData("Plot1", adReadings)
```

```
readings[0]=2.25
readings[1]=3.754
```

The ATEasy IDE

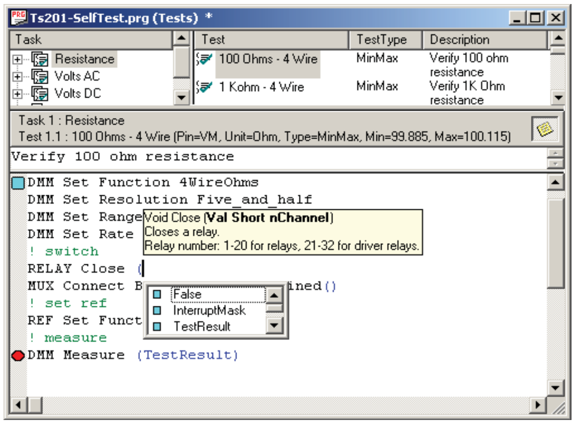
ATEasy 5.0

Programming

ATEasy applications can be created using menus or by typing commands directly. With menu commands, users can insert driver commands, procedure calls, and even flow control statements with a few clicks of the mouse. If the users choose to type, ATEasy's code completion tools provide suggestions on completing the unfinished statements. Code completion tools include Parameter suggestion and Command completion.

ATEasy also provides tool tips that describe the function call and arguments to the parameter that you are about to type. ATEasy displays information about symbols (variables, procedures, commands, etc.) when users hover over the symbol with a mouse.

ATEasy includes a Form Editor very similar to the Microsoft® Visual Basic™ form editor. It provides similar sets of forms types, menus, controls, and event programming. ATEasy's forms can accommodate any system-installed ActiveX controls, offering extensibility and flexibility to use hundreds of ActiveX controls and components available from third party vendors.



Task
Resistance
Volts AC
Volts DC

Test	TestType	Description
100 Ohms - 4 Wire	MinMax	Verify 100 ohm resistance
1 Kohm - 4 Wire	MinMax	Verify 1K Ohm resistance

Task 1 : Resistance
Test 1.1 : 100 Ohms - 4 Wire (Pin=VM, Unit=Ohm, Type=MinMax, Min=99.885, Max=100.115)

```
Verify 100 ohm resistance
```

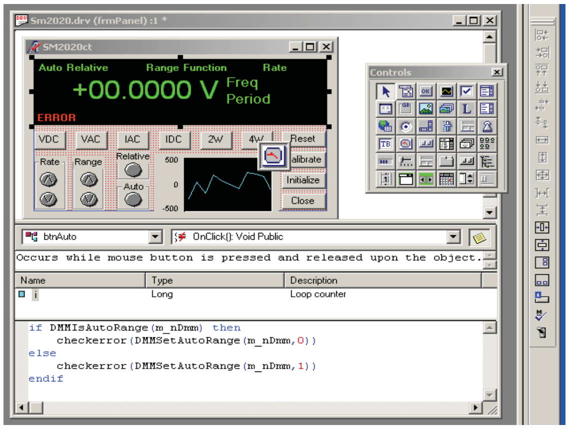
```
DMM Set Function 4WireOhms
DMM Set Resolution Five_and_half
DMM Set Range
DMM Set Rate
! switch
RELAY Close (
MUX Connect B                    ined()
! set ref
REF Set Funct
! measure
DMM Measure (TestResult)
```

Void Close (**Val Short nChannel**)
Closes a relay.
Relay number: 1-20 for relays, 21-32 for driver relays.

ATEasy's code completion tools speed test application development

Interface and Test Application Programming Model

ATEasy provides an event-driven approach to user interface programming for test applications. While a form is displayed for user interface to the test application, ATEasy test programs run in parallel, relieving the from "pumping" windows periodically as you would do in conventional user-interface programming. When the user clicks a button or selects a menu, an event is generated that interrupts the test program to handle and execute the event code. For time-critical applications, forms can be created in separate threads; this allows tests to run 'isolated' from your user interface events, with little or no effect on test timing.



```
Occurs while mouse button is pressed and released upon the object.
```

Name	Type	Description
i	Long	Loop counter

```
if DMMIsAutoRange(m_nDmm) then
    checkerror(DMMSetAutoRange(m_nDmm,0))
else
    checkerror(DMMSetAutoRange(m_nDmm,1))
endif
```

Build user forms quickly and easily in ATEasy

Multithreading

ATEasy provides full support for Windows®' multithreading model, which allows users to execute multiple code segments simultaneously. Synchronization objects such as semaphore and events allow users to synchronize thread execution to protect the application's resources from re-entrance. ATEasy's robust multithreading model allows users use any user-interface objects or ActiveX controls from any thread without any special programming.

Exception and Error Handling

ATEasy provides a unique approach to error and exception handling. Errors are generated by the **error** statement, run-time, or instruments communication. Once an error occurs it can be trapped and handled locally using the **try-catch** statement or at the module level using **OnError** module event. These minimize the developer's efforts on placing error-handling code throughout the application. When an error is generated, the application can decide whether to **ignore** or **retry** the statement that caused the error. It can also abort the application, reset, or re-throw the error, so other modules or ATEasy's default error handling can handle the error.

Instrument Drivers

ATEasy supports thousands of VXI Plug-and-Play, Function Panel (.fp files) or IVI instrument drivers from vendors such as Agilent Technologies and National Instruments, these instrument drivers can be import to ATEasy and use as is. ATEasy is also supplied with the full set of the currently available IVI class drivers, many other drivers for Geotest and non-Geotest instruments are also supplied. LabView Virtual Instruments (.vi files or VI librareries .llb files) can be used from ATEasy. ATEasy provides various tools for rapid creation and debugging of instrument drivers. These tools include I/O Tables that contain steps used to communicate and control instruments over various bus types (e.g., GPIB). ATEasy's internal

ATEasy 5.0

library also contains many functions that can control instruments (e.g., GPIB, RS232, TCP/IP, VXI, PC IO, memory functions, etc.). A communication monitor window shows the data sent or received from the instrument and can be used to identify and debug these instrument drivers.

Self-Documentation

ATEasy's user-defined commands allow users to create plain English-like Command statements, such as:

```
DMM Set Function VDC
DMM Measure (TestResult)
```

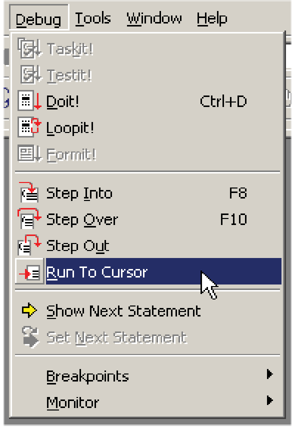
Statements such as these can control instruments or test systems. The resulting test code resembles the Test Requirement Documentation (TRD). ATLAS language users will find Command statements similar to ATLAS statements.

Integration With Microsoft® Source Safe

ATEasy provides seamless integration with MicrosoftSource Safe. Users can Check In, Check Out, or Add Files directly from ATEasy to Source Safe. Different versions of any ATEasy file can be compared to see what, when, where, and by whom files were modified.

Debugging

The ATEasy debugger provides the versatility of conventional software development tools. For example, while the application is paused, users can Step In, Step Out, and Step Over when executing code. Users can set breakpoints, Run to Cursor, and/or Set the Next Statement from which to run. ATEasy has windows that show Call Stack, Local Variables, and Watch; while hovering on variable in the code window, ATEasy displays its value.



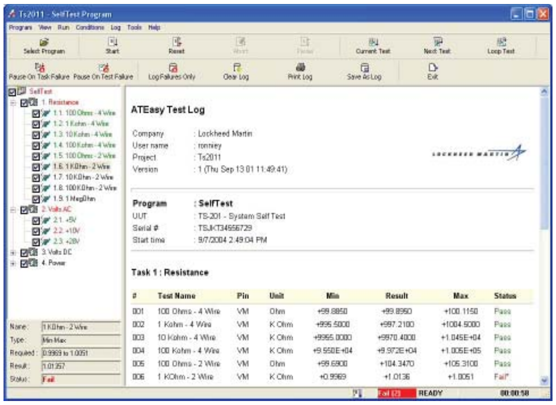
Debug Menu

ATEasy also has specialized instrument control debugging features such as the Communication Monitor that shows you what was sent or received from the bus to control the instrument. Other commands such as Doit! and Loopit! let execute selected or typed code to verify driver commands or procedures without executing the entire application.

The ATEasy debugger can execute a Test or a Task without executing the whole application, saving you the time needed to run the whole application until the required test is reached. Additional test program debugging features include: Skip Task/Test, Set Next Task/Test, Run to Task/Test, repeat Current Task/Test, Taskit!, TestIt!, and run conditions such as: Continuous, Task By Task, Test By Test, Stop on Failure, and more.

Built-In Customizable Test Executive

ATEasy comes with a ready-to-run Test Executive module. Once the Test Executive is used, the application has a complete test executive user interface that allows users to select and run a test program, select test to include or exclude from running, debug, view, and print test logs. The Test Executive module can be customized to accommodate any requirements. This plug-in architecture allows organizations to re-use the customized test-executive they have developed.



Name: 1 KOhm - 2 Wire
Type: Min Max
Required: 0.9969 to 1.0051
Result: 1.01357
Status: Fail

ATEasy Test Log

Company : Lockheed Martin
User name : ronniey
Project : Ts2011
Version : 1 (Thu Sep 13 01 11:49:41)

Program : SelfTest
UUT : TS-201 - System Self Test
Serial # : TSJKT34556729
Start time : 9/7/2004 2:49:04 PM

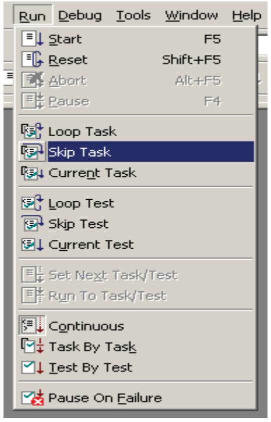
Task 1 : Resistance

#	Test Name	Pin	Unit	Min	Result	Max	Status
001	100 Ohms - 4 Wire	VM	Ohm	+99.8850	+99.8950	+100.1150	Pass
002	1 Kohm - 4 Wire	VM	K Ohm	+995.5000	+997.2100	+1004.5000	Pass
003	10 Kohm - 4 Wire	VM	K Ohm	+9955.0000	+9970.4000	+1.045E+04	Pass
004	100 Kohm - 4 Wire	VM	K Ohm	+9.550E+04	+9.972E+04	+1.005E+05	Pass
005	100 Ohms - 2 Wire	VM	Ohm	+99.6900	+104.3470	+105.3100	Pass
006	1 KOhm - 2 Wire	VM	K Ohm	+0.9969	+1.0136	+1.0051	Fail*

ATEasy's Test Executive

Testing Sequences

A Profile plug-in module allows you to create profile files to define custom testing sequences. Each profile contains a list of steps that specify the program, Task, or Test to run from the application and the number of times the step should be executed. The profile is fully integrated with the Test Executive module and can alternatively be used as a stand-alone module.



Run menu

ATEasy Modules

ATEasy applications are created from a project file that contains the application modules files. ATEasy modules are modeled after a test system. Three types of modules are available:

- Program Module – contains the application
- System Module –lists the application drivers and their

ATEasy 5.0

configuration. For example: a GPIB instrument driver configuration may have its address, terminator, etc.

- Driver Module – contains the commands and functions that are required to operate an instrument as well as the interfaces it supports (e.g., GPIB, RS232, etc).

Each ATEasy module contains sub-modules such as Forms (for user interface), Commands, Procedures, Variables, Types and Libraries.

Tests and Tasks

The Program module contains the necessary tests required for UUT. A Task consists of a group of Tests, which test the same block or logical unit in the UUT. Each Test contains code and various properties such as Name, Type, Pin, Unit, Result, Status, etc. Various test types are built-in, such as Min-Max (test results must fall within required Minimum and Maximum values), and are used to automatically generate the test log.

APPLICATIONS

- Functional Test Systems
- Instrumentation Systems
- Avionics
- Aerospace
- Military
- Medical
- Wireless Telecom
- Semiconductor
- Automotive
- Electro-Optical
- Process Control

SPECIFICATIONS

EDITING

- Drag/drop objects
- Cut/copy/paste/delete
- Find/replace across multiple modules for object's name, description, or code
- Merge objects
- Undo/redo (unlimited)
- Context menu for all objects

EDITING WINDOWS

- Build log
- Workspace window
- Test Log (HTML or text)
- Properties window with multi-tab (modeless)
- Debug log
- Variables window
- Workbook user interface module
- MDI document windows with Tree/ Object views

IDE CUSTOMIZATION

- Docked/Float/MDI windows
- Keyboard keys
- Menu
- Tool menu
- Toolbars

CODE GENERATION/EDITOR

- Syntax highlighting
- Insert Flow control menus
- Insert Commands cascading menus
- Auto Type information
- Auto Parameter information
- Auto Parameter suggestion
- Auto List commands
- Auto List members
- Insert Symbol dialog
- Bookmarks
- Font, tabs, auto indent

FORM

- Drawing
- Event programming
- ActiveX controls
- Menu bar, tool bar, and context menus
- Automatic sizing of control/form
- Container controls support
- Tool tip, What's This Help
- Win Help/HTML Help
- Procedures, Variables, Events
- MDI frame, MDI Child, and Normal Forms

FORM EDITOR

- Align controls
- Test Form
- Grid
- Space evenly controls
- Auto arrange controls
- Check duplicate mnemonics
- Same size controls
- Lock controls
- Margins
- Center across the form
- Tab order controls

COMPILE/BUILD

- Checkit!
- Check All
- Check Module
- Execute
- Build
- Re-Build
- Stop Build
- Show Error

TEST LEVEL DEBUGGING

- Loop Task/Test
- Current Task/Test
- Run to Task/Test
- Stop on Failure
- Continuous/Task By Task/Test By Test
- Pause on Task/Test Failure
- Prompt on Pause Failure
- Skip Task/Test
- Set Next Task/Test
- Taskit!
- Testit!
- Programit!
- Repeat Run
- Log Failures Only

SOURCE SAFE INTEGRATION

- Check In/Out
- Show Status/Properties
- Compare Files
- Add Files/Project
- Show History

SOURCE LEVEL DEBUGGING

- Reset
- Pause/Continue
- Set Next Statment
- Show Next Statement
- Step Over
- Step Out
- Step Into
- Abort
- Run to Cursor
- Breakpoints
- Doit!
- Formit!
- Loopit!

DEBUG WINDOWS

- Calls Stack/Locals with a separate tab for module variables and change value
- Watch window with expression evaluator
- Debug window for executing code in immediate mode without adding the code to a test program
- Monitor (communication) window with interface source filtering

ATEasy 5.0

SPECIFICATIONS (CONT'D)

INSTRUMENTATION INTERFACES

- GPIB: Computer Boards, Keithley/MetraByte/CEC, Agilent/HP, National Instruments
- VXI: National Instruments — MXI
- COMM: Any Windows-compatible serial and infrared port
- File/Device
- WinSock: Client/Server, UDP/TCP
- ISA, PCI, PXI, and PC-based instruments

STATEMENTS

- #iidef
- Abort
- Continue
- Error
- Exit
- Exitloop
- ExitProgram
- ExitTask
- Exit Test
- For Next
- GoTo
- If - Else
- Ignore
- Load
- Loop
- Pause
- Print
- ReDim
- Repeat - Until
- Reset
- Retry
- Return
- Run
- Select - Case
- Task
- Test
- tic type conversion
- Trace
- Try - Catch
- Unload
- While

DATA TYPES

BASIC TYPES

- Any
- Bool
- Byte
- BString
- Char
- Currency
- DateTime
- Double
- DWord
- Float
- Long
- Object
- Procedure
- Short
- String
- Variant
- WChar
- Word

USER TYPES

- Enum
- Structure
- Typedef

COM

- Create Object/Get Object
- Display type libraries content
- Early and Late Binding
- Load type libraries
- Objects can be used from any thread
- Support method, properties, and events
- Support for ActiveX controls

.NET

- Import .NET assemblies and use Components

DLLS

- Automatic type conversion
- CDecl or StdCall
- Calling and using DLL functions by name or ordinal
- Support for callback or procedure parameter
- Import .h files for function and type definition

EXTERNAL INSTRUMENTS DRIVER

- Import Lab Windows/CVI or Function Panel (.fp) drivers
- LabView VI (.vi/.llb) calling and browsing

IVI DRIVERS

- Drivers for DCPwr, Dmm, FGen, PwrMeter, RFSig Gen, Scope, SpecAn, and Switch
- Full support for IVI-C and IVI-COM

TEST EXECUTIVE

- Automatic integration to test program/application
- Automatic log (HTML or text)
- Customized UI and functionality
- Fault Analysis/Directory
- Networked users/groups-based login, privileges, and customization
- Optional touch panel user interface
- Test sequencing

BUILT-IN CONTROLS

- AButton
- AChart
- ACheckBox
- AComboBox
- ACommon-Dialog
- AGroupBox
- AImage
- AImageList
- ALable
- AListBox
- ALog
- APanel
- ARadioButton
- AScrollBar
- ASlider
- AStatusBar
- ASwitch
- ATab
- ATimer
- AToolbar

OPTIONS

- Directories
- Interfaces
- Log
- Source Control
- Tests
- Text Editor
- Workspace

UI CLASSES

- AAxis
- AClipboard
- AControl
- AFont
- AForm
- AImageListImage
- AMenu
- APicture
- APlot
- AStatusBarPane
- ATabPage
- AToolBar-Button
- AThumb

CLASSES

- ACriticalSection
- ADriver
- AEvent
- AMutex
- App
- AProgram
- ASemaphore
- ASystem
- ATask
- ATest
- AUser
- AUsers

PROCEDURE GROUPS

- ActiveX
- DDE
- Driver/Info
- File System
- GPIB
- Interfaces
- Interrupts
- Log File Information
- Math
- Misc
- Multi-Threading
- Port/Memory I/O
- Serial Communication
- String Manipulation
- Variants
- VXI
- WinSock

HELP/DOCUMENTATION

- Getting Started Manual
- User's Guide Manual (on-line)
- Preference/Programming Language Manual (on-line)
- HTML-based help with TOC, Index, Search
- Context sensitive help for dialogs and keyword

OTHER

- Module events
- Multi-Threading
- Exception Handling (try-catch, error, OnErro)

Note: Specifications are subject to change without notice

Geotest
Marvin Test Systems,Inc.

ATEasy 5.0

ORDERING INFORMATION

GT8000-xx	ATEasy single user software package with 1-year Subscription and Support
GT8003-xx	ATEasy with 1-year Subscription and Support and 3-day training at Geotest
GT8320-2Y-xx	ATEasy with 2-year Subscription and Support
GT8320-3Y-xx	ATEasy with 3-year Subscription and Support
GT8021-UG-xx	ATEasy Upgrade (from 3.0 to 4.0 or 5.0) with 1-year Subscription and Support
GT8022-UG-xx	ATEasy Upgrade (from 3.0 to 4.0 or 5.0) with 2-year Subscription and Support
ATEZ-LPT/USB	Replace any type of ATEasy license with another (i.e.,-KP to -NL, -SL to -KY, etc).
TRAINING	
ATEASY-DS	3 days ATEasy training at Geotest (Irvine, CA) for 1-3 persons. Call for larger groups.
ATEASY-DS5	5 days ATEasy training at Geotest (Irvine, CA) for 1-3 persons. Call for larger groups.
ATEASY-DS2	On-site, 3 days ATEasy training seminars for 1-3 persons. Call for larger groups.
ATEASY-DS25	On-site, 5 days ATEasy Quickstart™ training seminars for 1-3 persons. Call for larger groups.
SUPPORT AGREEMENTS	
GT8020	1-year Subscription and Support for ATEasy (requires continuous agreement)
GT8020-24	2-year Subscription and Support for ATEasy (requires continuous agreement)
GT8020-34	3-year Subscription and Support for ATEasy (requires continuous agreement)

Note: Replace -xx with -SL for Software License, -KP for Parallel Key, -KU for USB Key, or -NL for Network License (Network License requires a minimum of three licenses).

Geotest
Marvin Test Systems,Inc.