The Ensemble™ is Aerotech’s next-generation, multi-axis controller for moderate- to high-performance applications with high speed communication through 10/100 Base T Ethernet or USB interfaces. It offers easy to use, affordable multi-axis motion programming for laboratory experimentation, production testing, or advanced OEM automated manufacturing systems.

**Versatile, Stand-Alone Multi-Axis Control**

With the Ensemble stand-alone controller, up to ten axes of synchronized motion are offered in a distributed network of panel-mounted drives. This is excellent for applications where drives must be embedded into a machine at various locations or where panel space is at a premium. For desktop and rack-mount installations, the Ensemble is offered in a stand-alone 6-axis unit with integrated drives. Three additional panel mounted linear or PWM drives may be added externally to the six-axis unit for up to nine axes of motion control. The Ensemble can control any Aerotech brushless, brush, or stepper motors or stages in any combination. The simple parameter interface also allows easy integration to third-party motors and stages. The controller encoder interface includes TTL quadrature input or analog encoder input. Multiple Ensemble controllers can be controlled from a single Windows® PC through Ethernet or USB, allowing many more than ten axes of motion to be operated from one host PC.

**Powerful and Intuitive Programming Functionality**

Unlike most controllers on the market today, there is no need to understand a cryptic command set to generate motion. The intuitive interface allows a user to begin programming immediately. Ensemble online help further simplifies writing motion programs and includes many functional examples that can be easily modified for customer applications.
Ensemble DESCRIPTION

The Ensemble Integrated Development Environment software offers a graphical user interface in Windows® featuring an intuitive Program Editor, Variable Output window, Compiler Output window, Task State monitor, Network Explorer, and Solution Explorer. This interface enables users to easily monitor all aspects of their positioning system, no matter how complex. The Axis Control and Diagnostic screen interfaces are further supplemented by a fully functional autotuning utility that minimizes startup time and allows easy optimization of motion axes. System diagnostics are easily read from the interface. The Windows-based remote software package is included with each unit, which allows the user to upload/download programs, modify parameter files, and analyze motion with Aerotech’s advanced graphical tuning package, all from the convenience of a remote PC.

Each Ensemble axis uses the processing power of a 225 MHz double precision, floating-point DSP to offer exceptional performance in a variety of applications, including point-to-point motion, linear and circular interpolation, multi-axis error correction, 2D error mapping, direct commutation of linear and rotary brushless servomotors, and on-board servo autotuning. High-speed interrupts and data logging capabilities provide a real-time link to external systems. The Ensemble also offers high-speed position latching capability and optional single-axis PSO (Position Synchronized Output). Whether the requirement is simple point-to-point motion or complex velocity profiled contours with output on the fly, Ensemble ensures peak performance for critical applications.

Flexible Drives
Because the Ensemble can control many different types of motors, users have excellent flexibility in their motion system design. High accuracy linear motor air-bearings can be controlled from the same controller running lower precision drives with servo or stepper motors. Parameters are easily reconfigured for these various motors and feedback devices, so customers can adapt to changing system needs.

Allen-Bradley Interface
Combine proven PLC with proven motion control for easier integration, startup, and maintenance of medium- and high-end automation projects. The Aerotech EtherNet/IP™ interface enables AB PLCs (MicroLogix, CompactLogix™, or ControlLogix) to be integrated directly with the Ensemble. Motion can be directly programmed in the RSLogix 5000 environment or separate programs can be written on the controller and triggered from the AB PLC. Aerotech has two interfaces: ASCII and Register. Choose the PLC, motion controller, and interface that best fits your application needs.

EPICS Drivers
Each Ensemble installation includes full compatibility with the EPICS open source distributed control system. EPICS is used worldwide at leading light source (synchrotron) facilities and other government laboratories, allowing Ensemble to seamlessly integrate into applications at all major research institutions.

Enhancing a Legacy of Success
Although Ensemble is envisioned as a general-purpose, stand-alone controller, it carries forward a legacy of success built from Aerotech’s hugely successful A3200 and Soloist™ controllers. It offers enhanced capabilities that will make it an ideal choice for many aggressive motion control applications. The Ensemble motion control architecture builds upon the Soloist intuitive graphical user interface, while offering advanced features appropriate for multi-axis control. Ease of use is further improved with pre-coded LabVIEW® VIs, AeroBASIC™ programming functionality, .NET tools for C#, VB.NET, and C++/CLI and MATLAB® compatibility.
The Ensemble can be used to control multiple axes (from one to ten axes) of both brush and brushless drives and motors.
**Allen-Bradley EtherNet/IP Interface to Aerotech Ensemble**

The Aerotech Ethernet/IP interface enables AB PLCs (MicroLogix, CompactLogix, and ControlLogix) to integrate directly with the Ensemble motion control solutions. Motion can be directly programmed in the RSLogix 5000 environment or separate programs can be written on the controller and triggered from the AB PLC. Aerotech has two Ethernet/IP interfaces: ASCII and Register. Choose the PLC, motion controller, and Ethernet/IP interface that best fits your application needs.

The Allen-Bradley code snippets provided here are written in the graphical “relay ladder logic” syntax. Allen-Bradley also supports function block and structured text programming languages.

**ASCII COMMAND INTERFACE**

The ASCII command interface can be used to send ASCII text strings to the Ensemble and perform a set of actions such as commanding motion or retrieving diagnostic information. Our vendor-specific EtherNet/IP ASCII command interface object extends this functionality across EtherNet/IP.

The power of the ASCII command interface lies in its simplicity and ease of use. A text string is formed using an AeroBASIC™ command, followed by an End-Of-String (EOS) character. ASCII response data indicates whether or not the command was successfully executed. An ACK character is sent to indicate success, an NAK character is sent if there is a command error, and a FAULT character is sent if there is a task error. For commands that expect return data, the response character is followed by the return data, which is terminated by the EOS character. The EOS, ACK, NAK, and FAULT characters are configurable via Ensemble drive parameters.

To send the ASCII command from the PLC (programmed using Rockwell RSLogix) to the Ensemble, an MSG block is used (Figure 1). The EtherNet/IP message can be easily configured from within the RSLogix software as shown in Figure 2.

ASCII Text String (specified in the red box in Figure 2) is an RSLogix “tag”. The value of this tag is the ASCII command that will be sent to the Ensemble. The example in Figure 3 shows the command “HOME X Y” as it would be configured in RSLogix.

When the MSG block is activated, the ASCII command is sent to the Ensemble via the EtherNet/IP protocol. The Destination field of the message configuration can be used to specify a local tag for the ASCII command’s return data.

**Ensemble Code Snippet (AeroBASIC) for ASCII Interface**

For this interface (see Figure 4), the user does not need to write any AeroBASIC code on the Ensemble. The

---

*EtherNet/IP™ is a trademark of ODVA.*
*CompactLogix™ and RSLogix™ are trademarks of Allen-Bradley.*
Ensemble automatically receives and processes the commands. This functionality is built into our EtherNet/IP module.

**REGISTER INTERFACE**

Our vendor-specific EtherNet/IP register interface object allows the Ensemble’s built-in integer and double registers to be accessed via EtherNet/IP. This provides a flexible, general-purpose interface to the Ensemble that can be adapted to many different applications. Data consistency is guaranteed internally, so there is no need for concern when accessing these registers simultaneously via EtherNet/IP and AeroBASIC programs on the controller.

The Rockwell software is configured in very much the same way as the ASCII command interface. In this case, the message source (shown in red in Figure 5) is a data array.

The message configuration in Figure 5 is for the “Write Single Register” service. That service requires two pieces of data: the number of the register to write, and the value to be written. Therefore, we configure the message source tag as shown in Figure 6.

The value 100 is the register to write to, and the value to be written is 7. When the MSG block is activated, the register query is sent to the Ensemble via the EtherNet/IP protocol.

As you can see, this is a straightforward data interface from the PLC to the Ensemble. In addition to writing a single register, the interface also supports writing multiple registers in one message and reading single or multiple registers. This example was interfacing with an integer (32-bit) register, but the interface supports the same functionality with double-precision (64-bit) floating point values.

**Ensemble Code Snippet (AeroBASIC) for Register Interface**

To make use of the data that is being transferred to and from the PLC, the user can write an AeroBASIC program to respond to incoming register transfers, as well as write outbound register data. This combination of EtherNet/IP communication with the power and flexibility of AeroBASIC allows for the implementation of many unique applications. For example, the following AeroBASIC code snippet shows how the register interface can be used to control the Ensemble timebase value (set by the AeroBASIC TimeScale command).

**HEADER**

```plaintext`
`// The PLC writes timebase values to IntegerRegister 101
Define timebaseRegisterIndex 101
END HEADER`
```

**DECLARATIONS**

```plaintext`
`// Declare a global variable named "timebase"
Global timebase as Integer
END DECLARATIONS`
```

**PROGRAM**

```plaintext`
`// initialize the timebase register to 100% speed
RegS IntegerRegisters, timebaseRegisterIndex, 100
// ... do work in AeroBASIC ...
// Change timebase to value sent from PLC
Call GetTimebase()
TimeScale timebase
// ... continue working ...
END PROGRAM`
```

**FUNCTION GetTimebase() as void**

```plaintext`
`// Read the value from IntegerRegister 101 into the variable "timebase"
timebase = RegS(IntegerRegisters, timebaseRegisterIndex)
END FUNCTION`
```