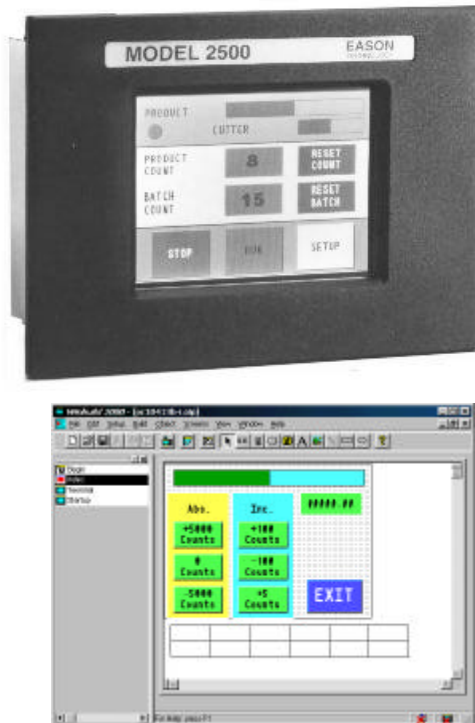

COMMUNICATING TO COMPUMOTOR 6K MOTION CONTROLLER



Rev 1.0
P/N: 50-00186-01

This document contains:

1. Eason WinBuild 2000 & Compumotor application development methodology.
2. Quick explanation of hardware & software setup.
3. How to arrange the driver to communicate variable data.
4. How to send/receive commands & data between Eason & motion controller.
5. Coordinating Eason-Compumotor I/O.

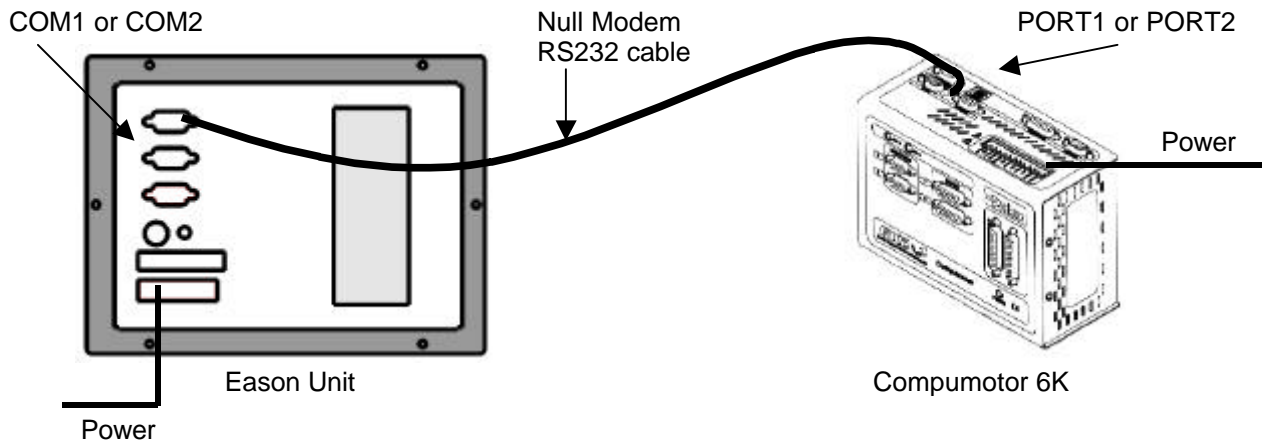
Suggested Application Development Process:

- 1. Load and review the simple 6K demo program with WinBuild2000**
- 2. Connect to the Compumotor 6K using the Compumotor Terminal Software.**
- 3. Tune your system using the Compumotor development software, and create your motion control program. Most aspects of motion critical tasks should be written and debugged here- feedback, control, I/O, etc.**
- 4. Identify variables in the 6K program you would like the Eason HMI to display/monitor/modify.**
- 5. Now expand your initial simple Eason 2000 program- use the Eason Compumotor driver to set up the 6K variables you wish to display/monitor/modify.**
- 6. Create the rest of your HMI application with WinBuild 2000. Use the initial program as a base, and add your HMI and control tasks.**

It is important to keep in mind that the Compumotor motion controller is just that- an excellent motion controller- thus most motion control aspects should be left to the 6K .PRG motion program to handle. Positioning, interpolation, & time-critical motion tasks should be programmed directly into the 6K.

The Eason 2000 Family product is an excellent HMI product- let the Eason handle the data storage, operator input, calculations, decision making, and communications in your application.

Optimal usage of both components will create an efficient, elegant solution.



Hardware Setup:

The Compumotor 6K motion controllers communicate to the Eason 2000 Family products via serial connection. All 2000 Family products have at least two ports available for RS232 communication, and those with expansion options can be fitted with more serial connections.

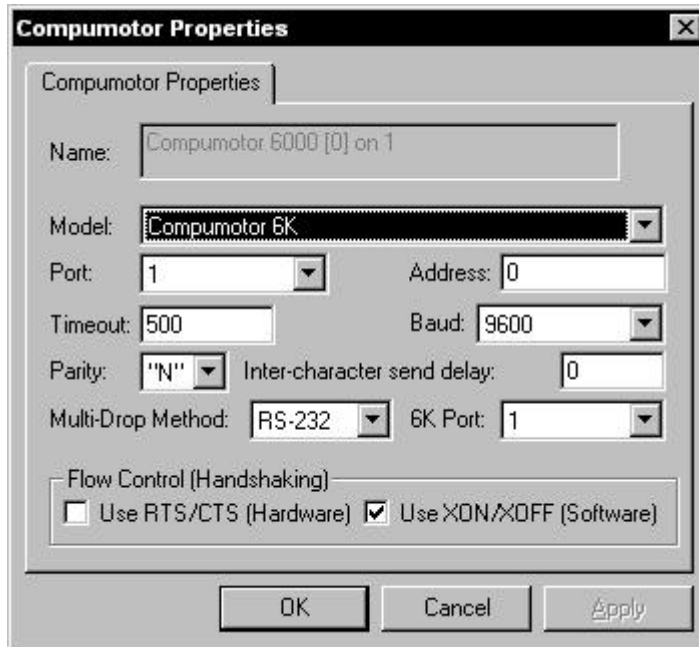
Even with a RS232 serial connection at 19200 baud, communication speed is still limited by the 6K's ability to process serial communications. Transfers are relatively quick, but care should be taken not to overload the mechanism with too many requests for data at a time, as the 6K can be slow to respond when overwhelmed or burdened with intense motion tasks.

Hardware connections between the Eason & Compumotor motion controller should consist of just a simple RS232 cable. Either communication port on the Eason is available for use, and typical applications use the PORT1 in the 6K, but can be configured to use PORT2. Default settings of 9600, 8, N, 1 are ok. The 6K driver in the Eason WinBuild 2000 software will send down the subtle commands necessary to the 6K for perfectly synchronous communication.

Software Drivers

Adding the Compumotor driver to your WinBuild 2000 project makes communicating to your motion controller easy.

Add the driver by clicking "SETUP | PROJECT | DRIVERS" and selecting the "External Serial: Compumotor" driver. Click on "Properties" to configure the driver for your model motion controller. Pay close attention to the 6K device address- this is normally "0" but your controller may be set differently.



Once the driver is added, you can set up the driver to monitor data/variables in your 6K motion controller. For instance, if you want to monitor the variable “VAR1” in the controller, just configure a tag as the following:

PLC Contact and Tag Setup					
	Tag Name	Source	Address	Type	Default
1	EasonTagName	Compumotor 6K [0] on 1	VAR1	Real	0

The *Tag Name* is the Eason variable name you will refer to whenever using this value in the Eason. *Source* is the Compumotor driver. The *Address* column you enter the command/variable you wish to read/write to. If you enter in an invalid value for the address, a prompt will appear telling you what a valid entry is for your driver. See “Supported Data Types” in the help system to learn more about this.

If a data display with your Eason tag is used on the screen, the Eason will poll the Compumotor as often as it can to obtain a current value to be displayed on screen. (Often once every screen refresh ~ 20ms)

Conversely, if you enter a new value for a tag, the driver will send that tag back to the 6K.

Sending/Receiving Commands

In addition to the driver providing easy data read/write capabilities, the driver also lets you manually send commands to the 6K, and read responses.

To send a command to the 6K- for instance to run a particular routine in your 6K program, you would use the Pseudocode: COM: PUT COM PORT value.

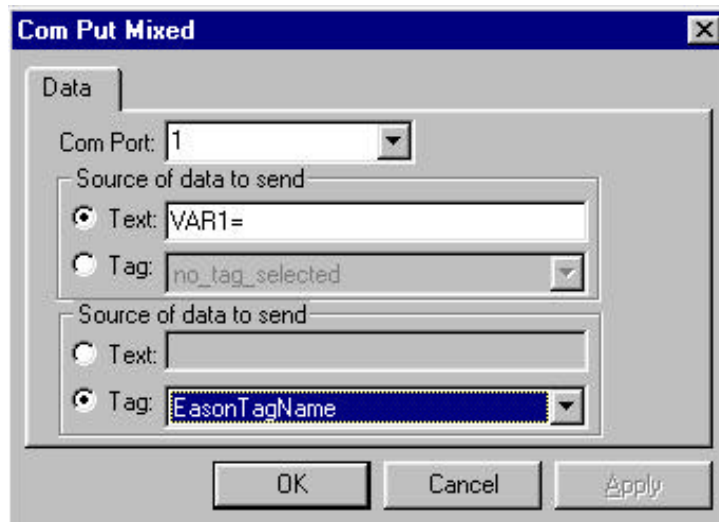
```
Code For Current Object
COM: PUT COM 1 VALUE "RUN BRIAN"
```

This code will send the command "RUN BRIAN" to the 6K to begin the executing the routine "BRIAN."

You can send down most Compumotor 6K: set output bits, jog, begin motion, etc.

You can also use the manual commands to send/receive data using the PUT COM/GET RESPONSE pseudocodes.

If you are using the recipe editor in the Eason 2000- using the manual commands is a convenient way to send that information to the 6K controller after you have selected the record you wish to use. Use the PUT COM PORT VALUE expression VALUE expression pseudocode. That will allow you send a combination of text & numeric variables to the 6K. For example:



This pseudocode will send to the 6K "VAR1=xxx" (Where the 'xxx' is whatever the current value of your tag "EasonTagName" is equal to.)

A Carriage Return terminates each transmission to the 6K, so each line of pseudocode is interpreted as a separate serial transmission to the 6K.

Coordinating Data I/O

It is encouraged to use the on-board 6K I/O for any motion-critical I/O. The Compumotor is designed to handle this quickly and easily. The Eason unit can handle non motion-critical I/O.

Sometimes you will want the Eason unit to know the status of the Compumotor I/O, or vice versa.

To read the status of an I/O point in the Compumotor, you first need to read that point into a 6K variable:

```
VAR1=2IN.1    (Reads binary input 1 of brick to into VAR1)
```

Then, just set up your Eason Compumotor 6K driver to have a tag that reads the address variable 'VAR1'. Whenever you want to know its status, just refer to it by the Eason tag name.