BAAM WCF Communications

BAAM HMI version 1.4.0 added WCF communications to enable 3rd party applications to easily monitor and adjust a running program. It also added a “extra” feedrate override value and extruder override value that are only accessible through the WCF interface. These override values are initialized to 0 whenever a program is loaded or rewound and they are added to the override values that are available in the HMI software. For example if the HMI sets the feedrate override to 105% and the extra WCF override value is 20%, the actual override value used will be 125%. Any positive or negative integer value can be used for the extra override.

The WCF commands are as follows:

string GetValue(int commandID) - This command is used to retrieve a variety of status values from the BAAM. The return value is always a string.

The possible values for commandID are:

0 - Extra feedrate override value
Return string: The current value of the extra feedrate override.

1 - Extra extruder override value
Return string: The current value of the extra extruder override.

2 - Run Status
Return string: 0 - No Program Loaded, 1 - Stopped, 2 - Ready To Run, 3 - Running, 4 - Finished

3 - Cnc Mode
Return string: 0 - Auto, 1 - Jog

4 - Program Name
Return string: The name of the current loaded program

5 - Total Run Time
Return string: The total estimated run time of the program in seconds

6 - Remaining Run Time
Return string: The time remaining in the current program in HH:MM:SS

7 - Remaining Layer Time
Return string: The time remaining in the current layer in HH:MM:SS

8 - Current Layer Number
Return string: The current layer number

9 - Total Layers
Return string: The total number of layers in the program

10 - Dwell Remaining
Return string: The dwell time remaining (in milliseconds) for the current line. This will be non-zero if the current line is executing a G4 or G104.

11 - Positions
Return string: The current position of the X,Y,Z and W axes in inches. Example: “X:0.000 Y:24.000 Z:0.000 W:0.000”

12 - Extruder Speed
Return string: The actual extruder speed in RPM

13 - Line Number
Return string: The current line number of the running program
100 - 199 - Get Global Variable
Return string: The value of the Global Variable with the commandID value. For example, GetValue(150) would return the value of Global Variable #150.

int SetValue(int commandID, string commandValue) - This command is used to change values in the BAAM control.

The possible return values for the SetValue call are:
0 – OK
1 – Invalid Parameter
2 – Internal Error

The possible values for commandID are:
0 - Extra feedrate override value
Sets the value of the extra feedrate override. Ex: SetValue(0, “10”) would set the extra feedrate override to 10%.
1 - Extra extruder override value
Sets the value of the extra extruder override. Ex: SetValue(1, “-22”) would set the extra extruder override to -22%.
2 - Simulate the continue button
This simulates pressing the Continue button on the HMI. If the program is currently executing a G104, the remainder of the dwell will be skipped. If the program is not currently on a G104, this command does nothing. CommandValue is not used for this command.
100 - 199 - Set Global Variable
This command is used to set the value of any of the Global Variables 100 through 199. Ex: SetValue(110, “-3.23”) would set Global Variable #110 to -3.23

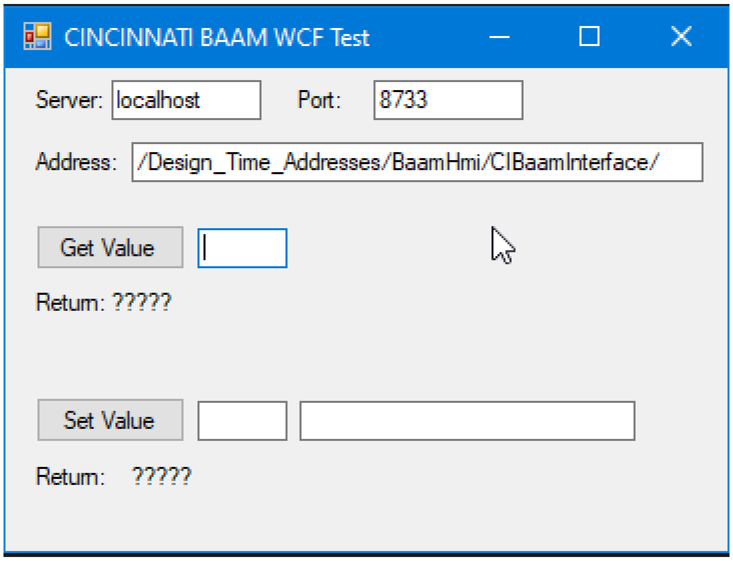
BAAM WCF Communications

Sample Code

To use the BAAM WCF calls, you must include CIBaamInterface.cs (included in the sample project) in your C# .NET project. The following code shows how to create a CIBaamInterfaceClient object and use it to read/write data from the BAAM control.

```
//Specify the binding to be used for the client.
BasicHttpBinding binding = new BasicHttpBinding();

//Specify the address to be used for the client.
EndpointAddress addr = new
    EndpointAddress(@"http://localhost:8733/Design_Time_Addresses/BaamHmi/CIBaamInterface/");

CIBaamInterfaceClient client = new CIBaamInterfaceClient(binding, addr);

string result = client.GetValue(3);   // result will be “0” if in Auto mode
client.SetValue(0, “10”);      //  set the extra feedrate override to 10%
client.SetValue(1, “15”);     //  speed up the extruder by 15%

client.Close();       // close the client when finished.
```

The sample project BaamWCFTest is a simple .NET WinForms project. It can be used to test the WCF connection to the CINCINNATI BAAM HMI. The Port and Address fields should be left at their default values. Server should be localhost if running on the same computer as the BAAM HMI, otherwise it should be the BAAM control’s computer name if you are running the test program on a networked computer.



CINCINNATI BAAM WCF Test

Server: localhost   Port: 8733

Address: /Design_Time_Addresses/BaamHmi/CIBaamInterface/

Return: ?????

Return: ?????